

Non parametric methods

Course of Machine Learning
Master Degree in Computer Science
University of Rome "Tor Vergata"

Giorgio Gambosi

a.a. 2017-2018

- The statistical approach to classification requires the (at least approximate) knowledge of $p(\mathcal{C}_i|\mathbf{x})$: in fact, an item \mathbf{x} shall be assigned to the class \mathcal{C}_i such that

$$i = \operatorname{argmin}_j \sum_{k \neq j} L_{kj} p(\mathcal{C}_k|\mathbf{x})$$

where L_{kj} is the misclassification cost of $\mathbf{x} \in \mathcal{C}_k$ assigned to \mathcal{C}_j

- In the case of uniform costs, $L_{kj} = 1$ for $k \neq j$ and $L_{jj} = 0$, this corresponds to assigning the item to \mathcal{C}_i , where $i = \operatorname{argmax}_j p(\mathcal{C}_j|\mathbf{x})$
- The same holds in the regression case, where $p(y|\mathbf{x})$ has to be estimated.

What do we assume to know of class distributions, given a training set \mathbf{X}, \mathbf{t} ?

- Case 1. The probabilities $p(\mathbf{x}|\mathcal{C}_i)$ are known: an item is assigned \mathbf{x} to the class \mathcal{C}_i such that

$$i = \operatorname{argmin}_j \sum_{k \neq j} L_{kj} p(\mathcal{C}_k | \mathbf{x})$$

where $p(\mathcal{C}_k | \mathbf{x})$ can be derived through Bayes' rule and prior probabilities, since $p(\mathcal{C}_k | \mathbf{x}) \propto p(\mathbf{x} | \mathcal{C}_k) p(\mathcal{C}_k)$

Probability distribution estimates: hypotheses

- Case 2. The **type** of probability distribution $p(\mathbf{x}|\boldsymbol{\theta})$ is known: an estimate of parameter values $\boldsymbol{\theta}_i$ is performed for all classes, taking into account for each class \mathcal{C}_i the subset of $\mathbf{X}_i, \mathbf{t}_i$ of items belonging to the class, that is such that $t = i$. Different approaches to parameter estimation:

1. Maximum likelihood: $\boldsymbol{\theta}_i^{ML} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} p(\mathbf{X}_i, \mathbf{t}_i | \boldsymbol{\theta})$ is computed. Item \mathbf{x} is assigned to class \mathcal{C}_i if

$$i = \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathcal{C}_k | \mathbf{x}) = \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathbf{x} | \boldsymbol{\theta}_k^{ML}) p(\mathcal{C}_k)$$

2. Maximum a posteriori: $\boldsymbol{\theta}_i^{MAP} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} p(\boldsymbol{\theta} | \mathbf{X}_i, \mathbf{t}_i)$ is computed. Item \mathbf{x} is assigned to class \mathcal{C}_i if

$$i = \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathcal{C}_k | \mathbf{x}) = \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathbf{x} | \boldsymbol{\theta}_k^{MAP}) p(\mathcal{C}_k)$$

3. Bayesian estimate: the distributions $p(\boldsymbol{\theta} | \mathbf{X}_i, \mathbf{t}_i)$ are estimated for each class and, from them,

$$p(\mathbf{x} | \mathcal{C}_i) = \int_{\boldsymbol{\theta}} p(\mathbf{x} | \boldsymbol{\theta}) p(\boldsymbol{\theta} | \mathbf{X}_i, \mathbf{t}_i) d\boldsymbol{\theta}$$

Item \mathbf{x} is assigned to class \mathcal{C}_i if

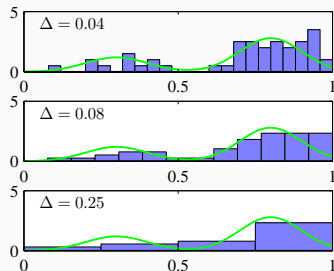
$$\begin{aligned} i &= \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathcal{C}_k | \mathbf{x}) = \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathcal{C}_k) p(\mathbf{x} | \mathcal{C}_k) \\ &= \underset{j}{\operatorname{argmin}} \sum_{k \neq j} L_{kj} p(\mathcal{C}_k) \int_{\boldsymbol{\theta}} p(\mathbf{x} | \boldsymbol{\theta}) p(\boldsymbol{\theta} | \mathbf{X}_i, \mathbf{t}_i) d\boldsymbol{\theta} \end{aligned}$$

- Case 3. No knowledge of the probabilities assumed.
- The class distributions $p(\mathbf{x}|\mathcal{C}_i)$ are directly from data.
- In previous cases, use of (parametric) models for a synthetic description of data in \mathbf{X}, \mathbf{t}
- In this case, no models (and parameters): training set items explicitly appear in class distribution estimates.
- Denoted as **non parametric** models: indeed, an unbounded number of parameters is used

Histograms

- Elementary type of non parametric estimate
- Domain partitioned into m d -dimensional intervals (bins)
- The probability $P_{\mathbf{x}}$ that an item belongs to the bin containing item \mathbf{x} is estimated as $\frac{n(\mathbf{x})}{n}$, where $n(\mathbf{x})$ is the number of element in that bin
- The probability density in the interval corresponding to the bin containing \mathbf{x} is then estimated as the ratio between the above probability and the interval width $\Delta(\mathbf{x})$ (typically, a constant Δ)

$$p_H(\mathbf{x}) = \frac{\frac{n(\mathbf{x})}{N}}{\Delta(\mathbf{x})} = \frac{n(\mathbf{x})}{N\Delta(\mathbf{x})}$$



- The density is a function of the position of the first bin. In the case of multivariate data, also from bin orientation.
- The resulting estimates is not continuous.
- Curse of dimensionality: the number of bins grows as a polynomial of order d : in high-dimensional spaces many bins may result empty, unless a large number of items is available.
- In practice, histograms can be applied only in low-dimensional datasets (1,2)

Kernel density estimators

- Probability that an item is in region $\mathcal{R}(\mathbf{x})$, containing \mathbf{x}

$$P_{\mathbf{x}} = \int_{\mathcal{R}(\mathbf{x})} p(\mathbf{z}) d\mathbf{z}$$

- Given n items $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$, the probability that k among them are in $\mathcal{R}(\mathbf{x})$ is given by the binomial distribution

$$p(k) = \binom{n}{k} P_{\mathbf{x}}^k (1 - P_{\mathbf{x}})^{n-k} = \frac{n!}{k!(n-k)!} P_{\mathbf{x}}^k (1 - P_{\mathbf{x}})^{n-k}$$

- Mean and variance of the ratio $r = \frac{k}{n}$ are

$$E[r] = P_{\mathbf{x}} \qquad \text{var}[r] = \frac{P_{\mathbf{x}}(1 - P_{\mathbf{x}})}{n}$$

- $P_{\mathbf{x}}$ is the expected fraction of items in $\mathcal{R}(\mathbf{x})$, and the ratio r is an estimate. As $n \rightarrow \infty$ variance decreases and r tends to $E[r] = P_{\mathbf{x}}$. Hence, in general,

$$r = \frac{k}{n} \simeq P(\mathbf{x})$$

Nonparametric estimates

- Let the volume of $\mathcal{R}(\mathbf{x})$ be sufficiently small. Then, the density $p(\mathbf{x})$ is almost constant in the region and

$$P_{\mathbf{x}} = \int_{\mathcal{R}(\mathbf{x})} p(\mathbf{z}) d\mathbf{z} \simeq p(\mathbf{x})V$$

where V is the volume of $\mathcal{R}(\mathbf{x})$

- since $P_{\mathbf{x}} \simeq \frac{k}{n}$, it then derives that $p(\mathbf{x}) \simeq \frac{k}{nV}$
- n is fixed: this implies that the estimate of $p(\mathbf{x})$ can be improved only by decreasing V . However, a volume too small makes many bins empty.
- Indeed, as $V \rightarrow 0$ with n fixed, the ratio $\frac{P_{\mathbf{x}}}{V}$ will tend either to 0 (if no items are in the bin anymore) or to ∞ (if the region shrinks around a point corresponding to an item)
- In practice, a compromise is necessary between large values (so that enough items are included in bins) and small values of V (to allow to consider $p(\mathbf{x})$ constant within a region)

Two alternative ways to exploit the estimate $p(\mathbf{x}) \simeq \frac{k}{nV}$

1. Fix V and derive k from data (kernel density estimation)
2. Fix k and derive V from data (K-nearest neighbor).

It can be shown that in both cases, under suitable conditions, the estimator tends to the true density $p(\mathbf{x})$ as $n \rightarrow \infty$.

Kernel density estimation: Parzen windows

- Region associated to a point \mathbf{x} : hypercube with edge length h (and volume h^d) centered on \mathbf{x} .
- Kernel function $k(\mathbf{u})$ (Parzen window) used to count the number of items in the unit hypercube centered on u

$$k(\mathbf{u}) = \begin{cases} 1 & |u_i| \leq 1/2 \\ 0 & \text{otherwise} \end{cases} \quad i = 1, \dots, d$$

- as a consequence, $k\left(\frac{\mathbf{x} - \mathbf{x}'}{h}\right) = 1$ iff \mathbf{x}' is in the hypercube of edge length h centered on \mathbf{x}
- the number of items in the hypercube is then

$$K = \sum_{i=1}^n k\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$$

- The estimated density is

$$p(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h^d} k\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$$

- Since

$$k(\mathbf{u}) \geq 0 \quad \text{and} \quad \int k(\mathbf{u}) d\mathbf{u} = 1$$

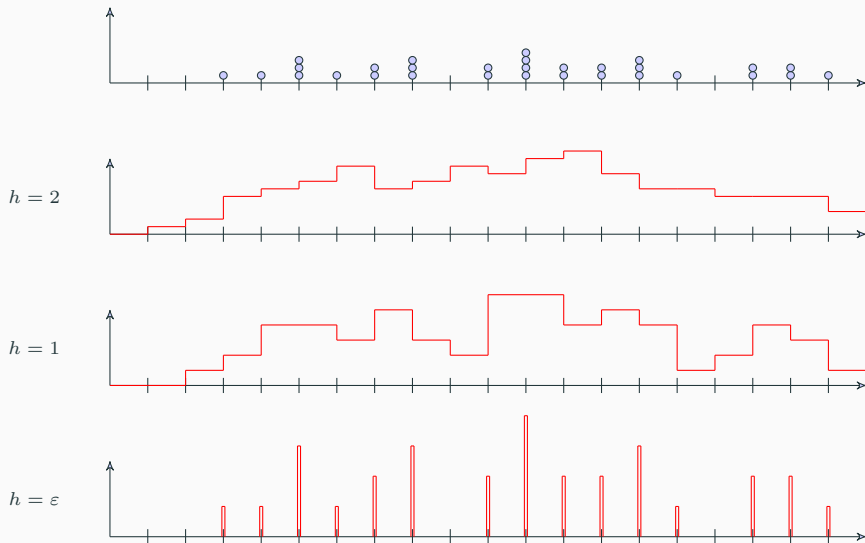
it derives

$$p(\mathbf{x}) \geq 0 \quad \text{and} \quad \int p(\mathbf{x}) d\mathbf{x} = 1$$

that is, $p_n(\mathbf{x})$ is a probability density

- Window size has a relevant effect on the estimate

Kernel density estimation: Parzen windows



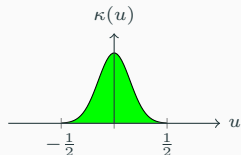
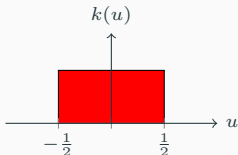
Kernels and smoothing

- Parzen windows drawbacks
 1. discontinuity of the estimates
 2. items in a region centered on \mathbf{x} have uniform weights: their distance from \mathbf{x} is not taken into account
- Solution: use of smooth kernel functions $\kappa(u)$

$$\int_{0^d}^{h^d} \kappa(\mathbf{x}) d\mathbf{x} = 1$$

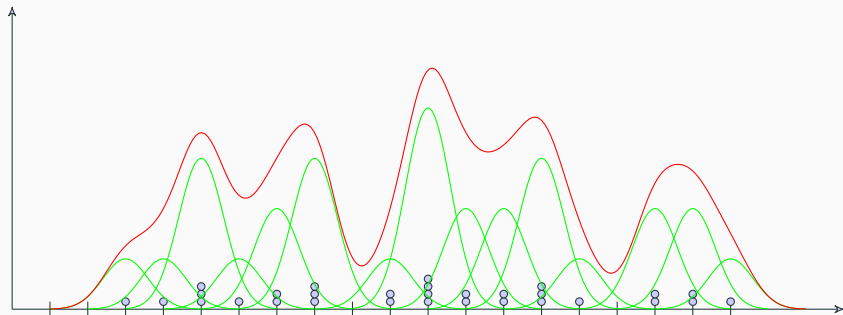
1. usually radial functions (functions of the distance from the center)
2. resulting estimate:

$$p(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^n \kappa\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$$



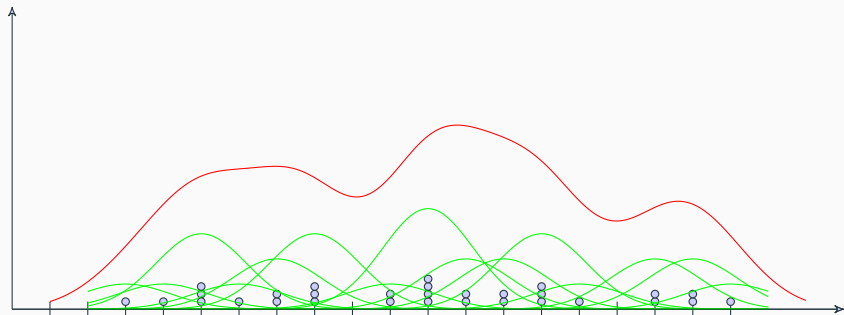
Kernels e smoothing

$$h = 1$$



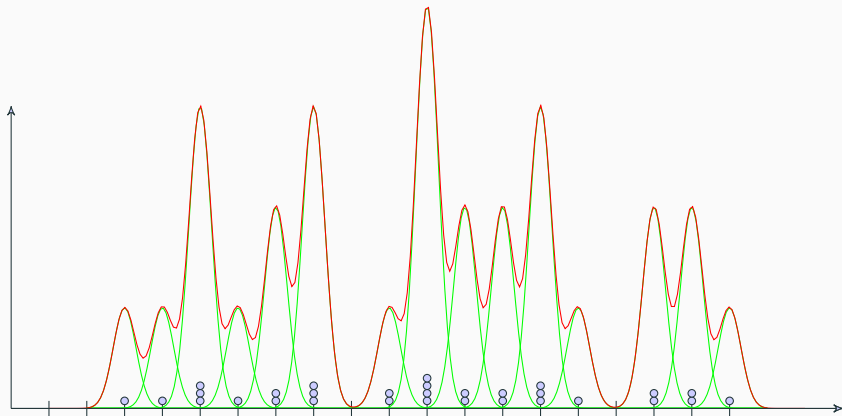
Kernel density estimation: Parzen windows

$$h = 2$$



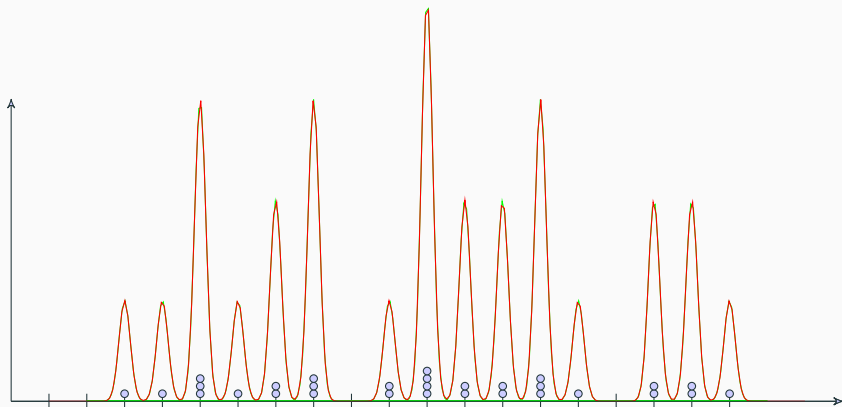
Kernel density estimation: Parzen windows

$$h = .5$$



Kernel density estimation: Parzen windows

$$h = .25$$



- The region around \mathbf{x} is extended to include k items
- The estimated density is

$$p(\mathbf{x}) \simeq \frac{k}{nV} = \frac{k}{nc_d r_k^d(\mathbf{x})}$$

where:

- c_d is the volume of the d -dimensional sphere of unitary radius
- $r_k^d(\mathbf{x})$ is the distance from \mathbf{x} to the k -th nearest item (the radius of the smallest sphere with center \mathbf{x} containing k items)

Classification through kNN

- To classify \mathbf{x}_i , let us consider a hypersphere of volume V with center \mathbf{x} containing k items from the training set
- Let k_i be the number of such items belonging to class \mathcal{C}_i . Then, the following approximation holds:

$$p(\mathbf{x}|\mathcal{C}_i) = \frac{k_i}{n_i V}$$

where n_i is the number of items in the training set belonging to class \mathcal{C}_i

- Similarly, for the evidence,

$$p(\mathbf{x}) = \frac{k}{nV}$$

- And, for the prior distribution,

$$p(\mathcal{C}_i) = \frac{n_i}{n}$$

- The class posterior distribution is then

$$p(\mathcal{C}_i|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_i)p(\mathcal{C}_i)}{p(\mathbf{x})} = \frac{\frac{k_i}{n_i V} \cdot \frac{n_i}{n}}{\frac{k}{nV}} = \frac{k_i}{k}$$

Classification through kNN

- Simple rule: an item is classified on the basis of similarity to near training set items
- To classify \mathbf{x} , determine the k items in the training nearest to it and assign \mathbf{x} to the majority class among them
- A metric is necessary to measure similarity.

