Report at the University of Porto

Study on alcohol consumption among students in Porto

Chair:	Department of Informatics Engineering University of Porto
Supervisor:	João Pedro Carvalho Leal Mendes Moreira
Study Program:	Master in Engineering and Data Science
Submitted by:	Karim Kousa, Philipp Merz 202102687, 202101811
Submission date:	02.01.2022

Table of Contents

Table of Contents....ii
List of Figures....iii
List of Tables....iv
1 Business understanding....5
2 Data Understanding....7
3 Data Preparation....11
4 Modelling....13
5 Evaluation....15
Appendix A....18

List of Figures

Figure 1: Histograms of the numerical variables....8
Figure 2: Distributions of the binomial variables....9
Figure 3: Correlation Matrix of the attributes....10
Figure 4: High-level layout of the model....11
Figure 5: Subprocess layout for data preparation....12
Figure 6: Cross-Validation layout for Neural Net-, AutoMLP-, Deep Learning-Algorithms....14
Figure 7: Cross-Validation layout for k-NN-, Naïve Bayes-, Decision Tree-, and Random Forest-Algorithms....14
Figure 8: Neural Net algorithm with and without data preprocessing....17

List of Tables

Table 1: Attributes used in the study 5
Table 2: Summary Statistics of the numerical variables 8
Table 3: Comparison of the predictive algorithms without normalization 16
Table 4: Comparison of the predictive algorithms with normalization 16

1 Business understanding

According to the Global Status Report on Alcohol and Health from 2018 (published by the World Health Organization (WHO)) the average consumption of pure alcohol by citizens aged 15 or older in Portugal was 12.3 litres. A large gap between males and females (20.5 and 5.1 litres) can be observed and we can also conclude that the consumption is above the European average of 9.8 litres. Further, the prevalence of heavy episodic drinking among students (15-19 years old) is reported to be 46.9 % for males and 12.8% for females. In this context, heavy episodic drinking refers to a consumption of at least 60 grams or more of pure alcohol on at least one occasion in the past 30 days. However, the reasons of the differences in the drinking behaviour are not explained by the WHO report. Therefore, we investigate the factors that influence the consumption of alcohol among students in Portugal in this report and create a model that is able to classify the alcohol consumption on workdays based on selected attributes into 5 categories. More specific, we propose a model that forecasts the workday alcohol consumption of students based on the attributes sex, age, family size, parental status, study time, number of failures in school, internet connection, higher school education, family relationship, free time, go out behaviour, number of absences in school and the final grade. Further, we are interested in how this pattern changes in comparison to weekend alcohol consumption. An overview of the attributes and further explanations can be found in Table 1. We divide the dataset into a training and validation set and repeat the process several times in order to obtain reliable results.

We build our analysis on a dataset from Porto, Portugal where about 400 students of a Secondary School were surveyed. The complete dataset can be downloaded from this website https://www.kaggle.com/uciml/student-alcohol-consumption.

Table 1: Attributes used in the study

Attribute	Description
Sex	Student's sex (binomial)
Age	Student's age (numeric: from 15 to 22)
Family size	Family size (binomial: ≤ 3 or > 3)
Parental status	Parent's cohabitation status (binomial: living together or living apart)

Study time	Weekly study time (numeric, grouped into 4 intervals: 0 – 2, 2 – 5, 5 – 10, 10+ hours)
Number of failures	Number of past class failures (numeric)
Internet connection	Internet access at home (binomial: yes or no)
Higher school education	Wants to take higher education (binomial: yes or no)
Family relationship	Quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
Free time	Free time after school (numeric: from 1 - very low to 5 - very high)
Go out behaviour	Going out with friends (numeric: from 1 - very low to 5 - very high)
Number of absences	Number of school absences (numeric)
Final grade	Final grade (numeric: from 0 to 20)
Workday alcohol consumption	Workday alcohol consumption (numeric: from 1 - very low to 5 - very high, target attribute)
Weekend alcohol consumption	Weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)

2 Data Understanding

In this section we provide first insights of the data that we use and describe our strategy for dealing with outliers and missing values. For this purpose we ran a python script on a Python 3.9 machine to visualize the distributions of our attributes. The file can be found in the Appendix A. Further, we used RapidMiner Studio 9.10 to obtain an overview on the dataset. Overall, the dataset comprises 395 rows and 33 columns. Each row includes the answers of one student to the survey that was conducted in the mathematics course of the two secondary schools Gabriel Pereira and Mousinho da Silveira in Porto in 2008. A simple query in the python script confirms that the dataset does not have any missing values which is why we can use the complete dataset. The data has many different variable types that are nominal, ordinal, binomial or numerical. Some examples are shown in the following. Note that we did not use all 33 attributes for our study.

- Nominal: student's sex, school, address, etc.
- Ordinal: workday alcohol consumption, free time, etc.
- Binomial: internet access, family size, etc.
- Numerical: age, number of absences, etc.

We can observe that many variables were categorized by the creators of the survey. For instance, the student's study time has the four categories < 2 hours, 2 to 5 hours, 5 to 10 hours, and >10 hours. The same holds for our target attributes workday alcohol consumption and weekend alcohol consumption, which is why we deal with a classification problem that is not binary. To obtain further insights on the data we can look at the distributions of the attributes that we selected for our study and that can be found in Table 1. The distributions of the numerical variables and those that can be grouped in categories are shown as histograms in Figure 1. We can observe that most of the attributes show an expected distribution and cannot visually identify any outliers. Nevertheless, when focusing on the number of absences the relatively large scale may indicate an outlier. However, we will address this at a later point. An interesting conclusion can be drawn from the alcohol consumption of students. Here, we observe that on weekends the distribution of each category is close to a uniform distribution while during the week only very few students drink alcohol more regularly. It indicates that some students show a very different drinking behaviour during the week and on the weekend.

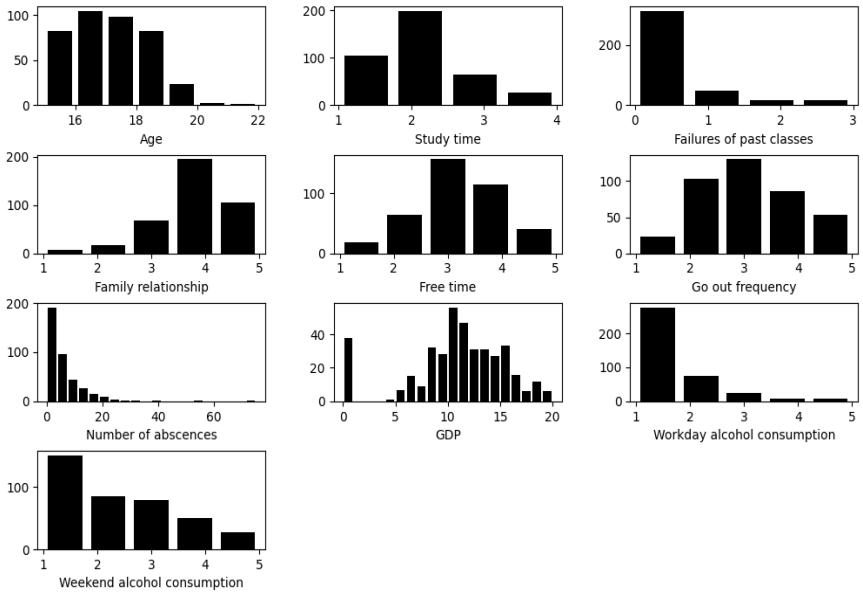

Figure 1: Histograms of the numerical variables

Table 2 reports the summary statistics of all numerical variables. We can observe that the scales of the attributes are very different, such that for example Absences ranges from 0 to 75, whereas No. of Failures has only values between 0 and 3. Therefore, a normalization is needed before applying the prediction algorithms.

Table 2: Summary Statistics of the numerical variables

	Age	Study time	No. of failures	Family relation	Free time	Go out frequency	Absences	GDP	Workday alcohol consumption	Weekend alcohol consumption
count	395.00	395.00	395.00	395.00	395.00	395.00	395.00	395.00	395.00	395.00
mean	16.70	2.04	0.33	3.94	3.24	3.11	5.71	10.42	1.48	2.29
std	1.28	0.84	0.74	0.90	1.00	1.11	8.00	4.58	0.89	1.29
min	15.00	1.00	0.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00
25%	16.00	1.00	0.00	4.00	3.00	2.00	0.00	8.00	1.00	1.00
50%	17.00	2.00	0.00	4.00	3.00	3.00	4.00	11.00	1.00	2.00
75%	18.00	2.00	0.00	5.00	4.00	4.00	8.00	14.00	2.00	3.00
max	22.00	4.00	3.00	5.00	5.00	5.00	75.00	20.00	5.00	5.00
median	17.00	2.00	0.00	4.00	3.00	3.00	4.00	11.00	1.00	2.00
variance	1.63	0.70	0.55	0.80	1.00	1.24	64.05	20.99	0.79	1.66

All the remaining attributes have binomial properties which is why we are representing the distributions as pie charts in Figure 2. We can record that the share of

males and females is approximately the same. This is crucial, since data with an imbalanced gender share is often biased due to very different behaviours and interests. Further, we can see that most families have more than one child with the parents living together. Interesting is also that nearly 17% of the students do not have internet access. This relatively high number results from the publishing date of the survey in 2008, where internet access was not as common as it is today. Finally, we can observe that almost all students strive for a higher school education.

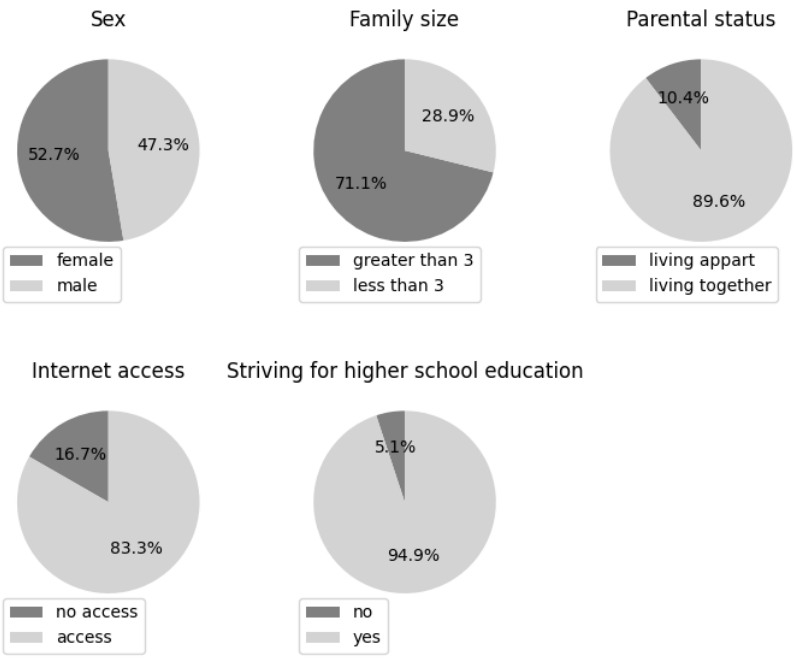

Figure 2: Distributions of the binomial variables

After providing first insights to the distribution and structure of the dataset we are now interested in the correlation of selected attributes to draw conclusions about the factors that influence alcohol consumption of Portuguese students. Figure 3 shows the correlation matrix of the numerical attributes. We observe that the workday consumption of alcohol does not show a clear correlation with one of the other attributes. We obtain the highest correlation for the weekend alcohol consumption with a correlation coefficient of 0.65, followed by go out frequency with a coefficient of 0.27. Thus, we cannot clearly identify the most important factors of student's workday alcohol consumption from our analysis so far and more advanced methods are required to come up with a reliable model.

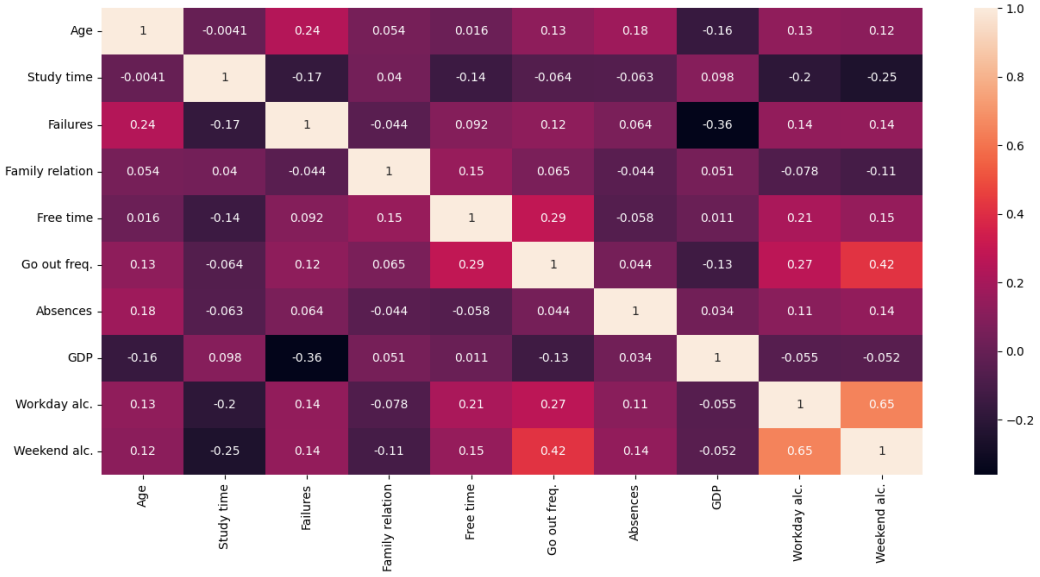

Figure 3: Correlation Matrix of the attributes

3 Data Preparation

For our further analysis we use RapidMiner Studio version 3.10. The high-level layout of our forecasting model is shown in Figure 4. In a first step we import the dataset from the CSV-file. Here we only select our attributes of interest that were also shown in Table 1. Next, we start the subprocess ‘Data Preprocessing’. We introduce this subprocess to maintain a clear and easy to follow layout. The subprocess is shown in Figure 5 and is discussed in particular in this paragraph. First, we use the ‘Set Role’ operator to assign the label to our target attribute ‘Workday alcohol consumption’. In order to evaluate different algorithms we use a ‘Performance’ operator within the cross validation of our model. However, this operator needs a nominal label attribute to calculate performance criteria for classification tasks. Therefore, we format the target attribute ‘Workday alcohol consumption’ using the ‘Numerical to Polynomial’ operator. Further, some predefined RapidMiner Studio algorithms, namely the *Deep Learning-* , *Neural Net-*, and *AutoMLP-Algorithms* require numerical instead of nominal attributes as input, which is why for these methods we additionally use the ‘Nominal to Numerical’ operator. As mentioned, our data does not have any missing values, which is why we can use all of the data. We showed in the last chapter that the attributes of the dataset have significantly varying value ranges and, therefore, may require normalization. However, not all predictive methods need normalized data and some algorithms obtain even better results without. In the next steps we remove outliers from the dataset. We detect the outliers based on the Euclidian distance to their nearest neighbours. Since our dataset is not very large (395 instances) we only restrict the removal of outliers to 10 instances and filter the dataset using the ‘Filter Example’ and ‘Select Attribute’ operators.

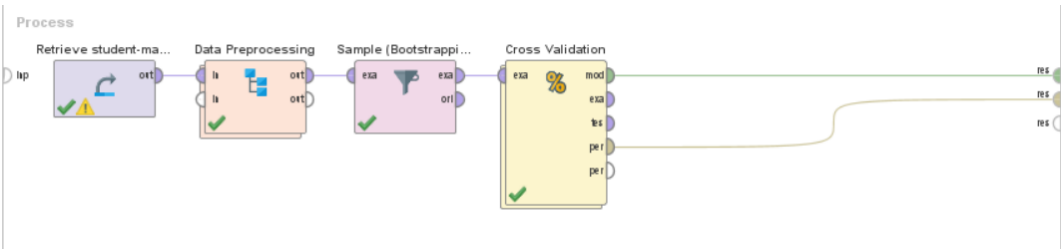

Figure 4: High-level layout of the model

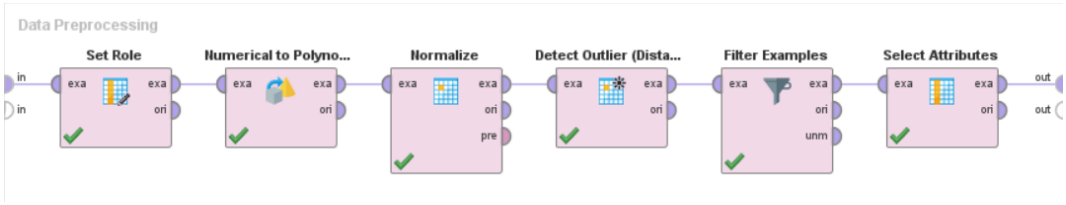

Figure 5: Subprocess layout for data preparation

4 Modelling

After preparing the data it is crucial to sample the data in order to obtain meaningful and reliable results. When looking at the distribution of our target attribute 'Workday Alcohol Consumption' in Figure 1 we can observe that the data is very imbalanced. Many students estimate their alcohol consumption during the week to be very low while only a few students have a higher consumption. The mean of 1.48 supports the thesis that most students fall into the first two classes. It is, therefore, important that an appropriate sampling technique is applied to prevent any bias and obtain more robust results. To satisfy these conditions we use the bootstrapping sampling technique that samples with replacement and, hence, produces duplicate instances in the sample. In RapidMiner Studio this can be obtained easily by using the 'Bootstrapping Sample' operator. As parameters we use 1.0 as relative sample ratio which yields in a sample that is as large as the original dataset and may contain duplicates. We choose the sample size because our dataset is relatively small and better results were achieved with higher sample ratios. For robust and reliable results we use cross-validation with ten folds in the next step. This means that we split the data to ten training sets and one validation set. We then apply the model obtained on the test data and measure the classification performance on the validation set. In the application of the process to the training sets we distinguish between the algorithms that require the additional 'Nominal to Numerical' operator and are shown in Figure 6, and the ones that do not need it which are presented in Figure 7. We multiply the preprocessed data to perform multiple machine learning algorithms at once. For the *Neural Net-*, and the *AutoMLP-Algorithms* we use the default settings of RapidMiner Studio. For the *Deep Learning-Algorithm* we define the epochs-parameter (how many times the dataset should be iterated) as 10.0. We proceed likewise for the algorithms of Figure 7. Here we use k=5 as the number of clusters for the *k-NN-Algorithm* even though experiments yielded slightly better results for k=2. However, to small values for k make the model too specific such that it fails when applied to a new dataset. For the *Decision Tree-Algorithm* we define the information gain as splitting criterion. This method usually has a bias for selecting attributes with a large number of values. Since our dataset only has one attribute (Number of Absences) that has considerably more values than the others this bias does not have a large impact on the results and is, therefore, the preferred splitting criterion. The same holds for the *Random-Forest-Algorithm* which also uses the

information gain as splitting criterion. Here, we create 100 trees and limit the maximum depth of each tree to ten. The performance measures reported by the model are the accuracy, the weighted mean (WM) recall, the WM precision, the absolute error, the relative error, the root mean squared error (RMSE), and the root relative squared error (RRSE).

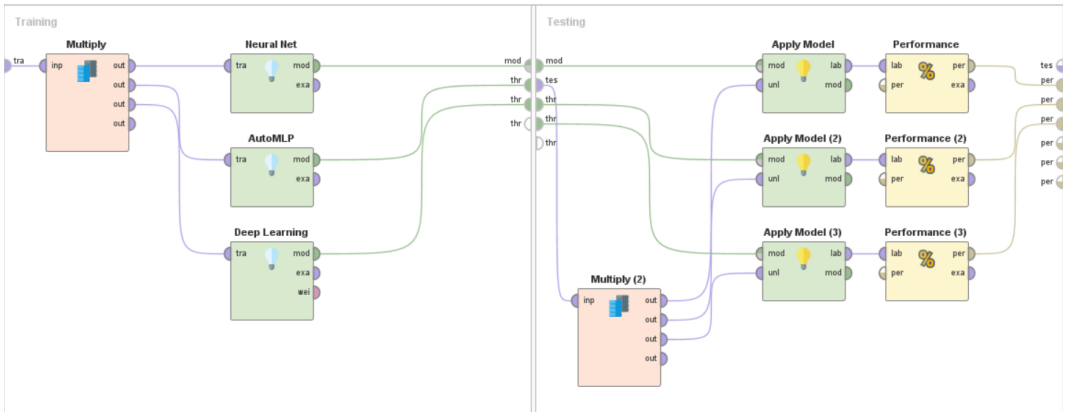

Figure 6: Cross-Validation layout for Neural Net-, AutoMLP-, Deep Learning-Algorithms

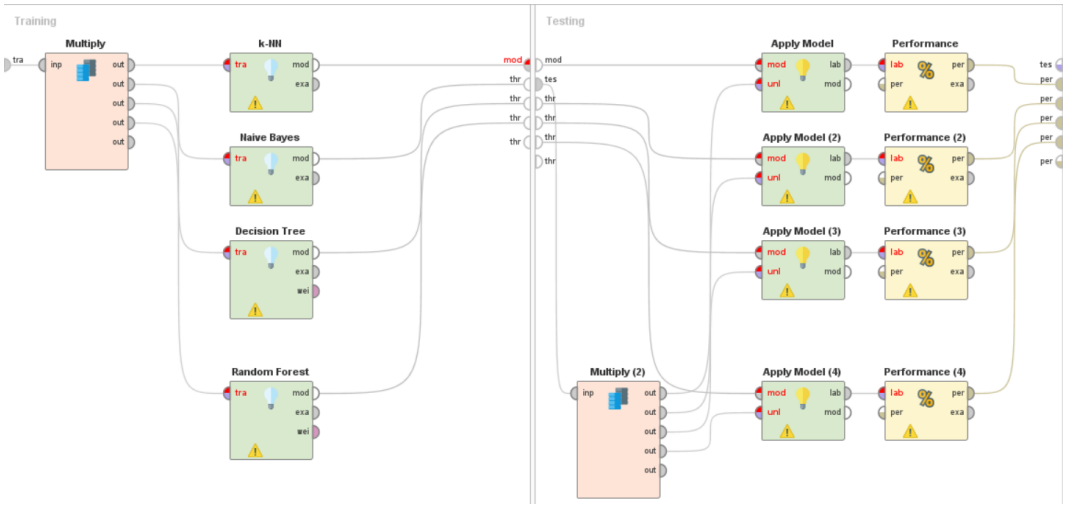

Figure 7: Cross-Validation layout for k-NN-, Naïve Bayes-, Decision Tree-, and Random Forest-Algorithms

5 Evaluation

To evaluate the different classifiers we use the measure that were mentioned in the last section. The overall performance of the model is measured with the accuracy. Further, we use the weighted mean recall to obtain the coverage of the true positive sample. In other words, what is the share of instances of a class that truly belong to it and were classified correctly. Next, we take the weighted mean precision to measure how many right instances were predicted for this class. The absolute error alludes to the magnitude of difference between the prediction of an observation and the true value of that observation. While the relative error is the ratio of the absolute error of a measurement to the measurement being taken. In other words, this type of error is comparative with the size of the item being estimated. The RMSE is the standard deviation of the residuals (prediction errors) and the RRSE measure is its relative measurement. Based on these evaluation measures we can then conclude which algorithm performs best. Table 3 reports the results of all algorithms without normalization, and Table 4 reports the results with normalization. The best results of each columns are highlighted in green. We can observe that the *Random Forest-Algorithm* clearly outperforms the other algorithms in both tables. Usually it shows a very good performance in many problems, which can be confirmed with our results. However, we also experienced the downside of this algorithm, namely the large computational effort. In comparison to other algorithms it took significantly longer to compute all the trees.

When looking at the other algorithms we can observe that the predictive power of the *k-NN-Algorithm* is only slightly worse than the other algorithms even though it is a rather simple algorithm. One reason for this may be that the dataset does not have many outliers that can influence this algorithm. Unsurprisingly the results of the *k-NN-Algorithm* were better with normalizing the data. This may be explained with the heterogeneity of the data, as many attributes have different value range, such as ‘Age’, ‘GDP’, and ‘No. of absences’ which show large deviations in the scales. Second, we have the *Naïve Bayes-Algorithm* which reports the lower performance in comparison to other predictive methods. We can explain this since the predictive attributes are probably not completely independent, which is one of the weaknesses of this algorithm. From the correlation matrix in Figure 3 we know that the alcohol consumption on weekends has the highest correlation to the alcohol consumption during on workdays. It is very likely that students that drink more

regularly on weekends also drink more during weekdays, hence the attributes may not be independent and the predictive performance of the *Naïve Bayes-Algorithm* is lowered. Surprisingly the performance of the *Deep-Learning-Algorithm* is worse than expected as it only yields average results. A possible problem for this algorithm might be that it works best when applied to large datasets. Since this is not the case in this study the performance may not be as good as possible. The *Decision-Tree-Algorithm* achieves the best results for the absolute and relative error. This means that it predicted the best values for each class. Overall, it has a good predictive power, probably since the local optima that are found by the algorithm are mostly global optima as well. Therefore, the *Random Forest-Algorithm* obtains even better results as many decision trees are created within this method. The *Neural Net-Algorithm* and *AutoMLP-Algorithm* behave very similar to the *Deep Learning-Algorithm* and likely have similar issues with the size of the dataset that limit their predictive power. Nevertheless, we can conclude that these algorithms can be applied easily to our dataset as they do not require intensive data preprocessing. Figure 8 supports this hypothesis as it compares the methods with and without data preprocessing. We can observe that only the weighted mean recall and precision show larger deviations.

Table 3: Comparison of the predictive algorithms without normalization

Algorithm	Accuracy	WM recall	WM precision	Absolute Error	Relative Error	RMSE	RRSE
k-NN (k=5)	74.56%	52.3%	52.02%	0.294	29.45%	0.425	0.893
Naïve Bayes	77.67%	64.52%	60.04%	0.254	25.42%	0.434	0.911
Deep Learning	83.66%	69.53%	71.07%	0.212	21.18%	0.368	0.775
Decision Tree	83.12%	62.35%	66.71%	0.182	18.17%	0.375	0.788
Random Forest	88.60%	74.66%	84.33%	0.214	21.35%	0.326	0.687
Neural Net	84.68%	72.36%	78.22%	0.196	19.65%	0.383	0.804
AutoMLP	82.86%	70.19%	71.05%	0.189	18.87%	0.387	0.813

Table 4: Comparison of the predictive algorithms with normalization

Algorithm	Accuracy	WM recall	WM precision	Absolute Error	Relative Error	RMSE	RRSE
k-NN (k=2)	79.74%	53.17%	75.99%	0.263	26.34%	0.413	0.919
Naïve Bayes	73.27%	51.52%	45.75%	0.293	29.31%	0.467	1.042
Deep Learning	79.48%	56.96%	66.33%	0.268	26.79%	0.417	0.93
Decision Tree	84.43%	70.68%	70.1%	0.188	18.81%	0.377	0.845

Random Forest	87.78%	71.32%	77.16%	0.23	23.03%	0.344	0.765
Neural Net	79.74%	54.31%	54.64%	0.273	27.28%	0.422	0.929
AutoMLP	81.04%	68.05%	68.53%	0.222	22.2%	0.413	0.915

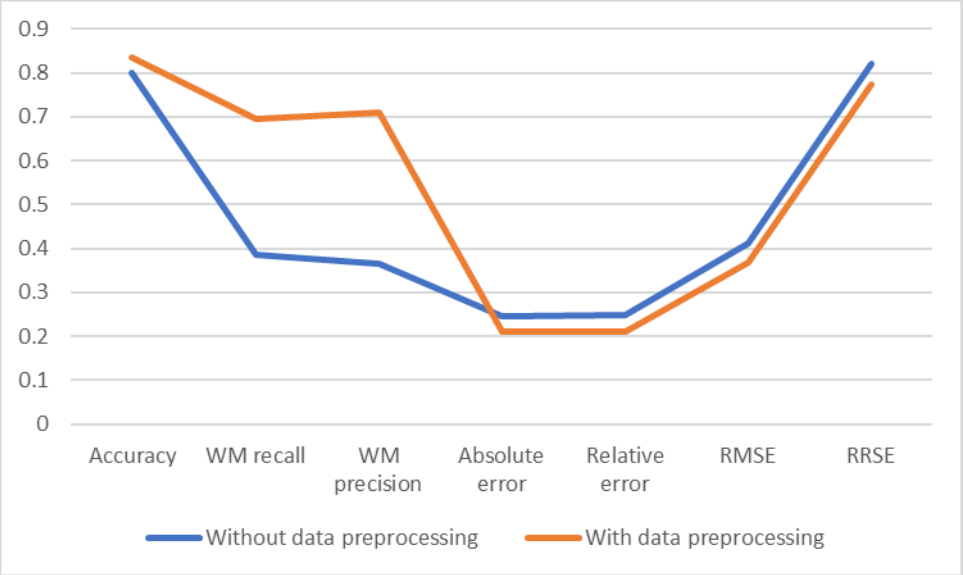

Figure 8: Neural Net algorithm with and without data preprocessing

Eventually, we can say that our data preprocessing, sampling technique and applied machine learning algorithms were selected appropriate and good results can be obtained with those methods. Further, the results also matched our expectations as it is a relatively small dataset. For future research we propose to intensify the research on the promising Artificial network-algorithms as they usually obtain robust and reliable results. However, more data has to be collected to see if improvements can be achieved. Apart from that the exclusion of the attribute 'Weekend alcohol consumption' could yield interesting results as it may be a dependent variable and strongly influences our target attribute.

Appendix A

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sn

# Describing the important columns for the study
consumption = pd.read_csv('student-mat.csv', sep=',')

# Shape of the dataframe
print(consumption.shape)

# Count missing values
print(consumption.isnull().sum())

# Get summary statistics of important qualitative columns
consumption1 = consumption[['age', 'studytime', 'failures',
'famrel', 'freetime', 'goout', 'absences', 'G3', 'Dalc',
'Walc']].copy()
print(consumption1.describe())
print('median: \n', consumption1.median(), '\n variance: \n', con-
sumption1.var())

# Plot the histograms
fig, axs = plt.subplots(4, 3, facecolor='w', edgecolor='k')
fig.subplots_adjust(hspace=.5, wspace=.3)
axs[0, 0].hist(consumption1.age, 7, rwidth=.8, color='k')
axs[0, 0].set_xlabel('Age')
axs[0, 1].hist(consumption1.studytime, 4, rwidth=.8, color='k')
axs[0, 1].set_xlabel('Study time')
axs[0, 2].hist(consumption1.failures, 4, rwidth=.8, color='k')
axs[0, 2].set_xlabel('Failures of past classes')
axs[1, 0].hist(consumption1.famrel, 5, rwidth=.8, color='k')
axs[1, 0].set_xlabel('Family relationship')
axs[1, 1].hist(consumption1.freetime, 5, rwidth=.8, color='k')
axs[1, 1].set_xlabel('Free time')
axs[1, 2].hist(consumption1.goout, 5, rwidth=.8, color='k')
axs[1, 2].set_xlabel('Go out frequency')
axs[2, 0].hist(consumption1.absences, 20, rwidth=.8, color='k')
axs[2, 0].set_xlabel('Number of abscences')
axs[2, 1].hist(consumption1.G3, 20, rwidth=.8, color='k')
axs[2, 1].set_xlabel('GDP')
axs[2, 2].hist(consumption1.Dalc, 5, rwidth=.8, color='k')
axs[2, 2].set_xlabel('Workday alcohol consumption')
axs[3, 0].hist(consumption1.Walc, 5, rwidth=.8, color='k')
axs[3, 0].set_xlabel('Weekend alcohol consumption')
fig.delaxes(axs[3, 1])
fig.delaxes(axs[3, 2])
plt.show()

# Plot the correlation matrix
labels = ['Age', 'Study time', 'Failures', 'Family relation',
'Free time', 'Go out freq.', 'Absences', 'GDP', 'Workday alc.',
'Weekend alc.']
corrMatrix = consumption1.corr()
sn.heatmap(corrMatrix, annot=True, xticklabels=labels, ytick-
labels=labels)
plt.show()

# Get summary statistics of important quantitative columns and
plot pie charts
fig, axs = plt.subplots(2, 3, facecolor='w', edgecolor='k')
fig.subplots_adjust(hspace=.5, wspace=.3)
```

```
axs[0, 0].pie(consumption.groupby('sex').size(), colors=['gray',
'lightgray'], startangle=90, autopct='%1.1f%%')
axs[0, 0].set_title('Sex')
axs[0, 0].legend(labels=['female', 'male'], loc='lower left',
bbox_to_anchor=(0,-0.2))
axs[0, 1].pie(consumption.groupby('famsize').size(), col-
ors=['gray', 'lightgray'], startangle=90, autopct='%1.1f%%')
axs[0, 1].set_title('Family size')
axs[0, 1].legend(labels=['greater than 3', 'less than 3'],
loc='lower left', bbox_to_anchor=(0,-0.2))
axs[0, 2].pie(consumption.groupby('Pstatus').size(), col-
ors=['gray', 'lightgray'], startangle=90, autopct='%1.1f%%')
axs[0, 2].set_title('Parental status')
axs[0, 2].legend(labels=['living appart', 'living together'],
loc='lower left', bbox_to_anchor=(0,-0.2))
axs[1, 0].pie(consumption.groupby('internet').size(), col-
ors=['gray', 'lightgray'], startangle=90, autopct='%1.1f%%')
axs[1, 0].set_title('Internet access')
axs[1, 0].legend(labels=['no access', 'access'], loc='lower left',
bbox_to_anchor=(0,-0.2))
axs[1, 1].pie(consumption.groupby('higher').size(), col-
ors=['gray', 'lightgray'], startangle=90, autopct='%1.1f%%')
axs[1, 1].set_title('Striving for higher school education')
axs[1, 1].legend(labels=['no', 'yes'], loc='lower left',
bbox_to_anchor=(0,-0.2))
fig.delaxes(axs[1, 2])
plt.show()
```