Chapter 15

Surrogate Data Analysis

In many practical situations we may want to classify or characterize our time series with few numbers (e.g., fractal dimension, Lyapunov exponent, model prediction error, or some entropy). These characteristic numbers then can be used for determining if the underlying dynamical systems is operating as expected or needs an attention. They could also be used as measures of some underlying hidden processes that may evolve on a slower time scales than the observed dynamics. Some examples include damage identification and monitoring, fault detection and localization, medical diagnosis, and etc. In all these cases we need to have trust in the estimated metrics, are they really reflecting the underlying dynamics or just giving us some numbers as a result of algorithmic calculations that have no physical meaning?

Many times we do not have a strong indication or evidence that what we are estimating can be trusted or is actually measure of some deterministic process. In this situations, we need to evaluate the probability that our measures are reflecting the physical reality by statistical hypothesis testing. However, due to the limited size of the experimental data we usually are limited in the ability to estimate the probability distribution of our metrics using, for example, different subsets of our data. Furthermore, it is usually fairly difficult to come up with simple null hypothesis for testing the complicated metrics we are using here. Due to all this, we usually resort to surrogate data analysis.

The main idea behind the surrogate data analysis is that we want to compare our particular nonlinear metric estimated from the available time series to the distribution of the same metric obtained from a large number of time series (i.e, surrogate data) that satisfying some null hypothesis. This null hypothesis could be that the time series are generated by some linear stochastic process (i.e., autoregressive moving average or ARMA). Then, we can estimate the probability that our metric is real or obtained purely by chance by observing if and where it falls in the obtained distribution obtained from the surrogates. Then, if the probability of it being true is high enough we can reject the null hypothesis.
The confidence level with which the null hypothesis can be rejected will depend on the number of surrogate signals used. We allow for a chance that null hypothesis could be rejected even if it is true. For example, if we require a 90% confidence level, we accept a 10% chance that the null hypothesis could be falsely rejected. To quantify this, we assign a value of 0.10 (10%) to the parameter $\alpha$.

The number of surrogate data sets required is then determined by $\alpha$. Statistics used to differentiate metrics, can be either single sided or double sided. Depending on which the number of surrogates, $\beta$, will vary according to the following rules:

**Single Sided Statistic:** We expect our data statistic to either be always higher or lower than our statistic measured from the surrogates:

$$\beta = \frac{1}{\alpha} - 1$$

**Double Sided Statistic:** We expect our surrogate statistic to be higher or lower than our data statistic:

$$\beta = \frac{2}{\alpha} - 1$$

For this method to be effective our surrogate data needs to have some properties that are identical or similar to our time series to establish the corresponding null hypothesis. Since, we are mainly interested in testing for nonlinear metrics, we usually require that all or some of the linear properties of the test time series and surrogates be shared. This would support a powerful null hypothesis that the time series are generated by the ARMA model. Generally, we could consider these three null hypothesis or their combinations:

1. The data is composed of random numbers drawn from some fixed, but unknown distribution.
2. The data is drawn from stationary linear stochastic process with Gaussian inputs.
3. The data is drawn from a stationary linear stochastic process with unknown distribution.

Each null hypothesis have a method of creating surrogate data sets associated with it. Each surrogate needs to have the same number of points as the original signal for the test to be valid. The three basic methods surrogate data generation methods are:

**Unconstrained randomization of time series:** This is a very simple form of surrogate data set. We will simply shuffle the time series in some random order.

**Phase randomized surrogates:** Here we require that the surrogates have the same power spectrum as our data, which is equivalent of requiring to having identical linear correlations. Hence, if there are nonlinear correlations in our test data, we should be able to reject the hypothesis. This method will preserve the power spectrum amplitudes of the time series while randomizing its phase. In particular,
1. Taking the digital fast Fourier transform (DFFT) of our measured time series \( \{ x_n \}_{n=1}^N \):

\[
X_n = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} x_k e^{\frac{2\pi i n k}{N}},
\]

(15.1)

2. Replace the complex components of \( X_n \) by random phases, \( X'_n = |X_n| e^{i\phi_n} \), where \( \phi_n \) are uniformly distributed in \([0, 2\pi]\).

3. Compute an inverse digital fast Fourier transform (IDFFT) to get the surrogate

\[
x'_n = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} X'_k e^{-\frac{2\pi i n k}{N}}.
\]

(15.2)

Iteratively refined surrogates: This is using the iterated amplitude adjusted Fourier transformed surrogates method as described in Ref [63]. This method preserves all the linear statistical and spectral properties in the surrogates (i.e., power spectrum, mean, variance, autocorrelation, probability distribution, etc.). However, any nonlinear structures in the data will be destroyed by the randomization procedure, which uses two step iterative method.

1. We start by sorting the original time series \( \{ x_n \}_{n=0}^{N-1} \) by its magnitude in ascending order into \( \{ y_n \}_{n=0}^{N-1} \) and storing it with the corresponding Fourier transform magnitudes:

\[
X_n = \frac{1}{N} \left| \sum_{k=0}^{N-1} x_k e^{\frac{2\pi i n k}{N}} \right|.
\]

(15.3)

2. The iterative process is initiated by randomly reshuffling \( \{ x_n \} \) to get \( \{ r_n^{(0)} \} \):

3. Take DFFT of \( \{ r_n^{(i)} \}_{n=0}^{N-1} \) to get:

\[
R_n^{(i)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} r_k^{(i)} e^{\frac{2\pi i n k}{N}}.
\]

(15.4)

4. Now replace the magnitudes \( |R_n^{(i)}| \) with \( X_n \), but keep the phase \( \psi_n^{(i)} = \arg R_n^{(i)} \) the same and take IDFFT to get

\[
x_n^{(i)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k e^{\psi_k^{(i)}} e^{-\frac{2\pi i n k}{N}}.
\]

(15.5)

5. Finally, transform \( \{ x_n^{(i)} \} \) into \( \{ r_n^{(i+1)} \} \) by rank-ordering according to stored \( \{ y_n \} \):

\[
r_n^{(i+1)} = y_{\text{rank}} \left( x_n^{(i)} \right).
\]

(15.6)

The idea is that iteratively we are converging to a fixed point \( r_n^{(i+1)} = r_n^{(i)} \) for large \( i \), which is usually reached after finite number of iterations.

15.1 Examples

To investigate the usefulness and abilities of this method we are going to look at two separate time series. The first time series will be a simple linear stochastic process, and the second will come from a simulation of the Lorenz attractor.
1.1.1 Example

The linear process can be described by the following:

$$x_n = 0.99x_{n-1} + \eta_n,$$  \hspace{1cm} (15.7)

where are \(\eta_n\) independent random numbers. To make this problem more interesting for our application, we will observe the process through a nonlinear function,

$$s_n = x_n^3.$$  \hspace{1cm} (15.8)

It should also be noted that the mean of the signal is removed prior to the nonlinear transformation. The process along with the nonlinear observation can be seen in Fig. 15.1 (left). It is very possible to see a signal like this in real life. Many times, especially in biological systems, we do not know the governing equations and processes of a system. When we take some measurement of that system there could easily be a nonlinear rescaling. In this case it would be inappropriate to analyze the time series using nonlinear time series analysis techniques. For the sake of this problem we are going to assume we know nothing about the underlying process and only see the nonlinear observation.

The first approach will be to test the null hypothesis (number 1) that the data is of a random process. The statistic used will be the autocorrelation at lag one, and the surrogate method will be unconstrained randomization of the time series.

We want to pass every test in this paper with a 99 % confidence level. This requires that \(\alpha = 0.01\), and therefore, \(\beta = \frac{1}{\alpha/M} - 1 = 99\) surrogates for single sided tests and \(\beta = \frac{2}{\alpha/M} - 1 = 199\) surrogates for double sided tests. The autocorrelation is expected to be much lower for a random signal than a linear stochastic signal, so this test will be single sided requiring 99 surrogates. Fig. 15.1 (right) shows the results of the statistic test. On the horizontal axis is the value of the autocorrelation and
Figure 15.2: PSD of the original time series and one surrogate data set (left), and autocorrelation of 99 phase randomized surrogates plotted along with the autocorrelation of the data (right).

the vertical axis is for the purpose of illustration only. All surrogates are given a value of 1 and the data is given a value of 1.5 for ease of picking out the differences. If the autocorrelation of the data was close to the range of the distribution of the surrogates it would be plotted here as well. However it is obvious that the autocorrelation of all surrogates is much lower than that of the data. The null hypothesis is then rejected with 99 % confidence level and we can declare that the observation is not a random distribution of numbers.

This test does not tell us about the nonlinearity of the system. The next null hypothesis we can take is that the time series is a stationary linear stochastic process with Gaussian inputs. Again we will use the autocorrelation as the statistic with 99 surrogates. The surrogates will be generated using the phase randomized process. The power spectrum is well preserved as can be seen in the PSD plot of Fig. 15.2 (left).

The autocorrelation is then computed for the time series and 99 of its surrogates and plotted in Fig. 15.2 (right). From this test the null hypothesis cannot be rejected and the assumption that this process is actually a linear stochastic process with Gaussian inputs can hold. On the other hand, the statistic could also not be powerful enough to distinguish the nonlinearity.

The final null hypothesis taken here is that the data is drawn from a linear stochastic process with unknown distribution. For this test we will be using the error of a one time-step-ahead prediction as our statistic. This test is double sided and will require 199 surrogates. The surrogate sets will be generated using the iteratively refined (polished) randomization scheme. Here the distribution and the power spectrum are preserved as described earlier. Fig. 15.3 (left) shows the PSDs of the signal and one surrogate. The two match fairly well together. This method will not produce perfect matching by nature, but can be refined with an increase of iterations. Here, the iterations were kept relatively low for the sake of computational time. The series and surrogates are embedded in three
dimensional space with unit delay.

Figure 15.3: PSD of signal and one polished surrogate (left) and prediction error (right).

Fig. 15.3 (right) shows that the null hypothesis cannot be rejected with 99% confidence here either. We can now reasonably assume that the data comes from a linear stochastic process, which we know is true.

15.1.2 Lorenz Time Series Example

The surrogates are generated using the refined iterative randomization method. Fig. 15.4 shows the results of the randomization. The preservation characteristics can be seen in Fig. 15.5 showing the

Figure 15.4: Lorenz data (top), surrogate (bottom)
PSD and histogram of the data and one surrogate. Noise is also added to the time series, \( s_n = x_n + \eta_n \), to make the problem more realistic. All signals are embedded using standard embedding parameters and the prediction algorithm is run identically. Fig. 15.6 shows the results of the statistical test.

![Histogram of Signal vs Surrogate](image1)

![Power Spectrum Magnitude](image2)

Figure 15.5: Comparison of distribution and PSD of Lorenz signal (left) and one surrogate (right)

Clearly, the null hypothesis can be rejected in every case giving us a 99 \% confidence level that this signal is not a linear stochastic process measured through a nonlinear function.

![Prediction Error](image3)

Figure 15.6: Lorenz attractor time series prediction error versus 199 surrogates