

Lecture 10: Non-parametric Clustering

Shukai Gong

1 Nonparametric Clustering

1.1 Kernel Density Estimation

Given i.i.d. samples, we wish to estimate their density function in a nonparametric way.

Kernel Density Estimation (KDE)

Given i.i.d. samples $\{\mathbf{x}_n\}_{n=1}^N$, the KDE of the samples is defined as

$$\hat{p}_h(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n) := \frac{1}{N} \sum_{n=1}^N K_h(\mathbf{x} - \mathbf{x}_n)$$

where h is the bandwidth of the kernel.

[Note]: For 1-dimension data, under Gaussian kernel, a rule of thumb for bandwidth selection is $h = \left(\frac{4\hat{\sigma}^5}{3N}\right)^{0.2} \approx 1.06\hat{\sigma}N^{-0.2}$, where $\hat{\sigma}$ is the standard deviation of the samples.

[Note]: When K_h is a Gaussian kernel, the KDE can actually be interpreted as a GMM model with **known parameters**.

Figure 1 shows a simple example of the KDE of a 1-dimension dataset. The left histogram is essentially a KDE graph with a gate kernel ($K_h(\mathbf{x}, \mathbf{x}') = \frac{1}{h} \cdot \mathbb{I}(\|\mathbf{x} - \mathbf{x}'\|_1 \leq h)$), and the right KDE curve is obtained by adding up the Gaussian kernels ($K_h(\mathbf{x}, \mathbf{x}') = \frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2h^2}\right)$) of each sample.

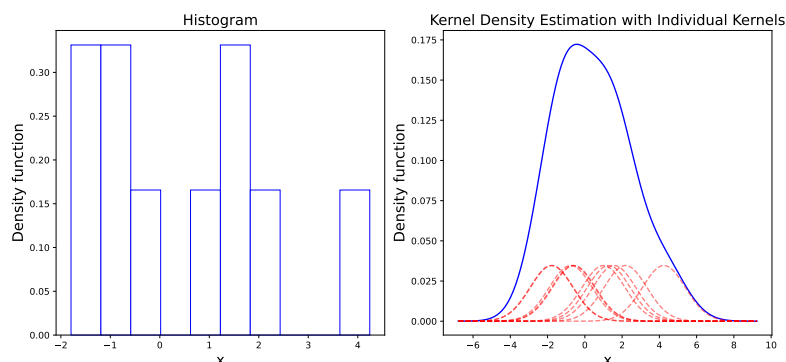


Figure 1: Kernel Density Estimation

1.2 Mean-Shift Algorithm

As an overview, the Mean-Shift clustering algorithm is a nonparametric clustering algorithm that works by iteratively shifting the data points towards the **densest** area of the data, and therefore forming clusters. Figure 2 shows a simplest 1D example of shifting process. The data points are shifted towards the peak of kernel density.

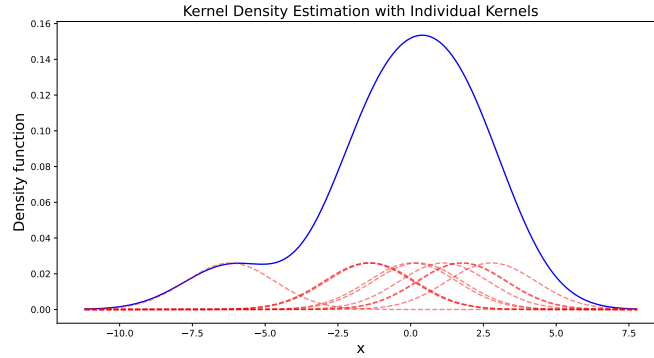


Figure 2: Shifting Data Points to the Densest Area

It's nonparametric because unlike K-means clustering, it doesn't require the number of clusters, K , to be specified beforehand. The number of clusters is determined based on the areas where the density of the data points is highest, and the densest areas are determined by the kernel density estimation introduced before.

Before we introduce the steps of Mean-Shift algorithm, we need to specify the definition of *the mean of a data point* in the algorithm. Given a set of samples $\{\mathbf{x}_n\}_{n=1}^N$ and a kernel function $K_h(\mathbf{x}, \mathbf{x}')$ with bandwidth h , the mean of a data point \mathbf{x} in its neighborhood $\mathcal{N}(\mathbf{x})$ is defined as

$$m(\mathbf{x}) = \frac{\sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x})} \mathbf{x}_i K_h(\mathbf{x}, \mathbf{x}_i)}{\sum_{\mathbf{x}_i \in \mathcal{N}(\mathbf{x})} K_h(\mathbf{x}, \mathbf{x}_i)}$$

the neighborhood $\mathcal{N}(\mathbf{x})$ can be defined as a circle centered at \mathbf{x} with radius h , or a KNN neighborhood with k nearest neighbors, or merely all data points. Here we can adopt $\mathcal{N}(\mathbf{x}) = \{\mathbf{x}_i\}_{i=1}^N$ for simplicity, that is,

$$m(\mathbf{x}) = \frac{\sum_{i=1}^N \mathbf{x}_i K_h(\mathbf{x}, \mathbf{x}_i)}{\sum_{i=1}^N K_h(\mathbf{x}, \mathbf{x}_i)}$$

one can notice that $m(\mathbf{x})$ coincides with the definition of Nadaraya-Watson estimator in kernel regression.

Mean-Shift Algorithm

- **Initialization:** Initialize the mean of each data point to its own value.

- **Mean-Shift Vector:** For each \mathbf{x}_i , the mean-shift vector is $m(\mathbf{x}_i) - \mathbf{x}_i = \frac{\sum_{i=1}^N \mathbf{x}_i K_h(\mathbf{x}, \mathbf{x}_i)}{\sum_{i=1}^N K_h(\mathbf{x}, \mathbf{x}_i)} - \mathbf{x}_i$.

- **Shifting:** Updating by $\mathbf{x}_i \leftarrow m(\mathbf{x}_i)$

- **Iteration:** Repeat steps 2 and 3 until convergence or maximum iteration is reached.

Figure 3 shows the Mean-Shift algorithm in action. Now we will justify why by updating $\mathbf{x}_i \leftarrow m(\mathbf{x}_i)$, the data points will eventually converge to their nearest peak of the kernel density. Suppose the kernel function is a Gaussian kernel $K_h(\mathbf{x}, \mathbf{x}') = \frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2h^2}\right)$, we compute the gradient of kernel density

$$\hat{p}_h(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N K_h(\mathbf{x} - \mathbf{x}_n) \text{ as}$$

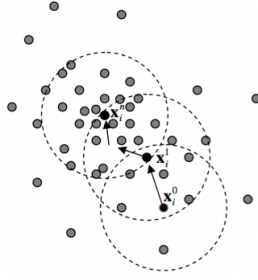


Figure 3: Mean-shift Algorithm

$$\begin{aligned}
\nabla_{\mathbf{x}} \hat{p}_h(\mathbf{x}) &= \frac{1}{N} \sum_{n=1}^N \nabla_{\mathbf{x}} K_h(\mathbf{x} - \mathbf{x}_n) = \frac{1}{N} \sum_{n=1}^N \nabla_{\mathbf{x}} \left(\frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_n\|_2^2}{2h^2}\right) \right) \\
&= \frac{1}{N} \sum_{n=1}^N \frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_n\|_2^2}{2h^2}\right) \frac{\mathbf{x} - \mathbf{x}_n}{h^2} = -\frac{1}{Nh^2} \sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n)(\mathbf{x} - \mathbf{x}_n) \\
&\propto \sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n)(\mathbf{x} - \mathbf{x}_n) \propto \frac{\sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n)(\mathbf{x}_n - \mathbf{x})}{\sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n)\mathbf{x}} = \frac{\sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n)\mathbf{x}_n}{\sum_{n=1}^N K_h(\mathbf{x}, \mathbf{x}_n)} - \mathbf{x} \\
&= m(\mathbf{x}) - \mathbf{x}
\end{aligned}$$

The gradient of the kernel density is proportional to the mean-shift vector, which means that by updating $\mathbf{x}_i \leftarrow m(\mathbf{x}_i)$, we are essentially doing a gradient ascent on the data points: $\mathbf{x}_n^{(t+1)} = \mathbf{x}_n^{(t)} + \eta \nabla_{\mathbf{x}} \hat{p}_h(\mathbf{x}_n^{(t)})$.

The Mean-Shift algorithm can also be considered as a variant of EM algorithm, where the **E-step** is computing $m(\mathbf{x}_i), \forall i$, and the **M-step** is updating $\mathbf{x}_i \leftarrow m(\mathbf{x}_i)$. Under the EM algorithm setup:

- Data: $\{\mathbf{x}_n\}_{n=1}^N$
- Latent variable: means / centroids $\{\mathbf{m}_k\}_{k=1}^K$
- E-step: estimating the mean $m(\mathbf{x}_i)$ conditioned on \mathbf{x}_i .
- M-step: maximizing $p(\mathbf{x}_i)$ (finding the nearest peak)

References

- [Latent Variable Model: Gaussian Mixture Model](#)
- [EM Algorithm for GMM and its Bayesian interpretation](#)
- [Mean Shift](#)
- [Machine Learning - Mean-Shift Clustering](#)
- [Wikipedia - Mean shift](#)