Journal of Information & Knowledge Management, Vol. 5, No. 4 (2006) 329–335

# A Forecasting Methodology Using Support Vector Regression and Dynamic Feature Selection

José Guajardo[*,‡], Richard Weber[*,§] and Jaime Miranda[†,¶]
[*]*Department of Industrial Engineering*
*University of Chile, Chile*
[†]*Department of Industrial Engineering*
*University Diego Portales, Chile*
[‡]*jguajardo@dii.uchile.cl*
[§]*rweber@dii.uchile.cl*
[¶]*jmiranda@dii.uchile.cl*

**Abstract.** Various techniques have been proposed to forecast a given time series. Models from the ARIMA family have been successfully used, as well as regression approaches based on e.g. linear, non-linear regression, neural networks, and Support Vector Regression. What makes the difference in many real-world applications, however, is not the technique but an appropriate forecasting methodology. Here, we propose such a methodology for the regression-based forecasting approach. A hybrid system is presented that iteratively selects the most relevant features and constructs the regression model optimizing its parameters dynamically. We develop a particular technique for feature selection as well as for model construction. The methodology, however, is a generic one providing the opportunity to employ alternative approaches within our framework. The application to several time series underlines its usefulness.

*Keywords*: Support vector regression; time series forecasting; feature selection.

## 1. Introduction

Forecasting problems are commonly faced by using two different approaches: time series models -such as ARIMA models or exponential smoothing- on one hand, or on the other hand, regression models such as linear or nonlinear regression, neural networks, regression trees, or Support Vector Regression. Hybrid models combining both approaches have been applied e.g. in (Aburto and Weber, 2007).

We present a forecasting methodology combining dynamic feature selection with regression models where we propose Support Vector Regression for model construction. Our methodology, however, is independent of the particular regression model, i.e., any other regression approach can be used within the proposed framework.

Section 2 provides a literature review related to feature selection and support vector regression. In Section 3 we develop the proposed forecasting methodology. Section 4 presents the results derived by our methodology as well as using alternative approaches. Section 5 concludes this work and points at future developments.

## 2. Review of the Related Literature

The aim of this section is to provide a review of the major approaches for feature selection, as well as to describe the support vector regression algorithm. Both issues are important to understand the methodology we are proposing.

### 2.1. *Feature selection*

Given the complexity of certain data mining applications, such as e.g. regression, it is desirable to select the most important features to construct the respective model.

This problem motivates efforts in developing feature selection methods. Some of the benefits that can be obtained by performing feature selection are (Guyon and Elisseeff, 2003; Rakotomamonjy, 2002; Famili *et al.*, 1997):

- improvements in the accuracy of the model,
- reduction of the computational times involved in model construction,
- facilitation of data visualization and model understanding, and
- reduction in the risk of overfitting.

Several approaches have been developed to carry out feature selection, being possible to group them into three major categories (Guyon and Elisseeff, 2003): filters, wrappers and embedded methods.

***Filter methods*** perform feature selection as a preprocessing step, independently of the learning algorithm used for model construction. An example of such a mechanism is variable ranking, using e.g., correlation coefficients between each feature and the dependent variable. Another filter approach selects features based on a linear model (this corresponds to the filter preprocessing step), and then constructs a non-linear model using the selected features (see e.g., Bi *et al.*, 2003). As can be seen, filter mechanisms do not depend on the regression algorithm applied, and consequently the selected features are not related to it. This is one of their main weaknesses, but they are commonly applied for simplicity reasons. Also, filter methods in general do not take into account the problem of multicollinearity.

On the other hand, ***wrapper methods*** define a subset selection approach. Their main idea is to assess subsets of variables according to their usefulness to a given learning algorithm (Guyon and Elisseeff, 2003). Wrapper methods became popular by works such as Kohavi and John (1997). Here, the learning algorithm is treated as a black box, and the *best* subset of features is determined according to the performance of the particular algorithm applied to build a regression model (e.g., linear or nonlinear regression, neural networks, support vector regression, among others). Wrapper methods need a criterion to compare the performance of different feature subsets (e.g., minimization of the mean absolute training error), as well as a search strategy to guide the process.

***Forward selection*** and ***backward elimination*** are two of the commonly used search strategies. A ***forward selection*** strategy starts with an empty set of features and incorporates in each iteration features considering their conjoint predictive power. On the other hand, a ***backward elimination*** procedure starts with all available features belonging to the set of selected features; in each iteration, the feature which produces the minimum decrease in predictive performance is eliminated. Both strategies need a stopping criterion in order to determine the "best" feature subset.

Wrappers are criticized because they seem to be "brut force" requiring exhaustive computation (Guyon and Elisseeff, 2003), but at the same time have the advantage of taking into account the performance of the predictive model to be used, as well as evaluating how a given subset of features performs as a whole.

Finally, ***embedded methods*** incorporate feature selection as part of the training process, i.e., feature selection is done when building the predictive model. Such mechanisms usually involve changes in the objective function of the applied learning algorithm and therefore are commonly associated to a specific predictor. Examples of embedded methods are decision trees (Breiman *et al.*, 1984; Quinlan, 1993), and the mechanisms developed in Miranda *et al.*, 2005; Rakotomamonjy, 2002 and Weston *et al.*, 2003. The methodology proposed in this paper uses wrapper methods and applies forward selection to guide the search strategy.

### 2.2. *Support Vector Regression*

Here we describe the standard Support Vector Regression (SVR) algorithm, which uses the $\varepsilon$-insensitive loss function, proposed by Vapnik (1995). This function allows a tolerance degree to errors not greater than $\varepsilon$. The description is based on the terminology used in Smola and Schölkopf (1998) and Müller *et al.* (1999).

Let $(\vec{x_1}, y_1), \ldots, (\vec{x_n}, y_n)$, where $\vec{xi} \in \Re^n$ and $yi \in \Re$ $\forall i$, be the training data points available to build a regression model. The SVR algorithm applies a transformation function $\Phi$ to the original data points[1] from the initial *Input Space*, to a generally higher-dimensional *Feature Space* ($F$). In this new space, we construct a linear model, which corresponds to a non-linear model in the original space:

$$\begin{gathered}\Phi : R^n \to F, w \in F \\ f(x) = \langle w, \Phi(x) \rangle + b\end{gathered} \tag{1}$$

The goal when using the $\varepsilon$-insensitive loss function is to find a function that fits current training data with a deviation less or equal to $\varepsilon$, and at the same time is as flat as possible. This means that one seeks for a small weight vector $\vec{w}$, e.g. by minimizing the norm $\|\vec{w}\|^2$ (Smola and Schölkopf, 1998). The following optimization problem is stated for such purpose:

$$\text{Min } \frac{1}{2}\|w\|^2 \tag{2}$$

s.t.

$$y_i - \langle w, \Phi(x_i) \rangle - b \le \varepsilon \tag{3}$$

$$\langle w, \Phi(x_i) \rangle + b - y_i \le \varepsilon \tag{4}$$

This problem could be infeasible. Therefore, slack variables $\xi_i$, $\xi_i^*$ are introduced to allow error levels greater than $\varepsilon$, arriving to the formulation proposed in Vapnik (1995):

$$\text{Min } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*) \tag{5}$$

s.t.

$$y_i - \langle w, \Phi(x_i) \rangle - b \le \varepsilon + \xi_i \tag{6}$$

[1] When the identity function is used, i.e. $\Phi(x) \to X$, no transformation is carried out and linear SVR models are obtained.

$$\langle w, \Phi(x_i)\rangle + b - y_i \le \varepsilon + \xi_i^* \tag{7}$$

$$\xi_i, \xi_i^* \ge 0, \quad i = 1, 2, \ldots, \ell. \tag{8}$$

This is known as the primal problem of the SVR algorithm. The objective function takes into account generalization ability and accuracy in the training set, and embodies the structural risk minimization principle (Vapnik, 1998). Parameter $C$ measures the trade-off between generalization ability and accuracy in the training data, and parameter $\varepsilon$ defines the degree of tolerance to errors.

To solve the problem stated in Eq. (5), it is more convenient to represent the problem in its dual form. For this purpose, a Lagrange function is constructed, and once applying saddle point conditions, the following dual problem is obtained:

$$\begin{aligned}
&\text{Max} -\frac{1}{2}\sum_{i,j=1}^{\ell}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle \Phi(x_i), \Phi(x_j)\rangle \\
&\quad -\varepsilon\sum_{i=1}^{\ell}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{\ell} y_i(\alpha_i - \alpha_i^*) \\
&\quad \text{s.t.} \\
&\quad \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*) = 0 \\
&\quad \alpha_i, \alpha_i^* \in [0, C].
\end{aligned} \tag{9}$$

The solution of this quadratic optimization problem will be function of the dual variables $\alpha_i$ and $\alpha_i^*$; it can be shown that the following relationships hold (Vapnik, 1998):

$$w = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)\Phi(x_i) \tag{10}$$

$$f(x) = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)K(x_i, x) + b \tag{11}$$

Here, the expression $K(x_i, x)$ is equal to $\langle \Phi(x_i), \Phi(x)\rangle$, which is known as the *Kernel Function* (Vapnik, 1998). The existence of such a function allows us to obtain a solution for the original regression problem, without considering explicitly the transformation $\Phi(x)$ applied to the data.

## 3. Development of the Proposed Forecasting Methodology

To build forecasting models, one has to deal with different issues such as feature selection, model construction, and model updating. In this section we present a framework to build SVR-based forecasting models, which takes into account all these tasks. First, a general description of this framework is provided; then, we present details about SVR model construction, feature selection, and model updating.

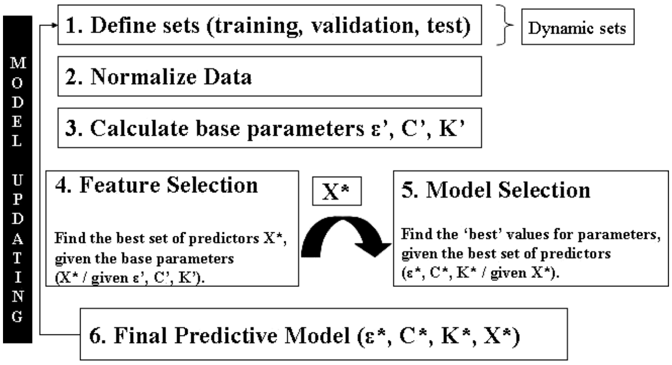
MODEL UPDATING
1. Define sets (training, validation, test)
Dynamic sets
2. Normalize Data
3. Calculate base parameters ε', C', K'
4. Feature Selection
Find the best set of predictors X*, given the base parameters (X* / given ε', C', K').
X*
5. Model Selection
Find the 'best' values for parameters, given the best set of predictors (ε*, C*, K* / given X*).
6. Final Predictive Model (ε*, C*, K*, X*)

Fig. 1. SVM-UP forecasting methodology.

### 3.1. *General framework of the proposed methodology*

Figure 1 provides a general view of our framework for SVR-based forecasting.

The first step consists of dividing the data into training, validation and test subsets. Training data will be used to construct the model, validation data is used for model and feature selection, and test data is a completely independent subset, which is helpful to provide an estimation of the error level that the model would have in reality. As we perform model updating, these subsets will be dynamically redefined over time.

The proposed methodology can be summarized as follows: first, calculate base parameters ($\varepsilon'$, $C'$ and kernel parameter $K'$) that "work well" under some general conditions. Next, and using these base parameters, perform feature selection using a wrapper method with forward selection strategy to obtain the best set of predictor variables $X^*$. Finally, using predictors $X^*$, return to the problem of model construction performing grid search (Chang and Lin, 2001) around base parameters to get the optimal parameters $\varepsilon^*$, $C^*$, $K^*$. Thus, at the end, the predictive model is determined by parameters $\varepsilon^*$, $C^*$, kernel function $K^*$ and predictors $X^*$.

The final element in this framework is model updating. We defined a strategy to update our models and their parameters dynamically over time.

### 3.2. *Model construction using Support Vector Regression within the proposed methodology*

A critical issue in SVR is the selection of the model parameters $\varepsilon'$ and $C'$, as well as the kernel function used to encode data to a higher dimensional space.

As discussed in Section 3.1, our approach to deal with model construction calculates initial values for SVR

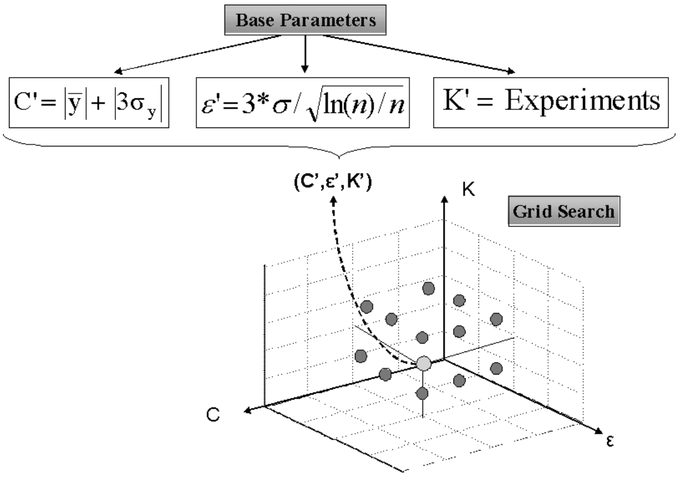

Fig. 2. SVM model selection.

parameters, and then performs grid search around them. This structure is shown in Fig. 2.

To calculate initial values for parameters $\varepsilon'$ and $C'$, we use the empirical rules proposed by Cherkassky and Ma (2004); see (Fig. 2). Experiments using various time series performed by the authors of this paper show that these rules are more effective than other empirical rules available in literature, like e.g. those proposed in Mattera and Haykin (1999).

With regard to the kernel function, we use the RBF kernel transformation to the original data, since this function has performed well under some general conditions (Smola and Schölkopf, 1998), and it is the most commonly used in practice; see e.g., Cherkassky and Ma (2005); Demiriz *et al.* (2001) and Momma and Bennett (2002). To set an appropriate value for the parameter of this kernel function ($\sigma$), we carry out some exploratory experiments trying different settings, and choosing the best one in terms of performance accuracy.

### 3.3. *Feature selection within the proposed methodology*

As we discussed in Section 2.1, there are several feature selection approaches in the context of time series prediction. We propose a wrapper method inspired by ideas presented in Kohavi and John (1997), which selects features guided by a forward selection strategy, as shown in Fig. 3.

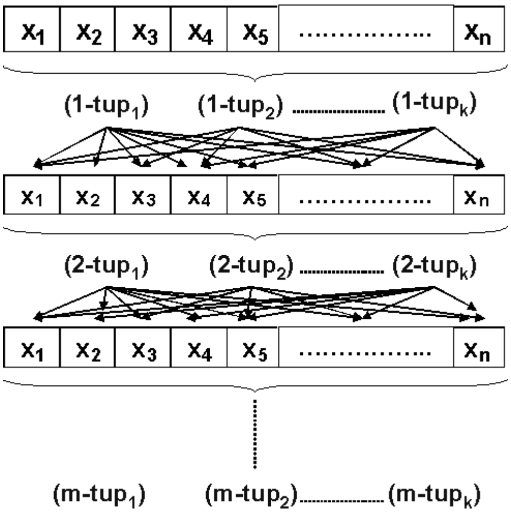
x1 x2 x3 x4 x5 ................ xn
(1-tup1) (1-tup2) ................ (1-tupk)
x1 x2 x3 x4 x5 ................ xn
(2-tup1) (2-tup2) ................ (2-tupk)
x1 x2 x3 x4 x5 ................ xn
(m-tup1) (m-tup2) ................ (m-tupk)

Fig. 3. Feature selection strategy.

Given the set of initial features $(x_1, \ldots, x_n)$, we define a maximum number of predictors $(m \leq n)$ to be included in the predictive model, taking into consideration the nature of the problem and the number of training data points. Also, the number of tuples[2] to be 'saved' in each iteration $(k)$ has to be defined. The greater the value of $k$, the greater the size of the search space induced by the strategy. Once defined both values, the strategy starts building models with each individual variable as a single predictor. In the second iteration, we mix the $k$ best individual predictors (1-*tuple*) with the remaining variables, and keep the $k$ best 2-*tuples* of predictors for the next iteration, and so on until we have the $k$ best $m$-*tuples* of predictors. Finally, we choose the best tuple of variables among the $m \cdot k$ best $i$-*tuples* computed in the iterative process $(i : 1, \ldots, m)$, to be the predictors $X^*$ that will be used in the final SVR model.

### 3.4. *Model updating within the proposed methodology*

The task of developing forecasting models for a time series problem could be affected by changes in the phenomena we are studying over time. Therefore a static model that works well in some period could provide poor predictions in some future period. To deal with this issue, we designed a way to address model updating when building a predictive model, in which the basic idea is to define a two part training set: the first part contains historical data, and the second part contains the most recent data. When a predefined number of new samples arrive, these observations are incorporated into the training set. This way patterns in new data are taken into account in model construction. Finally, the proportion of data belonging to the training and validation sets is kept stable over time by shifting data points from the historical part of the training set to the validation set, when we perform model updating.

[2] In this context, an $i$-tuple means a combination of $i$ different predictors used together in a predictive model.

## 4. Experiments and Results

We applied the proposed methodology to a real-world sales prediction problem, in which a company wants to predict the number of units that will be sold the following week, for 5 different products.

Consequently, we analyzed 5 time series representing weekly sales data, during the period that runs from January 2001 to September 2004 each. Figure 4 shows one of these 5 series, in which it can be seen that there is a strong monthly seasonal pattern, similarly to what occur in the rest of the series.

We applied the proposed methodology to the 5 time series using the following set of 23 initial features (independent variables). As a result we obtained for each series a different set of parameters describing the respective model. Following the wrapper strategy described previously, we also determined for each series a set of selected features. This way we defined the overall relevance of each one of the original features by counting how often each feature has been selected by the applied methodology. By doing this, we found that the most relevant features obtained using the SVM-UP methodology were lags 1, 2 and 8 of the series, together with a seasonal index and a binary variable indicating the type of month (month with four or five weeks).

Besides the application of our methodology using support vector machines (SVM-UP), we utilized the same framework but now with neural networks (NN-UP). Also, we developed ARMAX models to have a standard forecasting method as benchmark.

Tables 1, 2 and 3 show the accuracy error measures of mean absolute percentage error (MAPE), mean absolute error (MAE) and root mean squared error (RMSE), obtained over the test set, by using the 3 forecasting methods for predicting one period ahead sales for the five products.

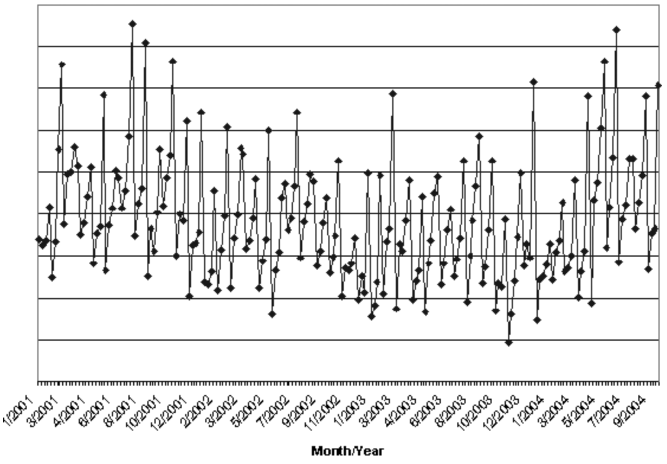

Fig. 4. Time series with weekly sales data.

Table 1. Mean absolute percentage error (the best result for each product is underlined).

| | Test set | | |
|---|---|---|---|
| Product | ARMAX models | NN-UP models | SVM-UP models |
| P1 | <u>10.98</u> | 14.31 | 11.86 |
| P2 | 13.64 | 14.66 | <u>11.44</u> |
| P3 | 6.87 | 6.66 | <u>6.51</u> |
| P4 | <u>10.53</u> | 10.56 | 11.01 |
| P5 | 10.90 | <u>9.85</u> | 9.98 |
| Average (5 Prod.) | 10.58 | 11.21 | <u>10.16</u> |

Table 2. Mean absolute error (the best result for each product is underlined).

| | Test set | | |
|---|---|---|---|
| Product | ARMAX models | NN-UP models | SVM-UP models |
| P1 | <u>292</u> | 350 | 342 |
| P2 | 347 | 368 | <u>283</u> |
| P3 | 103 | <u>89</u> | 96 |
| P4 | <u>268</u> | 288 | 284 |
| P5 | 328 | 280 | <u>264</u> |
| Average (5 Prod.) | 275 | 275 | <u>254</u> |

Table 3. Root mean squared error (the best result for each product is underlined).

| | Test set | | |
|---|---|---|---|
| Product | ARMAX models | NN-UP models | SVM-UP models |
| P1 | <u>446</u> | 462 | 535 |
| P2 | 477 | 420 | <u>352</u> |
| P3 | 143 | <u>126</u> | 138 |
| P4 | 380 | 440 | <u>375</u> |
| P5 | 562 | 375 | <u>354</u> |
| Average (5 Prod.) | 401 | 364 | <u>350</u> |

Though different error measures provide contradictory results in some specific products, the average error levels confirm that the proposed SVM-UP methodology consistently outperforms ARMAX and NN-UP results, for all the three error measures of MAPE, MAE and RMSE. This confirms the ability of the proposed methodology to provide accurate forecasts.

The advantage of the proposed methodology is not only limited to the forecasting performance, but also provides the capability of selecting the most relevant features and updating the respective model periodically. This increases its potential for solving practical business forecasting problems, by permitting the adjustment of forecasting models to continuous changes in the respective environment.

## 5. Conclusions and Future Works

We presented the methodology SVM-UP for time series forecasting, which combines a wrapper method with forward selection strategy to perform feature selection, and regression model construction using Support Vector Machines. Model selection is based on calculating initial values for the SVR parameters and then performing grid search around them. The final component of our methodology is model updating: we define a training set formed by past and most recent information; when a predefined number of new observations arrives, this most recent information is incorporated into the training set, and the proportion of data belonging to the training and validation sets is kept stable over time by shifting data points from the historical part of the training set to the validation set.

We have applied the proposed methodology using SVR as well as neural networks as regression models to a sales forecasting problem and compared its performance to a standard ARMAX approach. Comparing the respective results shows that our methodology performs slightly better and additionally provides a selection of the most important features. This last point increases the comprehension of the phenomenon we are studying, and could be useful to provide a better understanding of the regression model. Major advantages of the proposed methodology are expected when dealing with dynamic phenomena, where the performance of a forecasting model could be significantly improved by performing model updating.

Future work has to be done for predicting non-seasonal time series and for selecting the most appropriate parameters of the kernel function based on theoretical approaches.

## Acknowledgment

This work has been supported partially by the Millennium Nucleus "Complex Engineering Systems" (www.sistemasdeingenieria.cl) and the Chilean Fondecyt project 1040926. The first author (José Guajardo) would like to thank Penta Analytics SA, a Chilean consulting company where he carried out a part of this work during 2004.

## References

Aburto, L and R Weber (2007). Improved supply chain management based on hybrid demand forecasts. *Applied Soft Computing*, 7, 136–144, forthcoming.

Bi, J, KP Bennett, M Embrechts, C Breneman and M Song (2003). Dimensionality reduction via sparse support vector machines. *Journal of Machine Learning Research*, 3, 1229–1243.

Breiman, L, JH Friedman, RA Olshen and CJ Stone (1984). *Classification and Regression Trees.* Monterey: Wadsworth & Brooks.

Chang, C and C Lin (2001). *LIBSVM: A Library for Support Vector machines*, http://www.csie.ntu.edu. tw/ ∼ cjlin/libsvm.

Cherkassky, V and Y Ma (2004). Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, 17(1), 113–126.

Cherkassky, V and Y Ma (2005). Multiple model regression estimation. *IEEE Transactions on Neural Networks*, 16(4), 785–798.

Demiriz, A, K Bennett, C Breneman and M Embrechts (2001). Support vector machine regression in chemometrics. *Computing Science and Statistics.*

Famili, A, W-M Shen, R Weber and E Simoudis (1997). Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, 1(1), 3–23.

Guyon, I and A Elisseeff (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3, 1157–1182.

Kohavi, R and GH John (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1–2), 273–324.

Mattera, D and S Haykin (1999). Support vector machines for dynamic reconstruction of a chaotic system. In *Advances in Kernel Methods: Support Vector Machine*, B Schlkopf, J Burges, and A Smola (eds.), MIT Press.

Miranda, J, R Montoya and R Weber (2005). Linear penalization support vector machines for feature selection. *Lecture Notes in Computer Science*, 3776, 188–192.

Momma, M and KP Bennett (2002). A pattern search method for model selection of support vector regression. In *Proc. SIAM Conference on Data Mining.*

Müller, K, A Smola, G Rätsch, B Schölkopf, A Kohlmorgen and V Vapnik (1999). Using support vector machines for time series prediction. In, *Advances in Kernel Methods: Support Vector Machine*, B Schölkopf, J Burges and A Smola (eds.), MIT Press.

Quinlan, R (1993). *C4.5: Programs for Machine Learning.* San Mateo, Morgan Kaufmann Publishers.

Rakotomamonjy, A (2002). Variable selection using SVM-based criteria. *Journal of Machine Learning Research*, 2, 1357–1370.

Smola, AJ and B Schölkopf (1998). A tutorial on support vector regression. NeuroCOLT Technical Report NC-TR-98-030, Royal Holloway College, University of London, UK.

Vapnik, V (1995). *The Nature of Statistical Learning Theory.* New York: Springer-Verlag.

Vapnik, V (1998). *Statistical Learning Theory.* New York, NY: John Wiley and Sons.

Weston, J, A Elisseff, B Schölkopf and M Tipping (2003). Use of the zero norm with linear models and kernel methods. *JMLR*, 3, 1439–1461.

**José Guajardo** is an Industrial Engineer and Master in Operations Management from the University of Chile. He works as a consultant in Business Intelligence for the Division of External Projects in the Department of Industrial Engineering, at the University of Chile. He has participated in several consultancy projects in the topics of demand forecasting and credit risk management. His principal research interests are related to data mining, time series forecasting, support vector regression, demand planning and credit scoring. His research, mainly focussed on computational intelligence methods for time series forecasting, have been presented at several international conferences. He has been invited as a guest researcher by Lancaster University and the University of Hamburg.

**Richard Weber** is an Associate Professor at the Department of Industrial Engineering of the University of Chile and Director of the Master Programme of Operations Management at the same university. He holds a diploma in mathematics, a master and PhD in operations research from the University of Aachen, Germany. Dr. Weber has been a member of the programme committee of several international conferences, has served on the editorial board of international journals, and has published more than 60 papers in his research areas. His research interests include data mining, dynamic data mining, computational intelligence and hybrid systems. Dr. Weber has been a visiting professor at the University of Osaka Prefecture, The University of Tokyo and the University of Alberta, Canada. As a consultant in Business Intelligence, he has realised projects for private and public institutions in Latin America, Europe and Asia.

**Jaime Miranda** is an Industrial Engineer and holds a Master in Operations Management from the University of Chile. Currently he is realising the PhD programme on Engineering Systems and working as a Consultant in Business Intelligence and a full Professor in Industrial Engineering at the Diego Portales University. His research interests include data mining models for segmentation, classification and regression applied to customer retention, fraud detection and credit scoring, among others. He has presented several articles at international conferences and implemented Business Intelligence systems in the financial sector in Chile.