PRIFYSGOL
BANGOR
UNIVERSITY

School of Computer Science and Electronic Engineering
College of Environmental Sciences and Engineering

Examining the Link Between Course Structure and Student Performance using Machine Learning

Darsh Jadhav

Submitted in partial satisfaction of the requirements for the
Degree of Bachelor of Science
in Computer Science

Supervisor Dr. Cameron C. Gray

May 2020

Acknowledgements

> *It is during our darkest moments that we must focus to see the light.*
>
> **— Aristotle**

I would like to thank Dr. Cameron Gray providing his continuous support and guidance to help me complete this work. In addition to this, I would also like to thank him for listening to me when I needed it the most, being able to understand how I feel and provide the advice that kept me going throughout this difficult year.

I would also like to thank the University of Wisconsin-Madison for providing this data-set.

Statement of Originality

The work presented in this thesis/dissertation is entirely from the studies of the individual student, except where otherwise stated. Where derivations are presented and the origin of the work is either wholly or in part from other sources, then full reference is given to the original author. This work has not been presented previously for any degree, nor is it at present under consideration by any other degree awarding body.

Student:

Darsh Jadhav

Statement of Availability

I hereby acknowledge the availability of any part of this thesis/dissertation for viewing, photocopying or incorporation into future studies, providing that full reference is given to the origins of any information contained herein. I further give permission for a copy of this work to be deposited with the Bangor University Institutional Digital Repository, and/or in any other repository authorised for use by Bangor University and where necessary have gained the required permissions for the use of third party material. I acknowledge that Bangor University may make the title and a summary of this thesis/dissertation freely available.

Student:

Darsh Jadhav

Abstract

Learning analytics is a rapidly growing field within the area of data science, having the ability to improve student outcomes by optimising learning environments. This piece of work aims to examine the links between course structure and student performance, so that these results can be reported back to educators and used to make informed decisions about learning environments for students. This is done following a data science process, using feature selection to narrow down the relevant subsets of features in order to improve the predictive power of the machine learning models. The results from this work show that there is a weak link between total section counts and the evaluation class metric (class variable for student performance). This weak link is derived from its ability to increase the predictive power of a model (highest observe increase in accuracy = 9.77%), furthermore, the total section counts were part of the subsets of features which provided the highest accuracy and F-Measure results during the classification task. Many different subsets of features were tested to find a model with the highest predictive power, and the results showed that the initial data-set provided the best predictive power (65.51% accuracy and 0.640 overall F-Measure). These results told us that this data-set was unable to provide a subset of features which would give sufficient evidence to link course structure with student performance.

Contents

1 Introduction **ix**

1.1 Problem ix

1.2 Aim ix

1.3 Objectives x

2 Literature Review **xi**

2.1 Learning Analytics xi

2.2 Data Science xiii

2.3 Machine Learning xv

2.3.1 Types of Machine Learning xv

2.4 Feature Selection xvii

2.4.1 Filter Approach xvii

2.4.2 Wrapper Approach xviii

2.4.3 Generation Procedure xix

2.4.4 Evaluation Functions xix

2.4.5 Classifier Metrics xx

2.5 Training and Testing Protocols xxii

3 Model Design **xxiii**

3.1 Methodology xxiii

3.2 Metrics xxiv

3.2.1 Course Structure Metrics xxiv

3.2.2 Evaluation Class Metric xxv

3.3 Feature Selection xxvii

3.4 Classifier Experimentation xxxviii

4 Results and Evaluation **xliv**

4.1 Results xliv

4.2 Evaluation xlviii

5 Conclusion **xlix**

Bibliography **xlix**

6 Appendix **lv**

6.1 Distribution of Instances in Each Sample lv

6.2 List of Metrics/Features lvii

List of Tables

2.1 This is an example of a Confusion Matrix for a binary class problem (Positive / Negative). The rows represent the true classes and the columns represent the predicted classes. The numbers in each cell represent how many instances were assigned by the classifier for that particular class. Each cell in a Confusion Matrix is linked have a meaning, these meanings are put in brackets in the Confusion Matrix. (TP = True Positive, FN = False Negative, FP = False Positive, TN = True Negative). xxi

3.1 Experiment results on a random sub-sample when using the Random Forest classifier with LOO-CV, and the initial feature set. (TP = True Positive, FP = False Positive, AUC = Area Under [ROC] Curve) xxviii

3.2 Experiment results on a random sub-sample when using the Random Forest classifier with Resubstitution, and the initial feature set. (TP = True Positive, FP = False Positive, AUC = Area Under [ROC] Curve) . . xxix

3.3 Experiment results on the whole data-set containing all instances, when using the Random Forest classifier with Resubstitution, and the initial feature set. (TP = True Positive, FP = False Positive, AUC = Area Under [ROC] Curve) xxx

3.4 Experimental results when testing different subsets of features using random sub-samples, and C4.5/LOO-CV combination. Feature 55 is not included in the features selected list as it is the predicted variable (Evaluation Class Metric). Features Selected column uses index values for the features selected, the corresponding features for the indexes are shown in Table 6.2. The Feature Subset ID is used in order to refer to each feature subset during the report. .xxxiii

3.5 Experimental results when testing subsets of features found in Table 3.4, using a combination of C.45/Resubstitution on the whole data-set containing all instances. Feature 55 is not included in the features selected list as it is the predicted variable (Evaluation Class Metric). Features Selected column uses index values for the features selected, the corresponding features for the indexes are shown in Table 6.2. The Feature Subset ID is used in order to refer to each feature subset during the report. xxxvi

3.6 Experimental results showing the impact of total section counts (features 49-54) on the accuracy and F-Measures using C4.5 Decision Tree classifier and the different sample and protocol types (LOO-CV is used for random sub-sample, and resubstitution is used for the whole data-set. This is due to hardware constraints). Feature 55 is not included in the features selected list as it is the predicted variable (Evaluation Class Metric). Features Selected column uses index values for the features selected, the corresponding features for the indexes are shown in Table 6.2. The Feature Subset ID is used in order to refer to each feature subset during the report. (Resub. = Resubstitution, ? = Mathematically Undefined). xxxvii

3.7 These are the list of features used during the classification experiments. The Feature Subset ID is an identifier that is used during the report, the features are index values that correspond to Table 6.2. xxxviii

3.8 Results from testing the top 3 subsets of features (from Table 3.5) using different classifiers with resubstitution, using the whole data-set containing all instances. (Feature Subset ID is linked in Table 3.5, ? = Mathematically Undefined). xl

3.9 Confusion Matrix of Multilayer Perceptron (with resubstitution) test using feature subset 3 (ID). To understand how Confusion Matrices work, refer back to Table 2.1. xlii

6.1 Results comparing each random sub-sample with the whole data-set (consisting of all the instances), calculating the percentage of all instances in each class, the maximum difference (in percentage) between the random sub-sample and the whole data-set, and the standard deviation of all the samples. (N = Number of Instances, % = Percentage of instances in each class for each sample, Max % Diff (Δ%) = The maximum difference in percentage between random sub-samples and the whole data-set, $\Sigma(N)$ = Sum of all instances from each class). lvi

6.2 A list of all Metrics/Features used (Feature 55 is the predicted/class variable). lvii

6.2 A list of all Metrics/Features used (Feature 55 is the predicted/class variable). lviii

List of Figures

3.1 This graph shows the distribution of data in each class label from the output of the Evaluation Class Metric xxvii

4.1 This graph shows the difference in accuracy when the total section counts are included and excluded from the subset. This data is from Table 3.6 . xlv

4.2 This graph shows the accuracy achieved by each subset of results (using the whole data-set) when using a C4.5 Decision Tree Classifier with Resubstitution. The choice of axis (accuracy) scaling is to show the slight differences in accuracy, it would be more difficult to see the difference if the axis were scaled from 0-100%. This data is from Table 3.5. xlvi

4.3 This graph shows the accuracy achieved by the top three subsets of results (results shown in Table 3.5) when using the whole data-set (containing all instances), and when testing them using different classifiers. A table of results below the bars show the in-depth accuracy achieved by each combination of feature subsets and classifiers. This data is from Table 3.8.xlvi

Chapter 1

Introduction

Learning analytics is referred to as an area of research within data science regarding the examination and analysis of educational data [1]. The interpretation of the results allows institutions to provide the high quality education, thus providing additional support for student learning. Learning analytics has the ability to massively improve the way courses are taught, students with learning difficulties are identified, and eventually student success [2].

1.1 Problem

Students invest significant amounts of money and time in their education, yet not all students receive the best possible grades [3]. As the higher education industry has been marketised, there is a large introduction or rise of tuition fees [4], this makes it largely important to ensure that all students have the best possible chance in achieving the highest grades possible as they are investing large amounts of money and time into their education. Whilst poor student outcomes can be the fault of the student (possibly due to poor attendance, lack of motivation, and personal issues), it may also be down to other external factors, such as: parents, educators, institution, and educational system [5]. There may be certain factors from the institution that may have a large impact on student performance, a factor that will be extensively examined is course structure.

1.2 Aim

The aim of this research is to identify which parts of a course's structure have an influence on student performance. This research intends to provide more feedback to institutions and students, giving them additional guidance when they (institutions) structuring their courses, and when they (students) attend higher education.

1.3 Objectives

In order to understand which parts of course structure have an influence on student performance, there must be a process to analyse this data. In order to do so, the OSEMN framework (obtain, scrub, examine, model, interpret) has been adopted.

Conduct feature selection to identify particular subsets of features which have higher predictive powers compared to others, and find individual features that are able to significantly improve the predictive power of a model.

Machine learning models can be used in order to identify which course structures improve student performance, however, not all machine learning models will fit this problem, this means that it is necessary to evaluate which machine learning models suit best to this problem.

Chapter 2

Literature Review

2.1 Learning Analytics

Learning Analytics is an important field that refers to the investigation of data in order to report its findings in the context of learning [6]. Rebecca Ferguson mentions that the definition of learning analytics tends to come with two assumptions: using existing data which is readable by a machine, and the techniques used to handle the data can be applied for big data [7]. Big data is vasts amounts of data which tend to be too complex for a regular machine to process, a fundamental challenge that applications face when dealing with big data is the ability to extract important information from the large volumes of data available, making it important to evolve and adapt to make changes to future actions [8]. The purpose of this research is to extract information from data in order to provide more feedback to learners in the area of higher education. The benefits of learning analytics are that they have the ability to increase student retention and improve student performance [9]. Throughout this research, course structure and student performance is mentioned. What is meant by course structure and student performance? When referring to course structure, it can be broken down into many different factors, some include: course teaching methods, which days of the week are being taught on, which instructor will be teaching, and which room will the lesson be taught on. The objective is to assess these metrics (not limited to) and try and examine whether there are any specific course structures that may have an influence on student performance. Student performance is referred to as the grades achieved by the students studying the particular course.

Ben J. Arbaugh used statistical models to investigate whether there was an *optimal* course design for online MBA courses, he investigated into the technological and pedagogical characteristics of these online courses [10]. When investigating into his work, he has clearly stated the statistical approaches to this work. Statistical models are commonly used when investigating correlations between features, these models were able to support or disprove the hypotheses presented. Arbaugh did mention however that there were some limitations to his results, for instance, he stated that there was no “actual" measure for “student perceived learning" (one of the class variables), meaning that whilst he was able to find a correlation between both of his dependant variables, there might be no possible way for him to predict student learning. Whilst understanding Arbaugh’s methods, it has been clear that if there was some sort of nominal/numerical target class, it would be possible to adopt machine learning models rather than using statistical models in order to run supervised learning models for classification. This would allow the precision, recall, and accuracy to be tested, providing a better insight to the effectiveness of the models used. Another relevant study was conducted by Jasmine Paul and Felicia Jefferson. The aim of the study was to conduct a comparative analysis of student performance when comparing face to face to online course studies. They decided to set specific course structures, one design for face to face learning, and one for online course learning [11]. In this study, they decided to not use personal characteristics of students for the first question, but then adopted these variables for the second question (such as race, gender, ethnicity), but more rely on the effectiveness of the two types of learning structures. The metrics used for this study were the grades achieved by each student using a different course design (face to face or online learning). As they were using grade counts, they opted to use statistical tests, such as the traditional chi-squared test. What they found was that there were no significant differences in performance between the two course designs. A limitation they outlined was the lack of data (small sample size). A closely related study to this project was conducted by Shanna Smith Jaggers and Di Xu, they investigated how online course design features influence student performance, they investigate into any particular course design features that may have an impact on student performance, and how large of an impact do these features have on student learning outcomes [12]. What they managed to find was that interpersonal interactions were able to predict student grades in the course. They also mention that courses that utilised learning technologies, rather than reading intensive courses, did not have an impact on the student grades. Once again, a limitation that this type of research has come across is the lack of data (small sample size)

2.2 Data Science

Data science is a field of study that can be defined as the collection and analysis of data for information extraction [13]. Data science adopts many different machine learning algorithms, scientific methods and processes in order to gain additional knowledge from data which may be difficult by other means.

Data can come in three main categories: structured, semi-structured, and unstructured. Structured data refers to data which is highly organised, this compromises of explicit data types that are easily processed by a machine. Semi-structured data is organised data to an extent, this data can be moderately organised. Typical methods of organising semi-structured data involve using Extensible Markup Language (XML) or Resource Description Framework (RDF). A common example of semi-structured data is email. Unstructured data tends to be referred as all data that is not structured, examples of this include audio, video, and sensor data.

It is import to assess this data, making sure that there is no dirty data. This dirty data could be any type of inconsistencies within the data which may take away the integrity of the data. In order to make sure this data is clean for the machine to run the machine learning experiments, there must be a process that is followed to ensure that the data used is clean and relevant for the problem that is being solved. Hilary Mason and Chris Wiggins wrote an article regarding the general processes that each data scientist should be familiar with, they described a framework referred to as the OSEMN framework, OSEMN is an acronym for obtain, scrub, explore, model, and interpret [14].

The OSEMN framework is arranged in chronological order of the journey tackling a data science problem. Obtaining the data is the first step in solving a data science problem, being able to gather sufficient data which will be used for the machine learning models further down the process in order to solve the problem at hand. Whilst collecting data may seem straightforward, it is essential to ensure that this data can be used, this includes: making sure there are no copyrights in place, and ensuring that the data does not contain personal information of individuals (as this could result in legal implications).

The second step in the OSEMN framework is the scrubbing the data. Scrubbing the data includes analysing the data to find any irregularities and noise in the data, it is an essential step of the process as dirty data (data that consists of irregularities and noise) needs to be in a correct format for the machine to process the data, furthermore, dirty data will impact the performance of the machine learning model. Once the data is scrubbed so it is readable by both human and machine, the next step is to explore the data.

Exploring the data allows the human to understand the data, what features need to be tested in certain experiments, running feature selection to understand which features are relevant. The exploring step allows investigation into the finding the best possible machine learning model, this will require using training (learning) and testing data to test the model for predictive accuracy. Training data is used for the machine learning model to learn from the data, this allows the model to identify any patterns within the data which may improve prediction accuracy. Testing data is used when testing the model, this data will not present the classes to the model until after the model has given its predictions, the misclassified objects will account to the accuracy of the model.

The modelling step entails rigorous testing, ensuring that the model is able to provide sufficient evidence to provide an answer to a data-science problem. This step requires constant fine tuning of combinations of classifiers and training/testing protocols in order to provide meaningful results, ensuring that models are not overfitting to the data, and any imbalanced class problems handled appropriately. The models will test the subsets of features that were produced during feature selection, these subsets need to be tested in order to evaluate their worth, so that the modelling process are being training on meaningful data.

The final step of the OSEMN process is the interpretation of results. Interpreting the results from modelling will provide answers to the project's hypothesis. Furthermore, the results from the modelling stage will be in forms of numerical data, whilst displaying this data may be meaningful to a data-scientist, the results need to be communicated with the audience effectively. Methods of doing so include data visualisation and results tables with interpret-able results. This will make the findings of the project easily comprehensible for all audiences.

2.3 Machine Learning

Machine learning is a multi-disciplinary scientific field, it combines ideas from several different fields such as neuroscience, mathematics, and physics [15]. Machine learning uses algorithms and data structures to learn from training data, resulting in the ability to make predictions [16]. This tends to be achieved by identifying patterns within the data which humans tend to not be able to identify [17]. A data-set of size $\{X_1, X_2\}$ has X_1 objects and X_2 features (or attributes). A feature is a measurable characteristic of the data, also known as a column in the data-set. An object is a group of values populating the data, also known as the rows in data-set. Machine learning has developed rapidly from theories to reality. The likes of Alan Turing contributing to many different subject areas such as mathematics, biology, philosophy, and to future areas now known as computer science, Artificial Intelligence, and Machine Learning [18]. Turing's paper talked about 'automatic machines' being machines which are “completely" determined by configuration at each step of motion [19], this was considered to be a foundation of computer science, these machines were first formally named by Alonzo Church, 1936, referring to them as 'Turing Machines' [20]. The first reality of machine learning was in the form of representing how neurons of a brain work using electrical circuits, this was a simple model introduced by mathematician Walter Pitts and neurophysiologist Warren McCulloch [21], was a representation of neural networks. The applications of machine learning have evolved significantly, many learning analytics applications utilise machine learning in order to gain more information from educational data [22]. S. Kotsiantis et al. used machine learning to predict student performance in distance learning [23]. Using many different machine learning models (Naive Bayes, C4.5 Decision Tree, BP, Sequential Minimal Optimisation (SMO), Nearest Neighbour, and Logistic Regression), they managed to predict students with poor performance, specifically which student would pass or fail (with sufficient precision). Whilst they were able to achieve a 70.51% prediction accuracy with the Naive Bayes algorithm, they mentioned that in order for their prediction accuracy to increase, it would require more objects in the data-set [23].

2.3.1 Types of Machine Learning

There tends to be two primarily used types of machine learning, supervised (or predictive) and unsupervised learning. Supervised learning is an approach which uses a labelled data-set. The model will use training labelled data to identify any patterns within this data, then subsequently apply the model onto testing data. This testing data will be classified by identifying any patterns within the data. When the aim of the machine learning model is to correctly predict the output of Y, a categorical feature of class labels, the problem at hand is referred to as a pattern recognition or classification problem [24]. Classification problems are known to be discrete, meaning that each individual object will belong to a singular class [15]. The types of classification algorithms used depends on the type of classes, whether it is mutually exclusive or not. Mutual exclusivity is the logic to whether objects are either exclusive to one class or can belong in multiple. When classification is said to be mutually exclusive, it means that objects can only belong to one class. If classification is not mutually exclusive, then the objects can belong to multiple classes. An example of a mutually exclusive event is the outcomes of a coin flip, it is impossible for both outcomes to occur simultaneously. The classes used in the data used for this research can be said to be mutually exclusive as they are percentages of grade distribution. Another variant of the supervised learning problem is known as a regression problem, this is the prediction of a continuous output for X_2. This means that the classifier is given training data, and the predicted output from the testing data will be a precise value [25]. Unsupervised learning is the second primarily used type of machine learning. Unsupervised learning aims to use unlabelled input data to identify any patterns within the data without external input, in contrast to supervised learning using training and testing data to predict explicit target outputs [26]. This is commonly used in clustering algorithms and dimensionality reduction [24]. Clustering algorithms characterise the data into clusters of lower dimensions than can be seen as separable [26]. The goal of clustering is to identify the best learning model that will result in an optimal distribution of clusters and being able to associate each data point to the cluster of best fit [24]. Dimensionality reduction can be defined as transforming high-dimensional data into relevant lower-dimensional data [27]. Hypothetically, this reduced dimensionality data has more significant value as it strays away from the "curse of high-dimensional data" [28], the curse of dimensionality is the problem that as the number of features in a data-set increases, the amount of time for an algorithm to complete its task will increase to a large extent, in some cases exponentially [29]. The disadvantages of high-dimensional data have an impact on the effectiveness of clustering, and the efficiency and optimisation of the algorithms used.

2.4 Feature Selection

Feature selection is closely related to dimensionality reduction, in the relation that it aims to reduce the features present in the data-set, however, they are two different methods in doing so. Feature selection (also known as attribute selection) is the method in picking a subset of features that are linked to the target concept [30]. There are many different conceptual definitions of feature selection that have been covered by many different authors, however, these definition tend to be closely related. Some definitions include:

1. The aim of feature selection is to select a smaller subset of data (m) which derives from a data-set with a larger amount of (n) features in order to improve the criterion value (this could be the error rate of the classifier) for all subsets of size (m) [31].

2. Feature selection aims to select a subset of features (m) from a larger data-set in order to improve accuracy of predictions made by the classifier [32].

Feature selection is very important when it comes to learning models as they may not work as well on larger data-sets due to undesirable features. Feature selection is able to provide a better insight to the problem that is being solved, this is done by identifying and reducing the amount of redundant and irrelevant features in the data-set, and identifying the important features which link to the target concept [30]. Feature selection is a large portion of this research, it will be used to examine course structure and which aspects of course structure may have an influence on the grades that the students achieve. In order to do this, an feature selection approach needs to be followed, there are two main different types of approaches: filter and wrapper approaches [33].

2.4.1 Filter Approach

The filter approach adopts a strategy to select the minimal feature subset, this is to minimise the feature bias that may be presented. The filter approach evaluates features and their importance based on characteristics in the data, this means that it does not use learning algorithms. This feature selection method tend to be less computationally expensive and produce results quicker in comparison to the wrapper approach [34].

When using filter methods, it is important that there is a sufficient amount of data as the statistical models used would not provide the best possible results.

2.4.2 Wrapper Approach

The wrapper approach use inductive algorithms to evaluate the value of a feature subset. This method is seen to be “a superior alternative" when approaching supervised learning problems [35]. The wrapper approach is relatively simple to implement, however, this process does execute abundantly when retrieving results with a high computational cost [35]. This method has external validation as it is used in supervised learning, the class labels are available but only for comparison to estimate the accuracy of the classifier used, what this means is that the process in which the classifier learns from the data does not have access to the classes until after the testing process is over.

When examining the link between course structure and performance, the adopted feature selection method is the wrapper approach. The reason for adopting the wrapper approach is linking back to the objectives of this research: being able to perform feature selection to understand which aspects of course structure have an influence on student performance, and being able to maximise accuracy when classifying objects in the data-set. Feature selection using supervised learning techniques aim to maximise the accuracy of the classification model, in addition, wrapper approaches result in better results compared to filter approaches as they use classifiers to evaluate each feature [36]. When performing feature selection, it is important to evaluate the methods that will be used, will the feature selection model successfully identify the relevant and irrelevant features? Is there an optimal balance between performance and efficiency of the search methods? Will the model be able to provide an answer to the problem at hand? To evaluate which feature selection model would be best fit for the problem, Manoranjan Dash and Huan Liu categorised this process into four main steps: generation procedure, evaluation function, stopping criterion, and validation procedure [30]. For this research, it is necessary to understand generation procedure and evaluation function as these steps are assessed when choosing the feature selection models.

2.4.3 Generation Procedure

The generation procedure is the method to generate the succeeding feature candidate, it is also referred to as the search procedure. The process can either begin with zero features, all the features, or a randomly generated subset of features. There are three search methods that will be used when building our feature selection model: best first, greedy step-wise, and ranker search method. Best first search is a search method that traverses through the nodes in the data and selects the best possible feature with the lowest cost at that moment based on outlined rules [37]. Greedy step-wise search is a type of search method which traverses through the data to find best or worst feature at that current moment, this will be determined by the evaluation function. Greedy step-wise and best first search method are heuristic search approaches, they uses evaluation functions to take an informed decision as to which features should be the succeeding candidate feature, these search methods are effective at balancing between performance and efficiency which makes them useful for feature selection. Ranker search method will rank all the features in order of best to worst in relation to the class, this will be based on the evaluation function used alongside the classifier.

2.4.4 Evaluation Functions

Evaluation functions are used to measure the performance and quality of the features selected or the subsets generated, comparing the new feature or subset with the previous one to determine which one resulted in a better outcome. There are several different ways in evaluating features, the evaluation functions used to examine the link between course structure and student performance are classifier error rate measures and dependence measures:

1. Classifier error rate measures: The classifier error rate measure are used in wrapper methods. This is because the classifier that aims to predict the class labels are unable to see the class labels during training. The features/subsets with the lower error rates are selected whereas the features/subsets with a higher error rate are deemed irrelevant. This type of evaluation function is known to have a high accuracy, however, this comes at a price where the time complexity is high [30]. Root mean squared error (RMSE) is commonly used to evaluate a machine learning model, whilst many authors lean towards using RMSE (e.g. Barnston, 1992 [38]; Dennison and Roberts, 2003 [39]; Chai

and Draxler, 2014 [40]), many tend to disagree and use mean average error (MAE) due to the work of Willmott and Matsuura, 2005; stating that there is an element of ambiguity with RMSE. RMSE and MAE are both measures of prediction error of the classifier. Calculating RMSE and MAE is as follows [40]:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_{pred} - X_{obs})^2} \tag{2.1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(X_{pred} - X_{obs})| \tag{2.2}$$

- X_{pred} is the predicted value.
- X_{obs} is the observed/actual value.
- n is the total number of objects.

2. Dependence measures: Dependence measures are also known as correlation measures, these are used to compare values from several features to the target class. The preferred feature would be assigned to the subset based on which feature correlates mostly with the target class [30]. Dependence measures are able to generalise problems as it uses correlation calculations for evaluation.

2.4.5 Classifier Metrics

When analysing the success of a classifier, there are several different metrics that are commonly used. A method of analysing the performance of a classifier is to look at a confusion matrix. A confusion matrix (of size c x c, where c is the number of classes), shows the true and predicted classifications for each instance in the data-set [41]. An example of a confusion matrix is shown in Table 2.1.

Assessing how well a classifier performed comes down to looking at the distribution of numbers in the main diagonal compared to the rest of the cells. A classifier that has performed perfectly will have all the predicted instances on the main diagonal, whilst being surrounding by zero instances predicted in the FP and FN cells. Using a

True \Predicted	Positive	Negative
Positive	1030 (TP)	21 (FN)
Negative	13 (FP)	42 (TN)

Table 2.1: This is an example of a Confusion Matrix for a binary class problem (Positive / Negative). The rows represent the true classes and the columns represent the predicted classes. The numbers in each cell represent how many instances were assigned by the classifier for that particular class. Each cell in a Confusion Matrix is linked have a meaning, these meanings are put in brackets in the Confusion Matrix. (TP = True Positive, FN = False Negative, FP = False Positive, TN = True Negative).

confusion matrix, common metrics that are used can be calculated, these are show in Eq. (2.3) to Eq. (2.7).

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \tag{2.3}$$

$$ErrorRate = \frac{FP+FN}{TP+FN+FP+TN} \tag{2.4}$$

$$Precision = \frac{TP}{TP+FP} \tag{2.5}$$

$$Recall/Sensitivity = \frac{TP}{TP+FN} \tag{2.6}$$

$$F-Measure/F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.7}$$

Accuracy and Error Rate are commonly used metrics, where accuracy measures how many instances were predicted correctly. Error rate is essentially the inverse of accuracy, where it measure the amount of incorrectly classified instances. Whilst these metrics can be used, they often do not tell the full story (especially when there is a unbalanced class problem). This is where the F-Measure/F1 Score is adopted. Precision is used to describe how a classifier is able to predict the number of relevant instances from all the predicted values (for that specific class). Recall describes the number of predicted instances that are in the the correct class. F-Measure (also referred as the F1 Score) is the harmonic weighted mean of Precision and Recall [42], this means that this single metric is able to use both properties (Precision and Recall) to provide a better understanding of the classification problem, this is why it is often adopted for imbalanced class problems [43].

2.5 Training and Testing Protocols

Resubstitution is the defined as the sampling technique that uses the training data as the testing data, meaning that the model will learn from the same data that it will be tested on. Overfitting is an error that occurs during modelling when a function is aligns too closely to a set of data, the resubstitution method is not a good choice as it does not generalise to a new set of unseen data, this means that the model might be prone to overfitting the data. n-fold cross validation is a sampling technique that splits the data into n folds (divisions), leaving one fold for testing data and $n - 1$ folds as training data. The n-fold cross validation technique is widely adopted when the intention is to test on unseen data, this results in less bias upon classifying the objects. Percentage split is a simple sampling technique where the training and testing data is split based on the pre-defined percentage splits. This method tends to be overshadowed by the n-fold cross validation technique due to lower bias when testing the machine learning model. Leave-one-out Cross Validation is a type of n-fold cross validation, where n-folds is the number of instances in the data-set. This means that 1 instance is used for testing, whilst the rest of the instances are used for training.

Chapter 3

Model Design

This chapter will outline and describe the process to examine course structure and if there are any specific designs that have an influence on student performance - how students within each course perform. This will include the examination of data, the selected metrics, feature selection, model development process.

3.1 Methodology

The primary aim of this work is to examine whether course structure has a link to student performance. This meant that the first step in this work was to examine all the possible data available, and metrics which would be most suitable for this work. After examining all the possible metrics, a set of metrics were adopted. There were a large number of possible metrics could be used, however, this was reduced into a smaller number which were obtainable and consist of suitable information.

Once the metrics were identified, it was time to start obtaining an appropriate data-set. This involved assessing different data-sets with regards to which ones have the relevant data which consists of course structure and student performance. The data obtained for this work is secondary data, provided by the University of Wisconsin-Madison. This data contained course information for all courses taught between the years of 2006-2017, the course information consisted of the how the course was structured, furthermore, it also had the grade distributions of students studying that course.

After obtaining a suitable data-set, it was possible to commence with early data explorations. There are two main goals for data exploration: examining the data to gain an understanding with all the features in the data-set, and identifying any possible relationships; and cleaning dirty data. This data can be referred to as dirty for a number of reasons, such as: features consisting of unusable or missing data and coding errors. When the data was obtained it was in several different tables, certain values within fields were not readable by the machine. Therefore, the data was prepared in a format that it was in one large data-set. When looking into the data, it was clear that there were going to be issues that would rise when the machine will process the data, i.e. special characters in fields and missing values which invalidated the field for our use. The data was cleaned using a relational database management tool, consequently reducing/removing the problems in the data.

Once the data-set was cleaned and explored, the data-set was now used in a series of tests. These tests seek to identify the best possible combination of the metrics and producing a model which gives us the highest possible F-Measure when classifying the objects into classes. These tests can be split into two sections, feature selection and classification. The results from these tests provide information as to which models are appropriate for the data, and which ones are not; resulting in a modification of the model parameters to improve the accuracy and F-Measures of these models. Once a review of the tests are made, it will be possible to make interpret the results and provide information which helps answer the question - is there a link between course structure and student performance?

3.2 Metrics

The aim of these experiments is to examine the links between course structure and student performance, specifically if there are any certain combinations of these metrics which have an influence on student performance. This means that it is required to have large amounts of information, and the variety of different metrics available.

3.2.1 Course Structure Metrics

The initial data-set consisted of vast amounts of information that could be useful when analysing course structure, however, during the data cleaning and exploration process, some metrics which were initially adopted were removed as some of them were not relevant to the question, and some consisted of large amounts of missing data. The metrics used focused on course structure, specifically how courses were taught on a daily basis. The metrics used aimed to help provide information as to whether certain aspects of these course structures could influence student performance.

The metrics used broke down the types of teaching methods used, these are referred to as section types. There are 6 different section types:

- Labs
- Lectures
- Discussions
- Field Work
- Seminars
- Independent Study

Additional metrics used were:

- Whether a specific teaching (section) type was used for a course
- What days of the week were the students taught (each day of the week is associated to a section)
- The total number of teachings for each course (this was broken down into each section type)

To see the detailed list of all metrics used, look at Table 6.2

3.2.2 Evaluation Class Metric

In order to examine which course design could have a link to student performance, it was necessary to devise an evaluation metric of student performance. This student evaluation metric could have been implemented in several different ways as the data-set consisted of an in-depth distribution of the grades achieved by the students for each individual course. Whilst there was the possibility to simply predict the grade counts of each course, it was decided that the grade distribution data would be quantised into categorical class (predicted) variable (consists of discrete values), doing so makes this a multi-class classification problem. Our predicted variable (class variable) is a nominal data type which has ten different classes (ranging from 0 to 10), where instances placed in class 10 are perceived to have the best possible performance, and instances placed in class 0 are perceived as least successful. The method in which it was decided to calculate student performance was to take the top three grades (A, AB, B) that can be achieved by students in each course and consider them to be successful student performance. These grades are taken as a percentage of the total grades achieved and placed in classes ranging from 0 to 10. A_i is the number of students that achieved grade A for the i^{th} course, the same applies for AB_i and B_i respective to their grade letter. N_i in traditional notation is used for the total number of instances in a data-set, however, for the equation supplied above, N_i is defined as the accumulation of counts for all the grades achieved for the i^{th} course.

$$EvaluationClassMetric = \frac{A_i + AB_i + B_i}{N_i} \tag{3.1}$$

As stated before, the output of the Evaluation Class Metric will give us a nominal value as to how well students have performed in a course, Fig. 3.1 provides visualisation to the student performance distribution for all of the courses in the data-set. When looking at the distribution, the majority of instances are in the top half of the classes, showing that a large amount of students are achieving the top grades. Whilst the higher student performance classes do have a greater density in data, there is still enough data in the lower classes, meaning that it will be possible to run tests which give sufficient results.

When looking at 3.1, it shows that there is not a large problem in terms of class balance. Whilst there is still a sufficient amount of data present in the lower classes, the use of F-Measure will be more meaningful in comparison to classification accuracy, as F-Measure will give us a better representation of the balance between precision and recall.

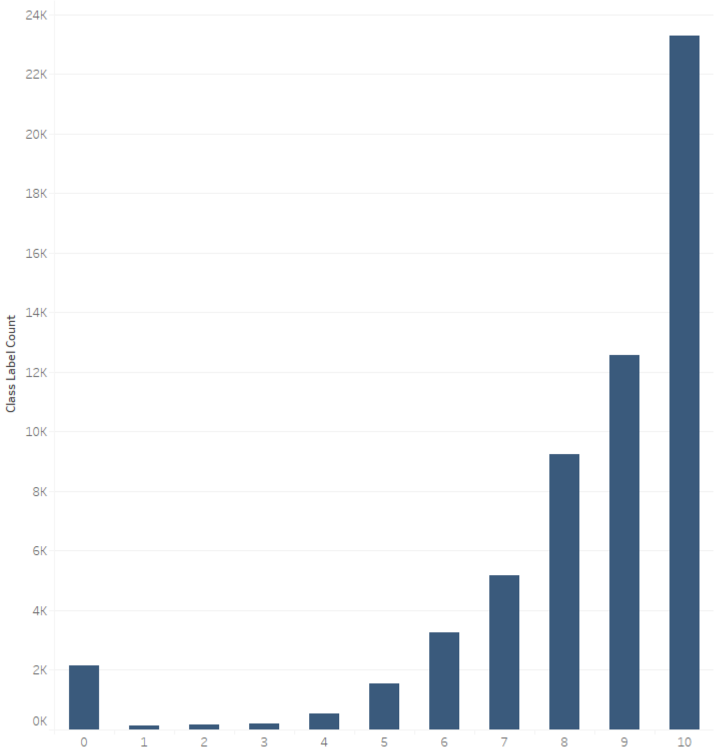


Figure 3.1: This graph shows the distribution of data in each class label from the output of the Evaluation Class Metric

3.3 Feature Selection

In order to ensure that the performance of the machine learning model is maximised, it is crucial to use feature selection. As defined, a core aim of feature selection is to improve the accuracy of a model by selecting a relevant subset of features from the original data-set [32]. In order to be able to identify which aspects of course structure have an influence on student performance, it is necessary to be able to find an initial link between the two as there needs to be some validation that it is possible to predict student performance based on specific course structures. This means that metrics stated in Section 3.2.1 need to have some correlation with the Evaluation Class Metric mentioned in Section 3.2.2. All of the experiments that are conducted will be making use of the Weka Machine Learning Toolkit [44]. When obtaining results, there will be certain results to analyse. As mentioned in Section 3.2.2, evaluation metric has multiple classes, meaning that this is a multi-class problem. The results of the classification task will give individual F-Measure values for each individual class. Weka will provide a weighted average of these individual F-Measures, giving an overall F-Measure of the experiment. These weightings are derived by calculating the proportions of each instance, in each class. This makes the model assume that each class has equal value when classifying the instances [45].

The initial experiment involved predicting the evaluation class metric (the predicted variable or class variable) by using the whole set of features. In order to this, it was decided to take random sub-samples (N = 5000) of the whole data-set, the random element means that the sub-samples will not necessarily be a complete representation of the whole data-set. This means that these random sub-samples will only be used initially, whilst the final testing will use the whole data-set (N = 58,234). This initial experiment was planned to use a Random Forest classifier with leave-one-out cross-validation (LOO-CV). Random Forest is a type of decision tree classifier, it uses an ensemble of decision trees, and these trees vote on the classification of each object (the tree growth is controlled by random vectors) [46]. The results from this experiment came back with an accuracy of 45.58%, and a weighted F-Measure value of 0.430. A table of these results is shown in Table 3.1. When looking at the results, it is evident that the accuracy and weighted average F-Measure is fairly low. When looking into the results for each individual class, it can be seen that the F-Measures across the board are poor, with a slight exception to class 10 being relatively higher than the other classes.

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	AUC
10	0.772	0.360	0.608	0.772	0.680	0.772
9	0.317	0.165	0.341	0.317	0.329	0.650
8	0.225	0.096	0.294	0.225	0.255	0.614
7	0.145	0.055	0.188	0.145	0.164	0.617
6	0.095	0.036	0.129	0.095	0.110	0.635
5	0.079	0.017	0.120	0.079	0.095	0.592
4	0.121	0.008	0.156	0.121	0.136	0.632
3	0.000	0.003	0.000	0.000	0.000	0.481
2	0.053	0.002	0.083	0.053	0.065	0.816
1	0.000	0.000	0.000	0.000	0.000	0.632
0	0.295	0.010	0.505	0.295	0.372	0.874
Weighted Average	0.456	0.208	0.417	0.456	0.430	0.698

Table 3.1: Experiment results on a random sub-sample when using the Random Forest classifier with LOO-CV, and the initial feature set. (TP = True Positive, FP = False Positive, AUC = Area Under [ROC] Curve)

The reason to use LOO-CV is to reduce overfitting. Resubstitution is prone to overfitting on data (smaller amounts of data tends to be more prone to overfitting). This can be down to the lack of variation when there is a smaller amount of data, in comparison to a larger amount of data. When using resubstitution instead of LOO-CV on the random sub-sample (using the same classifier and set of features), the accuracy majorly improves to 78.84% (change in accuracy = Δ33.26%), and a weighted F-Measure value of 0.783 (change in weighted average F-Measure = Δ0.353). A table of the results from the experiment using the Random Forest classifier with resubstitution is shown in Table 3.2. These results provide some reasoning as to why resubstitution is not used on the random sub-samples.

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	AUC
10	0.920	0.209	0.761	0.920	0.833	0.941
9	0.760	0.064	0.763	0.760	0.761	0.955
8	0.627	0.026	0.811	0.627	0.707	0.944
7	0.640	0.008	0.875	0.640	0.740	0.970
6	0.718	0.004	0.900	0.718	0.798	0.985
5	0.719	0.002	0.909	0.719	0.803	0.990
4	0.793	0.001	0.868	0.793	0.829	0.998
3	0.700	0.000	0.933	0.700	0.800	0.999
2	0.947	0.001	0.857	0.947	0.900	1.000
1	0.714	0.000	0.714	0.714	0.714	1.000
0	0.549	0.003	0.880	0.549	0.676	0.993
Weighted Average	0.787	0.106	0.796	0.787	0.783	0.953

Table 3.2: Experiment results on a random sub-sample when using the Random Forest classifier with Resubstitution, and the initial feature set. (TP = True Positive, FP = False Positive, AUC = Area Under [ROC] Curve)

Whilst resubstitution on smaller sub-samples of data does not provide meaningful results, it can be used on larger data-sets. This is because there is a large amount of variation in the large data-set, resulting in less bias when training the classifier, meaning that the likelihood of overfitting is minimal. When running the Random Forest classifier with resubstitution on the whole data-set, the accuracy result was 65.51%, and the overall F-Measure was 0.640. Table 3.3 shows the results of this experiment.

When conducting feature selection, the ideas were to run feature selection algorithms (such as classifier attribute evaluator, correlation attribute evaluator, and classifier subset evaluator) and make changes to improve the performance of the model when testing these subsets of features (aiming to perform better than the model when using the initial data-set with all the features retained). Now that there is an idea as to how the classification task performs when feature selection has not taken place (Table 3.1), the next stage is to conduct feature selection and test each subset of features, firstly testing using random sub-samples, then testing using all instances in the initial data-set.

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	AUC
10	0.890	0.304	0.662	0.890	0.759	0.888
9	0.611	0.124	0.576	0.611	0.593	0.878
8	0.414	0.038	0.670	0.414	0.512	0.862
7	0.454	0.014	0.763	0.454	0.569	0.916
6	0.493	0.008	0.788	0.493	0.606	0.937
5	0.509	0.004	0.789	0.509	0.619	0.960
4	0.474	0.001	0.796	0.474	0.595	0.983
3	0.410	0.001	0.553	0.410	0.471	0.991
2	0.375	0.001	0.640	0.375	0.473	0.996
1	0.328	0.001	0.577	0.328	0.418	0.997
0	0.342	0.007	0.659	0.342	0.450	0.977
Weighted Average	0.655	0.156	0.664	0.655	0.640	0.894

Table 3.3: Experiment results on the whole data-set containing all instances, when using the Random Forest classifier with Resubstitution, and the initial feature set. (TP = True Positive, FP = False Positive, AUC = Area Under [ROC] Curve)

The initial feature selection experiment utilised a Nearest Neighbour (1-NN) classifier and adopted a Best-First search method. The Nearest Neighbour classifier works by assigning labels to new objects based on other objects that are deemed to be the closest match. The closest match is determined by using measurable concepts, some examples include: Euclidean distance and Hamming distance [47]. This experiment resulted in 15 features being removed as they were deemed to be redundant (39 features left in the subset), this is because they provided no changes in accuracy or F-Measure for any subset of features tested. Once these features were removed, addition exploration and changes were conducted in order to find a subset of features which resulted in high accuracy and F-Measure values. This was done using a pruned C4.5 pruned Decision Tree classifier with a LOO-CV protocol. A Decision Tree classifier starts by using a set number of cases, this will then be used to create tree data structure, which can be used for classifying new cases. Cases are described by the set of features used. These decision trees split the cases into smaller subsets whilst developing the decision tree. The C4.5 decision trees tend to be a popular classifier used for classification tasks due to its computation time and reliability, this is because the C4.5 Decision Tree classifier produces small and accurate trees [48]. C4.5 Decision Trees also utilise a pruning method, this means that they will grow a tree and then delete parts of the tree in order to generalise the classifier, these parts can be sub-trees or branches. Table 3.4 shows the accuracy and F-Measure values for the subsets of features tested which gave the highest accuracy and F-Measure values when using the C4.5/LOO-CV combination.

When taking a look at Table 3.4, it can be seen that the accuracy and F-Measure values are very low, especially with the cases of class 1 and 3 achieving an F-Measure of 0.000 in every test. This is due to the highly imbalanced class problem that is present for classes 1, 2, and 3. This results in poor F-Measure values for each of these classes. To ensure that the random sub-samples are not causing this problem (as this could be down to highly varied distributions of instances in the classes for each random sub-sample compared to the whole data-set), some calculations are performed to measure the percentage of instances in each class, and the differences between the distributions of each class (comparing the highest difference to the whole data-set). The results of these calculations are in Table 6.1. When looking at these results, it can be seen that the maximum difference in percentage of instances in each class (comparing the random sub-samples and the whole data-set) is very low, with the highest value being -1.95% (to 3 significant figures) in class 10. Furthermore, looking at the standard deviations for all classes, it can be seen that the values are below 1, the highest standard deviation value is 0.997 (to 3 significant figures). These calculations show that the distribution of instances in each class is very similar for each random sub-sample (compared to the whole data-set).

When looking at Table 3.4, the results showed that feature subset 3 (ID) resulted in the highest accuracy and F-Measure values (accuracy = 49.40%, weighted average F-Measure = 0.457). This subset of features uses a range of different features, these features include:

- Labs (T/F)
- Labs on Monday - Friday
- Lectures (T/F)
- Lectures on Monday - Saturday
- Discussions (T/F)
- Discussions on Monday - Friday
- Field Work (T/F)
- Field Work on Monday to Saturday
- Seminars (T/F)
- Seminars Monday - Friday
- Independent Study (T/F)

- Total Counts of each section (Labs, Lectures, Discussion, Field Work, Seminars, Independent Study)

The features that were removed from the initial feature subset:

- Labs on Saturday and Sunday
- Lectures on Sunday
- Discussions on Saturday and Sunday
- Field Work on Sunday
- Seminars on Saturday and Sunday
- Independent Study on Monday - Sunday

Feature Subset ID	Features Selected (Index Values)	Number of Features Retained (n)	Random Sub-Sample (1-4)	Accuracy (%)	*Per-Class F-Measure*											*Weighted Average*
					10	9	8	7	6	5	4	3	2	1	0	
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	39	2	49.40	0.712	0.393	0.227	0.181	0.238	0.078	0.289	0.000	0.000	0.000	0.475	0.457
9	2-8, 10-16, 18-24, 26-32, 34-40, 42-54	48	1	49.04	0.710	0.376	0.173	0.151	0.162	0.137	0.073	0.000	0.162	0.000	0.329	0.441
2	2-5, 10-15, 18-19, 21, 29, 49-51, 53-54	19	3	48.78	0.705	0.376	0.236	0.174	0.207	0.147	0.289	0.160	0.000	0.000	0.428	0.451
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	39	3	48.48	0.699	0.389	0.243	0.230	0.147	0.153	0.237	0.000	0.364	0.000	0.301	0.449
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	39	1	47.74	0.706	0.330	0.167	0.158	0.170	0.103	0.078	0.000	0.158	0.000	0.329	0.429
1	1-6, 10-15, 17-22, 29-31, 33-38, 49-54	33	1	47.72	0.704	0.333	0.167	0.171	0.187	0.128	0.079	0.000	0.103	0.000	0.245	0.428
9	2-8, 10-16, 18-24, 26-32, 34-40, 42-54	48	4	47.48	0.705	0.364	0.225	0.177	0.140	0.128	0.228	0.000	0.364	0.000	0.301	0.438
5	1, 3-4, 6, 8, 10-15, 17-21, 26-28, 30, 49, 51, 53-54	24	1	47.46	0.709	0.336	0.158	0.153	0.159	0.130	0.069	0.000	0.108	0.000	0.316	0.429
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	39	4	47.30	0.702	0.345	0.246	0.179	0.183	0.136	0.104	0.000	0.000	0.000	0.361	0.438
6	4, 12-14, 18, 25, 28, 30, 49-51, 53-54	13	4	47.16	0.687	0.369	0.185	0.193	0.173	0.090	0.054	0.000	0.000	0.000	0.375	0.427
7	49-54	6	4	47.06	0.699	0.349	0.212	0.177	0.192	0.120	0.125	0.000	0.000	0.000	0.377	0.433
2	2-5, 10-15, 18-19, 21, 29, 49-51, 53-54	19	1	47.04	0.708	0.332	0.192	0.116	0.099	0.073	0.073	0.000	0.061	0.000	0.287	0.424
4	1, 3-4, 12-14, 17-22, 49, 53-54	15	2	46.99	0.683	0.363	0.218	0.182	0.202	0.143	0.107	0.000	0.000	0.000	0.169	0.424
6	4, 12-14, 18, 25, 28, 30, 49-51, 53-54	13	1	46.94	0.692	0.319	0.170	0.113	0.085	0.116	0.091	0.000	0.171	0.000	0.324	0.414
4	1, 3-4, 12-14, 17-22, 49, 53-54	15	1	46.90	0.695	0.300	0.161	0.148	0.168	0.112	0.053	0.000	0.000	0.000	0.270	0.413
6	4, 12-14, 18, 25, 28, 30, 49-51, 53-54	13	2	46.78	0.696	0.386	0.146	0.097	0.116	0.113	0.082	0.000	0.087	0.000	0.063	0.409
7	49-54	6	2	46.78	0.697	0.402	0.111	0.088	0.089	0.124	0.000	0.000	0.095	0.000	0.050	0.405
6	4, 12-14, 18, 25, 28, 30, 49-51, 53-54	13	3	46.68	0.682	0.388	0.204	0.097	0.076	0.095	0.000	0.000	0.000	0.000	0.471	0.420
2	2-5, 10-15, 18-19, 21, 29, 49-51, 53-54	19	2	46.66	0.695	0.369	0.168	0.199	0.113	0.086	0.070	0.000	0.087	0.000	0.090	0.418
1	1-6, 10-15, 17-22, 29-31, 33-38, 49-54	33	2	46.56	0.698	0.377	0.188	0.160	0.149	0.088	0.063	0.000	0.000	0.000	0.092	0.422
8	1, 9, 17, 25, 33, 41, 49-54	12	2	46.34	0.688	0.395	0.150	0.183	0.068	0.110	0.034	0.000	0.091	0.000	0.050	0.413
8	1, 9, 17, 25, 33, 41, 49-54	12	1	46.34	0.672	0.373	0.138	0.112	0.149	0.058	0.000	0.000	0.000	0.000	0.479	0.406
5	1, 3-4, 6, 8, 10-15, 17-21, 26-28, 30, 49, 51, 53-54	24	4	46.18	0.689	0.364	0.214	0.120	0.103	0.158	0.057	0.000	0.138	0.000	0.435	0.423
7	49-54	6	3	46.18	0.683	0.353	0.113	0.055	0.051	0.047	0.000	0.000	0.000	0.000	0.214	0.395
8	1, 9, 17, 25, 33, 41, 49-54	12	3	46.16	0.671	0.376	0.139	0.076	0.151	0.049	0.000	0.000	0.000	0.000	0.469	0.469
1	1-6, 10-15, 17-22, 29-31, 33-38, 49-54	33	4	46.12	0.689	0.360	0.161	0.188	0.158	0.129	0.098	0.000	0.000	0.000	0.346	0.422
7	49-54	6	1	46.10	0.680	0.343	0.135	0.036	0.051	0.055	0.000	0.000	0.065	0.000	0.249	0.395
8	1, 9, 17, 25, 33, 41, 49-54	12	4	45.96	0.685	0.377	0.146	0.193	0.097	0.134	0.036	0.000	0.087	0.000	0.060	0.410
4	1, 3-4, 12-14, 17-22, 49, 53-54	15	3	45.94	0.698	0.355	0.174	0.165	0.169	0.075	0.000	0.000	0.000	0.000	0.118	0.416
2	2-5, 10-15, 18-19, 21, 29, 49-51, 53-54	19	4	45.90	0.699	0.311	0.220	0.172	0.141	0.073	0.056	0.000	0.000	0.000	0.380	0.421
5	1, 3-4, 6, 8, 10-15, 17-21, 26-28, 30, 49, 51, 53-54	24	2	45.84	0.686	0.361	0.200	0.121	0.102	0.145	0.031	0.000	0.154	0.000	0.416	0.418
4	1, 3-4, 12-14, 17-22, 49, 53-54	15	4	45.76	0.691	0.357	0.176	0.189	0.139	0.088	0.034	0.000	0.080	0.000	0.071	0.414
9	2-8, 10-16, 18-24, 26-32, 34-40, 42-54	48	3	45.70	0.684	0.351	0.226	0.112	0.086	0.18	0.000	0.000	0.077	0.000	0.484	0.421
1	1-6, 10-15, 17-22, 29-31, 33-38, 49-54	33	3	45.36	0.691	0.350	0.182	0.099	0.109	0.120	0.052	0.000	0.080	0.000	0.480	0.415
9	2-8, 10-16, 18-24, 26-32, 34-40, 42-54	48	2	45.34	0.694	0.351	0.163	0.160	0.157	0.088	0.066	0.000	0.000	0.000	0.091	0.411
5	1, 3-4, 6, 8, 10-15, 17-21, 26-28, 30, 49, 51, 53-54	24	3	45.16	0.683	0.344	0.191	0.091	0.105	0.186	0.078	0.000	0.000	0.000	0.462	0.412

Table 3.4: Experimental results when testing different subsets of features using random sub-samples, and C4.5/LOO-CV combination. Feature 55 is not included in the features selected list as it is the predicted variable (Evaluation Class Metric). Features Selected column uses index values for the features selected, the corresponding features for the indexes are shown in Table 6.2. The Feature Subset ID is used in order to refer to each feature subset during the report.

To ensure that these results from the C4.5/LOO-CV (tested on random sub-samples) gave a good representation of the whole data-set (containing all instances), a second set of testing was conducted using the C4.5 Decision Tree classifier with resubstitution, this test was performed on the whole data-set rather than the sub-samples. Table 3.5 shows these results.

After a satisfactory amount of feature selection was conducted (Table 3.4 and 3.5 show the results), a general pattern showed when removing a certain set of features, the accuracy and F-Measure values tend to stay the same (negligible difference in accuracy and weighted average F-Measure), these features were deemed to be redundant. Once again, feature subset ID 3 achieved the highest accuracy of (49.40%), and the second highest weighted average F-Measure (0.522), whilst the highest weighted F-Measure was achieved by feature subset ID 9 (0.523). Feature subset 3, 9, and 1 have had very similar results in terms of accuracy and F-Measure values (for all classes). The differences in selected features is that the lower the amount of features in each subset, the more redundant features get removed. Furthermore, these subsets of features are able to retain a better F-Measure values for the under-represented classes (1-3), this is in comparison to all the other subsets of features.

Whilst there is a lack of predictive power in these results, certain features tend to be present in the top subsets found, these often are the total counts of each section type (labs, lectures, discussions, field work, seminars, independent study). Whilst these total counts fail to provide better accuracy and F-Measure values (in comparison to the other feature subsets), they tend to increase the accuracy and F-Measure values when combined with other features. It is possible to see that the top feature subsets will have these total counts (Feature Index Values: 49, 50, 51, 52, 53, 54). When looking into more detail, it is possible to see that the total independent study and lecture counts are always present in each subset. Table 3.6 shows comparisons for accuracy and F-Measures on a subset of features (feature subset ID = 3) when the total counts are removed, these tests are performed on a random sub-sample, and the whole data-set (using the C4.5 Decision Tree Classifier). The results show that there is an overall increase in accuracy and F-Measure, whilst feature subset ID 3.1 in both tests result in certain class F-Measures being mathematically undefined, these are results of division by zero when calculating the precision (both true and false positives are equal to zero).

This means that the results do not present a weighted average F-Measure, however, it is possible to look at the overall accuracy and individual class F-Measures to compare the performance of each subset of features. Looking at Table 3.6, it shows that the accuracy (when looking at random sub-samples /LOO-CV) decreases by 4.10% when removing the total section counts, and the F-Measures of individual classes (of the ones that have results) have gone down in each class. When looking at the results from using the whole data-set with resubstitution, the accuracy decreases by 9.77%, and the F-Measure values for each class decreases.

Feature Subset ID	Features (Index Values)	Number of Features Retained (n)	Accuracy (%)	*Per-Class F-Measure*											*Weighted Average*
				10	9	8	7	6	5	4	3	2	1	0	
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	39	55.8608	0.726	0.480	0.332	0.347	0.377	0.350	0.331	0.278	0.291	0.275	0.244	0.522
9	2-8, 10-16, 18-24, 26-32, 34-40, 42-54	48	55.8591	0.728	0.483	0.325	0.351	0.378	0.348	0.318	0.288	0.275	0.281	0.238	0.523
1	1-6, 10-15, 17-22, 29-31, 33-38, 49-54	33	55.7767	0.725	0.479	0.333	0.348	0.374	0.348	0.326	0.277	0.291	0.253	0.240	0.522
5	1, 3-4, 6, 8, 10-15, 17-21, 26-28, 30, 49, 51, 53-54	24	54.767	0.720	0.462	0.312	0.332	0.342	0.329	0.296	0.160	0.135	0.172	0.236	0.507
2	2-5, 10-15, 18-19, 21, 29, 49-51, 53-54	19	54.3703	0.721	0.462	0.301	0.320	0.321	0.313	0.272	0.159	0.196	0.182	0.223	0.503
4	1, 3-4, 12-14, 17-22, 49, 53-54	15	53.7195	0.704	0.442	0.301	0.333	0.357	0.341	0.269	0.120	0.061	?	0.175	?
6	4, 12-14, 18, 25, 28, 30, 49-51, 53-54	13	50.9548	0.700	0.419	0.219	0.224	0.232	0.262	0.256	0.138	0.136	0.192	0.202	0.456
8	1, 9, 17, 25, 33, 41, 49-54	12	49.1002	0.688	0.402	0.197	0.176	0.156	0.175	0.147	0.080	0.105	0.138	0.147	0.430
7	49-54	6	47.7539	0.681	0.392	0.102	0.060	0.132	0.150	0.135	0.055	0.116	0.113	0.136	0.397

Table 3.5: Experimental results when testing subsets of features found in Table 3.4, using a combination of C.45/Resubstitution on the whole data-set containing all instances. Feature 55 is not included in the features selected list as it is the predicted variable (Evaluation Class Metric). Features Selected column uses index values for the features selected, the corresponding features for the indexes are shown in Table 6.2. The Feature Subset ID is used in order to refer to each feature subset during the report.

Feature Subset ID	Features (Index Values)	Test Sample	Protocol	Accuracy (%)	*Per-Class F-Measure*											*Weighted Average*
					10	9	8	7	6	5	4	3	2	1	0	
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	Random Sub-sample	LOO-CV	49.40	0.712	0.393	0.227	0.181	0.238	0.078	0.289	0.000	0.000	0.000	0.475	0.457
3.1 (Without features 49-54)	1-6, 9-15, 17-22, 25-31, 33-38, 41	Random Sub-sample	LOO-CV	45.30	0.657	0.244	0.130	0.110	0.079	0.048	?	0.000	?	?	0.033	?
3	1-6, 9-15, 17-22, 25-31, 33-38, 41, 49-54	Whole Data-set	Resub.	55.86	0.726	0.480	0.332	0.347	0.377	0.350	0.331	0.278	0.291	0.275	0.244	0.522
3.1 (Without features 49-F54)	1-6, 9-15, 17-22, 25-31, 33-38, 41	Whole Data-set	Resub.	46.09	0.645	0.282	0.199	0.165	0.169	0.110	0.033	?	?	?	0.039	?

Table 3.6: Experimental results showing the impact of total section counts (features 49-54) on the accuracy and F-Measures using C4.5 Decision Tree classifier and the different sample and protocol types (LOO-CV is used for random sub-sample, and resubstitution is used for the whole data-set. This is due to hardware constraints). Feature 55 is not included in the features selected list as it is the predicted variable (Evaluation Class Metric). Features Selected column uses index values for the features selected, the corresponding features for the indexes are shown in Table 6.2. The Feature Subset ID is used in order to refer to each feature subset during the report. (Resub. = Resubstitution, ? = Mathematically Undefined).

When looking at all the experiments conducted in this section, it is possible to interpret that there is a set of redundant features which do not need to be included when testing classifiers. Furthermore, most subsets of features lack the the ability to provide indisputable evidence that certain aspects of course structure have an influence on student performance. It is possible to see that certain features are consistently present each subset of features, such as the total section counts. In order to ensure that the insufficient predictive power is because of overfitting and/or poor choice of classifier, a next series of tests (demonstrated in Section 3.4) will test the 3 different subsets of feature using different classifiers (using the all instances from the whole data-set), these 3 subsets will be selected based on the highest F-Measures (weighted average values) achieved in Table 3.5. A list of these 3 subsets of features are shown in Table 3.7.

Feature Subset ID	Features (Index Values)	Number of Features Retained (n)
3	1, 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 15, 17, 18, 19, 20, 21, 22, 25, 26, 27, 28, 29, 30, 31, 33, 34, 35, 36, 37, 38, 41, 49, 50, 51, 52, 53, 54	39
9	2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20, 21, 22, 23, 24, 26, 27, 28, 29, 30, 31, 32, 34, 35, 36, 37, 38, 39, 40, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54	48
1	1, 2, 3, 4, 5, 6, 10, 11, 12, 13, 14, 15, 17, 18, 19, 20, 21, 22, 29, 30, 31, 33, 34, 35, 36, 37, 38, 49, 50, 51, 52, 53, 54	33

Table 3.7: These are the list of features used during the classification experiments. The Feature Subset ID is an identifier that is used during the report, the features are index values that correspond to Table 6.2.

3.4 Classifier Experimentation

When receiving the results from the previous experiment, it is evident that the accuracy, and the F-Measure values for each class were insufficient. The subsets of features that were used had failed to show any strong evidence that there is a link between course structure and student performance. This meant that the next step was to evaluate the classifiers used, this is to ensure that a range of classifiers have been tested, in order to support the previous experiments, or provide new information that can be useful.

In an attempt to maximise the accuracy and F-Measure (both individual classes and overall), a set of experiments are run. The experiments used 3 subsets of features (shown in Table 3.7), using the whole data-set, and using the resubstitution protocol with the classifier. The results from these experiments aimed to have minimal overfitting. Table 3.8 contains all the results of these experiments.

Feature Subset ID	Classifier	Accuracy (%)	*F-Measure Per Class* 10	9	8	7	6	5	4	3	2	1	0	*Weighted Average*
3	Random Tree	65.491	0.757	0.589	0.519	0.583	0.622	0.634	0.617	0.503	0.498	0.481	0.469	0.643
3	1-Nearest Neighbour	65.491	0.757	0.589	0.519	0.583	0.622	0.634	0.617	0.503	0.498	0.481	0.469	0.643
3	Random Forest	65.488	0.759	0.593	0.512	0.569	0.606	0.619	0.595	0.468	0.473	0.418	0.448	0.639
9	Random Tree	65.422	0.756	0.588	0.518	0.582	0.621	0.633	0.615	0.503	0.498	0.481	0.468	0.642
9	1-Nearest Neighbour	65.422	0.756	0.588	0.518	0.582	0.621	0.633	0.615	0.503	0.498	0.481	0.468	0.642
9	Random Forest	65.419	0.758	0.592	0.511	0.569	0.605	0.617	0.592	0.471	0.473	0.422	0.448	0.639
1	Random Tree	65.403	0.756	0.588	0.519	0.583	0.622	0.634	0.616	0.503	0.493	0.475	0.460	0.642
1	1-Nearest Neighbour	65.403	0.756	0.588	0.519	0.583	0.622	0.634	0.616	0.503	0.493	0.475	0.460	0.642
1	Random Forest	65.400	0.758	0.592	0.511	0.569	0.606	0.618	0.594	0.462	0.467	0.400	0.439	0.638
3	C4.5 Decision Tree (Unpruned)	57.894	0.736	0.509	0.380	0.394	0.421	0.399	0.368	0.343	0.315	0.333	0.374	0.553
9	C4.5 Decision Tree (Unpruned)	57.875	0.736	0.507	0.378	0.391	0.420	0.404	0.372	0.337	0.324	0.349	0.372	0.553
1	C4.5 Decision Tree (Unpruned)	57.803	0.735	0.507	0.382	0.395	0.420	0.398	0.370	0.343	0.316	0.325	0.368	0.553
3	C4.5 Decision Tree (Pruned)	55.861	0.726	0.480	0.332	0.347	0.377	0.350	0.331	0.278	0.291	0.275	0.244	0.522
9	C4.5 Decision Tree (Pruned)	55.859	0.728	0.483	0.325	0.351	0.378	0.348	0.318	0.288	0.275	0.281	0.238	0.523
1	C4.5 Decision Tree (Pruned)	55.777	0.725	0.479	0.333	0.348	0.374	0.348	0.326	0.277	0.291	0.253	0.240	0.522
9	REP Tree	53.239	0.720	0.458	0.265	0.274	0.303	0.266	0.215	0.201	0.199	0.206	0.250	0.490
3	REP Tree	53.043	0.720	0.453	0.283	0.277	0.283	0.249	0.234	0.177	0.191	0.206	0.241	0.490
1	REP Tree	53.007	0.719	0.453	0.284	0.277	0.281	0.249	0.221	0.189	0.175	0.230	0.254	0.490
9	Decision Table	50.0893	0.713	0.428	0.250	0.177	0.192	0.175	0.154	0.046	0.025	?	0.141	?
1	Decision Table	50.0824	0.708	0.421	0.231	0.189	0.160	0.180	0.131	0.028	0.013	?	0.150	?
3	Decision Table	49.1946	0.699	0.411	0.217	0.151	0.122	0.148	0.125	?	0.013	?	0.120	?
9	Multilayer Perceptron	48.745	0.704	0.424	0.136	0.110	0.247	0.097	0.144	0.121	0.170	?	0.035	?
1	Multilayer Perceptron	48.231	0.695	0.407	0.129	0.091	0.225	0.085	0.159	0.111	0.154	?	0.011	?
3	Multilayer Perceptron	47.498	0.689	0.426	0.186	0.099	0.228	0.085	0.098	?	0.174	?	0.076	?
1	Naive Bayes	33.140	0.572	0.000	0.308	0.012	0.212	0.094	0.077	0.085	0.000	0.073	0.300	0.305
9	Naive Bayes	33.125	0.565	0.000	0.313	0.012	0.208	0.061	0.078	0.000	0.000	0.081	0.306	0.302
3	Naive Bayes	27.214	0.411	0.000	0.289	0.011	0.221	0.052	0.078	0.000	0.000	0.082	0.479	0.244

Table 3.8: Results from testing the top 3 subsets of features (from Table 3.5) using different classifiers with resubstitution, using the whole data-set containing all instances. (Feature Subset ID is linked in Table 3.5, ? = Mathematically Undefined).

When looking at the results, there is a clear spread in classifier performance. The Random Tree, 1-Nearest Neighbour, and Random Forest performed better than all the other classifiers, with similar results for the accuracy and F-Measure values (overall and individual class values). This was not a surprise as it was expected that these classifiers would perform well on this type of data-set. The Random Tree and 1-Nearest Neighbour classifier ended up outperforming the other classifiers, with the identical accuracy and F-Measure (both overall and individual classes) values (within each subset of features). Feature subset 3 (ID) achieved the highest accuracy (65.491%) and overall F-Measure (0.643) when using either the Random Tree or 1-Nearest Neighbour classifier. When running a C4.5 Decision Tree classifier, a different set of parameters were used, the main change in parameters was whether pruning was used or not. The results show that when pruning was set to true, the accuracy decreased by roughly 2%, and the overall F-Measure decrease by roughly 0.1 (this is when comparing the same feature subsets to it's own results).

As the initial feature selection results used the C4.5 pruning Decision Tree classifier, it was decided to use this classifier again, whilst also changing a key parameter when setting up the classifier. This parameter was deciding whether the decision tree will be pruned or not. When looking at the results, using pruning decreased the performance of the classifier (for all subsets of features). The highest performing subset of features (ID = 3) when using both pruned and unpruned C4.5 Decision Tree classifier resulted in an accuracy of 57.894% (unpruned), and an accuracy of 55.861% (pruned). The F-Measure also decreased, going from 0.553 (unpruned) to 0.522 (pruned). Whilst there is a decrease in both accuracy and F-Measure, it is important to note that the decrease is very small. Reduced Error Pruning (REP) Tree is based on the C4.5 Decision Tree classifier, it is a fast decision tree classifier, this classifier builds a decision tree based on reducing variance and the information gain [49]. This classifier did not perform as well as the C4.5 Decision Tree classifier, with the highest accuracy (from feature subset 9 [ID]) of 53.239% and an F-Measure value of 0.490.

When looking at the results from the Multilayer Perceptron (MLP) experiments, there is a question mark symbol in class 1 for all three MLP tests, and one question mark for class 3, feature subset 3 (ID). Weka returns this symbol when the value for that calculation is mathematically undefined. This tends to occur when there is a division by

True \Predicted	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
0	**87**	0	19	0	0	0	2	0	1	132	1896
1	0	**0**	35	0	0	0	0	0	1	13	76
2	0	0	**40**	0	0	0	3	0	3	22	84
3	0	0	42	**0**	0	1	17	0	10	48	87
4	0	0	43	0	**28**	14	65	13	61	207	96
5	0	0	44	0	4	**72**	295	57	168	650	262
6	2	0	32	0	5	37	**616**	127	477	1404	560
7	7	0	21	0	5	8	506	**293**	806	2456	1066
8	6	0	12	0	0	6	360	170	**1253**	4457	2964
9	30	0	7	0	1	6	245	52	1022	**7215**	4001
10	30	0	12	0	0	2	30	17	428	4726	**18056**

Table 3.9: Confusion Matrix of Multilayer Perceptron (with resubstitution) test using feature subset 3 (ID). To understand how Confusion Matrices work, refer back to Table 2.1.

zero, for example, looking at Table 3.9, there are zero instances classified for class 1 and 3. The equation for calculating precision (Eq. (2.5)) shows that there needs to be at least one instance classified in that class (whether it be a true or false positive) in order to calculate a value, otherwise there will be a division by zero which means that it will not be possible to calculate an F-Measure for that class.

The Naive Bayes classifier resulted in an accuracy of 33.140% and an overall F-Measure value of 0.305 (this was the highest achieved result for Naive Bayes, using feature subset 1). These results were poor, but the Naive Bayes classifier was expected to perform poorly for this data-set, due to fact that it makes assumptions that have a chance of being either correct, or incorrect (this is why it is referred to as naive). Naive Bayes makes the assumption that the predictive features in the data are independent (when given a class) [50], in simpler terms, the presence of a feature is not related to the presence of another feature (for a given class). This means that the classifier assumes that each feature in the subset contributes independently when classifying the object. This would not work well with this data-set, this is because during the feature selection phase, none of the features could provide a strong classification result when tested against the class (predicted) variable, meaning that the little predictive power that has been achieved by the best classifier and subset of features combination is due to a large combination of features (rather than the individual features).

To summarise, the results show that there is very little difference in performance between the top three subsets of features. When attempting to find the best classifier (in attempt to maximise F-Measure and accuracy), it was found that the Random Tree, Random Forest, and 1-Nearest Neighbour performed the best on this data-set. Whilst it was possible to improve the performance of the model from the initial stages of feature selection, none of the subsets of features were able to provide results which show strong predictive power when attempting to predict the class (predicted) variable.

Chapter 4

Results and Evaluation

4.1 Results

In order to find out whether there is a link between course structure and student performance, a set process had been put in place. This process included scrubbing, cleaning, and exploring the data. Exploring the data involved finding early relationships between features in the data-set. Conducting feature selection was a challenging task, as finding an optimal subset of features was not supported by the results shown in Section 3.3 and Section 3.4. The experiments consisted of testing the predictive power of individual and subsets of features. As the initial feature selection results did not provide strong subsets of features for the classification task, the next option was to see if any individual or subsets of features improved the performance of the model when testing (rather finding an optimal subset of features). Doing this investigation, it was noticed that most of the subsets of features which had the highest accuracy and overall F-Measure consisted of the total section counts (features 49-54 in Table 6.2). These features totalled the number of classes that took place for each section type (in each course). In one scenario, the accuracy of the model improved by 9.77% (the overall F-Measure for the subsets that do not use the total section counts were left as mathematically undefined as classes 1, 2, and 3 were unable to calculate precision due to a division by zero). Figure 4.1 shows how the accuracy increases when the total section counts are removed.

Since there is some improvements in accuracy when testing the total section counts, a next experiment was conducted to check whether the total section counts are able to predict the class (predicted) variable during the classification task. What was found is that it could achieve an accuracy of 47.75%, and an overall F-Measure of 0.397. This

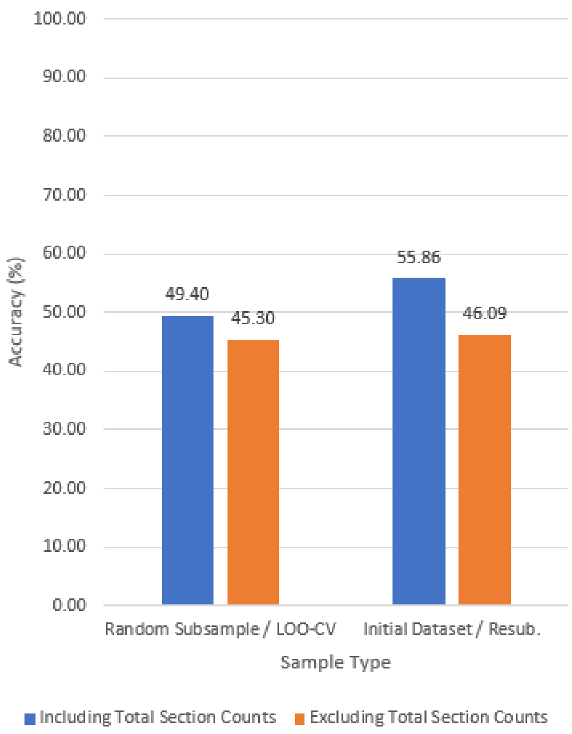


Figure 4.1: This graph shows the difference in accuracy when the total section counts are included and excluded from the subset. This data is from Table 3.6

ended up being the worst performing subset of features during the final feature selection task (the results for this are shown in Table 3.5 [Feature Subset ID = 7]). So what can be interpreted from this is that the total sections have an influence on the predictive power when combined with other subsets of features, meaning that this may be a link between course structure and student performance.

Upon completing the final feature selection experiments, it was found that most subsets of features were unable to predict the evaluation class metric with sufficient accuracy or F-Measure (results shown in Table 3.5). Figure 4.2 shows the top subsets of features that came out of the final feature selection task. The top three subsets of features were very close when it came to predictive power, whilst the accuracy and overall F-Measure were still low, it was decided to test these subsets of features using different classifiers. This was done to find the best classifier for the data-set, testing whether the classifiers used were overfitting the data, and to find out whether these subsets could provide sufficient results to provide an insight to whether there is a link between course structure and student performance.

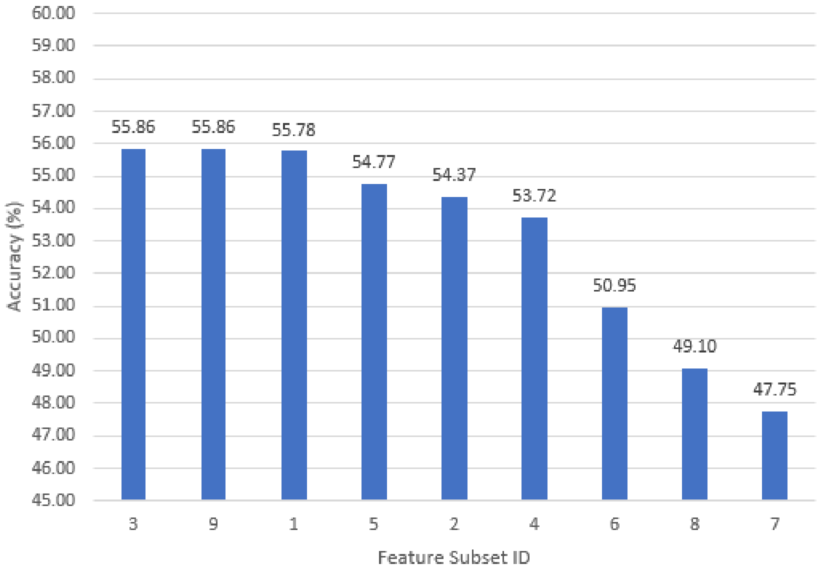
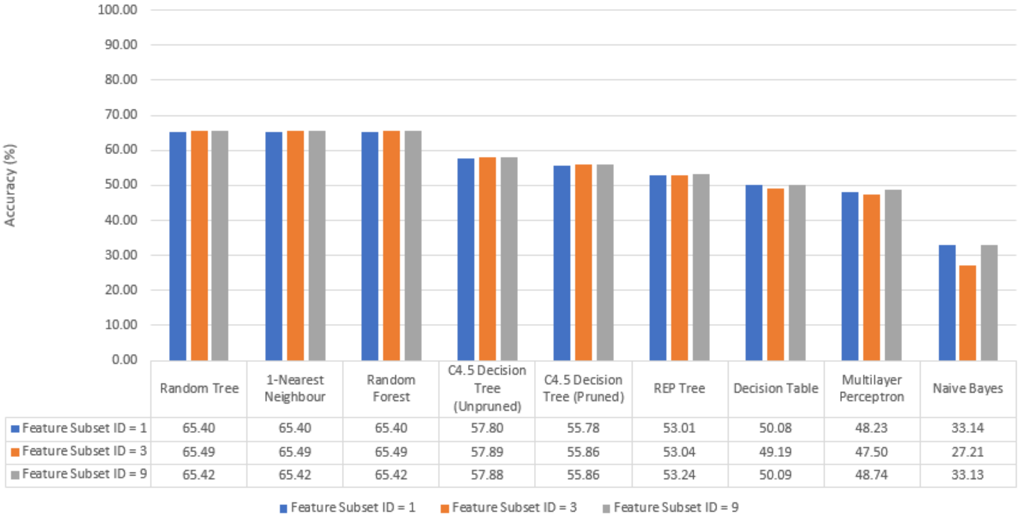


Figure 4.2: This graph shows the accuracy achieved by each subset of results (using the whole data-set) when using a C4.5 Decision Tree Classifier with Resubstitution. The choice of axis (accuracy) scaling is to show the slight differences in accuracy, it would be more difficult to see the difference if the axis were scaled from 0-100%. This data is from Table 3.5.



	Random Tree	1-Nearest Neighbour	Random Forest	C4.5 Decision Tree (Unpruned)	C4.5 Decision Tree (Pruned)	REP Tree	Decision Table	Multilayer Perceptron	Naive Bayes
■ Feature Subset ID = 1	65.40	65.40	65.40	57.80	55.78	53.01	50.08	48.23	33.14
■ Feature Subset ID = 3	65.49	65.49	65.49	57.89	55.86	53.04	49.19	47.50	27.21
■ Feature Subset ID = 9	65.42	65.42	65.42	57.88	55.86	53.24	50.09	48.74	33.13

Figure 4.3: This graph shows the accuracy achieved by the top three subsets of results (results shown in Table 3.5) when using the whole data-set (containing all instances), and when testing them using different classifiers. A table of results below the bars show the in-depth accuracy achieved by each combination of feature subsets and classifiers. This data is from Table 3.8.

The results shown in Figure 4.3 describe how each classifier performed when testing on three three different subsets of features. Looking at the results, Random Tree, Random Forest, and the 1-Nearest Neighbour classifier produced the highest accuracy, but when looking at these results, it is possible to see that the results do not provide sufficient predictive power (highest accuracy was achieved by feature subset 3 [ID], 65.49%). The interpretation to take from these results is that even by finding the best possible subset of features (in terms of accuracy and F-Measure results), it does not provide sufficient evidence to prove that there is a link between course structure and student performance (for this data-set).

After follow the data science process of obtaining and scrubbing the data, extensive testing on the data-set using feature selection techniques, and rigorously testing different classifiers to optimise the performance of models, the main question of this research needs to be answered, *is there a link between course structure and student performance?*

When analysing the data, initial relationships that would need to be investigated were tested, but they resulted in poor results which did not support the main question. Further investigation lead to believe that there may be a possibility that total section counts could have an influence on the evaluation class metric (this metric was used as a representation of student performance). Whilst there was not sufficient evidence that these features had a strong relationship with student performance, it was observed that they did improve the predictive power of the model. Furthermore, these features were found in most subsets which had the best possible performance in the classification tasks.

Therefore, it can be said that the total section counts (total lab count, total lecture count, total discussion count, total seminar count, and total independent study count) have an link to student performance, however, this is a weak link as it only made a slight impact on the accuracy and overall F-Measure of the classification tasks, and only improved other subsets (it was unable to provide sufficient results when these metrics were tested by themselves).

When looking at the best subsets of features formed, the results were insufficient. It was possible to increase the performance of these models by changing classifiers and fine tuning their parameters, however, these subsets were unable to provide strong predictive power during the classification task. Therefore, the final answer to this question is that there was not an optimal course structure that could provide sufficient results to prove whether there is a link between course structure and student performance.

4.2 Evaluation

Evaluating this project very important as there needs to be some analysis on the impact of this work. Learning analytics aims to optimise learning environments by examining and analysing data. This is important as it can support students, helping them maximise their potential, and have a positive influence on their progression. Using educational data will always have ethical concerns. Data must not be misused, and any data with personal information should be consensual, and stored securely. If data is misused, then legal issues may arise, all data must comply with the Data Protection Act. The data used in this project is made public by the University of Wisconsin-Madison, and does not consist of personal information of any students.

As this type of research has a large impact on learners and educators, it is essential that the outcomes of this work is accurate, otherwise it may provide misleading information that may have an impact on both educators and students. Educators provided with inaccurate findings may implement changes to their courses, in hopes that a new research finding will provide improvements for the environments in which their students learn in, however, this inaccurate information may have adverse impacts on student outcomes. This makes the critical analysis of any relationships found in the data very important; ensuring that all interpretations are presented with precaution, and with the sufficient evidence to support the findings from the research. In addition to this, interpretations will have an element of bias from the authors that analyse the data, this means that great care needs to be taken when interpreting results. The author of this piece of work has attempted to analyse the data extensively, ensuring that any interpretations made to prove or disprove the main question of this work is supported with sufficient evidence.

Chapter 5

Conclusion

The aim of this project was to examine the link between course structure and student performance, investigating whether there is a link, and if so, what are these links. This involved obtaining a data-set suitable for the question, preparing this data-set, conducting feature selection, and model creation. Feature selection aimed to improve the performance of the models, what was found is that the original data-set (prior to feature selection) had the highest accuracy (65.51%) and overall F-Measure (0.640). When applying the feature selection, the accuracy for the subsets of features would either: negligibly decrease or significantly decrease in accuracy and overall F-Measure. This allowed us to answer the question that there is not an optimal course structure which has an influence on student performance. One link that was found was the total section counts, these are 6 features which had the ability to improve a classifiers performance when testing a subset of features, furthermore, they were present in most of the best performing subsets of features. However, they did not hold any predictive power on their own. Whilst these features were able to improve the classifiers performance, this improvement was not substantial (only 9.77% in accuracy). What could be interpreted from these results is that there is a link between total section counts and student performance, but this link is relatively weak.

This data-set struggled to produce any subset of features which could be used predict the evaluation class metric (with sufficient predictive power). The next step would be to possibly, change the predicted variable. This could be moving onto predicted individual grade counts, or, balancing the classes in attempt to increase F-Measure and accuracy. The next steps should be moving onto conducting further tests, whilst the current set of features were exhaustively tested in attempt to predict the evaluation class metric, it may be possible that this metric was not a reasonable choice for this data-set.

Bibliography

[1] G. Siemens and R. S. d. Baker, “Learning analytics and educational data mining: towards communication and collaboration,” in *Proceedings of the 2nd international conference on learning analytics and knowledge*, pp. 252–254, 2012.

[2] K. Adam, N. A. A. Bakar, M. A. I. Fakhreldin, and M. A. Majid, “Big data and learning analytics: A big potential to improve e-learning,” *Advanced Science Letters*, vol. 24, no. 10, pp. 7838–7843, 2018.

[3] H. Lambert, “How the british degree lost its value,” 2019.

[4] M. Molesworth, R. Scullion, and E. Nixon, *The marketisation of higher education*. Routledge, 2010.

[5] E. R. Peterson, C. M. Rubie-Davies, M. J. Elley-Brown, D. A. Widdowson, R. S. Dixon, and S. E. Irving, “Who is to blame? students, teachers and parents views on who is responsible for student achievement,” *Research in Education*, vol. 86, no. 1, pp. 1–12, 2011.

[6] N. Sclater, A. Peasgood, and J. Mullan, “Learning analytics in higher education,” *London: Jisc. Accessed February*, vol. 8, no. 2017, p. 176, 2016.

[7] R. Ferguson, “Learning analytics: drivers, developments and challenges,” *International Journal of Technology Enhanced Learning*, vol. 4, no. 5/6, pp. 304–317, 2012.

[8] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, “Data mining with big data,” *IEEE transactions on knowledge and data engineering*, vol. 26, no. 1, pp. 97–107, 2014.

[9] B. Dietz-Uhler and J. E. Hurn, “Using learning analytics to predict (and improve)

student success: A faculty perspective," *Journal of interactive online learning*, vol. 12, no. 1, pp. 17–26, 2013.

[10] J. B. Arbaugh, "Is there an optimal design for on-line mba courses?," *Academy of Management Learning & Education*, vol. 4, no. 2, pp. 135–149, 2005.

[11] J. Paul and F. Jefferson, "A comparative analysis of student performance in an online vs. face-to-face environmental science course from 2009 to 2016," *Frontiers in Computer Science*, vol. 1, p. 7, 2019.

[12] S. S. Jaggars and D. Xu, "How do online course design features influence student performance?," *Computers Education*, vol. 95, pp. 270 – 284, 2016.

[13] V. Dhar, "Data science and prediction," *Communications of the ACM*, vol. 56, no. 12, pp. 64–73, 2013.

[14] H. Mason and C. Wiggins, "A taxonomy of data science," 2010-09-25.

[15] S. Marsland, *Machine learning: an algorithmic perspective*. CRC press, 2015.

[16] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," 1988.

[17] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.

[18] B. J. Copeland, *The essential turing*. Clarendon Press, 2004.

[19] "On computable numbers, with an application to the entscheidungsproblem," *J. of Math*, vol. 58, no. 345-363, p. 5, 1936.

[20] L. De Mol, "Turing machines," in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), Metaphysics Research Lab, Stanford University, winter 2019 ed., 2019.

[21] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[22] M. A. Chatti, A. L. Dyckhoff, U. Schroeder, and H. Thüs, “A reference model for learning analytics,” *International Journal of Technology Enhanced Learning*, vol. 4, no. 5-6, pp. 318–331, 2013.

[23] S. Kotsiantis, C. Pierrakeas, and P. Pintelas, “Predicting students’performance in distance learning using machine learning techniques,” *Applied Artificial Intelligence*, vol. 18, no. 5, pp. 411–426, 2004.

[24] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[25] Z. Ghahramani, “Unsupervised learning,” in *Summer School on Machine Learning*, pp. 72–112, Springer, 2003.

[26] P. Dayan, M. Sahani, and G. Deback, “Unsupervised learning,” *The MIT encyclopedia of the cognitive sciences*, pp. 857–859, 1999.

[27] L. Van Der Maaten, E. Postma, and J. Van den Herik, “Dimensionality reduction: a comparative,” *J Mach Learn Res*, vol. 10, no. 66-71, p. 13, 2009.

[28] A. Hinneburg and D. A. Keim, “Optimal grid-clustering: Towards breaking the curse of dimensionality in high-dimensional clustering,” 1999.

[29] D. Koller and M. Sahami, “Toward optimal feature selection,” tech. rep., Stanford InfoLab, 1996.

[30] M. Dash and H. Liu, “Feature selection for classification,” *Intelligent data analysis*, vol. 1, no. 3, pp. 131–156, 1997.

[31] P. Narendra and K. Fukunaga, “A branch and bound algorithm for feature subset selection,” *IEEE Transactions on Computers*, vol. 26, pp. 917–922, sep 1977.

[32] D. Koller and M. Sahami, “Toward optimal feature selection,” in *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, ICML’96, (San Francisco, CA, USA), p. 284–292, Morgan Kaufmann Publishers Inc., 1996.

[33] N. Zhong, J. Dong, and S. Ohsuga, “Using rough sets with heuristics for feature selection,” *Journal of intelligent information systems*, vol. 16, no. 3, pp. 199–214, 2001.

[34] M. Ciortan and towards data science, “Overview of feature selection methods,” 2019-07-26.

[35] L. Talavera, “An evaluation of filter and wrapper methods for feature selection in categorical clustering,” in *International Symposium on Intelligent Data Analysis*, pp. 440–451, Springer, 2005.

[36] A. G. Karegowda, M. Jayaram, and A. Manjunath, “Feature subset selection problem using wrapper approach in supervised learning,” *International journal of Computer applications*, vol. 1, no. 7, pp. 13–17, 2010.

[37] R. E. Korf, “Linear-space best-first search,” *Artificial intelligence.*, vol. 62, no. 1, pp. 41–78, 1993.

[38] A. G. Barnston, “Correspondence among the correlation, rmse, and heidke forecast verification measures; refinement of the heidke score,” *Weather and Forecasting*, vol. 7, no. 4, pp. 699–709, 1992.

[39] P. E. Dennison and D. A. Roberts, “Endmember selection for multiple endmember spectral mixture analysis using endmember average rmse,” *Remote sensing of environment*, vol. 87, no. 2-3, pp. 123–135, 2003.

[40] T. Chai and R. R. Draxler, “Root mean square error (rmse) or mean absolute error (mae)?,” *GMDD*, vol. 7, no. 1, pp. 1525–1534, 2014.

[41] S. Visa, B. Ramsay, A. L. Ralescu, and E. Van Der Knaap, “Confusion matrix-based feature selection.,” *MAICS*, vol. 710, pp. 120–127, 2011.

[42] D. M. Powers, “Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation,” 2011.

[43] C. Ferri, J. Hernández-Orallo, and R. Modroiu, “An experimental comparison of performance measures for classification,” *Pattern Recognition Letters*, vol. 30, no. 1, pp. 27–38, 2009.

[44] “Appendix b - the weka workbench,” in *Data Mining (Fourth Edition)* (I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, eds.), pp. 553 – 571, Morgan Kaufmann, fourth edition ed., 2017.

[45] C. Gray, *Learning Analytics Integrating Student Attendance Data*. PhD thesis, 11 2019.

[46] L. Breiman, “Machine learning, volume 45, number 1 - springerlink,” *Machine Learning*, vol. 45, pp. 5–32, 10 2001.

[47] L. Kuncheva, *Pattern Recognition and Neural Networks*. LULU COM, 2019.

[48] S. L. Salzberg, “C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993,” 1994.

[49] W. H. Mohamed, M. Salleh, and A. H. Omar, “A comparative study of reduced error pruning method in decision tree algorithms,” *2012 IEEE International Conference on Control System, Computing and Engineering*, pp. 392–397, 2012.

[50] G. H. John and P. Langley, “Estimating continuous distributions in bayesian classifiers,” *arXiv preprint arXiv:1302.4964*, 2013.

Chapter 6

Appendix

6.1 Distribution of Instances in Each Sample

Table 6.1: Results comparing each random sub-sample with the whole data-set (consisting of all the instances), calculating the percentage of all instances in each class, the maximum difference (in percentage) between the random sub-sample and the whole data-set, and the standard deviation of all the samples. (N = Number of Instances, % = Percentage of instances in each class for each sample, Max % Diff (Δ%) = The maximum difference in percentage between random sub-samples and the whole data-set, $\Sigma(N)$ = Sum of all instances from each class).

Sample Type	*Class*																						*Total Instances* (ΣN)
	10		9		8		7		6		5		4		3		2		1		0		
	N	%	N	%	N	%	N	%	N	%	N	%	N	%	N	%	N	%	N	%	N	%	
Random Sub-sample 1	2098	41.96	1061	21.22	757	15.14	406	8.12	262	5.24	139	2.78	58	1.16	20	0.40	19	0.38	7	0.14	173	3.46	5000
Random Sub-sample 2	1999	39.98	1132	22.64	771	15.42	455	9.10	256	5.12	136	2.72	39	0.78	18	0.36	14	0.28	8	0.16	172	3.44	5000
Random Sub-sample 3	1969	39.38	1111	22.22	831	16.62	418	8.36	273	5.46	136	2.72	47	0.94	19	0.38	16	0.32	9	0.18	171	3.42	5000
Random Sub-sample 4	1992	39.84	1117	22.34	770	15.40	474	9.48	271	5.42	113	2.26	44	0.88	15	0.30	12	0.24	9	0.18	183	3.66	5000
Full Data-set	23301	40.01	12579	21.60	9228	15.85	5168	8.87	3260	5.60	1552	2.67	527	0.90	205	0.35	152	0.26	125	0.21	2137	3.67	58234
Max % Diff (Δ%)	-1.95		-1.04		-0.77		0.75		0.48		0.41		-0.26		0.05		-0.12		0.07		0.25		
Standard Deviation	0.997		0.579		0.581		0.551		0.188		0.210		0.140		0.038		0.055		0.028		0.124		

6.2 List of Metrics/Features

Table 6.2: A list of all Metrics/Features used (Feature 55 is the predicted/class variable).

Feature Index Value	Feature Name	Data Type	Potential Values
1	Labs	Nominal	True or False
2	Labs on Monday	Nominal	True or False
3	Labs on Tuesday	Nominal	True or False
4	Labs on Wednesday	Nominal	True or False
5	Labs on Thursday	Nominal	True or False
6	Labs on Friday	Nominal	True or False
7	Labs on Saturday	Nominal	True or False
8	Labs on Sunday	Nominal	True or False
9	Lectures	Nominal	True or False
10	Lectures on Monday	Nominal	True or False
11	Lectures on Tuesday	Nominal	True or False
12	Lectures on Wednesday	Nominal	True or False
13	Lectures on Thursday	Nominal	True or False
14	Lectures on Friday	Nominal	True or False
15	Lectures on Saturday	Nominal	True or False
16	Lectures on Sunday	Nominal	True or False
17	Discussions	Nominal	True or False
18	Discussions on Monday	Nominal	True or False
19	Discussions on Tuesday	Nominal	True or False
20	Discussions on Wednesday	Nominal	True or False
21	Discussions on Thursday	Nominal	True or False
22	Discussions on Friday	Nominal	True or False
23	Discussions on Saturday	Nominal	True or False
24	Discussions on Sunday	Nominal	True or False
25	Field Work	Nominal	True or False
26	Field Work on Monday	Nominal	True or False
27	Field Work on Tuesday	Nominal	True or False
28	Field Work on Wednesday	Nominal	True or False
29	Field Work on Thursday	Nominal	True or False
30	Field Work on Friday	Nominal	True or False
31	Field Work on Saturday	Nominal	True or False

Table 6.2: A list of all Metrics/Features used (Feature 55 is the predicted/class variable).

Feature Index Value	Feature Name	Data Type	Potential Values
32	Field Work on Sunday	Nominal	True or False
33	Seminars	Nominal	True or False
34	Seminars on Monday	Nominal	True or False
35	Seminars on Tuesday	Nominal	True or False
36	Seminars on Wednesday	Nominal	True or False
37	Seminars on Thursday	Nominal	True or False
38	Seminars on Friday	Nominal	True or False
39	Seminars on Saturday	Nominal	True or False
40	Seminars on Sunday	Nominal	True or False
41	Independent Study	Nominal	True or False
42	Independent Study on Monday	Nominal	True or False
43	Independent Study on Tuesday	Nominal	True or False
44	Independent Study on Wednesday	Nominal	True or False
45	Independent Study on Thursday	Nominal	True or False
46	Independent Study on Friday	Nominal	True or False
47	Independent Study on Saturday	Nominal	True or False
48	Independent Study on Sunday	Nominal	True or False
49	Total Lab Count	Numeric	Positive Integer
50	Total Lecture Count	Numeric	Positive Integer
51	Total Discussions Count	Numeric	Positive Integer
52	Total Field Work Count	Numeric	Positive Integer
53	Total Seminar Count	Numeric	Positive Integer
54	Total Independent Study Count	Numeric	Positive Integer
55	*Evaluation Class Metric*	*Nominal*	*Multi-Class Values* *(0/1/2/3/4/5/6/7/8/9/10)*