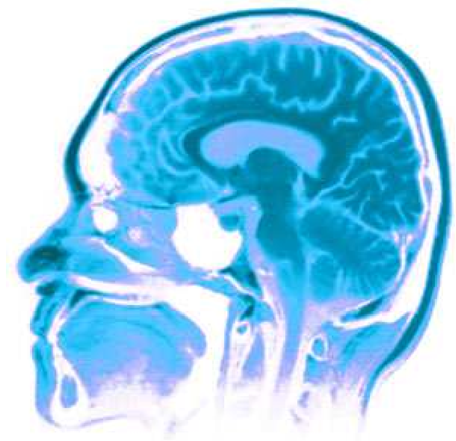
CPSC540



Gaussian Processes

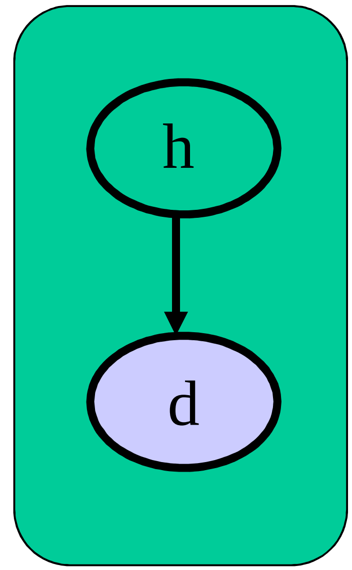
Nando de Freitas

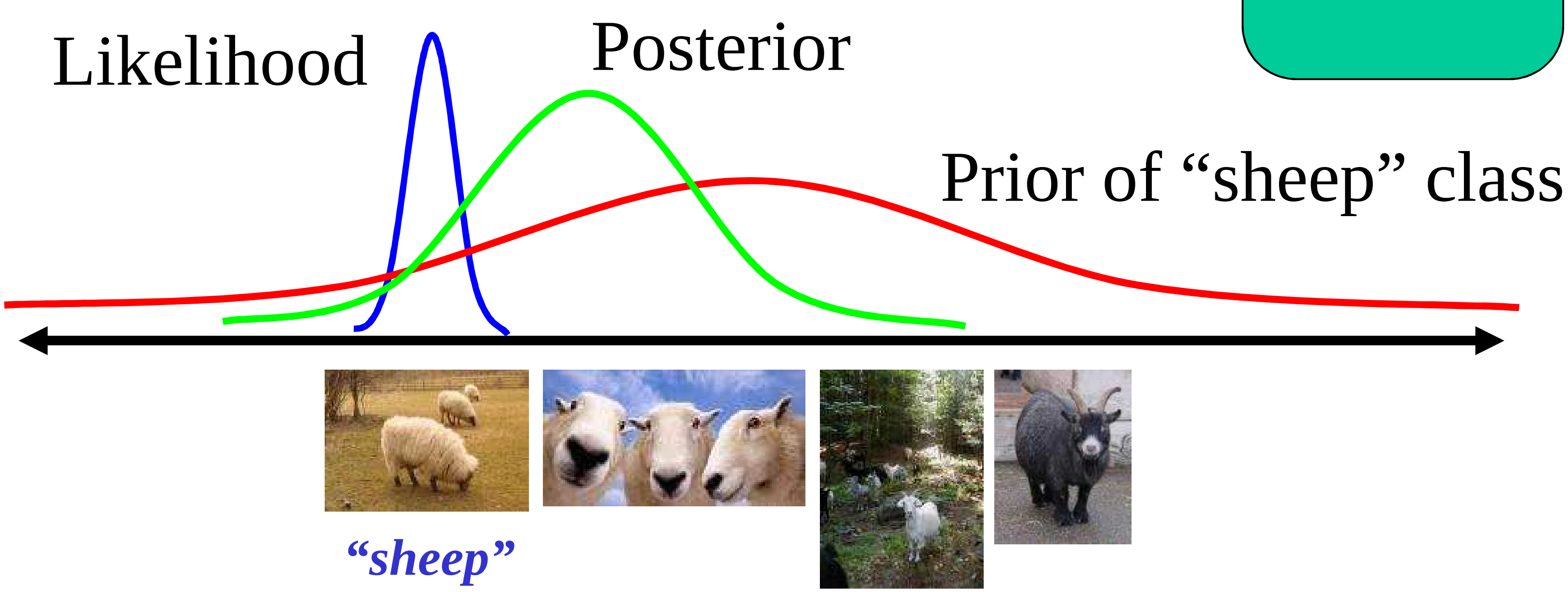
2011

KPM Book Sections: *16*

Learning and Bayesian inference

$$p(h \mid d) = \frac{p(d \mid h)\, p(h)}{\sum_{h' \in H} p(d \mid h')\, p(h')}$$





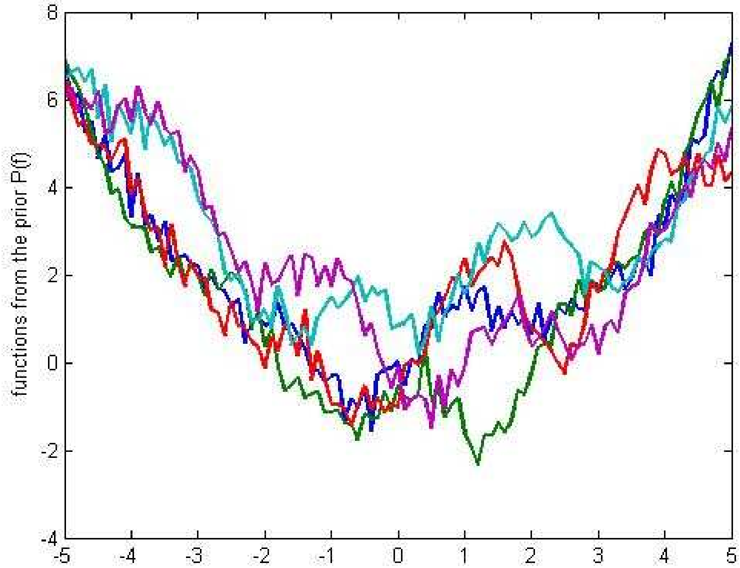
Nonlinear regression

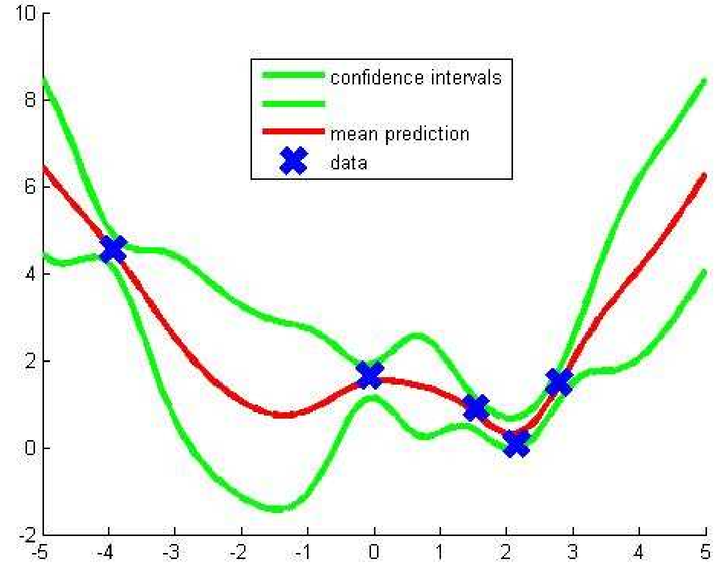
$$f \;\sim\; \mathcal{GP}(m,\; k)$$

$$m(x) = \frac{1}{4}x^2$$

$$k(x, x') = \exp(-\frac{1}{2}(x - x')^2)$$

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





$$\mu_i \;= m(x_i) \;=\; \frac{1}{4}x_i^2, \quad i = 1, \ldots, n \;\; \text{and}$$

$$\Sigma_{ij} \;= k(x_i, x_j) \;=\; \exp(-\frac{1}{2}(x_i - x_j)^2), \quad i, j = 1, \ldots, n$$

$$\mathbf{f} \;\sim\; \mathcal{N}(\mu, \Sigma)$$

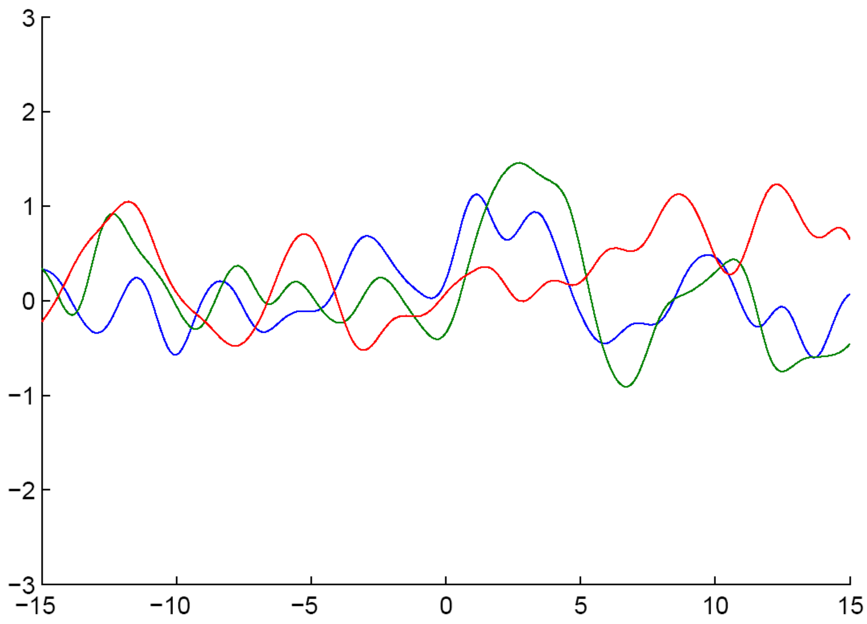
Sampling from *P(f)*

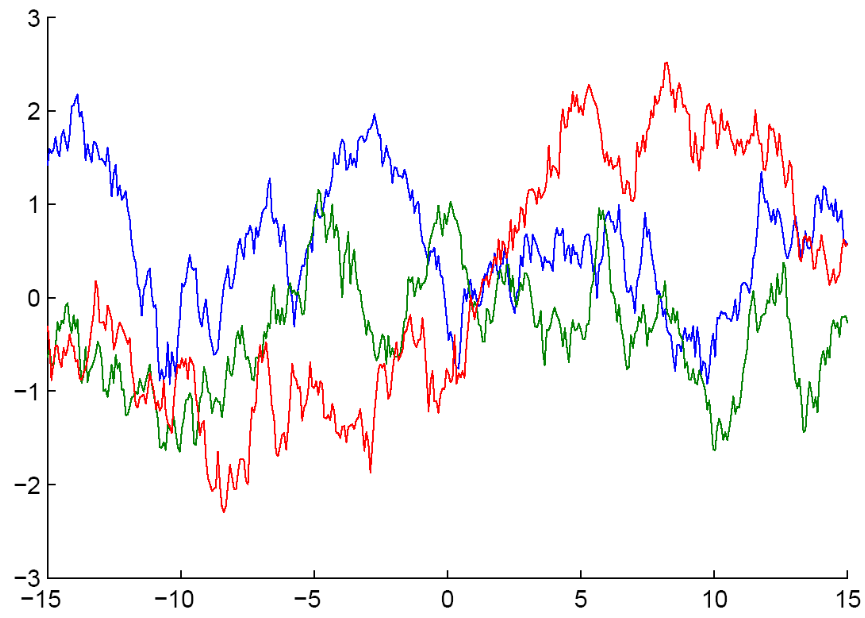
```
N = 5;                          % The number of training points.
sigma = 0.1                     % Noise variance.
h = 1;                          % Kernel parameter.

% Randomly generate training points on [-5,5].
X = -5 + 10*rand(N,1);
x = (-5:0.1:5)';                % The test points.
n = size(x,1);                  % The number of test points.

% Construct the mean and covariance functions.
m = inline('0.25*x.^2', 'x');    % another example:  m = inline('sin(0.9*x)','x');
K = inline(['exp((-1/(h^2))*(repmat(transpose(p),size(q)) - repmat(q,size(transpose(p)))).^2)'], 'p', 'q','h');


% Demonstrate how to sample functions from the prior:
L = 5;                          % Number of functions sampled from the prior P(f)
f = zeros(n,L);
for i=1:L,
  f(:,i) = m(x) + sqrtm(K(x,x,h)+sigma*eye(n))'*randn(n,1);
end;
plot(x,f,'linewidth',2);
```

$$r = \|\mathbf{x} - \mathbf{x}'\|$$

$$K_{\mathrm{SE}}(r) = a^2 \exp\left(-\frac{r^2}{2\lambda^2}\right)$$

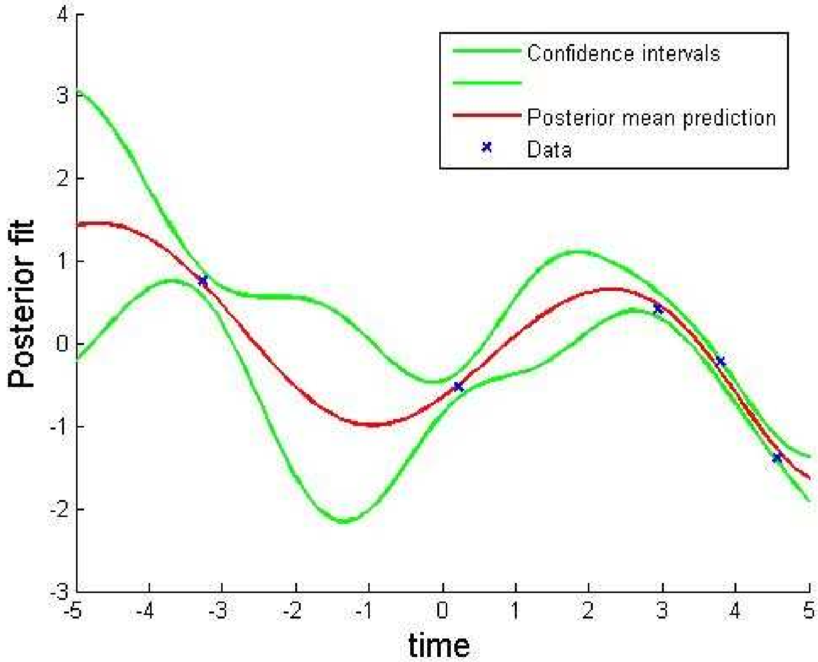
$$K_{\mathrm{Mat}}(r) = a^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu r}}{\lambda}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu r}}{\lambda}\right)$$

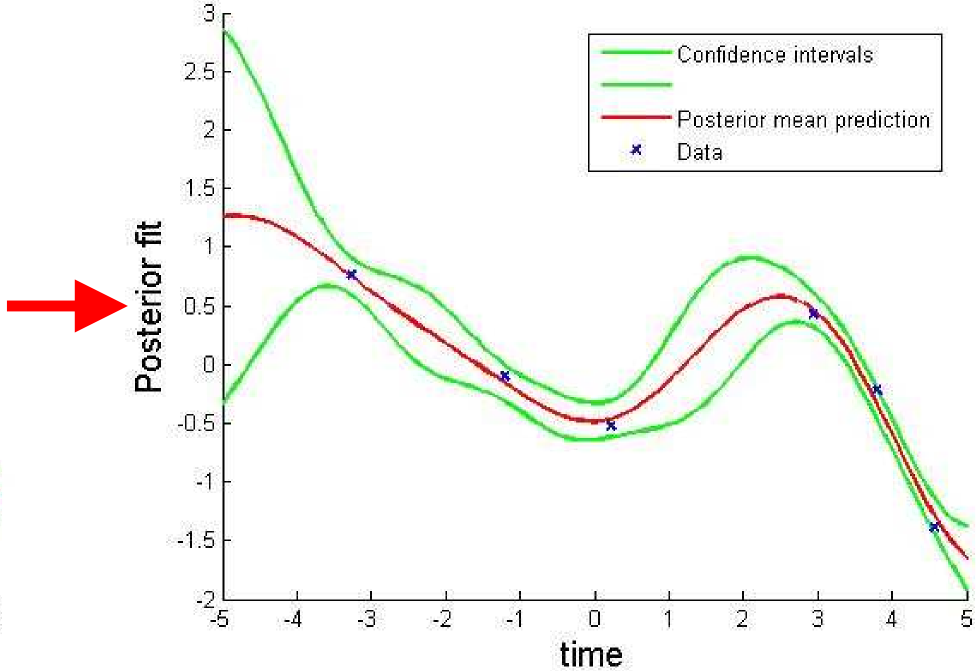


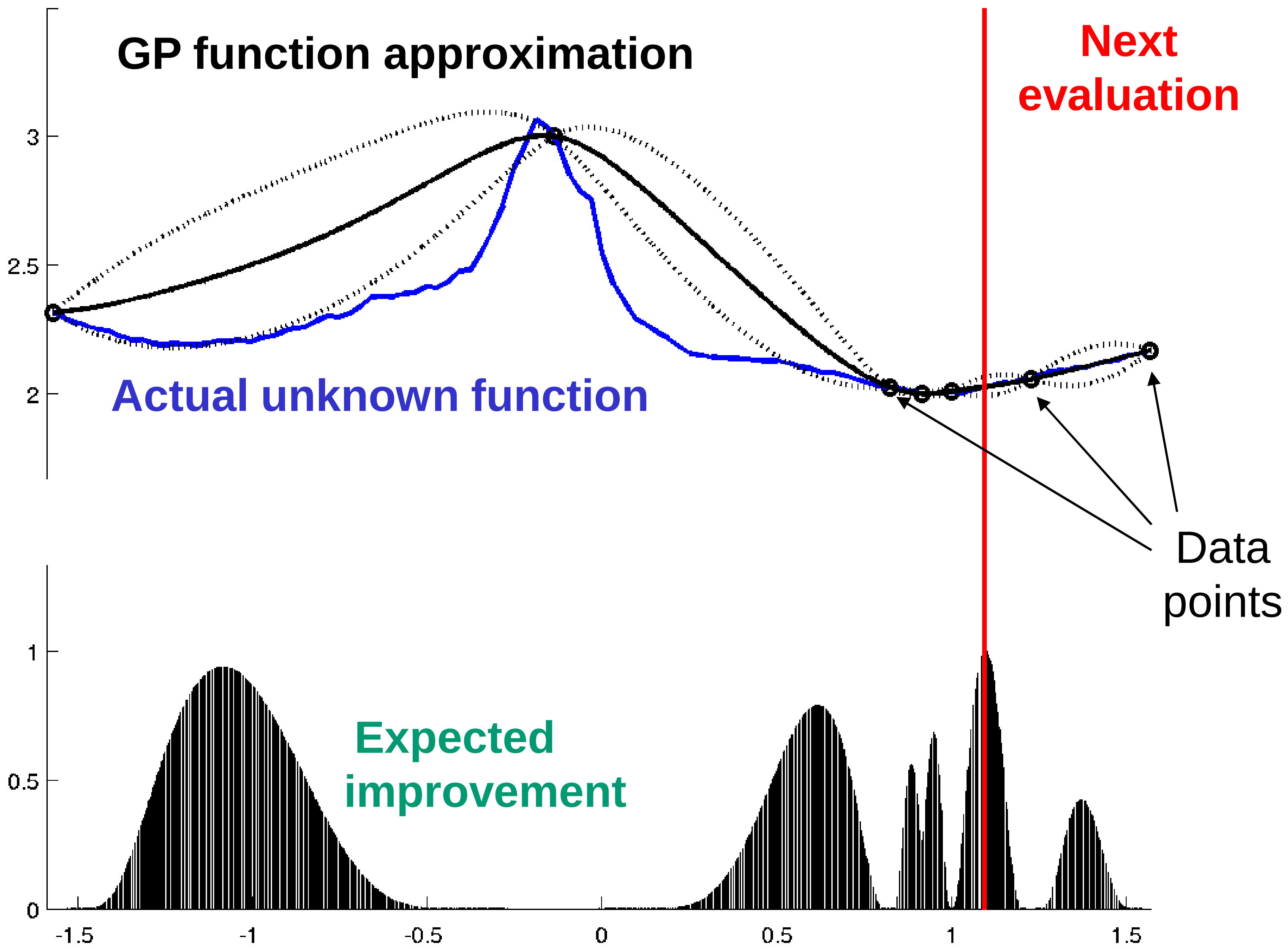


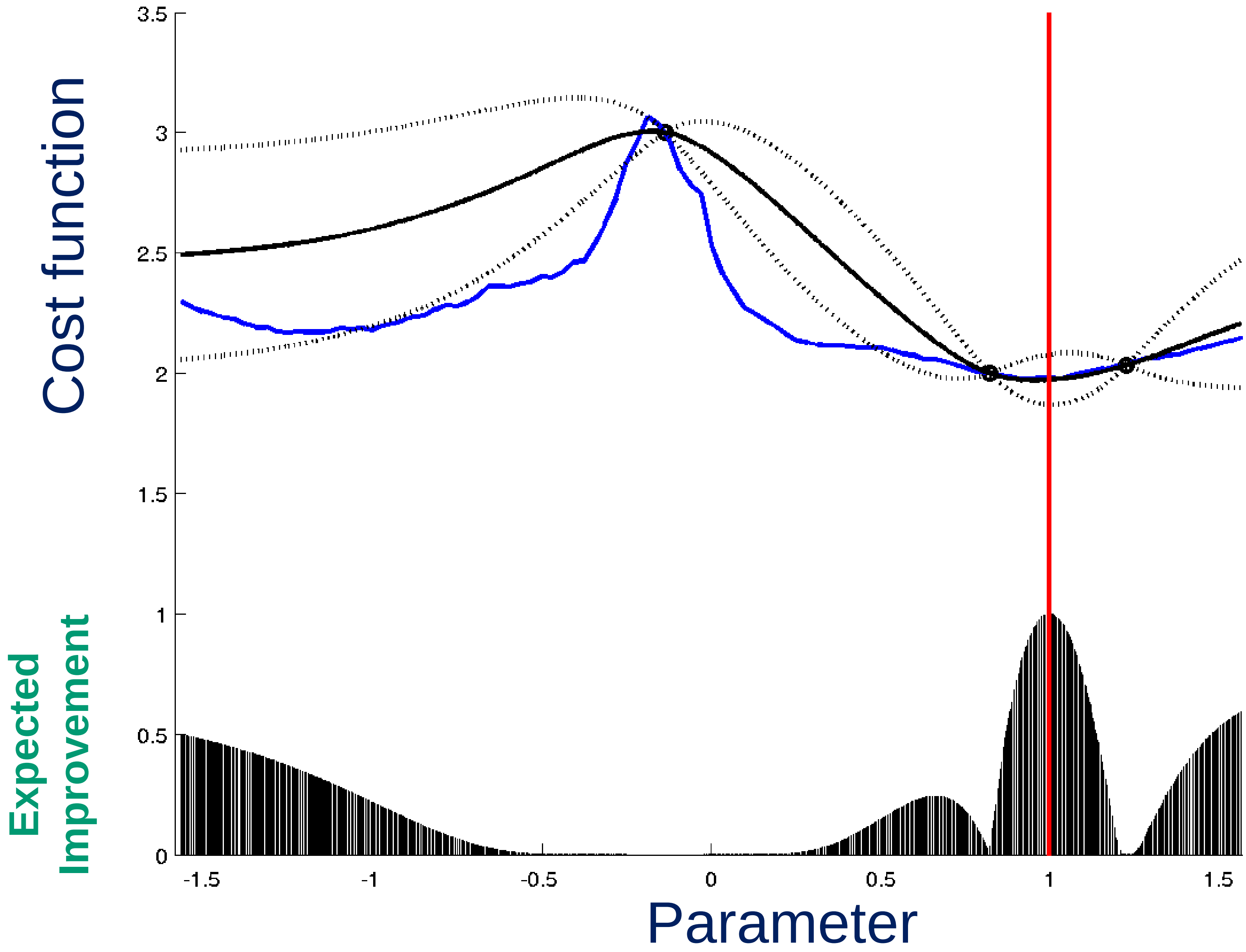
[Snelson, 2007]

Active learning with GPs



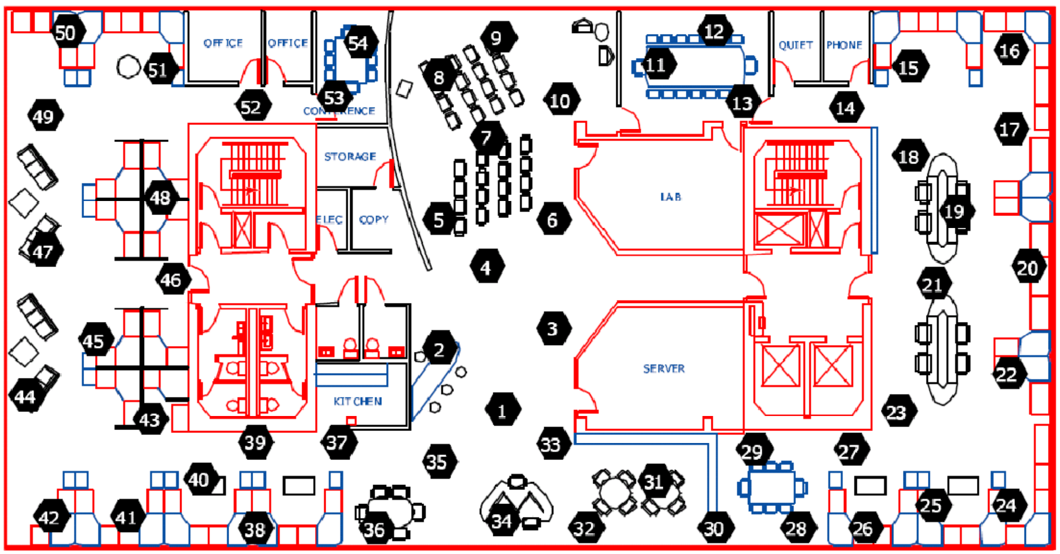


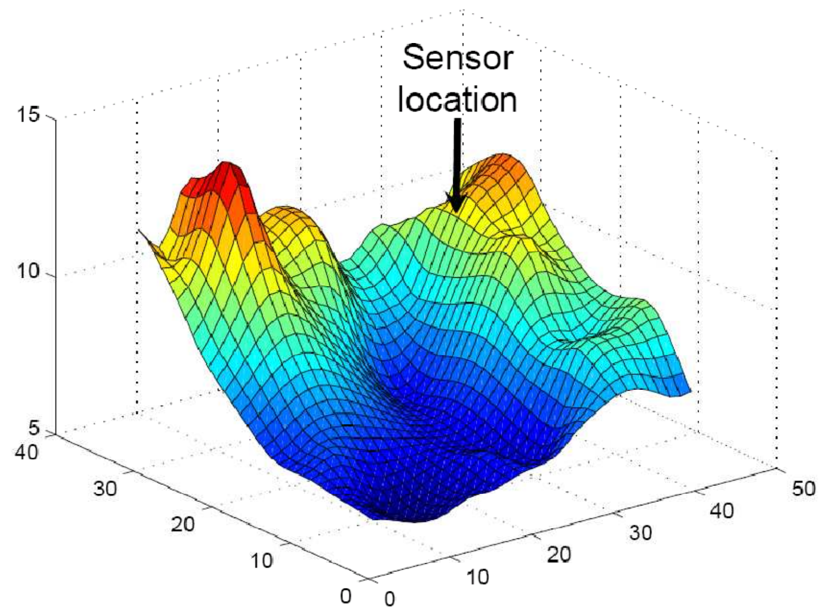

GP function approximation
Next evaluation
3
2.5
2
Actual unknown function
Data points
1
0.5
0
Expected improvement
-1.5
-1
-0.5
0
0.5
1
1.5



Cost function
Expected Improvement
Parameter
3.5
3
2.5
2
1.5
1
0.5
0
-1.5
-1
-0.5
0
0.5
1
1.5

2D sensor placement application by Andreas Krause and Carlos Guestrin





GP regression

Zero-mean GP prior

$$p(\mathbf{f}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$$

Gaussian noise / likelihood

$$y = f + \epsilon\,, \qquad \mathcal{E}[\epsilon(\mathbf{x})\epsilon(\mathbf{x}')] = \sigma^2 \delta_{\mathbf{x}\mathbf{x}'} \qquad p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{f}, \sigma^2 \mathbf{I})$$

The marginal likelihood (evidence) is Gaussian:

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

GP regression

$$(\mathbf{X}, \mathbf{f}, \mathbf{y}) = (\{\mathbf{x}_n\}, \{f_n\}, \{y_n\})_{n=1}^{N}$$ **Train set**

$$(\mathbf{X}_T, \mathbf{f}_T, \mathbf{y}_T) = (\{\mathbf{x}_t\}, \{f_t\}, \{y_t\})_{t=1}^{T}$$ **Test set**

Both sets are, by definition, jointly Gaussian:

$$p(\mathbf{f}, \mathbf{f}_T) = \mathcal{N}(\mathbf{0}, \mathbf{K}_{N+T})$$

$$\mathbf{K}_{N+T} = \begin{bmatrix} \mathbf{K}_N & \mathbf{K}_{NT} \\ \mathbf{K}_{TN} & \mathbf{K}_T \end{bmatrix}$$

The joint distribution of the measurements is:

$$p(\mathbf{y}, \mathbf{y}_T) = \mathcal{N}(\mathbf{0}, \mathbf{K}_{N+T} + \sigma^2 \mathbf{I})$$

GP regression

The predictive conditional distribution is Gaussian too:

$$p(\mathbf{y}_T|\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_T) \ ,$$

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

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

GP regression

```
% Generate random training labels.
F = m(X) + chol(K(X,X,h)+sigma*eye(N))'*randn(N,1);
M = m(X);

% COMPUTE POSTERIOR MEAN AND VARIANCE
S = eye(N)*sigma + K(X,X,h);
y = m(x) + K(X,x,h)*inv(S)*(F - M);
y = y';
for i = 1:n
  xi   = x(i);
  c(i) = sigma + K(xi,xi,h) - K(X,xi,h)*inv(S)*K(xi,X,h);
end

% Plot the mean and 95% confidence intervals.
plot(x,y-2*c,'g-','linewidth',3)
plot(x,y+2*c,'g-','linewidth',3)
plot(x,y,'r','linewidth',3)
plot(X,F,'bx','linewidth',15)
```

Parameter learning for GPs: maximum likelihood

$$\begin{aligned} L &= \log p(\mathbf{y}|\mathbf{x}, \theta) \\ &= -\frac{1}{2}\log|\Sigma| - \frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{y}-\boldsymbol{\mu}) - \frac{n}{2}\log(2\pi) \end{aligned}$$

For example, we can parameterize the mean and covariance:

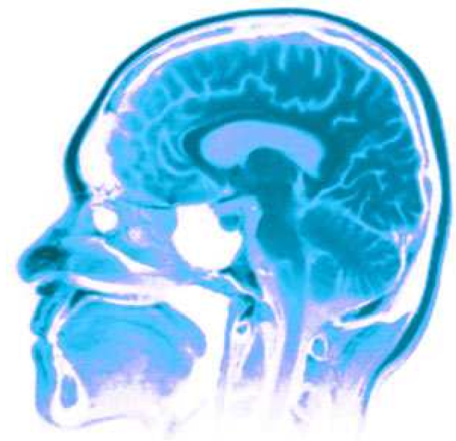
$$f \sim \mathcal{GP}(m,\ k),$$

$$m(x) = ax^2 + bx + c.$$

$$k(x, x') = \sigma_y^2 \exp\left(-\frac{(x-x')^2}{2\ell^2}\right) + \sigma_n^2 \delta_{ii'}$$

$$\theta = \{a, b, c, \sigma_y, \sigma_n, \ell\}$$

Next class



Directed Graphical Models

Nando de Freitas
2011
KPM Book Sections: *6*