Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

Local Learning of Tide Level Time Series using a Fuzzy Approach

E. Canestrelli, P. Canestrelli, M. Corazza, M. Filippone, S. Giove, F. Masulli

Abstract— **Forecasting the tide level in the Venezia lagoon is a very compelling task. In this work we propose a new approach to the learning of tide level time series based on the local learning procedure of Bottou and Vapnik, by considering the use of a fuzzy method for the selection of the closest patterns to the one to forecast. We made use also as learners of Support Vector Machines and of their ensembles based on Bagging and AdaBoost. The obtained forecasts of 500 randomly selected tide levels seem to be quite promising. Good performances are also noticed for forecasts of a set of 80 tide levels corresponding to exceptional periods with high tide and sea variabilities. The obtained forecasts of 80 selected tide levels compare very favorably with those of the baseline linear regressor model.**

I. Introduction

After the disastrous flood of November 1966 in Venezia (194 cm over the average sea level) the Municipality of Venezia set up the first observation service of the high tides. In December 1979 another severe flood occurred (166 cm over the average sea level). In consequence of these events, the municipal government decided to found the CPSM – *Centro Previsioni e Segnalazioni Maree* (Tide Forecasting and Signalling Center) – which mainly has to supply an effective alarm service regarding the occurrence of important or extraordinary high (and ebb) tides [5].

Forecasting the tide level is the most compelling task that the CPSM has to deal with. Mainly, this task is faced by means of statistical modelling. Currently, the CPSM has some multivariate regression models at its disposal, each of which is variously based on features regarding the tide level gauged at Venezia and the atmospheric pressure value gauged by more survey stations of the Adriatic sea and the Tyrrhenian one.

In this work we propose a new approach based on the the general local learning procedure by Bottou and Vapnik [1], by considering the use of a fuzzy method for the selection of closest patterns from the data set and of Support Vector Machines and of their ensembles based on Bagging and AdaBoost. The obtained results are compared with a baseline model consisting in the so–called MC – *Modello Completo* (Complete Model) – that is one of the well–performing models used by CPSM.

The rest of this paper is organized as follows. Section II shows the fuzzy approach to pattern selection. Section III describes the MC linear regressor baseline forecaster, while Section IV presents the base learners and their ensembles we have used in our approach. In Section V we present the data set and the methods we have used in the experimental validation, the results we have obtained and their discussion. Section VI draws the conclusions of the paper.

II. Fuzzy approach to pattern selection

As discussed in [1], the performances of learning algorithms (e.g., neural networks, Support Vector Machines - SVMs) can be satisfactorily increased if the set of learning data is properly selected. In particular, the concept of local learning is strongly advised and suggested. The local learning approach proposed by Bottou and Vapnik is based in the following procedure [1]:

For each test pattern:

1) Select the k closest patterns from the data set;
2) Train a learning machine using the above selected patterns (local learning);
3) Apply the above trained learning machine to predict the test pattern.

Such approach has been demonstrated to be very efficient for learning in non–homogeneous domains, such as the case of our data set. The main idea consists, given the actual pattern, in the selection of a suitable subset of all the data set. The dimension of such subset needs to be not so large in order to reduce as most as possible the learning time at each step, and at the same time, sufficiently representative for the capacity of the learning system. In such a way, the trade–off between capacity and number of patterns can be optimized. In this case study, we focus the attention to the forecasting problem of the tide over a time window of (at most) 48 hours, using a pattern of 75 features:

- 30 1–hour frequency tide levels in Venezia (from current time T to time $T-29$);
- 5 3–hour frequency atmospheric pressure values for each of the survey stations in Alghero, Bari, Genova and Venezia (from current time T to time $T-12$);
- 5 3–hour frequency quantities regarding the squares of the difference of the atmospheric pressure values, multiplied by the sign of the difference itself (signed squared gradient) for each of the following couple of survey stations: Dubrovnik and Bari; Pola and Rimini; Spalato and Termoli; Trieste and Ravenna; and Zara and

E. Canestrelli is with: the Department of Applied Mathematics, University Ca' Foscari of Venezia, Dorsoduro 3825/E, 30123 Venezia, Italy; the Consorzio Venezia Ricerche, Via della Libertà, 12 - 30175 Marghera (VE), Italy (email: canestre@unive.it). P. Canestrelli is with the Centro Previsioni e Segnalazioni Maree, Comune di Venezia, San Marco 4090, 30124 Venezia, Italy (email: paolo.canestrelli@comune.venezia.it). M. Corazza is with: the Department of Applied Mathematics, University Ca' Foscari of Venezia, Dorsoduro 3825/E, 30123 Venezia, Italy; the School for Advanced Studies in Venezia Foundation, Dorsoduro 3488/U - 30123 Venezia, Italy (email: corazza@unive.it). S. Giove is with the Department of Applied Mathematics, University Ca' Foscari of Venezia, Dorsoduro 3825/E, 30123 Venezia, Italy (email: sgiove@unive.it). M. Filippone and F. Masulli are with: the CNISM, Via della Vasca Navale 84, 00146 Roma, Italy; the Consorzio Venezia Ricerche, Via della Libertà, 12, 30175 Marghera (VE), Italy; the Department of Computer and Information Sciences, University of Genova, Via Dodecaneso 35, 16146 Genova, Italy (email: {filippone|masulli}@disi.unige.it).

1-4244-1380-X/07/$25.00 ©2007 IEEE

mean of Falconara and Pescara (from current time T to time $T-12$).

The locality component is here represented by a suitable triangular fuzzy membership function, with one parameter (the amplitude) dynamically adjusted in such a way that the neighborhood includes a significant number of similar sampled patterns. To avoid confusion, we name as current pattern the 75 component vector formed by the last sampled data at time t, and as a past pattern each of the sampled data formed by (consecutive) observed values sampled backward starting from any time τ before t, while we name as a *local pattern* each pattern belonging to the neighborhood of the current pattern. A local pattern is a past pattern selected after the search in the data base. Any (past) pattern is uniquely determined by a time value τ, with $\tau < t$, formed by the consecutive values of tide level, atmospheric pressure value and signed squared gradient starting from τ and collected backward (30 values for level tide, 20 for atmospheric pressure and 25 for signed squared gradient).

As we said above, for the selection of a proper subset of similar patterns, a suitable kernel function has to be selected [3]. In our proposal, we choose a triangular fuzzy membership function whose amplitude is fixed by default to the following values: 30 cm for the level tide, 20 hPa for the atmospheric pressure, 25 hPa^2 for the signed squared gradient. Those default values can be increased at each interaction in such a way to include at least a pre-fixed number of patterns in the local neighborhood. From numerical tests this scarcity situation of neighborhood patterns is quite rare (about 5 to 10 cases over 100). During this research, the current pattern is matched with all the past ones, comparing each component together. A necessary condition to be included in the local neighborhood requires that each component differs no more than the amplitude of the membership function. In so doing, the membership function filters the difference between all the 75 components, and only if all the filtered values are positive, they are aggregated to furnish a degree of similarity between the current pattern and the considered past pattern. Conversely, if the absolute difference between a component of the current pattern and the same component of the comparing (past) pattern overpasses the amplitude of the membership function, the pattern is discharged, avoiding the computation of the difference of the remaining components. This way, we observed a significant time saving with respect to other local algorithms based on distance operator for the selection of the neighborhood. If the considered past pattern passes the test for all the 75 components, it is formally included in the local neighborhood. To this purpose it is sufficient to recover the time value τ. The aggregation algorithm is a simple weighted averaging of all the filtered values component by component. After having compared the current pattern with all the past ones, the algorithm checks if the selected local patterns are less than the fixed minimum number (a changeable parameter). If the number is insufficient, the amplitude of the membership function is increased using a scale factor, until the obtained number is satisfactory. At the end, the following values are stored: $\tau(1)$, $\tau(2)$, ..., $\tau(N_t)$, $Sim(1)$, $Sim(2)$, ..., $Sim(N_t)$, where N_t indicates the number of the patterns in the local neighborhood for the current pattern starting (backward) at time t, $\tau(j)$ means the starting (backward) time of the j-th similar pattern, and $Sim(j)$ indicates its similarity degree with the current pattern, with $j = 1, \dots, N_t$. The patterns are then ordered following the decreasing order of similarity. Even if the procedure needs to be run at each current time t, all the numerical tests on the real data showed good performances mainly for the computational required time.

III. Baseline Forecaster

In this section we shall describe MC – *Modello Completo* (Complete Model) – that is a linear regressor model used by CPSM. In this paper we use MC as a baseline for our benchmarks.

In detail, this model utilizes 75 features described in previous section, and can be formalized as follows:

$$dh(T+h) = \sum_{i=0}^{29} a_i dh(T-i) + \sum_{k=1}^{4}\sum_{i=0}^{4} b_{k,i} p_k(T-3i) + \\ + \sum_{l=1}^{5}\sum_{i=0}^{4} c_{l,i} \mathrm{sign}\left(\Delta_k(T-3i)\right) \Delta_k^2(T-3i),$$

where $dh(t)$ denotes the meteorological contribution[1] to the tide level recorded by the survey station in Venezia at time t, T indicates the current time, h denotes the horizon time of the forecast, $p_k(t)$ indicates the atmospheric pressure recorded by the k–th survey station at time t, $\Delta_k(t)$ means the difference of atmospheric pressure recorded by the l–th couple of survey stations at time t, and a_i, $b_{k,i}$, $c_{l,i}$ indicate the coefficients.

IV. Base Learners and Ensembles

As base learners, we have used linear SVM [7] for regression implemented in R [10] through the function *svm* of package *e1071*. The implementation is the porting of Chang and Lin code [4], [6].

We used also ensemble methods [12] aggregating the output of a set of base learners and can increase generalization on the same data set, as they can boost margins, reduce variance, and also bias. The ensemble methods we have considered are the Bagging [2] and the Adaboost [9] algorithms that are based on data set re-sampling. We have implemented in R those ensemble techniques. The implementation of Adaboost for regression follows [8].[2]

[1] It is worth noting that the tide level is given by the summation of the astronomical contribution (easy to forecast) and the meteorological contribution (difficult to forecast).

[2] The package [11] is available at `http://mlsc.disi.unige.it/R`.

Horizon (h)	sdE (cm)	ME (cm)	$\max E$ (cm)	$\min E$ (cm)	MAE (cm)	Model
1	2.02	-0.09	8.63	-6.25	1.61	linear SVM $c = 1,\ k = 500$
2	3.47	0.09	11.32	-11.64	2.69	linear SVM $c = 1,\ k = 600$
3	4.40	0.11	16.58	-17.41	3.30	linear SVM $c = 1,\ k = 600$
6	5.44	0.48	15.93	-31.98	3.97	linear SVM $c = 1,\ k = 600$
12	6.29	0.10	23.07	-31.96	4.63	linear SVM $c = 1,\ k = 500$
24	8.44	-0.05	49.07	-38.07	5.93	linear SVM $c = 1,\ k = 400$
48	11.37	-0.43	34.81	-52.82	8.17	linear SVM $c = 1,\ k = 500$

TABLE I

Results obtained with SVM tested on the 500 patterns of *VEN-TIDES* problem.

Horizon (h)	sdE (cm)	ME (cm)	$\max E$ (cm)	$\min E$ (cm)	MAE (cm)
1	3.47	-1.09	8.55	-10.26	2.79
2	7.45	-2.98	17.98	-24.04	5.72
3	10.59	-2.91	15.09	-32.90	8.30
6	15.35	-6.71	22.70	-54.64	12.01
12	20.46	-11.54	26.51	-68.66	16.94
24	28.44	-17.11	40.00	-89.65	24.55

TABLE II

Results obtained with MC tested on the 80 patterns of *VEN-TIDES** problem.

Horizon (h)	sdE (cm)	ME (cm)	$\max E$ (cm)	$\min E$ (cm)	MAE (cm)	Model
1	3.48	-0.48	10.29	-7.82	2.78	linear SVM $c = 1,\ k = 1000$
2	5.54	-0.35	18.63	-17.17	4.19	linear SVM $c = 1,\ k = 900$
3	6.43	-0.46	17.18	-22.59	4.61	linear SVM $c = 1,\ k = 800$
6	10.91	-1.90	25.18	-29.41	8.17	linear SVM $c = 1,\ k = 1100$
12	10.07	-1.21	31.40	-29.13	7.51	linear SVM $c = 1,\ k = 1000$
24	11.68	2.04	29.72	-31.80	8.76	linear SVM $c = 1,\ k = 1400$

TABLE III

Results obtained with SVM tested on the 80 patterns of *VEN-TIDES** problem.

V. Experimental validation

A. Data and methods

The time series which are used concern the period from January 1, 1966 to December 31, 1990, and the number of valid patterns of the considered features is 153819 with 75 dimensions.

From the entire data set we considered two forecasting problems, namely *VEN-TIDES* and *VEN-TIDES**. In the first one we considered the forecast of 500 randomly selected tide levels (test set). As the training set for *VEN-TIDES* we used the set of 51266 3–hours patterns. Then we concentrated on the more challenging problem *VEN-TIDES** of forecasting 80 tide levels corresponding to exceptional periods with high tide and sea variabilities (test set). As training set for *VEN-TIDES** we used all the 154819 1–hour patterns in which the pressure value for the first hour is used as well as for the second and third hours.

We decided to implement forecasters with the temporal horizons 1, 2, 3, 6, 12, 24, and 48 hours for the first forecasting problems, and with the temporal horizons 1, 2, 3, 6, 12, and 24 hours for the second forecasting problems.

The forecaster are based on SVM for regression with cost $c = 1$ and insensivity epsilon-tube with $\varepsilon = 0.1$ and on their Bagging and AdaBoost ensembles.

In order to implement the proposed local learning approach, we used as proximity relation the fuzzy distance illustrated in Section II, using the following thresholds: 20

Horizon (h)	sdE (cm)	ME (cm)	$\max E$ (cm)	$\min E$ (cm)	MAE (cm)	Model
1	4.30	-0.01	13.50	-8.21	3.27	Adaboost $n = 40,\ k = 300$
1	3.77	-0.45	9.46	-7.75	3.05	Bagging $n = 20,\ k = 400$
1	4.06	-0.32	9.40	-8.82	3.25	Adaboost $n = 100,\ k = 400$
1	4.10	-0.31	9.83	-10.36	3.28	Adaboost $n = 20,\ k = 600$
2	7.01	-0.46	20.42	-19.89	5.44	Adaboost $n = 20,\ k = 400$
2	5.94	-0.66	15.43	-17.75	4.69	Bagging $n = 20,\ k = 400$
2	5.22	-0.42	15.34	-17.54	3.99	Bagging $n = 20,\ k = 600$
3	6.66	-1.06	23.06	-25.16	4.59	Adaboost $n = 20,\ k = 800$
3	6.34	-0.23	18.06	-22.07	4.49	Bagging $n = 20,\ k = 800$
6	10.51	-2.32	17.01	-38.94	7.47	Bagging $n = 20,\ k = 500$
6	10.31	-2.16	19.34	-36.90	7.40	Bagging $n = 40,\ k = 500$

TABLE IV

RESULTS OBTAINED WITH ENSEMBLES TESTED ON THE 80 PATTERNS OF *VEN-TIDES** PROBLEM.

cm for tides, 2000 Pa for pressures, 30 hPa^2 for the signed squared gradients.

Note that the for a successful implementation of the local learning we must find the optimal value of k, i.e., we must find a value of k trading off between generalization and learning speed (cardinality selection procedure).

It is worth noting that the local learning allows us to implicitly implement a model selection approach of *Leave One-Out* type, both for the training of the learning machine, and for its validation on the available data set.

For the cardinality selection task we used the following performance indexes:

- Standard Deviation of the Error:

$$sdE = \sqrt{\frac{\sum_{i=1}^{n} E_i^2}{n}}, \tag{1}$$

- Mean Error:

$$ME = \frac{\sum_{i=1}^{n} E_i}{n}, \tag{2}$$

- Minimum Error:

$$\min E = \min_{i=1,\dots,n} \{E_i\}, \tag{3}$$

- Maximum Error:

$$\max E = \max_{i=1,\dots,n} \{E_i\}, \tag{4}$$

- Mean Absolute Error:

$$MAE = \frac{\sum_{i=1}^{n} |E_i|}{n}, \tag{5}$$

where E_i is the difference between forecasted and true tide level on the i–th pattern and n is the cardinality of the test set.

B. Results and discussion

The results with the proposed local learning approach and linear SVMs for problem *VEN-TIDES* are reported in Table I, and Fig. 1(a) and (b) show the scatter plots and the linear regression lines for temporal horizons of 2 and 12 hours. These performances seem to be quite promising.

In Table II we report the results obtained on the selected temporal horizons by the baseline forecaster MC in the *VEN-TIDES** problem.

The results obtained on the same problem, using the proposed local learning approach with fuzzy selection of patterns and SVM are reported in Table III. For each different temporal horizons and for each one we performed cardinality selection by testing different values of k (see Fig. 2). The results are good and compare very favorably with those of MC. Fig. 1(c) shows the scatter plot and the linear regression line for a temporal horizon of 2 hours.

In Table IV some preliminary results of SVM ensembles using Bagging and Adaboost are presented (the parameter n represents the number of SVM base learner used in each ensemble). No significant improvements with respect the single SVM are noticeable. Sometimes the Bagging algorithm obtains a greater value of sdE than Adaboost, but gives a slighter better value for $\min E$, $\max E$, and ME. Fig. 1(d) shows the scatter plot and the linear regression line for a temporal horizon of 6 hours.

The software has been developed in R language [10], and special attention has been payed to its optimization. On a Pentium IV at 1.9 Ghz, the forecasting of an event using the local learning approach and a SVM base learner costs less that ten seconds, even when we select about 1000 nearest neighbors, and about ten seconds more for the compilation of the training set. As a consequence the actual implementation of the systems, allows its utilization on-line for tide forecasting, even using ensembles of SVM.

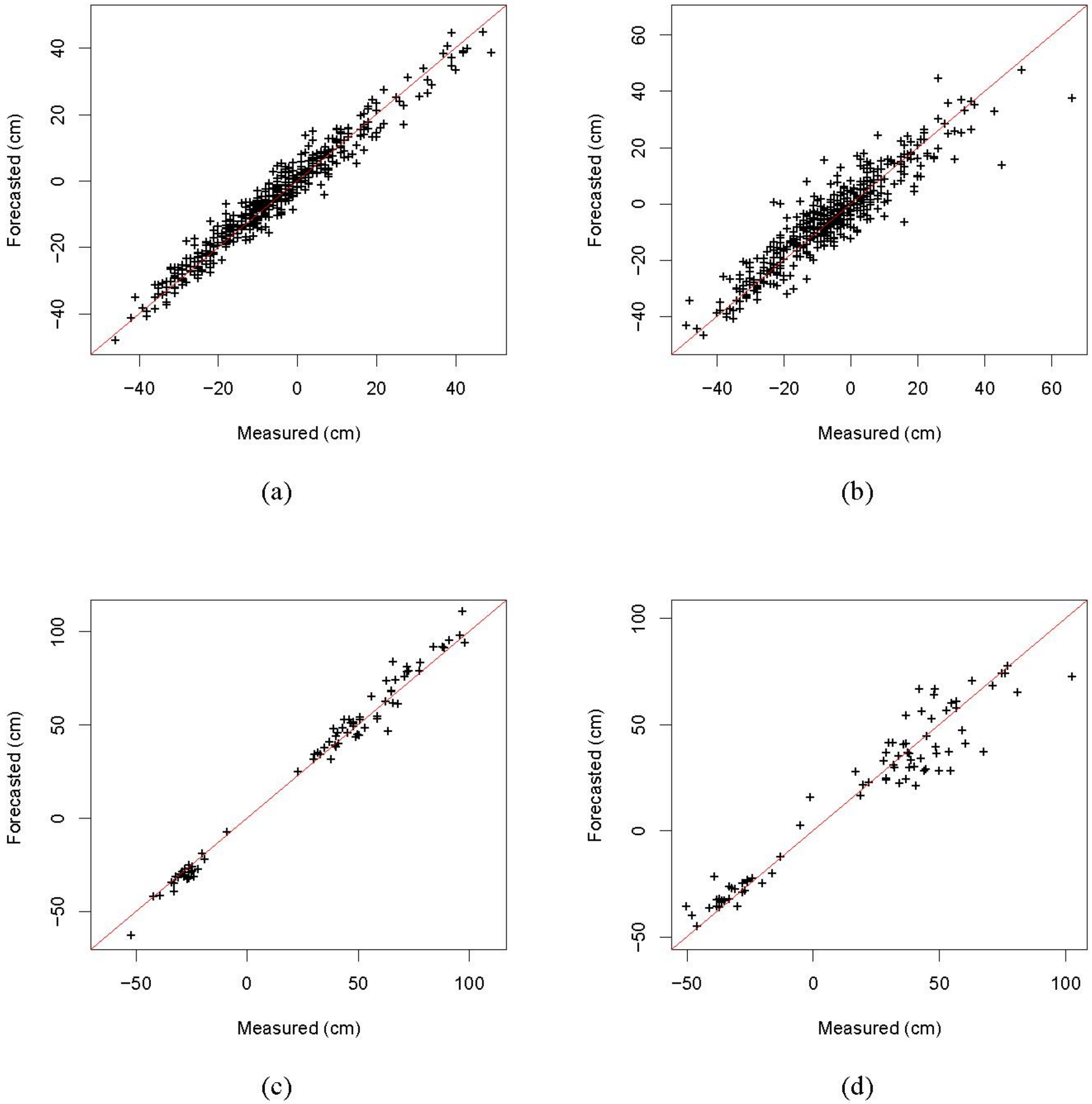

Fig. 1. (a) Scatter plot for linear SVM on the 500 patterns of *VEN-TIDES* problem – Horizon: 2 h ($c = 1$, $\varepsilon = 0.1$, $k = 600$). (b) Scatter plot for linear SVM on the 500 patterns of *VEN-TIDES* problem – Horizon: 12 h ($c = 1$, $\varepsilon = 0.1$, $k = 500$). (c) Scatter plot for linear SVM on the 80 patterns of *VEN-TIDES** problem – Horizon: 2 h ($c = 1$, $\varepsilon = 0.1$, $k = 800$). (d) Scatter plot for bagged ensemble on the the data base *VEN-TIDES** – Horizon: 6 h ($n = 20$, $k = 500$).

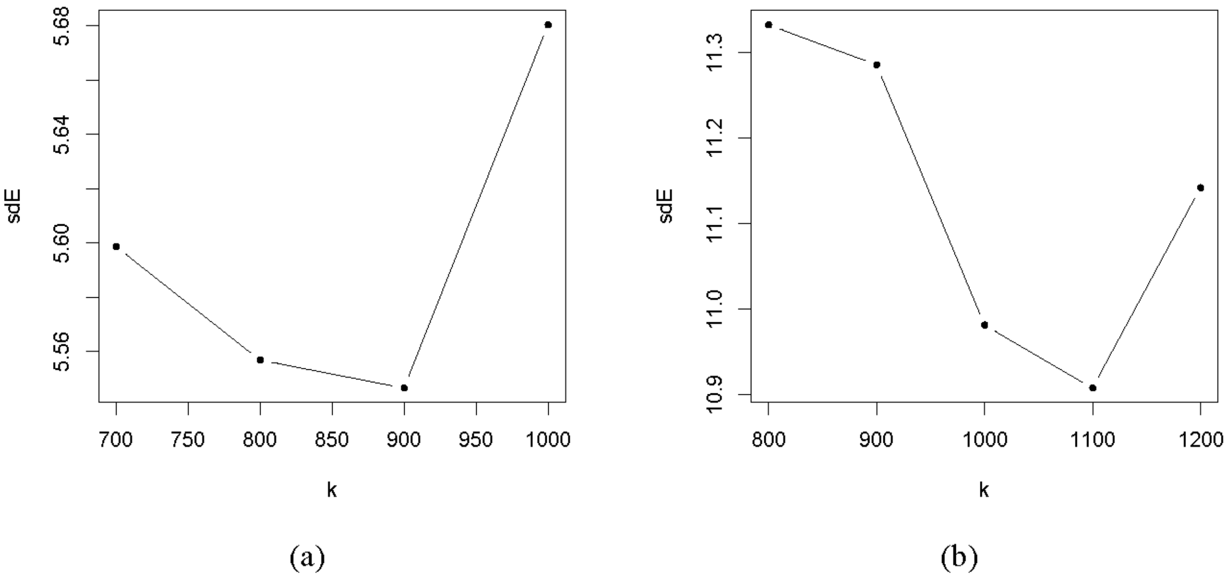

Fig. 2. Plot of the *sdE* obtained for the *cardinality selection* procedure in the *VEN-TIDES* problem: (a) 2 hours horizon; (b) 6 hours horizon

VI. Conclusions

Forecasting the tide level is the most compelling task that the CPSM – *Centro Previsioni e Segnalazioni Maree* (Tide Forecasting and Signalling Center) – of Venezia has to deal with.

In this work we propose a new approach to the learning of tide level time series based on the the general local learning procedure of Bottou and Vapnik [1], by considering the use of a fuzzy method for the selection of closest patterns from the data set where the locality component is represented by a suitable triangular fuzzy membership function with one parameter (the amplitude) dynamically adjusted in such a way that the neighborhood includes a significant number of similar sampled patterns. We made use also as learners of Support Vector Machines and of their ensembles based on Bagging and AdaBoost.

We performed an extensive experimental assessment using a time series on the period from January 1, 1966 to December 31, 1990, from which we obtained a data set of 153819 valid patterns 75 dimensions. The obtained forecasts of 500 randomly selected tide levels seem to be quite promising. Good performances are also noticed for forecasts of a set of 80 tide levels corresponding to exceptional periods with high tide and sea variabilities. The obtained forecasts of 80 selected tide levels compare very favorably with those of the baseline linear regressor model MC.

Acknowledgment

This paper reports some of the results obtained from May 10, 2005 to November 09, 2006 during the development of the research *Previsione di marea e diffusione alla popolazione veneziana* (Tide level forecasting and signalling to the Venetian population) financially supported by the Italian *Ministero dell'Istruzione, dell'Università e della Ricerca* (Ministry of Education, University and Research), and by the *Consorzio Venezia Ricerche* (Consortium Venezia Researches) of Venezia.

References

[1] L. Bottou and V. Vapnik. Local learning algorithms, *Neural Computation*, 4(6):888–900, 1992.

[2] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[3] J. Fan and I. Gijbels. *Local polynomial modelling and its applications*. Chapman and Hall, 1996.

[4] C. C. Chang and C J. Lin. *Training ν-support vector classifiers: Theory and algorithms*. Neural Computation, 13(9):2119–2147, 2001.

[5] P. Canestrelli and L. Zampato, "Sea–level Forecasting at the Centro Previsioni e Segnalazioni Maree (CPSM) of the Venice Municipality", in A. Fletcher, T. Spencer (Eds.), *Flooding and Environmental Challenges for Venice and its Lagoon: State of Knowledge*, Cambridge University Press, Cambridge (Great Britain), pp.85–98, 2005.

[6] C. C. Chang and C J. Lin. *Training ν-support vector regression: Theory and algorithms*. Neural Computation, 14(8):1959–1977, 2002.

[7] C. Cortes and V. Vapnik. Support Vector Networks. *Machine Learning*, 20(2):273–297, 1995.

[8] H. Drucker. Improving regressors using boosting techniques. In Douglas H. Fisher, editor, *ICML*, pages 107–115. Morgan Kaufmann, 1997.

[9] Y. Freund and R.E. Schapire. *Experiments with a New Boosting Algorithm.*. Proceedings of the Thirteenth Conference, ed: L. Saitta, Morgan Kaufmann, pp. 148–156, 1996.

[10] R. Ihaka and R. Gentleman. *R: A language for data analysis and graphics*. Journal of Computational and Graphical Statistics, 5(3):299–314, 1996.

[11] M. Filippone, F. Masulli and S. Rovetta, "ERAF: A R Language Package for Regression and Forecasting", in B. Apolloni, M. Marinaro, R. Tagliaferri (Eds.), *Biological and Artificial Intelligence Environments*, Springer–Verlag, Heidelberg (Germany), pp.165–173, 2005.

[12] G. Valentini and F. Masulli. Ensembles of Learning Machines, in M. Marinaro and R. Tagliaferri (Eds.), *Neural Nets WIRN Vietri–02, Series Lecture Notes in Computer Sciences*, Springer–Verlag, Heidelberg (Germany), vol. 2486, pp. 3–19, 2002.