Programming: Further Factors that Influence Success

Keith Quille
Department of Computer Science
Maynooth University
keith.quille@nuim.ie

Susan Bergin
Department of Computer Science
Maynooth University
susan.bergin@nuim.ie

Abstract

This paper revisits five previous studies spanning a decade, which were used in the development and validation of a computational model named PreSS (Predict Student Success) to predict student success in an introductory programming module with 80% accuracy. PreSS was initially developed over three studies from 2004 – 2006, recording data from 240 students across multiple institutions incorporating three studies. These studies examined 25 factors and six machine learning algorithms, resulting in a model that used only three factors and the naïve Bayes machine learning algorithm (coupled with multiple other statistical techniques) called PreSS. The authors completed two subsequent independent studies over the academic years 2013- 2015 that were used to investigate if the PreSS model was still valid on a modern data set while also collecting new data (factors) that were not considered during the initial development of PreSS. These two studies successfully validated PreSS, even with a decade separating the student profiles and landscapes.

This paper has two main objectives; the first objective is an investigation on factors gathered during the initial development of PreSS that were not used in the final model. This is important as several factors were found to be significant at the time but were excluded as their associated sample size was small and the goal was to develop the most generalizable model possible. PreSS is arguably a universal model for two reasons. 1) PreSS is independent of the programming language used. PreSS has been so far been exposed to 6 different programming languages that include: Java, C#, Processing, Python, Visual Basic and C++ and still maintained the same high level of accuracy. 2) PreSS is also independent of any specifics of the cohort sampled such as gender bias, institution bias, age bias etc. The second objective of this paper is an examination of additional data collected in two recent independent studies, to determine if incorporating some or all of these new factors could improve the accuracy of PreSS.

This paper reports on 90 experiments using data from all five previous studies, examining 126 possible factors. The work successfully identified 16 factors that when used in combination with the original PreSS factors either produced significant increases in prediction accuracy or exhibited noteworthy findings. Some of the 16 factors produced substantial gains in accuracy (in some cases in excess of 8%) or when integrated into the PreSS model, revealed interesting substitutions of factors.

1. Introduction

Computer Science (CS) non-progression rates in Ireland are alarming with a large number of students failing to progress each year. Currently non progression rates are 25% in CS, which is significantly higher than the national average of 16%. In two recent reports (2010 and 2016 respectively) that cover all Irish higher level institutions, CS was found to have one of the largest, if not the largest rate of non-progression across all National Framework of Qualification (NFQ) levels, from level 6 to level 8 (Liston et al. 2016; Mooney et al. 2010). In addition, CS is one of only two fields of study, where the non-progression rate has increased since the first report in 2010. It is well acknowledged that a main contributor is students struggle to succeed in their initial programming module, a staple in most first year CS courses. Early identification of students who are at risk of non-progression is often hindered by the very high student-lecturer ratio (100:1 or greater) and often lecturers are not aware of the student's progress until after the initial assessment. Numerous approaches have been trialled to varying degrees of success to improve learning and assessments outcomes. Our institution is no different with numerous initiatives implemented to improve outcomes, including the development of an automated adaptive assessment systems, (Traynor et al. 2006), novel teaching approaches such as problem based learning (O'Kelly et al. 2004) and more recently a Programming Support Centre run by peer tutors (Nolan et al. 2015). As computer programming is not a traditional Leaving Certificate subject or topic, there are no prior indicators of a student's ability available to lecturers at an early stage, to enable the introduction of appropriate interventions. This often renders interventions, or appropriate methodologies inadequate as their introduction may have been too late in the course delivery to make a significant difference.

2. Background

2.1. PreSS

PreSS is the result of work completed over three years, between 2004 and 2006. This work consisted of several parts: three studies (a pilot study, a main study and an epilogue study), the identification of early predictors of performance in an introductory programming module, an investigation of multiple machine learning models for predicting student performance and finally the development of a computational model called PreSS (Predict Student Success) (Bergin and Reilly 2006; Bergin 2006).

The three studies were part of a longitudinal study, which was completed over four institutions during the three years. These institutions consisted of a University, two Institutes of Technology, and a Community College thus spanning all levels of Higher Education, from the National Framework of Qualifications (NFQ) level 5 to level 8. The pilot study recorded data using a questionnaire, which in turn lead to the development of the instruments that were used in the main study. The main study recorded a substantial amount of data from 123 students during the academic year of 2004-2005. Four paper based instruments were used to collect the data in the main study, a background questionnaire, a programming self-esteem questionnaire, a self-efficacy questionnaire and a motivation and learning strategies questionnaire (Bergin 2006; Bergin et al. 2005; Bergin and Reilly 2005). An epilogue study with 21 participating students was also conducted after the main study and was used in further analysis and validation of findings from the main study.

In Bergin's main work 25 factors were examined, obtained from the data collected using the four instruments in the main study. Bergin used multiple statistical techniques when examining factor selection for use in the prediction model. Stratification was used as the initial step to ensure that all the data is in a state of homogeneity. Ten-times ten-fold cross validation (10FCV) was also implemented which is an acknowledged method of avoiding over fitting when accessing how the results of statistical analysis will generalise to a larger independent data set, Witten and Frank (2005). This method is more suitable than other conventional methods of statistical model validation, as the validation set is not represented in the training set, thus reducing bias and therefore over fitting of the accuracy prediction. Initially, Bergin examined over 40 logistic regression models using combinations of the 25 factors on the data from the main study. This resulted in a single prediction model that was marginally superior. This prediction model used three factors: `programming self-esteem`, `mathematical ability` (based on a high school mathematics exit examination) and `the number of hours per week that a student plays computer games`.

The pre-processed programming self-esteem data consisted of ten questions (Bergin 2006) and was based on the Rosenberg self-esteem questionnaire but modified to reflect a student's perception of their programming ability (Rosenberg 1965). Cronbach alpha values used to estimate the internal consistency of psychometric questionnaires, were compared to investigate if the modified Rosenberg questionnaire had an equivalent or greater internal consistency than the unmodified questionnaire, thus showing that the modified questionnaire is a valid questionnaire to measure programming self-esteem. The Cronbach alpha values for the unmodified Rosenberg self-esteem questionnaire were in the range of 0.82 to 0.88 and for the modified questionnaire, the alpha value was 0.91 (Bergin 2006). Principle Component Analysis (PCA) was used to reduce some attributes that consisted of multiple data points to one value which accounted for as much of the variance in the multiple data points as possible. PCA was used to reduce the 10 questions of the programming self-esteem attribute to one principal component (Bergin 2006). Bergin examined multiple prediction models using six machine learning algorithms: Nearest Neighbour, Decision Tree's, Naïve Bayes, Logistic Regression, Support Vector Machines (SVM) and Backpropagation Networks (Bergin and Reilly 2005) with the results of this work presented in Table 1 and revisited in a study (Bergin et al. 2015).

Algorithm	Accuracy%
Naïve Bayes	78.28
Logistic Regression	76.47
Backpropagation	75.46
Support Vector Machine	77.49
Decision Tree (C4.5)	74.49
K Nearest Neighbour (K=3)	71.64

Table 1: *Comparison of machine learning algorithms from Bergin's study*

The `programming self-esteem` and `mathematical ability` factors were found to have a positive relationship with performance while `the number of hours a student plays computer games` was found to have a negative effect. Bergin analysed the results of six different machine learning algorithms and selected naïve Bayes as the algorithm of choice for multiple reasons; firstly it produced the highest accuracy of the six algorithms examined, but also naïve Bayes is a probabilistic classifier, where the algorithm computes a probability of how confident it is for each prediction. For example, a student who was classified as a strong programmer and where naïve Bayes is not confident in its prediction may need further investigation. This is a very useful tool in the case of the PreSS model, as misclassifying a strong student as week is a significantly less adverse, than misclassifying a weak student as strong. This coupled with the fact that naïve Bayes is computationally inexpensive and thus is appropriate in a real-time web-based environment, which was outlined as possible future work of Bergin and in section 2.2.

2.2. PreSS#

PreSS# is a web based educational system, with the PreSS model at its core was developed during the academic year 2013-2014 (Quille et al. 2015), to continue Bergin's future work that was outlined after the development of PreSS. Bergin's model, used a paper based data collection method, and thus it was time consuming to gather and input the data, coupled with this the data collected also contained some partial and dirty data entries. If PreSS were to be used across multiple institutions with the results of the predictions expected in real time, the paper based collection method would be unsuitable. PreSS# could be used as a tool in an educational environment that would allow educators to make informed decisions about methodologies, differentiation and interventions at an earlier stage that will enable both weak and strong students to achieve their highest potential in real time. Additionally a large scale multi institutional study could be completed using PreSS#, examining the current effectiveness of the PreSS model.

The development of PreSS# consisted of four main phases. The independent development and validation of a PCA and naïve Bayes sub system, the integration of these two subsystems, into the main web application, the development of an intuitive, secure, mobile responsive interface and front end, and finally the validation of PreSS# using the same data that Bergin used in her main study. Both the PCA and naïve Bayes sub systems, were developed as an independent library of classes, which allowed for integration into PreSS# and any other future research and development. Both sub systems were independently validated using data and outcomes from Smith 2002 and Souza 2014. The PreSS# front end employed several techniques to ensure ease of use, rapid development techniques, security and services. Further details can be found in Quille et al. 2015. The system was validated using the same input data as Bergin's main study (Quille et al. 2015) and the results are presented in Table 2.

System Validation	Accuracy%
PreSS# (.NET)	80.39
PreSS (WEKA)	80.39

Table 2: *Results from PreSS validation, (Quille et al. 2015)*

A two tailed t-test for a binomial distribution was completed to confirm the validation of PreSS#, concluding that there was no statistical differences between the prediction accuracies of .NET and PreSS (using WEKA), with a $p_{(value)} = 1.0$ *and a* $t_{(value)} = 0.0$.

2.3. Study: Programming, Factors that Influence Success Revisited and Expanded

After the validation of PreSS#, the authors completed two independent studies during the academic years of 2013 – 2015, which were used to validate PreSS on a modern data set, (Quille et al. 2015). This study was conducted for two reasons. Firstly, the landscape, population and the student profile has significantly changed over the last decade since PreSS was developed. Secondly, validation of any study is important as all too often studies are undertaken and never repeated, thus the value of their findings remains questionable.

The two studies named: S14 (conducted in the 2013-2014 academic year) and S15 (conducted in the 2014-2015 academic year) were hosted by a Community College completing an Advanced Certificate in Computer Science. This is a similar institution to one of the institutions investigated in Bergin's main study. S14 consisted of 34 students with a 5:29 ratio of female to male students. The divide of successful students compared to non-successful students was 16:18 and the ratio between mature students (students under the age of 23) and non-mature students (students 23 years of age or older) was 12:22 respectively. The second study S15 consisted of 26 students. Four females and 22 males took part in the study. The overall final results showed the divide of weak to strong students at 9 to 17 and the ratio between mature students and non-mature students was 3:23 respectively. These statistics are highlighted to show the variation between the student cohorts even though they were conducted in the same institution. There were also further differences between Bergin's main study and S14, S15. The majority of students in the main study used Java as the introductory language, where as in S14 and S15 the language used was C#. This was the first time that C# was used with PreSS and to the authors knowledge, in any study to predict programming performance. C# was not commonly used in education or industry until sometime after PreSS was developed.

PreSS# was used to automatically collect the data, ensuring the data was clean and no data samples were missing due to the interfaces automatic validation techniques. An additional survey was completed alongside both, S14 and S15 studies, with the goal of capturing additional novel factors which may add value. The survey questions were selected based on hypothesis that some factors, which may have not existed or significantly impacted the model during Bergin's study, were worthy of examination and could contribute to increasing the PreSS model's prediction accuracy, details are provided in Appendix 1.

The results of this investigation were obtained by training the PreSS model using Bergin's 102 data samples from the main study conducted a decade previous. The process used in this investigation was identical to the process used by PreSS, using the same machine learning toolbox (WEKA) and statistical methods to compute the predictions. Both prediction accuracies using S14 and S15 were independently compared to Bergin's accuracy using a Welch's t-test and a binomial distribution to calculate the variance. The accuracy achieved in the S14 study produced a p value of 0.8592 and a t value of 0.1778 thus showing that there was no statistical difference between the two accuracies. The accuracy achieved in the S15 study produced a P value of 0.5669 and a T value of 0.5748 thus showing that there was no statistical difference between the two accuracies. This means that even though the PreSS model was developed nearly a decade ago, similar levels accuracy are achievable in a considerably changed landscape today.

This result was very significant as both S14 and S15 studies consisted of very different student profiles with age, results and gender ratios differing significantly. Furthermore the student profiles from both these studies, differed significantly to Bergin's student profiles. This was partly due to the time lapse between studies, institution representation (Community Colleges represented less than 7% of institutions in Bergin's main study) and programming language used. This considered, PreSS still produced statistically similar accuracies using S14 and S15. This implies that even though Bergin's PreSS model was developed over ten years ago, it is still valid, universal and not biased towards an institution, student profile, programming language, era or any other known factor, thus always achieving the same high level of accuracy.

3. Motivation

From unused factors in the final PreSS model, Bergin discussed the inclusion and combinations that may have value and were worth additional investigation, one in particular was `gender` and its relationship to the factor, `hours spent playing computer games`. Bergin noted that, removing `the number of hours students spent playing computer games` improves the classification of female students suggesting that its inclusion is not constructive in predicting female student performance. Bergin also noted that developing a prediction for each institution used in the main study resulted in higher classification accuracy for each individual institution. Finally Bergin noted that some factors investigated were only available in smaller sub sets of the main study data, due to restrictions in the data collection method and although showed value in the small sub sets could not be verified on a large data sample, thus were not included in the final PreSS model.

In addition new factors were captured by the additional surveys administered alongside S14 and S15, multiple investigations were now required to examine if the addition to or the replacement of the original PreSS factors with these new factors, captured during S14 and S15 may lead to an increase in the models prediction accuracies. Although these factors are not present on Bergin's main study data, and only on the two smaller studies, S14 and S15, if improvements to the model can be shown here, they should warrant inclusion in future studies to investigate their true value.

The aim and objectives for this work is to examine if the remaining ≈ 20% error in prediction accuracy of PreSS can be reduced, whilst still keeping the PreSS model universal. To do this an examination of the unused factors from Bergin's original work, (from the large amount of data collected in the main study) and additional new factors collected from the two studies S14 and S15 (Quille et al. 2015) will be carried out. Where any improvements or value is found, these factors will be included in an upcoming large study.

4. Main Study Revisited

4.1. Research Design

The authors revisited the data in Bergin's main study. There were a total of 123 students that participated with 113 recorded data points (factors). These 113 recorded data points consisted of questions and normalisation or summations/ differences of combinations of questions. An example of this was a variation of `the hours spent playing computer games` factor used in the PreSS model. This attribute was the `hours spent playing computer games during the duration of the introductory programming module`. Bergin also recorded `the amount of hours spent playing computer games before the commencement of the introductory programming module`. These two questions also had a third data point, the difference between the two. This is just to show that Bergin not only examined a large amount of questions, but also examined outputs using combinations of questions. These 113 data points were not all recorded for every student due to some institutions not being able to complete each or all of the four instruments used, due to time or administration constraints. Due to the large quantity of data points, a method was developed to identify data points that may have value or be of interest for further investigation. Standard deviation and therefore variance was calculated for each data point over all the students that it was recorded for. The rationale for this method was based on the hypothesis that if the variance was small for a data point, it may not have value as a predictor attribute due to the fact that if each student had a similar value (due to low variance) when that value was used in a naïve Bayes classifier, the probability for that attribute would be similar in each case adding no value to the predicted output. This was further confirmed when the variance was calculated on the three attributes used in PreSS, each having a relatively large variance, which was then used as a baseline variance for selection of further data points for inclusion in this study.

This resulted in the identification of multiple data points that were to be re-examined. These data points were then developed into experiments for further investigation. Thus 65 experiments were performed which included the addition and multiple exhaustive combinations and substitutions of the data point, with the original PreSS model's attributes and then this new model was statistically analysed using a student's t-test to examine if the new prediction accuracy produced, yielded a significant increase in accuracy above the PreSS model. Noteworthy findings were also recorded.

Although in total, there was 113 data points recorded in Bergin's main study, these data points were not recorded for every student as previously mentioned. This would result in a biased finding if, for example data point x was only recorded for 50 students and the prediction result of this new model which included x in some variant was obtained using a sample size, n = 50, and then the prediction accuracy was compared to the original PreSS prediction accuracy with a sample size, n = 102. To ensure an unbiased statistical comparison, the original PreSS model was re-run only on the students that were used in each investigation, resulting in a baseline prediction accuracy to be statistically compared with the new model incorporating the new data point. A Students *t*-test was used to examine if a significant difference was found in accuracy between the baseline accuracy and the experiment accuracy.

4.1. Experiment Results

Table 3 shows, of the 65 experiments conducted using the main study, the 12 experiments that resulted in a significant gain in accuracy or showed value when combined in some way with the PreSS model.

Experiment ID	New Model Attributes	Sample size (N)	Accuracy Increase %	*p* value	*t* value
1	Mathematical grade, hours spent playing computer games and what a student believed their final overall grade would be at the beginning of the module.	56	8.93	< 0.001 ***	13.917
2	Mathematical grade, programming self-esteem and the difference in hours spent playing computer games before the commencement of the course with that of the hours spent playing computer games during the course.	107	7.50	< 0.001 ***	12.200
3	Mathematical grade, programming self-esteem, hours spent playing computer games and what a student believed their final overall grade would be at the beginning of the module.	56	7.20	< 0.001 ***	11.013
4	Mathematical grade, programming self-esteem, hours spent playing computer games and a student's anxiety, fear and or uneasiness of examinations and assessment, and a focus on the negative while completing them.	68	2.94	< 0.001 ***	5.293
5	Mathematical grade, programming self-esteem, hours spent playing computer games and a student's feeling on their own ability on programming concepts, design of programming logic and completion of lab assignments.	110	1.82	0.0010 ***	3.165
6	Mathematical grade, programming self-esteem and the number of hours a week the student worked in a part time job.	111	1.81	0.0024 **	3.064
7	Mathematical grade, programming self-esteem, hours spent playing computer games and a student opting for more challenging / relevant course material when given a choice, even at the cost of a higher grade if less challenging material was chosen.	68	1.40	0.010 *	2.606
8	Mathematical grade, programming self-esteem, hours spent playing computer games and a belief the student has, that if they work hard that they will succeed and if they do not work, that they will fail and it will be the student's own fault that they did.	68	1.40	0.010 *	2.606
9	Mathematical grade, programming self-esteem, hours spent playing computer games and a student's belief that they will be successful in the course before it commences.	68	1.40	0.010 *	2.606

10	Mathematical grade, programming self-esteem, hours spent playing computer games and a normalisation of experiment ID 5 representing intrinsic questioning.	68	1.40	0.010 *	2.606
11	Mathematical grade, programming self-esteem, hours spent playing computer games and a normalisation of factor: "A student perusing a high grade in a class as their main motivation / learning outcome" representing extrinsic questioning.	68	1.40	0.010 *	2.606
12	Mathematical grade, programming self-esteem, hours spent playing computer games and gender.	107	0.00	0.1200	1.550

Table 3: *Bergin's main study unused factors that have shown value where*

** Significant at p < 0.05; ** Significant at p < 0.005; *** significant at p < 0.001;*

4.2. Summary & Discussion

From a total of 65 experiments run on Bergin's main study, 12 combinations of factors have emerged which may have added value.

From both experiments ID 1 and 3 from Table 3, the factor was: `the early expectation of the final grade that a student will achieve in the introductory module`. This factor of a students expected grade was recorded early in the course delivery and when directly added into the PreSS factors resulted in a very significant increase in accuracy which was at just over 7%. Furthermore when this factor replaced the `programming self-esteem` factor, (thus removing the `programming self-esteem` factor completely), the overall accuracy compared to the control accuracy showed the highest increase in accuracy for this study of just under 9%. This is a noteworthy result, which could contribute to the overall goal of increasing the prediction accuracy of PreSS.

In experiment ID 2, from Table 3, the addition of `the difference in time spent playing computer games prior to starting the course to the time they spent playing computer games during the course`, produced a very significant increase in accuracy, up to 8%. This although, seems related to the original PreSS factor of `hours spent playing computer games during the course`, the substitution of this difference as a factor may have value over the original PreSS factor and is worth further investigation.

From experiment ID 4, Table 3, which was multiple self-ranking questions summated to give an indicator of a `student's anxiety, fear and uneasiness towards formative assessment and examinations.` When this factor was added to the PreSS factors, the resultant model produced a significant increase in accuracy of almost 3%. This like in experiment ID 5, Table 3, may be an indicator of `programming self-efficacy`. The multiple experiments, id's 7 – 11, Table 3, also showed a significant in accuracy although the overall gain may be classified as minimal. None the less, they may add value to an updated model and for this reason further investigations should be completed using these data points.

In experiment ID 5, Table 3, a `student's feeling on their own ability on programming concepts, design of programming logic and completion of lab assignments`, added a significant increase to the models accuracy. The authors hypothesises that this may be related to a student's `programming self-efficacy` and due to the large increase in accuracy would warrant further analysis on a modern data set. The same additional analysis should be applied for the factor: `the amount of hours in part time employment while completing the course`, experiment ID 6, Table 3. Although the addition of this factor to the PreSS factors did not yield any value to the model, when it replaced `hours spent playing computer games` a significant increase in accuracy was found.

`Age` may have value in an updated PreSS model as it may help to further increase the models accuracy. Age was examined in all three studies. In Bergin's main study, the size of the data set that had age recorded was quite small and only had two of the four institutions represented, also in this dataset the age was recorded as a dichotomous value thus in these experiments age did not yield any significant increases, which was expected due to the quality of the data set.

`Gender` as a factor in this study (ID 12) corresponded with Bergin's findings, which showed no significant increase in accuracy when added to the PreSS model, but did have value when predicting the performance of only female students, when `the hours spent playing computer games` factor was replaced with gender. Unfortunately when `the hours spent playing computer games factor` was removed, the prediction accuracy for male students decreased. Thus if a suitable replacement factor `for the hours spent playing computer` games could be found then the introduction of the gender factor may have some real value specifically for predicting the performance of female students.

5. S15 additional survey experiments

5.1. Research Design

The authors had collected additional factors that were hypothesised to add value to the PreSS model (Appendix 1) during two studies S14 and S15 respectively. Using the added 12 questions, it resulted in 25 experiments. Only the survey from S15 was used, as the survey administered during the S14 study was administered towards the end of the introductory module, thus was regarded as the pilot survey. With the use of the PreSS# web based system for data collection, no student data entry had missing or dirty data, so all the 26 students from S15 were included in each experiment. As with the further analysis of Bergin's main study, 10FCV was used to obtain the prediction accuracy. A Students *t*-test was used to examine if a significant difference was found in accuracy between the baseline result and the experiment result as per section 3. In each experiment the accuracy was recorded.

5.1. Experiment Results

Table 4 shows the experiments where the findings were of value or showed a significant increase in prediction accuracy which is then discussed in section 6.

Experiment ID	**New Model Attributes**	**Sample size (N)**	**Accuracy Increase %**	***p* value**	***t* value**
1	Mathematical grade, programming self-esteem, hours spent playing computer games and age (actual age in years)	26	7.7	< 0.001 ***	15.175
2	Mathematical grade, programming self-esteem and gender.	26	3.85	< 0.001 ***	7.204
3	Mathematical grade, programming self-esteem, hours spent playing computer games and age (dichotomous age value for mature and non-mature students)	26	3.85	< 0.001 ***	7.204
4	Mathematical grade, hours spent playing computer games and the amount of time spent on social media.	26	0	1	0

Table 4: *S14 and S15 study additional survey factors that may have shown to have value where*

** Significant at p < 0.05; ** Significant at p < 0.005; *** significant at p < 0.001;*

5.2. Summary & Discussion

From a total of 25 experiments conducted on S15, 3 factors have emerged, that added value to the PreSS model using some variation of factors resulting in increases in the accuracy of the model or noteworthy findings.

When `age` was added as a factor to PreSS on S15 where `age` was represented in years, experiment ID 1, from Table 4, and in dichotomous form, experiment ID 3, from Table 4, both produced significant increases in accuracy over the original PreSS model. The authors believe that this may have value as an additional factor in the PreSS model based on prior experience. That is that mature students may rate their programming self-esteem lower than its true value, resulting in an false positive prediction, due to this skewing PreSS's strongest performing factor, `programming self-esteem`. Research has shown that there is a positive relationship between age and programming ability/ attainment, (Morrison and Murphy-Hill 2013). Therefore `age` may be able to increase the performance of the programming self-esteem factor if it is shown that mature students may negatively influence this predictor, and age is an appropriate factor to correct this.

The addition of `gender` did not yield an increase in accuracy when introduced into the original PreSS model. However, experiment ID 2 (Table 4), yielded a significant increase in accuracy at ~ 4% on the S15 dataset. Given the uptake of female students in computer science is low, any research that leads to a greater understand or identifies factors in predicting success for female students warrants further investigation.

Although `the time spent on social media` when added to the PreSS model factors, experiment ID 4, Table 4, did not return an increase in accuracy, in nearly all combinations; it did however produce an interesting outcome when it was substituted for the `programming self-esteem` factor. It resulted in an equivalent level of accuracy as the original PreSS model. This result in itself may not be of much value when trying to increase the accuracy of the model, but this direct input of `hours spent on social media` replacing `programming self-esteem`, may be masking some measure of `programming self-esteem`. Also for a web based system, if this hypothesis holds true, the use of social media as a factor may speed up the real time predictions, as the `programming self-esteem` factor requires statistical pre-processing in the form of PCA, whereas social media factor raw value is a direct input into the naïve Bayes algorithm.

6. Conclusions

It is well acknowledged that an early prediction system of student performance in an introductory programming module would be invaluable to educators for developing methodologies and administering interventions as early as possible. PreSS to the author's knowledge (including an extensive review of the literature), is one of the highest prediction systems developed with nearly 80% accuracy, administered at the earliest stage (of four weeks into the module) which has also been validated over a decade. Other prediction models that include the use of early aptitude tests, examinations or programming tests have only shown value at around eight weeks into the module at the first midterm or have not shown any significant prediction value at all (Porter & Zingaro 2014; Tukiainen & Mönkkönen 2002; Evans & Simkin 1989). Even with the high prediction accuracy of nearly 80%, the focus of this study was to identify factors that were recorded in Bergin's main study and in S14 and S15, that were not included in the PreSS model for various reasons but may have value as an additional factor or a replacement factor in an updated PreSS model that may improve upon the 80% prediction accuracy.

With each of these factors investigated and accuracies recorded, it must be noted that the results presented must be interpreted with caution. The main concern for this is due to the size of the data sets used in some experiments and the period in which they were conducted. That said some factors have made some significant increases in the accuracy when combined with the PreSS model. Some hypotheses have been made about the contributing reasons for the presented increase in accuracy.

1. **Final grade predicted by the student at beginning of the module**
   The replacement of the modified Rosenberg Self Esteem factor with a `students predicted final grade at a very early stage of four weeks into the module`, produced the largest increase in accuracy of just under 9%. The author's hypothesis is that this has a direct relationship to a student's programming self-esteem. A noteworthy point is when the modified Rosenberg self-esteem factor was added to the model with the student's final grade prediction, the models accuracy only increased by 7.2%, cautiously suggesting that this may be a conflicting or superior measure of a student's programming self-esteem.

2. **Age**
   The addition of `age` to the model produced the second largest increase in accuracy of just under 8%. This may again be related be programming self-esteem, as the authors believe from experience that mature students generally rate their own programming self-esteem lower than the actual value, with recent research confirming that mature students in fact find it easier to engage with programming related task compared to non-mature students (Lockood et al. 2016), and that there is a positive relationship between age and programming attainment (Morrison and Murphy-Hill 2013). This factor seems to assist the `programming self-esteem` attribute suggesting that it may correct the under rated value that some of the mature students submit to the model.

3. **The difference in the amount of hours a student spent playing computer games before the course to the amount of hours a student spends playing computer games during the course**
   `The hours spent playing computer games during the course` has already been included in the PreSS model, but when this factor is replaced with `the difference in hours a student spent playing computer games before the course commences to the hours spent playing computer games during the course`, the accuracy in the model increase by 7.5%. The author's hypothesis is that this could be related to the effort that a student may put into the course or how much time they may spend studying or practicing programming outside of the normal course hours.

4. **Social Media**

   Even though `the hours spent per day on social media` did not improve the accuracy of PreSS, it produced one of the most interesting findings as it was able to replace the `programming self-esteem factor` while keeping the accuracy at just under 80%. If this factor could replace the modified Rosenberg self-esteem questionnaire that requires four weeks of programming exposure before administration, this factor could allow the PreSS model to be used before the commencement of the course, also as it is not programming specific like the current modified Rosenberg questionnaire, it may also allow the PreSS model to be used in other education or industrial domains. This would make PreSS not only universal in an introductory programming course, but in a much large education / industrial space. Secondly this could significantly contribute to research currently in this space if a direct relationship is confirmed, that is that the time spent on social media may be indicative of a student's programming self-esteem or that students with lower self-esteem gain more from online social media maybe forming a relationship with the time they spend using it. This hypothesis aligns with current research in this space (Steinfield et al. 2008; Shaw and Gant 2002).

These early results warrant a further study with the inclusion of all of these factors on a large longitudinal study, to validate these findings.

7. Epilogue

Based on this work, a large scale study has commenced in the academic year of 2015-16. This international study is currently being conducted in 11 institutions, with the aim of improving upon the PreSS model, with hypothesised increases in the model's accuracy with the introduction of the identified factors from this research. Thirty four questions are being used in the study with over 700 participants. A study of this magnitude will be of significant value to the community and may provide substantial evidence for the work presented here. PreSS# was used to administer the study (Quille et al. 2015).

8. References

Rosenberg, M. (1965). Society and the adolescent self-image. Princeton, NJ: Princeton University Press.

Evans, G. E., & Simkin, M. G. (1989). What best predicts computer proficiency? Communications of the ACM Commun. ACM, 32(11), 1322-1327. doi:10.1145/68814.68817

Smith, L.I. (2002). A tutorial on principal components analysis. Retrieved from http://kybele.psych.cornell.edu/l.pdf. Cornell University, Tutorial.

Tukiainen, M. and Mönkkönen, E. (2002). Programming aptitude testing as a prediction of learning to program. 14th Workshop of the Psychology of Programming Interest Group, Brunel University, June 2002. Pp 47 – 52.

Shaw, L. H., & Gant, L. M. (2002). In Defense of the Internet: The Relationship between Internet Communication and Depression, Loneliness, Self-Esteem, and Perceived Social Support. CyberPsychology & Behavior, 5(2), 157-171. doi:10.1089/109493102753770552

O'Kelly, J., Mooney, A., Bergin, S., Gaughran, P., & Ghent, J. (2004). An overview of the integration of problem based learning into an existing computer science programming module. *Pleasure by Learning*.

Bergin, S., & Reilly, R. (2005). Programming: Factors that influence success. Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education - SIGCSE '05. doi:10.1145/1047344.1047480.

S Bergin, R Reilly. (2005). The influence of motivation and comfort-level on learning to program. 17th Workshop of the Psychology of Programming Interest Group, Sussex University June 2005.

Bergin, S., Reilly, R., & Traynor, D. (2005). Examining the role of self-regulated learning on introductory programming performance. Proceedings of the 2005 International Workshop on Computing Education Research - ICER '05. doi:10.1145/1089786.1089794

Witten, I. H., & Frank, E. (2005). Data mining: Practical machine learning tools and techniques. Amsterdam: Morgan Kaufman.

Bergin, S., & Reilly, R. (2006). Predicting introductory programming performance: A multi-institutional multivariate study. Computer Science Education, 16(4), 303-323. doi:10.1080/08993400600997096

S Bergin. (2006). A computational model to predict programming performance, Maynooth University PhD thesis.

Traynor, D., Bergin, S. and Gibson, J.P. (2006). Automated assessment in CS1. In Proceedings of the 8th Australasian Conference on Computing Education-Volume 52 (pp. 223-228). Australian Computer Society.

Steinfield, C., Ellison, N. B., & Lampe, C. (2008). Social capital, self-esteem, and use of online social network sites: A longitudinal analysis. Journal of Applied Developmental Psychology, 29(6), 434-445. doi:10.1016/j.appdev.2008.07.002

Mooney, O., Patterson, V., O'Connor, M. and Chantler, A. (2010). A study of progression in Irish higher education. Retrieved from http://www.hea.ie/sites/default/files/study_of_progression_in_irish_higher_education_2010.pdf

Morrison, P., & Murphy-Hill, E. (2013). Is programming knowledge related to age? An exploration of stack overflow. 2013 10th Working Conference on Mining Software Repositories (MSR). doi:10.1109/msr.2013.6624008

C.R. Souza. (2014). Accord.Net Framework. Open source Machine learning and Data mining Framework. Retrieved from http://accordframework.net/index.html.

Porter, L., & Zingaro, D. (2014). Importance of early performance in CS1.Proceedings of the 45th ACM Technical Symposium on Computer Science Education - SIGCSE '14. doi:10.1145/2538862.2538912

Quille, K., Bergin, S. and Mooney, A. (2015). PreSS#, a Web-Based Educational System to Predict Programming Performance. International Journal of Computer Science and Software Engineering (IJCSSE), 4(7), pp.178-189.

Bergin, S., Mooney, A., Ghent, J. and Quille, K. (2015). Using Machine Learning Techniques to Predict Introductory Programming Performance. International Journal of Computer Science and Software Engineering (IJCSSE), Volume 4, Issue 12, December 2015 ISSN (Online): 2409-4285

Quille, K., Bergin, S., Mooney, A. (2015). Programming: Factors that Influence Success Revisited and Expanded. International Conference on Engaging Pedagogy (ICEP), College of Computing Technology, Dublin, Ireland, Dec. 3 & 4, 2015

Nolan, K., Mooney, A. and Bergin, S. (2015). Facilitating student learning in Computer Science: large class sizes and interventions. International Conference on Engaging Pedagogy (ICEP), College of Computing Technology, Dublin, Ireland, Dec. 3 & 4, 2015

Liston, M., Frawley, D., Patterson, V. (2016). A Study of Progression in Irish Higher Education. Higher Education Authority. Retrieved from http://www.hea.ie/en/publications/2016

Lockwood, J., Mooney, A., Bergin, S. (2016). A neurofeedback system to promote learner engagement. Human-Computer Interaction (cs.HC). Retrieved from http://arxiv.org/pdf/1607.06232v1.pdf

Appendix 1

The additional survey questions administered during the S14 and S15 studies:

1. **Age Bracket?**
2. **Gender?**
3. **How many hours per day would you play computer games on a mobile device?**
4. **If you play games on a mobile device, what genre of games do you play the most?**
5. **How many hours per day would you play computer games on a Console, PC or laptop?**
6. **If you play games on a console, PC or laptop, what genre of games do you play the most?**
7. **How many hours per day would you use the internet (not including social media or messaging services)?**
8. **What would your primary use of the internet consist of (not including social media or messaging services)?**
9. **How many hours per day would you use a social networking service?**
10. **If you do use social networking, what particular service do you use the most?**
11. **How many hours per day would you use a messaging service?**
12. **If you do use messaging service, what particular service do you use the most?**