Skript Pattern Analysis Sommersemester 2017

Nils Häusler

 $\mathrm{May}\ 2,\ 2017$

1 Density Estimation

Let $p(\vec{x})$ denote a probability density function pdf then:

- 1. $p(\vec{x}) \ge 0$
- $2. \int_{-\infty}^{\infty} p(\vec{x}) \, d\vec{x} = 1$
- 3. $p(\vec{a} \le \vec{x} \le \vec{b}) = \int_{\vec{a}}^{\vec{b}} p(\vec{x}) d\vec{x}$

The task of density estimation is to obtain a continuous representation of the underlying pdf from a set of discrete samples (massumants). Note: that if we have the pdf we can do statistical analysis.

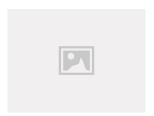
Parametric density estimation (mostly Pattern Recognition)

Make an assumption about the underlying distribution (e.g. Gausian, GMM) and determine the best fitting distribution parameters from the data. (ML estimation, MAP estimation)

Non-parametric density estimation We make no assumption of the underlying Model.

1.1 Parzen-Rosenblatt estimator

???Idea: Quantify the number of samples with a window



The Parzen window estimator interpolates the pdf from the observations in the neighbourhood of a position x, using an appropriate kernel/window function.

Short derivtion: Let p_R denote the probability that \vec{x} lies within region R:

$$p_R = \int_R p(\vec{x}) \, d\vec{x}$$

Now assume that $p(\vec{x})$ is approximately constant in R.

$$p_R \approx p(\vec{x}) \int_R d\vec{x}$$

For example, let R be a d-dimensional hypercube with side length h, then its volume¹² is h^d

$$p_R \approx p(\vec{x})V_R$$

Let $p_R = \frac{k_R}{N}$, we determine the probability of making observations in region R by counting the samples in $R (= k_R)$ and dividing by the total number of samples. Note: p_R is also called the "relative frequency"

$$p(\vec{x}) = \frac{p_R}{V_R} = \frac{k_R}{V_R N}$$

Lets write the parzen window estimator as a function of a kernel³ $k(\vec{x}; \vec{x_i})$, then

$$p(\vec{x}) = \frac{1}{h^d N} \sum_{i=1}^{N} k(\vec{x}; \vec{x_i})$$

 $where^4$

$$k(\vec{x}; \vec{x_i}) = \begin{cases} 1 & \text{when } \frac{|x_{i,k} - x_k|}{h} \le \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

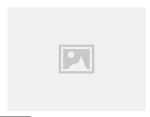
equivalently, if we use a (multivariate) gaussian kernel:

$$k(\vec{x}; \vec{x_i}) = \frac{1}{(2\pi)^d |\Sigma|} e^{-(\vec{x} - \vec{x_i})^T \Sigma^{-1} (\vec{x} - \vec{x_i})}$$

A note on applications

- General remark: We obtain a continuous pdf, i.e. desity estimation converts a list of measurments to a statistical model
- Specific example: We can sample from a pdf. This means that we have a princeple way of generating new / more / ... data that behaves / looks / ... similary to the observations.

Q: How can we (practically) sample from a pdf?



 $^{^{1}\}int_{R} d\vec{x}$ is just the volume of R 2 We also write V_{R} for the volume

 $^{^3}$ Omit h^d if the kernel is gaussian

 $^{^4}ec{x_i}$ and $ec{x}$ are not father apart then 0.5h in any dimension k

Compute through discretisation of the pdf cdf[i] = cdf[i-1] + pdf[i]. Then draw a uniformly distributed number (r) between 0 and 1. The sampled value is x where cdf[x] = r

Q: How can we determine a good window / kernel width h? Lets do ML est. with a cross-vaslidation (cv) (e.g. leave-one-sample-out cv)

$$p_{h,N-1}^{j}(\vec{x}) = \frac{1}{h^d N} \sum_{i=1(i \neq j)}^{N} k(\vec{x}; \vec{x_i})$$

We estimate the pdf from all samples except $\vec{x_j}$. $\vec{x_i}$ will be used to evaluate the quality of the pdf using window siye h.

Q: How do the results change with varing window size?



$$\hat{h} = \argmax_{h} L(h) = \argmax_{h} \prod_{j=1}^{N} p_{h,N-1}^{j}(\vec{x_{j}}) = \argmax_{h} \sum_{j=1}^{N} \log p_{h,N-1}^{j}(\vec{x_{j}})$$

The position of the maximum does not change, because the logarithm is a strictly monotonic function.

2 Mean Shift Algorithm

Purpose: Find maximum in pdf without actually performing a full density estimation.

Potential applications: Clustering, segmentation, ...

Assume that we have a full density estimator.

$$p(\vec{x}) = \frac{1}{N} \sum_{i=1}^{N} k(\vec{x}; \vec{x_i})$$

Idea: Maxima can be found, where the gradient of the pdf is zero.

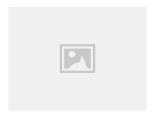


Figure 1: The kernel size indirectly controls the number of indentified maxima



Figure 2: One of the issues is, the case when a zero gradient is just between two finer maxima