

Research support-oriented MATLAB learning: tackling difficult concepts and promoting personalised learning

Chunhua Yang^{1*} & David R. R. Smith²

¹School of Mathematics and Physical Sciences; ²School of Life Sciences; University of Hull, Hull, HU6 7RX, United Kingdom

*Corresponding Author: c.yang@hull.ac.uk

Keywords: *Programming; MATLAB; Research support*

Abstract

This study investigated the acquisition of MATLAB programming skills by postgraduate students, and whether this learning was improved by research support-oriented teaching. Questionnaire surveys were given to academic staff asking about what they considered the most important knowledge and skills in programming to be. Questionnaire surveys were also given to students asking about what programming concepts they found the most difficult and confusing to understand. The intersection between what knowledge and skills in programming the researchers deemed the most important, and what areas in programming students had most difficulty with, was carefully addressed in subsequent teaching in a module teaching the essentials of programming to postgraduate students. Student learning performance, as measured by examination marks on the module, before and after the intersection concepts were emphasised was compared. The student learning performance improvement, together with interviews to students about their perceptions about programming, suggests that teaching oriented to research support is effective at increasing

student understanding of programming in MATLAB.

Introduction

MATLAB is a high-level computer programming language which is powerful, flexible and easy-to-use. MATLAB allows rapid prototyping of ideas and this has led to its wide spread use in research and industry. The ability to programme in MATLAB is therefore highly valued in education, research and industry.

The inclusion of teaching MATLAB programming at an early stage of the undergraduate degree is a widespread practice in engineering and science (Azemi, Bodek & Chinn, 2013; Nyamapfene & Lynch, 2016). In subject areas like psychology programming is usually not a part of undergraduate education though it is increasingly taught to postgraduate students. The increasing take-up of MATLAB programming at the postgraduate level has been facilitated by the publication of more accessible teaching resources aimed specifically at behavioural and cognitive science students (Rosenbaum, Vaughan & Wyble, 2015; Misirlisoy, 2017; Cohen, 2017).

Though the teaching of MATLAB programming has become increasingly popular in universities there has been, nevertheless, relatively little attention paid to

how best to teach what are complicated cognitive processes to students who are often not familiar with programming in general. However, there is evidence that, by fostering the integration of research and teaching practice in higher education settings, student engagement and graduate employability is substantially enhanced (Gresty, Heffernan, Pan & Edwards-Jones, 2015; Hay, Weller & Ashton, 2015). The present study investigated the acquisition of MATLAB programming skills by postgraduate students to see whether this learning was improved by teaching oriented to research support.

Methods

We explored whether teaching key concepts oriented towards research support could improve students' programming skills. We did this using initial pre-teaching intervention questionnaire surveys, interventions in teaching, and by conducting intensive individual interviews of students who had taken the MATLAB programming course. All necessary ethical clearances were obtained from the departmental ethics board.

Surveys

There were two pre-teaching intervention questionnaire surveys. Students were given questionnaire surveys which asked them what programming concepts they found the most difficult and confusing to understand and academic staff were given questionnaire surveys to gauge what they considered the most important knowledge and skills in programming to be.

In the student surveys, students enrolled in the MATLAB programming course were presented with a list of programming concepts from the course. They were asked to circle concepts that they thought were troublesome and to give a difficulty rating (Likert scale where 1 = the least troublesome through to 5 = the most troublesome). These questions are attached (Appendix – Survey questions to students). Student surveys were conducted for the academic years 2013/14 and 2014/15 when the MATLAB programming course was first taught.

In the staff surveys, academic staff was given a questionnaire which is detailed in Appendix – Survey questions to staff.

Experiment

Based on the information provided in the student and staff surveys we were able to identify a subset of concepts (intersection concepts) that were deemed to be important knowledge and skills in programming according to the academic staff, and which were highlighted by students as being areas which they found the most confusing and difficult to learn. We carefully planned to address these subset of concepts in subsequent teaching in a module teaching the essentials of programming to postgraduate students. The intersection concepts would be embedded into students' own research projects or example research projects if students did not have their own research projects to develop during the module. We also made sure that everyone had the chance to talk through the concepts throughout the research projects. These teaching-interventions were applied to student cohorts in academic years 2015/16 and 2016/17.

Interviews

As post-experiment interviews, a few open questions were asked to students about their prior programming experience and current learning experience. The interview questions are included in Appendix – Interview questions.

Results

Surveys

For the academic years 2013/14 and 2014/15 8 student participants were given questionnaires and 6 returned completed questionnaires. Table 1 shows both the average and total ratings of the concepts that were denoted as the most troublesome to students. *function* was determined as the most troublesome concept for the students. *Logical index*, *Loop*, *cell variable*, *branch statement*, *matrix variable*, and *string variable* received similar average ratings but much smaller total ratings than *function*.

Concept	Total	Average
Function	21	3.5
Logical index	14	2.8
Loop	12	4.0
Cell variable	12	4.0
Branch statement	10	3.3
Matrix variable	10	3.3
String variable	10	3.3

Table 1 Troublesome concept ratings

Among 6 research staff that use MATLAB programming, 4 returned completed questionnaires.

Most items of knowledge and skills identified as important by academic staff may include more than one individual item from the concept list as used by students. So it is important to evaluate which concepts contribute to the most important knowledge and skills, even though sometimes it is difficult to disentangle concepts for this purpose. The open-ended survey results from academics were coded and grouped into some concept categories. The important knowledge and skills, confirmed by academics’ research experiences, were as follows:

- To design experiments
- To understand functions/scripts
- To adapt existing scripts
- To create new functions

Typical statements from academic participants stated:

“Understanding the concept of using functions/scripts and how they work/structure. Ability to adapt existing experiment scripts.”

Another one said:

“Being able to understand programme written in MATLAB, being able to modify programmes in MATLAB and being able to program from scratch in MATLAB.”

However, one academic participant argued that functions were too hard to students:

“MATLAB is the wrong programming language for social scientists. I recommend Python and R and a stronger focus on the use of statistical modelling”

Most of the other academics clearly stated that modifying and applying functions were suitable for students in behavioural sciences but functions were not an easy concept to fully grasp.

The results from both student and staff questionnaires indicated that understanding existing functions and creating novel functions were considered both important to academic staff in their research and hard to learn and understand for students.

Experiment

Our teaching was focused on the intersection of areas from the surveys to students and staff, namely the *function* concept (using and writing functions). A familiarity with functions was developed by extensive practice using either students’ own research projects or example research projects.

There were large improvements in examination marks in the years when the teaching intervention was introduced (2015/16 and 2016/17) compared to previous years’ examination marks when there was no teaching intervention (2013/14 and 2014/15).

An independent two-sample t-test was conducted on exam marks before and after the teaching interventions. We predicted that there would be a significant difference between pre- and post-teaching interventions, with higher exam marks for the post-teaching intervention condition. The statistical test results were significant: $t(15)=-1.887, p=0.04$, equal variances assumed (Levene’s Test for Equality of Variances $F(1,15)=.009, p=.926$). The results indicate that exam marks were higher in the post-teaching intervention condition ($M = 68.60\%, SD = 19.34\%$) than the pre-teaching intervention condition ($M = 50.71\%, SD = 19.04\%$).

Interviews

Seven students were recruited to participate in the interviews. Some basic information on

the participants is listed in Table 2. The interviews were up to an hour in length and were recorded. The recordings were transcribed into a text document from which some brief extracts are reported in the Discussion. In this brief report we do not have space to develop a full Interpretative phenomenological analysis (Smith, Flowers & Larkin, 2009), so we only report preliminary findings.

Year of study	Number of participants	Number of participants with own research project
2014/15	2	0
2015/16	2	1
2016/17	3	3

Table 2 Basic participant information

Discussions

The present study is based on teaching interventions in an introductory MATLAB programming module provided for postgraduates. In this module students are assumed to have no prior programming experience and this was usually the case. Students were taught basic concepts in MATLAB programming and how to apply those concepts to develop their own behavioural experiment and to analyse experimental data. Students are presented with many practical examples in their subject contexts. We argue that their perceptions of the most troublesome programming concepts must be strongly linked to their learning motivation and background (purpose of study, learning experience). In other words, students with different learning backgrounds may have different perceptions about what aspects of programming are the most difficult to understand. The type of student enrolling on our module was very different from those students reading Computer Science, for instance, in term of perceptions of what material is the most difficult to learn (Robins, Rountree & Rountree, 2003).

From our questionnaire surveys, it is clear that staff members were interested in the broad application of programming while students were fixated on the details within processes.

This is due presumably to the level of sophistication in the knowledge bases of student and staff – rather like the difference between an experienced driver and the learner driver. However, we feel that the key to academic staff’ perceptions about programming is the use and writing of functions. This is exactly the key point of concern of the students who find functions a particularly troublesome concept. Hence we decided to make the teaching of functions one of the key points upon which we focussed our teaching intervention.

The improvement in student learning (pre-compared to post-teaching intervention), as evidenced by exam mark, suggests that students benefited from a focus on teaching the function concept. Teaching activities oriented to research support are evidently able to address module learning outcomes better because students’ skills in developing behavioural experiments and data analysis were much improved post-teaching intervention. (Key learning outcomes of the module as addressed in marks). MATLAB becomes their “own” research tool to support their research activities. Indirectly, student interviews confirm this as well.

The interviews show that learning MATLAB using research projects provides a very rich learning experience for students. Students were much more motivated than before, especially when they had the opportunity to apply module learning to their own research projects and if they received support from tutors directed to their own projects.

Moreover, the interviews show that students who applied MATLAB to their own research projects during the module were more likely to use MATLAB after the module had finished than students who did not attempt to apply it to their own research during the module. Some interesting student responses about research-oriented teaching were:

"It is quite useful ... We already have a big picture. We can go further/deeper..."

"I switch quite well between learning MATLAB and using it to my own research project. ... But a lot of things I learned very useful to the project I was

doing. ... So it worked quite well, it is directly relevant."

"I am very happy. It really works and you really helped me finish my first research paper during learning MATLAB."

Students' attitudes towards using MATLAB programming in research, have clearly become more positive after the teaching interventions. This is heavily suggestive that research oriented teaching practice is very effective in promoting learning and effective in motivating students.

Using students' own research projects makes students' learning personalised and directly relevant to their own research. This helped motivate the students coming to the module who wished to use MATLAB in their research, as evidenced from interview responses to the question "what does programming mean to you?" This should have significant implications to relevant module and programme design because personalised learning should be promoted whenever it is possible.

Summary

Teaching oriented to research support was shown to be effective at increasing student understanding of programming in MATLAB as measured by exam performance and as reported in post-teaching intervention interviews. Teaching oriented to research support requires tutors to adapt their teaching to address students' own research interests which helps make learning personally relevant to the students.

Acknowledgements

We are grateful for financial support from the School of Life Sciences at the University of Hull.

References

Azemi, A., Bodek, M. & Chinn, G. (2013). *Teaching an Introductory Programming*

Course Using Hybrid E-learning Approach. In: *Frontiers in Education Conference*. IEEE, pp. 1911-1913. DOI: 10.1109/FIE.2013.6685168.

Cohen, M.X. (2017). *MATLAB for Brain and Cognitive Scientists*. Cambridge, MA: MIT Press.

Gresty, K., Heffernan, T., Pan, W. & Edwards-Jones, A. (2015). *Minimising risks in research-informed teaching*. *Higher Education Review*, 48(1), pp. 40-67.

Hay, D.B., Weller, S. & Ashton, K. (2015). *Researcher-led teaching: embodiment of academic practice*. *Higher Education Review*, 48(1), pp. 25-39.

Misirlisoy, E. (2017). *Programming Behavioral Experiments with MATLAB and Psychtoolbox: 9 Simple Steps for Students and Researchers*. New York: Routledge.

Nyamapfene, A. & Lynch, S. (2016). *Systematic Integration of MATLAB into Undergraduate Mathematics Teaching: Preliminary Lessons from Two UK Institutions*. In: *IEEE Global Engineering Education Conference (EDUCON)*. IEEE, pp. 1145-1148. DOI: 10.1109/EDUCON.2016.7474699.

Robins, A., Rountree, J. & Rountree, N. (2003). *Learning and Teaching Programming: A Review and Discussion*. *Computer Science Education*, 13(2), pp. 137-172. DOI: 10.1076/csed.13.2.137.14200.

Rosenbaum, D.A., Vaughan, J. & Wyble, B. (2015). *MATLAB for Behavioral Scientists*. 2nd ed. New York: Routledge.

Smith, J., Flowers, P. & Larkin, M. (2009). *Interpretative Phenomenological Analysis: Theory, Method and Research*. London: SAGE.

Appendix: Survey and interview questions

Survey questions to students

Below is a list of concepts selected from lectures. Some concepts may make you feel troublesome to learn. Please circle all concepts that you think could be troublesome and add a rating (1: the least troublesome ~ 5: the most troublesome). If you think there are some other troublesome concepts which are missing from the concept list, please add them.

Vector variables
Matrix variables
Data format
Data concatenation and conversion
Maths operations on array
Randomisation
Writing scripts (M-files)
Debugging
Basic plotting
Relational operators and logical operators
Logical index
Branch statements: if-else and switch
Loops: for and while
Function
String data type and operations
Cell data type and operations
Structure data type and operations
Interfacing Microsoft Excel file
Cogent toolbox

Survey questions to staff

- What do you think are the most important knowledge and skills in MATLAB when you apply them in your research or you will apply them in the future?
- Are there any suggestions on how to teach MATLAB for psychology postgraduate students?

Interview questions

- 1) Did you have any previous experience of programming before the course?
- 2) What does programming mean to you?
- 3) Did you apply MATLAB in to your own research project during the module? If yes, how was that experience of own research project while learning MATLAB?
- 4) Have you used MATLAB for your own research after the module? If yes, how was that experience? If no, why did you not use MATLAB?
- 5) How useful do you feel that MATLAB module you learnt has been for your own research?