Probabilistic Machine Learning:
5. Bayesian Nonparametric Models

Tomasz Kajdanowicz, Piotr Bielak, Maciej Falkiewicz, Kacper Kania, Piotr Zieliński

Department of Computational Intelligence
Wroclaw University of Science and Technology

The presentation has been inspired and in some parts totally based on

1. Gershman, Samuel J., and David M. Blei. "A tutorial on Bayesian nonparametric models." Journal of Mathematical Psychology 56.1 (2012): 1-12
2. https://www.surajx.in/2019/07/bayesian-optimization---part-1-stochastic-processes/
3. Ghahgramani, Z., A Tutorial on Gaussian Processes (slides)
4. https://peterroelants.github.io/posts/gaussian-process-tutorial/

Table of Contents

Pre-reading

1. Gershman, Samuel J., and David M. Blei. "A tutorial on Bayesian nonparametric models." Journal of Mathematical Psychology 56.1 (2012): 1-7

Table of Contents

Introduction to bayesian nonparametric

Most data scientist ask questions:

- how many classes in mixture model should I use?
- how many factors should I use in factor analysis?

Then answer such questions and:

- fit several models, with different numbers of clusters or factors
- select one using model comparison metrics:
  1. how well the model fits the data
  2. complexity penalty, favors simpler models

Bayesian nonparametric (BNP):

- do not compare models that vary in complexity
- fit a single model that can adapt its complexity to the data
- allow the complexity to grow as more data are observed

BNP vs Bayesian

BNP:

- hidden structure is assumed to grow with the data
- model complexity is part of the posterior distribution
- structure is determined as part of analyzing the data

Bayesian:

- hidden structure is static
- structure is not modeled within posterior distribution
- structure must be specified in advance

Given the parameters θ, future predictions, x, are independent of the observed data, D:

$$P(x|\theta, D) = P(x|\theta)$$

(therefore θ capture everything there is to know about the data)

Non-parametric models

Two common model families:

- mixture models
- latent factor models

There are multiple suited to many data types: sequential, grouped, trees, relational, spatial, etc.

- Non-parametric models assume that the data distribution cannot be defined in terms a finite set of parameters. But they can often be defined by assuming an infinite dimensional θ.
- Usually we think of θ as a function
- The amount of information that θ can capture about the data D can grow as the amount of data grows

Table of Contents

RV recap

Random Variables (RV) are deterministic functions that maps all possible outcomes of a probabilistic experiment to some mathematical space, like $\mathbb{R}$.

Definition

Probability Space. For some probabilistic experiment, let Ω - called the sample space, denote the set of all possible outcomes for the experiment. Let F denote a collection of subsets of Ω, and let $P : F \to [0, 1]$ - called the probability measure, be a function that assigns to each element in F a value between $[0, 1]$ that is consistent with the **axioms of probability**. Then the triplet (Ω, F, P) is called a probability space.

Definition

Random Variable. Given a probability space (Ω, F, P), a Random Variable X is a function from Ω to a measurable space E. That is,

$$X : \Omega \to E$$

Stochastic Process (SP)

Stochastic Process:

- is a.k.a. Random Process
- represents a collection of Random Variables
- originally *process* was meaning *something that is a function of time* (SP primarily employed for investigating the evolution of probabilistic systems over time)
- generally, the process need not have anything to do with time

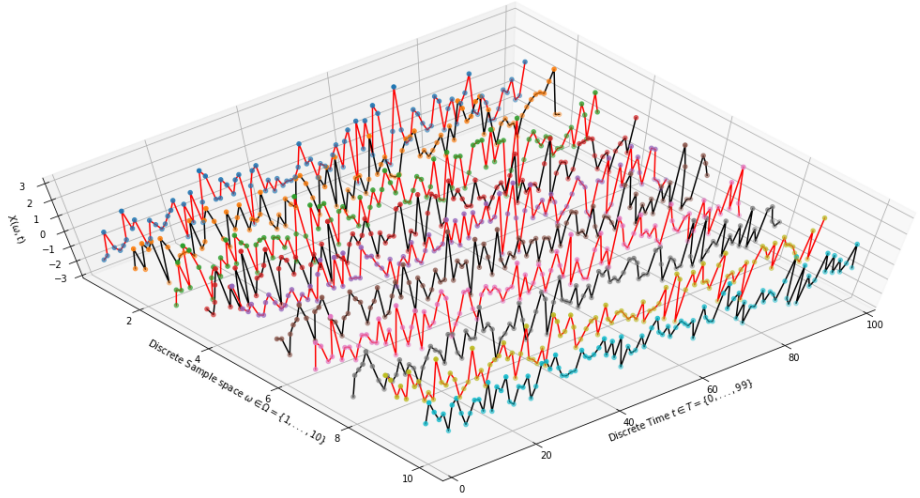
Definition

Stochastic Process. Given a probability space (Ω, F, P), a Stochastic Process, X is both a collection of Random Variables all taking values in the same measurable space E, as well as a mapping from the sample space Ω to a space of functions.

When X is represented as a collection indexed by a set T, it can be written as, $\boxed{X = \{X_t : t \in T\}}$, where X_t is an E-valued RV.
Alternatively, when X is represented as a mapping to a space of functions, it can be written as $\boxed{X : \Omega \to E^T}$

Example SP

- experiment collects samples as the system evolves over time
- each experiment run gives a sample path
- sample path is be modelled as a sequence of random variables, $X_1 = (Y_1, Y_2, \ldots)$
- the process is modelled as a family of random sequences $X = \{X_1, X_2, \ldots, X_n\}$



Index Set

index set T is a set that is used to index the RVs in a stochastic process.

- can be any valid mathematical set
- it is preferred that T is linearly ordered
- when T is countable, e.g. $\mathbb{N}$ or $1, \ldots, N$, the random process is **discrete**
- when T is uncountable, like $\mathbb{R}$ or $[0, 1]$, the random process is **continuous**

State Space

The state space of a random process are the values that the RVs in the collection can take.

- state space is discrete if it is countable $\rightarrow$ discrete-valued stochastic process
- state space is continuous if it is uncountable $\rightarrow$ continuous-valued stochastic process

Stationarity

If all the random variables in a stochastic process is identically distributed then the process is said to be **stationary** (distribution of the system from which we are sampling does not change over time).

Key takeaway from stochastic processes

- a stochastic process is a collection of random variables
- it can also be viewed as a probability distribution over a space of functions
- sampling a stochastic process gives a you a function which is a single time-dependent realization of the process

Table of Contents

Exchangeable Sequences

Definition

Exchangeable sequence. A sequence is exchangeable if its joint distribution is invariant under arbitrary permutation of the indices:

$$(X_1, X_2, \ldots) = (X_{\pi(1)}, X_{\pi(2)}, \ldots) \forall \pi \in S_\infty$$

Theorem

De Finetti's theorem. *$(X_i)_{i \in N}$ is exchangeable if and only if there exists a random probability measure θ on X such that $X_1, X_2, \ldots | \theta \propto$ iid θ*
($X_1, X_2, \ldots$ are conditionally independent given θ)

Interpretation:

- any probabilistic model of data which assumes that the order of the data does not matter, can be expressed as a Bayesian mixture of *iid* models.
- θ may in general need to be infinite dimensional (i.e. nonparameteric)

Example

Document processing and information retrieval

To highlight the difference between iid and infinitely exchangeable sequences, consider that

- assume that search engine use bag-of-words model
- this implies that the order of words in a document does not matter
- neverthelles, the words are definitely not iid

- If we see one word and it is a French word, we then expect that the rest of the document is likely to be in French.
- If we see the French words voyage (travel), passeport (passport), and douane (customs), we expect the rest of the document to be both in French and on the subject of travel.

Since we are assuming infinite exchangeability, there is some θ governing these intuitions. Thus, we see that θ can be very rich, and it seems implausible that θ might always be finite-dimensional. θ can be a stochastic process.

Table of Contents

Model representation

In **finite** dimensions, a probability model is usually defined by:

- density function
- or probability mass function

Infinite dimensional spaces

In infinite dimensional spaces, this approach is not generally feasible, we have to choose alternative mathematical representations

- weak distributions
- explicit representations
- implicit representations
- finite representations

Weak distribution

Weak distribution is a representation for the distribution of a stochastic process i.e. for a **probability distribution on an infinite-dimensional space**.

- assume that the dimensions of the space are indexed by $t \in T$
- the stochastic process can be regarded as the joint distribution P of an infinite set of random variables $\{X_t\}, t \in T$
- for any finite subset $S \subset T$ of dimensions, the joint distribution P_S of the corresponding subset $X_t, t \in S$ of random variables is a finite-dimensional marginal of P

The weak distribution of a stochastic process is the set of all its finite-dimensional marginals, that is, the set $\{P_S : S \subset T, |S| < \infty\}$.
Properties:

- this representation is generally not generative, because it represents the distribution rather than a random draw
- widely applicable

Weak distribution: example

The customary definition of the Gaussian process as an infinite collection of random variables

- each finite subset of Gaussian process that has a joint Gaussian distribution is an example of a weak distribution representation

Explicit representations

- directly describe a random draw from a stochastic process (rather than describing its distribution)
- example: stick-breaking representation of the Dirichlet process

Stick-breaking process This approach involves generating a random vector with a $Dir(\alpha)$ distribution by iteratively breaking a stick of length 1 into k pieces in such a way that the lengths of the k pieces follow a $Dir(\alpha)$ distribution.

1. Break the stick in u_1 position such that $u_1 \sim Beta(alpha_1, alpha_2 + alpha_3)$. Set $q_1 = u_1$ and the rest of the stick has $(1 - u_1)$ length.
2. Take the remaining of the stick and break it from $u_2 \sim Beta(alpha_2, alpha_3)$. Now, $q_2 = u_2 * (1 - u_1)$ and finally the last part has $q_3 = 1 - (q - 1 + q_2)$.

Stick breaking - Example

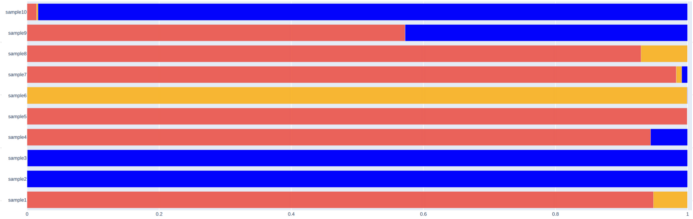


Figure: Sparse results with $\alpha = [0.1, 0.1, 0.1]$

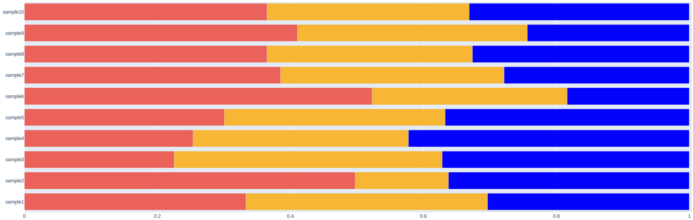


Figure: $\alpha = [10, 10, 10]$

Implicit representations

Any exchangeable sequence $X_1, \ldots, X_n$ uniquely defines a stochastic process θ, called the de Finetti measure, making the X_i's iid. If the X_i's are sufficient to define the rest of the model and their conditional distributions are easily specified, then it is sufficient to work directly with the X_i's and have the underlying stochastic process implicitly defined.

- Chinese restaurant process (an exchangeable distribution over partitions) - Dirichlet process as the de Finetti measure
- Indian buffet process (an exchangeable distribution over binary matrices) - beta process corresponding de Finetti measure

Implicit representations are useful in practice as they can lead to simple and efficient inference algorithms.

Example - Chinese restaurant process

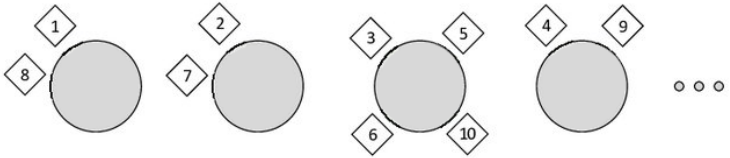
We will define a distribution on the space of partitions of the positive integers, $\mathbb{N}$. This would induce a distribution on the partitions of the first n integers, for every $n \in N$.

- Imagine a restaurant with countably infinitely many tables, labelled $1, 2, \ldots$
- customers walk in and sit down at some table according to the process:
  1. the first customer always chooses the first table
  2. the nth customer chooses the first unoccupied table with probability $\frac{\alpha}{n-1+\alpha}$
  3. or he chooses an occupied table with probability $\frac{c}{n-1+\alpha}$, where c is the number of people sitting at that table

α is a scalar parameter of the process

Example - Chinese restaurant process

A possible arrangement of 10 customers:



Let z_i be the table occupied by the customer i.
The probability of this arrangement is:

$$P(z_1, \ldots, z_{10}) = P(z_1)P(z_2|z_1)\ldots P(z_{10}|z_1, \ldots, z_9)$$
$$= \frac{\alpha}{\alpha} \cdot \frac{\alpha}{1+\alpha} \cdot \frac{\alpha}{2+\alpha} \cdot \frac{\alpha}{3+\alpha} \cdot \frac{1}{4+\alpha} \cdot \frac{2}{5+\alpha} \cdot \frac{1}{6+\alpha} \cdot \frac{1}{7+\alpha} \cdot \frac{1}{8+\alpha} \cdot \frac{3}{9+\alpha}$$

Example - Chinese restaurant process

Observations:

1. The probability of a seating is invariant under permutations. Permuting the customers permutes the numerators in the above computation, while the denominators remains the same. This property is known as **exchangeability**.
2. seating arrangement creates a partitions
3. the probability of any seating arrangement of 10 customers where **three** tables are occupied, with **three** customers each on **two** of the tables and the remaining **four** on the **third** table, will have the same probability as the seating in the example
4. the expected number of occupied tables k_n for n customers grows logarithmically

$$E[k_n|\alpha] = \alpha \log n, \text{ as } n \to \infty$$

Finite representation

Finite representation

This representation of Bayesian nonparametric models is as the **infinite limit** of finite (parametric) Bayesian models.

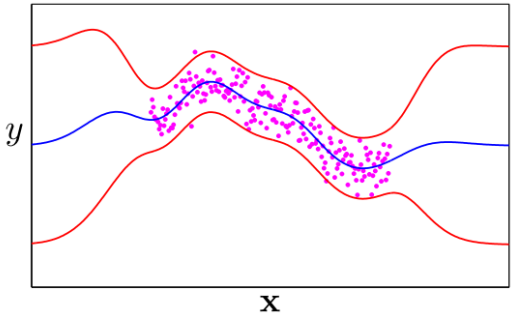
- Dirichlet Process mixtures can be derived as the infinite limit of finite mixture models with particular Dirichlet priors on mixing proportions
- Gaussian Process can be derived as the infinite limit of particular Bayesian regression models with Gaussian priors
- Beta Process can be derived as from the limit of an infinite number of independent beta variables

- these representations are sometimes more intuitive for practitioners familiar with parametric models
- not all Bayesian nonparametric models can be expressed in this fashion

Table of Contents

Nonlinear regression

Consider the problem of nonlinear regression:
You want to learn a **function f** with error bars from data $D = \{X, y\}$



A **Gaussian process** defines a distribution over functions $p(f)$ which can be used for Bayesian regression:

$$p(f|D) = \frac{p(f)p(D|f)}{p(D)}$$

Gaussian Processes

A Gaussian process defines a distribution over functions, $p(f)$, where f is a function mapping some input space X to $\mathbb{R}$.

$$f : X \to \mathbb{R}$$

f can be an infinite-dimensional quantity (e.g. if $X = \mathbb{R}$)

Let $\mathbf{f} = (\mathrm{f}(\mathrm{x}_1), \ldots, \mathrm{f}(\mathrm{x}_n))$ be an n-dimensional vector of function values evaluated at n points $x_i \in X$. Note $\mathbf{f}$ is a random variable.

Definition

$p(f)$ is a **Gaussian process** if for any finite subset $\{x_1, \ldots, x_n\} \subset X$, the marginal distribution over that finite subset $p(f)$ has a multivariate Gaussian distribution.

Gaussian process covariance functions (kernels)

Gaussian processes are distributions over functions $f(x)$ of which the distribution is defined by a mean function $m(x)$ and positive definite covariance function (or kernel) $k(x, x')$, with x the function values and (x,x') all possible pairs in the input domain:

$$f(x) \sim GP(m(x), k(x, x'))$$

Gaussian processes (GPs) are parameterized by:

- **mean function,** $m(x)$
- **covariance function, or kernel,** $k(x, x')$

where any finite subset $X = \{x_1, \ldots, x_n\}$ of the domain of x in the **marginal distribution** is a multivariate Gaussian distribution:

$$f(X) \sim N(m(X), k(X, X))$$

Multivariate vs process

- multivariate Gaussian caputures a finte number of jointly distributed Gaussians
- the Gaussian process doesn’t have this limitation
- mean and covariance are defined by a function !!!
- each input to this function is a variable correlated with the other variables in the input domain, as defined by the covariance function
- since functions can have an infinite input domain, the Gaussian process can be interpreted as an infinite dimensional Gaussian random variable

Covariance function as prior

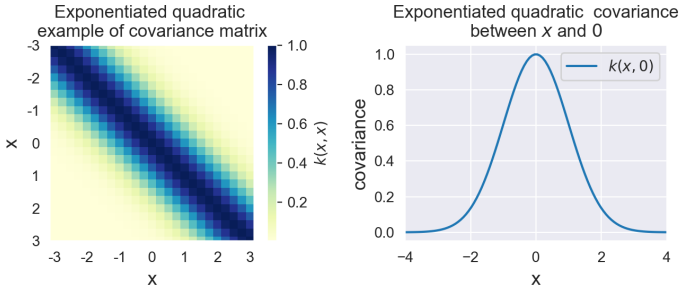
Covariance function:

- models the joint **variability** of the Gaussian process random variables
- also known as the kernel function
- it returns the modelled covariance between each pair in x_a and x_b
- implies a distribution over functions $f(x)$
- by choosing a specific kernel function k it is possible to set prior information on this distribution
- kernel function needs to be positive-definite in order to be a valid covariance function

Example kernel

exponentiated quadratic covariance function (also known as the RBF kernel)

$$k(x_a, x_b) = \exp\left(-\frac{1}{2\sigma^2}\|x_a - x_b\|^2\right)$$



- the covariance vs input zero is plotted on the right
- exponentiated quadratic covariance decreases exponentially the further away the function values x are from eachother

Sampling from prior

In practice:

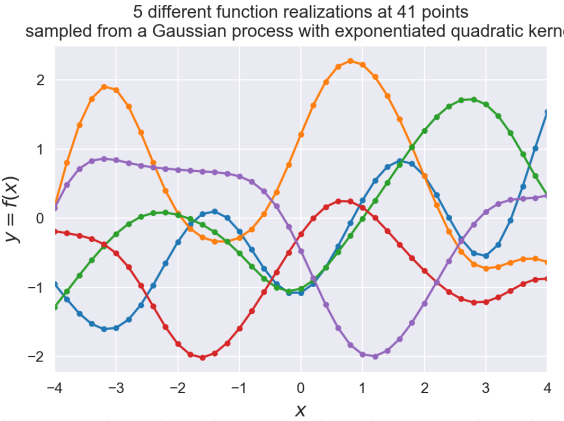
- we can't just sample a full function evaluation f from a Gaussian process distribution
- it would mean evaluating **m(x)** and k(x,x') at an infinite number of points
- as x can have an infinite domain

We can however sample function evaluations y of a function f drawn from a Gaussian process at a finite, but arbitrary, set of points X : $y = f(X)$

A finite dimensional subset of the Gaussian process distribution results in a **marginal distribution** that is a Gaussian distribution $y \sim N(\mu, \Sigma)$ with mean vector $\mu = m(X)$ and covariance matrix $\Sigma = k(X, X)$

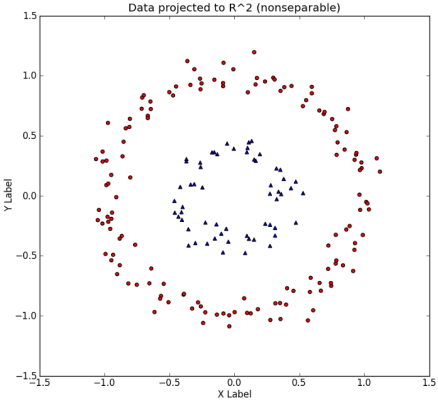
Sampling from prior

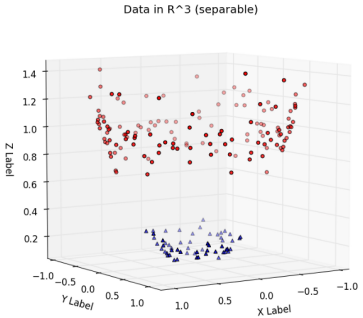
- sample of 5 different function realisations from a Gaussian process with exponentiated quadratic prior without any observed data
- done by drawing correlated samples from a 41-dimensional Gaussian $N(0, k(X, X))$ with $X = [X_1, \ldots, X_{41}]$



Kernel trick

- can the functions drawn from the Gaussian process distribution be non-linear?
- the non-linearity is because the kernel can be interpreted as implicitly computing the **inner product** in a different space than the original input space (e.g. a higher dimensional feature space)
- this is commonly known as the **kernel trick**





Gaussian processes for regression

- Gaussian processes model distributions over functions
- thus we can use them to build regression models
- let treat Gaussian process as a prior defined by the kernel function and create a posterior distribution given some data
- posterior distribution can then be used to predict the expected value and probability of the output variable y given input variables X

AIM:make predictions $y_2 = f(X_2)$ for n_2 new samples based on our Gaussian process prior and n_1 previously observed data points (X_1, y_1)

y_1 and y_2 are jointly Gaussian since they both should come from the same function

Gaussian processes for regression

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right)$$

where

$$\begin{aligned} \mu_1 &= m(X_1) && (n_1 \times 1) \\ \mu_2 &= m(X_2) && (n_2 \times 1) \\ \Sigma_{11} &= k(X_1, X_1) && (n_1 \times n_1) \\ \Sigma_{22} &= k(X_2, X_2) && (n_2 \times n_2) \\ \Sigma_{12} &= k(X_1, X_2) = \Sigma_{21}^T && (n_1 \times n_2) \end{aligned}$$

Σ_{11} is independent of Σ_{22} and vice versa

Conditional distribution

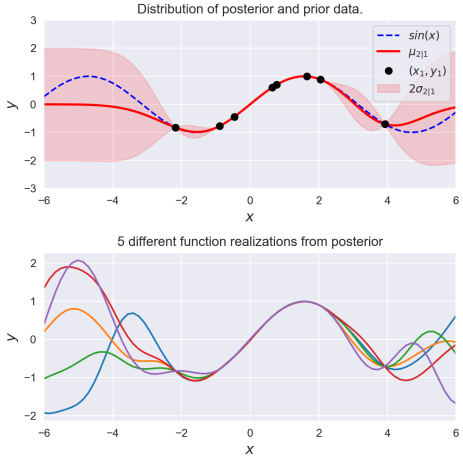
$$
\begin{aligned}
p(y_2|y_1, X_1, X_2) &= N(\mu_{2|1}, \Sigma_{2|1}) \\
\mu_{2|1} &= \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1) \\
&= \Sigma_{21}\Sigma_{11}^{-1}y_1 \quad \text{(if assume mean prior } \mu = 0\text{)} \\
\Sigma_{2|1} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}
\end{aligned}
$$

since $\Sigma_{11} = \Sigma_{11}^T$ we can write:

$$
\begin{aligned}
\mu_{2|1} &= \Sigma_{21}\Sigma_{11}^{-1}y_1 \\
&= (\Sigma_{11}^{-1}\Sigma_{12})^T y_1
\end{aligned}
\qquad
\begin{aligned}
\Sigma_{2|1} &= \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \\
&= \Sigma_{22} - (\Sigma_{11}^{-1}\Sigma_{12})^T\Sigma_{12}
\end{aligned}
$$

- prediction of y_2 corresponding to the input samples X_2 done by using the mean $\mu_{2|1}$ of the resulting distributions
- mean of the posterior predictions of a Gaussian process are weighted averages of the observed variables
- weighting is based on the covariance function k

Example



follow https://distill.pub/2019/visual-exploration-gaussian-processes/ for further insights

Gaussian process classification vs regression

Regression:

- likelihood function was Gaussian
- a Gaussian process prior combined with a Gaussian likelihood gives a Gaussian process posterior over functions
- everything remains analytically tractable

Classification:

- in classification models the targets are discrete class labels
- the Gaussian likelihood is inappropriate

Kernel density estimation (KDE)

- recall the Gaussian mixture model that is a parametric density estimator for data in $\mathbb{R}^D$
- it requires specifying the number K and locations μ_k of the clusters
- an alternative to estimating the μ_k is to allocate one cluster center per data point, so $\mu_k = \mathbf{x}_i$

$$\mathrm{P}(\boldsymbol{x}|D) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}|\boldsymbol{x_i}, \sigma^2\boldsymbol{I}) \tag{1}$$

and generalized by:

$$\mathrm{P}(\boldsymbol{x}|D) = \frac{1}{N}\sum_{i=1}^{N}\kappa(\boldsymbol{x} - \boldsymbol{x_i}) \tag{2}$$

kernel density estimator (KDE) a.k.a. Parzen window density estimator

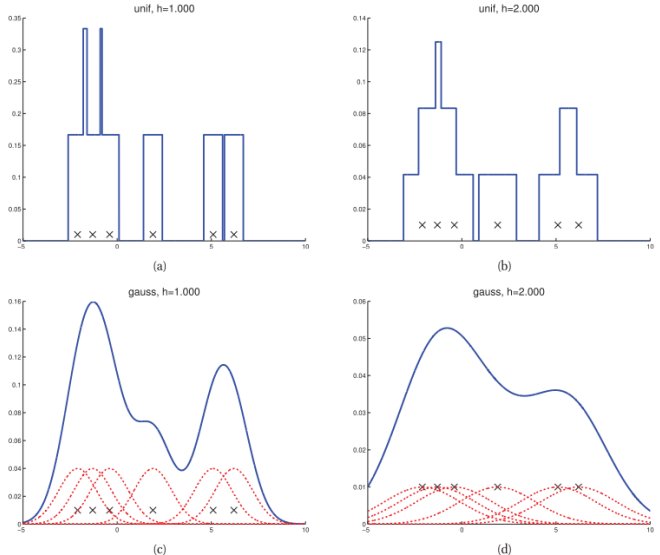
KDE features

- no model fitting is required (advantage over a parametric models)
- no need to pick K
- takes a lot of memory to store, and a lot of time to evaluate
- require to indicate the the interval at which we perform estimation - usually picked by estimate minimization (e.g. such as cross validation) of some frequentist risk

KDE - Example

Top: uniform kernel - equivalent to a histogram estimate of the density, shows how many data points land within an interval of size *h* around x_i

Down: gaussian



Hidden Markov Model

If you forgot about HMMs watch "A friendly introduction to Hidden Markov Models by Luis Serrano" https://www.youtube.com/watch?v=kqSzLo9fenk.