ESANN'2006 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 26-28 April 2006, d-side publi., ISBN 2-930307-06-4.

Time Series Prediction using DirRec Strategy

Antti Sorjamaa and Amaury Lendasse *

Helsinki University of Technology - Neural Network Research Centre
P.O. Box 5400, 02015 - Finland

Abstract. This paper demonstrates how the selection of Prediction Strategy is important in the Long-Term Prediction of Time Series. Two strategies are already used in the prediction purposes called Recursive and Direct. This paper presents a third one, DirRec, which combines the advantages of the two already used ones. A simple k-NN approximation method is used and all three strategies are applied to two benchmarks: Santa Fe and Poland Electricity Load time series.

1 Introduction

Time Series Prediction [1] is a challenge in many fields. In finance, one predicts stock exchange courses or stock market indices; data processing specialists predict the flow of information on their networks; producers of electricity predict the load of the following day. The common point to their problems is the following: how can one analyse and use the past to predict the future?

In many occasions, it's fairly simple and straightforward to predict the next value of the time series. But the further we delve into the future, the more uncertain we are and the bigger prediction errors we get.

The concept of *Long-Term* is not precise. It somewhat describes the amount of timesteps to be predicted toward the unknown future. One can find or create many different interpretations and definitions to the concept. In this paper, we predict 100 steps ahead and based on any definition that is considered to be long-term.

This paper demonstrates how the selection of Prediction Strategy is very important in the long-term prediction of time series. Two strategies are already used in the prediction purposes called Recursive and Direct. This paper presents a third one, DirRec, which combines aspects of the two already used ones.

In the next section, the three prediction strategies are explained and their pros and cons discussed from a theoretical point of view. In Section 3, the k-NN approximation method is presented with structure and input selection methods used. Finally, in Section 4, the prediction strategies are compared and the performances evaluated from a practical point of view.

*Part the work of A. Sorjamaa and A. Lendasse supported by the project of New Information Processing Principles, 44886, of the Academy of Finland. The work of A. Lendasse is supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

2 Time Series Prediction Strategies

2.1 Recursive

In the Recursive prediction strategy, the same model is used over and over again and the previous predictions are used with the original data set as inputs to evaluate the next prediction. In this way, we are applying one-step-ahead prediction many times, recursively.

If we use 4 previous values of the time series as inputs, we can write the Recursive strategy as

$$\begin{aligned} \hat{y}(t+1) &= f_{q_r}(y(t), y(t-1), y(t-2), y(t-3)), \theta(q_r)), \\ \hat{y}(t+2) &= f_{q_r}(\hat{y}(t+1), y(t), y(t-1), y(t-2)), \theta(q_r)), \\ \hat{y}(t+3) &= f_{q_r}(\hat{y}(t+2), \hat{y}(t+1), y(t), y(t-1)), \theta(q_r)), \\ &\vdots \end{aligned} \tag{1}$$

where $y(t)$ to $y(t-3)$ denote the 4 last values of the time series, $\hat{y}(t+1)$ to $\hat{y}(t+3)$ denote the 3 first predictions and $\theta(q_r)$ inholds the model parameters for the model structure f_{q_r}. The further we go, the more approximations are introduced to the input set and after $t+4$ all the inputs are approximations. Here the model structure q_r stays the same in each timestep. But if the errors are non-zero and the approximations are used as inputs again and again, the more and more cumulative prediction error is included in the approximations. In practice, this is the normal case, there is some error in every approximation, because the time series are partly stochastic processes.

2.2 Direct

Comparing to the Recursive strategy, the Direct strategy uses different models for each time step but always the real measured data as inputs. No approximations are introduced to the input set. Again, if we use 4 previous values of the time series as inputs, we can write the Direct strategy as

$$\begin{aligned} \hat{y}(t+1) &= f_{q_{d1}}(y(t), y(t-1), y(t-2), y(t-3)), \theta(q_{d1})), \\ \hat{y}(t+2) &= f_{q_{d2}}(y(t), y(t-1), y(t-2), y(t-3)), \theta(q_{d2})), \\ \hat{y}(t+3) &= f_{q_{d3}}(y(t), y(t-1), y(t-2), y(t-3)), \theta(q_{d3})), \\ &\vdots \end{aligned} \tag{2}$$

In this strategy, there's no cumulative error introduced through the inputs, because only original data set values are used in the approximation of future values. Each time step only the normal prediction error is present and there is no cumulation of prediction errors. Every time step incorporates its own model and may also have its own selection of inputs, if the input selection is used. These selections increase the calculation time considerably, but in practice give better results in the long-term prediction due to the lack of cumulative error [2].

2.3 DirRec

The DirRec strategy combines aspects from both, the DIRrect and the RECursive strategies. It uses a different model at every time step and introduces the approximations from previous steps into the input set. If we use 4 previous values of the time series as inputs, we can write the DirRec Strategy as

$$\begin{aligned} \hat{y}(t+1) &= f_{q_{dr1}}(y(t), y(t-1), y(t-2), y(t-3)), \theta(q_{dr1})), \\ \hat{y}(t+2) &= f_{q_{dr2}}(\hat{y}(t+1), y(t), y(t-1), y(t-2), y(t-3)), \\ &\quad \theta(q_{dr2})), \\ \hat{y}(t+3) &= f_{q_{dr3}}(\hat{y}(t+2), \hat{y}(t+1), y(t), y(t-1), y(t-2), \\ &\quad y(t-3)), \theta(q_{dr3})), \\ &\vdots \end{aligned} \tag{3}$$

Every time step the input set in increased with one more input, the approximation of the previous step. When we use the input selection, we can determine if the approximation is accurate enough to be included in the next step and so on. So in a way, the DirRec strategy gives not only the prediction of each step, but also information about the validity of the approximations done in the previous steps. If no input selection is used, the complexity of the model increases linearly and more and more inputs with prediction error are fed into the model. The cumulative prediction error increases also linearly, but is always less than in the Recursive strategy, because the real measurements remain as inputs.

3 *k*-Nearest Neighbors

The k-Nearest Neighbors (k-NN) approximation method is a very simple, but powerful method. It has been used in many different applications and particularly in classification tasks [3]. The key idea behind the k-NN is that similar training samples have similar output values. One has to look for a certain number of nearest neighbors, according to the Euclidean distance [3], and their corresponding output values to get the approximation of the desired output.

We calculate the estimation of the output simply by using the average of the outputs of the neighbors in the neighborhood as

$$\hat{y}_i = \frac{\sum_{j=1}^{k} y_{P(j)}}{k}, \tag{4}$$

where $\hat{y}_i$ represents the output estimation, $P(j)$ is the index number of the j^{th} nearest neighbor of sample $\mathbf{x}_i$ and k is the number of neighbors used.

We use the same neighborhood size for every data point, so we use a global k, which must be determined beforehand. The determination of the number of neighbors is done by the well known Leave-one-out method [4].

The k-NN has to somehow deal with the input selection problem, but in the case of k-NN, it is not so big problem, thanks to the simplicity of the k-NN.

The following section describes an input selection method called Forward-Backward algorithm used in the experiments.

3.1 Input Selection

When all possible input sets of size d are tested, it means that 2^d possible input variable sets are built and evaluated. d represents the maximum number of variables to be used in the evaluation. This kind of search is very time consuming, but it is guaranteed to give the global optimum in the defined search space.

The Forward-Backward Selection limits the number of input sets to be tested and is thus many times faster than the exhaustive search through the whole input set space. It is not guaranteed that in all cases this selection method will find the global optimal input set, even in theory it should be closer to the global optimum than the basic Backward or the Forward methods alone.

The state of each variable is changed one at a time: if a variable was already selected into input set, it is removed, and if not, it is put into the input set. This is done for all the inputs from x^1 to x^d and the error approximation using the k-NN and LOO is calculated with every selection. Then the action that gave the smallest error is done permanently to the input set.

In example, if we start with non-continuous input set of size 4 and describe the process of approximating one time step ahead as

$$\hat{y}(t+1) = f_q(y(t), y(t-1), y(t-3), y(t-5)), \theta(q)). \tag{5}$$

The next input set to be evaluated would be the same than in Equation (5) without the input variable $y(t)$. After that, $y(t-1)$ would be taken away from Equation (5) with $y(t)$ in the set and so on. The variable state change that gives the smallest error is permanently done to the input set and the process continues by estimating this new starting input set and all the variable state changes in the new set. In this way it is continued until a stopping criteria is fulfilled and the input set giving the smallest LOO error is selected. The selection method can be started from any input set, even from randomly initialized set.

4 Experimental Results

Two time series are used in the experiments: Santa Fe [1] and Poland Electricity Load [5] data sets. With Santa Fe we use 1000 first values for training, shown in Figure 1, and the rest over 9000 values for testing.

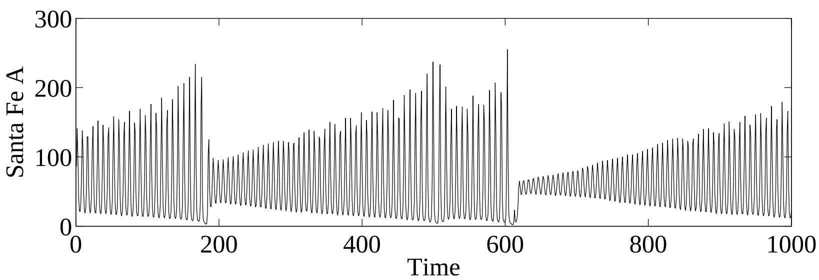

Fig. 1: Training set of Santa Fe data.

With Poland Electricity Load we use 1400 values for training, shown in Figure 2, and 201 values for testing. The maximum regressor size with both time series is 30 and maximum number of neighbors 300.

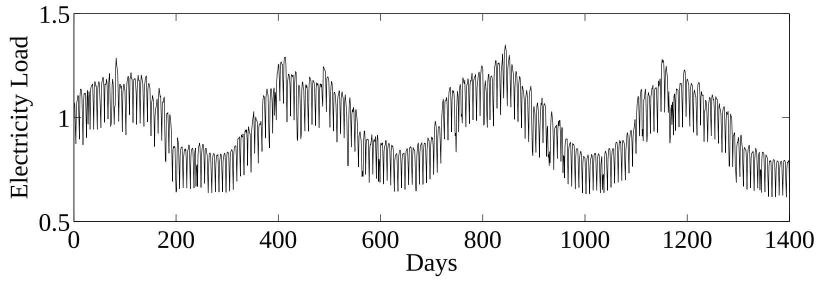

Fig. 2: Training set on Poland Electricity Load data.

In Figures 3 and 4, 100 first values of the test set of Santa Fe and the test set of Poland Electricity Load are predicted using the DirRec strategy.

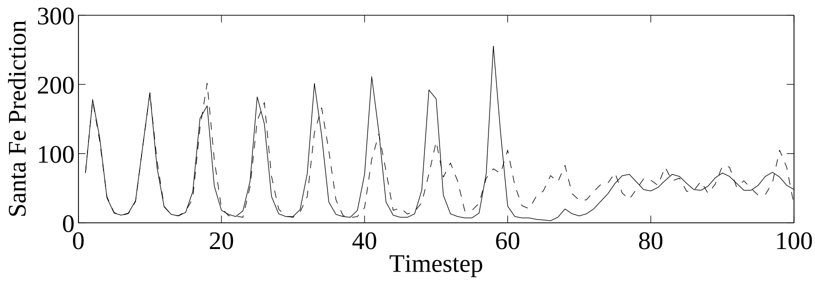

Fig. 3: Santa Fe, first 100 timesteps on the test set using the DirRec strategy. Solid line shows the real values and dashed one the prediction.

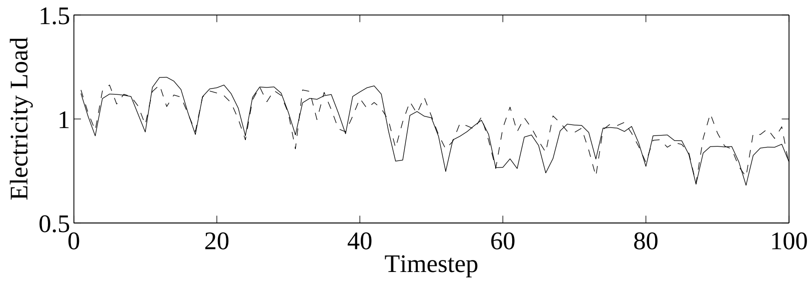

Fig. 4: Poland Electricity Load, first 100 timesteps on the test set using the DirRec strategy. Solid line shows the real values and dashed one the prediction.

Test errors over the whole test sets of the time series are shown in Table 1.

From the results table it can be seen that the Recursive strategy gives the worst performance according to the average test errors. The Direct strategy

	Average Test Errors	
	Santa Fe	Electricity Load
Recursive	3379	0.0318
Direct	1057	0.0124
DirRec	850	0.0098

Table 1: Average test errors of both data sets. The error values are averages of all timesteps with each strategy through the entire test set.

performs much better; it more than halves the average test error of the Recursive strategy with Poland Electricity Load and gives less than third of the error with Santa Fe. The DirRec strategy performs even better still decreasing the test errors on average with both time series.

5 Conclusions

In this paper, we have presented 3 strategies for the Long-Term Prediction of Time Series: Recursive, Direct and DirRec.

The DirRec strategy combines the advantages of the Recursive and Direct strategies and this is illustrated in the experiments section. With the Direct, the MSE on the test sets is reduced by 2/3 compared to the Recursive. With the DirRec, the MSE is reduced by 20 percent compared to the Direct.

The goal of this paper is not to show the quality of the k-NN approximator or the Forward-Backward input selection method. They have been used for the sake of their simplicity and low computational load. In the future, other approximation methods will be used to compare the three strategies.

Computational time for the Direct is roughly the computational time of the Recursive multiplied by the number of predicted timesteps.

In summary, if the calculation time is an issue, we suggest to use the Direct strategy. But if there is a need for a very good accuracy, we suggest to use the DirRec strategy.

References

[1] A.S. Weigend and N.A. Gershenfeld. *Times Series Prediction: Forecasting the Future and Understanding the Past.* Addison-Wesley Publishing Company, 1994.

[2] Yongnan Ji, Jin Hao, Nima Reyhani, and Amaury Lendasse. Direct and recursive prediction of time series using mutual information selections. In *Lecture Notes in Computer Science*, volume 3512 / 2005, pages 1010–1017. IWANN, Vilanova i la Geltrú, Barcelona, Spain, July 2005.

[3] Christopher M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

[4] Gavin C. Cawley and Nicola L. C. Talbot. Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers. *Pattern Recognition*, 36(11):2585–2592, 2003.

[5] Poland Electricity Load data website:
http://www.cis.hut.fi/projects/tsp/?page=Timeseries.