

2018

Programming: Predicting student success early in CS1. A re-validation and replication study

Keith Quille

Technological University Dublin, keith.quille@tudublin.ie

Susan Bergin

Maynooth University, Susan.Bergin@MU.ie

Follow this and additional works at: <https://arrow.tudublin.ie/ittsciart>



Part of the [Educational Methods Commons](#), and the [Other Computer Sciences Commons](#)

Recommended Citation

Keith Quille and Susan Bergin. 2018. Programming: predicting student success early in CS1. a re-validation and replication study. In Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE 2018). Association for Computing Machinery, New York, NY, USA, 15–20. DOI: 10.1145/3197091.3197101

This Article is brought to you for free and open access by the School of Science and Computing (Former ITT) at ARROW@TU Dublin. It has been accepted for inclusion in Articles by an authorized administrator of ARROW@TU Dublin. For more information, please contact arrow.admin@tudublin.ie, aisling.coyne@tudublin.ie, vera.kilshaw@tudublin.ie.

Programming: Predicting Student Success Early in CS1. A Re-validation and Replication Study

Keith Quille

Dept. of Computing, Institute of Technology Tallaght
Dept. of Computer Science, Maynooth University
Dublin/Kildare, Ireland
Keith.Quille@it-tallaght.ie

Susan Bergin

Maynooth University
Co Kildare, Ireland
Susan.Bergin@mu.ie

ABSTRACT

This paper describes a large, multi-institutional revalidation study conducted in the academic year 2015-16. Six hundred and ninety-two students participated in this study, from 11 institutions (ten institutions in Ireland and one in Denmark). The primary goal was to validate and further develop an existing computational prediction model called Predict Student Success (PreSS). In doing so, this study addressed a call from the 2015 ITiCSE working group (the second "Grand Challenge"), to "systematically analyse and verify previous studies using data from multiple contexts to tease out tacit factors that contribute to previously observed outcomes".

PreSS was developed and validated in a longitudinal study conducted over a three year period (twelve years previous from 2004-06). PreSS could predict with near 80% accuracy, how a student would likely perform on an introductory programming module. Notably this could be achieved at a very early stage in the module. This paper describes a revalidation of the original PreSS model on a significantly larger multi-institutional data set twelve years after its initial development and looks at recent research on additional factors that may improve the model. The work involved the development of a fully automated end-to-end tool, which can predict student success early in CS1, with an accuracy of 71%. This paper describes, in detail the PreSS model, recent research, pilot studies and the re-validation and replication study of the PreSS model.

CCS CONCEPTS

• **Social and professional topics** → **Computer science education; CS1;**

KEYWORDS

Computer Science Education; Programming; Success; CS1.

ACM Reference Format:

Keith Quille and Susan Bergin. 2018. Programming: Predicting Student Success Early in CS1. A Re-validation and Replication Study. In *Proceedings of 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '18)*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3197091.3197101>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ITiCSE'18, July 2-4, 2018, Larnaca, Cyprus

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5707-4/18/07...\$15.00

<https://doi.org/10.1145/3197091.3197101>

1 RESEARCH POSITIONING

In 2015 an ITiCSE working group identified a "critical need" for re-validation studies in the space of educational data mining and learning analytics [21]. The working group reported that the majority of the studies reviewed focused on simplistic metric analysis and are conducted within a single institution and a single course. This highlights a critical need for validation and replication to better understand the various contributing factors and the reasons why certain results occur in computer science education [21]. The working group concluded with several Grand Challenges, the second of which was to systematically analyse and verify previous studies using data from multiple contexts to tease out tacit factors that contribute to previously observed outcomes [21]. This paper contributes to the CSEd community and the working groups call (in particular the second grand challenge), by systematically analysing several studies spanning over a decade, and revalidating a computational prediction model, twelve years after it was developed, on a modern disparate multi-institutional data set.

2 LAYING THE FOUNDATIONS

Over twelve years ago, a detailed study on factors that influence success in CS1 was presented by Bergin at SIGCSE'05, in St. Louis, Missouri, USA [5]. Subsequently this led on to the development of a computational model that could predict student success with 77.5% accuracy (at a very early stage) in CS1, named PreSS (Predict Student Success) [3, 6]. PreSS has since been successfully used locally, but the model required paper based data collection with manual processing and computation. This time consuming process inhibited the uptake and usage of the model on a large scale.

Prior to the present study, three significant pieces of work were conducted, which laid the foundation for this paper. Initially a small limited scope justification study re-validated PreSS ten years after it was developed, on an independent modern data set. This led to the development of a fully automated end-to-end system with PreSS at its core, named PreSS#. This allowed for real time predictions and subsequently the expansion and uptake of PreSS. The third piece of foundational work, examined additional factors collected from two sources, that improved the accuracy of PreSS or in some way added value to the model. A discussion on these three bodies of work, along with the PreSS model is presented later in this section.

A substantial test of whether a model is generalizable (like PreSS) is to examine if it can achieve similar results on a large disparate multi-institutional data set many years after its creation. This re-validation of PreSS is the primary goal of this paper, while also making a significant contribution to the ITiCSE working group's

second Grand Challenge. The secondary goal of this paper (while also contributing to the second Grand Challenge) was to examine factors not used in PreSS model (identified in recent studies), to determine if the model can be improved.

2.1 The PreSS Model

PreSS was developed twelve years previously by three studies at four different institutions [3, 5, 6]. From an initial set of twenty five factors, the PreSS model uses three factors to predict student success, specifically, programming self-efficacy, mathematical ability based on a high school mathematics exit examination and number of hours per week a student plays computer games [3, 5, 6]. Detailed information on the factors, factor selection, data processing and the machine learning algorithm can be found in the references [3, 5, 6]. Six machine learning algorithms were examined in the development of PreSS and naïve Bayes was selected as it was found to have the highest prediction accuracy [3]. A decade later, a follow up study also examined multiple machine learning algorithms such as logistic regression, SMO (support vector machine), back propagation (artificial neural network) and C4.5 (decision tree) [4]. This study again confirmed that naïve Bayes was the most suitable machine learning algorithm for use in the PreSS model. PreSS was able to achieve an accuracy of 77.5% [3], with $n = 102$. In-fact using a variation of factors, higher performance could be achieved for a specific sample set, for example by gender or per institution, however Bergin's goal was to produce the most generic model possible to maximise generalizability.

2.2 Justification Study

Before any investment of development time was made, a small justification study was carried out to examine if PreSS was still an accurate predictor of programming success [30]. The study used two small independent data sets, collected in the academic years 2013-14 and 2014-15. The institution used was similar to one of the institutions used in the original PreSS study, a Community College. A further test of the generalizability of the PreSS model, was the programming language used in the justification study, C#. This was not represented in the original PreSS study as it was not commonly used at that time.

Due to the relatively small samples sizes ($n = 34$, $n = 26$ respectively), to reduce over-fitting the PreSS model was trained using the original 102 student data samples [3] in 2005. This ensured that the model could not simply learn the training set, it had to learn the underlying relationships. This also mimicked the use of PreSS in a real world situation, when a training set exists and the model has to compute a prediction without ever seeing the test set. PreSS was able to predict with a statistically similar accuracy ($< 1\%$ difference) to that a decade previous [30]. Although these results were very promising, given the small sample size and the fact that it only represented a single institution, a large study would be required to re-validate the model and ensure its generalizability. To do this a more streamlined data collection and prediction process was required.

2.3 PreSS#

In 2015 the PreSS model was developed into a web-based educational system named PreSS# [32]. PreSS# was able to predict student

success in real time, is scalable and uses a software as a service (SaaS) model. The system is fully automated allowing institutions to create users, run the prediction and examine outputs. The user interface and institution/ instructor interaction was further developed into a visualisation engine and analyser thus adding to PreSS# a streamlined user interface, an easy acquisition process, automatic modeling and reporting tools [14]. To ensure PreSS# was producing comparable results to that of PreSS, the original data set was used for validation. Both PreSS and PreSS# produced the same results, with no significant differences found [32], thus validating the online system.

2.4 Additional PreSS Factors

A study conducted in 2016 investigated additional factors that may increase the accuracy of PreSS or in some way add value to the model [31]. Two sources were examined; the first source consisted of factors gathered during the initial development of PreSS that were not used in the final model. The second source consisted of additional factors collected in the re-validation pilot study [31]. The work successfully identified 16 factors that when used in combination or substitution with the original PreSS factors either produced significant increases in prediction accuracy or were otherwise worthy of note. Further details on the study methodology and its findings can be found in the references [31]. These 16 additional factors were included in the revalidation study described in this paper.

3 LITERATURE REVIEW

A systematic approach was taken for the literature review. Over the years there has been a significant amount of research conducted in the space of introductory programming modules and predicting student success using educational data mining techniques. Thus an approach that would ensure the identification of relevant research, in the large quantity available would be required. Furthermore, as the focus of this study was the revalidation of previous educational data mining studies and not learning analytical studies, learning analytics was excluded from the search criteria. Search terms were identified that included: predicting, predict, CS1, introductory programming, factors, ability, performance, success, failure and student. Combinations of these terms were then searched in the ACM and IEEE databases with some additional searching in Google Scholar. Some searches returned hits in excess of 70,000 results. In the case of large search returns, the first 200 results were reviewed, where results were filtered on relevance to the search term. Where the search returned less than 200 results, all the results were reviewed. In total 1,884 articles were reviewed. From there, articles were short listed, based on their relevance to CS, factors and/or prediction models. This resulted in 82 articles (when repeating articles were removed due to being returned in multiple searches). From there a detailed analysis of each article and its relevance was conducted. Thus ensuring that the articles were predicting and/or examining factors for CS1 or other introductory programming courses (and not for example a business course). This final selection process resulted in 40 articles that were included in this literature review.

The main goal of the original PreSS model was to be generalizable. To accomplish this several criteria had to be met. These were: multi-institutional, longitudinal, large sample size, high prediction

accuracy, high sensitivity and early prediction accuracy. Many of these criteria were also highlighted in the ITiCSE working group report [21]. All of the 40 articles examined, exhibited one or many (but not all) of these criteria and have been grouped and presented under these headings to give the reader a sense of how the research available spans the criteria. As the foundational studies were already presented and discussed in detail in section 2, they will be omitted from the literature review.

3.1 Multi-Institutional

A criteria for a generalizable prediction model, would be that it was tested over several institutions, preferably in diverse districts or even countries. From the 40 articles examined, two were conducted in different institutions [8, 15]. It should be noted that two other studies were trained using the data from several other institutions, but only tested on a single institution [9, 37].

3.2 Longitudinal

Studies are often conducted once, on a single cohort. It is acknowledged that CS is a changing landscape, and studies should be repeated over several years, to examine if they stand the test of time. The literature review found that only four studies were conducted over more than one year or semester [1, 17, 26, 41]. This is concerning, if several prediction models, exhibit a strong prediction accuracy and value to the CSEd community, but were never revalidated.

3.3 Large Sample Size

For testing a prediction model, a reasonable sample size is required, a large sample size is desirable. Several studies suffered with relatively small samples sizes, 25 to 90 students or did not include the sample size at all. This may pose problems with over-fitting. Some models in an effort to combat the small sample size, while trying to minimise over-fitting (a common problem with small data sets), used methods like bootstrapping. In several cases with a large sample size, there was no model, just correlations reported. These have value, but unless they are developed into a final model, may not serve the practitioners in any useful immediate way, requiring more development. A positive finding was that 20 of the articles reported a sample size greater than 90 students [2, 8–10, 13, 16, 17, 19, 20, 22–26, 28, 34, 36, 39, 41, 42].

3.4 Prediction Accuracy

Several of the articles, reported no significant prediction accuracies or correlations. This was also concerning, as this whittled the list down considerably. A prediction slightly higher than that of chance was selected as a search criteria. A similar correlation coefficient was selected so not to rule out this research. Twelve articles reported significant prediction accuracies, although some did not predict early in CS1 [1, 7, 15, 19, 24, 26, 38–42]. All articles irrespective of time of prediction, were included in the literature review as they may give insight, into factors not considered in this research.

3.5 Prediction Timing

In addition to prediction accuracy, prediction timing examined the time in which the prediction could be made. The ideal timing is

the earliest possible point into CS1, thus allowing educators to implement interventions in a timely manner. A very positive finding in the literature is that some models were able to predict before the commencement of CS1. The limit for this sub group of articles was less than approximately 25% of the module completion, with 16 articles meeting this criteria [1, 2, 8–12, 15, 16, 19, 20, 27–29, 35, 38]. An interesting subset of prediction timing, is the intersection set of prediction accuracy and prediction timing. This resulted in only four articles [1, 15, 19, 38].

3.6 Date of Study

While reviewing the literature, and the dates of the studies (ranging back as far as 1975) from the 40 articles, a natural clustering of the dates became apparent. This unusual and anecdotal finding, found three natural clusters of predicting student success in CS1 research. The initial cluster centred around the early 1980's [2, 9, 11, 19, 20, 22, 24, 41], the second cluster was centred around the 2000's [7, 8, 12, 15, 34, 42] and the third cluster gathers in the past four years or so [1, 10, 13, 17, 23, 25–29, 35, 36, 38–40]. There is one exception [16], but one could assume that research in predicting success of failure in CS1 cycles almost every two decades. One finding that is noteworthy in the research, is that no study attempted to validate from one cluster to another. The studies did not seem to egress past their own cluster. This is disappointing when ones focus is to develop a model that performs well over time.

3.7 Prediction Sensitivity & Specificity

Prediction models are often presented with a high accuracy, but that alone does not always reveal the entire story of the model. For example, if 90% of the students in this study were strong. The model could predict every student as strong and report an accuracy of 90%. This on the face of it is a very successful model, but its ability to identify weak students (the models main goal) would in-fact be 0% as it predicted every student strong, so the 10% of students that were weak, were incorrectly identified. This is the measurement of sensitivity (discussed in detail in section 5), which measures the proportion of weak students correctly identified; which in this case was 0%. Specificity on the other hand is the proportion of strong students correctly identified, which in this example would result in 100%. Only three studies reported this additional information (in some cases indirectly, but it could be calculated) [7, 19, 26].

3.8 Revalidated Prediction Models

Models are more often than not, never revisited. In all of the literature reviewed, the authors could only find two such instances (the authors even went to the extent to email some of the authors directly to confirm if this was the case, if the article stated a possible follow up study).

Dehnadi

In 2006, Dehnadi developed a prediction model, that reported a 100% accuracy (also 100% sensitivity and specificity) [15]. This work seemed to be a breakthrough. It was disclosed at the the PPIG (Psychology of Programming Interest Group) workshop in 2006. Dehnadi, looked at several introductory programming groups while completing questions. From this Dehnadi built a mental model that

could predict success with 100% accuracy ($n = 60$ students). Based on this reported accuracy, two follow up studies were completed. In 2007, Caspersen repeated the study using approximately 142 students [12]. It should be noted at this point that Dehnadi completed the study in the UK, whereas Caspersen conducted the study in Denmark. The findings of Caspersen's work is best described in the abstract: "We have repeated their test in our local context in order to verify and perhaps generalise their findings, but we could not show that the test predicts students' success in our introductory programming course". Subsequently a study by Bornat in the following year (2008, co-authored by Dehnadi) examined six experiments, with more than 500 students, across six institutions and three countries [8]. Bornat reported that "the predictive effect of our test has failed to live up to that early promise".

Fowler & Glorfeld

In 1981, Fowler and Glorfeld, developed a predictive model using a sample size of 151 students in an introductory programming course [18]. The model was developed using three pools of data: personal, academic and aptitude. From this data a classification model was developed using the logistic discrimination model. The model produced an accuracy of 80.8% and not only was the accuracy presented but also the sensitivity and specificity. For identifying weaker students the model was 76.6% accurate. A year later Fowler and Glorfeld, revisited the study with a new cohort [19]. From the 1040 students enrolled in the introductory programming course, 150 were randomly selected. The model still performed well, although the accuracy decreased. This is perhaps to be expected when models are exposed to new data sets and being tested for generalizability. Fowler and Glorfeld reported that: "The validation study showed that the model would have a predictive accuracy of approximately 75% in actual application". While the model had a strong prediction accuracy, and was able to predict at an early stage in the CS1 module, the initial study and the re-validation were conducted in consecutive years, in the same institution. Thus it is difficult to assess how this model would perform 35 years after its development and generalize to other institutions, different academic levels and countries.

In summary the literature review revealed that while many of the 40 articles contribute significantly to the CSEd community, the process identified several of the ITiCSE working group concerns [21]. One of the most prominent issues is that the CSEd community find repeatability a challenge, replication studies are rare, and there is an alarming concern that they may still be rare a decade from now [21]. It is also concerning that from the desired generalizability criteria (at the heart of PreSS), not only do the studies fall short of acquiring all of the criteria, they struggle to acquire multiple criteria.

4 DATA COLLECTION

During the academic years 2015-16, a large-scale multi institutional study [33] took place in Ireland and Denmark (two universities, five institutes of technology (comparable in academic level to colleges in the US) and four community colleges). The data collected was captured under three categories. The first category captured student data, including background, institution / course and psychological data, which was captured at approximately 4-6 hours

into the introductory programming module (this is approximately, when students are 10% of the way through the module). The second category captured final CS1 performance data, such as grade. The final category was the learning analytics data, captured at the same time as the first category. It should be noted at this point while the data collected contains both educational data mining and learning analytics data, this is a revalidation of an educational data mining study. Thus the learning analytic data is outside the scope of this paper, but will be addressed in future work.

In total, 692 complete student data sets were used in the study. Six programming languages were used which included : Java ($n=553$), C# ($n=75$), Python ($n=33$), Processing ($n=24$), Visual Basic ($n=4$) and C++ ($n=3$). In total 45 factors were recorded from the data gathered in the first category. Details of the instruments used and all of the data collected can be found in the references [33].

5 PRESS REVALIDATION

5.1 Analysis Techniques

PreSS uses naïve Bayes as the machine learning algorithm. The rationale for this was its performance in previous studies [3, 4]. Ten-fold cross validation (10FCV) was also implemented to obtain a prediction accuracy for the models being tested. This is a best practice method to avoid over fitting [43] with machine learning algorithms.

Accuracy, sensitivity and specificity were recorded for each prediction. These recordings were calculated based on the following variables: true positive rate (TP - the correct identification of a weak student), true negative rate (TN - the correct identification of a strong student), false positive rate (FP - type I error) and the false negative rate (FN - type II error). As the overall priority of PreSS is to identify weak students, sensitivity was the main focus. All three measurements were recorded and presented for every prediction in this study.

5.2 Results

The PreSS model was unchanged from Bergin's original study [3], using the same three factors; programming self-efficacy, mathematical ability based on a high school mathematics exit examination and number of hours per week a student plays computer games [5]. The data reduction algorithm and the 10-fold cross validation techniques were also unchanged. The results and comparison between the original study and this study are presented in table 1.

Table 1: PreSS model; run on two disjunctive data sets, separated by twelve years.

Data Set	N	Accuracy	Sensitivity	Specificity
2005	102	77%	85%	66%
Current Study	692	67%	78%	53%

PreSS is still able to predict with a high accuracy (for two out of every three students), twelve years on, with an $n = 692$, from 11 institutions with diverse academic levels in two different countries. Although the accuracy reduced by 10%, this result is still significant as previous revalidation attempts on other models (section 3.8) have seen the model rejected or an accuracy reduction of 6% in the span of a single year, while based only in a single institution.

6 UPDATING THE PRESS MODEL

6.1 Background of Additional Factors

This study also examined multiple factor combinations not used in the original PreSS model, with the aim of developing an updated PreSS model. These additional factors were identified in a study conducted in 2016 as described in section 2.4 [31]. From this research, 16 additional factors were identified and added to the PreSS survey [31]. Additionally due to the online web based nature of the survey several other factors such as time it took to complete the survey, were also measured. In total, 42 factors were recorded in the PreSS# survey. This resulted in (after data reduction algorithms have been applied) 18 factors that could be examined as part of the PreSS model [31, 33].

6.2 Factor Selection

Due to the large number of combinations of factors, that could be used in the updated PreSS model, a factor selection process was chosen. This process was used mainly for repeatability. Two factor selection algorithms were chosen: correlation evaluation and information gain. Both of these algorithms were run on the data set and combinations of the highest ranked factors were examined.

6.3 Updated Models

In total 62 new models examined, resulting from the findings of section 6.2. The results for the updated PreSS models are presented in Table 2, where an ID to represent the models and the number of factors in the model are also presented. This papers main goal was the revalidation of PreSS, thus only a summary of the most successful models or models with a high value to the CSEd community are presented here, where this is by no means exhaustive.

Model ID 1, included the four factors: the students age in raw integer form, a students self-reported expected end of module result, a students mathematical ability normalized and a students programming self-efficacy. This model was selected as the primary candidate for the updated PreSS model as it reported the highest accuracy of all the models and one of the highest sensitivities.

Model ID 2, included the four factors: Institution type (University, College or Community College), if a student was mature or not (23 and over or under 23 years of age) in dichotomous form, a students mathematical ability normalized and a students programming self-efficacy. This model was also selected as a primary candidate for the updated PreSS model. Its accuracy was among the highest recorded on the full data set, but even more noteworthy is the models sensitivity was higher than model ID 1 and second highest overall.

Model ID 3, only included two factors: a students age in raw integer form and the time spent on social media per day. The models accuracy and specificity was significantly lower than that of Model ID 1 and 2, but interestingly this model had the highest sensitivity of any model examined. Although this model will not be considered for the final PreSS model, it may positively correlate the time spent on social media with struggling students in CS1, and perhaps other spaces. Further research is required on this model and its factors.

Table 2: PreSS updated models, based on factors developed from recent research

Model ID	Factors	Accuracy	Sensitivity	Specificity
1	4	71%	75%	66%
2	4	69%	80%	54%
3	2	60%	86%	26%

6.4 Updated Models Conclusion

The results are very positive. The increases in performance although small, show that PreSS may be able to produce further performance gains. Possible avenues that could be explored may include the re-examination of multiple machine learning algorithms, as this study used the same algorithm as Bergin a decade previous (naïve Bayes). Algorithms not commonly used a decade previous should also be considered.

7 CONCLUSION & FUTURE WORK

Over a decade after its conception, validation, and presentation at SIGCSE'05, PreSS has once again been put under the microscope. Never before has a model been revisited several years after its validation where the CS landscape has significantly changed, using a disparate cohort and still produced a high prediction accuracy. This paper builds upon the work of previous studies and makes a solid contribution to the CSEd community, laying the foundation for future revalidation and replication studies. In doing so this paper has addressed the second of the Grand Challenges as laid out by the ITiCSE working group in 2015. PreSS is able to predict with an accuracy of 71%, twelve years after it was first developed and validated. More importantly PreSS was able to identify weak students (the main goal of PreSS) with a sensitivity of 75-80%. It was anticipated that the accuracy would decline (down 6.5% compared to 77.5% in the original PreSS study), given the substantial increase in student numbers, diversity of institutions / academic levels and change in student profile over time. New factors such as social media usage rather than time spent playing computer games, could hypothetically be more useful now [31]. The performance is still significantly above that of chance. PreSS is now a tool of real value to the CSEd community, which is compounded by the fact that PreSS also has a finished end-to-end real time prediction tool (PreSS#), that is tried, tested and ready to roll out on a large scale.

The next phase of work will be to examine the learning analytics data collected at the same time as the educational data mining data. Future work will also examine the educational data mining data and the learning analytics data simultaneously, further developing the PreSS model. The authors believe that this may be a first in the CSEd community, as both domains may have correlations or interesting relationships that may explain phenomena while trying to identify students who are at risk of failing in CS1. Future work should also run PreSS# live in pilot institutions. This would be, from reviewing the literature, the first time that a CS1 prediction model was put into production. An advantage to the early identification of at risk students, is that useful and perhaps even tailored interventions could be deployed in a timely manor. In addition, by examining both control and test groups the authors could examine and explain the results that have arisen from the previous nine studies. In doing

so the authors may be able to contribute towards two additional calls of the ITiCSE working group, the Grand Challenges three and four [21].

REFERENCES

- [1] A. Ahadi, R. Lister, H. Haapala, and A. Vihavainen. Exploring machine learning methods to automatically identify students in need of assistance. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, ICER '15, pages 121–130, New York, NY, USA, 2015. ACM.
- [2] R. J. Barker and E. A. Unger. A predictor for success in an introductory programming class based upon abstract reasoning development. In *Proceedings of the Fourteenth SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '83, pages 154–158, New York, NY, USA, 1983. ACM.
- [3] S. Bergin. *A computational model to predict programming performance*. PhD thesis, Department of Computer Science, Maynooth University, 2006.
- [4] S. Bergin, A. Mooney, J. Ghent, and K. Quille. Using machine learning techniques to predict introductory programming performance. *International Journal of Computer Science and Software Engineering*, 4(12):323–328, 2015.
- [5] S. Bergin and R. Reilly. Programming: Factors that influence success. *SIGCSE Bull.*, 37(1):411–415, February 2005.
- [6] S. Bergin and R. Reilly. Predicting introductory programming performance: A multi-institutional multivariate study. *Computer Science Education*, 16(4):303–323, 2006.
- [7] G. D. Boetticher, W. Ding, C. Moen, and K.-B. Yue. Using a pre-assessment exam to construct an effective concept-based genetic program for predicting course success. In *Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '05, pages 500–504, New York, NY, USA, 2005. ACM.
- [8] R. Bornat, S. Dehnadi, and Simon. Mental models, consistency and programming aptitude. In *Proceedings of the Tenth Conference on Australasian Computing Education - Volume 78*, ACE '08, pages 53–61, Darlinghurst, Australia, Australia, 2008. Australian Computer Society, Inc.
- [9] D. F. Butcher and W. A. Muth. Predicting performance in an introductory computer science course. *Commun. ACM*, 28(3):263–268, Mar. 1985.
- [10] J. Campbell, D. Horton, and M. Craig. Factors for success in online cs1. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '16, pages 320–325, New York, NY, USA, 2016. ACM.
- [11] C. K. Capstick, J. D. Gordon, and A. Salvadori. Predicting performance by university students in introductory computing courses. *SIGCSE Bull.*, 7(3):21–29, Sept. 1975.
- [12] M. E. Caspersen, K. D. Larsen, and J. Bennedsen. Mental models and programming aptitude. In *Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, ITiCSE '07, pages 206–210, New York, NY, USA, 2007. ACM.
- [13] D. Cukierman. Predicting success in university first year computing science courses: The role of student participation in reflective learning activities and in i-clicker activities. In *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '15, pages 248–253, New York, NY, USA, 2015. ACM.
- [14] N. Culligan, K. Quille, and S. Bergin. Veap: A visualisation engine and analyzer for press#. In *Proceedings of the 16th Koli Calling International Conference on Computing Education Research*, Koli Calling '16, pages 130–134, New York, NY, USA, 2016. ACM.
- [15] S. Dehnadi. Testing programming aptitude. In *Proceedings of the 18th Annual Workshop of the Psychology of Programming Interest Group*, pages 22–37, 2006.
- [16] P. Denny, A. Luxton-Reilly, J. Hamer, D. B. Dahlstrom, and H. C. Purchase. Self-predicted and actual performance in an introductory programming course. In *Proceedings of the Fifteenth Annual Conference on Innovation and Technology in Computer Science Education*, ITiCSE '10, pages 118–122, New York, NY, USA, 2010. ACM.
- [17] A. Estey and Y. Coady. Can interaction patterns with supplemental study tools predict outcomes in cs1? In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '16, pages 236–241, New York, NY, USA, 2016. ACM.
- [18] G. C. Fowler and L. W. Glorfeld. Predicting aptitude in introductory computing: A classification model. *AEDS Journal*, 14(2):96–109, 1981.
- [19] L. W. Glorfeld and G. C. Fowler. Validation of a model for predicting aptitude for introductory computing. In *Proceedings of the Thirteenth SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '82, pages 140–143, New York, NY, USA, 1982. ACM.
- [20] T. R. Hostetler. Predicting student success in an introductory programming course. *SIGCSE Bull.*, 15(3):40–43, Sept. 1983.
- [21] P. Ihtantola, A. Vihavainen, A. Ahadi, M. Butler, J. Börstler, S. H. Edwards, E. Isohanni, A. Korhonen, A. Petersen, K. Rivers, M. A. Rubio, J. Sheard, B. Skupas, J. Spacco, C. Szabo, and D. Toll. Educational data mining and learning analytics in programming: Literature review and case studies. In *Proceedings of the 2015 ITiCSE on Working Group Reports*, ITiCSE-WGR '15, pages 41–63, New York, NY, USA, 2015. ACM.
- [22] J. Konvalina, S. A. Wileman, and L. J. Stephens. Math proficiency: A key to success for computer science students. *Commun. ACM*, 26(5):377–382, May 1983.
- [23] L. Lambert. Factors that predict success in cs1. *J. Comput. Sci. Coll.*, 31(2):165–171, Dec. 2015.
- [24] R. R. Leeper and J. L. Silver. Predicting success in a first programming course. In *Proceedings of the Thirteenth SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '82, pages 147–150, New York, NY, USA, 1982. ACM.
- [25] J. Leinonen, L. Leppänen, P. Ihtantola, and A. Hellas. Comparison of time metrics in programming. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*, ICER '17, pages 200–208, New York, NY, USA, 2017. ACM.
- [26] S. N. Liao, D. Zingaro, M. A. Laurenzano, W. G. Griswold, and L. Porter. Lightweight, early identification of at-risk cs1 students. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, ICER '16, pages 123–131, New York, NY, USA, 2016. ACM.
- [27] A. Lishinski, A. Yadav, R. Enbody, and J. Good. The influence of problem solving abilities on students' performance on different assessment tasks in cs1. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, SIGCSE '16, pages 329–334, New York, NY, USA, 2016. ACM.
- [28] A. Lishinski, A. Yadav, J. Good, and R. Enbody. Learning to program: Gender differences and interactive effects of students' motivation, goals, and self-efficacy on performance. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, ICER '16, pages 211–220, New York, NY, USA, 2016. ACM.
- [29] L. Porter, D. Zingaro, and R. Lister. Predicting student success using fine grain clicker data. In *Proceedings of the Tenth Annual Conference on International Computing Education Research*, ICER '14, pages 51–58, New York, NY, USA, 2014. ACM.
- [30] K. Quille and S. Bergin. Programming: Factors that influence success revisited and expanded. In *International Conference on Enguaging Pedagogy (ICEP), 3rd and 4th December, College of Computing Technology, Dublin, Ireland*, 2015.
- [31] K. Quille and S. Bergin. Programming: Further factors that influence success. *Psychology of Programming Interest Group (PPIG), 7th to 10th Spetember, University of Cambridge*, 2016.
- [32] K. Quille, S. Bergin, and A. Mooney. Press#, a web-based educational system to predict programming performance. *International Journal of Computer Science and Software Engineering (IJCSSE)*, 4(7):178–189, 2015.
- [33] K. Quille, N. Culligan, and S. Bergin. Insights on gender differences in cs1: A multi-institutional, multi-variate study. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '17, pages 263–268, New York, NY, USA, 2017. ACM.
- [34] N. Rountree, J. Rountree, and A. Robins. Predictors of success and failure in a cs1 course. *SIGCSE Bull.*, 34(4):121–124, Dec. 2002.
- [35] D. F. Shell, L.-K. Soh, A. E. Flanigan, and M. S. Peteranetz. Students' initial course motivation and their achievement and retention in college cs1 courses. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, SIGCSE '16, pages 639–644, New York, NY, USA, 2016. ACM.
- [36] W. T. Tarimo, F. A. Deeb, and T. J. Hickey. Early detection of at-risk students in cs1 using teachback/spinoza. *J. Comput. Sci. Coll.*, 31(6):105–111, June 2016.
- [37] B. Trstenjak and D. Donko. Determining the impact of demographic features in predicting student success in croatia. In *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on*, pages 1222–1227. IEEE, 2014.
- [38] B. Trstenjak and D. Āronko. Determining the impact of demographic features in predicting student success in croatia. In *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1222–1227, May 2014.
- [39] A. Vihavainen. Predicting students' performance in an introductory programming course using data from students' own programming process. In *Proceedings of the 2013 IEEE 13th International Conference on Advanced Learning Technologies*, ICALT '13, pages 498–499, Washington, DC, USA, 2013. IEEE Computer Society.
- [40] C. Watson, F. W. B. Li, and J. L. Godwin. Predicting performance in an introductory programming course by logging and analyzing student programming behavior. In *Proceedings of the 2013 IEEE 13th International Conference on Advanced Learning Technologies*, ICALT '13, pages 319–323, Washington, DC, USA, 2013. IEEE Computer Society.
- [41] D. M. Wiig. A bayesian probability approach to predicting student performance in introductory computer science courses. *SIGSMALL/PC Notes*, 15(1):3–19, Feb. 1989.
- [42] B. C. Wilson and S. Shrock. Contributing to success in an introductory computer science course: A study of twelve factors. In *Proceedings of the Thirty-second SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '01, pages 184–188, New York, NY, USA, 2001. ACM.
- [43] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, Amsterdam, 2016.