How to choose the covariance for Gaussian process regression independently of the basis

Matthias O. Franz, Peter V. Gehler
mof;pgehler@tuebingen.pg.de
Max-Planck-Institut für biologische Kybernetik, Tübingen, Germany

In Gaussian process regression, both the basis functions and their prior distribution are simultaneously specified by the choice of the covariance function. In certain problems one would like to choose the covariance independently of the basis functions (e. g., in polynomial signal processing or Wiener and Volterra analysis). We propose a solution to this problem that approximates the desired covariance function at a finite set of input points for arbitrary choices of basis functions. Our experiments show that this additional degree of freedom can lead to improved regression performance.

1. Introduction

A Gaussian process is completely specified by its mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$. The choice of the covariance function defines both the space of functions that can be generated by the Gaussian process, and a probability measure on that space. It also determines the function basis in which the regression solutions are expressed since all solutions $f(\mathbf{x})$ are linear combinations of *kernel functions* $k(\cdot, \mathbf{x}_i)$ with some expansion coefficients α_i

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i). \tag{1}$$

Consequently, the choice of the basis and the probability space are *tightly coupled* in the standard Gaussian process formulation.

In certain problems one would like to have the freedom of choosing the covariance function independently of the function basis, i.e., we would like to express our solution in some basis of kernel functions $k(\cdot, \mathbf{x}_i)$, but with a different prior covariance function $k_{\mathcal{GP}}(\cdot, \mathbf{x}_i)$ for the Gaussian process. This situation occurs, for instance, in classical nonlinear system identification where the transfer function $f(\mathbf{x})$ of the unknown system is modeled as a discretized Volterra or Wiener series. A recent study has shown that Volterra and Wiener series can be efficiently estimated by a regression in polynomial kernel functions (Franz & Schölkopf, 2006). Polynomial covariance functions, however, are often not very suitable for describing real-world problems as they imply a high covariance for distant inputs. In most

problems we have the reverse situation, i.e. nearby inputs typically result in similar outputs. This often leads to an inferior prediction performance of polynomial regression as compared to other, more localized covariance functions.

Here, we propose a method for decoupling the choice of the basis and covariance in Gaussian process regression. This is done by using both the weight space view and the function space view of Gaussian processes (Rasmussen & Williams, 2006) to make the contributions of the basis and the probability measure explicit. The proposed approach allows for approximating arbitrary covariance functions on a finite set of input points, without the need of changing the basis functions. As a proof of concept, we show that the typical disadvantages of polynomial covariance functions can be compensated.

2. Decoupling the covariance and basis

Instead of directly specifying the prior covariance $k(\mathbf{x}, \mathbf{x}')$ for solving the regression problem (often referred to as the *function space* view of Gaussian processes, see Rasmussen & Williams, 2006), there exists an alternative derivation, called the *weight-space view*. Here, one assumes that the regression solution $f(\mathbf{x})$ can be represented as weighted sum $f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}$ of m basis functions $\phi(\mathbf{x}) = (\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \ldots, \varphi_m(\mathbf{x}))^\top$, where the weights $\mathbf{w}$ from $\mathbb{R}^m$ are distributed according to $\mathcal{N}(0, \Sigma_w)$. This again defines a covariance function of the form

$$k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^\top \Sigma_w \phi(\mathbf{x}_j). \tag{2}$$

The two alternative views of Gaussian processes give us a handle on how to decouple the covariance and basis: we have to construct a suitable basis $\phi(\mathbf{x})$ from the kernel functions $k(\cdot, \mathbf{x}_i)$ along with a weight covariance Σ_w such that their covariance function $\phi(\mathbf{x}_i)^\top \Sigma_w \phi(\mathbf{x}_j)$ assumes the desired form $k_{\mathcal{GP}}(\mathbf{x}_i, \mathbf{x}_j)$. Of course, we cannot hope to approximate $k_{\mathcal{GP}}(\mathbf{x}_i, \mathbf{x}_j)$ at all possible input pairs, but the computation of the predictive mean and variance requires only evaluating the covariance function at either training or test inputs. Consequently, we have to approximate our desired covariance only at a finite set of input points.

An obvious choice for $\phi(\mathbf{x})$ would be the kernel functions themselves (the *empirical kernel map*, see Schölkopf & Smola, 2002), but here we consider only the *Kernel PCA Map* (Schölkopf & Smola, 2002)

$$\phi(x) = K^{-\frac{1}{2}} (k(x, x_1), k(x, x_2), \ldots, k(x, x_n))^\top \tag{3}$$

which usually leads to a better conditioned regression problems in other contexts. Having specified our basis, we have to find a suitable Σ_w to approximate $k_{\mathcal{GP}}(\mathbf{x}_i, \mathbf{x}_j)$ on a finite set $\mathcal{S} = \{x_1, \ldots, x_p\}$ of input points. Formally, we have a set of p^2 linear equations

$$k_{\mathcal{GP}}(x_i, x_j) = \phi(x_i)^\top \Sigma_w \phi(x_j) \; \forall x_i, x_j \in \mathcal{S} \tag{4}$$

which, in general, cannot be solved exactly. An approximate solution is given by

$$\Sigma_w = ([\phi(x_1), \ldots, \phi(x_p)]^\top)^+ K_{\mathcal{GP}}(\mathcal{S}) [\phi(x_1), \ldots, \phi(x_p)]^+, \tag{5}$$

dataset		k_{gauss}	k_{ihp}	$\mathcal{S}_{\text{train}}$	$\mathcal{S}_{\text{train}} \cup \mathcal{S}_{\text{test}}$
Boston housing	train	3.58	5.11	3.58	3.59
	test	8.36	9.79	9.53	8.3
KIN40K	train	0.59	9.84	0.59	2.57
	test	10.41	21.07	114	13.76
Stereo	train	1.80	2.55	1.80	1.80
	test	3.26	3.38	3.30	3.25

Table 1: Averaged mean squared error on training and test set (10 folds for Boston, 5 for KIN40K, 3 for Stereo). For the inhomogeneous polynomial kernel with approximated Gaussian covariance, Σ_w is computed either on the training inputs only ($\mathcal{S}_{\text{train}}$), or on both training and test inputs ($\mathcal{S}_{\text{train}} \cup \mathcal{S}_{\text{test}}$).

where $K_{\mathcal{GP}}(\mathcal{S}) = (k_{\mathcal{GP}}(x_i, x_j))_{ij}\ \forall x_i, x_j \in \mathcal{S}$ and A^+ denotes the Moore-Penrose pseudoinverse of A. The approximation accuracy clearly depends on the choice of the set $\mathcal{S}$ which describes the input region in which one wants to mimic the covariance function $C_{\mathcal{GP}}$. For prediction purposes it is at hand to use the training and, if available, the test inputs in the calculation of Σ_w. Note that no output values enter Equation (5), so we can use any possible input set from the region of interest, not necessarily from the training or test set.

3. Experiments

The decoupling approach was evaluated on three regression datasets: Boston Housing (Harrison & Rubinfeld, 1978), KIN40K (Schwaighofer & Tresp, 2003), and Stereopsis (Sinz, Quiñonero-Candela, Bakır, Rasmussen, & Franz, 2004). The application scenario for our decoupling technique are cases where the regression performance of the chosen basis is inferior to that of another Gaussian process which defines the target covariance. In these cases, a better approximation to the target covariance should also result in an increased performance if the approximation range is chosen to be sufficiently representative.

In a baseline experiment, we used standard Gaussian process regression with the Gaussian kernel $k_{\text{gauss}}(x, x') = \exp(-\|x - x'\|^2)$ and the inhomogeneous polynomial kernel $k_{\text{ihp}}(x, x') = (1 + x^\top x')^p$. Model selection was done by maximizing the log-likelihood. The results of the baseline experiment are shown in the 3rd and 4th column of Table 1. The Gaussian covariance leads to a better regression performance on all three datasets although on the Stereopsis dataset the difference is very small.

To test our approach, we chose the Gaussian covariance $k_{\text{gauss}}(x, x')$ as our target covariance as $k_{\mathcal{GP}}(x, x')$, and the KPCA map computed for k_{ihp} as our basis. Calculation of Σ_w was done either on the training inputs only, or on both training and test inputs. Due to the required expensive matrix inverse we did not use all but only a subset of 2000 test points from the KIN40K dataset in the calculation of Σ_w. The results are summarized in the 5th and 6th column of Table 1. The model

parameters were again selected by maximizing the standard log-likelihood criterion. The choice of the polynomial degree turned out to be sometimes problematic since the log-likelihood as a function of polynomial degree tended to be very flat. Therefore, we chose instead the polynomial degree which minimises the 2-norm between the entries of the actual and the desired covariance matrix.

4. Discussion

Using our technique one can approximate any given covariance function in a Gaussian process on a finite set of points, without having to change the basis functions of the regression. This technique can be helpful in all settings where one is restricted to a specific set of basis functions as, e.g., in polynomial regression, or in Wiener and Volterra analysis.

When only the training inputs were used for approximating the desired covariance, improvements over standard regression were only small, or - in the case of KIN40K - performance severely degraded because the training inputs are not sufficiently representative for the test input range. Since the training error was the same as in the original Gaussian covariance, this is a clear indicator of overfitting to the training data. However, the results consistently improved on all datasets when the approximation was computed on both training and test inputs, so this should be the method of choice when the performance of polynomial regression is to be improved.

References

Franz, M. O., & Schölkopf, B. (2006). A unifying view of Wiener and Volterra theory and polynomial kernel regression. *Neural Computation*, in press.

Harrison, D., & Rubinfeld, D. (1978). Hedonic prices and the demand for clean air. *J. Environ. Economics & Management*, **5**, 81 – 102. Data available from http://lib.stat.cmu.edu/datasets/boston.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. Cambridge, MA: MIT Press.

Schwaighofer, A., & Tresp, V. (2003). Transductive and inductive methods for approximate Gaussian process regression. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in Neural Information Processing Systems*, Vol. 15, pp. 953 – 960 Cambridge, MA. MIT Press.

Sinz, F., Quiñonero-Candela, J., Bakır, G. H., Rasmussen, C. E., & Franz, M. O. (2004). Learning Depth From Stereo. In C. E. Rasmussen, H. H. Bülthoff, M. A. Giese, & B. Schölkopf (Eds.), *Pattern Recognition, Proc. 26th DAGM Symposium*, Vol. 3175 of *LNCS*, pp. 245 – 252 Berlin. Springer.