Enhancements to a method of analogues forecasting algorithm

P.J. Moore† and M.A.Little ‡

†Mathematical Institute, Oxford University.
‡MIT media lab and Aston University.
Email: paul.moore@maths.ox.ac.uk, max.little@aston.ac.uk

Abstract—This paper presents a new forecasting algorithm for time series using the method of analogues approach. We provide some theoretical background to the method and its application to time series forecasting. Next we explain the details of the new algorithm and how it relates to existing methods. Finally we evaluate the algorithm on an electrocardiogram (ECG) time series.

1. Method of analogues forecasting

The method of analogues was originally proposed as a nonlinear forecasting method by Lorenz (1969) in the context of dynamical systems theory. The approach embeds the time series in a state space using delay co-ordinates, and learns the nonlinear function using local approximation (Farmer et al., 1987). In the context of statistical learning the method is related to k-nearest neighbour (kNN) models, which are a form of instance–based learning. They are used in supervised learning as a low bias function approximator (Hastie et al., 2009). Rather than using the training set to estimate model parameters it is searched for instances to use as predictors. In time series forecasting, the algorithm selects past sequences which are similar to the current sequence and uses their successor points to make a point prediction.

A theoretical analysis of the method from a time series perspective is given by Yakowitz (1987) and practical examples of the method's application to finance, energy and hydrology are given by Arroyo et al. (2009). Fernández-Rodríguez et al. (1999) apply nearest neighbour methods to forecasting nine european currencies and find that it outperforms a random walk model. Barkoulas et al. (2003) apply similar forecasting methods to US interest rates and find an improvement in prediction accuracy over linear benchmarks. Arora et al. (2011) provide a review in the more general context of financial forecasts.

The method has often been used in hydrology, such as the work by Karlsson et al. (1987) on rainfall runoff forecasting. Wu et al. (2010) find that the kNN method outperforms both neural network and ARMA models for forecasting river flow. Conversely, Toth et al. (2000) apply the method to flood prediction and find that neural networks outperform kNN prediction.

1.1. Simple nonlinear prediction

For a dynamical system, a set of vectors sampled at discrete times in its state space is described by,

$$\mathbf{x}_{t+1} = F(\mathbf{x}_t) \tag{1}$$

Assuming that the discrete time map $F : \mathbb{R}^m \to \mathbb{R}^m$, is continuous with respect to its argument, we can predict a future state $\mathbf{x}_{t+1}$ by searching past states for the one closest to the present state $\mathbf{x}_t$. If a state $\mathbf{x}_{t_0}$ is similar to $\mathbf{x}_t$, then the continuity assumption guarantees that $\mathbf{x}_{t_0+1}$ will be close to $\mathbf{x}_{t+1}$.

We can approximate the system state $\mathbf{x}_n$ as a reconstructed state $\mathbf{s}_n$ using Taken's delay embedding theorem (Takens et al., 1981). A clear introduction to state space reconstruction including its history is given in Casdagli et al. (1991). We rewrite the delay vector in terms of the observed variable y, as $\mathbf{s}_n = (y_{n-(m-1)\tau}, \ldots y_{n-\tau}, y_n), m \geq 1$. Here m is called the embedding dimension and τ is the embedding delay.

The m-dimensional reconstruction of the system dynamics facilitates a method of analogues forecasting method. A simple predictor takes the average of the 'successor' points to the neighbouring delay vectors formed from the time series,

$$\hat{y}_{t+1} = \frac{1}{|\mathcal{U}_\epsilon(\mathbf{s_t})|} \sum_{n:\mathbf{s}_n \in \mathcal{U}_\epsilon(\mathbf{s_t})} y_{n+1} \tag{2}$$

where y_{n+1} is the successor point to the vector $\mathbf{s}_n$, $\mathcal{U}_\epsilon$ is a neighbourhood of radius ϵ and $|\mathcal{U}_\epsilon(\mathbf{s_t})| \geq k$ where k is an adjustable parameter. Figure 1 illustrates a reconstructed state space with neighbouring vectors, and the neighbourhood boundary shown as circle.

1.2. Non-parametric regression

In the context of time series, an autoregression predictor is,

$$y_{t+1} = \mathcal{E}\left[Y_{t+1} \,|\, y = Y_{t-m+1}, \ldots, Y_t\right], \qquad (m \leq t) \tag{3}$$

Here, Y_t, $t = 0, 1, 2..$ is the tth random variable from the stationary and ergodic process $\{Y\}$. The assumption of stationarity ensures that the regression function $f(y)$ is not a function of time. The assumption of ergodicity ensures that the sample average converges to

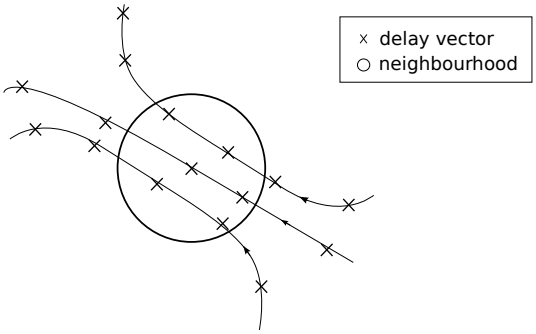

Figure 1: Representation of a reconstructed state space showing delay vectors and the neighbourhood of the current state.

the expectation, a requirement for the nearest neighbour estimator.

If we assume that $m < t$, then $\{Y\}$ is generated by a Markov chain[1] whose states are formed by delay vectors $\mathbf{s}_n = y_{n-m+1}, ..., y_n$, which have dimension m and unit time $\tau = 1$. We form a distribution of state transitions by defining the neighbourhood $\mathcal{U}_\epsilon(\mathbf{s_n})$ to give a sample of the transitions $p(s_{n+1}|\mathbf{s}_n)$. The best predictor for the Markov chain, assuming squared error loss, is approximated by the mean of this conditional distribution (Kantz et al., 2003, p261),

$$\hat{y}_{t+1} = \frac{1}{|\mathcal{U}_\epsilon(\mathbf{s_t})|} \sum_{n:\mathbf{s}_n \in \mathcal{U}_\epsilon(\mathbf{s_t})} y_{n+1} \qquad (4)$$

which is also the simple nonlinear predictor in the deterministic case, given in (2). A discussion of the relation between the stochastic and deterministic cases is given in Paparella et al. (1997). The algorithm for simple nonlinear prediction is shown as Algorithm 1.

Algorithm 1 Simple nonlinear prediction

1: Record the sequence $\mathbf{s}_t = (y_{t-\tau(m-1)} .. y_t)$
2: Using a distance measure $\mathcal{L}$ and a neighbourhood radius ϵ, find at least k sequences $\mathbf{s}_n$ close to $\mathbf{s}_t$
3: If at least k sequences are not found, expand the neighbourhood ϵ by a factor r and repeat from 2:
4: Using the candidate sequences $\mathbf{s}_n$, the next step forecast at time $t + 1$ is the average of their successor (image) points y_{n+1}.

1.3. Parameter selection

The simple nonlinear prediction method based on (2) depends on the embedding delay τ, the embedding dimension m and the neighbourhood $\mathcal{U}_\epsilon$. In the deterministic case m must be greater than twice the dimension of the system to guarantee that there will be no self-intersection in the reconstructed space, although it can be as small as the dimension (Casdagli et al., 1991, p59). The embedding delay τ can be determined from the attractor geometry if that is known: the attractor should be unfolded so that its extension in all dimensions is roughly the same. A simpler heuristic approach advocated by Kantz *et al.* (Kantz et al., 2003, p150) is to set τ to the autocorrelation length[2] of the time series. A discussion of optimal embedding parameter selection, with a criterion for selection is given in Small et al. (2004).

In the stochastic case the embedding delay is set to $\tau = 1$ in order to find the states whose transition probabilities we are to estimate. The best predictor for a Markov process of order p is formed by using delay vectors of length m where $m > p$. Kantz *et al.* (Kantz et al., 2003, p263) suggest using (4) to compute the forecast error as a function of m and choosing the value that minimises this error. One example of this approach applied to the deterministic Lorenz system is given in Meng et al. (2007).

1.4. Instance vector selection

The choice of instance vectors is determined by the neighbourhood radius ϵ and cardinality $|\mathcal{U}_\epsilon(\mathbf{s_t})| \geq k$, which is the number of instance vectors used for prediction. The value of k has to be high enough for the sample $p(s_{n+1}|\mathbf{s}_n)$ to be representative of the true distribution. The simple nonlinear prediction method expands the neighbourhood radius until at least k vectors are found. For low values of the embedding dimension (short prediction vectors) and a large training set, it should be possible to find k or more vectors. However, for a high embedding dimension or a short training set, the neighbourhood may expand to include spurious prediction vectors. This is a manifestation of the curse of dimensionality in which the training data becomes sparse as the dimension increases. The problem may also arise for individual states when the density of state vectors close to the current state is low.

Arora et al. (2011) address this problem by adjusting the neighbourhood radius according to the density of states. They propose the *fraction-Nearest Neighbor (f-NN)* method in which only the top fraction f of states are used for prediction, effectively varying the neighbourhood radius with state density. The optimum value for f and the embedding dimension m are estimated by minimising the in-sample forecast error. An evaluation on gross national product (GNP) time series shows that *f-NN* outperforms other nonlinear models such as Markov switching-AR (MS-AR) models and SETAR models.

[1] A Markov chain is a univariate Markov model in continuous space and discrete time

[2] The autocorrelation length is the time taken for the autocorrelation to decay to e^{-1}.

2. Algorithm enhancements

Here we propose a modified form of the simple nonlinear forecasting model which addresses two issues related to spurious instance vectors. The first change reduces the effect of errors from spurious vectors by using the median rather than the mean to average successor points. In (4) it is assumed that the prediction vectors $\mathbf{s_n}$ are close to the current state $\mathbf{s_t}$. However when spurious vectors are included in the neighbourhood, this assumption is violated. Under these circumstances the mean of the successor points y_{n+1} may not be the best predictor of the next state. We mitigate the effect of a minority of spurious vectors by using the median, so that (4) becomes,

$$\hat{y}_{t+1} = \underset{n:\mathbf{s}_n \in \mathcal{U}_\epsilon(\mathbf{s_t})}{\text{median}} \{y_{n+1}\} \tag{5}$$

The second change attempts to reduce the number of spurious vectors that are selected by modifying the forecasting algorithm. For time series derived from deterministic systems where the system dynamics are fixed and known, the values for the embedding delay τ and embedding dimension m can be estimated. In this case, fixing m and using the minimum neighbourhood radius to select prediction vectors from the training set provides the closest approximation to the present state. However, when the optimal embedding dimension is unknown, setting it to an arbitrary value and then expanding the neighbourhood is likely to select spurious vectors for prediction.

A further problem arises in predicting time series with mixed dynamics in which case a single value of embedding dimension may not be appropriate. The PPMD algorithm addresses these issues by using a fixed neighbourhood size and progressively decrementing the embedding dimension to search for instance vectors.

2.1. Algorithm

The PPMD algorithm searches the training set for sequences which resemble the sequence of length m preceding the point y_{t+1} as shown in Figure 2. If no sequences are found, rather than increasing the neighbourhood size, it reduces m and repeats until enough vectors are found. So when no instances of the sequence $y_{t-m+1}, \ldots y_t$ are found it uses the next shortest sequence $y_{t-m}, \ldots y_t$.

The algorithm is given in Algorithm 2. It differs from simple nonlinear forecasting, Algorithm 1, in the search method in step 3: and the use of the median in step 4: to derive the forecast from the distribution of successor points.

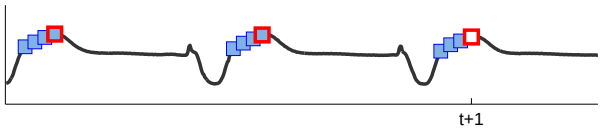

Figure 2: K-nearest neighbour forecasting for time series. The training set is searched for sequences that are similar to the candidate sequence comprising the three filled squares preceding $t+1$. The successor points (shown as blue with red borders) following the instance sequences are then used to estimate the next step forecast (white with red border).

Algorithm 2 PPMD

1: Record the sequence $\mathbf{s}_t = (y_{t-m+1} .. y_t)$
2: Using a distance measure $\mathcal{L}$ and a neighbourhood radius ϵ, find at least k sequences $\mathbf{s}_n$ close to $\mathbf{s}_t$
3: If at least k sequences are not found, set $m = m-1$ and repeat from 2:
4: Using the candidate sequences $\mathbf{s}_n$, the next step forecast at time $t+1$ is the median of the successor (image) points y_{n+1}.

2.2. Evaluation

We compare the performance of the PPMD algorithm with the simple nonlinear forecasting method implemented in the *TISEAN* software (Hegger et al., 1999). We apply both methods to the `qtdb/sel102` ECG data from Physiobank (Goldberger et al., 2000). A time plot of the final segment from the training set is shown in Figure 3.

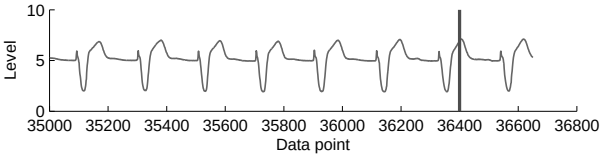

Figure 3: A segment from the `qtdb/sel102` ECG time series used for PPMD evaluation. The end of the training set at 36400 is marked by a vertical line. The subsequent 250 points are used as a validation set.

We optimise both the simple nonlinear (SNL) and PPMD forecasting methods using a grid search over the parameter space. For the SNL method the autocorrelation length is used to guide the choice of τ and we examine validation errors for $\tau = 1, 3, 10, 20$ and chose the value for which the RMS errors are lowest.

In the SNL case the optimum values are found to be $m = 9, \tau = 20$ and $k = 10$. The findings can be compared with those in (Kantz et al., 2003, p271) who report parameter values of $m = 8, \tau = 5$ and $k = 10$ for predicting 250ms of a different ECG time series

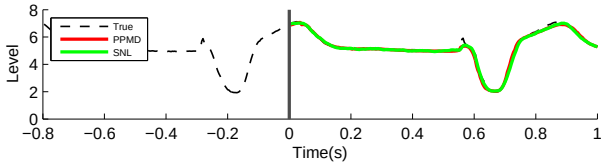

Figure 4: PPMD forecast method applied to the ECG validation set. The plot shows PPMD forecast performance compared with simple nonlinear (SNL) forecasts.

using simple nonlinear prediction. Time plots of the validation set prediction using the optimum parameters for the two methods are shown in Figure 4. A more detailed study of ECG prediction, along with a discussion of deterministic and stochastic aspects of the ECG is given in Kantz et al. (1998).

2.2.1. Out-of-sample results

Using the parameter values found from the validation set, we make predictions on separate test sets, starting at the point 40,000 and making forecasts of lengths 200, 500 and 1000. The test is repeated to make 5 different trials on different segments of the time series for each of the three test lengths. For a test length of 200 the test set starts at points 40,000, 40,200, 40,400, 40,600 and 40,800. For a test set length of 500, the test set starts at points 40,000, 40,500 and so on.

		Trial						
Length		*1*	*2*	*3*	*4*	*5*	*Mn*	*Median*
200	SNL	0.68	0.56	0.14	0.13	1.02	0.51	0.56
	PPMD	0.76	0.35	0.55	0.34	0.89	0.58	0.55
500	SNL	1.06	0.35	0.55	1.15	0.34	0.69	0.55
	PPMD	0.98	0.50	0.25	0.36	0.35	0.49	0.36
1000	SNL	1.32	0.71	0.80	1.37	0.93	1.02	0.93
	PPMD	1.19	0.67	0.80	1.24	0.76	0.93	0.80

Table 1: Out-of-sample errors for ECG data. The table shows the root mean square error for simple nonlinear and PPMD forecasts on ECG data. Trial predictions are made starting at five different points in the time series, which for a test set length of 200 are 40,000, 40,200 etc.

The results are shown in Table 1. For the five test sets of length 200, PPMD is comparable in accuracy to simple nonlinear forecasts. However, for test lengths of 500 and 1000, PPMD is slightly more accurate. The better performance of simple nonlinear forecasting for short time horizons is likely to be the result of using a smaller neighbourhood and so obtaining better resolution of time series features.

3. Conclusion

This paper has introduced a nearest neighbour forecasting algorithm (PPMD) which searches for instance vectors using a fixed neighbourhood radius and a varying instance vector length. Instead of using the mean of the image points as a predictor, it uses the median. We compared its forecasting performance with simple nonlinear forecasting using an ECG time series and found that the modified algorithm performed well in comparison with standard method of analogues forecasting.

References

• Arora S., Little M. A., McSharry P. E., 2011, Studies in Nonlinear Dynamics and Econometrics, 1 • Arroyo J., Maté C., 2009, International Journal of Forecasting, 25, 192 • Barkoulas J., Baum C., Chakraborty A., 2003, Boston College Working Papers in Economics • Casdagli M., Eubank S., Farmer J. D., Gibson J., 1991, Physica D: Nonlinear Phenomena, 51, 52 • Farmer J. D., Sidorowich J. J., 1987, Physical review letters, 59, 845 • Fernández-Rodríguez F., Sosvilla-Rivero S., Andrada-Félix J., 1999, International Journal of Forecasting, 15, 383 • Goldberger A. L. et al., 2000, Circulation, 101 • Hastie T., Tibshirani R., Friedman J., Hastie T., Friedman J., Tibshirani R., 2009. Springer • Hegger R., Kantz H., Schreiber T., 1999, Chaos: An Interdisciplinary Journal of Nonlinear Science, 9, 413 • Kantz H., Schreiber T., 1998in , IET, pp. 279–284 • Kantz H., Schreiber T., 2003, Nonlinear Time Series Analysis, 1st edn. Cambridge University Press • Karlsson M., Yakowitz S., 1987, Water Resources Research, 23, 1300 • Lorenz E. N., 1969, Journal of the Atmospheric sciences, 26, 636 • Meng Q., Peng Y., 2007, Physics Letters A, 370, 465 • Paparella F., Provenzale A., Smith L. A., Taricco C., Vio R., 1997, Physics Letters A, 235, 233 • Small M., Tse C. K., 2004, Physica D: Nonlinear Phenomena, 194, 283 • Takens F., et al., 1981, Lecture notes in mathematics, 898, 366 • Toth E., Brath A., Montanari A., 2000, Journal of Hydrology, 239, 132 • Wu C., Chau K., 2010, Engineering Applications of Artificial Intelligence, 23, 1350 • Yakowitz S., 1987, Journal of time series analysis, 8, 235