Manuscript submitted to
Foundations of Data Science
Volume **X**, Number **0X**, XX **200X**

doi:10.3934/fods.xxxxxxx

pp. **X–XX**

TEACHING DATA SCIENCE TO STUDENTS IN BIOLOGY USING R, RSTUDIO AND LEARNR: ANALYSIS OF THREE YEARS DATA

Guyliann Engels*, Philippe Grosjean

Numerical Ecology Department, Complexys and InforTech Institutes, University of Mons
Avenue du Champ de Mars, 8, 7000 Mons, Belgium

Frederique Artus

Pedagogical Support and Quality Assurance Department, University of Mons
Place du Parc, 20, 700 Mons, Belgium

Abstract. The courses in biostatistics in biology at the University of Mons, Belgium, were completely refactored in 2018 into data science courses include computing tools, version management, reproducible analyses, critical thinking and open data. Flipped classroom approach is used. Students learn with the online material and they apply the concepts on individual and group projects using a preconfigured virtual machine with R and RStudio. Activities (H5P, learnr or Shiny applications) are recorded in a MongoDB database (300,000+ events for 180+ students and 2,000+ GitHub repositories at several trends. (1) There is a relatively long lag period required for the students to get used to the computing environment, the teaching method and the data science in general. (2) Implication is very high, with more than 85% of the students that complete all the activities and get good to excellent assessment. (3) There is a gap between students' own perception of their their progress. (4) During COVID-19 pandemic lockdown, the intensity of the activities largely decreased during two weeks before returning to previous level, but for 3/4 of the students only. The remaining fraction never caught up. We hypothesize that the technical requirements or the lack of motivation during the lockdown were detrimental to roughly one student over ten, despite all the efforts the University deployed to reduce the social fracture. (see http://bds.sciviews.org). The content is expanded beyond statistics to https://github.com/BioDataScience-Course). The analysis of these data reveals skills achievements and their assessment results: they tend to underestimate

1. **Introduction.** In a context where there is an exponentially growing mass of data [18], a reproducibility crisis in Science [2], and a progressive adoption of Open Science practices [3], statistics were broadened to a wider discipline called Data Science. For the Data Science Association, "the Data Science means the scientific study of the creation, validation and transformation of data to create meaning" (http://www.datascienceassn.org/code-of-conduct.html). These changes also led to the emergence of data science programs in universities and higher schools [10, 8]. One example is the Harvard Data Science initiative (https://datascience.harvard.edu/about) started in 2017. With a broader approach,

2020 *Mathematics Subject Classification.* Primary: XXXXX; Secondary: YYYYY..
Key words and phrases. Data Science Teaching, Flipped Classroom, R, RStudio, git.
* Corresponding author: Guyliann Engels.

also comes a broaden public. The data science courses are not just limited to computer scientists, mathematicians or statisticians, but also welcome students in humanities, social sciences, and natural sciences (for instance, the data science training at Duke University [8]). Focus of such courses is for students to develop the ability to deal with real datasets in all their complexities, to be able to conduct reproducible analyses, and to interpret these data in the light of knowledge in their field of expertise.

The data transformation part of the job is a challenge for students with a poor or no background at all in computing. Students that are not used to deal with computer languages enter in a foreign world and have to deal with many exotic concepts, techniques and tools. Version control systems like git, and their internet hosting counterparts like GitHub, Gitlab or Bitbucket also make part of the tools that a data science course teaches and use [12, 15]. Presentation of the results and the use of document formats that dissociate content from presentation, namely LaTeX, Jupyter Notebook, or R Markdown to cite a few, also contribute to the large number of potentially new tools learners have to discover [4]. On the other hand, a student in computing science already masters one or more computing languages, is acquainted with version control systems, with databases and with the way data are manipulated and represented in a computer, but he may have difficulties to grasp the context of datasets related to different disciplines. A student in mathematics or statistics is familiar with various concepts that underpin the techniques used to analyze the data. Students in biology, medicine, psychology, social sciences, economics, … have obviously very different *a priori* knowledge. The gap between knowledge and requirements generates anxiety (see for instance [19]). The course must be organized in a way that learners progress by little steps to avoid exposition to much intimidating concepts and tools at once.

Suitable computer hardware and software environments are required to apply the concepts in the course. Different approaches range from using software accessed from a server (RStudio Cloud (`https://rstudio.cloud/` [TODO: add citation here]), Chromebook data science (`http://jhudatascience.org/chromebookdatascience/`) [TODO: add citation here]) to local installation on the student's computers. The former requires infrastructure to run the software on a server, and that software is only accessible to the students during the course. The later raises problems of license for proprietary software but also installation and configuration issues. An intermediary solution uses preconfigured virtual machines, or containers (e.g. Docker) [7, 5]. Such a solution is the most flexible one because it can be deployed almost anywhere (in the computer lab, at home, on a laptop, …). To apply theoretical concepts through exercises is a key aspect of learning data science [17]. Correct choice of software is critical and exposing students early with the tools they are most susceptible to use later in their work is desirable. This was highlighted by [1] for instance, in the case of ecological data.

These data science courses pose several pedagogical challenges due to the numerous and unfamiliar concepts that must be acquired by a heterogeneous class population. Learning objectives span a large range of cognitive abilities and, in these courses, the intended learning outcomes aim at developing high-level cognitive process abilities such as conceptual, procedural, and even metacognitive knowledge [16]. To meet such learning objectives, active learning methods are useful so that students could better catch up with these high-level cognitive skills [13]. Teaching and learning frameworks turn to a scenario including remote activities to be done

before in-class ones, individual and group problem-solving, peer instructions and ongoing assessment. Indeed, the flipped classroom approach allows students to be active in their learning, which has the benefit of improving student outcomes [13]. Moreover, it allows flexibility and enables students to work at their pace. Their various learning styles are respected as they are actors in their learning process [Spadafora & Zopito, 2018: TODO ref à ajouter]. Such frameworks are open, learning centered, and supported by a varied and rich environment [TODO: cite Bruton et al., 2011].

Recently, data science is also used to analyze the effect of different pedagogical practices on the outcome of these courses thanks to learning analytics [11]. A vast amount of data can be collected on students' activities, and the analysis of these data allows comparing the impact of different pedagogical approaches, or to quantify and document the impact of changes in the courses. [TODO: detail a little bit with one more reference]

At the University of Mons in Belgium, we have started to rework our biostatistics courses in the biology curriculum in 2018. A series of Data Science courses were introduced, both for our undergraduate and graduate students. These courses are inspired from precursor initiatives cited here above. The goal of these courses is to form biological data scientists capable of extracting meaningful information from raw biological data, and to do so in a reproducible way, with the correct application of statistical tools and an adequate critical mind. A preconfigured VirtualBox machine with R, RStudio, R Markdown, git, and a series of R packages preinstalled is used (https://www.sciviews.org/software/svbox/) as a very flexible way to deploy the same software environment both on the university computers and on student's own laptops.

As our course were reworked, we also decided to use flipped classroom and progressive adoption of suitable pedagogical practices with a cyclical approach that consists in stating goals, building pedagogical material with a large emphasis on numerical tools and collection of students' activities, and finally, analysis of the data collected. These results let us regulate our teaching activities the following academic year with refined goals and improved pedagogical techniques. Here, we present the main results spanning on three successive academic years from 2018 to 2021, including two particular periods where distance learning was forced due to Covid-19 pandemic lockdown. In this paper, we will focus on the following four questions:

- Transition from theory to practice is critical and tutorials build with learnr (https://rstudio.github.io/learnr/) are capstones in our courses. What cognitive workload and perceived workload do these tutorials represent for the students?
- Many universities and high-schools resort to exams at the end of an academic term. Do such final exams correctly assess the major learning outcomes that we expect from our data science courses that is, the ability to properly analyse biological data?
- Could we use learning analytics to spot suboptimal learning strategies and discriminate different student profiles in our biological data science courses?
- Did the quick shift from face-to-face to distance learning imposed by Covid-19 lockdown periods affected the production of our students and did it required increased exchanges with the teaching staff to support it?

2. **Methods.** The course material is available online (https://wp.sciviews.org) and is centralized in a Wordpress site. Students have to login with their GitHub account and their academic data are collected from the UMONS Moodle server (https://moodle.umons.ac.be). The courses are broken down into modules that amount roughly to 15h of work each. There are two sessions of 2h and 4h in-class (outside lockdown periods, of course), with roughly 3h of preparation at home before each session, and 3h of work to complete one module. Main activities in the class are analyzing actual data (projects), answering student questions, and lecturing briefly (1/4 h) on selected topics. Students propose and vote for the topics to be covered during these short lectures. Finally, we encourage students to help each other and to explain what they understand to their colleagues. Indeed, students' questions may be redirected by the teacher to other students that have already mastered the topic. Teachers rarely answer questions directly. When it is possible, they rather propose new tracks or ideas to investigate and help learners to find the solution by themselves. Students who go through the activities before the others are encouraged to help their colleagues too.

Regarding the timing, one module is taught every second week so that students have enough time to prepare the material at home before in-class session, and after it, to finalize their projects. As a term is made of 14 weeks, we do not teach more than six modules in a course unit to avoid compacting them to much in time. After reading the theory, students are exposed to exercises of four increasing levels of difficulty. They have thus to apply the concepts repeatedly but in different contexts, which breaks monotony and maintains a stimulating rhythm all along their progression. Once they have learn the principles in the book and self-assessed their comprehension of the concepts using H5P (https://h5p.org) exercises (level 1 difficulty), they have to get used to the software environment. Learnr tutorials (https://rstudio.github.io/learnr/, level 2) are used to gently introduce them to the R code required for the analyses by guiding them through their first data analysis. These tutorials are thus the entry point to the practice. Projects, first individual (level 3), then in groups of two to four students (level 4) represent the core activities. Evaluating these projects constitute, thus, the most important information to assess the competences of our students.

All students' activities in H5P exercises (self-assessing), and in learnr tutorials (transitioning smoothly from theory to practice) are recorded in a MongoDB database. The learnitdown R package (https://www.sciviews.org/learnitdown/) provides the code required to manage user login, user identification and activity tracking for this interactive material.

Projects containing the data, the analyses and the reports are hosted in GitHub repositories. These repositories are cloned and edited by the students in their virtual machines (SciViews Box) with RStudio (https://www.rstudio.com/products/rstudio/), either on their laptops or on the computers in the lab. We encourage our students to install the virtual machine for the course on their computer so that they can work comfortably at home and also use it for other activities too. Assignment and creation of the GitHub repositories for each student, or group of students, is orchestrated by GitHub Classroom (https://classroom.github.com). Reports are written in R Markdown (https://rmarkdown.rstudio.com/) that combines the prose with R code to produce analyses results, plots and tables directly inside the documents. All repositories are ultimately cloned by the teachers in a centralized area on our servers and data about commits (git logs) are collected using git

TABLE 1. Four levels of increasing difficulties in the exercises.

Level	Description	Type
L1	Interactive exercise in the course, direct feedback	h5p
L2	Tutorial with guided exercises, feedback and hints	learnr
L3	Individual and guided data analysis	individual project
L4	Free data analysis and reporting (by 2 or 4 students)	group project

version 2.31.1 and R version 4.0.5 [TODO: cite R properly here]. To give an idea of the data recorded, in 2020-2021 we have a little bit more than 3,500 events recorded for each student.

In distance learning, students' support was done via email and Discord (https://discord.com). At the end of an academic term, all recorded messages were collected into text files. These files were scraped using custom R code to create a table with key information (basically, who, when, and what) for each message. Surveys are done periodically in-class though Wooclap questionnaires (https://www.wooclap.com). Such a questionnaire was used to query perceived workload of the learnr tutorials. Wooclap allows to export data into Excel files. These data are then converted into a table in our database with an R script.

Data about users, courses, lectures and projects, as well as grading items (on average, more that 130 grading items were established for each student in 2020-2021) are anonymized: names, emails and all the personal information are replaced by random identifiers. The different tables are ultimately exported into CSV files and made public. These data are available at [… Zenodo?]. Data collection, treatment, and use respect European GDPR (General Data Protection Regulation) since each student had to agree explicitly with the way data are collected and used (including for research purpose) before each course begins. They can visualize their data through personalized reports at any time.

The course material is organized in a way that favors autonomy and self-assessment (direct feedback in the exercises, hints and retry buttons in case of wrong answers). Table 1 summarizes main characteristics of the exercises according to the difficulty level.

R and tidyverse packages (https://www.tidyverse.org) were user to prepare the data and for the analyses. A GitHub repository with the code used to create the figures and table in this paper is available at https://github.com/BioDataScience-Course/teaching_data_science_in_biology. [TODO: more details on statistic analyses, in particular SOM to put here + refs… Guyliann?]

A NASA-LTX questionnaire was used to study perceived workload to complete a task. It is composed of six questions on a Likert scale [14]. The questions concern mental load, physical load, time pressure, expected success, effort required, and frustration experienced during the accomplishment of the task. The average value for the six questions constitutes a Raw Task Load indeX (RTLX) [6] that we used to quantify how students feel when using these learnr tutorials.

3. **Results.** This study is performed on data originating from three successive courses that comprise 26 modules in total in 2020-2021. Table 2 summarizes the number of H5P, learnr, individual and group GitHub projects that students had to complete. Group projects usually span over several modules. It should be noted

TABLE 2. Number of students, modules, and exercises for each course. For the learnr tutorials, the first number is the amount of tutorial documents and the second number in brackets is the total number of questions in these tutorials (year 2020-2021).

Course	Students	Modules	H5P	Learnr	Indiv. projects	Group projects
A	59	12	59	24 (211)	10	4
B	45	8	29	11 (108)	12	2
C	26	6	19	7 (37)	7	1

that for course C, we also introduced a challenge in machine learning that replaced one group GitHub project. This challenge is omitted from the present analysis, being an isolate activity that is difficult to compare to the rest. However, this explains why there is only one group project in course C.

Retrospective data from 2018-2019 (only course A) and 2019-2020 (courses A and B) are also used when it is pertinent. For instance, final exams were only used during these two years. It should be kept in mind that the pedagogical material was written and improved progressively during the three academic years. The H5P exercises and the auto-checking of learnr answers were not available before 2020-2021. We do not use data corresponding to our older courses in biostatistics given in a more traditional way because we consider the comparison is not fair: the content of these courses is quite different. However, experience gathered with these old courses during 15 years was critical in the redesign of the new ones.

3.1. **Measured and perceived cognitive workload in learnr tutorials.** In our courses, learnr tutorials play an essential role in the progressive acquisition of competences because they are at the transition between the theory (online book chapters) and the practice (projects). These tutorials are interactive documents that recall main concepts, and take the students by the hand to perform their first data analysis step by step. At each step, they have at least one exercise or one quiz. The exercise consists in writing R code, or to fill missing parts in R code to progress in the analysis.

Our goal with these tutorials is to prepare the students optimally for the practice of data science. On the other hand, we do not want to exhaust their mental energy just before they start to work on their projects. The efficiency of these tutorials is qualitatively determined by observing the behavior of the students when they start their practical work, but we also have quantitative indicators available, like the number or retries necessary to complete an exercise on average, the number of exercises correctly answered, or the time needed to complete one tutorial.

A few tutorials were elaborated during the academic year 2018-2019, and positive feedback on their utility (both by direct observation of the students and thanks to their remarks) led us to systematize them into what we now call level 2 activities (see Table 2) in the form of learnr documents in 2019-2020. The tutorials were further refined in 2020-2021: we added contextual hints thanks to the gradethis R package (https://pkgs.rstudio.com/gradethis/). When students submit their answer to the exercises, the R code is parsed, analyzed and the result is compared with the solution. In case of differences, heuristics are used to provide contextual hints. Students can then refine their solution and resubmit it. This appears very

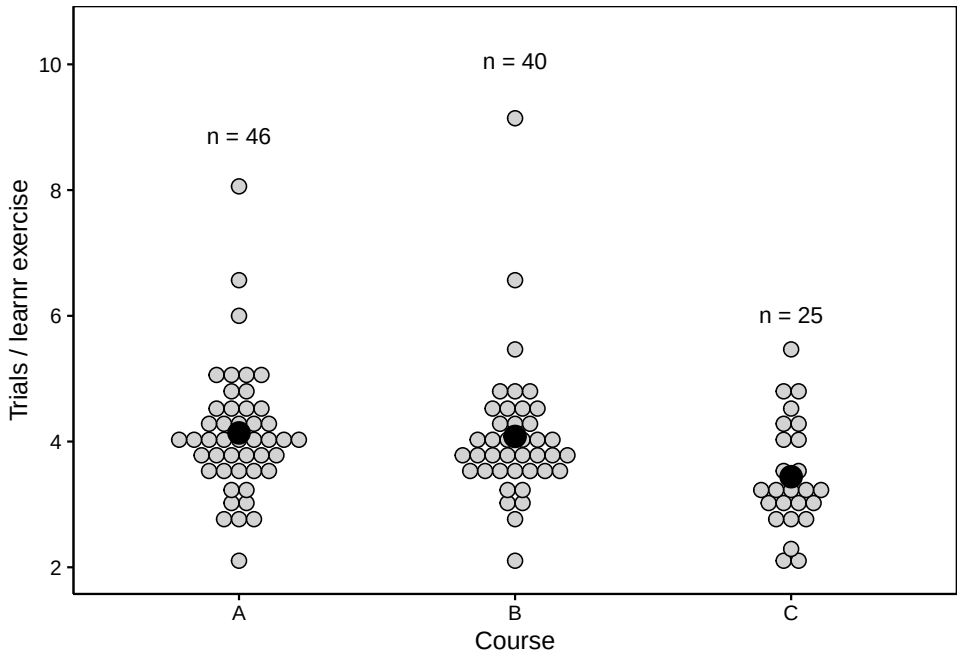

FIGURE 1. Average number of retries that where required for each student to find the right answer in learnr tutorials exercises (year 2020-2021). This measure is used as an indirect, but objective measurement of the cognitive workload. The black dot is the average for the whole classes and *n* is the number of observations.

efficient in self-teaching and self-assessing their competences before switching to the practice with confidence.

The objective measurement of the cognitive workload is estimated by using a proxy: the average number of entries that were required for each student to find the right answer in learnr tutorial exercises (Fig. 1). Only data from students that performed most of the exercises were used. This variable varies significantly between the three courses (ANOVA, $F(2,109) = 3.655$, p-value = 0.029). The students in course C need significantly fewer trials to find the right answer than students in courses A at α level of 5% (Tukey HSD, t = -2.489, p-value = 0.0375) and B (Tukey HSD, t = 0.0474, p-value = 0.047)

The perceived cognitive load required to perform these exercises is also determined on the same students and for the same exercises. The Raw Task Load indeX measures emotional state of the students after having completed a tutorial. This has, as far as we know, not been studied yet. We used a NASA LTX questionnaire to assess it across all three courses. Participation to the survey was high: 48/59 (81%), 35/45 (78%) and 18/26 (69%) for courses A, B, and C respectively.

The difficulty of the course, and thus, of the exercises in the tutorials increase from one course to the other. However, we do not observe an increase, neither in the number of retries, nor in the Raw Task Load indeX (Fig. 2). On the contrary, these appear significantly lower for course C than for course A at the α level of 5% (ANOVA, $F(2,98) = 3.588$, p-value = 0.031; Tukey HSD, t = -2.679, p-value = 0.023). The cognitive load perceived by the students diminishes at the same pace

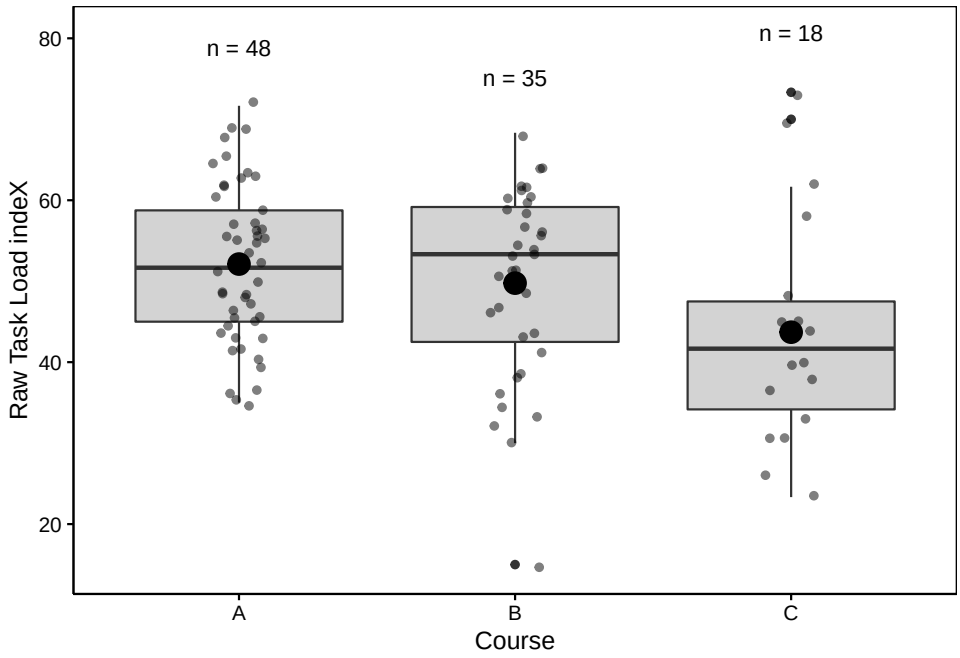

FIGURE 2. Perceived workload for the learnr tutorials in the three courses (year 2020-2021). The black circle is the mean RTLX value. The number above each box is the number of respondants.

as their ability to find the right answer in less trials. This may be a consequence of a more fluent R coding and a better mastering of the software environment.

3.2. **Final exam *versus* project.** An exam at the end of an academic term is a common practice as summative assessment. So, we compare grades our students got from such an exam with score they obtain directly in their projects in 2018-2019 and 2019-2020. The final exam is written in learnr, and it mixes a few questions about the theory with several partly solved data analyses that students have to explain, criticize and finalize during the exam session on the computer. The exam is thus largely focusing practical analysis, in line with the expected outcomes.

The comparison of the grades obtained by each student for a project and a final exam shows only a weak correlation between these two types of evaluations (Fig. 3). In 2018-2019, only one student failed in the project, while almost one third of them failed their final exams. The difficulty of the project was similar to previous years with the old course in biostatistics (and failure was not uncommon at that time). The flipped classroom approach leaves more time in class to work on practical applications, to ask questions, to discuss results, … We hypothesize that the very low failure rate in the projects could be explained by a better preparation to practical data analysis, but not to the final exam.

In 2019-2020, we raised a little bit the difficulty for the projects, resulting in a more widespread distribution of the results, but with a similar pattern showing very little correlation between the two evaluation methods. The same conclusion can be drawn for course B, with several students failing in one of the two evaluations, but not in the other one.

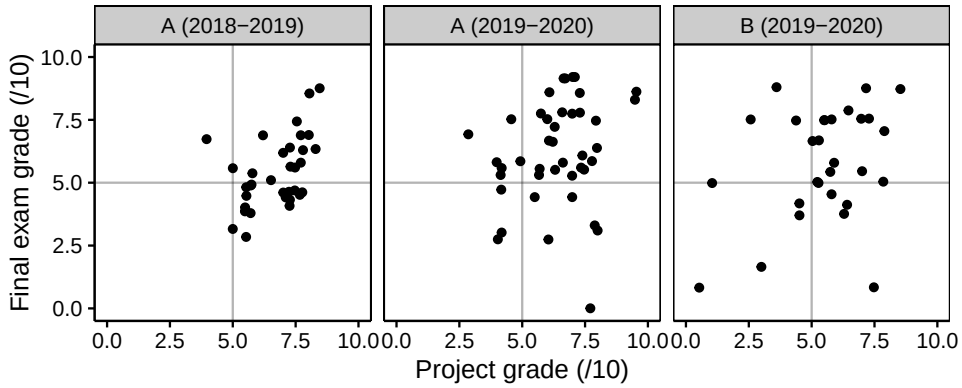

FIGURE 3. Grades obtained at the final exam in function of grades obtained for the projects for courses A and B during two years (course B was still in its old form in 2018-2019 and is thus not represented).

Despite, a summative assessment that includes a series of questions involving writing R code to analyze data in the final exam, it does not reflect properly the ability of the students to correctly process and analyze biological data, as in their projects. Alignment with intended learning outcomes seems thus to be more present in the projects. Due to these results, the final examination was abandoned for the academic year 2020-2021, and it has been replaced by an ongoing assessment of the students' activities across all four level exercises. These activities are analyzed in the following section.

3.3. **Students activity profiles with ongoing assessment.** Despite the fact that we have relatively homogeneous classes of students with similarly (low) level of knowledge in statistics and computing initially, the flipped classroom approach and the proactive attitude we expect from our learners (they must formulate questions correctly whenever they face a problem) conduct to different and contrasted learning strategies. Not all students ask questions. Some of them try to find solutions on their own. Some other prefer to ask their questions privately, while others have no problems exposing their difficulties on a public forum (the Discord channel dedicated to the course). The way and the timing learners progress in the exercises also largely vary. The schedule is not tight and only suggests a rhythm of progression. No student is penalized if the exercises are done later, as soon as they are completed before a final deadline. As expected, a part of our students prefer to stick to the proposed schedule, while others procrastinate and differ the completion of their exercises. Some strategies are more efficient than others. We analyzed records of the students' activities to distinguish these profiles and we compare them with the grade they obtain at the end of the course.

In 2020-2021, to support the ongoing assessment without a final exam, the activity of each student in level 1 (H5P) and 2 (learnr) exercises was exhaustively recorded in a database. For the GitHub projects (levels 3 and 4 exercises), it is the GitHub repositories and the git log data that are analyzed. During lockdown periods, exchange with students and answers to their questions were exclusively done by email, text or voice messages on Discord on private or public channels. Students

were allowed to freely choose their favorite tool to interact with the teachers and between each other. All these exchanges were recorded too.

The degree of completion of all the exercises was used to establish the final grade for the course, with a much higher weight on individuals and especially, on group projects. The weight was adjusted from course to course according to the importance of the different projects, mainly. To give an idea, for course A second term, level 1 H5P exercises accounted for 5%, 10% for level 2 learnr tutorials, 35% for level 3 individual projects and 50% for level 4 group works. On average, each student received more than 130 assessment items that accounted for their final grade. Two third of these assessments were established manually, using evaluation grids and based on their reports in the projects. The remaining third is made of scores automatically calculated from the various online exercises.

For the three courses, we recorded more than 450,000 events, which makes on average almost 3,500 events for each student. These data contain information to characterize the behavior and learning patterns that the students use. They are summarized into sixteen metrics.

For H5P exercises:

- trials/H5P ex.: the average number of trials for each H5P exercise (students can retry as much as they wish and they have immediate feedback if their answer is correct or not),
- correct H5P ex.: the fraction of H5P exercises that were correctly answered,

For learnr tutorial exercises:

- trials/learnr ex.: the average number of trials for each learnr exercise (here also, students can retry as much as they want), excluding quizzes,
- hints/learnr ex.: in learnr exercises, students can display hints to help them to solve the problems (but they loose 10% of the score for the exercise for each hint they reveal). This is the average number of hints per exercise that were displayed by each student,
- correct learnr ex.: the fraction of learnr exercises that were completed with a correct answer,
- time/learnr ex.: the average time required to finish one learnr exercise involving R code writing, thus excluding quizzes.

For individual and group projects:

- commits/ind. projects: the average number of commits done by a student in one individual project,
- contributions/ind. projects: the number of lines changed -added or subtracted- in the R Markdown reports by one student in one individual project, on average (this includes embedded R code for the processing, analysis and plotting of data),
- commits/group projects: same as above, but for group projects,
- contributions/group projects: same as above, but for group projects,
- percentage of contributions to group projects: the fraction of work the student did, relative to all the work done in group projects.

For support:

- questions/module: the number of questions student asked, divided by the number of modules in the course,

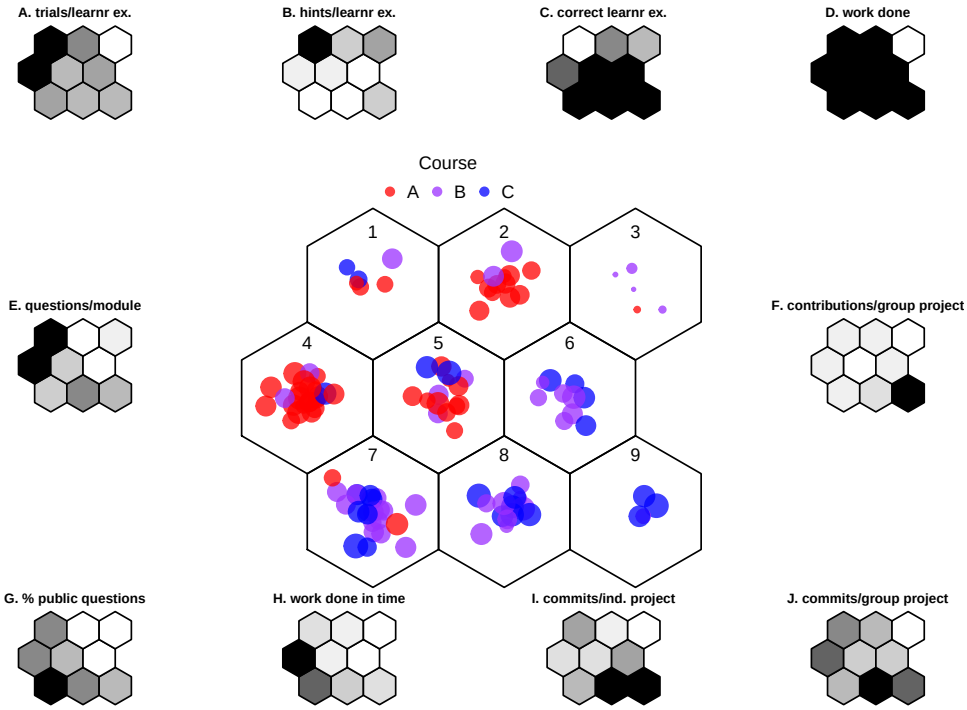

FIGURE 4. Self-organizing map of the student activities across the three courses (year 2020-2021). See the text for the explanations.

- percent of public questions: the fraction of questions that the student posted in a public channel (the Discord channel dedicated to the course that all the other students of the class can read),
- contributions/question: a metric that catches the relative "productivity" of the student related to the number of questions they ask.

Finally, global measurements:

- work done: the fraction of all exercises that the student finished,
- work done in time: the fraction the exercises done in the right time, that is, during the proposed calendar.

In our courses, we have a few students in mobility that come from various origins. The *a priori* knowledge is important in education. So, to avoid biases due to the past curriculum of the students, we restrict this analysis to the subpopulation that comes from the first year of Bachelor in Biology at UMONS only. A Kohonen' self-organizing map is used to create student profiles according to their activities (Fig. 4). A three by three hexagonal map was chosen, and students are thus classified into nine different groups.

In Fig. 4, the small peripheral plots in gray scale show how selected metrics distribute in the nine cells, from lowest value in white to highest value in black. They help to decipher the way students behave according to their profile. Metrics that are not represented in the figure exhibit similar patterns than others (for instance H5P metrics have a similar pattern as learnr metrics). Dots in the central plot are the various students, with color representing the course and the diameter of the dots

indicating the grade the students obtained at the end of the course. The following paragraphs detail information in that figure. The numbers between brackets mean the cell number in the central plot, and the upper case letters in brackets refer to the peripheral subplots.

Although most students finished all, or almost all exercises (D), cell (3) collects the few students that did only a tiny part of these exercises. These students obtained very low grades, of course. They belong to courses A and B. On the other hand, heavy workers are at the bottom (I & J), and good performers in learnr tutorials (C) are in cells (5-9).

- Cells (2 and 6) collect students that seldom ask questions (E), and that rarely appear on the public channel (G). Minor differences separate them. For instance, learners in cell (2) sometimes use hints (B), while those in cell (6) never do, also because they find the correct answer to the exercises more often by themselves (C). Asking questions is at the core of our pedagogical approach. So, these students do not play the game. However, they can possibly succeed. Some of them probably exchange with other students through different channels that we do not monitor. Cell (2) -more difficulties with learnr tutorials- mainly contains students belonging to course, A, while cell (6) contains students of courses B and C. There is a clear evolution in their behavior from one course to the other in term of ease in front of the exercises, even if they remain silent in the teacher-learner interactions.
- Among the students that have a hard time to figure out the answers to auto-evaluation exercises, cell (1) reassemble people that most heavily rely on hints (B), and are among those who need to retry the exercises more often before figuring out the correct answer (A), a characteristic they share with cell (4). These students also ask a lot of questions (E), both on the public and private channels (G, mid gray indicating a balance between public and private messages). Main difference between those two groups is that students in cell (4) try harder to find the answer without looking at the hints, while in cell (1) they give up more rapidly. Also these students respect the proposed schedule much more closely than all others (H). We have students in all courses there, but a majority from course A.
- All these cells (1-4 plus 6) are students that exhibit suboptimal behaviors in one or the other way. The remaining cells (5, 7-9) correspond to learner profiles that perform better from this point of view. Cell (5) is primarily represented by students from course A, but otherwise, also from course B and C. These are average actors in all metrics, except they are fluent with level 1 (H5P, not shown) and level 2 (C) exercises.
- Moving from cell (5) to (7), (8) and (9), we encounter increasingly top performers. The number of students from course A becomes progressively lower, while course B, and especially C dominate in these groups. In cell (7), they largely use the public channel (G) and also respect the schedule quite well (H) as main differences from those from cell (5). Students in cells (8) and (9) are not so often in time, but this is because they are heavier workers in the projects, both in the individuals (I) and in the groups (J) activities. This needs obviously more time. In cell (9) we also have the students that contribute the most to the reports in terms of lines added or deleted (F).

To summarize, at the top of the SOM map, cells (1-4, plus 6) contain students with suboptimal behaviors, cell (5) are average students, and cells (7-9) at the

bottom exhibit profiles corresponding to best performers. The pattern is also visible between courses A (mainly distributed at the top or center of the map) to B and C (more represented at the bottom). This suggests probably that students need time to get used to the course, its pedagogical approach, and/or the software environment they have to use. Since only a small fraction of the participating students fail, excluding the defeating ones in cell (4), the course pattern can hardly be explained by a filter of the low performers from one course to the other.

3.4. **Transition between face-to-face to distance learning.** Due to Covid-19 lockdown periods, distance learning had to be adopted abruptly. We analyze the activity collected during academic years 2019-2020 and 2020-2021 to evaluate the impact of these transitions on the progression of the students In Fig. 5, the academic term is divided into seven work periods of approximately two weeks each (remind that it is the suggested rhythm of the courses: one module every second week). The classes of the second term start at period Y1P09, since period Y1P08 is reserved for the exams. The courses of the first term of 2020-2021 begins at Y2P01. First lockdown started at period Y1P11 for one month and an half. Second lockdown stared at Y2P03 and lasted to the end of the second term (Y2P15). During the first lockdown, we quickly opened the dedicated Discord channels that were available without any latency.

Contributions par student (Fig. 5a) is relatively constant during the second year, starting essentially at Y2P03, when the second lockdown was established. The highest activity is observable at the end (Y2P15), although there is no module teach during that period. This is because of the late students that finalize their reports at the last minute. Y2P01, Y2P02 and Y2P09 exhibit the lowest activity, and these are the start of the first and second terms. Y2P01 and Y2P01 were also teach in face-to-face and they correspond to the start of all three courses.

The efficiency of learners-teachers interactions for that contribution, quantified by the contributions divided by the questions (Fig. 5b) is very widespread from one student to the other. That ratio spreads on several orders of magnitude. However, median value -the bar inside the boxes- varies much less. Global amount of questions during each period is less variable, as is the absolute contributions, leading to a rather stable ratio. Highest median ratios are observed at the last period of each term (Y2P07 and Y2P15) although no module was teach at that time. More contributions are observed relative to the questions at the end: students essentially finalize their reports.

The first year shows a different pattern. First, the lockdown period was restricted to the very end of second term. Only the last module in both course A and B remained. In Y1P12, when distance learning was first imposed, we observe a marked decrease in the contributions per student (Fig. 5a). It is heavily compensated in periods Y1P13 and Y1P14, which are by far the most busy periods of all. Period Y1P15 is not represented because it is after the deadline to finish all work that year.

Efficiency of support during the first year shows a similar pattern as for second year: extremely widespread from student to student. Median value is similar too, if not among the highest during periods Y1P11, Y1P13 and Y1P14. The productivity was thus not affected during that first lockdown, after a short lag time observable in Y1P12.

4. **Discussion.** Teaching data science to a population of students that are not very used to advanced computer techniques and tools, and that have only basic

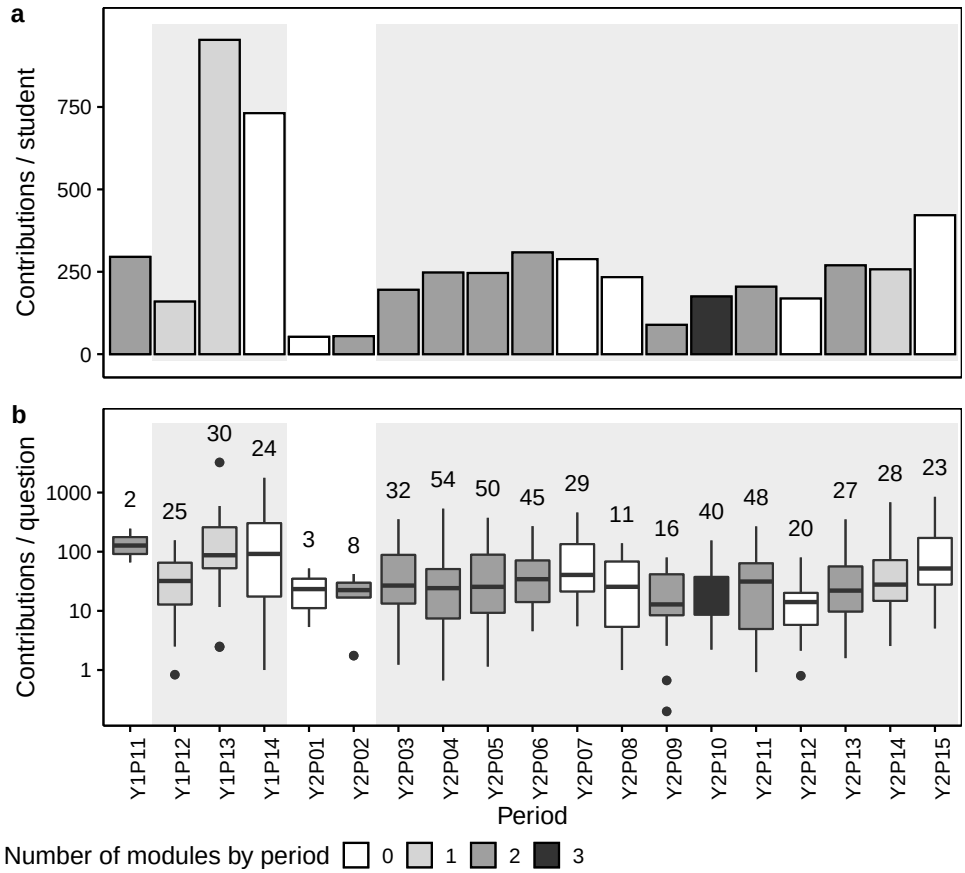

FIGURE 5. a. Average contributions of the students to the projects by periods of two weeks of course. b. Contributions by question asked (log scale) for each student as a proxy measurement of the efficiency of teacher-learner interactions on the progression in data analyses. Light gray background indicates periods where distance teaching was mandatory due to Covid-19 lockdown. The number of students that interacted during each period is indicated on top of the boxlots (Y1 is 2019-2020, Y2 is 2020-2021).

knowledge in mathematics and statistics is a hard task [21]. Our basic approach is to extend the training on a very long period: five successive terms spanning on three consecutive years (undergraduate and graduate). That way, the many different concepts they have to learn can be broken down into subunits (26 modules) that last for two weeks each. We also used highly developed flipped classrooms and blended teaching and learning (following Spadafora & Zopito's definition of "any educational model where online delivery ranges from 50% to 80%" [TODO: add citation here]), with an emphasis on proactive exchange with the teachers: students have to ask questions to progress. Overall, these are winning choices because most of our students were successful, excluding a few defeating ones. [9] also obtained good results using flipped classroom with its course targeting students in biology.

Before our courses, the students are more used to a traditional approach made of lectures followed by exercises where important concepts are repeated at the beginning of the practical sessions. They tend to have a passive attitude during lectures and they expect the teachers and assistants feed them with the key concepts. That attitude does not purposely work here. Proactive behavior and developing autonomy are required [13]. They thus have to engage themselves in a very different way of learning. The transition between the theory they read in the book and the projects where they have to apply these concepts is too abrupt without a progression in four stages, which are: (1) auto-evaluation exercises directly in the online book, (2) recall of the main concepts and guided step-by-step analysis of a first dataset with the learnr tutorials and (3) at least one guided individual project with another dataset, before (4) they are presented yet another dataset, with limited instructions this time. Task two, the learnr tutorial, was immediately spotted as a key activity in the learning process during academic year 2018-2019. So, we have focused our attention on these learnr documents. In 2020-2021, the use of an heuristic engine (gradethis) to provide contextual feedback on the errors students make in their answers was much appreciated. The measured RTLX index will serve in the future as a reference to gauge possible optimization of the tutorials, with lower perceived workload without sacrificing the content. The significant decrease in RTLX value from course A to C indicates that there is still a margin of progression. We would like to observe such a decrease sooner, perhaps already in the second course. It is indeed important "to maintain reflective and systematic approaches in both the development and evaluation [of our] blended approach" [TODO: cite Spadafora & Zopito].

Ultimately, the task that better evaluates their practical skills in biological data analysis, and which meets with the courses' outcomes, is the group project because students have to demonstrate what they can do in situation with a complex dataset, and minimal instructions. They have to figure out a suitable question, analyze the data to answer that question, present their results in a report and discuss what they found with a critical mind [1]. This task is complex, especially for undergraduate students. That is why we organize it in groups of two to four students, depending on the complexity of the problem. Here, they mobilize their collective intelligence to get results many of them would be hardly capable of achieving alone. The tracking of individual activity in the project thanks to git allows figuring out clearly what was the contribution of each student in the group and to score their individual contributions suitably.

Sometimes, groups do not work well, and one student has to do most of the work. This is a clear weakness in this approach, especially if one of the defeating students in Fig. 4 cell (3) is involved. If we could identify the profile of the different students relatively early during the course, we would be able to create better groupings with a blend of different complimentary profiles to enrich the experience of all learners. Maybe should we work exclusively with groups of four to mitigate the impact of one defeating student?

We observed a low correlation between performances in group projects and in grades obtained at final exams (Fig. 3). This led us to stop using final exams, at the benefit of an ongoing assessment with emphasis into projects. We had to set up a procedure to monitor and score the students' activity in all the exercises by automatic scoring online exercises, and to manually score all projects against evaluation grids. This is time-consuming despite the partially automatic scoring lowers the

charge. It would be interesting to investigate whether tools derived from learning analytics and computing science could second teachers in this work. For instance, reproducible research is one of the competences our students have to develop. It implies that their R Markdown documents should compile into reports in HTML or PDF format without any error. This criterion could be checked automatically.

Activity tracking in the exercises, primarily set up for the continuous evaluation, also offers the opportunity to study the way learning happens (or not). Learning analytics are primarily used to early predict success or failure. In our courses, failure rate is already rather low and essentially limited to a few defeating students that do not work at all. We are more interested to classify our participating students according to their behaviors. This paves the way toward a more inclusive pedagogy by spotting different kinds of suboptimal patterns (for instance, never asking questions, looking at hints too quickly without really trying to figure out the answer, being shy to discuss problems on public channels, ...) Once these patterns are evidenced, we can think of countermeasures. As an example for students that rarely post their questions publicly, we will test an alternate discussion channel were teachers never post but have read access. In case of an error, the teacher will contact the student privately to explain him what is wrong. That student would then have the responsibility to reexplain corretly to its siblings. This way, its error is never publicly spotted by the teacher. This approach was very successful with our so-called "eleve-assitants" -students from higher classes that have brilliantly succeeded in the course one or two years ago and that second the teaching staff-. With tools like the self-organizing map we should be able to predict suboptimal student profiles early. We could engage a discussion with the concerned persons to determine the cause and find a solution as early as possible. Learning analytics used that way would promote a differential pedagogical approach, a key for more inclusive teaching [20].

Forced distance learning, due to Covid-19 lockdown did not appear to be a barrier in the production of our students in their projects. A pattern is observed during first lockdown with a marked decrease in their contributions, followed by a large, compensatory activity. All this happened in a time frame of a couple of weeks. That was the time needed to adapt to the new situation. The most problematic aspect was the access to a powerful-enough computer for roughly 15 to 20% of our students. In a normal situation, these students had access to computers at the university, both during and outside of the courses. When lockdown was established, those students suffered most by a lack of hardware. However, to reduce the social fracture, the university quickly reacted and computers were lend to them. During the second lockdown, a larger part of our students had acquired their own computer, and solutions were immediately available for the others. Consequently, no lag time was observable in their production.

If the amount of questions and the efficiency of the teacher-learner interactions, quantified in term of contributions/question, remainded globally at a similar level in face-to-face or distance learning, their impact on the teachers timetables was very different. In distance learning, students tend to work at very different moments. Their questions are thus less concentrated during the courses. Also, an alternance between asynchronous work at home and synchronous work in the computer lab is more beneficial to interactions between students. The social and human components of teaching and learning are key factors that tend to vanish in pure distance learning. Contacts though videoconferences only partly compensate for these lack

of interactions because physical presence remains different to video chats. Blended learning combines the best of both worlds [13].

5. **Conclusion.** Teaching data science comes with challenges. The discipline is quite young, and we still are seeking the best pedagogical approach. After three years of teaching data science to undergraduate and graduate students in a curriculum in biology with revised pedagogical practices, we have our first cohort that has passed all three courses. There are still two optional courses available in second year of the Master if they want to push their data science skills further on. However, the three mandatory courses are designed to be sufficient by themselves. Globally, most students acquired the competences during these courses. We have the feeling that they are more mature and more capable in data science than with our previous courses in biostatistics given in a more traditional way. The impact of the revised approach to teach biological data science on the way learners manage data and data analysis will be observable during the following years. We will monitor how these students apply their skills in their Master thesis, and later, in their career or during their PhD thesis. Meanwhile, we will continue to improve our courses by further exploiting the data we accumulate on the activity of our students. Experience gathered during forced distance learning during Covid-19 lockdown will be used too to improve our courses. The radical changes that were required in that context showed that students can accommodate to a large extent, but also that a diversification of the activities is beneficial [TODO: citer Spadofora et Marini 2018… ou autre car déjà beaucoup cité dans le papier]. Speaking about diversification, in 2020-2021 we have successfully tested a kaggle-like challenge (`https://www.kaggle.com/competitions`) in one of the machine learning modules. Such ludic activities would also contribute to the diversification of pedagogical practices, interest and motivation of the students [TODO: refs needed if possible]. We would also be happy to share experience with other teachers in data science. Altogether, we are on the way to reshape the post-covid teaching landscape, and it will probably be quite different to what we are used today!

REFERENCES

[1] L. A. Auker and E. L. Barthelmess, Teaching R in the undergraduate ecology classroom: approaches, lessons learned, and recommendations, *Ecosphere*, **11** (2020), e03060.

[2] M. Baker, 1,500 scientists lift the lid on reproducibility, *Nature*, **533** (2016), 452–454.

[3] G. C. Banks, J. G. Field, F. L. Oswald, E. H. O'Boyle, R. S. Landis, D. E. Rupp, S. G. Rogelberg. Answers to 18 Questions About Open Science Practices, *Journal of Business and Psychology*, **34** (2019), 257–270.

[4] B. Baumer, M. Cetinkaya-Rundel, A. Bray, L. Loi and N. J. Horton, R Markdown: Integrating A Reproducible Analysis Tool into Introductory Statistics, *Technology Innovations in Statistics Education*, **8** (2014).

[5] C. Boettiger, An Introduction to Docker for Reproducible Research, *SIGOPS Oper. Syst. Rev.*, **49** (2015), 71–79.

[6] - J.C. Byers, A. Bittner and S. Hill, Traditional and raw task load index (TLX) correlations: Are paired comparisons necessary?, *Advances in Industrial Ergonomics and Safety*, 1989, 481–485.

[7] M. Cetinkaya-Rundel and C. Rundel, Infrastructure and Tools for Teaching Computing Throughout the Statistical Curriculum, *American Statistician*, **72** (2018), 58–65.

[8] M. Cetinkaya-Rundel and V. Ellison, A Fresh Look at Introductory Data Science, *Journal of Statistics Education*, **0** (2021), 1–27.

[9] P. Compeau, Establishing a computational biology flipped classroom, *PLoS Computational Biology*, **15** (2019), 1–8.

[10] D. Donoho, 50 Years of Data Science, *Journal of Computational and Graphical Statistics*, **26** (2017), 745–766.
[11] R.A. Estrellado, E. A. Emily, J. Mostipak, J. M. Rosenberg and I. C. Velasquez, *Data science in education using R*, 1st edition, Routledge, London, England, 2020.
[12] J. Fiksel, L. R. Jager, J. S. Hardin and M. A. Taub Using GitHub Classroom to teach statistics, *Journal of Statistics Education*, **27** (2019), 110–119.
[13] D. R. Krathwohl, Active learning increases student performance in science, engineering, and mathematics, *Proceedings of the National Academy of Sciences*, **111** (2014), 8410–8415.
[14] S. G. Hart and L. E. Staveland, Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research, *Advances in Psychology*, **52** (1998), 139–183.
[15] C. Hsing and V. Gennarelli, Using GitHub in the Classroom Predicts Student Learning Outcomes and Classroom Experiences: Findings from a Survey of Students and Teachers, In Proceedings of *the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*, 2019, 672–678.
[16] D. R. Krathwohl, A Revision of Bloom' s Taxonomy: An overview, *Theory Into Practice*, **41** (2002), 212–218.
[17] K. Larwin and D. Larwin, A meta-analysis examining the impact of computer-assisted instruction on postsecondary statistics education: 40 years of research, *Journal of Research on Technology in Education*, **43** (2011), 253–278.
[18] V. Marx, The big challenges of big data, *Nature*, **498** (2013), 255–260.
[19] J. A. Onwuegbuzie and V. A. Wilson, Statistics Anxiety: Nature, etiology, antecedents, effects, and treatments–A comprehensive review of the literature, *Teaching in Higher Education*, **8** (2003), 195–209.
[20] G. Siemens, Learning Analytics: The Emergence of a Discipline, *American Behavioral Scientist*, **57** (2013), 1380–1400.
[21] B. Sousa and D. Gomes, Teaching With R—-A Curse or a Blessing?, In Proceedings of *the Tenth International Conference on Teaching Statistics (ICOTS10 '18)*, 2018, 672–678.

Received xxxx 20xx; revised xxxx 20xx.

E-mail address: Guyliann.Engels@umons.ac.be
E-mail address: Philippe.Grosjean@umons.ac.be
E-mail address: Frederique.Artus@umons.ac.be