Who will drop out of university? Using logistic regression, neural networks and decision trees to predict student drop out rates

FYS-STK4155 Project 3

Helene Lane and Manuela Leal Nader
University of Oslo
(Dated: December 15, 2025)

Student dropout is a complex problem that most universities are confronted with. It's complexity comes from the many different underlaying causes, which makes it harder for institutions to evaluate which measures would create the greater positive effect. In this scenario, machine learning methods could play an important part, by either identifying students with a higher risk of dropping out, or by recognizing common aspects that lead to it. Based on data from a previously published dataset [1], we have used different machine learning models to classify students within three groups: enrolled, graduate and dropout. Logistic regression and neural networks achieved the highest accuracy, both reaching 77.97%. Even though it led to a lower accuracy (75.7%), decision trees identified the `Curricular units 2nd semester (approved)` category as having the highest importance. Results like these can contribute to deeper our understanding of such a problem.

I. INTRODUCTION

Universities in countries like Norway rely on students completing to get funding [2]. Therefore, these institutions may want to identify students who are at risk of dropping out, or common factors that influence this decision. In this context, machine learning presents itself as a powerful resource. Models can be trained on data from previous students in order to predict the likelihood of dropout [3].

In this report, we will compare the accuracy of three machine learning models on categorizing students as `enrolled`, `dropout`, or `graduate`. The dataset used corresponds to students from the Polytechnic University of Portalegre between 2008/2009 and 2018/2019 [1]. The three models were chosen to be Logistic Regression due to its simplicity, Neural Networks for its robustness, and Decision Trees for its explainability.

The outline of the report is as follows. In section II, we describe the dataset used in this project and cover the methods used: logistic regression, neural networks, and decision trees. This section is concluded with a description of our implementation, including the processing of the data set and our use of AI tools. In section III, we will present our results comparing the performance of the three chosen methods for a classification problem using the dataset. Finally, section IV summarizes our findings and presents some perspectives for future work with this type of data.

II. METHODS

A. Data

We used the data set "Predict Students' Dropout and Academic Success" [1] which was collected with the goal of predicting which students are at risk of dropping out or spending extra years to complete their degree at a Polytechnical university [3]. The data set contains 4424 Portuguese students enrolled across eleven academic years from 2008/2009 to 2018/2019 [3]. There are two publications from the original authors of the data set, one from 2021 [3] and one from 2023 [4]. These two publications include slightly different aspects of the data set which is why both publications are referenced here.

These students came from a variety of undergraduate degrees from the Polytechnic University of Portalegre. The data set has 35 features per student. These features can be divided into demographic features like age and gender, social-economic features like parent's qualifications, academic features from time of enrolment like type of degree and admission grade, academic features during the first semester like number of curricular units and average grade, and macro-economic features like GDP that year and inflation rate [4]. These features are either binary like debtor, discrete like age at enrolment, continuous like grade or nominal like occupation of parent [4].

The target values are `graduate`, `enrolled`, and `dropout`. Students in the class `graduate` completed their degree in due time, students in `enrolled` completed their degrees but used up to three additional years to do so, and students in the `dropout` category either dropped out completely or spent more than three extra years to obtain their degree [4]. The data set is not balanced, as shown in figure 1.

The goal of this data set was to see if they could predict which students were at risk of failing so that the university could aim extra resources at these students to help them complete [3]. With this goal, correctly identifying the minority classes is important.

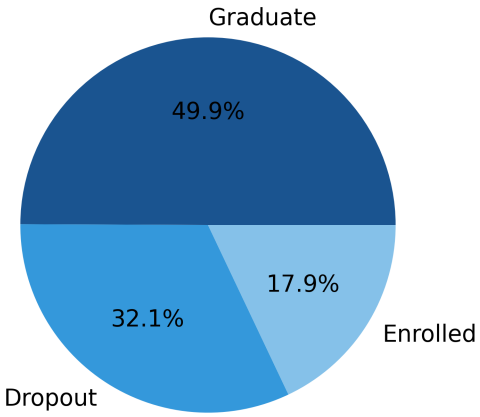

Figure 1. **Distribution of the target categories within the dataset.**

B. Logistic regression

Logistic regression is a probabilistic classification method [5]. For a binary problem it models the probability of a sample x_i belonging to a category $y_i = 0, 1$ as a logistic function:

$$p(y = 1 \mid \mathbf{x}) = \sigma(z) = \frac{1}{1 + e^{-z}}, \qquad z = \mathbf{w}^\top \mathbf{x} + b\,, \quad (1)$$

where $\mathbf{w}$ is the weight vector and b the bias. For this case, the loss function is called the binary cross-entropy [6]:

$$\ell(\mathbf{w}, b) = -\sum_{n=1}^{N} \left[y^{(n)} \log \hat{y}^{(n)} + (1 - y^{(n)}) \log(1 - \hat{y}^{(n)}) \right], \quad (2)$$

where $\hat{y}^{(n)} = \sigma(\mathbf{w}^\top \mathbf{x}^{(n)} + b)$. Regularization can be added in order to avoid overfitting [7], being the L2 penalty a common choice:

$$\ell_\lambda(\mathbf{w}, b) = \ell(\mathbf{w}, b) + \tfrac{\lambda}{2} \|\mathbf{w}\|_2^2, \quad (3)$$

where $\|\mathbf{w}\|_2^2 = \sum_j w_j^2$. Classification is performed by thresholding the predicted probability (commonly at 0.5, though the threshold can be adjusted). For multiclass problems, the softmax model is used [6]:

$$p(y = i \mid \mathbf{x}) = \frac{e^{\mathbf{w}_i^\top \mathbf{x} + b_i}}{\sum_{k=1}^{K} e^{\mathbf{w}_k^\top \mathbf{x} + b_k}}\,. \quad (4)$$

Since there is no closed-form solution for the optimal parameters, the loss function is minimized numerically through optimization methods.

C. Feed forward neural networks

A feed-forward neural network (FFNN) is a model whose basic architecture consists of an input layer, one or more hidden layers, and an output layer[7]. Each layer contains a tunable number of nodes. In a fully connected FFNN, each node in layer l receives the outputs of all nodes in layer $l - 1$ and computes

$$z_j^l = \sum_{i=1}^{n_{l-1}} w_{ji}^l \, a_i^{l-1} + b_j^l\,, \quad (5)$$

then applies an activation function:

$$a_j^l = f^l\big(z_j^l\big)\,, \quad (6)$$

where n_{l-1} is the number of units in layer $l - 1$, w_{ji}^l and b_j^l are the weights and biases, and f^l may vary by layer. Training adjusts the weights and biases to minimize a chosen loss function using optimization algorithms with gradients computed by backpropagation [8]. For classification problems, a common choice of cost function is the cross-entropy loss (Equation 2)[6]. Convergence and final performance depend on many factors, one of them being the learning rate, which defines the size of the step taken at each iteration towards the minimum of the cost function.

1. *Activation functions*

Activation functions are a crucial part of neural networks because they introduce nonlinearity into the model[9]. The choice of activation for the final layer depends on the task, while there is more flexibility for hidden layers. For classification problems, the final-layer activation should produce values that can be interpreted probabilistically (e.g., a sigmoid for binary classification or a softmax for multiclass classification). In this project, three activation functions were tested for the hidden layers - ReLU, Logistic and Tanh - and Softmax was used for the output layer. Here, we briefly introduce them.

a. ReLU One of the most commonly used activation functions is the rectified linear unit (ReLU) function, which is given by:

$$\mathrm{ReLU}\,(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \le 0\,, \end{cases} \quad (7)$$

and its derivative is:

$$\frac{d\,\mathrm{ReLU}}{dx}\,(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x < 0\,. \end{cases} \quad (8)$$

b. Logistic The logistic activation function, commonly known as sigmoid, was already discussed as part of logistic regression (See equation 1). Its derivative is commonly employed in the context of backpropagation as:

$$\frac{d\,\sigma(x)}{dx} = \sigma(x)\big(1 - \sigma(x)\big)\,. \quad (9)$$

since it uses the previously computed activation $\sigma(x)$.

c. Tanh The hyperbolic tangent (Tanh) is given by:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}\,, \tag{10}$$

and its derivative can be expressed as:

$$\frac{d\,\tanh(x)}{dx} = 1 - \tanh^2(x)\,. \tag{11}$$

d. Softmax The softmax activation function is given by 4. It returns a vector of positive values that sum to 1, which can be interpreted as a categorical probability distribution. Its derivative is:

$$\frac{\partial s_i}{\partial z_j} = s_i(\delta_{ij} - s_j), \tag{12}$$

where δ_{ij} is the Kronecker delta.

D. Decision trees

Decision tree is a quite popular method that can be applied both to regression and classification problems. Its popularity is most likely due to its interpretability, since it can be understood as a sequence of binary decisions [9].

The general structure of a tree is as follows: root node, interior nodes and final leaf nodes (also referred to as leaves)[6]. All nodes are then connected through branches. The root node receives the input, which is faced with the first test of some attribute. The number of branches leaving the root node are the possible values taken by this attribute. This process is then repeated for all interior nodes until the input reaches the leaf corresponding to its output. In the case of classification, the output is the category to which the input belongs to.

Since this is a classification problem we used the gini index as the cost function which is given by

$$g = \sum_{k=1}^{K} p_{mk}\,(1 - p_{mk})\,. \tag{13}$$

where m gives the region and p_{mk} represent the majority class of the region m. K is the number of classification classes [6]. This encourages the tree to make regions where there is a high proportion of data points given one class [9].

E. Evaluation of the models

1. Accuracy

The accuracy of a model provides the percentage of correctly classified samples. It is calculated by:

$$\text{Accuracy} = \frac{\sum_{i=0}^{n-1} I(y_i = \tilde{y}_i)}{n}\,. \tag{14}$$

2. Confusion matrix

Confusion matrices provides an overview of the model performance per category, with predictions on one axis and true labels on another [6]. The diagonal contains the true positives for each class, that is, the number of correctly labelled samples within that class. On the other hand, the off-diagonals offers some insight on how the samples were misclassified.

3. ROC curve

The Receiver Operating Characteristic (ROC) curve shows the trade-off between true positive rate and false positive rate as a decision threshold is varied [6]. For a binary classifier, the true positive rate (TPR) is given by:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}\,, \tag{15}$$

while the false positive rate (FPR) is:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}\,, \tag{16}$$

where TP are the true positive counts, FN the false negatives, FP the false positives and TN the ture negatives, all determined at the same threshold. For comparison, a random classifier would yield a ROC curve corresponding to the diagonal ($TPR = FPR$).

The area under the ROC curve (AUC) represents the performance across thresholds, a value of 1 would indicate a perfect model, and it could be interpreted as the probability that a randomly chosen positive instance receives a higher score than a randomly chosen negative one. It can be extended to multiclass problems by applying a one-vs-rest ROC curve to each category, or by computing the macro averaged AUC, given by:

$$\text{AUC}_{\text{macro}} = \frac{1}{K}\sum_{i=1}^{K} \text{AUC}_i\,, \tag{17}$$

and the micro averaged AUC, computes the TPR and FPR by combining all classes into one and reducing the multiclass problem to a binary one.

4. Cummulative gain

The cummulative gain curve is an evaluation tool for binary classifiers, but it can be also extended to multiclass problems using a one-vs-rest philosophy. It shows the fraction of the total positive instances captured when using only a fraction of the population, previously ranked by model confidence [6].

F. Implementation

1. Preprocessing of data

After analysing the data, some changes were made to the original dataset. The categories `nationality` and `application mode` were dropped. Since the vast majority of students are Portuguese, we reduced redundancy by only keeping the category `international`. Regarding `application mode`, there were too many options and it was not clear what type of information they carried.

Also, we reduced the number of options for the following categories: `Previous qualification`, `Mother's qualification`, `Father's qualification`, `Mother's occupation` and `Father's occupation`. That was done to reflect the level of qualification (Basic Education, General Course, Higher Education, Higher Education - Master or Doctorate, Higher Technical Course, Illiterate, Incomplete Basic Education, Technical Course, Technological Specialization Course, Unknown) and the type of work (Manual work, Military, Office Workers, Other, Service industry, Student), rather than being very specific about it. For details, see tables in Appendix A.

Before training, we also explored the use of ordinal and one hot encoding, as well as robust and standard scaling of some features. Categorical features such as the ones mentioned above and boolean ones (zero for false and 1 for true) were not scaled.

2. Structure of code

The training of all models were performed with the library scikit-learn [10]. The dataset was split into 80% training and 20% testing using scikit-learn's *train_test_split* with the *stratify* option in order to preserve the class proportions. The functionalities used are listed in table I.

Matplotlib [11] and seaborn [12] were used for making the figures, as well as pandas [13] and numpy [14] as auxiliary libraries.

G. Use of AI tools

GPT UiO was used to solve small problems with the code like how to ensure the class names displayed on the decision tree were correct and to properly format a table in Latex. The exported conversation can be accessed in the Github repository of the project (m-nader/FYS-STK4155-group14), inside a folder called LLM. In addition, Copilot and GPT UiO were used to debug code.

III. RESULTS AND DISCUSSION

Three classification methods were used to fit the data: Logistic Regression, Neural Networks and Decision Trees. With the first one, some tests were performed to better understand the dataset and optimize its preprocessing.

A. Logistic regression

Using the default configuration of scikit-learn's logistic regression, we performed training of the original and the adjusted data, with different preprocessing options. The accuracy scores are shown in figure 2.

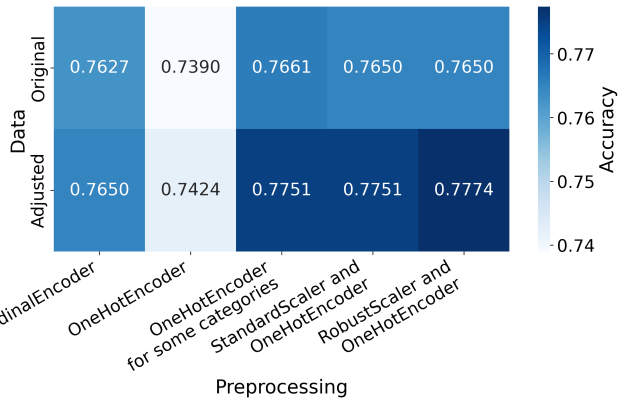

Figure 2. **Accuracies using different data preprocessing methods:** Original and adjusted data with OrdinalEncoder, OneHotEncoder applied to all categories, OneHotEncoder applied to some categories, OneHotEncoder applied to some categories with StandardScaler or RobustScaler.

It is possible to observe that the adjustments made had a positive impact in accuracy, and that the use of OneHotEnconder was only beneficial when applied to certain categories. That is most likely because, if the categories were not specified, the numerical ones as `GDP` and `admission grade` were also encoded. So, we excluded them and the ones that were already boolean. For the adjusted data, robust scaling provided an additional increase in accuracy. Therefore, all the results from now on were obtained with OneHotEncoder applied to some categories (described in the methods) and RobustScaler to others.

We also benchmarked the different options of scikit-learn's logistic regression. Figure 3 shows the accuracy using the different solvers implemented in scikit-learn (liblinear was not used since it doesn't allow for multinomial targets) and different configurations - default, using L2 penalty, setting fit_intercept equal to false and using cross-validation (CV).

From it, we notice that the inclusion of L2 penalty and the choice of solver did not produce an effect on the accuracy, and the use of cross validation actually led to a decrease in it. On the contrary, setting fit_intercept to false

Table I. **Scikit-learn functionalities used in this project.**

Type of action	Scikit-learn functionalities
Preprocessing data	`OrdinalEncoder`, `OneHotEncoder`, `StandardScaler`, `RobustScaler`, `ColumnTransformer`, `train_test_split`
Logistic Regression	`LogisticRegression`, `LogisticRegressionCV`
Neural Networks	`MLPClassifier`
Decision Tree	`tree` (e.g. `DecisionTreeClassifier`)
Cross-validation	`cross_val_score`
Analysis	`accuracy_score`, `confusion_matrix`, `label_binarize`

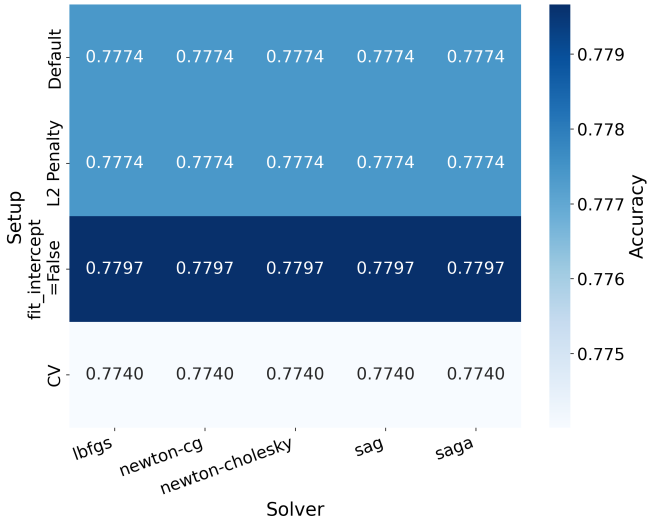

Figure 3. **Accuracies using different options within scikit-learn's logistic regression method:** Default options, applying L2 penalty, setting fit_intercept equal to false and using cross-validation (CV). Training was done based on the adjusted data, with OneHotEncoder for some categories and RobustScaler for others.

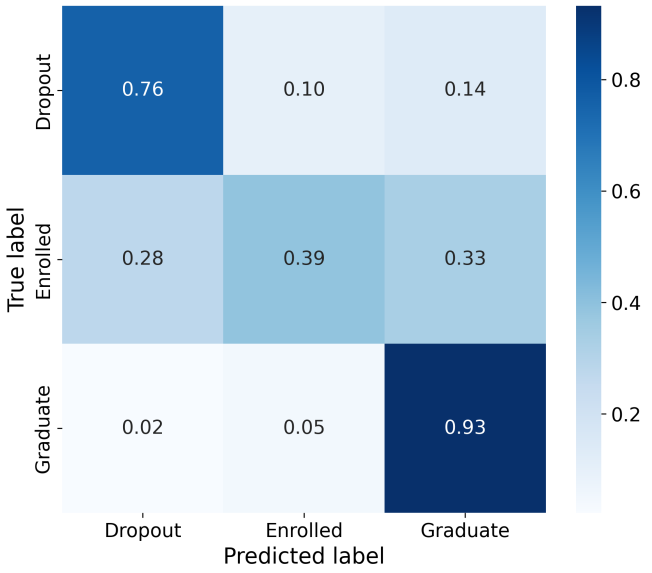

Figure 4. **Confusion matrix of logistic regression:** Best performing logistic regression model with newton-cg solver, fit_intercept equal to false, adjusted data, OneHotEncoder for some categories and RobustScaler for others.

led to a slight improvement, which could be explained by the fact that the data was previously scaled. Figure 4 displays the confusion matrix for one of the models with the highest accuracy, using newton-cg as the solver.

The `enrolled` category had the lowest accuracy among the three, with only 38% being correctly assigned to it, while almost all `graduate` instances were correctly classified. This is consistent with the dataset being imbalanced, since these classes have the lowest and highest representation, respectively. For the same model, we also plotted the ROC curve 5 and the cumulative gain for each category (See figures 14,15 and 16 in Appendix A). Again, it is clear that the `enrolled` category is the worst when it comes to model accuracy, but the model is still performs better than a random classifier (Dotted black line).

B. Neural network

We applied a neural network to see if we could get better results than logistic regression. We used the same processing as with the logistic regression model, with OneHotEncoder on some features and RobustScaler on others. To start, we tuned the learning rate and the L2-regularisation hyperparameter (λ) with mostly default settings for initialising the neural network. A variety of learning rates and L2-hyperparameters led to a similar performance (accuracies between 75% and 77%) as shown in Figure 6.

As the figure shows, two learning rates looked like good choices with multiple L2-hyperparameters. We kept them constant at $\eta = 0.0001$ and $\lambda = 0.001$.

When we tested out a variety of neural network sizes (Figure 7), there was no clear result on a deeper or a wider network being best for this data. As three hidden layers with 50 neurons per layer and one hidden layer with 150 neurons performed well out of the tested net-

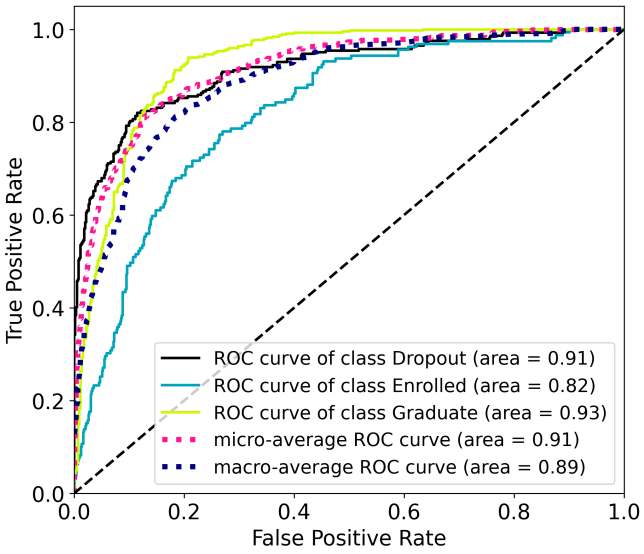

Figure 5. **ROC curve for the best performing logistic regression model:** adjusted data with OneHotEncoder for some categories and RobustScaler for others, newton-cg as the solver and fit intercept equal to false.

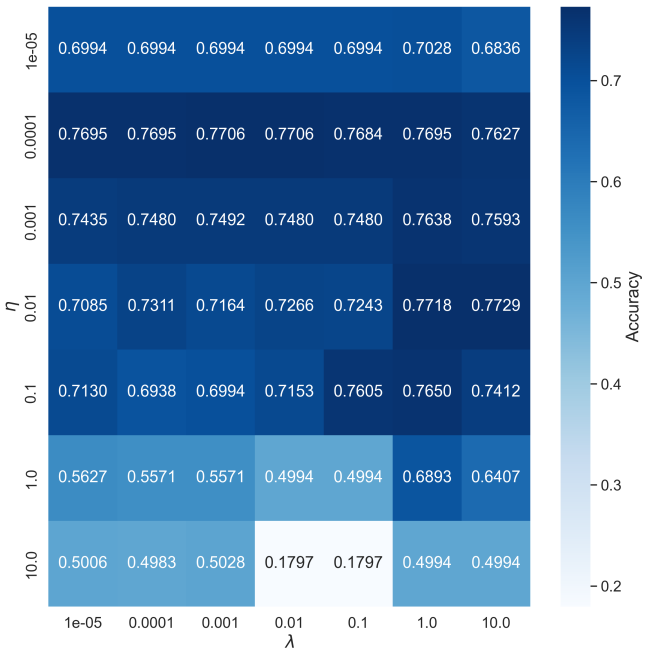

Figure 6. **Accuracies using different learning rates and L2-hyperparameters:** Accuracies on the test set. NN using Adam as the solver, one hidden layer with 200 neurons and ReLU as the activation function.

works, we chose a middle size network - two hidden layers with 100 neurons - that had a similar performance as a staring point to test different activation functions for the hidden layers (Figure 8).

Based on figure 8, we are able to see that the logistic activation function was the worst. Also, the number of

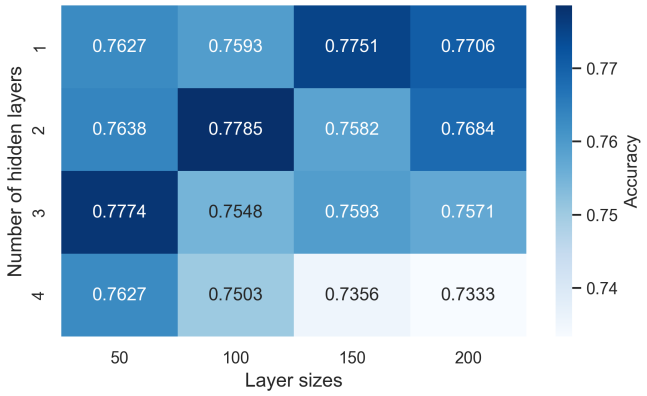

Figure 7. **Accuracies using different number of layers and neurons per hidden layer:** With a set learning rate of 0.0001 and L2-hyperparameter of 0.001, and ReLU as the activation function.

neurons that led to a better performance was dependent on the activation function, for ReLU the best number of neurons was 100 and for tanh the best number of neurons was 50. The last one performed best out of these nine. We used this set up when testing for best solver (See Figure 13 in Appendix A), which was the default one - ADAM.

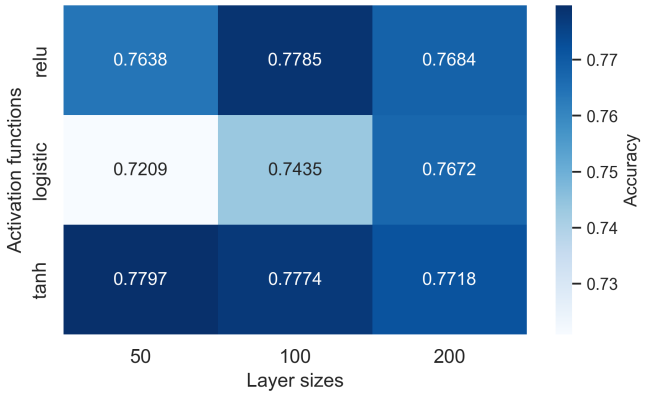

Figure 8. **Accuracies using different activation functions and varying the number of neurons in the two hidden layers:** With a set learning rate of 0.0001 and L2-hyperparameter of 0.001.

As a summary, the neural network with the highest accuracy had this architecture: 2 hidden layers of 50 neurons each, a learning rate of 0.0001, L2-hyperparameter of 0.001, tanh as activation function and ADAM solver. We plotted the confusion matrix for it in Figure 9.

Similarly as our best logistic regression model, this neural network struggles with correctly identifying `enrolled`. Again, only 38 % of the students in this category get classified correctly. 93 % of `graduate` gets categorised correctly with a couple of percent getting misclassified. `Dropout` gets classified correctly 76 % of the time.

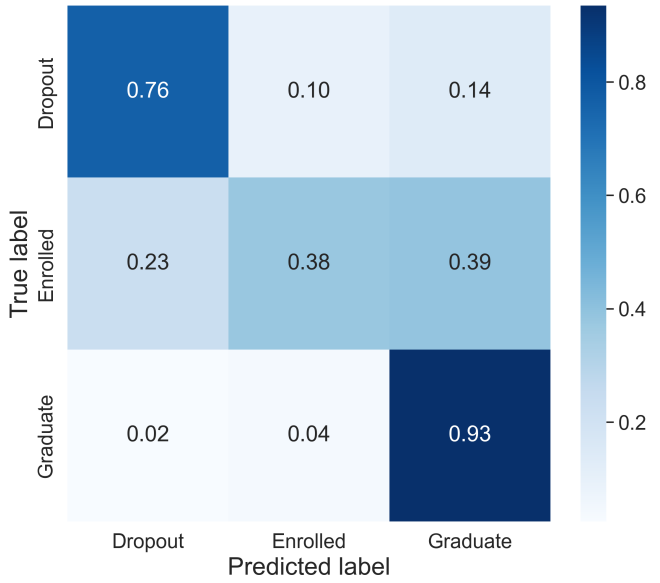

Figure 9. **Confusion matrix of NN:** Best performing NN model with 2 hidden layers of 50 neurons each, a learning rate of 0.0001, L2-hyperparameter of 0.001, tanh as activation function and solver ADAM.

C. Decision tree

Decision trees can be easier to understand for us humans [5] since we can plot the decisions the model has made after learning from the training data. From early explorations of the decision tree, we discovered that the default settings from sklearn would overfit the data by making very large trees. To prevent overfitting, we tuned for the best performing depth as shown in Table II.

Table II. **Decision tree training and test accuracies.** Accuracies of decision trees with various maximum depths.

Depth	Test accuracy	Train accuracy
1	0.706	0.705
2	0.716	0.714
3	0.733	0.741
4	0.736	0.750
5	0.757	0.775
6	0.731	0.790
7	0.732	0.820
⋮	⋮	⋮
22	0.677	1.000

The table shows the deep trees overfit, since the accuracy on the test set starts to decrease as the accuracy on the train set reaches perfection. This results in a very large tree with few samples per final leaf node. One way to prevent overfitting is to limit the number of final leaf nodes. We tested out various values for the maximum number of leaf nodes but this did not lead to a better tree. Deeper trees with a limited number of final leaf nodes did not perform better than the tree with depth 5. Even limiting leaf nodes on the depth 5 tree did not change the performance from 0.757. The confusion matrix on figure 10 was obtained for a decision tree with depth 5 and a maximum of 20 final leaf nodes.

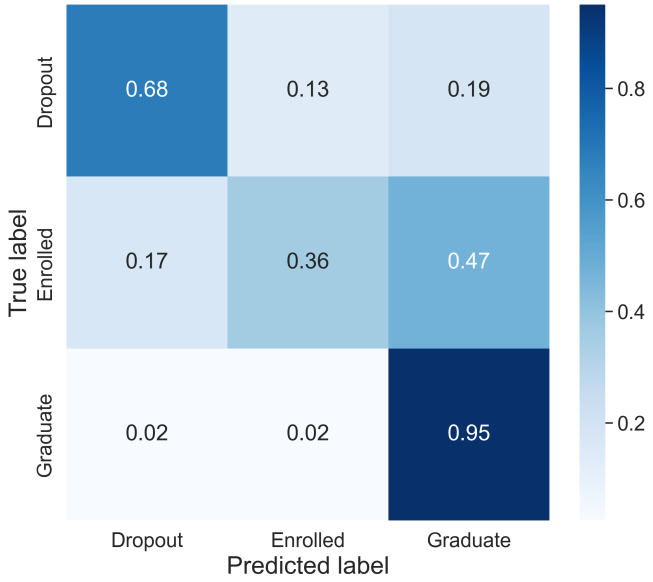

Figure 10. **Confusion matrix of decision tree** Best performing decision tree model with depth 5 and a maximum of 20 final leaf nodes.

As with our other models, `enrolled` has the lowest accuracy and `graduate` has the best. 47% of the students in `enrolled` are misclassified as `graduate`.

Since decision trees are so called white box models, we can both plot the tree and access the importance of the different features. Figure 11 shows the 15 most important features.

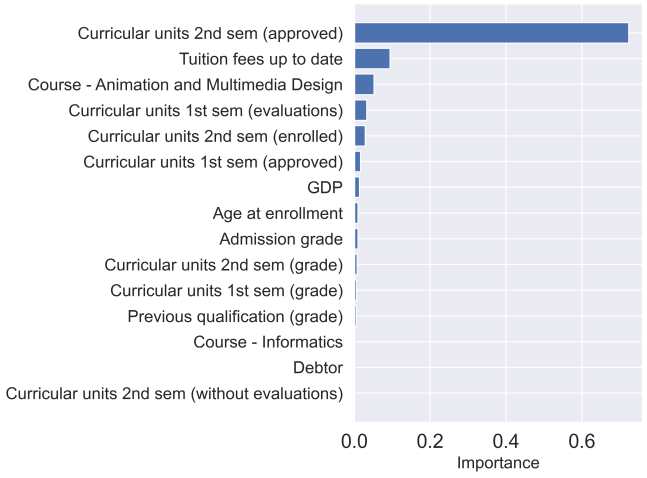

Figure 11. **Feature importance** The 15 most important features for the decision tree with depth 5.

Many of these features relate to curricular units the

students have taken or are going to take. This follows from what we know of the data set. The students who drop out before the first semester do not take any courses for the second semester and most who drop out do this during the second semester [4]. This last fact could explain why the curricular units 2nd semester is so important.

Other important features are previous grades from their undergraduate degrees and admission grades. In Norway, older students drop out at a higher rate than younger students [15]. Age at enrolment being important matches with these results from a different European country. Economic factors like macro-economic GDP and micro-economic ones, like Tuition fees up to date and Debtor, are important features. This is supported by literature on completion factor for students: Tinto (1993) [16] write that economic factors influence students choice to drop out of higher education. GPD will affect cohorts and the financial crisis of 2008 is present in this data for the year 2008/2009 and following years. Other research using this data set also found that curricular units in the 2nd semester were important features, along with tuition fees up to date and age at enrolment [4].

Classic factors like parent's occupations and qualification are not ranked amongst the most important features. As in ref. [15], these often correlate with academic success. One explanation for why our model does not consider them important may be the OneHotEncoder splitting these into several features for our redefined larger groups and that no single occupation or education level has had enough importance to appear in the top 15 most important features.

Two surprising features in this list are the two courses Animation and Multimedia Design and Informatics. To see how these features and the others are used in the model we have to plot the tree.

Even though the best model had a depth of 5, we plot a decision tree with depth 3 to make it possible to read it. The tree with depth 5 is still pretty large. The tree with depth 3 can be found in Figure 12. Here, the branches to the left are if the condition is true and the ones to the right are if the condition is false. At the first node, instances with a scaled value of less than -0.125 for the feature `Curricular units 2nd semester (approved)` followed the path to the left, the others to the right.

Each node highlights a couple of important pieces of information, such as the total number of instances left in it and how these are distributed across the target classes. The colours show which target class is most common of the test instances left in each node. Higher saturation in colour means that the model is more sure of the target class, while white means it is unsure. Ideally, each leaf node has a strong saturation. The colours are purple for `graduate`, green for `enrolled`, and orange for `dropout`. In general, the tree splits `dropout` to the left and `graduate` to the right.

The plot of the tree shows how the courses might appear in a tree. In the last layer, we get Course 171 which is Animation and Multimedia Design. For this node, if the student is from Animation and Multimedia Design, they are less likely to drop out. This leaf is white and has many false positives for `graduate`, but it manages to find all of the remaining `graduate` instances from the node above it. According to the model, students with few curricular units 2nd semester approved are less likely to drop out if they study Animation and Multimedia Design.

For students with many units approved in their second semester, whether or not they are up to date on their tuition fees can predict whether they will graduate. If the tuition fees are up to date, the model predicts they will graduate. But if the tuition fees are not up to date and GDP is low, the student will drop out. If GDP is high, the student will spend up to three years extra before graduating. Economic factors are a common reason students give for dropping out [16] and if a student is behind on tuition fees they may be reconsidering whether they are on the correct path or not.

D. Comparing our models

The best logistic regression model in this project ties with the best neural network model, both with 77.97 % accuracy, which exceed the accuracy achieved by the best decision tree model (75,70 %). The original research with this data used Logistic Regression (accuracy = 0.61), Support Vector Machine (accuracy = 0.60), Decision Tree (accuracy = 0.65), and Random Forest (accuracy = 0.72) [3]. All of our models outperform these results based on accuracy. We used the recommended data split which was 80 % for training and 20 % for testing [1]. However, we introduced some data preprocessing which could explain the increase in accuracy, such as combining multiple labels for several features as explained in the implementation subsection.

All models struggle to classify the `enrolled` class correctly, consistent with previous work [3, 4]. Both the neural network and the decision tree more often confuse `enrolled` students with `graduate` (Figures 9 and 10). The overall performance of logistic regression on this class is not greatly improved, but it tends to misclassify `enrolled` students slightly differently: it assigns a larger fraction of `enrolled` cases to `dropout`, while the neural network assigns more to `graduate`. From an institutional perspective, mislabeling an `enrolled` student as `dropout` is preferable to mislabeling them as `graduate`, since the former better highlights the risk of extended study. These distributions are close to a third split, but still better than randomly classifying `enrolled`. The original research using this data also struggled categorising enrolled correctly [3], and so did the follow up study [4]. However, they applied some methods to balance the data set and achieved better categorisations of `enrolled`.

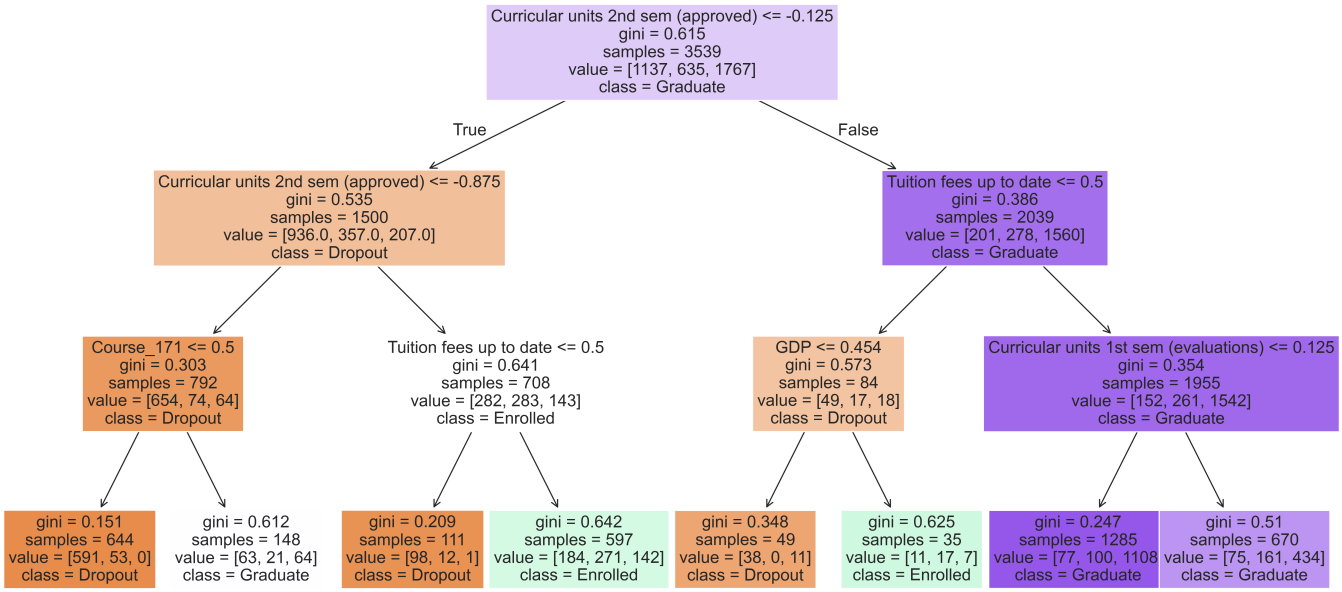

Figure 12. **Decision tree with depth 3** Visualisation from sklearn of a decision tree.

IV. CONCLUSION

By processing the data, which involved combining some of the categories in the difficult features and removing two features, we were able to achieve a higher accuracy than previous analysis of the same data set. Even though our logistic regression model and neural network tie for accuracy, we conclude that, for this data, the logistic regression model is the best because the neural network model misclassified `enrolled` into `graduate` at a higher rate than the logistic regression model. The original goal in gathering data set was to help higher education institutions identify students which may drop out or spend longer time on their degrees. When these students are identified, the university can spend more resources on them and potentially help them complete. Therefore, the best model for this task is logistic regression because it achieved a higher true positive rate for the `enrolled` category.

The decision tree performs pretty well and has the benefit of being interpretable. For the university, it is probably useful to get some insight into which factors the model learned as significant in the `dropout` category. Institutions that do not have access to such rich data about each student could instead look at the most important features and apply measures accordingly at their institution. The tree with depth 3 shows that, for students with few approved subjects in the second semester, the students from the animation course are more likely to graduate on time. It might be worth looking into the teaching there, if it is particularly good or if there are other factors like intrinsic motivation within this group that favours this outcome.

Tinto (1993) [16] highlights that there are institutional aspects that affect student drop out rates. The only feature which ties to these institutional aspects is the course the students are taking. But this does not include other aspects like contact between students and faculty or the students social experience on campus, which students themselves claim to have an influence on their choice whether or not to stay [16]. Institutional changes could be applied to reduce dropout rates without analysing every student with one of these models.

However, there are several limitations to our work. The data set is unbalanced and we only applied one measure against this: ensuring that the *train_test_split* kept the same ratio between the classes as the original data set. Previous work on this data applied the following measures: SMOTE, Balanced Random Forest classifier and East Ensemble classifier [4]. Future work should apply sampling measures to deal with the unbalanced data set. Also, we believe that training a Random forest could provide a better insight into feature importance and could, as they often do, perform better than decision trees alone [6].

The purpose of the original collection and analysis of the data was to make models to help Polytechnical universities to identify students who are at risk of dropping out or spending extra years obtaining their degrees [4]. With this in mind, decision trees are useful to include because of their white box nature. The institutions may want to identify factors which affect student the risk of student drop out. But since both the nerual network and logistic regression performed better, these are best

suited for indentifying individual students who are at risk of dropping out or spending extra time to complete their studies.

[1] V. Realinho, M. V. Martins, J. Machado, and L. Baptista, Predict Students' Dropout and Academic Success, UCI Machine Learning Repository (2021), DOI: https://doi.org/10.24432/C5MC89.

[2] N. M. of Education and Research, Finansiering av universiteter og høyskoler (2022).

[3] M. V. Martins, D. Tolledo, J. Machado, L. M. T. Baptista, and V. Realinho, Early prediction of student's performance in higher education: A case study, in *Trends and Applications in Information Systems and Technologies*, edited by Á. Rocha, H. Adeli, G. Dzemyda, F. Moreira, and A. M. Ramalho Correia (Springer International Publishing, Cham, 2021) pp. 166–175.

[4] M. V. Martins, L. Baptista, J. Machado, and V. Realinho, Multi-class phased prediction of academic performance and dropout in higher education, Applied Sciences **13**, 10.3390/app13084702 (2023).

[5] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition. Springer Series in Statistics* (Springer, New York, 2009).

[6] M. Hjorth-Jensen, Applied data analysis and machine learning (2023).

[7] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016) `http://www.deeplearningbook.org`.

[8] M. A. Nielsen, *Neural networks and deep learning*, Vol. 25 (Determination press San Francisco, CA, USA, 2015).

[9] C. M. Bishop, *Pattern recognition and machine learning by Christopher M. Bishop*, Vol. 400 (Springer Science+ Business Media, LLC Berlin, Germany:, 2006).

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research **12**, 2825 (2011).

[11] J. D. Hunter, Matplotlib: A 2d graphics environment, Computing in Science & Engineering **9**, 90 (2007).

[12] M. L. Waskom, seaborn: statistical data visualization, Journal of Open Source Software **6**, 3021 (2021).

[13] T. pandas development team, pandas-dev/pandas: Pandas (2020).

[14] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, Array programming with NumPy, Nature **585**, 357 (2020).

[15] H. Hølleland, P. Birkeland, M. Helle, and Åsgeir Kjetland Rabben, Mangfold og ulikhet i høyere utdanning (2024).

[16] V. Tinto, *Leaving college : rethinking the causes and cures of student attrition*, 2nd ed. (The University of Chicago Press, Chicago, 1993).

Appendix A

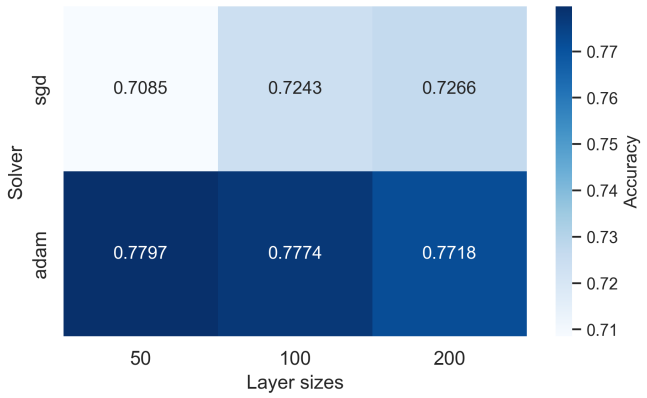

Figure 13. **Accuracies using different solvers and neurons per hidden layer:** With a set learning rate of 0.0001 and L2-hyperparameter of 0.001, tanh as the activation function and two hidden layers.

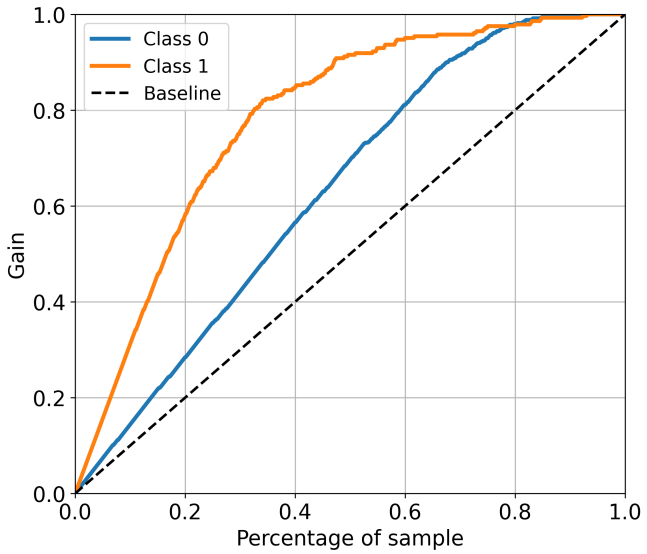

Figure 14. **Cumulative gain plot for the `dropout` category:** For the best performing logistic regression model.

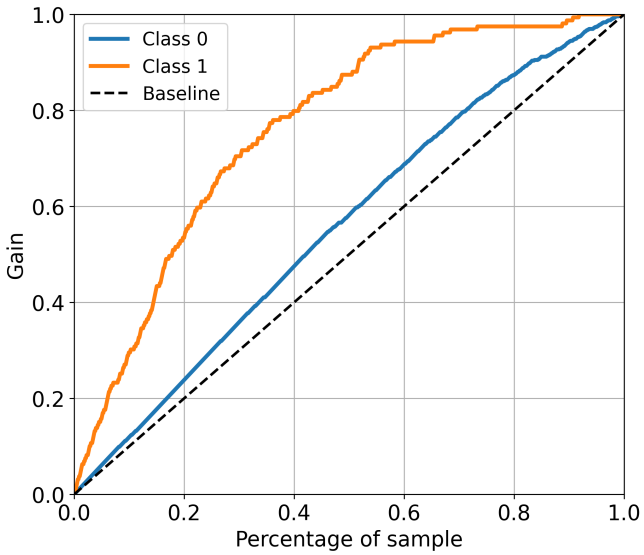

Figure 15. **Cumulative gain plot for the `enrolled` category:** For the best performing logistic regression model.

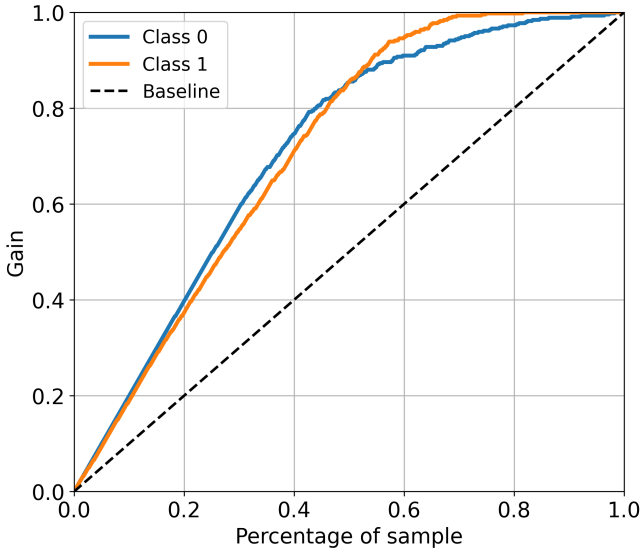

Figure 16. **Cumulative gain plot for the `graduate` category:** For the best performing logistic regression model.

Table III. **Mapping of category `Previous qualification` from original to grouped labels.**

Original label	Grouped label
1 — Secondary education	Basic Education
2 — Higher education - bachelor's degree	Higher Education
3 — Higher education - degree	Higher Education
4 — Higher education - master's	Higher Education - Master or Doctorate
5 — Higher education - doctorate	Higher Education - Master or Doctorate
6 — Frequency of higher education	Higher Education
9 — 12th year of schooling - not completed	Incomplete Basic Education
10 — 11th year of schooling - not completed	Incomplete Basic Education
12 — Other - 11th year of schooling	Incomplete Basic Education
14 — 10th year of schooling	Incomplete Basic Education
15 — 10th year of schooling - not completed	Incomplete Basic Education
19 — Basic education 3rd cycle (9th/10th/11th year) or equiv.	Incomplete Basic Education
38 — Basic education 2nd cycle (6th/7th/8th year) or equiv.	Incomplete Basic Education
39 — Technological specialization course	Technological Specialization Course
40 — Higher education - degree (1st cycle)	Higher Education
42 — Professional higher technical course	Higher Technical Course
43 — Higher education - master (2nd cycle)	Higher Education - Master or Doctorate

Table IV. **Mapping of category `Mother's qualification` from original to grouped labels.**

Original label	Grouped label
1 — Secondary Education - 12th Year of Schooling or Eq.	Basic Education
2 — Higher Education - Bachelor's Degree	Higher Education
3 — Higher Education - Degree	Higher Education
4 — Education - Master's	Higher Education - Master or Doctorate
5 — Higher Education - Doctorate	Higher Education - Master or Doctorate
6 — Frequency of Higher Education	Higher Education
9 — 12th Year of Schooling - Not Completed	Incomplete Basic Education
10 — 11th Year of Schooling - Not Completed	Incomplete Basic Education
11 — 7th Year (Old)	Incomplete Basic Education
12 — Other - 11th Year of Schooling	Incomplete Basic Education
14 — 10th Year of Schooling	Incomplete Basic Education
18 — General commerce course	General Course
19 — Basic Education 3rd Cycle (9th/10th/11th Year) or Equiv.	Incomplete Basic Education
22 — Technical-professional course	Technical Course
26 — 7th year of schooling	Incomplete Basic Education
27 — 2nd cycle of the general high school course	Basic Education
29 — 9th Year of Schooling - Not Completed	Incomplete Basic Education
30 — 8th year of schooling	Incomplete Basic Education
34 — Unknown	Unknown
35 — Can't read or write	Illiterate
36 — Can read without having a 4th year of schooling	Incomplete Basic Education
37 — Basic education 1st cycle (4th/5th year) or equiv.	Incomplete Basic Education
38 — Basic Education 2nd Cycle (6th/7th/8th Year) or Equiv.	Incomplete Basic Education
39 — Technological specialization course	Technological Specialization Course
40 — Higher education - degree (1st cycle)	Higher Education
41 — Specialized higher studies course	Higher Education
42 — Professional higher technical course	Higher Technical Course
43 — Higher Education - Master (2nd cycle)	Higher Education - Master or Doctorate
44 — Higher Education - Doctorate (3rd cycle)	Higher Education - Master or Doctorate

Table V. **Mapping of category `Father's qualification` from original to grouped labels.**

Original label	**Grouped label**
1 — Secondary Education - 12th Year of Schooling or Eq.	Basic Education
2 — Higher Education - Bachelor's Degree	Higher Education
3 — Higher Education - Degree	Higher Education
4 — Higher Education - Master's	Higher Education - Master or Doctorate
5 — Higher Education - Doctorate	Higher Education - Master or Doctorate
6 — Frequency of Higher Education	Higher Education
9 — 12th Year of Schooling - Not Completed	Incomplete Basic Education
10 — 11th Year of Schooling - Not Completed	Incomplete Basic Education
11 — 7th Year (Old)	Incomplete Basic Education
12 — Other - 11th Year of Schooling	Incomplete Basic Education
13 — 2nd year complementary high school course	Basic Education
14 — 10th Year of Schooling	Incomplete Basic Education
18 — General commerce course	General Course
19 — Basic Education 3rd Cycle (9th/10th/11th Year) or Equiv.	Incomplete Basic Education
20 — Complementary High School Course	Basic Education
22 — Technical-professional course	Technical Course
25 — Complementary High School Course - not concluded	Basic Education
26 — 7th year of schooling	Incomplete Basic Education
27 — 2nd cycle of the general high school course	Basic Education
29 — 9th Year of Schooling - Not Completed	Incomplete Basic Education
30 — 8th year of schooling	Incomplete Basic Education
31 — General Course of Administration and Commerce	General Course
33 — Supplementary Accounting and Administration	General Course
34 — Unknown	Unknown
35 — Can't read or write	Iliterate
36 — Can read without having a 4th year of schooling	Incomplete Basic Education
37 — Basic education 1st cycle (4th/5th year) or equiv.	Incomplete Basic Education
38 — Basic Education 2nd Cycle (6th/7th/8th Year) or Equiv.	Incomplete Basic Education
39 — Technological specialization course	Technological Specialization Course
40 — Higher education - degree (1st cycle)	Higher Education
41 — Specialized higher studies course	Higher Education
42 — Professional higher technical course	Higher Technical Course
43 — Higher Education - Master (2nd cycle)	Higher Education - Master or Doctorate
44 — Higher Education - Doctorate (3rd cycle)	Higher Education - Master or Doctorate

Table VI. **Mapping of category `Mother's occupation` from original to grouped labels.**

Original label	**Grouped label**
0 — Student	Student
1 — Representatives of the Legislative Power and Executive Bodies, Directors, Directors and Executive Managers	Office Workers
2 — Specialists in Intellectual and Scientific Activities	Office Workers
3 — Intermediate Level Technicians and Professions	Manual work
4 — Administrative staff	Office Workers
5 — Personal Services, Security and Safety Workers and Sellers	Service industry
6 — Farmers and Skilled Workers in Agriculture, Fisheries and Forestry	Manual work
7 — Skilled Workers in Industry, Construction and Craftsmen	Manual work
8 — Installation and Machine Operators and Assembly Workers	Manual work
9 — Unskilled Workers	Manual work
10 — Armed Forces Professions	Military
90 — Other Situation	Other
99 — (blank)	Other
122 — Health professionals	Service industry
123 — teachers	Service industry
125 — Specialists in information and communication technologies (ICT)	Office Workers
131 — Intermediate level science and engineering technicians and professions	Office Workers
132 — Technicians and professionals, of intermediate level of health	Service industry
134 — Intermediate level technicians from legal, social, sports, cultural and similar services	Service industry
141 — Office workers, secretaries in general and data processing operators	Office Workers
143 — Data, accounting, statistical, financial services and registry-related operators	Office Workers
144 — Other administrative support staff	Office Workers
151 — personal service workers	Service industry
152 — sellers	Service industry
153 — Personal care workers and the like	Service industry
171 — Skilled construction workers and the like, except electricians	Manual work
173 — Skilled workers in printing, precision instrument manufacturing, jewelers, artisans and the like	Manual work
175 — Workers in food processing, woodworking, clothing and other industries and crafts	Manual work
191 — cleaning workers	Manual work
192 — Unskilled workers in agriculture, animal production, fisheries and forestry	Manual work
193 — Unskilled workers in extractive industry, construction, manufacturing and transport	Manual work
194 — Meal preparation assistants	Manual work

Table VII. **Mapping of category `Father's occupation` from original to grouped labels.**

Original label	**Grouped label**
0 — Student	Student
1 — Representatives of the Legislative Power and Executive Bodies, Directors, Directors and Executive Managers	Office Workers
2 — Specialists in Intellectual and Scientific Activities	Office Workers
3 — Intermediate Level Technicians and Professions	Manual work
4 — Administrative staff	Office Workers
5 — Personal Services, Security and Safety Workers and Sellers	Service industry
6 — Farmers and Skilled Workers in Agriculture, Fisheries and Forestry	Manual work
7 — Skilled Workers in Industry, Construction and Craftsmen	Manual work
8 — Installation and Machine Operators and Assembly Workers	Manual work
9 — Unskilled Workers	Manual work
10 — Armed Forces Professions	Military
90 — Other Situation	Other
99 — (blank)	Other
101 — Armed Forces Officers	Military
102 — Armed Forces Sergeants	Military
103 — Other Armed Forces personnel	Military
112 — Directors of administrative and commercial services	Office Workers
114 — Hotel, catering, trade and other services directors	Service industry
121 — Specialists in the physical sciences, mathematics, engineering and related techniques	Office Workers
122 — Health professionals	Service industry
123 — teachers	Service industry
124 — Specialists in finance, accounting, administrative organization, public and commercial relations	Office Workers
125 — Specialists in information and communication technologies (ICT)	Office Workers
131 — Intermediate level science and engineering technicians and professions	Office Workers
132 — Technicians and professionals, of intermediate level of health	Service industry
134 — Intermediate level technicians from legal, social, sports, cultural and similar services	Service industry
135 — Information and communication technology technicians	Office Workers
141 — Office workers, secretaries in general and data processing operators	Office Workers
143 — Data, accounting, statistical, financial services and registry-related operators	Office Workers
144 — Other administrative support staff	Office Workers
151 — personal service workers	Service industry
152 — sellers	Service industry
153 — Personal care workers and the like	Service industry
154 — Protection and security services personnel	Service industry
161 — Market-oriented farmers and skilled agricultural and animal production workers	Manual work
163 — Farmers, livestock keepers, fishermen, hunters and gatherers, subsistence	Manual work
171 — Skilled construction workers and the like, except electricians	Manual work
172 — Skilled workers in metallurgy, metalworking and similar	Manual work
174 — Skilled workers in electricity and electronics	Manual work
175 — Workers in food processing, woodworking, clothing and other industries and crafts	Manual work
181 — Fixed plant and machine operators	Manual work
182 — Assembly workers	Manual work
183 — Vehicle drivers and mobile equipment operators	Manual work
191 — cleaning workers	Manual work
192 — Unskilled workers in agriculture, animal production, fisheries and forestry	Manual work
193 — Unskilled workers in extractive industry, construction, manufacturing and transport	Manual work
194 — Meal preparation assistants	Manual work
195 — Street vendors (except food) and street service providers	Service industry