Data Analysis Lab

The purpose of this lab is to provide an insider look in the field of Data Analysis, using tools such as Computing Cluster to work with large sets of experimental data, such as gravitational wave data. The main goal of the lab is to detect a signal injected in Gaussian noise. For this purpose the student will have to make usage of computing resources. The lab yields to a basic understanding of:

- time series data with stationary noise and signals
- statistical concepts (moments, median, higher moments)
- template placement, mismatch statistics, ROC curve
- sensitivity computational cost analysis
- paralelization with a computing cluster using Condor

The practical work is divided into four exercises. A guide complemented with useful links and documentation is given. In order to proceed to the next exercise, the student is asked to demonstrate solutions to short set of problems.

The lab is targeted to Bachelor or Master students in Physics. The student should have knowledge of:

- C programming language
- scripting language (Python is particularly useful)
- basic statistical working knowledge (mean, variance, higher moments), time series analysis

Exercise 1: Detecting a signal in noise

A detector records data $d(t)$ which consist of noise $n(t)$ and, possibly, a signal $s(t)$. The noise can lead to two unwanted results. A *false positive* result is a detection of a signal which is not present in the noise. A *false negative* result is non-detected signal. We will simulate detector data using Gaussian noise. Two hypothesis are possible. Either no signal is present in the data or the data contain a signal:

$$\begin{aligned} H_0 &: d(t) = n(t) \\ H_1 &: d(t) = n(t) + s(t). \end{aligned}$$

For the signal $s(t)$ we use a simple model:

$$s(n) = A\sin(n\Delta t f + \phi), \tag{1}$$

where the amplitude A, the phase ϕ and the frequency f are unknown parameters and Δt is the time difference between two adjacent signal records.

We would like to introduce and compare three different analysis methods. The first method is a scalar product of a signal with particular parameters and the data set:

$$S_1 = \frac{1}{N}\sum_{n=0}^{N-1} d(n)s(n, f, \phi), \tag{2}$$

where the test function $s(n, f, \phi)$ is normalised. The second test is the so-called χ-*square* test:

$$S_2 = -\frac{1}{N}\sum_{n=0}^{N-1}(d(n) - s(n, f, \phi))^2, \tag{3}$$

and in the third test we compute the power spectrum of the data:

$$S_3 = \frac{1}{N}\left|\sum_{n=0}^{N-1} d(n)e^{i\Delta t f 2\pi + \phi}\right|^2. \tag{4}$$

We try to find the parameters f and ϕ which minimise S_1, S_2 or S_3. We introduced a minus in equation 3 to search for a minimum of S_2. It is easy to see that S_3 is independent of ϕ. We define limits and find candidates for a signal if $S_l(f_i, \phi_i) < \gamma_l$ for the methods $l = \{1, 2, 3\}$. In figure 1 the two distributions are plotted, the blue curve is the probability distribution of pure noise for each value of S_l, meanwhile the green one is the probability distribution of signal plus noise for each value of S_l. If the signal is too faint and the γ_l is too small we will not detect this signal. In this case we do have a false negative result, green region in figure 1. If γ_l is too big the analysis of (signal-less) noisy data yield signal candidates. We do have a false positive detection, blue area in figure 1.

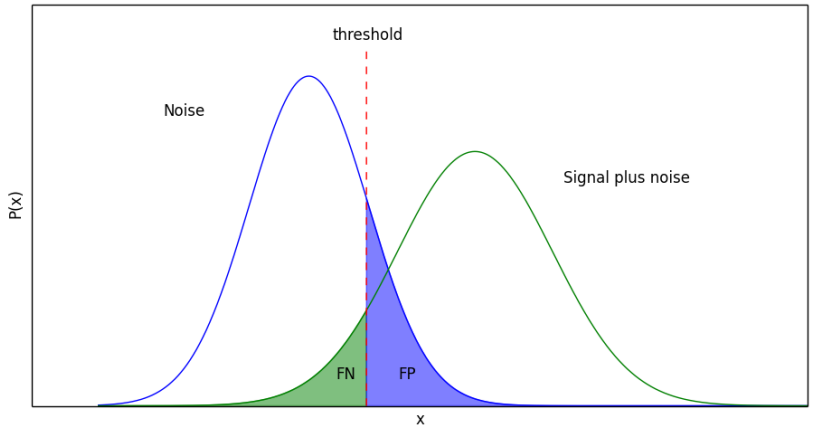

Figure 1: Probability of x for both hypothesis, noise alone and signal plus noise. Being x the value of S_l for a given parameter point.

To compute the threshold we will use the Neyman-Pearson criteria. Since normally we will deal with faint signals we will chose a low false negative ratio, in order not to lose any signal, and use it to compute γ. Normally the true positive rate is plotted versus the false positive rate on the well-known ROC curves (Figure 2). It is a plot that shows the performance of a detector by plotting its probability of detection (true positive rate) versus the probability of false alarm at various threshold settings. It should be always above the 45 ° line labelled as C, since it would be attained by a detector which, ignoring all data, bases its decision on flipping a coin. As opposed to the perfect detector whose ROC curve looks like line A. Line B would be an example of a non perfect detector with a good performance. We pretend to be in a region close to the red point, achieving then a high true positive rate with the lowest possible false positive rate.

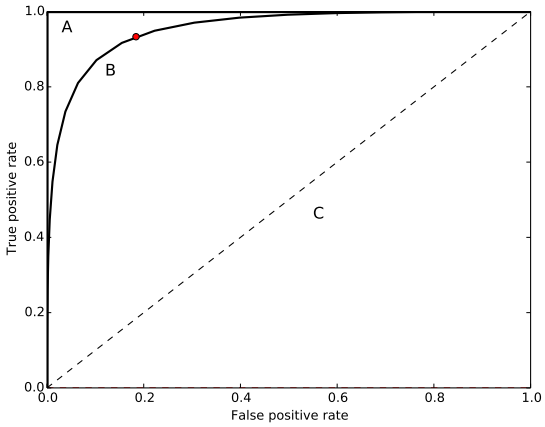

Figure 2: Example of a ROC curve.

Properties of Gaussian noise

We will deal with Gaussian noise, that is noise having a probability density function (PDF) equal to that of the normal distribution. It will be also white, that means noise with a uniform power across all frequencies. One of the most important properties of Gaussian noise for our purposes is that the sum of Gaussian random variables is also a Gaussian random variable, with its mean being the sum of the means, and its variance being the sum of the variances. Then the square of the standard deviation is the sum of the squares of the standard deviations. It is easy to proof it using the Fourier transform of the normal distribution, also known as characteristic function, with expected value μ and variance σ^2:

$$\phi(t) = e^{(\mathbf{i}t\mu - \frac{\sigma^2 t^2}{2})} \tag{5}$$

Summing two different random variables with normal distributions:

$$\phi_{X+Y}(t) = \phi_X(t)\phi_Y(t) = e^{(it\mu_X - \frac{\sigma_X^2 t^2}{2})} e^{(it\mu_Y - \frac{\sigma_Y^2 t^2}{2})} = e^{(it(\mu_X+\mu_Y) - \frac{(\sigma_X^2+\sigma_Y^2)t^2}{2})} \tag{6}$$

Thus if we increase by four the number of data points the standard deviation is multiplied by two, the computational cost will increase also by four but the level of signal compared to the level of noise will grow only by two since the power of the signal increases linearly with the number of data points, but the noise increases as the root of the number of data points. Thus the signal-to-noise ratio (SNR), which is the ratio of the mean of the data to the standard deviation of the noise, will increase as $\frac{N}{\sqrt{N}}$. That is, as the root of the number of data points.

Scalar product method in depth

Another way to name the scalar product method is the maximum likelihood method, based on Bayesian statistics. Since it is important for gravitational wave detection we will introduce the basis here. First, we will define some basic probability concepts such as conditional probability, that is the probability that a given event A is true given that another event B is true:

$$P(A|B) = \frac{P(A,B)}{P(B)} \tag{7}$$

where $P(A,B)$ is the joint probability of A and B being true at the same time, and then it is the product of $P(A)$ and $P(B)$. Combining $P(A|B)$ and $P(B|A)$ we obtain the Bayes theorem:

$$P(B|A) = \frac{P(B)P(A|B)}{P(A)} \tag{8}$$

Furthermore, introducing the completeness relationship:

$$P(A) = P(A|B)P(B) = P(A|\neg B)P(\neg B) \tag{9}$$

in equation 8:

$$P(B|A) = \frac{P(B)P(A|B)}{P(A|B)P(B) + P(A|\neg B)P(\neg B)} = \frac{\Lambda(B|A)}{\Lambda(B|A) + \frac{P(\neg B)}{P(B)}} \tag{10}$$

where we define the likelihood ratio as:

$$\Lambda(B|A) = \frac{P(A|B)}{P(A|\neg B)} \tag{11}$$

Turning to the two hypothesis of the beginning (signal is present on the data, or data consist only of noise):

$$\Lambda(H_1|d) = \frac{P(d|H_1)}{P(d|H_0)} \tag{12}$$

Then, the hypothesis H_1 will be true for our data if Λ exceeds a certain threshold. Now, we will focus on how to define this threshold and compute Λ.

If noise is Gaussian, we can compute the probability densities under the different hypothesis. If H_0:

$$p(d|H_0) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{\frac{-1}{2\sigma^2}\sum_{n=0}^{N-1} d^2(n)} \tag{13}$$

While under H_1, since $n = d - s$:

$$p(d|H_1) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{\frac{-1}{2\sigma^2}\sum_{n=0}^{N-1} [d(n)-s(n)]^2} \tag{14}$$

Thus the logarithm of the likelihood ratio is:

$$ln(\Lambda) = \frac{-1}{2\sigma^2}\left(\sum_{n=0}^{N-1}[d(n) - s(n)]^2 - d^2(n)\right) \tag{15}$$

Since s(n) is data independent, we can put it on the other side of the equation to define the threshold:

$$S_0 = \sum_{n=0}^{N-1} d(n)s(n) > \gamma \tag{16}$$

And S_0 will have a maximum at the right parameters of the signal.

Goal

Understand how to detect a signal embedded in noise and learn how to use *prober*.

Guidance

1. Plot various realisations of signal plus noise. Use different signal amplitudes with *generate_source* until you can not distinguish the signal because it is too faint. To learn how to use it *generate_source --help*.

2. Learn to use *prober*, our small black box program computing a scalar result based upon input time series and parameter point (f, φ). For more information use *./prober --help*

3. Learn how *prober* results change, varying the parameters of an injected signal while probing the same parameter point.

4. Plot/document result findings: how large may the mismatch between injection and probe parameter may become? What happens with weak signals (small amplitude)?

5. Develop an strategy to find signal with *prober* if the signal parameters (A, f, φ) are unknown.

Recommended literature

- Percival, Walden: "Spectral Analysis for Physical Applications : Multitaper and Conventional Univariate Techniques" [1]
- Porat "A Course in Digital Signal Processing" [2]

[1] http://www.amazon.de/Spectral-Analysis-Physical-Applications-Conventional/dp/0521435412
[2] http://www.amazon.de/Digital-Signal-Processing-Electrical-Electronics/dp/0471149616