KNOWLEDGE DISCOVERY COURSE, CENG542 – TERM PROJECT

PREDICTING ALCOHOL CONSUMPTION AMONG STUDENTS

GROUP MEMBERS:

- **GULİZ AKKAYA, 283078008**
- **ISMAIL CEBECİ, 272001030**
- **BURAK TOPÇU, 283078027**

ABSTRACT

The persisting high levels of alcohol consumption among the majority of university students highlight the need for effective preventive and therapeutic interventions for both genders. The aim of our research on this subject is to increase the prevalence of excessive drinking among university students. While the sale and use of alcohol for those aged 18 and over is legally allowed, those under the age of 18 also consume a lot of alcohol. Therefore, there is no concentration on university students only. In addition, drinking habits do not depend on marital status; and most students don't drink alone. It is better known that they consume recreational alcohol rather than melancholy. In this proposal, we will investigate alcohol consumption among students. The alcohol consumption tendency of a particular student will be determined. We used the classification approach to find the predictive model used to predict alcohol trends. The database we use includes various features detailed in the following parts. We tried to predict the alcohol tendency of the students by determining the most effective of all traits, that is, by removing the traits that had the least impact on our research from the classification process. Eventually, it is observed that above 90% prediction accuracy is reached with some features over 650 students.

INTRODUCTION

The uncontrolled alcohol consumption among students is a significant issue. Students are exposed to negative effects on memory function and learning ability because of alcohol consumption. As a result of high level of alcohol consumption, students may have some problems in daily and academic life. For this reason, it is important to analyze the factors that are related to alcohol consumption among students and this information may be utilized to find solutions for this issue.

There are several studies that are performed to predict and analyze the alcohol consumption behaviors among students. Various algorithms and datasets are utilized for the classification task of student alcohol consumption in these studies. For instance, a classification study [2] uses different algorithms such as decision tree and random forest to perform the classification using the student alcohol

consumption on the dataset [1]. According to the results for mentioned work, it is stated that the random forest algorithm provides higher accuracy results than decision tree algorithm. Furthermore, various variables that are related to student alcohol consumption level such as gender, go-out level and study time are identified in the study. Besides, four different decision tree algorithms such as BFTree, J48, RepTree and Simple Cart are used for the classification of students using another student alcohol consumption dataset that is generated with university students' data [3]. According to the experimental results of this study, it is observed that BFTree algorithm provides more accurate results than other decision tree algorithms. Another study utilizes two variations of neural network algorithms such as a self-tuning multilayer perceptron classifier (AutoMLP) and the standard MLP neural network to predict the alcohol consumption behaviors among secondary school student [4]. As a result of the study, the classification performance results are presented in terms of the classification accuracy, squared error and root mean squared error and it is stated that AutoMLP produces higher accuracy results than standard MLP.

In this project, we investigate the features of students that are related to alcohol consumption using Pearson correlation method and feature selection at first. Afterwards, we utilize 4 different outlier detection methods such as Isolation Forest, Elliptic Envelope, Local Outlier Factor and OneClass SVM to find outliers in the data. Moreover, we perform classification on student alcohol consumption dataset [1] using different algorithms known as Support Vector Machines (SVM), Stochastic Gradient Descent (SGD), Random Forest (RF), Gradient Boosting (GB), Decision Tree (DT) and AdaBoost. According to the experimental results, we observe that age, study time, go out level, number of school absences, course grades and weekend alcohol consumption level features are the most related attributes to the alcohol consumption level of students. Furthermore, the experimental results show that Support Vector Machines algorithm produces highest accuracy values compared to the other classification algorithms.

BACKGROUND

Support Vector Machines (SVM)

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenge. In the SVM algorithm, each data item is plotted as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, classification is performed by finding the hyper-plane that differentiates the two classes very well. The benefit is that you can capture much more complex relationships between your datapoints without having to perform difficult transformations on your own. SVM can be imported into python from sklearn library, and an example usage described in the appendix section [5][9].

Random Forests Algorithm

Random forests create decision trees on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It also provides a pretty good indicator of the feature importance. Random forests are considered as a highly accurate and robust method because of

the number of decision trees participating in the process. Random Forests algorithm can be imported from sklearn.ensemble library in python and an example usage described in the appendix section [6][9].

Gradient Boosting Algorithm

Gradient boosting classifiers are the AdaBoosting method combined with weighted minimization, after which the classifiers and weighted inputs are recalculated. The objective of Gradient Boosting classifiers is to minimize the loss, or the difference between the actual class value of the training example and the predicted class value. Gradient Boosting algorithm is imported from sklearn.ensemble library in python and it's used like in appendix [8][9].

Decision Trees Algorithm

The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation. Decision Trees algorithm is imported from sklearn.tree library in python and it's used like in appendix [9].

AdaBoost Algorithm

An AdaBoost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases [9].

EXPERIMENT METHODOLOGY

Dataset

Firstly, the dataset collected from secondary education of two Portuguese schools [1] is used to examine relations between the student's features and student's performance. The explanations for the features are shared in table 1.

Table 1: Dataset description

Features	Explanations
school	student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
sex	student's sex (binary: 'F' - female or 'M' - male)
age	student's age (numeric: from 15 to 22)
address	student's home address type (binary: 'U' - urban or 'R' - rural)
famsize	family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
Pstatus	parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
Medu	mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 â€“ 5th to 9th grade, 3 â€“ secondary education or 4 â€“ higher education)
Fedu	father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 â€“ 5th to 9th grade, 3 â€“ secondary education or 4 â€“ higher education)

Mjob	mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
Fjob	father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g., administrative or police), 'at_home' or 'other')
reason	reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
guardian	student's guardian (nominal: 'mother', 'father' or 'other')
traveltime	home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
studytime	weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
failures	number of past class failures (numeric: n if 1<=n<3, else 4)
schoolsup	extra educational support (binary: yes or no)
famsup	family educational support (binary: yes or no)
paid	extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
activities	extra-curricular activities (binary: yes or no)
nursery	attended nursery school (binary: yes or no)
higher	wants to take higher education (binary: yes or no)
internet	Internet access at home (binary: yes or no)
romantic	with a romantic relationship (binary: yes or no)
famrel	quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
freetime	free time after school (numeric: from 1 - very low to 5 - very high)
goout	going out with friends (numeric: from 1 - very low to 5 - very high)
Dalc	workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
Walc	weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
health	current health status (numeric: from 1 - very bad to 5 - very good)
absences	number of school absences (numeric: from 0 to 93)
G1	first period grade (numeric: from 0 to 20)
G2	second period grade (numeric: from 0 to 20)
G3	final grade (numeric: from 0 to 20, output target)

As shown in the table 1, some of the features are string values. Before training with these metrics, string ones converted into a numeric format. For the 2 labeled features, binary conversion is done as 1 is for ‘yes’ and 0 is for ‘no’ attributes. Additionally, the features where the label amount is greater than 2 such as reason, Mjob, Fjob is represented with the integers are used to represent each of these attributes such that 1 is used for teachers, 2 is used for health care related jobs and so on.

Feature Selection and Dimension Reduction

Before setting up a training model, we try to observe the relationships between the features especially for 'Walc' (alcohol consumption in the weekend) and 'Dalc' (alcohol consumption in a workday) attributes by using the Pearson correlation method. The correlation results between the attributes are projected into a heat map format and shared in the below figure 1.

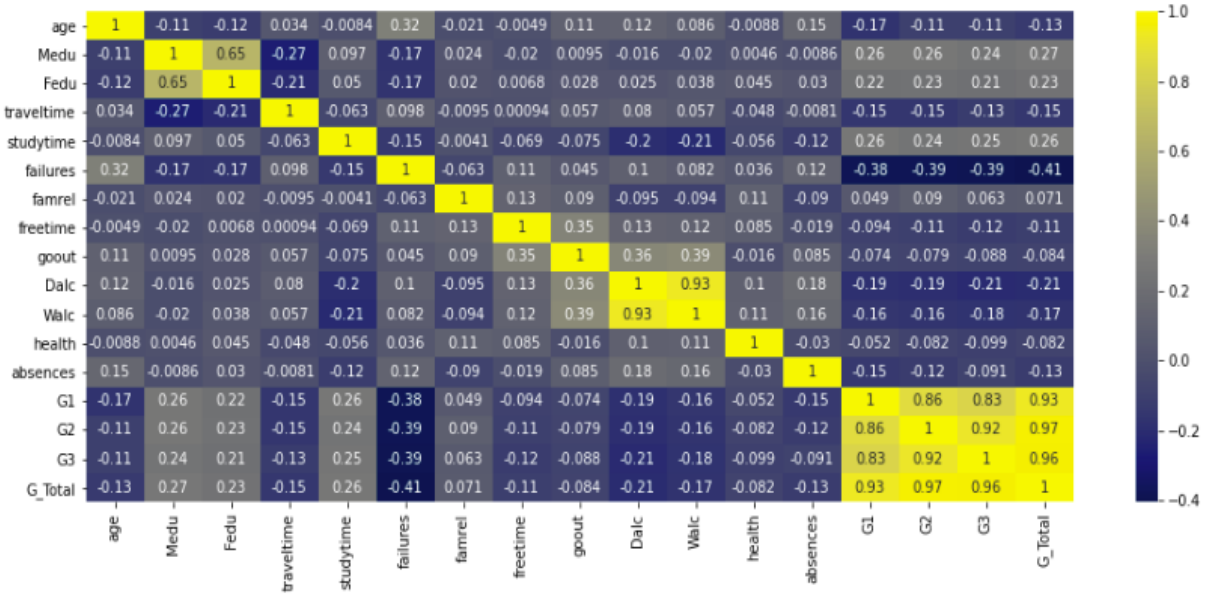

Figure 1: Pearson Correlations between the features.

After calculating the correlation values, the expected relations between the features that will manage the classifications are obtained. Dalc has strong relation with Walc and, relatively low correlation with age, study time, failures, free time, go out, health, absence and grades features. Other features have too low correlation with Dalc. Thus, remaining features are eliminated during the experiments.

Apart from the dropped features, feature selection is done with recursive feature elimination and cross-validated selection (RFECV) built-in function by assigning the RandomForestClassifier as estimator, StratifiedKFold(2) for the cross validation and scoring with respect to accuracy. Lastly, remaining features are age, study time, go out, absences, grade (G1, G2, G3) and Walc features.

Outlier Detection

After data set is manipulated to capture meaningful patterns, most of the outlier samples have to be dropped since they decrease the classification success by disturbing the classification models with irrelevant samples. To do so, we tried to 4 different outlier detection methods:

- Isolation Forest method
- Elliptic Envelope method
- Local Outlier Factor method
- One Class SVM method.

Each of these methods has already been implemented in sci-kit library. Hence, we can import them into our code and use them. Trainings with and without those outliers are done and results are recorded. Most successful outlier detector is the EllipticEnvelope method. Contamination parameter is selected as 0.1 for the configuration of outlier detector. With the help of this method, 65 of 650 outlier sample is discarded and corresponding results with and without outlier detector is shared in the trainings section.

Trainings

Trainings are done before and after from the above steps. Data set is divided into train set and test set with 0.8 and 0.2 proportions respectively. As mentioned above, Dalc is the target label, and the remaining features are the training attributes. This splitting is also done with the help of test_train_split() built-in function that splits data randomly with respect to specified portions.

Firstly, the data set trained with 6 different classifiers without feature selection and outlier detection and training accuracy is around 40% which is lower accuracy than toss a coin. Thus, we manipulate our data set with the above-mentioned methods. Our classifier models are described below with their configuration parameters:

- Support Vector Machine with default parameters
- Stochastic Gradient Classifier with penalty = l1 regularization, alpha = 0.001 (regularization parameter), tol = 10^-3 (tolerance), eta0 = 0.1 (learning parameter)
- Random Forest Classifier with number of estimators = 1000, random state=1 and class weight=balanced (effect of weights during classification)
- Gradient Boosting Classifier with number of estimators=1000, random state=1
- Decision Tree Classifier with default parameters
- Ada-Boost Classifier with default parameters

Before tuning parameters, each of these classifiers used with its default parameters. Afterwards, the classifier that result in more accuracy is enhanced by tuning new parameters as in SVM, RF and GB classifiers. Also, some pipelined methods are tried such as scale the data set with different Standard Scaler of the sci-kit at first and classifier model at second step to increase the training accuracy by making pipeline. All the experimental results including feature selection, outlier detection and training accuracies are shared in the results section.

RESULTS

Without feature selection and outlier detection methods, the specified 6 classifier methods result in the accuracies shared in figure 2. The trainings are done with and without Standard Scale built-in function of sci-kit library. To introduce Standard Scale, make_pipeline function is used. Since we observed that some of the features are not relevant and representative for our classification task, we drop some of them with RFECV method. Then, the same training is done after dimension reduction process and outlier detection is not still applied in those results. At last step, we realized that some of the samples are outlier such that our classification models cannot classify them with the configured models. Thus, we used

EllipticEnvelope method mentioned in outlier detection section to detect and drop outlier samples from our data set. After implementing outlier detection, same trainings are done with the specified 6 classification models and corresponding results are shared in the figure 2.

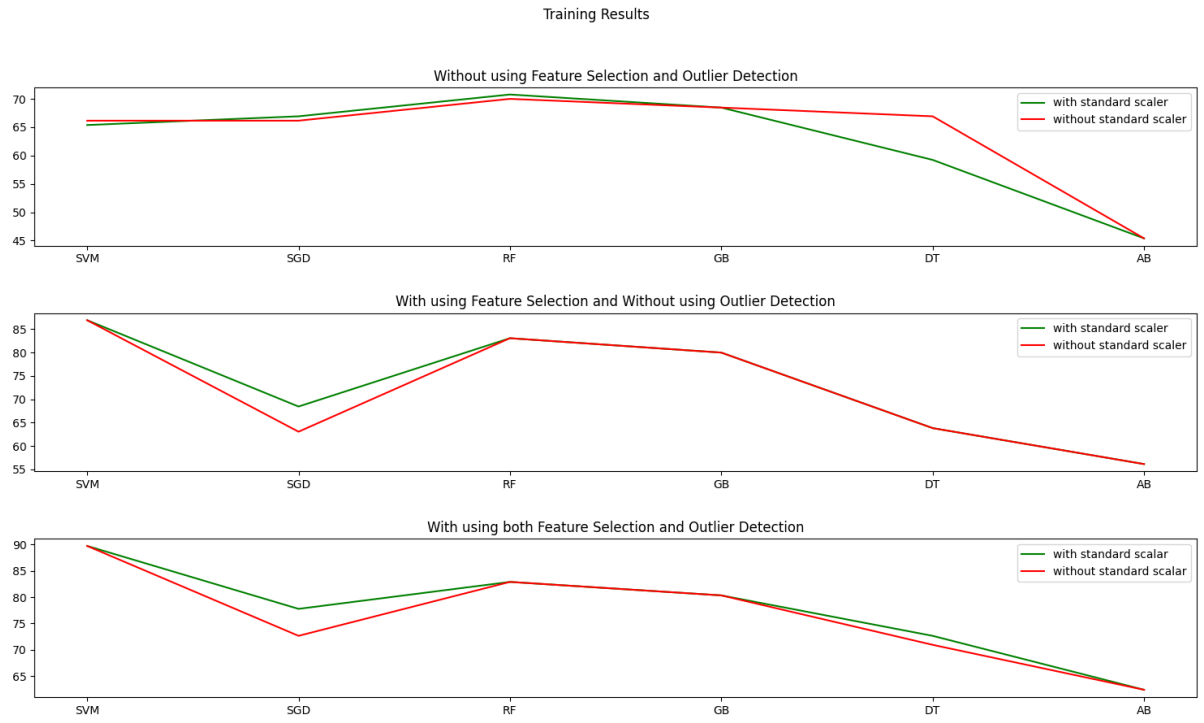

Figure 2: Training Results.

As a result, we achieved 93,16% accuracy with specified 8 features over 32 features and 584 samples over 650 samples. According to our data set and accuracy results, age, study time, go out, absences, grade (G1, G2, G3) and Walc features are representative in terms of predicting alcohol consumption of the students among the 'GP' - Gabriel Pereira and 'MS' - Mousinho da Silveira students.

CONCLUSION

In this project, we perform several experiments to analyze student alcohol consumption using a dataset. We investigate the various attributes of students that affect the alcohol consumption level by utilizing correlation and feature selection methods. According to the experimental results, we can state that the most related attributes to student alcohol consumption are age, study time, go out level, number of school absences, course grades and weekend alcohol consumption level. In addition, we perform outlier detection to improve the accuracy results of classification. Furthermore, we present experimental results for 6 different classification algorithms, and we observe that SVM algorithm produces higher accuracy results than others as generating %93.1 accuracy. The dataset that is used for this project is limited to only the students from 2 different schools. As future work, the dataset can be enhanced by

collecting bigger dataset from various schools and more accurate experimental results can be obtained in this way.

REFERENCES

[1] Cortez, Paulo & Silva, Alice. (2008). Using data mining to predict secondary school student performance. EUROSIS.

[2] A. Pisutaporn, B. Chonvirachkul and D. Sutivong, "Relevant factors and classification of student alcohol consumption," *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*, 2018, pp. 1-6, doi: 10.1109/ICIRD.2018.8376297.

[3] Pal, Saurabh and Chaurasia, Vikas, Is Alcohol Affect Higher Education Students Performance: Searching and Predicting Pattern Using Data Mining Algorithms (June 22, 2017). International Journal of Innovations & Advancement in Computer Science IJIACS ISSN 2347 – 8616 Volume 6, Issue 4 April 2017, Available at SSRN: https://ssrn.com/abstract=2991214

[4] Palaniappan, S., Hameed, N.A., Mustapha, A., Samsudin, N.: Classification of alcohol consumption among secondary school students. JOIV: International Journal of Informatics Visualization 1, 224 (2017). https://doi.org/10.30630/joiv.1.4-2.64

[5] "Understanding Support Vector Machine (SVM) algorithm from examples (along with code)" https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

[6] "Understanding Random Forests Classifiers in Python"

https://www.datacamp.com/community/tutorials/random-forests-classifier-python?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631748&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adpostion=&utm_creative=278443377092&utm_targetid=aud-299261629574:dsa-429603003980&utm_loc_interest_ms=&utm_loc_physical_ms=1012783&gclid=Cj0KCQjwna2FBhDPARIsACAEc_VD9XkCwPOPyL-z73BZBe7CyQo9HBtVLfgDFpm6r4zxwDRXIiVaI1caArZvEALw_wcB

[8] "Gradient Boosting Classifiers in Python with Scikit-Learn"

https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-learn/#:~:text=The%20objective%20of%20Gradient%20Boosting,and%20the%20predicted%20class%20value.

[9] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

APPENDIX

Training algorithms

SVM:

- *from sklearn import svm*
- *clf = svm.SVC()*
- *clf.fit(X_train, Y_train)*
- *svc_pred = clf.predict(X_test)*

SGD:

- *from sklearn.linear_model import SGDClassifier*
- *model = make_pipeline(StandardScaler(), SGDClassifier(penalty = 'l1', alpha = 0.001, max_iter=10000, tol=1e-3, eta0=0.1))*
- *model.fit(X_train, Y_train)*
- *prediction = model.predict(X_test)*

max_iter: The maximum number of passes over the training data (aka epochs).

tol: The stopping criterion.

penalty: The penalty (aka regularization term) to be used. Defaults to 'l2' which is the standard regularizer for linear SVM models. 'l1' and 'elasticnet' might bring sparsity to the model (feature selection) not achievable with 'l2'.

alpha: Constant that multiplies the regularization term. The higher the value, the stronger the regularization.

RF:

- *from sklearn.ensemble import RandomForestClassifier*
- *rf_model = make_pipeline(StandardScaler(), RandomForestClassifier(n_estimators = 1000, random_state=1, class_weight='balanced'))*
- *rf_model.fit(X_train, Y_train)*
- *prediction = rf_model.predict(X_test)*

random_state: Controls both the randomness of the bootstrapping of the samples used when building trees.

max_depth: The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.

GB:

- *from sklearn.ensemble import GradientBoostingClassifier*
- *gb_model = GradientBoostingClassifier(n_estimators=1000, random_state=1)*

- *gb_model.fit(X_train, Y_train)*
- *prediction = gb_model.predict(X_test)*

n_estimators: The number of boosting stages to perform. Gradient boosting is robust to over-fitting, so a large number usually results in better performance.

Decision Trees:

- *dt_model = DecisionTreeClassifier()*
- dt_model.fit(X_train,Y_train)
- dt_prediction=dt_model.predict(X_test)

AdaBoost:

- *ab_model = AdaBoostClassifier()*
- ab_model.fit(X_train,Y_train)
- ab_prediction = ab_model.predict(X_test)

Table 2: Training results using neither feature selection nor outlier detection.

	Without Standard Scale	With Standard Scale
SVM	**0.70**	**0.69**
SGD	**0.68**	**0.68**
RF	**0.70**	**0.70**
GB	0.65	0.66
DT	0.56	0.57
AB	0.48	0.48

Table 3: Training results with using feature selection and without using outlier detection.

	Without Standard Scale	With Standard Scale
SVM	**0.86**	**0.86**
SGD	0.60	0.68
RF	0.80	0.80
GB	0.70	0.70
DT	0.59	0.6
AB	0.53	0.53

Table 4: Training results with using both feature selection and outlier detection.

	Without Standard Scale	With Standard Scale
SVM	**0.93**	**0.93**
SGD	0.73	0.76
RF	**0.85**	**0.85**
GB	0.78	0.78
DT	0.52	0.58
AB	0.62	0.62