Applying Machine Learning to Improve Curriculum Design

Robert Ball, Linda Duhadway, Kyle Feuz, Joshua Jensen, Brian Rague and Drew Weidman
School of Computing, Weber State University, Ogden, Utah USA
{robertball, lindaduhadway, kylefeuz, joshuajensen1, brague, dweidman}@weber.edu

ABSTRACT

Creating curriculum with an ever-changing student body is difficult. Faculty members in a given department will have different perspectives on the composition and academic needs of the student body based on their personal instructional experiences. We present an approach to curriculum development that is designed to be objective by performing a comprehensive analysis of the preparation of declared majors in Computer Science (CS) BS programs at two universities. Our strategy for improving curriculum is twofold. First, we analyze the characteristics and academic needs of the student body by using a statistical, machine learning approach, which involves examining institutional data and understanding what factors specifically affect graduation. Second, we use the results of the analysis as the basis for applying necessary changes to the curriculum in order to maximize graduation rates. To validate our approach, we analyzed two four-year open enrollment universities, which share many trends that help or hinder students' progress toward graduating. Finally, we describe proposed changes to both curriculum and faculty mindsets that are a result of our findings. Although the specifics of this study are applied only to CS majors, we believe that the methods outlined in this paper can be applied to any curriculum regardless of the major.

CCS CONCEPTS

• Computer Science Education • Computational Science and Engineering Education

KEYWORDS: curriculum, machine learning, objective reasoning

ACM Reference format:
Robert Ball, Linda Duhadway, Kyle Feuz, Joshua Jensen, Brian Rague and Drew Weidman. 2019. Applying Machine Learning to Improve Curriculum Design. In *Proceedings of SIGCSE '19: The 50th ACM Technical Symp. on Computing Science Education Proceedings (SIGCSE'19).* ACM, Minneapolis, MN, USA. ACM. NY, NY, 7 pages. https://doi.org/10.1145/3287324.3287430

1. Introduction

When discussing curriculum design, faculty may be inordinately influenced by personal experience and anecdotal evidence.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
SIGCSE '19, February 27-March 2, 2019, Minneapolis, MN, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-5890-3/19/02…$15.00
https://doi.org/10.1145/3287324.3287430

The specific academic scenarios of a few students may drive curriculum discussion in a direction in which statistics and facts about actual student performance are dismissed or ignored. Under these conditions, disagreements among faculty about which curriculum best serves the students are likely to emerge. What courses should be taught? How should these courses be sequenced? Beyond the basic requirements for ABET accreditation, what should be done to help students graduate?

This paper presents a more objective approach to refining curriculum in order to meet student needs. We recognize there are both similarities and differences among institutions, computer science (CS) programs, and the students enrolled in those programs. For the purpose of this paper we define curriculum to be the courses offered in a department, the content taught in these courses, the sequence of courses a student takes to graduate, and the instructional modality (e.g. online, traditional lecture, hybrid, etc.).

Our approach to improving curriculum is twofold. First, we analyze the characteristics and academic needs of the student body by using an objective, statistical, machine learning approach, which involves examining institutional data and understanding what factors specifically affect graduation. Second, we use the results of the analysis as the basis for applying necessary changes to the curriculum in order to maximize graduation rates.

For this analysis we specifically investigated only academic, age, and gender factors while purposely leaving out financial, marital, race, and social factors. While there are many important socioeconomic and personal factors that affect graduation (see Related Works), for this study we address only those academic factors that were readily and reliably accessible across all students. Many social factors such as pregnancy, marital status, etc. impact only some members of the student community during their academic careers while certain academic factors, such as a student's level of mathematical competence, can be consistently monitored over time for all students.

In this study we examined every course undergraduate CS majors attended during a twelve-year interval at two institutions with similar Carnegie classification profiles (Master's University; very high undergraduate; open enrollment). We analyzed our own institution's student transcript data then proceeded to identify a similar institution to determine if the trends we discovered were unique to our particular program. We found similar trends at both institutions. A total of 4,266 anonymized individual transcripts were analyzed for graduation factors from the two institutions. Transcripts were acquired only for those students who declared a CS major between 2005 – 2017 and enrolled in at least one CS course during that time period. Student course grades were not a transcript selection criterion.

The overarching question driving this study is the following: *What curriculum changes can faculty implement to help our students graduate?*

Our answer is to first understand the factors that influence student success, at which point we can then create the courses needed to help them succeed. For example, if most students struggle with math then what can faculty do to mitigate that hurdle? Our approach was to first systematically discover that such a problem exists before trying to fix it. As part of the context of this study, we also recognized that students have multiple paths available to earn the same degree and come from many academic backgrounds and as such there is no one particular path of courses that exactly maps to all students. Although the specifics of this study are applied only to CS majors, we believe that the methods outlined in this paper may be applied to any curriculum regardless of the major.

In this paper we present related work, describe statistical trends that may or may not lead to graduation, show important academic weaknesses by leveraging Machine Learning (ML) algorithms, and explain how there is no effective "one size fits all" curriculum. Finally, we detail our curriculum plans resulting from this analysis and relate these findings to the work of other researchers.

2. Related Work

The subject of institutional retention of students has been heavily studied. Many of these studies have focused primarily on the retention rate of the entire institution [4, 5, 8, 14]. Considering that many sources of governmental funding for institutions are driven by factors such as retention and student graduation rates, the abundance of studies on the topic is not a surprise.

According to the National Center for Education Statistics, open enrollment institutions have a retention rate of approximately 59% and among those who matriculate at these institutions only about 32% end up graduating within 6-years of their first institutional attendance (NCES, 2018). Studies have shown that there are many factors that may contribute to the remaining 68% of students who do not complete their degree within the 6-year timeframe, such as student-faculty relationships, institutional support services (e.g. student counseling and advisement), students developing a sense of belonging to the institution, and pre-college preparation.

Quantitative factors that affect graduation include cumulative GPA, first-semester GPA, race, and income [2, 5, 6, 11, 15, 16]. While the aforementioned studies focused on students starting and finishing at a single institution, Jones-White conducted a study of students at the University of Minnesota-Twin Cities utilizing data from the National Student Clearinghouse that considered all the institutions a student engaged with while earning a degree. They found that the factors previously discovered generally held true, even if a student transferred to different institutions to complete a degree [7].

Taken as a whole, these previous studies demonstrate the complexity of the problem facing higher education when addressing graduation rates. An information-driven approach to this issue will increase the likelihood of success. One such study was conducted at Harvey Mudd College to determine strategies to increase the recruitment and success of female CS students [1]. This study carefully considered and applied the qualitative factors previously described, encouraging social interaction among female students even before they arrived on campus for their first semester. Examples included offering students the opportunity to attend the Grace Hopper Celebration of Women in Computing and grouping students into CS1 course sections based on prior CS experience. The results of those efforts continue to validate the factors found in the studies discussed.

3. General Approach

In trying to understand student performance trends and likelihood of graduation we used two main approaches. The first approach is top-down involving creativity, insight, data visualization, and statistical inference. We classified the students into different academic categories. For example, we found that students who had taken dual or concurrent enrollment (CE) courses in high school (courses that counted towards both high school and college credit) had higher GPA's than most students that did not take CE courses. However, these same CE students often transferred to more selective institutions.

The second approach is bottom-up, identifying factors about each student from their transcripts (e.g. age, gender, concurrent enrollment credit) and entering these data into Machine Learning (ML) algorithms. We employed ML algorithms for two purposes: (1) to determine what factors were the most important in predicting graduation; and (2) to verify ML algorithm accuracy. Verification of the ML algorithms involved comparison with a baseline majority class algorithm and three-fold cross validation to determine how well the ML algorithms understood the student population and trends.

4. Statistics (Top-Down) Approach

Our first approach of evaluating the student populations from the two institutions is called Exploratory Data Analysis. The general approach is to use statistics, visualization, creativity, and intuition in order to gain insight into the data. Although each institution is different and thus might reflect unique student performance trends when compared with other institutions, the conclusions from studies such as Jones-White indicate that we should in fact expect very similar trends among open enrollment institutions. Consequently, although we began our analysis with only data from our own institution (University *Anonymous1*), interest in comparing data across institutions prompted the subsequent acquisition of anonymized transcripts from University *Anonymous2* for the same twelve-year period of time. (Internal Review Board (IRB) approval from both institutions was secured, and student identities were kept confidential.)

Although we found many unique and interesting trends that affect only our respective institutions, the most important trends that likely generalize to other institutions can be found in Table 1. Dual or concurrent enrollment (CE) courses are courses that students took in high school that also count for University credit. Advanced Placement (AP) is similar to CE but is based on taking a third-party exam to determine qualification for university credit. Math course category counts indicate the *first* math course

students enrolled in at any institution of higher education and thus takes into account math courses transferred from other institutions. We define *Developmental Math* as any math course before college algebra. *Introductory Math* is any math course that includes college algebra up to any other math course before the calculus level. *Advanced Math* is calculus level and above. For instance, if a student is in the *Advanced Math* category then the first math course taken by the student at any university is calculus or higher. On the other hand, if the student is in the *Developmental Math* category then their first math course at the university was a math course at a level lower than College Algebra.

Transfer Credit means that the student took a course at another institution and transferred it to their current institution for credit. For example, CS 101 was taken elsewhere, but articulates and transfers with credit to our institution. *Transfer Credit w/o CS* means that the student transferred credit, but no CS courses. *CS Transfer Credit* means that at least one CS course was transferred. *Traditional Student* classifies a student who (a) is less than 20-years-old, (b) did not enroll in any concurrent courses, (c) did not receive AP credit, and (d) did not transfer any credits from another institution.

Table 1. Results from both institutions (number of students).

	total	GPA	average age	% female	% graduated in CS
All students:	4266	2.86	26.37	8.27%	22%
Concurrent credit	991	2.94	22.2	8.68%	22%
Advanced Placement (AP)	387	3.10	23.11	8.79%	35%
Math					
Developmental	656	2.29	26.98	9.76%	10%
Introductory	1988	2.71	24.82	7.65%	27%
Advanced	1023	3.09	26.92	6.55%	55%
Age					
< 20 years old	676	2.42	18.28	10.36%	15%
20 - 24 years old	1403	2.71	22.06	6.77%	29%
>= 25 years old	1918	2.88	32.57	8.97%	31%
Developmental math	656	2.29	26.98	9.76%	10%
Non-Developmental math	3610	2.92	26.26	8.01%	28%
Developmental English	300	2.20	27.32	8.33%	14%
Non-Developmental English	3966	2.82	26.3	8.27%	33%
Transfer Credit					
Transfer credit w/o CS	2336	2.85	27.49	8.69%	25%
CS transfer credit	998	2.91	28.93	6.71%	43%
No transfer credit	2336	2.55	27.49	8.69%	21%
Traditional Students:	148	2.28	18.31	5.41%	11%

Table 1 reveals the general trends for the different groups. We specifically show how GPA, age, gender, and graduation rate vary among the different groups. For example, the *Advanced Math* group showed the highest graduation rate (40.86%) followed by the *CS Transfer Credit* group (36.17%). In contrast, *Developmental Math* students were the least likely to graduate (9.3%). Although the graduation rate of the *Advanced Placement* group (30.49%) is relatively high, it is interesting to note this group of students were more likely to leave the open enrollment institutions examined in this study to go to more selective institutions.

Performing a one-way ANOVA on the GPA's of the different groups, we obtain statistical significance with f=123.73 p<0.01. Figure 1 shows the various GPA's of the different groups. The vertical bars show the standard error of each GPA. If the standard error bars cross then there is not statistical significance between them, otherwise there is statistical significance. For example, the *Advanced Math* and *Advanced Placement* groups overlap showing that there is not statistical significance between their GPA's, but they both have statistical significantly higher GPA's than all the other groups.

Performing a one-way ANOVA on the graduation rates of the different groups, we get the following: f=105.03 p<0.01. Figure 2 shows the various graduation rates of the different groups. Similar to Figure 1, the vertical bars show the standard error of each group.

Before comparing Figure 1 to Figure 2, the reader might expect that a high GPA would necessarily imply a high graduation rate. However, upon careful inspection of these figures, one can see that the trend of higher GPA does not always correlate with higher graduation rates in Computer Science for a given student category.

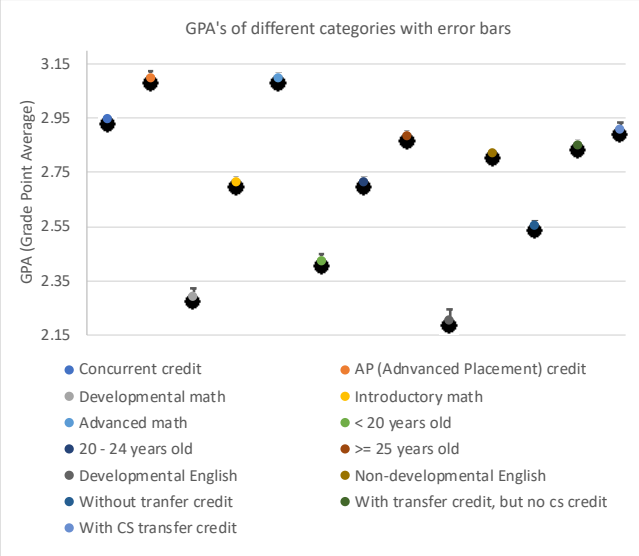

Figure 1. The average GPA's of the groups from Table 1. The legend reads from left to right. Non-overlapping standard error bars signify statistical significance between groups.

For example, the GPA of *Advanced Placement* students and students that started in *Advanced Math* is similar. However, the AP students graduate far less than the *Advanced Math* group. Why? It appears that *Advanced Placement* students and *Concurrent Credit* students may view the open enrollment institutions examined in this study as a launching pad to transfer to other, more selective universities. It is highly likely that many of these students eventually graduate, but not with their original institutions.

Another example, in this instance involving transfer credit, shows that the *CS Transfer Credit* group in Figure 1 has a slightly higher GPA compared to the *Transfer Credit w/o CS* group, but only slightly (2.9 to 2.8 respectively). However, for graduation rates, the *CS Transfer Credit* group has approximately double the graduation rate (24.6% to 11.5% respectively). In this example, it appears prior exposure to CS increases the chances of earning the degree. There are admittedly many different ways to think about and categorize students based on the data. However, the best categories are clearly those that identify features and characteristics that support and improve student graduation rates.

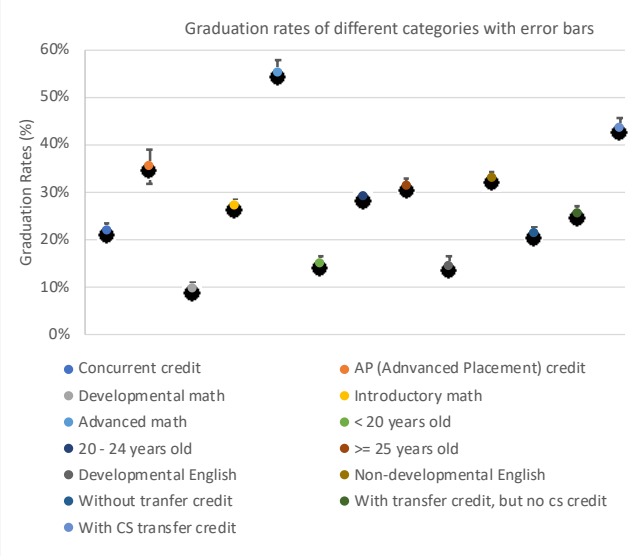

Figure 2. The average graduation rates of the groups from Table 1. The legend reads from left to right. Non-overlapping standard error bars signify statistical significance between groups.

5. Machine Learning (Bottom-Up) Approach

In contrast to the statistical approach, in which the researcher is responsible for identifying meaningful categories to perform the analysis on the student data, our machine learning approach lets the data speak for itself. In supervised machine learning the goal of the learning algorithm is to learn a mapping from a feature space X (e.g. age, gender, etc.) to a label space Y (e.g. graduation rate) [10]. In our scenario the features are extracted from the student transcripts and the label space is the binary output of whether the student graduated from the institution with a BS degree in CS.

From the student transcripts we extract the following features:

- Age (when taking their first CS program class at a university)
- Gender
- First English course
- First Math course
- International status
- Advanced Placement credit
- Transfer credit
- CS Transfer credit
- Concurrent credit
- GPA (either overall or at a particular semester)

All of above features are binary values except for Age, First English course, First Math course, and GPA. We use a one-hot encoding to convert first English course into two binary features (Developmental English and Introductory English). Similarly, First Math course is converted to three binary features (Developmental Math, Introductory Math, Advanced Math). Age and GPA are both scaled to the range [0,1] using a non-linear quantile transformation such that each feature's probability distribution function is mapped to a uniform distribution. This technique is robust to outliers so that the scale remains consistent with and without the outliers present.

The first objective of our machine learning approach is to quantify the effect each individual feature has on graduation. To do this we chose learning algorithms that allow this influence to be assessed. Decision trees are one good choice because each node in the tree uses exactly one feature to determine which branch to take next. This leads to the Gini importance factor [3] which is a measure of the importance of each feature in the overall tree.

Logistic regression is another algorithm that allows direct measuring of the effect that each feature has on the overall decision of the learning algorithm. In logistic regression, a coefficient is learned for each feature which determines how much the given feature affects the predicted value. The magnitude of the value of the coefficient is an indication of how important the feature is in determining the class label. The sign of the coefficient is an indication of whether the feature leads to the positive or negative class (graduate vs non-graduate). This coefficient is equivalent to the log-odds ratio [13].

We are specifically interested in determining which features best lead to graduation and which do not. In addition, we look at how those features change as the student progresses through the current curriculum. Do certain factors become more or less important over time in predicting graduation? Does the accuracy of the prediction improve the further a student progresses in the program?

To accomplish this objective, we look at the graduation outcomes of students enrolled in four specific courses in the program typically taken in the following sequence: CS 1, CS 2, Data Structures and Algorithms, and Operating Systems. All the extracted features remain the same during this analysis except for overall GPA. Overall GPA is replaced by the semester GPA at the time the specified course was completed.

Table 3 shows which features are influential on the Decision Tree (GR) and Logistic regression (OR) models. The relative importance of each feature at the point of each course are shown with the smaller number being more important and the higher numbers being less important. The omission of data for the Operating System course is addressed later in Section 5. Unremarkably, GPA is the most important feature for all courses across both models. In other words, the higher the GPA of the student, the more likely they graduated.

Table 3. Ranking of feature importance at different points in a student's curriculum progression. GR is a ranking based on the Gini Importance factor extracted from the decision tree model. OR is a ranking based on the log-odds ratio extracted from the Logistic Regression model. * indicates the feature is correlated with the negative class (non-graduate) in the Logistic Regression model. 'DSA' = 'Data Structures and Algorithms.'

	Overall		CS 1		CS 2		CS DSA	
Feature	GR	OR	GR	OR	GR	OR	GR	OR
Age	3	7	4	7	4	13	4	9
Gender	7	12	8	11*	8	8	8	10
Dev_Engl	7	4	8	13	8	9	10	4
Intro_Engl	4	2	8	5	8	5	5	5
Dev_Math	7	9*	7	8	8	7	10	11*
Intro_Math	6	11	3	3	3	3	9	8
Adv_Math	2	3	2	2	2	2	2	2
Intnl	7	8	8	4	8	11	7	7*
AP	7	6*	5	6*	5	4*	6	3*
Transfer (any credit)	7	13	8	12	7	12	10	13*
CS Transfer	5	5	8	10*	8	10*	10	11
Concurrent	7	10*	6	9	6	6*	3	6*
GPA	1	1	1	1	1	1	1	1

There are several similar outcomes from the two models. Both ML algorithms rank GPA and Advanced Math high (with Intro English being high for overall courses) while ranking gender, international status, and non-CS transfer credit low. Since GPA as well as English and Math class enrollments are gender-neutral measures, these rankings indicate that when examining measures of academic preparation alone (social factors notwithstanding), females are as likely to succeed in CS programs as their male counterparts, which validates the work by Alvarado and Dodds [1].

In table 3, the rank listed for Age indicates relatively high significance for the Gini importance factor (GR) when compared to the log-odds ratio (OR). This is likely because the effect of age is non-linear: as age increases graduation rates increase up to a certain point after which the graduation rate again decreases. The decision tree can account for this by splitting age at different points. However, Logistic Regression is forced to treat age linearly which leads to age having less influence on the predicted graduation.

The log-odds ratio rankings listed in table 3 indicate that AP and concurrent credit are both the only consistently negative predictors of graduation. This is consistent with what we deduced in Section 4 where students start at one institution in question but soon transfer to other (usually more selective) institutions to finish their degree.

The second objective of the machine learning approach is to predict which students will graduate from the institution with a BS degree in Computer Science. This objective encompasses two sub-goals: (1) to validate that the results in Table 3 are better than a baseline guess and (2) to further understand how the ML algorithms can be used for understanding the student population as they progress through the program.

Table 4. Classification Accuracy at different points in a student's curriculum progression. Each number is a percentage of accuracy. 100 would indicate perfect prediction and 0 would indicate no success at all in prediction. The bold numbers show the highest prediction rate per course.

	DT	LR	Ada	RF	MC
All	71.29	70.53	**72.69**	68.75	62.07
CS1	66.45	65.83	**66.72**	61.61	63.15
CS2	61.38	**63.87**	63.31	58.55	56.19
DSA	63.20	**64.56**	64.15	57.00	56.59
OS	75.11	76.65	**76.93**	68.08	75.46

We compare four different machine learning (ML) algorithms: Decision Tree (DT), Logistic Regression (LR), AdaBoost DT ensemble (Ada), and Random Forest (RF) [12]. We also compare against the baseline accuracy of choosing the majority class (MC). The MC algorithm simply chooses what most students do. For example, if 75% of the students do not graduate then the MC algorithm chooses 'not graduate' for every student and is, therefore, correct 75% of the time. If any given ML algorithm does not perform significantly better than the baseline MC algorithm then we conclude that the ML algorithm in that case is not effective. In addition, we use three-fold cross validation where the data is randomly split into three chunks - two chunks are used for training and the third chunk is used for testing. This training and testing process is applied a total of three times - once for each of the three chunks - and then the results are averaged. In other words, we made sure that we were predicting on data randomly ordered so that we did not bias our results.

Table 4 shows the classification accuracy for the different ML algorithms for each course compared to the baseline majority class (MC) algorithm. For example, for all courses in the dataset (first row in Table 4) the AdaBoost DT ensemble (Ada) algorithm had the best accuracy (72.69%), a 10.62% improvement over the MC baseline algorithm. There is a general improvement in predicting graduation rates from CS 1 to Data Structures and Algorithms (DSA) when compared to the MC baseline. For CS 1 there is a 3.57% improvement over the MC baseline, for CS 2 there is a 7.68% improvement, and for DSA there is a 7.97% improvement.

The astute reader may have noticed that data for the Operating System (OS) course is not shown in Table 3. It is because there is a noticeable lack of improvement between any of the classification algorithms and choosing the MC baseline for the OS course with only a 1.47% improvement. In other words, the ML algorithms are not significantly better than the baseline algorithms at predicting graduation for students in the OS course. As a result, no feature rankings were shown for the OS course in Table 3 because they would not have been valid.

These results indicate that the features extracted from the student transcript help predict which students will successfully complete the program early in a student's curriculum progression but may be less helpful by the time a student is taking upper-division courses. We suspect that if a student reaches the upper-division courses life events and circumstances not captured by a student transcript have a more significant effect on graduation than those features captured on the student's transcripts.

6. From Study to Practice

Unsurprisingly, the biggest issue regarding student graduation success appears to be academic preparation. However, the results from this study suggest a double-edged sword: if the student is underprepared, then they are less likely to graduate, but if they are over-prepared then they are more likely to transfer to a more selective institution. What makes a student 'unprepared'? We found that starting in developmental Math and English courses has a negative effect on a student's prospects for graduation. Unprepared students also have a lower GPA, which is a secondary measure of their success rate at a university. The unprepared students start off with lower GPA's than their prepared counterparts, potentially activating a downward spiral. These students begin with developmental Math and/or developmental English and end up not doing well in those courses. They may then proceed to CS1 and obtain marginally passing grades. If they continue with the program then their GPA is biased low, which becomes a fundamental indicator of their lack of success.

On the other hand, there are two main types of 'prepared' students: those students with higher GPA's that begin the program in Advanced Math and those students who earn high school credit (concurrent and AP students). The first group does well and demonstrates the highest graduation rates while the student with high school credit often transfers to other institutions. Identifying these student academic groupings is just the first step. Once the factors have been identified that contribute to or detract from graduation success we have the difficult task of effectively managing and modifying our curriculum to match our student's needs. Advocating general university policies, such as more choices for day care for older students with young children, is beyond the scope of this paper. However, there are many things that institutions can do to help different social groups succeed.

The study at Harvey Mudd College cited in Section 2 shows that changing curriculum to increase the percentage of women in CS1 can be effective [1]. Based on our analysis, we now plan to offer specific sections of CS1 to address the needs of beginning CS students who have had little or no background in programming or software design and will most likely have attended classes in developmental Math and/or developmental English. Another potential change involves offering math courses oriented to CS majors. For example, many universities across the US offer Business Calculus or Engineering Calculus. These courses are specifically designed to meet the needs of those majors, typically eschewing the proof-heavy traditional mathematics approach. Our analysis also indicates that many of our students who take Developmental Math fail to graduate in CS. Developmental Math courses designed specifically for CS majors may help these students acquire the requisite mathematical skillsets that will increase the likelihood of success in subsequent CS classes. On the other end of the student preparation spectrum, we also plan to offer CS honors-track courses targeting those students that have Advanced Placement (AP) or concurrent credit. The course content and activities in these honors-track classes will offer an academic challenge to these students in order to increase retention and reduce the number of transfers to other institutions.

To support student advising and manage the various pathways available to students as they navigate the CS program, we plan to introduce a survey website that will direct students toward the appropriate sections/courses based on their prior classes and past academic experiences. Although having many different sections of introductory courses customized to student needs may appear daunting, the results from Section 5 show that the influence of academic preparation on graduation diminishes as students progress to upper division courses. In other words, the beginning courses help all students reach a minimal level of competence that allows them to succeed in later courses. Thus, our results suggest that having multiple sections or sequences of courses later in the curriculum is not necessary to ensure student success. Lastly, since age is such a strong factor as indicated by one of the ML algorithms, what can a department do to support the scholastic needs of both older and younger students, and how do these needs differ? We have experienced success with night courses, similar to courses offered during the day, but allowing working adults the option and convenience to continue their education.

7. Conclusion

Anecdotal, emotional accounts of the experiences of a few students should not be the driving factor in curriculum development. We advocate an objective, comprehensive study of the overall student body. By looking at the data highlighting the factors that contribute to or detract from graduation success, a department can provide their students the individual attention that they need. Using statistics and machine learning algorithms, decisions can be made about curriculum modifications that will improve student success. Since starting this study, faculty discussions regarding curriculum have been less emotionally-charged and faculty as a whole have applied their logical minds towards solving curriculum issues, generating a more positive feeling about program design.

This paper centers on two important components of curriculum development. First, we have discovered a number of trends related to student graduation rates that should hold across open enrollment institutions nationwide. Such trends show that students whose first courses in Math and/or English are developmental experience a detrimental effect on their prospects for graduation, but that students who have obtained high school credit, such as concurrent credit or AP credit, often find the introductory courses not sufficiently challenging and transfer to more selective institutions.

We have identified many different characteristics about students and how these features impact their graduation. Understanding an individual student's academic background and guiding that student to an appropriate set of courses (e.g. developmental courses for CS majors, calculus for engineers, introductory computer science courses, etc.) tailored to their individual needs is a much more effective strategy when compared to a one-size-fits all approach.

Second, the related works section clearly illustrates that predicting graduation is a multi-dimensional problem that is difficult to quantify. Social issues, race, gender, financial aid, etc. are all factors to consider when trying to understand graduation rates.

However, by objectively looking at the data for a particular institution, one can use statistics and Machine Learning (ML) algorithms to determine the factors in a student's academic pathway that significantly influence successful degree completion.

8. REFERENCES

[1] Alvarado, C., & Dodds, Z. (2010). Women in CS: an evaluation of three promising practices. Proceedings of the 41st ACM tech. symposium on CS education, (pp. 57-61).

[2] Attewell, P., Heil, S., & Reisel, L. (2011). Competing explanations of undergraduate noncompletion. American Educational Research Journal, 48, 536-559.

[3] Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

[4] Crosling, G., Heagney, M., Thomas, L., & others. (2009). Improving student retention in higher education: Improving teaching and learning. Australian Universities' Review, 51, 9.

[5] Delen, D. (2010). A comparative analysis of machine learning techniques for student retention management. Decision Support Systems, 49, 498-506.

[6] Ishitani, T. T. (2006). Studying attrition and degree completion behavior among first-generation college students in the United States. The Journal of Higher Education, 77, 861-885.

[7] Jones-White, D. R., Radcliffe, P. M., Huesman, R. L., & Kellogg, J. P. (2010). Redefining student success: Applying different multinomial regression techniques for the study of student graduation across institutions of higher education. Research in Higher Education, 51, 154-174.

[8] Lykourentzou, I., Giannoukos, I., Nikolopoulos, V., Mpardis, G., & Loumos, V. (2009). Dropout prediction in e-learning courses through the combination of machine learning techniques. Computers & Education, 53, 950-965.

[9] McFarland, J., Hussar, B., Brey, C., Snyder, T., Wang, X., Wilkinson-Flicker, S., . . . others. (2017). The Condition of Education 2017. NCES 2017-144.

[10] Mitchell, T. M. (1997). Machine Learning (1 ed.). New York, NY, USA: McGraw-Hill, Inc.

[11] O'Keeffe, P. (2013). A sense of belonging: Improving student retention. College Student Journal, 47, 605-613.

[12] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[13] Peng, C.-Y. J., Lee, K. L., & Ingersoll, G. M. (2002). An Introduction to Logistic Regression Analysis and Reporting. The Journal of Educational Research, 96, 3-14.

[14] Porter, K. B. (2008). Current trends in student retention: A literature review. Teaching and Learning in Nursing, 3, 3-5.

[15] Rogulkin, D. (2011). Predicting 6-Year Graduation and High-Achieving and At-Risk Students. Online Submission.

[16] Warburton, E. C., Bugarin, R., & Nunez, A.-M. (2011). Bridging the Gap: Academic Preparation and Postsecondary Success of First-Generation Students. Statistical Analysis Report. Postsecondary Education Descriptive Analysis Reports.