Computational Statistics
Gaussian Processes

Luca Bortolussi

Department of Mathematics and Geosciences
University of Trieste

Office 238, third floor, H2bis
luca@dmi.units.it

Trieste, Winter Semester 2015/2016

Outline

KERNELS

- The notion of kernel comes from the theory of integral operators on a space $\mathcal{X}$ with measure μ. A real kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defines an integral operator T_k (applied to integrable f) as:

$$(T_k f)(\mathbf{x}) = \int_{\mathcal{X}} k(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) d\mu(\mathbf{y})$$

- A kernel is positive semidefinite if, for all $f \in L_2(\mathcal{X}, \mu)$:

$$\int_{\mathcal{X} \times \mathcal{X}} k(\mathbf{x}, \mathbf{y}) f(\mathbf{x}) f(\mathbf{y}) d\mu(\mathbf{x}) d\mu(\mathbf{y}) \geq 0$$

- Equivalently, a kernel is positive (semi)definite if for any collection of n points $\{\mathbf{x_i} \mid i = 1, \ldots, n\}$, the Gram matrix K, $K_{ij} = k(\mathbf{x_i}, \mathbf{x_j})$ is positive (semi)definite (Mercer's theorem).
- The Gram matrix of a symmetric kernel, $k(\mathbf{x}, \mathbf{y}) = k(\mathbf{y}, \mathbf{x})$, is symmetric.

EIGENFUNCTIONS

- An eigenfunction ϕ with eigenvalue λ of k satisfies

$$(T_k\phi)(\mathbf{y}) = \int_{\mathcal{X}} k(\mathbf{x},\mathbf{y})\phi(\mathbf{x})d\mu(\mathbf{x}) = \lambda\phi(\mathbf{y})$$

- There can be an infinite number of eigenfunctions, which can be ordered w.r.t. decreasing eigenvalues, and they can be chosen orthogonal, i.e. such that $\int \phi_i(\mathbf{x})\phi_j(\mathbf{x})d\mu(\mathbf{x}) = \delta_{ij}$
- A kernel can be decomposed using eigenfunctions:

Theorem 4.2 *(Mercer's theorem). Let $(\mathcal{X},\mu)$ be a finite measure space and $k \in L_\infty(\mathcal{X}^2,\mu^2)$ be a kernel such that $T_k : L_2(\mathcal{X},\mu) \to L_2(\mathcal{X},\mu)$ is positive definite (see eq. (4.2)). Let $\phi_i \in L_2(\mathcal{X},\mu)$ be the normalized eigenfunctions of T_k associated with the eigenvalues $\lambda_i > 0$. Then:*

1. *the eigenvalues $\{\lambda_i\}_{i=1}^\infty$ are absolutely summable*
2. $$k(\mathbf{x},\mathbf{x}') = \sum_{i=1}^{\infty} \lambda_i \phi_i(\mathbf{x})\phi_i^*(\mathbf{x}'), \tag{4.37}$$

holds μ^2 almost everywhere, where the series converges absolutely and uniformly μ^2 almost everywhere. □

Reproducing Kernel Hilbert Spaces

Definition 6.1 *(Reproducing kernel Hilbert space). Let $\mathcal{H}$ be a Hilbert space of real functions f defined on an index set $\mathcal{X}$. Then $\mathcal{H}$ is called a reproducing kernel Hilbert space endowed with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ (and norm $\|f\|_{\mathcal{H}} = \sqrt{\langle f, f \rangle_{\mathcal{H}}}$) if there exists a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ with the following properties:*

1. *for every $\mathbf{x}$, $k(\mathbf{x}, \mathbf{x}')$ as a function of $\mathbf{x}'$ belongs to $\mathcal{H}$, and*
2. *k has the* reproducing property $\langle f(\cdot), k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}} = f(\mathbf{x})$. □

See e.g. Schölkopf and Smola [2002] and Wegman [1982]. Note also that as $k(\mathbf{x}, \cdot)$ and $k(\mathbf{x}', \cdot)$ are in $\mathcal{H}$ we have that $\langle k(\mathbf{x}, \cdot), k(\mathbf{x}', \cdot) \rangle_{\mathcal{H}} = k(\mathbf{x}, \mathbf{x}')$.

The RKHS uniquely determines k, and vice versa, as stated in the following theorem:

Theorem 6.1 *(Moore-Aronszajn theorem, Aronszajn [1950]). Let $\mathcal{X}$ be an index set. Then for every positive definite function $k(\cdot, \cdot)$ on $\mathcal{X} \times \mathcal{X}$ there exists a unique RKHS, and vice versa.* □

RKHS AND EIGENFUNCTIONS

- The functions belonging to the RKHS associated with a kernel k can be written as a linear combination of the eigenfunctions ϕ_j of k: $f(\mathbf{x}) = \sum_j f_j \phi_j(\mathbf{x})$, with $\sum_j f_j^2/\lambda_j < \infty$ (this is a smoothness constraint).
- Such functions define an Hilbert space H with inner product $\langle f, g \rangle_H = \sum_j \frac{f_j g_j}{\lambda_j}$
- This Hilbert space is the RKHS corresponding to kernel k:

$$\langle f(\cdot), k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}} = \sum_{i=1}^{\infty} \frac{f_i \lambda_i \phi_i(\mathbf{x})}{\lambda_i} = f(\mathbf{x}). \tag{6.2}$$

Similarly

$$\langle k(\mathbf{x}, \cdot), k(\mathbf{x}', \cdot) \rangle_{\mathcal{H}} = \sum_{i=1}^{\infty} \frac{\lambda_i \phi_i(\mathbf{x}) \lambda_i \phi_i(\mathbf{x}')}{\lambda_i} = k(\mathbf{x}, \mathbf{x}'). \tag{6.3}$$

- Furthermore, the norm of $k(\mathbf{x}, \cdot)$ is $k(\mathbf{x}, \mathbf{x}) < \infty$: it belongs to H.

KERNEL FUNCTIONS: CLASSIFICATION

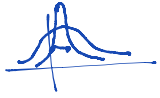
A kernel $k(\mathbf{x}, \mathbf{y})$ can be classified w.r.t dependence on $\mathbf{x}$ and $\mathbf{y}$.

- Stationary kernel: it is a function of $\mathbf{x} - \mathbf{y}$ (invariant to translations).
- Isotropic kernel: it is a function of $\|\mathbf{x} - \mathbf{y}\|$ (invariant to rigid motions).
- Dot-product kernel: it is a function of $\mathbf{x}^T\mathbf{y}$ (invariant w.r.t. rotations with respect to the origin).

Continuity properties of the GPs and kernels k.

- Continuity in mean square of a process f at $\mathbf{x}$: for each $\mathbf{x_k} \to \mathbf{x}$, it holds that $\mathbb{E}[\|f(\mathbf{x_k}) - f(\mathbf{x})\|^2] \to 0$.
- A process is continuous in m.s. at $\mathbf{x}$ iff k is continuous at $k(\mathbf{x}, \mathbf{x})$. For stationary kernels, k must be continuous at zero.
- If k is $2k$th differentiable, than f is kth differentiable (in m.s.).

GAUSSIAN KERNEL

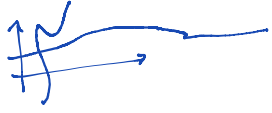


- The Gaussian or Squared Exponential kernel is defined by

$$k(\mathbf{x}, \mathbf{y}) = \alpha \exp\left[-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{\lambda^2}\right]$$

- α is called the amplitude, it regulates the magnitude of variance at each point $\mathbf{x}$. λ, instead, is the characteristic length-scale, which regulates the speed of decay of the correlation between points.
- The Gaussian kernel is isotropic and among the most used in computational statistics, and its RKHS is dense in the space of continuous functions over a compact set in $\mathbb{R}^n$!
- The Automatic-Relevance Detection Gaussian Kernel generalises the GK as

$$k(\mathbf{x}, \mathbf{y}) = \alpha \exp\left[-\sum_j \frac{|x_j - y_j|^2}{\lambda_j^2}\right]$$



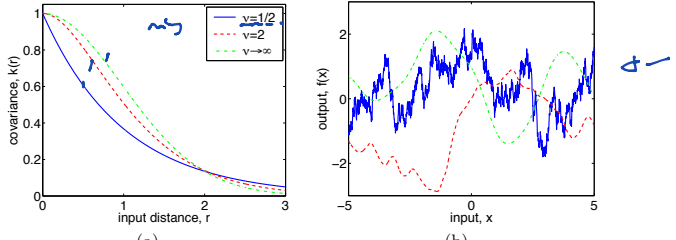
MATÉRN KERNEL

- The Matérn kernel is defined by

$$k_{\text{Matern}}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}r}{\ell}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}r}{\ell}\right), \tag{4.14}$$

with positive parameters ν and ℓ, where K_ν is a modified Bessel function

- If $\nu > h$, then the process with Matérn kernel is h times differentiable (in m.s.) For $\nu \to \infty$, then the MK becomes the GK.
- Examples of Matern Kernel:



(a) (b)

MATÉRN AND EXPONENTIAL KERNEL

- Typical choice for MK is $\nu = p + 1/2$, giving

$$k_{\nu=p+1/2}(r) = \exp\left(-\frac{\sqrt{2\nu}r}{\ell}\right)\frac{\Gamma(p+1)}{\Gamma(2p+1)}\sum_{i=0}^{p}\frac{(p+i)!}{i!(p-i)!}\left(\frac{\sqrt{8\nu}r}{\ell}\right)^{p-i}. \quad (4.16)$$

It is possible that the most interesting cases for machine learning are $\nu = 3/2$ and $\nu = 5/2$, for which

$$\begin{aligned} k_{\nu=3/2}(r) &= \left(1+\frac{\sqrt{3}r}{\ell}\right)\exp\left(-\frac{\sqrt{3}r}{\ell}\right), \\ k_{\nu=5/2}(r) &= \left(1+\frac{\sqrt{5}r}{\ell}+\frac{5r^2}{3\ell^2}\right)\exp\left(-\frac{\sqrt{5}r}{\ell}\right), \end{aligned} \quad (4.17)$$

- for $\nu = 1/2$, we get the Exponential Kernel

$$k(\mathbf{x}, \mathbf{y}) = \exp\left[-\frac{\|\mathbf{x}-\mathbf{y}\|}{\ell}\right]$$

which in one dimension corresponds to the Ornstein-Ulembeck process (the model of velocity of a particle undergoing Brownian motion), which is continuous but nowhere differentiable.

POLYNOMIAL KERNEL

- Simple dot-products kernels are the polynomial kernel, for p integer:

$$k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}')^p$$

- This corresponds to a kernel obtained by a set of polynomial basis functions:

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= (\mathbf{x} \cdot \mathbf{x}')^p = \Big(\sum_{d=1}^{D} x_d x'_d \Big)^p = \Big(\sum_{d_1=1}^{D} x_{d_1} x'_{d_1} \Big) \cdots \Big(\sum_{d_p=1}^{D} x_{d_p} x'_{d_p} \Big) \\ &= \sum_{d_1=1}^{D} \cdots \sum_{d_p=1}^{D} (x_{d_1} \cdots x_{d_p})(x'_{d_1} \cdots x'_{d_p}) \triangleq \boldsymbol{\phi}(\mathbf{x}) \cdot \boldsymbol{\phi}(\mathbf{x}'). \end{aligned} \tag{4.23}$$

- The basis functions ϕ_m are given by all monomials of degree p, i.e. $\sum m_j = p$:

$$\phi_{\mathbf{m}}(\mathbf{x}) = \sqrt{\frac{p!}{m_1! \cdots m_D!}} x_1^{m_1} \cdots x_D^{m_D}. \tag{4.24}$$

COMPOSITION OF KERNELS

Kernels can be composed according to certain rules, giving rise to new kernels.

Techniques for Constructing New Kernels.

Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$ and $k_2(\mathbf{x}, \mathbf{x}')$, the following new kernels will also be valid:

$$
\begin{aligned}
k(\mathbf{x}, \mathbf{x}') &= c k_1(\mathbf{x}, \mathbf{x}') && (6.13)\\
k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x}) k_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}') && (6.14)\\
k(\mathbf{x}, \mathbf{x}') &= q\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.15)\\
k(\mathbf{x}, \mathbf{x}') &= \exp\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.16)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') && (6.17)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') k_2(\mathbf{x}, \mathbf{x}') && (6.18)\\
k(\mathbf{x}, \mathbf{x}') &= k_3\left(\boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}')\right) && (6.19)\\
k(\mathbf{x}, \mathbf{x}') &= \mathbf{x}^{\mathrm{T}} \mathbf{A} \mathbf{x}' && (6.20)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}'_a) + k_b(\mathbf{x}_b, \mathbf{x}'_b) && (6.21)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}'_a) k_b(\mathbf{x}_b, \mathbf{x}'_b) && (6.22)
\end{aligned}
$$

where $c > 0$ is a constant, $f(\cdot)$ is any function, $q(\cdot)$ is a polynomial with nonnegative coefficients, $\boldsymbol{\phi}(\mathbf{x})$ is a function from $\mathbf{x}$ to $\mathbb{R}^M$, $k_3(\cdot, \cdot)$ is a valid kernel in $\mathbb{R}^M$, $\mathbf{A}$ is a symmetric positive semidefinite matrix, $\mathbf{x}_a$ and $\mathbf{x}_b$ are variables (not necessarily disjoint) with $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$, and k_a and k_b are valid kernel functions over their respective spaces.

Outline

MARGINAL LIKELIHOOD

- In order to do model selection (e.g. between different kernels) we can use the marginal likelihood.
- This can be used also to set hyperparameters of the kernel functions, like the amplitude or the lengthscale of the Gaussian kernel.
- For GP, we can compute the marginal likelihood analytically:

$$\mathcal{L} = \log p(\mathbf{y}|X) = \log \int p(\mathbf{f}|X) p(\mathbf{y}|\mathbf{f}, X) d\mathbf{f}$$

which gives

$$\mathcal{L} = -\frac{1}{2}\mathbf{y}^T (K + \sigma^2 I)^{-1}\mathbf{y} - \frac{1}{2}\log|(K + \sigma^2 I)| - \frac{N}{2}\log 2\pi$$

- This follows also by observing that $\mathbf{y} \sim \mathcal{N}(\mathbf{0}, K + \sigma^2 I)$.

MARGINAL LIKELIHOOD

The log marginal likelihood

$$\mathcal{L} = -\frac{1}{2}\mathbf{y}^T(K + \sigma^2 I)^{-1}\mathbf{y} - \frac{1}{2}\log|(K + \sigma^2 I)| - \frac{N}{2}\log 2\pi$$

has three terms

- $-\frac{1}{2}\mathbf{y}^T(K + \sigma^2 I)^{-1}\mathbf{y}$ is the data fit.
- $-\frac{1}{2}\log|(K + \sigma^2 I)|$ is a complexity penalty.
- $-\frac{N}{2}\log 2\pi$ is a constant.

MARGINAL LIKELIHOOD - HYPERPARAMETERS

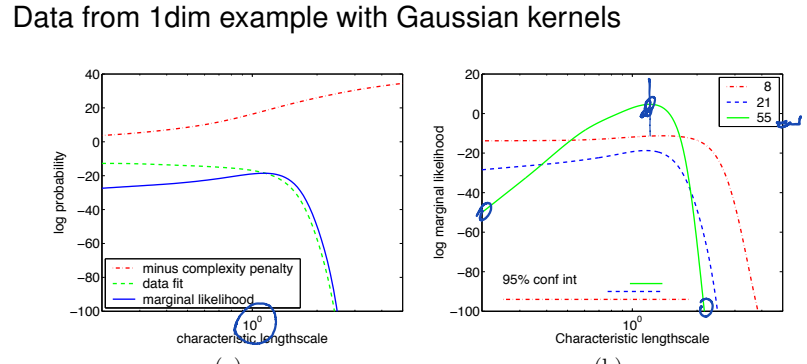


Figure 5.3: Panel (a) shows a decomposition of the log marginal likelihood into its constituents: data-fit and complexity penalty, as a function of the characteristic length-scale. The training data is drawn from a Gaussian process with SE covariance function and parameters $(\ell, \sigma_f, \sigma_n) = (1, 1, 0.1)$, the same as in Figure 2.5, and we are fitting only the length-scale parameter ℓ (the two other parameters have been set in accordance with the generating process). Panel (b) shows the log marginal likelihood as a function of the characteristic length-scale for different sizes of training sets. Also shown, are the 95% confidence intervals for the posterior length-scales.

MARGINAL LIKELIHOOD - HYPERPARAMETERS

Data from 1dim example with Gaussian kernels

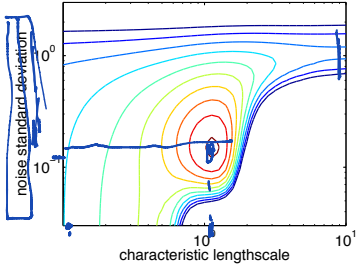


Figure 5.4: Contour plot showing the log marginal likelihood as a function of the characteristic length-scale and the noise level, for the same data as in Figure 2.5 and Figure 5.3. The signal variance hyperparameter was set to $\sigma_f^2 = 1$. The optimum is close to the parameters used when generating the data. Note, the two ridges, one for small noise and length-scale $\ell = 0.4$ and another for long length-scale and noise $\sigma_n^2 = 1$. The contour lines spaced 2 units apart in log probability density.

MARGINAL LIKELIHOOD - HYPERPARAMETERS

Data coming from a sample of a 1dim GP with Gaussian kernel and hyperparameters $\lambda = 1$, $\alpha = 1$, $\sigma = 0.1$.

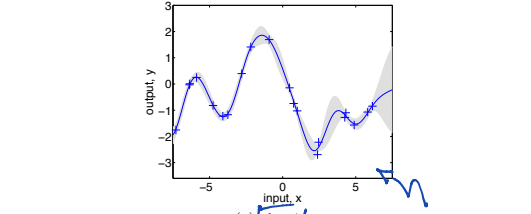


Figure 2.5: (a) Data is generated from a GP with hyperparameters $(\ell, \sigma_f, \sigma_n) = (1, 1, 0.1)$, as shown by the + symbols. Using Gaussian process prediction with these hyperparameters we obtain a 95% confidence region for the underlying function f (shown in grey). Panels (b) and (c) again show the 95% confidence region, but this time for hyperparameter values $(0.3, 1.08, 0.00005)$ and $(3.0, 1.16, 0.89)$ respectively.

HYPERPARAMETER OPTIMISATION

- In order to set the hyperparameters, we can maximise the log marginal likelihood:

$$\mathcal{L} = -\frac{1}{2}\mathbf{y}^T(K + \sigma^2 I)^{-1}\mathbf{y} - \frac{1}{2}\log|(K + \sigma^2 I)| - \frac{N}{2}\log 2\pi$$

- Its derivative w.r.t. an hyperparameter θ is

$$\begin{aligned}\frac{\partial}{\partial\theta_j}\log p(\mathbf{y}|X,\boldsymbol{\theta}) &= \frac{1}{2}\mathbf{y}^\top K^{-1}\frac{\partial K}{\partial\theta_j}K^{-1}\mathbf{y} - \frac{1}{2}\operatorname{tr}\left(K^{-1}\frac{\partial K}{\partial\theta_j}\right)\\ &= \frac{1}{2}\operatorname{tr}\left((\boldsymbol{\alpha}\boldsymbol{\alpha}^\top - K^{-1})\frac{\partial K}{\partial\theta_j}\right) \quad \text{where} \quad \boldsymbol{\alpha} = K^{-1}\mathbf{y}.\end{aligned} \tag{5.9}$$

- The derivative is relatively cheap to compute, once we invert the matrix K. Hence we can use gradient methods to optimise $\mathcal{L}$.
- Purely Bayesian methods (giving a prior on hyperparameters) are complicated by the in general complex functional form (no conjugate prior).

NON-CONSTANT PRIOR MEAN

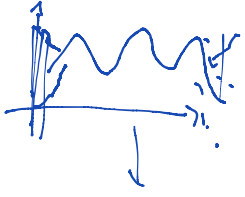
- The typical choice for the prior mean is the zero function. Data is processed by subtracting the sample mean from the observations.
- As an alternative, one can either use a deterministic function for the priori mean (and subtract if from data, adding it back to predictions), or use a generalised linear model for the prior mean:

$$g(\mathbf{x}) = f(\mathbf{x}) + \mathbf{h}(\mathbf{x})^\top \boldsymbol{\beta}, \text{ where } f(\mathbf{x}) \sim \mathcal{GP}\big(0, k(\mathbf{x}, \mathbf{x}')\big), \tag{2.39}$$

- If we put a Gaussian prior over coefficients β, we can treat them in a Bayesian way, and get a GP:

$$g(\mathbf{x}) \sim \mathcal{GP}\big(\mathbf{h}(\mathbf{x})^\top \mathbf{b}, k(\mathbf{x}, \mathbf{x}') + \mathbf{h}(\mathbf{x})^\top B \mathbf{h}(\mathbf{x}')\big), \tag{2.40}$$

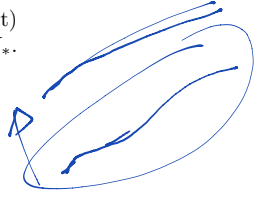
NON-CONSTANT PRIOR MEAN



- In this way, we obtain the following predictive distribution at a point $\mathbf{x}^*$:

$$\begin{aligned} \bar{\mathbf{g}}(X_*) &= H_*^\top \bar{\boldsymbol{\beta}} + K_*^\top K_y^{-1}(\mathbf{y} - H^\top \bar{\boldsymbol{\beta}}) = \bar{\mathbf{f}}(X_*) + R^\top \bar{\boldsymbol{\beta}}, \\ \text{cov}(\mathbf{g}_*) &= \text{cov}(\mathbf{f}_*) + R^\top (B^{-1} + H K_y^{-1} H^\top)^{-1} R, \end{aligned} \tag{2.41}$$

where the H matrix collects the $\mathbf{h}(\mathbf{x})$ vectors for all training (and H_* all test) cases, $\bar{\boldsymbol{\beta}} = (B^{-1} + HK_y^{-1}H^\top)^{-1}(HK_y^{-1}\mathbf{y} + B^{-1}\mathbf{b})$, and $R = H_* - HK_y^{-1}K_*$.



- The new predictive distribution has mean $H_*^T\bar{\beta}$ (from the linear model) plus a term coming from the GP model of residuals.
- Taking a flat prior (limit for $B^{-1} \to$ matrix of zeros):

$$\begin{aligned} \bar{\mathbf{g}}(X_*) &= \bar{\mathbf{f}}(X_*) + R^\top \bar{\boldsymbol{\beta}}, \\ \text{cov}(\mathbf{g}_*) &= \text{cov}(\mathbf{f}_*) + R^\top (H K_y^{-1} H^\top)^{-1} R, \end{aligned} \tag{2.42}$$

where the limiting $\bar{\boldsymbol{\beta}} = (HK_y^{-1}H^\top)^{-1}HK_y^{-1}\mathbf{y}$. Notice that predictions under

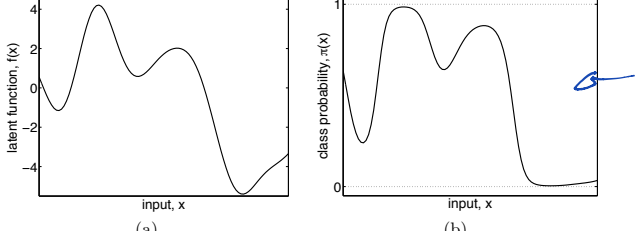
Outline

FROM LOGISTIC REGRESSION TO GP CLASSIFICATION

- The idea behind GP classification is to extend logistic (or probit) regression, by assuming the following model for the class conditionals:

$$\pi(\mathbf{x}) = p(C_1|\mathbf{x}) = \sigma(f(\mathbf{x})) \text{ where } f \sim GP(\mu, k)$$

- f is often call latent function. Note that π is a random function, as f is.



(a) (b)

Figure 3.2: Panel (a) shows a sample latent function $f(x)$ drawn from a Gaussian process as a function of x. Panel (b) shows the result of squashing this sample function through the logistic logit function, $\lambda(z) = (1 + \exp(-z))^{-1}$ to obtain the class probability $\pi(x) = \lambda(f(x))$.

GP CLASSIFICATION

- f is often call latent or nuisance function. It is not observed directly. We only observe at a point $\mathbf{x}$ the realisation of a Bernoulli random variable with probability $\pi(\mathbf{x})$.
- Inference at a test point $\mathbf{x}^*$ is done, as usual in a Bayesian setting, in two steps:
  1. Compute the posterior f^* of f at the prediction point $\mathbf{x}^*$.

$$p(f_*|X,\mathbf{y},\mathbf{x}_*) = \int p(f_*|X,\mathbf{x}_*,\mathbf{f})p(\mathbf{f}|X,\mathbf{y})\,d\mathbf{f}, \tag{3.9}$$

  with $p(\mathbf{f}|X,\mathbf{y}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|X)/p(\mathbf{y}/X)$ by Bayes theorem.
  2. Compute the predictive distribution at $\mathbf{x}^*$

$$\bar{\pi}_* \triangleq p(y_*=+1|X,\mathbf{y},\mathbf{x}_*) = \int \sigma(f_*)p(f_*|X,\mathbf{y},\mathbf{x}_*)df_* \tag{3.10}$$