Variable selection for Gaussian process regression through a sparse projection

Chiuoo Park\textsuperscript{a}, David J. Borth\textsuperscript{b}, Nicholas S. Wilson\textsuperscript{c}, and Chad N. Hunter\textsuperscript{c}

\textsuperscript{a}Industrial and Manufacturing Engineering, Florida State University, Tallahassee, FL, USA; \textsuperscript{b}University of Dayton Research Institute, Dayton, OH, USA; \textsuperscript{c}Materials and Manufacturing Directorate, Air Force Research Laboratory, Dayton, OH, USA

ABSTRACT
This article presents a new variable selection approach integrated with Gaussian process regression. We consider a sparse projection of input variables and a general stationary covariance model that depends on the Euclidean distance between the projected features. The sparse projection matrix is considered as an unknown parameter. We propose a forward stagewise approach with embedded gradient descent steps to co-optimize the parameter with other covariance parameters based on the maximization of a non-convex marginal likelihood function with a concave sparsity penalty, and some convergence properties of the algorithm are provided. The proposed model covers a broader class of stationary covariance functions than the existing automatic relevance determination approaches, and the solution approach is more computationally feasible than the existing Markov chain Monte Carlo sampling procedures for the automatic relevance parameter estimation with a sparsity prior. The approach is evaluated for a large number of simulated scenarios. The choice of tuning parameters and the accuracy of the parameter estimation are evaluated with the simulation study. In comparison to some chosen benchmark approaches, the proposed approach has demonstrated improved accuracy in the variable selection. It is applied to an important problem of identifying environmental factors that affect atmospheric corrosion of metal alloys.

1. Introduction
Gaussian Process (GP) regression or Kriging is a non-parametric Bayesian approach for regression analysis (Rasmussen and Williams, 2005). The terms, GP regression and Kriging, originated from different fields, but they are technically equivalent, although the narrow definition of a GP regression often refers to a simple Kriging versus an ordinary Kriging. The GP regression has an analytical closed-form solution and has desirable properties, e.g., it is the best unbiased linear predictor and it provides the mean prediction and the credible interval of the prediction. A major challenge with the GP regression is its performance deterioration as the number of input variables increases, namely the big-p issue, where \( p \) stands for the number of the input variables. According to Liu and Guillas (2017), the upper error bound of the GP regression is proportional to \( n^{-1/p} \), which reduces as the number of data increases, but the reduction rate decreases as \( p \) increases. This is the main reason for data analysts to try to reduce the input dimension by means of a Dimension Reduction (DR) or a Variable Selection (VS) technique. This article is mainly concerned with the VS for the GP in which a large number of inputs are available, but only a small fraction of these inputs are informative.

The main motivating example of this work is to identify the environmental factors most influential to metal corrosion. The outcome will be exploited to design an accelerated corrosion testing protocol using a custom environmental chamber that can simulate real-world conditions including temperature, relative humidity, salt water spray, background gases, and artificial sunlight. Developing the protocol will require two preliminary steps: first identifying the control factors and then calibrating the control factor levels. The benefit of this exercise will be to reduce the number of factors to account for when conducting an experiment in the laboratory test chamber that produces similar metal corrosion to that occurring in a natural environment. Pre-selecting a subset of more influential factors is highly desirable for a more efficient design of the accelerated corrosion test protocol. In addition to the motivating example, the methodology described in this article can be applied to many other applications that benefit from the VS of informative variables among a large number of available input factors in a regression analysis. For example, in the modern manufacturing environment, a large number of sensors are distributed to collect process information in real-time, and that sensor information is often integrated into a GP surrogate model for future process prediction and optimization. Selecting the relevant sensor information for a more accurate GP surrogate model is highly desirable. The proposed approach can be applied as the VS approach in the GP surrogate modeling.

The VS has been often performed as a preliminary step for a GP regression, first performing VS and then running
the GP regression with reduced inputs (Liu and Guillas, 2017). In this step-by-step approach, the VS is not optimized for the follow-up GP learning task. A better approach is to integrate the VS within the GP regression, optimizing the VS step for the GP regression. The existing integrated approaches for the VS in GP are categorized into two studies: automatic relevance determination (Williams and Rasmussen, 1996; Savitsky et al., 2011) and ranking input variables by their relevance to the regression response (Piironen and Vehtari, 2016; Paananen et al., 2019). In the first approach, the inverses of the length scale parameters of a covariance function are used to determine the relevance of input variables to the response variable. To promote the shrinkage of near-zero parameter values to zeros, sparse priors such as a spike-and-slab prior (Savitsky et al., 2011) and a horseshoe prior (Vo and Pati, 2017) are applied as prior distributions of the parameter values. This shrinkage would make selection of the input variables relevant to the regression response. The parameter estimation and GP regression involve expensive Markov Chain Monte Carlo (MCMC) computations, due to the introduction of the sparse prior. Its application is limited with a specific choice of the covariance function called the Automatic Relevance Determination (ARD) covariance function. Other more general forms of the ARD covariance functions are discussed in the literature (Rusmassen and Williams, 2005; Wen and Yin, 2013), but they are discussed mainly for the DR, not for the VS. Its extension for the VS is not straightforward, because of the complexity of incorporating the sparsity regularization of the ARD model parameters into the general ARD covariances. The main contribution of this article is to develop the VS in the GP with general ARD covariance models.

The remainder of this article is organized as follows. Section 2 reviews the relevant literature and covers our major contribution. Section 3 covers a new modeling approach for a sparse projection of the input variables in GP regression and the numerical optimization to estimate the model parameters. Section 4 analyzes the numerical performance of the new approach with a comprehensive set of simulated scenarios, comparing it to the results from some chosen benchmark approaches. Section 5 shows the numerical performance of the new approach with a motivating example of identifying environmental factors affecting atmospheric corrosion of a metal alloy. Conclusions are drawn in Section 6.

2. Related works

This section covers the existing VS approaches for the GP regression. A popular VS method for the GP is based on the ARD approach (Williams and Rasmussen, 1996). In that approach, the length scale parameters of a covariance function are used to determine the relevance of input variables to the response variable. For example, a popular ARD covariance function is the squared exponential covariance function in the form of

\[ c_{\text{ARD}}(x_1, x_2) = \sigma_f^2 \exp \left\{ -\sum_{j=1}^{p} w_j (x_{1j} - x_{2j})^2 \right\}, \]

where \( \sigma_f^2 \) is the overall variance, \( x_{1j} \) is the \( j \)th element of the input vector \( x_1 \), and \( w_j \geq 0 \) is the inverse of the length scale parameter associated with the \( j \)th input. The inverse length scale \( w_j \) is also referred to as the relevance parameter of the \( j \)th input, because a smaller \( w_j \) value is favored to maximize a likelihood function when the \( j \)th input variable is more independent of the response variable. When \( w_j \) is zero, the \( j \)th input would have no effect on the response variable. Numerically, the likelihood maximization does not give zero \( w_j \) values. For the variable selection purpose, a sparse prior can be posed to induce more zero values on the relevance parameters. Popular sparsity priors are spike-and-slab prior (Savitsky et al., 2011) and horseshoe prior (Vo and Pati, 2017). The resulting Bayesian VS requires computationally expensive MCMC samplings.

Another popular approach is a VS based on ranking input variables by their relevance to the response variable. Some Kullback–Leibler (KL) divergence and conditional probabilities are used as a measure of relevance. Piironen and Vehtari (2016) evaluated the KL divergence of the posterior distributions (of the response value) for a full GP model (containing all input variables) and a reduced model (containing a subset of the input variables). The reduced model grows iteratively through a forward stepwise selection of input variables, starting with an empty model and adding to the model one input variable every iteration that improves the KL divergence most. Paananen et al. (2019) evaluated the relevance of each input variable to the response variable using a sensitivity measure. The sensitivity measure is defined as the degree of change in the posterior distribution of the response value under a small perturbation in each input dimension, and the degree of change is quantified by the KL divergence of the posterior distributions before and after the small perturbation. The same paper proposed another relevance measure, based on the variability of the posterior mean prediction of the response variable under a small perturbation of each input dimension. These rank measures were used to determine the relevance of the input variables to the response variable, but determining how many of the input variables are selected was not discussed in those papers.

There have been trials to generalize the ARD approach with a broader class of covariance forms. Let \( d_{\text{ARD}}(x_1, x_2) = [(x_1 - x_2)^T D(x_1 - x_2)]^{1/2} \) represent the ARD distance, where \( D \) is a diagonal matrix with \( \omega_j \) being the \( j \)th diagonal element. The covariance \( c_{\text{ARD}}(x_1, x_2) \) follows a general form, \( c_{\text{ARD}}(x_1, x_2) = c_{\text{iso}}(d_{\text{ARD}}(x_1, x_2)) \), where \( c_{\text{iso}}(d) \) refers to a stationary covariance function. For example, \( c_{\text{iso}}(d) = \sigma_f^2 \exp(-d^2) \) is the square exponential form, \( c_{\text{ARD}} \) is exactly same as \( c_{\text{ARD}} \). Generally speaking, \( c_{\text{iso}} \) can be chosen as other stationary covariance forms such as the exponential covariance \( c_{\text{exp}}(d) = \sigma_f^2 \exp(-d) \), where \( \sigma_f^2 \geq 0 \) is the variance parameter. It can be the Matérn covariance in the form of

\[ c_{\text{Matérn}}(d) = \sigma_f^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left( \sqrt{2\nu \frac{d}{\rho}} \right)^\nu K_\nu \left( \sqrt{2\nu \frac{d}{\rho}} \right), \]

where \( \Gamma(\nu) \) is the Gamma function, \( K_\nu \) is the Bessel function of the second kind, and \( \nu > 0 \) and \( \rho > 0 \) are the covariance parameters.

Moreover, the ARD distance can be generalized to a more flexible form. Rusmassen and Williams (2005, Chapter
5) discussed in their book the factor analysis distance, 
\[ d_{fa}(x_1, x_2) = \{(x_1 - x_2)^T (\Lambda \Lambda^T + D)(x_1 - x_2)\}^{1/2} \]
where \( \Lambda \) is a \( p \times q \) matrix, \( q < p \), and \( D \) is a \( p \times p \) diagonal matrix of positives, and the distance can be combined with a stationary covariance function to define a new covariance model, \( c_{fa}(x_1, x_2) = c_{iso}(d_{fa}(x_1, x_2)) \).

The authors stated that the \( q \) columns of \( \Lambda \) could identify a few projection directions of the original inputs that are highly relevant to the response variable. However, there is an identifiability issue with \( \Lambda \), because \( \Lambda \Omega \) for an arbitrary orthonormal matrix \( \Omega \) (including all rotation matrices) would achieve the same distance, and there are infinitely many versions of \( \Lambda \Omega \) with different column directions that achieve the same factor distance. Tripathy et al. (2016) proposed the active subspace distance, 
\[ d_{as}(x_1, x_2) = \{(x_1 - x_2)^TV^TDV(x_1 - x_2)\}^{1/2}, \]
where \( V \) is a \( q \times p \) projection matrix with \( VV^T = I \) and \( D \) is a diagonal matrix of positives. In this parameterization, the projection matrix \( V \) defines a low-dimensional projection of the input features, and the diagonal matrix \( D \) defines the weights on the input features. When the diagonal elements of \( D \) are all distinct, the columns of the matrix \( V \) are uniquely identified. The authors combined the Matérn 32 covariance with the active subspace distance. The iterative optimization for \( V \) and \( D \) is proposed based on the marginal likelihood maximization criterion. Since \( V \) is an orthogonal matrix, optimizing for \( V \) involves a complex orthogonality-preserving iteration based on the Cayley transform (Wen and Yin, 2013). This approach is useful for the DR. Sparsifying \( V \) for the VS while preserving the orthogonality is not straightforward.

Both of the general ARD modeling approaches are in the form of \( c_{iso}(d_{s}(x_1, x_2)) \), where
\[ d_{s}(x_1, x_2) = \{(x_1 - x_2)^T Q(x_1 - x_2)\}^{1/2}, \]
where the \( p \times p \) matrix \( Q \) should be positive semi-definite for \( d_{s} \) being a proper distance satisfying positivity and triangle inequalities. The distance \( d_{s} \) is referred to as the Mahalanobis distance or generalized L2 distance in the literature (Chandra, 1936). They are different in how they factorize the parameter matrix \( Q \). The model based on \( d_{as} \) uses the low-rank factorization \( Q = (\Lambda \Lambda^T + D) \) with a \( p \times q \) matrix \( \Lambda \) and a diagonal matrix \( D \), as in Rusmussen and Williams (2005), whereas the one with \( d_{as} \) uses the spectral decomposition \( Q = V^TDV \) with \( V \) has \( p \times p \) right-orthogonal matrix \( V \), as in Tripathy et al. (2016). The constraints on the factor matrices such as orthogonality and triangularity make the optimization of the factor matrices difficult. Adding the sparsity constraint to the already complicated optimization is not easy. Therefore, they are not appropriate models for variable selection. Here in this article, we consider a sparse factorization of the \( Q \) for VS and propose an efficient numerical optimization to jointly optimize the sparse factor matrix and other parameters for efficient VS in the GP regression.

3. GP regression with a sparse low-rank projection

To precisely describe the proposed idea consider a general regression problem of estimating an unknown regression function \( f \) that relates a \( p \)-dimensional input \( x \in \mathbb{R}^p \) to a real response \( y \), using noisy observations \( D = \{(x_i, y_i), i = 1, \ldots, N\} \).
\[ y_i = f(x_i) + \epsilon_i, \quad i = 1, \ldots, N, \]
where \( \epsilon_i \sim \mathcal{N}(0, \sigma^2) \) is white noise, independent of \( f(x_i) \). In the GP regression, the underlying regression function \( f \) is assumed to be a realization of a GP with zero mean and covariance function \( c_\epsilon \). In this article, we consider a stationary covariance in the form of \( c_\epsilon(x_i, x_j) = c_{iso}(d_{s}(x_i, x_j); \theta) \), where \( c_{iso} \) is a stationary covariance, and \( \theta \) is the parameter(s) of the stationary covariance, and \( d_{s} \) is in the form of (1) that depends on a positive definite parameter matrix \( Q \). In this article, we consider a simpler factorization of the parameter matrix,
\[ Q = S^T S, \]
where \( S \) is a \( q \times p \) real matrix, that is not required to be an orthogonal matrix or an upper trapezoidal triangular Cholesky factor. Without the orthogonality and upper triangularity constraints, the matrix \( S \) is not uniquely identified as in the low-rank factorization used by Rusmussen and Williams (2005), because \( Q \Omega S \) for an arbitrary \( q \times q \) orthonormal matrix \( \Omega \), also gives the same form of the factorization. Among infinitely many \( S \) that factorizes \( Q \) in the form of (2), we seek a sparse factor \( S \) that satisfies
\[ TR(S) \leq \mu, \]
where \( TR(S) = ||S||_R \) is the sparsity norm on \( S \) for \( r \leq 1 \). We have two reasons for placing the sparsity constraint. The sparsity constraint resolves the identifiability issue, and more importantly, it is hoped that \( d_{s} \) only depends on a very few variables of the \( p \) original inputs for a better interpretation of the GP regression result. Please note that with the factorization, the distance \( d_{s} \) can be written as
\[ d_{s}(x_i, x_j) = ||S(x_i - x_j)||_2, \]
where \( || \cdot ||_2 \) is the L2 norm. The matrix \( S \) projects the \( p \) original inputs to \( q \)-dimensional features. If the projection matrix is sparse, one can have each of the projection features as a linear combination of only a very few original inputs. Below we propose a numerical optimization for jointly optimizing the sparse \( S \) and other covariance parameters.

To describe the solution approach to optimize the parameters, we introduce a common set of notations. We denote the collection of observed input locations, \( X = [x_1, \ldots, x_N]^T \), and we denote the collection of observed response variables, \( y = [y_1, \ldots, y_N]^T \). With the GP prior, the prior distribution of \( f = [f(x_1), \ldots, f(x_N)]^T \) is the multivariate normal distribution, \( f|X, S, \theta \sim \mathcal{N}(0, C_{S, \theta}) \), where \( C_{S, \theta} \) is an \( N \times N \) matrix with \((i,j)\)th entry \( c_{\theta}(x_i, x_j) \). The conditional distribution of \( y \) is \( y|f, \sigma^2 \sim \mathcal{N}(f, \sigma^2 I) \). Let \( \phi_C = \{\theta, \sigma^2\} \) present a set of the covariance parameters and the noise variance parameter. The marginal distribution of \( y \) given \( X, \phi_C \) and the distance parameter \( S \) can be derived as a multivariate normal distribution,
\[ y|X, S, \phi_C \sim \mathcal{N}(0, \sigma^2 I + C_{S, \theta}). \]
The parameter set, \( \phi_C \) and \( S \), are jointly optimized by minimizing the negative log likelihood function,
\[ \mathcal{L}(S, \phi_C) = \frac{1}{2} y^T (\sigma^2 I + C_{S, \theta})^{-1} y + \frac{1}{2} \log |\sigma^2 I + C_{S, \theta}| \]  
(4)

under a sparsity constraint on \( S \),

\[ \mathcal{R}(S) \leq \mu \]

or equivalently its Lagrange relaxation is solved

Minimize \[ \mathcal{L}(S, \phi_C) + \lambda \mathcal{R}(S), \]  
(5)

where \( \lambda > 0 \) is the Lagrange multiplier. The solution depends on a choice of two tuning parameters, the rank parameter \( q \) and the sparsity parameter \( \lambda \). We will discuss a numerical optimization of problem (5) for a choice of the tuning parameters in Section 3.1, and some technical details of the optimization are in the Online Supplementary Material - Appendix A. The choice of the tuning parameters will be covered in Section 3.2.

### 3.1. FSEG: Forward stagewise with embedded gradient descent steps for parameter estimation

In this section, we present a numerical approach to solve problem (5) for the joint estimation of the covariance parameter \( \phi_C \) and the distance parameter \( S \). The objective function of the problem consists of the likelihood term \( \mathcal{L} \) and the \( r \)-norm sparsity penalty term \( \mathcal{R} \). A sparsity penalized optimization problem has been studied in different problem settings. When the likelihood term is in a quadratic form and the penalty term is a \( 1 \)-norm, the problem is known as the Lasso problem. The forward stagewise regression algorithm was successful in solving the Lasso problem (Efron et al., 2004). The approach was later generalized by Zhao and Yu (2007) for a convex likelihood term (or empirical loss) with the \( 1 \)-norm penalty (B-Lasso) and a convex likelihood with a convex penalty function (the generalized B-Lasso). The major advantage of using the forward stagewise regression approach is that it generates the solution path containing the solutions over a wide range of \( \lambda \) values, so the selection of the sparsity parameter \( \lambda \) can be done by evaluating the solutions in the path with a model selection criterion. All of the convergence proofs in the existing works are based on the strong convexity or convexity assumption on the objective function including the likelihood and penalty term. For our problem (5), the likelihood term is non-convex, and the penalty term is also non-convex for \( r < 1 \). Therefore the convergence results in the past works are not applicable. In general, the forward stagewise method and its variants belong to a steepest coordinate descent method, which does not provide a guarantee of convergence to local optimality for general non-convex objection functions (Nuttini et al., 2013), for which a gradient descent method with the full gradient provides a better convergence. However, the gradient descent approach numerically does not give a sparse solution even with a large \( \lambda \) value, and a numerical truncation of the outcome is necessary. The iterative soft-shrinkage algorithm (ISTA) may be considered as an alternative. However, the proximal gradient approach developed for the Lasso problem with a convex loss function plus the Lasso penalty does not provide a guarantee of the convergence for non-convex objective functions. Here we propose a combination of the forward stagewise approach with a gradient descent method, which basically runs the forward stagewise iterations with embedded gradient steps to complement the limited convergence of the coordinate descent steps. The approach inherits the good features of the conventional forward stagewise approaches, i.e., providing the solution path for different \( \lambda \) values. The new approach is referred to as the forward stagewise with embedded gradient descent step or in short FSEG.

To describe the approach, let \( \phi \) denote a large vector concatenating the elements of \( S \) and \( \phi_C \) with its initial \( q \times p \) elements from \( S \) and the remaining elements from \( \phi_C \), and let \( J \) denote the total number of the elements in the large vector. Consider a problem of finding \( \phi \) that minimizes

Minimize \[ \Gamma(\phi; \lambda) = \mathcal{L}(S, \phi_C) + \lambda \mathcal{R}(S), \]  
(6)

where \( \mathcal{L} \) is a non-convex function, and \( \mathcal{R} \) is concave for \( r < 1 \) (convex for \( r = 1 \)). Here the \( \phi \) vector is unconstrained, even though some covariance parameters should have positive values, e.g., \( \sigma^2 \). For those positively constrained parameters, we include the logarithms of the parameters in the \( \phi \) vector, so the \( \phi \) vector can be still unconstrained. We like to generate the solution path of the problem, including the local minimum of \( \Gamma(\phi; \lambda) \) for each value of \( \lambda \) ranging from zero to infinity, where the solution path implies a series of the solutions of problem (6),

\( \phi^{(t)}, \lambda^{(t)}; t = 1, 2, \ldots \),

where \( \phi^{(t)} \) denotes the \( t \)-th solution achieved with \( \lambda = \lambda^{(t)} \). The initial solution \( \phi^{(0)} \) is set to one obvious minimum for \( \lambda^{(0)} = \infty \). When the \( \lambda \) value is infinity, the optimal solution of (6) would be \( S = 0 \) and \( \phi_C = \arg \min_\phi \mathcal{L}(0, \phi_C) \). Otherwise, the objective value of (6) would diverge to \( \infty \), because of the infinite \( \lambda \mathcal{R}(S) \) term. We start with the initial solution, and update the solution iteratively to other solutions, using the following forward stagewise steps. A forward stagewise regression belongs to a coordinate descent algorithm, which iteratively updates the solution along a chosen coordinate direction with a small step size \( \epsilon \). A coordinate descent step can be written as

\[ \phi^{(t+1)} = \phi^{(t)} + s e_j, \]

where \( |s| = \epsilon \), and \( e_j \) is a \( J \times 1 \) vector of all zeros except for the \( j \)-th element being one. The \( j \) indicates the variable to be updated, and \( s \) defines the direction and magnitude of the update. First try the coordinate descent direction on \( \Gamma \) for \( (j, s) \),

\[ (j_0, s_0) = \arg \min_{j \in \{1, \ldots, J\}, s \epsilon} \Gamma(\phi^{(t)} + s e_j; \lambda^{(t)}). \]

Following this coordinate descent direction would make an improvement of \( \Gamma \) by making a little change in one coordinate of \( S \) or \( \phi_C \). If the improvement is more than or equal to a small tolerance parameter \( \zeta \),

\[ \Gamma(\phi^{(t)}; \lambda^{(t)}) - \Gamma(\phi^{(t)} + s_0 e_{j_0}; \lambda^{(t)}) \geq \zeta, \]  
(7)

we take the coordinate direction to update the solution,
\[ \phi^{(t+1)} = \phi^{(t)} + s_t e_{j_t}, \]  
(8)

and keep \( \lambda^{(t+1)} = \lambda^{(t)} \). Otherwise, \( \Gamma \) can only be reduced to a minor extent, along any coordinate directions for the current \( \lambda \) value. This implies one of two scenarios: (i) the iteration is close to a local minimum of \( \Gamma \) for the current \( \lambda \), or (ii) \( \Gamma \) would not improve along any of the coordinate directions, although the current solution \( \phi^{(t)} \) is far from a local minimum, i.e., the coordinate descent steps were stuck in the middle of the path to a local minimum. The latter case may happen for non-convex objective functions, due to the direction of the update in a coordinate descent step being restricted to one coordinate direction at a time, and any of the coordinate directions may not give any improvement in \( \Gamma \), for which the numerical iteration stops possibly before reaching a local minimum. To escape from being stuck, we relax the improvement direction from the coordinate-wise direction to the support-limited gradient by running one gradient descent step,

\[ \phi^{(t+1/2)} = \phi^{(t)} - \varepsilon_g \nabla \text{supp} \Gamma(\phi^{(t)}; \lambda^{(t)}), \]  
(9)

In Equation (9), the fractional step number \((t + 1/2)\) is used to index the intermediate solution \( \phi^{(t+1/2)} \), \( \nabla \text{supp} \Gamma(\phi^{(t)}; \lambda^{(t)}) \) is the support-limited gradient of \( \Gamma \) evaluated at \( \phi^{(t)} \), and the step size \( \varepsilon_g \) can be chosen using a line search. Here “support” implies the support of the solution, \( \text{supp}(\phi^{(t)}) = \{j = \{1, \ldots, J\}; e_j^T \phi^{(t)} \neq 0\} \), and the “support-limited” implies that the \( j \)th element of the gradient vector \( \nabla \text{supp} \Gamma(\phi^{(t)}; \lambda^{(t)}) \) is set to zero if \( j \notin \text{supp}(\phi^{(t)}) \); more details of the support-limited gradient can be found in Appendix A. As we discuss in Appendix A, the support-limited gradient is well defined. This support-limited update finds the update along a combination of multiple coordinates belonging to the support, instead of one coordinate direction, so it finds improvement directions that are not considered in the coordinate descent. On the other hand, the support of the solution with the update remains the same as that of \( \phi^{(t)} \), so the sparsity is maintained unlike in the conventional gradient descent with the full gradient. If the result of the gradient step satisfies

\[ \Gamma(\phi^{(t)}; \lambda^{(t)}) - \Gamma(\phi^{(t+1/2)}; \lambda^{(t)}) \geq \zeta, \]  
(10)

we take the result,

\[ \phi^{(t+1)} = \phi^{(t+1/2)}. \]  
(11)

Otherwise, the \( \Gamma \) value cannot be further reduced with the current \( \lambda \) value. In order to reduce the \( \lambda \) value, we take the following forward stagewise step

\[ \phi^{(t+1)} = \phi^{(t)} + s_t e_{j_t}, \]  
(12)

where the coordinate descent direction is chosen to improve the non-penalized likelihood term \( L \) among the first \( q \times p \) coordinates of \( \phi \),

\[(j_t, s_t) = \arg\min_{j \in (1, \ldots, J), s \in \mathbb{S}^q} \Gamma(\phi^{(t)} + s e_j; \lambda = 0).\]

In Appendix B, we discuss the effectiveness of the support-limited gradient step with numerical examples. Summarizing these results, in many numerical cases, the algorithm with no support-limited gradient step works well, but it can perform very poorly for some worst case scenarios, e.g., high noise cases. For those cases, the support-limited gradient step can make a significant improvement in variable selection.

The \( \lambda \) value changes from \( \lambda^{(t)} \) to

\[ \lambda^{(t+1)} = \min \left\{ \lambda^{(t)}, \frac{L(S^{(t)}; \phi^{(t)}) - L(S^{(t+1)}; \phi^{(t+1)}; \lambda)}{\mathcal{R}(S^{(t+1)}) - \mathcal{R}(S^{(t)})} - \zeta \right\}, \]

(13)

where \( S^{(t)} \) and \( \phi^{(t)} \) are the corresponding parts of \( \phi^{(t)} \). The step reduces the \( \lambda \) value to the smallest value that does not worsen the objective function \( \Gamma \). If \( \lambda^{(t+1)} > 0 \), the iteration continues. Otherwise, the iteration stops, because the \( \lambda \) value cannot be further reduced. The whole algorithm is summarized in Algorithm 1.

**Algorithm 1:** Forward stagewise with embedded gradient descent step (FSEG)

**Input:** rank \( q \), step size \( \varepsilon \), tolerance parameter \( \zeta \), maximum number of iterations \( T_{\text{max}} \)

**Output:** solution path \( \{\{\phi^{(t)}, \lambda^{(t)}\} : t = 1, \ldots, T_{\text{max}}\} \)

1. **Initialization.** Set the initial solution \( \phi^{(0)} \) with \( S^{(0)} = 0 \) and

\[ \phi^{(0)} = \arg\min \{L(S^{(0)}; \phi) \}, \quad \phi^{(0)} = \arg\min_{\phi^{(0)}} \{L(S^{(0)}; \phi) \}. \]

Set the initial value of \( \lambda \) to \( \lambda^{(0)} = \infty \).

2. **for** \( t = 0; T_{\text{max}} \) **do**

3. **Find** the coordinate descent direction on \( \Gamma \),

\[(j_t, s_t) = \arg\min_{j \in (1, \ldots, J), s \in \mathbb{S}^q} \Gamma(\phi^{(t)} + s e_j; \lambda^{(t)}).\]

4. **if** \( \Gamma(\phi^{(t)}; \lambda^{(t)}) - \Gamma(\phi^{(t)} + s_t e_{j_t}; \lambda^{(t)}) \geq \zeta \) **then**

5. **Update** \( \phi^{(t+1)} = \phi^{(t)} + s_t e_{j_t}. \)

6. **Update** \( \lambda^{(t+1)} = \lambda^{(t)} \).

7. **else**

8. **Take** the support-limited gradient descent step (9) to achieve \( \phi^{(t+1/2)} \). If \( \Gamma(\phi^{(t)}; \lambda^{(t)}) - \Gamma(\phi^{(t+1/2)}; \lambda^{(t)}) \geq \zeta \) **then**

9. **Set** \( \phi^{(t+1)} = \phi^{(t+1/2)} \) and continue.

10 **else**

11. **Take** the following forward stagewise step,

\[ \phi^{(t+1)} = \phi^{(t)} + s_t e_{j_t}, \]

where the coordinate descent direction and step size are determined as

\[(j_t, s_t) = \arg\min_{j \in (1, \ldots, J), s \in \mathbb{S}^q} \Gamma(\phi^{(t)} + s e_j; \lambda = 0).\]

12. **Update** \( \lambda^{(t+1)} = \min \left\{ \lambda^{(t)}, \frac{L(S^{(t+1)}; \phi^{(t+1)}; \lambda)}{\mathcal{R}(S^{(t+1)}) - \mathcal{R}(S^{(t)})} - \zeta \right\}. \]

13 **Stop** the procedure.

The algorithm draws a monotone sequence of the solutions in terms of the values \( \Gamma(\phi; \lambda) \) with variable step greater than or equal to \( \zeta \) as described in Theorem 1.

**Theorem 1.** For any iteration \( t \), the objective function \( \Gamma \) value is improved by at least \( \zeta \).
\[ \Gamma(\phi^{(t+1)}; z^{(t+1)}) \leq \Gamma(\phi^{(t)}; z^{(t)}) - \xi. \]

\textbf{Proof.} The iteration \( t \) updates \( \phi^{(t+1)} \) and \( z^{(t+1)} \), following one of two possible paths. When the condition (10) holds, the iteration follows (11) and \( z^{(t+1)} = z^{(t)} \). Otherwise, the iteration follows (12) and (13). It is obvious to show that the update (11) with \( z^{(t+1)} = z^{(t)} \) satisfies \( \Gamma(\phi^{(t+1)}; z^{(t+1)}) \leq \Gamma(\phi^{(t)}; z^{(t)}) - \xi \). Here, we focus on proving that the updates (12) and (13) also satisfy the condition. If

\[ \frac{L(S^{(t)}; \phi^{(t)}, \phi^{(t)}_C) - L(S^{(t+1)}, \phi^{(t+1)}_C)}{R(S^{(t+1)} - R(S^{(t)}) \leq \lambda^{(t)} \]

in the lambda update, we have

\[ \begin{cases} 
\lambda^{(t+1)} = \frac{L(S^{(t)}, \phi^{(t)}_C)}{R(S^{(t)}) - R(S^{(t)})} - \xi, \quad \text{and} \\
\lambda^{(t+1)} \leq \lambda^{(t)}. 
\end{cases} \]

The first condition in (14) gives

\[ \begin{align*}
&\lambda^{(t+1)}(R(S^{(t)} - R(S^{(t)})) = L(S^{(t)}, \phi^{(t)}_C) - L(S^{(t)}, \phi^{(t)}_C) - \xi \\
&\Leftrightarrow \lambda^{(t+1)} = L(S^{(t)}, \phi^{(t)}_C) + \lambda^{(t+1)}R(S^{(t)}) - \xi \\
&\Leftrightarrow \Gamma(\phi^{(t)}; \lambda^{(t+1)}) = L(S^{(t)}, \phi^{(t)}_C) + \lambda^{(t+1)}R(S^{(t)}) - \xi.
\end{align*} \]

Since \( \lambda^{(t+1)} \leq \lambda^{(t)} \),

\[ \begin{align*}
\Gamma(\phi^{(t)}; \lambda^{(t+1)}) &= L(S^{(t)}, \phi^{(t)}_C) + \lambda^{(t+1)}R(S^{(t)}) - \xi \\
&\leq L(S^{(t)}, \phi^{(t)}_C) + \lambda^{(t)}R(S^{(t)}) - \xi \\
&= \Gamma(\phi^{(t)}; \lambda^{(t)}) - \xi.
\end{align*} \]

Otherwise, if \( \frac{L(S^{(t)}, \phi^{(t)}_C) - L(S^{(t+1)}, \phi^{(t+1)}_C) - \xi}{R(S^{(t+1)} - R(S^{(t)})} > \lambda^{(t)} \),

\[ \Gamma(S^{(t+1)}; \lambda^{(t+1)}) < \Gamma(S^{(t)}; \lambda^{(t)}) - \xi. \]

Since \( \lambda^{(t+1)} = \lambda^{(t)} \),

\[ \Gamma(S^{(t+1)}; \lambda^{(t+1)}) < \Gamma(S^{(t+1)}; \lambda^{(t+1)}) - \xi. \]

Since the solution sequence is monotone with respect to the corresponding objective value and the objective function is bounded below by zero, the sequence converges in a finite number of iterations by the bounded convergence theorem.

### 3.2. Tuning parameter selection

There are two tuning parameters, the sparsity parameter \( \lambda \) and the rank parameter \( q \). For a choice of \( q \in \{1, \ldots, Q_{\text{max}}\} \), the proposed FSEG iterates the solutions to problem (6) for different choices of the \( \lambda \) value. Let us denote the solution path for a choice of \( q \) by

\[ \{(\phi^{(t)}_q, z^{(t)}); t = 1, \ldots, T_{\text{max}}\}. \]

The solutions in the path contain very sparse solutions (in earlier iterations corresponding to larger \( \lambda \) values) to very dense solutions (in later iterations corresponding to smaller \( \lambda \) values). Among them, we like to choose the best solution with respect to a model selection criterion, the Bayesian Information Criterion (BIC). We evaluate the BIC for each solution in the solution path,

\[ BIC(\phi_q^{(t)}, z_q^{(t)}) = 2L(\phi_q^{(t)}, z_q^{(t)}) + \|\phi_q^{(t)}\|_0 \log(N), \]

where \( \| \cdot \|_0 \) is the 0-norm, and choose the solution that gives the smallest BIC value,

\[ t_q = \arg \min_{t=1, \ldots, T_{\text{max}}} BIC(\phi_q^{(t)}, z_q^{(t)}). \]

Then, the \( z_q^{(t_q)} \) corresponds to the \( \lambda \) value that corresponds to the chosen solution. We also tried to choose the value of the rank parameter \( q \), based on the BIC criterion,

\[ q' = \arg \min_{q=1, \ldots, Q_{\text{max}}} BIC(z_q^{(t_q)}, z_q^{(t_q)}). \]

However, numerically, the BIC-based choice has shown a tendency of overestimating \( q \). For the overestimated values, the corresponding choice of \( S \) was very sparse in many rows, in that many rows have only one non-zero element, for which the overall 0-norm values \( \|\phi_q^{(t)}\|_0 \) are not much different for smaller choices of \( q \). Therefore, for choosing \( q \), we modified the BIC criterion (15) slightly to

\[ mBIC(\phi_q^{(t)}, z_q^{(t)}) = 2L(\phi_q^{(t)}, z_q^{(t)}) + q\|\phi_q^{(t)}\|_2 \log(N), \]

where \( S_q \) is the \( S \) value of \( \phi_q^{(t)} \), and \( \| \cdot \|_2 \) is the (2,0)-matrix norm that counts the number of non-zero columns of a matrix. The modification in the BIC criterion is an empirical ad-hoc approach. The approach modification worked better in most of numerical cases, including comprehensive simulated scenarios and real data studies.

### 4. Simulated examples

This section presents the numerical performance of the proposed variable selection approach with a number of simulated scenarios. We generate 27 simulated scenarios with different settings, each of which is characterized by a unique setting of simulation input parameters. For each scenario, we perform 25 simulation runs for replicated experiments. Each of the simulation runs starts with generating a dataset for a regression analysis with \( p \) input variables, including \( p_0 \) inputs relevant to the response variable and \( p - p_0 \) irrelevant inputs. The data generation follows random sampling steps described below:

- **Inputs:** noise variance \( \sigma^2 \), covariance parameter \( \theta \), rank parameter \( q \), \( p \), and \( p_0 \).
- **Outputs:** \( N \) records of input variables and response variable, \( X, y \)
- **Step 1.** Take an \( N \times p \) input matrix \( X = (x_1, x_2, \ldots, x_N)^T \) with each row \( x_i \sim \text{Uniform}([0,1]^p) \) independently for \( i = 1, \ldots, N \).
- **Step 2.** Sample the distance parameter \( S \) as follows. Let \( A \) denote a \( p_0 \times q \) random matrix with each of the elements independently sampled from \( N(0,1) \). Take the QR decomposition, \( A = O_{p_0} R \), where \( O_{p_0} \) is a \( p_0 \times p_0 \) orthonormal matrix and \( R \) is a \( p_0 \times q \) upper triangular matrix, and take a \( q \times p_0 \) submatrix \( O_{q_0} \), made of the first \( q \) rows of the orthonormal matrix for \( q < p_0 \). Sample a \( q \)
$\times q$ diagonal matrix $D$ with each diagonal element independently sampled from an inverse gamma distribution, $\text{Gamma}^{-1}(1,1)$. Set $S_q = DO_q$ and augment the $q \times p_0$ matrix $S_q$ to a $q \times (p - p_0)$ zero matrix. Randomly reorder the columns of the augmented matrix, which is set to $S$.

**Step 3.** Given $S$ from the previous step, we define a covariance function,

$$c_S(x_i, x_j) = c_{\exp}(d_S(x_i, x_j); \theta),$$

where $c_{\exp}$ is an exponential covariance function with variance parameter $\theta$. Sample $y | X, S, \theta, \sigma^2 \sim N(0, \sigma^2 I + C_{S, \theta})$.

We fix $p = 10$ and vary $p_0 \in \{3, 5, 7\}$. We also try different values of $\sigma^2 \in \{0.1^2, 0.3^2, 0.5^2\}$, while fixing the signal variance $\theta = 1$, which would create different signal-to-noise ratios. We also vary the rank parameter $q \in \{1, 2, 3\}$. The number of possible combinations of the $p_0, \sigma^2$ and $q$ values is 27, and one unique setting serves as a simulation scenario. For each scenario, we perform 25 replicated simulation runs by generating 25 datasets, and the outcomes reported in this section are the statistics of the 25 outcomes, the mean and standard deviation. We first report an in-depth analysis of the outcomes from the proposed approach in Sections 4.2 and 4.3. Section 4.4 reports the comparison to three benchmark variable selection approaches, including the KL-divergence-based forward stepwise selection approach (Piironen and Vehtari, 2016, KL-F), KL-divergence-based sensitivity analysis (Paananen et al., 2019, KL-S), and variability-of-the-posterior-mean approach (Paananen et al., 2019, VAM). We have not included the comparison to the MCMC sampling approach (Savitsky et al., 2011), mainly due to its computational slowness.

In the simulation study, we apply the 1-norm sparsity penalty and set $T_{\max} = 100$, $\epsilon = 10^{-3}$ and $\xi = 10^{-6}$ for our approach. For all three benchmark approaches, we use the BIC to choose the number of variables selected.

### 4.1. Numerical illustration of the proposed FSEG algorithm

For a given choice of $q^*$, the FSEG algorithm generates the solutions of problem (5) and corresponding $\lambda$ values, $\{S^{(t)}, \phi^{(t)}_C, \lambda^{(t)}; t = 1, 2, \ldots\}$. We evaluate the BIC criterion (15) of each of the solutions to choose the best solution that balances the sparsity and the fit to data. We previously denoted the selected solution by $S^{(t*)}$ in Section 3.2. In this section, we illustrate how that works with a simulation scenario with $p_0 = 5$, $q = 1$ and $\sigma^2 = 0.5^2$. 

![Figure 1. Solution path of the proposed FSEG approach for a simulation scenario with $p_0 = 5$, $q = 1$ and $\sigma^2 = 0.5^2$.](image)
Our proposed approach has two tuning parameters: the rank parameter \( q \) and the sparsity parameter \( \lambda \). The rank \( q \) determines the rank of \( Q \) in the distance \( d_s \) or equivalently the row size of the matrix \( S \), and the sparsity parameter \( \lambda \) determines the number of zero elements in the projection matrix \( S \), which is related to the number of variables selected. We first analyze the choice of \( q \) for the simulated scenarios in this section. We know the values of \( q \) used to generate simulation scenarios, which are compared to the estimated \( q^* \) achieved using the model selection approach described in Section 3.2. Table 1 contains the percent splits. Since the accuracy does not depend significantly on \( \sigma^2 \) and \( p_0 \), we report the percents for each distinct \( q \) and \( q^* \) combination. The overall bias of the estimation can be achieved by taking the mean of the observed \( q^* - q \) values over \( 27 \times 25 \) runs, which was -0.0148. The percent with \( q = q^* \) is 79.63%, and the percent of \( |q - q^*| \leq 1 \) is 95.56%.

We would also like to highlight the sensitivity of the proposed algorithm to the choice of the other two input parameters: tolerance parameter \( \xi \) and step size \( \epsilon \). The two parameters occur in almost all iterative optimization algorithms. Their values are often set to, for example, 10^-3 and 10^-6 for all simulated scenarios. The sensitivity of the algorithm’s performance to the choices of the parameters is not very high. For example, for the simulation scenario with \( q = 2 \), \( \sigma^2 = 0.3^2 \) and \( p_0 = 5 \), changing \( \epsilon \) from \( 10^{-4} \) and \( 10^{-2} \) did not change the variable selection result. For the same simulation setting, changing \( \xi \) from \( 10^{-9} \) to \( 10^{-3} \) did not change the result.

### 4.3. Analysis of the estimated projection matrix

In this section, we analyze how the estimated projection matrix \( S^{(q^*)} \) compares with the ground truth, i.e., the value of \( S \) used for simulation data generation. We calculated the Frobenius norm of the ground truth and the estimated one, which quantifies the Mean Square Error (MSE) of the estimate. Before the calculation, we reordered the rows of \( S^{(q^*)} \) so that the row-reordered matrix matches best to \( S \). The row reordering is necessary for comparing the two matrices, because the row reorder does not make any changes in both of the marginal likelihood and the sparsity penalty, so the \( S^{(q^*)} \) estimated by the proposed FSEG could have a different row ordering.

Figure 2 summarizes the variations of the MSE values for different values of \( q \) and \( p_0 \); the numbers for individual simulated scenarios can be found in the Online supplementary material - Appendix C. Both the mean and standard deviations did not vary much in \( p_0 \) and \( \sigma^2 \), but they changed significantly with \( q \). For a higher rank \( q \), there are more errors. This is because the size of \( S \) is proportional to \( q \), and there are many error sources involved for estimating a larger matrix. To better illustrate this, we chose two simulation scenarios, one with \( q = 2 \) and another with \( q = 3 \) and compared the estimated \( S^{(q^*)} \) with the ground truth \( S \). Figures 3 and 4 show the comparison. As seen in Figure 3, the estimated projection matrix is very close to the ground truth for \( q = 2 \). We see a larger estimation bias for \( q = 3 \). However, as seen in Figure 4, the bias is not significant. The estimated matrix values largely follow the trends of the ground matrix values, but they have some minor differences. This shows the proposed approach is still effective for \( q = 3 \).

### 4.4. Hit-and-miss of relevant variables

For each simulation scenario, we also analyze the variables identified by the proposed approach, which are compared to the set of \( p_0 \) relevant variables used in the data generation procedure (regarded as the ground truth). The variables identified by the proposed approach are achieved as the variables corresponding to the non-zero columns in the estimated \( S^{(q^*)} \). Let \( A \) denote the set of \( p_0 \) relevant variables used in the simulation data generation, and let \( \hat{A} \) denote the set of the variables identified by the proposed approach. We count the false negative rate (FNR) and the faulty positive rate (FPR) error of \( \hat{A} \) versus \( A \).

\[
\begin{align*}
\text{FNR} & = \frac{|A - \hat{A}|}{|A|} \\
\text{FPR} & = \frac{|\hat{A} - A|}{10 - |A|},
\end{align*}
\]

where \(|\cdot|\) denotes the set cardinality, and - is a set difference operator. The FNR and FPR values are calculated, and the means and standard deviations of the two values are taken over 25 simulation runs for each of the simulated scenarios. The same evaluations are performed for some chosen benchmark methods, including the KL-divergence-based forward stepwise selection approach (Piironen and...
Vehtari, 2016, KL-F), KL-divergence-based sensitivity analysis (Paananen et al., 2019, KL-S), and variability-of-the-posterior-mean approach (Paananen et al., 2019, VAM). The individual statistics are reported in the supplementary material - Appendix D. Figure 5 summarizes the outcomes with two box plots showing the overall FNR and FPR statistics for all 27 simulated scenarios. Typically, if the FPR value was lowered, the FNR value would increase. The graphical plot shows what approaches provide better trade-offs between the FPR and FN values. The overall FPR values are comparable among all the compared methods, which have shown more differences in the TPR values. The
The proposed approach achieves the highest TPR values (i.e., lowest FNR values) among the compared methods for most of the compared scenarios. In particular, the proposed approach exhibits a larger gap to the benchmark approaches for the scenarios with high noise variance $\sigma^2$. The proposed approach is robust to high noises.

### 4.5. Computation times

All of these numerical experiments are performed in MATLAB on a desktop computer with Intel(R) Core(TM) i5-10400T and 16 GB memory. The computation times for different simulation scenarios were similar. Here we just report the average computation times for all simulated scenarios. The computation times are 10.98, 294.45, 11.52 and 93.51 seconds for the proposed approach, KL-F, KL-S and VAM respectively. Our proposed approach is computationally competitive.

### 5. Real example: Environmental corrosion analysis

This section presents the application of the proposed VS approach to identify the environmental factors most

![Figure 5. Performance comparison of the four variable selection approaches against the ground truth. The panels (a), (c) and (e) show the box plots of the false negative rates (FNR) grouped by different noise levels, and the panels (b), (d) and (f) show the box plots of the false positive rates (FPR). The center line in each box indicates the median value of the FNR or FPR values for each of the compared methods, and the upper and lower bounds of each box represent the 75% and 25% quantiles respectively. All crosses represent the outliers.](image)
influential to metal corrosion. The outcome will be exploited
to design an accelerated corrosion testing protocol using a
custom environmental chamber that can simulate real-world
conditions including temperature, relative humidity, salt
water spray, background gases, and artificial sunlight.
Developing the protocol requires two preliminary steps: first
identifying the control factors and then calibrating the con-
tral factor levels. The benefit of this exercise will be to
reduce the number of factors to account for when conduct-
ing an experiment in the laboratory test chamber that pro-
duces similar metal corrosion to that occurring in a natural
environment. Pre-selecting a subset of more influential fac-
tors is highly desirable for a more efficient design of the
accelerated corrosion test protocol.

For the variable selection, the U.S. Air Force deployed
two measurement systems to collect necessary data, the
Corrosion & Coatings Evaluation System (CorRES™), and
the Weather Instrumentation and Specialized Environmental
Monitoring Platform (WISE-MP), shown in Figure 6(a).
Both systems were placed at a test site operated by the U.S.
Naval Research Laboratory in Key West FL. The two meas-
urement systems produced the periodic measurements of 27
environmental factors that potentially affect atmospheric
corrosion of aluminum alloy (AA) specimens attached to
the sensing systems, including different temperature meas-
urements, relative humidity, concentrations of several corro-
sive gases, and other weather conditions such as the
intensities and durations of rain, hail, and wind. A complete
list of the factors can be found in Figure 6(b). The galvanic
corrosion current flowing through the AA specimen was
also measured to quantify the degree of corrosion of the
specimen. In total, 18,016 records of the environmental fac-
tors and corrosion current measurements were collected
over a 3 month period from May 31, 2019 to August
22, 2019.

The corrosion currents were related to the environmental
factors through a GP regression model, and our approach
was applied to select a subset of the 27 environmental fac-
tors that contribute most to accurate GP modeling. To
evaluate the outcome of the GP modeling and variable selec-
tion, we randomly split the 18,016 records into two sets: a
training set for training the GP regression with the proposed
variable selection and a test set for evaluating the outcome.
The split ratio was eight to one, eight for the training set
and one for the testing set. The training set was composed of 14,411 records, for which the marginal likelihood calculation would take a very long time. We used an approximation to the marginal likelihood and the corresponding GP regression, based on the patchwork Kriging (Park and Apley, 2018). In the approximation scheme, the data is partitioned into $K$ subsets, $\{X_k, y_k\}; k = 1, ..., K$, and the approximate likelihood is defined as a sum of the likelihoods over the subsets,

$$2L_a(S, \phi_C) = \sum_{k=1}^{K} \left[ y_k^T (\sigma^2 I + C_{S,\theta}^{(k)})^{-1} y_k + \log |\sigma^2 I + C_{S,\theta}^{(k)}| \right].$$

where $C_{S,\theta}^{(k)}$ is the covariance function evaluated for the $k$th subset, $X_k$. We used $K = 40$, and the covariance function used in the simulation study is applied. We set $\epsilon = 10^{-2}, \xi = 10^{-6}$ and $T_{max} = 1000$ for the proposed FSEG, and the $q$ and $\lambda$ were chosen by the model selection described in Section 3.2. The chosen value of $q$ is two. With the approximate scheme, the total computation time for the variable selection and regression analysis was 1959 seconds on the same machine that we used for the simulation study.

Figure 7 shows the solution path for $q = 2$ and the corresponding lambda values over the first 100 iterations. The lowest BIC value was achieved at iteration $t = 80$, for which the sparsity parameter $\lambda$ was 0.6516. The solution achieving the lowest BIC value was selected as the final estimate of the GP parameters, $\phi_C$ and $S$. The estimate of $S$ provided the relevance of 27 variables to the galvanic corrosion. According to the estimate, 10 among 27 variables are relevant to the corrosion rate. The 10 relevant variables are highlighted with yellow colors in Figure 6(b), including air temperature, surface temperature, heating temperature, effective relative humidity, electrochemical impedance, rain-related weather conditions, and the concentration of $O_3$.

We evaluated the outcome of the variable selection and the corresponding GP model both quantitatively and qualitatively. For the quantitative judgment, we fit two regression models to the training dataset: one GP regression model with a full set of the 27 environmental factors and another

| Approach                     | MSE    | NLPD  |
|------------------------------|--------|-------|
| full GP with all 27 factors  | 0.0475 | -0.6035 |
| KL-F (selected 19 factors)   | 0.0826 | -0.7458 |
| KL-S (selected 15 factors)   | 0.0480 | -0.8756 |
| VAM (selected 15 factors)    | 0.0433 | -0.9241 |
| Our Approach (selected 10 factors) | 0.0434 | -1.0865 |

Table 2. Comparison of several benchmark approaches and proposed VS approaches for the environmental corrosion data.

Figure 7. Solution path of the proposed FSEG approach for the WISE-MP corrosion dataset. The $q = 2$ is chosen using the model selection procedure in Section 3.2: (a) shows the solution path generated by the FSEG, (b) shows the corresponding trace of the sparsity parameter values applied, (c) shows the BIC versus iteration $t$ with the lowest BIC value circled, and (d) shows the negative log likelihood value versus iteration $t$. 
GP model with the 10 selected factors, and we compared the prediction accuracies of the two models in terms of their posterior mean and variance estimates. For the full GP, we used the standard ARD covariance function, \( c_{\text{se,ard}} \). For comparison of prediction accuracy, we calculated two performance metrics on the test data, denoted by \( \{(x_t, y_t) : t = 1, \ldots, T\} \), where \( T \) is the test set size. Let \( \mu_t \) and \( \sigma_t^2 \) denote the estimated posterior mean and variance at location \( x_t \). The first measure is the MSE

\[
\text{MSE} = \frac{1}{T} \sum_{t=1}^{T} (y_t - \mu_t)^2,
\]

which measures the accuracy of the mean prediction \( \mu_t \) at location \( x_t \). The second measure is the Negative Log Predictive Density (NLPD)

\[
\text{NLPD} = \frac{1}{T} \sum_{t=1}^{T} \left[ \frac{(y_t - \mu_t)^2}{2\sigma_t^2} + \frac{1}{2} \log(2\pi\sigma_t^2) \right].
\]

The NLPD quantifies the degree of fitness of the estimated predictive distribution \( N(\mu_t, \sigma_t^2) \) for the test data. These two criteria are used broadly in the GP regression literature. A smaller value of MSE or NLPD indicates better performance. Table 2 compares the MSE and NLPD values. The reduced model with the ten selected inputs performed better in both the MSE and the NLPD. This means that the mean and posterior variance estimates with the reduced model better are a fit to the test data, so the 10 selected variables correlate well to the corrosion current.

We also qualitatively evaluated the 10 selected variables based on a corrosion scientist’s expert judgment. The 10 selected variables are regarded as important factors influencing environmental corrosion. Temperature and relative humidity have been identified as major drivers of corrosion in many existing works (Qifei et al., 2009; Friedersdorf et al., 2019). The electrochemical impedance, measured using an AC signal at a high frequency (25 kHz), and the ozone \( O_3 \) level are among the factors popularly studied (Hm-1 Integrated Vehicle Health Management Committee, 2019). Ozone is a strong oxidizer that can lead to significant corrosion and material degradation at a high exposure level, and the impedance can be inversely correlated to the amount of pollutants on the surface of a metal specimen (Friedersdorf et al., 2019). The effect of rain on the corrosion behavior of aluminum is more complicated as it can both reduce corrosion by washing inorganic pollutants off the surface as well as increase corrosion by scrubbing gases out of the air, becoming acid rain. It is generally accepted that rain reduces the corrosion rate on aluminum (Vargel, 2020, page 245). The evidence for this is that outdoor samples covered from the rain have a higher corrosion rate than ones left out in the rain.

We also applied the three benchmark approaches (KL-F, KL-S and VAM) to the same dataset. Figure 8 compares which variables are selected by our proposed approach and the three benchmark approaches. Our approach selected 10 environmental variables, whereas KL-S and VAM selected 15 variables, and KL-F selected 19 variables. The selected variables from different approaches overlapped to a large extent, with nine variables (\( x_2, x_3, x_6, x_9 \) to \( x_{13}, x_{21} \)) selected by all approaches. The quality of the selections are compared in terms of the MSE and NLPD of the GP model learned with the selected variables. Our approach and VAM outperformed the other two approaches in MSE, and our approach achieved the best NLPD among all.

6. Conclusion

We presented a novel VS approach for GP regression, based on a sparse projection of input variables. The approach can be thought of as a generalization of the ARD with a sparsity prior. The major distinctions from the existing approaches are that our approach estimates the sparse projection matrix jointly with other covariance parameters through a marginal likelihood maximization with a sparsity regularization on the projection matrix, while many existing approaches use slow MCMC samplings. In our initial numerical trials, we have tried a simple gradient descent and a quasi-Newton–Raphson algorithm, but they did not give satisfactory outcomes. In particular, the projection matrix tends to be very dense even with a large sparsity penalty. We proposed a forward stagewise regression with embedded gradient descent steps. The numerical approach is an extension of the existing forward stagewise Lasso for a non-convex objective function. We provided some convergence properties. The proposed approach worked successfully for many simulated scenarios, and its VS accuracy outperformed some benchmark approaches for most of the simulated scenarios. The approach was also applied to an important problem of identifying environmental factors that affect atmospheric corrosion of metal alloys, and its VS outcome is evaluated both quantitatively and qualitatively.
Acknowledgments

We would like to acknowledge support for this work from Air Force Research Laboratory, Materials and Manufacturing Directorate and members of the AFRL Corrosion Integrated Product Team.

Funding

The work was conducted under U.S. Federal Government Contract No FA8650-15-D-5405.

Notes on contributors

Chiwoo Park received his BS in industrial engineering at Seoul National University and PhD degree in industrial engineering at Texas A&M University in 2011. He is currently Associate Professor in the Department of Industrial and Manufacturing Engineering at Florida State University and a principal investigator at High Performance Materials Institute. His research includes data science and applied statistics, especially unstructured data analysis involving imagery, functional and shape data.

David J. Borth received his BS in chemical engineering and MS in materials engineering at the University of Dayton. He is currently a corrosion engineer in the Coating Corrosion and Erosion Group at the University of Dayton Research Institute.

Nicholas S. Wilson is a materials research engineer in Materials and Manufacturing Directorate, Air Force Research Laboratory, Wright-Patterson Air Force Base, Ohio. His research encompasses the area of development and evaluation of materials and processes for corrosion prevention and control of Air Force weapon systems. His research interests center on new materials and test methodologies for future aircraft coating systems, including electrochemical evaluation of corrosion protective coatings, metal-rich primers, multi-metal structures, accelerated testing and lifetime prediction, and corrosion modeling and simulation.

Chad N. Hunter is a Senior Materials Engineer and Corrosion Integrated Product Team lead at Air Force Research Laboratory, Materials and Manufacturing Directorate at Wright-Patterson Air Force Base, OH. Chad has an MS from the University of Virginia in Materials Science and Engineering, and a PhD in Materials Engineering from University of Dayton. Dr. Hunter has 25 years of experience working in a variety of technical areas including corrosion engineering, aircraft materials and processes, failure analysis, tribological materials, thermal management materials, and materials test protocol development.

ORCID

Chiwoo Park https://orcid.org/0000-0002-2463-8901

References

Chandra, M.P. (1936) On the generalised distance in statistics. Proceedings of the National Institute of Sciences of India, 2, 49–55.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004) Least angle regression. Annals of Statistics, 32(2), 407–499.

Friedersdorf, F., Demo, J., Brown, N. and Kramer, P. (2019) Electrochemical sensors for continuous measurement of corrosion and coating system performance in outdoor and accelerated atmospheric tests, in Advances in Electrochemical Techniques for Corrosion Monitoring and Laboratory Corrosion Measurements, ASTM International, West Conshohocken, PA, pp. 91–113.

Hm-1 Integrated Vehicle Health Management Committee. (2019) SAE AIR 6970: Environment spectra and corrosivity monitoring using electrochemical and electrical resistance sensors. Technical report, SAE International.

Liu, X. and Guillas, S. (2017) Dimension reduction for Gaussian process emulation: An application to the influence of bathymetry on tsunami heights. SIAM/ASA Journal on Uncertainty Quantification, 5(1), 787–812.

Nutini, J., Schmidt, M., Laradji, I., Friedlander, M. and Koepke, H. (2015) Coordinate descent converges faster with the Gauss-Southwell rule than random selection, in Proceedings of the 32nd International Conference on Machine Learning, Volume 37, JMLR, W&CP, Lille, France, pp. 1632–1641.

Paananen, T., Piironen, J., Andersen, M.R. and Vehtari, A. (2019) Variable selection for Gaussian processes via sensitivity analysis of the posterior predictive distribution, in 22nd International Conference on Artificial Intelligence and Statistics, PMLR, Naha, Okinawa, Japan, pp. 1743–1752.

Park, C. and Apley, D. (2018) Patchwork kriging for large-scale Gaussian process regression. Journal of Machine Learning Research, 19(1), 269–311.

Piironen, J. and Vehtari, A. (2016) Projection predictive model selection for Gaussian processes, in 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP), IEEE Press, Piscataway, NJ, pp. 1–6.

Qifei, Z., Shuangqing, S. and Junguo, W. (2009) Atmospheric corrosion and its influencing factors of aluminum and aluminum alloys. Corrosion and Protection, 30(6), 359–365.

Rusmassen, C. and Williams, C. (2005) Gaussian Process for Machine Learning. MIT Press, Cambridge, MA.

Savitsky, T., Vannucci, M. and Sha, N. (2011) Variable selection for nonparametric Gaussian process priors: Models and computational strategies. Statistical Science, 26(1), 130–149.

Tripathy, R., Bilionis, I. and Gonzalez, M. (2016) Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation. Journal of Computational Physics, 321, 191–223.

Vargel, C. (2020) Corrosion of Aluminium. Elsevier, Amsterdam, The Netherlands.

Vo, G. and Pati, D. (2017) Sparse additive Gaussian process with soft interactions. Open Journal of Statistics, 7(4), 567–588.

Wen, Z. and Yin, W. (2013) A feasible method for optimization with orthogonality constraints. Mathematical Programming, 142(1–2), 397–434.

Williams, C.K. and Rasmussen, C.E. (1996) Gaussian processes for regression. Advances in Neural Information Processing Systems, 8, 514–520.

Zhao, P. and Yu, B. (2007) Stagewise lasso. Journal of Machine Learning Research, 8, 2701–2726.