Gaussian Processes: A One-Page Guide

Aditya Sengupta

September 2020

Disclaimer: this is not at all (could not possibly be) comprehensive. I just wanted to see if I could get the basics into one page!

1 Motivation

Consider time-series data $(t_i, y_i)_{i=1}^N$ to which we want to fit a model $y = \mathcal{N}(f_\theta(t), \Sigma)$. The overall (maximum-likelihood) problem is to find the $\hat{\theta}$ that maximizes the probability that the y_is actually are from that model. In practice, covariance matrices Σ that users provide can only be diagonal; it's difficult to provide a physically-motivated covariance between each pair of datapoints. The idea of Gaussian process regression is to provide a fit to a general framework like this, that will achieve this aim of covariance and fit time-series data better without needing more physical information than is really possible.

2 Math

The Gaussian pdf is $p(y; \mu, \Sigma) = (2\pi)^{-\frac{N}{2}} \det(\Sigma)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\vec{y} - \mu)\Sigma^{-1}(\vec{y} - \mu)\right)$. Plugging in our mean model $f_\theta(t)$ for μ, evaluating this over our datapoints, and taking a log (for numerical stability, to convert products to sums) we get a likelihood function to maximize:

$$\mathcal{L}(\theta) = \log\, p(\{y_n\} \,|\, \theta) = -\frac{1}{2} \sum_{n=1}^{N} \left[\frac{(y_n - f_\theta(t_n))^2}{\sigma_n^2 + s^2} + \log 2\pi(\sigma_n^2 + s^2) \right],$$

where we describe the covariance matrix as being $\sigma_n^2 + s^2$ down the diagonal and 0 elsewhere. Here, σ_n^2 parameterizes *modeled* (physically-known) variance, while s^2 parameterizes *unmodeled* independent variance and can be tuned. In general s can also vary with time.

We can define this more compactly in vector form: let $\vec{r}_\theta = \begin{bmatrix} y_1 - f_\theta(t_1) & y_2 - f_\theta(t_2) & \dots & y_N - f_\theta(t_N) \end{bmatrix}^\intercal$ and let $C = \left[(\sigma_i^2 + s^2)\delta_{ij}\right]_{1 \le i,j \le N}$. Then the form of the log-likelihood we can work with is

$$\log\, p(\{y_n\} \,|\, \theta, s) = -\frac{1}{2} \vec{r}_\theta^\intercal C^{-1} \vec{r}_\theta - \frac{1}{2} \log\det C - \frac{N}{2} \log 2\pi.$$

Here we bring in the Gaussian process part by defining a covariance relationship. Currently, C lacks any *co*variance information, i.e. we're assuming every (t_i, y_i) is independent of every other (t_j, y_j), which is not physically true as it is a time-series and they're influenced by the same systematics. To fix that, let the new covariance matrix K be parameterized by a set of parameters α:

$$K_\alpha = C + \left[k_\alpha(t_i, t_j)\right]_{1 \le i,j \le N}.$$

α is a set of hyperparameters that we want to fit, ideally with low dimension – if you allowed a joint fit to every element of the new covariance matrix, you'd have $N^2 - N$ parameters and would just overfit. The function k_α is a *kernel* function, and is also a *radial basis function*: it only depends on the distance between its inputs. This aligns with what we might want with time-domain covariance, as closer points are more highly correlated. Some common forms of useful kernel functions are:

Squared exponential	$k_\alpha(t_i, t_j) = \alpha^2 \exp\left(-\frac{1}{2}\frac{(t_i - t_j)^2}{l^2}\right)$
Squared exponential + sinusoid	$k_\alpha(t_i, t_j) = \alpha^2 \exp\left(-\frac{1}{2}\frac{(t_i - t_j)^2}{l^2}\right)\cos\left(\frac{2\pi(t_i - t_j)}{p}\right)$
Periodic	$k_\alpha(t_i, t_j) = \alpha^2 \exp\left(-\frac{2\sin^2(\pi\lvert t_i - t_j\rvert/p)}{l^2}\right)$

Gaussian process regression is just optimizing collectively (using nonlinear optimization or MCMC) over (θ, s, α): respectively, mean model parameters, the unmodeled noise parameter, and the covariance parameters. Formally, the optimization problem comes straight out of the Gaussian likelihood with the new covariance definition:

$$(\hat{\theta}, \hat{s}, \hat{\alpha}) = \arg \max_{\theta, s, \alpha} \left[-\frac{1}{2} \vec{r}_\theta^\intercal K_{\alpha,s}^{-1} \vec{r}_\theta - \frac{1}{2} \log\det K_{\alpha,s} - \frac{N}{2} \log 2\pi \right]$$

3 Further Issues to Investigate

- Precisely computing covariance becomes costly: it's $O(N^3)$. Subsample data, get a better computer, approximately compute results.
- Choice of the best kernel function and/or way to characterize s; varies by the science case and systematics!

References (and better follow-up guides for detailed knowledge!)

1. "Gaussian Processes Tutorial", talk by Daniel Foreman-Mackey at the Kavli Institute for Theoretical Physics, UC Santa Barbara. `http://online.kitp.ucsb.edu/online/exostar19/foremanmackey/`
2. "Gaussian Processes for Machine Learning", C. E. Rasmussen & C. K. I. Williams. `http://www.gaussianprocess.org/gpml/chapters/RW.pdf`
3. "The Kernel Cookbook: Advice on Covariance Functions", David Duvenaud. `https://www.cs.toronto.edu/~duvenaud/cookbook/`