

A set of functions to analyze sound time series developed in python

Cecilia Jarne

cecilia.jarne@unq.edu.ar



Common task on data Analysis



AN INTRODUCTION TO KEY DATA SCIENCE CONCEPTS

Definitions of 10 fundamental terms for data science and machine learning.



model (n)

[mədəl] / noun

1. a mathematical representation of a real world process; a predictive model forecasts a future outcome based on past behaviors.



training (v)

[treɪnɪŋ] / verb

1. the process of creating a model from the training data. The data is fed into the training algorithm, which learns a representation for the problem, and produces a model. Also called "learning".



classification (n)

[klaesəfə'keʃən] / noun

1. a prediction method that assigns each data point to a predefined category, e.g., a type of operating system.



training set (n)

[treɪnɪŋ set] / noun

1. a dataset used to find potentially predictive relationships that will be used to create a model.



feature (n)

[fɪtʃər] / noun



algorithm (n)

[ælgo'rɪðəm] / noun

1. a set of rules used to make a calculation or solve a problem.



regression (n)

[rə'greʃən] / noun

1. a prediction method whose output is a real number, that is, a value that represents a quantity along a line. Example: predicting the temperature of an engine or the revenue of a company.



target (n)

[tɑ:gət] / noun

1. in statistics, it is called the dependent variable; it is the output of the model or the variable you wish to predict.



test set (n)

[test set] / noun

1. a dataset, separate from the training set but with the same structure, used to measure and benchmark the performance of various models.



overfitting (v)

[ou'verfɪtɪŋ] / verb



Common task on data Analysis



 model (n) [ˈmoʊdəl] / noun 1. a mathematical representation of a real world process; a predictive model forecasts a future outcome based on past behaviors.	 algorithm (n) [ælˈgoːrɪðəm] / noun 1. a set of rules used to make a calculation or solve a problem.
 training (v) [trɪnɪŋ] / verb 1. the process of creating a model from the training data. The data is fed into the training algorithm, which learns a representation for the problem, and produces a model. Also called "learning".	 regression (n) [rɪɡrɛʃən] / noun 1. a prediction method whose output is a real number, that is, a value that represents a quantity along a line. Example: predicting the temperature of an engine or the revenue of a company.
 classification (n) [klæsəfɪkeɪʃən] / noun 1. a prediction method that assigns each data point to a predefined category, e.g., a type of operating system.	 target (n) [tɔːrget] / noun 1. in statistics, it is called the dependent variable; it is the output of the model or the variable you wish to predict.
 training set (n) [trɪnɪŋ set] / noun 1. a dataset used to find potentially predictive relationships that will be used to create a model.	 test set (n) [test set] / noun 1. a dataset, separate from the training set but with the same structure, used to measure and benchmark the performance of various models.
 feature (n) [fɪtʃər] / noun 1. also known as an independent variable or a predictor variable, a feature is an observable	 overfitting (v) [oʊvərflɪtɪŋ] / verb 1. a situation in which a model that is too complex for the data has been trained to predict

Time series

A time-series T is an ordered sequence of n real-valued variables:

$$T = (t_1, \dots, t_n), t_i \in R$$

is often the result of the observation of an underlying process in the course of which values are collected from measurements made at uniformly spaced time instants and according to a given sampling rate.

Common task on time series data mining

- **Representation and indexing:**

How to represent the time series data and the approach of transform to another domain for dimensionally reduction

- **Segmentation:**

How to cut time series in meaningful parts.

- **Similarity measure:**

Could be between different time series or between segments of the same time series.

- **Visualization and mining:**

Including: pattern discovery, clustering, classification and other tasks.

The type of data: Bird songs



@Gabrielluislio

Cecilia Jarne

Four ideas to show:

- **Representation:** How to obtain envelop (adaptable to different kinds of data).
- **Dimensionally reduction:** How to obtain fundamental frequency of tonal sounds (Canary bird songs example).
- How to cut signal and obtain certain **features**.
- How to align segments (**Signal averaging**).

Representation: An heuristic way to obtain signal envelop

- First step: to take absolute value of signal $S(t)$ meaning $|S(t)|$
- Second step: to divide the $|S(t)|$ in k bunches of N samples:

$$|S(t)| = S_1(t) + \dots + S_i(t) + \dots + S_k(t)$$

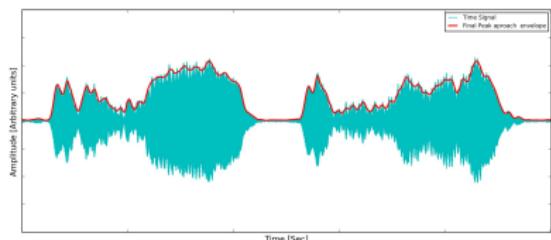
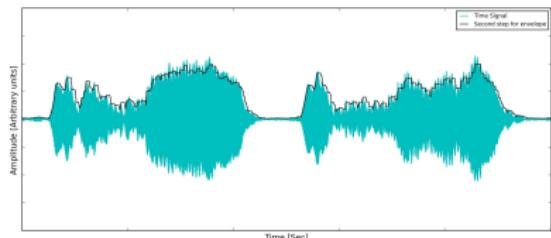
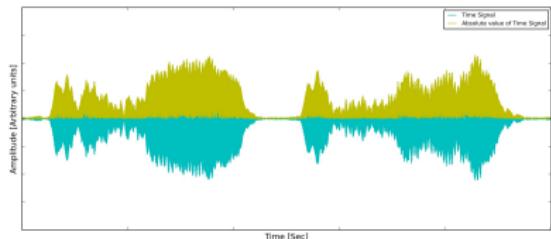
(Each $S_i(t) > 0$). The maximum value is: $\text{Max}(|S_i(t)|) = M_i$ corresponding to the j element value in each bunch.

- The third step: a lowpass-filter applied to the resulted signal $R(t)$ to get rid of the remaining staircase ripple $\hat{S}(t)$.

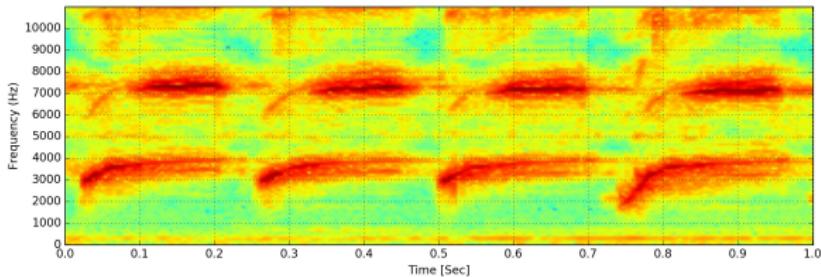
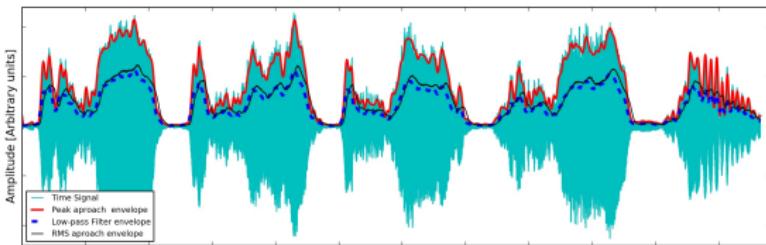


<https://arxiv.org/abs/1703.06812>

Representation: An heuristic way to obtain signal envelop

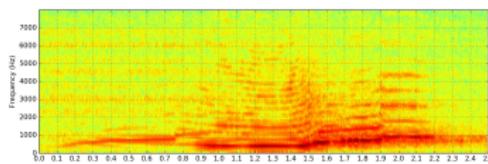
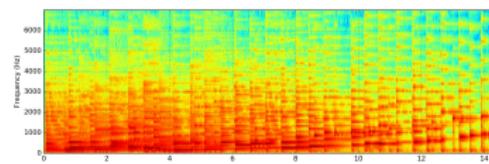
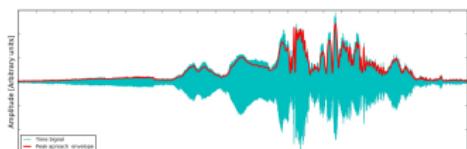
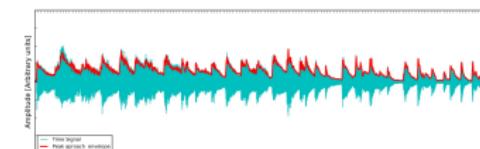
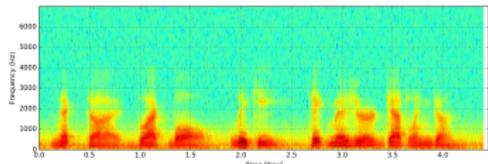
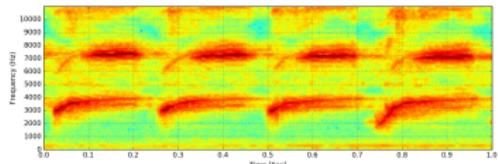
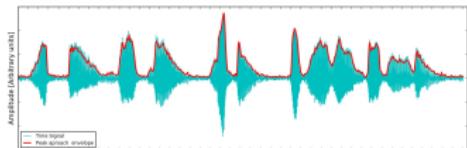
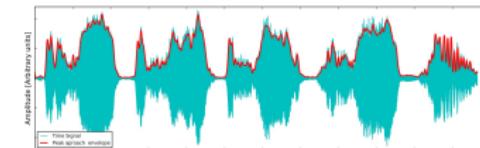


Representation: Comparison with other methods



Representation: An heuristic way to obtain signal envelop

Different examples of application:



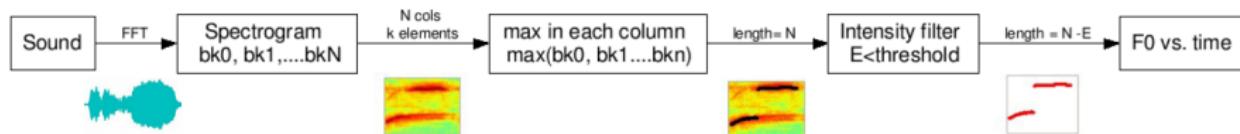
Code available in: <http://ceciliajarne.web.unq.edu.ar/investigacion/>

Cecilia Jarne

Dimensionally reduction: fundamental frequency (tonal)

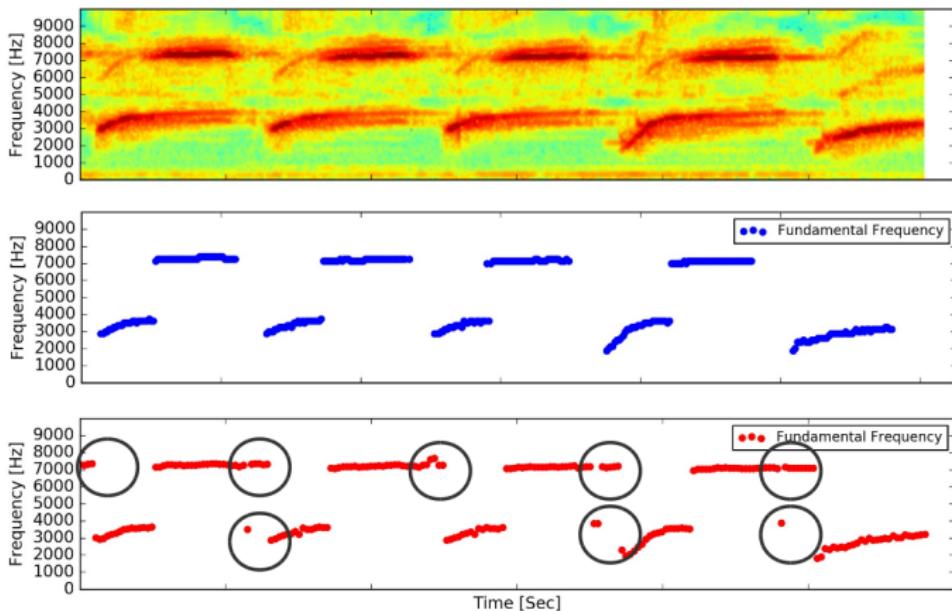
An algorithm to obtain fundamental frequency for tonal sounds:

- Spectrogram function is used over the consecutive signal segments.
- Only the value of maximum amplitude is saved as a new variable in each temporal bin.
- A second intensity filter is used to get rid of the noise and keep only the values representing the desired sound (consider only sounds that exceed a certain threshold).



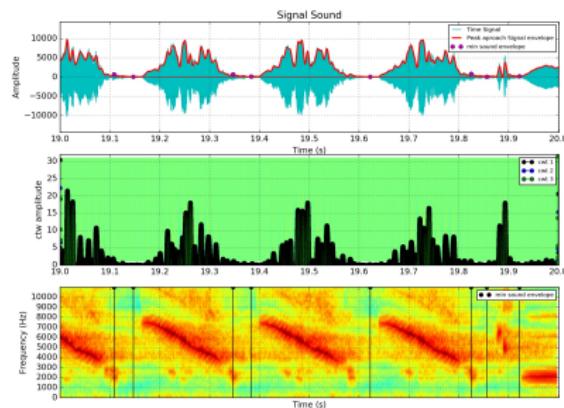
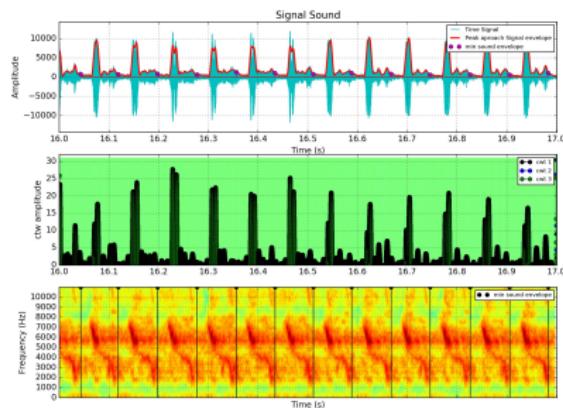
Code available in: [urlhttp://ceciliajarne.web.unq.edu.ar/investigacion/](http://ceciliajarne.web.unq.edu.ar/investigacion/)

An algorithm to obtain fundamental frequency for tonal sounds:



Dimensionally reduction: Wavelet decomposition

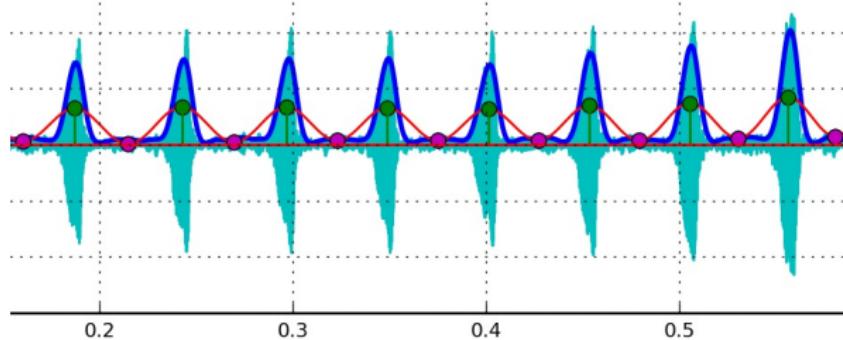
Performs a continuous wavelet transform on data, using the wavelet function.



```
1 import scipy.signal as signal  
2 ...  
3 cwtmatr = signal.cwt(filtered_aver, signal.ricker, widths)
```

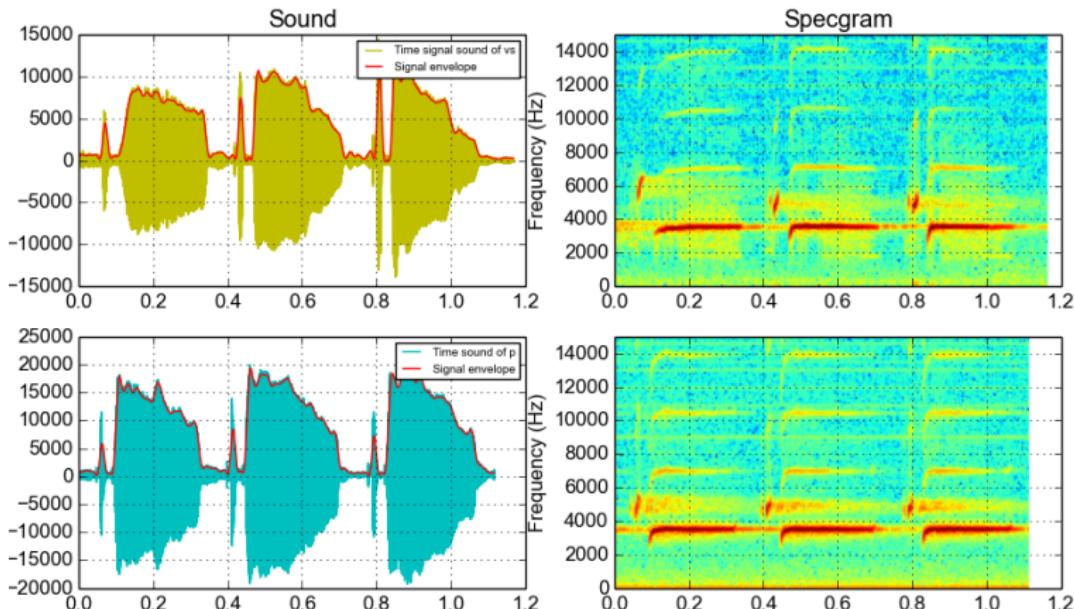
Syllable individualization: An idea for periodic section of time series

A very low pass filter applied and then finding the minima:



A way to align similar signals using correlation

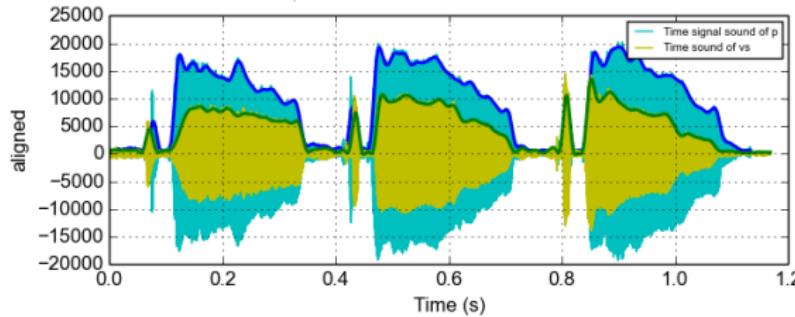
Two sound segments with similar frequency behavior.



A way to align similar signals using correlation

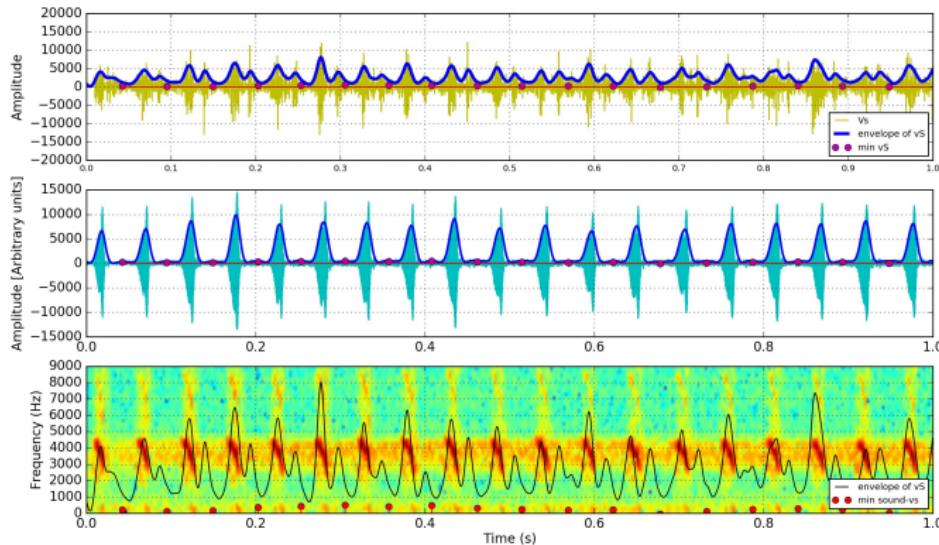
If we need to align the signals:

```
1 for target in [data]:  
2     print "target:",target  
3     if len(target)>0:  
4         dx      = np.mean(np.diff(time_pedacito))  
5         cros_cor = np.correlate(abs(signal),target,mode='full')  
6         pepe_cor = np.argmax(cros_cor)  
7         shift   = (pepe_cor - len(target))*dx
```



Alignment for averaging:

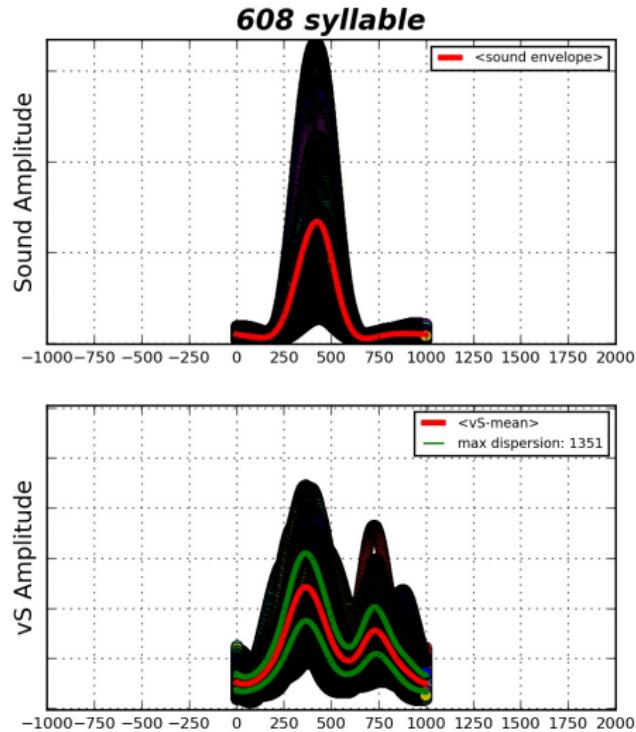
An example of multivariate time series



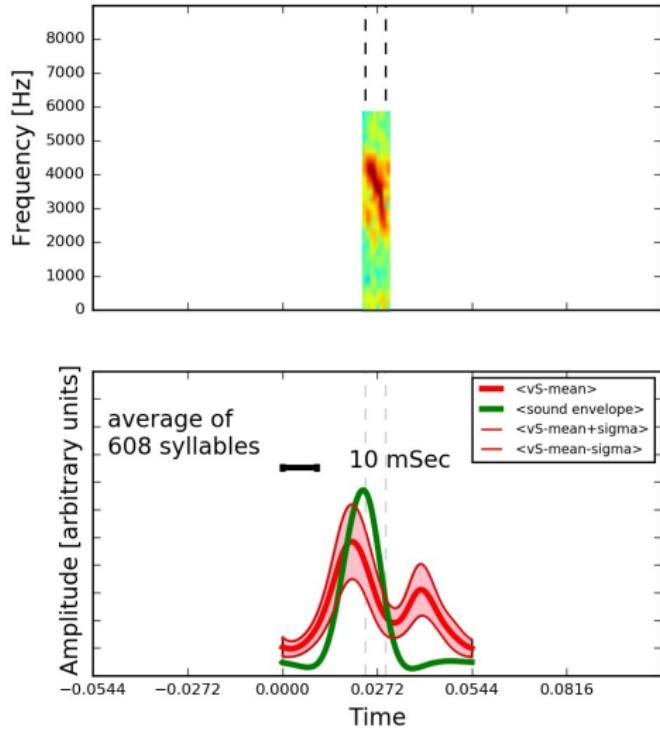
A way to align similar signals using correlation:procedure

- Cut signal(each sound segment is automatically cut by using a burly envelope)
- True envelope is estimated.
- We rescale by re-sampling each segment to a fix number of points.
- To synchronize all syllable of multiple segments, we used fafeature of the sound envelope (In most cases the value maximum, or the minimum.)
- We take the average in each position.

A way to align similar signals using a signal feature



A way to align similar signals using correlation



Conclusions

- An heuristic and simple technique for envelope estimation is proposed and a open source code is provided.
- An method to obtain fundamental frequency with a simple method.
- Some ideas to segment and align time series are shown.
- A method to estimate an average pattern is proposed.

Our group: Thanks!!!



Pablo Alcain, Rodrigo Lugones, Graciela Molina y yo

Our Motivation

What is Different in the Scientific Software Development Process?

- Requirements often are not that well defined
- Floating-point math limitations and the chaotic nature of some solutions complicate validation
- An application may only be needed once
- Few scientists are programmers (or managers)
- Often projects are implemented by students (inexperienced in science and programming)
- Correctness of results is a primary concern, less so the quality of the implementation

Our work

2015/ABR. ICTP-SAIFR

2015/NOV. SMN
(Curso Intensivo y personalizado)

2016/MAR. WTPC16 - UBA



2016/NOV.
CURSO POSTGRADO UNQ

2017/MAR. WTPC17 - UNT



More about our works



Tercer workshop de técnicas de programación científica:

<https://wtpc.github.io/>

<https://twitter.com/workshopTPC>

