Machine Learning in Robotics
Nonparametric Density Estimation

Prof. Dongheui Lee

Institute of Automatic Control Engineering
Technische Universität München

dhlee@tum.de

Density Estimation - Motivation

- In our previous lecture on decision theory, we saw that the optimal classifier could be expressed as a family of discriminant functions

$$g_i(\mathbf{x}) = p(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)p(\omega_i)}{p(\mathbf{x})}$$

- Decision rule was

$$\text{choose } \omega_i \text{ if } g_i(\mathbf{x}) > g_j(\mathbf{x}), \quad \forall j \neq i$$

- We need to estimate both prior $p(\omega_i)$ and likelihood $p(\mathbf{x}|\omega_i)$
- During next lectures, techniques to estimate the likelihood density function $p(\mathbf{x}|\omega_i)$ will be introduced

Approaches for Density Estimation

Parametric Approach

- A given form for the density function is assumed (i.e., Gaussian) and the parameters of the function (i.e., mean and variance) are optimized by fitting the model to the data set
- Parametric density estimation is often referred to as Parameter Estimation

Non-Parametric Approach

- No functional form for the density function is assumed. The density estimate is driven entirely by the data
- called Parameter Estimation
  - Kernel Density Estimation
  - Nearest Neighbor Rule

Histogram Density Model

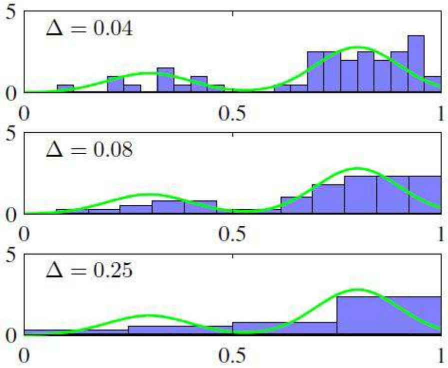
- The simplest form of non-parametric density estimation is the familiar *histogram*
- Standard histogram : Divide the sample space into distinct bins of width Δ_i and approximate the density at each bin by the fraction of points in the training data that fall into the corresponding bin i.

$$p_i = \frac{n_i}{n\Delta_i} \qquad \int p(\mathbf{x})d\mathbf{x} = 1$$

where n_i is the number of observations of x falling in bin i and n is the total number of observations.
Often, $\Delta_i = \Delta$.

Properties of Histogram Density Model

A histogram density model is dependent on the choice of histogram bin-width Δ.



- If Δ is very small, the resulting density model is very spiky
- If very large, the model is too smooth

Properties of Histogram Density Model

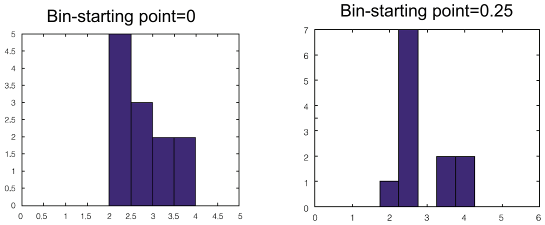
A histogram density model is dependent on the choice of edge location for the bins

- Dataset $X = [2.3\ 2.4\ 2.34\ 2.41\ 2.71\ 2.65\ 3.34\ 3.73]$
- Bin-width $\Delta = 0.5$

Properties of Histogram Density Model

A histogram density model is dependent on the choice of edge location for the bins

- Dataset $X = [2.3\ 2.4\ 2.34\ 2.41\ 2.71\ 2.65\ 3.34\ 3.73]$
- Bin-width $\Delta = 0.5$



Properties of Histogram Density Model

- A very simple form of density estimation
- The density estimate depends on the starting position of the bins and bin-width Δ

Properties of Histogram Density Model

- A very simple form of density estimation
- The density estimate depends on the starting position of the bins and bin-width Δ
- The discontinuities of the estimate are not due to the underlying density, they are only an artifact of the chosen bin locations

Properties of Histogram Density Model

- A very simple form of density estimation
- The density estimate depends on the starting position of the bins and bin-width Δ
- The discontinuities of the estimate are not due to the underlying density, they are only an artifact of the chosen bin locations
- A much more serious problem is the curse of dimensionality, since the number of bins grows exponentially with the number of dimensions

General Formulation of Non-parametric density estimation

- The probability that a vector $\boldsymbol{x}$, drawn from a distribution $p(x)$, will fall in a region $\Re$ of the sample space is $P = \int_{\Re} p(\boldsymbol{x}) d\boldsymbol{x}$

General Formulation of Non-parametric density estimation

- The probability that a vector $\boldsymbol{x}$, drawn from a distribution $p(x)$, will fall in a region $\Re$ of the sample space is $P = \int_{\Re} p(\boldsymbol{x}) d\boldsymbol{x}$
- Suppose that n independently and identically distributed samples $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(n)})$ are drawn from the probability $p(\boldsymbol{x})$. The probability that K of these n vectors fall in $\Re$ is given by the binomial law

$$Bin(K|n, P) = \frac{n!}{K!(n-K)!} P^K (1-P)^{n-K}$$

General Formulation of Non-parametric density estimation

- The probability that a vector $\boldsymbol{x}$, drawn from a distribution $p(x)$, will fall in a region $\Re$ of the sample space is $P = \int_{\Re} p(\boldsymbol{x}) d\boldsymbol{x}$
- Suppose that n independently and identically distributed samples $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)})$ are drawn from the probability $p(\boldsymbol{x})$. The probability that K of these n vectors fall in $\Re$ is given by the binomial law

$$Bin(K|n, P) = \frac{n!}{K!(n-K)!} P^K (1-P)^{n-K}$$

- If n is very large, $P = \frac{K}{n}$

General Formulation of Non-parametric density estimation

- The probability that a vector $\boldsymbol{x}$, drawn from a distribution $p(x)$, will fall in a region $\Re$ of the sample space is $P = \int_{\Re} p(\boldsymbol{x}) d\boldsymbol{x}$
- Suppose that n independently and identically distributed samples $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(n)})$ are drawn from the probability $p(\boldsymbol{x})$. The probability that K of these n vectors fall in $\Re$ is given by the binomial law

$$Bin(K|n, P) = \frac{n!}{K!(n-K)!} P^K (1-P)^{n-K}$$

- If n is very large, $P = \frac{K}{n}$
- If the region $\Re$ is very small, $P = \int_{\Re} p(\boldsymbol{x}) d(\boldsymbol{x}) \simeq p(\boldsymbol{x}) V$

General Formulation of Non-parametric density estimation

- The probability that a vector $\boldsymbol{x}$, drawn from a distribution $p(x)$, will fall in a region $\Re$ of the sample space is $P = \int_{\Re} p(\boldsymbol{x}) d\boldsymbol{x}$
- Suppose that n independently and identically distributed samples $(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(n)})$ are drawn from the probability $p(\boldsymbol{x})$. The probability that K of these n vectors fall in $\Re$ is given by the binomial law

$$Bin(K|n, P) = \frac{n!}{K!(n-K)!} P^K (1-P)^{n-K}$$

- If n is very large, $P = \frac{K}{n}$
- If the region $\Re$ is very small, $P = \int_{\Re} p(\boldsymbol{x}) d(\boldsymbol{x}) \simeq p(\boldsymbol{x}) V$

$$p(\boldsymbol{x}) \simeq \frac{K}{nV}$$

General Formulation of Non-parametric density estimation

Discussion on underlying assumptions

- In practice the value of n is fixed (the total number of examples)

General Formulation of Non-parametric density estimation

Discussion on underlying assumptions

- In practice the value of n is fixed (the total number of examples)
- In order to improve the accuracy of the estimate $p(x)$ we could let V to approach zero, but then the region $\Re$ would become so small that it would enclose no examples

General Formulation of Non-parametric density estimation

Discussion on underlying assumptions

- In practice the value of n is fixed (the total number of examples)
- In order to improve the accuracy of the estimate $p(x)$ we could let V to approach zero, but then the region $\Re$ would become so small that it would enclose no examples
- This means that in practice we will have to find a compromise value of the volume V
  - Large enough to include enough examples within $\Re$
  - Small enough to support the assumption that $p(x)$ is constant within $\Re$

General Formulation of Non-parametric density estimation

Discussion on underlying assumptions

- In practice the value of n is fixed (the total number of examples)
- In order to improve the accuracy of the estimate $p(x)$ we could let V to approach zero, but then the region $\Re$ would become so small that it would enclose no examples
- This means that in practice we will have to find a compromise value of the volume V
  - Large enough to include enough examples within $\Re$
  - Small enough to support the assumption that $p(x)$ is constant within $\Re$

Two approaches

- Fix V and determine K from the data: Kernel Density Estimation (KDE)
- Fix K and determine V from the data: k Nearest Neighbor (kNN)

General Formulation of Non-parametric density estimation

Discussion on underlying assumptions

- In practice the value of n is fixed (the total number of examples)
- In order to improve the accuracy of the estimate $p(x)$ we could let V to approach zero, but then the region $\Re$ would become so small that it would enclose no examples
- This means that in practice we will have to find a compromise value of the volume V
  - Large enough to include enough examples within $\Re$
  - Small enough to support the assumption that $p(x)$ is constant within $\Re$

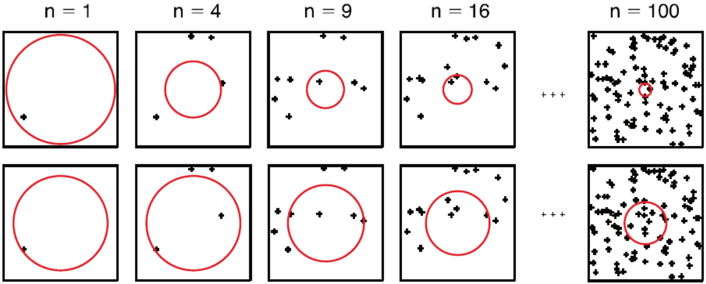
Two approaches

- Fix V and determine K from the data: Kernel Density Estimation (KDE)
- Fix K and determine V from the data: k Nearest Neighbor (kNN)
- As $n \to \infty$, both approaches become close to the true probability density

General Formulation of Non-parametric density estimation

KDE $V_n = 1/\sqrt{n}$

kNN $k_n = \sqrt{n}$



[Duda, Hart, Stock, 2001]

KDE using a Parzen Window

- Nonparametric density estimation general formula $p(\boldsymbol{x}) \simeq \frac{K}{nV}$

KDE using a Parzen Window

- Nonparametric density estimation general formula $p(\boldsymbol{x}) \simeq \frac{K}{nV}$
- Region $\Re$: a small hypercube centered on the estimation point $\boldsymbol{x}$, $V = h^m$

KDE using a Parzen Window

- Nonparametric density estimation general formula $p(\boldsymbol{x}) \simeq \frac{K}{nV}$
- Region $\Re$: a small hypercube centered on the estimation point $\boldsymbol{x}$, $V = h^m$
- Kernel function

$$k(\boldsymbol{u}) = \begin{cases} 1 & \text{if } \left|u_{(j)}\right| \leq 0.5 \quad , \forall j = 1, \ldots .m \\ 0 & \text{otherwise} \end{cases}$$

KDE using a Parzen Window

- Nonparametric density estimation general formula $p(\boldsymbol{x}) \simeq \frac{K}{nV}$
- Region $\Re$: a small hypercube centered on the estimation point $\boldsymbol{x}$, $V = h^m$
- Kernel function

$$k(\boldsymbol{u}) = \begin{cases} 1 & \text{if } \left|u_{(j)}\right| \leq 0.5 \quad, \forall j = 1, \ldots .m \\ 0 & \text{otherwise} \end{cases}$$

- For the dataset $\boldsymbol{X} = \left\{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)}\right\}$, the total number of points inside the hypercube is $K = \sum_{i=1}^{n} k\left(\frac{(\boldsymbol{x}-\boldsymbol{x}^{(i)})}{h}\right)$

KDE using a Parzen Window

- Nonparametric density estimation general formula $p(\boldsymbol{x}) \simeq \frac{K}{nV}$
- Region $\Re$: a small hypercube centered on the estimation point $\boldsymbol{x}$, $V = h^m$
- Kernel function

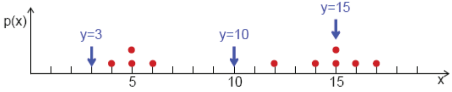
$$k(\boldsymbol{u}) = \begin{cases} 1 & \text{if } \left|u_{(j)}\right| \leq 0.5 \quad , \forall j = 1, \ldots m \\ 0 & \text{otherwise} \end{cases}$$

- For the dataset $\boldsymbol{X} = \left\{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)}\right\}$, the total number of points inside the hypercube is $K = \sum_{i=1}^{n} k\left(\frac{(\boldsymbol{x}-\boldsymbol{x}^{(i)})}{h}\right)$
- Density estimate

$$p(\boldsymbol{x}) = \frac{1}{nh^m} \sum_{i=1}^{n} k\left(\frac{(\boldsymbol{x} - \boldsymbol{x}^{(i)})}{h}\right)$$

Parzen Estimator Simple Example

- Given the dataset below, use Parzen windows to estimate the density $p(x)$ at $x = 3, 10, 15$. Use a bandwidth of $h = 4$.
  $\boldsymbol{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(n)}\} = \{4, 5, 5, 6, 12, 14, 15, 15, 16, 17\}$

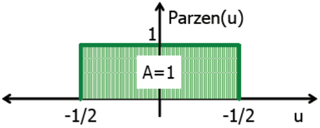


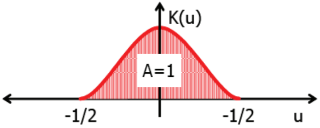
- Estimate $p(x = 3)$, $p(x = 10)$, $p(x = 15)$

KDE using a Smooth Kernel

- KDE using the parzen window
  - Discontinuity
  - Equal weights for all data points
- If using smooth Kernel function which $k(u) > 0 \ , \quad \int k(u) du = 1$
- For example, a Gaussian

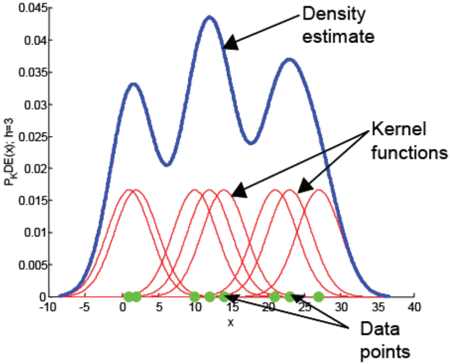
$$p(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi} h} \exp\left(- \frac{\left| \boldsymbol{x} - \boldsymbol{x}^{(i)} \right|^2}{2h^2} \right)$$





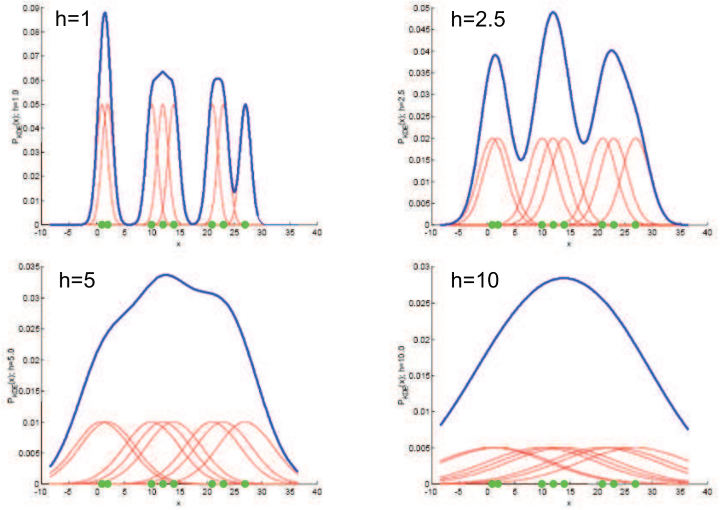
KDE using a Smooth Kernel

- Similar to the Parzen window, estimator is a sum of bumps placed at the data points
- The kernel function determines the shape of the bumps
- The parameter h, also called the *smoothing parameter* or *bandwidth*, determines their width

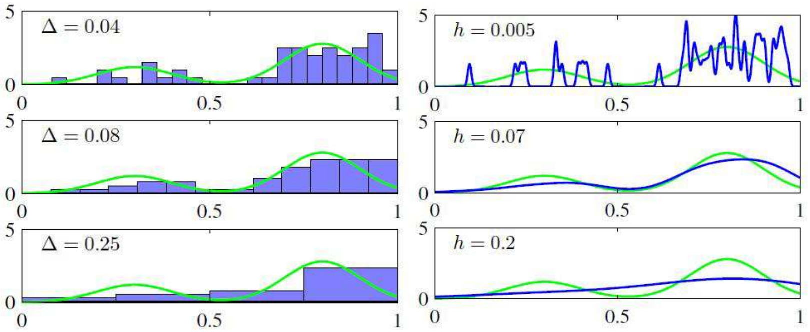


Choosing the bandwidth

- Large h : over-smoothing
- Small h : sensitive to noise



Histogram vs. KDE using a smooth Kernel

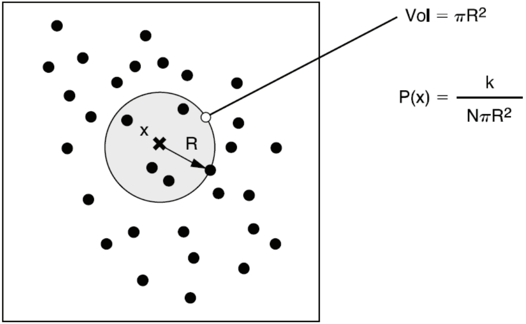


kNN Density Estimation

- In the kNN method we grow the volume surrounding the estimation point $\mathbf{x}$ so that it encloses a total of K points
- The density estimate then becomes

$$p(\mathbf{x}) \simeq \frac{K}{nV}$$

V is the volume that contains K points



kNN Density Estimation

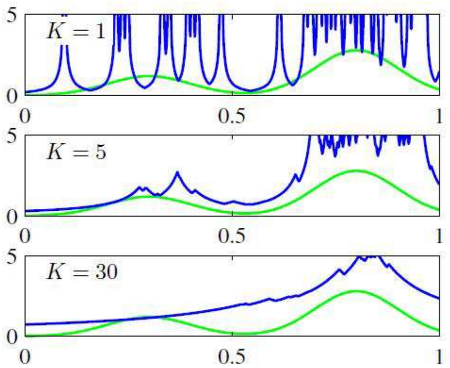


Illustration of K nearest neighbor density using the same data set as in previous examples. We see that the parameter K governs the degree of smoothing, so that a small value of K leads to a very noisy density model (top panel), whereas a large value (bottom panel) smooths out the bimodal nature of the true distribution (shown by the green curve) from which the data set was generated.

K Nearest Neighbor Rule (k-NNR)

An intuitive classification method

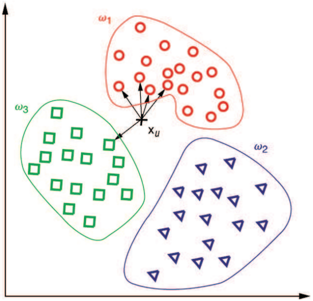
- to classify unlabeled examples based on their similarity with examples in the training set
- To find K closest labeled examples and assign the query data to the class that appears most within the K examples

k-NNR

- An integer K
- A set of labeled examples (training data)
- A metric to measure "closeness"

Example

- A query point $\boldsymbol{x}_u$
- $K = 5$
- Classification result: ω_1



K Nearest Neighbor Rule (k-NNR)

- k-NNR classification
  - K-nearest neighbor (kNN) density estimation technique
  - Use Bayes theorem
- Problem setting
  - n datapoints
  - For each cluster ω_i, n_i datapoints are included
  - Classify a new point $\boldsymbol{x}$. Find the cluster which has the maximum $p(\omega_i|\boldsymbol{x})$
- Solution

$$\sum_{\forall i} K_i = K$$

$$p(\boldsymbol{x}|\omega_i) =$$

$$p(\boldsymbol{x}) =$$

$$p(\omega_i) =$$

$$p(\omega_i|\boldsymbol{x}) =$$

K Nearest Neighbor Rule (k-NNR)

- k-NNR classification
  - K-nearest neighbor (kNN) density estimation technique
  - Use Bayes theorem
- Problem setting
  - n datapoints
  - For each cluster ω_i, n_i datapoints are included
  - Classify a new point $\boldsymbol{x}$. Find the cluster which has the maximum $p(\omega_i|\boldsymbol{x})$
- Solution

$$\sum_{\forall i} K_i = K \quad , \quad p(\boldsymbol{x}|\omega_i) = \frac{K_i}{n_i V}$$

$$p(\boldsymbol{x}) = \frac{K}{nV} \quad , \quad p(\omega_i) = \frac{n_i}{n}$$

$$p(\omega_i|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\omega_i)p(\omega_i)}{p(\boldsymbol{x})} = \frac{K_i}{K}$$

Nonparametric Density Estimation

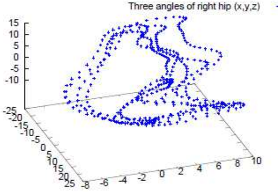
- Big storage requirements
  - Have to save entire training dataset
- Requires large dataset for realistic density estimation
- Expensive computational cost on recall in the case of a large dataset

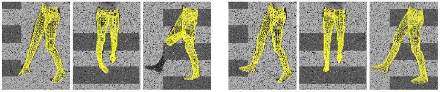
Nonparametric Density Estimation for Human Pose Tracking

- An object model is assumed.
- The pose parameters of the model are learned so that the model optimally explains object's image data.
- The joint probability of a pose x and an image feature C is given by:
  $p(x, C|I) = \frac{p(I|C,x)p(C|x)p(x)}{p(I)}$, $\quad I$: input image.
- Non-parametric density estimation is realized to capture the complex configuration of human pose.

T. Brox, T. Rosenhahn, U. Kersting and D. Cremers. *Nonparametric Density Estimation for Human Pose Tracking*. Springer-Verlag, 2006.

Nonparametric Density Estimation





Without prior With prior

- The prior probability for the joint angle Θ is learned through non-parametric density model; Parzen-Rosenblatt estimator: $p(\Theta) = \frac{1}{\sqrt{2\pi\sigma N}} \sum_{i=1}^{N} exp(-\frac{(\Theta_i - \Theta)^2}{2\sigma^2})$.
- N: number of training samples Θ_i.
- σ : tuning parameter.

Segmentation and appearance model building from an image sequence

- Segment a human given multiple video frames; select features which do not change over time.
- Non-parametric kernel-based PDF estimator is used for segmentation of the human.

L. Zhao and L. S. Davis. *Segmentation and appearance model building from an image sequence*. ICIP 2005.

Appearance model building: steps

- Select a constant appearance human body model.
- Estimate the probability of a pixel x belonging to the foreground f: $P_{fg} = \sum_i P_{fg}(x_i)\Pi_{j=1}^{m} K(\frac{y_j - x_{ij}}{\sigma_j})$.
- Estimate the probability of a pixel x belonging to the background b: $P_{bg} = \sum_i P_{bg}(x_i)\Pi_{j=1}^{m} K(\frac{y_j - x_{ij}}{\sigma_j})$.

Announcements

- Further Reading
  - Duda, Chapter 4.1-4.5
  - Bishop, Chapter 2.5, 3.2
  - Mitchell, Chapter 8