Statistical Techniques in Robotics (16-831, F08) Lecture #17 (Tuesday October 21^{st})

Gaussian Processes

Lecturer: Drew Bagnell *Scribe: Veaceslav Arabagi*

1 Motivational examples for Gaussian Processes (GP)

- State of the art nonlinear regressions done with GP Bayes filter
- Used in reverse kinematics, dynamics, modeling
- Concrete example: *Modeling of soil properties* [1]
  - Task: Model the distribution of soil pH throughout Honduras land
  - A limited set of datapoints of soil pH at fixed locations through the country is provided
  - Variables influencing soil pH: sunlight, rainfall amount, topography, vegetation etc
  - Used state of the art GP Bayes filter to predict the mean soil pH and the prediction variance, or confidence (1)

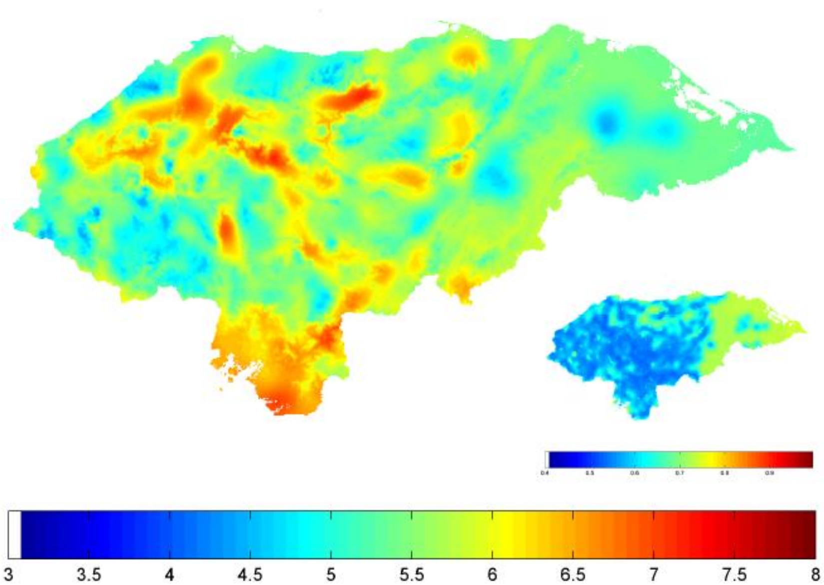

Figure 1: Predicted map of pH in topsoil and 67% confidence interval.

[1] J. P. Gonzalez, S. Cook, T. Oberthur, A. Jarvis, J. A. Bagnell and M. Bernardine, *Creating Low-Cost Soil Maps for Tropical Agriculture using Gaussian Processes*, JCAI 2007, Jan 2007.

2 High level idea behind GP's

- Cconsider a function as a really long list of numbers, a vector of numbers
- For discrete set of locations build a Gaussian distribution on the vector
- Generalize discrete covariance to continuous functions
- Graphically

Consider $f(\vec{x}) = [f(x1), f(x2), f(x3), \ldots, f(10)]^T$ then we can form a Gaussian distribution on this vector of the form:

$$p(f(\vec{x})) = \frac{1}{z} e^{-(\vec{f}-\mu_f)^T \Sigma^{-1} (\vec{f}-\mu_f)} \tag{1}$$

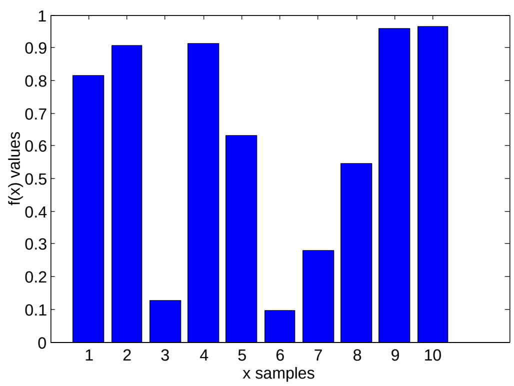

Figure 2: Sample points for GP filter

Generalizing the idea of a distribution of a vector $f(\vec{x})$ to continuous functions and using the expansion of the matrix exponent:

$$(\vec{f} - \mu_f)^T \Sigma^{-1} (\vec{f} - \mu_f) = \sum_i \sum_j (f_i - \mu_i) \Sigma_{ij}^{-}1 (f_i - \mu_i) \tag{2}$$

we obtain a distribution of continuous functions:

$$p(f) = \frac{1}{z} e^{-\frac{1}{2} \int dx \int dx' f(x) \Delta(x,x') f(x')} \tag{3}$$

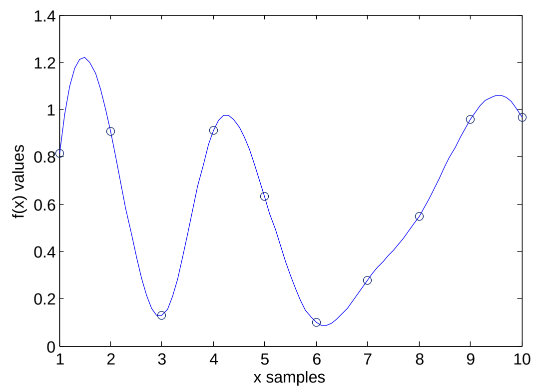

Figure 3: Sample function passing through training points for GP filter

However, in the continuous function case the normalizer $\frac{1}{z}$ is effectively an integral over all functions, which is infinite. This is a big problem, however there exists a solution: marginalize all functions by looking at a discrete number of points. To do that we apply the marginalization rules:

$$\mu_f = 0,$$
$$\Sigma_{ff} = \begin{pmatrix} \kappa(x_1, x_1) & \kappa(x_1, x_2) & \cdots & \kappa(x_1, x_n) \\ \kappa(x_2, x_1) & \kappa(x_2, x_2) & \cdots & \kappa(x_2, x_n) \\ \vdots & & & \vdots \\ \kappa(x_n, x_1) & \kappa(x_n, x_2) & \ldots & \kappa(x_n, x_n) \end{pmatrix} \tag{4}$$

where $\kappa(x, x')$ is the kernel function and effectively portrays how each point x influences point x'. The kernel function is one of the most important concepts in Gaussian processes and is used to form the covariance Σ.

2.1 Kernel functions

Kernel functions must be positive definite and symmetric as the matrix Σ they for has those properties.
Sample kernel functions:

- $\kappa(x,x') = e^{\frac{-(x-x')^2}{L}}$
  This is one of the simplest kernel functions that works surprisingly well. The parameter L is the characteristic length and defines the distance over which the correlation of 2 points (x,x') is effective. The smaller the length scale, the less each point will influence its neighbor.
- $\kappa(x,x') = \delta(x-x')$
  Effectively says that each point has no influence on its neighbor.
- $\kappa(x,x') = \vec{x}^T\vec{x'}$
  This kernel function results in Bayes linear regression.

3 How to apply algorithm

Consider a few datapoints in $\mathbb{R}^2$ as shown on right. We are interested in predicting the value of $f(x^*)$.

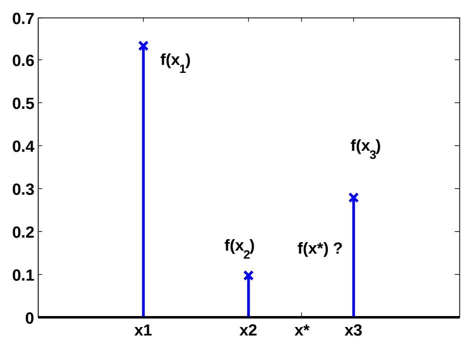

Figure 4: Training datapoints x1,x2,x3 and one unknown datapoint $f(x^*)$.

Let $\vec{x} \equiv [x_1, x_2, x_3]^T$ and $\vec{f} \equiv [f(x_1), f(x_2), f(x_3)]^T$ then we have the distribution of points:

$$\begin{pmatrix} \vec{f} \\ f(x^*) \end{pmatrix} \sim \mathbb{N}\left(0, \begin{bmatrix} K_{\vec{x}\vec{x}} & K_{\vec{x}x^*} \\ K_{x^*\vec{x}} & K_{x^*x^*} \end{bmatrix}\right) \tag{5}$$

We are interested in the conditional distribution $p = f(x^*)|f(x_1)f(x_2)f(x_3)$ therefore we can compute the mean and variance of the value $f(x^*)$ using rules for conditionalizing the Gaussian distribution.

$$\boxed{\begin{aligned} \mu_{f^*} &= K_{\vec{x}x^*}^T K_{\vec{x}\vec{x}}^{-1} \vec{f} \\ \Sigma_{f^*} &= K_{x^*x^*} - K_{\vec{x}x^*}^T K_{\vec{x}\vec{x}}^{-1} K_{\vec{x}x^*} \end{aligned}}$$