INTEGRATING R AND PYTHON INTO AN APPLIED ECONOMETRICS COURSE

Lucía Carolina Varela
Universidad Abierta Interamericana, Argentina
luciacarolina.varela@uai.edu.ar

The digital transformation and the rapidly changing demands for technology skills represent challenges that traditional higher education faces today. However, due to Argentinian universities' oversight, undergraduate students do not develop coding competencies to perform complex statistical analyses of datasets. This study considers an Information and Communication Technology approach that sought to integrate R and Python into an Econometrics course by merging theoretical concepts with applied aspects in a backward design course while blending technology and multimedia resources with active learning in a flipped-format classroom. This empirical research aims to explore to what extent the strategy deployed was a relevant and impactful means for developing students´ coding skills needed to perform econometric analysis. Students´ final empirical assignments show that, by the end of the course, they were able to solve real-world data challenges working with programming languages in R Markdown documents and in the RStudio Integrated Development Environment.

LOOKING BACK. LOOKING FORWARD

The increasing digital transformation and the continuously changing demands for technology skills represent challenges that traditional higher education faces today and in particular Statistics education. However, in Argentinian universities, statistics and econometrics courses focus on developing outdated skills, thus preventing students from fostering Information and Communication Technology (ICT) proficiency and seizing the benefits of technological change in a digital economy. Thereby and despite data and information exploding in today's globalized world, undergraduate students do not acquire the coding competencies needed to perform complex statistical analyses of datasets. Furthermore, enhancing strong computational skills empowers today´s learners by helping optimize their learning time, allowing them to focus on developing higher-order thinking skills, i.e., critical thinking and decision-making, rather than memorizing algorithms based on repetition. Skills that contribute to productivity and enable them to prosper and flourish in society. In this context, central to these concerns is the question of integrating: *"To which degree do we want to integrate developing coding competency with developing skills in understanding data?"*

RESEARCH DESIGN AND IMPLEMENTATION

This paper considers an approach based on ICT that sought to integrate two in-demand programming languages -R and Python- into an applied Econometrics course by merging theoretical concepts with applied aspects in a backward design course (Wiggins & McTighe, 2005) while blending technology and multimedia resources with active learning (Prince, 2004), and a flipped classroom methodology (Bergmann & Sams, 2012). Regarding the concerns stated in the introduction, the purpose behind this teaching strategy was twofold. First, the course needed to be suitable for students who did not possess coding skills, but concurrently, it needed to be instructive, providing economics students with the necessary knowledge to tackle econometric problems. This empirical study aims to explore to what extent this strategy was a relevant and impactful means for developing students´ coding skills needed to perform econometric analysis.

Strategy, setting and participants

The course under study consists of a 16-week face-to-face undergraduate-level Econometrics course at an Argentinian university that covers topics such as multivariate regression analysis, multicollinearity, autocorrelation, heteroscedasticity, probit/logit regression, simultaneous-equation models, and basic time series methods. This course is required for undergraduate degrees in Economics. Moreover, all participants handled Excel skills but, except for one student, had not been exposed to any programming language.

In: EM Jones (Ed.), *Fostering Learning of Statistics and Data Science*
Proceedings of the Satellite conference of the International Association for Statistical Education (IASE),
July 2023, Toronto, Canada. https://iase-web.org

Methodology

This approach was designed with the interplay between theory, application, and computing at its core. As stated above, the methodology deployed aimed to bridge the gap between the university and the professional world by merging theoretical concepts with applied aspects in a backward design course (Wiggins & McTighe, 2005), blending technology and multimedia resources with active learning (Prince, 2004) in a flipped classroom environment (Bergmann & Sams, 2012).

In regards to computing, R was the primary programming language used throughout the course because of its ability to perform statistical analysis, while Python was essentially introduced to obtain, analyze, and wrangle data. RStudio was treated as the default Integrated Development Environment (IDE), considering its several advantages. It is available in open source, provides a graphical user interface that makes R much easier for beginners, plus its tab autocompletion functionality, and allows students to execute Python and R code in R Markdown documents. Furthermore, the reticulate package enables users to work seamlessly with R and Python by automatically or manually converting Python objects to R objects and vice versa.

Within this framework, topics covered throughout the course were adapted to a technology-enabled form of instruction. Hence, in addition to in-class instruction, all course material, multimedia resources, assignments, and assessment reports were made available to students through the University's virtual platform. These online learning resources, coupled with active student activities in the classroom (Prince, 2004), promoted a flipped-course environment (Bergmann & Sams, 2012). These various elements are set out below.

Multimedia resources

The online course content designed for each class comprised PowerPoint presentations with voice-over (video format) to approach the econometrics´ theoretical core and screencasts with audio narration (video format) in the RStudio IDE in R Markdown documents for applications, seeking a seamless transition between theory and practice while developing students´ coding competency. Essentially, screencasts in the RStudio IDE were thought to concisely but rigorously show students how to implement the previously explained theoretical frameworks in practice in a technology-rich environment. Thereby, these digital videos presented them with several exercises and activities for each topic covered in the course, showing step-by-step the econometric methods required for the solutions and providing students with explained codes. These activities were designed to integrate R and Python codes with the contents of the textbook Basic Econometrics by Gujarati & Porter (2010), which served as a basis for the course. Figures 1 and 2 present screenshots of exercises solved in detail step-by-step in multimedia presentations placed online on the University's virtual platform, providing thus an out-of-class opportunity for students to practice problem-solving.

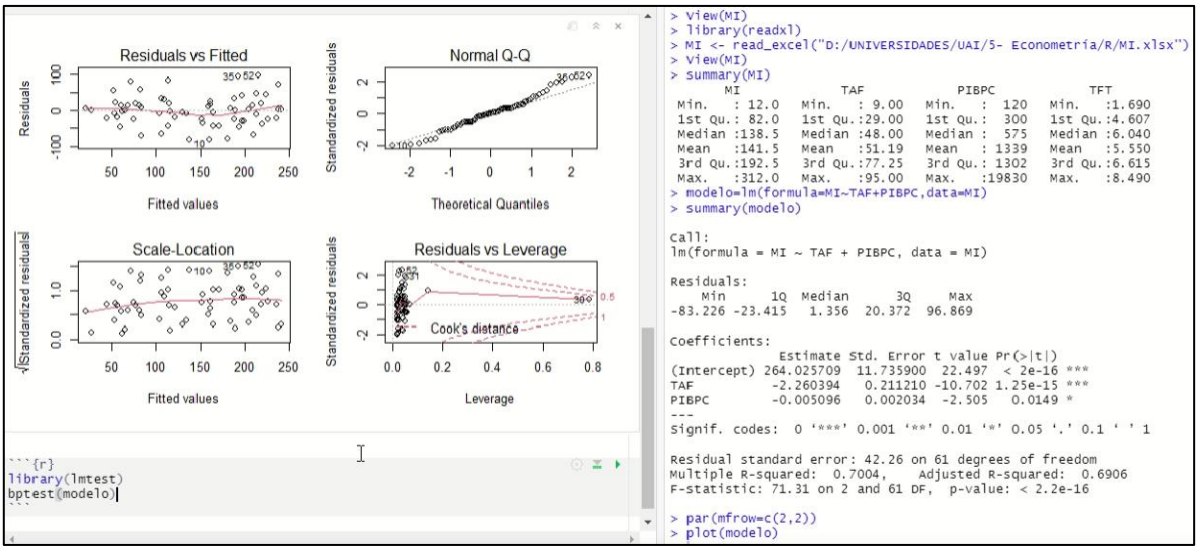

Figure 1. Screenshot: Testing for heteroscedasticity in regression analysis with R language in RStudio, importing data from Excel into R

Figure 2. Screenshot: Obtaining, analyzing, and wrangling financial data with Python language, converting later the Python object to an R object with the reticulate package in the RStudio IDE, and seamlessly continuing with the application of econometric methods using R language

Course design: assignments, statistical indicators, and learning outcomes

The course was designed from the learning outcomes (Wiggins & McTighe, 2005), conceiving thirteen computer-based assignments to foster students´ problem-solving skills in technology-rich environments. These hands-on learning activities included one real-world application (final empirical assignment) in which students were encouraged to develop new research by implementing the techniques introduced by way of problem sets (twelve assignments) with their own real data. Therefore, all assignments focused on active learning situations and scenarios (Prince, 2004), and followed an increasing difficulty level. Figure 3 specifies a classification according to Bloom´s revised taxonomy (Anderson & Krathwohl, 2001), with the indicators constructed to assess students´ academic progression in each underlying type of skill.

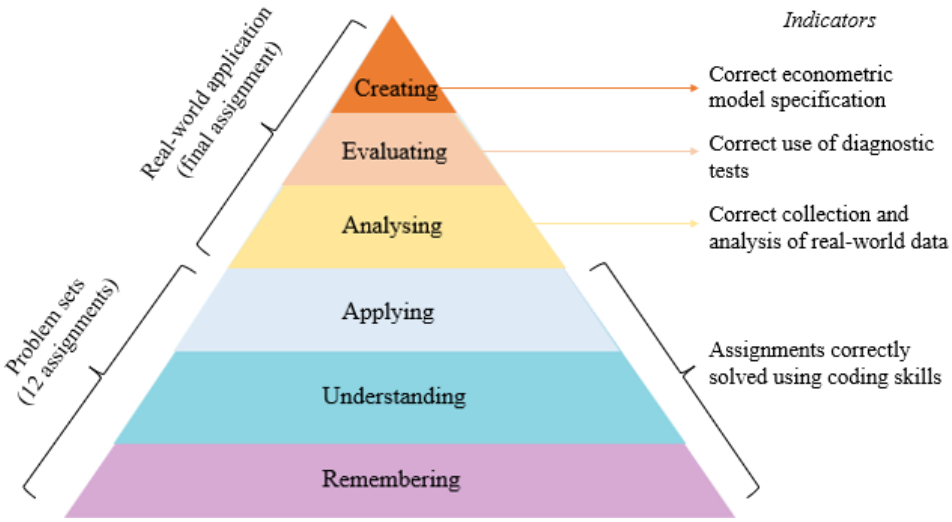

Figure 3. Adapted Bloom´s Revised Taxonomy with built indicators

RESULTS

Data collection and analysis

To explore to what extent the strategy deployed was a relevant and impactful means for developing students´ coding skills needed to perform econometric analysis, data was collected from

students' learning activities (cf. figure 3). As indicated above, these assignments had to be solved individually and using computational methods (i.e., programming languages). Students had to submit their solutions on a weekly basis, except for the final empirical assignment.

The student population for this study was relatively uniform. The course was composed of individuals pursuing an undergraduate degree in Economics who had successfully completed the Statistics course -a necessary foundation and enforced requisite for the Econometrics course- in which they got hands-on experience using statistical functions, tools and Add-Ins in Excel spreadsheets. Given this relative homogeneity in their educational background, for the analysis and aiming to identify patterns that will deepen the findings of this study, participants were clustered based on their competency level in coding skills at the beginning of the course.

Group 1

Students handled Excel skills but had not been exposed to any programming language and did not attend any additional coding courses. They correctly solved weekly problem sets with coding skills. Regarding their final assignment, they achieved a coding competency level that enabled them to implement the techniques introduced by way of problem sets (cf. figure 1) with their own data (imported from Excel into R) to a real-world application. They were able to build econometric models and performed a regression analysis with appropriate diagnostic and specification tests working with R in an R Markdown document in the RStudio IDE.

Group 2

In the course under study, only one student had a minimum baseline of programming knowledge with Python language. This student solved weekly problem sets with coding skills. Regarding his final assignment, he achieved a coding competency level that enabled him to work with R and Python. He correctly converted Python objects to R objects with the reticulate package and seamlessly continued the econometric analysis using R language (cf. figure 2). He worked in R Markdown documents in the RStudio IDE and correctly interpreted R´s outputs from the theoretical structure that supported his empirical work.

FINDINGS, ISSUES AND CHALLENGES

A number of findings emerge from the previous analysis. First, final empirical assignments from all students show that, by the end of the course, they were able to solve real-world data challenges specifying and diagnostically testing econometric models with programming languages in R Markdown documents in the RStudio IDE. Second, the pre-course competency level of students in coding skills was predictive of performance; in this order, results indicate that Group 1 developed coding competencies, while Group 2 improved it, suggesting that basic coding knowledge would be favorable to efficiently incorporate a second programming language into an applied Econometrics course.

From a theoretical perspective, these findings suggest that the presented approach is suitable for integrating the development of coding competency with the development of econometric skills into an applied Econometrics course in higher education. As regards the implications of the findings for practice, this research shows that the strategy considered had a positive impact on learning-teaching progress, fostering technology-supported teaching as well as enabling undergraduate economics students to gain the competency level in coding skills needed to solve econometric problems.

Beyond these specific findings, this paper raises a significant issue that needs further consideration. For econometricians, the computer is the primary applied tool, and in particular, programming languages have become powerful tools in the construction and analysis of econometric models. Consequently, integrating the development of coding competency with the development of econometric skills in undergraduate economics students becomes a central issue that impacts how easy or how difficult the applied research process will be for future professionals and represents a pressing challenge to Argentinian universities.

REFERENCES

Anderson, L. & Krathwohl, D. (2001). *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. New York: Addison Wesley Longman, Inc.

Bergmann, J. & Sams, A. (2012). *Flip Your Classroom: Reach Every Student in Every Class Every Day*. Washington DC: International Society for Technology in Education.

Bloom, B. (1956). *Taxonomy of educational objectives: Handbook I, The cognitive domain*. New York: David McKay & Co.

Gujarati, D. & Porter, D. (2010). *Basic Econometrics* (5th ed.). New York: McGraw-Hill.

Prince, M. (2004). Does Active Learning Work? A Review of the Research. *Journal of Engineering Education, 93*(3), 223-231. http://dx.doi.org/10.1002/j.2168-9830.2004.tb00809.x

Wiggins, G. & McTighe, J. (2005). *Understanding by design* (2nd ed.). Alexandria, VA: Association for Supervision and Curriculum Development.