v. 3, n. 26, 2023

Journal of
Engineering
Research

DEVELOPMENT OF A WEB APPLICATION WITH SHINY/R TO PROMOTE THE TEACHING-LEARNING OF STATISTICS

Alvaro Toledo
Universidad Bernardo O'Higgins

Katherine Delgado
Universidad Bernardo O'Higgins

Daniel Montenegro
SYMMTEC Chile

Abstract: This article presents the experience of developing a virtual laboratory prototype to promote the teaching-learning of statistics through the Shiny package of the statistical software R. Shiny allows the creation of a user interface based on buttons and selection windows that allow interaction with elements presented to the user. The aim of this study is to show the general structure and the considerations of developing a tool of this type, including the modules for the main interface, the activities included in the virtual laboratory and the possible interaction options, mentioning its limitations. The programmed modules focused on simple and multiple linear regression analysis considering the study of the assumptions of this type of models. In conclusion, Shiny is a powerful tool to create technological teaching tools, since it allows the creation of friendly modules that facilitate the application of statistical content. This study is recommended for those who are new to interactive application development and have considered open-source software as a resource.
Keywords: ICT, Web application, Shiny/R, Teaching-Learning of Statistics

INTRODUCTION

The need for process automation in different areas of knowledge has grown rapidly over the years, presenting a considerable increase during the recent COVID-19 pandemic. Education is not exempt from this phenomenon, leading its different actors to develop technological solutions that allow access and communication between teachers and students, in addition to providing ICT tools that favour the learning of the contents of the respective academic curricula. According to Gámiz (2009):

> ICTs provide us with new means to be able to transmit information instantly and with formats that provide greater reality and quality to messages, establishing new communication systems that break the barriers of time and physical space. (p.47)

Authors such as Carrillo & Maldonado (2009) refer to the importance of ICT in the educational process, along the same lines as Ferro et al. (2009) highlighting the advantages of using ICT in the teaching-learning process from the point of view of the teachers themselves. From this need the idea of generating quality ICT tools arose in order to provide teachers with a mechanism to transfer methodologies to meet the needs of the productive sectors, including highly complex concepts and tools to develop Machine Learning solutions. The integration of ICT to teaching is a motivational strategy for students, since, among other advantages, it increases the methodological variety through the generation of graphical user environments, enhances data processing through the visualization of graphics, allows the generation of simulations for the study of the variability of the parameters in real time, just to mention a few.

In relation to ICT and in the teaching-learning process of Statistics, studies such as those by Lunsford et al. (2006) and Hagtvedt et al. (2007) based on input and output questionnaires show considerable improvements in the performance of students who implement technological solutions acquired in the process of learning statistics. According to Chance & Rossman (2006):

> In teaching statistics, it is important to both model the use of technology as a problem-solving tool and take advantage of the ability to automate calculations and graphs, facilitate the exploration of statistical concepts through dynamic, interactive visual environments, and experiment first-hand the stochastic Nature of statistical processes. (p.1)

Although a large number of studies show the need to use software or interactive applications in the teaching-learning process of statistics, there are difficulties related to at least two points. Potter et al. (2016), emphasize: "despite the large collection of existing applet teaching tools found on the web, eventually an instructor may encounter a problem finding an existing applet that perfectly suits their needs" (p. 1). Almerich et al. (2011) analyze the training needs of ICT teachers and deepen these deficiencies and complexities. Related information can be found in Toledo & Vicencio (2021) and Das (2019) where the lack of software knowledge on the part of teachers is addressed.

The idea of a working virtual laboratory prototype using a web application developed in the Shiny package of R software offers a new context for teaching and learning free from the constraints of time and space. Accordingly, the development project was based on the following general objective: Develop a virtual laboratory prototype for the teaching-learning process of Statistics and, on the other hand, on the following specific objectives: 1. Design and implement a graphical environment prototype for different users, navigation and operation of the virtual laboratory for learning Statistics stage by stage. 2. Design the operational module of the virtual laboratory for learning Statistics for the modality of challenges. Its design was also based on Machine Learning, allowing the Virtual Laboratory to develop personalized learning paths that help students achieve their goals, performing the process of filtering, grouping and selecting content regardless of the student's learning pace and the starting point you take. The centerpiece of this application is a detailed visual grid, which together with the idea of the simulator allows the student significant learning unlike traditional methods.

The following sections will present the programming tools used, the methodology, the different stages and resources dedicated to the development of this web application.

TOOLS AND METHODOLOGY

R SOFTWARE AND RSTUDIO

The virtual laboratory application for learning Statistics was designed and programmed fundamentally in R language (R Core Team, 2022). R (https://www.r-project.org/) is a collaborative and open project, distributed under the GNU license, of an object-oriented interpreted programming language developed by scientists at the Department of Statistics at the University of Auckland, New Zealand, Ross Ihaka and Robert Gentleman in 1993. R promotes the creation of packages for collaboration between users and the growth of the language. It is one of the most widely used programming languages in scientific research, machine learning, data mining, biomedical research, bioinformatics, financial mathematics, and data science in general. On the other hand, RStudio is an integrated development environment (IDE: Integrated Development Environment, in English) for the R programming language, dedicated to statistical computing and graphics. It includes a console, syntax editor that supports code execution, as well as tools for tracing, debugging, and workspace management.

The use of R as a tool for learning statistics has been widely described in the literature. Authors such as Garfield & Ben-Zvi (2008), Harraway (2012) and Zieffler et al. (2012) show that concepts such as randomness, sampling, and the study of variability through simulations can be effectively explained by visualizing them with the use of statistical software. Among them, R., in Toledo & Vicencio (2022a, 2022b), addresses the use of visualizations of statistical concepts through the graphic package ggplot2 (Wickham , 2011) pointing out the difficulties of the "graph grammar" exposed by Wilkinson (2005) and the basis of ggplot2 programming , but also showing the advances of other R packages that use the power of ggplot2 but simplify enormously the number of instructions that must be programmed to obtain the visualizations.

THE SHINY PACKAGE OF R SOFTWARE

Shiny (Chang et al., 2015), is a web application framework for R conceptualized by RStudio Chief Technology Officer Joe Cheng and his team as a tool designed to help R programmers create interactive web applications, reports, and analysis without need to know HTML, CSS or JavaScript (Figure 1). According to Varma & Virmani (2017), "Shiny is essentially an HTML document hosted on a computer running R" (p. 5). Shiny has gained popularity with the first Shiny Developer Conference held in January 2016 and increasing use in peer-reviewed scholarly articles (Kasprzak et al., 2020). According to Toledo & Vicencio (2023), "Researchers from different areas are seeing in Shiny an opportunity to bring statistical concepts closer to students through a programmable interface, which can be complemented with existing tools in R" (p. 15). There is a vast literature regarding applications based on Shiny in the context of teaching-learning statistics. Authors such as Arnholt (2019) show the use of Shiny as a tool for understanding the relationship between sample size and power of the test. In Berg (2021) the understanding of Bayesian reasoning is promoted and shows the differences between Bayesian and non-Bayesian approaches with the support of the Shiny interface. In Depaoli et al. (2020) Shiny applications are presented to help understand and teach a *priori sensitivity analysis* by implementing Bayesian estimation methods. In Miranda (2019) a Shiny application is presented, which allows the teacher to show the probability density in three steps (graphical comparison between empirical and theoretical distribution, empirical distribution differences when varying the sample size and representation of density and distribution accumulated). Satyahadewi & Perdana (2021) present and discuss the use of Shiny/R for the comparison of means. Stratton et al. (2021) present an application based on Shiny focused on teaching sampling distributions and properties of estimators. Von Borries & De Castro (2022) show the roc_app application that helps students understand the Receiver Operating Characteristic (ROC) curve, as well as other concepts associated with binary classification models. Finally, Williams & Williams (2018) present the R Shiny application to help the confidence interval learning process using graphs and data from the US National Basketball Association. Applications in other fields can be found in Wojciechowski et al. (2015) who present the application of the Shiny package to the simulations of pharmacometric models. In Gopinath et al. (2021) Shiny is used for data analysis in agricultural research. In Johnson et al. (2021) an application developed in Shiny is presented to teach the different stages of a geostatistical analysis in a more interactive way. In Moore et al. (2022) Shiny is used for research and exploration based on environmental data.

Shiny is an application that can be used in many contexts because it has the advantage that, in addition to facilitating the creation of interactive web applications and control panels directly from R, it can be integrated with other developments, such as, *R Markdown* (Baumer et al., 2014), *ggplot2*, dashboard and other applications based on HTML, CSS and JavaScript. "This sharing of applications is done through the cloud, intranet or the Internet and can even be run on smartphone browsers, which makes them ideal for use inside and outside the classroom" (Jorge, 2018).

THE STRUCTURE OF SHINY

Shiny consists of two components: the user interface (*user interface*) called *ui.R* and the server function (*server*) called *server.R* (Figure 2). The user interface contains the code for the layout and appearance of the application, with default options restricted to appearance. The layouts can be customized and changes to the appearance can be made if the programmer has some knowledge of HTML or CSS. For standard applications, simple commands are sufficient and knowledge of HTML or CSS language is not required for adjustments. The server role hosts all the code that drives the application's functionality and can use all built-in programs available to R and RStudio users. A quick look at Shiny can be done in a few steps, just install the Shiny package in R (instruction: install.packages ("shiny"), library (shiny)) Table 1 summarizes a list of shortcuts with eleven examples in Shiny.

Most of the programmed code is in the *user* interface (ui.R file*)*. As an observation, to obtain the different applications it is necessary to install a series of integrated packages for the generation of statistical and graphic tools. These are: *shiny* (main package for the generation of web applications)*, shinythemes* (for the generation of styles in shiny)*, DT* (provides an R interface to the DataTables JavaScript library)*, ggplot2* (for advanced plotting)*, nortest* (for normality testing)*, tseries* (for time series analysis)*, RcmdrMisc* (statistical analysis several)*, lmtest* (test for linear regression models)*, readr* (for reading different types of data files, for example: XLSX, CSV, TXT, among others)*. The application is fully installed on the rsudio.github.io* servers, making it possible to migrate according to the center's requirements. The place where the outputs are defined is in the *server*, file server.R. Grolemund (2015) adds:

> Shiny creates a reactive context between the two objects known as a *call-back*, which is a command to re-evaluate the observer. Multiple observers can be bound to the same reactive value and the server will queue all *call-backs* and execute each *call-back* in case of a flush (p.26).

DEVELOPMENT OF THE VIRTUAL LABORATORY

The work was divided into five stages:

Stage 1: programming the style file in CSS language for the visual environment of the virtual tutor. Percentage of total development: 13%.

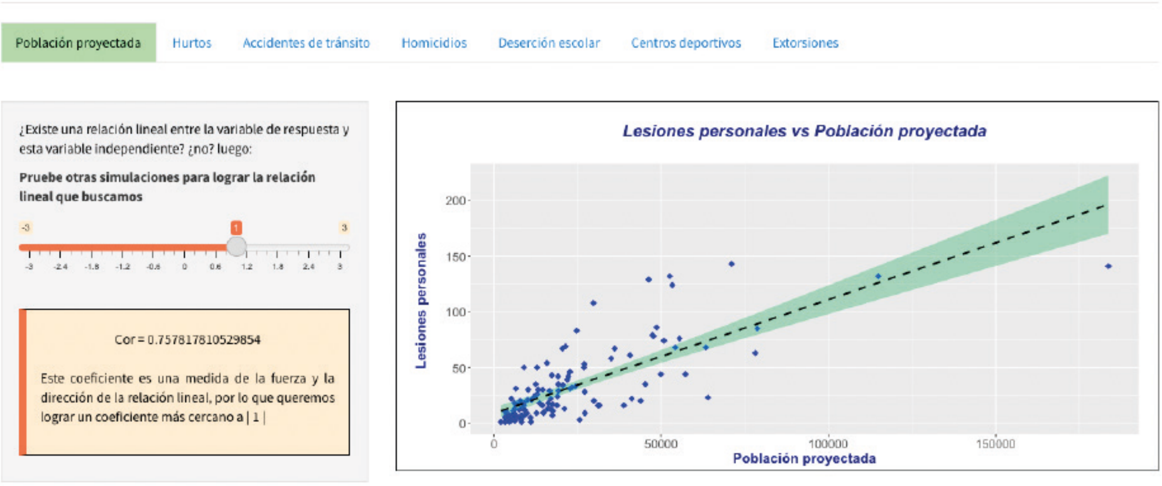
Stage 2: programming of the global file in R language to control the database, filters for managing it, structure of conditions and generation of links between the data. Total development percentage: 33%.

Stage 3: programming of the *users interface* in R language using different dedicated libraries, that is, the presentation and structure of the virtual tutor. Total development percentage: 58%.

Some characteristics of the programming carried out at this stage through the *user interface* are the following:

- Implementation of *fluidPage()* which is the function that creates the *user interface*, that is, the navigation page.

- The application was programmed with navigation through the options menu in mind. *TabsetPanel* was used with *type = "tabs"*. Within this, each call to *tabPanel ()* specified the layout within each tab. In this app, the layout of the top bar changes depending on the tab, and for this reason it is also defined *within tabPanel*. *selectInput ()* that indicates a different element to choose.



Shiny example (executable code in R)	Functionality
runExample("01_hello")	histogram
runExample("02_text")	Tables and data charts
runExample("03_reactivity")	reactive expression
runExample("04_mpg")	global variables
runExample("05_sliders")	slider bars
runExample("06_tabsets")	Windows with partitions
runExample("07_widgets")	Help text with submit button
runExample("08_html")	Shiny app built in HTML
runExample("09_upload")	File upload wizard
runExample("10_download")	File download wizard
runExample("11_timer")	automated timer

Figure 1. Window created with Shiny (part of the virtual lab project)

Table 1: Examples of Shiny applications and their functionalities

- After defining the sidebar, R was told what would be displayed in the app body using mainPanel (showOutput ("lines", "highcharts")), where mainPanel() indicates that the app body is present and showOutput that will show an output defined in the server.

- Programming elements in HTML and CSS language were used.

Stage 4: programming the *server* in R language to display graphical and analytical results of the binding actions of the *users interface* and the simulators. Total percentage of development: 83%.

server programming:

- Started with *output $histogram* renderChart2 (), which indicates that we are working with the histogram *output* and will create a chart (*renderChart2 which creates rCharts* objects).

- Presentation of other available charts, for example: boxplot and quantile-quantile chart. Programming elements in HTML and CSS language were used.

Stage 5: *user interface* and *server* programming that simulates the interaction with artificial intelligence. Total development percentage: 100%.

RESULTS

The virtual laboratory is an environment that allows students and professors to carry out statistical calculations, data analysis, graph creation and visualization in general, exploring different real or simulated statistical scenarios, all through interaction with tabs available in the application. Each tab allows access to different modeling steps, which are the following:

- Home tab: Includes the presentation of the tutor and the database available for learning (initial version). It is represented by the icon of a "house" (Figure 3), this option allows you to return to the tutor's presentation screen. The upper part allows navigation to any of the modeling steps or go to the evaluation and available glossary. At the bottom of this tab it is the database available to perform the linear regression model. In future implementations, the option of loading, editing and downloading databases are considered.

- Step 1: Graphic Test and Analytical Test. It allows to generate simulations (Figure 4). A brief explanation of the fundamentals and theoretical aspects are detailed in the first part. In the graphic test, a simulator was programmed on the left part of the window to control the values of the random errors defined by the model, supported by a corresponding graph. Three graphs allow the verification of the assumption of normality of the response variable: a histogram, a box plot and a quantile-quantile graph. The slider available for simulations generates transformed graphs. The user should geometrically achieve the normality of the response variable. This window allows you to navigate to some other window available on the web with fundamentals and theory related to this step. The analytical tests available are Shapiro-Wilk, Anderson- Darling, Cramer-von Mises, Kolmogorov-Smirnov and Jarque-Bera. Once the test is selected, a statistical summary of the same is presented in the window, in addition to the conclusion regarding the rejection or not of the statistical hypothesis.

- Step 2: Exploratory analysis.

Scatterplots were programmed according to the prediction variables available in the database. It is possible to run simulations to answer the question "Is there a linear

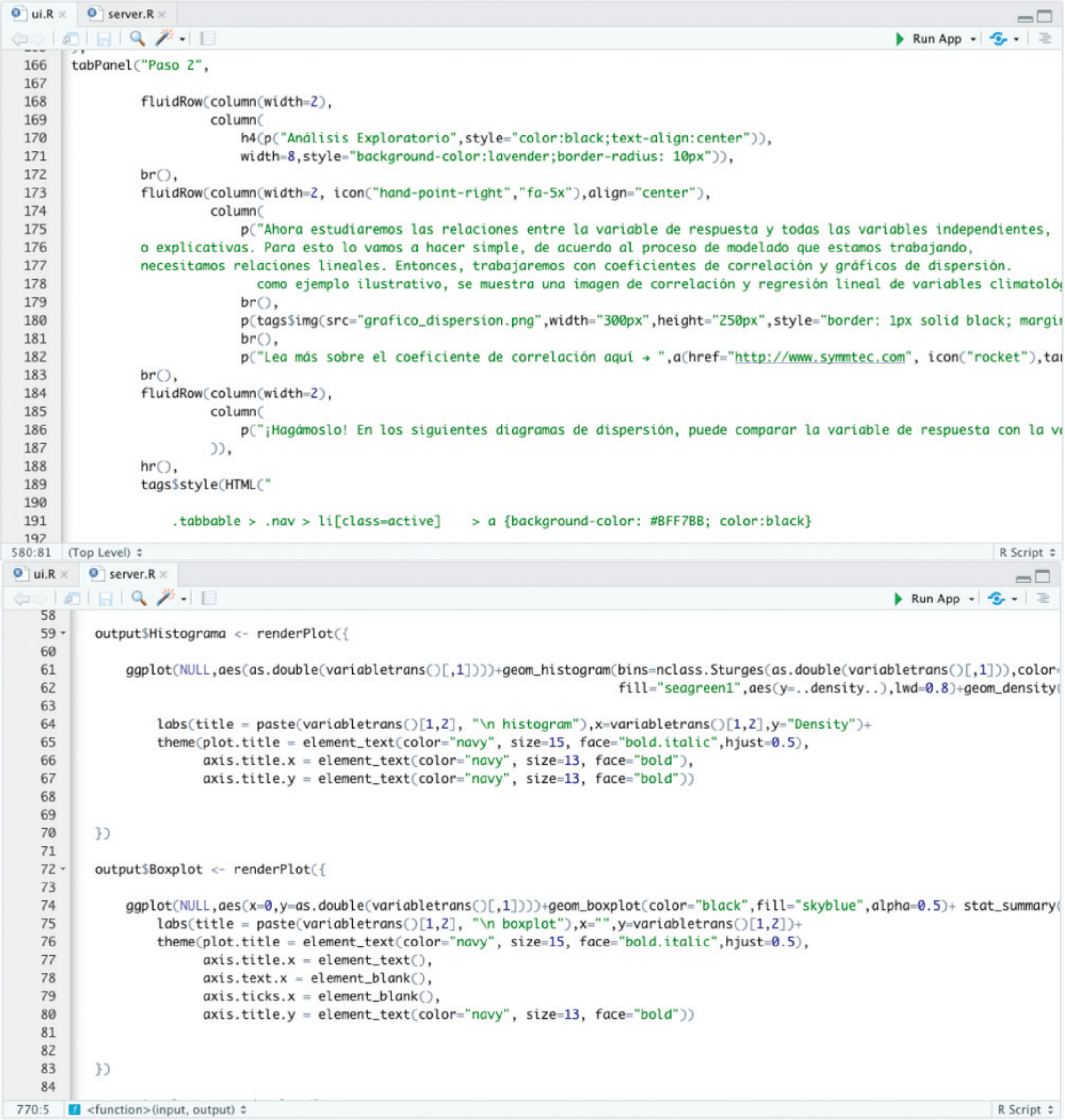


Figure 2: Above: Part of the programming window of the *user interface* (ui.R). Below: Part of the *server* programming window (server.R). In both images some lines of programming, elements of the HTML and CSS language stand out.

Figura 4: Menú de navegación en el tutor virtual para el aprendizaje de la estadística

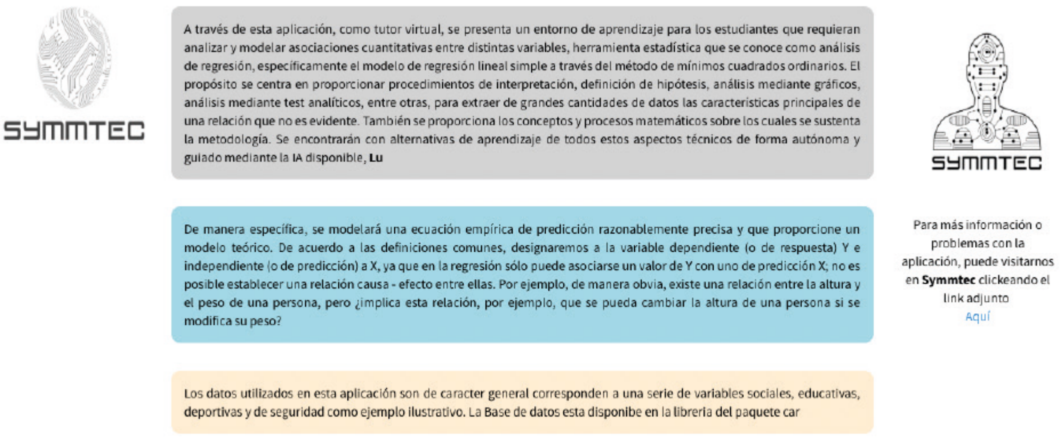


Figure 3: Presentation of the virtual tutor for learning statistics. "Home" tab. At the top, there are quick access tabs to the home zones, Step 1 (Normality Test), Step 2 (Exploratory Analysis), Step 3 (Multicollinearity Analysis), Step 4 (Final Model), Step 5 (Assumptions for model residuals), Pencil (unit evaluation and glossary)

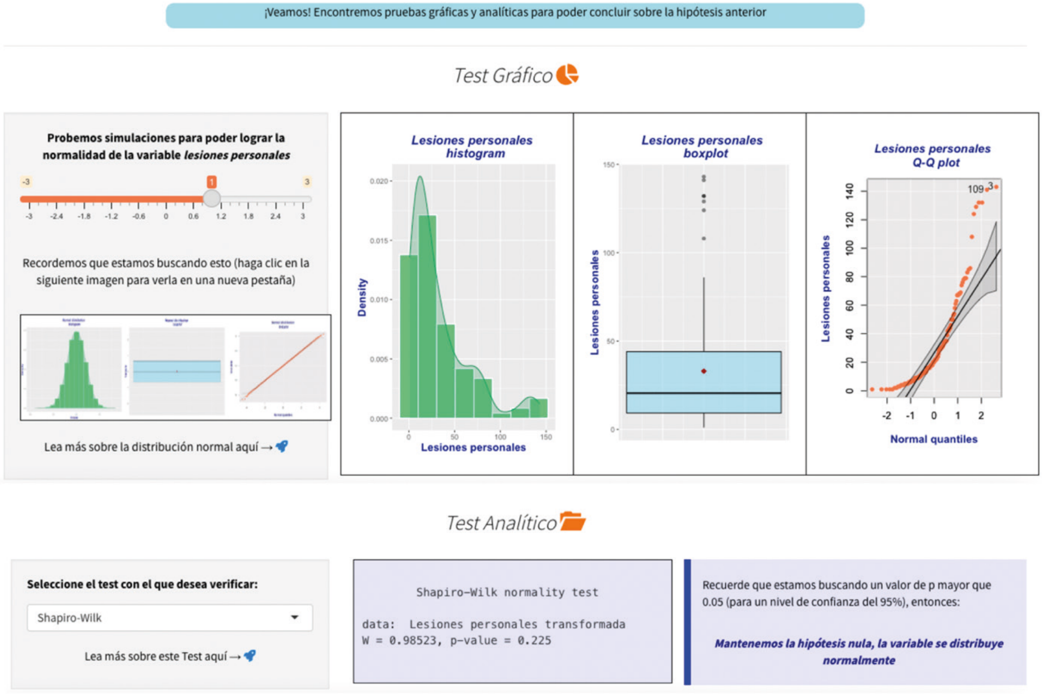


Figure 4: Graphic test of Step 1 (upper figure): Normality. The graphs shown are associated with a value of 1. Analytical test of Step 1 (bottom figure): Normality. The Shapiro-Wilk Test is presented for the simulated situation of graphic normality shown in the previous image. The statistical summary and the maintenance of the null hypothesis are presented, and with it the conclusion that the values distribute normally.

Journal of Engineering Research ISSN 2764-1317 DOI 10.22533/at.ed.3173262301083

relationship between the response variable and this independent variable?" that includes analytical test. In this step of the application, the relationship between the response variable and all the independent variables (or predictor or explanatory variables) is studied, looking for a linear relationship according to the model. Correlation coefficients and scatter plots are presented as statistical elements. For the user, as an illustrative example, an image of correlation and linear regression of climatological variables is presented for the design of ecotechnologies and bioclimatological architecture. The corresponding mathematical equation is also presented. As additional options, a simulator was programmed that allows adjusting the correlation coefficients and associated dispersion graph.

- Step 3: Multicollinearity. The variance inflation factor (VIF) was added to verify the presence or absence of multicollinearity (correlated predictor variables) in the multiple linear regression model. This window allows the selection of the different variables of the model to determine the existence of multicollinearity by means of the variance inflation factors and a summary of the model for each selection.

- Step 4: Final model (Figure 5). Presentation of the final model mathematically, statistically and analytically. In this step the final model is built. For this, you must select the independent variables that you want to include in the model and especially select for which of them you want to include some non-linearity in the model, the above related to the previous step. In this step, it is very important to achieve a high fitted coefficient and also to make the model parameters statistically significant. All previous model building steps are linked in this application window.

- Step 5: Assumptions of the residuals of the model.

In addition to the assumptions discussed in the previous steps, the final model has three other assumptions associated with the residuals (difference between the observed value of the model's dependent variable and the expected value calculated using the linear regression equation). These are: Normality, Homoscedasticity, and no auto-correlation (Figure 6). The analyzes of the three assumptions were programmed in independent tabs with graphic test and analytical test.

- Evaluation and glossary. The application incorporated an evaluation for the user, according to the knowledge acquired in three levels. The glossary allows reviewing the theoretical contents of the different statistical tools used.

DISCUSSION OF RESULTS

Shiny package of the R software allows the creation of a web application with a virtual laboratory function for the process of learning and teaching statistics. The fulfillment of the specific objective of the implementation of a prototype of a graphical environment and virtual laboratory operation in stages was 100% fulfilled. As observations regarding the results by stages at a general level, it is observed that the Shiny package has the necessary tools for the complete programming of the interface. Functions are available for the creation of divisions (windows) on the interface with certain limitations to the quantity and disposition of these, but sufficient for the objective of this project. Additional packages such as: *shinythemes* (for generating shiny styles), *DT* (provides an R

Journal of Engineering Research ISSN 2764-1317 DOI 10.22533/at.ed.3173262301083

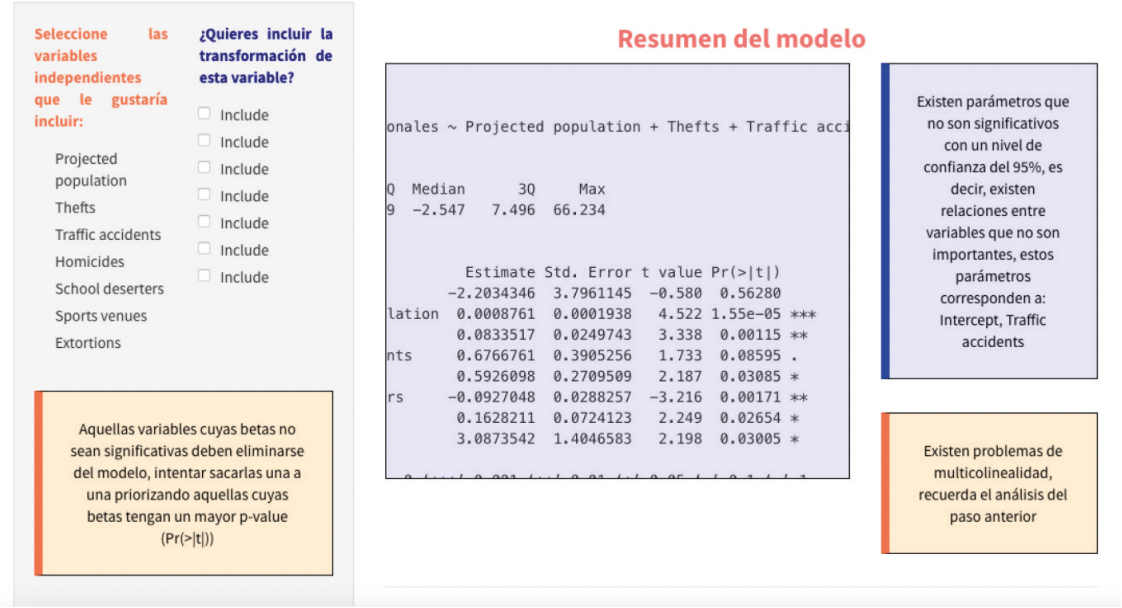


Figure 5: In step 4: Final model, a statistical summary of the variables and their selected transformations is presented. Along with the above, informative windows are presented regarding the significance of the beta variables of the model as well as the level of significance or expected confidence.

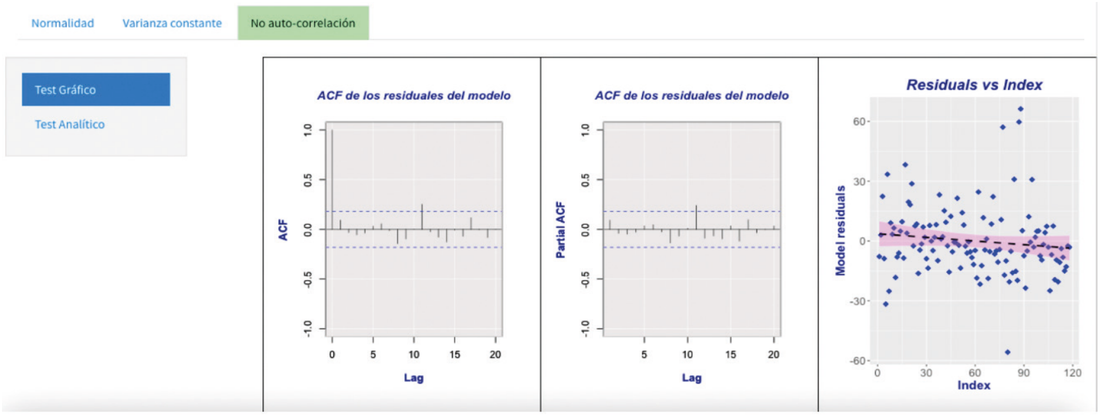


Figure 6: Window of auto-correlation graphical test for the residuals.

interface to the DataTables JavaScript library), *ggplot2* (for advanced plot generation), *nortest* (for normality testing), *tseries* (for time series analysis), *RcmdrMisc* (various statistical analyses) and *lmtest* (test for linear regression models) worked without problems. There are limitations associated with data import, which must be addressed in subsequent versions. Regarding the graphic outputs, these meet the basic requirements, although there are observations from the aesthetic. The size of the areas destined to the computational output has limitations and cases were observed where the output is not observed at 100% or the font is too small. The graphs are of good size, as are the outputs that include bounded statistics (for example, a correlation report). The sliding bars fulfill the function of providing immediate simulation tools, this is useful for the explanation of compliance or non-compliance with the assumptions of linear regression and the consequences of not meeting these assumptions. The residual analysis module allows to clearly observe the quality of the fit of the regression line (in the case of simple linear regression) and allows to study elements such as the assumptions about them and even expand content regarding the effects of outlier data on the line of regression. The operational module for the challenge modality (questions section) is at a basic level. Although it presents three levels of difficulty with a feedback system, the guiding artificial intelligence system must be implemented for the recommendation of problems adapted to the difficulties of the respondent. Limitations of the Shiny package are observed for the implementation of these solutions.

CONCLUSIONS AND FUTURE DEVELOPMENTS

Shiny is a powerful tool for creating teaching technology tools. The interfaces created are friendly and facilitate the application of the statistical contents for the users, although, from the point of view of the creation of the modules, it can be complex for those who do not have programming knowledge in the R language. From the technical point of view , the construction of the virtual laboratory based on this web application meets the objectives of the study in 5 stages: 1. Programming of the style file in CSS language for the visual environment of the virtual tutor, 2. Programming of the global file in R language for the control of the database (filters for managing it, structure of conditions and generation of links between the data), 3. Programming of the *users interface* in R language using different dedicated libraries, 4. Programming of the *server* in R language to display results graphics and analytics of the binding actions of the *users interface* and the simulators and, 5. Programming of the *user interface* and *server* independent of the previous ones in R language for the added value proposition that simulates the interaction with artificial intelligence. From the construction experience, slight difficulties were observed with the programming of Shiny's ui.R and server.R due to some incompatibilities of R packages with the application, as well as difficulties related to code entry. The virtual laboratory prototype processes the 5 steps associated with linear regression defined in this project: presents normality tests for the variable to be predicted, presents exploratory data analysis (correlation and scatter plots), presents multicollinearity analysis, presents final model including tests associated with the significance of the predictors and the model and presents residual analysis. The virtual laboratory prototype meets the functional prototype requirements. Although most of the functions defined in the specific objectives are fulfilled, there are certain limitations of the prototype that will be considered for future developments. For example, the prototype has a unique database available for the learning linear regression through, however, the possibility of uploading, editing and/or downloading your own databases data for linear regression modeling could be very useful. Several of the tabs allow you to navigate to other web applications that support the fundamentals of each step of the linear regression modeling process this would allow have a reservoir of information stored in another self-developed application. The available glossary in the application could store such information, although it requires a construction of said information. The glossary could become a repository of content, exercise guides, illustrative examples, or data tables that can be used to develop modeling of a linear regression by using the proposed application. As a general conclusion, the virtual statistics laboratory is a platform that provides an interactive and practical learning experience in statistics that allows teachers and students to apply concepts related to data description, data visualization, statistical tests and adjustment of regression models, linear in simulated or real data set.

As future research based on the use of this virtual laboratory prototype (and even considering future updates of this prototype) the following needs will be considered: 1. Study whether this tool facilitates the transition for teachers from theoretical concepts related to statistics and Machine Learning to the application of these in real data sets. 2. Study if this tool facilitates the formation of capacities for the development of solutions based on Machine Learning in students of various disciplines. 3. Determine if this tool facilitates the transmission of statistical content if it is used in different virtual, hybrid or face-to-face modalities. 4. Explore the simplification of programming work provided by the ChatGPT tool (OpenAI, 2023) for creating Shiny interface or interface elements in Shiny by installing and using the *GPTstudio package* (Nivard, Wade, Calderon, 2023). As a complement to this point, Erickson (2023) presents a discussion on the effectiveness of the use of *GPTstudio* for the creation of quality R code and Lubiana et al. (2023) present a series of tips for the programmer derived from the application of ChatGPT and the *GPTstudio package* in computational biology, which could be extended to any discipline.

ACKNOWLEDGMENT

The present text was born within the framework of the UBO/VVCMEI-12 Project awarded in the R+D-based Innovation contest 2021-2022 of the Vice-Rector for Research with the Environment and Research of the Bernardo O'Higgins University, Chile.

Acknowledgments to the Interdisciplinary Engineering Nucleus of the Faculty of Engineering, Science and Technology of the Bernardo O'Higgins University (NII - UBO).

REFERENCES

Almerich, G., Suárez-Rodríguez, J., Belloch, C., & Bo, R. M. (2011). Las necesidades formativas del profesorado en TIC: perfiles formativos y elementos de complejidad. *RELIEVE-Revista Electrónica de Investigación y Evaluación Educativa*, 17(2).

Arnholt, A. T. (2019). Using a Shiny app to teach the concept of power. *Teaching Statistics*, *41*(3), 79-84.

Baumer, B., Cetinkaya-Rundel, M., Bray, A., Loi, L., & Horton, N. J. (2014). R Markdown: Integrating a reproducible analysis tool into introductory statistics. *arXiv preprin*t arXiv:1402.1894.

Berg, A. (2021). Bayesian explorations with dice. *Teaching Statistics*, 43(3), 114-123.

Carrillo, J. A. O., & Maldonado, Á. O. (2009). La comunicación didáctica digital en el espacio europeo de educación superior. *ICONO 14, Revista de comunicación y tecnologías emergentes*, 7(3), 64-83.

Chance, B., & Rossman, A. (2006). Using simulation to teach and learn statistics. In *Proceedings of the Seventh International Conference on Teaching Statistics* (p. 1-6). Voorburg, The Netherlands: International Statistical Institute.

Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2017). Shiny: web application framework for R. *R package version*, 1(5), 2017.

Das, K. (2019). Role of ICT for Better Mathematics Teaching. *Shanlax International Journal of Education*, 7(4), 19-28.

Depaoli, S., Winter, S. D., & Visser, M. (2020). The importance of prior sensitivity analysis in Bayesian statistics: demonstrations using an interactive Shiny App. *Frontiers in psychology*, 11, 1-18.

Erickson, J. S. (2023). *RPIrates: Fun with OpenAI, GPTStudio and R!* https://dspace.rpi.edu/handle/20.500.13015/6516

Ferro Soto, C., Martínez Senra, A. I., & Otero Neira, M. C. (2009). Ventajas del uso de las TICs en el proceso de enseñanza-aprendizaje desde la óptica de los docentes universitarios españoles. *Edutec: Revista electrónica de tecnología educativa*, 29, 1-12.

Gámiz Sánchez, V. M. (2009). *Entornos virtuales para la formación práctica de estudiantes de educación: implementación, experimentación y evaluación de la plataforma aulaweb* [Doctoral Dissertation, University of Granada]. https://digibug.ugr.es/handle/10481/2727

Garfield, J. & Ben-Zvi, D. (2008). *Developing Students' Statistical Reasoning: Connecting Research and Teaching Practice*. Kluwer Academic Publishers.

Gopinath, P. P., Parsad, R., Joseph, B., & Adarsh, V. S. (2021). grapesAgri1: collection of shiny apps for data analysis in agriculture. *Journal of Open Source Software*, 6(63), 3437

Grolemund, G. (2015). *Shiny-How to understand reactivity in R*. https://shiny.rstudio. com/articles/understanding-reactivity.html.

Hagtvedt, R., Jones, G. T., & Jones, K. (2007). Pedagogical simulation of sampling distributions and the central limit theorem. *Teaching Statistics*, 29(3), 94-97.

Harraway (2012). Learning Statistics Using Motivational Videos, Real Data and Free Software, *Technology Innovations in Statistics Education*, 6(1). https://doi.org/10.5070/T561000186.

Johnson, O., Fronterre, C., Diggle, P.J., Amoah, B. & Giorgi, E. (2021) MBGapp: A Shiny application for teaching model-based geostatistics to population health scientists. *PLoS ONE*, 16(12): e0262145. https://doi.org/10.1371/journal.pone.0262145

Jorge, N. I. C. (2018). Simulações e Visualizações No Ensino das Probabilidades e Estatística Uma Aplicação Shiny [Doctoral Dissertation, University of Beira Interior (Portugal)].

Kasprzak, P., Mitchell, L., Kravchuk, O., & Timmins, A. (2020). Six Years of Shiny in Research-Collaborative Development of Web Tools in R. *R Journal*, 12(2).

Lubiana, T., Lopes, R., Medeiros, P., Silva, J. C., Goncalves, A. N. A., Maracaja-Coutinho, V., & Nakaya, H. I. (2023). Ten Quick Tips for Harnessing the Power of ChatGPT/GPT-4 in Computational Biology. *arXiv* preprint arXiv:2303.16429.

Lunsford, M. L., Rowell, G. H., & Goodson-Espy, T. (2006). Classroom research: Assessment of student understanding of sampling distributions of means and the central limit theorem in post-calculus probability and statistics classes. *Journal of Statistics Education*, 14(3).

Miranda Freire, S. (2019). Using Shiny to illustrate the probability density function concept. *Teaching Statistics*, 41(1), 30-35.

Moore, T. N., Thomas, R. Q., Woelmer, W. M., & Carey, C. C. (2022). Integrating Ecological Forecasting into Undergraduate Ecology Curricula with an R Shiny Application-Based Teaching Module. *Forecasting*, 4(3), 604-633.

Nivard, M., Wade, J., Calderon, S. (2023). *gptstudio: Use Large Language Models Directly in your Development Environment*. https://github.com/MichelNivard/gptstudio.

OpenAI. (2023). ChatGPT (Mar 14 version).

Potter, G., Wong, J., Alcaraz, I., & Chi, P. (2016). Web application teaching tools for statistics using R and shiny. *Technology Innovations in Statistics Education*, 9(1).

R Core Team (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

Satyahadewi, N., & Perdana, H. (2021, May). Web Application Development for Inferential Statistics using R Shiny. In *1st International Conference on Mathematics and Mathematics Education (ICMMEd 2020)* (p. 425-429). Atlantis Press.

Stratton, C., Green, J. L., & Hoegh, A. (2021). Not just normal: Exploring power with Shiny apps. *Technology Innovations in Statistics Education*, 13(1).

Toledo, Á. & Vicencio, I. (2021). El uso de TIC para el aprendizaje de la Estadística: un estudio en universidades chilenas. *Educação e ensino na era da informação*, 40-58.

Toledo, Á. & Vicencio, I. (2022a). Una revisión de aplicaciones interactivas para explicar conceptos estadísticos utilizando el paquete shiny del software R. In *CUICIID 2022: congreso Internacional sobre Comunicación, Innovación, Investigación y Docencia. Libro de actas* (p. 596). Fórum Internacional de Comunicación y Relaciones Públicas (Fórum XXI).

Toledo, Á. & Vicencio, I. (2022b). Visualización de datos mediante instrucciones simplificadas en librerías basadas en GGPLOT2 de R (software). In *Tic Docentes* (p. 435-446). Thomson Reuters Aranzadi.

Toledo, Á. & Vicencio, I. (2023). *Aplicaciones interactivas basadas en el paquete SHINY/R para explicar conceptos estadísticos*. Human Review. International Humanities Review / Revista Internacional de Humanidades, 17(4), 1-15.

Varma, J. R., & Virmani, V. (2017). Shiny alternative for Finance in the Classroom. *Indian Institute of Management*, India.

Von Borries, G. F., & De Castro Quadros, A. VV. (2022). *ROC App: an application to understand roc curves*. Brazilian Journal of Biometrics, 40(2).

Wickham, H. (2011). ggplot2. Wiley Interdisciplinary Reviews: *Computational Statistics*, 3(2), 180-185. https://doi.org/10.1002/wics.147.

Wilkinson, L. (2005). *The Grammar of Graphics (2nd edition)*. Springer-Verlag.

Williams, I. J., & Williams, K. K. (2018). Using an R shiny to enhance the learning experience of confidence intervals. *Teaching Statistics*, 40(1), 24-28.

Wojciechowski, J., Hopkins, A. M., & Upton, R. N. (2015). Interactive pharmacometric applications using R and the shiny package. *CPT: pharmacometrics & systems pharmacology*, 4(3), 146-159.

Zieffler, A., Park, J., Garfield, J., delMas, R. & Bjornsdottir, A. (2012). The Statistics. Teaching Inventory: A Survey on Statistics Teachers' Classroom Practices and Beliefs. *Journal of Statistics Education*, 20(1). https://doi.org/10.1080/10691898.2012.11889632.