Applying REC Analysis to Ensembles of Sigma-Point Kalman Filters

Aloísio Carlos de Pina LONG@COS.UFRJ.BR
Gerson Zaverucha GERSON@COS.UFRJ.BR
Department of Systems Engineering and Computer Science, COPPE/PESC, Federal University of Rio de Janeiro, C.P.68511 - CEP. 21945-970, Rio de Janeiro, RJ, Brazil

Abstract

The Sigma-Point Kalman Filters (SPKF) is a family of filters that achieve very good performance when applied to time series. Currently most researches involving time series forecasting use the Sigma-Point Kalman Filters, however they do not use an ensemble of them, which could achieve a better performance. The REC analysis is a powerful technique for visualization and comparison of regression models. The objective of this work is to advocate the use of REC curves in order to compare the SPKF and ensembles of them and select the best model to be used.

1. Introduction

In the past few years, several methods for time series prediction were developed and compared. However, all these studies based their conclusions on error comparisons.

Results achieved by Provost, Fawcett and Kohavi (1998) raise serious concerns about the use of accuracy, both for practical comparisons and for drawing scientific conclusions, even when predictive performance is the only concern. They indicate ROC analysis (Provost & Fawcett, 1997) as a superior methodology than the accuracy comparison in the evaluation of classification learning algorithms. Receiver Operating Characteristic (ROC) curves provide a powerful tool for visualizing and comparing classification results. A ROC graph allows the performance of multiple classification functions to be visualized and compared simultaneously and the area under the ROC curve (AUC) represents the expected performance as a single scalar.

But ROC curves are limited to classification problems. Regression Error Characteristic (REC) curves (Bi & Bennett, 2003) generalize ROC curves to regression with similar benefits. As in ROC curves, the graph should characterize the quality of the regression model for different levels of error tolerance.

The Sigma-Point Kalman Filters (SPKF) (van der Merwe & Wan, 2003) is a family of filters based on derivativeless statistical linearization. It was shown that Sigma-Point Kalman Filters achieve very good performance when applied to time series (van der Merwe & Wan, 2003).

Current research on time series forecasting mostly relies on use of Sigma-Point Kalman Filters, achieving high performances. Although most of these works use one of the filters from the SPKF family, they do not use an ensemble (Dietterich, 1998) of them, which could achieve a better performance. Therefore, the main goal of this paper is to advocate the use of REC curves in order to compare ensembles of Sigma-Point Kalman Filters and choose the best model to be used with each time series.

This paper is organized as follows. The next section has a brief review of REC curves. Then, a summary of the main characteristics of the Sigma-Point Kalman Filters is presented in Section 3. An experimental evaluation comparing the REC curves provided by each algorithm and ensembles of them is reported in Section 4. Finally, in Section 5, the conclusions and the plans for future research are presented.

2. Regression Error Characteristic Curves

Results achieved by Provost, Fawcett and Kohavi (1998) indicate ROC analysis (Provost & Fawcett, 1997) as a superior methodology to the accuracy comparison in the evaluation of classification learning algorithms. But ROC curves are limited to classification problems. Regression Error Characteristic (REC) curves (Bi & Bennett, 2003) generalize ROC curves to regression with similar benefits.

The REC curve is a technique for evaluation and comparison of regression models that facilitates the visualization of the performance of many regression functions simultaneously in a single graph. A REC graph contains one or more monotonically increasing curves (REC curves), each corresponding to a single regression model.

Appearing in Proceedings of the third Workshop on ROC Analysis in Machine Learning, Pittsburgh, USA, 2006. Copyright 2006 by the author(s)/owner(s).

One can easily compare many regression functions by examining the relative position of their REC curves. The shape of the curve reveals additional information that can be used to guide modeling.

REC curves plot the error tolerance on the *x*-axis and the accuracy of a regression function on the *y*-axis. Accuracy is defined as the percentage of points predicted within the tolerance. A good regression function provides a REC curve that climbs rapidly towards the upper-left corner of the graph, in other words, the regression function achieves high accuracy with a low error tolerance.

In regression, the residual is the analogous concept to the classification error in classification. The residual is defined as the difference between the predicted value *f*(x) and actual value *y* of response for any point (x, *y*). It could be the squared error $(y - f(x))^2$ or absolute deviation $| y - f(x) |$ depending on the error metric employed. Residuals must be greater than a tolerance *e* before they are considered as errors.

The area over the REC curve (AOC) is a biased estimate of the expected error for a regression model. It is a biased estimate because it always underestimates the actual expectation. If *e* is calculated using the absolute deviation (AD), then the AOC is close to the mean absolute deviation (MAD). If *e* is based on the squared error (SE), the AOC approaches the mean squared error (MSE). The evaluation of regression models using REC curves is qualitatively invariant to the choices of error metrics and scaling of the residual. The smaller the AOC is, better the regression function will be. However, two REC curves can have equal AOC's but have different behaviors. The one who climbs faster towards the upper-left corner of the graph (in other words, the regression function that achieves higher accuracy with a low error tolerance) may be preferable. This kind of information can not be provided by the analysis of an error measure.

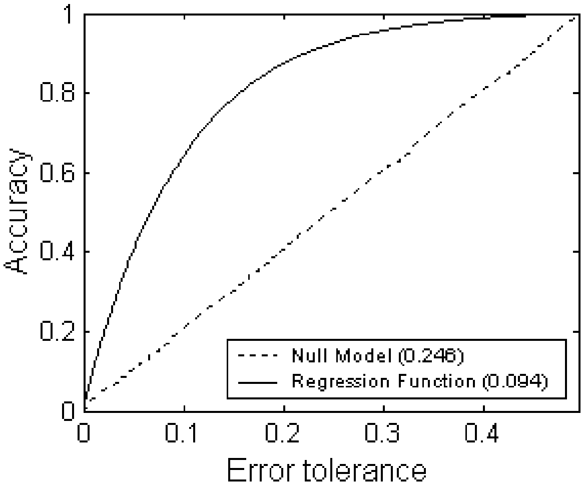

Figure 1. Example of REC graph.

In order to adjust the REC curves in the REC graph, a null model is used to scale the REC graph. Reasonable regression approaches produce regression models that are better than the null model. The null model can be, for instance, the mean model: a constant function with the constant equal to the mean of the response of the training data.

An example of REC graph can be seen in Figure 1. The number between parentheses in the figure is the AOC value for each REC curve. A regression function dominates another one if its REC curve is always above the REC curve corresponding to the other function. In the figure, the regression function dominates the null model, as should be expected.

3. Sigma-Point Kalman Filters

It is known that for most real-world problems, the optimal Bayesian recursion is intractable. The Extended Kalman Filter (EKF) (Jazwinsky, 1970) is an approximate solution that has become one of the most widely used algorithms with several applications.

The EKF approximates the state distribution by a Gaussian random variable, which is then propagated through the "first-order" linearization of the system. This linearization can introduce large errors which can compromise the accuracy or even lead to divergence of any inference system based on the EKF or that uses the EKF as a component part.

The Sigma-Point Kalman Filters (SPKF) (van der Merwe & Wan, 2003), a family of filters based on derivativeless statistical linearization, achieve higher performance than EKF in many problems and are applicable to areas where EKFs can not be used.

Instead of linearizing the nonlinear function through a truncated Taylor-series expansion at a single point (usually the mean value of the random variable), SPKF rather linearize the function through a linear regression between *r* points, called sigma-points, drawn from the prior distribution of the random variable, and the true nonlinear functional evaluations of those points. Since this statistical approximation technique takes into account the statistical properties of the prior random variable the resulting expected linearization error tends to be smaller than that of a truncated Taylor-series linearization.

The way that the number and the specific location of the sigma-points are chosen, as well as their corresponding regression weights, differentiate the SPKF variants from each other. The SPKF Family is composed by four algorithms: Unscented Kalman Filter (UKF), Central Difference Kalman Filter (CDKF), Square-root Unscented Kalman Filter (SR-UKF) and Square-root Central Difference Kalman Filter (SR-CDKF).

Now we will present a brief overview of the main characteristics of the Sigma-Point Kalman Filters. See (van der Merwe & Wan, 2003) for more details.

3.1 The Unscented Kalman Filter

The Unscented Kalman Filter (UKF) (Julier, Uhlmann & Durrant-Whyte, 1995) derives the location of the sigma-points as well as their corresponding weights so that the sigma-points capture the most important statistical properties of the prior random variable *x*. This is achieved by choosing the points according to a constraint equation which is satisfied by minimizing a cost-function, whose purpose is to incorporate statistical features of *x* which are desirable, but do not necessarily have to be met. The necessary statistical information captured by the UKF is the first and second order moments of *p*(*x*).

3.2 The Central Difference Kalman Filter

The Central Difference Kalman Filter (CDKF) (Ito & Xiong, 2000) is another SPKF implementation, whose formulation was derived by replacing the analytically derived first and second order derivatives in the Taylor series expansion by numerically evaluated central divided differences. The resulting set of sigma-points for the CDKF is once again a set of points deterministically drawn from the prior statistics of *x*. Studies (Ito & Xiong, 2000) have shown that in practice, just as UKF, the CDKF generates estimates that are clearly superior to those calculated by an EKF.

3.3 Square-Root Forms of UKF and CDKF

SR-UKF and SR-CDKF (van der Merwe & Wan, 2001) are numerically efficient square-root forms derived from UKF and CDKF respectively. Instead of calculating the matrix square-root of the state covariance at each time step (a very costly operation) in order to buid the sigma-point set, these forms propagate and update the square-root of the state covariance directly in Cholesky factored form, using linear algebra techniques. This also provides more numerical stability.

The square-root SPKFs (SR-UKF and SR-CDKF) achieve equal or slightly higher accuracy when compared to the standard SPKFs. Besides, they have lower computational cost and a consistently increased numerical stability.

4. Experimental Evaluation

Since the experiments described in (Bi & Bennett, 2003) used just one data set and their results were only for REC demonstration, we first did tests with two well-known regression algorithms using 25 regression problems, in order to better evaluate the REC curves as a tool for visualizing and comparing regression learning algorithms.

Then we present the results of the comparison by using REC curves of SPKFs and EKF applied to time series and finally we investigate the use of an ensemble method (stacking (Wolpert, 1992)) with the tested models, evaluating it with REC curves, as suggested by Bi and Bennett (2003). In this work, 12 time series with real-world data were used in order to try to establish a general ranking among the models tested. The names and sizes of the used time series are shown in Table 1. All data are differentiated and then the values are rescaled linearly to between 0.1 and 0.9. As null model we choose the mean model, a constant function with the constant equal to the mean of the response of the training data.

Table 1. Time series used in the experimental evaluation.

Time series	Data points
A[1]	1000
Burstin[2]	2001
Darwin[2]	1400
Earthquake[2]	2097
Leuven[3]	2000
Mackey-Glass[4]	300
Series 1[5]	96
Series 2 [5]	96
Series 3 [5]	96
Soiltemp[2]	2306
Speech[2]	1020
Ts1 [2]	1000

4.1 Preliminary Results with Regression

Initial experiments were carried out in order to reinforce the conclusions reached out by Bi and Bennett (2003) in favor of the use of REC curves as a mean to compare regression algorithms (similarly to arguments for ROC curves in classification).

We have used REC curves in order to compare the performance of the Naive Bayes for Regression (Frank, Trigg, Holmes & Witten, 2000) to the performance of Model Trees (Quinlan, 1992). Naive Bayes for Regression (NBR) uses the Naive Bayes methodology for numeric prediction tasks by modeling the probability distribution of the target value with kernel density estimators. Model Tree predictor is a state-of-the-art method for regression. Model trees are the counterpart of

[1] Data from a competition sponsored by the Santa Fe Institute. (http://www-psych.stanford.edu/%7Eandreas/Time-Series/SantaFe)

[2] Data from the UCR Time Series Data Mining Archive (Keogh & Folias, 2002).

[3] Data from the K.U. Leuven competition. (ftp://ftp.esat.kuleuven.ac.be/pub/sista/suykens/workshop/datacomp.dat)

[4] Numerical solution for the Mackey-Glass delay-differential equation.

[5] Data of monthly electric load forecasting from Brazilian utilities (Teixeira & Zaverucha, 2003).

decision trees for regression tasks. They have the same structure as decision trees, but employ linear regression at each leaf node to make a prediction. In (Frank, Trigg, Holmes & Witten, 2000) an accuracy comparison of these two learning algorithms is presented and its results show that Model Trees outperform NBR significantly for almost all data sets tested.

The 25 regression data sets used in this study were obtained from the UCI Repository of Machine Learning Databases (Blake & Merz, 2006). With 16 of the data sets the Model Tree predictor clearly outperforms NBR, as can be seen, for instance, in Figure 2. The number between parentheses in the figure is the AOC value for each REC curve. Note that the REC curve for Model Tree covers completely the REC curve for NBR, becoming clear the superiority of the former algorithm when applied to this specific data set.

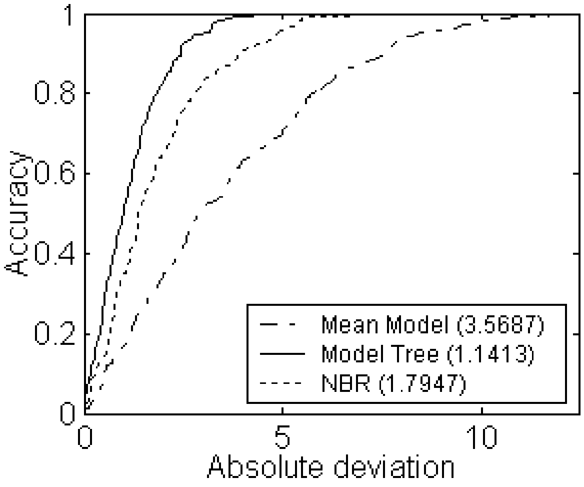

Figure 2. REC graph used to compare the performances of NBR and Model Tree when applied to data set pwLinear.

4.2 Comparing SPKFs by means of REC Curves

First, we have compared UKF and CDKF with their square-root forms, SR-UKF and SR-CDKF respectively. As expected, the REC curves for UKF and for SR-UKF are very similar. This means that the difference between the performances of the models provided by UKF and SR-UKF was negligible. The same fact could be verified with the REC curves for CDKF and SR-CDKF. Therefore, because of these results and the other advantages mentioned before in Section 3, we have continued our experiments only with the square-root forms of the SPKF.

By analyzing the generated REC graphs, we could verify that, for most time series, the model provided by SR-UKF dominates the models provided by SR-CDKF and EKF, that is, the REC curve for the SR-UKF model is always above the REC curves for SR-CDKF and EKF. Therefore, the model provided by SR-UKF would be preferable. An example is shown in Figure 3.

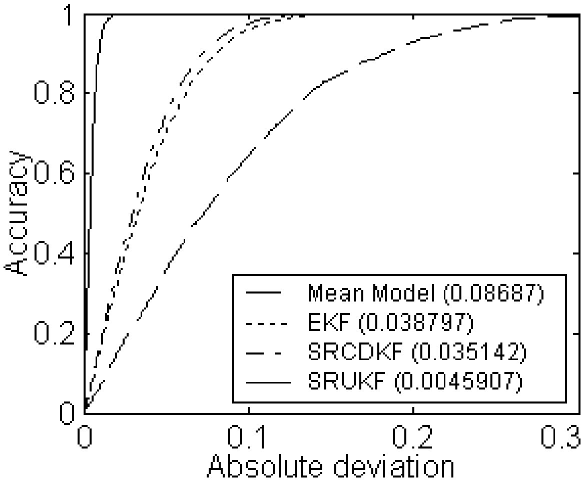

Figure 3. EKF and SPKFs applied to Burstin time series.

SR-UKF was outperformed by SR-CDKF only for the Mackey-Glass time series (Figure 4). Note that the curves cross each other at error tolerance of 0.7. SR-CDKF and EKF achieved similar performances for almost all time series, as can be seen, for instance, in Figure 5. However, the analysis of the AOC's gives a small advantage to SR-CDKF. The lower performance of EKF when compared to the others is probably caused by the non-linearity of the series. Therefore, SR-UKF consistently showed to be the best alternative to use with these series, followed by SR-CDKF and EKF, in this order. The Model Tree predictor and NBR were also tested for the prediction of the time series, but both provided poor models.

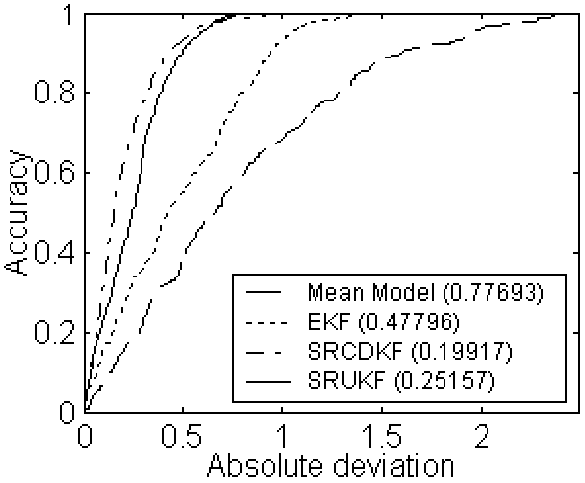

Figure 4. EKF and SPKFs applied to Mackey-Glass time series.

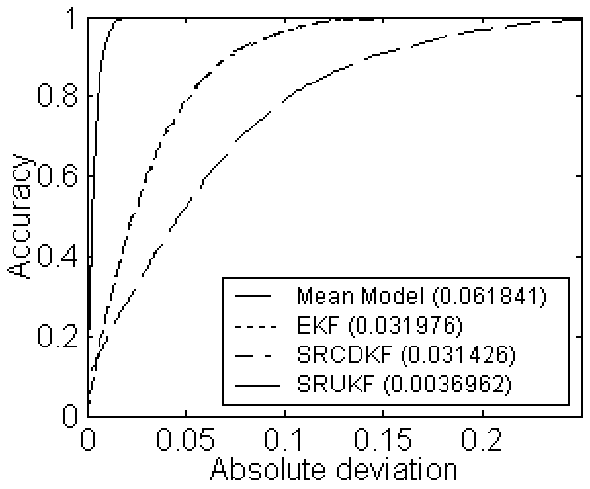

Figure 5. EKF and SPKFs applied to Earthquake time series.

4.3 Stacking of Sigma-Point Kalman Filters

Stacking (Wolpert, 1992) is an ensemble method (Dietterich, 1998) used to combine different learning algorithms. It works as follows. Suppose we have a set of different learning algorithms and a set of training examples. Each of these algorithms, called base learners, is applied to the training data in order to produce a set of hypotheses. The results computed by this set of hypotheses are combined into new instances, called meta-instances. Each "attribute" in the meta-instance is the output of one of the base learners and the class value is the same of the original instance. Another learning algorithm, called meta-regressor (or meta-classifier, for classification), is trained and tested with the meta-instances and provides the final result of the stacking.

We have used stacking to build ensembles of SPKFs and EKF. A Model Tree predictor was chosen as a meta-regressor not only because it achieved good results in the initial experiments, but also because it is a state-of-the-art regression method and it has already been successfully used as a meta-classifier for stacking (Dzeroski & Zenko, 2004), outperforming all the other combining methods tested.

Table 2. Stackings built.

Stackings	Base learners
Stacking 1	EKF, SR-CDKF
Stacking 2	EKF, SR-UKF
Stacking 3	SR-CDKF, SR-UKF
Stacking 4	EKF, SR-CDKF, SR-UKF

In order to determine which subset of algorithms can provide the best ensemble, we built four models by stacking: one containing the square-root SPKFs and EKF, and the others leaving one of them out. If we were testing several algorithms we could use a method to build the ensembles (Caruana & Niculescu-Mizil, 2004). Table 2 shows the stackings built: Stacking 1 is composed by EKF and SR-CDKF, Stacking 2 is composed by EKF and SR-UKF, Stacking 3 is composed by SR-CDKF and SR-UKF, and Stacking 4 is composed by EKF, SR-CDKF and SR-UKF. The REC curves show that all stackings that have the SR-UKF as a base learner achieve similar high performances. This can be seen, for example, in Figure 6.

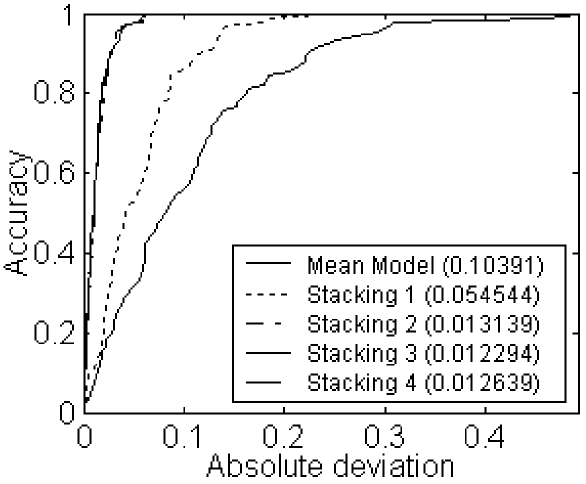

Figure 6. Stackings applied to Series 2 time series.

Table 3 shows the AOC values of the REC curves provided for the stackings with SR-UKF as a base learner. By analyzing the values we can see that among the three stackings that contain the SR-UKF, those who have SR-CDKF as a base learner achieve a slightly better performance. Since the number of time series for which Stacking 3 achieved the best performance is almost the same number of time series for which Stacking 4 was the best, we have considered that the inclusion of EKF as a base learner does not compensate the overhead in terms of computational cost. Thus, the model chosen as the best is that provided by Stacking 3 (SR-CDKF and SR-UKF as base learners).

Table 3. AOC's of the REC curves provided for the stackings with SR-UKF as a base learner.

Time series	Stacking 2	Stacking 3	Stacking 4
A	0.001366	0.001497	**0.001310**
Burstin	0.001740	**0.001613**	0.001740
Darwin	**0.013934**	0.014069	0.014052
Earthquake	0.000946	**0.000943**	0.000946
Leuven	0.005172	0.005190	**0.005142**
Mackey-Glass	0.228064	0.133420	**0.128672**
Series 1	0.001167	0.001306	**0.001111**
Series 2	0.013139	**0.012294**	0.012639
Series 3	0.000800	**0.000717**	0.000767
Soiltemp	0.000884	**0.000780**	0.000782
Speech	0.000714	0.000713	**0.000706**
Ts1	0.005010	0.005044	**0.004881**

By comparing the best stacking model (SR-CDKF and SR-UKF as base learners and Model Tree predictor as meta-regressor) to the best individual algorithm (SR-UKF) we could verify that the stacking achieved a significantly higher performance for all time series tested. This can be clearly noted in Figure 7.

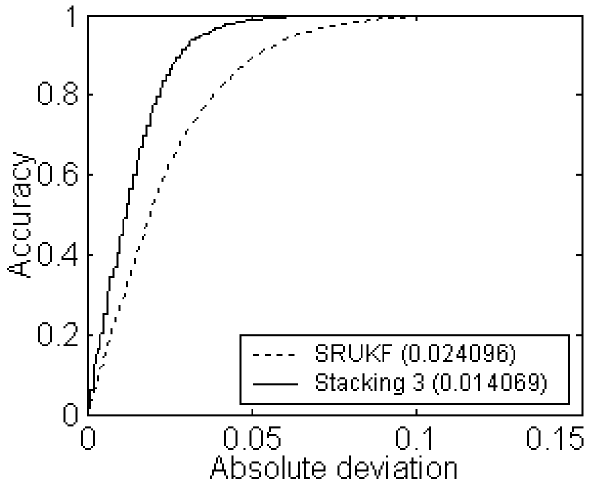

Figure 7. SR-UKF and Stacking 3 applied to Darwin time series.

5. Conclusions and Future Works

We have used REC curves in order to compare the SPKF family of filters (state-of-the-art time series predictors) and ensembles of them, applied to real-world time series.

The results of the experiments pointed SR-UKF as the best SPKF to use for forecasting with the series tested. Further experiments showed that a stacking composed by SR-CDKF and SR-UKF as base learners and a Model Tree predictor as meta-regressor can provide a performance statistically significantly better than that provided by the SR-UKF algorithm working individually. The REC curves showed to be very efficient in the comparison and choice of time series predictors and base learners for ensembles of them.

Currently, we are conducing tests with REC curves in order to compare Particle Filters (Doucet, de Freitas & Gordon, 2001), sequential Monte Carlo based methods that allows for a complete representation of the state distribution using sequential importance sampling and resampling. Since Particle Filters approximate the posterior without making any explicit assumption about its form, they can be used in general nonlinear, non-Gaussian systems. As a future work we intend to investigate further the use of ensembles with SPKFs, as well as with Particle Filters.

Acknowledgments

We would like to thank Eibe Frank for providing the source code for the NBR algorithm. The Sigma-Point Kalman Filters were obtained from the ReBEL Toolkit, under Academic License from OGI School of Science and Engineering, Rudolph van der Merwe and Eric A. Wan. The implementation of Model Trees was obtained from the WEKA System (http://www.cs.waikato.ac.nz/~ml/weka/). The authors are partially financially supported by the Brazilian Research Agency CNPq.

References

Bi, J., & Bennett, K. P. (2003). Regression Error Characteristic Curves. *Proceedings of the 20th International Conference on Machine Learning (ICML 2003)* (pp. 43-50). Washington, DC.

Blake, C. L., & Merz, C. J. (2006). UCI Repository of Machine Learning Databases. Machine-readable data repository. University of California, Department of Information and Computer Science, Irvine, CA. [http://www.ics.uci.edu/~mlearn/MLRepository.html]

Caruana, R., & Niculescu-Mizil, A. (1997). An Empirical Evaluation of Supervised Learning for ROC Area. *Proceedings of the First Workshop on ROC Analysis (ROCAI 2004)* (pp. 1-8).

Dietterich, T. G. (1998). Machine Learning Research: Four Current Directions. *The AI Magazine*, *18*, 97-136.

Doucet, A., de Freitas, N., & Gordon, N. (2001). *Sequential Monte-Carlo Methods in Practice*. Springer-Verlag.

Dzeroski, S., & Zenko, B. (2004). Is Combining Classifiers with Stacking Better than Selecting the Best One?. *Machine Learning*, *54*, 255-273.

Frank, E., Trigg, L., Holmes, G., & Witten, I. H. (2000). Naive Bayes for Regression. *Machine Learning*, *41*, 5-25.

Ito, K., & Xiong, K. (2000). Gaussian Filters for Nonlinear Filtering Problems. *IEEE Transactions on Automatic Control*, *45*, 910-927.

van der Merwe, R., & Wan, E. (2001). Efficient Derivative-Free Kalman Filters for Online Learning. *Proceedings of the 9th European Symposium on Artificial Neural Networks (ESANN'2001)*. Bruges.

van der Merwe, R., & Wan, E. (2003). Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. *Proceedings of the Workshop on Advances in Machine Learning*. Montreal, Canada.

Jazwinsky, A. (1970). *Stochastic Processes and Filtering Theory*. New York: Academic Press.

Julier, S., Uhlmann, J., & Durrant-Whyte, H. (1995). A New Approach for Filtering Nonlinear Systems. *Proceedings of the American Control Conference* (pp. 1628-1632).

Keogh, E., & Folias, T. (2002). The UCR Time Series Data Mining Archive. University of California, Computer Science & Engineering Department, Riverside, CA.

Provost, F., & Fawcett, T. (1997). Analysis and Visualization of Classifier Performance: Comparison Under Imprecise Class and Cost Distributions. *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD'97)* (pp. 43-48). AAAI Press.

Provost, F., Fawcett, T., & Kohavi, R. (1998). The Case Against Accuracy Estimation for Comparing Classifiers. *Proceedings of the 15th International Conference on Machine Learning (ICML 1998)* (pp. 445-453). San Francisco: Morgan Kaufmann.

Quinlan, J.R. (1992). Learning with Continuous Classes. *Proceedings of the 5th Australian Joint Conference on Artificial Intelligence* (pp. 343-348). Singapore: World Scientific.

Teixeira, M., & Zaverucha, G. (2003). Fuzzy Bayes and Fuzzy Markov Predictors. *Journal of Intelligent and Fuzzy Systems*, *13*, 155-165.

Wang, Y., & Witten, I. H. (1997). Induction of Model Trees for Predicting Continuous Classes. *Proceedings of the poster papers of the European Conference on Machine Learning.* University of Economics, Faculty of Informatics and Statistics, Prague.

Wolpert, D. (1992). Stacked generalization. *Neural Networks*, *5*, 241-260.