



UNITED INTERNATIONAL UNIVERSITY

Knowledge Discovery by Mining United International University Student Data

A Thesis Submitted to the Department of Computer Science & Engineering of
United International University

By

S.M. Habibur Rahman Shabu

ID : 011123043

Mohammad Ali

ID : 011123075

Mushfique Ahmed

ID : 011123015

Under the Supervision of
Prof. Dr. Chowdhury Mofizur Rahman
Professor & Pro-Vice Chancellor
United International University
Dhaka-1209, Bangladesh

May 2017

Declaration

We certify that this thesis is our own work, based on our personal study and research under the supervision of Prof. Dr. Chowdhury Mofizur Rahman, Professor & Pro-Vice Chancellor, United International University, Bangladesh. We have acknowledged all material and sources used in its preparation, whether they are books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. We also certify that this thesis has not previously been submitted in any other university.

Habibur Rahman Shabu
ID: 011 123 043

Mohammad Ali
ID: 011 123 075

Mushfique Ahmed
ID: 011 123 015

In my capacity as supervisor of candidate's thesis, I certify that the above statements are true to the best of my knowledge.

Prof. Dr. Chowdhury Mofizur Rahman
Professor & Pro-Vice Chancellor
United International University
Dhaka -1209, Bangladesh.

Approval

The thesis titled “Knowledge Discovery by Mining United International University Student Data” has been submitted to the Department of Computer Science and Engineering, United International University. And it has been accepted as a partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science & Engineering on May, 2017, from the following students and has been accepted in a satisfactory manner.

<u>Student's name:</u>	<u>Id:</u>
S.M. Habibur Rahman Shabu	011 123 043
Mhammad Ali	011 123 075
Mushfique Ahmed	011 123 015

I certify that I read this thesis and that in my opinion it is adequate in scope and quality as a dissertation of the Degree of Bachelor of Computer Science and Engineering.

Prof. Dr. Chowdhury Mofizur Rahman
Professor & Pro-Vice Chancellor
United International University
Dhaka -1209, Bangladesh.

Abstract

The main objective of higher education institutions is to provide quality education to its students. One way to achieve highest level of quality in higher education system is by discovering knowledge for prediction regarding enrolment of students in a particular course, alienation of traditional classroom teaching model, prediction about students' performance and so on. The knowledge is hidden among the educational data set and it is extractable through data mining techniques. Present paper is designed to justify the capabilities of data mining techniques in context of higher education by offering a data mining model for higher education system in the university. In this research, the classification task is used to evaluate student's performance and as there are many approaches that are used for data classification. The student will also get the idea how to take the courses and what will better for them.

By this task we extract knowledge that describes students' performance in end semester. It helps earlier in identifying the dropouts and students who need special attention and allow the teacher to provide appropriate advising/counseling.

Acknowledgements

We would like to thank our thesis supervisor Prof. Dr. Chowdhury Mofizur Rahman, Professor & Pro-Vice Chancellor, United International University, Bangladesh. We fell grateful and wish our profound indebtedness to our honorable supervisor. Deep knowledge and keen interest of our supervisor in the field of Data Mining influenced us to carry out our thesis.

Finally, we would like to thank our parents for their unconditional love and supports. Their love provided us inspiration and was our driving force that has pushed us to complete this dissertation.

Contents

Declaration.....	II
Abstract.	IV
Acknowledgements.....	V
List of Figures.....	VIII
List of Tables.....	VIII
Chapter 1: Introduction	1
Section 1.1: Machine Learning	
Section 1.2: Data Mining	
Section 1.3: Educational Data Mining	
Section 1.4: Objective of the thesis	
Section 1.5: Organization of the thesis	
Chapter 2: Related works.....	4
Section 2.1: Predict Academic Performance	
Section 2.2: Analysis of Student Performance	
Section 2.3: A prediction for Student's Performance Using Classification Method	
Section 2.4: Educational Data Mining & Students' Performance Prediction	
Section 2.5: A prediction for performance improvement using classification	
Section 2.6: Mining Educational Data to Analyze Students' Performance	
Chapter 3: Data Set	8
Section 3.1: Educational Data	
Section 3.2: Seeing Through Data	

Chapter 4: Classifier Model	26
Section 4.1: Linear classifiers	
Section 4.2: Decision tree	
Section 4.3: Naïve Bayes Classifier	
Section 4.4: Random Forest	
Section 4.5: J48 Classifier (C4.5)	
Chapter 5: Experimentatl Results	33
Section 5.1: Attribute Selection	
Section 5.2: Experiments	
Section 5.3: Comparison of Results	
Chapter 6: String matching	38
Section 6.1: fuzzy string matching algorithm	
Section 6.2: Fuzzywuzzy	
Section 6.3: Flowchart	
Section 6.4: Course Data	
Section 6.5: Create String as course sequence	
Section 6.6: Apply String Matching Algorithm	
Section 6.7: Discover Effective String	
Section 6.8: Effective Course Sequence	
Chapter 7: Conclusions and Future Work	49
Section 7.1: Conclusion	
Section 7.2: Future Work	
Reference	51

List of figures

Figure 1.1: Data Mining Discovery Process

Figure 3.1: Number of times courses taken by all students and split with grades

Figure 3.2: Number of times courses taken by Cluster one [$CGPA \geq 3.5$] students and split with grades

Figure 3.3: Number of times courses taken by Cluster two [$3 \leq CGPA < 3.5$] students and split with grades

Fig 3.4: Number of times courses taken by Cluster three [$2.2 \leq CGPA < 3$] students and split with grades

Figure 3.5: Average credit taken per trimester

Figure 3.6: Average credit taken per trimester of without retake students

Figure 3.7: Percentage of three different cluster for former residence Dhaka and outside Dhaka students

Figure 3.8: Percentage of three different cluster for Male and Female students

Figure 3.9: Number of students graduated per trimester

Figure 4.1: A Linear Classification Model.

Figure 4.2: An example of Decision Tree.

Figure 4.3: A Random Forest Model

Figure 4.4: Working procedure Random Forest Tree

List of tables

Table 3.1: Feature Description of Education Dataset.

Table 5.1: Training sets

Table 5.2: Correctly Classified Instance percentage for result prediction

Table 5.3: Correctly Classified Instance percentage for require time prediction

Chapter 1: Introduction

Data mining is the computing process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems ^[1]. It is an interdisciplinary subfield of computer science. The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use. The data collected from different applications require proper method of extracting knowledge from large repositories for better decision making. Knowledge discovery in databases (KDD), often called data mining, aims at the discovery of useful information from large collections of data ^[2]. The main functions of data mining are applying various methods and algorithms in order to discover and extract patterns of stored data ^[3]. Data mining and knowledge discovery applications have got a rich focus due to its significance in decision making and it has become an essential component in various organizations. Educational Data Mining is an emerging discipline, concerned with developing methods for exploring the unique and increasingly large-scale data that come from educational settings, and using those methods to better understand students, and the settings which they learn in.

There are increasing research interests in using data mining in education. This new emerging field, called Educational Data Mining, concerns with developing methods that discover knowledge from data originating from educational environments ^[4]. Educational Data Mining uses many techniques such as Decision Trees, Neural Networks, Naive Bayes, K- Nearest neighbor, J48 and many others.

Using these techniques many kinds of knowledge can be discovered such as association rules, classifications and clustering. The discovered knowledge can be used for prediction regarding enrolment of students in a particular course, choosing best courses for students, sequence matching courses divide different category, prediction about students' performance and so on.

Section 1.1: Machine Learning

Machine learning is the subfield of computer science that, gives computers the ability to learn without being explicitly programmed ^[5]. Evolved from the study of pattern recognition and computational learning theory in artificial intelligence ^[6], machine learning explores the study and construction of algorithms that can learn from and make predictions on data ^[7] - such algorithms overcome following strictly static program instructions by making data-driven predictions or decisions ^[8], through building a model from sample inputs. Machine learning is used in Web search, spam filters, recommender systems, ad placement, credit scoring, fraud detection, stock trading, drug design, and many other applications.

Resurging interest in machine learning is due to the same factors that have made data mining and Bayesian analysis more popular than ever. Things like growing volumes and varieties of available data, computational processing that is cheaper and more powerful, and affordable data storage. All of these things mean it's possible to quickly and automatically produce models that can analyze bigger, more complex data and deliver faster, more accurate results – even on a very

large scale. And by building precise models, an organization has a better chance of identifying profitable opportunities – or avoiding unknown risks.

Section 1.2: Data Mining

Data mining is the computing process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems ^[9]. It is an interdisciplinary subfield of computer science ^[10]. The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use. Data mining is the analysis step of the "knowledge discovery in databases" process or (KDD) ^[11].

Data is been increasing day by day. Whatever we are doing whether liking a picture on Facebook or sending an e-mail, it generates data. That's how we have these days huge amount of data. Data mining is the process of finding hidden information and patterns from a huge database. Alternatively, it has been called exploratory data analysis, data driven discovery and deductive learning. Mainly in data mining we try to extract patterns or information from a dataset and build an understandable structure by transforming the dataset. It is a challenge of data mining to choose the architecture for working with dataset because it varies with dataset. Most of the data mining works follow a basic architecture.

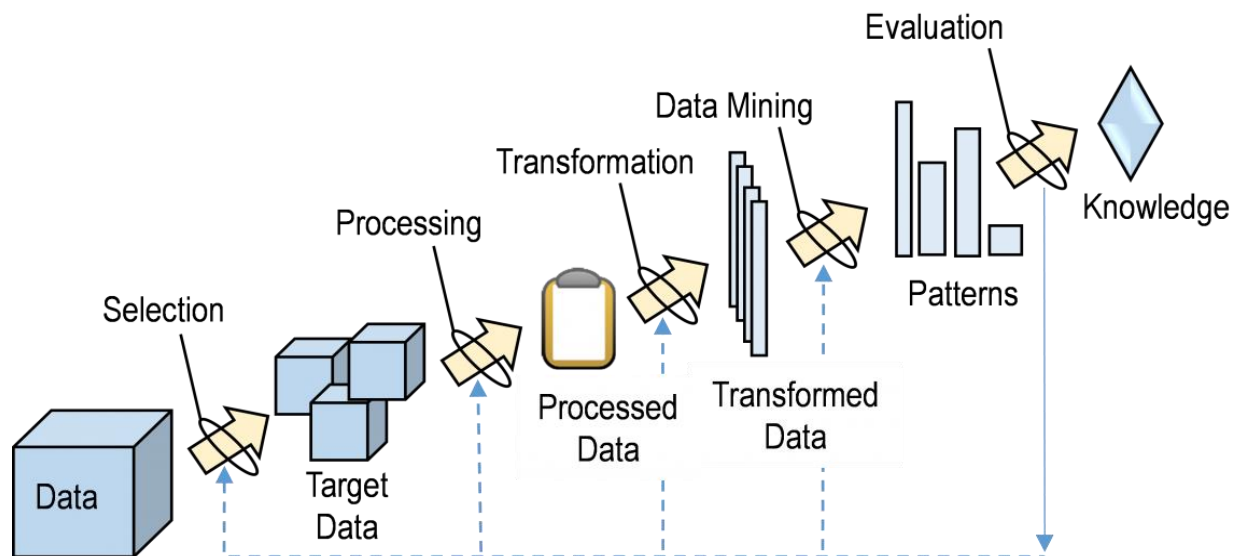


Figure 1.1: Data Mining Discovery Process

Section 1.3: Educational Data Mining

Educational Data Mining is an emerging discipline, concerned with developing methods for exploring the unique and increasingly large-scale data that come from educational settings, and using those methods to better understand students, and the settings which they learn in.

Whether educational data is taken from students' use of interactive learning environments, computer-supported collaborative learning, or administrative data from schools and universities, it often has multiple levels of meaningful hierarchy, which often need to be determined by properties in the data itself, rather than in advance. Issues of time, sequence, and context also play important roles in the study of educational data.

Section 1.4: Objective of the thesis

The main objective of this paper is to use data mining methodologies to study student's performance and knowledge discovery of United International University student data. Data mining provides many tasks that could be used to study the student performance. In this research, the classification task is used to evaluate student's performance and as there are many approaches that are used for data classifier model, decision tree method, Levenshtein distance algorithm is used here. Information's like gender, semester results, residence data were collected from the student's management system, to predict the performance at the end of the semester. This paper investigates the exploring the data of United International University student's.

Section 1.5: Organization of the thesis

This thesis organized as follows:

Chapter 2 describes the related papers that we have studied, which actually build up the importance of mining educational data. The chapter presented their work in different sections.

Chapter 3 describes the information and structure of our collected dataset. Here, we've also discussed about data visualization and showed some graphical representation of the data.

Chapter 4 provides a brief introduction of classifier models and describes different types of classifier models that have been used in machine learning to mine information. Especially the chapter focuses on those regression models which we have applied in our research and can be useful in the related field.

Chapter 5 aims to describe the experimental results that we've found by using some of our dataset. The results that we've found demonstrated through some tables to understand the result comparison.

Chapter 6 describes effective Course Sequence and we've found a effective string.

Chapter 7 presents the conclusion, summarizes the thesis contribution and discusses the future work.

Chapter 2: Related works

In recent years, various types of data mining techniques are being applied for mining educational data. Many researchers are involving themselves to the contribution of educational data mining and so many others for improvement of the modern world. The research papers that we have studied with the necessary information will be discussed in this chapter.

Section 2.1: Predict Academic Performance

Elizabeth León Guzman, Fabio Augusto González Osorio and their research group midas ^[12] in 2013 present research proposes an approach to Educational Data Mining at the Universidad Nacional de Colombia through the definition of models that integrate clustering and classification techniques to analyze academic data, corresponding to the students who joined the University to the programs of Agricultural and Computer and Systems Engineering between 2007-03 and 2012-01. These techniques are intended to acquire a better understanding of the attrition during the first enrollments and to assess the quality of the data for the classification task, which can be understood as the prediction of the loss of academic status due to low academic performance. Different models were built to predict the loss of academic status in different scenarios such as: in the first four enrollments regardless when; at a specific academic period using only the admission process data and then, using academic records. Experimental results show that the prediction of the loss of academic status is improved when adding academic data. Two algorithms, Naïve Bayes and a decision tree, were used to create classification models to predict the loss of academic status due to low of academic performance. Several models were

learned to test the configuration. Naïve Bayes results were better on the test set. The decision trees results were more consistent regarding that subject making it more reliable when testing on new data. The classification results showed similar values to works reported in the literature using similar datasets.

Section 2.2: Analysis of Student Performance

Dr. Syed Akhter Hossain and his research team ^[13] aim of this thesis paper is to analyze student performance using data mining in 2014. Data mining is the process of prediction, extracting data. Prediction regarding student performance can help a student to take decision. It can help not only the current students but also the future students, to take decision. In this way they can avoid poor performance which will help to enhance their performance. This is also a guideline to take decision. To understand student performance, a survey was conducted by Military Institute of Science Technology with the support from the CSE department and the peer learners of different classes. The data collected from survey was normalized, validated and revalidated. After thorough investigation on the survey data, based on statistical analysis techniques, different observations were recorded in the form of graphical illustration in order to find the relations. The experimental analysis of the data through result from the survey was satisfactory which led towards further study. In order to proceed further through data mining based on the understanding of the survey, data was collected form the central database of MIST where the main aim was to relate CGPA and student performance. They investigated different properties of the data; collected and developed a classification hypothesis in order to apply data mining algorithms. The experimental results are validated against test data and interesting co-relations are observed. In the future further rigorous study to match between demographic data and academic data will lead to much determining factors in order to predict the student performance. The final step to validating classification tree, which is to run our test set through the model and ensure the accuracy of the model when evaluating the test set is not too different from the training set. For validating data they use 2 fold cross validation. Comparing the "Correctly Classified Instances" from the test set (61.9 percent) with the "Correctly Classified Instances" from the training set (91.6 percent).

SECTION 2.3: A prediction for Student's Performance Using Classification Method

Abeer Badr El Din Ahmed, Ibrahim Sayed Elaraby^[14] their main objective of this paper is to use data mining methodologies to study student's performance in end General appreciation. It aims at the discovery of useful information from large collections of data ^[15]. The main functions of data mining are applying various methods and algorithms in order to discover and extract patterns of stored data ^[16]. Data mining provides many tasks that could be used to study the student performance. In this research, the classification task is used to evaluate student's performance and as there are many approaches that are used for data classification, the decision tree method is used here. The data set used in this study was obtained from a student's database used in one of the educational institutions, on the sampling method of Information system department from session 2005 to 2010. Decision tree method is used on student's database to predict the student's performance on the basis of student's database. They use some attribute were

collected from the student's database to predict the final grade of student's. This study will help the student's to improve the student's performance, to identify those students which needed special attention to reduce failing ration and taking appropriate action at right time.

SECTION 2.4: Educational Data Mining & Students' Performance Prediction

Amjad Abu Saa^[17] analyses students' data and information to classify students, or to create decision trees or association rules, to make better decisions or to enhance student's performance is an interesting field of research, which mainly focuses on analyzing and understanding students' educational data that indicates their educational performance, and generates specific rules, classifications, and predictions to help students in their future educational performance.

Classification is the most familiar and most effective data mining technique used to classify and predict values. Educational Data Mining (EDM) is no exception of this fact, hence, it was used in this research paper to analyze collected students' information through a survey, and provide classifications based on the collected data to predict and classify students' performance in their upcoming semester. The objective of this study is to identify relations between students' personal and social factors, and their academic performance. This newly discovered knowledge can help students as well as instructors in carrying out better enhanced educational quality, by identifying possible underperformers at the beginning of the semester/year, and apply more attention to them in order to help them in their education process and get better marks. In fact, not only underperformers can benefit from this research, but also possible well performers can benefit from this study by employing more efforts to conduct better projects and research through having more help and attention from their instructors.

Section 2.5: Data Mining: A prediction for performance improvement using classification

Brijesh Kumar Bhardwaj and Saurabh Pal^[18] developed generating of a data source of predictive variables, identifying of different factors, which effects a student's learning behavior and performance during academic career. In this research Bayesian classification method is used on student database to predict the students division on the basis of previous year database. This study will help to the students and the teachers to improve the division of the student. This study will also work to identify those students which needed special attention to reduce failing ration and taking appropriate action at right time. Their result was found that the students' performance is highly dependent on their grade obtained in Senior Secondary Examination. it is also found that the second high potential variable for students' performance is their living location.

Section 2.6: Mining Educational Data to Analyze Students' Performance

Baradwaj and Pal^[19] conducted a research on a group of 50 students enrolled in a specific course program across a period of 4 years (2007-2010), with multiple performance indicators, including "Previous Semester Marks", "Class Test Grades", "Seminar Performance", "Assignments", "General Proficiency", "Attendance", "Lab Work", and "End Semester Marks". They used ID3

decision tree algorithm to finally construct a decision tree, and if-then rules which will eventually help the instructors as well as the students to better understand and predict students' performance at the end of the semester. Furthermore, they defined their objective of this study as: "This study will also work to identify those students which needed special attention to reduce fail ration and taking appropriate action for the next semester examination". Baradwaj and Pal selected ID3 decision tree as their data mining technique to analyze the students' performance in the selected course program; because it is a "simple" decision tree learning algorithm.

Chapter 3: Data Set

Section 3.1: Educational Data

The data we used is collected from United International University. The data we collected from year 2003 to 2017. Initially there was data for all undergraduate completed students. We have used the data of non-transferred CSE students. The used data included student's course history with 8 attributes and 15000 instances and basic info with 4 attributes for all 238 students.

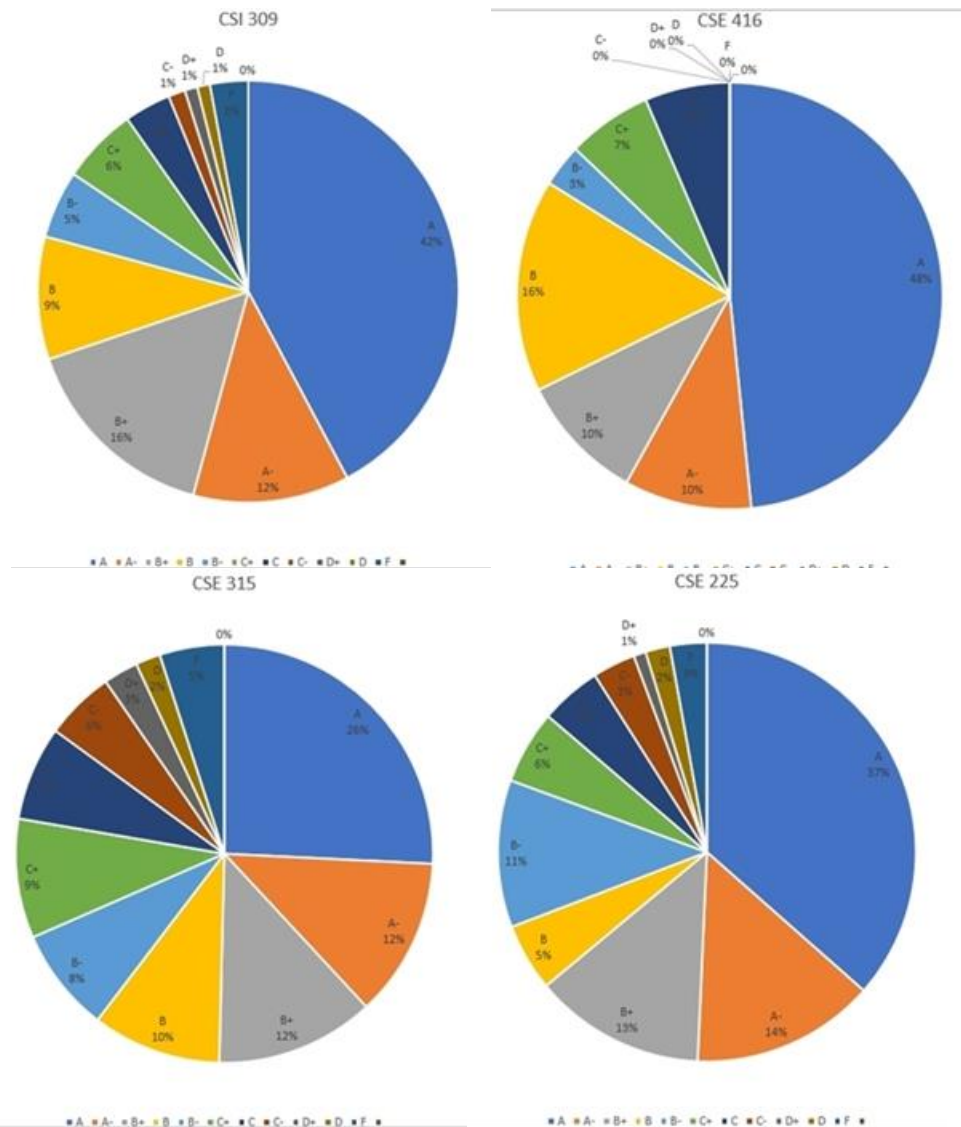
Name	Type	Description
STUDENTID	Nominal	Students unique ID
COURSEID	Nominal	COURSEID of Courses taken by all students
TRIMESTER	Nominal	Courses taken by students in which trimester
GRADE	Nominal & Numeric	Grade and grade point for every course taken by each student
CREDIT	Numeric	Credit hour for each course
ISTRANSFER	Nominal	Is Student transferred from another institute 'YES' or 'NO'
REQ_TRIMESTER	Numeric	No of trimester required for a student to complete graduation
RETAKE_NUMBER	Numeric	No of times course retake/repeat by each student
CGPA	Numeric	Final CGPA of all student
GENDER	Nominal	Male or Female student
PERMANENTADDR	Nominal	Dhaka/Other area
PRESENTADDR	Nominal	Dhaka/Other area

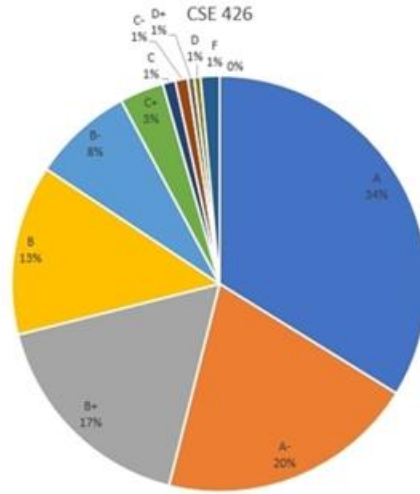
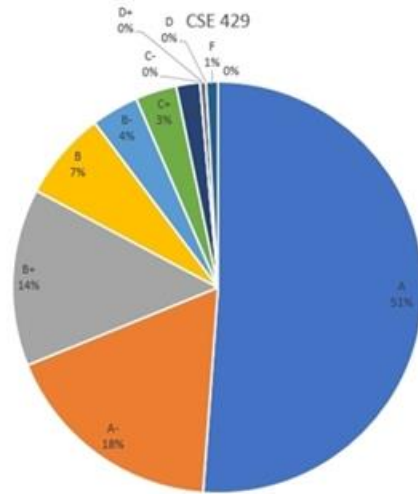
Table 3.1: Feature Description of Education Dataset.

Section 3.2: Seeing through Data

Presenting Data in a pictorial or graphical format is known as data visualization. We have followed several methods to visualize our data. We have plotted pie chart, clustering, percentage for our attributes or features.

Pie Chart: Pie Chart of some of the important features are shown below:



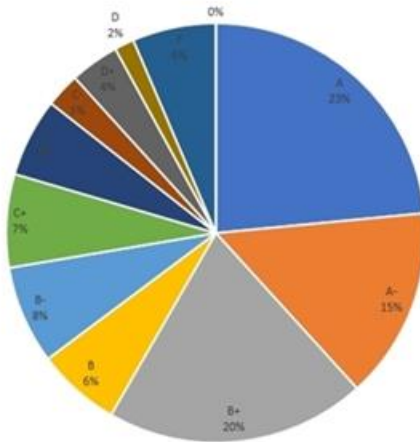


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

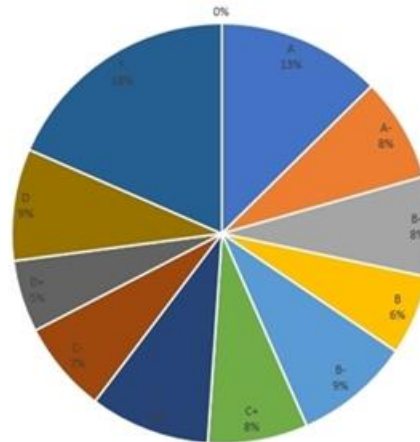
CSI 311

■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

MATH 151

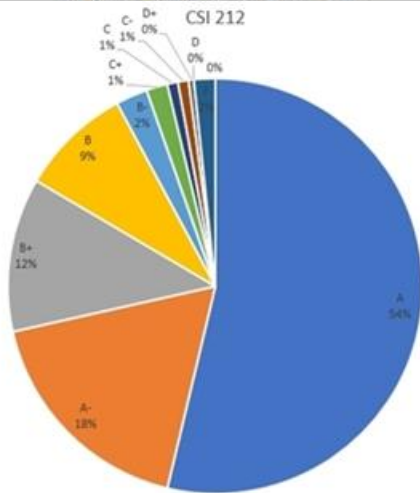


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

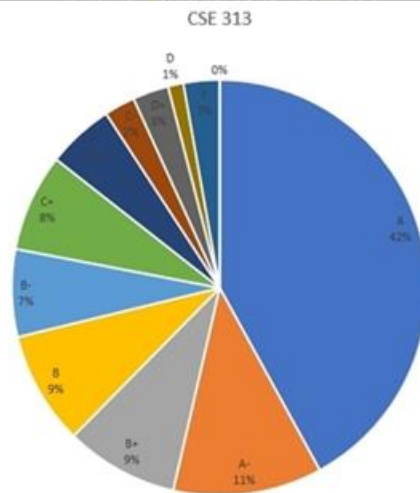


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

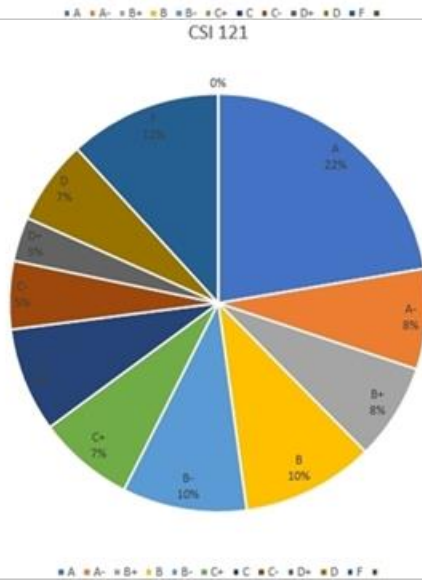
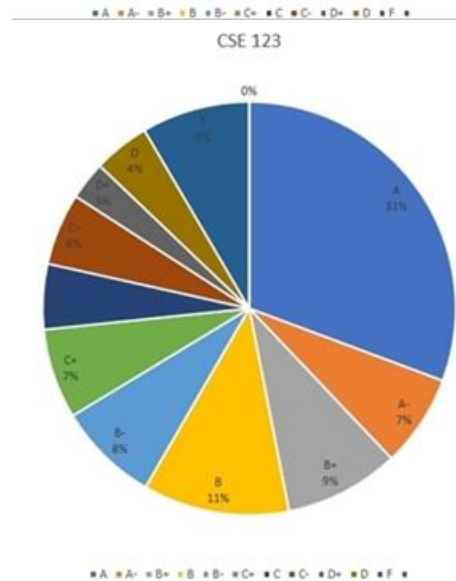
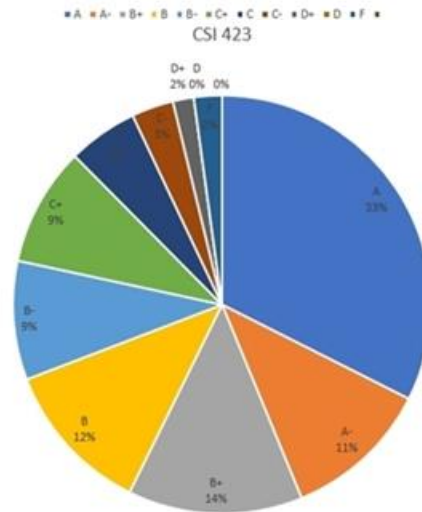
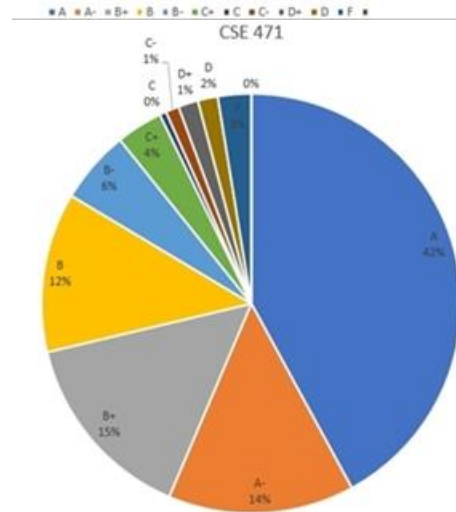
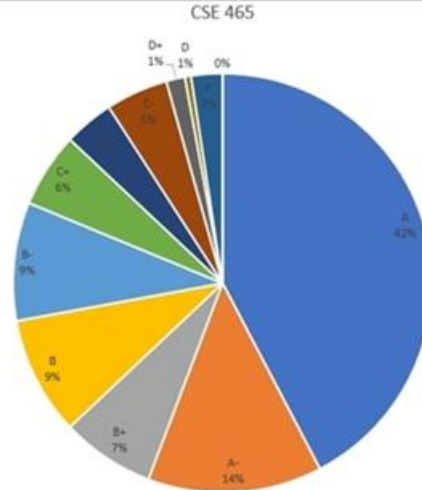
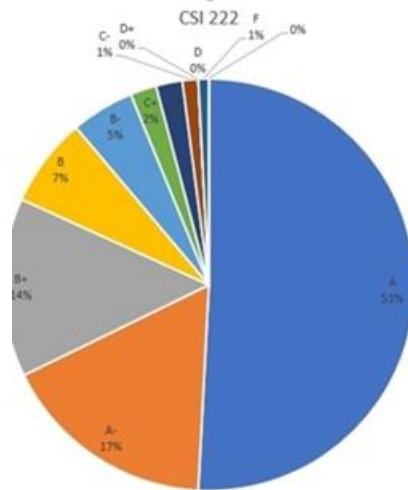
CSE 313

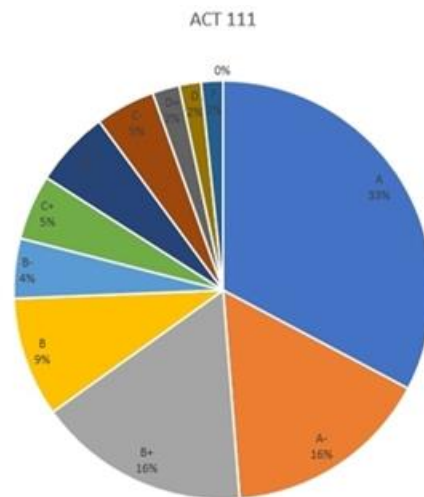
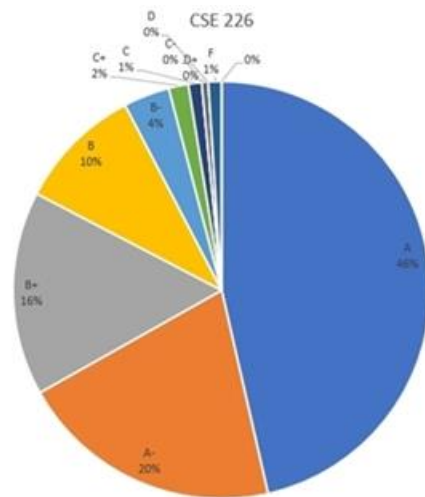


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

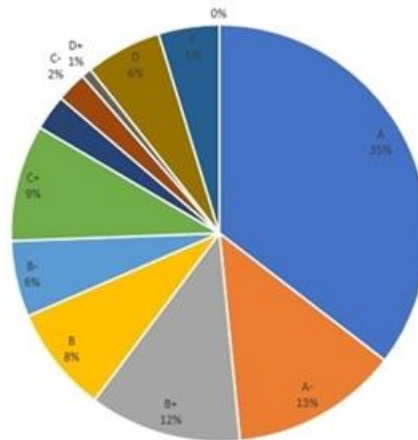


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

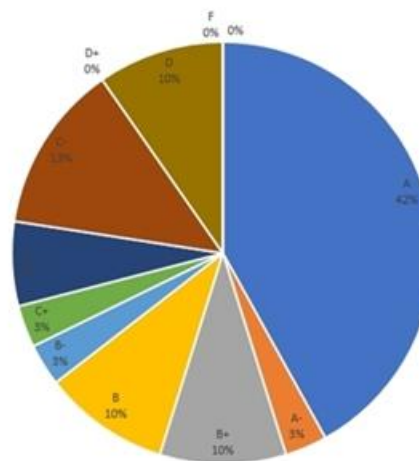




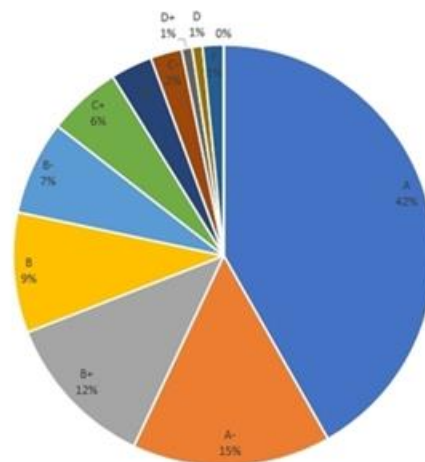
CSI 218



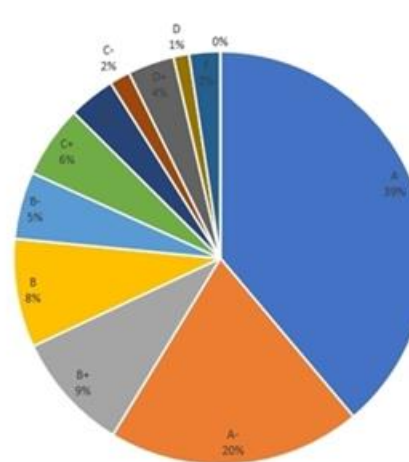
CSI 412



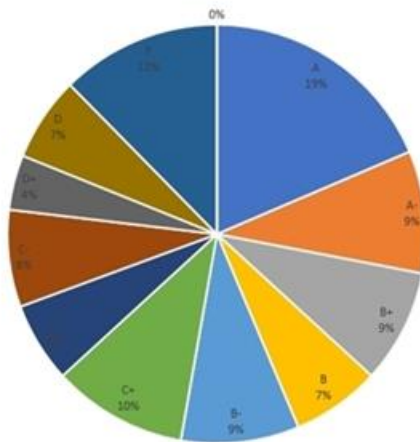
CSI 219



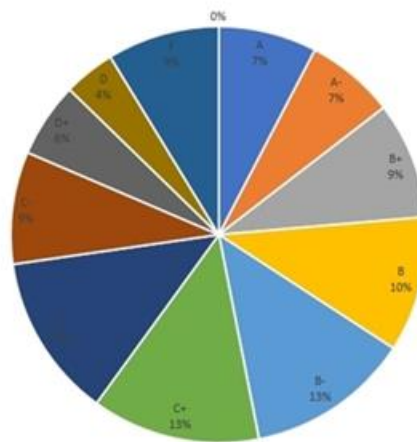
CSI 322



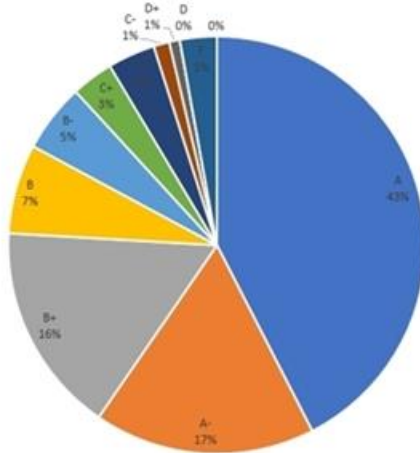
MATH 183



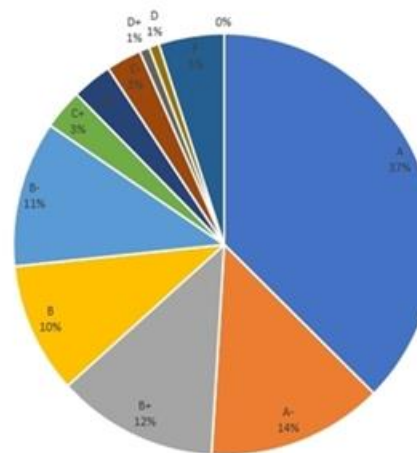
ENG 101



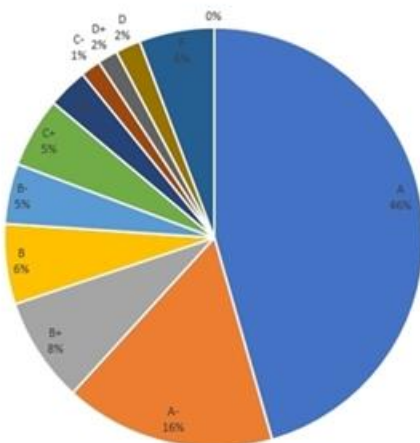
CSI 321



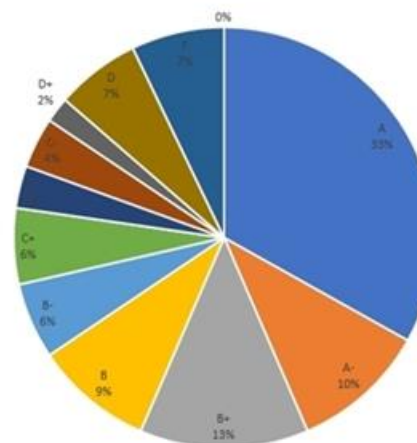
CSI 221

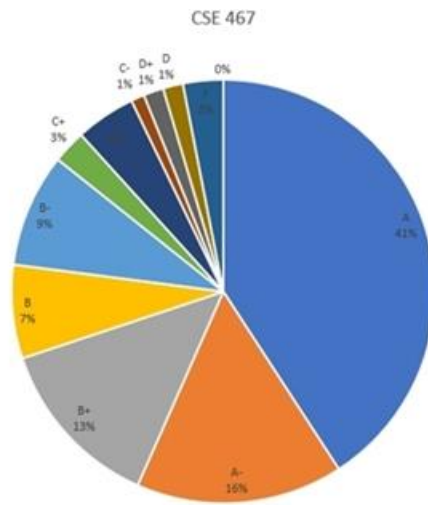


CSI 211



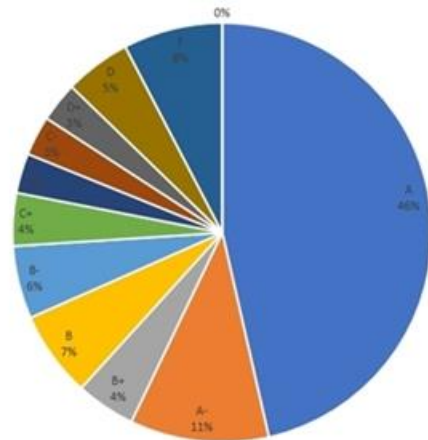
CSI 422





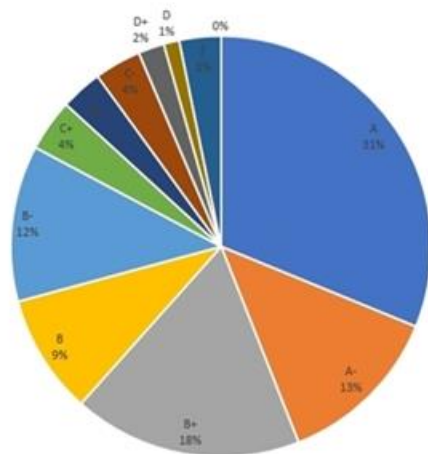
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSI 310

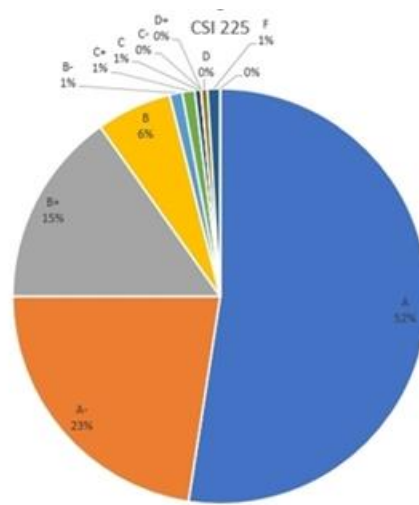


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSI 228

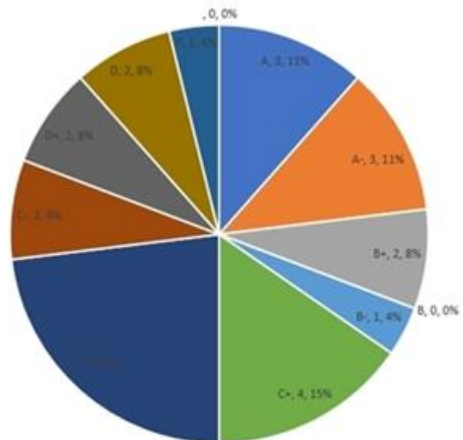


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■



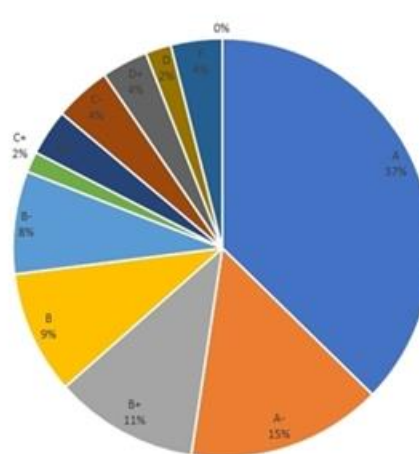
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSE 477

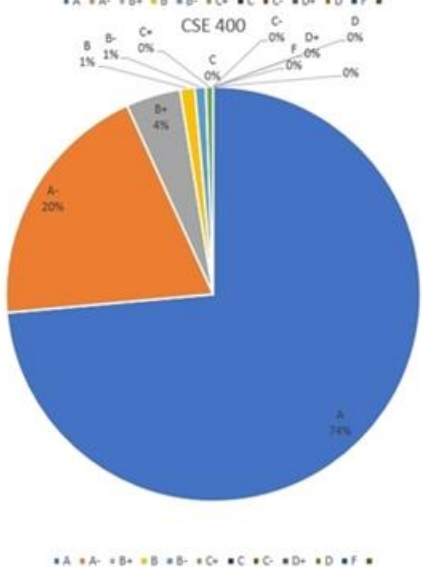
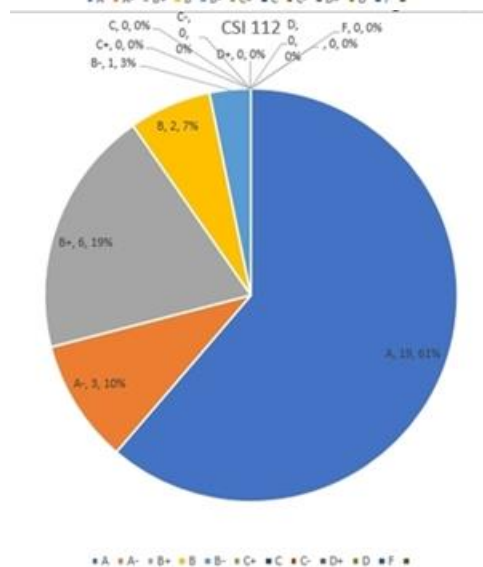
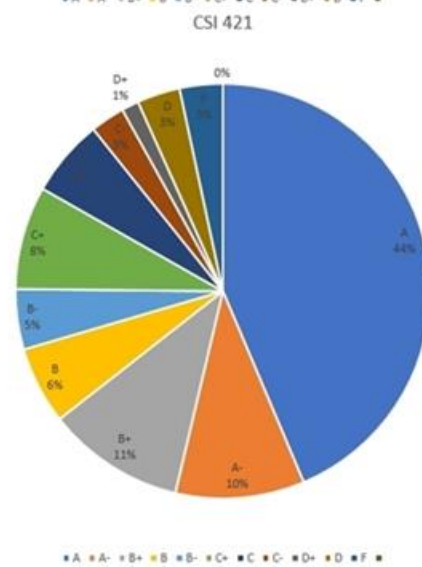
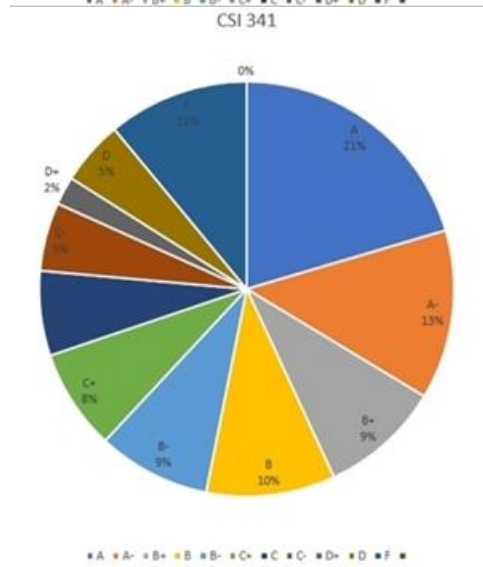
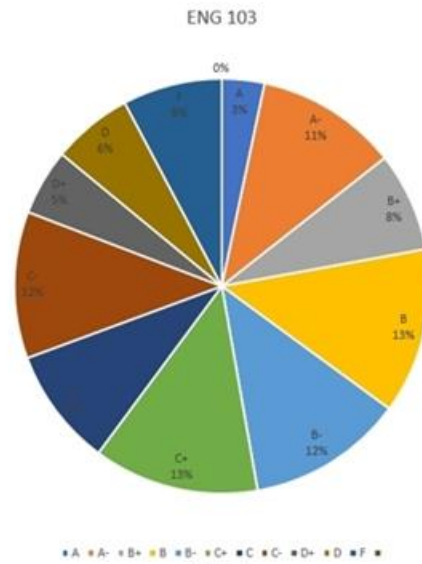
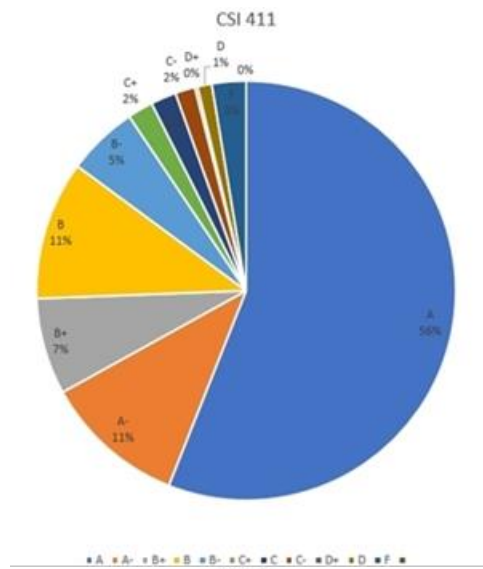


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

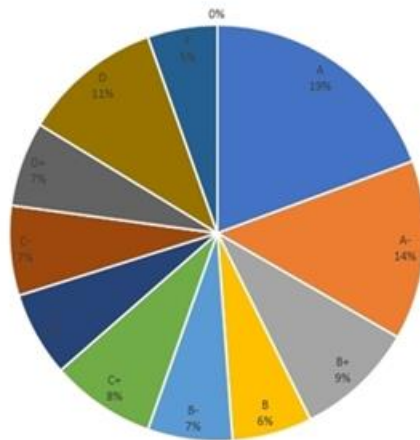
CSE 324



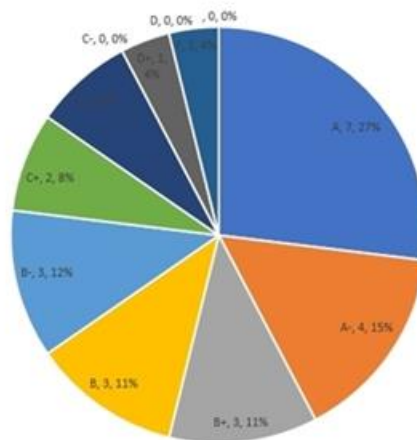
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■



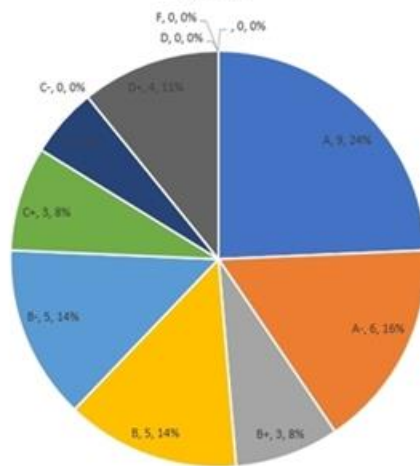
MATH 187



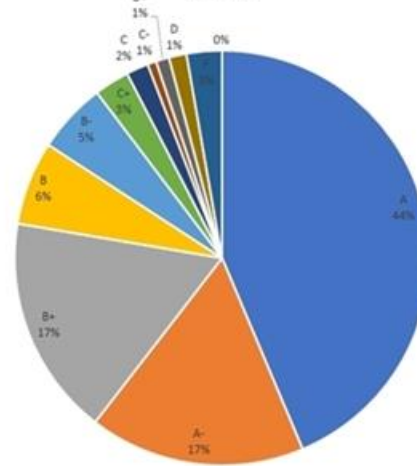
CSE 461



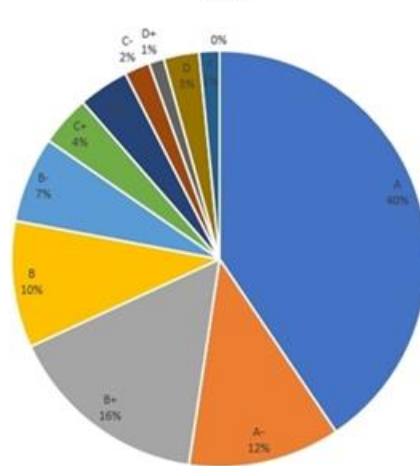
SOC 101



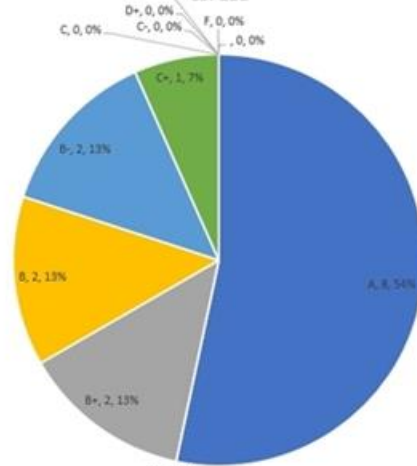
CSE 124



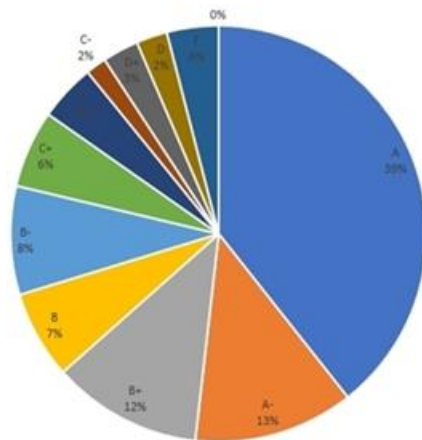
CSI 217



CSI 111

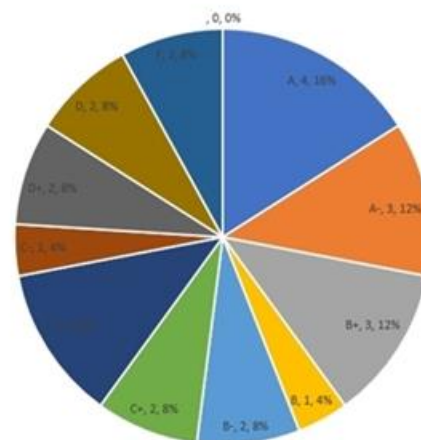


ECO 213



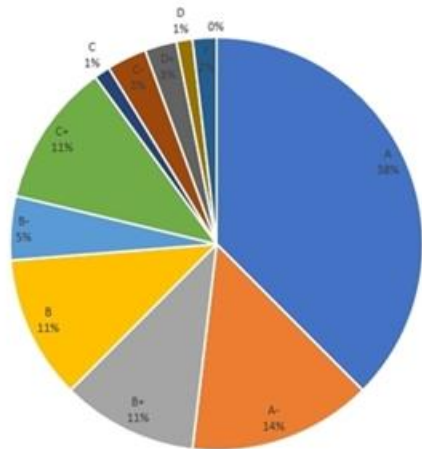
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

PHY 101



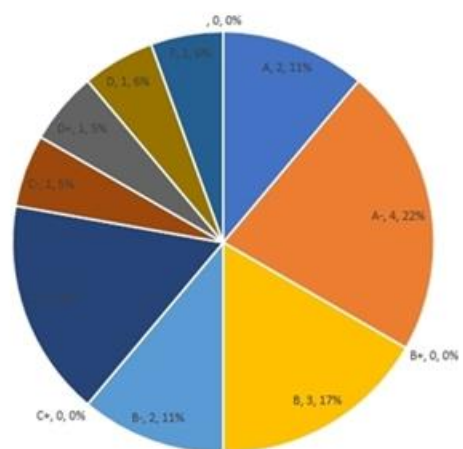
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSI 416



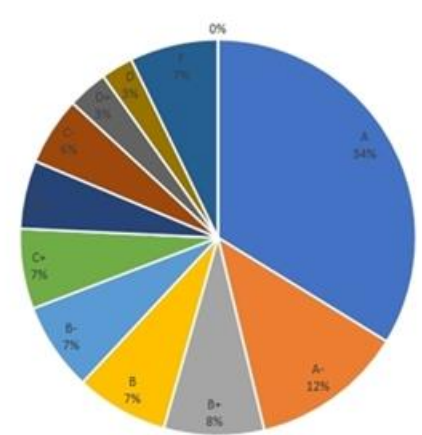
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSI 124



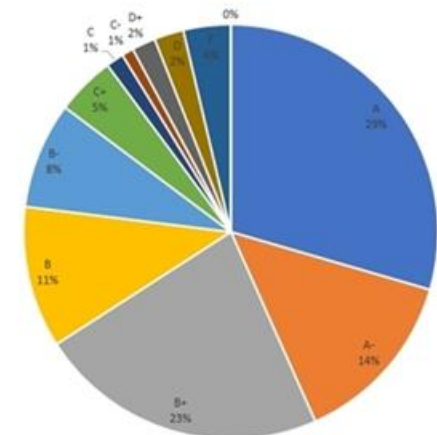
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSE 425



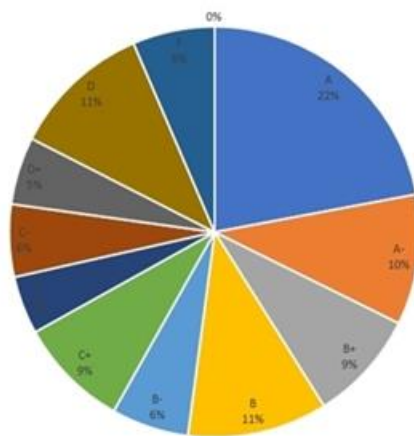
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSI 342



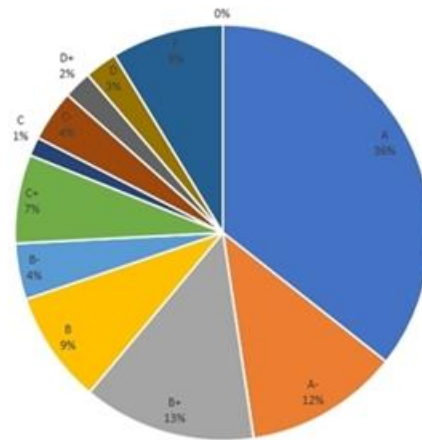
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSE 323



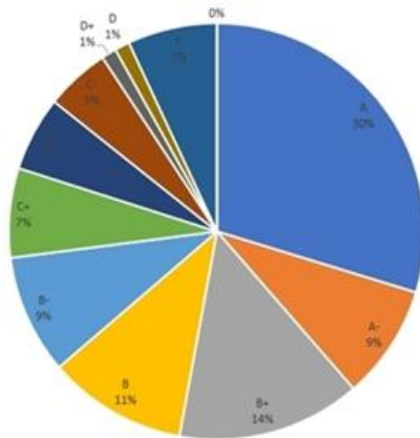
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSI 122



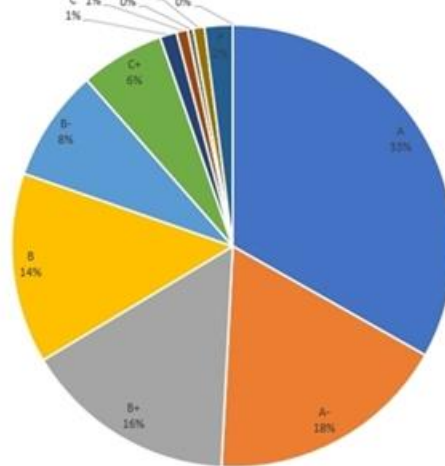
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

STAT 205



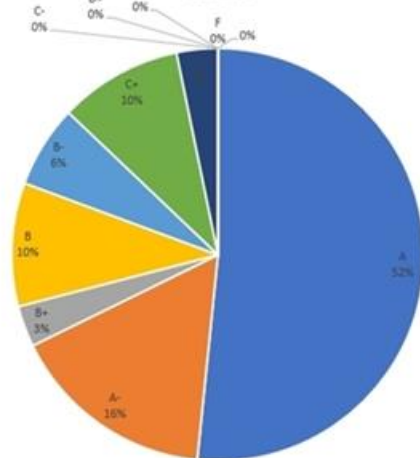
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSE 236



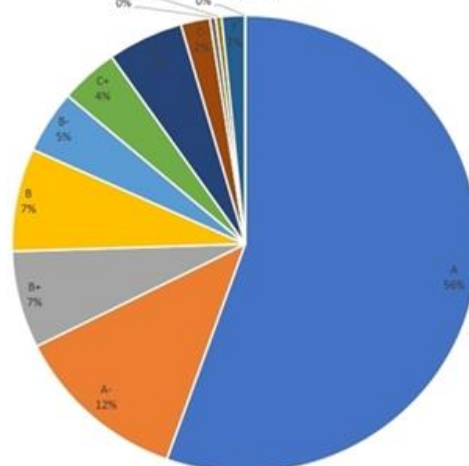
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

CSE 415

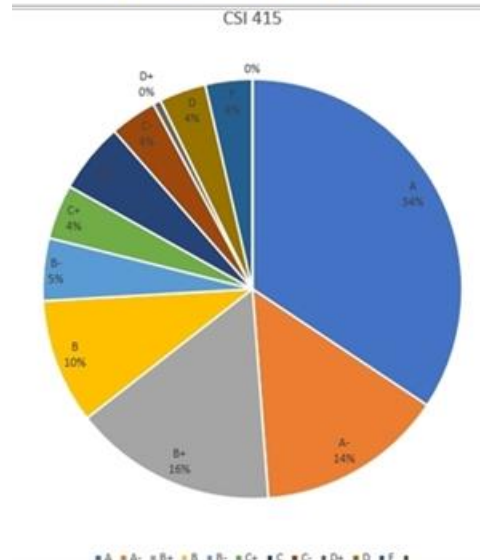
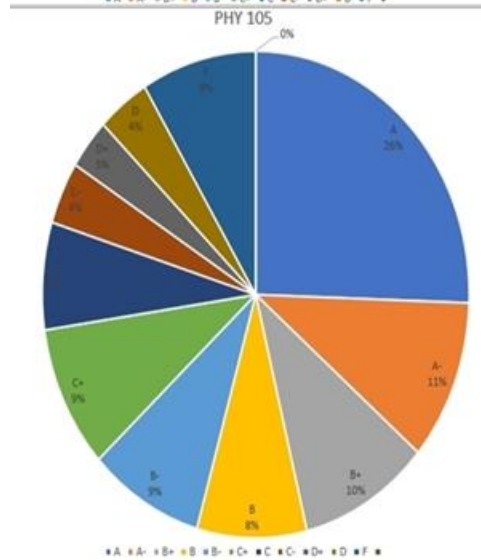
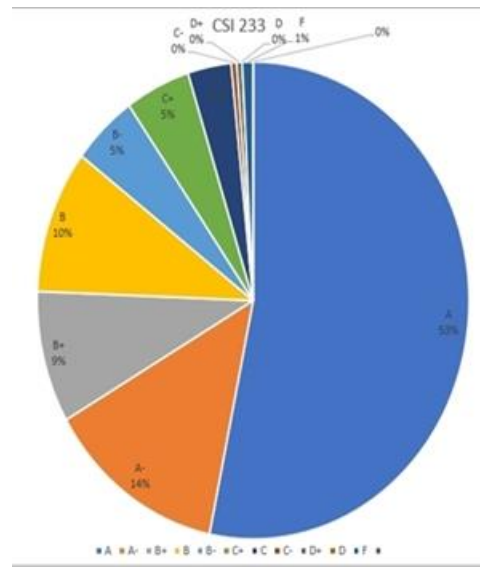
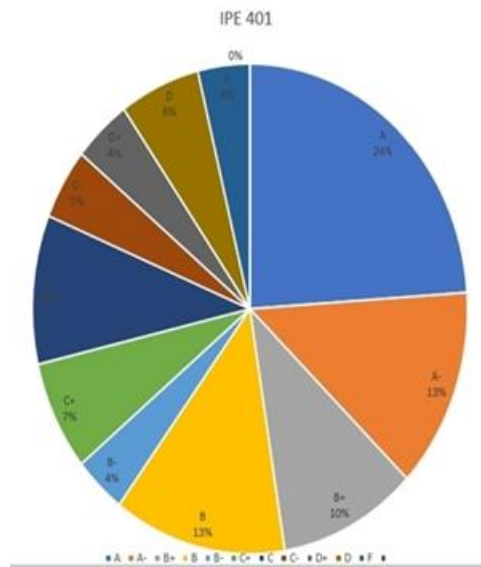


■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■

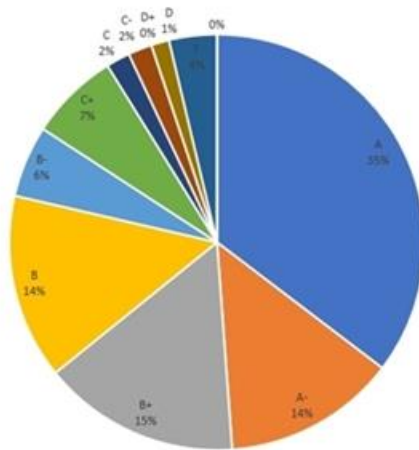
CSI 229



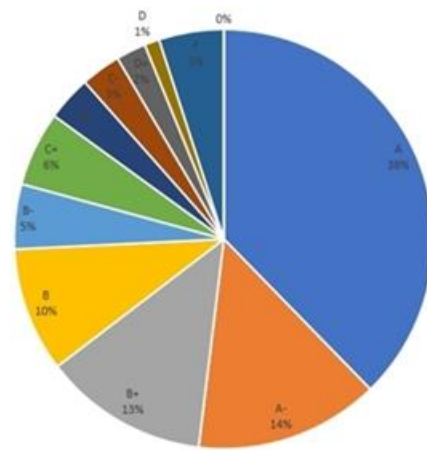
■ A ■ A- ■ B+ ■ B ■ B- ■ C+ ■ C ■ C- ■ D+ ■ D ■ F ■



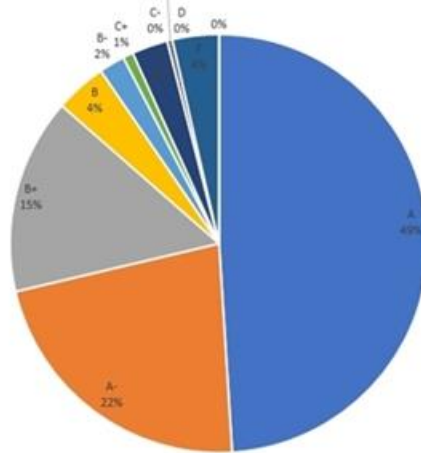
PHY 106



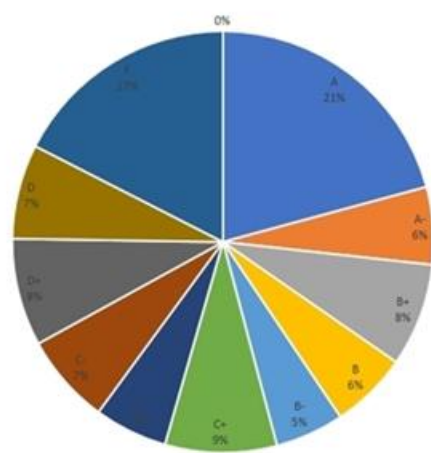
CSI 227



CSI 312



MATH 201



Histogram: Histograms of some of the important features are shown below:

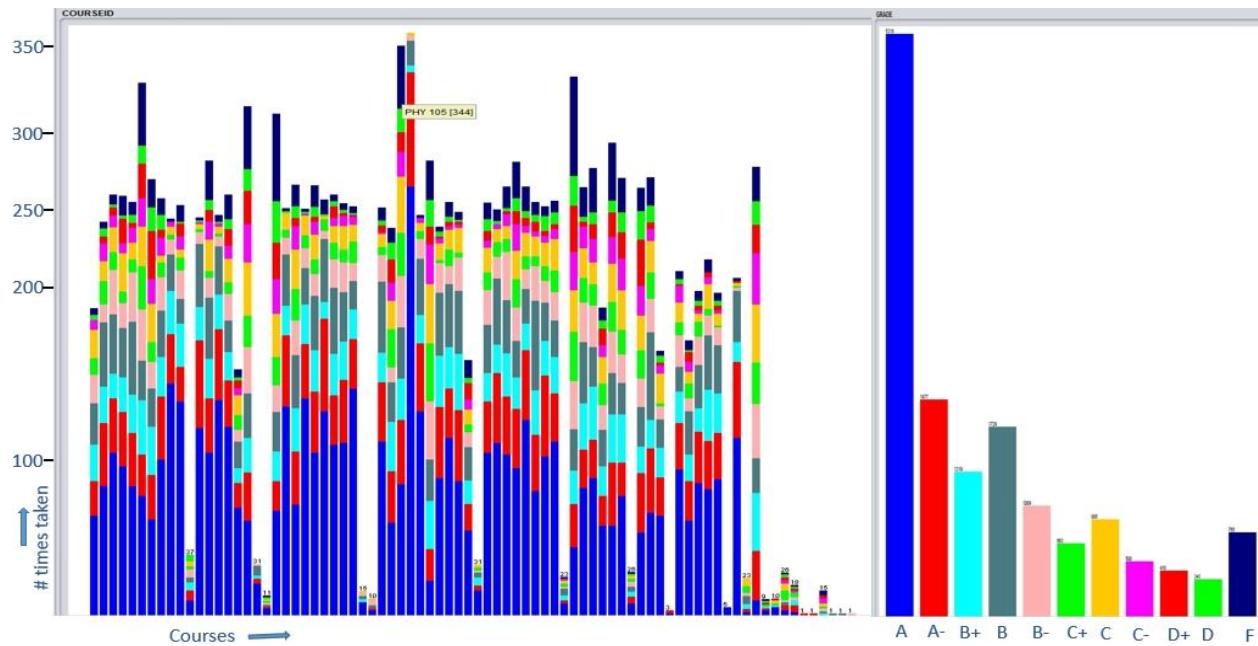


Figure 3.1: Number of times courses taken by all students and split with grades

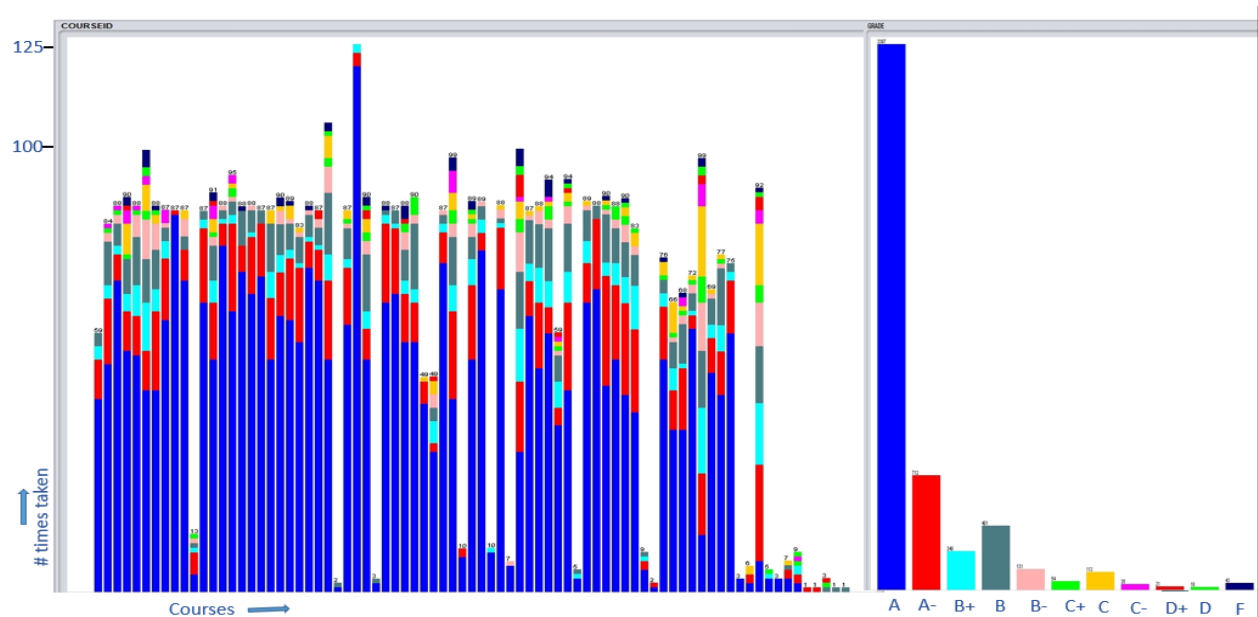


Figure 3.2: Number of times courses taken by Cluster one[CGPA>=3.5] students and split with grades

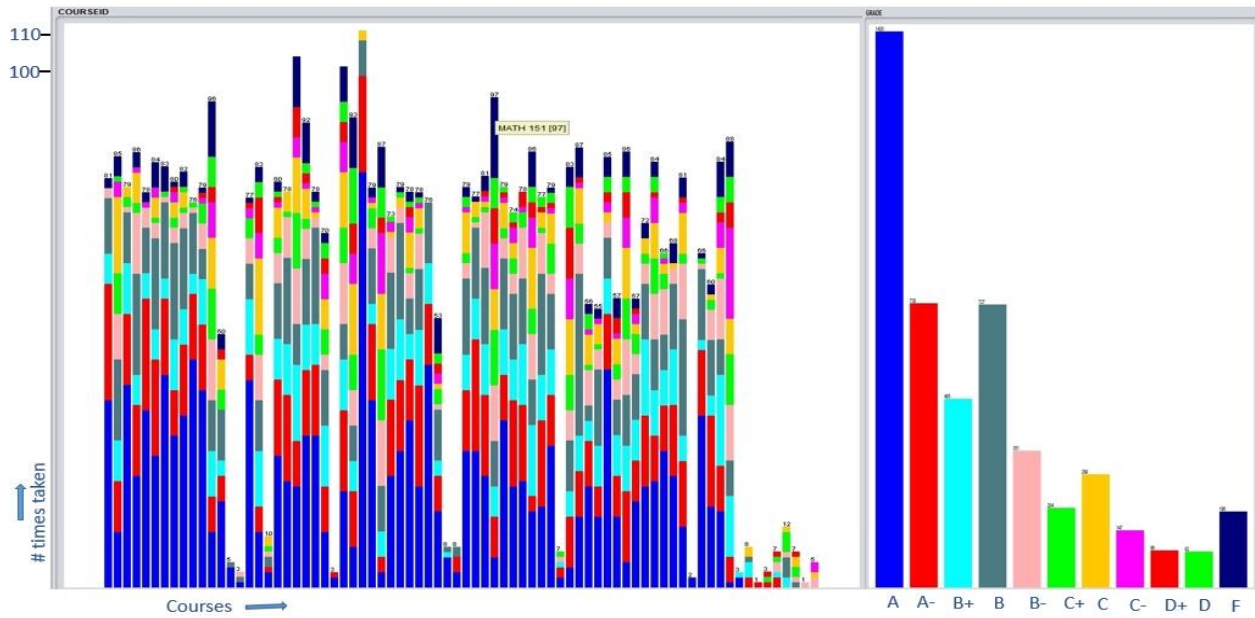


Figure 3.3: Number of times courses taken by Cluster one [$3 \leq \text{CGPA} < 3.5$] students and split with grades

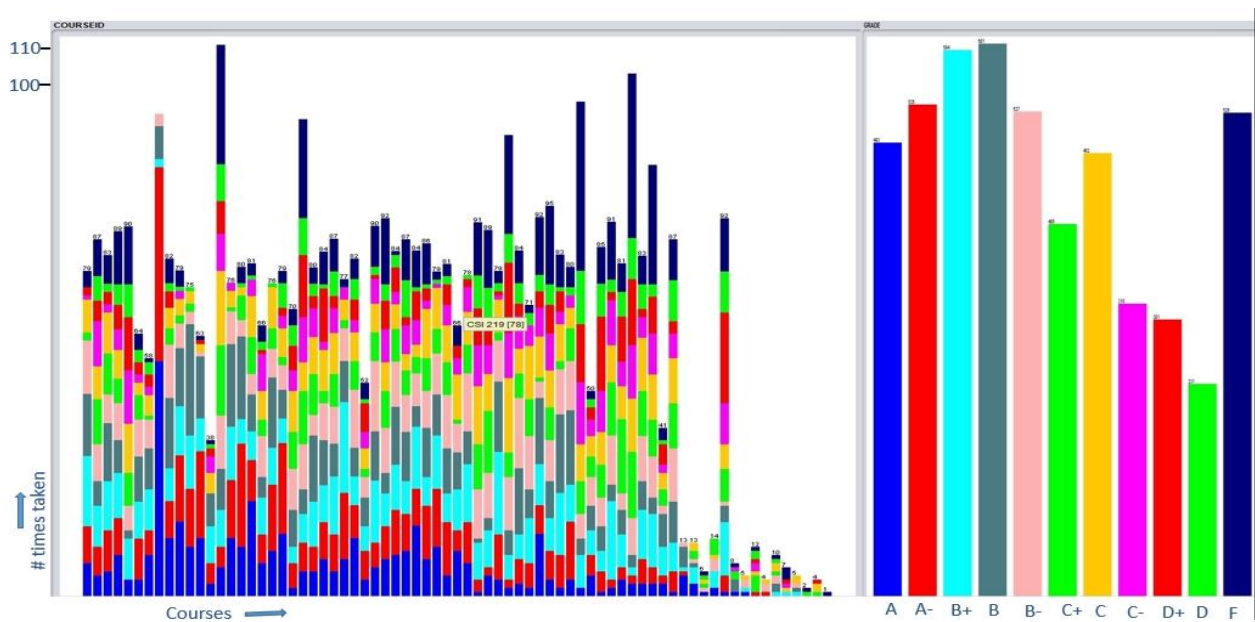


Figure 3.4: Number of times courses taken by Cluster one [$2.2 \leq \text{CGPA} < 3$] students and split with grades

Comparison Graph: Comparison graph are given below:

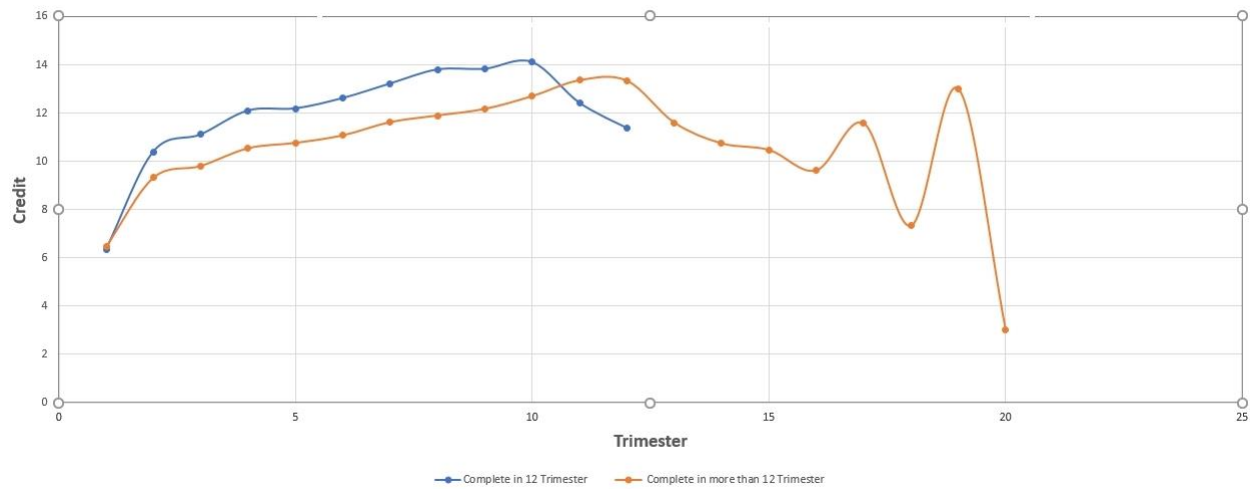


Figure 3.5: Average credit taken per trimester

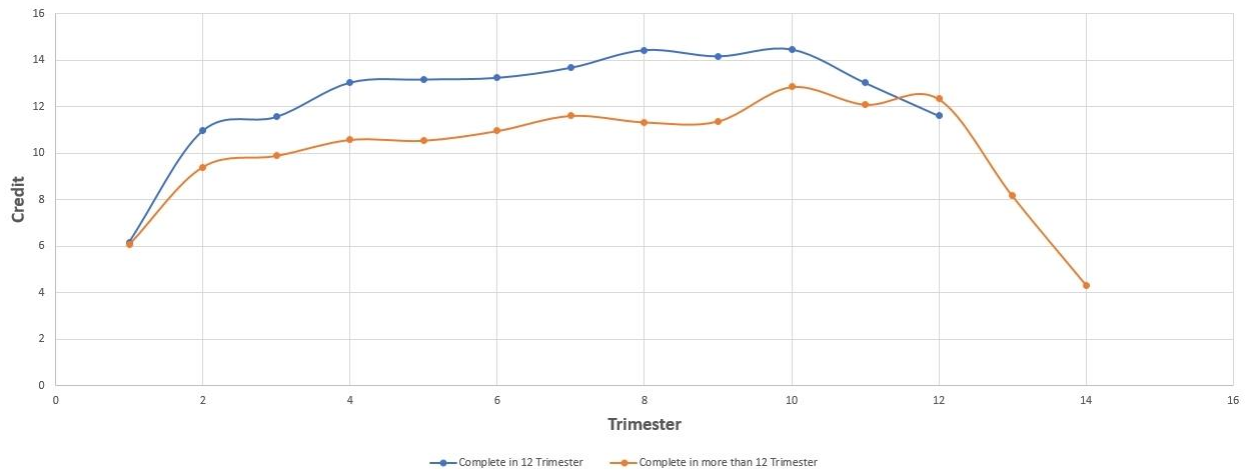


Figure 3.6: Average credit taken per trimester of without retake students

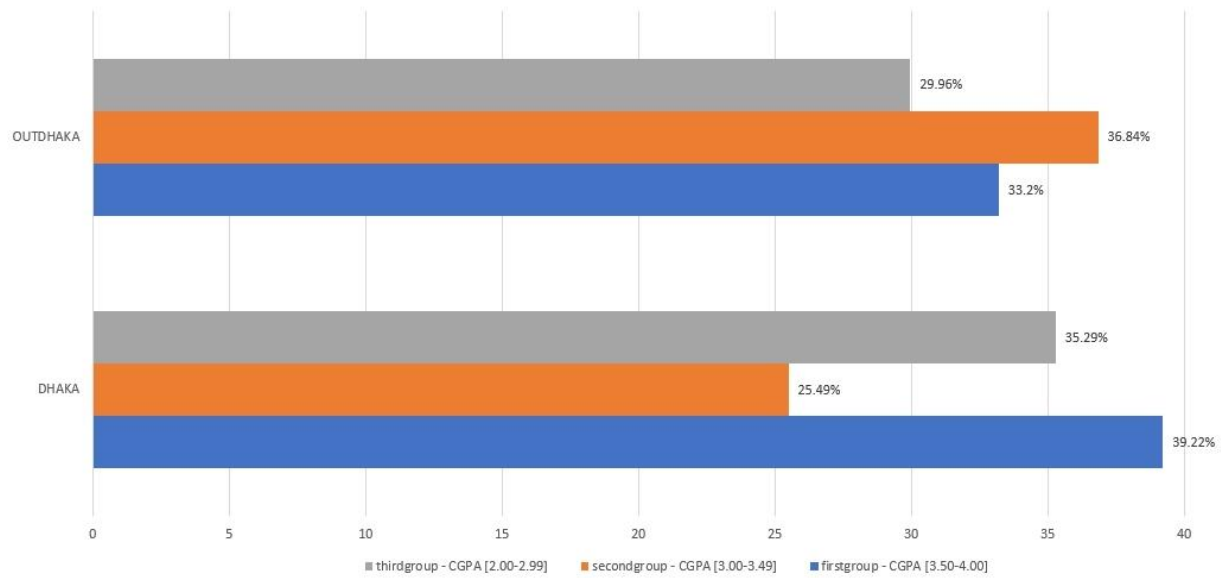


Figure 3.7: Percentage of three different cluster for former residence Dhaka and outside Dhaka students

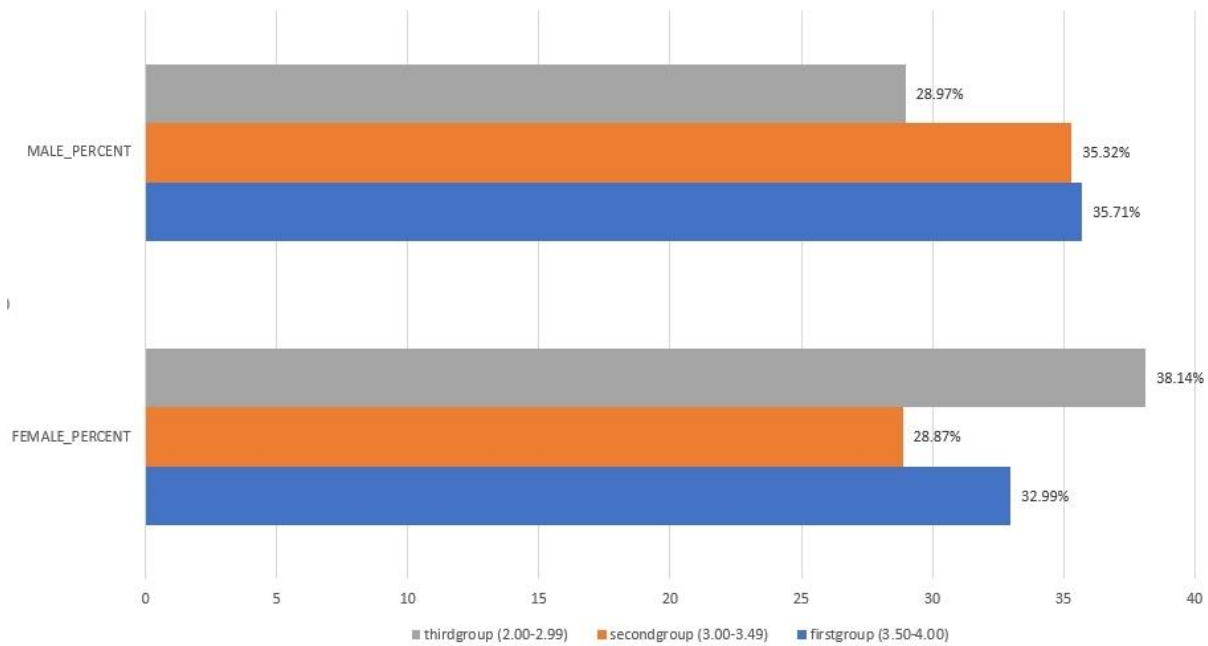


Fig 3.8: Percentage of three different cluster for Male and Female students

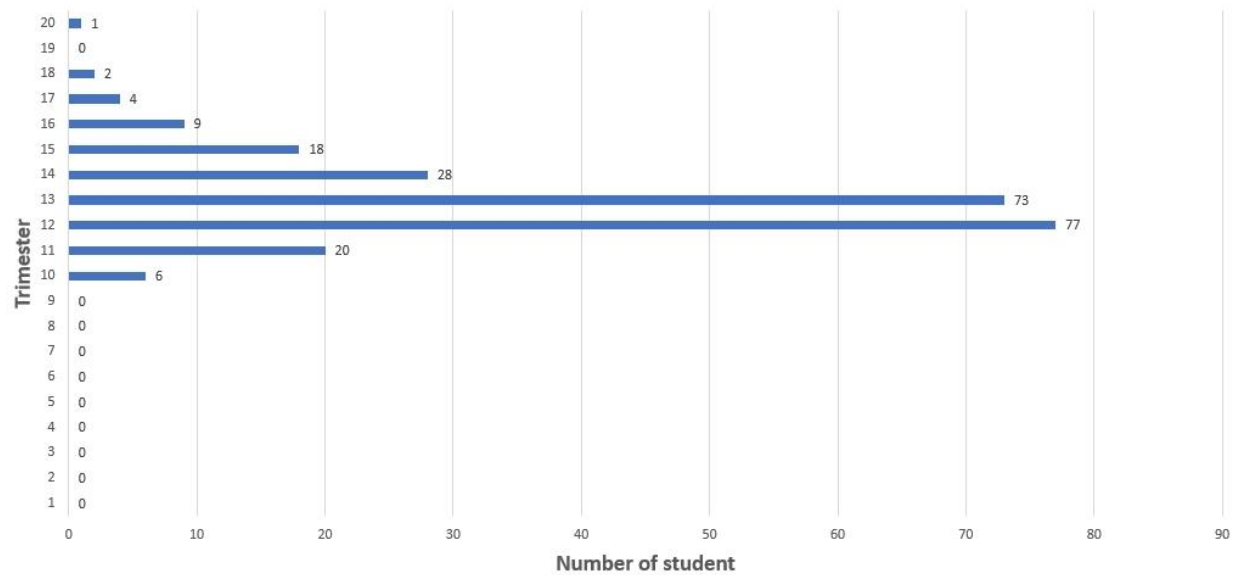


Figure 3.9: Number of students graduated per trimester

Chapter 4: Classifier Model

Classification is a data mining function that assigns items in a collection to target categories or classes. The goal of classification is to accurately predict the target class for each case in the data. For example, a classification model could be used to identify loan applicants as low, medium, or high credit risks.

In the terminology of machine learning, ^[20] classification is considered an instance of supervised learning, i.e. learning where a training set of correctly identified observations is available. The corresponding unsupervised procedure is known as clustering, and involves grouping data into categories based on some measure of inherent similarity or distance.

An algorithm that implements classification, especially in a concrete implementation, is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm, that maps input data to a category. Classification has many applications in customer segmentation, business modeling, marketing, credit analysis, and biomedical and drug response modeling.

Examples of classification algorithms include:

1. Linear classifiers.
2. Decision trees.

There are some other classifier models as well which we are going to discuss later on.

Section 4.1: Linear classifiers

A linear classifier achieves this by making a classification decision based on the value of a linear combination of the characteristics. An object's characteristics are also known as feature values and are typically presented to the machine in a vector called a feature vector.

If the input feature vector to the classifier is a real vector \vec{x} , then the output score is

$$y = f(\vec{w} \cdot \vec{x}) = f\left(\sum_j w_j x_j\right),$$

Where \vec{w} is a real vector of weights and f is a function that converts the dot product of the two vectors into the desired output. (In other words, \vec{w} is a one-form or linear functional mapping \vec{x} onto \mathbf{R} .) The weight vector \vec{w} is learned from a set of labeled training samples. Often f is a simple function that maps all values above a certain threshold to the first class and all other values to the second class. A more complex f might give the probability that an item belongs to a certain class.

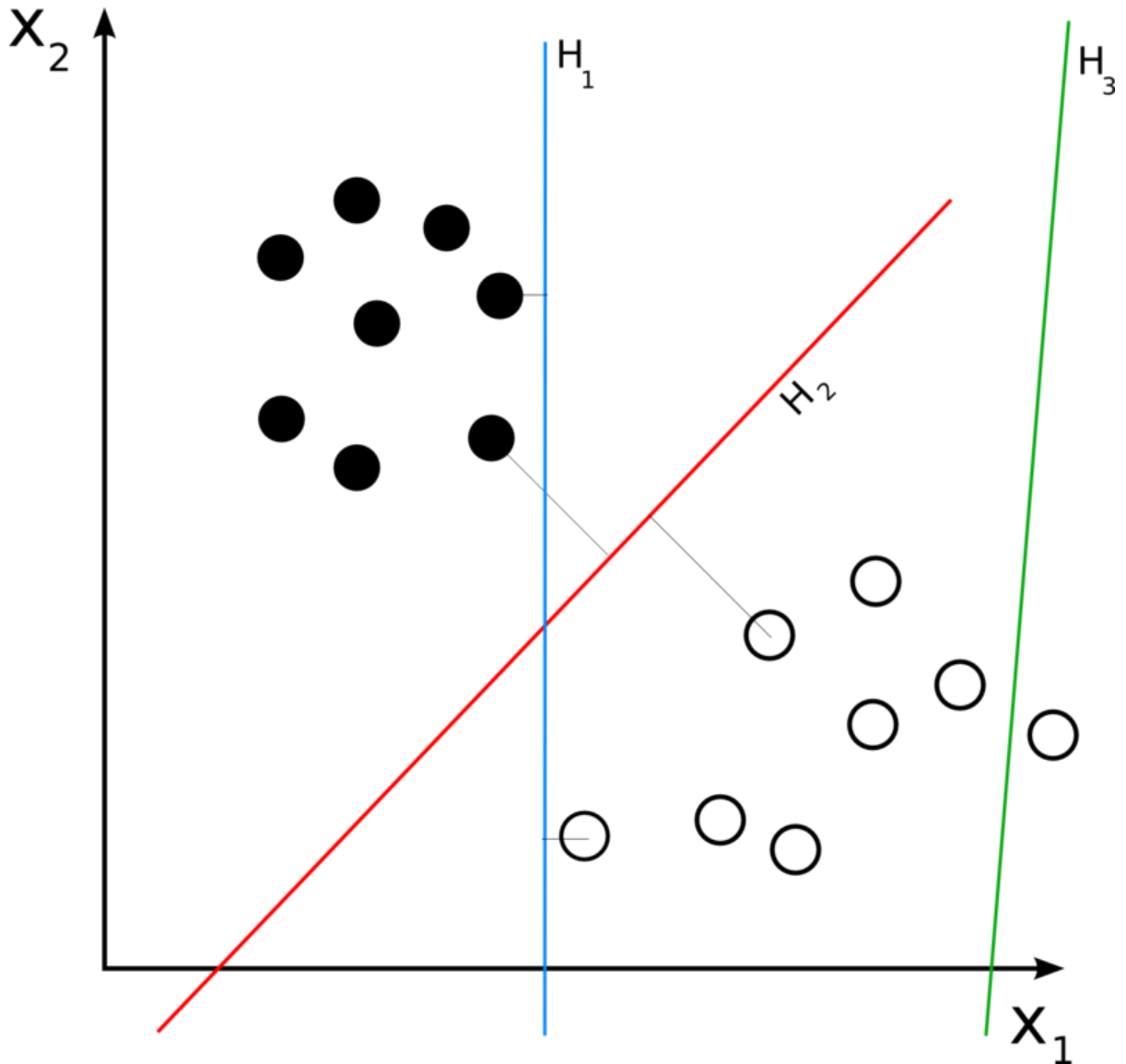


Figure 4.1: A Linear Classification Model.

Section 4.2: Decision tree

A decision tree is a schematic, tree-shaped diagram used to determine a course of action or show a statistical probability. Each branch of the decision tree represents a possible decision,

occurrence or reaction. The tree is structured to show how and why one choice may lead to the next, with the use of the branches indicating each option is mutually exclusive. Decision trees are commonly used in operations research, specifically in decision analysis, to help identify a strategy most likely to reach a goal, but are also a popular tool in machine learning.

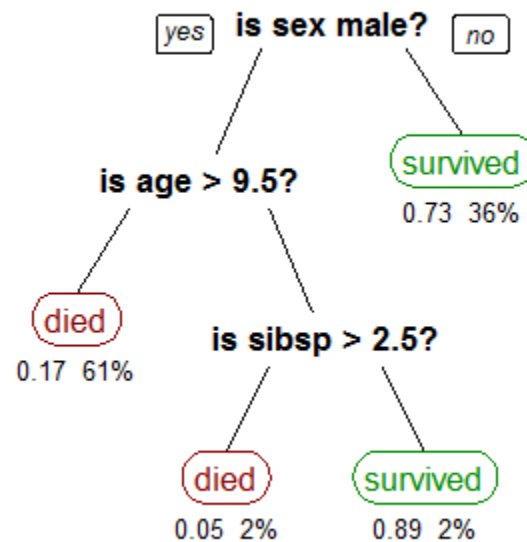


Figure 4.2 : An example of Decision Tree.

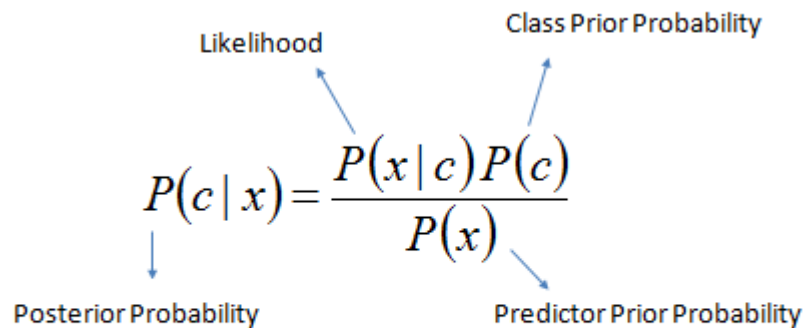
Decision tree learning is a method commonly used in data mining.^[21] The goal is to create a model that predicts the value of a target variable based on several input variables. An example is shown in the diagram at right. Each interior node corresponds to one of the input variables; there are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf.

Section 4.3: Naïve Bayes Classifier

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability $P(c|x)$ from $P(c)$, $P(x)$ and $P(x|c)$. Look at the equation below:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$


$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c)$$

Above,

- $P(c/x)$ is the posterior probability of *class* (c , *target*) given *predictor* (x , *attributes*).
- $P(c)$ is the prior probability of *class*.
- $P(x/c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

Section 4.4: Random Forest

The random forest (Breiman, 2001) is an ensemble approach that can also be thought of as a form of nearest neighbor predictor.

Ensembles are a divide-and-conquer approach used to improve performance. The main principle behind ensemble methods is that a group of “weak learners” can come together to form a “strong learner”. The figure below provides an example. Each classifier, individually, is a “weak learner,” while all the classifiers taken together are a “strong learner”.

The data to be modeled are the blue circles. We assume that they represent some underlying function plus noise. Each individual learner is shown as a gray curve. Each gray curve (a weak learner) is a fair approximation to the underlying data. The red curve (the ensemble “strong learner”) can be seen to be a much better approximation to the underlying data.

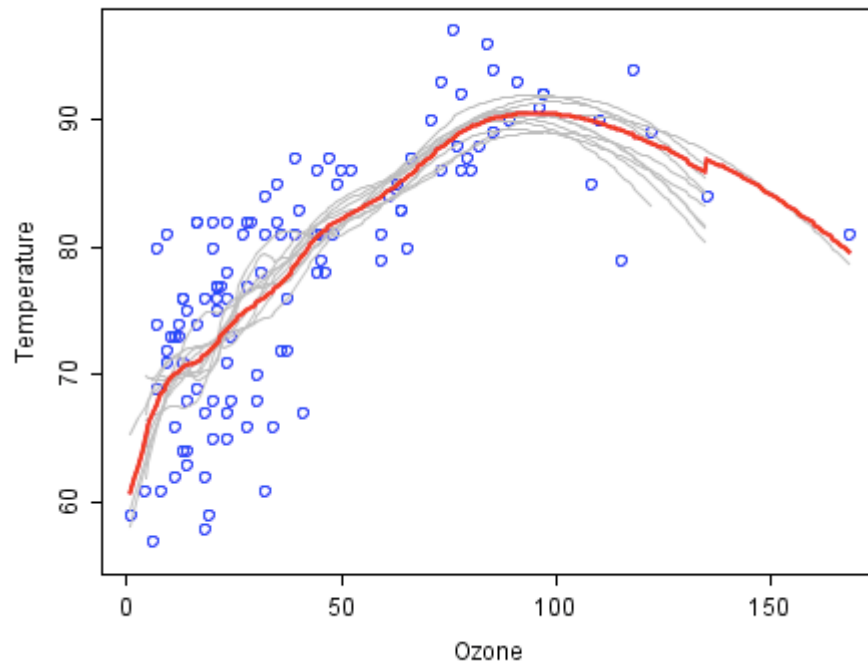


Figure 4.3: A Random Forest Model

The random forest (see figure below) takes this notion to the next level by combining trees with the notion of an ensemble. Thus, in ensemble terms, the trees are weak learners and the random forest is a strong learner.

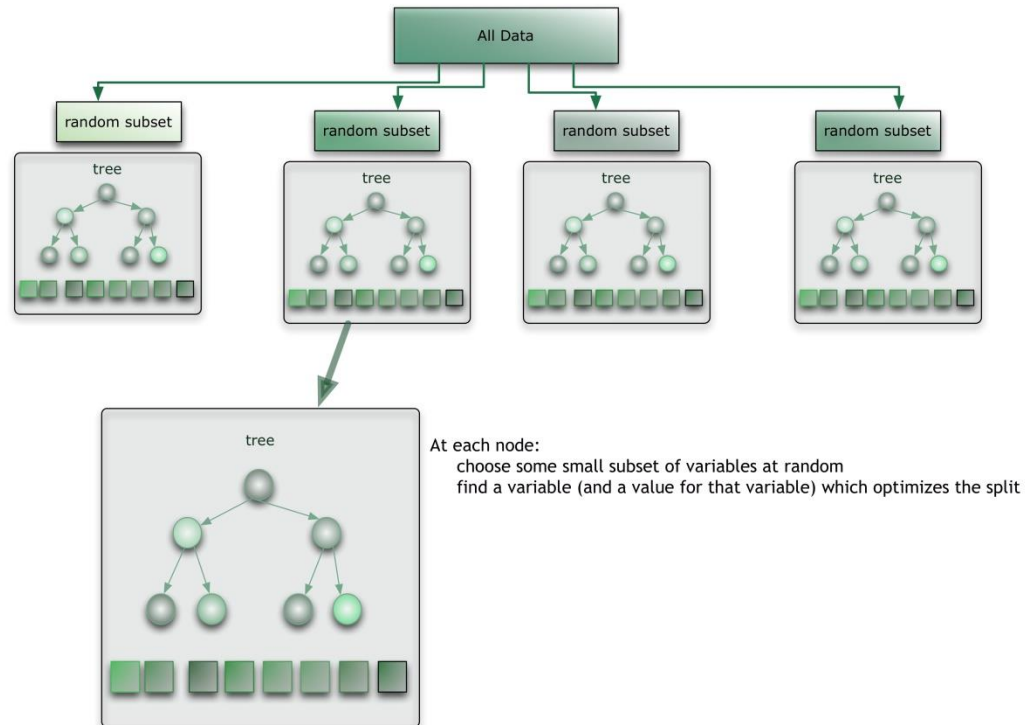


Figure 4.4: Working procedure Random Forest Tree

Here is how such a system is trained; for some number of trees T :

1. Sample N cases at random with replacement to create a subset of the data (see top layer of figure above). The subset should be about 66% of the total set.
2. At each node:
 1. For some number m (see below), m predictor variables are selected at random from all the predictor variables.
 2. The predictor variable that provides the best split, according to some objective function, is used to do a binary split on that node.
 3. At the next node, choose another m variables at random from all predictor variables and do the same.

Depending upon the value of m , there are three slightly different systems:

- Random splitter selection: $m = 1$
- Breiman's bagger: $m = \text{total number of predictor variables}$
- Random forest: $m \ll \text{number of predictor variables}$. Breiman suggests three possible values for m : $\frac{1}{2}\sqrt{m}$, \sqrt{m} , and $2\sqrt{m}$

Section 4.5: J48 Classifier (C4.5)

C4.5 is an algorithm used to generate a decision tree developed by Ross Quinlan^[22] C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier. It became quite popular after ranking #1 in the *Top 10 Algorithms in Data Mining* pre-eminent paper published by Springer LNCS in 2008.^[23]

J48 is an open source Java implementation of the C4.5 algorithm in the WEKA data mining tool. This algorithm also handles the missing values in the training data. After the tree is fully constructed, this algorithm performs the pruning of the tree.

Chapter 5: Experimental Result

To check the performance of classifier models over our collected dataset we have applied three classifier models. First we selected some important attributes to begin the experiment. Then we applied Naïve Bayes classifier, Random Forest classifier and used J48 to create decision tree.

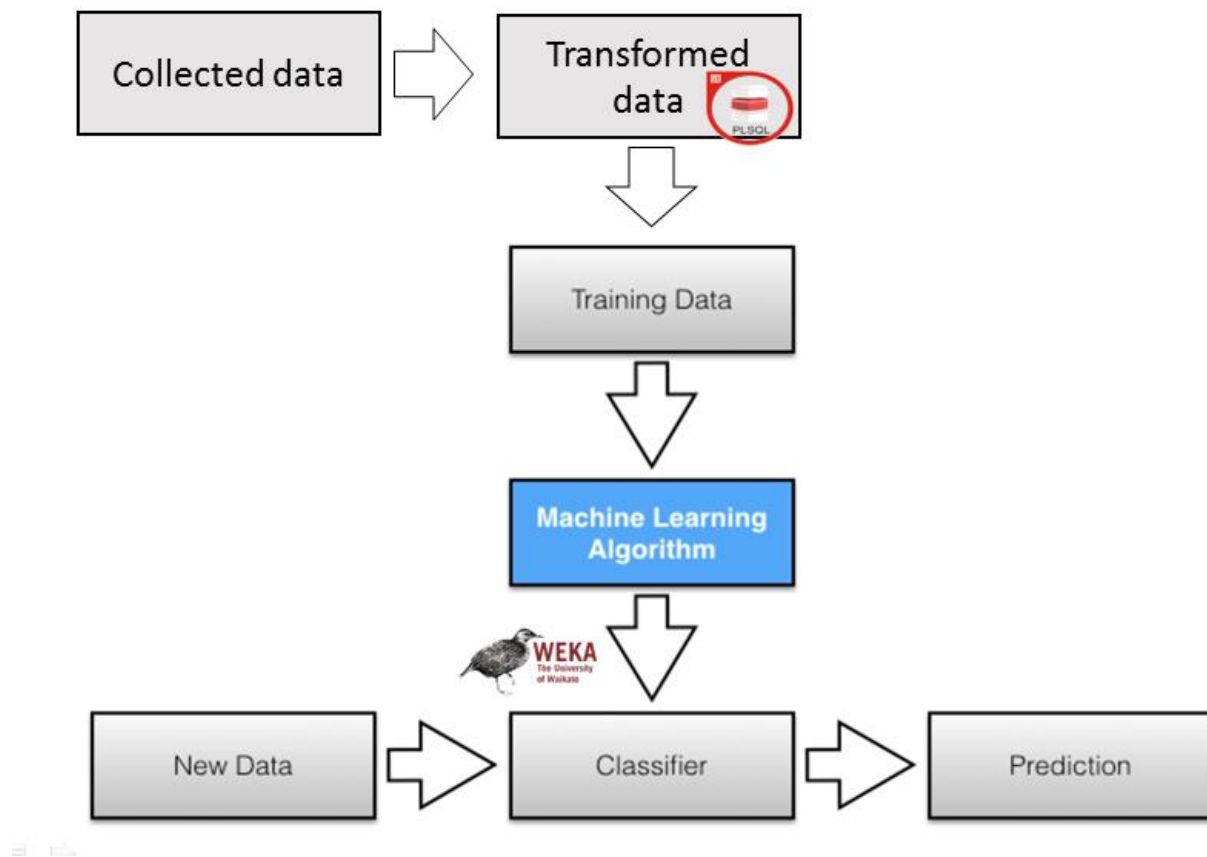


Figure: Experiments work flow

Section 5.1: Attribute Selection

We have taken all the course result of each students from first five trimester and their gender as attributes. And we have chosen final result (CGPA) cluster and graduation appropriate require time (Yes/No) as class attribute for four different types of training set. They were: all courses of maiden five trimester, Computer science courses of maiden five trimester, Computer Engineering courses of maiden five trimester and Mathematical courses. So there were four for result prediction and four for require time prediction total eight tables that we applied the classifier models on.

Type A	All courses of maiden five trimester
Type B	Computer science courses of maiden five trimester
Type C	Computer Engineering courses of maiden five trimester
Type D	Mathematical courses

Table 5.1: Training sets

For: Type A

SPL	SPLLAB	ENG_I	CALCULUS	DISCRETE	DATA	DATALAB	LINEAR	PHYSICS	ENG_II	CIRCUIT	DLD	DLDLAB	ELECTRONICS	ELECTRONICSLAB	ALGORITHM	NUMERIC	FOURIER
-----	--------	-------	----------	----------	------	---------	--------	---------	--------	---------	-----	--------	-------------	----------------	-----------	---------	---------

For: Type B

SPL	DISCRETE	DATA	ALGORITHM	NUMERIC	SPLLAB	DATALAB	ALGOLAB
-----	----------	------	-----------	---------	--------	---------	---------

For: Type C

DLD	CIRCUIT	ELECTRONICS	CA	DLDLAB	ELECTRONICSLAB
-----	---------	-------------	----	--------	----------------

For: Type D

CALCULUS	LINEAR	FOURIER	VECTOR
----------	--------	---------	--------

Section 5.2: Experiments

For the experiment, we have used WEKA data mining tool to implement the classifier models. The result that we get after implementing each classifier models is then used to predict. Basically there are five types of results that have been found in WEKA after applying each classifier model. And those 5 types of results are:

1. Correctly Classified Instances: It means that the built classifier classified the instance correctly.
2. Mean Absolute Error: Basically, it calculates that how much the predicting value is close to the real or absolute value. The greater value is preferable when comparing among different algorithms.
3. Root Mean Squared Error: The difference between the actual value and predicting value. The smaller value is preferable when comparing among different algorithms.
4. Relative Absolute Error: It is the percentage of the Root Mean Squared Error. The smaller value is preferable when comparing among different algorithms.
5. Root Relative Squared Error: This simple predictor is just the average of the actual values. The basic equation is given below:

$$\text{Root Relative Squared Error} = (\text{Root Mean Squared Error (RMSE)} / \text{Measured Value}) \times 100$$

The results that we have found are described with the tables below. The regression model with better performance marked in the table.

	J48	Random Forest	Naïve Bayes
Type A	68.4874 %	79.8319 %	73.1092 %
Type B	62.1849 %	63.8655 %	65.1261 %
Type C	68.0672 %	74.7899 %	68.4874 %
Type D	65.7261 %	63.8655 %	65.5462 %

Table 5.2: Correctly Classified Instance percentage for result prediction

As we can see here **Random Forest** outperformed other classifier models.

	J48	Random Forest	Naïve Bayes
Type A	68.4874 %	72.2689 %	74.3697 %
Type B	63.0252 %	65.9664 %	70.1681 %
Type C	64.2857 %	61.3445 %	71.0084 %
Type D	58.4034 %	61.3445 %	68.4874 %

Table 5.3: Correctly Classified Instance percentage for require time prediction

As we can see here **Naïve Bayes** outperformed other classifier models.

Section 5.3: Comparison of Results

After calculate the result of different category of courses we can say that Random Forest classifier model works better than other two models for predict final result and Naïve Bayes works better than other two for predict require time. Type A accuracy rate is better than other types.

Chapter 6: String matching

Section 6.1: Approximate string matching algorithm

In computer science, approximate string matching (often colloquially referred to as fuzzy string searching) is the technique of finding strings that match a pattern approximately (rather than exactly). The problem of approximate string matching is typically divided into two sub-problems: finding approximate substring matches inside a given string and finding dictionary strings that match the pattern approximately. The closeness of a match is measured in terms of the number of primitive operations necessary to convert the string into an exact match. This number is called the edit distance between the string and the pattern.

One possible definition of the approximate string matching problem is the following: Given a pattern string and a text string, finds a substring in T , which, of all substrings of T , has the smallest edit distance to the pattern P .

A brute-force approach would be to compute the edit distance to P for all substrings of T , and then choose the substring with the minimum distance. However, this algorithm would have the running time $O(n^3 m)$.

The most common application of approximate matchers until recently has been spell checking. With the availability of large amounts of DNA data, matching of nucleotide sequences has become an important application^[24]. Approximate matching is also used in spam filtering. String matching cannot be used for most binary data, such as images and music. They require different algorithms, such as acoustic fingerprinting.

Section 6.2: FuzzyWuzzy

Fuzzywuzzy a python library for fuzzy string matching, its have different types of string matching algorithm. Such as Wratio, Qratio ,UWratio, Partial_ratio, partial_token_set_ratio and so on.

WRatio

WRatio (s1, s2, force_ascii=True) Return a measure of the sequences' similarity between 0 and 100, using different algorithms. partial_ratio (s1, s2) "Return the ratio of the most similar substring as a number between 0 and 100.

partial token set ratio

partial_token_set_ratio (s1, s2, force_ascii=True, full_process=True) partial_token_sort_ratio (s1, s2, force_ascii=True, full_process=True) Return the ratio of the most similar substring as a number between 0 and 100 but sorting the token before comparing.

UWRatio

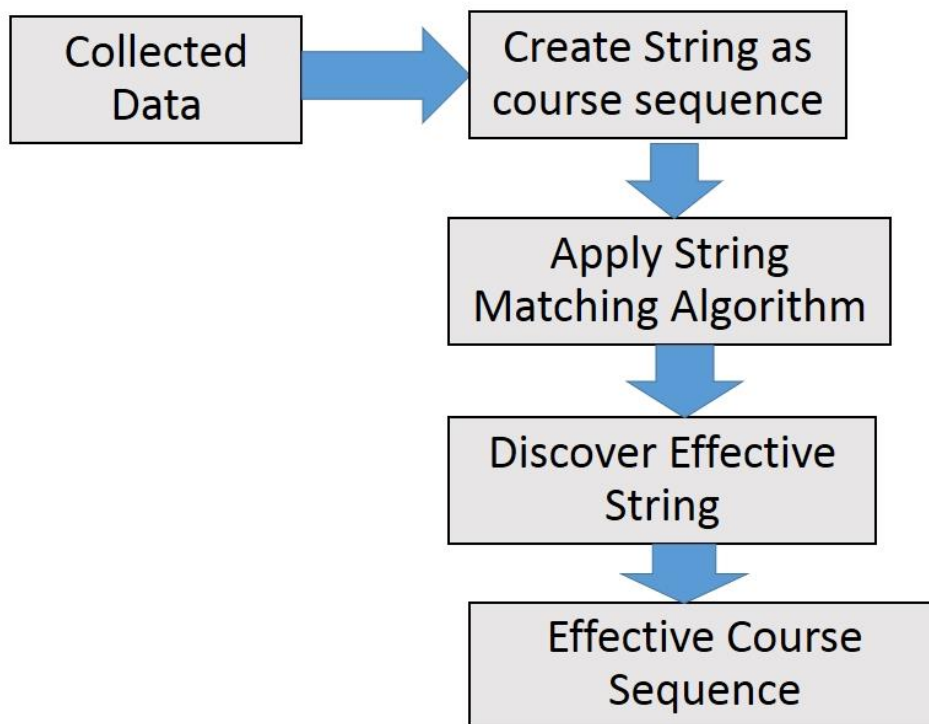
UWRatio (s1, s2) Return a measure of the sequences' similarity between 0 and 100, using different algorithms. Same as WRatio but preserving unicode.

Token sort ratio

token_sort_ratio(s1, s2, force_ascii=True, full_process=True) Return a measure of the sequences' similarity between 0 and 100 but sorting the token before comparing.

We use **Wratio** in our research paper because it return the ratio of the most similar substring.

Section 6.3: Work flow



30

Section 6.4: Course Data

COURSEID	CHARACTER
MATH 125	"
CSI 342	N1
CSE 113	%
PHY 127	C
CSI 412.	I1
CSE 430	P1
CSE 425	L
PHY 104.	C1
CSI 122	A1
CSE 323	K
CSE 415	P
STAT 205	*
PHY 103	C
CSI 229	,
CSI 415	Q
CSE 236	.
PHY 105	C
PSY 101	Z
CSI 422.	R1
PHY 106	C1
CSI 416.	Q1
CSE 463	\
CSI 233)
CSI 227	D
CSI 312	M1
CSI 231	H
CSE 315	:
MATH 201	+
CSE 224	E1
ACT 110	<
CSE 225	E
CSI 309	H
CSI 218.	B1
CSE 420	L1
CSE 429	P

CSI 424.	S1
CSE 426	L1
CSE 234.	.
CSI 311	M
CSI 212	F1
CSI 222	J1
CSI 232	H1
MATH 415	,
CSE 313	(
IPE 401	;
CSI 113	%
MATH 151	"
CSE 465]
CSE 471	{
CSI 212.	F1
CSI 121	A
PHY 117	C
CSE 123	G
CSE 223	E
CSI 423	S
CSE 453	>
MATH 157	&
CSE 457	?
CSI 424	S1
ENG 105	!
CSE 224.	E1
CSI 228.	D1
CSE 124.	G1
CSE 324.	K1
CSI 322.	O1
CSI 122.	A1
CSE 469	—
CSE 226.	E1
CSE 473	
CSI 218	B1
CSI 211	F
CSI 221	J
CSI 321	O
CSI 422	R1
CSI 412	I1

CSI 322	O1
ENG 101	!
CSE 413	L
ACT 111	<
CSI 219	#
CSE 467	^
MATH 183	Y
CSI 312.	M1
CSE 419	L
CSI 342.	N1
CSE 418	P
CSI 310	H1
CSE 477	~
CSI 228	D1
ENG 103	\$
CSI 341	N
CSI 421	R
CSI 411	I
CSE 324	K1
CSE 400	[
CSE 461	[
MATH 203	(
MATH 205	*
CSI 222.	J1
CSI 232.	F1
MATH 187	&
CSE 481	@
CSI 217	B
MATH 135	#
ECO 213	/
CSE 414	L1
MATH 215	Y
PHY 104	C1
CSI 317)
CSE 124	G1
PHY 101	C
CSI 416	Q1

After this process, insert the table in oracle database and assign character corresponding subject for each student.

STUDENTID	COURSEID	CHARACTER
011033039	CSI 217	B
011033039	MATH 135	#
011033039	CSI 227	D
011033039	CSI 228	D1
011033039	CSI 311	M
011033039	CSI 121	A
011033039	CSI 312	M1
011033039	CSI 218	B1
011033039	CSI 211	F
011033039	ENG 103	\$
011033039	CSI 212	F1
011033039	PHY 117	C
011033039	ECO 213	/
011033039	CSI 221	J
011033039	CSE 234	.
011033039	CSI 222	J1
011033039	CSI 321	O
011033039	MATH 125	"
011033039	CSI 232	H1
011033039	CSI 231	H
011033039	CSE 414	L1
011033039	CSI 342	N1
011033039	MATH 215	Y
011033039	CSI 422	R1
011033039	CSI 341	N
011033039	CSI 421	R
011033039	MATH 415	,
011033039	CSI 411	I
011033039	CSE 324	K1
011033039	CSI 122	A1
011033039	CSI 412	II
011033039	CSE 123	G
011033039	CSE 323	K
011033039	PHY 104	C1
011033039	CSE 223	E
011033039	CSI 423	S
011033039	CSE 315	:
011033039	CSE 313	(
011033039	CSE 415	P

011033039	CSI 322	O1
011033039	STAT 205	*
011033039	CSE 400	[
011033039	CSE 453	>
011033039	MATH 201	+
011033039	CSE 113	%
011033039	IPE 401	;
011033039	ENG 101	!

For this process, we use PL/SQL programming and then we construct string for every student.
Code for sequence string of courses:

create or replace procedure create string is

reqtime number;

CURSOR student_list IS

select distinct studentid from studentcourse1 order by studentid;

stdlist_val student_list%ROWTYPE;

CURSOR student_smster(stdid in varchar2) IS

select studentid, trimester from studentcourse1 group by studentid, trimester having
studentid=stdid order by trimester;

stdsmster_val student_smster%ROWTYPE;

CURSOR student_word(stdandsmster in varchar2, semester in varchar2) IS

select studentid, trimester, word from studentcourse1 where studentid=stdandsmster and
trimester=semester order by word; stdword_val student_word%ROWTYPE;

begin

EXECUTE IMMEDIATE 'update studentcourse1 set STUDENTCOURSE1.word=(select
COURSES.word FROM courses where
STUDENTCOURSE1.COURSEID=COURSES.COURSEID)';

FOR stdlist_val IN student_list LOOP

str:= NULL;

FOR stdsmster_val IN student_smster(stdlist_val.studentid) LOOP

FOR stdword_val IN student_word(stdlist_val.studentid, stdsmster_val.trimester) LOOP

str:= str || stdword_val.word;

EXIT WHEN student_word%NOTFOUND;

END LOOP;

```

EXIT WHEN student_smster%NOTFOUND;
END LOOP;
insert into studentstring values(stdlist_val.studentid,str,null,null);
EXIT WHEN student_list%NOTFOUND;
END LOOP;
commit;
end;

```

Section 6.5: Create String as course sequence

STUDEN TID	STRINGS
011091028	!"\$AA1Y#&BB1%DD1EE1<CC1FF1(),:+/III;GG1HH1JJ1KK1MM1NN1LL1OO1[^*PP1RR1{QQ1[]
011091031	!"\$Y#&BB1*DD1EE1%CC1FF1(),:+/III;GG1HH1JJ1A1KK1MM1NN1ALL1OO1[^<PP1RR1{)QQ1[]
011091032	!"\$A1Y#&BB1%EE1<CC1D1FF1(),:+/III;GG1HH1JJ1DKK1MM1NN1LL1OO1[^*PP1RR1{AQQ1[]
011091034	!"\$AA1Y#&BB1%DD1EE1<CC1FF1(),:+/III;GG1HH1JJ1KK1MM1NN1LL1OO1[^*PP1RR1{QQ1[]
011091035	!"\$AA1Y#&BB1%DD1E1<CC1FF1),:E(+.II1;GG1HH1JJ1KK1MM1NN1/LL1OO1[^*PP1RR1{QQ1[]
011091043	!"\$AA1Y#&BB1%DD1EE1<CC1FF1(),:+/III;GG1HH1JJ1;KK1MM1N1LL1OO1[^*PP1RR1{NQQ1[]
011092002	!"\$AA1Y%BB1#DD1EE1(.FF1GG1&)HH1,:.II1;JJ1L+MM1^<KK1L1PP1[{OO1SS1]CC1NN1[/RR1*
011092003	!#E%A1E1A&BB1"DD1C1FF1).JJ1*:HH1III1LL1{<MM1PP1]^\$G1KK1OO1Y+,/;SS1[(GNN1RR1[C
011092005	!"\$AA1Y#%BB1DD1GG1*<EE1(FF1):.HH1III1JJ1K1&CC1KLL1NN1,/;QQ1^MM1PP1S{OO1S1][+
011092006	!"\$AA1Y#%BB1C+C1DD1E1(.FF1GG1&)HH1*,:EII1;JJ1KK1LL1<MM1PP1SS1NN1RR1]/OO1[{{[
011092007	!"\$AA1Y#%BB1+DD1E1.FF1GG1&())HH1*,:EII1;JJ1LL1CC1KK1MM1^<PP1RR1]NOO1[{/N1SS1[
011092010	!"A1#BB1Y\$%A<DD1EE1(FF1GG1):.HH1III1JJ1K1&MM1^KLL1NN1,CC1OO1{*/;PP1[QQ1SS1[]+
011092015	!"\$AA1Y#%BB1+DD1EE1.FF1GG1&())HH1*,:II1;JJ1LL1CC1KK1MM1^PP1RR1]NOO1[{/<N1QQ1[
011092016	"\$!AA1Y#%BB1EE1DD1FF1JJ1Z&(+.<HH1):CIK*^,GG1MM1N]K1OO1[{LL1N1SS1[PI1P1C1RR1
011092023	!"#\$AA1Y%BB1EE1(+<DD1JJ1Z*;FF1GG1):.HH1,II1KK1]&MM1SS1[^LL1N

	N1RR1CC1OO1PP1[{
011092025	!"\$AA1Y#%BB1CDD1E1(.FF1GG1&)HH1*.;EII1;JJ1LL1+MM1^<KK1PP1C1N [{}OO1SS1]/N1QQ1[
011093002	!"\$AA1Y#%BB1DD1EE1(*FF1&,.CH)+<JJ1:II1L/GG1MM1;H1KK1NOPSS1[{ C1L1N1[^O1P1RR1[
011093003	!"AA1#%BB1DD1EE1Y.FF1G1()J1,N1L1&C1GM1/LPP1]H1II1KOO1{\$*CHK 1[:SS1^+MN[;<JRR1
011093004	!"\$AA1Y#%BB1&DD1EE1+,<FF1)*.CC1(II1J:;J1NN1HH1KK1LL1GMM1^{O O1SS1[]/G1PP1QQ1[
011093006	!"AA1#%BB1DD1EE1Y(*.FF1GG1)JJ1:;NN1II1LL1&CC1HMM1+/PP1];<\$KO O1{H1K1QQ1[SS1[^
011093007	!"AA1Y#%BB1DD1EE1(*.FF1,<JJ1:;NN1GII1L1&/L]C1H1MM1OPP1{\$HKK 1O1[^+CG1RR1SS1[
011093010	!"AA1Y#BB1%DD1&,FF1)<J1:;CII1/GG1MM1;H1K1NO*C1EE1LL1N1^{+O1 PP1RR1\$(SS1[[J]HK
011093011	"!AA1#\$BB1Y%CC1DEE1(*,D1GG1)/.HH1&+<FF1I;JJ1LL1N:MM1PP1[^KK1 N1OO1SS1{II1QQ1[]
011093014	!"\$AA1Y#BB1DD1EE1,<FF1%)*.CC1(II1J:;J1NN1HH1KK1LL1GMM1^{OO1S S1[]/G1PP1QQ1[&+
011093018	!"\$AA1Y#%BB1&CC1DD1,<FF1)*+.EE1(II1JJ1:;NHH1KK1LL1GMM1^{OO1 SS1[]/G1N1PP1QQ1[
011093019	!"#%AA1<BB1DD1EE1Y),FF1(*.;CGG1II1JJ1/C1HH1MM1LL1NN1{&PP1]^:R R1S1[\$KK1OO1[+S
011093022	!"\$%Y#AA1CC1*BB1EE1(DD1)FF1HH1<II1JJ1,GMM1^NNOO1SS1[]{&./;N1Q +KK1L1PP1Q1[:LG1
011093024	"!Y#%A1AEE1(BFF1&B1CDD1)<JJ1:;II1/GG1MM1;C1H1KK1LN[L1N1P^{+\$+ O1P1RR1[]*HOS1S
011093028	!#%AA1BB1EE1(*DD1.FF1H),<JJ1:II1L/GG1MM1;H1KK1NOPS1["C1L1N1^{ +O1P1RR1[]\$&CSY
011093029	!"\$AA1Y#%BB1DD1EE1(*FF1&,.CH)<JJ1:II1L/GG1MM1;C1H1KK1NOSS1[L 1N1P[^+?O1P1QQ1[
011093030	!"\$AA1Y#%BB1&CC1DD1,<FF1)*.EE1(II1JJ1:;NN1HH1KK1LL1GMM1^{G1 OO1SS1[]+/PP1QQ1[
011093031	!"\$AA1Y#BB1C1DD1,<FF1%).*EE1(II1JJ1:;N1HH1KK1LL1GMM1^{*G1NOO1 []CPP1Q&+/Q1SS1[
011093032	!"\$AA1Y#BB1&C1DD1,<FF1%).*EE1(IJ1:;NN1HH1K1LGMM1{*G1OO1[]CP Q+II1KL1P1Q1^/SS1[
011101001	!"\$Y#%BB1&DD1EE1+CC1F1,.:F)*<NN1GG1II1KK1AA1HJJ1LL1/MM1SS1{ OO1QQ1[^(;H1PP1[
011101002	!"\$%BB1Y#&DED1E1+FF1)*.</G1II1CHJJ1LL1C1NN1SS1{:MM1PP1]KK1O O1RR1^(;AA1GH1[
011101003	!"\$%AA1#BB1YDD1EE1()+,FF1&*.:;<CGG1II1JJ1/C1HH1MM1LL1NN1O{PP 1RR1]^KK1O1SS1[[

011101004	!"\$AA1Y#%BB1D1E1&DE)CC1FF1,..NN1/GG1KK1(JJ1LL1*IMM1{I1OO1QQ1^+PP1[];HH1SS1[
011101005	!"\$Y#%BB1&DD1EE1CC1FF1(*HH1).AA1:<JJ1,LL1G1II1]KK1MM1^GNN1{PP1QQ1SS1OO1[+;/;
011101011	!"\$AA1Y#&BB1E(+DD1,:E1FF1)*.KK1G1II1JJ1CHH1LL1MM1C1NN1OSS1{GPP1QQ1]O1[^%/<
011101012	!"\$YAA1C#%BB1&DD1EE1(,;FF1)*.<GG1II1J1HH1LM1L1NN1SS1{OO1PP1[^+C1KK1[]/JQQ1:M
011101014	!"\$AA1Y#%BB1DD1EE1(+FF1,:HH1)KK1N1<II1JJ1GG1^{*/MM1].OO1SS1C1LL1[NPP1;CQQ1&
011101016	!"AA1#%BB1DD1EE1+C1FF1(*HH1)/:NN1,<CGG1;IJ1Y&.KK1LL1MM1[]{I1OO1PP1^\$QQ1SS1[
011101017	!"\$AY#%&+BB1CC1A1DD1EE1(),FF1*..;<GG1II1JJ1/HH1MM1LL1NN1O{PP1RR1]^KK1O1SS1[[
011101018	!"A1YA#BB1EE1(CDD1)+<FF1&,II1.HH1NN1JJ1LL1{\$%MM1C1KK1OO1RR1*/;]^:GG1PP1SS1[
011101024	!"#BB1AA1DD1%FF1+GG1Y*,<EE1&().;:CC1II1HJK1L1/J1KLN{MM1N1]\$O1PP1[@H1RR1SS1[
011101025	!"\$A1#%BB1DD1EE1Y(+.FF1),:GG1II1KK1ACC1JJ1LL1*MM1NN1{/<HH1]O1PP1SS1[&;QQ1[^
011101026	!"\$AA1#%BB1C1&DD1EE1+;FF1(),..*<KK1/GG1II1NCJJ1LL1N1{MM1RR1]OO1PP1SS1YHH1[^
011101028	!"<AA1#BB1Y%C1DD1EE1()*,.GG1HII1&FF1JJ1LL1MM1NSS1N1OO1P1[H1KK1QQ1[]^ [+;/;J\$CP
011101039	!"\$AA1Y#%BB1CC1&DD1EE1F,:H*+.<JJ1()F1GG1N1/H1I^;LL1MM1{I1OPPISS1[KK1O1[NQQ1]

Section 6.6: Apply String Matching Algorithm

Then we cluster this string according to student CGPA and apply Fuzzy String Matching algorithm for each cluster and each student to another. Average the CGPA which group of student's algorithm results is 70 up. Select this string which average CGPA is high. This is the best string.

	STUDEN...	UP70	AVGC...	STRING
1	011112006	14	3.65428...	!"\$AA1Y%\$BB1#CC1DD1EE1+<FF1GG1)*,.:HH1II1/JJ1KK1LL1MM1NN1^OO1QQ1];PP1SS1{([
2	011122017	14	3.65	!AA1"#\$BB1CC1Y%DD1EE1&<FF1)+.:(*,G1/HH1II1:JJ1LKK1L1MM1]NN1OO1PP1SS1{~GRR1[
3	011122035	13	3.64153...	!AA1"#\$BB1CC1Y%DD1EE1+FF1GG1&,<(*.HH1:;II1N/JJ1KK1LL1MM1NN1S]OO1PP1RR1~S1
4	011112001	13	3.63769...	!"\$AA1Y%\$BB1CC1DD1EE1%)+FF1(*,./:HH1II1JJ1KK1LL1MM1NN1^OO1]{G1PP1SS1<QQ1[
5	011103025	12	3.6125	!"\$AA1Y%\$BB1*CC1DD1&EE1FF1()+.GG1:<HH1II1JJ1LL1NN1{;KK1MM1]OO1PP1RR1[,/QQ1[
6	011112010	12	3.61083...	!"A1%\$ABB1CC1Y%DD1EE1)FF1GG1(*.HH1I,:<I1+JJ1KK1/LL1NN1MM1PP1^{}OO1RR1SS1[&;]
7	011113034	12	3.60333...	!"AA1Y%\$BB1CC1DD1EE1+,<FF1()**.GG1/:HH1II1JJ1LL1NN1;KK1MM1^OO1PP1QQ1{&@RR1[
8	011102014	11	3.58090...	!"\$AA1Y%\$BB1:CC1DD1EE1(+;FF1GG1)*,./II1JJ1KK1HH1LL1NN1{MM1PP1SS1[]OO1QQ1[^
9	011122011	10	3.536	!AA1"#\$BB1CC1Y%DD1EE1+<FF1GG1(),./&*HH1JJ1:;II1NKK1LL1MM1NN1PP1QQ1^OO1]~RR1[
10	011113007	10	3.536	!"\$AA1Y%\$BB1CC1DD1EE1&+FF1G()**,./:HH1II1JJ1LL1NN1G1KK1MM1RR1;<SS1]@OO1PP1[{
11	011111003	10	3.534	"\$AA1Y%\$BB1CC1DD1EE1),FF1%\$HH1/:JJ1(.II1^{;KK1MM1[]NOO1+GG1SS1!LL1RR1<N1PP1
12	011091002	10	3.534	!"\$AA1Y%\$BB1%\$DD1EE1<CC1FF1(),:JJ1+./II1;GG1HH1{KK1MM1NN1LL1OO1[^PP1RR1]QQ1[
13	011091010	8	3.44	!"\$AA1Y%\$BB1%\$DD1EE1<CC1FF1(),:+. /II1;GG1HH1JJ1KK1MM1NN1LL1OO1[^*PP1RR1]QQ1[]
14	011103021	6	3.29333...	!"\$A1Y%\$BB1CC1&DD1EE1()*+,FF1GG1:.HH1JJ1/:KK1MM1II1OO1^{<LL1N1[]PP1QQ1RR1[AN
15	011082007	4	2.9725	!"AA1%\$EE1&BB1CC1%\$DD1*FF1GG1Y,<F1H(:JJ1KK1MM1NN1LL1OO1SS1^)^PP1[{:II1QQ1[{/
16	011073008	1	0	A1!C1BB1EE1Y%\$DD1%\$CFF1()+.<*:AF1HJJ1II1MM1NN1["/KK1OO1[L L1QQ1\GG1SS1]^,;PP1
17	011072001	1	0	!"AA1#<BB1CDD1EE1\$.JJ1Y%+FF1&,C1GG1(*L M/:F1HN)KK1MM1N1^{II1OO1PP1[L1PQQ1Z\
18	011092023	1	0	!"\$AA1Y%\$BB1EE1(+<DD1JJ1Z*;FF1GG1)..HH1,II1KK1]&MM1SS1[^LL1NN1RR1CC1OO1PP1{

Section 6.7: Discover Effective String

Code for decode string

```
import cx_Oracle
from fuzzywuzzy import fuzz
db = cx_Oracle.connect('student/oracle')
cur=db.cursor()

cur=cur.execute('select * from datastring')
string='!AA1"#$BB1CC1Y%DD1EE1&<FF1)+.:(*,G1/HH1II1:JJ1LKK1L1MM1]NN1OO1PP
1SS1{~GRR1['
courseId=[]
character=[]
course=[]
one='1'
for data in cur:
    courseId.append(data[0])
    character.append(data[1])
lenth=len(string)

for f in range(lenth-1):
    if string[f+1]==one:
        s=string[f] + string[f+1]
        for j in range(len(courseId)):
            if character[j]==s:
                course.append(courseId[j])
                break
```



```

else:
    for j in range(len(courseId)):
        if character[j]==string[f]:
            course.append(courseId[j])
            break

print(course)
db.commit()
db.close()

```

Section 6.8: Effective Course Sequence

Finally we found an Effective course sequence

Effective Sequence

['ENG 101', 'CSI 121', 'CSI 122', 'MATH 151', 'CSI 219', 'ENG 103', 'CSI 217', 'CSI 218', 'PHY 127', 'PHY 104.', 'MATH 183', 'CSE 113', 'CSI 227', 'CSI 228', 'CSE 225', 'CSE 226.', 'MATH 187', 'ACT 111', 'CSI 211', 'CSI 212', 'CSI 233', 'MATH 201', 'CSE 236', 'IPE 401', 'CSE 313', 'STAT 205', 'CSI 229', 'CSE 124', 'ECO 213', 'CSI 231', 'CSI 310', 'CSI 411', 'CSI 412', 'CSE 315', 'CSI 221', 'CSI 222', 'CSE 425', 'CSE 323', 'CSE 324.', 'CSE 426', 'CSI 311', 'CSI 312', 'CSE 465', 'CSI 341', 'CSI 342', 'CSI 321', 'CSI 322.', 'CSE 415', 'CSE 430', 'CSI 423', 'CSI 424.', 'CSE 471', 'CSE 477', 'CSE 123', 'CSI 421', 'CSI 422.']

Chapter 7: Conclusions and Future work

Section 7.1: Conclusions

In this thesis, collected data was in haphazard manner because of different software used in different times to store data. It's a big challenge for us to make this data usable for PL/SQL. Classifier method is used on student database to predict the students division on the basis of student database. This study will help to the students and the teachers to improve the division of the student. Then the sequence method is used and we find out an effective course sequence for the student. This study will work to identify those students which needed special attention to reduce failing ration and taking appropriate action at right time. Present study shows that academic performances of the students are not always depending on their own effort. Our investigation shows that other factors have got significant influence over students' performance. This proposal will improve the insights over existing methods.

Section 7.2 Future Work

In this research there is not enough data for prediction. If we get the admission data then we can predict accurate in our future prediction. If we also get SSC and HSC data then we can predict more accurate in our future prediction. We will deep analysis sequence of subject and we will find out a better course plan.

References

- [1] "Data Mining Curriculum". ACM SIGKDD. 2006-04-30. Retrieved 2014-01-27.
- [2] Heikki, Mannila, Data mining: machine learning, statistics, and databases, IEEE, 1996.
- [3] U. Fayyad, Piatetsky, G. Shapiro, and P. Smyth, From data mining to knowledge discovery in databases, AAAI Press / The MIT Press, Massachusetts Institute Of Technology. ISBN 0-262-56097-6, 1996.
- [4] J. Han and M. Kamber, "Data Mining: Concepts and Techniques," Morgan Kaufmann, 2000.
- [5] Machine Learning and Optimization
- [6] [Http://www.britannica.com/EBchecked/topic/1116194/machine-learning](http://www.britannica.com/EBchecked/topic/1116194/machine-learning).
- [7] Ron Kohavi; Foster Provost (1998). "Glossary of terms". Machine Learning. 30: 271–274.
- [8] Machine learning and pattern recognition "can be viewed as two facets of the same field."
- [9] "Data Mining Curriculum". ACM SIGKDD. 2006-04-30. Retrieved 2014-01-27.
- [10] Clifton, Christopher (2010). "Encyclopædia Britannica: Definition of Data Mining". Retrieved 2010-12-09.
- [11] Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996). "From Data Mining to Knowledge Discovery in Databases" (PDF). Retrieved 17 December 2008.
- [12] RAJIBUSSALIM, MInfoTech "DATA MINING FOR STUDYING THE IMPACT OF REFLECTION ON LEARNING" 21 May 2014.
- [13] Dr. Syed Akhter Hossain "Analysis of Student Performance using Data Mining" December 2014.
- [14] Abeer Badr El Din Ahmed¹, Ibrahim Sayed Elaraby², "World Journal of Computer Application and Technology 2(2): "43-47, 2014.
- [15] Brijesh Kumar Baradwaj, Saurabh Pal, Data mining: machine learning, statistics, and databases, 1996.
- [16] Nikhil Rajadhyax, Rudresh Shirwaikar, Data Mining on Educational Domain, 2012.
- [17] (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 5, 2016.
- [18] (IJCSIS) International Journal of Computer Science and Information Security, Vol. 9, No. 4, April 2011.
- [19] Baradwaj, B.K. and Pal, S., 2011. Mining Educational Data to Analyze Students' Performance. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No. 6, 2011.
- [20] Alpaydin, Ethem (2010). Introduction to Machine Learning. MIT Press. p. 9. ISBN 978-0-262-01243-0.
- [21] Rokach, Lior; Maimon, O. (2008). Data mining with decision trees: theory and applications. World Scientific Pub Co Inc. ISBN 978-9812771711.
- [22] Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.
- [23] Umd.edu - Top 10 Algorithms in Data Mining.
- [24] [Zobel, Justin](#); [Dart, Philip](#) (1995). "Finding approximate matches in large lexicons". *Software-Practice & Experience*. **25** (3): 331–345..