JOURNAL OF LEARNING ANALYTICS

Volume 7(2), 84-94. http://dx.doi.org/10.18608/jla.2020.72.5

Scaling the Student Journey from Course-Level Information to Program Level Progression and Graduation: A Model

Pablo Munguia[1], Amelia Brennan[2]

Abstract

No course exists in isolation, so examining student progression through courses within a broader program context is an important step in integrating course-level and program-level analytics. Integration in this manner allows us to see the impact of course-level changes to the program, as well as identify points in the program structure where course interventions are most important. Here we highlight the significance of program-level learning analytics, where the relationships between courses become clear, and the impact of early-stage courses on program outcomes such as graduation or drop-out can be understood. We present a matrix model of student progression through a program as a tool to gain valuable insight into program continuity and design. We demonstrate its use in a real program and examine the impact upon progression and graduation rate if course-level changes were made early on. We also extend the model to more complex scenarios such as multiple program pathways and simultaneous courses. Importantly, this model also allows for integration with course-level models of student performance.

Keywords
Program analytics, matrix model, program pathways, learning journey

Submitted: 18.07.2019 — **Accepted:** 23.06.2020 — **Published:** 19.09.2020

Corresponding author[1] Email: pablo.munguia@flinders.edu.au Address: Flinders University, Bedford Park, SA, 5042, Australia
[2] Email: mia.brennan87@gmail.com Address: Department of Education, Hobart, TAS, 7000, Australia

1. Introduction

Learning analytics is gaining momentum in education as researchers and instructors learn how to use student-generated data to inform class activity and provide feedback (e.g., Chiappe & Rodríguez, 2017). Much of the research performed in this field has focused on course-level outcomes where students are directly engaging with their teachers, such as using student online engagement behaviours to predict their academic outcomes (Daud et al., 2017; Macfadyen & Dawson, 2012; Stretch, Cruz, Soares, Mendes-Moreira, & Abreu, 2015), or understanding the impact of teaching practices and engagement (Bangert, 2008; Garrison & Cleveland-Innes, 2005; Shea, Li, & Pickett, 2006).

However, courses do not exist in isolation, and the connectivity and progression between courses is a challenging component of learning analytics (Raji, Duggan, DeCotes, Huang, & Vander Zanden, 2017). At the within-course scale, learning analytics can identify engagement as well as immediate actions that an instructor can undertake to improve student outcomes within that course. At the between-courses scale (i.e., program level), analytics becomes challenging as programs require continuity, with the underlying assumption that if students pass a course, they have the necessary knowledge required for the courses immediately following within a program. Students' abilities and engagement behaviours will affect their outcomes and change as they complete a course and move on to the next, while other variables such as instructor and course design may only change with variation of teaching staff, adaptations to content, or changes to teaching strategy, which influences the many students passing through. The relationship between course-level variables such as instructor and student engagement, and program-level variables such as program design and retention of skills by students, requires first to understand student progression and the definition of relevant metrics. While the data may be available at institutions, visualizing student flows is challenging (Raji et al., 2017), let alone understanding the drivers behind the observed patterns.

Predictive models of student outcomes have already been extended to the program level by many researchers, with a combination of demographic data and previous or early course results used as inputs (Golding & Donaldson, 2006; Jeffreys, 2007; Kabakchieva, Stefanova, & Kisimov, 2010; Munguia, 2020; Nghe, Janecek, & Haddawy, 2007; Shah & Burke, 1999; Yehuala, 2015). Predictive models have used Markov chain approaches to generate the probability of students finishing programs and consider dimensions such as demographic data and field of study (Nichols, 2007; Shah & Burke, 1999). Alternative approaches to program level analytics have also been developed, such as the definition of a set of metrics based on interaction data to quantify courses and their relationships within a program (Ochoa, 2016). Similarly, curricular analytics seeks to assess the coherence of a program curriculum and relate this to student behaviour (Brennan, Sharma, & Munguia, 2019; Mendez, Ochoa, Chiluiza, & de Wever, 2014; Pechenizkiy, Trcka, De Bra, & Toledo, 2012). Indeed, program-level models such as Markov chain models often focus on the final outcomes (e.g., Nichols, 2007), or demographic data (e.g., Shah & Burke, 1999), they have not incorporated course-level, learning-associated information such as individual assessments, teacher information, or capstone courses.

The progression of students through a program is a relatively new topic within program-level analytics (Raji et al., 2017). It is useful for identifying the typical progression of students (Jeffreys, 2015; Shah & Burke, 1999) and identifying significant courses (Asif, Merceron, Ali, & Haider, 2017). Campagni, Merlini, Sprugnoli, and Verri (2015) developed a measure of distance from the real pathways taken by students through a program to the "ideal" pathway (as defined by an expert in the university) and observed that students who closely followed the ideal tended to perform better; they also pointed out that this model could be used to identify poor program structures. Analytics operating at the program level provides insight into continuity and student progression, which allows for designing programs that best achieve graduate outcomes (Munguia, 2020). Ideally, the two scales of learning analytics (course and program) would be integrated; as Ochoa (2016) points out, "While the focus on course-level analytics could help to improve the learning process in the classroom, only a holistic approach could ensure that these improvements are also reflected in the efficiency and effectiveness of learning programs."

Here we highlight the importance of the program-level scale and the relationships between courses, using a representative model of student progression through a program, with course-level analytics explicitly integrated. We do not attempt to generate complex models that can incorporate all possible student pathways through a university. Our objective is to create a general framework that allows for scalability and the incorporation of course-level activities.

The program-level model uses a stage-based grouping of students within a program, multiplied by a matrix containing course-level measures of students progressing to the next stage of the program, remaining in the same stage, or dropping out of the program altogether. We present a case study of student progression through an online program and demonstrate how the model may be used to predict course populations in the future and identify key points of intervention.

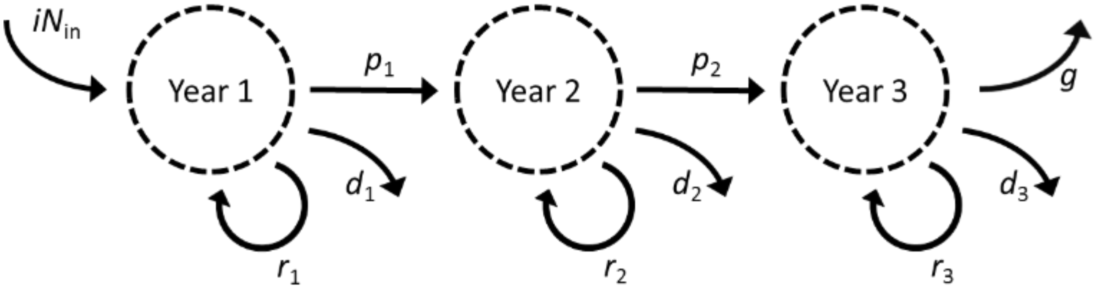

Figure 1. Progression of students through a three-year program. At the end of each year, a student can either progress (or graduate), drop out, or repeat the year, with probabilities p_i (g), d_i and r_i respectively. The incoming cohort is N_{in}.

We also extend the model to accommodate the increasingly complex pathways that students often follow through a degree, including undertaking multiple courses simultaneously and choosing different elective courses. Finally, we outline how this model can be integrated with course-level models of progression and may therefore be treated as a null program-level model from which we can test alternative hypotheses associated with student behaviours, instructor engagement, and program qualities.

2. The Matrix Model

The starting point for our model of student progression represents a three-year program (Figure 1). In this model, at the end of each year, students may either progress to the next year (or graduate from the final year), repeat the year, or drop out of the program. The probabilities of each option are denoted p_i (or g for graduating students), r_i, and d_i, and together sum to 1 for each i. More complex program pathways are considered in section 4.

The matrix population model works as follows. The student population at some time t is written as n_t, where n is a vector containing the numbers of students within each stage. After an appropriate time step,[1] the population is denoted n_{t+1}, and is obtained by left-multiplying n_t by a Lefkovitch matrix **A** (Caswell, 2006; Lefkovitch, 1965):

$$\mathbf{A}\, n_t = n_{t+1} \qquad (1)$$

The population in, for example, ten years, n_{t+10}, is therefore obtained by multiplying by **A** ten times.

The matrix **A** contains all the information about the likelihoods of progressing or repeating each stage; in the first instance, these likelihoods can be considered as measured proportions of students who progress or repeat the course. It has a matrix form where the diagonal elements are the proportions of students who repeat each stage (the $r_i s$), and the sub-diagonal elements are the proportions progressing to the next stage (the $p_i s$). The drop-out proportions (dis) are not explicitly included, since $d_i = 1 - p_i - r_i$.

For the three-year model, it is written explicitly as

$$\begin{bmatrix} a & 0 & 0 & 0 & 0 \\ 1 & r_1 & 0 & 0 & 0 \\ 0 & p_1 & r_2 & 0 & 0 \\ 0 & 0 & p_2 & r_3 & 0 \\ 0 & 0 & 0 & g & 0 \end{bmatrix} \begin{bmatrix} N_{\text{in}} \\ N_1 \\ N_2 \\ N_3 \\ N_g \end{bmatrix}_t = \begin{bmatrix} N_{\text{in}} \\ N_1 \\ N_2 \\ N_3 \\ N_g \end{bmatrix}_{t+1} \qquad (2)$$

N_1, N_2, and N_3 are the numbers of students in years one, two, and three respectively. Ng is the number of students who graduated at the end of the previous year; once another time step passes, they are no longer considered.

The number of students new to the program each year is N_{in}. Since these students are drawn from the general population, but controlled by both the admissions process and demand, N_{in} is scaled by a variable a to represent this. For example, a might be equal to 1 when there is a constant influx of students each year, or it may be subject to small changes (such as an increase in student interest) or large (such as program cancellation pushing a to 0). Variation in N_{in} each year is therefore controlled by a in the previous year.

If the values of p_i, r_i remain constant, the *proportions* of students in different stages of a population tend towards a stable distribution, which is calculable without iterating over equation 2. Instead it is governed by the right eigenvalue equation, $\mathbf{A}\,\mathbf{v}_m = \lambda_m \mathbf{v}_m$, in which $\mathbf{v}_m$ is a vector that represents the stable population up to some scale factor; the corresponding eigenvalue λ_m is the factor by which each stage population changes with every time step.

If the incoming population *also* remains constant, this leads to a stable population overall, which is reached after just a few iterations of equation 1. Mathematically, this is a direct result of setting $a = 1$ in equation 2: as the matrix **A** is a lower-triangular matrix (where all entries above the diagonal are zero), the eigenvalues are simply the diagonal entries. Since there is (in this model with $a = 1$) a constantly replenishing source population, at least one unit eigenvalue exists, indicating the existence of a stable population distribution.[2] The corresponding eigenvector $\mathbf{v}_{\lambda=1}$ is a scaled population vector; the actual i^{th} stage population is then calculated by

$$N_i = a \times N_{in} \times \frac{v_i}{v_1} \qquad (3)$$

Although equation 3 allows us to predict the number of graduates in a given year (N_g), a slightly different parameter of interest, is the proportion of a cohort that will graduate within the nominal length of the course (call this m years), and how many will graduate one, two, or more years later.

Given the model and parameters p_i, r_i, we can write these as

[1] $t + 1$ should be the minimum time in which a student is able to progress to the next stage. For most higher education programs, this will be one semester; for simplicity here, we set it to be one year.

[2] While alternative eigenvalues of the course exist (indeed, they are the other entries along the diagonal: the r_i and 0), the corresponding eigenvectors are either trivial (such as [0,0,0,0,1]), non-physical (containing “negative” students), or non-meaningful (the first element is 0, with no incoming students and a quickly dwindling population).

$$\begin{aligned}
\text{graduation } (m \text{ years}) &= \prod_{l=1}^{m} p_l \\
\text{graduation } (m+1 \text{ years}) &= \sum_{j=1}^{m} r_j \prod_{l=1}^{m} p_l \\
\text{graduation } (m+2 \text{ years}) &= \sum_{j=1}^{m} \sum_{k \geq j}^{m} r_j r_k \prod_{l=1}^{m} p_l
\end{aligned} \tag{4}$$

3. Case Study

A short online program from an Australian university that comprises five consecutive courses (101–105) is used to demonstrate the use of the matrix. This program, which commenced in 2017 with 97 students in 101, teaches app development. The program has been running for six terms with a new cohort entering 101 each term. The original program structure allows for students to progress to subsequent courses without successfully passing those earlier (however, a passing mark in course 105 *is* required to graduate), but a smaller dataset, limited to only students who passed early courses progressing to later courses, was constructed; all other students were considered to have dropped out. While this process is expected to affect the overall pass rates of the courses, we consider the dataset to be reasonable to demonstrate our matrix model. We followed 312 unique students and five instances of course 101, four of 102, three of 103, three of 104, and two of 105. We calculated the average rate of progression and repeating, observing a decrease of student influx at an average rate of *a* = 0.74 each term. This value of *a*, along with the resultant progression rates, is shown in the program matrix below (equation 5). Note that, in this program, there were no recorded instances of students repeating a failed course the following term.

$$\begin{bmatrix}
0.74 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.52 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.64 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.73 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.66 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.77 & 0
\end{bmatrix} \tag{5}$$

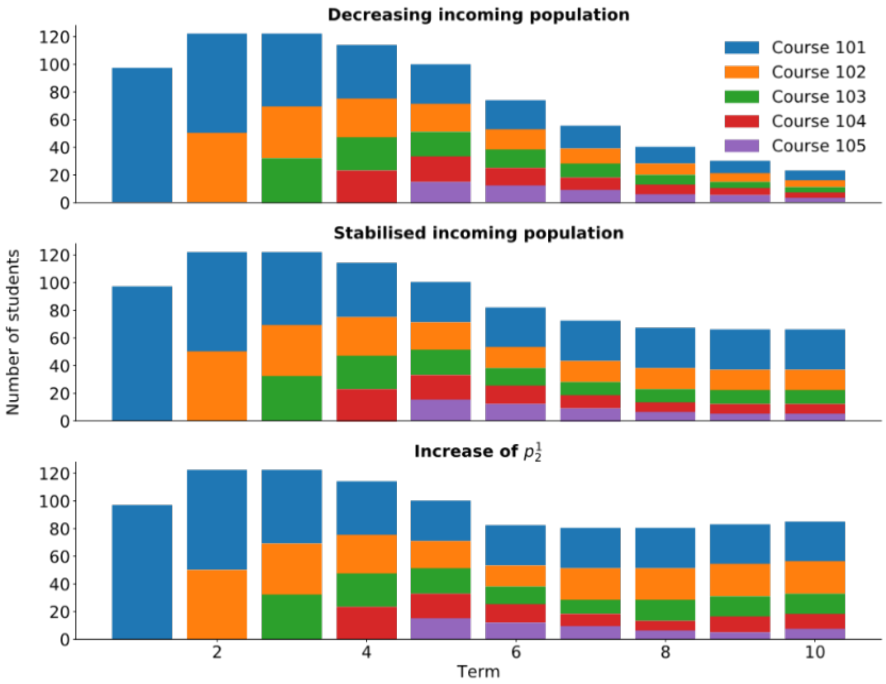

Figure 2. The course populations over time for a program with five courses taken consecutively, for three differing scenarios.

The incoming cohort decreased in number each term (a <1, highlighted in blue in equation 5; see the blue area of Figure 2), causing the overall populations in all courses to reach a peak (as the first group of students pass through) then decline (Figure 2, top), as fewer and fewer students enter the program. If the number of new students to the program were to stabilize ($a \rightarrow 1$), the subsequent course populations themselves would then stabilize (Figure 2, middle; stabilization occurs at $t = 5$.) A third scenario could aim to increase progression from the first course to the second by, say 80% through course-level interventions (observed for $p^1{}_2$, the rate of progression from course 101 to 102; yellow highlight in equation 5). This targeted intervention would cascade to increasing the graduation population soon after (Figure 2, bottom; increase in $\mathrm{p}^1{}_2$ occurs at $t = 6$). This simple example demonstrates how course-level changes have significant program level implications. This concept will be explored further in section 5. The highest rate of drop-out in our dataset occurred after course 101, when 48% of students failed the course and did not repeat it. If an intervention at this point convinced some of these students to instead repeat the course rather than dropping out completely, the overall graduation rate of the program would increase according to equation 4 (Figure 3, dashed line). However, if the intervention were introduced earlier, and instead was directed towards assisting those students to pass the course and progress to 102, this would lead to an even greater increase in the final graduation rate (solid line).

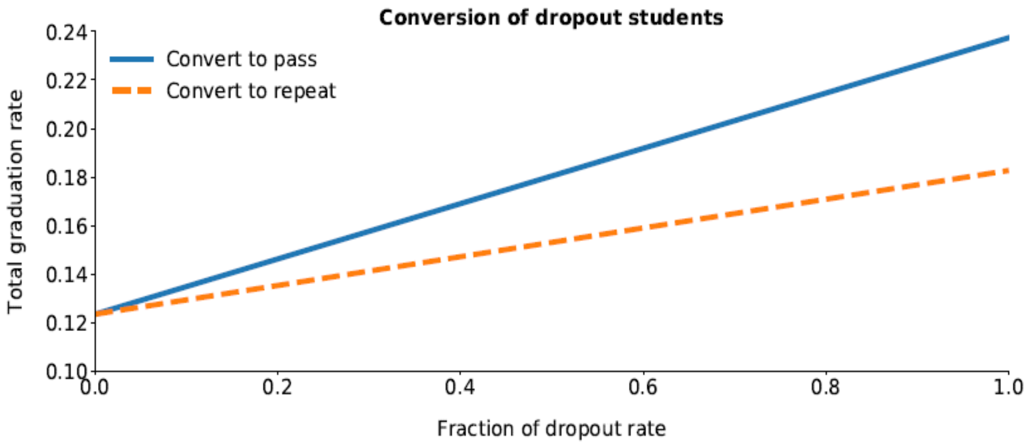

Figure 3. Changes to the total graduation rate if students who dropped out were converted to either pass the subject and progress (solid blue line) or repeat the subject (dashed orange line).

4. Extensions to the Model

The model described above focuses on a linear progression of single courses until graduation. We present this simplistic scenario to establish the mathematical relationship between retention, progression, attrition, and graduation. However, universities often offer programs with varying degrees of complexity that may have multiple pathways in the form of combination of majors, minors, and elective courses; therefore we present an expansion of the model incorporating these multiple pathways. Further, students may be co-enrolled in more than one course per term, as is the case with full-time students. We therefore provide an expansion of the model that incorporates co-enrollment. These two extensions of the model are probably the most common pathways found in universities, yet our aim is not to be exhaustive, as there may be many other different pathways. We focus on presenting a model that provides generality and incorporation of first principles, and importantly the incorporation of course-level learning and teaching elements (section 5) allowing practitioners to incorporate their own pathways.

4.1. Multiple Pathways

Often, university students may follow multiple pathways through a program, depending on electives and major and minor components. Our model can be easily extended to handle such scenarios, by redefining “stages” as simply courses that students can move between, instead of requiring that they pass through each stage. This is represented again by a matrix, **A**, where we change the notation slightly: $p^i{}_j$ is the probability of progressing from course i to course j, or of repeating when i=j. The total progression rate from course i then becomes $\Sigma i p^i{}_j$. For example, a hypothetical two-year program with two possible courses in each year (A/B and C/D), and all four possible pathways leading to graduation, is represented as

$$\begin{bmatrix} a & 0 & 0 & 0 & 0 & 0 \\ e_A & r_A^A & f_B^A & 0 & 0 & 0 \\ e_B & f_A^B & r_B^B & 0 & 0 & 0 \\ 0 & p_C^A & p_C^B & r_C^C & f_D^C & 0 \\ 0 & p_D^A & p_D^B & f_C^D & r_D^D & 0 \\ 0 & 0 & 0 & g^C & g^D & 0 \end{bmatrix} \begin{bmatrix} N_{\text{in}} \\ N_A \\ N_B \\ N_C \\ N_D \\ N_g \end{bmatrix}_t = \begin{bmatrix} N_{\text{in}} \\ N_A \\ N_B \\ N_C \\ N_D \\ N_g \end{bmatrix}_{t+1} \qquad (6)$$

Here, as examples, p_C^A is the proportion of students passing course *A* and progressing to course *C*, r_A^A is the proportion who fail and repeat course *A*, and f_B^A is the proportion who fail course *A* and take instead the alternative for the year, course *B*. e_A and $e_B = 1 - e_A$ are the proportion of the incoming population that take courses *A* and *B* respectively. The total progression rate from course *A* is given by $p_C^A + p_D^A$, and the total graduation rate from the program is the sum of g^C and g^D weighted by N_C and N_D.

This model then affords visibility over the pathways that student groups take through a program. If *A* → *C* and *B* → *D* are very separate pathways, p_C^A and p_D^B will be large relative to p_D^A and p_C^B; while similar values would indicate the flexibility of the program regarding the accessible range of course combinations. With many more courses available, the matrix of equation 6 would become large. However it would retain the simple block structure that displays pathway steps at a glance. Course instructors with little visibility over the flow of students through a program are able to see both where their students are coming from (with $p_{course}^i N_i$), and where they are likely to go next (with p_j^{course}).

4.2. Simultaneous Courses

The final extension to the matrix model explored here is to allow multiple courses undertaken simultaneously, reflecting the fact that most students study several courses each semester. In this scenario the *Ni* become *enrollments* in course *i*, rather than *students*, so a single student is represented in *n* of the *Ni*, where *n* is the number of courses undertaken simultaneously. The p_j^i become $\frac{1}{n} p_j^i$, where p_j^i is still the probability of doing course *i* followed by course *j*, however, without the requirement that a student have *passed* course *i*; that is, the visibility over course rates is lost, but the visibility of pathways is retained. However, when displayed alongside a table of pass/drop/repeat rates for each course, the story of how each course sits within the context of the broader program is clear.

Finally, this extension also allows for the inclusion of part-time students, where the $\frac{1}{n} p_j^i$ term becomes

$$\frac{\sum_s \frac{1}{n_s}}{N_i} \qquad (7)$$

where the sum is over all students who took course *j* immediately after course *i*.

5. Integrating Course-Level Elements

Markov chain models and population-level models are often created with the purpose of following cohort progression (Chiappe & Rodríguez, 2017; Nichols, 2007; Shah & Burke, 1999). These models often focus on high-level variables; in the case of higher education, these would be attrition, retention, progression, and graduation as a function of program type, illuminated by demographic data (Nichols, 2007; Shah & Burke, 1999). Yet, these matrix-based models can offer greater insights into specific courses and provide the ability to incorporate teaching and learning components. The same can be said with more complex models focusing on intercourse relationships and student pathways (Raji et al., 2017), where the focus was to visualize student trajectories of their university experience.

So far, the matrix model described in previous sections takes the p_j^i as single variables that can be replaced with their measured values. However, much effort in the learning analytics community has gone towards developing predictive models of student performance within a course (Daud et al., 2017; MacFadyen & Dawson, 2012; Stretch et al., 2015), as well as the likelihood of students dropping out (Cohen & Shimony, 2016; Dekker, Pechenizkiy, & Vleeshouwers, 2009; Yukselturk, Ozekes, & Türel, 2014). If we assume that the only other option is to repeat a subject, these can be combined to predict when a student is likely to repeat a course. Alternatively, independent models of student retention following failure of a course may be developed.

Previously developed course-level models of student performance can be integrated into our stage-based program-level model to study the larger-scale impacts of course-level changes. As a simple example, let the probabilities of student x passing or repeating course i, pix and rix, be functions of a limited set of variables: 1) individual student raw ability (encapsulated in their GPA), 2) level of engagement (denoted Ex), and 3) teaching-related variables associated with the course, such as teacher engagement (Ti) and the so-called course infrastructure (for example, the resources available for students, or the LMS design, denoted Ii). All of these variables (and many more) have been incorporated into many predictive models of student outcomes, dependent on or impacted by student behaviour (e.g., Brennan et al., 2019; Daud et al., 2017; MacFadyen & Dawson, 2012; Stretch et al., 2015), teaching practice (Bangert, 2008; Yukselturk et al., 2014), and course infrastructure (Fritz, 2016; Garrison & Cleveland-Innes, 2005; Gašević, Dawson, Rogers, & Gašević, 2016). Notably, these are expected to be heavily interrelated (Garrison, 2011; Garrison & Cleveland-Innes, 2005).

Table 1. Hypothetical Cases of Course-Level Changes and their Program-Level Impacts*

Course-level change	**Potential program-level impact**
The required entrance score to a program is increased.	The average GPA increases for first-year students, increasing p_1 according to equation 8. p_2 and later pis would also be lifted as the incoming cohort average GPA is pushed up everywhere.
Teachers are given additional funding to hire TAs.	Teacher engagement (Ti) increases, pushing up pi (with ri being correlated) for the relevant course, with flow-on effects for all later stages.
A new, more engaging LMS system is introduced institute-wide.	All Iis increase; student engagement (Ei) may also lift, pushing up pis at all stages.
A program is introduced to offer students support if they fail a course in first year.	Retention increases and r_1 is lifted, eventually increasing the numbers of students at all stages in the program.
Assessment objectives are incorporated into program-level outcomes.	Mapping of assessment difficulty influencing progression and skill acquisition relative to program outcomes is achieved.

* *Note:* In all cases, application of equations 3 and 4 would enable projection of the resulting population and graduation rates.

Within this structure, the probabilities of student x passing or repeating stage i can be written as

$$\begin{aligned} p_{ix} &= f\,(GPA_{ix},E_{ix},T_i,I_i) \\ r_{ix} &= g(GPA_{ix},E_{ix},T_i,I_i), \end{aligned} \tag{8}$$

where f and g are functions that represent the course-level models of progression and of repeating a course. The actual forms of these will, in general, be highly dependent on the specific program and institution. However the methodology described here is designed to be generally applicable.

The average probability of progression for all students within stage i is pi = Σx pix, and likewise for ri. These can then be integrated into the program-level model through the matrix A. In this way, if one or more of the input variables of equation 8 were altered, the flow-on effects at the program level would become explicitly clear. Some hypothetical examples are listed in Table 1. Significantly, the impact on student outcomes (through a calculated change in graduation rates, for example) could be estimated. This process could also be reverse-engineered, to identify the areas that most significantly have a positive (or negative) impact on student outcomes, and then direct resources to these areas.

In the final stage in a program (e.g., year 3), the g_i can be further expanded to incorporate end-of-program elements in the same fashion as shown in equation 8. For example, these stages can incorporate exit examinations or aspects associated with senior students, including elements about their next phase, such as internship experience, job interviews, or support with CV construction, for example.

The course-level equations introduced in this section can prove very powerful as they can help map how course-level interventions affect program-level progression and vice versa. One important challenge in learning analytics is to ensure that student progression is correctly associated with observed and expected learning outcomes (Gašević et al., 2016; Munguia, 2020; Ochoa, 2016).

6. Using the Model: Identifying and Incorporating Solutions

We have presented the mathematical description of a model incorporating course- and program-level metrics to better understand student progression through a program. The elegance of the model allows for user input required in two places (Figure 4). First the cohort vector, **v**, needs to have the initial state (first element) of an incoming cohort. Next, the pathways must be defined, presumably the most common ones or the ones of relevance for a particular analysis. Since this is a program-specific model, it is not important to know where the d_i students end up, just that they have left the program. By defining the pathways, matrix **A** is being populated. The model can be run iteratively, and the cohort vector will fill with expected cohort levels for each year. Varying elements of the matrix systematically allow for understanding how each element affects cohorts at different stages.

The model does have two main limitations. First, initial runs of the model must be made to understand the model itself; each element in matrix A must be treated as a proportion of students falling into each category at each stage. Therefore, it requires real data to be constructed. Fortunately, the main elements, retention, progression, graduation, and drop-outs are readily available data for each program. Just analyzing these proportions allows us to understand where the major changes in cohort stages take place, and what the drivers behind them are. Incorporating course-level data may be more time consuming but does not hold up the use of the model. However, once the model parameters are understood by the user, the proportions created in matrix A can then be used as probabilities to create predictions of future cohorts (Figure 4). This is key, as monitoring program success can be evaluated between observed (P_o) and expected (P_e) patterns, and there are various statistical tests that can be used to determine similarities or differences between P_o and P_e.

Second, student behaviour is still an unknown in this and every other Markov chain model. Therefore, it is dangerous to make assumptions associated with the student reasoning behind the pattern. Data associated with behaviours needs to be considered and complimentary models created to better understand the diversity of student behaviours (e.g., Brennan et al., 2019; Raji et al., 2017). Use of this model should focus on cohort-level patterns and responses to course- and program-level actions.

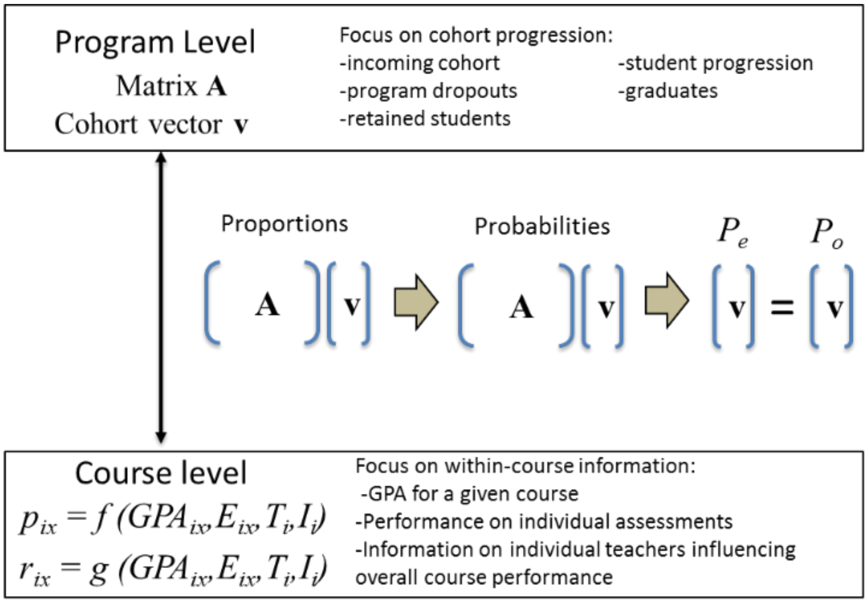
Program Level
Matrix **A**
Cohort vector **v**
Focus on cohort progression:
-incoming cohort
-program dropouts
-retained students
-student progression
-graduates
Proportions
Probabilities
P_e
P_o
A
v
A
v
v
=
v
Course level
$p_{ix} = f\,(GPA_{ix}, E_{ix}, T_i, I_i)$
$r_{ix} = g\,(GPA_{ix}, E_{ix}, T_i, I_i)$
Focus on within-course information:
-GPA for a given course
-Performance on individual assessments
-Information on individual teachers influencing overall course performance

Figure 4. Relationship between the program-level matrix model and the course-level functions.
Note: The matrices and vectors show the progressive use of the model, from construction of matrix A using proportions deriving from real data, to treating these as probabilities affecting student cohorts, to the final evaluation of matrix elements by comparing expected (P_e) and observed (P_o) patterns in the cohort vectors.

When students are exposed to these models, they can quickly understand the relationship between course sequence and program-level success (Raji et al., 2017). The model shows students how a particular sequence of three courses generates better grades over time in computer science. Here, our model expands on these insights, as it can incorporate assessment-level information and teacher-level variables such as engagement and quality. Therefore, it is recommended that students be exposed to these models, and their feedback considered when making decisions to better understand student decision-making.

7. Conclusion

Scaling learning analytics from courses to programs provides a thorough pedagogical view of how program objectives can be achieved through coordination and visibility of student pathways. The matrix model of progression describes the flow of students through a program, by including quantitative data on rates of progression, repeated courses and drop-out, and displaying these in a manner that visualizes the pathways students are taking through their program. With this model, program managers can understand how their cohorts are increasing or decreasing, where to direct resources to have the greatest impact, and where bottlenecks are located in a program, allowing for better planning. Course coordinators can see where their course sits within a program, where their students are coming from, and where they are likely to go.

To see how the pathways taken by successful students differ from those taken by students who perform poorly, the progression rates of a program matrix could be recalculated following a filtering of the input data to particular groups of students (Munguia, 2020). Further understanding the choices behind these alternative pathways would then assist in developing recommended pathways or other options for the latter group. Like the work of Campagni et al. (2015), our model allows for comparison between alternative pathways through a program; however, it also provides clear visibility of the pathways themselves alongside the rates of progression, failure, and drop-out at each stage. Insight may be garnered into alternative pathways that differ from those pre-defined by program experts in unexpected ways, or into pathways that lead to the greatest likelihood of success by all students, or that confirm whether a program has been well structured. It is particularly important to support non-traditional students (who may be older, recent immigrants, first generation university attendees, or studying part-time), so being able to both understand the pathways through a program that these students take — as well as devise course-level strategies to support them and predict the program-level impact — are important implications of this work.

Integrating course-level predictive models extends this model from descriptive and diagnostic analytics (which remain extremely useful) towards the prescriptive realm, where the impacts of course level changes can be modelled and explored before being recommended for implementation, and where hypotheses around student behaviours, instructor engagement, and program qualities can be tested. For example, the level of program-level visibility this model offers — alongside course-level predictive models of student outcomes — means the model can be expanded to address whether particular assessments or skill sets are located in the right course. It could also be extended to include graduate outcomes such as employment and satisfaction, since these feed back to influence potential incoming students through university rankings and reputation. Unifying the analytics of program and course through such a model encourages us to consider each in the context of the other, and understand the impact that changes made in one arena will have on the other.

Declaration of Conflicting Interest

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

No funding was received for this project.

Acknowledgements

We would like to thank the various people who provided constructive criticism during several iterations of this project.

References

Asif, R., Merceron, A., Ali, S. A., & Haidera, N. G. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers & Education, 113*, 177–194. https://dx.doi.org/10.1016/j.compedu.2017.05.007

Bangert, A. (2008). The influence of social presence and teaching presence on the quality of online critical inquiry. *Journal of Computing in Higher Education, 20*(1), 34–61. https://dx.doi.org/10.1007/BF03033431

Brennan, A., Sharma, A., & Munguia, P. (2019). Diversity of online behaviours associated with physical attendance in lectures. *Journal of Learning Analytics*, *6*, 34–53. https://dx.doi.org/10.18608/jla.2019.61.3

Campagni, R., Merlini, D., Sprugnoli, R., & Verri, M. C. (2015). Data mining models for student careers. *Expert Systems with Applications, 42*(13), 5508–5521. https://dx.doi.org/10.1016/j.eswa.2015.02.052

Caswell, H. (2006). *Matrix population models*. Wiley Online Library. https://dx.doi.org/10.1002/9780470057339.vam006m

Chiappe, A., & Rodríguez, L. P. (2017). Learning analytics in 21st century education: A review. *Ensaio: Avaliação e Políticas Públicas em Educação, 25*(97), 971–991. https://dx.doi.org/10.1590/s0104-40362017002501211

Cohen, A., & Shimony, U. (2016). Dropout prediction in a massive open online course using learning analytics. *Proceedings of the World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education* (E-Learn 2016) 14–16 November 2016, Washington, DC, USA (pp. 616–625). Chesapeake, VA: Association for the Advancement of Computing in Education (AACE).

Daud, A., Aljohani, N., Abbasi, R., Lytras, M. D., Farhat Abbas, F., & Alowibdi, J. S. (2017). Predicting student performance using advanced learning analytics. *Proceedings of the 26th International Conference on World Wide Web Companion* (WWW '17 Companion), 3–7 April 2017, Perth, Australia (pp. 415–421). New York: ACM. https://dx.doi.org/10.1145/3041021.3054164

Dekker, G., Pechenizkiy, M., & Vleeshouwers, J. (2009). Predicting students drop out: A case study. In T. Barnes et al. (Eds.), *Proceedings of the 2nd International Conference on Educational Data Mining* (EDM2009), 1–3 July 2009, Cordoba, Spain (pp. 41–50). International Educational Data Mining Society.

Fritz, J. (2016). LMS course design as learning analytics variable. In J. Greer, M. Molinaro, X. Ochoa, & T. McKay (Eds.), *Proceedings of the 1st Learning Analytics for Curriculum and Program Quality Improvement Workshop* (PCLA 2016), 25 April 2016, Edinburgh, UK (pp. 15–19).

Garrison, D. R. (2011). *E-learning in the 21st century: A framework for research and practice*. Milton Park/Abingdon, UK: Taylor & Francis.

Garrison, D. R., & Cleveland-Innes, M. (2005). Facilitating cognitive presence in online learning: Interaction is not enough. *The American Journal of Distance Education, 19*(3), 133–148. https://dx.doi.org/10.1207/s15389286ajde1903_2

Gašević, D., Dawson, S., Rogers, T., & Gašević, D. (2016). Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success. *The Internet and Higher Education, 28*, 68–84. https://dx.doi.org/10.1016/j.iheduc.2015.10.002

Golding, P., & Donaldson, O. (2006). Predicting academic performance. *Proceedings of the 36th Annual Frontiers in Education Conference* (FIE 2006), 27 October–1 November 2006, San Diego, CA, USA (pp. 21–26). Washington, DC: IEEE Computer Society.

Jeffreys, M. R. (2007). Tracking students through program entry, progression, graduation, and licensure: Assessing undergraduate nursing student retention and success. *Nurse Education Today, 27*(5), 406–419. https://dx.doi.org/10.1016/j.nedt.2006.07.003

Jeffreys, M. R. (2015). Jeffreys's nursing universal retention and success (NURS) model: Overview and action ideas for optimizing outcomes A-Z. *Nurse Education Today, 35*(3), 425-431. https://dx.doi.org/10.1016/j.nedt.2014.11.004

Kabakchieva, D., Stefanova, K., & Kisimov, V. (2010). Analyzing university data for determining student profiles and predicting performance. In M. Pechenizkiy et al. (Eds.), *Proceedings of the 4th Annual Conference on Educational Data Mining* (EDM2011), 6–8 July 2011, Eindhoven, Netherlands (pp. 347–348). International Educational Data Mining Society.

Lefkovitch, L. P. (1965). The study of population growth in organisms grouped by stages. *Biometrics, 21*(1), 1–18. http://dx.doi.org/10.2307/2528348

Macfadyen, L. P., & Dawson, S. (2012). Numbers are not enough: Why e-learning analytics failed to inform an institutional strategic plan. *Journal of Educational Technology & Society, 15*(3), 149.

Mendez, G., Ochoa, X., Chiluiza, K., & de Wever, B. (2014). Curricular design analysis: A data-driven perspective. *Journal of Learning Analytics, 1*(3), 84–119. https://dx.doi.org/10.18608/jla.2014.13.6

Munguia, P. (2020). Preventing student and faculty attrition in times of change. In D. Burgos (Ed.), *Radical solutions in higher education* (pp. 115–129). Netherlands: Springer. https://dx.doi.org/10.1007/978-981-15-4526-9_8

Nicholls, M. G. (2007). Assessing the progress and the underlying nature of the flows of doctoral and master degree candidates using absorbing Markov chains. *Higher Education, 53*, 769–790. https://dx.doi.org/10.1007/s10734-005-5275-x

Nghe, N. T., Janecek, P., & Haddawy, P. (2007). A comparative analysis of techniques for predicting academic performance. *Proceedings of the 37th Annual Frontiers in Education Conference* (FIE 2007), 10–13 October 2006, Milwaukee, WI, USA (pp. T2G–7). Washington, DC: IEEE Computer Society.

Ochoa, X. (2016). Simple metrics for curricular analytics. In J. Greer, M. Molinaro, X. Ochoa, & T. McKay (Eds.), *Proceedings of the 1st Learning Analytics for Curriculum and Program Quality Improvement Workshop* (PCLA 2016), 25 April 2016, Edinburgh, UK (pp. 20-24).

Pechenizkiy, M., Trcka, N., De Bra, P., & Toledo, P. A. (2012). CurriM: Curriculum mining. In K. Yacef, O. Zaiane, A. Hershkovitz, M. Yudelson, & J. Stamper (Eds.), *Proceedings of the 5th International Conference on Educational Data Mining* (EDM2012), 19–21 June 2012, Chania, Greece (pp. 216–217). International Educational Data Mining Society.

Raji, M., Duggan, J., DeCotes, B., Huang, J., & Vander Zanden, B. (2017). Modelling and visualizing student flow. *IEEE Transactions on Big Data*. https://dx.doi.org/10.1109/TBDATA.2018.2840986

Shah, C., & Burke, G. (1999). An undergraduate student flow model: Australian higher education. *Higher Education*, *37*, 359–375. https://dx.doi.org/10.1023/A:1003765222250

Shea, P., Li, C. S., & Pickett, A. (2006). A study of teaching presence and student sense of learning community in fully online and web-enhanced college courses. *The Internet and Higher Education, 9*(3), 175–190. https://dx.doi.org/10.1016/j.iheduc.2006.06.005

Stretch, P., Cruz, L., Soares, C., Mendes-Moreira, J., & Abreu, R. (2015). A comparative study of classification and regression algorithms for modelling students' academic performance. In O. C. Santos et al. (Eds.), *Proceedings of the 8th International Conference on Educational Data Mining* (EDM2015), 26–29 June 2015, Madrid, Spain (pp. 392–395). International Educational Data Mining Society.

Yehuala, M. A. (2015). Application of data mining techniques for student success and failure prediction (The case of Debre Markos University). *International Journal of Scientific & Technology Research, 4*(4), 91–94.

Yukselturk, E., Ozekes, S., & Türel, Y. K. (2014). Predicting dropout student: An application of data mining methods in an online education program. *European Journal of Open, Distance and E-learning, 17*(1), 118–133. https://doi.org/10.2478/eurodl-2014-0008