Dan Stahlke
PHYS694

# Nonlinear Noise Reduction

Noise reduction is an essential step in any measurement process. Traditionally noise reduction has made use of linear techniques such as the Fourier transform which work by separating the signal from the noise in frequency space or another linear space. Many signals however cover the same spectral region as the noise and so need an improved filtration technique.[1] If a signal comes from a low dimensional deterministic chaotic system (and sometimes even if it doesn't) then nonlinear methods can be used with amazing success. The algorithm described in this paper involves projection onto a low dimensional chaotic attractor that has been embedded into a higher dimensional space. This nonlinear projection method works well for stationary deterministic systems and with care can be used with slowly varying non-stationary data as well. Applications include cleaning of noisy speech or ECG signals and extraction of a small fetal ECG signal that is mixed with a maternal ECG signal.

## Phase Space

A deterministic flow can be described by a set of variables which evolve over time $\dot{\vec{x}} = \vec{f}(\vec{x})$. Most of the time the $\vec{x}$ are not measured directly but indirectly at discrete time intervals via a measurement function $\vec{h}(\vec{x})$.[2] For the purpose of discussion assume that only a single scalar value $s_n$ is measured at each time $n\delta t$, although it is easy to generalize to the case of multi-valued measurements. At first glance it may seem that too much information has been lost in projecting the tuple $\vec{x}$ onto the scalar $s_n$ but it turns out that the chaotic attractor can be reconstructed using a set of delay vectors $\vec{s}_n = (s_n, s_{n-\nu}, \ldots, s_{n-(m-1)\nu})$. A theorem by Takens and Sauer et al. shows that this works for almost all lags $\nu$, all sampling rates $\delta t$, and all smooth measurement functions $\vec{h}$ as long as the delay vector has a dimension of at least $m > 2D$ where $D$ is the box-counting dimension of the attractor.[3]

The box-counting dimension may not be known ahead of time so some caution must be taken in choosing the embedding dimension. There is usually no harm in choosing the dimension to be too high other than some additional computational complexity. ECG data are often embedded in a space of dimension 50 or higher. Several techniques are available for testing whether the chosen dimension is high enough. In particular, the projected data must be deterministic (ie. paths must not cross).

## Simple Noise Reduction

Most nonlinear prediction or noise reduction methods involve searching the history of the signal for trajectories similar to the portion under consideration. Trajectories that are similar in the past will tend to diverge in the future due to the positive Lyapunov exponents of $\vec{f}(\vec{x})$ and trajectories that are similar in the future will tend to have divergent histories due to the negative Lyapunov exponents and these divergences are exaggerated by the measurement noise.[4] On the other hand, trajectories that are similar in both past and future will tend to have temporal center points that match well. The simplest noise reduction then consists of replacing each sample with the average of all samples that have similar past and future trajectories:

---

1 T. Schreiber, H. Kantz, *Nonlinear projective filtering II: Application to real time series. (1998)*
2 R. Hegger, H. Kantz, Embedding of sequences of time intervals. (1997)
3 R. Hegger, H. Kantz, Embedding of sequences of time intervals. (1997)
4 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 52.

$$\hat{s}_{n_0-m/2} = \frac{1}{|\mathcal{U}_\epsilon(\vec{s}_{n_0})|} \sum_{s_n \in \mathcal{U}_\epsilon(\vec{s}_{n_0})} s_{n-m/2}$$

where $\mathcal{U}_\epsilon(\vec{s}_{n_0})$ represents the set of all delay vectors in the vicinity of $\vec{s}_{n_0}$.[5] A good choice for $\epsilon$ is usually about 2-3 times the noise amplitude, with a lower value being less likely to introduce artifacts.[6] Typically the Euclidean norm is used but it is sometimes beneficial to use for example the max norm.[7]

The processes of finding the neighbors of each point in phase space is the most expensive step in most nonlinear noise reduction algorithms.[8] The naive approach of checking the distance between each pair gives a running time of $O(N^2)$ but using a binary tree approach can give $O(NlnN)$ time and a box assisted search can potentially reduce the search time to $O(N)$.[9] Typically the entire algorithm is iterated 2-5 times with a smaller value of $\epsilon$ used in each iteration as the noise decreases.[10]

## Locally Projective Noise Reduction

A more refined approach involves projecting the measured trajectories onto the surface of a low dimensional attractor. The justification for this is that the data should ideally be confined to a chaotic attractor of a certain dimension and any deviation from this attractor is therefore due to measurement noise. The attractor is locally approximated by a tangent space whose shape is determined using a local principal component analysis.[11] The largest few principle components correspond to the subspace of the attractor and the remaining components correspond purely to noise.[12] The trajectories are then projected onto the attractor with the effect of reducing noise. This can be thought of as a local version of the singular systems approach that takes into account the curved structures of nonlinear systems.[13] It can also be thought of as a locally first-order (linear) approximation of phase space structure whereas the simple algorithm described in the previous section was a locally zeroth-order (constant) approximation.[14]

As in the simple algorithm of the previous section, the first step is to compute the neighborhood $\mathcal{U}_n$ of each vector $s_n$. Principal component analysis is performed by first computing the mean

$$\bar{s}^{(n)} = \frac{1}{|\mathcal{U}_n|} \sum_{k \in \mathcal{U}_n} \vec{s}_k$$

and then the covariance matrix

$$C_{ij} = \sum_{n' \in \mathcal{U}_n} (s_{n'} - \bar{s}^{(n)})_i (s_{n'} - \bar{s}^{(n)})_j.$$ [15]

The eigenvectors of $C_{ij}$ will then represent the semi-axes of the ellipsoid best approximating the cloud of points in $\mathcal{U}_n$. Ideally the covariance matrix will have large eigenvalues spanning the attractor manifold and low eigenvalues in other directions.[16] Projecting each vector onto the space spanned by the largest eigenvalues will move it closer to the manifold thereby creating a more accurate

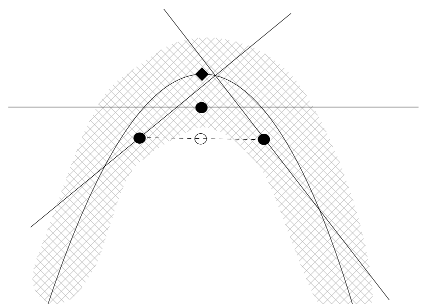
5 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 53
6 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 54
7 T. Schreiber and M. Richter, *Fast nonlinear projective filtering in a data stream.* (1999)
8 T. Schreiber and M. Richter, *Fast nonlinear projective filtering in a data stream.* (1999)
9 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 56
10 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 56
11 H. Kantz and T. Schreiber, *Nonlinear projective filtering I: Background in chaos theory.* (1998)
12 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 157
13 T. Schreiber and M. Richter, *Fast nonlinear projective filtering in a data stream.* (1999)
14 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 52, 156
15 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 158
16 H. Kantz and T. Schreiber, *Nonlinear projective filtering I: Background in chaos theory.* (1998)

approximation of the true dynamics of the system. Each measurement value occurs in $m$ different vectors (where $m$ is the dimension of the embedding) so an average needs to be taken over the corrections corresponding to each embedding vector that contains a given measurement.[17] Because of this averaging the vectors will not be projected exactly onto the manifold but will still be moved closer to it. The procedure must be iterated several times to reach convergence.[18]

There is a trade off involved in the choice of neighborhood size. $\epsilon$ must be larger than the noise level in order to give a fair representation of the manifold shape but must also be small enough so that curvature effects don't interfere with the linear approximation.[19]

Since vectors are projected towards the center of mass of their neighborhood there is a tendency for corrections to be biased toward the direction of curvature as can be seen in the figure below. The end result after noise reduction is then a manifold that has the same details and invariant quantities (such as fractal dimension) but is somewhat distorted in comparison to its true shape. There is a simple correction for this drift – the average centers of mass of the adjacent neighborhoods (open dot in figure) will tend to be twice as far from the manifold (diamond) as the center of mass $\vec{s}_n$ (closed dot in center). A corrected center of mass is therefore given by

$$\bar{\bar{s}}^n = 2\bar{s}^{(n)} - \frac{1}{|\mathcal{U}_n|} \sum_{n' \in \mathcal{U}_n} \bar{s}^{(n')}.$$ [20]



The centers of mass (closed dots) tend to be biased towards the direction of curvature. The average of the center of mass of adjacent neighborhoods (open dot) is about twice as far from the ideal location (closed diamond). This can be used to estimate an ideal center location.

The simple noise reduction algorithm described in the beginning of this paper made corrections only to the temporal center points of the delay vectors to avoid magnification of errors due to positive or negative Lyapunov exponents and it is advantageous to make use of a similar technique here. This is typically done by transforming the vectors using a weight matrix $R$ corresponding to a metric that penalizes corrections to the first and last elements of the delay vector[21] (eg. $R_{00} = R_{mm} = 1000$ and all other diagonal elements set to 1).[22]

Modifications can be made to this algorithm to make it suitable for real-time use. First of all, in a real-time situation only phase space vectors from the past are available. Thus only causal vectors are included in the neighborhoods and to cope with possibly changing dynamics (and for computational simplicity) only vectors that are more recent than a time $\Delta n$ are used. The cutoff $\Delta n$ is chosen so that the number of vectors in the neighborhood does not exceed a specified $U_{max}$.[23] The algorithm then proceeds as before but in the interest of speed the principle component information is reused for all

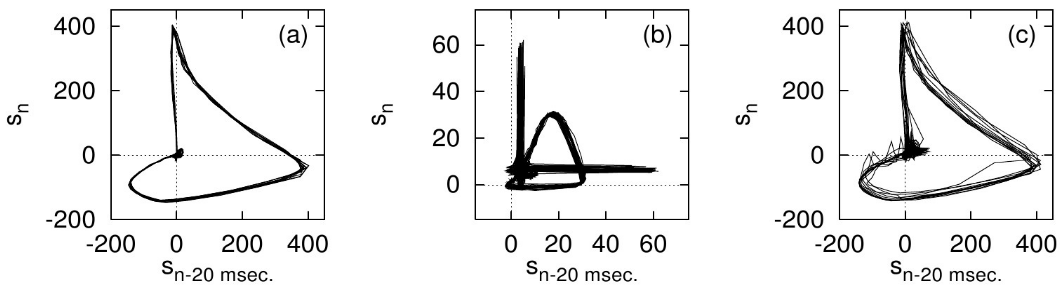
---

17 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 159
18 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 159
19 H. Kantz and T. Schreiber, *Nonlinear projective filtering I: Background in chaos theory.* (1998)
20 H. Kantz and T. Schreiber, *Nonlinear projective filtering I: Background in chaos theory.* (1998)
21 H. Kantz and T. Schreiber, *Nonlinear projective filtering I: Background in chaos theory.* (1998)
22 H. Kantz, T. Schreiber, Nonlinear Time Series Analysis (1997), page 160
23 T. Schreiber and M. Richter, *Fast nonlinear projective filtering in a data stream.* (1999)

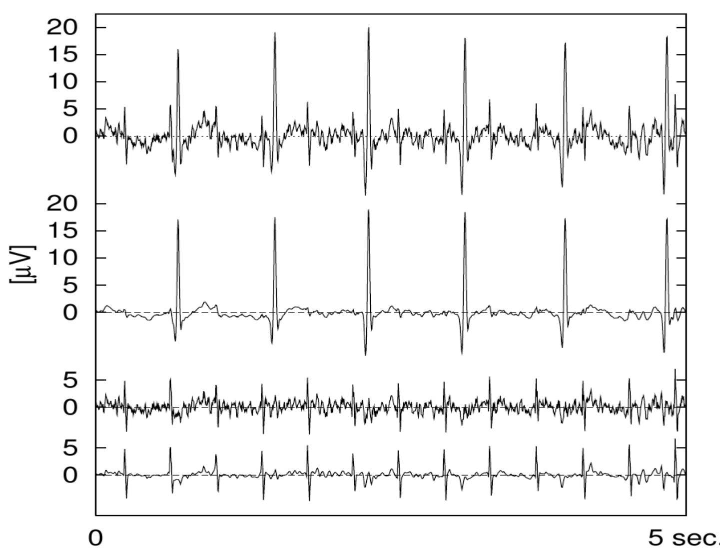
vectors in a small region of phase space.[24] Between the limited temporal range of points considered and the recycling of principle components the algorithm is fast enough that it could run three iterations on a real-time data stream sampled at 91 Hz on a computer from 1998.[25]

## Application: Extraction of Fetal ECG

The previous section dealt with removing noise from a signal but nonlinear projection can in fact be used to extract any small signal that has been mixed with an appropriate larger signal. In particular, it is possible to separate a weak fetal ECG signal that has been measured along with a mother's ECG.[26] This is something that is not at all possible using Fourier techniques because the two ECGs have nearly identical spectral components. A local projective noise reduction is done on the combined signal using a neighborhood size large enough to encompass the fetal signal and the fetal signal is removed as if it was noise. The removed signal (fetal + noise) is then cleaned to remove actual measurement noise. What remains is a fetal signal that, while not revealing its full structure, is good enough to at least determine the pulse rate.[27]




(a) An ECG signal represented in delay coordinates. (b) A small and faster ECG signal similar to a fetal component. The shape appears different because the pulse rate is faster. (c) Superposition of the two signals. The fetal signal causes a small perturbation but the shape of the manifold stays the same.[28]




Extraction technique applied to real-world data: the first line shows combined signal, the second line shows result of first application of noise reduction (only maternal component remains), the third line shows the data that was removed and consists of the fetal signal along with measurement noise, and the last line shows the result after a second application of noise reduction which gives a cleaned version of the fetal signal.[29]

24 T. Schreiber and M. Richter, *Fast nonlinear projective filtering in a data stream.* (1999)
25 T. Schreiber and M. Richter, *Fast nonlinear projective filtering in a data stream.* (1999)
26 M. Richter, T. Schreiber, D. T. Kaplan, *Fetal ECG extraction with nonlinear state-space projections. (1998)*
27 M. Richter, T. Schreiber, D. T. Kaplan, *Fetal ECG extraction with nonlinear state-space projections. (1998)*
28 M. Richter, T. Schreiber, D. T. Kaplan, *Fetal ECG extraction with nonlinear state-space projections. (1998)*
29 M. Richter, T. Schreiber, D. T. Kaplan, *Fetal ECG extraction with nonlinear state-space projections. (1998)*

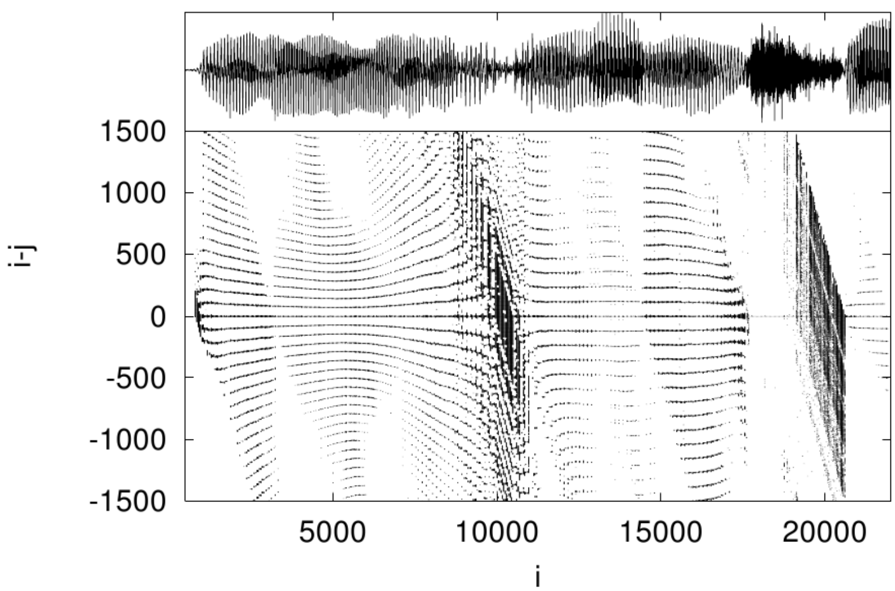
## Non-stationary Data

Locally projective noise reduction makes the assumption that the system under consideration produces a trajectory that lies on a low dimensional manifold and this is more likely to be true if the data is stationary (ie. if the parameters that define the dynamics are fixed). Most real-world systems are however not stationary. For example biological signals such as ECG or breath rate data cannot be expected to remain stationary for long periods of time because conditions in the body are constantly fluctuating. Speech is an example of a signal that remains stationary for a time (the duration of a phoneme) and then rapidly changes in character.[30]

The obvious solution is to split the time series into small segments with the hope that the data is mostly stationary during each segment. The disadvantage is that there will be less data to draw from and therefore fewer vectors in each of the neighborhoods and this will severely impact phase space methods.[31]

Another solution is to increase the embedding dimension. Instead of using $m > 2D$ for the embedding dimension, $m > 2(D + P)$ is used where $P$ is the number of non-stationary parameters. This is only mathematically correct if the $P$ parameters themselves form a stationary chaotic system but in practice it works well with only the condition that the parameters vary slowly and have only rare sudden changes.[32] In this higher dimensional embedding space the neighbors of a vector should all correspond to states with similar parameters. The advantage over the time segmentation method is the larger database of values since measurements from different time periods that have the same parameters will be available.[33]

If a transition takes place suddenly the delay vectors corresponding to the transition period will not have any neighbors in phase space. This can be used as a primitive detection of transitions as seen in the figure below which corresponds to a human voice. For each time $i$ a point is plotted at all neighboring times $j$ which occupy the same neighborhood of phase space. The stripes represent periodicity and the blank areas represent transitions between phonemes.[34]

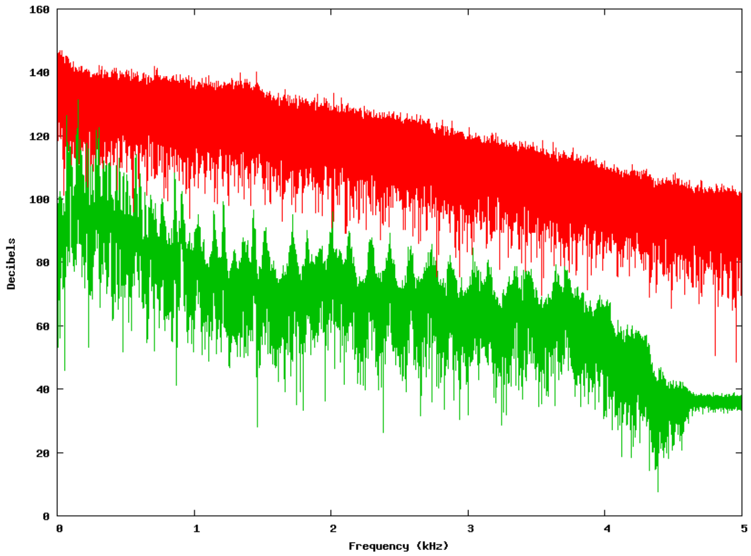



30 K. Urbanowicz, H. Kantz, Improvement of speech recognition by nonlinear noise reduction.
31 R. Hegger, H. Kantz, L. Matassini, and T. Schreiber, *Coping with nonstationarity by over-embedding. (2000)*
32 R. Hegger, H. Kantz, L. Matassini, and T. Schreiber, *Coping with nonstationarity by over-embedding. (2000)*
33 R. Hegger, H. Kantz, L. Matassini, and T. Schreiber, *Coping with nonstationarity by over-embedding. (2000)*
34 R. Hegger, H. Kantz, L. Matassini, and T. Schreiber, *Coping with nonstationarity by over-embedding. (2000)*

## Experiment: Extracting Audio From Lorenz Attractor

Kevin Cuomo and Alan Oppenheim created an electronic circuit that can obscure audio signals my mixing them with the output of an analog computer that simulates the Lorenz system. The receiving end consists of another Lorenz system which is designed to fall into synchrony with the transmitter. The synchronized Lorenz signal is then subtracted out and the original audio signal is recovered - fuzzy, but recognizable.[35] I wanted to see if the decoding could also be done using a technique similar to what was used to extract the fetal ECG signal.

100,000 samples of a Lorenz trajectory were generated using the parameters $\sigma = 10, \beta = \frac{8}{3}, \rho = 28,$ and $\Delta t = 0.1.$ The first 10 seconds of Beethoven's 5th where then recorded at 10,000 samples/sec and scaled to be approximately 40dB quieter (100 times smaller numerically) than the Lorenz data. The continuous spectrum of the Lorenz system completely obscures the audio signal and renders any sort of Fourier technique ineffective.




Spectrum of Lorenz signal (top) and audio signal (bottom). The audio signal is approximately 40dB quieter (100 times smaller numerically) and is therefore completely obscured when added to the Lorenz signal.

To decode the data I used the *project* command from the TISEAN package[36] which implements nonlinear projective noise reduction as described in this paper. I used an embedding dimension of 15 and projected onto a manifold of dimension 3, iterating 10 times. The cleaned signal then represented the Lorenz system with the Beethoven signal removed. Subtracting this from the mixed Lorenz+Beethoven signal then produced the original Beethoven, with some distortion. The recovered audio didn't sound too good but it was amazing that it was recovered at all – the Lorenz+Beethoven signal subjectively sounds completely identical to the raw Lorenz signal.

The recovered audio stream has a loud crackling and sounds like it is being played off of a badly scratched record. My guess is that the noise reduction does not perform optimally when the Lorenz trajectory skips from one lobe to another of the attractor, leading to a "pop" sound whenever this happens. Perhaps as the trajectory transfers to the other lobe and gets injected back into the saddle point the attractor is less planar than it is elsewhere. This could possibly interfere with the principle component analysis in these regions. The dimension of the attractor is approximately 2.06[37] so Taken's theorem demands an embedding dimension of at least 5. It is possible that 3 dimensions are sufficient everywhere except for the places where the two lobes connect to each other. I tried using a variety of

35 S. Strogatz, Nonlinear Dynamics and Chaos. (1994)
36 http://www.mpipks-dresden.mpg.de/~tisean
37 http://en.wikipedia.org/wiki/Lorenz_attractor

embedding dimensions and manifold dimensions and found the best results to be an embedding dimension of 15 (the largest allowed by the program) and a projection dimension of 3.

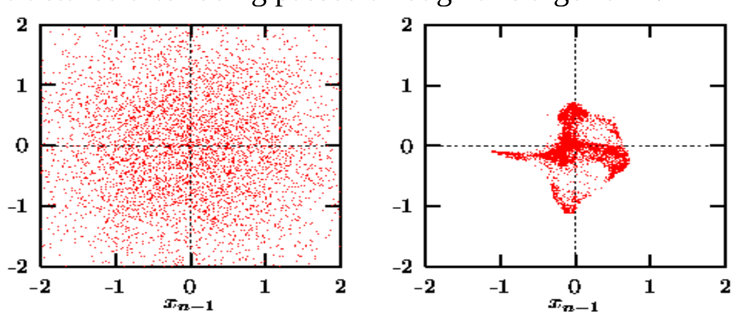
It occurs to me that the projective filter is not completely optimal for fractional dimensional systems. For example, the direct product of three Cantor sets has a dimension less than 2.0, but the Euclidean dimension which is relevant to the principle component analysis is 3, a whole integer value too high. This could have possible consequences in the case of the Lorenz system whose Hausdorff dimension is so close to 2 but probably can be nowhere represented in a space of Euclidean dimension less than 3. It is not obvious to me how the algorithm could be modified to account for this.

## Conclusion

Nonlinear projective noise reduction can potentially be a powerful technique but it certainly is no panacea. When a data set comes from a low dimensional, stationary, deterministic chaotic system the algorithm works like a charm even if the noise level is quite high whereas linear filters would be ineffective. If the data to be processed consists of both deterministic and stochastic components (such as ECG signals) the noise reduction is still useful and even if the system is non-stationary the embedding dimension can be raised as long as the non-stationary parameters vary slowly.

Some amount of trial and error is involved in choosing the parameters such as embedding dimension, manifold (projection) dimension, and neighborhood size. If the box counting dimension of the attractor is known then the theorem by Takens and Sauer states that a minimum embedding dimension is $m > 2D$, but there are some practical considerations that need to be taken into account such as possible non-stationarity of the system and ensuring that different parts of the attractor are separated by at least the neighborhood size, which becomes a concern when noise is high. Choosing an embedding dimension that is too high is usually safe and can aid separation in situations of high noise levels but adds to computational complexity in the search for neighboring points and computation of principal components. The dimension of the subspace that the algorithm projects to is even less clear cut. It would seem that the box counting dimension rounded up to the next integer value would be good here but in practice the optimal value depends upon the particular situation.[38] The neighborhood size must be larger than the noise level and in fact the type of noise to be removed can be chosen by varying the neighborhood size. In this way the nonlinear technique can be seen as selecting a signal based upon amplitude as opposed to Fourier techniques which select based upon frequency.[39]

Care needs to be taken when interpreting the results of noise reduction – even Gaussian noise will be made to look structured after being passed through this algorithm:




Gaussian noise, before and after noise reduction[40]

It is therefore important to test the data for determinism or to check whether the results are different compared to a surrogate data set that is randomly generated with the same statistical properties as the

38 http://www.mpipks-dresden.mpg.de/~tisean/Tisean_3.0.1/docs/chaospaper/node24.html

39 http://www.mpipks-dresden.mpg.de/~tisean/Tisean_3.0.1/docs/chaospaper/node24.html

40 http://www.mpipks-dresden.mpg.de/~tisean/Tisean_3.0.1/docs/chaospaper/node24.html

real data.[41]

The extraction of the fetal ECG signal is an impressive example but I think that it represents a best case scenario. I downloaded a sample data set from the PhysioBank database[42] and tried to reproduce the results but it didn't work because the fetal ECG was orders of magnitude weaker than in the textbook example. This is probably a function of electrode placement. It is impressive that the nonlinear projective technique is able to perform this task with no programmed knowledge about the system being monitored but for optimal results very specialized techniques can be used. A quick search on the Internet will reveal descriptions of systems built of many modules usually including neural networks. Still, it says a lot that one algorithm can extract a signal from both an ECG and from a Lorenz system with no specialized modifications needed in either case.

41 http://www.mpipks-dresden.mpg.de/~tisean/Tisean_3.0.1/docs/surropaper/Surrogates.html
42 http://www.physionet.org/physiobank

## Works Cited


R. Hegger, H. Kantz, L. Matassini, and T. Schreiber
*Coping with nonstationarity by over-embedding*
Phys. Rev. Lett. **84** (2000) 4092.

Krzysztof Urbanowicz, Holger Kantz
*Improvement of speech recognition by nonlinear noise reduction*
arXiv:physics/0507044v5

M. Richter, T. Schreiber, and D. T. Kaplan
*Fetal ECG extraction with nonlinear state-space projections*
IEEE Trans. Bio-Med. Eng. **45**, 133 (1998)

T. Schreiber and M. Richter
*Fast nonlinear projective filtering in a data stream*
Int. J. Bifurcat. Chaos **9**, 2039 (1999)
chao-dyn/9803008

H. Kantz and T. Schreiber
*Nonlinear projective filtering I: Background in chaos theory*
in Proceedings of NOLTA 1998, Presses Polytechniques et Universitaires Romandes, Lausanne (1998)
chao-dyn/9805024

T. Schreiber and H. Kantz
*Nonlinear projective filtering II: Application to real time series*
in Proceedings of NOLTA 1998, Presses Polytechniques et Universitaires Romandes, Lausanne (1998)
chao-dyn/9805025

Rainer Hegger and Holger Kantz
*Embedding of sequences of time intervals*
in Europhys. Lett. **38** 267-272 (1997)

Kantz, Schreiber
*Nonlinear Time Series Analysis*
Cambridge University Press (1997)

S. Strogatz
*Nonlinear Dynamics and Chaos*
Perseus Books Publishing (1994)

The TISEAN software package
http://www.mpipks-dresden.mpg.de/~tisean

Wikipedia: Lorenz Attractor
http://en.wikipedia.org/wiki/Lorenz_attractor