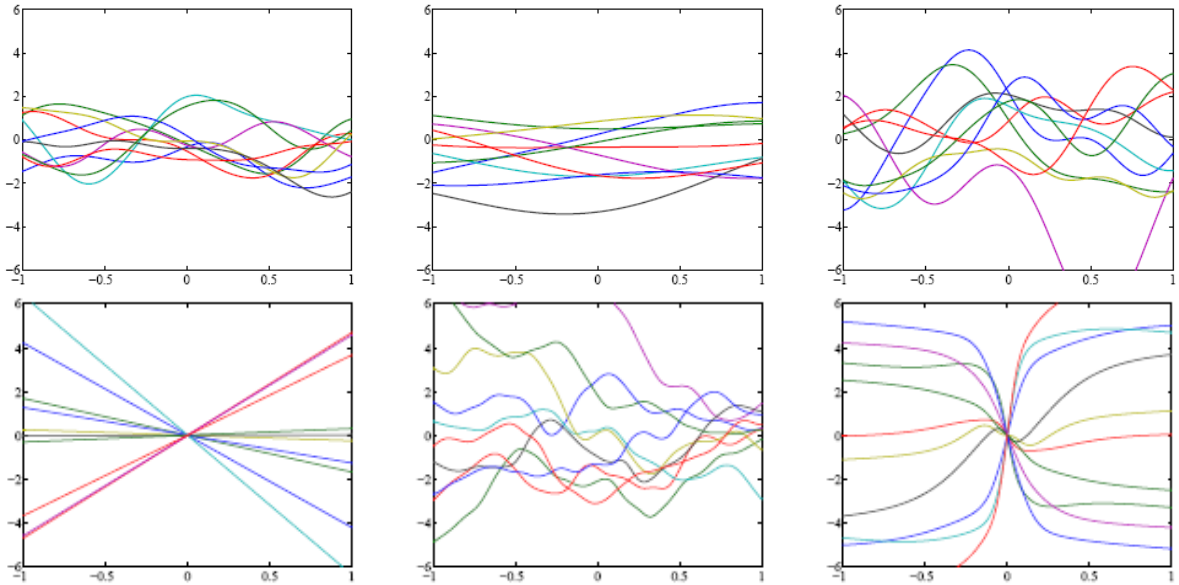
Nonparmeteric Bayes & Gaussian Processes



Baback Moghaddam

baback@jpl.nasa.gov

Machine Learning Group

Outline

- Bayesian Inference
- Hierarchical Models
- Model Selection
- Parametric *vs*. Nonparametric
- Gaussian Processes
  - regression
  - classification
- Summary

Bayesian Inference

Likelihood x *Prior*

Posterior

$$p(\theta \mid D, M) = \frac{p(D \mid \theta, M)\ p(\theta \mid M)}{p(D \mid M)}$$

Evidence

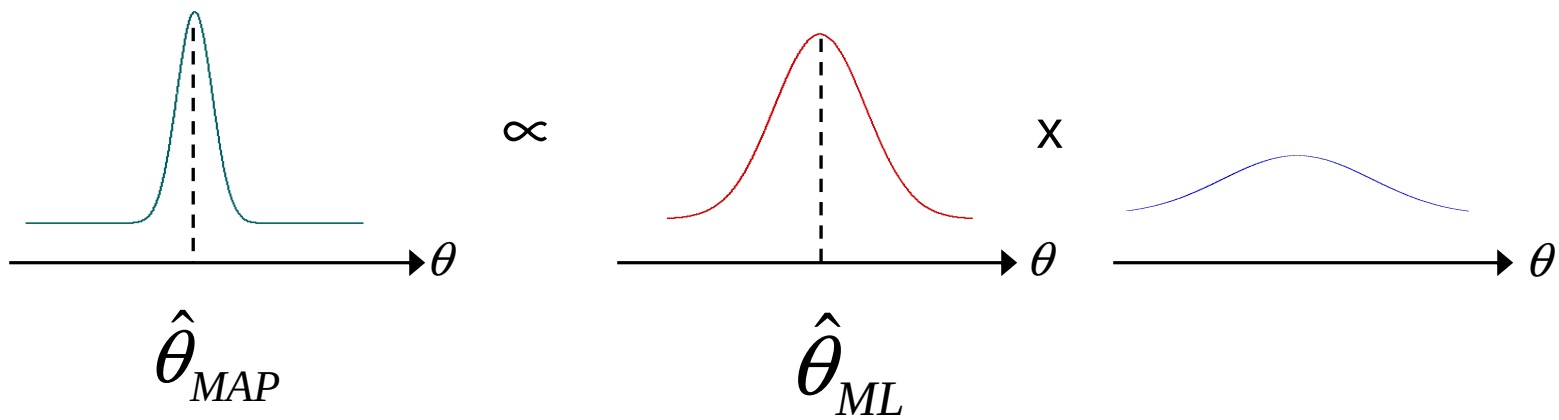
The *evidence* for our model $\boldsymbol{M}$ is also called "Marginal Likelihood'

$$p(D \mid M) = \int p(D \mid \theta, M)\ p(\theta \mid M)\, d\theta$$

Bayesian Nutshell

Posterior *Likelihood* x *Prior*

$$p(\theta \mid D, M) \propto p(D \mid \theta, M)\; p(\theta \mid M)$$



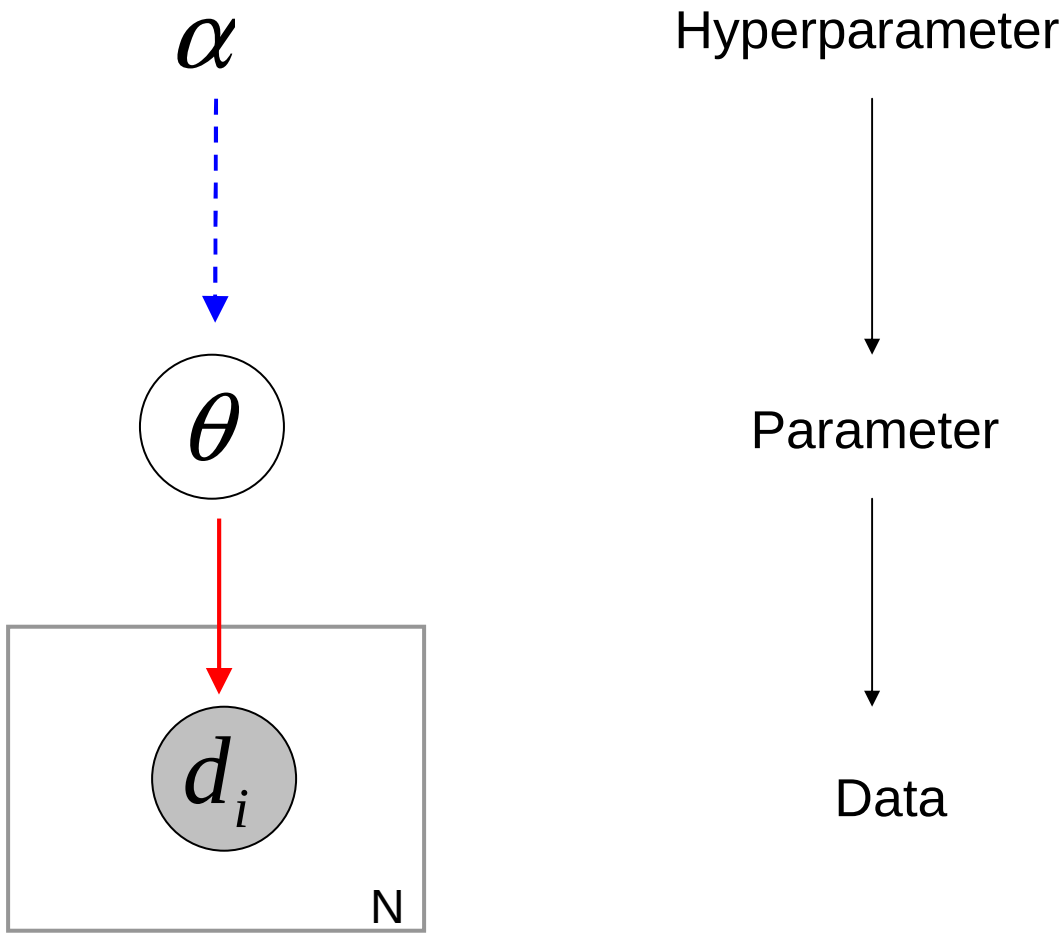
Hierarchical Models

Prior

$$p(\theta \mid \alpha, M)$$

Likelihood

$$p(D \mid \theta, M)$$



Hierarchical Models

Hyperprior

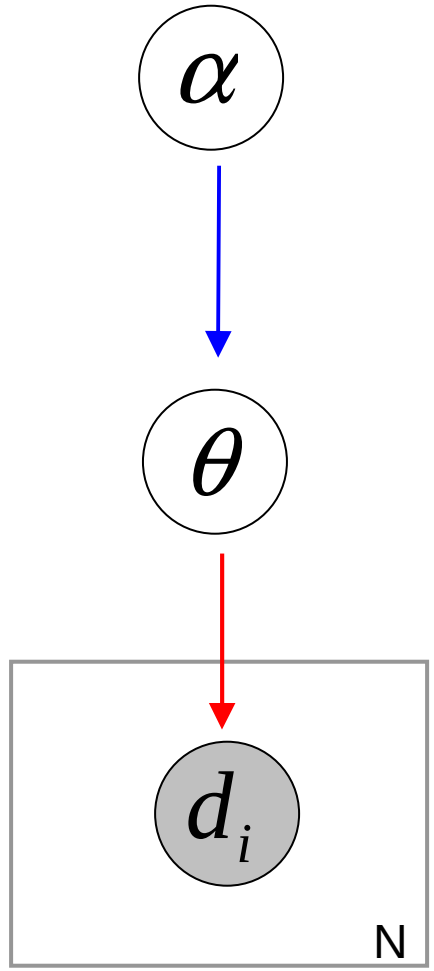
$$p(\alpha \mid M)$$

Prior

$$p(\theta \mid \alpha, M)$$

Likelihood

$$p(D \mid \theta, M)$$



Hyperparameter

Parameter

Data

Bayesian Hierarchy

- Level 1: infer parameters

$$p(\theta \mid D, \alpha, M) = \frac{p(D \mid \theta, M)\; p(\theta \mid \alpha, M)}{\boxed{p(D \mid \alpha, M)}}$$

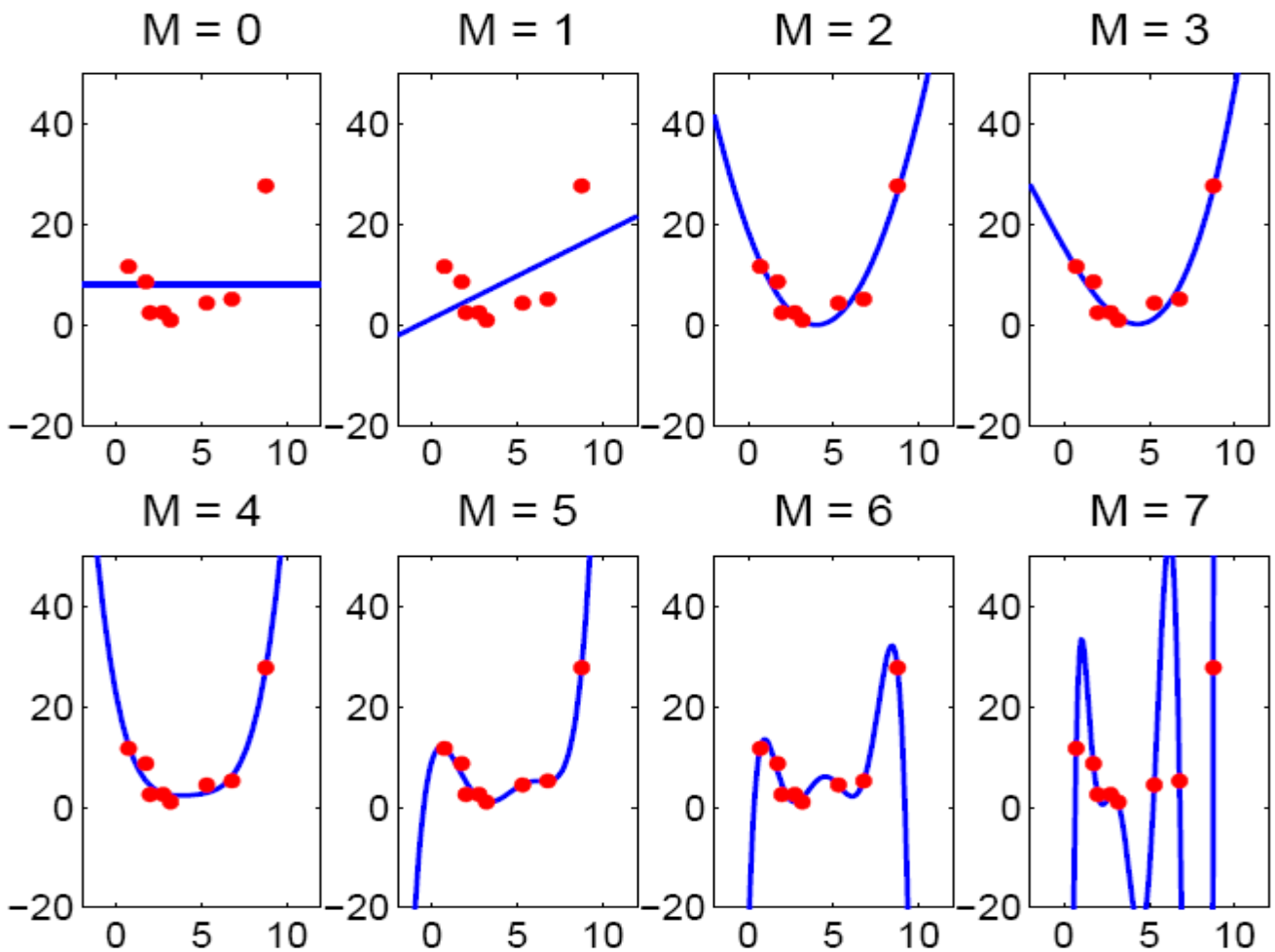
- Level 2: infer hyper-parameters

$$p(\alpha \mid D, M) = \frac{p(D \mid \alpha, M)\, p(\alpha \mid M)}{\boxed{p(D \mid M)}}$$

- Level 3: infer models

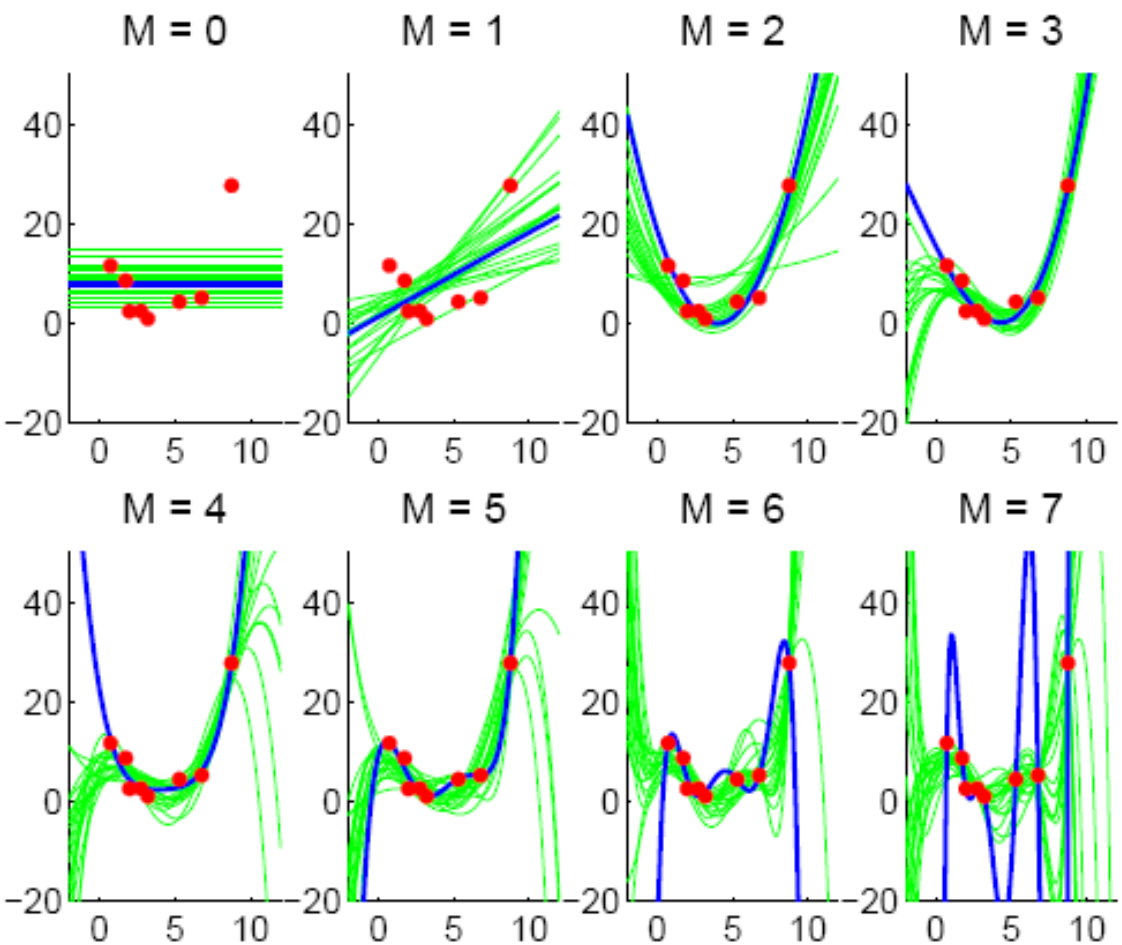
$$p(M \mid D) \propto p(D \mid M)\; p(M)$$

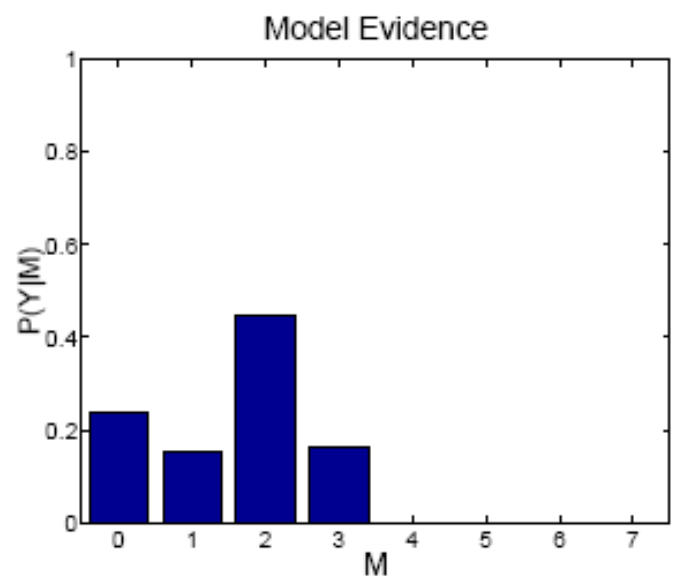
Model Selection



by *Zoubin Ghahramani*

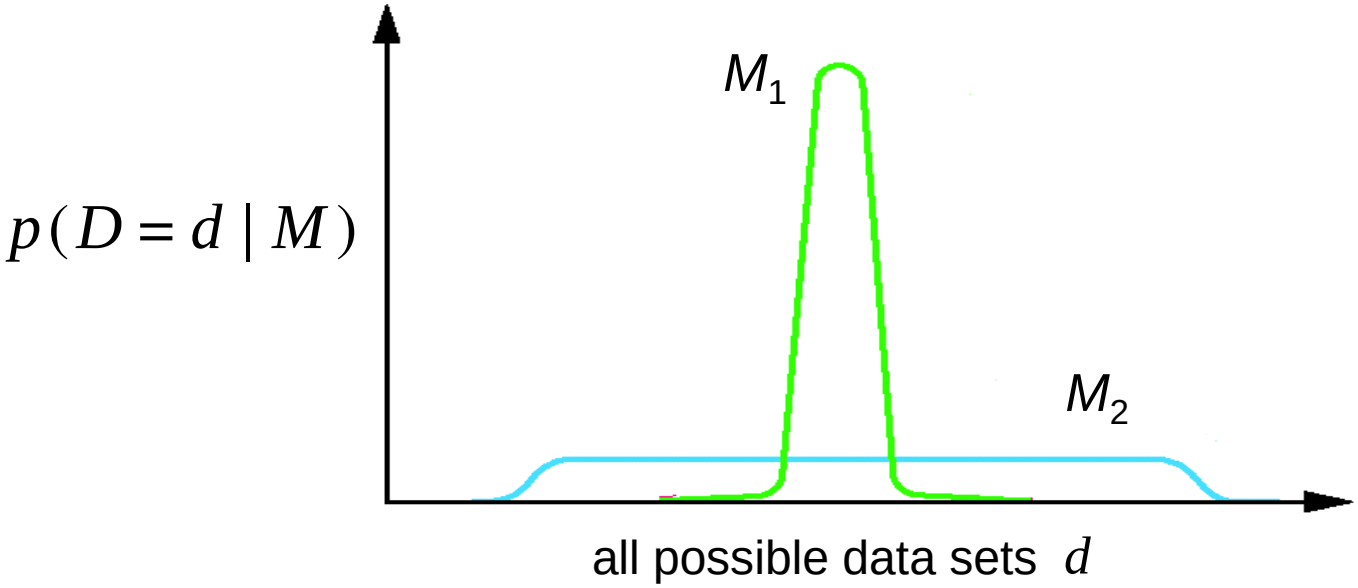
Bayesian Model Selection





by *Zoubin Ghahramani*

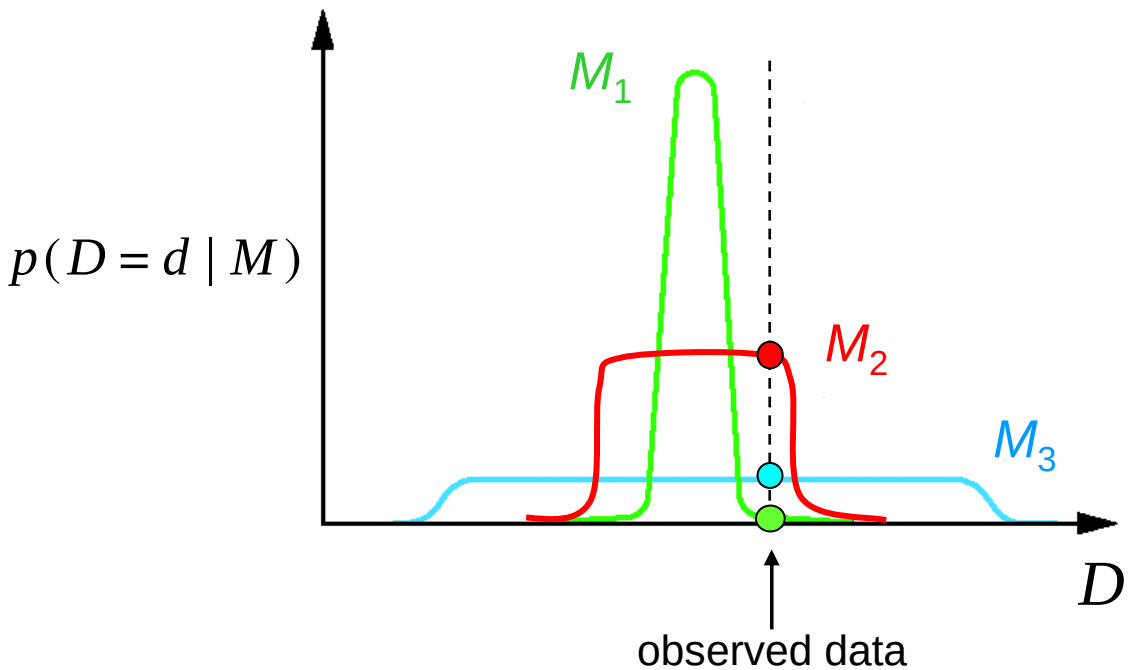
Bayesian Occam's Razor

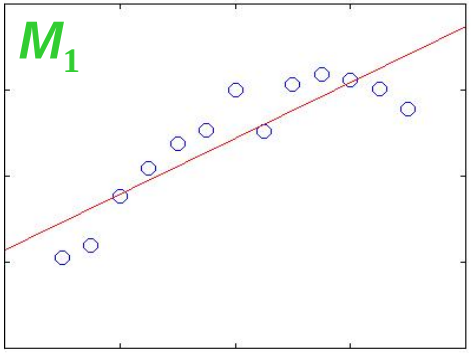


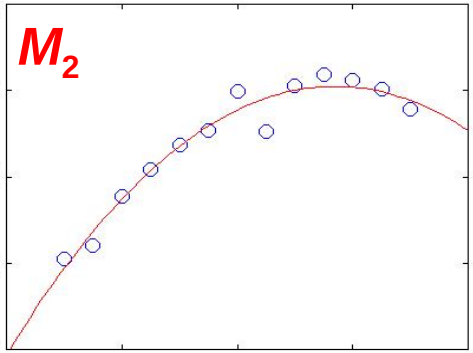
for any model M $$\sum_{\text{all } d \in D} p(D = d \mid M) = 1$$

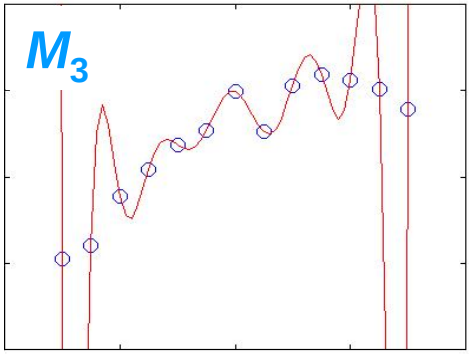
The law of ***conservation of belief*** states that models that explain many possible data sets must necessarily assign each of them a <u>low</u> probability.

Bayesian Occam's Razor









M_1 : the *too simple* model is unlikely to generate *this* data

M_3 : the *too complex* model is a little better but still unlikely

M_2 : the *just right* model has the highest marginal likelihood

All the Bayesics

$$P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)P(\theta)}{P(\mathcal{D})}$$

$P(\mathcal{D}\|\theta)$	likelihood of θ
$P(\theta)$	prior probability of θ
$P(\theta\|\mathcal{D})$	posterior of θ given $\mathcal{D}$

Model Comparison:

$$\begin{aligned} P(m|\mathcal{D}) &= \frac{P(\mathcal{D}|m)P(m)}{P(\mathcal{D})} \\ P(\mathcal{D}|m) &= \int P(\mathcal{D}|\theta, m)P(\theta|m)\, d\theta \end{aligned}$$

Prediction:

$$\begin{aligned} P(x|\mathcal{D}, m) &= \int P(x|\theta, \mathcal{D}, m)P(\theta|\mathcal{D}, m)d\theta \\ P(x|\mathcal{D}, m) &= \int P(x|\theta)P(\theta|\mathcal{D}, m)d\theta \qquad \text{(for many models)} \end{aligned}$$

by *Zoubin Ghahramani*

Approximation Methods

- for the evidence and posterior integrals
  - Laplace's method
  - Bayesian Information Criteria (BIC = MDL)
  - Akaike Information Criteria (AIC)
  - Variational Bayes (VB)
  - Expectation Propagation (EP)
  - Markov Chain Monte Carlo (MCMC)
  - Exact ("Perfect") Sampling
  - *etc*

Parametric *vs.* Nonparametric

- Parametric
  - the total # of parameters is **fixed** (property of its distribution)
  - so it doesn't depend (grow) with the # of data points collected
  - for prediction, knowing θ means you can throw away your data

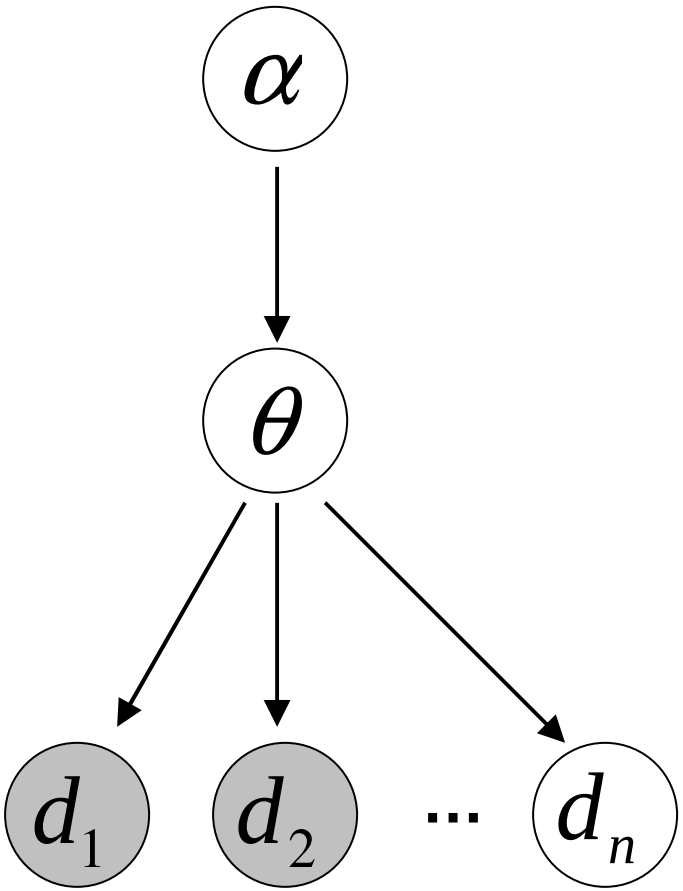
$$p(d_{new} \mid D) \propto p(d_{new} \mid \theta)\, p(\theta \mid D)$$

- Nonparametric
  - # of parameters can grow with the number of data
  - so the model can "adapt" to the data's complexity
  - but this typically means you can **not** throw away your data
  - future predictions require access to the previous training set

$$p(d_{new} \mid D, \alpha)$$

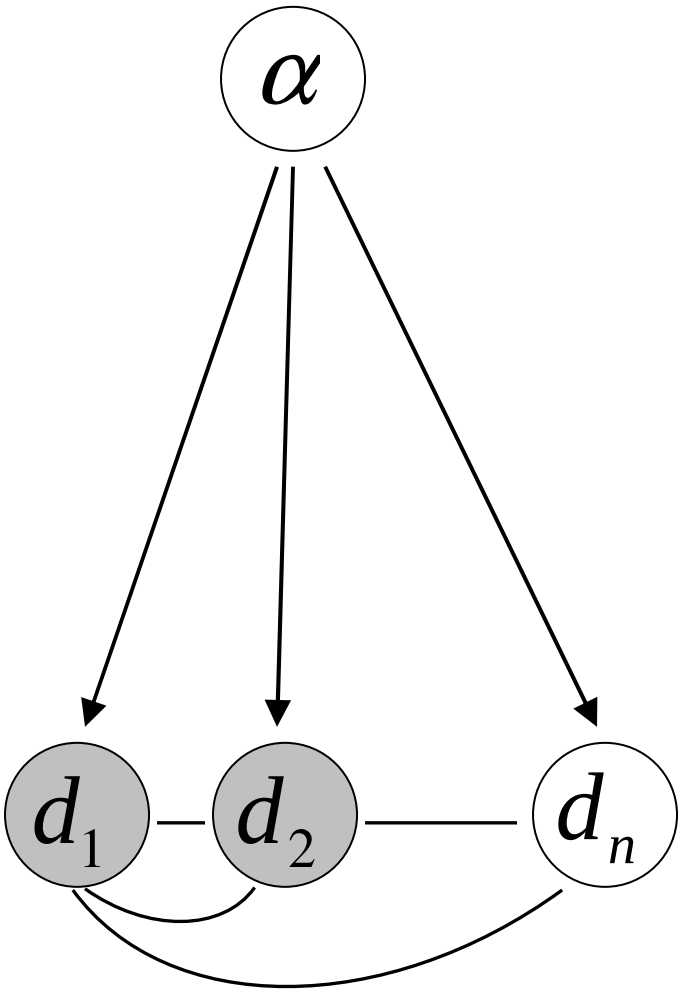
Parametric

$$p(d_n \mid D) \propto p(d_n \mid \theta)\, p(\theta \mid D)$$



Nonparametric

$$p(d_n \mid D, \alpha)$$



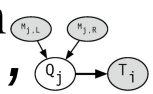
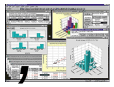
Nonparametric Bayesian Models

- Gaussian Process
- Dirichlet Process
  - Chinese restaurant process
  - Polya urn model
  - Stick-breaking models
  - Pitman-Yor process
- Indian buffet process
- Polya trees
- Dirichlet diffusion trees
- Infinite Hidden Markov Models

Gaussian Process Regression

- for modeling, prediction, curve fitting

$$y = f(\mathbf{x}) + \varepsilon$$

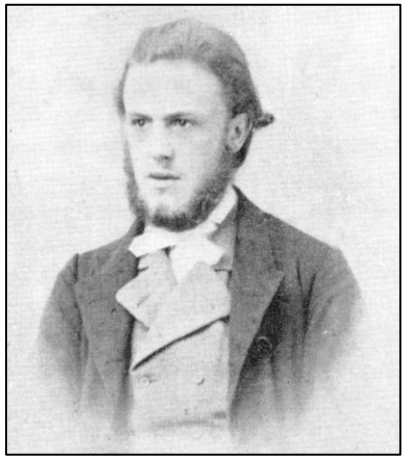
- some (mis)conceptions:
  - not curve fitting! we want $p(y \mid x)$
  - not just regression with Gaussian noise
  - input x can be : $\mathbb{R}, \mathbb{Z}, \mathbb{R}^n,$   , "ATGC"
  - output y (hence f) is a scalar $\mathbb{R}$

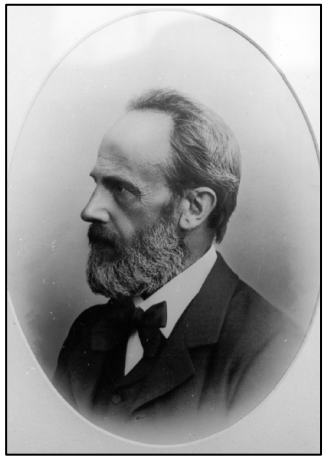
Early GP History

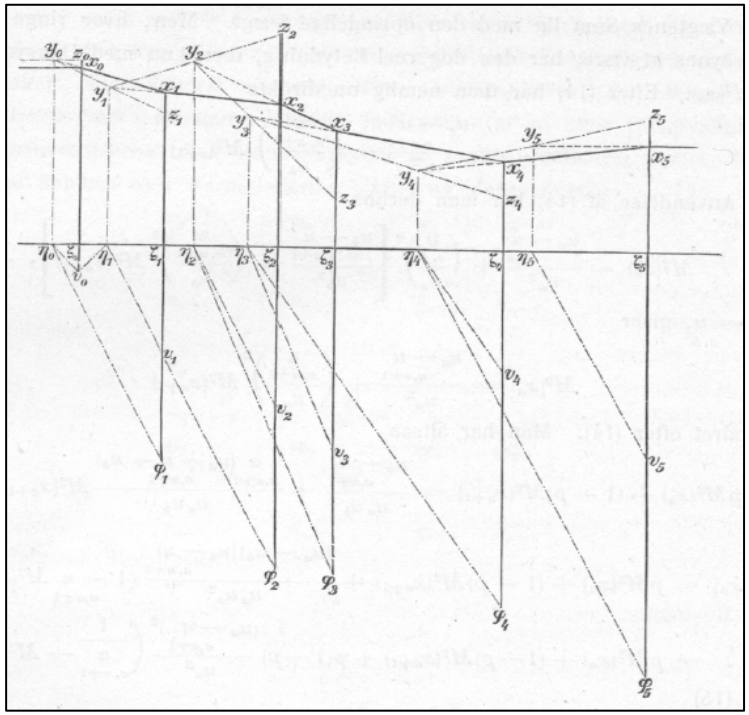
- Thiele (1880)
- Kolmogorov (1941), Wiener (1949)
- Thompson (1956) : meteorology
- Matheron (1963) : *Kriging* in Geostatistics
- Whittle (1963) : geospatial prediction
- **O'Hagan (1978) : general Bayesian regression**
- Ripley (1981): Bayesian spatial models

Thorvald Nicolai Thiele

- Danish Astronomer, Actuarian, Statistician
- Born 1838, died 1910
- “General Theory of Observations” (1889)
- Find “best predictor” for time series $x(t)$
- Formulated a “Kalman Filter” for a GP
  - rediscovered by Kalman & Bucy (1960)







Recent GP History

- Bar-Shalom & Fortman (1988) : Kalman filters
- Poggio & Girosi (1989) : Generalized RBFs
- Wahba (1990) : ARMA models & splines
- Cressie (1993) : spatial statistics (2D/3D)
- Neal (1996) : MLP = GP
- Williams & Rasmussen (1997) : general ML
- Saunders (1998) : KRR (Kriging rediscovered)
- *etc etc* (just see NIPS, UAI, AISTATS, ICML …)

Two Equivalent Views

GP models

Weight Space	Random Process
linear	nonlinear
parametric	nonparametric
finite-dim	infinite-dim

Weight Space View

Parametric and finite-dimensional

- Regression with basis functions ϕ $\quad f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}$

  - *e.g.*, cubic polynomials $\quad \phi(x) = [1 \quad x \quad x^2 \quad x^3]^T$

$$w = [w_0 \quad w_1 \quad w_2 \quad w_3]^T$$

- Gaussian prior on weights $\quad \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_p)$

- let $K = \langle \mathrm{f}\,\mathrm{f}^{\mathrm{T}} \rangle = \mathrm{Cov}(\mathrm{f})$ $\quad K = \Phi^\top \Sigma_p \Phi$

- vector f is jointly Gaussian $\quad \mathbf{f}|X \sim \mathcal{N}(\mathbf{0}, K)$

Random Process View

non-parametric and infinite-dimensional

imagine increasing the length of that f vector to infinity

$$K = \Phi^\top \Sigma_p \Phi \longrightarrow K_{ij} = k(x_i, x_j)$$

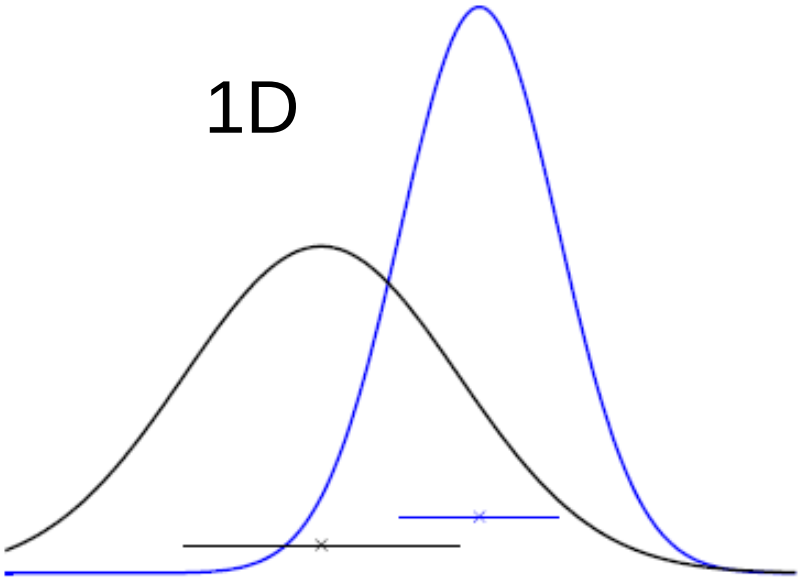
Informally speaking

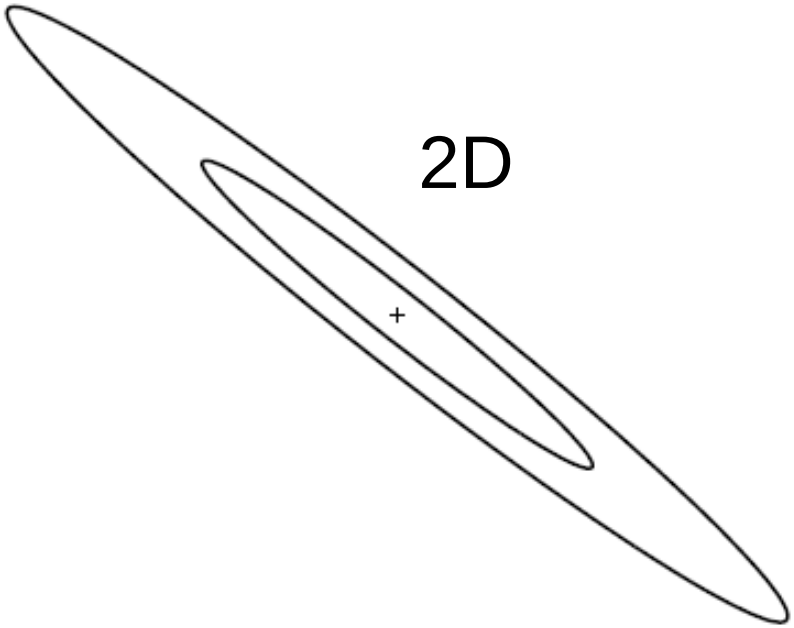
- the infinite-dimensional vector f becomes a *function* $\mathrm{f}(x)$
- its covariance matrix K becomes a *kernel function* $k(x_i,x_j)$
- in the limit $\mathrm{f}(x)$ becomes a stochastic *Gaussian Process*

$$\boxed{f(\mathbf{x}) \sim \mathcal{GP}\big(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\big)}$$

- mean function $m(\mathrm{x})$
- kernel function $k(\mathrm{x}, \mathrm{x}')$

Our Beloved Gaussian





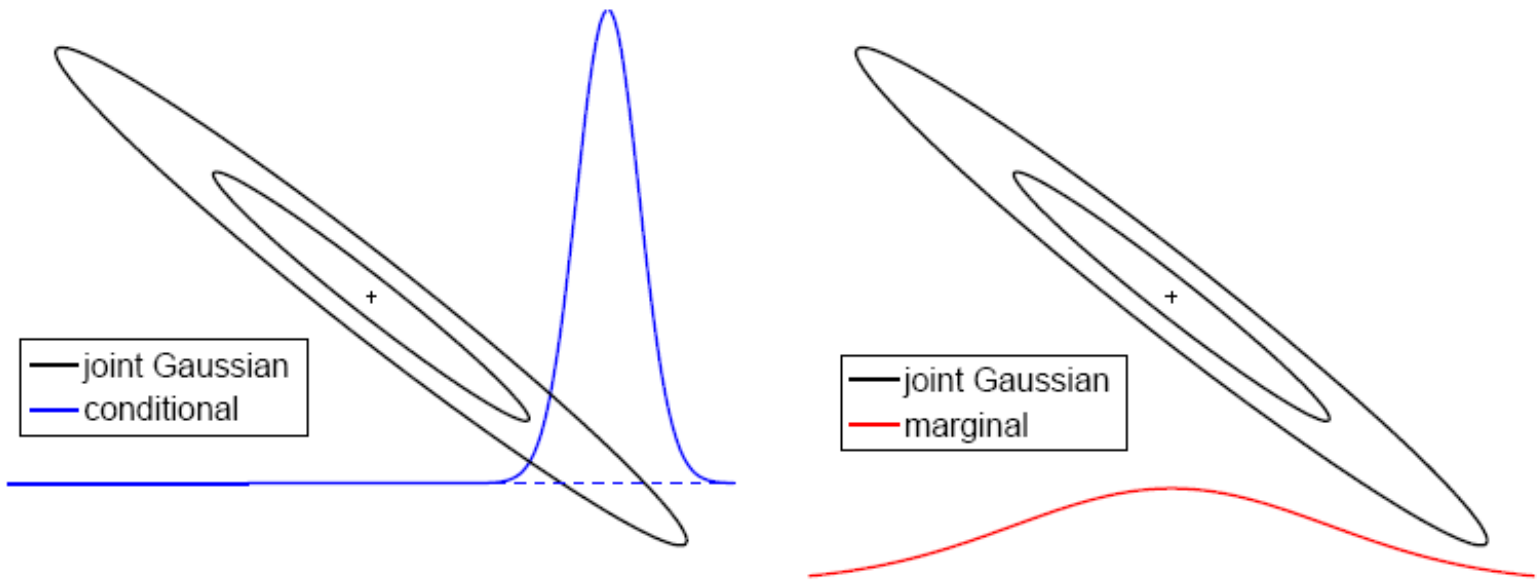
The Gaussian distribution is given by

$$p(\mathbf{x}|\mu, \Sigma) = \mathcal{N}(\mu, \Sigma) = (2\pi)^{-D/2}|\Sigma|^{-1/2} \exp\left(-\tfrac{1}{2}(\mathbf{x}-\mu)^\top \Sigma^{-1}(\mathbf{x}-\mu)\right)$$

where μ is the mean vector and Σ the covariance matrix.

by *Carl Rasmussen*

Gaussians beget Gaussians



Both the conditionals and the marginals of a joint Gaussian are again Gaussian.

Also : the product of Gaussians is Gaussian

e.g., Gaussian **prior** x Gaussian **likelihood** à Gaussian **posterior**

by *Carl Rasmussen*

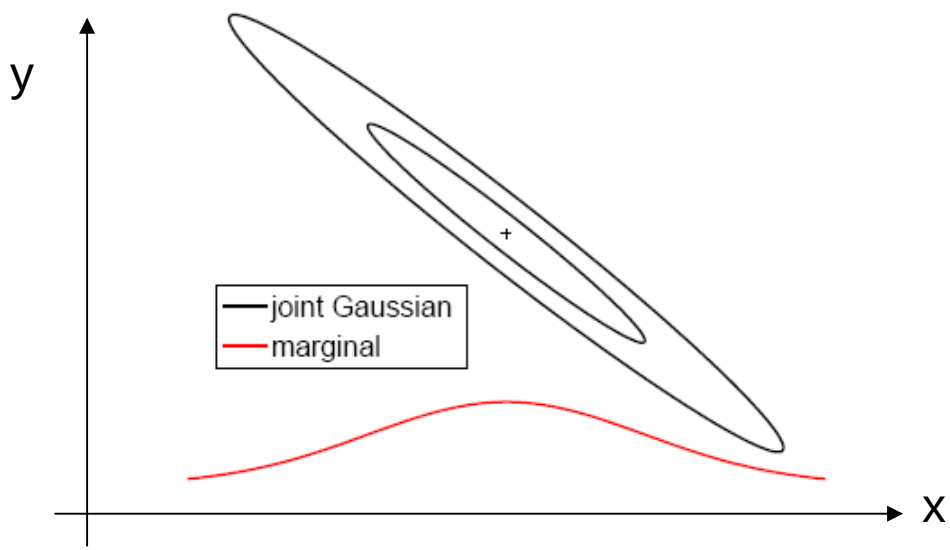
Marginalization

Recall:

$$p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{y}) d\mathbf{y}.$$

For Gaussians:

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}\right) \implies p(\mathbf{x}) = \mathcal{N}(\mathbf{a}, A)$$



$$\mathrm{a} = \mathrm{mean}(\mathrm{x})$$

$$\mathrm{b} = \mathrm{mean}(\mathrm{y})$$

$$A = \mathrm{cov}(\mathrm{x}, \mathrm{x}')$$

$$B = \mathrm{cov}(\mathrm{x}, \mathrm{y})$$

$$C = \mathrm{cov}(\mathrm{y}, \mathrm{y}')$$

by *Carl Rasmussen*

Conditional Gaussians

- Jointly Gaussian (sub)vectors $\mathbf{x}$ and $\mathbf{y}$

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}\right)$$

- Conditional density of $\mathbf{x}$ given $\mathbf{y}$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{a}+BC^{-1}(\mathbf{y}-\mathbf{b}), \boxed{A-BC^{-1}B^\top})$$

Schur complement of *C*

Definition of Gaussian Process

A *Gaussian process* is a generalization of a multivariate Gaussian distribution to infinitely many variables.

Informally: infinitely long vector $\simeq$ function

> **Definition**: *a Gaussian process is a collection of random variables, any finite number of which have (consistent) Gaussian distributions.* $\square$

A Gaussian distribution is fully specified by a mean vector, μ, and covariance matrix Σ:

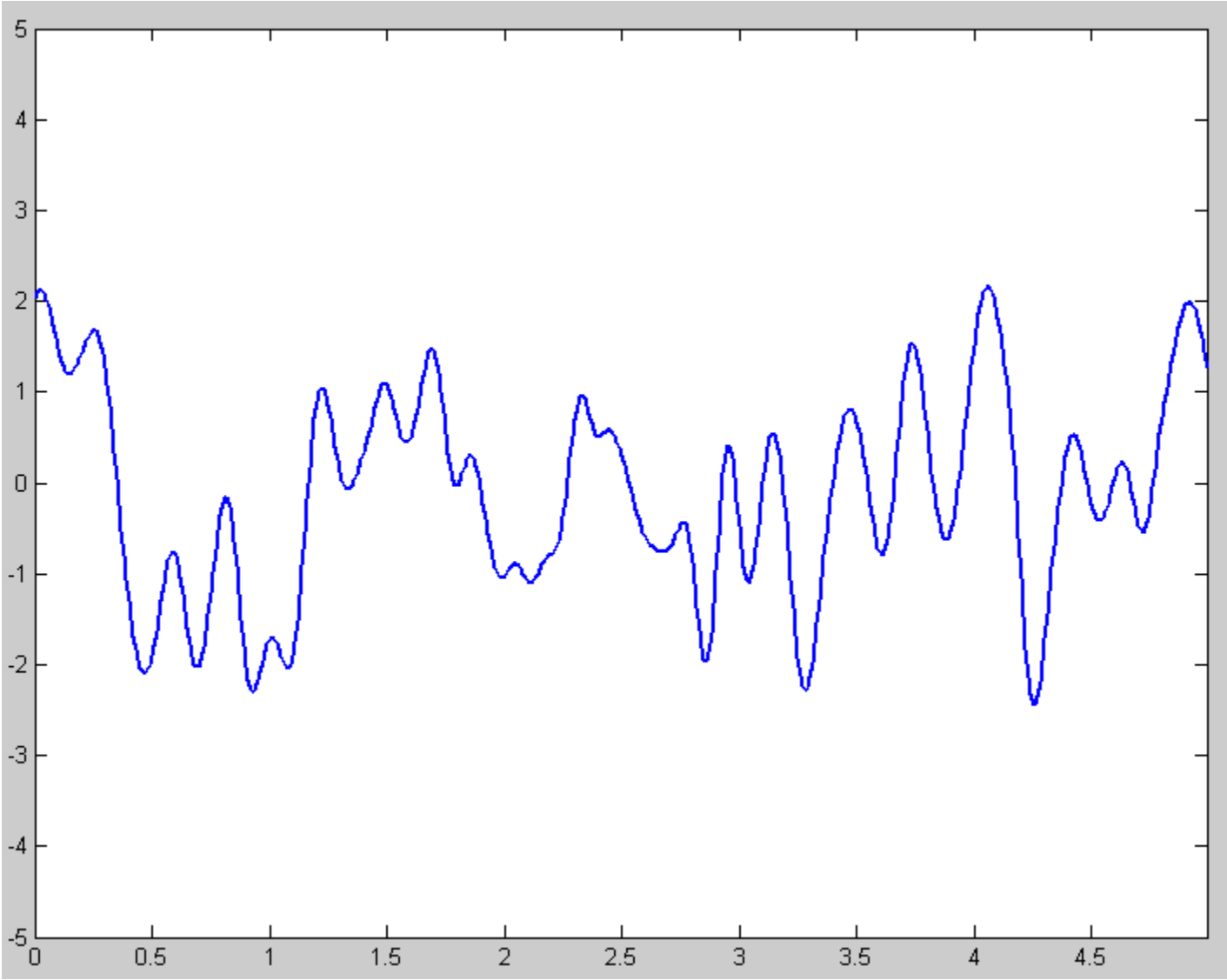
$$\mathbf{f} = (f_1, \ldots, f_n)^\top \sim \mathcal{N}(\mu, \Sigma), \quad \text{indexes } i = 1, \ldots, n$$

A Gaussian process is fully specified by a mean function $m(x)$ and covariance function $k(x, x')$:

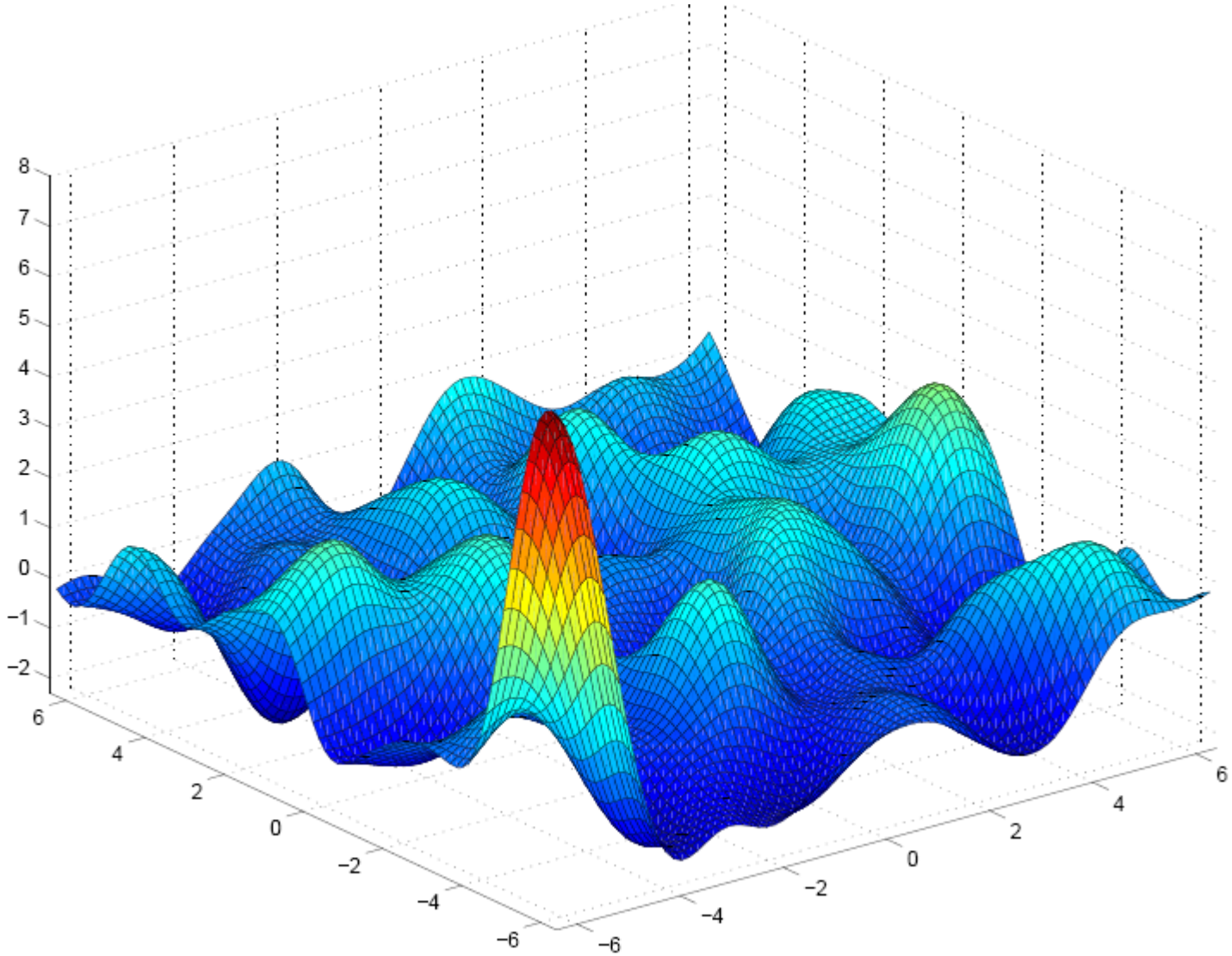
$$f(x) \sim GP(m(x), k(x, x'))$$

by *Carl Rasmussen*

1D GP sample (RBF)



2D GP sample (RBF)



Covariances (Kernels)

Where does the covariance matrix $\mathbf{K}$ come from?

- Covariance matrix constructed from *covariance function*:

$$\mathbf{K}_{ij} = K(x_i, x_j)$$

- Covariance function characterizes correlations between different points in the process:

$$\boxed{K(x, x') = \mathcal{E}[f(x)f(x')]}$$

not a function of ***f***

- Must produce positive semidefinite covariance matrices $\mathbf{v}^\top \mathbf{K} \mathbf{v} \geq 0$

- Ensures consistency

by *Carl Rasmussen*

Squared Exponential (RBF) Kernel

$$K(x, x') = {\sigma_0}^2 \exp\left[-\frac{1}{2}\left(\frac{x - x'}{\lambda}\right)^2\right]$$

- Intuition: function variables close in input space are highly correlated, whilst those far away are uncorrelated
- λ, σ_0 — hyperparameters. λ: lengthscale, σ_0: amplitude
- Stationary: $K(x, x') = K(x - x')$ — invariant to translations
- Very smooth sample functions — infinitely differentiable

by *Carl Rasmussen*

Nonstationary Covariances

- Linear covariance: $K(x, x') = \sigma_0^2 + xx'$
- Brownian motion (Wiener process): $K(x, x') = \min(x, x')$
- Periodic covariance: $K(x, x') = \exp\left(-\frac{2\sin^2\left(\frac{x-x'}{2}\right)}{\lambda^2}\right)$
- Neural network covariance

$$k(x, x') = \tanh(a\, x \cdot x' + b)$$ not a valid kernel (psd)!

$$k_{\mathrm{NN}}(\mathbf{x}, \mathbf{x}') = \frac{2}{\pi}\sin^{-1}\left(\frac{2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}}'}{\sqrt{(1 + 2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}})(1 + 2\tilde{\mathbf{x}}'^\top \Sigma \tilde{\mathbf{x}}')}}\right)$$

by *Carl Rasmussen*

Matern Class of Covariances

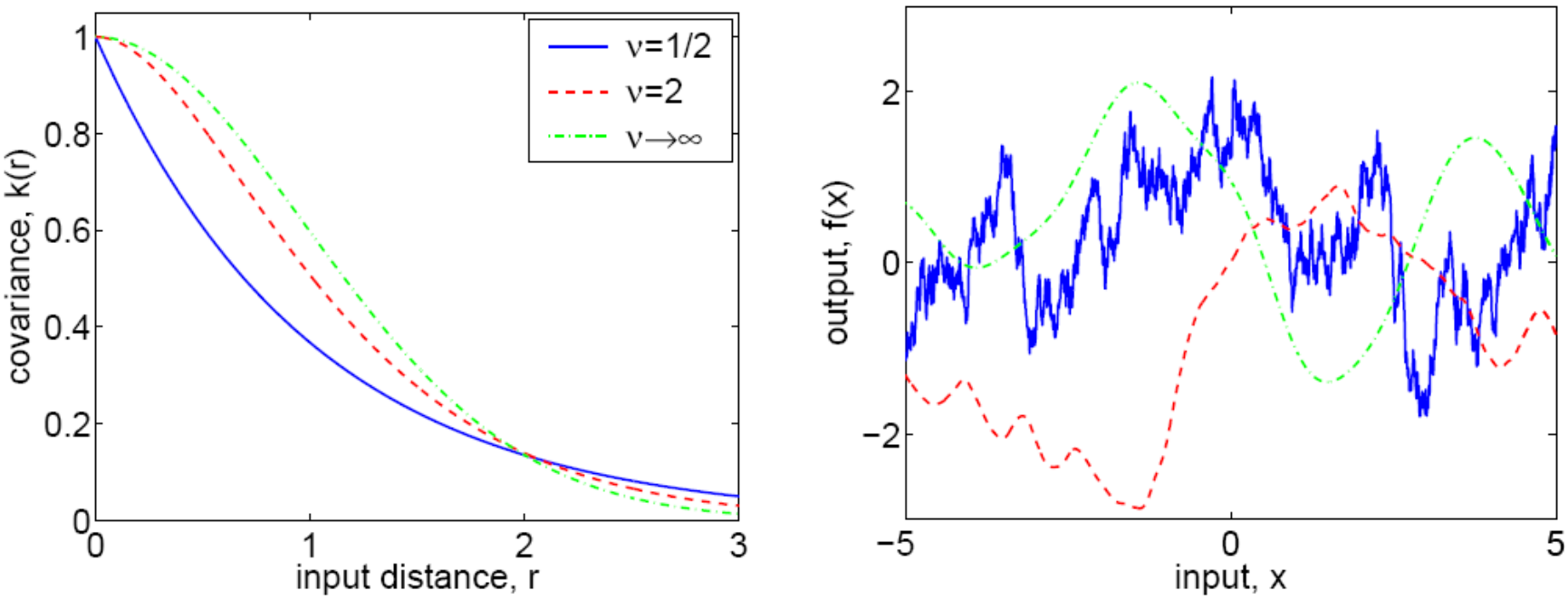
$$K(x, x') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}|x - x'|}{\lambda} \right)^{\nu} K_{\nu} \left(\frac{\sqrt{2\nu}|x - x'|}{\lambda} \right)$$

where K_ν is a modified Bessel function.

- Stationary, isotropic
- $\nu \to \infty$: SE covariance
- Finite ν: much rougher sample functions
- $\nu = 1/2$: $K(x, x') = \exp(-|x - x'|/\lambda)$, OU process, very rough sample functions

by *Carl Rasmussen*

Matern Class of Covariances



(a) (b)

Figure 4.1: Panel (a): covariance functions, and (b): random functions drawn from Gaussian processes with Matérn covariance functions, eq. (4.14), for different values of ν, with $\ell = 1$. The sample functions on the right were obtained using a discretization of the x-axis of 2000 equally-spaced points.

by *Carl Rasmussen*

Rational Quadratic Kernels

The *rational quadratic* (RQ) covariance function:

$$k_{\mathrm{RQ}}(r) \;=\; \left(1 + \frac{r^2}{2\alpha\ell^2}\right)^{-\alpha}$$

with $\alpha,\ \ell > 0$ can be seen as a *scale mixture* (an infinite sum) of squared exponential (SE) covariance functions with different characteristic length-scales.

Using $\tau = \ell^{-2}$ and $p(\tau|\alpha, \beta) \propto \tau^{\alpha-1}\exp(-\alpha\tau/\beta)$:

$$\begin{aligned} k_{\mathrm{RQ}}(r) \;&=\; \int p(\tau|\alpha, \beta) k_{\mathrm{SE}}(r|\tau) d\tau \\ &\propto \int \tau^{\alpha-1} \exp\left(-\frac{\alpha\tau}{\beta}\right) \exp\left(-\frac{\tau r^2}{2}\right) d\tau \;\propto\; \left(1 + \frac{r^2}{2\alpha\ell^2}\right)^{-\alpha}, \end{aligned}$$

by *Carl Rasmussen*

Rational Quadratic Kernels

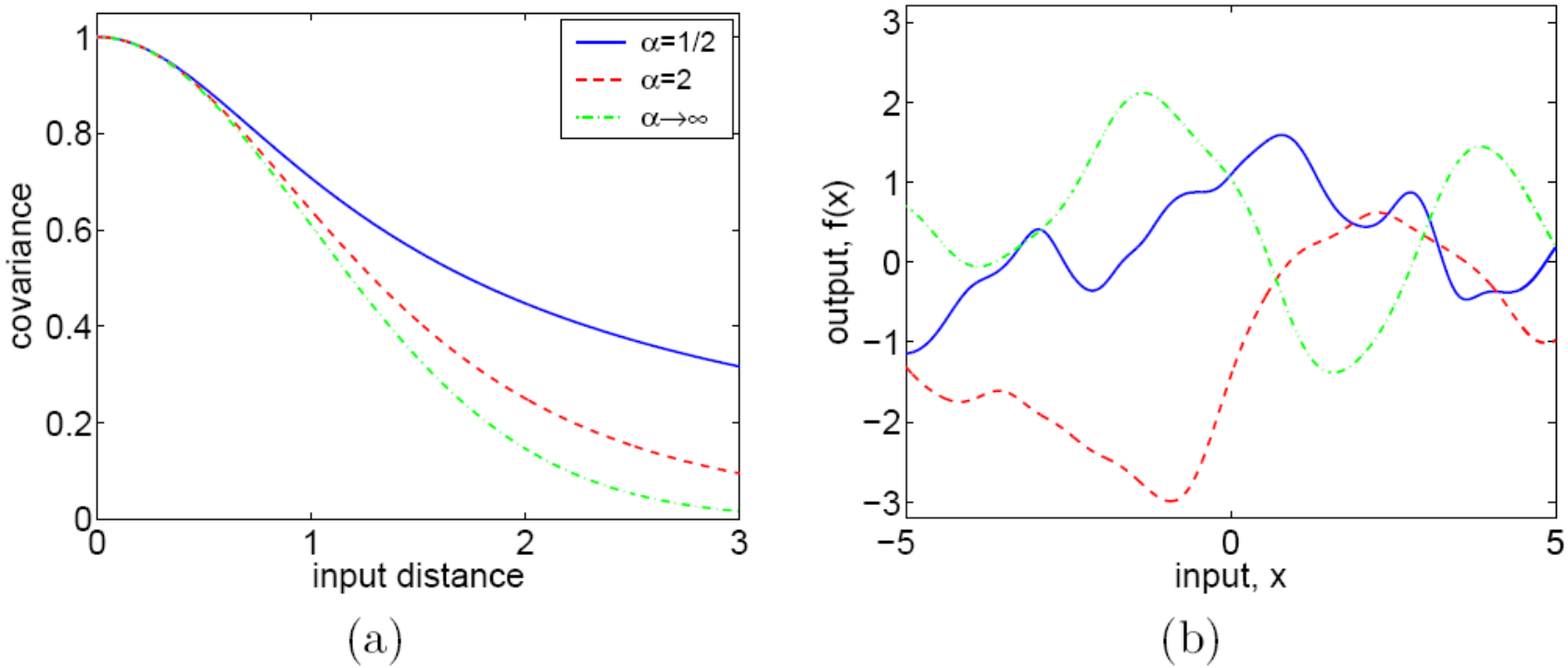


Figure 4.3: Panel (a) covariance functions, and (b) random functions drawn from Gaussian processes with rational quadratic covariance functions, eq. (4.20), for different values of α with $\ell = 1$. The sample functions on the right were obtained using a discretization of the x-axis of 2000 equally-spaced points.

by *Carl Rasmussen*

Building New Covariances

There are several ways to combine covariances:

- Sum: $K(x, x') = K_1(x, x') + K_2(x, x')$
  addition of independent processes

- Product: $K(x, x') = K_1(x, x') K_2(x, x')$
  product of independent processes

- Convolution: $K(x, x') = \int dz\, dz'\, h(x, z) K(z, z') h(x', z')$
  blurring of process with kernel h

by *Carl Rasmussen*

Matlab Demo 1

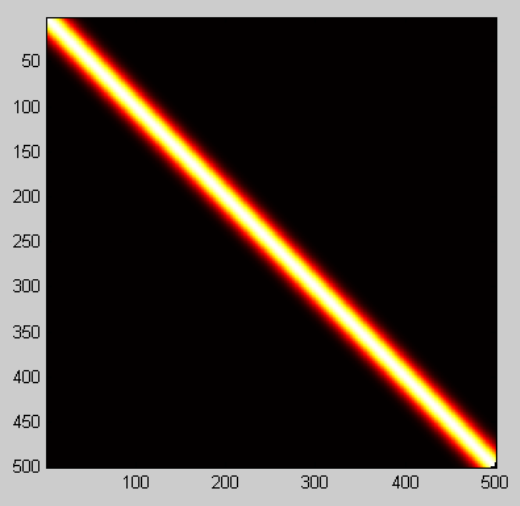
Sampling from a GP

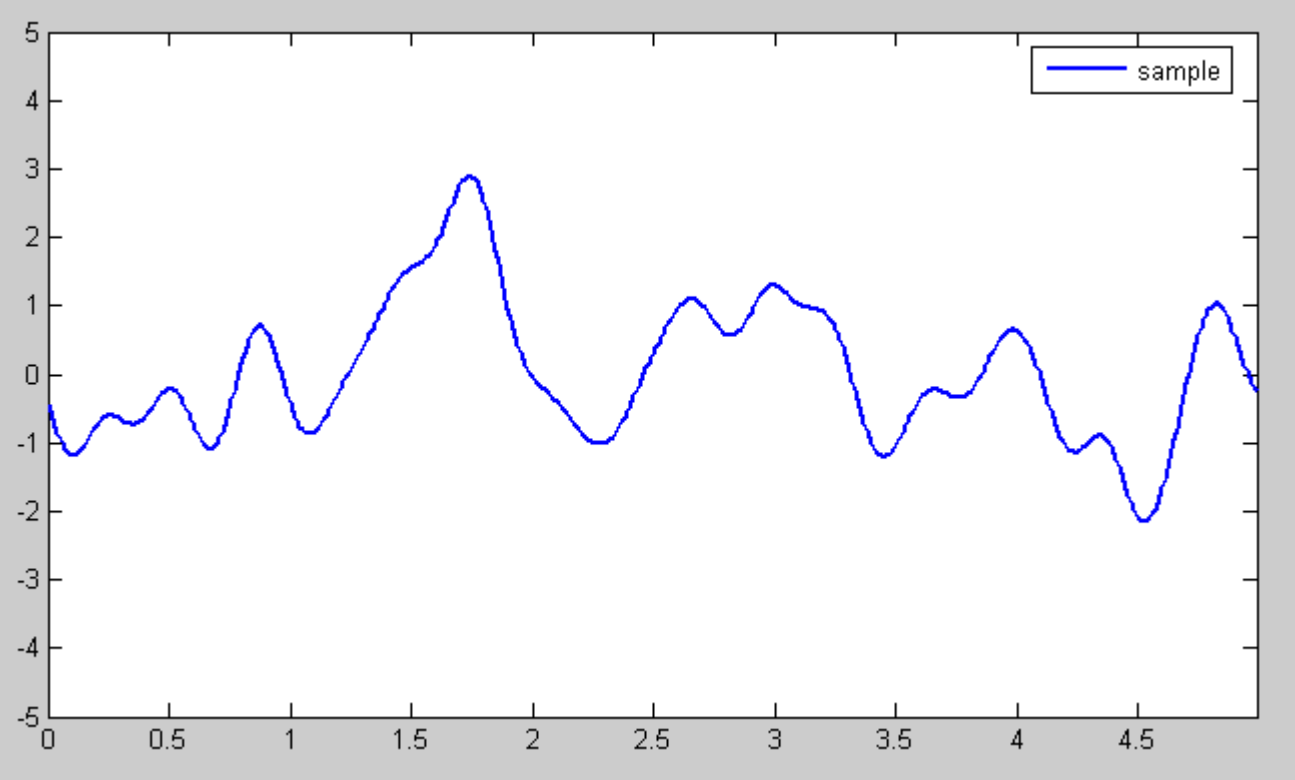
Matlab:

```
% given GP(fmean,K)

L = chol(K)';   % K = L*L';

while 1
    f = L*randn(n,1) + fmean;
    plot(f)
    pause
end
```

Demo 1: Sampling from Prior

Prior Cov





Conditional Gaussians

- Jointly Gaussian (sub)vectors x and y

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}\right)$$

- Conditional density of x given y

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\underbrace{\mathbf{a}+BC^{-1}(\mathbf{y}-\mathbf{b})}_{\text{mean}}, \underbrace{A-BC^{-1}B^\top}_{\text{covariance}})$$

GPR with Gaussian Noise

Data generated with Gaussian white noise around the function f

$$y = f + \epsilon \qquad \mathcal{E}[\epsilon(x)\epsilon(x')] = \sigma^2\delta(x - x')$$

Equivalently, the noise model, or *likelihood* is:

$$p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{f}, \sigma^2\mathbf{I})$$

Integrating over the function variables gives the *marginal likelihood*:

$$\begin{aligned} p(\mathbf{y}) &= \int \mathrm{d}\mathbf{f}\, p(\mathbf{y}|\mathbf{f})p(\mathbf{f}) \\ &= \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma^2\mathbf{I}) \end{aligned}$$

by *Carl Rasmussen*

GPR Prediction

N training input and output pairs $(\mathbf{X}, \mathbf{y})$, and T test inputs $\mathbf{X}_T$

Consider joint training and test marginal likelihood:

$$p(\mathbf{y}, \mathbf{y}_T) = \mathcal{N}(\mathbf{0}, \mathbf{K}_{N+T} + \sigma^2\mathbf{I}) \ , \quad \mathbf{K}_{N+T} = \begin{bmatrix} \mathbf{K}_N & \mathbf{K}_{NT} \\ \mathbf{K}_{TN} & \mathbf{K}_T \end{bmatrix} \ ,$$

Condition on training outputs: $p(\mathbf{y}_T|\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_T)$

$$\boldsymbol{\mu}_T = \mathbf{K}_{TN}[\mathbf{K}_N + \sigma^2\mathbf{I}]^{-1}\mathbf{y}$$
$$\boldsymbol{\Sigma}_T = \mathbf{K}_T - \mathbf{K}_{TN}[\mathbf{K}_N + \sigma^2\mathbf{I}]^{-1}\mathbf{K}_{NT} + \sigma^2\mathbf{I}$$

Gives correlated predictions. Defines a predictive Gaussian *process*

by *Carl Rasmussen*

GPR Prediction

- Often only marginal variances ($\operatorname{diag} \boldsymbol{\Sigma}_T$) are required — sufficient to consider a single test input $\mathbf{x}_*$:

$$\mu_* = \mathbf{K}_{*N}\boxed{[\mathbf{K}_N + \sigma^2\mathbf{I}]^{-1}\mathbf{y}}$$
$$\sigma_*^2 = K_* - \mathbf{K}_{*N}[\mathbf{K}_N + \sigma^2\mathbf{I}]^{-1}\mathbf{K}_{N*} + \sigma^2 \,.$$

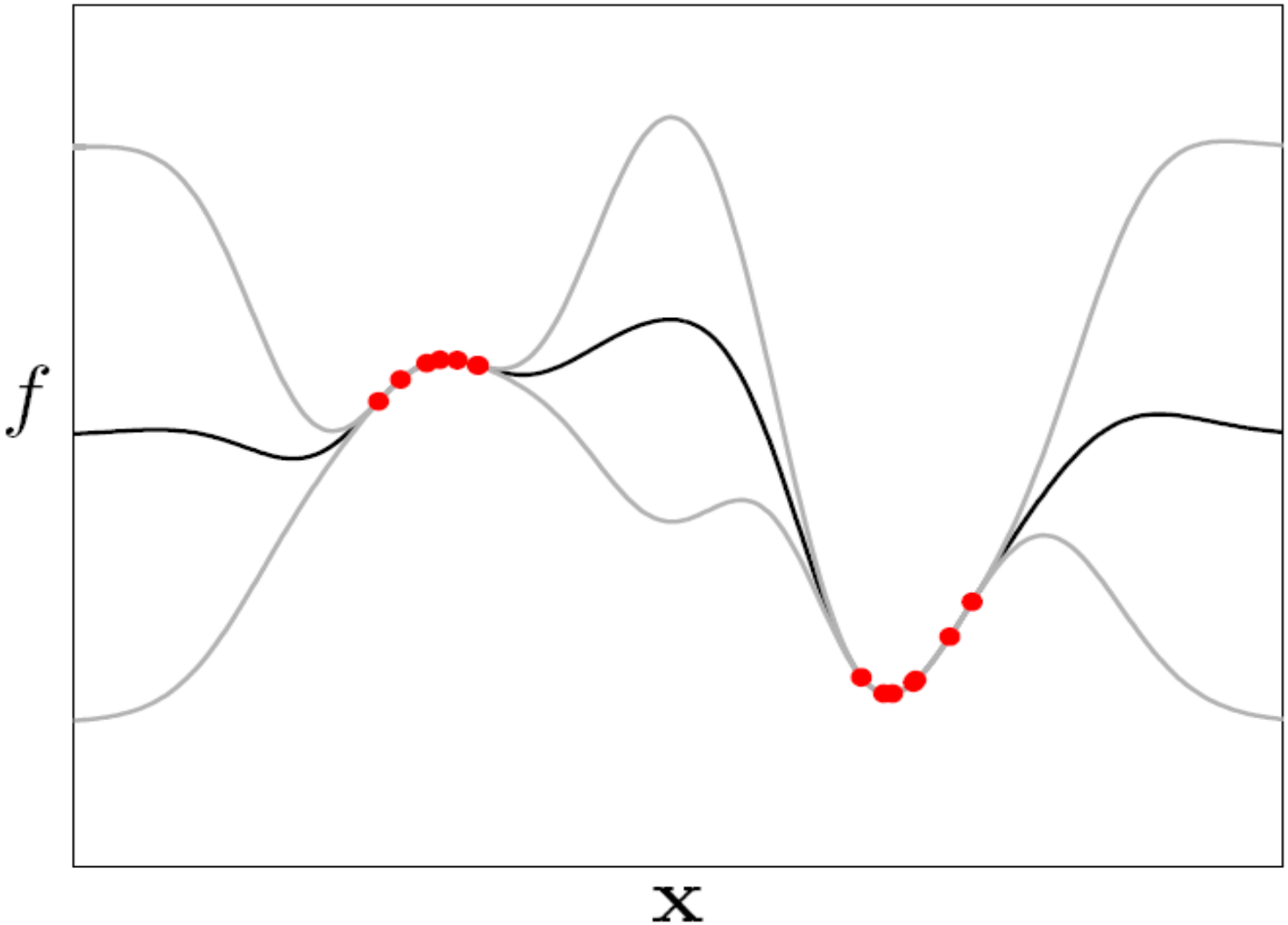
- Mean predictor is a linear predictor: $\mu_* = \mathbf{K}_{*N}\boxed{\boldsymbol{\alpha}}$ "parameter"

- Inversion of $\mathbf{K}_N + \sigma^2\mathbf{I}$ costs $\mathcal{O}(N^3)$

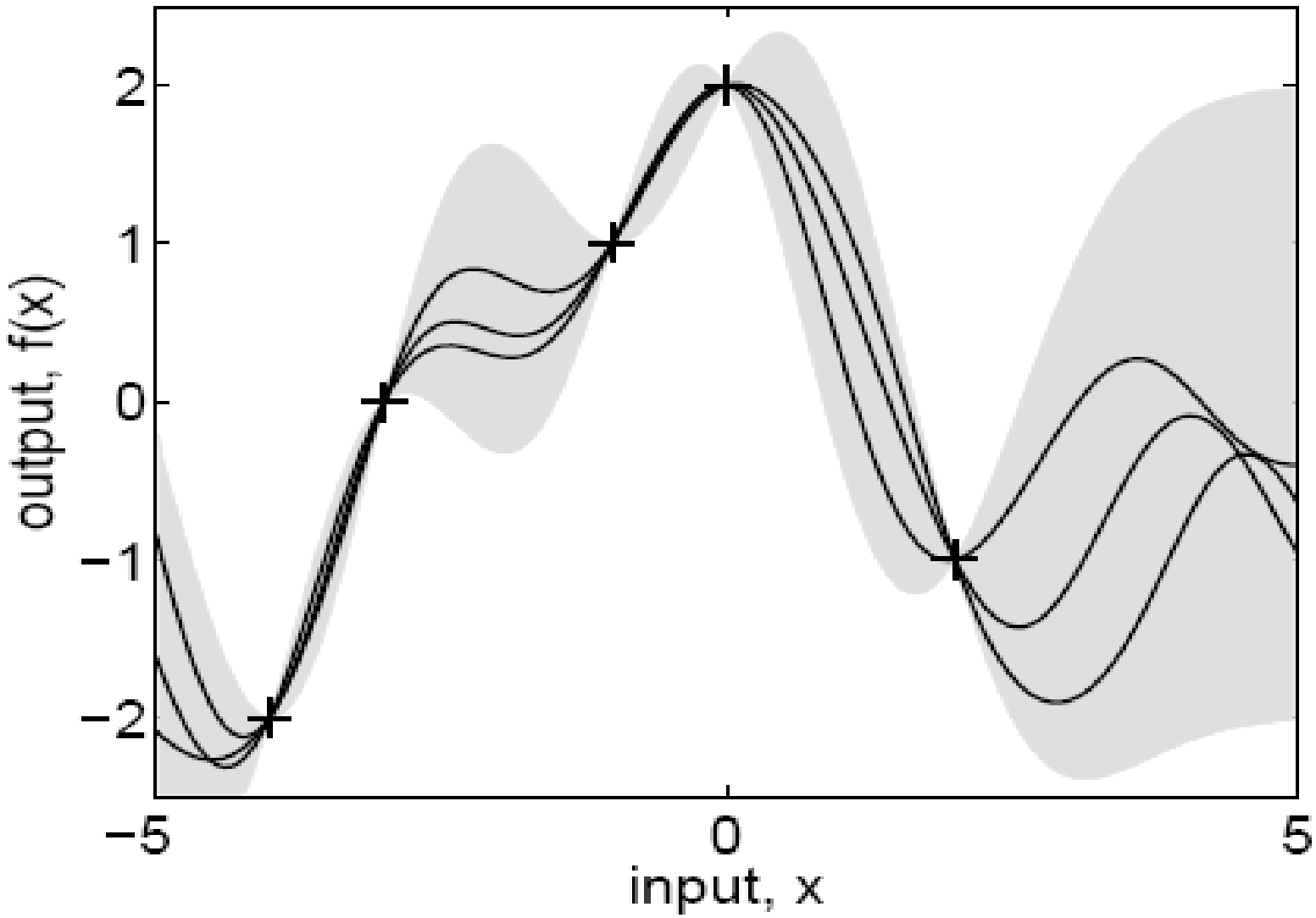
- Prediction cost per test case is $\mathcal{O}(N)$ for the mean and $\mathcal{O}(N^2)$ for the variance

by *Carl Rasmussen*

posterior mean +/- 2σ



posterior mean +/- 2σ



Matlab Demo 2

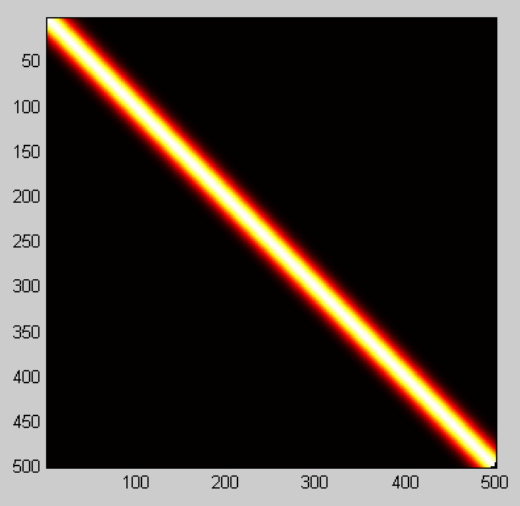
Posterior Sampling for GPR

```
% train set: (x,y)
L = chol(K)';
a = L'\(L\y);   % a = inv(K)*y;
ML = -y'*a/2 – sum(log(diag(L))) + const;
```

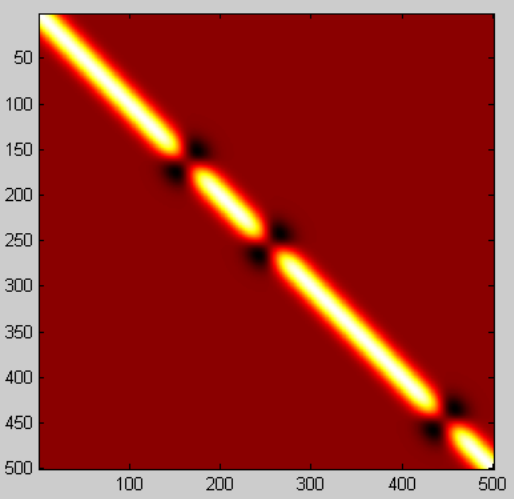
```
% test set: xt  (no noise)
ftmean = Kt*a;
V = L\Kt;
Kt = K – V'*V;  % Kt = K – Kt'inv(K)*Kt
Lt = chol(Kt)';
while 1
    ft = Lt*randn(m,1) + ftmean;
    plot(ft)
end
```

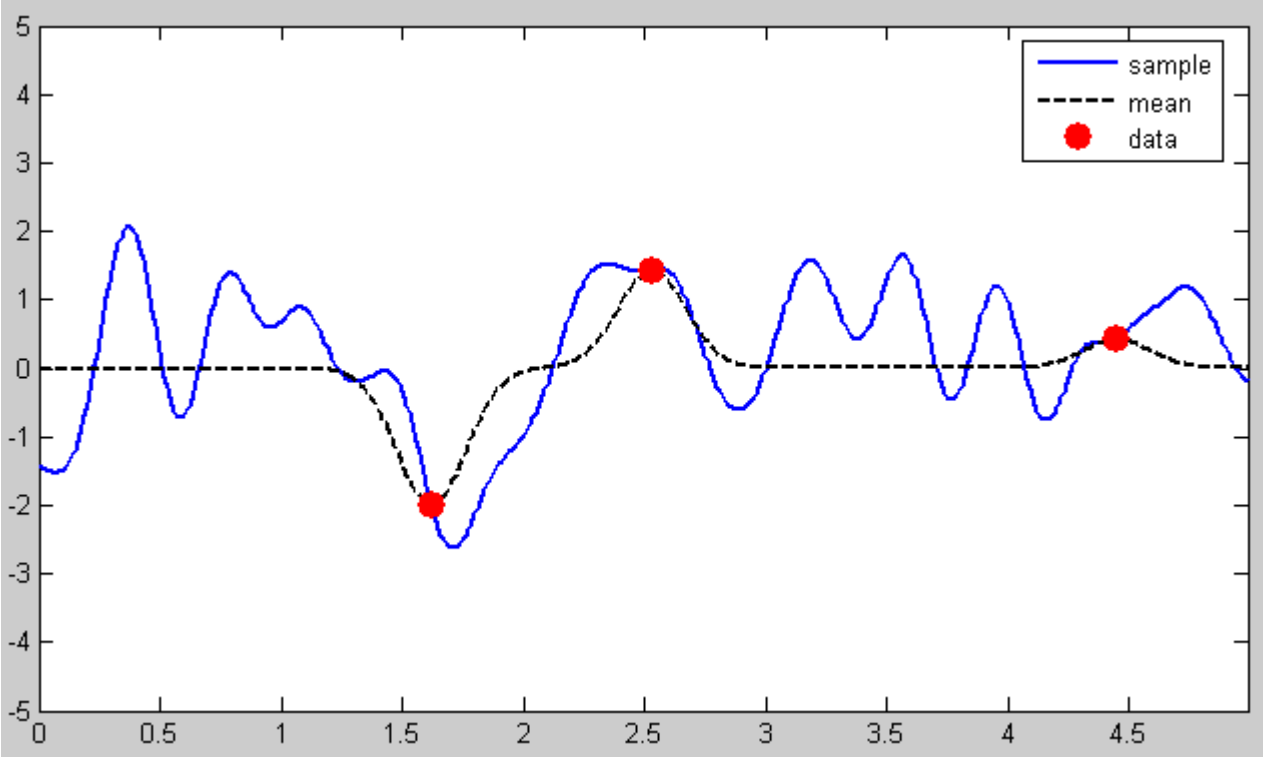
Demo 2: Sampling from Posterior

Prior Cov



Posterior Cov





All the Bayesics

$$P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)P(\theta)}{P(\mathcal{D})}$$

$P(\mathcal{D}\|\theta)$	likelihood of θ
$P(\theta)$	prior probability of θ
$P(\theta\|\mathcal{D})$	posterior of θ given $\mathcal{D}$

Model Comparison:

$$P(m|\mathcal{D}) = \frac{P(\mathcal{D}|m)P(m)}{P(\mathcal{D})}$$

$$\boxed{P(\mathcal{D}|m) = \int P(\mathcal{D}|\theta, m)P(\theta|m)\, d\theta}$$

Prediction:

$$P(x|\mathcal{D}, m) = \int P(x|\theta, \mathcal{D}, m)P(\theta|\mathcal{D}, m)d\theta$$

$$P(x|\mathcal{D}, m) = \int P(x|\theta)P(\theta|\mathcal{D}, m)d\theta \qquad \text{(for many models)}$$

by *Zoubin Ghahramani*

GP Marginal Likelihood

Log marginal likelihood:

$$\log p(\mathbf{y}|\mathbf{x}, M_i) \;=\; -\frac{1}{2}\mathbf{y}^\top K^{-1}\mathbf{y} - \frac{1}{2}\log|K| - \frac{n}{2}\log(2\pi)$$

is the combination of a data fit term and complexity penalty. Occam's Razor is automatic.

Learning in Gaussian process models involves finding

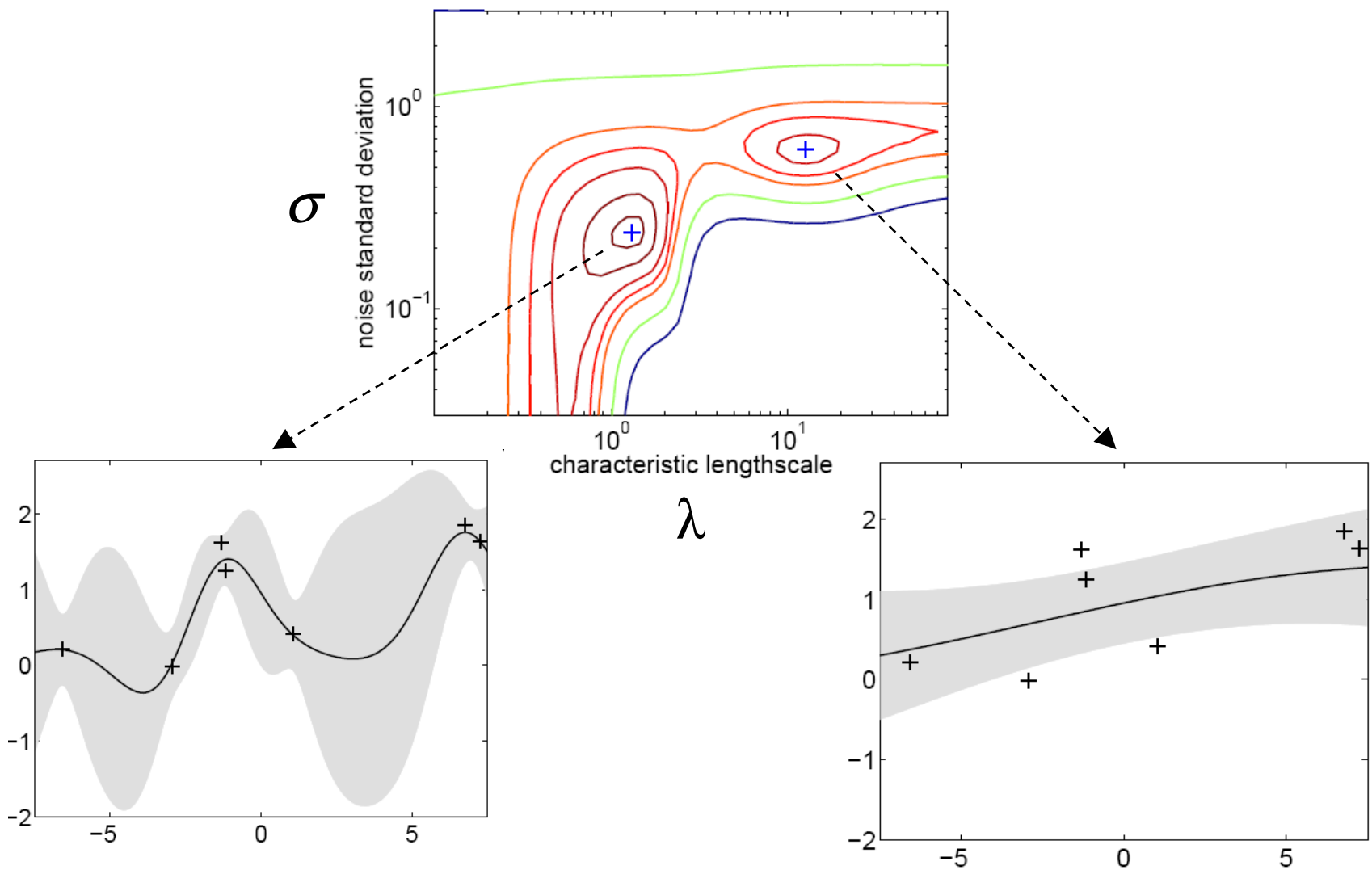
- the form of the covariance function, and
- any unknown (hyper-) parameters θ.

This can be done by optimizing the marginal likelihood:

$$\frac{\partial \log p(\mathbf{y}|\mathbf{x}, \theta, M_i)}{\partial \theta_j} \;=\; \frac{1}{2}\mathbf{y}^\top K^{-1}\frac{\partial K}{\partial \theta_j}K^{-1}\mathbf{y} - \frac{1}{2}\,\mathrm{trace}(K^{-1}\frac{\partial K}{\partial \theta_j})$$

by *Carl Rasmussen*

ML for RBF : $\exp(-(x-x')^2/\lambda^2) + \sigma^2\delta_{xx'}$



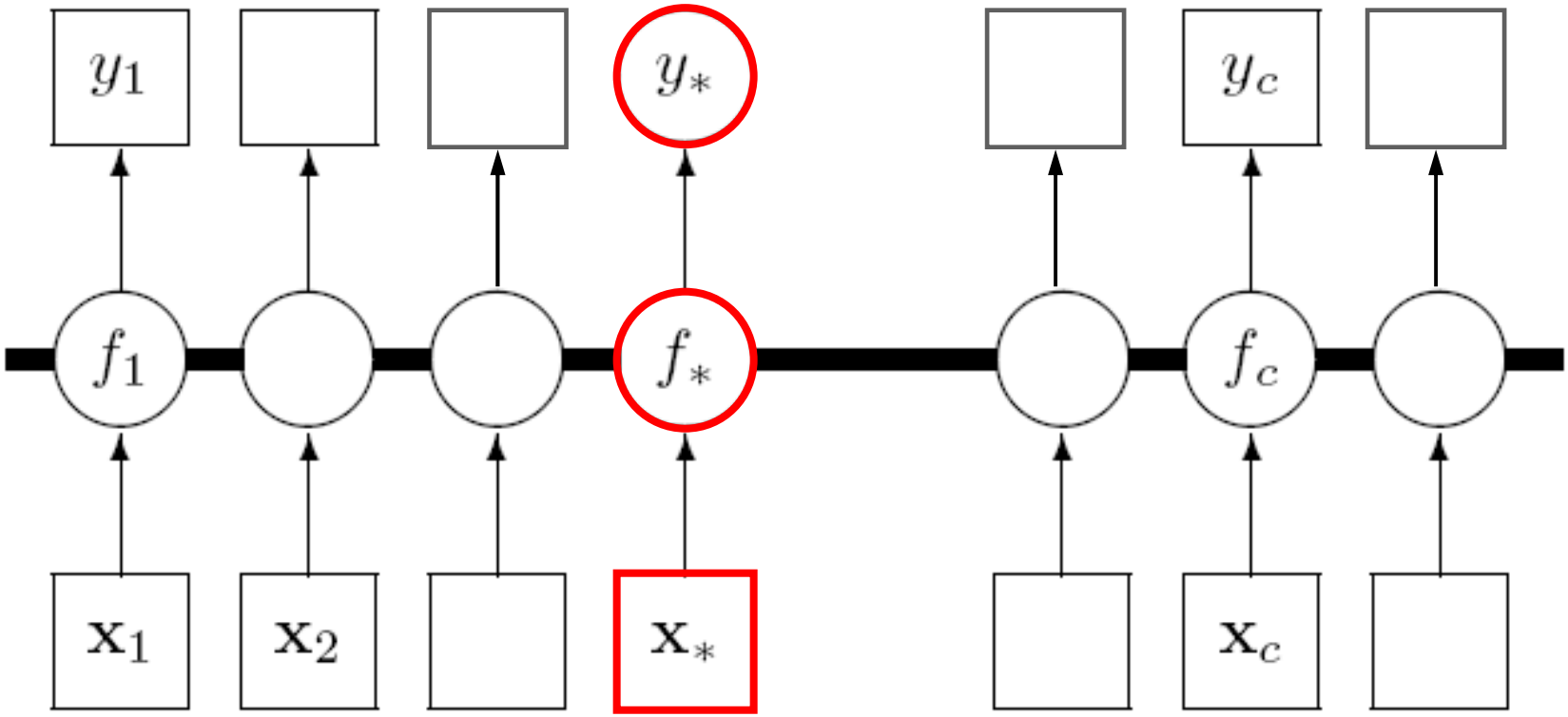
by *Carl Rasmussen*

Sparse GPs

- Problem for large data sets: training GP $\mathcal{O}(N^3)$, prediction $\mathcal{O}(N^2)$ per test case
- Recent years — many approximations developed — reduce cost to $\mathcal{O}(NM^2)$ training and $\mathcal{O}(M^2)$ prediction per test case
- Based around a low rank (M) covariance approximation
- See Quiñonero Candela and Rasmussen [2005] for a review of regression approximations
- Classification more complicated, so simpler approximations such as IVM[5] may be more suitable

[5] Lawrence et al., 2003

Graphical Model of GPs

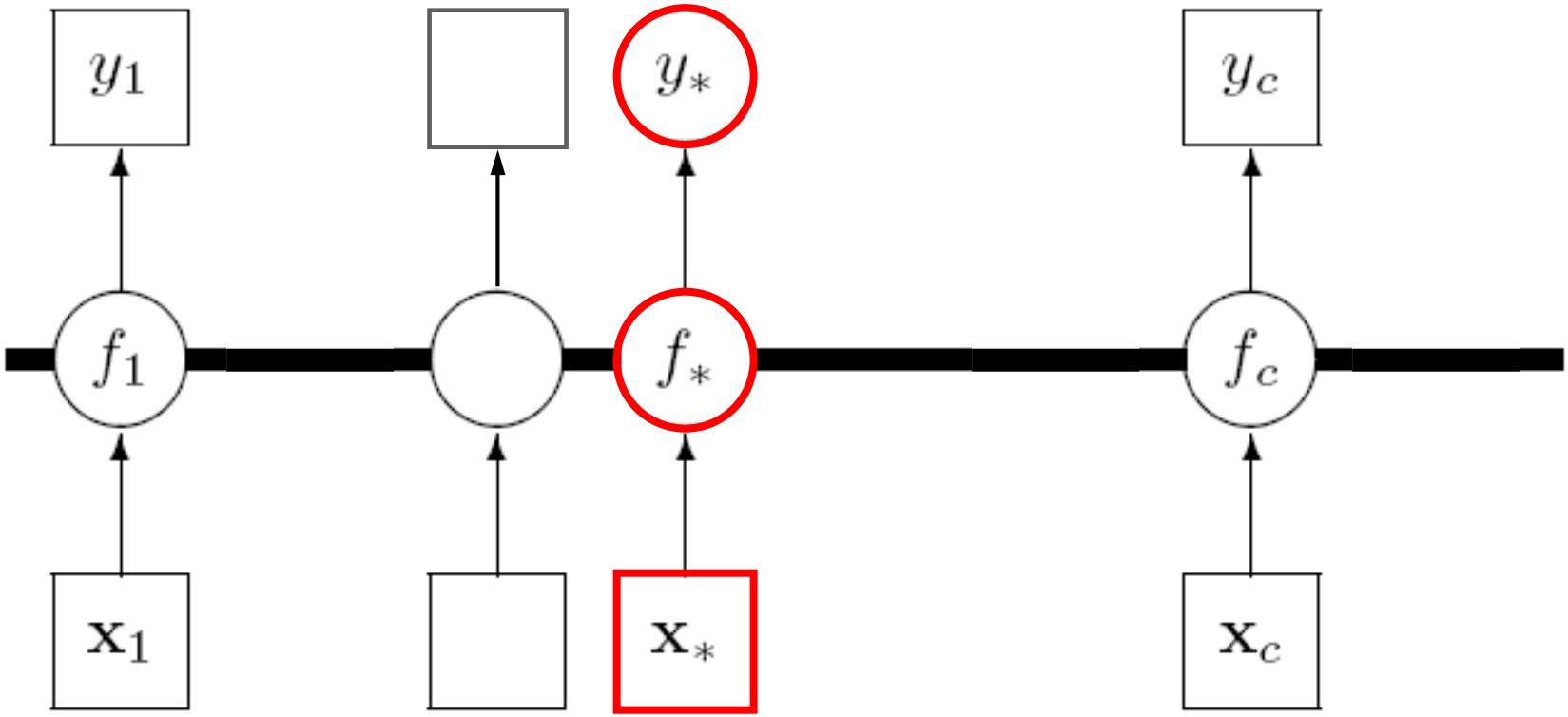


observed

unknown

by *Carl Rasmussen*

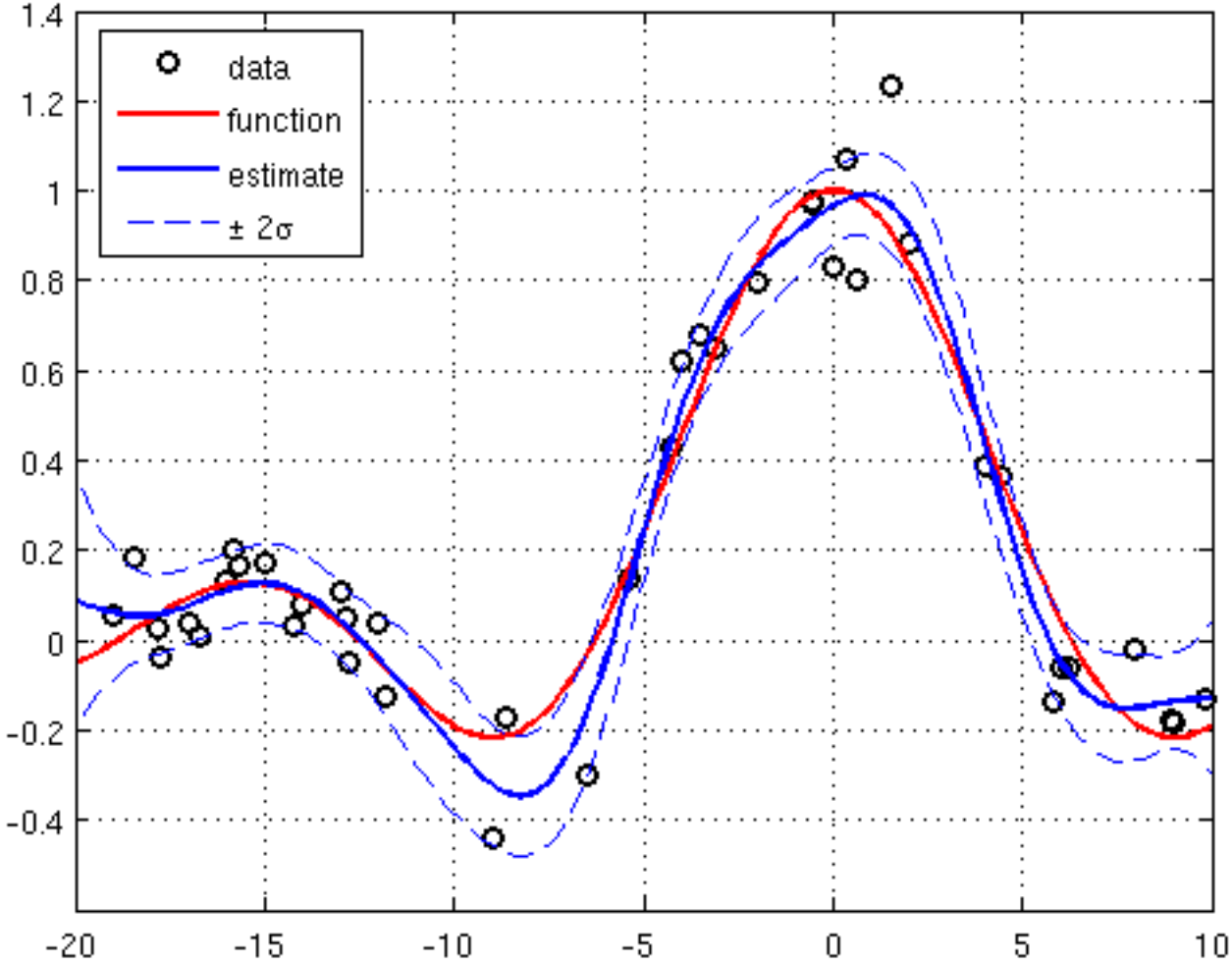
Sparse GP

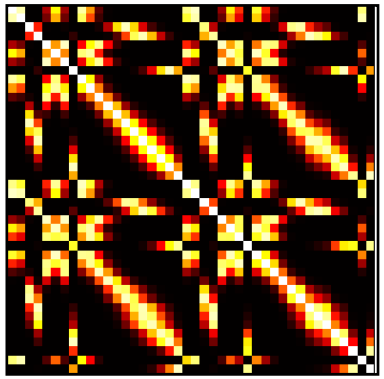


observed

unknown

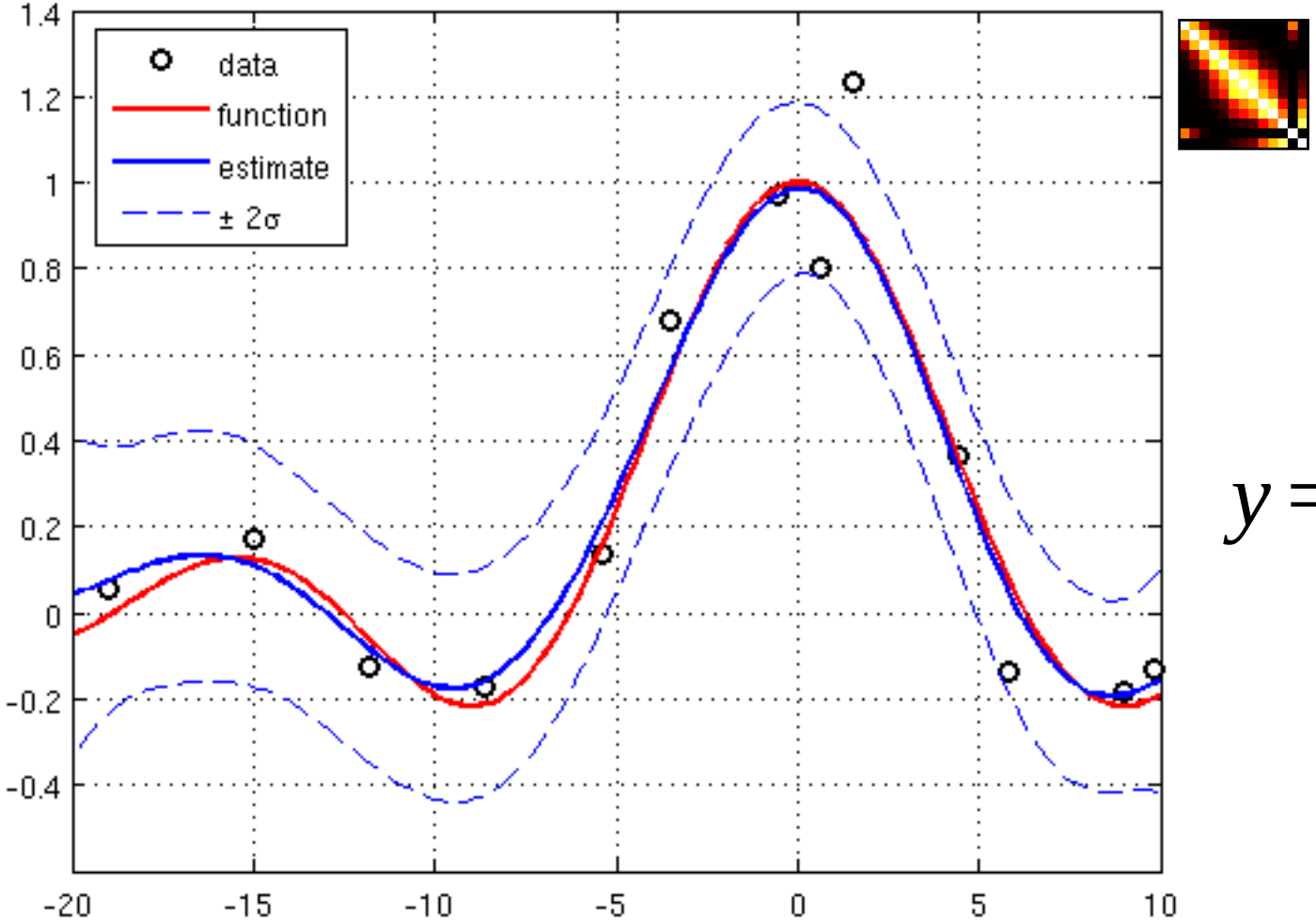
Full GPR (all the data)





$$y = K\alpha$$

Sparse GPR (subset of data)



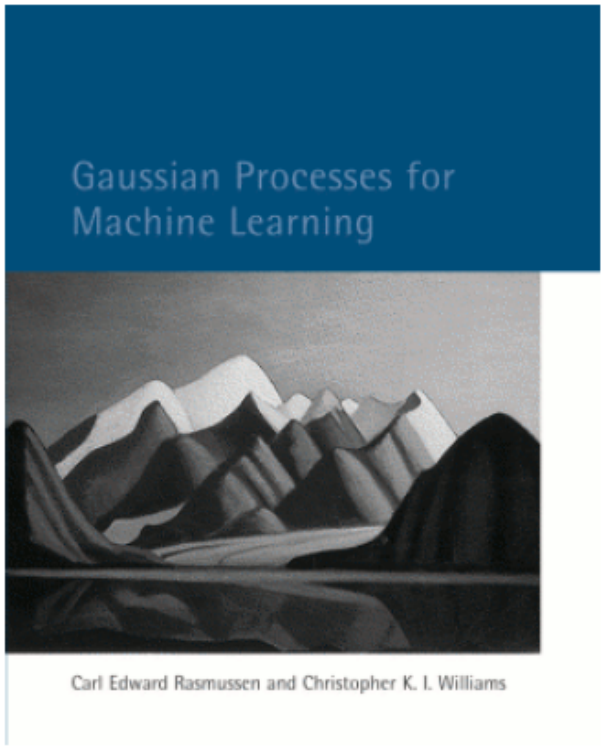
$$y = K_{ss}\alpha_s$$

Advantages of GPs

- uses probability theory (it's not a hack!)
- yields full predictive distributions
  - can be a building block : $p(y|x)$
  - posterior sampling
- automatic learning of kernel parameters
- principled (efficient) model selection
- ideal for limited training data
- have good generalization performance

The GP Bible (for ML folk)

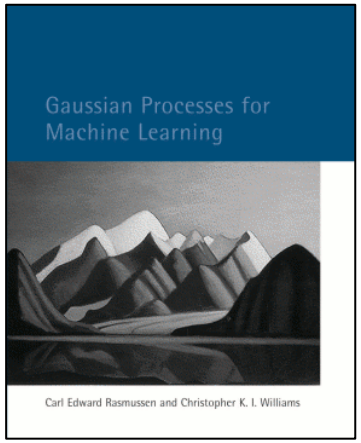
all the chapters
are available online!



The GP book: Rasmussen and Williams, 2006

Basic GP (Matlab) code available:
http://www.gaussianprocess.org/gpml/

Example: GPR Pseudocode



input: X (inputs), $\mathbf{y}$ (targets), k (covariance function), σ_n^2 (noise level), $\mathbf{x}_*$ (test input)

$L := \text{cholesky}(K + \sigma_n^2 I)$

$\boldsymbol{\alpha} := L^\top \backslash (L \backslash \mathbf{y})$

$\bar{f}_* := \mathbf{k}_*^\top \boldsymbol{\alpha}$ } predictive mean eq. (2.25)

$\mathbf{v} := L \backslash \mathbf{k}_*$

$\mathbb{V}[f_*] := k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{v}^\top \mathbf{v}$ } predictive variance eq. (2.26)

$\log p(\mathbf{y}|X) := -\frac{1}{2}\mathbf{y}^\top \boldsymbol{\alpha} - \sum_i \log L_{ii} - \frac{n}{2}\log 2\pi$ eq. (2.30)

return: $\bar{f}_*$ (mean), $\mathbb{V}[f_*]$ (variance), $\log p(\mathbf{y}|X)$ (log marginal likelihood)