Parametric and Nonparametric Generative Methods. MCMC.

Denis Derkach, Maksim Artemev, Artem Ryzhikov

CS HSE faculty, Generative Models, spring 2020

Contents

Nonparametric Density Estimation

Problem Statement

We have n iid rv $X_1, \ldots, X_n \sim F$, where F is a distribution function with density p. We need to estimate p at x, i.e. construct $\hat{p}_n(x) = \hat{p}_n(x; X_1, \ldots, X_n)$, without making any assumption about its functional form.

NB: Here we try to esimate $p(x)$ not separating the set into x's and $y's$. We will also have an example of using the methods for classification.

You can see for example R Duda et al, Pattern Classification, Chap 4.

Density Estimation

For one trial we can estimate the probability P that a given vector drawn from unknown distribution $p(x)$ inside a region R:

$$P = \int_R p(x')dx'$$

For n data points from iid measurements the probability to have k of them inside region R:

$$P(k) = \binom{n}{k} P^k (1-P)^k$$

Some Considerations

- Expected value of k/n is thus

$$\mathbb{E}(k/n) = P$$

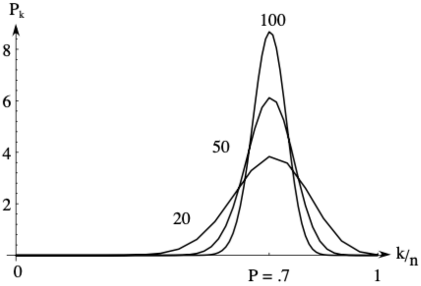
- Variance:

$$\mathbb{V}ar(k/n) = \frac{P(1-P)}{n}$$

We need to have the best estimation of $p(x)$ at a point x

Approximations Needed

Approximation I:
P is only reproduced well in case of $n \to \infty$:



Approximation II:
R should be very small such that we can say that $p(x) \approx$ const:

$$\int_R p(x')dx' \simeq p(x)V,$$

where x is some point within R and V is the volume enclosed by R.

Elementary Volume Estimation

We thus have

$$p(x) \simeq \frac{k/n}{V},$$

where we need:

- R is large enough to contain enough datapoints k;
- R is small enough to have $p(x)$ approximately constant.

Convergence Study

If we form a sequence of regions $R_1, R_2, \dots$ containing x. R_i has volume V_i and contains k_i samples. Than the n:-th estimate $\hat{p}(x)$ of $p(x)$ is

$$\hat{p}_n(x) \simeq \frac{k_n/n}{V_n}.$$

with conditions:

- $\lim_{n\to\infty} V_n = 0$;
- $\lim_{n\to\infty} k_n = \infty$;
- $\lim_{n\to\infty} k_n/n = 0$.

Choice of Optimal Strategy

For $\hat{p}_n(x) \simeq \frac{k_n/n}{V_n}$ to be optimal we can:

- fix the volume V_n and determine k_n from data (kernel density estimation method);
- fix the value k_n and determine V_n from data (k-nearest neighbours estimation method).

L2 Risk function

Definition

For a given estimate $\hat{p}_n(x)\, \forall x \in \mathbb{R}$, we can write out risk function based on Mean Integrated Squared Error:

$$MISE(\hat{p}_n, p) = \mathbb{E}_p \left[\int_{\mathbb{R}} (\hat{p}_n(x) - p(x))^2 dx \right].$$

This function can be used to estimate the optimal parameters for the best convergence rate.

NB: This is not the only risk function. Others may be based on L_p norm or different divergences.

Risk function for convergence

The Risk function can be rewritten as:

$$MISE(\hat{p}_n, p) = \int\limits_{\mathbb{R}} bias^2(x)dx + \int\limits_{\mathbb{R}} \mathrm{Var_p\hat{p}_n(x)dx}$$

we thus can thing of it as the representation of bias-variance trade-off.
The optimal parameter will minimise MISE.

Empirical Risk

In practice, it's hard to estimate the minimising parameter, since it normally depends on the unknown $p(x)$. Instead, since

$$\int_{\mathbb{R}} (\hat{p}_n(x) - p(x))^2 dx = \int_{\mathbb{R}} \hat{p}_n(x)^2 dx - 2 \int_{\mathbb{R}} \hat{p}_n(x) p(x) dx + \int_{\mathbb{R}} p(x)^2 dx,$$

it is enough to minmise

$$\mathcal{J}(h) = \int_{\mathbb{R}} \hat{p}_n(x)^2 dx - 2 \int_{\mathbb{R}} \hat{p}_n(x) p(x) dx.$$

Cross-validation for risk function

We can write out

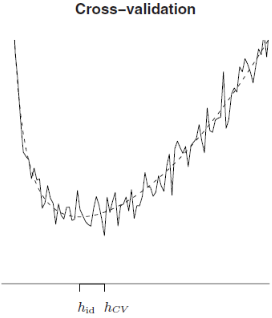
$$\hat{\mathcal{J}}(h) = \int\limits_{\mathbb{R}} [\hat{p}_n(x)]^2 dx - \frac{2}{n}\sum_{i=1}^{n} \hat{p}_{(-i)}(X_i),$$

where $\hat{p}_{(-i)}$ - is an estimate without $i-$th observation.
We will than have:

$$\mathbb{E}\hat{\mathcal{J}}(h) \approx \mathbb{E}\mathcal{J}(h).$$

Cross-validation for optimal paameters

Typically $\hat{\mathcal{J}}(h)$ will look like:



Thus instead of unknown $MISE$ we can minimise $\hat{\mathcal{J}}(h)$ and find optimal θ_{cv}, which will be close to optimal θ_{MISE}.

Histogram definition

The easiest way to estimate density is using histogram.
Consider interval $[a, b) \ni X_1, \dots X_n$.
Divide it into M equal parts Δ_i of the size $h = \frac{b-a}{M}$:

$$\Delta_i = [a + ih, a + (i+1)h), i = 0, 1, \dots, M - 1].$$

Let k_i - number of measurements inside Δ_i;

$$\hat{p}_n(x) = \frac{1}{nh} \sum_{i=0}^{M-1} k_i \mathbb{I}\{x \in \Delta_i\}$$

Note: with $x \in \Delta_i$ and small h:

$\mathbb{E}_p \hat{p}_n(x) = \frac{\mathbb{E}\nu_j}{nh} = \frac{\int_{\Delta_j} p(u)du}{h} \approx \frac{p(x)h}{h} = p(x)$

Histogram: Smoothing Parameter choice

Let us choose h - smoothing parameter
For $x_0 \in \Delta_j$:

$$bias(x_0) = \mathbb{E}_p \hat{p}_n(x_0) - p(x_0) = \frac{1}{h}\int\limits_{\Delta_j} p(x)dx - \frac{1}{h}\int\limits_{\Delta_j} p(x_0)dx =$$

$$= \frac{1}{h}\int\limits_{\Delta_j} (p(x) - p(x_0))dx \approx \frac{1}{h}\int\limits_{\Delta_j} p'(x_0)(x - x_0)dx \approx$$

$$\approx p'(x_0)[a + (j + \frac{1}{2})h - x_0]$$

Smoothing Parameter choice

$$\int_a^b bias^2(x_0)dx_0 = \sum_{j=0}^{N-1} \int_{\Delta_j} bias^2(x_0)dx_0 =$$

$$= \sum_{j=0}^{N-1} \int_{\Delta_j} [p'(x_0)]^2[a + (j + \frac{1}{2})h - x_0]^2 dx_0 \approx$$

$$\approx \sum_{j=0}^{N-1} [p'(a + (j + \frac{1}{2})h)]^2 \int_{\Delta_j} (a + (j + \frac{1}{2})h - x_0)^2 dx_0$$

$$= \sum_{j=0}^{N-1} [p'(a + (j + \frac{1}{2})h)]^2 \left(- \frac{(a + (j + \frac{1}{2})h - x_0)^3}{3} \right) \Bigg|_{\Delta_j} \approx$$

$$\approx \left(\int_a^b [p'(x)]^2 dx \right) \frac{h^2}{12}.$$

Smoothing Parameter choice

$$\mathbb{V}ar_p\hat{p}_n(x_0) = \mathbb{V}ar_p\frac{\nu_j}{nh} = \frac{1}{(nh)^2}\mathbb{V}ar_p\nu_j =$$

$$= \frac{1}{(nh)^2}n\int\limits_{\Delta_j} p(x)dx(1 - \int\limits_{\Delta_j} p(x)dx) \approx \frac{1}{nh^2}\int\limits_{\Delta_j} p(x)dx$$

$$\int\limits_a^b \mathbb{V}ar_p\hat{p}_n(x_0)dx_0 = \sum_{j=0}^{N-1}\left(\frac{1}{nh^2}\int\limits_{\Delta_j} p(x)dx\right)h =$$

$$= \frac{1}{nh}\int\limits_a^b p(x)dx = \frac{1}{nh}$$

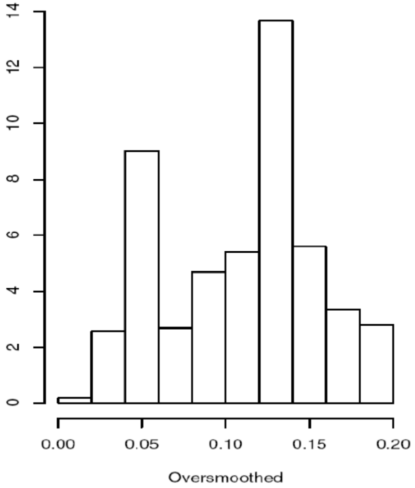
MISE: Smoothing Parameter choice

Thus,

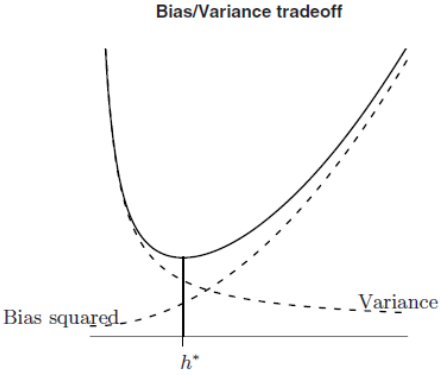
$$MISE(\hat{p}_n, p) = \left(\int\limits_{\mathbb{R}} [p'(x)]^2 dx \right) \frac{h^2}{12} + \frac{1}{nh}$$

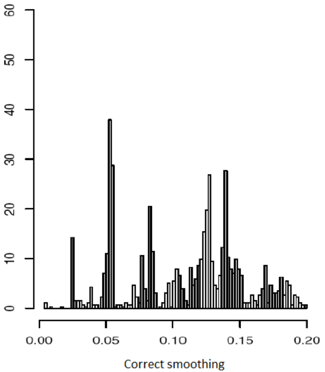
The bigger h, the bigger bias we have and the smaller variance.
If we have too large h - oversmoothing, too low - undersmoothing.

Example of non-optimal choice



Example of optimal choice





Minimisation of MISE

Optimal h can be obtained by analysing previous equation:

$$h^* = \frac{1}{n^{\frac{1}{3}}} \left(\frac{6}{\int\limits_{\mathbb{R}} [p'(x)]^2 dx} \right)^{\frac{1}{3}}.$$

Which means:

$$MISE(\hat{p}_n, p) \approx \frac{C}{n^{\frac{2}{3}}},\ \text{где}\ C = \left(\frac{3}{4}\right)^{\frac{2}{3}} \left(\int\limits_{\mathbb{R}} [p'(x)]^2 dx \right)^{\frac{1}{3}}.$$

Thus with optimal h, $MISE$ for histogram is converging at a rate $n^{-\frac{2}{3}}$.

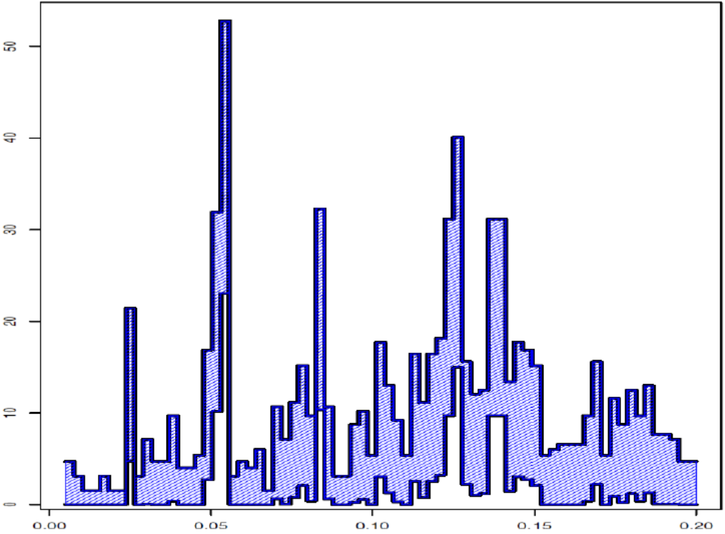
Confidence belt

If M_n - number of bins in the histogram that provides $\hat{p}_n$ estimate, with $M_n \to \infty$ and $\frac{M(n)\log(n)}{n} \to \infty$ for $n \to \infty$.
For

$$p_-(x) = (max\{\sqrt{\hat{p}_n(x)} - C, 0\})^2, p_+(x) = (\sqrt{\hat{p}_n(x)} + C)^2,$$

where $C = \frac{1}{2} z_{\frac{\alpha}{2M}} \sqrt{\frac{M}{n(b-a)}}$
Than $(p_-(x), p_+(x))$ is $1 - \alpha$ confidence belt for $\hat{p}_n$.

Confidence belt for histogram density



Summary: histogram estimates

- Relatively efficient in memory (does not store dataset).
- Can be built sequentially.
- Obtained estimate is not smooth.
- For higher dimensions the convergence is quite slow.

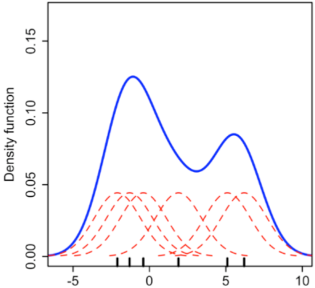
Kernel density estimation

The problem of smoothness is coming from non-smooth definition of bin. Can we keep the same approach (fixing V_n) but make the estimate smoother?

Definition

Kernel density estimate looks like:

$$\hat{p}_n(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right), h \text{ — } \underline{\text{bandwidth}}$$



Types of Kernels

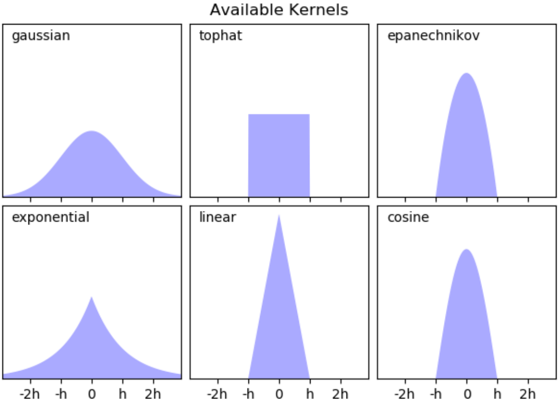
Remember, that we need $V_n \to 0$ with $n \to \infty$. Thus, the volume of kernel must fall faster than $1/n$.

Definition

Kernel - function K such that

$$K(x) \geq 0, \int\limits_{\mathbb{R}} K(x)dx = 1, \int\limits_{\mathbb{R}} xK(x)dx = 0, \sigma_K^2 \equiv \int\limits_{\mathbb{R}} x^2 K(x)dx$$

Kernel examples



Asymptotically choice of K influences the quality of estimate to a much smaller extent than bandwidth choice h (although it still does).

Picture credit:http://scikit-learn.org/stable/modules/density.html

Parzen(-Roseblatt) window

Note that the classical definition of Parzen window method includes a "Kernel" that looks more like a histogram:

$$K(x) = \begin{cases} 1, |x - h| < h/2, \\ 0, \text{otherwise.} \end{cases}$$

In this way we loose smoothness of estimate. That is why, in many modern books Parzen window method is somehow equivalent to KDE.

Bandwidth choice

We can again estimate bias and variance:

$$bias(x) = \mathbb{E}_p \hat{p}_n(x) - p(x) \frac{1}{2} \sigma_K^2 h^2 p''(x)$$

thus

$$\int_{\mathbb{R}} (bias(x))^2 dx = \frac{1}{4} \sigma_K^4 h^4 \int_{\mathbb{R}} [p''(x)]^2 dx.$$

In the same way:

$$\int_{\mathbb{R}} \mathbb{V}ar_p \hat{p}_n(x) dx = \frac{1}{nh} \int_{\mathbb{R}} K^2(z) dz$$

Bandwidth Choice

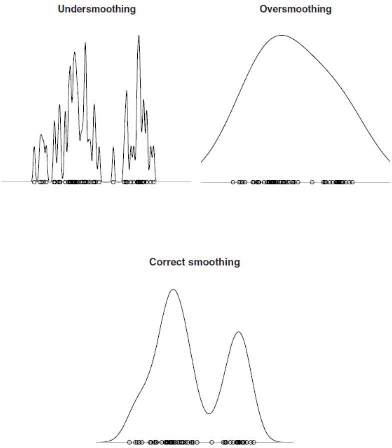
$$MISE(\hat{p}_n, p) \approx \frac{1}{4}\sigma_K^4 h^4 \int\limits_{\mathbb{R}} (p''(x))^2 dx + \frac{1}{nh} \int\limits_{\mathbb{R}} (K(x))^2 dx$$

is minimised for $h = h^*$:

$$h^* = \left(\frac{1}{n} \frac{\int\limits_{\mathbb{R}} (K(x))^2 dx}{\left(\int\limits_{\mathbb{R}} x^2 K(x) dx \right)^2 \left(\int\limits_{\mathbb{R}} p''(x))^2 dx \right)} \right)^{\frac{1}{5}}$$

Thus $MISE(\hat{p}_n, p) = O\left(n^{-\frac{4}{5}}\right)$. Generally, one can prove that the convergence cannot be faster than this.

Oversmoothing vs undersmoothing



We have the same problems like in histogram case, when we have undersmoothing and oversmoothing.

Multi-dimension

We can estimate h^* for d dimensional problem, and obtain $MISE = O(n^{-\frac{4}{4+d}})$, thus, the convergence rate is faster than histogram. However, both are slower than the MSE of a MLE $\mathcal{O}(n^1)$. This reduction of error rate is the price we have to pay for a more flexible model (we do not assume the data is from any particular distribution but only assume the density function is smooth).

Reminder

For $\hat{p}_n(x) \simeq \frac{k_n/n}{V_n}$ to be optimal we can:

- fix the volume V_n and determine k_n from data (kernel density estimation method);
- fix the value k_n and determine V_n from data (k-nearest neighbours estimation method).

Let's check what we can do with the second strategy.

k-nearest neighbour method

For a given point x and $R_k(x)$ denoting the distance from x to its k-th nearest neighbour point, the kNN density estimator estimates the density by

$$\hat{p}_{knn}(x) = \frac{k}{n}\frac{1}{V_d R_k(x)},$$

with the latter term taking into account the volume of a d-dimensional ball with radius being $R_k(x)$ and V_d is the volume of d-dimensional ball.

kNN: example

Let our data is 1D $X = \{1, 2, 6, 11, 13, 14, 20, 33\}$. What is the kNN density for $k = 2$ at $x = 5$?

- The distances to 5 are $\{4, 3, 1, 6, 8, 9, 15, 28\}$. Thus $R_2(5) = 3$.
- We can estimate:

$$\hat{p}_{knn}(5) = \frac{2}{8}\frac{1}{2R_2(x)} = \frac{1}{24},$$

- Note that for $k = 5$, $\hat{p}_{knn}(5) = \frac{5}{64}$, which is quite different.

Thus, we have a strong dependency on k. It have to be chosen such that $p(x)$ is approximately constant in every ball.

kNN: bias and variance

For 1D problem MISE can be estimated:

$$MISE(\hat{p}_{knn}(x)) = \mathcal{O}(\frac{k^4}{n^4} + \frac{1}{k}).$$

With optimal $k = C_0 n^{4/5}$ we can estimate:

$$MISE(\hat{p}_{knn}(x)) = \mathcal{O}(n^{-4/5}).$$

Multidimensional estimation

For d-dimensional problem bias will be:

$$bias(\hat{p}_{knn}(x)) = \mathcal{O}(\left(\frac{k}{n}\right)^{2/d} + \frac{1}{k}).$$

Variance will be:

$$\mathbb{V}ar(\hat{p}_{knn}(x)) = \mathcal{O}(\frac{1}{k}).$$

This is very different from the KDE approach, while it keeps the same MISE convergence rate. The idea is that the variance of estimate will depend only on k due to the fact that we cover k events always. But since we do not limit the distance, we can bias our estimate quite significantly.

Summary so far

- Nonparametric methods are efficient in low dimensional estimation, but not as efficient as parametric.
- One can control the convergence rate for bias or variance of estimate, but the sum will have very similar convergence speed.
- One can also introduce basis approach, where $\hat{p}(x)$ is obtained in series of basis functions.
- There are methods that are mixture of KDE and kNN, where one can obtain intermediate results.
- In order to speed up the convergence, once can analyse manifolds in the d-dimension.

Gaussian Mixture Model

Parametric methods

We have seen in Lecture 1 that a parametric method can be used to create a generative model. Let us recall this.
We have a set of iid rv $X_1, \ldots, X_n \sim P$, where P is the underlying population CDF and it has a PDF p.
We can assume that the distribution is Gaussian and thus will obtain two parameters using Maximum Likelihood Estimate:

$$\hat{\mu} = \bar{X}_n; \hat{\sigma}^2 = S_n^2 = \frac{1}{n-1} \sum (X_i - \bar{X}_n)^2$$

This will create a pdf estimate:

$$\hat{p}_n(x) = \frac{1}{\sqrt{2\pi\hat{\sigma}_n^2}} e^{-\frac{1}{2\pi\hat{\sigma}_n^2}(x-\hat{\mu}_n)}.$$

There is a big problem however.

Convergence of parametric methods

We converge to $\bar{p}(x)$ and in general there is no guarantee that $\bar{p}(x) = p(x)$.
However, we can check the convergence, we will obtain that:

$$\hat{p}(x) - \bar{p}(x) = \mathcal{O}(1/\sqrt{n}).$$

Much faster than the nonparametric! A very nice property that we want to keep.

Gaussian Mixture Model

We can use a mixture of distributions (like, Gaussians) to have a better estimate of true PDF. Gaussian Mixture Model proposes:

$$p_{GMM} = \sum_{l=1}^{K} \pi_l \phi(x; \mu_l, \sigma_l^2),$$

where $\pi_l \geq 0$ are the weights: $\sum \pi_l = 1$. Here the number K is a tuning parameter that specifies the number of Gaussians in our model.

GMM estimate

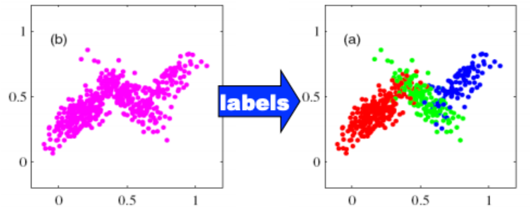
We thus have $3K-1$ parameters. Estimation of them can be done using Maximum Likelihood Estimate:

$$\hat{\pi}_1, \hat{\mu}_1, \sigma_1^2, \dots \sigma_K^2 = \arg\max_{\hat{\pi}_i, \hat{\mu}_i, \sigma_i^2} \sum_{i=1}^{n} \log \left(\sum_{i=1}^{K} \pi_i \phi(x; \mu_i, \sigma_i^2) \right)$$

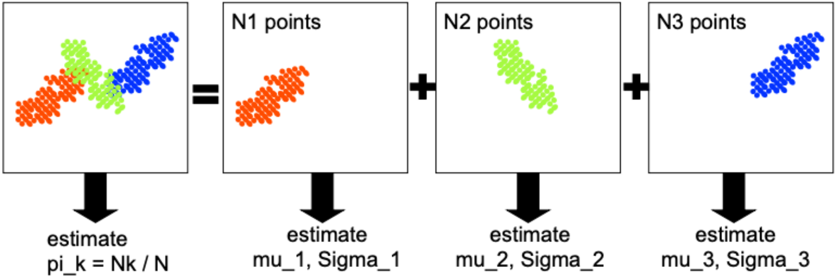
Things get worth, when we go to multidimension, instead of σ^2, we need to estimate matrix Σ.

Note, that GMM is easy to interpret as a signal that comes from different sources. However, it's hard to fit the likelihood - EM algorithm should be used.

EM for GMM: idea



And we can easily estimate each Gaussian, along with the mixture weights!



Suppose some oracle told us which point comes from which Gaussian.

EM for GMM: insight

Since we don't know the latent variables, we instead take the expected value of the log likelihood with respect to their posterior distribution P(z|x,theta). In the GMM case, this is equivalent to "softening" the binary latent variables to continuous ones (the expected values of the latent variables)

$$\ln p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \underbrace{z_{nk}}_{\text{unknown discrete value 0 or 1}} \{\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \Sigma_k)\}$$

$$\mathrm{E}_{\mathbf{z}}[\ln p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})] = \sum_{n=1}^{N} \sum_{k=1}^{K} \underbrace{\gamma_k(\mathbf{x}_n)}_{\text{known continuous value between 0 and 1}} \{\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \Sigma_k)\}$$

Where $\gamma_j(\mathbf{x}_n)$ is $P(z_{nk} = 1)$

EM-algorithm for GMM

E $$\gamma_j(\mathbf{x}_n) = \frac{\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_k \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$ ownership weights

M

$$\boldsymbol{\mu}_j = \frac{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)\mathbf{x}_n}{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)}$$ means

$$\boldsymbol{\Sigma}_j = \frac{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)(\mathbf{x}_n - \boldsymbol{\mu}_j)(\mathbf{x}_n - \boldsymbol{\mu}_j)^\top}{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)}$$ covariances

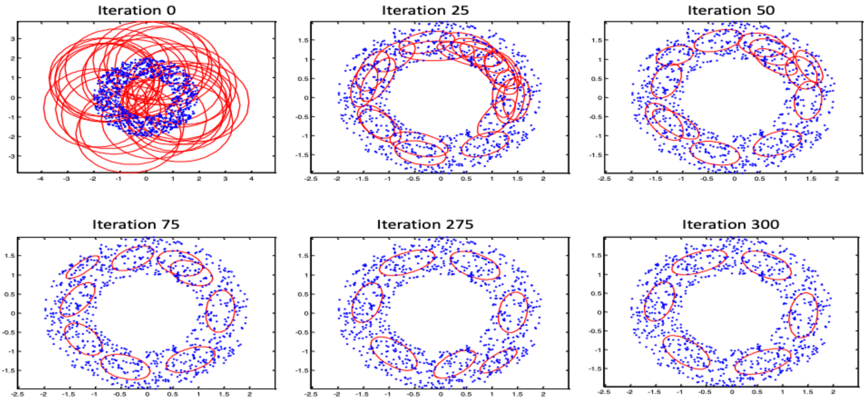
$$\pi_j = \frac{1}{N}\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)$$ mixing probabilities

EM graphics

Insert video here.

GMM: example

Training set: $n = 900$ examples from a uniform pdf inside an annulus, model: GMM with $K = 30$ Gaussian components



GMM shortcomings

- Identifiability problem. We cannot distibguish between two exchanged solutions.
- Computation problem. We need to use EM algorithm to find solution.
- Choice of K. A very difficult task, one may use a model selection technique to choose it, however, no simple rule exists.

Summary of classical density estimation

Type	Method	Convergence rate	Tuning parameter	Limitation
Parametric	Parametric model	$O\left(\frac{1}{\sqrt{n}}\right)$	None	Unavoidable bias
	Mixture model	$O\left(\frac{1}{\sqrt{n}}\right)$	K, number of mixture	Hard to compute
Nonparametric	Histogram	$O\left(\frac{1}{n^{1/3}}\right)$	b, bin size	Lower convergence rate
	Kernel density estimator	$O\left(\frac{1}{n^{2/5}}\right)$	h, smoothing bandwidth	
	K-nearest neighbor	$O\left(\frac{1}{n^{2/5}}\right)$	k, number of neighbor	
	Basis approach	$O\left(\frac{1}{n^{2/5}}\right)$	M, number of basis	

You can see for example Yen Chi Chen, Learning Theory, Lec 8.