2022

Development of a Multivariate Poisson Hidden Markov Model for Application in Educational Data Mining

Shahab Boumi
University of Central Florida

DEVELOPMENT OF A MULTIVARIATE POISSON HIDDEN MARKOV MODEL FOR APPLICATION IN EDUCATIONAL DATA MINING

By

SHAHAB BOUMI
B.S. University of Kurdistan, 2013
M.Sc. Sharif University of Technology, 2015

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Industrial Engineering and Management Systems
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2022

Major Professor: Adan Vela

ABSTRACT

Managers and policymakers in higher education institutions try to improve graduation rates and decrease halt rates. To achieve this goal, it is important to understand academic and demographic factors that correlate with academic performance. Many studies in the field of education analytics have identified student grade point averages (GPA) as an important indicator and predictor of final academic outcomes (graduating or halting their studies). While semester-to-semester fluctuations in GPA are considered normal, significant changes in academic performance may warrant more thorough investigation and consideration, particularly with regard to final academic outcomes. However, it is challenging to represent complex academic trajectories over an academic career. To overcome this challenge, in this dissertation two different Hidden Markov Models (HMMs) are developed to provide a standard and intuitive classification over students' academic-performance levels. This leads to a compact representation of academic-performance trajectories. Next, the relationship between different academic-performance trajectories and their correspondence to final academic success are explored. Based on student transcript data from the University of Central Florida, the proposed HMMs are trained using sequences of students' course grades for each semester to estimate the students' academic-performance levels. Through the HMMs, the analysis follows the expected finding that higher academic performance levels correlate with lower halt rates. However, in this dissertation, we identify many scenarios in which both improving or worsening academic-performance trajectories actually correlate to higher graduation rates. This counter-intuitive finding is made possible through the two proposed HMMs.

To My Family

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION AND BACKGROUND

Despite the importance of higher education, in the United States it is noted to have numerous inefficiencies and disparities in a variety of contexts. Based on the National Student Clearinghouse Research Center, on average, only 58% of students who began college in Fall 2012 have earned their degrees within six years, with the remaining graduating in more than six years or leaving college without a degree (referred to as halting) [1]. The effect of inefficiencies in higher education is even noted after graduation, as many employers believe college graduates lack the necessary qualifications and skills to succeed in the workplace [2]. They believe that many graduates are not properly prepared for their jobs and lack the basic skill sets, including critical thinking and problem-solving. B. Derek argues that due to the weakness of course design in the education system, college students today spend much less time on their course work when compared to their predecessors 50 years ago [2].

In an effort to minimize students' halt rate, universities need to proactively identify risk factors impacting academic performance. Statistical tools should then be used to track these factors, facilitating interventions to provide support. Earlier studies suggest that one of the most important risk factors corresponds to student demographics. For example, students coming from families with lower incomes are more likely to halt their education when compared to students coming from families with higher incomes. Similar disparities are noted across ethnicity, race, gender, and other demographic features. One study conducted by the Pew Research Center [3] suggests a performance gap between male and female students; female students have a higher graduation rate when compared to males. This difference is more significant for students aged between 25 and 34 rather than students aged 25 and younger, showing that gender and age are two tied demographic features that are correlated with academic-performance. Furthermore, based on the

National Clearinghouse Research Center [4], students with a white race/ethnicity have a higher six-year graduation rate (62.0%) when compared to Hispanic (45.8%) or black (38.0%) students.

Several other features are studied by Marbouti et al. [5]. For example, they showed that Hispanic, first-generation, low-income students have a lower GPA and take fewer classes per semester when compared to other students, and therefore, take more time to graduate. Another class of studies has introduced enrollment types, including full-time and part-time, as a predictor for students' halt/graduate status. Feldman et al. [6] showed that on average, students with full-time enrollment have higher retention rates when compared to part-time students. Another study conducted by Darolia [7] illustrates that students who take more credits in their first semester are more likely to complete their program and graduate. The impact of admission type, including first-time-in-college (FTIC) or transfer admission on students' GPA is investigated in a study by L. Smith et al. [8]. Their results suggest that transfer students, in general, are expected to have a lower relative GPA when compared to their FTIC counterparts as a result of so called *transfer shock*.

GPA has received significant attention among researchers as a critical identifier to predict students' graduation rates. For example, Plagge et al. [9] show that among several important demographic, academic, and financial features, including gender, race/ethnicity, and estimated family contribution, the first Fall semester GPA is the most significant identifier for predicting first-to-second-year retention rate. In another study, Ojha et al. [10] showed that semester GPA and cumulative credit hours are the most powerful predictors for graduation delay. Despite the importance of GPA and course grades, it is crucial to acknowledge that those are not the *cause* for the students' halt or graduation, but rather merely a proximate measure or approximate outcome[1]. The reasons that students leave college can include doubts about their success, their lack of enjoyment, a lack of proper support, and so on. In fact, these non-observable reasons will result in the observable outcomes of lower course grades and GPAs.

While other factors as mentioned earlier (e.g., demographic features) could also impact stu-

[1] There is one exception for that, as a student with a GPA less than 2 is on academic probation and should leave college.

dents' performance, in this dissertation, the main focus is on students' course grades and their correlation with their halt/graduation status. This manuscript provides a comprehensive study assessing the shortfalls of the current machine learning studies that take GPA as input to predict students' outcomes; building on this, an alternative representation of GPA and course grade are proposed to guarantee better prediction accuracy.

The importance of identifying factors that impact students' halt/graduate status, along with the critical shortcomings of these methods, has stimulated a large body of scientific research to identify curricula, teaching modalities and techniques, policies, and other solutions to improve graduation rates in higher education. The wide range of solutions draws from both qualitative and quantitative techniques. Of particular interest to this dissertation are technical approaches that make use of educational data sets to evaluate, analyze, and improve the performance of students and higher education systems [11, 12]. Educational Data Mining (EDM) is one of the recent but most popular and effective methods within that pool. EDM is defined as ”an emerging discipline concerned with developing methods for exploring the unique types of data that come from educational settings and using those methods to better understand students, and the settings which they learn in” [13]. Romero et al. have defined four main stakeholders and users engaged with EDM, including Learners, Educators, Researchers, and Administrators.

1.1 Related works

In this section, a systematic literature review is conducted to identify the most popular areas that EDM focuses on and gaps in the literature. Hence, a comprehensive search is performed to find previous works in this area. The applied search terms in this systematic literature review are developed based on the steps of PRISMA [14]. The databases which are used in this systematic literature review are Springer Link, IEEE Xplore, ACM Digital Library, ProQuest Computer Science Collection, Science Direct, Engineering Village, and EBSCOhost. Keywords used in the search area are as follows: (educational data mining OR educational machine learning) AND (algorithms OR

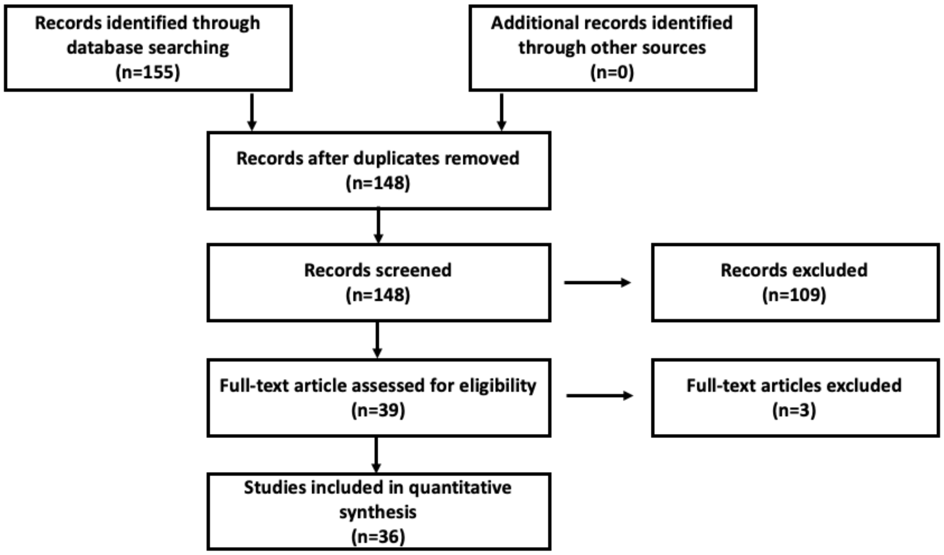

Records identified through database searching (n=155)
Additional records identified through other sources (n=0)
Records after duplicates removed (n=148)
Records screened (n=148)
Records excluded (n=109)
Full-text article assessed for eligibility (n=39)
Full-text articles excluded (n=3)
Studies included in quantitative synthesis (n=36)

Figure 1.1: Chart of the selection strategy

Table 1.1: Classification of the papers included in the review

Category	number of papers
Predicting students' academic performance	8
Understating factors that impact students' academic performance	20
Designing recommendation systems	8

technique OR method OR application OR system) AND (students' GPA OR students' academic performance) AND (estimation OR prediction) AND (students' academic behavior).

Figure 1.1 displays the selection strategy chart according to PRISMA guidelines. The conducted review contains a total of 36 papers written by 84 authors. Of the 36 papers, 26 were published in journal papers, and 10 were published in conferences proceedings. The existing body of papers in this systematic literature review that focus on the application of data mining in the field of education can be grouped into three main categories: (1) predicting students' academic performance, (2) understating factors that impact students' academic performance, and (3) designing recommendation systems. The number of papers presented in this literature review for each category is summarized in Table 1.1. Each of these domains is explained in more detail below:

1.1.1 Predicting students' academic performance

The main goal of this application area is to predict students' academic performance, which could be beneficial for learners, educators, and university policy-makers in better understanding and improving performances. Some research in this area focuses on assessing different machine learning algorithms (such as decision trees, neural networks, support vectors machines, etc.) by comparing their accuracy in predicting students' performance, including retention and course grades.

Some studies have used machine learning approaches to predict students' retention and identify at-risk students. For example, Delen's [15] study tries to predict student retention by using five years of institutional data and different data mining techniques, including individual and ensembles models (bagging, busting, and information fusion). The individual models include artificial neural networks, decision trees, support vector machines, and logistic regression. The results show that the ensembles models have better performance than individual models for predicting student retention, with roughly 80% prediction accuracy. Furthermore, among the individual models, the SVM models have the best accuracy, followed by decision trees, neural networks, and logistic regression. Moreover, regardless of the models used, balanced data sets provide better results when compared to unbalanced data sets.

In a similar study, Lin et al. [16] used the Neural Network algorithm to predict engineering student retention based on cognitive and non-cognitive factors. The cognitive variables consist of 11 items that represent students' high school performance. Some non-cognitive variables are leadership, surface learning, teamwork, self-efficacy, motivation, meta-cognition, expectancy-value, and major decisions. Their results show that the applied neural network has a consistent predictive performance when trained and tested on different cohorts. Furthermore, a model that was built based on the 2004 freshman cohort's data kept its predictive power when tested on 2005 and 2006 cohorts. In another study, He et al. [17] applied two different Logistic regression models, named Simultaneously Smoothed Logistic Regression (LR-SIM) and Sequentially Smoothed Logistic Regression (LR-SEQ), to identify at-risk students in Massive Open Online Courses (MOOCs). At-risk students in their study were defined as students who are at risk of not completing online courses. They considered a basic Logistic Regression (LR) as a baseline and compared the results of their proposed models to this baseline. The results showed that the LR-SIM model has higher performance (AUC) when compared to the other models in terms of identifying at-risk students at an early stage.

Ghorbani et al. [18] compared the performance of different resampling methods that predicted student dropouts using two different data sets where the sets are imbalanced. The applied resampling models in this study were Random Over Sampler, SMOTE, SMOTE-ENN, SVM-SMOTE, Borderline SMOTE, and SMOTE-Tomek. Furthermore, in order to evaluate the performance of the mentioned resampling models with the imbalanced data set, different machine learning algorithms such as Random Forest, XG-boost, Support Vector Machine, Logistic Regression, Artificial Neural Network, and K-Nearest-Nighbor were used in this study. Using prediction accuracy, sensitivity, precision, and F1-score as performance evaluation metrics, the random forest classifier combined with the SVM-SMOTE resampling model have the highest performance in predicting student performance. In a similar study, Lagman et al. [19] applied ensemble models to predict student graduation in a Filipino university. The applied machine learning models in this study were Naive Bayes, Decision Tree, Neural Network, and Logistic Regression. The results demonstrated that adding the Boosting approach to the Logistic regression significantly increases the graduation rate prediction accuracy.

Many studies in the field of educational data mining focused on predicting students' course grades using their academic and demographic features [20, 21]. For example, Marbouti et al. [20] used different prediction methods, including Logistic Regression (LR), Decision Tree (DT), Support Vector Machine (SVM), Naive Bayes Classifier (NBC), Multi-Layer Perceptron Neural Network (MLP), and K-Nearest Neighbor (KNN) to predict student course grade early in the semester. The inputs for this model were homework grades, quiz grades, and midterm exams; the models' output was the students' grade letter at the end of the semester. Their results suggest that for courses with at least 120 students, the NBC has the highest total accuracy when compared to the other models. In a similar study, Huang et al. [22] used the grades for prerequisite courses (Statistic, Calculus I and II), three midterm exams, and students' cumulative GPA as inputs to predict students' final grades in an engineering Dynamics course. Using information from a pool of 323 students during four semesters, they show that the support vector machine had a better performance in predicting students' grades when compared to multiple linear regression, multi-layer perception network, and radial basis functions. In another study, [23] et al. applied logistic regression to predict if students will get a grade less than B in Physics 1 and 2 courses using data already available in the first week of the semester. They used several inputs such as students' cumulative GPA, homework grades, gender, race, and first-generation status. The proposed model was able to predict students' grades with 73% accuracy in Physics 1 and 81% accuracy in Physics 2.

1.1.2 Understanding factors that impact students' academic performance

While the main goal of the reviewed research in the previous section was predicting students' academic performance, the aim of this application domain is to understand the features that affect students' academic performance.

Some research has used clustering approaches to group students using significant features that can affect their academic performance. Grouping students in this way can guide investigations on where, why, and how universities should track sensitive groups and assign resources. For example, Marbouti et al. [5] used students' demographic information and past academic records to analyze students' success. They used a clustering method (K-means) to group students based on their academic and demographic features. Some of these features included financial status, enrollment status, first-generation status, housing status, and transfer status. Their results indicated that Hispanic, first-generation, low-income students have a lower GPA and take fewer classes per semester when compared to other students, which causes them to graduate with delays. In a similar study, Alfiani et al. [24] applied the K-means clustering method to classify students based on their demographic characteristics including origin and gender, and academic features such as course grades and the average of course attending. Comparing the academic performance of the clusters revealed that the grades of certain courses such as optimization and production planning is the most important feature to consider when clustering.

In another study, Khan [25] applied a clustering approach to examine the academic performance of 400 students, including 200 girls and 200 boys, at a secondary school in India. The study was aimed at investigating the impact of different factors such as personality, cognition, and demographic variables on student academic success. He used a cluster sampling technique to divide the data set into different clusters based on the mentioned variables. Then random samples were selected for analysis. The result of this study indicated that female students with high socioeconomic status have higher academic success when compared to other clusters. These results are confirmed in another study conducted by Goldrick et al. [26], where they claimed that students who come from families with low-level income are more likely to leave college without degrees. Their study also shows that offering additional financial aid to these students can significantly help them finish their program within four years. Similarly, several other studies show that students with low socioeconomic backgrounds are more likely to leave university or have a longer transition time from high school to university [27, 28, 29, 30]. Other demographic features that may affect students' performance include race [31], age [32], life changes including career and marriage [33], and gender [34]. All these studies show that students' demographic features have meaningful correlations with their academic success. Without such findings, academic institutions may fall short in identifying at-risk students early in their academic careers, and consequently fail to assign supporting financial and educational resources optimally.

Some other studies have tried to identify, extract, and analyze hidden patterns that might exist in students' academic behaviors to investigate their impact on academic performance. Sequence modeling and Hidden Markov models are two machine learning methods that are used extensively in this area to investigate students' academic behavior, such as enrollment trajectory and engagement. For example, many studies in the field of educational data mining have studied the impact of students' academic trajectories on degree completion. Jardins et al. [31] investigated the relationship between continuous and non-continuous enrollment on graduation rates. Analysis of the data collected from the University of Minnesota-Twin Cities shows that students with stop-outs (non-continuous or interrupted enrollment) are more likely to leave school without a degree when compared to students who continuously enroll at the university. In a similar study, Burley et al. [35] showed that college students who register for consecutive semesters with no interruptions have higher GPAs and graduation rates when compared to students who skip one or more semesters.

In another study, Crosta [36] investigated the impact of students' enrollment patterns on academic success using a clustering approach. The enrollment pattern in the study focused on two aspects: patterns of enrollment status, including full-time or part-time semesters, and patterns of enrollment continuity, including consecutive enrollment or skipping one or more terms. The results show a positive correlation between the number of full-time semesters and students' academic success. Also, students who enroll in their college without skipping terms are more likely to graduate than students with non-continuous enrollment. Bahr [37] developed a clustering approach based on students' courses and enrollment patterns during their academic careers. He used thirteen different variables, of which eight variables correspond to various forms of credit hours (i.e., transferable English units). The five remaining variables are the average number of units attempted per semester, number of noncredit courses, the percentage of courses completed, number of registered semesters, and the number of academic years. Attewell et al. [38] used the concept of *academic momentum perspective* to examine the effect of students' enrollment trajectories on degree completion. Their results indicate that the number of credits that students earn in their first semester creates an enrollment trajectory that has a positive correlation with degree completion.

Another academic behavior of students which has gained attention among education researchers over the past decade is academic engagement. Many studies have indicated that there is a strong relationship between student academic performance and student engagement. G.D. Kuh

[39] defines engagement as the amount of time students spend in collaboration with educational facilitates to enhance their knowledge and learning. The Community College Survey of Student Engagement (CCSSE) defines five student engagement benchmarks [40]: active and collaborative learning, student effort, academic challenge, student-faculty interaction, and support for learners. Additional studies focus on investigating the relationship between students' engagement and students' outcomes [41]. Research conducted using CCSSE data demonstrates that greater the student engagement correlates with better academic performance, including average GPA, retention, graduation rate, and persistence. For example, Price et al. [42] provides empirical evidence that shows student engagement with active and collaborative learning has a significant impact on student graduation rates.

1.1.3 Designing recommendation systems

Colleges and departments introduce prerequisites, GPA requirements, and curricula in an effort to improve student outcomes. Students seeking a degree at college often have difficulties choosing elective courses since some departments may propose various courses. Also, students might not have enough experience to decide which courses are the most suitable for them. Furthermore, students might not be aware of which courses could help them achieve their career goals; they may also not know if they have the required skills to pursue their goals. Traditionally, faculty propose prerequisites based on their experience and academic knowledge, which is subjective and may not always lead to the best results, impacting students' careers. The main goal of recommendation systems in EDM is to take advantage of machine learning techniques to best inform students regarding academic enrollment. These systems measure the academic impact of course sequencing on students' performance and propose the optimal order to lead to the highest course grades for students.

One recommendation system, named the degree compass recommendation system, is used to predict the sequence of courses through a post-secondary degree program in order to help students graduate [43]. For example, Morsy et al. [44] proposed an approach using a recurrent neural network (RNN) to create a course recommendation system intending to maintain or improve students' cumulative GPA. In another study, Wang et al [45] used demographic and academic features, including gender, personality, learning style, grades, and cognitive style as inputs for a random forest model to propose the best course sequence for students to improve academic performance.

Another course recommendation system known as the course agent system considers students' career goals and interests in the recommendation process [43]. For example, Upendran et al. [43] proposed a course recommendation system to identify courses that are important for students who want to get admitted to a college. While most of the predictions used students' career goals or current job trends as inputs for the prediction, they also considered students' grades and students' cognitive ability for the model's input. In a similar study, Farzan et al. [46] proposed a course recommendation system based on students' assessment of course relevance to their career goals.

A recommendation system based on association rules considers not only the past preference of students, but also the preferences of similar past students when recommending courses [47]. For example, Khorasani et al. [48] applied a Markov-based collaborative filtering model as a course recommendation system for students based on the sequence of courses they have taken in previous semesters.

1.2 Research questions and contributions

In reviewing the existing body of literature, it was found that many studies in the field of education analytics have identified students' course grades or GPA as an important indicators and predictors of students' final academic outcomes (graduation or halting). However, using GPA as to determine student outcomes may cause issues, since the grades that a student gets in any one semester might not necessarily reflect the true academic capability of a student. For example, a student might perform poorly in a semester because of other external factors, such as a change in grading policy, sickness, or other circumstances that cause a student to have a lower or higher GPA in a semester. Therefore, while semester-to-semester fluctuations in GPA are considered normal, significant changes in academic performance may warrant more thorough investigation and consideration, particularly with regard to final academic outcomes. In Chapter 2, it will be shown that using GPA and course grades to predict students' final academic outcomes, as done in the literature, is not able to distinguish between normal or significant changes in academic performance. There then is the assertion that there are academic-performance indicators that can distinguish normal changes from significant changes in students' academic performance.

In this dissertation, the Hidden Markov Model (HMM), an unsupervised machine learning approach, is used to define a students' academic features, termed the students' academic-performance level, to predict students' final outcomes. The HMM, first introduced in a series of articles by Leonard E. Baum in the second half of the 1960s [49], is a statistical approach in which a system is modeled through a stochastic process. Unlike ordinary Markov Models in which the states are directly observable, in HMMs, states' statuses are hidden. The aim is to estimate the hidden states' statuses using observations generated from the hidden states. Using HMMs, this dissertation provides a standard and intuitive classification of students' academic-performance levels, which leads to a compact representation of academic-performance trajectories. This dissertation contributes to this field of by using sequences of students' course grades for each semester to introduce students' academic-performance levels as an academic feature that may capture both normal and significant changes in students' academic performance. The relationship between different academic-performance trajectories and their correspondence to final academic success is also explored.

These aims are achieved based on different HMMs. In the literature, HMMs with different properties have been used based on the types of problems and their corresponding data structures. HMM's properties refer to the structure of HMM observation including observation type (e.g., discrete or continuous), observation dimension (e.g., univariate or multivariate), and HMM emission distribution (e.g., Categorical or Gaussian, etc.). Table 1.2 shows the gaps in the literature in terms

Table 1.2: Gaps in the literature

	Categorical	Poisson	Gaussian
Univariate	✓	✓	✓
Multivariate	✓	×	✓

of the number of observation dimensions and emission distributions. As shown the table below, HMMs with univariate observation and different emission matrices have been addressed in previous studies [50, 51, 52]. Also based on the table, while HMMs with categorical and Gaussian distributions and multivariate observations have been used in the literature, there is no work on the multivariate observations with a Poisson emission. Therefore, another contribution of this dissertation is to provide a multivariate hidden Markov model in which observations are generated from states with Poisson distribution.

This dissertation seeks to answer the following research questions within the context of higher education:

(RQ-1) Does the proposed academic feature (students' academic performance level) outperform the traditional features (i.e., students' semester GPA) in predicting students' graduate status?

(RQ-2) Does a decrease in academic performance proportionally increase the risk of halting?

(RQ-3) If a student improves their academic performance, does their likelihood of graduation equal that of students that always performed at higher academic-performance levels?

In the process of answering the above research questions, this dissertation develops a Multivariate Poisson hidden Markov model to address the research gap indicated in Table 1.2. Accordingly, a fourth research question is generated:

(RQ-4) What are the computational benefits and modeling benefits of a multivariate Poisson HMM when compared to a standard empirical HMM?

This dissertation aims to answer the above-mentioned research questions by proposing a new academic feature, students' academic-performance level. Then, after generating students' academic trajectories during their academic careers based on the proposed feature, the correlations between students' educational features (i.e., halt rate) and different academic trajectories are investigated using hypothesis tests. This aims to find the students' academic-performance trajectories which are associated with the most halt rates, and accordingly identify at-risk students. The obtained results can help students improve their academic performance and guide educators and university policy-makers in assigning their educational and financial resources efficiently to the most in-need students.

1.3 Outline of the dissertation

The remainder of the dissertation is organized as follows: First, Chapter 2 begins by providing a review of machine learning techniques commonly used in educational data mining. In this chapter, a short-fall analysis of machine-learning techniques is performed to highlight the strengths and weaknesses of machine learning techniques when modeling the impact of academic performance on student graduation rates. To this end, six different machine learning methods are introduced and applied to University of Central Florida (UCF) student data records to predict students' graduation/halt status. Two different studies are conducted to discuss two limitations that these techniques have when predicting students' academic outcomes. Finally, based on another study, shortcomings of traditional features used in EDM as factors that impact students' academic performance are investigated. Overall, this chapter provides motivation to answer the mentioned research question.

Next, Chapter 3 provides an introduction and description of the academic data records from UCF used throughout this dissertation. Descriptions of relevant data fields are provided; additionally, this chapter provides a description of common measures that are used to evaluate student academic performance (i.e., course grade, graduate/halt, etc), with a brief analysis correlating demographic features to final academic outcomes. Then, after stating shortcomings associated with the data set, the chapter ends by reporting key statistical measures related to UCF. Providing the statistical measures gives a greater context regarding the student population and notes unique features which may limit the transferability of this dissertation's research findings to other universities.

Chapter 4 intends to answer research questions 2 and 3. In the chapter, a new academic feature, named the academic performance level, is introduced using a Hidden Markov model. This new measure seeks to estimate the true academic-performance of students, regardless of mild semester-to-semester deviations in course grades. Then, after using the introduced academic-performance levels to classify students based on their academic performance trajectories, the correlation between different academic performance trajectories and halt rates is investigated. Furthermore, some hypothesis tests are conducted to see how students' demographic features, such as gender and race/ethnicity, correlate with students' academic-performance levels. At the end of the chapter, the limitations of the proposed HMM are discussed to highlight the need for an alternative approach to tackle those limitations.

Chapter 5 provides the formulation of the Hidden Markov models algorithms, including the Forward algorithm, the Backward algorithm, the Baum-Welch algorithm, and the Viterbi algorithm. While mathematical formulations of the mentioned algorithms are clear, there exist challenges (i.e., underflow) with the computational implementation. Thus, the limitations in the computation implementation of Hidden Markov models are discussed. To support subsequent chapters, two solutions, log-sum-exponential trick and log space formulation, are proposed to overcome computation limitations. Finally, pseudo-code is provided that enables the implementation of more complex HMM not currently supported by existing software libraries in Python.

In Chapter 6, the use of a Multivariate Poisson Hidden Markov Model (MPHMM) is proposed to overcome the mentioned limitations of the applied basic HMM in Chapter 4. In this chapter, after introducing the MPHMM, its formulation and implementation along with the required replacement in the proposed pseudocode are explained. Then, some case studies are simulated to compare the performance of the proposed MPHHM and the basic empirical HMM for both Baum-Welch and

Viterbi algorithms. This chapter answers research question 4 using simulation approaches.

In Chapter 7, the proposed MPHMM is applied to UCF student data records to compare the results with the findings from the basic empirical HMM in chapter 4. These comparisons will be conducted from qualitative and quantitative points of view. Thus, research question 4 is answered using a real data set. Finally, in order to address the first research question, the proposed academic-performance level obtained by the MPHMM and the basic HMM along with the traditional features (e.g., semester GPA) are used as inputs for different logistic regression models to determine which of these features have a better performance in predicting students graduation/halt status.

Finally, the dissertation is concluded in Chapter 8 with a summary of the findings. After presenting the goals, approaches, and key findings for all chapters, the contributions of this dissertation are discussed.

CHAPTER 2
SHORT-FALL ANALYSIS IN PREDICTING STUDENTS' PERFORMANCE AND OUTCOMES

One prominent focus in educational data mining is the application of machine-learning techniques to predict student performance. Such outcomes can be binary or discreet categorizations, such as graduating/halting, passing/failing a course, or final major. Alternatively, machine learning techniques may also produce continuous measures like GPA or time-to-graduation. A related, but fundamentally distinct focus area of EDM research seeks to understand factors that impact student performance. This type of research is particularly beneficial to university stakeholders when creating policies, support systems, and intervention programs that target at-risk students to improve overall performance (i.e., retention and graduation rates). A review of prior work on the subject is provided in an effort to understand the differences between these two research approaches. After reviewing prior work, a subset of the associated machine-learning techniques are applied to a student-record data set to evaluate performance. In doing so, the strengths and weaknesses of the techniques and the underlying approaches are evaluated, highlighting the need for improved modeling approaches.

2.1 Predicting student outcomes

Machine learning in higher education is used to improve retention and graduation rates at academic institutions, with most of these applications using machine learning to make predictions for individual students. While these approaches are useful in some cases, for the most part they do not help university policy-makers at scale. Universities are often more interested in developing policies or finding mitigation strategies that target groups of students, as opposed to predicting whether an individual student will pass or fail. In this section, the following items from previous research are investigated: machine learning models commonly used in such studies, model inputs and outputs, model performance evaluation, and their findings. These studies are conducted from the years 2010 to 2021.

Some studies have applied machine learning models to identify at-risk students. For example, Hu et al. [53] developed an early warning system to inform students who are at risk of leaving online courses before completion. In their model, they considered time-dependent variables as their model inputs to predict student performance. Some of these variables were the number of logins in the system, total time online, average time per session, total time viewing material, number of assignment delays, and number of incompleted assignments. They applied three machine learning models, namely Logistic Regression (LR), Adaptive Boosting (AdaBoost), and Classification and Regression Tree (CART), of which, the CART model had better performance in terms of the type I and type II errors.

In another study, Lin et al. [16] used the Neural Network algorithm to predict engineering students' retention based on cognitive and non-cognitive factors. The cognitive variables consisted of 11 items that represent students' high school performance. Some non-cognitive factors include leadership, surface learning, teamwork, self-efficacy, motivation, meta-cognition, expectancy-value, and major decision. Their results show that the applied neural network has a consistent predictive performance when trained and tested on different cohorts. Furthermore, the model, built based on the 2004 freshman cohort's data, kept its predictive power when tested on 2005 and 2006 cohorts.

In a similar study, Lakkaraju et al. [54] developed a machine learning framework to identify at-risk students. In their research, at-risk students are defined as students who are at risk of failing to complete their program or completing the program beyond the required time to graduation. Five different machine learning models, including random forest, logistic regression, AdaBoost, support vector machine, and decision tree, were applied to a data set collected from two different districts in the U.S. which contain about 200,000 students . Using recall, precision, and AUC for the models' performance evaluation, results indicated that the random forest model has a higher performance when compared to other models in identifying at-risk students.

Other studies have used machine learning techniques to predict students' course grades. For example, Marbouti et al. [20] applied six different prediction methods, including Logistic Regression (LR), Decision Tree (DT), Support Vector Machine (SVM), Naive Bayes Classifier (NBC), Multi-Layer Perceptron Neural Network (MLP), and K-Nearest Neighbor (KNN) to identify at-risk students early in the semester by predicting their grade letter in a given course. The inputs of their model were the grades for homework, quizzes, team participation, design projects, mathematical modeling activities, and midterm exams; and the models' output was the students' grade letters at the end of the semester. They used the concepts of total accuracy, recall, precision, and F-1 score to evaluate the model's performance. The results suggested that for courses with at least 120 students, NBC has the highest performance compared to the other models.

In a similar study, Iqbal et al. [55] applied three machine learning methods, Collaborative Filtering (CF), Matrix Factorization (MF), and Restricted Boltzmann Machines (RBM), to predict students' grades in particular courses. The data set was collected from the Information Technology University; students with a bachelor's degree in Electrical Engineering were considered for this case study. Iqbal et al. [55] also proposed a feedback model to inform students if they needed to put more effort into their courses based on the predicted grades. Using the Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE) to evaluate the model's performance, it was found that RBM has the highest accuracy when compared to the other proposed models for course grade prediction.

Predicting whether a student will pass or fail a course is another area where machine learning is used in many studies. For example, Ahmad et al. [56] used machine learning models to predict academic performance for first-year students in a computer science course. The data set, collected from the UniSZA in Malaysia during the years 2007 to 2014, contains academic and demographic information. Naive Bayes, Decision Tree, and Rule-Based classification were applied to the data set to predict if students would pass or fail a course. Some of the inputs in these models were race, gender, family income, English grade, Malay languages, hometown, and high school GPA. Results showed that the rule-based classifier has the highest accuracy when compared to the Naive Bayes and Decision Tree classifiers.

Osmanbegovic et al. [57] compared the performances of three different machine learning models, Multilayer perception (MLP), Naive Bayes (NB), and Decision Tree (DT), to predict students' success, where success was defined as passing a given exam. The data set was collected from the University of Tuzla during the 2010-2011 academic year, and contained demographic and academic information of first-year students from the department of economics. The results illustrated that the Naive Bayes model outperforms both the Decision Tree and Multilayer perception models in terms of total accuracy.

In similar research, Kotsiaritis et al. [58] used ensemble models to predict students' final examination score with two values, pass or fail, in the Hellenic Open University. The ensemble models in this study used a combination of three different machine learning models, namely Naive Bayes (NB), Neural Network (NN), and WINNOW (a linear online algorithm similar to NN). The input of the models was the grades of four different written assignments and the output was a passing or failing grade for the course. Their results indicated that the ensemble model had a higher accuracy (lower error rate) when compared to other machine learning models such as random forest, AdaBoost, and voted perceptron models.

2.2 Identifying factors that effect students outcomes

The previous section analyzed some papers aimed at predicting students' academic performance using machine learning models. In this section, additional studies that focus on identifying factors that impact students' retention are reviewed. Understanding these factors can help university policy-makers in proposing strategies that reduce performance gaps that may exist between student demographic factors such as gender, age, family income, etc. Essentially, if these performance gaps are discovered, then appropriate interventions may be identified and applied to reduce the performance gaps.

Many studies in the field of EDM have identified academic features, including GPA, course grades, enrollment type, and student learning behaviors as significant factors that impact students' outcomes. For example, Ojha et al. [10] investigated the impact of academic and demographic features to predict students' graduation delay at the University of New Mexico (UNM) using machine learning techniques. They used Support Vector Machines, Gaussian Processes, and the Deep Boltzmann Machines for their predictions. The inputs of these models were high school GPA, composite ACT score, gender, first and second semester GPA in UNM, number of credits taken by the students before the second semester, and gender. These models classified students into three categories: students with no delay in graduation, students with one year delay in graduation, and students with more than one year delay in graduation. The results showed that semester GPA and cumulative credit hours are the most significant factors in the prediction of graduation delay.

Plagge [9] used two different artificial neural networks to identify factors which impact first-year to second-year retention rate based on a data set from Columbus State University from the years 2005 to 2010. Some of the models inputs are students' age, gender, race, in-state status, distance from home, high school GPA, estimated family contribution, parent's highest education level, Fall semester core course count, and Fall GPA. The dependent variable was a binary variable that indicates if a student registers for the second Fall semester or not. Their analysis indicated that the Fall semester core course count and Fall GPA are the most significant factor that affects first-year to second-year retention rate.

Ahmed et al. [59] employed two different kinds of Decision Trees (DT) to predict students' final grades in a given course and identify the most significant factors that impacted the students' grades. They considered ten different variables as the model input, including the students' department, high school degree of students, midterm grade, lab test grade, seminar performance, class attendance, and so on. They used the concept of Information Gain (IG) and entropy to determine

Table 2.1: Online learning behavior variables

Learning Process stage	Variables
Preparation	Number of course Introduction views
	Number of course register
	Number of course login
Progress	Number of course progress page checked
Resource learning	Time of resource watch
	Completion of resource watch
	Number of resource repeated watch
	Degree of resource repeated watch
Forum interaction	Number of forum browse
	Number of forum post
	Number of forum reply

the most significant variable that affected students' final grades. The results demonstrated that midterm grades have the highest information gain and was considered the root node in the applied decision trees.

Zheng et al. [60] investigated the relationship between students' learning behavior and their academic performance in an online course. The learning behavior was classified into different learning processes and variables were considered for each. The learning process includes the following stages: preparation, progress, resource learning, forum interaction, and test. Variables for each stage are shown in Table 2.1. Using logistic regression and correlation analysis, the study identified the factors which were significant in students' success prediction. For example, results indicated that most students who started to watch course materials a few days before the test have a lower grade when compared to students who viewed course materials over a longer period of time.

Enrollment type, including full-time and part-time, is another important academic feature used in EDM which affects students' performance. For example, Crosta [36] investigated the impact of student enrollment patterns on student dropout using a clustering approach. The enrollment pattern in the study focused on two aspects: patterns of enrollment status, including full-time or part-time semesters, and patterns of enrollment continuity, including consecutive enrollment or skipping one or more terms. The results showed a positive correlation between the number of full-time semesters and students' academic success. Also, students who enroll in their college consecutively without skipping terms are more likely to graduate than students with non-continuous enrollment.

Demographic features such as gender, age, family income, and first-generation status have been used widely in EDM as significant features that impact students' academic performance. For example, Kovacic [61] applied machine learning algorithms to predict student dropout rates at the Open Polytechnic of New Zealand using socio-demographic and academic features. After applying a feature selection method to identify input variables with a strong relationship to the dependent variable, three different decision trees, namely CHAID, exhaustive CHAID, and QUEST were applied to predict student dropout. In this research, results indicated that some background information such as student gender, disability, and work status are significant factors to categorize students into successful and unsuccessful groups.

In another study, Djulovic and Li [62] applied classification approaches to predict the retention rates for freshman students at Eastern Washington University. Four different predictive models including Naive Bayes, Neural Network, Decision Tree, and Rule Induction were applied to the data set. Age, gender, high school GPA, student SAT score, cumulative GPA, and financial aid status were the input variables and the dependent variable was if freshman students were retained in the next year. Among all the models, the Naive Bayes classifier had the highest negative recall rate. Their analysis indicated that student financial aid status is a significant variable that has a considerable effect on student retention.

A comprehensive model with different features is presented by Morell et al. [63]. They studied the effect of different features on university GPA. These features are high school GPA, level of high school education, teacher experience, ethnicity, family income, and gender. Their research seeks to find underlying factors which impact student performance at a major public university. There are four sets of explanatory factors: The degree program in which the students are enrolled, the background of students' families, the traits of schools at which students graduate, and the demographics of the high school at which the students attended. The results show that personal background, including gender, ethnicity, and family income has a significant impact on the university GPA.

2.3 Limitations of traditional models and features in EDM

In the previous section, some applications of machine learning methods in EDM research were briefly described and reviewed. A common theme in each of these studies is examining how demographic and academic features impact student academic performance. In this section, shortcomings of the previous studies in predicting final student outcomes are discussed from two different views: (1) the methods themselves, and (2) the features by which the methods are trained. A performance analysis is performed by examining three specific modeling problems that consider various ways that academic features are represented when applying machine-learning techniques.

To begin, six different machine-learning techniques are considered; these techniques are commonly applied to identify students who are at risk of halting enrollment. Students are considered as halted if they leave school without a degree or graduate in more than six years (the 6-year graduation rate method). These studies are conducted based on records of 36,261 first-time-in-college (FTIC) students who started their education at UCF during the years 2008, 2009, and 2010. Short descriptions of the applied models are provided as follows:

- **Random Forest (RF)** is a reliable machine-learning technique used for both classification and regression problems. This technique utilizes ensemble learning to construct a group of decision trees that partitions data based on attributes (i.e. variables) until all data in each node has one class label (in this problem, halt or graduate status).

- **Support Vector Classifier (SVC)** uses hyperplanes and support vectors to classify data points by solving an optimization problem that maximizes the margin of each class. One key advantage of SVCs is that they are computationally effective when working with high dimensional linear and non-linear data; however, this efficiency degrades with larger data sets. Non-linear SVC is used to find an optimum surface when the two classes are not linearly separable.

- **Logistic Regression (LR)** is another popular model in educational data mining research. While linear regression models are used in regression problems, logistic regression models are the corresponding standard model used to address classification problems. Logistic Regression classifies data points by computing the probability of each class label (e.g., graduate/halt) for each data point. One major benefit of Logistic Regression models is that their corresponding coefficients are interpretable and well understood; as such Logistic Regression is not considered to be a *black box* technique.

- **Gaussian Naive Bayes Classifier (GNB)** is another probabilistic classifier that classifies data points by computing a conditional probability distribution; this approach utilizes the Bayes rule and assumes the input variables are independent of each other. While in most cases this assumption is not true (for example the number of A's and W's a student earns is not independent), previous studies have shown that the performance of GNBs is similar to other advanced machine learning models such as SVC [64].

- **Gradient Boosting (GB)** is used for both regression and classification tasks based on ensemble learners. The technique essentially works by recursively stacking simple models that address and correct the residual errors of prior models.

- K**-Nearest Neighbor (KNN)** is a non-parametric algorithm that is used for both classification and regression problems. This method classifies data points by a majority vote of its K-neighbors. Therefore, instead of estimating model parameters, the algorithm only computes the distance between data points and classifies them based on the distances.

The six machine-learning techniques above will be analyzed by considering their application to a series of related studies identifying first-time-in-college (FTIC) students who are at risk of halting their enrollment based on early college performance (i.e. first two years of grades). Here, halting is defined as two consecutive years without enrolling for classes (either full-time or part-time). Each application of the machine-learning techniques will take a set of features associated with academic performance as its input and return a prediction of halting or graduating; the demographic attributes of students and other relevant features (e.g. part-time vs full-time enrollment status) are not considered in these models. While the lack non-academic factors is clearly a limitation, the purpose of these studies is to highlight that fundamental issues remain with algorithms and representations of academic performance measures when making classification predictions like halting or graduating.

Predicting final outcomes based on a cumulative grade-count model. The first study under consideration develops prediction models that take as their input the cumulative number of A's, B's, C's, D's, F's, and W's that a student obtains over their first two academic semesters to predict if the student will halt enrollment or graduate. The required inputs and output of the model set-up are illustrated in Figure 2.1; as shown, the machine-learning models have six numerical inputs corresponding to grade counts.

The data set used in this study to train the machine-learning models is based on 36,261 anonymized first-time-in-college (FTIC) student records from the University of Central Florida (UCF), a large public 4-year institution (see chapter 3 for additional contextual information regarding UCF). As indicated by the student records data-set and publicly available data, the majority of UCF students graduate with a degree (75% vs 25%, see chapter 3). The grade count distribution also skews towards successful outcomes corresponding to higher letter grades. Based on Figure 2.2, the distribution of A, B, C, D, F, and W (withdrawal) grades, for students at UCF is 44.7%, 31.8%, 13.7%, 3.1%,3.5%, and 3.2%. The grade distribution considers both students that halted their enrollment or graduated from UCF.

With the broader goal of increasing persistence and graduation rates, when analyzing the six machine-learning models greater attention is paid to their performance in correctly identifying

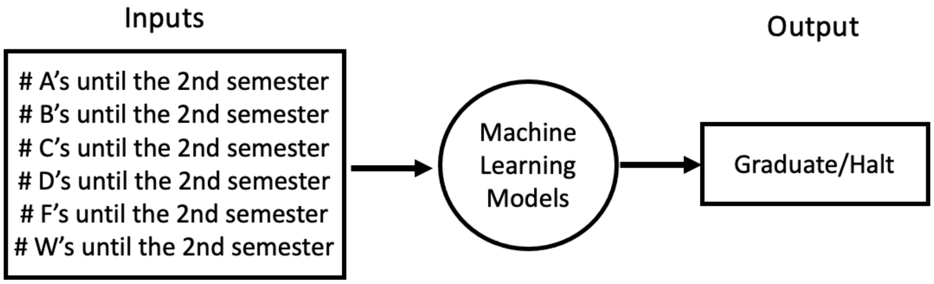

Inputs
# A's until the 2nd semester
# B's until the 2nd semester
# C's until the 2nd semester
# D's until the 2nd semester
# F's until the 2nd semester
# W's until the 2nd semester
Machine Learning Models
Output
Graduate/Halt

Figure 2.1: Inputs and output for the applied machine learning for predicting students' performance

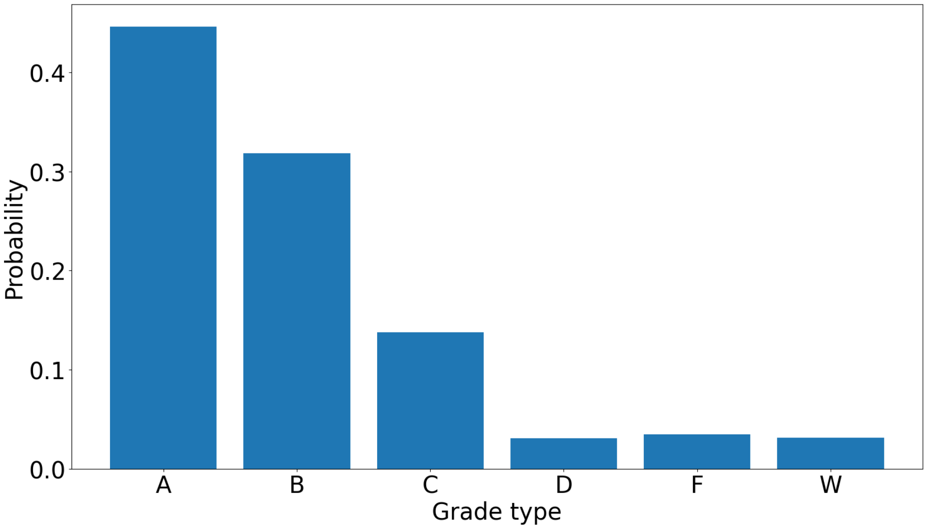


Figure 2.2: Grade count distribution for students at UCF

students that halt their education. This measurement objective is reasonable given that the majority of students ultimately graduate, as such, any predictive model should have an accuracy rate of at least 75%. Furthermore, there is limited value in correctly identifying high-performing students that will graduate given their traditionally high graduation rates (see chapter 4 and chapter 7).

The UCF data set is divided into training and testing data sets. The training data set is used to build the models, while the test set is used to evaluate the models' performance. Evaluating the various models makes use of total accuracy, halt accuracy, and graduate accuracy; the corresponding equations for each metric are provided in Equation 2.1 through Equation 2.3. For all models, the F_1 score, which is a harmonic mean of recall and precision, is also computed (Equation 2.4). Here, the recall refers to the accuracy of predicting students who halted enrollment. Meanwhile, precision refers to the fraction of students correctly predicted to halt out of all students predicted to halt.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total number of students}} \tag{2.1}$$

$$\text{Halt-Accuracy} = \frac{\text{True Positives}}{\text{Total number of halt students}} \tag{2.2}$$

$$\text{Graduation-Accuracy} = \frac{\text{True Negatives}}{\text{Total number of graduated students}} \tag{2.3}$$

$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision + Recall}} \tag{2.4}$$

In the equations above, True Negative refers to the number of students who graduate that are not identified as students at-risk of halting. The number of True Positives is the number of students who halt enrollment and are identified as at-risk students. Meanwhile, False Negatives are those students who halt enrollment but are classified as graduating by the predictive models. Finally, False Positives refer to the number of students who graduated but are identified as at-risk stu-

Table 2.2: Results for the machine learning models on the test set based on students' performance during two semesters (FTIC students)

Models	RF	SVC	LR	GNB	GB	KNN
Total Accuracy	80.1%	81.6%	81.7%	80.7%	81.6%	76.4%
Halt-Accuracy	48.5%	47.3%	48.7%	54.5%	48.9%	50.5%
Graduation-Accuracy	92.8%	95.5%	95.0%	91.2%	94.9%	86.9%
F1 Score	58.3%	59.7%	60.5%	61.9%	60.5%	55.2%
True Negative	7910	8136	8099	7774	8084	7407
True Positive	1670	1629	1678	1878	1685	1739
False Negative	1776	1817	1768	1568	1761	1707
False Positive	611	385	422	747	437	1114

dents. Typically, False Positives and False Negatives are known as type I error and type II errors, respectively.

Table 2.2 provides summary statistics of the six machine-learning models trained based on count grades. The logistic regression model, serving as a baseline given its simplicity and propensity in EDM research, has a total accuracy of 81.7%. The halt-accuracy, graduation-accuracy, and F_1 score for the logistic regression model is 48.7%, 95.0%, and 60.5% respectively. With a graduation-accuracy of 95.5%, the support vector classifier has the best performance when predicting graduation. While this model has the highest total accuracy among all the machine learning models, it has the lowest halt-accuracy when compared to the other models (47.3%). In contrast, the Gaussian Naive Bayes classifier model has the best performance when identifying at-risk students among all machine learning models; it has a halt-accuracy of 54.5%. However, the total accuracy and graduation-accuracy for this model are 80.7% and 91.2% respectively.

As a point of interest, the same study above is repeated using training and testing data sets based on transfer students at UCF. As before, student grade counts for the first two semesters are used in order to predict halting or graduation. Table 2.3 provides the summary statistics for the six standard machine-learning techniques that are applied when building the prediction models. Comparing the results in Table 2.2 and Table 2.3, the machine-learning models have a higher prediction accuracy

Table 2.3: Results for the machine learning models on the test set based on students' performance during two semesters (Transfer students)

Models	RF	SVC	LR	GNB	GB	KNN
Total Accuracy	85.5%	86.3%	86.4%	84.8%	86.4%	81.5%
Halt-Accuracy	59.0%	57.2%	59.2%	64.7%	59.6%	59.3%
Graduation-Accuracy	94.3%	95.9%	95.4%	91.4%	95.2%	88.8%
F1 Score	66.9%	67.5%	68.5%	67.9%	68.4%	61.4%
True Negative	14879	15134	15059	14432	15017	14011
True Positive	3076	2985	3089	3374	3109	3091
False Negative	2137	2228	2124	1839	2104	2122
False Positive	903	648	723	1350	765	1771

for transfer students when compared to FTIC students. For example, the halt-accuracy for the Random Forest model increases from 48.8% to 59.0% for transfer students when compared to FTIC students. On average, the models have 10.15% higher halt-accuracy for transfer students compared to FTIC students. However, similar to the model trained based on FTIC students, SVC has the highest graduation-accuracy among all models applied to the transfer students (95.9%), but the lowest halt-accuracy when compared to other models (57.2%). Furthermore, based on two tables, GNB has the highest accuracy among all models in predicting halted students (54.5% and 64.7% for FTIC and transfer students, respectively).

As tools, the six machine learning models struggle to surpass a 50% accuracy when predicting halting. Their limitations can be traced to numerous factors, some of which include the exclusion of demographic and other non-academic features (e.g. financial aid). Regardless, the prediction models have a more important flaw in that they do not account for possible interventions or changes that students might make in their educational careers in response to their performance. The models also make an implicit, yet fundamentally wrong assumption that halting is in response to academic performance. So while university policies may dictate that students with poor academic performance be placed on academic probation, and ultimately subject to disqualification, for the most part, low grades only correlate with the decision to halt. In many cases, the decision to halt can be

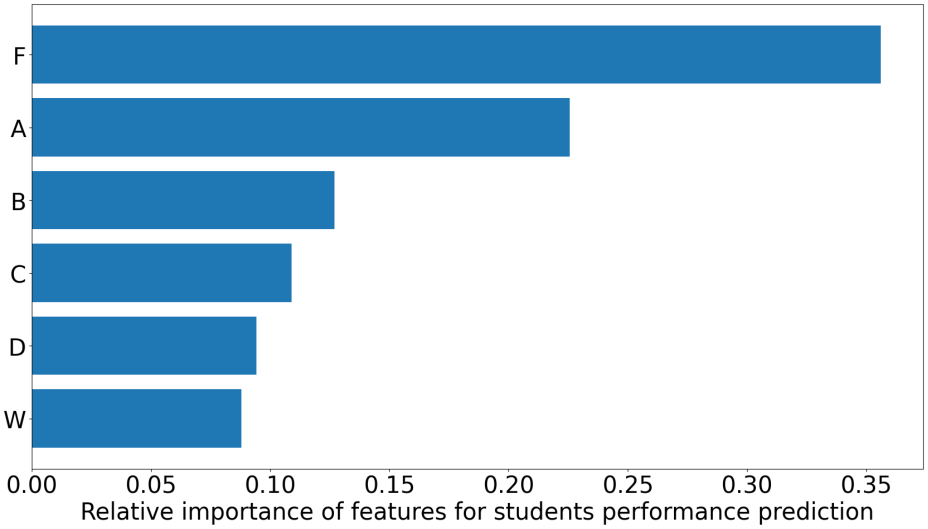


Figure 2.3: Relative importance of features for students' performance prediction based on students' records during two semesters using random forest (FTIC students)

tied to other reasons like finances, work, family obligations, stress, lack of support, etc. As such, low academic performance is at best a proximate cause or in most cases a byproduct of the true reasons for halting college. Based on this argument, better usage of machine-learning models is not for prediction but rather for understanding correlations and factors that influence halting.

One way to understand the factors that influence student outcomes is to perform feature importance or perturbation analysis on the underlying models. This process is usually straightforward for linear regression models and logistic regression models, as feature importance is closely tied to the magnitude of model coefficients. However, this approach is not without limitations, especially when features are co-linear. Programming implementation of Random Forest models (e.g. the Scikit-learn's RandomForestClassifier module in Python) often includes the necessary tools to perform a feature importance analysis. Figure 2.3 illustrates the relative feature importance for predicting if a student halts when using the Random Forest model. Based on the results, the numbers of F's and A's that students make during their academic careers are the two most important features that correlate with final academic outcomes (halt/graduate). As seen in the figure, the number of W's has the lowest effect on the students' halt/graduate status, since the W grade in this research includes medical withdrawals, which is not necessarily an academic performance indicator.

For the remaining five machine-learning models, Python's Scikit-learn library does not have this same ability to calculate feature importance. As such, an alternative approach for calculating the feature importance is used for all six machine-learning models based on a feature drop-out approach. For each of the six machine-learning techniques, seven different models are constructed: one using the full data set, and the remaining six models are constructed using a subset of the data features. As such, the first model variant is trained using all features (i.e. grade counts); it is considered to be a full model. For the other six model variants, one feature is removed and the machine-learning model is trained based on the remaining data features. For example, in model/X, grade counts for X are excluded when training the model, where X is A, B, C, D, F, or W. The relative importance of each grade is computed based on the following equation:

$$\text{Relative importance (F)} = \text{Accuracy(Halt}|\text{Full model)-Accuracy(Halt}|\text{Full model/F)}$$

Based on the formula, the relative importance for each grade is the difference in the halt-accuracy between the full model and halt-accuracy of the model variant trained with fewer features. For example, for the SVC model, the halt-accuracies for the full model and the model variant without the F grade counts (model/F) are 47.3% and 42.2%, respectively. The relative importance of this feature (number of F grades) corresponding to a drop in halt-accuracy is 5.1 (47.3-42.2) for SVC. Figure 2.4 provides the relative importance of all machine-learning models and variants. Based on the figure, as was shown for the random forest, excluding the number of F's students earns in the first year has the greatest impact on the halt-accuracy, and as such it is considered to be the feature with the greatest relative importance. The number of course withdrawals is the second most important feature for the SVC, LR, and KNN models. However, for the GNB and GB models, the second most important grades are D and A grades, respectively. The number of B and C grades are usually the least informative for all the models. While the number of A grades has

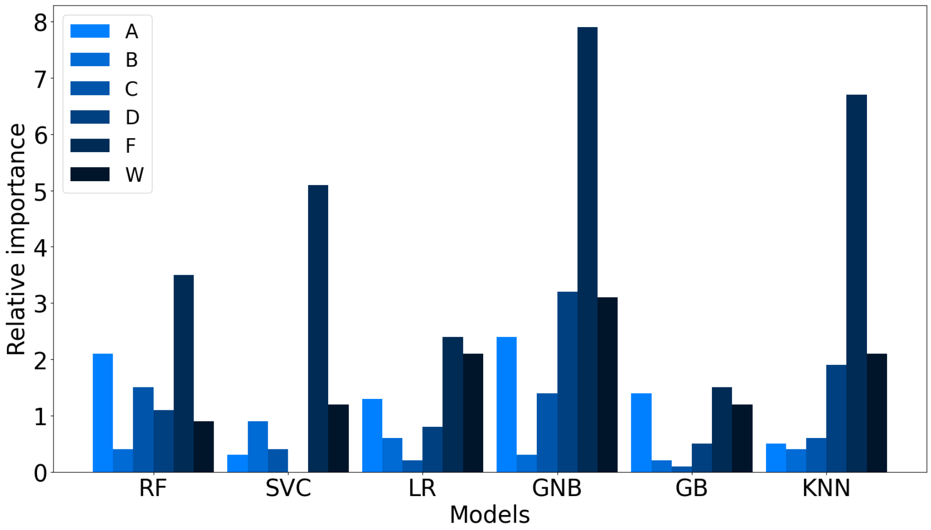


Figure 2.4: Relative importance of features for students' performance prediction based on students' records during two semesters for all models (FTIC students)

some importance for the LR, GB, and GNB models, the feature is not predictive for SVM when focusing on halt-accuracy. Unfortunately, due to the black-box nature of these techniques, it is not clear why this distinction exists.

Most of the papers reviewed earlier in this chapter are aimed at predicting students' performance using demographic and academic features. One main limitation common to many of the models is the timing of the predictions. While early predictions after one or two semesters may have low prediction accuracy, a late prediction may not provide sufficient time to intervene for at-risk students. In order to investigate the trade-offs between early detection and improved model accuracy of later predictions, the six machine-learning techniques are retrained based on 1, 2, 3, 4, and 5 semesters worth of grade data to produce a total of 30 model variants. The accuracies of the model variants are then compared to each other to quantify the improvements in prediction accuracy according to the number of semester grades considered. Figure 2.5 illustrates the impact of the number of semesters on the halt-accuracy for the various machine learning models. As was expected, the more semester grade-data is fed into the training data set, and are ultimately used as part of the prediction model, the better the halt-accuracy. For example, for the Random

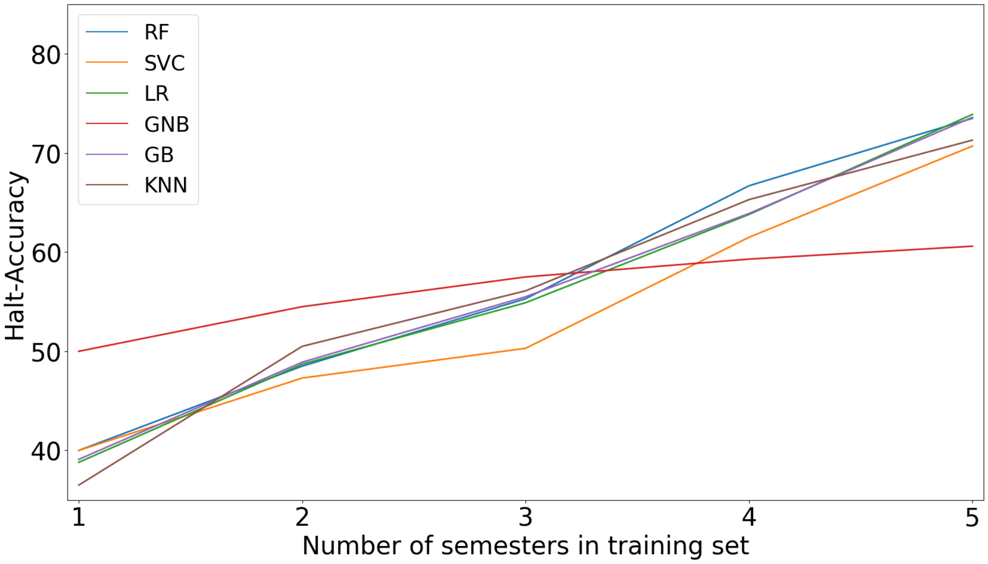


Figure 2.5: Impact of the number of semesters in the training set on accuracy (halt) for FTIC students

Forest model, when considering four semesters worth of grades, the halt-accuracy improves from 40.0% to 73.5%. A similar pattern is observed for the other machine-learning models. Similarly, as shown in Figure 2.6, the improvement in halt-accuracy as the number of semesters is increased is also observed when developing a halt-prediction model for transfer students. Interestingly, the improvement in halt-accuracy is smaller for the GNB model. So while the GNB model has the highest halt-accuracy when making predictions based on two semesters of grade data, it has the lowest accuracy when considering 5 semesters of grade data.

Predicting final outcomes based on a semester grade-count model. In the previous study, the cumulative number of A's, B's, C's, D's, F's, and W's that a student obtains until a given semester (i.e., the second semester) were used as inputs to predict if the student halts enrollment or not. Here, a similar model is developed, but instead of using cumulative grade counts over semesters, the semester grade counts serve as the input to the prediction model. Using this input structure, for a model based on N semesters, there will be $6N$ inputs used for the new model structure, while the previous cumulative grade-count models had six inputs. Figure 2.7 depicts the inputs for the proposed model structure under consideration. An advantage of a semester grade-count model

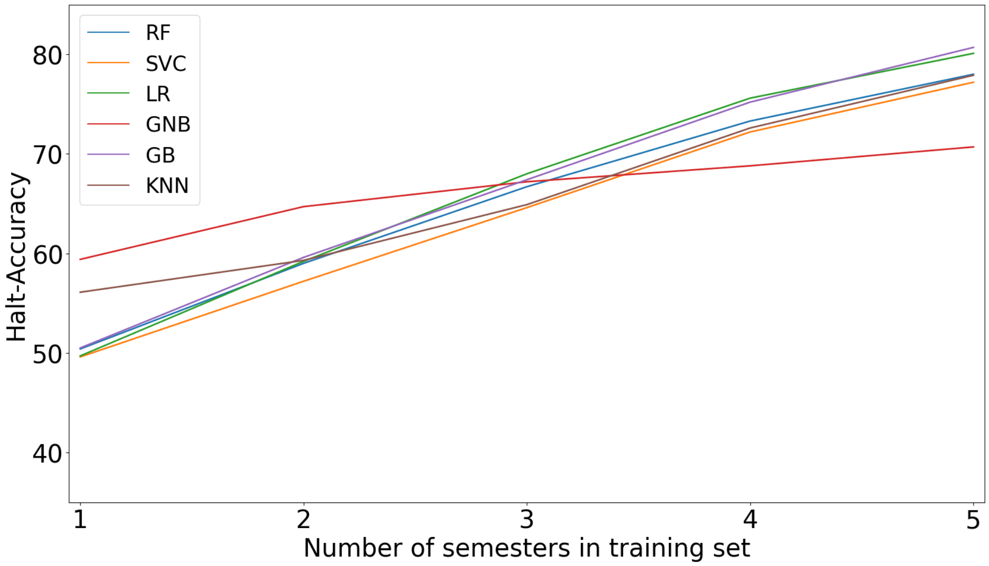


Figure 2.6: Impact of the number of semesters in the training set on accuracy (halt) for transfer students

over the previous model structure is that semester grade-count models can take into consideration if performance in a specific semester is more predictive of the final academic outcome. Additionally, a semester grade-count model may be able to account for improvement or deterioration of academic performance when making predictions as semester grade data is separated, not aggregated.

The same six machine-learning techniques are used to build prediction models based on the new inputs to see how this change impacts the models' performances. Accuracy results for the models are summarized in Table 2.4. By comparing Table 2.2 and Table 2.4, it is seen that while the halt-accuracy has been slightly increased for the RF model (from 48.5% to 50.9%) and in the GNB model (from 54.4% to 58.0%), there is no significant change for the remaining machine-learning models as the difference is less than 1%. In summary, changing the models' inputs in this problem does not change the models' performance significantly. The lack of meaningful improvement for four of the models suggests that addressing research questions RQ-2 and RQ-3 may not yield much value, as there appears to be a limited benefit of considering time-sequences of grade data. By extension, such a finding may imply that if a student's grades decrease from high to low and another student's grades increased from low to high, meaning each has the same cumulative GPA,

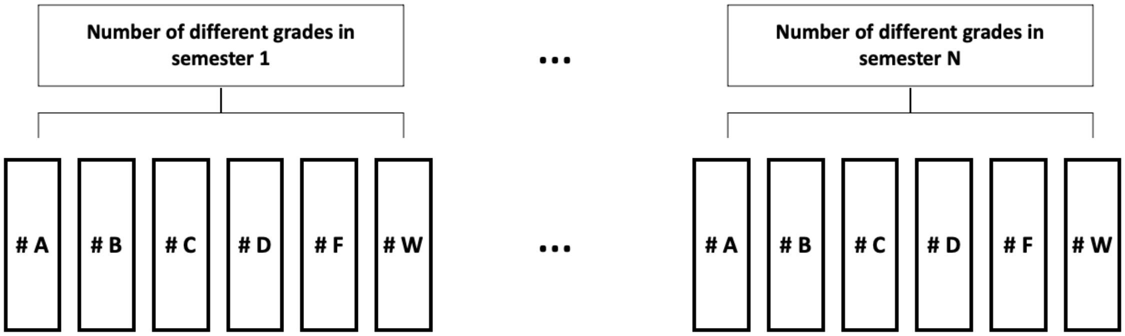

Number of different grades in semester 1
Number of different grades in semester N
# A
# B
# C
# D
# F
# W
# A
# B
# C
# D
# F
# W

Figure 2.7: Inputs of the model created based on the grades number in N semesters

Table 2.4: Results for the machine learning models on the test set based on students' performance during two semesters separately (FTIC students)

Models	RF	SVC	LR	GNB	GB	KNN
Total Accuracy	80.0%	81.5%	81.7%	80.2%	81.7%	73.2%
Halt-Accuracy	50.9%	47.6%	48.8%	58.0%	49.2%	51.3%
Graduate-Accuracy	90.9%	95.4%	95.0%	89.2%	94.9%	82.1%
F1 Score	58.7%	59.1%	60.5%	62.8%	60.8%	52.5%
True Negative	7746	8130	8097	7598	8083	6994
True Positive	1754	1610	1680	1990	1697	1768
False Negative	1692	1836	1766	1448	1749	1678
False Positive	775	391	424	923	438	1527

they may have an equal probability of halting or graduating. Such a finding would be somewhat unexpected.

Limitations of traditional academic features in predicting student outcomes. In previous studies, some of the limitations of the machine-learning methods in predicting student academic outcomes were discussed in the context of cumulative versus semester grade counts. In this study, the nature of the academic input features and how they impact the classification of machine-learning models are examined. In the process, a key shortcoming related to single-semester deviations in academic performance is identified.

Table 2.5: Accuracy results for the machine learning models using semester GPAs.

Models	RF	SVC	LR	GNB	GB	KNN
Total Accuracy	79.6%	77.2%	73.3%	77.5%	76.3%	73.3%
Halt-Accuracy	42.4%	58.6%	65.0%	58.3%	58.5%	42.3%
Graduate-Accuracy	87.5%	81.1%	75.0%	81.6%	80.1%	79.9%
F1 Score	42.1%	47.4%	46.0%	47.6%	46.4%	35.7%
True Negative	7080	6566	6073	6606	6484	6464
True Positive	728	1007	1117	1001	1005	726
False Negative	990	711	601	717	713	992
False Positive	1012	1526	2019	1486	1608	1628

Many studies have used semester GPAs and cumulative GPAs as inputs for machine-learning models to predict student academic performance and outcomes [10, 65, 66]. For example, Gershenfeld et. al [67] applied a set of logistic regression models to predict if undergraduate students graduate within six years. Their results suggest that a low first-semester GPA is a significant factor in predicting if students will graduate within six years.

While using student GPA measures as inputs in educational data mining is common, utilizing GPA to analyze academic performance trajectories (i.e. times-series) may not be ideal. For example, it is common for a student's GPA to fluctuate on a semester-to-semester basis. Fluctuations in performance and semester GPA can result from both internal factors associated with a course (e.g., grading policies, course difficulty) and external factors unrelated to a course (e.g., illness the week of an exam). And while machine-learning techniques are able to account for noisy input features across samples, it is not immediately clear that machine-learning techniques are able to mediate the impacts of noise within a sample; thus,, normal fluctuations in GPA between semesters may cause incorrect classifications.

In order to show the impact of grade fluctuations on model accuracy, the same six different machine learning techniques are used to train a set of graduate/halt prediction models based on semester GPA (as opposed to grade counts). The models will be trained to predict graduating or halting based on the first four semester GPAs of students. A summary of the accuracy measures of the models are provided in Table 2.5. Interestingly, the total accuracy of the four-semester GPA models decreased when compared to the grade-count models based on two semesters. The decrease in total accuracy is directly tied to both the decreases in graduation-accuracy and halt-accuracy. For example, based on Figure 2.5, the halt-accuracy for models trained based on grade-count during the first four semesters is 66.7%, 61.5%, 63.8%, 59.3%, 63.9%, and 65.3% for the RF, SVC, LR, GNB, GB, and KNN, respectively, which are higher than the corresponding halt-accuracies provided in Table 2.5. Broadly speaking, a greater number of input features generally results in greater model accuracy, as such reducing the number of input features from 6-to-1 is expected to come with a decrease in model accuracy (particularly since GPA is a measure derived from semester grade counts). However, the change in accuracy is more meaningful, as the previous sensitivity analysis indicated that specific grade-counts have greater importance when making predictions (i.e. the number of F grades).

For each of the six models, GPA data for two different synthetic students are evaluated. The first synthetic student maintains a consistently high GPA over four semesters. The second synthetic student performs well for three of the four semesters; in their second semester, they earn a 1.3 GPA. Table 2.6 reports the classification for each of the machine-learning models, along with the corresponding prediction probability (as provided by the predict_proba function in Sklearn). The prediction probability is an estimate that a particular sample falls into the underlying classes. In the table, the halt and graduate probabilities for the KNN model is not available since this method does not classify data points using probabilistic approaches. The study of the halt probabilities indicates that the GNB model is the most sensitive to a singular fluctuation in GPA when compared to the other models since the difference in halt probability between students 1 and 2 for this model (76.9%-7.6%) is greater than the same difference of the other models. In contrast, logistic regression is the least sensitive model when changing to semester-to-semester GPAs when compared to other models. Regardless, each of the six models predicts that student 2 will halt their academic career.

Table 2.6: Impact of students' GPA change on model prediction

Model	Student Num.	GPA	Probability (Halt)	Model prediction
RF	1	3.55, **3.7**, 3.6, 3.4	18.0%	Graduate
	2	3.55, **1.3**, 3.6, 3.4	55.0%	Halt
SVC	1	3.55, **3.7**, 3.6, 3.4	27.1%	Graduate
	2	3.55, **1.3**, 3.6, 3.4	72.9%	Halt
LR	1	3.55, **3.7**, 3.6, 3.4	25.9%	Graduate
	2	3.55, **1.3**, 3.6, 3.4	54.1%	Halt
GNB	1	3.55, **3.7**, 3.6, 3.4	7.6%	Graduate
	2	3.55, **1.3**, 3.6, 3.4	76.9%	Halt
GB	1	3.55, **3.7**, 3.6, 3.4	24.7%	Graduate
	2	3.55, **1.3**, 3.6, 3.4	61.1%	Halt
KNN	1	3.55, **3.7**, 3.6, 3.4	–	Graduate
	2	3.55, **1.3**, 3.6, 3.4	–	Graduate

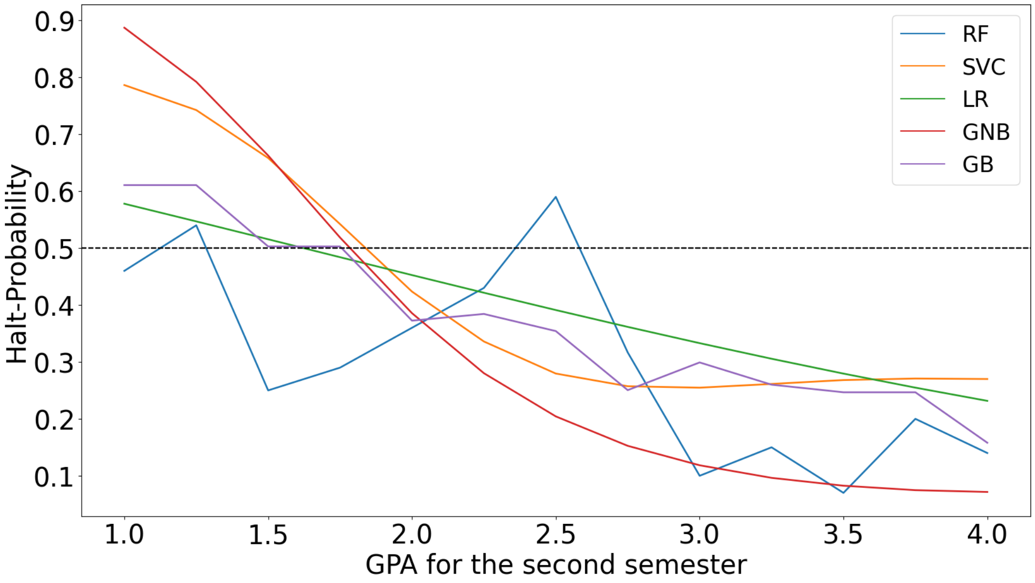


Figure 2.8: Impact of students' GPA change on model prediction

In the previous table, two different values for second semester GPA were considered to investigate the impact of perturbations on model predictions. Figure 2.8 extends this analysis by continuously varying the perturbation in the second semester GPA over a wider range. In other words, the input to the machine-learning models is the following vector: [3.55, X, 3.6, 3.4], where X is considered as a variable and is replaced by a value between 1 and 4. As illustrated in Figure 2.8, a decrease in the second semester GPA continuously increases the probability of halting for all models except Random Forest. The GNB, LR, and SVC models have smooth response curves since these models compute halt-probability using continuously differentiable functions. In contrast, RF has discrete breaks that appear random; the jagged response line for the Random Forest model is likely associated with the underlying non-linear random partitioning technique it utilizes. The black horizontal line at 0.5 indicates the cut-off threshold determining if the model predicts if a student will halt or graduate. If the obtained halt probability is less than the threshold (0.5) the prediction is graduate, otherwise, the models' prediction for the output is halt. As it is shown in the figure, for a GPA less than 1.5, all models except Random Forest predict the student halts enrollment.

As a point of comparison, student records are mined to find FTIC students who maintained a minimum predefined GPA for three of their first four semesters, but had a single-semester drop in GPA below 2.0 (the drop in GPA is permitted to occur in any of their first four semesters). A summary of the records, including counts and halt percentages, is provided in Table 2.7. As the table indicates, of students that maintained a GPA over 3.5 for three of the four semesters, while having a single semester drop in GPA, only 14.8% halted. As the minimum GPA over the three semesters is lowered, the halting rate increases. For students that had a minimum GPA between 3.0 and 3.25, with one semester below 2.0, 33.5% would eventually halt. Cumulatively, over all students that had a minimum GPA above 3.0, 30.6% percent would halt. The 14.8% halt rate for students with a minimum GPA above 3.5 is much smaller than the class probabilities indicated by the machine-learning models. While not a direct comparison, the stark distinction suggests that

Table 2.7: Graduation rates for students with a single-semester drop in GPA.

Min GPA	Halt	Graduate	Total	Halt Rate	Cumulative Halt Rate
3.5	9	52	61	14.8%	14.8%
3.25	33	92	125	37.5%	26.4%
3.00	93	211	304	33.5%	30.6%
2.75	187	368	555	37.5%	33.7%
2.50	341	601	942	39.8%	36.2%
2.25	515	816	1331	44.73%	38.7%

traditional machine-learning models are unable to address such cases of grade fluctuations.

2.4 Conclusion

In this chapter, different machine learning techniques were applied to UCF student record data to discuss shortcomings of previous studies in predicting student final outcomes from two different aspects: (1) the machine learning methods, and (2) the features that these models are trained by. In the first study, different prediction models were developed to predict if a student will halt enrollment or graduate using the cumulative number of A's, B's, C's, D's, F's, and W's. Based on the inputs and output of the model, these models assume that halting is in response to academic performance. This assumption is wrong since in many cases the decision to halt could be due to other factors such as financial issues, lack of support, etc. Furthermore, the obtained results indicated that these models are sensitive to the time when predictions are made. Based on the results, the more semesters fed into the training set, the higher the halt-accuracy. While early predictions may have low prediction accuracy, a late prediction may not provide enough time for at-risk students to improve their performance. Therefore, finding an optimal time for making the prediction is another limitation for these models. In the second study, similar machine learning models were developed to predict students' graduation/halt status, but instead of using cumulative grade counts over the semesters, the semester grade-count was used as the input for the model. The lack of significant improvements in the results implies that these models are not able to capture the patterns of students' academic-performance (e.g., decreasing or increasing) over their academic careers. Finally, in the third study, the six machine learning models used students' semester GPA as inputs to predict if students will graduate or halt enrollment. The first goal of this study was to investigate the performance of GPA as input for the machine learning models in predicting students' final outcomes and compare it with the grade-count feature used in the previous studies. The results indicated that the machine learning models trained by grade-count have higher accuracy (total, graduate, and halt accuracy) when compared to the models built by GPA. In other words, the grade-count variable is a better feature in predicting halt when compared to GPA. The second goal of this study was to analyze the impact of grade fluctuation on model accuracy. Results illustrated that using GPA as inputs for machine learning models is challenging, since a change in students' semester GPA that could be because of internal and external factors can impact model prediction significantly. In chapter 4 of this dissertation, a Hidden Markov Model is applied to propose a new academic feature in predicting students' performance to tackle this shortcoming.

CHAPTER 3
ACADEMIC DATA RECORDS

UCF is a public university located in Orlando, Florida. With almost 72,000 students from 50 states and 144 countries, UCF has one of the largest undergraduate student bodies in the United States [68]. Opened in 1968 as Florida Technological University, the university was later renamed to the University of Central Florida in 1978. In addition to the main campus, UCF has several other campus locations. These include: Cocoa, Daytona Beach, Ocala, Kissimmee, Palm Bay, and Leesburg. UCF is composed of 13 colleges that offer 231 different degree programs, including 103 bachelor's, 91 Master's, 34 doctoral, and 3 specialist programs [68].

The studies presented in this dissertation make use of anonymized undergraduate student records collected from the University of Central Florida (UCF) between the years 2008 to 2017; as such, some time is spent describing the available data and features unique to UCF's student body. Next, this chapter provides a description of common measures that are used to evaluate student academic performance (i.e., course grade, graduate/halt, etc), with a brief analysis correlating demographic features to final academic outcomes. After analyzing students' financial aid status and its correlation to their academic performance, the limitations of the introduced data set are discussed.

3.1 UCF Student Data Records

The data set used in this dissertation is collected and provided by the Institutional Knowledge Management (IKM) department at UCF. The data set includes records covering approximately 170,000 students over a ten-year period from 2008 to 2017. IKM's mission is to inform and guide UCF in operational and executive decisions by collecting, organizing, interpreting, and reporting institutional data and providing advanced analysis through scientific research [69]. IKM's core activities

Table 3.1: Variables and corresponding descriptions in the Admission Demographics table

Variables	Descriptions
Identifier	Unique identifier for student
Gender	Male, Female
Ethnicity	White, Black, Hispanic, Asian, Multi, Hawaii/Pac
Birth date	YYYY-MM-DD
First generation in College	Binary
Admission type	FTIC, Transfer
High school GPA	e.g. 3.5
Cip code[1]	e.g. 14.3501
State of Primary Residence	e.g. Florida

are divided into four categories: (1) providing UCF faculty, staff, and leadership with data and analysis with the aim of supporting students, personnel, and academic programs; (2) preparing reports about UCF facilities, budgeting, students, and expenditures; (3) providing insights on research, faculty, and students in order to support performance goals strategic and enrollment management; and (4) responding to surveys which support UCF's recognition and ranking in U.S. News and World Report, World University Rankings, and so on.

The student records data set provided by IKM contains a wide variety of information about students at UCF, including but not limited to: demographic information, admission information for enrolled students, bachelor degrees awarded, courses taken by students at UCF, and Free Application for Federal Student Aid (FAFSA) reported family income. All this information is organized into seven different MySQL data tables. These tables together with their corresponding data values are summarized below:

Admission Demographics: This table provides self-reported demographic and admission information for students who have enrolled at UCF. Details of variables for this table are summarized in Table 3.1. Key features include: gender, race/ethnicity, birth date, and admission type. In regards to race and ethnicity, these values are collapsed into a single field. As such, for Hispanic students, additional race information is not available.

Degrees Awarded: This table contains all information regarding bachelor's degrees awarded.

Table 3.2: Variables and corresponding descriptions in the Degrees Awarded table

Variables	Descriptions
Identifier	Unique identifier for student
Academic Career	Undergraduate
Degree	BA, BDES, BM, BS
Term Awarded	e.g. Fall 2009
Degree College	e.g. College of Science
STEM[2] Flag	Binary
Degree GPA	e.g. 3.5

This includes double majors and minors, which are denoted both by their Classification of Instructional Programs (CIP) code, their nominal degree name, and affiliated college. Data regarding master's and doctoral degrees is not included. The corresponding variables are shown in Table 3.2.

UCF Courses Taken: This table contains information about all courses taken by students for each semester they are enrolled at UCF; these courses include both undergraduate and graduate-level courses, some of which might correspond to requirements of 5-year BS/MS degrees. Table 3.3 lists the variables in the UCF Courses Taken table. Courses taken at other universities or community colleges are not included.

Table 3.3: Variables and corresponding descriptions in the UCF Courses Taken

Variables	Descriptions
Identifier	Unique identifier for student
Course number	e.g. MAC2147
Course letter grade	e.g. A and C
Course section type	Class lecture, Online
Course credits	e.g. 3
Enrollment term	e.g. Fall 2009
Cip code	e.g. 14.3501
Course campus location	e.g. Main campus

Term Enrollment: Students' term-to-term enrollment information, include their major,

Table 3.4: Variables and corresponding descriptions in the Term Enrollment table

Variables	Descriptions
Identifier	Unique identifier for student
Academic Career	Undergraduate
Academic Level	Freshman, Sophomore, Junior, Senior
Enrollment term	e.g. Fall 2009
Previous enrollment term	e.g. Summer 2009
Term GPA	e.g. 3.5
Cumulative term GPA	e.g. 3.5
Cip code	e.g. 14.3501
Pell Grant[3]	Dollar amount

semester GPA, academic level, and other variables are provided in Table 3.4. This table is partially based on summary data derived from course data (e.g. GPA, academic level).

Family Income: Students' family income by financial year (if provided via FAFSA) is reported in this table.

Table 3.5: Variables and corresponding descriptions in the Family Income Taken table

Variables	Descriptions
Identifier	Unique identifier for student
Financial year	YYYY
Family income	Dollar amount

Student Involvement: For all students, this table indicates whether each student has participated in a program listed in the Table 3.6.

Table 3.6: Variables and corresponding descriptions in the Student Involvement table

Variables	Descriptions
Identifier	Unique identifier for student
Participation term	e.g. Fall 2009
Recreation Wellness Center	Binary
Undergraduate Research	Binary
Experiential Learning	Binary
Peer Tutoring	Binary

Test Credits: While courses taken at other universities are not included in the student data records, this table includes detailed information regarding student test scores and any credit earned in each registered course. Course credit may be transferred or resulted from Advanced Placement exams.

Table 3.7: Variables and corresponding descriptions in the Test Credits table

Variables	Descriptions
Identifier	Unique identifier for student
Course title	e.g. Composition I
Course prefix	e.g. ENC
Course number	e.g. 1101
Earned Credit Flag	Binary
Test score	e.g. 95

3.2 UCF study body demographics

The University of Central Florida is a relatively diverse university, drawing from various student populations covering a wide range of socioeconomic statuses, admission types, genders, race/ethnicities, etc. Table 3.8 provides students' self-reported gender and ethnicity information along with their admission type (FTIC and transfer). As indicated by the table, the majority of the student population at UCF are transfer students (59%). This indicates UCF is a common destination for students who wish to transfer from community colleges to universities.

Like most public institutions, women constitute the majority; 56% of students during the 10 past years have been female, while the rest identify as male (44%). And while most students (55%) self-report as white, the percentages of Hispanic and African-American students are quite large at (24%) and (11%), respectively. The *other* group in the table covers American-Indian, Asian, Native Hawaiian, and Multi-racial students that constitute (24%) of UCF. When compared to national statistics, the average percentage of Hispanic students at American universities between the years of 2010 and 2016 is reported to be 16% [70]. UCF, with 24% of its student body self-reported as Hispanic, has a significantly higher proportion when compared to the national average. Accordingly, UCF has recently been designated to be a Hispanic serving institution by the US Department of Education [71]. While other universities such as the University of California, Los Angeles and University of California, Berkeley have a large population of Hispanic students (19.7% and 14.1%), these universities are not considered as Hispanic serving institutions. Another factor used to determine if a university is considered to be a Hispanic serving institution is the fraction of Hispanic students who receive financial aid. On this basis, the ratio of Hispanic students with low family income and first-generation students at UCF allows the university to be designated to be a Hispanic serving institution by the US Department of Education. Such key characteristics indicate that UCF has significantly different student demographics when compared to other universities in the United States, even when compared to local higher education institutions. For example, Rollins College, a private liberal arts college in Winter Park, FL, has a lower fraction of Hispanic students (19%) and a higher median family income when compared to UCF ($123,400 vs $94,500); on this basis, Rollins College is not considered to be a Hispanic serving institution. Given UCF's unique study body demographics, some findings reported in this dissertation may not apply to other universities as demographic factors may influence student academic performances.

Table 3.8: Students' demographic distribution at UCF over 10 years

Students demographic	Percentage
Female	56%
Male	44%
First-Time-in-College	41%
Transfer	59%
White	55%
Hispanic	24%
African-Am.	11%
Other[4]	10%

3.3 Academic Performance and Outcomes

As mentioned earlier in this chapter, the data set includes student course performance and final educational outcomes. Given that this dissertation aims to analyze students' academic performance and outcomes, this section provides details to provide context regarding academic performance at UCF. First, and foremost, UCF has an estimated graduation rate of 75%; this value does not distinguish between admission type (FTIC and transfer), sex (male and female), and race/ethnicity. Additionally, it is worth noting that all students who have completed their program are considered as a graduate, regardless of the time taken to complete their degree program. In contrast, a student is considered to halt if he/she does not register for three consecutive semesters at UCF. Based on federal regulations, American universities typically use the six-year graduation rate to report statistics regarding their students' final educational outcomes (graduate/halt). A program's graduation rate is defined as the percentage of first-time-in-college (FTIC) students who complete the program within 150% of the standard enrollment time for their degree [72]. For a four-year degree program, students who earn their degrees within six years are considered graduates, hence the name six-year graduation rate. Using the IKM supplied data set, the six-year graduation rate at UCF for first-time-in-college (FTIC) students starting in 2008 is 71.2%, with 28.8% of students graduating in more than six years or halting enrollment. Given that the majority of students at UCF are transfer students, and a large portion is also part-time [73], the six-year graduation rate is not

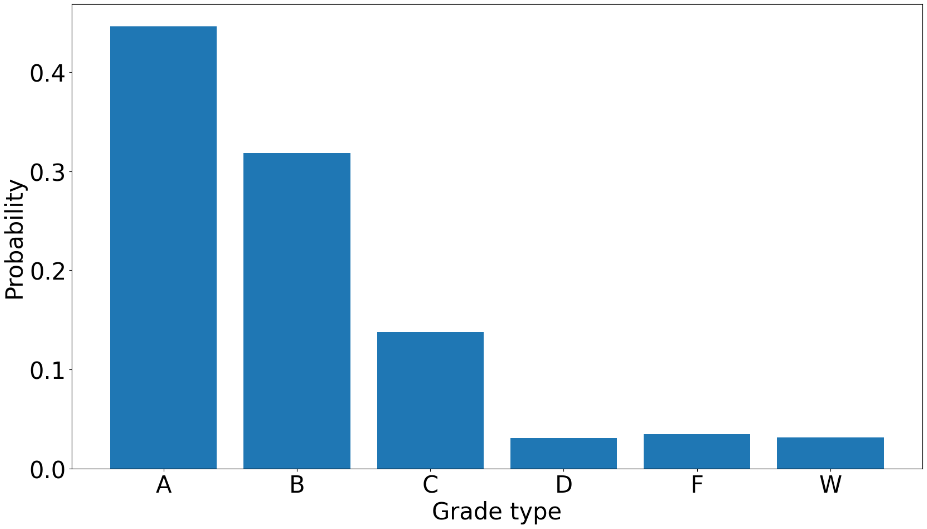


Figure 3.1: Distribution over students' GPA at UCF

necessarily an accurate or representative reflection of the academic outcomes at UCF.

These academic data records include course grades for every student (see Table 3.3). Figure 3.1 provides the distribution of grades over every student at UCF. Based on the figure, the grade distribution is positively skewed, with the probability of getting A, B, C, D, F, and W grades at UCF being 44.7%, 31.8%, 13.7%, 3.1%, 3.5%, and 3.2% respectively. Figure 3.2 illustrates the same kind of distribution but separated by students that graduate and students that halt. As the figure indicates, the student grade distribution is strongly tied to final academic success outcomes (graduate/halt). Based on the figure, the probability of getting A or B in any given course for students who complete their program at UCF is, on average, greater than the same probabilities for students who halt enrollment. Also, students who halt enrollment at UCF earn C, D, and F grades with higher probabilities while also withdrawing more often when compared to students who graduate. Keep in mind that a W grade is not an academic grade and is not considered when calculating GPA; this is particularly relevant because students often withdraw for non-academic reasons. For example, students without sufficient financial aid or funding may be forced to work to earn money and support themselves. In a study by Lijuan and Rey [74] the researchers showed that there is a significant relationship between student family income and withdrawal rates. Another common non-academic reason for withdrawing includes medical issues. Based on Figure 3.2, while W grades are not directly tied to academic performance, students who halt enrollment tend to have a higher probability of registering a W grade when compared to students who graduate.

A hypothesis chi-squared test was conducted to assess if the difference between grade distributions for students who graduate (N=73682) and students who halt (N=33189) enrollment is statistically significant. The null (H_o) and alternative (H_A) hypotheses are as follows:

H_o: Both graduated and halted students have the same grade distribution.

H_A: Graduated and halted students have different grade distributions.

The null hypothesis in this test states that the grade distribution for both halting and graduating students is the same. On the other hand, the alternative hypothesis is that both groups of students have different grade distributions. The obtained p-value (1×10^{-8}) for the test rejects the null hypothesis test and indicates that the difference is statistically significant. Therefore, halting and grade distribution have a strong correlation at UCF. There is research that support this conclusion at other universities as well [75, 76, 77]. For example, Bail et al. [75] show that students' semester GPA is a significant factor in predicting students' graduation rates. In another paper, Bolden et. al [78] conducted a study based on a data set collected from three universities, including Case Western Reserve University, Transylvania University, and St. Martins's University, to investigate the factors that impact students' academic success. Their results indicated that DFW grades are an important predictor of student retention, persistence, and success. Furthermore, they indicated that students who had a lower SAT/ACT score and lower high school GPAs are more likely to get DFW grades in colleges.

Since a part of this dissertation focuses on students' academic performance trajectories, it is useful to investigate how students' grade distributions change over time. Figure 3.3 shows the grade distribution for students at various academic levels, including Freshman, Sophomore, Junior,

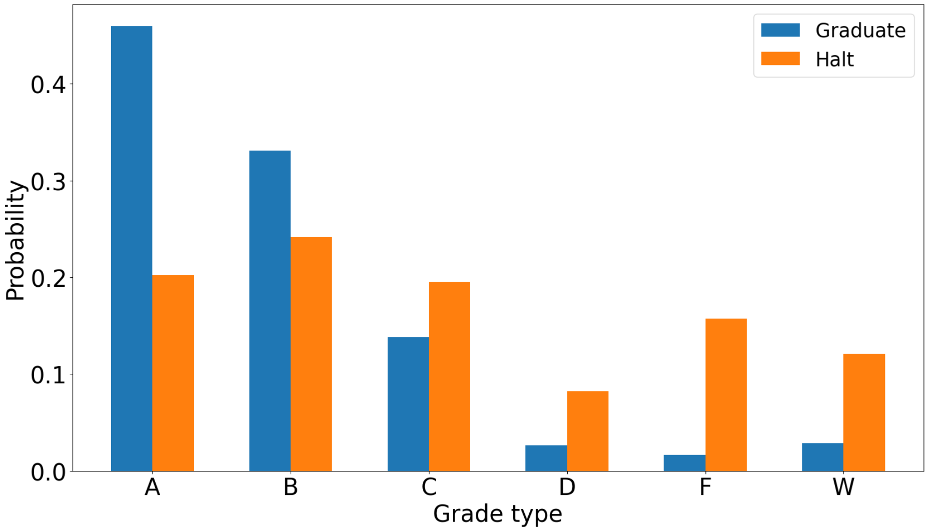


Figure 3.2: Grade distribution over students at UCF for graduated and halted students

and Senior. Similar to other universities, these academic levels are defined according to the number of course credits that students earn over their academic careers. At UCF, these academic levels are defined based on the indicated criteria on the passed course credits in Table 3.9. Based on the figure, while seniors have the highest chance (45%) to earn A grades in any given semester, freshman students are the least likely (36%) to get A grades.

A Chi-square post-hoc hypothesis test was conducted to see if the grade distribution differs between students with different academic levels. The null hypothesis test is that the grade distribution among all academic levels is the same. The alternative hypothesis states that grade distribution for at least one of the academic levels differs from the others. The obtained p-value of all tests between possible pairs of academic levels is 0, indicating that students with different academic levels have different grade distributions. For example, seniors (N=344394) statistically are more likely to get A grades when compared to freshman (N=76316). On the other hand, freshman are more likely to earn Bs (34% and 31% for freshman and senior students, respectively), Cs (17% and 13% for freshman and senior students,), and Fs (6% and 2%) grades when compared to seniors. These findings are supported by research conducted by Addus et. al [79], where they showed that the number of students with low GPAs decreases from freshman year to senior year. Two commonly

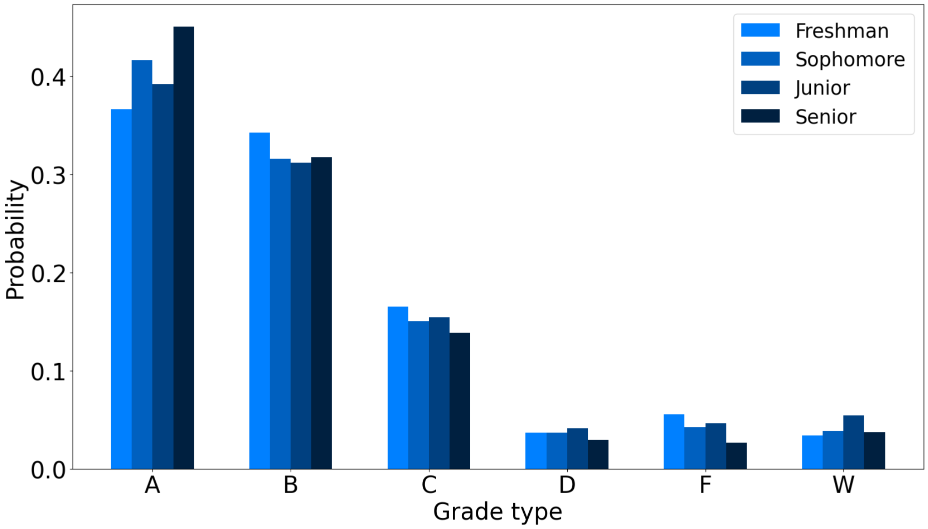


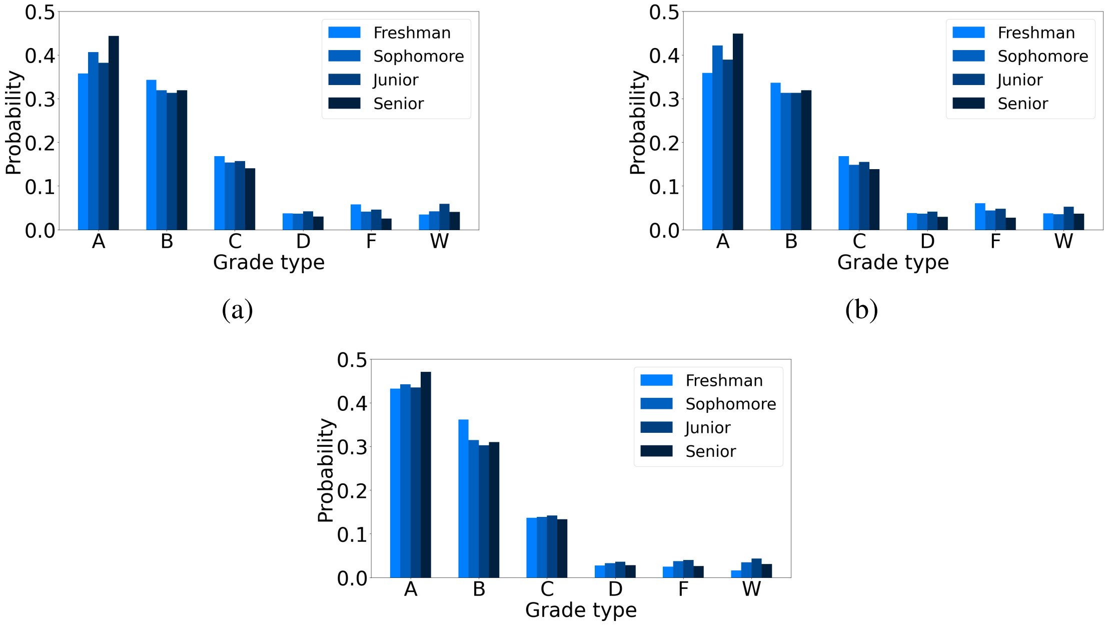
Figure 3.3: Grade distribution over students' academic levels at UCF

Table 3.9: Definition for different students' academic levels at UCF

Academic level	Number of credits
Freshman	0-29
Sophomore	30-59
Junior	60-89
Senior	+ 90

proposed reasons for these differences in GPA grade distributions of freshman and senior students is that seniors have adapted to their academic environment and also take courses more aligned with their academic interests. Alternatively, poor-performing students may have been filtered out and already halted.

Figure 3.4 compares grade distributions for students at different academic levels during the Fall (a), Spring (b), and Summer (c) semesters. Based on the figure, students with different academic levels have similar grade distributions, except for W grades, during the Fall and Spring semesters. As an example, during the Fall and Spring semesters, seniors are most likely to get A grades, followed by sophomores, juniors, and freshmen, respectively. These similarities are seen for B, C, D, and F grades. However, with regard to W grades during Fall semesters, sophomores are more likely to withdraw from courses when compared to freshmen students; the opposite pattern is observed during Spring semesters for these two groups of students. Furthermore, by comparing



(a)

(b)

(c)

Figure 3.4: Grade distribution over students' academic levels in the Fall (a), Spring (b), and Summer (c) semesters

the grade distributions between all semesters, it is seen that students at different academic levels have different patterns in getting C, D, and F grades during Summer semesters when compared to Fall and Spring semesters.

3.4 Financial aid and academic outcomes

Like most universities, the annual family income of UCF students varies widely, as shown in Figure 3.5. Many studies use student family income as a demographic factor to model and better understand academic outcomes [80, 81, 82]. These studies have shown that students with lower family incomes are more likely to halt enrollment. The corresponding drop in graduation rates can be traced to a variety of reasons. For example, about 25% of first-generation students are low-income, and around 90% of them halt enrollment at colleges. Based on research conducted by Accredited Schools Online [83], parents of first-generation students are less likely to be sympathetic to the issues and stresses that their children have in college. These students usually have less

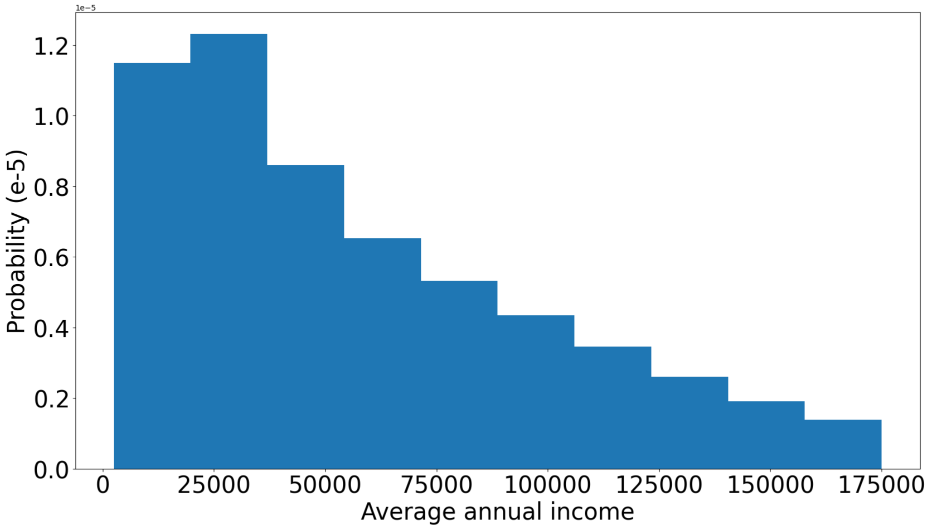


Figure 3.5: 5th-95th distribution of annual family income averaged over each students academic career

support from their parents for selecting the right college and major, applying for financial aid, and so on. Also, students who come from a family with low income are less likely to have professional and personal mentors who help them through challenges that they may have during their academic careers.

Figure 3.6 illustrates the connection between family income and graduation/halting for students at UCF. As the figure suggests, students who halt enrollment have a lower mean annual family income (\$52798) when compared to students who graduate (\$59712). A t-test hypothesis test is conducted to see if the mean of annual family income between these two groups is statistically significant. The null hypothesis in the test says that students who halt enrollment have the same average annual family income as the students who graduate. However, the alternative hypothesis states that the average annual family income differs between students who halt and students who graduate. The obtained p-value (4.8×10^{-80}) indicates that students who graduate (N=77342) statistically have a higher family income than students who halt enrollment (N=24136). Therefore, by assigning financial aid to students who come from families with low incomes, university policymakers may improve graduation rates in universities for cases in which the reasons for halting

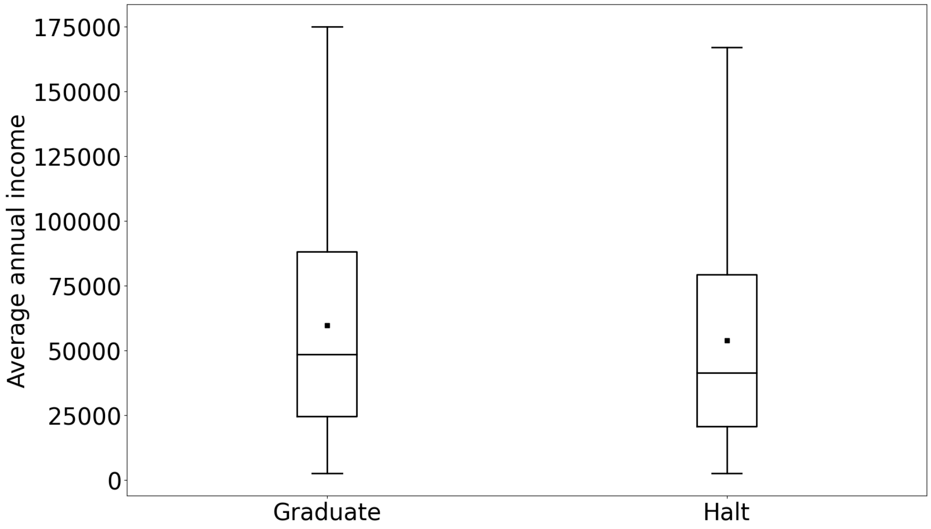


Figure 3.6: Average annual income for graduated and halted students

are due to economic barriers.

3.5 Limitations

The IKM data set, while comprehensive, still has several limitations that constrained the width and depth of the studies presented in this dissertation. Explaining such limitations is crucial as it guides towards more practical data collection and organization, leading to more informed analysis and thus producing better insights. For example, the data set under consideration did not include academic and demographic information on students who halted enrollment and left school without a degree. Such information, if present, could improve the analysis of students' performance, their academic patterns (including GPA, enrollment, etc.), and the corresponding consequences on their outcomes. Moreover, for students who transferred to UCF from community colleges or other universities, records of their academic performance, such as transcripts and GPAs corresponding to their previous institutions, were not included. This data could reveal invaluable information on students' academic backgrounds and enrollment behavior upon their admission to UCF, and therefore, enable more accurate, timely, and informed analysis. This would help universities support students and prevent potential halts.

The data set also lacks critical information in terms of students' financial and demographic status. Student family income and financial status data are provided via FAFSA, and therefore, if a student has not applied for this application, the student's yearly family income is reported as zero. The lack of student Expected Family Contribution (EFC) information is another shortcoming of the existing data set. EFC is an indicator of students' eligibility for receiving federal financial aid. Having information on both students' EFC and their family income provided by FAFSA can help in studying the impact of students' financial status on their academic performance. Other issues correspond to missing or misleading information in the data-set as related to self-reported demographic information, such as ethnicity/race and gender. For example, there is no clear distinction between race and ethnicity; while Hispanic is an ethnicity, it is recorded as a race in the data set. Also, for gender, a non-binary identity is not included in the self-reported application, which could affect the research findings on the impact of gender on students' academic performance.

3.6 Conclusion

In this section, the data set that used in this dissertation was introduced. These students' records were collected by Institutional Knowledge Management during the years 2008 to 2017. As explained, the data set contains useful information regarding student demographics and academic features. Also, the chapters provided some statistics about UCF students. The findings from students' academic analysis indicated that while 75% of students graduate from UCF, the remaining 25% halt enrollment. Furthermore, it was shown that grade distribution for these two categories of students (graduated and halted) are statistically different. Also, we saw that students with different academic levels have different grade distributions when compared to each other. Finally, the conducted analysis in this section illustrated that UCF is a unique university in two aspects: first, most students enroll at UCF as transfer students, and second, UCF has a large population of Hispanic students. This uniqueness may put limitations on how transferable the results of this dissertation are, and as such, they might not apply to other universities. As an example, it was shown that the student's family income, which is a known key demographic factor, is a significant feature correlated to students' academic performance, which can be traced to be a causal factor in some cases.

CHAPTER 4
APPLICATION OF HIDDEN MARKOV MODELS TO ANALYSIS STUDENTS' ACADEMIC-PERFORMANCE TRAJECTORIES

Based on the National Student Clearinghouse Research Center, the six-year graduation rate for students who initiated their college education in 2012 is only 58%, with 42% of students either halting enrollment or taking longer than six years to graduate [1]. Halting college is broadly understood to impose irreversible mental, financial, and time losses to students [84].

To date, numerous reasons have been identified to explain why students choose to halt their education; they include financial problems, lack of interest in studies, lack of study progression, and inadequate information and guidance [85]. In many cases, the reasoning is related to academic performance; however, the decision to halt one's education is not a straight line between perceived poor grades and halting. As D. Shippee notes, poor grades may lead to the feeling of depression [86], or lead students to shift career goals [87]. Thus, low academic performance can instead be viewed as an underlying factor, predictor, or leading indicator of halting. Along these lines, many studies have recognized GPA as a fair predictor or leading indicator of students persisting or halting their academic career [21, 88, 20]. Zahedi et al. [21] investigated the relationship between graduation rate and numerous variables, including gender, race, transfer status, major, number of terms registered, and cumulative GPA. Their findings suggest that cumulative GPA is one of the most powerful measures for predicting graduation. In a similar study, Pappas et al. [88] showed that students' cumulative GPA is a significant factor in identifying students at risk of dropping out among computer science students. Although cumulative GPA is demonstrated to be a valuable indicator and predictor of a students' academic outcomes, the measure is a summary statistic. As such, cumulative GPA does not convey how a student's GPA evolved over time. Using cumulative GPA as a predictive measure essentially ignores underlying patterns that could convey meaningful status changes.

To better understand, model, and characterize students at-risk of halting, I believe it is necessary to mathematically model academic-performance trajectories. The desire to model academic performance trajectories is driven by the following research questions:

Research question: Does a decrease in academic performance proportionally increase the risk of halting?

Research question: If a student improves their academic performance does their graduation rate match those students that always performed well?

This study is aimed at analyzing students' academic-performance trajectories to see if these trajectories assist in answering the research questions above. To address the research questions, I start by designing a Hidden Markov Model (HMM) to convert complex student academic records to a compact discrete representation I denote as an *academic-performance level*. The HMM is tuned using anonymized student records of 36261 first-time-in-college (FTIC) students. After applying the HMM, the resulting compact representation of the academic-performance trajectories is grouped into meaningful clusters for analysis. Groups are compared and contrasted to understand how academic-performance trajectories correlate to halting. Accordingly, the primary contributions of this chapter are: **(1)** a technique for generating a compact representation of a student's academic-performance trajectory and **(2)** an improved understanding of the relationship between academic-performance trajectories and final academic outcomes.

4.1 Problem Statement

This chapter considers the problem of mathematically modeling the trajectory dynamics of a student's academic performance. The goal of addressing this problem is to identify a compact trajectory representation of academic performance that enables the classification and clustering of undergraduate students into meaningful groups. These groups will then be compared and contrasted to explore the relationship between academic performance trajectories and final educational out-

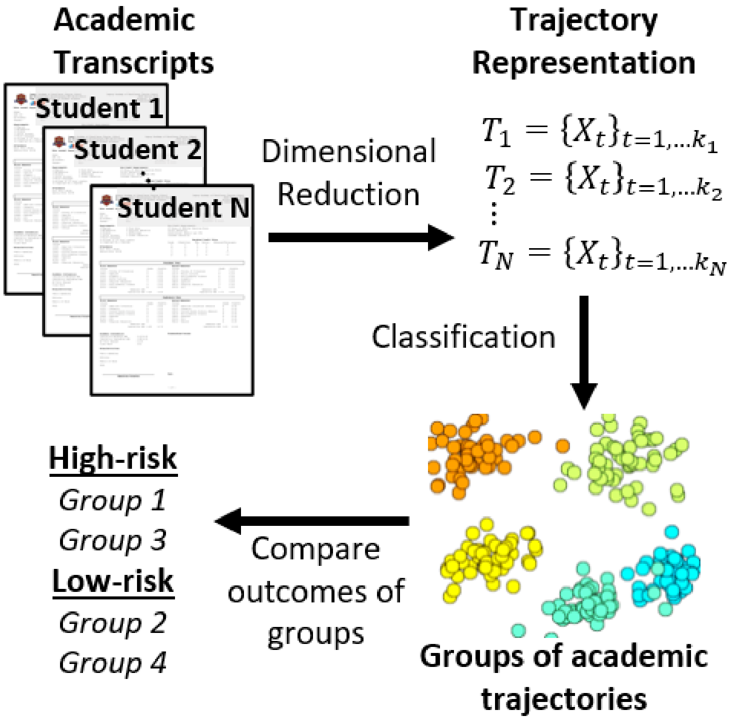

Academic Transcripts
Student 1
Student 2
Student N
Dimensional Reduction
Trajectory Representation
$T_1 = \{X_t\}_{t=1,\dots k_1}$
$T_2 = \{X_t\}_{t=1,\dots k_2}$
$T_N = \{X_t\}_{t=1,\dots k_N}$
Classification
Groups of academic trajectories
Compare outcomes of groups
High-risk
Group 1
Group 3
Low-risk
Group 2
Group 4

Figure 4.1: Representation of overall problem and solution methodology

comes, specifically halting. A pictorial representation of the overall problem and methodology is provided in Figure 4.1.

Here, I start with the notional definition of an academic-performance trajectory which is a temporal sequence of states indicating a student's academic performance. A frequently encountered category of academic performance trajectories is student transcripts; each semester corresponds to a time-step t in a trajectory $\hat{T}_i$ while courses are taken, and grades received each semester correspond to trajectory states, $\hat{X}_t$. The challenge of working with such data is the dimensionality of the representation – specifically all the combinations of courses and grades in $\hat{X}_t$.

The goal of this chapter is to identify a mapping $\hat{T}_i \to T_i$ that translates an academic performance trajectory into a lower-dimensional representation T_i, where the state X_t at each semester corresponds to an academic-performance level. As an example, consider a case in which each student's transcript is first converted to a sequence of grade counts given by the number of A's, B's, C's, D'/F's (combined), W's they received in a semester. One such tuple, (1,0,1,1,2), would indicate that in the corresponding semester, a student received the following marks: one A, no B's, one C, one D or F, and two withdrawals. Over an academic career, a semester-to-semester sequence might look like the following:

$$(1,0,1,1,2),(2,1,1,1,0),(0,3,1,0,0),...,(3,1,0,0,0).$$

A mapping of the full trajectory to a compact trajectory representation might be $L_3, L_2, L_2, ..., L_1$, where each L_i in the time-ordered sequence corresponds to a specified academic-performance level (e.g. L_1 is low, L_2 is middle, and L_3 is high).

Unlike prior studies, I explicitly assume the grades a student earns exist within a random process whereby students' course grades are not deterministic. As an analogy, in an imaginary multiverse, a student could take the same course one-hundred times and earn different scores each time. Seemingly random processes or disturbances that might affect a student's scores, and ultimately final grade, can be due to external factors (e.g., illness the week of an exam) or internal factors (e.g., grading policies) related to the course. Accordingly, instead of each academic-performance level, L_i, being a straightforward mapping of the semester GPA, random processes and disturbances that create a noisy GPA signal must be filtered to estimate the academic-performance level.

4.2 Methodology

In order to map full academic trajectories to a more compact representation, I develop and apply a Hidden Markov Model (HMM). Here a brief modeling overview of the HMM is provided; a more detailed description of the underlying models and algorithms s provided in chapter 5. Application of HMMs to model students' academic behaviors and performance has been documented in several studies [51, 89, 90, 91, 92]. For example, in work by Kaser et al. [93], the authors apply an HMM to examine the impact of students' exploration strategies on learning. Their findings suggest that exploration strategy is important to learning outcomes. Using a similar framework to these studies, I develop and apply an HMM to model and understand the impact of academic-performance trajectories and final educational outcomes.

In this section, I provide an overview of the Hidden Markov Model used in analyzing students' *academic-performance trajectories*.

4.2.1 Representing Academic Performance using a Hidden Markov Model

As depicted in Figure 4.2, an HMM represents the dynamics of a system as it moves between operating states or modes (e.g., Modes 1, 2, and 3 in the figure). When operating within a state or mode, the system generates a mode-related output O_i at each time-step. For the problem under consideration here, the modes correspond to the academic-performance level of a student, and the observations refer to the student grade counts in any given semester. Because of the randomness assumption stated in section 4.1, the grade counts observed might not map directly to the expected academic-performance level of a student, and as such, I say the mode or academic-performance level of a student is not directly observable from their grades. Instead, through a series of grade observations, I can estimate or infer the most likely academic-performance level of the student in any semester given the likelihood of such a sequence of mode transitions (i.e., trajectory).

Each Hidden Markov Model is initialized by a set of modes $S = \{s_1, s_2, \ldots, s_N\}$. Unlike any other types of Markov models in which the modes are observable, in HMMs, modes are hidden; that is, it is unknown at which mode the system dwells at any particular moment. After developing an HMM, one application of an HMM is then to estimate a variable-length sequence of the hidden modes estimates given a series of observations, $O = o_1, o_2, \ldots, o_T$, that are generated from the modes. Within this framework, each HMM is defined by the following parameters: (1) $A = [a_{i,j}] \in \mathcal{R}^{NxN}$, a transition matrix containing the single-step probabilities by which a system moves between modes; (2) $B = b_i(o_t)$, an emission matrix describing the probability of generating each observation o_t when the system is in mode s_i; and (3) π, the initial probability distribution for the system at its start.

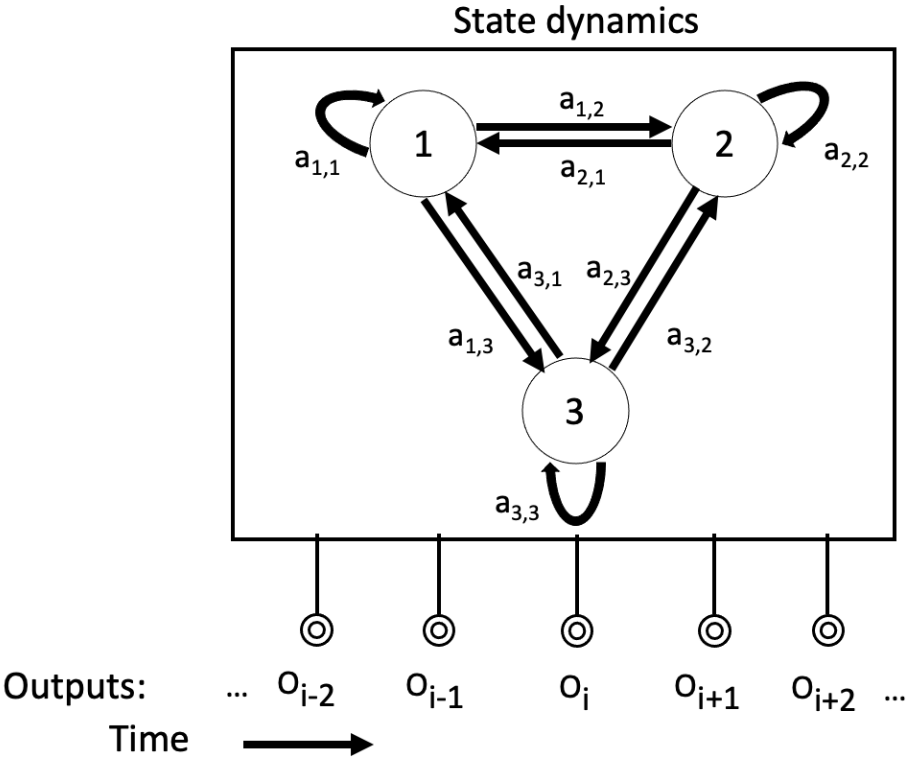


Figure 4.2: Representation of a simple Hidden Markov Model

The first step to build an HMM, known as the LEARNING STEP, is to estimate the optimal values for the parameters $\lambda = (A, B, \pi)$. The Baum-Welch algorithm, an expectation-maximization algorithm, is used to estimate the optimal parameter-set λ^*. This algorithm takes initial guesses for each of the parameters and iteratively improves the estimations of the parameters by computing the likelihood of any sequence of observations given λ. After obtaining the optimal λ^*, the next step is to predict the sequence of hidden modes for a given observation sequence. This step, known as DECODING STEP uses the Viterbi algorithm, which takes observation sequences and λ^* as input and returns estimated hidden modes as the output.

In our case, the first challenge with utilizing an HMM is the dimensionality of the observable outputs, i.e., grade counts. As described in section 4.1, each semester, I represent a student's academic performance according to a tuple with five elements, in which the first, second, and third elements are integers indicating the number of courses with A, B, and C grades. The fourth and fifth elements are variables indicating the number of D or F grades, and how many courses the student has withdrawn from. Under the assumption that the grade counts earned each semester are dependent (e.g., the number of A's earned is correlated to the number of B's, C's, etc.), it is reasonable to consider each unique grade-tuple to be a possible observation. Within the data-set under consideration, over 687 unique grade combinations in a semester have been observed. The most common grade combination was (4,0,0,0,0), which occurred 5.8% of the time. Meanwhile, there are over 20 grade combinations that occur only once, for example (1,1,1,0,4).

Given a large number of grade combinations, the wide distribution spread, and the relative sparseness in which many cases occur, it is necessary to perform a partial reduction in the dimensionality of the output space. Instead of each output being a tuple of unconstrained counts, elements of the grade-tuple are capped. The grade-tuple is adjusted to contain the number of A's, B's, and C's up to a maximum value of 3. Furthermore, instead of storing the number of D/F and W grades, these values are converted to binary variables. In this new representation, the original grade-tuple $(5, 0, 2, 0, 3)$ for a student who had registered for 10 classes would be converted to $(3, 0, 2, 0, 1)$. Thus, because students do not typically register for more than 5 classes, and because there is a strong correlation between grades, the impact of clipping is believed to be limited. After applying the new reduction, the total number of unique observations is reduced to 216 grade-tuples.

Within the HMM framework, the next step is to specify a predetermined number of modes for the model; again, each mode corresponds to an academic-performance level. The inclusion of each additional state increases the required number of parameters used to define the model (i.e., size of λ^*); it is critical to balance over-fitting (too many parameters) and model coarseness (too few parameters). For a model with N and M outputs, the total number of parameters within an HMM to be tuned is $N(N + M - 2) + N$; the tuned parameters correspond to a probability transition matrix, the emission probability for each grade tuple when operating in a state, and the initial state distribution. Through a combination of managing the number of modes and the number of grade-tuples represented, I can ensure there is sufficient data to tune the parameters.

As a rough initialization, students are classified based on their last semester's cumulative GPAs

Table 4.1: Cumulative GPA for the four academic-performance levels obtained by the hierarchical clustering method

Level number	Cumulative GPA
1	CGPA≤2.5
2	2.5<CGPA≤3
3	3.0<CGPA≤3.5
4	3.5<CGPA≤4.0

(CGPA) into eight academic-performance levels. The first level corresponds to students with a CGPA less than 2.0; level 2 includes students with a CGPA between 2.0 and 2.25, level 3 includes students with a CGPA between 2.25 and 2.5, and so on for higher levels. Next, for each academic-performance level, the distribution over grades including A, B, C, DF, and W was computed. Since the grade distributions observed over adjacent academic-performance levels were quite close, a round of aggregation over academic-performance levels was initiated using their pairwise distance. Based on [94], the distance between two Multinomial distributions, Δ, is computed with the following equation:

$$4\sin^2(\Delta/2) = (\sqrt{p_A} - \sqrt{p'_A}) + ... + (\sqrt{p_W} - \sqrt{p'_W})$$

In the above equation, p_A through p_W corresponds to the probability of grade A through grade W in a given academic-performance level. Given the pairwise distances between academic-performance levels and using the hierarchical clustering method, the eight different academic-performance levels were further grouped into four main levels. The resulting four academic-performance levels are considered as the hidden states of the proposed HMM. The cumulative GPAs associated with these states are shown in Table 4.1.

Based on an HMM model with 4 academic-performance levels and 216 possible grade-tuples, it is possible to provide examples of the desired input and output of the HMM model; see Table 4.2. In the table, student number 1 has enrolled in four semesters and has four grade-tuples recorded in their academic history. In the first semester, there are no courses with A, B, or C grades as

Table 4.2: Example students' grade-tuple sequences and corresponding estimated academic-performance level sequences

Student	Grade-tuple Seq.	Academic-per. Seq.
1	00011,10100,00211,00011	1,1,1,1
2	11110,11200,21001,13000	2,2,2,2
3	22000,32000,30100	3,3,3
4	30000,31000,30000	4,4,4
5	30000,30000,00010,00010	4,4,1,1

the first three digits are all zero. The fourth and fifth digits in the grade-tuple indicate that the student has grades DF and W in the first semester. The grade-tuple for the rest of the semesters is explained in the same manner. The estimated academic-performance level for this student in all enrolled semesters is 1. As it is shown in the table, while students number 1 through 4 have consistent academic-performance levels during their academic careers, the academic-performance level for student number 5 changed from 4 to 1 in their third semester.

4.2.2 Initialization and Training

For this particular problem, the process for initializing and training the HMM is complex, especially given the total number of parameters that define the emission distributions (i.e., 4 states $\times$ 216 possible grade-tuple outputs).

First, I described the process for providing an initial guess for the smaller 4x4 transition matrix, which has 12 linearly independent parameters (out of 16 total). To identify an initial guess for HMM transition matrix, students' transcripts are analyzed to estimate how their GPAs moved between the four levels identified in Table 4.1 – this is equivalent to assuming a student GPA directly maps to an academic performance level (removing the randomness assertion stated earlier).

The matrix below provides the resulting initial transition matrix (percentage scale):

$$\text{Initial guess for A} = \begin{bmatrix} 81.81 & 17.10 & 1.02 & 0.07 \\ 7.90 & 78.43 & 13.52 & 0.15 \\ 0.74 & 8.54 & 84.44 & 6.28 \\ 0.34 & 1.00 & 10.32 & 88.34 \end{bmatrix}$$

The distribution of students' academic-performance in their first semester is considered as an initial guess for the vector π. The matrix below shows the initial guess for π (percentage scale):

$$\text{Initial guess for } \pi = \begin{bmatrix} 20.14 & 19.07 & 33.50 & 27.29 \end{bmatrix}$$

As it was explained earlier, an HMM observation in this study for a student in a given semester is defined as a tuple with five elements. Elements 1 to 3 store the number of courses with grades A, B, and C, respectively, and elements 4 and 5 are binary variables, determining if the student has D/F and W grades in that semester. In order to find an initial guess for the emission matrix, it is supposed that elements of the observed grade-tuples (i.e. the grade-counts) are independent of each other, with each coming from a Poisson distribution (in the tuning process, the independence assumption between grade counts is removed). Accordingly, the Poisson parameters describing the numbers of A, B, C, etc. grades earned by students in each academic level can be learned from academic data records. Using the Poisson distributions and their parameters, for each academic-performance level, the probability of observing each grade-tuple, i.e., the grades counts in each tuple, is computed using a joint but separable Poisson probability density function. Since there is a total of 216 different grade-tuple combinations in our data set, the initial emission matrix has four rows (corresponding to the four hidden states) and 216 columns (corresponding to the 216 observations). Due to space limitations, instead of showing the initial emission matrix with 864 parameters, the expected values (EV) of the counts for each grade across the 216 possible observations are computed. The results are shown in the following matrix:

$$\text{Initial EV(grade)} = \begin{bmatrix} 0.393 & 0.834 & 0.894 & 0.559 & 0.224 \\ 0.817 & 1.322 & 0.852 & 0.250 & 0.128 \\ 1.477 & 1.387 & 0.462 & 0.087 & 0.074 \\ 2.342 & 0.756 & 0.095 & 0.012 & 0.033 \end{bmatrix}$$

In the above matrix, rows 1 through 4 correspond to academic-performance levels 1 to 4, while columns 1 to 5 provide the expected value for the number of courses with an A, B, C, D/F, and W grade each semester. For example, for students in academic-performance level 3 (the third row), in each semester, the students are expected to earn 1.477 A's, 1.387 B's, and 0.462 C's.

After initializing HMM parameters, the next step is to estimate their optimal value using the Baum-Weltch algorithm. The optimal estimates, obtained after 20 iterations for algorithm convergence, are listed below:

$$\text{A} = \begin{bmatrix} 86.11 & 10.93 & 2.44 & 0.52 \\ 8.50 & 80.00 & 11.41 & 0.09 \\ 0.29 & 7.58 & 84.90 & 7.23 \\ 0.02 & 0.02 & 5.64 & 94.32 \end{bmatrix}$$

$$\text{EV(grade)} = \begin{bmatrix} 0.306 & 0.661 & 0.861 & 0.606 & 0.290 \\ 0.783 & 1.470 & 0.907 & 0.211 & 0.097 \\ 1.765 & 1.421 & 0.302 & 0.038 & 0.055 \\ 2.554 & 0.459 & 0.023 & 0.002 & 0.020 \end{bmatrix}$$

$$\pi = \begin{bmatrix} 12.99 & 36.74 & 35.97 & 14.30 \end{bmatrix}$$

Again, instead of providing the probability of observing each of the 864 possible grade-tuples, the expected earned grades is reported.

4.3 Analysis

The trained values of the HMM reported in the methodology helps to analyze academic-performance trajectories. Based on the estimated transition matrix A, most students maintain their academic-performance level (mode) from one semester to the next. Looking at the diagonal of the matrix, 86.11% of students in academic-performance level 1 in semester t will remain at the same academic-performance level in semester $t + 1$. These probabilities for students in academic-performance levels 2, 3, and 4 are 80.00%, 84.90%, and 94.32%, respectively. Based on these values, we are able to assert, with the exception of performance-level 1, that increasing performance-levels are increasingly stable, so a student in performance-level 4 is more likely to remain at the same performance-level when compared to a student in performance-level 2 or 3.

Matrix EV(grade) provides the expected values for grade counts using the estimated emission matrix. As illustrated in the matrix, students in academic-performance level 4 on average have more courses with A grades compared to the students in academic-performance level 1 (2.554 $>$ 0.306). Technically, while W grades are not used to compute semester GPAs, the last column of the matrix EV(grade) indicates there is a significant relationship between W grades and students' academic-performance levels. Based on the columns, students with academic-performance level 1 are more likely to withdraw from courses in any given semester compared to students in other academic-performance levels (0.290 $>$ 0.097 $>$ 0.055 $>$ 0.020). Finally, matrix π provides an estimate of the distribution of academic-performance levels for students' first semester. Based on the matrix, most students start their academic careers at academic-performance levels 2 and 3 (36.74%+35.79%).

Figure 4.3 depicts the distribution of students' academic-performance level at UCF. Levels 1 through 4 correspond to the students who maintained a consistent academic-performance level over all enrolled semesters. The group *Other* corresponds to the students who had a change between academic-performance levels during their academic career at UCF. The synthetic student number 5 in Table 4.2 is one such example for this category. This student has an academic-performance

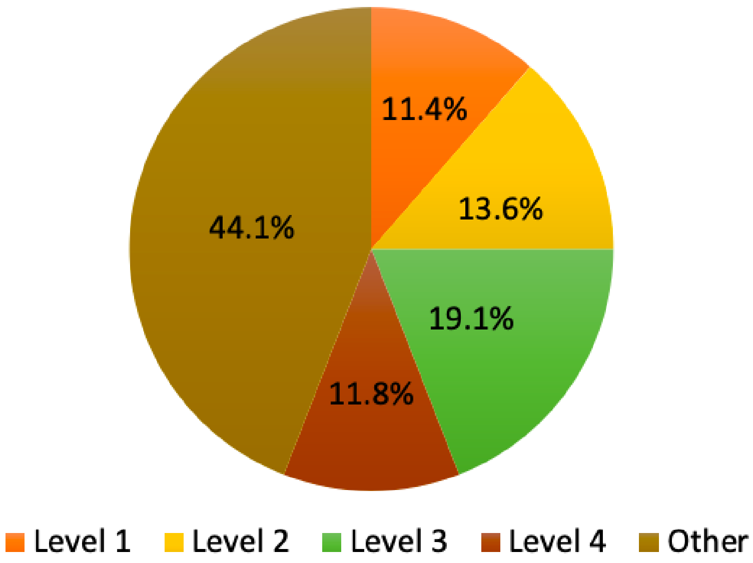


Figure 4.3: Distribution over students' academic-performance level at UCF

level of 4 in their first two semesters and then change to academic-performance level 1 in their third semester. As reported in the figure, 44.1% of students change academic-performance levels during their academic careers.

Table 4.3 compares distributions over academic-performance levels for students with different genders. As the table suggests, female students are more likely to have a academic-performance level of 4 compared to male students (7.9%<14.3%). Also, female students are associated with academic-performance level 1 with less probability compared to male students (8.7%<15.5%)[1]. Based on these results, there is a significant correlation between students' gender and their academic-performance levels at UCF, where, female students on average have a higher academic-performance level compared to male students. These results confirm previous findings obtained by Masic et al. [95], in which female students on average have a higher performance than male students.

[1] Conducted chi-squared test demonstrates that the difference between academic-performance levels distributions for female and male students is statistically significant (*p-value*=0).

Table 4.3: Gender distribution for students with different academic-performance levels at UCF

	Level 1	Level 2	Level 3	Level 4	Other
Female	8.7%	12.8%	21.0%	14.3%	43.2%
Male	15.5%	16.4%	17.0%	7.9%	43.2%

Table 4.4: Race distribution for students with different academic-performance levels at UCF

	Level 1	Level 2	Level 3	Level 4	Other
White	10.7%	14.1%	19.8%	12.7%	42.7%
Hispanic	13.0%	14.4%	19.0%	9.9%	43.7%
African-Am.	16.1%	17.7%	15.0%	4.9%	46.3%
Other race	12.7%	13.5%	18.1%	12.1%	43.6%

Table 4.4 shows academic-performance level distributions for students with different race/ethnicity. As illustrated in the table, White students have the highest rates of being in academic-performance level 4, followed by Hispanic students and African-Am. students, respectively (4.9%<9.9%<12.7%). Also, self-reported white students are less likely to have an academic-performance level of 1 compared to Hispanic and African-Am. students (10.7%<13.0%<16.1%)[2]. These findings are aligned with statistics reported by the U.S department of education [96], which show there exists an academic performance gap between students of different ethnicities and races. This academic gap indicates that white students have the highest academic performance- in terms of degree awarded rate-, followed by Hispanic students, followed by American-Am. students.

Of particular interest to this paper is the *Other* group, which consists of students whose academic-performance changes levels during their academic career. Among these students, 75.2% switch their academic-performance levels only once, 20.5% switch their academic-performance levels twice, and 4.3% change their academic-performance levels more than 2 times. The distribution of switching types for students with one switch is summarized in Table 4.5. Based on the table, the more common switches between academic-performance levels corresponds to level 3 →level 4 (28.4%), level 2 →level 3 (27.6%), and level 2 →level 1 (21.1%). For students with

[2] Conducted chi-squared test demonstrates that the difference between academic-performance levels distributions for students with different races is statistically significant (*p-value*=0).

Table 4.5: Distribution of switching types for students with one switch in academic-performance level

Switch	Ratio (N)	Switch	Ratio (N)
level 1 →level 2	2.6% (313)	level 3 →level 1	0.2% (29)
level 1 →level 3	1.2% (139)	level 3 →level 2	13.1% (1566)
level 1 →level 4	0.5% (63)	level 3 →level 4	28.4% (3416)
level 2 →level 1	21.1% (2540)	level 4 →level 1	0.0% (1)
level 2 →level 3	27.6% (3317)	level 4 →level 2	0.0% (0)
level 2 →level 4	0.0% (0)	level 4 →level 3	5.3% (636)

Table 4.6: Comparing halt rates between students with no switches in academic-performance level and one switch in academic-performance level

Staying in	Halt rate (N)	Switching	Halt rate (N)
Level 1	97.0% (4146)	level 1 →level 2	8.9% (313)
		level 1 →level 3	6.5% (139)
		level 1 →level 4	4.8% (63)
Level 2	39.8% (4933)	level 2 →level 1	72.0% (2540)
		level 2 →level 3	4.3% (3317)
		level 2 →level 4	–
Level 3	20.6% (6896)	level 3 →level 1	72.4% (29)
		level 3 →level 2	8.7% (1566)
		level 3 →level 4	2.0% (3416)
Level 4	11.8% (4295)	level 4 →level 1	100% (1)
		level 4 →level 2	–
		level 4 →level 3	3.5% (636)

one switch in their academic-performance level, 60.3% improve their academic-performance levels, while 39.6% worsen. For the group with two switches, the percentages of students ultimately improving and worsening their academic-performance level are 41.1% and 15.7%, respectively. Moreover, 43.2% of these students go back to their initial academic-performance level after two switches (for example, level 3 →level 2 →level 3).

Table 4.6 compares the halt rate for a subset of academic-performance level trajectories. Column *Staying in* corresponds to students who kept a consistent academic-performance level. As the corresponding halt rate column suggests, the higher the academic performance level, the lower the halt rate (97.0% $>$ 39.8% $>$ 20.6% $>$ 11.8%). Furthermore, it is seen that students that switch academic-performance levels from level 1 to other levels had a substantially decreased halt rate (from 97.0% to 8.9%, 6.5%, and 4.8% for levels 2, 3, and 4). Also, switching from any level to level 1 increases the halt rate significantly. For example, all students whose academic-performance levels changed from 4 to 1 left school without a degree. The halt rate for students that switched from level 3 to 1, and from level 2 to 1 are 72.4% and 72.0%. However, there are surprising results for students starting at academic-performance level 3, where it is observed that students who decrease to academic-performance level 2 have a lower halt rate than if than students that had remained at level 3 (20.6%>8.7%). The conducted proportion hypothesis test suggested that the difference is statistically significant (p-value= $4.75 \times e^{-28}$). A similar pattern is observed for students whose academic-performance levels are changed from 4 to 3. When combined, this evidence implies that switching from a high academic-performance level to a lower academic-performance level does not necessarily result in a higher halt rate. Another surprising result emerges from the grouping and analysis of the academic trajectories that involve improving academic performance. For students that improve their academic performance, their halt rate is substantially lower than those students that maintained a consistent academic performance level. For example, the halt rate for students who transitioned from level 2 to level 3 is lower than those students who were always at level 3 (4.3% $<$ 20.6%). The conducted proportion hypothesis test suggested that the difference is statistically significant (p-value= $4.98 \times e^{-102}$). Even students that transitioned from level 1 to level 2 have a lower halt rate than those students consistently in level 4.

4.4 Conclusion and limitations

In this chapter, a new academic feature corresponding to academic-performance levels, was introduced using a Hidden Markov Model. Then, the students' academic-performance trajectories were analyzed to investigate their correlations to the students' final academic outcomes in terms of halt rates. Unlike traditional statistical methodologies, the proposed approach is able to provide a standard point of reference for comparing a student's GPA both across their enrolled semesters and to other students. The proposed HMM model takes the sequence of course grades over multiple semesters and returns the sequence of estimated academic-performance levels. The estimated HMM parameters illustrate that while W grades are not involved in computing students' GPA, there is a significant relationship between students' academic-performance level and withdrawing from courses. Also, by tracking students' transitions between four main academic-performance levels, it was observed that a significant portion of students (44.1%) switch between academic-performance levels during their education.

Furthermore, analyzing and comparing the halt rate for students with consistent academic performance levels, evidence suggests that students who constantly maintain a low academic-performance level are more likely to leave school without a degree. Therefore, the following findings are the answers to the mentioned research questions at the beginning of this chapter:

Finding: For students with a change in their academic-performance level, it was shown that switching from any academic-performance levels to level 1 increases the halt rate substantially.

Findings: The higher academic-performance level, the lower is the halt rate.

Findings: Any switch from level 1 to other level, substantially decrease in halt rate.

Findings: Any switch from other levels to level 1, substantially increase in halt rate.

Findings: Switching from a high academic-performance level to a low academic performance level does not necessarily increase the halt rate.

Findings: Students that improve their academic performance levels, decrease their halt rate below students that maintain consistent levels.

As discussed in this chapter, one of the main challenges of HMMs correspond to the dimensionality of the observations. In order to tackle this problem, two different approaches were used in this chapter. First, an upper limit was considered on the elements of the grade-tuple which restricted the number of A's, B's, and C's to the maximum of 3 and transformed the number of D/F's and W's grades to binary variables. Second, since the probability of some grade combinations is zero, each unique grade-tuple was considered as one possible HMM observation. This assumption decreased the total number of unique observations to 216 (from 687). Although leveraging these problem-specific features helped reduce the size of the problem that was addressed in this chapter, in general, such conditions may not always exist. For a model with N states and M observation, the total number of parameters within an HMM to be tuned is equal to $N(N+M-2)+N$, as such, high dimensionality can affect the accuracy of an HMM model in the majority of settings. In Chapter 6, an alternative approach is proposed to tackle this problem.

CHAPTER 5

HIDDEN MARKOV MODELS: MODELING AND IMPLEMENTATION

Markov models are commonly used to model systems and processes that exhibit stochastic dynamics, specifically systems and processes that switch between states or operating modes. When representing a system as a Markov model, modeling allows for probabilistic switching between states over each time-step according to static probability distributions. Falling within the general branch of mathematics of stochastic processes, Markov models are defined by a number of unique properties that simplify their representation, thereby allowing for tractability when special cases are considered, specifically, the observability of system states. In the standard Markov model (sometimes called a Markov chain), modeling assumes that the state of the system is *fully-observable* and known at all times. In alternative forms, the system state may be modeled as *partially-observable* or even completely *hidden* so that operating states are not directly known. The usage and application of fully-observable, partially-observable, and hidden Markov models are problem-dependent. Regardless, when learning model parameters for a Markov model (e.g. switching probabilities) from data, or when estimating the current state of a system from observable data, the hidden Markov model (HMM) is more challenging to interpret than the standard and the partially observable Markov model, since the dynamics of the underlying system in the HMM is unknown and must be inferred.

In this chapter, after briefly introducing and reviewing the hidden Markov model, the mathematical and algorithmic implementations of the Forward algorithm, the Backward Algorithm, the Baum-Welch algorithm, and the Viterbi algorithm are discussed. These algorithms are necessary when learning model parameters or estimating system states based on observed data. While the mathematical derivations of these algorithms are well-known, the computational implementation of the algorithms presents unique challenges due to numeric underflow (i.e. extremely small numbers). And while numerous techniques have sought to overcome the underflow challenge by calculating probabilities in the log space and applying the log-sum-exponential trick, the contribution of this chapter is the presentation of a simpler solution based on unit normalization of probabilities. The benefit of a normalization approach is that it is simple while ensuring that state probabilities never reach the numerical tolerance of computer systems. An additional contribution of this chapter and the subsequent chapter is the pseudocode and software implementation in Python. The software implementation is flexible enough to implement the Multivariate Poisson hidden Markov models in chapter 6, making it the first known implementation of this kind.

5.1 Mathematical representations of Markov models

A graphical representation of a simple Markov model with three observable states (e.g., states 1, 2, and 3) is illustrated in Figure 5.1. For an arbitrary Markov model, the set of N available states the system can operate in is denoted by $S = \{s_1, s_2, \ldots, s_N\}$. Assuming the system evolves randomly, at each time-step t the random variable X_t, indicating the current state of the system, draws from S; accordingly, x_t references an instantiation of the random variable. Accordingly, as depicted in the figure the state of Markov model at each time-step t, denoted x_t, is completely observable from $t = 0$ until when the system is terminated. In the figure, the transition probabilities between states are indicted by the values $a_{i,j}$, which refer to the probability that the system switches from state i to state j at any given time-step. For example, if at time-step t the system is in state 1 (so that $x_t = s_1$), the system will remain in state 1 ($x_{t+1} = s_1$) with the probability of $a_{1,1}$ or will switch to state 2 ($x_{t+1} = s_2$) with the probability of $a_{1,2}$ at time-step $t+1$. Since each value $a_{i,j}$ indicates the probability of $p(X_{t+1} = s_j | X_t = s_i)$, the laws of probability require that for an N-state Markov model $\sum_{j=1}^{N} a_{i,j} = 1 \; \forall i \in \{1, \ldots, N\}$.

Standard Markov models have two key properties that significantly simplify their representation, along with associated calculations. The first, known as the Markovian property, assumes that the probability of switching to any particular state is dependent solely on the cur-

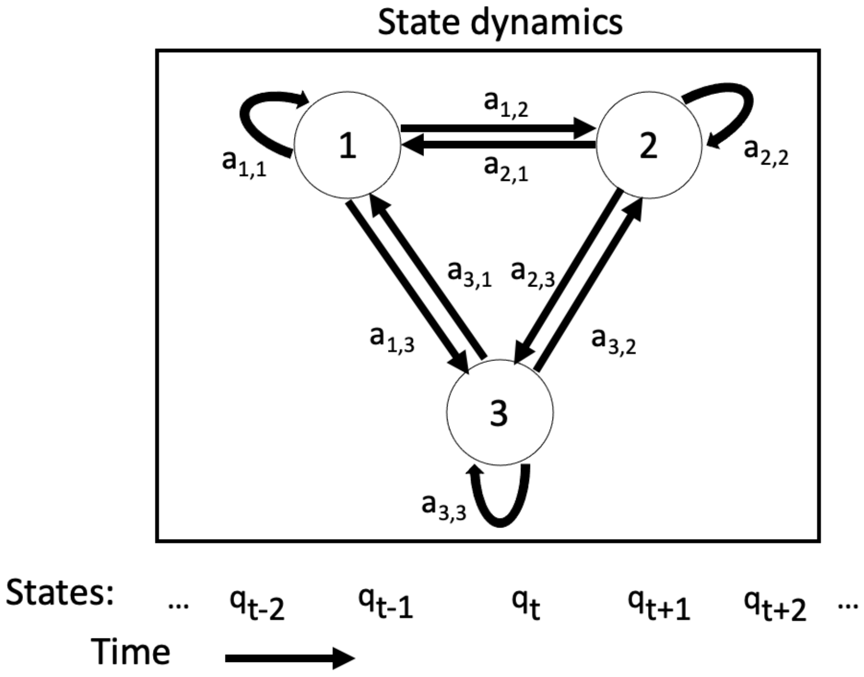


Figure 5.1: Representation of a simple Markov Chain

rent state, and not prior states that preceded it [97]. In other words, the future state has no relationship with the past. This property is mathematically represented by the probability equation $P(X_t = x_t|x_1, x_2, ..., x_{t-1}) = P(X_t = x_t|x_{t-1})$. Additionally, Markov chains assume static transition probabilities so that $a_{i,j} = p(X_{t+1} = x_j|X_t = s_i)\ \forall t$.

For a Markov model the current state of a system can described by a state vector $u \in \mathcal{R}^N$, where $u_i(t)$ refers to the probability that the system is operating in state i at time-step t. That is to say, $u_i(t) = P(X_t = s_i)$. When an observation is made, whereby the state of the system is observed to be in an arbitrary state i, then the state vector distribution collapses into an indicator vector so that $u_i(t) = 1$, while $u_j(t) = 0\ \forall j \neq i$ and $j \in \{1, \ldots, N\}$. When this representation of the probabilistic state of the system is combined the Markovian property and static transition probabilities, $a_{i,j}$, the one-step dynamics of a Markov model, describing the flow of probabilities between states in time, can be written using the equation $u(t+1) = A^T u(t)$ where $A = [a_{i,j}]$. The one-step transition equation can be extended to calculate the state distribution $u(t+n)$ $n-$steps

Table 5.1: Components of each Markov chain model

Component	Description
$S = \{s_1, s_2, ..., s_N\}$	The set of N states
$A = [a_{i,j}]$	Transition matrix $A \in \mathcal{R}^N$, where each $a_{i,j}$ indicates the single-step probability by which a system moves from state i to state j
$\pi = [\pi_1, \pi_2, ..., \pi_N]^T$	Initial state distribution where each π_i describes the probability that the system begins in state i at time-step $t = 1$.

after starting with an initial distribution $u(t)$, resulting in the equation $u(t+n) = (A^n)^T u(t)$.

The study of Markov models is often interested in calculating the long-term or steady-state probabilities associated with a particular model. That is to say, calculating $\lim_{t \to \infty} u(t)$ or solving for an Eigenvectors ν that satisfies the equation $\nu = A^T \nu$; for many Markov models the long-term probabilities are the same solution to the Eigenvalue problem. In this dissertation focusing on the academic performance of students, long-term probabilities and steady-state probabilities are of limited usefulness and not considered; given that students typically enroll in a university for a fixed number of semesters, the notion of long-term probabilities does not apply. Instead, it is more relevant to consider the initial state of students when entering a university and the ending state of students, whether it be graduating or halting according to 6-year criteria used to evaluate universities. Within the context of Markov models, the initial state distribution is denoted $\pi \in \mathcal{R}^N$, where the i^{th} term $\pi_i = P(X_1 = s_i)$.

A summary of the relevant parameters describing a standard Markov model, along with their corresponding descriptions is summarized in Table 5.1.

5.2 Parameter learning for a Markov model

In many cases, the initial modeling of a system requires learning the Markov model parameters described in Table 5.1 using historical data. For the most part, this process is not particularly difficult, as the maximum likelihood estimate (MLE) of the parameters, A and π, are essentially calculated using a counting procedure. The learning process begins with a historical data set $\mathcal{D}$ with R state sequences of arbitrary length, $\mathcal{D} = \{D_1, D_2, \ldots, D_R\}$. As denoted, $D_r = \{x_1^r, x_2^r, x_3^r.., x_T^r\}$ representing the r^{th} sequence, which is of length T, contains the observed state of the system at each time-step. The first step in the learning process is to identify the total number of possible states, N, which is given by the number of unique states observed in the historical data such that $N = card(\bigcup_{r=1}^{R} D_r)$.

The initial state distribution π and state transition matrix A are estimated using the maximum likelihood estimate. Notionally for an arbitrary parameter θ, the maximum likelihood estimate $\hat{\theta}$ satisfies $\hat{\theta} = argmax_\theta P(\mathcal{D}|\theta)$, where $\mathcal{D}$ corresponds to observed data and θ is in an allowable set Θ. Calculating the MLE of π using $P(\mathcal{D}|\pi)$ starts by assuming that data samples are independent, and that the initial state distribution can be estimated with the first state, x^r, of each sequence D_r. Accordingly, the joint probability over the observed data set $\mathcal{D}$ can be expanded so that

$$P(\mathcal{D}|\pi) = P(\{D_1, D_2, \ldots, D_R\}|\pi) = \prod_{r=1}^{R} P(D^r|\pi) = \prod_{r=1}^{R} p(x_1^r|\pi). \tag{5.1}$$

Equation 5.1 can be rewritten according to the number of times each initial state is observed to occur over all the state sequences. So if C_i is a count denoting the number of times state i is observed at the beginning of the R state sequences in the set $\mathcal{D}$, then $C_i = card(r|x_1^r = i)$. The probability formula in Equation 5.1 can then be expressed as the product of the probability of each initial state i being observed C_i times so that

$$P(\mathcal{D}|\pi) = \prod_{r=1}^{R} p(x_1^r|\pi) = \prod_{i=1}^{N} p(i|\pi)^{C_i} = \prod_{i=1}^{N} {\pi_i}^{C_i}$$

Since $argmax_x f(x) = argmax_x \ln f(x)$, it follows that for the standard Markov model $argmax_\pi P(\mathcal{D}|\pi_o) = argmax_\pi \sum_{i=1}^{N} C_i \ln \pi_i$. Ultimately, solving for the maximum likelihood estimate of π requires solving the following optimization program:

$$\begin{aligned} \min_{\pi_o} \quad & \sum_{i=1}^{N} C_i \ln \pi_i \\ \text{s.t.} \quad & \sum_{i=1}^{N} \pi_i = 1 \\ & \pi_i \geq 0 \end{aligned} \tag{5.2}$$

The solution to the optimization problem in Equation 5.2 can be computed by taking the partial derivatives of the corresponding Lagrange equation,

$$L(\pi, \lambda, \mu) = \sum_{i=1}^{N} C_i \ln \pi_i - \lambda \sum_{i=1}^{N} (\pi_i - 1) - \sum_{i=1}^{N} \mu_i \pi_i,$$

in terms of the decision variables π_i and setting them equal to 0. For a given π_i, solving for $\frac{\partial L(\pi,\lambda,\mu)}{\partial \pi_i} = 0$ yields $C_i/\pi_i - \lambda - \mu_i = 0$. Assuming $C_i > 0$, and given the requirement that the Lagrange variables satisfy $\lambda \geq 0$ and $\mu_i \geq 0$, then it follows that $\pi_i > 0$. With this knowledge, according to the complementary slackness condition $\mu_i = 0$ since $\mu_i \pi_i = 0 \ \forall i$. In which case,

$$\pi_i = C_i/\lambda, \tag{5.3}$$

while noting that λ is still unknown. However, because π is a probability distribution, such that

$$\sum_{i=1}^{N} \pi_i = 1, \tag{5.4}$$

when substituting Equation 5.3 into Equation 5.4, we are left with $\sum_{i=1}^{N} C_i/\lambda = 1$, which implies $\lambda = \sum_{i=1}^{N} C_i = R$. If follows then that $\pi_i = C_i/R$. In other words, the maximum likelihood estimator of π is given by counting the number of times that state i appears first in each sequence divided by the total number of sample sequences. In the degenerate case when there are no observations in which a sequence starts with state i so that $C_i = 0$, then when following the convention that $0 \ln 0 = 0$, it is implied that $\pi_i = 0$.

Following a similar derivation, the maximum likelihood estimation for the probability of transitioning from state i to state j can be calculated. Essentially the same process is repeated for a given state i by identifying and extracting all states sequences pairs that pass through state i and jump to another state. In this case, the maximum likelihood estimation for the i^{th} row of the transition matrix A is obtained by the summation over the number of times that system is observed to switch from state i to state j, and then normalizing by the total number of transitions from state i to all states. In other words,

$$\hat{a}_{i,j} = \frac{Counter(i \rightarrow j)}{\sum_{k \in S} Counter(i \rightarrow k)} \tag{5.5}$$

5.3 Mathematical representations of hidden Markov models

As depicted in Figure 5.2, similar to the standard Markov models, a hidden Markov model represents the dynamics of a system as it moves between operating states or modes (e.g., state 1, state 2, and state 3 in the figure). When operating within a state, the system generates a state-dependent output o_t at each time-step t; this output can be deterministic or random, however, in most cases, the output is drawn from a distribution. Unlike standard Markov models in which the state of the system is visible (x_t in Figure 5.1 is known), in the case of HMMs, the states are not directly observable, and as such, they can only be inferred by observing the sequence of generated outputs.

Like with the standard Markov model, a hidden Markov model is associated the parameters S, A, and π. Again, $S = \{s_1, s_2, ..., s_N\}$ represents the set of N possible states in the system model, $A = [a_{i,j}] \in \mathcal{R}^{NxN}$ is the transition matrix, where each $a_{i,j}$ denotes the probability of transitioning from state i to state j at any given time-step (i.e. $s_i \rightarrow s_j$), and π corresponds to the initial state distribution. In addition to these parameters, each state i is associated with an emission distribution, $b_i(o)$. That is to say, as a system moves between states over T time-steps it produces a sequence of observation $O = o_1, o_2, ..., o_T$. In the most general form, when a system is operating in a specific state s_i at time-step t, the output o_t is generated according to a unique probability

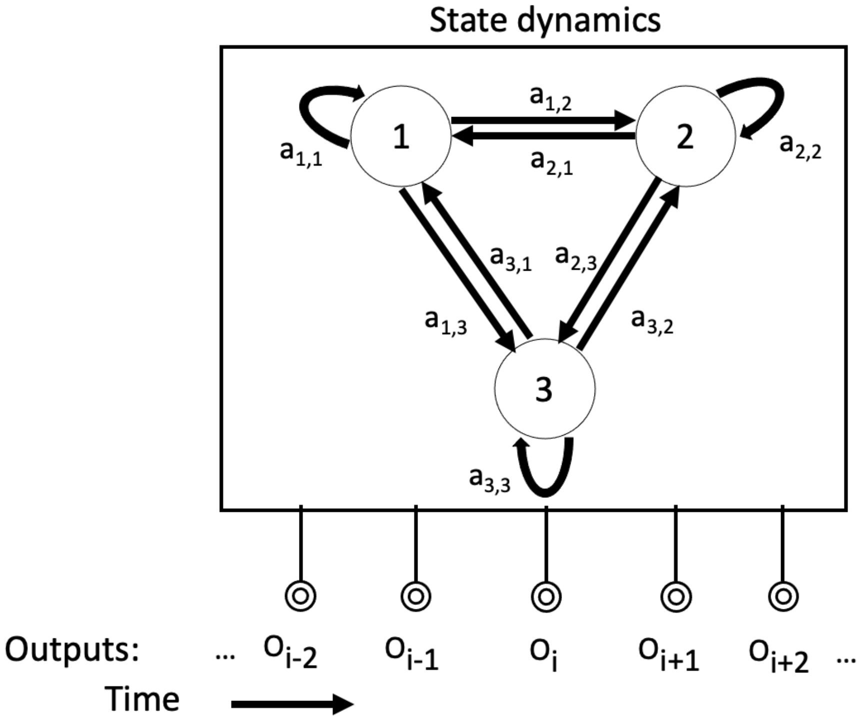


Figure 5.2: Representation of a simple Hidden Markov Model

distribution denoted as $b_i(o)$, whereby $b_i(o_t)$ corresponds to the probability/likelihood of generated the observation. The emission distribution, $b_i(o)$, typically refers to a probability mass function or probability density function depending on if the emission distribution is discrete or continuous. When grouped together, the individual state-dependent emission distributions form the parameter B, such that $B = \{b_1(o), b_2(o), \ldots, b_N(o)\}$ A summary of the parameter set characterizing a hidden Markov model, along with the corresponding descriptions, is provided in Table 5.2.

As an extension of the standard Markov model, HMMs maintain a similar version of the Markovian property: the probability of a future hidden state only depends on the current hidden state, as such the probability of a future hidden state is independent of past hidden states. HMMs have an additional independence assumption associated with observed values, whereby any observation at time-step t is only a function of the current state x_t generating the observation, and therefore is

Table 5.2: Components of each hidden Markov model

Component	Description
$S = s_1, s_2, ..., s_N$	Set of N operating states
$A = [a_{i,j}]\ s.t.\ A \in \mathcal{R}^{NxN}$	Transition matrix A, where each $a_{i,j}$ indicates the single-step probability by which a system moves from state i to state j, i.e. $P(q_{t+1} = s_j \| q_t = s_i)$. Each row of matrix A forms a distribution such that $\sum_{j=1}^{N} a_{i,j} = 1\ \forall i$
$\pi = [\pi_1, \pi_2, ..., \pi_N]^T$	Initial distribution π. Each π_i corresponds to the probability that the system begins in state i. Also, $\sum_{i=1}^{N} \pi_i = 1$
$B = \{b_1(o), \ldots, b_i(o), \ldots, b_N(o)\}$	An set of emission distributions describing the conditional probability of observation o at any time-step t generated from any particular state i

Table 5.3: Difference between Markov chain and Hidden Markov models in terms of model parameters

Model	Q	A	B	π
Markov Chain	✓	✓	×	✓
Hidden Markov Model	✓	✓	✓	✓

independent of all other states $\{x_1, \ldots, x_T\} \setminus x_t$ and observations $\{o_1, \ldots, o_T\} \setminus o_t$. In other words:

$$P(o_t|x_1, x_2, ..., x_T, o_1, o_2, ..., o_T) = P(o_t|x_t)$$

Table 5.3 highlights the differences between the standard Markov model and the hidden Markov model in terms of the model parameters. As mentioned before, the difference between these two models is in the state visibility. Because the states are not visible for HMMs, the states can only be estimated based on the state-dependent emission associated with the parameter B. So while for the standard Markov model the state transition matrix can be estimated through direction observations (recall Equation 5.5), for hidden Markov models, an additional step of estimating the current state at each time-step is required in order to estimate the transition matrix A. It follows, that only once a state estimate is available at each time-step can the emission distribution set B be estimated. However, the process of estimating the current state requires an initial guess of both A and B. This engenders in a cyclic procedure where the best estimate of the current state, along with A and B, are iteratively updated until they all converge.

With the structure of the hidden Markov model described, the next step is to document the algorithms used in practice when iteratively learning the parameters of a hidden Markov model and when applying hidden Markov models to estimate states at each time step. Performing these two tasks requires addressing the following three fundamental problems, each with its corresponding algorithm:

- **Problem 1 (Likelihood)**: Given an HMM with assumed parameters $\lambda = (A, B, \pi)$ and an observation sequences O, computed the likelihood $P(O|\lambda)$.
  Algorithm: Forward

- **Problem 2 (Estimating States)**: Given an HMM with assumed parameters $\lambda = (A, B, \pi)$ and an observation sequences O, estimate the most probable sequence of hidden states.
  Algorithm: Viterbi

- **Problem 3 (Learning Parameters)**: Given set of a set of observation sequences and assumed number of states n, estimate the hidden Markov model parameters $\lambda = (A, B, \pi)$.
  Algorithm: Baum-Welch

5.4 Likelihood calculations for hidden Markov models

The problem of state estimation and parameter learning for a hidden Markov model begins by addressing the likelihood problem, which corresponds to computing the likelihood probability $P(O|\lambda)$ given a HMM with the parameter set $\lambda = (A, B, \pi)$ and observation sequence O. Because each state is mutually exclusive, applying the Law of Total Probability, the likelihood function can be decomposed based on all possible terminal states, x_T, as follows:

$$P(O|\lambda) = \sum_{i=1}^{N} P(O, x_T = i|\lambda)$$

The algorithm for computing the likelihood function, commonly called the Forward algorithm, is used to calculate each of the terms $P(O, x_T = i|\lambda)$. It begins by defining probability value $\alpha_i(t)$ as the probability of observing the sequence of observations $o_1, \ldots, o_t$ and ending in state i at time-step t. The corresponding probability definition is $\alpha_i(t) = P(o_1, \ldots, o_t, x_t = i|\lambda)$. Recognizing that $P(O, x_T = i|\lambda) = \alpha_i(T)$, the likelihood function can be rewritten as

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_i(T)$$

A detailed derivation and explanation of $\alpha_i(t)$ can be read in a number of resources on the topic here: [98]. The goal of this chapter is to explore the programming implementation of calculating $\alpha_i(t)$, with the goal of adjusting it as needed to suit a generalized implementation of HMMs. More specifically, one objective of this dissertation is to develop, program, and validate a HMM implementation that allows for state-dependent emission distributions to be modeled as a Poisson distribution (see chapter 7). Based on the scope of this dissertation, the calculation for $\alpha_i(t)$ is now presented without significant discussion.

Calculation of $\alpha_i(t)$ begins with the initialization equation for $\alpha_i(1) = P(o_1, x_t = i|\lambda)$ given by

$$\alpha_i(1) = \pi_i b_i(o_1), \text{ for } 1 \leq i \leq N \tag{5.6}$$

where π refers to the initial starting distribution for the state of the system. In most cases the true value of π is unknown, and as such an estimate is provided. To calculate $\alpha_i(t)$ for subsequent time-steps, the updated equation

$$\alpha_j(t) = \sum_{i=1}^{N} \alpha_i(t-1) a_{i,j} b_j(o_t), \text{ for } 2 < t \leq T \text{ and } 1 \leq j \leq N \tag{5.7}$$

is applied. As stated, Equation 5.6 can be calculated without encountering numerical underflow issues as long as each of the initial state probabilities π_i and state-observation probabilities $b_i(o_1)$ are within numerical tolerances. However, the iterative forward calculation for Equation 5.7 leads to underflow issues as the probability of an observation sequence approaches zero when the sequence is longer regardless of the ending state. That is to say $P(o_1..o_t|\lambda) \to 0$ as t grows; even sequences with lengths less than 10 can produce underflow issues for small systems.

Closely associated with the Forward Algorithm is the Backward Algorithm, which seeks to calculate $\beta_i(t) = P(o_{t+1}, \ldots, o_T|\lambda, x_t = i)$. So instead of providing the probability of observing a sequence of emissions from time-step 1 to time-step t while passing through state i so that $x_t = s_i$, the Backward Algorithm calculates the probability of observing the remaining sequence of emissions from time-step $t + 1$ to time-step T, given that the system terminates at $x_t = s_i$. The benefit of calculating $\beta_i(t)$ is that when combined with $\alpha_i(t)$ it becomes possible to calculate $\gamma_i(t) = P(x_t = s_i|o_1, \ldots, o_T, \lambda)$, corresponding to the total probability of being in a given state i at any given time, t. It is important to note, that while $\gamma_i(t)$ can be used to calculate the most likely state at any instance, that is not equivalent to calculating the most likely state sequence given a series of observations. The process for calculating $\beta_i(t)$ begins by initialization $\beta_i(T) = 1$ for all states i. Next, the backward induction calculation is given by

$$\beta_i(t) = \sum_{j=1}^{N} = \beta_j(t+1)a_{i,j}b_j(o_{t+1}). \tag{5.8}$$

The calculation of $\gamma_i(t)$ follows by applying Bayes Law and independence between observations, so that

$$\gamma_i(t) = P(x_t = s_i|o_1, \ldots, o_T, \lambda) = \frac{P(x_t = s_i, o_1, \ldots, o_T|\lambda)}{P(o_1, \ldots, o_T|\lambda)} = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^{N} \alpha_j(t)\beta_j(t)}$$

Like with the Forward Algorithm, the calculations associated with $\beta_i(t)$ and $\gamma_i(t)$ often result in underflow issues whereby the products of the probability values quickly reach the numerical tolerances (i.e. the dynamic range) of computing systems as the number time-steps and states increases.

5.5 State estimation for hidden Markov models

Since the state of a hidden Markov model is not directly observable, it is necessary to infer the most likely state sequence using available observations; this problem is commonly referred to as *decoding*. For hidden Markov models, the decoding task is accomplished using the Viterbi algorithm. The foundation of the Viterbi algorithm begins by defining the term $\delta_i(t)$ where

$$\delta_i(t) = \max_{x_1,\ldots,x_{t-1}} P(x_1, \ldots, x_{t-1}, x_t = s_i, o_1, \ldots, o_t|\lambda) \tag{5.9}$$

corresponds to the probability of the most likely partial state sequence $x_1, \ldots, x_{t-1}$ that results in the observation sequence $o_1, \ldots, o_t$. While it is theoretically possible that Equation 5.9 could be used iteratively over all possible options when finding the optimal state sequence, such an implementation is practically infeasible given that the number of combinations grows exponentially with the number of states, observations, and time-steps. The Viterbi algorithm utilizes a dynamic programming approach to significantly reduce the decision-space that is explored. Essentially, the Viterbi algorithm uses all optimal state sequences up to a prior time-step and state in order to find the optimal state sequence to the considered state at the current time-step. Through this inductive reasoning, it is possible to find the optimal state sequence over all time-steps.

The Viterbi algorithm is initialized with

$$\delta_i(0) = \pi_i b_i(o_1) \quad \forall i \in \{1, \ldots, N\}.$$

The following recursion step is then used to construct the set of N optimal sequences up to each state i at time-step $t \in \{2, \ldots, T\}$ stored in $\psi_i(t)$:

$$\begin{aligned} \delta_j(t+1) &= \max_{i \in \{1,\ldots,N\}} \delta_i(t-1) a_{i,j} b_j(o_t) \\ \psi_j(t) &= \operatorname*{argmax}_{i \in \{1,\ldots,N\}} \delta_i(t-1) a_{i,j} b_j(o_t) \end{aligned}$$

At termination, the probability corresponding to the optimal state sequence, P^*, and the optimal end-state, $x^*(T)$ is given by

$$P^* = \max_{i \in \{1,\ldots,N\}} \delta_i(T)$$
$$x(T)^* = \underset{i \in \{1,\ldots,N\}}{\operatorname{argmax}} \delta_i(T)$$

Starting from the optimal end-state $x(T)^*$, the full optimal state-sequence can be extracted by back-tracking through $\psi_j(t)$, with the equation

$$x(t)^* = \psi_{x(t+1)^*}(t+1),\ t = T-1, T-2, \ldots, 1,$$

which is essentially an application of the Principle of Optimality.

Once again, as with the Forward and Backward algorithms, the Viterbi algorithm is vulnerable to underflow issues associated with propagating probabilities during the recursion step.

5.6 Parameter learning for a Hidden Markov model

Learning the parameter set $\lambda = (A, B, \pi)$ is a complex, yet well-documented process. Without a clear determination of the state at each time-step the estimates of A, B, and π – which all reference the states in some manner – can only be inferred from probabilistic observations. Complicating the learning of model parameters is that the corresponding optimization has no guarantees of convexity, as such, it is possible that there exist multiple optimal solutions to the maximum likelihood estimates of A, B, and π.

For a hidden Markov model, the process for learning the parameter set λ makes use of the Baum-Welch algorithm, a special case of the expectation-maximization algorithm. The expectation-maximization algorithm is an iterative process that is best understood by the two named steps:

Expectation Estimate the missing and latent variables in the data set (i.e. the current state of the system at each time-step).

Maximization Maximize the parameters of the model in the presence of the data (i.e. the parameter set λ).

Following these two steps repeatedly. First, using an initial guess for the parameter set λ the likelihood of the hidden states associated with an observation sequence is estimated. Next, using the estimated hidden states the estimate of the parameter set λ is updated. This process of estimating the hidden states, followed by updating the estimate of the parameter set λ is repeated until there is convergence.

The expectation step begins by calculating two quantities: **(1)** the likelihood of being in state i at time-step t given an observed sequence $O = \{o_1, \ldots, o_T\}$; and **(2)** the likelihood of transitioning from state i to state j at time-step t given the same observed sequence O. The first quantity corresponds to $\gamma_i(t)$ described in section 5.4. The later quantity is introduced here, and denoted as $\xi_{i,j}(t)$ where

$$\begin{aligned}\xi_{i,j}(t) &= \frac{P(x_t = s_i, x_{t+1} = s_j, O|\lambda)}{P(O|\lambda)} \\ &= \frac{\alpha_i(t)a_{i,j}\beta_j(t+1)b_j(o_{t+1})}{\sum_{k=1}^{N}\sum_{w=1}^{N} P(x_t = s_k, x_{t+1} = s_w, O|\lambda)} \\ &= \frac{\alpha_i(t)a_{i,j}\beta_j(t+1)b_j(o_{t+1})}{\sum_{k=1}^{N}\sum_{w=1}^{N} \alpha_k(t)a_{k,w}\beta_w(t+1)b_w(o_{t+1})}\end{aligned}$$

Each of these quantities is based on the current estimate of the parameter-set λ, which gets updated over multiple iterations of the expectation-maximization process.

Next, utilizing the quantities $\gamma_i(t)$ and $\xi_{i,j}(t)$, in the maximization step the current estimated parameters $\hat{\pi}$, $\hat{A}$, and $\hat{B}$ are updated. The updated estimated for $\hat{\pi}$ comes directly from $\gamma_i(t)$; as $\pi_i = P(x_1 = s_i|O, \lambda)$ it follows that

$$\hat{\pi}_i = \gamma_i(1). \tag{5.10}$$

The calculation for the estimated transition matrix $\hat{A} = [\hat{a}_{i,j}]$ follows a process similar to maximum likelihood estimation detailed in section 5.2. Instead of calculating the maximum likelihood estimate $\hat{a}_{i,j}$ according to the notional equation

$$\hat{a}_{ij} = \frac{Counter(i \to j)}{\sum_{q \in Q} Counter(i \to q)}$$

the Baum-Welch algorithm accounts for the uncertainty in the state estimate. As such, the estimate equation for $\hat{a}_{i,j}$ is given by

$$\hat{a}_{i,j} = \frac{\text{Expected number of state transition from } i \text{ to } j}{\text{Expected number of state transition from } i} = \frac{\sum_{t=1}^{T-1} \xi_{i,j}(t)}{\sum_{t=1}^{T-1} \sum_{k=1}^{N} \xi_{i,k}(t)}, \tag{5.11}$$

which can be understood as a weighted average based on the probability of passing between two state.

In the case of estimating the state-dependent emission distributions $B = \{b_1(o), \ldots, b_N(o)\}$ the process is dependent on the underlying distribution. Regardless, this step still requires the maximization of a likelihood function corresponding to the probability of generating an observation based on the underlying probability distribution. The process for deriving the distribution parameters, θ_i, for an arbitrary state state-distribution $b_i(o)$ can be understood by first considering the case when it is assumed that the state sequence at each time-step, $x(t)$, is known. In this case, solving for the optimal distribution parameters θ_i^* is simply given by $\theta_i^* = \text{argmax}_{\theta_i} P(O_{s_i}|\theta_i)$ where O_{s_i} is the set of observations when the system is known to be in state i, so that $O_{s_i} = \{o_t | x_t = s_i\}$. For a hidden Markov model, because the current state of the system is unknown, instead of optimizing $P(O_{s_i}|\theta_i)$ directly, a weighted optimization is considered where the weightings are given by the probability that the observation is generated from state i. Accordingly,

$$\theta_i^* = \underset{\theta_i}{\operatorname{argmax}} \frac{\sum_{t=1}^{T} P(o_t|\theta_i) P(x_t = s_i|O)}{\sum_{t=1}^{T} P(x_t = s_i|O)} \tag{5.12}$$

$$= \underset{\theta_i}{\operatorname{argmax}} \sum_{t=1}^{T} P(o_t|\theta_i)\gamma_i(t) \tag{5.13}$$

$$= \underset{\theta_i}{\operatorname{argmax}} \sum_{t=1}^{T} b(o_t|\theta_i)\gamma_i(t) \tag{5.14}$$

since $P(x_t = s_i|O) = \gamma_i(t)$, and the denominator can be ignored as it is a constant value that does not affect the optimal solution. Unlike the previous calculations discussed in this chapter, solving for the optimal distribution parameters θ_i for each state-dependent distribution $b_i(o)$ does not typically induce underflow.

5.7 Computational implementation of HMM algorithms

In the previous sections, the underlying calculations associated with hidden Markov models were discussed. Despite their mathematical derivations, [99, 50], the previously discussed algorithms do not work in practice. The primary weakness is traced to the Forward and Backward algorithms in which it is common for the equations to have numeric underflow issues as the probabilities become increasingly small. Once underflow occurs, then all subsequent calculations as part of the Baum-Welch algorithm, and subsequently, the Viterbi algorithm, fails. In the next sections, the problem of underflow is resolved using two techniques. The first technique utilizes log-based calculations along with the log-sum-exponential trick. The second technique, which is shown to require simpler calculations, is based on the inclusion of a normalization step.

5.7.1 Reformulating HMM in log scale to resolve the underflow issue

As noted before, the primary reason that the implementations of HMMs fail is that the probability values exceed the dynamic range of most computers when performing calculations, thereby resulting in the underflow. A common solution to address the underflow problem is to reformulate all equations in a log-scale space, followed by using the log-sum-exp trick. While these approaches are commonly used, this section contributes explicit pseudocode used as part of programming implementations.

Working in the log-space begins by calculating $\log(\alpha_j(1))$. From Equation 5.6 it follows that $\log(\alpha_j(1)) = \log(\pi_i) + \log(b_i(o_1))$. Furthermore, from Equation 5.7

$$\begin{aligned} L\alpha_j(t) &= \log(\alpha_t(j)) \\ &= \log\left(\sum_{i=1}^{N} \alpha_{t-1}(i) a_{i,j} b_j(o_t)\right) \\ &= \log\left(\sum_{i=1}^{N} e^{\log(\alpha_i(t-1)a_{i,j}b_j(o_t))}\right) \\ &= \log\left(\sum_{i=1}^{N} e^{\log(\alpha_i(t-1))+\log(a_{i,j})+log(b_j(o_t))}\right) \end{aligned} \tag{5.15}$$

While the term $\log(\alpha_i(t-1)) + \log(a_{i,j}) + log(b_j(o_t))$ can be calculated without difficulty, it is often a large negative number (e.g. $<< -500$), so that when computing the exponential in the formula above, the result is an extremely small value (e.g. $<< e^{500} = 7.124576e - 218$). In the literature, a commonly used solution for this is known as the log-sum-exp trick. As an example, suppose that the aim is to calculate $\log \sum_{i=1}^{N} \exp(x_i)$. The following equation shows how this trick works:

$$\begin{aligned} \log\left(\sum_{i=1}^{N} \exp(x_i)\right) &= \log\left(\sum_{i=1}^{N} \exp(x_i - y)\exp(y)\right) \\ &= \log\left(\exp(y)\sum_{i=1}^{N} \exp(x_i - y)\right) \\ &= y + \log\left(\sum_{i=1}^{N} \exp(x_i - y)\right) \\ &\sim y \quad \text{when all other } x_i - y << 0 \end{aligned}$$

The key to the above equation is that the best value for y is $y = \max\{x_i\}$. In this case, even if all x_i values are large-magnitude negative numbers, the effect of arithmetic underflow can be mitigated. As an example, suppose we have the following numbers: $x_1 = -2510$, $x_2 = -2500$, and $x_3 = -2515$. In this case, $y = -2500$, therefore, there are no difficulties in computing the following equation:

$$
\begin{aligned}
\log\left(\sum_{i=1}^{N} \exp(x_i)\right) &= y + \log\left(\sum_{i=1}^{N} \exp(x_i - y)\right) \\
&= -2500 + \log\left(\exp(-10) + \exp(0) + \exp(-15)\right) \\
&= -2500 + 0.00001984933 \\
&= -2499.99998015
\end{aligned}
$$

Thus, when calculating $L\alpha_j(t)$ in Equation 5.15, the final step requires application of the log-sum-exp trick to term $\sum_{i=1}^{N} \exp\left(\log(\alpha_i(t-1)) + \log(a_{i,j}) + log(b_j(o_t))\right)$, where each $x_i = \log(\alpha_i(t-1)) + \log(a_{i,j}) + log(b_j(o_t))$. The same approach is used when computing the backward probabilities in Equation 5.8. After computing the log values for the α and β, the next step is to compute the $\xi_{i,j}(t)$ and $\gamma_j(t)$ in log scale using the following equations:

$$\begin{aligned}
\log \xi_{i,j}(t) &= \log \left(\frac{\alpha_i(t) a_{i,j} b_j(o_{t+1}) \beta_j(t+1)}{\sum_{k=1}^{N} \alpha_k(T)} \right) \\
&= \log \alpha_i(t) + \log a_{ij} + \log b_j(o_{t+1}) + \log \beta_{t+1}(j) - \log \left(\sum_{k=1}^{N} \alpha_k(T) \right) \\
&= \log \alpha_i(t) + \log a_{ij} + \log b_j(o_{t+1}) + \log \beta_{t+1}(j) - \log \left(\sum_{k=1}^{N} e^{\log \alpha_k(T)} \right)
\end{aligned}$$

and

$$\begin{aligned}
\log \gamma_i(t) &= \log \left(\frac{\alpha_i(t) \beta_i(t)}{\sum_{k=1}^{N} \alpha_k(T)} \right) \\
&= \log \alpha_i(t) + \log \beta_i(t) - \log \left(\sum_{k=1}^{N} \alpha_k(t) \right) \\
&= \log \alpha_i(t) + \log \beta_i(t) - \log \left(\sum_{k=1}^{N} e^{\log \alpha_k(t)} \right)
\end{aligned}$$

As with $\alpha_i(t)$ and $\beta_i(t)$, the log-sum-trick is applied when calculating $\log \xi_{i,j}(t)$ and $\log \gamma_t(j)$. The exponential of each of the terms are then calculated to generate the values for $\xi_{i,j}(t)$ and $\gamma_t(j)$ that are used as part of the maximization-step to generate estimates for A, B, and π using the equations and process described in section 5.4.

5.7.2 Normalizing the forward and backward probabilities

An alternative approach to resolve the underflow issue is to normalize the forward and backward probabilities at each time-step. While normalizing $\alpha_i(t)$ and $\beta_i(t)$ results in different values, the end result when calculating $\zeta_{i,j}(t)$ and $\gamma_{i,j}(t)$ remains the same. As such, the normalization technique does not affect the maximize-step when estimating the parameter set $\lambda = (A, B, \pi)$.

Let $\bar{\alpha}_i(t)$ correspond to the normalized value of $\alpha_i(t)$ so that for $\hat{c}_\alpha(t) = \sum_{i=1}^{N} \alpha_i(t)$, $\bar{\alpha}_i(t) = \alpha_i(t)/\hat{c}_\alpha(t)$. Similarly, $\bar{\beta}_i(t)$ is introduced at the normalized value of $\beta_i(t)$ scaled by the coefficients $\hat{c}_\beta(t)$. When computing $\gamma_t(j)$ and $\xi_{i,j}(t)$, the usage of $\bar{\alpha}_i(t)$ and $\bar{\beta}_i(t)$ in place of $\alpha_i(t)$ and $\beta_i(t)$

does not affect the resulting values. This is demonstrated below:

$$\begin{aligned}
\frac{\bar{\alpha}_i(t)\bar{\beta}_i(t)}{\sum_{j=1}^{N}\bar{\alpha}_j(t)\bar{\beta}_j(t)} &= \frac{\alpha_i(t)/\hat{c}_\alpha(t)\beta_i(t)/\hat{c}_\beta(t)}{\sum_{j=1}^{N}\alpha_j(t)/\hat{c}_\alpha(t)\beta_j(t)/\hat{c}_\beta(t)} \\
&= \frac{\frac{1}{\hat{c_\alpha}(t)\hat{c_\beta}(t)}\alpha_i(t)\beta_i(t)}{\frac{1}{\hat{c_\alpha}(t)\hat{c_\beta}(t)}\sum_{j=1}^{N}\alpha_j(t)\beta_j(t)} \\
&= \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^{N}\alpha_j(t)\beta_j(t)} \\
&= \gamma_i(t)
\end{aligned} \tag{5.16}$$

and

$$\begin{aligned}
\frac{\bar{\alpha}_i(t)a_{i,j}\bar{\beta}_j(t+1)b_j(o_{t+1})}{\sum_{k=1}^{N}\sum_{w=1}^{N}\bar{\alpha}_k(t)a_{k,w}\bar{\beta}_w(t+1)b_w(o_{t+1})} &= \frac{\alpha_i(t)/\hat{c}_\alpha(t)a_{i,j}\beta_j(t+1)/\hat{c}_\beta(t)b_j(o_{t+1})}{\sum_{k=1}^{N}\sum_{w=1}^{N}\alpha_k(t)/\hat{c}_\alpha(t)a_{k,w}\beta_w(t+1)/\hat{c}_\beta(t)b_w(o_{t+1})} \\
&= \frac{\frac{1}{\hat{c_\alpha}(t)\hat{c_\beta}(t)}\alpha_i(t)a_{i,j}\beta_j(t+1)b_j(o_{t+1})}{\frac{1}{\hat{c_\alpha}(t)\hat{c_\beta}(t)}\sum_{k=1}^{N}\sum_{w=1}^{N}\alpha_k(t)a_{k,w}\beta_w(t+1)b_w(o_{t+1})} \\
&= \frac{\alpha_i(t)a_{i,j}\beta_j(t+1)b_j(o_{t+1})}{\sum_{k=1}^{N}\sum_{w=1}^{N}\alpha_k(t)a_{k,w}\beta_w(t+1)b_w(o_{t+1})} \\
&= \xi_{i,j}(t)
\end{aligned} \tag{5.17}$$

5.8 Psuedocode implementation

In this section, the pseudocode for the Baum-Welch algorithm is presented. In support of the Baum-Welch algorithm, the Forward and Backward algorithms are also presented using the normalization technique described in subsection 5.7.2. When training a hidden Markov model using the Baum-Welch algorithm it is assumed that a large set of R observations $\mathcal{O} = \{O_1, \ldots, O_R\}$ are available. While traditional pseudocode for the Baum-Welch algorithm is typically written in a generic form, the pseudocode provided here assumes the pseudocode will be implemented in the Python programming language using the NumPy library. As such, Python-specific functions like einsum are used to represent the summations in Equation 5.16, and Equation 5.17. Additionally, the pseudocode is written to indicate sections of code that benefit from utilizing the multiprocessing module or standard list comprehension to speed up calculations. List comprehension and multiprocessing.Pool.starmap() are interchangeable where ever a list comprehension statement is listed within an OpenPool loop.

First, pseudocode for the Baum-Welch algorithm is presented. The algorithm takes as its input a set of observations $\mathcal{O}$, an initial estimate for the HMM parameters A, B, and π, and the desired number of iterations. The output of the Baum-Welch algorithm is an updated estimate for A, B, and π. The first set of statement focus on calculating $\alpha^r(t)$, $\beta^r(t)$, $\gamma^r(t)$, and $\xi^r(t)$ for the r^{th} observation sequence. For observation O_r of length T, the dimension of each variable are as follows: $\alpha^r \in \mathcal{R}^{N \times T}$, $\beta^r \in \mathcal{R}^{N \times T}$, $\gamma^r \in \mathcal{R}^{N \times T}$, and $\xi^r \in \mathcal{R}^{N \times N \times T}$.

Algorithm 1 Baum-Welch algorithm

```
 1: function bw($\mathcal{O}, A, B, \pi$,I)
 2:     for $i = 0$ to $I$ do
 3:         OpenPool ($NumCores$) do
 4:             $\vec{P}$=[probObsState($O, B$) for $O \in \mathcal{O}$]
 5:             $\vec{\alpha}$=[calcAlpha($A, \pi, P$) for $P \in \vec{P}$]
 6:             $\vec{\beta}$=[calcBeta($A, P$) for $P \in \vec{P}$]
 7:             $\vec{\gamma}$=[calcGamma($\alpha, \beta$) for $\alpha, \beta \in \vec{\alpha}, \vec{\beta}$]
 8:             $\vec{\xi}$=[calcXi($A, \alpha, \beta, P$) for $\alpha, \beta, P \in \vec{\alpha}, \vec{\beta}, \vec{P}$ ]
 9:         ClosePool
10:         $A_{num} = [\xi$.sum(axis=2) for $\xi \in \vec{\xi}$].sum(axis=0)
11:         $A = (A_{num}^T / A_{num}$.sum(axis=1)$)^T$
12:         $B$=[updateEmission($i, \mathcal{O}, \vec{\gamma}$) for $i \in \{1, \ldots, N\}$]
13:         $\pi = [\gamma(0)$ for $\gamma \in \vec{\gamma}]$.sum(axis=0)
14:         $\pi = \pi / R$
15:     end for
16:     return $A, B, \pi$
17: end function
```

In computing each α^r, β^r, and ξ^r it is beneficial to have ready the probability of an observation conditioned a given state, i.e. $b_i(o_t) = P(o_t|s_i)$. As such, these probabilities are precomputed in line 4 of Algorithm 1 for all states and stored in a matrix $P^r \in \mathcal{R}^{N \times T}$, where each $p_{i,t} = P(o_t|s_i)$ is calculated using the function probObsState(O, B). The general structure of the function call probObsState(O, B) is provided below. The update function updateEmission$(i, \mathcal{O}, \vec{\gamma})$ for the emission distributions remains unique to the distribution type, as such, it is not specified here.

Algorithm 2 Calculate State Dependent Probability of Observations

```
function probObsState($(O, B)$)
    for $t = 1$ to $T$ do
        for $i = 1$ to $N$ do
            $P[i, t] = b_i(o_t)$
        end for
    end for
    return $P$
end function
```

When implementing Equation 5.10 and Equation 5.11 to update π and A, minor adjustments to the equations are required to account for training over the complete set of R observations. The adjustment essential computes a weighted average over the R samples. Accordingly,

$$\hat{\pi}_i = \frac{\sum_{r=1}^{R} \gamma_i^r(1)}{R} \tag{5.18}$$

and

$$\hat{a}_{i,j} = \frac{\sum_{r=1}^{R} \sum_{t=1}^{T-1} \xi_{i,j}^r(t)}{\sum_{r=1}^{R} \sum_{t=1}^{T-1} \sum_{k=1}^{N} \xi_{i,k}^r(t)}, \tag{5.19}$$

where $\gamma_i^r(t)$ and $\xi_i^r(t)$ correspond to the R^{th} observation sequence. However, because the denominator in Equation 5.19 serves as normalization to ensure a proper distribution, its calculation is not necessary. Instead, the normalization can be performed after calculating all the numerator values; these steps occur on lines 10-11 in Algorithm 1. Similarly, calculation of the numerator values in Algorithm 3-Algorithm 6 are denoted by the subscript num.

5.9 Conclusion

This chapter was aimed at introducing the concept of the hidden Markov model (HMM), a common approach that is used to model stochastic systems. Different sections of this chapter dis-

Algorithm 3 Forward algorithm

```
function CALCALPHA($A$,$\pi$,$P$)
    $\alpha_{num}$ =EINSUM(’$i, i \rightarrow i$’,$\pi$,$P[:, 0]$)
    $\alpha[:, 0] = \alpha_{num}/\alpha_{num}$.sum()
    for $i = 1$ to $T$ do
        $\alpha_{num}$ =EINSUM(’$j, ji, i \rightarrow i$’,$\alpha[:, t-1]$,$A$,$P[:, t]$)
        $\alpha[:, t] = \alpha_{num}/\alpha_{num}$.sum()
    end for
    return $\alpha$
end function
```

Algorithm 4 Backward algorithm

```
function CALCBETA($A$,$P$)
    beta[:,0]=1/N
    P=FLIPLR(P)
    for $t = 1$ to $T$ do
        $\beta_{num}$ =EINSUM(’$j, ij, j \rightarrow i$’,$\beta[:, tb-1]$,$A$,$P[:, tb-1]$)
        $\beta[:, tb] = \beta_{num}/\beta_{num}$.sum()
    end for
    return FLIPLR($\beta$)
end function
```

Algorithm 5 Gamma Calculation

```
function CALCGAMMA($\alpha$,$\beta$)
    $\gamma_{num}$=EINSUM(’$it, it \rightarrow it$’,$\alpha$,$\beta$)
    $\gamma$=EINSUM(’$it, t \rightarrow it$’,$\gamma_{num}$,1/$\gamma_{num}$.sum(axis=0))
    return $\gamma$
end function
```

Algorithm 6 Xi Calculation

```
function CALCXI($A$,$\alpha$,$\beta$,$P$)
    $\xi_{num}$=EINSUM(’$it, ij, jt, jt \rightarrow ijt$’,$\alpha[:, 0:-1]$,$A$,$\beta[:, 1:]$,$P[:, 1:]$)
    $\xi$=EINSUM(’$ijt, t \rightarrow ijt$’, $\xi_{num}$,1/$\xi_{num}$.sum(axis=(0,1)))
    return $\xi$
end function
```

cussed the main HMM algorithms, including the Forward algorithm, the Backward algorithm, the Baum-Welch algorithm, and the Viterbi algorithm. Furthermore, it was explained that while the mathematical formulation of these algorithms is obvious, there exist some challenges (i.e., underflow) with the computational implementation. Finally, the two introduced approaches, including the log-sum-exponential trick and the log space formulation were explained as solutions to tackle the computation limitations.

CHAPTER 6
PERFORMANCE COMPARISON BETWEEN MULTIVARIATE POISSON HMM AND BASIC HMM

In chapter 2, machine learning techniques were applied and evaluated on their ability to predict and model halting rates of undergraduate students when considering academic grade measures. One of the major findings of the chapter was that predictions based on grade-counts proved to be more informative than predictions using GPA measures. Additionally, another study in the chapter also indicated that existing machine learning models struggle to account for deviations in academic performance on a semester-to-semester basis. As such, models appeared to significantly over-classify regularly high-performing students with one poor-performing semester as more likely to halt. To overcome this shortcoming, chapter 4 proposes a basic categorical hidden Markov model to model and link academic performance to final academic outcomes. While able to provide insights into halting, the chapter highlighted that a large number of parameters is required to define the model, which would be prohibitive when applying the machine learning technique at smaller institutions with less available training data (in this case student records).

In order to develop a more concise HMM that accurately reflects the academic performance levels of students, this chapter proposes the development and application of a Multivariate Poisson HMM (MPHMM) leveraging the computation framework described in chapter 5. In this chapter, after developing the necessary equations for the implementation of a Multivariate Poisson HMM, the benefits of the proposed HMM are compared to the basic categorical HMM using simulation. The MPHMM is shown to outperform basic HMMs when working with Poisson generated state-dependent observations.

6.1 Hidden Markov Model Variants

HMMs can be used to model various classes of problems according to their underlying data structure and emission types. Here, underlying data structures and emission types refer to observation type (e.g., discrete or continuous), observation dimension (e.g., univariate or multivariate), and associated distribution (e.g., Gaussian, categorical, Poisson). For example, an HMM can generate observations from a univariate random variable coming from a finite categorical distribution. With this, the categorical distribution is representative of a discrete probability distribution with k items in the sample space [97] defined by the probability mass function (PMF) $f(X = i) = p_i$, where p_i is the probability of observing event i. The use of univariate categorical emissions as part of hidden Markov modeling is quite common and finds appropriate support through numerous programming libraries (e.g Python's sklearn.hmm.MultinomialHMM); accordingly, chapter 4 used existing libraries when modeling academic performance levels. One major shortcoming of existing categorical HMM software implementations is that they typically do not allow for explicit modeling of multivariate categorical emissions. As another common HMM emission model makes use of Gaussian distributions. In this case, the corresponding HMM observations are continuous values where the likelihood of an observation (x) is computed using the probability density function of the Gaussian distribution

$$f(x) = \frac{1}{\pi\sqrt{2\pi}}exp(-\frac{(x-\mu)^2}{2\sigma^2})$$

where each hidden state is associated with its own μ and σ. Like categorical HMM, Gaussian HMM finds broad support through software implementations (e.g Python's sklearn.hmm.GaussianHMM), which in fact allow for both univariate and multivariate emission models.

Given the various emission models, it is often important to select the correct emission type corresponding to the problem under consideration, doing so allows for faster learning of model parameters and improved accuracy when estimating hidden states. To provide evidence for these assertions and to support the subsequent development and utilization of a Poisson HMM a brief case study is provided here. In the case study, a set of random values, x_i, are generated from a Poisson distribution with a known distribution parameter (e.g., μ=2). The generated data is then fit to two different distribution models: a Poisson distribution and a categorical distribution. While it should be expected that both the Poisson and categorical models can fit the generated data, the benefits of each are not explicitly clear. The aim, therefore, is to investigate which of these two distributions provides a better fit for a given fixed-size training data set. Here, the fit is measured by the chi-square distances between the actual Poisson distribution that generated the data and fit distributions.

Fitting a Poisson distribution to a data set requires estimating the μ parameter of the distribution. For a given sample of N independently generated observation data x_i for $i = 1, ..., N$, this process makes use of the maximum likelihood equation. This process begins by asserting the underlying probability distribution

$$p(x_i) = e^{-\mu} \frac{\mu^{x_i}}{x_i!}, \qquad 1 \leq x_i \leq N \tag{6.1}$$

and defining the likelihood function based on the set of observation data

$$L(\mu | x_1, \ldots, x_N) = \prod_{i=1}^{N} (\frac{e^{-\mu} \; \mu^{x_i}}{x_i!}) = e^{-\mu N} \frac{\mu^{\sum_{i=1}^{N} x_i}}{\prod\limits_{i=1}^{N} x_i}.$$

Because all x_i are known to be non-negative (with at least one value being positive for a sufficiently large set), the optimal parameter estimate $\hat{\mu} > 0$ can be solved by treating the problem as unconstrained optimization. Taking the natural log of the likelihood function

$$lnL(\mu) = -\mu N + \sum_{i=1}^{N} x_i ln(\mu) - ln(\prod_{i=1}^{N} x_i) \tag{6.2}$$

and setting the derivative to 0

$$\frac{dL}{d\mu} = -N + \sum_{i=1}^{N} x_i \frac{1}{\mu} = 0$$

yields the following solution:

$$\hat{\mu} = \frac{\sum_{i=1}^{N} x_i}{N} \tag{6.3}$$

Accordingly, based on the maximum likelihood approach, the μ parameter for the estimated Poisson distribution is obtained by taking the average of all generated observations.

In the case of estimating the parameters $p_1, \ldots, p_N$ for a categorical distribution of size k, the process follows that provided in section 5.2. The result is a frequentist solution where probabilities for each emission are in proportion to the relative frequency in which they are observed in the training data set. In other words, given a set of N training observations, the optimal estimate for the probability of each observation x_i is given by

$$\hat{p}(x_i) = \frac{\text{\# of } x_i \text{ in the training set}}{N} \tag{6.4}$$

Fit comparison between the estimated distributions (i.e. categorical and Poisson) with the true Poisson distribution is not without some complications. The original Poisson distribution has infinite support, while categorical distributions are defined over finite sample spaces. As such, there is no standard or equitable means for calculating the distances between the estimated distributions and the original Poisson distribution. To overcome this issue, all distributions are represented as truncated finite distributions based on the largest observed value x_{max} in the training data set, where $x_{max} = \max\{x_1, \ldots, x_N\}$. To represent the generating Poisson distribution as a finite truncated Poisson distribution, the first step is to calculate the probability of each possible observation over the range $[0, x_{max}]$, which is computed using the μ parameter and the probability mass function in Equation 6.1. The intermediate result is a set of probabilities $\tilde{P}(\mu) = \{\tilde{p}(0|\mu), ..., \tilde{p}(x_{max}|\mu)\}$. To form a proper distribution, the probabilities must be rescaled so that they sum to 1. Accordingly, let $P(\mu) = \{p(0|\mu), ..., p(x_{max}|\mu)\}$ represent the corresponding the rescaled truncated distribution where each $p(i|\mu) = \tilde{p}(i|\mu)/\sum_{j=1}^{N} \tilde{p}(j|\mu)$. Using the same strategy, the empirically estimated Poisson distribution is represented as a categorical distribution with probabilities $P(\hat{\mu}) = \{p(0|\hat{\mu}), ..., p(x_{max}|\hat{\mu})\}$, where $\hat{\mu}$ corresponds to the optimal estimate of μ.

The distances between the original transformed truncated Poisson distribution $P(\mu) = \{p(0|\mu), ..., p(x_{max}|\mu)\}$ and one of the estimated distribution P^{est} is calculated using the chi-square distance (C.D.) given by

$$C.D. = \sum_{i=0}^{x_{max}} \frac{(p(i|\mu) - p^{est}(i))^2}{p(i|\mu)}. \tag{6.5}$$

In Equation 6.5, the values $p^{est}(i)$ are taken from $P(\hat{\mu})$ of the Poisson estimate or correspond to $\hat{p}(x_i)$ of the empirical categorical distribution calculated using Equation 6.4.

Following the generation of sample training data sets based on $\mu =$ 2, 5, and 9, the chi-square distances between the actual and estimated distributions are calculated according to Equation 6.5. Figure 6.1 plots the chi-square distances (in log scale) between the transformed (true) Poisson distribution with the estimated Poisson distribution and the estimated categorical distribution. As indicated by the results in the figure, for all μ values and sample sizes, the chi-square distance between the estimated Poisson distribution and the actual Poisson distribution is at least an order magnitude smaller than the empirically estimated categorical distribution. Furthermore, when fitting a Poisson distribution, a categorical distribution requires almost 100x more data to achieve an equivalent fit as an estimated Poisson model.

To better understand the benefits of using a Poisson model, Figure 6.2 provides the $10^{th} - 90^{th}$ percentile box plots for the chi-square distances (in log scale) between a generating Poisson distribution ($\mu = 5$) with the estimated Poisson distribution and estimated categorical distribution. For each sample size, 1000 different data sets are generated to compare the mean, median, and variance of the computed distances. From the figure, it is observable that for each training sample size the Poisson distribution model outperforms the categorical distribution model in terms of the mean (green points) and median (orange lines) of the chi-square distances. On the other hand, the distances for the categorical distributions have smaller variances compared to distances for the

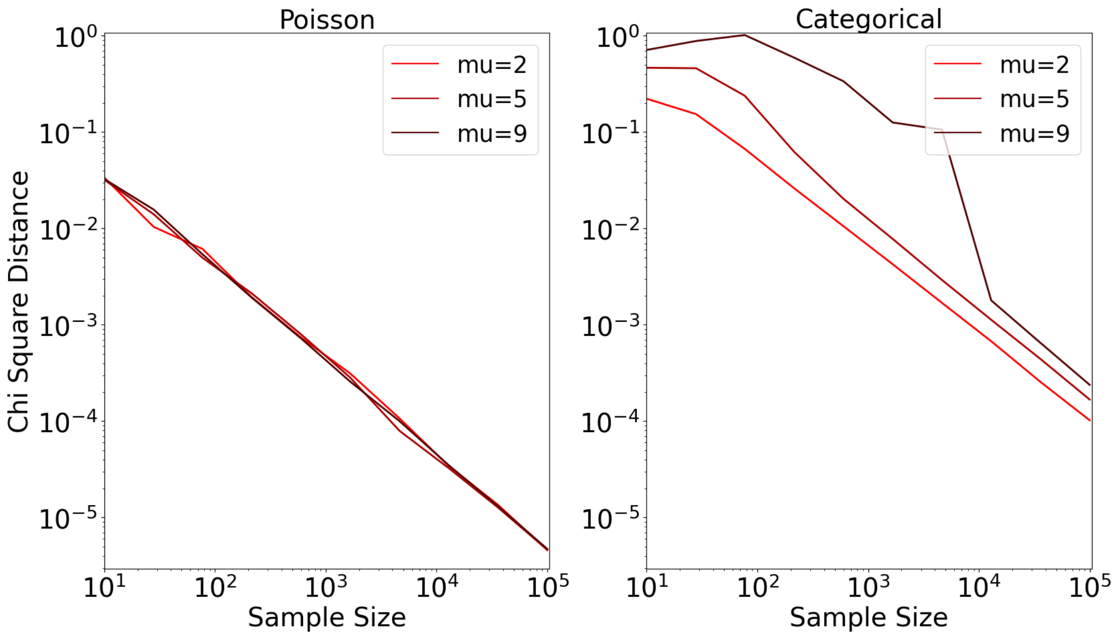


Figure 6.1: The chi-square distance (in log scale) between the converted main Poisson distribution with the converted estimated Poisson distribution and estimated categorical distribution

Poisson distributions, as indicated by the smaller inter-quartile ranges.

Based on the results presented in Figure 6.1 and Figure 6.2, utilizing a Poisson distribution model for Poisson data is significantly preferred over a categorical distribution model as it is able to achieve the same level of accuracy using a fraction of the training data. So although there exist libraries that provide for HMM implementations using categorical distributions, there is a real benefit to developing software for a Poisson HMM.

Findings: When data points are generated from a Poisson distribution, fewer samples are needed when fitting a Poisson distribution compared to a categorical distribution

6.2 Multivariate Poisson Hidden Markov Model

In chapter 4, a basic HMM using categorical emissions is used to model the dynamics of a student's academic performance level over their academic career. As part of the modeling process, student grade-counts are mapped to possible observation tuples. The problem with this approach is that the number of possible unique grade-count observations is large, and as such, the number of parameters

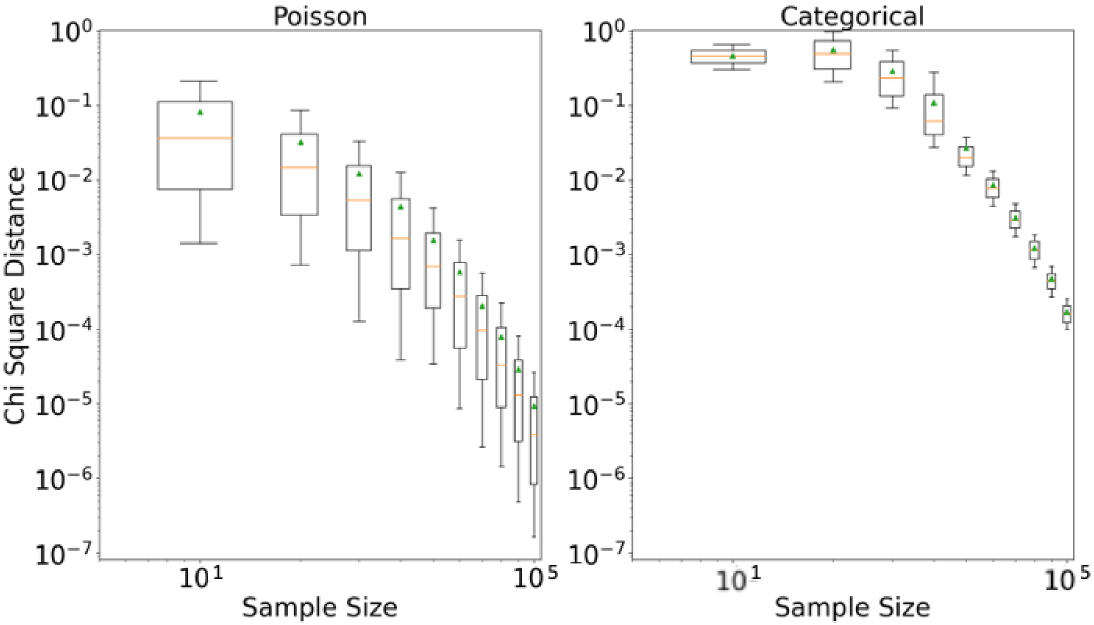


Figure 6.2: Box plot for the chi-square distance (in log scale) between the converted main Poisson distribution with the converted estimated Poisson distribution and estimated categorical distribution

that are required to be estimated grows quickly. To tackle the dimensionality problem caused by the large number of grade combinations, an upper limit on the tuple elements representing the grade-counts is proposed so that the maximum number for A's, B's, and C's put in a tuple is capped at 3. Furthermore, instead of storing the number of D/F and W grades, these values were converted into binary variables, with 1 indicating that at least one such grade exists in the student's semester grades. Through this compression process, the total number of possible observations decreased from 687 to 216. However, even with the compression, the number of parameters required to model academic performance levels is high and likely requires large data sets in order to train.

In this chapter, a combination of two different approaches is used to tackle the HMM dimensionality problem. First, instead of converting grades tuples to unique observations as part of categorical distributions, the grade tuples are represented using a multivariate distribution in which a multivariate observation output corresponds to a tuple composed of grade-counts (e.g. number of A's, number of B's). A schematic of such an HMM is illustrated in Figure 6.3 where each ob-

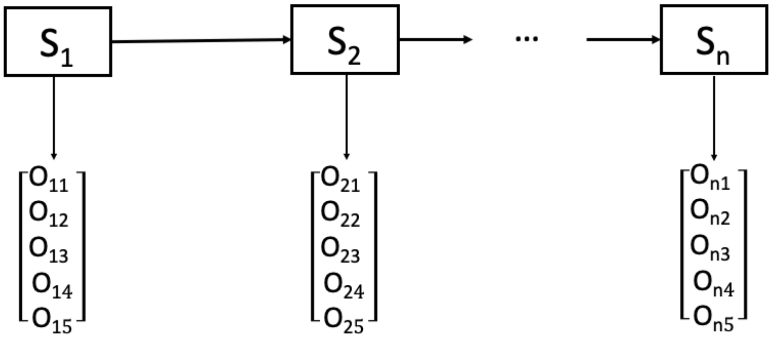

S_1
S_2
...
S_n
O_{11} O_{12} O_{13} O_{14} O_{15}
O_{21} O_{22} O_{23} O_{24} O_{25}
O_{n1} O_{n2} O_{n3} O_{n4} O_{n5}

Figure 6.3: Representation of a multivariate Hidden Markov Model

servation is a five-dimensional variable. In the construction of a multivariate HMM, it is assumed that the elements of each observation are mutually independent; note, however, the distribution parameters over the multivariate are linked through the state.

Representing grade-counts as a multivariate does not reduce the number of parameters in itself, especially when still using categorical emissions. However, if each element of the multivariate is represented by a Poisson distribution, then each distribution can be described by a single parameter. As demonstrated in section 6.1, if the underlying data does in fact come from a Poisson distribution, then utilizing a Poisson emission model will require significantly fewer data to learn the model parameters than the equivalent categorical distribution.

As evidence to support the usage of a Poisson emission for grade-counts as part of a multivariate emission model, the empirical distribution of the number of A's students earn in a given semester is shown in Figure 6.4. The empirical distribution closely matches the shape of the Poisson distribution. Accordingly, a chi-square hypothesis test is conducted to check the compatibility of the Poisson distribution to students' grade counts (i.e., the number of A's). The null hypothesis of this test states that the number of A's that students earn each semester follows a Poisson distribution with $\mu = 1.623$. On the other hand, the alternative hypothesis states that the number of A's does not have the Poisson distribution. In order to perform the chi-square hypothesis test, the ex-

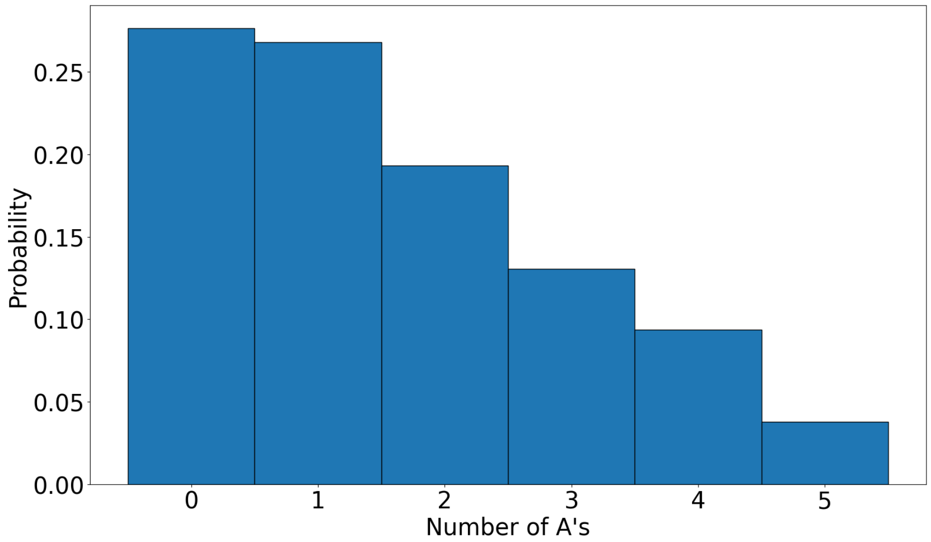


Figure 6.4: Probability density function for the Poisson distribution fitted on number of A grades

pected and observed counts are capped at 5, so that all events greater than or equal to 5 are binned together. The resulting p-value=$0.99987651 > .01$ ($N = 334177$) indicates that the null hypothesis cannot be rejected. Hypothesis testing, when applied to the number of B's, C's, D/F's, and W's, also failed to reject the corresponding null hypothesis that the grade-counts follow a Poisson distribution. Therefore, it is likely that an HMM that utilizes Poisson emissions could improve the results, however, such an assertion must be tested and validated on real student data for each state-dependent emission (see chapter 7).

In order to model semester grade-count using a multivariate Poisson distribution, the next step is to embed the Poisson probability density function into the HMM software implementation discussed in chapter 5. The primary difference between a categorical HMM and the Poisson HMM is in the representation of the emission distribution used when computing the likelihood of particular observations being generated from each of the hidden states. Specifically, the function PROBOBSSTATE(O, B) described in the pseudocode is adjusted. Recalling that O represents an observation sequence, and B is a set of state-dependent emission distributions, the probability of a state i generating an observation o_t at time-step t is denoted by $b_i(o_t)$. Now because o_t is assumed to be an independent multivariate, the probability $b_i(o_t)$ is given by the product

$$b_i(o_t) = \prod_{k=1}^{K} b_{i,k}(o_{t,k}) \tag{6.6}$$

where $b_{i,k}(o_{t,k})$ represents the probability distribution of the emission's k^{th} feature. For a Poisson multivariate, it follows that

$$b_{i,k}(o_{t,k}) = e^{-\mu_{i,k}} \times \frac{\mu_i^{o_{t,k}}}{o_{t,k}!} \tag{6.7}$$

where $\mu_{i,k}$ is the Poisson parameter for the k^{th} feature of state i's emission.

Considering this replacement, it is necessary to use the Poisson probability calculations in both the Baum-Welch and Viterbi algorithms separately. In other words, $b_i(o_t)$ in Algorithm 2 should be replaced with Equation 6.6 and Equation 6.7 in the corresponding equations when computing α (Equation 5.6 and Equation 5.7) and β (Equation 5.8), γ (E-step), and ξ (E-step) in the Baum-Welch algorithm.

Estimating the parameters of the emission distributions as part M-step requires adjusting the update equations based on a Poisson distribution. Following a similar structure of the likelihood function in Equation 6.2 and Equation 6.3, along with the generalized update equation for the parameters of a state-dependent emission distribution Equation 5.12 the update for the Poisson parameter $\mu_{i,k}$ for the k^{th} feature of state i's emission is

$$\mu_{i,k} = \frac{\sum_{t=1}^{T} \alpha_i(t)\beta_i(t)o_{t,k}}{\sum_{t=1}^{T} \alpha_i(t)\beta_i(t)} \tag{6.8}$$

In updating each $\mu_{i,k}$ in Equation 6.8, it is worth noting that following the pseudocode in the previous chapters, all these equations are computed using the normalized α and β to prevent underflow.

6.3 Viterbi algorithm performance comparison between basic HMM and MPHMM

As shown earlier in this chapter, it is beneficial to select a probability distribution that matches the underlying data set in order to minimize the required size of the training data. The implication of such a result is that when constructing the HMMs, matching the emission distributions to the data will improve performance when learning parameters as part of the Baum-Welch algorithm. In this section, the benefits of using a Poisson emission distribution are explored in the context of the Viterbi algorithm. The corresponding research question is as follows:

Research question: Is there any performance differences when estimating hidden states between a multivariate Poisson HMM and a basic categorical HMM?

In order to address this research question, a multivariate Poisson HMM and a basic categorical HMM will be used to estimate the hidden state of a system based on a set of observations. The HMM that is more accurately able to identify the true state of the system is the better model. To perform this experiment synthetic test data, which includes matched state sequences and observation sequences, must be generated. The first step is to generate state sequences based on a common transition matrix A and initial state distribution π. For this study, the following parameters are utilized (with values represented as percentages) when generating state sequences.

$$\text{A} = \begin{bmatrix} 81.81 & 17.10 & 1.02 & 0.07 \\ 7.90 & 78.43 & 13.52 & 0.15 \\ 0.74 & 8.54 & 84.44 & 6.28 \\ 0.34 & 1.00 & 10.32 & 88.34 \end{bmatrix}$$

$$\pi = \begin{bmatrix} 20.14 & 19.07 & 33.50 & 27.29 \end{bmatrix}$$

Next state-dependent observations for each of the state sequences are generated. Here, the observations are assumed to be generated from a five-dimensional multivariate Poisson distribution with the following Poisson parameters:

$$\text{States Poisson Parameters } (\mu) = \begin{bmatrix} 0.398 & 0.837 & 0.905 & 1.036 & 0.316 \\ 0.835 & 1.360 & 0.859 & 0.326 & 0.156 \\ 1.570 & 1.430 & 0.463 & 0.101 & 0.083 \\ 2.840 & 0.766 & 0.095 & 0.013 & 0.037 \end{bmatrix}$$

In the matrix, rows and columns correspond to states and observation vector dimensions, respectively. For example, if the system is in state 1, the first, second, third, fourth, fifth elements of the observation are generated from a Poisson distribution with the mean of 0.398, 0.837, 0.905, 1.036, and 0.316, respectively.

A MPHMM (based on the proposed adjustments in section 6.2) is able to interpret the generated observations directly when performing the state-estimation step, however, the categorical HMM assumes a different observation structure. Using the compression procedure described in chapter 4 every combination of the Poisson multivariate is mapped to a single observation, which when taken together forms a categorical distribution with 216 possible univariate observations (again, these map to the same observations in chapter 4). The probabilities for each possible observation are derived from the original Poisson distributions.

The above procedure is repeated to generate eight different data sets, each of them containing 5000 matched to state and observation sequences; and the length of sequences for these eight data sets is 3, 4, 5, 6, 7, 8, 9, and 10. For both the basic categorical HMM and MPHMM, the Viterbi algorithm is applied to the observation sequence to estimate the hidden states. The estimated hidden states are compared to the actual (generated) hidden states to compute the estimation accuracy for both models.

Two different approaches are used to compute the performance of the Viterbi algorithm for the basic HMM and MPHMM. In the first approach, after calculating the percentage of incorrect estimates or *misestimates* for each sequence, the average of the percentages over all the 5000 se-

Table 6.1: Computing estimation accuracy for the Viterbi algorithm using the mean misestimation percentage method

Sample	Actual states	Estimated states	# Misestimations	% Misestimations
1	2,2,2,**3**,3	2,2,2,**2**,3	1	20%
2	1,1,2,2,2	1,1,2,2,2	0	0%
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
5000	**1**,2,2,**1**,2	**2**,2,2,**2**,2	2	40%

Table 6.2: Computing estimation accuracy for the Viterbi algorithm using the mean mis-estimation percentage method

Sample	Actual states	Estimated states	Sum abs. estimations err.	Mean abs. estimations err.
1	1,1,1,**3**,**1**	1,1,1,**1**,**2**	|3-1|+|1-2|=3	60%
2	1,1,2,2,2	1,1,2,2,2	0	0%
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
5000	**1**,2,2,**1**,**2**	**2**,2,2,**2**,2	|1-2|+|1-2|=2	40%

quences is computed. Then, 1-mean(% misestimates) is reported as the accuracy of HMM model when using the Viterbi algorithm. Table 6.1 provides an example of how this approach works when computing estimation accuracy for the Viterbi algorithm when the length of the state and observation sequences are 5. For example, for sample 1, only one state-estimation is incorrect, indicated by the bold number. Since the state sequence length is 5, then the misestimation percentage for this sample is 20%. Figure 6.5 shows the Accuracy (1-%misestimation) of the Viterbi algorithm for both the basic HMM and MPHMM for sequences with different lengths. As is illustrated in the figure, when increasing the length of sequences, accuracy when applying the Viterbi algorithm for both basic HMM and MPHMM increases. Furthermore, for all data sets, the MPHMM outperforms the basic HMM when estimating the true hidden state; and while the average difference in accuracy is limited to between 3%-4%, the difference is consistent.

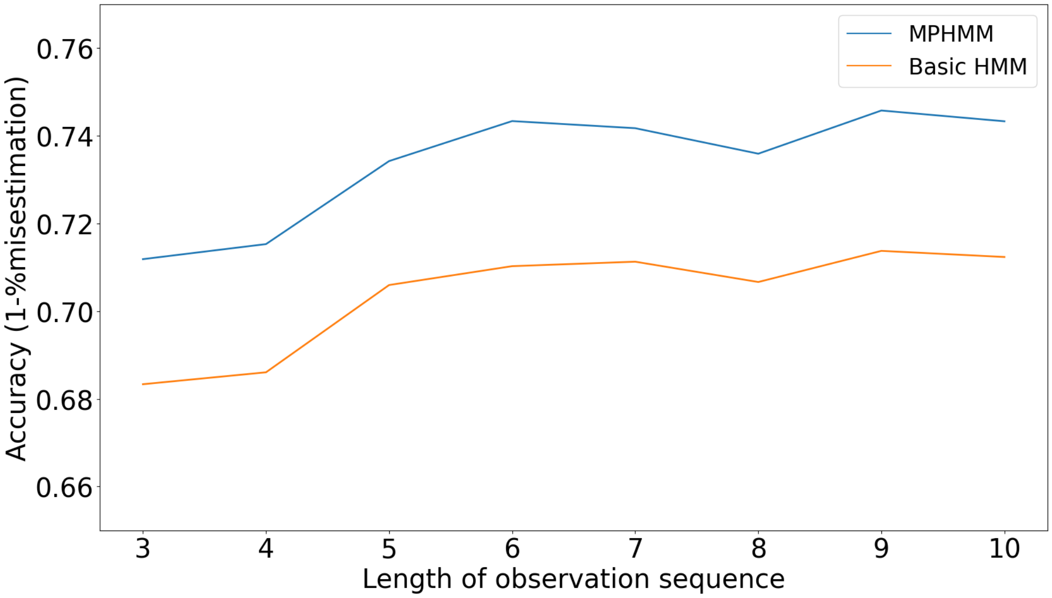


Figure 6.5: Accuracy (1-%misestimation) of the Viterbi algorithm for both the basic HMM and MPHMM for sequences with different lengths

As an alternative approach to assess accuracy, the mean absolute error in estimates is considered. Table 6.2 provides an example application of the mean absolute error when computing the estimation accuracy of the competing HMMs. For each state sequence, after calculating the sum of absolute error, the obtained value is divided by the length of the state sequence. Then, an average is taken over all mean absolute errors to compute the mean absolute error over all the samples. By subtracting the value from 1, the estimation accuracy is computed for each HMM. For example, for sample one, while the fourth actual state is 3, the estimated state according to the Viterbi algorithm is 1. Thus, the absolute error for this state estimate is |3-1|=2. For the fifth state, the absolute error is 1, therefore, the sum absolute error for the first state sequence is calculated to be 3. After dividing the sum absolute error by the length of the sequence (5 in this example), the mean absolute error for this sample is 60%. Figure 6.6 reports the accuracy (1-%abs. error) for both the basic HMM and MPHMM when applying the Viterbi algorithm for sequences with different lengths. Similar to the previous accuracy evaluation, the MPHMM model performs slightly better than the basic HMM in estimating hidden states for all state sequences with different lengths. Furthermore,

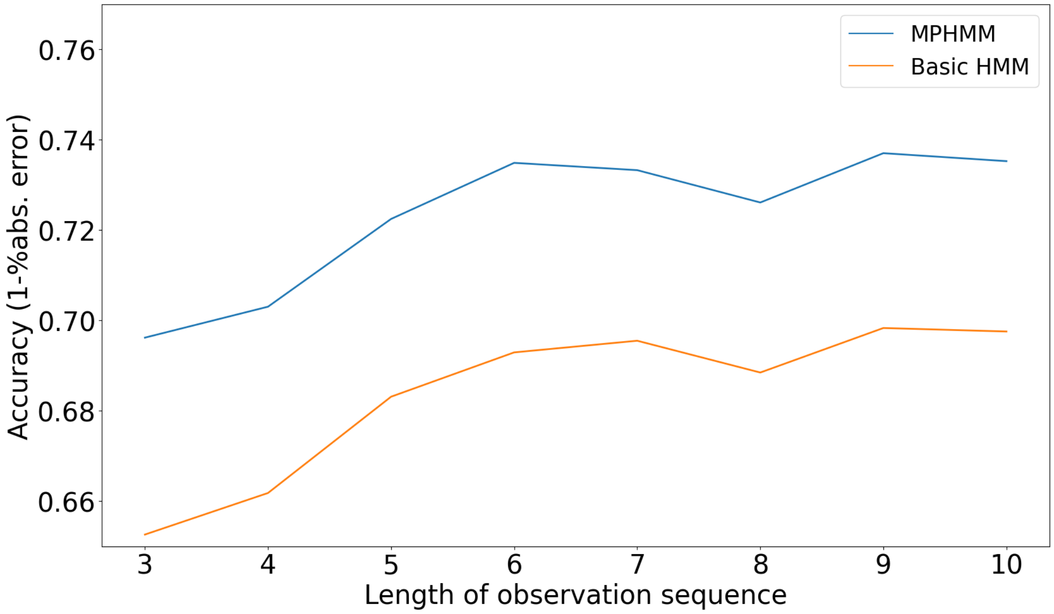


Figure 6.6: Accuracy (1-%abs. error) of the Viterbi algorithm for both the basic HMM and MPHMM for sequences with different lengths

the same pattern is seen in this figure, whereby when increasing the length of the sequences, the accuracy of the models and Viterbi algorithm will be improved for both applied HMMs.

Findings: MPHMM outperforms the basic HMM in terms of state estimation accuracy when observations are generated from a Poisson distribution.

As a final note, while the performance of the MPHMM in estimating the hidden state is only marginally better than the basic HMM, it is likely that an improved parameter estimation step will have a greater impact on the accuracy of the decoding step. That is to say, improving the parameter estimation through the usage of a MPHMM allows for better performance in the decoding step.

6.4 Conclusion

This chapter was aimed at comparing the performance of the learning and estimation steps associated with the basic HMM and MPHMM. Using simulation, it was shown that when data points are generated from a Poisson distribution, fewer samples are needed when fitting a Poisson distribution compared to when a categorical distribution is fitted. In this chapter, after introducing the Multivariate Poisson Hidden Markov model, its implementation was discussed in the context of the pseudocode provided in chapter 5. Then, the Viterbi algorithms of the basic HMM and MPHMM were applied to the observation sequences which were generated from Poisson distributions. The two different estimation accuracy comparisons between the basic HMM and MPHMM indicated that when observation sequences are generated from a Poisson distribution, the MPHMM outperforms the basic HMM in terms of hidden state estimation accuracy. In the next chapter, the performance of these two HMMs is compared when applied to the UCF student records data.

CHAPTER 7
APPLICATION OF MULTIVARIATE POISSON HIDDEN MARKOV MODELS IN EDUCATIONAL DATA MINING

In the previous chapter, the performance of the basic Hidden Markov Model (HMM) and the Multivariate Poisson Hidden Markov Model (MPHMM) were compared using simulations. In this section, the goal is to apply the proposed MPHHM to the UCF data and answer the following research question:

Research question: Is there any differences between the obtained results from the multivariate Poisson HMM and the basic HMM using the UCF data?

Similar to the basic HMM in chapter 4, in this section, the developed MPHMM takes the sequence of students' grade tuple as inputs to estimate students' academic-performance levels. In the basic HMM, each unique grade tuple was converted to an integer to accommodate a categorical emission matrix; the MPHMM allows using the original observation vectors (the grade tuples) as it can take multidimensional inputs. After applying the Baum-Welch algorithm, the following estimations are obtained for both Basic HMM and MPHMM parameters:

$$A^{(\text{Basic HMM})} = \begin{bmatrix} 86.11 & 10.93 & 2.44 & 0.52 \\ 8.50 & 80.00 & 11.41 & 0.09 \\ 0.29 & 7.58 & 84.90 & 7.23 \\ 0.02 & 0.02 & 5.64 & 94.32 \end{bmatrix}, A^{(\text{MPHMM})} = \begin{bmatrix} 70.61 & 25.52 & 2.94 & 0.93 \\ 9.65 & 78.79 & 11.55 & 0.01 \\ 0.08 & 8.51 & 84.27 & 7.14 \\ 0.10 & 0.00 & 7.46 & 92.44 \end{bmatrix}$$

$$EV(Grade)^{\text{(Basic HMM)}} = \begin{bmatrix} 0.306 & 0.661 & 0.861 & 0.606 & 0.290 \\ 0.783 & 1.470 & 0.907 & 0.211 & 0.097 \\ 1.765 & 1.421 & 0.302 & 0.038 & 0.055 \\ 2.554 & 0.459 & 0.023 & 0.002 & 0.020 \end{bmatrix}$$

$$\mu^{\text{(MPHMM)}} = \begin{bmatrix} 0.176 & 0.422 & 0.732 & 1.523 & 0.613 \\ 0.724 & 1.424 & 0.981 & 0.289 & 0.123 \\ 1.813 & 1.494 & 0.294 & 0.033 & 0.055 \\ 3.254 & 0.440 & 0.021 & 0.002 & 0.019 \end{bmatrix}$$

$$\pi^{\text{(Basic HMM)}} = \begin{bmatrix} 12.99 & 36.74 & 35.97 & 14.30 \end{bmatrix}, \pi^{\text{(MPHMM)}} = \begin{bmatrix} 7.78 & 40.20 & 38.29 & 13.73 \end{bmatrix}$$

As in the basic HMM discussed in Chapter 4, rows 1 through 4 of the matrix $A^{\text{(MPHMM)}}$ correspond to academic-performance levels 1 to 4 (percentage scale). Based on the estimated transition matrix $A^{\text{(MPHMM)}}$ and the main diagonal, most of the students maintain their academic performance levels from one semester to the next. Furthermore, based on these numbers, students in academic-performance level 4 are more likely to maintain their academic-performance levels when compared to other students. These results, in general, are similar to those obtained from the basic HMM, however, there are some differences between the transition matrices for the basic HMM and MPHMM for students with academic-performance level 1. The MPHMM suggests a lower chance for students in this level to maintain their performance from one semester to the next (70.61%) when compared to the basic HMM (86.11%). While the transition probability from academic-performance level 1 to academic-performance levels 3 and 4 are similar between the two models, it is more likely for students in academic-performance level 1 to switch to the academic-performance level 2 based on the MPHMM when compared to the basic HMM (25.52%>10.93%). The differences between the basic HMM and the MPHMM on transition probabilities for other academic-performance levels are negligible (within 2%).

In matrix $\mu^{\text{(MPHMM)}}$, rows 1 through 4 correspond to academic-performance levels 1 to 4, and columns 1 to 5 provide the expected value for the number of courses with an A, B, C, D/F, and W grade each semester. In the basic HMM applied in Chapter 4, since the emission matrix had 216 columns, the expected values of different grades $EV(Grade)^{\text{(Basic HMM)}}$ were computed for each state. Comparing matrix $\mu^{\text{(MPHMM)}}$ with matrix $EV(Grade)^{\text{(Basic HMM)}}$, we find that the Poisson parameters for the D/F's and W's grades are higher than the expected values for those grades in the $EV(Grade)^{\text{(Basic HMM)}}$. This is because in the basic HMM, the number of D/F and W grades were transferred to binary variables. Furthermore, since there is no constraint on the number of grades in the MPHMM, for academic-performance level 4, the mean number of A grades is higher in $\mu^{\text{(MPHMM)}}$ (3.254) compared to that of $EV(Grade)^{\text{(Basic HMM)}}$ (2.554).

Matrix $\pi^{\text{(MPHMM)}}$ (percentage scale) indicates that the probability of students starting their academic careers in performance levels 1, 2, 3, and 4 are 7.78%, 40.20%, 38.29%, and 13.73%, respectively. While the probability of starting school with academic-performance level 4 (13.73%) is close to that of the basic HMM (14.30%), this percentage differs between the two models for other levels. For example, the basic HMM suggests a relatively higher chance for students to start their career at UCF with academic performance level 1 (12.99% in basic HMM versus 7.78% in MPHMM).

Table 7.1 illustrates the distribution of students' academic-performance levels at UCF for the MPHMM and basic HMM. Similar to the basic HMM, the MPHMM suggest that most of the students change their academic-performance level during their academic careers (*Other* group). Comparing performance level distributions for the two models, although the percentages of students with academic-performance levels 2, 3, and 4 are similar (the differences are within 2%), the percentages of students in academic-performance level 1 and in the $Other$ group differ notably. The basic HMM puts 11.4% of students in academic-performance level 1, while that number is only around 4.7% for the MPHMM. This result is aligned with the earlier observation that

Table 7.1: Distribution of students' academic-performance levels at UCF using the MPHMM and basic HMM

Academic-performance level	Basic HMM	MPHMM
level 1	11.4%	4.7%
level 2	13.6%	12.3%
level 3	19.1%	17.9%
level 4	11.8%	10.8%
Other	44.1%	54.3%

students with academic-performance level 1 are less likely to keep their academic performance levels from one semester to the next in the MPHMM model when compared to the basic HMM (70.61%<86.11%). In general, the numbers on the main diagonal of the MPHMM transition matrix are smaller than those of the basic HMM, meaning students are less able to maintain their performance levels, and thus more students are classified under the $Other$ group in the HPHMM model.

In the rest of this section, the academic behavior of students who switch their academic-performance level is investigated and the results are compared to that of the basic HMM. Among students in the *Other* group, 66.2% change their academic performance levels once, 22.7% change their academic performance levels twice, and the remaining students (11.1%) switch their academic-performance level more than 2 times. Compared with the results from the basic HMM, the MPHMM suggest a smaller number of single switch (66.2% vs 75.2% in Basic HMM) students and a larger number of double switches in academic-performance levels. Furthermore, it is more likely for students to have more than 2 switches in the MPHMM when compared to the basic HMM (11.1%>4.3%). Table 7.2 shows the distribution of switching types for students with one switch in academic-performance level using the MPHMM and the basic HMM. Similar to the results from the basic HMM, the three most common switches between academic-performance levels correspond to level 3 →level 4 (26.3%), level 2 →level 3 (25.6%), and level 2 →level 1 (24.8%). For students who change their academic performance levels once, 53.5% improve their academic-performance levels and 46.5% of students decrease to a lower academic-performance

Table 7.2: Distribution of switching types for students with one switch in academic-performance level using the MPHMM and the basic HMM

Switch	Basic HMM	MPHMM (N)	Switch	Basic HMM	MPHMM (N)
level 1 →level 2	2.6% (313)	1.4% (178)	level 3 →level 1	0.2% (29)	0.0% (0)
level 1 →level 3	1.2% (139)	0.1% (18)	level 3 →level 2	13.1% (1566)	14.2% (1846)
level 1 →level 4	0.5% (63)	0.1% (18)	level 3 →level 4	28.4% (3416)	26.3% (3429)
level 2 →level 1	21.1% (2540)	24.8% (3230)	level 4 →level 1	0.0% (1)	0.1% (14)
level 2 →level 3	27.6% (3317)	25.6% (3333)	level 4 →level 2	0.0% (0)	0.0% (0)
level 2 →level 4	0.0% (0)	0.0% (0)	level 4 →level 3	5.3% (636)	7.4% (967)

level. These percentages for the basic HMM were 60.3% and 39.6%, respectively.

Figure 7.1 compares the distribution of academic-performance switches between the MPHMM and the basic HMM. A chi-square hypothesis test was conducted to see if these two distributions differ from each other statistically. The null hypothesis is that the academic performance level switches obtained by both models have the same distributions, and the alternative hypothesis conveys that the mentioned distributions differ from each other. The obtained p-value is close to 1, meaning the null hypothesis cannot be rejected.

Table 7.3 illustrates the halt rates for students with zero or one switches in their academic-performance levels for both MPHMM and the basic HMM. As expected and similar to the results obtained by the basic HMM, the MPHMM suggests that students with higher academic-performance levels have lower halt rates (100.0% > 47.2% > 21.3% > 12.7%). Also, any switches from level 1 to the other levels result in a reduction in halt rates (from 100.0% to 64.6%, 22.2%, and 11.1% for levels 2, 3, and 4, respectively). These decreases are less significant when compared to the decreases obtained from the basic HMM (from 100.0% to 8.9% for switching to level 2, to 6.5% for switching to level 3, and to 4.8% for switching to level 4). Furthermore, based on the table, students who switch to level 1 show an increase in their halt rate. For example, for the MPHMM, the halt rates for students who switched their academic-performance level from 4 to 1 is increased from 12.7% to 85.7%. These percentages from the basic HMM are 11.8% and 100.0%, respectively. Based on the results, while there are some differences in the percentage of students falling within each category, both the MPHMM and the basic HMM show similar trends for all

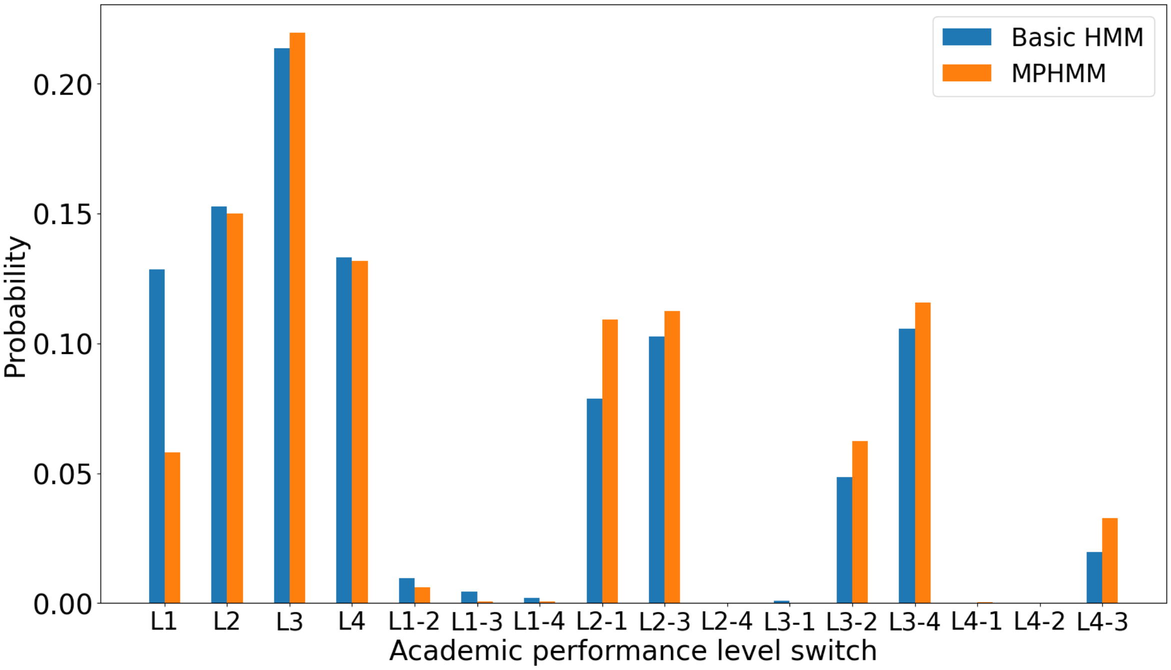


Figure 7.1: Distribution over academic performance level switches using the MPHMM and basic HMM

main features.

Interestingly, similar to the basic HMM, switching from a high academic-performance level to a lower academic-performance level does not necessarily result in a higher halt rate in the MPHMM. For example, students who switch from academic-performance level 4 to 3 have a lower halt rate when compared to students who remain in academic performance level 4 (3.8%<12.7%). The conducted proportion hypothesis test suggested that the difference is statistically significant (p-value= $2.36 \times e^{-15}$). These numbers for the basic HMM are 3.5% and 11.8%, respectively. A similar trend is observed for students who change their academic-performance level from 3 to 2. Another surprising observation is that students who improve their academic-performance from a low level to a higher level have a smaller halt rate when compared to students who consistently performed at the corresponding higher level. For example, students who change their academic-performance level from 3 to 4 have a lower halt rate when compared to students who were always at performance at level 4 (2.2%<12.7%) The conducted proportion hypothesis test suggested that

Table 7.3: Comparing halt rates between students with no switches in academic-performance level and one switch in academic-performance level using the MPHMM and the basic HMM

Staying in	Basic HMM (N)	MPHMM (N)	Switching	Basic HMM	MPHMM
Level 1	97.0% (4146)	100.0% (1722)	level 1 →level 2	8.9%	64.6%
			level 1 →level 3	6.5%	22.2%
			level 1 →level 4	4.8%	11.1%
Level 2	39.8% (4933)	47.2% (4444)	level 2 →level 1	72.0%	97.2%
			level 2 →level 3	4.3%	4.4%
			level 2 →level 4	–	–
Level 3	20.6% (6896)	21.3% (6503)	level 3 →level 1	72.4%	–
			level 3 →level 2	8.7%	13.0%
			level 3 →level 4	2.0%	2.2%
Level 4	11.8% (4295)	12.7% (3903)	level 4 →level 1	100.0%	85.7%
			level 4 →level 2	–	–
			level 4 →level 3	3.5%	3.8%

the difference is statistically significant (p-value= $1.04 \times e^{-62}$). These two interesting results are aligned with those obtained from the basic HMM in chapter 4.

However, there are some differences between the results of these models. For example, based on the MPHMM, students who switch from level 1 to level 3 have a similar halt rate when compared to students who stay in academic-performance level 3 (22.2% vs 21.3%). The conducted proportion hypothesis test suggested that the difference is not statistically significant (p-value= 0.924). The values for the results obtained by the basic HMM are 6.5% and 20.6%, respectively, which are not close to each other. Also, the conducted proportion hypothesis test suggested that the difference is statistically significant (p-value= $4.21 \times e^{-5}$).

Finding 1: While there are some numerical differences between the results obtained by the MPHMM and the basic HMM, both models provide the same answer to the research questions in Chapter 4.

Finding 2: The hypothesis tests indicate that there is no statistically significant difference between the results from the MPHMM and the basic HMM.

7.1 HMM observations distribution

As discussed earlier in this section, the number of A grades that students get each semester follows a Poisson distribution. In this section, it is determined if a similar distribution is observed for students with different academic-performance levels. As an example, we investigate the distribution over the number of A, B, C, DF, and W grades for students with academic-performance levels 2. Figure 7.2 illustrates the results obtained from the basic HMM, showing a Poisson distribution for each of the grades. For example, for students with the academic-performance level 2, the number of A grades that students get in each semester has a Poisson distribution with $\mu = 0.77$. This parameter for grades B, C, DF, and W is 1.53, 0.92, 0.25, and 0.11 respectively. The chi-square hypothesis tests ($N = 34971$) are conducted for all the grade distributions to find if the distributions are truly Poisson. The null hypothesis is that the distribution of each grade is Poisson, and the alternative hypothesis is that the distributions are not Poisson. The obtained p-values of the hypothesis tests for all grades are close to 1, meaning the null hypothesis cannot be rejected, and the number of grades follows Poisson distributions.

Figure 7.3 shows the same results but for the MPHMM. Similar to the distributions obtained from the basic HMM, the grade count distributions for all grades of level 2 students are Poisson in the MPHHM. A similar chi-square hypothesis test is conducted with p-values close to 1, meaning the number of different grades follows a Poisson distribution. Given these results and since the observations have a Poisson distribution, using MPHMM is expected to improve the model performance for both the Baum-Welch and Viterbi algorithms.

7.2 Advantages of the proposed student academic-performance level over traditional criteria in predicting student performance

As mentioned in the introduction, the main goal of this dissertation is to answer the following research question:

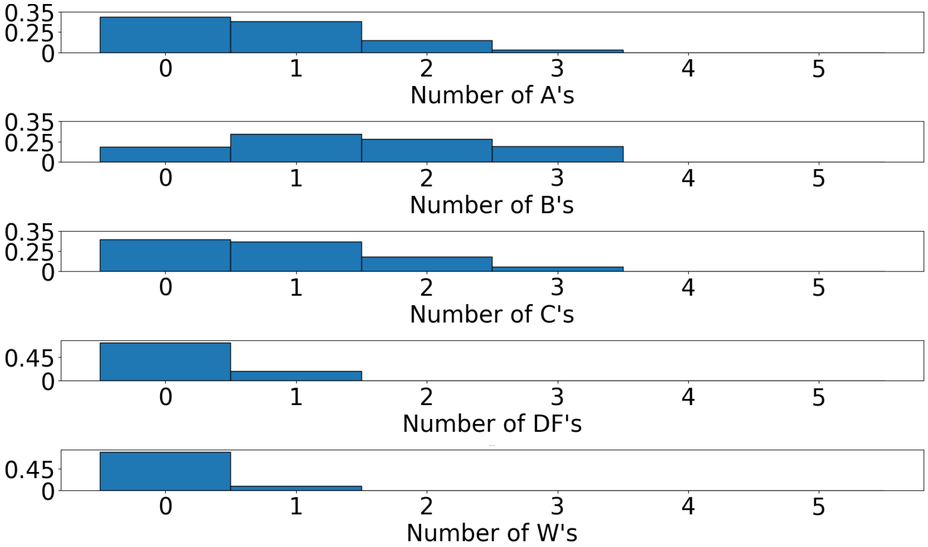


Figure 7.2: Distribution over the number of A, B, C, DF, and W grades for students with academic-performance level 2 obtained by the basic HMM

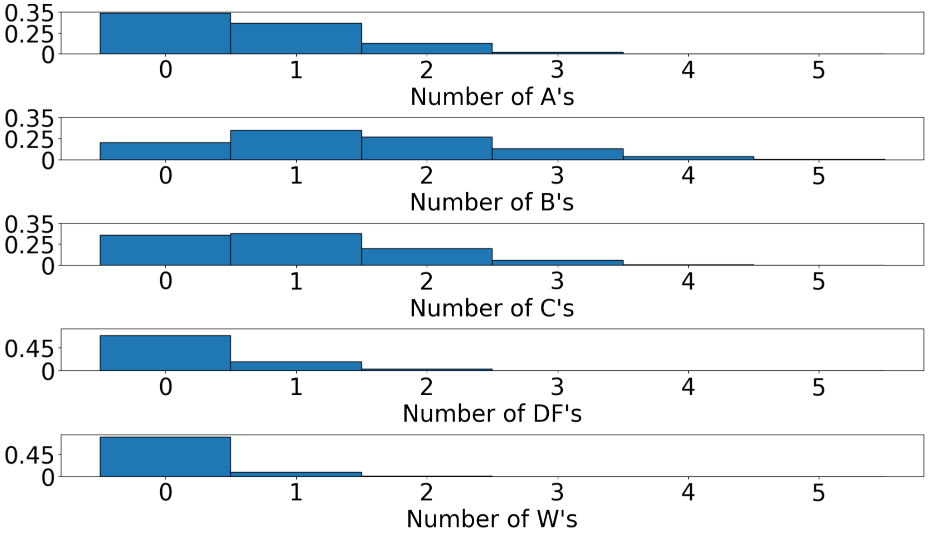


Figure 7.3: Distribution over the number of A, B, C, DF, and W grades for students with academic-performance level 2 obtained by the MPHMM

Research question: Does the proposed academic feature (student academic-performance level) outperform traditional features (i.e., semester GPA) in predicting whether students halt or graduate?

In the rest of this section, two different studies are conducted to answer this question by comparing the accuracy of the new proposed academic feature and the traditional features in predicting whether students graduate or halt.

Performance comparison between the proposed academic feature and traditional features in predicting students' final outcomes. In the first study, four different logistic regression models are created to predict whether students halt or graduate. For the inputs, the first model takes student GPA, the second model takes clustered GPA, the third model takes academic-performance level from the basic HMM, and the fourth and last model takes academic-performance levels from the MPHMM. The prediction accuracy of these models is then compared to investigate the performances of the input features.

Logistic Regression: GPA Input

In previous studies, semester GPA has been considered one of the main predictors of student academic performance [10, 65, 66]. In a study closely related to this chapter, Gershenfeld et. al [67] applied a set of logistic regression models to predict if undergraduate students graduate within six years. Their results indicated that a low first-semester GPA is a significant factor in explaining why students do not graduate within six years.

As a benchmark, in this section, student GPAs are used through the first four semesters as inputs to the logistic regression to identify students who are at risk of halting. The student's GPA in each of the four semesters is represented by one independent variable. The dependent variable for each student is a binary indicator showing if the student halts or graduates. Considering this setup, the following formula computes the halting probability for each student using their GPA in each semester. In the formula, GPA_i refers to students' GPAs for semester i, and B_i is the corresponding coefficient estimated by the logistic regression in the training step.

$$P(halt)^{GPA} = \frac{e^{B_0+B_1\times GPA_1+B_2\times GPA_2+B_3\times GPA_3+B_4\times GPA_4}}{e^{B_0+B_1\times GPA_1+B_2\times GPA_2+B_3\times GPA_3+B_4\times GPA_4}+1}$$

Logistic Regression: Clustered GPA Input

One shortcoming of the approach presented in the previous study is that it might over-weigh semester-to-semester fluctuations in student GPA, which are not considered an indicator of significant performance changes in general. Some recent studies have used clustering approaches to categorize students based on their GPA and then investigate the relationship between GPA categories and their academic performance [100, 5, 101]. This approach results in a more smooth and robust academic-performance trajectory for students.

The clustering begins by considering each data point as a separate cluster. Then, it performs the following phases in multiple iterations: (1) identifying the two clusters which are close to each other and (2) merging the two most similar clusters. By applying this approach, the UCF student GPAs are clustered into four different levels which are used as inputs for the logistics regression model to predict students' halt rate. These different GPA levels are shown in Table 7.4. Similar to the logistics regression in the previous model, the first four semesters are used to train the model and estimate the coefficient values in the following equation. The equation is then used to estimate the halting probability for each student. In the formula, $GPAL_i$ and B_i are student GPA levels in semester i and the corresponding coefficient, respectively.

$$P(halt)^{GPAL} = \frac{e^{B_0+B_1\times GPAL_1+B_2\times GPAL_2+B_3\times GPAL_3+B_4\times GPAL_4}}{e^{B_0+B_1\times GPAL_1+B_2\times GPAL_2+B_3\times GPAL_3+B_4\times GPAL_4}+1}$$

Table 7.4: Semester GPA levels obtained by hierarchical clustering

Level number	Cumulative GPA
1	GPA≤2.5
2	2.5<GPA≤3
3	3.0<GPA≤3.5
4	3.5<GPA≤4.0

Logistic Regression: Performance-Level Input HMM\MPHMM

Finally, in this section, the students' academic-performance levels are estimated using the basic Hidden Markov Model and the Multivariate Poisson Hidden Markov Model, and then fed as inputs to a logistic regression model to predict at-risk students. Similar to the clustered GPA model, this model is less sensitive to normal fluctuations in students' academic performance. That said, these two approaches have some important differences as well.

The first difference between these models and the previous GPA-based models is that in previous models, the logistic regressions were trained based on academic features that were directly observable (GPA). However, in the models studied in this section, the inputs to the logistic regression (student academic-performance levels) are hidden states that are estimated based on observations (the number of A, B, C, D/F, and W grades).

The second difference is that in the first and second logistic regression models, the academic feature in one semester (GPA) was only representative of a student's performance in that specific semester, and was independent of student performance in other semesters. However, in the models trained by HMMs, student academic performance level for each semester is estimated based on student performance in the previous and following semesters. Therefore, student academic-performance trajectories in these models are smoother than those in the previously discussed models. Table 7.4 indicates the inputs and output of these four different logistic regression models.

Table 7.5: Inputs and output of three different logistic regression models to identify at-risk students

Model number	Models inputs	Models output
1	Semester GPA	Graduate/Halt
2	Semester GPA levels obtained by the hierarchical clustering	
3	Academic-performance level estimated by the basic HMM	
4	Academic-performance level estimated by the MPHMM	

Table 7.6: The results of the four different logistic regression models used to predict student academic performance

Models	Model by GPA	Model by Clustering	Basic HMM	MPHMM
Total Accuracy	69.6%	65.7%	64.6%	66.4%
Halt-Accuracy	65.8%	68.8%	78.6%	77.2%
Graduate-Accuracy	70.3%	65.2%	63.4%	64.6%
F1 Score	37.4	35.6	38.0	38.7

Models evaluation

In this section, the four logistics regression models presented in the four previous studies are compared based on their accuracy in predicting if students halt enrollment. The results of these four logistic regression models are summarized in Table 7.6. Based on the table, models 3 and 4 (in which the inputs are student academic-performance level, estimated by the basic HMM and MPHMM) offer higher accuracy in predicting enrollment halt (78.6% and 77.2%, respectively) than the models trained by semester GPA (65.8%) and clustered GPA (68.8%). Also, Figure 7.4 demonstrates that logistic regressions that are built based on the variables created by the HMMs have higher AUC-ROC curves than models created based on traditional features. These results suggest that academic performance level outperforms the traditional features used in the literature in predicting student final outcomes (graduating or halting).

Findings: the proposed student academic-performance level outperforms traditional features in identifying at-risk students.

Performance comparison between models trained by the proposed academic feature and GPA when there is fluctuation in student performance. In Chapter 2, it was shown that using

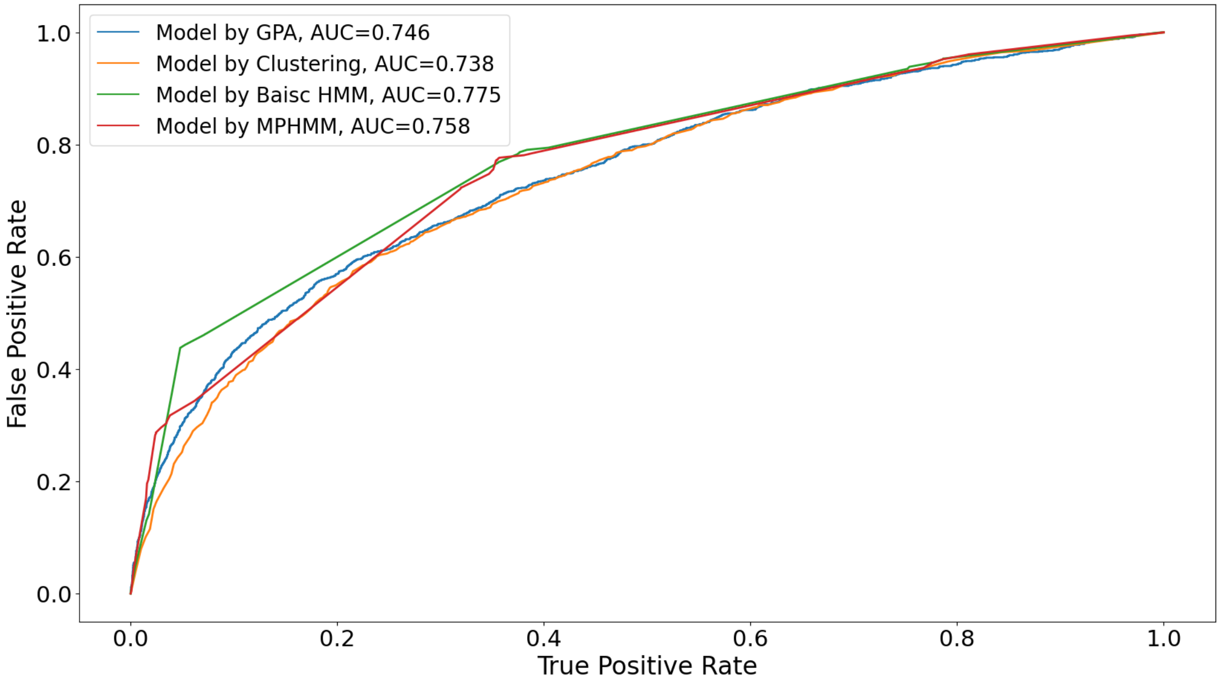


Figure 7.4: AUC - ROC Curve for the four different logistic regression models

semester GPA as input for logistic regression to predict if students halt or graduate could be challenging when there are fluctuations in GPA trajectories. While many of these semester-to-semester fluctuations in GPA may be normal or exceptions, logistic regression and other machine learning models consider them significant indicators of student change in academic performance, which in many cases is misleading. Table 2.6 provided an example of a student with four GPA records, out of which only one was significantly low (Records: 3.55, 1.3, 3.6, 3.4). Machine learning models classified this student as *halt* despite overall high performance.

In this study, a similar experiment was designed to find if machine learning models behave differently when trained by academic-performance levels obtained by the MPHMM. Figure 7.5 shows the impact of student GPA fluctuations on logistic regression prediction trained by student academic performance level (MPHMM). In the figure, the semester GPA sequences for two students are shown. Similar to the example in chapter 2, while these two students have similar semester GPAs in semesters 1, 3, and 4, their second semester GPA differs significantly (3.75 ver-

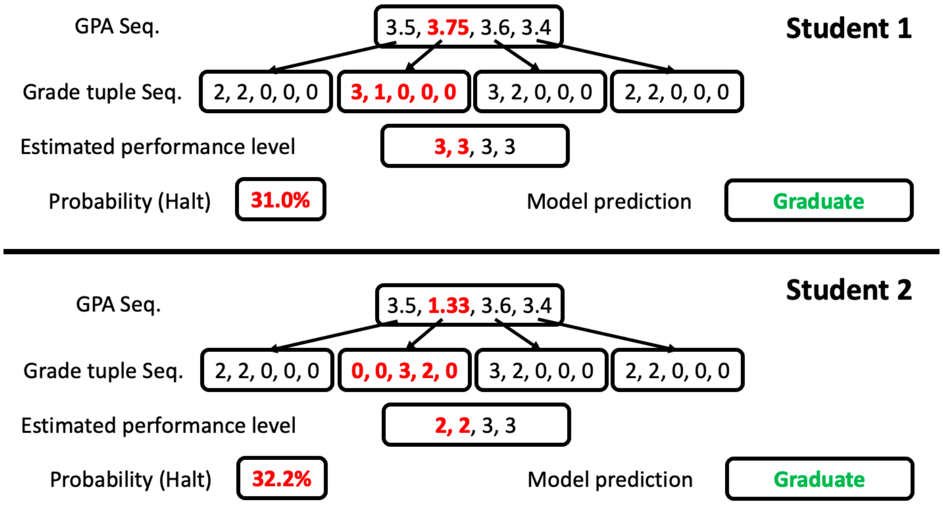


Figure 7.5: Impact of student GPA change on logistic regression prediction trained by student academic-performance level (MPHMM)

sus 1.33). Since the MPHMM works with grade tuples as its inputs, for each student and each semester, the corresponding number of A, B, C, DF, and W grades are computed and fed into the model. For example, student 1 has a GPA of 3.5 in the first semester. The corresponding grade tuple of that semester is (2,2,0 0,0) for (A,B,C,DF,W), respectively. Using the MPHMM, the estimated academic-performance level from semester 1 to semester 4 for the first and second students are (2,2,2,2) and (1,1,2,2), respectively. Using these estimated academic-performance levels as input in the logistic regression, the resulting halt probability for student 1 is 31.1%, implying that this student will graduate and not halt. Unlike the result from GPA-based models, the second student is also considered to graduate, with a halting probability of 32.2%.

These two studies indicate the superior accuracy of the proposed academic-performance level feature when compared to traditional GPA-based features in predicting student performance. This improved accuracy mainly resulted from the inherent integration and correlation between records in different semesters, which are captured by the HMM and MPHMM. This provides a more robust outlook on student performance by preventing normal semester-to-semester fluctuations from warping results.

Findings: The proposed academic-performance level outperforms GPA in halt prediction

models in the cases where there are fluctuations in student performance.

7.3 Conclusion

In the first section of this chapter, the performance of MPHMM was compared against the basic HMM using UCF record data. The Baum-Welch algorithm was used to estimate model parameters, and the results suggested that both models show a similar trend when it comes to transition probability between different levels. The conducted hypothesis tests indicated that there is no statistically significant difference between the results obtained from these models. Despite the similarities, there were few cases where the two models suggested different results; for example, the probability of a student switching from level 1 to other levels is higher in the MPHMM than in the basic HMM. This also results in more students being classified as $Other$ (referring to students who switch their academic level at least once) in the MPHMM.

In the second section of this chapter, two different studies were used to assess the prediction accuracy of the academic-performance level against traditional GPA-based features. In the first study, four different logistic regression models were implemented to compare the performance of the proposed academic-performance level to semester GPA in predicting if students graduate or halt. The first and second models used semester GPA and GPA clusters, respectively, as inputs in the logistic regression model, whereas the third and fourth models were fed by student academic-performance levels obtained from the basic HMM and MPHMM, respectively. Results indicated that the proposed academic-performance level outperforms the traditional GPA-based feature in identifying students who halt enrollment. The second study was to emphasize the importance of input features in logistic regression prediction accuracy. When training the logistic regression model with semester-to-semester GPA, the model was unable to identify which grade fluctuations were relevant in identifying at-risk students; that is, when a more integrated input was provided (academic-performance level), the model could distinguish between negligible and relevant grade fluctuations, and as such provide a better prediction.

CHAPTER 8
CONCLUSION

In order to improve graduation rates and decrease halt rates in universities and higher education institutions, it is critical to understand academic and demographic factors that correlate to student academic performance. The importance of such factors in predicting students' halt/graduate status, and the advantage of recent statistics and data mining tools in facilitating such predictions, has stimulated a large body of scientific research known as Educational Data Mining (EDM). Among students' academic features, many studies in the field of EDM have identified students' grade point averages (GPAs) and course grades as the most important factors affecting halt rates [9].

The *first objective* of this dissertation was to assess the strengths and weaknesses of common machine learning techniques as their traditional inputs for modeling the impact of academic performance on student graduation rates. After the shortcomings of the traditional inputs (i.e., GPA) were discussed, an alternative feature was introduced to overcome the identified limitations; this was the *second objective* of the research. The formulation and fundamental concept of the basic empirical hidden Markov model algorithm was discussed to provide a clear step-by-step thought process on why and how this alternative approach was developed. After proposing a new academic feature called student academic-performance level, the *third objective* was to identify the relationship between student academic-performance level and student halt rate. These results highlighted an important limitation of the basic HMM, which was its low accuracy for the cases where data was generated from a Poisson distribution. As the *fourth objective* of this research, a Multivariate Poisson Hidden Markov Model (MPHMM) was developed to overcome this limitation.

In the rest of this chapter, section 8.1 to section 8.4 provides a summary of this thesis' contribution corresponding with the four objectives, and section 8.5 discusses the limitations of this dissertation and proposes avenues for future research.

8.1 Limitations of machine learning models and traditional features in EDM

As mentioned earlier, GPA and course grades have been considered important academic features for machine learning models to predict student outcomes. In this dissertation, the limitations of previous studies in predicting whether a student halts or graduates was analyzed from two points of view: (1) limitations in the machine learning techniques, and (2) the features that these models were built off of.

Six different machine learning models, including Random Forest, Support Vector classifier, Logistic Regression, Gaussian Naive Bayes classifier, Gradient Boosting, and K-Nearest Neighbor, were applied to UCF student records to predict if students graduated or halted enrollment. In the first study, the mentioned machine learning models used the cumulative number of A, B, C, D, F, and W grades that a student obtained over their first two academic semesters as the input to predict if the student will halt enrollment or ultimately graduate. One main limitation common to many of these models is the timing of when the predictions are made. While early predictions after one or two semesters may have low prediction accuracy, a late prediction may not provide sufficient time to intervene with at-risk students. The conducted analysis indicated that the more semester grade-data used in the training set, the higher the model accuracy. Therefore, balancing the trade-off between early detection against improved model accuracy is challenging.

In the second study, a similar model was developed, but instead of using cumulative grade counts over multiple semesters, semester grade counts served as input for the prediction model. While the previous cumulative grade-count models had six inputs, a model using this input structure based on N semesters will now have $6N$ inputs. The same six machine learning models were applied based on the new input structure. One advantage of these models over the previous models is that the new models could determine in which semester a specific student had better performance. The obtained results illustrated that there is no significant improvement in results compared to the results from the previous models. Such a finding shows that if two students have the same cumulative GPA, even if they have different grade patterns (i.e., high-to-low and low-to-high), they will have the same probability of halting based on these models. In fact, these models do not consider the impact of patterns in students' academic performance (e.g., decreasing or increasing grades) on their final academic outcomes. In other words, these models assume that a student's performance in each semester is independent of the student's performance in other semesters.

In the third study, the same six machine learning techniques were used to predict whether students graduate or halt, however, semester GPA was considered as the input instead of grade-count. The obtained results indicated that grade-count is a more accurate input feature than GPA in predicting if a student would graduate or halt. Such a result was expected as a greater number of input features (6 for the grade-count and 1 for GPA) results in higher model accuracy. Furthermore, this study used GPA data of two different students to investigate how fluctuations in GPA could impact model accuracy. The first student had high GPAs (more than 3.4) in four semesters. On the other hand, the second student had high GPAs for three semesters but had a drop in GPA in the second semester (1.3). All six machine learning models classified student 1 as graduated and student 2 as halted. In other words, having a low GPA among three high GPAs caused the second student to be classified as halted. Therefore, as another limitation, using GPA as the input for the machine learning model is challenging since it could not distinguish between normal and significant changes in student academic performance.

8.2 Proposing student academic-performance level using HMM as an alternative to traditional features

In Chapter 2, the limitations of traditional features (i.e., semester GPA) in predicting student performance is investigated. In this dissertation, academic-performance level, a new academic feature, was introduced as an alternative to GPA that is used in literature as an important indicator of student outcomes. Proposing this new academic feature is considered as the *first contribution* of this dissertation. Unlike traditional statistical methodologies, the proposed approach is able to provide a standard point of reference for comparing a student's GPA both across their enrolled semesters and to other students.

In this study, the Hidden Markov Model (HMM) (an unsupervised learning technique) was proposed to define student academic-performance level. Markov models are commonly used to model systems and processes that exhibit stochastic dynamics, specifically systems and processes that switch between states or operating modes. However, in the Hidden Markov model, unlike other Markov models, the statuses of states are hidden, and the aim is to estimate the hidden states based on the sequence of observations which are generated from the hidden state. In this study, the proposed HMM takes the sequence of course grades tuple over multiple semesters (the HMM observation) and returns the sequence of estimated academic-performance levels (HMM hidden states). The course grade tuples consist of the number of A, B, C, DF (combined), and W grades a student receives in a semester.

As the first finding of this study, it was shown that while W grades are not considered when computing student GPA, there is a significant correlation between withdrawing from a course and academic-performance level. Also, results indicated that students who consistently keep a low academic-performance level are more likely to halt enrollment than other students. Furthermore, the higher the academic-performance level, the lower the halt rate. Analyzing the halt rate for students who change their academic-performance level provided other results. For example, students who increase their academic-performance levels from the lowest level to other levels will have a significant decrease in their halt rates. Finally, this study had two surprising results. First, switching from a high academic-performance level to a low academic performance level does not necessarily increase the halt rate. Second, students that improve their academic performance have halt rates below students that maintain consistent levels.

8.3 Developing a Multivariate Poisson Hidden Markov model for analyzing students' academic-performance trajectories

In Chapter 4 of this dissertation, a basic empirical HMM was developed to propose a new feature, called academic performance level, as an alternative to traditional features (i.e., GPA) that were commonly used to predict student performance. Despite the novelty of the proposed feature, the chapter highlighted that a large number of parameters are required to properly formulate the model, which would be prohibitive when applying this machine learning technique at smaller institutions with less available training data. It was found that since the number of grades students earn each semester follows a Poisson distribution, the development of a Multivariate Poisson Hidden Markov Model (MPHMM) could overcome the observed limitation. As such, the *second contribution* of this dissertation was the development of the necessary equations for implementing the Multivariate Poisson HMM, which to the best of the author's knowledge, had not been investigated in literature previously. A simulation model was used in Chapter 6 to compare the benefits of the proposed MPHMM to the basic categorical HMM. The findings of the simulation indicated the superior performance of the MPHMM in terms of parameters and hidden state estimation when data points are generated from a Poisson distribution.

In the second part of this chapter, the MPHMM was applied to UCF data to find if any difference exists between the results from the MPHMM and the basic empirical HMM in analyzing student academic-performance trajectories. The findings showed that while there are some numerical differences in student transition between different performance levels, both models suggest the same general pattern. For example, the halt rates for students who switch their academic-performance level from 4 to 1 was increased from 12.7% to 85.7% in the MPHMM. The percentages for this in the basic HMM were 11.8% and 100.0%, respectively. In fact, a similar pattern of significant halt rate increase was observed in both models, despite the minor numerical differences.

8.4 Advantage of the proposed students' academic-performance level over traditional approaches

As was mentioned earlier, as a contribution, this dissertation proposed an alternative to traditional features used in predicting student outcomes. Two different studies were conducted to compare the performance of the proposed academic feature and traditional features in predicting whether students graduate or halt. In the first study, four logistic regression models were developed to predict if students halt or graduate. For the models' input, the first model used student GPA, the second used cluster GPA, the third model used the academic-performance level obtained by the basic HMM, and the last model used the academic-performance level from the MPHMM. While the first two logistic regression models were built based on the observable (traditional) features, the estimated hidden states (academic-performance levels) were used as inputs for models 3 and 4, which are not directly observable. Prediction accuracy indicated that the proposed academic-performance level obtained by the HMMs outperforms traditional features in identifying students who halt enrollment.

In Chapter 2, it was shown how using student semester GPA as input for machine learning models in predicting whether students halt or graduate could be challenging when there are fluctuations in students' GPA trajectories. While many of these semester-to-semester fluctuations in GPA might be normal or exceptions, logistic regression and other machine learning models consider them significant indicators of changes in academic performance which, in many cases, is misleading. In the chapter, two different students were selected that had the same GPA for their first, third, and fourth semesters. However, student 2 had a lower GPA than student 1 in the second semester. All applied machine learning models classified student 1 as graduated and student 2 as halted. Thus having one low GPA among three high GPAs caused student 2 to be classified as halted. In the second study in Chapter 7, for the same two students, the corresponding grade counts for each semester's GPA were extracted. Then, after applying the MPHMM to estimate academic-performance levels for students 1 and 2, logistic regression was trained based on the estimated academic-performance. In other words, the estimated hidden states from the MPHMM fed into the logistic regression to predict if students 1 and 2 graduated or halted enrollment. Unlike Chapter 2, in this section, both students were classified as halted, which implied that the proposed academic-performance level outperforms GPA in the cases where there are fluctuations in student performance. This finding provides a more robust outlook on student performance by taking into account normal semester-to-semester fluctuations in their record history.

8.5 Limitations and future works

This study, like any other research, is not without limitations. One of its limitations is that it only considers students' semester-to-semester course grades to predict halt status. As explained in the literature section, other academic factors like enrollment type (full-time and part-time) could also significantly affect student performance. Since student enrollment types may change from one semester to the next, including this feature in the proposed HMMs is logically possible. The only barrier to this implementation is that unlike the number of grades that has a Poisson distribution, the enrollment type follows a binary distribution (either full-time or part-time), and therefore a multivariate HMM is needed to support different distributions for each observation dimension. As an avenue for future research, an extension to the proposed MPHMM in this dissertation can be developed with a 6-dimensional observation vector; dimensions 1 to 5 may correspond to the number of A, B, C, DF, and W grades (with a Poisson distribution), and the sixth dimension designating if a student is full-time or part-time (which follows a binary distribution).

In Chapters 4 and 7 of this dissertation, an arbitrary number (i.e., 4) was used for the number of hidden states determining academic performance level. However, some studies have indicated that the number of hidden states could impact model accuracy in terms of likelihood [102]. Based on the literature, AIC and BIC are two popular criteria to determine the optimal values for the number of hidden states. Therefore, finding an optimal value for the number of student academic-performance levels (the hidden states) and comparing the new results with the obtained results in this dissertation could another future work.

In Chapter 7, it was shown that although both MPHMM and basic HMM show the same general pattern in regards to student academic performance-level transitions, there are some numerical differences between the results from the two models. Furthermore, those numerical differences were more evident for students with academic-performance level 1. It is an interesting open question for future research to investigate the reasons for such differences and advise on what factors are driving a smaller or bigger gap in the obtained results. Those findings may lead to a deeper understating of the MPHMM behavior and its advantages over the basic HMM.

In this dissertation, only the impact of a student changing academic-performance level on halt rate was investigated. However, in reality, there are other factors that change from one semester to the next that could impact student performance. For example, some students change their major, which could be a significant factor in student halt rate. For instance, students who change their major from engineering to business are more likely to have an increase in GPA, and thus a higher chance of graduation. Therefore, as another avenue for future research, the impact of such factors along with changes in students' academic-performance levels on halt rates will be investigated.

The results obtained in this dissertation are based on data collected at the University of Central Florida between 2008 and 2017. As was mentioned in Chapter 3, UCF is a unique university in terms of students demographics (e.g., a significant portion of Hispanic and transfer students). While UCF's population size makes it a good candidate for doing educational research, this demographic uniqueness may yield conclusions that are not representative of other universities. To understand how universities' different demographics impact student performance, future research may apply the proposed HMMs to data from other universities and compare the results.

Furthermore, the data set used in this dissertation has some limitations. Explaining such limitations is crucial as it guides towards more practical data collection and organization, leading to more informed analysis and thus producing better insights. For example, the data set under consideration did not include academic and demographic information on students who halted enrollment and left school without a degree. Such information, if present, could improve the analysis of students performance, their academic patterns (including GPA, enrollment, etc.), and the corresponding consequences on outcomes. Moreover, for students who transferred to UCF from community colleges or other universities, their academic performance such as transcripts and GPAs corresponding to their previous institutions were not included.

The impact of the Covid-19 pandemic on higher education is another limitation of this dissertation. The analysis conducted in this study is based on data collected before the pandemic. However, as many papers suggested, the Covid-19 pandemic has affected higher education and student performance significantly [103, 104]. For example, the University of Central Florida shifted grading policy after the pandemic, allowing students to have satisfactory or unsatisfactory grades in place of letter grades [105]. Due to the new grade structure, applying the proposed HMMs to post-Covid student records could be impossible. Since the new grading policy may have caused some teachers to grade more leniently than before, when applying the proposed HMMs to post-Covid student record data, the results might be different from the results in this dissertation.

In this dissertation, six different machine learning models were used, with semester grade counts as inputs, to predict if students graduate or halt enrollment. One aim of this study was to investigate if these models could determine in which semester a specific student has a better performance. The obtained results indicated that these models are not able to consider the impact of patterns in student academic performance (e.g., decreasing or increasing) on student final outcomes. Development of Recurrent Neural Network (RNN) models for this problem and comparing its results to the results from the six applied machine learning models, could be considered another idea for future studies.

APPENDIX: IRB APPROVAL

UCF

UNIVERSITY OF CENTRAL FLORIDA

Institutional Review Board
FWA00000351
IRB00001138, IRB00012110
Office of Research
12201 Research Parkway
Orlando, FL 32826-3246

NOT HUMAN RESEARCH DETERMINATION

April 7, 2022

Dear Adan Vela:

On 4/7/2022, the IRB reviewed the following protocol:

Type of Review:	Initial Study
Title of Study:	Development of a multivariate Poisson hidden Markov model for application in educational data mining
Investigator:	Adan Vela
Study Team:	Shahab Boumi
IRB ID:	STUDY00004148
Funding:	Name: US Department of Homeland Security
Grant ID:	
Documents Reviewed:	• HRP-250, Category: IRB Protocol; • List of primary data features used in study, Category: Other

The IRB determined that the proposed activity is not research involving human subjects as defined by DHHS and FDA regulations.

IRB review and approval by this organization is not required. This determination applies only to the activities described in the IRB submission and does not apply should any changes be made. If changes are made and there are questions about whether these activities are research involving human in which the organization is engaged, please submit a new request to the IRB for a determination. You can create a modification by clicking **Create Modification / CR** within the study.

If you have any questions, please contact the UCF IRB at 407-823-2901 or irb@ucf.edu. Please include your project title and IRB number in all correspondence with this office.

Sincerely,

Katie Kilgore

Katie Kilgore
Designated Reviewer

REFERENCES

[1] D. Shapiro, A. Dundar, F. Huie, P. K. Wakhungu, A. Bhimdiwala, and S. E. Wilson, “Completing college: A national view of student completion rates–fall 2012 cohort (signature report no. 16).,” *National Student Clearinghouse*, 2018.

[2] B. Derek, “Improving the quality of education,” *Retrieved on*, vol. 26, no. 04, p. 2018, 2017.

[3] P. R. cente. “What’s behind the growing gap between men and women in college completion?, url = https://pewrsr.ch/3n2ZuDn.” ().

[4] N. C. R. Center. “Graduation rates and race, url = https://www.insidehighered.com/news/2017/04/26/college-completion-rates-vary-race-and-ethnicity-report-finds.” ().

[5] F. Marbouti, J. Ulas, and C.-H. Wang, “Academic and demographic cluster analysis of engineering student success,” *IEEE Transactions on Education*, 2020.

[6] M. J. Feldman, “Factors associated with one-year retention in a community college,” *Research in Higher Education*, vol. 34, no. 4, pp. 503–512, 1993.

[7] R. Darolia, “Working (and studying) day and night: Heterogeneous effects of working on the academic performance of full-time and part-time students,” *Economics of Education Review*, vol. 38, pp. 38–50, 2014.

[8] N. L. Smith, J. R. Grohs, and E. M. Van Aken, “Comparison of transfer shock and graduation rates across engineering transfer student populations,” *Journal of Engineering Education*, vol. 111, no. 1, pp. 65–81, 2022.

[9] M. Plagge, “Using artificial neural networks to predict first-year traditional students second year retention rates,” in *Proceedings of the 51st ACM Southeast Conference*, 2013, pp. 1–5.

[10] T. Ojha, G. L. Heileman, M. Martinez-Ramon, and A. Slim, “Prediction of graduation delay based on student performance,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2017, pp. 3454–3460.

[11] 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.

[12] Z. K. Papamitsiou and A. A. Economides, "Learning analytics and educational data mining in practice: A systematic literature review of empirical evidence.," *Educational Technology & Society*, vol. 17, no. 4, pp. 49–64, 2014.

[13] R. Baker *et al.*, "Data mining for education," *International encyclopedia of education*, vol. 7, no. 3, pp. 112–118, 2010.

[14] L. Shamseer *et al.*, "Preferred reporting items for systematic review and meta-analysis protocols (prisma-p) 2015: Elaboration and explanation," *Bmj*, vol. 349, 2015.

[15] D. Delen, "A comparative analysis of machine learning techniques for student retention management," *Decision Support Systems*, vol. 49, no. 4, pp. 498–506, 2010.

[16] P. Imbrie, J. J.-J. Lin, and A. Malyscheff, "Artificial intelligence methods to forecast engineering students' retention based on cognitive and non cognitive factors," in *2008 Annual Conference & Exposition*, 2008, pp. 13–222.

[17] J. He, J. Bailey, B. Rubinstein, and R. Zhang, "Identifying at-risk students in massive open online courses," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, 2015.

[18] R. Ghorbani and R. Ghousi, "Comparing different resampling methods in predicting students' performance using machine learning techniques," *IEEE Access*, vol. 8, pp. 67 899–67 911, 2020.

[19] A. C. Lagman, L. P. Alfonso, M. L. I. Goh, J. Lalata, J. P. H. Magcuyao, and H. N. Vicente, "Classification algorithm accuracy improvement for student graduation prediction using ensemble model," *International Journal of Information and Education Technology*, vol. 10, no. 10, pp. 723–727, 2020.

[20] F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, "Models for early prediction of at-risk students in a course using standards-based grading," *Computers & Education*, vol. 103, pp. 1–15, 2016.

[21] L. Zahedi, S. J. Lunn, S. Pouyanfar, M. S. Ross, and M. W. Ohland, "Leveraging machine-learning techniques to analyze computing persistence in undergraduate programs," in *2020 ASEE Virtual Annual Conference Content Access*, https://peer.asee.org/34921, Virtual On line: ASEE Conferences, Jun. 2020.

[22] S. Huang and N. Fang, "Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models," *Computers & Education*, vol. 61, pp. 133–145, 2013.

[23] C. Zabriskie, J. Yang, S. DeVore, and J. Stewart, “Using machine learning to predict physics course outcomes,” *Physical Review Physics Education Research*, vol. 15, no. 2, p. 020 120, 2019.

[24] A. P. Alfiani, F. A. Wulandari, *et al.*, “Mapping student’s performance based on data mining approach (a case study),” *Agriculture and Agricultural Science Procedia*, vol. 3, pp. 173–177, 2015.

[25] Z. N. Khan, “Scholastic achievement of higher secondary students in science stream.,” *Online Submission*, vol. 1, no. 2, pp. 84–87, 2005.

[26] S. Goldrick-Rab and S. W. Han, “Accounting for socioeconomic differences in delaying the transition to college,” *The Review of Higher Education*, vol. 34, no. 3, pp. 423–445, 2011.

[27] R. D. Cox, “Complicating conditions: Obstacles and interruptions to low-income students’ college “choices”,” *The Journal of Higher Education*, vol. 87, no. 1, pp. 1–26, 2016.

[28] S. Goldrick-Rab, R. Kelchen, D. N. Harris, and J. Benson, “Reducing income inequality in educational attainment: Experimental evidence on the impact of financial aid on college completion,” *American Journal of Sociology*, vol. 121, no. 6, pp. 1762–1817, 2016.

[29] H. T. Rowan-Kenyon, “Predictors of delayed college enrollment and the impact of socioeconomic status,” *The Journal of Higher Education*, vol. 78, no. 2, pp. 188–214, 2007.

[30] R. S. Wells and C. M. Lynch, “Delayed college entry and the socioeconomic gap: Examining the roles of student plans, family income, parental education, and parental occupation,” *The Journal of Higher Education*, vol. 83, no. 5, pp. 671–697, 2012.

[31] S. L. DesJardins, D. A. Ahlburg, and B. P. McCall, “The effects of interrupted enrollment on graduation from college: Racial, income, and ability differences,” *Economics of Education Review*, vol. 25, no. 6, pp. 575–590, 2006.

[32] J. A. Jacobs and R. B. King, “Age and college completion: A life-history analysis of women aged 15-44,” *Sociology of Education*, pp. 211–230, 2002.

[33] J. Roksa and M. Velez, “A late start: Delayed entry, life course transitions and bachelor’s degree completion,” *Social forces*, vol. 90, no. 3, pp. 769–794, 2012.

[34] H. Taniguchi and G. Kaufman, “Degree completion among nontraditional college students,” *Social Science Quarterly*, vol. 86, no. 4, pp. 912–927, 2005.

[35] Burley, B. Butner, B. Cejda, and Hansel, "Dropout and stopout patterns among developmental education students in texas community colleges," *Community College Journal of Research and Practice*, vol. 25, no. 10, pp. 767–782, 2001.

[36] P. M. Crosta, "Intensity and attachment: How the chaotic enrollment patterns of community college students relate to educational outcomes," *Community College Review*, vol. 42, no. 2, pp. 118–142, 2014.

[37] P. R. Bahr, "The bird's eye view of community colleges: A behavioral typology of first-time students based on cluster analytic classification," *Research in Higher Education*, vol. 51, no. 8, pp. 724–749, 2010.

[38] P. Attewell, S. Heil, and L. Reisel, "What is academic momentum? and does it matter?" *Educational Evaluation and Policy Analysis*, vol. 34, no. 1, pp. 27–44, 2012.

[39] G. D. Kuh, "What student affairs professionals need to know about student engagement," *Journal of college student development*, vol. 50, no. 6, pp. 683–706, 2009.

[40] K. McClenney, C. N. Marti, and C. Adkins, "Student engagement and student outcomes: Key findings from," *Community college survey of student engagement*, 2012.

[41] J.-S. Lee, "The relationship between student engagement and academic performance: Is it a myth or reality?" *The Journal of Educational Research*, vol. 107, no. 3, pp. 177–185, 2014.

[42] D. V. Price and E. Tovar, "Student engagement and institutional graduation rates: Identifying high-impact educational practices for community colleges," *Community College Journal of Research and Practice*, vol. 38, no. 9, pp. 766–782, 2014.

[43] D. Upendran, S. Chatterjee, S. Sindhumol, and K. Bijlani, "Application of predictive analytics in intelligent course recommendation," *Procedia Computer Science*, vol. 93, pp. 917–923, 2016.

[44] S. Morsy and G. Karypis, "Learning course sequencing for course recommendation," 2018.

[45] Y.-h. Wang, M.-H. Tseng, and H.-C. Liao, "Data mining for adaptive learning sequence in english language instruction," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7681–7686, 2009.

[46] R. Farzan and P. Brusilovsky, "Social navigation support in a course recommendation system," in *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, Springer, 2006, pp. 91–100.

[47] N. Bendakir and E. Aimeur, “Using association rules for course recommendation,” in *Proceedings of the AAAI workshop on educational data mining*, vol. 3, 2006, pp. 1–10.

[48] E. S. Khorasani, Z. Zhenge, and J. Champaign, “A markov chain collaborative filtering model for course enrollment recommendations,” in *2016 IEEE International Conference on Big Data (Big Data)*, IEEE, 2016, pp. 3484–3490.

[49] L. E. Baum and T. Petrie, “Statistical inference for probabilistic functions of finite state markov chains,” *The annals of mathematical statistics*, vol. 37, no. 6, pp. 1554–1563, 1966.

[50] X. Yin, “Estimation of hidden markov model,” *Lehigh Uniersity*, 2018.

[51] M. Tadayon and G. J. Pottie, “Predicting student performance in an educational game using a hidden markov model,” *IEEE Transactions on Education*, 2020.

[52] R. Luckin *et al.*, “Modeling learning patterns of students with a tutoring system using hidden markov models,” *Artificial intelligence in education: Building technology rich learning contexts that work*, vol. 158, p. 238, 2007.

[53] Y.-H. Hu, C.-L. Lo, and S.-P. Shih, “Developing early warning systems to predict students’ online learning performance,” *Computers in Human Behavior*, vol. 36, pp. 469–478, 2014.

[54] H. Lakkaraju *et al.*, “A machine learning framework to identify students at risk of adverse academic outcomes,” in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 1909–1918.

[55] Z. Iqbal, J. Qadir, A. N. Mian, and F. Kamiran, “Machine learning based student grade prediction: A case study,” *arXiv preprint arXiv:1708.08744*, 2017.

[56] F. Ahmad, N. H. Ismail, and A. A. Aziz, “The prediction of students’ academic performance using classification data mining techniques,” *Applied Mathematical Sciences*, vol. 9, no. 129, pp. 6415–6426, 2015.

[57] E. Osmanbegovic and M. Suljic, “Data mining approach for predicting student performance,” *Economic Review: Journal of Economics and Business*, vol. 10, no. 1, pp. 3–12, 2012.

[58] S. Kotsiantis, K. Patriarcheas, and M. Xenos, “A combinational incremental ensemble of classifiers as a technique for predicting students’ performance in distance education,” *Knowledge-Based Systems*, vol. 23, no. 6, pp. 529–535, 2010.

[59] A. Ahmed and I. S. Elaraby, “Data mining: A prediction for student’s performance using classification method,” *World Journal of Computer Application and Technology*, vol. 2, no. 2, pp. 43–47, 2014.

[60] W. Zhang, X. Huang, S. Wang, J. Shu, H. Liu, and H. Chen, “Student performance prediction via online learning behavior analytics,” in *2017 International Symposium on Educational Technology (ISET)*, IEEE, 2017, pp. 153–157.

[61] Z. Kovacic, “Early prediction of student success: Mining students’ enrolment data.,” 2010.

[62] A. Djulovic and D. Li, “Towards freshman retention prediction: A comparative study,” *International Journal of Information and Education Technology*, vol. 3, no. 5, pp. 494–500, 2013.

[63] J. R. Betts and D. Morell, “The determinants of undergraduate grade point average: The relative importance of family background, high school resources, and peer group effects,” *Journal of human Resources*, pp. 268–293, 1999.

[64] J. D. Rennie, L. Shih, J. Teevan, and D. R. Karger, “Tackling the poor assumptions of naive bayes text classifiers,” in *Proceedings of the 20th international conference on machine learning (ICML-03)*, 2003, pp. 616–623.

[65] J. E. Timer and M. I. Clauson, “The use of selective admissions tools to predict students’ success in an advanced standing baccalaureate nursing program,” *Nurse Education Today*, vol. 31, no. 6, pp. 601–606, 2011.

[66] R. L. Wilson and B. C. Hardgrave, “Predicting graduate student success in an mba program: Regression versus classification,” *Educational and psychological measurement*, vol. 55, no. 2, pp. 186–195, 1995.

[67] S. Gershenfeld, D. Ward Hood, and M. Zhan, “The role of first-semester gpa in predicting graduation rates of underrepresented students,” *Journal of College Student Retention: Research, Theory & Practice*, vol. 17, no. 4, pp. 469–488, 2016.

[68] U. O. C. FLORIDA. “Ucf facts.” (2020).

[69] ——, “About aip.” (2020).

[70] L. Musu-Gillette, C. De Brey, J. McFarland, W. Hussar, W. Sonnenberg, and S. Wilkinson-Flicker, “Status and trends in the education of racial and ethnic groups 2017. nces 2017-051.,” *National Center for Education Statistics*, 2017.

[71] U. O. C. FLORIDA. “U.s. department of education designates ucf as a hispanic serving institution.” (2019).

[72] V. Tinto, *College student retention: Formula for student success*. Greenwood publishing grouP, 2005.

[73] S. Boumi and A. E. Vela, “Improving graduation rate estimates using regularly updating multi-level absorbing markov chains,” *Education Sciences*, vol. 10, no. 12, p. 377, 2020.

[74] L. Zhai and R. Monzon, “Community college student retention: Student characteristics and withdrawal reasons.,” 2001.

[75] F. T. Bail, S. Zhang, and G. T. Tachiyama, “Effects of a self-regulated learning course on the academic performance and graduation rate of college students in an academic support program,” *Journal of college reading and learning*, vol. 39, no. 1, pp. 54–73, 2008.

[76] B. J. Hosch, “The tension between student persistence and institutional retention: An examination of the relationship between first-semester gpa and student progression rates of first-time students,” in *Association for Institutional Research Annual Forum, Seattle, WA*, 2008.

[77] ——, “Institutional and student characteristics that predict graduation and retention rates,” in *annual meeting of the North East Association for Institutional Research, Providence, RI*, 2008.

[78] R. Bolden Eddie; Conyers and K. King. “Predictors for retention, persistence, and success.” (2020).

[79] A. A. Addus, D. Chen, and A. S. Khan, “Academic performance and advisement of university students: A case study,” *College Student Journal*, vol. 41, no. 2, pp. 316–327, 2007.

[80] B. Fadem, M. Schuchman, and S. S. Simring, “The relationship between parental income and academic performance of medical students.,” *Academic Medicine*, 1995.

[81] I. Mushtaq and S. N. Khan, “Factors affecting students’ academic performance,” *Global journal of management and business research*, vol. 12, no. 9, pp. 17–22, 2012.

[82] G. Considine and G. Zappalà, “The influence of social and economic disadvantage in the academic performance of school students in australia,” *Journal of sociology*, vol. 38, no. 2, pp. 129–148, 2002.

[83] A. S. Online. “About aip.” (2020).

[84] P. Garibaldi, F. Giavazzi, A. Ichino, and E. Rettore, “College cost and time to complete a degree: Evidence from tuition discontinuities,” *Review of Economics and Statistics*, vol. 94, no. 3, pp. 699–711, 2012.

[85] E. Hovdhaugen and P. O. Aamodt, “Learning environment: Relevant or not to students’ decision to leave university?” *Quality in higher education*, vol. 15, no. 2, pp. 177–189, 2009.

[86] N. D. Shippee and T. J. Owens, “Gpa, depression, and drinking: A longitudinal comparison of high school boys and girls,” *Sociological Perspectives*, vol. 54, no. 3, pp. 351–376, 2011.

[87] E. Hull-Blanks, S. E. R. Kurpius, C. Befort, S. Sollenberger, M. F. Nicpon, and L. Huser, “Career goals and retention-related factors among college freshmen,” *Journal of Career Development*, vol. 32, no. 1, pp. 16–30, 2005.

[88] I. O. Pappas, M. N. Giannakos, and L. Jaccheri, “Investigating factors influencing students’ intention to dropout computer science studies,” in *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, 2016, pp. 198–203.

[89] M. Homsi, R. Lutfi, R. M. Carro, and B. Ghias, “A hidden markov model approach to predict students’ actions in an adaptive and intelligent web-based educational system,” in *2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications*, IEEE, 2008, pp. 1–6.

[90] K. E. Boyer *et al.*, “Investigating the relationship between dialogue structure and tutoring effectiveness: A hidden markov modeling approach,” *International Journal of Artificial Intelligence in Education*, vol. 21, no. 1-2, pp. 65–81, 2011.

[91] M. H. Falakmasir, J. P. González-Brenes, G. J. Gordon, and K. E. DiCerbo, “A data-driven approach for inferring student proficiency from game activity logs,” in *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, ACM, 2016, pp. 341–349.

[92] L. Nguyen, “A new approach for modeling and discovering learning styles by using hidden markov model,” *Global Journal of Human Social Science Linguistics & Education*, vol. 13, no. 4, 2013.

[93] T. Käser, N. R. Hallinen, and D. L. Schwartz, “Modeling exploration strategies to predict student performance within a learning environment and beyond,” in *Proceedings of the seventh international learning analytics & knowledge conference*, 2017, pp. 31–40.

[94] A. Bhattacharyya, “On a measure of divergence between two multinomial populations,” *Sankhyā: the indian journal of statistics*, pp. 401–406, 1946.

[95] A. Mašić, E. Polz, and S. Bećirović, “The relationship between learning styles, gpa, school level and gender,” *European Researcher*, vol. 11, no. 1, pp. 51–60, 2020.

[96] C. de Brey *et al.*, “Status and trends in the education of racial and ethnic groups 2018. nces 2019-038.,” *National Center for Education Statistics*, 2019.

[97] S. M. Ross, *Introduction to probability models*. Academic press, 2014.

[98] L. Rabiner and B. Juang, “An introduction to hidden markov models,” *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[99] J. Eisner, “An interactive spreadsheet for teaching the forward-backward algorithm,” in *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics*, 2002, pp. 10–18.

[100] J. Jamesmanoharan, S. H. Ganesh, M. L. P. Felciah, and A. Shafreenbanu, “Discovering students’ academic performance based on gpa using k-means clustering algorithm,” in *2014 World Congress on Computing and Communication Technologies*, IEEE, 2014, pp. 200–202.

[101] M. Shovon, H. Islam, and M. Haque, “An approach of improving students academic performance by using k means clustering algorithm and decision tree,” *arXiv preprint arXiv:1211.6340*, 2012.

[102] O. Ibe, *Markov processes for stochastic modeling*. Newnes, 2013.

[103] G. Marinoni, H. Van’t Land, T. Jensen, *et al.*, “The impact of covid-19 on higher education around the world,” *IAU global survey report*, vol. 23, 2020.

[104] E. M. Aucejo, J. French, M. P. U. Araya, and B. Zafar, “The impact of covid-19 on student experiences and expectations: Evidence from a survey,” *Journal of public economics*, vol. 191, p. 104 271, 2020.

[105] U. of Central Florida. “Graduate student s/u grading policy, url = https://www.ucf.edu/coronavirus/graduate-student-s-u-grading-policy-faqs/.” ().