SOFTWARE—PRACTICE AND EXPERIENCE
Softw. Pract. Exper. 2015; **00**:1–21
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/spe

BIOMON: DATA ANALYSIS AUTOMATION IN BIOMONITORING †

Rodrigo Bühler [1] *, Jorge Luis Victória Barbosa

[1] *Academic Master in Applied Computing, Concentration Area: Modeling and Simulation, University of Rio do Sinos Valley - Unisinos, São Leopoldo, RS, Brazil*

SUMMARY

In the field of research, data analysis plays a fundamental role in understanding the dynamics of an environment or event subjected to observation. Scientists and researchers develop their hypothesis regarding an observed issue and test it against field evidence collected and tested via scientific methodologies, techniques or procedures. In order to support the researchers in testing their assumptions many different technological tools are accessed, each one with its own paradigm in terms of usage and technical specificities. This characteristic forces the user to deal not only with challenges related to one field of study, but also others related to the operation of each required software. Biomon is a software model designed to support the researchers to in their most important task, interpret their field data. The contribution resides in sharing responsibilities between the researcher that owns the data and the statistician, that masters tools and techniques of data analysis. The collaboration happens when the statistician prepares a script file with the steps, sequence of statistic methods, to execute a data analysis and the researcher with the help of Biomon applies such steps over data collected in the field. As a proof of concept Biomon was implemented in programming language R and evaluated by biomonitoring researchers with the use of TAM (Technology Acceptance Model) acceptance test. The final results were quite revealing presenting good rates in the evaluation of ease of use and usefulness. Areas of improvement were highlighted in the layer of user interface showing that it is an important area in the reduction of the technological complexity of a software.

Received . . .

KEY WORDS: research, data analysis, statistics, R, TAM,

1. INTRODUCTION

When a researcher is working in the field collecting evidences, whether inside a forest looking for new species or in an office digging databases, lots of data are collected, organised and stored. This data is evidence of a target subject collected in a specific moment in time and space, pieces of a continuum that tell something about what is observed. In order to make assumptions or prove a hypothesis it is necessary to run comparisons, measure tendencies or even describe patterns from other information basis that may be inferred from the data. The inference is supported by a collection of statistical methods that combined become an analysis that represents an observed aspect of the collected data.

Scientific papers with application of sets of statistical methods in the field of data analysis, such as [1], with the application of randomisation testing with Monte Carlo methods such as bootstrapping and multivariate analysis of variance (MANOVA) or other making use of Bayesian

*Correspondence to: E-mail: rodrigo.buhler@ymail.com

inference methods described in [2] and [3] are examples of the combination of statistical methods for data analysis.

With the help of technological tools, researchers are able to collect, organise and store the field data that afterwards, making use of statistical softwares, are analysed in search of answers to important questions, evidence to their assumptions or confirmation to their proposed hypothesis. The model presented in this paper aims at adding value to this last step, in an attempt to make it more natural to the researcher that usually has to deal with skills that are different from their area of study and because of the valuable amount of time spent in its execution.

The reduction of the overall complexity proposed is achieved by the distribution of workload and complexity between the statistician and the researcher provided by the software. Biomon presents an implementation of an architecture in which the statistician uses some skills to deliver sets of analysis configuration script files. The researcher chooses the most proper analysis approach to apply over data collected in the field (figure 1). As it was described in [4] both researcher and statistician have different approaches and levels of knowledge in terms of technological tools and languages.

The software architecture aims at delivering an environment where both statistician and researcher may interact asynchronously. Biomon makes use of analysis script files defined by a statistician to execute data analysis, and these files describe the proper sequence of statistical methods and their parameters. The support provided to the researcher is dedicated to a clean graphical user interface (GUI) in which few steps are necessary to inform the field data and with the help of a chosen analysis file, execute the required data analysis.

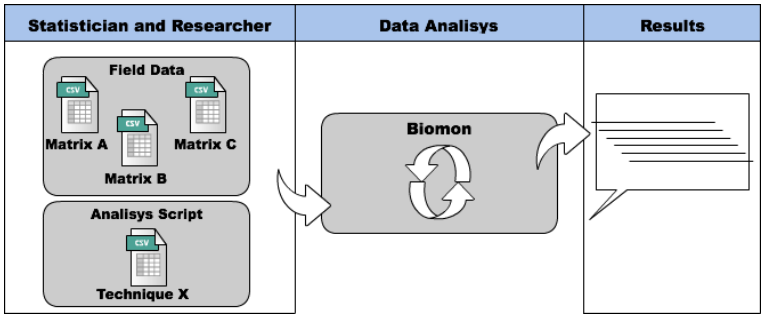


Figure 1. Biomon High Level Representation.

The statistician is able to provide a script file with a sequence of statistical methods to execute field research database analysis. The file format chosen is the coma separated values (CSV), mainly because it is largely supported by most of the spreadsheet software. The statistician with the knowledge of the file structure presented in this paper is able to describe in a configuration file a statistical data analysis accessing functions and data available in the execution environment. Such file is interpreted by Biomon and the steps executed with the data informed by the researcher.

The participation of the researcher has few interactions and low complexity, the figure 2 presents an illustration on the expected participation of each role in terms of complexity and interaction with software. Taking into account that the researcher already has ones field data properly organized in spreadsheets, the software provides an interface in which is possible to choose the data that will be statistically analysed. In order to execute any sort of analysis, the researcher has to inform the way it shall happen, and this takes place by informing to the software the analysis script file provided by a statistician, who can share such files as any other electronic information.

2. MODEL

Next are presented details on regards to the way Biomon is deployed followed by a description of its architecture. The participation and responsibility of the statistician and the researcher are presented as an enclosing description of the overall model purpose.

DOI: 10.1002/spe

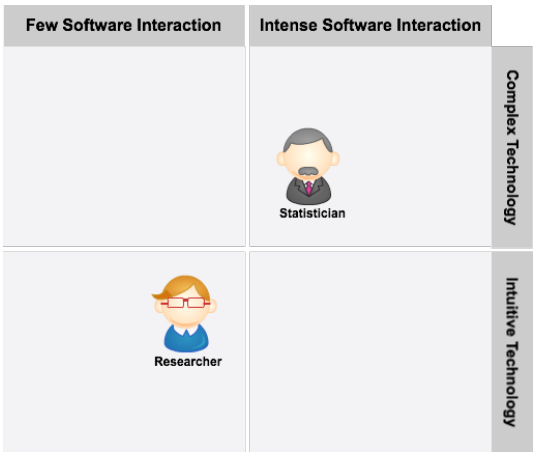


Figure 2. Interaction *vs* Technological Complexity

2.1. Deployment

The software model is split in three main layers, as defined by the Model-Views-Controllers design pattern firstly described in [5] and [6]. Such approach supports low-impact improvement in the overall software and provides a contextualised maintenance environment. Each layer may be delivered in separated packages of libraries, figure 3, that combined become a instance of usable software.

2.1.1. Views layer: The Biomon application delivers in the view layer the interface in which the researcher interacts with the software. Considering that most of the complexity is already handled by the model layer and controllers (detailed next) there are few expected options in terms of interface artifacts the user might fece with. Such simplification is key for a software that proposes ease of use and low demand in terms of technological skills.

A clean and acclimated interface aims at delivering to the final users, the researchers, a tool for data analysis that support their needs and help them to focus on what is most important, that is the results interpretation. Together with the possibility to run all the necessary analysis steps in one single place the user interface provides to the users an unique environment free of different paradigms of usage, without the need for master skills other than the user's area of study.

2.1.2. Controllers layer: The controllers as the name suggests are, together, the central piece of the software controlling its behaviour to the user interaction. The behaviour delivered via user interface is supported by this layer that handles the user requests and takes care of the resource calls returning a feedback whenever it makes necessary.

Behaving as a router of user calls the controllers are designed to respond to the external interaction calling one or a set of resources necessary to execute an action. The mentioned resources delivered by the model layer (detailed next) are accessed in a predefined sequence in order to return to the user a proper feedback that may vary from a single load of field data until a complete execution of steps of an analysis. The same way the direct interaction with the user is transferred to the view layer that access the software resources via calls to the controllers.

2.1.3. Model layer: The model is the software layer where the application is modelled and the main class and methods are defined. The analysis script interpreter engine is delivered on this layer, and specialisations may be built on top of it. These specialisations may cover different areas of research just requiring a dedicated GUI (the view) and routines to manage its behaviour and interaction with the model (the controllers).

For Biomon the main purpose of the model layer is to delivery the resources necessary to interpret the script with the analysis steps and execute them over the informed field data. At this stage the

DOI: 10.1002/spe

modelling should be as generic as possible transferring the specialisation to the next layers, this way covering a wide range or data analysis fields. Such approach fosters the implementation of new instances of software and platforms acclimated to the field of study it is intended to support.

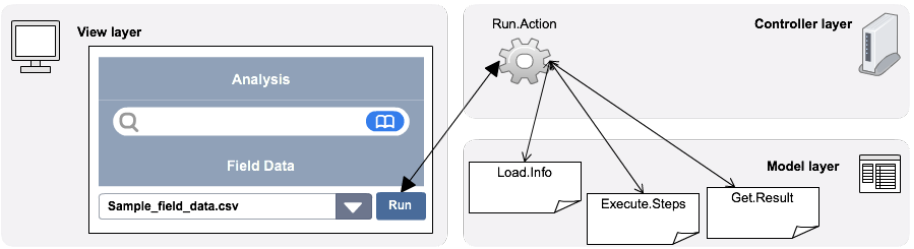


Figure 3. Model-Views-Controllers

2.2. *Architecture*

In order to present the architecture of Biomon, the describing model 4+1 View Model of Software Architecture ([7]) is applied with few adaptations, aiming at covering the software specificities. The following items present each of the proposed view models with a description of their components and the way they behave integrated as one piece of software.

1. **Logical View:** Figure 4 presents the Logical View in which the so-called functional requirements are presented, or in other words, what is provided to the final user. The "Field data" and "Analysis CSV" are what the user would interact directly with at the interface of Biomon that also provides action triggers handled by the controllers, which, in turn, forwards these calls to the routines implemented in the model that execute them.

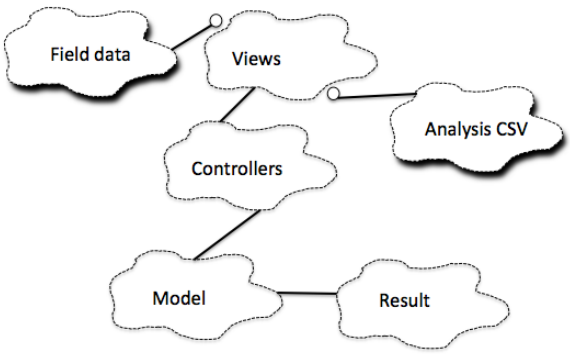


Figure 4. Logical View

2. **Process View:** Figure 5 provides a more detailed picture of the internal interaction of Biomon. The Process View considers a deeper aspect of the architecture taking into account non-functional requirements, describing the way the methods are split for the sake of specialisation and maintainability.
3. **Development View:** Figure 6 presents the current modularisation of Biomon and what is expected from each module. The representation bottom-up describes a user action and the way the software responds to such interaction in each of the layers (implemented in packages). It is possible to notice that layer **1.Views** has the most interactive participation, allowing the user to trigger actions handled by layer **2.Controllers** that has both interactive and reactive characteristics, translating the user requests into a sequence of calls to layer **3.Model** which, as it is expected, reacts to these requests.

DOI: 10.1002/spe

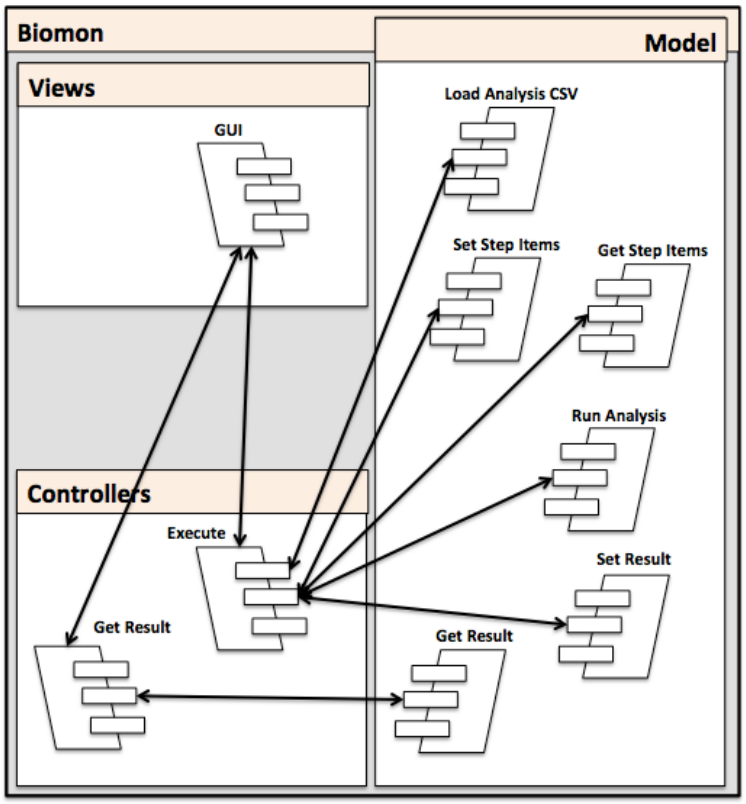


Figure 5. Process View

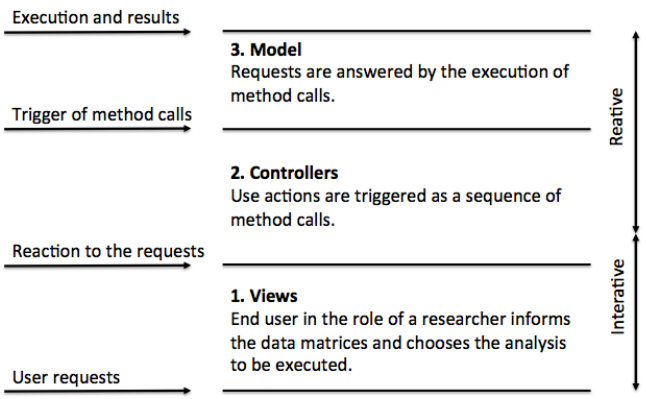


Figure 6. Development View

4. **Physical View:** The physical view relates to the physical distribution of the designed solution, covering distributed systems and hardware architectures. Biomon is composed by a set of packages in the same local environment, there is no access to any sort of external resource. That is why figure 7 is limited to presenting this assemblage.
   As already stated before Biomon is presented as a set of three packages representing the MVC (Model-Views-Controllers) model architecture. The view package joins the user interface artefacts such graphical items or instructions in-line for implementations in command line. In order to handle the user actions triggered by the interface the controllers package delivers a set of calls to methods implemented on the model package.

DOI: 10.1002/spe

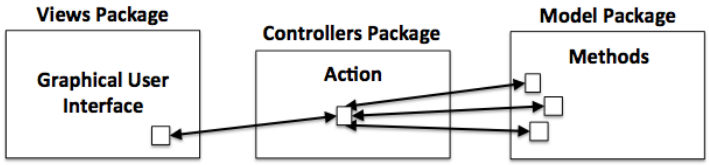


Figure 7. Physical View

5. **+1 View:** The **+1 View** relates to the representation of the integration of all views in practical examples or cases of use. A representation showing the way the researcher will interact with the interface and the flow of such interaction is presented in the figure 8.

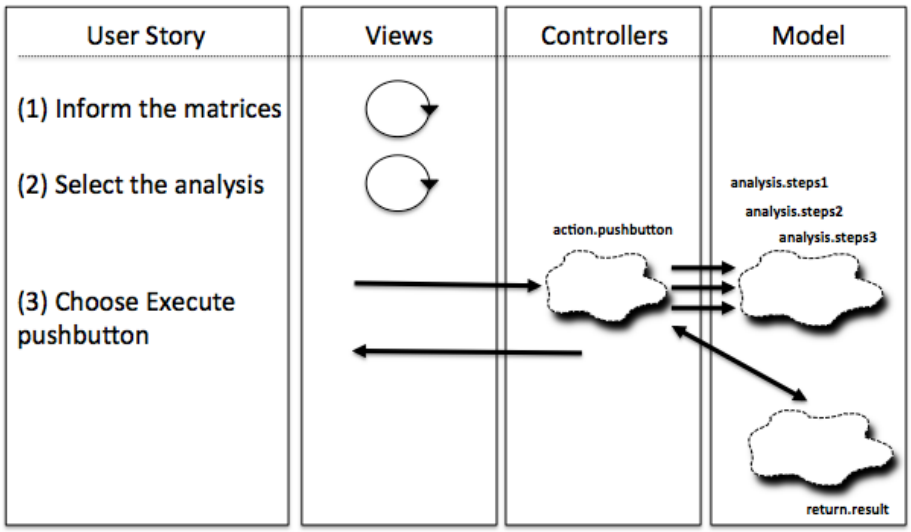


Figure 8. +1 View

2.3. *The Roles*

The presented model proposes a shared participation on the software execution between the researcher and the statistician. These roles are presented separately just for description purposed, in real life they may be the same person. Considering the area of expertise of each character Biomon delimits their responsibilities and boundaries.

2.3.1. Statistician: In the context of Biomon the statistician is responsible for delivering a script file with the steps for running a analysis over data collected in the field. There is no need for get contact with the data, the scripting process has well defined rules that abstract the such information allowing the statistician to develop a dynamic file that may fit to a wide range of data collections.

In order to promote such dynamism a protocol is stablished between the statistician and the instance of Biomon. This contract defines global variables informed by the researcher (direct or indirectly) and accessible by the script delivered by the researcher. An example of such statement is presented in the table I:

2.3.2. Researcher: From the researcher it is expected the information of field data and the analysis file to Biomon so the analysis steps execution may take place and the result presented. This way the

DOI: 10.1002/spe

Table I. Global Variable Definition Example

Global Variable	Description
gbl_mtx_environment	Environment data matrix
gbl_mtx_species	Species data matrix

analysis can be interpreted by the researcher, that can run the same sequence of calculations with a new set of data the many times it makes necessary.

With no need for learning how to access different interfaces from several other softwares neither re-arrange the format of the data collected in the field the researcher is free to invest ones time on collecting data and interpreting the analysis results.

3. RELATED WORKS

The observation presented next take into account the way the statistical support is provided by the available softwares. Some examples present as a purpose cover an analysis context delivering a generalist support to different statistical techniques, covering a wide range of options. In other hand it is possible to observe solutions highly specialised covering a specific statistical technique delivering to the final users a combination of tools and features to support them to refine the informed data and execution parameters.

As a matter of presenting practical examples of the above mentioned softwares next are briefly described four solutions from the academic field, well known as supporting tools for data analysis. The samples presented will consider analysis and technique contexts and a brief description is presented on the way the user interacts with them.

3.1. Multiv & SYNCSA:

Multiv is a free for use tool for statistic calculations on regards to exploratory analysis, randomisation testing and bootstrap resampling with multivariate data. Does it present to the users a command line interface where they navigate informing a chosen function in a list of options typing a letter that represents the chosen one.

The download and instructions may be found at `http://ecoqua.ecologia.ufrgs.br/ecoqua/MULTIV.html`

In a quite similar way SYNCSA is also a free for use statistical software released in command line, but still slightly more focused on ecology data analysis. The tool covers matrix correlations between well defined data structures. The author describes which information is expected and the calculations consider such definition.

The download and instructions may be found at `http://ecoqua.ecologia.ufrgs.br/ecoqua/syncsa.html`. There is also an implementation in R of SYNCSA available at `https://cran.r-project.org/web/packages/SYNCSA/` with most of the original functions implemented.

Due to the nature and context of their implementation Multiv and SYNCSA are often referenced by publications in the area of ecology such as [8], [9], [10] and [11].

Both Multiv and SYNCSA may be considered tools dedicated to cover a defined analysis context, supporting users in a specific area of research and making use of different statistical methods. This way fostering new interpretation approaches providing a wide range of possibilities of data analysis.

3.2. Past:

Past was formerly designed to support studies in palaeontology, currently it is delivered in versions for operational systems Windows ®and Mac OS X ®. The software presents to the final user a GUI (Graphical User Interface) with menus, buttons and icons that guide the user on the execution of statistical methods and results presentation.

DOI: 10.1002/spe

With the aim at supporting the context of palaeontology studies, Past offers to the researcher a set of statistical methods that are applied over data collected in such studies. Despite that it may be possible to adapt the usage of the software in areas other than paleontology, past presents to the user an environment to support the targeted area of study.

The download and instructions may be found at http://folk.uio.no/ohammer/past/index_old.html (ver. 2.17c) , for the former version and at http://folk.uio.no/ohammer/past/ for the current version (August 2015 - 3.08).

Past citations may be found in several publications related to Palaeontology area of study, such as the most recent ones [12], [13] and [14] or other areas like ecology [15].

In the same way as Multiv and SYNCSA it is possible to state that Past is a statistical software dedicated to support an analysis context. Despite it may be feasible to apply some of the provided tools in areas other than the main one, it may require some sort of adaptation.

3.3. PC-ORD:

In a different approach if compared to the softwares previously presented, PC-ORD offers and environment that aims at providing to the user tools to execute data ordination and classification. Despite most of the applications are in the area of biology the software can easily be applied in any other area of study that may demand such techniques.

The operation takes place in an graphical interface, where de imported data can be edited and transformed with the use of the provided statistical methods.

The download and instructions may be found at https://www.pcord.com.

PC-ORD citations may be found in several publications, not necessarily under the same context, such as [16], [17] and [18].

Different from what was presented so far, PC-ORD covers a specific statistical technique (or techniques related) not directly related to any area of research (analysis context).

3.4. CANOCO:

Just like PC-ORD, the software CANOCO provides to the final user a graphical interface to inform and process research data. It is delivered support for multivariate data analysis via several statistical techniques such as Canonical Correspondence Analysis (CCA) and Principal Correspondence Analysis (PCA) between others.

The data may be imported from MS Excel®spreadsheets, processed and the result presented making use of the visualisation tools of CANOCO.

The download and instructions may be found at http://www.canoco5.com.

CANOCO itations may be found in several publications, not necessarily under the same context, such as [19], [20] and [21].

4. PROTOTYPE

4.1. User Interface

4.2. Action Controllers

4.3. Data Analysis Model

4.4. Statistician

The statistician has a central participation in the software life cycle, he or she is responsible for connecting the engineering implemented on Biomon with the final user needs. To reach such objective, a few simple rules are defined in order to link the statistician knowledge and the data analysis. Basic skills on the use of spreadsheet software are required to translate R functions and parameters into lines and columns of a CSV analysis file. Previous experience on the use of R may become helpful when deciding the way an analysis shall proceed.

DOI: 10.1002/spe

4.4.1. CSV file layout The CSV file layout represents the specification implemented as a means to inform the software about the steps to be followed. Lines and columns described below represent the current software release specification. The file may be created manually in a simple text editor, but it may get hard-working for more complex analysis assemblages or even because some functions have too many parameters. That is why it is highly recommended to use a spreadsheet software. Most of them (if not all) have a functionality to read and export CSV files.

The next items describe what is expected from such analysis file:

- **Composition:** each line shall contain at least four columns as presented in the table II. The first column shall contain an identification that is represented by a string with the name of the step. The second column is the R command to be called, its parameters are described in the third column informing the parameter type and the fourth column its value.

Table II. CSV file layout

Column 1	Column 2	Column 3	Column 4
Identification	R Command	Parameter Type	Parameter Value

- **Supported parameter types:** as described above, for each parameter value informed it is necessary to inform what kind of value it is. That is why the Biomon interpreter will coerce such value to an informed type and below are described the types supported and the correspondent R function they represent (table III).

Table III. Parameter Value Type

Value Type	R related function
null	as.null(variable)
numeric	as.numeric(variable)
double	as.double(variable)
character	as.character(variable)
logical	as.logical(variable)
vector	c(variable)
list	list(variable)

- **More parameters:** some R functions require more than one parameter. For such cases columns three and four shall be repeated; in other words, the next parameters shall be represented by a new pair of columns with the type and value of the next parameters. This procedure is repeated until all the parameters are informed.
- **Values already calculated:** calculations already performed in previous lines may be accessed as a parameter for new calculations with the use of the token @. Each execution has its result stored into a results table following the same sequence of the methods defined in the CSV file. This way when a parameter for a calculation is the result of an already executed line, the statistician just needs to do the following:
  - In column Parameter Type inform the token @;
  - In column Parameter Value inform the line number the result should be taken from.

  This way the functions of the package **analyz** will inform as a function parameter an already calculated value, with no need for coercion as the value already has its final format/type.
- **Complex parameters:** some functions require not only a single value, but also a more complex data type, such as a matrix or data frame. The statistician then combines the steps described above, for instance, creates a vector with the required list of values, then sending the result of this step as a parameter. This way a complex parameter (vector) will be sent to the function instead of a list of values.
- **Global parameters:** when implementing the analysis files, the statistician does not have access to the researcher data, or they may vary depending on the set of data to be analysed

DOI: 10.1002/spe

under the same perspective. To handle this issue Biomon delivers to the statistician a set of global variables that are accessible in time of creating the analysis file and entered in time of software execution. In a similar way as described for the already calculated values, the access to the global variables is made by the use of a special token, as described below:

  - In column Parameter Type inform the token $;
  - In column Parameter Value inform the name of the variable to be read (see table **??**).

- **CSV standards:** The remaining standards and definitions that may apply are found at http://tools.ietf.org/html/rfc4180.

The final result of an analysis definition should look like the figures below. The file presented represents a mean calculation of two pre-defined variables. The objective of the example is to present a simple output of a CSV file.

- Figure 9 presents the appearance of an analysis file to determine the mean in a spreadsheet editor;

	A	B	C	D	E	F
1	v_A	as.numeric	numeric	7		
2	v_B	as.numeric	numeric	3		
3	v_Vector	c	@	1	@	2
4	Mean	mean	@	3		
5						
6						

Figure 9. Analysis file in a spreadsheet editor

- Figure 10 presents the same analysis file above, but in a plain text editor.

```
v_A,as.numeric,numeric,7,,
v_B,as.numeric,numeric,3,,
v_Vector,c,@,1,@,2
Mean,mean,@,3,,
```

Figure 10. Analysis file in a plain text editor

It is expected from the statistician a little knowledge in biomonitoring and R as the current development is acclimated under such themes. There is no need to access any real data in time of assembling the analysis file other than few generic examples to ensure the calculation of the expected results. Such independence is key for the good of compatibility between the different sets of collected data that the analysis should support.

4.5. Researcher

The researcher plays the role of software consumer. The main reason for this software is to help analysis execution with collected field data. Considering that most of the hard working has already been done, it is necessary just to interact with the interface in order to inform the data and choose the most suitable analysis according to the needs.

DOI: 10.1002/spe

There are many techniques that can be used to collect, store and organise research field data. For the proposed software we considered what is described in [22]. Table IV presents the matrices supported by the package **biomonCore**. The researcher should consider this distribution when informing to Biomon the data collected in the field.

Table IV. Matrices of Field Data.

Matrix	**Rows** **Observation**	**Columns** **Variables**	**Content**
Attributes (B)	Species	Attributes	Quantitative values or Presence/absence
Species (W)	Sample Unit	Species	Quantitative values or Presence/absence
Space (S)	Sample Unit	Geogr.Coordenates	A Real value representing a coordinate variable
Factors (T)	Attributes	Species	T = B' * W (transposed B multiplied by W)
Environment (E)	Sample Unit	Environ.Variables	Quantitative values

As a matter of standardisation it is expected that the matrices contain header names in the first row and row names in the first column just like figure 11.

	A	B	C	D	E	F	G	H	I	J
1	Species	Trait_A	Trait_B	Trait_C	Trait_D	Trait_E	Trait_F	Trait_G	Trait_H	Trait_i
2	pos2	1	1	0	0	0	0	0	0	0
3	magy	1	1	0	0	0	0	0	0	1
4	phpa	1	0	1	0	0	0	0	0	1
5	lecy	0	0	0	1	0	1	0	1	0
6	bsp4	1	0	1	0	0	0	0	0	1
7	brle	1	1	0	0	0	0	0	0	1
8	phbr	1	0	1	0	0	0	0	0	1
9	lep3	1	1	0	0	0	1	1	0	0
10	gdin	1	1	0	0	0	0	0	0	0
11										

Figure 11. Data matrix

4.5.1. Scenario For a clear understanding of the way the researcher will interact with the interface and run an analysis, we present below a short description of a user story and a graphical representation (figure 12) on how it takes place.

1. The researcher informs the matrices with the data collected in the field;
2. The researcher informs the analysis script file with the steps to be executed;
3. The researcher executes Biomon with the informed data;
4. The result of the last step is presented back the user.

4.6. Design

Description ...

4.6.1. View Taking into account the proposed distribution of responsibilities, which considered the skills of each character, a low technological complexity and few interaction steps are expected in the execution of a data analysis by the researcher. It is up to the application Biomon to deliver such environment. Figure 13 presents the GUI the researcher will interact with.

4.6.2. Controllers Description ...

DOI: 10.1002/spe

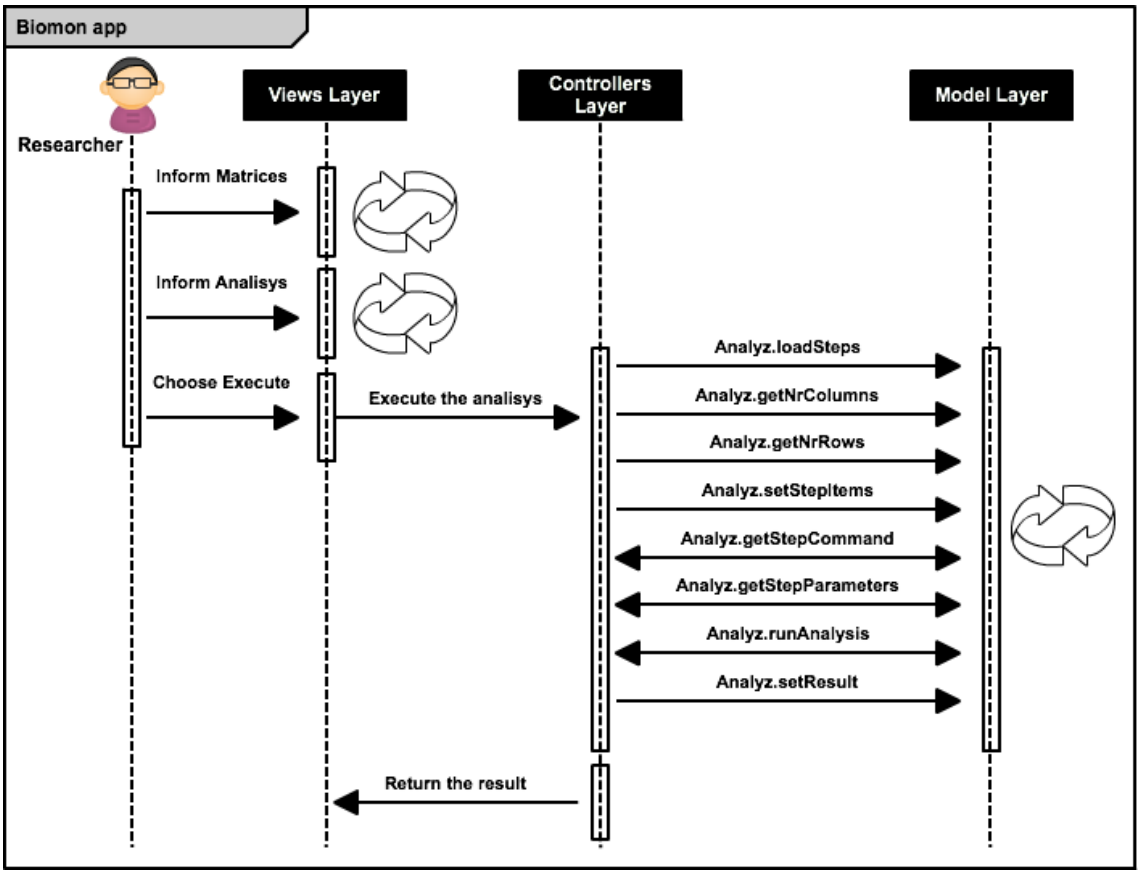


Figure 12. Scenario View

4.6.3. Model Description ...

4.7. Use case

In order to present how Biomon works in a practical way, we present next a use case scenario with a description of each necessary step. The application delivered with this paper contains sample files for analysis and a set of data matrices. The field data delivered is a small part of a real research data base organized just for test purposes. The analysis file, due to its complexity and importance, is described in details to reinforce the statements described earlier in this paper.

- **User Interface:** the researcher (final user) need to have some knowledge on R environment and commands. As it was already stated, Biomon is composed by three packages **biomon**, **biomonCore** and **analyz**, all delivered with this paper or accessible via GitHub link described in chapter 9. This last option gives you access to the under development versions of each package. The first step is to install these packages in your R environment and execute Biomon in command line the same way as any other R function:
  ```
  library("biomon")
  biomon()
  ```
  In a correct execution a web browser opens and the Biomon GUI is presented to the researcher, just like figure 13. The interface the user has access displays:
  - Upload buttons and the paths Biomon will search for the analysis files and data matrices (figure 14);
  - Drop-down list where the user informs the analysis file to be interpreted by biomon(figure 15);

DOI: 10.1002/spe

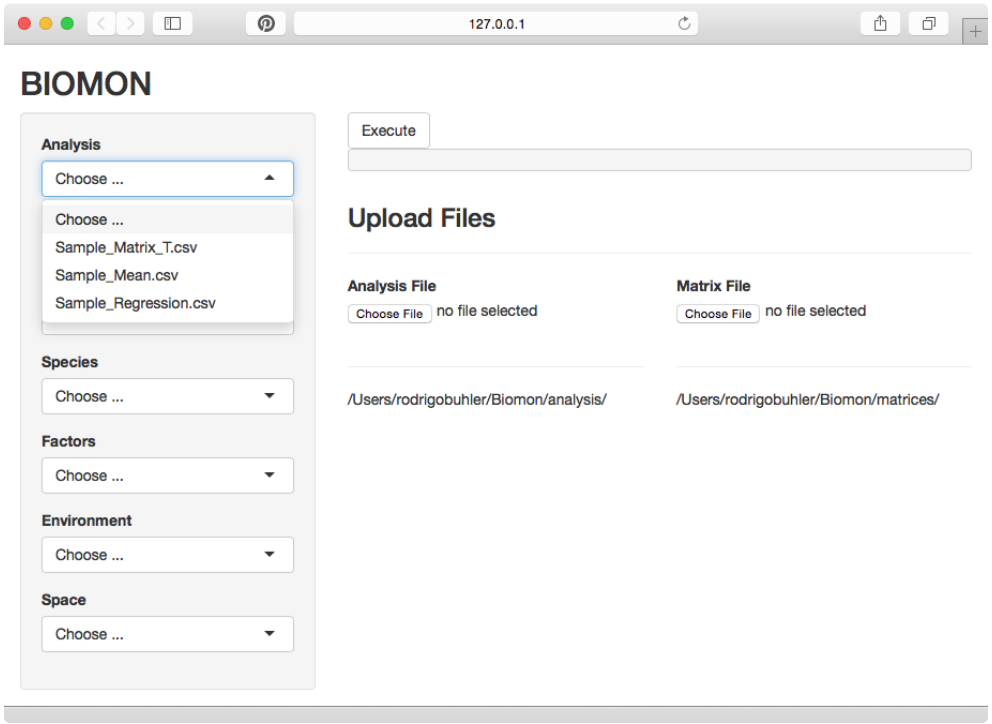


Figure 13. Biomon interface

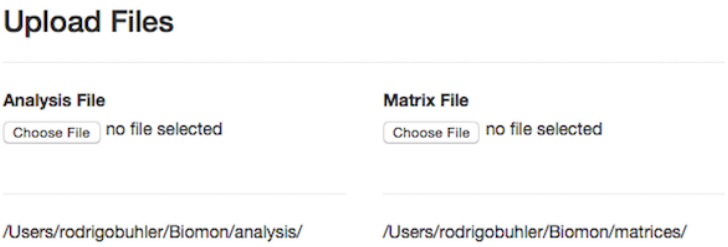


Figure 14. Biomon Repositories

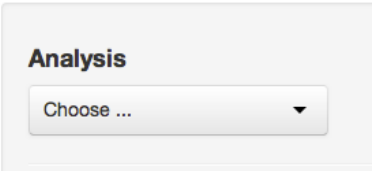


Figure 15. Biomon Analysis Files

  - Drop-down lists where the user informs the data matrices that relates to each data context (figure 16);
  - Execute pushbutton and the area the result and messages are displayed (figure 17).

- **Analysis file:** For this use case we will use the analysis file Sample_Matrix_T.csv that is assembled to generate the Factor matrix. Basically what is needed is to transpose the data matrix Attributes and multiply the new matrix by the matrix Species resulting in the Factor matrix that will be written in a physical CSV file and made available for further processing. For that it is required from the researcher that he or she informs the Attributes and Species

DOI: 10.1002/spe

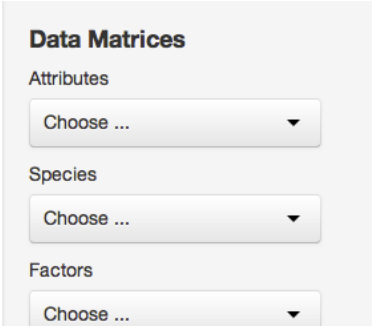


Figure 16. Biomon Data Matrices

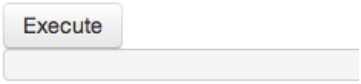


Figure 17. Biomon Execute

matrices as well as the analysis file mentioned before and chooses the Execute pushbutton. As a result, a new data matrix is available in the matrices repository and the content of this matrix is presented below the pushbutton in the interface.
In order to reach the result described above those steps were assembled into the analysis file as presented in table V and described in details below. It is important to notice that the numbers in the first column and the letters in the first row are presented just as matter of reference, they are not part of the final script.

1. **Read the matrix Attributes:** as a first step the matrix Attributes is read from the CSV file the user informed. In the analysis file the first column contains a short identification of the step, and this is the same for all lines. From column B onwards, the steps are defined and parametrised in a way Biomon can interpret and execute in R.
   For this step it is defined that function "read.table" is called to handle the data import. The first parameter is the CSV "file" path (parameter type $ value gbl.mAttributes) to be read followed by the logical TRUE for the parameter "header", the character "," for the parameter "sep", a NULL for "quote", a character "." for "dec" and a numeric 1 for "row.names". The result of this execution is stored in the first position of the results internal table and may be accessed with the parameter type @ value 1.
2. **Read the matrix Species:** basically the same as before, that is, the matrix Species is loaded with a small adjustment in the value for the parameter "file" where the global variable to be called is gbl.mSpecies. The result of this step follows the same rule already described and is stored in the second position of the results internal table. This result procedure is repeated for all remaining steps.
3. **Attribute data as matrix:** with the matrices loaded to the R environment, that it is necessary to ensure they have the proper format loading the result of line one as function parameter and informing Biomon to execute "as.matrix".
4. **Species data as matrix:** the same as the previews step, but for the result of line two.
5. **Transpose matrix:** at this point it is possible to transpose the Attributes matrix. For that, it is necessary to inform the function "t" and the position where to get the loaded matrix.
6. **Multiplication of matrices:** now it is possible to execute the main step of this script file, the matrices multiplication. The function choosen is "tcrossprod" and the parameters are the position of the matrices Attributes' (transposed) and Species in the results internal table.

DOI: 10.1002/spe

7. **New file name and path:** in order to write the new table in a physical CSV file it is necessary to define a path and a name. This step concatenates the global variable gbl.matrices (matrices repository) with the vector "New_Factor.csv". The result is a path for a new file into the matrices repository, this way it is accessible for further processing by Bimon.
8. **Write down the file:** everything it is necessary to create the new data matrix is available, then the function "write.csv" is informed in this step with the following parameters:
   - Type @ value 6 - position six in the results table for the parameter "x";
   - Type @ value 7 - position seven in the results table for the parameter "file";
   - Type logical value "FALSE" - for the parameter "append".
   - Type logical value "FALSE" - for the parameter "quote".

   This way a new CSV file is created in the matrices repository and will be accessible by Biomon.
9. **Read the created matrix:** in order to present as a result, the content of the new created matrix is loaded back to Biomon. As it was done in step one and two, the file is loaded with the same set of parameters just informing the path for the new file from the position seven in the results table.
10. **Present a result:** as it was already stated before, this release of Biomon just presents the value of the last execution, this way it is necessary to execute a last step in order to present a result to the user. For that we choose the function "as.matrix()" that returns a matrix from its parameter, in this case the new loaded matrix.

The description above illustrates the importance the statistician has for the software usability. The more functional and well assembled the analysis file is, the more value they aggregate to Biomon as a data analysis tool. It is also possible to notice that the complexity derives more from the assemblage of the analysis scripts than the software operation. This way the researcher does not need to waste time dealing with technological demands and can focus on interpreting the results.

Table V. Sample_Matrix_T.csv

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	read_attr	read.table	$	gbl.mAttributes	logical	TRUE	character	","	null	null	character	.	numeric	1
2	read_spec	read.table	$	gbl.mSpecies	logical	TRUE	character	";"	null	null	character	.	numeric	1
3	matrix1	as.matrix	@	1										
4	matrix2	as.matrix	@	2										
5	transp_mtx	t	@	3										
6	fact_mtx	tcrossprod	@	5	@	4								
7	path	paste0	$	gbl.matrices	character	New_Factor.csv								
8	write	write.csv	@	6	@	7	logical	FALSE	logical	FALSE				
9	read_spec	read.table	@	7	logical	TRUE	character	";"	null	null	character	.	numeric	1
10	matrix	as.matrix	@	9										

5. SOFTWARE ACCEPTANCE TEST

As it was observed in [23], the lack of software acceptance tests has being determinant in the failure of projects in information systems area. Developments driven by performance are affected by the rejection of the users in operate such systems. Taking into account that the acceptance test has fundamental participation in the identification of failure or success factors in information systems, Biomon was evaluated on regards to its acceptance by researchers of the field of biomonitoring.

5.1. Methodology

The methodology followed was the execution of a prototype and next applied a survey to the user on regards to the perceived experience. To the invited researchers it was sent a guide describing a use case (Appendix **??**) and a survey form (Appendix **??**). After the execution of the described steps the researchers had conditions to evaluate each stage of a real situation. From the software

DOI: 10.1002/spe

installation, in an R environment, until the execution of a real scenario, freely executing exploratory tests whenever it was considered necessary. At the end the researchers were invited to evaluate their degree of acceptance for a group of statements.

The survey was applied to six researchers with different academic and professional profiles as described in the table VI:

Table VI. Researchers

Identification	Academic Profile	Professional Profile
Interviewee 1	Doctoral student in Ecology	Researcher with Doctoral scholarship at UFRGS
Interviewee 2	Doctoral student in Environmental Quality	Master in Environmental Quality
Interviewee 3	Doctoral student in Ecology	Researcher with Doctoral scholarship at UFRGS
Interviewee 4	Master in Ecology	
Interviewee 5	Doctor in Ecology	Professor at UFRGS
Interviewee 7	Doctor in Environmental Sciences	Researcher at the state Zoobotanica Foundation

5.2. Test Case

The researchers invited to participate on the software acceptance test received all the same set of documents via mail. When executing the tests the users were not assisted *in loco*, any support was provided via electronic means. This way the researchers had the opportunity to evaluate the software usability looking for answers of their doubts in the software by their own.

The test case (Appendix **??**) applied describe the steps necessary to execute a analysis in the Biomon. It starts with a brief description of the data matrices along with the software installation and finally the test case description.

5.3. Survey

The survey (Apêndice **??**) was created based on TAM (Technology Acceptance Model) firstly described on [24] and published on [25]. The proposed methodology aims at predicting and describing the user acceptance of technological systems based on their two major perceptions, the ease of use and software usefulness.

The users perception that the software does not aggregate any value to their work, characterize the usefulness perception. In other hand the effort spent on operating an information system describe the perception of ease of use. This last one has such an importance to the user that may override the perception of usefulness.

For the survey's evaluation it was proposed a set of statements (tabela VII) where the researchers had to score each sentence following the Likert scale [26], with 5 options of score varying from **1. I totally agree until 5. I totally disagree**.

Table VII. Likert Items Statements

Group	Item	Statement
Ease of Use	1	I have comprehended the usage of all the screen elements without external help.
	2	The interface delivers all the necessary functionalities for the software operation.
	3	I could follow the whole test case steps without any difficulties.
	4	In general I consider the software Biomon easy to use.
	5	I consider the interface simple and pleasant.
Usefulness	6	Taking into account the way I analyze my field data I consider that Biomon may ease my daily job.
	7	I could replace the way I do my analysis by the use of only Biomon.
	8	I enjoyed the de way the software shares the responsibilities between the statistician and the researcher.
	9	I consider important the possibility of easily share analyse script files.
	10	With Biomon I can save time on executing the analysis and invest more on interpreting the results.

Although it was on purpose kept not clear to the researchers, the sentences were grouped into two sets of statements with different objectives. The first one trying to evaluate the user experience in terms of software usage and interface.

DOI: 10.1002/spe

The second set of statements concerned of aspect more practical, such as the replacement of the current data analysis procedures by the proposed software.

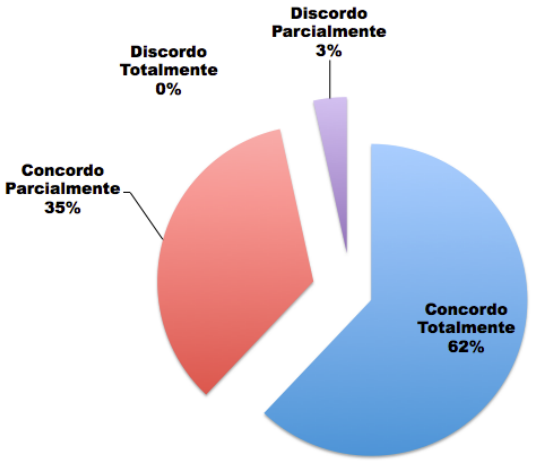
At the end of the testing process the evaluations were compiled as follows:

6. EASE OF USE

Ease of use [27] concerns the impression the user experience in time of operating the software. It takes into account if the software usage is free of effort and will happen in a more natural and intuitive way. In order to evaluate such aspect the researchers were invited to score statements related to the interpretation and execution of the test case.

The figure 18 presents the scores compilation of the test case steps execution aspect. It is possible to observe that approximately two thirds (62%) of the scores corroborate to the fact that the perception of ease of use is clear. Other positive point that must be highlighted is that only 3% of the scores regards to the score *I partially disagree*.

Figure 18. Facilidade de Uso Avaliação



Fonte: Elaborada pelo autor

Dentre as cinco questões ilustradas na figura 19 pode-se destacar as questões dois e três como pontos de atenção pois apresentam índices muito altos, o que sinaliza um certo desacordo com a realidade do pesquisador. Estas questões se referem a informações de ajuda na interface e facilidade na execução dos passos do caso de teste.

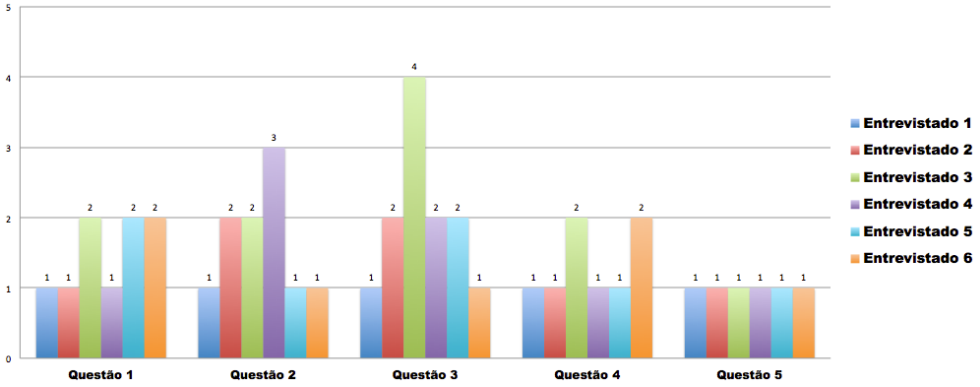
As avaliações destas questões apontam para a necessidade de um maior refino na elaboração da interface do software para que na sua execução as situações onde possam haver dúvidas o usuário tenha condições de resolvê-las com a ajuda do próprio software.

7. USEFULNESS

A utilidade percebida se caracteriza [27] pelo nível de confiança que o usuário tem de que o software em uso pode efetivamente agregar valor ao seu trabalho. Desta forma as afirmações deste grupo procuraram colocar o protótipo e sua execução no contexto de análise de dados do pesquisador convidado, para que este possa avaliar de maneira fiel o cenário executado.

DOI: 10.1002/spe

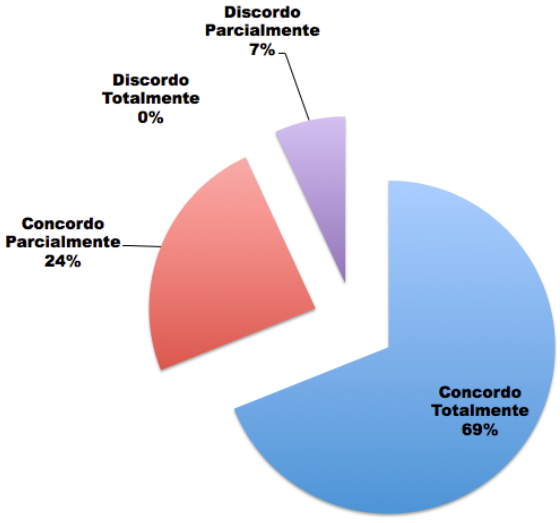
Figure 19. Facilidade de Uso Questões



Fonte: Elaborada pelo autor

É possível observar na figura 20 que mais de dois terços (69%) das afirmações foram consideradas totalmente alinhadas com a realidade do entrevistado. Já os resultados de discordância em relação as afirmações somam apenas 7% do total geral.

Figure 20. Avaliação da Utilidade Percebida



Fonte: Elaborada pelo autor

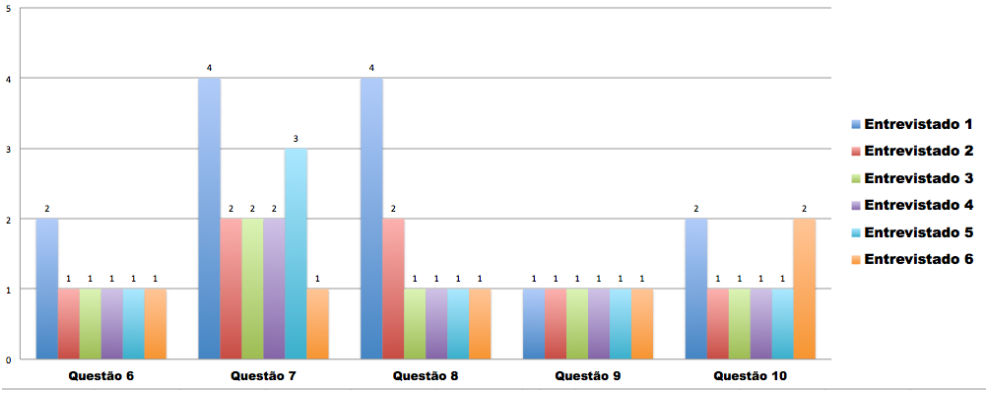
Dentre as cinco questões ilustradas na figura 21 pode-se destacar as questões sete e oito por apresentam índices discrepantes em relação ao restante do conjunto, o que sinaliza uma falta de consenso a respeito.

A questão sete trata da possibilidade de substituição total da maneira como o pesquisador faz suas análises de dados de campo pela proposta apresentada. É de se esperar que esta questão receba índices mais elevados por se tratar da avaliação de um protótipo com poucas funcionalidades, mas ainda assim não houve qualquer indicação de rejeição absoluta a ideia.

DOI: 10.1002/spe

A questão oito trata da separação dos papeis do estatístico e do pesquisador, novamente é possível observar que não há consenso. Apesar de boa parte dos entrevistados terem concordado totalmente com a afirmação um pesquisador assinalou para uma discordância parcial. Um fator que pode explicar os índices desta questão é o nível de conhecimento estatístico que o pesquisador possui, o que o aproxima ou afasta do papel do estatístico. Desta forma o fato de não haver mais a necessidade de lidar diretamente com a elaboração das análises é interpretado como algo que agregaria valor ao seu trabalho como pesquisador. Por outro lado, quando há um maior conforto na elaboração das análises a divisão dos papeis pode representar uma perda do controle ou da possibilidade de refino dos passos a serem executados.

Figure 21. Utilidade Percebida Questões



Fonte: Elaborada pelo autor

Ao final da avaliação os pesquisadores eram convidados a deixarem algum comentário textual a respeito da experiência. Algumas das observações feitas pelos entrevistados são reproduzidas a seguir:

- *"Acho que o Biomon facilitará em muito minhas análises, para isso preciso dominar melhor o programa. Até aqui achei prático e de fácil utilização.'*
- *"Considero a experiência promissora e interessante. A possibilidade de interagir mais e melhor com um estatístico profissional é realmente animadora. A cada dia novas e desafiadoras metodologias surgem e a análise dos dados, ou melhor, a simples compreensão estatística demanda muito tempo. Isto, muitas vezes, nos faz desistir de uma análise nova, ficando com aquelas que já conhecemos. Neste sentido, uma relação mais próxima com um estatístico e, ao mesmo tempo, com uma interface amigável de análise pode de fato contribuir muito com o progresso da pesquisa e com o aumento e a qualidade das publicações geradas com dados coletados em campo. Acredito que, com um prazo maior de contato entre ambos (estatístico e o pesquisador) possa de fato conduzir a interessantes resultados e análises."*

Os dois comentários acima descrevem diferentes pontos de vista em relação a experiência vivenciada pelos pesquisadores durante a execução do caso de teste, mas cobrem os objetivos propostos. Enquanto que o primeiro comentário abordada as características de praticidade e facilidade de uso o segundo comentário faz referência a uma das formas propostas pelo Biomon para atender as referidas características. Com a separação dos papeis de pesquisador e estatístico a complexidade das análises é compartilhada com entre o pesquisador que possui a necessidade e o estatístico que possui o conhecimento técnico que possibilita a elaboração de soluções para atender a estas necessidades.

DOI: 10.1002/spe

8. DISCUSSION

This paper presents a proposal of implementation of software for data analysis automation in Biomonitoring with the objective to deliver a tool that helps the researcher to focus attention on what is most important, in this case, the data collection and analysis result interpretation. To reach such objective the software architecture considered aspects such as clear definition of responsibilities on customising and operation of the software, delivering an environment where statisticians and researchers may work asynchronously, each on their own area of expertise.

The Biomon workflow contemplates two distinct roles, each responsible for delivering to the software contribution based on each area of expertise. The architecture allows the assemblage of analysis files that are read and interpreted by the software. These files contain the steps necessary to execute a statistical analysis and are the responsibility of a statistician or a skilled user. On the other hand, it is required from the researcher (final user) the information of research data in a defined format, taking into account the context of biomonitoring. This way both analysis file and field research data may interact and give the final user the choice to execute determined data analysis, using specific research information.

The proposed software was acclimated in the context of biomonitoring as a matter of proof of concept, and the packages delivered were implemented in a way that they may be applied in subjects other than biomonitoring requiring little adjustments. This characteristic fosters the support of different areas of research and facilitates the adaptation to specific situations or needs in the area of data analysis automation.

A functional version of Biomon is delivered with this paper. It is a version that implements everything that is described and detailed in this proposal and has the objective of presenting a functional software that may be executed and tested. It deserves some improvements in terms of error handling and information to the user on what may be wrong when things do not happen as expected.

The conclusion is that the main objective has been accomplished. The proposal could be implemented and statisticians and researchers may work collaboratively each in their own area of expertise, successful analysis files may be easily shared between researchers or even slightly adapted to cover different aspects of data analysis. The final result is a set of packages that may be used as a base for new developments or even to reduce the time and effort spent on developing such software.

9. CODE

Biomon is a completely open source development and may be accessed/installed from:

1. GitHub
   - Package **analyz** - `https://github.com/rbuhler/analyz`
   - Package **biomonCore** - `https://github.com/rbuhler/biomonCore`
   - Package **biomon** - `https://github.com/rbuhler/biomon`

REFERENCES

1. Pillar VDP, Orlóci L. On randomization testing in vegetation science: multifactor comparisons of relevé groups. *Journal of Vegetation Science* 1996; **7**(4):585–592, doi:10.2307/3236308.
2. Smith AF, Roberts GO. Bayesian computation via the gibbs sampler and related markov chain monte carlo methods. *Journal of the Royal Statistical Society. Series B (Methodological)* 1993; :3–23.
3. de Oña J, López G, Mujalli R, Calvo FJ. Analysis of traffic accidents on rural highways using latent class clustering and bayesian networks. *Accident Analysis & Prevention* 2013; **51**:1–10, doi:10.1016/j.aap.2012.10.016.
4. Wallace BC, Dahabreh IJ, Trikalinos TA, Lau J, Trow P, Schmid CH. Closing the gap between methodologists and end-users: R as a computational back-end. *J Stat Softw* 2012; **49**(5):1–15.
5. Reenskaug T. Thing-model-view-editor—an example from a planningsystem. *technical note, Xerox Parc* 1979; .
6. Reenskaug T. Models-views-controllers. *Technical note, Xerox PARC* 1979; **32**:55.

7. Kruchten PB. The 4+ 1 view model of architecture. *Software, IEEE* 1995; **12**(6):42–50, doi:10.1109/52.469759.
8. Cruz P, De Quadros FLF, Theau JP, Frizzo A, Jouany C, Duru M, Carvalho PCF. Leaf traits as functional descriptors of the intensity of continuous grazing in native grasslands in the south of brazil. *Rangeland Ecology & Management* May 2010; **63**(3):350–358, doi:10.2111/08-016.1.
9. Hepp LU, Milesi SV, Biasi C, Restello RM. Effects of agricultural and urban impacts on macroinvertebrates assemblages in streams (rio grande do sul, brazil). *Zoologia (Curitiba)* 2010; **27**(1):106–113, doi:10.1590/S1984-46702010000100016.
10. Duarte LDS, Bergamin RS, Marcilio-Silva V, Seger GDDS, Marques MCM. Phylobetadiversity among forest types in the brazilian atlantic forest complex. *PloS one* August 2014; **9**(8):10, doi:10.1371/journal.pone.0105043.
11. Pereira D, Mansur MCD, Duarte LD, de Oliveira AS, Pimpao DM, Callil CT, Ituarte C, Parada E, Peredo S, Darrigran G, *et al.*. Bivalve distribution in hydrographic regions in south america: historical overview and conservation. *Hydrobiologia* July 2014; **735**(1):15–44, doi:10.1007/s10750-013-1639-x.
12. Ausín B, Flores JA, Sierro FJ, Bárcena MA, Hernández-Almeida I, Francés G, Gutiérrez-Arnillas E, Martrat B, Grimalt J, Cacho I. Coccolithophore productivity and surface water dynamics in the alboran sea during the last 25kyr. *Palaeogeography, Palaeoclimatology, Palaeoecology* 2015; **418**:126–140, doi:10.1016/j.palaeo.2014.11.011.
13. Chester SG, Bloch JI, Boyer DM, Clemens WA. Oldest known euarchontan tarsals and affinities of paleocene purgatorius to primates. *Proceedings of the National Academy of Sciences* 2015; **112**(5):1487–1492.
14. Nicholson DB, Mayhew PJ, Ross AJ. Changes to the fossil record of insects through fifteen years of discovery. *PloS one* 2015; **10**(7), doi:10.1371/journal.pone.0128554.
15. Kneitel JM, Lessin CL. Ecosystem-phase interactions: aquatic eutrophication decreases terrestrial plant diversity in california vernal pools. *Oecologia* 2010; **163**(2):461–469, doi:10.1007/s00442-009-1529-0.
16. Lee SH, Kang HJ, Park HD. Influence of influent wastewater communities on temporal variation of activated sludge communities. *Water research* 2015; **73**:132–144, doi:InfluenceofinfluentwastewatercommunitiesontemporalvariationofactivatedsludgecommunitiesInfluenceofinfluentwastewatercommunitiesontemporalvariationofactivatedsludgecommunities.
17. Herrera-Rangel J, Jiménez-Carmona E, Armbrecht I. Monitoring the diversity of hunting ants (hymenoptera: Formicidae) on a fragmented and restored andean landscape. *Environmental entomology* 2015; :nvv103doi:10.1093/ee/nvv103.
18. Semenova TA, Morgado LN, Welker JM, Walker MD, Smets E, Geml J. Long-term experimental warming alters community composition of ascomycetes in alaskan moist and dry arctic tundra. *Molecular ecology* 2015; **24**(2):424–437, doi:10.1111/mec.13045.
19. Neumann JL, Griffiths GH, Hoodless A, Holloway GJ. The compositional and configurational heterogeneity of matrix habitats shape woodland carabid communities in wooded-agricultural landscapes. *Landscape Ecology* 2015; :1–15doi:http://dx.doi.org/10.1007/s10980-015-0244-y.
20. Bluhm C, Scheu S, Maraun M. Oribatid mite communities on the bark of dead wood vary with log type, surrounding forest and regional factors. *Applied Soil Ecology* 2015; **89**:102–112, doi:http://dx.doi.org/10.1016/j.apsoil.2015.01.013.
21. Nylund L, Nermes M, Isolauri E, Salminen S, Vos dW, Satokari R. Severity of atopic disease inversely correlates with intestinal microbiota diversity and butyrate-producing bacteria. *Allergy* 2015; **70**(2):241–244, doi:http://dx.doi.org/10.1111/all.12549.
22. Pillar VD, Duarte LdS, Sosinski EE, Joner F. Discriminating trait-convergence and trait-divergence assembly patterns in ecological community gradients. *Journal of Vegetation Science* 2009; **20**(2):334–348.
23. Davis FD. User acceptance of information technology: system characteristics, user perceptions and behavioral impacts. *International journal of man-machine studies* 1993; **38**(3):475–487, doi:10.1006/imms.1993.1022.
24. Fishbein M, Ajzen I. *Belief, attitude, intention and behavior: An introduction to theory and research*. Reading, Mass. : Addison-Wesley Pub. Co., 1975.
25. Davis FD. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly* 1989; **14**(3):319–340, doi:10.2307/249008.
26. Likert R. A technique for the measurement of attitudes. *Archives of psychology* 1932; .
27. Marangunić N, Granić A. Technology acceptance model: a literature review from 1986 to 2013. *Universal Access in the Information Society* 2015; **14**(1):81–95, doi:10.1007/s10209-014-0348-1. URL http://dx.doi.org/10.1007/s10209-014-0348-1.