A Sparse Covariance Function for Exact Gaussian Process Inference in Large Datasets

Arman Melkumyan
Australian Centre for Field Robotics
The University of Sydney
NSW 2006, Australia
a.melkumyan@acfr.usyd.edu.au

Fabio Ramos
Australian Centre for Field Robotics
The University of Sydney
NSW 2006, Australia
f.ramos@acfr.usyd.edu.au

Abstract

Despite the success of Gaussian processes (GPs) in modelling spatial stochastic processes, dealing with large datasets is still challenging. The problem arises by the need to invert a potentially large covariance matrix during inference. In this paper we address the complexity problem by constructing a new stationary covariance function (Mercer kernel) that naturally provides a sparse covariance matrix. The sparseness of the matrix is defined by hyper-parameters optimised during learning. The new covariance function enables exact GP inference and performs comparatively to the squared-exponential one, at a lower computational cost. This allows the application of GPs to large-scale problems such as ore grade prediction in mining or 3D surface modelling. Experiments show that using the proposed covariance function, very sparse covariance matrices are normally obtained which can be effectively used for faster inference and less memory usage.

1 Introduction

Gaussian processes (GPs) are a useful and powerful tool for regression in supervised machine learning [Rasmussen and Williams, 2006]. The range of applications includes geophysics, mining, hydrology, reservoir engineering and robotics. Despite its increasing popularity, modelling large-scale spatial stochastic processes is still challenging. The difficulty comes from the fact that inference in GPs is usually computationally expensive due to the need to invert a potentially large covariance matrix during inference time, which has $\mathcal{O}\left(N^3\right)$ cost. For problems with thousands of observations, exact inference in normal GPs is intractable and approximation algorithms are required.

Most of the approximation algorithms employ a subset of points to approximate the posterior distribution at a new point given the training data and hyper-parameters. These approximations rely on heuristics to select the subset of points [Lawrence *et al.*, 2003; Seeger *et al.*, 2003], or use pseudo targets obtained during the optimisation of the log-marginal likelihood of the model [Snelson and Ghahramani, 2006].

In this work, we address the complexity problem differently. Instead of relying on sparse GP approximations, we propose a new covariance function which provides intrinsically sparse covariance matrices. This allows exact inference in GPs using conventional methods. As the new sparse covariance function can be multiplied by any other valid covariance function and the result is a sparse covariance matrix, a lot of flexibility is given to practitioners to accurately model their problems while still preserving sparseness properties. We call the GPs constructed using our sparse covariance function Exact Sparse Gaussian Processes (ESGPs). The main idea behind is the formulation of a valid and smooth covariance function whose output equals to zero whenever the distance between input observations is larger than a hyper-parameter. As with other hyper-parameters, this can be estimated by maximising the marginal likelihood to better model the properties of the data such as smoothness, characteristic length-scale, and noise. Additionally, the proposed covariance function in much resembles the popular squared exponential in terms of smoothness, being four times continuously differentiable. We empirically compare ESGP with local approximation techniques and demonstrate how other covariance functions can be integrated in the same framework. Our method results in very sparse covariance matrices (up to 90% of the elements are zeros in in-ground grade estimation problems) which requires significantly less memory while providing similar performance.

This paper is organised as follows. In Section 2 we review the basics of GP regression and introduce notation. Section 3 summarises previous work on approximate inference with GPs. Section 4 presents our new intrinsically sparse covariance function and its main properties. We evaluate the framework providing experimental results in both artificial and real data in Section 6. Finally, Section 7 concludes the paper and discusses further developments.

2 Gaussian Processes

In this section we briefly review Gaussian Processes for regression and introduce notation. We consider the supervised learning problem where given a training set $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$ consisting of N input points $\mathbf{x}_i \in \mathbb{R}^D$ and the corresponding outputs $y_i \in \mathbb{R}$ the objective is to compute the predictive distribution $f\left(\mathbf{x}_*\right)$ at a new test point $\mathbf{x}_*$. A Gaussian process model places a multivariate Gaussian distribution over the space of function variables $f(\mathbf{x})$ mapping input to output spaces. The model is specified by defining a mean function

$m(\mathbf{x})$ and the covariance function $k(\mathbf{x}, \mathbf{x}')$ resulting in the Gaussian process written as

$$f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right).$$

Denoting groups of these points as $(X, \mathbf{f}, \mathbf{y}) = (\{\mathbf{x}_i\}, \{f_i\}, \{y_i\})_{i=1}^{N}$ for the training set and $(X_*, \mathbf{f}_*, \mathbf{y}_*) = (\{\mathbf{x}_{*,i}\}, \{f_{*,i}\}, \{y_{*,i}\})_{i=1}^{N}$ for the testing points, the joint Gaussian distribution with $m(\mathbf{x}) = 0$ is

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right), \quad (1)$$

where $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, and K is used to denote the covariance matrix computed between all points in the set. If we assume observations with Gaussian noise ϵ and variance σ^2 such that $y = f(\mathbf{x}) + \epsilon$, the joint distribution becomes

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma^2 I & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right). \quad (2)$$

A popular choice for the covariance function is the *squared exponential* used in this paper for comparisons in the experiment section:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T M (\mathbf{x} - \mathbf{x}')\right) \quad (3)$$

with $M = \text{diag}(\mathbf{l})^{-2}$ where $\mathbf{l}$ is a vector of positive numbers representing the length-scales in each dimension.

2.1 Inference for New Points

By conditioning on the observed training points, the predictive distribution can be obtained as

$$p\left(f_* \mid X_*, X, \mathbf{y}\right) = \mathcal{N}\left(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*\right), \quad (4)$$

where

$$\begin{aligned} \boldsymbol{\mu}_* &= K\left(X_*, X\right)\left[K\left(X, X\right) + \sigma^2 I\right]^{-1} \mathbf{y} \\ \boldsymbol{\Sigma}_* &= K\left(X_*, X_*\right) - K\left(X_*, X\right)\left[K\left(X, X\right) + \sigma^2 I\right]^{-1} \\ &\quad K\left(X, X_*\right) + \sigma^2 I. \end{aligned} \quad (5)$$

From Equation 5, it can be observed that the predictive mean is a linear combination of N kernel functions each centred on a training point, $\boldsymbol{\mu}_* = \sum_{i=1}^{N} \alpha_i k\left(\mathbf{x}_i, \mathbf{x}_*\right)$, where $\boldsymbol{\alpha} = \left(K\left(X, X\right) + \sigma^2 I\right)^{-1} \mathbf{y}$. A GP is also a *best unbiased linear estimator* [Cressie, 1993; Kitanidis, 1997] in the mean squared error sense. During inference, most of the computational cost takes place while computing the inversion in Equation 5, which is $\mathcal{O}\left(N^3\right)$ if implemented naïvely.

2.2 Learning Hyper-Parameters

Commonly, the covariance function $k(\mathbf{x}, \mathbf{x}')$ is parametrised by set of hyper-parameters $\boldsymbol{\theta}$, and we can write $k\left(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}\right)$. These parameters allow for more flexibility in modelling the properties of the data. Thus, learning a GP model is equivalent to determining the hyper-parameters of the covariance function from some training dataset. In a Bayesian framework this can be performed by maximising the log of the marginal likelihood w.r.t. $\boldsymbol{\theta}$:

$$\log p\left(\mathbf{y} \mid X, \theta\right) = -\frac{1}{2}\mathbf{y}^T K_y^{-1} \mathbf{y} - \frac{1}{2}\log\left|K_y\right| - \frac{N}{2}\log 2\pi \quad (6)$$

where $K_y = K(X, X) + \sigma^2 I$ is the covariance matrix for the targets $\mathbf{y}$. The marginal likelihood has three terms (from left to right), the first accounts for the data fit; the second is a complexity penalty term (encoding the Occam's Razor principle) and the last is a normalisation constant.

Eq. (6) is a non-convex function on the hyper-parameters $\boldsymbol{\theta}$ and therefore only local maxima can be obtained. In practice, this is not a major issue since good local maxima can be obtained with gradient descent techniques by using multiple starting points. However, this requires the computation of partial derivatives resulting in:

$$\frac{\partial}{\partial \theta_j} \log p\left(\mathbf{y} \mid X, \boldsymbol{\theta}\right) = \frac{1}{2}\mathbf{y}^T K^{-1} \frac{\partial K}{\partial \theta_j} K^{-1} \mathbf{y} - \frac{1}{2}\text{tr}\left(K^{-1}\frac{\partial K}{\partial \theta}\right). \quad (7)$$

Note that this expression requires the computation of partial derivatives of the covariance function w.r.t $\boldsymbol{\theta}$.

3 Related Work

Recently, there have been several methods proposed to tackle the problem of GP inference in large datasets. However, most of these approaches rely on approximation techniques. A common and simple procedure is to select a subset of data points and perform inference using only these points. This is equivalent to ignoring part of the data which makes the selection of the subset very important. In [Lawrence *et al.*, 2003], the selection of points is based on the differential entropy. Similarly, [Seeger *et al.*, 2003] suggest the use of another information theory quantity, the information gain.

Another interesting procedure is to select a subset of data points to act as an *inducing* set and project this set up to all the data points available. This is known as sparse GP approximation [Williams and Seeger, 2001; Smola and Bartlett, 2001; Candela and Rasmussen, 2005] which usually performs better than simply selecting a subset of data points. However, the definition of the inducing set is difficult and can involve non-convex optimisations [Snelson and Ghahramani, 2006].

Local methods have been applied in geostatistics for a long time [Wackernagel, 2003]. The idea is to perform inference by evaluating the covariance function only at points in the neighbourhood of a query point. This method can be effective but the definition of the neighbourhood is crucial. Our method is inspired by this approach but rather than defining the neighbourhood manually, we obtain it automatically during learning. An interesting idea on combining local and global methods (such as the sparse Gaussian process) was proposed in [Snelson and Ghahramani, 2007]. We compare our method to theirs in Section 6.

This work differs from other studies by not addressing the GP inference problem through an approximation technique. Rather, it proposes a new covariance function that naturally generates sparse covariance matrices. This idea was used in [Wendland, 2005] with piecewise polynomials but extensions to multiple dimensions is difficult due to the need to guarantee positive definiteness. A similar formulation to ours was proposed in [Storkey, 1999]. However, there is no hyperparameter learning and the main properties are not analysed.

To the best of our knowledge, this work is the first to demonstrate with real examples how the complexity problem can be addressed through the construction of a new sparse covariance function allowing for exact GP inference in large datasets.

4 Exactly Sparse Gaussian Processes

For very large datasets, the inversion or even storage of a full matrix $K(X,X)+\sigma^2 I$ can be prohibitive. In geology problems for example, it is not uncommon to have datasets with 100K points or more. To deal with such large problems while still being able to perform exact inference in the GP model, we develop the covariance function below. First, note that the mean prediction in Eq. 5 can be rewritten as a linear combination of N evaluations of the covariance function, each one centred on a training point, $\boldsymbol{\mu}_* = \sum_{i=1}^{N}\alpha_i k\left(\mathbf{x}_*, \mathbf{x}_i\right)$, where $\boldsymbol{\alpha} = \left(K\left(X,X\right)+\sigma^2 I\right)^{-1}\mathbf{y}$. To avoid the inversion of the full matrix, we can instead develop a covariance function whose output vanishes outside some region $\mathcal{R}$, so that $k(\mathbf{x}_*, \mathbf{x}_i) = 0$ when $\mathbf{x}_i$ is outside a region $\mathcal{R}$. In this way, only a subset of $\boldsymbol{\alpha}$ would need to be computed which effectively means that only few columns of $\left(K\left(X,X\right)+\sigma^2 I\right)^{-1}$ need to be computed, significantly reducing the computational and storage costs as $\mathcal{R}$ diminishes. As we shall see, the region can be specified automatically during learning.

4.1 Intrinsically Sparse Covariance Function

The covariance function we are looking for must vanish out of some finite region $\mathcal{R}$ for exact sparse GP regression. It must produce smooth curves but it should not be infinitely differentiable so that it can be applicable to problems with some discontinuities. For our derivation, the function $g\left(x\right) = \cos^2\left(\pi x\right) H\left(0.5-|x|\right)$ was chosen, which due to $\cos^2\left(\pi x\right) = \left(\cos\left(2\pi x\right)+1\right)/2$ is actually the cosine function shifted up, normalised and set to zero out of the interval $x \in \left(-0.5,\, 0.5\right)$. The cosine function was selected as the basis function due to the following reasons: 1) it is analytically well tractable; 2) integrals with finite limits containing combinations of its basic form can be calculated in closed form; and 3) the cosine function usually provides good approximations for different functions, being the core element for Fourier analysis. Here and afterwards $H\left(\cdot\right)$ represents the Heaviside unit step function. As it stands, the chosen basis function $g\left(x\right)$ is smooth on the whole real axis, vanishes out of the interval $x \in \left(-0.5,\, 0.5\right)$ and has discontinuities in the second derivative. To derive a valid covariance function we conduct calculations analogous to presented in [Rasmussen and Williams, 2006]. Using the transfer function $h\left(x;u\right) = g\left(x-u\right)$ the following 1D covariance function is obtained:

$$k_1\left(x,x'\right) = \sigma\int_{-\infty}^{\infty} h\left(\frac{x}{l};u\right) h\left(\frac{x'}{l};u\right) du. \tag{8}$$

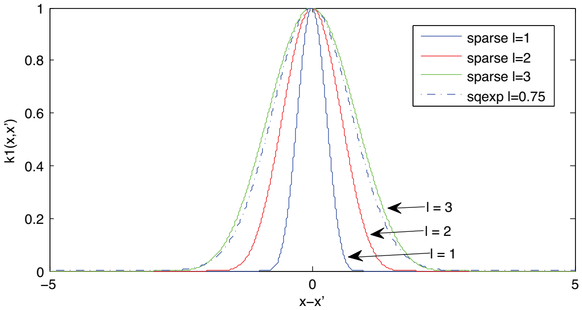


Figure 1: Plot showing the output of the covariance function for different values of Δx.

Due to the chosen form of the basis functions, the integral in Eq. (8) can be analytically evaluated (see Appendix A for details) to result in:

$$k_1\left(x,x';l,\sigma_0\right) = \begin{cases} \sigma_0\left[\frac{2+\cos\left(2\pi\frac{d}{l}\right)}{3}\left(1-\frac{d}{l}\right)+\frac{1}{2\pi}\sin\left(2\pi\frac{d}{l}\right)\right] & \text{if } d<l \\ 0 & \text{if } d\geq l \end{cases} \tag{9}$$

where $\sigma_0 > 0$ is a constant coefficient, $l > 0$ is a given scale and d is the distance between the points:

$$d = \left|x-x'\right|. \tag{10}$$

From Eq. (8) it follows that for any points x_i and any real numbers a_i where $i = 1, 2, \ldots, n$ the inequality $\sum_{i,j=1}^{n} a_i a_j k_1\left(x_i,x_j\right) = \sigma\int_{-\infty}^{\infty}\left(\sum_{i=1}^{n} a_i h\left(\frac{x_i}{l};u\right)\right)^2 du \geq 0$ holds, so that the constructed covariance function is positive semi-definite.

Based on Eqs. (9)-(10) we calculate that

$$k|_{d=l} = \left.\frac{\partial k}{\partial d}\right|_{d=l} = \left.\frac{\partial^2 k}{\partial d^2}\right|_{d=l} = \left.\frac{\partial^3 k}{\partial d^3}\right|_{d=l} = \left.\frac{\partial^4 k}{\partial d^4}\right|_{d=l} = 0, \quad \left.\frac{\partial^5 k}{\partial d^5}\right|_{d=l} = -4\pi^4 \neq 0 \tag{11}$$

which shows that the covariance function k_1 is continuous and has continuous 4th derivative at $d = l$.

The function $k_1\left(\Delta x, 0; l, 1\right)$ is compared with squared exponential in Figure 1. Note that it follows the squared exponential covariance function closely but vanishes when $|\Delta x| \geq l$.

4.2 Extending to Multiple Dimensions

This covariance function can be extended to multiple dimensions in the following ways:

1. Using direct products for all axes:

$$k_M^{(1)}\left(\mathbf{x},\mathbf{x}';\mathbf{l},\sigma_0\right) = \sigma_0\prod_{i=1}^{D} k_1\left(x_i,x_i';l_i,1\right) \tag{12}$$

where D is the dimensionality of the points and $\mathbf{l}$ is the vector of the characteristic lengths, $\mathbf{l} = \left(l_1, l_2, \ldots, l_D\right)^T$.

2. Using Mahalanobis distance:

$$k_M^{(2)}(r;\sigma_0,\Omega) = \begin{cases} \sigma_0 \left[\frac{2+\cos(2\pi r)}{3}(1-r) + \frac{1}{2\pi}\sin(2\pi r)\right] & \text{if } r < 1 \\ 0 & \text{if } r \geq 1 \end{cases} \tag{13}$$

where $\sigma_0 > 0$, Ω is positive semi-definite and

$$r = \sqrt{(\mathbf{x}-\mathbf{x}')^T \,\Omega\, (\mathbf{x}-\mathbf{x}')}, \quad \Omega \geq 0. \tag{14}$$

After this point we will frequently use the short notation k_M for the function $k_M^{(2)}(r;\sigma_0,\Omega)$.

4.3 Important properties of the new covariance function

The developed multi-dimensional covariance function in both forms $k_M^{(1)}$ and $k_M^{(2)}$ has the following remarkable properties:

1. It vanishes out of some finite region $\mathcal{R}$:
$$\mathcal{R}_1 = \left\{r \in \mathbb{R}^D : k_M^{(1)}(r) \neq 0\right\} \tag{15}$$
$$\mathcal{R}_2 = \left\{r \in \mathbb{R}^D : k_M^{(2)}(r) \neq 0\right\} \tag{16}$$
Sizes of the regions $\mathcal{R}_1$ and $\mathcal{R}_2$ can be controlled via the characteristic lengths l_i. Moreover, these sizes can be learnt from data by maximising the marginal likelihood as common in the GP framework.
2. All the derivatives up to (and including) the fourth order derivative are continuous, which guarantees mean square differentiability up to the corresponding order of the sample curves in GPs. There are discontinuities for the fifth order gradient.
3. The region $\mathcal{R}_1$ is a D dimensional rectangle and the region $\mathcal{R}_2$ is a D dimensional ellipsoid.
4. In the case of 1 dimension $k_M^{(1)}$ and $k_M^{(2)}$ become identical.
5. The covariance function is anisotropic, i.e. has different inner properties for different dimensions.
6. This covariance function leads to sparse covariance matrices and allows GP inference in large datasets without the need for approximations.

5 Partial Derivatives for Learning

Learning the GP requires the computation of the covariance function partial derivatives w.r.t. the hyper-parameters (Eq. (7)). Based on Eqs.(9), (12), the following expressions for the partial derivatives of $k_M^{(1)}(\mathbf{x},\mathbf{x}';\mathbf{l},\sigma_0)$ can be calculated:

$$\frac{\partial k_M^{(1)}}{\partial \sigma_0} = \frac{1}{\sigma_0} k_M^{(1)} \tag{17}$$

$$\frac{\partial k_M^{(1)}(x_i,x_i';l_i,\sigma_0)}{\partial l_i} = \frac{4\sigma_0}{3}\frac{k_M^{(1)}(x_i,x_i';l_i,\sigma_0)}{k_1}\frac{d_i}{l_i^2} \times \left[\pi\left(1-\frac{d_i}{l_i}\right)\cos\left(\pi\frac{d_i}{l_i}\right) + \sin\left(\pi\frac{d_i}{l_i}\right)\right]\sin\left(\pi\frac{d_i}{l_i}\right) \tag{18}$$

where $i = 1, 2, ..., D$.

For the second case, if Ω is diagonal and positive definite, it can be expressed via the characteristic lengths as follows:

$$\Omega = \operatorname{diag}\left(\frac{1}{l_1^2}, \frac{1}{l_2^2}, ..., \frac{1}{l_D^2}\right). \tag{19}$$

From Eqs. (14), (19) it follows that

$$r = \sqrt{\sum\nolimits_{k=1}^{D}\left(\frac{x_k - x_k'}{l_k}\right)^2}. \tag{20}$$

Based on Eqs. (13), (19)-(20) the following gradient components of this multi-dimensional covariance function k_M can be obtained:

$$\frac{\partial k_M}{\partial \sigma_0} = \frac{2+\cos(2\pi r)}{3}(1-r) + \frac{1}{2\pi}\sin(2\pi r), \text{ if } 0 \leq r < 1 \tag{21}$$

$$\frac{\partial k_M}{\partial l_j} = \frac{4\sigma_0}{3}\left[\pi(1-r)\cos(\pi r) + \sin(\pi r)\right] \times \frac{\sin(\pi r)}{r}\frac{1}{l_j}\left(\frac{x_j - x_j'}{l_j}\right)^2, \text{if } 0 < r < 1. \tag{22}$$

$$\operatorname{grad} k_M = 0, \text{ if } r \geq 1. \tag{23}$$

In Eq. (22), r is in the denominator, so that direct calculations cannot be carried out using Eq. (22) when $r = 0$. However, using the equality

$$\lim_{r\to 0}\frac{\sin(\mu r)}{r} = \mu \tag{24}$$

one can directly show that

$$\lim_{r\to 0}\frac{\partial k_M(r;\sigma_0,\Omega)}{\partial l_j} = \lim_{r\to 0}\frac{4\sigma_0}{3}\pi^2\frac{1}{l_j}\left(\frac{x_j - x_j'}{l_j}\right)^2 = 0. \tag{25}$$

Based on Eq. (25) it must be taken directly

$$\left.\frac{\partial k_M}{\partial l_j}\right|_{r=0} = 0, \; j = 1, 2, ..., D. \tag{26}$$

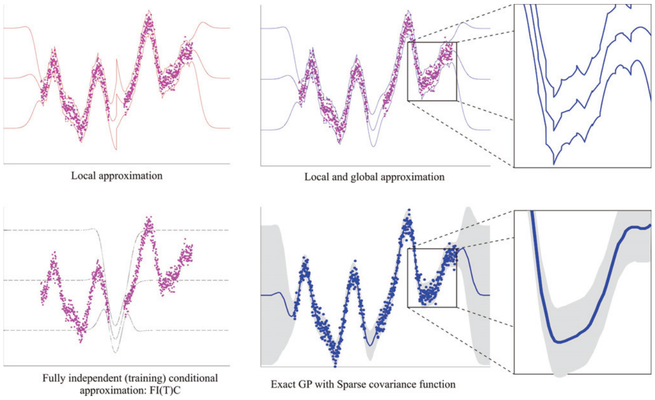
Eqs. (21)-(23), (26) fully define the gradient of the new covariance function k_M at every point and can be directly used in the learning procedure.

6 Experiments

This section provides empirical comparisons between exact sparse GP, conventional GP with squared exponential and approximation procedures.

6.1 Artificial Dataset

In this experiment we compare the exact GP with the proposed covariance function against the approach proposed in [Snelson and Ghahramani, 2007]. The data is essentially the same as presented in the experiment section in [Snelson and Ghahramani, 2007]. As can be observed in Figure 2, the



Local approximation

Local and global approximation

Fully independent (training) conditional approximation: FI(T)C

Exact GP with Sparse covariance function

Figure 2: Comparison between exact sparse GP and the local and global approximation. FI(T)C stands for Fully independent (training) conditional approximation. Details can be found in [Snelson and Ghahramani, 2007]. Note that exact sparse GP provides a much smoother curve.

sparse covariance function provides a much smoother prediction for the underlying function than the combination of local and global approximations. This example shows qualitatively that in some situations approximation methods can lead to discontinuities. The same does not occur in the exact sparse GP inference.

6.2 Rainfall Dataset

In this experiment we compare the exact sparse GP with the exact GP with the squared exponential covariance function and the covariance function obtained by the multiplication of both of them. The dataset used is a popular dataset in geostatistics for comparing inference procedures and is known as the Spatial Interpolation Comparison dataset [Dubois *et al.*, 2003](SIC) [1]. The dataset consists of 467 points measuring rainfall in 2D space. We divide these points into two sets, inference and testing. The inference set contains the points used to perform inference on the testing points. For each case the experiment is repeated 1500 times with randomly selected inference and testing sets. Figure 3 shows the normalised squared error for the different covariance functions and the standard deviation (one sigma for each part of the bar) as a function of the number of inference points. As the number of inference points increases, so does the size of the covariance matrix. The results demonstrate that very similar errors are obtained for the different covariances. However, the sparse covariance function produces sparse matrices thus requiring much less floating point operations. Figure 4 shows the percentage of zeros in the covariance matrix as a function of the number of inference points. As can be observed, the percentage of zeros grows quickly as more inference points are added and it reaches its limit around 100 inference points. Although the percentage of zeros reaches its limit, the number of zeros in the covariance matrix continues to increase because the size of the covariance matrix increases with the number of inference points. Also worth noticing is the performance of the multiplication between the two covariance functions. The error is essentially the same as for the sparse covariance function alone but the percentage of zeros is significantly smaller. This example demonstrates the benefits of the proposed approach in reducing storage and number of operations for similar accuracy.

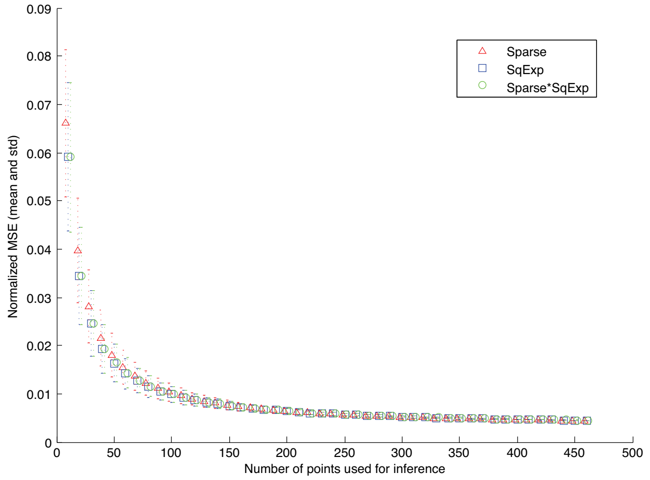


Figure 3: Normalised Mean Square Error for the SIC dataset. The error is essentially the same for both covariance functions, with the exact sparse performing slightly worse with fewer inference points but similar with more inference points (at a much lower computation cost).

6.3 Iron Ore Dataset

In this dataset the goal is to estimate iron ore grade in 3D space over a region of 2.5 cubic kilometres. The dataset is from an open pit iron mine in Western Australia. About 17K samples were collected and the iron concentration measured with X-Ray systems. We divide the 17K dataset points into inference and testing sets. The inference set is taken arbitrarily from the dataset and from the remaining points the testing points are arbitrarily chosen. The experiments are repeated 500 times. Figure 5 shows the normalised mean squared error and the standard deviation (one sigma for each part of the bar) in the cases of squared exponential, sparse covariance functions and their product. The results demonstrate that all the three lead to similar errors with the sparse covariance function performing slightly better with the increase of the number of inference points. Figure 6 shows that although they result in similar errors, the sparse and the product lead to about 48% of zeros in the covariance matrix, which is 120K to 12M cells exactly equal to zero when the number of inference points varies from 500 to 5000. This example demonstrates that the proposed method provides greater savings for bigger datasets.

6.4 Speed Comparison

This experiment demonstrates the computational gains in using the proposed sparse covariance function. Synthetic datasets containing 1000, 2000, 3000 and 4000 points were generated by sampling a polynomial function with white

[1] The SIC dataset can be downloaded at: http://www.ai-geostats.org

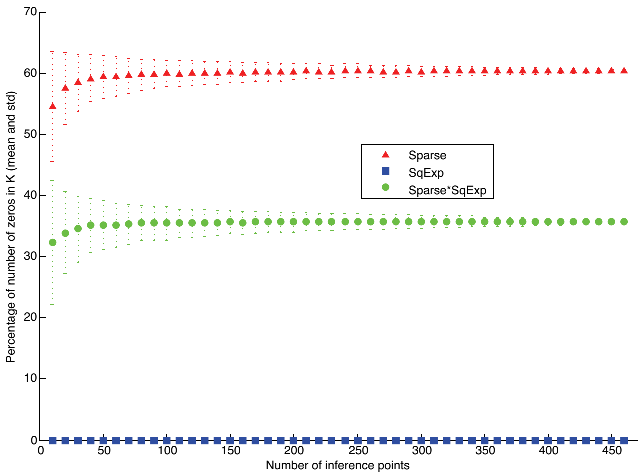


Figure 4: Percentage of zeros in the covariance matrix as a function of the number of inference points.

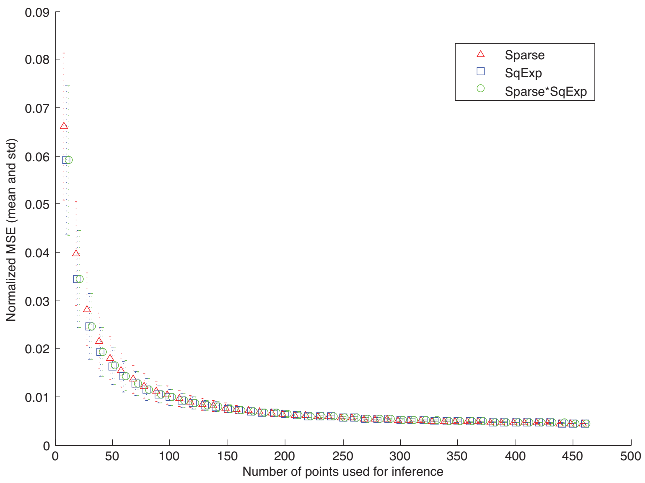


Figure 5: Normalised Mean Square Error for the iron ore dataset. The performance of the sparse covariance function is equivalent to squared exponential. Due to the computational cost using the squared exponential, we stop the experiment with 5000 inference points although the sparse covariance function could accommodate all the 17K points.

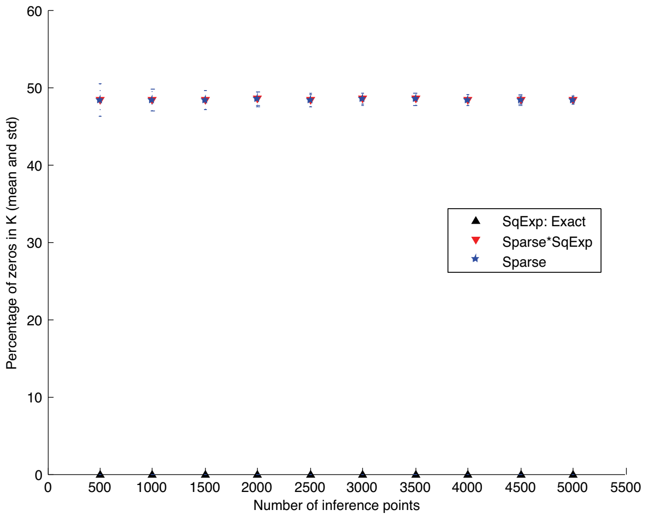


Figure 6: Percentage of zeros for the iron ore grade estimation problem.

noise. We compare the speed of a GP with the sparse covariance function to a GP with the squared exponential covariance function for different length scales and corresponding number of non-zero elements in the covariance matrix. The results are presented in Figure 7. The code is implemented in Matlab and uses the sparse matrix implementation package provided. Further gains could be obtained in more efficient sparse matrix packages. As the number of points in the datasets increases, the speed up becomes more evident. With 4000 points, the sparse covariance function is faster than the squared exponential for up to 70% of non-zeros elements in the covariance matrix. After this point, the computational cost of the sparse matrix implementation becomes dominant. As in general the sparse covariance function provides covariance matrices much sparser, speed gains can be quite substantial (in addition to storage gains).

7 Conclusions

This paper proposed a new covariance function constructed upon the cosine function for analytical tractability that naturally provides sparse covariance matrices. The sparseness of the data is controlled by a hyper-parameter that can be learnt from data. The sparse covariance function enables exact inference in GPs even for large datasets, providing both storage and computational benefits. Although the main focus of this paper was on GPs, it is important to emphasise that the covariance function proposed is also a Mercer kernel and therefore can be applied to kernel machines such as support vector machines, kernel principal component analysis and others [Schölkopf and Smola, 2002]. The use of the sparse covariance function in other kernel methods is objective of our future work.

Acknowledgements

We thank Edward Snelson for providing the pictures for the comparison with local and global sparse Gaussian process approximation. This work has been supported by the Rio Tinto

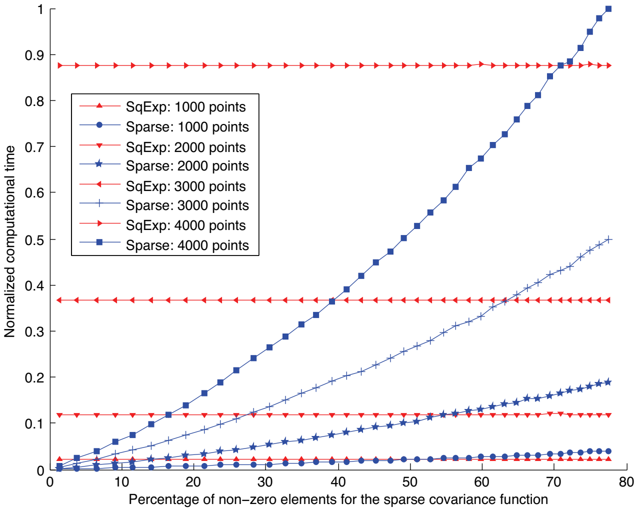


Figure 7: Normalised computational time versus number of non-zero elements for the sparse covariance function in datasets of different sizes. The performance of the (non-sparse) squared exponential covariance function is also included for comparison. Note that the computational gains increase with the size of the datasets.

Centre for Mine Automation and the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

References

[Candela and Rasmussen, 2005] J. Quiñonero Candela and C. E. Rasmussen. A unifying view of sparse Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

[Cressie, 1993] N. Cressie. *Statistics for Spatial Data*. Wiley, 1993.

[Dubois *et al.*, 2003] G. Dubois, J. Malczewski, and M. De Cort. Mapping radioactivity in the environment. spatial interpolation comparison 1997. In *Office for Official Publications of the European Communities*, Luxembourg, 2003.

[Kitanidis, 1997] P. K. Kitanidis. *Introdcution to Geostatistics: Applications in Hydrogeology*. Cambridge University Press, 1997.

[Lawrence *et al.*, 2003] N. Lawrence, M. Seeger, and R. Herbrich. Fast sparse gaussian process methods: The information vector machine. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 625–632. MIT Press, 2003.

[Rasmussen and Williams, 2006] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[Schölkopf and Smola, 2002] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.

[Seeger *et al.*, 2003] M. Seeger, C. K. I. Williams, and N. Lawrence. Fast forward selection to speed up sparse gaussian process regression. In *AISTATS*, 2003.

[Smola and Bartlett, 2001] A. Smola and P. Bartlett. Sparse greedy gaussian process regression. In *Advances in Neural Information Processing Systems 13*, pages 619–625. MIT Press, 2001.

[Snelson and Ghahramani, 2006] E. Snelson and Z. Ghahramani. Sparse gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems 18*, pages 1257–1264. MIT press, 2006.

[Snelson and Ghahramani, 2007] E. Snelson and Z. Ghahramani. Local and global sparse Gaussian process approximations. In *AISTATS*, 2007.

[Storkey, 1999] A. J. Storkey. Truncated covariance matrices and toeplitz methods in gaussian processes. In *9th International Conference on Artificial Neural Networks*, 1999.

[Wackernagel, 2003] H. Wackernagel. *Multivariate Geostatistics*. Springer, 2003.

[Wendland, 2005] H. Wendland. *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2005.

[Williams and Seeger, 2001] C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems 13*, pages 682–688. MIT Press, 2001.

A Detailed Derivation

The covariance function is constructed by evaluating the integral

$$k_1\left(x,x'\right) = \sigma \int_{-\infty}^{\infty} g\left(\frac{x}{l}-u\right) g\left(\frac{x'}{l}-u\right) du \qquad (27)$$

where

$$g\left(x\right) = \cos^2\left(\pi x\right) H\left(0.5-|x|\right) \qquad (28)$$

and $H\left(x\right)$ is the Heaviside unit step function. From Eq. (28) it follows that $g\left(x\right)=0$ if $|x| \geq 0.5$ so that from Eq. (27) we have

$$k_1\left(x,x'\right) = 0 \;\text{ if }\; \left|x-x'\right| \geq l. \qquad (29)$$

If $|x-x'| < l$ then the integrand of Eq. (27) is nonzero only when $u \in \left(\frac{\max\left(x,x'\right)}{l} - 0.5, \frac{\min\left(x,x'\right)}{l} + 0.5\right)$ therefore

$$k_1\left(x,x'\right) = \sigma \int_{\frac{\max\left(x,x'\right)}{l}-\frac{1}{2}}^{\frac{\min\left(x,x'\right)}{l}+\frac{1}{2}} \cos^2\left(\pi\frac{x}{l}-\pi u\right)\cos^2\left(\pi\frac{x'}{l}-\pi u\right) du \qquad (30)$$

Using the identities $\cos^2\left(x\right) = \left(\cos\left(2x\right)+1\right)/2$ and $2\cos\left(x\right)\cos\left(y\right) = \cos\left(x-y\right) + \cos\left(x+y\right)$ the indefinite integral of the integrand of Eq. (30) can be analytically calculated:

$$\begin{aligned} J\left(u\right) &= \int \cos^2\left(\pi\frac{x}{l}-\pi u\right)\cos^2\left(\pi\frac{x'}{l}-\pi u\right) du \\ &= \frac{2+\cos\left(2\pi\frac{x-x'}{l}\right)}{8}u + \frac{1}{4\pi}\cos\left(\pi\frac{x-x'}{l}\right)\sin\left(2\pi u - \pi\frac{x+x'}{l}\right) \\ &\quad + \frac{1}{32\pi}\sin\left(4\pi u - 2\pi\frac{x+x'}{l}\right) \qquad (31) \end{aligned}$$

From Eqs. (31) and (30) one has that if $|x-x'| < l$ then

$$k_1\left(x,x'\right) = \sigma\left[J\left(\frac{\min\left(x,x'\right)}{l}+\frac{1}{2}\right) - J\left(\frac{\max\left(x,x'\right)}{l}-\frac{1}{2}\right)\right] \qquad (32)$$

which after algebraic manipulations becomes

$$k_1\left(x,x'\right) = \sigma_0\left[\frac{2+\cos\left(2\pi\frac{d}{l}\right)}{3}\left(1-\frac{d}{l}\right) + \frac{1}{2\pi}\sin\left(2\pi\frac{d}{l}\right)\right] \qquad (33)$$

where $d = |x-x'|$ and $\sigma_0 = 3\sigma/8$. Finally, combining Eqs. (29) and (33), we obtain Eq. (9).