Mining Sakai to Measure Student Performance: Opportunities and Challenges in Academic Analytics

(Research in Progress)

Eitel J.M. Lauría
School of Computer Science and Mathematics
Marist College
Poughkeepsie, NY 12601
Eitel.Lauria@marist.edu

Joshua Baron
Senior Academic Technology Officer
Marist College
Poughkeepsie, NY 12601
Josh.Baron@marist.edu

Abstract

In this paper we lay out ongoing work on the Open Academic Analytics Initiative (OAAI)[1], a project aimed at developing, deploying and releasing an open-source environment for academic analytics designed to increase student content mastery, semester-to-semester persistence and degree completion in higher education. .As a result, we expect to see increases in adoption of academic analytics, particularly among institutions using the open-source Sakai Collaboration and Learning Environment, in both the short- and long-term. The paper provides a preliminary report on how this project intends to address the use of academic data (combining course management system logged data with student records and demographics) to create data mining models that can help predict student performance and take corrective actions.

Subject areas: Higher Education, Academic Analytics, Data Mining, Enterprise Systems

1. Introduction

The Open Academic Analytics Initiative (OAAI) aims at addressing one of the most critical needs facing higher education and our nation today: increasing the number of students who complete postsecondary degrees. Even those of us who work in higher education are often shocked by the evidence of what has become a national tragedy. Across all types of four-year institutions, of those students starting bachelor degree programs in 2001, only 36% completed them within four years. If we look at graduation rates over six years for this same population the figure improves to 58% yet it leaves one to wonder what happened to the 42% who did not succeed. Even more alarming, the four-year degree completion rate drops to 21% and 25% for Black and Hispanic students respectively [1]. Similarly, only 28% of all students pursuing certificates or associates degrees in 2004 from two-year institutions completed their programs within three years. As a result, the United States now ranks 12th in the world in the percentage of 25- to -34-year-olds with an associates degree or higher [2]. Given the importance that educational attainment now plays in our ability to compete in a global marketplace, it is not surprising that leaders from both the public and private sector are calling for new and innovative strategies to address this national crisis.

[1] The Open Academic Analytics Initiative is funded through the Next Generation Learning Challenges (NGLC), a collaborative, multi-year grant program aimed at increasing college readiness and completion through applied technology. NGLC partners include EDUCAUSE, the League for Innovation in the Community College, the International Association for K-12 Online Learning (INACOL), and the Council of Chief State School Officers (CCSSO). Funding is being provided by the Bill & Melinda Gates Foundation and The William and Flora Hewlett Foundation.

Academic analytics is the term used to describe the application of data mining techniques to develop predictive models that can help monitor and anticipate student performance and take action in issues related to student teaching and learning. Academic analytics, which "combines select institutional data, statistical analysis, and predictive modeling to create intelligence upon which students, instructors, or administrators" [3] can act as a means to improve academic success, holds great potential to provide new and innovative technological tools for improving course and degree completion. EDUCAUSE [4] provides a broader definition as the "intersection of technology, information, management culture, and the application of information to manage the academic enterprise".

Data mining, a discipline combining elements of artificial intelligence and applied statistics, is the process of extracting patterns from large data sets. Data mining is applied successfully in a wide scope of domains ranging from business and scientific settings to law enforcement. Applications include customer profiling, cross-selling, fraud detection, drug discovery, intrusion detection, and DNA sequencing among many others.

Academic analytics has received considerable attention within higher education, including being highlighted in the recently released 2011 Horizon Report [5]. This interest can, in part, be traced to the work at Purdue University which has moved the field of academic analytics from the domain of research to practical application through the implementation of Course Signals [6]. Results from initial Course Signal pilots between fall 2007 and fall 2009 have demonstrated significant potential for improving academic achievement. The continuous stream of data being collected, and stored in course management systems paired with academic and demographic student data can be applied as input to build predictive models using data mining techniques that can be used to implement data driven decision making practices.

Despite this early success, academic analytics remains an immature field that has yet to be implemented broadly across a range of institutional types, student populations and learning technologies. Previous work done at other institutions as in the case of Purdue university can be used as a framework of reference for research and practice, but cannot be applied directly in a different academic context as there may (considerable) differences in student populations, types of academics institutions and the course management systems put in place. As pointed out by [7] , other course management systems may collect different tracking information, and therefore, the comparison with samples from other course management systems may not be valid. This work investigates the issue of using data extracted from the Sakai Collaborative Environment as input to the predictive modeling tasks. Starting in 2006, Marist College has transitioned to Sakai as the course management system of choice. Sakai is an open source platform developed by a large consortium of higher education institutions and used by many of these colleges and universities, including: Stanford University, Yale University, University of Michigan, Indiana University and University of Cambridge among others. Marist College is a prominent member of this consortium. A customized version of Sakai known as *ilearn* was implemented at Marist College bringing with it a wealth of innovative instructional tools and enterprise-level features, among them extensive event logging capabilities that could be exploited as data collection tools for use in academic analytics.

The OAAI will focus on developing an open-source ecosystem for teaching and learning activities that also produces learning analytics that can drive effective educational interventions designed to improve student engagement and degree completion. To support real-world adoption, OAAI will base its development on open-source technologies already in widespread use at educational institutions, and on established protocols and standards that will enable an even wider variety of existing open-source and proprietary technologies to make use of OAAI code and practices. To further advance the field of academic analytics, the OAAI, will:

- Lay out a research methodology to collect and pre-process input data, and develop models that can be used to perform inferential queries on student performance based on course management system (Sakai) data and student academic records.
- Research the portability of predictive models used in academic analytics to better understand how models developed for one academic context can be effectively deployed by other institutions and overtime, be enhanced through open-source community collaboration
- Develop a portable API that will capture user activity data for use by data mining tools in academic analytic tasks.
- Document best practices related to deploying academic analytics using Sakai and the open source data mining tools

In this paper we lay out the methodological framework for the development of predictive models of student success. The initial pilot will use data from Marist College using open source machine learning (data mining) software[2], utilizing previous work at Purdue University as a reference model for comparison purposes. We don't report results as we are in the early stages of execution of the project. The approach is both informative and comprehensive in terms of the proposed techniques at each step of the research framework with the purpose of providing awareness and exposure to alternative ways of doing academic data analysis. In section 2 we provide a brief account of previous work in academic analytics; we follow in section 3 with a detailed description of the methodological framework; in section 4 we discuss data quality issues and its impact on the training of supervised learning models. We conclude with final remarks and pointers to future work. A full depiction of the data (predictors and target features) is provided in the Appendix.

2. Previous Work in Academic Analytics

Academic analytics is a new discipline that has emerged in higher education as a follow up of the successful application of data mining and predictive analytics in the business world [3]. Although still a developing field, ongoing work in this area has shown promising results. In 2005, researchers at the University System of Georgia were able to predict with up to 74% accuracy, based on high school GPA and SAT mathematics scores, the likelihood that a student would successfully complete an online course [3].

[2] Work will be developed in Pentaho (http://www.pentaho.com/), an open source business intelligence suite.

Romero et al [8] have investigated the application of data mining techniques using Moodle data (a popular open source course management system). Talavera [9] used clustering to discover patterns reflecting user behaviors in learning management systems. Laurie and Timothy [10] used data mining as a strategy for assessing asynchronous discussion forums in online courses. Campbell [7] combined factor analysis and logistic regression to develop predictive models trained with data extracted from course management system usage and student demographics. Recently Purdue University, based Campbell's seminal dissertation [7], has moved the field of academic analytics from the domain of research to practical application through the implementation of Course Signals. Acting as an early academic warning system, Course Signals, now supported by SunGard Higher Education (SGHE), utilizes "data collected by instructional tools (such as the course management system) to determine in real time which students might be at risk" [11] to not complete their course. Once identified, these students can receive "interventions" via notifications sent by their instructor which guide them to appropriate academic support resources, such as online practice exams or tutoring assistance, along with encouragement to use them. Results from initial Course Signals pilots between fall 2007 and fall 2009 demonstrate the significant potential this type of early warning system holds for improving course completion, semester-to-semester persistence rates, and mastery of content learning outcomes. For example, in a "gateway" Biology course with 300 students there were 12% higher levels of B and C course grades in sections using Course Signals versus control sections and a corresponding 14% decrease in the number of Ds and Fs [11]. Although more longitudinal data will need to be collected to determine impacts on six-year cohort graduation rates, data from the past several years has shown 6-10% improvements in freshman-to-sophomore and sophomore-to-junior year retention rates [12]. Shifts of this nature provide initial evidence that early warning systems coupled with interventions can improve content mastery, course completion, and persistence rates. As a result of these early successes, a number of institution-specific academic analytics projects have been started in the last few years, including University of Maryland - Baltimore County ("Check My Activities" project), Grand Rapids Community College (Project ASTRO), and Northern Arizona University (Grade Performance Status or GPS project) (Next Generation Learning Challenges, n.d.), demonstrating a growing interest in this technology.

3. Methodological Framework

One of the main goals of the OAAI is to develop predictive models of student performance using student data. The term 'prediction' is generally used to characterize models (based on statistical techniques and data mining algorithms) designed for predicting new outcomes or scenarios based on new observations. Prediction is different from 'explanation', where the goal is to build models that explain underlying causal structure and to assess the explanatory power of such models. Explanatory models are perhaps the most typical in empirical research in social science; they have been the dominant force in IS empirical research in topics related to determining causality and critical factors of technology acceptance and usage, and technology implementation. In spite of the momentum gained by disciplines such as data mining and predictive analytics, their use in IS research has been rare. As pointed out by [13], the different context in which explanatory and predictive modeling operate (testing causation based

on theoretical hypothesis vs. data driven prediction) gives way to differences in the kind of variables used in the modeling process, the goal of the model building process and its constraints; and the metrics used for model evaluation. Shmueli and Koppius organize these differences along the following dimensions (see [13] for more details):

- Variables of interest: variables in explanatory models are usually conceptual constructs drawn out of a dimensionality reduction process through transformation to a low dimensional feature space (e.g. factor analysis). The focus is on analyzing the causal relationships between those theoretical constructs. Instead, in predictive modeling the focus of the analysis are usually the observed variables themselves or a subset of them usually selected through a feature selection process; the approach is more pragmatic, as prediction does not seek interpretation.
- Model building goal: In explanatory models the goal is to maximize model fit, whereas predictive models are more concerned with good generalization (better predictive models are those that make better predictions based on new observable data)
- Constraints: Explanatory models must lead to causal interpretation, which, means that the variables (theoretical constructs) and their relationships must be meaningful and aligned with theory. Instead, predictive modeling is data driven: interpretation is not critical
- Model assessment: explanatory power is measured according to goodness of fit (e.g., R^2 and statistical significance of estimates) ; instead predictive models are evaluated by predictive accuracy

These differences among prediction and explanation have a direct impact on our methodological framework, as we sought to develop predictive models. An adequate balance between prediction and explanation in the context of this project might be desirable: predictive models are data oriented but nonetheless can be used for theory development as long as careful semantic constraints (based in theory) are put in place.[3] Arnold [11] points out that predictive analytics alone can do little to help students succeed academically, therefore improving our understanding of best practices related to student interventions and the factors leading to student successful performance remains an important issue.

The data mining (machine learning) models considered in our work are based on supervised techniques given that labeled training data is available (data sets used for training purposes carry both input features describing student characteristics and course management system events, as well as student academic performance).

We replicate Campbell's approach [7] (factor analysis and logistic regression) and use it as a reference model for prediction of student performance (predicting which students are in "need of help", defined as likely to receive a grade of C or lower)

[3] Causally constrained Bayesian networks provide an interesting framework for the development of probabilistic models that can be used to perform probabilistic reasoning and assess causality (more on this later on)

We subsequently develop a number of classification models based on supervised machine learning algorithms: C4/5/C5.0 decision trees, and support vector machines (SVM) classifiers; For the purpose of inference (including prediction and causal explanation), we build a Bayesian network learnt from data (more details on the planned machine learning algorithms later on).

Our methodological framework consists of six phases (see figure 1), namely Collect data, Reduce Data, Rescale/Transform Data, Partition Data, Build Models, and Evaluate Models. The first four phases deal with preparing the input data used to build (train) and evaluate models.

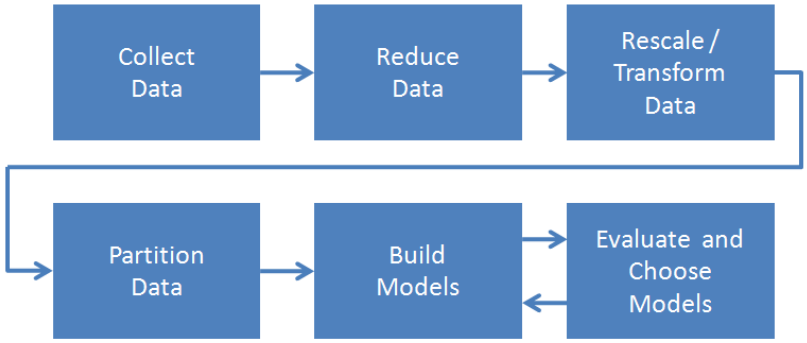

Figure 1. Methodological Framework

3.1 Phase One (Collect Data)

Phase One (Collect Data) comprises a data extraction process from diverse sources and an initial pre-processing of the data to handle missing values, outliers, extraneous (incomplete) records and calculation of derived features.

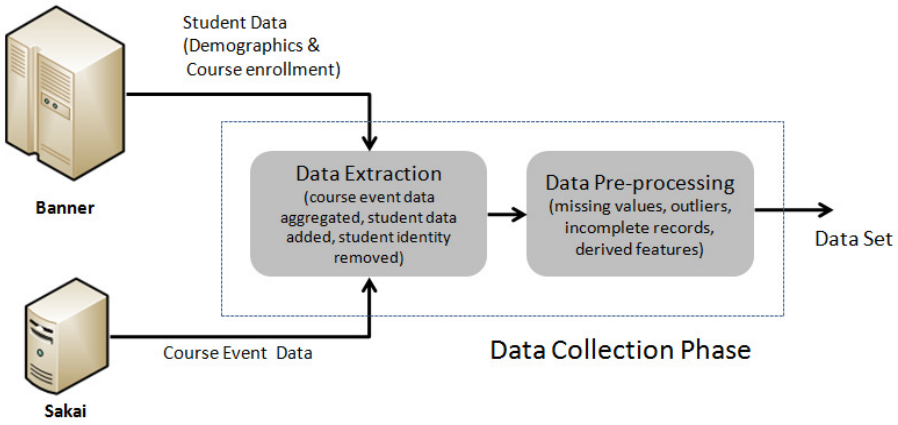

Figure 1. Data Collection Phase

Data is extracted from ilearn (Sakai's implementation at Marist College), and student demographic data and course enrollment data student records system (student data is stored and processed by Banner[4], Sungard's ERP for higher education, recently

[4] http://www.sungardhe.com/

implemented at Marist College). Identifying student information is removed during the data extraction process [5].

Data for the pilot study at Marist College consists of a sample of students enrolled in courses during one semester (Fall 2010)[6]. Sakai logs data of individual course events tracked by each of the tools used by an instructor in a given course shell: common tools include Announcements, Lessons, Calendar, Chat Room, Discussion Forums, Resources, Assignments, Test & Quizzes, and Messages (email). A detailed description of the tracking information collected by Sakai tools can be found in [14].

Course event data logged by Sakai are collected and aggregated to produce consolidated records per course and student. A record with summary measures pertaining to course events is generated for every course in which a student participated. This means that multiple cases may come from a single student (as many as the number of courses taken by a student). Each case also includes the student's academic performance in the course (e.g. the letter grade) and is enriched with student demographic data (e.g. age, gender, race, SAT scores, GPA) and course enrollment data (e.g. course name, course subject, number of students in the course). Whenever possible, course event data is extracted as a ratio or proportion rather than as an absolute value. The goal is to avoid variability introduced by differences in workload and course assessment criteria imposed by different faculty members or determined by course characteristics. For example, instead of recording the number of assignments completed by the student, we choose to record the proportion of assignments completed by a student, computed as the ratio between the number of assignments completed by the student in question and the average number of assignments completed by all students in the course. A complete description of the data set is provided in Appendix A.
Records without corresponding final course grades are removed from the data set. Potential outliers are detected at an initial exploratory data analysis stage, following the data extraction process and considered for removal. Missing data requires additional treatment (see Section 4 for a description of common data quality issues).

3.2 Phase Two (Reduce Data)

Phase Two (Reduce Data) applies dimensionality reduction techniques on the data set produced in Phase One, to reduce the number of variables and parameters requiring estimation. Most machine learning processes (on which data mining algorithms are typically based) are characterized by high dimensional data. Query accuracy and efficiency usually degrades rapidly as the dimension increases: as the number of variables and parameters grow, the number of data samples required to estimate those variables and parameters grow exponentially. This problem is sometimes referred to as the "curse" of dimensionality [15]. Thus, for a given sample size, there are a maximum number of variables and parameters that can be estimated with accuracy. Besides, not all of these

[5] The data collection process must comply with Marist College University's Human Subjects Institutional Review Board's (IRB) regulations regarding protection of human subjects. In addition to IRB, there are Family Educational Rights and Privacy Act (FERPA) issues as well that need to be addressed.

[6] If we find that the data set is too small we may then opt to also work with the spring data.

dimensions are relevant and some are redundant. There are two main approaches to reduce dimensionality: feature selection and feature transformation to a low dimensional space. Feature selection deals with selecting a subset of the original features; the original representation of the features in the data set is not changed[7] . In contrast, transformation based methods modify the input features, mapping them to a low feature space.

Multiple techniques are available for feature selection, usually grouped into three main families: filter, wrapper, and embedded methods, according to how the learning model is used in the feature selection process [16, 17]. Filter algorithms pre-select features, without running the learning model, based on the intrinsic characteristics of the data; wrapper methods, instead, wrapper look for the optimal subset of features by running a learning model on the subset; embedded methods are similar to wrappers but less computationally expensive and less prone to overfitting; they combine subset selection and learning into one single optimization problem[8]; they are usually specific to given learning algorithms (e.g. C4.5 / C5.0 decision trees [18] and certain support vector machine implementations [19]). We will use embedded methods for most of the predictive models we build in Step Four (see Table 1).

<table>
<tr><th>Reduce Data</th><th>Transform & Discretize</th><th>Partition Data</th><th colspan="2">Build Models</th><th>Evaluate and Choose Models</th></tr>
<tr><td>Linear Feature Transformation (Factor Analysis)</td><td>Transform</td><td>70% Train
20% Validate
10% Test</td><td rowspan="3">Prediction (classification)</td><td>Logistic Regression</td><td>Predictive Accuracy. Validation with held out data</td></tr>
<tr><td>Embedded Feature Selection</td><td>-</td><td>70% Train
20% Validate
10% Test</td><td>C4.5 / C5.0 Decision Tree</td><td>Predictive Accuracy. Validation with held out data</td></tr>
<tr><td>Embedded Feature Selection</td><td>-</td><td>70% Train
20% Validate
10% Test</td><td>Support Vector Machines</td><td>Predictive Accuracy. Validation with held out data</td></tr>
<tr><td>Embedded Feature Selection</td><td rowspan="2">Transform & Discretize</td><td rowspan="2">70% Train
20% Validate
10% Test</td><td rowspan="2">Inference (prediction, diagnosis, causal explanation)</td><td>Bayesian Networks

(Model Selection: search the space of BNs)</td><td rowspan="2">Average Predictive Accuracy over nodes.
Validation with held out data</td></tr>
<tr><td>Linear and Nonlinear Feature Transformation</td></tr>
</table>

Table 1. Reduce, Transform and Discretize Data; Build, Evaluate and Choose Models

[7] Except for discretization or rescaling, as described later on.

[8] Due to the nature of the feature selection technique, the feature selection and model building steps are sometimes combined. This may be the case throughout our own analyses

Feature transformation techniques include factor analysis; principal component analysis (PCA); and more recently nonlinear techniques, such as kernel PCA [20] and local linear embedding [21]. PCA computes the linear projections of greatest variance from the top eigenvectors of the covariance matrix. Factor Analysis captures the shared variance of variables, and therefore models the correlation structure of the data. Factor analysis is often preferred when the goal of the analysis is to detect structure; it is the data reduction technique of choice in social science. Nonlinear techniques are appropriate for high dimensional data containing nonlinear structures that are invisible to PCA, factor analysis and other linear techniques. With the purpose of using Campbell's approach [7] as a reference framework, we apply factor analysis to reduce course management data to its principal components and use the resulting constructs as input to the model building process with logistic regression. We will also consider using linear and non linear feature transformation when building Bayesian network models.

3.3 Phase Three (Rescale and Transform Variables)

In Phase Three (Rescale and Transform Variables) we rescale continuous distributions and/or discretize continuous variables, according to some defined scheme, depending on the requirements of the specific supervised learning algorithms considered in Step Five. For example, logarithmic transformation may be useful to accommodate any assumptions regarding the normal distribution of continuous features. Discretization is typically used in Bayesian learners (especially Bayesian networks)[9]. We will consider constraining the number of bin or intervals in the discretization process in order to reduce the number of parameters of the Bayesian network that can be accurately estimated relative to the size of the training data set. For example, in a 10 variable binary Bayesian network, where each variable can have only binary (i.e., two values), say "low" and "high," if each node has three parents, on average, there would be $10\times2^3 = 80$ parameters to estimate; if each variable is coded using a 1-5 Likert scale , the number of parameters would increase to $10\times5^3=1250$. There is evidently a tradeoff between the granularity of the discretization process and the accuracy of the parameter estimates. Consequently training data sets should be as large as possible in order to be able to attain higher measurement levels.

3.4 Phase Four (Partition Data)

In Phase Four (Partition Data), input data is divided in three subsets: a training data set, and validation data set, and a test data set. The training data set is used to build the models, Once the models are learnt from data, they have to be validated with unseen data. A validation subset is used for this purpose, in which the actual value of the target variable is known and can be use to test the accuracy of the candidate models. The validation data set is often used to fine-tune the model building process and choose among competing algorithms and architectures (different parameter configurations on a given machine learning algorithm). Once a model or set of models is chosen a final test

should be performed to compute a realistic estimate of the performance of the model(s) on unobserved data.

Partitioning the data into three distinct subsets is an ideal situation that can be constrained if there is limited input data available, given that a reduced amount of input data has a negative effect on model accuracy (model parameter estimates have large variance if the number of data samples is limited). Different heuristics have been proposed to determine the minimum amount of data needed to train accurate models. Shmueli [22] suggests a minimum of ten data samples per predictor variable as a rule of thumb. Other authors [23] have proposed that, for classification tasks, a minimum of $6 \times p \times m$ data samples be considered, where is p the number of features and m the number of class values. With limited data, partitioning must be replaced with other methods (e.g. n-fold cross-validation). In our case, as we collect a large amount of data (multiple events of a course management system over multiple courses, with multiple students), we don't envision an issue in producing sufficient data to train, validate and test the models. We propose a ratio of 80% of the data used for training, 209% for validation and 10% for testing. The ratio follows standard data mining practice.

3.5 Phase Five (Build Models)

In Phase Five (Build Models) we train different models with the training dataset, using logistic regression and three machine learning approaches (C5.0 decision trees, support vector machines and Bayesian networks).

Logistic regression: is a highly popular parametric classification method, a member of a familiy of statistical models called generalized linear models. The target (class) variable is a function of the linear combination of the predictor features. The link function is the logistic function or logit. The logistic regression model predicts the probability of occurrence of a given class value by fitting the data to a logit function. Logistic regression makes no assumption about the distribution of the predictors, but the user must decide on the inclusion of predictors in the model, as well as of any interaction terms. As in the case of linear regression, multicollinearity can have a negative effect on the parameter estimates, inflating their variance, and therefore affect the model fit.

C4.5 / C5.0 Decision trees: Developed by Quinlan [18] the C5.0 algorithm and its predecessor (C4.5) learn decision trees from data, graphical representations of rules that constitute the basis for prediction. The set of rules inferred through the algorithmic approach describe a class to which an object or event belongs. C4.5 / C5.0 decision trees are non-parametric learning techniques that use a recursive procedure to progressively partition the training data into groups according to a partition rule that maximizes the homogeneity of the target feature in each of the obtained groups. In our case, the class is the target feature that measures academic performance of students (e.g. a cutoff of a C grade or lower defines poor academic performance; a grade above C defines good academic performance. Decision trees have proven to be extremely robust when dealing with data of varying quality (e.g., missing values), and are quite valuable at describing the predictor features and their value ranges that specify a given class value. This makes the algorithm good at both predicting and describing the nature of the prediction (i.e. the prediction is not the result of a black box).

Support Vector Machines: Generally referred to as SVMs, these are powerful discriminative models initially proposed by Vapnik [19], based on the idea of classifying data in categories by finding an optimal decision boundary that is as far away from the data in each of the classes as possible. SVMs use a kernel function to map data to a high-dimensional feature space so that data points can be classified even when the data is not otherwise linearly separable. SVM implementations have been extensively tested and are considered state of the art for their classification accuracy. The predictive accuracy of an SVM classifier is dependent upon its configuration (choice of kernel function and the kernel's parameters). Several configurations are built and tested to choose an SVM model with the highest predictive performance.

Bayesian Networks (BNs): The set of factors that lead to academic success or failure typically cover a wide spectrum of issues. With the purpose of being able to analyze the dependency among these variables and their mutual interplay from an integrated perspective, we resort to BNs. These are graphical models based on the notion of conditional independence that encode the joint probability distribution of a set of variables in a compact manner using a directed graph to describe the probabilistic dependencies among variables. The directed graph in the Bayesian network provides the overall dependency structure among the variables, and the conditional probability distribution at each node quantifies the directed dependencies. A key feature of BNs is that they enable the user to model and reason about uncertainty BNs can also be interpreted causally: although the notion of causality is slippery and open to interpretation, Lauritzen [24] points out that graphical models, in particular those based on directed graphs, have natural causal interpretations and thus form a language in which causal concepts can be discussed and analyzed in precise terms. For more details on graphical models, BNs and causality see [25].

BNs can be learnt from data (both the graph topology and the conditional probability parameters), adding some constraints based on causal interpretation (e.g. precedence of variables) to limit the search space. The learning process identifies the BN that fits the data best. We apply a heuristic search algorithm that uses a scoring metric to search for optimal networks in the space of BNs, identify several candidate networks, and estimate their associated parameters. Various search strategies will be considered including greedy strategies (K2) and the simulated annealing algorithm, with several local score metrics (BDe, MDL), as implemented in Weka[10].

3.5 Phase Six (Evaluate and Choose Models)

In Phase Six (Evaluate and Choose Models), we validate the models learnt in Phase Five using the validation data set. Base on these evaluations we fine tune the models and its configurations, selecting a small set of candidate models. These candidate models are subsequently tested using the test data set to gauge a realistic measure of their predictive power.

[10] Weka is an open source machine learning Java library, part of the Pentaho suite, and one of the software tools of choice in this project. See http://weka.sourceforge.net/manuals/weka.bn.pdf for more details.

		Predicted			
		$Class_1$	$Class_2$	...	$Class_k$
Actual	$Class_1$	n_{11}	n_{12}	...	n_{1k}
	$Class_2$	n_{21}	n_{22}	...	n_{2k}
	...	...	...	...	...
	$Class_k$	n_{k1}	n_{k2}	...	n_{kk}

Figure 3. Confusion Matrix

Predictive accuracy is by far the most popular evaluation method to assess predictive models, including Bayesian networks. In practice, models are applied on the validation data set where all features (predictors and targets) are known, and predictions are computed for each of the instances in the data set. Those results are then compared to the target values in the data set and measures of predictive accuracy are computed. When the target variable is discrete (i.e. a classification task,), most accuracy measures are derived from the confusion matrix, which summarizes the number of correct and incorrect predictions made by the classifier on the validation (or test) data set (see Figure 3). For example the sum of the values along the main diagonal divided by the sum of all values represents the proportion of correct predictions, a classical measure of predictive accuracy. The off-diagonal cells represent counts of misclassification. As most of the models that we develop deal with classification, we will use these accuracy metrics to measure predictive performance. Cost-sensitive evaluation methods can also be applied so long as the costs associated with misclassification and the benefits of correct classification are available.

In the case of Bayesian networks, a measure of predictive accuracy is computed for each node by predicting the cases of each variable in the validation (or test) data set given the values of the variables in their Markov blanket[11]. Then the predictive accuracy measures are averaged to obtain an overall predictive accuracy of the Bayesian network's structure.

4. Data Quality Issues

Data mining and predictive modeling are affected by input data of diverse quality. This means that the overall performance of a data mining technique is tied to the quality of data available to develop data mining models. Data quality can be analyzed along several dimensions: completeness, accuracy, consistency, integrity among others. Complete data may, in fact, be of poor quality if it is inaccurate (e.g. the data objects do not accurately represent the values they are expected to model). Data mining has as a prerequisite a data quality enhancement activity, but Lauría & Tayi [26] contend that determining which data quality dimension should be improved, and to what extent, is not straightforward. It requires a thorough understanding of the domain of application, the nature of the input data, and the characteristics of the proposed data mining technique.

[11] The Markov blanket of a node in a BN is composed of the node's parents, the node's children and the parents of the node's childrenthe set the nodes

When considering issues of data quality two primary concerns must be addressed. First, we will need to review the data, particularly the event log data from Sakai/iLearn, to ensure that technical problems did not result in erroneous data being collected or, possibly, that no data was collected at all. Staff in our Information Technology offices will conduct a data integrity review of the event log data and will make note of any data elements which appear to be corrupted or missing. These data will then be reviewed prior to the analysis work and removed if concerns exist.

Second, we will need to make sure that we include course tool data coming only from courses in which a minimum level of tool usage has been achieved. For example, if in a given course the Assignment tool was only used by 2 out of 30 students it would likely indicate that this tool was not central to the level of effort students were dedicating to the course work. Following the same guidelines as Campbell [7] used in his research, for a course tool to be counted at least 50% of the students in the course will have had to use the tool at least once.

As mentioned in section 3.1, variability among courses in terms of assessment and student activity (as demanded by the instructor) is an issue of major concern, as it could inaccurately depict differences in behavior among students of different courses that should otherwise reflect similar behavior. This has the effect of increasing the variability among data samples (by increasing the variability of course management related features) and therefore reduce the relative sample size of the data use to train the models. For example, a Bayesian learner computes the relative frequencies of a predictor with respect to the target (class) value to estimate the model parameters (conditional probabilities of the predictors given the class). If the variability of the data samples is large, even with large amounts of data, the variance of the estimates might be high, with detrimental consequences on the accuracy of the model. In order to avoid these issues we have decided to replace absolute values related to course management events with ratios and proportions.

5. Conclusion

This paper reports the initial stages of the Open Academic Analytics Initiative and provides a detailed description of the methodology to be used to develop predictive models in academic analytics This research derives its motivation from the need of introducing alternative research methods and model development approaches capable of developing tools that can be used in practical settings to predict academic performance and carry out early detection of students at risk .The methodology presented in this research will be initially applied on real-world data extracted from Marist College transactional systems: its open source course management system (Sakai / ilearn) and student demographics and course enrollment data. We hope that this methodological framework is used by other higher education institutions as a template to facilitate development of predictive models for academic success using Sakai data and that the results by the Open Academic Analytics Initiative will demonstrate a reference implementation of the learning analytics algorithms and reports used to generate educational interventions for participating, at-risk students.

Appendix A. Description of the Data set

Source	Feature	Description	Data Type
Sakai	Avg Site Visits per week	The total number of times per week the student enters a course	Continuous
Sakai	Percent Lesson Content Accessed	The total number of times a section in the Lessons tool is accessed by a student / The total number of times a section in the Lessons tool is accessed in the course	continuous
Sakai	Percent Discussion Postings	The total number of discussion postings by student / total number of discussion postings in the course	continuous
Sakai	Percent Discussion Postings read	The total number of discussion postings opened by student / total number of discussion postings opened in the course	continuous
Sakai	Percent Assessments completed	The number of assessments completed by the student / The number of assessments completed by all students in the course	continuous, interval
Sakai	Percent Assessments opened	The total number of assessments opened by the student./ the total number of assessments open by all students in the course. Note: If a student opens the same assessment multiple times, the system records each entry.	continuous, interval
Sakai	Percent Assignments completed	The number of assignments completed by the student / The number of assignments completed by all students in the course	continuous
Sakai	Percent Assignments opened	The total number of assignments opened by the student./ the total number of assignments open by all students in the course. Note: If a student opens the same assignments multiple times, the system records each entry.	continuous
Banner	Subject	The Dept from which the course is offered.	Discrete, nominal
Banner	Course	The course identification	discrete, nominal
Banner	Course size	The number of students in the course/ section	discrete, ordinal
Banner	Course length	The length of the course, measured in weeks	discrete
Banner	Course Grade	The final course grade of the student. Entries are A, B, C, D,F, I, or W. If the student drops the course within the official drop/add window, the course grade field will be null.	discrete, ordinal
Banner	Course completion	Course completion was defined as students completing the course within the normal semester timeframe. In other words, students who did not withdraw or receive an incomplete	discrete, nominal
Banner	High School Rank	The high school rank as expressed as a percentile.	continuous
Banner	SAT Verbal Score	The numeric SAT verbal score.	continuous
Banner	SAT Math Score	The numeric SAT mathematics score.	continuous
Banner	ACT Composite Score	The ACT composite score.	continuous
Banner	Aptitude score	Defined as the SAT composite score or the converted ACT to SAT score. In the cases in which students have both SAT and ACT scores, the SAT score will remain	continuous

Banner	SAT Composite score	Defined as the sum of the SAT verbal and SAT math scores.	continuous
Banner	Birth Date	The birth date of the student	continuous
Banner	Age	Converted from the birth date, expressed in years.	continuous
Banner	Race	The race of the student (self-reported)	discrete, nominal
Banner	Gender	The gender of the student (self-reported).	discrete, nominal
Banner	Full-time or Part-time Status	Code for full-time or part-time student based on the number of credit hours currently enrolled.	discrete, nominal
Banner	Class Code	The current academic standing of the student as expressed by the number of semesters of completed coursework. Ranges from one to eight for undergraduate students. One (1) indicates a first semester freshman. Four (4) would indicate a second semester sophomore.	discrete, ordinal
Banner	Cumulative GPA	Cumulative university grade point average (four point scale).	Continuous
Banner	Semester GPA	Semester university grade point average (four point scale).	Continuous
Banner	University Standing	Current university standing such as probation, dean's list, or semester honors.	discrete, nominal
Banner	Academic success	Defined as students completing the course within the normal timeframe and receiving a grade of C or better.	Discrete 1= success 0 = failure

References

1. U.S. Department of Education, N.C.f.E.S. *2001-02 to 2007-08 Integrated Postsecondary Education Data System, Fall 2001, and Spring 2002 through Spring 2008*. June 2009 [cited 2/15/2011]; Available from: http://nces.ed.gov/programs/digest/d09/tables/dt09_331.asp
2. *The College Completion Agenda 2010 Progress Report*, in *College Board Advocacy & Policy Center*. 2010.
3. Baepler, P. and C.J. Murdoch, *Academic Analytics and Data Mining in Higher Education.* International Journal for the Scholarship of Teaching and Learning 2010. 4(2).
4. Goldstein, P.J., *Academic Analytics: The Uses of Management Information and Technology in Higher Education*. December 2005, EDUCAUSE Center for Applied Research.
5. Johnson, L., et al., *The 2011 Horizon Report*. 2011, The New Media Consortium: Austin, Texas.
6. Pistilli, M.D. and K.E. Arnold, *Purdue Signals: Mining Real-time Academic Data to Enhance Student Success.* About Campus. , July-August 2010: p. 22-24.
7. Campbell, J.P., *Utilizing student data within the course management system to determine undergraduate student academic success: An exploratory study*, in *Educational Studies*. 2007, Doctoral Dissertation, Purdue University. p. 219.

8. Romero, C., S. Ventura, and E. Garcia, *Data mining in course management systems: Moodle case study and tutorial.* Comput. Educ., 2008. 51(1): p. 368-384.
9. Talavera, L. and E. Gaudioso. *Mining student data to characterize similar behavior groups in unstructured collaboration spaces*. in *Workshop on AI in CSCL*. 2004.
10. Laurie, P.D. and E. Timothy, *Using data mining as a strategy for assessing asynchronous discussion forums.* Comput. Educ., 2005. 45(1): p. 141-160.
11. Arnold, K.E. (2010) *Signals: Applying academic analytics*. Educause Quarterly Volume,
12. Campbell, J.P., *Opening the Door to New Possibilities Through the Use of Analytics*, in *EDUCAUSE Learning Initiative 2011 Annual Meeting*. 2011: Washington, DC.
13. Shmueli, G. and O. Koppius, *Predictive Analytics in Information Systems Research.* MIS Quarterly, (forthcoming).
14. Lonn, S. *Sakai Data Analysis Working Group*. n.d. [cited 4/28/2011]; Available from: https://confluence.sakaiproject.org/display/UDAT/Log+Events+Descriptions+-+Sakai+2.6.x.
15. Bellman, R., *Adaptive control processes: a guided tour*. 1961, Princeton, N.J.,: Princeton University Press. 255 p.
16. Guyon, I. and A. Elisseeff, *An Introduction to Variable and Feature Selection.* Journal of Machine Learning Research, 2003. 3: p. 1157-1182.
17. Kohavi, R. and G. John, *Wrappers for feature selection.* Artificial Intelligence, December 1997. 97(1-2): p. 273–324.
18. Quinlan, J.R., *C4.5 : programs for machine learning*. The Morgan Kaufmann series in machine learning. 1993, San Mateo, Calif.: Morgan Kaufmann Publishers. x, 302 p.
19. Vapnik, V.N., *The nature of statistical learning theory*. 2nd ed. Statistics for engineering and information science. 2000, New York: Springer. xix, 314 p.
20. Schölkopf, B., A. Smola, and K.-R. Müller, *Nonlinear Component Analysis as a Kernel Eigenvalue Problem.* Neural Computation, 1998. 10(5): p. 1299-1319.
21. Roweis, S.T. and L.K. Saul, *Nonlinear dimensionality reduction by locally linear embedding.* Science, 2000. 290: p. 2323--2326.
22. Shmueli, G., N.R. Patel, and P.C. Bruce, *Data mining for business intelligence : concepts, techniques, and applications in Microsoft Office Excel with XLMiner*. 2nd ed. 2010, Hoboken, N.J.: Wiley.
23. Delmater, R. and M. Hancock, *Data mining explained : a manager's guide to customer-centric business intelligence*. 2001, Boston: Digital Press. xix, 392 p.
24. Lauritzen, S., *Causal inference from graphical models*, in *In Complex Stochastic Systems*. 2001, Chapman and Hall/CRC Press.
25. Pearl, J., *Causality : models, reasoning, and inference*. 2000, Cambridge, U.K. ; New York: Cambridge University Press. xvi, 384 p.
26. Lauría, E.J.M. and G.K. Tayi, *Statistical machine learning for network intrusion detection: a data quality perspective.* International Journal of Services Sciences, 2008. 1(2): p. 179-195.