Deterministic Chaos and Neural Nets

by

A. Ronald Gallant
Fuqua School of Business
Duke University
Durham NC 27708-0120 USA

References

Nychka, Douglas W., Stephen P. Ellner, Daniel F. McCaffrey, and A. Ronald Gallant (1990), "Statistics for Chaos," *Statistical Computing and Statistical Graphics Newsletter* 1, 4–11.

Gallant, A. Ronald and Halbert L. White Jr. (1992) "On Learning the Derivatives of an Unknown Mapping with Multilayer Feedforward Networks," *Neural Networks* 5, 129–138. Revised and reprinted in White Jr., Halbert L. (1992), *Artificial Neural Networks*, Blackwell, Oxford UK, 206–223.

McCaffrey, Daniel F., Stephen P. Ellner, A. Ronald Gallant, and Douglas W. Nychka (1992), "Estimating the Lyapunov Exponent of a Chaotic System with Nonparametric Regression," *Journal of the American Statistical Association* 87, 682–695.

Topics

- Chaos looks random
- Caused by stretching and folding
- Lives on an attractor
- Exhibits sensitive dependence on initial conditions
- Generates a natural invariant ergodic measure on the attractor
- Taken's theorem justifies state space reconstruction
- Reconstruction using neural nets
- Detection of chaos

Chaos is generated by a nonlinear dynamical system in either discrete or continuous time.

Not all nonlinear systems exhibit chaotic dynamics, but a linear system cannot.

A discrete time dynamical system (or nonlinear autoregression) is written

$$x_{t+1} = f(x_t, x_{t-1}, \cdots, x_{t-d+1})$$

An example is the Henon map

$$x_{t+1} = 1 - a x_t^2 + b x_{t-1}$$

$$a = 1.4$$

$$b = 0.3$$

Its output looks random (next figure)

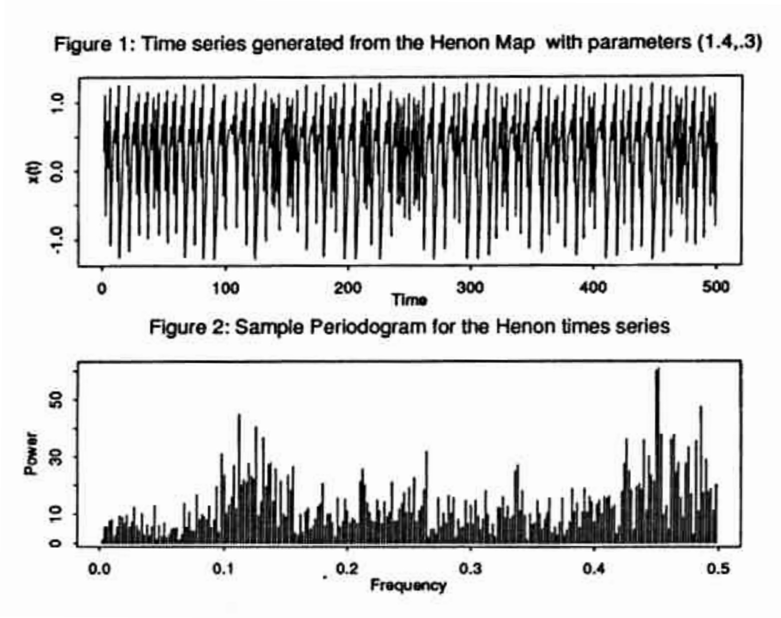


Figure 1: Time series generated from the Henon Map with parameters (1.4,.3)

Figure 2: Sample Periodogram for the Henon times series

Henon map:

$$x_{t+1} = 1 - 1.4x_t^2 + 0.3x_{t-1}$$

State Space Form

$$X_t = \begin{pmatrix} x_t \\ x_{t-1} \\ \vdots \\ x_{t-d+1} \end{pmatrix}$$

$$X_{t+1} = F(X_t)$$

$$F : \mathcal{X} \to \mathcal{X} \subset \Re^d$$

Example: Henon Map

$$\begin{pmatrix} x_{t+1} \\ x_t \end{pmatrix} = \begin{pmatrix} 1 - ax_t^2 + bx_{t-1} \\ x_t \end{pmatrix}$$

Attractor $\quad \mathcal{Z} \subset \mathcal{X} \subset \Re^d$ (next figure for Henon)

Stretching and folding (next figure for Henon)

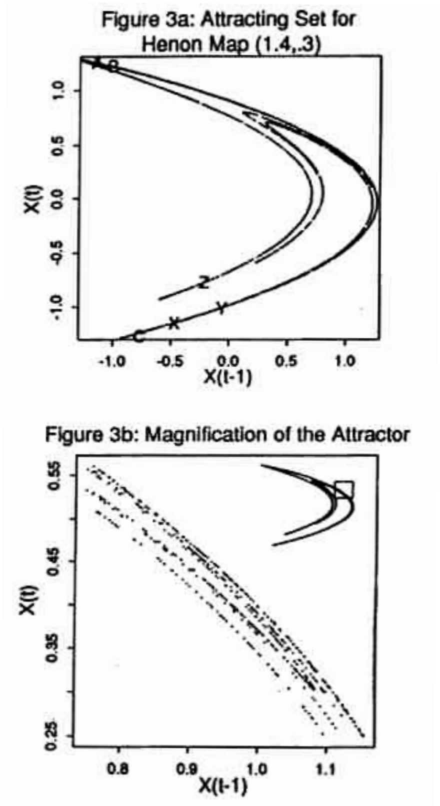


Figure 3a: Attracting Set for Henon Map (1.4,.3)

Figure 3b: Magnification of the Attractor

- $\mathcal{X} = (-1.5, 1.5) \subset \Re^2$, $\mathcal{Z}$ is indicated by the set in 3a
- Stretching: Close points X, Z sent to distant points A, C.
- Folding: Distant points X, Y sent to close points A, B.

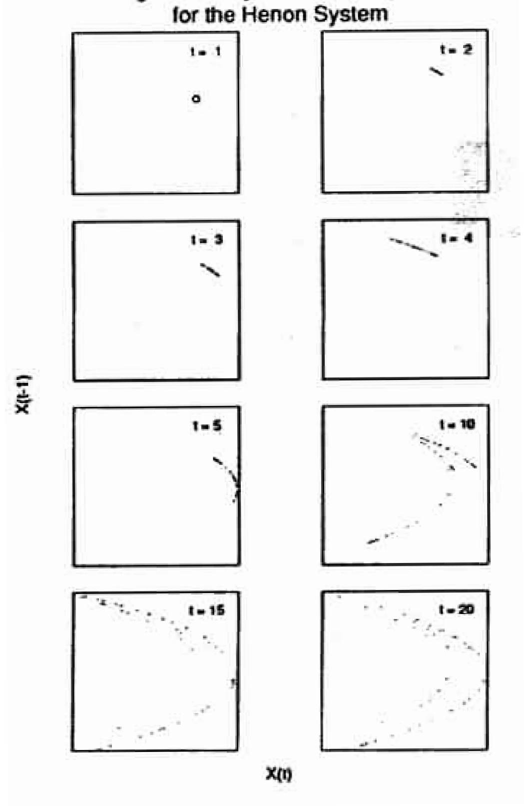
Sensitive Dependence on Initial Conditions

Points initially close together get dispersed throughout the attractor.

Example: Henon Map (next figure)

$$\begin{pmatrix} x_{t+1} \\ x_t \end{pmatrix} = \begin{pmatrix} 1 - ax_t^2 + bx_{t-1} \\ x_t \end{pmatrix}$$

Figure 4: Trajectories of 50 points for the Henon System



Discretized Mackey-Glass

More interesting than the Henon map because it exhibits some features of data from financial markets.

$$\begin{aligned} x_t &= f(x_{t-1}, x_{t-5}) \\ &= x_{t-1} + 10.5 \left[\frac{0.2 x_{t-5}}{1 + (x_{t-5})^{10}} - 0.1 x_{t-1} \right] \end{aligned}$$

Transformed Mackey-Glass

$$y_t = Q_T(x_t)$$

- t-density on six degrees of freedom

$$f_T(x) = \frac{\Gamma(7/2)}{\sqrt{6\pi}\Gamma(3)} \left(1 + x^2/2\right)^{-7/2}$$

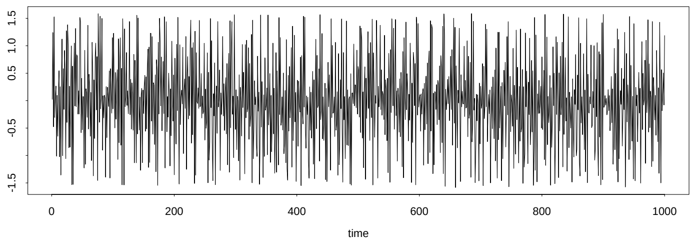
- t-distribution on six degrees of freedom

$$F_T(x) = \int_{-\infty}^{x} f_T(t)\, dt$$

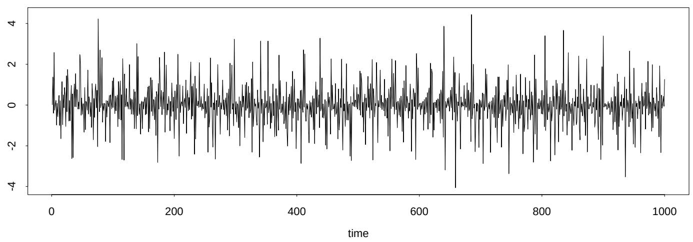
- quantile function

$$Q_T(p) = F_T^{-1}(p)$$

Discretized Mackey-Glass

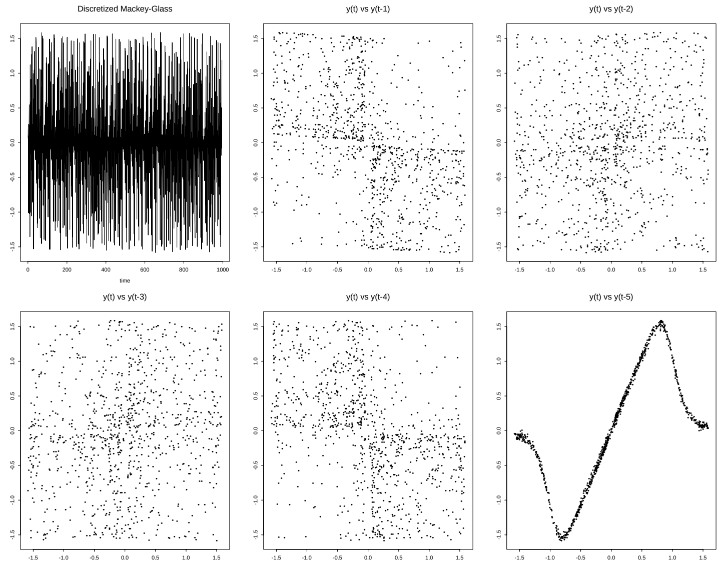


Six Degree Freedom t Quantiles of Discretized Mackey-Glass

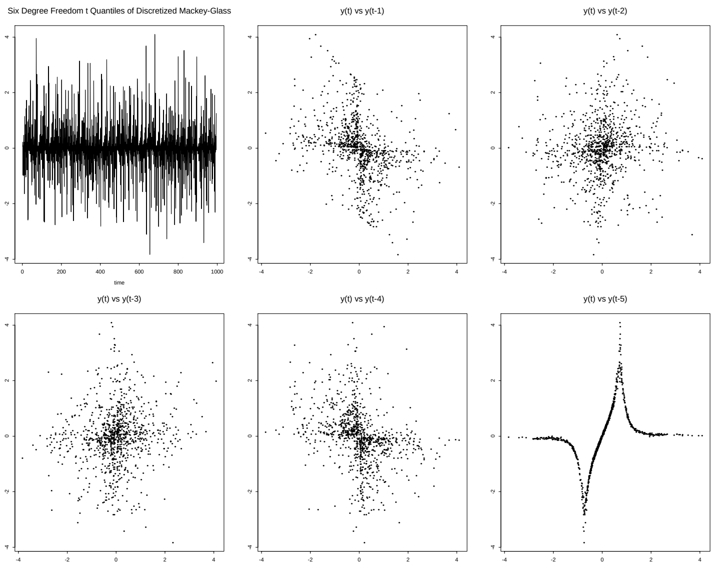


Discretized Mackey-Glass: $x_t = x_{t-1} + 10.5 \left[\frac{0.2 x_{t-5}}{1+(x_{t-5})^{10}} - 0.1 x_{t-1} \right]$

Transformed Mackey-Glass: $y_t = Q_T(x_t)$



The attractor of the discretized Mackey-Glass equation.



The attractor of the transformed Mackey-Glass equation.

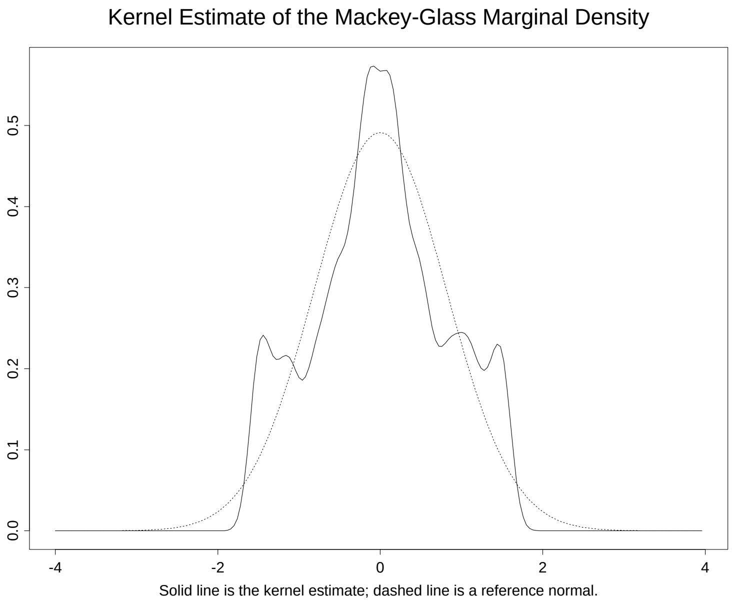
Natural invariant measure

$$\mu(A) = \lim_{n\to\infty} \frac{1}{n} \#\{x_t \text{ in } A : 1 \le t \le n\}$$

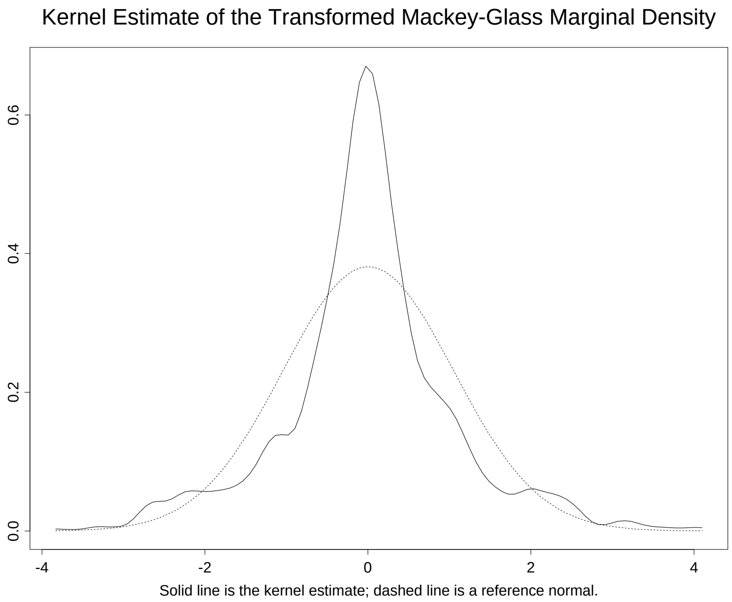
$\mathcal{Z}$ is the attractor

Ergodic

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} g(x) = \int_{\mathcal{Z}} g(x)\, d\mu(x)$$



Kernel estimate of the natural invariant measure of the discretized Mackey-Glass equation.

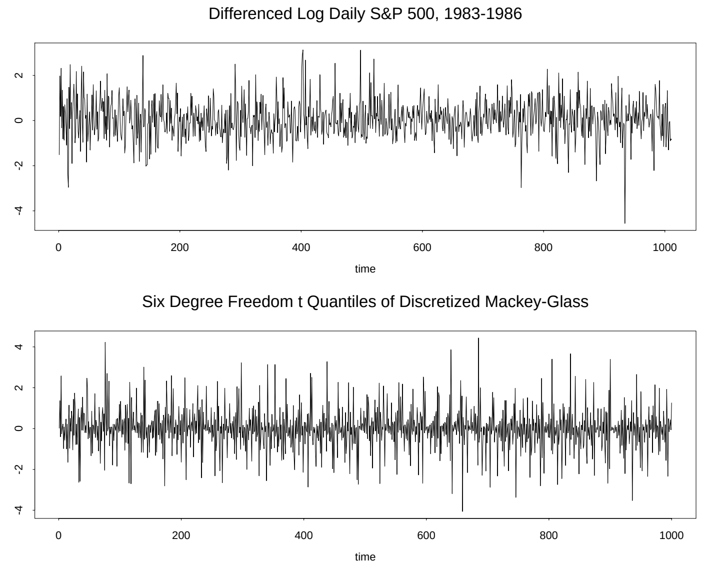


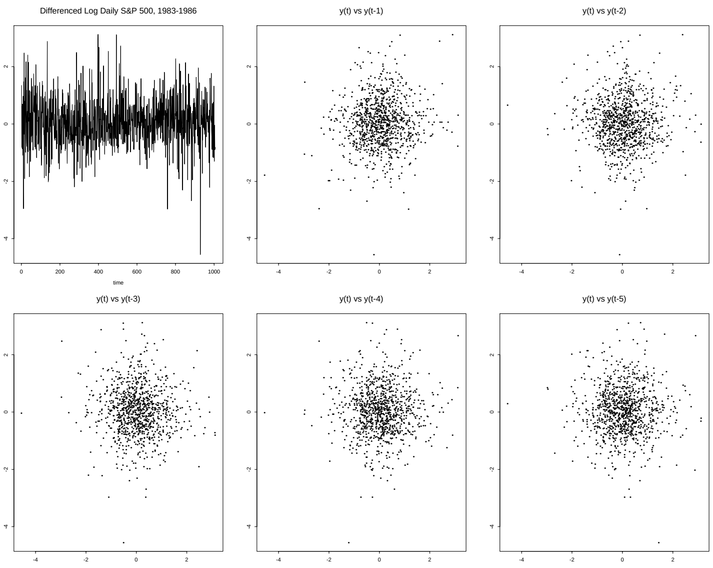
Kernel estimate of the natural invariant measure of the transformed Mackey-Glass equation.

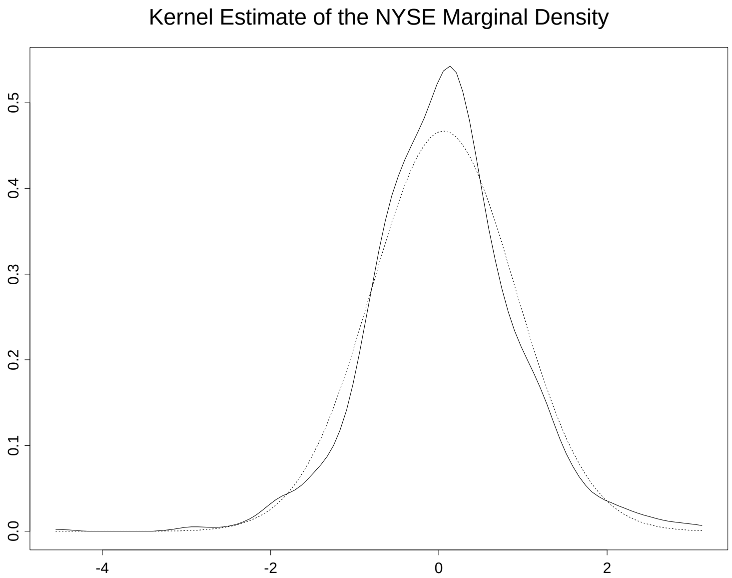
Comparisons with some random processes

- Daily returns on the S&P 500, 1983–1986

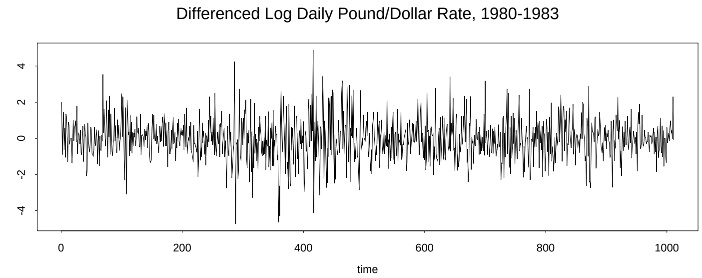
- Daily returns on the British pound to U.S. dollar exchange rate, 1980–1983

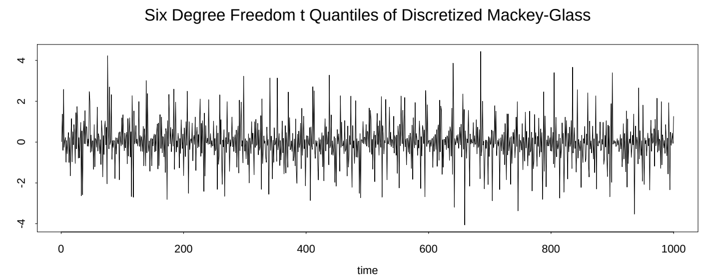


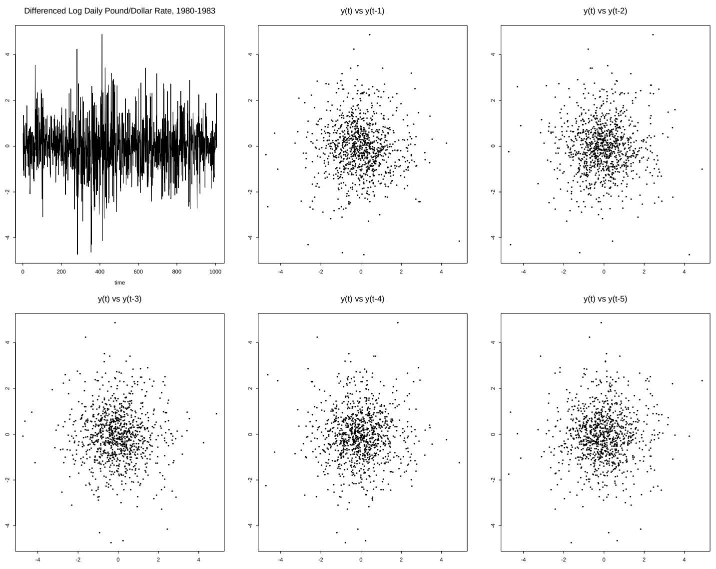


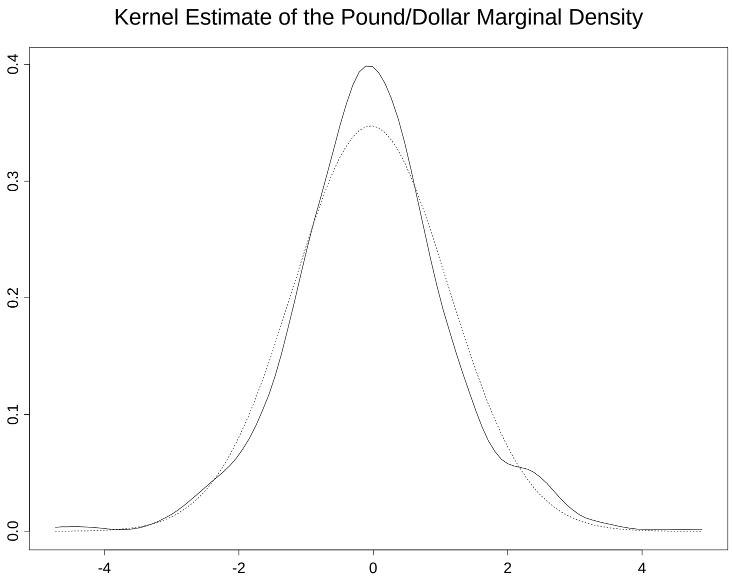


Solid line is the kernel estimate; dashed line is a reference normal.









Solid line is the kernel estimate; dashed line is a reference normal.

Taken's Theorem

If x_t is an element of the state vector of a discrete or continuous time chaotic process then x_t has the representation

$$x_{t+1} = g(x_t, \cdots, x_{t-d+1})$$

for some d and some g; equivalently,

$$X_{t+1} = G(X_t)$$

for some d and some G, where

$$X_t = \begin{pmatrix} x_t \\ x_{t-1} \\ \vdots \\ x_{t-d+1} \end{pmatrix}.$$

Importance

This result justifies the use of nonparametric methods to recover g. Neural nets are particularly useful in this connection because, unlike most nonparametric methods, they can interpolate as well as smooth.

Discretized Mackey-Glass

True relation

$$
\begin{aligned}
x_t &= f(x_{t-1}, x_{t-5}) \\
&= x_{t-1} + 10.5\left[\frac{0.2x_{t-5}}{1+(x_{t-5})^{10}} - 0.1x_{t-1}\right]
\end{aligned}
$$

We shall attempt to recover f by fitting functions of the form

$$x_t = g(x_{t-1}, \ldots, x_{t-d});$$

specifically, by fitting neural nets.

Single Hidden Layer Feedforward Neural Net

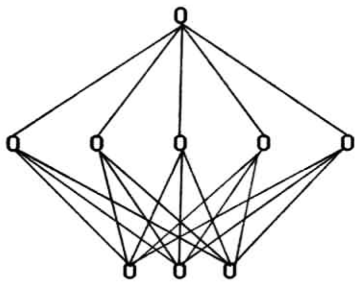
$$
\begin{aligned}
&g_K(x_{t-1}, \cdots, x_{t-5}) \\
&\qquad = \beta_0 + \sum_{j=1}^{K} \beta_j G(\gamma_{0j} + \gamma_{1j}x_{t-1} + \cdots + \gamma_{5j}x_{t-5})
\end{aligned}
$$

$$G(u) = \exp(u)/[1+\exp(u)]$$

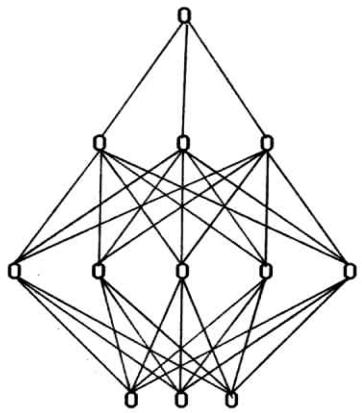
Learning rule

Choose the $\beta's$ and $\gamma's$ to minimize

$$\frac{1}{n}\sum_{t=1}^{n}\left[x_t - g_K(x_{t-5}, \cdots, x_{t-1})\right]^2$$



Single Hidden Layer Feedforward Neural Net



Double Hidden Layer Feedforward Neural Net

Using a Single Hidden Layer Feedforward Neural Net

$$
\begin{aligned}
g_K(x_{t-1}, \cdots, x_{t-5}) & \\
&= \beta_0 + \sum_{j=1}^{K} \beta_j G(\gamma_{0j} + \gamma_{1j} x_{t-1} + \cdots + \gamma_{5j} x_{t-5}) \\
G(u) &= \exp(u)/[1 + \exp(u)]
\end{aligned}
$$

to Recover Mackey-Glass Dynamics

$$
\begin{aligned}
x_t &= f(x_{t-1}, x_{t-5}) \\
&= x_{t-1} + 10.5 \left[\frac{0.2 x_{t-5}}{1 + (x_{t-5})^{10}} - 0.1 x_{t-1} \right]
\end{aligned}
$$

Performance

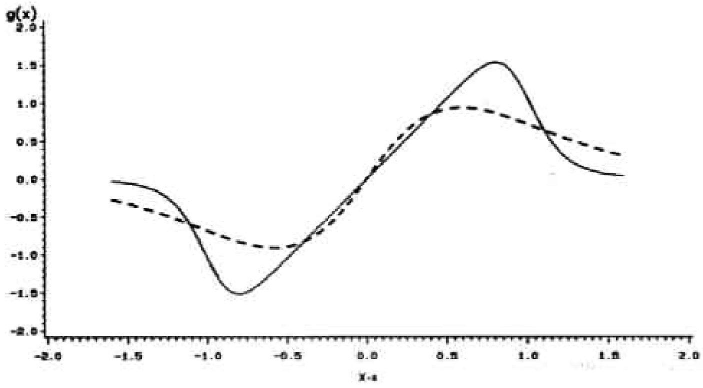
Measures of performance (following tables and figures)

TABLE 1
Predictor Error and Error in Sobolev Norm of an Estimate of the Nonlinear Map of a Chaotic Process by a Neural Net

K	n	PredErr($\hat{g}_K$)	$\lVert g^* - \hat{g}_K \rVert_{1,\infty,\chi}$	$\lVert g^* - \hat{g}_K \rVert_{1,2,\chi}$	Saturation Ratio
3	500	0.3482777075	3.6001114788	1.3252165780	17.9
5	1,000	0.0191675679	0.5522597668	0.1604392912	28.6
7	2,000	0.0177867857	0.4145203548	0.1141557050	40.8
9	4,000	0.0134447868	0.2586038122	0.0719887443	63.5
11	8,000	0.0012308988	0.1263063691	0.0196351730	103.9

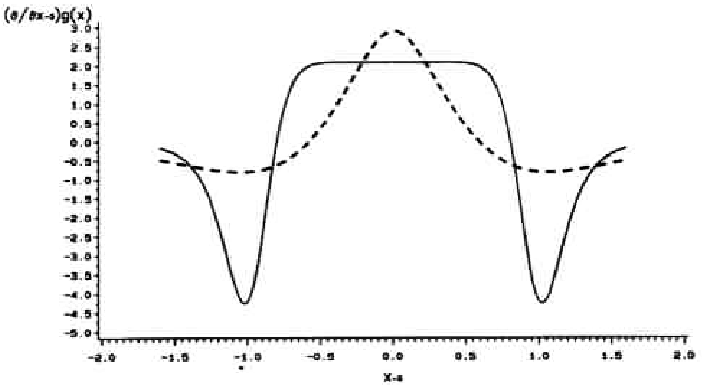
TABLE 2
Sensitivity of Neural Net Estimates

K	n	PredErr($\hat{g}_K$)	$\lVert g^* - \hat{g}_K \rVert_{1,\infty,\chi}$	$\lVert g^* - \hat{g}_K \rVert_{1,2,\chi}$	Saturation Ratio
7	500	0.0184102390	0.3745884175	0.1325439320	10.2
7	2,000	0.0177867857	0.4145203548	0.1141557050	40.8
11	500	0.0076063363	0.7141377059	0.1115357981	6.5
11	4,000	0.0015057013	0.0858882780	0.0210710677	51.9
11	8,000	0.0012308988	0.1263063691	0.0196351730	103.9
15	8,000	0.0020546210	0.1125778860	0.0336124596	76.2



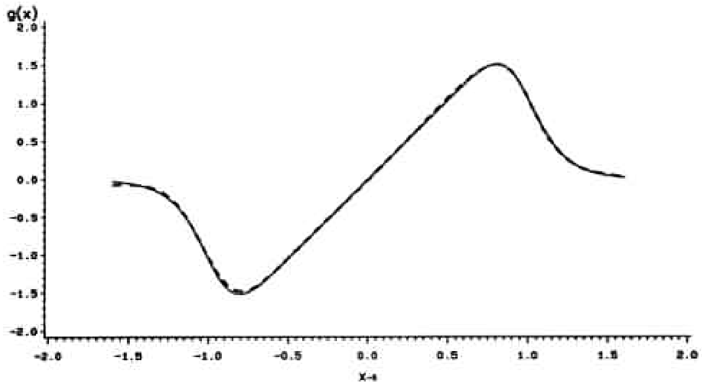
Note: Estimate is dashed line, x = (x₋₅, 0, 0, 0, 0)

FIGURE 1. Superimposed nonlinear map and neural net estimate; $K = 3$, $n = 500$.



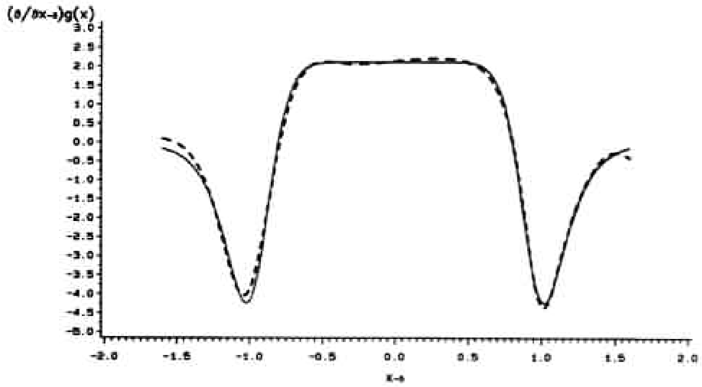
Note: Estimate is dashed line, x = (x-s, 0, 0, 0, 0)

FIGURE 2. Superimposed derivative and neural net estimate; $K = 3$, $n = 500$.



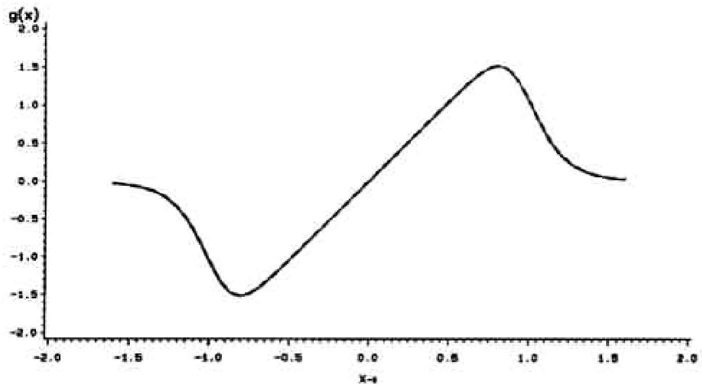
Note: Estimate is dashed line, x = (x-s, 0, 0, 0, 0)

FIGURE 3. Superimposed nonlinear map and neural net estimate; $K = 7$, $n = 2{,}000$.



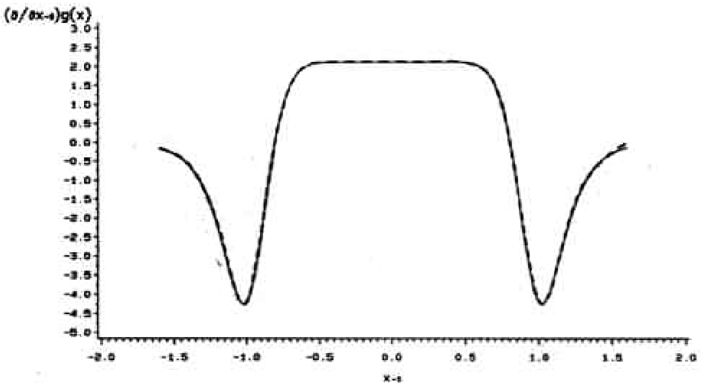
Note: Estimate is dashed line, x = (x-s, 0, 0, 0, 0)

FIGURE 4. Superimposed derivative and neural net estimate; $K = 7$, $n = 2{,}000$.



Note: Estimate is dashed line, x = (x-s, 0, 0, 0, 0)

FIGURE 5. Superimposed nonlinear map and neural net estimate; $K = 11$, $n = 8{,}000$



Note: Estimate is dashed line, x = (x_{-1}, 0, 0, 0, 0)

FIGURE 6. Superimposed derivative and neural net estimate; K = 11, n = 8,000.

Detection of Chaos

- Lyapunov exponents
- A measure of senstitivity to initial conditions
- Fit neural net.
- If Lyapunov exponent larger than zero, then chaos.

Dominant Lyapunov Exponent

$$\lambda = \lim_{t\to\infty} \frac{1}{t} \log \|J_{t-1} \cdot J_{t-2} \cdot .. \cdot J_0\|$$
$$J_t = (\partial/\partial x')F(X_t)$$

- $F(X_t)$ is the dynamical system in state space form.
- $\|A\|$ is the Euclidean norm of Ay where y is chosen to make the Euclidean norm of Ay as large as possible.
- For t large enough, any $y \neq 0$ will work.

Interpretation

Two initially close points: $X_0^{(1)}$ and $X_0^{(2)}$

Iterate them t steps ahead: $X_t^{(1)}$ and $X_t^{(2)}$

First order approximation:

$$X_t^{(2)} - X_t^{(1)} \doteq J_{t-1}J_{t-2}..J_1J_0[X_0^{(2)} - X_0^{(1)}]$$

$$\lambda \doteq \frac{1}{t}\log \|J_{t-1}J_{t-2}\cdots J_0[X_0^{(2)} - X_0^{(1)}]\|$$
$$\doteq \frac{1}{t}\log \|X_t^{(2)} - X_t^{(1)}\|$$

If $\lambda > 0$ then $X_t^{(1)}$ and $X_t^{(2)}$ diverge exponentially fast

"sensitive dependence on initial conditions"

Estimating the Lyapunov Exponent

When $F(X_t)$ is estimated from data $\{X_t\}_{t=1}^{n}$, one averages blocks of size M

$$\widehat{\lambda}_i \;=\; \frac{1}{M}\log \|J_{iM-1}\cdot J_{iM-2}\cdot ..\cdot J_{iM-M}\|$$

$$\widehat{\lambda} \;=\; \frac{1}{n/M}\sum_{i=1}^{n/M}\widehat{\lambda}_i$$

where $M = \log(n)$

Table 1. Estimated Lyapunov Exponents for the Hénon and Rössler Systems Without Noise

Hénon system[c]						
		d				
Map estimate	*N*	*1*	*2*	*3*	*4*	*5*
Local spline[a]	2,500	5.7602 (.042)	.4188[e] (.005)	.0750 (.011)	−.0251 (.013)	.0259 (.013)
Neural net	2,000	.1147	.4106	.4227	.4236[e]	—
Projection pursuit	2,000	—	.4163	.4058	.4026[e]	—

Rössler system[d]								
		d						
Map estimate	*N*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
Local spline[b]	2,500	7.1229 (.055)	.0992 (.004)	.0461[e] (.002)	1.7099 (.011)	1.567 (.21)	—	—
Radial basis	2,000	—	.0629	.7778[e]	10.24	10.26	—	—
Neural net	2,000	—	.0010	.1272	.6940	.0482	.0414	.0466[e]
Projection pursuit	2,000	—	—	.0966[e]	.0146	−.2792	−.0640	—

NOTE: The value of *m* for the local spline estimates was the smallest integer such that $2m > d$. For the radial basis function estimates 200 basis functions were used. *N* is the length of the data series, and *d* is the dimension of the model.

[a] Average of five estimates with standard deviation.

[b] Average of ten estimates with standard deviation.

[c] Correct value of λ is approximately .418 (Wolf et al. 1985, p. 289).

[d] Correct value of λ is approximately .04505 (Wolf et al. 1985, p. 289).

[e] Estimate of λ_1 that corresponds to the minimum expected prediction error, $\hat{\sigma}^2$.

Table 2. Estimated Lyapunov Exponents as a Function of Block Size M for the Hénon Map With Noise

		M				
Map estimate	*d*	*50*	*100*	*500*	*2,000*	*20,000*
Local spline[a]	2	.421 (.015)	.426 (.014)	.417 (.015)	.416 (.015)	
Neural net[b]	2	.416 (.020)	.412 (.020)	.408 (.020)	.408 (.019)	
Exact map[c]	2	.415 (.010)	.414 (.009)	.409 (.009)	.408 (.009)	.408 (.009)
Local spline	5	.417 (.009)	.431 (.009)	.406 (.008)	.404 (.008)	
Neural net	5	.417 (.007)	.411 (.009)	.406 (.009)	.405 (.009)	

NOTE: Each Lyapunov exponent estimate is the average of the exponents obtained from *N/M* disjoint blocks of the data series. The data series consisted of $N = 2{,}000$ values for $M \leq 2{,}000$, and $N = 20{,}000$ for $M = 20{,}000$; *d* is the dimension of the model.

[a] Average of 14 estimates with standard deviation.

[b] Average of 16 estimates with standard deviation.

[c] Average of 200 estimates using the true Jacobian matrix. The standard deviation has been adjusted to be comparable with the other estimates (reported S.D. = sample S.D./$\sqrt{B}$ where $B = N/M$.) The correct value of λ is approximately .408.