Statistical Pattern Recognition
Assignment 3

Ali Gholami

Department of Computer Engineering & Information Technology
Amirkabir University of Technology

https://aligholamee.github.io
aligholami7596@gmail.com

Abstract

In this paper, we'll review the *parametric* techniques to estimate the *unknown* parameters of data distributions. We'll use, *MLE* and *Bayesian* estimation for *parameter estimation*. Also, we'll delve into the *non-parametric* techniques to estimate the unknown *density* of data distribution. We'll use *Kernel Density Estimation* methods such as *Parzen Windows* and other techniques such as *Histogram* and *k-NN* density estimation.

Keywords. *Parameter Estimation, Density Estimation, Non-parametric Methods, Parametric Methods, Kernel Density Estimation, Maximum Likelihood Estimation, Bayesian Estimation, Histogram Density Estimation, K-NN Density Estimation.*

1 General Maximum Likelihood Estimation

Let x_k, $k = 1, 2, ..., N$ denote independent training samples from one of the following densities. Obtain the Maximum Likelihood estimate of θ in each case.

(a) $f(x_k;\theta) = \frac{x_k}{\theta^2}\exp(\frac{-x_k^2}{2\theta^2})$ where $x_k \geq 0$ and $\theta \geq 0$

(b) $f(x_k;\theta) = \sqrt{\theta}x_k^{\sqrt{\theta}-1}$ where $0 \leq x_k \leq 1$ and $\theta \geq 0$

Solution

(a) Substituting the given density inside the *MLE* equation yields the following results.

$$\hat{\theta} = \arg\max_\theta \{P(D|\theta)\} = \arg\max_\theta \{\sum_{k=1}^{n} \ln P(x_k|\theta)\} \tag{1.1}$$

$$\hat{\theta} = \arg\max_\theta \{\sum_{k=1}^{n} \ln \frac{x_k}{\theta^2}\exp(\frac{-x_k^2}{2\theta^2})\}$$

$$\nabla_\theta l(\theta) = 0$$

where $l(\theta)$ is $\sum_{k=1}^{n} \ln \frac{x_k}{\theta^2} \exp(\frac{-x_k^2}{2\theta^2})$ in this case. Performing the gradient on the given equation yields the following results.

$$\sum_{k=1}^{n} (\frac{-2}{\theta} + \frac{x_k^2}{\theta^3}) = 0$$

The simplified estimate of unknown θ is given below.

$$\hat{\theta} = \sqrt{\frac{\sum_{k=1}^{n} x_k^2}{2N}}$$

(b) Substituting the given density in the Maximum Likelihood method yields the following result.

$$\hat{\theta} = \arg\max_{\theta} \{\sum_{k=1}^{n} \ln \sqrt{\theta} x_k^{\sqrt{\theta}-1}\} \tag{1.2}$$

we can obtain the estimate for the unknown θ:

$$\nabla_{\theta} l(\theta) = 0$$

where $l(\theta)$ is $\sum_{k=1}^{n} \ln \sqrt{\theta} x_k^{\sqrt{\theta}-1}$ in this case. Performing the gradient on the given equation yields the following results.

$$\frac{n}{2\theta} + \frac{1}{2\sqrt{\theta} \sum_{k=1}^{n} \ln x_k} = 0$$

multiplying the whole equation by θ results in the following equation:

$$\hat{\theta} = \frac{n^2}{(\sum_{k=1}^{n} \ln x_k)^2}$$

2 Uniform Maximum Likelihood Estimation

Let x have a uniform density

$$f_x(x|\theta) \sim U(0, \theta) = \begin{cases} \frac{1}{\theta} & 0 \leq x \leq 0 \\ 0 & otherwise \end{cases}$$

(a) Suppose that n samples $D = x_1, x_2, ..., x_n$ are drawn independently according to $f_x(x|\theta)$. Show that the maximum likelihood estimate for θ is $max[D]$.

(b) Suppose that n = 5 points are drawn from the distribution and the maximum value of which happens to be $max x_k = 0.6$. Plot the likelihood $f_x(D|\theta)$ in the range $0 \leq \theta \leq 1$. Explain in words why you do not need to know the values of the other four points.

Solution

(a) Substituting the uniform density function in the Maximum Likelihood method yields the following results.

$$\hat{\theta} = \arg\max_{\theta} \{\sum_{k=1}^{n} \ln \frac{1}{\theta}\} \tag{2.1}$$

This equation can be written as

$$\hat{\theta} = \arg\max_{\theta} \{l(\theta)\}$$

where $l(\theta) = \sum_{k=1}^{n} \ln \frac{1}{\theta}$. Performing a gradient on $l(\theta)$ would give us the Maximum Likelihood Estimate of θ.

$$\sum_{k=1}^{n} \frac{-1}{\theta} = 0 \rightarrow \frac{n}{\theta} = 0$$

thus

$$\hat{\theta} \rightarrow \inf$$

Since $\hat{\theta} \rightarrow \inf$ and $\hat{\theta} \, \epsilon \, \{x_1, x_2, ..., x_n\}$ we'll have:

$$\hat{\theta} = max[D]$$

(b) Since $\hat{\theta} = max[D]$ we can simply plot the diagram as following.

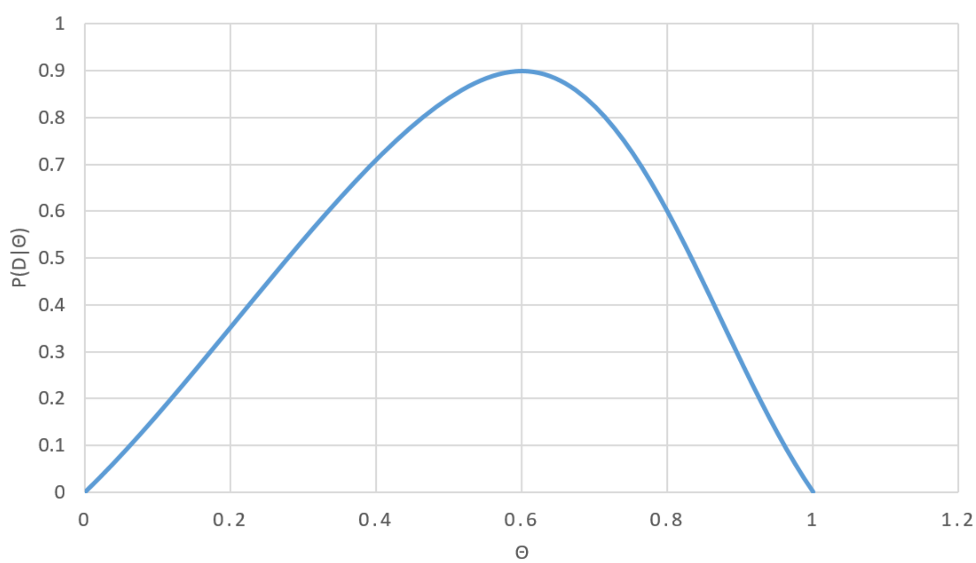

Figure 2.1: Maximum Likelihood Estimation of θ.

The other four points wouldn't have maximized the likelihood $P(D|\theta)$.

3 Density Estimation; Histogram, Parzen Windows and K-NN Method

In this section, we'll be implementing top non-parametric methods for density estimation. We've implemented the 1-D and 2-D Scenario of density estimation in the *src* folder. The experiment results are provided here.

Histogram Density Estimation

Core Definition

In this method, we'll divide the range of available samples to multiple bins. Then we'll count the number of samples in each bin. Let k_i be the number of sample in ith bin and V the size of bins($V = h^d$ where h is the size of the bin in each dimension). The density for the ith bin can be estimated using the following formula. n is the number of total samples.

$$\hat{p}_{(x)} = \frac{k}{n * V} \tag{3.1}$$

Python Implementation

Full implementation with guiding comments can be found in *src* folder. Note that in this implementation i've not used the internal bindings for kernel density estimation of Sklearn.

- (1-Dimensional) — ($\mu = 5$) — ($var = 3$) — ($bin = 2$) — ($|D| = 100$)

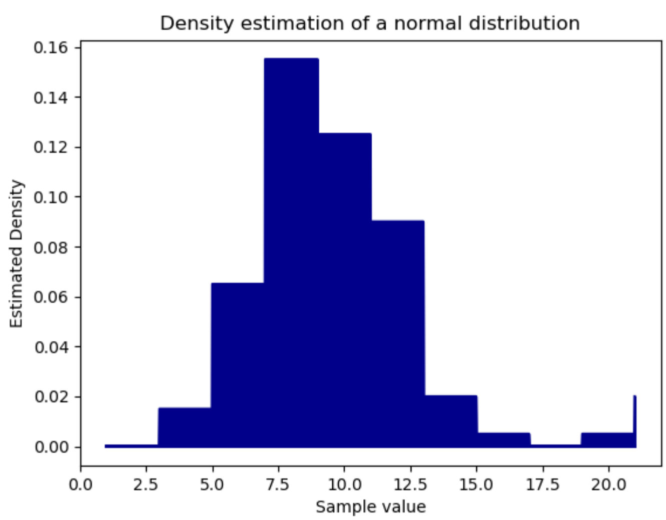

Figure 3.1: 1-D Histogram Density Estimation — Bin Size = 2

- (2-Dimensional) — ($\mu = \begin{bmatrix} 8 & 8 \end{bmatrix}$) — ($var = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$) — ($bin = 1$) — ($|D| = 5000$)

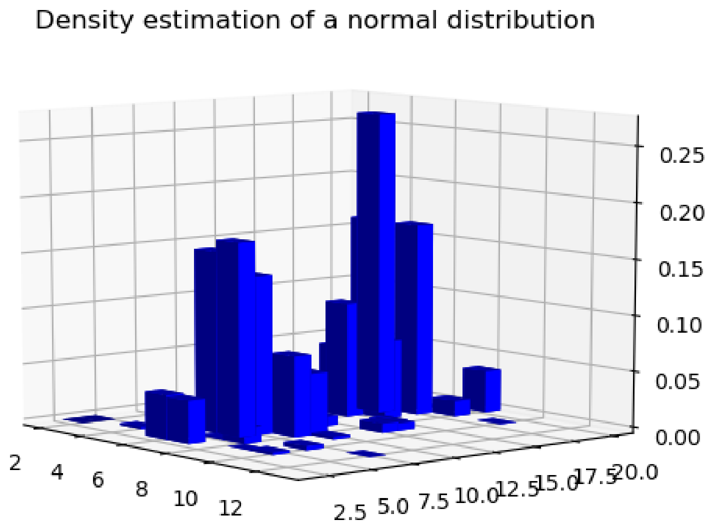

Figure 3.2: 2-D Histogram Density Estimation.

Bin Size Selection Analysis

The results in figure 3.1 was given with a bin size of 2. Smaller bin size result in a sharp and spiky estimation. However, choosing a bigger bin size results in an smoother estimation. As an example, figure 3.3 and 3.4 illustrates this phenomenon.

Density Estimation with Parzen Windows(KDE)

Core Definition

In the method, each of the kernels are represented by $\Phi(x)$. These kernels will be placed on every single sample derived from the main distribution and the estimated density will be represented as following. In this equation, the k stands for the number of samples we have derived from the main distribution. h is called *Bandwidth* and h^d illustrates the Parzen window in a d dimensional space.

$$\hat{p}_{(x)} = \frac{1}{n * h^d} \sum_{i=1}^{k} \Phi(\frac{x - x_i}{h}) \tag{3.2}$$

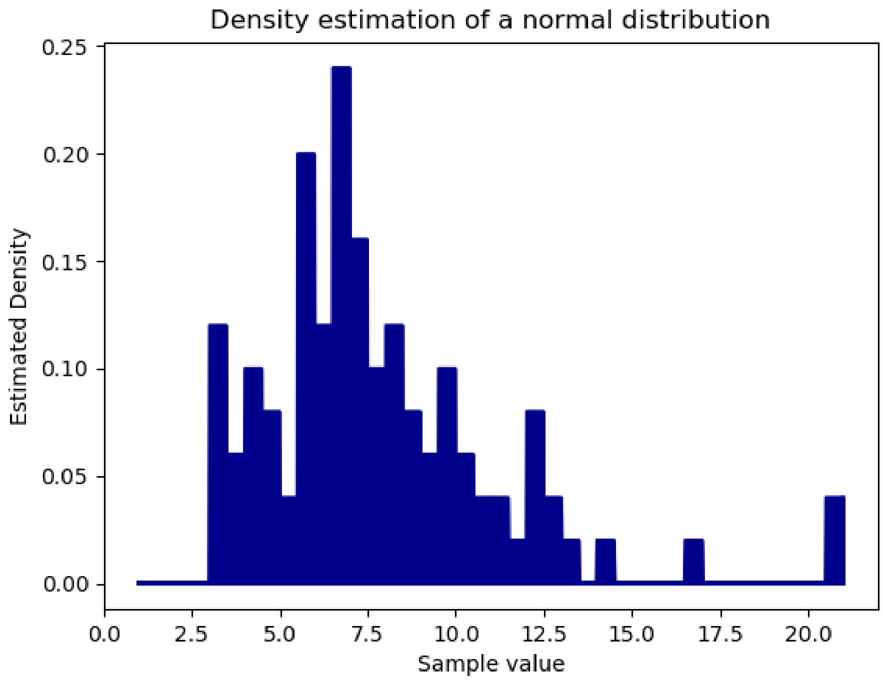

Figure 3.3: 1-D Histogram Density Estimation — Bin Size = 0.5

Python Implementation

This section is implemented using *Numpy, Matplotlib* and *Scipy*. The gray density illustrates the distribution of our data and the blue one represents the estimate using Gaussian windows.

Bandwidth Selection Analysis

The results in figure 3.6 is given with a bin size of 0.9. Smaller bandwidths result in a sharp and spiky estimation. However, choosing a bigger bandwidth results in an smoother estimation. As an example, figure 3.7 and 3.8 illustrates this phenomenon.

Kernel Effect Analysis

Let's change the kernels from *Gaussian* to *Epanechnikov* kernel. The result is given in the figure 3.9. Please visit this link for a complete explanation of the *Epanechnikov* kernel.

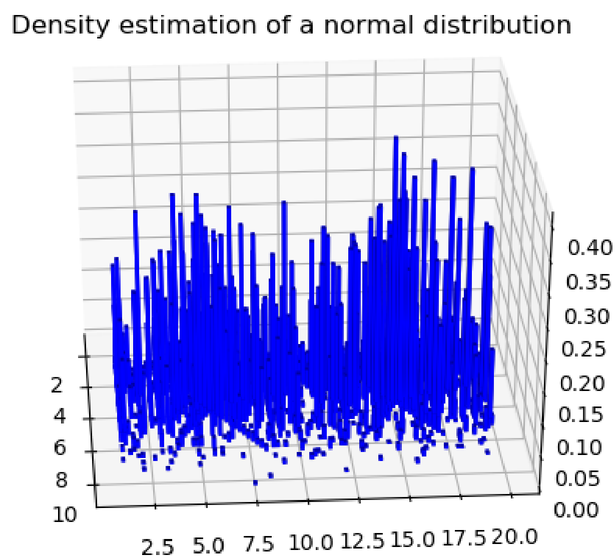

Figure 3.4: 2-D Histogram Density Estimation — Bin Size = 0.2 — Increased Sample Size

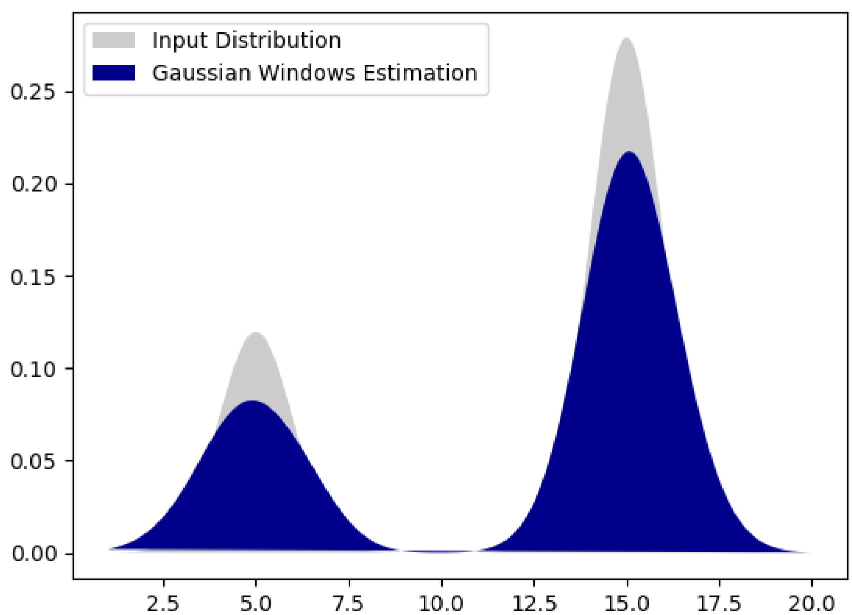

Figure 3.5: 1-D Multi-modal Gaussian Density Estimation using Gaussian Kernels.

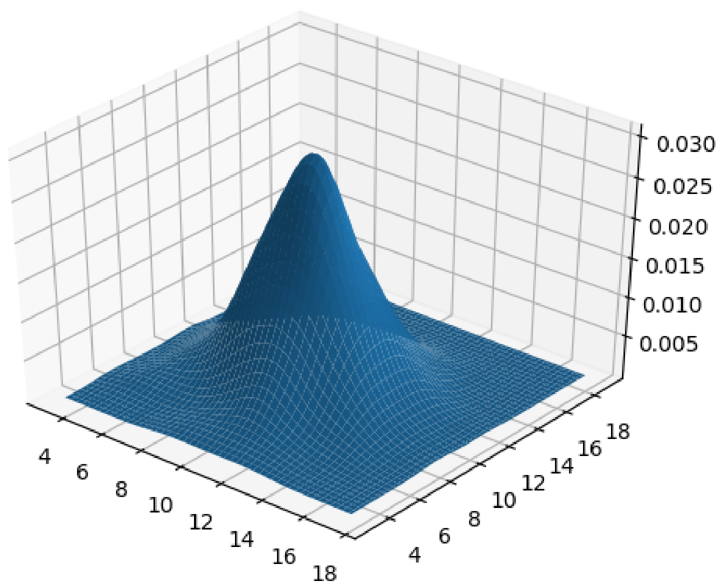

Figure 3.6: 2-D Gaussian Density Estimation using Gaussian Kernels — Bandwidth = 0.9

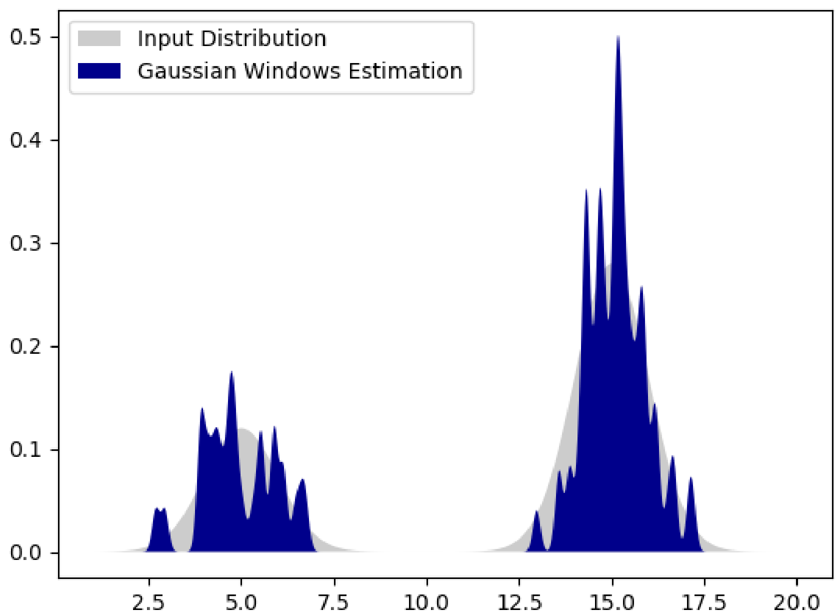

Figure 3.7: 1-D Multi-modal Gaussian Density Estimation using Gaussian Kernel — Bandwidth = 0.1

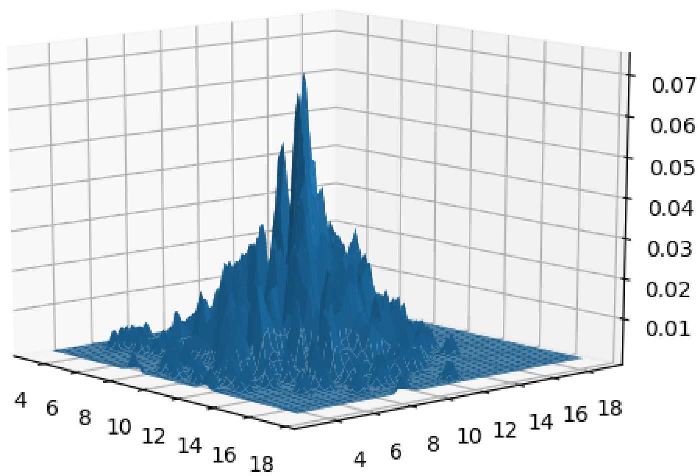

Figure 3.8: 2-D Gaussian Density Estimation using Gaussian Kernels — Bandwidth = 0.2

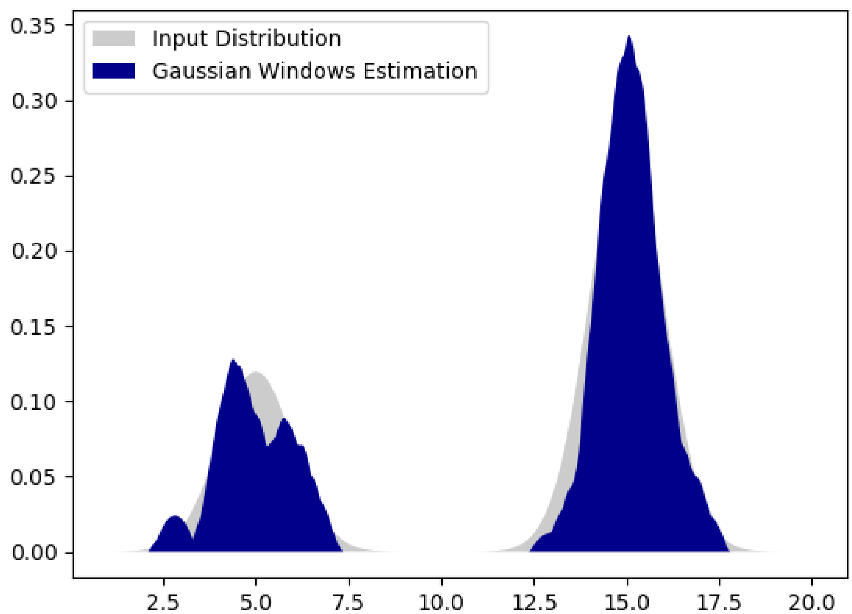

Figure 3.9: 1-D Gaussian Density Estimation using Epanechnikov Kernels — Bandwidth = 0.6