GEORGE MASON UNIVERSITY

Data Analytics Engineering

Spring 2023

Predicting CEC Graduate Course Demand

DAEN 690

Project Report

Darvik Kunal Banda
Erick A Torres
Joe B Brock
Sagar D Goswami
Telina Andrianaivo

About the Cover

Dr. Isaac Gang is an Associate Professor at the George Mason University College of Engineering and Computing, Volgenau School of Engineering, MS Data Analytics Engineering (DAEN) program.

He joined the DAEN faculty in the Fall of 2020 from Texas A&M University-Commerce (TAMUC) where he served as an Assistant Professor of Computer Science as well as the department's Outreach Coordinator. Before coming to TAMUC, Dr. Gang was an Assistant Professor of Computer Science and Engineering at the University of Mary Hardin-Baylor (UMHB) and an Adjunct Professor of Computer Science at the University of Southern Mississippi's School of Computing before joining UMHB.

Dr. Gang is a former DOE grant winner, former President and Board Member of the Association of Computer Educators in Texas (ACET), Industry Advisory Board (IAB) Coordinator, and the Director of CS For All.

His current and primary teaching responsibilities at Mason largely involves Data Analytics Engineering graduate courses along with a mix of CS and AIT graduate courses. He is an affiliate faculty member of GMU's C^4I & Cyber Center.

Dr. Gang's primary research agenda involves Big Data Analytics (emphasis on data bias and data governance), Cyber Security (ransomware, steganography, and cyberbullying), and Image/Signal Processing.

Contents

Table of Contents

This page intentionally left blank

Abstract

Abstract

INSTRUCTIONS

[NOTE: The project abstract is a separately graded assignment in the course. The final approved project abstract is to be copied word-for-word from the other assignment into this report.]

Write one paragraph of no more than 300 words that summarizes your project. Here are the typical kinds of information found in most abstracts which you should use as an outline as you develop your abstract.

1. The context or background information for your research; the general topic under study; the specific topic of your research.
2. The central questions or statement of the problem your research addresses.
3. What's already known about this question, what previous research was conducted or shown.
4. The main reason(s), the exigency, the rationale, the goals for your research — why is it important to address these questions? Are you, for example, examining a new topic? Why is that topic worth examining? Are you filling a gap in previous research? Applying new methods to take a fresh look at existing ideas or data? Resolving a dispute within the literature in your field?
5. Your research and/or analytical methods.
6. Your main findings, results, or arguments.
7. The significance or implications of your findings or arguments.

Your abstract should be intelligible on its own, without a reader's having to read your entire paper.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

This page intentionally left blank

Report

Section 1: Problem Definition

1.1 Background

George Mason University (GMU) has been experiencing significant growth, with new courses being introduced every year. Many factors drive GMU's success, which includes its acceptance rate (89.2%), high retention rate (86%), and low-cost tuition, which are appealing to those looking to apply themselves at their next job (Data USA). GMU also caters its efforts to those pursuing part-time graduate degrees with full-time jobs, but there is a problem. This large population size is excellent for GMU, but a complex problem of catering to eager students. Some percentage of the 38,541 students attending GMU may encounter issues with applying to a new class. Staffing issues, class schedules, and other factors may perpetuate these issues, but all fall under the umbrella of waitlisting.

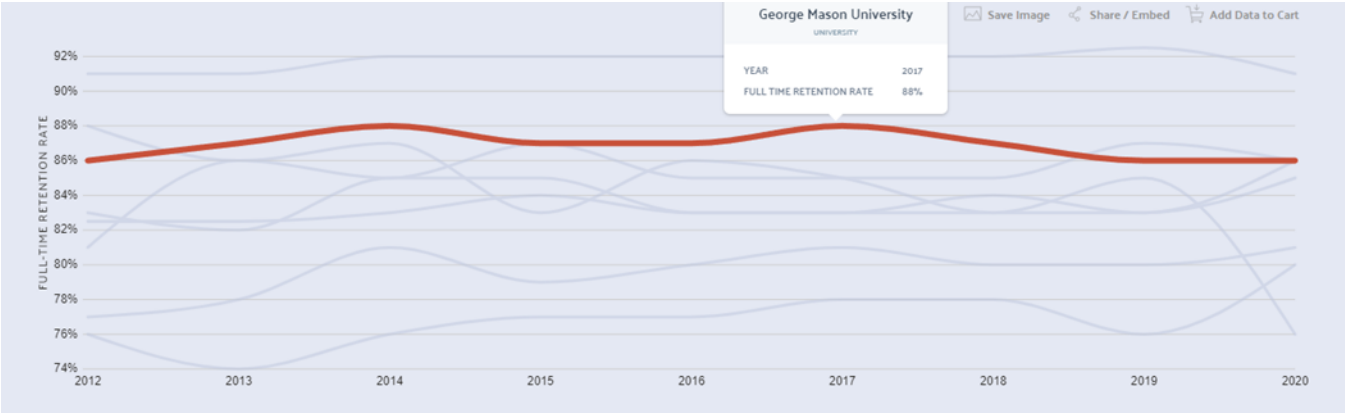


Figure 1 Retention Rate - George Mason University

Waitlisting is a concern for the Office of Provost (GMU's Academic Office) because it hinders student prospects for future employment. 32% of all Mason students are enrolled part-time, with the most common degrees encompassing much-needed fields such as Software Development, Management, and Law/Criminal Justice (Data USA). When part-time students are waitlisted, there is a possibility they will not complete their degree on time if their last classes are required. This limits their job prospects, but it is detrimental to the university. Waitlisting entails that professors may have to accommodate the class size and may be unable to devote enough time to certain students if they can't predict the class size. In addition, students that graduate late because of waitlisting will receive less favorable coverage when applying for new jobs.

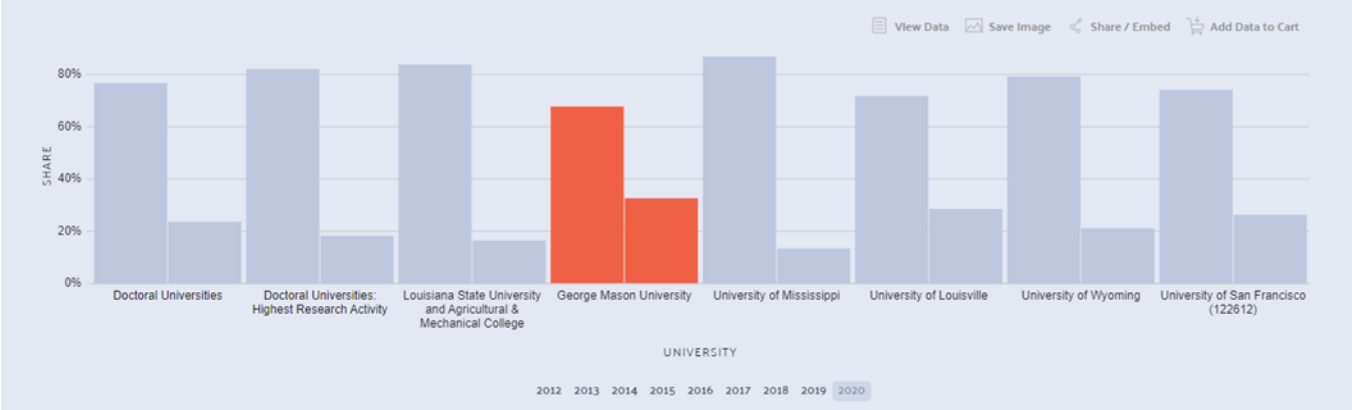


Figure 2 Full-Time vs. Part-Time Student Share

Colleges and universities in the United States prepare students for commercial jobs in various industries, including engineering, business, healthcare, and technology. Attending a university program in the United States can help you develop valuable skills that you can take advantage of anywhere in the world. In addition, it enables you to find a job in a field that offers good pay and benefits. GMU is one of the universities in Virginia that attract many international students. An international student holds an F1 visa. It is a nonimmigrant visa and is intended only for those who wish to stay temporarily in the United States to complete a professional training program.

To maintain your F1 status, you must continue to enroll full-time in your accredited professional major program. Being on a waitlist or not being able to register for classes could be a massive annoyance or even affect an international student's visa. In addition, the Office of Provost needs a concrete plan for courses for foreign students because they bring plenty of revenue to the university.

Some have pointed out online courses to alleviate issues regarding waitlisting. Online courses can fit more people in a class without having physical space, but online courses are not as accessible to some students. For instance, an international student can only take one online course per semester.

Worse, they can exasperate professor-student communications. If only 47% of students pass at what is considered "normal time," they will retain much more negative opinions about the education they received despite it being acceptable. Waitlisting demands must be better managed, as the cost threatens GMU's student satisfaction, funding, time, and reputation. To further emphasize and compare how important GMU's standing is, consider Harvard's "Time to Complete" with GMUs:

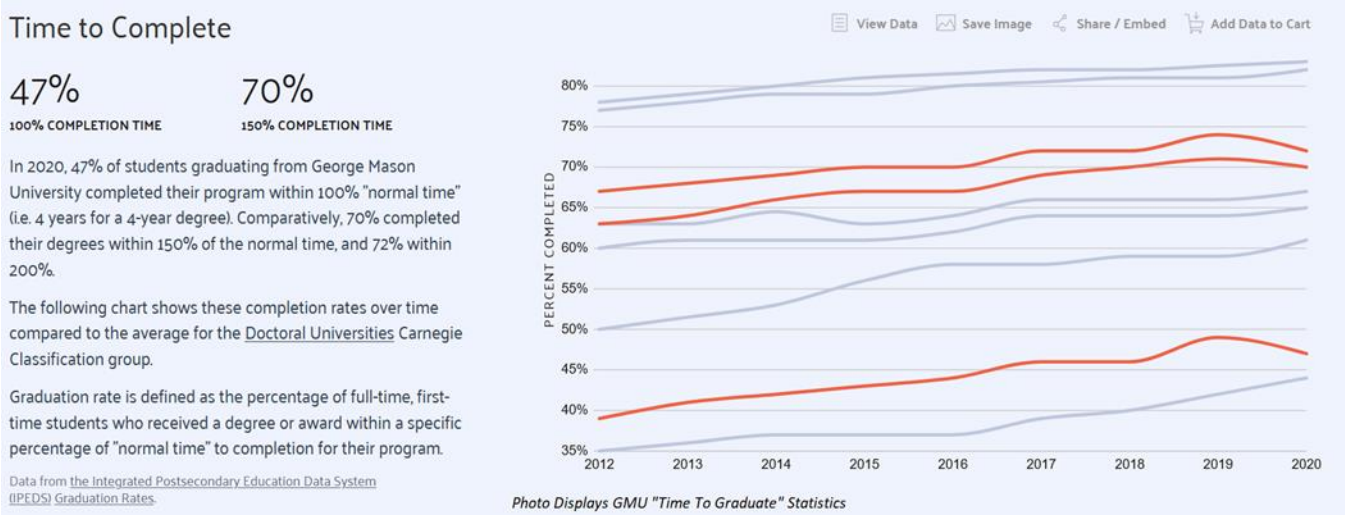


Figure 3 Time to Complete

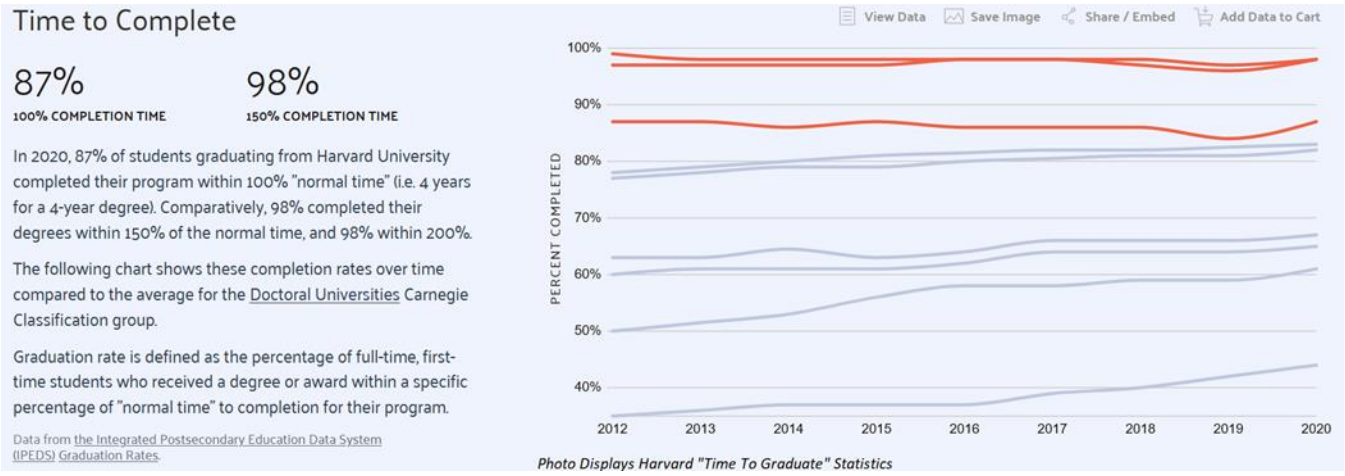


Figure 4Time to Complete: Graduation

It should be noted that Harvard is one of the top schools in the United States, but it's not the only example. James Madison University is much closer to George Mason University regarding its ranking. They have similar net prices ($18,592 JMU, $18,285 GMU), student expenses ($10,938 JMU, $12,105 GMU), and loan default rates (2.05% JMU, 2.58% GMU). The differences between the two schools can be seen in the "Time to Complete," Diversity, and the number of full-time to part-time students. Further analysis would have to confirm this, but it could be argued that JMU benefits from slightly smaller net costs and default rates because more applicants can graduate on time. Compare this to GMU's cost growth within one year (9.29%).

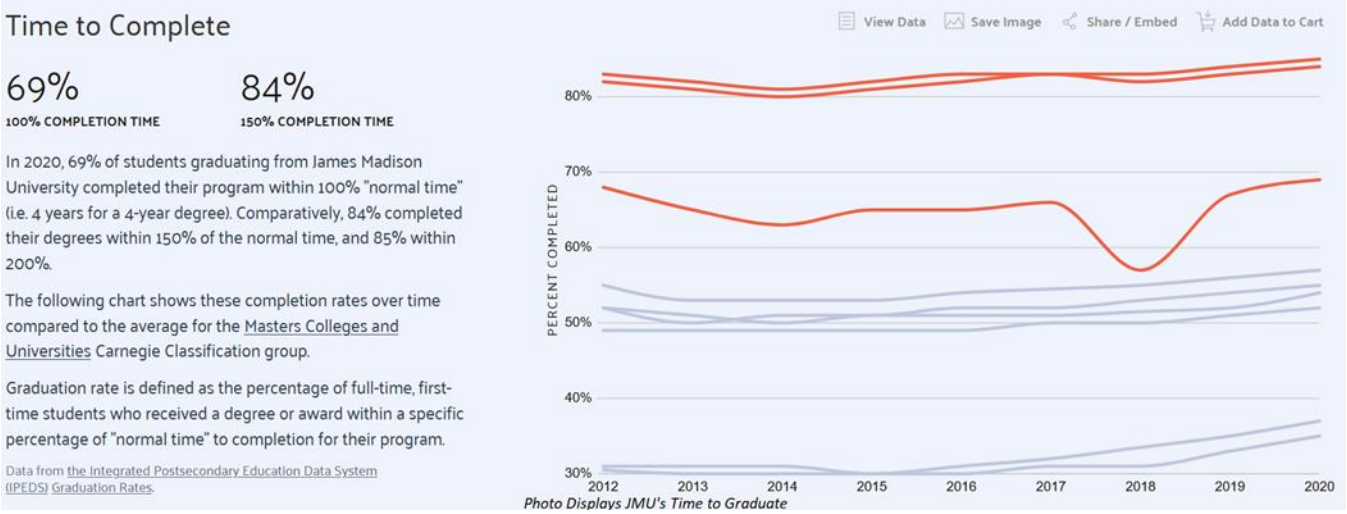


Figure 5 Time to Complete: Graduation 2

George Mason University does have many attractive features, but the total "Time to Complete" harms its overall mission and goal. These examples are used briefly to identify why George Mason's waitlisting issue could potentially affect its success, but what concrete details can be identified from the information presented? Based on conversations with the Office of Provost, it has been determined that there were five primary areas from which these issues stem: Core courses, Electives, Pre-requisites classes, and Study Interests. These areas pose key issues in predicting if a class will require waitlisting. Students have varied interests and are subject to different strategies for obtaining their desired class schedules. A common tactic desperate students will utilize includes applying to multiple waitlisted classes to receive the one they need. This may benefit the students looking to get an exact precisely course. Still, it highlights an issue that causes students to be unable to graduate on time properly. The problem can be exponentially more complex when including different grade levels (Freshman, Sophomore, Junior, and Senior), funding for a given course, the type of course, penalties, and more. The COVID pandemic has only worsened these issues, which has introduced a temporal bias into our data set.

These issues have severe economic costs to GMU as it affects the institution's ability to receive funding T-Tip Goals. If GMU cannot fill Computer Science or STEM-related majors, it puts GMU at a competitive disadvantage when expanding and growing. Waitlisting is a severe problem that must be addressed. This paper will explain in detail the solutions used to resolve waitlisting issues. Each section presents vital facets of the ideation, implementation, and resolution of the problems caused by waitlisting. It must be stated that this is not a completed list. More solutions can be implemented to assist in the prediction and minimization of waitlisted students. This paper's purpose is to provide a solution to tan answering the total number of waitlisted students.

1.2 Problem Space

This paper's fundamental goal is to minimize the number of waitlisted seats available and predict the number of seats for a given class. This project deals with different working parts that are highly variable and subject to change. Anticipating those results may be difficult. Biased information is constantly present, including but not limited to temporal selection, confirmation, and survivorship biases. To address the five areas effectively, the following constraints have been included. First, the Office of Provost has limited the total scope of this project to only focus on CEC graduates. This limits the pool significantly, as there are only so many graduate students.

Outside of the Office of Provost, other factors must be considered. Core Courses, Electives, and Pre-Requisite classes are likely the most predictive factors to whether a student can graduate on time. While the main goal is

to minimize waitlisting applications, the ability of students to graduate on time is a significant factor associated with this. Questions that must be answered prior to the conclusion of this question revolve around variable independence and limitations. If there is no discernable difference between these three types of classes for students, then all limitations listed are affected. Core Courses and Pre-Requisite classes are likely the most important for defining constraints. Electives are widely available, as opposed to core or pre-requisite classes.

The problem statement will also require measuring the depth of any given Pre-Requisite classes. If a given Pre-Requisite class has a certain number of past references to other types, the percentage chance of that class's waitlist growing over time could be possible. It is necessary to confirm this assumption because it will help determine if courses with more prerequisites correlate with waitlisted classes. Between Core Courses and Pre-Requisite classes, Pre-Requisite classes are more important. Core Classes can have pre-requisite attached to them, but Pre-Requisite classes require prior courses to be available to be taken. This causes scheduling problems, in which the solution is to take the Core Class needed, but the course itself might be available.

Studying Student Interests is essential, but it can only be predicted in a few ways. All core areas mentioned previously contain some variability, but student interests are the most prone to change. Models that could measure student satisfaction, degree change rate, and drop rate to predict Student Interest. However, these qualifiers are not something that can truly be measured. Students may make these changes for any reason, but these qualifiers make for an excellent reasonable control against waitlisting. If a student makes a change to a last-minute change to drop a class, it should be an indication that the student could not pursue their interest course class.

Non-waitlisted courses, Bridge Courses, Labs, and Scheduling Conflicts are specific examples of how Student Interest affects waitlisting. Non-waitlisted courses include complex requirements that waitlists are not allowed. This could be because there needs to be more resources for that class, or the class size is determined by the amount of professors willing to teach it. Bridge Courses are required courses that are needed for potential graduate students to be enrolled in their chosen master's program. This could be used as a constraint to filter students based on their academic eligibility within datasets to come, but in conjunction with their specified degree it could predict their performance. For example, Engineering students may have the same classes as Computer Scientists, but may fail more classes on average because they don't have the same professional knowledge. Labs are a problem constraint because they are required to operate in conjunction with a given class. If a class is available, but the lab is not, the student might not be able to complete a necessary portion of their degree for that reason.

Classes that are allowed to be waitlisted have a maximum capacity of 99 by default. This can cause many problems regarding expectations for the total class size, since waitlisting operates on a first come/first serve basis. Specializations can exasperate this problem, as more competitive fields are less likely to have open spots available. Whether the class is online also affects the ability for a student to graduate on time, as some students do not have the ability to apply for online. International Students are bounded by their credit status, and if the classes they are required to take are only online it may cause issues in the future.

Student's scheduling conflicts can inflame waitlisting problems. Classes that are assigned for only certain parts of the day will dissuade certain majors from applying to them. Student behavior cannot be controlled, and it may be difficult for professors to coordinate others on which time slots are available. More timeslots would be ideal, but there is a limited number of professors to assign those timeslots.

Working around these issues is an optimization problem which requires time and patience. All problems listed are constraints that must be accurately defined and documented to prevent biases. The primary fear for these problems resolution is not to find a resolution, but finding a resolution that is ill-fated for minimizing waitlisting positions. Most post-COVID data will likely have to be scrapped or thoroughly examined. This includes the initial

starting months of the pandemic and the months thereafter. Student sentiment and purchasing decisions during these months will have to be heavily monitored, as they indicate what changed throughout this time period if the COVID-era is going to be used.

Pre-Requisite course selection is the most important field, and aggregation functions are needed to count how many pre-requisites are required for each class. Labs must be fitted with any pre-requisite and core classes that are found. Core Course's size must be identified and compared against Pre-Requisite courses. The problems presented by these solutions are varied. It might not be enough to reduce the bias, and inversely it could remove too much data. The total graduate student population is expected to be low, but focusing on Pre-Requisite courses is the best way to handle the variability.

Students will have to be split into different groups based on the faculty that they have available, types of courses that they have, and their academic standing. Filtering by online and in-person classes would help create clearer data, but it is dependent on ease-of-use. T-Tip Funding Goals will be a massive constraint for this project. It will prioritize which waitlisted classes that provide the largest bankroll for the university. This could cause a serious collection bias, so it must be monitored.

These provided constraints help guide the project to a unified solution, but other smaller issues are present based on the technology. AWS provides a technical framework for how our solutions will come together within the GMU region. The shortcomings of using AWS will regard time, money, and space, provided that GMU decides to expand this service to other AWS regions. The licensing for LP solvers could be harm GMU if the technology does not turn out to be as useful. There are many technical failings that could occur, from a lack of technical staff managing the cloud environment to poor configuration management.

Disk Size and CPU usage is likely not going to be an issue for any data scientists implementing the system described by this paper following it's completion. What will be an issue is a lack of communication between the computer science side of this equation and the data science/analytics side. The technical knowledge listed throughout this paper must be used to understand and resolve the issues provided. The research provided presents examples of how this was done in prior projects, and is used as an inspiration for this project.

1.3 Research

Upon searching for previous work on this topic, and evaluating applicability, merits, and demerits of each techniques, we have shortlisted following literature that can directly/indirectly provide a better starting point for our project.

1.3.1 Music Recommendation System

One of the research projects we found was on A Music Recommendation System with Mathematical Optimization (Gurobi Optimization, 2022). Businesses need music recommender systems to choose songs that are likely to be enjoyed from the huge online or personal databases. This research created a recommendation system using a mixture of predictive and prescriptive analytics. According to this research," the predictive component foresees what users might be into based on their past music preferences, while the prescriptive component uses these predictions to create an optimally diverse recommendation list." (Gurobi Optimization, 2022)

We can use a similar style of predictive analytics and focus on classes students will most likely enroll in at GMU. If our data focuses on the available classes for each student but also the student's preference over those available classes. Using a prediction model to learn the students' preferences of classes using collaborative filtering, we can list the classes that the Provost should offer during the semester by picking the list from the top preferred classes.

1.3.2 Enrollment Predictions with Machine Learning

Researchers at California State University: Channel Islands (CSU-CI) conducted a research study where they demonstrated a proof of concept of using Data Analytics techniques in AWS SageMaker to predict Student Enrollments at the California State University (Soltys, Dang, Reilly, & Soltys, 2021). They targeted this research towards SEM Practitioners and Data Analytics Technical Practitioners, who are specifically making Enrollment Management decisions.

They used a combination of various ML Algorithms like AWS SageMaker XGBoost (Exteme Gradient Boosting), and Time-Series Analysis to predict the probability of a student enrolling. The data was distributed in 80% Training Set and 20% Test Set. The frequency distribution of Student Enrollment Probabilities was then filtered by a set threshold value of 0.09 (9%) and summed to obtain the expected student enrollment for that term.

They also remarked, that while the predictions are made at the student level, this technique does not provide a definitive answer on whether a student will enroll or not. However, it performs exceptionally well when aggregating the results to make predictions for entire batch of students for that term. Even with limited data points, they were able to predict True-Negatives at 75% accuracy, and True-Positives at 61%. By considering more variables, setting weights to penalize False-Negatives, and setting some tolerance value, the University can make better estimates to ensure enough resource allocation for student enrollments, well in advance.

1.3.3 Machine Learning Methods for Course Enrollment Prediction

The researchers at San Diego State University (SDSU) conducted a research on predicting course enrollment based on various metrics pertaining to University, Demographics, Admissions, etc. They used previous research on Conditional Probability Analysis as a base model, and improved on it using Tree Models like Classification and Regression Trees (CART) and Random Forest (Shao, Ieong, Levine, Stronach, & Fan, 2022).

In the Conditional Probability Analysis technique, the students are divided into various categories like First-Time/Current/Transfer Students, Pre-requisites met/unmet, First Attempt/Multiple attempts, etc. Predictions for the probability of students registering for this course was computed for each of those categories, and summed to get the final student enrollment prediction for a given course.

Additionally, they also worked on CART and Random Forest Models, building on the Conditional Probability. From the results, it was clear that CART performed better than Conditional Probability. However, it was observed that the error for both the models exploded, when predicting 2 years in future. Random Forest on the other hand, gave consistent results with a very low error rate for both the years. The error rates for all these models are as follow:

Term	Actual Enrollment	Predicted Enrollment, by Method					
		Conditional Probability Analysis		Classification and Regression Tree		Random Forest	
		n	Error (%)	n	Error (%)	n	Error (%)
Fall 2018	650	627	3.5	655	0.8	658	1.3
Fall 2019	396	312	21.2	335	15.5	393	0.8

Figure 6 Comparing ML Methods for Course Enrollment Prediction

1.4 Solution Space

This paper's solution to the problem statements listed prior considers the technical limitations first. Our system's primary goal is to minimize and predict waitlisting times, so the following tools will be used to accomplish this task: AWS, Gurobi, and Python. The target size for the input is likely going to be under 1GB of information, but it could be larger dependent on the data scraped. Security is an issue for this project, as private student information under the input folder. Those without proper access should not be allowed in, and this should be a programmable feature for those who are developers looking to improve upon this iteration. A scheduler should also be provided to our system that triggers whenever a new input file is introduced. AWS provides a resolution to both issues through its services. AWS's Internet Gateway would allow for public subnets to access the internet and manage the total load for the site. Public subnets here are defined as information that the average user has access to, whereas private subnets focus on information that the system itself has access to.

The public subnet's job here is to primarily check whether S3 bucket information has been updated by accessing it through it's role permissions. This is an automatic process, so AWS Lambda will be called to start and call an EC2 instance in the public subnet. The public EC2 instance will use this as a time to scrape any data from the CEC website. That data is routed to a private PostgreSQL database within our database. Following the insertion of the scraped data, the EC2 subnet instance analyzes the data. It filters out information derived from both the scraped data and the inputted file, creating value constraints. These constraints will limit special cases and the track total number of pre-requisites. An evaluation will be processed through Gurobi for the expressed purpose of minimizing the waitlist. Gurobi is used in the public subnet as opposed to the public subnet because an ENI is required for EC2 instance to function properly. Once the data is collected, data from the prior dataset will be fed through an ARIMA and Linear Regression model to determine time series data and variable importance. The information is then fed back into the S3 storage unit, with all resultant analysis for that time period listed.

The diagram listed below provides a simplified list of the architecture this paper is hoping to develop:

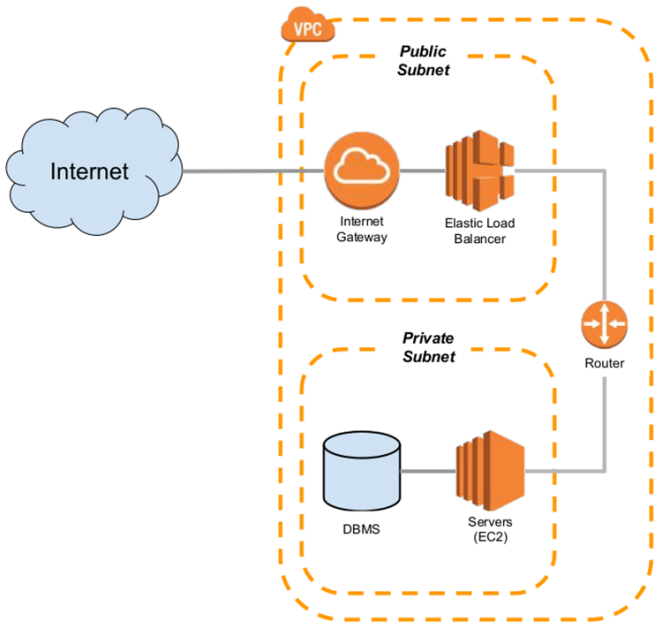


Figure 7 Simplified Cloud Architecture for the project.

1.5 Project Objectives

Project Objectives are imperative, as the team will invest in tools such as to AWS, Gurobi, and basic automation scripting (Python/Bash) to complete the task. AWS has many useful data science and analytics tools that must be incorporated into the EC2 instances presented through Boto3, AWS's API. If the team is able to master these tools, they will have gained skills that will help them in industry Data Science jobs and greatly assist the Office of Provost. Completion of the project requires that the system successfully predicts waitlisted classes, identify key predictors for waitlisted classes, and minimizes the overall waitlist. This solution can then be replicated by other teams to apply to undergraduate classes, saving the university time and money.

1.6 Primary User Stories

The following user stories is capture the core of our requirements gather process based on our primary objectives:

- "As a User, I would like to have access to all the required tools and software so that way I can begin the project."
- "As a Scrum Master, I would like to manage all the files related to development in a secured private Version Management Storage so that the team can collaborate, and files are protected against issues relating to versioning/updates"
- "As a Product Owner, I want to understand the Data Acquisition process fully so that the team is ready with appropriate steps and tools to work on the data acquisition, transfer, and storage."

The following user stories regard our Research Requirements:

- "As a Developer/Product Owner/Scrum Master, I need to learn and understand Enrollment Management in order to better identify the needs and requirements of this project."

1.7 Product Vision

1.7.1 Scenario #1

The Office of Provost would benefit from minimized waitlists. Less courses available mean less students being able to take a class. Frustrated students create a more negative class experience, biasing students against the class. By reducing the waitlist, more students can participate in the classes thus generating more money to the university.

1.7.2 Scenario #2

Waitlists can deter academic progress for professors, since adding students means allocating more resources. If individual students have issues with a class, their specific issues may take time to properly resolve. Professors who are unable to handle extra students or students that are likely to drop will have a harder time giving students the best quality education. Our tool will not fix the organization issues that professors may have, but it will provide professors better predict volatile waitlisting rates for a class.

1.7.3 Scenario #3

Parents benefit knowing that their students will not be forced to apply for another class. Going to college is an expensive endeavor for schools who are retaining students and parents who are paying for the classes. Students who are waitlisted for classes will be in a worse position to graduate, and late graduations negatively affect job outlooks. Parents should be comfortable knowing that their adult children won't pass their graduation date because their classes weren't available when they needed them.

Section 2: Datasets

2.1 Overview

INSTRUCTIONS

Provide a descriptive overview of your datasets.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.2 Field Descriptions

INSTRUCTIONS

Described your dataset field. Make sure you study the example below and you will more than likely expand these fields:

1. URL (Type: string) – The web address or Universal Resource Locator for the webpage that contained the news article. This includes the protocol (http or https), host name, and subdomain. Some URLs also include parameters (text following '?') or named anchors (text following a '#'). Each URL can only be present once in the database, even if the webpage is not static over time.
2. Title (Type: string) – The title of the news article as parsed by the Newspaper 3K module. This field may be null (~150 articles in our dataset do not have titles).
3. Authors (Type: string) –The authors of the news article as parsed by the Newspaper 3K module. This field may be null (~23,000 articles do not have authors) and articles with multiple authors have their names joined with a comma into a single string. This field may also pick up descriptions of the author, including their titles and background.
4. Publication Date (Type: datetime) – The article publication date and time as parsed by the Newspaper 3K module. The datetime is displayed in ISO 8601 format (YYYY-MM-DD Thh:mm:ss+offset). Publish dates without specified times are assumed to be published at midnight. Publication dates with time information, but without a time zone listing, are assumed to be in Eastern Standard Time. This field is not allowed to be null.
5. Text (Type: string) – The text of an article as parsed by Newspaper 3K. This field may be null (~8,000 articles do not have text) as some news stories are delivered as only video, audio, or a picture. The mean word count for text is 538.9 across all news sources.
6. Tags (Type: string) – Article tags as determined by Newspaper 3K. These appear to be important (rare or "topicy") words taken from the article text, not meta tags contained in the article's HTML. Multiple tags are concatenated with a comma into a single string.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.3 Data Context

INSTRUCTIONS

Provide a description of the data context.

Data context is the set of circumstances that surround a collection of data. Capturing and interpreting context is a basic step in data analysis. Use of out-of-context data is a common source of errors in scientific research, business decisions, and professional advice.

In business analytics (BA), gathering context from external sources can provide useful information about events that have significance for the organization. Context for an unexplained surge in sales, for example, could be provided by pulling in data from news and social media as well as less obvious sources, such as weather over that period. Explored in context, it may be able to identify external causes for the increase, and that information might be used to guide future business decisions.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.4 Data Conditioning

INSTRUCTIONS

Describe the data conditioning required for each data set.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.5 Data Quality Assessment

INSTRUCTIONS

At a minimum you must assess your data sets with the following attributes:

- Completeness
- Uniqueness
- Accuracy
- Atomicity
- Conformity
- Overall Quality

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.6 Other Data Sources

INSTRUCTIONS

If you are considered other data sources, however, you decided not to use these sources provide some reason why they were not utilized.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.7 Storage Medium

INSTRUCTIONS

Discuss the storage medium selected for the project data set storage.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.8 Storage Security

INSTRUCTIONS

Discuss the storage security required for the project data set storage.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

2.9 Storage Costs

INSTRUCTIONS

Discuss storage costs associated with the storage medium used for the project data set storeage,

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Section 3: Algorithms & Analysis / ML Model Exploration & Selection

3.1 Solution Approach

INSTRUCTIONS

Provide a detailed discussion of the solution approach. Include discussions on any of the following:

1. Systems Architecture
2. Systems Security
3. Systems Data Flows
4. Algorithms & Analysis
5. Machine Learning (delete this subsection for non-machine learning projects.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

3.1.1 Systems Architecture

3.1.2 Systems Security

3.1.3 Systems Data Flows

3.1.4 Algorithms & Analysis

3.2 Machine Learning

INSTRUCTIONS

For Machine Learning projects discuss the model exploration and selection process. Delete this report subsection for non-machine learning projects.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

3.2.1 Model Exploration

3.2.2 Model Selection

Section 4: Visualizations / ML Model Training, Evaluation, & Validation

4.1 Overview

INSTRUCTIONS

Provide an overview of what was accomplished during Sprint 4. Focus visualizations for non-machine learning projects.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

4.2 Visualizations

4.3 Machine Learning

INSTRUCTIONS

For Machine Learning projects, discuss your approach to the following with respect to the ML Model:

1. Training,
2. Evaluation, and
3. Validation of the ML Model.

Delete this report subsection for non-machine learning projects.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

4.3.1 Model Training

4.3.2 Model Evaluation

4.3.3 Model Validation

Section 5: Findings

INSTRUCTIONS

Discuss the major findings of the project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Section 6: Summary

INSTRUCTIONS

Summarize the overall project and results for the reader. What did you discover, prove, disprove, etc.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Section 7: Future Work

INSTRUCTIONS

This is critical section of the report. Propose future follow-on work or next step(s) for the project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Appendix

Appendix A: Glossary

Term	Definition

INSTRUCTIONS

Place all terms which require definitions in the Appendix A: Glossary.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Appendix B: GitHub Repository

Overview

INSTRUCTIONS

Provide a GitHub Link and the README.MD content. Do not just provide a link to the GitHub repository but provide a narrative paragraph which introduces the project. This section should mirror the look and feel of a well-documented professional GitHub site.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

GitHub Repository Link

GitHub Repository Contents

Appendix C: Risks

Sprint 1 Risks

Table 1 Sprint 1: Risks

Risk Name	*Description*	*Probability*	*Impact*	*Mitigation*
Miscommunication	*The team gets confused as to what is expected.*	*Medium*	*High*	*Encourage YouTrack to be checked and updated daily.*
Task Management	*The large amount of tasks can be overwhelming.*	*Medium*	*Medium*	*Organize YouTrack to provide more guidance on tasks to be completed.*
Lack of Research	*Hard to find research related to our topic.*	*Medium*	*Medium*	*Find other ways of completing our research of prior studies.*
Scheduling Issues	*It is a challenge to get everyone together at one time.*	*High*	*Medium*	*Encourage independent work and communicate work being completed*

Sprint 2 Risks

INSTRUCTIONS

Include the risk table associated with the Sprint. Below the risk table provide a narrative description of how the risks and mitigation plans were identified, what the team got correct, what the team could have done differently, how accurate was the team in identifying the risks, did the team encounter any unanticipated risks, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Sprint 3 Risks

INSTRUCTIONS

Include the risk table associated with the Sprint. Below the risk table provide a narrative description of how the risks and mitigation plans were identified, what the team got correct, what the team could have done differently, how accurate was the team in identifying the risks, did the team encounter any unanticipated risks, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Sprint 4 Risks

INSTRUCTIONS

Include the risk table associated with the Sprint. Below the risk table provide a narrative description of how the risks and mitigation plans were identified, what the team got correct, what the team could have done differently, how accurate was the team in identifying the risks, did the team encounter any unanticipated risks, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

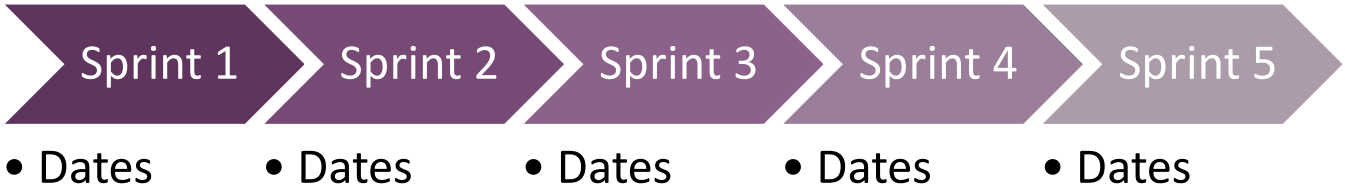
Sprint 5 Risks

INSTRUCTIONS

Include the risk table associated with the Sprint. Below the risk table provide a narrative description of how the risks and mitigation plans were identified, what the team got correct, what the team could have done differently, how accurate was the team in identifying the risks, did the team encounter any unanticipated risks, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Appendix D: Agile Development



Scrum Methodology

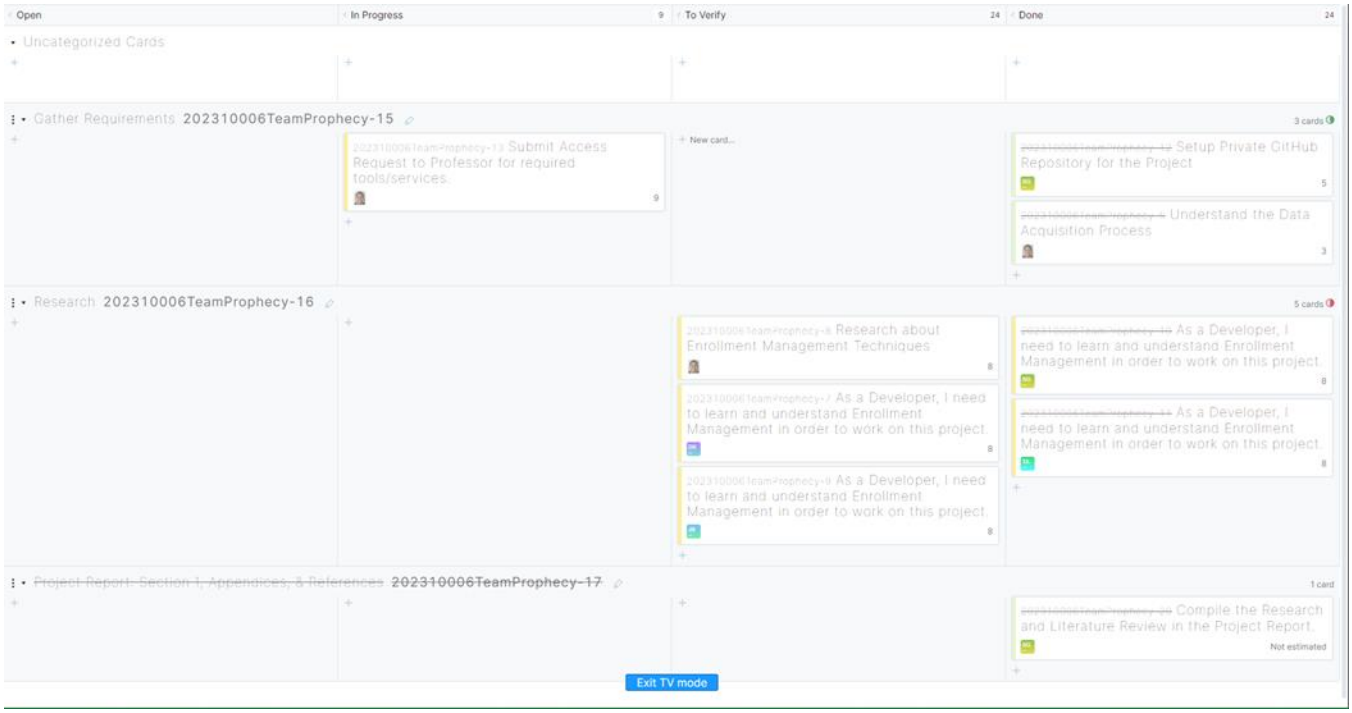


Figure 8 Sprint 1: Agile Board

Sprint 1 Analysis

INSTRUCTIONS

Provide a narrative of the team's efforts during this Sprint. Be sure to include – but not be limited to – how the team identified the User Stories, how well the team performed with the various tasks, how easy/difficult it was for the team to manage their activities during the Sprint, what did the team do correct, what could/should the team have done differently, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Sprint 2 Analyis

INSTRUCTIONS

Provide a narrative of the team's efforts during this Sprint. Be sure to include – but not be limited to – how the team identified the User Stories, how well the team performed with the various tasks, how easy/difficult it was for the team to manage their activities during the Sprint, what did the team do correct, what could/should the team have done differently, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Sprint 3 Analysis

INSTRUCTIONS

Provide a narrative of the team's efforts during this Sprint. Be sure to include – but not be limited to – how the team identified the User Stories, how well the team performed with the various tasks, how easy/difficult it was for the team to manage their activities during the Sprint, what did the team do correct, what could/should the team have done differently, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Sprint 4 Analysis

INSTRUCTIONS

Provide a narrative of the team's efforts during this Sprint. Be sure to include – but not be limited to – how the team identified the User Stories, how well the team performed with the various tasks, how easy/difficult it was for the team to manage their activities during the Sprint, what did the team do correct, what could/should the team have done differently, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Sprint 5 Analysis

INSTRUCTIONS

Provide a narrative of the team's efforts during this Sprint. Be sure to include – but not be limited to – how the team identified the User Stories, how well the team performed with the various tasks, how easy/difficult it was for the team to manage their activities during the Sprint, what did the team do correct, what could/should the team have done differently, etc. Think of this writeup as a "lessons learned" that you would like to pass along to any project team thinking of doing a similar project.

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

Reference

Works Cited

Data USA. (n.d.). *George Mason University.* Data USA. Retrieved from https://datausa.io/profile/university/george-mason-university/#:~:text=George%20Mason%20University%20received%2021%2C198%20undergraduate%20applications%20in,accepted%20for%20enrollment%2C%20representing%20a%2089.2%25%20acceptance%20rate.

Gurobi Optimization. (2022). A Music Recommendation System with Mathematical Optimization. Retrieved from https://colab.research.google.com/notebooks/intro.ipynb#scrollTo=GJBs_flRovLc

Gurobi Optimization. (2022). Music Recommendation System: Jupyter Notebook Model for a Music Recommendation System. Gurobi Optimization. Retrieved from https://www.gurobi.com/jupyter_models/diverse-music-recommendation-system/

Shao, L., Ieong, M., Levine, R. A., Stronach, J., & Fan, J. (2022). Machine Learning Methods for Course Enrollment Prediction. *Strategic Enrollment Management Quarterly, 10*(2), 11-29. Retrieved from https://www.proquest.com/docview/2697182295/fulltextPDF/A3E7B94D67954EA8PQ/1?accountid=14541

Soltys, M., Dang, H., Reilly, G. R., & Soltys, K. (2021). Enrollment Predictions with Machine Learning. *Strategic Enrollment Management Quarterly, 9*(2), 11-18. Retrieved from https://www.proquest.com/docview/2606939441/fulltextPDF/6C409A77790B4A22PQ/4?accountid=14541

INSTRUCTIONS

The References section of this document makes use of the Microsoft Word References feature to insert research citations by recording them directly into the document. All citations are to follow the IEEE citation format. Use the Bibliography drop down to have Microsoft Word dynamically create your Works Cited section

here in IEEE citation format.

To learn more about the IEEE Citation guidelines click on the document links below.

1. IEEE-Reference-Guide.pdf
2. IEEE Citation Guidelines2.doc (ieee-dataport.org)

DELETE THIS TEXT BOX AFTER YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS.

This page intentionally left blank