Function Extrapolation of Noisy Data using Converging Lines

Yiming Zhang* and Nam-Ho Kim†
*University of Florida, 333 MAE-A, Gainesville, FL 32611-6250, U.S.A.
†Professor, Mechanical and Aerospace Engineering, University of Florida, 333 MAE-A, Gainesville, FL 32611-6250, U.S.A and AIAA Associate Fellow

Chanyoung Park‡
‡Postdoctorate Associate, Mechanical and Aerospace Engineering, University of Florida, 333 MAE-A, Gainesville, FL 32611-6250, U.S.A

Raphael T. Haftka§
§Distinguished Professor, Mechanical and Aerospace Engineering, University of Florida, 333 MAE-A, Gainesville, FL 32611-6250, U.S.A and AIAA Fellow

This paper is focused on extrapolating noisy data to a single inaccessible point using the method of converging lines. Matrix multiplication computation time, which is a two-dimensional unimodal and monotonic function of matrix dimensions, was adopted as a test problem. Using the method of converging lines, multi-dimensional extrapolation was first transformed into series of one-dimensional extrapolation towards one point. One-dimensional extrapolation results at the extrapolation point were combined using Bayes method. One-dimensional long-range extrapolation was then performed differently depending on the noise level of the data. For low noise levels, the data pattern was clear using logarithmic transformation and standard polynomial regression was used for fitting it. For significant noise levels, ridge regression was used to reduce overfitting. This paper proposes a new scheme to determine the ridge parameter by minimizing prediction error at training samples. This scheme was evaluated based on manufactured data.

I. Introduction

SURROGATE modeling is widely used to approximate mechanical systems for optimization or reliability analysis. The goal of surrogate modeling is to estimate the function value efficiently and accurately at a large number of points.

Function estimation procedure is called interpolation when evaluation points are inside the convex hull of sampled points, while it is termed extrapolation1-3 otherwise, as shown in Fig. 1. For one-dimensional samples, convex hull is set by the lower- and upper- bounds of samples. Surrogate-based interpolation has been extensively used for many engineering problems. Extrapolation is usually associated with large uncertainty and commonly encountered in the following three situations:

1) Sampling schemes such as Latin Hypercube sampling, which are adopted to determine locations of samples, may cause voids near the boundaries of domain which are outside the convex hull of samples.
2) For function estimation in high-dimensional space, too many points are required to avoid extrapolation. For example, in a twenty-dimensional hyper-cube, more than million points (at all the vertices) are required to avoid extrapolation.

3) Prediction of function value at points beyond the domain where it is possible and affordable to sample.

Extrapolation has been broadly studied for specific circumstances, such as forecasting one-dimensional time series.\(^4\)\(^5\) Major sources of extrapolation uncertainty include potential inconsistency of function behavior between sampling domain and extrapolation domain, uncertainty of model form and coefficients estimation. Good correlation of function behavior between sampling domain and extrapolation domain is critical to extrapolation accuracy and requires physical understanding for assessing.

The method of converging lines\(^6\)\(^-\)\(^7\) has been developed for extrapolation of multi-dimensional function value at a single point and proved to be effective for certain deterministic functions (i.e. drag coefficient function). This method transforms multi-dimensional extrapolation into a series of one-dimensional extrapolations towards the target point. One advantage of this method is that the reliability of extrapolation could be improved significantly by validating prediction from different lines. Kriging surrogate was recommended to extrapolate smooth functions because kriging allows effective identification of how far extrapolation is accurate\(^6\)\(^-\)\(^7\). However, as often data are noisy, this paper explores the use of other surrogates when data is noisy.

We adopted matrix the computation time of matrix multiplication as a test problem. This test problem is a monotonic and unimodal function of the two matrix dimensions. Inconsistency between sampling domain and extrapolation is unlikely as the algorithm to multiply matrix is the same. Therefore the main challenges are due to model form and coefficient estimation especially when making predictions far from samples. Regularization method such as Ridge regression has been reported to be a promising tool to reduce overfitting due to noise in the interpolation domain\(^8\). Extrapolation accuracy could be also improved using appropriate ridge parameters.\(^9\) One objective of the paper is to develop a scheme for determining ridge parameter for extrapolation.

The outline of the paper is as follows. The method of converging lines is described in Section 2, and the matrix multiplication computation time function is described in section 3. Section 4 discusses extrapolation strategy for dense data with small noise (i.e. high-quality samples). Section 5 discusses how to extrapolate sparse data with large noise (i.e. low-quality samples) with regularization method. A new scheme to determine ridge parameter is proposed and evaluated using manufactured data. Section 6 provides a summary and concluding remarks.

II. Function Extrapolation using Method of Converging Lines

The method of converging lines serves for extrapolation toward one target point, which includes sampling, one-dimensional extrapolation and post-processing techniques. The method of converging lines follows the following steps:

1) Sampling: Instead of one of the standard space filling sampling approaches, such as shown in Fig. 2(a), select several lines from sampling domain towards one inaccessible extrapolation point as shown in Fig. 2 (b). Line \(l_i\) originates from extrapolation point and intersection angles between \(l_i\) and \(l_j\) at target point is denoted as \(\alpha_{ij}\) in normalized variable space. Lines close to one another are usually highly correlated, which complicates combining their predictions. Hence the guideline used for selecting lines is to maximize the angles between lines

\[
\max_{i \neq j} \left( \min(\alpha_{ij}) \right)
\]  

Figure 1. Function interpolation and extrapolation
2) Generate samples along each line.
3) Perform one-dimensional extrapolation using surrogates along each line separately.
4) Check consistency between extrapolation results from different lines using an appropriate criterion. Consistent results are combined based on Bayes theory assuming that they are independent and their uncertainty is normally distributed to obtain the mean and standard deviation of the estimate as shown in Eq. (2). Where $p_i$ is the extrapolation from line $i$, and $p_p$ is posterior prediction and $p_i \sim N(\mu_i, \sigma_i^2)$, $p_p \sim N(\mu_p, \sigma_p^2)$.

$$\begin{align}
\frac{1}{\sigma_p^2} &= \sum_{i=1}^{n} \frac{1}{\sigma_i^2} \\
\frac{\mu_p}{\sigma_p^2} &= \sum_{i=1}^{n} \frac{\mu_i}{\sigma_i^2}
\end{align}$$

(2)

Figure 2. Illustration of (a) space-filling sampling (LHS here) and (b) Method of converging lines using 15 samples when the target point is at the origin.

III. Computation Time for Matrix Multiplication

Matrix multiplication is a basic operation in engineering calculation, and the execution time of matrix multiplication is a basic indicator of computation platform performance\textsuperscript{10, 11}. The matrix multiplication routine used in this paper could estimate execution time for multiplication of two matrices having dimensions of $M \times N$ and $N \times M$. While the number of floating point operations is easy to calculate, computation time is not proportional to the number of floating point operations because of different features of the computing platform such as cache size. Furthermore, the computation time is not deterministic as it depends on other computing or communication events occurring on the computing platform. Therefore Rudolph etc.\textsuperscript{10} described the use of measurements for different matrix sizes fit to a kriging surrogate for obtaining an explicit estimate as function of matrix dimensions using noisy data. Since matrix multiplication can be expensive for large sizes, the objective here is to check on the feasibility of estimating matrix multiplication time for large-size matrices from results obtained from lower size matrices. Overall
function trends and lines for performing extrapolation are illustrated in Fig. 3. Function value increases monotonically along any line. The multiplication time varies by orders of magnitude.

When taking measurements of matrix multiplication times on a computing platform, the level of noise depends on competing usage of the platform. The numerical test of matrix multiplication was performed using a platform with an Intel Ivy Bridge processor. A number of hardware and software precautions were adopted to reduce uncertainty of measured execution time. The most significant hardware modifications included:

1. Disabling all of the dynamic clocking features of the Intel processor- both in the processor and memory hierarchy.
2. Disabling the dynamic power-management features of the Intel processor-including all dynamic limitations on power, current, and temperature.

The software precautions include:
1. Performing a large number of runs, inversely related to the problem size and asymptotic run time of the algorithm.
2. Repeating the sampling of execution time, omitting outliers, and averaging the median group.
3. Increasing the priority of the benchmarking processes.

The Extrapolation point was set to (N, M)=(511,511). The design space was set to be $N \in [16,511], M \in [16,511]$ from where $N \in [264,511], M \in [264,511]$ was set to be the inaccessible domain. This means that along M and N directions the extrapolation distance is the same as that of sampling domain.

Low-quality and high-quality data sets were generated for studying the corresponding extrapolation strategies. Low-quality data denotes sparse samples with large noise level compared to the range of function values. High-quality data denotes dense data with small noise level. Denote the ratio between estimated standard deviation of noise and range of function value as $r_{low}$ and $r_{high}$ for low-quality data and high-quality data respectively. For the data of matrix multiplication function, $r_{low} \sim O(1\%)$ and $r_{high} \sim O(10\%)$ in logarithmic coordinate. Logarithmic transformation in following analysis implies common logarithms.

High-quality data is presented in Fig. 3 (a) in logarithmic coordinate. It’s clear that it’s a unimodal function with monotonic trend along M and N direction. Three lines were generated as shown in Fig. 3(b). Low-quality data include 11 samples along each line in accessible domain and generated without precautions. High-quality data include 30 samples along each line in accessible domain and generated with precautions. Computation time of low-quality data were larger than high-quality data for same matrix size.

Figure 3. High-quality data of matrix-multiplication function and line selections for method of converging lines

IV. Extrapolation of Dense Data with Small Noise

There were 61 uniform samples along each line as shown in Fig. 4. 30 samples on the left were adopted for sampling and the rest samples were used for validation. Two-dimensional matrix size was transformed to be one-
dimensional coordinate using L2 norm (vector length in the N,M plane). It is assumed that the average computation
time is a smooth curve and noise has similar coefficient of variation. Outliers may happen due to the periodical
cache refreshing at certain critical matrix size. Linear regression is based on the assumption that the noise level is
the same everywhere. Therefore logarithmic transformation was applied, which transforms constant coefficient of
variation to constant variance. In addition, we found that logarithmic transformation of 1D matrix size makes the
function trend close to linear. The analysis of high-quality data will be performed in log-log coordinate. Indeed,
trying different polynomial fits, a linear polynomial response surface was adopted based on leave-one-out cross-
validation. The linear fit along the three lines is shown in Fig.5. High-quality data usually enables data pattern
analysis which is always recommended.

Table 1. Prediction of high-quality data at extrapolation points along three lines ( \( p_i \) is the computation
time in natural logarithm)

<table>
<thead>
<tr>
<th></th>
<th>Line 1</th>
<th>Line 2</th>
<th>Line 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prediction ( p_i )</td>
<td>-0.62</td>
<td>-0.73</td>
<td>-0.75</td>
</tr>
<tr>
<td>Prediction variance</td>
<td>( 0.04^2 )</td>
<td>( 0.04^2 )</td>
<td>( 0.03^2 )</td>
</tr>
</tbody>
</table>

Extrapolation results along the three lines are summarized in Table 1. The Relative discrepancy was computed
based on Eq.(3). Where \( std \) denotes standard deviation at the extrapolation, \( p_i \) denotes prediction along line \( i \). Large
\( d_{ij} \) implied inconsistency between lines.

\[
d_{ij} = \frac{|p_i - p_j|}{\max(\text{std})}
\]

(3)

Figure 4. High quality data of matrix multiplication computation time function

Considering \( d_{12} = 2.75, d_{13} = 3.25, d_{23} = 0.5 \), line 1 is identified to be an inconsistent prediction. It could be seen
in Fig. 3 and Fig. 4 that samples along line 1 were outliers. The appearance of outlier events beyond scope of this

American Institute of Aeronautics and Astronautics
paper and would be investigated in other work. Line 2 and line 3 were then identified to be reasonable and combined based on Eq. (2). The combined result was $N(-0.7423, 0.02^2)$ and true function value $-0.8153$ in logarithmic coordinate. The relative difference at extrapolation point was 9% in logarithmic coordinate and 18% in natural coordinate. 18% seems a reasonable relative error for long-range extrapolation of noisy data. Extrapolation error (i.e. 3.5 standard deviation of estimated noise) was larger than the 95% confidence interval (i.e. 1.96 standard deviation of estimated noise). This may be due to that the true function isn’t exactly a straight line.

Figure 5. Extrapolation of high-quality data along three lines in log-log coordinate with base 10.

V. Extrapolation of Sparse Data with Large Noise

The True function form and trend are usually difficult to estimate from low-quality samples. Therefore only function value was processed in logarithmic coordinate and matrix size was modeled in natural coordinate. Extrapolation will be performed in semi-log coordinate with base 10. While estimating function values based on low-quality data, one important source of uncertainty for surrogate modeling is over-fitting noise which may be magnified in the extrapolation domain. Regularization methods such as ridge regression has been reported to be effective for reducing over-fitting noise and improving extrapolation accuracy. Ridge regression is essentially a penalized polynomial response surface. That is, the regression objective $E$ to be minimized is the sum of the squares of the errors plus a penalty for the magnitude of the regression coefficients as shown in Eq. (4). The regression coefficients are denoted as $\beta$ and computed according to Equation (5). $\lambda$ is the parameter to be specified. Predictions will be the same as polynomial response surface when $\lambda \rightarrow 0$. Predictions will approach the mean value of samples when $\lambda \rightarrow \infty$.

$$E = \sum_i (y_i - x_i^T \beta)^2 + \lambda \sum_j \beta_j^2 \quad (4)$$
$$\beta = (X^T X + \lambda I)^{-1} X^T y \quad (5)$$

The ridge parameter has been mainly selected for interpolation purpose in previous studies. We propose a method to determine regularization parameter of ridge regression for extrapolating one-dimensional functions. Samples are first divided into training points and test points based on their distance towards the target extrapolation point as shown in Fig. 4. Extrapolation domain is set to have the same length as sampling domain. Therefore the range of training points are similar to that of test points. Ridge parameter was determined by simulating extrapolation in the sampling domain.

For a given $\lambda$, ridge regression was developed based on training points and corresponding root-mean-square error(RMSE) could be obtained at test points as shown in Fig. 6. $\lambda$ is then selected to minimize the mean square error at test points. Extrapolation towards target point was then performed based on both training points and test points with this optimum $\lambda$.

The predictors of ridge regression were standardized and function values were centered. As shown in Eq. (6). The design matrix $X$ didn’t contain constant terms. Denote $X_i$ as the $i$th column of $X$. Each column was
standardized based on mean and standard deviation as shown in Eq. (7). Function value was centered as shown in Eq. (8). The intercept term would be computed based on estimated regression coefficients. Standardization was usually adopted for interpretability of coefficients and reducing ill-conditioning\textsuperscript{12,13}. Standardization procedure may change $\beta$ significantly while incorporating training samples for extrapolation with a fixed $\lambda$. Considering the noise level and relative range between samples will be similar, standardization effect is neglected while modeling data with large noise.

$$X = \begin{bmatrix} x_1 & x_1^2 & x_1^3 & \ldots & x_1^p \\ x_2 & x_2^2 & x_2^3 & \ldots & x_2^p \\ x_3 & x_3^2 & x_3^3 & \ldots & x_3^p \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_n & x_n^2 & x_n^3 & \ldots & x_n^p \end{bmatrix}$$

(6)

$$X_i^{(n)} = \frac{X_i - \text{mean}(X_i)}{\text{std}(X_i)}$$

(7)

$$y^{(0)} = y - \text{mean}(y)$$

(8)

The proposed procedure is evaluated using manufactured data. We simulated large sets of samples with artificial noise to test the performance of proposed regularization method. A cubic function was adopted to approximate function along line 1 as shown in Eq. (9). Gaussian noise was generated from $\mathcal{N}(0,0.28^2)$ at samples. 50 uniform samples are first generated as shown in Fig. 7. Target function is a representative of functions with small curvature.

$$f(x) = -0.0187 - 1.5230x - 1.0646x^2 - 0.9326x^3, \quad x \in [0,511]$$

(6)

Figure. 7 compares standard polynomial regression to ridge regression. Fig. 7(c) shows a box plot of relative error at extrapolation point based on 100 sets of samples. Polynomial regression was substantially improved by ridge regression selecting regularization parameter by simulating extrapolation. Extrapolation based on regularization method may avoid huge errors. We then tested the influence of polynomial order, noise level and number of samples on extrapolation accuracy using Ridge regression. As seen from Fig. 8, noise level is the most significant factor on extrapolation accuracy. Ridge regression was then used to extrapolate three lines of matrix multiplication function as shown in Fig. 9. True function value could be bounded by the estimated error bound which is in the 95% confidence interval obtained based on bootstrapping\textsuperscript{14}. The prediction variance for ridge regression wasn’t from normal distribution. The confidence interval was approximated based on a normal distribution which could cover that based on ridge regression for computation convenience. This approximation would only affect prediction slightly considering the magnitude of prediction variance are similar.

### Table 2. Prediction of low-quality data at extrapolation points along three lines. ( $p_i$ is the computation time in natural logarithm)

<table>
<thead>
<tr>
<th>Line 1</th>
<th>Line 2</th>
<th>Line 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prediction $p_i$</td>
<td>0.13</td>
<td>-0.10</td>
</tr>
<tr>
<td>Prediction variance</td>
<td>$0.15^2$</td>
<td>$0.20^2$</td>
</tr>
</tbody>
</table>

The low-quality data along three lines were modeled using ridge regression with cubic polynomials as shown in Fig. 9. Extrapolation results are summarized in Table 2. The Relative discrepancy was computed based on Eq.(3). Considering $d_{i2} = 2, d_{i3} = 1.4, d_{i3} = 0.2$, all the lines are considered to be consistent. The maximum discrepancy $d_{12}$ was slightly larger than 1.96 which is the magnitude for 95% confidence interval. But prediction at extrapolation points and the standard deviation of prediction had similar order of magnitude, large uncertainty of prediction would
be expected. The combined result based on Eq.(2) was $N\left(-0.09, 0.06^2\right)$ and true function value was -0.18 in logarithmic coordinate. The relative difference at extrapolation point was 50% in logarithmic coordinate and 23% in natural coordinate. The large magnitude of prediction variance implied large uncertainty of extrapolation.

**Figure 6.** Regularization parameter is determined by simulating extrapolation from training points to test points

**Figure 7.** Comparing extrapolation of line 1 from matrix multiplication function using Ridge regression and polynomial response surface.

**VI. Summary**

We investigated how to extrapolate multi-dimensional noisy function using converging lines in this paper. It’s noticed that data should be modeled based on noise level of samples. High-quality data was likely to enable data pattern analysis and hence improved prediction accuracy. Using method of converging lines identified inconsistent prediction alone lines. For long-range extrapolation, the difference between combined extrapolation and function value was as small as 18%. Low-quality data maybe encountered for practical problems and ridge regression was selected to reduce overfitting noise. We proposed to determine the ridge parameter by dividing samples as test points and training points. Surrogate was developed based on training points and optimum parameter was then obtained by minimizing RMSE at test points. The proposed procedure proved to be effective to reduce large error and stabilize prediction based on manufactured data based on a unimodal function with monotonic trend.
We are trying to apply proposed procedure for estimating convergence of discretized simulation. In the future work the effect of data standardization will be investigated. Determining ridge parameter based on simulating extrapolation may be also extended to multi-dimensional modeling.

![Influence of polynomial order](image1)
![Influence of noise level](image2)
![Influence of sample number](image3)

**Figure 8.** Influence of polynomial order, noise level and number of samples on extrapolation accuracy using Ridge regression. Extrapolation accuracy is obtained based on 100 sets of samples generated along Line 1 of matrix multiplication function using Monte Carlo simulation.

![1D matrix size vs computation time](image4)

**Figure 9.** One-dimensional extrapolation of matrix multiplication function using Ridge regression along three lines of Fig. 3. Extrapolation in natural coordinate is provided in parentheses.

**Acknowledgments**

This work is supported by the U.S. Department of Energy, National Nuclear Security Administration, Advanced Simulation and Computing Program, as a Cooperative Agreement under the Predictive Science Academic Alliance Program, under Contract No. DE-NA0002378. The software routine for matrix multiplication and high-quality samples for matrix multiplication software were gratefully furnished by Dylan Rudolph from High-performance Computing and Simulation Research Lab at University of Florida.
References


