USING CLASSIFICATION AND CLUSTERING TO PREDICT AND UNDERSTAND STUDENT BEHAVIOR IN AN INNOVATION-BASED LEARNING COURSE

A Thesis
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Lauren Nichole Singelmann

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Department:
Electrical and Computer Engineering

June 2020

Fargo, North Dakota

NORTH DAKOTA STATE UNIVERSITY

Graduate School

Title

USING CLASSIFICATION AND CLUSTERING TO PREDICT AND UNDERSTAND STUDENT BEHAVIOR IN AN INNOVATION-BASED LEARNING COURSE

By

Lauren Nichole Singelmann

The supervisory committee certifies that this thesis complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Daniel Ewert
Chair

Benjamin Braaten

Warren Christensen

Approved:

July 15,2020
Date

Benjamin Braaten
Department Chair

ABSTRACT

One of the Grand Challenges for Engineering is advancing personalized learning, but challenges remain to identify and understand potential student pathways. This is especially difficult in complex, open-ended learning environments such as innovation-based learning courses. Student data from an iteration of an innovation-based learning course were analyzed using two educational data mining techniques: classification and clustering. Classification was used to predict student success in the course by creating a model that was both interpretable and robust (accuracy over 0.8 and ROC AUC of over 0.95). Clustering grouped student behavior into four main categories: Innovators, Learners, Surveyors, and Surface Level. Furthermore, noteworthy variables from each model were extracted to discover what factors were most likely to lead to course success. The work presented contributes to gaining a better understanding of how engineering students innovate and brings us closer to solving the Grand Challenge of advancing personalized learning.

ACKNOWLEDGEMENTS

I would like to extend many thanks to all of my committee members. Dr. Dan Ewert has had faith in me since the very beginning of my engineering career. He has instilled in me a love for working on tough problems with good people, and I believe that is what engineering is truly about. Dr. Ben Braaten has provided me with opportunities and support that has allowed me to flourish. Dr. Warren Christensen has helped me grow in my understanding of discipline-based education research and has given me the confidence to to dive into this area.

Thanks also goes out to the Electrical and Computer Engineering Department and the College of Engineering for their administrative and financial support which has allowed me to travel around the world to learn and share my research. These experiences have helped me grow professionally and personally.

In addition, I would like to thank my lab mates: Enrique Alvarez, Ellen Swartz, Mary Pearson, and Ryan Striker. Your feedback and support have been a huge help as I have been working on this thesis. I'd especially like to thank Enrique for all of his work on the learning objective logging portal. Without you, this work would not have been possible.

I would also like to thank Scott Pryor, my supervisor and mentor. I have greatly valued our discussions about education, engineering, and life, and the ideas discussed have helped guide the direction of this work.

Thank you also to some of the amazing women that have provided me with inspiration and support including Nancy Rossland and Olivia Gravel.

Next, I would like to acknowledge the Engineering Squad, including Dani, Paul, Ian, Annie, Phil, and John. This wouldn't be an engineering paper if I didn't mention you all. You have been a constant during my entire engineering education. Thank you all for the memories, support, and love.

Finally, I'd like to thank my family for their love and support, and Ben for encouraging me to stay true to myself. You all have been there to listen to my struggles, but also celebrate in successes both big and small.

DEDICATION

To all future engineers. I hope you find your path and stay true to yourself.

PREFACE

"Scientists study the world as it is; engineers create the world that has never been."

- Theodore von Karman

For the past few years I have spent endless hours teaching thousands of Kindergarten-12th Grade students what engineering is. When I started, I often struggled with developing a definition of engineering because engineers can do just about anything. However, I often found myself coming back to von Karman's idea of creation. Rather than studying facts and figures, engineers create and innovate.

In order to help students better understand this idea of creation and how to "think like an engineer," I often present them with the Engineering Design Process, a seemingly basic list of actions that explains how engineers solve problems. However, the more and more I spend time with this tool, the more and more it frustrates me. Surely no one goes from step to step as smoothly and easily as the perfectly spaced diagram leads us to believe. The recipe for innovation must go deeper than that, and this work is a first step in better understanding that.

How do engineers create? What leads to a successful idea? How we can empower future engineers to create a world we want to live in? Engineering is not a linear process, but rather a complex, awe-inspiring machine involving people and processes working together to achieve great things. This work strives to understand that complex, awe-inspiring machine and to illustrate the complexities of the true Engineering Design Process.

So, come along for the journey. If you are here for the educational piece, I hope you learn something about engineering and data mining. If you're here for the engineering, I hope you learn something about pedagogy and education. By joining both of these areas, I truly believe we can better prepare the next generation of engineers, the ones that can create the world that has never been.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

EDM	Educational Data Mining
IBL	Innovation-Based Learning
KDD	Knowledge Discovery in Databases
SQL	Structured Query Language
ROC	Receiver Operating Characteristic
AUC	Area Under Curve
ECG	Electrocardiogram
DSK	Discipline-Specific Knowledge
SVM	Support Vector Machine
LR	Logistic Regression
KNN	K-Nearest Neighbors
MC	Majority Class
BME	Biomedical Engineering
DK	Depth of Knowledge

LIST OF APPENDIX FIGURES

Figure Page

1. INTRODUCTION

1.1. A World Full of Grand Challenges

Since the beginning of our civilization, engineers have been the drivers in making technical advancements such as the steam engine, electricity, the automobile, and the internet. So what comes next? The National Academies for Engineering have created a list of fourteen Grand Challenges that a team of top scientists and engineers have determined to be the most important problems to solve within the next century [1]. These challenges are shown in Figure 1.1.

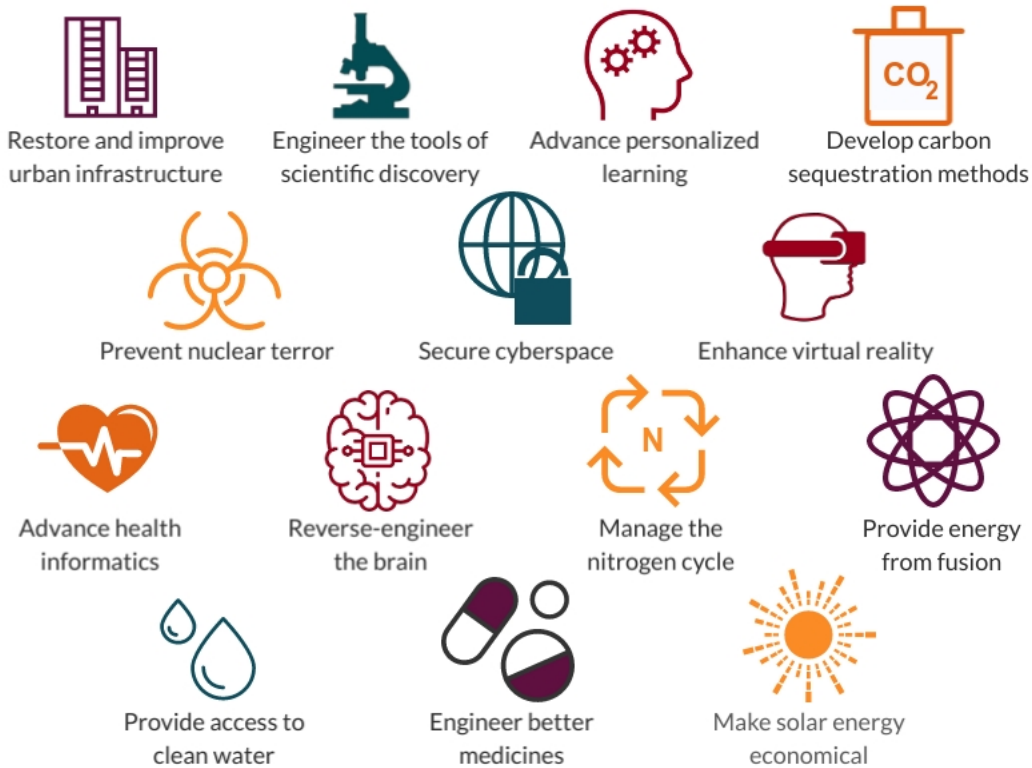

Figure 1.1. Grand Challenges for Engineering [1]

National organizations and experts in engineering and the sciences have agreed that in order to solve the Grand Challenges and other complex problems, engineers need to have skills beyond their technical expertise. When exploring the desired student outcomes from ABET (the accrediting agency for engineering programs) [2], the portrait of the Engineer of 2020 painted by the National Academy of Engineering [3], and a metastudy about what engineering employers want from recent graduates [4], five consistent themes emerge: 1) an ability to solve complex problems, 2) communication skills, 3) teamwork, especially on interdisciplinary teams, 4) entrepreneurship and business intelligence, and 5) the ability to be a lifelong learner.

1.2. An Educational Model to Promote Innovation

In order to give students an opportunity to develop these skills, an upper level Cardiovascular Engineering course was revamped using the Innovation-Based Learning (IBL) model [5]. Rather than being assessed on homework, tests, and quizzes, these students were assessed on their ability to apply what they were learning in class to an innovation project. Success in the course is defined by achieving external value, which involves making an impact outside of the classroom and getting some sort of outside review (e.g. presenting a peer-reviewed poster at a conference, competing in a business plan competition, or submitting and getting a review on an invention disclosure). Students work on their problem solving abilities by identifying a project with their group, communication by presenting about their work both in and outside of class, teamwork by working on teams (many of which have students from multiple majors or programs), entrepreneurship by exploring ways to create value with their project, and lifelong learning skills by creating their own pathway through the course. Many students have found success in this model [5], but questions still remain about what made these students successful and how this can be transferred to other courses and experiences. When looking at a dynamic system involving students, instructors, projects, successes, and failures, typical IF-THEN relationships aren't able to paint a full picture of the inner workings of the course [6]. Therefore, just as the students in this course aim to tackle complex problems, the course itself must be explored as a complex problem [7].

1.3. Understanding Complexity and its Implications

The idea of complexity can be better understood in the context of the Cynefin Framework, shown in Figure 1.2. *Complex* systems are defined as a collection of elements that interact in a dynamic way. The interactions are often elaborately interconnected and nonlinear. Each element in the system has its own behavior, and changes in the behavior of one part change the entire system. The system behaves in conditions far from equilibrium, meaning it is always fluctuating and responding to changes within its parts. Even throughout all the changes, emergent patterns are generated, accomplishing some function [8]. Many of the Grand Challenges listed in Figure 1.1

are examples of *complex* systems. From the network of deeply interconnected neurons within the brain [9], to the hierarchical biological interactions that contribute to engineering better medicines [10], to the effects of agriculture and human behavior on the nitrogen cycle [11], these problems require analysis at the system level. Learning is a complex process as well; students, instructors, groups, classes, and universities, and external stakeholders are all interacting in unpredictable and tightly interwoven ways [12].

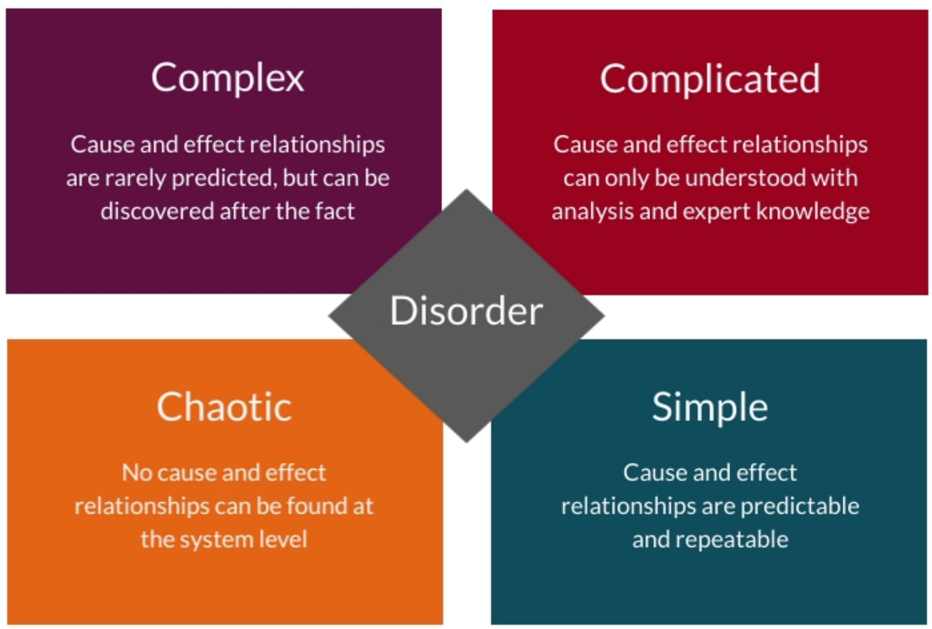

Figure 1.2. The Cynefin Framework is broken up into four domains: simple, complicated, complex, and chaotic. The simple domain consists of cause-and-effect relationships which are well understood. The complicated domain also has consistent relationships, but understanding them requires expert understanding. Relationships in the complex domain are interwoven and dynamic; modeling alone cannot predict future relationships with certainty, but relationships can be discovered. Finally, the chaotic domain does not have relationships that can be modeled or discovered [98]

However, educators and researchers often map learning as a *complicated* process rather than a *complex* process. In circuits courses, for example, learning is usually mapped in a linear way; if you follow the steps, you'll reach the right answer. To solve a complicated problem, first you need to be able to understand electric quantities like voltage and current, and that allows you to understand basic circuit elements like resistors and capacitors. When you understand how these elements work alone, then you can start to learn about how they work together in parallel and series.

These concepts lead into learning about voltage nodes and current loops, and those concepts build into being able to understand the circuit. There are multiple steps to the process, but it is mapped in a linear and straightforward way.

We even often simplify engineering down to a *complicated* process. The Engineering Design Process tells us that to solve an engineering problem, first we ask a question, then we research more about the problem and what solutions are available, we brainstorm solutions, we make a plan, we create a prototype, we test it out, and then we continue to improve it until we have a solution [13].

Modeling learning and engineering as a *complicated* process might work in the most basic of cases, but it fails when we start to look at more complex problems such as what happens when students guide their own learning or when engineers are working on developing a new and innovative product or solution. In these cases, the learning process transfers from being mapped as a *complicated* problem to a *complex* problem. Work still needs to be done to better map and understand these processes holistically. Models of complex systems need to incorporate experiment and observation of the real world, mathematical modeling and analysis, and underlying theory [14]. By understanding the art of learning and innovation as a *complex* system, we can improve our ability to prepare our students to meet the aspirations of the Engineer of 2020 and solve the problems our ever-changing world is facing.

1.4. Outline

This work explores how data mining techniques can be used to discover information about open-ended learning, specifically on data collected about students' learning process when working on innovation projects in a cardiovascular engineering course. It will look specifically at two machine learning methods: classification and clustering. Classification is a machine learning method used to predict which class an item will belong to. In our case, we will use classification to train a model to predict student performance. Success was defined as creating at least one deliverable with high external value (e.g. presenting at a conference, competing in a business competition, submitting an invention disclosure, or publishing a paper). Clustering is a machine learning method used to group similar items together (in this case, students). The biggest difference between classification and clustering is that classification is a supervised machine learning method, meaning we tell the algorithm which students fell into each group. Clustering, on the other hand, is an unsupervised machine learning method, meaning there are no pre-existing labels.

We have two main research questions, each with three subquestions:

1. Can classification models predict how students will perform in the course?
   (a) Which classification algorithms and feature sets have the best performance when differentiating between top performing students and lower performing students?
   (b) At what point in the semester can the classification algorithms predict student trajectory with sufficient accuracy?
   (c) Which features are most likely to differentiate top performing students and lower performing students?
2. Can clustering models tell us more about how students approach the course?
   (a) How does the unsupervised clustering model group students?
   (b) Do students change clusters throughout the course? If so, how?
   (c) What variables are most strongly weighted in forming these clusters?

Chapter 2 outlines the importance of knowledge discovery and data mining, specifically in the context of *complex* problems. Chapter 3 introduces educational data mining, its history, and the most common algorithms and methods. Chapter 4 explores the literature on using educational data mining in more complex contexts, specifically, open-ended learning environments. Chapter 5 details the data mining process, including choosing the data and running appropriate tasks and algorithms; excerpts from Chapter 5 were adapted from a research paper titled "Design and Development of a Machine Learning Tool for an Innovation-Based Learning MOOC," published at *IEEE LWMOOCs 2019* and was co-authored by Lauren Singelmann, Ellen Swartz, Mary Pearson, Ryan Striker, and Enrique Alvarez Vazquez. Chapter 6 explores the format of the IBL course and explores the student population; this chapter is adapted from a research paper titled "Student-Developed Learning Objectives: A Form of Assessment to Promote Professional Growth," accepted for publication at the *American Society for Engineering Education 2020 conference* and was co-authored by Lauren Singelmann, Enrique Alvarez, Ellen Swartz, Mary Pearson, and Ryan Striker. Chapter 7 and Chapter 8 explore the results and evaluate the use of classification and clustering, respectively. Chapter 7 was adapted from a research paper titled "Predicting and Understanding

Success in an Innovation-Based Learning Course" accepted for publication at the *Educational Data Mining 2020 conference* and was co-authored by Lauren Singelmann, Enrique Alvarez, Ellen Swartz, Ryan Striker, Mary Pearson, and Dan Ewert. Chapter 8 was adapted from a research paper titled "Innovators, Learners, and Surveyors: Clustering Students in an Innovation-Based Learning Course," accepted for publication at the *IEEE Frontiers in Education 2020 conference* and was co-authored by Lauren Singelmann, Enrique Alvarez, Ellen Swartz, Ryan Striker, Mary Pearson and Dan Ewert. Chapter 9 discusses the meaning of the results and how they can drive educational reform. Chapter 10 summarizes the work and suggests future directions both short and long term.

2. KNOWLEDGE DISCOVERY, DATA MINING, AND MACHINE LEARNING

2.1. Introduction

In the age of information, data is more readily available than ever, but analyzing this data presents its own set of challenges. Machine learning is the use of algorithms to complete a specific task without being explicitly programmed. One of the main applications of machine learning is data mining, or using algorithms to find patterns within data [15]. The result of this process is known as knowledge discovery in databases. These emerging patterns should be useful and relevant, and they are usually not able to be found by a human analyzing the data [16]. Machine learning, data mining, and knowledge discovery in databases are becoming prominent in areas ranging from the stock market to online marketing and advertising to music and movie recommendations [17]. These algorithms and methods are also being used widely within the engineering field, including to solve some of the Engineering Grand Challenges discussed in Chapter 1. This chapter will explore how machine learning, data mining, and knowledge discovery in databases are being used to solve five of the Engineering Grand Challenges. By understanding how these skills can be used to solve complex problems like the ones presented in the Grand Challenges, we can apply similar strategies to the complex data collected from the IBL course. In addition, this chapter will illustrate how the work presented in the following chapters can contribute to the work aiming to solve the Grand Challenges, specifically in advancing personalized learning.

2.2. Machine Learning and Data Mining to Solve the Engineering Grand Challenges

2.2.1. Advancing Health Informatics

Machine learning and data mining have been growing in the use of multiple areas of medical research, contributing to solving the Grand Challenge of advancing health informatics. Five main applications were identified in [18]:

- Translational bioinformatics - studying molecular processes (e.g. genetics, protein interactions) to better understand how and why organisms behave and respond the way they do
- Medical imaging - using image analysis to make more accurate diagnoses

- Pervasive sensing - the use of multiple kinds of sensors (e.g. wearables, implantable sensors, ambient sensors) to make health recommendations
- Medical informatics - analyzing patient data (e.g. doctor visit records, lab results, medication plans, and immunization records) to better understand potential risk factors and relationships
- Public health - exploring how the health of community members can be effected by the factors in that community (e.g. air and water quality)

There have been significant advances in the use of machine learning and data mining in the health field [18, 19], but this is an excellent example of a complex problem, meaning there are also significant challenges. These challenges can be summarized by the four Vs: volume, velocity, variety, and veracity [20]. Volume refers to the sheer amount of data available, and this amount is growing as more information is being added to medical databases. Velocity refers to the speed at which data is accumulating, but also the urgency required to make timely medical decisions. Variety speaks to the different types of available data whether that be quantitative or qualitative, text or images, or structured or unstructured. Finally, veracity refers to the potential for noisy data, biased data, or abnormalities that could lead to training an algorithm with data that is not representative of the patient population [20]. Researchers are working on ways to account for these issues [18, 20], and their work influences not only the medical community, but those working on using machine learning and data mining to solve any complex problem.

Like most complex problems, health informatics illustrates the importance of using multi-modal data in order to explore problems with multiple lenses and viewpoints [18], ensuring that you are accounting for the possibility of biases and inaccurate data [19], integrating expert domain knowledge into the data mining process [18], and being careful to not make sweeping claims and assumptions [19].

2.2.2. Making Solar Energy Economical

Although renewable energy is on the rise, fossil fuels are still the predominant source for electricity in the world. In order for renewable energy sources such as solar energy to become widely implemented, they need to become more affordable and easier to implement into current infrastructure [1]. Therefore, the next Grand Challenge is making solar energy more economical. Machine learning can be used to help this integration by improving solar radiation forecasting.

Various models have been created to help predict how much solar power will be added to the grid, allowing the grid's infrastructure to adapt accordingly without causing outages [21]. However, like many of the Grand Challenges, solar energy falls into the complex domain because the energy output relies on multiple variables, making predictions extremely difficult [22]. Although simple models have shown some success [23], in order to improve efficiency and make solar energy feasible, these predictions must be improved by accounting for future changes rather than relying only on past data to train the model [22].

Machine learning making solar energy more economical illustrates why it is important to avoid only using past data when working with complex systems. As the climate is changing, it is important to integrate models that account for future changes and consider expert understanding of the problem [22].

2.2.3. Restoring and Improving Urban Infrastructure

Infrastructure around the world needs to be repaired and replaced in order to continue to support new technologies and increasing populations, making this another of the Grand Challenges [1]. Machine learning can either be used as a tool to better map and improve existing infrastructure or can be implemented into the infrastructure itself, helping to optimize the way we live, work, and travel.

Keeping infrastructure safe and up-to-date has always been a challenge, but one issue that is becoming more prevalent is being able to upgrade buried infrastructure like pipes and cables, especially because there are few existing comprehensive maps or plans [1, 24, 25, 26]. Some researchers are using machine learning models to predict which areas are most likely to experience failure by training the models with data from previous failures [24]. Prior to the use of computer-aided vision, inspection and prediction involved experts spending hours looking at camera footage. Now, this analysis can be automated by using data from pipe scanners, ultrasound, laser profiling, and infrared technology. Each of these tools has its strengths and weaknesses, so using models that can analyze this multimodal data can improve accuracy [25]. Some systems combine rules identified by experts in the field with machine learning in order to create hybrid information systems, helping to place some bounds around this complex problem [26].

Some engineers are working to integrate machine learning right into the infrastructure itself, revitalizing our current system. Within transportation, applications include travel route optimization, streamlining finding parking by using sensors and cameras, making smarter streetlights, and collecting more data from cars, buses, and bicycles to better understand and optimize how vehicles travel [27]. Others are implementing machine learning into the construction of buildings and other civil projects in order to analyze these projects as complex adaptive systems. Creation of infrastructure involves many people, resources, and considerations, so this well-rounded approach has started to lead to buildings that are more safe, green, and inexpensive [28].

Once again, the importance of being able to understand and integrate multimodal data is clear when attempting to solve complex problems [25, 27, 28]. In addition, these complex models make use of expert reasoning in order to improve their versatility and flexibility [26, 28].

2.2.4. Securing Cyberspace

As the world becomes more reliant on the cloud for storing data and personal information, the need for securing cyberspace is more important than ever, adding this to the Grand Challenges list. Previously, "firewalls" or "perimeter defense" were integrated as main points of security, blocking potential intruders from being able to access secure data. However, these techniques are becoming older and weaker as hackers get more innovative in bypassing the security [1]. Some engineers are working towards securing cyberspace by using machine learning to identify potential attacks and respond before important information can be breached. There are three main types of machine learning applied to cybersecurity including misuse-based (using information about past attacks to identify attacks that are similar), anomaly-based (using normal activity as a baseline and looking for unusual activity that might signal an attack), and hybrid (a combination of misuse-based and anomaly-based) [29].

The use of machine learning in cybersecurity illustrates the importance of adaptable models that can be retrained quickly and efficiently [29]. In order to make real-time decisions about a problem, the model must be able to take in new information and integrate it into the model. Similar to the problems seen with using only existing data in the solar energy problem, if only old data is used, algorithms will do poorly in identifying new types of attacks [30].

2.2.5. Advancing Personalized Learning

Researchers such as Piaget [31], Kolb [32], and Dewey [33] have all explored and modeled the different ways that people learn, but yet many instructors and institutions still use a one-size-fits-all approach [1]. Engineers are helping improve personalized learning by working to better understand the science of learning, how learners vary, and how data can be collected and used to improve the learning experience [1]. For example, learning analytics and educational data mining aim to collect data from students in order to better understand how students navigate learning environments. This information can then be used to help instructors or make direct recommendations to students. Educational data mining will be discussed more in depth in Chapter 3.

2.3. Conclusion

Across all of these problems, three themes continue to emerge about using machine learning to solve complex problems: 1) using multimodal data to create a well-rounded model, 2) implementing expert domain knowledge to help steer your model, and 3) continuing to adjust and adapt your model over time. Therefore, these themes must be considered when working with educational data as well. Although engineers are far from solving the Grand Challenges, the work that has already been done can help guide analysis of the IBL data. Similarly, careful analysis and understanding of the IBL data can help add to the ever-growing body of work towards solving the Grand Challenges, especially within education.

3. EDUCATIONAL DATA MINING

3.1. Introduction

Educational data mining (EDM) is a field with the goal of "developing methods for exploring the unique and increasingly large-scale data that come from educational settings and using those methods to better understand students, and the settings which they learn in" [34]. EDM emerged as a field in the late-2000s because of the growing popularity of computer-based learning tools and online learning, making data available for researchers to analyze [35, 36]. Applications of EDM as listed in [35] include predicting student performance, making suggestions for instructors, analyzing and visualizing educational data, grouping students, and improving courseware.

3.2. Techniques

There are five main techniques used in EDM including classification, clustering, association analysis, sequential pattern analysis, and process mining. These tasks are depicted in Figure 3.1 [37].

3.2.1. Classification

Classification is placing each object into a category by using its properties [37]. The four main functions of classification in academic data as compiled by [37] are:

- Predicting academic success - This can be predicting success at the university or program level, including who will graduate on time, who will dropout, or who is most likely to need additional support and advising.

- Predicting course outcomes - This prediction task is at the course level, including who is most likely to pass/fail, who might dropout, and who might get what scores on certain assignments.

- Predicting success in the next task - This application uses data from previous tasks in order to predict success on the upcoming tasks.

- Metacognitive skills, habits, and motivation - This can consist of better understanding the ways students learn, their engagement level, and what resources might benefit students most.

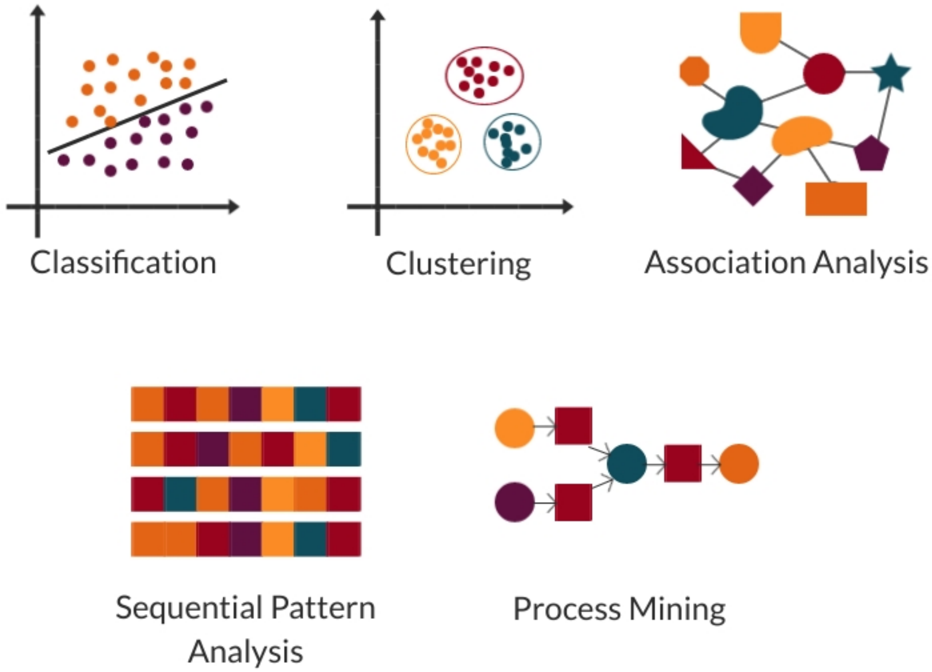

Figure 3.1. Types of tasks commonly performed in educational data mining

3.2.2. Clustering

Clustering is an unsupervised form of EDM, meaning no categories are predetermined. Instead, the algorithms look for similar data points and groups them together into clusters. Uses of clustering in EDM as listed in [38] include:

- Analyzing student motivation, attitude, and behavior - This data can take the form of survey responses, annotations on class materials, or student interviews to better understand what patterns can be found in students' goals or motivations for taking the course
- Understanding learning preferences - Various learning models and theories have been published in order to help better understand how students learn. By clustering students into these groups, teachers or learning platforms can present personalized content and recommendations.
- e-Learning - This application of clustering takes data directly from learning management systems to form activity clusters. The data can include click streams, discussion posts, visit length and frequency, etc.

- Collaborative learning - Clustering can also be used to better understand and evaluate collaborative work by using online activity of members of a group or expert observations, potentially improving group dynamics and helping students grow in their ability to work well in a team.

3.2.3. Association Analysis

Association analysis is a task that finds variables within data sets that have a high affinity for each other. This produces IF-THEN rules that are an easy and straightforward way to understand relationships within the data [37]. Association analysis is one of the most common tasks to be performed on educational data because the rules discovered can usually be easily understood by instructors. Although it does not have the predictive power of classification, it does offer more flexibility. Classification aims to put data points into specific categories, meaning there is usually one variable that is being considered the dependent variable (e.g. predicting pass or fail students). In association analysis, on the other hand, rules can be formed with any variables, therefore finding relationships that may not have been discovered in a classification problem.

[35] lists a variety of tasks that have been conducted using association analysis including:

- Using student behavior to recommend learning materials - By assessing what students know and by looking at what resources similar students found helpful, the algorithm can recommend the most helpful resource for the student.

- Finding relationships that could help an instructor improve a course - For example, maybe if a student completes a certain activity, then they are likely to get a higher score on a certain assessment.

- Finding groups of questions that students are likely to perform similarly on - By exploring what questions have similar response rates, questions can be grouped into categories. An instructor can then look at those categories to see if there are underlying misconceptions that are causing students to answer groups of questions incorrectly.

3.2.4. Sequential Pattern Analysis

Sequential pattern analysis is the exploration of patterns of events. Each sequence contains subsequences made up of items or events. By looking at subsequences that occur most often across sequences, information can be gained about how the order of events effects the outcome for that particular sequence [37]. Sequential pattern analysis differs from the other tasks previously mentioned because it focuses on the temporal component.

Four applications of sequential pattern analysis as reported in [37] are included below:

- Create appropriate learning paths - Sequence mining can help improve the order of activities and lessons in a course to improve student understanding of the content.

- Adapt and customize resource delivery - The creation of learning paths can be taken one step further in order to customize the delivery based on student behavior. The system can assess the student's learning strategies and content knowledge and recommend resources accordingly.

- Identify interaction sequences that are most likely to lead to problems or success - Student pathways can also be used in order to find what user patterns are most likely to differentiate between success or failure of an activity.

- Recommend links and web pages for a student to visit next - Similar to a website that recommends a product based off of other user's shopping activities, sequence pattern analysis can recommend web pages that a user might find helpful based off of what other users have looked at.

3.2.5. Process Mining

Process mining takes sequential pattern analysis one step further; it allows for multiple events happening at once and explores variables associated with each event rather than just the event itself [39]. Whereas sequence mining can be thought of as a discrete signal, process mining can be thought of as a continuous signal.

The three levels of education processes that can be explored using process mining according to [37] include:

- At the university level - This level can explore things such as how students progress through their curriculum, how students change their major, or factors that might effect likeliness of leaving a program or the university.

- At the course level - This level explores how students progress through assignments, what resources they view or return to, or even how they approach exams or other assessments. This is especially helpful in e-Learning and blended learning classes because students can interact with the material at different times and rates.

- At the project level - Depending on the nature of the project, it might also be possible to explore how students work on specific projects. For example, work has been done to track how students work together on programming assignments by exploring the commits submitted by each group member.

3.3. Conclusion

These five methods are the building blocks of most EDM problems, but there is still much to be solved in this field. In 2010, [35] conducted a review of the most influential work in EDM up to that point. The review ended with future work and research lines that the authors believe will be the most important and influential. They include making tools that are easier for educators to use, integrating tools directly into e-learning platforms, standardizing data and models, and tuning traditional data mining algorithms to better fit educational models. Although these future lines of work are older, they still are relevant to any EDM research; it is important to have tools that educators can use, integrate these tools into existing platforms, and ensure that education theory is being considered and implemented when appropriate.

More recently, the 2020 Educational Data Mining Conference put out a call for papers that fit within their topics of interest. The entire list can be found at [40], but the 3 listed below have inspired the direction of the research presented.

- Modeling real-world problem solving in open-ended domains
- Data mining to understand how learners interact in formal and informal educational contexts
- Modeling student and group interaction for collaborative and/or competitive problem-solving

This call for understanding how learners interact, how groups work on problem-solving, and how students navigate in open-ended domains directly inspires the analysis detailed in this thesis. Researchers are just starting to scratch the surface in these areas, but current progress can help direct the analysis of the IBL data.

4. EDUCATIONAL DATA MINING IN OPEN-ENDED LEARNING ENVIRONMENTS

4.1. Introduction

Open-ended learning is a pedagogical approach that leverages students' intrinsic motivation to learn. Students are encouraged to develop their own approach to learning, and there are multiple ways to solve a problem or learn about a concept [41]. Open-ended learning requires students to work on authentic problems and practice metacognition. Benefits of open-ended learning include increased motivation, improved student engagement, and better retention of information [42]. IBL is an example of an open-ended learning environment because students are expected to determine their own way to meet course objectives by tying cardiovascular engineering concepts to a project of their group's choosing.

In addition, we draw from research from a closely related field to open-ended learning environments: self-regulated learning. Self-regulated learning is the practice of "planning, monitoring, and modifying" cognition [43]. Many open-ended learning environments are designed to help students practice self-regulated learning [44]. For example, IBL students are practicing self-regulated learning because they are writing their own learning objectives, monitoring their progress throughout the semester, and adjusting their plan as needed.

EDM has shown great potential in allowing us to better understand open-ended learning environments and self-regulated learning [45, 46]. This chapter will explore both open-ended learning environments and self-regulated learning in a variety of applications: online learning environments, programming, and project-based learning. Studies that use EDM to explore these applications will be summarized, and we will detail how these studies inspire the exploration and analysis of the IBL data.

4.2. Online Learning Environments

Online learning environments have been gaining popularity over recent years due to better affordability, convenience, and freedom [47]. This added freedom means students can navigate the environments in many ways, and these trajectories can be tracked by the online learning platform.

For example, one group of researchers created a biology tutoring program that allows students to interact with complex material in multiple forms. The students can take quizzes to assess their knowledge, add notes, and make and track progress on goals. The researchers were then able to cluster similar students and then look for action sequences that occurred most frequently in each cluster [48]. Another group of researchers created a learn-by-teaching environment where students teach a virtual "student" about a topic in order to demonstrate their learning. Their actions in the platform could be tracked and compared to their performance in the course [49]. Some preliminary work has also been done to explore student behavior in lab [50] and simulation environments [51].

In contrast to exploring student behavior, other researchers have taken an approach that integrates student attitudes and perceptions by having them take surveys asking about course difficulty, quality, and level of support; [52] used this approach in an SAT math course, and [53] used it in a language tutoring software.

This use of EDM with online learning management systems can guide the development and use of our own learning management system for the IBL course. By tracking student behaviors over time, patterns can be found which lead to better understanding of student learning in the course.

4.3. Programming

Learning how to program is another example of an open-ended learning environment. When working on a program, there are many different solutions and many different ways to get to each of those solutions, making an unbounded number of potential states [54, 55]. Advances in EDM and machine learning make these tools a great resource in better understanding how students learn how to program, if and how we can predict student success in programming, and how to automatically assess and provide feedback to students [54, 55, 56].

A review on the use of EDM on programming data broke studies up into three main categories: students, environments, and programming [56]. In the student category, algorithms are designed to be able to tell what concepts a student knows or if they are at risk of dropping out or not passing the course. The environment category aims to be able to provide feedback to students based off of their work or automatically grade assignments. The programming category works to explore the programming process rather than just the final solution; it looks at if a solution is working to approach a goal, what features make up a successful program, and how errors effect the code [56].

Some of the studies found within the programming field can be adapted to the IBL student data. For example, one group of researchers has attempted to use the programming actions of students to predict if they will need help with the assignment. They also explored at what point in the activity their model had the best performance [54, 57]. This study directly inspired the classification experiments presented in Chapter 7; we explored many of the same classification methods and applied a similar trajectory exploration in order to see at what point in the semester we could tell if a student would be successful.

Similarly, another group of researchers used coding data over time to not only predict student success, but also cluster student behavior into distinct groups. These groups provide more insight into different approaches students can take, rather than just sorting students into binary successful/unsuccessful groups [55]. This study showcased the potential benefits of using clustering as performed in Chapter 8 of this thesis.

Another group of researchers has developed a platform that can observe how students are working on a program and make suggestions that are personalized to the progress they have made so far [58]. This sort of work could inspire future work within the IBL data; if pathways emerge from the data, hints can be generated for students that help them reach success while still aligning with their current learning goals.

4.4. Project-Based Learning

Project-based learning is another example of an open-ended learning environment that is being explored using EDM. Project-based learning is a form of constructionism, or a learning theory that focuses on students creating an artifact in order to better cement learning [59, 60]. Some researchers have asserted that EDM is the most appropriate tool to study constructionism because it allows for multiple correct solutions and strategies, especially when using unsupervised methods such as clustering [61].

One application of EDM to explore contructionism and project-based learning is within computer-aided drafting projects. In order to better understand how students use different experimentation strategies, researchers have used log data from a computer-aided drafting environment as students worked on designing an energy-efficient house [62, 63]. The use of log data gives an accurate temporal representation of student learning [64].

Similarly, EDM can be used to explore students working on robotics projects. Some researchers have tracked body language, how students are positioned in relation to each other, noise level, and types of code blocks in their Adruino code [65, 66]. Other researchers have looked at the type and order of actions completed (e.g. adding a move block, attaching a sensor, etc.) [67].

EDM is not widely used in project-based learning yet, but researchers have argued that there is great potential in using EDM to better understand how students work on open-ended problems [61], and some preliminary work has shown to be successful [62, 63, 65, 66, 67].

4.5. Conclusion

The use of EDM in online learning environments, programming, and project-based learning show that there is great potential in using EDM to study the IBL data. Classification can be used to help identify factors that lead to success in the learning environment, and clustering can be used to identify student pathways beyond a binary successful/unsuccessful classification.

5. METHODS

5.1. Introduction

In order to discover patterns and information from the data, we must identify and preprocess our dataset, choose appropriate algorithms, and decide appropriate evaluation metrics. This chapter discusses the methods by detailing each step in the Knowledge Discovery in Databases (KDD) Process as seen in Figure 5.1 [16]. This chapter is meant to give greater detail about how various metrics are calculated, the theory behind various steps, and how the tasks and algorithms were chosen.

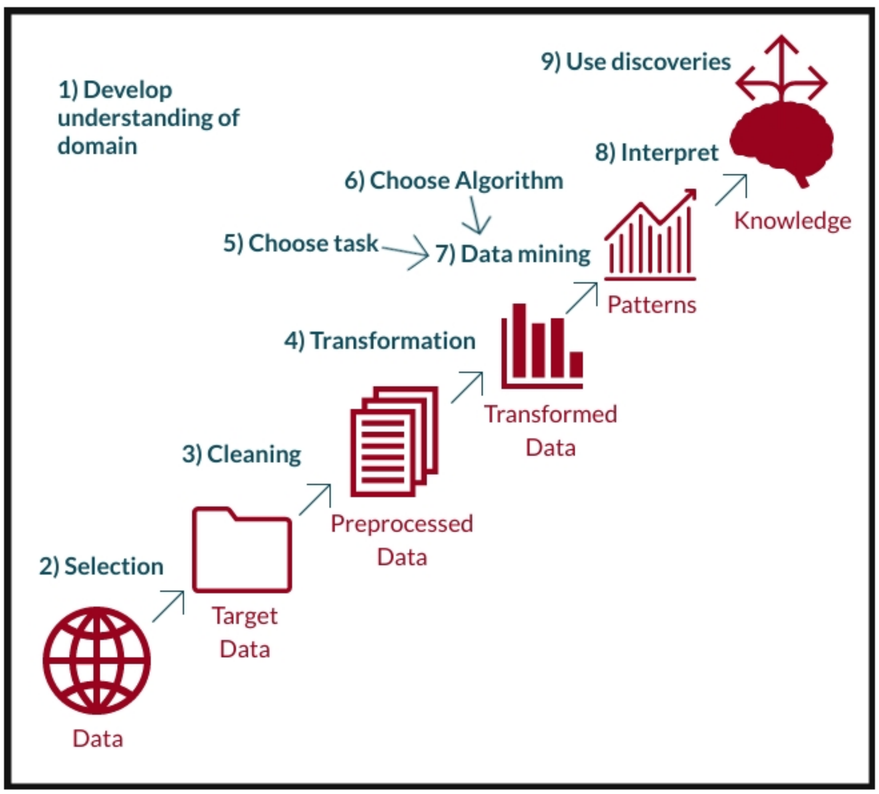

Figure 5.1. The process for knowledge discovery and data mining [16]

Some material in this chapter was drawn directly from [68], a publication co-authored by Lauren Singelmann, Ellen Swartz, Mary Pearson, Ryan Striker, and Enrique Alvarez Vazquez. Lauren Singelmann had primary responsibility for developing the data structure used here. Lauren Singelmann also drafted and revised all versions of this chapter. Other authors served as reviewers of the content.

5.2. The KDD Process

5.2.1. Developing an Understanding of the Application Domain

This step consists of researching the problem at hand as well as any prior knowledge in order to determine appropriate goals. Learning more about the application domain is where researchers should start, but should also be an important component throughout the entire KDD process. In addition to using data mining and machine learning on our data, various other approaches were taken to get a full picture of the course. The pedagogy behind the work was explored, the learning objectives were analyzed by the research team, and survey data from the students was collected and analyzed. The work from this step makes up Chapter 6.

5.2.2. Selecting and Creating a Data Set

Creating the data set requires the researcher to investigate what data is already available, decide what other data is needed, and determine an appropriate way to collect that data and get it into one data set. We chose to start with log data because it is shown to be more accurate than relying on student's reporting of their learning process and metacognition [64].

5.2.2.1. Feature Selection

Features were collected at the following levels: 1) Class, 2) Student, 3) Learning Objective, and 4) Deliverables.

- Class - The highest level that data was collected is the class or cohort. Each semester the class is held, the class and online learning analytics system change. Therefore, it is important to have this information tied to each iteration of the class. This allows the system to group students that were taking the class at the same time, students that had the same instructor, or students that had the same in-class meeting location.

- Student - The second level of data that was collected is at the student level. Information was collected before and during the course to help the system understand the student's background and how they progress through the course. Demographic information was collected so we could explore how specific groups progress through the open-ended course.

- Learning objective - Each student was able to log multiple learning objectives within the system. Because these learning objectives changed throughout the course, all changes were recorded. Students could type in their titles and descriptions, but they also sorted learning objectives into predefined categories to help group similar learning objectives together. In addition, learning objectives were also categorized using Bloom's Revised Taxonomy [69]. Low-level learning objectives might consist of tasks like memorizing the structures and functions of the heart, while high-level learning objectives might consist of tasks like creating a new component for a pacemaker. This classification process helps them determine appropriate ways to assess and document their learning.

- Deliverable - The last level at which data was collected is the milestone/deliverable level. Students divided all learning objectives into various deliverables. For example, to accomplish the learning objective of presenting at a conference, the student might need to submit an abstract, receive feedback, create a poster, and present the poster. Each of these is a deliverable that will be monitored within the system. In addition, the students were able to mark off when each deliverable was completed, creating a virtual timeline of the student's learning. Because the deletion of a deliverable can be just as telling as the creation of one, all records were tracked in real time.

5.2.2.2. Feature Collection

All features were collected through an online learning management system for students in the course. Students log in with their unique username and password to ensure that each student sees personalized content. They then can add, edit, or delete learning objectives and deliverables.

5.2.3. Preprocessing and Cleaning

Preprocessing consists of dealing with missing data, noise, and outliers. The preprocessing step can be as simple as removing incomplete data or as complicated as using statistical methods to try to predict values for a missing data attribute.

Every time a student added or adjusted a learning objective, a new entry was added to a Structured Query Language (SQL) database. Entries from the database were then exported to a .CSV file. In order to prepare the data for analysis, the data were converted from raw log events to entries in a spreadsheet that broke each event into multiple columns (e.g. time of event, type of event, name of learning objective, category of learning objective, etc.)

Another part of the preprocessing step included adding missing information. The original goal is that no human intervention would need to take place within the SQL database, but the original database was not displaying the number associated with each deliverable. The bug was fixed, but previous entries had to be fixed manually.

5.2.4. Data Transformation

Data transformation is making changes to the data so they are able to be used by data mining algorithms. This step consists of choosing which features will be used, reducing dimension, or discretizing numerical attributes.

For this step, the data from the spreadsheet from the previous step were converted to quantifiable features (e.g. number of learning objectives, time of first learning objective, number of deletions, etc.) Similarly, any text that students wrote was extracted; words were then tokenized and counted in Python.

Next, all features were scaled to a value between 0 and 1. For some of our classification models, feature selection was performed by using a function in the scikit-learn library called *selectKBest* which allows us to input a value for K and a performance metric. Our performance metric was Chi-Square, which is calculated using Equation 5.1 where O_i represents the observed values and E_i represents the expected values.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \tag{5.1}$$

Therefore, the K features with the highest Chi-Square value are the only ones used in the final model. The comparison of the models that did and did not use feature selection is included in Chapter 7.

5.2.5. Choosing Appropriate Data Mining Task

As seen in Chapter 3, there are five main tasks performed in EDM: classification, clustering, association analysis, sequential pattern analysis, and process mining. We chose to start with classification in order to predict which students would be successful in the course and clustering in order to see if any other patterns emerged.

5.2.6. Choosing the Data Mining Algorithm

After choosing your tasks, there are multiple algorithms that can be chosen from. One deciding factor when choosing algorithms is the type of data collected. Some algorithms allow only for categorical data, others only allow for quantized data, and others allow for both. In addition, the size of the data set is important to consider. With the small, complex data sets often found in educational data, overfitting can be a big problem. When overfitting occurs, algorithm might perform well on training data, but poorly on new data. There also needs to be a balance between power, flexibility, and ease of understanding. Instructors and other non-experts in data mining are more likely to understand and trust the algorithm's findings if they are able to understand how they work. All of these variables must be weighed when choosing an appropriate data mining algorithm.

5.2.6.1. Choosing Classification Algorithms

Possible choices for classification include decision trees, nearest neighbors classifiers, neural networks, support vector machines, Bayesian classifiers, and logistic regression. These algorithms are shown in Figure 5.2.

Decision trees are a common, simple form of classification. The decision tree is made up of branches, where each branch represents a possible outcome. Although decision trees are easy to understand, they can struggle with educational data because they are prone to overfitting if the data sets are small and are often unstable (a small change in data can lead to a major change in the structure of the tree).

In order to use something that updates more efficiently with new data, K-Nearest Neighbor Classifiers can be considered. K-Nearest Neighbor Classifiers work by finding similar data points rather than building a global model. This algorithm is fairly simple to understand and implement because it only needs two parameters: number of neighbors to consider K and distance d. Choosing K is straightforward, but one of the main disadvantages of this method is the difficulty of choosing d. Because the features of educational data sets are often on different scales, it is often necessary to develop a weighted distance function, meaning lots of data is needed to correctly determine the weights. One possible solution is to remove features that are not relevant to the classification, but that can be difficult to determine.

Neural networks are extremely common in machine learning problems because of their flexibility and power. Neural networks are comprised of algorithms inspired by the workings of the human brain; information moves through the network by passing through nodes that each perform a different function on the data. As more and more data is passed through the network, the more the network learns and the more accurate it becomes, making it especially good at image and language processing. However, they are usually not the most appropriate algorithms for educational data because they require large amounts of numeric data and a lot of knowledge about how to best train the model. Educational data sets usually do not have enough quantifiable data and are therefore prone to overfitting. In addition, neural networks are seen as somewhat of a black box, meaning that the workings might be difficult to understand.

If our data set is small, support vector machines might be a more appropriate algorithm because of their ability to find nonlinear class boundaries even with little data. Support vector machines work by using kernel functions that implicitly map the data to a higher dimension without using valuable computing time. When the data is converted to a higher dimension, a linear class boundary can be found without overfitting. Sometimes this algorithm struggles to deal with outliers, but this can be avoided by using soft margins (allowing some data points to be misclassified by the algorithm if it appears that the point is an outlier). Support vector machines also have some disadvantages; like neural networks, all features must be quantized, and it can be difficult to understand the workings of the algorithm.

Naive Bayes is another algorithm that is common in EDM. The algorithm takes training data and determines the probability of all features for each possible class. It can then return the most probable class given new sets of feature values. The algorithm works using categorical input variables, so any numerical variables must be discretized. Problems can occur when using Naive Bayes if a variable does not appear in the training set but appears in later data. In addition, Naive Bayes assumes that all variables are independent of each other, which is not often the case in real life. However, there are ways to help offset these issues, and Naive Bayes is still widely used because of its ease and relatively good performance compared to other models.

Finally, logistic regression is another algorithm that works to find a class boundary that minimizes the classification error. It is similar to a support vector machine, but it calculates error in a slightly different way.

With our data, we wanted to choose algorithms that 1) allowed for quantifiable features (not just categorical), and 2) had the ability to extract the most pertinent features. Predicting the right class can be helpful, but understanding why a prediction was made can be even more helpful for an instructor. Therefore, support vector machines, logistic regression, and K-Nearest Neighbors were chosen as options.

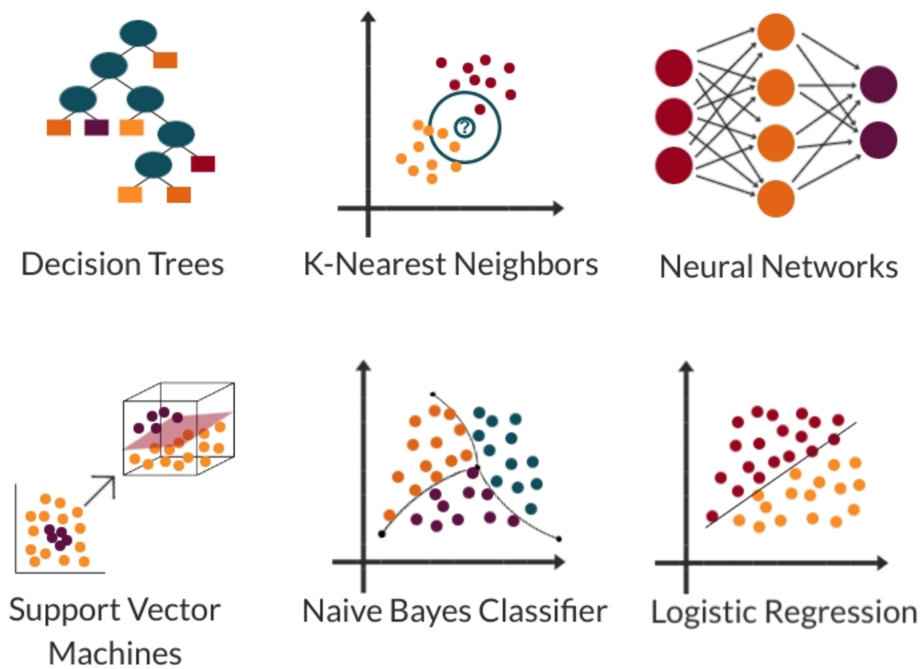

Figure 5.2. Common classification algorithms

5.2.6.2. Choosing Clustering Algorithms

Within clustering, there are two main types: hierarchical and partitional [16]. Hierarchical clustering determines which students are most closely related and forms a tree with each student acting as branch. Partitional clustering breaks students into a specific number of groups; each student falls into only one group.

We chose hierarchical clustering because by looking at the branches of the tree, we can see how many clusters emerge. In addition, information can be gained by seeing the which students and clusters are most closely related. We used a Ward's agglomerative hierarchical clustering as detailed in [70].

5.2.7. Employing the Data Mining Algorithm

This step consists of running the data mining algorithm and tuning its parameters until an acceptable result is achieved. The algorithms were all written and run in Python using the following libraries: Numpy [71] to perform mathematical functions, scikit-learn [72] to perform machine learning tasks and evaluation, pandas [73] to read and write CSV files, and NLTK: The Natural Language Toolkit [74] to process the raw text data extracted from students' learning objectives. Python is often used in EDM because of its power and flexibility [75]. In addition, Python is open source and could be more easily implemented into a learning management system.

5.2.8. Evaluation

Evaluation is quantifying the performance of the results acheived in Step 7. This step also consists of comparing each of the models that were built to see which has the highest performance.

5.2.8.1. Evaluation of Classification Models

For classification, a variety of performance metrics can be used for each model. For this work, accuracy, recall, F1 score, and Receiver Operating Characteristic Area Under the Curve (ROC AUC) were calculated and used for evaluation because they are commonly used in classification EDM problems [54].

Accuracy of a model calculates the percentage of correct classifications. It is the most commonly reported performance metric, but does not always give a complete picture, so other metrics should be used as well. In our case, accuracy tells us how many students are model correctly classified into either successful/unsuccessful. Accuracy can be calculated using Equation 5.2.

$$Accuracy = \frac{TruePositive + TrueNegative}{Total} \tag{5.2}$$

Recall, also known as sensitivity, gives the proportion of true positives to total positives. In our case, recall tells us how many of the low performing students were correctly identified. Recall was chosen as a performance metric because we would rather have a student who was on track in the course be identified as potentially needing help than have a student who is not on track be identified as a student who doesn't need as much help. Recall can be calculated using Equation 5.3.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{5.3}$$

The F1 score of a model is the harmonic mean of precision (how many of the identified positives were actually positives) and recall. F1 score is advantageous because it takes into account both precision and recall. In our case, precision looks at how many of the students that we identified needing help actually needed help, and recall tells us how many students that needed help actually were identified as needing help. F1 score can be calculated using Equation 5.4.

$$F_1 = 2 * \frac{Precision * Recall}{Precesion + Recall} \tag{5.4}$$

A ROC is a graph that has True Negatives on the X-Axis and True Positives on the Y-Axis. It plots points for the entire range of thresholds (or cutoff points between classes). The typical threshold value of a classifier is 0.5. If the model predicts a probability of less than 0.5 for an item, that item is classified as a 0. If the the model predicts a probability of more than 0.5, that item is classified as a 1. The ROC allows us to see how the classifier works at any probability threshold. In order to quantify the ROC, we can calculate the AUC. A ROC Auc of 1 means that the model perfectly differentiates between 0s and 1s, a ROC AUC of 0.5 means that the model is equivalent to random guessing, and a ROC AUC of less than 0.5 means that the model is worse than randomly guessing. In the context of the IBL data, a high ROC AUC means that there was clear separation between the students that would achieve high external value and those that wouldn't.

For all of these metrics, ten-fold cross validation was used, which is a practice often found in EDM (e.g. [76, 77, 78]. Many machine learning problems split their data into a training set and a test set; the model is trained with the training set and accuracy is calculated for the test set to see how well the model works on new data. However, when working with a small data set of students who are all very different, it is imperative to use all students' data when training the model; if any student is not included in the model, important insights might be missed. That being said, if all students are included in the training set, there is no test set to evaluate the model. Therefore, a method called cross validation should be used. In k-fold cross validation, the dataset is split up into k equal parts (folds). k-1 folds are then used to train the data, and the model is tested using the last fold. Accuracy, recall, F1 score, and ROC AUC can then be calculated for that iteration. This is then repeated k times. Each fold is used as training data in k-1 of the iterations and as test data during 1 of the iterations. The average of the accuracy, recall, F1 score, and ROC AUC can then be calculated across all k iterations to get a final value [79].

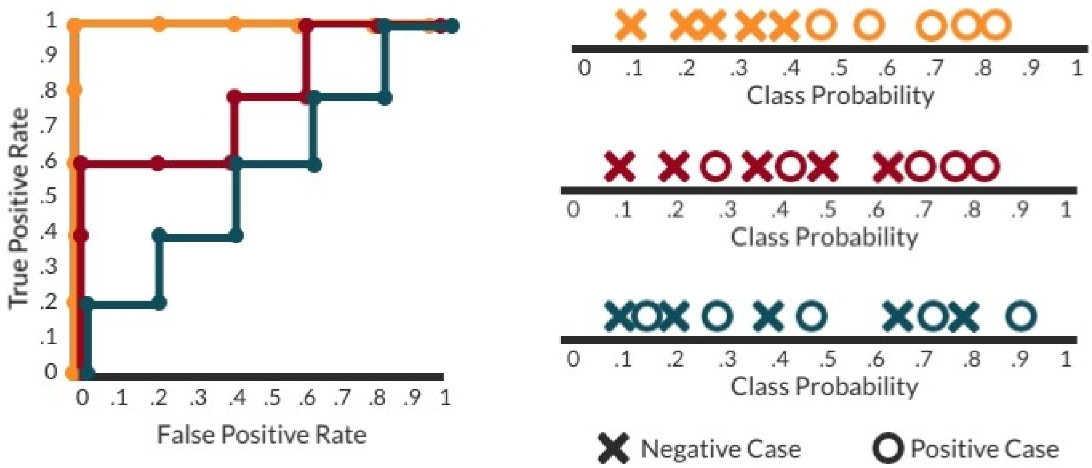

Figure 5.3. The Receiver Operating Curves (ROCs) for 3 different datasets are graphed. The false positive rate is graphed on the X-axis and the true positive rate is graphed on the y-axis. Each point on the graph represents a different decision boundary, or the value that differentiates between positive and negative classes. The yellow dataset has a clear boundary between the Xs (negative cases) and Os (positive cases). Therefore, the ROC has an AUC of 1. The red dataset still has some separation in the Xs and Os; the ROC AUC is 0.8. The blue dataset has the least separation in the Xs and Os; the ROC AUC is 0.6. Note: Although the yellow dataset has a perfect ROC AUC score of 1, the classifier itself would not have a perfect classification. The decision boundary of a classifier is automatically set to 0.5. Therefore, if we split the yellow dataset at 0.5, the O at about 0.45 would be classified as a negative case, putting the classifier accuracy at 0.9.

5.2.8.2. Evaluation for Clustering Models

Evaluation for clustering is less straightforward. Because there are no classes assigned to each data point, it is not possible to calculate accuracy or other quantifiable metrics. Rather, the success of a clustering method is determined by the amount of information the results provide [37]. If the clustering model provides new insights, then the model can be deemed successful. In addition to exploring the information provided by the model, we also compared the algorithm's clustering of students to the instructors' clustering of students. This is not necessarily a measure of accuracy, but the results of this comparison can help us gain insights into the model.

5.2.9. Using Discovered Knowledge

By using the discovered knowledge, changes can be made within a system. The success of the entire process is determined in this step because it becomes clear if the knowledge gained was only applicable in one snapshot or if it can be applied at a wider scope. Multiple patterns and insights were discovered during the KDD process, but future work will be needed in order to see how well the models hold up.

5.3. Conclusion

This chapter detailed the KDD process and gave details about the algorithms chosen and evaluation metrics. Chapters 6, 7, and 8 all use the theory and methods presented.

6. DEVELOPING AN UNDERSTANDING OF INNOVATION-BASED LEARNING

6.1. Introduction

The first step of the KDD process is to have a strong understanding of the domain you are working in. Therefore, the work detailed in this chapter helped guide all future decisions and conclusions. This chapter will discuss the motivation behind Innovation Based Learning (IBL), detail the process of IBL, and explain how it is applied in a Cardiovascular Engineering course. It will also give examples of student learning objectives, discuss the reliability of IBL as an assessment form, and speak to student attitudes about the course. Finally, it will present some key takeaways from the assessment process for both instructors and educators. This initial exploration of the assessment process provided us with a better understanding of the students and course, which plays a role in the analyses discussed later in the thesis.

6.2. The Motivation behind Innovation Based Learning

In the world of engineering (and beyond), an ability to innovate is of the utmost importance. However, traditional high-stakes assessments have been shown to squelch innovation both for the instructors organizing the course and the students that are working within the boundaries of the course [81]. Therefore, work is being done to design assessment that allows for student freedom with strategies like project-based learning and learning portfolios [82]. Many researchers have found benefits when implementing more opportunities for student-directed learning both in higher education [42, 83, 84, 85] and the K-12 system [86]. Giving students ownership and flexibility increases motivation [84, 42], improves student engagement [87, 42], helps with information retention [87, 88], and promotes lifelong learning [89, 88].

6.3. Format

The assessment process has three main components: 1) Students develop their own learning objectives and share them with the class and instructor, 2) Students use Bloom's Taxonomy to

Some material in this chapter was drawn directly from [80], a publication co-authored by Lauren Singelmann, Enrique Alvarez, Ellen Swartz, Mary Pearson, and Ryan Striker. Lauren Singelmann drafted and revised all versions of this chapter. Other authors served as reviewers of the content.

help illustrate to what level they will learn each of their desired objectives, and 3) Students will be assessed based on the amount of external value they achieve through their objectives.

6.3.1. Student-Developed Learning Objectives

Normal assessment usually has instructors develop learning objectives and ways to assess that those learning objectives are met. However, in this form of assessment, students fill this role. While working on a project and learning course content, students are expected to write learning objectives that explain what they will learn, to what level they will learn it, and how they will demonstrate it. By writing learning objectives, students are taking part in the process of metacognition, which helps solidify both content and skills [90]. To give students ideas for objectives, categories are given to the students from which they can choose. These categories range from literature review to data collection to conference presentations to business models.

6.3.2. Bloom's Taxonomy

Because of the large amount of freedom when writing objectives, Bloom's 3D Taxonomy of Learning [69] is used to help provide students with scaffolding. In the first week of class, students are taught about the taxonomy (shown in Figure 6.1) and learn about how to build from low-level to high-level learning. Students start by showing low-level learning (e.g. writing a report that shows understanding of concepts) which then builds into high-level learning (e.g. publishing a paper about the creation of an experimental procedure). Classifying learning with Bloom's 3D provides structure while still allowing for student freedom.

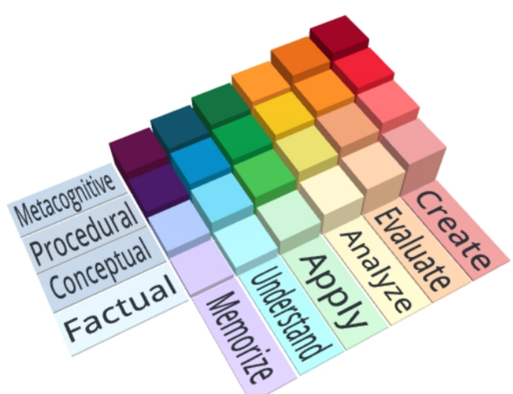

Figure 6.1. Bloom's 3D Taxonomy adopted from [15]

6.3.3. External Value

Assessment in the course is done by measuring external value, which consists of 1) providing value outside the classroom, and 2) some sort of external review from the scientific community or end-users. For example, an in-class presentation would be lower external value than presenting at a business pitch competition. Other examples of external value are shown in Table 6.1. Students feel invested because they have the freedom to choose a form of external value that most closely aligns with their personal and professional goals, and they are able to work on meaningful solutions that benefit their community.

Table 6.1. Examples of deliverables at each level of external value

Level of external value	Examples
Low	Tests, quizzes, homework, in-class surveys, reviewing others' evidence, documented general assistance to the class
Medium	Standard operating procedures, non-refereed conferences, providing expertise to other research groups in a lab
High	Invited outreach activities, refereed conferences, refereed journal manuscripts, scholarships, fellowships, awards, invention disclosures, business pitches, business plan competitions

6.4. Application in a Cardiovascular Engineering Course

6.4.1. Structure of the Course

A 3-credit Cardiovascular Engineering Course offered by the Department of Electrical and Computer Engineering has used a form of this assessment style for the last four years, and student data was collected during the most recent iteration of the course. Students learn five main cardiovascular engineering concepts (functional block diagram of the cardiovascular system, resistance and compliance concepts, pressure/volume loops and time domain, ECG, and arterial systems) and are expected to demonstrate their competency in each of these areas. These objectives fall under the category of Discipline Specific Knowledge 0 (DSK0), the only required objective. Students watch videos outside of class about each of the topics and are expected to come to class ready to participate in worksheets and discussion. Beyond DSK0, students are allowed to write their own objectives and edit them as the course goes on. Class time is dedicated to both digging deeper into DSK0 concepts and having students present learning updates where they share their objectives with the class and instructors, get feedback, and offer support to other groups [5, 91]. If students show competency in each of the five areas of DSK0, they are at a grade level of a C. If students apply the knowledge to a project, they are at a B grade level. Finally, if students achieve high external value with their project, they will receive the grade of an A.

6.4.2. Choosing a Team and Topic

As students decide on learning objectives, most of the learning is based around an innovation project that teams choose. At the beginning of the semester, students look at cardiovascular-related funding opportunity announcements from agencies like National Science Foundation and National Institute of Health to determine projects of interest. From there, students pitch project ideas and form teams based around the projects [92]. Students are not evaluated based on their ability to solve the problem presented in the funding opportunity announcement, but rather on their ability to demonstrate how they applied their learning to their innovation project and share it with a broader audience.

6.4.3. Logging Learning Objectives

Students use an online portal to log learning objectives and corresponding deliverables, allowing them to track progress on each objective in real time [93]. Each student has multiple learning objectives, and each learning objective may have one or more deliverables. Learning objectives are categorized with Bloom's Taxonomy and the Learning Objective codes. Deliverables are categorized with the level of external value, expected completion dates, and current progress level (not started, in progress, and completed).

6.5. Methods

6.5.1. Participants

28 students chose to share their learning objectives during the span of the course. Of those 28 students, 22 were male and 6 were female, and the mean age of the group was 26.5. There were 13 undergraduate seniors, 3 Masters, and 11 PhD students (1 student did not provide a response to this question). A variety of majors and programs were also represented in the sample. The class is offered by the department of Electrical and Computer Engineering, but other departments allow their students to take the class for technical elective credit. 9 students were in Biomedical Engineering, 9 in Electrical Engineering, 5 in Mechanical Engineering, 4 in Computer Engineering

and 1 in Health, Nutrition, and Exercise Science. This offering of the course was different from other years in part because students from a partner university and distance students were allowed to enroll. 20 of the students in the study were from the local university, 5 were from a partner university, and 3 were distance students.

6.5.2. Learning Objective Collection

Throughout the semester, all students logged their progress in an online portal where they could add, categorize, update, and delete learning objectives and deliverables. Whenever a student made an addition, change, or deletion, the event was logged in a searchable database [68]. Therefore, in addition to being able to analyze the end state for each student, we could also analyze the steps that it took to get there. Trace data were collected because self-reported data about metacognition is often inaccurate. Although this method loses some student perspective, it does gain temporal accuracy by being able to reference the log data directly [64].

6.5.3. Assessment Reliability

In order to measure the reliability of the assessment process, all six members of the instructor team graded the level of external value of each of the students in the class. Two of the raters consistently attended class, and all six have either taken or taught the class in the past. The raters discussed the grading criteria, but did not discuss individual instances until after the individual ratings had been completed. The interrater reliability was calculated by taking Fleiss' Kappa, which measures agreement while factoring out agreement due to chance [94].

6.5.4. Post-Survey

During the last two weeks of the semester, students were asked to fill out an online survey about the class. 24 students responded to the survey, and it should be noted that the students that completed the survey are not necessarily the same students that agreed to share their learning objectives. 26 questions were asked, 5 of which were most pertinent to the research questions of this paper and will be discussed here. Other questions were focused on topics such as the team formation process, team composition, and use of various software in the class. Those questions will be published in other papers exploring these specific topics. The 5 questions are listed below:

- Note something that this class has inspired you to do better or differently. (Free response)

- Describe new competencies you have learned/developed (skills, tools, methods, software). (Free response)
- Describe qualities you have discovered about yourself (characteristics, traits, features) through the class. (Free response)
- I would recommend this class to another student. (5-point Likert scale)
- How would you describe the class to a peer? Pick all that apply. (15 words were included as options as well as a space for students to write their own)

The first three questions were adapted from Jaeger [95]. The goal of these questions was to assess student perceptions about skills they have gained without priming them by asking about specific skills. The goal of the last two questions was to assess overall student sentiment about the class.

6.6. Results

6.6.1. Learning Objective Collection Results

Of the 28 students that shared their learning objectives, 18 clearly achieved high external value, 8 were borderline, and 2 did not provide any evidence at all. Learning objectives for two example students are included below. Student A is an example of a student that clearly achieved high external value. Student B is an example of a student that attempted an innovation project but was not quite up to the level of high external value.

Student A had five main objectives, each with clear deliverables and linked evidence. They are shown in Table 6.2. The student's project was to work with their team to build a prototype of a multi-parameter biosensor. In addition to learning the main fundamentals of the cardiovascular system (DSK0), they also determined that they needed to better understand how ECG, cough frequency, and respiratory rate relate to each other (DSK3). They showed that they learned these topics by summarizing what was read in a variety of publications. In addition to gaining information, they also worked on team conduct, evaluating a product, and outreach communication. They demonstrated external value by presenting at a poster session and getting review, sharing their work with a wider audience, and winning an award for their work. Many of student A's objectives are at the metacognition and/or creation level.

Table 6.2. Student A's learning objectives and deliverables

Learning Objective	Deliverables	Linked Evidence
Learn the fundamentals of the cardiovascular system (DSK0: Fundamental Cardio Concepts, Understand, Conceptual)	Create a functional block diagram	Virtual copy of notes
	Understand R&C relationship and pressure volume loops	Virtual copy of notes
	Understand ECG concepts	Virtual copy of notes
	Understand the arterial system	Virtual copy of notes
Develop a measurable relationship between ECG, cough frequency, and respiratory rate (DSK3: Learning outside of student's College, Create, Procedural)	Compile resources that discuss the relationship between variables	List of 12 peer-reviewed journal articles
	View resources given by other teammates to develop understanding of biosensors	Link to a shared folder with papers from other group members
	Determine limits of variables	A summary slideshow with information compiled in literature review and applied to project
Collaborate with team to complete project (RM6: Team conduct, Evaluate, Metacognitive)	Create Gantt chart to map out project timeline	Link to Gantt chart document
	Create google drive to compile all documents/progress of project	Link to shared drive
Develop the multi parameter biosensor into plan for a prototype (ES5: Product evaluation, Create, Procedural)	Create layout of expected device	Block diagram of the device
	Create a 3D model of the expected device	Screenshots of model
	Begin a material analysis for future material selection in prototype stage	Document with advantages and disadvantages of various materials for each design component
Communicate technical knowledge that relates to the group project (PC7: Outreach communication, Create, Metacognitive)	Create a poster that communicates overall idea of project	Copy of poster
	Obtain feedback from class for revisions of symposium poster	Copy of the poster with new revisions
	Present poster at graduate symposium	Photo of group at the symposium

Student B also listed 5 objectives with evidence, but the external value of the work is less clear. Their learning objectives and deliverables are shown in Table 6.3. Some possible ways for the student to improve would have been to find a clearer need to fulfill, rather than making a website that might not be helpful or easy to find. Also note that the Bloom's categorization levels do not have anything at the creation or metacognitive level.

6.6.2. Assessment Reliability Results

After all six raters scored all students, Fleiss' Kappa was calculated to determine interrater reliability. One rater had a misunderstanding about some of the students' deliverables, so some of that rater's scores were adjusted before large group discussion began. Kappa was 0.412 before this adjustment, and was 0.505 after the adjustment. A score close to 0 is considered no better agreement that if the raters had randomly scored the subjects; a score close to 1 is considered almost

Table 6.3. Student B's learning objectives and deliverables

Learning Objective	Deliverables	Linked Evidence
Class Learning (DSK0: Fundamental Cardio Concepts, Understand, Factual)	Make connection of in class learning	Summary of notes
	In-class worksheets	Virtual copy of worksheets completed in class
Website (DSK2: Learning in student's College, Evaluate, Conceptual)	Create a template	Link to the website
	Website content review	Link to a form where website users can submit feedback
	Design critique	No evidence linked; marked as still in progress
Code documentation (DSK3: Learning outside of student's college, Understand, Procedural)	Make READme documentation	Link to code repository with READme file
	LO contribution	Document explaining how work was split between group members
Cardiovascular genetics (DSK3: Learning outside of student's College, Understand, Factual)	Find/read an educational journal	Link to 13 online sources
	Create intro to genomics video	Link to video
	Add information to website	Link to website that student created about cardiovascular genetics
Distance collaboration tools (DSK3: Learning outside of student's College, Understand, Procedural)	Use Google Drive for working collaboration	Link to shared drive
	Discord	Copies of meeting minutes (meetings were held using the program, Discord)

perfect agreement. Although there is no officially agreed upon benchmark for Fleiss' Kappa, 0.4-0.6 is considered moderate agreement, meaning there is still room for improvement in increasing reliability. Suggestions for improving interrater reliability is included in the Discussion section under *Takeaways for Evaluators*.

6.6.3. Survey Results

For the three open response questions, six gained skills were identified from the emergent coding. The categories and the number of students who mentioned each skill are shown in Figure 6.2. The most commonly mentioned skill gained was communication. Student responses that were coded in the communication category ranged from improving presentation skills to better communicating projects to non-engineers to technical writing. A large number of students also had responses mentioning teamwork, leadership, and teaching others. In addition, some students mentioned gained technical skills, including five responses about software and programming, two about cardiovascular engineering, two about web design, and one about electronics skills.

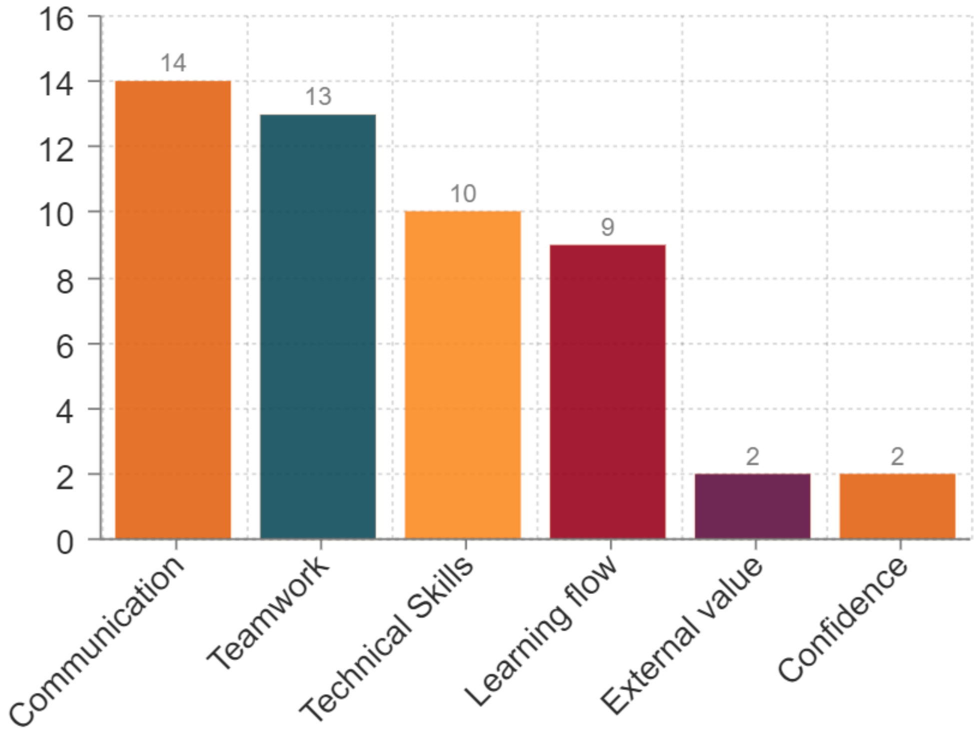

Figure 6.2. Top skills gained identified by students

The next category was learning flow, or the ability to direct your own goals and learning. Nine students included a response with this theme; they mentioned that they learned how to set goals, go out and find new information, and assess themselves. Two students mentioned the idea of external value or being able to identify and solve a problem that meets a clear need, and two other students mentioned improvements to their confidence.

Figure 6.3 shows how students responded to the prompt, "I would recommend this class to another student," and Figure 6.4 shows the top words chosen to describe the class to a peer. Top words included time-consuming with 17 responses, satisfying with 14 responses, beneficial with 12 responses, frustrating with 11 responses, and motivating with 11 responses. Students could choose multiple words from the list or write in their own responses.

6.7. Discussion

6.7.1. Takeaways for Educators

This unique form of assessment puts students in the driver's seat, allowing them to focus on learning both content and skills that are called for by ABET and engineering employers and

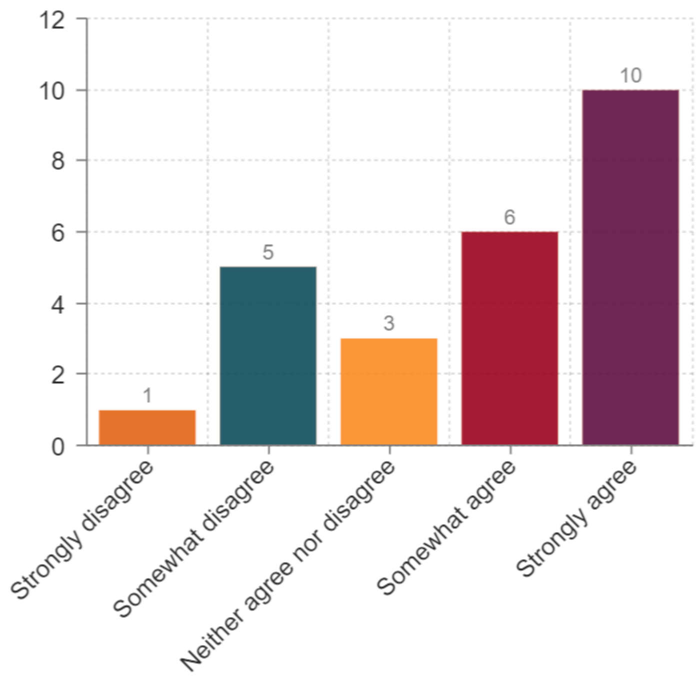

Figure 6.3. Responses to the question: “I would recommend this class to another student.”

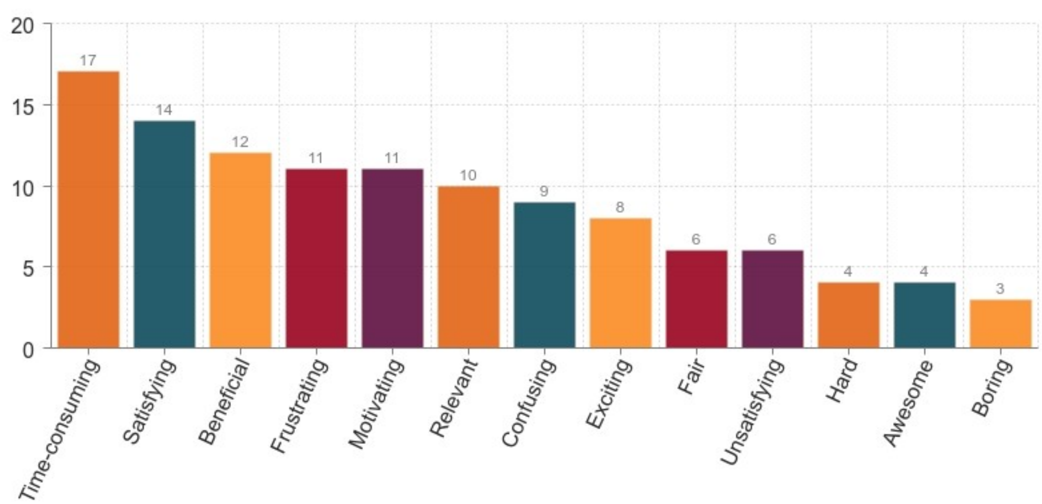

Figure 6.4. Top words chosen to describe the class to a peer

institutions. The assessment strategy requires students to identify and solve complex problems (ABET Desired Student Outcome 1), develop a solution (ABET Desired Student Outcome 2), and run experiments and collect data (ABET Desired Student Outcome 6). Students recognize that they are working to improve communication, leadership, and their ability to acquire new knowledge (ABET Student Outcomes 3, 5, and 7, respectively). Many students met and exceeded expectations by publishing, presenting, and submitting invention disclosures while gaining knowledge about cardiovascular engineering. However, to help more students succeed and decrease confusion, more time should have been spent at the beginning of the semester defining terms and how students will be assessed. For example, when an instructor says “research paper”, they often think of a peer-reviewed publication; a student, on the other hand, might think of writing a summary of existing information. Spending time defining some of these outcomes at the beginning of the semester will help students plan accordingly and rise to the challenge at hand. Another way to better support struggling students is to encourage more entrepreneurial thinking. Who is their customer/audience, and what are their wants/needs? By focusing on these ideas, students can better understand the idea of external value and find more ways to add external value to their work. Finally, reviews should occur early and often. By communicating what students are doing well and what they can improve upon, they begin to feel more comfortable with the control they have.

6.7.2. Takeaways for Evaluators

For those that are evaluating students at the end of the semester, it is important to make sure both the students and instructors know how you plan to evaluate different cases. For example, what will you do if there is missing evidence? Will you allow students to provide it after the due date? For this class, students were allowed to clarify the level of external value of a deliverable by providing more information, but they were not allowed to add more evidence after the due date passed. On a similar note, what happens if DSK0 is not included and the fundamental cardiovascular engineering elements are not demonstrated? In this class, students were dropped a letter grade, but that does bring up more questions about how to ensure that all students are meeting the main content-oriented learning objectives of the class. Finally, most disagreement in evaluators was about the difference between a B and a C. To get a B, students were supposed to apply their knowledge to an innovation project, but what happens if their project isn’t innovative? How can we recognize the effort they put in, but also encourage them to focus more on the innovation? These questions need to be answered by any instructional team that is considering implementing this assessment technique.

6.8. Conclusion

Having a strong understanding of the many facets of the course is an important component of being able to make claims from the data mining algorithms, especially with complex problems. This chapter gave background about the pedagogy and course, presented example learning objectives, and discussed assessment challenges. These conclusions will be used to guide the work in Chapter 7 on the classification models and Chapter 8 on the cluster models and will be an integral part of discussing the results as a whole in Chapter 9.

7. CLASSIFICATION RESULTS

7.1. Introduction

One of the main research questions was if we could use classification methods to predict which students will perform well in the course. In order to explore this problem, two types of feature sets (quantitative and text) and three types of algorithms (support vector machine, K-nearest neighbor, and logistic regression) were compared. In addition, the performance of the model over time was assessed in order to discover how early in the semester the model can be used for accurate prediction. Finally, the most important features that differentiated between top performing students and lower performing students were extracted in order to better understand what activities differentiate between high and low performing students.

This chapter details the feature sets and algorithms used, compares the results from each of the models, graphs model performance over time during the semester, and presents the features that the models deemed most important. Finally, the results and overall insights gained are discussed.

7.2. Methods

7.2.1. Feature Collection

Two main types of features were used and compared: quantitative data and text data. The quantitative features that were extracted from the data are shown in Table 7.1.

For the text data, all learning objective and deliverable titles and descriptions were extracted for each student. Using the scikit-learn library in Python, all the words that students wrote in their objectives and deliverables were tokenized, counted, and scaled.

7.2.2. Models and Feature Sets

In order to predict which students would achieve high external value during the course of the semester, three classifier models were tested: Support Vector Machine (SVM), Logistic Regression (LR), and K-Nearest Neighbors (KNN). These three models were chosen because they have some level of interpretability, an important feature in EDM [97]. In order for instructors to

---

Some material in this chapter was drawn directly from [96], a publication co-authored by Lauren Singelmann, Enrique Alvarez, Ellen Swartz, Ryan Striker, Mary Pearson, and Dan Ewert. Lauren Singelmann drafted and revised all versions of this chapter. Other authors served as reviewers of the content.

Table 7.1. Types of quantitative features collected by the platform

Category	Description	Example features
Total counts	Countable features from end of semester data	Number of planned learning objectives, number of logins, number of deletions, etc.
Quarter-Based Progress	The course was split up into quarters and progress was calculated for each quarter to see how students broke up their work	Number of deliverables completed during quarter 2, number of learning objectives deleted during quarter 4, etc.
Specific learning objectives	These features checked for the presence of different learning objective categories	Presence of *Invention Disclosure* objective, number of *Fundamentals of Research* objectives, etc.
Level of learning	Calculated from the Bloom's Revised Taxonomy categorizations and level of external value	Number of high external value deliverables, average level of Bloom's

use the discovered information, they need to be able to understand where it was derived from. The baseline model was a Majority Class (MC) classifier.

In addition to comparing the models, both the text and quantitative features were compared. For each set, we also compared using all features to using the top K features. K was optimized and set a 24 for text and 15 for quantitative.

7.2.3. Evaluation Metrics

Each model was evaluated by calculating accuracy, recall, F1 score, and Area Under Receiver Operating Characteristic Curve (AUC). Accuracy is the proportion of correctly classified students to all students. Recall is the proportion of students that the model identified as not being on track to success to the number of total students that did not achieve high external value during the course. F1 score is a performance metric that takes the harmonic mean of precision and recall. AUC is the area under the Receiver Operating Characteristic (ROC) curve which shows how well the model can differentiate between the two classes. All models were evaluated using ten-fold cross validation.

7.2.4. Trajectory

In addition to exploring models that were developed by using each student's final learning objectives and deliverables, we were also able to explore how prediction power of the models changed during the course of the semester. Models were created using daily snapshots of all students to see at which point in the semester the model can begin predicting student success.

7.3. Results

7.3.1. Comparing Models and Feature Sets

Table 7.2 shows the accuracy, recall, F1 score, and AUC for each of the models and feature sets explored. These classifiers used all available data during the semester. Almost all models performed better than the MC baseline test. The text features consistently performed better than the quantitative features, and using feature selection usually improved the model as well. The top models are SVM and LR, both using the top 24 text features. In addition to having low performance, the quantitative models are also difficult to assess in real time. The most relevant features of the quantitative models can give us some information, but they are not as helpful when making predictions. Therefore, we'll focus on using the text models moving forward.

Table 7.2. Performance metrics for each of the models using end of semester data

Feature Type	Model	Accuracy	Recall	F1	AUC
Baseline	MC	.6	-	-	.5
All Text Features	SVM	.783	.85	.758	.831
	LR	**.883**	.85	**.866**	**.972**
	KNN	.583	**.95**	.533	.700
Top 24 Text Features	SVM	**.917**	**.85**	**.9**	.937
	LR	**.917**	**.85**	**.9**	**.952**
	KNN	.783	**.85**	.767	.832
All Quantitative Features	SVM	.7	.6	**.648**	**.704**
	LR	**.717**	.5	.612	.697
	KNN	.567	**.7**	.482	.523
Top 14 Quantitative Features	SVM	.667	.5	.563	**.851**
	LR	**.7**	.5	.597	.798
	KNN	.667	**.9**	**.615**	.65

Halfway through the semester, it would be helpful to know which students are most likely to be successful and which could use extra help. Therefore, classifier models were also created using text data for only the first half of the semester. The results are shown in Table 7.3. From this, we see that using SVM with the top 24 text features gives us the best performance.

Table 7.3. Performance metrics for each of the text-based models halfway through the semester

Feature Type	Model	Accuracy	Recall	F1	AUC
Baseline	MC	.6	-	-	.5
All Text Features	SVM	.7	.6	.631	.732
	LR	**.733**	.6	**.665**	**.756**
	KNN	.6	**.9**	.55	.726
Top 24 Text Features	SVM	**.817**	.8	**.792**	**.979**
	LR	.8	.7	.758	.967
	KNN	.783	**1**	.767	.742

7.3.2. Exploring Model Trajectory

We found that our model performs fairly well at the midpoint in the semester, so our next experiment was to see at what point in the semester we can begin to differentiate between top-performers and lower-performers. All models used the 24 top text features. Figures 7.1 and 7.2 show the accuracy and AUC of the models over time, respectively.

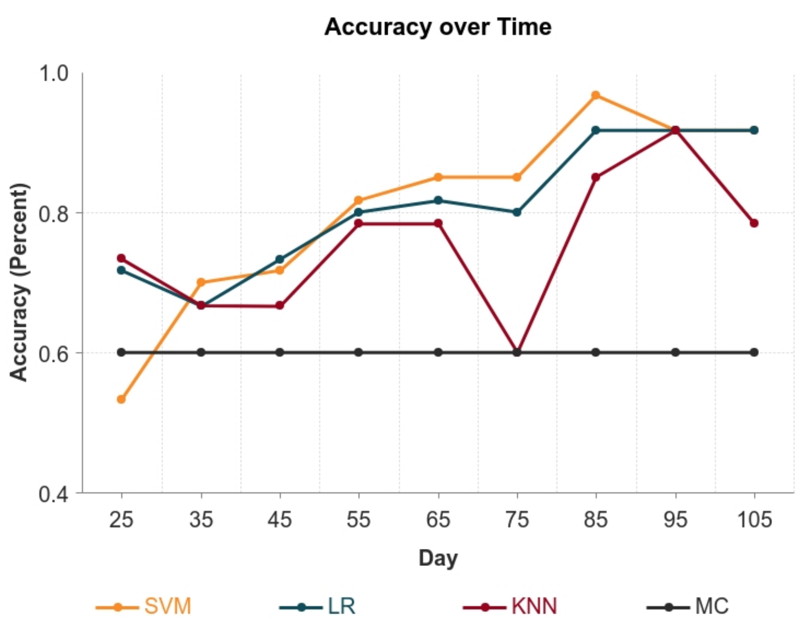

Figure 7.1. Accuracy of the text-based models over time compared with the baseline MC classifier

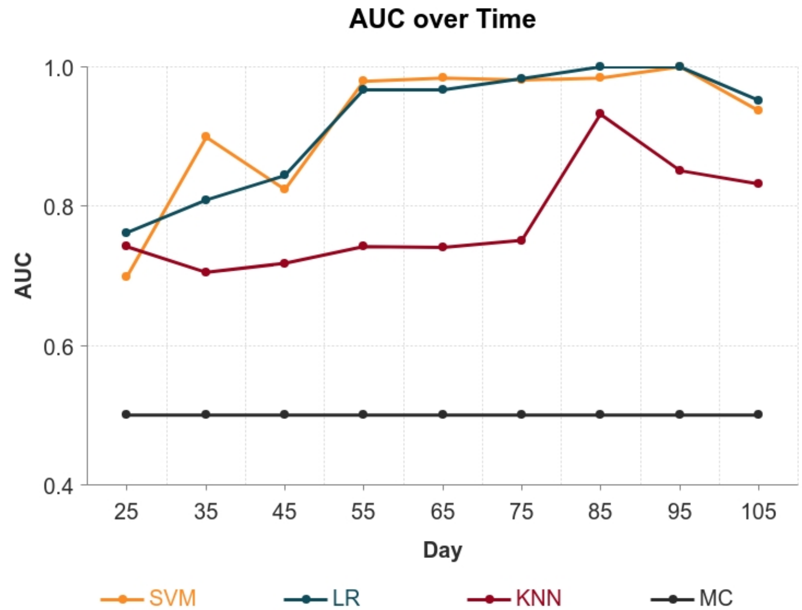

Figure 7.2. AUC of the text-based models over time compared with the baseline MC classifier

7.3.3. Knowledge Discovery

In order to better understand what features are most significant in predicting success, we wanted to be able to extract the most pertinent features. By using linear classifier models instead of black-box models like neural networks and other deep-learning models, we are able to calculate Chi-Square and the weights of each feature. Chi-Square tells us which features are not independent of their classification, meaning they are more likely to differentiate between classes. The greater the Chi-Square value, the greater dependence on classification, meaning that feature is a strong differentiator. Weight can tell us which class a feature is more likely to be found in. A positive weight is more associated with successful students, and a negative weight is more associated with unsuccessful students.

Figure 7.3 shows the 24 features with the largest Chi-Square value. If the weight showed that the word was more likely to be found in a low-performing student, the Chi-Square value was multiplied by -1 to allow for easier interpretation. The top words that differentiated low-performing students were *information*, *presentation*, *engineering*, *website*, *loops*, *review*, and *feedback*. The top words that differentiated high-performing students were *sensor*, *signal*, *model*, *device*, *idea*, and *symposium*.

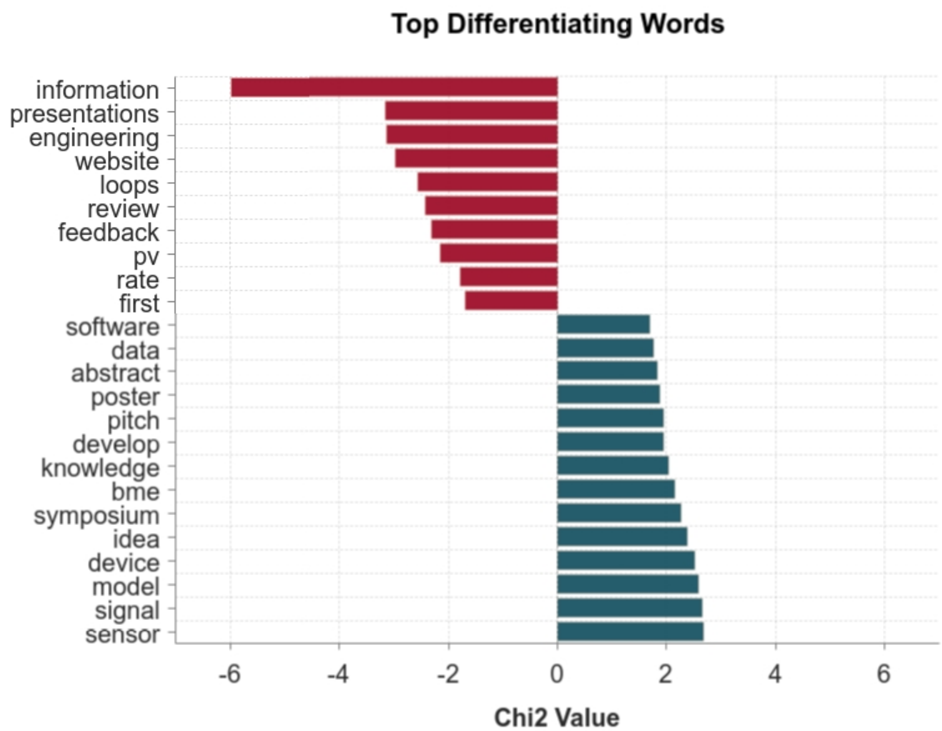

Figure 7.3. The top 24 text features that differentiated the most between successful and unsuccessful students. Words with positive Chi-Square values were more associated with successful students. Words with negative Chi-Square values were more associated with unsuccessful students.

Table 7.4 shows the quantitative features that had the highest Chi-Square values. The weights were used to know which group the variable was more likely to be present in. Top students were more likely to have data analysis, data collection, journal manuscripts, and general Mechanisms of Research learning objectives. Unsuccessful students were more likely to have providing critique and outreach communication learning objectives.

Table 7.4. Quantitative features with the highest Chi-Square values

Variable	Chi-Square	Group
Presence of MR4: Data analysis	3.882	Successful
Presence of RM3: Providing critique	3.091	Unsuccessful
Total number of Mechanisms of Research Learning Objectives	2.146	Successful
Presence of MR3: Data collection	1.941	Successful
Presence of PC5: Journal manuscript	1.941	Successful
Presence of PC7: Outreach communication	1.807	Unsuccessful

7.4. Discussion

7.4.1. Insights Gained

Unsurprisingly, top students were more likely to mention work on their abstracts, posters, pitches, and presence at the BME Symposium (an on-campus biomedical engineering conference). Low-performing students were more likely to have deliverables like websites and outreach activities. Although websites could be high impact deliverables, they can also just be a report of students' lower-level learning. For outreach activities, this can be interpreted broadly and could be outreach to a classmate or small group rather than a visit of high impact.

In addition, successful students were more likely to have words related to the design process such as *idea*, *develop*, and *data*. Unsuccessful students were more likely to mention words like *information*, *presentations*, *review*, and *feedback*. We believe these words appeared in low-level students because they were activities required by the class. Therefore, top students did not see the need to write specific learning objectives about them, but lower performing students added them in an attempt to have more items logged.

7.5. Conclusion

Modeling student learning in open-ended learning environments can be challenging, but SVM classifiers show potential in being able to predict which students will be successful in an IBL course. Models had accuracy of over 80% and AUC of over .95 by the midpoint in the semester. This accuracy increased to over 90% by the last few weeks of the semester. By using linear models, we could also gain insight as to what features differentiated between successful and unsuccessful students. Using these results can help instructors know which students could use extra support and lead to more understanding about how students progress through problem-solving environments in general. By understanding how to better support our students in the innovation process, we can foster the next generation of problem-solvers.

8. CLUSTERING

8.1. Introduction

The second research question explored is about what insights can be gained about the course by using clustering models. How do students fall into clusters, and what clusters do successful students fall into? Do students change clusters over time? If so, how? Finally, what words are most likely to differentiate between clusters?

In this chapter, the clustering process is explained along with how trajectory of students and most pertinent features were determined. Next, the learning frameworks of Bloom's Revised Taxonomy of Learning and Webb's Depth of Knowledge are introduced. The clustering results, student trajectory, and most pertinent features are then presented. These clustering results are then compared with instructor observations in order to better identify strengths and weaknesses of the clustering model. The clusters are then clearly defined and mapped to the Cynefin framework, Bloom's Taxonomy of Learning, and Webb's depth of knowledge. To conclude, insights will be shared and limitations of the model will be discussed.

8.2. Methods

8.2.1. Clustering

Using the scikit-learn library in Python, all the words that students wrote in their learning objectives and deliverables were tokenized and counted. The cosine similarity was then calculated to compare each student with every other student. Agglomerative clustering was then performed on the cosine similarities. This creates a tree that shows which students are most closely related to each other. By looking at the output in Figure 8.1, it was clear that four main clusters emerged. Each student is represented by a branch and is connected to other students at junctions. The lower the dissimilarity coefficient of the junction, the more similar those students were. The colors were added in order to better visualize each of the clusters.

Some material in this chapter was drawn directly from [98], a publication co-authored by Lauren Singelmann, Enrique Alvarez, Ellen Swartz, Ryan Striker, Mary Pearson, and Dan Ewert. Lauren Singelmann drafted and revised all versions of this chapter. Other authors served as reviewers of the content.

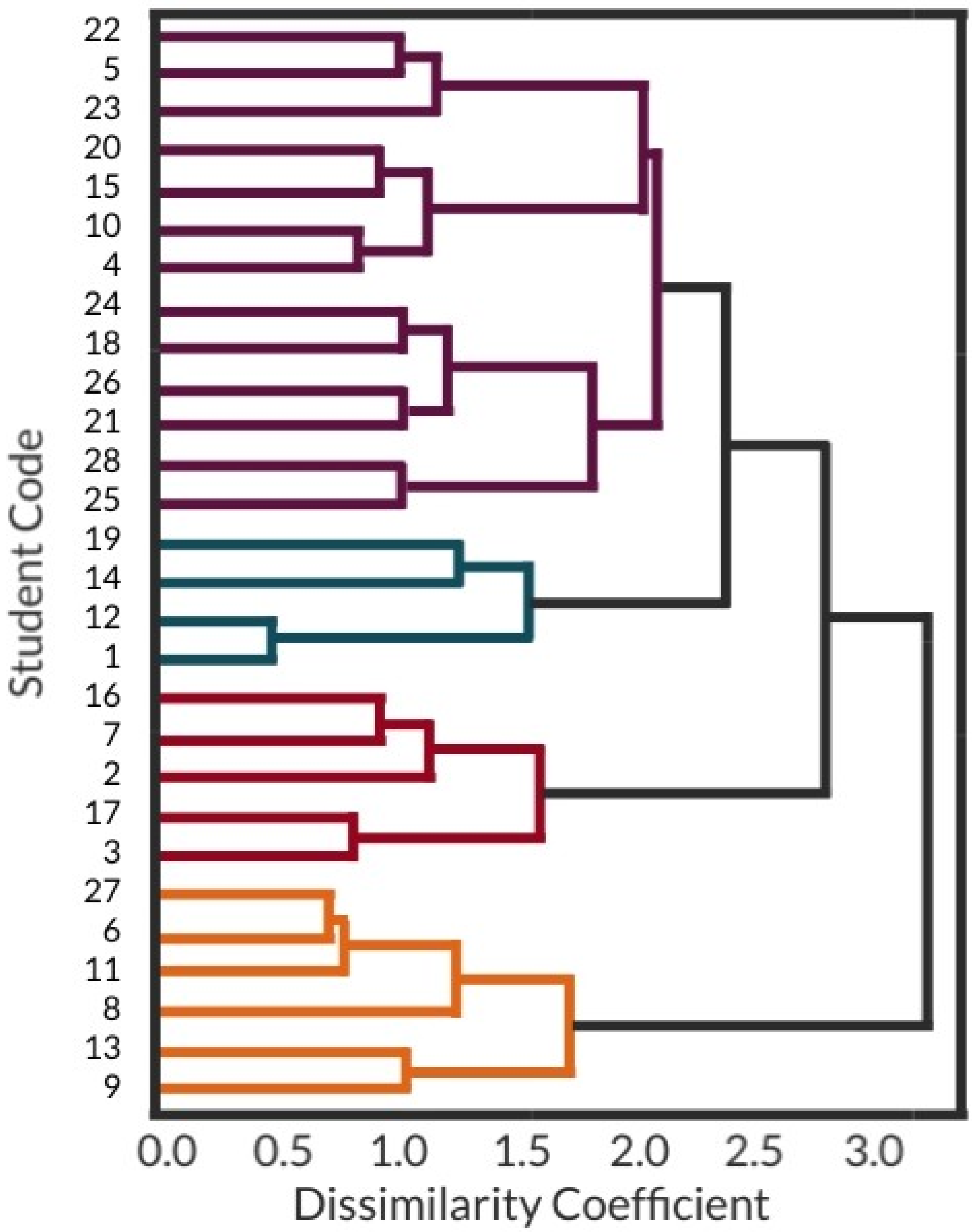

Figure 8.1. A dendrogram showing the hierarchical agglomerative clustering performed on the students. Each branch represents a student. Students that are connected by a branch are most closely related, and the lower the dissimilarity coefficient of a connection, the closer they are related. For example, students 1 and 12 are most closely related because their branches connect at a dissimilarity coefficient of about 0.5. From the dendrogram, it was clear that 4 main clusters emerged from the data. Each of the clusters is colored to help visualize the natural breaks in clusters.

8.2.2. Mapping Student Trajectory

After each of the students had been placed into a cluster, a support vector machine classification model was trained to predict which cluster a new student would fall into. Data for each student was then broken up by day, and the cluster was predicted for each student over time. For example, the dataset for a student on Day 50 would include all the learning objectives and deliverables they had added by Day 50, but none of the data following Day 50. Each student's cluster can then be mapped over time in order to see if and how clusters change.

8.2.3. Extracting Most Pertinent Features

Using the classification model, Chi-Square was calculated for each word, showing which words are more likely to differentiate between classes. The greater the Chi-Square value, the greater dependence on classification, meaning that word is a strong differentiator. Words with the highest Chi-Square values were then sorted into their associated cluster. These words along with analysis of each cluster contributed to the naming of each cluster.

8.2.4. Comparing with Instructor Observations

After the clusters had been formed and named, the descriptions of each cluster were given to two of the instructors. These descriptions can be found in Table 8.1. The instructors then individually grouped the students into clusters based off of their own observations. They then both came together to discuss any students that they disagreed on and came to a final decision about all students. Their results were then compared with the clustering algorithm's results. The inter-rater reliability was calculated by finding Cohen's Weighted Kappa. Cohen's Weighted Kappa was chosen because it accounts for ordered categories. For example, a mismatch of learner and innovator would be weighted as a closer match than a mismatch of surface level and innovator. Finally, the instructors were given the algorithm's groupings and were asked about any discrepancies in order to learn why discrepancies may have occurred.

8.2.5. Mapping Clusters to Learning Frameworks

Each cluster was mapped to three frameworks: the Cynefin framework, Bloom's Revised Taxonomy of Learning, and Webb's Depth of Knowledge.

The Cynefin framework can be broken up into four domains: simple, complicated, complex, and chaotic. Things in the simple domain are consistent. The cause and effect relationships between

Table 8.1. Cluster descriptions

Name	Description
Surface Level	Compiled some information about a topic but got little to no review and did not reach any audience
Researcher	Explored and summarized existing information about a topic and compiled it to share with others
Learner	Worked to become a subject matter expert in a specific area and applied that expertise to a problem
Innovator	Used the engineering design process to develop a new and unique solution to a problem

components are predictable and repeatable. The complicated domain similarly has predictable cause and effect relationships, but understanding them requires a domain expert. The complex domain has much more intertwined interactions that cannot easily be predicted or understood. These interactions can be measured and patterns can be found to gain a better understanding of the system, but those patterns might not hold true in the future. Finally, the chaotic domain has no cause and effect relationships, meaning virtually no information can be gained [99].

Bloom's Revised Taxonomy of Learning consists of six levels of learning: memorizing, understanding, applying, analyzing, evaluating, and creating [69]. The memorizing level can be demonstrated by activities like defining, reproducing, and listing. Understanding includes actions like describing, explaining, and extrapolating. The application level involves skills such as implementing, demonstrating, and calculating. Analysis can be demonstrated by comparing, contrasting, and examining. Evaluating includes activities like judging, defending, and assessing. Finally, the top level of creating can be demonstrated by constructing, designing, and developing [100].

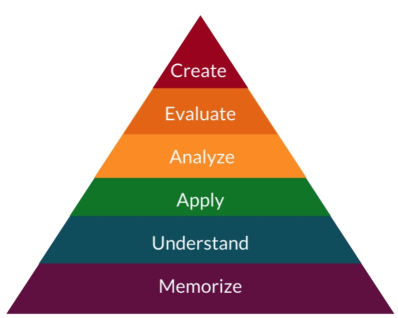

Figure 8.2. Bloom's Revised Taxonomy of Learning [69]

Webb's Depth of Knowledge is broken up into four levels: recalling and reproducing, applying basic knowledge and skills, thinking strategically, and thinking extensively [101]. DK1: Recalling and reproducing is a student's ability to remember definitions, formulas, and simple processes and procedures. All of the information they need to complete the task has already been provided to them. For DK2: Applying basic knowledge and skills, students are describing, explaining, and interpreting information. No complex reasoning is needed to answer these questions, but it does require students to take more than one step to solve the problem. DK3: thinking strategically requires reasoning and planning. Students working in DK3 are solving non-routine problems and proposing solutions to problems. Rather than just being able to explain a relationship, they are backing their explanations up with evidence and application of knowledge. DK3 is mapped to the complicated domain. For DK4: thinking extensively, students are relating multiple big variables and concepts in order to understand a topic at a deep level. Students must weigh options in order to decide the best way to approach the problem and make multiple decisions during their learning process. DK4 is mapped to the complex domain [101].

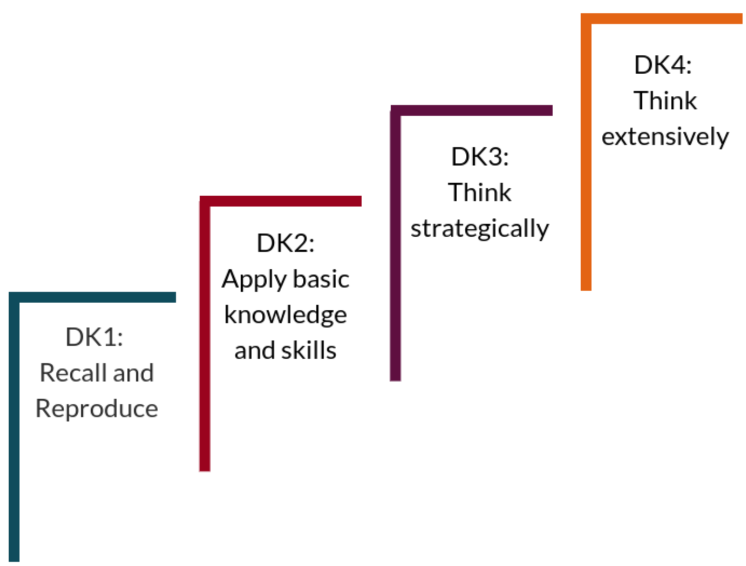

Figure 8.3. Webb's Depth of Knowledge [100]

8.3. Results

8.3.1. Clustering Results

Four clusters emerged from the data, and they were later named based off of the words that differentiated them from the other clusters. More information about how the clusters were named can be found in subsection 8.3.3. Of the 28 students, 13 were classified as Innovators, 5 were classified as Learners, 4 were classified as Surveyors, and 6 were classified as Surface Level. Figure 8.4 shows what clusters top performing and lower performing students fell into, and Figure 8.5 shows the cluster breakdown by year in school.

All Innovators were considered top performing students, meaning they had a high external value deliverable by the end of the semester. All Surface Level students were considered lower performing students. 4 of the 5 Learners were considered top performing students; the 5th did not complete any of their planned deliverables.

Undergraduate seniors fell into all 4 of the cluster categories; 6 were Surface Level, 3 were Surveyors, 3 were Learners, and 1 was an Innovator. Graduate students only fell into the Learner and Innovator categories; 12 were Innovators and 2 were Learners. 1 student did not provide a year in school, so that student was omitted from Figure 8.5.

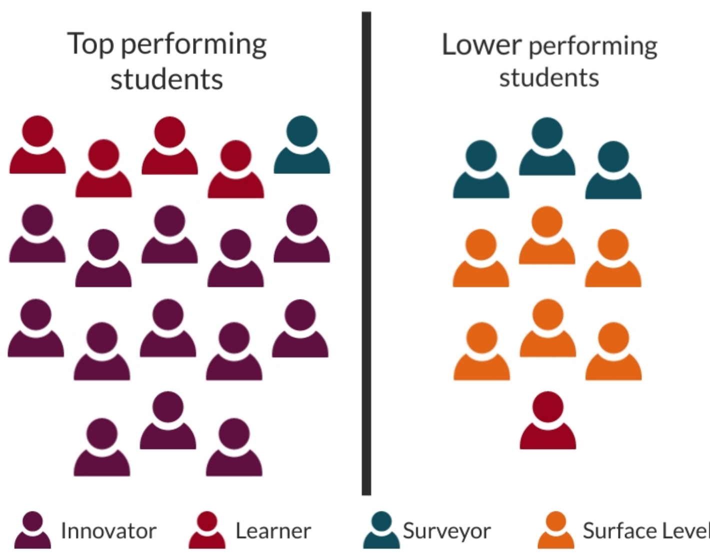

Figure 8.4. The number of students in each cluster in relation to their performance in the class. Top performing students were students that had a high external value deliverable during the course of the semester (e.g. peer-reviewed publication or presentation, invention disclosure, etc.

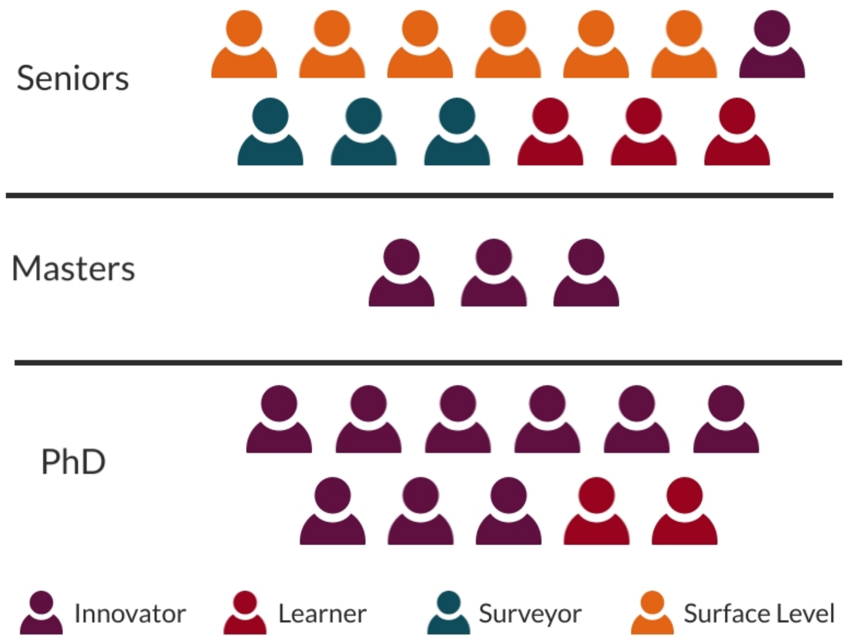

Figure 8.5. The number of students in each cluster in relation to their status in school.

8.3.2. Student Trajectory Results

The cluster trajectory for each student was mapped as shown in Figure 8.6. At the beginning of the semester, there was not enough logged information to group the students. By the third week, many students were starting to log learning objectives and deliverables to prepare for their first group in-class presentation. By the time of the presentation, about half of the students were already grouped into the cluster that they would stay in for the rest of the semester. By the halfway point of the semester, all but one student was classified into their final cluster.

Many of the groups had members that all ended up in the same cluster, and almost all groups had members that were either consistently high performers or consistently low performers. Only Group D had 3 members that were low performers and 1 member that was a high performer.

8.3.3. Extracting Most Pertinent Features Results

The Innovator cluster was differentiated by their use of the words *application, knowledge, analysis, data, symposium, literature, create, processing,* and *patent.* These words are tied closely to the high external value deliverables that the students created or the engineering design process, giving us the name of this cluster. The Learner cluster was differentiated by their use of the words *learning, course, concepts, study,* and *pitch.* Most of the Learners focused on learning the course content as well as completing online courses related to their topic. *Pitch* was a popular word

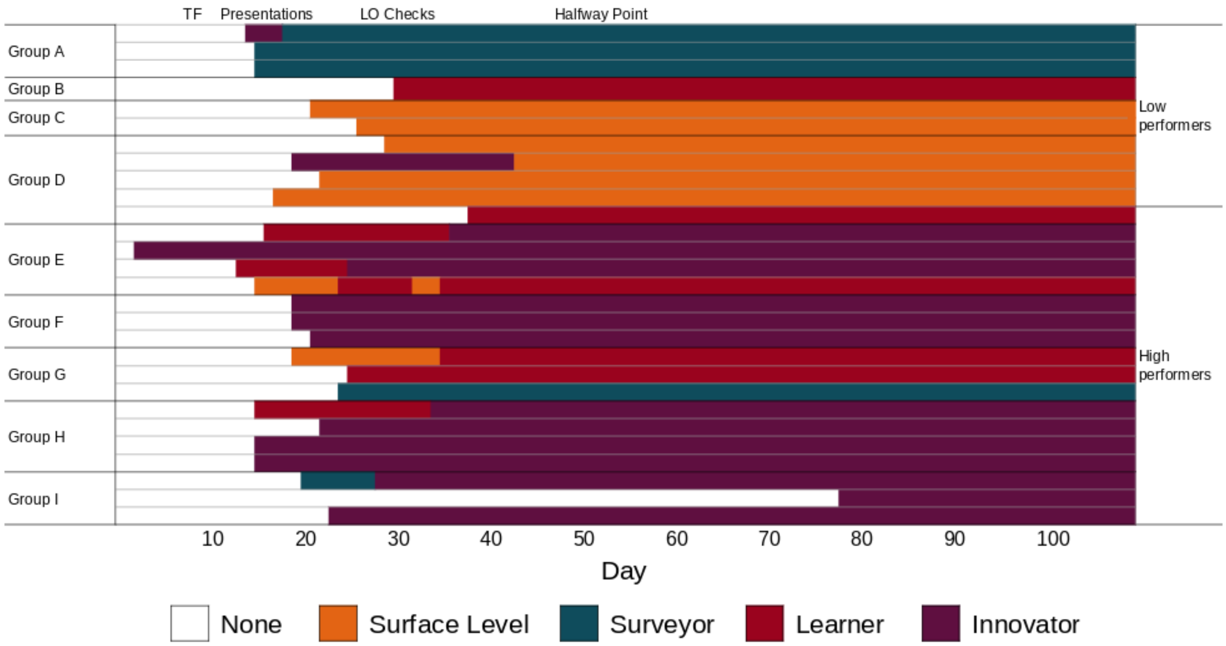

Figure 8.6. Student trajectory through the course. Each row corresponds to a different student, and students are grouped based off of their teams and their success in the course. Various milestones are marked at the top of the figure including team formation (TF), the 1st presentations that teams gave (Presentations), the first learning objective checks (LO Checks), and the midpoint in the semester (Halfway Point).

because some of the Learners competed in a business pitch competition which was often discussed in their learning objective and deliverable logging. The name for this cluster came directly from the presence of the word *learning*. The top words for Surveyors were *presentation, paper, research,* and *writing*. These students focused on reviewing and summarizing existing knowledge about a particular area and sharing it both through presentations and a research paper. Although the word *survey* did not appear in this cluster's learning objectives, the name was chosen to differentiate this group from those doing scientific research. The Surface Level cluster was differentiated by their use of the words *website, video, layout, analysis,* and, most strongly, *information*. These students found information about a topic and created websites or videos to compile their work. These students did not dive deep into their topic and did not share their learning more broadly, giving this cluster the Surface Level name. The words that differentiated most between each cluster appear in Figure 8.7.

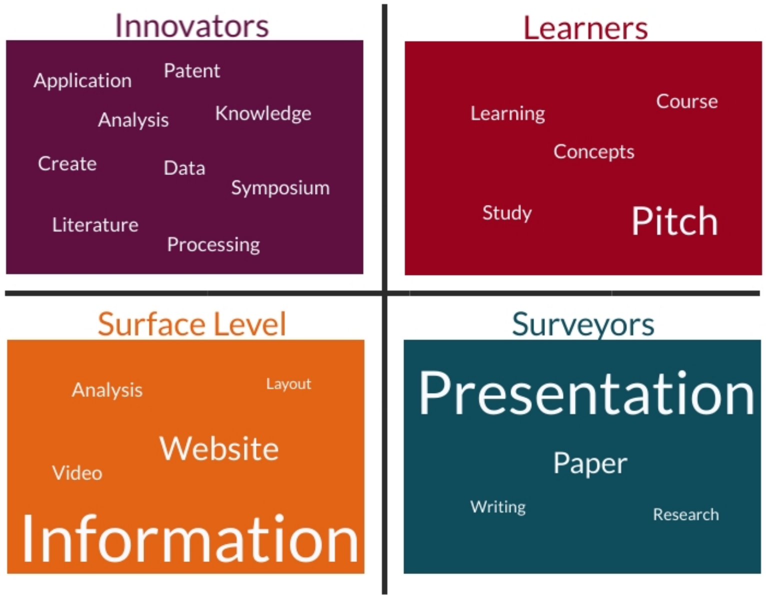

Figure 8.7. The top words that differentiated between each cluster. Larger words had larger Chi-Square values, meaning they more strongly differentiate students in that cluster from students in other clusters.

8.3.4. Comparing with Instructor Observations

Figure 8.8 shows how the instructor and algorithm classifications compare. The shaded diagonal represents cases where the algorithm and instructor classifications match. 18 of the 27 classifications matched (the student that did not complete any deliverables was not included in this chart because the instructors agreed there was not enough information to classify that student). The weighted Kappa between the algorithm and the instructors was 0.608. Kohen's Weighted Kappa of 0 means random agreement and Kappa of 1 means perfect agreement. Although there is no firmly agreed upon appropriate ranges of Kappa, 0.608 is considered moderate agreement by most experts [94].

After instructor interviews were completed and the mismatches were analyzed, it became clear that discrepancies occurred for two main reasons: 1) difficulties recognizing and classifying a learner, and 2) the algorithm's inability to tell quality of work. Reason 1 led to 7 differences in classification, and reason 2 led to 2 differences in classification. Difficulties in recognizing and classifying learners may have occurred because of the ambiguity of the word *learning*. It is hard to differentiate Surveyors from Learners because of the difficulty in comparing lower level and higher level learning, and it is hard to differentiate Learners from Innovators because the deliverables for both Learners and Innovators look similar. This problem also occurs when differentiating Webb's DK3: Thinking Strategically from Webb's DK2: Apply basic knowledge and skills and DK4: Thinking extensively [101].

The algorithm's inability to tell quality of work caused 2 students to be classified as Innovators by the algorithm and surface level by the instructors. Although the students were writing about the right things, they did not perform at the level that the instructors expected of them.

		Instructor Classifcations			
		Surface Level	Surveyor	Learner	Innovator
Algorithm Classifications	Surface Level	5		1	
	Surveyor		2	2	
	Learner			1	3
	Innovator	2		1	10

Figure 8.8. Comparison of algorithm classification and instructors classification. The number in each box is the number of students that fell into that classification. The shaded boxes along the diagonal represent matches between the algorithm and the instructors. Non-shaded boxes are mismatches between the algorithm and the instructors.

8.4. Discussion

8.4.1. Cluster Descriptions and Examples

By looking at the top words that differentiated between each cluster and at each student's learning objectives and deliverables, we were able to understand more about each cluster and map them to Bloom's Revised Taxonomy, Webb's Depth of Knowledge framework, and the Cynefin Framework. Bloom's Revised Taxonomy and Webb's Depth of Knowledge framework both serve as ways to map the level of learning that the students in the clusters reach. The Cynefin framework details how the students in each of the clusters defines the problem at hand; students can approach the problem as if it were simple, complicated, or complex. This mapping is shown in Figure 8.9.

- *Surface Level*: Surface Level students compiled some information about a topic but did not reach an audience or get review on their work. Students in this cluster started with innovative ideas, but eventually moved to creating websites and videos that summarized existing resources. These deliverables were not classified as high impact because the videos and websites had little to no visits or reviews. Rather than focusing on learning and gaining knowledge, these students focused on collecting information, mapping them to Webb's DK1: recalling and reproducing. For Bloom's Taxonomy, these learners fall somewhere between memorizing and understanding; their deliverables show that they were able to reiterate existing information, but they do not show direct evidence of understanding. An example of a group of students that fell into this cluster was one that chose a topic, read a handful of resources, and put the information they read onto a website. These students are are mapped to the simple domain because they explored only the certainties of their problem.

- *Surveyor*: Surveyors explored and summarized existing information about a topic and compiled it to share with others. It is important to note that the Surveyor cluster is not performing original scientific research. Rather, they are reviewing and summarizing existing work in order to share it with a wide range of audiences. The main difference between this group and the Surface Level cluster is that these students dug deeper into their topic and made an effort to share their work with others, mapping them to Webb's DK2: applying basic knowledge and skills. Because they were able to share their work for a variety of audiences and contexts, they showed evidence of achieving the Understanding level of Bloom's. An example of a group with students in this category was one that chose a broad topic, reviewed existing resources, and gave a presentation about their findings. This differed from the website because the students had to explain their content and answer questions. This cluster also approached the course in the context of the simple domain of Cynefin. They gained a broad understanding of a topic, but they did not contribute new information or ideas to the field.

- *Learner*: Learners became content experts in a specific area to complete their projects. Learners differ from Surveyors because the Learners dive deep into one topic rather than learning about something more broadly. These students reached Webb's DK3: thinking strategically because they used their learning to solve a problem. In addition, they have applied their new skills and knowledge to make contributions to the field, mapping them to the Bloom's Apply and Analyze levels. Most of the students in the Learner cluster did show an ability to innovate, but they rooted their work more deeply in their learning. Some of the students that fell into this cluster worked on a project where they used an Application Programming Interface (API) to add new features to an existing system. They added new value to the system, but they could use existing resources to work step-by-step through the project. Learners approach the course within the complicated domain of the Cynefin framework. They aim to have expert understanding of a topic in order to succeed and add value.

- *Innovator*: Innovators are defined by their work to solve a problem with no clear answer. In order to work in this domain, the students must be able to devise a solution, test it, and use the information gained to improve the solution. Having an expert understanding alone cannot lead to a perfect solution. Rather, Innovators must come at the problem from multiple ways to better understand the project and improve a solution, mapping them to Webb's DK4: thinking extensively. Innovators reach the Bloom's levels of Evaluate and Create because they are evaluating potential solutions in order to better understand the problem and create a new one. An example of a group of Innovators worked on a new wearable sensor. The students needed to combine knowledge from multiple areas in order to create a product. Innovators approach the course from the complex domain; they understand that there is no clear or straightforward answer, but they are able to determine what they will be able to learn in order to move towards a better understanding of the problem and work towards a possible solution.

<table>
<tr><td>Cluster Name</td><td colspan="2">Surface Level</td><td colspan="2">Surveyors</td><td colspan="2">Learners</td><td colspan="2">Innovators</td></tr>
<tr><td>Cynefin Framework</td><td colspan="2">Simple</td><td colspan="2">Simple</td><td colspan="2">Complicated</td><td colspan="2">Complex</td></tr>
<tr><td>Bloom's Taxonomy of Learning</td><td colspan="2">Memorize</td><td colspan="2">Understand</td><td>Apply</td><td>Analyze</td><td>Evaluate</td><td>Create</td></tr>
<tr><td>Webb's Depth of Knowledge</td><td colspan="2">DK1: Recalling and reproducing</td><td colspan="2">DK2: Applying basic knowledge and skills</td><td colspan="2">DK3: Thinking strategically</td><td colspan="2">DK4: Thinking extensively</td></tr>
</table>

Figure 8.9. Clusters and their mapping to the Cynefin Framework, Bloom's Taxonomy of Learning, and Webb's Depth of Knowledge.

8.4.2. Insights Gained

By exploring the breakdown of clusters, it is clear that Innovators and Learners were more likely to complete a high external value deliverable during the course. In addition, all Surveyors and Surface Level students were undergraduates. This illustrates that undergraduate students might need additional support to grow into Learners or Innovators. The three undergraduate Learners and one undergraduate Innovator all were in groups that had both undergraduate and graduate students, potentially adding to their success. In the future, it may be wise to encourage more groups that mix both undergraduate and graduate students together to help foster growth, but questions still remain about why mixed groups found more success. Top undergraduates may be more likely to choose complex projects or join up with graduate students, for example.

From the trajectory results, we see that it is possible for students to switch to a different cluster during the semester, especially if the majority of their team is in that cluster. Because it is possible to switch clusters, the instructor can play a role in providing guidance for students in the Surface Level and Surveyor clusters. By being able to categorize students (either by observation or by using the trained classifier), instructors can try to better guide students into the Learner or Innovator clusters.

The trajectory results also illustrate the importance of providing student feedback early and often during the semester. By the midpoint in the semester, most students were already settled into the cluster they would stay in. Rather than waiting until later in the semester, instructors should try to provide a formal review sometime during the second quarter of the semester to ensure the students have time to adjust their plans if needed. In addition, it is possible that more feedback near the end of the semester might have caused students to switch clusters later in the semester.

8.4.3. Limitations

Two of the major limitations of this work are the algorithm's inability to understand the context of words and its inability to understand the quality of the deliverables being created. For example, if a student uses the word *research* in their learning objectives, the algorithm cannot tell the difference between *survey and summarize research* and *scientific research*. Similarly, if a student has a firm understanding of the process of the class and is able to write learning objectives and deliverables that have clear high external value, the algorithm will not be able to tell if a student is actually creating high quality work. Therefore, this tool is not designed to take the place of instructor review, but rather to supplement it.

In addition, because this data is only from one semester, the developed clusters and classifier model may not perform similarly in future semesters. Therefore, the goal is not to develop one perfect model, but rather continue to explore how the model might change over time. Important insights into improving engineering education may be found by looking into these changes.

8.5. Conclusion

Clustering students by using their learning objective data provides new insights about how students navigate an Innovation-Based Learning course. Four main clusters emerged from this data set: Surface Level, Surveyors, Learners, and Innovators, each with their own words that differentiate them from other clusters. Students can switch clusters during the semester, and many of those switches seem to be related to group behavior. This work could lead to better understanding of how students innovate and solve problems, allowing for advancements in personalized education, group matchmaking, and even assessment. By implementing these tools, these education models can be scaled up, allowing more students to grow in their ability to work within the complex domain. They develop problem solving, adaptability, and creativity, helping them tackle big problems and become an Engineer of 2020.

9. DISCUSSION

9.1. Insights

One of the most encouraging recurring themes in the data analysis is growth. The clustering results show that students can change clusters during the semester to switch from a Surface Level or Surveyor to become a Learner or Innovator. That being said, this directly illustrates the importance of providing feedback early and often during the semester in order to get students on track to make this change. The classification and clustering results show that these students did not change clusters or classifications after the midpoint in the semester. Therefore, feedback should be given early in the semester, but continued throughout.

Another takeaway is the potential benefit of having diversity in groups. We saw that groups made up entirely of undergraduate students consisted of almost all Surface Level and Surveyors, whereas mixed groups consisted of mostly Learners and Innovators. By diversifying these groups, hopefully group members can help each other move to the Learner or Innovator categories.

Finally, we saw a recurring theme of how the same word can have different meanings, confusing both students and our algorithms. For example, in Chapter 6, we saw that one of the issues with assessment was the differing definitions for "research paper", and in Chapter 8, the difference between scientific research and surveying research was discussed. By discussing some of the possible deliverables and what the expectations are for those deliverables at the beginning of the semester, students and instructors will be working with a more similar understanding of terms throughout the course.

For future semesters, this work led us to three recommendations: 1) provide both informal and formal feedback before the midpoint in the semester, 2) encourage groups to mix both undergraduate and graduate students, and 3) spend more time at the beginning of the semester making sure student understand the expectations for different types of deliverables. By implementing these recommendations, hopefully more students will be able to make the shift to a Learner or Innovator.

9.2. Limitations

Just as the world around us is transforming every day, so are our students. Therefore, these models will need to continue to evolve and improve as students change their approach to the class.

Like with many of the complex problems presented in Chapter 2, our models need to be updated as more information becomes available and as student populations continue to grow and adapt. Aiming for consistently high performing models is not a realistic goal for this work. Rather, we can use the knowledge discovery from these models to better understand how students move through these environments and aim to better support them. The number of students used in this study was small, so it is still to early to make significant claims or to generalize our findings to other contexts. However, there does seem to be potential in using EDM to better understand how IBL students move through the model.

There are also some limitations due to the models and feature sets that were used. Because text data showed the most promise in classifying students, our final models for both classification and clustering focused on the text features. As of now, these models only accounted for the number of times each word was mentioned, meaning the models do not account for the context the words were used in. In addition, there are other factors that are not yet included such as as the quality of deliverables uploaded, group dynamics, and temporal components (e.g. the order in which deliverables were completed, when group presentations occurred, etc.) Many of these components are things an instructor would know, reiterating the importance of being able to harness both the power of the algorithm and the wisdom of the instructor.

9.3. Future Work

Moving forward, we have identified three main future directions: collecting more learning objective data, exploring the use of other machine learning tasks, and diving deeper into individual students' experiences.

In upcoming iterations of the course, we are planning to continue to collect data to see how well our current models perform with new students. By having a larger sample size, we can continue to improve our models and identify patterns in student trajectory. If these pathways are better understood, instructors could recommend specific next steps that help students stay on track while staying true to their own learning goals. In addition, collecting data from students at other universities will reveal how well our models transfer to other student populations. The universities in this study were both public research universities, but future work could include students from private universities or non-traditional engineering programs.

Another possible avenue for future work is to explore the use of association analysis, sequential pattern analysis, and process analysis. These techniques could allow for more patterns to be identified and better understanding of student behavior at the temporal level. Many EDM researchers use a hybrid approach (e.g. performing cluster analysis to find similar students and then performing sequential pattern analysis to find commonalities in each cluster as seen in [48]). Exploring hybrid approaches may also help uncover other patterns in the data.

Finally, more work can be done to dive deeper into individual students to better understand the context of their work. For example, it might be insightful to do an interview with a student that switches clusters during the semester to see if they had any changes in attitude, understanding of the course, group dynamics, etc. These interviews can be coded by researchers or analyzed by a program like SenseMaker, a tool that analyzes narratives in order to find recurring themes [102].

As with any Grand Challenge, it is imperative to explore the problem from multiple angles in order to fully understand the domain you're working in. The analysis completed so far has shown great promise, but collecting more data, exploring new machine learning techniques, and exploring other forms of data collection will help us get a full picture of how engineering students tackle problems and innovate new solutions.

10. CONCLUSION

This work demonstrated the potential for using classification and clustering models to understand more about students in an Innovation-Based Learning course. A support vector machine classification model that used text written in students' learning objectives and deliverables was developed that achieved over 80% accuracy and ROC AUC of over 0.95 by the midpoint in the semester. Words more likely to be included in objectives written by successful students included words that were directly related to high external value deliverables (e.g. *symposium*, *abstract*, and *poster*) and words related to the engineering design process (e.g. *idea*, *data*, and *develop*). Words more likely to be found in the learning objectives of less successful students included *website*, *information*, and *presentations*.

A hierarchical clustering model grouped students into four main groups that were then named Innovators, Learners, Surveyors, and Surface Level. Innovators were more likely to use words such as *create*, *data*, *patent*, and *symposium*. Learners were more likely to use words such as *learning*, *study*, *course*, and *concepts*. Surveyors were more likely to use words such as *presentation*, *paper*, *writing*, and *research*. Surface Level students were more likely to include *website*, *information*, and *video* in their learning objectives. About 1/4 of the students changed clusters during the semester, and all but 1 student remained in the same cluster from the midpoint in the semester to the end.

Although the IBL data is a complex problem, patterns and information were still able to be found and extracted using these algorithms. These insights can be used to improve future iterations of the course and potentially to better understand how engineering students innovate and work on projects in other contexts. By improving these experiences, we are helping students develop the skills that engineering organizations and professionals are calling for such as the ability to solve complex problems, communicate, work on a team, think like an entrepreneur, and practice lifelong learning.

Not only are we working to solve the Engineering Grand Challenge of advancing personalized learning, but we are also working to better educate our engineering students, preparing them to take on the Engineering Grand Challenges and any other challenge that comes their way.

REFERENCES

[1] N. A. of Engineering, *NAE Grand Challenges for Engineering.* National Academies Press, 2008.

[2] ABET, "Criteria for accrediting engineering programs." [Online]. Available: https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2016-2017/#GC3

[3] N. A. of Engineering, *The engineer of 2020: Visions of engineering in the new century.* National Academies Press, 2004.

[4] H. J. Passow and C. H. Passow, "What competencies should undergraduate engineering programs emphasize? a systematic review," *Journal of Engineering Education*, vol. 106, no. 3, pp. 475–526, 2017.

[5] E. Swartz, M. Pearson, R. Striker, L. Singelmann, and E. Alvarez Vazquez, "Innovation-based learning on a massive scale," in *6th International Conference on Learning with MOOCs.* IEEE, 2019.

[6] B. Davis and D. Sumara, "'if things were simple...': complexity in education," *Journal of Evaluation in Clinical Practice*, vol. 16, no. 4, pp. 856–860, 2010.

[7] B. Davis, "Complexity and education: Vital simultaneities," *Educational Philosophy and Theory*, vol. 40, no. 1, pp. 50–65, 2008.

[8] P. Cilliers, *Complexity and postmodernism: Understanding complex systems.* routledge, 2002.

[9] D. S. Bassett and M. S. Gazzaniga, "Understanding complexity in the human brain," *Trends in cognitive sciences*, vol. 15, no. 5, pp. 200–209, 2011.

[10] G. H. Fernald, E. Capriotti, R. Daneshjou, K. J. Karczewski, and R. B. Altman, "Bioinformatics challenges for personalized medicine," *Bioinformatics*, vol. 27, no. 13, pp. 1741–1748, 2011.

[11] J. N. Galloway, A. R. Townsend, J. W. Erisman, M. Bekunda, Z. Cai, J. R. Freney, L. A. Martinelli, S. P. Seitzinger, and M. A. Sutton, "Transformation of the nitrogen cycle: recent trends, questions, and potential solutions," *Science*, vol. 320, no. 5878, pp. 889–892, 2008.

[12] L. Kuhn, "Complexity and educational research: A critical reflection," *Educational Philosophy and Theory*, vol. 40, no. 1, pp. 177–189, 2008.

[13] U. of Colorado Boulder, "Engineering design process." [Online]. Available: https://www.teachengineering.org/k12engineering/designprocess

[14] N. H. Sabelli, "Complexity, technology, science, and education," *The Journal of the learning sciences*, vol. 15, no. 1, pp. 5–9, 2006.

[15] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, pp. 3–24, 2007.

[16] O. Maimon and L. Rokach, "Data mining and knowledge discovery handbook," 2005.

[17] B. Marr, "The top 10 ai and machine learning use cases everyone should know about," Sep 2016.

[18] D. Ravì, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G.-Z. Yang, "Deep learning for health informatics," *IEEE journal of biomedical and health informatics*, vol. 21, no. 1, pp. 4–21, 2016.

[19] A. L. Beam and I. S. Kohane, "Big data and machine learning in health care," *Jama*, vol. 319, no. 13, pp. 1317–1318, 2018.

[20] R. Fang, S. Pouyanfar, Y. Yang, S.-C. Chen, and S. Iyengar, "Computational health informatics in the big data age: a survey," *ACM Computing Surveys (CSUR)*, vol. 49, no. 1, pp. 1–36, 2016.

[21] P. Lauret, C. Voyant, T. Soubdhan, M. David, and P. Poggi, "A benchmarking of machine learning techniques for solar radiation forecasting in an insular context," *Solar Energy*, vol. 112, pp. 446–457, 2015.

[22] N. Sharma, P. Sharma, D. Irwin, and P. Shenoy, "Predicting solar generation from weather forecasts using machine learning," in *2011 IEEE international conference on smart grid communications (SmartGridComm)*. IEEE, 2011, pp. 528–533.

[23] H. A. Kazem, J. H. Yousif, and M. T. Chaichan, "Modeling of daily solar energy system prediction using support vector machine for oman," *International Journal of Applied Engineering Research*, vol. 11, no. 20, pp. 10 166–10 172, 2016.

[24] A. M. Sattar, Ö. F. Ertuğrul, B. Gharabaghi, E. A. McBean, and J. Cao, "Extreme learning machine model for water network management," *Neural Computing and Applications*, vol. 31, no. 1, pp. 157–169, 2019.

[25] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, vol. 29, no. 2, pp. 196–210, 2015.

[26] S. K. Sinha and M. A. Knight, "Intelligent system for condition monitoring of underground pipelines," *Computer-Aided Civil and Infrastructure Engineering*, vol. 19, no. 1, pp. 42–53, 2004.

[27] F. Zantalis, G. Koulouras, S. Karabetsos, and D. Kandris, "A review of machine learning and iot in smart transportation," *Future Internet*, vol. 11, no. 4, p. 94, 2019.

[28] J. Whyte, "The future of systems integration within civil infrastructure: A review and directions for research," in *INCOSE International Symposium*, vol. 26, no. 1. Wiley Online Library, 2016, pp. 1541–1555.

[29] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications surveys & tutorials*, vol. 18, no. 2, pp. 1153–1176, 2015.

[30] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35 365–35 381, 2018.

[31] J. H. Flavell, "The developmental psychology of jean piaget." 1963.

[32] A. Y. Kolb and D. A. Kolb, "Learning styles and learning spaces: Enhancing experiential learning in higher education," *Academy of management learning & education*, vol. 4, no. 2, pp. 193–212, 2005.

[33] J. Dewey, "Experience and education," in *The Educational Forum*, vol. 50, no. 3. Taylor & Francis Group, 1986, pp. 241–252.

[34] R. S. Baker and K. Yacef, "The state of educational data mining in 2009: A review and future visions," *JEDM— Journal of Educational Data Mining*, vol. 1, no. 1, pp. 3–17, 2009.

[35] C. Romero and S. Ventura, "Educational data mining: a review of the state of the art," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 6, pp. 601–618, 2010.

[36] R. S. Baker and P. S. Inventado, "Educational data mining and learning analytics," in *Learning analytics*. Springer, 2014, pp. 61–75.

[37] C. Romero, S. Ventura, M. Pechenizkiy, and R. S. Baker, *Handbook of educational data mining*. CRC press, 2010.

[38] A. Dutt, M. A. Ismail, and T. Herawan, "A systematic review on educational data mining," *Ieee Access*, vol. 5, pp. 15 991–16 005, 2017.

[39] W. Van Der Aalst, "Process mining: Overview and opportunities," *ACM Transactions on Management Information Systems (TMIS)*, vol. 99, no. 99, 2012.

[40] "Educational data mining 2020: Call for papers." [Online]. Available: http://educationaldatamining.org/edm2020/

[41] S. M. Land, "Cognitive requirements for learning with open-ended learning environments," *Educational Technology Research and Development*, vol. 48, no. 3, pp. 61–78, 2000.

[42] P. Pacharn, D. Bay, and S. Felton, "The impact of a flexible assessment system on students' motivation, performance and attitude," *Accounting Education*, vol. 22, no. 2, pp. 147–167, 2013.

[43] P. R. Pintrich and E. V. De Groot, "Motivational and self-regulated learning components of classroom academic performance." *Journal of educational psychology*, vol. 82, no. 1, p. 33, 1990.

[44] J. R. Segedy, J. S. Kinnebrew, and G. Biswas, "Using coherence analysis to characterize self-regulated learning behaviours in open-ended learning environments," *Journal of Learning Analytics*, vol. 2, no. 1, pp. 13–48, 2015.

[45] P. H. Winne and R. S. Baker, "The potentials of educational data mining for researching metacognition, motivation and self-regulated learning," *JEDM— Journal of Educational Data Mining*, vol. 5, no. 1, pp. 1–8, 2013.

[46] P. Blikstein and M. Worsley, "Multimodal learning analytics and education data mining: using computational technologies to measure complex learning tasks," *Journal of Learning Analytics*, vol. 3, no. 2, pp. 220–238, 2016.

[47] B. Busteed, "Online education: From good to better to best?" Mar 2019.

[48] F. Bouchet, R. Azevedo, J. S. Kinnebrew, and G. Biswas, "Identifying students' characteristic learning behaviors in an intelligent tutoring system fostering self-regulated learning." *International Educational Data Mining Society*, 2012.

[49] J. S. Kinnebrew, K. M. Loretz, and G. Biswas, "A contextualized, differential sequence mining method to derive students' learning behavior patterns," *JEDM— Journal of Educational Data Mining*, vol. 5, no. 1, pp. 190–219, 2013.

[50] J. Cuadros, L. González-Sabaté, S. Romero, M. L. Guenaga, J. G. Zubía, and P. Orduña, "Educational data mining in an open-ended remote laboratory on electric circuits. goals and preliminary results." in *EDM*, 2015, pp. 578–579.

[51] M. Arseneault, "Applying machine learning the assessment of problem-solving skills," 2019.

[52] I. Arroyo and B. P. Woolf, "Inferring learning and attitudes from a bayesian network of log file data." in *AIED*, 2005, pp. 33–40.

[53] P. I. Pavlik Jr, "Mining the dynamics of student utility and strategy use during vocabulary learning," *JEDM— Journal of Educational Data Mining*, vol. 5, no. 1, pp. 39–71, 2013.

[54] Y. Mao, R. Zhi, F. Khoshnevisan, T. Price, T. Barnes, and M. Chi, "One minute is enough: Early prediction of student success and event-level difficulty during a novice programming task," *The 12th International Conference on Educational Data Mining*, pp. 119–128, 2019.

[55] L. Wang, A. Sy, L. Liu, and C. Piech, "Learning to represent student knowledge on programming exercises using deep learning." *International Educational Data Mining Society*, 2017.

[56] P. Ihantola, A. Vihavainen, A. Ahadi, M. Butler, J. Börstler, S. H. Edwards, E. Isohanni, A. Korhonen, A. Petersen, K. Rivers *et al.*, "Educational data mining and learning analytics in programming: Literature review and case studies," in *Proceedings of the 2015 ITiCSE on Working Group Reports*, 2015, pp. 41–63.

[57] R. Zhi, T. W. Price, N. Lytle, Y. Dong, and T. Barnes, "Reducing the state space of programming problems through data-driven feature detection," in *Educational Data Mining in Computer Science Education (CSEDM) Workshop@ EDM*, 2018.

[58] T. W. Price, Y. Dong, and T. Barnes, "Generating data-driven hints for open-ended programming." *International Educational Data Mining Society*, 2016.

[59] I. E. Harel and S. E. Papert, *Constructionism.* Ablex Publishing, 1991.

[60] S. Han and K. Bhattacharya, "Constructionism, learning by design, and project based learning," *Emerging perspectives on learning, teaching, and technology. Retrieved April*, vol. 29, p. 2007, 2001.

[61] M. Berland, R. S. Baker, and P. Blikstein, "Educational data mining and learning analytics: Applications to constructionist research," *Technology, Knowledge and Learning*, vol. 19, no. 1-2, pp. 205–220, 2014.

[62] C. Vieira, M. H. Goldstein, Ş. Purzer, and A. J. Magana, "Using learning analytics to characterize student experimentation strategies in the context of engineering design," *Journal of Learning Analytics*, vol. 3, no. 3, pp. 291–317, 2016.

[63] C. Vieira, Y. Y. Seah, and A. J. Magana, "Students' experimentation strategies in design: Is process data enough?" *Computer Applications in Engineering Education*, vol. 26, no. 5, pp. 1903–1914, 2018.

[64] N. E. Perry and P. H. Winne, "Learning from learning kits: gstudy traces of students' self-regulated engagements with computerized content," *Educational Psychology Review*, vol. 18, no. 3, pp. 211–228, 2006.

[65] D. Spikol, E. Ruffaldi, and M. Cukurova, "Using multimodal learning analytics to identify aspects of collaboration in project-based learning." Philadelphia, PA: International Society of the Learning Sciences., 2017.

[66] D. Spikol, E. Ruffaldi, L. Landolfi, and M. Cukurova, "Estimation of success in collaborative learning based on multimodal learning analytics features," in *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*. IEEE, 2017, pp. 269–273.

[67] D. Scaradozzi, L. Cesaretti, L. Screpanti, and E. Mangina, "Identification of the students learning process during education robotics activities," *Frontiers in Robotics and AI*, vol. 7, p. 21, 2020.

[68] L. Singelmann, E. Swartz, M. Pearson, R. Striker, and E. Alvarez Vazquez, "Design and development of a machine learning tool for an innovation-based learning mooc," © 2019 IEEE. Reprinted, with permission.

[69] D. R. Krathwohl and L. W. Anderson, *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives.* Longman, 2009.

[70] F. Murtagh and P. Legendre, "Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion?" *Journal of classification*, vol. 31, no. 3, pp. 274–295, 2014.

[71] T. Oliphant, "NumPy: A guide to NumPy," USA: Trelgol Publishing, 2006–. [Online]. Available: http://www.numpy.org/

[72] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau,

M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[73] T. pandas development team, “pandas-dev/pandas: Pandas,” Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[74] E. Loper and S. Bird, “Nltk: the natural language toolkit,” *arXiv preprint cs/0205028*, 2002.

[75] S. Slater, S. Joksimović, V. Kovanovic, R. S. Baker, and D. Gasevic, “Tools for educational data mining: A review,” *Journal of Educational and Behavioral Statistics*, vol. 42, no. 1, pp. 85–106, 2017.

[76] S. K. Yadav and S. Pal, “Data mining: A prediction for performance improvement of engineering students using classification,” *arXiv preprint arXiv:1203.3832*, 2012.

[77] E. Yukselturk, S. Ozekes, and Y. K. Türel, “Predicting dropout student: an application of data mining methods in an online education program,” *European Journal of Open, Distance and e-learning*, vol. 17, no. 1, pp. 118–133, 2014.

[78] D. Kabakchieva, “Predicting student performance by using data mining methods for classification,” *Cybernetics and information technologies*, vol. 13, no. 1, pp. 61–72, 2013.

[79] R. Kohavi *et al.*, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Ijcai*, vol. 14, no. 2. Montreal, Canada, 1995, pp. 1137–1145.

[80] L. Singelmann, E. Alvarez Vazquez, E. Swartz, M. Pearson, and R. Striker, “Student-developed learning objectives: A form of assessment to promote professional growth,” in *American Society for Engineering Education Annual Conference*. © 2020 American Society for Engineering Education. Reprinted, with permission.

[81] J. Looney, “Assessment and innovation in education,” 2009.

[82] J. Biggs, “What the student does: teaching for enhanced learning,” *Higher Education Research & Development*, vol. 31, no. 1, pp. 39–55, 2012.

[83] A. Cook, “Assessing the use of flexible assessment,” *Assessment & Evaluation in Higher Education*, vol. 26, no. 6, pp. 539–549, 2001.

[84] T. Wanner and E. Palmer, “Personalising learning: Exploring student and teacher perceptions about flexible learning and assessment in a flipped university course,” *Computers & Education*, vol. 88, pp. 354–369, 2015.

[85] R. A. Francis, “An investigation into the receptivity of undergraduate students to assessment empowerment,” *Assessment & Evaluation in Higher Education*, vol. 33, no. 5, pp. 547–557, 2008.

[86] J. F. Pane, E. D. Steiner, M. D. Baird, and L. S. Hamilton, “Continued progress: Promising evidence on personalized learning.” *RAND Corporation*, 2015.

[87] B. Irwin and S. Hepplestone, “Examining increased flexibility in assessment formats,” *Assessment & Evaluation in Higher Education*, vol. 37, no. 7, pp. 773–785, 2012.

[88] J. Casey and P. Wilson, “A practical guide to providing flexible learning in further and higher education,” *Quality Assurance Agency for Higher Education Scotland, Glasgow. Available online at: http//www. enhancementthemes. ac. uk/documents/flexibleDelivery/FD_Flexible_ Learning_JCaseyFINALWEB. pdf [accessed 15 November 2010]*, 2005.

[89] D. Boud, F. Dochy *et al.*, “Assessment 2020. seven propositions for assessment reform in higher education,” 2010.

[90] D. H. Schunk, “Metacognition, self-regulation, and self-regulated learning: Research recommendations,” *Educational psychology review*, vol. 20, no. 4, pp. 463–467, 2008.

[91] M. Pearson, E. Swartz, R. Striker, E. Alvarez Vazquez, and L. Singelmann, “Driving change using moocs in a blended and online learning environment,” in *6th International Conference on Learning with MOOCs*. IEEE, 2019.

[92] E. Alvarez Vazquez, M. Pearson, L. Singelmann, R. Striker, and E. Swartz, “Federal funding opportunity announcements as a catalyst of students’ projects in mooc environments,” in *6th International Conference on Learning with MOOCs*. IEEE, 2019.

[93] R. Stiker, M. Pearson, E. Swartz, L. Singelmann, and E. Alvarez Vazquez, "21st century syllabus: Aggregating electronic resources for innovation based learning," in *6th International Conference on Learning with MOOCs*. IEEE, 2019.

[94] M. L. McHugh, "Interrater reliability: the kappa statistic," *Biochemia medica: Biochemia medica*, vol. 22, no. 3, pp. 276–282, 2012.

[95] K. Jaeger-Helton, B. Smyser, and H. McNamus, "Capstone prepares engineers for the real world, right? abet outcomes and student perceptions," in *2019 ASEE Annual Conference & Exposition*. ASEE, 2019.

[96] L. Singelmann, E. Alvarez Vazquez, E. Swartz, R. Stiker, M. Pearson, and D. Ewert, "Predicting and understanding success in an innovation-based learning course," in *Educational Data Mining 2020 Conference*. © 2020 IEEE. Reprinted, with permission.

[97] C. Romero and S. Ventura, "Data mining in education," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, no. 1, pp. 12–27, 2013.

[98] L. Singelmann, E. Alvarez Vazquez, E. Swartz, R. Stiker, M. Pearson, and D. Ewert, "Innovators, learners, and surveyors: Clustering students in an innovation-based learning course," in *IEEE Frontiers in Education 2020 Conference*. ASEE, Accepted for publication in 2020.

[99] C. F. Kurtz and D. J. Snowden, "The new dynamics of strategy: Sense-making in a complex and complicated world," *IBM systems journal*, vol. 42, no. 3, pp. 462–483, 2003.

[100] C. J. Stanny, "Reevaluating bloom's taxonomy: What measurable verbs can and cannot say about student learning," *Education Sciences*, vol. 6, no. 4, p. 37, 2016.

[101] N. L. Webb, "Depth-of-knowledge levels for four content areas," *Language Arts*, vol. 28, no. March, 2002.

[102] Cognitive Edge, "Sensemaker." [Online]. Available: https://sensemaker.cognitive-edge.com/

APPENDIX

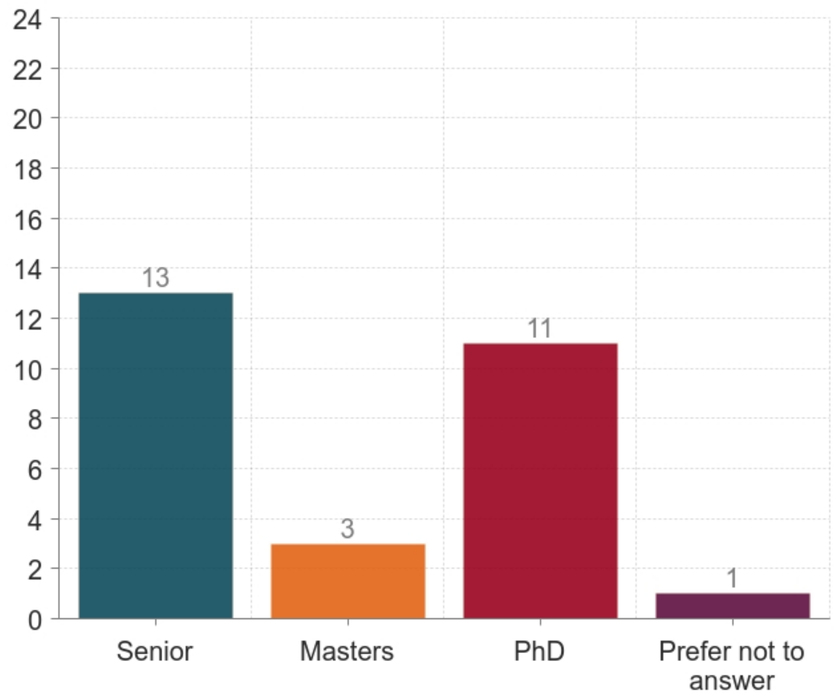

Figure A.1. Level of school

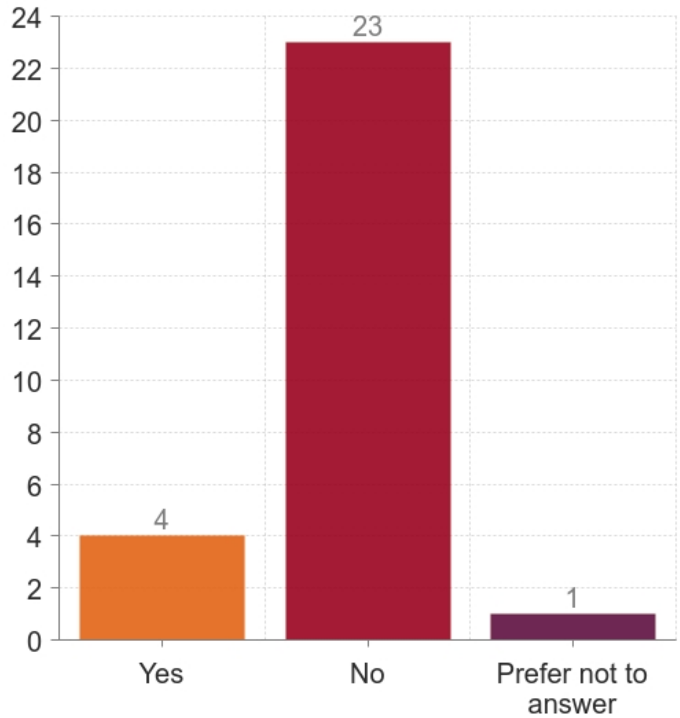

Figure A.2. First generation college student

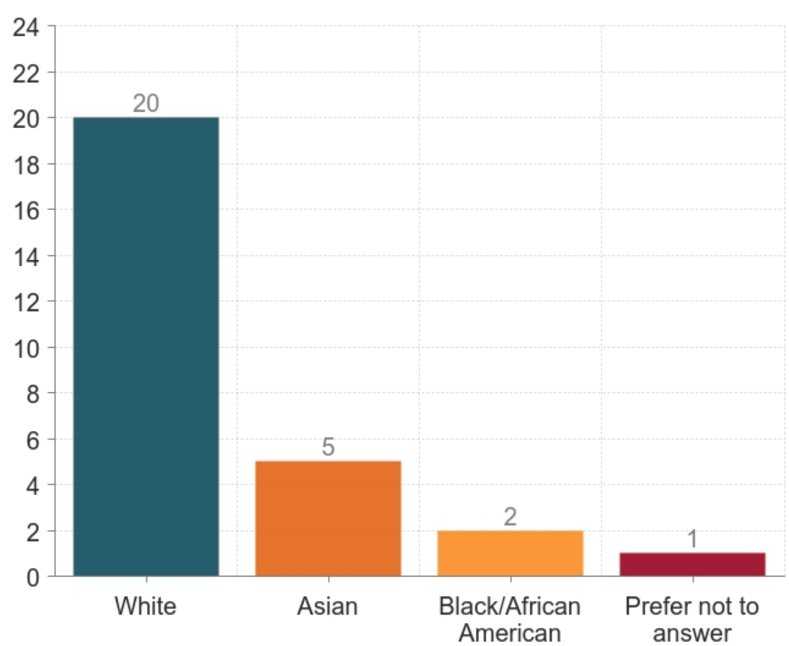

Figure A.3. Ethnicity

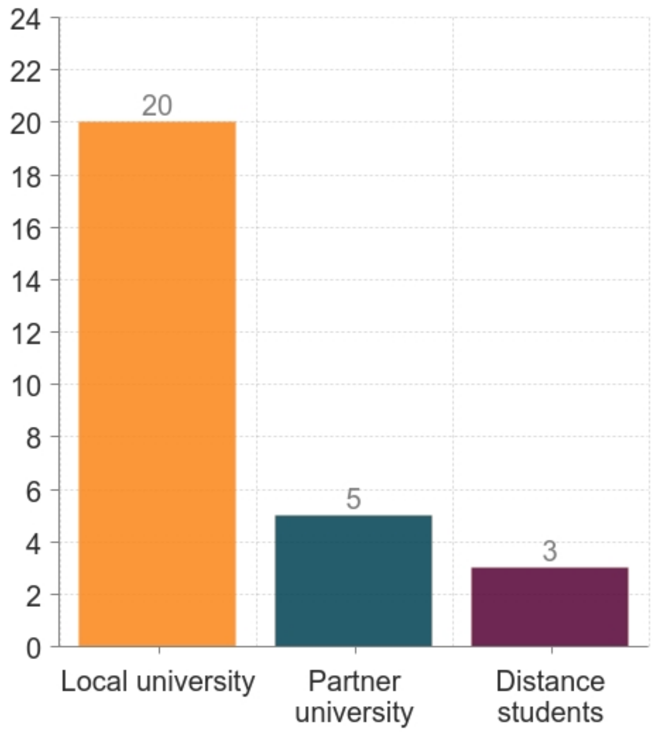

Figure A.4. School

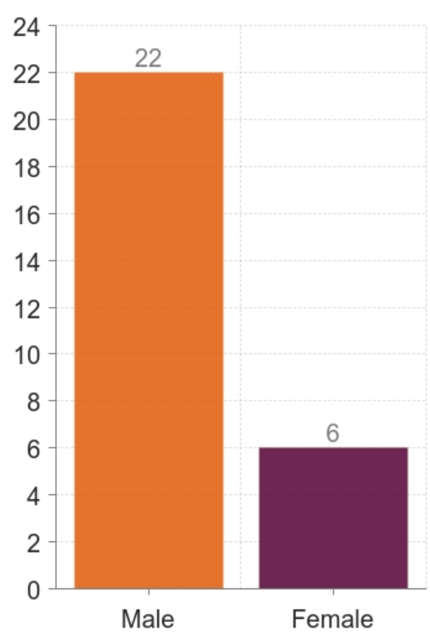

Figure A.5. Gender

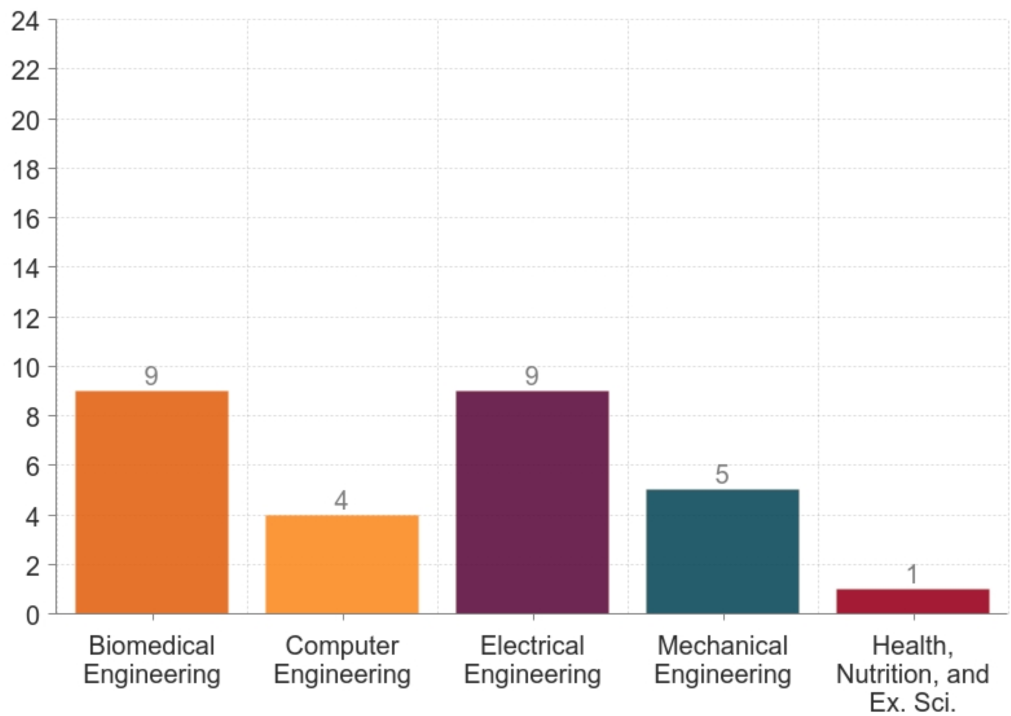

Figure A.6. Major or program