

A comparison of different methods for building Bayesian kriging models

Younus Al-Taweel^{1*}, Najlaa Sadeek²

* Corresponding Author



1. Department of Mathematics, College of Education for Pure Science, University of Mosul, Iraq. younus.altaweel@uomosul.edu.iq

2. Department of Mathematics, College of Education for Pure Science, University of Mosul, Iraq. najlaasadeek3@gmail.com

Abstract

Kriging is a statistical approach for analyzing computer experiments. Kriging models can be used as fast running surrogate models for computationally expensive computer codes. Kriging models can be built using different methods, the maximum likelihood estimation method and the leave-one-out cross validation method. The objective of this paper is to evaluate and compare these different methods for building kriging models. These evaluations and comparisons are achieved via some measures that test the assumptions that are used in building kriging models. We apply kriging models that are built based on the two different methods on a real high dimensional example of a computer code. We demonstrate our evaluation and comparison through some measures on this real computer code.

Key Words: Kriging models; Computer codes; Maximum likelihood estimation; Cross validation; Robot Arm function.

Mathematical Subject Classification: 62F15, 62J20.

1. Introduction

Computer codes have been widely used in different sciences to describe physical systems. A computer code is a function $y = \eta(\mathbf{x})$ where $\mathbf{x} \in \Omega \in \mathbb{R}^p$ and y is usually a scalar. Computer codes of physical systems are typically complex and so they need a high number of runs to understand how the output of the model changes as the input variables change. However, running computer codes at a high number is computationally very intensive in that obtaining a value of y may take many hours or even many days of computational time. Thus, only a small number of runs can be conducted and analyzed.

One solution to this time-consuming problem of running computer codes is to use a statistical kriging model as a surrogate of the computer code. Kriging models have increasingly become popular for analyzing computer experiments. They can be used to obtain approximations and predictions of complex computer codes. In addition, they can be used to quantify the uncertainty in these predictions. The idea is to build a statistical kriging model of the objective computer code based on a small number of runs of the computer code. Thus, for any point \mathbf{x}^* , the aim of kriging models is to predict a value $y = \eta(\mathbf{x}^*)$ conditional on the available outputs. This kriging model is assumed to be smooth and continuous.

The first use of kriging models was by Sacks et al. (1989). Then, kriging models have been used in many fields of science. They have become an essential tool and provide accurate results as surrogate models for many physical systems. For example, in engineering, Li and Sudjianto (2005) constructed a kriging model to obtain approximations of an expensive power cylinder model. Forrester et al. (2007) also used a kriging model as a surrogate for approximating a multi-variable aircraft wing model. Hung et al. (2009) developed a kriging model for optimizing a hard turning process which is a metal cutting process for producing machined parts out of hard materials. Le Gratiet and Garnier (2014) applied a multi-fidelity kriging model for emulating a hydrodynamic recursive model that simulates a second-order turbulence model for gaseous mixtures. A hierarchical kriging model was used by Abdallah et al. (2019) as a surrogate for a wind turbine loads model.

In environment and climate science, Karaman et al. (2009) analyzed and interpolated Topsoil organic matter, subsoil organic matter and plant nitrogen data using the kriging method. Rossouw et al. (2010) fitted a kriging model for a commercial octene production model. Ly et al. (2013) used a kriging model for interpolating rainfall data in hydrological models. Kriging models were used by Fentanes et al. (2018) for obtaining simulated data of soil moisture models. In health science, Drignei (2008) presented a multidimensional kriging model that can be used for predicting the development of a virtual brain tumor model.

Kriging models can be built using two different methods. By building the kriging model, we mean estimating the model parameters and constructing the model. The first method for estimating the correlation parameters is the maximum likelihood estimation (MLE) method. The second method is the leave-one-out cross validation (LOO CV) method. The LOO CV method requires more computations than the MLE method in estimating the model parameters.

We always find in the literature for building kriging models that the kriging models for complex computer codes are built based on one method for estimating the correlation parameters. People often choose a method for estimating the correlation parameters in building kriging models without providing much explanation for their choice. The two methods for estimating the parameters, however, may give different estimated values. The set of kriging model parameters may affect the performance of kriging models especially with a small number of design points. This means that if the parameters were not estimated well, we may not obtain good approximations for the computer code outputs or the variances may not be quantified well.

Therefore, the objective of our paper is to compare and evaluate the two methods for building kriging models. This will be done using some measures that are based on the differences between the predictions and the true values of the computer code. The paper is organized as follows. In Section 2, we review the process of building kriging models for complex computer codes. Section 3 presents the existing methods for estimating the kriging model parameters. In Section 4, we discuss the design of experiment that is used in the study. Measures for validating kriging models are presented in Section 5. We demonstrate our comparison through a real example of a computer code in Section 6. Finally, the conclusion is given in Section 7.

2. Building kriging models

In this section, we describe how kriging models can be constructed under a Bayesian perspective. Kriging models are based on the mathematical representation of the outputs of the computer code. The computer code is assumed as a random process that can be represented by a Gaussian process as a prior distribution. Thus, the kriging model is a probability distribution that can be used for obtaining predictions of computer codes and quantifying the uncertainty in these predictions.

$$\hat{y} = \sum_{i=1}^p \beta_i \eta_i(\mathbf{x}) + z(\mathbf{x}), \quad (1)$$

where β is a vector of unknown coefficient parameters and $z(\mathbf{x})$ is assumed to be a realization of a stochastic Gaussian process with zero mean and non-zero covariance,

$$z(\mathbf{x}) \sim GP(0, \sigma^2 R(\mathbf{x}_i - \mathbf{x}_j)), \quad (2)$$

where σ^2 is a constant variance. In this work, we use the Matérn correlation function that is given by

$$R_{\nu=5/2}(\mathbf{x}_i - \mathbf{x}_j) = \prod_{i=1}^p \left(1 + \frac{\sqrt{5}|x_i - x_j|}{\theta_i} + \frac{5|x_i - x_j|^2}{3\theta_i^2} \right) \exp \left(-\frac{\sqrt{5}|x_i - x_j|}{\theta_i} \right), \tag{3}$$

where $\theta_i \geq 0$. The correlation parameter, θ_i , determines the smoothness of the function predictions whereas the parameter, $\nu = \frac{5}{2}$ manages the derivative existence of the computer code.

For a given input \mathbf{x} , we assume that the output $y = \eta(\mathbf{x})$ is a realization of a normal random variable $Y \sim N(\mu, \sigma^2)$. Suppose now we have a set of input points $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ for the computer code. We put all the outputs of the computer code at the n input points in a random vector $\mathbf{y} = \{y_1 = \eta(\mathbf{x}_1), \dots, y_n = \eta(\mathbf{x}_n)\}$.

Conditional on \mathbf{y} , the linear predictor of the output at a point \mathbf{x}^* is given by

$$\hat{y}(\mathbf{x}^*) = \lambda^T(\mathbf{x}^*)\mathbf{y}, \tag{4}$$

for $\mathbf{x}^* \in \Omega$. We consider $\hat{y}(\mathbf{x}^*)$ as an uncertain variable. Then, we find the best linear kriging predictor (BLKP), $\hat{y}(\mathbf{x}^*)$, in equation (4) that minimizes the mean squared error (MSE) of the predictions

$$\text{MSE}(\hat{y}(\mathbf{x}^*)) = E[\lambda^T(\mathbf{x}^*)\mathbf{y} - y(\mathbf{x}^*)]^2, \tag{5}$$

under the constraint,

$$E[\lambda^T(\mathbf{x}^*)\mathbf{y}] = E[y(\mathbf{x}^*)]$$

where $y(\mathbf{x}^*) = \eta(\mathbf{x}^*)$.

Now, let $\mathbf{f}(\mathbf{x}^*) = \{\eta_1(\mathbf{x}^*), \dots, \eta_k(\mathbf{x}^*)\}^T$ be a set of regression functions. In this case, the kriging model is known as universal kriging whereas it will be called ordinary kriging if $\mathbf{f}(\mathbf{x}^*) = \{1\}$. In this work, universal kriging models are used. Furthermore, let $F = [\mathbf{f}(\mathbf{x}_1), \dots, \mathbf{f}(\mathbf{x}_n)]^T$ be a vector of regression functions that are evaluated at the inputs points, $\mathbf{x}_1, \dots, \mathbf{x}_n$. Also, suppose \mathbf{R} is the correlation matrix for the n input points

$$\mathbf{R} = \begin{pmatrix} R(\mathbf{x}_1, \mathbf{x}_1) & \cdots & R(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ R(\mathbf{x}_n, \mathbf{x}_1) & \cdots & R(\mathbf{x}_n, \mathbf{x}_n) \end{pmatrix}. \tag{6}$$

This correlation matrix should be positive semi-definite and satisfies that $R(\mathbf{x}, \mathbf{x}) = 1$.

Therefore, conditional on the true values of the outputs, \mathbf{y} , the prediction of the output, which is the BLKP of $\hat{y}(\mathbf{x}^*)$, at the point \mathbf{x}^* is a conditional normal with posterior mean

$$\hat{y}(\mathbf{x}^*) = \mathbf{f}(\mathbf{x}^*)^T \hat{\boldsymbol{\beta}} + \mathbf{r}^T \mathbf{R}^{-1}(\mathbf{y} - \mathbf{F} \hat{\boldsymbol{\beta}}) \tag{7}$$

and the posterior variance, $V_1^*(\cdot, \cdot)$, is

$$\sigma_{y^*}^2 = \hat{\sigma}^2 \left[1 - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} + \mathbf{u}^T (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{u} \right], \tag{8}$$

where

$$\hat{\boldsymbol{\beta}} = (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \mathbf{R}^{-1} \mathbf{y} \tag{9}$$

$$\mathbf{r} = \{R(\mathbf{x}^*, \mathbf{x}_1), \dots, R(\mathbf{x}^*, \mathbf{x}_n)\}^T \tag{10}$$

$$\mathbf{u} = \mathbf{F}^T \mathbf{R}^{-1} \mathbf{r} - \mathbf{y}. \tag{11}$$

(Martin and Simpson, 2005). From now, for simplicity, we will refer to $\hat{y}(\mathbf{x}^*)$ by \hat{y} .

3. Estimating the correlation parameters

To find the distribution of Y , we have to find values of the parameters, β, σ^2 and θ . The constant variance, σ^2 , and the coefficient parameters, β , can be found analytically. The likelihood is given by

$$\eta(\hat{\beta}, \hat{\sigma}^2, \mathbf{R}|\mathbf{y}) = \frac{1}{\hat{\sigma}^2 \sqrt{|\mathbf{R}|} (2\pi)^n} e^{-\frac{1}{2\hat{\sigma}^2} (\mathbf{y} - \mathbf{F}\hat{\beta})^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{F}\hat{\beta})}. \tag{12}$$

By taking the logarithm and using a flat prior for β, σ^2 , the posterior mode of β is

$$\hat{\beta} = (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \mathbf{R}^{-1} \mathbf{y} \tag{13}$$

and the posterior mode of σ^2 is given by

$$\hat{\sigma}^2 = \frac{1}{n} (\mathbf{y} - \mathbf{F}\hat{\beta})^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{F}\hat{\beta}). \tag{14}$$

The correlation parameters θ , however, cannot be found analytically. Thus, we can obtain the maximum likelihood estimate of the correlation parameters, θ , from the likelihood function and then assume them as the true values of θ . The maximum likelihood estimate $\hat{\theta}$ of the true values of the correlation parameters θ can be found by optimizing the following equation (Santner et al., 2003)

$$\hat{\theta} = \arg \min_{\theta} \left[\frac{1}{2} \log(\det(\mathbf{R}))\theta + \frac{n}{2} \log(2\pi\sigma^2) + \frac{n}{2} \right]. \tag{15}$$

The leave-one-out cross validation (LOO CV) method can also be used as an alternative method for estimating the correlation parameters. The LOO CV method is based on iterating the space of the parameter for trying to minimize the error of the model. The optimal parameters are determined by minimizing the following equation (Santner et al., 2003)

$$\hat{\theta} = \arg \min_{\theta} \left[\sum_{i=1}^n (y_i - \hat{y}_i)^2 \right]. \tag{16}$$

After obtaining estimates of the correlation parameters, the kriging model is built based on these estimated values. Kriging models can be implemented using a package in R called `DiceKriging` by Roustant et al. (2012).

4. Design of experiment

In this section, we give details of the design that is used in this study. We used the maximin Latin hypercube design (maximin LHD). The maximin LHD is a space-filling design which depends on measuring the distance between the design points. We first calculate the minimum distance between design points. Then, the maximin LHD maximizes the minimum distance between the design points. The maximin LHD is given by

$$\mathbf{X} = \max_{\mathbf{X} \subset \Omega} \min_{\{x, x'\} \in \mathbf{X}} dis(x_i, x_j). \tag{17}$$

Therefore, the maximin LHD ensures that the design points are uniformly spread in the input space (Fang et al., 2010).

5. Validating kriging models

Kriging models are statistical models that are based on some assumptions. Therefore, it is necessary to use measures to see the performance of kriging models and see whether the assumptions that are used in building kriging models are reasonable or not. In this work, we use two measures for validating kriging models. The first measure is the root relative squared error (RRSE) that is based on the differences between the kriging predictions and the true values of the computer code. The root relative squared error is given by

$$RRSE = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (\bar{y} - y_i)^2}}, \tag{18}$$

where \bar{y} is the mean of the true values of the computer code (Overstall and Woods, 2016). The RRSE tells us how far away are the predicted values from the true value of the computer code. As $RRSE \rightarrow 0$, the kriging predictions will be accurate.

The second measure is the individual standardized errors (ISE) that are given by

$$ISE = \frac{y_i - \hat{y}_i}{\sqrt{\sigma_{y^*}^2}}, \tag{19}$$

where $\sigma_{y^*}^2$ is the posterior variance of the kriging model given by equation (8) (Kolachalama et al., 2007). The individual standardized errors should be between -2 and 2 if the kriging model is valid (Bastos and O’Hagan, 2009). Thus, if large values of ISE are found, then the predictions of the kriging model will not be accurate and so the kriging model will not be valid.

The ISE are not only based on the differences between the kriging predictions and the true values of the computer code, but also depend on the uncertainty in the kriging predictions. This is because the differences between the kriging predictions and the true values of the computer code are normalized by the squared root of the kriging variance.

6. Illustrative Example

In this section, the Robot Arm function is considered as an example of a computer code. The Robot Arm function was used by An and Owen (2001) as an example of implementing quasi-regression methods. The Robot Arm function models the robot arm position that has four segments. The shoulder is considered fixed at the origin, whereas each of the four segments has a length L_i and is set at ϑ_i angle for $i = 1, \dots, 4$. The Robot Arm function is given by

$$\eta(\mathbf{x}) = (u^2 + v^2)^{\frac{1}{2}}, \tag{20}$$

where

$$u = \sum_{i=1}^4 L_i \cos \left(\sum_{j=1}^i \vartheta_j \right)$$

$$v = \sum_{i=1}^4 L_i \sin \left(\sum_{j=1}^i \vartheta_j \right)$$

The output of the Robot Arm function is the distance between the origin and the end of the robot arm. The Robot Arm function has eight input variables and they are $\mathbf{x} = (\vartheta_1, \vartheta_2, \vartheta_3, \vartheta_4, L_1, L_2, L_3, L_4)$. The ranges of the input variables are: $\vartheta_i \in [0, 2\pi]$ for $i = 1, \dots, 4$ where $\vartheta_1, \vartheta_2, \vartheta_3$ and ϑ_4 are the angles of the first, second, third and fourth segments of the arm respectively, $L_i \in [0, 1]$ for $i = 1, \dots, 4$ where L_1, L_2, L_3 and L_4 are the lengths of the first, second, third and fourth segments of the arm respectively.

We generated a set of 40 design points, $\mathbf{x}_1, \dots, \mathbf{x}_{40}$, by the maximin LHD. Then, the outputs of the Robot Arm function were obtained at these 40 points, $\mathbf{y} = \{y_1 = \eta(\mathbf{x}_1), \dots, y_n = \eta(\mathbf{x}_{40})\}$. Before we fit the kriging models, the domain of the Robot Arm function was mapped to be in $[0, 1]^8$.

Based on the 40 design points, the correlation parameters were estimated using the two methods, the MLE method equation (15), and the LOO CV method, equation (16), and they are $\hat{\theta}_{MLE} = (1.97, 1.94, 1.94, 1.96, 1.97, 1.97, 1.95, 1.94)$ for the MLE method and $\hat{\theta}_{CV} = (1.97, 1.94, 1.39, 1.96, 1.97, 1.97, 0.73, 1.78)$ for the LOO CV method. We can notice that some inputs with the MLE method have larger correlation parameters than the corresponded ones with the LOO CV method. This means that the computer code is more smooth with these inputs with the MLE method than those with the LOO CV method.

Conditional on the $\hat{\theta}_{MLE}$ and $\hat{\theta}_{CV}$, kriging models were built using two methods for estimating the correlation pa-

rameters, the MLE and the LOO CV. We used a zero mean function and the Matérn correlation function equation (3) in the package `DiceKriging`. Then, a set of 16 points, generated using maximin LHD, was used for validating the kriging models, conditioned on y . Figure 1 shows the kriging models for the Robot Arm function with 95% credible intervals based on the MLE method and the LOO CV method.

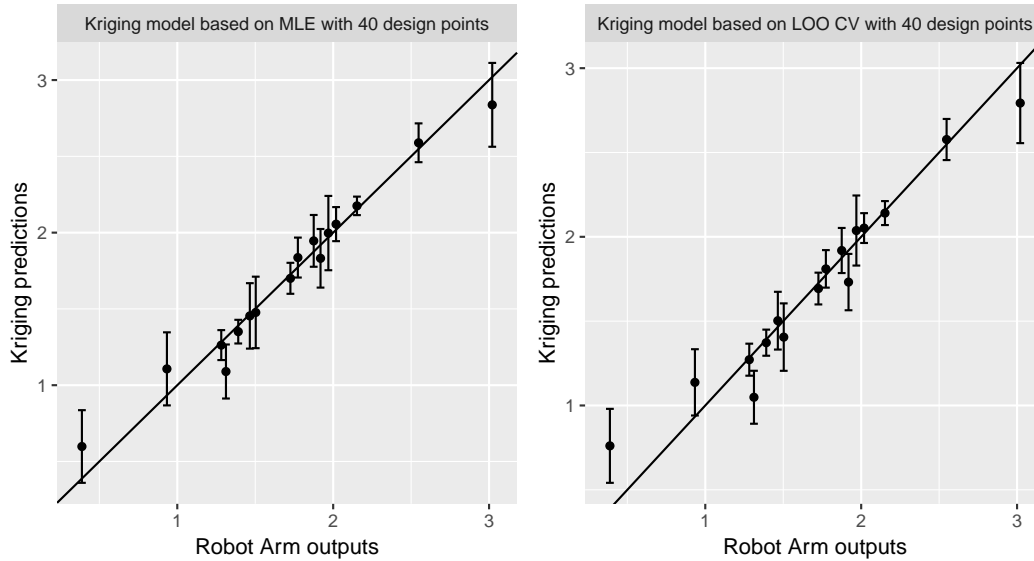


Figure 1: The predictions of kriging models against the true values for the Robot Arm function with 95% credible intervals based on the MLE method and the LOO CV method. The kriging predictions seem to be good approximations of the computer code outputs. Some points of the kriging models are not very close to the $y=x$.

The kriging predictions are reasonable as the majority of them lie around the $y = x$ line. Furthermore, the uncertainty, which is represented by the errorbars, of many points seems to be small. This indicates that the predictions of the kriging models are good approximations of the computer code outputs with 95% credible intervals. However, several points seem to be not close to the $y = x$ line, especially with the LOO CV method. This means that the kriging models are not valid in some parts of the input space. This may also indicate that the predictions that are obtained by the MLE method are more accurate than those that are obtained by the LOO CV method.

6.1. Measures for the kriging models

Now, we compare and investigate more closely the performance of the kriging models that were built using the different methods for estimating the correlation parameters, the MLE method and the LOO CV method. This comparison between the kriging models is conducted via some measures. We used two different set sizes of the design points, $5p$ and $10p$, where p is the dimension of the input. The size of the validation points was set to be $2p$.

6.1.1. The root relative squared error

The first measure that we calculated for validating and comparing the kriging models is the root relative squared error (RRSE) equation (18). The RRSE measures the overall performance of the kriging models. The RRSE was calculated for the kriging models that were built based on the MLE and the LOO CV methods. We replicated each of the combinations of the design points and the validation points 100 times with different sets of design and validation points for each replication.

Figure 2 shows the box-plots of the RRSE for the kriging models that are based on the MLE method and for the kriging models that are based on the LOO CV method with different 40 design points and a different set of 16 validation points for each replication. The box-plots are based on 100 values of the RRSE.

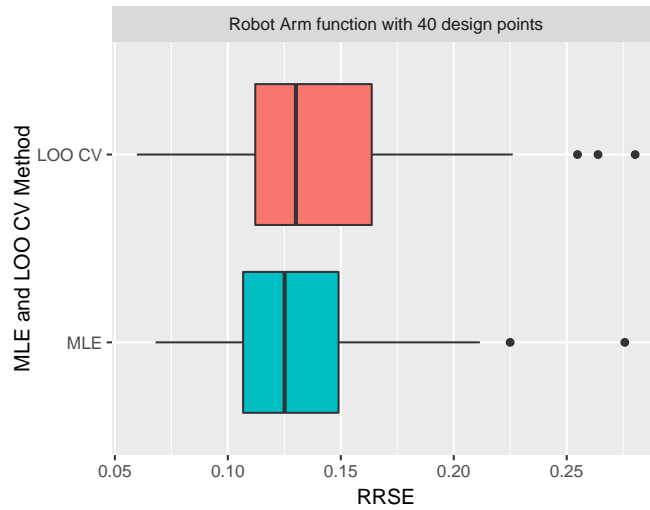


Figure 2: The box-plots of the RRSE for the kriging models of the Robot Arm function that are based on the MLE method and the LOO CV method with 40 design points and a set of 16 validation points. The values of the RRSE for kriging models that are based on the MLE method are smaller than those for kriging models that are based on the LOO CV method.

It can be seen from Figure 2 that the values of the RRSE are small for kriging models of both methods. The kriging models that are based on the MLE method show smaller values of the RRSE than those of the kriging models that are based on the LOO CV method. Moreover, the box-plots of the RRSE for the kriging models show some outliers but the outliers for the kriging models that are based on the LOO CV method are more than those for the kriging models that are based on the MLE method.

We increased the set size of the design points to become 80 and the processes were repeated again. Figure 3 shows the box-plots of the RRSE for the kriging models that are based on the MLE method and for the kriging models that are based on the LOO CV method with 80 design points and a set of 16 validation points. The box-plots are also based on 100 values of the RRSE.

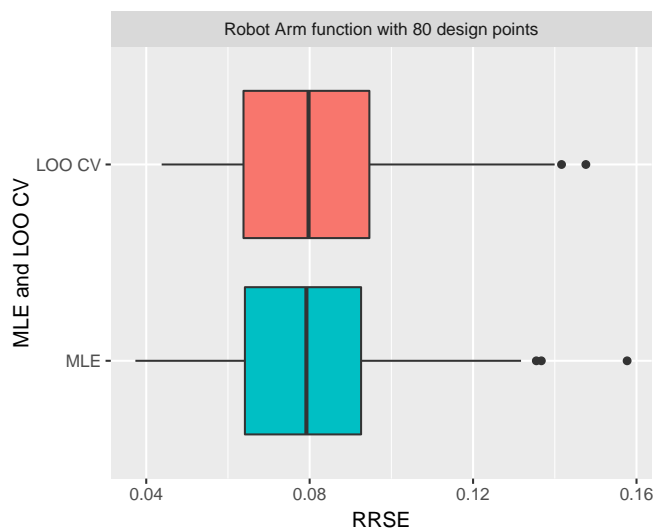


Figure 3: The box-plots of the RRSE for the kriging models of the Robot Arm function that are based on the MLE method and the LOO CV method with 80 design points and a set of 16 validation points. The values of the RRSE for kriging models that are based on the MLE method and on the LOO CV method are very small.

Now, it can be shown from Figure 3 that the values of the RRSE are very small for kriging models that are based on the MLE method and the LOO CV method. Moreover, the values of the RRSE are smaller than before for the two methods. Figure 3 also shows that the values of the RRSE for the kriging models that are based on the MLE method are also smaller than those from the kriging models that are based on the LOO CV method. However, the values of the RRSE for the kriging models that are based on the LOO CV method are now very close to those for the kriging models that are based on the MLE method. This indicates that the predictions of the kriging models that are based on the LOO CV method were improved by increasing the design points.

We can conclude that the values of the RRSE, for kriging models that are based on the MLE method and for kriging models that are based on the LOO CV method, have become smaller by increasing the design points. Furthermore, based on the values of the RRSE, the performance of the kriging models that are based on the MLE method is better than that of the kriging models that are based on the LOO CV method with a small number of design points. This is because the predictions of the kriging models that are based on the MLE method are more closer to the true values of the Robot Arm function than those of the kriging models that are based on the LOO CV method.

The performance of the kriging models that are based on the LOO CV method, however, was improved by increasing the design points. The predictions of the kriging models that are based on the LOO CV method tend to be as good as the predictions of the kriging models that are based on the MLE method by increasing the design points.

6.1.2. The individual standardized errors

We applied another measure that does not only depend on the differences between the predictions and the true values of the Robot Arm function, but also considers the uncertainty in the predictions of the kriging model. We calculated the individual standardized errors (ISE) equation (19). Thus, we consider the plot of the ISE against the index to compare the performance of the kriging models that are based on the two different methods for estimating the correlation parameters. Figure 4 shows the ISE against the index for the kriging model that is based on the MLE method and for the kriging model that is based on the LOO CV method with 40 design points and a set of 16 validation points.

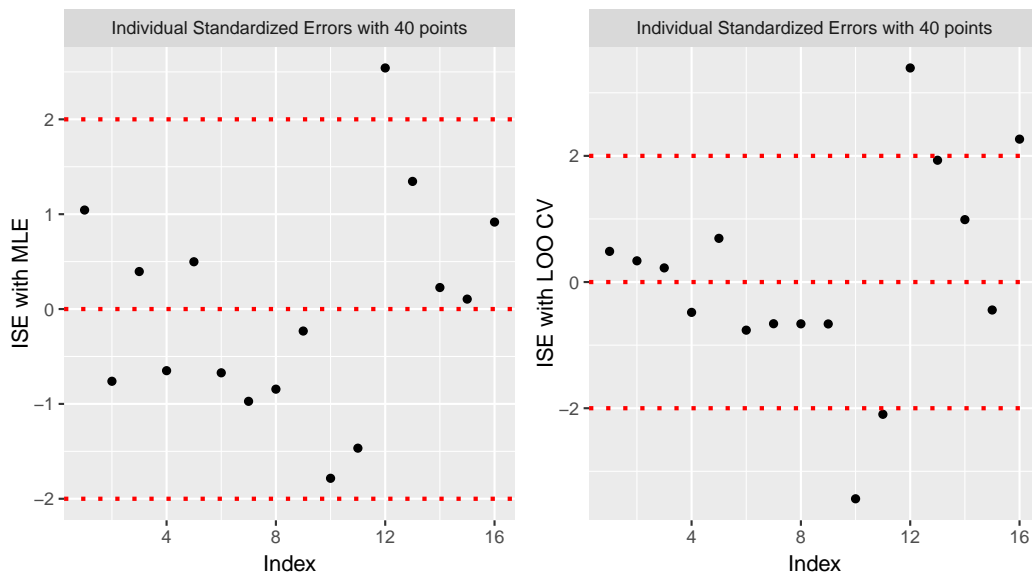


Figure 4: The plots of the ISE for the kriging model that is based on the MLE method and for the kriging model that is based on the LOO CV method with 40 design points and a set of 16 validation points. Many large ISE are seen in the plot of the kriging model that is based on the LOO CV method.

The left panel of Figure 4 shows the ISE for the kriging model that is based on the MLE method. It can be seen that majority of the ISE are small and only one error lies outside the bounds, but close to the bounds. The right panel of Figure 4 shows the ISE for the kriging model that is based on the LOO CV method. It can be shown that several large

ISE are seen in the plot.

We also increased the set size of the design points to be 80 and the processes were repeated again. Figure 5 shows the plots of the ISE for the kriging model that is based on the MLE method and for the kriging model that is based on the LOO CV method with 80 design points and a set of 16 validation points.

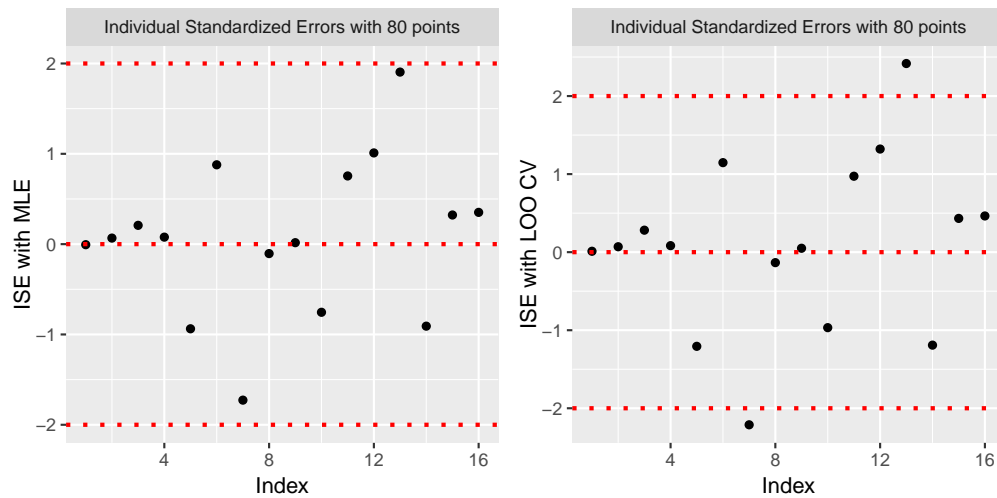


Figure 5: The plots of the ISE for the kriging model that is based on the MLE method and for the kriging model that is based on the LOO CV method with 80 design points and a set of 16 validation points. The ISE seem to be reasonable with two errors are seen in the plot of the kriging model that is based on the LOO CV method.

Now, the ISE for the kriging model that is based on the MLE method are also small where all of them lie inside the bounds. The ISE for the kriging model that is based on the LOO CV method in the right panel of Figure 5 are now small with only two ISE lie outside the bounds. These two errors, however, are very close to the bounds.

We can conclude from the plots of the ISE that the kriging model that is based on the LOO CV method may not perform well with a small number of design points. The performance the kriging model that is based on the LOO CV method may become as good as that of the kriging model that is based on the MLE method when an adequate set size of design points is used. This is because the ISE for the kriging model that is based on the LOO CV method have become smaller with 80 design points. We have seen an adequate number of the design points should be no less than ten times the input dimension, $10p$.

7. Conclusion

In this work, we reviewed a popular statistical method, called kriging, for approximating and analyzing complex computer codes. Estimating the kriging model parameters can be conducted using two different methods, the maximum likelihood estimation method and the leave-one-out cross validation method. According to our results, the use of the leave-one-out cross-validation method for estimating the parameters in building kriging models is not very reliable with a small set size of design points. The performance of the leave-one-out cross-validation method may become as good as the performance the maximum likelihood estimation method when a reasonable set size of design points is used in building kriging models. Another disadvantage of estimating the parameters by the leave-out-out cross validation method is that it takes a longer time than the maximum likelihood estimation method to achieve the calculations. This is because we need to calculate the distance between all the design points and this will be computationally expensive with high-dimensional computer codes.

References

1. Abdallah, I., Lataniotis, C., and Sudret, B. (2019). Parametric hierarchical kriging for multi-fidelity aero-servoelastic simulators application to extreme loads on wind turbines. *Probabilistic Engineering Mechanics*, 55:67–77.

2. An, J. and Owen, A. (2001). Quasi-regression. *Journal of complexity*, 17(4):588–607.
3. Bastos, L. S. and O'Hagan, A. (2009). Diagnostics for gaussian process emulators. *Technometrics*, 51(4):425-438.
4. Drignei, D. (2008). Fast statistical surrogates for dynamical 3d computer models of brain tumors. *Journal of Computational and Graphical Statistics*, 17(4):844–859.
5. Fang, K.-T., Li, R., and Sudjianto, A. (2010). *Design and modeling for computer experiments*. CRC Press.
6. Fentanes, J. P., Badiée, A., Duckett, T., Evans, J., Pearson, S., and Cielniak, G. (2018). Kriging-based robotic exploration for soil moisture mapping using a cosmic-ray sensor. arXiv preprint arXiv:1811.05384.
7. Forrester, A. I., Sobester, A., and Keane, A. J. (2007). Multi-fidelity optimization via surrogate modelling.
8. *Proceedings of the royal society a: mathematical, physical and engineering sciences*, 463(2088):3251–3269.
9. Hung, Y., Joseph, V. R., and Melkote, S. N. (2009). Design and analysis of computer experiments with branching and nested factors. *Technometrics*, 51(4):354–365.
10. Karaman, M. R., Susam, T., Er, F., Yaprak, S., and Karkacier, O. (2009). Simulation of organic matter variability on a sugarbeet field using the computer based geostatistical methods. *World Academy of Sci*, 56:594–598.
11. Kolachalama, V. B., Bressloff, N. W., and Nair, P. B. (2007). Mining data from hemodynamic simulations via Bayesian emulation. *Biomedical engineering online*, 6(1):47.
12. Le Gratiet, L. and Garnier, J. (2014). Recursive co-kriging model for design of computer experiments with multiple levels of fidelity. *International Journal for Uncertainty Quantification*, 4(5).
13. Li, R. and Sudjianto, A. (2005). Analysis of computer experiments using penalized likelihood in gaussian kriging models. *Technometrics*, 47(2):111–120.
14. Ly, S., Charles, C., and Degre, A. (2013). Different methods for spatial interpolation of rainfall data for operational hydrology and hydrological modeling at watershed scale: a review. *Biotechnologie, Agronomie, Societeet Environnement*, 17(2):392–406.
15. Martin, J. D. and Simpson, T. W. (2005). Use of kriging models to approximate deterministic computer models. *AIAA journal*, 43(4):853–863.
16. Overstall, A. M. and Woods, D. C. (2016). Multivariate emulation of computer simulators: model selection and diagnostics with application to a humanitarian relief model. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*.
17. Rossouw, R., Coetzer, R., and Pretorius, P. (2010). Simulation experiments for maximising the availability of a commercial octene production facility. *ORiON*, 26(1):53–77.
18. Roustant, O., Ginsbourger, D., and Deville, Y. (2012). Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization.
19. Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423.
20. Santner, T. J., Williams, B. J., Notz, W., and Williams, B. J. (2003). *The design and analysis of computer experiments, volume 1*. Springer.