Aerospace Science and Technology 75 (2018) 74–87

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

A fast-prediction surrogate model for large datasets

John T. Hwang [a,*], Joaquim R.R.A. Martins [b]

[a] Peerless Technologies Corporation, Beavercreek, OH 45324, United States
[b] Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109, United States

ARTICLE INFO

Article history:
Received 21 May 2017
Received in revised form 25 December 2017
Accepted 30 December 2017
Available online 31 January 2018

Keywords:
Surrogate modeling
Response surfaces
Metamodels
Regression
Multidisciplinary design optimization
Krylov methods

ABSTRACT

Surrogate models approximate a function based on a set of training points and can then predict the function at new points. In engineering, kriging is widely used because it is fast to train and is generally more accurate than other types of surrogate models. However, the prediction time of kriging increases with the size of the dataset, and the training can fail if the dataset is too large or poorly spaced, which limits the accuracy that is attainable. We develop a new surrogate modeling technique—regularized minimal-energy tensor-product splines (RMTS)—that is not susceptible to training failure, and whose prediction time does not increase with the number of training points. The improved scalability with the number of training points is due to the use of tensor-product splines, where energy minimization is used to handle under-constrained problems in which there are more spline coefficients than training points. RMTS scales up to four dimensions with 10–15 spline coefficients per dimension, but scaling beyond that requires coarsening of the spline in some of the dimensions because of the computational cost of the energy minimization step. Benchmarking using a suite of one- to four-dimensional problems shows that while kriging is the most accurate option for a small number of training points, RMTS is the best alternative when a large set of data points is available or a low prediction time is desired. The best-case average root-mean-square error for the 4-D problems is close to 1% for RMTS and just under 10% for kriging.

1. Introduction

A surrogate model is an approximation that is cheaper or more convenient to evaluate than the underlying model it approximates. The most common use of surrogate models is to replace a known expensive computational model when a large number of repeated evaluations is required, e.g., for optimization or uncertainty quantification. Another common application is when we want to obtain a continuous function from a fixed dataset, e.g., when the data is obtained experimentally or from legacy code. A third application is smoothing an underlying model with a lower order of continuity, perhaps to achieve differentiability for gradient-based optimization [1].

In discussions of surrogate models, it is beneficial to separate the construction and evaluation of the model, because most such models have parameters that are precomputed during the construction stage. Here, we refer to the evaluation of the model as *prediction*. Given n_x inputs and n_w parameters, the prediction is the evaluation of

$$y = f(\mathbf{x}, \mathbf{w}), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is an input vector, $y \in \mathbb{R}$ is the output variable, and $\mathbf{w} \in \mathbb{R}^{n_w}$ is the vector of model parameters. We refer to the construction of the model as *training*, and thus to the dataset as *training points*. The objective of training is to compute the model parameters $\mathbf{w}$ that satisfy or approximate

$$\bar{y}_i \approx f(\bar{\mathbf{x}}_i, \mathbf{w}), \qquad \forall 1 \leq i \leq n_t \tag{2}$$

where $(\bar{\mathbf{x}}_1, \bar{y}_1), \ldots, (\bar{\mathbf{x}}_{n_t}, \bar{y}_{n_t})$ are the n_t training points.

Because of its wide applicability and usefulness, surrogate modeling has been a topic of active research for decades [2]. In engineering, kriging is one of the most commonly used methods for several reasons [3,4]. First, it is the most accurate method overall for small or moderate numbers of training points ($n_t < 10^3$), as we confirm in Section 4. Second, kriging training and prediction times scale well with the number of dimensions, n_x, enabling its use in high-dimensional problems where n_x can be as high as $\mathcal{O}(10^2)$. Third, its stochastic interpretation provides an estimate of the prediction error via the variance of the prediction point. However, the disadvantages of kriging include the increase in prediction time with the number of training points, and the propensity of the training to fail when the training points are too close to each other.

* Corresponding author.
E-mail address: hwangjt@umich.edu (J.T. Hwang).

https://doi.org/10.1016/j.ast.2017.12.030
1270-9638/

These disadvantages limit the maximum number of training points that kriging can handle, which in turn limits the accuracy that can be achieved when many training points are available.

In this paper, we are primarily motivated by applications in which the surrogate model is a part of a larger model. For example, the surrogate model might approximate aircraft aerodynamic performance with respect to the flight conditions, where the surrogate is a part of a multidisciplinary model that includes other disciplines represented by other models that may or may not be surrogates. If the set of training points is fixed, then the surrogate model can be trained once in advance and used repeatedly each time the multidisciplinary model is run. This is the case in many problems in multidisciplinary design optimization (MDO), and it encourages emphasizing prediction time more than training time. Predictions are made repeatedly to converge the multidisciplinary system, which in turn is done once per optimization iteration. In some problems, this can lead to millions of predictions for a surrogate model trained once [5]. Other applications, such as surrogate-based optimization, place more weight on a lower training time because the training occurs at every optimization iteration.

We develop a new surrogate modeling method for low-dimensional problems ($n_x \leq 4$) that we call *regularized minimal-energy tensor-product splines* (RMTS). RMTS is generally slower to train than kriging, but it has a fast prediction time that does not increase with the number of training points. Moreover, it can work with much larger numbers of training points, meaning that when large datasets are available, e.g., when the data source is a fast but nondifferentiable model, the accuracy that can be achieved with RMTS is expected to be higher than with kriging, as we show in Section 4. Interest in tensor-product splines has declined in the last few decades because of their poor scaling with n_x; however, modern computing hardware mitigates these scaling limitations and enables RTMS to scale up to four-dimensional problems. Moreover, tensor-product splines enable prediction that is orders of magnitude faster than kriging when the number of training points is large ($n_t > 10^4$). RMTS uses energy minimization and regularization to improve accuracy with small datasets and to handle unstructured datasets, i.e., training points not arranged in a structured grid.

RMTS is available under an open-source license as part of the *surrogate modeling toolbox* (SMT).[1] All the benchmarking problems, as well as the other surrogate modeling approaches considered in this paper, are included in the SMT repository, so our results are fully reproducible.

The paper is organized as follows. In Section 2, we review some of the surrogate modeling methods that are commonly used in engineering: polynomials, splines, artificial neural networks, support-vector regression, inverse-distance weighting, radial basis functions, and kriging. In Section 3, we present the equations and solution algorithms of RMTS. In Section 4, we use a benchmarking suite to evaluate RMTS and to compare the surrogate modeling methods in terms of training time, prediction time, and accuracy. We also discuss the use of RMTS in a practical MDO context dealing with aircraft mission optimization.

2. Review of surrogate modeling methods

In engineering, a surrogate model is also known as a *response surface* in some contexts, or as a *metamodel*, reflecting the idea that it is a model of an underlying model. In this paper, we use *surrogate model* throughout to remain consistent, while noting that different terms are used in other contexts.

[1] https://github.com/SMTorg/smt.

Surrogate modeling approaches can be classified as *interpolation* (if the surrogate model matches the true function value at each point in the training dataset) or *regression* (if it does not). Regression methods smoothly approximate noisy data, and they include polynomials, splines, artificial neural networks (ANN), and support vector regression (SVR). Interpolation methods attempt to smoothly and accurately fit non-noisy data, and they include inverse distance weighting (IDW), radial basis functions (RBFs), and kriging. These methods are extensively discussed in the literature [6–8].

Since RMTS is classified as an interpolation method, we review the regression methods briefly and explain the interpolation methods in more detail. Section 4 presents results comparing RMTS to IDW, RBFs, and kriging, so we also present the equations for each, in the form they are implemented for the benchmarking.

2.1. Regression methods

2.1.1. Polynomial regression

Polynomial regression uses low-order global polynomials in multiple variables to approximate the training data. Polynomial response surfaces were originally introduced by Box and Wilson [9]. They have the advantage of simplicity, making them fast and easy to work with. However, they lack flexibility, and therefore for many types of problems they are less accurate than other methods.

2.1.2. Splines

The most successful surrogate modeling method using splines is multivariate adaptive regression splines (MARS), developed by Friedman [10]. MARS uses basis functions that are piecewise linear in each dimension and adaptively splits the basis functions using a greedy algorithm. MARS scales well with problem dimension (n_x), but the downside is that both the training and prediction times increase with the number of knots, which is tied to accuracy.

2.1.3. Artificial neural networks

ANNs work with an interconnected set of nodes that compute an activation signal based on inputs, just as neurons in the brain fire based on impulses. These nodes are arranged in layers, where one layer consists of the n_x inputs, another layer consists of the n_y outputs, and the remaining layers are known as *hidden layers*. Compared to the typical surrogate modeling techniques in engineering, neural networks display slower convergence of error versus the number of training points. On the other hand, they are capable of dealing with significantly higher-dimensional problems, such as speech and character recognition.

2.1.4. Support vector regression

SVR [11] also has its roots in machine learning, but it has been successful as a method for surrogate modeling in engineering applications. It is typically derived first as an optimization problem that finds the most "flat" linear approximation with a prescribed precision. The dual problem yields an equivalent form with a dot product between the input vectors in the objective function, and replacing this dot product with another function leads to the general SVR method. Choosing the Gaussian function turns out to be similar to RBFs with a Gaussian kernel, except that it performs regression with a prescribed tolerance rather than interpolation.

2.2. Interpolation methods

2.2.1. Inverse distance weighting

IDW, also known as Shepard's method [12], uses a linear combination of the training outputs, where the coefficients are computed from the inverse of the distance from the prediction point to each training point. It exactly interpolates unstructured data