



Web Interface for Data Exploration
Version 2.0

Philippe Santenoise^{1,2}

¹*INRAE, UR 1138 Biogéochimie des Ecosystèmes Forestiers, 54280
Champenoux, France*

²*INRAE, UMR 1434 Silva, 54280 Champenoux, France*

December 2022

Contents

1	Quick introduction of WIDEa	3
2	WIDEa installation procedure	4
2.1	Preliminary informations	4
2.2	Auto tasks	4
2.2.1	Installer files	4
2.2.2	Command files	6
2.3	Manual tasks	7
3	Partition of WIDEa into several panels	8
4	Detailed description of P1	11
4.1	Data presentation	11
4.2	Data loading	12
4.3	Creating subsets of data	13
4.4	Graph type selection	16
4.5	Variable selection	17
4.6	Adding a model	20
5	Detailed description of P2	27
5.1	Graphic tab	27
5.2	Flag tab	33
5.3	Statistics tab	36
6	Detailed description of GA	38
6.1	Action buttons	38
6.2	Graph examples	40
	References	50

1 Quick introduction of WIDEa

WIDEa is a R-based software aiming to provide users with a multitude of functionalities integrated into a web interface (shiny application) to explore, clean and analyse “big” environmental and (in/ex situ) experimental data. More specific data can be used with WIDEa, such as data measured on a temporal scale and infrared spectral of near/mid regions. WIDEa requires no programming knowledge and no internet connection once installed on your computer.

Once the dataset is loaded, WIDEa offers the possibility to add a list of conditions to generate sub-data sets, the creation of new qualitative/quantitative variables (concatenation and writing functions of one or more variables), a fully interactive data visualization (multiple graph types, 2D/3D) and a simplified management of atypical data (manual selection on visualized data and classification by quality code). WIDEa is also designed as a decision support tool and allows to perform statistical calculations on visualized data (basic statistics, linear regression, hypothesis tests, etc.) and apply (non) linear mixed effects models (with weighted residuals or not) on loaded data to check residual assumptions and the model robustness.

In next sections, the manual explains how to install WIDEa using different operating systems (Windows, Mac, Linux), describes precisely all functionalities available on WIDEa and presents a few examples to help new users become familiar with them.

2 WIDEa installation procedure

2.1 Preliminary informations

WIDEa requires that a R version 3.5 or greater is previously installed on your computer (if not, see <https://cran.r-project.org/bin/>).

A total of 17 R packages are necessary for the proper functioning of WIDEa: shiny [4], shinyBS [3], shinyjs [1], shinyFiles [11], shinythemes [5], shinybusy [12], V8 [14], plotly [15], htmltools [6], htmlwidgets [16], bindrcpp [13], scales [17], data.table [7], arrangements [9], car [8], DT [18], colourpicker [2]. The installation of all R packages is scheduled during the WIDEa installation procedure.

Various procedures are proposed to users to install WIDEa. They are based on auto (two options : with either installer or command files) or manual (with R console) tasks. All required files to execute these procedures are available from <https://github.com/PhilippeSantenoise/WIDEa>. The procedure with auto tasks only applies to the following operating systems: Windows and Mac.

2.2 Auto tasks

2.2.1 Installer files

This option is performed using an installer file for Windows/Mac OS. The installer file is an exe/jar file named respectively **WIDEa_setup_win** and **WIDEa_setup_macos** (see github link in section 2.1). As no Developer ID is associated to the installer file, its execution can be refused by the operating system used. In this case, users can follow the other option described in the section 2.2.2.

The installer file for Mac OS is a jar file. It requires that a Java Development Kit (JDK) version is installed on your computer. Once a JDK version has been installed, the installer file must be executed with a right-click and by selecting **Open**.

The installer file executes a list of automated tasks for the two operating systems, as follows:

1. Paste the **WIDEa** directory (including R scripts, batch/bash scripts used by the installer file, WIDEa logo images and the license file) at the desired location in your computer;
2. (Windows only) Create the **WIDEa files** directory in user's documents and add to it a txt file including the R software location (R.exe) on your computer (essential to load WIDEa R scripts);
3. Install R packages mentioned in the section 2.1;
4. Copy/paste the WIDEa logo (**WIDEa_header_img**) in the R shiny library (www folder precisely) used as the header of the web interface;
5. (Windows only) Create a **WIDEa** shortcut (locations: WIDEa directory, Start Menu, Desktop) used to launch the web interface;
6. Create an uninstaller file (exe/jar file named **uninstall**) in the WIDEa directory (see the **Uninstaller** folder for Mac OS).

As no shortcut file is created by the Mac OS installer, the web interface is launched from the exe file named **launcher_macos** and located in the WIDEa directory. An optional task can be (manually) executed to change the appearance of this previous file, as follows:

- Open the png picture **WIDEa_icon** located in the WIDEa directory (see Image folder);
- Click on **Select All** in **Edit** tab and copy the selection;
- Select **Get Info** with a right-click on the exe file;

- Click on the picture in top-left of the **Get Info** window and paste the previous selection.

WIDEa can be uninstalled with an exe/jar file named **uninstall** for Windows/Mac OS. Otherwise, if R packages are not all installed, please execute the file named **install_Rpackages_win** (Windows) or **install_Rpackages_macos** (Mac) in the **Command** folder.

2.2.2 Command files

First, the **WIDEa** directory, available from the github repository (link given in the section 2.1), must be copy/paste at the desired location on your computer. After this, WIDEa can be installed by executing the Windows/Mac OS command file named respectively **WIDEa_install_win** and **WIDEa_install_macos** from the **Command** folder. Command files execute tasks 2 to 4 described in the section 2.2.1.

Once the installation procedure is finished, WIDEa can then be launched by executing the Windows/Mac OS command file named respectively **launcher_win** and **launcher_macos** from the **Command** folder. An optional task can be done to change the launcher appearance for Mac OS (explained at the end of the section 2.2.1) or create a shortcut file on your desktop for Windows. Several steps are necessary to its creation, such as:

- Select **Shortcut** after a right-click on the launcher file;
- Cut/paste the shortcut on your desktop;
- Select **Properties** after a right-click on the shortcut;
- Click on the **Change Icon** button;
- Select the ico picture named **WIDEa_icon** and located in the WIDEa directory (see Image folder);

- Validate the shortcut properties with the **Ok** button.

WIDEa can also be uninstalled as follows:

- Execute the Windows/Mac OS command file named respectively **uninstall_win** and **uninstall_macos** from the **Command** folder (deletion of all files corresponding to tasks 2 and 4 of the section 2.2.1);
- Remove the **WIDEa** directory.

2.3 Manual tasks

A list of manual tasks are necessary to install WIDEa on your computer, such as:

- Copy/paste the **WIDEa** directory, available from the github repository (link given in the section 2.1), at the desired location on your computer;
- Load the **install_packages** R script located in the WIDEa directory (see **R_script** folder) by using the *source* function in a R console;
- Follow the task 4 mentionned in the section 2.2.1 (WIDEa logo available from the **Image** folder).

After completing all tasks, the WIDEa launcher (given by the **WIDEa_launcher** R script, same location as the previous R script) can be executed by entering the following codes in a R console:

```
# Path corresponding to WIDEa directory (named s_WIDEa_path)
# A trailing slash is placed at the end of the path
s_WIDEa_path <- "enter/the/path/to/WIDEa/"
# Loading WIDEa_launcher R script (with the source function)
# The paste0 function is used to concatenate strings together
source(paste0(s_WIDEa_path, "R_script/WIDEa_launcher.r"))
```

3 Partition of WIDEa into several panels

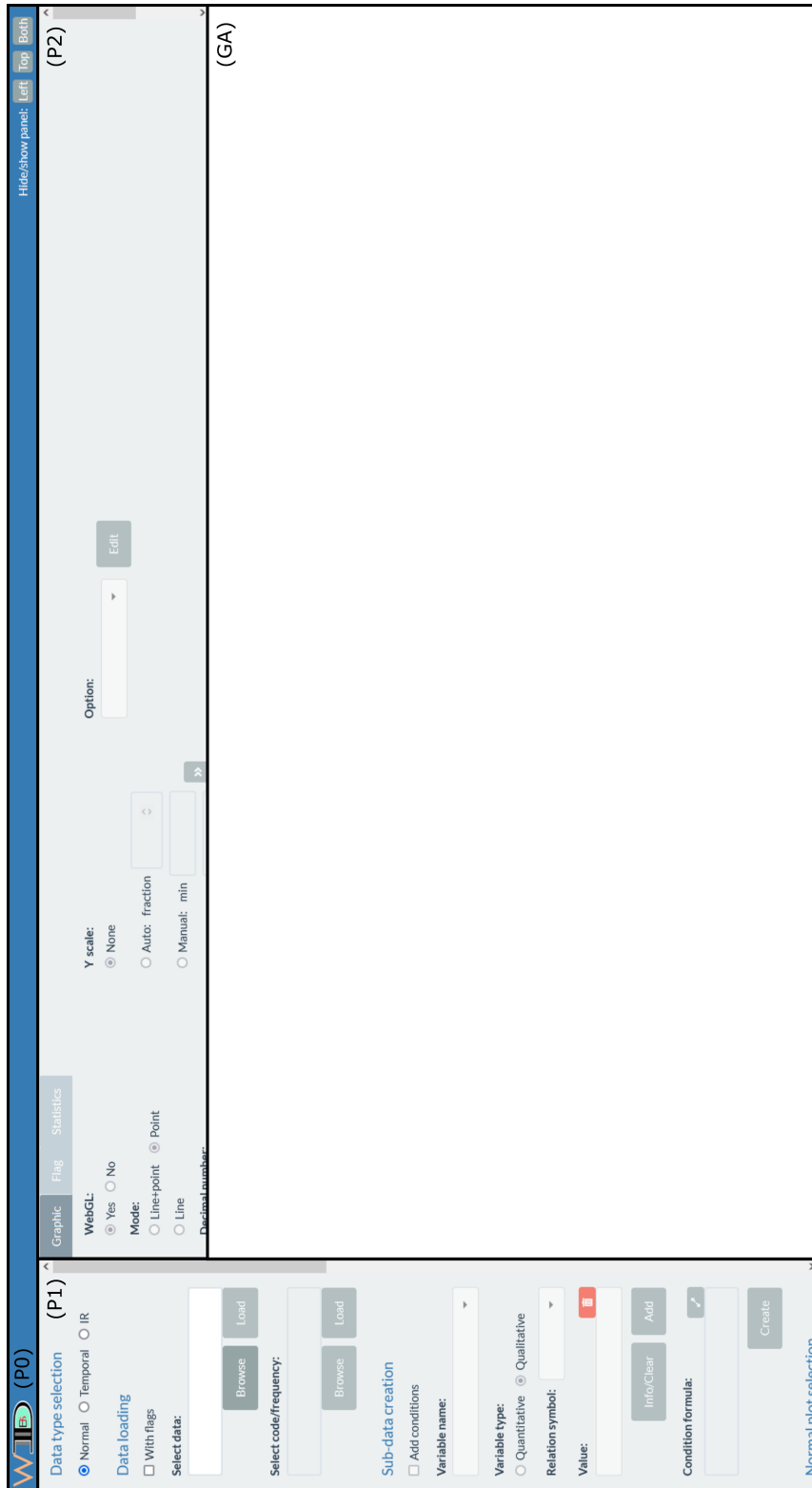


Figure 1. Web interface divided into 4 panels: P0, P1, P2 and GA

The web interface is divided into 4 panels called GA, P0, P1 and P2 (Figure 1). Panels are described below:

(GA) Panel dedicated to the graph area.

(P0) Panel with 2 functions:

- Inform users about the loading of actions performed on the web interface (Figure 2) ;
- Show/hide P1 and P2 to resize GA with the corresponding buttons (Left = P1, Top = P2 and Both = P1 + P2).

For the first function, it is advisable to wait until the end of the loading before performing a new action on the web interface.

(P1) Panel used to load data, create sub-data sets and new qualitative/quantitative variables, select the graph type and variables and display the selection in GA.

(P2) Panel containing 3 tabs with the following functions:

Graphic The tab allows changing basic graphics settings (label, color, opacity, point type/size, scale, etc.);

Flag The tab allows managing/saving atypical data (named flag data hereafter) manually selected on the graph with a left mouse click. The graph is then updated after saving flag data;

Statistics The tab is used to perform statistical calculations and display them on the graph.

Otherwise, each action performed on the web interface can cause a warning/error message window located at the bottom-right of GA and colored in yellow/red. In the next sections, “input controls” is used as a generic term to designate a set of inputs including text field, check box and radio/action button.



Figure 2. Loading animation

4 Detailed description of P1

4.1 Data presentation

The web interface accepts data file in txt/csv format with the column headers contained only in the first row and a dot character used as the decimal separator. As the function *fread* (data.table R package) is used to load data, the separator between columns is automatically detected from a set of 6 characters: white space, comma, tabulation, vertical bar, colon, semicolon. In addition, column names should not have spaces or special characters and missing values are designated with NA characters.

Variables are separated into 3 groups in the following sections: quantitative (numeric values and only integer values), qualitative (string values and only integer values) and date (time units). An integer variable can be designated as a quantitative/qualitative variable.

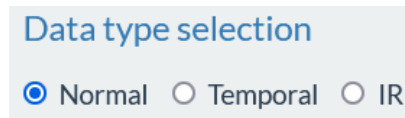


Figure 3. Radio button used to select the data type

A total of 3 data types can be selected from the radio button in Figure 3: normal, temporal and IR. One to two datasets (noted D_1 and D_2) are required depending on the selected type and are described as follow:

- Normal D_1 is a collection of quantitative/qualitative variables. An *ID* variable with unique values can be added to data and will be used to identify atypical values saved from Flag tab in P2 (row number if no *ID* variable precised).
- Temporal D_1 is composed of a date variable with unique values (used as an *ID* variable) and quantitative variables measured over the given time interval.

By considering codes $\%Y$, $\%m$, $\%d$, $\%H$, $\%M$ as units of years, months, days, hours and minutes respectively, the combination of time units allowed for the date variable is $\%Y\%m\%d$, $\%d\%m\%Y$, $\%Y\%m\%d\%H\%M$ and $\%d\%m\%Y\%H\%M$. In addition, the separator between time units is the following set of 6 characters: white space, h letter, dash, forward slash, colon, dot.

IR Type corresponding to absorption spectral data on near/mid infrared (NIR/MIR) region collected from a set of l_1 (> 1) and l_2 (> 1) equidistant frequencies (called also wave numbers in cm^{-1}) respectively. Each infrared region is studied separately and corresponds to 2 datasets.

D_1 contains spectra arranged in rows and qualitative variables used as supplementary information (ID variable and/or others). Column names assigned to spectra are frequencies, coded “ $M1$ ”, \dots , “ Ml_1 ” (resp. “ $N1$ ”, \dots , “ Nl_2 ”) for MIR (resp. NIR) region.

D_2 contains two columns named “*Code*” and “*Frequency*” and values are codes/frequencies of the whole spectrum.

In case the number of frequencies is reduced in D_1 related to spectra pre-processings (derivative and Savitzky-Golay methods, deletion of noisy frequency intervals, etc.), then missing frequencies will be automatically added using codes/frequencies in D_2 (except for frequencies lying to interval limits) in order to keep a break in line graph.

4.2 Data loading

Input controls in Figure 4 allow to:

- Add flag data related to D_1 (if existing: see section 5.2 for its creation from the Flag tab) by checking the corresponding box. The flag data will be loaded at the same time as D_1 ;

- Enter D_1 (D_2 resp.) file path in the text input field manually or with the browse button;
- Load D_1 (D_2 resp.) with the corresponding button once the text input field is filled.

The image shows a web interface titled "Data loading". At the top, there is a checkbox labeled "With flags". Below this, there are two distinct sections for data input. The first section, labeled "Select data:" and D_1 , contains a text input field and two buttons: "Browse" and "Load". The second section, labeled "Select code/frequency:" and D_2 , also contains a text input field and "Browse" and "Load" buttons. The interface is clean and uses a light gray background with blue text for labels and titles.

Figure 4. Input controls used to load all data (D_1 , D_2 and flag)

If a problem occurs while loading data, then the corresponding text input field will be colored in red (green in the opposite case) and an error message will be returned (e.g. incorrect path, missing variables for IR data type, flag data not found, etc.).

After loading D_1 , a warning message will be displayed if a flag data already exists but the corresponding box is not checked. In this case, the existing flag data will be replaced by another one if new values are saved during the current session.

4.3 Creating subsets of data

This section based on creating subsets of D_1 is only reserved to the normal/IR data type. A subset of data is defined by two elements: the declaration of conditions (can be unique) and the formula used to combine these conditions (the condition itself if unique).

Figure 5. Input controls used to create a subset of D_1 . The main part is dedicated to: (1) adding each condition individually and (2) writing a formula to associate all conditions.

The first element is available once the box used to add conditions is checked (Figure 5). Conditions are then built individually using a multitude of input controls such as: the variable name, the variable type (quantitative/qualitative), the relation symbol and the value. The variable type (radio button) is automatically assigned when the variable name is selected, except for integer variables. The last two input controls are dependent on the variable type. A total of 8 relation symbols (6 + 2 symbols for quantitative/qualitative variables respectively) are available, such as: = (equal to), != (unequal to), < (less than), > (greater than), <= (less than or equal to), >= (greater than or equal to), %in% (belongs to) and !%in% (does not belong to). The value input field is a list with the unique values of the selected variable (into ascending order) for qualitative variables and a text field which

only numeric values are allowed (dot character as the decimal separator) for quantitative variables.

After having filled in all the input controls, a condition can be added by clicking on the add button. A unique code is assigned to each condition added. The code starts with the lowercase letter *c* and is followed by the number of the condition introduced. All conditions added are kept in a data inventory (Figure 6) available by clicking on the info/clear button. One or more conditions can be deleted by selecting the corresponding row(s) of the data inventory (by clicking directly on it or by using the first two buttons below the data inventory) and pressing the clear button. The data inventory will then be automatically updated after the deletion of the selected condition(s).


Conditions added

Condition	Variable name	Relation symbol	Value
c1	Species	%in%	"Species 1"
c2	Site	%in%	"Site 1"
c3	Site	%in%	"Site 2"
c4	D	>	40
c5	H	<=	35

Deselect all
Select all
Clear
Close

Figure 6. An example of data inventory including 3 conditions on qualitative variables (*c1*, *c2* and *c3*) and 2 conditions on quantitative variables (*c4* and *c5*). The txt file named *data_tree* is used as D_1 (see psD_1 in the section 6.2). *D* and *H* in the second column correspond to the tree diameter (cm) and total height (m) respectively. The clear button is used to delete the selected conditions. All conditions can be selected/deselected with the corresponding buttons or individually by clicking on each row.

The second element is available as soon as a first condition is added. The text input field used to write the formula (Figure 5) must be filled in to create a subset of D_1 and must include all conditions of the data inventory called by their unique code. The *c1*

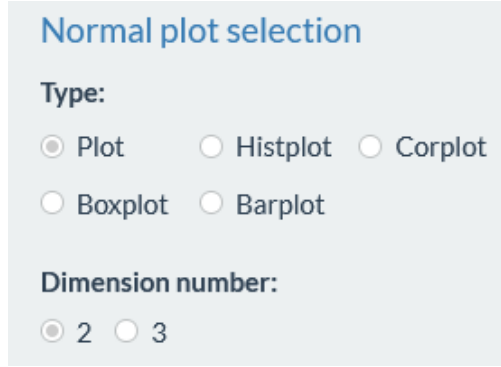
code is used as a formula if the data inventory has a single condition. The *AND* and *OR* boolean operators are used to associate conditions and are represented by the ampersand and the vertical bar characters respectively. Using the data inventory in Figure 6 as an example, a complex formula can be written (paranthesis character is allowed in the formula) as follows: $c1 \& ((c2 \& c4) | (c3 \& c5))$. The button  enlarges the text input field dedicated to the formula and displays the data inventory to facilitate its writing when a large number of conditions are added.

After completing the two elements, the subset of D_1 is built by clicking on the create button and an error will be returned if no match is found with the formula.

4.4 Graph type selection

Radio buttons shown in Figure 7 are enabled only for the normal data type and a total of 6 graph types is available: 2D/3D scatter-plot (type = plot and dimension number = 2 or 3, named 2D/3D plot in the following), box-plot (type = boxplot) with a linear method used to compute quartiles (“H&L-2” listed in Langford et al. 2006 [10]), histogram (type = histplot), bar-plot (type = barplot) and the correlation matrix (type = corplot) calculated from Pearson’s method (*cor* function). A simple left-click on each cell of the correlation matrix allows to create a combined graph in a new window such as a 2D plot and a histplot for the selected pair of variables. A linear regression and an equality test (hypothesis: slope = 0) based on a Student’s t-distribution is added to the 2D plot to check the linearity assumption.

A point-to-point graph (2D plot with connected points or not) is used to represent quantitative variables (NIR/MIR spectra resp.) over a given time (frequency resp.) interval for the temporal (IR resp.) data type. A range slider is added below the main graph to select a custom time/frequency interval.



Normal plot selection

Type:

☒ Plot
 ☐ Histplot
 ☐ Corplot

☐ Boxplot
 ☐ Barplot

Dimension number:

☒ 2
 ☐ 3


Figure 7. Radio buttons used to select different types of graphics for the normal data type. The 2D plot (default value) is fixed for the other data types.

4.5 Variable selection

The selection of variables is different when considering the case where a model is added (normal data type: calibration/validation) and will be described in more detail in the section 4.6. For the other cases, variables are selected from the following input fields named: *ID*, *X*, *Y*, *Z* and *Group* (Figure 8). Each input field is enabled/disabled depending on the selected data/graph type and accepts a certain type (quantitative, qualitative and date) and number of variables (Table 1).

X, *Y* and *Z* input fields only concern the normal/temporal data type and define variables to be plotted on the corresponding graph axes in GA. As NIR/MIR spectra data are automatically selected by combining D_1 and D_2 , all previous input fields are disabled for the IR data type. Moreover, other inputs controls are available to manipulate variables selected from *X*, *Y* and *Z* input fields or add a complementary information, such as:

- A function $f(x)$, $g(y)$ and $h(z)$ can be applied to quantitative variables using the Yes/No radio button (not available for corplot). This function must be known by the R software (does not require the loading of new packages) and its application differs according to the data type. In the case of the normal data type, all variables

entered into the corresponding input field are used to calculate a new variable using the function. This function is defined as mandatory when the number of variables is strictly greater than one. A unique code is assigned to each variable, starting with lowercase letters x , y and z respectively and followed by the number of the selected variable(s) (examples: $\text{sqrt}(x1) + x2$, $\log(y1 + 1)$, etc.). In the case of the temporal data type, the same function is applied to each variable of the Y input field. Variables are then designated by a same code corresponding to the lowercase letter y (examples: $\text{sqrt}(y)$, $\text{exp}(y)$, etc.). Codes can be found by clicking on the button  to help users write the function;

- A *concatenation* checkbox allows you to increase the number of qualitative variables to be filled in the X input field (one variable by default). A new variable named “.concat1.” is then created into D_1 when the display button is clicked and correspond to the concatenation of the selected variables. The concatenation option is not possible if this new variable already exists in D_1 (a warning message is returned after D_1 loading);
- The *format* of the date variable selected from the X input field needs to be specified and corresponds to 4 combinations of time units mentioned in section 4.1.

ID and *Group* input fields are optional (Yes/No radio button) and are only available for the normal/IR data type. The *ID* input field considers a variable described in section 4.1 and will be automatically filled (in a disabled state) if a flag data including an *ID* variable is loaded in the current session. In case the flag data are identified from row numbers (no *ID* variable), then the *ID* input field will be empty and disabled. The *Group* input field is used to color/split data into groups of the considered qualitative variable. The data splitting is generally displayed on the same graph, except for the corplot where a correlation matrix is calculated for each group using a new input field created in GA after clicking on the display button. A *concatenation* checkbox is also available next to this second input field and allows to create the “.concat2.” variable into D_1 .

These last two input fields are not available for the temporal data type because the role of the ID variable is already given by the date variable and the data coloration/splitting is performed from Y quantitative variables. In addition, the ID input field is only available for 2D/3D plot of the normal data type. The reason is that flag data are created from these two graph types (section 5.2).

Variable selection

ID:
☐ Yes ☒ No

Random: ☐ Yes ☒ No 🗑️

X:
☐ Concatenation 🗑️

f(x): ☐ Yes ☒ No ↕️

format:

Y: 🗑️

g(y): ☐ Yes ☒ No ↕️

Z: 🗑️

h(z): ☐ Yes ☒ No ↕️

Weighted residuals: ☐ Yes ☒ No
☐ with groups: 🗑️

variance function: ↕️

Group:
☐ Yes ☒ No
☐ Concatenation 🗑️

Clear Display

Figure 8. Input fields used to select a set of variables: ID (flag data identification for normal/IR data), variables attributed to X -, Y - and Z -axis and $Group$ (splitting D_1 into groups for normal/IR data). Input fields ($Random$ and $Weighted residuals$) colored in red are only enabled if a model is specified (discussed in details in the next section 4.6). The display (clear resp.) button is used to create (remove resp.) the graph in GA after filling in all the input fields.

Data/graph type	ID	X	Y	Z	$Group$
2D plot	1	≥ 1	≥ 1	0	≥ 1
3D plot	1	≥ 1	≥ 1	≥ 1	≥ 1
Normal boxplot	0	≥ 1	≥ 1	0	≥ 1
Normal histplot	0	≥ 1	0	0	≥ 1
Normal barplot	0	≥ 1	0	0	≥ 1
Normal corplot	0	0	> 1	0	≥ 1
Temporal	0	1	≥ 1	0	0
IR	1	0	0	0	≥ 1

Table 1. Summary of the availibility of ID , X , Y , Z and $Group$ input fields per data/graph type and the type/number of variables to be filled in. A zero value means that the input field is not available. Quantitative, qualitative and date variables are designated with a cell colored in red, green and blue, respectively. No type is required for the ID variable. A number colored in red means that the variable has unique values.

4.6 Adding a model

The web interface can be used to apply (not calibrate: see Zuur et al. 2009 [19]) a wide range of statistical models following a normal distribution: linear/non-linear mixed effects models with/without interaction between random effects and/or weighted residuals. Different results are provided depending on the chosen strategy: calibration (study of residual assumptions) and validation (observed vs fitted values). The assumptions about residuals (calibration strategy) are checked from a list of graphs, such as: the comparison of residual empirical-theoretical distributions (Q-Q plot: normality), (standardized or not) residuals vs fitted values/independent variables (homoscedasticity and independency). This list of graphs will be available from a new input field created in GA, once the model is integrated in P1 and validated with the display button.

The model integration is realised in 4 steps, briefly detailed below:

- Creating of model parameters data (D_3) file;
- Enabling input controls used to load D_3 ;
- D_3 loading;
- Model description (form, variables) using variable selection input controls.

The D_3 file format is the same as D_1 and D_2 files (section 4.1). D_3 contains a minimum of 2 columns named “*parameter*” and “*value*” corresponding to the list of parameters present in the model with their values. A code is assigned to each model parameter (Table 2) and the following Gompertz model of H-D relationship is used as an example:

$$y = 1.3 + (A1 - 1.3) \cdot \exp \left(-\exp \left(a2 \cdot \frac{a3 - x1}{A1 - 1.3} + 1 \right) \right) + \epsilon \quad (\text{Eq 1})$$

$$A1 = a1 + a1|re1:re2$$

With y the tree height (H in meter), $x1$ the diameter at breast height (D in cm), ($a1, a2, a3$) model coefficients (called also fixed effects coefficients), $re1:re2$ random effects of the interaction between the site and the species (2×2 levels) and $a1|re1:re2$ random effects coefficients (applied to $a1$) normally distributed with mean 0 and a certain variance. The residual term ϵ is also normally distributed with mean 0 and variance σ^2 weighted by the following function Φ :

$$\Phi(\sigma^2) = (\sigma \cdot x1^{d1|gr1})^2 \quad (\text{Eq 2})$$

With $d1|gr1$ are coefficients varying between sites (designated with $gr1$).

Values of the “*parameter*” column from the H-D Gompertz model are a , ($a|re, d|gr$) and σ codes (Table 2). Each ($a|re, d|gr$) code is duplicated by the given number of levels and new columns are added to D_3 to define different levels. These new columns are

then named with (re, gr) codes (3 columns in the present case: *re1*, *re2* and *gr1*). Table 3 represents D_3 created from the H-D Gompertz model.

Code	Description	Example
<i>a</i>	fixed effects coefficients	<i>a1, a2, a3</i>
<i>re</i>	qualitative variable used in random effects	<i>re1, re2</i>
<i>d</i>	coefficients of the residual weighting function	<i>d1</i>
<i>gr</i>	qualitative variable used in the residual weighting function	<i>gr1</i>
<i>(a re, d gr)</i>	coefficients varying between qualitative variable levels (random effect coefficients or not)	<i>a1 re1:re2, d1 gr1</i>
<i>sigma</i>	residual standard deviation	<i>sigma</i>

Table 2. Description of model parameter codes illustrated with an example given by the H-D Gompertz model: Eq 1 and Eq 2

The *sigma* parameter is used to calculate standardized residuals and check the normality assumption (Q-Q plot) in case the calibration procedure is selected. As only fitted values are calculated from the validation procedure, this parameter is not necessary and can be removed from D_3 . Furthermore, if the *sigma* parameter is missing in D_3 for the calibration procedure, the Q-Q plot will be removed from the list of graphs and residuals will be not standardized.

After D_3 creation, a series of actions must be executed to enable input controls related to its loading (detailed in Figure 9). A first part is to follow steps presented in the section 4.2 using the normal data type and selecting a 2D plot as graph type. D_1 represents the model data (dependent/independent variables and qualitative variables assigned to *re* and *gr* codes) and a flag data can also be loaded (section 5.2 for its creation). A second

part is to select the model strategy (calibration/validation) with the radio button from the normal graph type selection.

parameter	value	gr1	re1	re2
$a1$	35.7	NA	NA	NA
$a2$	1.75	NA	NA	NA
$a3$	10.05	NA	NA	NA
σ	8.02	NA	NA	NA
$d1 gr1$	-0.24	Site 1	NA	NA
$d1 gr1$	-0.25	Site 2	NA	NA
$a1 re1:re2$	5.86	NA	Site 1	Species 1
$a1 re1:re2$	-4.79	NA	Site 1	Species 2
$a1 re1:re2$	-2.14	NA	Site 2	Species 1
$a1 re1:re2$	1.07	NA	Site 2	Species 2

Table 3. Example of D_3 created from the H-D Gompertz model: Eq 1 and Eq 2. D_3 size increases with the model complexity: only fixed effects model (red), adding of a residual weighting function (green) and random effects (blue). Empty cells are represented by NA characters (in R).

A first check is carried out when loading D_3 about: parameter code assignment, column names and the presence of the σ parameter for the calibration strategy. A warning/error message will be displayed in GA if the σ parameter is missing or the D_3 structure is incorrect, respectively. In case D_3 is loaded without any error, a certain number of input controls are then enabled depending on the model strategy (calibration/validation).

Figure 9. List of necessary actions to enable D_3 : (1) select the normal data type, (2) load D_1 and select the (3) 2D plot as graph type and (4) the model strategy. The txt file named *data_tree* (psD_1) are described in the section 6.2.

These input controls are illustrated with the H-D Gompertz model in Figure 10 (calibration strategy) and are described below:

- The *Random* input field is optional (Yes/No radio button) and allows to add qualitative variables associated to *re* codes ;
- X and $f(x)$ input fields are used to introduce the independent variables (required type: quantitative) and the model form respectively;
- The Y input field define the dependent variable (required type: quantitative) and a $g(y)$ function can be specified. Only one variable can be selected from the Y input field;

- The *Weighted residuals* input controls is also optional (Yes/No radio button) and only applies to the calibration strategy. A first input field considers the variance function and a second input field (enabled with a checkbox above) allows you to designate qualitative variables associated to *gr* codes.

Variable selection

ID: ☐ Yes ☒ No

Random: ☒ Yes ☐ No

Site Species : re1, re2

X: ☐ Concatenation

D

f(x): ☒ Yes ☐ No Eq 1

1.3 + (a1 + a1|re1:re2 - 1.3) * exp

format:

Y:

H

g(y): ☐ Yes ☒ No

Z:

h(z): ☐ Yes ☒ No

Weighted residuals: ☒ Yes ☐ No

☒ with groups:

Site : gr1

variance function: Eq 2

(sigma * x1^d1|gr1)^2


Group: ☐ Yes ☒ No

☐ Concatenation

Clear Display

Figure 10. Integration of the H-D Gompertz model using the calibration strategy. Model variables (*re1*, *re2*, *x1*, *y* and *gr1*) are filled in *Random*, *X*, *Y* and *Weighted residuals* input fields. Eq 1 and Eq 2 are entered in *f(x)* and *variance function* input fields. A1 is replaced by $a1 + a1|re1:re2$ in the *f(x)* input field. *ID* and *Z* input fields (colored with a red area) are disabled in this section.

The number assigned to each (*re*, *gr*) code depends on the order of the introduction

of qualitative variables in the corresponding input fields. Independent variables (X input field) follow the same rule as (re , gr) codes and the dependent variable (Y input field) is always designated by the lowercase letter y . These codes can be found using the button  and thus help users to write the model form, $g(y)$ and the variance function. Moreover, the *Group* input field can be used to change the color of data points in the graphs (except the Q-Q plot), depending on levels of the considered qualitative variable.

A last check is executed after the display button is clicked. It verifies the concordance between D_3 and informations filled in the variable selection input controls and the calculation of model results.

5 Detailed description of P2

5.1 Graphic tab


A total of 10 graph settings are available from the Graphic tab (Figure 11) and are enabled/disabled depending on the data/graph type used after loading all required data (D_1 , D_2) or creating a subset of D_1 . Some graph settings (label, color, opacity, point type/size) are available by selecting them in the *option* input field and clicking on the edit button. The graph settings are the following:

WebGL	This setting allows to increase the loading speed of the graph in GA and improves the ability to display more elements and their interactivity. Briefly, some non-essential elements are removed from the graph about the temporal/IR data type (duplicated graph inside the range slider). The WebGL setting is then recommended when the D_1 size is huge but it is only available for the 2D plot for now.
Mode	This setting only applies to the temporal/IR data type and is used to connect/disconnect points of each Y variables or NIR/MIR spectra represented in the graph: line+points, only points or lines.
Decimal number	This setting is used to modify the decimal number of (X, Y, Z) coordinates obtained with a mouseover (integer values required from the <i>manual</i> input field) and is disabled for histplot, barplot and corplot.
Bin width	This setting is used to set a custom bin width of histplot. The <i>manual</i> input field considers a value between 0 and the X range (i.e. the difference between max-min values and noted r_X). If a <i>Group</i> variable is added, then r_X is calculated for each group and the maximal value is retained.
Y scale	This setting allows to manage the Y -axis scale for the temporal/IR data type which is disabled (by default) because of the range slider added on the

X -axis (time/frequencies interval). Thus, the Y -axis scale can be manually set with *min/max* input fields (the decimal symbol is a dot) or automatically calculated from the local X -axis interval. The *fraction* input field available from the auto scaling is to add a top/bottom margin corresponding to a ratio of the local Y range. Its value is between 0 and 0.1 (the decimal symbol is a comma).

Label	This setting is used to add a main title and replace the default label of (X , Y , Z) axes by a custom label (Figure 12). The custom label is written from a text input field. The <code>
</code> code can be used in the text input field to break lines.
Color	This setting is used to modify the color of the data represented in the graph (Figure 13, not available for corplot). The custom color is selected from a colour picker widget returning an uppercase hexadecimal value in the associated input field. The color name can be entered in this input field and will be transformed into an uppercase hexadecimal value. The default color is defined by the uppercase hexadecimal value <code>#1C86EE</code> if unique or by a hue palette else.
Opacity	This setting modifies the opacity of the data represented in the graph (Figure 14, not available for corplot). Data added with statistical methods (section 5.3) are not affected by this setting. An input field is used to enter a custom value between 0 and 1 (0.7 as default value).
Point type	This setting is used to change the point type for 2D/3D plot and boxplot (Figure 15). A different number of point types are available from these graph types such as: 24 for 2D plot and boxplot (coded 1 to 24) and 8 for 3D plot (coded 1 to 8). The custom point type is selected from an input field returning the list of codes (1 as default value corresponding to a filled circle).
Point size	This setting changes the point size for 2D/3D plot and boxplot (Figure 16).

An input field is used to enter the custom point size (6 as default). The centroid method available from the Statistics tab (section 5.3) is affected by this setting such as the size of centroids is twice the size of points.

All graph settings are taken into account as soon as the graph is created in GA with the display button (P1). The graph is then automatically updated when new custom values are validated. Custom values entered in each *manual* input field (decimal number, bin width, Y scale) need to be validated by pressing the button  when a graph is already created. Custom values of the last 5 graph settings (label, color, opacity, point type/size) are added to each data inventory (Figure 12, Figure 13, Figure 14, Figure 15, Figure 16) by selecting the row and clicking on the add button. In addition, other buttons available below each data inventory (select all, deselect all, clear, ok) are used to select/deselect all rows and remove/validate the custom value(s) respectively.

A similar approach is used to organize the data inventory of the following graph settings: color, opacity, point type/size. Firstly, default values are shown in each data inventory when a graph is displayed in GA. Secondly, the number of rows depend on the *Group* input field (a single row if no *Group* variable is selected) for the normal/IR data type and on the Y input field for the temporal data type. Thirdly, a missing value in the default value column (only for the normal/IR data type) means that the level of the *Group* variable is missing with the current variable selection (P1). Otherwise, these 4 graph settings do not affect flag data (section 5.2) represented in the graph (GA).

Graphic
Flag
Statistics

WebGL:
☒ Yes ☐ No

Mode:
☐ Line+point ☒ Point
☐ Line

Decimal number:
☒ Auto ☐ Manual: <div> </div> <div> </div>

Bin width:
☒ Auto ☐ Manual: <div> </div>

Y scale:
☒ None
☐ Auto: fraction <div> </div>
☐ Manual: min max <div> </div>

Option:
 <div> </div> <div> Edit </div>

Figure 11. List of graph settings available from the Graphic tab

Edit label

Custom label:

Label	Text (custom)
title	
x	log(D)
y	log(H)

Deselect all
Select all
Clear
Add
Ok
Close

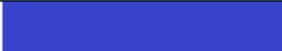



Figure 12. Graph setting used to edit the title and axis labels. The txt file named *data_tree* is used as D_1 (psD_1 , section 6.2). In this figure, a custom label has been assigned to X and Y axes.

Edit color/opacity

☒ color ☐ opacity

Custom color:

#FFFFFF

Level (Group)	Hexadecimal (custom)	Color (custom)	Color (default)
Site 1	#3943CC		
Site 2	#219629		

Deselect all

Select all

Clear

Add

Ok

Close

Figure 13. Graph setting used to edit the color (available by selecting the color option with the radio button). The txt file named *data_tree* is used as D_1 (psD_1 , section 6.2). In this figure, a custom color is assigned to each level of the *Group* variable *Site*.

Edit color/opacity

☐ color ☒ opacity

Custom opacity:

1

Level (Group)	Opacity (custom)	Opacity (default)
Site 1	1	0.7
Site 2	0.5	0.7

Deselect all

Select all

Clear

Add

Ok









Close









Figure 14. Graph setting used to edit the opacity (available by selecting the opacity option with the radio button). The txt file named *data_tree* is used as D_1 (psD_1 , section 6.2). In this figure, a custom opacity is assigned to each level of the *Group* variable *Site*.









Edit point type/size

☒ type ☐ size

Custom type:

1:  2:  3:  4:  5:  6:  7:  8: 

9:  10:  11:  12:  13:  14:  15:  16: 

17:  18:  19:  20:  21:  22:  23:  24: 

Level (Group)	Point type (custom)	Point type (default)
Site 1	11	1
Site 2	17	1

Deselect all Select all Clear Add Ok Close

Figure 15. Graph setting used to edit the point type (available by selecting the type option with the radio button). The txt file named *data.tree* is used as D_1 (psD_1 , section 6.2). In this figure, a custom point type is assigned to each level of the *Group* variable *Site*.

Edit point type/size

☐ type ☒ size

Custom size:

6

Level (Group)	Point size (custom)	Point size (default)
Site 1	9	6
Site 2		6

Deselect all Select all Clear Add Ok Close

Figure 16. Graph setting used to edit the point size (available by selecting the size option with the radio button). The txt file named *data.tree* is used as D_1 (psD_1 , section 6.2). In this figure, a custom point size is assigned to the first level of the *Group* variable *Site*.

5.2 Flag tab

The Flag tab (Figure 17) is available under certain conditions (numeroted 1 to 4 below) as soon as a graph is displayed in GA. A click event listener is also added to the current graph corresponding to the manual selection (left mouse click) of data points considered as outliers. All commands related to Flag tab are then disabled when:

1. The 2D/3D plot type is not selected (normal data type concerned);
2. X , Y or Z is transformed with $f(x)$, $g(y)$ or $h(z)$ respectively (normal/temporal data type concerned);
3. A calibration/validation model is used (normal data type concerned);
4. A method is selected in the Statistics tab (normal/IR data type concerned).

The selected data points are highlighted in black and can be removed (one by one or all) or saved with buttons available from this tab: clear, clear all and save, respectively. Once the save button is clicked, a serie of actions is executed in the following order:

- The selected data points are collected in a new flag data or the existing flag data loaded from the corresponding check box (named *with flags*, P1);
- Flag data are written in a csv/txt file (same format as D_1) at the D_1 location. The filename is the same as D_1 followed by the code “*norm_flag*”, “*temp_flag*” or “*ir_flag*” if the data type is normal, temporal or IR, respectively. A supplementary code “*_withID*” is added at the end of the filename if an ID variable is used to identify the selected data points;
- The flag box (P1) is automatically checked if no flag data was previously loaded;
- The current graph is updated with information based on the previous selected data points. A new color is assigned to these data points depending on the quality code (option described more precisely below).

Input controls in the Flag tab allows to manage the manual selection of data points (rule, type) and provide additional information in flag data as follows:

Quality code	The radio button is used to categorize selected data points according to a quality code (qc) numeroted 1 and 2. Data points saved in flag data with a $qc = 2$ are deleted on the graph (not in the file corresponding to D_1) and colored in red. Data points designated by a $qc = 1$ are only highlighted in orange. A unique $qc (= 2)$ is retained for the normal data type. In addition, these new data points highlighted in orange/red will not be displayed on the graph when one of the three first conditions cited above is effective. However, the deletion of data points saved with a $qc = 2$ will always be taken into account and a warning message will inform about the number of concerned points.
Action	The radio button is used to assign a rule on the click event. The first rule (add new flags) authorizes the selection of points not already saved in flag data. The second rule (replace $qc = 1$ with 2) only accepts the selection of points previously saved in flag data with a $qc = 1$ and changes the qc value after clicking on the save button. This second rule is disabled for the normal data type (only $qc = 2$).
Variable	The check boxes are used to designate which variables (X, Y, Z) are considered as outliers. These check boxes are only available for the normal data type and must be specified before starting the selection of data points.
Draw	The radio button is used to change the type of selection between single point (point) and point-to-point (interval). The point-to-point selection is only available from the temporal data type and requires to click twice on the graph to define the interval boundaries. Multiple intervals will be automatically created if missing values exist in the selected interval. The single point selection is different for the IR data type and allows you to select a whole MIR/NIR spectrum by clicking on one of these data points.

Com- The text input field is used to add a commentary in flag data related to the
mentary selected data points (disabled for the normal data type). No comment will
be added if the corresponding text input field is left blank.

Figure 17. Input controls available from the Flag tab

The flag data is organized differently depending on the used data type (column name precised into quotes below), such as:

- | | |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Normal | Selected points are described by a first column corresponding to the <i>ID</i> variable or the D_1 row number (" <i>.row_num.</i> "). The others columns are (<i>X</i> , <i>Y</i> , <i>Z</i>) variables concerned and forms a binary matrix with 1 used to identify these points. |
| Temporal | Selected points/intervals are introduced row by row and are described by start/end dates (" <i>date_start</i> " and " <i>date_end</i> " respectively), the <i>Y</i> variable concerned (" <i>var_name</i> "), a <i>qc</i> and a comment (NA characters if not precised). |
| IR | Informations on selected MIR/NIR spectra are also entered row by row and are given by the same first column as the normal data type, a <i>qc</i> and a comment. |

5.3 Statistics tab

The Statistics tab offers several statistical calculation methods for the normal/IR data type (Figure 18). This tab is disabled when data points (related to the Flag tab) are currently selected on the graph. Data points saved in flag data with a $qc = 2$ are obviously not used in the different statistical calculations performed from this tab. Adding a *Group* variable (P1) allows to apply separately these methods on each of its levels. Moreover, some methods are available depending on the selected graph type (2D/3D plot, boxplot, histplot) or model strategy after loading all required data (D_1 , D_2 , D_3) and are described as follows:

Normal 2D/3D plot

The methods that can be applied to a 2D plot are the linear regression method (*lm* function), the 95 % confidence ellipse (*dataEllipse* function used under the package *car*) and the centroid. These previous methods are all disabled when a calibration model is selected in P1. As the two first methods require a two-dimensional plan, they are also disabled for a 3D plot. Otherwise, supplementary information is displayed with a mouseover on the regression line, such as the equation, the intercept/slope value, R^2 and $RMSE$. The regression line information are partly modified if a validation model is used. The $RMSE$ value is then replaced with the p-value of two equality tests based on a Student's t-distribution, such as hypotheses are: intercept = 0 and slope = 1.

Boxplot

A unique method is available and allows to add the mean/standard deviation (*sd*) on each box as new information.

Histplot

The methods available are the density curve of the X variable distribution (*density* function with a Gaussian kernel smoothing) and the density curve

following a normal distribution with the mean/*sd* of the *X* variable (*dnorm* function). Two vertical lines are added to the density curve (first method) corresponding to the mean/median (dashed and plain lines respectively). Additional information are displayed with a mouseover on this density curve, such as a set of values used to describe the *X* variable distribution (size, mean, median, *sd*) and the p-value of two normality tests: Shapiro-Wilk (*shapiro.test* function) and Kolmogorov-Smirnov (*ks.test* function). The p-value of the Shapiro-Wilk test may be missing because of a restriction on the maximum sample size (= 5000). Other p-values are added to information if a *Group* variable is used and concern variance equality tests (between samples defined for each level): Bartlett (*bartlett.test* function) and Levene (*leveneTest* function with mean/median center).

IR A unique method is available and allows to calculate the mean spectrum.

The screenshot shows a software interface with three tabs: 'Graphic', 'Flag', and 'Statistics'. The 'Statistics' tab is active. It contains several sections of options:

- Normal:**
 - Plot:**
 - ☐ Add linear regression
 - ☐ Add confidence ellipsoid
 - ☐ Add centroid
 - Histplot:**
 - ☐ Add density curve
 - ☐ Add normal density curve
 - Boxplot:**
 - ☐ Add mean/sd
- IR:**
 - ☐ Add mean spectrum

Figure 18. List of statistical calculation methods available from the Statistics tab

6 Detailed description of GA

6.1 Action buttons

A list of buttons is available above the graph depending on the data/graph type and the Statistic tab option used. Their functions can be arranged in three main categories: saving graphs as a picture file (2 buttons), data visualization management (14 buttons) and adding new elements on the graph (1 button). Buttons are described in the order of main categories as follows:



The button open a new window (Figure 19) corresponding to the management of picture parameters, such as the filename, the height/width (in pixel) and the format (png, jpeg and svg). The ok button available from this window is used to validate new parameters. All parameters are reseted if a new graph is created on GA with the display button (P1).



The button allows to save the picture file at the location specified by the web browser used.



The button concerns the zoom mode (only available from the normal data type: 2D/3D plot and histplot). Areas can be manually selected on the graph when the mode is activated.



The button is a mode used to move horizontally/vertically the plane (same availability as the zoom mode).



The two buttons allow to zoom in/out on the graph (only available from the normal data type: 2D plot and histplot).



The two buttons are used to select different rotation modes for the 3D plot.



The button is a mode used to enable/disable the mouseover event (only available from a 3D plot).



The button is used to return the last saved camera position from the 3D plot. A camera position is saved after each update of the graph.



The two buttons return the same result and allow to reset axes corresponding to the whole graph (other button used for the temporal data type and cited below).

1 month

3 months

6 months

all

The four buttons only concern the temporal data type. A time interval (X -axis) can be manually/automatically selected on the graph. Different time periods are considered for the second selection: 1 month, 3 months, 6 months or all (reset X -axis).



The button only appears when the linear regression option is selected in the Statistics tab and allows to access to a new window (Figure 20). The regression line information can then be filled in an input field from the previous windows and added to a custom position on the graph (validation with the ok button). The reset button is used to remove information added on the current graph.

Picture details ×

Enter name:

Picture_name

Height: 800

Width: 1000

Format: png

Ok

Figure 19. Window used to change picture parameters (default values above)

Linear regression information ×

X position: ☒ Left ☐ Right

Y position: ☒ Top ☐ Bottom

Information:

Reset Ok

Figure 20. Window used to add linear regression information on the graph

6.2 Graph examples

A list of pseudo-data have been specifically created to help users to test the different functionalities cited in sections above. These datasets (detailed in Table 4) are available from <https://github.com/PhilippeSantenoise/WIDEa> (see Data folder) and are

illustrated from several graph examples below.

Data type	Code (loading field)	Github data name
Normal	$psD_1 (D_1)$, $parD (D_3)$	<i>data_tree</i> , <i>data_param</i>
<u>Details:</u> Three variables have been randomly generated following known equations (Gompertz, Power) for two (broadleaved) tree species based on two sites: total height (H in m), diameter at breast height (D in cm) and stem biomass (kg). The data size is 400 trees, i.e. 100 trees for each combination of species and sites. Data have been used to calibrate the H-D Gompertz model (Eq 1 and Eq 2) with $parD$ as the inventory of estimated parameters.		
Normal	$psD_2 (D_1)$	<i>data_water_table_chemistry</i>
<u>Details:</u> The content of 11 mineral elements (in ppm) of a water table have been randomly generated following a linear relation between several pairs of elements: F, Cl, S, P, Fe, Mn, Mg, Al, Ca, Na and K. The data size is 300 samples.		
Temporal	$psD_3 (D_1)$	<i>data_weather</i>
<u>Details:</u> Data are the random simulation of the daily temperature (degree Celsius) and the daily humidity (%) in 2020 at Nancy (France). The simulation have been realized from some measured daily values to retain global trends for each month.		
IR	$psD_4 (D_1)$, $cfD (D_2)$	<i>data_MIR_spectra</i> , <i>data_MIR_code_freq</i>
<u>Details:</u> Data are randomly simulated MIR spectra (size = 100) and informations on MIR code/frequency (cm^{-1}) respectively. A set of spectra obtained with the Diffuse Reflectance Infrared Fourier Transform Spectroscopy (DRIFTS) technique on tree branch bark powder samples have been used for the random simulation. The considered MIR interval is 4000 to 550 cm^{-1} with a 4 cm^{-1} resolution (i.e. 1790 frequencies).		

Table 4. Presentation of all data available from <https://github.com/PhilippeSantenoise/WIDEa/tree/main/Data> and description of generated pseudo-data (psD_1 , psD_2 , psD_3 , psD_4 , psD_5). Pseudo-data correspond to a specific type mentioned in the section 4.1: normal (psD_1 , psD_2), temporal (psD_3), IR (psD_4). The loading field (D_1 , D_2 and D_3) is the location where data must be loaded in P1 (see sections 4.2 and 4.6).

Example 1 (psD_1):

psD_1 can be used to create a list of graphs after selecting the normal data type, loading data (D_1 text field: Figure 4) and providing the requested variables (X , Y , Z , $Group$). Figure 21 presents an example of 4 graphs displayed in GA. The procedure used to generate these graphs is detailed in Table 5.

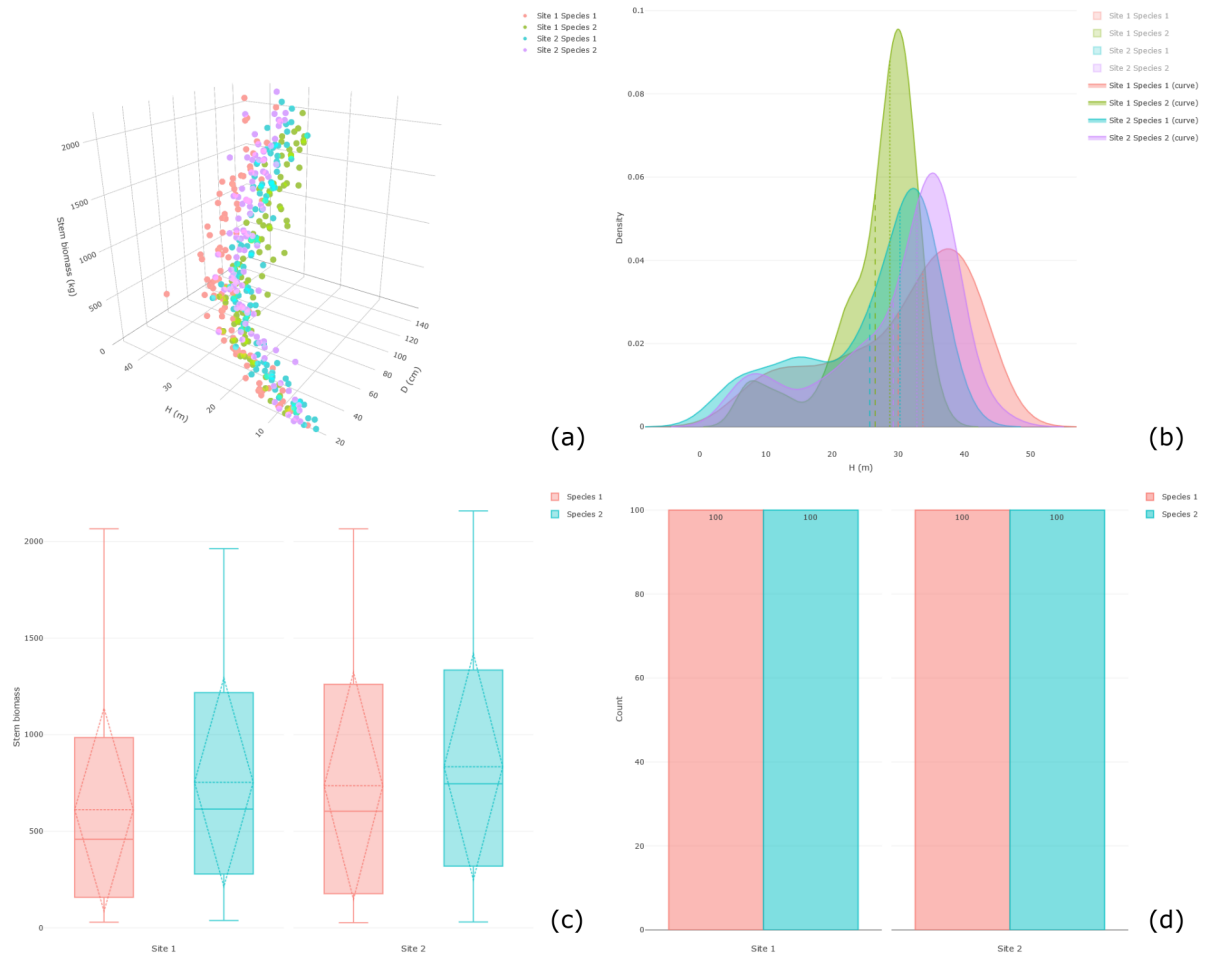


Figure 21. Graphs created from psD_1 : (a) 3D plot with (D, H, Stem biomass) as (X , Y , Z) variables, (b) histplot (deselected) with H as X variable and the associated density curve (Statistics tab), (c) boxplot with (Site, Stem biomass) as (X , Y) variables and informations on mean/sd values (Statistics tab) represented by a dot line and diamond top/bottom vertices respectively and (d) barplot with Species as X variable. A $Group$ variable is used for each graph: Site \times Species (concatenation) for (a) and (b) and only Site for (c) and (d).

Step	Panel (Section/Tab)
1	P1 (Data type selection, Data loading)
	(all): Select “Normal” as data type and load psD_1 from D_1 text field.
2	P1 (Normal plot selection)
	<p>Select the graph type (Model = “None”),</p> <p>(a): “Plot”; (b): “Histplot”;</p> <p>(c): “Boxplot”; (d): “Barplot”.</p> <p>Select “3” as the dimension number for the graph noted (a).</p>
3	P1 (Variable selection)
	<p>Select variables using the corresponding input fields,</p> <p>(a): “D” (X), “H” (Y), “Stem_biomass” (Z); (b): “H” (X);</p> <p>(c): “Site” (X), “Stem_biomass” (Y); (d): “Site” (X).</p> <p>Add a <i>Group</i> variable by enabling the input field with the Yes/No radio button.</p> <p>Select “Site” and “Species” for graphs noted (a) and (b) after checking the concatenation box. Select only “Site” for graphs noted (c) and (d).</p>
4	P2 (Graphic)
	<p>Open the graph setting used to edit labels by selecting “label” in the <i>option</i> input field and clicking on the edit button. Write each axis label in the <i>custom label</i> input field,</p> <p>(a): “D (cm)” (X), “H (m)” (Y), “Stem biomass (kg)” (Z);</p> <p>(b): “H (m)” (X).</p> <p>Select the row of the data inventory (label column: x, y or z) and click on the add button. Confirm the label of all axes by pressing the ok button.</p>
5	P2 (Statistics)
	<p>Add supplementary informations by checking the box named,</p> <p>(b): “Add density curve”; (c): “Add mean/sd”.</p>
6	P1
	(all): Create the graph with the display button.

Table 5. Detailed procedure to create graphs noted (a), (b), (c) and (d) in Figure 21. The legend of the graph (b) is partly deselected by clicking on it.

Example 2 (psD_1):

In this example, a 2D plot with custom values for several graph settings (label, color, opacity and point type/size) is created from a subset of psD_1 (Figure 22). Steps are described as follows:

- (Step 1) Select “Normal” as data type and load psD_1 from D_1 text field (P1: Data type selection, Data loading);
- (Step 2) Enable input controls used to create a subset of psD_1 by checking the *add conditions* box (P1: Sub-data creation);
- (Step 3) Add conditions into a data inventory by filling in information about the variable name/type, the relation symbol and the value and clicking on the add button. A total of 5 conditions (coded $c1$ to $c5$) are added individually and detailed in Figure 6 (P1: Sub-data creation);
- (Step 4) Enter the following formula used to combine all conditions in the text field below the add button: $c1 \& ((c2 \& c4) | (c3 \& c5))$. The formula consist of retaining only “Species 1” (for all sites) and trees which D is strictly greater than 40 cm for “Site 1” or trees which H is less than (not strictly) 35 m for “Site 2”. The subset of psD_1 is then created by clicking on the create button (P1: Sub-data creation);
- (Step 5) Keep default values (“Plot” as type, “2” as dimension number and “None” as model) for the graph type (P1: Normal plot selection);
- (Step 6) Select “D” (coded $x1$) and “H” (coded $y1$) from the X and Y input fields respectively. Enable the text field (Yes/No radio button) corresponding to $f(x)$ and $g(y)$ functions and use the logarithmic function (\log) to transform X and Y variables respectively: $\log(x1)$ and $\log(y1)$. Enable the *Group* input field (Yes/No radio button) and select “Site” (P1: Variable selection);
- (Step 7) Build the graph with the display button (P1).

- (Step 8) Select “label” from the *option* input field to edit axis labels. Custom labels assigned to X -axis and Y -axis are detailed in Figure 12 (P2: Graphic);
- (Step 9) Select “color/opacity” from the *option* input field to edit the color/opacity of the *Group* variable levels. Custom colors/opacities assigned to the *Group* variable levels are detailed in Figure 13 and Figure 14 (P2: Graphic);
- (Step 10) Select “point type/size” from the *option* input field to edit the point type/size of the *Group* variable levels. Custom point types/size assigned to the *Group* variable levels (one level for the custom point size) are detailed in Figure 15 and Figure 16 (P2: Graphic);
- (Step 11) Add the linear regression method by checking the corresponding box (P2: Statistics);
- (Step 12) Deselect the “Site 2” regression line in the graph legend (GA);

In steps 8 to 10, some actions (add custom values to a data inventory and confirm them) are required to apply all graph settings to the current graph and are described in the section 5.1.

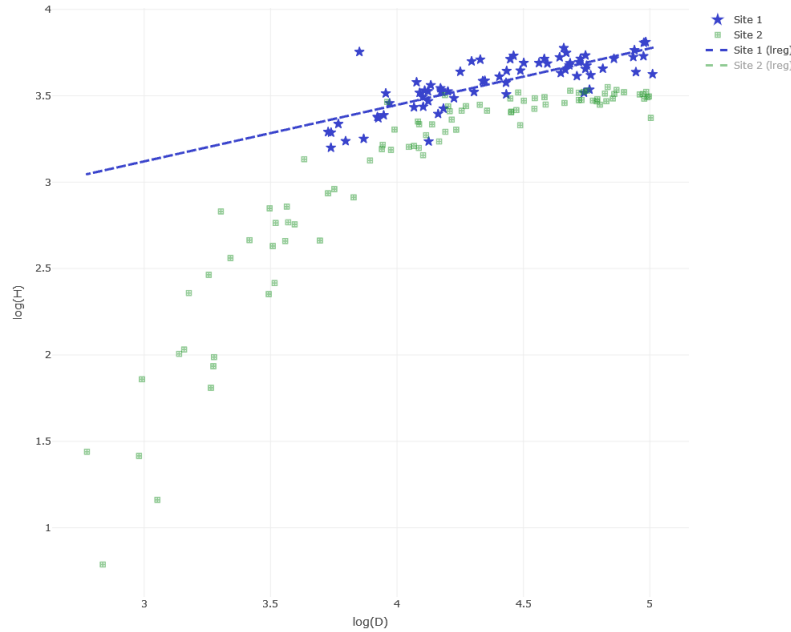


Figure 22. 2D plot from a subset of psD_1 : study of the linear relation between $\log(H)$ and $\log(D)$.

Example 3 (psD_1 , $parD$):

The H-D Gompertz model results (Figure 23) can also be displayed in GA, by using psD_1 as D_1 file and $parD$ as D_3 file. The procedure is fully described in Figure 9 and Figure 10.

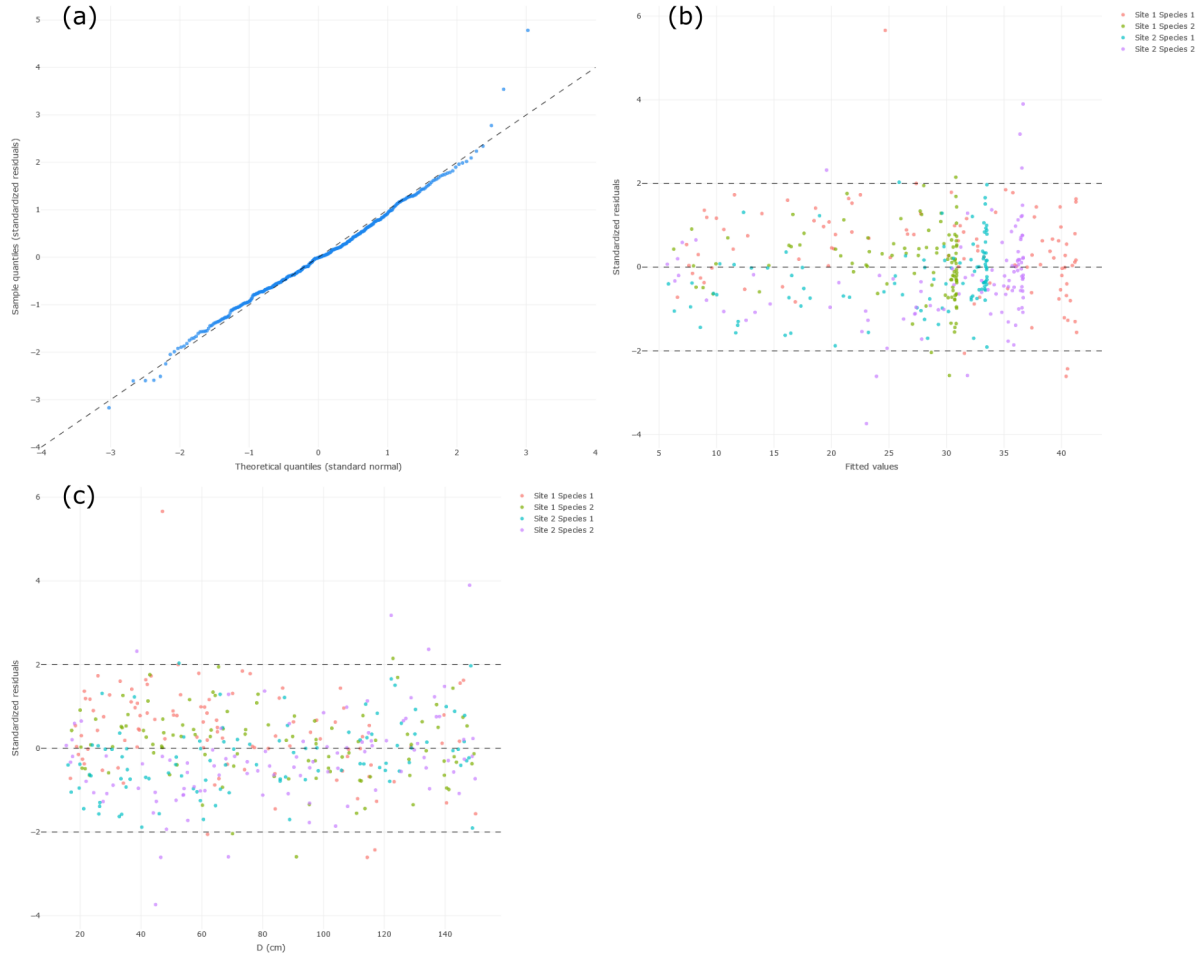


Figure 23. Graphs used to study assumptions on the H-D Gompertz model (psD_1 and $parD$) residuals: (a) qqplot on the standardized residuals distribution, (b) 2D plot on standardized residuals vs fitted values and (c) 2D plot on standardized residuals vs D. A *Group* variable is also added to color data per Site \times Species (concatenation) for (b) and (c).

Example 4 (psD_2):

A correlation matrix (Figure 24) can be built from psD_2 , by following the procedure below:

- (Step 1) Select “Normal” as data type and load psD_2 from D_1 text field (P1: Data type selection, Data loading);
- (Step 2) Select “Corplot” as the graph type (P1: Normal plot selection);
- (Step 3) Select all variables from the Y input field (P1: Variable selection);
- (Step 4) Create the correlation matrix with the display button (P1).

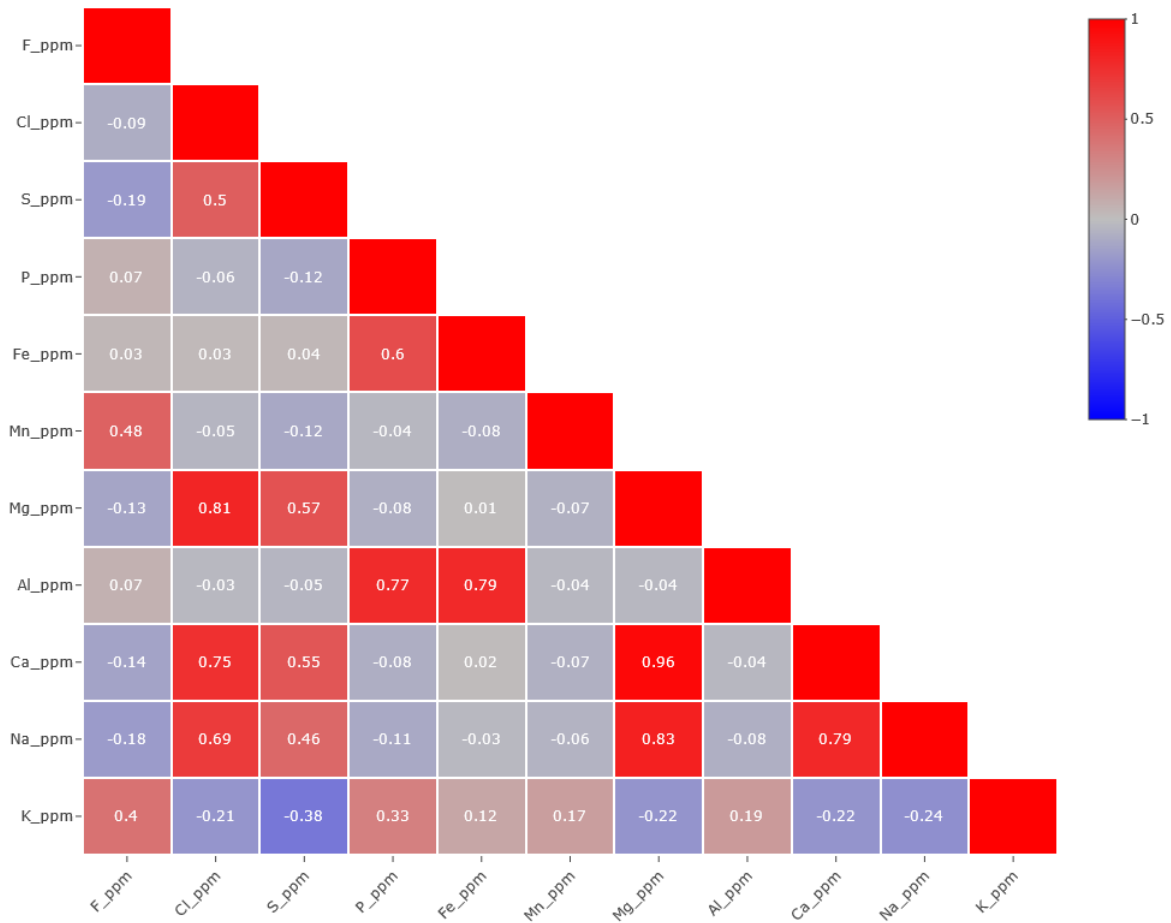


Figure 24. Corplot created from psD_2 : study of linear correlations of the content of 11 mineral elements of a water table.

Example 5 (psD_3):

psD_3 are used as an example of temporal data. The Daily temperature/humidity in 2020 can then be displayed in GA (Figure 25) by following steps below:

- (Step 1) Select “Temporal” as data type and load psD_3 from D_1 text field (P1: Data type selection, Data loading);
- (Step 2) Select “Date”, “%Y%m%d” and the pair (“Temperature”, “Humidity”) from X , $format$ and Y input fields respectively (P1: Variable selection);
- (Step 3) Click on “No” and “Line+point” from WebGL and Mode radio buttons respectively (P2: Graphic);
- (Step 4) Create the graph with the display button (P1).

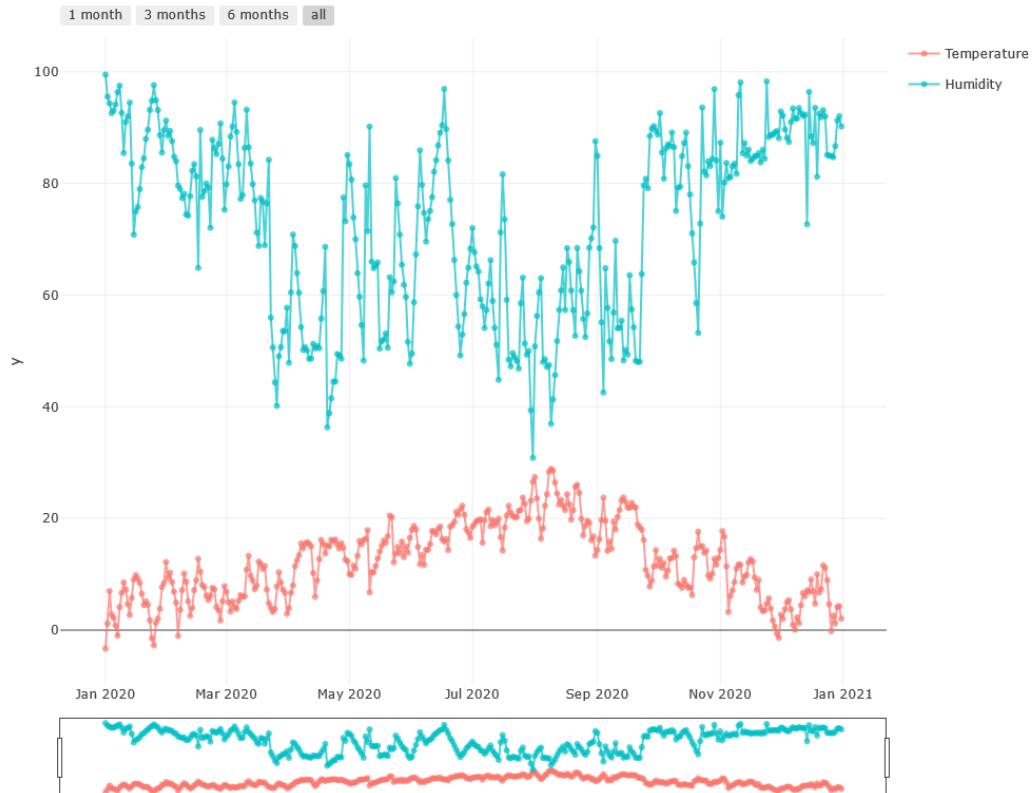


Figure 25. 2D plot created from spD_3 : the daily temperature and the daily humidity (Y variables) in 2020 (X variable: Date with the %Y%m%d format) at Nancy.

Example 6 (psD_4):

psD_4 and cfD are used as an example of IR data. MIR spectra of absorption can be directly displayed in GA (Figure 26) by following steps below:

- (Step 1) Select “IR” as data type and load psD_4 and cfD from D_1 and D_2 text fields respectively (P1: Data type selection, Data loading);
- (Step 2) Click on “Yes” and “Line” from WebGL and Mode radio buttons respectively (P2: Graphic);
- (Step 3) Create the graph with the display button (P1).

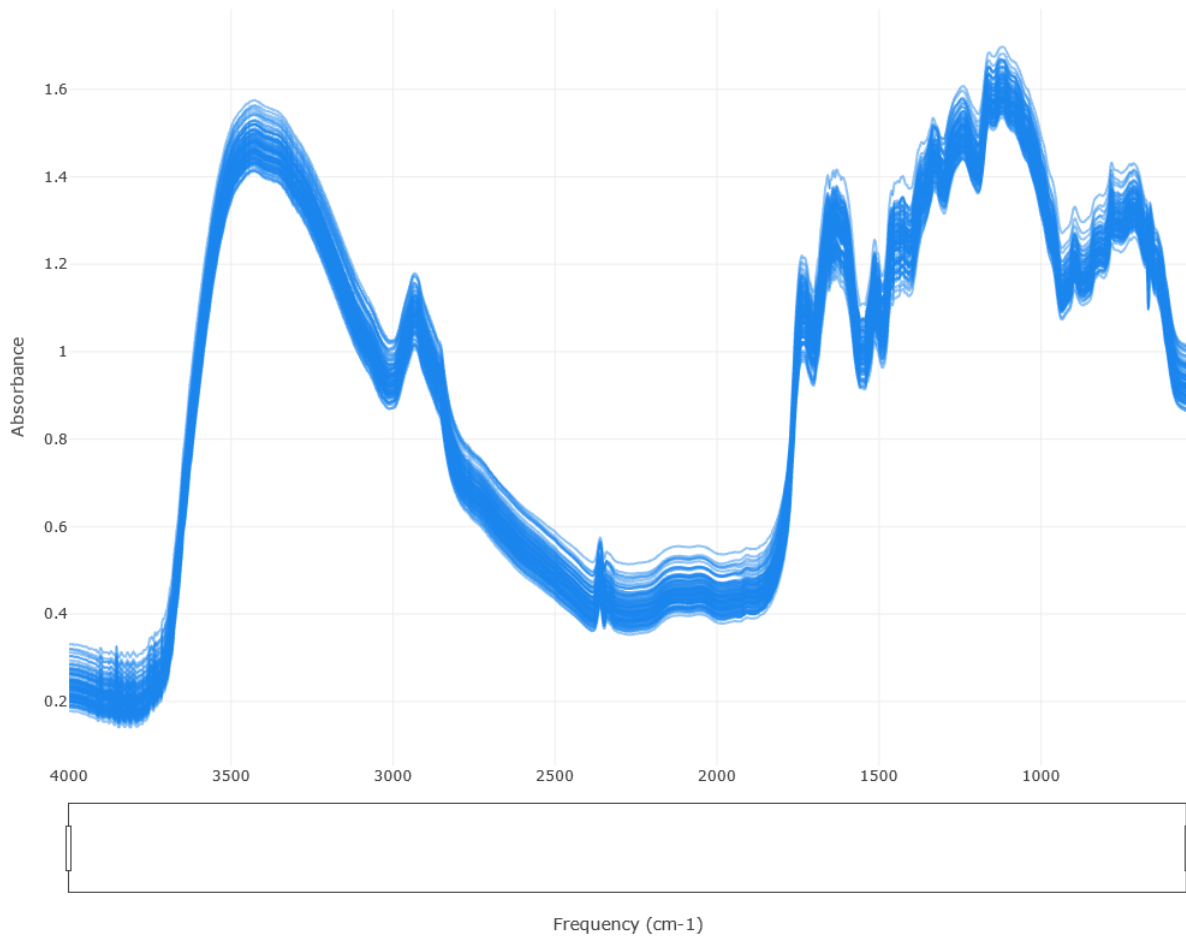


Figure 26. 2D plot created from psD_4 and cfD : MIR spectra of absorption on the considered frequency interval.

References

- [1] Attali Dean (2020), *shinyjs: Easily Improve the User Experience of Your Shiny Apps in Seconds*, shinyjs.
- [2] Attali Dean (2022), *colourpicker: A Colour Picker Tool for Shiny and for Selecting Colours in Plots*, colourpicker.
- [3] Bailey Eric (2015), *shinyBS: Twitter Bootstrap Components for Shiny*, shinyBS.
- [4] Chang Winston et al. (2021), *shiny: Web Application Framework for R*, shiny.
- [5] Chang Winston (2021), *shinythemes: Themes for Shiny*, shinythemes.
- [6] Cheng Joe et al. (2021), *htmltools: Tools for HTML*, htmltools.
- [7] Dowle Matt et al. (2021), *data.table: Extension of 'data.frame'*, data.table.
- [8] Fox John et al. (2019), *An {R} Companion to Applied Regression*, Sage, Thousand Oaks {CA}, <https://socialsciences.mcmaster.ca/jfox/Books/Companion/>.
- [9] Lai Randy (2020), *arrangements: Fast Generators and Iterators for Permutations, Combinations, Integer Partitions and Compositions*, arrangements.
- [10] Langford Eric (2006), *Quartiles in Elementary Statistics*, 14:3, DOI: 10.1080/10691898.2006.11910589.
- [11] Lin Pedersen Thomas et al. (2020), *shinyFiles: A Server-Side File System Viewer for Shiny*, shinyFiles.
- [12] Meyer Fanny et al. (2020), *shinybusy: Busy Indicator for 'Shiny' Applications*, shiny-busy.
- [13] Müller (2018), *bindrcpp: An 'Rcpp' Interface to Active Bindings*, bindrcpp.
- [14] Ooms Jeroen (2020), *V8: Embedded JavaScript and WebAssembly Engine for R*, V8.

- [15] Sievert Carson (2020), *Interactive Web-Based Data Visualization with R, plotly, and shiny*, Chapman and Hall/CRC (ISBN: 9781138331457), <https://plotly-r.com>.
- [16] Vaidyanathan Ramnath et al. (2020), *htmlwidgets: HTML Widgets for R*, htmlwidgets.
- [17] Wickham Hadley et al. (2020), *scales: Scale Functions for Visualization*, New York, NY: Springer.
- [18] Xie Yihui et al. (2022), *DT: A Wrapper of the JavaScript Library 'DataTables'*, DT.
- [19] Zuur, Alain F. (2009), *Mixed Effects Models and Extensions in Ecology With R*,