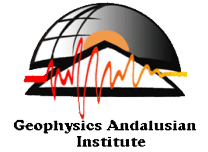


# GRANADA UNIVERSITY (UGR)



**THEORETICAL PHYSICS AND THE COSMOS  
DEPARTMENT**

**ANDALUSIAN INSTITUTE OF GEOPHYSICS AND  
PREVENTION OF SEISMIC DISASTERS**

## ***“Seismic Records Event Cut and Save System”***

*(Sistema para Cortar y Guardar Eventos de los Registros Sísmicos)*

**USER MANUAL: VERSION 1.0**

**Author:**

**Ligdamis A. Gutiérrez E. PhD.**

**Masaya Volcano, Nicaragua**

**Photo by: Ligdamis A. Gutiérrez E.**

*Granada, Spain 2021 - 2023*

## General Index

<a href="#">1.- Introduction</a>	3
<a href="#">2.- Initial Screen of the System</a>	4
<a href="#">2.1.- Elements of the Initial Screen</a>	6
<a href="#">3.- Main Cutting Interface</a>	7
<a href="#">3.1.- Analysis screen elements</a>	7
<a href="#">3.1.1.- Selection Block</a>	8
<a href="#">a) Drop-down ComboBox with filter list</a>	8
<a href="#">b) RadioButton, Employment time</a>	8
<a href="#">c) Trace number (default = 0)</a>	8
<a href="#">d) Input frequency and time data (initial/final)</a>	9
<a href="#">e) Cleaning Button</a>	9
<a href="#">f) Plot Signal Button</a>	9
<a href="#">3.1.2.- Route Block</a>	10
<a href="#">a) Load record Button</a>	10
<a href="#">b) Save Directory Selection Button</a>	12
<a href="#">3.1.3.- Command Block (Back/Exit)</a>	13
<a href="#">a) Go Back button</a>	13
<a href="#">b) Exit button</a>	13
<a href="#">3.2.- Filter type selection</a>	14
<a href="#">3.3.- Selecting entries by filter</a>	14
<a href="#">3.3.1.- Lowpass filter</a>	15
<a href="#">3.3.2.- Highpass filter</a>	16
<a href="#">3.3.3.- Bandpass filter</a>	17
<a href="#">3.3.4.- Bandstop filter</a>	19
<a href="#">4.- File path and directory upload examples</a>	20
<a href="#">5.- Errors in registration or entries validation</a>	21
<a href="#">6.- Filter and Cut Sections Results</a>	22
<a href="#">6.1.- Lowpass filtering and log cut selection results examples</a>	22
<a href="#">6.2.- Results with Start/End time selection and record cut examples</a>	25
<a href="#">7.- Chart Toolbar (Matplotlib Library)</a>	27
<a href="#">7.1.- Save the charts</a>	27
<a href="#">7.2.- Editing the axes and images of the Charts</a>	29
 <a href="#">CONCLUSION</a>	 32
<a href="#">Acknowledgments</a>	32
 <a href="#">Appendix A</a>	 33
<a href="#">A1.- Installation of Python and Additional Libraries</a>	33
<a href="#">A1.1. Installation Package Contents</a>	33
<a href="#">A1.2.- Installing Python on Windows</a>	33
<a href="#">A1.3.- Additional libraries installing</a>	34
<a href="#">A1.4 Automatic installation of libraries in Windows and Linux from the PIP</a>	37
 <a href="#">Appendix B</a>	 38
<a href="#">Python libraries install, for the accurate system functioning</a>	38

## 1.- Introduction

The "*Seismic Records Events Cut and Save System*" provides a user-friendly interface that allows for easy and efficient management of seismic record trace cutting, resulting in isolated events that facilitate operator analysis. The final event cutting result can be stored in two basic formats: "MSEED" and "SAC.". In this way, they can subsequently be used in calculations, analyses, and various scientific investigations. The system allows the use of two cutting methodologies: a) the use of predetermined start and end times, or b) the use of filtering techniques to determine the cross-section area much more easily and reliably using a resulting graph. These processes provide a reliable, automated tool that assists the operator in the spectral analysis process of seismic signals.

The application, through its incorporated libraries, allows for the reading of various seismic formats such as SAC, MSEED, GSE2, EVT, WAV, among others. Various filtering techniques can then be applied, automatically providing added value to the operator's expertise by enabling faster and more accurate identification of seismic waves compared to manual analysis in a continuous record.

The first version of this system consists of a single interface that includes tools for signal filtering and the process of cutting a specific event from an original seismic trace, whether measured in minutes, hours, or days. The main interface includes an English version of the system. However, the documentation, including this document, is available in Spanish. In the appendices, you will find information about the folder structure and its contents. The system also provides the ability to store the conversion of events at multiple frequencies according to the observer's criteria, which can be used in seismic institutes and observatories. Additionally, if desired, the event or record being converted can be graphically displayed, and the graphical results can be saved in various formats such as PNG, JPG, EPS, PS, PDF, RAF, TIF, among others.

The module and the entire system have been developed in Python, version 3.8.6. (The set of libraries is compatible with version 3.10.10). Additionally, a series of open-access libraries are included, which, in conjunction with Python, enable the use of graphical and analytical tools, providing ease of use and enhancing computational power for the user. Some of the main elements and libraries used are listed below:

- **Matplotlib:** Used for creating static, animated, and interactive visualizations in Python. (<https://matplotlib.org/stable/users/index.html>).
- **NumPy:** A library for numerical operations in Python. (<https://numpy.org/doc/stable/user/quickstart.html>).
- **PyQt5:** A tool that links with the graphical library Qt5 in C++ (<https://pypi.org/project/PyQt5/>).
- **Obspy:** A Python toolbox for seismology. (<https://docs.obspy.org/>).
- **Tkinter:** Graphical User Interface (GUI) (<https://docs.python.org/3/library/tkinter.html>)

Another key feature of the system is its definition as a cross-platform application, meaning it can operate on various platforms or operating systems, such as Windows (7, 8, 10, 11) in both 32-bit and 64-bit versions. It also supports Linux systems, such as Ubuntu and other similar systems (Debian, Red Hat, Fedora, SUSE, etc.), macOS, and Android for tablets and mobile devices (with Python appropriately adapted for these devices).

**NOTE:** In the appendices of this document (*as well as in the Readme.txt and Initial\_requirements.txt files*), you can find general information on installation for Windows and Linux systems, as well as guidelines for installing the main programs and additional libraries required by Python to properly execute the developed programs in its environment.

## 2.- Initial Screen of the System

In the appendices of this document and in the “**README.txt**” file included in the “**Documents**” folder, you will find instructions for installing the system on Windows (*the process on Linux systems is similar*). Essentially, you need to perform two actions:

- Copy the “Set\_tools\_System\_1\_1” folder to “My Documents” on Windows.
- Copy the “Set\_tools\_System\_1\_1.bat” file to the Windows desktop.

Additionally, there are instructions for installing the necessary Python libraries on the system. Once “Set\_tools\_System\_1\_1.bat” has been copied to the desktop, you need to right-click on it and select “Run as administrator.”



Fig. 1 Popup Window when Right-Clicking on the “Set\_tools\_System\_1\_1.bat” File

In the window that opens, click the “Yes” button when prompted with “*Do you want to allow this app to make changes to your computer?*” This is a warning message. However, the application does not make any changes, so you should trust its execution.

Upon clicking “Yes,” the following command window opens, welcoming you to the system.

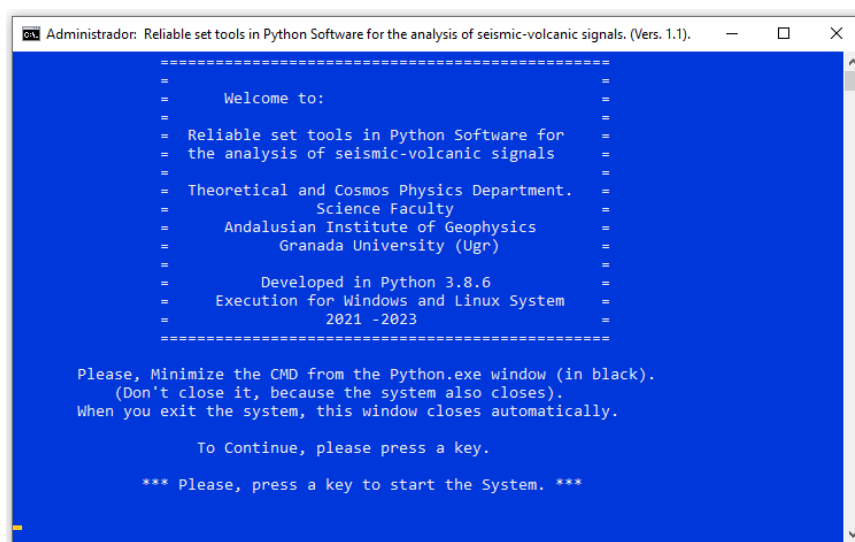


Fig. 2 Welcome Screen and Instructions for Loading the System.

After reading the information in the window, you simply need to press any key to access the system's initial screen. The folder should already be copied to “**My Documents**,” and the “Set\_tools\_System\_1\_1.bat” file contains all the loading instructions.

The system's initial screen is “**Menu.py**”. It appears when any key is pressed on the Welcome screen. Additionally, the Python command window or console is displayed, similar to the following:

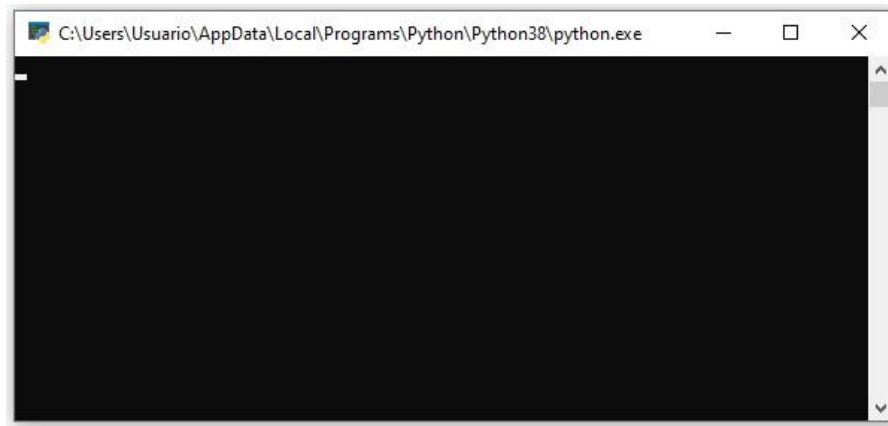
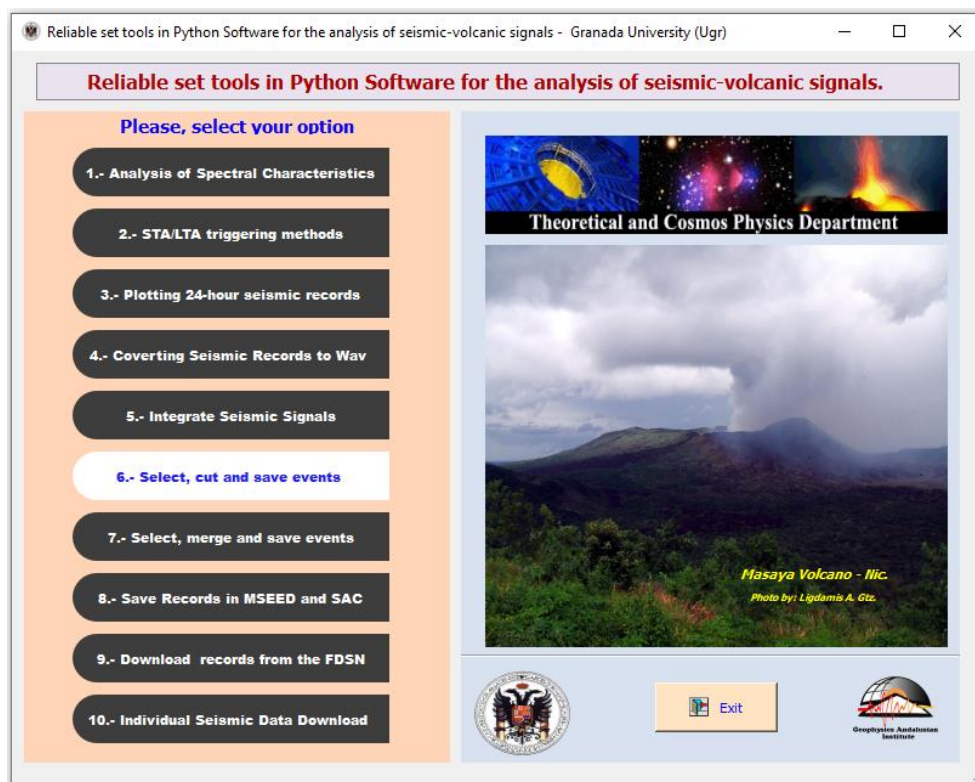


Fig. 3 Python Console (CMD) Window (Should be minimized)

To avoid obstructing the view, you can and should "minimize" this screen. Do **not** close it, as this would also close the system's startup window. Once you have finished working with the system, this window will close automatically. The initial presentation screen of the system (the module menu) “**Menu.py**” is as follows:



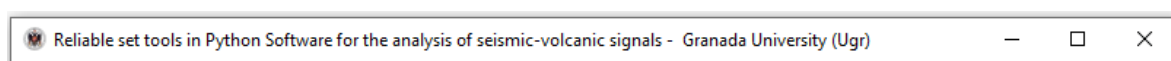
**NOTE:**  
When pressing or clicking on a module button, the startup window will close and the module window will open (*this may take a little time depending on the PC's memory. It is recommended to have at least 8 GB of memory in the system, with 16 GB being ideal*).

Fig. 4 Main Menu Screen. The module to be worked on is highlighted. Module 6 (Select, cut and save events).

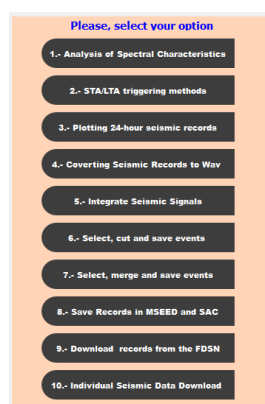
## 2.1.- Elements of the Initial Screen.

As observed in the previous figure, the initial or presentation screen is a simple window composed of:

- A top toolbar with basic information about the module.
  - On the left side, there are 10 execution buttons or command buttons for each module of the system.
  - At the bottom, there is a command button that allows for exiting the system.
  - Additionally, it features a background image representing a volcano (Masaya in Nicaragua), and three images with the logos of the University of Granada, the Andalusian Institute of Geophysics, and the Department of Theoretical and Cosmic Physics.
- At the top, the icon of the University is visible, along with the module title and a reference to the University of Granada (UGR).



- On the left side, there are 10 execution buttons or command buttons for each module of the system. When the mouse pointer is placed over each button, it is highlighted in white to indicate that it is being selected. Clicking on a button closes the startup menu window and opens the window for the indicated module (this may take a little time depending on the PC's memory).



- At the bottom, there is a command button: **Exit**. When the mouse pointer is placed over each button, a text appears indicating the action of that button (Exit System, Start System).



If you click the “Exit” button, a window will appear asking the user if they are sure they want to leave the system.

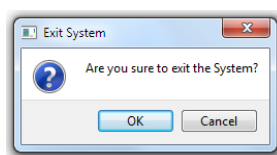


Fig. 5 Text Box Indicating Whether You Want to Exit the System.

If you click “**OK**,” the screen will close and the system will exit. If you click “Cancel,” you will remain on the initial screen.

### 3.- Main Cutting Interface.

The "Cutting main screen" is the module's main interface, where the activities that comprise the tools for reading records, filtering, and cutting events in seismic records are performed. This screen consists of the following sections:

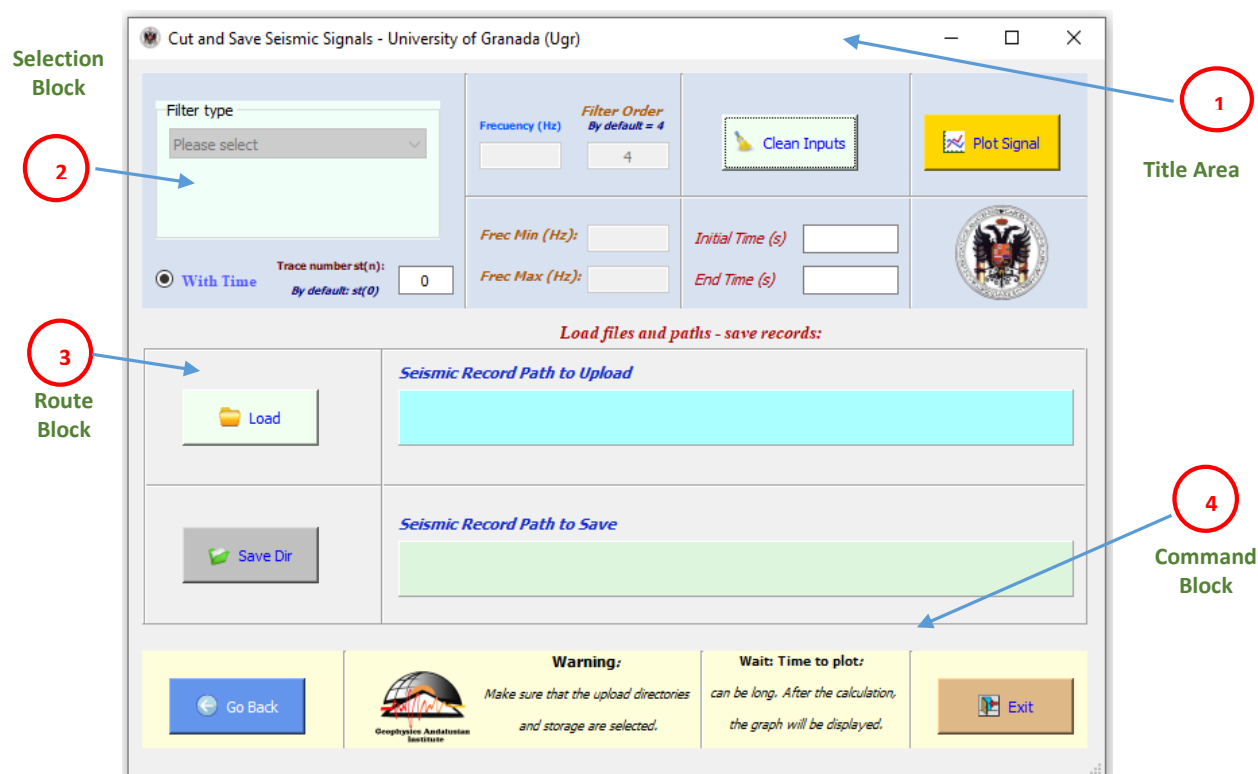
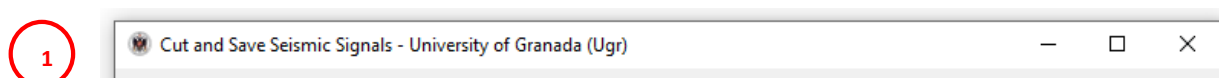


Fig. 6 System Analysis Screen Elements.

- 1) **Title Area:** (Module name and the University)
- 2) **Selection Block:** a) Command buttons: Plot Signal, Clean Input, b) Filter Types, c) Data Entry: (Frequency, Filter Order, Trace No., Minimum and Maximum Frequency, Initial and Final Time). d) RadioButton with default value of True.
- 3) **Route Block:** a) Physical location path of the record to be analyzed, b) Location of the destination directory. c) Command buttons: Load file and Save directory.
- 4) **Command Block:** Command buttons (Go Back, and Exit).

#### 3.1.- Analysis screen elements.

The previous screen is made up of various elements for its use. At the top, it displays the program name, icon, and the university name as the title (1)



The elements that make up the main screen are detailed below.

Added to number (1), the 3 elements of the initial fusion interface screen have been distributed into three main blocks, numbered (2-4) in red circles.

### 3.1.1.- Selection Block. (2)

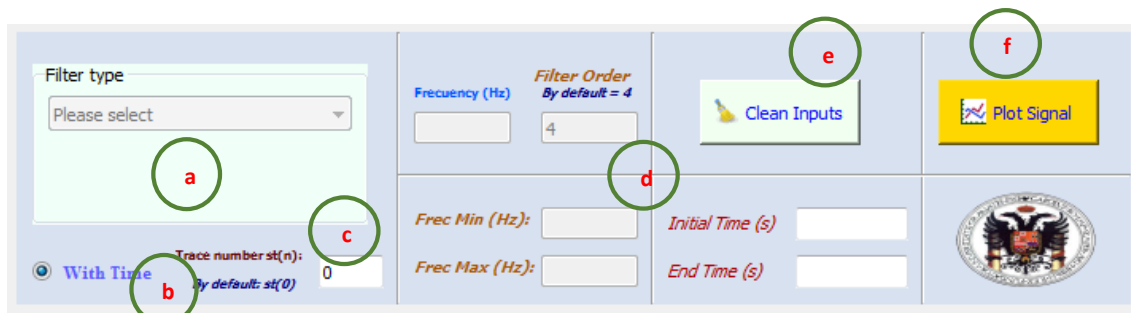


Fig. 7 Block of selection elements. In the green circles: a) Filter Type Combo Box, b) Radio Button, c) Trace number, d) Data input: Frequency, Filter Order, Minimum Frequency, Maximum Frequency, Start Time, End Time. e) Command button: Clean inputs, f) Plot Signal.

This block is configured (green circles in the previous figure) firstly by sections (a) and (b) that group the elements:

#### a) Drop-down ComboBox with filter list.



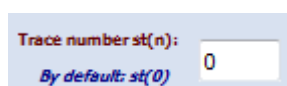
This element is initially disabled when the system is run because the system is initially configured to cut using start and end times. It is activated once the RadioButton element is clicked, disabling time-based selection and starting selection through signal filters (the available filter types are shown in the image above: low-pass, high-pass, band-pass, and band-stop).

#### b) RadioButton element, initially set to true.



If enabled, the system is set to perform the cutoff using the start and end times. If disabled, filters are used to display the signal graph

#### c) Trace number (initialized to 0 by default).



This option is useful if you have a seismic record containing multiple traces, such as an MSEED. This allows you to select the trace containing the seismic events you deem appropriate or need to cut. This option is enabled when you select to cut by time or filter.

**d) Input frequency and time data (initial/final).**

- 1) Data entry: Frequency in Hz (for Lowpass and Highpass filters).
- 2) Filter Order (initialized as 4 by default).
- 3) Minimum Frequency in Hz (for Bandpass and Bandstop filters).
- 4) Maximum Frequency in Hz (for Bandpass and Bandstop filters).

This section is disabled in time-cut mode and is only enabled when you disable the time RadioButton and access the filter type selection.

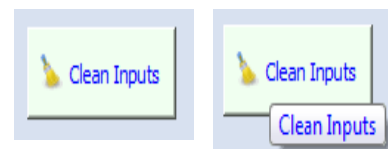
The last section of this block is marked (d) in the green circle. It corresponds to the data entries for:

- 1) Initial Time [Initial Time (s)]
- 2) End Time [End Time(s)].

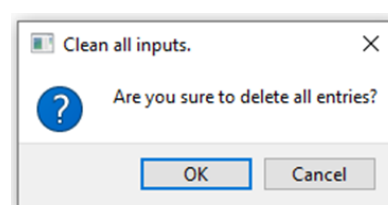
These areas are active at the beginning by default (RadioButton checked) and correspond to the initial and final time that you want to give to the event cut.

**e) Clean Inputs Button.**

Clicking this button clears all entries on the screen. It will clean the data entry text boxes and the path or folder where the record to be downloaded will be stored, and it will delete all active entries with data at that moment. It restores the initial values of the main interface (see Fig. 6).



When you place the mouse pointer over it, a message appears indicating its function. Clicking the button opens a window asking the user if they are sure they want to delete the data entries. If confirmed, it clears all entries and returns the interface to its initial state. Otherwise, the current entries remain in the interface.



**f) Plot Signal Button.**



Once the method to be used (Start/End time or Filters) has been selected and the data input has been completed, clicking this button will display the resulting graph and prompt the user to save the cutoff displayed in the resulting graph. When you hover the mouse pointer over these two action buttons, a message appears showing the actions they perform.

To do this, the time at which a particular event occurs within the seismic record must be known in advance.

### 3.1.2.- Route Block. (3)

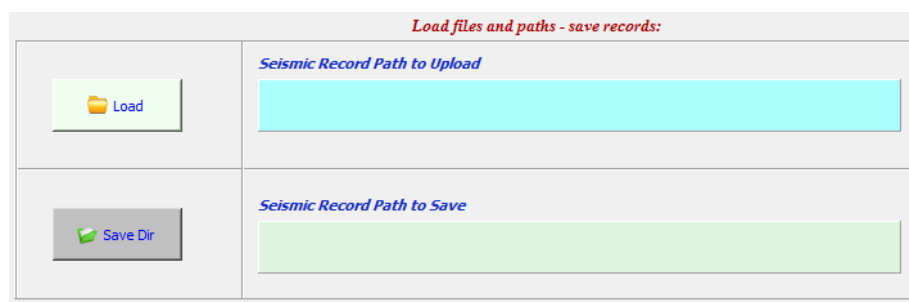


Fig. 8 Route Block, consisting of: File Load and Save Dir buttons. The areas where the Path will be displayed for each action are shown.

The image shows the "*Load Record*" button. When hovered over, the mouse pointer displays a message indicating its function. It searches for and loads seismic records using various formats. The "*Save Dir*" button searches for the desired folder or directory to store the event cut. To the right of each button, the areas where the paths for these actions will be displayed are shown.

a) **Command button "Load"**: Ensure that the files and directory are selected.



The action of the *Load Record* button allows users to click and open an explorer window (by default, it points to the root directory "C" of the PC), presenting options for various types of formats to search for and enabling the search within the computer's directory. This is shown in the following screen.

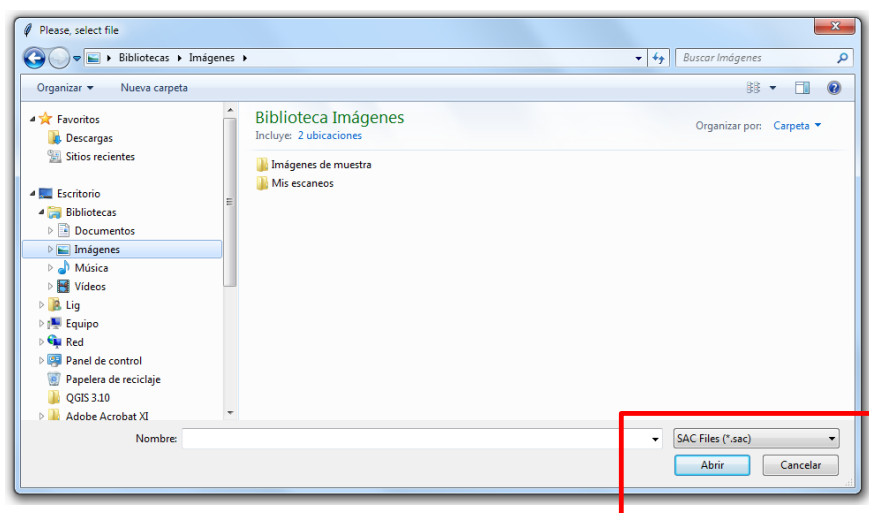


Fig. 9 Record Selection Screen.

In this screen (*the language is determined by the operating system*), records are selected according to the desired format (**red box**) such as SAC, MSEED, GSE2, EVT, etc. This is made possible through the “**Obspy**” library for reading seismic formats.

Once selected, clicking the “**Open**” button will load the record onto the analysis screen. Conversely, clicking the “**Cancel**” button will return the action to the analysis screen. The process of selecting a record is shown in the following screen.

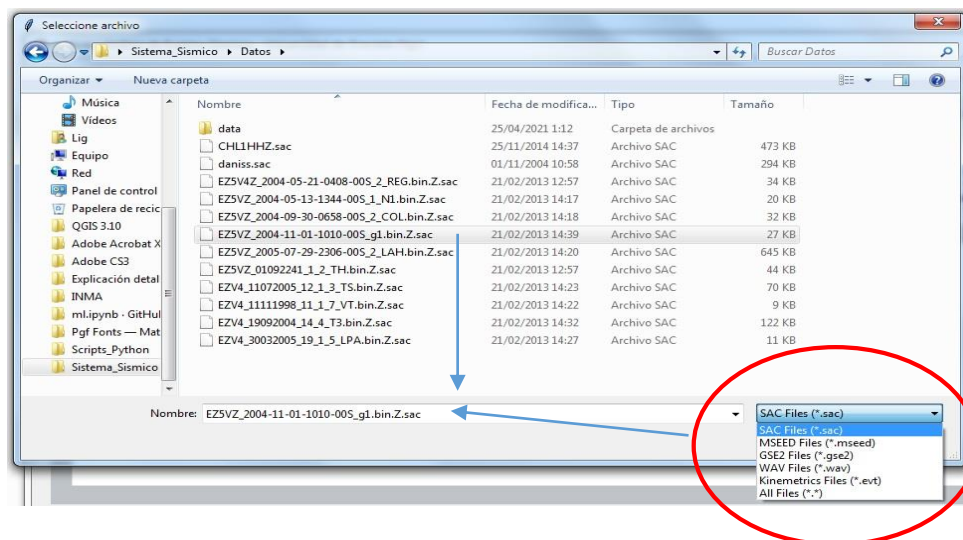


Fig. 10 Pantalla ejemplo de Selección de un registro de formato “SAC”.

In the screen, at the bottom right, indicated by the red circle and highlighted by the arrow, is the list of the most commonly supported and/or used seismic formats in observatories and institutes worldwide (SAC, *MSEED*, *GSE2*, WAV, *EVT*).

When you select a specific type, the records are displayed according to that format. For example, the “SAC” files stored in “[Data\\_examples](#)” Clicking on the desired record, as you can see, places it in the “Name” box. At this point, click the “**Open**” button presented in the previous screen. This loads the address or “**Path**” of the record's physical location in the system. This path will be displayed in the “*Seismic Record Path to Upload*” box, located to the right of the “**Load**” button.

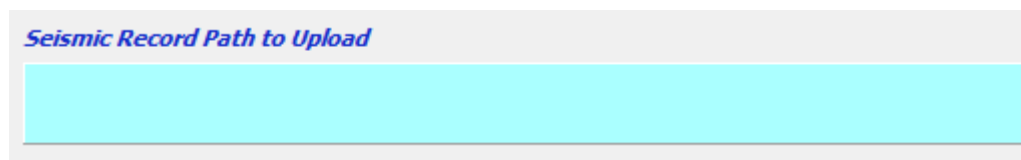


Fig. 11 File path box, which displays the location of the record.

This is an important aspect, as it determines whether the physical file where the record is stored on the computer can be subsequently located for analysis. If the file is invalid, cannot be found, or the parameters are incorrect, a validation window will appear indicating this. (*Cfr. Fig. 29, Pág. 21*).

b) **Directory selection button where the cut record will be saved (Save Dir).**



The "*Save Dir*" button, when hovered over with the mouse pointer, displays a message indicating its function. Clicking it opens a browser window (by default, the path is located in the "C" root directory of the PC), offering the option to select a folder or directory where the cut event will be stored. This can be seen in the following screen shot.

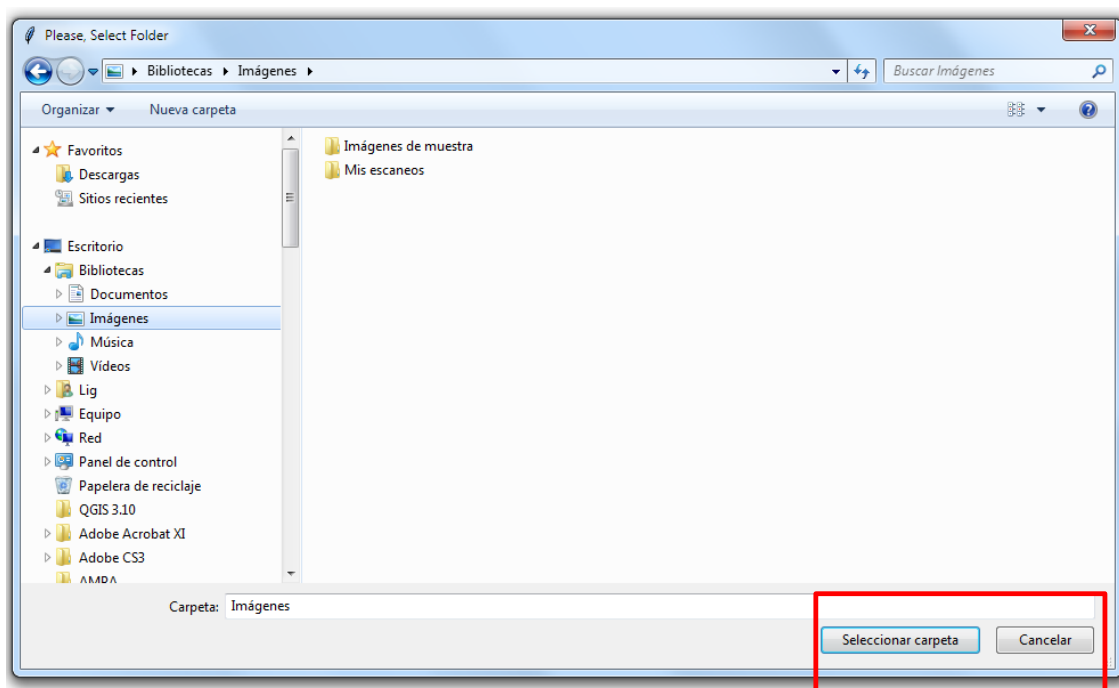


Fig. 12 Selecting the folder where the event to be cut will be stored.

On this screen (the language is determined by the operating system), click the "*Select Folder*" button. Click on the folder or directory where you want the event to be cut from the seismic record to be saved (**red box**). Otherwise, click the "*Cancel*" button and the action returns to the main cutting screen.

This path will be displayed in the "*Seismic Record Path to Save*" box, located to the right of the "*Save Dir*" button.

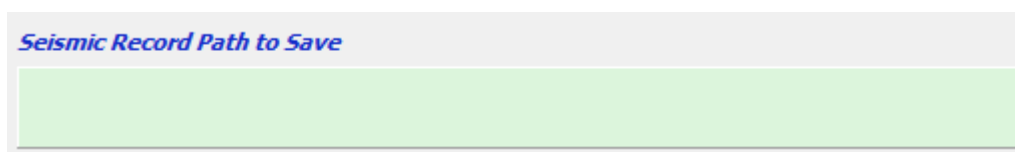


Fig. 13 Folder path box presenting the location of the record to be saved.

### 3.1.3.- Commands Block. (4)

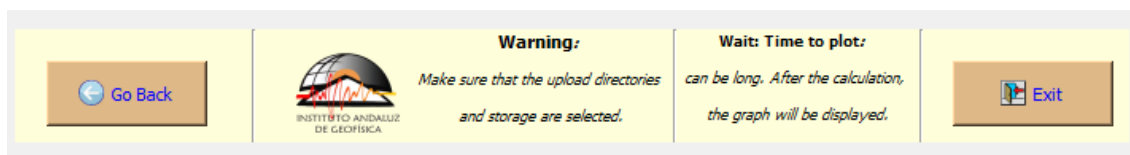


Fig. 14 Command block, composed of: "Go Back" and "Exit" buttons.

The image shows the "*Go Back*" button, which searches for and loads seismic records using various formats. The "*Exit*" button exits the system. Two messages are displayed; the first is a "Warning" message, which indicates ensuring that both the path to upload the source file and the directory where the result file will be saved are selected. The second message indicates that you should wait a moment for the graph to appear (this occurs especially when working in the start and end time selection mode), as the calculation process can be slow. Once the process is complete, the graph will be displayed (*it is recommended to select the filter option for long-time records, such as hours or days, to better represent the graph*).

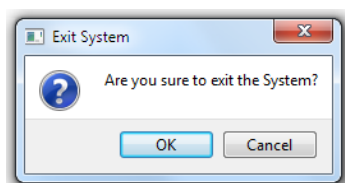
- a) "*Go Back*" **Button**: Allows returning to the initial system presentation screen. Hovering the mouse pointer over it displays a message indicating its function.



- a) "*Exit*" **button**: Allows for a complete exit from the system (*after displaying a prompt asking if you want to leave the system*). Hovering the mouse pointer over the button displays a message indicating its function.



In the same way as on the home screen, if the "*Exit*" **button** is clicked, a window will appear asking the user if they are sure they want to leave the system.



Clicking "**OK**" closes the screen and completes the system exit. "Cancel" returns to the analysis screen.

### 3.2.- Filter Type Selection.

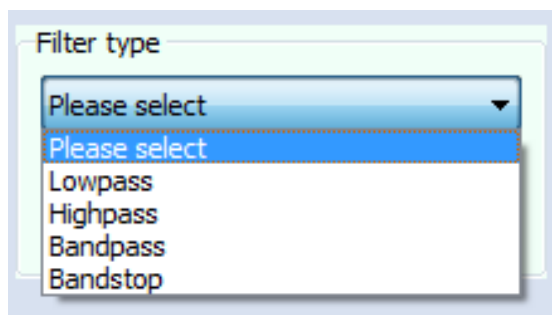


Fig. 15 Filter Type Selection via List.

The previous figure shows a dropdown list with various types of filters available for analyzing records. As observed, it includes filter types such as *Lowpass*, *Highpass*, *Bandpass*, and *Bandstop*. Selecting each of these items will activate one or more of the checkboxes in the continuous section, which correspond to the data inputs required for performing the calculations. At the initial value "*Please Select*", the inputs return to their default values.

### 3.3.- Selection of entries by filter.

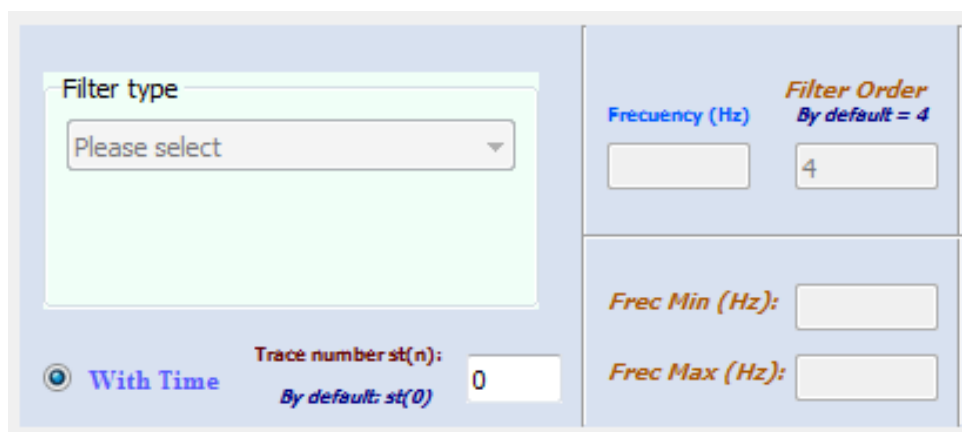


Fig. 16 Selection of input to calculate, record trace number (default = 0), filter order number (default = 4).

The inputs correspond to the filter type selected by the operator (*the default filter order is 4*). Initially, all checkboxes are disabled; they only become enabled depending on the filter type selected. Activation/Deactivation is performed as follows:

#### 1.- *Lowpass* and *Highpass* Filter.

**Activate to:** {Frequency (Hz), Filter Order, Trace number st (n)}.

**Deactivates:** {Frec. Min (Hz), Frec. Max (Hz)}.

#### 2.- *Bandpass* and *Bandstop* Filter.

**Activate to:** {Frec. Min (Hz), Frec. Max (Hz), Filter Order, Trace number st (n)}.

**Deactivates:** {Frequency (Hz)}.

The trace number defaults to 0. If there are multiple traces, for example in MSEED or SEISAN records, that contain either all three components or multiple values, the desired trace number will be entered starting from zero in this text box. Inputs are validated to accept only numbers, including decimals and negatives. Regarding the filter type, the input parameters activate the following check boxes:

### 3.3.1.- *Lowpass*<sup>1</sup> Filter Type.

Fig. 17 Example of *Lowpass* Filter Selection. In the red circle: "Frequency" input, in the yellow circle: "Filter Order", in the green circle: "Trace Number".

When selecting this type of filter, the following checkboxes are activated:

- Filter Frequency (Hz):** Floating-point value. Here, you should enter a valid frequency value for the filter calculation (Red circle). In the example, the frequency value entered is "0.86 Hz."
- Filter Order:** Integer value. This field should be set to a valid filter order for the calculation. Typically, this is order 2 or 4 (by default, it is set to 4 for all analyses) (Yellow circle).
- Trace Number st (0):** Integer value. Determines the trace number to be analyzed. By default, and in formats like "SAC," this value is "0." For multiple traces, such as in "MSEED" formats, the user specifies which trace number they wish to analyze.

The "Frec. Min (Hz)/Minimum Frequency" and "Frec. Max (Hz)/Maximum Frequency" checkboxes remain inactive.

Once the actions for loading the record are completed and the filter type is selected, click on the "Plot Signal" button. If an attempt is made to graph (by clicking the Plot Signal button) with invalid input or a nonexistent record or out-of-range format in any of the fields, a validation dialog box will appear, indicating that the action must be performed.

These dialog boxes also ensure that the program can continue running without severe interruptions to the system due to lack of data. The validations for incorrect entries are as follows:

<sup>1</sup> The Lowpass filter blocks high-frequency signals and allows low-frequency signals (frequencies below the cutoff frequency) to pass through.

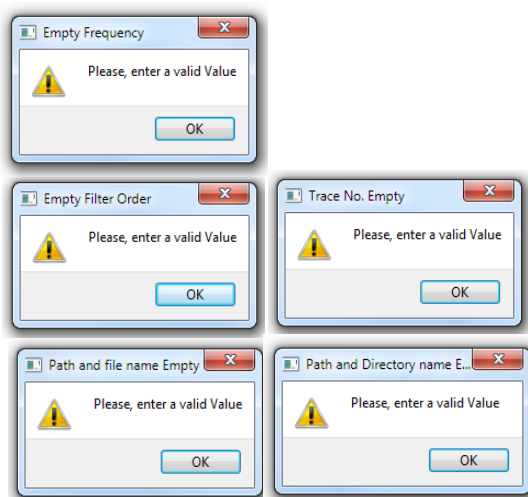


Fig. 18 Validation of entries for the *Lowpass* filter.

Once the inputs are corrected by entering a valid value or one within the permitted range, the program can continue and display the corresponding graph.

### 3.3.2.- *Highpass*<sup>2</sup> Filter type.

Fig. Example of *Highpass* Filter Selection. In the red circle: "Frequency" input, in the yellow circle: "Filter Order", in the green circle: "Trace Number".

Similar to the previous filter, the following checkboxes are activated:

- Filter Frequency (Hz):** Floating-point value. Here, you should enter a valid frequency value for the filter calculation (Red circle). In the example, the frequency value entered is "6 Hz."
- Filter Order:** Integer value. This field should be set to a valid filter order for the calculation. Typically, this is order 2 or 4 (by default = 4, for all analyses) (Yellow circle).
- Trace Number st (0):** Integer value. Determines the trace number to be analyzed. By default, and in formats like "SAC," this value is "0." For multiple traces, such as in "MSEED" formats, the user specifies which trace number they wish to analyze.

The "Frec. Min (Hz)/Minimum Frequency" and "Frec. Max (Hz)/Maximum Frequency" checkboxes remain inactive.

<sup>2</sup> The **Highpass Filter** blocks low-frequency signals and allows high-frequency signals (*frequencies higher than the cutoff frequency*) to pass through.

Once the actions for loading the record are completed and the filter type is selected, click on the "Plot Signal" button. If an attempt is made to graph (by clicking the [Plot Signal](#)) with invalid input or a nonexistent record or out-of-range format in any of the fields, a validation dialog box will appear, indicating that the action must be performed:

These dialog boxes also ensure that the program can continue running without severe interruptions to the system due to lack of data. The validations for incorrect entries are as follows:

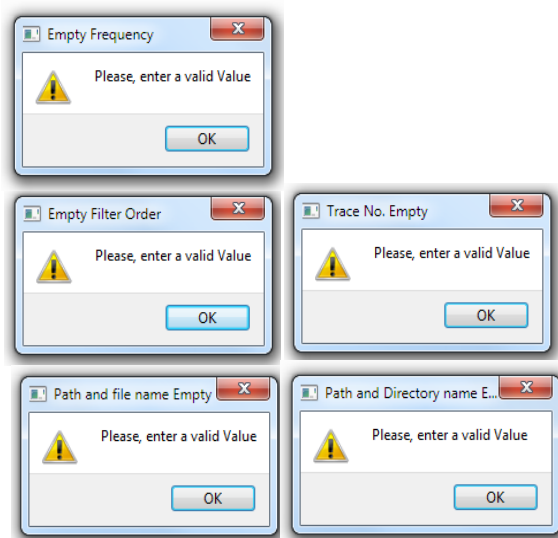


Fig. 20 Validation of entries for the [Highpass](#) filter

Once the inputs are corrected by entering a valid value or one within the permitted range, the program can continue and display the corresponding graph.

### 3.3.3.- [Bandpass](#)<sup>3</sup> Filter Type.

Fig. 21 [Bandpass](#) Filter Selection Example. In the red circle, the entries "Minimum Frequency (Hz) and Maximum Frequency (Hz)"; yellow circle -> "Filter Order"; green circle -> Trace No.

<sup>3</sup> The Bandpass filter allows spectral content to pass only within a range around the central frequency. This range is defined by a minimum frequency and a maximum frequency. It removes noise associated with both low and high frequencies that are outside this specified range.

Unlike the previous two, the following boxes are activated:

- Minimum Filter Frequency (Hz):** Floating-point value. A valid value must be entered here for the minimum frequency (red circle) to calculate the start of the center window. In the example, the value is "3 Hz."
- Maximum Filter Frequency (Hz):** Floating-point number value. A valid value must be entered here for the maximum frequency (red circle) to calculate the end of the center window. In the example, the value is "15 Hz."
- Filter Order:** Integer number value. In this field, you must enter a valid value for the filter order you wish to use for the calculation. Generally, order 2 or 4 (by default, all analyses are set to 4), (Yellow circle).
- Trace number st (0):** Integer number value: Determines the value of the trace to be analyzed. By default, and in formats such as "SAC", this value is equal to "0". In the case of multiple traces, such as in "MSEED" formats, the observer determines which trace number to analyze.

The "Frequency (Hz)/Filter Frequency" box remains inactive.

Once the actions are completed, click the "Integral Plot" button. If you want to graph the data by clicking the "Plot Signal"/Graph button (with an invalid or empty input field), a validation will appear in any of the fields, displaying a dialog box that indicates the action must be performed. These dialog boxes also allow the program to continue executing without a severe system interruption due to missing data. The validations for incorrect inputs are as follows:

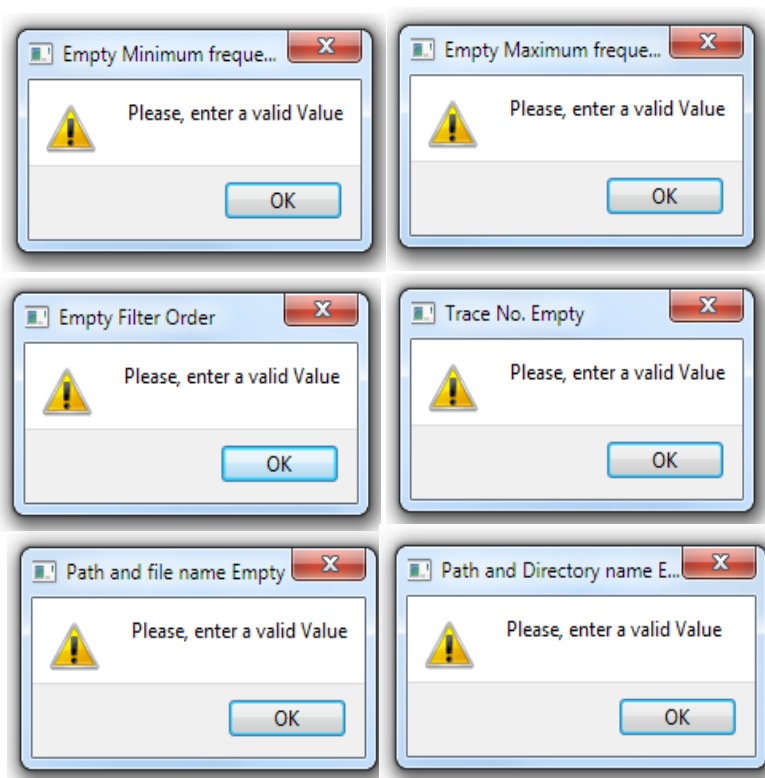


Fig. 22 Validation of inputs for *Bandpass* filter.

### 3.3.4.- Bandstop<sup>4</sup> Filter Type

Fig. 23 Example Bandstop Selection. In the red circle are the “Minimum Frequency (Hz) and Maximum Frequency (Hz)” inputs, yellow circle -> “Filter Order”, green circle -> Trace No.

In the same way as the Band-Pass filter, the Band-Stop filter activates the following checkboxes:

- e) **Minimum Filter Frequency (Hz)**: Floating-point value. A valid value must be entered here for the minimum frequency (red circle) to calculate the start of the center window. In the example, the value is “3 Hz.”.
- f) **Maximum Filter Frequency (Hz)**: Floating-point number value. A valid value must be entered here for the maximum frequency (red circle) to calculate the end of the center window. In the example, the value is “15 Hz.”.
- g) **Filter Order**: Integer number value. In this field, you must enter a valid value for the filter order you wish to use for the calculation. Generally, order 2 or 4 (by default, all analyses are set to 4), (Yellow circle).
- h) **Trace number st (0)**: Integer number value: Determines the value of the trace to be analyzed. By default, and in formats such as "SAC", this value is equal to "0". In the case of multiple traces, such as in "MSEED" formats, the observer determines which trace number to analyze.

The “Frequency (Hz)/Filter Frequency” box remains inactive.

Once the actions are completed, click the "Integral Plot" button. If you want to graph the data by clicking the "Plot Signal"/Graph button (with an invalid or empty input field), a validation will appear in any of the fields, displaying a dialog box that indicates the action must be performed. These dialog boxes also allow the program to continue executing without a severe system interruption due to missing data.

The validations for incorrect inputs are as follows:

<sup>4</sup> The Bandstop filter blocks signals with frequencies between the upper and lower cutoff frequencies. In other words, it removes or attenuates a specific band of frequencies.

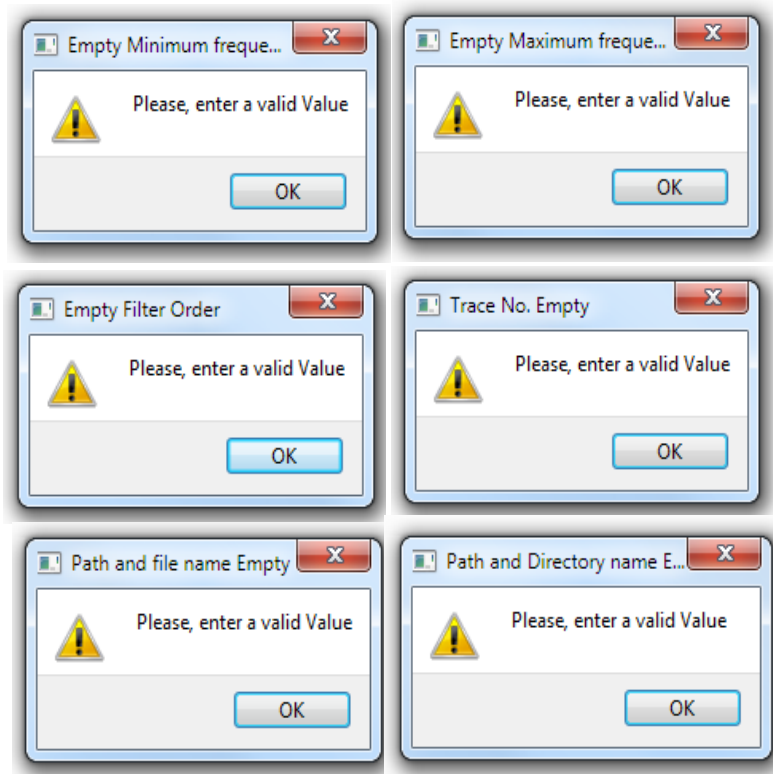


Fig. 24 Validation of inputs for the *Bandstop* filter.

#### 4.- Examples of file path and directory upload.

##### *Seismic Record Path to Upload*

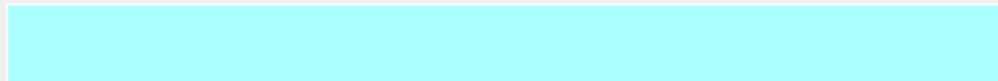


Fig. 25 Physical path block of the file to be uploaded.

File path presentation area (Seismic Record Path to Upload): This area displays the path (Disk/folder/file), the path or "path" of the physical file (where it is stored on the computer), so that it can be called by the system to perform the required calculations.

An example of the output when performing the calculations can be seen in the following image.

##### *Seismic Record Path to Upload*

C:/Users/Lig/Desktop/Data\_examples/EZ5VZ\_2004-11-01-1010-00S\_g1.bin.Z.sac

Fig. 26 Example of the physical path of the file to be uploaded.

The "**Path**" (on the computer) to the location of the record is observed.

[Back to Index](#)

In the case of the folder or directory.

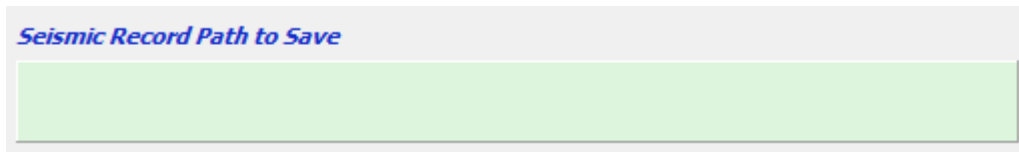


Fig. 27 Physical path block of the folder to select to store the cut event.

An example of the output when performing the calculations can be seen in the following image.

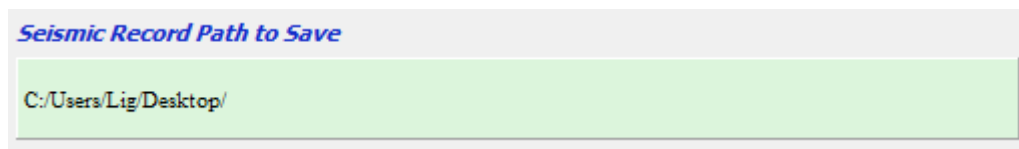


Fig. 28 Example of the physical path of the folder or directory selected to store the cut.

The "**Path**" (on the computer) of the location where the record will be stored is observed.

#### 5.- Validation of errors in registration or entries.

Clicking the "**Plot button**" will trigger an error validation if an invalid entry, non-existent record, or out-of-range format occurs. A dialog box will appear with an alert message indicating the issue. This allows the user to modify the entries or select a valid record without causing the system to crash or halt. The screen displayed is as follows:

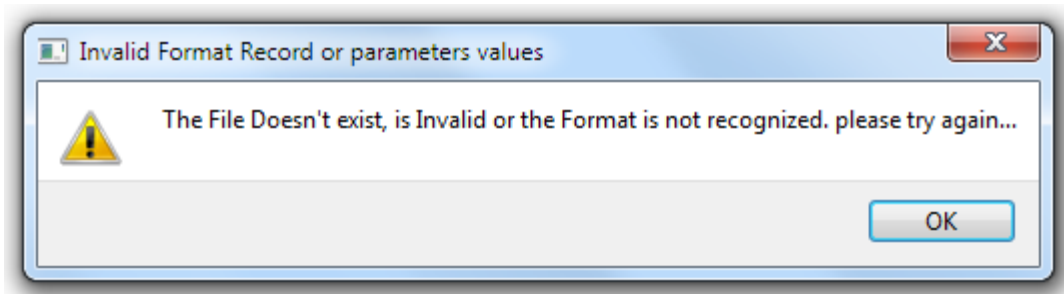


Fig. 29 Validation of invalid entries, formats, or incorrect records.

According to the message in the previous figure, an error has occurred because the format is not recognized, or the record does not exist. Additionally, the parameters or entries might be out of the allowed range according to the signal being analyzed. By pressing the "**OK button**", you return to the system to select a valid file or correct the erroneous entries. This way, the program continues to run without issues.

## 6.- Filter and Cut Section Results.

Next, examples of the final results of the cutting, graphing, and saving process of the record will be presented, according to the format selected by the user.

### 6.1.- Example of results from low-pass filtering and record cutoff selection.

According to the entire process described above, the process for cutting an event within a seismic trace, with a low-pass filter, is very simple, and consists of the following steps:

712 / 5.000

- Uncheck the "**With Time**" radio button to activate the filter dropdown menu.
- Open or select a specific record (the file path is displayed as: Seismic Record Path to Upload). By default, the initial path is located in the root directory "C" of the PC (whether on a Windows or Linux system).
- Open or select the folder or directory where the clipped signal will be saved. (The file path is displayed as: Seismic Record Path to "Save").
- Select the low-pass filter.
- Enter the filter parameters.
- Click the "**Plot Signal**" button to graph the record.

The selection of these parameters is shown in the following figure.

The screenshot displays the 'Cut and Save Seismic Signals - University of Granada (Ugr)' application window. The interface is organized into several sections. At the top, there are input fields for 'Filter type' (set to 'Lowpass'), 'Frequency (Hz)' (3), and 'Filter Order' (4, with a note 'By default = 4'). Below these are fields for 'Trace number st(n):' (0) and 'By default: st(0)'. To the right are buttons for 'Clean Inputs' and 'Plot Signal'. The middle section, titled 'Load files and paths - save records:', contains two rows. The first row has a 'Load' button and a text field for 'Seismic Record Path to Upload' containing 'C:/Users/Usuario/Desktop/Data\_examples/EZ5VZ\_2004-11-01-1010-00\$\_g1.bin.Z.sac'. The second row has a 'Save Dir' button and a text field for 'Seismic Record Path to Save' containing 'C:/Users/Usuario/Desktop/'. At the bottom, there is a 'Warning' section with a logo and text: 'Warning: Make sure that the upload directories and storage are selected.' Next to it is a 'Wait: Time to plot:' section with text: 'can be long. After the calculation, the graph will be displayed.' Finally, there are buttons for 'Go Back' and 'Exit'.

Fig. 30 Example of the inputs for a selection through a **Lowpass** filter.

The output of this analysis will consist of a graph in two parts: the filtered signal, and an area where the established cutoff will be displayed. Prior to the cutoff, the resulting graphs were zoomed in (*using the Zoom tool*) (See Matplotlib Tools, pp. 27-32).



The first graphic screen is as follows.

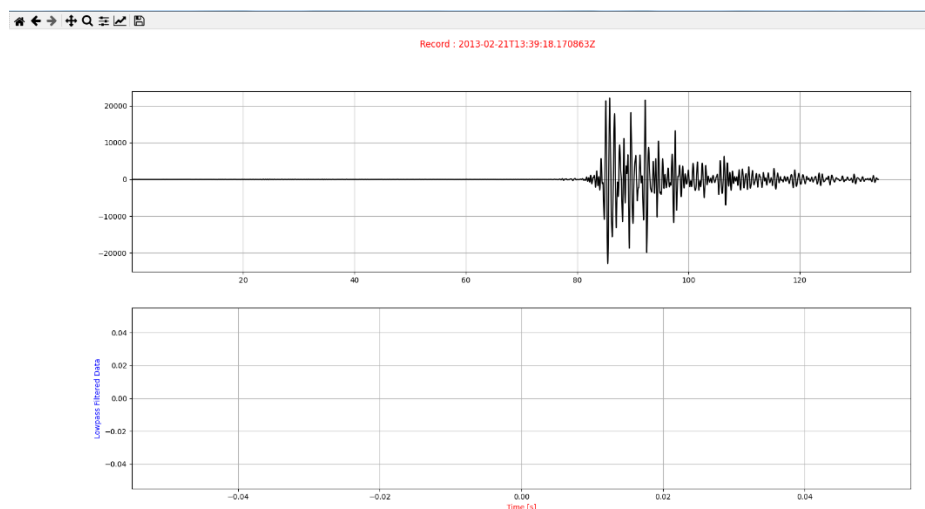


Fig. 31 Graphical output resulting from clicking the “Plot Signal” button.

The graph shows the complete recording, but the goal is to crop the event. To do this, you can even use the magnifying glass tool at the top to zoom in. Then, you must deactivate the magnifying glass tool (*otherwise, the cropping area selection won't work*). Click on the area you consider the starting point, drag the mouse to the end of the event, and release the mouse button there. In this example, the selection is from second 80 to second 120. The area will be highlighted in blue as follows.

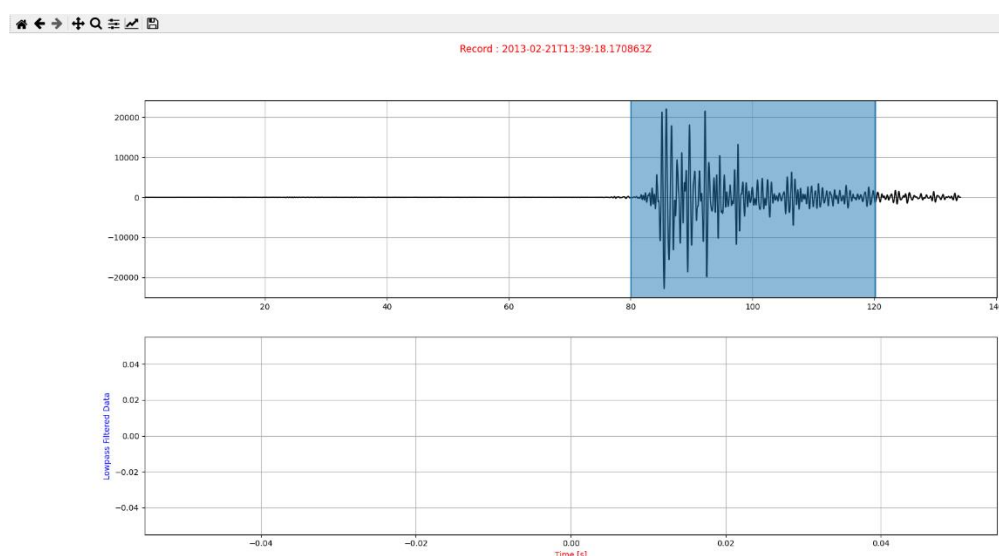


Fig. 32 Area of the resulting graph after clicking at the start, dragging the cursor and selecting (in blue) what you want to cut (80-120 s).

Once you release the mouse cursor, the selected event will be highlighted at the bottom, and a window will appear asking if you want to save the selection in "MSEED" or "SAC" format. If you do not wish to save, click the "Cancel" button, and the chart will close, allowing you to access another event. The image is as follows.

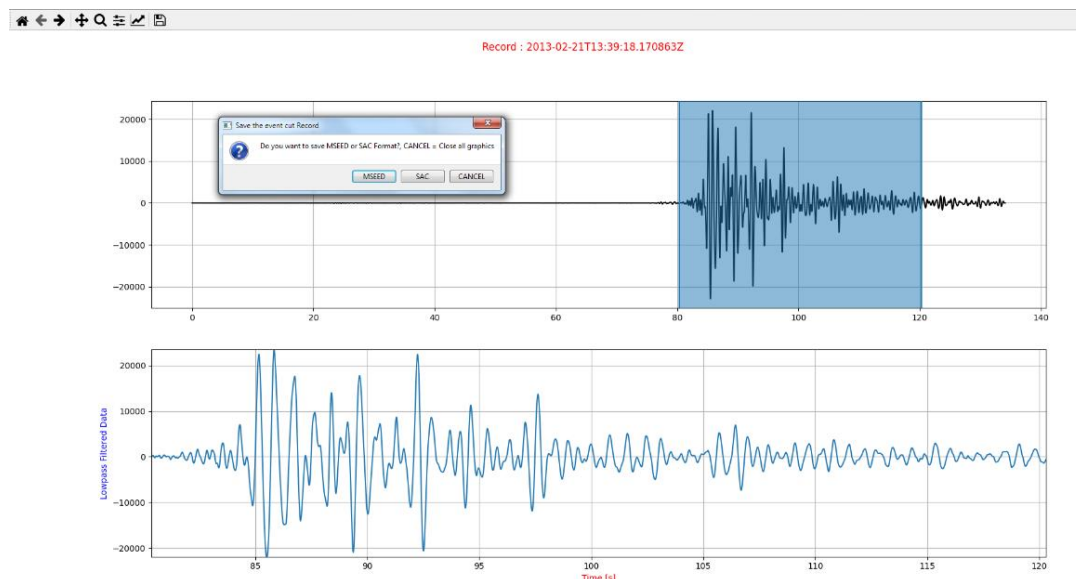


Fig. 33 The cut area is located at the bottom of the event. A dialog box appears asking which format, "MSEED" or "SAC", the record should be stored in.

The following image shows an enlargement of the window for selecting the types of formats with which the cut can be stored.

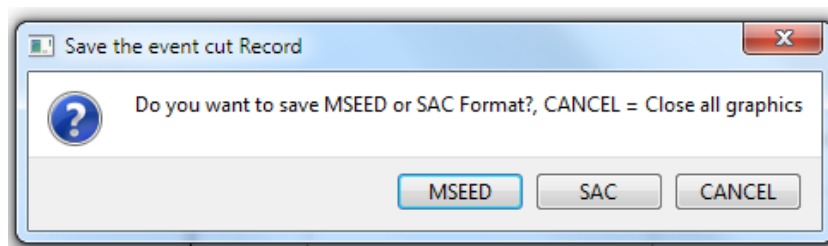


Fig. 34 Window for selecting the format in which the clipped events can be saved: "MSEED" and "SAC". By default, the "MSEED" format is selected. If you do not want to save the clipping, click the "CANCEL" button, which cancels the operation and closes the chart.

By selecting the format type by clicking on the corresponding button, a window appears indicating that the operation has been successfully executed.

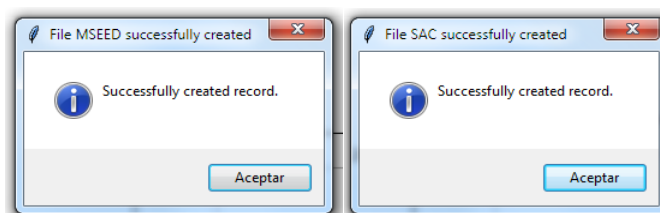


Fig. 35 Dialog boxes indicating the correct creation and storage of the file (MSEED or SAC).

## 6.2.- Example of results with selection of Start/End time and recording cutoff.

According to the entire process described above, the process for cutting an event within a seismic trace, by selecting an Initial/Final time, is very simple, and consists of the following steps:

- a) Check the "**With Time**" radio button to disable the filter dropdown and activate the time areas.
- b) Open or select a specific record (the file path is displayed as: Seismic Record Path to Upload). By default, the initial path is located in the root directory "C" of the PC, whether on a Windows or Linux system.
- c) Open or select the folder or directory where the clipped signal will be saved. (The folder path is displayed as: **Seismic Record Path to Save**).
- d) Enter the desired Start and End time parameters (For this example, we will use the same values as in the previous example, 80-120 s).
- e) Click the "**Plot Signal**" button to graph the record.

The selection of these parameters is shown in the following figure.

The screenshot displays the 'Cut and Save Seismic Signals - University of Granada (Ugr)' application window. The interface is organized into several sections:

- Filter and Time Selection:** Includes a 'Filter type' dropdown (set to 'Please select'), 'Frequency (Hz)' and 'Filter Order' (By default = 4) input fields, and a 'Clean Inputs' button. Below this, the 'With Time' radio button is selected, and 'Trace number st(n):' is set to 0 (By default: st(0)).
- Time Parameters:** 'Frec Min (Hz):' and 'Frec Max (Hz):' input fields are present. 'Initial Time (s)' is set to 80 and 'End Time (s)' is set to 120.
- File Paths:** Under the heading 'Load files and paths - save records:', there are two sections:
  - Seismic Record Path to Upload:** A text box containing 'C:/Users/Usuario/Desktop/Data\_examples/EZ5VZ\_2004-11-01-1010-00\$\_g1.bin.Z.sac'.
  - Seismic Record Path to Save:** A text box containing 'C:/Users/Usuario/Desktop/'.
- Bottom Section:** Contains a 'Go Back' button, a 'Warning' message (Make sure that the upload directories and storage are selected.), a 'Wait: Time to plot:' message (can be long. After the calculation, the graph will be displayed.), and an 'Exit' button.

Fig. 36 Example of inputs for a selection through start and end time.

The resulting graph consists of a single chart with the event already clipped. This process, as indicated on the main screen, may take some time, depending on the size of the record. For large records exceeding one hour, it is recommended to use the clipping process with filters, as it is much faster. The following figure shows this result with the dialog box for saving according to the desired format.

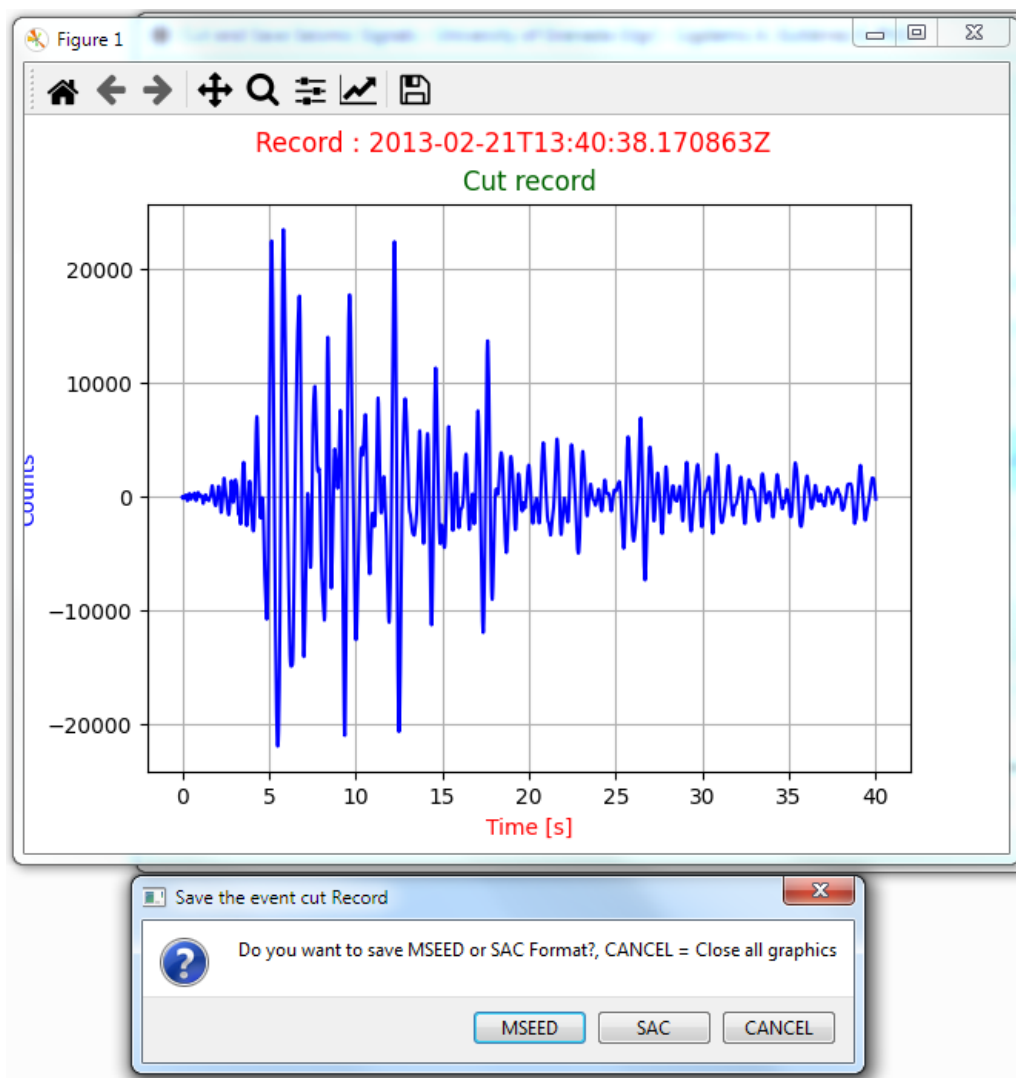


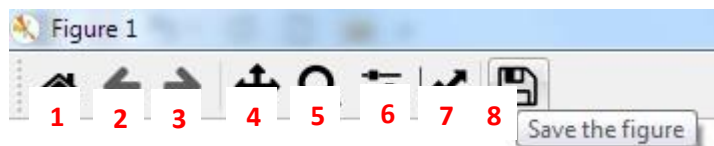
Fig. 37 Graphic output resulting from clicking on the “Plot Signal” button, with the dialog window to choose the “MSEED” or “SAC” format type that the user indicates.

The event is observed in the single window according to the assigned time (40 seconds).

**IMPORTANT NOTE:** The events presented here, whether by filter selection or time, are displayed according to the filtered signal. However, the stored event corresponds to a snippet of the original recording, exactly as it appears in the read format (unfiltered). Therefore, the snippet will be similar (unfiltered) when saved. The user can then apply filters to perform any further analysis they deem necessary.

## 7.- Toolbar of Graphs (Matplotlib Library).

In the construction of graphs, the *Matplotlib* library's graph screen has a set of very useful tools that allow you to visualize, edit, and save graphs in various formats. At the top of the Matplotlib graph screen that appears when a graph is created, there is a toolbar similar to the following:



From left to right, the icons representing the actions to be performed are:

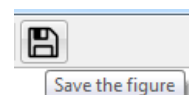
1. [Reset original view](#): Restores all graphs to the initial state.
2. [Back to previous view](#): Preview of the selected graph.
3. [Forward to next view](#): Forward view of the next image.
4. [Left button pans, Right button zooms, x/y fixes axis, CTRL fixes aspect](#): Moves the graph and axes left or right, zooms in or out, fixes the x/y axis, and fixes the aspect with CTRL.
5. [Zoom to rectangle](#): Through a rectangle, zooms in on the selected graph.
6. [Configure subplots](#): Configuration of subplots (Borders and spacings).
7. [Edit axis, curve and image parameters](#): Editing the parameters of the graph. Select the axes or graph and edit elements such as title, coordinates (X, Y), and curve parameters (lines, markers) in styles, colors, and size.
8. [Save the figure](#): Saves the graph in several formats.

This document does not delve into each of them; it only highlights the use of those that are generally more commonly used, such as (1, 2, 5, 7, and 8).

In the above graphs, the use of the zoom tool (5) has been observed. Tools 2 and 3 allow zooming in or out individually for each graph. Option 1 allows restoring all elements or subplots of the graph to their initial values (*each individual graph or part of the window*). As for option 8, it allows saving the graph in various formats. The rest are straightforward, and it is up to the user to explore each of them. Now, the processes for "*editing*" and "*saving or storing*" the graphs (*Numbers 7 and 8*) are detailed below.

### 8.1.- Saving Graphs.

*The process of saving graphs is very simple. Click on the icon of tool number 8 (Save the figure).*



This opens an explorer window, similar to those in Windows (depending on the language or system used), where you can select the folder or directory where the graph will be saved.

Additionally, provide a name and select the desired format type. This can be done at the bottom of the explorer window (red circle in the image), where various format types available for saving are selected. The screen resembles the following.

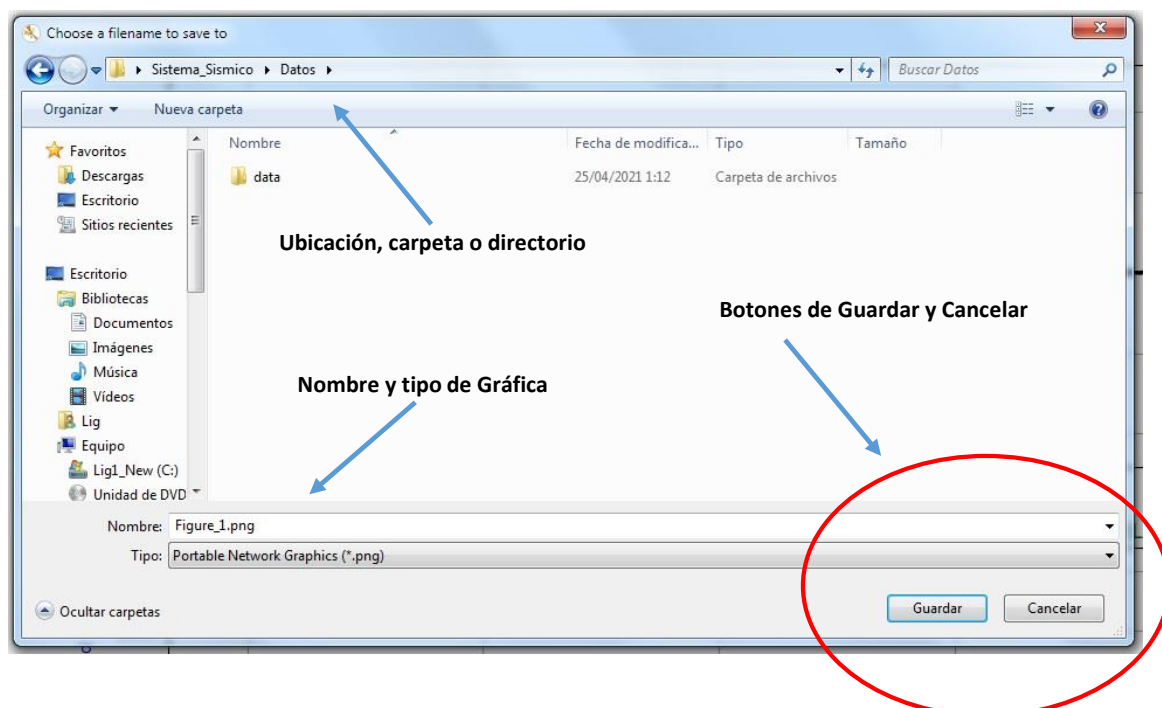


Fig. 38 Screen that allows you to save the graph, selecting a name and choosing various format types. "Save" and "Cancel" buttons are provided to complete or cancel the process.

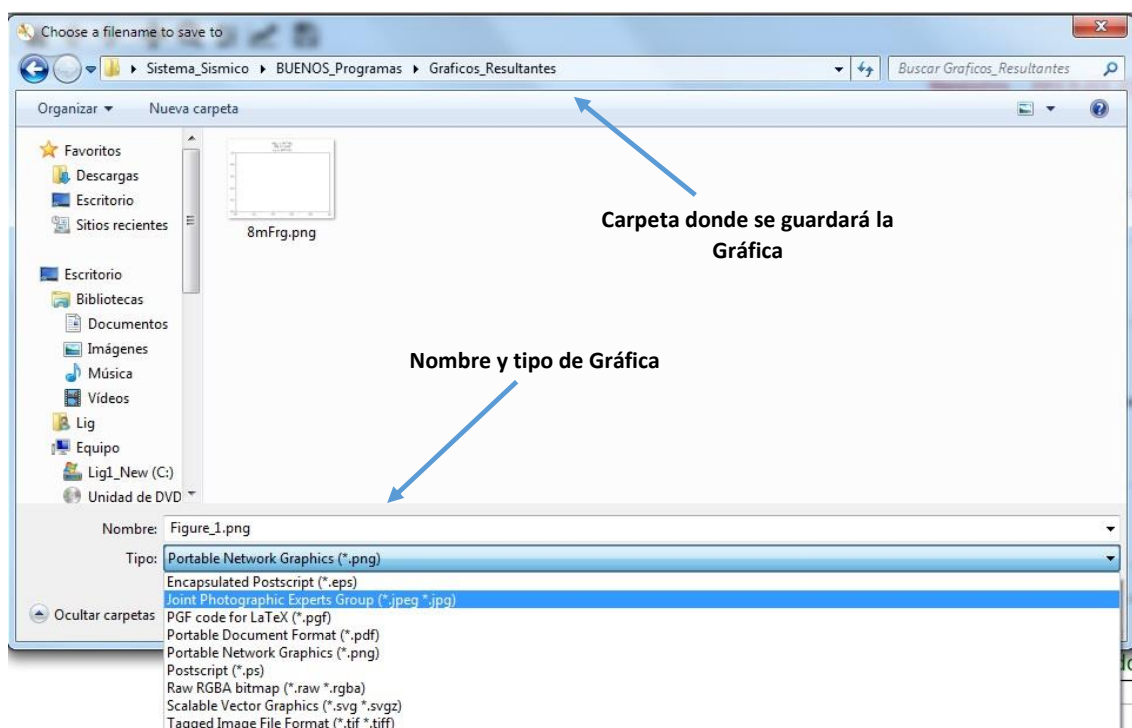


Fig. 39 Screen where you can see the types of formats available to save the graph.

The previous figure shows a list of the available file formats, the following image presents this list in more detail:

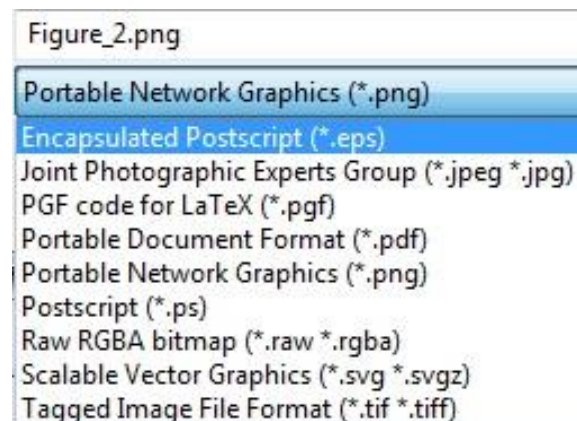
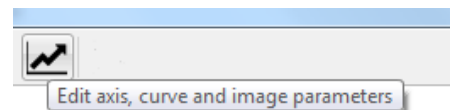


Fig. 40 List of available formats for saving the graph generated by the analysis.

Once you have selected both the name and the desired file format, and the location of the folder or directory where the graph will be saved, click the "Save" button (See Fig. 40), and the graph will be stored and available for further use as needed.

## 7.2.- Editing Axes and Images in Graphs.

Through the "Edit" button, point 7 (See page 27) of the graph toolbar ([Edit axis, curve and image parameters](#)), it is possible to edit or modify the parameters of the axes, images, and curves of the graphs.



For example, to modify the parameters of the image of a spectrogram, click on this command button. A "Customize" dialog box appears, indicating which of the "axes" in the graph areas you want to edit or modify. After selecting, click the "OK" button. This dialog box is similar to the following.

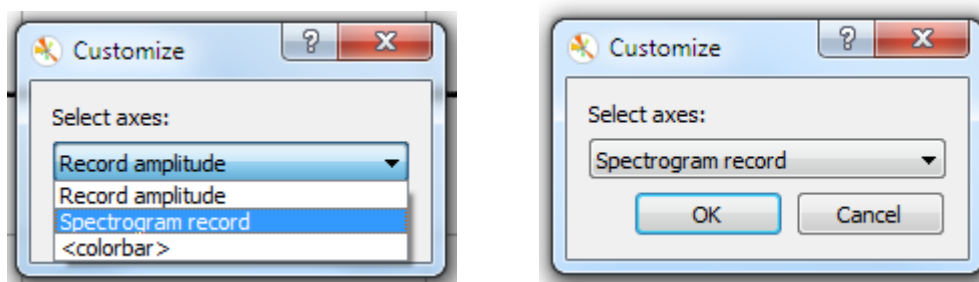


Fig. 41 Customize dialog box, the spectrogram axis has been selected.

Once the desired axis is selected, and the "OK" button is clicked, a new window with the options in the figure is presented. Here, various values of the selected axis are edited, in this case, the spectrogram (*Axes and Images*). The dialog box is as follows.

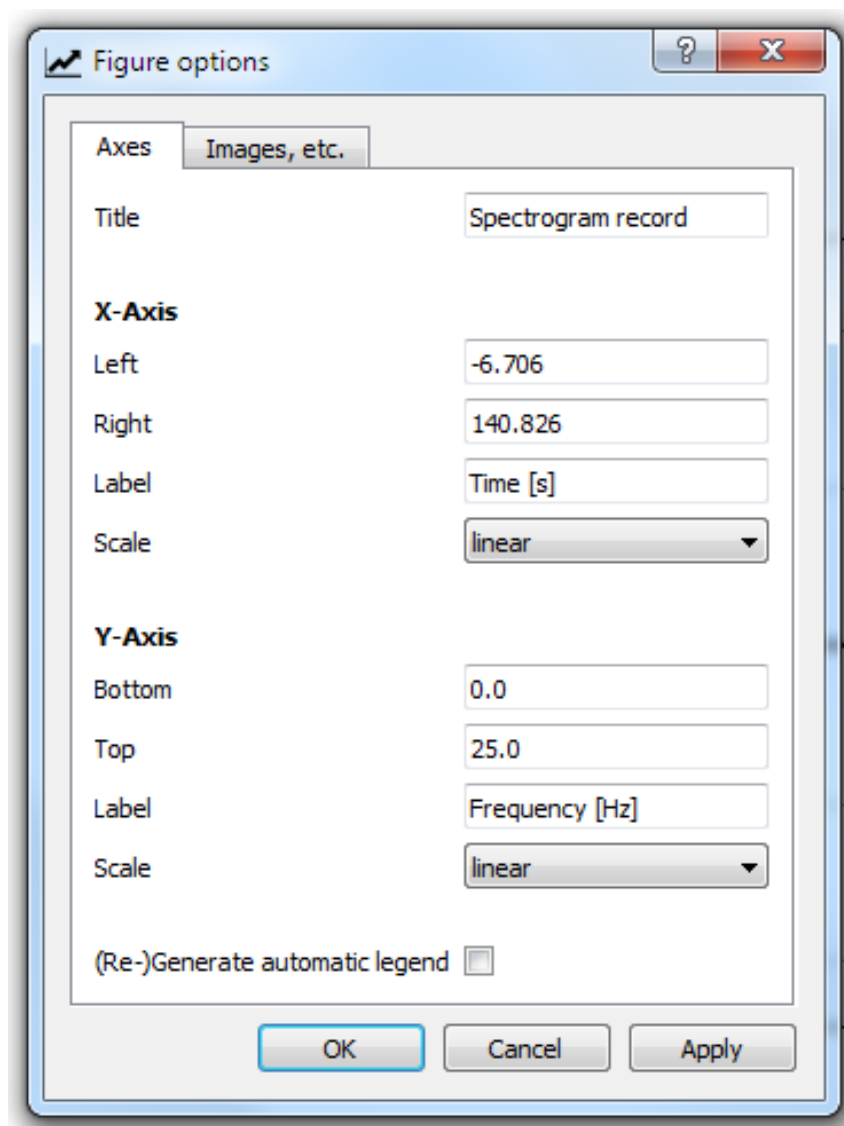


Fig. 42 Dialog box for editing options for the title and axes of the figure.

In this "[Axes](#)" section, as you can see, you can edit or modify the values or parameters of the title and the "[X](#)" and "[Y](#)" axes of the graph. For our example, we want to modify the image, so we will select the tab indicating this option. The image presented is as follows.

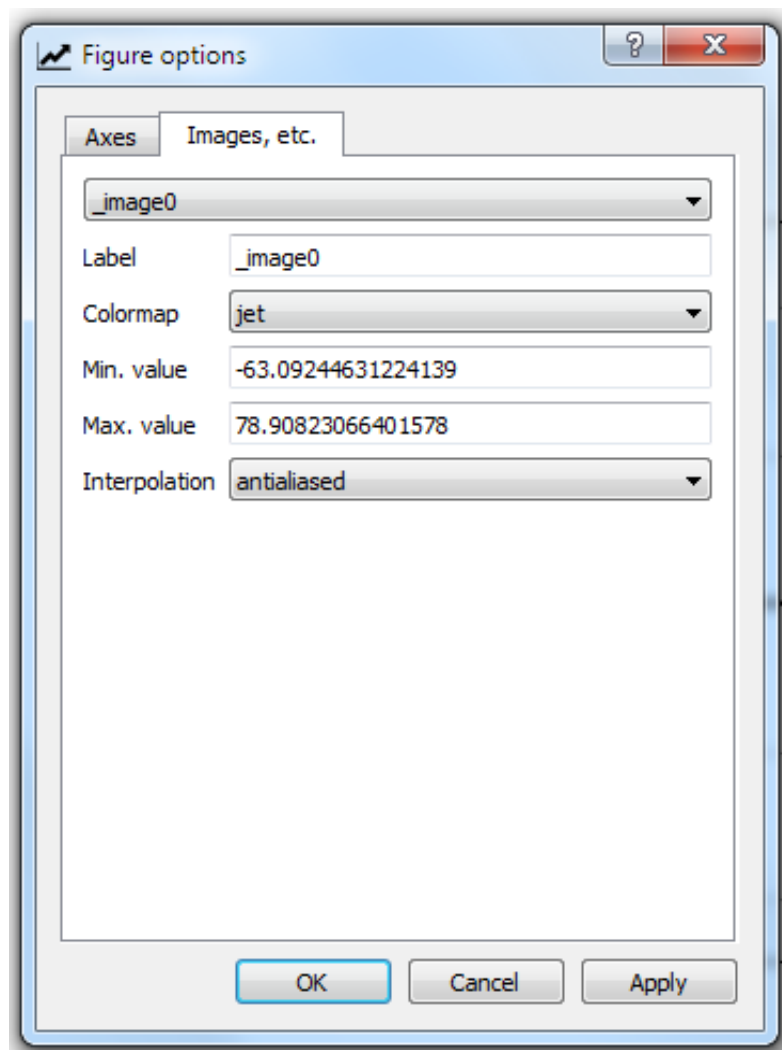


Fig. 43 Dialog box for editing options for image parameters.

As seen in the image, various parameters can be modified, including labels, the color map or "*Colormap*", used in the spectrogram, minimum and maximum values, and interpolation. The default value for the "*Colormap*" is set to "**jet**". The minimum and maximum values for this color map and the interpolation used are assigned by default to the image, but they can be modified according to the operator's interest.

The list of editable parameter values for both "*Colormap*" and "*Interpolation*" is presented in the figure on the next page.

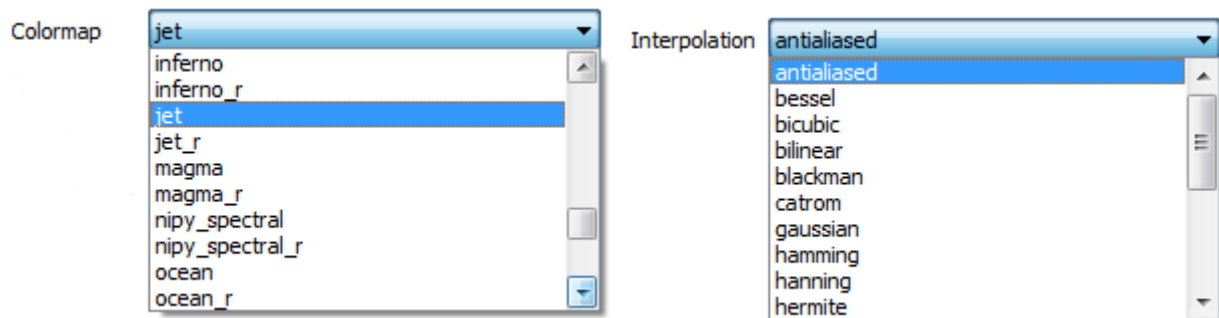


Fig. 44 Dialog boxes for editing some of the parameters of "Colormap" and "Interpolation" to select in the graph.

**CONCLUSION:** The system is designed to be easy to use, access, and understand. A user-friendly interface offers reliable technological support to the human operator in the analysis of seismic records, both tectonic and volcanic. The simplicity of this first version lies in its single module, which includes several filter analyses and the process for cutting a segment or event within the record, for subsequent saving in MSEED or SAC format. In later versions, additional modules may be added, containing various types of analysis to further the study and research of the scientific community.

### Acknowledgments:

This software and its documentation are the result of research from Spanish projects:

- a) PID2022-143083NB-I00, "LEARNING", funded by MCIN/AEI /10.13039/501100011033
- b) JMI and LG were partially funded by the Spanish project PROOF-FOREVER (EUR2022.134044)
- c) PRD was funded by the Ministerio de Ciencia e Innovación del Gobierno de España (MCIN), Agencia Estatal de Investigación (AEI), Fondo Social Europeo (FSE), and Programa Estatal de Promoción del Talento y su Empleabilidad en I+D+I Ayudas para contratos predoctorales para la formación de doctores 2020 (PRE2020-092719).
- d) Spanish Project PID2022-143083NB-I00 founded by MCIN/AEI/10.13039/501100011033 and by FEDER (EU) "Una manera de hacer Europa".

PLEC2022-009271 "DigiVolCa", funded by MCIN/AEI, funded by MCIN/AEI/10.13039/501100011033 and by EU «NextGenerationEU/PRTR», 10.13039/501100011033.

**END of the document.**

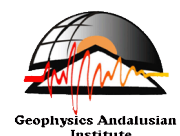
**Ligdamis A. Gutiérrez E. PhD.**

Theoretical and Cosmos Physics Department, Faculty of Sciences.

Geophysics Andalusian Institute.

Granada University (Ugr)

Granada, Spain – 2023



## APPENDIX A

### A1.- Installation of Python and Additional Libraries

#### A1.1. Package Content.

The main folder “*Analysis\_System\_1*”, It contains two folders with the programs (codes and interfaces in English and documentation in Spanish and English), organized as follows:

- a) Folder “*Reliable\_set\_tools\_system\_1*”: (*seismic analysis system assembly*). This folder must be copied to “*My Documents*”. Contains the following elements:
  - a. Subfolder: “*Images*” ” Images necessary for program interfaces.
  - b. Program: *Menu.py*. Startup program, presentation and calling of individual modules.
  - c. Program: “*Cut\_1.py*”. Main analysis program with filters, for cutting events within a seismic record.
- b) Folders (*Document\_ES* and *Document\_EN*, depending on the language; *English* or *Spanish*). For *English* “*Document\_EN*”: It consists of the following items:
  - a. User Manual for the module: “*6\_Manual\_Cut\_Save\_Signals\_Events\_EN.pdf*” in PDF, written in English, with the necessary documentation for the use of the system interfaces. In addition, the rest of the manuals for the other modules (1-10) that make up the system are found.
  - b. “*Initials Requirements.txt*” file. File containing the libraries needed to be installed on Windows through “Pip”, once Python is installed.
  - c. File “*README.txt*”: File with general instructions for system installation.
  - d. File “*Set\_tools\_System\_1\_1.bat*”, batch processing executable file. It must be copied to the desktop, from there by right clicking “run as administrator”, it will start the system by calling the main menu. The file will automatically search for the startup program (*Menu.py*) that is located in the “*Set\_tools\_System\_1\_1*” folder that has been previously copied to “*My Documents*” and will start Python, executing said program.

The system has all the elements (*programs and interfaces*) in English, except for the user manual, which is written in both Spanish and English. To install on Windows, two main actions should be taken after downloading and extracting the “. Rar” files. The first is to copy the entire folder (a) to the “*My Documents*” folder on the PC.

- a) From the main folder (*Analysis\_System\_1*), copy the subfolder “*Document\_EN*” to “*My Documents*” in Windows.
- b) Copy the file “*Set\_tools\_System\_1\_1.bat*”, from the “(Document/Document\_ES or Document\_EN)”, depending on the version (Spanish or English), to the Windows desktop.

This ensures the proper use of the program. Now, we will proceed with the installation of the Python language and additional Python libraries on Windows.

#### A1.2.- Installing Python on Windows.

Python is an interpreted, multi-platform, and multiparadigm programming language (*it works on various operating systems, including Windows, Linux, and Mac*), utilizing two or more programming paradigms within a program-object-oriented, reflective, imperative, and functional.

In addition, Python can be enriched by a large number of programming modules, libraries, packages, or libraries installed through its package manager, "**Pip**." On Linux, the Python program and its manager "**Pip**" are installed together with the operating system. In Windows systems, however, where Python is not a native language, it is necessary to install this language beforehand by downloading the appropriate version from the Python distribution website at the following address: <https://www.python.org/downloads/>

On the website, the correct version should be selected based on the type of operating system on the computer, including whether it is 32 or 64 bits.

To be installed on both 32 and 64-bit systems, it is essential to note that this document and the software were created with the version available at that time, which was "[Python 3.8.6](#)", and many more versions have emerged since then. A more modern and adaptable version to the software (recommended) is "[Python 10.10](#)".

Users need to check if more advanced versions do not interfere with some of the installed libraries, such as "[Obspy](#)," for example. This is because everything related to Linux systems is constantly changing with updates that Python and Linux-based systems make. It is advisable to visit the website and download the most stable or tested updated version of Python that works well with this software.

Once downloaded, run it as an administrator (*right-click and "run as administrator"*), and the software installation wizard will guide you through the necessary steps (*just follow the instructions*).

The process takes only a few minutes. It is "recommended" to indicate during the process, when asked, to include an access path in the system's "**Path**" so that Python can be accessed from any location in Windows. If this is not done during the installation process, it must be done manually by modifying the environment variables (*more complicated*) to include the path from where Python is installed. This will not be necessary (*if indicated at the beginning*) through the installation wizard.

### **A1.3.- Installation of Additional Libraries.**

The next step is to ensure that Python and its file manager or package manager (**Pip**) have been installed correctly. "**Pip**" (*file and library manager*) is crucial because it allows the installation of additional libraries that Python needs to run the created programs. To do this, open the Windows console window, or "[CMD](#)." The [CMD](#), or command prompt, is a command-line interpreter.

Accessing the CMD is possible by typing, searching for the Windows logo key (a window), located between the "[Ctrl](#)" and "[Alt](#)" keys at the bottom left of the keyboard. Pressing this key, plus (+) the letter "[R](#)" key, will open a "[Run](#)" program window, similar to the following.



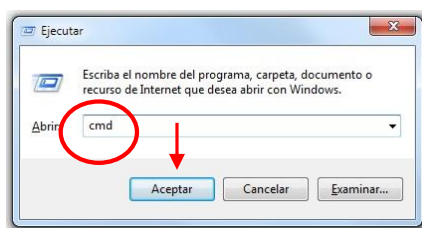


Fig. A1 Screen run in Windows. In the **red circle**, type "**cmd**" and click "**OK**."

As seen in the figure above, type "**cmd**," click "**OK**," which will open the Windows command prompt window.

Another way to do this is at the bottom of the desktop, in (W7) or next to (W10) the Windows "**Start**" button. There is the search section, indicated by the magnifying glass icon. This indicates a search for programs, similar to the following.

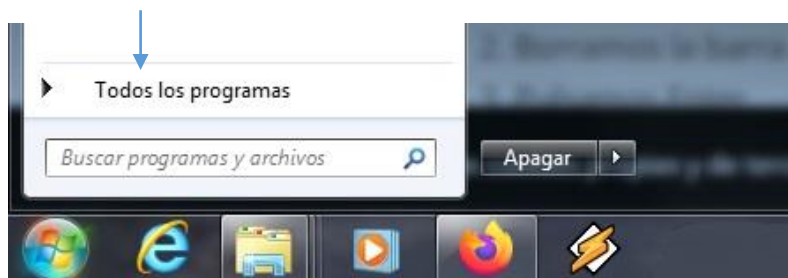


Fig. A2 Windows Program Search Screen.

In the box that says "Search programs and files" (Windows 7) or "*Type here to search*" (Windows 10), type "**cmd**" as well. This action or the previous one will bring up the Windows command prompt (**CMD**), similar to the following (W7).

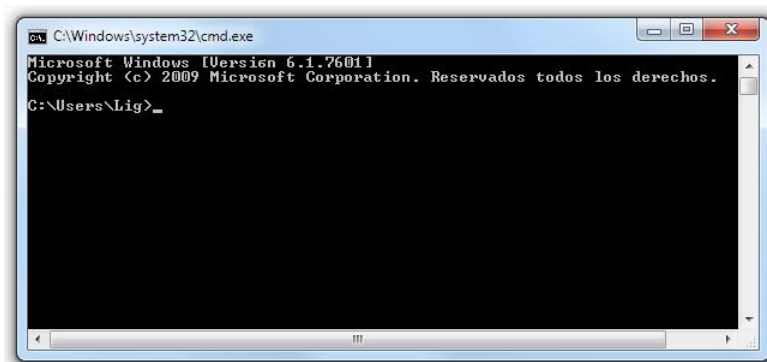


Fig. A3 Windows 7 Command Prompt (CMD) Screen.

The same applies to versions: Windows 10 (W10) or Windows 11 (W11).

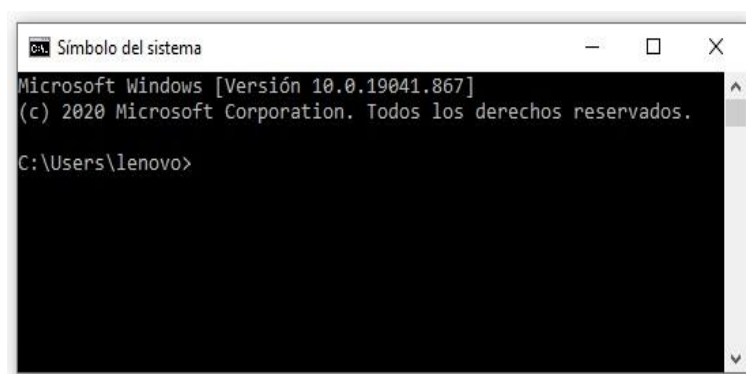


Fig. A4 Command Prompt (CMD) Screen in Windows 10.

Once on this screen, to verify that both Python and its package manager "[pip](#)" have been installed correctly, type the following commands: [Python -V](#), and to check "[pip](#)," type: [pip -V](#). This is shown in the following figure.

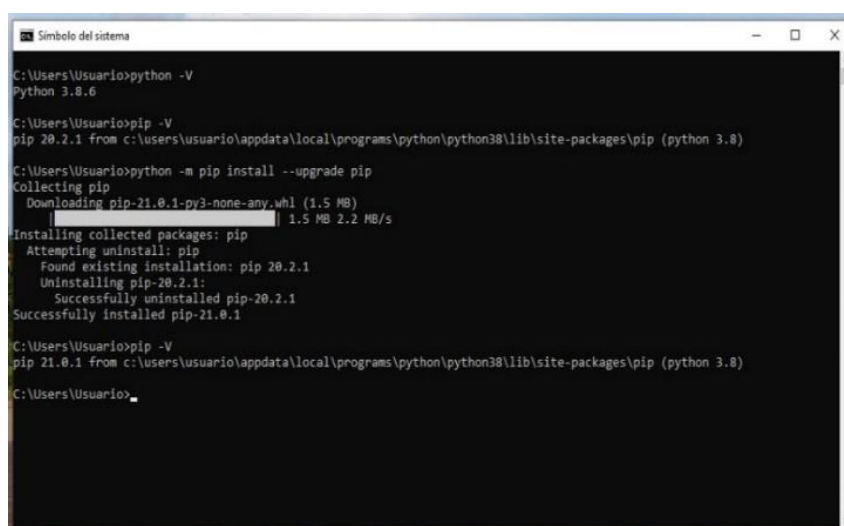


Fig. A5 CMD Screen, indicating Python and pip versions in Windows.

The output of typing "[-V](#)" in Python indicates invoking the installed version. In this case, it can be seen that it is "3.8.6" (*The library set is compatible with 3.10.10 version*). This has been possible from any location in the system because the Python script has been installed, remember, in the "**path**" or route found in the system's environment variables. Also, after typing "[pip -V](#)", it can be seen that the Pip version is "20.2.1." At this point, it is recommended to update this version since, by default, "[Pip](#)" is installed along with "[Python](#)", but it does not install the latest or most up-to-date version. To do this, in the CMD window or console, type the following command (*Windows/Linux*): On Windows, type "[python](#)," and on Linux, type "[python3](#)".

**Windows:** > [python -m pip install --upgrade pip](#) | **Linux:** \$ [sudo python3 -m pip install --upgrade pip](#)

This indicates that "[Pip](#)" will be updated to its most recent version (*On Linux, as a "superuser," i.e., with "sudo" at the beginning*). It is shown in the following screen.



```
Símbolo del sistema
C:\Users\Usuario>python -V
Python 3.8.6

C:\Users\Usuario>pip -V
pip 20.2.1 from c:\users\usuario\appdata\local\programs\python\python38\lib\site-packages\pip (python 3.8)

C:\Users\Usuario>python -m pip install --upgrade pip
Collecting pip
  Downloading pip-21.0.1-py3-none-any.whl (1.5 MB)
    | 1.5 MB 2.2 MB/s
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 20.2.1
    Uninstalling pip-20.2.1:
      Successfully uninstalled pip-20.2.1
  Successfully installed pip-21.0.1

C:\Users\Usuario>pip -V
pip 21.0.1 from c:\users\usuario\appdata\local\programs\python\python38\lib\site-packages\pip (python 3.8)

C:\Users\Usuario>
```

Fig. A6 Screen showing the update and verification of the new version of pip in Windows.

As can be seen, when typing again (**pip -V**), once Pip is updated, the version is 21.0.1. With this, Python and Pip are already installed and updated. Pip, as mentioned, is very important because with this manager, all the necessary libraries and packages are installed so that Python applications can be executed correctly and without errors. To use the system, you must proceed to install the necessary packages or libraries through Pip.

Next, we will proceed to explain how, in a simple and completely automatic way, the most commonly used and general libraries that Python needs will be installed on the system. Libraries such as, for example, "*obspy*," which is the open-source library or software based on Python for processing seismological data. Also, "*matplotlib*," which is a library for generating graphics from data contained in lists or arrays in Python and its mathematical extension "*NumPy*," among others, which the system needs for its execution (*See Annex B*).

#### A1.4 Automatic Installation of Libraries on Windows and Linux from PIP.

The advantage of having already installed and updated Pip in Windows is that you can install all the libraries that Python needs to run the system.

Additionally, in the "*Document*" folder, the "*Readme.txt*" file contains instructions for this installation. So the user only needs to follow the instructions, and the necessary packages will be installed on the computer (PC) automatically by Pip, both on Windows and Linux. The required libraries are in the file called "*Initial\_requirements.txt*", included in the "*Document*" folder of the downloaded installation files and in **Annex B**.

In a Windows Command Prompt (**Cmd**) window, actions are taken for each of the commands indicated in the file, following the instructions. The installation should not present problems on Windows and Linux systems. If any library encounters an error during installation (*shown in red in CMD*), you should consult the documentation for that library or check if the correct or recommended version of Python is being installed (*version 3.8.6 and/or 3.10.10*). The installation on Linux systems (*See README.txt*) is similar and simpler. Copy the main folder to the desktop, the personal folder, etc. From that location, open a command prompt, and simply type:

“\$ *python3 Menu.py*” to start the system.

## APPENDIX B:

### INSTALL PYTHON LIBRARIES FOR THE PROPER FUNCTIONING OF THE SYSTEM.

1.- **PIP:** The **Pip** (*Preferred Installer Program*) is the package or package management manager used to install and manage software packages written in Python. When installing Python, Pip is installed by default. To check the version of Python or PIP, type the following in a console or CMD:

```
python -V / pip -- And to see the list of installed pip packages: -> pip list
```

Usually, you need to update the version of pip with which Python is installed. For this, type the following command in the command prompt (CMD). In Linux and Mac systems, "**sudo**" is placed at the beginning to indicate super-user permissions.

```
Python -m pip install --upgrade pip / (LINUX) -> sudo python -m pip install --upgrade pip
```

Once downloaded and installed, you can check the version again with the first command, and you will see that the version has changed and been updated. Now that pip is updated, we will proceed to install the necessary packages for Python to work correctly with the applications.

2.- **PyQt Installation:** This is a Python binding for the Qt library written in the C++ language. It is used for creating and using graphical user interfaces (GUI) in Python. Type the following in the command prompt (CMD).

```
pip install PyQt5 / (LINUX & Mac) -> sudo python install PyQt5
```

3.- **Matplotlib library Installation.** Matplotlib is the library that allows the creation and visualization of graphics. Type the following:

```
pip install matplotlib / (LINUX & Mac) -> sudo python install matplotlib
```

4.- Install the **Obspy** library. This library is for handling seismic signals. Type the following:

```
pip install obspy / (LINUX & Mac) -> sudo python install obspy
```

5.- Install **Thinter**: Thinter is a graphical user interface (GUI). Type the following:

```
pip install tk / (LINUX & Mac) -> sudo python install tk
```

6.- Install **quantecon**: This library is used for spectrum estimation, Periodogram, Fourier transform. Type the following:

```
pip install --upgrade quantecon / (LINUX & Mac) -> sudo python install --upgrade quantecon
```

7.- Update a library for **matplotlib**. To avoid problems with graphics, install the following:

```
pip install msvc-runtime / (LINUX & Mac) -> sudo python install msvc-runtime
```

8.- Install **easygui** for the graphical interface:

```
pip install easygui / (LINUX & Mac) -> sudo python install easygui
```

9 Install **PyWavelets** for CWT handling.

```
pip install PyWavelets / (LINUX & Mac) -> sudo python install PyWavelets
```

10.- Install **plotly** for handling and assisting with graphics along with Matplotlib.

```
pip install plotly / (LINUX & Mac) -> sudo python install plotly
```

11.- Install "**pyaudio**", for audio management. Python bindings for PortAudio v19, the cross-platform audio I/O library

```
python -m pip install pyaudio / (LINUX & Mac) -> sudo apt-get install python3-pyaudio
```

Finally, type "**pip list**" to see the installed libraries. Optionally, you can create a file called "**requirements.txt**" that will contain all the libraries that the PC will use. The "**requirements.txt**" file must be in the current directory. The instruction to do this is as follows:

```
pip freeze > requirements.txt
```