GP models for regression

~~W2b~~

Th2a 1

Back to the prediction of $t^{(N+1)}$ given t_N.

$$p(t_{(N+1)}, t_N) = p(t^{(N+1)} | t_N)\, p(t_N)$$

joint density is Gaussian → conditional distribution must be Gaussian ← Gaussian

$$\Downarrow$$

must be Gaussian

Notation: $C_{N+1} = (N+1)\times(N+1)$ cov. matrix for $t_{N+1} = (t^{(1)}, \dots, t^{(N)}, t^{(N+1)})$

write as

$$C_{N+1} \equiv \begin{pmatrix} [C_N] & [k] \\ [k^T] & \kappa \end{pmatrix}$$

and we get

$$p(t^{(N+1)} | t_N) \propto \exp\left[-\frac{1}{2}[t_N, t^{(N+1)}]\, C_{N+1}^{-1} \begin{bmatrix} t_N \\ t^{N+1} \end{bmatrix}\right)$$

$\Rightarrow$ we need C_{N+1}^{-1}

More elegant:

$$C_{N+1}^{-1} = \begin{bmatrix} M & m \\ m^T & \mu \end{bmatrix}$$

where

$$\mu = \left(\varkappa - k^T C_N^{-1} k\right)^{-1}$$

$$m = -\mu C_N^{-1} k$$

$$M = C_N^{-1} + \frac{1}{\mu} m m^T$$

so that

$$p(t^{(N+1)} \mid t_N) = \frac{1}{Z} \exp\left[-\frac{\left(t^{(N+1)} - \hat{t}^{(N+1)}\right)^2}{2\sigma_{\hat{t}_{N+1}}^2}\right]$$

with

mean: $\hat{t}^{(N+1)} = k^T C_N^{-1} t_N$

variance: $\sigma_{\hat{t}_{N+1}}^2 = \varkappa - k^T C_N^{-1} k$

We can make a prediction for t^{N+1} in the form of a Gaussian (i.e. we have an uncertainty) with a cost N^3 (inversion of C_N)

In fact, we might as well predict M new target values $\mathbf{t}_M = \{ t^{(N+i)} \}_{i=1}^{M}$ at once.

$N \times M$

$$C_{N+M} = \begin{pmatrix} [C_N] & [\quad k \quad] \\ [k^T] & [\quad \kappa \quad] \end{pmatrix}$$

$M \times M$

$$p(t_M | t_N) = \frac{1}{Z} \exp\left[-\frac{1}{2} (t_M - \hat{t}_M)^T \Sigma_M^{-1} (t_M - \hat{t}_M) \right]$$

where the mean vector

$$\hat{t}_M = \underset{M \times N}{k^T} \; \underset{N \times N}{C_N^{-1}} \; \underset{N \times 1}{t_N} \qquad (M \times 1)$$

and a covariance matrix Σ_M

$$\Sigma_M = \kappa - k^T C_N^{-1} k \qquad (M \times M)$$

How do we fix C_N

(by adjusting hyperparameters θ in)

$$C(x, x'; \theta)$$

?

Well, recall that

$$p(t_N) = \frac{1}{Z} e^{-\frac{1}{2} t_N^T C_N^{-1} t_N}$$

where $[C_N]_{nn'} = C(x^{(n)}, x^{(n')}; \theta) + \sigma_\nu^2 \delta_{nn'}$

A frequentist would find the maximum of $p(t_N)$ or $\log(p)$

$\Rightarrow$ minimize $\underset{1\times N}{t_N^T} \; \underset{N\times N}{C_N^{-1}} \; \underset{N\times 1}{t_N}$

A Bayesian might instead marginalize over θ.

$$p(t_M | t_N) = \int d\theta \; p(t_M, \theta | t_N)$$

$$= \int d\theta \; p(t_M | \theta, t_N) \, p(\theta)$$

[Note: rarely performed in ML community]

GP emulator

Is a fast and approximate mimic of a full simulation model.

Predict $y = f(x)$ at x where both y and x can be multi-dim.

It is often structured as follows

$$f_i(x) = \sum_j \beta_{ij}\, g_{ij}(x_{A,i}) + u_i(x_{A,i}) + w_i(x)$$

where $i = 1, \ldots, q$ is the output dimension

Active inputs $x_{A,i}$

We can attempt to reduce the input dimension from D (x) to p (x_A) by e.g. AIC, BIC, etc.

The remaining inactive inputs should be captured by the nugget term $w(x)$

Regression model $\beta_{ij}\, g_{ij}(x_A)$

- where g_{ij} are known deterministic functions of x_A (often a low poly.) and β_{ij} are regression coeff.

Often the large scale global behavior of $f(x)$ is mimiced with a low-order (≤ 3) polynomial.

<u>Gaussian process</u>

$u(x)$ mimics the local behavior of $f(x) - \beta g(x)$

This term will produce a mean and a variance for its output $\Rightarrow$ uncertainty.

<u>Nugget term</u>

A white noise process

$$\mathrm{Cov}(w_i(x), w_i(x')) = \sigma_{w_i}^2 \, \delta(x - x')$$

<u>Training</u>

Model evaluations performed at well chosen points in the input space. Typically use space-filling methods (such as LHS)