Conditionally dependent strategies for multiple-step-ahead prediction in local learning

Gianluca Bontempi, Souhaib Ben Taieb

Machine Learning Group, Département d'Informatique
Faculté des Sciences, ULB, Université Libre de Bruxelles
1050 Bruxelles - Belgium
e-mail: gbonte@ulb.ac.be, sbentaie@ulb.ac.be

Abstract

Computational intelligence approaches to multiple-step-ahead forecasting rely either on iterated one-step-ahead predictors or direct predictors. In both cases the predictions are obtained by means of multi-input single-output modeling techniques. This paper discusses the limits of single-output approaches when the predictor is expected to return a long series of future values and presents a multi-output approach to long term prediction. The motivation for this work is that, when predicting multiple steps ahead, the forecasted sequence should preserve the stochastic properties of the training series. This is not guaranteed for instance in direct approaches where predictions for different horizons are performed indepedendently. We discuss here a multi-output extension of conventional local modeling approaches and we present and compare three distinct criteria to perform conditionally dependent model selection. In order to assess the effectiveness of the different selection strategies, we carry out an extensive experimental session based on the 111 series of the NN5 competition.

1. Introduction

Multiple-step-ahead prediction (also known as trace forecast) of a time series φ^t in a nonlinear setting is considered a difficult task both in parametric (Guo, Bai, & An (1999)) and nonparametric (Chen, Yang, & Hafner (2004)) time series literature. While under linear assumptions the parametric form of the multiple-step-ahead predictor depends only on the one-step-ahead parametric model, in nonlinear settings the analytical derivation of the multiple-step predictor requires the knowledge of the innovation distribution. Numerical and Monte Carlo methods are proposed in the literature to compute the multiple-step-ahead predictors (see Pemberton (1987); Tong (1983)) when both the nonlinear one-step-ahead predictor and the innovation distribution are known. Additional results for unspecified innovation distributions are given in Guo et al. (1999). These approaches however are based on parametric assumptions and their resulting accuracy depends heavily on the adequacy of the model. In the last two decades, computational intelligence methods (like artificial neural networks (Lapedes & Farber (1987); Zhang, Patuwo, & Hu (1998)) or nearest-neighbors techniques (Farmer & Sidorowich (1987))) have drawn attention in the forecasting community. These methods are examples of nonparametric methods where assumptions on the regression function as well as the innovation distribution are avoided and only historical data are used to learn the stochastic dependency between the past and the future. Currently, the two most common computational intelligence strategies for multiple-step-ahead forecasting are iterated prediction and direct prediction. In the first strategy, a multiple-step-ahead prediction task with horizon H is tackled by iterating H times a one-step-ahead predictor. Once the predictor has estimated the future series value, this value is fed back as an input to the following prediction. Hence, the predictor takes as inputs approximated values instead of actual observations with evident negative consequences in terms of error propagation. Examples of iterated approaches are recurrent neural networks (Williams & Zipser (1989)) or local learning iterated techniques (Farmer & Sidorowich (1987); McNames (1998)). Another way to perform *H-step-ahead* forecasting consists in estimating a set of H prediction models each returning a direct forecast at time $t + k$, $k = 1, \ldots, H$ (Sorjamaa, Hao, Reyhani, Ji, & Lendasse (2007)). Direct methods often require higher functional complexity (Tong (1983)) than iterated ones in order to model the stochastic dependency between two series values at two distant instants (Guo et al. (1999)). The debate about which of the two methods is superior is still open. Results from Zhang & Hutchinson (1994); Sorjamaa et al. (2007) support the idea that direct prediction is better than the iterated method. However, other research works (Weigend, Huberman, & Rumelhart (1992); Bontempi, Birattari, & Bersini (1999b)) provide experimental evidence in favour of iterated methods. In computational intelligence literature we have also examples of research works where the two approaches have been successfully combined (Sauer (1994),Zhang & Hutchinson (1994)). In spite of their diversity, iterated and direct techniques for multiple-step-ahead prediction share a common feature: they model from data a multi-input single-output mapping whose output is the variable φ^{t+1} in the iterated case and the variable φ^{t+k} in the direct case, respectively. This paper advocates that when a very long term prediction is at stake and a stochastic setting is assumed, the modeling of a single-output mapping neglects the existence of stochastic dependencies between future values, (e.g. between φ^{t+k} and φ^{t+k+1}) and consequently biases the prediction accuracy. A possible way to remedy to this shortcoming is to move from the modeling of single-output mappings to the modeling of multi-output dependencies. This requires the adoption of multi-output techniques where the predicted value is no more a scalar quantity but a vector of future values of the time series. However, the simplest way to deal with a multi-output task consists in transforming it into a set of independent problems, one per output. This is typically the case of the direct strategy which can be ineffective once the outputs noise are correlated like in time series.

The contribution of the paper is an extension of the local modeling approach (Fan & Gijbels (1996)) to the multi-output setting which aims to preserve in the prediction the conditional dependence (e.g. correlation) between outputs. The use of single-output local modelling (aka nearest neigbour or smoothing techniques (Härdle & Vieu (1992))) in time series is not new (for a statistical review see Gao (2007)). Auestad & Tjostheim (1990) and Härdle & Vieu (1992) proposed the use of the ordinary Nadaraya-Watson estimator to estimate the direct predictor. The effectiveness of local methods in the computational intelligence community became evident during the Santa Fe forecasting competition when the method proposed by Sauer (1994) showed very good performance and returned the second best prediction for the A data set. This success was confirmed in 1999 by the KULeuven competition when the first two entries were taken by local learning techniques (McNames, Suykens, & Vandewalle (1999); Bontempi, Birattari, & Bersini (1998)).

It is well-known (Atkeson, Moore, & Schaal (1997)) that the adoption of a local approach requires the choice of a set of model parameters (e.g. the number k of neighbors, the kernel function, the distance metric). In this paper we will limit to consider the selection of the number of neighbors (also known as bandwidth selection in nonparametric statistics) which is typically considered to be the most important task to address the bias/variance trade-off. In the literature, bandwith selection is perfomed by rule-of-thumb strategies (Fan & Gijbels (1995)), plug-in techniques (Ruppert, Sheather, & Wand (1995)) or cross-validation methods (Atkeson et al. (1997)).

This paper presents and compares a set of original criteria for bandwidth selection in multi-output local predictors. The first criterion is a multi-output extension of the constant leave-one-out approach (Bontempi, Birattari, & Bersini (1999a)). Such extension has been first introduced in (Bontempi (2008); Ben Taieb, Bontempi, Sorjamaa, & Lendasse (2009)) and discussed in a recent paper by Ben Taieb, Sorjamaa, & Bontempi (2010). Here we intend to go a step forward with respect to the existing work by comparing this criterion with other selection strategies. In particular we will focus on criteria which do not rely on cross-validation but take advantage of the multiple and sequential nature of the predicted output. The rationale of these criteria consists in assessing different neighborhoods and the associated forecasts by comparing the stochastic properties of the returned prediction with the ones of the original series. Two assessment measures, a linear and a nonlinear ones, are used: the first compares the autocorrelation properties of the prediction and of the historical series, the second one measures the log-likelihood of the predicted sequence on the basis of a one-step-ahead model fitted to the training series.

The resulting methods are intensively assessed against their direct and iterated counterparts on the 111 long term prediction tasks provided by the NN5 competition[1] (Crone (2008)). Note that the aim of the experimental session is not to make an exhaustive comparison of the multi-output LL technique with other state-of-the-art forecasting techniques but rather to show how for a given learning strategy (namely local modeling) the introduction of the multi-output setting can sensibly improve the performance in long term prediction.

2. Multi-step-ahead and multi-output models

Let us consider a stochastic time-series described by the Nonlinear Auto Regressive (NAR) model (Fan & Yao (2005))

$$\varphi^{t+1} = f\left(\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1}\right) + w(t+1) = f(X) + w(t+1) \tag{1}$$

where m (*dimension*) is the number of past values taken into consideration, d is the lag time, w is a zero-mean noise term and

$$X = \{\varphi^{t-d}, \varphi^{t-d-1}, \dots, \varphi^{t-d-m+1}\}$$

denotes the lag vector. Suppose we have measured the series up to time t and that we intend to forecast the next H, $H \geq 1$, values. The problem of predicting the next H values boils down to the estimation of the distribution of the H dimensional random vector

$$Y = \{Y^1, \dots, Y^H\} = \{\varphi^{t+1}, \dots, \varphi^{t+H}\}$$

[1] http://www.neural-forecasting-competition.com

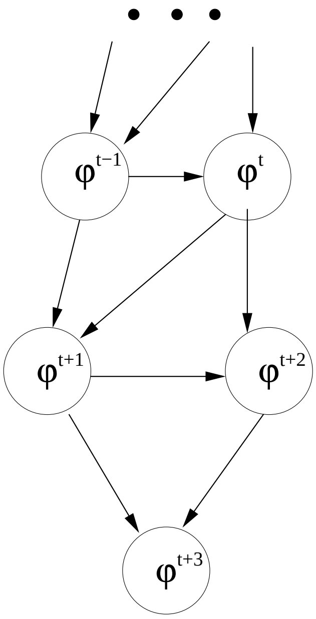

Figure 1: Graphical modeling representation of the conditional density $p(Y|X)$ for $H = 3$, $m = 2$, $d = 0$.

conditional on the value of X. In other terms, the stochastic dependency (1) between a future value $\varphi^{t+h}, h > 0$ of the time series and the past observed values X induces the existence of a multivariate conditional probability density $p(Y|X)$ where $Y \in \mathbb{R}^H$ and $X \in \mathbb{R}^m$. This distribution can be highly complex in the case of a large dimensionality m of the series and a long term prediction horizon H (Verleysen & Francois (2005)). An easy way to visualize and reason about this complex conditional density is to use a probabilistic graphical model approach. Probabilistic graphical models (Jensen (2001)) are graphs in which nodes represent random variables, and the lack of arcs represent conditional independence assumptions. For instance the probabilistic dependencies which characterize a multiple-step-ahead prediction problem for a time series of dimension $m = 2$, lag time $d = 0$ and horizon $H = 3$ can be represented by the graphical model in Figure 1. Note that in this figure, $X = \{\varphi^t, \varphi^{t-1}\}$ and $Y = \{\varphi^{t+1}, \varphi^{t+2}, \varphi^{t+3}\}$. This graph shows that φ^{t-1} has a direct influence on φ^{t+1} but only an indirect influence on φ^{t+2}. At the same time φ^{t+1} and φ^{t+3} are not conditionally independent given the vector $X = \{\varphi^t, \varphi^{t-1}\}$.

Any forecasting method which aims to perform multiple-step ahead prediction implements (often in an implicit manner) an estimator of the highly multivariate conditional density $p(Y|X)$. The graphical model representation can help us in visualizing the differences between the two most common multiple-step-ahead approaches, the iterated and the direct one. The iterated prediction approach replaces the unknown random variables $\{\varphi^{t+1}, \dots, \varphi^{t+H-1}\}$ with their estimations $\{\hat{\varphi}^{t+1}, \dots, \hat{\varphi}^{t+H-1}\}$. In graphical terms this method models an approximation (Figure 2) of the real conditional density where the topology of conditional dependencies is preserved though non observable variables are replaced by their noisy estimators.

The direct prediction approach transforms the problem of modeling the multivariate distribution $p(Y|X)$ into H distinct and parallel problems where the target conditional density is $p(\varphi^{t+h}|X)$, $h = 1, \dots, H$. The topology of the dependencies of the original condition density is then altered as shown in Figure 3. Note that in graphical model terminology this is equivalent to make a *conditional independence assumption*

$$p(Y|X) = p(\{\varphi^{t+1}, \dots, \varphi^{t+H}\}|X) = \prod_{h=1}^{H} p(\varphi^{t+h}|X)$$

Such assumption is well known in the machine learning literature since it is exploited by the Naive Bayes classifier to simplify multivariate classification problems. Figures 2 and 3 visualize the disadvantages associated to the adoption of the iterated and the direct method, respectively. Iterated methods may suffer of low performance in long horizon tasks. This is due to the fact that they are essentially models tuned with a one-step-ahead criterion and therefore they are not able to take temporal behavior into account. In terms of bias/variance decomposition we can say that the iterated approach returns a non biased estimator of the conditional density $p(Y|X)$ since it preserves the dependencies between the components of the vector Y though it suffers of high variance because of the propagation and amplification of the prediction error.

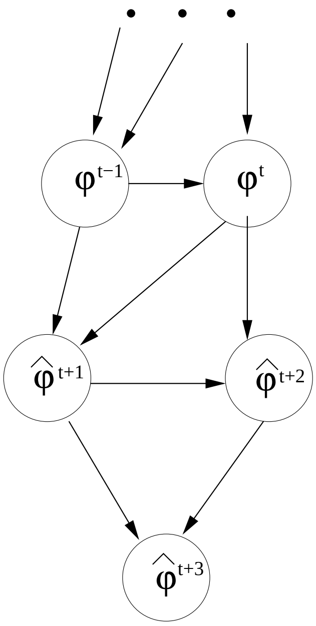
φ^{t-1}
φ^{t}
$\widehat{\varphi}^{t+1}$
$\widehat{\varphi}^{t+2}$
$\widehat{\varphi}^{t+3}$

Figure 2: Graphical modeling representation of the distribution modeled by the iterated approach in the $H = 3$, $m = 2$, $d = 0$ prediction problem.

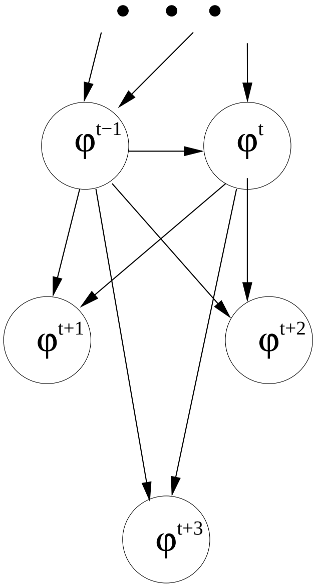

Figure 3: Graphical modeling representation of the distribution modeled by the direct approach in the $H = 3$, $m = 2$, $d = 0$ prediction problem.

On the other side, direct methods, by making an assumption of conditional independence, neglect complex dependency patterns existing between the variables in Y and consequently return a biased estimator of the multivariate distribution $p(Y|X)$.

In order to overcome these shortcomings, this paper proposes a multi-input multi-output approach where the modeling procedure does not target any more single-output mappings (like $\varphi^{t+1} = f(X) + w$ or $\varphi^{t+k} = f^k(X) + w$) but the multi-output mapping

$$Y = F(X) + W$$

where $F : \mathbb{R}^m \rightarrow \mathbb{R}^H$ and the covariance of the noise vector W is not necessarily diagonal (Matias (2005)). The multi-output model is expected to return a multivariate estimation of the joint distribution $p(Y|X)$ and, by taking into account the dependencies between the components of Y, to reduce the bias of the direct estimator. For multi-output prediction problems the availability of learning algorithms is much more reduced than in the single-output case (Matias (2005)). Most of existing approaches propose what is actually done by the direct approach, that is to decompose the problem into several multi-input single-output problems by making the assumption of conditional independence. What we propose here is to remove this assumption by using a multivariate estimation of the conditional density. For this purpose we adopt a nearest neighbor estimation approach where the problem of adjusting the size of the neighborhood (bandwidth) is either solved by extending a strategy successfully adopted in our previous work on Lazy Learning (Bontempi et al. (1999a); Birattari, Bontempi, & Bersini (1999)) or tackled by using two novel selection criteria.

3. A locally constant method for multi-output regression

We discuss here a locally constant multi-output regression method to implement a multiple-step-ahead predictor. The idea is to return, instead of a scalar, a vector that smoothes the continuation of the trajectories which at time t resemble the most to the trajectory X. This method is a multi-output extension of the Lazy Learning (LL) algorithm (Bontempi et al. (1999a); Birattari et al. (1999)) and is referred to as LL-MIMO. Lazy Learning is an instance of local learning (Atkeson et al. (1997)), a computational intelligence method which performs predictions by fitting simple local models in the neighborhood of the point to be predicted. The local learning procedure requires to keep in memory the entire dataset and, each time a prediction is required for a given input (also called query point), it performs three basic operations: it ranks the samples according to the distance to the query, it selects a number of nearest neighbors and eventually fits a local model. The Lazy Learning algorithm we consider here relies on the PRESS statistic, a convenient way of assessing the quality of a local model by linear cross-validation. The PRESS statistic can be easily computed as a by-product of a constant or linear fitting and enables a fast yet accurate selection of the number of neighbors.

In order to apply local learning to time series forecasting, the time series is embedded into a dataset D_N made of N pairs (X_i, Y_i), where X_i is a temporal pattern of length m, and the vector Y_i is the consecutive temporal pattern of length H. Suppose the series is measured up to time t and assume for simplicity that the lag $d = 0$. Let us denote

$$\bar{X} = \{\varphi^t, \ldots, \varphi^{t-m+1}\}$$

the query point, i.e. the lag embedding vector at time t. Given a metric on the space $\mathbb{R}^m$ let us order increasingly the set of vectors X_i with respect to the distance to $\bar{X}$ and denote by $[j]$ the index of the jth closest neighbor of $\bar{X}$. For a given number k of neighbors the H-step prediction is a vector whose hth component is the average

$$\hat{Y}_k^h = \frac{1}{k} \sum_{j=1}^{k} Y_{[j]}^h,$$

where $Y_{[j]}$ is the output vector of the jth closest neighbor of $\bar{X}$ in the training set D_N.

The adoption of a local approach to solve a prediction task requires the definition of a set of model parameters (e.g. the number k of neighbors, the kernel function, the parametric family, the distance metric). In local learning literature different methods exist to automatically select the adequate configuration (Atkeson et al. (1997); Birattari et al. (1999)) by adopting tools and techniques from the field of linear statistical analysis. In the following two sections we will present two criteria to assess and compare local models with different number of neighbors: a multi-output extension of the local leave-one-out and a measure of discrepancy between the training series and the forecasted sequence which rely either on linear or nonlinear measures. The third subsection presents an averaging strategy which aims to cope with the large dimensionality of the prediction due to the large horizon H.

3.1. A multi-output version of the PRESS statistic

PRESS statistic, proposed by Allen (1974), is a simple, well-founded and computationally efficient way to perform *leave-one-out* (l-o-o) cross-validation and to assess the performance in generalization of local linear models. By assessing the performance of each local model, alternative configurations can be tested and compared in order to select the best one in terms of expected prediction. The idea consists in associating to the estimation $\hat{Y}_k$ returned by a number k of neighbors a multiple-step leave-one-out error

$$E_k = \frac{1}{H}\sum_{h=1}^{H}\left(e^h\right)^2,$$

where e^h is the leave-one-out error of a constant model used to approximate the output at the h-th step. In case of constant model the l-o-o term is easy to derive (Bontempi (1999))

$$e^h = \sum_{j=1}^{k} e_j^h, \quad e_j^h = Y_{[j]}^h - \frac{\sum_{i \neq j} Y_{[i]}^h}{k-1} = k\frac{Y_{[j]}^h - \hat{Y}_k^h}{k-1}.$$

The optimal number of neighbors can be then defined as the number

$$k^* = \arg\min E_k, \tag{2}$$

which minimizes the leave-one-out error. Note that unlike a direct approach based on LL, the number of neighbors is the same for all the horizons $h = 1, \ldots, H$. In the following we will define as M-CV the local modeling multi-output forecaster whose number of neighbors is selected according to (2).

3.2. A selection criterion based on stochastic discrepancy

This section presents a selection criterion which is not based on leave-one-out (or similar training and test strategies) but rather on a measure of stochastic discrepancy between the forecasted sequence

$$\hat{Y}_k = \{\hat{Y}_k^1, \hat{Y}_k^2, \ldots, \hat{Y}_k^H\} = \{\hat{\varphi}^{t+1}, \hat{\varphi}^{t+2}, \ldots, \hat{\varphi}^{t+H}\}$$

and the training time series

$$\varphi = \{\varphi^1, \ldots, \varphi^t\}.$$

The rationale is that, since the prediction horizon H is supposed to be large, we have enough samples to estimate reliably some descriptive statistics (e.g. autocorrelation or partial correlation in a linear case (Anderson (1971))) of the forecasted series $\hat{Y}_k$ and compare them with the analogous quantities of the training series. The lower will be the discrepancy between the descriptors of the forecasting and the descriptors of the training series φ, the better will be the quality of the returned forecast $\hat{Y}_k$. As for (2), once a discrepancy measure Δ_k is associated to the forecast made with k neighbors, we can define a selection criterion

$$k^* = \arg\min \Delta_k, \tag{3}$$

which aims to find the number of neighbors k for which the forecasted sequence has the closest stochastic properties with respect to the training series.

Two kind of discrepancy measures are proposed here:

1. a linear discrepancy measure based on linear autocorrelation and partial correlation

   $$\Delta_k^{(1)} = \|\rho(\hat{Y}_k) - \rho(\varphi)\|_2 + \|\pi(\hat{Y}_k) - \pi(\varphi)\|_2$$

   where $\| \cdot \|_2$ is an Euclidean norm, $\rho(\hat{Y}_k)$ $(\pi(\hat{Y}_k))$ is the autocorrelation (partial correlation)vector of size m of the predicted sequence $\hat{Y}_k$ and $\rho(\varphi)$ $(\pi(\varphi))$ is the autocorrelation (partial correlation) vector of the series φ.

2. a nonlinear discrepancy measure

   $$\Delta_k^{(2)} = -\sum_{h=1}^{H} \log p(\hat{Y}_k^h | \varphi) \tag{4}$$

   where $p(\hat{Y}_k^h|\varphi)$ is the likelihood of observing the value $\hat{Y}_k^h$ given the stochastic properties of the training set φ. A way to approximate $\Delta_k^{(2)}$ is to have recourse to a model of the training set, for instance a one-step-ahead-predictor, and use it to measure the likelihood of the forecast $\hat{Y}_k$. Once a Gaussian hypothesis is made, we have then that minimizing $\Delta_k^{(2)}$ boils down to minimize the quantity

   $$\Delta_k^{(2)} \approx \sum_{h=m}^{H} (\hat{Y}_k^h - \hat{y}_k^h)^2$$

   where $\hat{y}_k^h$ is the prediction returned by a one-step-ahead predictor trained with the series φ when the lag vector is $\{\hat{Y}_k^{h-1}, \ldots, \hat{Y}_k^{h-m+1}\}$.

Note that the second discrepancy measure represents a nonlinear version of the first measure since it uses a nonlinear model (in our case again a local model) to compare the stochastic properties of the training set and forecast vector. In the following we will denote by M-D1 the local modeling multi-output forecaster whose number of neighbors is selected according to (3) when $\Delta = \Delta^{(1)}$ and by M-D2 the forecaster when $\Delta = \Delta^{(2)}$, respectively.

3.3. Averaging strategy

The major difficulty of multiple-step-ahead forecasting is related to the large dimensionality of Y which makes the multivariate estimation vulnerable to large variance effects (Verleysen & Francois (2005)). A possible countermeasure to such a side effect is the adoption of combination strategies, which are well reputed to reduce variance in case of low bias estimators. The idea of combining predictors is well known in the time series literature (Sauer (1994)). What is original here is that a multi-output approach allows the availability of a large number of estimators once the prediction horizon H is long. Think for example to the case where $H = 20$ and we want to estimate the value φ^{t+10}. A simple way to make such estimate more robust and accurate is to compute and combine several long term estimators which have an horizon larger than 10 (e.g. all the predictors with horizon between 10 and 20).

What we propose is an averaging strategy such that the prediction at time $t + h$ is given by

$$\hat{\varphi}^{t+h} = \frac{\sum_{j=h}^{H} \hat{Y}_{(j)}^{h}}{H - h + 1},$$

where in this case the notation $\hat{Y}_{(j)}^{h}$ is used to denote the h^{th} term of the vector prediction returned by a LL-MIMO trained for an horizon $j \geq h$. Note that this averaging strategy demands the training of a number of different multiple predictors $\hat{Y}_{(j)}, j = 1, \ldots, H$ with different horizons.

4. Experiments and final considerations

The experimental session aims to assess the improvement that a multi-output strategy brings to a forecasting technique based on local learning (e.g. a local constant model). In order to setup a consistent and reliable benchmark, we took inspiration from the out-of-sample experimental strategy proposed in Tashman (2000), by considering a large number of time series and using a rolling-origin evaluation. In particular, the experimental session uses the 111 time series of the *NN5 Competition* (complete dataset). These time series have all the same length and contain the daily retirement amounts from independent cash machines at different, randomly selected locations within England. They show different patterns of single or multiple overlying seasonality, including day of the week effects, week or month in the year effects, calendar effects. In our forecasting experiments, we adopt a sliding prediction window with three different horizons $H = 20$, $H = 50$ and $H = 90$.

Note that the goal of the experimental session is not to benchmark the LL-MIMO techniques against the best performing approaches in NN5 (Andrawis & Atiya (2010)) but to provide a number of experimental conditions to compare five multiple-step-ahead strategies: (i) a conventional iterated approach denoted by IT, (ii) a conventional direct approach denoted by DIR, (iii) a multi-output M-CV approach, (iii) a multi-output M-CV-C approach with averaging (Section 3.3), (iv) a multi-output M-D1-C approach with averaging and (v) a multi-output M-D2-C approach with averaging.

In all the considered techniques the learner is implemented by the same local learning technique whose number of neighbors range in the same interval $[5, k_b]$ with k_b parameter of the algorithm. In order to perform a statistically sound comparison, all the techniques are tested under the same conditions in terms of test intervals, embedding order m, values of k_b and lag time d. Since the goal is not to compete with the winning approaches of the NN5 competition but rather to increase the statistical power of the comparison, neither input selection nor seasonality preprocessing was carried out. In detail 3 test intervals are considered with starting points at $t = 489$, $t = 562$ and $t = 635$. Also the values of m are allowed to range in $\{2, 5, 8, 11, 14, 17, 20\}$, the values of d in $\{0, 1\}$ and the values of k_b in $\{15, 20, 25\}$. Note that, if H is the horizon and t is the time origin, the task consists in predicting the continuation of the series $\{\varphi^t, \dots, \varphi^{t+H-1}\}$ with a model learned on the basis of $\{\varphi^1, \dots, \varphi^{t-1}\}$.

Table 1 compares the average Relative Absolute Error (RAE) of the five techniques for the 111 NN5 datasets. The RAE measure was first introduced in Armstrong & Collopy (1992) and measures the ratio of the absolute prediction error to the analogous absolute error from a random walk. Table 2 compares the average Symmetric Mean Absolute Percentage Error (SMAPE) of the five techniques for the 111 NN5 datasets. The SMAPE measure was the main accuracy measure considered in the NN5 competition (Crone (2008)).

Horizon	**IT**	**DIR**	**M-CV**	**M-CV-C**	**M-D1-C**	**M-D2-C**
H=20	0.829	0.693	0.698	0.681	0.685	0.680
H=50	0.923	0.675	0.680	0.656	0.661	0.654
H=90	0.933	0.694	0.699	0.674	0.680	0.673

Table 1: Average RAE for different horizons.

Horizon	**IT**	**DIR**	**M-CV**	**M-CV-C**	**M-D1-C**	**M-D2-C**
H=20	0.435	0.373	0.376	0.369	0.371	0.367
H=50	0.471	0.358	0.360	0.353	0.355	0.351
H=90	0.467	0.347	0.350	0.338	0.341	0.337

Table 2: Average SMAPE for different horizons.

The average results of Table 1 and Table 2 consistently support two main considerations. First, MIMO strategies largely outperform the conventional iterated method as shown for $H = 50$ by RAE=0.923 (IT) and RAE=0.680 (M-CV). Second, all the averaging versions of the MIMO strategies (RAE= $0.656, 0.661$ and 0.654 for M-CV-C, M-D1-C and M-D2-C, respectively) outperform the direct approach (RAE=0.675). Note that similar accuracy patterns are encountered when we restrict to consider specific ranges of values for the hyperparameters. This is the case of Table 3 where the RAE results concern small embedding orders, of Table 4 where the neighborhhod is kept fixed and of Table 5 where only the zero lag is taken into consideration.

Horizon	**IT**	**DIR**	**M-CV**	**M-CV-C**	**M-D1-C**	**M-D2-C**
H=20	0.897	0.719	0.723	0.698	0.702	0.698
H=50	0.956	0.698	0.704	0.673	0.678	0.672
H=90	0.964	0.715	0.720	0.691	0.697	0.690

Table 3: Small embedding orders ($m = 2, 5, 8, 11$): average RAE for different horizons.

Horizon	**IT**	**DIR**	**M-CV**	**M-CV-C**	**M-D1-C**	**M-D2-C**
H=20	0.833	0.698	0.703	0.684	0.687	0.682
H=50	0.927	0.680	0.685	0.657	0.661	0.655
H=90	0.937	0.700	0.704	0.677	0.681	0.675

Table 4: Small neighborhood ($k_b = 15$): average RAE for different horizons.

These considerations are endorsed by the additional analysis based on paired permutation tests between pairs of approaches for each dataset. Table 6 summarizes the paired significativity tests

Horizon	**IT**	**DIR**	**M-CV**	**M-CV-C**	**M-D1-C**	**M-D2-C**
H=20	0.735	0.693	0.697	0.681	0.686	0.680
H=50	0.734	0.676	0.681	0.656	0.661	0.654
H=90	0.757	0.696	0.701	0.676	0.681	0.674

Table 5: Zero Lag ($d = 0$): average RAE for different horizons.

by showing how many times the proposed LL-MIMO techniques are significantly (pval < 0.05) better/worse than the conventional techniques. Note that the p-values were corrected by the Bonferroni test in order to account for multiple testing.

$H = 20$	**IT**	**DIR**
M-CV	107/0	6/60
M-CV-C	108/0	55/4
M-D1-C	107/0	38/11
M-D2-C	108/0	59/4

$H = 50$	**IT**	**DIR**
M-CV	104/1	10/60
M-CV-C	105/1	62/10
M-D1-C	105/1	46/17
M-D2-C	105/1	68/6

$H = 90$	**IT**	**DIR**
M-CV	110/0	11/61
M-CV-C	109/0	62/6
M-D1-C	109/0	45/19
M-D2-C	110/0	66/6

Table 6: Number of times that the multi-output techniques (in the rows) are significantly better/worse (permutation test pval < 0.05) than the conventional multiple-step-ahead techniques (in the columns).

The last analysis focuses on the residuals of the predictions. Table 7 contains some information about the residuals of the different prediction methods for all the horizons. In particular we report

		IT	**DIR**	**M-CV**	**M-CV-C**	**M-D1-C**	**M-D2-C**
H=20	Mean residual	0.321	0.194	0.211	0.225	0.229	0.246
	Mean abs acf	0.135	0.143	0.142	0.138	0.138	0.139
H=50	Mean residual	-0.159	-0.240	-0.195	-0.165	-0.155	-0.142
	Mean abs acf	0.187	0.153	0.152	0.146	0.146	0.146
H=90	Mean residual	0.148	0.151	0.214	0.253	0.266	0.257
	Mean abs acf	0.196	0.152	0.152	0.145	0.145	0.145

Table 7: Residual analysis. For each horizon H, the first line contains the mean of the residuals and the second one the average value of the autocorrelation function of the residual series. Note that the 95% confidence interval is at $2/\sqrt{H} = 0.283$.

the mean of the residuals and the mean absolute value of the autocorrelation function of the residual series (lags ranging from 2 to 10) to test the whiteness of the residuals. Note that, also in terms of whiteness of the residuals, the previous considerations are confirmed.

In summary, the experimental results show that for long term prediction tasks the proposed MIMO strategies largely outperform conventional iterated methods. With respect to direct methods, the MIMO strategies appear to be more accurate only when the averaging version of the algorithms is taken into consideration. This is probably due to the large variance of the MIMO estimator for a small number of samples. However, its unbiased nature lets the technique take advantage of the averaging operator as shown by the fact that, in a large majority of tasks and for different horizons, the averaged multi-output approach significantly outperforms the direct approach. As far as the neighbor selection is concerned, it is interesting to remark that the criteria discussed in Section 3.2 appear to be competitive with the conventional leave-one-out approaches. In particular, the nonlinear discrepancy criterion is able to outperform the cross-validated approach.

The above results support the idea that the specific nature of the long term forecasting task should lead researchers to propose design innovative approaches which avoid the assumption of conditional independence. Conditional independence implies the adoption of a reductionist approach where prediction is typically decomposed into a set of low-level and independent tasks. Reductionist approaches (like direct forecasting) strive for accuracy at specific time points without caring for the global properties of the returned prediction. We are confident that the methodology discussed in this paper will pave the way to a more integrated and system-level philosophy in forecasting.

5. Conclusion

Multiple-step-ahead forecasting is a difficult problem yet to be solved. So far the mainstream approach consists in adapting short term prediction techniques to the long term scenario. This paper advocates the need of a major shift in the design of forecasting techniques for long term. The paper proposes two main innovative principles related to the fact that the outcome of a long term forecasting technique is not a single value but a series itself. The first novelty is the adoption of a multi-output prediction techniques which aims to learn and preserve the stochastic dependency between sequential predicted values. The second idea is that, since the prediction is a time series, global criteria at the series level can be successfully employed to design the forecaster. The proposed approach was assessed by means of a large number of testing configurations and was shown to significantly outperform the iterated and the direct approaches.

References

Allen, D. M. (1974). The relationship between variable and data augmentation and a method of prediction. *Technometrics*, *16*, 125–127.

Anderson, T. W. (1971). *The statistical analysis of time series.* J. Wiley and Sons.

Andrawis, R. R. & Atiya, A. F. (2010). Forecast combination model using computational intelligence/linear models for the NN5 tie series forecasting competition. *International Journal of Forecasting.* To appear.

Armstrong, J. & Collopy, F. (1992). Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting*, *8*, 69–80.

Atkeson, C. G., Moore, A. W., & Schaal, S. (1997). Locally weighted learning. *Artificial Intelligence Review*, *11*(1–5), 11–73.

Auestad, B. & Tjostheim, D. (1990). Identification of nonlinear time series: first order characterization and order determination. *Biometrika*, *77*, 669–687.

Ben Taieb, S., Bontempi, G., Sorjamaa, A., & Lendasse, A. (2009). Long-term prediction of time series by combining direct and mimo strategies. In: *Proceedings of the 2009 IEEE International Joint Conference on Neural Networks*, pp. 3054–3061, Atlanta, U.S.A.

Ben Taieb, S., Sorjamaa, A., & Bontempi, G. (2010). Multiple-output modelling for multi-step-ahead forecasting. *Neurocomputing*, *73*, 1950–1957.

Birattari, M., Bontempi, G., & Bersini, H. (1999). Lazy learning meets the recursive least-squares algorithm. In: Kearns, M. S., Solla, S. A., & Cohn, D. A. (eds.), *NIPS 11*, pp. 375–381, Cambridge: MIT Press.

Bontempi, G. (1999). *Local Learning Techniques for Modeling, Prediction and Control.* Ph.D. thesis, IRIDIA- Université Libre de Bruxelles.

Bontempi, G. (2008). Long term time series prediction with multi-input multi-output local learning. In: *Proceedings of the 2nd European Symposium on Time Series Prediction (TSP), ESTSP08*, pp. 145–154.

Bontempi, G., Birattari, M., & Bersini, H. (1998). Lazy learning for iterated time series prediction. In: Suykens, J. A. K. & Vandewalle, J. (eds.), *Proceedings of the International Workshop on Advanced Black-Box Techniques for Nonlinear Modeling*, pp. 62–68, Katholieke Universiteit Leuven, Belgium.

Bontempi, G., Birattari, M., & Bersini, H. (1999a). Lazy learning for modeling and control design. *International Journal of Control*, *72*(7/8), 643–658.

Bontempi, G., Birattari, M., & Bersini, H. (1999b). Local learning for iterated time-series prediction. In: Bratko, I. & Dzeroski, S. (eds.), *Machine Learning: Proceedings of the Sixteenth International Conference*, pp. 32–38, San Francisco, CA: Morgan Kaufmann Publishers.

Chen, R., Yang, L., & Hafner, C. (2004). Nonparametric multistep-ahead prediction in time series analysis. *Journal of Royal Statistical Society B*, *66*, 669–686.

Crone, S. (2008). Results of the NN5 time series forecasting competition,. In: *WCCI 2008, IEEE World Congress on Computational Intelligence.*

Fan, J. & Gijbels, I. (1995). Adaptive order polynomial fitting: bandwidth robustification and bias reduction. *J. Comp. Graph. Statist.*, *4*, 213–227.

Fan, J. & Gijbels, I. (1996). *Local Polynomial Modelling and Its Applications.* Chapman and Hall.

Fan, J. & Yao, Q. (2005). *Nonlinear Time Series.* Springer.

Farmer, J. D. & Sidorowich, J. J. (1987). Predicting chaotic time series. *Physical Review Letters*, *8*(59), 845–848.

Gao, J. (2007). *Nonlinear time series: semiparametric and nonparametric methods.* Chapman and Hall.

Guo, M., Bai, Z., & An, H. (1999). Multi-step prediction for nonlinear autoregressive models based on empirical distributions. *Statistica Sinica*, pp. 559–570.

Härdle, W. & Vieu, P. (1992). Kernel regression smoothing for time series. *Journal of Time Series Analysis*, *13*, 209–232.

Jensen, F. V. (2001). *Bayesian Networks and Decision Graphs.* Springer.

Lapedes, A. & Farber, R. (1987). Nonlinear signal processing using neural networks. Tech. Rep. LA-UR-87-2662, Los Alamos National Laboratory, Los Alamos, NM.

Matias, J. M. (2005). Multi-output nonparametric regression. In: *Progress in Artificial Intelligence*, pp. 288–292.

McNames, J. (1998). A nearest trajectory strategy for time series prediction. In: *Proceedings of the International-Workshop on Advanced Black-Box Techniques for Nonlinear Modeling*, pp. 112–128, Belgium: K.U. Leuven.

McNames, J., Suykens, J., & Vandewalle, J. (1999). Winning contribution of the K.U. Leuven time-series prediction competition. *International Journal of Bifurcation and Chaos*, pp. 1485,1501.

Pemberton, J. (1987). Exact least squares multi-step prediction from nonlinear autoregressive models. *Journal of Time Series Analysis*, *8*, 443–448.

Ruppert, D., Sheather, S. J., & Wand, M. P. (1995). An effective bandwidth selector for local least squares regression. *Journal of American Statistical Association*, *90*, 1257–1270.

Sauer, T. (1994). Time series prediction by using delay coordinate embedding. In: Weigend, A. S. & Gershenfeld, N. A. (eds.), *Time Series Prediction: forecasting the future and understanding the past*, pp. 175–193, Harlow, UK: Addison Wesley.

Sorjamaa, A., Hao, J., Reyhani, N., Ji, Y., & Lendasse, A. (2007). Methodology for long-term prediction of time series. *Neurocomputing*, *70*(16-18), 2861–2869.

Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: an analysis and review. *International Journal of Forecasting*, *16*, 437–450.

Tong, H. (1983). *Threshold models in Nonlinear Time Series Analysis.* Berlin: Springer Verlag.

Verleysen, M. & Francois, D. (2005). The curse of dimensionality in data mining and time series prediction. In: Cabestany, J., Prieto, A., & Hernandez, F. (eds.), *IWANN 2005*, vol. 3512 of *Lecture Notes in Computer Science*, Springer.

Weigend, A. S., Huberman, B. A., & Rumelhart, D. E. (1992). Predicting sunspots and exchange rates with connectionist networks. In: Casdagli, M. & Eubank, S. (eds.), *Nonlinear modeling and forecasting*, pp. 395–432, Addison-Wesley.

Williams, R. & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, *1*, 270–280.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting*, *14*, 35–62.

Zhang, X. & Hutchinson, J. (1994). Simple architectures on fast machines: practical issues in nonlinear time series prediction. In: Weigend, A. S. & Gershenfeld, N. A. (eds.), *Time Series Prediction: forecasting the future and understanding the past*, pp. 219–241, Addison Wesley.