The Change Point Kernel

Andrew Gordon Wilson

November 12, 2013

Saatchi et al. (2010) and Osborne (2010), for instance, have introduced Gaussian process models for change points. In this note, we discuss how the Gaussian process regression network (Wilson et al., 2012, 2011), and minor adaptations to the GPRN, can also provide a framework for change point modelling.

The GPRN is designed to model a p dimensional function $\boldsymbol{y}(x)$, with signal and noise correlations that vary with $x \in \mathcal{X}$, an arbitrary input space (although we are typically interested in $x \in \mathbb{R}^M$).

A GPRN models $\boldsymbol{y}(x)$ as

$$\boldsymbol{y}(x) = W(x)[\boldsymbol{f}(x) + \sigma_f \boldsymbol{\epsilon}] + \sigma_y \boldsymbol{z}\,, \tag{1}$$

where $\boldsymbol{\epsilon} = \boldsymbol{\epsilon}(x)$ and $\boldsymbol{z} = \boldsymbol{z}(x)$ are respectively $\mathcal{N}(0, I_q)$ and $\mathcal{N}(0, I_p)$ white noise processes. I_q and I_p are $q \times q$ and $p \times p$ dimensional identity matrices. $W(x)$ is a $p \times q$ matrix of independent Gaussian processes such that $W(x)_{ij} \sim \mathcal{GP}(0, k_w)$, and $\boldsymbol{f}(x) = (f_1(x), \ldots, f_q(x))^\top$ is a $q \times 1$ vector of independent GPs with $f_i(x) \sim \mathcal{GP}(0, k_{f_i})$. The GPRN prior on $\boldsymbol{y}(x)$ is induced through GP priors in $W(x)$ and $\boldsymbol{f}(x)$, and the noise model is induced through $\boldsymbol{\epsilon}$ and $\boldsymbol{z}$.

We represent the GPRN in Figure 1.

The latent *node functions* $\hat{\boldsymbol{f}}(x)$ are connected together to form the outputs $\boldsymbol{y}(x)$. The strengths of the connections change as a function of x; the weights themselves – the entries of $W(x)$ – are functions. Old connections can break and new connections can form. Indeed the high level idea behind the GPRN – a graphical model with connections which vary with the inputs – only depends on the nodes and weights being functions of x (they do not need to be GPs). This is an *adaptive* network, where the signal and noise correlations between the components of $\boldsymbol{y}(x)$ vary with x.

We label the length-scale hyperparameters for the kernels k_w and k_{f_i} as $\boldsymbol{\theta}_w$ and $\boldsymbol{\theta}_f$ respectively. We often assume that all the weight GPs share the same covariance kernel k_w, including hyperparameters. Roughly speaking, sharing length-scale hyperparameters amongst the weights means that, a priori, the strengths of the connections in Figure 1 vary with x at roughly the same rate.

Underlying the GPRN is a non-stationary kernel which can be learned from the data. Conditioned on the weights $W(x)$, each of the outputs $\boldsymbol{y}_i(x)$, $i = 1, \ldots, p$,

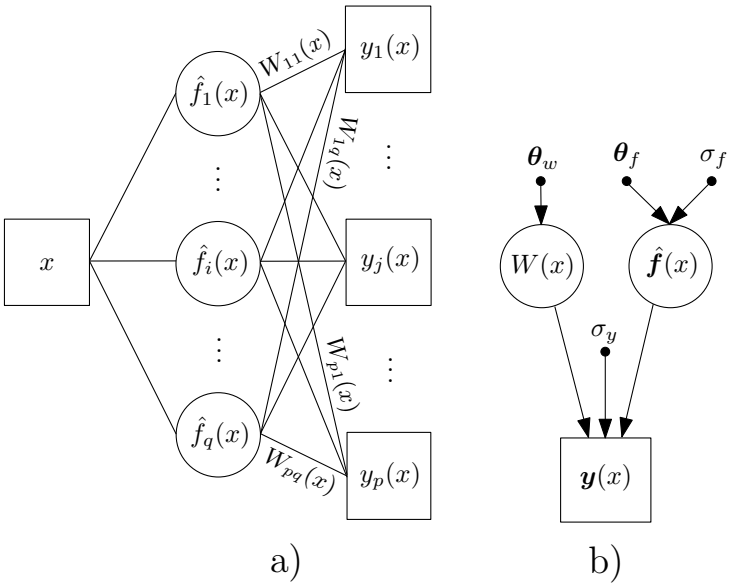

Figure 1: Structure of the Gaussian process regression network. Latent random variables and observables are respectively labelled with circles and squares, except for the weight functions in a). a) This neural network style diagram shows the q components of the vector $\hat{\boldsymbol{f}}$ (GPs with additive noise), and the p components of the vector $\boldsymbol{y}$. The links in the graph, four of which are labelled, are latent random weight *functions*. Every quantity in this graph depends on the input x. This graph emphasises the adaptive nature of this network: links can change strength or even disappear as x changes. b) A directed graphical model showing the generative procedure with relevant variables. Hyperparameters are labelled with dots.

is a Gaussian process with kernel

$$k_{y_i}(x_a, x_w) = \sum_{j=1}^{q} W_{ij}(x_a) k_{\hat{f}_j}(x_a, x_w) W_{ij}(x_w) + \delta_{aw}\sigma_y^2 \,, \,, \tag{2}$$

where $\hat{f}_i(x) = f_i(x) + \sigma_f \epsilon \sim \mathcal{GP}(0, k_{\hat{f}_i})$. Notice: 1) the amplitude of the covariance function, $\sum_{j=1}^{q} W_{ij}(x) W_{ij}(x')$, is non-stationary (input dependent); 2) even if each of the kernels k_{f_j} has different *stationary* length-scales, the mixture of the kernels k_{f_j} is input dependent and so the effective overall length-scale is non-stationary; 3) the kernels k_{f_j} may be entirely different: some may be periodic, others squared exponential, others Brownian motion, and so on. Therefore the overall covariance kernel may be continuously switching between regions of entirely different covariance structures.

We now further explore this property of changing covariance structures, assuming for clarity that $y(x) \in \mathbb{R}^1$. In an adaptive network, a univariate output variable $y(x)$ is an input dependent mixture of functions:

$$y(x) = w_1(x) f_1(x) + w_2(x) f_2(x) + \cdots + w_q(x) f_q(x) \,. \tag{3}$$

If the node functions $f_1(x), \ldots, f_q(x)$ have different covariance functions – or covariance functions with different hyperparameters – then $y(x)$ will switch between different covariance regimes. We can imagine if f_1 has a squared exponential (SE) kernel, f_2 has an Ornstein-Uhlenbeck (OU) kernel, and f_3 has a periodic kernel, $y(x)$ could switch between regions with smooth, OU, and periodic covariance structure, or some mixtures of these structures.

The changes in covariance structure can be made more discrete by warping the weight functions through sigmoid functions:

$$y(x) = \sigma(w_1(x)) f_1(x) + \cdots + \sigma(w_q(x)) f_q(x) \,. \tag{4}$$

If we consider two node functions, and wish σ to act as a switch between the two functions, we can adapt the model to

$$y(x) = \sigma(w(x)) f_1(x) + \sigma(-w(x)) f_2(x) \,. \tag{5}$$

If $w(x), f_1(x), f_2(x)$ are all Gaussian processes (GPs), we can imagine the model accounting for arbitrarily many change-points between f_1 and f_2. Conditioned on $w(x)$, $y(x)$ is a Gaussian process with kernel

$$k(x, x') = \sigma(w(x)) k_1(x, x') \sigma(w(x')) + \sigma(-w(x)) k_2(x, x') \sigma(-w(x')) \,, \tag{6}$$

where k_1 and k_2 are the kernels of f_1 and f_2. A simple special case of the kernel in Eq. (6) can be obtained when $w(x) = a x^\top x + b$, a simple linear function.

Inference and predictions, as well as learning the hyperparameters of $w(x)$, can be performed using variational methods as in Wilson et al. (2012), or following the more recent Nguyen and Bonilla (2013) which specifically focuses on efficient variational methods for Gaussian process regression networks.

Acknowledgements: We thank David Knowles and Zoubin Ghahramani for helpful discussions.

References

Nguyen, T. V. and Bonilla, E. V. (2013). Efficient variational inference for Gaussian process regression networks. In *International Conference on Artificial Intelligence and Statistics*, pages 472–480.

Osborne, M. (2010). *Bayesian Gaussian processes for sequential prediction, optimisation and quadrature*. PhD thesis, University of Oxford.

Saatchi, Y., Turner, R. D., and Rasmussen, C. E. (2010). Gaussian process change point models. In *International Conference on Machine Learning*, pages 927–934.

Wilson, A. G., Knowles, D. A., and Ghahramani, Z. (2011). Gaussian process regression networks. *arXiv preprint arXiv:1110.4411*.

Wilson, A. G., Knowles, D. A., and Ghahramani, Z. (2012). Gaussian process regression networks. In *International Conference on Machine Learning*.