Efficacy of data fusion using convolved multi-output Gaussian processes

Shrihari Vasudevan, Arman Melkumyan and Steven Scheding

Australian Centre for Field Robotics, The University of Sydney, NSW 2006, Australia
Email: shrihari.vasudevan@ieee.org | a.melkumyan@acfr.usyd.edu.au | s.scheding@acfr.usyd.edu.au

Abstract

This paper evaluates the efficacy of a machine learning approach to data fusion using convolved multi-output Gaussian processes in the context of geological resource modeling. It empirically demonstrates that information integration across multiple information sources leads to superior estimates of all the quantities being modeled, compared to modeling them individually. Convolved multi-output Gaussian processes provide a powerful approach for simultaneous modeling of multiple quantities of interest while taking correlations between these quantities into consideration. Experiments are performed on large scale data taken from a mining context.

Keywords - Gaussian process, Machine learning, Data fusion, Geological resource modeling, Mining

1 Introduction

Gaussian processes (GPs) [Rasmussen and Williams, 2006] are powerful non-parametric Bayesian learning techniques that can handle correlated, uncertain and incomplete data. GPs yield a continuous domain representation of the data and hence can be sampled at any desired resolution (multi-scale model). They model and use the spatial correlation of the given data to estimate the values for unknown points of interest. GPs perform *Kriging* [Matheron, 1963, Cressie, 1993] interpolation. As a case in point, the work [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2009] modeled large scale terrain using GPs. It proposed the use of the non-stationary neural network kernel to model large scale discontinuous spatial data and empirically demonstrated that this kernel was superior (in modeling) to the stationary squared exponential kernel and at least as good as most standard interpolation techniques for a range of terrain (in terms of sparsity/complexity/discontinuities). Data fusion in the context of Gaussian processes is necessitated by the presence of multiple, multi-sensor, multi-attribute, incomplete and/or uncertain data sets of the entity being modeled. This paper presents an empirical evaluation of a machine learning approach to performing data fusion with Gaussian processes (based on convolved GPs) in the context of vector valued data. The objectives are to understand (1) if simultaneous modeling of multiple quantities of interest using GPs (i.e. modeling and using the correlations between them and

hence performing data fusion) is better than modeling these quantities independently and (2) if non-stationary kernels are more effective than stationary kernels for modeling geological data. Experiments are performed on large scale data obtained from a mining context.

2 Related work

This section first reviews recent works in the machine learning and related communities that addressed the problem of data fusion with Gaussian processes (GPs) and then positions the proposed approach within the wider geostatistics literature on which many of the techniques being presented are based.

Preliminary attempts at data fusion with GPs were shown in [El-Beltagy and Wright, 2001] and [Murray-Smith and Pearlmutter, 2005]. The former demonstrated how a GP could be used to model an expensive process by first modeling a GP on an approximate or cheap process and subsequently using the many input-output data from the approximate process together with the few samples available of the expensive process in order to learn a GP for the latter. The work [Murray-Smith and Pearlmutter, 2005] attempted to generalize arbitrary transformations on GP priors through linear transformations. It hinted at how this framework could be used to introduce heteroscedasticity (random variables with non-constant variance) and how information from different sources could be combined. However, specifics on how the fusion could actually be performed were beyond the scope of the work. Girolami in [Girolami, 2006] integrated heterogeneous feature types within a Gaussian process classification setting, in a protein fold recognition application domain. Each feature representation was represented by a separate GP. Fusion used the idea that individual feature representations were considered independent and hence a composite covariance function could be defined in terms of a linear sum of Gaussian process priors. Reece et al. in [Reece, Roberts, Nicholson, and Lloyd, 2011] integrated "hard" data obtained from sensors with "soft" information obtained from human sources within a Gaussian process classification framework. It used heterogeneous information-types as mutually independent sources of information that were transformed into the kernel representation (a kernel for each kind of information) and combined using a product rule. While [Girolami, 2006] and [Reece, Roberts, Nicholson, and Lloyd, 2011] demonstrated how multiple information-types may be combined using kernel methods, the approach presented in this paper addresses the problem of improving the predictions of several different (heterogeneous) quantities being simultaneously modeled by explicitly modeling correlations between them.

Recent approaches demonstrating data fusion with Gaussian processes in the context of large scale

terrain modeling were based on heteroscedastic GPs [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010a] and dependent GPs [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b, 2011]. These addressed the problem of fusing multiple, multi-sensor data sets of a single quantity of interest. This paper describes the framework for extending this concept to multiple heterogeneous quantities of interest. The work [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010a] treated the data-fusion problem as one of combining different noisy samples of a common entity (terrain) being modeled. The GP representing the fused output was assumed to have non-constant noise variance; the noise variance at any point was dependent on the input data from the different data sets. In the Machine Learning community, this idea is referred to as heteroscedastic GPs [Goldberg, Williams, and Bishop, 1998, Kersting, Plagemann, Pfaff, and Burgard, 2007]. Vasudevan et al in [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b, 2011] treated the data fusion problem as one of improving GP regression through modeling the spatial correlations (auto and cross covariances) between several dependent GPs representing the respective data sets. This idea was inspired by recent machine learning contributions in multi-task or multi-output or dependent GP modeling including [Bonilla, Chai, and Williams, 2007] and [Boyle and Frean, 2004], the latter being based on [Higdon, 2002]. In Kriging terminology, this idea is akin to Co-kriging [Wackernagel, 2003, Goovaerts, 1997]. The work [Vasudevan, 2012] performed a model complexity analysis of multiple approaches to data fusion using GPs, applied in the context of large scale terrain modeling. Melkumyan et al, in [Melkumyan and Ramos, 2011], compared various combinations of stationary kernel including the squared exponential (SQEXP), Matern 3/2 and a sparse covariance function [Melkumyan and Ramos, 2009] in the context of geological resource modeling. This paper evaluates a machine learning approach to data fusion using a convolved GP model applied to a multi-output (vector valued output) problem of geological resource modeling; multiple kernel combinations including the best results from past works mentioned above are compared.

The paper [Álvarez, Rosasco, and Lawrence, 2012] presents an extensive review of kernels for vector valued functions. Drawing parallels between regularization and Bayesian perspectives, the linear model of corregionalization (LMC) [Wackernagel, 2003, Goovaerts, 1997] is presented as a generalized model on which the geostatistical approaches to multivariate modeling and subsequently machine learning / Gaussian process modeling of multivariate data are based. The LMC incorporates a nested framework wherein outputs are first expressed as a linear combination of spatially uncorrelated processes, each of these processes is a linear sum of uncorrelated functions. Within this framework, kernels are broadly

classified as being separable (a Sum of Separable kernels) or non-separable (e.g. divergence-free and curl-free kernels). Separable kernels expresses the covariance between outputs as a product of two terms - one that captures the covariance between outputs, not considering the inputs and the other that captures the covariance between inputs, not considering the outputs. Such separation is not obtained in a nonseparable kernel formulation. Among approaches to developing nonseparable kernels, the process convolution (PC) approach provides a different and flexible way of modeling vector valued (multi-output or multi-task) functions. In the PC approach, each component of a vector output is modeled as a base process (e.g. a GP) convolved with a smoothing kernel and treated with a noise process (e.g. a GP). This transformation of base processes (or latent functions) contrasts with the LMC formulation. The PC approach captures the trend and complexity of the data through the parameters of the smoothing kernel (and additional parameters as required); the LMC relies on a repertoire of latent functions, a linear sum of which will adequately describe the data. Under assumption of Dirac delta smoothing kernels for each output, the covariance between outputs as obtained by the PC approach reduces to that of the LMC approach [Álvarez, Rosasco, and Lawrence, 2012]. This paper adopts the process convolution approach to modeling vector valued data. The proposed approach augments the PC based covariance between input data with a symmetric matrix of free parameters that is intended to capture the signal variance and similarities between outputs being modeled.

This paper reports a detailed multi-metric performance comparison experiment in a geological resource characterization context, performed between a convolved multi-output GP, an equivalent set of GPs (derived from the multi-output GP parameters) and a set of independently optimized GPs, to provide for an exact and an independent comparison between them. The objective is to quantify the benefit (if any) of simultaneous modeling of the multiple quantities by modeling and using the correlations between them as against modeling each of these quantities separately. This paper also compares data fusion using multiple stationary kernels and a nonstationary kernel in the context of modeling geological data. Further discussion on the broader issues relating to the application of the approach in practical problems and the tying together of different prior works that have studied this approach [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b, 2011, Melkumyan and Ramos, 2011] is presented in the technical report version of this paper [Vasudevan, Melkumyan, and Scheding, 2012].

3 Approach

3.1 Gaussian processes for scalar and vector valued functions

Gaussian processes [Rasmussen and Williams, 2006] (GPs) are stochastic processes wherein any finite subset of random variables is jointly Gaussian distributed. They may be thought of as a Gaussian probability distribution in function space. They are characterized by a mean function $m(\mathbf{x})$ and the covariance function $k(\mathbf{x}, \mathbf{x}')$ that together specify a distribution over functions. In the context of geological resource modeling, each $\mathbf{x} \equiv (east, north, depth)$ (3D coordinates) and $f(\mathbf{x}) \equiv z$, the concentration of the quantity being modeled. Although not necessary, the mean function $m(\mathbf{x})$ may be assumed to be zero by scaling/shifting the data appropriately such that it has an empirical mean of zero.

The covariance function or kernel models the relationship between the random variables corresponding to the given data. It can take numerous forms [Rasmussen and Williams, 2006, chap. 4]. The stationary squared exponential (or Gaussian) kernel (SQEXP) is given by

$$k_{SQEXP}(\mathbf{x}, \mathbf{x}', \Sigma) = \sigma_f^2 \,.\, \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Sigma (\mathbf{x} - \mathbf{x}')\right) , \tag{1}$$

where k is the covariance function or kernel; $\Sigma = diag[\ l_{east}\ ,\ l_{north}\ ,\ l_{depth}\ ]^{-2}$ is a d x d diagonal length-scale matrix (d = dimensionality of input = 3 in this case), a measure of how quickly the modeled function changes in the east, north and depth directions; σ_f^2 is the signal variance. The set of parameters $\{\ l_{east}\ ,\ l_{north}\ ,\ l_{depth}\ ,\ \sigma_f\ \}$ are referred to as the kernel hyperparameters.

The non-stationary neural network (NN) kernel [Neal, 1996, Williams, 1998a,b] takes the form

$$k_{NN}(\mathbf{x}, \mathbf{x}', \Sigma) \;=\; \sigma_f^2 \,.\, \frac{2}{\pi}\, \arcsin\left(\frac{2\tilde{\mathbf{x}}^T \Sigma \tilde{\mathbf{x}}'}{\sqrt{(1 \,+\, 2\tilde{\mathbf{x}}^T \Sigma \tilde{\mathbf{x}})(1 \,+\, 2\tilde{\mathbf{x}}'^T \Sigma \tilde{\mathbf{x}}')}}\right) , \tag{2}$$

where $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}}'$ are augmented input vectors (each point is augmented with a 1), Σ is a $(d+1)$ x $(d+1)$ diagonal length-scale matrix given by $\Sigma = diag[\ \beta\ ,\ l_{east}\ ,\ l_{north}\ ,\ l_{depth}\ ]^{-2}$, β being a bias factor and d being the dimensionality of the input data. The variables $\{\ \beta\ ,\ l_{east}\ ,\ l_{north}\ ,\ l_{depth}\ ,\ \sigma_f\ \}$ constitute the kernel hyperparameters. The NN kernel represents the covariance function of a neural network with a single hidden layer between the input and output, infinitely many hidden nodes and using a Sigmoidal transfer function [Williams, 1998a] for the hidden nodes. Hornik, in [Hornik, 1993], showed that such neural networks are universal approximators and Neal, in [Neal, 1996], observed that the functions produced by such a network would tend to a Gaussian process. Prior work in [Vasudevan,

Ramos, Nettleton, and Durrant-Whyte, 2009] found the NN kernel to be more effective than the SQEXP kernel at modeling discontinuous data.

The Matern 3/2 kernel is another stationary kernel differing from the SQEXP kernel in that the latter is infinitely differentiable and consequently tends to have a strong smoothing nature, which is argued as being detrimental to modeling physical processes [Rasmussen and Williams, 2006]. It takes the form specified in Equation 3.

$$k_{MATERN3}(x, x', \Sigma) \;=\; \sigma_f^2 \,.\, \prod_{1 \le k \le d} (1 + \frac{\sqrt{3} r_k}{l_k}) \exp\left(- \frac{\sqrt{3} r_k}{l_k} \right) \tag{3}$$

where $k \,\epsilon\, 1 \ldots d$ is the dimension of the input data (d = dimensionality of input = 3 in this case), $\Sigma = [\; l_{east} \;,\; l_{north} \;,\; l_{depth} \;]$ is a 1 x d length-scale matrix and σ_f^2 is the signal variance. The set of parameters $\{\; l_{east} \;,\; l_{north} \;,\; l_{depth} \;,\; \sigma_f \;\}$ is referred to as the kernel hyperparameters.

Regression using GPs uses the fact that any finite set of training (evaluation) data and test data of a GP are jointly Gaussian distributed. Assuming noise free data, this idea is shown in Expression 4 (hereafter referred to as Equation 4). This leads to the standard GP regression equations yielding an estimate (the mean value, given by Equation 5) and its uncertainty (Equation 6).

$$\begin{bmatrix} \mathbf{z} \\ f_* \end{bmatrix} \;\sim\; N\left(0 \,,\, \begin{bmatrix} K(X,X) & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix} \right) \tag{4}$$

$$\bar{f_*} = K(X_*, X)\; K(X,X)^{-1}\; \mathbf{z} \tag{5}$$

$$\mathrm{cov}(f_*) = K(X_*, X_*) - K(X_*, X) K(X,X)^{-1} K(X, X_*) \tag{6}$$

For n training points $(X, \mathbf{z}) \;=\; (\mathbf{x_i}, z_i)_{i=1 \ldots n}$ and n_* test points (X_*, f_*), $K(X, X_*)$ denotes the $n \times n_*$ matrix of covariances evaluated at all pairs of training and test points. The terms $K(X,X)$, $K(X_*, X_*)$ and $K(X_*, X)$ are defined likewise. In the event that the data being modeled is noisy, a noise hyperparameter (σ) is also learnt with the other GP hyperparameters and the covariance matrix of the training data $K(X,X)$ is replaced by $[K(X,X) + \sigma^2 I]$ in Equations 4, 5 and 6. GP hyperparameters may be learnt using various techniques such as cross validation [Rasmussen and Williams, 2006], maximum-a-posteriori estimation using Markov Chain Monte Carlo techniques [Rasmussen and Williams, 2006, Williams, 1998b] and maximizing the marginal likelihood of the observed training data [Rasmussen and

Williams, 2006, Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2009]. This paper adopts the latter most approach based on the intuition that it may be more suited for large data sets. The marginal likelihood to be maximized is described in Equation 7.

$$\log\ p(\mathbf{z}|X,\theta)\ =\ -\frac{1}{2}\mathbf{z}^T K(X,X)^{-1}\mathbf{z} - \frac{1}{2}\log|K(X,X)| - \frac{n}{2}\log(2\pi) \tag{7}$$

In the geological resource modeling problem considered in this paper, the objective is to model concentrations of multiple elements across the field of interest. The data fusion aspect of this problem is the improved prediction of each one of these concentrations by integration or use of all other concentrations available. If the concentration of each element is modeled using a separate GP, the objective is to improve one GPs prediction estimates given all other GP models. Simultaneous modeling (and consequently data fusion) of multiple concentrations can be achieved by the use of multi-output (MOGP) or dependent GPs [Vasudevan, 2012], also known as multi-task GP if the input points for each element are different. One way of developing such models (specifically, kernels for vector valued functions) is through the process convolution approach. GP regression for vector valued output is presented in the following paragraphs; the next subsection discusses the process convolution approach and specifies the kernel functions used.

Let the number of outputs (element concentrations) that need to be simultaneously modeled be denoted by nt. Equations 4, 5 and 6 represent respectively the MOGP data fusion model, the regression estimates and their uncertainties, subject to the following modifications to the basic notation. The set $\mathbf{z} = [\ \mathbf{z}_1\ ,\ \mathbf{z}_2\ ,\ \mathbf{z}_3\ ,\ ...\ ,\ \mathbf{z}_{nt}\ ]'$ represents the output values of the selected training data (element concentrations) from the individual nt outputs (elements) that need to be simultaneously modeled. The term $X = [\ X_1\ ,\ X_2\ ,\ X_3\ ,\ ...\ ,\ X_{nt}\ ]$ denotes the input location values (east, north, depth) of the selected training data for each of the outputs. Any kernel [Rasmussen and Williams, 2006] may be used and even different kernel could be used for different data sets using the technique demonstrated in [Melkumyan and Ramos, 2011] (for stationary kernel) or the convolution process technique demonstrated in [Higdon, 2002, Boyle and Frean, 2004, Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b, 2011] and in this paper (for both stationary and nonstationary kernel). The covariance matrix of the training data

is given by

$$K(X,X) \equiv \begin{bmatrix} K_{11}^{Y} & K_{12}^{Y} & \dots & K_{1\,nt}^{Y} \\ K_{21}^{Y} & \dots & \dots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ K_{nt\,1}^{Y} & \dots & \dots & K_{nt\,nt}^{Y} \end{bmatrix}$$

where $K_{ii}^{Y} = K_{ii}^{U}(X_i, X_i) + \sigma_i^2 I$ and $K_{ij}^{Y} = K_{ij}^{U}(X_i, X_j)$ represent the auto-covariance of the i^{th} data set with itself and the cross covariance between the i^{th} and j^{th} data sets respectively. These terms model the covariance between the noisy observed data points (z values). Thus, they also take the noise components of the individual data sets / GPs into consideration. The corresponding noise free terms are respectively given by K_{ii}^{U} and K_{ij}^{U}. These are derived by using the process convolution approach to formulating Gaussian processes; details of this follow in the next subsection. The covariance matrix between the test points and training points is given by

$$K(X_*, X) = [K_{i1}^{U}(X_*, X_1), K_{i2}^{U}(X_*, X_2), \dots, K_{i\,nt}^{U}(X_*, X_{nt})] \ ,$$

where $i \,\epsilon\, \{1 \dots nt\}$ is the GP that is being evaluated given all other GPs. The matrix $K(X, X_*)$ is defined likewise. Finally, the covariance of the test points is given by

$$K(X_*, X_*) \ = \ K_{ii}^{U}(X_*, X_*) + \sigma_i^2 I \ ,$$

assuming the i^{th} GP needs to be evaluated for the particular test point. The mean and variance of the concentration estimate can thus be obtained by applying Equations 5 and 6, after incorporating multiple outputs, multiple GP/noise hyperparameters and deriving appropriate auto and cross covariances functions that model the spatial correlation between the individual data sets. Data fusion is thus achieved in the MOGP approach by correlating individual outputs and using this correlation information to improve the prediction estimates of each of them.

3.2 Convolved GPs, auto and cross covariance functions for vector valued functions

The process convolution approach to modeling GPs, proposed in [Higdon, 2002] models a GP as the convolution of a smoothing kernel and a Gaussian white noise process. Thus, the trend and complexity of the data is captured by the parameters of the smoothing kernel. In the case of vector valued outputs, each output (element concentration) can be assumed to be modeled by a separate convolved GP; the

covariance function between two outputs can be derived as a function of the respective smoothing kernel. The work [Higdon, 2002] expressed a relationship between the smoothing kernel and the corresponding covariance function through the Fourier transform and noted that for stationary isotropic kernels, there existed a one-to-one relationship between the covariance function and its smoothing kernel and that for non-isotropic and/or non-stationary kernels, there was no unique solution to the smoothing kernel. The paper suggested that the smoothing kernel for a covariance function could be obtained as the Inverse Fourier Transform of the square root of the spectrum (Fourier transform) of the covariance function. The process convolution approach to multi-output GPs has been used with the stationary SQEXP kernel in [Boyle and Frean, 2004, Álvarez and Lawrence, 2009, Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b, Melkumyan and Ramos, 2011] and the nonstationary NN kernel in [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2011, Vasudevan, 2012]. Given the smoothing kernel of the covariance functions in consideration, the cross-covariance terms can be derived as a kernel correlation between the respective smoothing kernels as demonstrated in [Higdon, 2002, Boyle and Frean, 2004, Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2011, Vasudevan, 2012, Melkumyan and Ramos, 2011]. A more detailed discussion of the approach can be obtained from the technical report version of this paper [Vasudevan, Melkumyan, and Scheding, 2012] and the aforementioned references.

Assume two GPs $N(0, k_i)$ and $N(0, k_j)$, with with length scale matrices Σ_i and Σ_j. Based on [Boyle and Frean, 2004], the cross and auto covariances for the stationary SQEXP kernel are given by Equations 8 and 9 respectively. The corresponding expressions for the nonstationary NN kernel are derived in [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2011, Vasudevan, 2012] and given in Equations 10 and 11 respectively. For the Matern 3/2 kernel, the expressions for the cross covariance and auto covariance are derived in [Melkumyan and Ramos, 2011] and given in Equations 12 and 13 respectively. Based on [Melkumyan and Ramos, 2011], the cross covariance function between a SQEXP and Matern 3/2 kernel is given by Equation 14.

$$K_{ij}^{U}(x, x') = K_f(i, j)\, \frac{(2\pi)^{\frac{d}{2}}}{|\Sigma_i + \Sigma_j|^{\frac{1}{2}}}\, \exp\left(-\frac{1}{2}(x - x')^T \Sigma_{ij} (x - x')\right) \tag{8}$$

where

$$\Sigma_{ij} = \Sigma_i(\Sigma_i + \Sigma_j)^{-1}\Sigma_j = \Sigma_j(\Sigma_i + \Sigma_j)^{-1}\Sigma_i$$

$$K_{ii}^{U}(x, x') = K_f(i, i)\, \frac{(\pi)^{\frac{d}{2}}}{|\Sigma_i|^{\frac{1}{2}}}\, \exp\left(-\frac{1}{4}(x - x')^T \Sigma_i (x - x')\right) \tag{9}$$

$$K_{ij}^{U}(x,x') = K_f(i,j)\ 2^{\frac{d+1}{2}} \frac{|\Sigma_i|^{\frac{1}{4}}|\Sigma_j|^{\frac{1}{4}}}{|\Sigma_i\ +\ \Sigma_j|^{\frac{1}{2}}}\ k_{NN}(\mathbf{x},\mathbf{x}',\Sigma_{ij}) \tag{10}$$

where

$$\Sigma_{ij}\ =\ 2\ \Sigma_i\ (\Sigma_i\ +\ \Sigma_j)^{-1}\ \Sigma_j$$

$$K_{ii}^{U}(x,x')\ =\ K_f(i,i)\ k_{NN}(\mathbf{x},\mathbf{x}',\Sigma_i) \tag{11}$$

$$K_{ij}^{U}(x,x')\ =\ K_f(i,j)\ \prod_{1\leq k\leq d} \frac{2l_{ik}^{\frac{1}{2}}l_{jk}^{\frac{1}{2}}}{l_{ik}^2 - l_{jk}^2}\left(l_{ik}e^{-\sqrt{3}\frac{r_k}{l_{ik}}} - l_{jk}e^{-\sqrt{3}\frac{r_k}{l_{jk}}}\right) \tag{12}$$

where $k\ \epsilon\ 1\ldots d$ is the dimension of the input data, l_i and l_j are the length scales for the two Matern 3/2 kernel based GPs i and j, l_{ik} and l_{jk} are the k^{th} length scales (corresponding to the k^{th} dimensions) of these GPs and $r_k\ =\ |x_k - x'_k|$ is the distance in the k^{th} dimension between the input data.

$$K_{ii}^{U}(x,x')\ =\ K_f(i,i)\ k_{MATERN3}(\mathbf{x},\mathbf{x}',\Sigma_i) \tag{13}$$

$$K_{ij}^{U}(x,x')\ =\ K_f(i,j)\ \prod_{1\leq k\leq d} \sqrt{\lambda_k}\left(\frac{\pi}{2}\right)^{1/4} e^{\lambda_k^2}\left[2\cosh\left(\frac{\sqrt{3}r_k}{l_{Mk}}\right) - e^{\frac{\sqrt{3}r_k}{l_{Mk}}}\ \mathrm{erf}\left(\lambda_k + \frac{r_k}{l_{SEk}}\right) - e^{-\frac{\sqrt{3}r_k}{l_{Mk}}}\ \mathrm{erf}\left(\lambda_k - \frac{r_k}{l_{SEk}}\right)\right] \tag{14}$$

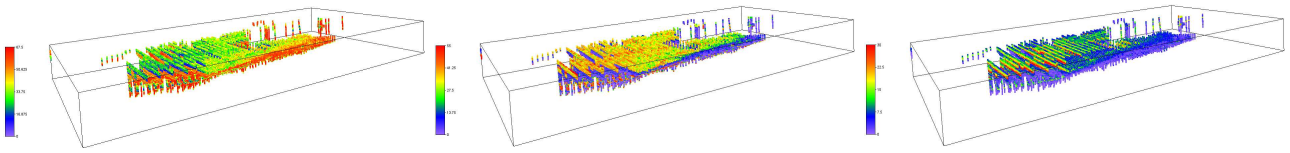
where $\lambda_k = \frac{\sqrt{3}}{2}\frac{l_{SEk}}{l_{Mk}}$, $\mathrm{erf}\,(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\mathrm{d}t$, $k\ \epsilon\ 1\ldots d$ is the dimension of the input data, l_{SE} and l_M are the respective length scales for the SQEXP and Matern 3/2 kernel based GPs i and j, l_{SEk} and l_{Mk} are the k^{th} length scales (corresponding to the k^{th} dimensions) of these GPs and $r_k\ =\ |x_k - x'_k|$ is the distance in the k^{th} dimension between the input data.

In Equations 10 and 11, the term, $k_{NN}(\mathbf{x},\mathbf{x}',\Sigma_{ij})$, is the NN kernel for two data $\mathbf{x}$, $\mathbf{x}'$ and length scale matrix Σ_{ij}. It is given by Equation 2, excluding the signal variance term (σ_f^2). Likewise, in Equation 13, $k_{MATERN}(\mathbf{x},\mathbf{x}',\Sigma_i)$ refers to the Matern 3/2 kernel for two data $\mathbf{x}$, $\mathbf{x}'$ and length scale matrix Σ_{ij}, given by Equation 3 (excluding the σ_f^2 term). The K_f terms in Equations 8, 9, 10 and 11 are inspired by [Bonilla, Chai, and Williams, 2007]. This term models the similarity between individual outputs and also incorporates the signal variance. It is a symmetric matrix of size nt x nt and is learnt along with the other GP hyperparameters. Thus, the hyperparameters of the system that need to be learnt include $(nt.(nt+1))/2$ output similarity values, $nt\,.\,2$ or $nt\,.\,3$ length scale values respectively for the

individual SQEXP/MATERN3 or NN kernels and nt noise values corresponding to the noise in the observed data sets. Learning these hyperparameters by adapting the GP learning procedure described before (Equation 7) for multiple outputs [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b, 2011].

4 Experiments

4.1 Data set



(a) Element-1 (E1) concentration (b) Element-2 (E2) concentration (c) Element-3 (E3) concentration

Figure 1: The geological resource data set. Figures 1(a), 1(b) and 1(c) respectively show the concentrations of three elements over the region of interest. The central region of points is surrounded by sparse sets of points which are not pre-filtered when applying the proposed algorithm.

Experiments were conducted on a large scale geological resource data set made up of real sensor data obtained from a mine site. The data consists of 63,667 measurements from a 3478.4 m x 1764.6 m x 345.9 m region in Australia that has undergone drilling and chemical assays to determine its composition. The holes are generally 25-100m apart and tens to hundreds of meters deep. Within each hole, data is collected at an interval of 2m. The measurements include the (east, north, depth) position data along with the concentrations of three elements, Element-1, Element-2 and Element-3, hereafter denoted as E1, E2 and E3 respectively. The names of these elements have been withheld at the request of the sponsor of this work. These three elements are known to be correlated and hence the objective is to use each of their GP models to improve the others' prediction estimates by capturing the correlation between these quantities. The data set is shown in Figure 1.

4.2 Testing procedure

The objective of the experiment was to compare the convolved multi-output GP approach with a conventional GP approach and quantify if the data fusion in the MOGP actually improves estimation. A second objective of the experiments was to compare the nonstationary NN kernel with the stationary SQEXP kernel, the Matern 3/2 kernel and a combination of them (Matern 3/2 - Matern 3/2 - SQEXP)

that proved effective in prior testing [Melkumyan and Ramos, 2011]. Towards these aims, a ten fold cross validation experiment was performed on the data set, with each of the kernels. This was motivated by the work [Kohavi, 1995], which suggests a ten fold stratified (similar number of samples in each fold) cross validation as the best way of testing the estimation accuracy of machine learning methods on real world data sets.

The MOGP and simple GP approaches each require an optimization step for model learning. The optimization step in each method can result in different local minima in each trial (and with each kernel). Thus, to do a one-on-one comparison between the two approaches and quantify their relative performances, an exact comparison is required. The performance comparison experiment presented in this paper provides an *exact* comparison between the MOGP and GP approaches. To do this,

- The best available MOGP parameters were found for each kernel. From this, appropriate subsets of the parameters were chosen for the GP approach. The idea of the exact comparison is to use separate GPs with parameters derived from the MOGP parameters and test the effect of discarding the other correlated data sources.
- The approaches were compared on identical test points and identical training/evaluation points selected for each of the test points.
- It is also necessary that the covariance function for the simple GP approach *must* be identical to the auto-covariance function of the DGP approach. For this reason, the auto-covariance function (for both kernels) is used as the covariance function for the GP approach to data fusion.

In addition to this, three independently optimized GPs (denoted as GPI here after) were optimized for E1, E2 and E3 and their estimates for the same set of test points were also compared. Thus the effect of information integration in the context of the geological resource modeling can be seen in terms of both an exact comparison (MOGP vs GP) and an independent comparison (MOGP vs GPI).

For the ten fold cross validation, a "block" sampling technique (see Figure 2) was used, a 3D version of the "patch" sampling method used in [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2009]. The idea was that rather than selecting test points uniformly, blocks of data test the robustness of the approach better as the support points to the query point are situated farther away (outside the block) than in uniform point selection. The GP models are used to predict (interpolate) the concentrations at all points within a test block; the larger the block size, the farther the support data to make predictions for points within the block. The data set is gridded into blocks of different sizes. Collections of blocks

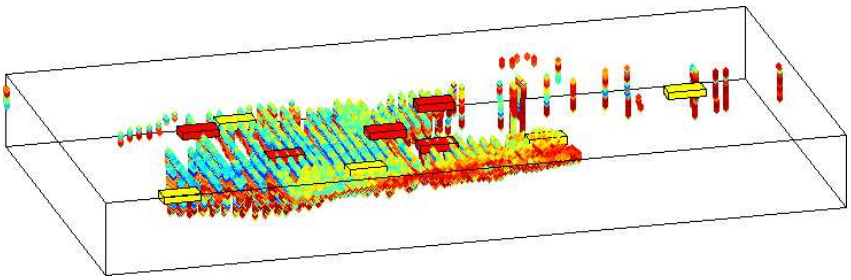


Figure 2: Example of 3D block sampling of a geological resource data set. Blocks may be sampled of different sizes. The red and yellow blocks represent blocks from two of the ten folds used in cross validation testing. Test points within these blocks have "support" data away from them, outside the blocks. This sampling method is therefore a stronger test of the robustness of an approach to estimating the quantity of interest, as compared to uniformly sampling test points. The estimation errors however, will be higher than that obtained for a uniformly sampled set of points.

represent individual folds. In each cross validation test, one fold was designated as a test fold and points from it were used exclusively for testing. All other folds together constituted the evaluation data, a small subset of which were labeled as the training data. Note that this technique of testing will naturally lead to larger errors. For the test fold, the E1, E2 and E3 concentrations (and error metrics defined in the following section) are estimated first using the MOGP approach, then with the GP approach for each of the three elements using parameters derived from the MOGP parameters and finally, with an independently optimized GP for each of the three quantities.

Table 1: Block sizes tested and implications on results - 10 fold cross validation with block sampling, 63667 points spread over 3478.4 m x 1764.6 m x 345.9 m

Block size (m)	Number of points in fold with MIN test points	Number of points in fold with MAX test points	Comments on cross validation test
22 x 11 x 2	6209	6454	Most stratified cross validation Least prediction error
44 x 22 x 4	6183	6456	stratification ↓ prediction error ↑
87 x 45 x 9	5807	6739	stratification ↓ prediction error ↑
174 x 89 x 18	5133	7549	stratification ↓ prediction error ↑
348 x 177 x 35	4976	9662	stratification ↓ prediction error ↑
696 x 353 x 70	1204	10371	Least Stratified cross validation Highest prediction error

Block sizes were chosen empirically, in proportion (arbitrarily rounded up or down) to the dimensions of the whole data set and with a view of performing a stratified cross validation test. The block sizes chosen and the resulting implications on the cross validation testing are shown in Table 1. The smaller block size of 22m x 11m x 2m results in each fold having a similar number of points (i.e. numbers of points in folds with min/max test points are similar) and thus results in the most stratified cross validation test. With increasing block size, prediction error increases (support data is farther away), stratification is reduced and hence, variance in prediction error also increases. Uniform sampling of test

points may be considered as a limiting case of block sampling with the smallest block size possible.

4.3 Metrics

Multiple metrics have been used to understand the various methods being tested. They are briefly described below. These are evaluated for each test point in each fold of the cross validation test. The result would then be represented by the mean and standard deviations of all values across all folds.

1. *Squared Error (SE):* This represents the squared difference between the predicted concentration and the known concentrations for the set of test points. The mean over the set of all test points (Mean Squared Error or MSE) is the most popular metric for the context of this paper. Referring Equations 5 and 6, for the i^{th} test point,

$$SE(i) \ = \ (\bar{f}_*(i) - z_i)^2$$

2. *Variance (VAR):* This represents the variance (uncertainty) in the predicted concentrations for the set of test points. a lower VAR is a good outcome, only if the SE is also low. A model that has high SE and low VAR would be a poor model as this result would suggest that the model is confident of its inaccurate estimates. A better outcome would be a model with high SE and correspondingly high VAR i.e. a model that has inaccurate predictions but is also uncertain about these predictions.

3. *Negative log probability / Log loss (NLP):* Inspired by [Rasmussen and Williams, 2006, page 23], this is a measure of the extent to which the model (including the GP model, kernel, parameters and evaluation data) explain the current test point. The lower the value of this metric, the better the model. For the i^{th} test point,

$$NLP(i) \ = \ \frac{1}{2} log(2\pi\sigma_*^2) + \frac{(\bar{f}_*(i) - z_i)^2}{2\sigma_*(i)^2}$$

4.4 Results

Figure 3 shows the predicted concentrations of E1, E2 and E3 over the entire region of interest as well as 2D section views of this output and the uncertainty of the predictions that constitute it; these were produced using multi-output GPs using the Neural Network kernel. Tables 2, 3 and 4 show the results of the cross validation testing on the geological resource data set with the Neural Network (NN), Matern

Table 2: E1 concentration estimation; 10 fold cross validation results using block sampling of various block sizes; Multi-output GP (MOGP) vs GP derived from MOGP (GP) vs Independently optimized GP (GPI) using Neural Network (NN), Matern 3/2 (MM), Squared exponential (SQEXP) and a Matern 3/2 - Matern 3/2 - Squared Exponential (MS) kernel combination on identical test data. The error metrics are expressed in squared units (squ).

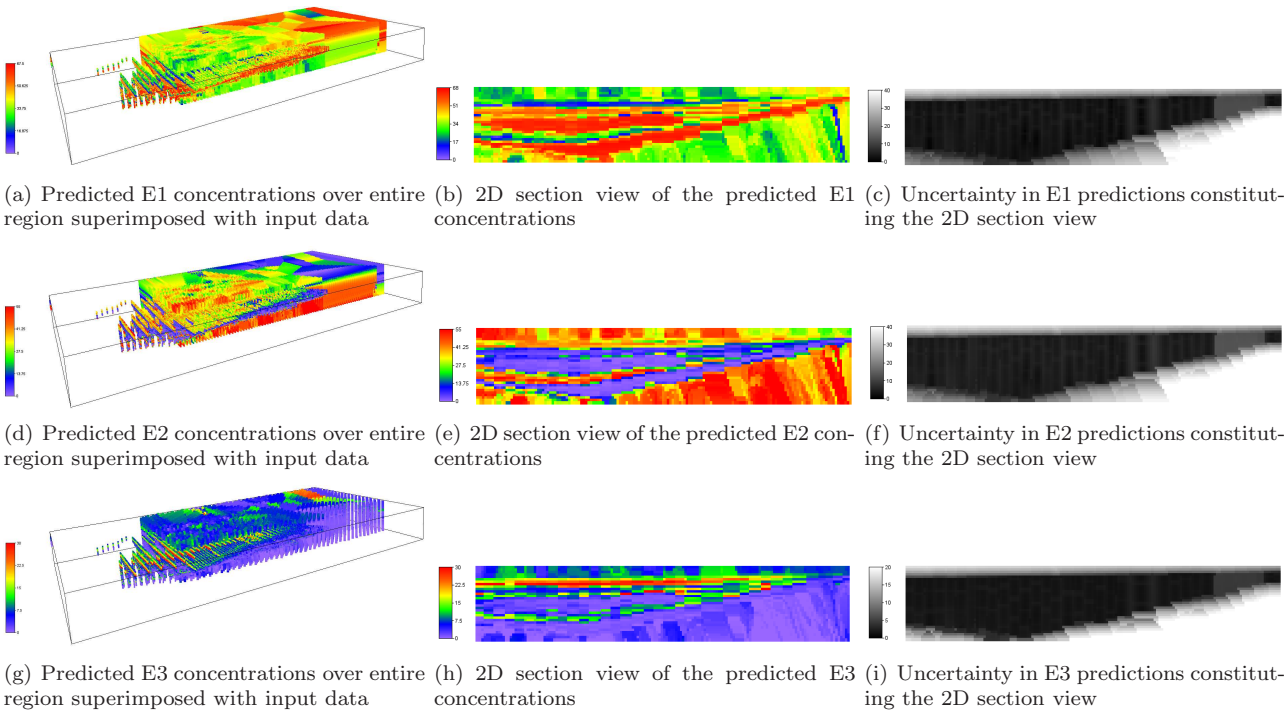
Block size	Method	NN kernel			MM kernel			SQEXP kernel			MS kernel		
(m)		SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)
22 x 11 x 2	MOGP	1.59	1.47	1.68	2.66	0.96	2.35	23.14	0.39	31.63	34.08	33.48	3.18
		(7.25)	(0.28)	(2.75)	(8.66)	(0.15)	(5.01)	(55.62)	(0.10)	(76.20)	(76.16)	(1.48)	(1.14)
	GP	36.52	14.93	3.56	41.42	5.96	5.77	44.43	0.51	45.37	43.49	36.98	3.30
		(86.50)	(8.23)	(3.25)	(96.79)	(5.50)	(9.97)	(122.12)	(0.74)	(103.28)	(95.17)	(5.59)	(1.27)
	GPI	41.28	73.23	3.34	45.26	76.44	3.38	52.96	86.80	3.45	45.26	76.44	3.38
		(90.35)	(7.30)	(0.60)	(96.79)	(5.23)	(0.62)	(107.62)	(4.74)	(0.61)	(96.79)	(5.23)	(0.62)
44 x 22 x 4	MOGP	1.86	1.81	1.74	2.79	1.17	2.19	24.29	0.48	28.69	39.49	34.57	3.26
		(9.49)	(0.59)	(2.66)	(10.51)	(0.24)	(4.53)	(57.54)	(0.20)	(68.94)	(87.09)	(1.71)	(1.27)
	GP	52.75	26.55	3.60	65.00	13.91	5.05	76.89	1.07	50.71	57.55	40.75	3.46
		(124.18)	(17.09)	(2.72)	(149.17)	(14.52)	(7.27)	(235.51)	(3.92)	(113.15)	(124.37)	(10.77)	(1.48)
	GPI	55.81	81.28	3.45	58.74	81.80	3.47	65.30	89.95	3.53	58.74	81.80	3.47
		(119.69)	(11.10)	(0.71)	(124.07)	(8.02)	(0.74)	(132.02)	(6.61)	(0.72)	(124.07)	(8.02)	(0.74)
84 x 45 x 9	MOGP	3.38	3.24	1.91	7.50	1.65	3.04	30.08	0.85	21.35	52.13	36.90	3.42
		(22.92)	(5.06)	(2.56)	(28.75)	(0.47)	(6.07)	(70.98)	(0.53)	(49.50)	(109.11)	(2.07)	(1.47)
	GP	85.88	58.60	3.71	114.05	44.66	4.44	265.25	8.06	45.35	91.75	57.17	3.71
		(187.61)	(39.70)	(2.00)	(242.32)	(37.46)	(4.35)	(853.00)	(25.34)	(95.72)	(190.74)	(33.92)	(1.63)
	GPI	85.53	100.04	3.63	86.81	97.42	3.64	91.63	101.22	3.67	86.81	97.42	3.64
		(171.85)	(21.34)	(0.81)	(175.15)	(15.97)	(0.85)	(179.65)	(13.42)	(0.85)	(175.15)	(15.97)	(0.85)
174 x 89 x 18	MOGP	14.60	10.97	2.24	32.05	2.24	7.29	52.96	1.54	19.27	83.79	39.05	3.81
		(88.59)	(27.66)	(2.70)	(96.26)	(0.61)	(15.78)	(122.09)	(0.78)	(41.43)	(166.79)	(2.41)	(2.09)
	GP	128.39	113.03	3.84	156.56	95.10	4.23	701.62	34.69	28.36	154.34	104.53	3.94
		(261.10)	(88.51)	(1.60)	(306.22)	(56.46)	(3.05)	(1787.63)	(59.68)	(59.58)	(319.96)	(106.83)	(1.58)
	GPI	124.52	129.20	3.79	124.93	121.09	3.80	128.86	122.29	3.82	124.93	121.09	3.80
		(235.22)	(45.85)	(0.88)	(240.35)	(26.43)	(0.95)	(244.59)	(23.44)	(0.96)	(240.35)	(26.43)	(0.95)
348 x 177 x 35	MOGP	73.42	64.36	3.00	112.99	2.74	19.76	114.47	2.45	24.18	155.36	40.86	4.64
		(213.89)	(86.94)	(2.09)	(206.86)	(0.64)	(32.27)	(214.94)	(0.98)	(42.85)	(257.28)	(2.50)	(3.08)
	GP	204.57	249.34	4.06	215.40	153.09	4.25	1091.93	124.02	16.84	290.14	329.34	4.23
		(387.09)	(232.37)	(1.39)	(373.05)	(71.73)	(2.51)	(2801.29)	(94.50)	(43.56)	(541.45)	(376.16)	(1.21)
	GPI	189.14	199.86	3.98	189.21	151.49	4.00	190.92	155.24	4.01	189.21	151.49	4.00
		(335.76)	(120.66)	(0.90)	(327.56)	(45.31)	(1.04)	(331.84)	(44.34)	(1.03)	(327.56)	(45.31)	(1.04)
696 x 353 x 70	MOGP	180.64	173.67	3.61	206.18	2.97	33.83	214.11	2.89	37.05	243.80	41.95	5.66
		(368.05)	(154.95)	(1.56)	(349.73)	(0.52)	(54.10)	(357.84)	(0.84)	(58.21)	(380.23)	(2.21)	(4.46)
	GP	325.98	562.94	4.31	301.60	192.63	4.43	871.72	185.80	12.58	460.23	976.69	4.55
		(546.05)	(523.54)	(1.19)	(452.75)	(64.46)	(2.29)	(2481.90)	(93.03)	(39.08)	(775.98)	(915.85)	(1.01)
	GPI	291.72	362.43	4.19	282.28	180.05	4.23	283.43	183.92	4.23	282.28	180.05	4.23
		(465.37)	(271.48)	(0.80)	(428.51)	(49.34)	(1.12)	(430.84)	(49.45)	(1.11)	(428.51)	(49.34)	(1.12)

Table 3: E2 concentration estimation; 10 fold cross validation results using block sampling of various block sizes; Multi-output GP (MOGP) vs GP derived from MOGP (GP) vs Independently optimized GP (GPI) using Neural Network (NN), Matern 3/2 (MM), Squared exponential (SQEXP) and a Matern 3/2 - Matern 3/2 - Squared Exponential (MS) kernel combination on identical test data. The error metrics are expressed in squared units (squ).

Block size	Method	NN kernel			MM kernel			SQEXP kernel			MS kernel		
(m)		SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)
22x11x2	MOGP	3.85	3.13	2.10	3.88	2.02	2.23	52.32	0.04	650.87	36.09	39.09	3.20
		(17.59)	(0.53)	(2.87)	(18.47)	(0.36)	(4.74)	(231.51)	(0.10)	(1848.02)	(102.25)	(2.48)	(1.27)
	GP	36.86	19.44	3.34	40.79	8.45	4.37	75.67	0.05	668.93	46.60	43.34	3.31
		(118.99)	(8.89)	(3.10)	(129.77)	(6.63)	(7.41)	(869.23)	(0.30)	(1880.52)	(134.94)	(6.65)	(1.36)
	GPI	49.79	93.47	3.44	53.61	97.26	3.47	60.25	85.95	3.48	53.61	97.26	3.47
		(143.90)	(7.40)	(0.71)	(151.69)	(5.97)	(0.72)	(161.56)	(5.68)	(0.86)	(151.69)	(5.97)	(0.72)
44x22x4	MOGP	4.79	3.80	2.20	4.72	2.53	2.29	88.25	0.10	657.09	42.82	40.31	3.28
		(22.21)	(1.06)	(2.87)	(22.80)	(0.58)	(4.43)	(352.70)	(0.30)	(1840.03)	(117.40)	(2.88)	(1.40)
	GP	55.51	32.14	3.49	64.20	18.19	4.19	181.27	0.26	694.20	64.50	47.76	3.46
		(174.24)	(18.07)	(2.70)	(192.62)	(17.10)	(5.58)	(1385.73)	(2.48)	(1901.88)	(186.52)	(12.97)	(1.58)
	GPI	68.69	101.11	3.55	71.21	102.80	3.56	77.86	89.38	3.57	71.21	102.80	3.56
		(195.85)	(10.86)	(0.86)	(201.30)	(8.82)	(0.88)	(211.96)	(9.01)	(1.04)	(201.30)	(8.82)	(0.88)
84x45x9	MOGP	8.04	6.22	2.37	10.97	3.69	2.81	211.05	0.56	461.55	56.72	42.90	3.44
		(40.48)	(6.00)	(2.71)	(44.78)	(1.12)	(5.02)	(753.57)	(1.15)	(1290.67)	(146.64)	(3.41)	(1.64)
	GP	95.98	66.23	3.71	116.98	54.56	4.05	1140.18	4.72	532.52	105.69	67.37	3.69
		(274.95)	(41.41)	(2.27)	(318.33)	(43.24)	(3.56)	(8156.99)	(22.50)	(1420.66)	(288.37)	(42.25)	(1.70)
	GPI	105.20	119.02	3.72	105.27	119.32	3.72	115.96	103.63	3.74	105.27	119.32	3.72
		(277.85)	(20.10)	(1.03)	(279.95)	(17.19)	(1.06)	(301.71)	(23.63)	(1.22)	(279.95)	(17.19)	(1.06)
174x89x18	MOGP	21.49	15.88	2.66	37.85	5.09	4.85	402.24	1.92	228.66	90.32	45.49	3.79
		(102.60)	(29.16)	(2.61)	(117.51)	(1.44)	(8.77)	(1144.48)	(2.06)	(752.97)	(211.07)	(3.73)	(2.23)
	GP	142.62	123.42	3.88	165.30	112.93	4.07	3510.60	25.86	312.01	170.93	125.06	3.91
		(356.16)	(91.33)	(1.88)	(394.82)	(64.77)	(2.70)	(14425.39)	(59.06)	(914.26)	(420.58)	(134.64)	(1.59)
	GPI	148.71	146.71	3.88	147.83	145.39	3.88	164.86	133.30	3.92	147.83	145.39	3.88
		(347.79)	(41.34)	(1.09)	(351.45)	(28.92)	(1.12)	(376.69)	(41.47)	(1.25)	(351.45)	(28.92)	(1.12)
348x177x35	MOGP	82.02	72.71	3.23	119.86	6.26	10.28	419.72	4.61	134.53	167.10	48.08	4.54
		(233.18)	(90.66)	(2.04)	(236.67)	(1.52)	(16.06)	(1196.06)	(2.90)	(599.11)	(320.09)	(4.15)	(3.23)
	GP	219.37	265.84	4.09	232.89	178.16	4.17	8397.84	114.38	164.10	314.92	414.05	4.23
		(484.76)	(239.30)	(1.71)	(475.45)	(81.64)	(2.19)	(28045.42)	(102.39)	(633.41)	(689.64)	(500.88)	(1.24)
	GPI	213.44	213.16	4.05	216.33	182.88	4.06	234.24	194.54	4.11	216.33	182.88	4.06
		(442.90)	(107.76)	(1.09)	(443.91)	(51.20)	(1.18)	(459.31)	(59.86)	(1.17)	(443.91)	(51.20)	(1.18)
696x353x70	MOGP	196.72	189.18	3.75	227.42	6.86	17.23	420.95	6.14	94.13	273.23	49.71	5.58
		(379.37)	(162.78)	(1.48)	(370.75)	(1.26)	(24.68)	(1016.76)	(2.62)	(458.35)	(440.21)	(4.07)	(4.37)
	GP	340.01	594.95	4.34	331.10	223.12	4.37	5910.35	189.13	105.40	493.28	1309.98	4.59
		(603.14)	(544.29)	(1.33)	(534.42)	(73.21)	(1.77)	(23857.18)	(106.28)	(514.02)	(924.16)	(1309.73)	(0.96)
	GPI	314.51	365.16	4.24	317.37	218.58	4.27	331.32	236.80	4.32	317.37	218.58	4.27
		(529.84)	(250.80)	(0.91)	(519.71)	(58.44)	(1.15)	(531.79)	(60.61)	(1.13)	(519.71)	(58.44)	(1.15)

Table 4: E3 concentration estimation; 10 fold cross validation results using block sampling of various block sizes; Multi-output GP (MOGP) vs GP derived from MOGP (GP) vs Independently optimized GP (GPI) using Neural Network (NN), Matern 3/2 (MM), Squared exponential (SQEXP) and a Matern 3/2 - Matern 3/2 - Squared Exponential (MS) kernel combination on identical test data. The error metrics are expressed in squared units (squ).

Block size	Method	NN kernel			MM kernel			SQEXP kernel			MS kernel		
(m)		SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)	SE (squ) mean (std)	VAR (squ) mean (std)	NLP mean (std)
22 x 11 x 2	MOGP	1.60	1.29	1.64	1.29	0.98	1.57	7.30	0.64	6.50	19.15	36.32	2.98
		(6.82)	(0.35)	(2.49)	(5.42)	(0.14)	(2.84)	(20.47)	(0.10)	(16.50)	(45.80)	(0.17)	(0.63)
	GP	9.09	3.84	2.83	9.65	2.36	3.57	10.47	0.75	8.21	19.18	36.47	2.98
		(26.21)	(1.76)	(3.67)	(27.41)	(1.44)	(6.46)	(29.99)	(0.40)	(21.19)	(45.75)	(0.19)	(0.63)
	GPI	9.69	14.64	2.58	11.19	16.90	2.66	12.06	18.46	2.70	12.06	18.46	2.70
		(27.69)	(3.24)	(0.90)	(31.47)	(1.16)	(0.94)	(33.46)	(1.15)	(0.91)	(33.46)	(1.15)	(0.91)
44 x 22 x 4	MOGP	1.90	1.68	1.71	1.50	1.19	1.62	7.99	0.73	6.49	22.53	36.35	3.03
		(8.29)	(0.66)	(2.17)	(7.41)	(0.23)	(2.92)	(21.94)	(0.19)	(16.04)	(52.33)	(0.20)	(0.72)
	GP	12.80	6.38	2.91	14.47	4.52	3.50	15.59	1.07	9.56	22.56	36.51	3.03
		(37.03)	(3.66)	(3.17)	(40.66)	(3.53)	(5.34)	(45.59)	(1.45)	(23.82)	(52.27)	(0.23)	(0.72)
	GPI	12.88	16.79	2.69	14.16	18.14	2.76	14.92	19.41	2.79	14.92	19.41	2.79
		(36.55)	(4.65)	(1.00)	(39.23)	(1.79)	(1.08)	(40.78)	(1.73)	(1.05)	(40.78)	(1.73)	(1.05)
84 x 45 x 9	MOGP	2.87	2.79	1.88	3.08	1.65	1.99	9.19	1.09	5.46	29.82	36.43	3.13
		(13.66)	(1.97)	(2.06)	(12.59)	(0.42)	(3.53)	(24.11)	(0.48)	(12.14)	(63.83)	(0.31)	(0.88)
	GP	20.35	13.22	3.01	24.26	12.01	3.33	35.53	3.73	9.73	29.82	36.61	3.13
		(55.84)	(8.54)	(2.45)	(64.08)	(8.52)	(3.63)	(99.80)	(7.16)	(21.78)	(63.64)	(0.36)	(0.87)
	GPI	19.56	21.69	2.86	20.50	21.61	2.92	21.21	22.73	2.94	21.21	22.73	2.94
		(51.95)	(8.26)	(1.08)	(53.15)	(3.60)	(1.19)	(54.63)	(3.76)	(1.17)	(54.63)	(3.76)	(1.17)
174 x 89 x 18	MOGP	6.63	6.00	2.17	8.67	2.19	3.07	14.56	1.73	5.51	39.96	36.66	3.27
		(35.07)	(8.73)	(2.24)	(24.92)	(0.54)	(4.93)	(34.46)	(0.70)	(10.69)	(80.47)	(0.54)	(1.10)
	GP	29.91	25.03	3.13	32.57	23.31	3.28	73.85	12.06	7.34	39.81	36.89	3.26
		(81.55)	(19.65)	(2.01)	(79.36)	(12.64)	(2.58)	(189.74)	(14.90)	(14.06)	(79.82)	(0.66)	(1.08)
	GPI	28.27	29.28	3.01	28.87	26.65	3.07	29.50	28.35	3.09	29.50	28.35	3.09
		(69.54)	(16.01)	(1.10)	(69.91)	(6.01)	(1.27)	(72.11)	(6.17)	(1.24)	(72.11)	(6.17)	(1.24)
348 x 177 x 35	MOGP	22.76	23.57	2.64	25.66	2.62	5.81	27.36	2.58	6.50	54.46	37.62	3.46
		(76.19)	(36.17)	(1.91)	(50.80)	(0.57)	(8.42)	(55.29)	(0.89)	(10.74)	(99.06)	(1.59)	(1.32)
	GP	48.12	55.21	3.32	47.08	34.95	3.39	96.50	33.92	5.49	54.31	38.13	3.45
		(113.43)	(53.40)	(1.82)	(100.71)	(15.85)	(2.25)	(228.76)	(22.17)	(10.17)	(98.51)	(2.22)	(1.30)
	GPI	43.19	46.90	3.20	42.96	32.36	3.25	43.24	34.15	3.25	43.24	34.15	3.25
		(97.09)	(36.64)	(1.15)	(92.63)	(10.28)	(1.40)	(93.69)	(10.70)	(1.35)	(93.69)	(10.70)	(1.35)
696 x 353 x 70	MOGP	50.93	81.32	3.10	43.57	2.86	8.55	46.94	3.00	8.98	66.54	40.95	3.58
		(123.14)	(99.31)	(1.43)	(77.06)	(0.48)	(12.36)	(81.94)	(0.77)	(13.24)	(109.25)	(5.93)	(1.32)
	GP	73.71	129.87	3.57	64.20	43.42	3.58	93.45	47.21	4.85	67.03	43.80	3.57
		(146.92)	(126.46)	(1.65)	(112.93)	(14.04)	(2.05)	(205.91)	(21.02)	(8.96)	(109.94)	(11.83)	(1.27)
	GPI	65.75	83.71	3.41	61.34	38.04	3.48	61.26	39.36	3.47	61.26	39.36	3.47
		(122.28)	(72.39)	(1.03)	(108.31)	(10.97)	(1.39)	(108.52)	(10.98)	(1.34)	(108.52)	(10.98)	(1.34)



(a) Predicted E1 concentrations over entire region superimposed with input data

(b) 2D section view of the predicted E1 concentrations

(c) Uncertainty in E1 predictions constituting the 2D section view

(d) Predicted E2 concentrations over entire region superimposed with input data

(e) 2D section view of the predicted E2 concentrations

(f) Uncertainty in E2 predictions constituting the 2D section view

(g) Predicted E3 concentrations over entire region superimposed with input data

(h) 2D section view of the predicted E3 concentrations

(i) Uncertainty in E3 predictions constituting the 2D section view

Figure 3: Figures 3(a), 3(b) and 3(c) respectively show the predicted E1 concentrations (over the entire region) superimposed with the input data, a 2D section-view of the output data and the uncertainty in the predicted concentrations for the 2D view. The corresponding figures for E2 and E3 are 3(d), 3(e), 3(f) and 3(g), 3(h), 3(i) respectively. Expectedly, the uncertainty is low around regions where input/given data exist and rapidly rises for predictions away from such areas - typically, the fringe areas. The 2D section view (Figure 3(b)) shows two red regions corresponding two regions of high E1 concentration. The corresponding regions in Figures 3(e) and 3(h) show low E2 and E3 concentrations respectively.

3/2 (MM), Squared Exponential (SQEXP) and Matern 3/2 - Matern 3/2 - Squared exponential (MS) kernels. Trends from the three tables are visualized through numerous graphs located in the appendix and summarized below. For a more complete set and large sized graphs, see [Vasudevan, Melkumyan, and Scheding, 2012].

1. Prediction error (SE) increases with increase in test block size.

   - See Figures 4(a), 4(d), 4(f), 4(h) for E1, 5(a), 5(d), 5(f), 5(h) for E2 and 6(a), 6(d), 6(f), 6(h) for E3.
   - This behavior is expected. It happens because the support training data required for regressing at a test point is situated farther away. Increasing the test block size also results in reduced stratification as one fold of the cross validation may have e.g. 10,000 test points whereas another may have only 1000 points. This results in increased standard deviation of prediction error. A ten fold stratified cross validation is generally considered to be the most representative of performance measure [Kohavi, 1995], however testing multiple larger block sizes provides a better understanding of the model's behavior and robustness.

2. NN kernel based MOGP/GP models trained faster than other kernels

- Further optimization of each of the MOGP/GP models could yield better results. The results shown are the result of a reasonable amount of optimization applied to each kernel and GP model. Typically, multiple attempts were performed and the best results obtained were pursued/used. One iteration consisted of a stochastic optimization step (simulated annealing) and/or a gradient based optimization step (Quasi Newton optimization with BFGS Hessian update) with 10,000 training data chosen uniformly from the data. This work uses a "block-learning" approximation [Vasudevan, Ramos, Nettleton, and Durrant-Whyte, 2010b] which approximates the total marginal likelihood as a sum of a sequence of marginal likelihoods computed over blocks of points comprising the training data. The size of the block is defined by the computational resources available. The stochastic optimization step was the most time consuming part; each attempt was started with completely random parameters. The code was unoptimized MATLAB code running typically on an 8-core processor based machine. Most times, not all the cores were used for the same process; multiple processes also shared the same system. Note that the experiments in this paper do not use analytical gradients for the optimization of the hyperparameters; this was a design choice made in the interest of stability and comparability of the optimization results across kernels. The use of analytical gradients can significantly reduce the total training time. Training time may also be reduced significantly by various other ways including other approximations, intelligently setting initial parameters, scaling the data etc.

Model	Kernel	Number of training attempts, iterations Total training time for successful attempt
MOGP	NN	2 attempts, 3 iterations, total training time = 78.89 hours
	MM	3 attempts, 2 iterations, total training time = 222.15 hours
	SQEXP	4 attempts, 1 iteration, total training time = 92.52 hours
	MS	2 attempts, 3.5 iterations, 3 iterations took 113.69 hours
GPI	NN	3 attempts, 2 iterations, total training time = 41.07 hours
	MM	2 attempts, 1 iteration, total training time = 48.91 hours
	SQEXP	2 attempts, 1 iteration, total training time = 47.67 hours

Rather than the individual training times, the relative amount of training (under similar con-

ditions, with different kernel) required to produce a reasonable set of parameters is of more interest. Experience suggests that the NN kernel based MOGP/GP models converged faster and better as compared to other kernels.

3. MOGP models based on the NN kernel outperform other kernels tested.

   - See Figures 4(a), 4(b), 4(c) for E1, 5(a), 5(b), 5(c) for E2 and 6(a), 6(b), 6(c) for E3.
   - The NN kernel is the best performing kernel of the four tested, across all block sizes tested. The MOGP based on the NN kernel produces lower SE (better estimate) and reduced NLP (better model) estimates than other kernels tested.
   - For small block sizes, both the NN and MM kernel are competitive; in case of E3, the MM even marginally outperforms the NN kernel for the two smallest block sizes tested. Note however that considering all test sizes and all three elements, the observation is that the MM kernel produces lower VAR for a higher SE, meaning that it is more confident of its SE values which are worse/higher than those of the NN kernel. This makes its NLP higher and the model poorer than an MOGP based on the NN kernel. Note also that as the test block size increases, the advantage in performance of the MOGP based on the NN kernel over that based on the MM kernel becomes more distinctive. Not only are the SE values smaller for the NN kernel, the NLP values remain in the same range whereas those of the MM kernel rise significantly. This proves that the MOGP-NN is better performing and more robust than the MOGP-MM. The latter property suggests that the MOGP-NN will be able to cope better with incomplete data sets.
   - Both the MS and SQEXP kernels are not competitive with respect to the NN or MM kernels considering both the SE and NLP metrics. These kernels are discussed individually in the following paragraphs.

4. MOGP models perform significantly better than three separate GPs (using the MOGP parameters) or three independently optimized GPs as information fusion improves estimation.

   - See Figures 4(d) and 4(e) for E1, 5(d) and 5(e) for E2 and 6(d) and 6(e) for E3.
   - For the NN kernel, the MOGP metrics are always lower than the corresponding derived GP (GP) or independent GP (GPI) metrics - lower SE (better estimate) with lower NLP (better

model). This clearly demonstrates the benefits of information fusion across heterogeneous information sources so as to improve individual predictions using the MOGP model.

- From Tables 2, 3 and 4, the average reduction in error (i.e. improvement in performance) of MOGP models over GP/GPI models for the smallest, intermediate and largest test block sizes are -
  - E1
    - 22 x 11 x 2 - 95.6% over GP, 96.2% over GPI
    - 84 x 45 x 9 - 96.1% over GP, 96.0% over GPI
    - 696 x 353 x 70 - 44.6% over GP, 38.1% over GPI
  - E2
    - 22 x 11 x 2 - 89.6% over GP, 92.3% over GPI
    - 84 x 45 x 9 - 91.6% over GP, 92.4% over GPI
    - 696 x 353 x 70 - 42.1% over GP, 37.5% over GPI
  - E3
    - 22 x 11 x 2 - 82.4% over GP, 83.5% over GPI
    - 84 x 45 x 9 - 85.9% over GP, 85.3% over GPI
    - 696 x 353 x 70 - 30.9% over GP, 22.5% over GPI

These numbers demonstrate significant improvements in performance, even in very large test block sizes, when using the MOGP-NN model for correlated data.

5. The MS kernel was uncompetitive

- See Figures 4(h) and 4(i) for E1, 5(h) and 5(i) for E2 and 6(h) and 6(i) for E3.
- The MS kernel is not competitive with respect to the NN and MM kernels as discussed earlier. However, the MOGP using this kernel combination proves to be better than a derived GP and an independently optimized GP with respect to the SE metric. From the NLP perspective, the MOGP-MS model is more competitive than the other GP models for small block sizes. For larger block sizes, using an independently optimized GP proves to be a more trust worthy modeling option as the increase in error is met with a corresponding increase in uncertainty (hence low NLP) for the independent GP models. The exception to this behavior is seen in

the results for E3, the MOGP model is poor in this case. This is attributed to do with inferior parameters relevant to the element E3 obtained from the optimization process.

- The MS kernel performs better than the SQEXP with respect to the NLP metric and hence can be trusted more (prediction error compensated by prediction uncertainty), but in two of the three elements (E1 and E3), its SE was inferior to that of the SQEXP.

6. The SQEXP kernel was uncompetitive and unreliable

   - See Tables 2, 3 and 4; see Figures 4(a), 4(b) 4(c), 4(f), 4(g), 4(h) and 4(i) for E1, 5(a), 5(b), 5(c), 5(f), 5(g), 5(h) and 5(i) for E2 and 6(a), 6(b), 6(c), 6(f), 6(g), 6(h) and 6(i) for E3.
   - The MOGP-SQEXP model performs poorly in comparison with the equivalent models using the NN/MM kernels, with respect to both SE and NLP.
   - For elements E1 and E3, the MOGP-SQEXP has a better SE than the corresponding model based on the MS kernel; it has an SE better than the corresponding derived/independent GP models but an inferior (overconfident or low uncertainty) VAR and a fluctuating NLP trend. For element E2, the MOGP-SQEXP is worse off than both the equivalent model based on the MS kernel as well as its corresponding GP models.
   - Considering the results for E2, the NLP is directly proportional to the SE and inversely to the prediction variance. At the smallest block size, the MOGP-SQEXP produces relatively high SE (with respect to e.g. MOGP-NN) but very low prediction variance. This basically suggests that the model is confident of its poor estimates - a bad outcome. This results in a high NLP and poor model. As the block size increases, the prediction variance increases more relative to the prediction error resulting in the decreasing NLP trend. For elements E1 and E3, the largest block size results in a stronger increase in prediction error than the variance in the prediction resulting in an increase in NLP. Overall, the MOGP-SQEXP model is poor.
   - The SQEXP kernel is a limiting case of the MM kernel; both are stationary kernels. Considering the behavior of the GPI model using the SQEXP kernel and its competitive results with respect to those of the GPI-MM kernel, it is possible that the poor performance of the MOGP-SQEXP (as compared to the MOGP-MM) is due to poor optimization output (a bad local minima).

7. In general, the stationary kernels tested seemed to have an inadequate increase in prediction uncertainty with increasing test block size and worsening predictions. This leads a higher NLP metric

and a poor model that is overly confident of its worsening predictions. This behavior can be attributed to the correlation profile of the stationary kernels tested - they all share the "correlation decreases with increasing distance of support data from point of interest" trend. This results in stationary kernels not being able to cope with large test block sizes as the support data is farther away (i.e. less correlated and not of much use). In contrast, the nonstationary NN kernel has a sigmoidal profile that can handle this issue across a range of test block sizes.

8. The SE metric taken alone can be misleading. The experiments have reinforced the need for a multi-metric analysis. The SE metric only provides information on the prediction error but it does not describe the prediction uncertainty which is very important in understanding if a model is reliable or otherwise. The VAR and NLP metrics provided key insights on the difference in performance between different models and kernels. A model that is very confident of its poor predictions is unreliable (as was the case for the SQEXP kernel). Worsening predictions (due to increasing test block size) is itself not a bad outcome, provided it is met with an equivalent increase in prediction uncertainty.

9. Further discussion of the results in the context of broader practical issues like determining if a MOGP model is indeed good or identifying the kind of GP model suited for a particular task, are available in the technical report version of this paper [Vasudevan, Melkumyan, and Scheding, 2012].

5 Conclusion

This paper empirically studied the problem of geological resource modeling using a machine learning approach to convolved multi-output Gaussian processes (MOGPs). The concentrations of three elements were modeled and predicted over a region of interest using multi-output Gaussian processes (MOGPs; joint modeling of multiple outputs) as well as individual GPs for each of these quantities separately. The paper demonstrates that MOGPs perform significantly better than individual GPs at the modeling problem as they effectively integrate heterogeneous sources of information (concentrations of individual elements) to improve the predictions of each of them. The benefits of information integration using the MOGP as against independent GPs for the task of geological resource modeling have been quantified by a multi-metric, multi-kernel and multi-test-size cross validation study that performed both an exact and an independent comparison between MOGPs and GPs. Multi-output Gaussian process models based

on the Neural Network kernel was shown to be a competitive and robust modeling option across a range of test block sizes.

Acknowledgements

This work has been funded by the Rio Tinto Centre for Mine Automation.

References

M. A. Álvarez and N. D. Lawrence. Sparse convolved gaussian processes for multi-output regression. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 57–64. 2009.

M. A. Álvarez, L. Rosasco, and N. D. Lawrence. Kernels for Vector-Valued Functions: A Review. *Foundations and Trends in Machine Learning*, 4:195–266, 2012.

E. Bonilla, K. M. Chai, and C. Williams. Multi-task Gaussian process prediction. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 153–160. MIT Press, Cambridge, MA, 2007.

P. Boyle and M. Frean. Dependent Gaussian processes. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 217–224. MIT Press, Cambridge, MA, 2004.

N. Cressie. *Statistics for Spatial Data*. Wiley-Interscience, 1993.

M. A. El-Beltagy and W. A. Wright. Gaussian processes for model fusion. In *International Conference on Artificial Neural Networks (ICANN)*, 2001.

M. Girolami. Bayesian data fusion with gaussian process priors: An application to protein fold recognition. In *Workshop on Probabilistic Modeling and Machine Learning in Structural and Systems Biology (PMSB)*, 2006.

P. W. Goldberg, C. K. I. Williams, and C. M. Bishop. Regression with Input-dependent Noise: A Gaussian Process Treatment. In M. I. Jordan, M. J. Kearns, S. A. Solla, and L. Erlbaum, editors, *Advances in Neural Information Processing Systems (NIPS) 10*. MIT Press, Cambridge, MA, 1998.

P. Goovaerts. *Geostatistics for Natural Resources Evaluation*. Oxford University Press, 1997.

D. Higdon. *Quantitative Methods for Current Environmental Issues*, chapter Space and Space-Time Modeling Using Process Convolutions, pages 37–54. Springer, 2002.

K. Hornik. Some new results on neural network approximation. *Neural Networks*, 6(8):1069–1072, 1993.

K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard. Most Likely Hetereoscedastic Gaussian Process Regression. In *International Conference on Machine Learning (ICML)*, 2007.

R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Joint Conferences on Artificial Intelligence (IJCAI)*, volume 14, pages 1137–1145, 1995.

G. Matheron. Principles of Geostatistics. *Economic Geology*, 58:1246–1266, 1963.

A. Melkumyan and F. Ramos. A sparse covariance function for exact gaussian process inference in large data sets. In *International Joint Conferences on Artificial Intelligence*, volume 21, pages 1936–1942, 2009.

A. Melkumyan and F. Ramos. Multi-Kernel Gaussian Processes. In *in the proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2011.

R. Murray-Smith and B. A. Pearlmutter. *Deterministic and Statistical Methods in Machine Learning, LNAI 3635*, chapter Transformations of Gaussian Process priors, pages 110–123. Springer-Verlag, 2005.

R. M. Neal. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics 118. Springer, New York, 1996.

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

S. Reece, S. Roberts, D. Nicholson, and C. Lloyd. Determining intent using hard/soft data and gaussian process classifiers. In *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, 2011.

S. Vasudevan. Data fusion using gaussian processes. *Elsevier Journal of Robotics and Autonomous Systems*, 2012. URL `http://dx.doi.org/10.1016/j.robot.2012.08.006`. Available online 25 August 2012.

S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte. Gaussian Process Modeling of Large Scale Terrain. *Journal of Field Robotics*, 26(10):812–840, 2009.

S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte. Heteroscedastic Gaussian processes for data fusion in large scale terrain modeling. In *the International Conference for Robotics and Automation (ICRA)*, 2010a.

S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte. Large-scale terrain modeling from multiple sensors with dependent Gaussian processes. In *in the proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, October 2010b.

S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte. Non-stationary dependent Gaussian processes for data fusion in large scale terrain modeling. In *in the proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011.

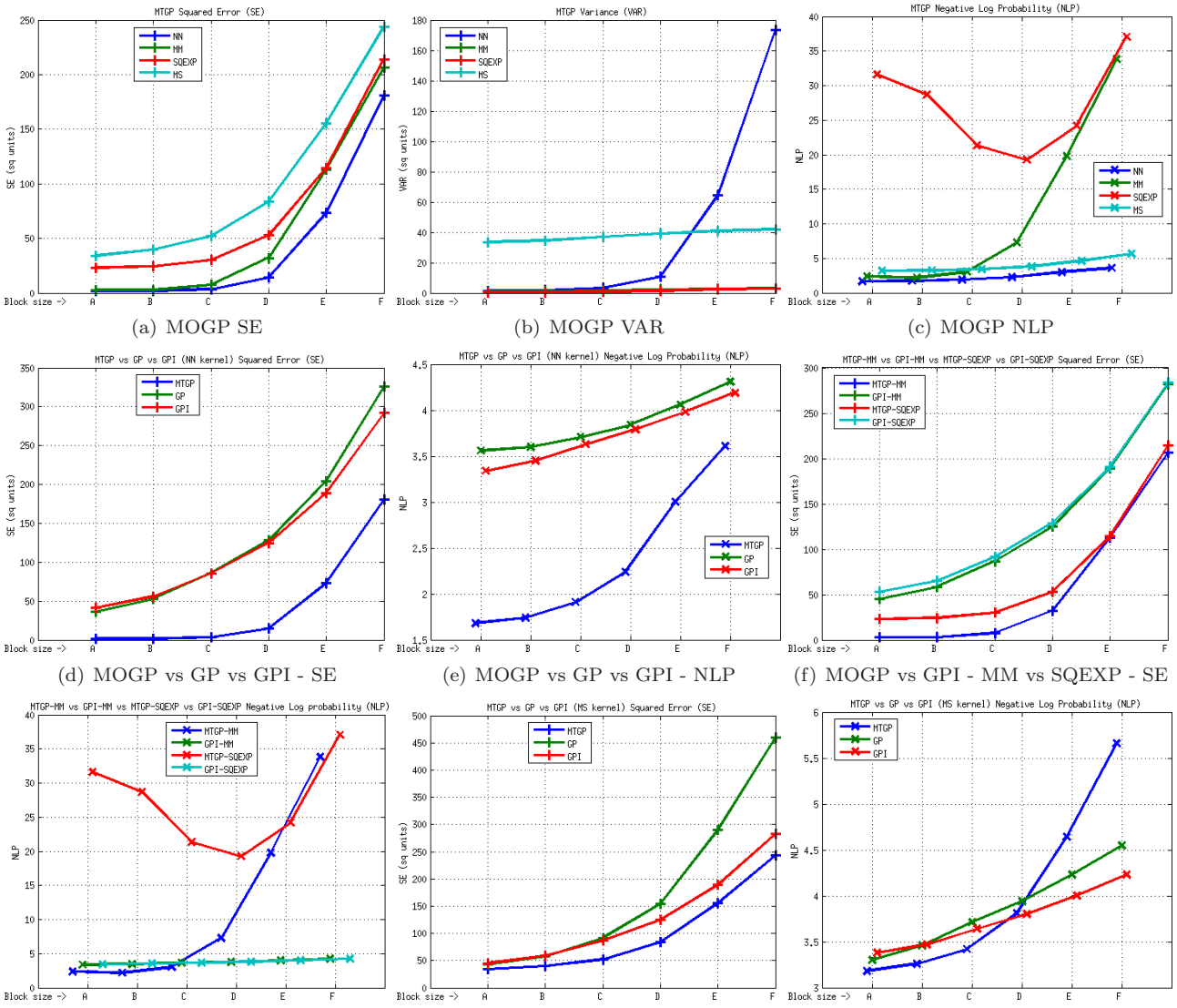
S. Vasudevan, A. Melkumyan, and S. Scheding. Information fusion in multi-task Gaussian process models. Technical report, Australian Centre for Field Robotics, The University of Sydney, 2012. arXiv report 1210.1928, available online at `http://arxiv.org/abs/1210.1928`.

H. Wackernagel. *Multivariate geostatistics: an introduction with applications*. Springer, 2003.

C. K. I. Williams. Computation with infinite neural networks. *Neural Computation*, 10(5):1203–1216, 1998a.

C. K. I. Williams. Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 599–622. Springer, 1998b.

Appendix: Graphs of results obtained in Tables 2, 3 and 4

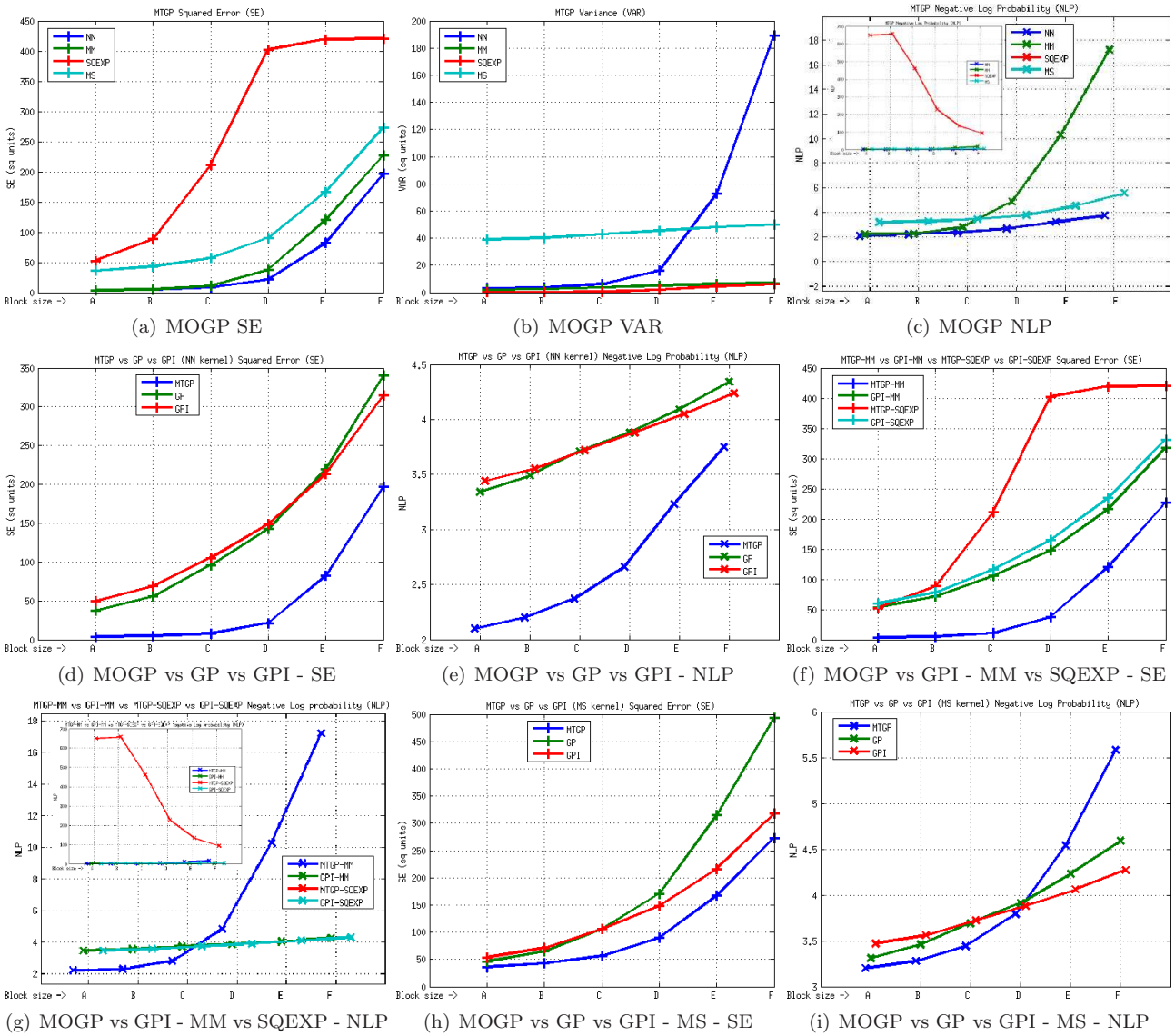


(a) MOGP SE (b) MOGP VAR (c) MOGP NLP

(d) MOGP vs GP vs GPI - SE (e) MOGP vs GP vs GPI - NLP (f) MOGP vs GPI - MM vs SQEXP - SE

(g) MOGP vs GPI - MM vs SQEXP - NLP (h) MOGP vs GP vs GPI - MS - SE (i) MOGP vs GP vs GPI - MS - NLP

Figure 4: Element E1 - key trends. Test block sizes (m) - A (22 x 11 x 2), B (44 x 22 x 4), C (84 x 45 x 9), D (174 x 89 x 18), E (348 x 177 x 35) and F (696 x 353 x 70).

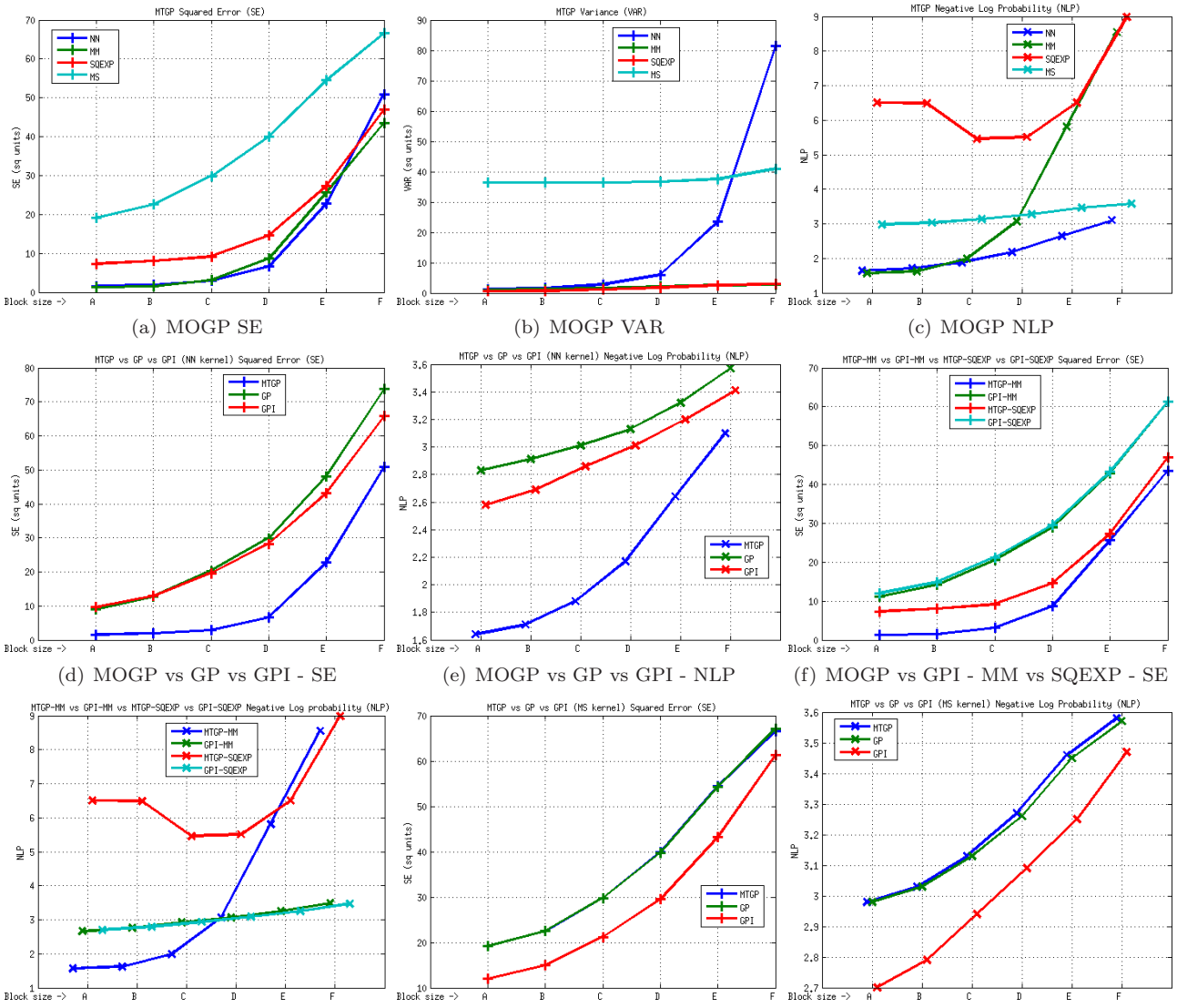


(a) MOGP SE (b) MOGP VAR (c) MOGP NLP

(d) MOGP vs GP vs GPI - SE (e) MOGP vs GP vs GPI - NLP (f) MOGP vs GPI - MM vs SQEXP - SE

(g) MOGP vs GPI - MM vs SQEXP - NLP (h) MOGP vs GP vs GPI - MS - SE (i) MOGP vs GP vs GPI - MS - NLP

Figure 5: Element E2 - key trends. Test block sizes (m) - A (22 x 11 x 2), B (44 x 22 x 4), C (84 x 45 x 9), D (174 x 89 x 18), E (348 x 177 x 35) and F (696 x 353 x 70).



(a) MOGP SE (b) MOGP VAR (c) MOGP NLP

(d) MOGP vs GP vs GPI - SE (e) MOGP vs GP vs GPI - NLP (f) MOGP vs GPI - MM vs SQEXP - SE

(g) MOGP vs GPI - MM vs SQEXP - NLP (h) MOGP vs GP vs GPI - MS - SE (i) MOGP vs GP vs GPI - MS - NLP

Figure 6: Element E3 - key trends. Test block sizes (m) - A (22 x 11 x 2), B (44 x 22 x 4), C (84 x 45 x 9), D (174 x 89 x 18), E (348 x 177 x 35) and F (696 x 353 x 70).