Teaching relational database fundamentals: a lack-of-progress report

Charles Boisvert[1]

Sheffield Hallam University, City Campus, Howard Street, Sheffield S1 1WB, UK,
c.boisvert@shu.ac.uk

Abstract. This paper presents an account of five successive years teaching a relational databases module, describing and evaluating changes introduced. We highlight some changes that improve the students' learning, but also the difficulty of achieving certainty when evaluating interventions.

Keywords: Computer Science Education, Database Systems, quasi-experimengt

1 Introduction

This paper discusses the successive attempts to improve teaching of a module on relational database systems over five academic years. During the period, various problems were identified and interventions attempted to alleviate them. But not all attempts were fruitful, and not all were carefully controlled. Comparing results over time, we identify some interventions which were successful, but some hypotheses were found to be without foundation, and some choices that should not be repeated.

The paper presents first a general introduction to the module and the data that has been collected for it, then presents a history of changes over time, before evaluating the effectiveness of those changes, and discussing oiur ability to evaluate interventions reliably in practical conditions.

2 Background information: module and data available

2.1 Module description

The module proposed at Sheffield Hallam University, that we will consider here, has varied little in six years, from September 2012 to the present. It covers the basis of relational databases, their design using top-down and bottom-up methods, the use of a Database Management System (Oracle) and the basics of the SQL language.

It is destined to *Business and Enterprise* students who study courses involving both some business and some Information Technology. In earlier years, an identical module was delivered to more technical cohorts of students, and one

of the events in these five years is that these technical students' were given a different curriculum.

The duration and time devoted to the module has not changed in the five years: the module lasts the whole academic year, with two hours of contact time each week, one for a lecture and one for practical work in a smaller group. The recommended amount of study time has also not changed.

2.2 Data available: a first look

Over the period we have kept data on student marks and work completed each year, as well as information about the nature and dates of the homework used. Cohort numbers vary widely, with the minimum in 2015-16 being nearly half the largest group two years earlier. This fluctuation reflects that of the courses the students join each year.

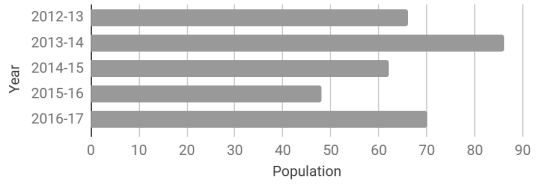


Fig. 1. Student population, for each year that past data is available. Students registered, but not submitting work and attending examinations are not included.

Not all the data, however, is suitable for analysis. In particular, of the 5 years for which data is available, one stands out for the unusual distribution of the results. The results for the year 2013-14 stand out for their unusual distribution, and for being lower overall than each of the later years and than 2012-13.

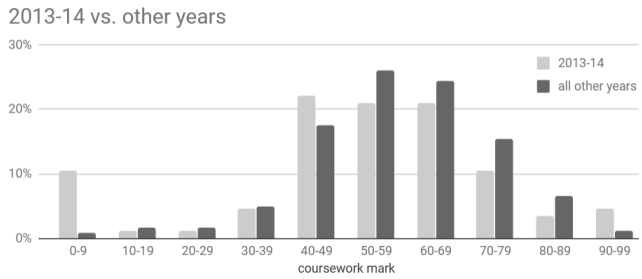


Fig. 2. the year 2013-14 lies out for the poorer module results, and the unusual results distribution.

These peculiar results are explained by events of that year. A policy enforced to simplify assessment required the module team to limit the number of pieces of work carried out by students to just two - one piece of coursework, and one examination. Without preparation for the students, without the opportunity for early formative feedback, with a single, complex piece of individual homework due late in the academic year, the results were unusually poor. They then had to be moderated, adapting the marking to make sure the module results were kept acceptable. The unusual distribution illustrated fig. 2 is a product of these results and moderation.

The policy was abandoned a year later. The unusual situation means that that year's results is an outlier for most purposes. But these events gave the impetus to curate an archive of key module results and data, and to be able to identify trends and provide a factual basis for making and evaluating decisions.

The data set is therefore made of work submission and marks data for the academic year 2012-13 and the three years of 2014-17, plus outlier data for the year 2013-14.

3 Evolutions of the module

In this section, we describe the successive changes introduced in the period. Some were very deliberate and chosen with the intention to improve the module, but others were in reaction to events or needs becoming apparent; and finally, some simply resulted from opportunities becoming available. We describe them mainly, in historical order, to facilitate later evaluation.

3.1 The need for change

Databases for Business has a difficult history. It is necessary for its students, but it is also challenging. In the courses - itBusiness and ICT, *IT with Business Studies* - that include it, this second year module is the first in which students are required to use any computer language; neither have they been taught logic or set theory previously.

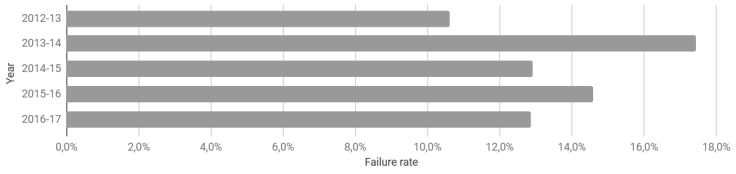


Fig. 3. First-time failure rate over the successive years. Even excluding the year 2013-14, module changes have not brought an improvement for the regular minority of failing students.

With such demands on the students, it is not surprising that every year some of them fail to complete the work satisfactorily. The module failure rate shown in fig. 3 makes this clear.

3.2 Improving an SQL workbook

The module has been supported by a study book for SQL since several years before the data presented here started to be collected.

The workbook combines practice exercises and pointers to key information. Learning material, in this work, is deliberately limited to reminders and references to other materials. This both dedicates the workbook to practice, and encourages the students to refer to more complete information; but navigation - section naming, titles, order of topics addressed - is kept consistent with other learning material.

This work requires a lot of care and the module team has constantly worked to update and improve the visual quality, the text, and the referencing of the workbook since 2012-13. Updates have continued throughout the years, identifying poorer explanations and examples, to communicate the subject better with each new edition.

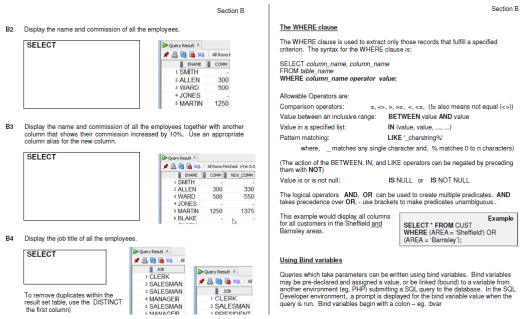


Fig. 4. SQL exercises from the Workbook

3.3 Succession of coursework tasks

In 2013-14, as we discussed in section 2.2, the practice of multiple small marked tasks ended. Four separate marked tasks were replaced by a single large piece of coursework. Since then, the coursework has returned to two marked pieces.

3.4 A spin-off module

A year later, from 2014-15, the module destined for more technical students was made separate, with the aim to adapt the teaching to each cohort. The

difference had always been clear between business students for whom SQL is the only computer language they encounter, and software engineering students who practice many, and study the theoretical underpinnings needed to understand many more; it was more visible still as the exceptional effect of that year was primarily felt by the business students.

3.5 Delaying the examination and introducing video material

The next action, in 2015-16, was to re-organise the examination: instead of testing the students at the end of the first semester, in January, the test was moved to the end of the academic year in May. The aim was to allow time for the students to develop their understanding and practice of `SQL`. To support this practice, questions were redesigned to facilitate the release of past examinations texts to students.

Finally, a set of videos was recorded and made available –on a Youtube channel– which emphasises, and delivers asynchronously, key elements and summaries of the material. The material is carefully selected to emphasise key points, then scripted to make sure that each film is short, focused on a single point, and clear. This keeps each video under 10 minutes. In keeping with the approach to vary the points of view on subject information, the topics addressed are segmented to match the sections in the workbook, and the workbook was edited to reference relevant video material at key points.

Fig. 5. A subset of the video material. The few minutes' duration of each recording is visible by each thumbnail, resulting in multi-part topics

3.6 Introducing automatic SQL feedback

the module team had long hoped to introduce automatic SQL feedback. It is clear that learning SQL needs to be supported by more than pen and paper practice [4, 5]. In 2016-17, a student developed `TestSQL`, an interactive web application to facilitate this study [3]. The system runs a relational database imported in

`sqlite` format, and dynamically constructs questions for that database. Students answers receive several checks to give appropriate feedback.

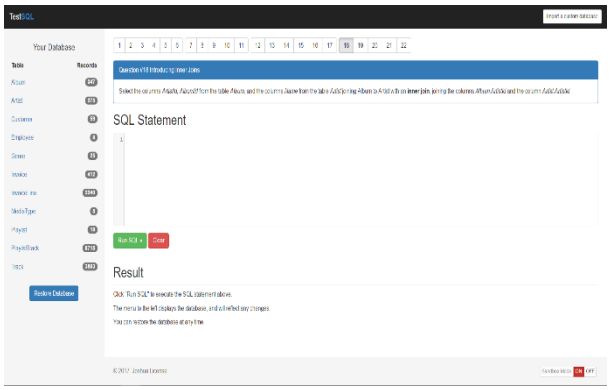


Fig. 6. A `TestSQL`session: automatic feedback gives the student immediate information about their query

Being the result of a student project, the work was available late in the academic year, but was immediately adapted to support the students preparing for examinations.

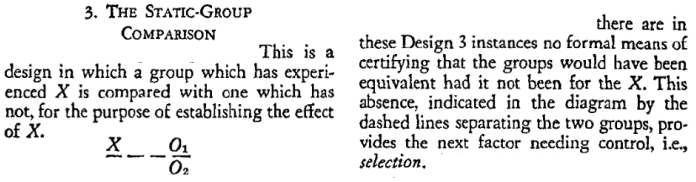
`TestSQL`was also adapted this year (2017-18) by developing a set of prepared questions to match the exercises and example data used in the workbook, and a further evaluation is in progress.

4 evaluating interventions

The data available provides some basis to support evaluating the successive interventions. To understand whether the changes introduced have made any difference, we analyse the results data for the sets of years before, and after, particular changes were implemented.

This follows a 'quasi experimental' method, and we should remain aware of the limits of the approach. [2] discusses different designs, discussing potential threats to validity of each. Ours is illustrated fig. 7 and they raise the important objection, that we cannot guarantee the groups compared are identical except for the intervention.

We're also aware that the changes year-on-year are not isolated interventions, and therefore it is difficult to attribute changes in the results to a specific chosen action. Nevertheless, this is the best, we may say, the *least worst*, method available. This is characteristic of the difficulties encountered in a practice setting. To quote again [2], 'insofar as the designs become complex, it is because... of the experimenter's lack of complete control'.



3. THE STATIC-GROUP COMPARISON

This is a design in which a group which has experienced X is compared with one which has not, for the purpose of establishing the effect of X.

$$\frac{X}{} \quad \frac{O_1}{O_2}$$

there are in these Design 3 instances no formal means of certifying that the groups would have been equivalent had it not been for the X. This absence, indicated in the diagram by the dashed lines separating the two groups, provides the next factor needing control, i.e., *selection*.

Fig. 7. Facsimile of the 1963 classic study of quasi experimental designs [2], showing our conditions

4.1 Video material and later exam in 2014-15

In this first example, we compare the two years of 2012-13 and 2014-15 on one hand, to the two of 2015-2017. In the later two years, we hoped to improve the students' prospects with three improvements: a later examination, video materials on SQL, and sets of past examination questions.

The results data contain marks for SQL examination questions, which have always been difficult for these students. The distribution of marks appears to show a clear improvement in the second group, although the chi square test gives a probability $p = 0.053$: it is a low probability, but not so low that we should rule out the difference being a result of chance.

range	15-17	before
0-6	1,8%	5,5%
7-13	4,5%	10,3%
14-20	10,9%	15,8%
21-27	25,5%	17,1%
28-34	21,8%	15,1%
35-41	12,7%	13,7%
42-48	11,8%	14,4%
49-55	7,3%	8,2%
56-66	3,6%	0,0%

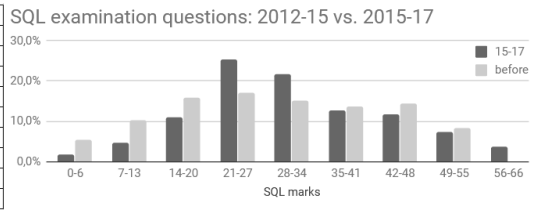


Fig. 8. Distribution of SQL marks before, and since, the 2015 examination.

To try to identify which of the changes were most beneficial, we can compare these results to the second part of the examination, which focuses on security and concurrency questions. The benefit apparent in SQL questions is not reproduced in this second part of the examination. As the Databae Administration was supported by past questions, but not supported by video material, and as it was presented in the final weeks of the year, a few weeks before the exam, we can hypothesise that the availability of past questions was not as effective, and that the better marks are the result of having more time to prepare and the better revision material.

Two more evaluation results deserve a mention. The `TestSQL`tool introduced in 2017 had no visible effect on the students' SQL performance: but, introduced late in the year with little support, this may be due more to the circumstances than to the tool itself. Finally, some of the team have long held the belief that

range%	15-17	before
0-9	10,2%	10,2%
10-19	11,0%	11,7%
20-29	18,6%	13,3%
30-39	15,3%	4,7%
40-49	16,1%	11,7%
50-59	13,6%	18,0%
60-69	11,0%	7,8%
70-79	2,5%	7,8%
80-89	0,0%	7,8%
90-99	1,7%	7,0%

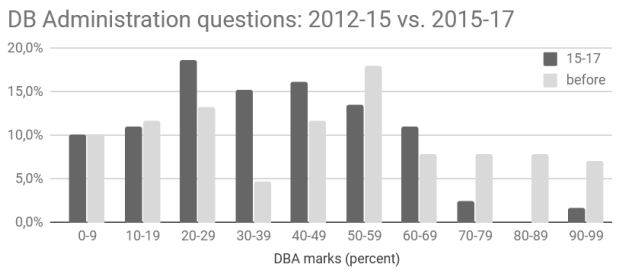


Fig. 9. Distribution of Database Administration marks before, and since, the 2015 examination.

the two courses studying this module at present have widely different results. Statistical testing clearly indicates they do not: the probability of differences between the distribution of marks between them, in 2016-17, was 0.75.

5 conclusion

We present this work in the belief that, as [1] write, 'negative results can be as valid as positive results in the scientific endeavor': that is, although we would desire both greater scientific rigour and more positive results for interventions presented here, there are valuable lessons to be learnt from the succession of attempts, partial successes and downright failures in the five years of data.

The difficulty in evaluating separate interventions shows the importance of collecting and analysing traces, which can provide fine grained details on students activity. But we hope that this work shows that where such traces cannot be available, investigation does not have to stop.

References

1. J. Boustedt, R. McCartney, J. Tenenberg, S. D. Anderson, C. M. Eastman, D. D. Garcia, P. V. Gestwicki, and M. S. Menzin. It seemed like a good idea at the time. *ACM SIGCSE Bulletin*, 40(1):528–529, 2008.
2. D. T. Campbell and J. C. Stanley. Experimental and quasi-experimental designs for research. *Handbook of research on teaching. Chicago, IL: Rand McNally*, 1963.
3. J. License. testsql: Learn sql the interactive way. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '17, pages 376–376, New York, NY, USA, 2017. ACM.
4. S. I. Mehta and N. W. Schlecht. Computerized assessment technique for large classes. *Journal of Engineering Education*, 87(2):167–172, 1998.
5. J. R. Prior. Assesql: an online, browser-based sql skills assessment tool. In *Proceedings of the 2014 conference on Innovation & technology in computer science education*, pages 327–327. ACM, 2014.