A COMPARISON OF METAMODELING TECHNIQUES ADDRESSING THE NUMBER OF INDEPENDENT VARIABLES AND THE PRESENCE OF NOISY DATA

Sergio Butkewitsch
Empresa Brasileira de Aeronáutica S.A. – Av. Brigadeiro Faria Lima, 2170, São José dos Campos – SP – Brazil – 12227-901
sergio.choze@embraer.com.br

Abstract. *Given the importance of empirical modeling (i.e., metamodeling) to engineering design in general and multidisciplinary design optimization (MDO) in particular, this work is devoted to compare the performance and the accuracy of three widely used metamodeling strategies: response surfaces, neural networks and support vector machines. The comparison focuses on the scalability of each method with the increase of the number of independent variables, as well as their robustness when the database used in their construction is corrupted by noise. The introduction reviews the general framework used to formulate metamodels, emphasizing the statistical nature that underlies all sorts of different available approaches. In the sequence, the particular types of metamodels addressed in this paper are described in deeper detail. This account starts by polynomial response surfaces, whose ease of implementation and spread of use are helpful for the understanding of the fundamentals that guide the remainder of the work. From this basis, a review of artificial neural networks is also carried out, pointing out their conceptual similarities and operational differences with respect to response surfaces. Likewise, a summary of support vector machines is presented. This theoretical review is followed by applications aimed at providing practical insight for the comparison of the metamodels, their performance and accuracy. Thus, response surfaces, artificial neural networks and support vector machines are built up in order to predict the resonant frequencies and natural mode shapes of dynamic systems, given their constructive parameters (stiffnesses and inertias) as independent variables. Within this framework, the number of independent variables is proportional to the number of degrees-of-freedom necessary to describe the motion of each dynamical system. Therefore, enlarged equations of motion also mean that the metamodel is formulated with respect to an increased amount of independent variables. This rationale is used to test the scalability of each metamodeling strategy with the growth of independent variables, that is, how much each type of metamodel needs augmented databases to be constructed as its own dimension is increased. Next, random noise is simulated as to corrupt the database available for the formulation of the metamodels, and the robustness of response surfaces, neural networks and support vector machines is compared. The assessment of the performance and robustness of each metamodeling technique is supported by the availability of closed form solutions for all of the physical quantities whose prediction is sought. All the comparisons derived from the dynamical systems analysis and simulation, as well as the metamodels construction, lead to the conclusions and outline of future research work.*

Keywords*: metamodels, response surfaces, neural networks, support vector machines*

1. Introduction

Optimization techniques are established tools from the theoretical viewpoint and, therefore, find their place in many different kinds of application. Indeed, intensive research activities are performed in order to consolidate and spread their already wide applicability and usefulness.

Recent works (Alexandrov and Kodiyalam, 1998) have emphasized that besides the myriad of algorithms devoted for actually solving optimization problems, their formulation plays an equally important role. The proper combination of formulation and solution methods is then paramount for the successful application of optimization techniques. Often times, it results in the need for testing many alternative formulations, something that demands flexibility and agility to be achieved at reasonable costs.

This calls for the use of metamodels, very flexible representations of design spaces where a given optimization procedure is to be performed. Opposite to general models such as differential equations and the like, metamodels are intended to represent the specific behaviour of a given system with respect to a pre-defined set of parameters. They are constructed from the data gathered by observing the system of interest. So, as far as data provided by numerical procedures and/or physical experiments are processed and used to create symbolic models of a physical reality, metamodeling (also called empirical modeling) techniques are being applied, according to the description in section 2.

2. Framework for the development of metamodels

The construction of metamodels is, indeed, the solution of a parameter identification problem, based on observed data. Some sort of optimization procedure takes place to calculate the most suitable values of the parameters. This allows for the use of a single schema, consisting of four steps, to develop any kind of metamodel:

- Data sampling: a design space, including a range of design possibilities, is sampled in order to reveal its contents and tendencies. Considerate sampling strategies, known as “Design of Experiments’ (the acronym DOE is of standard use), are frequently employed to optimize the ratio between the sampling effort and the insight generation;
- Model choice: the nature of the metamodel itself is determined, tacking into account that the relations contained in the data gathered in the previous step have to be represented, with the highest possible accuracy;
- Model fitting: Some optimization procedure is used to determine the most adequate values of the parameters that represent the model chosen in the preceding step. Therefore, this is the parameter identification itself, and differences in its implementation may affect the efficacy of metamodel ;
- Quality verification: the three foregoing steps are sufficient to build a first tentative model, whose overall quality and usefulness have to be evaluated by adequate sets of metrics. Each combination of design space sampling, model choice and fitting procedure leads to the use of specific verification procedures.

Figure 1 (Butkewitsch, 2001) shows the most common combinations of the aforementioned steps, which lead to the most widely accepted types of metamodels. The primary goal of this work is to explore the advantages and drawbacks of some of these combinations with respect to statistical modeling applied to the solution of engineering problems.

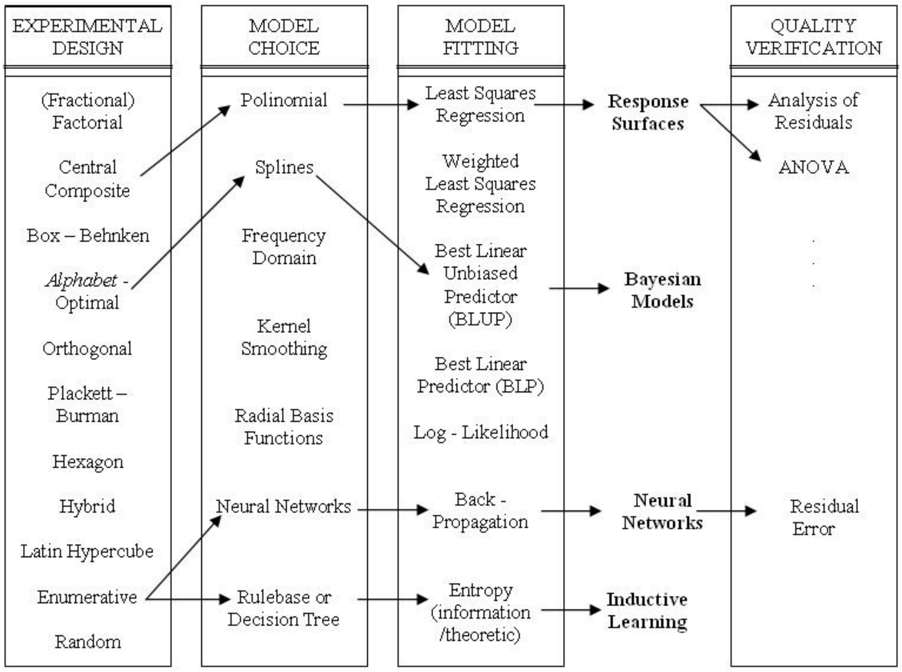


Figure 1. Possible combinations for the development of metamodels

3. Review of selected metamodels

This section is devoted to reviewing the theoretical fundamentals involved in the formulation of the three types of metamodels to be compared throughout the paper: response surface models (or RSM, in sub-section 3.1), artificial neural networks (or ANN, in sub-section 3.2) and support vector machines (or SVM, in sub-section 3.3).

3.1. Response surfaces models (RSM)

In the case of the RSM, the matrix form of the least squares formulation, intended for the analysis in several dimensions, is adopted, as shown in Eq. (1):

$$\{Y\} = [X] \cdot \{B\} + \{d\} \tag{1}$$

where {Y} is the vector of responses (dependent variables) corresponding to each of the rows of matrix [X], that are, on their turn, a combination of independent variables, assembled during the data sampling performed to build the metamodel. The vector {d} contains free, random error terms, whilst the vector of model parameters {B} can be estimated according to Eq. (2). The accuracy of the overall procedure is mostly dependent on the fact that {δ} is supposed to contain independent and identically distributed terms, according to a normal probability density function.

$$\{B\}=\left([X]'\cdot[X]\right)^{-1}\cdot[X]'\cdot\{Y\} \tag{2}$$

Generally, a full quadratic polynomial in the form of Eq. (3) is obtained and used to predict the value of the dependent variables ($\hat{y}$) out coming from brand new combinations of the independent ones (x). The free term b_0 is equivalent to the grand average of the data sampled to build the metamodel.

$$\hat{y}(x)=b_0+\sum b_i\cdot x_i+\sum b_{ii}\cdot x_i^2+\sum b_{ij}\cdot x_i\cdot x_j \tag{3}$$

3.2. Artificial neural networks (ANN)

The general idea of artificial neural networks, proposed to mimic the behaviour of biological neuronal systems, has been split into many alternative forms of implementation. In the present work, radial basis functions networks have been chosen due to their inherent pattern recognition abilities (Bishop, 1995).

In this kind of ANN, the activation of each processing unit, or neuron, is based on the distance between an input (i.e., the independent variable) and the target (i.e., the intended value for the dependent variable, according to a function whose mapping is the goal of the metamodel). It can be interpreted as a scalar form of mapping the projection of each input over its corresponding output, since the distances are computed in a pairwise manner (Chen *et al*., 1991).

Therefore, any estimated output ($\hat{y}$) results from a relationship similar to Eq. (4).

$$\hat{y}(x)=\sum w_i\cdot f(x)\cdot\| x_i-y_i\| \tag{4}$$

where y_i are the targets (outputs observed in a procedure that collects a number of samples), w_i a set of weights and $\phi(x)$ are basis (or radial basis) functions of the inputs, such as the Gaussian function shown in Eq. (5). The parameter σ^2, the variance of the Gaussian distribution, determines if the metamodel is "sharp" (delivering accurate predictions only in the vicinity of the sampled data) or smooth (not necessarily accurate around the known samples, but capable of mapping the overall process that generated them). These opposite tendencies are known as "overfit" and "underfit", and a good compromise between them, through the choice of an adequate value for σ^2, is not always a trivial task.

$$f(x)=\exp\left(-\frac{x^2}{2\cdot s^2}\right) \tag{5}$$

Analogously to RSMs, matrices of inputs and the corresponding output vector can be supplied in a multidimensional context.

3.3. Support Vector Machines (SVM)

Adequate sampling of the design space is essential to build capable metamodels. Usually, larger samples result in increased statistical confidence about the capability of the metamodel, and this relation is amplified with the growth of the number of independent design variables. This phenomenon is often summarized as "the curse of dimensionality", and may prevent the use of metamodels for the representation of design spaces that are functions of a large number (i.e., more than 20) of independent design variables.

However, "the curse of dimensionality" may be worked around if the optimization procedure aimed at identifying the parameters of the metamodel is formulated conveniently. For such a strategy, consider now a vector (as opposed to the scalar approach shown in section 3.2) projection of inputs towards their corresponding outputs (Cherkassky and Mulier, 1998):

$$\hat{y}(x)=\langle w\cdot x\rangle+b \tag{4}$$

where $w\cdot x$ stands for the dot product (i.e., vector projection) between the support vector w and the inputs x, and b a bias term. The following requirements are now imposed over the estimator $\hat{y}$:

- It is a function whose distance from the outputs corresponding to the sampled inputs is no greater than ε (Clarke *et al*., 2003). Figure 2 illustrates this condition graphically:

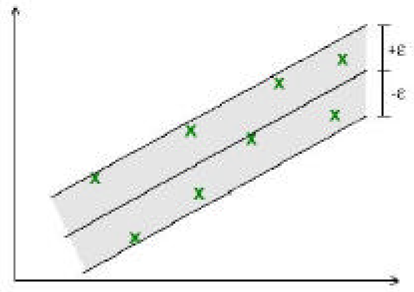


Figure 2. The ε boundary on the distance between the estimator function $\hat{y}$ and the known (sampled) outputs

- Considering that Eq. (4) represents an hyper-plane, it is said to be canonical (Vapnik, 1995) when, for a given input data set x_i, Eq.(5) holds:

$$\min \left\| \langle w \cdot x_i \rangle + b \right\| = 1 \tag{5}$$

Of course, when one thinks on the minimization of the norm of the estimator in Eq. (5), it has to be applied over the support vector w, since the data x_i results from the sampling. With a clear objective to be minimized and imposing the concept of Fig. (2) as a set of constraints, the following optimization problem can be formulated:

$$\begin{aligned} &\min \|w\| \\ &\text{subject to:} \\ &\begin{cases} y_i - \langle w \cdot x_i \rangle - b \le e \\ \langle w \cdot x_i \rangle + b - y_i \le e \end{cases} \end{aligned} \tag{6}$$

The optimization procedure represented in Eq. (6) is the one devoted for the optimal estimation of the parameters (w and b) of the support vector regressor $\hat{y}$. Its performance depends on the initial values chosen for w, as well as how strict is the constraint ε.

The idea of having an optimal hyper-plane for classification, with extensions to regression, is not an exclusiveness of the SVM framework. Indeed, certain classes of ANNs, such as the single and multi-layered Perceptrons, animated by the back-propagation algorithm (not addressed in this paper), also rely on this philosophy.

However, SVMs introduce a new rationale by re-writing the optimization procedure of Eq. (6) in its dual form (Vanderplaats, 1998), as shown in Eq. (7):

$$\begin{aligned} &\max \begin{cases} -\frac{1}{2} \cdot \sum \left(l_i - l_i^*\right) \cdot \left(l_i - l_i^*\right) \cdot \langle x_i \cdot x \rangle \\ -e \cdot \sum \left(l_i + l_i^*\right) + \sum y_i \cdot \left(l_i - l_i^*\right) \end{cases} \\ &\text{subject to:} \\ &\begin{cases} \sum \left(l_i - l_i^*\right) = 0 \\ l_i, l_i^* \ge 0 \end{cases} \end{aligned} \tag{7}$$

where λ_i are the Lagrange Multipliers associated with the constrained optimization procedure, that is solved when their values are equal to λ_i^* . Transforming the optimization problem into its dual form yields two advantages (Clarke *et al.*, 2003). First, the optimization problem is now a quadratic programming problem with linear constraints and a positive definite Hessian matrix, ensuring a unique global optimum. For such problems, highly efficient and thoroughly tested quadratic solvers exist. Second, as can be seen in Eq. (7), the input vectors only appear inside the dot product. The dot product of each pair of input vectors is a scalar, regardless of the dimension of the vectors being multiplied. In this way, the dimensionality of the input space is hidden from the remaining computations, providing means for addressing the "curse of dimensionality", the problem that gave margin to this section about SVMs and their utilization wthin this work.

4. Issues for the comparison of metamodels

Objective criteria have to be selected in order to indicate the advantages and drawbacks of each kind of metamodel, leading to the establishment of guidelines for their application in the context outlined in section 1. For this reason, the following characteristics are investigated in the remainder of this paper, mainly within the development of the case studies described in section 5:

- Scalability, to be understood as the ability to escape from "the curse of the dimensionality", that is, to avoid the need for ever larger sample sizes with the growth of the number of independent design variables. In the scope of this paper, this feature is measured by means of two resources:
    - The definition of two different testing case studies, whose number of independent design variables differ by a factor of more than two;
    - The use of opposite sampling strategies for each of the case studies: both traditional corner sampling DOE (Montgomery, 1991) and space filling DOE (Giunta *et al.*, 2003) are tested, with strongly different sample size for each number of independent variables.

- Robustness, as the capacity of filtering out spurious noise from the observed data, preventing it from deviating the metamodel from the true underlying relationship between inputs and outputs. Random noise levels of 1%, 5% and 10% of the average output values are added to the observed samples and the metamodels are re-estimated in order to evaluate the robustness of each approach (RSM, ANN and SVM).

- Accuracy, classified as follows:
    - Recognition capability, as the ability to recall data points used for the construction of the metamodel;
    - Generalization capability, as the ability of delivering correct predictions for brand new data, apart from the set used to build the metamodel.

A good compromise between both of these characteristics assures that neither "underfit" nor "overfit" can jeopardize the performance of the metamodel.

- Convenience and ease of use, analyzed from two complementary viewpoints:
    - Ease of implementation, accessing the programming necessary to generate the metamodel;
    - Ease of configuration, related to the choice of parameters that may affect the performance of the metamodel once it has already been programmed and is fully available.

5. Description of selected case studies

Both of the case studies intended to test and demonstrate the characteristics of the metamodels being compared consist of the determination of the fundamental vibrating frequency (ω_1) of two dynamic systems. At a very general level, it is possible by means of the Lagrange formalism expressed in Eq. (8):

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i} + \frac{\partial R}{\partial \dot{q}_i} = Q_i \quad (8)$$

where the total kinetic (T) and elastic (U) energies are derived with respect to the generalized coordinates of the cinematic field ($q_i, \dot{q}_i$) to express the equilibrium in face of the generalized forces Q_i.

5.1. Continuous beam with lumped oscillator: 5 independent variables

The system is depicted in Fig. (3), which also shows the assumptions for the lumped oscillator parameters that result in an overall of five independent design variables for this problem, namely: the material's modulus of elasticity (E), the material's volumetric density (ρ), the cross sectional area of the beam (S), the area moment of inertia of the beam's cross section (I) and the length of the beam (L). Figure 3 also displays the closed form solution for the fundamental vibrating frequency, whose complete development can be found in Lalane *et al.* (1984).

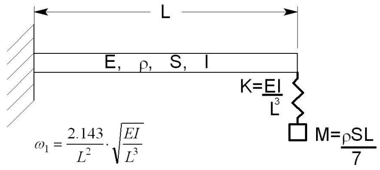


Figure 3. System considered for the first case study: a cantilever beam with a lumped oscillator at the free end

5.2. Seven d.o.f. vehicle: 11 independent variables

A complete application involving this vehicle model is present in the work of Brasil and Colombo (2005), which is used as the source of the formulation necessary to compute ω_1. Figure 4 contains a description of this model, showing its main parameters, also listed in Tab. 1 along with the assumptions made to produce a case study with eleven independent variables.

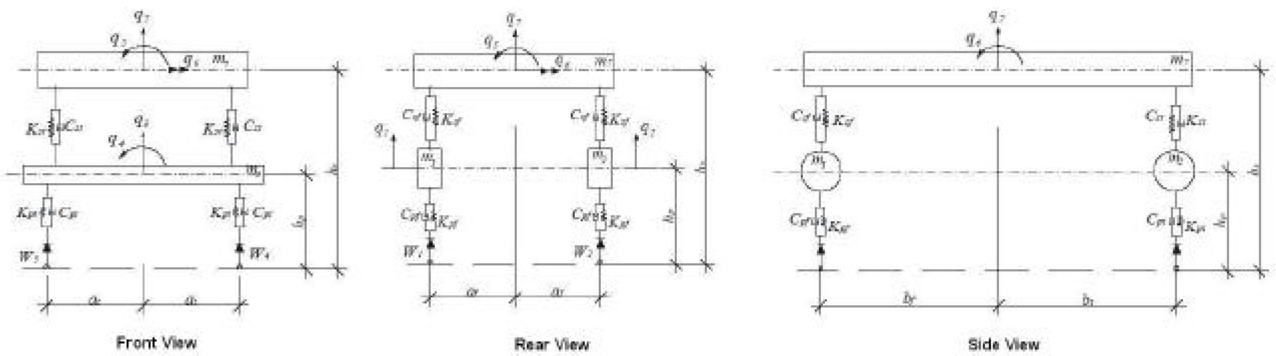


Figure 4. System considered for the second case study: a seven degrees-of-freedom vehicle

Table 1. Parameters of a 7 d.o.f. vehicle model, converted into an 11 design variable case study.

Label	Physical Meaning	Status
m1/ m2	forward right/ left independent suspension mass	4 active independent variables
m3	mass of the rear rigid axle	
m7	Main suspended mass (of the vehicle)	
Ksf/ Kst	Forward/ rear suspension stiffness	4 active independent variables
Kpf/ Kpt	Forward/ rear tire stiffness	
Csf/ Cst	Forward/ rear suspension damping coefficient	4 inactive design variables (Damping is neglected. The response of interest is the first undamped natural frequency)
Cpf/ Cpt	Forward/ rear tire damping coefficient	
af	width of the front suspension	af is made equal to at, resulting in just one additional active independent variable
at	width of the rear suspension	
bf	longitudinal distance from the front suspension to the vehicle cg	2 active independent variables
bt	longitudinal distance from the rear suspension to the vehicle cg	
Total number of active independent design variables		11

6. Results presentation and interpretation

The findings for each comparison criterion are explained within this section and qualitatively summarized in Tab. 2.

- Scalability:

Owing to the sampling power of space filling DOEs for numerical (i.e., deterministic) experiments, a recent trend in statistics, the RSMs scaled outstandingly in the case studies developed. Theoretical and practical evidence, however, suggest that this behaviour tends to be strongly problem dependent. ANNs displayed good scalability, comparable to that of SVMs. Theoretically, this later type of metamodel should have scaled even better than ANNs but, perhaps, the number of variables has to grow beyond the amount tested within this work as to make this supposed advantage appear. A test case with maybe hundreds of variables should be an adequate way to address this question, and a suggestion in this direction is proposed for future research.

As a general rule, observed in both case studies, ANNs and SVMs are worse to break-even than RSMs in terms of sample size, mainly with respect to generalization. Once they do, however, they scale better than RSMs. Given this trend, and that the ability to generalize is more important than the recognition capability for engineering applications, ANNs and SVMs should be used for problems with larger amounts of design variables, and RSMs instead.

- Robustness:

Due to the averaging process underlying the creation of RSMs, they present satisfactory robustness in the presence of spurious noise, being capable of filtering it out while capturing the signal responsible for outlining the design space being metamodeled. Up to 10% noise levels, all the model coefficients for both case studies remained within the 95% confidence interval estimated for the initial, noise free data sets.

ANNs and SVMs, on their turn, tend to interpolate the noise, mainly when their configuration parameters are chosen such that they tend to "overfit". This tendency may be attenuated at the expense of limiting the recognition capabilities of the metamodel.

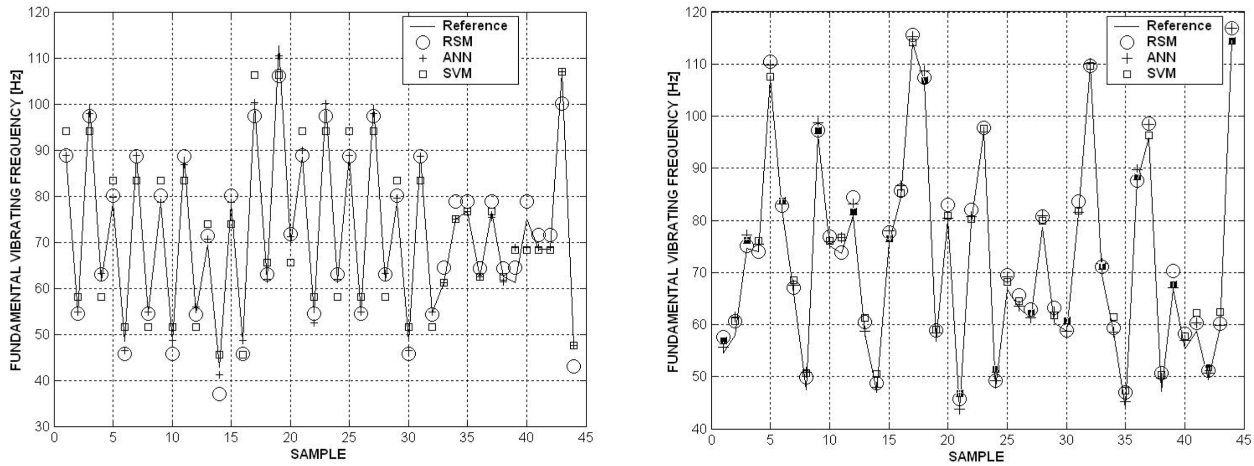
- Accuracy (recognition and generalization):

RSM have displayed good recognition and generalization capabilities provided that the sample size is adequate and that the hypothesis of normally distributed residuals is fulfilled. For the first case study, the average absolute recognition and generalization errors amount 3.08% and 2.68% respectively, whilst for the second these figures correspond to 5.15% and 5.18%. These results are compatible with the explained variances of each model, measured by the adjusted multiple correlation coefficients of 95.05% and 93.48% for the first and second case studies, respectively.

For ANNs and SVMs, it is possible to privilege either recognition or generalization, depending on how their configuration parameters are chosen. However, serious difficulties may arise when one is interested in the optimal balance between recognition and generalization capabilities, as explained in the following topic.

For the best possible setups achieved during the development of this work, the average absolute recognition/generalization errors yielded by ANNs are equal to 1.60% / 2.26% and 3.88% / 2.12% for the first and second test cases, respectively. For SVMs, the corresponding figures are 4.86% / 2.54% and 5.81% / 4.19%.

Figure 5 illustrates how accurately each metamodel behaved for the first case study.



(a) (b)

Figure 5. Recognition (a) and generalization (b) capabilities for each type of metamodel

- Convenience (implementation and configuration):

The multivariate least-squares regression underlying the RSM technique is a simple matrix operation, as shown in Eq. (2), and no special implementation difficulty exists. The coding of ANNs and SVMs, on the other hand, is much more laborious, mainly in the later case, which demands the availability of a powerful optimizer to solve the parameter identification procedure expressed through Eq. (7), whose numerical conditioning is seldom well balanced.

As a direct consequence, RSMs decline any kind of configuration. From one hand, it ensures easy and readiness of use but, on the other, little can be done when the results are poor, except manipulating the sampled database itself and, in extreme cases, choose other sampling strategies. Radial basis ANNs allow for the configuration of the dispersion term σ^2 (see Eq. (5)), in order to balance overfit *versus* unerfit. The use of optimization techniques to automate the choice of σ^2 would increase the implementation difficulty. The configuration of SVMs means choosing proper values for at least two parameters, both belonging to Eq. (7): the initial values for the support vectors w and the amount of allowable dispersion, ε. For this reason, SVMs are even more difficult to configure than radial basis ANNs and, in practice, this means that completely different results may arise from different combinations of configuration parameters, as experienced with both case studies developed within this work.

Table 2. Qualitative summary of comparison criteria, based on the results of the two case studies developed.

	Scalability	Robustness	Recognition	Generalization	Implementation	Configuration
RSM	Uncertain	Good	Data dependent	Data dependent	Very easy	Unnecessary
ANN	Good	Parameter dependent	Parameter dependent	Parameter dependent	Laborious	Laborious
SVM	Good	Parameter dependent	Parameter dependent	Parameter dependent	Very laborious	Very laborious

7. Concluding remarks

This work has explored three of the many possibilities regarding metamodeling tools, which have displayed themselves as powerful data processing techniques, given that they are used within the adequate contexts. Accounting for this scenario, several usage guidelines have been presented.

The main such contextual limitation for RSMs is related to the data available for the construction of the metamodel. Approaches that allow for more flexibility and generality about the probability features of the database, such as generalized linear modeling, may likewise ensure that this technique becomes more flexible and usable. This should be the target of further research effort regarding RSMs.

For ANNs, that have performed suitably in both of the case studies developed, an useful continuation would be to explore more sophisticated architectures, adding a layer of linear neurons to the radial basis functions used in this work.

Considering SVMs and the difficulty to explore its power through proper configuration, it should be suggested that heuristic optimization techniques (i.e., genetic algorithms, particle swarm optimization, etc.) are used to determine the most suitable values for its parameters. This would increase the implementation complexity of SVMs, but the pay-off may be satisfactory. The sizes of the problems addressed in this work were not enough to explore the most alleged advantage of SVMs, which is its theoretical capacity to scale against the so-called "curse of dimensionality". This suggests that, in practice, this tool should be reserved for really large applications.

8. Acknowledgements

The author acknowledges the support of EMBRAER S.A. for his involvement in research activities. Prof. Reyolando M. L. R. F. Brasil is praised for kindly providing the basis for the construction of the second case study.

9. References

Alexandrov, N. M. and Kodiyalam, S., 1998, "Initial Results of an MDO Method Evaluation Study", AIAA-98-4884.
Bishop, C.M., 1995, "Neural Networks for Pattern Recognition", Oxford University Press, NY, USA, 482 p.
Brasil, R. M. L. R. F. and Colombo, A., 2005, "A 7-Degree-of-Freedom Mathematical Model of a Vehicle Subjected to Random Pavement Irregularities", Proceedings of the XI Diname, Ouro Preto, MG, Brazil, pp.
Butkewitsch, S., 2001, "On the use of "meta-models" to account for multidisciplinarity and uncertainty in design analysis and optimization", Proceedings of the First MIT Conference on Computational Fluid and Solid Mechanics, Vol. 1, pp. 599-602.
Chen, S., Cowan, C. F. N. and Grant, P. M., 1991, "Orthogonal Least Squares Learning Algorithm for Radial Basis Functions", IEEE Transactions on Neural Networks, Vol. 2, No. 2, pp. 302-309.
Cherkassky, V. S. and Mulier, F., 1998, "Learning from Data: Concepts, Theory and Methods", John Wiley & Sons, NY, USA, 441p.
Giunta, A. A., Wojtkiewicz Jr., S. F. and Eldred, M. S., 2003, "Overview of Modern Design of Experiments for Computational Simulations", AIAA-2003-0649
Clarke, S. M., Griebsh, J. H. and Simpson, T. W., 2003, "Analysis of Support Vector Regression for Approximation of Complex Engineering Analyses", Proceedings of the of DETC '03, Chicago, USA.
Lalanne, M., Berthier, P. and Der Hagopian, J., 1984, "Mechanical Vibrations for Engineers", John Wiley & Sons, New York, USA, 266 p.
Montgomery, D. C., 1991, "Design and Analysis of Experiments", John Wiley & Sons, New York, USA, 787 p.
Vanderplaats, G.N., 1998, "Numerical Optimization Techniques for Engineering Design", Vanderplaats Research and Development Inc., Colorado Springs, CO, USA, 417 p.
Vapnik, V., 1995, "The Nature of Statistical Learning Theory", Springer, NY, USA, 314 p.

10. Responsibility notice

The author is the only responsible for the printed material included in this paper.