Lecture 3: Gaussian Process Regression - fitting GPs to data

Motivation: we wish to recover the underlying smooth temperature field at unobserved locations

Two ingredients: 1. a prior probability model over a suitable function space
2. likelihood function for observed data

Ingredient 1: Gaussian Process Prior model

Two views: weigh-space view, function space view
(GPs for Machine Learning, Rasmussen & Williams)

Weight space view: map inputs $t \in D$ into a high-dimensional feature space spanned by basis functions $\Phi(t) = (\varphi_1(t), \dots, \varphi_p(t))^T$

$$X(t) = \Phi(t)^T w, \qquad w = (w_1, \dots, w_p)^T \in \mathbb{R}^p$$

we then place a MVN prior on the weights as

e.g. $\quad \underline{w} \sim N_p(\underline{0}, \Sigma_w)$

then $X(t) \sim N\left(\underbrace{\Phi(t)^T E(w)}_{0}, \ \Phi(t)^T \Sigma_w \Phi(t)\right)$

The function space view: Assume that $\{X(t) : t \in D\}$ is a GP

$$X \sim GP(m_0(t), C_0(t,s))$$

(for most applications $m_0(t) = 0$)

We define the covariance function directly as

$$C(t,s) = \Phi(t)^T \Sigma_w \Phi(s)$$

Instead of defining a distribution over w, we define a distribution directly over functions X. This allows us to use infinite-dimensional basis sets.

Inaredient 2: Likelihood of the data-generating model

sets.

Ingredient 2: Likelihood of the data-generating model

Note on notation: we will use notation $C(t,t)$ where $t \in \mathbb{R}^n$ to describe a $n \times n$ matrix with $(i,j)^{th}$ element $C(t_i, t_j)$

We observe a finite number of noisy observations $y = (y_1, \dots, y_n)^T$ at input locations $t = (t_1, \dots, t_n)^T$, $t_i \in D$. Our model is:

$$Y_i = X(t_i) + \varepsilon_i, \quad \varepsilon_i \overset{ind}{\sim} N(0, \sigma^2)$$

The covariance of the observed values $Y = (Y_1, \dots, Y_n)$

$$\mathrm{Cov}(Y) = C_0(t,t) + \sigma^2 I_{n \times n}$$

where $C_0(t,t)$ is the $n \times n$ covariance matrix with entries $C_{0\,ij} = C_0(t_i, t_j)$.

GP Regression: Posterior Predictive Inference

Let's consider predictions at discrete locations $t^* = (t_1, \dots, t_m)^T$.

To predict the values of $X(t^*)$ at unobserved locations $t^* \in D^m$ we write the joint distribution of Y and $X(t^*)$ as

$$\begin{bmatrix} Y \\ X(t^*) \end{bmatrix} \sim N_{n+m}\left(0, \begin{bmatrix} C_0(t,t) + \sigma^2 I_{n \times n} & C_0(t,t^*) \\ C_0(t^*,t) & C_0(t^*,t^*) \end{bmatrix}\right)$$

Next, we use properties of MVN to find the conditional distribution of $X(t^*)$ given y:

$$X(t^*) \mid Y=y \sim N_m\left(m_n(t^*), C_n(t^*, t^*)\right)$$

where

$$m_n(t^*) = C_0(t^*, t)\left[C_0(t,t) + \sigma^2 I\right]^{-1} y$$

where

$$m_n(t^*) = C(t^*, \underline{t})\left[C_0(\underline{t},\underline{t}) + \sigma^2 I_{n\times n}\right]^{-1} y$$

$$C_n(t^*, t^*) = C_0(t^*, t^*) - C_0(t^*, t)\left[C_0(t,t) + \sigma^2 I_{n\times n}\right]^{-1} C_0(t, t^*)$$

<u>Comment</u>:
- Bayesian sequential updating can be conducted in sequence by taking the posterior from one update to be the prior for the next update (Kalman filitering)
- Computational cost of updating a GP given n data points is proportional to n^3

<u>Example 1</u>: one-dimensional domain
GP_prior_to_posterior_update.R

<u>Example 2</u>: two-dimensional domain
GP_regression_temperature_Ukraine.R

<u>Infinite-dimensional case</u> - make inference over all of D

Define linear operator $\mathcal{L}: \mathcal{H} \to \mathcal{H}'$ (e.g. integration, differentiation, composition)

<u>Prior model</u>: $X \sim GP(m_0(t), C_0(t,s))$

<u>Data generating model</u>: $Y_i = \mathcal{L}X(t_i) + \varepsilon_i$, $\varepsilon_i \overset{ind}{\sim} N(0, \sigma^2)$

$t = (t_1, \ldots, t_n)^T$

<u>Posterior</u>: $X \sim GP(m_n(t), C_n(t,s))$

$t^*, s^* \in D$

$$m_n(t^*) = m_0(t^*) + C_0\mathcal{L}^*(t^*, t)\left[\mathcal{L}C_0\mathcal{L}^*(t,t) + \sigma^2 I_{n\times n}\right]^{-1}(y - \mathcal{L}m_0(t))$$

$$C_n(t^*, s^*) = C_0(t^*, s^*) - C_0\mathcal{L}^*(t^*, t)\left[\mathcal{L}C_0\mathcal{L}^*(t,t) + \sigma^2 I_{n\times n}\right]^{-1}\mathcal{L}C_0(t, s^*)$$

adjoint of $\mathcal{L}$