Tree-Based Response Surface Analysis

Siva Krishna Dasari[1], Niklas Lavesson[1], Petter Andersson[2], and Marie Persson[1]

[1] Department of Computer Science and Engineering, Blekinge Institute of Technology, SE-371 79, Karlskrona, Sweden
[2] Engineering Method Development, GKN Aerospace Engine Systems Sweden, Dept. 9635 - TL3, SE-461 81, Trollhättan, Sweden

Abstract. Computer-simulated experiments have become a cost effective way for engineers to replace real experiments in the area of product development. However, one single computer-simulated experiment can still take a significant amount of time. Hence, in order to minimize the amount of simulations needed to investigate a certain design space, different approaches within the design of experiments area are used. One of the used approaches is to minimize the time consumption and simulations for design space exploration through response surface modeling. The traditional methods used for this purpose are linear regression, quadratic curve fitting and support vector machines. This paper analyses and compares the performance of four machine learning methods for the regression problem of response surface modeling. The four methods are linear regression, support vector machines, M5P and random forests. Experiments are conducted to compare the performance of tree models (M5P and random forests) with the performance of non-tree models (support vector machines and linear regression) on data that is typical for concept evaluation within the aerospace industry. The main finding is that comprehensible models (the tree models) perform at least as well as or better than traditional black-box models (the non-tree models). The first observation of this study is that engineers understand the functional behavior, and the relationship between inputs and outputs, for the concept selection tasks by using comprehensible models. The second observation is that engineers can also increase their knowledge about design concepts, and they can reduce the time for planning and conducting future experiments.

Keywords: machine learning, regression, surrogate model, response surface model.

1 Introduction

The design phase is an important step of product development in the manufacturing industry. In order to design a new product, the engineers need to evaluate suitable design concepts. A concept is usually defined by a set of design variables, or attributes. The design variables represent various design choices such as the material type or thickness of a specific part. During the design phase, several concepts are defined by providing different attribute values. Engineers may opt to use a combination of computer aided design (CAD) modeling and computer-simulated experiments instead of real experiments, in order to reduce the time, cost and risk. The simulations contribute to a better understanding of the functional behavior and predict possible failure modes in future product use [15]. They are used to identify interesting regions in the design space and to understand the relationship between design variables (inputs) and their effect on design objectives (outputs) [12]. However, one single computer-simulated experiment can take a significant amount of time to conduct. For instance, to design a part of an aero engine, an engineer has to simulate, in order to select an optimal product design, several variants where sets of parameters are studied with respect to different aspects, such as strength and fatigue, aero performance and producibility. Conducting simulations for each concept is impractical, due to time constraints. In order to minimize the time consumption and simulations, engineers use methods such as design of experiments and surrogate models, or response surface models, for design space exploration [6].

Surrogate modeling is an engineering method used when an outcome of interest cannot be directly measured [14]. The process of surrogate model generation includes sample selection, model generation and model evaluation. Sample selection is used to select a set of input samples using different types of sampling strategies (e.g., random sampling) for model generation [7]. The next step is to construct surrogate models from a small set of input samples and their corresponding outputs. The purpose of surrogate modeling is to find a function that replaces the original system and which could be computed faster [7]. This function is constructed by performing multiple simulations at key points of the design space; thereafter the results are analyzed and then the selection of an approximation model to those samples follows [7]. In machine learning, this type of learning of an approximation function from inputs and outputs is called a supervised learning problem. The approximation function is real valued so the problem is delimited to supervised regression learning. The challenge of surrogate modeling is the generation of a surrogate that is as accurate as possible by using the minimum number of simulation evaluations. This motivates the generation of surrogate models in an efficient way that can be used in concept selection.

Statistical approaches have been used to construct surrogate models using a technique called response surface methodology [4]. Engineers use statistical regression analysis to find the relationship between inputs and outputs. They usually generate regression functions by fitting a curve to a series of data points. Another engineering design strategy to generate surrogate models is the use of a black box model (e.g., support vector machines) [10]. The problem with black box models is the lack of information about the functional behavior and the mapping between inputs and outputs. Black box models can be accurate but they are not comprehensible, and there is a need to generate accurate and comprehensive surrogate models in order to understand the model behaviour. In this study, we use machine learning algorithms for response surface analysis, and we addresses the supervised regression problem with tree models. Tree models are used to create comprehensible models that are easy to interpret [22], since they reveal the mapping process between inputs and outputs. We can thus interpret and learn about the approximation function between the inputs and the outputs. The motivation for selecting tree methods in this study is, tree has a graphical structure, and tree model representation follows the divide and conquer approach and this structure provides the information about important attributes. Mathematical equations and non-linear models are difficult to understand due to the model representations [9]. We hypothesize that comprehensible models can be used to increase the understanding about design spaces with few simulation evaluations while maintaining a reasonable accuracy level. In our study, we used M5P tree and random forest tree methods for response surface modeling. These two methods have their tree nature in common, thus, we refer to them as "tree based learning" in this study.

2 Aim and Scope

The focus of this study is to use supervised machine learning algorithms for response surface models. The goal of this study is to empirically investigate how tree models perform on design samples from concept selection tasks, and to determine which regression tree induction approach yields the best performance. We hypothesize that tree models will create accurate and comprehensive models for response surfaces. The tree algorithms are applied to real-world data from the aerospace industry. Tree methods (M5P and random forests) are compared with non-tree methods (support vector machines and linear regression) to explore potential differences in various aspects of performance which is accuracy of the response surface models. This study will not focus on the choice of sampling strategy or dataset generation strategies in order to optimize the learning process. Instead, performance is measured on pre-existing and anonymized real-world data.

3 Related Work

Gorissen et al. presents a surrogate modeling and adaptive sampling toolbox for computer based design. This toolkit brings together algorithms (support vector machines, kriging, artificial neural networks) for data fitting, model selection, sample selection (active learning), hyper parameter optimization, and distributed computing in order to empower a domain expert to efficiently generate an accurate model for the problem or data at hand [10].

Ahmed and Qin used surrogate models for design optimization of a spiked blunt body in hypersonic flow conditions. This study constructed four surrogate models, namely a quadratic response surface model, exponential kriging, gaussian kriging and general exponential kriging based on the values of drag and heating responses. The authors concluded that exponential kriging surrogate produces a relatively better prediction of new points in the design space and better optimized design [1]. Haito et al used surrogate model for optimization of an underwater glider and compared several experimental design types and surrogate modeling techniques in terms of their capability to generate accurate approximations for the shape optimization of underwater gliders. The authors concluded that combination of multi-island genetic algorithm and sequential quadratic programming is an effective method in the global exploration, and showed that the modified method of feasible direction is an efficient method in the local exploration [12].

Robert et al introduced the use of the treed Gaussian process (TGP) as a surrogate model within the mesh adaptive direct search framework (MADS) for constrained black box optimization. Efficiency of TGP method has been demonstrated in three test cases. In all test cases, MADS-TGP is compared with MADS alone and MADS with quadratic models. Finally, the authors concluded that TGP is taking more execution time to compare with other two methods but TGP provides the quality of the solution for one of the test cases. For the other two test cases, TGP gives better solutions compared to the other methods [11].

Machine learning methods such as support vector machines, artificial neural networks have already been used extensively for surrogate models [1] [10]. These methods are black box models and there are no comprehensible models that have been developed using machine learning for surrogate models. To the best knowledge of the authors, tree-based models from machine learning for response surface analysis have not been investigated for concept selection tasks in product development. Thus, this study is focused on tree methods to generate surrogate models.

4 Background

In many modern engineering problems, accurate simulations are used instead of real experiments in order to reduce the overall time, cost, or risk [7]. It is impossible to evaluate all possible concepts by conducting simulations to identify the most suitable concept. For instance, an engineer gets requirements to design a product, but he or she might not have enough time to test all concepts by conducting simulations. Thus, engineers can run few simulations using few concepts to generate a surrogate model to predict unseen concepts for design space exploration. Design optimization, design space exploration, and sensitivity analysis are possible through surrogate model generation [6].

Engineers choose a set of concepts using suitable sampling strategies. Latin hypercube sampling (LHS) is one of the most common sampling strategies currently used to select input concepts for surrogate model generation. The concepts can be changed by many different input variables such as the materials for various parts, thickness, colors, lengths, etc. The different variants of concepts are represented in 3D using CAD software. CAD/CAE (computer aided engineering) is the use of computer systems to assist in the creation, modification, analysis, or optimization of a design [2]. Through a CAD model, we can get outputs from each concept or design, which indicates how the design performs, for example strength, stiffness, weight etc. The final step is surrogate model generation based on inputs and outputs.

4.1 Methodology

In this section, we briefly introduce the studied machine learning methods for response surface modeling and the common performance metrics for regression problems. In this study, we use root mean-squared error (RMSE) [22] and the correlation coefficient [17] to evaluate the predictive performance. The RMSE is calculated as the sum of squared differences of the predicted values and the actual values of the regression variable divided by the number of predictions. This RMSE gives an idea to the engineer about the difference between actual values and predicted values. The correlation coefficient (CC) measures the strength of association between the predicted values and the actual values [17]. The following equations show the RMSE [22] and the correlation coefficient (CC) [17].

$$RMSE = \frac{1}{N}\sum_{i=1}^{n} \left(\hat{y} - y\right)^2 \tag{1}$$

Where $\hat{y}$ is the predicted value and y is the actual value.

$$CC = \sum_{i=1}^{n} \frac{\left(\hat{y_i} - \bar{\hat{y}}\right)\left(y_i - \bar{y}\right)}{\sqrt{\sum_{i=1}^{n} \left(\hat{y_i} - \bar{\hat{y}}\right)^2 \left(y_i - \bar{y}\right)^2}} \tag{2}$$

Where $\hat{y_i}$ is the predicted value; y_i is the actual value; $\bar{\hat{y}}$ is the mean value of the predicted values; and $\bar{y}$ is the mean value of the actual values.

The main purpose of this study is to investigate the performance of tree models for response surface analysis. Hence, we have selected the M5P algorithm and the RF algorithm. The M5P and random forests (RF) algorithms are tree models and these two models show the functional behavior between the inputs and the outputs in a comprehensible way. To compare tree model performance against a traditional benchmark, we have selected two more models linear regression (LR) and support vector machines (SVM). These algorithms are regression methods, but these two algorithms do not show the function behavior between inputs and outputs.

Linear regression is a statistical method for studying the linear relationship between a dependent variable and a single or multiple independent variables. In this study, we use linear regression with multiple variables to predict a real-valued function. The linear regression model is considered in the following form [22].

$$x = w_0 + w_1 a_1 + w_2 a_2 + \ldots + w_k a_k \tag{3}$$

Where x is the class; a_1, a_2, $\ldots$ a_k are the attribute values; w_0, w_1 $\ldots$ w_k are weights. Here, the weights are calculated from the training data. The linear regression method is used to minimize the sum of squared differences between the actual value and the predicted value. The following equation shows the sum of squares of the difference [22].

$$\sum_{i=1}^{n}\left(x^{(\mathrm{i})}-\sum_{j=0}^{k} w_j {a_j}^{(\mathrm{i})}\right)^2 \tag{4}$$

Where the equations shows the difference between the i^{th} instance's actual class and its predicted class.

M5P Quinlan developed a tree algorithm called M5 tree to predict continuous variables for regression [16]. There are three major steps for the M5 tree construction development: 1) tree construction; 2) tree pruning; and 3) tree smoothing. Detailed descriptions for these three steps are available in [16]. The tree construction process attempts to maximize a measure called the standard deviation reduction (SDR). Wang modified the M5 algorithm to handle enumerated attributes and attribute missing values [21]. The modified version of the M5 algorithm is called the M5P algorithm. The SDR value is modified to consider missing values and the equation is as follows [21].

$$SDR = \frac{m}{|T|} \times \beta(i) \times \left[sd(T) - \sum_{j\in L,R} \frac{|T_j|}{|T|} \times sd(T_j)\right] \tag{5}$$

Where T is the set of cases; T_j is the j^{th} subset of cases that result from tree splitting based on set of attributes; $sd(T)$ is the standard deviation of T ; and $sd(T_i)$ is a standard deviation of T_i as a measure error; m is the number of training cases without missing values for the attribute; $\beta(i)$ is the correction factor for enumerated attributes; T_L and T_R are the subsets that result from the split of an attribute.

SVM This method is used for both classification and regression and it is proposed by Vapnik [20]. In the SVM method, N-dimensional hyperplane is created that divides the input domain into binary or multi-class categories. The support vectors are located near to the hyperplane, and this hyperplane separates the categories of the dependent variable on each side of the plane. The kernel functions are used to handle the non-linear relationship. The following equation shows the support vector regression function [5].

$$\bar{y_i} = \sum_{j=1}^{n}(\alpha_j - \alpha_j^*)K(x_i, x_j) + b \tag{6}$$

where K is a kernel function; α_j is a Lagrange multiplier and b is a bias. Detailed descriptions of these concepts of SVM can be found in [18] [20].

Random Forest This method is an ensemble technique developed by Breiman. It is used for both classification and regression [3], and it combines a set of decision trees. Each tree is built using a deterministic algorithm by selecting a random set of variables and random samples from a training set. To construct an ensemble, three parameters need to be optimized: (1) *ntree*: the number of regression trees grown based on a bootstrap sample of observations. (2) *mtry*: the number of variables used at each node for tree generation. (3) *nodesize*: the minimal size of the terminal nodes of the tree [3]. An average of prediction error estimation of each individual tree is given by mean squared error. The following equation shows the mean squared error (MSE) [3].

$$MSE = n^{-1} \sum_{i=1}^{n} [\hat{Y}(X_i) - Y_i]^2 \tag{7}$$

Where $\hat{Y}(X_i)$ is the predicted output corresponding to a given input sample whereas Y_i is the observed output and n represents the total number of out of bag samples.

5 Experiments and Analysis

In this section, we present the experimental design used to compare the methods for response surface modeling. We use the algorithm implementations available from the WEKA platform for performance evaluation [22]. The experimental aim is to determine whether tree models are more accurate than mathematical equation-based models. To reach this aim, the following objectives are stated:

1. To evaluate the performance of LR, M5P, SVM and RF for response surface modeling.
2. To compare tree models and non-tree models on the task of design space exploration.

5.1 Dataset Description

The algorithms are evaluated on two concept-selection data sets obtained from the aerospace industry. These datasets are from simulations and sampled by using LHS. The first dataset consists of 56 instances with 22 input features and 14 output features. The second data set includes 410 instances defined by 10 input features and three output features. In the company which is aerospace industry, engineers generate one regression model for each output feature. For this single output model, we have 14 sub data sets for the first dataset, and three sub datasets for the second dataset. We generate 14 new single-target concept-selection data sets, D1-1 to D1-14 by preserving its input features and values, and selecting a different output feature for each new data set. Using the same procedure as for the first data set, we generate three new single-target concept-selection data sets, D2-1 to D2-3.

Table 1. Performance comparison on 17 datasets

Data set	LR	M5P	RF	SVM	LR	M5P	RF	SVM
	RMSE (rank)				*CC (rank)*			
D1-1	0.5787(2)	0.2059(1)	2.0553(4)	0.9553(3)	0.995(2)	0.9994(1)	0.9700(4)	0.9908(3)
D1-2	10.8545(3)	5.2926(1)	10.4724(2)	11.6372(4)	0.8273(4)	0.9640(1)	0.8900(2)	0.8373(3)
D1-3	0.2838(3)	0.2726(2)	0.3155(4)	0.2545(1)	-0.1562(2)	-0.0232(1)	-0.3133(4)	-0.1696(3)
D1-4	0.0062(1)	0.0062(1)	0.0171(3)	0.0091(2)	0.9922(1)	0.9922(1)	0.9688(3)	0.9859(2)
D1-5	0.2414(3)	0.2252(2)	0.2720(4)	0.2178(1)	-0.0585(3)	0.1302(1)	-0.2878(4)	0.1817(2)
D1-6	0.0051(2)	0.0050(1)	0.0151(4)	0.0080(3)	0.9945(2)	0.9947(1)	0.9724(4)	0.9884(3)
D1-7	0.1416(3)	0.1421(1)	0.1714(4)	0.1442(2)	-0.6527(4)	-0.0952(1)	-0.3265(3)	-0.1366(2)
D1-8	0.0232(2)	0.0127(1)	0.0459(4)	0.0315(3)	0.9792(2)	0.9938(1)	0.9661(4)	0.9766(3)
D1-9	0.0907(2)	0.0888(1)	0.1067(4)	0.0928(3)	-0.6381(4)	-0.0125(1)	-0.3362(3)	-0.0495(2)
D1-10	0.0232(2)	0.0122(1)	0.0464(4)	0.0318(3)	0.9801(2)	0.9945(1)	0.9727(4)	0.9777(3)
D1-11	4.4332(3)	3.9521(2)	5.5322(4)	2.9258(1)	0.9805(3)	0.9846(2)	0.9747(4)	0.9916(1)
D1-12	0.0196(1)	0.0199(2)	0.0254(4)	0.0237(3)	0.8211(1)	0.8175(2)	0.6747(4)	0.7251(3)
D1-13	0.0419(1)	0.0419(1)	0.0482(3)	0.0466(2)	0.1186(1)	0.1137(2)	-0.0592(3)	-0.0984(4)
D1-14	0.1549(2)	0.1648(4)	0.1248(1)	0.1580(3)	0.4980(3)	0.4335(4)	0.7143(1)	0.5057(2)
D2-1	0.0676(4)	0.0647(2)	0.0602(1)	0.0661(3)	0.6655(4)	0.6995(2)	0.7482(1)	0.6853(3)
D2-2	0.1270(3)	0.0673(1)	0.0757(2)	0.1306(4)	0.5190(4)	0.9031(1)	0.8639(2)	0.5194(3)
D2-3	1.2226(2)	1.1370(1)	1.2752(4)	1.2469(3)	0.4312(3)	0.5445(1)	0.4918(2)	0.4296(4)
Avg. rank	2.29	1.47	3.29	2.58	2.64	1.41	3.05	2.70

5.2 Evaluation Procedure

We use cross-validation to maximize training set size and to avoid testing on training data. Cross-validation is an efficient method for estimating the error [13]. The procedure is as follows: the dataset is divided into k sub samples. In our experiments, we choose k = 10. A single sub-sample is chosen as testing data and the remaining k - 1 sub-samples are used as training data. The procedure is repeated k times, in which each of the k sub-samples is used exactly once as testing data and finally all the results are averaged and single estimation is provided [13]. We tuned the parameters for RF and SVM. For RF, we use a tree size of 100, and for SVM, we set the regularization parameter C to 5.0, and the kernel to the radial basis function. These parameters are tuned in WEKA [22]. We start with a C value of 0.3 and then increase with a step size of 0.3 until the performance starts to decrease. We select the number of trees starting from a low value and then increase up to 100 for improved accuracy.

5.3 Experiment 1

In this section we address the first objective. For this purpose, we trained the four methods with 10 fold cross-validation on datasets D1-1 to D1-14 and D2-1 to D2-3. For this experiment, we normalized the D2-1, D2-2 and D1-14 datasets. Table 1 shows the RMSE values, CC values and the ranks for the four methods.

Analysis: For 11 out of 17 datasets the use of the M5P tree method yields the best results with respect to the RMSE metric. The LR and SVM algorithms outperformed the other algorithms for three datasets each, and the last method: RF yields the lowest RMSE for only two datasets. When it comes to the CC performance metric, M5P tree yields the best performance for 11 datasets, and LR yields the best performance for three datasets. The other methods, RF and SVM, yield the best CC for two datasets.We observe that tree models (M5P in 11 cases and RF in 2 cases) are performing better in a majority of cases compared to the other models for LHS sampled datasets. The reason for this could be that tree models divide the design space into regions and create a separate model for each region, whereas SVM and LR create single model over the entire design space. Tree models are in general regarded as more comprehensible models than the other investigated models [9]. We observe that tree methods could be used to gain knowledge of design samples for design space exploration, by finding the decision paths from the root of the tree to the top branches. For instance, an engineer using a tree method to predict the output value y based on the input values $x_1, x_2, x_3, \ldots, x_n$, can increase his understanding of the relationship between inputs and output by analyzing their mapping. On the other hand, when the engineer wants to predict a new y value for various concepts, there is a possibility to reduce the time because the engineer has already reached an understanding about the model, and can also make informed decisions regarding future experiments.

Our experiment requires statistical tests for comparing multiple algorithms over multiple datasets. The Friedman test is a non-parametric statistical test that can be used for this purpose [8]. It ranks the algorithms for each dataset based on the performance. The best performing algorithm gets a rank of 1 and the second best algorithm gets a rank of 2 and so on, and finally it compares the average ranks of the algorithms [8]. The common statistical method for testing the significant differences between more than two sample means is the analysis of variance (ANOVA) [19]. ANOVA assumes that the samples are drawn from normal distributions [8]. In our study, the error measure samples cannot be assumed to be drawn from normal distribution hence we violate the ANOVA parametric test. The hypothesis is:
H_o: LR, M5P, SVM and RF methods perform equally well with respect to predictive performance
H_a: There is a significant difference between the performances of the methods

The statistical test produces a p-value of 0.0003 for RMSE, and a p-value of 0.0016 for CC. The p-value is less than the 0.05 significance level. We therefore reject the null hypothesis and conclude that there is a significant difference between the performances of methods. Furthermore, we conducted post a hoc test for pairwise comparisons to see the individual differences. For this purpose, we used the Nemenyi test [8]. Table 2 shows the p-values for the pairwise comparison.

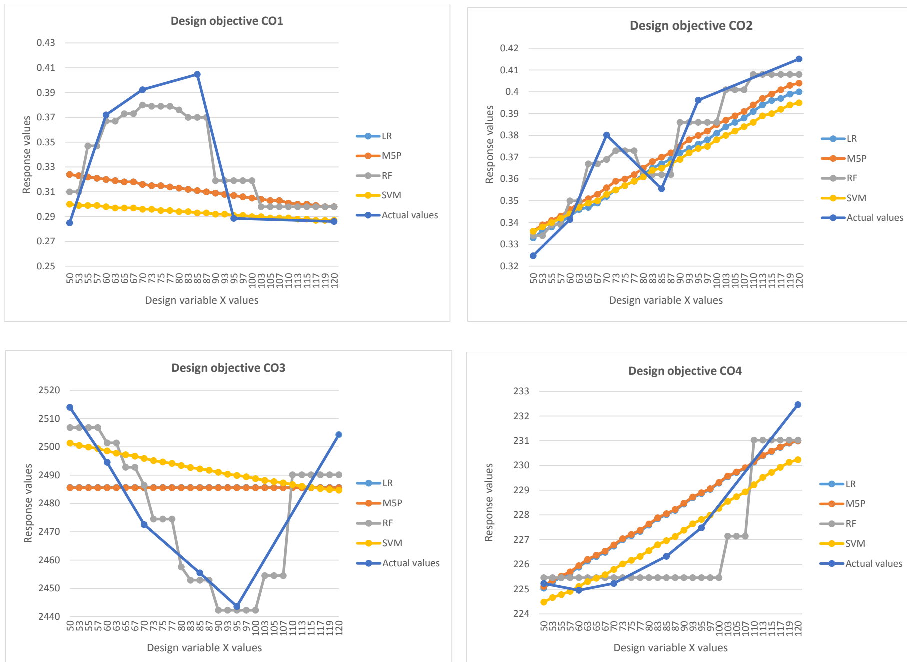

Fig. 1. Plots for four design objectives using four methods

5.4 Experiment 2

In this section, we address the second objective to compare tree models and non-tree models on the task of design space exploration. We created 14 validation datasets contain 22 features with 30 instances. The input data has the form of input 30 instances for design variable (input) X values equally distributed between 50 and 120. This input set was created based on six existing concept instances provided by an engineer, by incrementing the value of one of its inputs with a predefined step size and within a predefined interval, to explore the response, or impact, on different design objectives (outputs) when varying a specific design variable, which design variable X values are unequally distributed in the range from 50 to 120. In general, the experiment produced as many as 14 different design objectives, but in Experiment 2 we focus on four design objectives.

The four selected design objectives are identified by the engineer as challenging outputs (design objectives CO1 to CO4), i.e., more difficult to predict and of higher priority. One of the design variables is defined by the engineer as the key input (here called design variable X value). These four design objectives and response variables have high importance in order to build a particular part in the flight engine. For example, if the product is aircraft engine, then the design variables can be length, width, curvature etc., and the design objective is to find the shape for aircraft wing. Figure 1 shows four design objectives (sub-plots),

Table 2. Pairwise comparisons

Pairwise comparison	*RMSE* *p-value*	*CC* *p-value*
M5P-RF	0.0002	0.0015
M5P-SVM	0.0394	0.0182
M5P-LR	0.4611	0.0665
LR-SVM	0.6296	0.9667
RF-SVM	0.4611	0.8848
LR-RF	0.0394	0.6296

design variable X values on *x-axis* and response value on *y-axis*. For design objectives CO1, CO3 and CO4, the result of predictions is same for LR and M5P. The first observation is that RF accurately predicts the actual values, at least in the case of design objectives CO1 to CO3. The RF plot appears to have changing trends approximately following that of the labeled dataset (Actual value). The predicted output values of RF are also closest to the actual value for the majority of instances. The other models predicted output values that seems completely monotonic, and appear to almost follow a straight line. For the design objective CO4, SVM fits well to the actual values. These observations indicate an advantage for RF over the other models with regard to fitting the challenging outputs.

6 Conclusions and Future work

The main goal was to investigate the performance of tree models for response surface modeling. We studied two tree methods (M5P and RF) and two non-tree methods (LR and SVM). Experiments were conducted on aerospace concept selection datasets to determine the performance. The results show that tree models perform at least as well as or better than traditional black-box models. We addressed the single-output regression problem for response surface models. Our future work will contrast this work with a multi-output regression approach to explore tree-based surrogate model comprehensibility further.

Acknowledgments
This work was supported by the *Knowledge Foundation* through the research profile grants *Model Driven Development and Decision Support* and *Scalable Resource-efficient Systems for Big Data Analytics*

References

1. Ahmed, M., Qin, N.: Comparison of response surface and kriging surrogates in aerodynamic design optimization of hypersonic spiked blunt bodies. In: 13th International Conference on Aerospace Sciences and Aviation Technology, May 26th–28th, Military Technical College, Kobry Elkobbah, Cairo, Egypt (2009)

2. Bell, T.E., Bixler, D.C., Dyer, M.E.: An extendable approach to computer-aided software requirements engineering. Software Engineering, IEEE Transactions on (1), 49–60 (1977)
3. Breiman, L.: Random forests. Machine learning 45(1), 5–32 (2001)
4. Carley, K.M., Kamneva, N.Y., Reminga, J.: Response surface methodology. Tech. rep., DTIC Document (2004)
5. Chen, K.Y., Wang, C.H.: Support vector regression with genetic algorithms in forecasting tourism demand. Tourism Management 28(1), 215–226 (2007)
6. Couckuyt, I., Gorissen, D., Rouhani, H., Laermans, E., Dhaene, T.: Evolutionary regression modeling with active learning: An application to rainfall runoff modeling. In: Kolehmainen, M., Toivanen, P., Beliczynski, B. (eds.) Adaptive and Natural Computing Algorithms, Lecture Notes in Computer Science, vol. 5495, pp. 548–558. Springer Berlin Heidelberg (2009)
7. Crombecq, K., Couckuyt, I., Gorissen, D., Dhaene, T.: Space-filling sequential design strategies for adaptive surrogate modelling. In: The First International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering (2009)
8. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
9. Freitas, A.A.: Comprehensible classification models: A position paper. SIGKDD Explor. Newsl. 15(1), 1–10 (Mar 2013), http://doi.acm.org/10.1145/2594473.2594475
10. Gorissen, D., Couckuyt, I., Demeester, P., Dhaene, T., Crombecq, K.: A surrogate modeling and adaptive sampling toolbox for computer based design. Journal of Machine Learning Research 11, 2051–2055 (2010)
11. Gramacy, R.B., Le Digabel, S.: The mesh adaptive direct search algorithm with treed Gaussian process surrogates. Groupe d'études et de recherche en analyse des décisions (2011)
12. Gu, H., Yang, L., Hu, Z., Yu, J.: Surrogate models for shape optimization of underwater glider pp. 3–6 (Feb 2009)
13. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: IJCAI. vol. 14, pp. 1137–1145 (1995)
14. Nikolos, I.K.: On the use of multiple surrogates within a differential evolution procedure for high–lift airfoil design. International Journal of Advanced Intelligence Paradigms 5, 319–341 (2013)
15. Pos, A., Borst, P., Top, J., Akkermans, H.: Reusability of simulation models. Knowledge-Based Systems 9(2), 119–125 (1996)
16. Quinlan, J.R., et al.: Learning with continuous classes. In: Proc. of the 5th Australian joint Conference on Artificial Intelligence. vol. 92, pp. 343–348. Singapore (1992)
17. Quinn, G.P., Keough, M.J.: Experimental design and data analysis for biologists. Cambridge University Press (2002)
18. Scholkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2001)
19. Sheskin, D.J.: Handbook of parametric and nonparametric statistical procedures. CRC Press (2003)
20. Vapnik, V.: The nature of statistical learning theory. springer (2000)
21. Wang, Y., Witten, I.H.: Inducing model trees for continuous classes. In: Proceedings of the Ninth European Conference on Machine Learning. pp. 128–137 (1997)
22. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann (2011)