Enhancing Computer Students' Academic Performance through Explanatory Modeling

Full Paper

SACLA 2019

Leah Mutanu[1[0000-0001-5164-6424]] and Philip Machoka[2[0000-0001-7008-7111]]

[1,2] United States International University - Africa, 14634-00800, Nairobi, Kenya
{lmutanu[1],pmachoka[2]}@usiu.ac.ke

Abstract. A key challenge facing today universities is the growing attrition rate of computer studies students, attributed to poor academic performance. While extensive research has been conducted on how to enhance student's performance in computer programming, very little research investigates other computer courses, more so in sub-Saharan Africa. This study set out to address this gap by conducting experiments that revealed some of the factors that influence a student's overall academic performance at university through explanatory modeling. Results obtained showed that a student's background in mathematics and their performance in the Introduction to Information Systems course were key in determining performance. Unexpectedly, prior computer skills or secondary school grades had less impact. The strategies identified for enhancing student' performance include an emphasis on building a student's mathematics background, providing a string teaching approach to foundational computing courses, re-structuring of courses in the computer program, and linking courses across the curriculum. Therefore, explanatory modeling creates an opportunity to adopt a proactive approach to enhancing the performance of computer studies students.

Keywords: Computer Science Education, Academic Performance, Explanatory Modeling.

1 Introduction

Globally, Higher Education is the fastest growing segment of post-secondary education. However, this sector faces a myriad of challenges. Key among them is student retention. The challenges, however, vary from region to region. In the USA, for example, the challenge is retaining more women and people from underrepresented minorities (African-Americans, Hispanics, and Native Americans) in computer-related studies. In Kenya, education quality is cited as one of the key challenges facing public university education [1]. Universities also face low enrolment numbers in computer-related degree programs as compared to other degree programs. Another challenge is the high number of student dropout rates. While various reasons exist why students drop out, student academic performance has been identified as one of the biggest drivers [2]. A study by Njoroge et al. [3] investigating student attrition rates in private universities in Kenya revealed that academic performance contributed to increased attrition rates. The study recommended mechanisms to be put in place for early detection of attrition risk supported by technology to ensure students pursue their studies to completion.

The objective of this research was to pursue Njoroge et al. recommendation, that is, to identify factors in a student's learning environment that can serve as indicators of students' academic performance. This research focused on academic factors that impede students' performance in computer-related studies. The remainder of the paper is organized as follows: Section 2 outlines a review of related work; Section 3 discusses the research methodology; Section 4 presents the findings and discussions; Section 5 highlights the key limitations of the study, and Section 6 concludes and makes recommendations for future work.

2 Review of Related Work

High dropout rates are common in Computer related degree programs at universities. Two main causes of the problem as pointed out by Gárcia-Mateos [4] are motivation and complexity of these courses. In determining these problems, however, their study did not factor in the student's prior education background before joining a university. The assumption was that any student who has enrolled in a computer-related program was qualified to undertake the degree course. However, this study revealed that there are prior academic background factors that can influence a student's academic performance. This study did not consider students' social, economic, cultural, and geographic factors because their solution requires a different approach. A study by Kumar [5] found that different demographic groups (economic status, gender, race, major, and type of institution) required different intervention approaches in Computer Science. This study, therefore, set out to explore a students' prior academic background before joining university rather than their demographic groups.

A study conducted by AlMurtadha et al., [6] in Saudi Arabia investigated the key factors that influenced computer studies students' academic performance. Computer studies students are those that are taking a degree program in any computer related studies such as Computer Science and Information Systems. A number of factors were identified in the study. Some of the factors included: age, gender, student major, means of transport to school, parents education level, English proficiency level, sitting position, fear of exam, study-schedule, drug abuse, daily-sleeping hours, twitter account use, sports engagement, hobbies, and community service engagements. The study revealed that a student's major influenced their academic performance. Other factors that influenced performance included their age and gender. A student's academic background in English language and parents' education level partly influenced academic performance while their campus social life did not. From this study, it can be observed that a number of factors influence students' academic performance. These factors can be categorized into social, economic, and academic. While all those factors are significant, this study focuses only on those factors that academic institutions can influence, i.e. the academic factors, through the provision of remedial courses or alternative teaching approaches. They include the students' academic background in Mathematics, Computer and English courses. A research study by Nash [7] points out that Students joining Tertiary education are not always equipped with the skills that can enhance their learning experience. University students are required to use computers to access course materials, write assignments with good grammar, and perform calculations. To be able to cope with these demands, students need a general understanding of computer, language and mathematics skills. Additionally, factored in this study is the students' academic performance at university.

A study done by Garcia and Al-Safadi [8] on factors affecting students' academic performance in computer programming found that classroom management skills were key to improving student performance. The study, however, focused only on the performance of one subject – computer programming and not the entire degree program. More recently, Wang et al., [9] also conducted a similar study that focused on enhancing students' computer programming performance. They recommend a more comprehensive study that focused on other courses offered in the degree program rather than only on one subject. While there is a lot of research on how technology enhances students' performance, very little research focusses on how to enhance student performance in computer-related studies. To the best of the authors' knowledge, most of the existing research in the area identified factors influencing only a single subject that is computer programming, ignoring other subjects in the degree programs. This study addresses this gap by looking at performance in all courses in the entire degree program. Furthermore, there are limited studies in the area focusing on institutions within sub-Saharan Africa.

The work of Garcia and Al-Safadi [8] that investigated factors affecting students' academic performance was based on students' perceptions. They concluded that instructors' classroom management skills such as preparation for the topic and teaching techniques influenced performance. Instructional materials did not have a big influence. A related study that also sought student's reflection revealed that education pedagogy was a key factor in influencing students' performance in computer related courses [10]. In their work, Barlow-Jones and Westhuizen [11], show the relationship between university pre-entry attributes and student performance in computer programming. Their findings revealed that there was a correlation between previous programming experience and performance in programming modules while there was no correlation between the socio-economic status, educational background, high school Mathematics, and English scores. Their study however also collected data from students' opinions rather than examining actual students' scores. This research presents an alternative approach that does not focus on students' perceptions but on student's actual performance. In effect, it paints a more accurate picture of the background factors that influence performance.

Computer students often require a special set of digital skills that other degree programs do not demand. This is especially challenging in developing countries where the digital divide gap is large. The lack of infrastructure and low household income often denies students the opportunity to engage with technology adequately especially during their formative years of schooling. Chikumba [12] highlights the extent of the problem in Malawi where private secondary schools performed better in computer studies than public secondary schools due to poor investment in computers, teaching materials and staff required to deliver the subject. Due to this challenge, it is not a requirement for students joining many universities in developing countries offering computer-related degree programs to have the requisite technical background. However, some courses can be recommended for students intending to take computer related studies. For example, Akinola and Nosiru, [13], recommends *a priori* knowledge of Physics and Mathematics is essential for a student to excel better in Computer related studies. They also mention that better teaching methods and techniques can enhance student performance by changing students' perception of computer-related courses. A different study on factors promoting success in Computer Science by Wilson [14] revealed the Computer Science performance predictive factors as the student's comfort level and math background. No significant influence was based on gender differences. The performance in the introduction to programming courses also had a positive influence on success. Research on how language influences performance revealed that contrary to the generally accepted view that achievement in high school mathematics courses is the best individual predictor of success in undergraduate Computer Science, success in English correlates better with the actual performance [15]. Their study, however, was conducted in a social context where many students were not native English speakers.

In their study, Akinola and Nosiru, [13] used *Fuzzy sets* operation approach to arrive at the factors that influence performance. They posit that selecting the factors that influence students' performance in computer programming is not an easy task as it involves human decision making which can be imprecise or subjective. This problem cannot be handled effectively by probability theories. Fuzzy set handles problems that involve the need to seek consensus among many decision makers. Students gave their opinions according to their own criteria for each factor by selecting a value and the union of their evaluations to all the currently available alternatives was represented in the form of a fuzzy set. They state that the approach helped in eliminating outlier *decision-making*, hence, presenting a more accurate and reliable result. Although faculty (instructors) have traditionally found ways of identifying performance challenges, there is a need to enhance the process through innovative ways of modeling data. The objective of the research was to model student data, captured over a period, with a view of providing causal explanations to performance for purposes of early intervention. Data mining techniques have been used extensively in research to develop early warning systems for risk aversion. Within the health domain, early warning systems have been defined as surveillance systems that collect information on epidemic-prone issues such as diseases in order to trigger prompt public interventions [16]. Fuzzy logic is used to map risk patterns. The research described in this paper therefore aimed at modeling computer students' academic performance in order to find ways of enhancing it.

3 Methodology

The study was conducted at a private university in Kenya. The university offers two computer-related degree programs, *Computer Science* and an *Information Systems*. Both programs had the same pre-entry admission requirements for secondary school final exam grade point average (GPA), however, the Computer Science program required higher secondary school scores in Mathematics and Physics for admission into the program. Secondary data was collected on students taking both programs for a period of five years. Only data recorded in the Students' Information Systems (SIS) could be used i.e. the student's GPA at secondary school, placement tests score (skills assessment score on admission to university), student major, gender, current year of study, year of admission, current GPA for courses taken so far at the university, and the country of origin. The research focused on academic factors that the institutions could influence or use to influence practices at the university to enhance the students' academic performance. This research aimed at modeling computer students' academic performance in order to find ways of enhancing it. It achieved this through two research objectives; (i) Identification of academic factors influencing student academic performance in computer-related studies, and (ii) Establishing ways of enhancing the computer-related degree programs curriculum to address academic factors influencing student academic performance in computer-related studies.

The data set constituted 6000 students who had joined the university over a five-year period (2014 – 2018). A random sample was required for analysis. The study used the formula in equation 1 was used to calculate the study sample size.

$$SampleSize = \frac{\frac{z^2 \text{ x } p\ (1-p)}{e^2}}{1+(\frac{z^2 \text{x } p(1-p)}{e^2 N})} \quad (1)$$

N = population size • e = Margin of error (percentage in decimal form) • z = z-score • p = sample proportion

A confidence level of 95% was used which requires a Z-score of 1.96. The z-score is the number of standard deviations a given proportion is away from the mean. A 3% margin of error was selected. The sample size was therefore obtained as 907. The study, therefore, picked 1000 student from a population of 6000 students' records to work with. This sample was picked to act as a test set for experiments in predictive modeling however, future experiments will focus on using the entire sample size. However, through the data cleaning process, several records were discarded giving a final sample size of 858 students. This number was found to be acceptable as it still gave the desired 3% margin of error.

For ethical purposes, identifying features such as the students' ID number, students name and addressed were filtered from the data. An SQL script was written and used for the extraction to ensure that all the data was extracted anonymously. The resultant data was placed on an excel spreadsheet for data cleaning purposes and subsequent analysis. To ensure that the data was an accurate representation of the five years, the year of admission was also retrieved. Cleaning of the data was done using an MS-Excel

Spreadsheet enable records with data entry errors or missing data to be dropped. From the data cleaning exercise, 142 records were dropped. Data Analytics was then conducted with open source data mining tools Weka [17] and R [18]. The IBM SPSS statistical analysis tool [19] was also used to explore and validate the results. The use of several tools enriched the process by validating the results obtained. Clustering and Decision tree algorithms were used to classify the data. From the Data mining exercise, patterns in the data were identified and used for explanatory modeling. The research design was descriptive where the characteristics of correlations between two or more entities were explored and visualization techniques used to represent the data. Quantitative research techniques were used to emphasize objective measurements and the statistical, mathematical, or numerical analysis of data collected through surveys, or by manipulating pre-existing statistical data using computational techniques. In this research data was gathered and used to generalize across groups of students to explain performance.

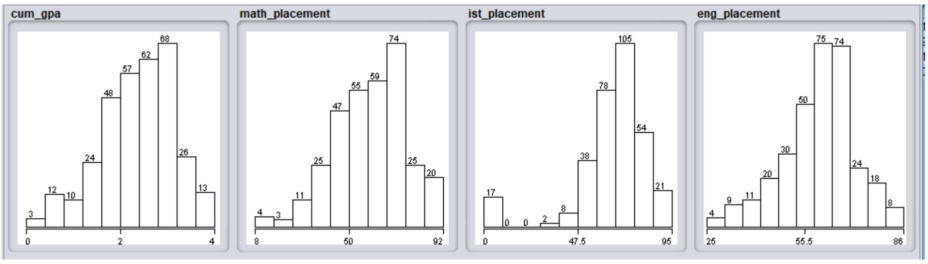
4 Findings and Discussions

This section identifies patterns in the data for purposes of modeling the performance of students taking undergraduate computer-related degree programs at the university. It describes the techniques used and the results obtained.

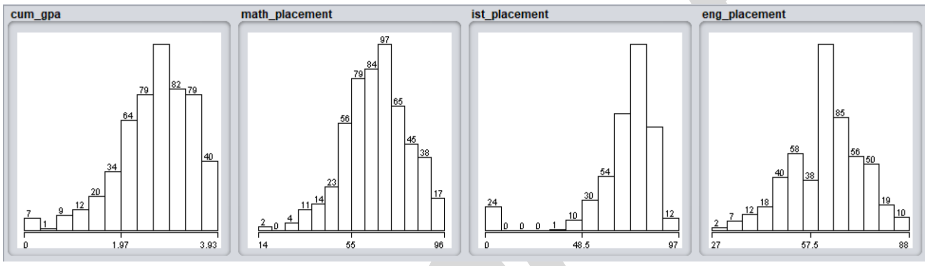
4.1 Factors Influencing Cumulative GPA

The research set out to identify some academic factors that influence students' performance based on existing literature described in section 2. Specifically, the research analyzed factors such as the student's final GPA at secondary school, placement tests score on admission to the university, students major, gender, current year of study, year of admission, test scores for courses done at the university, and the country of origin. Placement tests were given to students upon joining the university. Scores from these tests are used to assess student's prior knowledge in Mathematics, English, and computing skills upon admission. Students who do not meet the cut off points are required to take remedial classes for a semester in the respective courses. The prior mathematics knowledge that computer programs require include basic algebra and statistics. The degree programs also offer additional mathematics courses such as discrete mathematics and algebra. Most of the computer courses offered at the university have Mathematics courses as a pre-requisite. These include courses such as Data Structures, Decision Analysis, and Data analytics. However, the first computer programming course – Fundamentals to Programming Logic - assumes no prior knowledge of programming and does not have any mathematics prerequisites. The focus of the course is on imparting procedural programming skills.

Distribution of the frequencies for the data used in this study is represented in Fig. 1. The figure shows that there were more male students in both programs. Given a pass mark of 60% majority of the students managed to pass the placement tests in IST (Computer placement) and English. The performance in Mathematics was not as good, however, it should be noted that Computer Science competes with other Science and Engineering disciplines for students with a good Mathematics Background.

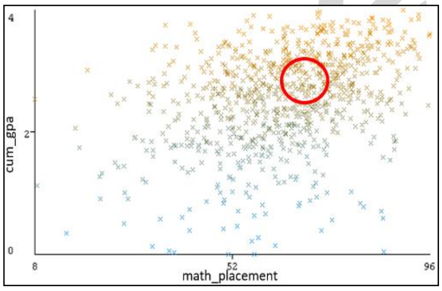


(a) Computer Science Program

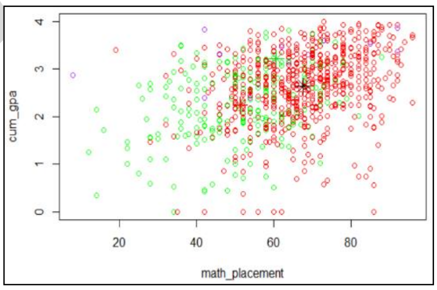


(b) Information Systems Program

Fig. 1. Sample Frequency Distributions of the Datasets used for the study.



(a) EM Clustering Algorithms



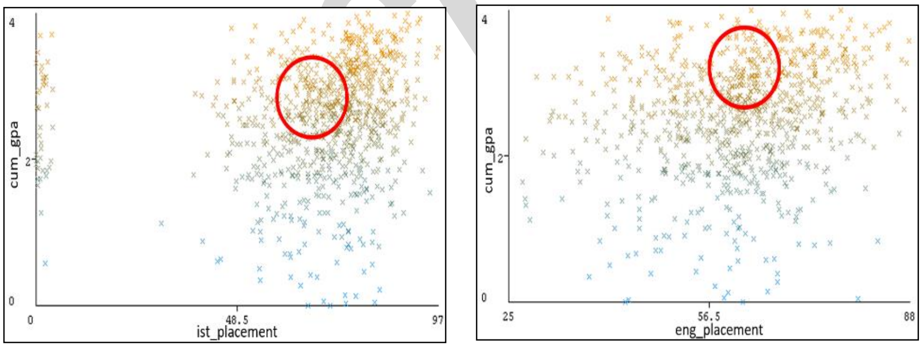
(b) K-Mean Clustering Algorithm

Fig. 2. Sample Frequency Distributions of the Datasets used for the study.

The data was analyzed using clustering algorithms to give an indication of which factors influenced the cumulative GPA. For this research Expectation–Maximization (EM) clustering technique [20] was preferred because it provides better optimization than distance-based or hard membership algorithms, such as K-Means [21]. EM easily accommodates categorical and continuous data fields, thus, making it the most effective technique available for proper probabilistic clustering. K-Means clustering is a method of vector quantization that partitions observations into clusters based on the nearest mean, serving as a prototype of the cluster. K-Means was also used for purposes of validating the results. The Data had to be converted into numeric representations to perform K-mean clustering.

Determining the Significance of Factors Used. Cluster analysis showed that some variables have a bigger impact on the student's cumulative GPA than others. For example, Fig. 2 shows that a higher math placement score resulted in a higher student's cumulative GPA. Both EM and K-Means clustering algorithms gave the same results. Similar results were also observed for the student's year of study where students at their third and fourth years of study had higher cumulative GPAs than those in their first and second years of study.

However, similar tests showed that the English and Information Systems Technology (IST) placement scores had less impact on the student's cumulative GPA as depicted in Fig. 3. This showed that students do not require prior knowledge in Computers to perform well in computer-related degree programs. Prior knowledge in computers refers to computer literacy skills. The cluster centers of high cumulative GPA scores were close to the 50% test scores. However, further investigation to understand why factors such as previous computer knowledge were not influential contrary to popular belief is required. Similar results were also observed for the student's secondary school final exam GPA despite the expectation that students who performed well in high school would also perform well at university. The secondary school final GPA's did not have a big impact on the cumulative GPA scores at university. This is expected given that the students selected for the program had similar high school final GPA scores based on the pre-entry requirements.



(a) IST Placement Test *(b) English Placement Test*

Fig. 3. Impact of IST and English placement scores.

Other factors examined such as the student's gender, degree program (Computer Science or Information Systems), and country of origin had almost no visible impact at all when a cluster analysis was run. These findings concur with the findings of a similar study on factors promoting success in Computer Science by Wilson [14]. However, further investigation to understand why factors such as gender were not influential contrary to popular belief is required. The Independent Samples t-test [22] was used to compare the difference in cumulative GPA means where two independent groups existed such as male and female students or Computer and Information Systems degree. The purpose of the t-test was to determine whether the difference between the two groups was statistically significant. A t-test conducted on the student's cumulative GPA versus student's gender showed that the difference in cumulative GPA was not statistically significant between the two groups as tabulated in Table 1 (a) The results $t(323.035) = -7.300$, $p = 0.000$, show that there is no statistically significant difference in the variances between the two groups. The p-value is below the critical significance level of 0.05. Running a similar t-test on the student's degree program versus the cumulative GPA also confirmed that there was no statistically significant difference in the GPA reported between the two groups (Table 1 (b)). The results in the table, $t(664.239) = 5.017$, $p = 0.000$, show that there is no statistically significant difference in the variances between the two groups. The p-value is below the critical significance level of 0.05. The results statistically confirmed that gender and the degree program had no significant influence on the student's cumulative GPA.

Table 1. Determining the significance of gender and degree program.

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
cum_gpa	Equal variances assumed	3.535	.060	-6.838	855	.000	-.427573882	.0625259324	-.550296183	-.304851581
	Equal variances not assumed			-7.300	323.035	.000	-.427573882	.0585698688	-.542800423	-.312347341

(a) Student Gender

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
cum_gpa	Equal variances assumed	.822	.365	5.052	855	.000	.2713842314	.0537198889	.1659459258	.3768225370
	Equal variances not assumed			5.017	664.239	.000	.2713842314	.0540949558	.1651665244	.3776019384

(b) Student Degree

It is worth noting that further analysis showed that there were more male students admitted in the two programs although female students had comparatively higher cumulative GPA's as illustrated in Table 2 (a). The difference, however, was not statistically significant. Both computer-related degree programs had low female enrolments. Additionally, Students in the Information Systems Degree program were more than those in the Computer Science program. This was expected because the former program had been offered for a longer period. However, it was observed that the Computer Science program had students with slightly higher cumulative GPA than those from the Information Systems program as illustrated in Table 2 (b) although the difference was also not statistically significant.

Table 2. Impact of gender and degree program on cumulative GPA.

	Male	Female
Mean Cumulative GPA	2.46533	2.892108
Total	673	185

(a) *Impact of gender and degree program on cumulative GPA.*

	Information Systems	Computer Science
Mean Cumulative GPA	2.659755	2.387734
Total	535	323

(b) *Impact of gender and degree program on cumulative GPA.*

Table 3. Statistical significance using Pearson Correlation.

		cum_gpa	Math placement
cum_gpa	Pearson Correlation	1	.364**
	Sig. (2-tailed)		.000
	N	857	857
Math placement	Pearson Correlation	.364**	1
	Sig. (2-tailed)	.000	
	N	857	857

**. Correlation is significant at the 0.01 level (2-tailed).

(a) Correlation (Math placement test and cumulative GPA

		cum_gpa	Ist placement
cum_gpa	Pearson Correlation	1	.146**
	Sig. (2-tailed)		.000
	N	857	857
ist_placement	Pearson Correlation	.146**	1
	Sig. (2-tailed)	.000	
	N	857	857

**. Correlation is significant at the 0.01 level (2-tailed).

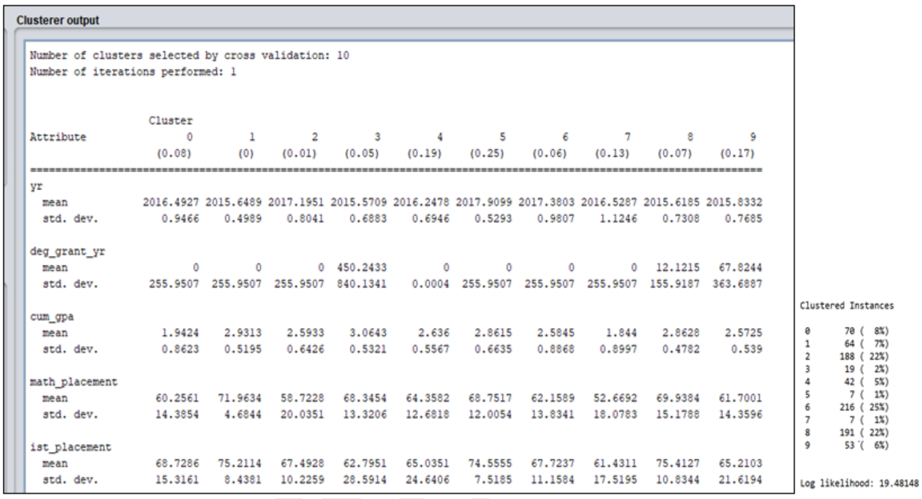
(b) Correlation (IST placement test and cumulative GPA

To assess the significance of each of these factors a Pearson Correlation [23] analysis was conducted. The bivariate Pearson Correlation produces a sample correlation coefficient, r, which measures the strength and direction of linear relationships between pairs of continuous variables. The Pearson Correlation was used to present the statistical evidence for variables that impact the cumulative GPA or not. The results of the correlation confirmed the observations made from the clustering tests that mathematics placement tests scores influenced the student's cumulative GPA. There is a statistically significant correlation between Mathematics placement test scores and a student's cumulative GPA. The Pearson's r is positive indicating that when one variable increases the second variable also increases. The Sig (2-Tailed) value is less than or equal to 0.05 showing that the relationship is statistically significant. However, the correlation between a student's IST placement test score and a student's cumulative GPA was not statistically significant (Table 3) because the value of the Pearson's coefficient r is close to zero. The correlation between a student's country of origin and a student's cumulative GPA was not statistically significant.

Mining Patterns in the data. The results obtained so far identified variables that were significant in determining a student's cumulative GPA and the extent to which each variable impacted the cumulative GPA. They, however, do not describe patterns in the data showing how all the variables jointly impacted the cumulative GPA. This is useful when modeling relationships between variables. In order to describe patterns in the data, further analysis of all the variables was required to show how they influence each other. For this task, Clustering and decision tree algorithms were applied to the data. When a cluster analysis was conducted with the EM clustering algorithm, three large clusters were identified. The results presented in Fig. 4 show three significant clusters labelled 2 (GPA-2.6), 6 (GPA-2.5), and 8 (GPA-2.1) as having the highest cluster densities i.e. 22%, 25% and 22% respectively. These clusters are groups of students with similar characteristics. The math placement scores for each cluster were observed to be 59% 53% and 70% respectively (Fig. 4). The high densities showed that majority of the students had these characteristics. The figure further shows two clusters labeled 1 and 3 with high cumulative GPA scores of 2.9 and 3.1 respectively. The Mathematics placement scores are also higher at 71% and 68% respectively. This confirms the earlier results that math placement score influenced the cumulative GPA. Although clustering showed the existence of relationships, it is difficult to tell at a glance what the relationships were. Additional analysis was required to describe the factors that formed each cluster.

Running decision tree algorithms on the data revealed the relationship between different variables as illustrated in Fig. 5. From the results, it can be observed that the Introduction to Information Systems course (IST1020), the year of study (cl), and the math placement test played the most significant role in determining a student's cumulative GPA. The Introduction to Information Systems course imparts general computer literacy skills to students. It does not include any computer programming. A few second level courses were also found to have an impact, that is Computer Organization (taken by the Information Systems degree students) and Computer Networks (taken by the

Computer Science degree students). The Computer Organization course exposed students to computer architecture and assembly language while the Computer Networks course introduced students to data communication protocols and devices. Both courses were taken by students as soon as they completed the introduction to Information Systems course. The higher-level courses had no impact. The findings brought out the importance of laying a strong foundation for students during their initial courses in computing. The approach to the foundation classes served to either propel students to success or failure. It was observed that seasoned instructors who were Professors in the department taught higher-level courses or graduate programs while early career instructors and part-time instructors were left to handle introductory level courses in the department. This approach needed to change to enhance students' performance.


Clusterer output

Number of clusters selected by cross validation: 10
Number of iterations performed: 1

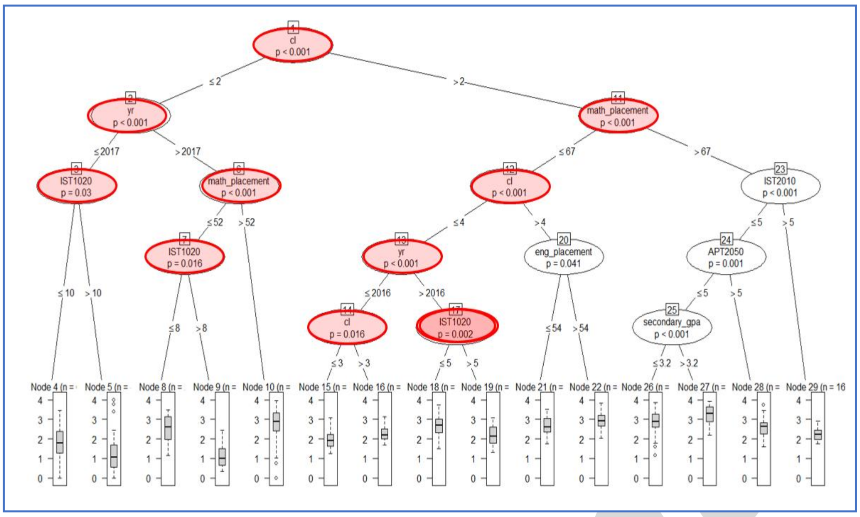
Attribute	Cluster 0 (0.08)	1 (0)	2 (0.01)	3 (0.05)	4 (0.19)	5 (0.25)	6 (0.06)	7 (0.13)	8 (0.07)	9 (0.17)
yr										
mean	2016.4927	2015.6489	2017.1951	2015.5709	2016.2478	2017.9099	2017.3803	2016.5287	2015.6185	2015.8332
std. dev.	0.9466	0.4989	0.8041	0.6883	0.6946	0.5293	0.9807	1.1246	0.7308	0.7685
deg_grant_yr										
mean	0	0	0	450.2433	0	0	0	0	12.1215	67.8244
std. dev.	255.9507	255.9507	255.9507	840.1341	0.0004	255.9507	255.9507	255.9507	155.9187	363.6887
cum_gpa										
mean	1.9424	2.9313	2.5933	3.0643	2.636	2.8615	2.5845	1.844	2.8628	2.5725
std. dev.	0.8623	0.5195	0.6426	0.5321	0.5567	0.6635	0.8868	0.8997	0.4782	0.539
math_placement										
mean	60.2561	71.9634	58.7228	68.3454	64.3582	68.7517	62.1589	52.6692	69.9384	61.7001
std. dev.	14.3854	4.6844	20.0351	13.3206	12.6818	12.0054	13.8341	18.0783	15.1788	14.3596
ist_placement										
mean	68.7286	75.2114	67.4928	62.7951	65.0351	74.5555	67.7237	61.4311	75.4127	65.2103
std. dev.	15.3161	8.4381	10.2259	28.5914	24.6406	7.5185	11.1584	17.5195	10.8344	21.6194

Clustered Instances

0	70	(8%)
1	64	(7%)
2	188	(22%)
3	19	(2%)
4	42	(5%)
5	7	(1%)
6	216	(25%)
7	7	(1%)
8	191	(22%)
9	53	(6%)

Log likelihood: 19.48148

Fig. 4. Mining patterns through clustering.



Key: IST2010 – Computer Organization, APT2050 – Computer Networks, IST1020 – Introduction to Information Systems

Fig. 5. Classification of factors influencing cumulative GPA using decision trees.

Table 4. Relationship between courses in a degree program.

COMPUTER SCIENCE

	YEAR 1 & YEAR 2														YEAR 3 & YEAR 4								
COMPUTER SCIENCE – YEAR 1 & YEAR 2	1	0.268	0.159	0.139	0.4	0.285	0.187	0.095	0.119	0.117	-0.023	0.135	0.104	0.062	0.006	0.078	0.144	0.083	-0.004	0.118	0.007	0.074	-0.024
	0.268	1	0.492	0.229	0.388	0.536	0.275	0.251	0.346	0.33	0.009	0.274	0.353	0.276	0.082	0.311	0.317	0.256	0.098	0.244	0.025	0.216	0.232
	0.159	0.492	1	0.22	0.339	0.399	0.3	0.255	0.46	0.265	0.053	0.235	0.344	0.179	0.215	0.239	0.355	0.243	0.165	0.189	0.031	0.139	0.069
	0.139	0.229	0.22	1	0.408	0.263	0.426	0.385	0.385	0.327	0.022	0.395	0.347	0.185	0.03	0.334	0.307	0.201	0.201	0.292	0.024	0.234	0.12
	0.4	0.388	0.339	0.408	1	0.363	0.319	0.216	0.304	0.284	0.013	0.265	0.297	0.15	0.086	0.229	0.247	0.186	0.184	0.231	0.07	0.163	0.057
	0.285	0.536	0.399	0.263	0.363	1	0.336	0.269	0.26	0.411	0.018	0.342	0.344	0.277	5E-04	0.26	0.351	0.255	0.097	0.243	0.05	0.13	0.099
	0.187	0.275	0.3	0.426	0.319	0.336	1	0.366	0.428	0.421	0.053	0.337	0.334	0.236	0.088	0.303	0.344	0.284	0.157	0.261	0.117	0.098	0.114
	0.095	0.251	0.255	0.385	0.216	0.269	0.366	1	0.472	0.469	0.11	0.409	0.43	0.387	0.053	0.43	0.379	0.315	0.208	0.411	0.114	0.175	0.116
	0.119	0.346	0.46	0.385	0.304	0.26	0.428	0.472	1	0.502	0.022	0.365	0.4	0.336	0.181	0.393	0.414	0.384	0.278	0.255	0.095	0.227	0.104
	0.117	0.33	0.265	0.327	0.284	0.411	0.421	0.469	0.502	1	0.017	0.397	0.459	0.422	0.041	0.408	0.461	0.372	0.335	0.305	0.068	0.133	0.075
	-0.023	0.009	0.053	0.022	0.013	0.018	0.053	0.11	0.022	0.017	1	0.09	0.015	0.062	-0.011	0.083	-0.014	-0.036	0.083	0.124	0.144	-0.03	-0.034
	0.135	0.274	0.235	0.395	0.265	0.342	0.337	0.409	0.365	0.397	0.09	1	0.367	0.271	0.028	0.409	0.366	0.286	0.148	0.28	0.052	0.135	0.094
	0.104	0.353	0.344	0.347	0.297	0.344	0.334	0.43	0.4	0.459	0.015	0.367	1	0.348	0.057	0.323	0.378	0.325	0.139	0.267	0.076	0.111	0.162
	0.062	0.276	0.179	0.185	0.15	0.277	0.236	0.387	0.336	0.422	0.062	0.271	0.348	1	0.054	0.388	0.327	0.423	0.337	0.381	0.102	0.187	0.261
COMPUTER SCIENCE – YEAR 3 & YEAR 4	0.006	0.082	0.215	0.03	0.086	5E-04	0.088	0.053	0.181	0.041	-0.011	0.028	0.057	0.054	1	0.066	0.095	0.237	0.041	0.086	0.35	-0.013	-0.015
	0.078	0.311	0.239	0.334	0.229	0.26	0.303	0.43	0.393	0.408	0.083	0.409	0.323	0.388	0.066	1	0.37	0.407	0.298	0.402	0.073	0.224	0.188
	0.144	0.317	0.355	0.307	0.247	0.351	0.344	0.379	0.414	0.461	-0.014	0.366	0.378	0.327	0.095	0.37	1	0.481	0.202	0.359	0.119	0.234	0.216
	0.083	0.256	0.243	0.201	0.186	0.255	0.284	0.315	0.384	0.372	-0.036	0.286	0.325	0.423	0.237	0.407	0.481	1	0.225	0.379	0.098	0.212	0.191
	-0.004	0.098	0.165	0.201	0.184	0.097	0.157	0.208	0.278	0.335	0.083	0.148	0.139	0.337	0.041	0.298	0.202	0.225	1	0.226	0.047	0.243	0.089
	0.118	0.244	0.189	0.292	0.231	0.243	0.261	0.411	0.255	0.305	0.124	0.28	0.267	0.381	0.086	0.402	0.359	0.379	0.226	1	0.188	0.239	0.25
	0.007	0.025	0.031	0.024	0.07	0.05	0.117	0.114	0.095	0.068	0.144	0.052	0.076	0.102	0.35	0.073	0.119	0.098	0.047	0.188	1	-0.017	-0.019
	0.074	0.216	0.139	0.234	0.163	0.13	0.098	0.175	0.227	0.133	-0.03	0.135	0.111	0.187	-0.013	0.224	0.234	0.212	0.243	0.239	-0.017	1	0.403
	-0.024	0.232	0.069	0.12	0.057	0.099	0.114	0.116	0.104	0.075	-0.034	0.094	0.162	0.261	-0.015	0.188	0.216	0.191	0.089	0.25	-0.019	0.403	1

(a) Computer Science Degree

INFORMATION SYSTEMS

	YEAR 1 & YEAR 2									YEAR 3 & YEAR 4															
INFORMATION SYSTEMS – YEAR 1 & YEAR 2	1	0.078	0.079	0.01	0.02	-0.017	-0.018	-0.017	0.033	0.029	0.038	-0.02	-0.02	0.063	-0.003	0.011	0.102	0.007	0.018	0.013	0.096	0.007	-0.03	-0.009	0.073
	0.078	1	0.204	0.076	0.12	0.055	-0.024	0.06	0.118	0.062	-0.036	-0.025	-0.012	0.044	0.004	-1E-04	0.007	-0.009	-0.017	-0.008	0.007	0.046	-0.017	-0.002	-0.017
	0.079	0.204	1	0.004	0.127	-0.039	-0.055	0.032	0.131	0.136	-0.01	-0.019	-0.042	0.126	0.033	0.13	0.114	-0.012	0.038	0.043	0.076	0.075	-0.034	0.112	0.027
	0.01	0.076	0.004	1	0.416	0.002	0.217	0.282	0.333	0.478	0.058	0.294	0.081	0.223	0.316	0.209	0.135	0.021	0.28	0.11	0.159	0.233	0.013	0.039	0.048
	0.02	0.12	0.127	0.416	1	-0.002	0.067	0.329	0.547	0.418	-0.037	0.291	0.045	0.247	0.329	0.4	0.216	5E-04	0.328	0.221	0.197	0.3	0.026	0.104	0.114
	-0.017	0.055	-0.039	0.002	-0.002	1	-0.015	-0.003	-0.008	0.015	-0.008	0.106	-0.003	-0.009	0.009	0.016	-0.009	-0.004	0.055	-0.009	0.05	0.016	-0.006	-0.006	-0.004
	-0.018	-0.024	-0.055	0.217	0.067	-0.015	1	0.219	0.029	0.141	0.167	0.082	-0.018	-0.033	0.132	-0.005	-0.045	0.057	0.111	-0.054	-0.034	0.059	0.026	-0.03	-0.025
	-0.017	0.06	0.032	0.282	0.329	-0.003	0.219	1	0.268	0.326	0.025	0.407	0.035	0.276	0.419	0.379	0.261	-0.003	0.467	0.219	0.184	0.469	0.035	0.189	0.058
	0.033	0.118	0.131	0.333	0.547	-0.008	0.029	0.268	1	0.366	-0.027	0.218	0.029	0.149	0.264	0.293	0.286	-0.017	0.28	0.275	0.232	0.178	0.004	0.053	0.052
INFORMATION SYSTEMS – YEAR 3 & YEAR 4	0.029	0.062	0.136	0.478	0.418	0.015	0.141	0.326	0.366	1	0.003	0.433	0.071	0.237	0.297	0.333	0.337	-0.012	0.288	0.326	0.286	0.332	0.025	0.081	0.053
	0.038	-0.036	-0.01	0.058	-0.037	-0.008	0.167	0.025	-0.027	0.003	1	-0.032	-0.01	-0.027	0.03	-0.036	-0.031	0.113	0.003	-0.033	-0.018	-0.033	0.009	-0.022	-0.014
	-0.02	-0.025	-0.019	0.294	0.291	0.106	0.082	0.407	0.218	0.433	-0.032	1	0.244	0.278	0.462	0.474	0.365	-0.015	0.455	0.275	0.357	0.425	0.061	0.193	0.129
	-0.02	-0.012	-0.042	0.081	0.045	-0.003	-0.018	0.035	0.029	0.071	-0.01	0.244	1	0.021	0.154	0.19	0.025	-0.005	-0.005	0.023	0.168	0.033	6E-04	0.018	0.112
	0.063	0.044	0.126	0.223	0.247	-0.009	-0.033	0.276	0.149	0.237	-0.027	0.278	0.021	1	0.369	0.468	0.387	-0.012	0.346	0.261	0.535	0.397	0.132	0.199	0.158
	-0.003	0.004	0.033	0.316	0.329	0.009	0.132	0.419	0.264	0.297	0.03	0.462	0.154	0.369	1	0.415	0.438	0.006	0.364	0.368	0.312	0.423	0.091	0.168	0.093
	0.011	-1E-04	0.13	0.209	0.4	0.016	-0.005	0.379	0.293	0.333	-0.036	0.474	0.19	0.468	0.415	1	0.558	-0.009	0.431	0.463	0.512	0.414	0.061	0.176	0.272
	0.102	0.007	0.114	0.135	0.216	-0.009	-0.045	0.261	0.286	0.337	-0.031	0.365	0.025	0.387	0.438	0.558	1	-0.014	0.444	0.583	0.543	0.409	0.08	0.237	0.269
	0.007	-0.009	-0.012	0.021	5E-04	-0.004	0.057	-0.003	-0.017	-0.012	0.113	-0.015	-0.005	-0.012	0.006	-0.009	-0.014	1	-0.002	-0.015	2E-04	-0.007	-0.01	-0.01	-0.006
	0.018	-0.017	0.038	0.28	0.328	0.055	0.111	0.467	0.28	0.288	0.003	0.455	-0.005	0.346	0.364	0.431	0.444	-0.002	1	0.312	0.339	0.348	0.097	0.261	0.164
	0.013	-0.008	0.043	0.11	0.221	-0.009	-0.054	0.219	0.275	0.326	-0.033	0.275	0.023	0.261	0.368	0.463	0.583	-0.015	0.312	1	0.375	0.261	0.177	0.158	0.234
	0.096	0.007	0.076	0.159	0.197	0.05	-0.034	0.184	0.232	0.286	-0.018	0.357	0.168	0.535	0.312	0.512	0.543	2E-04	0.339	0.375	1	0.393	0.102	0.185	0.134
	0.007	0.046	0.075	0.233	0.3	0.016	0.059	0.469	0.178	0.332	-0.033	0.425	0.033	0.397	0.423	0.414	0.409	-0.007	0.348	0.261	0.393	1	0.08	0.147	0.074
	-0.03	-0.017	-0.034	0.013	0.026	-0.006	0.026	0.035	0.004	0.025	0.009	0.061	6E-04	0.132	0.091	0.061	0.08	-0.01	0.097	0.177	0.102	0.08	1	0.11	0.036
	-0.009	-0.002	0.112	0.039	0.104	-0.006	-0.03	0.189	0.053	0.081	-0.022	0.193	0.018	0.199	0.168	0.176	0.237	-0.01	0.261	0.158	0.185	0.147	0.11	1	0.114
	0.073	-0.017	0.027	0.048	0.114	-0.004	-0.025	0.058	0.052	0.053	-0.014	0.129	0.112	0.158	0.093	0.272	0.269	-0.006	0.164	0.234	0.134	0.074	0.036	0.114	1

(b) Information Systems Degree

$r \geq 0.9$, $0.8 \leq r < 0.9$, $0.7 \leq r < 0.8$, $0.6 \leq r < 0.7$, $0.5 \leq r < 0.6$, $0.4 \leq r < 0.5$, $0.3 \leq r < 0.4$, $0.2 \leq r < 0.3$, $r < 0.2$

(Key - Pearson's correlation coefficient r)

The study also sought to establish how courses taught at various levels influenced each other. To achieve this a Pearson Correlation was performed across the various courses offered. The courses are mapped against each other and the Pearson Correlation values provided. The lower-level courses are courses offered in year 1 and year 2 indicated in green in the column and row headings. The higher-level courses are courses offered in year 3 and 4 indicated in blue in the column and row headings. The results presented by the correlation matrix in Table 4 (a) showed that for the Computer Science program lower level courses appeared to influence each other positively (Pearson Correlation value above 0.4), such that a high grade in one would most likely mean a high grade in the other. A closer examination revealed that this occurred between related courses such as *Computer Organizations* and *Operating Systems* or *Web Design* and *Computer Programming*. Related courses in this context are considered as courses that are a pre-requisite of the other. For example, *Computer Organizations* is a pre-requisite course for *Operating systems* and *Computer Programming* is a pre-requisite for *Web Design.* Unexpected relationships identified were between *Computer Organization* and *Computer Programming* which are courses that are not indicated as prerequisites of each other. This, however, might be attributed to the fact that *Assembly Language Programming* was part of the *Computer Organization* course.

For the Information Systems program, fewer courses seemed to influence each other as depicted by the presence of more white cells in Table 4 (b) (Pearson Correlation value less than 0.2). This was a possible indication of the lack of proper linking when teaching courses across the curriculum. Some significant relationships, however, existed among related higher-level courses such as System Analysis and Design, Object Oriented Programming, and Database Management Systems or Decision Analysis, Data Structures and Algorithms, and Mobile Programming. The relationships were expected because the courses were related and, therefore had been stated as pre-requisites of each other in the curriculum. One unexpected relationship, however, was identified between the Digital Laboratory and Object-Oriented Programming where students who did well in Digital Laboratory also did well in Object-Oriented Programming and vice versa. The two courses had distinct content and, therefore, had not been indicated as pre-requisites of each other in the curriculum. Further research is required to investigate the cause of this relationship.

4.2 Strategy for Enhancing Performance

The experiments described in Section 4.1 provided insights into weaknesses in the current degree programs offered at the university. Results from this study can offer an opportunity to review the programs and align courses properly to enhance student's academic performance in computer-related programs. Strategies on the courses to be considered when admitting students into the program, the courses to be emphasized or restructured in the curriculum, and ways of reducing attrition rates through performance are discussed in this section.

Preference on Student Mathematics Background. From the analysis conducted, the findings showed that the results of the Mathematics placement test scores had a positive impact on the student's cumulative GPA. Students with a strong Mathematical background performed better than students who were weaker in the subject. This was, however, not the case with the computing and English skills backgrounds. Courses in computer-related studies, therefore, need to admit students with a strong mathematical background. As mentioned earlier, Computer Science and Information Systems competes with Science and Engineering disciplines for students with a good Mathematics background. Therefore, an alternative strategy to attract students with excellent Mathematics background needs to be sought. Additionally, there is a need to add mathematics courses such as Discrete Mathematics, Formal Logic, Differential, and Integral Calculus to make up for weaknesses in the required areas.

Emphasis on Introduction to computers course. Another key area that influenced the student's cumulative GPA is their performance in the Introduction to Information Systems course. Students who performed well in this course ended up with good overall cumulative GPA scores. Emphasis, therefore, needs to be put on how the courses are taught to develop student's interest as well as lay a good solid foundation for the two computing programs. Basic skills that will be required in advanced courses should be instilled at the foundation level. Further research on the specific skills required at this level is required. Such skills should, therefore, be integrated into the Introduction to Information Systems course. Further senior and experienced instructors who traditionally teach higher level courses and graduate courses should be encouraged to take lower-level courses to ground and mentor the students well during foundation courses.

Structure of the curriculum. The results further showed that there was a significant positive correlation among courses in the Computer Science degree. Students who did well in the introductory programming courses also did well in the advanced programming course. This was, however, not the case for the Information Systems degree program. This was attributed to the fact that the Computer Science degree was developed after the Information Systems degree and, therefore structured better to accommodate lessons learned from the Information Systems degree. The structure of the Information Systems degree program has since been revised to address some of these challenges. It is envisaged that the results of these structural changes will reflect in future studies.

Linking Courses across the curriculum. The researchers also noted the need to relate courses when teaching the program as well as re-examining prerequisites in the program. While relationships were observed among programming courses, the same was not apparent among networking courses or computer security courses. Students who did well in the lower-level networking and security courses should also be able to do well in similar courses at the advanced level. The absence of well-defined relationships could, however, be due to the lack of courses at a higher level to advance courses in those tracks. Further research is required to establish the reason for the lack of these relationships.

Prediction modeling for an early warning system. Machine Learning and Data Mining techniques can be combined to identify weaknesses in a degree program that may not be easy to detect manually. Through prediction modeling, forecasts on students' performance can enable stakeholders to proactively take measures to enhance student's performance. Automation of the process also eliminates human biases that can interfere with the process. It presents an opportunity for instructors to identify weaknesses in the programs offered with minimal effort. Through the provision of early warnings, student attrition rates can be reduced and subsequently the quality of education can be improved.

The paper performed explanatory modeling to reveal some of the factors that influence academic performance. The suggested strategies are based on anecdotal evidence about the actual effectiveness and efficiency of the suggested solution. They, however, form a basis for further research to investigate the effectiveness of the suggested approaches.

5 Limitations of the Study

This study investigated the academic performance of students taking computer-related degree programs at the university. The general objective of the study was finding out whether factors influencing student academic performance can be modeled to provide explanations on students' performance with a view of enhancing their performance. Although the study was conducted in a single private university, the results have the potential to be generalized for other institutions. Further research in this area should include a comparative study among several institutions to provide recommendations for enhancing the approach.

Traditionally many universities emphasize that mathematics and physics skills are required for admission into computer-related degree programs. Findings from this research confirmed that background skills in mathematics were a key factor in determining a student's performance. The results provide a basis for investigating students' performance. Further research is required to describe the correlations with a view of establishing whether they are pointers to actual correlations or simply mere casualties. Further, the study did not investigate the relationship between the student's background knowledge in physics and their academic performance. This is because physics skills were not assessed by the university and no records of the students' secondary scores for physics were captured in the university's Student Information System.

Student's academic performance is influenced by several factors originating from their social, economic, and academic backgrounds. This research, however, only focused on academic factors arising from a student's educational background as well as their current academic performance in courses taken at the university. These were factors that could easily be collected from the Student's Information Systems. Additionally, they are factors that the institution can address to enhance the student's academic performance.

Findings from the study revealed the need to improve on the teaching approaches for lower level courses as they have an impact on the student's cumulative GPA later in the program. However, the study did not investigate how to do this. Further research is required to explore how different teaching approaches influence the cumulative GPA.

6 Conclusions and Future Research Directions

A review of existing literature revealed that few similar studies had been done. Most of the research focused on enhancing the performance of specific courses in computing programs rather than all courses across the curriculum. Several studies conducted used students' perceptions which is important but might not always give the actual picture. This study set out to address this gap by analyzing performance across all courses in a program with an emphasis on factors that the institution can influence. Further, it made use of actual student scores to evaluate factors that influence performance. Data analytics techniques were used to classify data and provide explanations. The results obtained showed that a student's mathematics background and their performance in their introductory computing course were key in determining performance in their computer studies. Unexpectedly, prior computer skills or secondary school grades had less influence on a student's performance. The non-academic factors measured such as country of origin and gender were also not significant in determining a student's academic performance. The research also suggested strategies for enhancing student performance in computer-related degree programs.

Further research in this area should be conducted to validate the effectiveness of the suggested strategies. More extensive research should also be conducted among both private and public universities in the region as well as among other degree programs. Other factors that influence a student's academic performance such as their social, cultural, and economic backgrounds would also enhance the findings of research in this area. Investigating the influence of other academic factors such as student's background in physics would also enhance the study. In this study, explanatory modeling laid the foundation for our current research in predictive modeling. The goal was to show that explanatory modeling can be used to identify factors that influence performance hence setting the case for predictive modeling of academic performance. The use of prediction models such as Regression Analysis and Neural Networks can provide a proactive approach to the problem. While this research is not exhaustive it presents an opportunity for researchers in computer studies education to find ways of enhancing student's academic performance through explanatory and predictive modeling techniques.

References

1. Odhiambo, G. O.: Higher education quality in Kenya: A critical reflection of key challenges. Quality in Higher Education, 17(3), 299-315 (2011).
2. Giannakos, M. N., Pappas, I. O., Jaccheri, L., & Sampson, D. G.: Understanding Student Retention in Computer Science Education: The role of environment, gains, barriers, and usefulness. Education and Information Technologies, 22(5), 2365-2382 (2017).
3. Njoroge MM, Wang'eri T, Gichure C.: Examination Repeats, Semester Deferments and Dropping Out as Contributors of Attrition Rates in Private Universities in Nairobi County Kenya. International Journal of Education and Research, 4(3) (2016).
4. Gárcia-Mateos G, Fernández-Alemán J. L.: A course on algorithms and data structures using on-line judging. In ACM SIGCSE Bulletin, Vol. 41, No. 3, 45-49. ACM (2009).

5. Kumar AN.: Need to consider variations within demographic groups when evaluating educational interventions. In ACM SIGCSE Bulletin Vol. 41, No. 3, pp. 176-180. ACM (2009).
6. AlMurtadha YM, Alhawiti KM, Elfaki AO, Abdalla OA.: Factors Influencing Academic Achievement of Undergraduate Computing Students. International Journal of Computer Applications, 975:8887 (2016).
7. Nash J.: Computer skills of first-year students at a South African university. In Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association 2009 Jun 29, pp. 88-92. ACM (2009).
8. Garcia, R.A., & Al-Safadi, L.A.: Comprehensive Assessment on Factors Affecting Students' Performance in Basic Computer Programming Course towards the Improvement of Teaching Techniques. International Journal for Infonomics. 6. 682-691. 10.20533/iji.1742.4712.2013.0079 (2013).
9. Wang, X. M., Hwang, G. J., Liang, Z. Y., & Wang, H. Y.: Enhancing students' computer programming performances, critical thinking awareness and attitudes towards programming: An online peer-assessment attempt. Journal of Educational Technology & Society, 20(4), pp. 58-68 (2017).
10. Pretorius H. W, Hattingh M. J.: Factors influencing poor performance in systems analysis and design: student reflections. In Annual Conference of the Southern African Computer Lecturers' Association, pp. 251-264. Springer, Cham (2017).
11. Barlow-Jones G, van der Westhuizen D.: Pre-entry Attributes Thought to Influence the Performance of Students in Computer Programming. In Annual Conference of the Southern African Computer Lecturers' Association, Springer, Cham, pp. 217-226 (2017).
12. Chikumba PA.: Student Performance in Computer Studies in Secondary Schools in Malawi. International Conference on e-Infrastructure and e-Services for Developing Countries, Nov 25 pp. 113-121. Springer, Berlin, Heidelberg (2010).
13. Akinola, O. S., & Nosiru, K. A.: Factors influencing students' performance in computer programming: a fuzzy set operations approach. International Journal of Advances in Engineering & Technology, 7(4), pp. 1141-1149 (2014).
14. Wilson BC.: A study of factors promoting success in computer science including gender differences. Computer Science Education. 12(1-2): pp. 141-64 (2002).
15. Rauchas S, Rosman B, Konidaris G, Sanders I. Language performance at high school and success in first year computer science. In ACM SIGCSE Bulletin Vol. 38, No. 1, pp. 398-402. ACM (2006).
16. Namuye, S., Platz, M., Okanda, P., & Mutanu, L.: Leveraging health through early warning systems using mobile and service-oriented technology. In 2015 IST-Africa Conference, pp. 1-10, IEEE (2015).
17. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH.: The WEKA data mining software: an update. ACM SIGKDD Explorations newsletter. Nov 16;11(1): pp. 10-8 (2009).
18. Team, R. Core.: R: A language and environment for statistical computing. R Development Core Team, Version 2.6.2, 201 (2013).
19. Kremelberg, D.: Practical statistics: A quick and easy guide to IBM® SPSS® Statistics, STATA, and other statistical software. SAGE publications, (2010).
20. Kim H, Song HY.: Formulating human mobility model in a form of continuous time Markov chain. Procedia Computer Science. pp. 389-96 (2012).
21. MacQueen J.: Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability Vol. 1, No. 14, pp. 281-297 (1967).
22. Kim T. K.: T-test as a parametric statistic. Korean journal of anesthesiology. 68(6):540 (2015).
23. Sedgwick P.: Pearson's correlation coefficient. BMJ. 345:e4483, (2012).