Student Degree Recommendation System

Islam Ehab Anwar

Supervisor: Dr. Ahmed Elsheikh

Faculty of Computer Science

University of Prince Edward Island, Cairo Campus

May 2023

Acknowledgements

Words cannot express my appreciation and gratitude towards my professor, project adviser, and dean of the faculty of computer science Dr. Ahmed Elsheikh for his unwavering support and guidance. I would also like to thank Dr. Reham Hossam for her significant help and guidance in the technical specifics of this project. Additionally, this would not have been possible without my research assistants Heba Abdelkader and Ola Galal who without their guidance and experience I would not have been able to achieve this. Furthermore, I would like to thank the university for providing the necessary data for commencing this project.

I am also grateful to Dr. Amal Mohamed, professor at the faculty of computer science for her unconditional moral support and significant impact in inspiring me to pursue this field of study.

Lastly, I would be remiss in not mentioning my family, especially my parents, and brother. Their belief and encouragement have kept my spirits and motivation high during this trying process. I would never have reached this stage without them.

Abstract

Recommendation systems are at the forefront of many business applications, e-commerce, and digital software. Relevant research involves creating various types of recommendation systems including content-based filtering, collaborative filtering, and hybrid methods that appear to be the most effective recommendation system approaches. With the growing popularity of machine learning, the objective of this thesis is to create a viable student degree recommendation system that predicts and recommends student specializations or majors based on input data. The proposed pipeline includes data exploration, which analyzes the underlying nature of the data. Following exploration is the preprocessing stage, which investigates missing values, outlier detection, and feature selection techniques to clean the dataset. Hyperparameter optimization is executed to maximize model performances. The proposed model is a Random Forest model alongside other supervised classifiers including Decision Tree, Gaussian Naive Bayes, and Bernoulli Naive Bayes. Results indicate that Random Forest outperformed all other classifiers with consistent accuracies above 0.7, ahead of soft voting classifiers in a synthetically generated dataset, as well as the student dataset with specialization and major as the label variables in independent iterations. For interpretability, Random Forest provided the simplest explanations regarding feature interactions with the target variable including feature importance, Shapley values, and partial dependence plots. Biology, credit hours completed, and the school type were among the most numerically influential features when analyzing the impact on confidence of prediction.

Abbreviations

PCA - Principal Component Analysis

RS - Recommendation Systems

CF - Collaborative Filtering

CB - Content Based

DT - Decision Tree

RF - Random Forest

GNB - Gaussian Naive Bayes

BNB - Bernoulli Naive Bayes

AI - Artificial Intelligence

ML - Machine Learning

PDP - Partial Dependence Plot

RMSE - Root Mean Squared Error

AUC - Area Under the Curve

XAI - Explainable Artificial Intelligence

CGPA - Cumulative Grade Point Average

TP - True Positives

FP - False Positives

TN - True Negatives

FN - False Negatives

LIME - Local Interpretable Model-Agnostic Explanations

Table of Contents

Acknowledgements i

Abbreviations iii

List of Figures ix

List of Tables x

1 Introduction **1**

1.1 Motivation and Problem Statement 1

1.2 Thesis Outline 2

2 Background **3**

2.1 Basic Definitions & Terminologies 3

2.1.1 Data Exploration and Preprocessing 3

2.1.2 Feature Selection and Model Selection 5

2.1.3 Supervised Learning Techniques 6

2.1.4 Interpretation of Machine Learning Models 9

2.1.5 Recommendation Systems 12

2.2 Content-Based Filtering 17

2.3 Collaborative Filtering 17

2.4 Knowledge-Based Approaches 18

2.5 Hybrid Methods 19

3 Methodology **20**

3.1 System Overview 20

3.2 Student Data . . . 21
3.2.1 University Student Dataset . . . 21
3.3 Data Exploration . . . 23
3.3.1 Univariate Exploration . . . 23
3.3.2 Bivariate Exploration . . . 25
3.4 Data Preprocessing . . . 26
3.4.1 Missing Values . . . 27
3.4.2 Feature Encoding . . . 27
3.4.3 Outlier Detection & Analysis . . . 28
3.4.4 Data Augmentation . . . 28
3.5 Feature Selection . . . 29
3.6 Model Selection and Fusion . . . 30
3.7 Hyperparameter Optimization . . . 30
3.8 Local Model Interpretations . . . 31
3.8.1 Feature Importance . . . 31
3.8.2 Partial Dependence Plots (PDP) . . . 31
3.8.3 Shapley Values . . . 32

4 Results and Discussion **33**

4.1 Dataset . . . 33
4.2 Evaluation Metrics . . . 34
4.2.1 Criteria of Selection . . . 35
4.3 Results . . . 35
4.3.1 Synthetic Analysis . . . 35
4.3.2 Student Specialization Analysis . . . 43
4.3.3 Student Major Analysis . . . 47
4.4 Discussion . . . 49
4.4.1 Student Specialization . . . 50
4.4.2 Student Major . . . 56

5 Conclusion, Recommendations, and Future Work **65**

5.1 Conclusion 65

5.2 Future Work 66

List of Figures

2.1 A simple example of outliers in a 2-dimensional data set. *Source*: [Singh and Upadhyaya, 2012]. 5

2.2 Supervised Learning Process. *Source*: [Reader, 2021]. 6

2.3 Random Forest Procedure. *Source*: [Albert et al., 2022]. 9

2.4 Sample decision path for an observation in decision tree. *Source*: Done by the researcher. 10

2.5 The interaction strength for each of the input features with all other features for predicting the high risk of hypertension. *Source*: [Elshawi et al., 2019]. 11

2.6 Example Partial Dependence Plots. *Source*: [Erickson et al., 2021]. 12

2.7 Classification of Recommendation Approaches. *Source*: [Raghuwanshi and Pateriya, 2019]. 13

2.8 A high-level architecture of content-based recommendation system. *Source*: [Koene et al., 2015]. 14

2.9 Shows the methods that integrate CB characteristics into the CF approach. Mitigates the cold start problem in collaborative filtering. *Source*: [Thorat et al., 2015]. 16

2.10 Shows the methods that incorporate CF characteristics into a CB approach. *Source*: [Thorat et al., 2015]. 16

3.1 Architecture of System Overview. *Source*: Done by the researcher 21

3.2 Features Summary Statistics. *Source*: Done by the researcher 23

3.3 Histograms for the features. *Source*: Done by the researcher. 24

3.4 Pie plots of categorical features. *Source*: Done by the researcher. 25

3.5 Box plots between 2 features with specialization. *Source*: Done by the researcher. . 26

3.6 Checking for missing values. *Source*: Done by the researcher. 27

3.7 Sample of multivariate outliers using mahalanobis and χ^2. *Source*: Done by the researcher. 28
3.8 PCA plots. *Source*: Done by the researcher. 29

4.1 Histograms for synthetic features and bar chart for label. *Source*: Done by the researcher. 37
4.2 Scatter pair plot of all synthetic features. *Source*: Done by the researcher. 38
4.3 Feature importance of synthetic features. *Source*: Done by the researcher. 38
4.4 Correlation matrix of synthetic features. *Source*: Done by the researcher. 39
4.5 Explained and cumulative variance for synthetic principal components. *Source*: Done by the researcher. 40
4.6 Feature importance of optimized random forest model on synthetic data. *Source*: Done by the researcher. 41
4.7 Shapely values on synthetic training data. *Source*: Done by the researcher. 41
4.8 Partial dependence plots for features across different class values. *Source*: Done by the researcher. 42
4.9 Feature importance on specialization label set. *Source*: Done by the researcher. . . 44
4.10 Correlation matrix of specialization label set. *Source*: Done by the researcher. . . . 44
4.11 Explained and cumulative variance for features principal components. *Source*: Done by the researcher. 45
4.12 Bar plot for specialization. *Source*: Done by the researcher. 45
4.13 Feature importance for major label set. *Source*: Done by the researcher. 47
4.14 Correlation matrix for major label set. *Source*: Done by the researcher. 48
4.15 Explained and cumulative variance plots for major label set. *Source*: Done by the researcher. 49
4.16 Bar plot for the major variable. *Source*: Done by the researcher. 49
4.17 Structure of decision tree model. *Source*: Done by the researcher. 53
4.18 Feature importance of random forest after training and testing. *Source*: Done by the researcher. 54

4.19 Shapley values for entire training data. *Source*: Done by the researcher. 55
4.20 Shapley values for school type feature. *Source*: Done by the researcher. 55
4.21 Shapley values for cumulative GPA feature. *Source*: Done by the researcher. 56
4.22 Partial dependence plots of shapley values for biology feature. *Source*: Done by the researcher. 57
4.23 Partial dependence plots of shapley values for cumulative GPA feature. *Source*: Done by the researcher. 58
4.24 Feature importance of random forest after training and testing on major as label. *Source*: Done by the researcher. 61
4.25 Shapley values for entire training data on major as label. *Source*: Done by the researcher. 62
4.26 Shapley values for credit hours completed feature on major as label. *Source*: Done by the researcher. 62
4.27 Shapley values for A-Level math feature on major as label. *Source*: Done by the researcher. 63
4.28 Partial dependence plots of shapley values for credit hours completed on major as label. *Source*: Done by the researcher. 63
4.29 Partial dependence plots of shapley values for a level math on major as label. *Source*: Done by the researcher. 64

List of Tables

3.1 Data Set Features Summary 22

4.1 Hyperparameter optimization for various models on synthetic data set. 39
4.2 Comparison of classifiers performances on synthetic data. 40
4.3 Feature encoding of specialization variable using label encoder. 43
4.4 Hyperparameter optimization for various models on specialization label set. 46
4.5 Comparison of classifiers performances on specialization label set. 46
4.6 Feature encoding of the major variable using label encoder. 47
4.7 Hyperparameter optimization for various models on major label set. 50
4.8 Comparison of classifiers performances on major label set. 50
4.9 Outlier detection & analysis of specialization label set. 51
4.10 Outlier detection & analysis of the major label set. 60

Chapter 1

Introduction

1.1 Motivation and Problem Statement

In recent years, the accelerating advancement of internet technology has left us with an abundant amount of information available. Various sources of data are now publicly accessible to users such as news content, e-commerce websites, as well as digital entertainment [Wang et al., 2019] [Guo et al., 2020]. A significant challenge with having an overload of data available is ensuring the quality of the information provided to users from their corresponding searches as well as results that match their interests and preferences. Preferences can vary depending on the background such as educational or entertainment purposes.

As a result, various recent researchers such as R. Bodily et al. [Bodily and Verbert, 2017] and N. Thai-nghe et al. [Thai-Nghe et al., 2010] have attempted to create tailor-made solutions to provide users with a series of recommendations catering to specific user preferences. However, there is a lack of research performed in the educational field about the choice of degree or major in which a student enrolls at a university. This provides suitable motivation to pursue this field of study to understand what factors affect the degree chosen by a prospective student.

Recommendation systems are a type of information filtering system that predicts a user's preference for a product or service based on their past behavior, preferences, and interests [Singh et al., 2021]. These systems have become increasingly popular in recent years due to the growth of e-commerce

and online services [Singh et al., 2021]. They are used in a variety of applications such as movie recommendations, music recommendations, and product recommendations [Singh et al., 2021]. The goal of this thesis is to explore the different types of recommendation systems and their effectiveness in different domains [Singh et al., 2021].

R. Bodily et al. [Bodily and Verbert, 2017] and N. Thai-nghe et al. [Thai-Nghe et al., 2010] provide the most similar and relevant literature in using recommendation systems for student data from a procedural perspective. However, the gap lies in the aspect of interpretability. The research question at hand is how can single samples be analyzed to provide clear interpretations using an effective pipeline?

A student degree recommendation system is a problem that has not been tackled by many recent publishers. With the vast diversity of university programs available to prospective students, many are left wondering what their real passion is or if a particular degree is the one for them. A recommendation algorithm is suitable to mitigate this issue, by analyzing student academic performance during the latter end of high school as well as other factors such as familial status, personal status, and other daily lifestyle choices that could shape the overall performance of a student and impact the recommended degrees. Furthermore, a deep dive into the concept of interpretability of the proposed recommendation system and how that can build model trust and reliability by understanding why decisions were made at the local level of the chosen dataset.

1.2 Thesis Outline

The rest of the thesis is as follows: Chapter 2 describes background research regarding terminologies and definitions, a literature review regarding supervised as well as deep learning techniques used with recommendation systems. Chapter 3 discusses the methodology to develop an effective student degree recommendation system. Chapter 4 offers insight into the results and discussion. Chapter 5 provides the conclusion and future work.

Chapter 2

Background

This chapter provides the necessary knowledge and related research to this project. This project applies a variety of approaches to create a viable student degree recommendation system. The background section will be divided into five sections: Terminologies and Definitions, Content-Based Filtering, Collaborative Filtering, Knowledge-Based, and Hybrid Methods.

2.1 Basic Definitions & Terminologies

This section covers all relevant terminologies and definitions for areas of interest that are directly correlated with a student degree recommendation system. This section is divided into five subsections: Data Exploration and Preprocessing, Feature Selection and Model Selection, Recommendation Systems, Supervised Learning Techniques, and Model Interpretability.

2.1.1 Data Exploration and Preprocessing

- Data exploration is among of one the vital initial steps to extracting underlying patterns within a data set [Zhang, 2016]. Univariate data exploration methods, analyze a single feature's patterns such as measures of central tendency, measures of dispersion, histogram plots, and kurtosis [Zhang, 2016]. Bivariate data exploration techniques such as correlation heat

maps, scatter plots, and box plots are used frequently for extracting relations between a pair of numerical variables or numerical-categorical variables respectively [Zhang, 2016].

Real-world data can be very messy. In most cases, it can be very difficult to input raw data into a machine-learning pipeline and generate predictions or accurate results without data preprocessing.

- Data preprocessing is responsible for transforming the data set into a favorable format to be input to a model for performing classification. There are a variety of preprocessing steps such as checking for missing values using correlation heat maps, and feature encoding [Famili et al., 1997].

- Feature encoding transforms categorical features into numerical features for more effective analysis [Famili et al., 1997].

- Outlier detection is another vital aspect of data preprocessing. Outliers are defined as data points that do not exhibit normal behavior relative to the rest of the data [Singh and Upadhyaya, 2012].

Outliers can sometimes be difficult to identify due to many factors including the nature of the data set or the fact that some outliers can be referred to as meaningful outliers, which provide more information to the rest of the data set than when they are removed. Several techniques including box plots and Mahalanobis distance are deployed to detect outliers in the univariate and multivariate space respectively [Singh and Upadhyaya, 2012].

- Data augmentation is a crucial preprocessing step that is used to mitigate unbalanced data. Unbalanced data refers to an imbalance in the classes of the data set, which can affect model predictions and induce model bias [Mohammed et al., 2020].

This can be fixed by employing Random Oversampling [Mohammed et al., 2020]. Random Oversampling is a commonly used augmentation technique that balances the data by generating random samples for the minority classes by taking advantage of samples in the majority classes with replacement. However, it can be prone to overfitting [Mohammed et al., 2020].

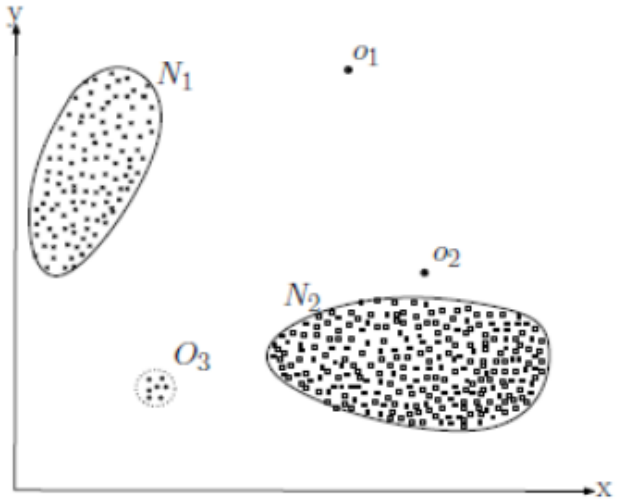


Figure 2.1: A simple example of outliers in a 2-dimensional data set. *Source*: [Singh and Upadhyaya, 2012].

2.1.2 Feature Selection and Model Selection

- Feature selection is regarded as an effective step in preparing data for data mining and machine learning procedures [Li et al., 2017]. Its objectives include constructing simpler and more effective models, thus improving the performance and interpretability of the data [Li et al., 2017]. This in turn provides valuable information for which models are more suitable after performing feature selection.

- Feature importance is a common measure used to identify the most important features in classification performance. Feature importance can be measured on a local or global scale. This can be useful in choosing k number of features to simplify the complexity of the data set for input to a classification model [Saarela and Jauhiainen, 2021]. Correlation heat maps can also be useful. Features that have a high correlation with the target variable are considered good variables. The Chi-square test is also a viable option for categorical features in a data set. Chi-square is calculated between the categorical feature and the target variable and the best k features are chosen based on their scores [Franke et al., 2012]. Furthermore, Principal Component Analysis (PCA) is another feature selection technique that identifies

the most meaningful principal components to re-express the data in a lower dimension while preserving as much information as possible [Kurita, 2019].

- Regarding model selection, there are a variety of subjective metrics that can be used to decide which model is best for a data set. First and foremost is whether a model is overfitting by memorizing the data rather than generalizing. Moreover, the complexity of the model is important to be able to extract interpretable information from the predictions.

2.1.3 Supervised Learning Techniques

- Supervised learning is a subcategory within machine learning. Supervised learning is defined by labeled datasets that can be used to train algorithms to accurately classify data or predict outcomes [Caruana and Niculescu-Mizil, 2006].

Training data is applied to a supervised algorithm to generate an effective classifier that can correctly classify data as well as form reasonable predictions. Many common algorithms are housed within supervised learning that operates on categorical as well as continuous data such as decision trees (DT), random forests (RF), Gaussian Naive Bayes, and Bernoulli Naive Bayes.

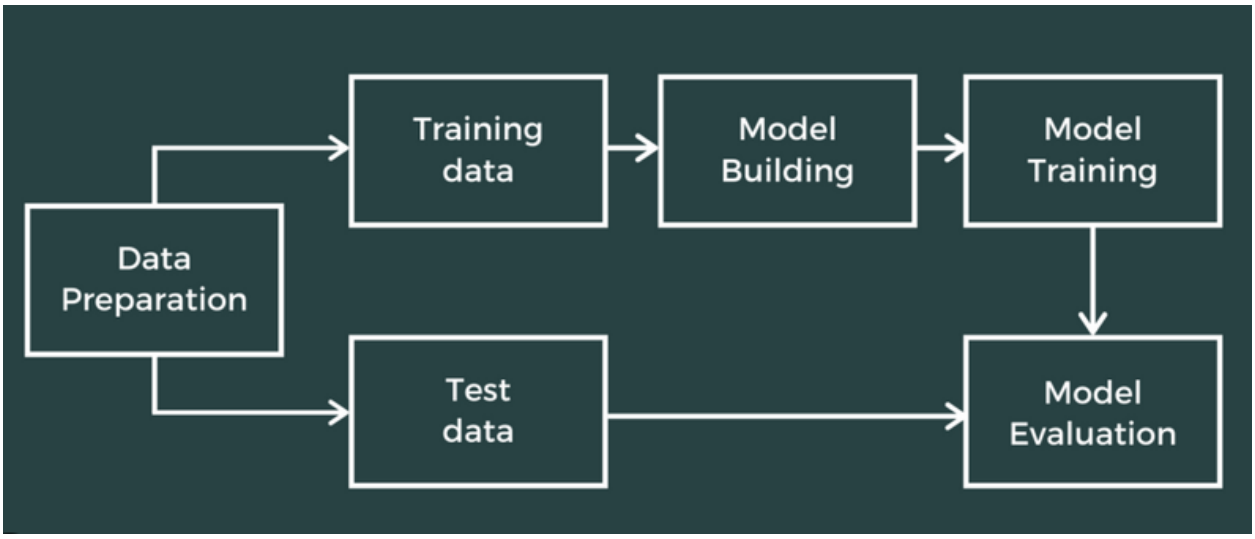


Figure 2.2: Supervised Learning Process. *Source*: [Reader, 2021].

- Decision trees are one of the simplest supervised learning algorithms due to their interpretability and simple execution compared to other algorithms [Priyam et al., 2013].

A series of questions are posed against the set of features input into the model. Each question is contained within a unique node that based on the decision tree algorithm, forms a splitting condition by which to traverse a side of the tree. This repeats iteratively until a stopping condition is met such as a hyper-parameter condition or the tree evaluates a decision path for all possible outcomes of the data [Kingsford and Salzberg, 2008].

There are various hyper-parameters for decision tree classifiers. However, only a few of the most impact ones will be considered such as: *criterion*, *max depth*, and *ccp alpha*.

The error criterion is a function that measures the quality of the split in the tree. One of the criteria is entropy, which gives a measure of the impurity of the nodes. The entropy is lowest when a single probability equals 1 and the rest are 0. The Gini impurity is another criterion measure, which ranges from 0 to 0.5 based on the impurity of a split. The goal is to minimize the impurities of these splits to maximize information gain from all splits in the tree [Kingsford and Salzberg, 2008].

$$Entropy = -\sum_{i=1}^{m} p_i log(p_i)\text{hi} \tag{2.1}$$

$$GiniImpurity = 1 - \sum_{i=1}^{m} p_i^2 \tag{2.2}$$

where:

p_i = Probability of the class i.

p_j = Probability of the class j.

The *max depth* references the maximum depth of the tree. This can be an important hyper-parameter to consider as this affects the number of splits in the tree as well as the quality of splits. If no restriction is given to the max depth, then the tree can continue splitting until every split accounts for an outcome in the training data. Consequently, this will produce many impure splits and is not a favorable approach. Optimizing this hyper-parameter is crucial to ensure a balance between the number of splits as well as the purity of the splits.

The regularization parameter is known as *ccp alpha* which is a pruning parameter that deletes nodes to potentially minimize overfitting. As a result, improving the reliability and generalization of a proposed decision tree classifier and eliminate any impure leaf nodes [Kingsford and Salzberg, 2008].

Decision trees are very useful in classification problems to break down labeled data through effective splits. However, they do have disadvantages such as the overfitting with large amounts of data. Furthermore, decision trees struggle to handle large data since a single tree will result in many node splits and lead to overfitting [Jijo and Abdulazeez, 2021].

- Random forest is an ensemble supervised learning algorithm that uses a collection of decision trees to perform its classification or regression [Albert et al., 2022].

Its basic functionality relies on a method called bagging, in which subsets of features of the training data are taken and used for training with the decision trees. The final classification or output is based on a majority vote between all the trees in the forest [Albert et al., 2022].

Regarding random forest hyper-parameters, the most impactful parameters are: *number estimators*, *criterion*, *max depth*, and *ccp alpha*. The criterion, max depth, and ccp alpha have already been discussed. However, the number of estimators in a random forest can significantly impact classification accuracy.

Modifying the number of estimators in a forest can mitigate the overfitting effect of a single decision tree by smoothing the decision boundary. As the number of trees increases, the decision boundary is smoothed gradually to reduce overfitting and generate strong accuracies.

The advantages of random forests are that they are easy to train and predict, as well as ease of use with high dimensional data. However, they tend to generate bias when dealing with categorical variables [Albert et al., 2022]. Furthermore, random forests are considered black box models as they are difficult to interpret how predictions are made due to the presence of multiple trees.

Among other supervised techniques are Naive Bayes algorithms that take advantage of Bayes' rule given the assumption that all observations of a dataset are independent [Webb et al., 2010]. Naive Bayes algorithms such as Gaussian Naive Bayes and Bernoulli Naive Bayes offer many theoretical advantages in comparison to tree-based models such as incremental learning where the

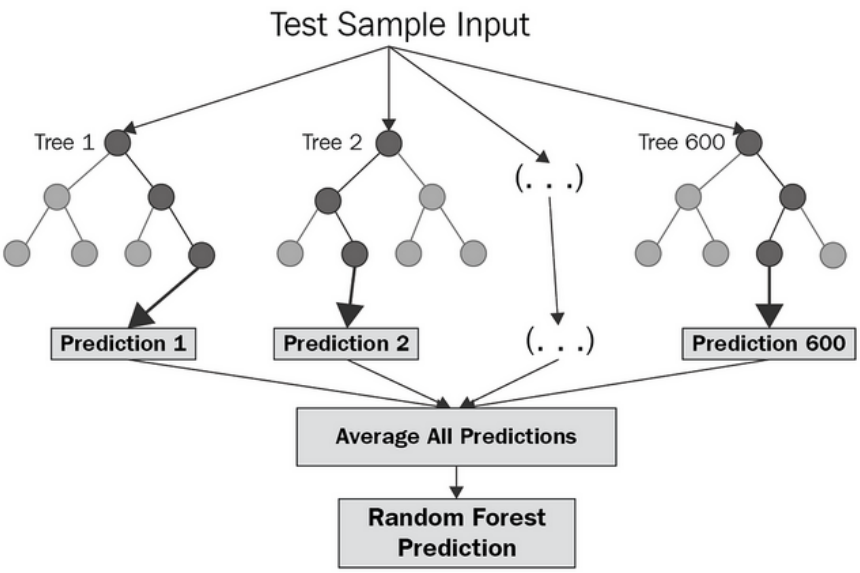


Figure 2.3: Random Forest Procedure. *Source*: [Albert et al., 2022].

probabilistic algorithms update the initial probabilities of each training sample to provide more accurate predictions. Furthermore, they are effective in handling noise within training data. However, certain Naive Bayes algorithms work best with a certain set of features that follow properties such as normality, or are binary variables by nature [Webb et al., 2010].

2.1.4 Interpretation of Machine Learning Models

Recommendation systems provide personalized recommendations to each user. Therefore, interpreting why a certain output was given is a crucial aspect of recommendation systems. This is where model interpretability is introduced. Some algorithms that are used in recommendation systems such as random forest are classified as black box models that do not provide information regarding their internal functions and why a certain output was produced [Molnar, 2022].

- Interpretability is the degree to which a person can understand why a decision was made. It is among one of the most important parts of predictive modeling as understanding why a decision was made by a model implies model reliability and trust [Molnar, 2022].

Interpretability has a large scope that pertains to algorithm transparency as well as global or local interpretations of our data [Molnar, 2022]. Consequently, interpretability in a recommendation system is vitally important to invoke trust from the user that the predicted output was reliable and provided explanations as to why a certain decision was made by the system. There are a variety of interpretability techniques including decision paths, feature importance, partial dependence plots, and Shapley values.

The black box problem in machine learning refers to the inability of a model to explain how it arrived at a particular decision or prediction. In other words, it is difficult to understand how a model arrived at its output because the model's internal workings are not transparent. This lack of transparency can make it difficult to trust machine learning models and can limit their usefulness in certain applications. The black box problem is one of the biggest issues facing AI/ML because most out-of-the-box machine learning systems only make the inputs and outputs of your model observable [Barredo Arrieta et al., 2020].

- Decision tree paths are among the most effective local interpretative methods used to understand how a prediction is made for each sample [Izza et al., 2020].

Concerning a decision tree model, a decision tree path illustrates the traversed nodes in the tree that lead to a final prediction in a leaf node [Izza et al., 2020]. Figure 2.8 highlights visited nodes for sample 0 to be predicted.

```
Rules used to predict sample 0:

decision node 0 : (X_test[0, 4] = 0.35) > 0.012500000186264515)
decision node 16 : (X_test[0, 13] = 0.75) > 0.2638888955116272)
decision node 38 : (X_test[0, 6] = 0.391304347826087) > 0.021739130839705467)
```

Figure 2.4: Sample decision path for an observation in decision tree. *Source*: Done by the researcher.

- Feature importance is another measure of interpretability. It is used to show the ranking of the importance of features in the prediction process. This is performed by assessing which features increase the error more often than others [Elshawi et al., 2019].

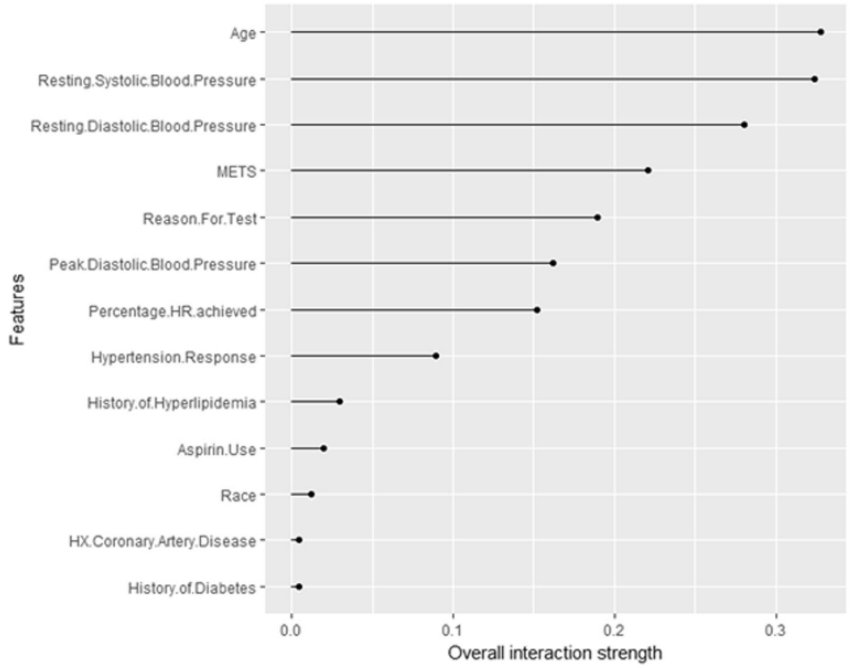


Figure 2.5: The interaction strength for each of the input features with all other features for predicting the high risk of hypertension. *Source*: [Elshawi et al., 2019].

- Partial Dependence Plots (PDP) are a very effective interpretation tool that offers insights into prediction results. PDPs are effective in representing the interaction between the target variable and a feature of interest by computing the average interaction [Elshawi et al., 2019].

- Shapley values are a method for assigning credit to each feature in a machine learning model's prediction [Barredo Arrieta et al., 2020].

The Shapley value can be defined as a function that uses only the marginal contributions of players as the arguments. Shapley values are a widely used approach from cooperative game theory that come with desirable properties such as illustrating contributions of each component to a relative outcome [Barredo Arrieta et al., 2020].

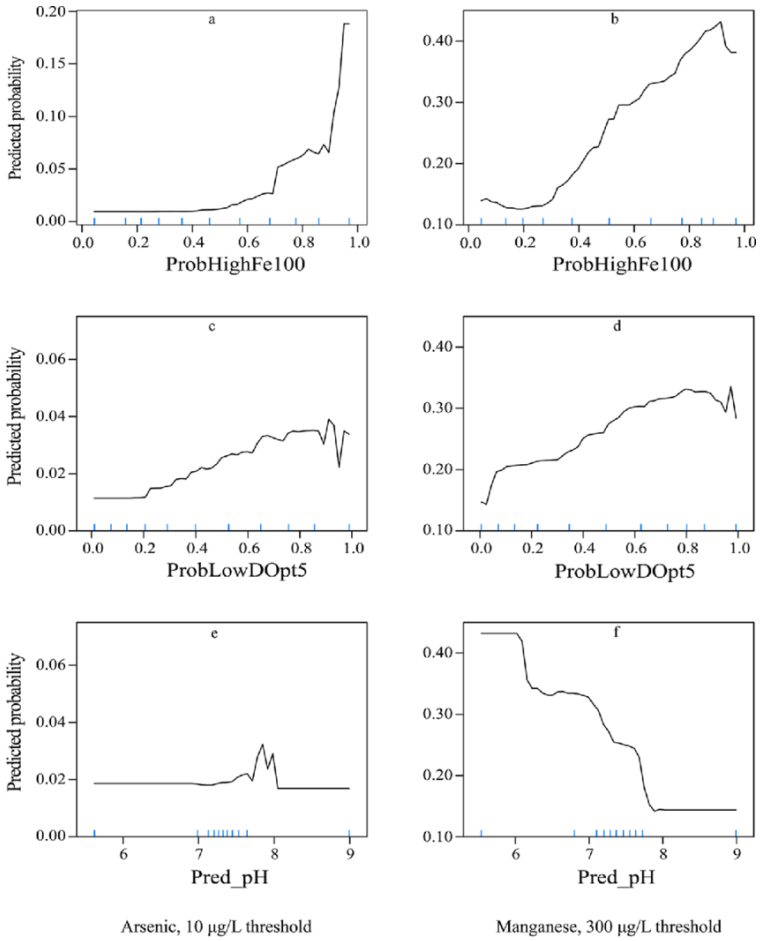


Figure 2.6: Example Partial Dependence Plots. *Source*: [Erickson et al., 2021].

2.1.5 Recommendation Systems

Recommendation systems (RS) are sophisticated machine learning algorithms that recommend a set of items or a collection of data to a user based on their specific preferences [Wang et al., 2019]. There is a plethora of recommender systems that are created and modified to cater to a vast range of fields such as e-commerce, digital entertainment, as well as the educational industry [Wang et al., 2019].

With the evolution of recommendation systems and the ways in which they were created, they were sub-categorized into various approaches to provide different kinds of recommendations based on different factors.

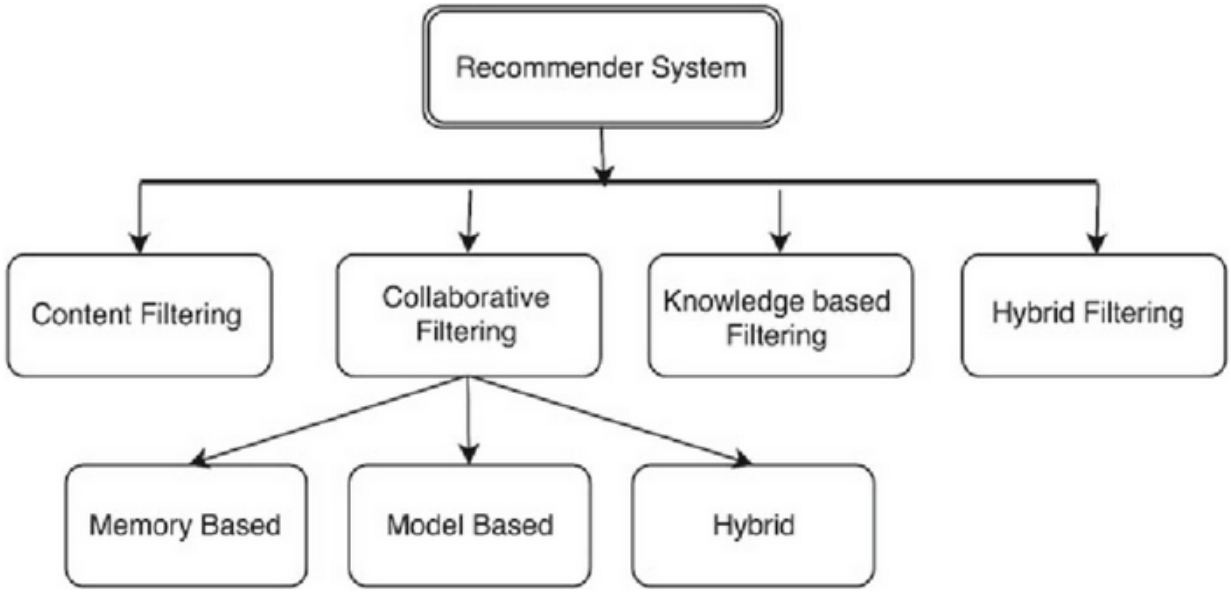


Figure 2.7: Classification of Recommendation Approaches. *Source*: [Raghuwanshi and Pateriya, 2019].

A type of recommendation approach is a content-based recommendation system. Content-based filtering works by using a first component known as a content analyzer. A content analyzer is used to structure unstructured data to withdraw relevant information [Lops et al., 2011]. Its main objective is to represent the relevant information in a matter for further preprocessing steps. Feature extraction methods are utilized to change the representation of the original information. After a content analyzer, the representation is fed into a profile learner.

A profile learner is then used to identify unique user preferences and try to generalize recommendations for each user [Lops et al., 2011]. Machine learning techniques are then used to produce a model that generates recommendations based on user interactions with certain items.

A filtering component takes advantage of the previous module to suggest relevant recommendations based on matching a user representation to that of the potential recommended items. This results in a list of items based on potential interest to the user [Lops et al., 2011]. A cosine similarity or some distance metric is deployed to identify the similarity between the resultant vector and the item vector.

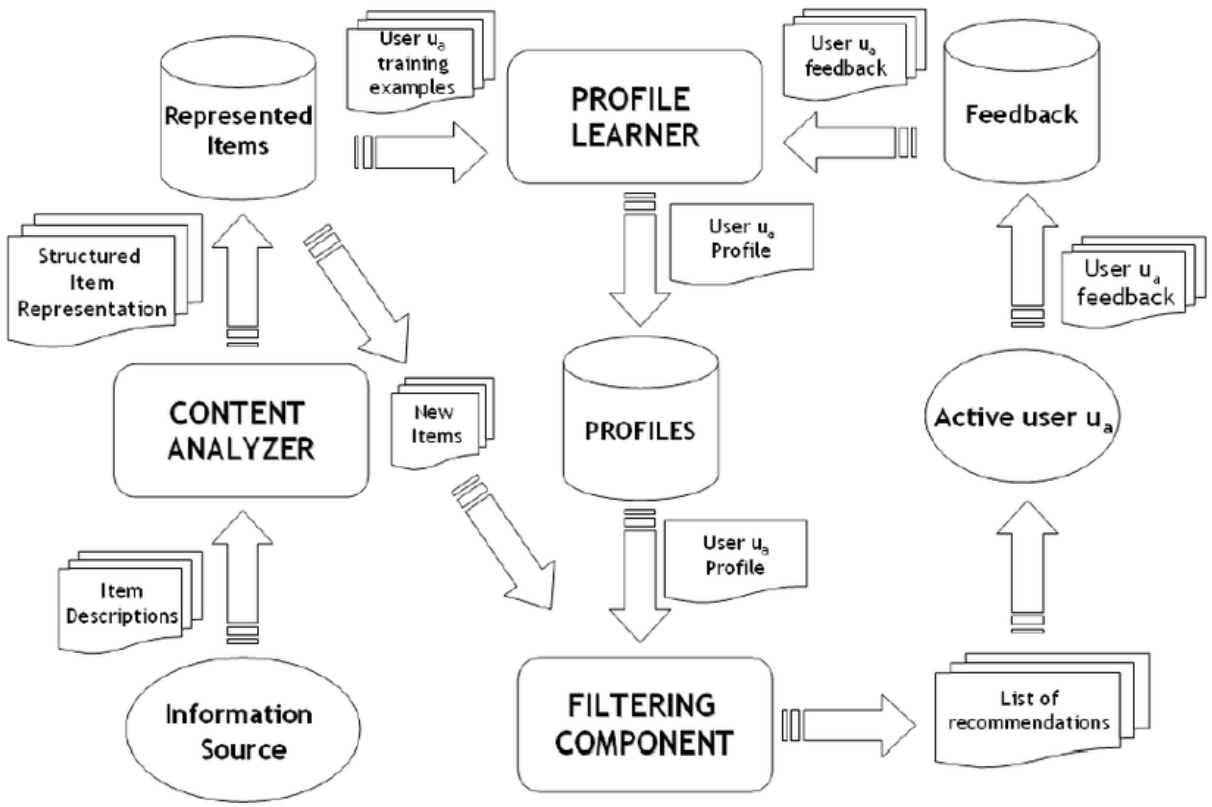


Figure 2.8: A high-level architecture of content-based recommendation system. *Source*: [Koene et al., 2015].

Another popular recommendation approach is collaborative filtering (CF), which works by representing unique users and items with unique representation vectors. Recommendations are considered using a matrix representing the users and their respective relations with all items. A collaborative filtering approach uses the relationships between users and their items through a similarity measure such as cosine similarity [Elahi et al., 2016]. There are various ways to implement a collaborative approach such as neighborhood-based models, and latent-factor models.

Neighborhood-based models can be either user or item-based. User-based approaches take advantage of the user rating in the representation vector as well as other users' similar representations [Elahi et al., 2016]. The item-based approach uses other users' ratings or scores of a certain item to predict a user's relation to an item.

Latent-factor models such as matrix factorization use a matrix of latent factors per user. The objective is to split the original matrix into two matrices S and M such that an approximation can be formed and predictions made for new items.

$$R \approx SM^T \tag{2.3}$$

where:

R = Original orthogonal rating matrix

S = $|U|*$F matrix. Represents the relation between students and their features

U = Matrix of unique users and relations with items

M^T = $|I|*$F matrix. Represents the relation between an item and features

F = Number of factors and is a parameter to be optimized

However, a disadvantage of CF recommendation is suffering from data sparsity and the cold start issue, by which there is no initial data for the recommendation system to generate initial recommendations. Hence other techniques have been developed to tackle these problems [Guo et al., 2020].

Another common recommendation approach is knowledge-based systems. Knowledge-based recommendation systems are heterogeneous graphs, whose nodes represent the unique users, and the edges are the relations between the user and the corresponding items. Knowledge-based systems can be designed in a variety of ways, including embedding-based methods, path-based methods, as well as unified methods which combine the former techniques to build the latter [Guo et al., 2020].

Embedding-based methods are created by building knowledge graphs with several item representations to better model users more accurately. Information such as user-specific graphs can be effective in providing unique representation and hence more accuracy. Entity embedding is the underlying methodology in embedding-based methods to extract key information from graph structures [Guo et al., 2020].

Path-based methods typically incorporate matrix factorization to enrich the user graph. Furthermore, interpretability is emphasized during the recommendation process when opting for a

path-based method. This is performed by matching the similarity of the item or user based on the meta-path level [Guo et al., 2020]. A unified approach builds on the knowledge graphs and semantic path patterns of both embedding-based and path-based methods respectively as well as inheriting the interpretability feature from path-based methods.

Our last recommendation approach is a hybrid recommendation system. They can be implemented in a variety of manners such as individually implementing a content-based and collaborative filter approach and aggregating the results of both. Another technique is generating an intermediate model that fuses characteristics from more than one recommendation approach [Thorat et al., 2015].

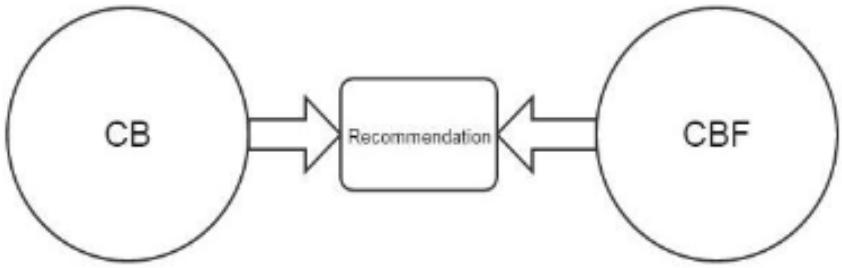


Figure 2.9: Shows the methods that integrate CB characteristics into the CF approach. Mitigates the cold start problem in collaborative filtering. *Source*: [Thorat et al., 2015].

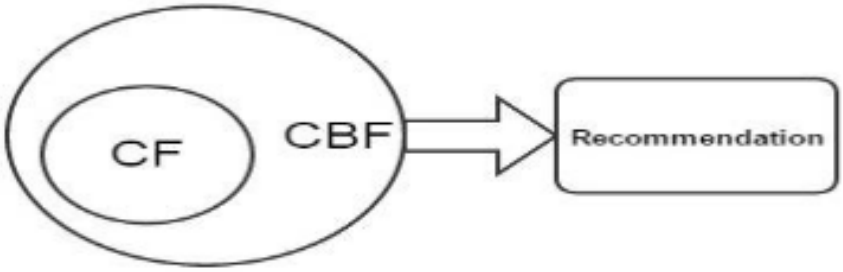


Figure 2.10: Shows the methods that incorporate CF characteristics into a CB approach. *Source*: [Thorat et al., 2015].

Predictions performed by hybrid systems can vary significantly depending on the original class which they are from. Predictions can be based on a weighted aggregation of prediction scores, or a

mixing of recommendations' predictions and picking one out. Feature combination and augmentation also play a role as a characteristic of hybrid systems as that can affect prediction outcome [Thorat et al., 2015].

2.2 Content-Based Filtering

Content-based recommendation systems are a popular technique used to gather information about user preferences. They are based on past preferences of users and recommendations are suggested with similar items of similar characteristics [Khanal et al., 2020]. Chen et al. [Chen et al., 2017] used correlation analysis in an experiment regarding the education field to group certain courses. This was performed by segmenting the data into three categories based on a rule-space model using a content-based approach to optimize the learning path for each individual.

Similarly, Shu et al. [Shu et al., 2018] utilized the historical data of students to create predictions regarding the provided learning materials using a content-based algorithm. The most common learning algorithms used in this domain are fuzzy-based as well as rule-based clustering that relies on a probabilistic methodology, and similarity between neighbors.

The advantages of content-based recommendation systems are that they handle each user as an independent user, meaning that user relations aren't influenced by other user relations. Furthermore, they provide transparency behind decisions based on content features [Lops et al., 2011]. However, a disadvantage of content-based recommendation is the over-reliance on past data to create predictions. It cannot overcome the cold start problem of having no data in the initial stages as well as data sparsity [Khanal et al., 2020].

2.3 Collaborative Filtering

Another favored technique in recommendation is collaborative filtering, which has been used frequently in recent systems as it mitigates the drawbacks of content-based filtering as was discussed earlier [Khanal et al., 2020]. Liu [Liu, 2019] recommended a collaborative filtering ap-

proach that focused on the influence of e-learning group behavior to improve the accuracy of predictions even in the presence of data sparsity. Moreover, collaborative filtering has been used with unsupervised learning.

El-Bishouty et al. [El-Bishouty et al., 2019] utilized a k-means algorithm to extract the learning path and objects of interest for each learner. In addition, M. Aljunid et al. [Aljunid and Dh, 2020] proposes a deep learning method for a collaborative filtering recommendation system using the concept of matrix factorization implemented within the deep network. Results demonstrated on the 100k and 1M movie lens datasets show an improved Root Mean Squared Error (RMSE) in comparison to other models such as cosine similarity and the dot product of matrix factorization [Aljunid and Dh, 2020].

Collaborative filtering does improve on content-based systems, but it still comes with its own set of difficulties. It can be difficult to create relations between attributes to their respective items, which can affect recommendation accuracy. It also suffers from cold-start and scalability issues [Khanal et al., 2020].

2.4 Knowledge-Based Approaches

Knowledge-based approaches provide recommendations based on how certain item features meet user needs. H. Wang et al. [Wang et al., 2019] proposed a knowledge graph convolutional network. The purpose of this architecture was to capture relationships between several items of interest to a user through data mining techniques. Some of the featured techniques involved association rules to identify relations between attributes on a graph. This was performed by sampling data from respective neighbors per entity in a particular graph and fusing the information already gained with the bias to calculate an accurate representation of each entity's graph relations. Three datasets were used for testing including movie, book, and music recommendation datasets. Results indicated that the proposed network outperformed the baseline recommendation techniques with an Area Under the Curve (AUC) of 0.9 or higher with two of the three datasets [Wang et al., 2019].

Wan and Niu [Wan and Niu, 2018] used a knowledge-based approach with an underlying self-organization method to propose learning objects. This improved accuracy but suffered from the increased time of computations and stacking of multiple algorithms.

2.5 Hybrid Methods

Certain hybrid methods have been experimented by Khanal et al. by combining content-based and collaborative filtering to counter the cold start problem [Khanal et al., 2020]. Hussain et al. [Hussain et al., 2019] opted to use a collection of models including an artificial neural network, decision tree, logistic regression, and support vector machine to predict the troubles students face during an online learning course.

Moreover, P. Kouki et al. [Kouki et al., 2019] deployed a hybrid system based on a probabilistic model to provide personalized recommendations. Furthermore, they pursued explainable artificial intelligence (XAI) from various perspectives including a crowd-sourced approach and mixed model statistical analysis to understand the relations between users and their relations. There are global and local interpretations to recommendation systems. Global interpretations rely on understanding interactions of certain features on outcome in an average manner across the entire dataset. Local interpretations analyze a single observation to understand the interactions for that single sample [Kouki et al., 2019].

Similarly, Karga and Satratzemi [Karga and Satratzemi, 2018] used a similarity matrix to create the relations between learners and their respective learning paths according to their needs and preferences. This methodology allows both content-based and collaborative filtering methods to complement each other's weaknesses while improving prediction accuracy [Khanal et al., 2020].

Chapter 3

Methodology

This chapter provides a comprehensive review of the methods used to create a functioning student degree recommendation system. This chapter is divided into the following eight sections: System Overview, Student Data, Data Exploration, Data Preprocessing, Feature Selection, Model Selection, Hyperparameter Optimization, and exclusively Local Model Interpretations as global interpretations are not the key objective of this thesis project.

3.1 System Overview

The primary objective of this thesis is to create a student degree recommendation system. That primary objective is comprised of a series of sub-goals to ensure a reliable and interpretable system. An emphasis of importance not is placed on model accuracies but on interpretation of predictions generated by our proposed system. An explanation for an outcome can be a very effective feature to provide users with reasoning behind the systems' recommendations and to potentially mitigate the black box effect or lack of interpretability of some of the proposed supervised classifiers and deep learning models.

This section illustrates the overall system overview of our proposed recommendation system. Various sequential steps are taken in order to obtain a reliable and trustworthy system. The first stage of the system is the data collection process. This involves choosing a dataset as well as

providing a brief description of the nature of the dataset. Following data collection is data exploration, which is comprised of a series of techniques and strategies to obtain a more transparent understanding of the data.

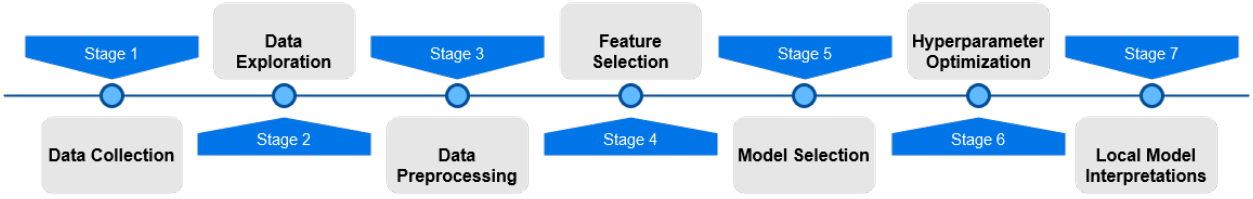


Figure 3.1: Architecture of System Overview. *Source*: Done by the researcher

Shortly after data exploration, several preprocessing steps will be undertaken to prepare the data for training during the later stages of the proposed system. Moreover, feature selection will be performed to observe the importance of certain features on the chosen evaluation metrics, the interpretability of the proposed model, as well as impact on decisions made by the models. Upon exploring feature selection, model selection is the next stage to observe performance comparisons as well as comparisons of model interpretability in a local space. Hyperparameter optimization is the next step to maximize the effectiveness of the proposed classifiers. Post optimization, local model interpretations are performed using a plethora of techniques to gain further insight into the inner workings of how an outcome or prediction was evaluated for each user.

3.2 Student Data

This section discusses the chosen data set used for training the proposed recommendation system as well as explanations of the nature of the data set and its features are also discussed.

3.2.1 University Student Dataset

The data set of choice is an entire record of information regarding a single undergraduate-level university student. Several features are collected to illustrate the performance of an individual student during his final years of high school such as their A-level mathematics or English grades.

A vast amount of features regarding academic performance are collected with care to ensure that our proposed model can effectively identify patterns and extract meaningful insights.

Furthermore, other variables were collected such as the actual major and specialization of each student, with both acting as the dependent variables in separate cases for comparison of results interpretation. Other factors are also considered including whether they are a transfer student or pursued a different curriculum of high school education such as IB or IG. More features are illustrated for potential post-processing and identifying the most impact or significant features relative to the predictions.

Table 3.1: Data Set Features Summary

Feature	**Explanation**
school type	The curriculum of education obtained during high school e.g. IB, IG
alevel math	Stores a numerical value between 0 and 100
olevel math	Stores a numerical value between 0 and 100
chem	Stores a numerical value between 0 and 100
phy	Stores a numerical value between 0 and 100
bio	Stores a numerical value between 0 and 100
english	Stores a numerical value between 0 and 100
ap course	A binary feature, whether student took any AP courses
cs ig	A binary feature, whether student took a computer science IG course
adv math	A binary feature, whether student took an advanced mathematics course
international student	A binary feature, a student had foreign high school education
transferred courses	Stores number of courses transferred
major	Describes student's chosen major e.g. BUS, CS, ENG
specialization	Describes student's chosen specialization e.g. Finance, Video Game
cgpa	Cumulative GPA of each student using 4.3 scale
credit hours completed	Describes number of completed credit hours of each student

3.3 Data Exploration

A variety of data exploration techniques are utilized in this section. A simple analysis of the data will be conducted to get an idea of the data set by extracting the first 5 observations as well as the shape of the data set to see how many student observations are available.

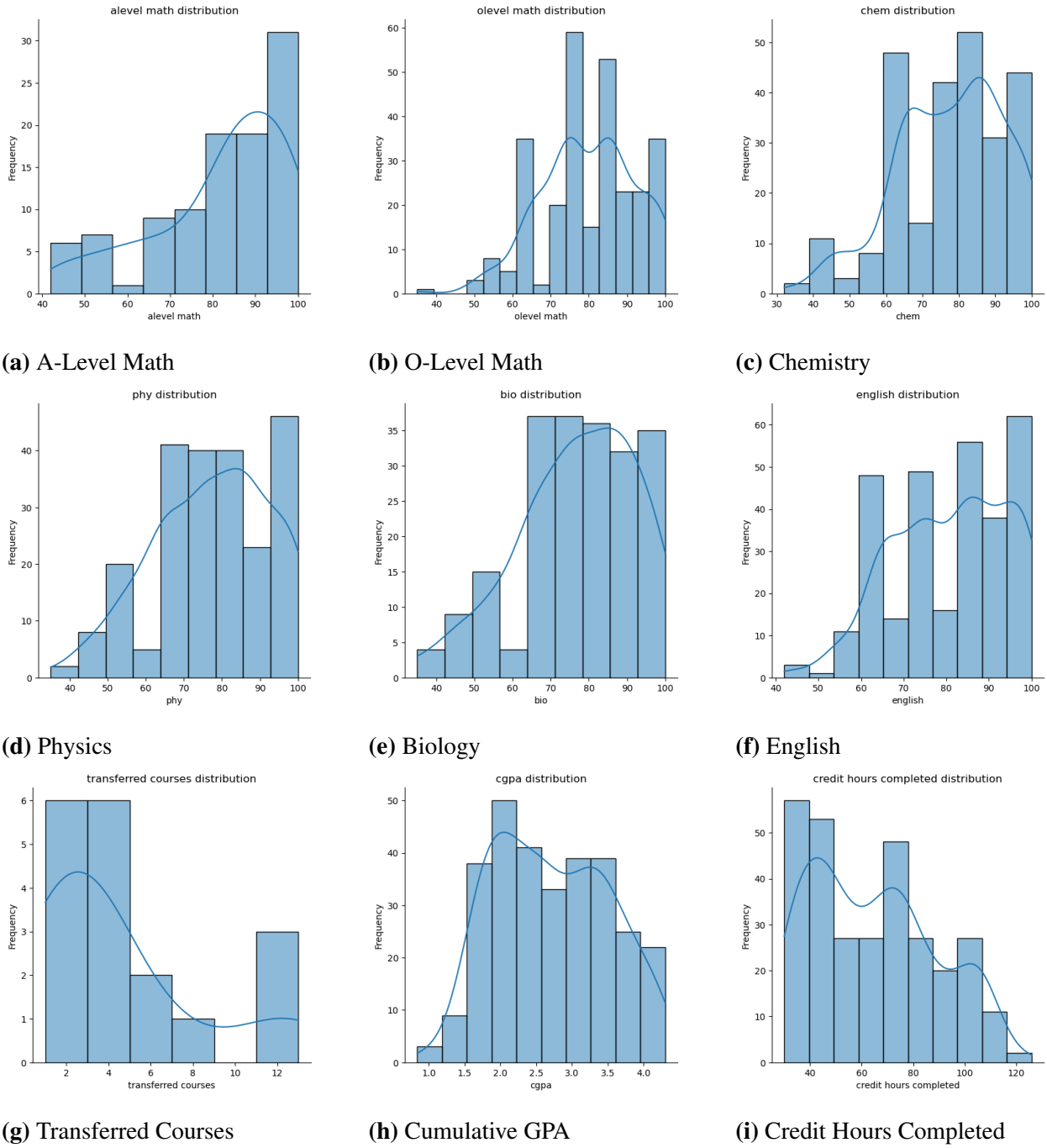
3.3.1 Univariate Exploration

Regarding univariate data exploration, the first plan of action is to describe the numerical features obtaining basic statistics such as the mean and standard deviation. This is performed to develop an understanding of the numerical variables and how they are reflected in the data set. Furthermore, histograms are plotted to find the distributions of the numerical features as well as the kurtosis to potentially find outliers.

	count	mean	std	min	25%	50%	75%	max
alevel math	297.0	28.111111	40.006812	0.00	0.000	0.00	75.00	100.0
olevel math	299.0	75.391304	22.076987	0.00	70.000	78.00	88.00	100.0
chem	298.0	66.563758	30.631446	0.00	65.000	75.00	86.75	100.0
phy	297.0	58.565657	35.788841	0.00	45.000	74.00	85.00	100.0
bio	297.0	54.070707	37.490337	0.00	0.000	69.00	85.00	100.0
english	299.0	80.812709	14.156545	0.00	70.000	84.00	93.00	100.0
ap course	298.0	0.013423	0.115270	0.00	0.000	0.00	0.00	1.0
cs ig	298.0	0.721477	7.331929	0.00	0.000	0.00	0.00	100.0
adv math	298.0	0.338926	0.474141	0.00	0.000	0.00	1.00	1.0
international student	299.0	0.090301	0.287093	0.00	0.000	0.00	0.00	1.0
transferred courses	299.0	0.267559	1.422017	0.00	0.000	0.00	0.00	13.0
cgpa	299.0	2.711940	0.798185	0.84	2.015	2.65	3.33	4.3
credit hours completed	299.0	64.568562	24.141595	30.00	45.000	63.00	81.00	126.0

Figure 3.2: Features Summary Statistics. *Source*: Done by the researcher

Concerning any categorical features, different techniques will be deployed such as how many categories are present in a certain feature, the most common category, value counts for each category, and a plot visualizing the proportions of categories as a percentage. Box plots are utilized to



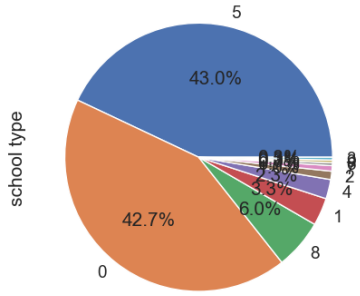
(a) A-Level Math **(b)** O-Level Math **(c)** Chemistry

(d) Physics **(e)** Biology **(f)** English

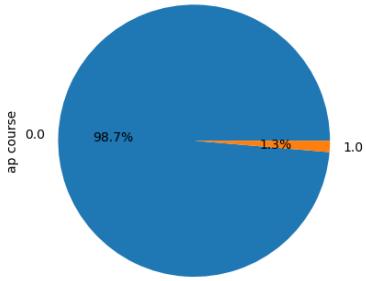
(g) Transferred Courses **(h)** Cumulative GPA **(i)** Credit Hours Completed

Figure 3.3: Histograms for the features. *Source*: Done by the researcher.

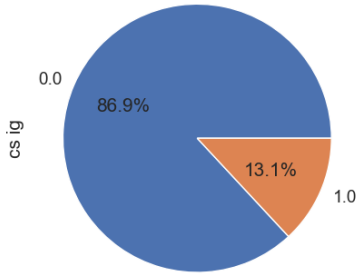
understand the proportion between values in each feature alongside bar plots to have a definitive sum of each category per feature.



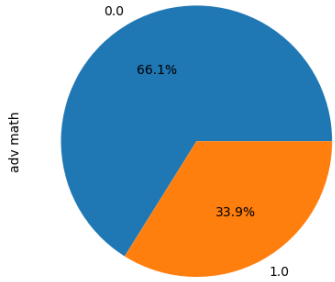
(a) School Type



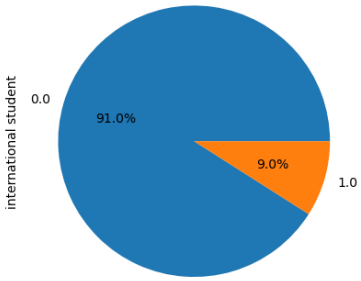
(b) AP Course



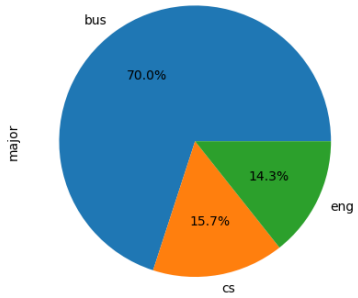
(c) CS IG



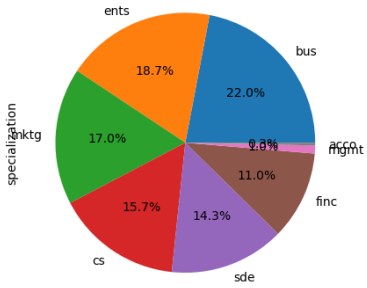
(d) Adv Math



(e) International Student



(f) Major



(g) Specialization

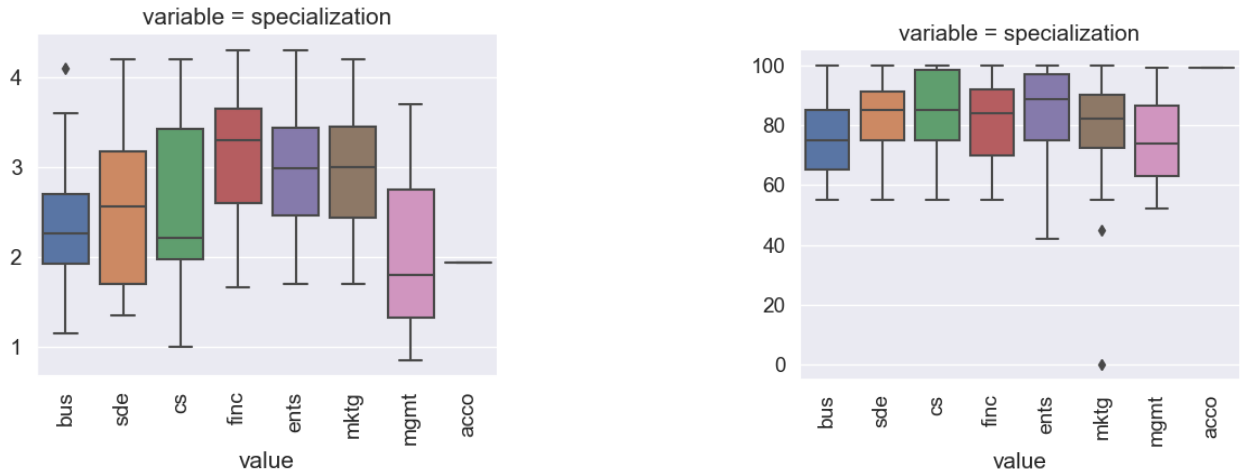
Figure 3.4: Pie plots of categorical features. *Source*: Done by the researcher.

3.3.2 Bivariate Exploration

In bivariate exploration, relationships between certain numerical variables with other numerical variables as well as numerical with categorical are explored. Regarding the former, scatter plots will be used to understand the relationships between two numerical features. As for the latter,

box plots will be the primary tool to understand the relations between a numerical and categorical variable.

One of the most useful measures of data exploration is a correlation heat map. It can serve as a great indicator for feature selection by visualizing relationships between a pair of numerical variables. For analyzing two categorical variables, the χ^2 test will be used to gain insight regarding the independence between the two variables by comparing it with a significance level of 95%, even though all categorical variables will be transformed later on in the preprocessing stage.



(a) CGPA-Specialization

(b) English-Specialization

Figure 3.5: Box plots between 2 features with specialization. *Source*: Done by the researcher.

Figure 3.5a and 3.5b show the cumulative GPA amongst each specialization as well as the English scores for each specialization respectively. Finance students have higher GPA than other specializations for the former, while in the latter the highest English scores belong to the Computer Science and Entrepreneurship students.

3.4 Data Preprocessing

In this section, the data preprocessing steps are highlighted to prepare the raw data to be trained for a machine learning model. The same data set is used twice, once with specialization as the label and major as the label for the other data set. For the former, the major variable is dropped as well as

the school id as they do not provide much information. As for the latter, specialization and school id will be removed.

3.4.1 Missing Values

The first step in the preprocessing stage will be to check missing values, which can occur through human error or failures of measurement.

To check for missing values, built-in pandas methods will be utilized such as *isnull()* to check for any null values in any of the variables in the data.

Missing values will be mitigated using a filling method that replaces missing values with the median value of each feature relative to the respective specialization or major category. However, none of the observations will be deleted or modified as those observations may have meaningful information to contribute to the data set.

```
specialization           0
school id                0
school type              0
alevel math              3
olevel math              1
chem                     2
phy                      3
bio                      3
english                  1
ap course                2
cs ig                    2
adv math                 2
international student    1
transferred courses      1
major                    0
cgpa                     1
credit hours completed   1
dtype: int64
```

Figure 3.6: Checking for missing values. *Source*: Done by the researcher.

3.4.2 Feature Encoding

Transforming categorical features including the target variable into numerical features using a label encoder. Upon changing all the categorical features into numeric values, the data is rear-

ranged to have the target variable at the start of the data frame, which has no significance in terms of model performance, purely for convenience.

3.4.3 Outlier Detection & Analysis

Outlier detection is a vital aspect of data preprocessing. Outliers can occur due to many reasons such as data entry errors, and measurement errors, or the observation can be a natural outlier within our data.

Concerning univariate outliers, box plots are used to visualize any potential outliers in a single feature of the data. However, for multivariate outliers, the Mahalnobis distance is deployed. Using Mahalanobis distance, the χ^2 test will be used to find outlier rows in the multivariate context with a statistical confidence of 95%. After obtaining the potential list of outliers, a simple numerical analysis will be performed to understand why those observations we're classified as outliers beyond just the respective Mahalanobis distance value.

	specialization	alevel math	olevel math	chem	phy	bio	english	ap course	cs ig	adv math	international student	transferred courses	cgpa	credit hours completed	school type	mahalanobis	p
18	1	81.0	84.0	67.0	0.0	67.0	74.0	1.0	0.0	1.0	0.0	0.0	2.62	102.0	0	84.929611	2.125744e-11
20	1	62.0	50.0	0.0	0.0	70.0	61.0	0.0	0.0	1.0	1.0	0.0	2.02	39.0	0	28.691843	2.609874e-02
21	2	0.0	85.0	65.0	76.0	92.0	88.0	1.0	0.0	0.0	1.0	0.0	3.88	109.0	0	91.679644	1.225575e-12
37	7	0.0	77.0	52.0	52.0	57.0	65.0	0.0	0.0	0.0	0.0	13.0	3.95	126.0	6	90.049163	2.450151e-12
43	2	0.0	60.0	68.0	62.0	0.0	61.0	1.0	0.0	0.0	0.0	0.0	2.16	33.0	0	88.781455	4.192313e-12

Figure 3.7: Sample of multivariate outliers using mahalanobis and χ^2. *Source*: Done by the researcher.

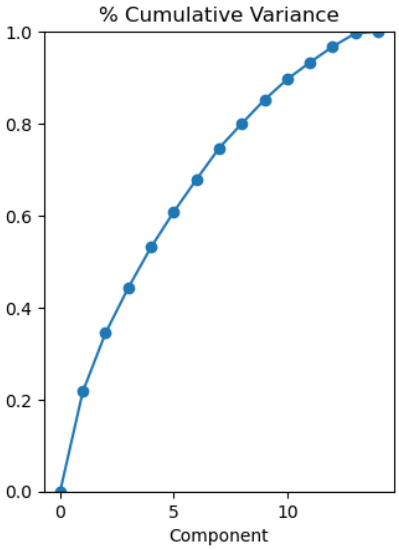
3.4.4 Data Augmentation

The final preprocessing step will be to perform data augmentation to increase the size of the data. The nature of the label variable is very unbalanced, which could lead to biased predictions. Therefore, data augmentation is proposed using a random over sampler to mitigate this. Robust scaling will be performed before augmentation because some of the models that will be used alongside the proposed model require that the data be scaled to reduce the range of values within the data.

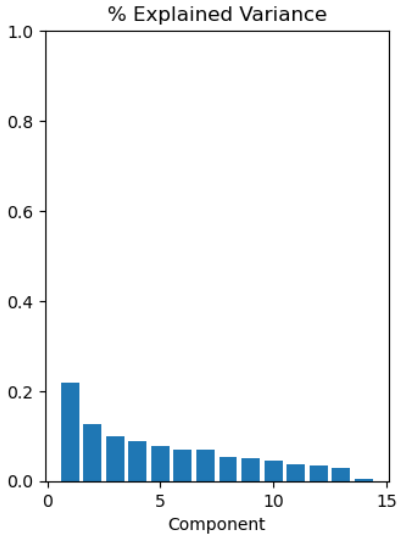
3.5 Feature Selection

Feature selection is an important part of the analysis as it will save a lot of computational expenses as well as provide more effective predictive power to the proposed model. Three techniques will be used to understand more about which features in the data provide more information or are considered important for predictions. The first feature selection technique is the correlation matrix of the data as was mentioned previously in Figure 3.7. The objective in interpretation is to identify features that correlate highly with the target variable while also being relatively uncorrelated with other features. A key note to mention is that the correlation will be calculated without the zero values present in some of the numerical features. This ensures the computations are not biased or heavily affected by meaningful zero values.

The second technique is tree-based feature importance. Utilizing the random forest classifier, feature importance is plotted to visualize which features a random forest considered to be among the most important while taking several subsets of the feature set. Thirdly, principal component analysis is computed to analyze the most impactful principal components and to potentially reduce the dimensionality of the data.



(a) Cumulative variance of principal components



(b) Individual variance of principal components

Figure 3.8: PCA plots. *Source*: Done by the researcher.

3.6 Model Selection and Fusion

Regarding model selection and fusion, a Random Forest model is proposed to perform classification. Random forest models excel in providing a weighted average of predictions between their estimators. Furthermore, random forest is a model that is considered resilient to overfitting as well as producing lower complexity tree structures. For the basis of having interpretable results, low-complexity trees are exactly what is needed.

Other supervised classifiers will be used for comparison including decision tree classifiers, Gaussian naive Bayes, Bernoulli naive Bayes, as well as a few soft voting classifiers that will combine the knowledge gained from each model.

The Decision tree was chosen to understand if a single tree could generate the same results as an ensemble of trees. The Gaussian and Bernoulli models were chosen to compare between probabilistic models and tree-based models. Voting classifiers were employed to understand how models can be fused to improve accuracy and interpretability. Other classifiers that used stacking or boosting were not considered due to the small size of the data available.

3.7 Hyperparameter Optimization

Regarding hyperparameter optimization, a grid search technique is used to optimize the aforementioned classifiers. Grid search will perform an exhaustive search for the optimal hyperparameters given a parameter space per classifier. By optimizing the models, the prediction accuracy becomes more accurate and confident, which will have a cascading effect on model interpretations.

Performing a grid search requires a parameter grid for each involved classifier. Hyperparameter grids are provided for all models whose respective hyperparameters could have significant effects on accuracy, and interpretation both in the positive and negative context.

Random Forest and Decision Tree parameter grids were optimized based on the number of estimators, pruning parameter, max depth, and error criterion with varying values. The voting classifiers were optimized through the weights of the classifiers and whether it was a soft or hard voting classifier.

3.8 Local Model Interpretations

This section provides a summary of the variety of methods used to perform local interpretations after the training and testing stage. This section is divided into 4 subsections, with each section being used on a variety of different classifiers to understand how each model can generate different interpretations for a single observation or set of observations. The following subsections are Feature Importance, Partial Dependence Plots (PDP), and Shapley Values.

3.8.1 Feature Importance

Feature importance is another useful metric for interpretation. This will be used with both decision trees and the random forest classifiers to understand how feature importance can change with a single tree or an ensemble of trees.

Feature importance is calculated using Gini importance as shown in Equation 3.1. This evaluates the feature importance for a binary classification task. However, for a multi-class classification, a one-vs-all or one-vs-one mechanism can be used by the models to account for the larger number of label values. A total average of the node impurities is calculated over the ensemble and a value is returned for each feature [Saarela and Jauhiainen, 2021].

$$\text{Gini Importance} = p_1(1 - p_1) + p_2(1 - p_2) \tag{3.1}$$

where:

p_1 = Probabilities of class 1

p_2 = Probabilities of class 2

3.8.2 Partial Dependence Plots (PDP)

Partial dependence plots are another useful interpretation tool to understand the nature of the interactions of the features with the target variable. Only after the models have been fitted can partial dependence plots be applied to understand how each pair of variables affect each other.

The prediction value is set to a subset of various values and allows the features to interact and the plot aims to visualize the interaction between an independent feature and the target label [Elshawi et al., 2019].

3.8.3 Shapley Values

For the last and most significant interpretation tool, Shapley values are among the most effective local interpretation tools for this use case. Shapley values excel at not only identifying what features are important, but how those features impact model predictions. Using the shapley library, the previous methods such as feature importance will be used as a supplement to Shapley values to visualize which features in descending order are the most impactful and how they affect the prediction score across every sample in the training data.

The basic premise of Shapley values is that a feature is chosen and removed from the set of features in the dataset temporarily. The model is evaluated without the removed feature and then evaluated again by inserting the feature back. The difference of the model evaluations is computed and that represents the individual contribution of each feature [Elshawi et al., 2019].

Shapley values were favored compared to another local measure like LIME because they provide clear numerical feedback regarding each feature. As opposed to LIME, which only provides a value between -1 and 1 that signifies the direction of influence of a feature. Other techniques could have been used but were ignored to maintain a contained scope using only three distinct measures of interpretability.

Chapter 4

Results and Discussion

This chapter highlights the main analyses of this thesis. An emphasis is placed on the data collection process, evaluation metrics used for measuring model efficiency, as well as system analysis ranging from the nature of the data set to the post-testing interpretation of the predictions.

4.1 Dataset

The dataset of choice was a student information data set. This dataset spans a collection of 300 observations, which consist of a variety of variables as illustrated in Table 3.1. The variables are the following:

- school type
- alevel math score
- olevel math score
- chemistry score
- physics score
- biology score
- english score

- ap course binary variable
- cs ig binary variable
- adv math binary variable
- international student binary variable
- number of transferred courses
- major (label 1)
- specialization (label 2)
- cumulative GPA
- credit hours completed

It was entirely collected by the university through manual input into a text file. The sample selection methodology is purely based on available data of recent graduates and future graduate students available to the university.

The dataset contains two significant variables: specialization and major which will both be independently used as the target variable for classifying a student's specialization or major based on the rest of the independent features.

4.2 Evaluation Metrics

Regarding the evaluation metrics, a collection of classification metrics were chosen that could assess the performance of the proposed random forest model alongside the other supervised classifiers: Decision Tree, Gaussian Naive Bayes, and Soft Voting Classifiers. These metrics are the F1-score, precision, recall, and accuracy as shown in Equations 4.1 to 4.4.

$$F1 - \text{Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.1}$$

$$Precision = \frac{\text{TruePositives } (TP)}{TP + \text{ FalsePositives } (FP)} \tag{4.2}$$

$$Recall = \frac{TP}{TP + \text{ FalseNegatives } (FN)} \tag{4.3}$$

$$Accuracy = \frac{TP + \text{ TrueNegatives } (TN)}{TP + TN + FP + FN} \tag{4.4}$$

where:

TP = True positives (Outcome where model correctly predicts the right specialization/major)

FP = False positives (Outcome where model correctly predicts the wrong specialization/major)

TN = True negatives (Outcome where model correctly predicts an incorrect specialization/major value)

FN = False negatives (Outcome where model incorrectly predicts the wrong specialization/major value)

4.2.1 Criteria of Selection

The criteria of selection for choosing the best model is the accuracy evaluation metric. By identifying the sum total of correctly classified observations divided by the sum total of observations, this will provide a reliable metric to compare the models at hand.

4.3 Results

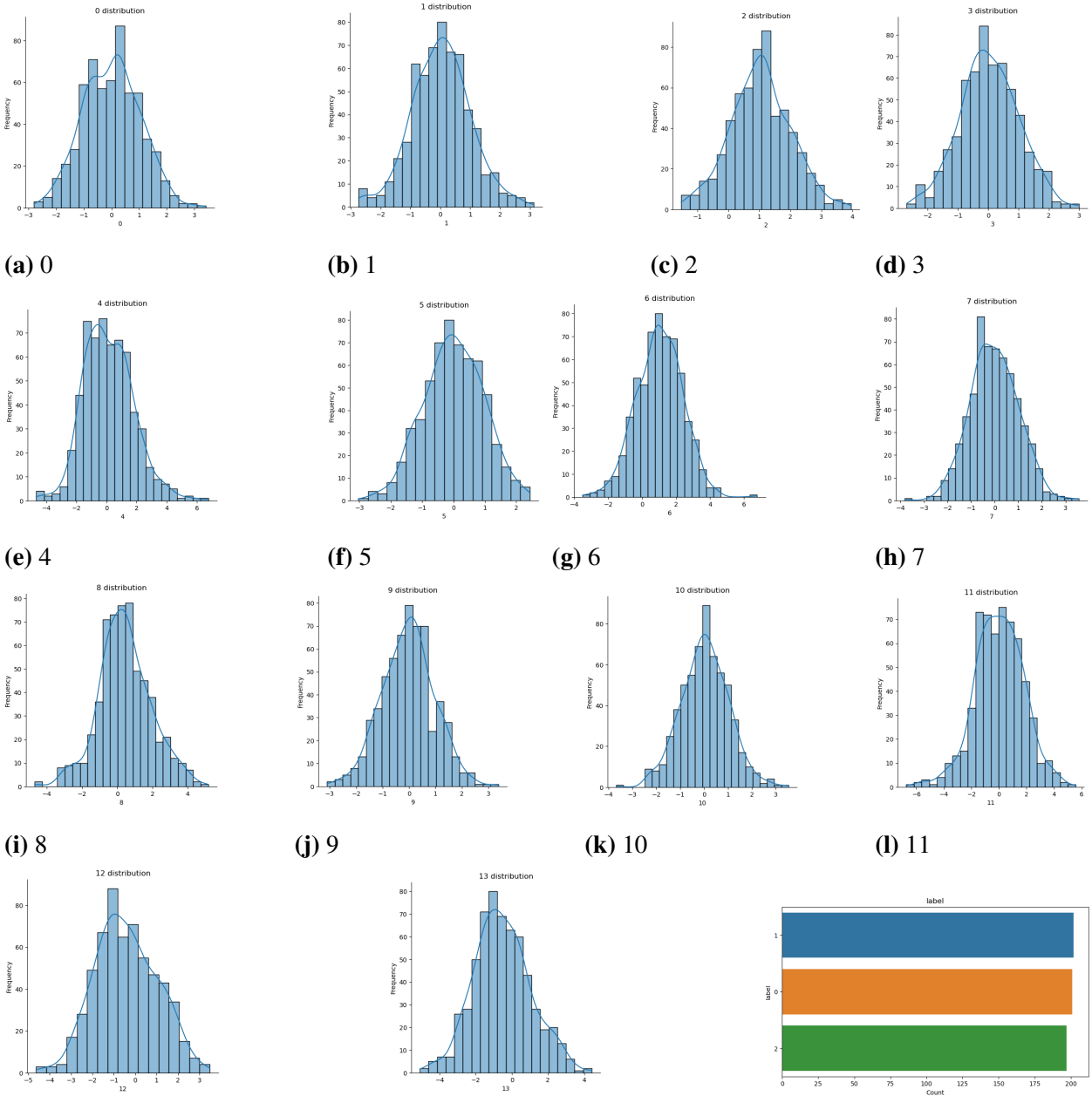
4.3.1 Synthetic Analysis

Before executing the methodology pipeline to the chosen dataset, synthetic data was created using *make classification* from the sklearn library to test the efficiency of the pipeline. The dataset is purely numerical and is comprised of 600 observations, a total of 14 features (excluding the target variable), and 5 features declared as informative to the target variable. Furthermore, the number of classes is set to 3 to try to replicate the multi-class task of the original dataset.

For data exploration, Figure 4.1 illustrates the distribution of the features as well as the bar plot for the label variable. The features are all normally distributed and with a balanced label, which is expected as *make classification* randomly clusters the observations to a normal distribution as is also shown by the random scatter of points for many of the scatter plots in Figure 4.2.

Regarding preprocessing, there are no missing values present in the data. However, 22 outliers were detected based on applying the Mahalanobis distance and computing the p-value with confidence of 95%. The target variable is automatically balanced due to *make classification* generating the data to be balanced automatically.

Regarding feature selection, the three techniques mentioned in the pipeline were: Random Forest Feature Importance, Correlation Matrix, and PCA. The feature important plot in Figure 4.3 highlights the top 7 most important features. It would seem that even though 5 features were explicitly told to be informative, the importance was relatively distributed among the top 7 and perhaps even further down to the other features. Figure 4.4 shows the correlation matrix. Among the most highly correlated features with the target label is feature 5 in the positive direction, while the rest of the features are either weakly correlated in the positive or negative direction. However, there seems to be a substantial amount of correlations between the features themselves such as 0 and 6, which could indicate some redundant features. Figure 4.5 shows the principal components of the features, and it shows that dimensionality reduction isn't an option as there is no clear cut-off point or elbow in the cumulative variance plot.



(a) 0 **(b)** 1 **(c)** 2 **(d)** 3

(e) 4 **(f)** 5 **(g)** 6 **(h)** 7

(i) 8 **(j)** 9 **(k)** 10 **(l)** 11

(m) 12 **(n)** 13 **(o)** Bar chart for label

Figure 4.1: Histograms for synthetic features and bar chart for label. *Source*: Done by the researcher.

For model selection, the proposed model is a random forest classifier as well as a collection of other supervised models. These models will be optimized using a grid search to maximize their performance with the synthetic data.

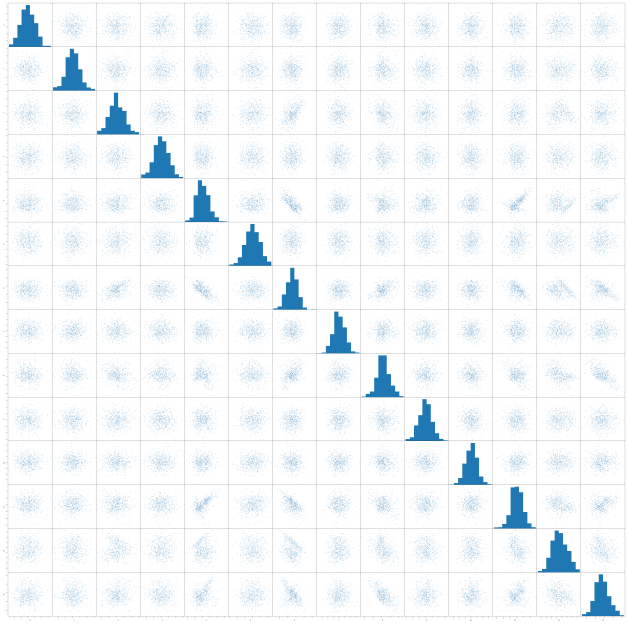


Figure 4.2: Scatter pair plot of all synthetic features. *Source*: Done by the researcher.

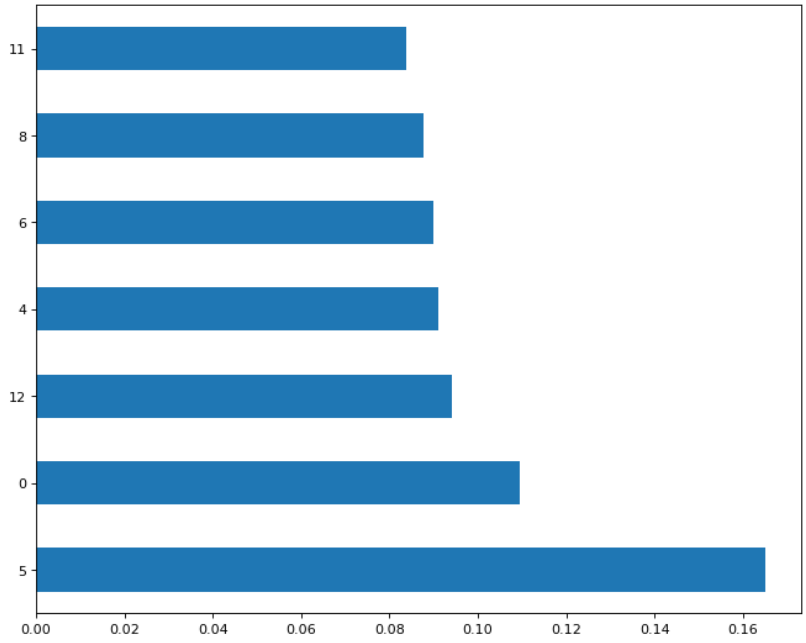


Figure 4.3: Feature importance of synthetic features. *Source*: Done by the researcher.

The optimized hyperparameters for some of the chosen models are displayed in Table 4.1 as probability-based models had no hyperparameters to optimize.

The dataset was split into a train-test split of 80% and 20% respectively. The classification report metrics are displayed in Table 4.2.

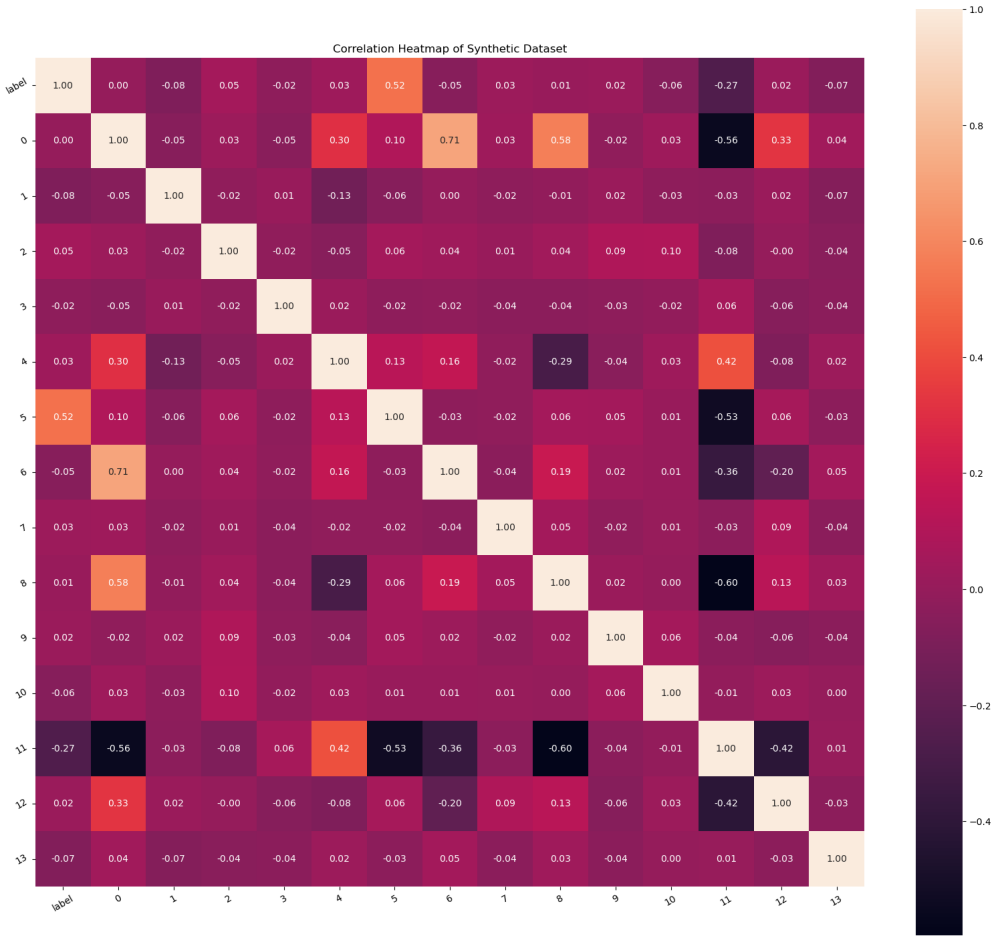


Figure 4.4: Correlation matrix of synthetic features. *Source*: Done by the researcher.

Model	Hyperparameter	Value
Random Forest	Estimators	70
	Criterion	Entropy
	Max Depth	None
	CCP Alpha	0.001
Decision Tree	Criterion	Gini
	Max Depth	50
	CCP Alpha	0.005
Voting Classifiers	Voting	Soft
	Weights	(2,1,1)

Table 4.1: Hyperparameter optimization for various models on synthetic data set.

Feature importance was the first measure of local model interpretations, alongside computing the shapely values and plotting them in a partial dependence plot. Figure 4.6 displays the fea-

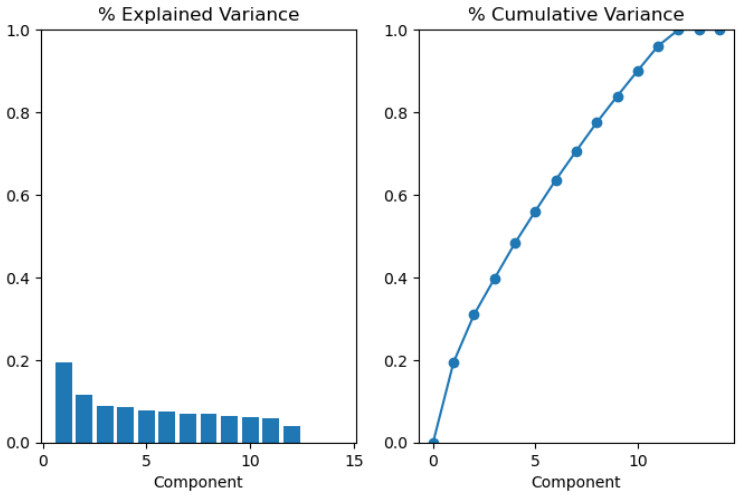


Figure 4.5: Explained and cumulative variance for synthetic principal components. *Source*: Done by the researcher.

Classifier	**F1**	**Precision**	**Recall**	**Accuracy**
Random Forest	0.81	0.81	0.81	0.81
Decision Tree	0.76	0.76	0.76	0.76
Gaussian Naive Bayes	0.72	0.72	0.72	0.72
Voting Classifier (RF, DT, Gaussian NB)	0.79	0.80	0.79	0.79
Voting Classifier (RF, Bernoulli NB, Gaussian NB)	0.80	0.80	0.80	0.80
Best Model (Random Forest)	**0.81**	**0.81**	**0.81**	**0.81**

Table 4.2: Comparison of classifiers performances on synthetic data.

ture's importance. Importance seems to be fairly well distributed among the top 7 features despite the explicit instruction of having only 5 informative features. This has changed because the proposed random forest takes priority when evaluating the important features among its ensemble of estimators.

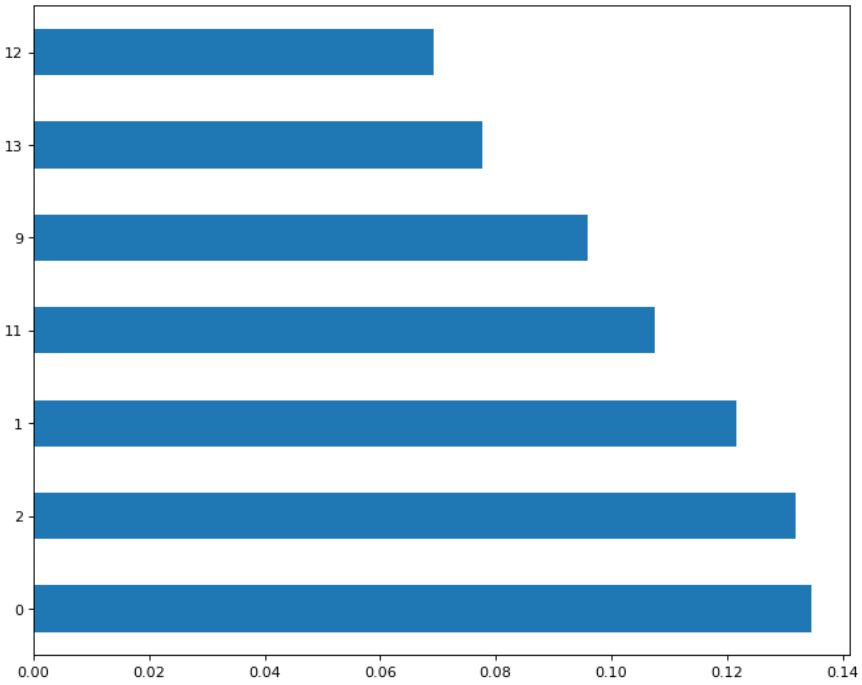


Figure 4.6: Feature importance of optimized random forest model on synthetic data. *Source*: Done by the researcher.

As for computing the shapely values, Figure 4.7 highlights the trend among the training samples with the y-axis representing the prediction accuracy of each sample while the x-axis represents the individual numerical contribution of each feature towards their respective sample prediction accuracy.

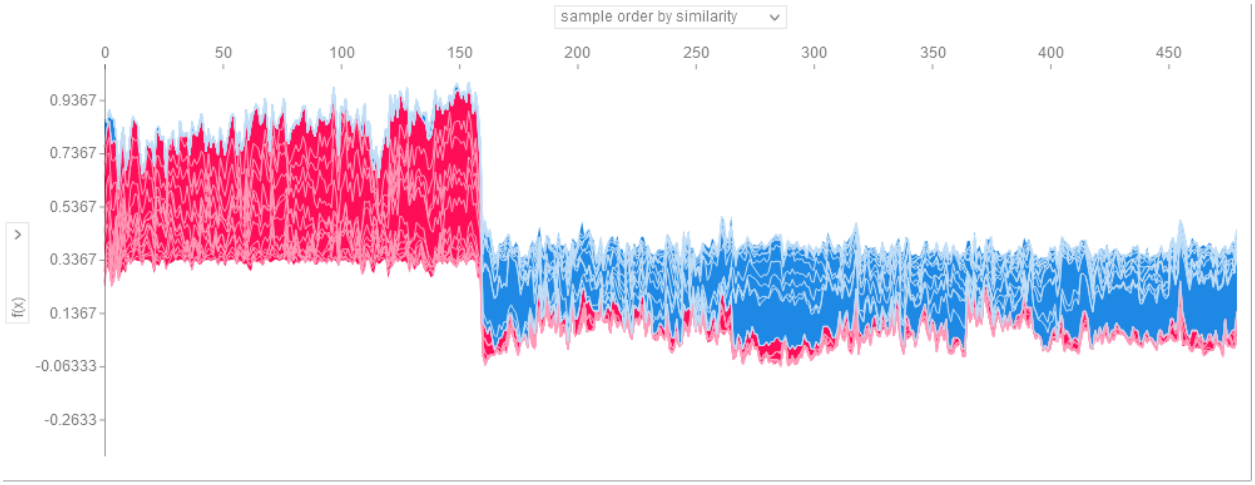
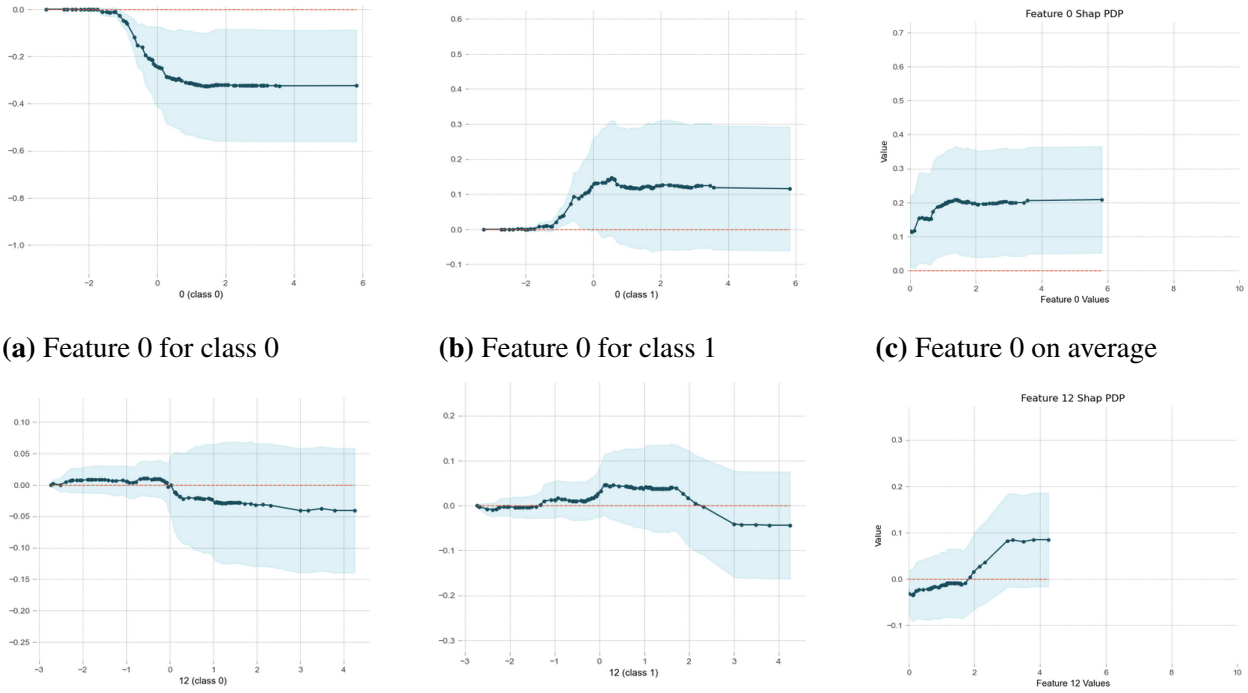


Figure 4.7: Shapely values on synthetic training data. *Source*: Done by the researcher.

Plotting these individual features using a partial dependence plot will aid in discovering the sole impact of a feature on the prediction accuracy of a sample as shown in Figure 4.8.



(a) Feature 0 for class 0

(b) Feature 0 for class 1

(c) Feature 0 on average

(d) Feature 12 for class 0

(e) Feature 12 for class 1

(f) Feature 12 on average

Figure 4.8: Partial dependence plots for features across different class values. *Source*: Done by the researcher.

4.3.2 Student Specialization Analysis

As for the student dataset with specialization as the target variable, the same pipeline is followed to produce effective results and extract valuable interpretable information. The major variable is omitted from the dataset as both variables imply each other and wouldn't allow the rest of the features to contribute as much. The advanced math variable is also removed as it correlated very highly with the A-level math scores and was considered a redundant variable. Concerning data exploration, Figure (3.3-3.4) provides detailed insight into the features' distributions as well as the categorical features' proportions. Furthermore, Figure 3.5 provides sample box plots regarding relationships between the specialization label along with some of the numeric features such as cumulative GPA or English scores.

Figure 3.7 shows that there are present outliers among the dataset that needs to be addressed. Missing values are mitigated by firstly grouping by the specialization and then filling any empty values with the median value of the respective features corresponding to which specialization group they belong to. In addition to handling missing values, feature encoding is applied to the specialization and school-type features to convert them to discrete values. Table 4.3 provides a clear summary of the label before and after encoding. Concerning outlier detection, 16 outliers were detected among the 7 classes, which will be discussed later.

Original Label	Encoding
ACCO (Accounting)	0
BUS (Business)	1
CS (Computer Science)	2
ENTS (Entrepreneurship)	3
FINC (Finance)	4
MGMT (Management)	5
MKTG (Marketing)	6
SDE (Sustainable Design Engineering)	7

Table 4.3: Feature encoding of specialization variable using label encoder.

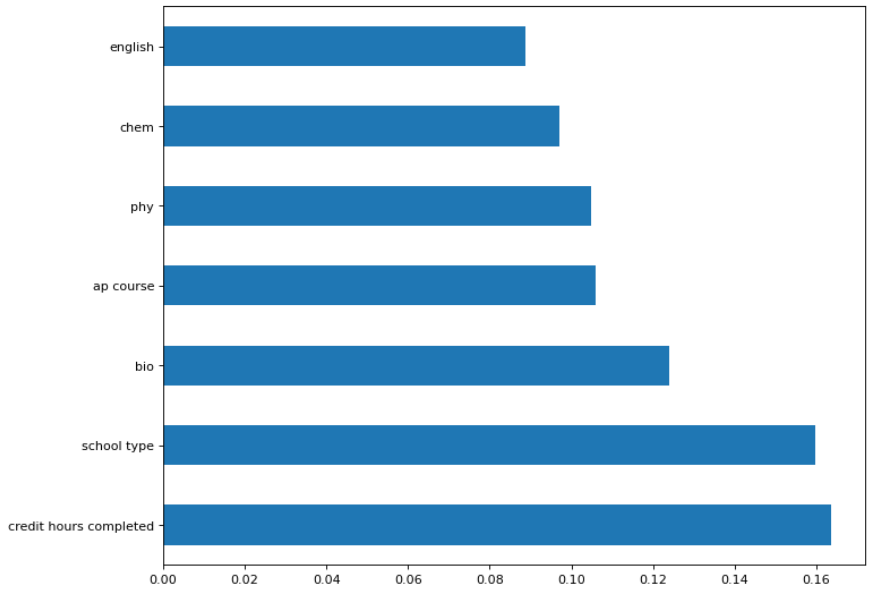


Figure 4.9: Feature importance on specialization label set. *Source*: Done by the researcher.

In terms of feature selection, the most impactful features based on a base random forest model are the credit hours completed, school type, as well as biology scores as shown in Figure 4.9. The correlation matrix is highlighted in Figure 4.10.

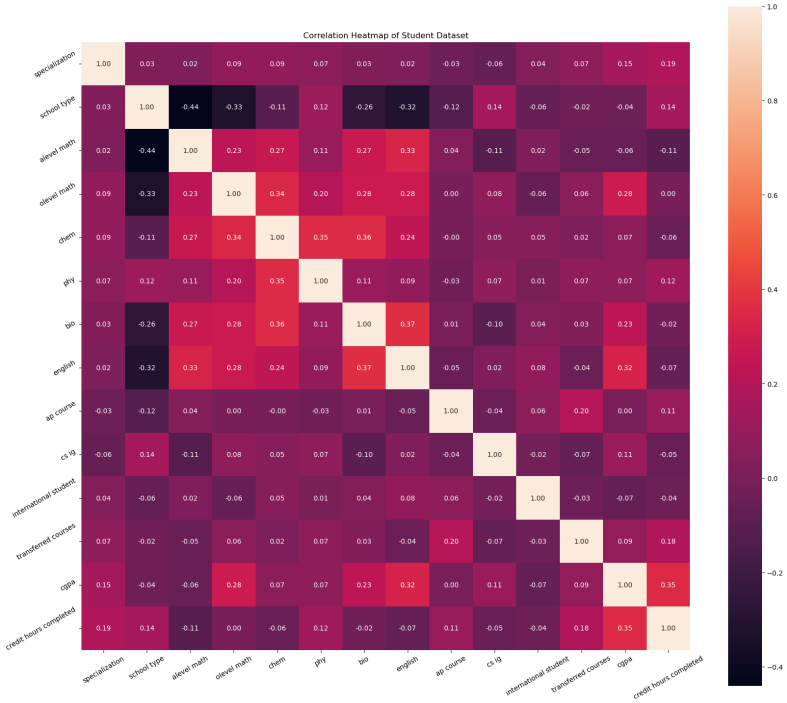


Figure 4.10: Correlation matrix of specialization label set. *Source*: Done by the researcher.

Figure 4.11 visualizes the explained and cumulative variance for the principal components to identify a potential possibility of reducing dimensionality without losing too much information.

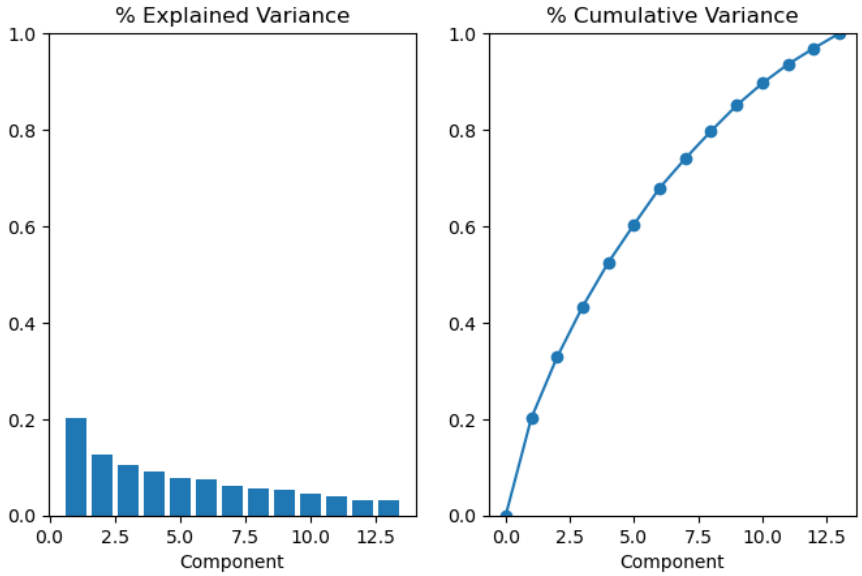


Figure 4.11: Explained and cumulative variance for features principal components. *Source*: Done by the researcher.

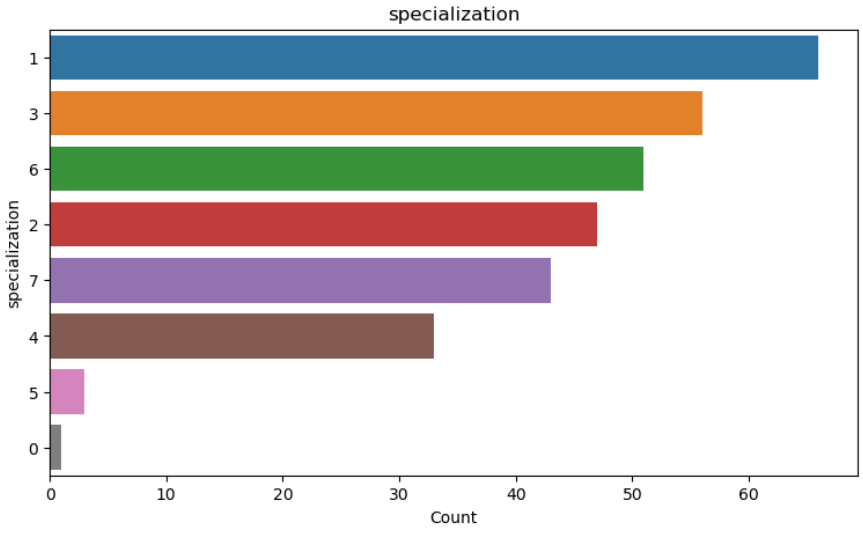


Figure 4.12: Bar plot for specialization. *Source*: Done by the researcher.

As shown in Figure 4.12, there is a severe imbalance in the target label. This is mitigated by first using a robust scaler and performing data augmentation using random oversampling to

increase the number of observations in the minority classes. The number of samples increased from 300 to 528 samples, with 66 samples per class.

Hyperparameter optimization is the following stage with Table 4.4 illustrating the optimized hyperparameters for the proposed random forest model in addition to the other classifiers.

Model	Hyperparameter	Value
Random Forest	Estimators	125
	Criterion	Gini
	Max Depth	15
	CCP Alpha	0.001
Decision Tree	Criterion	Entropy
	Max Depth	None
	CCP Alpha	0.05
Voting Classifiers	Voting	Soft
	Weights	(2,1,1)

Table 4.4: Hyperparameter optimization for various models on specialization label set.

Table 4.5 showcases the common classification metrics for the array of classifiers used. This was performed on a train-test split of 80-20% respectively with stratification on the label set.

Classifier	**F1**	**Precision**	**Recall**	**Accuracy**
Random Forest	0.71	0.72	0.71	0.71
Decision Tree	0.49	0.53	0.47	0.47
Gaussian Naive Bayes	0.49	0.62	0.43	0.43
Voting Classifier (RF, DT, Gaussian NB)	0.72	0.74	0.71	0.71
Voting Classifier (RF, Bernoulli NB, Gaussian NB)	0.66	0.68	0.66	0.66
Best Model (Voting Classifier - RF, DT, GNB)	**0.72**	**0.74**	**0.71**	**0.71**

Table 4.5: Comparison of classifiers performances on specialization label set.

4.3.3 Student Major Analysis

The same procedure was performed on the real-world dataset but with the student major as the target variable. The specialization, school id, as well as advanced math binary variable, were all removed for this segment of the thesis. Regarding any missing values, Figure 3.6 showcases the same missing values with the major as with specialization as the target. Missing values were filled with the median of each feature corresponding to grouping by the major. Concerning feature encoding, both the major and school type were converted to numerical variables. Table 4.6 illustrates the label-encoded values.

Original Label	Encoding
BUS (Business)	0
CS (Computer Science)	1
ENG (Engineering)	2

Table 4.6: Feature encoding of the major variable using label encoder.

Outlier detection found 19 potential outliers, of which 10 belong to the business class and the remaining to the other classes. This will be discussed further shortly. Concerning feature selection, the same three techniques were utilized once more. Figure 4.13 displays the feature importance relative to the major variable.

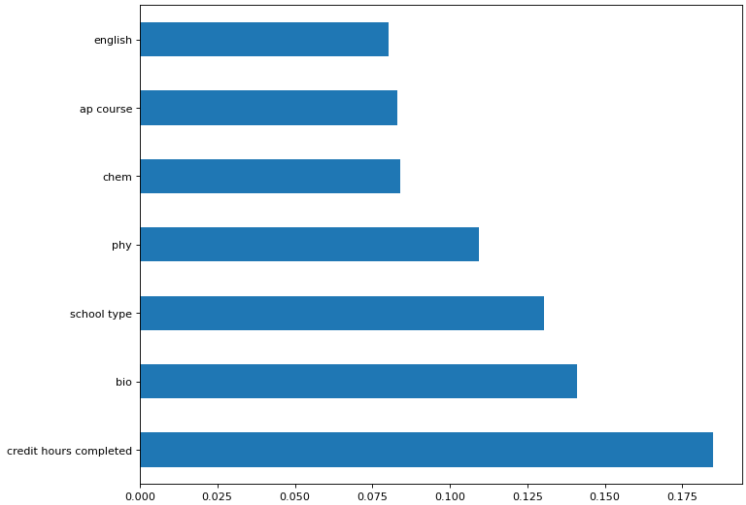


Figure 4.13: Feature importance for major label set. *Source*: Done by the researcher.

The correlation matrix is also displayed in Figure 4.14 with a variety of correlations between the features themselves, as well as a series of high and low correlations with the target variable. Figure 4.15 highlights the explained and cumulative variances by running PCA.

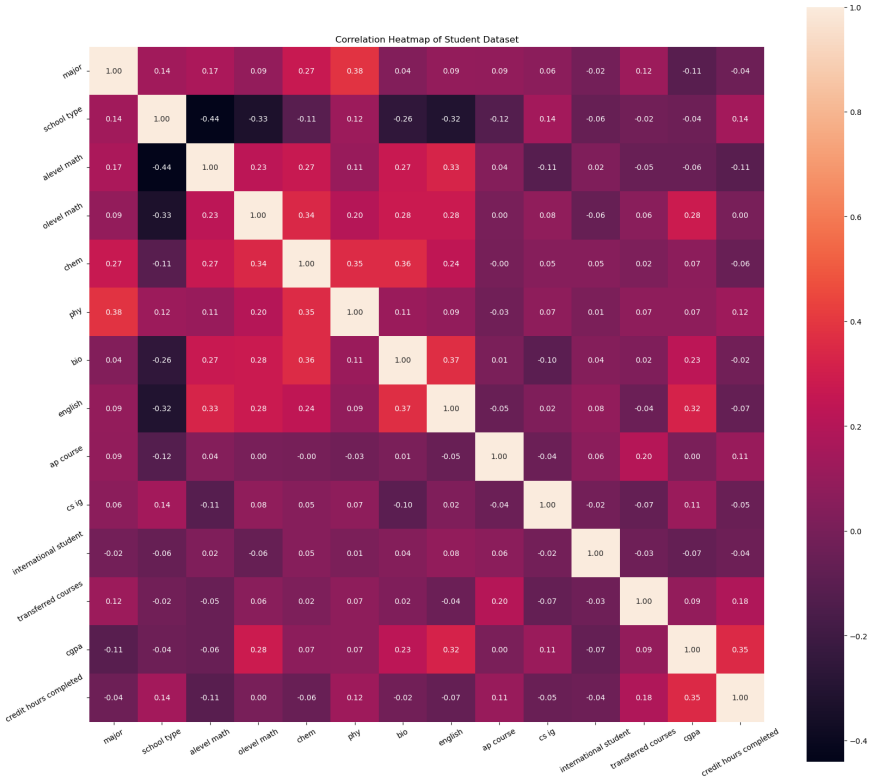


Figure 4.14: Correlation matrix for major label set. *Source*: Done by the researcher.

Random oversampling was again used for data augmentation as the business class had a severely greater sample size than the other two classes. This is supported by Figure 4.16. Performing the augmentation increased the data samples from 300 to 630, of which 210 belong to each respective class.

Hyperparameter optimization was performed with grid search cross-validation. The optimized parameters with the major as the label are displayed in Table 4.7.

Post-hyperparameter optimization, training, and testing were performed with Table 4.8 highlighting the performances of each model across a variety of classification metrics.

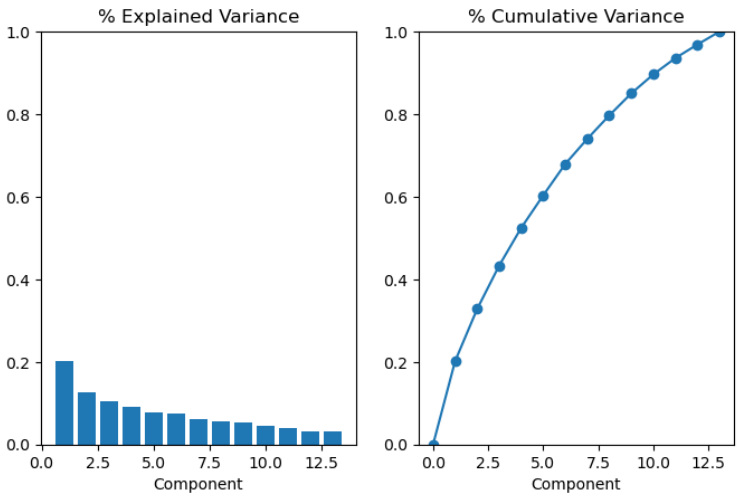


Figure 4.15: Explained and cumulative variance plots for major label set. *Source*: Done by the researcher.

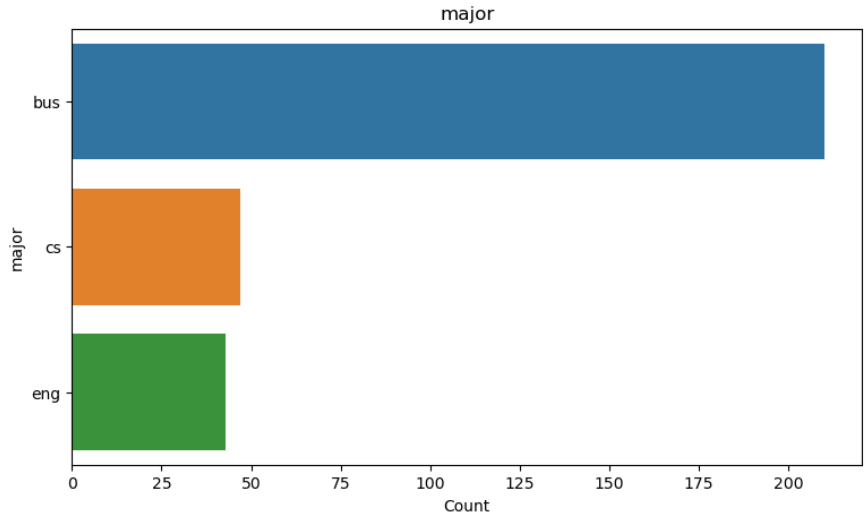


Figure 4.16: Bar plot for the major variable. *Source*: Done by the researcher.

4.4 Discussion

This section provides extensive information regarding the interpretability of the entire proposed pipeline ranging from the outlier analysis, feature selection procedures, classification results, as well as interpretability tools.

Model	Hyperparameter	Value
Random Forest	Estimators	175
	Criterion	Entropy
	Max Depth	20
	CCP Alpha	0.001
Decision Tree	Criterion	Entropy
	Max Depth	20
	CCP Alpha	0.001
Voting Classifiers	Voting	Soft
	Weights	(2,1,1)

Table 4.7: Hyperparameter optimization for various models on major label set.

Classifier	**F1**	**Precision**	**Recall**	**Accuracy**
Random Forest	0.96	0.97	0.96	0.96
Decision Tree	0.90	0.93	0.89	0.89
Gaussian Naive Bayes	0.58	0.67	0.57	0.57
Voting Classifier (RF, DT, Gaussian NB)	0.95	0.96	0.95	0.95
Voting Classifier (RF, Bernoulli NB, Gaussian NB)	0.95	0.95	0.94	0.94
Best Model (Random Forest)	**0.96**	**0.97**	**0.96**	**0.96**

Table 4.8: Comparison of classifiers performances on major label set.

4.4.1 Student Specialization

This subsection explains the results from the student dataset with the specialization as the label. Regarding feature importance using random forest, the most significant features for determining the specialization are the credit hours completed, school type, and biology scores as is evident in Figure 4.9. In realistic terms, the number of credit hours completed generally does not correlate with choosing a specialization. However, the school type such as whether a student was an 'IG' or 'American' student can have a significant impact on which specialization a prospective university student may choose. Numerical scores such as biology or physics tend don't tend to provide much information as many students (particularly in Egypt) tend to take a variety of similar core courses regardless of what they might pursue in the future. Hence, some of these courses could be throwaway courses that students took just to claim another course completed.

On the other hand, the correlation matrix in Figure 4.10 has slightly contradictory information to be shown. Among the highest correlations with the target variable are the credit hours completed

and cumulative GPA. However, GPA isn't considered an important feature for prediction. Moreover, there are a multitude of correlations between the numerical features, which is to be expected as most students taking for example biology would naturally pair it with another science course.

Performing PCA and the plots illustrated in Figure 4.11 supports the theory that all the features have some kind of information to provide, some more than others. Reducing the dimensionality would have come at the cost of severe information loss and removing variables, which may have an underlying impact on the specialization prediction such as cumulative GPA or advanced math.

Analysis of Outliers

With regards to outlier detection, comparing the p-value of each sample with an α of 95% produced 16 outliers, of which they were distributed among 4 of the 7 classes as shown in Table 4.9. As is evident in Table 4.9, the three outliers in BUS had much higher average a-level math scores than the global average in the dataset. Considering CS students, two of the five had transferred courses, which may have indicated that some were transfer students as well as all the outliers having an exceedingly high average o level math score than the rest of the students in CS.

As for MKTG and SDE, there is a similar trend in that all the outliers in both respective classes have significantly lower average grade scores in comparison to the average grades of the entire dataset. This is within reason as many students entering MKTG tend to have lower scores in high school and use it as a fallback. As for SDE, this suggests that students have low scores because many of their courses during high school were not picked towards planning for an engineering degree in sustainable design, which is considered a novel degree in Egypt.

Class (Label)	Number of Outliers	General Trend
BUS (1)	3	Much higher a level math scores
CS (2)	5	All high o level math scores than global average
MKTG (6)	5	Significant drop in average numeric scores
SDE (7)	3	All numeric features are significantly lower than global average

Table 4.9: Outlier detection & analysis of specialization label set.

Model Accuracy

In terms of model accuracies, Table 4.5 shows that the best-performing model was a soft voting classifier that combined a Random Forest, Decision Tree, and a Gaussian Naive Bayes model to perform the prediction. There could be a variety of reasons for this classifier to perform best. Firstly, optimizing the hyperparameter using grid search concluded that an unbalanced weight of (2,1,1) to the respective classifiers would provide the most effective performance. Secondly, Random Forest having the largest weight among the classifiers helped to avoid overfitting as well as giving the most resilient classifier the most impactful vote. Decision trees and GNB tend to perform poorly unless a set of underlying assumptions are met for each.

Decision trees tend to overfit with the increasing complexity of the data provided. Hence, the poor performance across the board. Similarly with Gaussian Naive Bayes where the model assumes that the input features all follow a Gaussian distribution. Failure to satisfy this key assumption is why the probability-based model performed so poorly. However, by utilizing the powerful nature of random forests alongside the aforementioned classifiers, a much more effective model was able to be created.

Local Model Interpretations

From an interpretability perspective, the random forest would also outperform any of the other classifiers as each estimator's tree structure would be at a much lower complexity than a single decision tree that is extremely deep, contains many impure nodes, and harms the ability to interpret local samples.

Figure 4.17 shows the tree structure of the decision tree model. Interestingly, the first splitting condition is based on the O-level math score of a student, which contradicts other interpretability measures that showed that O-level math wasn't among the top 3 features. Yet in the decision tree model, it is the root node split leading into either a split based on biology or physics scores. School type and chemistry are also the other two variables used further in the splitting process which suggests that the decision tree hasn't been able to utilize all the features at its disposal.

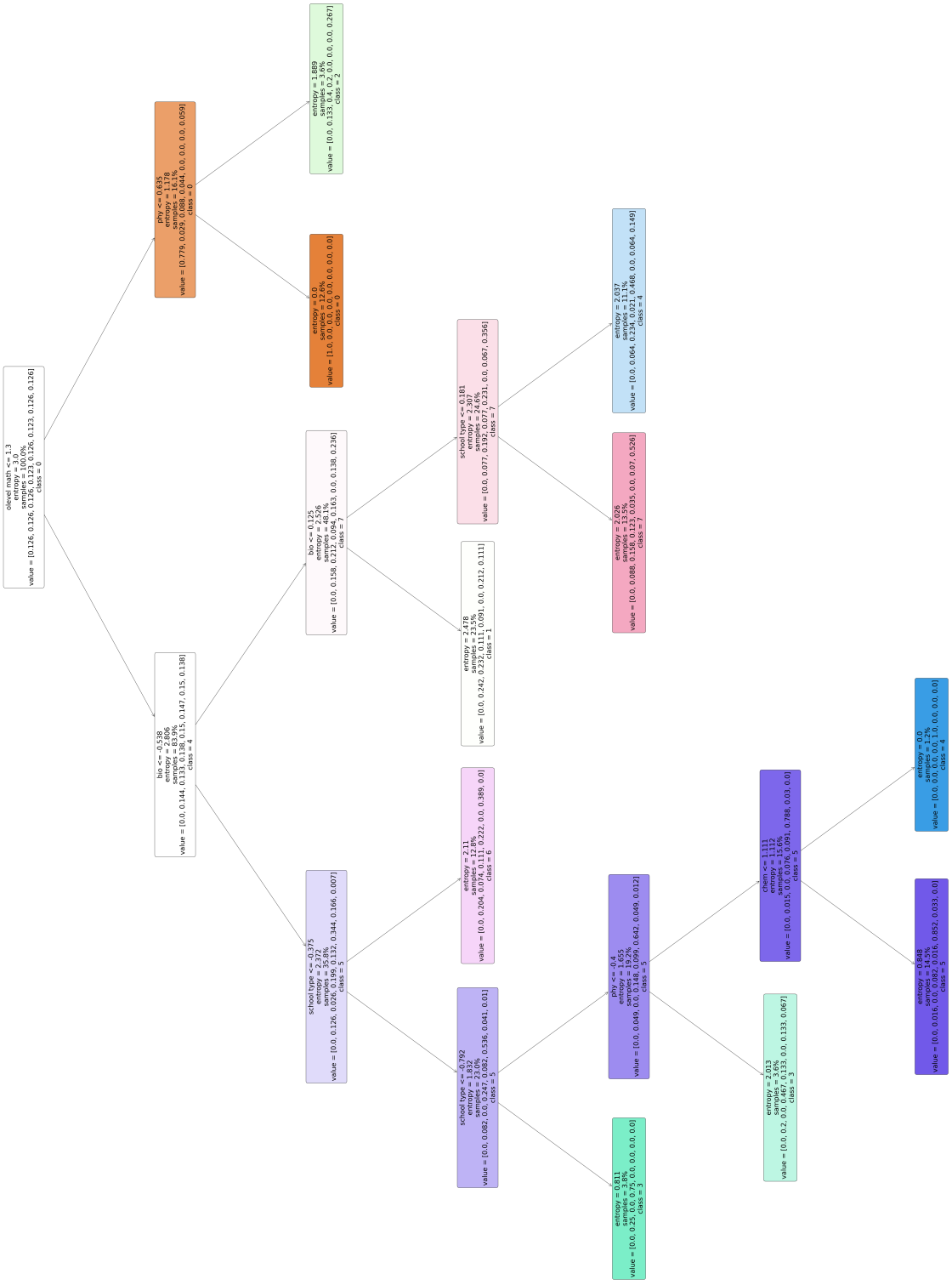


Figure 4.17: Structure of decision tree model. *Source*: Done by the researcher.

Analyzing the random forest feature important post-training, it is clear from Figure 4.18 that the same features before training were among the most important features again including credit hours completed, biology, and school type. Random forest is taking advantage of every feature at its disposal as it operates on multiple trees with different subsets of features to produce an average weighted result. Hence why the importance of the features is much more evenly spread as opposed to the decision tree, which utilizes only a small amount of features.

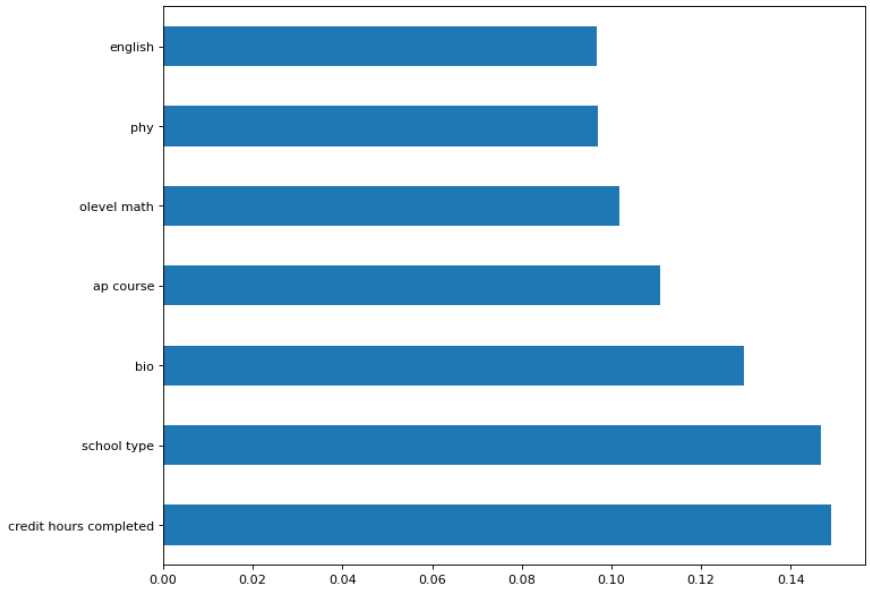


Figure 4.18: Feature importance of random forest after training and testing. *Source*: Done by the researcher.

Regarding Shapley values, they can provide strong local interpretations of how much each feature individually contributes to prediction confidence. Figure 4.19 represents the overall summary plot of the Shapley values for the entire training data consisting of 450 training samples. What is represented on the y-axis is the prediction confidence of each sample and it highlights how each feature contributes to the prediction in either a positive (red indicator) or negative (blue indicator) trend.

Diving further into the individual features, Figure 4.20 showcases how the school type numerically impacts the prediction outcome of each sample. The graph highlights how the school-type feature impacts the prediction accuracy of the respective sample. A trend is observed here with a

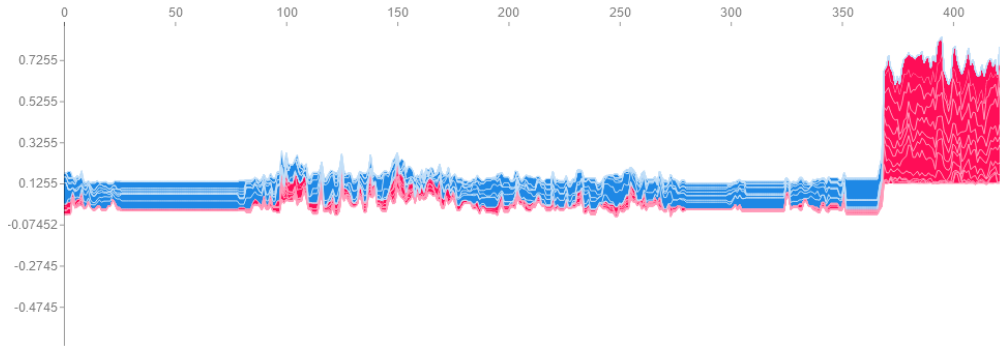


Figure 4.19: Shapley values for entire training data. *Source*: Done by the researcher.

wide majority of the training samples being impacted negatively by school type. This suggests that the school type affects the prediction confidence with a baseline value of 0.1255 with increasing or decreasing effects depending on the sample itself, with a wide variation in impact.

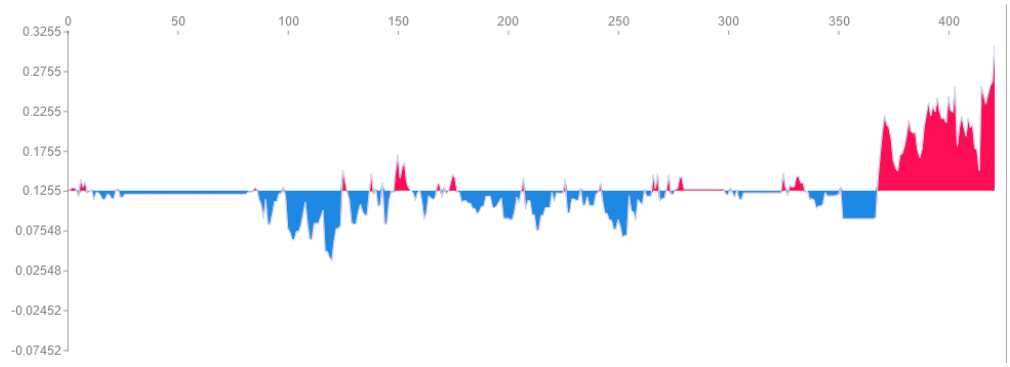


Figure 4.20: Shapley values for school type feature. *Source*: Done by the researcher.

In contrast to the school type, Figure 4.21 showcases the effect of the cumulative GPA on the prediction. A very unexpected trend is observed here as cumulative GPA was expected to potentially impact the choice of specialization. However, based on Figure 4.21, the cumulative GPA provides very little numerical influence on the prediction confidence. This could be due to the presence of other important features that end up overshadowing the GPA, or that the feature doesn't provide enough information to support strong predictions by itself.

Regarding plotting the Shapley values, Figure 4.22 provides a comprehensive overview of the biology feature and how it impacts each class in the label. It can be observed that the interaction of the biology feature differs based on the class label in both a positive, negative, or fluctuating trend

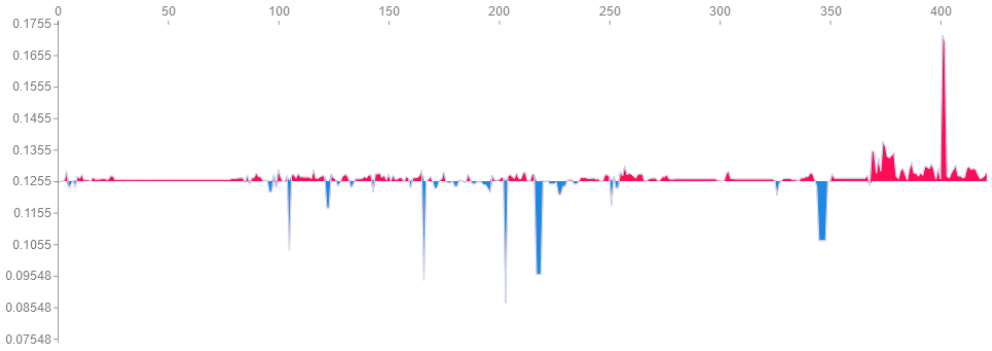


Figure 4.21: Shapley values for cumulative GPA feature. *Source*: Done by the researcher.

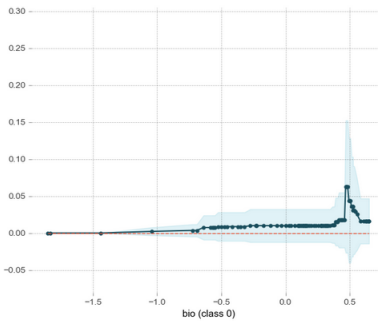
in Shapley values. However, sub-figure (g) in Figure 4.22 illustrates the average partial dependence plot of the Shapley values, indicating an overall positive trend in the prediction outcome when the influence of biology increases.

Contrasting the previous plots with a less important feature, Figure 4.23 showcases the partial dependence plots of the cumulative GPA. The trend rate for the GPA is far weaker than the biology feature for every class label, where they little to no numerical impact as evidenced by the y-axis values in the plots. Sub-figure (g) in Figure 4.23 also highlights the average influence of the GPA, which is almost negligible even with a high value of the feature.

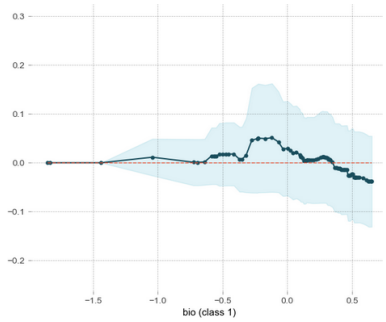
4.4.2 Student Major

Regarding the student major as the target variable, many different results were produced in comparison to the specialization as the target variable. From a feature importance perspective, a similar trend is observed with credit hours completed, biology, and the school type were the most important features with a base random forest model, which is evident in Figure 4.13.

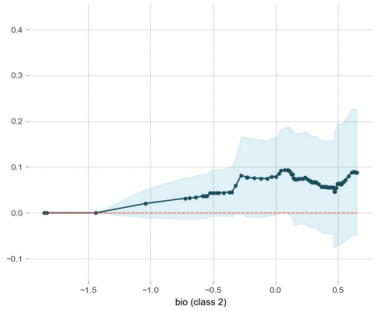
Cumulative GPA again seems to not provide much to the prediction outcome. This is supported by the correlation matrix of the major label set in Figure 4.14 which exhibits a negative correlation between the label and the cumulative GPA. This suggests GPA does not have a positive effect on the chosen major. In contrast, physics and chemistry scores seem to have the highest correlation with the label, which suggests a trend among a group of students who take those courses tend to lean towards STEM majors such as engineering or computer science as opposed to business.



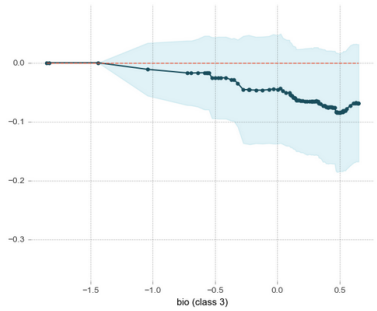
(a) Biology on class 0



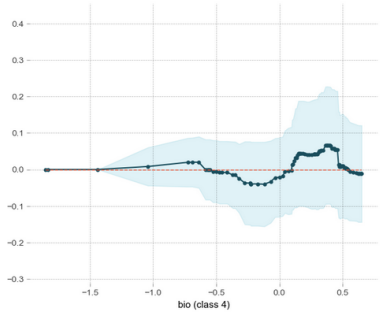
(b) Biology on class 1



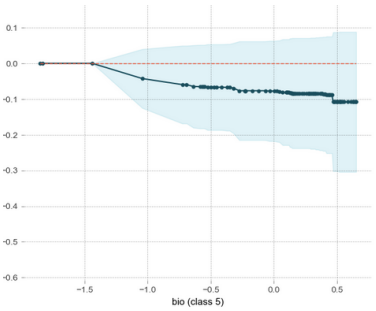
(c) Biology on class 2



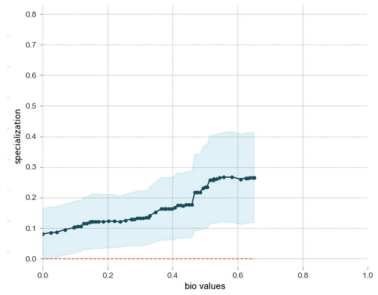
(d) Biology on class 3



(e) Biology on class 4

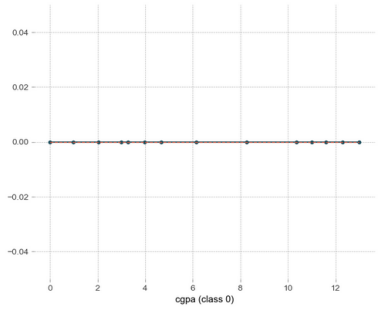


(f) Biology on class 5

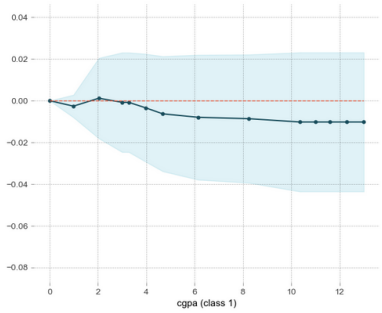


(g) Biology on average

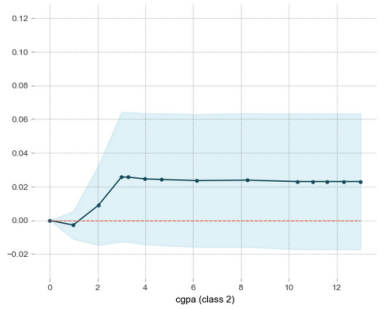
Figure 4.22: Partial dependence plots of shapley values for biology feature. *Source*: Done by the researcher.



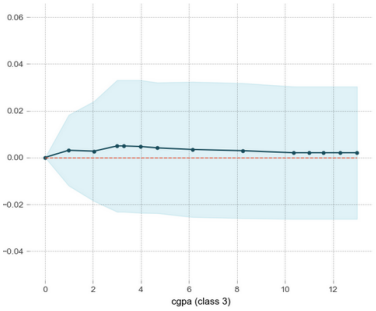
(a) Cumulative GPA on class 0



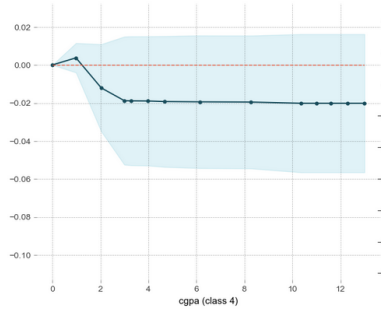
(b) Cumulative GPA on class 1



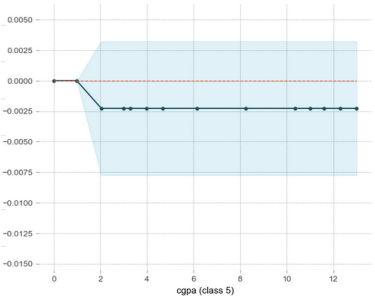
(c) Cumulative GPA on class 2



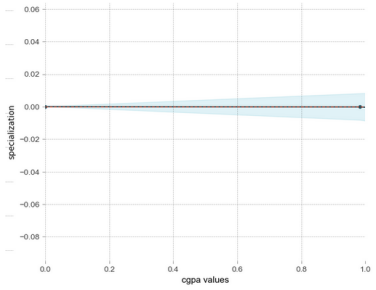
(d) Cumulative GPA on class 3



(e) Cumulative GPA on class 4



(f) Cumulative GPA on class 5



(g) Cumulative GPA on average

Figure 4.23: Partial dependence plots of shapley values for cumulative GPA feature. *Source*: Done by the researcher.

Another point worth mentioning is the sub-matrix of correlations between the numeric variables, which again suggests that certain courses taken by students are usually accompanied by a select group of other courses before choosing their respective major.

Concerning dimensionality reduction, Figure 4.15 shows how the explained variance is evenly distributed between the entirety of the dataset and the principal components. As a result, dimensionality reduction is not an option as loss of information will adversely affect the final results and potentially interpretation measures. This also solidifies the concept that many of these features individually do not provide much information but paired with a certain subset of features can enhance the performance of the proposed models significantly from an accuracy as well as interpretation perspective.

Analysis of Outliers

19 outliers were detected using the Mahalanobis distance, which is distributed among the 3 majors as shown in Table 4.10. Regarding the BUS outliers, more than half the outliers in this category have extremely inflated English scores relative to the rest of the dataset. Furthermore, some of their other numeric scores such as biology and physics are much higher than the global average. This suggests that some students who have entered BUS tend to have been students in high school who performed very well in STEM courses but opted to go business. This can inflate the average scores of the rest of the class samples.

For the CS outliers, the same 5 observations are the exact same outliers for both the major and specialization label set, with 3 having markedly higher English scores as well as O-level math scores. Engineering outliers have higher average A-level math scores than the global average in the Engineering field, while other numeric features lack scores in the below-average region. This supports the fact that students entering sustainable design engineering were admitted with lower-than-average scores although their math scores were satisfactory enough to accept into the major.

Class (Label)	Number of Outliers	General Trend
BUS (0)	10	Over half have extremely high English scores, inflated scores
CS (1)	5	3 of 5 have very high English scores
ENG (2)	4	Numeric features are significantly less than global average

Table 4.10: Outlier detection & analysis of the major label set.

Model Accuracy

The most effective model as supported by Table 4.8 is the lone random forest. Prediction accuracies are far superior with the major as the label in comparison to the specialization as the label. Reasons for the random forest as the best model correspond to many explanations. For both tree models, the error criterion was optimized to entropy, which tends to overfit with tree models while also distributing feature importance.

Random forest is generally resilient to overfitting the data in comparison to a decision tree. The voting classifiers nearly replicate the performance of the lone random forest, but with a greater weight assigned to the random forest, it would be safe to assume that the bulk of the strong performance is derivative of the presence of the ensemble model.

Local Model Interpretations

Regarding interpretability, random forest outperformed most of the evaluation metrics in the accuracy department. As the proposed model, it will be analyzed using the various measures discussed when the major is the label of the dataset.

Concerning the decision tree structure as another indicator of the decision-making process, the model splits the root node based on biology and expands to other numeric features like physics and chemistry. However, the tree is extremely complex and severely impacts the interpretability of how a decision is made for the major. Therefore, it is not a favorable measure to interpret the decision-making process.

As for feature importance, a pattern is emerging with the top 7 features distinctly being highlighted in Figure 4.24. This demonstrates that some features are more impactful than others in the decision-making process. This could be due to the little amount of information provided by other

variables such as the binary variables including whether they are an international student or have taken a computer science course in IG.

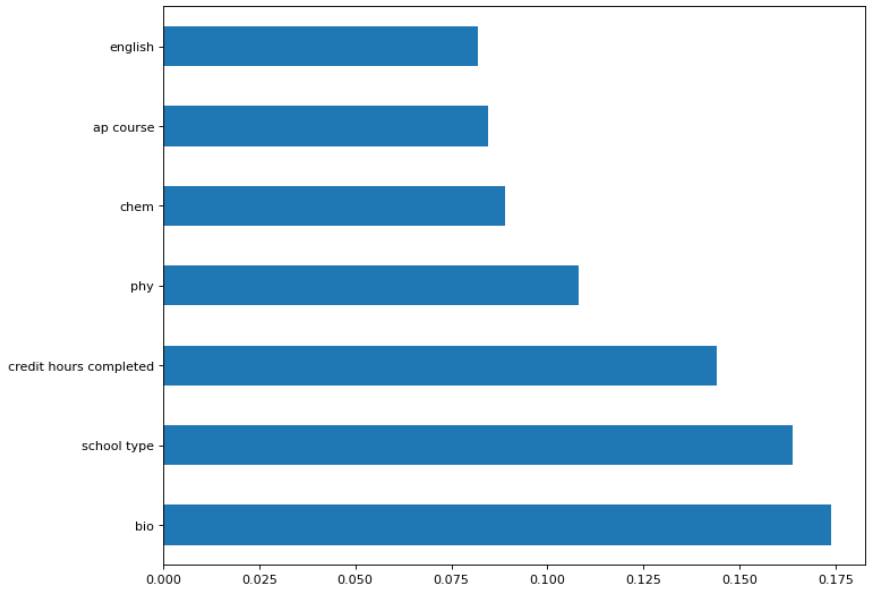


Figure 4.24: Feature importance of random forest after training and testing on major as label. *Source*: Done by the researcher.

Figure 4.25 shows the summary of the shapley values for the training data when major is the label. There seems to be a much higher range of confidence in the predictions with a series of features contributing to the outcome with varying effects. Most of the training data predictions are impacted negatively by various features and the most important features are explored.

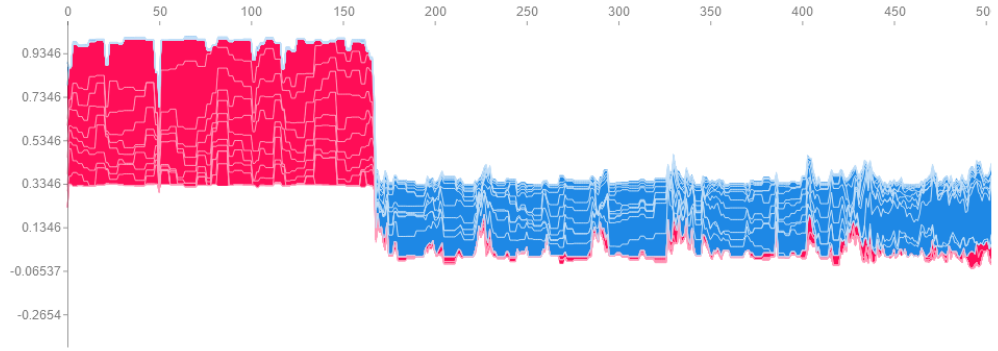


Figure 4.25: Shapley values for entire training data on major as label. *Source*: Done by the researcher.

One example is the number of credit hours completed as shown in Figure 4.26, which shows a fluctuating numerical contribution to the prediction outcome. The prediction confidence can be significantly influenced by this feature by up to 0.25 in either direction, which is a quarter of the prediction confidence.

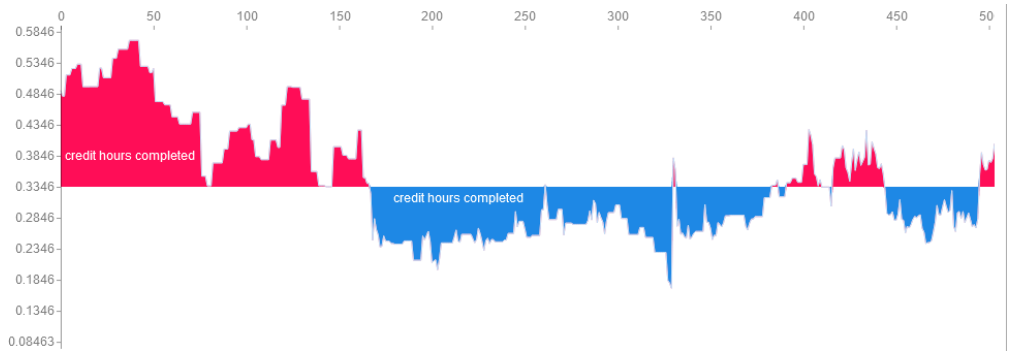


Figure 4.26: Shapley values for credit hours completed feature on major as label. *Source*: Done by the researcher.

This is contrasted by the Shapley value plot of a-level math in Figure 4.27. The plot shows that A-level math scores don't have quite as much impact as the number of credit hours. However, it

can still impact predictions by up to 0.18 in either a positive or negative direction depending on the training sample. Naturally, less important features will have more condensed shapely plots with a smaller numeric contribution to the label prediction.

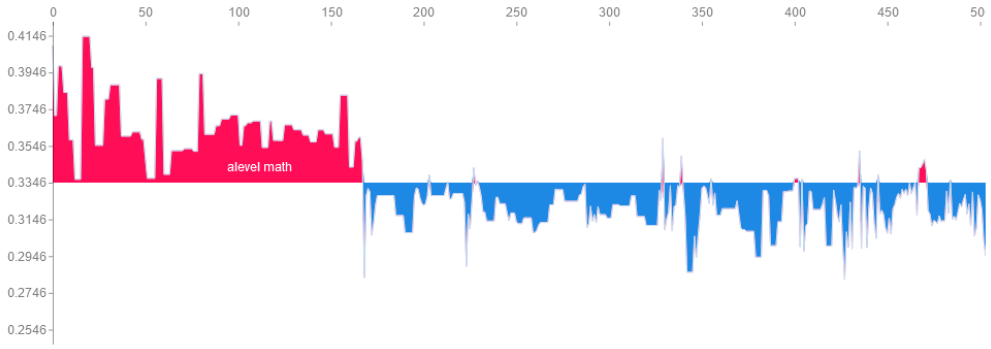
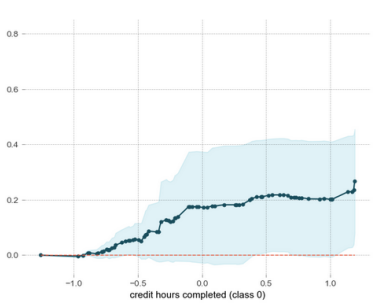
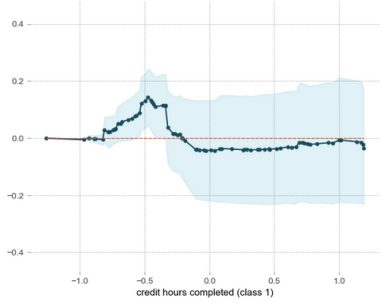


Figure 4.27: Shapley values for A-Level math feature on major as label. *Source*: Done by the researcher.

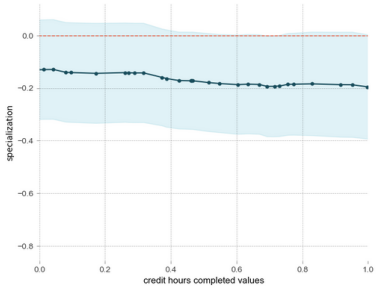
The last interpretability tool is the partial dependence plot of the shapley values for credit hours as illustrated in Figure 4.28. Credit hours completed seems to have an increasingly positive influence relative to classifying business majors, whereas for classifying computer science majors, the credit hours completed seems to increase until a turning point as shown in sub-figure (b) in Figure 4.28. However, the average effect of the feature is ultimately a negative contribution.



(a) Credit Hours Completed on class 0



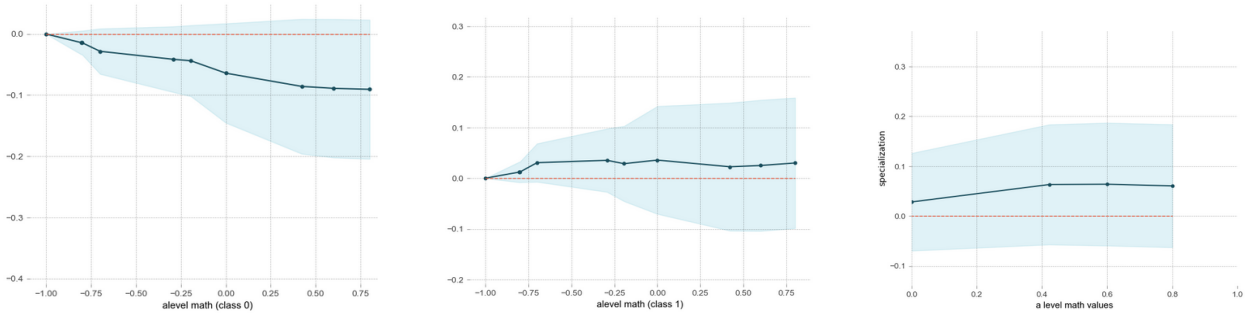
(b) Credit Hours Completed on class 1



(c) Credit Hours Completed on average

Figure 4.28: Partial dependence plots of shapley values for credit hours completed on major as label. *Source*: Done by the researcher.

As for A-level math, Figure 4.29 also highlights the impact based on the chosen major. However, the influence of the feature is far less than other more significant features in the dataset. Even though sub-figure (c) shows a generally positive trend in the training predictions, the depth of impact is almost negligible with an increase between 0 and 0.1.



(a) A-Level Math on class 0 **(b)** A-Level Math on class 1 **(c)** A-Level Math on average

Figure 4.29: Partial dependence plots of shapley values for a level math on major as label.
Source: Done by the researcher.

Chapter 5

Conclusion, Recommendations, and Future Work

This chapter highlights the key takeaways from the thesis research and evaluates the objectives mentioned at the start of the thesis, as well as improvements for future work.

5.1 Conclusion

The objective of this thesis was to create a viable and sophisticated student degree recommendation system. By going through the proposed pipeline, many relationships and insights were discovered including the detection of missing values, outlier detection as well as feature selection techniques. Understanding how each underlying feature impacts the target variable has been a focal point of this thesis as system interpretation was the key objective of this project.

In addition, the optimization of the hyperparameters, training, and testing of the supervised classifiers concluded that Random Forest was the best undisputed model across all datasets. Its performance in the synthetic dataset and the real dataset with both label sets outperformed all other proposed supervised classifiers. From the perspective of interpretability, the decision tree model was informative in providing the tree structure. However, it was far too deep and complex of a

structure to draw any meaningful conclusions regarding the decision-making process. Particularly for any local samples or future predictions.

Random Forest was once again the stand-out performer for interpretability as many insights were able to be extracted. Evaluation metrics highlighted the dominant performance of the Random Forest model in comparison to a single Decision Tree, the probability-based models, as well as the voting classifiers. Techniques including the feature importance, Shapley values, and the partial dependence plots of the Shapley values were integral to diving deeper into the decision-making process.

Key features such as the biology score, the school type, and number of credit hours completed were universally influential in providing a strong confidence of prediction. Contradictory to those features are the cumulative GPA, which had very marginal if not negligible effect on the predictions. The partial dependence plots visualizes the behavioral trend in shapley values when the values of our features change per class. Providing a more comprehensive report on impact of each feature on every possible target value.

Irrespective of the best model, recommendation systems continue to be an ever-growing field that generates interest and curiosity. A field that continues to encourage innovations for models and pipelines that extract analytical information at increasing depths of detail. Furthermore, with the growing popularity of artificial intelligence, how long will it be before AI tools are at the forefront for building state-of-the-art recommendation systems for all types of applications and frameworks?

5.2 Future Work

In terms of future work, there is plenty to be improved on this thesis. Regarding certain aspects of the proposed pipeline, a stronger emphasis needs to be placed on the feature selection stage. It is a vitally important aspect of the pipeline to simplify the problem at hand and improve the performances all around. In addition, utilizing some more interpretation techniques. Interpretation in this thesis was aimed at identifying underlying local interpretations between the features and their respective target variable. However, a point of improvement would be to approach interpretation

from a feature-by-feature basis to investigate any potential interactions between the features and how that might have a cascading effect on the final prediction.

Concerning any specific recommendations, it would be vital that this project be developed further with an emphasis on local interpretation tools similar to the ones mentioned in this thesis such as Shapley values. Furthermore, more analysis needs to be done on the influence of the outliers present as well as various other preprocessing steps. This could be vital to not only maximize the potential of this proposed model, but enhancing it further for local interpretability. Finally, an important recommendation is to attempt to commercialize this model by integration with a web application for students within the university (after the testing phase) to bring this idea to life and analyze any shortcomings or further improvements brought forth by the relevant stakeholders.

Bibliography

[Albert et al., 2022] Albert, C., Isgor, O., and Angst, U. (2022). Exploring machine learning to predict the pore solution composition of hardened cementitious systems. *Cement and Concrete Research*, 162:107001.

[Aljunid and Dh, 2020] Aljunid, M. F. and Dh, M. (2020). An efficient deep learning approach for collaborative filtering recommender system. *Procedia Computer Science*, 171:829–836.

[Barredo Arrieta et al., 2020] Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115.

[Bodily and Verbert, 2017] Bodily, R. and Verbert, K. (2017). Review of research on student-facing learning analytics dashboards and educational recommender systems. *IEEE Transactions on Learning Technologies*, 10(4):405–418.

[Caruana and Niculescu-Mizil, 2006] Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168.

[Chen et al., 2017] Chen, Y.-H., Tseng, C.-H., Huang, C.-L., Deng, L. Y., and Lee, W.-C. (2017). Recommendation system based on rule-space model of two-phase blue-red tree and optimized learning path with multimedia learning and cognitive assessment evaluation. *Multimedia Tools and Applications*, 76:18237–18264.

[El-Bishouty et al., 2019] El-Bishouty, M. M., Aldraiweesh, A., Alturki, U., Tortorella, R., Yang, J., Chang, T.-W., Graf, S., et al. (2019). Use of felder and silverman learning style model for online course design. *Educational Technology Research and Development*, 67(1):161–177.

[Elahi et al., 2016] Elahi, M., Ricci, F., and Rubens, N. (2016). A survey of active learning in collaborative filtering recommender systems. *Computer Science Review*, 20:29–50.

[Elshawi et al., 2019] Elshawi, R., Al-Mallah, M. H., and Sakr, S. (2019). On the interpretability of machine learning-based model for predicting hypertension. *BMC medical informatics and decision making*, 19(1):1–32.

[Erickson et al., 2021] Erickson, M., Elliott, S., Brown, C., Stackelberg, P., Ransom, K., Reddy, J., and Cravotta, C. (2021). Machine-learning predictions of high arsenic and high manganese at drinking water depths of the glacial aquifer system, northern continental united states. *Environmental Science Technology*, XXXX.

[Famili et al., 1997] Famili, A., Shen, W.-M., Weber, R., and Simoudis, E. (1997). Data preprocessing and intelligent data analysis. *Intelligent data analysis*, 1(1):3–23.

[Franke et al., 2012] Franke, T. M., Ho, T., and Christie, C. A. (2012). The chi-square test: Often used and more often misinterpreted. *American journal of evaluation*, 33(3):448–458.

[Guo et al., 2020] Guo, Q., Zhuang, F., Qin, C., Zhu, H., Xie, X., Xiong, H., and He, Q. (2020). A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568.

[Hussain et al., 2019] Hussain, M., Zhu, W., Zhang, W., Abidi, S. M. R., and Ali, S. (2019). Using machine learning to predict student difficulties from learning session data. *Artificial Intelligence Review*, 52:381–407.

[Izza et al., 2020] Izza, Y., Ignatiev, A., and Marques-Silva, J. (2020). On explaining decision trees. *arXiv preprint arXiv:2010.11034*.

[Jijo and Abdulazeez, 2021] Jijo, B. T. and Abdulazeez, A. M. (2021). Classification based on decision tree algorithm for machine learning. *evaluation*, 6:7.

[Karga and Satratzemi, 2018] Karga, S. and Satratzemi, M. (2018). A hybrid recommender system integrated into lams for learning designers. *Education and Information Technologies*, 23:1297–1329.

[Khanal et al., 2020] Khanal, S. S., Prasad, P., Alsadoon, A., and Maag, A. (2020). A systematic review: machine learning based recommendation systems for e-learning. *Education and Information Technologies*, 25:2635–2664.

[Kingsford and Salzberg, 2008] Kingsford, C. and Salzberg, S. L. (2008). What are decision trees? *Nature biotechnology*, 26(9):1011–1013.

[Koene et al., 2015] Koene, A., Perez, E., Carter, C., Statache, R., Adolphs, S., O'Malley, C., Rodden, T., and McAuley, D. (2015). Ethics of personalized information filtering. pages 123–132.

[Kouki et al., 2019] Kouki, P., Schaffer, J., Pujara, J., O'Donovan, J., and Getoor, L. (2019). Personalized explanations for hybrid recommender systems. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pages 379–390.

[Kurita, 2019] Kurita, T. (2019). Principal component analysis (pca). *Computer Vision: A Reference Guide*, pages 1–4.

[Li et al., 2017] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., and Liu, H. (2017). Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6):1–45.

[Liu, 2019] Liu, X. (2019). A collaborative filtering recommendation algorithm based on the influence sets of e-learning group's behavior. *Cluster Computing*, 22(Suppl 2):2823–2833.

[Lops et al., 2011] Lops, P., De Gemmis, M., and Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. *Recommender systems handbook*, pages 73–105.

[Mohammed et al., 2020] Mohammed, R., Rawashdeh, J., and Abdullah, M. (2020). Machine learning with oversampling and undersampling techniques: overview study and experimental results. In *2020 11th international conference on information and communication systems (ICICS)*, pages 243–248. IEEE.

[Molnar, 2022] Molnar, C. (2022). *Interpretable Machine Learning*. 2 edition.

[Priyam et al., 2013] Priyam, A., Gupta, R. K., Rathee, A., and Srivastava, S. K. (2013). Comparative analysis of decision tree classification algorithms.

[Raghuwanshi and Pateriya, 2019] Raghuwanshi, S. and Pateriya, R. (2019). *Recommendation Systems: Techniques, Challenges, Application, and Evaluation: SocProS 2017, Volume 2*, pages 151–164.

[Reader, 2021] Reader, T. C. (2021). Introduction to supervised machine learning.

[Saarela and Jauhiainen, 2021] Saarela, M. and Jauhiainen, S. (2021). Comparison of feature importance measures as explanations for classification models. *SN Applied Sciences*, 3:1–12.

[Shu et al., 2018] Shu, J., Shen, X., Liu, H., Yi, B., and Zhang, Z. (2018). A content-based recommendation algorithm for learning resources. *Multimedia Systems*, 24(2):163–173.

[Singh and Upadhyaya, 2012] Singh, K. and Upadhyaya, S. (2012). Outlier detection: applications and techniques. *International Journal of Computer Science Issues (IJCSI)*, 9(1):307.

[Singh et al., 2021] Singh, P., Dutta Pramanik, P., Dey, A., and Choudhury, P. (2021). Recommender systems: An overview, research trends, and future directions. *International Journal of Business and Systems Research*, 15:14–52.

[Thai-Nghe et al., 2010] Thai-Nghe, N., Drumond, L., Krohn-Grimberghe, A., and Schmidt-Thieme, L. (2010). Recommender system for predicting student performance. *Procedia Computer Science*, 1(2):2811–2819.

[Thorat et al., 2015] Thorat, P. B., Goudar, R. M., and Barve, S. (2015). Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4):31–36.

[Wan and Niu, 2018] Wan, S. and Niu, Z. (2018). An e-learning recommendation approach based on the self-organization of learning resource. *Knowledge-Based Systems*, 160:71–87.

[Wang et al., 2019] Wang, H., Zhao, M., Xie, X., Li, W., and Guo, M. (2019). Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*, pages 3307–3313.

[Webb et al., 2010] Webb, G. I., Keogh, E., and Miikkulainen, R. (2010). Naïve bayes. *Encyclopedia of machine learning*, 15:713–714.

[Zhang, 2016] Zhang, Z. (2016). Univariate description and bivariate statistical inference: the first step delving into data. *Annals of translational medicine*, 4(5).