***Multivariate Gradient Analysis for Evaluating and Visualizing a Learning System Platform for Computer Programming***

Richard Mather, Buckinghamshire New University, United Kingdom

The European Conference on Technology in the Classroom 2014
Official Conference Proceedings

**Abstract**

This paper explores the application of canonical gradient analysis to evaluate and visualize student performance and acceptance of a learning system platform. The subject of evaluation is a first year BSc module for computer programming. This uses 'Ceebot', an animated and immersive game-like development environment. Multivariate ordination approaches are widely used in ecology to explore species distribution along environmental gradients. Environmental factors are represented here by three 'assessment' gradients; one for the overall module mark and two independent tests of programming knowledge and skill. Response data included Lickert expressions for behavioural, acceptance and opinion traits. Behavioural characteristics (such as attendance, collaboration and independent study) were regarded to be indicative of learning activity. Acceptance and opinion factors (such as perceived enjoyment and effectiveness of Ceebot) were treated as expressions of motivation to engage with the learning environment. Ordination diagrams and summary statistics for canonical analyses suggested that logbook grades (the basis for module assessment) and code understanding were weakly correlated. Thus strong module performance was not a reliable predictor of programming ability. The three assessment indices were correlated with behaviours of independent study and peer collaboration, but were only weakly associated with attendance. Results were useful for informing teaching practice and suggested: (1) realigning assessments to more fully capture code-level skills (important in the workplace); (2) re-evaluating attendance-based elements of module design; and (3) the overall merit of multivariate canonical gradient approaches for evaluating and visualizing the effectiveness of a learning system platform.

iafor

The International Academic Forum
www.iafor.org

## 1 Introduction

The two aspects of the study reported below concern: (1) the educational context, in this case an investigation of an approach for learning and teaching computer programming; and (2) the primary objective, an evaluation of a novel means for exploring complex data that commonly arise from such multivariate studies.

Regarding the first aspect (the approach taken for introducing programming), it is widely accepted that students find that learning to programming is challenging and an obstacle to progression to later stages of higher education. The paper "Learning and Teaching Programming: A Review and Discussion" by Robins and co-workers (2003) at Otago University, clearly summarises that "Novice programmers suffer from a wide range of difficulties and deficits. Programming courses are generally regarded as difficult, and often have the highest dropout rates".

Experiences at Buckinghamshire New University, where modules in programming underpin computing courses, reflect the findings of Robins *et al*. (*ibid.)* and of others reporting student difficulties in understanding both introductory and higher level programming concepts (for example Milne and Rowe, 2002).

Many learning and teaching applications therefore endeavour to make the subject less intimidating and more accessible to novice programmers through creative use of graphical and interactive development environments or immersive game-like interfaces. Widely used examples of such learning environments include: Alice (Cooper *et al.*, 2000); Lego Mindstorms (Barnes, 2002); BlueJ (Kölling *et al.*, 2003), Greenfoot (Henriksen & Kölling, 2004) and Scratch (Resnick *et al*., 2009).

In this study, students use the Ceebot application, designed for learning industry-standard C-language syntax and object-oriented principles (Huber, 2008; Maragos & Grigoriadou, 2005). Ceebot employs a dynamic landscape populated with robotic devices that may be programmed to interact with each other, 'alien' life and to perform tasks on inanimate objects (see Figure 1).

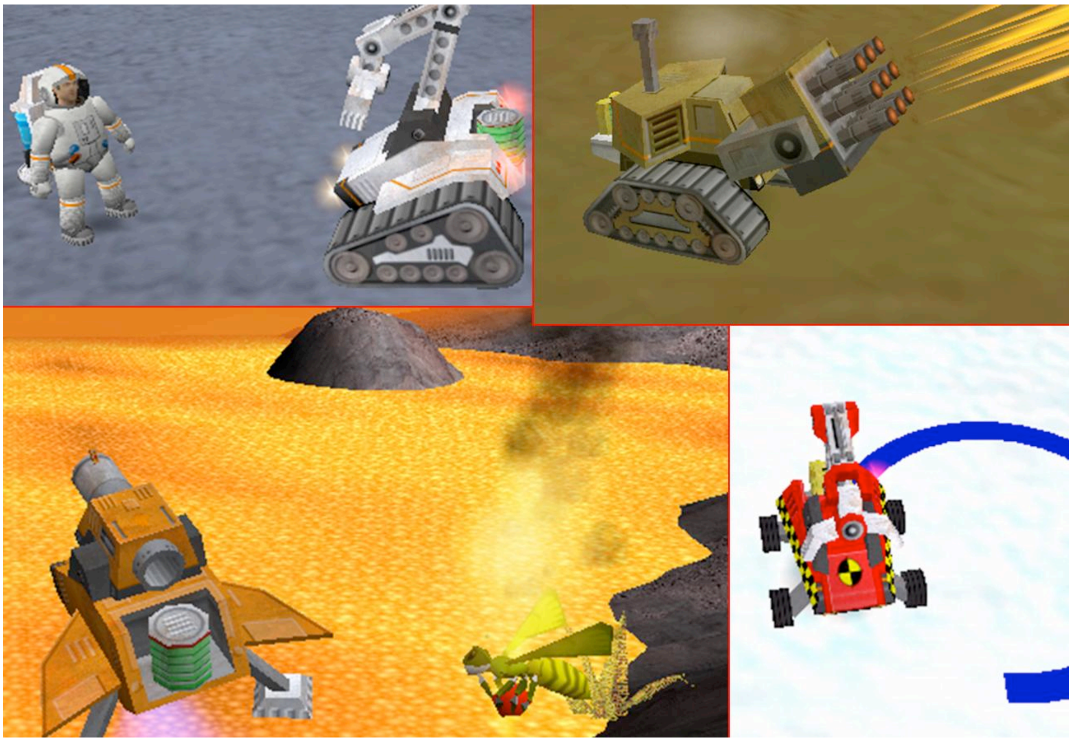

Figure 1 Screen capture from Ceebot showing a small section of 'bots' that may be programmed to move, pick up objects, shoot, fly (bottom left) and draw (bottom right).

Concerning the second aspect of research interest (the means of data analysis), the nature and type of data available is normally determined by the intention of research, the context, the research style and strategy for data collection and analysis (Cohen, Manion and Morrison, 2011). As with many exploratory investigations surrounding the efficacy and acceptance of educational environments, data sets are often unavoidably complex and multivariate as a consequence of response behaviours, potential explanatory variables and interaction effects. Moreover, in exploratory analyses involving questionnaire data, it may be desirable to first screen or filter variables for explanatory power and for collinearity or other redundancy (Cohen, Manion and Morrison, *ibid.)*.

Common exploratory approaches include correlation analysis (Pearson's Product Moment Correlation coefficient and Spearman's Rank Order Correlation coefficient). Although strictly concerned with bivariate relationships both are often used in matrices to explore patterns in multivariate data (Sokal and Rohlf, 1995).

Among gradient-type tools, Principle Component Analysis (PCA) is a true multivariate tool that is widely used for exploratory purposes. Although a useful means for investigating multivariate relationships, the ordination axes describing variation only represent orthogonal directions in the entire data set and are not directly related to explanatory data (Sokal and Rohlf, *ibid.)*.

Similar to PCA, the alternative approach of Canonical Correspondence Analysis (CCA) has the advantage that response scores are regressed on explanatory data, so ordination axes are constrained to explanatory variables. Canonical gradient analysis techniques are widely used by the ecology scientific community. Correspondence analysis (CA) was pioneered by ecologists from the 1970s and found to be well suited to describing unimodal species distributions. Canonical Correspondence Analysis (CCA) was developed by ter Braak for ecological sciences (ter Braak, 1986) and is highly regarded by community ecologists for investigating the distribution and abundance of species along environmental gradients (Šmilauer and Lepš, 2014).

However, CCA assumes unimodal distribution of response variables and is insensitive to direction of relationship. Thus this study evaluates Redundancy Analysis (RDA), first publicised by van den Wollenberg in 1977. RDA possesses two advantages that ordination axes are constrained to explanatory variables and, through applying a linear ordination, does not rely on assumptions of unimodality. It is, in effect, the canonical equivalent of PCA (ter Braak, 1987). Like CCA, RDA is a valued tool among ecologists and environmental scientists. CCA and RDA may also be used in a 'partial' form to filter effects of background variables so that residual variation may be analysed against explanatory factors of interest. One example of this being a study of forest condition in which relationships with atmospheric pollution were analysed after first 'removing' variation in data sets attributable to meteorological effects (Mather *et al.*, 1995).

## 2 Aim and objectives

The primary intention of this investigation is to evaluate a Redundancy Analysis as a multivariate statistical tool for exploring student engagement and performance in a learning environment. As a consequence, this fulfilled a secondary aim of revealing interrelationships between student behaviours, preferences and achievement using the Ceebot environment for learning computer programming.

## 3 Method

First year degree students enrolled on courses in computing, games development and software engineering and attending a module on introductory computer programming were invited to participate in this study. Of a possible eighty students thirty five made fully valid returns (no missing data) for a questionnaire with a combined test and also completed the final module assessment.

The combined test and questionnaire comprised: (1) a self-evaluation of perceived difficulty; (2) tests of commonly used terms/definitions and of code skill and understanding; and (3) twenty questions with Likert scale responses (see Table 1) designed to gauge individual acceptance of the Ceebot environment, preferences, behaviours and approaches to completing work. Likert scales intentionally allowed neutral responses.

Two further variables for module mark and attendance (both as percentages) were included for each student record.

Measures were taken to ensure that participants were willing and consented to recordings. The reasons for study, the ownership, protection and the distribution of information were clearly explained. All findings are published anonymously.

Questionnaire returns were subjected to checks for completeness, accuracy and uniformity, following established recommendations of Moser and Kalton (1977). Data were collated in spreadsheets and, for purposes of canonical analyses with the Canoco 5 application (ter Braak and Šmilauer, 2012), divided into response and explanatory data. Although other statistical packages allow canonical analysis, Canoco 5 was selected for reasons of a dedicated canonical specification that is subject to ongoing research and development. It also offers powerful graphing tools for visualising ordinations (Šmilauer and Lepš, *op. cit.*).

Although a conventional approach might perhaps be to regard that module grades and test scores were 'responses' to predisposing explanatory variables (e.g. motivation, attendance, collaborative inclination, as indicated by questionnaire data), in initial analyses RDA axes were constrained to the key learning performance indices of interest. In other words module grades and test scores were initially reversed to become explanatory variables and questionnaire data became response variables.

Reasons for adopting this 'switched' perspective included that module grades and test results were more representative of true gradients than the limited range of Likert categories. There were also a relatively large number of questionnaire variables for which, in the context of this exploratory study, it was highly probable that many would be unrelated to the learning performance variation of interest. In addition to potentially weak explanatory power, there was also a strong likelihood that much questionnaire variation was intercorrelated and collinear.

Given the landscape ecology origins of canonical and redundancy analysis (in which species and other biological variation is commonly investigated against explanatory environmental gradients, there was also conceptual consistency in this converse view. Thus learning achievements (as indicated by grade and test results) represented positions along gradients in a learning landscape; these positions being in part determined by behaviours, preference, acceptance characteristics.

## 4. Results and Discussion

### *4.1 Central tendency in questionnaire responses.*

Initial screening for central tendency in questionnaires (Table 1) revealed only one item in which the overall response pattern was entirely symmetrically distributed around a neutral mode (Question 18 in Table 1). For all other questions Likert distributions were clearly skewed towards either agreeing or disagreeing to the assertion made. Overall consistent 'polarities' between similar but alternative viewpoints concerning acceptance of Ceebot (e.g. questions 3, 5, 6, 9, 11 and 13) and motivation (e.g. questions 10, 12, 16, 17, 19 and 20), suggested that questionnaires had been completed accurately and diligently.

**Table 1 Summary of frequency of Likert category against questionnaire returns.**

| Question | Strongly Agree | Agree | Neutral | Disagree | Strongly Disagree |
|---|---|---|---|---|---|
| 1. It is very helpful to discuss Ceebot problems with friends. | **21** | 13 | 0 | 1 | 0 |
| 2. It is always possible to find information to complete exercises. | 3 | 11 | **14** | 4 | 3 |
| 3. Ceebot animated environment aids understanding. | 10 | **16** | 7 | 2 | 0 |
| 4. I find it useful to draft designs and algorithms on paper. | 1 | 12 | **13** | 4 | 5 |
| 5. Ceebot does not help me remember fundamental concepts. | 0 | 7 | 3 | **19** | 6 |
| 6. Ceebot is enjoyable. | 6 | **20** | 8 | 1 | 0 |
| 7. No formal lectures are required – just Ceebot notes. | 2 | 7 | 6 | **18** | 2 |
| 8. Like this module to be commercially recognised qualification. | 5 | **18** | 8 | 4 | 0 |
| 9. Ceebot graphics are distracting. | 2 | 2 | 7 | **19** | 5 |
| 10. Un-assessed multiple-choice tests would help with learning. | 3 | **19** | 10 | 3 | 0 |
| 11. It would be quicker to learn to program without Ceebot. | 2 | 8 | 10 | **11** | 4 |
| 12. Easiest way to complete logbooks is to cut and paste code. | 5 | **14** | 11 | 3 | 2 |
| 13. Ceebot is good for learning C-programming for employment. | 7 | **22** | 6 | 0 | 0 |
| 14. Other websites are helpful for completing exercises. | 0 | 4 | 6 | **18** | 7 |
| 15. I'm worried that Ceebot may not help me get a job. | 2 | 6 | **12** | 10 | 5 |
| 16. I only work on Ceebot exercises in practical sessions. | 1 | 2 | 4 | **21** | 7 |
| 17. 2+ hours extra work is needed to complete the week's tasks. | 7 | **22** | 1 | 5 | 0 |
| 18. More exercises than needed to understand concepts covered. | 1 | 9 | **15** | 9 | 1 |
| 19.I work on Ceebot exercises at home. | 14 | **17** | 2 | 2 | 0 |
| 20. I'd like an e-forum to discuss Ceebot problems. | 14 | **15** | 4 | 2 | 0 |

Table 1 Notes: (1) mode category is bold and underlined; (2) questions are abbreviated from full questionnaire form for the purposes and convenience of tabular display.

*4.2 The interpretation of redundancy analyses and ordination diagrams.*

The ordination diagrams presented in Figures 2 and 3 are correlation biplots in which axes are scaled to unit length and increment (ter Braak, 1992; Šmilauer and Lepš, *op. cit.*). Response variables are represented by blue arrows (or vectors) and explanatory variables are represented by red arrows. The length of arrows is proportional to their standard deviations and the cosines of their angular separations between each other and the axes (regardless of whether explanatory or response variables) corresponds to their correlation coefficients, i.e. $r \approx \cos\Theta$ (Corsten and Gabriel, 1976; ter Braak, 1987; Šmilauer and Lepš, *op. cit.*). Thus perpendicular relationships between response and explanatory arrows and axes (i.e. approximating to cosine $90^0$) indicate near zero correlation ($r \approx 0$) while parallel relationships (whether in same or opposing directions) represent correlations approaching unity (cos $0^0$ or $180^0$ corresponding to $r$=1 or -1 respectively).

Summarising, the heads of arrows indicate the direction of maximum variation in the value of corresponding variable. The longer an arrow the greater the importance of the variable effect in the model and also the greater the confidence in the inferred correlation (ter Braak, 1987; ter Braak and Prentice, 1988). Variable arrows and ordination axes in the same direction are positively correlated, perpendicular vectors are not correlated and those pointing in opposing directions are negatively correlated.

It is important to note that a $180^0$ shift in correlation polarity may simply reflect that a questionnaire item is expressed with a negative rather than a positive assertion.

In tables of summary statistics (Tables 2 and 3) entries are only made for the first two axes because these describe the great majority of explainable variation in response data. The first row states eigenvalues. These express the proportion of all variation (unity) explained by an axis; hence their equivalence to percentage expressions for cumulative variation on the second row. The pseudo-canonical correlations on the third row express the correlation between response based and explanatory-variable based axes (Šmilauer and Lepš, *op. cit.*). The final entries for explained fitted variation are only concerned with variation described by the model and express the proportion explained by the axis concerned.

Notes following summary statistic tables describe: (1) the total response variation explained by explanatory variables and an adjusted figure to compensate for inflatory bias due to small sample sizes (Šmilauer and Lepš, *ibid.)*; and (2) a pseudo-*F* statistic is derived and may be interpreted in the same way as in ANOVA of the regression model (Šmilauer and Lepš, *ibid.)*. The probability *P* is derived from a Monte Carlo permutation test. This involves random permutation of response data with respect to explanatory variables. Thus, if after 999 permutations, 43 random permutations produced eigenvalues greater than that for the original data, *P* would be (43+1) / (999+1) = 0.044.

*4.3 Findings from Exploratory Redundancy Analyses.*

The result of redundancy analysis of all data (using the 'converse' view that grade and test achievement variables represented gradients that explained distributions of behaviours, preference, acceptance responses) is represented by the ordination of Figure 2, with summary statistics presented in Table 2.

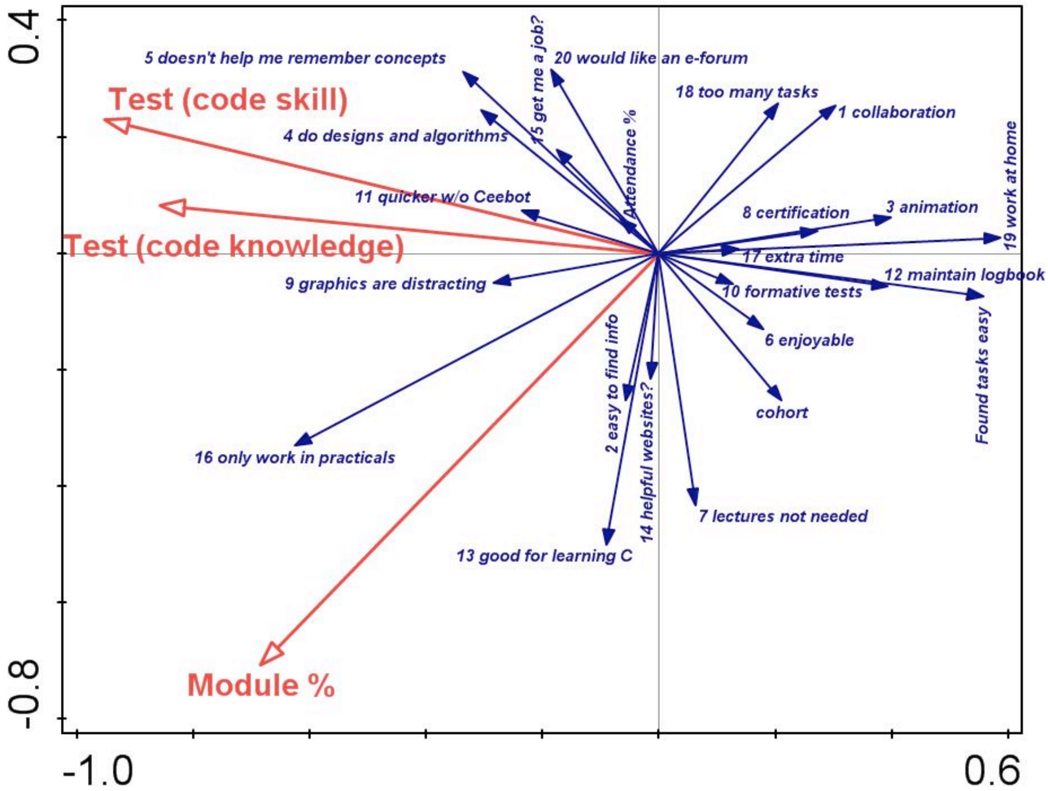


Figure 2. Ordination Biplot for the Redundancy Analysis of student behaviours, preference, acceptance responses (as indicated by blue arrows representing questionnaire returns and class attendance) against positions along learning performance gradients (as indicated by red arrows indicating module grade and tests of programming knowledge and coding skill).

The first (horizontal) axis describes most variation in response variables at approximately 9% (Table 2 eigenvalue 0.0915 and cumulative percentage 9.15%). The direction and length of the two "Test (code ... )" vectors indicate their overall influence on the first axis and relative effectiveness in describing the greater proportion of explainable response variation. The second axis is uncorrelated to the first axis, apparently most strongly influenced by Module % but weakly related to the two "Test (code skill/knowledge)" vectors. This axis describes a further 5% of variation in response data. Table 2 coefficients ~0.85 and ~0.70 express that the correlation between response based and explanatory-variable based axes are highly significant. The Monte Carlo permutation test confirms the overall significance of the model ($p$=0.044).

It is clearly evident that the two "Test (...)" items explain most variation in response (behaviour) data, although the overall module grade (Module %) is also strongly related to response variation in both first axis and the orthogonal second axis**.** Those response variables most strongly related to explanatory variation of interest included behaviours of independent study and homework (16 - only work in practical sessions; 19 - work at home) and peer collaboration (1), but surprisingly weakly associated with 'Attendance'. The latter is the shortest and least significant vector in the entire model. Further investigation revealed that this apparent anomaly may be partly attributed to a small group of students with advanced subject knowledge who did not attend regularly.

**Table 2 Summary statistics for Redundancy Analysis and Ordination presented in Figure 2.**

| *Statistic* | *Axis 1* | *Axis 2* |
|---|---|---|
| Eigenvalues | 0.0915 | 0.0516 |
| Explained all variation (cumulative %) | 9.15 | 14.31 |
| Pseudo-canonical correlation | 0.8523 | 0.6975 |
| Explained fitted variation (cumulative %) | 54.04 | 84.52 |

Table 2 Notes: (1) Explanatory variables account for 16.9% all variation (adjusted explained variation is 4.5%); (2) Permutation Test Results (on all axes): pseudo-$F$=1.4; $P$=0.044.

Explanatory variables (red in Figure 2) were clearly effective in describing response variation. However, the orthogonal relationship between the key element of assessment (Module %) and the un-assessed tests on code understanding and skill, unexpectedly suggested that strong module performance was not necessarily a reliable predictor of programming ability. This finding was clearly of pedagogic concern. A simple correlation check (Pearson product-moment) also suggested that although "Test (code understanding)" was significantly correlated with "Module %" ($r$=0.56, $p$<0.001), "Test (code skill)", was not correlated with overall assessment grade ($r$=0.27, $p$<0.117).

There were strong correlations between "commitment" indicators (16 "only work in practical - disagree", 19 "work at home – agree" and 12 "maintaining logbook - agree").

Among other exploratory patterns of interest was the correlation between response 13 ("good for learning C type languages") and the second axis, as well as a strong relationship to overall "Module %" and the evident orthogonal relationship with "Test (code skill)". This indicated that appreciation of Ceebot may not have been so strongly expressed by more adept programmers. Similarly, the strong negative correlation between the indicator for collaboration (stated as "It is very helpful to discuss Ceebot problems with friends" in the questionnaire and labelled "1 collaboration" in Figure 2) and "Module %" suggested that such behaviour was more greatly valued by those achieving high overall module grades than those who were 'purely' proficient at programming.

Although there are other correlations and patterns deserving of pedagogic attention, the aforementioned represent the most significant and, with respect to this analysis, are perhaps within limits of model interpretation.

The proportion of overall variation explained by learning performance indicators may appear low. However, this is not surprising given the exploratory nature of questionnaire items and the fact that some questions will unavoidably introduce variation that is unrelated to learning performance.

Among objective measures used to filter such extraneous variation was stepwise forward selection of factors explaining most variation in learning performance. To do this it was necessary to adopt a conventional view that the learning performance indicators “Module %”, “Test (code skill)” and “Test (code understanding)” were responses to explanatory characteristics (behaviours, preference and acceptance characteristics) expressed in the questionnaire.

The following analysis therefore uses forward selection and also excludes variables that contribute little to the overall model, such as “Attendance”, or are redundant through collinear/inter-correlation with other variables, for example “Test (code understanding)”. It also observes Canoco’s over-fitting alert, based on termination criterion of Blanchet *et al*. (2008). This suggests when further stepwise inclusion is unadvisable on the grounds that adding another predictor would increase the $R^2$ (adjusted) to a value greater than that would be otherwise obtained by fitting the full model with all predictors.

After removing obvious sources of collinearity, stepwise selection observing termination criterion resulted in a simpler model (Figure 3) with only six questionnaire items. A reduction in ‘unexplainable’ variation is partially responsible for increased eigenvalues and overall variation explained by the model (Table 3 indicates the first axis accounts for approximately 35% of overall variation and the second axis accounts for some further 13%). However, a direct comparison should not be made with the earlier RDA due to the changed focus of analysis. The overall model is highly significant (Table 3 notes: pseudo-$F$=15.7; $P$=0.002).

The alignment of item 19 (full form, “I work on Ceebot exercises at home”) with the first axis and the two indicators of learning achievement, and the overall length of vector suggests that this is the most important item describing variation in learning performance. This is confirmed by the statistics for stepwise regression (Table 4) that indicate item 19 alone accounts for more than 33% of the explainable variation and that its contribution is also highly significant (pseudo-F 6.5, $P$=0.004).

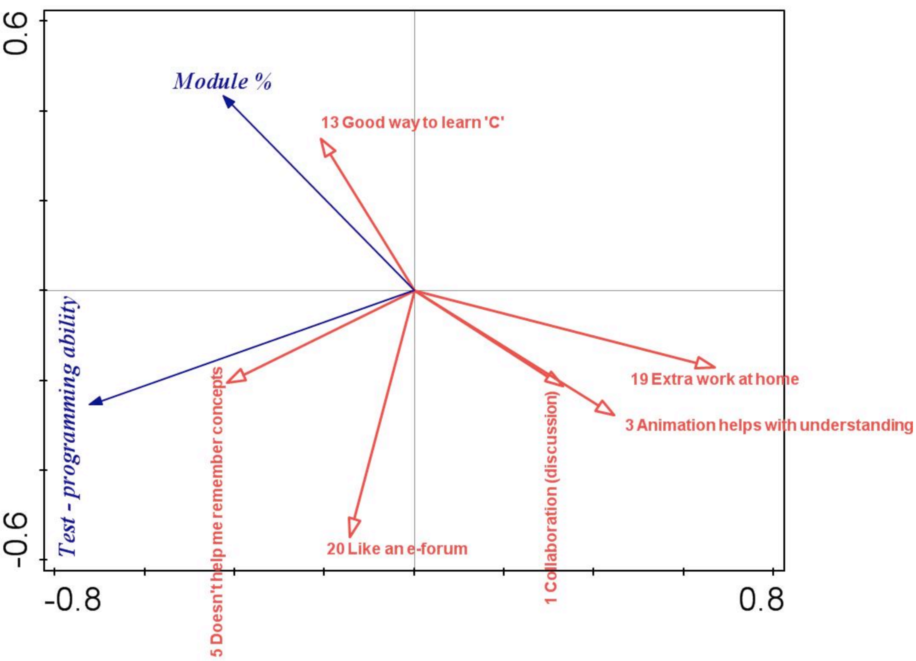

**Figure 3 Ordination Biplot for Redundancy Analysis of learning performance responses (blue vectors) against step-wise selections of behaviours, preference and acceptance (red arrows). Notes:** (1) response and explanatory variables are 'switched' in relation to the earlier analysis so that learning performance are responses and questionnaire variables are explanatory, thereby allowing stepwise inclusion of key variables and elimination of collinear or otherwise redundant terms; and (2) stepwise selection is terminated on according to Canoco 5's internal over-fitting warning based on criterion of Blanchet *et al*. (2008).

Of items remaining after stepwise selection, 13 "Ceebot is good for learning C ...", 20 "I'd like an e-forum ...", 5 "Ceebot doesn't help me remember concepts ..." and "3 Ceebot environment aids understanding ..." each account for some 14-15% of explainable model variation (Table 4.). The last item included in stepwise selection, "1 Collaboration (discussion)" in Figure 3, accounts for a smaller proportion of model variation.

The alignment of Ceebot acceptance indicators (item 5, disagreeing that Ceebot doesn't help with remembering concepts and the test of programming ability; item 3 agreeing that Ceebot animation assists understanding and a balanced alignment between both indicators of learning performance) suggests overall appreciation in Ceebot as a learning platform.

**Table 3 Summary statistics for Redundancy Analysis and Ordination presented in Figure 3.**

| *Statistic* | *Axis 1* | *Axis 2* |
|---|---|---|
| Eigenvalues | 0.3517 | 0.1260 |
| Explained all variation (cumulative %) | 35.17 | 47.77 |
| Pseudo-canonical correlation | 0.7494 | 0.5806 |
| Explained fitted variation (cumulative %) | 73.63 | 100.00 |

Table 3 Notes: (1) Explanatory variables account for 47.8% all variation (adjusted explained variation is 37.0%); (2) Permutation Test Results (on all axes): pseudo-$F$=15.7; $P$=0.002.

Two items suggest the value of measures for peer communication and collaborative working (20 "I'd like an e-forum ..."; 1 "Helpful to discuss Ceebot tasks with friends ..."). Item 13 is somewhat anomalous in that no one disagreed that Ceebot is good for learning C (i.e. the entire range of Likert responses were only in categories 1, 2 and 3), thus its alignment with Module % suggests that respondents generally agreed or were neutral but didn't 'strongly agree' with this assertion.

**Table 4 Summary statistics for Predictors included in Stepwise Forward Selection in the Redundancy Analysis and Ordination presented in Figure 3.**

| *Variable* | *Variation Explained %* | *Model Contribution %* | *Pseudo-F* | *P* |
|---|---|---|---|---|
| 19 I work on Ceebot exercises at home | 16.1 | 33.6 | 6.5 | 0.004 |
| 13 Ceebot is good for learning C ... | 6.7 | 14.0 | 2.8 | 0.084 |
| 20 I'd like an e-forum ... | 6.9 | 14.5 | 3.1 | 0.064 |
| 5 Ceebot doesn't help me remember concepts ... | 7.6 | 15.8 | 3.7 | 0.028 |
| 3 Ceebot environment aids understanding ... | 6.8 | 14.2 | 3.6 | 0.018 |
| 1 Helpful to discuss Ceebot tasks with friends ... | 3.8 | 7.9 | 2.1 | 0.128 |

## 5. Conclusions and Recommendations

With respect to the primary aim, "to evaluate a Redundancy Analysis as a multivariate statistical tool for exploring student engagement and performance in a learning environment", findings indicate that RDA was appropriate and useful for describing patterns of student behaviour and preferences associated with measures of 'success'. The canonical facility to directly focus or constrain analysis to gradients of interest, combined with powerful biplot visualisation of variable influence, vector association and collinear effects provide the researcher with a robust method for identifying pedagogically meaningful influences. In this study, centred on the Ceebot learning environment, RDA was found to be effective in screening indicators and behaviours that may be predictors of learning performance and of acceptance of the environment.

The secondary aim, "of revealing interrelationships between student behaviours, preferences and achievement using the Ceebot environment for learning computer programming", was only achievable because both conditions were satisfied that: (1) RDA was demonstrated to be an appropriate form of analysis; and (2) that most questionnaire items were, to greater or lesser extents, valid predictors for the measures of learning performance.

Although a number of patterns of pedagogical interest were noted, key and significant findings were: (1) the weak association between overall module assessment and the computer programming skill ability; and (2) the three indices of learning performance were correlated with behaviours of independent study and peer collaboration but not with attendance. From a perspective of teaching practice, results indicated that logbook based assessments may need to be revised to more closely align with programming skills valued in the workplace. There was also some evidence that requirements for attendance may be reconsidered, perhaps relaxed for experienced computer-programmers able to demonstrate prior learning. Additionally, opportunities for collaborative learning (discussion) were valued and positively associated with learning performance. From a learner perspective RDA revealed that the key predictor of success was commitment to continue module work outside timetabled sessions.

Regarding ongoing work and recommendations for readers interested in using canonical tools, the canonical viewpoint of learner behaviours determining positions along landscape gradients of performance is novel perspective. This, however, requires further investigation to determine whether such a paradigm may aid in detecting and encouraging behavioural transformations that 'predict' success. It is anticipated that research will continue to use RDA to monitor effects of modifications to teaching practice. One such planned modification is the inclusion of formative tests to develop code-skills.

## 6. Acknowledgements

This study was made possible through the generous and diligent participation of colleagues and students delivering and attending BSc modules in "Programming Concepts" and "Application Programming" at Bucks New University. The research was funded by Bucks New University.

## 7. References


Barnes, D.J. (2002) Teaching introductory Java through Lego Mindstorms models. *Proceedings of the 33rd SIGCSE technical symposium on computer science education*, 147-151.

Blanchet, F.G., Legendre, P. & Borcard, D. (2008) Forward selection of explanatory variables. *Ecology 89* (9), 2623-2632.

Cohen, L., Manion, L. & Morrison, K. (2011) *Research Methods in Education* (seventh edition). Routledge, UK.

Cooper, S., Dann, W. & Pausch, R. (2000) Alice: a 3-D tool for introductory programming concepts. *Proceedings of the 5th Annual CCSC Northeastern Conference*, Mahwah, NJ, 107-116.

Corsten, L.C.A. & Gabriel, K.R. (1976). Graphical exploration in comparing variance matrices. *Biometrics 39*, 159-168.

Henriksen, P. & Kölling, M. (2004) Greenfoot: combining object visualisation with interaction. *ACM Conf. on Object Oriented Prog. Systems Languages and Applications*, 73-82. DOI=http://doi.acm.org/10.1145/1028664.1028701.

Huber, M. (2008) *Bemerkungen für Lehrpersonen* (Notes for teachers). [PDF file]. Retrieved February 21, 2014, from http://www.ceebot.org/index.php?option=com_remository&Itemid=53&func=fileinfo&id=6.

Kölling, M., Quig, B., Patterson, A. & Rosenberg, J. (2003) The BlueJ System and its Pedagogy. *Computer Science Education, 13* (4), 249-268.

Maragos, K. & Grigoriadou, M. (2005) Towards the design of Intelligent Educational Gaming Systems. *Proceedings of the AIED 05 WORKSHOP 5*, 35-38.

Mather, R.A., Freer-Smith, P. & Savill, P. (1995) Analysis of the changes in forest condition in Britain 1989 to 1992. *Forestry Commission Bulletin 116*. HMSO, London.

Milne, I. & Rowe, G. (2002) Difficulties in Learning and Teaching Programming – Views of Students and Teachers. *Education and Information Technologies 7* (1), 55-66.

Moser, C. & Kalton, G. (1977) *Survey Methods in Social Investigation* (second edition). Heinemann, London.

Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. & Kafai, Y. (2009) Scratch: Programming for all. Comm. *ACM 52* (11), 60–67.

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

Šmilauer, P. & Lepš, J. (2014) *Multivariate Analysis of Ecological Data using CANOCO 5*. Cambridge University Press.

Sokal, R.R. & Rohlf, F.J. (1995) *Biometry: The Principles and Practices of Statistics in Biological Research* (third edition). WH Freeman and Co., New York.

ter Braak, C.J.F. (1986) Canonical correspondence analysis: a new eigenvector technique for multivariate direct gradient analysis. *Ecology 67* (5), 1167-1179.

ter Braak, C.J.F. (1987) Ordination. In Jongman, R.H., C.J.F. ter Braak and O.F.R. van Tongeren, editors. *Data Analysis in Community Ecology*. Pudoc, Wageningen, The Netherlands.

ter Braak, C.J.F. (1992) *CANOCO - a FORTRAN program for Canonical Community Ordination*. Microcomputer Power. lthaca. New York.

ter Braak, C.J.F. & Prentice, I.C. (1988) A theory of gradient analysis. *Advances in Ecological Research 18*, 271-317.

ter Braak, C.J.F. & Šmilauer, P. (2012) *Canoco Reference Manual and User's Guide: Software for Ordination (Version 5.0)* Microcomputer Power www.canoco.com.

van den Wollenberg, A.L. (1977) Redundancy analysis. An alternative for canonical correlation analysis. *Psychometrika 42*, 207-219.