Forecasting Chaotic Time Series: A Nonparametric Approach With Applications to Synthetic and Wind Data

Cameron Bracken

Math 561 October, 2007

Abstract

In this paper a method for statistical forecasting of chaotic time series is presented. This method fits locally weighted polynomial (LWP) models to embedded time series (which are subsets of the original time series) effectively taking into account the underlying dynamics of the system (that produced the time series). The method is generally a short term approach. Successful short term forecasts were made of synthetic time series from the Lorenz and Henon systems which are known to be chaotic. The method did better at with the Lorenz System than with the Henon system. An attempt was made to apply this method to wind speed data from a buoy southeast of Eureka but for various possible reasons, the forecasts were utterly useless. Reasons are discussed for this poor predictability.

1 Introduction and Background

Forecasting extremely noisy and/or chaotic time series has always been problematic in practice and until recently, nearly impossible to do with any amount of skill. Part of the problem is distinguishing Chaos from noise in the first place. Theoretically, most natural processes could be modeled deterministically but in practice this does not usually work. Most of the time we have no idea what even the variables in the dynamic system are! Luckily, statistics come to the rescue. Though unfortunately they don't usually do well either in cases of extremely noisy/chaotic data. Figure 1 is an example of a **really** noisy/chaotic wind speed time series from Buoy 22 in Humboldt Bay (which we will actually attempt to forecast later on). Really, a combination of deterministic and statistical approaches would be ideal. This paper goes over just such a technique. The approach uses a technique called attractor reconstruction to reconstruct the phase space of the dynamic system that produced the time series and then uses a statistical model to make forecasts. According to Regonda et al. [2005], this marriage of methods will "handily outperform" most traditional (linear) methods. This paper largely parallels the approach laid out in Regonda et al. [2005] which can be considered a general reference for this entire paper.

1.1 Traditional Time Series Modeling

Traditionally time series are modeled and forecasted by fitting either a moving average (MA), autoregressive (AR), regression or multiple regression models [Chatfield 1980]. The problem with MA and AR method is that with noisy and/or chaotic data they quickly converge on the mean of the time series which isn't useful at all Regonda et al. [2005]. Also the problem with traditional regression is that it involves certain assumptions about the data (normality of residuals, homoscedasticity) that may simply be untrue in some practical applications [Grantz et al. 2005]. Nonparametric statistics are an appealing alternative to traditional methods because they make no assumptions about the distribution of the data and have the ability to capture arbitrary nonlinearities. Downsides are computation time

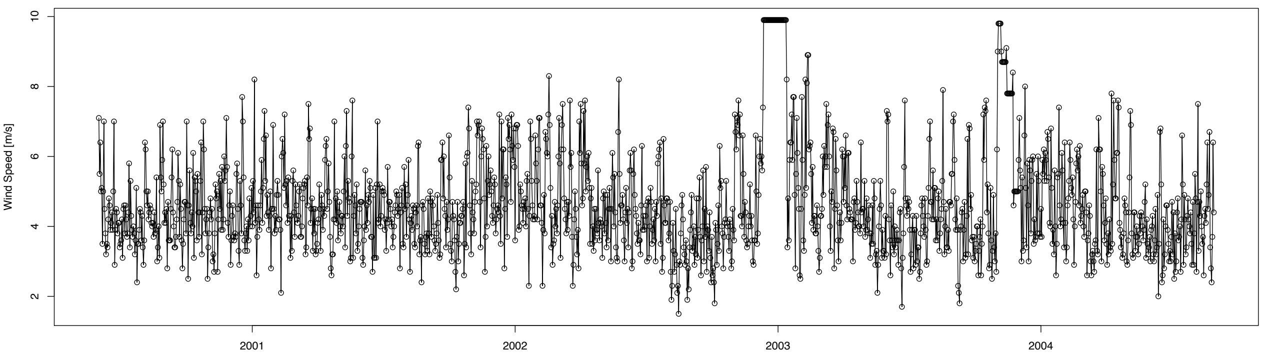


Figure 1: A really noisy and/or chaotic time series of wind data from Humboldt Bay June 2000 – August 2004

and decreased portability. To reproduce a parametric model only the parameters are needed but in nonparametric methods the entire data set is needed.

1.2 Locally Weighted Polynomial Regression

In this paper a nonparametric regression method involving locally weighted polynomials (LWP) is used as implemented in the LOCFIT package [Loader 1999]. The LWP method involves fitting a low order ($p \leq 2$) polynomial function to K nearest neighbors of the current predictor values where $K = \alpha n$, $\alpha = [0, 1)$ and n is the number of data points. The generalized cross-validation (GCV) statistic is a general measure of predictive risk. The optimal values of p and α are those which minimize the GCV statistic

$$GCV(p, \alpha) = \frac{\sum_{i=1}^{n} \frac{e_i^2}{n}}{\left(1 - \frac{m}{n}\right)^2} \tag{1}$$

where m is the number of parameters and e_i is the residual. An interesting note about LWP is if $\alpha = p = 1$ then the method just collapses to traditional linear regression.

1.3 Attractor Reconstruction or Phase Space Reconstruction

Strogatz [1994] describes the technique of attractor reconstruction as "surprising" and relates a story in which Ed Lorenz cited attractor reconstruction as the most surprising development in nonlinear dynamics. This paper was begins with the premise that such a surprising technique must have some amount of merit. By itself, attractor reconstruction is the technique of using a single time series to reconstruct the phase space of a deterministic system that produced the time series [Packard et al. 1980; Takens 1981]. The idea is that the dynamics of a time series, $\{x_1, x_2, ..., x_n\}$, are captured by or embedded in the D-dimensional phase space defined by $\mathbf{x}_t = \{x_t, x_{t+\tau}, ..., x_{t+(D-1)\tau}\}$ where t is time and τ is the mean recurrence time or delay time. Figure 2 shows the delay time for the Lorenz system. In a well behaved chaotic system, τ is the average distance between local maxima or minima and is fairly easy to pick out when the resolution is good enough. Correctly estimating the delay time is crucial for accurate attractor reconstruction. If τ is too small then the $x_{t+\tau}$'s will not be independent and if τ is too large then information about dynamics of the system will be lost Regonda et al. [2005]. For discrete time time series τ will be an integer. For continuous time series we must sample at discrete time intervals , Δt so in practice we use the index as the delay time (the actual delay time is $\tau \Delta t$).

D is related to the dimension or number of variables that are in the dynamical system which produced our original time series. If d is the dimension of the original system then $D = 2d + 1$ is sufficient to characterize the dynamics of the system according to Takens [1981] but in practice $D > d$ is adequate [Regonda et al. 2005]. In this method, the dimension of the dynamical system is reduced to a parameter of a statistical model! The method as a whole is described in the next section.

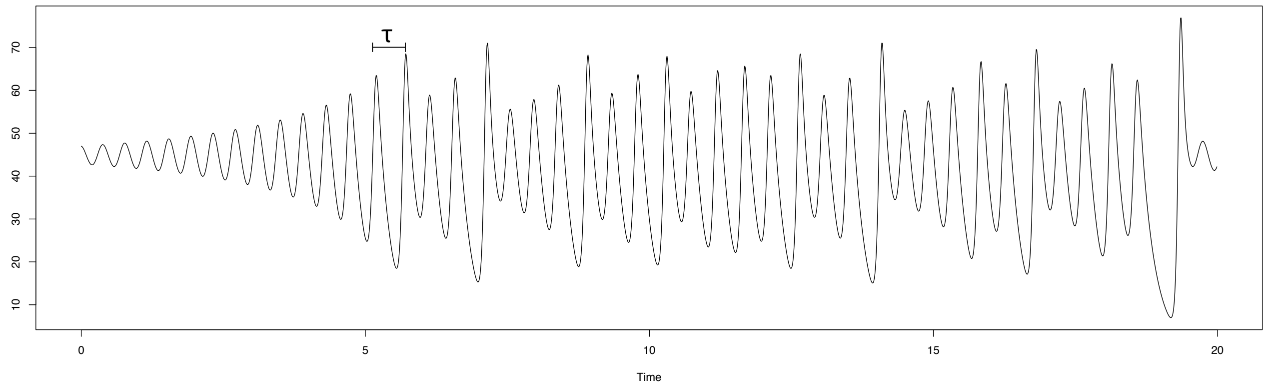


Figure 2: The delay time shown in the z coordinate of the Lorenz system with σ=16, r=45.92, b=4, Δt=0.01 and an the initial condition (-14, -13, 47).

2 Methodology

This section describes the steps necessary to forecast time series using the two pieces described above.

The first step is to reconstruct the phase space. This is done by choosing a τ and a D and then gathering the corresponding embedded time series. In practice there are actually a large number of embedded time series that depend on how much data we have before the time we wish to forecast. If we want to forecast from time I onward then there are n embedded time series where $n = I - 1 - (D-1)\tau - 1$. The -1's have been separated out to show explicitly that the index moves -1 for two separate reasons (the reason is apparent later). We gather all the embedded time series into a matrix P where

$$P = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{I-1-(D-1)\tau-1} \end{bmatrix} = \begin{bmatrix} x_1 & x_{1+\tau} & \cdots & x_{1+(D-1)\tau} \\ x_2 & x_{2+\tau} & \cdots & x_{2+(D-1)\tau} \\ \vdots & \vdots & \vdots & \vdots \\ x_{I-2-(D-1)\tau} & x_{I-2-(D-1)\tau+\tau} & \cdots & x_{I-2-(D-1)\tau+(D-1)\tau} \end{bmatrix}.$$

If you look carefully the bottom rightmost element in P simplifies to x_{I-2}. Generally each i, j entry of P is given by

$$P_{i,j+1} = x_{i+j\tau}$$

where $i \in \{1, 2, ..., I-1-(D-1)\tau-1\}$ and $j \in \{0, 1, ..., D-1\}$. In addition to recording the conditions in the phase space we record the next successive embedded series entry and say that is what happened next in the phase space given the corresponding embedded time series in P. $x_{j+(D-1)\tau+1}$ is collected for all the embedded series and placed in a vector $\mathbf{s}$ where

$$\mathbf{s} = \begin{bmatrix} x_{1+(D-1)\tau+1} \\ x_{2+(D-1)\tau+1} \\ \vdots \\ x_{I-2-(D-1)\tau+(D-1)\tau+1} \end{bmatrix}$$

Here the last entry in s simplifies to $I - 1$ which is the last time before we start the forecast! For clarification the dimensions of P are $n \times D$ and the dimensions of $\mathbf{s}$ are $n \times 1$.

For even further clarification here is a simple numeric example. If we want to start forecasting at I=1000 and we have data starting from t=1 with $\Delta t = 1$ and we are trying out $D = 3$ and $\tau = 5$. There are $n = I - 2 - (D-1)\tau = 1000 - 2 - (3-1) \cdot 5 = 988$ embedded time series! We can seen now that computation time may start to become an issue as we try to include more previous data.

P and $\mathbf{s}$ are do not have a time subscript because once they are calculated for one combination of D and τ they do not change. P and $\mathbf{s}$ will be different for any combination of D and τ of which there are quite a lot. The statistics come in in the way we choose the "optimal" values of D and τ. The "optimal" values of D and τ will be those which minimize the GCV statistic just like before. So now $GCV(\alpha, p, D, \tau)$ is the value we want to minimize. The actual model we use is a LWP model such that P is the matrix of predictors and $\mathbf{s}$ is the response variable. P and $\mathbf{s}$ together tell us what the response of the system given a certain state.

Usual multiple regression methods pick a single "best model" though the GCV values of several models may be close and discarding all but the top model could forfeit important information about the system. This lends to an alternate method whereas all models with GCV values within 5-20% of the top model are included and used to make predictions as well. This suite of models is then used to obtain an ensemble of predictions. This ensemble of predictions also gives an idea of the forecast uncertainty, i.e. the wider variation in ensemble members, the more uncertain the prediction.

The model for predicting T time steps into the future has the form

$$x_{I+T} = f(\mathbf{z}_I) \tag{2}$$

where $T = 0$ is considered the first future point and $\mathbf{z}_t$ is called the feature vector (but is actually just the current state of the system). $\mathbf{z}_t$ is just the missing embedded time series that we didn't define (remember we ended at $I-2$) where

$$\mathbf{z}_t = \begin{bmatrix} x_{I-(D-1)\tau-1} \\ x_{I-(D-2)\tau-1} \\ \vdots \\ x_{I-1} \end{bmatrix}.$$

For every new time step T into the future we need to reconstruct $\mathbf{z}_t$ (the state of the system) and include the newly forecasted points. Since P and $\mathbf{s}$ are only defined once and we are including forecasted points in our feature vector, forecasts will get worse as T increases.

2.1 Forecast Algorithm

The forecast algorithm as a whole proceeds as follows:

1. Construct P and $\mathbf{s}$ for a particular combination of α, p, D, and τ.
2. Fit a LWP model to P and $\mathbf{s}$ and calculate the GCV value for the particular fit.
3. Repeat steps 1-2 for all combinations of α, p, D, and τ.
4. Collect the parameter values for all models within 5-20% of the model with the lowest GCV value. Which percentage to use depends on what will capture a reasonable number of models (preferably <100 due to computational restrictions).
5. Form an ensemble by making a prediction from each of the captured models for T time steps into the future.

3 Data

3.1 Henon System

The Henon System is defined by the mapping

$$x_{n+1} = y_n + 1 - a x_n^2 \tag{3}$$
$$y_{n+1} = b x_n \tag{4}$$

Using this discrete time mapping 4000 points were generated with a=1.4, b=0.3. Figure 3 shows a portion of the synthetic data. The y coordinate of the henon system was chosen to forecast. Plotting y[i] vs. y[i+1] unfolds the attractor considerably more than y[i] vs. y[i+2] and y[i] vs. y[i+3] suggesting that $\tau = 1$ for the system (Figure 4). The system has two variables suggesting D=2. This preemptive knowledge drove the search for τ between 1-5 and D between 1-3.

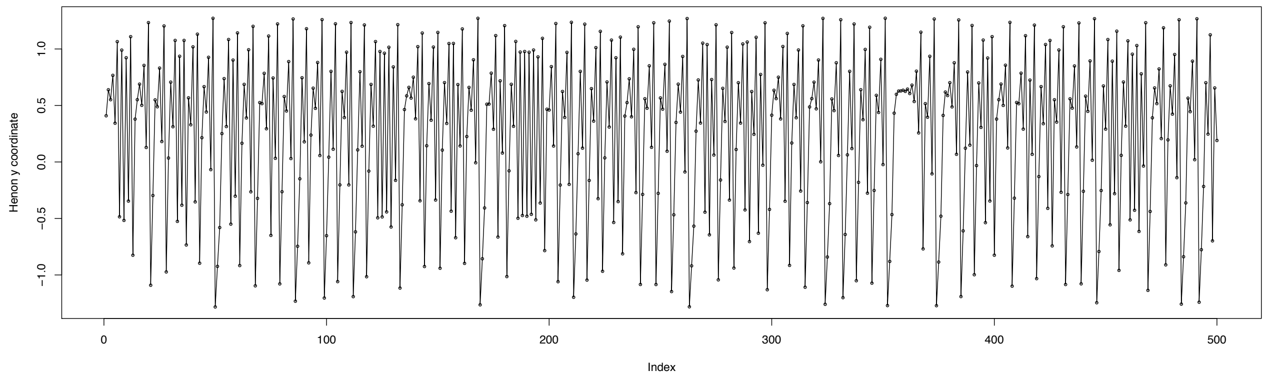


Figure 3: The y coordinate of the Henon system with a=1.4, b=0.3, $(x_0, y_0) = (0, 0)$.

3.2 Lorenz System

The Lorenz system is

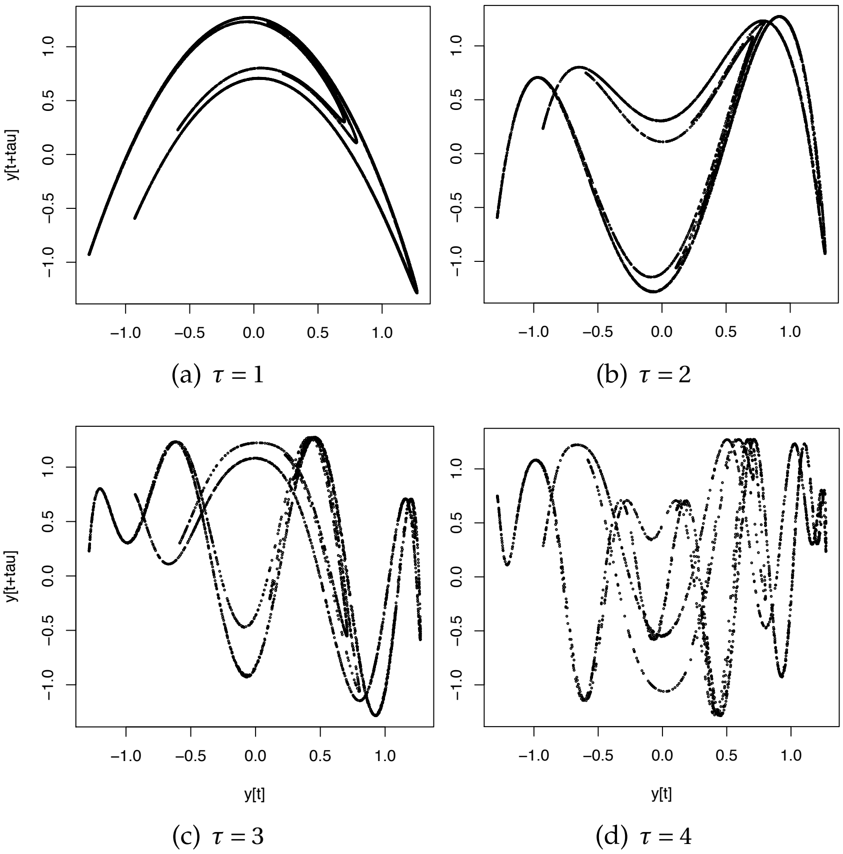
$$\dot{x} = \sigma(y - x) \tag{5}$$
$$\dot{y} = r x - y - x z \tag{6}$$
$$\dot{z} = x y - b z. \tag{7}$$

Using σ=16, r=45.92, b=4, Δt=0.01 and $(x_0, y_0, z_0) = (-14, -13, 47)$ 4000 points were generated. The z coordinate of the system was chosen to simulate. Figure 5 is a piece of the the time series. There are three coordinates in the Lorenz system suggesting D=3. The search was conducted for τ between 1-20 and D between 2-4.

3.3 Buoy 22 Wind Data

Wind speed data was obtained from the NOAA website[1]. The data is from buoy station 46022, dubbed "Buoy 22", which resides in Humboldt bay southwest of Eureka. Wind

[1] http://www.ndbc.noaa.gov/station_history.php?station=46022



(a) $\tau = 1$ (b) $\tau = 2$

(c) $\tau = 3$ (d) $\tau = 4$

Figure 4: The attractor is unfolded the most when $\tau = 1$.

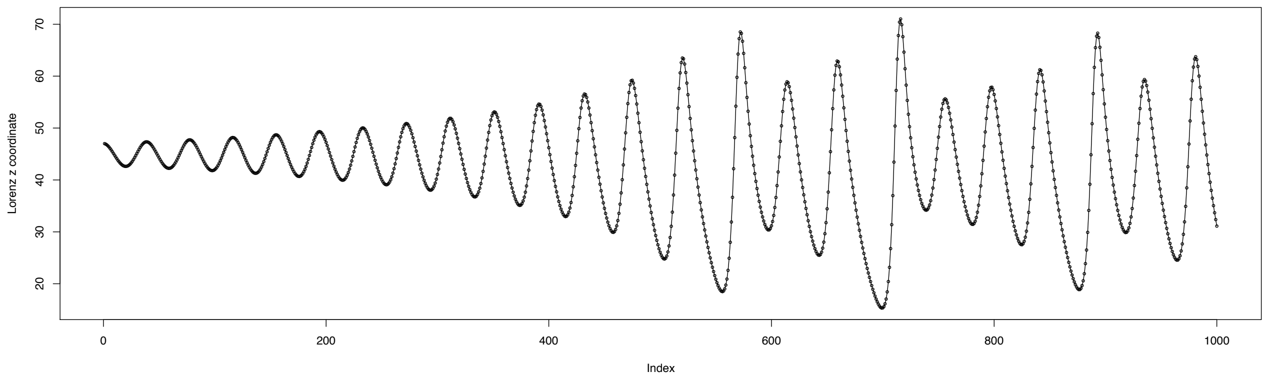


Figure 5: z coordinate of the Lorenz system with σ=16, r=45.92, b=4, Δt=0.01 and an the initial condition (-14, -13, 47).

speeds are in m/s. Wind data is notoriously noisy and this data is no different. Part of the reason this data was examined was to see if dynamics could be distinguished from the noise. The search was conducted for τ between 1-20 and D between 1-6.

4 Results

All forecasts were made on normalized time series values. Forecasts were compared to an AR model which is shown in all the following figures as a solid light blue line. The AR model does not take into account any information about the dynamics of the system.

4.1 Henon System

Figure 6 shows some predictions from the synthetic Henon data. Four models were within 20% of the top model. All of the ensemble members uses (interestingly enough) $D = 3$, $\tau = 2$, $p = 2$ with vaious values of α. In (a), (b), and (c) of Figure 6 the first few time steps of the forecast are smack dab on the actual values and there is no variability in the predictions. In (d) Figure 6 it was difficult to make accurate predictions because the series changes rapidly around index 3500. As we forecast more points the variability in the predictions increases. Beyond about 10 time steps, the forecasts are fairly worthless. Though opposed to the AR model the predictions are phenomenal. The AR model doesn't correctly forecast any points and quickly converges to the mean of the series.

4.2 Lorenz System

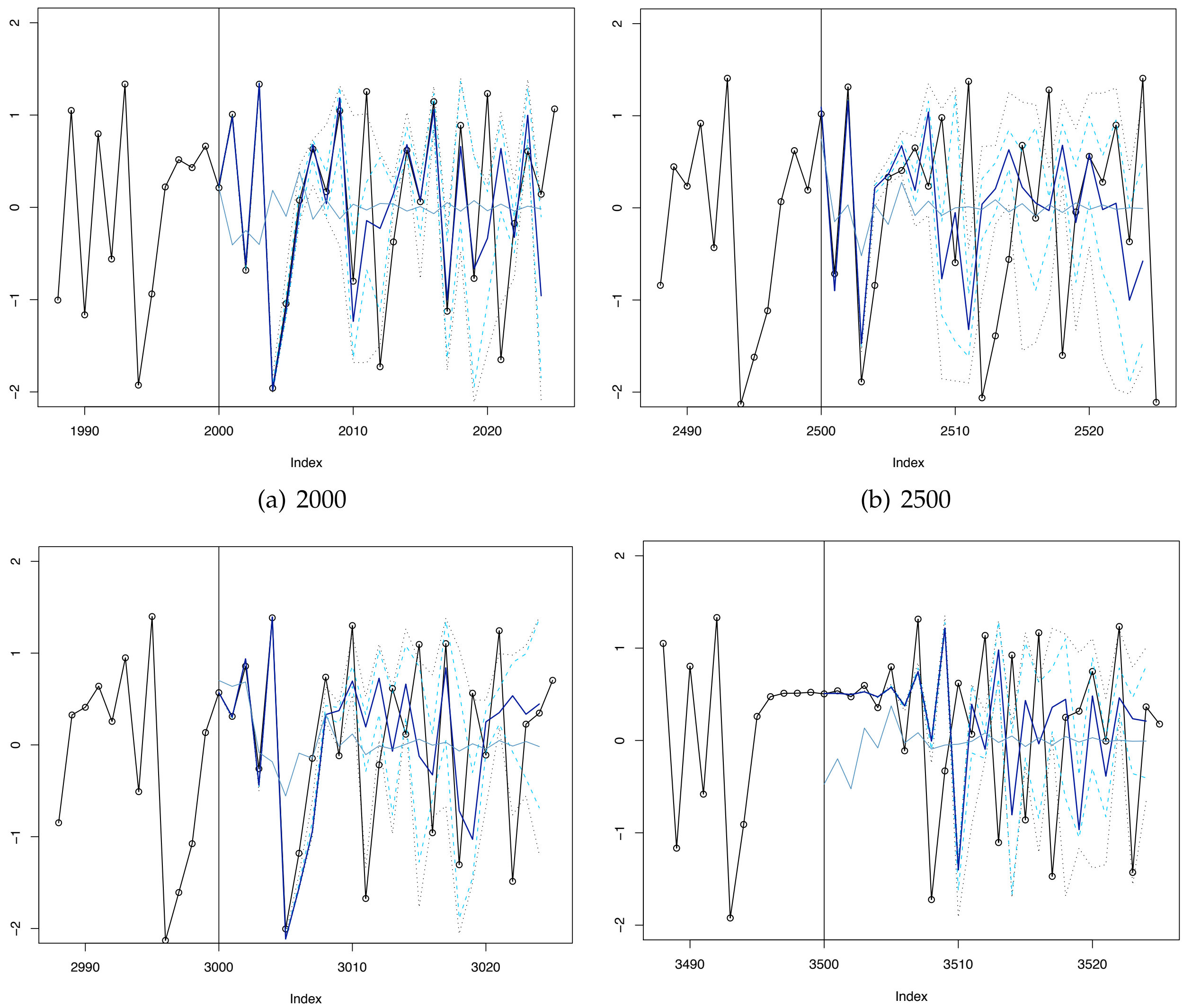
Figure 7 shows some predictions from the synthetic Lorenz data. There were 4 models within 20% of the best model. The ensemble members uses D =2 or 3, τ = 2 or 7, p = 2 with various values of α. In (a) of Figure 7 the the forecasts are spectacular being hardly distinguishable from the actual values for 100 forecastes points. This is partially because index 2000 corresponds to a time when the trajectory is making small spirals around the center of one arm of the attractor where trajectories are reasonably predictable. In (b), (c), and (d) of Figure 7 the system is going through more rapid change and is therefore less predictable. The forecasts do vary well for at least 50 time steps. The AR model predicts the form of the series but not the magnitude.

4.3 Buoy 22 Wind Data

Figure 8 shows some predictions of the normalized wind speed data (if they can be called forecasts). What can be said about these forecasts but that they are useless. They predict the same value at every forecast point There were 20 models within 5% of the best model (though there were 2000 models within 20%). The ensemble members used every possible combination of D, τ, p and α. This is one indication that there is not a strong chaotic attractor driving the system. Interestingly enough the forecasted values are not at the mean of the series but slightly offset. One consolation is that the AR model does not do any better at making forecasts.

Some possible reasons that this data set cannot be predicted are:

1. The underlying deterministic system is not chaotic. It is possible that the time series was generated by a non-chaotic deterministic process.

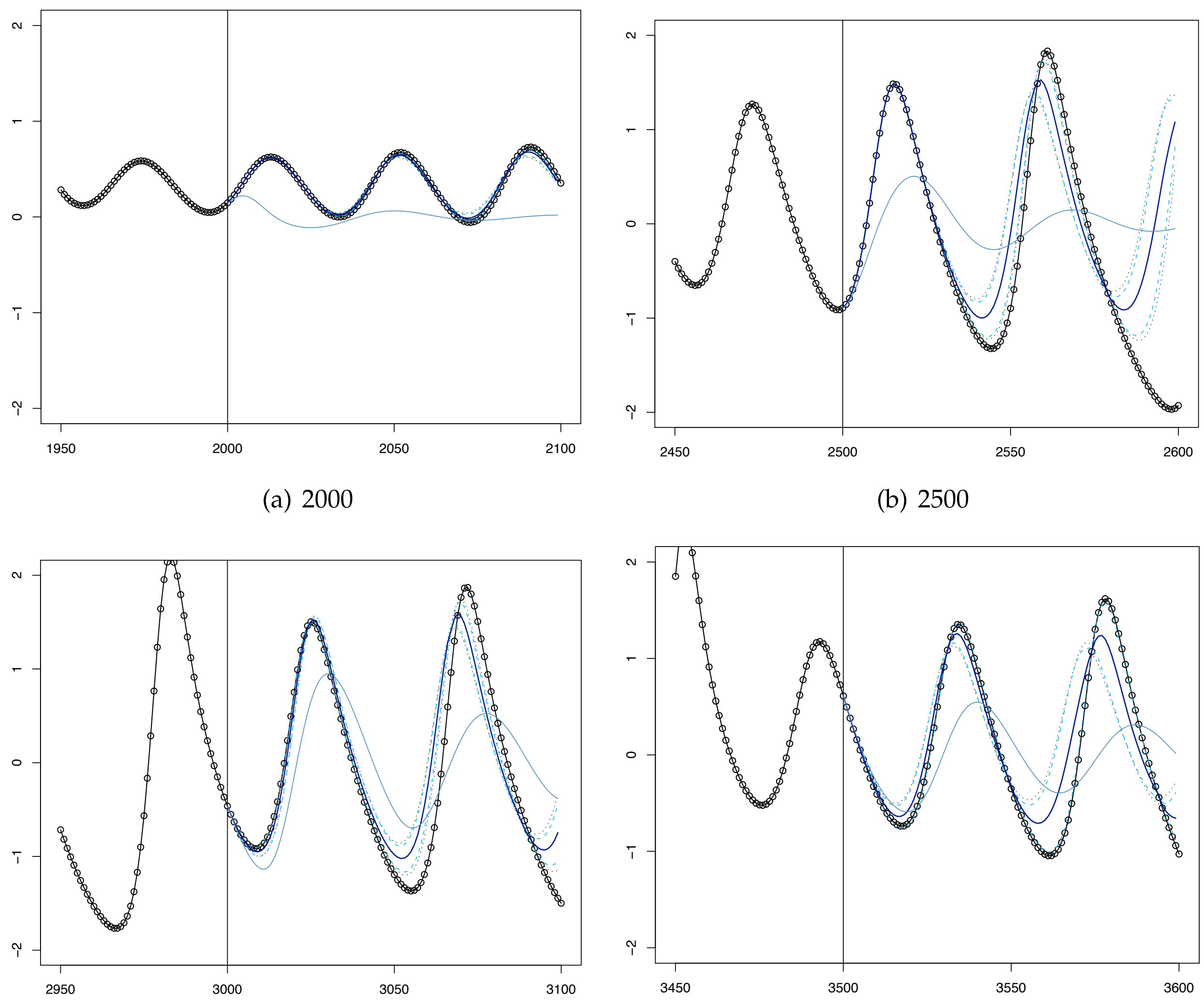


(a) 2000 (b) 2500

(c) 3000 (d) 3500

Figure 6: Forecasts for the normalized Henon System starting from various Indicies. The dotted black lines are the 5th and 95th percentile of the forecast. The dashed light blue lines are the 25th and 75th percentiles of the forecast. The solid dark blue line is the median of the forecasted values. The solid light blue line is the predictions of an autoregressive model.

2. The noise in the data overwhelms any real data that is present and therefore any chaos that may be present in the system.
3. The data set is not long enough. Long data sets are usually required with wind data because of its noisy nature.
4. It is hard to admit but though it preformed resonably well with synthetic data sets, maybe that is all this dynamical/statistical model is good for.



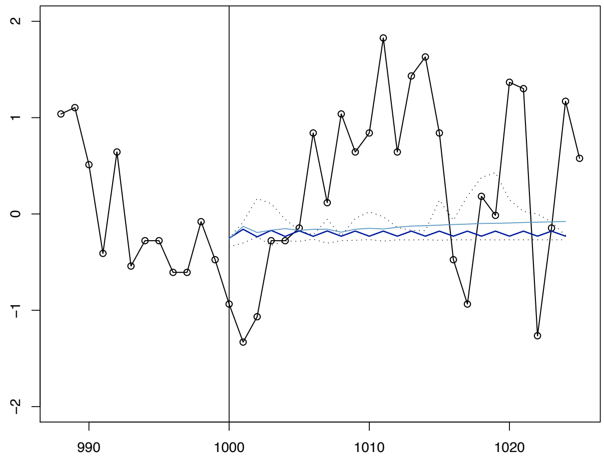
(a) 2000

(b) 2500

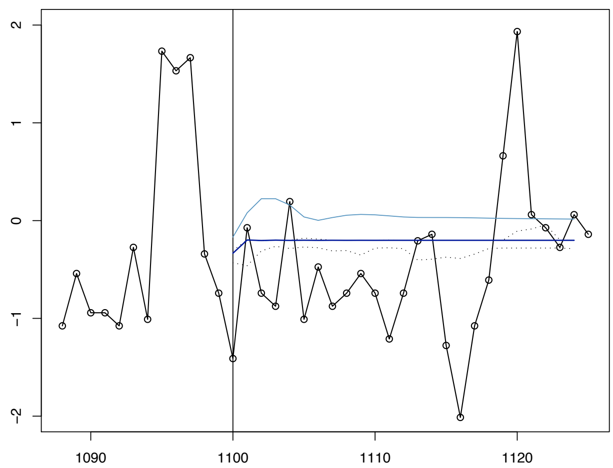
(c) 3000

(d) 3500

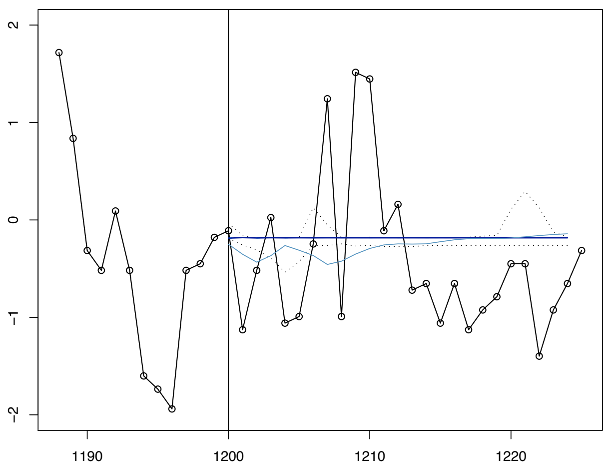
Figure 7: Same as figure 6 but for the normalized z coordinate of the Lorenz system.



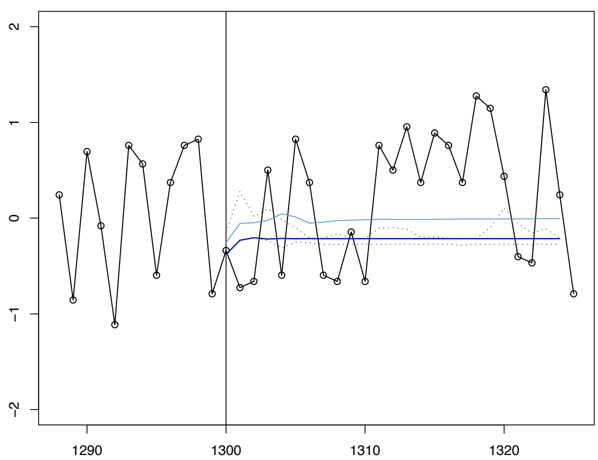
(a) March 1, 2003



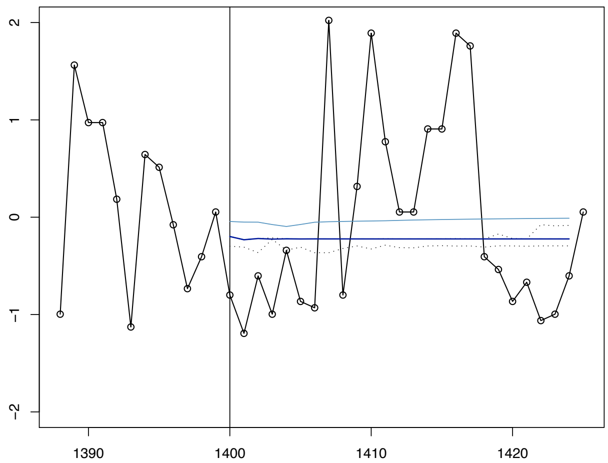
(b) June 9, 2003



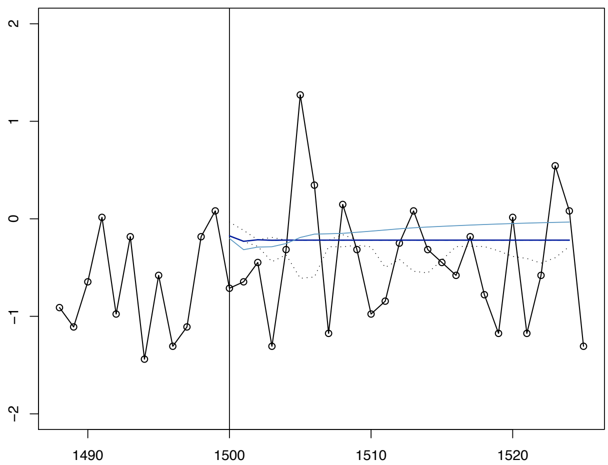
(c) September 17, 2003



(d) December 26, 2003



(e) April 4, 2004



(f) July 13, 2004

Figure 8: Same as figure 6 but for the normalized Buoy 22 wind data.

5 Conclusions

In this paper we presented a method for statistical forecasting of chaotic time series. This method incorporates information about the underlying dynamics of the system and is generally a short term approach. Successful short term forecasts were made of synthetic time series from the Lorenz and Henon systems which are known to be chaotic. The method did better at with the Lorenz System than for the Henon system. An attempt was made to apply this method to wind speed data but for various possible reasons, the forecasts were utterly useless. A better approach to forecast wind speed data might be to attempt to forecast distributional statistics instead of actual values.

6 Acknowledgements

I am most grateful to Satish Regonda for his example code and to Colin Ritter for his wind data analysis program which I was dreading having to write. Thanks also to Aaron Greenberg for his edits.

References

C. Chatfield. *The Analysis of Time Series*. Chapman and Hill, 1980.

K. Grantz, B. Rajagopalan, M. Clark, and E. Zagona. A technique for incorporating large-scale climate information in basin-scale ensemble streamflow forecasts. *Water Resources Research*, 41, 2005.

C. Loader. *Local Regression and Liklihood*. Springer, 1999.

N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. D. Shaw. Geometry from a time series. *Physics Review Letters*, 45:712–716, 1980.

S. K. Regonda, B. Rajagopalan, U. Lall, M. Clark, and Y.-I. Moon. Local polynomial method for ensemble forecast of time series. *Nonlinear Processes in Geophysics*, 12:1–10, 2005.

S. H. Strogatz. *Nonlinear Dynamics and Chaos*. Westview Press, 1994.

F. Takens. Detecting strange attractors in turbulence. *Dynamical Systems and Turbulence, Lecture Notes in Mathematics*, 898:366–381, 1981.