
Matching Filter for event detection

Report

UNIVERSITY OF CHILE
FACULTY OF PHYSICAL AND MATHEMATICAL SCIENCE

AUTHOR: FRANCISCO MUÑOZ
PROFESSORS: FRANCISCO FÖRSTER
PABLO HUIJSE

LAST UPDATE: MONDAY 4TH JUNE, 2018
SANTIAGO, CHILE

Abstract

Por definir.

Table of Contents

1. Introduction	1
2. Fourier Transform	1
2.1. Properties of the Fourier Transform	1
2.2. Some useful Fourier Transforms	2
2.3. Window Function	3
2.3.1. Fixing periodicity	3
2.4. Nyquist-Shannon Sampling Theorem	4
2.5. Nyquist frequency	4
2.6. Discrete Fourier Transform	5
2.7. Frequency Sampling	5
3. Fast Fourier Transform	6
3.1. Non-uniform Fast Fourier Transform	6
4. Fourier Transform as Signal Decomposition	6
4.1. What is a Signal Decomposition?	6
4.1.1. Ridge Regression	7
5. Expressing Fourier Transform as Matrix	8
6. Matched Filter	9

1. Introduction

por definir.

2. Fourier Transform

The Fourier Transform it's a Mathematical toll that allow us to decompone singals by a given dictionary or in other words, to apply a change of basis to a given signal. Thinking on a continuous signal $s(t)$, it's Fourier Transform is given by the following integral, where $i = \sqrt{-1}$ denotes the imaginary unit:

$$\tilde{s}(f) = \int_{-\infty}^{\infty} s(t)e^{-2i\pi ft} dt = \mathcal{F}[s(t)] \quad (1)$$

And the inverse transform is:

$$s(t) = \int_{-\infty}^{\infty} \tilde{s}(f)e^{2i\pi ft} df = \mathcal{F}^{-1}[\tilde{s}(f)] \quad (2)$$

2.1. Properties of the Fourier Transform

Some usefull properties of the Fourier Transform are:

Fourier Transform is a linear operator. This means that, given a constant a and functions $s(t)$, $h(t)$, we can write:

$$\mathcal{F}[s(t) + h(t)] = \mathcal{F}[s(t)] + \mathcal{F}[h(t)] \quad (3)$$

$$\mathcal{F}[as(t)] = a\mathcal{F}[s(t)] \quad (4)$$

A time-shift impart a phase in the fourier Transform. For a given function $s(t)$, it's satisfied:

$$\mathcal{F}[s(t - t_0)] = \mathcal{F}[s(t)]e^{-2i\pi ft_0} \quad (5)$$

The Power Spectrum correspond to the square of the absolute value of the Fourier Transform. This computation allow us to remove the phase and the complex part which is changed to real by the absolute function.

$$\mathcal{P}_s = |\mathcal{F}[s(t)]|^2 \quad (6)$$

Convolution Theorem. The definition of convolution for two functions $s(t)$ and $h(t)$, is given by:

$$[s(t) * h(t)] = \int_{-\infty}^{\infty} s(\tau)h(t - \tau)d\tau \quad (7)$$

Calculating the Fourier Transform of this convolution leads to:

$$\mathcal{F}[s * h] = \mathcal{F}[s] \cdot \mathcal{F}[h] \quad (8)$$

Also, Fourier Transform of a point-wise multiplication of two functions leads to a convolution of Fourier Transforms:

$$\mathcal{F}[s \cdot h] = \mathcal{F}[s] * \mathcal{F}[h] \quad (9)$$

In practice using Fourier to calculate a convolution is faster than just calculate the convolution, that's why this is a very important concept.

Multiplication theorem. For a Singal which is a point-wise multiplication of two functions $s(t)$ and $h(t)$, multiplication theorem states:

$$\int_{-\infty}^{\infty} s(t)h^*(t)dt = \int_{-\infty}^{\infty} \tilde{s}(f)\tilde{h}^*(f)df \quad (10)$$

Correlation. The cross-correlation of two real signals $s(t)$ and $h(t)$ is defined as:

$$R_{sh}(t) \triangleq \int_{-\infty}^{\infty} s^*(\tau)h(\tau + t)d\tau \quad (11)$$

Since $s(t)$ is a real signal we have $s^*(t) = s(t)$. Also, if we apply the Multiplication Theorem (ec. 10) along together with time-shift (ec. 5) we get:

$$R_{sh} = \int_{-\infty}^{\infty} \tilde{s}(\omega)\tilde{h}^*(\omega)e^{2i\pi\tau}d\omega \quad (12)$$

This correspond to the Inverse Transform of the Convolution of the two signals (ec. 8)

$$R_{sh} = \mathcal{F}^{-1}[\mathcal{F}[s * h]] \quad (13)$$

A particular behavior is when calculating the auto-correlation of a signal, which correspond to the Inverse Fourier Transform of the Power Spectral Density:

$$R_{hh} = \mathcal{F}^{-1}[|\tilde{h}(f)|^2] = \mathcal{F}^{-1}[\mathcal{P}_h(f)] \quad (14)$$

2.2. Some useful Fourier Transforms

Fourier Transform of a Sinusoid is a Delta Function. need to insert a figure in this section

$$\mathcal{F}[\cos(2\pi f_0 t)] = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)] \quad (15)$$

Fourier Transform of a Dirac Comb is a Dirac Comb. The Dirac Comb is an infinite sequence of Dirac delta function placed at even intervals of size T :

$$III_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \quad (16)$$

thus, the fourier Transform correspond to a Dirac Comb in the frequency domain:

$$\mathcal{F}[III_T(t)] = \frac{1}{T} III_{1/T}(f) \quad (17)$$

2.3. Window Function

The Continuous Fourier Transform use the particularity that continuous functions are well defined for all times ($-\infty < t < \infty$). This is just an idealization that doesn't work on real signals because this only involve a finite span of time with some finite rate of sampling. The Fourier Transform of a discrete measured data will no longer be just the transform of the continuous underlying function, but rather the transform of the point-wise product of the signal($s(t)$) and the observing window($W(t)$), which will be a rectangular window of dirac comb.

$$s_{obs}(t) = s(t)W(t) \quad (18)$$

Using the convolution theorem (ec. 9) the transform of eq. 18 will be:

$$\mathcal{F}[s_{obs}(t)] = \mathcal{F}[s(t)] * \mathcal{F}[W(t)] \quad (19)$$

2.3.1. Fixing periodicity

Fourier Transform works only for periodic signals, this means, for signals where its first point has exactly the same value as the last point (continuity in edges). But in real world, a measured signal has always some grade of noise which lead to a discontinuity in the edges. In order to fix this discontinuity we use a window function.

For these cases, the window function will not be a rectangular, or at least, usually we don't want to use this kind of window. The most useful windows are kind-of curve like *Hann window*, *Blackmann window* or *Tukey window*. We are going to use mostly the *Tukey window* which is: should use a example plot here for all windows.

$$w(n) = \begin{cases} \frac{1}{2} \left[1 + \cos \left(\pi \left(\frac{2n}{\alpha(N-1)} - 1 \right) \right) \right] & 0 \leq n < \frac{\alpha(N-1)}{2} \\ 1 & \frac{\alpha(N-1)}{2} \leq n \leq (N-1)(1 - \frac{\alpha}{2}) \\ \frac{1}{2} \left[1 + \cos \left(\pi \left(\frac{2n}{\alpha(N-1)} - \frac{2}{\alpha} + 1 \right) \right) \right] & (N-1)(1 - \frac{\alpha}{2}) < n \leq (N-1) \end{cases} \quad (20)$$

Here, α define the form of the window, when $\alpha = 0$ we get a rectangular window and for $\alpha = 1$ we get a *Hann window*.

2.4. Nyquist-Shannon Sampling Theorem

sampling is a process of converting a signal (function-like of continuous time and/or space) into a numeric sequence, thus the sampling theorem states that, if a function $f(x)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart. This means that a sufficient sample-rate for uniform sampling is therefore anything larger than $2B$ samples per second, or from other perspective, for a given sample rate f_s , **perfect reconstruction is guaranteed possible** for a bandlimit $B < f_s/2$.

In the most general case of nonuniform sampling, this theorem states that a band-limited signal can be perfectly reconstructed from its samples if the **average sampling rate** satisfies the Nyquist condition. this means that even if uniformly spaced samples can be easier reconstructed (easier algorithms), it is not a necessary condition for perfect reconstruction.

2.5. Nyquist frequency

The nyquist frequency is defined as half of the sampling rate of a discrete signal when we have evenly-sampled data

$$\nu_N = 0.5f_s \quad (21)$$

And correspond to the highest frequency limit of one interval over which a periodogram is uniquely defined. If we go far away from this nyquist limit, the periodogram (and the simple Fourier Transform) will have repeated information.

For unevenly-sampled data, the symmetry that allow to define this frequency as eq. 21 is broken and using an average sampling rate as in the Sampling Theorem doesn't work. In order to find a best approximation to the real nyquist frequency (called *pseudo-nyquist frequency*) there are several methods in literature. The one implemented here correspond to the method developed by Koen who give a calculation formula to be solved for the nyquist frequency for arbitrary time spacing of measurements, this is the smallest positive root of

$$SS(\nu) = \sum_{l=1}^{N-1} \sum_{k=l+1}^N [\sin 2\pi\nu(t_k - t_l)]^2 = 0 \quad (22)$$

This calculation will take $O(N_\nu N^2)$ where N_ν correspond to the number of frequencies to try in the formula. Since is a very expensive calculation, generally it is more useful to know what is the maximum frequency that we are interested in.

2.6. Discrete Fourier Transform

We can pass from continuous space to discrete space by applying a Rectangular Window with a dirac comb (eq. 19) This is because the delta serve to collapse the Fourier Integral into a Fourier Sum. Considering the most general case, where a continuous signal $s(t)$ is observed at an **irregular grid** of size N_t , leads to dirac comb:

$$III_{\{t_j\}}(t) = \sum_{j=1}^{N_t} \delta(t - t_j) \quad (23)$$

Using this, the direct transform will be:

$$\begin{aligned} \mathcal{F}[s_{obs}(t)] &= \tilde{s}_k = \mathcal{F}[s(t)III_{\{t_j\}}(t)] \\ \tilde{s}_k &= \int_{-\infty}^{\infty} s(t) \sum_{j=1}^{N_t} \delta(t - t_j) e^{-2\pi i f t} dt \\ \tilde{s}_k &= \sum_{j=1}^{N_t} \int_{-\infty}^{\infty} s(t) \delta(t - t_j) e^{-2\pi i f t} dt \\ \tilde{s}_k &= \sum_{j=1}^{N_t} s_j e^{-2\pi i t_j k \Delta f} \end{aligned} \quad (24)$$

Following the same idea we can get the inverse transform, considering that the **frequencies will always be in a regular grid** so $f_k = k\Delta f$:

$$s_j = \Delta f \sum_{k=1}^{N_f} \tilde{s}_k e^{2\pi i t_j k \Delta f} \quad (25)$$

the total number of frequencies (N_f) and how we define the Δf will be explained more clearly in the next sections.

2.7. Frequency Sampling

Since we are considering the most general case of Fourier Transform for non-uniform sampling, it's necessary to define clearly the frequency grid, more precisely we need to define the frequency limit on the low end and the high end, in addition to define the grid spacing.

low-end frequency limit. For a set of observations in a time interval of length T , a signal with frequency $1/T$ will complete exactly one oscillation in the observing window. Choosing a low-end frequency limit could be this frequency or just set the minimum frequency to zero for simplicity. Also is possible to give a custom low-end limit based on what frequencies we want to explore.

high-end frequency limit. In order to not miss information that could be relevant, it's necessary to look for the maximum frequency that could give us information, this is given by

the nyquist frequency defined in section 2.5. Also is posible to give a custom high-end limit based on what frequencies we want to explore.

grid spacing. Choose the correct grid spacing it's also an important point, if we use a too fine grid will leads to unnecessarily long computational times, while, use a grid too coarse leads to missing narrow peaks that fall between grid points. A razonable idea is to choose grid spacing smaller than the spected width of the periodogram peaks, these peaks has a widht of $\sim 1/T$ where T is the duration of the window function. To ensure that our grid sufficiently samples each peak, we can over-sample by some factor (n_0) using a grid spacing of

$$\Delta f = \frac{1}{n_0 T} \quad (26)$$

And with this we can know the total number of frequencies to evaluate

$$N_f = n_0 T f_{max} \quad (27)$$

choose the value of n_0 is kind-of arbitrary and we can find uses in literature in the range of $n_0 = 5$ to $n_0 = 10$.

3. Fast Fourier Transform

The normal Discrete Fourier Transform showed in section 2.6 takes times $O(N^2)$ to compute wheter is the direct or the inverse transform, this is a very expensive computation and can be improved to $O(N \log(N))$ usign the algorithm of the Fast Fourier Transform(FFT). NEED TO READ MORE ABOUT THE ALGORITHM

3.1. Non-uniform Fast Fourier Transform

The same idea used in the optimization of the Discrete Fourier Tranform for evenly-sampled data called FFT can be used for unevnely-sampled data and we call this Non-unifor Fast Fourier Transform, using the same considerations. NEED TO EXPLORE MORE THE ALGORITHM

4. Fourier Transform as Signal Decomposition

4.1. What is a Signal Decomposition?

given a signal $s \in \mathbb{R}^N$ we want to find a representative set of coefficients $\beta \in \mathbb{R}^M$ such that:

$$\hat{s} = \Phi \hat{\beta} \quad (28)$$

Where $\Phi \in \mathbb{R}^{N \times M}$ is known as the dictionary matrix. Every column of this dictionary matrix correspond to an *atom*, i.e. the constituents of signal s . This dictionary is called *complete* when $M = N$. This dictionary can be either pre-defined or learned from data. We will select the atoms previously, for example, in a Fourier Dictionary we choose the atoms by knowing what frequencies we want.

Solve eq. 28 for a complete dictionary that is invertible means find $\hat{\beta}^*$ such that:

$$\hat{\beta}^* = \Phi^{-1} \hat{s} \quad (29)$$

When the matrix is not invertible, the solution to eq. 28 is obtained by solving the Least Square problem:

$$\min_{\hat{\beta}} \frac{1}{2} \|\hat{s} - \Phi \hat{\beta}\|_2^2 \quad (30)$$

Where $\|\cdot\|_2^2$ means the square of the L2-norm, the solution to this problem is:

$$\hat{\beta}^* = (\Phi^T \Phi)^{-1} \Phi^T \hat{s} \quad (31)$$

Which requires $(\Phi^T \Phi)$ to be invertible, means, having linear independent columns. The matrix $(\Phi^T \Phi)^{-1} \Phi^T$ correspond to the left Moore-Penrose pseudo-inverse of Φ .

Then complete dictionary is not often the better dictionary, here we are more interested in the overcomplete dictionary, this is a dictionary with more atoms than data samples ($M > N$). For overcomplete dictionaries (28) is under-determined and many solutions may exists. When Φ is overcomplete, the product $(\Phi^T \Phi)$ is always a singular matrix and then (31) cannot be compute. In order to get an optimal its necessary to include a regularization coefficient which impose conditions to help choose one of the many solutions of the underdetermined system. There are two kind of regularization usually used, one is smooth-ness and the other is sparsity, the first refers to the least-rough or least-complex solution this condition usually leads to dense solutions with almost all points non-zeros, and the second one refers to solution with less non-zero points and thus a more sparse solution. here we will show many method that try to find a good β .

4.1.1. Ridge Regression

this is an example of a regression that use as regularization smooth-ness, the ridge regression is defined as:

$$\min_{\hat{\beta}} \alpha \|\beta\|_2^2 + \|s - \Phi \beta\|_2^2 \quad (32)$$

Where $\alpha > 0$ is the trade-off between reconstruction error and smooth-ness of the solution. Te solution to 32 is:

$$\beta^* = (\Phi^T \Phi + \alpha I)^{-1} \Phi^T s \quad (33)$$

The discrete Fourier Transform described in section 2.6 can be interpreted as a change of basis of the signals which means change the time representation of the signal \vec{s} to a frequency representation $vec\beta$ using a sine/cose basis. This will means that we can write:

$$\vec{s} = \Phi \cdot \vec{\beta} \quad (34)$$

Where Φ is the Dictionary/Basis to use in the change of representation. This Dictionary is the Fourier Matrix where every colummn (atom ϕ) correspond to a sine/cosine wave of a given frequency $k\Delta f$:

$$\phi_k = e^{2\pi i t_j k \Delta f} \forall t_j \quad (35)$$

and in order to represent Direct and Inverse Fourier Transform Correctly it's neccessary to use an orthogonal Basis which means an square matrix Dictionary with a defined Inverse, if this matrix is defined correctly (ofter in the evenly-sampled case) his adjoint will be his inverse in a way that allow us to define the Adjoint Transform and use it as the Inverse Transform.

$$\vec{\beta} = \Phi^T \cdot \vec{s} \quad (36)$$

Sadly for unevenly-sampled data we usually need to compute a frequency grid higher than the time grid producing a oversampling Dictionary, and even if we choose the same number of frequencies than times, there is no guarantee that the resulting Fourier Matrix will be unitary and invertible in a way that his adjoint transform be his Inverse Transform. Because of this, we need to look for a way to compute the Pseudo-inverse if the dictionary (Φ^+) instead of the Inverse, usually we can compute this by Single Value Decompostion, where the left decomposition (known as):

$$\Phi^+ \quad (37)$$

5. Expresing Fourier Transform as Matrix

The transform described in ec. 24 and 25 can be represented matricially. Here the signal $s(t)$ will be a vector \vec{s} and the Fourier Transform will be the product of this vector with the fourier matrix $\Phi^{N_f \times N_t}$ where $\Phi_{jk} = e^{2\pi i t_j k \Delta f}$, here the .

$$(\vec{s})_j = \sum_{k=0}^{N_t} s_k e^{2\pi i t_j k \Delta f} = F \cdot \vec{s} \quad (38)$$

6. Matched Filter

Matched filter correspond to find the optimal linear filter for maximizing the Signal To Noise Ratio (SNR) in the presence of additive stochastic noise. For this we considere some data $s(t)$ with finite-energy which can be only Gaussian noise $s(t) = n(t)$ or a signal in addition to the noise $s(t) = n(t) + h(t)$, that are the input to a filter with impulse response $h(t)$, and produces the output signal given by the conovlution:

$$y(\tau) = \int_{-\infty}^{\infty} s(\tau - t)h(t)dt \quad (39)$$

Here the impulse response $h(t)$ correspond to the time reverse of the template

When we want to know if some data contain or not a well-known signal, the best method to use, if the data has stationary and Gaussian noise, is the Matched Filter. This method consis of, for a data which is noise plus some underlying signal $x(t) = s(t) + n(t)$, we whan to check, if this underlying singal $s(t)$ correspond to some well-known template $h(t)$. For this we estimate a Signal-to-Noise Ratio calculating the convolution of the data with the desired template divided by the self correlation of the noise knowed as Power Spectral Density. As we seen in section 2, the convolution of two signal can be expresedeadly in terms of fourier Transform. The most general definition of Matched Filter is:

$$\chi(t_0) = 2 \int_{-\infty}^{\infty} \frac{\tilde{s}(f)\tilde{h}^*(f)}{\mathcal{P}_n(f)}df \quad (40)$$

Where $\tilde{s}(f)$ and $\tilde{h}(f)$ are the Fourier Transform and $\mathcal{P}_n(f)$ is the One-Sided Power Spectral Density of the noise.

$$\mathcal{P}_n() \quad (41)$$