# PREDICTING STUDENT DROPOUT IN HIGHER EDUCATION

Data Science Foundations CIS4047-N-BF1-2024

Jennifer Roberts, E5147249

## ABSTRACT

High dropout rates from higher education negatively impacts institutions, and efforts to provide highly-qualified workers for the economy. Predicting which students are most likely to drop out would allow universities to put extra support in place and hopefully prevent this outcome. Many dropout models use data from the first year, which limits their usefulness as half of dropouts occur during this period. This study created a model for a specific institution based only on data available at the time of enrolment, using a dataset from the Instituto Politécnico de Portalegre (Realinho *et al.* 2022) and a Support Vector Machine model. The best result came from using a radial kernel, with a balanced accuracy of 0.6936. The findings suggest that such a model is possible, but may require additional features to improve usefulness.

**Keywords:** student dropout, higher/tertiary education, educational data mining (EDM), machine learning, support vector machine (SVM), binary classification, prediction.

## TABLE OF CONTENTS

## TABLE OF FIGURES

# INTRODUCTION

Since the middle of the 20th century, the percentage of people attending higher education has increased significantly, and there are plans to increase it still further. The current goal of the European Union is that 45% of 25-34-year-olds will be degree-qualified by 2030. The aim is to create a highly-skilled workforce to ensure that Europe can compete in the global economy, as well as increasing societal and individual well-being (Council of the European Union, 2021).

Unfortunately, high dropout rates in higher education are a significant barrier to this (European Commission, 2015). The OECD (2022) found that, on average, 21% of full-time undergraduates dropped out by the expected end date of the course. Dropping out is not always a negative outcome for an individual student, but when it is it can negatively affect mental health. For higher-education institutions, high dropout rates can reduce funding, lower reputation and consume limited resources, as well as affecting the overall student experience.

Research into factors that influence dropout has been conducted for at least a century (Behr *et al*., 2020, cited in Kocsis and Molnár, 2024). If higher education institutions can predict which students are most likely to drop out, they can proactively offer extra support and hopefully prevent this. Over the past decade or so, attention has turned to machine learning models (ML models) to make these predictions.

It should be noted that the term "dropout" is not applied consistently (Agrusti, Bonavolontà and Mezzini, 2019). In some cases (including the dataset used in this paper), students who transfer to a different course and/or institution are considered to have dropped out, in others they aren't. Consequently, the dropout rates in this paper appear unusually high. In addition, dropout rates can be measured at different times (from after the first semester to years after planned graduation). In this dataset, student outcomes are measured at the expected end date of the course.

OECD (2022) figures suggest that half of dropouts occur in the first year, so a model relying on academic data from that time is of limited use. This study only considers information that can be known pre-enrolment. If an effective model could be created solely from this data, it would allow higher education institutions to identify at-risk students before they begin their courses, so that support can be in-place from day one.

In addition, this study only considers traditional machine-learning methods, namely Naive Bayes (NB), Logistic Regression (LR), Decision Trees (DT), K Nearest Neighbours (KNN) and Support Vector Machines (SVM).

# LITERATURE REVIEW

## Broad Overview

The list of factors that may affect dropout is extensive, although the main categories are considered to be Academic, Psychological, Demographic and Environmental (Alyahan & Düştegör, 2020, cited in Kocsis and Molnár, 2024). Given that the factors affecting dropout rates, and their relative importance, vary across different institutions and countries, and over time due to changing cultural, political and economic conditions, a one-size-fits-all model of student dropout is unlikely to be achieved (Kocsis and Molnár, 2024). However, models for a specific institution have shown more promise.

Much research has been done into using machine learning for dropout prediction, but no consensus has emerged on the best model. Recent systematic reviews (Kocsis and Molnár, 2024; Agrusti, Bonavolontà and Mezzini, 2019; Albreiki, Zaki and Alashwal, 2021) illustrate the wide range of approaches, but do not agree on the most common choice. For Kocsis and Molnár (2024), who studied academic performance more broadly, it was NN. For those who studied dropout exclusively, Agrusti, Bonavolontà and Mezzini (2019) found it was decision trees (including random forests), while Albreiki, Zaki and Alashwal (2021) grouped all traditional machine learning methods.

Of course, most common doesn't mean most effective. Every study in Kocsis and Molnár (2024) that used more than one model favoured a different one. The other studies did not consider success rates.

Kocsis and Molnár (2024) identified several difficulties with comparing studies. These include:

1. Differing definitions of dropout (see introduction).
2. Different variables and/or measurements (most use pre-existing data, so are restricted by availability).
3. Most only tried one algorithm (87 of their 95).
4. Inconsistent evaluation measures.

## Studies Using This Dataset

The dataset in this study (Kaggle, 2024) ("the original dataset") was created by researchers at the Instituto Politécnico de Portalegre in Portugal. The first version was introduced in Martins *et al.* (2021), and it was then edited and detailed further (by the same authors) in Realinho *et al.* (2022) ("the original authors"). It was compiled from several different sources and covers 17 undergraduate degrees at that institution between 2008/2009 and 2018/2019. There is a similar dataset available on UCI (2021) linked to Martins *et al.* (2021). Small differences in some papers

suggest there may be other versions. The version linked to the original authors is now restricted, although the Kaggle version names that as its source.

The full dataset contains 4424 records of 35 attributes, reduced from 12,992 of 398. Many of these were duplicates, and some will have contained identifying information. However, they also removed records from discontinued courses. They do not state how many were deleted, but this data is potentially valuable. In addition, the version from Martins *et al.* (2021) contains data on the students' previous grades, which could have been useful for the model. This may have been for privacy reasons, or due to lots of missing information. There is also no information on what, or how many, missing values or outliers may have been imputed, or how.

Given the different versions, and the fact that this study does not consider all the variables in the original dataset, some of the previous work on feature importance does not apply. Even allowing for that, it is inconclusive.

In their paper, the original authors looked at feature relationships and importance. They used the Pearson Correlation Coefficient (PCC) to identify the highest correlations (among variables in this study) as between Nationality and International, and Mother Occupation and Father Occupation. However, given that the PCC is designed for continuous variables and all of these (and many others in the original dataset) are non-ordinal categorical variables, these results may be misleading (Brownlee, 2020).

To test feature importance, the original authors chose Permutation Feature Importance (PFI). This method is used in conjunction with an ML model. It takes each feature in turn, shuffles the values randomly, and then runs the model to see what effect that has on the score (F1 was used here). PFI is a useful tool, but has two major weaknesses. Firstly, it is adversely affected by multicollinearity. Secondly, it identifies the features most important for the model used to calculate it, not necessarily in general (Scikit-Learn, no date). The original authors looked at the common results of four models, but all were ensemble and decision-tree-based, and no evaluation metrics were provided, so the results' applicability to different models cannot be guaranteed.

Singh & Karthikeyan (2024) combined Ant Colony Optimisation with RF, NN, LR, KNN and SVM. Their best feature set (0.9012 accuracy with RF) had 11 features. Their second-best (0.8962, NN) had seven. Only five appeared in both sets. The third best (0.8961, RF) had nine – two of which didn't appear in either of the previous two sets (unfortunately, it's not clear what these features are, due to inconsistent numbering and ordering). Other studies have used different feature selection methods and have come up with different results. Only one feature from Singh & Karthikeyan's (2024) best set (Course) was in the top ten for all methods used by the original authors. Seven didn't appear at all.

The data is divided into three possible outcomes at the expected end date of the course: Graduate, Dropout (including transfers to other courses/institutions) and (still) Enrolled. The original authors took the view that the Enrolled class were a distinct group of students who might have different needs from those in the Dropout class. (It would be interesting to read a follow-up study that tests this assumption). Some researchers have followed their recommendation to treat this as a multi-class problem, but others have taken a different approach.

## Studies Dropping the Enrolled Class

In these studies, the Enrolled class is regarded as having an uncertain outcome, as students will ultimately either dropout or graduate. All instances in this class are dropped and the problem becomes a binary classification task.

Table 1 is a good illustration of the comparison difficulties identified by Kocsis and Molnár (2024) (see above). All four papers use the same dataset, but favour different models. The highest scores came from balancing and feature selection, but so did the worst. The second highest used no balancing or feature selection at all. All the studies used multiple models, but not the same ones. No single evaluation metric was used by all four. It should also be noted that none of these studies had been peer-reviewed at the time of access (two were conference papers and two pre-prints). The best model is far from clear.

This study will also take the approach of dropping the Enrolled class. However, as it does not use all the variables from the dataset, results are likely to turn out different again.

*Table 1: Studies dropping the Enrolled class*
*Only the traditional ML models used have been included*

| **Paper** | **Notes** | **Results** |
|---|---|---|
| Domínguez-Gómez et al. (2024) | Balanced with oversampling. 10+ features selected with five methods, 10-fold CV. | **1. KNN - 0.99 F1**<br>2. DT - 0.95 F1<br>3. SVM - 0.94 F1 |
| Gupta et al. (2024) | Unbalanced. 11 features selected. | **1. LR - 0.91 F1, 0.93 AUC**<br>2. SVM - 0.90 F1, 0.93 AUC<br>3. DT - 0.87 F1, 0.84 AUC<br>4. NB - 0.74 F1, 0.50 AUC |
| Kim, Yoo & Kim (2023) | Unbalanced. No feature selection. | **1. SVM - 0.95 AUC**<br>2. KNN - 0.92 AUC<br>3. DT - 0.91 AUC |
| Singh and Karthikeyan (2024) | Balanced with oversampling. 11 features selected. | **1. DT - 0.87 F1, 0.87 AUC**<br>2. LR - 0.86 F1, 0.87 AUC<br>3. KNN - 0.85 F1, 0.85 AUC |

# TECHNICAL IMPLEMENTATION

## Pre-processing

A cleaning process had already been performed by the original authors. The dataset was confirmed to have no missing values (NAs or empty strings), no duplicate rows or columns, and no single-value columns. No outliers were found.

As this study only considered pre-enrolment factors, features not related to this (13 columns) were deleted, along with Application Order as no information on the category meanings was included by the original authors (see Appendix A). After dropping the Enrolled class, 3630 instances of 19 attributes were left, with a 61% Graduate / 39% Dropout split.

## Re-Categorization of Variables

Most of the features are categorical, however examining the different categories raised some concerns.

First, Previous Qualification, Mother/Father Qualification and Mother/Father Occupation all had duplicate/overlapping categories (see Appendix B). This is likely the result of multiple databases being combined into one dataset and/or using free-text inputs. Also, the levels of Previous Qualification and Mother/Father Qualification have a natural order, but were categorised seemingly at random.

The second issue is category size. Some of the features have high cardinality (Mother/Father Occupation has 46 categories) and many of these categories have only a small number of instances (e.g. in Father Occupation, 31 have less than 10). With such a small sample size, we cannot distinguish pattern from noise and the risk of over-fitting the model is high (Sangani, 2021).

New variables Previous Education, Mother Education and Father Education were created from these, and all have the same three categories. Five categories were tried, but showed two pairs with little difference between them.

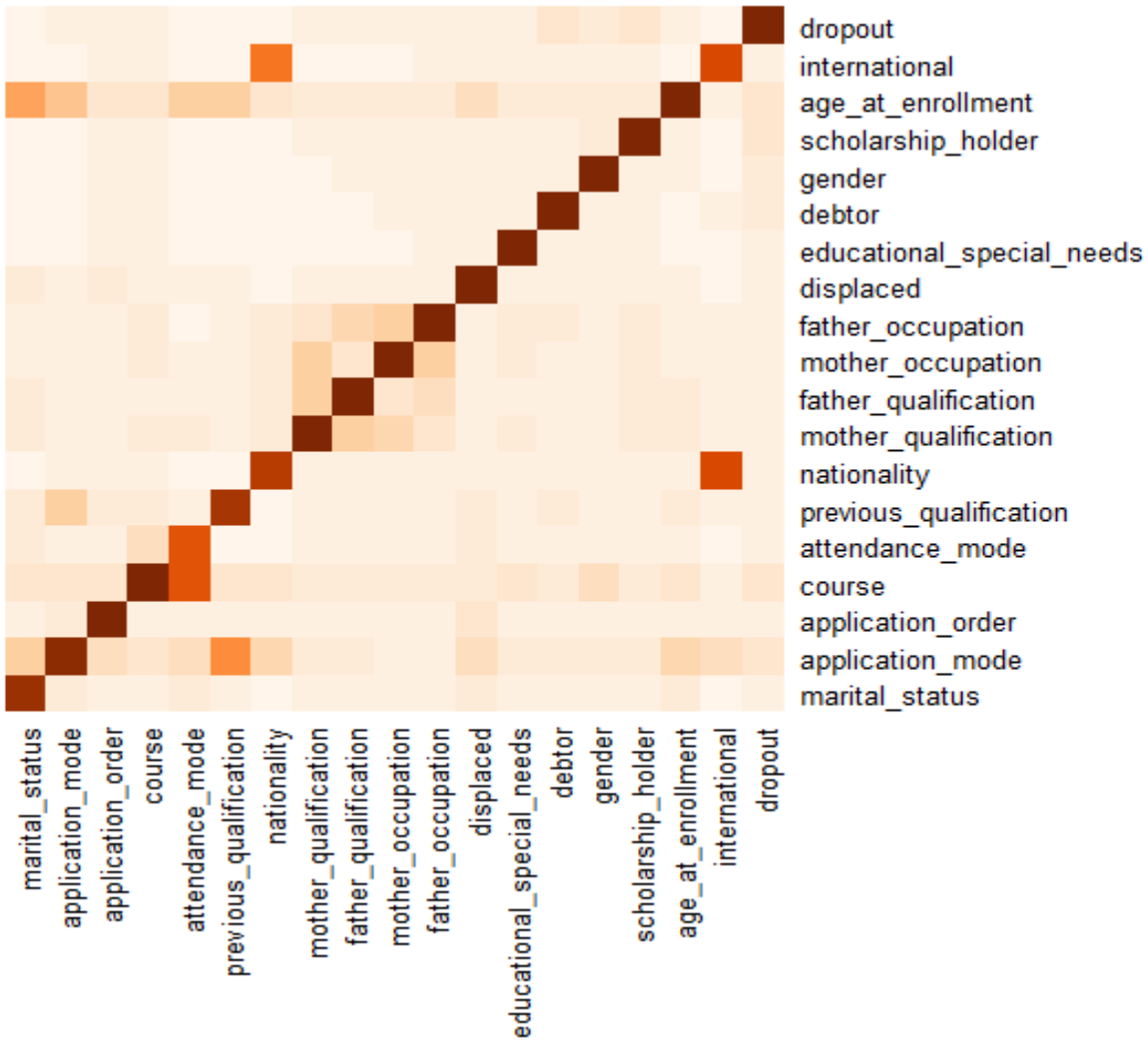
There were six categories in Marital Status, four of which had few instances. The largest category was Single, so one option was to divide them into Single and Other. However, whether someone is living with a partner seemed more likely to be relevant than their legal status. Therefore, the variable was divided functionally, between those who are (whether married or not) and those who aren't.

Appendix B contains full details of how categories were combined, along with other variables.

## Feature Relationships

Mutual information (MI) scores were used to find feature relationships, as it suits all types of variables and both linear and non-linear relationships. Figure 1 shows the most significant.

*Figure 1: Mutual information heatmap*




*Nationality and International*

98% of instances are home students with Portuguese nationality. The rest are international students, split between various small Nationality categories (see above). Therefore, Nationality will not be used in the model.

*Course and Attendance Mode*

There are two courses that are run both in the daytime and the evening. The two attendance modes have separate categories in the Course variable. There are no evening students except on

the marked evening courses. Therefore, Attendance Mode will be excluded as it adds no new information.

*Age at Enrollment and Marital Status/ Single*

As Single did not show a significant relationship with Dropout, it was not used in the model.

*Application Mode and Previous Qualification*

As well as the 'general competition' for school leavers, the Portuguese system has different application modes for holders of specific qualifications, international students, students over 23 without a secondary qualification and transfer students (Eurydice, 2024). Consequently, Application Mode overlaps with Age at Enrollment, Previous Qualification and International, and will be excluded from this study. The only unique information in Application Mode is whether the students are transferring from another course/institution, which was captured in a new Transfer variable.

*Parents*

To get the most information in the fewest variables, it was decided to reduce the number of parent variables from four to two. New features were created for the education/profession of the student's same-sex and opposite-sex parents, and the higher education level of both parents. For education, the highest MI score came from the same-gender parent. For profession, it was the opposite.

*Course and Gender*

Previous research has suggested that gender is a predictor of academic performance, including dropout, with males more likely to drop out than females (Kocsis and Molnár, 2024; Casanova *et al*., 2018; OECD, 2022). The heatmap does not identify any relationship between Course and Gender. However, the gender breakdown for different courses tells a different story (Figure 2).

The percentage of males and percentage of dropouts on each course were calculated, and they have a strong positive correlation (0.7). However, there are variations across courses independent of gender (Figure 3). The biggest outlier is Basic Education, which is 96% female but has a dropout rate of 60%. Given that previous work has found gender to be a strong predictor of dropout, the decision was made to leave both variables in the model.

*Figure 2: Bar chart of gender balance for each course.*

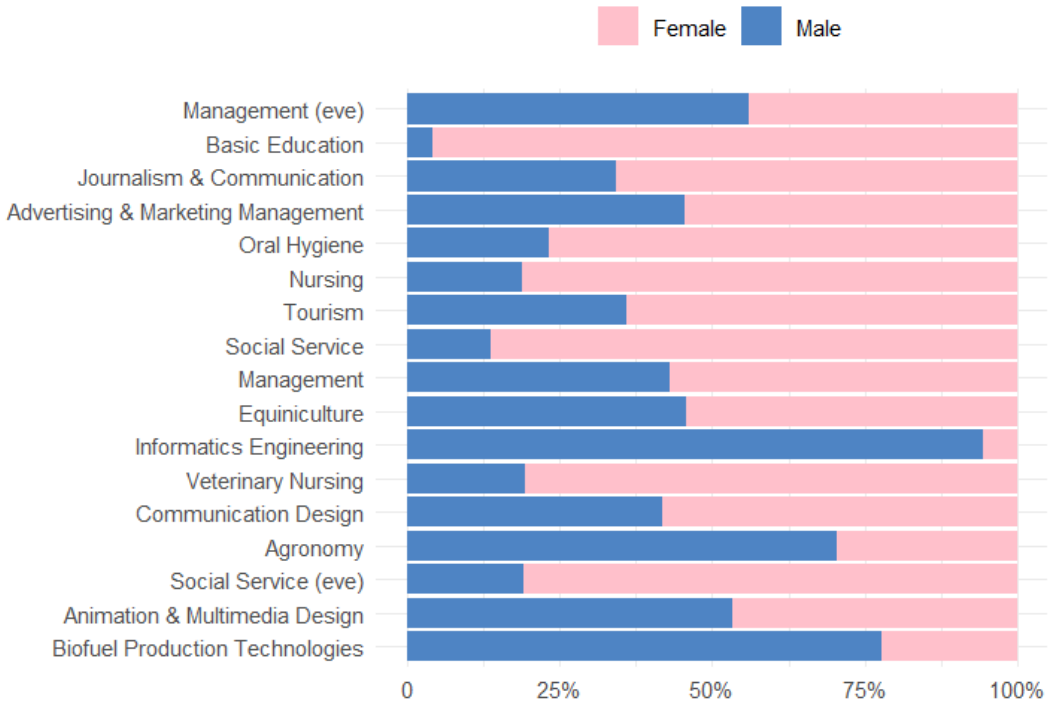
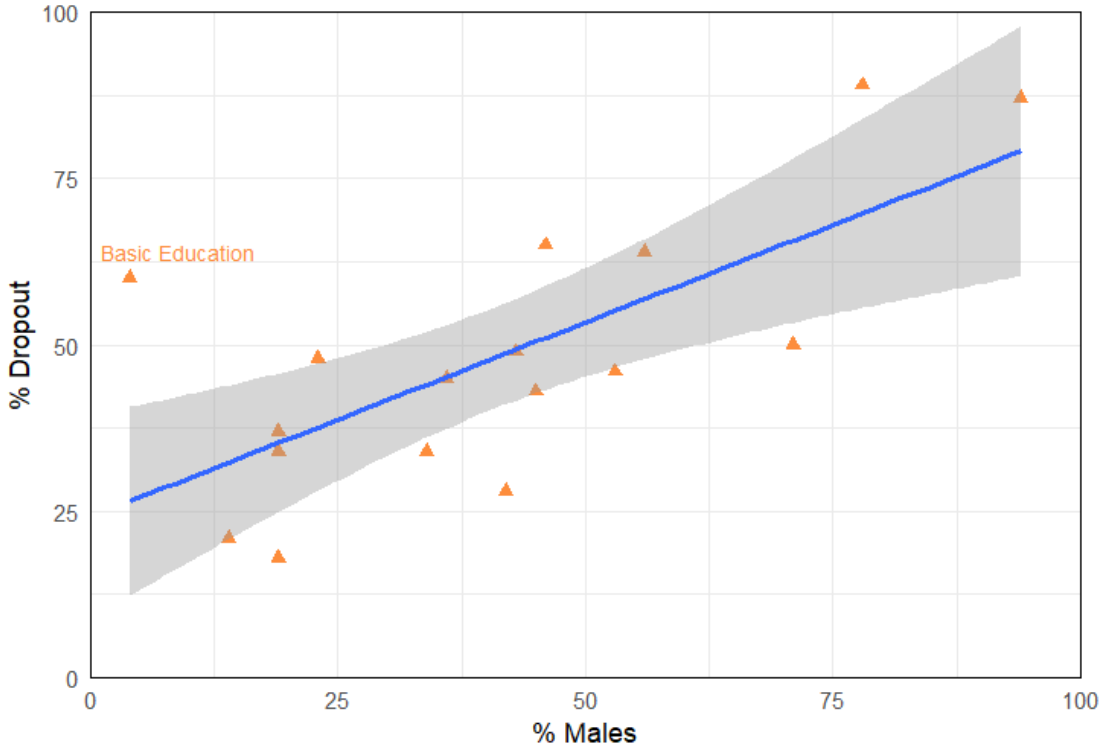



*Figure 3: Scatter plot of percentage of males vs percentage of dropouts by course*



## Feature Effects on Dropout Rates

*Figure 4: Bar charts of effect on dropout - binary variables*

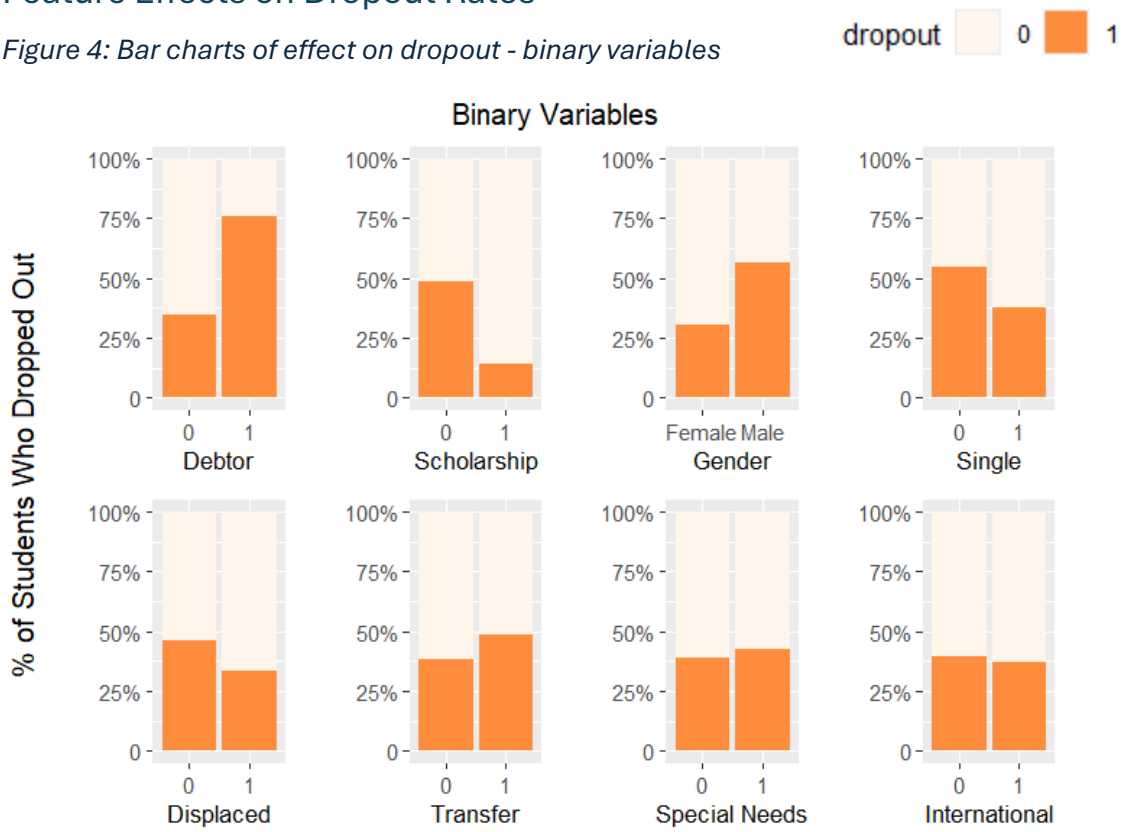
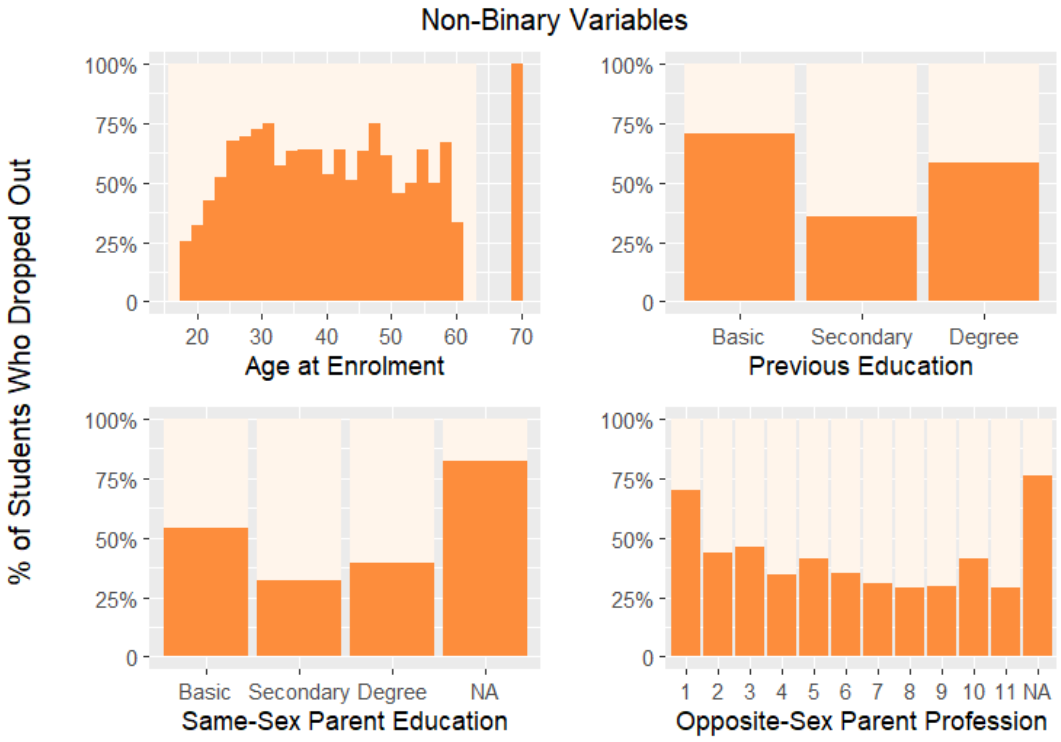



*Figure 5: Effect on dropout - non-binary variables*



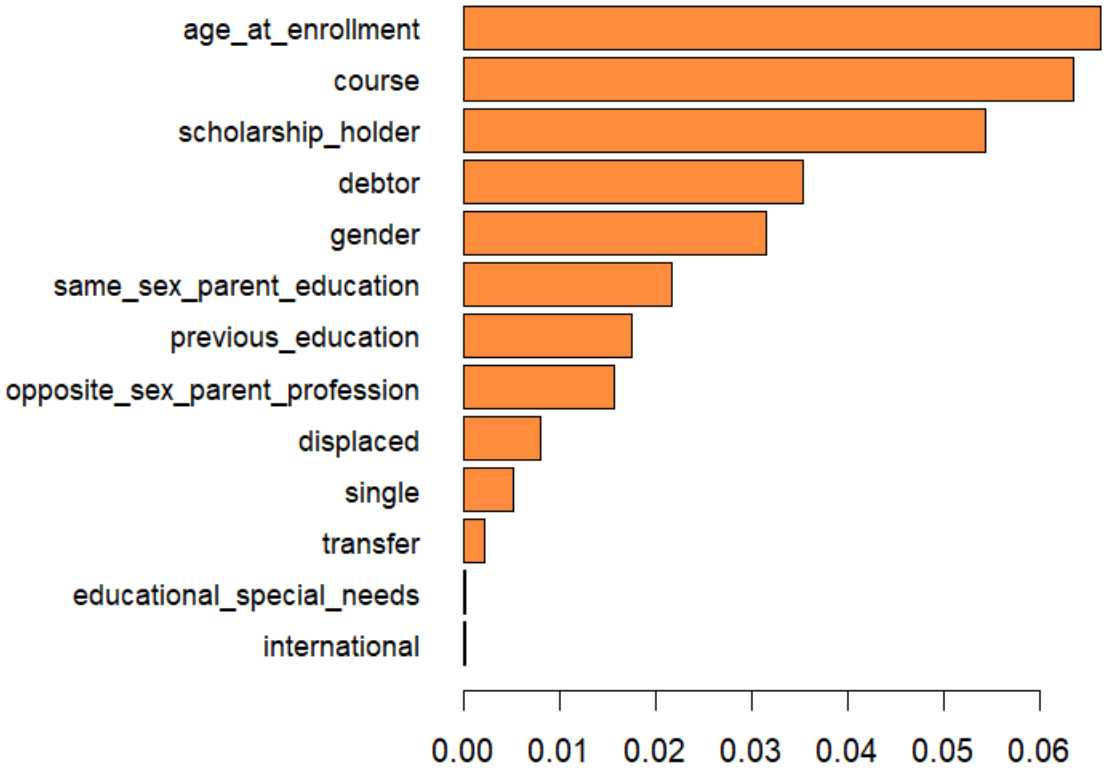
The binary variables (Figure 4) are shown roughly in order of the difference they make to dropout rates between the two categories. Non-Binary variables (Figure 5) are shown before imputation of missing values.

Age at Enrollment shows an upward trend until about age 25, but then no clear pattern. Attempts were made to transform this variable, but didn't achieve a better predictor. All ages were left in the model, but given the low number of students over 60 removing those instances would be an option.

Both Previous Education and Same-Sex Parent Education show the same trend; dropout rates are highest among those who didn't complete secondary and lowest among those who did, but do not have a degree.

Interestingly, the parent profession most likely to produce dropouts was 'Student'.

*Figure 6: Bar chart of mutual information scores with Dropout*




A Chi-Square Test of Independence was also performed on the categorical variables. This confirmed there is no significant relationship between Educational Special Needs or International, and Dropout. All other variables tested as related to Dropout.

## Feature Selection

The features selected for the model were: Age at Enrollment, Course, Scholarship Holder, Debtor, Gender, Same Sex Parent Education, Previous Education, Opposite Sex Parent Profession and Displaced.

## Model Selection

The dataset is structured and labelled, suitable for a supervised classification algorithm. As stated above, only traditional MI models were considered. Some algorithms require more computing time and power than others, but this dataset was sufficiently small for this not to matter.

The dataset is unbalanced, but the imbalance is mild (Google, 2024) with 39% in the minority class. Previous studies using the unbalanced data (Kim, Yoo & Kim, 2023; Gupta *et al.*, 2024) had similar results to those using balancing methods (Domínguez-Gómez *et al.*, 2024; Singh and Karthikeyan, 2024). Therefore, this did not affect model choice either.

The boundary between classes in classification can be either linear or non-linear. LR and NB only model linear ones, KNN and DT only non-linear, while SVM can do both (using different kernels). Given that the nature of the decision boundary here was unknown, a model was needed that could handle both types, and SVM performed well in the three similar works that used it.

## Data Preparation

Data preparation and modelling was done using the caret library in R Studio.

The data was left unbalanced. It was split 80:20 between the training and testing sets. Caret uses a stratified split that produced the same ratio of Graduate to Dropout in both sets. Imputation of missing values was then performed in each set separately.

For optimum performance, scaling was done inside the model training. Caret then scales the data within each cross-evaluation fold (10 were used), preventing data leakage across folds.

## Parameter Tuning

Each model was run with 20 randomly-selected pairs of parameters using Caret's inbuilt function.

# PERFORMANCE EVALUATION

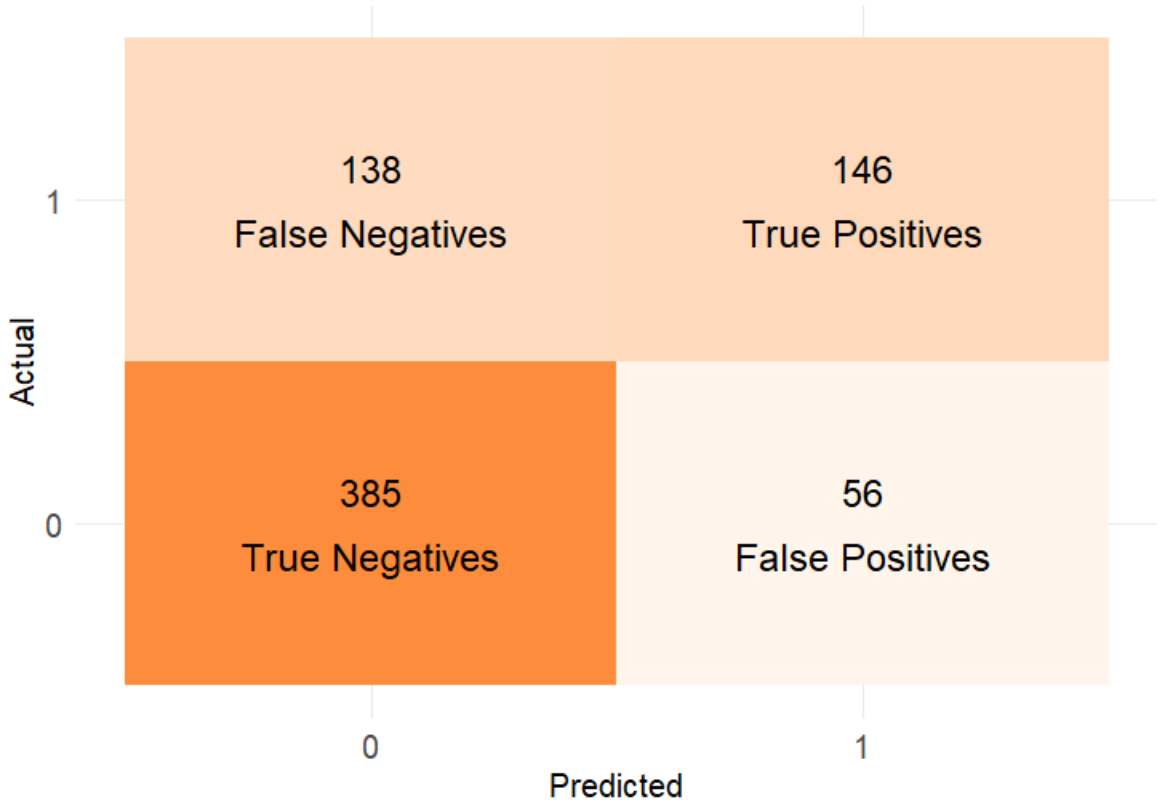
## Results

Table 2 shows the results of the SVM model with each kernel. Balanced Accuracy, F1 and AUC were chosen to evaluate the model as all of them are suitable for unbalanced classes and together they present a broad evaluation of the model's effectiveness.

*Table 2: Model results*

| Kernel | Balanced Accuracy | F1 Score | ROC - AUC |
|---|---|---|---|
| **Linear** | 0.6846 | 0.7967 | 0.6846 |
| **Radial** | **0.6936** | **0.7988** | **0.6936** |
| **Polynomial** | 0.6597 | 0.7896 | 0.6597 |

*Figure 7: Confusion matrix for radial model*




## Discussion

The radial kernel achieved the highest score across all three measures, therefore it can confidently be named the best of the three. However, this may change if more variables are included in the model.

Across all metrics, this model is less accurate than those of previous researchers with this dataset. This is understandable, given that this model only includes data known at the time of enrollment. However, there are several other possible reasons for why this model is not as accurate as would be wished.

Firstly, while the recategorisation of variables will have eliminated noise, it may also have obscured some pattern. Adding data from the year 2019-2020 onwards might assist with distinguishing the two, but would also increase complexity due to the disruption caused by Covid-19. Combining categories differently (particularly in Mother/Father Occupation, where there were many possible approaches) might have produced better results. In addition, as previous researchers make no mention of addressing this problem, it seems likely that their models suffered some degree of overfitting.

Secondly, the selection of variables may not have been optimal. While mutual information is a powerful method, it is sensitive to different feature scales and noise in the data (Geeks for Geeks, 2024), both of which were present here. Both Course and Gender were included in the model, but this may not have been the best approach (see above).

Thirdly, it is also possible that the decision not to address the mild imbalance in the data negatively affected results.

Interpretation of results was hampered by the limited information on some of the variables. Given the high proportion of students described as Displaced (55%), it may not refer to those forced to leave their homes (although Portugal has had several major wildfires in recent years). It might refer to students living away from home, but that is only a guess. Also, it is unclear if Debtor refers only to student loans, or any type of debt.

Inconsistency is also a concern, especially within Previous Qualification. Assuming that 'Frequency of Higher Education' does refer to current students (see Appendix B), 97% of transfer students haven't been put in this category, but listed under their last completed qualification.

Attempts were made to reduce data leakage (see above), but exploratory data analysis (EDA) was performed on the whole dataset. For best results, the data should have been split at the start and EDA performed only on the training set (Mucci, 2024).

## CONCLUSION

This study suggests that an institution-level model for predicting student dropout using only pre-enrolment data is possible. However, its usefulness should be improved by including additional features, particularly previous academic marks.

This study only considered traditional machine-learning methods, but some previous authors have had better results with this dataset using ensemble methods (Gupta *et al.* 2024; Kim, Yoo

and Kim, 2024). There are also many modelling parameters in Caret (and other packages) that might improve the model (but would require additional computing power). Future studies should consider a broader range of modelling options.

Models concerned with predicting human behaviour will always have relatively high failure rates, as they cannot account for unexpected life events, like serious illness or bereavement. Most higher education institutions don't routinely collect data on psychological factors, such as adaptability and stress management, but these factors do seem to affect dropout (Kocsis and Molnár, 2024). However, getting this data for all students would consume significant resources and would invade students' privacy. Even using the existing data raises legal and privacy issues.

Portugal is subject to the General Data Protection Regulation (GDPR), which controls the collection and use of personal data. To comply with this, students need to be informed that this model is in use. The institution should also know what data the model uses and how much importance it attaches to each variable. This can be difficult with black-box models (Klushin, 2022) but could be checked using permutation feature importance or similar.

The GDPR (and ethics) requires fair decision-making. While machines are not biased, data can be (Wachter-Boettcher, 2017). In this dataset, Father/Mother Occupation/Qualification (and new variables based on them) assume that each student has one male and one female parent, which is not necessarily the case[1]. Also, Gender is restricted to male and female only, which ignores those with other gender identities. Marital Status distinguishes between being married and living with a partner, but same-sex marriage was not made legal in Portugal until May 2010 (Equaldex, 2024) - after the data had started being collected.

ML models also require human oversight. Using the model to make autonomous decisions about which students receive extra support would open a legal and ethical minefield (Church, 2019), so it should only be used as a guide that can be overridden by human judgement (although that can also introduce bias). Ongoing monitoring is required to check that the model is making fair decisions. It is also vital that the model is only used for its intended purpose; using it to make admissions decisions would be highly unethical, and negatively affect efforts to widen access to higher education.

To sum up, using machine learning to predict student dropout is possible and may be helpful, but must be carefully implemented to avoid subjecting higher education institutions to unnecessary risks.

[1] Adoption by same-sex couples has only been legal in Portugal since 2016 (Equaldex, 2024), but this doesn't take account of informal family arrangements.

# REFERENCES


Agrusti, F., Bonavolontà, G. and Mezzini, M. (2019) *University Dropout Prediction Through Educational Data Mining Techniques: A Systematic Review*. Italy: Journal of e-learning and Knowledge Society.

Albreiki, B., Zaki, N. and Alashwal, H. (2021) *A Systematic Literature Review of Student Performance Prediction Using Machine Learning Techniques*. Basel: MDPI.

Bobbitt, Z. (2021) *What is Balanced Accuracy? (Definition & Example)*. Available at: https://www.statology.org/balanced-accuracy/ (Accessed: 29 Dec 2024).

Brownlee, J. (2020) *How to Choose a Feature Selection Method For Machine Learning*. Available at: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/ (Accessed: 16 Dec 2024).

Casanova, J.R., Cervero, A., Núñez, J.C., Almeida, L.S. and Bernardo, A. (2018) *Factors That Determine the Persistence and Dropout of University Students*. Spain: Colegio Oficial de Psicología del Principado de Asturias (COPPA)

Church, P. (2019) *AI & the GDPR: Regulating the minds of machines*. Available at: https://www.linklaters.com/en/insights/blogs/digilinks/ai-and-the-gdpr-regulating-the-minds-of-machines (Accessed: 23 Dec 2024).

Council of the European Union (2021) *Council Resolution on a strategic framework for European cooperation in education and training towards the European Education Area and beyond (2021-2030) 2021/C 66/01*. Luxembourg: Publications Office of the European Union.

Domínguez-Gómez, D., Sánchez-Jiménez, E., Hernández, Y., González Torres, J. and Ortiz-Hernandez, J. (2024) 'Enhancing Dropout Prediction Models Through Feature Selection Techniques'. MICAI 2024. INAOE, Tonantzintla, Puebla, México, 21-25 Oct 2024. Available at: https://www.researchgate.net/publication/385248111_Enhancing_Dropout_Prediction_Models_Through_Feature_Selection_Techniques (Accessed: 7 Nov 2024)

Equaldex (2024) *LGBT Rights in Portugal*. Available at: https://www.equaldex.com/region/portugal (Accessed: 23 Dec 2024).

European Commission (2015) *Dropout and completion in higher education in Europe : main report*. Luxembourg: Publications Office of the European Union.

Eurydice (2024) *Portugal > Higher Education*. Available at: https://eurydice.eacea.ec.europa.eu/national-education-systems/portugal/higher-education (Accessed 2 Nov 2024).

Geeks for Geeks (2024) *Information Gain and Mutual Information for Machine Learning*. Available at: https://www.geeksforgeeks.org/information-gain-and-mutual-information-for-machine-learning/ (Accessed: 20 Dec 2024).

Google (2024) *Datasets: Imbalanced datasets*. Available at: https://developers.google.com/machine-learning/crash-course/overfitting/imbalanced-datasets (Accessed 30 Oct 2024).

Gupta, K., Gupta, K., Dwivedi, P. and Chaudhry, M. (2024) 'Binary Classification of Students' Dropout Behaviour in Universities using Machine Learning Algorithms'. INDIACom 11. New Delhi, India, 28 Feb - 1 Mar 2024. Available at: https://ieeexplore.ieee.org/document/10498546 (Accessed: 31 Oct 2024)

Kaggle (2022) *Predict students' dropout and academic success*. Available at: https://www.kaggle.com/datasets/thedevastator/higher-education-predictors-of-student-retention (Accessed: 9 Oct 2024)

Kim, S., Yoo, E. and Kim, S. (2023) 'Why Do Students Drop Out? University Dropout Prediction and Associated Factor Analysis Using Machine Learning Techniques'. Available at: https://arxiv.org/abs/2310.10987 (Accessed: 31 Oct 2024)

Klushin, P. (2022) *GDPR Considerations When Training Machine Learning Models*. Available at: https://www.qwak.com/post/gdpr-considerations-when-training-machine-learning-models (Accessed: 23 Dec 2024).

Kocsis, A. and Molnár, G. (2024) *Factors Influencing Academic Performance and Dropout Rates in Higher Education*. Oxford: Oxford Review of Education

Martins, M.V., Tolledo, D., Machado, J., Baptista, L.M.T. and Realinho, V. (2021) 'Early prediction of student's performance in higher education: a case study', WorldCIST'21. Terceira Island, Azores, Portugal, 30–31 March and 1–2 April 2021. Berlin: Springer.

Mucci, T. (2024) *What is data leakage in machine learning?* Available at: https://www.ibm.com/think/topics/data-leakage-machine-learning (Accessed: 27 Dec 2024).

OECD (2022) *Education at a Glance 2022: OECD Indicators*. Paris: OECD Publishing. Available at: https://www.oecd-ilibrary.org/education/education-at-a-glance-2022_3197152b-en (Accessed Oct 2024).

Realinho, V., Machado, J., Baptista, L. and Martins, M. V. (2022) *Predicting Student Dropout and Academic Success*. Basel: MDPI.

Sangani, R. (2021) *Dealing with features that have high cardinality*. Available at: https://towardsdatascience.com/dealing-with-features-that-have-high-cardinality-1c9212d7ff1b (Accessed 17 Dec 2024).

Santos Lopes, M., Sismeiro Pereira, P. and Fortunato Vaz, P. (2022) 'Dropout Phenomenon in a Higher Education Institution in the North of Portugal, and the Reasons Behind It', EDULEARN22 Conference. Palma, Mallorca, Spain, 4-6 July 2022. Valencia: IATED. Available at: https://bibliotecadigital.ipb.pt/bitstream/10198/25727/1/SANTOSLOPES2022DRO.pdf (Accessed: Oct 2024).

Scikit-Learn (no date) *User Guide > 4. Inspection > 4.2. Permutation feature importance*. Available at: https://scikit-learn.org/stable/modules/permutation_importance.html (Accessed: 16 Dec 2024).

Singh, A.K. and Karthikeyan, S. (2024) 'A Wrapper Feature Selection Method for Predicting Student Dropout in Higher Education'. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5002077 (Accessed: 14 Nov 2024)

UCI (2021) *Predict Students' Dropout and Academic Success*. Available at: https://archive.ics.uci.edu/dataset/697/ (Accessed: 15 Nov 2024)

Wachter-Boettcher, S. (2017) *Technically Wrong: Sexist Apps, Biased Algorithms, and Other Threats of Toxic Tech*. New York: W.W.Norton & Company, Inc.

# APPENDICES

## Appendix A: Original Variables

Table 3 is the original table of variables that appeared in Realinho *et al.* (2022). Those variables not being considered in this paper are coloured grey.

NOTE: Most of the variables described as Numeric/discrete or Numeric/binary are label-encoded categorical variables.

*Table 3: Original variables*

| Group | Variable | Type |
|---|---|---|
| Demographic data | Marital status | Numeric/discrete |
| | Nationality | Numeric/discrete |
| | Displaced | Numeric/binary |
| | Gender | Numeric/binary |
| | Age at enrollment | Numeric/discrete |
| | International | Numeric/binary |

| | | |
|---|---|---|
| Socioeconomic data | Mother's qualification | Numeric/discrete |
| | Father's qualification | Numeric/discrete |
| | Mother's occupation | Numeric/discrete |
| | Father's occupation | Numeric/discrete |
| | Educational special needs | Numeric/binary |
| | Debtor | Numeric/binary |
| | Tuition fees up to date | Numeric/binary |
| | Scholarship holder | Numeric/binary |
| Macroeconomic data | Unemployment rate | Numeric/continuous |
| | Inflation rate | Numeric/continuous |
| | GDP | Numeric/continuous |
| Academic data at enrollment | Application mode | Numeric/discrete |
| | Application order | Numeric/ordinal |
| | Course | Numeric/discrete |
| | Daytime/evening attendance (named Attendance Mode in dataset) | Numeric/binary |
| | Previous qualification | Numeric/discrete |
| Academic data at the end of 1st semester | Curricular units 1st sem (credited) | Numeric/discrete |
| | Curricular units 1st sem (enrolled) | Numeric/discrete |
| | Curricular units 1st sem (evaluations) | Numeric/discrete |
| | Curricular units 1st sem (approved) | Numeric/discrete |
| | Curricular units 1st sem (grade) | Numeric/continuous |
| | Curricular units 1st sem (without evaluations) | Numeric/discrete |
| Academic data at the end of 2nd semester | Curricular units 2nd sem (credited) | Numeric/discrete |
| | Curricular units 2nd sem (enrolled) | Numeric/discrete |
| | Curricular units 2nd sem (evaluations) | Numeric/discrete |
| | Curricular units 2nd sem (approved) | Numeric/discrete |
| | Curricular units 2nd sem (grade) | Numeric/continuous |
| | Curricular units 2nd sem (without evaluations) | Numeric/discrete |
| Target | Target | Categorical |

## Appendix B: Mutation of Variables

*Re-categorised Variables*

Re-categorised variables used for the model are marked (New) and the original variables (Old).

*Table 4: Marital Status to Single*

| **Single (New)** | | **Marital Status (Old)** | |
|---|---|---|---|
| Number | Label | Number | Label |
| 0 | Living with Partner | 2<br>5 | Married<br>[De] Facto union |
| 1 | Single | 1<br>3<br>4<br>6 | Single<br>Widower<br>Divorced<br>Legally separated |

*Table 5: Previous Qualification to Previous Education*

| **Previous Education (New)** | | **Previous Qualification (Old)** | |
|---|---|---|---|
| Number | Label | Number | Label |
| 0 | Did not complete secondary | 7<br>8<br>9<br>10<br>11<br>12<br><br>13 | 12th year of schooling—not completed<br>11th year of schooling—not completed<br>Other - 11th year of schooling<br>10th year of schooling<br>10th year of schooling—not completed<br>Basic education 3rd cycle (9th/10th/11th year) or equivalent<br>Basic education 2nd cycle (6th/7th/8th year) or equivalent |
| 1 | Completed secondary | 1<br>6<br>14<br>16 | Secondary Education<br>Frequency of Higher Education *<br>Technological specialization course<br>Professional higher technical course |
| 2 | Completed first degree | 2<br>3<br>4<br>5<br>15<br>17 | Higher education -bachelor's degree<br>Higher education - degree<br>Higher education -master's degree<br>Higher education –doctorate<br>Higher education -degree (1st cycle) *<br>Higher education -master's degree (2nd cycle) * |

Notes:

- After some research, it seems 'Frequency of Higher Education' is a word-for-word translation of 'Frequência do Ensino Superior'. In this context, it actually means 'Attending/Attended/Attendance at Higher Education'. 12 of 15 in this category are transfer students (i.e. currently attending higher education). The other three are most likely returning to higher education after previously dropping out. However, for all the other transfer students their last *completed* qualification has been included instead.
- '1st cycle' is all of a bachelor's degree, '2nd cycle' all of a masters.

*Table 6: Mother/Father Qualification to Mother/Father Education*

| **Mother/Father Education (New)** | | **Mother/Father Qualification (Old)** | |
|---|---|---|---|
| Number | Label | Number | Label |
| 0 | Did not complete secondary | 7 | 12th Year of Schooling—not completed |
| | | 8 | 11th Year of Schooling—not completed |
| | | 9 | 7th Year (Old) |
| | | 10 | Other - 11th Year of Schooling |
| | | 11 | 2nd year complementary high school course |
| | | 12 | 10th Year of Schooling |
| | | 14 | Basic Education 3rd Cycle (9th/10th/11th Year) or Equivalent |
| | | 17 | Complementary High School Course—not concluded |
| | | 18 | 7th year of schooling |
| | | 20 | 9th Year of Schooling—not completed |
| | | 21 | 8th year of schooling |
| | | 25 | Cannot read or write |
| | | 26 | Can read without having a 4th year of schooling |
| | | 27 | Basic education 1st cycle (4th/5th year) or equivalent |
| | | 28 | Basic Education 2nd Cycle (6th/7th/8th Year) or equivalent |
| 1 | Completed secondary | 1 | Secondary Education—12th Year of Schooling or Equivalent |
| | | 6 | Frequency of Higher Education |
| | | 13 | General commerce course |
| | | 15 | Complementary High School Course |
| | | 16 | Technical-professional course |
| | | 19 | 2nd cycle of the general high school course |
| | | 22 | |

| | | | |
|---|---|---|---|
| | | 23<br>29<br>31<br>32 | General Course of Administration and Commerce<br>Supplementary Accounting and Administration<br>Technological specialization course<br>Specialized higher studies course<br>Professional higher technical course |
| 2 | Completed degree | 2<br>3<br>4<br>5<br>30<br>33<br>34 | Higher Education - bachelor's degree<br>Higher Education - degree<br>Higher Education - master's degree<br>Higher Education – doctorate<br>Higher education - degree (1st cycle)<br>Higher Education - master's degree (2nd cycle)<br>Higher Education - doctorate (3rd cycle) |

Notes:

- Values in original category 24 (Unknown) were replaced with the mode.

*Table 7: Mother/Father Occupation to Mother/Father Profession*

| **Mother/Father Profession (New)** | **Mother/Father Occupation (Old)** | |
|---|---|---|
| Number | Number | Label |
| 1 | 1 | Student |
| 2 | 2<br>17<br>18 | Representatives of the Legislative Power and Executive Bodies, Directors and Executive Managers<br>Directors of administrative and commercial services<br>Hotel, catering, trade, and other services directors |
| 3 | 3<br>19 | Specialists in Intellectual and Scientific Activities<br>Specialists in the physical sciences, mathematics, engineering, and related techniques |
| 4 | 4<br>20<br>21<br>22<br>23<br>24<br>25 | Intermediate Level Technicians and Professions<br>Health professionals<br>Teachers<br>Specialists in finance, accounting, administrative organization, and public and commercial relations<br>Intermediate level science and engineering technicians and professions<br>Technicians and professionals of intermediate level of health<br>Intermediate level technicians from legal, social, sports, cultural, and similar services |

| | | |
|---|---|---|
| | 26 | Information and communication technology technicians |
| 5 | 5 | Administrative staff |
| | 27 | Office workers, secretaries in general, and data processing operators |
| | 28 | Data, accounting, statistical, financial services, and registry-related operators |
| | 29 | Other administrative support staff |
| 6 | 6 | Personal Services, Security and Safety Workers, and Sellers |
| | 30 | Personal service workers |
| | 31 | Sellers |
| | 32 | Personal care workers and the like |
| | 33 | Protection and security services personnel |
| | 45 | Meal preparation assistants |
| | 46 | Street vendors (except food) and street service providers |
| 7 | 7 | Farmers and Skilled Workers in Agriculture, Fisheries, and Forestry |
| | 34 | Market-oriented farmers and skilled agricultural and animal production workers |
| | 35 | Farmers, livestock keepers, fishermen, hunters and gatherers, and subsistence |
| 8 | 8 | Skilled Workers in Industry, Construction, and Craftsmen |
| | 36 | Skilled construction workers and the like, except electricians |
| | 37 | Skilled workers in metallurgy, metalworking, and similar |
| | | Skilled workers in electricity and electronics |
| | 38 | Workers in food processing, woodworking, and clothing and |
| | 39 | other industries and crafts |
| 9 | 9 | Installation and Machine Operators and Assembly Workers |
| | 40 | Fixed plant and machine operators |
| | 41 | Assembly workers |
| | 42 | Vehicle drivers and mobile equipment operators |
| 10 | 10 | Unskilled Workers |
| | 43 | Unskilled workers in agriculture, animal production, and fisheries and forestry |
| | 44 | Unskilled workers in extractive industry, construction, manufacturing, and transport |
| 11 | 11 | Armed Forces Professions |
| | 14 | Armed Forces Officers |
| | 15 | Armed Forces Sergeants |
| | 16 | Other Armed Forces personnel |

Notes:

- Values in original categories 12 (Other Situation) and 13 (Blank) were replaced with the mode.

*Table 8: Target to Dropout*

| **Dropout (New)** | **Target (Old)** |
|---|---|
| 1 | Dropout |
| 0 | Graduate |
| Dropped | Enrolled |

*Categories where re-categorisation revealed redundant information*

*Table 9: Groups in Application Mode*

| **Application Group (New)** | | **Application Mode (Old)** | |
|---|---|---|---|
| Number | Label | Number | Label |
| 0 | General | 1<br>8<br>9 | 1st phase - general contingent<br>2nd phase - general contingent<br>3rd phase - general contingent |
| 1 | Transfer Students | 2<br>10<br>11<br>13<br>14<br>16<br>18 | Ordinance No. 612/93<br>Ordinance No. 533-A/99, item b2) (Different Plan)<br>Ordinance No. 533-A/99, item b3 (Other Institution)<br>Transfer<br>Change in course<br>Change in institution/course<br>Change in institution/course (International) |
| 2 | Holders of Other Qualifications | 4<br>15<br>17 | Holders of other higher courses<br>Technological specialization diploma holders<br>Short cycle diploma holders |
| 3 | International | 3<br>6<br>7 | 1st phase—special contingent (Azores Island)<br>International student (bachelor)<br>1st phase—special contingent (Madeira Island) |
| 4 | Over 23 | 12 | Over 23 years old |
| 5 | Uncertain | 5 | Ordinance No. 854-B/99 (Special Regimes)* |

Notes:

- Ordinance No. 854-B/99 (Special Regimes) covers multiple categories.

*Table 10: Nationality*

| **Nationality** | | |
|---|---|---|
| 1—Portuguese | 8—Angolan | 15—Romanian |
| 2—German | 9—Cape Verdean | 16—Moldova (Republic of) |
| 3—Spanish | 10—Guinean | 17—Mexican |
| 4—Italian | 11—Mozambican | 18—Ukrainian |
| 5—Dutch | 12—Santomean | 19—Russian |
| 6—English | 13—Turkish | 20—Cuban |
| 7—Lithuanian | 14—Brazilian | 21—Colombian |

*Other Variables*

*Table 11: Courses*

| **Course** | |
|---|---|
| Number | Label |
| 1 | Biofuel Production Technologies |
| 2 | Animation and Multimedia Design |
| 3 | Social Service (evening attendance) |
| 4 | Agronomy |
| 5 | Communication Design |
| 6 | Veterinary Nursing |
| 7 | Informatics Engineering |
| 8 | Equiniculture * |
| 9 | Management |
| 10 | Social Service |
| 11 | Tourism |
| 12 | Nursing |
| 13 | Oral Hygiene * |
| 14 | Advertising and Marketing Management |
| 15 | Journalism and Communication |
| 16 | Basic Education * |
| 17 | Management (evening attendance) |

Notes:

- From the school website (https://www.ipportalegre.pt/pt/oferta-formativa/), ‘Equiniculture’ should be ‘Equine Culture’, ‘Oral Hygiene’ is ‘Dental Nursing’ and ‘Basic Education’ is ‘Educational Studies’.

*Table 12: Gender*

| **Gender** | |
|---|---|
| Number | Label |
| 0 | Female |
| 1 | Male |

All other variables (except Age at Enrollment) are 1 – Yes, 0 – No.

## Appendix C: Code

Note: Most of the files were written as R notebooks within an R project, which is why the code is divided into sections. R script files containing duplicate code for exporting images have not been included here.

**Basic Cleaning**

```
# Load library

library(tidyverse)


# Import original data file

data <- read_csv("university.csv")


```

```
#BASIC CLEANING

```{r}


# Duplicate rows

sum(duplicated(data)) # None


# Duplicate columns

sum(duplicated(t(data))) # None
```
```

```
# Empty rows
nrow(data[apply(is.na(data) | data=="",1,all),]) # None

# Empty columns
ncol(data %>% select(where(function(x) all(is.na(x) | x==")))) # None

# Columns with single variable
sum(apply(data,2,function(x) min(x)==max(x))) # None

# NA and blank
sum(is.na(data)) # No NAs in dataset
sum(!grepl("", data)) # No empty strings either.

```

# ALTER VARIABLE NAMES TO STANDARD FORMAT

```{r}
colnames(data) <- str_squish(colnames(data))

# Change spaces to underscores
colnames(data) <-str_replace_all(colnames(data)," ","_")

# Remove trailing underscore
colnames(data) <- str_replace(colnames(data), "_$","")

# Remove 's from Father's and Mother's
colnames(data) <- str_replace(colnames(data),"r's","r")

# Correct spelling of 'Nacionality.
```

```
data <- data %>% rename("Nationality" = "Nacionality")

colnames(data) <- tolower(colnames(data))

# Remove / from daytime/evening_attendance
data <- data %>% rename("attendance_mode" = "daytime/evening_attendance")

# Remove brackets
colnames(data) <- str_replace(colnames(data), "\\(", "")
colnames(data) <- str_replace(colnames(data), "\\)", "")
```

```
# The dataset has three options for outcome, Dropout, Graduate, (still) Enrolled. To simplify the model, I'm going to change this to two options: Dropout or Not Dropout, dropping the Enrolled class.

```{r}
#ADJUST TARGET VARIABLE

# Rename the original 'target' variable as 'outcome'
data <- data %>% rename("outcome" = "target")

# Drop all the rows with outcome 'Enrolled'
data <- subset(data, outcome != "Enrolled")

# Create binary variable 'Dropout'
data <- data %>% mutate (
  dropout = ifelse(outcome == "Dropout",1,0)
)
```
```

# Check again that no columns with single variable

sum(apply(data,2,function(x) min(x)==max(x))) #None

```

# NON-RELEVANT COLUMNS

#For this model, I'm only considering pre-enrollment factors (i.e. only information available before enrollment), so all the curricular_unit columns and tuition fees will be deleted. Also the macro-economics factors.

```{r}

data <- data %>% select(-c(

 curricular_units_1st_sem_credited,

 curricular_units_1st_sem_enrolled,

 curricular_units_1st_sem_evaluations,

 curricular_units_1st_sem_approved,

 curricular_units_1st_sem_grade,

 curricular_units_1st_sem_without_evaluations,

 curricular_units_2nd_sem_credited,

 curricular_units_2nd_sem_enrolled,

 curricular_units_2nd_sem_evaluations,

 curricular_units_2nd_sem_approved,

 curricular_units_2nd_sem_grade,

 curricular_units_2nd_sem_without_evaluations,

 tuition_fees_up_to_date,

 unemployment_rate,

 inflation_rate,

 gdp,

 application_order

))

```

#Remaining size of the dataset.

```{r}

dim(data)

```

# Export Clean File

```{r}

write.csv(data, "university_clean.csv")

```

**Transformation of Variables**

---

title: "EDA Cleaning"

output: html_notebook

---

```{r}

# Load library

library(tidyverse)

# Assign dataframe

data <- read_csv("university_clean.csv")

data <- data %>% select(-...1)

```

# Convert categorical columns to factors

# (This is stripped out in the conversion to csv)
```

```{r}

data <- data %>%
    mutate(across(-c(age_at_enrollment),function(x) as.factor(x)))

str(data)

```

# VARIABLE CHANGES

# A number of variables had lots of categories with very few data points in them. I've re-categorised these into broader, more useful ones.

```{r}

# ALL OF THESE NEED TO BE TURNED INTO FACTORS AFTER THEY'RE CREATED!

# MARITAL STATUS
# Combining into 'Single' and 'Not Single'

data <- data %>% mutate (
 single = case_when(
  marital_status %in% c(1,3,4,6) ~ 1, #Single, Widower, Divorced, Legally Separated
  marital_status %in% c(2,5) ~ 0 #Married, De Facto
  ) %>%
  factor()
)

```

```{r}

# PREVIOUS_QUALIFICATION

table(data$previous_qualification)

data <- data %>% mutate(
  previous_education = case_when(
    previous_qualification %in% c(7,8,9,10,11,12,13) ~ 0, # Anything below secondary graduation
    previous_qualification %in% c(1,6,14,16) ~ 1, # Completed secondary, inc. post-secondary, less than bachelors
    previous_qualification %in% c(2,3,4,5,15,17) ~ 2 #Bachelors or above
  ) %>%
    factor(levels = c(0,1,2), ordered = TRUE) # Unlike most of the factors, education levels is ordinal
)

# 'Frequency of higher education' in post-secondary

# Initially had 5 groups, but very little difference between 'basic education' and 'didn't finish secondary' and between 'finished secondary' and 'post-secondary, less than bachelors'. Combined further.

table(data$previous_education)

# PARENT QUALIFICATIONS

# This was troublesome. There are a number of now-discontinued qualifications that were difficult to classify.

# There are unknowns in these, which will be replaced with the mode once the data is split.

table(data$mother_qualification)
```

```
data <- data %>% mutate(
  mother_education = case_when(
    mother_qualification %in% c(9,14,20,21,25,26,27,28,7,8,10,11,12,14,17,18,19) ~ 0, #Didn't finish secondary
    mother_qualification %in% c(1,13,15,16,19,22,23,6,29,31,32) ~ 1, #Finished secondary
    mother_qualification %in% c(2,3,4,5,30,33,34) ~ 2, #Bachelors or above
    mother_qualification %in% c(24) ~ NA #Unknown
  ) %>%
    factor(levels = c(0,1,2), ordered = TRUE)
)

data <- data %>% mutate(
  father_education = case_when(
    father_qualification %in% c(9,14,20,21,25,26,27,28,7,8,10,11,12,17) ~ 0, #Didn't finish secondary
    father_qualification %in% c(1,13,15,16,19,22,23,6,29,31,32) ~ 1, #Finished secondary
    father_qualification %in% c(2,3,4,5,30,33,34) ~ 2, #Bachelors or above
    father_qualification %in% c(24) ~ NA #Unknown
  ) %>%
    factor(levels = c(0,1,2), ordered = TRUE)
)

# PARENT OCCUPATION

data <- data %>% mutate(
  mother_profession = case_when(
    mother_occupation %in% c(12,13) ~ NA, #Unknown
    mother_occupation %in% c(1) ~ 1,
```

```
  mother_occupation %in% c(2,17,18) ~ 2,
  mother_occupation %in% c(3,19) ~ 3,
  mother_occupation %in% c(4,20,21,22,23,24,25,26) ~ 4,
  mother_occupation %in% c(5,27,28,29) ~ 5,
  mother_occupation %in% c(6,30,31,32,33,44,45,46) ~ 6,
  mother_occupation %in% c(7,34,35) ~ 7,
  mother_occupation %in% c(8,36,37,38,39) ~ 8,
  mother_occupation %in% c(9,40,41,42) ~ 9,
  mother_occupation %in% c(10,43,44) ~ 10,
  mother_occupation %in% c(11,14,15,16) ~ 11
 ) %>%
  factor()
)

data <- data %>% mutate(
 father_profession = case_when(
  father_occupation %in% c(12,13) ~ NA, #Unknown
  father_occupation %in% c(1) ~ 1,
  father_occupation %in% c(2,17,18) ~ 2,
  father_occupation %in% c(3,19) ~ 3,
  father_occupation %in% c(4,20,21,22,23,24,25,26) ~ 4,
  father_occupation %in% c(5,27,28,29) ~ 5,
  father_occupation %in% c(6,30,31,32,33,44,45,46) ~ 6,
  father_occupation %in% c(7,34,35) ~ 7,
  father_occupation %in% c(8,36,37,38,39) ~ 8,
  father_occupation %in% c(9,40,41,42) ~ 9,
  father_occupation %in% c(10,43,44) ~ 10,
  father_occupation %in% c(11,14,15,16) ~ 11
 ) %>%
```

```
  factor()
)

table(data$father_occupation)

# The biggest dropout rates are in 'unknown' (no use) and 'student'!
```

# APPLICATION MODE

```{r}

data <- data %>% mutate(
    application_group = case_when(
  application_mode %in% c(1,8,9) ~ 0, # General
  application_mode %in% c(2,10,11,13,14,16,18) ~ 1, # Course/institution transfer/reenrollment
  application_mode %in% c(3,4,5,6,7,12,15,17) ~ 2 # Other
 ) %>%
  factor()
)

# TRANSFER

# From the MI scores, application mode is too related to age and previous qualification. Going to try just separating out transfer students.

data <- data %>% mutate(
    transfer = case_when(
    application_group %in% c(1) ~ 1,
    application_group %in% c(0,2) ~ 0
```

```
  ) %>%
    factor()
)
```

# PARENTS

```{r}

# SAME SEX PARENT EDUCATION/PROFESSION
# Would this correlate better with Dropout?

data <- data %>% mutate(
  same_sex_parent_education = case_when(
    gender == 1 ~ father_education,
    gender == 0 ~ mother_education
  ) %>%
    factor()
)

data <- data %>% mutate(
  same_sex_parent_profession = case_when(
    gender == 1 ~ father_profession,
    gender == 0 ~ mother_profession
  ) %>%
    factor()
)

#No, both lower.
```

```
# OPPOSITE SEX PARENT EDUCATION/PROFESSION
# Must test assumptions

data <- data %>% mutate(
  opposite_sex_parent_education = case_when(
    gender == 0 ~ father_education,
    gender == 1 ~ mother_education
  ) %>%
    factor()
)

data <- data %>% mutate(
  opposite_sex_parent_profession = case_when(
    gender == 0 ~ father_profession,
    gender == 1 ~ mother_profession
  ) %>%
    factor()
)

# Both lower

# Highest education level of parent?

data <- data %>% mutate(highest_parent_education = case_when(
    as.numeric(father_education) >= as.numeric(mother_education) ~ father_education,
    as.numeric(father_education) < as.numeric(mother_education) ~ mother_education
  ) %>%
    factor()
)
```

```
# No improvement

# Group Courses by School (Info from website)

data <- data %>% mutate(academic_school = case_when(
        course %in% c(1,2,5,7,9,14,17) ~ 0, #ESTGD
        course %in% c(4,6,8) ~ 1, #ESBE
        course %in% c(3,10,11,15,16) ~ 2, #ESECS
        course %in% c(12,13) ~ 3 #ESS
        ) %>%
        factor()
)

```

```
# Age

```{r}
# Under 25

data <- data %>% mutate(under_25 = ifelse(age_at_enrollment < 25, 1, 0))
data <- data %>% mutate(under_24 = ifelse(age_at_enrollment < 24, 1, 0))

#Not giving better results

data <- data %>% mutate(age_band = case_when(
        age_at_enrollment < 25 ~ 0,
        age_at_enrollment >= 25 & age_at_enrollment < 35 ~ 1,
        age_at_enrollment >= 35 & age_at_enrollment < 45 ~ 2,
```
```

```
      age_at_enrollment >= 45 & age_at_enrollment < 55 ~ 3,
      age_at_enrollment >= 55 ~ 4)
      )

# Not better either.

```

**Exploratory Data Analysis**

```
---
title: "EDA university"
output: html_notebook
---
```{r}
library(tidyverse)
library(gridExtra)
```

# Marital Status

```{r}

table(data$marital_status)

# 1.Single, 2.Married, 3.Widower, 4.Divorced, 5.Common-law Marriage, 6.Legally Separated

ggplot(data) +
 geom_bar(
   mapping = aes(x = marital_status, fill = dropout),
```
```

```
  position = "fill"
  )

# Way more single than anything else. If going to use, probably combine into 'single' and 'living with partner'.

table(data$single)

#0.Married or living with partner, 1.Other

ggplot(data) +
 geom_bar(mapping = aes(x = single, fill = dropout),
       position = "fill"
       )

```

```{r}
# COURSE AND ATTENDANCE MODE

ggplot(data) +
 geom_bar(mapping = aes(x = course, fill = attendance_mode),
       position = "fill")

# The course list has these as evening attendance. Everything else is daytime. So this is useless.
# Why did this not show up on the dataset paper?

social_service <- data %>%
          select(c(course,attendance_mode, dropout)) %>%
          filter(course == c(3,10))
```

```
ggplot(social_service) +
 geom_bar(mapping = aes(x = attendance_mode, fill = dropout),
        position = "fill")


management <- data %>%
          select(c(course, attendance_mode, dropout)) %>%
          filter(course == c(9,17))


ggplot(management) +
 geom_bar(mapping = aes(x = attendance_mode, fill = dropout),
        position = "fill")


# So, attendance mode makes a difference, but surely it's already captured in the different courses.


```

# COURSE AND ACADEMIC SCHOOL

```{r}

table(data$course)


ggplot(data) +
 geom_bar(mapping = aes(x=course, fill=dropout),
       position = "fill")
# Course looks important. Might have to drop the smallest one. Could group them into subject areas?
```

```
table(data$academic_school)

ggplot(data) +
 geom_bar(mapping = aes(x = academic_school, fill=dropout),
        position = "fill")

# Think we're losing too much detail here.

```

```
# BINARY VARIABLES

```{r}

#table(data$gender)

ggplot(data) +
 geom_bar(
   mapping = aes(x = gender, fill=dropout),
   position = "fill")

# More female students that male, but dropout numbers are nearly the same.

#table(data$scholarship_holder)

ggplot(data) +
 geom_bar(
   mapping = aes(x = scholarship_holder, fill=dropout),
   position = "fill")
```
```

```
# Going to want that in.

#table(data$educational_special_needs) #Only 50 of them.

ggplot(data) +
 geom_bar(mapping = aes(x = educational_special_needs, fill=dropout),
 position = "fill")

# No real difference here. The university must have good support systems!

#table(data$nationality) # Same as the international/home split, so could just delete this column
#table(data$international)

prop.table(table(data$displaced))

ggplot(data) +
 geom_bar(mapping = aes( x = displaced, fill = dropout),
 position = "fill")

# Loads of displaced students. Seem to be less likely to drop out.

table(data$debtor)

ggplot(data) +
 geom_bar(mapping = aes(x = debtor, fill = dropout),
 position = "fill")

ggplot(data) +
 geom_bar(
```

```
  mapping = aes(x = single, fill = factor(dropout)),
  position = "fill")

#Dropout rate is clearly higher in those married/living with a partner

```

# International / Nationality

```{r}
table(data$nationality) # Same as the international/home split, so could just delete this column
table(data$international)

prop.table(table(data$nationality))

# Nationality starts at 1.

# Home students who are not Portuguese. None
rowSums(data %>%
      select(nationality,international) %>%
      filter(as.integer(nationality) != 1, as.integer(international) == 0))

# International students who are Portuguese. None
rowSums(data %>%
      select(nationality,international) %>%
      filter(nationality %in% c('1'), international %in% c('1')))

ggplot(data) +
 geom_bar(
  mapping = aes(x = international, fill=dropout),
```

```
  position = "fill")

# Proportions are about the same and there's hardly any international students anyway, so drop this.

```

# Age at Enrollment

```{r}

table(data$age_at_enrollment) # Group that?

ggplot(data) +
 geom_histogram(mapping = aes(x= age_at_enrollment, fill=dropout),
 position = "fill")

ggplot(data, aes(x= age_at_enrollment, fill = dropout)) +
 geom_histogram(position = "fill") +
 xlim(17,30)

# Bit of a trend up to late 20s, then no. But trying to capture this isn't getting better results.

```

# Previous Education and Parent Education

```{r}

#table(data$mother_qualification)
```

```
# Definitely condense the qualification tables, since so many have less than 10.

#table(data$mother_occupation)

# Another one for condensing

ggplot(data) +
 geom_bar(mapping = aes( x = mother_occupation, fill = dropout),
 position = "fill"
       )



ggplot(data) +
 geom_bar(mapping = aes(x = previous_education, fill = factor(dropout)),
       position = "fill")

ggplot(data) +
 geom_bar(mapping = aes(x = mother_education, fill = factor(dropout)),
       position = "fill")

ggplot(data) +
 geom_bar(mapping = aes(x = mother_education))

table(data$mother_education) #tiny amount in 3, few in 0, few in 5 but that's unknown

table(data$father_education) #still few in 3, loads in 0 not many in 1

ggplot(data) +
 geom_bar(mapping = aes(x = father_education, fill = factor(dropout)),
```

```
        position = "fill")



```

# Parent professions

```{r}
ggplot(data) +
 geom_bar(mapping = aes(x = mother_profession, fill = factor(dropout)),
        position = "fill"
 )

ggplot(data) +
 geom_bar(mapping = aes(x = father_profession, fill = factor(dropout)),
        position = "fill"
        )

table(data$mother_profession) # Four of these are still under 100.
table(data$father_profession)

```


# Application Mode

```{r}
table(data$application_mode) # don't know what half the options mean.

ggplot(data) +
 geom_bar(
```

```
  mapping = aes(x = application_mode, fill=factor(dropout)),
  position = "fill"
 )

# Further investigation required. Probably group them.

table(data$application_group)

ggplot(data) +
 geom_bar(
  mapping = aes(x = application_group, fill = factor(dropout)),
  position = "fill"
 )
```

# Transfer

```{r}
table(data$transfer)

ggplot(data) +
 geom_bar(mapping = aes(x = transfer, fill = factor(dropout)),
       position = "fill")

# Small difference, and that has a much smaller MI score.

transfers <- data %>%
         select(c(transfer, previous_qualification)) %>%
         filter(transfer == 1)
```

```
table(transfers)

#6 is Frequency of Higher Education

# transfer students in other PQ cats
others <- ((355-12)/355)*100
others
```

```{r}
colSums(is.na(data))/nrow(data)
```

# Previous Qualification
```{r}
freqs <- data %>%
      select(c(previous_qualification, age_at_enrollment,application_mode)) %>%
      filter(previous_qualification == 6)

data %>%
 select(c(previous_qualification, transfer)) %>%
 filter(transfer == 1)
```

```{r}
ggplot(data) +
 geom_bar(mapping = aes(x = course, fill = gender),
       position = "fill") +
 xlab("") +
```

```
  ylab("") +
  coord_flip() +
  scale_fill_manual(values = c("pink", "#4E84C4")) +
  labs(title = "Course Gender Split") +
  scale_x_discrete(labels = c("Biofuel Production Technologies", "Animation & Multimedia Design", "Social Service (eve)", "Agronomy", "Communication Design", "Veterinary Nursing", "Informatics Engineering", "Equiniculture", "Management", "Social Service", "Tourism", "Nursing", "Oral Hygiene", "Advertising & Marketing Management", "Journalism & Communication", "Basic Education", "Management (eve)")) +
  scale_y_continuous(labels = c("0", "25%", "50%", "75%", "100%")) +
  theme_minimal() +
  theme(legend.position = "none")


ggplot(data) +
  geom_bar(mapping = aes(x = course, fill = dropout),
           position = "fill") +
  coord_flip()


# Percent male on each course


percent_male <- data %>%
            group_by(course) %>%
            reframe(course_percent_male = round(mean(gender==1)*100),
                    course_percent_dropout = round(mean(dropout==1)*100)
                    )


cor(percent_male$course_percent_male, percent_male$course_percent_dropout) #0.7


ggplot(percent_male, aes(x = course_percent_male, y = course_percent_dropout)) +
  geom_point() +
```

```
  scale_x_continuous(expand = c(0, 0), limits = c(0, 100)) +
  scale_y_continuous(expand = c(0, 0), limits = c(0, 100)) +
  labs(title = "Percentage of Males vs Percentage of Dropouts by Course\n", x = "% Males", y = "% Dropout")



# Outlier is Basic Education

p.all <- ggplot(data) +
        geom_bar(mapping = aes(x = course, fill = dropout),
                 position = "fill") +
        coord_flip()

women <- data %>%
        filter(gender == 0)

p.women <- ggplot(women) +
          geom_bar(mapping = aes(x = course, fill = dropout),
            position = "fill") +
          coord_flip() +
          labs(title = "Women")

men <- data %>%
       filter(gender == 1)

p.men <- ggplot(men) +
        geom_bar(mapping = aes(x = course, fill = dropout),
         position = "fill") +
        coord_flip() +
```

```
        labs(title = "Men")

grid.arrange(p.all, p.women, p.men)

#There are definitely similarities if you split by gender, but the patterns aren't the same.
```

# Gender and other variables

```{r}
ggplot(data) +
 geom_boxplot(mapping = aes(x = age_at_enrollment, fill = gender)) +
 ylab("") +
 theme(legend.position = "none", axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
 scale_fill_manual(values = c("pink", "#4E84C4")) +
 labs(title = "Age Distribution by Gender") +
 xlab("Age at Enrollment")

ggplot(data) +
 geom_bar(mapping = aes(x = debtor, fill = gender),
          position = "fill") +
 scale_fill_manual(values = c("pink", "#4E84C4"))

ggplot(data) +
 geom_bar(mapping = aes(x = scholarship_holder, fill = gender),
          position = "fill") +
 scale_fill_manual(values = c("pink", "#4E84C4"))
```

```
ggplot(data) +
 geom_bar(mapping = aes(x = displaced, fill = gender),
          position = "fill") +
 scale_fill_manual(values = c("pink", "#4E84C4"))

ggplot(data) +
 geom_bar(mapping = aes(x = previous_education, fill = gender),
          position = "fill") +
 scale_fill_manual(values = c("pink", "#4E84C4"))

```
```

**Mutual Information**

```
---
title: "Mutual Information - University"
output: html_document
---
```{r}
# Install libraries
library(infotheo)
library(tidyverse)
library(RColorBrewer)

data <- data

# Impute mode function
impute_mode <- function(data,columns)
{
 mode_function <- function(x) names(sort(table(x), decreasing = TRUE))[1]
```

```
  data %>%
    mutate(across(all_of(columns), ~ ifelse(is.na(.), mode_function(.), .)))
}

# Hex codes for the colour palette
brewer.pal(8, "Oranges")
```

# Original Variables

```{r}

original_vars <- data %>%
            select(c(dropout, marital_status, application_mode, course, attendance_mode, previous_qualification, nationality, mother_qualification, father_qualification, mother_occupation, father_occupation, displaced, educational_special_needs, debtor, gender, scholarship_holder, age_at_enrollment, international)) %>%
            impute_mode(c("mother_qualification", "father_qualification", "mother_occupation", "father_occupation"))

mut_info_org <- mutinformation(original_vars)

heatmap(mut_info_org, scale = "column", Colv = NA, Rowv = NA, margins = c(12,3), main = "Heatmap of MI Scores", col = brewer.pal(8, "Oranges"))

mi_scores_org <- mut_info_org[-1,"dropout"] %>%
          sort()

par(mar = c(4,13,0,0))
barplot(mi_scores_org,
      horiz = T,
```

```
    las = 1,
    col = "#FD8D3C",
    xlab = "Mutual Information Score with Dropout (Original Variables")

```

# All the parent variables

```{r}

parents <- data %>%
      select(c(dropout, mother_education, mother_profession, father_education, father_profession, opposite_sex_parent_education, opposite_sex_parent_profession, same_sex_parent_education, same_sex_parent_profession, highest_parent_education)) %>%
      impute_mode(everything())

mut_info_parents <- mutinformation(parents)

heatmap(mut_info_parents, scale = "column", Colv = NA, Rowv = NA, margin = c(12,3), col = brewer.pal(8, "Oranges"))

mi_scores_parents <- mut_info_parents[-1,"dropout"] %>%
       sort()

par(mar = c(4,13,0,0))
barplot(mi_scores_parents,
    horiz = T,
    las = 1,
    col = "#FD8D3C",
    xlab = "Mutual Information Score with Dropout")
```

```

# Everything Not Eliminated

```{r}

rest <- data %>%
    select(c(dropout, course, age_at_enrollment, scholarship_holder,
same_sex_parent_education, opposite_sex_parent_profession, debtor, gender,
previous_education, displaced, single, international, educational_special_needs, transfer)) %>%
    impute_mode(c("same_sex_parent_education", "opposite_sex_parent_profession"))

mut_info_rest <- mutinformation(rest)

heatmap(mut_info_rest, scale = "column", Colv = NA, Rowv = NA, margin = c(12,3), col =
brewer.pal(8, "Oranges"))

mi_scores_rest <- mut_info_rest[-1,"dropout"] %>%
        sort()

par(mar = c(4,13,5,0))
barplot(mi_scores_rest,
    horiz = T,
    las = 1,
    col = "#FD8D3C",
    main = "Mutual Information Score with Dropout")

```

Chi-Squared Test

# Chi - Squared

```{r}

```
library(RColorBrewer)

library(dplyr)


df_chi <- data %>%

    select(c(course, displaced, educational_special_needs, debtor, gender, scholarship_holder, same_sex_parent_education, opposite_sex_parent_profession, previous_education, single, international, dropout))


matrix <- data.matrix(df_chi) # turn data into matrix

cols <- asplit(matrix, 2) # split matrix into columns


# Apply the chi test to the table of every pair of columns in the matrix.

# sim = TRUE clears errors 'approximation' messages.

# $p.value gets the score for each one.


chis <- outer(cols, cols, Vectorize(function(x,y) chisq.test(table(x,y), sim = TRUE)$p.value))


heatmap(chis, scale = "column", Colv = NA, Rowv = NA, margins = c(12,3), main = "Heatmap of P Values", col = brewer.pal(8, "Oranges"))


# With p-values, we actually want lower scores, so we're looking for the white bits


# Confirms no relationship between International/Education Special Needs and Dropout


drop_scores <- chis[-c(3,11),12]


barplot(drop_scores)


#All the rest are the same.
```

```

**Model**

# MODEL

```{r}
library(tidyverse)
library(caret)
library(e1071)
library(kernlab)
library(pROC)
library(RColorBrewer)

# Impute mode function
impute_mode <- function(data,columns)
{
 mode_function <- function(x) names(sort(table(x), decreasing = TRUE))[1]

 data %>%
  mutate(across(all_of(columns), ~ ifelse(is.na(.), mode_function(.), .)))
}
```

# Features

```{r}
model_set <- data %>%
        select(c(dropout, age_at_enrollment, course, scholarship_holder, debtor, gender, same_sex_parent_education, previous_education, opposite_sex_parent_profession, displaced))

# Convert everything except dropout to numeric, so can scale it.
```
```

```
model_set[,-1] <- model_set[,-1] %>%
                  sapply(as.numeric)

```

# Data splitting

```{r}
set.seed(5678)

train_rows <- createDataPartition(model_set$dropout, p = 0.8, list = FALSE, times = 1)

df_train <- model_set[train_rows, ]
df_test <- model_set[-train_rows,]

# Check set proportions
test_prop <- nrow(df_test)/(nrow(df_test) + nrow(df_train))
test_prop # 0.1997 in test set, so split is fine.

# Check dropout proportions
drop_prop <- nrow(filter(df_test, dropout == 1)) / (nrow(filter(df_train, dropout == 1)) +
nrow(filter(df_test, dropout == 1)))
drop_prop # 0.1999 in test set. Great.
```

# Model settings

```{r}
# Impute NAs in relevant columns
# Doing this in each set separately to avoid data leakage.
```

```
df_train <- df_train %>%
        impute_mode(c("same_sex_parent_education", "opposite_sex_parent_profession"))

df_test <- df_test %>%
        impute_mode(c("same_sex_parent_education", "opposite_sex_parent_profession"))

# 10-fold cross-validation
# The random search is for parameter tuning. It will try however many random combinations of tuning features that tuneLength tells it to in each model.
train_control <- trainControl(method = "cv", number = 10, search = "random")

# Test Labels
test_labels = as.numeric(df_test$dropout)
```

# Linear Model

```{r}
lin_start_time <- Sys.time()
svm_linear <- train(dropout ~ ., data = df_train, method = "svmLinear", trControl = train_control, preProcess = "scale", tuneLength = 20)
# The data is scaled here to prevent data leakage between cross-validation folds.

lin_end_time <- Sys.time()
lin_training_time <- lin_end_time - lin_start_time
print(svm_linear)

lin_test_start_time <- Sys.time()
linear_predictions <- predict(svm_linear, newdata = df_test)
```

```
lin_test_end_time <- Sys.time()
lin_test_time <- lin_test_end_time - lin_test_start_time

cat("Linear Training Time:", lin_training_time, "\n")
cat("Linear Testing Time:", lin_test_time, "\n")
cat("Best C value:", svm_linear$bestTune[1,1])

best_c_value <- svm_linear$bestTune[1,1]
```

```
# Linear Model Evaluation
```r
# Confusion Matrix

cm_linear <- confusionMatrix(linear_predictions, df_test$dropout, mode = "prec_recall")

fourfoldplot(cm_linear$table, color = c("#FFF5EB", "#FD8D3C"), main = "Confusion Matrix - Linear")

as.data.frame(cm_linear$table)

ggplot(as.data.frame(cm_linear$table), aes(x = Prediction, y = Reference, fill = Freq)) +
 geom_tile() +
 scale_fill_gradient(low = "#FFF5EB", high = "#FD8D3C") +
 theme_minimal() +
 geom_text(aes(label = Freq), color = "black", size = 5, nudge_y = 0.1) +
 geom_text(aes(label = c("TN", "FP", "FN", "TP")), size = 5, nudge_y = -0.1) +
 labs(x = "Predicted", y = "Actual", title = "Confusion Matrix") +
```
```

```
 theme(legend.position="none", plot.title = element_text(hjust=0.5, size = 14), axis.text = element_text(size = 12), axis.title = element_text(size = 12))


# ROC - AUC


lin_test_preds = as.numeric(linear_predictions)


lin_roc_curve <- roc(test_labels, lin_test_preds)

plot(lin_roc_curve, col = "#FD8D3C", main = "ROC Curve - Linear", lwd = 2)

lin_auc_value <- auc(lin_roc_curve)

text(0.6,0.4, paste("AUC =", round(lin_auc_value, 4)), col = "#FD8D3C")


```

# Radial Model


```{r}


rad_start_time <- Sys.time()

svm_radial <- train(dropout ~ ., data = df_train, method = "svmRadial", trControl = train_control, preProcess = "scale", tuneLength = 20)


rad_end_time <- Sys.time()

rad_training_time <- rad_end_time - rad_start_time

print(svm_radial)


rad_test_start_time <- Sys.time()

rad_predictions <- predict(svm_radial, newdata = df_test)

rad_test_end_time <- Sys.time()

rad_test_time <- rad_test_end_time - rad_test_start_time
```

```
cat("Radial Training Time:", rad_training_time, "\n")

cat("Radial Testing Time:", rad_test_time, "\n")
```

```

# Radial Model Evaluation

```{r}
# Confusion Matrix

cm_rad <- confusionMatrix(rad_predictions, df_test$dropout, mode = "prec_recall")

cm_rad

fourfoldplot(cm_rad$table, color = c("#FFF5EB", "#FD8D3C"), main = "Confusion Matrix - Radial")

as.data.frame(cm_rad$table)

ggplot(as.data.frame(cm_rad$table), aes(x = Prediction, y = Reference, fill = Freq)) +
  geom_tile() +
  scale_fill_gradient(low = "#FFF5EB", high = "#FD8D3C") +
  theme_minimal() +
  geom_text(aes(label = Freq), color = "black", size = 5, nudge_y = 0.1) +
  geom_text(aes(label = c("TN", "FP", "FN", "TP")), size = 5, nudge_y = -0.1) +
  labs(x = "Predicted", y = "Actual", title = "Confusion Matrix") +
  theme(legend.position="none", plot.title = element_text(hjust=0.5, size = 14), axis.text = element_text(size = 12), axis.title = element_text(size = 12))
```
```

# ROC - AUC

rad_test_preds = as.numeric(rad_predictions)

rad_roc_curve <- roc(test_labels, rad_test_preds)

plot(rad_roc_curve, col = "#FD8D3C", main = "ROC Curve - Radial", lwd = 2)

rad_auc_value <- auc(rad_roc_curve)

text(0.6,0.4, paste("AUC =", round(rad_auc_value, 4)), col = "#FD8D3C")

```

# Polynomial Model

```{r}

poly_start_time <- Sys.time()

svm_poly <- train(dropout ~ ., data = df_train, method = "svmPoly", trControl = train_control, preProcess = "scale", tuneLength = 20)

poly_end_time <- Sys.time()

poly_training_time <- poly_end_time - poly_start_time

print(svm_poly)

poly_test_start_time <- Sys.time()

poly_predictions <- predict(svm_poly, newdata = df_test)

poly_test_end_time <- Sys.time()

poly_test_time <- poly_test_end_time - poly_test_start_time

cat("Polynomial Training Time:", poly_training_time, "\n")

cat("Polynomial Testing Time:", poly_test_time, "\n")

```

# Polynomial Model Evaluation

```{r}

# Confusion Matrix

cm_poly <- confusionMatrix(poly_predictions, df_test$dropout, mode = "prec_recall")

cm_poly

fourfoldplot(cm_poly$table, color = c("#FFF5EB", "#FD8D3C"), main = "Confusion Matrix - Polynomial")

as.data.frame(cm_poly$table)

ggplot(as.data.frame(cm_poly$table), aes(x = Prediction, y = Reference, fill = Freq)) +
  geom_tile() +
  scale_fill_gradient(low = "#FFF5EB", high = "#FD8D3C") +
  theme_minimal() +
  geom_text(aes(label = Freq), color = "black", size = 5, nudge_y = 0.1) +
  geom_text(aes(label = c("TN", "FP", "FN", "TP")), size = 5, nudge_y = -0.1) +
  labs(x = "Predicted", y = "Actual", title = "Confusion Matrix") +
  theme(legend.position="none", plot.title = element_text(hjust=0.5, size = 14), axis.text = element_text(size = 12), axis.title = element_text(size = 12))

# ROC - AUC
```
```

```
poly_test_preds = as.numeric(poly_predictions)


poly_roc_curve <- roc(test_labels, poly_test_preds)

plot(poly_roc_curve, col = "#FD8D3C", main = "ROC Curve - polynomial", lwd = 2)

poly_auc_value <- auc(poly_roc_curve)

text(0.6,0.4, paste("AUC =", round(poly_auc_value, 4)), col = "#FD8D3C")
```


# Model Comparison


```{r}


comp_matrix <- round(matrix(c(cm_linear$byClass['Balanced Accuracy'],
cm_rad$byClass['Balanced Accuracy'], cm_poly$byClass['Balanced Accuracy'],
cm_linear$byClass['F1'],  cm_rad$byClass['F1'], cm_poly$byClass['F1'], lin_auc_value[1],
rad_auc_value[1], poly_auc_value[1], lin_training_time, rad_training_time, poly_training_time,
lin_test_time, rad_test_time, poly_test_time), ncol = 5),4)


colnames(comp_matrix) <- c("Balanced Accuracy", "F1 Score", "AUC", "Training Time", "Testing
Time")

rownames(comp_matrix) <- c("Linear", "Radial", "Polynomial")


comp_matrix


# The polynomial model is taking way longer than that to train.


matrix2 <- round(matrix(c(cm_linear$byClass['Balanced Accuracy'], cm_rad$byClass['Balanced
Accuracy'], cm_poly$byClass['Balanced Accuracy'], cm_linear$byClass['F1'],
cm_rad$byClass['F1'], cm_poly$byClass['F1'], lin_auc_value[1], rad_auc_value[1],
poly_auc_value[1]), ncol = 3),4)
```

```
colnames(matrix2) <- c("Balanced Accuracy", "F1 Score", "AUC")
rownames(matrix2) <- c("Linear", "Radial", "Polynomial")

matrix2
```