Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/070404

Forecasting meteorological time series using soft computing methods: an empirical study

Elena Bautu[*] *and Alina Barbulescu*

Department of Mathematics and Computer Science, Ovidius University, Constanta, Romania

Received: 29 Nov. 2012, Revised: 19 Jan. 2013, Accepted: 11 Feb. 2013
Published online: 1 Jul. 2013

Abstract: The interest of researchers in different fields of science towards modern soft computing data driven methods for time series forecasting has grown in recent years. Modeling and forecasting hydrometeorological variables is an important step in understanding climate change. The application of modern methods instead of traditional statistical techniques has lead to great improvement in past studies on meteorological time series. In this paper, we employ Support Vector Regression (SVR) and automatic model induction by means of Adaptive Gene Expression Programming (AdaGEP) for modeling and short term forecasting of real world hydrometeorological time series. The investigated time series datasets cover annual, respectively monthly data, on temperature and precipitation, measured at several meteorological stations in the Black Sea region. Two performance measures were used to assess the efficiency of the models obtained for forecasting, alongside statistical testing of the goodness of fit via the Kolmogorov-Smirnov test. Based on the results of rigourous experiments, we conclude that the models obtained by the AdaGEP algorithm are more competent in forecasting the time series considered in this paper than the models produced with the SVR algorithm.

Keywords: Time series forecasting, Adaptive Gene Expression Programming, Support Vector Regression.

1 Introduction

Real world processes are very hard to summarize and predict. Hydrometeorological series are not an exception, since they are influenced by a diversity of phenomena and factors from the environment. The processes that drive their behavior are almost never very well described using a single mathematical equation.

Traditional methods for modeling time series come from the statistics literature and address the issue of deriving linear models. Although the resulting models are easy to interpret, these modeling methods impose strong limitations, such as the stationarity of the time series, the independence and normality of the residuals (see, for example [1]). Also, they lack the ability to detect non-linear traits in data.

Non-linear modeling techniques make little assumptions on the data distribution; a review of state of the art is provided by [2]. Among them, modern heuristic approaches based on Artificial Neural Networks [3] or Evolutionary Computation (EC) techniques, such as Genetic Programming (GP) [4,5,6], have been shown to obtain very good results in modeling geophysical series. Gene Expression Programming (GEP) is a technique based on classical GP, that uses a simpler, yet more powerful, representation. In our study, we used the improved adaptive GEP algorithm (AdaGEP), which is a hybrid between GEP and a classical bit-string Genetic Algorithm, rendering a powerful modeling tool. AdaGEP was used previously to model the dynamic behaviour of a process that generates a time series with very good results [7].

Support Vector Regression (SVR) is a nonlinear regression method based on Support Vector Machines (SVM). SVMs are very succesfull in the field of data minig for solving classification problems. The SVM algorithm is built based on the *structural* risk minimization principle, which means that it tries to minimize an upper bound of the generalization error. This is an important advantage over most neural networks, which implement the *empirical* risk minimization principle, thus minimizing the misclasification rate on the training data. SVR is a technical adaptation of the SVM algorithm built for tackling regression problems. Although the use of SVR is less spread, succesfull applications in various domains can be found in the literature [8].

[*] Corresponding author e-mail: ebautu@univ-ovidius.ro

This paper presents a comparative empirical study on hydrometeorological series modeling and forecasting with AdaGEP and SVR. The series used in this study represent annual, respectively monthly, temperature and precipitation data, gathered at several meteorological stations in the Black Sea region. There exist four seasons in this region, which endows the datasets used in this paper with challenging characteristics from a modeling point of view. The significant seasonal changes in both temperature and precipitation lead to an increased difficulty of their modeling and prediction using these data-driven methods.

We perform a systematic evaluation of the performances of the investigated methods, employing extensive datasets. The obtained models are validated against unseen test data and their predictions are compared to assess their generalization power. The paper does not propose a novel modeling method. We emphasize on the methodology used for GEP and SVR for forecasting and offer hints about which is the most appropriate choice of method based on the series characteristics.

The paper is structured as follows. The time series prediction problem is presented in section 2, stating the basic principles of state space reconstruction. A brief presentation of the GEP algorithm is contained in section 3, with an emphasis on the particular adaptive variant of GEP used in this paper in subsection 3.1. A brief account of SVR is presented next (section 4), in order to make this paper as self-contained as possible. Section 5 presents the methodology for the experimental study. A detailed discussion of the results is presented in section 6. The conclusion section ends the paper.

2 Time series prediction

We approach time series prediction in a symbolic regression fashion. The task is to identify the mathematical formulas which best describe the underlying mechanisms that produced the data.

Consider $\{x(t)\}$ a time series that was generated by a dynamical system, e.g. recording the temporal variation of temperature [16]. In practice, we deal with a sample of data from the time series, as a dataset containg an ordered set of observations (i.e. real valued numbers) of a variable $\{x_1, x_2, x_3, \ldots, x_n\}$, where n is the size of the dataset. The problem is to find a model that approximates the observed values of the variable as well as possible.

We take on the state space approach of embedding the series into a low dimensional Euclidean space [16]. A state vector is represented as

$$X_t = (x_t, x_{(t-\tau)}, \ldots, x_{(t-(d-1)\tau)}), \tag{1}$$

where τ is the time delay and d is the embedding dimension. Takens [19] proved that if the embedding dimension is big enough, then there exists an equation of the form

$$x_{(t+p)} = f^*(X_t),$$

where f^* is a function that predicts future values of the series $\{x_T\}$ using past values and p is the prediction horizon. We use AdaGEP and SVR to find the appropriate function that uses as input a time lagged vector of values from the series.

3 Gene Expression Programming

Evolutionary Computation (EC) techniques are governed by the principle of natural selection: the best adapted individual has the most chances to survive and reproduce in the next generations. A population of candidate models are randomly initialized and evolved, through repeated loops of recombination, mutation, reproduction, until a termination condition is fulfiled.

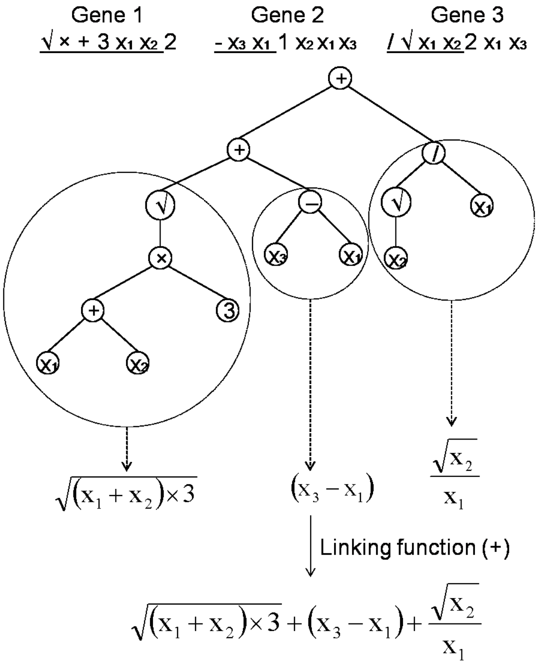
Gene Expression Programming (GEP) is an automatic model induction technique that pertains to the field of EC; it was proposed by Ferreira in [9], as an improvement of the standard Genetic Programming technique. The individuals represent complex models expressed as hierarchical mathematical expressions. They are encoded in linear strings of functional symbols, variables and constants. GEP individuals are free from any constraints regarding the form of the expression they encode. Due to the linear encoding, GEP benefits from the phenotype-genotype separation [10].

The GEP individual is a linear string of symbols – functional symbols, variables (representing inputs to the model) and constants. An example of such an individual, in the context of forecasting time series, is presented in Figure 1. It is composed of three genes. Each gene encodes a different expression tree, represented circled in Figure 1(a). This encoding is obtained by a breadth first traversal of the expression tree. As it can be noted, each tree decodes to a valid mathematical expression. For example, the second gene has the first symbol "-". It is a binary operation, which means that on the second level in the sub-tree we will find the operands of the "-" operation. These operands are the next symbols from the gene, respectively x_3 and x_1. Since both of these symbols are constants, it means that the decodification of the gene ends here. The resulting mathematical expression encoded by the gene is thus $x_3 - x_1$. The rest of the symbols in the gene are inactive for the moment. In the chromosome presented in Figure 1, the three expressions encoded by the genes are finally linked by means of the linking operator (addition in this case), resulting the mathematical model encoded by the GEP individual.

The general structure of the GEP algorithm is depicted in Figure 1(b). In our experiments, the termination criterion is the maximum number of generations and the solution is the best individual throughout alll generations.

For more in depth information on GEP and its applications, we refer the reader to Ferreira's excellent monograph [10].

(a) GEP individual: the mathematical model can be expressed as an hierarchical expression tree and is encoded as three linear genes in the algorithm.



(b) The general structure of the GEP algorithm.

```
t ← 0
  1.Generate the initial population of candidate solutions $P_0$.
  2.while (termination criterion is not met) do
      (a)$t \leftarrow t+1$
      (b)Select $P_t$ from $P_{t-1}$ using a selection scheme
      (c)Evolve the individuals in $P_t$ using genetic operators
         (mutation, recombination, transposition)
      (d)Evaluate individuals in $P_t$ and assign them fitness values
  3.designate solution
```

Fig. 1: The Gene Expression Algorithm.

3.1 Adaptive GEP

In this paper we use an adaptive variant of the GEP algorithm, referred as AdaGEP in the following.

An important step in the design of a GEP-based algorithm is to choose the chromosomal architecture. The number of genes in the chromosome dictates the complexity of the encoded solutions. We proposed in [11] a hybridization of the GEP algorithm with a bit string Genetic Algorithm that solves the problem of finding the appropriate number of active genes to be used by the algorithm during a run. The adaptive mechanism implemented in AdaGEP allows the algorithm to change the number of active genes during evolution. The adaptation takes place at the chromosome level, hence different chromosomes may have different numbers of active genes. This is particularly useful for the problem of time series forecasting, when no information is available with respect to the expected model type and/or complexity. Comparative studies of AdaGEP and GEP models for time series forecasting proved the better performance of AdaGEP [11,12].

We implemented AdaGEP as an extension of the gep package for ECJ[1].

4 Support Vector Regression

The SVM algorithm is built upon the foundation offered by the theory of statistical learning. It was initially designed to solve classification problems. In basic two class classification the goal is to determine an optimal hyperplane that separates the two classes. The SVM algorithm approaches this problem by mapping the training data into a higher dimensional feature space using a function Φ and then constructing, in the new feature space, a maximum margin separating hyperplane of the two classes. The support vectors are the points on the boundary of the classes that are closest to the separating hyperplane. The idea is to transform the input data into a new feature space where the data is linearly separable. The SVM algorithm takes advantage of the "kernel trick" by using a kernel function to compute the hyperplane without explicitly computing the mapping into the feature space [16].

In the following, we briefly present the SVR algorithm, following the presentation in [13,16].

The basic idea behind Support Vector Regression is to map the data x into a higher dimensional feature space $\mathscr{F}$ using a nonlinear mapping Φ, and then to solve a linear regression problem in the new space [14,15,16]:

$$f(x) = (\omega \cdot \Phi(x)) + b, \text{ with } \Phi : R^n \to \mathscr{F},\ \omega \in \mathscr{F}, \quad (2)$$

where b is a threshold. In ε-SVR we want to find a function f that has at most ε deviation from the actual observed values for each training datum in the dataset.

In the case of linear functions, we have to solve the following optimization problem:

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2}\|\omega\|^2 \\ &\text{subject to:} && y_i - \omega x_i - b \le \varepsilon, \\ & && \omega x_i + b - y \le \varepsilon. \end{aligned}$$

[1] ECJ is an open-source evolutionary computation research system developed in Java at George Mason University's Evolutionary Computation Laboratory and available at http://cs.gmu.edu/ eclab/projects/ecj/

In case of infeasible constraints, slack variables ξ, ξ^* are introduced [13]. The constrained optimization problem becomes:

$$\begin{aligned} &\text{minimize} \quad \tfrac{1}{2}\|w\|^2 + C\textstyle\sum_{i=1}^{m}(\xi + \xi^*) \\ &\text{subject to:} \quad y_i - \omega x_i - b \leq \varepsilon + \xi^*, \\ &\qquad \omega x_i + b - y < \varepsilon + \xi, \\ &\qquad \xi, \xi^* \geq 0, \end{aligned}$$

for all $i = 1, \dots, m$. Parameter ε controls the number of support vectors and errors that exceed a given threshold. The constant $C > 0$ determines the trade-off between the flatness of $f(x)$ and the amount up to which deviations larger than ε are tolerated [13].

To make the algorithm nonlinear, the training patterns are processed by the mapping Φ as described in equation 2 and then the standard SVR algorithm is applied. In practice, the dual formulation of problem 3 is often more easily solved [17]:

$$\begin{aligned} &\min_{\alpha,\alpha^*} \quad \tfrac{1}{2}(\alpha - \alpha^*)^T Q(\alpha - \alpha^*) + \\ &\qquad \varepsilon \textstyle\sum_{i=1}^{m}(\alpha_i + \alpha_i^*) + \\ &\qquad \textstyle\sum_{i=1}^{m} z_i(\alpha_i - \alpha_i^*) \\ &\text{subject to:} \quad e^T(\alpha - \alpha^*) = 0, \\ &\qquad 0 \leq \alpha_i, \alpha_i^* \leq C, \forall i = 1, \dots, m, \end{aligned}$$

where

$$Q_{ij} = K(x_i, x_j) = \Phi(x_i)^T \Phi(x_j)$$

and K is a kernel function. The approximate solution is

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

The types of kernel functions that are most used are: linear, polynomial, radial basis function (RBF) and sigmoid [18]. The kernel function used in this paper is the RBF kernel:

$$K(x_i, x_j) = \exp -\gamma \left\|x_i - x_j\right\|^2, \qquad (3)$$

where γ is a parameter that needs to be set prior to running the SVM algorithm.

In this work, we used the SVM implementation provided by the software library LibSVM[2] [17]. For a detailed description of SVR, we refer the reader to [13].

5 Methodology

Each series is preprocessed and turned into a set of w-dimensional data $(x_{t-w}, \dots, x_{t-1}, x_t)$. Using AdaGEP and SVR, we search the prediction function that uses $(x_{t-w}, \dots, x_{t-1})$, $w + 1 \leq t \leq n$ as input variables in order to approximate x_t. The appropriate window size w is searched using additional information we have on the time series, while at the same time trying out numerous combinations.

We report in the following the results obtained with our selection of window size. For the annual time series, $w = 5$ was found to offer most satisfactory results. Hence, the temperature in a particular year is predicted using the mean temperature in the previous 5 years. For the monthly temperature series, we used as input for the model the value in the previous month and the value in the same month, recorded last year, hence $x_t = f(x_{t-1}, x_{t-12})$, ($\tau = 1$, $d = 13$ in equation 1).

The original datasets are divided into separate training and test sets. The training set is used in the learning phase. The test set is used to verify the generalization capability of the model; it is not used during trainig. For the monthly data, the test set contains the last 24 items, meaning that the models obtained have to predict the behaviour of the variable over the next 24 months. For the annual data, the test set contains the last 10 items, which means that the models are used to predict mean annual temperatures for the next 10 years.

For AdaGEP, the raw data is used to train the algorithm and then to test the model. The fitness function is based on the MSE, meaning that better individuals have smaller MSEs. The operator rates are used at standard values proposed in the literature [6,9,10]. We note that the total number of genes in a chromosome is 5, but the algorithm searched for the optimal number of active genes during evolution. The number of independent runs is 50. We report the best solution model identified over all runs.

For the application of the SVR algorithm, we follow the guidelines proposed in [17]. First the original data are scaled into the range of $[-1, 1]$ in order to independently normalize each feature component to the specied range. The procedure is performed for both training and test data. The purpose is to ensure that larger values of the input attributes do not overwhelm small value inputs.

For the ε-SVR employed in our study, we need to find the appropriate parameters C and γ. We use a grid search procedure to identify the best (C, γ) combination of parameters, in a 10-fold cross-validation scheme. This way, the SVR algorithm is supposed to avoid overfitting the training data [18]. The best (C, γ) pair is then used to train the SVR algorithm on the training dataset. The resulting model is afterwords used to predict the unseen values in the test set.

5.1 Performance measures

The efficiency of the models obtained with AdaGEP and SVR is assessed using as performance indicators the Mean Absolute Prediction Error (MAPE) and Mean Squared Error (MSE):

$$MSE = \frac{\sum_{i=1}^{no_o}(x_i - x_i^*)^2}{n}$$

[2] Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

$$MAPE = \frac{\sum_{i=1}^{n} \left| \frac{x_i - x_i^*}{x_i} \right|}{n}$$

where x_i is the i-th value in the original time series and x_i^* is the value predicted by the model. Note that the *MSE* is a scale-dependent accuracy measure, while the *MAPE* is scale independent, hence it can be used to compare forecast performance across different data sets [20].

AdaGEP is a probabilistic algorithm, so for consistency purposes, besides the MSE of the best model, we report the mean MSE and the standard deviation of the MSE of the solutions obtained in all independent runs for a given dataset. The comparisons between the models with respect to MSE and MAPE are statistically validated using standard t-tests to check for significant differences in means (at significance level 0.05).

The Kolmogorov-Smirnov test is used to decide whether a sample of data comes from a population with a specific distribution. In our case, we use it to decide whether two independent samples come from the same *unknown* distribution. For example, we have a sample that contains monthly temperatures recorded for 24 consecutive months and a sample of data generated by the AdaGEP model, and we want to test if the these samples come from the same population. If the test rejects the null hypothesis (that they *do* belong to the same population), it means that the model did not fit the proper distribution of the original data, hence its forecasts are questionable.

5.2 Datasets

The experiments carried out in the present study used series of temperature and precipitations, collected at several stations in the Black Sea region. The data series are described in Table 1.

Series	Station	Type	Variable	Period
CMP	Constanta	Mon.	Precip.	01.1965 – 12.2005
SMP	Sulina	Mon.	Precip.	01.1965 – 12.2005
CMT	Constanta	Mon.	Temp.	01.1961 – 12.2008
SMT	Sulina	Mon.	Temp.	01.1961 – 12.2008
CAT	Constanta	Ann.	Temp.	1965 – 2005
JAT	Jurilovca	Ann.	Temp.	1965 – 2005
SAT	Sulina	Ann.	Temp.	1965 – 2005
TAT	Tulcea	Ann.	Temp.	1965 – 2005

Table 1: Description of the datasets. Note: precip is short for precipitation, temp. is short for temperature, mon. is short for monthly, ann. is short for annual

6 Results and discussion

The experiments reported in this paper were composed of a training phase and a testing phase for both AdaGEP and SVR, for each dataset. The models produced were evaluated with respect to MSE and MAPE. The main objective was to produce forecasts, hence the performance comparison on the test data set is of most importance. Nevertheless, we also report the performance indicators values obtained in the training phase, in order to give a complete account on how well the methods performed. The summary of results is contained in tables 2, 3, 4 and 5.

According to the MSE on the test set (Table 2), it is obvious that the models obtained with AdaGEP offer significantly better predictions than the models obtained with SVR, since their MSEs are significantly smaller (by several orders of magnitude, excepting the case of series SMP).

Series	AdaGEP model MSE	SVR model MSE
	Test	
CAT	0.98	18.9
TAT	0.93	7.61
SAT	1.51	6.29
JAT	0.75	3.33
CMT	5.80	54.28
SMT	6.67	60.55
CMP	42.70	3639.18
SMP	2202.71	2409.38

Table 2: MSE for the models on the *test* dataset.

During the learning phase, for the datasets of annual temperatures, the MSEs reported by the SVR models are significantly smaller than those of the AdaGEP models (Table 3). This means that the SVR models learned better the training data. For example, one can observe in Figure 2 that the SVR model overlaps perfectly the original training data, while the GEP model does not. Given that the SVR model predictions on the (previously unseen) test dataset are worse with respect to MSE (Figure 3), we argue that the SVR models overfit the training data.

In the literature, overfitting is often reported as a result of a poor choice of parameters. We remind the reader that the choice of parameters C and γ in SVR was performed doing a grid search in a 10-fold cross-validation procedure. This procedure was used with success to avoid overfitting [17, 18]. Hence, although we do not exclude the parameter choices as main cause of overfitting, future work will concentrate on dealing with this issue.

Regarding the AdaGEP algorithm, by analysing the Mean MSE and the standard deviation of the MSE of all 50 solutions obtained in each experiment, we conclude that AdaGEP is stable and produces consistent results.

The MAPE values are independent of the scale of the data and are used to compare forecast accuracy across many series [20].

For the annual series, we note that on the training dataset, the MAPE values are almost identical. Since the series contain mean annual temperature values from four

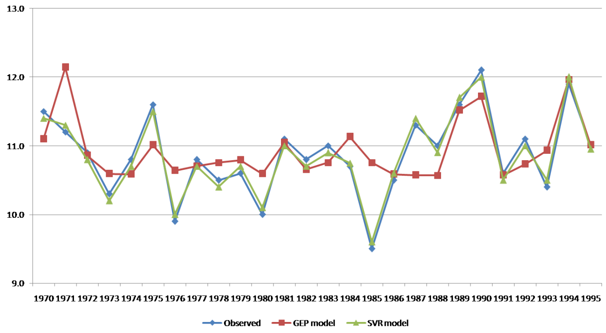


Fig. 2: The plot of actual observed values versus the values predicted by the AdaGEP and the SVR models, on the *training* dataset of the *JAT* series.

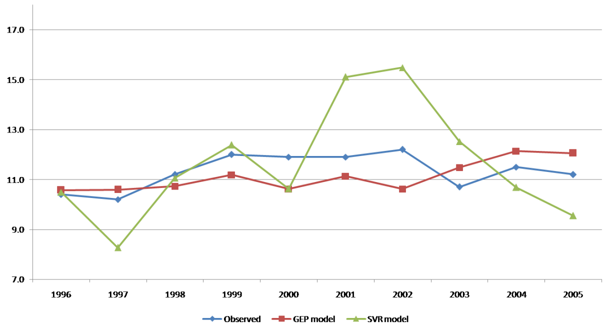


Fig. 3: The plot of actual observed values versus the values predicted by the AdaGEP and the SVR models, on the *test* dataset of the *JAT* series.

Series	AdaGEP model MAPE (%)	SVR model MAPE (%)
	Test	
CAT	0.06	0.26
TAT	0.06	0.20
SAT	0.07	0.17
JAT	0.06	0.12
CMT	0.17	0.71
SMT	0.05	0.97
CMP	2.88	1.67
SMP	1.15	1.36

Table 4: MAPE for the models on the *test* dataset.

Series	AdaGEP model MAPE (%)	SVR model MAPE (%)
	Training	
CAT	0.03	0.009
TAT	0.04	0.009
SAT	0.03	0.01
JAT	0.03	0.008
CMT	1.83	5.79
SMT	0.28	3.23
CMP	3.00	1.85
SMP	2.16	1.13

Table 5: MAPE for the models on the *training* dataset.

stations in the same region, it is clear that they have very similar characteristics, which explains why the algorithms learned them equally well. It is also to be noted that on training, the MAPE of the SVR models was smaller than those of the AdaGEP models, which reinforces the fact that the SVR models learned the training data better. But the AdaGEP models dominate SVR models in predicting mean annual temperatures with respect to MAPE, too, and comes to support our hypothesis of the overfitting of SVR models.

Series	AdaGEP model MSE (Mean, Stdev)	SVR model MSE
	Training	
CAT	0.31 (0.32, 0.02)	0.39
TAT	0.33 (0.41, 0.03)	0.01
SAT	0.28 (0.37, 0.01)	0.06
JAT	0.21 (0.31, 0.02)	0.009
CMT	5.82 (5.16, 0.13)	63.55
SMT	4.65 (4.84, 0.09)	66.55
CMP	1458.51 (1458.51,0)	629.60
SMP	814.77 (818.04, 1.47)	598.52

Table 3: MSE for the models on the *training* dataset. For the AdaGEP models, we report in the parantheses statistics (Mean MSE, standard deviation of the MSE) over all 50 solutions designated in the independent runs of the algorithm.

For the monthly temperature series (CMT and SMT), the situation is somewhat similar, in the sense that the AdaGEP models have significantly smaller MAPE than the SVR models on test data. The same is not true for the monthly precipitation series at Constanta. The MAPE on the forecast (test) data of the SVR model for CMP is significantly smaller than the MAPE of the AdaGEP model (although the situation was reversed with respect to MSE). Also, for the SMP, the MAPE on test data of the SVR does not differ significantly from the MAPE of the GEP.

By visual inspection of the graphs of the observed values plotted against the values predicted by the models, we gain more insight into the results. Due to space limitations, we do not include in the paper the graphs of the actual versus the predicted values for all the time series employed in the study. We only include a selection that we consider to be representative. For example, since the annual series are very similar, we include, for exemplification purposes, the chart of the forecasts by the AdaGEP and SVR models for the test dataset for the Jurilovca series (Figure 3). The plot of the models predictions against the training set reflect the fact that both the AdaGEP model and the SVR model learned

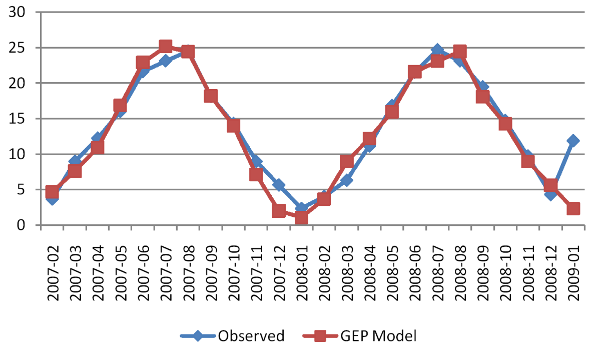


Fig. 4: The plot of actual observed values versus the values predicted by the AdaGEP model, on the *test* dataset of the *CMT* series. As it can be noted, the predictions of the AdaGEP model are almost perfect.

almost perfectly the training data (in the case of the annual series).

Figure 4 represents the plot of the actual monthly temperatures at Constanta station against the values predicted by the GEP model, from the test dataset. The same is true for the monthly temperature series at Sulina (SMT), which resembles the monthly temperature series from Constanta (CMT). Therefore the graphs for SMT are not included.

We include the charts of the monthly precipitations at Constanta, for both the training and the test datasets (figures 6 and 5). We can easily observe that, although the AdaGEP model is superior to the SVR model both in terms of MSE and of MAPE, the forecasts on the test data are not very satisfactory. Some observations are mandatory: the CMP series is very long, covering a period of 40 years. There may exist multiple change points, where the process behaviour changed, meaning that the derivation of a single model to characterize and predict the series may be unappropriate. Further work on this issue is needed. Also, we emphasize that we constructed models using only the series data, without any supplementary variables. The inclusion of such variables in constructing the models may improve the forecasts.

Similar remarks can be made for the Sulina monthly precipitation data. The complexity of the dataset is visible in Figure 7, which depicts the models obtained on the training data. Although the MSE and the MAPE of the SVR model are slightly larger than those of the AdaGEP model, indicating the AdaGEP model as a better forecaster, from Figure 8 we would be tempted to say that the SVR model appears to follow more closely the trend of the original data.

We used the Kolmogorov-Smirnov goodness of fit test to check whether the data forecasted by each model matched the statistical distribution of the original sample data. For the monthly temperature data (SMT and CMT), the KS test reveals that the AdaGEP model generates data from the same distribution like the original data, both on training and on test. For SMT and CMT, the distribution of the SVR model predicted data differs significantly from the observed data distribution. For the monthly precipitation data CMP, the data forecasted by the AdaGEP model and the data predicted by the SVR model pertain to the same statistical distribution like the original test data (the p-values obtained in the KS test were 0.675 for AdaGEP and 0.441 for SVR). For SMP, there are no significant differences between the SVR model predictions and the original data, but the distribution of the original data differs significantly from that of the AdaGEP model predictions. For all the annual temperature series, there were no significant differences in the distributions of the original observed data and the data generated by the SVR models on the training data. On the test data, the distributions were similar among the AdaGEP models, but differed significantly for the SVR models.

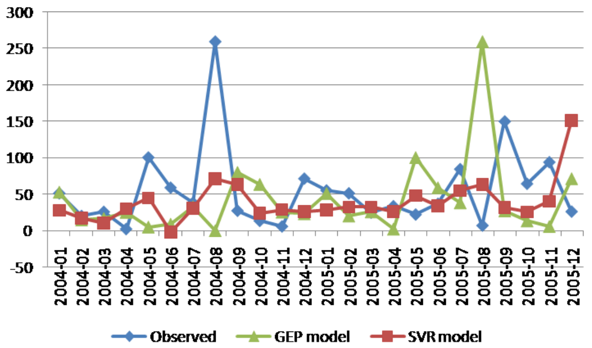


Fig. 5: The plot of actual observed values versus the values predicted by the AdaGEP and the SVR models, on the *test* dataset of the *CMP* series.

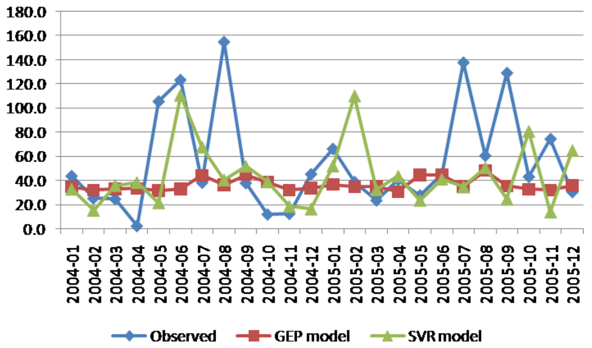


Fig. 8: The plot of actual observed values versus the values predicted by the AdaGEP and the SVR models, on the *test* dataset of the *SMP* series.

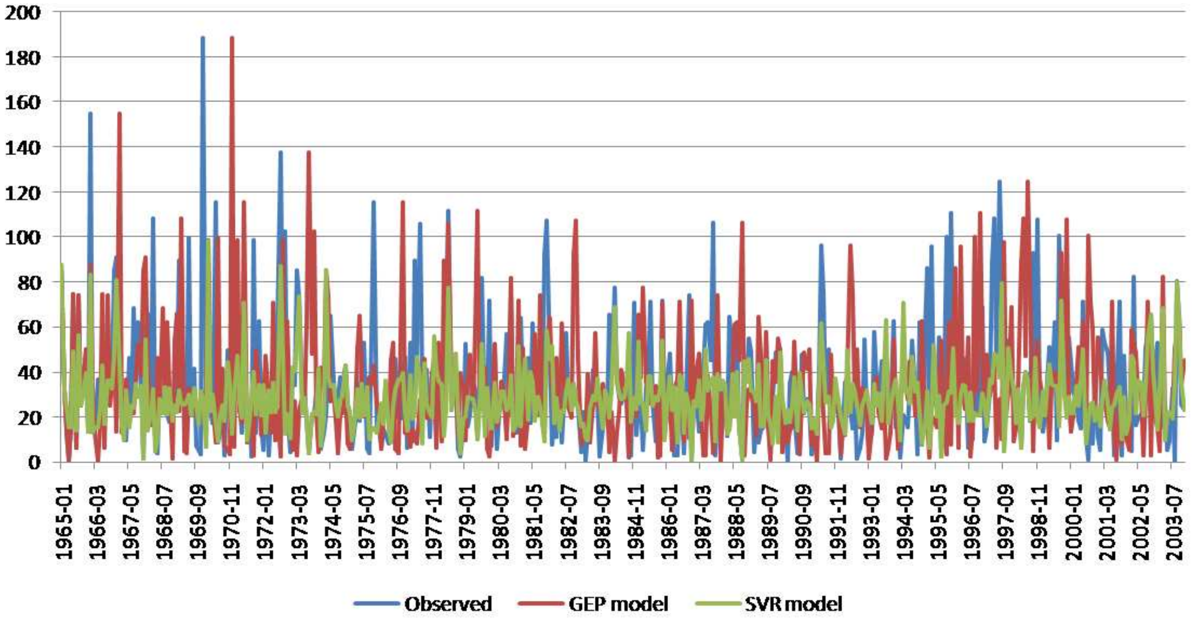


Fig. 6: The plot of actual observed values versus the values predicted by the AdaGEP and the SVR models, on the *training* dataset of the *CMP* series.

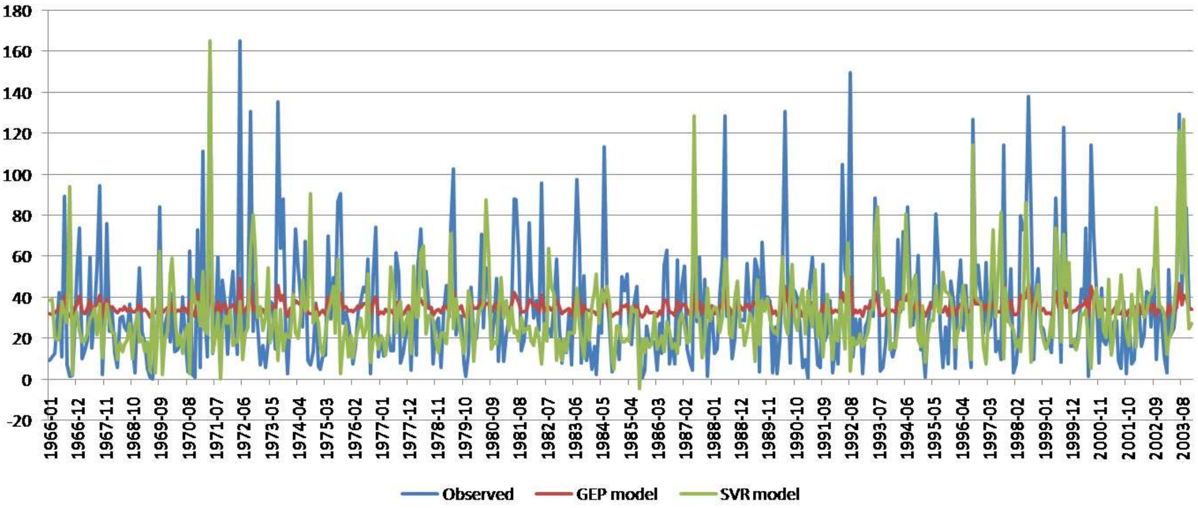


Fig. 7: The plot of actual observed values versus the values predicted by the AdaGEP and the SVR models, on the *training* dataset of the *SMP* series.

7 Conclusion

We have examined how well AdaGEP and SVR work for the prediction of future values in time series representing records of temperature and precipitation, gathered at several locations in the Black Sea region. We evaluated the models using two performance measures – one that is dependent on the scale of the input data (MSE) and one that is independent of the scale (MAPE). We note that the perfomances of the models obtained with the two methods vary among the datasets. The size and the structure of the training set affects the modeling and forecasting in a significant manner.

Both SVR and AdaGEP show excellent learning of the training dataset, for both annual and monthly series.

The models produced learned almost perfectly the training datasets. SVR scores for the training data were better on the annual time series and on the monthly temperature series. The same is not true for the prediction test set. The results demonstrate that AdaGEP is more competent, in terms of MSE, for forecasting in all situations. Testing with KS of goodness of fit, the results indicate that AdaGEP identified the distributions in more cases than SVR. Overall, the comparison results indicate that the AdaGEP models perform better than the SVM models for forecasting the temperature and precipitation based on the particular data sets used in this study.

Future work will concentrate on statistical analysis of the time series as a preprocessing stage (e.g. change point detection). Also, as far as SVR is concerned, we need to revise the reasons behind the overfitting behaviour. Also, since SVR relies on the expertise of the researcher for setting its parameters, an interesting idea would be to evolve the kernel function and the SVR parameters.

References

[1] J. Adamowski, H. Fung Chan, S. O. Prasher, B. Ozga-Zielinski and A. Sliusarieva, Comparison of multiple linear and nonlinear regression, autoregressive integrated moving average, artificial neural network, and wavelet artificial neural network methods for urban water demand forecasting in Montreal, Canada, Water Resour. Res., **48, W01528**, 14 PP., (2012)

[2] J.G. De Gooijer and R.J. Hyndman. 25 years of time series forecasting. Intl. Journal of Forecasting, **22(3)**, 443 – 473, (2006).

[3] H. Niskaa, T. Hiltunena, A. Karppinenb, J. Ruuskanena, M. and Kolehmainen, Evolving the neural network model for forecasting air pollution timeseries, Engineering Applications of Artificial Intelligence, **17(2)**, 159 – 167, (2004).

[4] P. Coulibaly, Downscaling daily extreme temperatures with genetic programming, Geophysical Research Letters, **31**, L16203, (2004).

[5] Y.-S. T. Hong, P.A. White and D.M. Scott, Automatic rainfall recharge model induction by evolutionary computational intelligence, Water Resources Research, **41**, W08422, 13 PP, (2005).

[6] A. Barbulescu and E. Bautu, Mathematical models of climate evolution in Dobrudja, Theoretical and Applied Climatology, **100(1-2)**, 29 – 44, (2010).

[7] A. Barbulescu and E. Bautu, A Hybrid Approach for Modeling Financial Time Series, Int. Arab J. of Inf. Tech, **9(4)**, 327 – 335, (2012).

[8] N. Sapankevych and R. Sankar, Time Series Prediction Using Support Vector Machines: A Survey, IEEE Computational Intelligence Magazine, **4(2)**, 24 – 38, (2009)

[9] C. Ferreira, Gene expression programming: a new adaptive al-gorithm for solving problems, Complex Systems, **13(2)**, 87 – 129, (2001).

[10] C. Ferreira, Gene Expression Programming: Mathematical Modeling by an Artificial Intelligence, 2nd Edition, (Springer-Verlag, Germany, 2006).

[11] E. Bautu, A. Bautu and H. Luchian, AdaGEP - An Adaptive Gene Expression Programming Algorithm, pp. 403 – 406, Ninth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2007), (2007).

[12] A. Barbulescu and E. Bautu, Meteorological time series modeling using an adaptive gene expression programming, Proceedings of the 10th WSEAS international conference on evolutionary computing, p.17 – 22, Prague, Czech Republic (2009).

[13] A.J. Smola and B. Scholkopf, A tutorial on support vector regression. Statistics and Computing **14(3)**, 199 – 222, (2004).

[14] V. Vapnik, The Nature of Statistical Learning Theory, (Springer Verlag, New York, 1995).

[15] H. Ince, Support Vector Machine for Regression and Applications to Financial Forecasting. In Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00), Vol. 6. IEEE Computer Society, Washington, DC, USA, 6348-, (2000).

[16] K.-R. Muller, A.J. Smola, G. Rotsch, B. Schokopf, J. Kohlmorgen and V. Vapnik. 1999, Using support vector machines for time series prediction. In Advances in kernel methods, Bernhard Scholkopf, Christopher J. C. Burges, and Alexander J. Smola (Eds.), 243 – 253, (MIT Press, Cambridge, MA, USA, 1999).

[17] C.-C. Chang and C.-J. Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, **2:27:1–27:27**, (2011).

[18] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, A practical guide to support vector classification. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, 2003.

[19] F. Takens, Detecting strange attractors in fluid turbulence. In D. Rand and L.S. Young, editors, Dynamical Systems and Turbulence, 366 – 381. (Springer-Verlag, Berlin, 1981).

[20] R.J. Hyndman and A.B. Koehler, Another look at measures of forecast accuracy, International Journal of Forecasting, Elsevier, **22(4)**, 679 – 688, (2006).

Elena Bautu received her BSc degree in Computer Science from Al. I. Cuza University, Iasi, Romania in 2003, her MSc degree in Applied Mathematics from the Ovidius University, Constanta, Romania in 2005 and the PhD degree in Computer Science (Artificial Intelligence) from Al. I. Cuza University, Iasi, Romania in 2010. Her research interests include evolutionary computation and data mining, with a focus on time series forecasting. She is the author of over 30 research articles, of which 16 are indexed in the ISI Thomson database. She served as reviewer for Software Engineering, Transactions on Computers, Transactions on Signal Processing, Geophysical Research Letters, Il Nuovo Cimento B "Basic Topics in Physics", Journal of Medicine and Medical Sciences.

Alina Barbulescu Studies: Bachelor in Mathematics and Law, Bucharest University, Master in Analysis and Operators Theory (University of Bucharest, Romania), PhDs. in Mathematics (Al. I. Cuza University, Iasi, Romania) and Economics (Academy of Economic Studies, Bucharest, Romania). Publication: 90 articles (18 ISI), 18 books. Reviewer: Geophysical Research Letters; Journal of Hydrology; Water SA; Chemical Engineering Communications; An. St Univ. Ovidius Constanta, Mathematica; International Journal Mathematics and Computation, International Journal Mathematical Manuscripts, Journal of Applied Polymers Sciences, International Journal of Mathematics and Mathematical Sciences Editor: International Journal of Mathematics and Computation (and International Journal of Applied Mathematics and Statictics;Invited editor: Mathematical Methods, Computational Techniques, Intelligent Systems, WSEAS Press, 2010, ISSN: 1790-2769 152, ISBN: 978-960-474-188-5; Analele Stiintifice ale Universitatii Ovidius Constanta, seria Matematica, vol.XVII (3), 2009; Annals of Ovidius Univ. of Constanta, Civil Engineering Series, Special issue dedicated to the 5-th International Conference Dynamical Systems and Applications, 2009; Seminar Series in Mathematics, Miscellanea 1: Second Part of the Proceedings of Conference of Romanian Math. Soc., Ovidius University Press, Constanta, 2000 Chairperson: MAMECTIS10, Sousse, Tunisia, 3-6.05.2010; The 5th ICDSA, Constanta, Romania, 15 - 18. 06.2009;7-th World Congress in Probability and statistics, Singapore, 14-19.07.2008;Conference 2007: Dynamical systems and applications, Izmir, Turkya, 1-6.07.2007, 22nd European conference on operational research, Prague, 8-11.07.2007; Conference 2004: Dynamical systems and applications, Antalya, Turkya, 5-10.07.2004; International Conference on Applied Mathematics, Baia Mare, Romania, 23-26.09.2004.