

An Introduction to XLSTAT-R

I. Legal information

The R logos used in the XLSTAT interface and documentation are transformations of the original R logo. The transformed versions are available here <https://help.xlstat.com/customer/portal/attachments/833263> and here <https://help.xlstat.com/customer/portal/attachments/833377>. Addinsoft respects the specifications of the Creative Commons (<https://creativecommons.org>), as requested by the Attribution-ShareAlike 4.0 International (CC BY-SA 4.0).

XLSTAT connects to R using R scripts and does not alter the functioning of R in any way. XLSTAT original code is used to run R, and to exchange data with R. XLSTAT, the most user friendly commercial software, uses its own routines to reformat and display the R results in an easy to handle and customizable way. The user can take advantage of the many Excel features to further customize results tables or charts.

XLSTAT-R is a free feature of XLSTAT, as long as you are using the trial version or if you are a licensed user of a paid solution (Base or superior). XLSTAT-R is not included in XLSTAT Free.

Addinsoft provides a series of connectors to R functions free of charge. Support is only provided free of charge for XML connectors that have been developed by Addinsoft. Support for third party connectors is provided against a charge that will depend on the complexity of the request. For any special request, please go to <https://help.xlstat.com/>.

Addinsoft declines any responsibility for any type of problem that would be related to the use of an R function or of an add-on package for R. R functions are developed by the R core team or independent developers and Addinsoft is not responsible for any flaw contained in the R code.

II. XLSTAT-R, R and editing XML files

XLSTAT-R uses the R version you have installed on your computer. We recommend you install R version 3.4.1 or later. XLSTAT does not include an R installation package and as a consequence, the end-user must install R. We recommend that you download the R software at:

Windows:

<https://cran.r-project.org/bin/windows/base/>

Mac:

<https://cran.r-project.org/bin/macosx/>

To edit the XML files, we recommend you use *Sublime* (Windows and Mac) or *Notepad++* (Windows only), or any editor that takes into account the file format and can make the difference between code parts and user inputs.

Sublime can be downloaded at:

<http://www.sublimetext.com>

Notepad++ can be downloaded at:

<https://notepad-plus-plus.org/download/>

Editing XML files might look complex at the beginning but it is very simple once you know how to navigate in the file. In the last part of the document, you can find a documentation of the code elements that you can use within XLSTAT-R.

III. Getting started

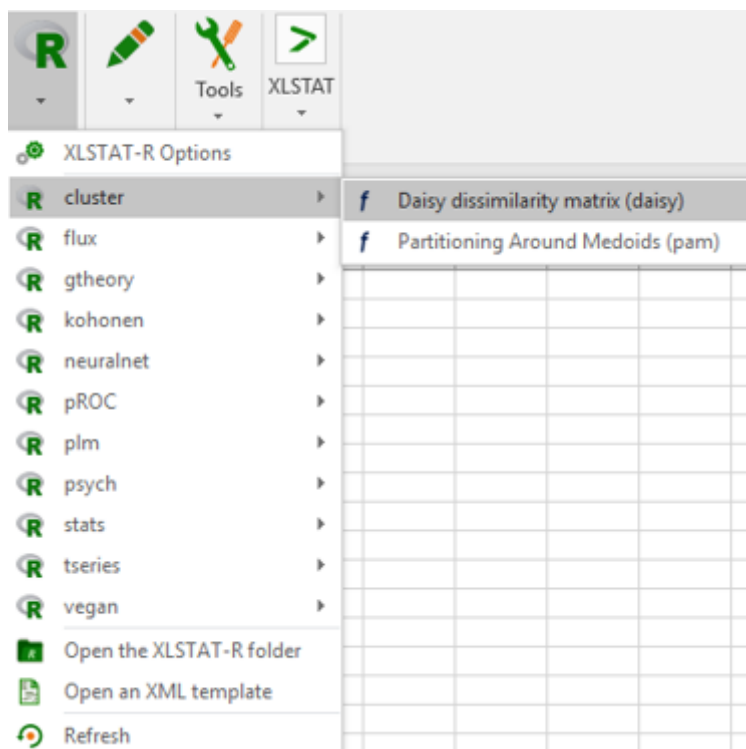
In this chapter you will find information on how to create your own connector between XLSTAT and R and to customize results.

1. Activating XLSTAT-R

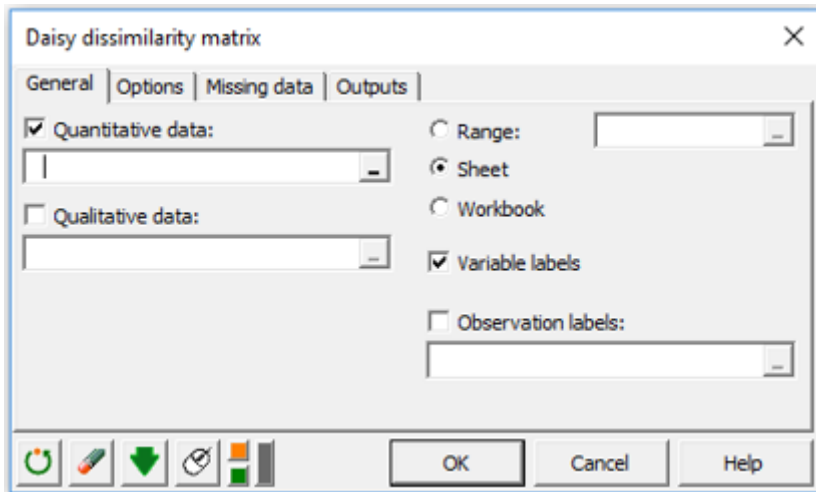
If you are a licensed XLSTAT user, to activate XLSTAT-R all you need to do is install XLSTAT 2017.6 or later. If you are not a license user, you should install the XLSTAT trial version, or, if you have already used a trial in the past, you need to order a license.

2. Using an existing R function with XLSTAT-R in Excel

Once R is installed, go to **XLSTAT > Advanced features > XLSTAT-R** and click on one of the existing functions. See an example below.



The corresponding feature dialog box opens:



Then, you simply need to select the data and configure the dialog box like in standard XLSTAT function. Click **OK** to launch the computations. If you have never set up analysis in XLSTAT before, have a look at an example [here](#).

3. Creating your own R features using XLSTAT-R in Excel

You can also add your own features in XLSTAT-R. Check our tutorial on how to deploy a simple R function in Excel using XLSTAT-R.

As part of this procedure, you will need to edit XML files. We recommend that you use *Sublime* (Windows and Mac) or *Notepad++* (Windows only), or any editor that takes into account the file format and can make the difference between code parts and user inputs.

Sublime can be downloaded at <http://www.sublimetext.com>

Notepad++ can be downloaded at <https://notepad-plus-plus.org/download/>

You can find a documentation of the code elements that you can use within XLSTAT-R.

IV. The XLSTAT-R `<xml>` syntax

XLSTAT-R is the first and currently unique product that allows to produce user friendly interfaces for the R software that can be used within Microsoft Excel, and that take advantage of the high quality XLSTAT statistical features and outputs.

This document explains the concepts that are necessary to develop an XML connector that can be used by XLSTAT-R to produce a user-friendly interface to run an R function without leaving Microsoft Excel. The concepts are simple enough so that anyone, even with no particular IT skills, who needs to run an R function, can develop his or her own connector.

1. Introduction: A brief description of the XML standard and a few rules to know before developing your first XLSTAT-R function are reported here.

The Extensible Markup Language (XML) is a language that allows to describe structures and that sets a standard for their description.

Structures are composed of nodes, in which other nodes can be nested. For example, the *tree* structure has a *trunk* as a first level **node** (named the **root node**), then *branches* as second level nodes (or subnodes), and *leaves* as third level nodes. The different types of nodes have different set of **attributes** that can take different types of values. For example, the trunk could have as attribute the color of the bark, it's height. Branches could be described by their main direction, whether they've been cut or not. The leaves could be described by their color and length and width.

A few rules must be respected. A node that has subnodes starts with a `<Node arguments>` instruction, and ends with `</Node>` to set the end of the node section. In between, lines describing subnodes can be inserted (for example, `<Trunk barkcolor="white" ><Branch id="1" direction="north"></Trunk>` would be a description example of a tree with one branch. A node that does not have subnodes must be contained into brackets (for example, `<Leaf size="10" color="green" />`), with before the right-side bracket, a `/` to confirm this is the end of the content for that particular node. You notice there is no `/` in that case. This is because it will be added after the description of the subnodes (here the branches), using the following convention: `</Trunk>`.

We recommend that you indent the code to make it easy to read, and each indentation level must correspond to a type of node. The values of the attributes must be inserted into quotation marks or apostrophes. If you would like to add comments in your XML code, you must insert it between `<!--` and `-->`. Special characters like `<` `>` and `&` must not be used, except in the cases described above for the brackets and for special characters for `&`. If you need to use `<` or `>` in the text of the XLSTAT-R dialog boxes, you can respectively use `<` and `>` in your text which are common in html or xml languages. Attributes do not need to respect a special order, but we recommend you always use the same order for easier management and better readability.

The essential operator of R, `<-` that allows to set values or load objects cannot be used in XML. You can replace it by `=` and XLSTAT automatically replaces it by `<-` when sending instructions to R.

2. The Method structure

`<Method text="..." datastructure="..." function="..." group="..." packages="..." uniqueid="..." family="..." question="..." synonyms="..." >`. Method is the base or root structure for XLSTAT-R.

There will be one xml file and one Method for each function you want to add to XLSTAT-R. The attributes of the Method node are described below:

text: *Required*. The text is the plain name of the method.

datastructure: *Required*. This attribute can take the following values:

model tells XLSTAT, that an equation must be created for R to describe the relationship between dependent and explanatory variables. This will require that you know which field corresponds to the Ys (Example: see the `plm(plm)` case).

data tells XLSTAT, that the data sets that are listed are taken individually (Example: see the `som(kohonen)` case).

function: *Required*. The name of the R function or of your question (for example, `t.test` for the Student t test).

group: *Required*. The name of the group into which you want to put the function. Typically, the R package name. If you are developing your own functions, you can create your own group.

packages: *Required*. The name of the packages that must be loaded so that the function runs. There should only be more than one package in the list if your script calls functions from several non-default packages. Packages must be separated by a comma (for example `packages="pack1,pack2"`).

uniqueid: *Required*. A unique function number that allows XLSTAT to save the selected settings for a particular function when it is run, and to reload them at the next run.

family: *Optional*. The family of methods (for example, OLS for linear regression). If the method belongs to several families of methods, you can list them using the comma to separate them (for example two layers Kohonen maps belong to neural networks, classification and predictive methods). Default value: "".

question: *Optional*. The typical question this method helps answering. For example, “are two proportions equal”. Default value: "".

synonyms: *Optional*. Other possible names of the method (for example, OLS for linear regression). If several synonyms are available, you can list them using the comma to separate them. Default value: "".

Example:

```
<Method text="Daisy dissimilarity matrix" datastructure="data"
function="daisy" group="cluster" packages="cluster,qgraph"
family="Unsupervised Machine Learning" question="How can I
create a dissimilarity matrix from data of mixed types?"
synonyms="Dissimilarity Matrices for Mixed Data" >
```

3. Sub nodes of the Method structure

Methods contain several general nodes (AuthorXML, AuthorRFunction, CompatibilityXLSTAT, CompatibilityR, DateCreated, DateModified) and the Tab nodes that allow to describe all the content necessary to generate the dialog boxes.

<AuthorXML> corresponds to the author(s) of the XML connector.

<AuthorRFunction> corresponds to the author(s) of the R function.

<CompatibilityXLSTAT> defines the minimum version of XLSTAT required for compatibility.

<CompatibilityR> defines the minimum version of R required for compatibility.

<DateCreated> specifies the date when the connector was created.

<DateModified> specifies the date when the connector was last modified.

<PackageVersion> specifies the version of the package with which the connector was designed.

Example:

```
<AuthorXML>Addinsoft</AuthorXML>
<AuthorRFunction>John Smith</AuthorRFunction>
<CompatibilityXLSTAT>2017.4</CompatibilityXLSTAT>
<CompatibilityR>3.1.1</CompatibilityR>
<DateCreated>2017.08.01</DateCreated>
<DateModified>2017.08.05</DateModified>
<DateModified>mypackage_0.12-34</DateModified>
```

4. Tabs

The XLSTAT R for Excel desktop versions display traditional XLSTAT dialog boxes that are split into different tabs that allow structuring the inputs provided by the user. Tabs should be used to organize the inputs into homogeneous groups. You can use as many tabs as you want, though we recommend our usual General/Options/Missing data/Outputs/Charts structure.

<Tab text="..." enabled="..." leftcolumnwidth="..." >: A tab opens a section where all the controls to be used in a tab must be listed. At the end of the section, you must add **</ Tab>**.

text: Required. The text that is displayed in the header of the tab. For the usual values, General/Options/Missing data/Outputs/Charts, you can add a # before the text, if you want that it is automatically translated in the different languages included in XLSTAT.

enabled: Optional. Defines if the control can be clicked or not (if it is disabled – enabled="false", it is greyed out and not usable). If it is disabled, a change of the value of another control can make it enabled. This can be controlled by code related attributes (onchange, onclick). Default value: "true".

leftcolumnwidth: Optional. Defines the width of the left part of the tab. It is by default 155, but for the *Missing data* tab, we recommend setting it to 220. Default value: "155".

rightcolumnwidth: Optional. Defines the width of the right part of the tab. Default value: "155".

Example:

```
<Tab text="Additional options" enabled="false" />
```

This creates a tab that will only become clickable if a change of a control in another tab changes its **enabled** property to "true".

5. Controls to add in Tabs

XLSTAT-R helps adding controls to the tabs - or tabs - of the dialog box. Column, NextColumn and Spacer allow to organize the content, while the other elements are directly controlling the content that is being displayed.

<Column>: Starts a new column in the dialog box. In Excel desktop version, there can only be two or less columns in a page. The content of a column can be preceded by this node, and must end with </Column>. Alternatively, <NextColumn /> can be used after the end of the content of the first column if there is a second column in the tab, otherwise it is useless.

<NextColumn />: When this node is detected (it is not displayed), all the next controls will be in the right part of the current tab.

<CheckBox clickcode="..." tooltip="..." default="..." enabled="..." left="..." nextrow="..." refeditlinked="..." reportgroup="..." text="..." visible="..." name="..." >:

A CheckBox is a control that you can check or uncheck. Contrary to an RadioButton, it is not dependent on the value of other controls, unless you add code to create the dependence.

chartname: Optional. If the CheckBox is used to offer the user the possibility to display or not a chart, you need to specify the chartname. In that case, the same chartname must be used in the <RResults...> node to detail if the chart is generated by XLSTAT or R. Default value: "".

onclick: The VBA code that is executed when the user checks or unchecks the CheckBox.

tooltip: Optional. The text that is used when you hover the mouse over the CheckBox. If it is not entered the value of the text attribute is used. Default value: "".

default: Optional. The default value of the CheckBox. Default value: "false".

enabled: Optional. Defines if the control can be used or not (it is then greyed out and not usable). If it is disabled, a change of the value of another control can make it enabled. This can be controlled by code related options (onchange, onclick) of other controls. Default value: "true".

left: Optional. Sets the number of pixels by which the control should be moved on the left. If you use a + before the value, the offset on the left will be by 12 times the number after the +. For example, "+2" creates an indentation of 24 pixels. Default value: "0".

name: Optional. The name of the control used by Excel internally. If not provided, XLSTAT-R sets an arbitrary name. Default value: "".

nextrow: Optional. Tells if the next control should be on a new row or not. Default value: "true".

output: Optional. Tells if the checkbox corresponds to an output that the user can choose to display or not. Set it to true only for options that correspond to outputs. Default value: "false".

refeditlinked: Optional. The refedit that depends on the CheckBox. Code is automatically generated so that activating the CheckBox activates the RefEdit. Default value: "".

reportgroup: Optional. The group of options to which this option belongs. Groups are displayed on the same line in the header of the XLSTAT report. If no value or "0" is provided, the value will not be listed in the header of the report. If "0" or another integer is provided, then the rname will be used to replace the codes in the <RScript> nodes. Default value: "".

rname: Optional. If provided, the value of this attribute tells XLSTAT the name of the R object that the end-user can specify here. CheckBoxes should typically be used for the R options that can either take the TRUE or FALSE values. XLSTAT-R uses also CheckBoxes for output options (to give the user the option to display or not a table or a chart) and the rname is not necessary in such cases. Default value: "".

text: Optional. The text that is used in the report when displaying the chosen value. Default value: "".

visible: Optional. The text that is used in the report when displaying the chosen value. Default value: "true".

Example:

```
<CheckBox text="Quantitative variables:" name="CheckBoxD2" rname=""  
refeditlinked="RefEditD2" onclick="If Not CheckBoxD2 then  
CheckBoxD3=True" default="true" />
```

This makes that when the checkbox is clicked (checked or unchecked), the RefEditD2 enabled property is changed (enabled if checked, disabled if unchecked), and if the CheckBoxD2 is not checked and if neither is CheckBoxD3, then CheckBoxD3 is set to True (meaning it is now checked).

```
<ComboBox rname="..." default="..." enabled="..." left="..." name="..." nextrow="..."  
onchange="..." reportgroup="..." text="..." tooltip="..." visible="..." width="..." >
```

A combobox is a list where you add several items (text or numbers) that are different options to choose for the user. XLSTAT-R only gives you the possibility to choose one option in the list as this is coherent with the syntax of R functions. This element is not sufficient on its own. You must include one or more items under the ComboBox level.

rname: **Required.** If provided, the value of this attribute tells XLSTAT the name of the R object that the end-user can specify here.. This value must be provided.

onchange: The code that is executed when the user changes his choice in the list.

default: Optional. The default value. It is provided as the index in the list, knowing that the indexes start at 0. Default value: "0".

enabled: Optional. Defines if the control can be used or not (it is then greyed out and not usable). If it is disabled, a change of the value of another control can make it enabled. This can be controlled by code related options (onchange, onclick) of other controls. Default value: "true".

left: Optional. Sets the number of pixels by which the control should be moved on the left. If you use a + before the value, the offset on the left will be by 12 times the number after the +. For example, "+2" creates an indentation of 24 pixels. Default value: "0".

name: Optional. The name of the control used by Excel internally. If not provided, XLSTAT-R sets an arbitrary name. Default value: "".

nextrow: Optional. Tells if the next control should be on a new row or not. Default value: "true".

reportgroup: Optional. The group of options to which this option belongs. Groups are displayed on the same line in the header of the XLSTAT report. If no value or "0" is provided, the selected ComboBox item will not be listed in the header of the report. If "0" or another integer is provided, then the rname will be used to replace the codes in the <RScript> nodes. Default value: "".

text: Optional. The text that is used in the report when displaying the chosen value. Default value: "".

tooltip: Optional. The text that is used when you hover the mouse over the control. If it is not entered the text value is used. Default value: "".

visible: Optional. The text that is used in the report when displaying the chosen value. Default value: "true".

width: Optional. Defines the width of the control. Default value: "155".

Example: see `<items ... />`.

`<item rname="..." text="..." />`

An Item is a member of the ComboBox control. It describes an element of the list contained in the ComboBox.

rname: **Required.** The value of this attribute tells XLSTAT the name of the R object that the end-user can specify here. This value must be provided.

text: Optional. The plain text to use in the list. If it is not provided or left empty (its default value), the rname is used instead. Default value: "".

Example: To handle the possible alternative hypotheses for a test, you can create a ComboBox with the following items:

```
<ComboBox text="Alternative hypothesis" name="ComboBoxAH"
rname="alternative">
```

```

    <item text="Lower tail" rname="less"/>
    <item text="Two sided" rname="two.sided"/>
    <item text="Upper tail" rname="greater"/>
</ComboBox>

```

<Spacer height="..." />: This is a virtual control (it is not displayed) that allows you to set the number of pixels that are inserted between the previous and next controls.

height: Optional. Number of pixels between the previous and next controls Default value: "6".

Note: You can use this control even if there no control below it. It is taken into account to define the height of the tab.

Example:

```
<Spacer height="10" />
```

This creates a vertical separation of 10 pixels between the previous and next controls, or creates a margin of 10 pixels below the last control in the current column of a given tab of the dialog box.

<Labels text="..." enabled="..." left="..." nextrow="..." name="..." visible="..." />

An Item is a member of the ComboBox control. It describes an element of the list contained in the ComboBox.

text: **Required**. The plain text to use as the label caption. Default value: "".

enabled: Optional. Defines if the control if the control should be greyed out (enabled="false") or not. If it is disabled, a change of the value of another control can make it enabled. This can be controlled by code related options (onchange, onclick) of other controls. Default value: "true".

left: Optional. Sets the number of pixels by which the control should be moved on the left. If you use a + before the value, the offset on the left will be by 12 times the number after the +. For example, "+2" creates an indentation of 24 pixels. Default value: "0".

name: Optional. The name of the control used by Excel internally. If not provided, XLSTAT-R sets an arbitrary name. Default value: "".

nextrow: Optional. Tells if the next control should be on a new row or not. Default value: "true".

visible: Optional. Tells if the control should be visible or not by default. Default value: "true".

<Group enabled="..." left="..." nextrow="..." name="..." text="..." visible="..." />

The Group control is used to group a series of controls. It is necessary to Group RadioButtons if you want them to be alternatives for the same option.

enabled: Optional. Defines if the control should be greyed out (enabled="false") or not. If it is disabled, a change of the value of another control can make it enabled. This can be controlled by code related options (onchange, onclick) of other controls. Default value: "true".

left: Optional. Sets the number of pixels by which the control should be moved on the left. If you use a + before the value, the offset on the left will be by 12 times the number after the +. For example, "+2" creates an indentation of 24 pixels. Default value: "0".

name: Optional. The name of the control used by Excel internally. If not provided, XLSTAT-R sets an arbitrary name. Default value: "".

nextrow: Optional. Tells if the next control should be on a new row or not. Default value: "true".

style: Optional. The style of the control. Use 0 if you do not want to display a border and 1 to display a border. Default value: "0".

text: Optional. The plain text to use as the group caption (text displayed at the top). Default value: "".

visible: Optional. Tells if the control should be visible or not by default. Default value: "true".

Example: see RadioButton

```
<RadioButton default="..." enabled="..." left="..." name="..." nextrow="..." onclick="..."  
refeditlinked="..." reportgroup="..." text="..." tooltip="..." visible="..." >:
```

A RadioButton is a control that you can check. Contrary to a CheckBox that you can also uncheck, it is dependent on the value of other controls. To uncheck it, you need to check another RadioButton that belongs to the same group of alternatives. So that a set of RadioButtons depend on each other and represent alternatives for the same option, you should group them in the same Group. For a first level on the left or right part of each tab, this is not necessary as XLSTAT already groups controls in a group (see the Missing tab of the t.test example).

tooltip: Optional. The text that is used when you hover the mouse over the RadioButton. If it is not entered the value of the text attribute is used. Default value: "".

default: Optional. The default value. It is the default value of the RadioButton. Default value: "false".

enabled: Optional. Defines if the control can be used or not (it is then greyed out and not usable). If it is disabled, a change of the value of another control can make it enabled. This can be controlled by vba code related options (onchange, onclick) of other controls. Default value: "true".

groupname: Optional. If it is entered, XLSTAT will use the text contained in the groupname to specify in the report which is the option the user chose. If it is not specified, the selected option is not displayed in the report. Default value: "".

left: Optional. Sets the number of pixels by which the control should be moved on the left. Default value: "0".

name: Optional. The name of the control used by Excel internally. If not provided, XLSTAT-R sets an arbitrary name. Default value: "".

nextrow: Optional. Tells if the next control should be on a new row or not. Default value: "true".

onclick: The code that is executed when the user checks or unchecks the CheckBox.

reportgroup: Optional. The group of options to which this option belongs. Groups are displayed on the same line in the header of the XLSTAT report. If no value ("" or "0" is provided, the value of the RadioButton will not be listed in the header of the report. If "0" or another integer is provided, then the rname will be used to replace the codes in the <RScript> nodes. Default value: "".

rname: Optional. If provided, the value of this attribute tells XLSTAT the name of the R object that the end-user can specify here. Default value: "".

text: Optional. The text that is used in the report when displaying the chosen value. Default value: "".

tooltip: Optional. The text that is used when you hover the mouse over the control. If it is not entered the value of the text attribute is used. Default value: "".

visible: Optional. Tells if the control should be visible or not by default. Default value: "true".

Example:

```
<Group style="0" left="+1" >
  <RadioButton text="Center" groupname="rescale" default="true"
  reportgroup="4" />
  <RadioButton text="Standardize" groupname="rescale"
  default="false" enabled="true" reportgroup="4" />
</Group>
```

The fact that the two RadioButtons are put in the same Group allows to make them possible alternatives of the same option, here the rescaling of the data. However, so that the XLSTAT report displays only the chosen alternative, it is also necessary to specify that they are alternatives, using the groupname attribute. The groupname can be anything unique and shared across the alternative RadioButtons. It is also used if there is a corresponding R option, to set the value of an R option to the right value.

`<Input default="..." enabled="..." left="..." name="..." nextrow="..." onclick="..." reportgroup="..." text="..." tooltip="..." type="..." visible="..." width="..." >`:

Use the Input control to let a user enter a text or a value. If the text field is specified, depending on the value of the optional attribute, a checkbox and/or a label will be added before the input field automatically.

default: Optional. The default value. It is the default value of the Input field. Default value: "".

enabled: Optional. Defines if the control can be used or not (it is then greyed out and not usable). If it is disabled, a change of the value of another control can make it enabled. This can be controlled by vba code related options (onchange, onclick) of other controls. Default value: "true".

left: Optional. Sets the number of pixels by which the control should be moved on the left. Default value: "0".

min: If the input is greater than that value, XLSTAT will prompt the user and invite him to change the input value.

max: If the input is greater than that value, XLSTAT will prompt the user and invite him to change the input value.

nextrow: Optional. Tells if the next control should be on a new row or not. Default value: "true".

optional: Optional. If set to false (the default), this attribute indicates that the Input control is preceded by a Label control. If set to true, this attribute indicates that the selection control is preceded by a CheckBox control and a Label. Default value: "false".

optiondefault: Optional. If the **optional** attribute is set to true, the optiondefault defines if, by default, the CheckBox is checked (true) or not (false). This defines the default enabled status of the Input control. Default value: "false".

reportgroup: Optional. The group of options to which this option belongs. Groups are displayed on the same line in the header of the XLSTAT report. If no value ("" or "0" is provided, the value of the control will not be listed in the header of the report. If "0" or another integer is provided, then the rname will be used to replace the codes in the <RScript> nodes. Default value: "", if rname="" and "0" if rname is not empty.

rname: Optional. If provided, the value of this attribute tells XLSTAT the name of the R object that the end-user can specify here. Default value: "".

rname_option: Optional. If the optional attribute is set to true, this R object will hold the status of the checkbox that activates the control or not.

text: Optional. The text that is used in the report when displaying the chosen value. Default value: "".

tooltip: Optional. The text that is used when you hover the mouse over the control. If it is not entered the value of the text attribute is used. Default value: "".

type: Optional. Tells what is expected in the input field. . Default value:

visible: Optional. The text that is used in the report when displaying the chosen value.
Default value: "true".

width: Optional. Defines the width of the control. Default value: "155".

Example:

```
<Input text="Iterations" tooltip="Enter the maximum number of iterations" rname="maxiter" type="integer" default="200" min="1" max="10000" left="115" width="40" reportgroup="2" />
```

This code allows to create an input field that allows the user to enter the number of iterations. The default value is set to 200. The minimum and maximum are respectively set to 1 and 10000 and XLSTAT will check that the value entered by the user is in that interval. The left is set to 115 so XLSTAT will use the first 115 pixels to display a label showing "Iterations".

6. Nodes for data Selection

One of the major advantages of XLSTAT is the two ways it offers for selecting data (on sheet selection with the mouse, or, as it is more usual for statisticians, selections of variables in a list).

```
<RefEdit text="..." tooltip="..." enabled="..." left="..." onebyone="..." rname="..." rtype="..." uniquecol="..." name="..." type="..." yvar="..." />
```

The RefEdit control allows to select data. XLSTAT offers two different ways to select data. You can either select data with the mouse anywhere on a sheet. Or you can select variables or samples in a list that has automatically been built by XLSTAT. The following attributes are available for the DataField control.

text: Required. The plain text to use in the report to describe the dataset being selected.
Default value: ""

tooltip: Optional. The text that is used when you hover the mouse over the DataField. If it is not entered the value of the text attribute is used. You should use it to give some information to the user on the type of data that need to be selected. Default value: "".

dataset: Optional. You can use the dialog box to select several datasets. You should put in the same dataset data that are homogeneous in size (number of rows). For example, if you are using a function that uses a model (say in linear regression), you can use the same data set for Ys and Xs. Or for RDA you can use the same dataset for the two tables. However, if you use tables of different sizes you should use different datasets. For example, with the t test, the first sample should be in dataset 1 and the second in dataset 2. Default value: "1".

enabled: Optional. Defines if the control can be used or not (it is then greyed out and not usable). If it is disabled, a change of the value of another control can make it enabled. This

can be controlled by code related options (onchange, onclick) of other controls. Default value: "true".

left: Optional. Sets the number of pixels by which the control should be moved on the left. If you use a + before the value, the offset on the left will be by 12 times the number after the +. For example, "+2" creates an indentation of 24 pixels. Default value: "0".

onebyone: Optional. Set this attribute to "true" so that if there are multiple columns provided in the data selection, then, one analysis is performed separately for each column. Default value: "false".

rname: Optional. The name of the data set that is used in the R function. If the rname is not provided, the data will be used for XLSTAT internal use only (typically for observation labels). Use different rnames for different datasets. Default value: "".

rtype: Optional. Default value: "dataframe". Currently, all input data are considered as being dataframes.

uniquecol: Optional. Set this attribute to true if you want XLSTAT to control if the user selected only one column as expected. Default value: "false".

name: Optional. The name of the control used by Excel internally. If not provided, XLSTAT-R sets an arbitrary name. Default value: "".

type: Optional. The XLSTAT data type. 1 (or "double" or "d" or "numeric") corresponds to a table of doubles (for example use it for Ys in a model). 2 corresponds to second table of doubles (for example use it for Xs). 3 (or "string" or "s" or "text" or "numeric") corresponds to a table of strings. 4 corresponds to a table of strings. 5 (or "weight" or "w") corresponds to a vector of weights. 6 corresponds a second vector of weights. 7 corresponds to a vector of strings that are used to characterize groups or subsamples. 11 corresponds to a vector of dates. Default value: "1".

Example:

```
<RefEdit tooltip="Select the time series in this field"
type="1" rname="x" onebyone="true" />
```

This code allows to generate the selection field that allows the user to define what are the 'x' data that will be used by R and, thanks to the onebyone option, XLSTAT knows that, if the user selects several columns, the function should be run for each column separately one after the other. The text of the Label or of the CheckBox is defined by the value of the text attribute.

```
<DataField text="..." tooltip="..." enabled="..." left="..." onebyone="..." optional="..."
optiondefault="..." rname="..." rtype="..." uniquecol="..." name="..." type="..." yvar="..." />
```

The DataField control corresponds to the grouping of a Label or a CheckBox control with a RefEdit. It has the same attributes as the RefEdit control, with two additional attributes to let XLSTAT-R know whether you need to use a Label or a CheckBox, and, if it is a CheckBox, what's its default value:

optional: Optional. If set to false (the default), this attribute indicates that the selection control is preceded by a Label control. If set to true, this attribute indicates that the selection control is preceded by a CheckBox control. Default value: "false".

optiondefault: Optional. If the **optional** attribute is set to true, the optiondefault defines if, by default, the CheckBox is checked (true) or not (false). This defines the default enabled status of the RefEdit control. Default value: "false".

rname_option: Optional. If the **optional** attribute is set to true, this R object will hold the status of the checkbox that activates the control or not.

Example:

```
<DataField text="Time series" tooltip="Select the time series in this field" type="double" rname="x" optional="true" optiondefault="false" onebyone="true" />
```

This code allows to do the same as the RefEdit example and to make XLSTAT add a CheckBox field before that is not activated by default.

```
<YQuantifield text="..." tooltip="..." enabled="..." left="..." onebyone="..." optional="..."  
optiondefault="..." rname="..." rtype="..." uniquecol="..." name="..." />
```

The YQuantifield node corresponds to a DataField where type="double" and yvar="true".

```
<YQualifield text="..." tooltip="..." enabled="..." left="..." onebyone="..." optional="..."  
optiondefault="..." rname="..." rtype="..." uniquecol="..." name="..." type="..." yvar="..." />
```

The YQualifield node corresponds to a DataField where type="string" and yvar="true".

```
<XQuantifield text="..." tooltip="..." enabled="..." left="..." onebyone="..." optional="..."  
optiondefault="..." rname="..." rtype="..." uniquecol="..." name="..." />
```

The XQuantifield node corresponds to a DataField where type="double" and yvar="false".
The default value for the text attribute is "Quantitative variables".

```
<XQualifield text="..." tooltip="..." enabled="..." left="..." onebyone="..." optional="..."  
optiondefault="..." rname="..." rtype="..." uniquecol="..." name="..." type="..." yvar="..." />
```

The XQualifield node corresponds to a DataField where type="string" and yvar="false". The
default value for the text attribute is "Qualitative variables".

If the XQuantifield or XQualifield nodes are present, a label before their respective labels or
checkboxes is added with "Explanatory variables" as text.

```
<ObsLabelsField text="..." add="..." />
```

The ObsLabelsField node corresponds to a CheckBox and a RefEdit for observation labels.
The XLSTAT datatype (type) is automatically set to "8" and the "onecolumn" attribute is set
to "true". The default text is "Observation labels".

add: Optional. The **add** attribute allows to specify if you want to add the observation labels
at the beginning of the first data object to load in R. Notice that R expects that all row labels
are different, which is not always convenient. XLSTAT adds them as a regular column to avoid
this constraint. If you want to use them as labels, your script should include an instruction so
that the first column becomes row labels and the remaining the regular dataset. You can use
the following code:

```
<ScriptLine code="newdata=olddata [ , -  
1 ] ; rownames (newdata) =olddata [ , 1 ] ; olddata=newdata ; rm (newdata) "  
/>
```

7. Special nodes

These nodes have been created to simplify the process of building interfaces.

<AutoOutputs stats="..." />

Use this node to ask XLSTAT to automatically generate the checkboxes that correspond to each RResult node and that correspond to an output.

stats: Optional. Set this attribute to true if you want XLSTAT to display the option to display the descriptive statistics of the selected data. Default value: "false".

<AutoCharts />

Use this node to ask XLSTAT to automatically generate the checkboxes that correspond to each RResult node and that correspond to a chart.

<VarHeaders text="..." />

The VarHeaders node corresponds to a CheckBox that allows the user to specify if the variable or column or series (...) headers are part of the selection or not. The default text is "Variable labels" and it is always checked by default. It should be positioned as the first element of the second column of the first tab. It is automatically positioned below the default output options (Range/Sheet/Workbook) in the first tab.

<Missing options="..." default="..." />

The Missing control is used to define the options displayed in the "Missing data" tab if available.

default: Optional. Enter which option is the default for the management of missing data (see options for the possible values). Default: "refuse".

options: Optional. The list of the features that are made available to handle missing data. Possible values are

- refuse: The user is alerted if missing values are found and the dialog box is displayed again.
- remove: The observations with missing values are removed from the dataset.
- ignore: The missing data are left in the dataset; This option is only relevant if the function that is called is able to handle missing data.
- replace_meanmode: The missing data are replaced by their mean (quantitative data) or their mode (qualitative data).
- replace_nn: The missing data are replaced by their nearest neighbor.
- replace_0: The missing data are replaced by 0.
- replace_previousnext: The missing data are replaced by the mean of the previous and next values.

Example:

```
<Missing options="refuse,remove,replace_meanmode,replace_nn"
default="remove" />
```

This generates 5 radio buttons, as there is one for refuse, one for remove, a general one to control the replace options and one for each replace option. The Remove option is the one checked by default.

8. Nodes to control outputs

This section gives the syntax for the RResults and Result nodes that allow to control how results are displayed, for both tables and charts.

<RResults></RResults>

The RResults node opens a section where you can insert as many <Result /> nodes as you want.

```
<Results rname="..." chartname="..." charttype="..." colformats="..." collabels="..."
datatype="..." color="..." hidden="..." rowlabels="..." rplotcode="..." rplotformat="..."
show="..." text="..." />
```

rname: Required. The name of the r result that corresponds to the result being described in the Result node. If the r function is myfunc, then the rname name1 corresponds to what you would obtain by doing myfunc(...)\$name1.

chartname: Optional. This attribute which is also available for checkboxes is used to determine whether the user wants to show the chart or not. If XLSTAT-R cannot find a CheckBox with the same chartname, the chart is displayed by default. Default value: "".

charttype: Optional. This attribute, if different from "r", where it means the chart is generated by R, allows to specify the type of XLSTAT chart that should be shown. The following options are currently available:

barchart(x/y) or barchart(x/y/z), where x is the position of the column used for categorical abscissa in the outputs table, y the position of the ordinates in the outputs column, and z is an optional value that tells the position of the lower bound values if a confidence interval is available.

coeffchart(z/alpha), where z is 0 if not confidence interval is available, and if a confidence interval is available, it gives the position of the column of the lower bound values in the coefficients table. alpha is the significance level used in the intervals (typically 0.05). Displays a bar chart of model coefficients with confidence intervals if available.

xyline(x/y), where x is the position of the column used for abscissa in the outputs table, and y the position of the ordinates in the outputs column. Displays a line chart.

xyscatter(x/y), where x is the position of the column used for abscissa in the outputs table, and y the position of the ordinates in the outputs column. Displays a scatter plot.

collabels: Optional. Use "" or "no" or do not specify anything if you do not want to use any column labels. Use "rdesc" if you want to use the labels provided by R. Use "obs" if you want to use the observation labels selected in the Observation labels field of the dialog box. Use "var" if you want to use the variable labels. Use expvar if you want to use the labels of the explanatory labels. Default value: "".

colformats: Optional. Enter a list of formats, separated by commas that should be used to set the format of each column. -1 corresponds to the standard XLSTAT numeric format, as set in the XLSTAT options dialog box (3 decimals is the default). -2 corresponds to the "General" format of Excel. -3 corresponds to the p-value format of XLSTAT (see the XLSTAT Options dialog box to control the number of decimals). -4 and -5 correspond to percentages with respectively 2 and 0 decimals. Positive values give the number of decimals to display. If you enter less values than there are columns, the last value is used for the remaining columns. Default value: "-1".

color: Optional. Enter the RGB color you want to apply to the chart. Example: color="(255/0/0)" for red. Default value: "".

datatype: Optional. The r format of the results table. Default value: "double".

format: Optional. This option allows to tell XLSTAT if a special formatting should be applied to a results table. Current available values are "covar" for a covariance matrix and "correl" for a correlation matrix. Default value: "".

hidden: Optional. Set this attribute to "true" if you want to hide the table of results in a hidden sheet. This is typically the case if you want to create a chart based on some values that are not interesting to the user, for example if you want to display a chart that shows the convergence of a process throughout iterations. Default value: "false".

rowformats: Optional. Enter a list of formats, separated by commas that should be used to set the format of each row. -1 corresponds to the standard XLSTAT numeric format, as set in the XLSTAT options dialog box (3 decimals is the default). -2 corresponds to the "General" format of Excel. -3 corresponds to the p-value format of XLSTAT (see the XLSTAT Options dialog box to control the number of decimals). -4 and -5 correspond to percentages with respectively 2 and 0 decimals. Positive values give the number of decimals to display. If you enter less values than there are rows, the last value is used for the remaining rows. Default value: "-1".

rowlabels: Optional. Use "" or "no" or do not specify anything if you do not want to use any row labels. Use "rdesc" if you want to use the labels provided by R. Use "obs" if you want to use the observation labels selected in the observation labels field of the dialog box. Use "var" if you want to use the variable labels. Use "expvar" if you want to use the labels of the explanatory labels. Default value: "".

rplotcode: Optional. This attribute allows you to enter the code that will generate the desired R plot. The code can either be simple or complex (see the kohonen(som) example for a table of plots). Default value: "".

rplotformat: Optional. This attribute allows you to choose the format of the images generated by R. Possible values are "emf" or "png". Default value: "png".

rpltheight: Optional. This attribute allows you to determine the height of the plot that is generated by R. Default value: "500".

rplwidth: Optional. This attribute allows you to determine the width of the plot that is generated by R. Default value: "500".

show: Optional. This attribute allows you to force the display of a result ("true") or not to display it "false". If you leave it to its default value "", the results will be displayed or not depending on the value of the corresponding checkbox, meaning the one with the same rname, if there is one. Default value: "".

text: Optional. The plain text to use as title before the result table is displayed. Default value: "".

9. Code

This section explains how you can use the RScript and ScriptLine nodes to customize the implementation of the R function.

<RScript replacebyvalue="..." ></RScript>

The RScript node opens a section where you can insert as many <ScriptLine /> nodes as you want.

replacebyvalue: **Required.** If set to true, XLSTAT replaces the rnames found in the code by their value as defined in the interface by the user, when the **replace** argument in the <ScriptLine /> nodes is set to true. Otherwise XLSTAT looks for the rnames and set them to their value in the code before the code listed in the <Scriptline> is added. In other words, it creates corresponding R objects named after the rnames attributes and set to the value specified by the user.

<ScriptLine code="..." engine="..." replace="..." />

code: **Required.** Enter the R code that you want to run.

engine: Optional. Tells which software should be used to run the code. Default value: "r".

replace: Optional. The replace attribute tells XLSTAT-R whether some parameters in the code should be replaced by their value as defined by the user in the interface. If replace is set to false, the code is used as provided. Default value: "false".

Example:

```
<RScript replacebyvalue="true" >  
  <ScriptLine code="Means=c(mean(x1[,1]),mean(x2[,1]))" />  
  <ScriptLine code="results=t.test(x1,x2, paired, alternative,  
var.equal)" />  
</RScript>
```

```
<RScript replacebyvalue="false" >  
  <ScriptLine code="Means=c(mean(x1[,1]),mean(x2[,1]))" />
```

```
<ScriptLine code="results=t.test(x1,x2, paired=pa, alternative=
alternative, var.equal=veq)"/>
</RScript>
```

The two codes are equivalent. In the first one, we show three different strategies: Either you use the arguments names and set them first using the rnames that will be replaced by their interface value (here for paired and var.equal), or you use another object, set its value to the rname, or you use another object, set its value to the rname that can also be the name of the argument.

The code corresponding to replacebyvalue="true" is more compact and writes:

```
Means=c(mean(x1[,1]),mean(x2[,1]))
results=t.test(x1,x2, paired=FALSE, alternative='two.sided',
var.equal=FALSE)
```

The code corresponding to replacebyvalue="false" is more compact and writes:

```
paired<-FALSE
alternative<-'two.sided'
veq<-FALSE
Means=c(mean(x1[,1]),mean(x2[,1]))
results=t.test(x1,x2, paired=pa, alternative=alternative,
var.equal=veq)
```