# Lab Topic 4

# Learning objectives

1. Normalize a given signal vector such that the Likelihood Ratio (LR) test (Binary hypothesis) for stationary Gaussian noise with a specified PSD has a given Signal to Noise Ratio (SNR)
   - We often shorten this statement to say: "Normalize the *signal* to have a given SNR."
2. Generate a set of data realizations under the null and alternative hypotheses and estimate the SNR of the LR test to (a) understand the meaning of SNR, and (b) verify that the signal normalization was done correctly
3. Compute the GLRT when only the signal amplitude is unknown
4. Obtain the significance of the GLRT for given test data

# Codes

- Generating stationary Gaussian noise: **NOISE/statgaussnoisegen.m**
  - See script **NOISE/colGaussNoiseDemo.m** for how to use this function
- Taking an inner product: **DETEST/innerprodpsd.m**
- Calculating the normalization factor for a given SNR and noise: **DETEST/normsig4psd.m**
- Demo for estimating the SNR of the LR test (binary hypotheses) from multiple data realizations: **DETEST/SNRcalc.m**
- Demo for calculating the GLRT when only the amplitude is unknown: **DETEST/GLRTcalc.m**

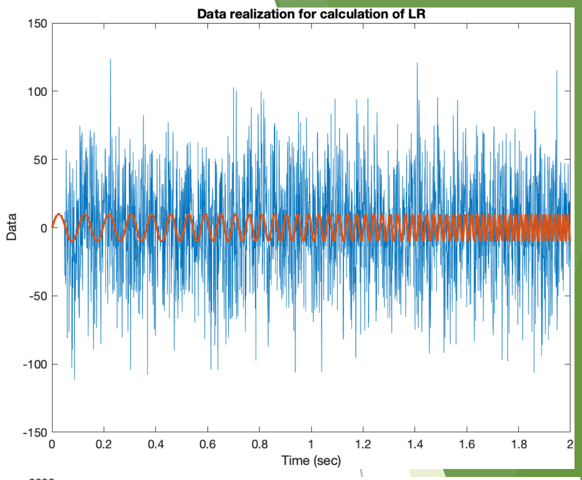
$$L_G = \max_{\Theta} \left\langle \bar{y}, \bar{q}\left(\underbrace{\Theta}_{known}\right)\right\rangle^2 \rightarrow L_G = \langle \bar{y}, \bar{q}(\Theta)\rangle^2$$

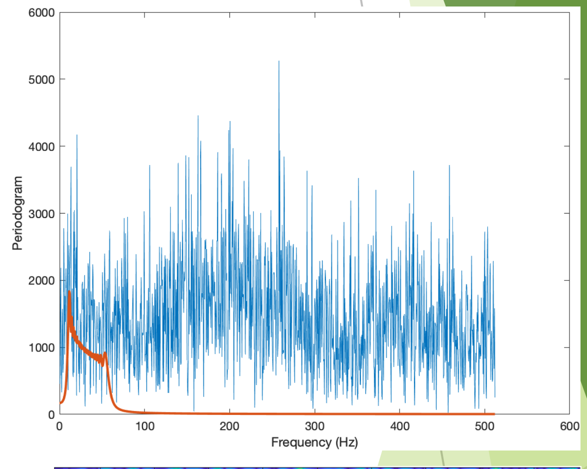
# Tasks: Set 1

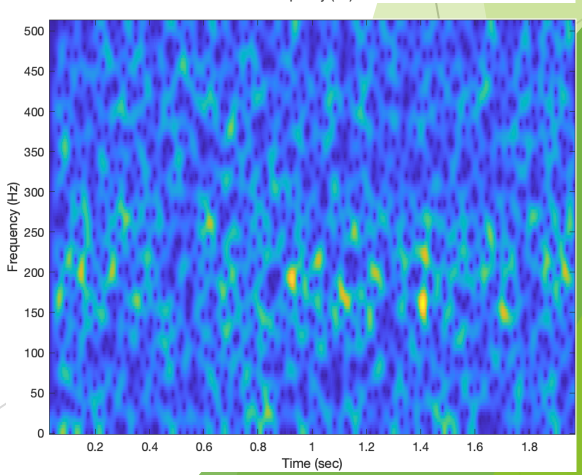
- Copy **DETEST/SNRcalc.m** to a new script **SNRcalcMod1.m** and modify it to use your assigned signal
- Plot the data (signal+noise) realization and the signal in the time domain
- Plot the periodogram of the signal and data
- Plot the spectrogram of the data

**Parts of SNRcalc.m to be modified**

```
%%
% Generate the signal that is to be normalized
a1=10;
a2=3;
a3=3;
% Amplitude value does not matter as it will be changed in the normalization
A = 1;
sigVec = crcbgenqcsig(timeVec,1,[a1,a2,a3]);
```



# Tasks: Set 2

- Copy **DETEST/SNRcalc.m** to a new script **SNRcalcMod2.m** and modify it to work with your assigned signal and the initial LIGO design sensitivity PSD (review earlier labs)
  - Learn how to use the 'interp1' function in Matlab and interpolate the PSD given in NOISE/ iLIGOSensitivity.txt, which are at irregularly spaced frequencies, to the required DFT frequencies
- Run the script to estimate the SNR of the LR test
- Does the estimated SNR come close to the one you had normalized the signal with? (It should!)

**Parts of SNRcalc.m to be modified**

```
%%
% Generate the signal that is to be normalized
a1=10;
a2=3;
a3=3;
% Amplitude value does not matter as it will be changed in the normalization
A = 1;
sigVec = crcbgenqcsig(timeVec,1,[a1,a2,a3]);
```

```
%%
% We will use the noise PSD used in colGaussNoiseDemo.m but add a constant
% to remove the parts that are zero. (Exercise: Prove that if the noise PSD
% is zero at some frequencies but the signal added to the noise is not,
% then one can create a detection statistic with infinite SNR.)
noisePSD = @(f) (f>=100 & f<=300).*(f-100).*(300-f)/10000 + 1;
```

# Tasks: Set 3

- Follow the demo script **DETEST/GLRTcalc.m** to write a function **glrtqcsig.m** that calculates the GLRT for a quadratic chirp signal with unknown amplitude

$$L_G = \langle \bar{y}, \bar{q}(\Theta) \rangle^2 \text{ where } \bar{q}(\Theta) = \frac{\bar{s}(\Theta)}{\|\bar{s}(\Theta)\|}$$

  - Inputs:
    - Data vector $\bar{y}$
    - PSD vector (for positive DFT frequencies)
    - vector of input parameters $a_1,\ a_2,\ a_3$
  - Output:
    - GLRT value

```
%% Compute GLRT
%Generate the unit norm signal (i.e., template). Here, the value used for
%'A' does not matter because we are going to normalize the signal anyway.
%Note: the GLRT here is for the unknown amplitude case, that is all other
%signal parameters are known
sigVec = crcbgenqcsig(timeVec,A,[a1,a2,a3]);
%We do not need the normalization factor, just the  template vector
[templateVec,~] = normsig4psd(sigVec,sampFreq,psdPosFreq,1);
% Calculate inner product of data with template
llr = innerprodpsd(dataVec,templateVec,sampFreq,psdPosFreq);
%GLRT is its square
llr = llr^2;
```

# Estimating significance

- Significance $\alpha$ of the observed value $\Gamma^{(observed)}$ of a detection statistic $\Gamma$

$$\alpha = \Pr\left\{\Gamma \geq \Gamma^{(observed)} \middle| H_0\right\}$$

- To estimate the significance, draw a large number of trial values of $\Gamma$ under $H_0$
  - Trial value $\Rightarrow$ generate a data realization and compute the value of $\Gamma$ for it
- Then,

$$\Pr\left\{\Gamma \geq \Gamma^{(observed)} \middle| H_0\right\} \approx \frac{Number\ of\ times\ \Gamma \geq \Gamma^{(observed)}}{Total\ number\ of\ data\ realizations}$$

- The approximation above becomes better as the total number of trials increases

# Tasks: Set 4

- You have been provided 3 data realizations named **DETEST/data<n>.txt** where $n = 1, 2, 3$
- Each realization is a time series with sampling frequency of 1024 Hz.
- The signal in each data realization is a quadratic chirp but the amplitude is unknown (could be zero also!)
  - The signal parameters are: a1=10; a2=3; a3=3;
- The noise PSD used is the one in **DETEST/SNRCalc.m**

```
noisePSD = @(f) (f>=100 & f<=300).*(f-100).*(300-f)/10000 + 1;
```

- Use **glrtqcsig.m** to calculate the GLRT for each of the 3 data realizations
- Generate M data realizations under $H_0$ (signal absent) and use the associated GLRT values to estimate the significance of each of the 3 GLRT values for the 3 given data realizations
  - Keep increasing M until the obtained significance stabilizes
  - If your code is slow, think about how to speed it up
  - Learn to use the ‘profile’ command in Matlab to examine which parts of your code are consuming the most time and what you can do to speed it up

# Conclusion

- The function **glrtqcsig.m** will serve as the fitness function that will be optimized by PSO to obtain the GLRT
$$L_G = \max_{\Theta} \langle \bar{y}, \bar{q}(\Theta) \rangle^2$$
- This optimization problem and the results associated with it are fully discussed in the textbook
- The main difference is that we have generalized the textbook problem to the case of colored Gaussian noise