11-2022

From Machine Learning to Deep Learning: A comprehensive study of alcohol and drug use disorder

Banafsheh Rekabdar
Portland State University, rekabdar@pdx.edu

David L. Albright
University Of Alabama At Birmingham

Haelim Jeong
University Of Alabama At Birmingham

Sameerah Talafha
Southern Illinois University

Citation Details

Rekabdar, B., Albright, D. L., McDaniel, J. T., Talafha, S., & Jeong, H. (2022). From machine learning to deep learning: A comprehensive study of alcohol and drug use disorder. Healthcare Analytics, 2, 100104.

Healthcare Analytics 2 (2022) 100104

Contents lists available at ScienceDirect

Healthcare Analytics

journal homepage: www.elsevier.com/locate/health

From machine learning to deep learning: A comprehensive study of alcohol and drug use disorder[☆]

Banafsheh Rekabdar[a], David L. Albright[b,*], Justin T. McDaniel[c], Sameerah Talafha[d], Haelim Jeong[b]

[a] Computer Science Department, Portland State University, United States of America
[b] School of Social Work, University of Alabama, United States of America
[c] School of Human Sciences, Southern Illinois University, United States of America
[d] School of Computing, Southern Illinois University, United States of America

ARTICLE INFO

Keywords:
Artificial intelligence
Machine learning
Deep learning
Screening
Brief Intervention, and Referral to Treatment (SBIRT)
Substance Abuse and Mental Health Services Administration's (SAMHSA)

ABSTRACT

This study aims to train and validate machine learning and deep learning models to identify patients with risky alcohol and drug misuse in a Screening, Brief Intervention, and Referral to Treatment (SBIRT) program. An observational cohort of 6978 adults was admitted in the western region of Alabama at three medical facilities between January and December of 2019. Data were cleaned and pre-processed using data imputation techniques and an augmented sampling data method. The primary analysis involved the multi-class classification of alcohol and drug misuse. Our study shows that accurate identification of alcohol and drug use screening instrument scores was best accomplished with mixed-effects models following the imputation of missing data using the Generative Adversarial Imputation Networks (GAIN) method and then followed by applying the Synthetic Minority Over-sampling TEchnique-Nominal Continuous (SMOTE-NC) data augmentation method. Although mixed models are commonly employed in studies of electronic health records (EHRs), using the GAIN method followed by SMOTE-NC for diagnosing alcohol and drug use disorder is novel and original.

1. Introduction

Although few addiction science accomplishments are translated into clinical practice in sustainable ways, the Substance Abuse and Mental Health Services Administration's (SAMHSA) Screening, Brief Intervention, and Referral to Treatment (SBIRT) approach has been implemented in several clinics throughout the United States [1]. SBIRT involves a brief screening for substance use or mental health before a patient's clinic visit, followed by a clinicianś scoring of the screening and subsequent offering of a brief intervention, brief treatment, or referral to treatment, depending on the severity of the patientś screening score [2]. Cross-site evaluations have shown that SBIRT programs are effective regarding healthcare costs and changes in critical end-points [2]. However, SBIRT lacks the predictive capability for populations at risk for substance use disorders (SUD). Therefore, a predictive model is needed to detect the potential risk for SUD. Recently, a subsector of artificial intelligence, machine learning (ML), has been used in substance use research. ML can provide a predictive model based on pattern identification and computational learning [3]. Despite using ML algorithms to detect the likelihood of substance use patterns based on electronic medical records [4], little research has employed ML methods within the context of SBIRT [5]. Applied ML models within the context of SBIRT may create greater service efficiency by enabling clinicians to understand better the contextual factors associated with SBIRT outcomes and identify types of patients at greater risk for substance use disorders.

2. Literature review

Approximately 20.1 million Americans have a substance use disorder (SUD); however, only about 10% of affected individuals receive treatment [6]. Generally, addiction-related behaviors are harder to detect compared to other physical symptoms [7]. The integration of SUD screening and treatment within a primary care setting permits individuals who are being seen by a medical provider for other health issues to be treated for SUD, thus, increasing the likelihood of detecting SUD in a patient and providing services that they might have otherwise missed. As such, SBIRT services fill a critical need in the American populationś substance use treatment and recovery needs.

[☆] https://github.com/meera-m-t/Comprehensive-prediction-analysis-of-alcohol-and-drug-use-disorder-using-machine-deep-learning-algor.

[*] Corresponding author.

E-mail addresses: rekabdar@pdx.edu (B. Rekabdar), dlalbright@ua.edu (D.L. Albright), jtmcd@siu.edu (J.T. McDaniel), sameerah.talafha@siu.edu (S. Talafha), hjeong5@crimson.ua.edu (H. Jeong).

https://doi.org/10.1016/j.health.2022.100104
Received 30 March 2022; Received in revised form 21 August 2022; Accepted 24 August 2022

Data are routinely collected from patients in SBIRT programs via electronic health records (EHRs). Because patients are more likely to report SUD to their primary care physician, the data contained within EHRs in SBIRT programs is highly accurate [8]. The vast amount of data collected from SBIRT programs for a patient's EHR provides an opportunity for learning about trends and patterns akin to SUD in primary care-seeking patients.

Artificial intelligence prediction systems, developed with Machine Learning (ML) and Deep Learning (DL) techniques or algorithms, have been introduced within the healthcare setting to overcome physical limitations. ML is an empirical method of predicting and measuring potential risk factors [9], as it can detect and diagnose, as well as provide predictive outcomes [10], such as in the context of detecting thyroid cancer [11]. However, presently, there is little research employing ML methods within the context of SBIRT [5] despite its capability of detecting the likelihood of substance use patterns based on electronic medical records [4].

According to Bi and colleagues [12], ML is a branch of computer science that "emphasizes predictive accuracy over hypothesis-driven inference, usually focusing on large, high-dimensional data sets". For example, support vector machines, an ML algorithm based on kernel functions that overcome the rigid assumptions of some frequentist statistical models [13], such as least squares linear regression's linearity assumption, enables an EHR analyst to discover nuanced relationships between variables as the algorithm learns from the data. As a result, ML models provide prediction equations for outcomes (e.g., substance use disorder severity) more accurately.

ML has been extensively used in certain health care scenarios. For example, during the COVID-19 pandemic, many researchers turned to ML and DL for purposes of improving screening, predictions, forecasting, contact tracing, and drug development [14]. For example, [15] used deep learning in order to predict the ending point of the pandemic in Canada. In another example, [16] used hybrid wavelet-autoregressive integrated moving average models and regression trees to conduct real-time forecasting of death rates associated with COVID-19. Such models have incredible clinical utility. In the same way that ML has been used to improve conditions relative to the COVID-19 pandemic, we sought to use ML in order to improve conditions related to alcohol and drug use screenings in primary care clinics.

Using ML to understand better patient characteristics associated with outcomes in the context of an SBIRT program could significantly improve SUD patient identification and service provision. Although theory-driven analysis is beneficial in understanding a baseline set of factors associated with an outcome, MLś emphasis on atheoretical learning from data provides an opportunity to discover previously unknown predictors of an outcome [17]. Therefore, the purpose of the present study was to explore the predictive accuracy of a suite of ML and DL algorithms within the context of an SBIRT program in Alabama to recommend a predictive model of SUD. Given that ML is an atheoretical approach to data analysis [18], we did not specify apriori hypotheses in this study.

3. Study area and dataset

3.1. Procedure

Executed in the western region of Alabama at three medical facilities, the screening, brief intervention, and referral to treatment in Alabama (AL-SBIRT) program served adult patients via universal and annual pre-screening for alcohol, drug, and tobacco use. AL-SBIRT commenced with a population-based, universal screening procedure in the waiting areas of medical facilities before the patient's appointment. As a part of the screening, each patient reported demographic data and details of their alcohol, drug, and tobacco use. All self-report data were retrieved with a tablet or desktop computer. Screening tool scores were calculated automatically and given to social workers or nurses through a web-based portal called "Wellness Tracker." Score severity on each tool was used to determine patient service recommendations. Patients who screened positive for any use of tobacco products were referred to a tobacco quitline [19].

Individuals who exhibited either no risk or low risk for drug or alcohol use were provided educational feedback about alcohol and drugs. Individuals who showed a mild risk for substance misuse were offered a 30-minute brief intervention (variable name = BI). Each BI comprised a brief negotiated interview based on motivational interviewing techniques [20]. Persons who showed moderate risk were offered BI and up to 12 brief treatments (variable name = BT). Each BT session was roughly 60 min and was driven by principles of Integrated Change Therapy [21]. Persons displaying severe risk were referred to an in-network specialty treatment (variable name = RT). All services were provided after the patient supplied consent. AL-SBIRT ended services when a person no longer wanted them. Individuals who screened positive on more than one tool were coded as 1 in a variable named COSCREEN.

3.2. Measures

The screening tool used in this study included the following components: demographics, tobacco, alcohol, and drugs. The demographics component of the tool was based on the SAMHSA mandated Government Performance and Results Act (GPRA) tool. Demographics compiled in the GPRA tool and available for use in this study included sex (variable name = SEX), ethnicity (variable name = HISPANIC), race (variable name = RACE), veteran status (variable name = VET), active duty military status (variable name = ACTIVE), previous military deployment status (variable name = DEPLOY), and age (variable name = AGE).

In this study, the tobacco use question (variable name = TOBMONTH) was asked as follows: "In the past 30 days, how many days did you use tobacco products (cigarettes, dip, chew, electronic cigarettes, etc.)?" We used the U.S-Alcohol Use Disorders Identification Test (U.S-AUDIT) to detect unhealthy alcohol use [19]. The U.S.-AUDIT is a 10-item tool validated for use in primary health care settings [22]. An example item from the AUDIT follows: "How often during the last year have you found that you were not able to stop drinking once you had started?" A score between 7–15 for females and 8–15 for males resulted in a suggestion for BI, while scores from 16–24 resulted in a suggestion for BT, and scores higher than 25 resulted in an RT. We included a variable in this study that was coded as one of the patients who used alcohol in the previous 30 days and 0 if the patient did not use alcohol in the past 30 days (variable name = ANYALC). We also included a variable in this study that measured the number of days in the past 30 days on which binge drinking (i.e., 5 or more drinks on one occasion) occurred (variable name = BINGEDAYS).

The DAST-10 was used as a drug misuse screening tool [23]. The DAST-10 is a 10-item self-report survey of drug use-related topics in the past year (variable name = DAST). An example question from the DAST-10 follows: " have you had medical problems as a result of your drug use (e.g., memory loss, hepatitis, convulsions, bleeding)?" AL-SBIRT patients who had a score of 1 or 2 were offered a BI, while patients who scored between 3 and 5 were supplied with BT. Patients who scored between 6 and 10 were referred to treatment (RT).

Other drug and alcohol use outcomes measured in this study included the following. DRUGDAYS (i.e., variable name) measured the number of days in the last 30 days in which any illicit drug was used. ALCDRUG (i.e., variable name) was coded dichotomously, where a 1 equaled co-use of alcohol and illicit drugs on the same occasion and 0 equaled no co-use of alcohol and illicit drugs. DAYSCOCAINE (i.e., variable name) measured the number of days in the last 30 days a person used cocaine. MARYJDAYS (i.e., variable name) measured the number of days in the past 30 days in which a person used marijuana. OTHER DRUGS (i.e., variable name) measured the number of days in the past 30 days in which a person used any other illicit drug. Finally, INJECT (i.e., variable name) was coded dichotomously, where a 1 equaled any injection drug use in the past 30 days, and a 0 equaled no injection drug use.

Table 1
Interpretation of DAST-10 score categories indicating the degree of consequences related to drug abuse.

DAST-10 Score	Degree of Problem related to drug abuse	Suggestion action	Class name
0	No problem reported	Non at this time	Class-0
1–2	Low level	Monitor, re-assess at a later date	Class-1
3–5	Moderate level	Further investigation	Class-2
6–8	Substantial level	Intensive assessment	Class-3
9–10	Sever level	Intensive assessment	Class-4

Table 2
Interpretation of AUDIT score categories indicating the extent of alcohol involvement along a broad continuum of severity.

AUDIT score	Intervention	Risk level	Class name
0–7	Alcohol education	Zone I	Class-0
8–15	Simple advice	Zone II	Class-1
16–40	Referral to specialist for diagnoses evaluation and treatment	Zone III	Class-2

3.3. *Sample*

This studyś sample comprises persons pre-screened by the AL-SBIRT program for substance use between January 2019 and December 2019. De-identified electronic health records for 6,978 adults were acquired from the three medical facilities. Because the data in this study were forwarded to the authors with identifying data already removed, this study was deemed exempt from Institutional Review Board review. Thus, the research in this study consisted of secondary data. The sample was primarily middle-aged (M = 44.21), female (52.16%), non-Hispanic (97.41%), and non-white (67.18%).

4. Overview method

Our study method was comprised of two main stages. Fig. 1 shows each step applied in both stages and summarizes the study.

Initial data pre-processing occurred in the first step. This step is crucial and needs to be adequately implemented to build accurate models for the performance analysis. "Cleaning Dataset" in Fig. 1 is the original dataset obtained by applying initial pre-processing to the survey data (dropping the columns that had 70% null values, data type conversion, etc.). Subsequently, we determine the target variable (DAST or AUDIT), which is the dependent feature we want to understand deeply. We group the DAST columns by the range method using the pivot table Table 1 [24]. Then we repeat the same process for AUDIT columns based on Table 2 [24].

We reserve 80% of the dataset for training, "Training Dataset," where we apply different deep/ML models, and we reserve 20% of the dataset for testing, "Testing Dataset," where we evaluate these models. Subsequent operations were applied in two stages: (1.) in the first stage, "Stage (1)", the training dataset is used for different DL/ML classifiers, including both fixed and mixed-effects models, after dropping the rows with missing values; (2.) in the second stage, "Stage (2)", before training the classifiers, another type of pre-processing application is involved, which includes imputing missing values and over-sampling. After imputation of the null values and creating "Imputed Dataset", we apply the over-sampling method (SMOTE-NC) to achieve a more balanced dataset, called "Balanced Dataset".

During the testing phase, we load a best-fitting saved model in Stage (1) and Stage (2) for a specific classifier and apply it to the testing dataset and evaluate the model based on four evaluation metrics (see Section 7).

In this study, comprehensive prediction and analysis were successfully completed in both stages (see Section 8).

5. Pre-processing techniques

Data pre-processing techniques play a crucial role in the success of ML/DL models and increase the quality of training data. Accordingly, we apply imputation of missing data and over-sampling techniques.

5.1. *Handling missing data*

Missing values in survey research constitute the main obstacle to accurate survey analysis. Dropping all rows with null values in a small dataset is not ideal. A great deal of research has been done to develop and improve imputation methods for missing survey values in the past two decades, and research studies are still underway. These methods aim to compensate for missing data so that the analysis file may be subject to any form of analysis without the need for more study of the missing data. Our missing survey data arose due to what might be called "partial non-response," which occurs when a respondent in a multi-stage survey gives information for some but not all stages of data collection. Partial non-response can be handled by retaining all rows in the analysis file and imputing all missing responses. In this study, we applied various DL and classical ML mechanisms to the task of imputation. Good results have been gained by using classical ML techniques, such as the K-Nearest Neighbor (KNN) imputer [25] and Multiple Imputations by Chained Equations (MICE) [26]. Recently DL techniques, such as Multiple Imputation with Denoising Autoencoders (MIDAS) [27], DataWig [28], and Generative Adversarial Imputation Networks (GAIN) [29], have been shown to possibly have even greater potential compared to the classical ML models.

We evaluate various imputation methods based on the ability of the process to find the correct value in each column that has missing data. To do that, we drop all rows that have null values from the original dataset to create an imputation dataset, which is then used to create a Missing Completely at Random (MCAR) [30] variable (20%). Following that, we impute the missing values using different imputation methods. Lastly, we compute the accuracy between the imputation values and the actual values in the imputation dataset. We choose the most efficient method to apply to the missing values in the original training dataset based on the results.

5.1.1. *K-Nearest Neighbor (KNN) imputation*

The KNN imputation approach [25] is used to detect the K-nearest neighbors of missing values from all complete samples in a given dataset and then replace them with the mean of the neighbors if the target attribute is numeric, indicated by the mean rule, or using the most frequent one occurring in the neighbors if the target attribute is categorical, shown by the majority rule.

5.1.2. *Multiple Imputations by Chained Equations (MICE)*

MICE [31], sometimes known as "sequential regression of multiple imputations," has stood out in statistical research as an effective technique for imputing missing data. In contrast to single imputation, constructing multiple imputations considers the statistical uncertainty in the imputations. Moreover, the chained equations model is adaptable in addressing different data types (e.g., continuous or discrete). MICE uses a process called Predictive Mean Matching (PMM) to choose which values are imputed, which considers each missing value as the target variable and the remaining variables as predictors, and then fills a missing value in based on the most relative predicted value for its value from the fitted model.

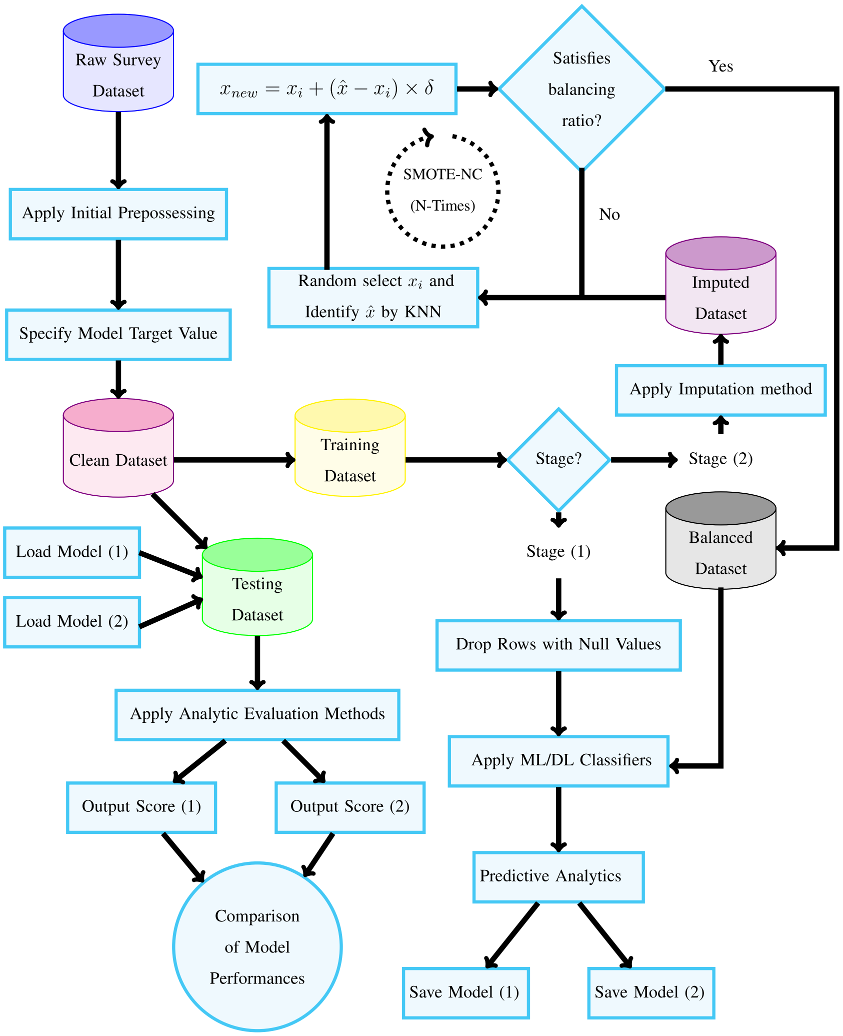


Fig. 1. The overall flowchart of our proposed approach.

5.1.3. DataWig

Popular imputation approaches, such as KNN and MICE, focus on imputing missing values in matrices, that is, imputation of numerical values from other numerical values. On the other hand, DL models address imputation for more heterogeneous data types; heterogeneous data may be ordinal or categorical. DataWig [28] combines DL feature extractors with automatic tuning of hyperparameters in an end-to-end fashion using the symbolic API of Apachemxnetto on both CPUs and GPUs. In DataWig, the user sets the imputation column (referred to as the output column) and other columns which contain helpful information for imputation (referred to as input columns). Then, depending on the data type in the imputation column (numerical or categorical variables), the trained model will be fitted using either regression or cross-entropy loss.

5.1.4. Multiple Imputation with Denoising Autoencoders (MIDAS)

MIDAS [27] is a multi-scale computational approach that has accuracy, speed, and scalability by relying on advanced computation and theories in DL. MIDAS uses a class of unsupervised neural networks known as "denoising autoencoders," which consider missing values as corrupted data and derives imputations from a model trained to reduce the reconstruction error on the initially observed part. In addition, functional flexibility allows MIDAS to create complicated and simple relationships among variables, which provides the foundation for performance gains across different data types and structures.

5.1.5. Generative Adversarial Imputation Networks (GAIN)

GAIN [29] uses the Deep Generative Network (GAN) architecture to impute missing values. The generator observes partial elements of an original data vector, then imputes the missing values conditioned on which are observed, and eventually outputs a completed vector that includes observed values. Next, the discriminator takes the vector to determine which elements were observed and imputed. Finally, the discriminator is provided with additional information about the "missingness" of the original data in a hint vector to force the generator to learn the desired distribution.

5.2. Synthetic Minority Over-sampling Technique for Nominal and Continuous features (SMOTE-NC) to make our dataset balanced

Imbalanced data is a common occurrence vis-a-vis actual data where the samples in one class outnumber others. Balanced classes are substantial for ML/DL models in classification tasks. However, these models might be biased toward the majority class, leading to underfitting or overfitting problems. SMOTE creates synthetic instances for minority–classes based on the information provided in the data [32]. SMOTE performs better than simple sampling (such as oversampling/under sampling) by preventing over/under-fitting problems. In this study, we used SMOTE-NC [33], an extension of the SMOTE algorithm, which synthetically creates mixed data types with continuous and categorical features. The particular idea of SMOTE-NC is performed as follows: determine the KNN of sample x_i in the class, choose N samples randomly and record each of them as $\hat{x}$. Eventually, the new sample x_{new} is given by interpolation as follows:

$$x_{new} = x_i + (\hat{x} - x_i) \times \delta \qquad i = 1, \ldots, N, \tag{1}$$

where δ is a random number uniformly distributed within the range (0,1).

It is very important to mention that SMOTE-NC is only applied to the training dataset and not the testing dataset to avoid contaminating and producing biases in the models.

6. ML/DL for alcohol and drugs use disorder analysis

Both AUDIT and DAST scores have promising clinical utility in diagnosing different physical and mental illnesses and have additional utility in learning about health behaviors. Consequently, examining these metrics, studying the related features, and forecasting future possibilities are important. With this objective, state-of-the-art ML/DL models were adopted. ML/DL models were implemented using the python library scikit-learn and Tensorflow to predict AUDIT and DAST scores. The prediction approach allowed for the control of the early intervention (e.g., BI and BT) for individuals with alcohol and drug use risks and timely referral to more intense substance abuse treatment (e.g., RT) for those with substance use disorders. To develop the optimal approach for predicting AUDIT and DAST scores, we implemented three steps:

1. We determined the best ML/DL-based prediction systems on our dataset to uncover the best predictive models.
2. Since our dataset contained group-level characteristics (SEX, VET, AGE), we used mixed-effects models that considered the impact group membership has on an outcome of interest. For example, we studied how the number of days of past-month drug use (DRUGDAYS) affected DAST scores among different sex types (SEX).
3. We compared mixed random and fixed-effect models for their ability to provide accurate predictions.

6.1. ML/DL algorithms

Predictive analytics are essential processes which use data analysis methods to create data-driven predictions. These processes employ statistical data analyses or ML/DL methods to develop forecasting models to estimate future observations. For example, we could estimate the probability of receiving RT based on their DAST score. No one algorithm works best for every problem, and it is especially relevant for predictive models. In this study, we compare the performances of several standard ML/DL multi-class classifiers based on supervised learning. Each has a different level of implementation complexity and can draw linear or non-linear classification borders.

For common ML techniques, we used (1.) Multinomial Logistic Regression (MLR) [34], (2.) Support Vector Machines (SVM) [35], (3.) Decision trees (DT) [36], (4.) Random Forest (RF) [37], and (5.) Gradient Boosting Decision trees (GBDT) [38]. These five classification models yield high accuracy in real-world applications.

MLR is used to forecast categorical placement for a target variable based on one or multiple predictors. It is a straightforward extension of binary logistic regression (a two classes classifier) that allows for more than two outcome variable categories. Like binary logistic regression, the MLR model employs maximum likelihood estimation to compute the probability of categorical membership. SVM is a classifier model that tries to draw the optimal boundaries (known as hyperplanes) between different classes in N-dimensional space. DT has tree-like constructions in which the nodes consider the modelś inputs and the leaves as decision outputs. The class label can be forecast with adequate reasoning when traversing the tree to classify a new observation. RF is an ensemble model that builds and merges different DT models; each DT is formed from a random subset of predictors. Its output will be the most popular class that draws the most votes from the DTs in the forest. Because of its simplicity and diversity, RF produces an overall better model than DT. Like the RF model, GBDT combines DTs to create a more robust model, but GBDT differs from RF in the way it builds its DTs. GBDT builds a DT one at a time, where each new one helps to correct errors made by the previously trained DT model, while RF builds and trains each DT independently.

The main advantage of traditional ML is its speed and relative simplicity. In addition, some of these algorithms are human interpretable, important for failure analysis, model improvement, and discovering insights and statistical regularities. Today, traditional ML algorithms are significantly overshadowed by DL, which has achieved state-of-art results for different classification problems. To study the effect of DL models on our study, we suggest using three different supervised models: (a.) Deep Neural Network (DNN) [39], (b.) Variational Auto-Encoder (VAE) [40] with KNN model [25] (VAE-KNN), and (c.) Graph Neural Network (GNN) [41].

The DNN model is designed to make intelligent decisions mimicking human brains. It links problem-solving processes in a chain of events, where the next process is activated once one process has solved a problem. In DNN, the input forwardly feeds from the input layer to the output layer over several hidden layers. Usually, ReLU, Sigmoid, or TanH are used as activation functions that dictate the feed-forward flow of data between the layers. The number of nodes in the output layer must be the same as the number of classes. SoftMax, used in the output layer, assigns decimal probabilities to each class in a multi-class problem. Those decimal probabilities must add up to 1. During training, the weight of each layer is updated using the backpropagation technique [42].

In the VAE-KNN model, the encoder of the VAE is used to extract the features as latent representations, while the KNN is then used for classification. VAE encodes the input data into latent representations and then reconstructs the input to learn meaningful representations. It uses the encoding–decoding process to impose the input as a probability distribution on the latent spaces, where the distribution of the output from the decoder matches the observed data. The Variance of Evidence Lower Bound (ELBO) is VAE's loss used to train the model [40].

After training VAE end-to-end, the encoder will have latent vectors that are lower dimensionality but with more informational features. From this new vector of features, the computed distances should be more significant, thus providing a way to choose better neighbors and, eventually, better classification performance.

GNN is a neural network that works on graphs. The graph is a data structure that has two primary ingredients: nodes (a.k.a. vertices), which are connected by the second ingredient (i.e., edges G(V; E)). The nodes can be conceptualized as graph entities or objects, and the edges are any relationship those nodes may have. For example, each record (row) consisting of a set of features for a data table is represented by a vertex on G, and an edge weight is taken as the euclidean distance between every two nodes. A model used to predict an attribute of each node in a graph is commonly known as node classification. For instance, each node can be labeled by a categorical class (binary or multiclass classification) or predict a continuous number (regression).

6.2. Features selection

Feature selection is essential for finding the ideal subset of features with crucial information and maximizing the model performance. In this study, we used the stepwise procedure [43] to simplify and interpret the models. Stepwise is a combination of two algorithms: forward selection and backward elimination. Both forward selection and backward elimination are simple algorithms that perform the variable selection by including or excluding (respectively) variables from the model. Both begin with an initial model (no model/full model) and apply a selection/elimination procedure under a particular criterion, such as Akaike's Information Criterion (AIC) or Bayesian Information Criterion (BIC).

AIC [44] is a modification of the log-likelihood and penalizes the likelihood of the data given by the model by taking model complexity into account. However, when the model involves many parameters or its likelihood is poor, the model's credibility will drastically decrease. BIC [45] has extended AIC by adopting a Bayesian perspective. It can be approximated as a reformulation of the AIC by considering the sample size. This implies that the likelihood of complicated models is less penalized when the models are derived from large samples. AIC attempts to realize the unknown model that has a high-dimensional reality.

To sum up, AIC measures the information loss by adopting a working model (considered model) instead of the actual model. On the other hand, the BIC indicates the actual model among a finite set of candidates. Thus, AIC is best for forecasting, while BIC is best for an explication as it lets consistent estimation of the fundamental data generating task.

6.3. Mixed effects modeling

Over the last three decades, mixed-effect models [46] have been frequently employed in many subjects in the biological, physical, and social sciences. They are robust tools when the data includes group-level trends, as they are flexible for combining information from various sources.

A mixed model is a statistical model encompassing both fixed-effects and random-effect terms. While the fixed-effects terms include variables whose significant values are all represented in the data file, the random-effects terms include variables whose importance in the data file are considered a random sample from a larger set of values. Random-effect variables are used to explain the dependent predictors in a model and comprise two types: crossed and nested variables, which are properties of the data, not the model.

The main difference between fixed and random-effects variables is that fixed-effects variables support forecasting only the categories of features employed for training. In contrast, random-effects variables allow predicting something related to the population from which the samples are drawn.

We take into consideration the random-effects factors side by side with the fixed-effects factors because the concern of our studies is not about experimental impacts present only in the people who participated in the experiment, but rather in impacts present in drug and alcohol misusers everywhere, either within the study area, or drug and alcohol misusers in general. We adopted two mixed-effects models: Mixed-effects Multinomial Logistic Regression (MMLR) and Mixed-effects Random Forests (MRF).

MMLR [47] is an extension of MLR that allows for more than two categories of the dependent variable. It uses a maximum marginal likelihood solution to evaluate the probability of categorical membership by applying quadrature to numerically integrate over the distribution of random effects. MRF [48] models have extended the use of RF to analyze hierarchical data. The model maintains the flexibility and capability of complex modeling patterns within the data and can handle both continuous and discrete covariates.

7. Performance metrics

Model evaluation is vital for building an effective ML/DL model and measuring how accurately it forecasts the expected output. The most frequent classification evaluation metric used is " Accuracy". The belief that a model is good with an accuracy of 99% is common. However, this metric alone does not tell the whole story, as it can still provide misleading results. This is where these additional performance evaluations come in, as they help elicit more meaning from the model. We used four metrics to evaluate the models, as follows.

7.1. Accuracy

Accuracy [49], in Eq. (2), represents the proportion of actual cases expected from all classes. Its measurement takes into account four elements. True-Positive (TP)/False-Positive (FP) values point to when the class is true and was classified as true/false, whereas True-Negative (TN) False-Negative (FN) values happen when the class is false and was classified as false/true.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

7.2. F1-score

F1-score [49] is a technique for connecting the recall and precision of the model by representing them in the harmonic average of a model. It is widely used for estimating many types of ML/DL classifiers. The F1-score is computed as follows.

$$F1 - score = 2 \times (\frac{Precision \times Recall}{Precision + Recall}) \tag{3}$$

where,

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

and

$$Recall = \frac{TP}{TP + FN}. \tag{5}$$

The F1 score is instrumental when you are dealing with imbalanced class problems. These are problems when one class can dominate the dataset. So, when dealing with imbalanced classes, the F1 score is a far superior metric compared to accuracy.

7.3. Cohen's kappa

Cohen's kappa statistic is an excellent measure that can handle multi-class and imbalanced class problems very well. Cohen's kappa [50] can be used to determine the degree of agreement between the model predictions and manually established forecasting, in which each classifies the same number of elements into finite categories. It is an excellent statistical measurement that can handle imbalanced and multi-class problems. Cohen's kappa is given as follows:

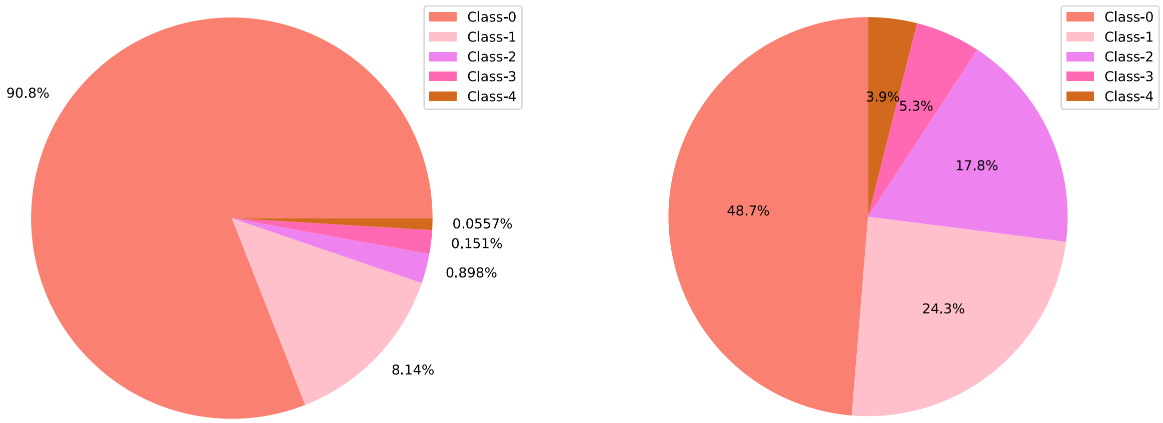
$$k = \frac{p_o - p_e}{1 - p_e}, \tag{6}$$

where p_o is the actual agreement, and p_e is the predicted agreement between the two methods. It basically tells how much better a classifier is performing over the performance of a classifier that simply guesses at random according to the frequency of each class. The value of Cohen's kappa is always less than or equal to 1, whereas the value of 0 or less refers to the useless classifier.

7.4. Receiver Operating Characteristic (ROC)

The ROC curve represents a graphical implementation of a classifier's performance rather than a single value like most other metrics. It is found by computing and plotting the TP rate against the FP rate for a single classifier at various thresholds. Area Under the Curve (AUC) explains the ROC curve's separability degree. Generally, an AUC value of 0.9 is deemed outstanding, while 0.7 to 0.8 is considered acceptable, and a weight of 0.5 cannot discriminate reasonably.

ROC curves are typically used in binary classification to study the output of a classifier. To extend ROC curve and ROC area to multi-label classification, it is necessary to binarize the output. One ROC curve can be drawn per label, but one can also draw a ROC curve by considering each element of the label indicator matrix as a binary prediction (micro-averaging). Another evaluation measure for multi-label classification is (macro-averaging), which gives equal weight to the classification of each label.



(a) (b)

Fig. 2. The frequency distribution of the target variable DAST, before (on the left) and after (on the right) applying the SMOTE-NC method. It is essential to mention that since oversampling the minority class might lead to overfitting, we did not exaggerate in increasing the number of samples in the minority classes, and we take into our consideration the percentages mentioned in the studies of the WHO report [51].

Table 3
The test accuracy of the imputation techniques for 22 features and the average accuracy of all these features.

Name of column	KNN	MICE	DataWig	MIDAS	GAIN
DAST	0.943	0.951	0.929	0.937	0.939
SEX	0.663	0.778	0.707	0.676	0.685
HISPANIC	0.931	0.944	0.981	0.987	0.991
RACE	0.255	0.228	0.621	0.663	0.683
VET	0.784	0.866	0.808	0.791	0.840
ACTIVE	0.771	0.899	0.811	0.801	0.866
DEPLOY	0.798	0.814	0.717	0.706	0.841
AUDIT	0.957	0.958	0.734	0.948	0.957
COSCREEN	0.983	0.973	0.979	0.978	0.991
BI	0.980	0.972	0.985	0.990	0.993
BT	0.974	0.982	0.991	0.992	0.995
RT	0.994	0.993	0.994	0.990	0.995
ANYALC	0.936	0.931	0.724	0.969	0.967
BINGEDAYS	0.970	0.966	0.742	0.974	0.993
DRUGDAYS	0.967	0.944	0.735	0.975	0.975
ALCDRUG	0.921	0.911	0.740	0.960	0.980
DAYSCOCAINE	0.916	0.923	0.742	0.942	0.954
MARYJDAYS	0.924	0.914	0.812	0.912	0.943
OTHERDRUGS	0.899	0.890	0.802	0.902	0.911
INJECT	0.781	0.745	0.722	0.754	0.771
TOBMONTH	0.984	0.961	0.955	0.975	0.982
AGE	0.210	0.211	0.345	0.400	0.415
Average	0.843	0.853	0.799	0.874	0.894

8. Results and discussion

Most real-life data retrieved from an administrative database has missing data, incorrect data, duplicated data, outliers, and uneven data distribution. Therefore, it is required to rectify these problems before analyzing data and getting accurate results. The initial dataset we retrieved had 6978 observations and 52 features (see Table 12). As the initial step, we filled some null values based on logical relationships between some of these features, such as between BIRTH and AGE or between AUDIT and ANYALC. After that, the features in which more than 70% of their observations were null values were eliminated. Therefore, 22 features remain. Additionally, we determined the data type in each column as numerical or categorical. SEX, DAST, AUDIT, VET, ACTIVE, DEPLOY, COSCREEN, BI, BT, RT, INJECT, and TOBMONTH were categorical data, whereas the rest were numerical. We need to convert these categorical variables to numbers such that the model is able to understand and extract valuable information [52].

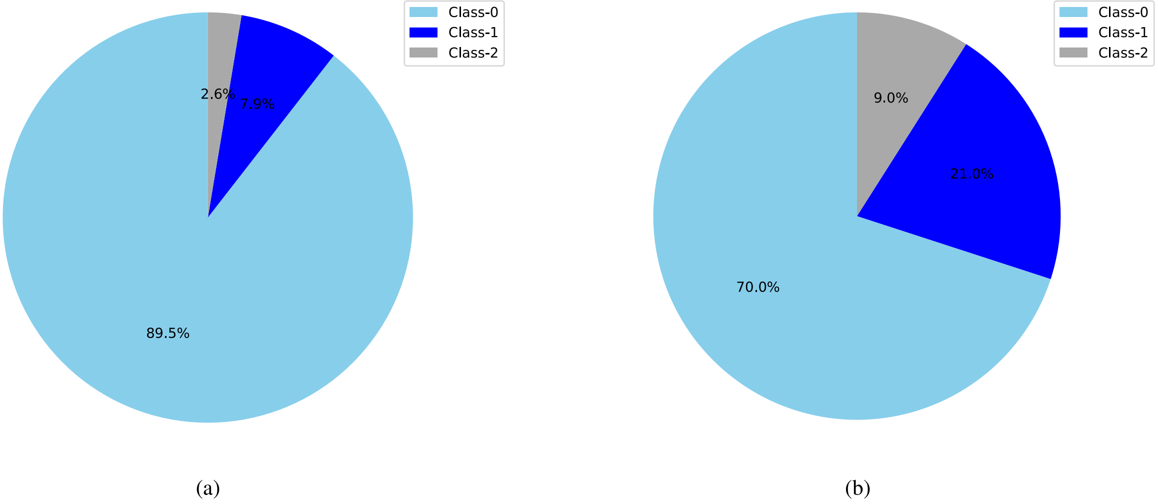
We train the classification models independently in two different stages. In Stage (1), we drop the rows that contain missing values, resulting in 3676 rows (where 2941 for training and 735 for testing). In Stage (2), two pre-processing applications were employed: imputing missing data and over-sampling (augmentation) of our training set to increase the number of samples and have a more balanced dataset (where 18765 for training and 735 for testing).

The (80%–20%) train-test split procedure is used to estimate the performance of the ML/DL classification algorithms. Empirical studies show that the best results are obtained if we use 20%–30% of the data for testing and the remaining 70%–80% of the data for training. Ideally, if we had enough data, we would set aside a validation set and use it to assess the performance of our prediction models [53]. However, this is not possible since our dataset is considered relatively small to use in training ML/DL models. To address this problem, we used k-fold cross-validation [54], where we divided the training dataset into 10 folds ($k = 10$), each fold being (10%) of the whole training dataset. A bias–variance trade-off is associated with the choice of k in k-fold cross-validation. Given these considerations, one performs k-fold cross-validation using $k = 5$ or $k = 10$. These values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor very high variance [55]. In our case, a 20% validation set ($k = 5$) might reduce the size of the training set below the desired level, while 10% ($k = 10$) provide enough variance in the training set and does not affect the size of the training dataset.

Dealing with missing values can be challenging, as it requires a careful examination of the data to identify the type and pattern of missingness, supplemented by a clear understanding of different imputation methods. Our assumption about missing data was MCAR, where the probability of being missing is the same for all cases. First, we explored the performance of the five different imputation techniques (see Section 5.1) in terms of the ability to impute the correct values in each column. Then we calculated the average accuracy for each technique. Table 3 shows that GAIN performed better than other techniques.

We used the SMOTE-NC procedure to create synthetic examples to address the class imbalance in the categorical targets (DAST/AUDIT). For instance, Fig. 2 and Fig. 3 displays the results of running SMOTE-NC against the minority class.

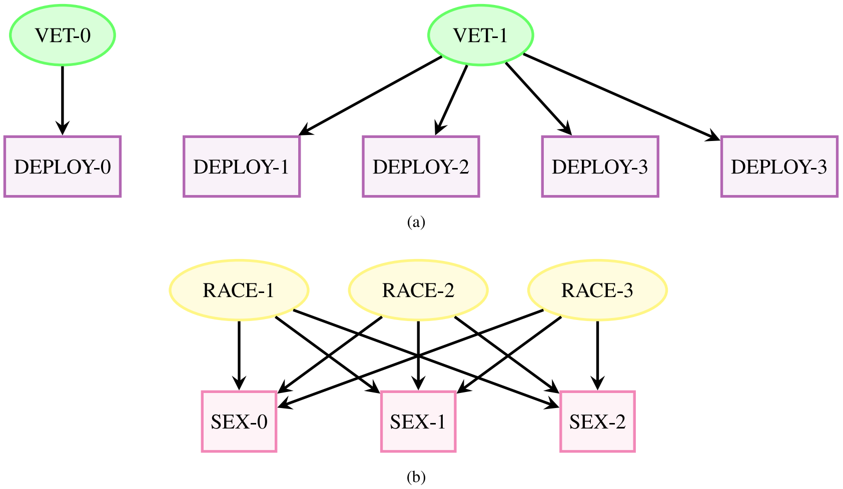
All experiments were performed in an Intel(R) Xeon(R) W-2102, CPU @ 2.90 GHz ×4, 7.5 GiB RAM, Ubuntu 18.04.5 LTS, Quadro RTX 8000/PCIe/SSE2.



(a)

(b)

Fig. 3. The frequency distribution of the target variable AUDIT, before (on the left) and after (on the right) applying the SMOTE-NC method. It is essential to mention that since oversampling the minority class might lead to overfitting, we did not exaggerate in increasing the number of samples in the minority classes, and we take into our consideration the percentages mentioned in the studies of the WHO report [51].



(a)

(b)

Fig. 4. The relationships between the levels of random-effects: (a) Nested random-effects (1|VET/DEPLOY), (b) Crossed random-effects (1|RACE)+(1|SEX).

Table 4
Architectural details of the VAE model used for VAE-KNN classifier,. Dense[n] is a Dense layer with n units. Batch Normalization(BN) , ReLU is Rectified Linear Units (ReLU) activation and FC[c] is a fully connected layer with c output classes, for which there are 5 for DAST and 3 for AUDIT.

Encoder
Input Layer: Dense[22], BN, ReLU
Hidden Layer (1): Dense[50], BN, ReLU
Hidden Layer (2): Dense[50], BN, ReLU
Output Layer: FC [5/3] (Mean), || FC [5/3] (Std.dev)
Decoder
Input Layer: Dense[5/3], BN, ReLU
Hidden Layer (1): Dense[50], BN, ReLU
Hidden Layer (2): Dense[50], BN, ReLU
Output Layer: Dense[22], BN, ReLU

Table 5
The architectural details of the GNN model. GCN[l] is a Graph Convolution Network layer with l features, Dense[n] is a Dense layer with n units. Batch Normalization(BN), Dropout[x] is a dropout layer with probability x, ReLU is Rectified Linear Units (ReLU) activation, and FC[c] is a fully connected layer with c output classes, for which there are 5 for DAST and 3 for AUDIT.

GNN Classifier
Input Layer: GCN [22], ReLU, Dropout[0.5]
Hidden Layer (1): Dense[50], BN, ReLU
Hidden Layer (2): Dense[50],BN, ReLU
Output Layer: Dense[5/3], BN, Softmax

8.1. Experiments

8.1.1. Experiment (1): we applied the ML/DL models for classification of DAST/AUDIT scores using all the features as predictors

In this Experiment, DAST/AUDIT was classified by employing traditional ML classifiers built using the `scikit-learn` library in Python. In addition, we also applied deep learning models. Our DNN classifier consists of 4 layers. Layer 1 was the input layer with 22 units (number of the features), layer 2 and layer 3 were hidden layers with 50 units, and layer 4 was the output layer with 5/3 units (number of the classes in the target variable DAST/AUDIT). The hidden layers underwent the ReLU activation function, and the output layer underwent SoftMax activation function calculations. The model was built using the `tensorflow.keras` library in Python.

Table 6
The performance evaluation of ML/DL models in Experiment (1) of Stage (1) and Stage (2) for classifying DAST score. For each classifier in Stage (1) and Stage (2), we computed the value of Accuracy, Cohen's Kappa, and F1 scores, where the F1 score was computed per class.

DT Classifier							RF Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			1.00			0.99			0.99			0.93
C1			0.96			0.99			0.87			0.97
C2	0.991	0.872	0.92	0.976	0.988	0.99	0.972	0.461	0.44	0.923	0.566	0.99
C3			0.22			0.87			0.00			0.49
C4			0.00			0.63			0.00			0.16
MLR Classifier							DNN Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			1.00			0.98			0.96			1.00
C1			0.93			0.97			0.00			1.00
C2	0.989	0.965	0.92	0.958	0.974	0.97	0.923	0.045	0.25	0.990	0.989	1.00
C3			0.77			0.80			0.00			0.94
C4			0.00			0.59			0.00			0.88
GBDT Classifier							SVM Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			1.00			1.00			1.00			0.99
C1			0.95			1.00			0.91			0.98
C2	0.988	0.962	0.93	0.997	0.997	1.00	0.986	0.899	0.73	0.979	0.988	0.99
C3			0.44			0.98			0.91			0.94
C4			0.00			0.96			0.00			0.83
VAE-KNN Classifier							GNN Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			0.98			1.0			0.98			0.98
C1			0.82			1.00			0.68			0.98
C2	0.967	0.877	0.75	0.996	0.997	1.00	0.956	0.440	0.25	0.963	0.967	0.99
C3			0.77			0.98			0.00			0.82
C4			0.00			0.96			0.00			0.55

Table 7
The performance evaluation of ML/DL models in Experiment (1) of Stage (1) and Stage (2) for classifying AUDIT score. For each classifier in Stage (1) and Stage (2), we computed the value of Accuracy, Cohen's Kappa, and F1 scores, where the F1 score was computed per class.

DT Classifier							RF Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			0.99			0.99			0.99			0.98
C1	0.986	0.734	0.89	0.976	0.949	0.95	0.977	0.600	0.68	0.966	0.938	0.93
C2			0.00			0.93			0.50			0.93
MLR Classifier							DNN Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			0.99			0.97			0.98			0.99
C1	0.986	0.882	0.79	0.953	0.902	0.91	0.957	0.000	0.00	0.988	0.979	0.97
C2			1.00		0.902	0.89			0.00			0.98
GBDT Classifier							SVM Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			0.99			0.99			0.99			0.98
C1	0.986	0.868	0.84	0.991	0.985	0.98	0.986	0.868	0.84	0.972	0.937	0.94
C2			0.84			0.99			0.77			0.93
VAE-KNN Classifier							GNN Classifier					
Stage (1)				Stage (2)			Stage (1)			Stage (2)		
Class	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score	Accuracy	Cohen's Kappa	F1-score
C0			0.96			0.99			0.97			0.99
C1	0.923	0.000	0.00	0.991	0.985	0.98	0.912	0.918	0.00	0.976	0.948	0.95
C2			0.00			0.99			0.00			0.92

Table 8
The performance evaluation of the ML/DL models in Experiment (2) of Stage (1) for classifying DAST after applying bidirectional stepwise for both criteria, AIC and BIC, as shown on Eq. (7), and Eq. (8). We computed the value of Accuracy, Cohen’s Kappa, and F1 scores, where the F1 score was computed per class.

DT Classifier							RF Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.992	0.838	1.00	0.995	0.976	1.00	0.990	0.669	1.00	0.995	0.980	1.00
C1			0.97			0.98			0.96			0.99
C2			0.83			0.88			0.83			0.94
C3			0.67			0.33			0.00			0.33
C4			0.00			0.00			0.00			0.00
MLR Classifier							DNN Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.990	0.938	1.00	0.988	0.895	1.00	0.993	0.816	1.00	0.994	0.932	1.00
C1			0.93			0.93			0.97			0.98
C2			0.83			0.94			0.83			0.88
C3			0.75			0.00			0.75			0.00
C4			0.40			0.00			0.00			0.00
GBDT Classifier							SVM Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.994	0.864	1.00	0.995	0.952	1.00	0.990	0.794	1.00	0.987	0.892	1.00
C1			0.98			0.98			0.93			0.91
C2			0.86			0.94			0.83			0.71
C3			0.75			0.33			0.75			0.00
C4			0.00			0.00			0.00			0.00
VAE-KNN Classifier							GNN Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.994	0.872	1.00	0.966	0.801	0.98	0.991	0.843	1.00	0.993	0.960	1.00
C1			0.96			0.79			0.94			0.97
C2			0.86			0.70			0.77			0.82
C3			0.89			0.00			0.89			0.33
C4			0.00			0.67			0.00			0.00

Table 9
The performance evaluation of the ML/DL models in Experiment (2) of Stage (1) for classifying AUDIT after applying bidirectional stepwise for both criteria, AIC and BIC, as shown on Eq. (11), and Eq. (12). We computed the value of Accuracy, Cohen’s Kappa, and F1 scores, where the F1 score was computed per class.

DT Classifier							RF Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.989	0.890	1.00	0.987	0.880	0.99	0.989	0.870	1.00	0.987	0.845	0.99
C1			0.89			0.88			0.90			0.90
C2			0.80			0.84			0.80			0.78
MLR Classifier							DNN Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.985	0.899	1.00	0.953	0.902	0.97	0.986	0.829	1.00	0.988	0.979	0.99
C1			0.90			0.91			0.85			0.97
C2			0.88			0.89			0.50			0.98
GBDT Classifier							SVM Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.990	0.892	1.00	0.988	0.870	0.99	0.984	0.868	0.99	0.984	0.805	0.99
C1			0.89			0.90			0.81			0.85
C2			0.80			0.84			0.62			0.62
VAE-KNN Classifier							GNN Classifier					
AIC				BIC			AIC			BIC		
Class	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score	Accuracy	Cohen’s Kappa	F1-score
C0	0.940	0.617	0.97	0.991	0.985	0.99	0.988	0.878	1.00	0.976	0.948	0.99
C1			0.59			0.98			0.87			0.95
C2			0.67			0.99			0.74			0.92

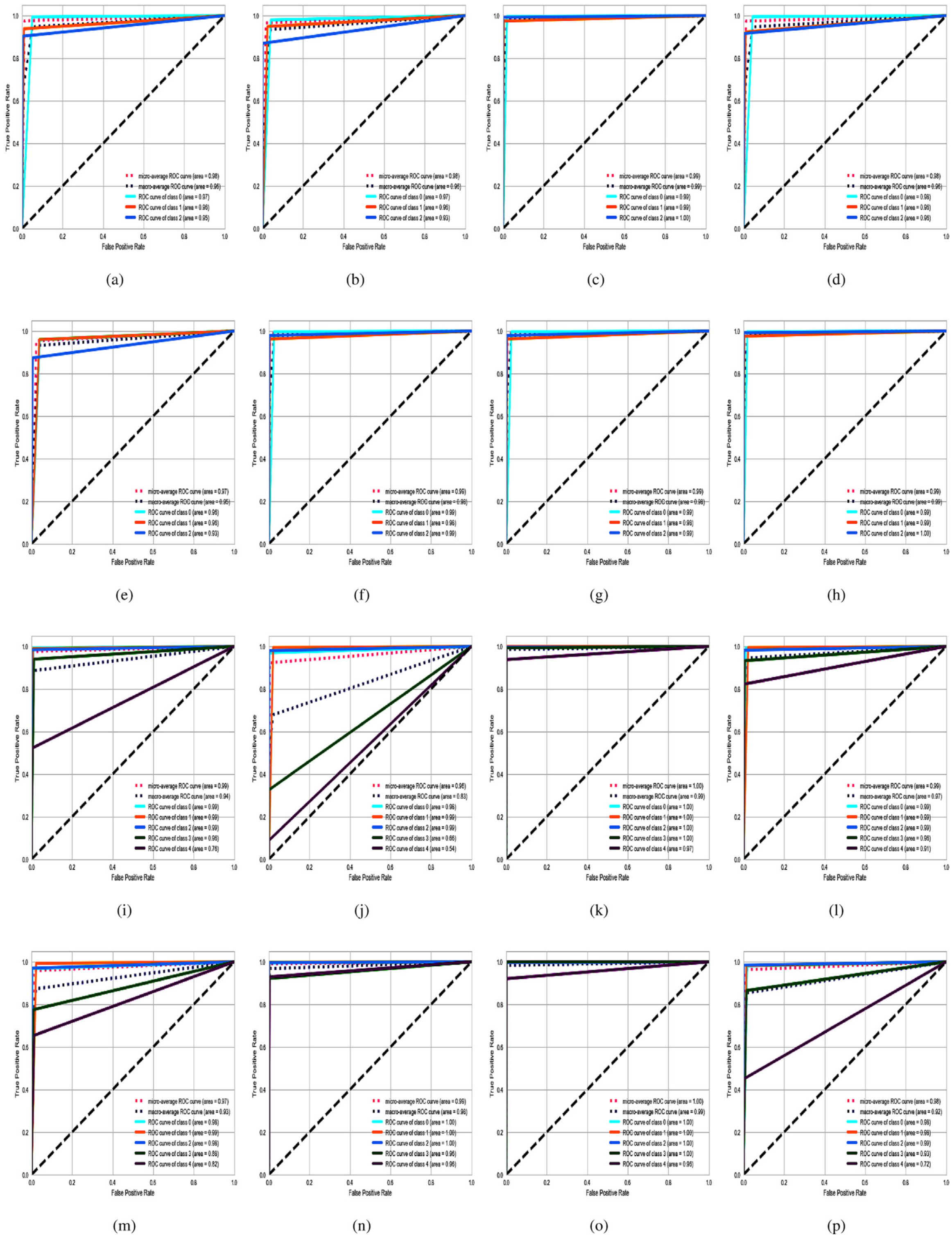


Fig. 5. ROC curves for Experiment (1) of Stage (1); (a) DT, (b) RF, (c) GBDT, (d) SVM, (e) MLR, (f) DNN, (g) VAE-KNN, (h) GNN, and Stage (2); (i) DT, (j) RF, (k) GBDT, (l) SVM, (m) MLR, (n) DNN, (o) VAE-KNN, (p): GNN for classification DAST score.

The architectural details of the VAE model used in the VAE-KNN classifier are shown in Table 4. Once the VAE was trained, we generated latent spaces from the encoder and then used KNN with ($k = 5$), where k is the number of nearest neighbors to include in the majority of the voting process, to get the best prediction in the testing phase. Next, we created the GNN model using the library `tensorflow.Kerasgraph` in python with StellarGraphś GCN Supervised Graph Classification class [56]. The architectural details of the model are shown in Table 5.

The performance evaluation metrics, such as classification accuracy, F1-score, and Cohen's Kappa, were automatically computed to evaluate the performance of these classifiers and tabulated into Tables 6 and 7. Additionally, we visualized the performance of the multi-class classification problem by using a ROC/AUC curve that shows the relationship between the true positive rate (sensitivity) against the false positive rate ($1 - specificity$) for the different possible cut points of the target. Figs. 5 and 6 show the ROC graphs with 8 classifiers labeled (a) through (h) for

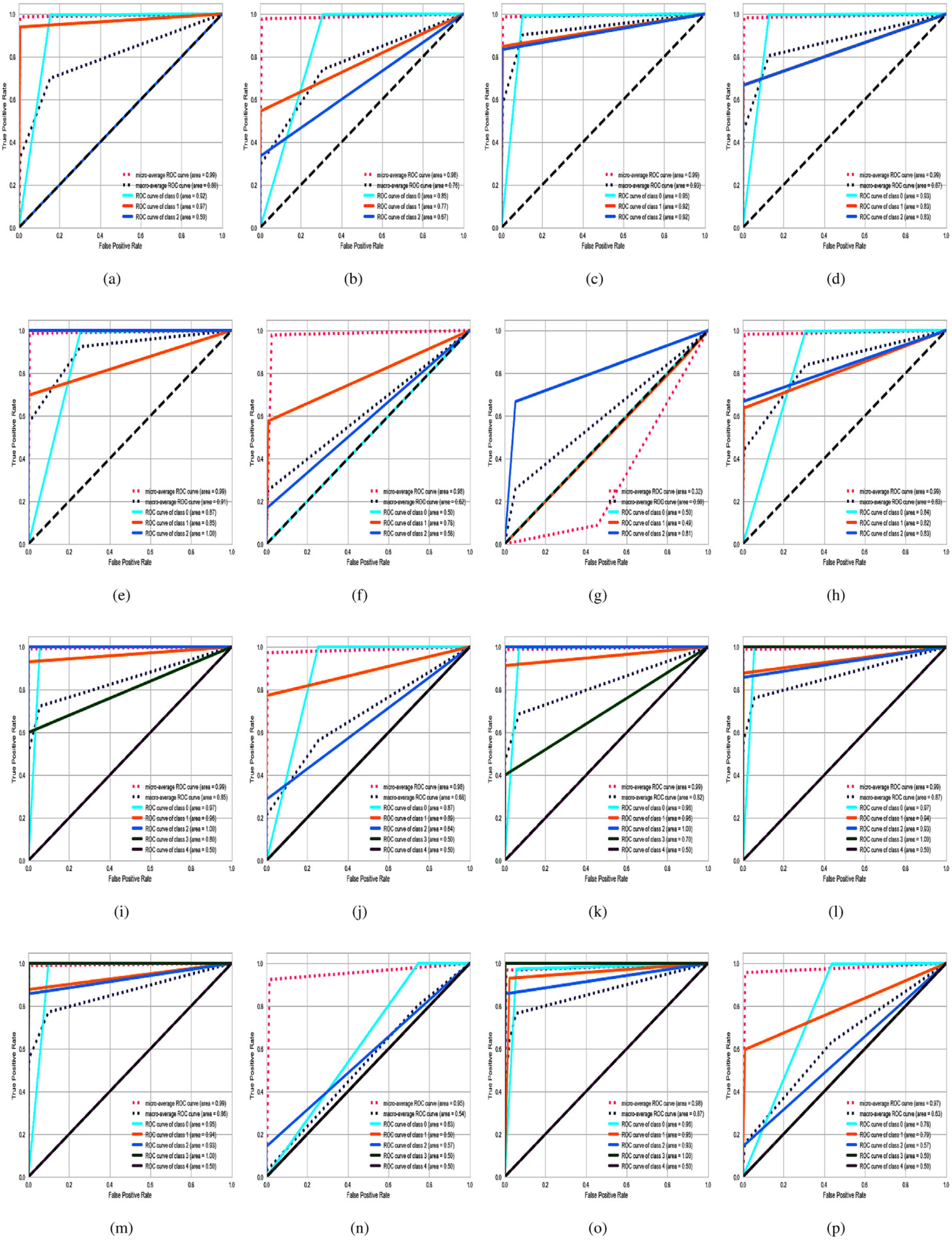
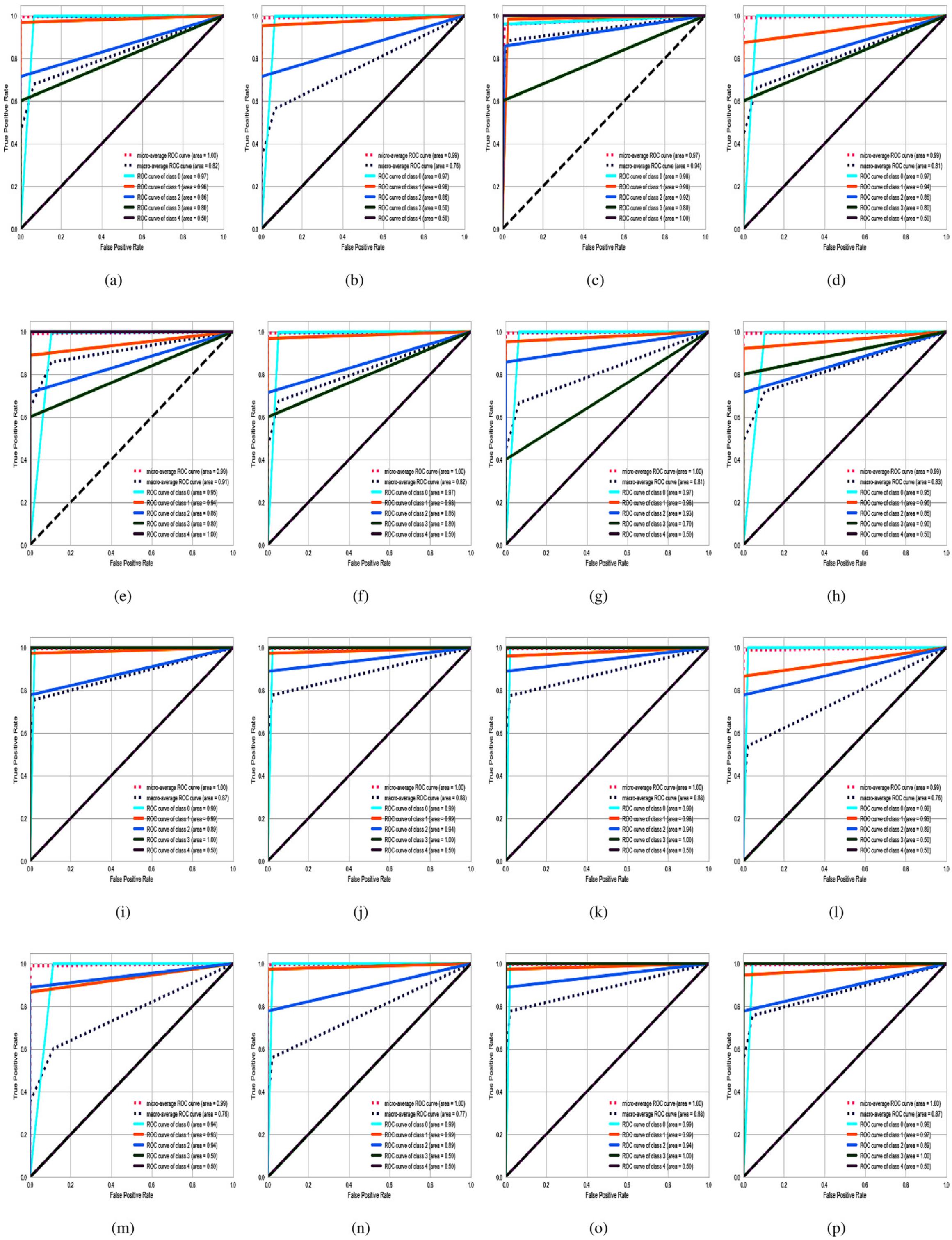


Fig. 6. ROC curves for Experiment (1) of Stage (1); (a) DT, (b) RF, (c) GBDT, (d) SVM, (e) MLR, (f) DNN, (g) VAE-KNN, (h) GNN, and Stage (2); (i) DT, (j) RF, (k) GBDT, (l) SVM, (m) MLR, (n) DNN, (o) VAE-KNN, (p): GNN for classification AUDIT score.

Stage (1) and (i) through (p) for Stage (2) covering both targets, DAST and AUDIT score. In Stage (1) "without pre-processing," results showed that the classification with the imbalanced dataset produced high accuracy. Still, the test data had a low F1 score for the minority classes, and most models yielded lower Cohen's Kappa values. ROC/AUC curves show an unequal distribution of classes in the dataset in such a manner that the rare class constituted a minimal amount of data, and the classifiers could not predict the rare class very well. GBDT performed well compared to other classifiers. Given that GBDT is a sequential process, every subsequent iteration focuses on the incorrect prediction from the previous iteration.

Most of the traditional ML classifiers ignore the minority class and, in turn, perform poorly. Therefore, in Stage (2), "with pre-processing," the data augmentation technique "SMOTE-NC" was used to address the imbalanced classification problem and synthesize new samples for our dataset.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

Fig. 7. ROC curves for Experiment (2) of Stage (1) for classification DAST based on AIC criterion; (a) DT, (b) RF, (c) GBDT, (d) SVM, (e) MLR, (f) DNN, (g) VAE-KNN, (h) GNN, and AIC criterion; (i) DT, (j) RF, (k) GBDT, (l) SVM, (m) MLR, (n)DNN, (o)VAE-KNN, (p) GNN.

For Stage (1), the results in Tables 6 and 7 show that traditional ML models, such as the GBDT classifier, perform better than DL models. Additionally, the F1 score was poor for predicting minority classes in the ML/DL models; the Cohen ś Kappa value was also low. On the other hand, the performance of all the classifiers improved in Stage (2) after applying pre-processing techniques. In addition, the performance of the deep learning models ‘‘DNN’’ and ‘‘VAE-KNN’’ were slightly better than most of the ML methods when trained with large sample sizes (in Stage (2)), and the performance of the ROC/AUC curves in Figs. 5 and 6 improved as well.

8.1.2. *Experiment (2): we applied the features selection model before training the ML/DL models*

We used bi-directional stepwise as a feature selection technique (a combination of forwarding selection and backward elimination) to improve the classification and create a more straightforward model to

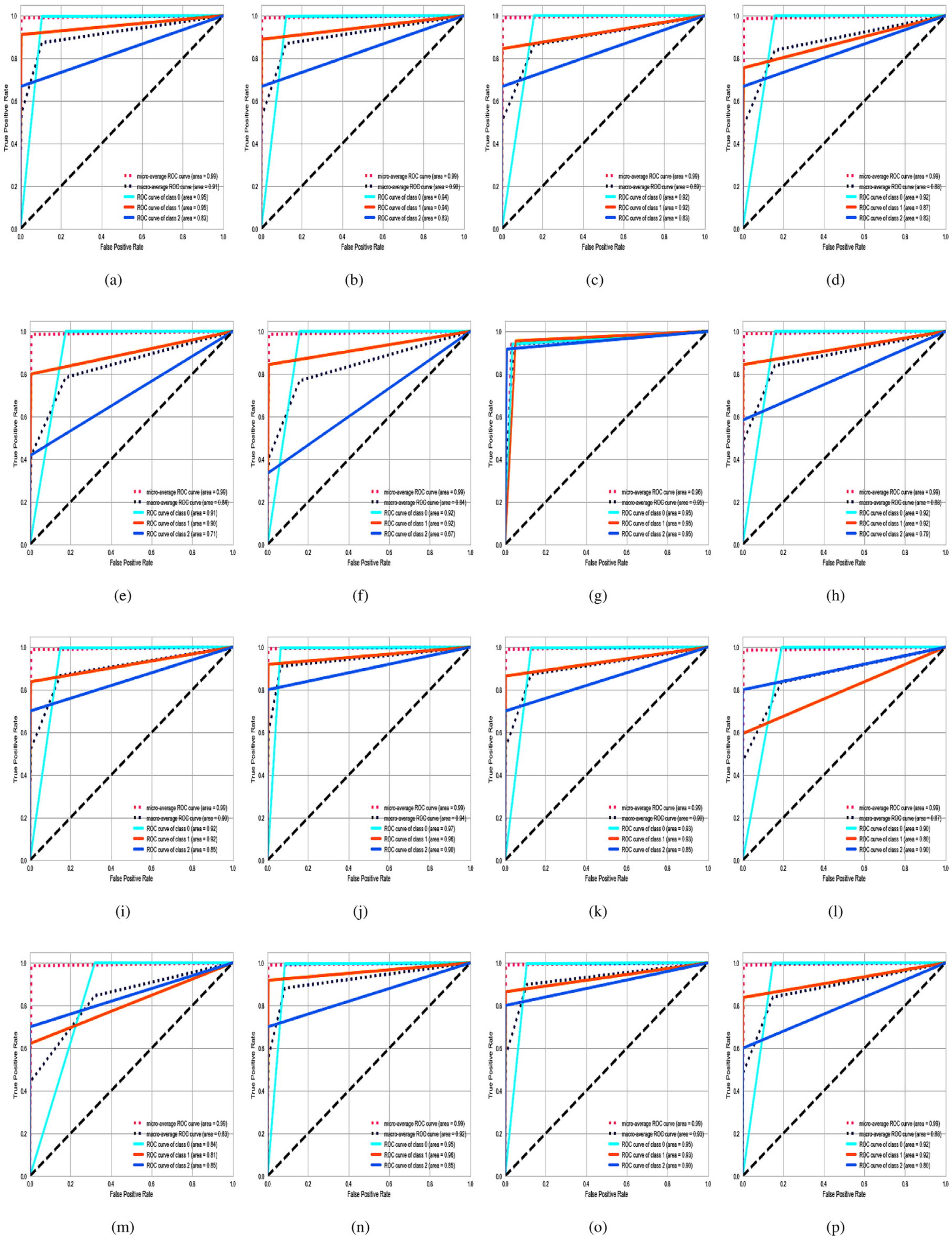


Fig. 8. ROC curves for Experiment (2) of Stage (1) for classification AUDIT based on AIC criterion; (a) DT, (b) RF, (c) GBDT, (d) SVM, (e) MLR, (f) DNN, (g) VAE-KNN, (h) GNN, and AIC criterion; (i) DT, (j) RF, (k) GBDT, (l) SVM, (m) MLR, (n) DNN, (o) VAE-KNN, (p) GNN.

interpret by decreasing the modelś complexity and handling overfitting, resulting in including all the features in the models. The main goal is to find the most relevant features affecting the DAST/AUDIT score, thus allowing us to build applicable models. The results of running the stepwise model with the AIC/BIC criterion in both stages are shown as follows:

$$\begin{aligned}\textbf{DAST\_Stage(1)\_AIC} \sim & \; SEX + RACE + AUDIT + BI + \\ & BT + RT + DRUGDAYS + TOBMONTH\end{aligned} \tag{7}$$

$$\textbf{DAST\_Stage(1)\_BIC} \sim AUDIT + BI + BT + RT + DRUGDAYS \tag{8}$$

$$\begin{aligned}\textbf{DAST\_Stage(2)\_AIC} \sim & \; SEX + RACE + VET + AUDIT + \\ & COSCREEN + BI + BT + RT + ANYALC + BINGEDAYS \\ & + DRUGDAYS + ALCDRUGS + MARYJDAYS + INJECT \\ & + AGE + TOBMONTH\end{aligned} \tag{9}$$

Table 10
The performance evaluation of the mixed-effect models (MEMLR and MERF) in Experiment (3) of Stage (1) for classifying DAST score. We computed the value of Accuracy, and F1 scores, where the F1 score was computed per class.

Model	MEMLR		MERF	
	Accuracy	F1 Score	Accuracy	F1 Score
DAST ∼ AUDIT+BI+BT+RT+DRUGDAYS+TOBMONTH+RACE+SEX	0.985	1.00 0.93 0.83 0.75 0.40	0.990	1.00 0.96 0.83 0.00 0.00
DAST ∼ AUDIT+BI+BT+RT+DRUGDAYS+TOBMONTH+VET+DEPLOY+RACE+SEX	0.971	0.99 0.89 0.77 0.70	0.980	0.98 0.92 0.80 0.00
AUDIT+BI+BT+RT+DRUGDAYS+TOBMONTH+VET+DEPLOY+RACE+SEX	0.981	0.20 1.00 0.93 0.83 0.74 0.37	0.988	0.00 1.00 0.93 0.81 0.00 0.00
DAST ∼ AUDIT+BI+BT+RT+DRUGDAYS+TOBMONTH+(1|RACE)+(1|SEX)	0.989	1.00 0.94 0.85 0.76 0.50	0.991	1.00 0.97 0.87 0.20 0.10
DAST ∼ DAST ∼ AUDIT+BI+BT+RT+DRUGDAYS+TOBMONTH+(1|VET/DEPLOY)	0.975	0.99 0.90 0.78 0.72 0.30	0.984	0.99 0.93 0.81 0.00 0.00
DAST ∼ AUDIT+BI+BT+RT+DRUGDAYS+TOBMONTH+(1|VET/DEPLOY)+(1|RACE)+(1|SEX)	0.983	1.00 0.95 0.86 0.77 0.54	0.990	1.00 0.96 0.87 0.20 0.20

$$\begin{aligned}\textbf{DAST\_Stage(2)\_BIC} \sim {} & SEX + RACE + VET + AUDIT + \\ & COSCREEN + BI + BT + RT + ANYALC + BINGEDAYS \\ & + DRUGDAYS + MARYJDAYS + INJECT + AGE \\ & + TOBMONTH\end{aligned} \tag{10}$$

$$\begin{aligned}\textbf{AUDIT\_Stage(1)\_AIC} \sim {} & DAST + BI + BT + RT + ANYALC \\ & + BINGEDAYS + DRUGDAYS\end{aligned} \tag{11}$$

$$\textbf{AUDIT\_Stage(1)\_BIC} \sim DAST + BI + BT + RT + ANYALC \tag{12}$$

$$\begin{aligned}\textbf{AUDIT\_Stage(2)\_AIC} \sim {} & DAST + SEX + HISPANIC + RACE \\ & + VET + ACTIVE + COSCREEN + BI + BT + RT + ANYALC \\ & + BINGEDAYS + DRUGDAYS + ALCDRUGS \\ & + DAYSCOCAINE + MARYJDAYS + AGE + TOBMONTH\end{aligned} \tag{13}$$

$$\begin{aligned}\textbf{AUDIT\_Stage(2)\_BIC} \sim {} & DAST + SEX + VET + COSCREEN \\ & + BI + BT + RT + ANYALC + BINGEDAYS + DRUGDAYS \\ & + ALCDRUGS + DAYSCOCAINE + MARYJDAYS + AGE \\ & + TOBMONTH\end{aligned} \tag{14}$$

The equations show that since they exist in all DAST equations, AUDIT, BI, BT, RT, and DRUGDAYS are the most effective features for predicting the DAST score. On the other hand, DAST, BI, BT, RT, and ANYALC are the most effective features for predicting the AUDIT score.

After applying the stepwise technique, we calculated Accuracy, F1-score, and Cohen's Kappa for all the classifiers. The results of Experiment (2) for Stage (1) were recorded in Tables 8 and 9. Moreover, the results of the ROC/AUC curves are shown in Figs. 7 and 8. The results show a slight performance improvement compared to the results of Stage (1) in Experiment (1). We did not record the results in Stage (2) after applying the feature selection technique because the difference was insignificant compared to Experiment (1).

8.1.3. *Experiment (3): we applied mixed-effect for classification of DAST/AUDIT scores*

In this experiment, we examined how the fixed-effects features such as BI, BT, RT, DRUGDAYS, MARYJDAYS, INJECT, TOBMONTH, etc., and the random-effects features such as race (RACE), gender (SEX), military veteran status (VET) and deployment history (DEPLOY), effected (DAST/AUDIT). We considered a model implementing the (VET/DEPLOY) subset relations between the observed impacts on our target to be the nested effects model. In contrast, a model implementing the (RACE/SEX) subset relations served as the crossed random effects model (see Fig. 4). We aim to show whether or not the mixed-effects models are as helpful as other predictive models we applied in Experiment (2). We used MMLR/MRF to predict the target (DAST/AUDIT) and used Accuracy and F1 score as performance metrics. The results were recorded in Table 10, and Table 11. The results show that the mixed-effects models outperform the traditional ML models (fixed-effects model) in Stage(1).

9. Conclusion and future work

This study sought to determine the applicability of ML/DL techniques within the context of SBIRT, a substance use prevention approach often housed within primary care clinics. Results of our experiments showed that accurate classification of alcohol and drug use screening instrument scores are best accomplished with mixed-effects models following the imputation of missing data by the GAIN method. Although mixed models are commonly employed in studies of electronic health records (EHRs), as in the case of the COVID-19 pandemic [14] the use of the GAIN method in this context is novel; however, we show that the GAIN method may be an efficient and accurate way of analyzing data from EHRs that contain many missing values.

ML/DL has become increasingly popular in health care. With the growing acceptance of electronic health records in clinics across the

Table 11
The performance evaluation of the mixed-effect models (MEMLR and MERF) in Experiment (3) of Stage (1) for classifying AUDIT score. We computed the value of Accuracy, and F1 scores, where the F1 score was computed per class.

Model	MEMLR		MERF	
	Accuracy	F1 Score	Accuracy	F1 Score
AUDIT ∼ DAST+BI+BT+RT+ANYALC+BINGEDAYS+DRUGDAYS+RACE+SEX	0.984	1.00 0.91 0.88	0.988	1.00 0.91 0.81
AUDI T ∼ DAST+BI+BT+RT+ANYALC+BINGEDAYS+DRUGDAYS+VET+DEPLOY	0.981	1.00 0.89 0.85	0.985	1.00 0.87 0.80
AUDIT ∼ DAST+BI+BT+RT+ANYALC+BINGEDAYS+DRUGDAYS+VET+DEPLOY+RACE+SEX	0.982	1.00 0.90 0.86	0.987	1.00 0.89 0.80
AUDIT ∼ DAST+BI+BT+RT+ANYALC+BINGEDAYS+DRUGDAYS+(1|RACE)+(1|SEX)	0.987	1.00 0.95 0.89	0.990	1.00 0.94 0.83
AUDIT ∼ DAST+BI+BT+RT+ANYALC+BINGEDAYS+DRUGDAYS+(1|VET/DEPLOY)	0.92	1.00 0.99 0.88	0.98	1.00 0.91 0.84
AUDIT ∼ DAST+BI+BT+RT+ANYALC+BINGEDAYS+DRUGDAYS+(1|VET/DEPLOY)	0.98	1.00 0.93 0.88	0.98	1.00 0.92 0.82

Table 12
The types of datasets used in this study include the following: (a) The original dataset "Raw Survey Dataset" with 52 features and 6978 records, (b) The dataset with 22 features and 6978 records, called "Cleaned Dataset," which we applied initial pre-possessing techniques to, and where all the columns were having more than 70% of missing values were discarded, (c) The dataset with 22 features and 3676 records used in Stage (1), where we dropped the rows which contained missing values, and (d) The dataset with 22 features and 19500 records used in Stage (2), where we applied the imputation and over-sampling methods.

Type of dataset	Name of features	Number of features	Number of samples
Raw Survey Dataset	SEX, HISPANIC, RACE, VET, ACTIVE, DEPLOY, MILFAM, COSCREEN, SUICIDEATTEMPT, BI, BT, RT, ANYALC, BINGEDAYS, DRUGDAYS, ALCDRUGS, DAYSCOCAINE, MARYJDAYS, ANYOPIATEDAYS, METHADONE, HALLUC, METHDAYS, OTHERDRUGS, INJECT, WHERELIVE, PREGNANT, CHILDREN, JOBTRAIN, EDUC, EMPLOY, INCOME, ARRESTED, CRIMES, HEALTHSTAT, ANYSEX, SEXCONTACT, SEXUNPROTECT, EVERHIVT, HIVRESULT, DEPRESSDAYS, ANXIETYDAYS, HALLUCINATE, ATTENDOTHER, ATTENDAA, AATIMES, OTHERTIMES, FAMILYINT, SBIRTCONT, AGE, TOBMONTH, AUDIT, DAST.	52	6978
Cleaned Dataset	SEX, HISPANIC, RACE, VET, ACTIVE, DEPLOY, AUDIT, ALCDRUG, COSCREEN, BI, BT, RT, ANYALC, BINGEDAYS, DRUGDAYS, MARYJDAYS, DAYSCOCAINE, INJECT, AGE, TOBMONTH, AUDIT, DAST	22	6978
Dataset in Stage (1)	"	22	3676
Dataset in Stage (2)	"	22	19500

Note: " means the same as above.

United States and the rapid acceleration of computing power, it is possible to understand factors that predict various health outcomes [57]. However, the application of ML/DL algorithms within the context of SBIRT has been sparse. The present study shows how ML/DL can be used for SBIRT patient data to predict alcohol and drug use outcomes.

In studies using ML/DL, it is common to report F1 scores and prediction accuracy indices. For example, reports of ML/DL algorithms to predict disease outcomes using electronic medical records have been published with F1 scores of 0.81 and prediction accuracy indices of 0.92 [58]. However, our most successful models for AUDIT and DAST prediction had F1/prediction accuracy indices of 0.99/0.94 and 0.99/0.93. As such, the results of our experiments may indicate that (a) using the GAIN method for missing data imputation and (b) using military service status as a predictor of health outcomes may enhance model precision.

Although much discussion and additional analysis are needed to refine and enhance the predictive capability of ML/DL models within the context of SBIRT and the broader health care landscape, there is also a need to encourage the translation of ML/DL models into clinical practice [59]. Creating machine/deep-learning-based clinical tools for medical providers tasked with screening for and providing brief interventions for alcohol and drug use is needed. For example, suppose researchers created a mobile application that included input fields for demographic characteristics. In that case, the clinician could input patient characteristics in the application, and the trained/tested ML/DL algorithm would automatically calculate the risk for AUDIT and DAST severity scores. Engaging in research translation, as previously described, could corroborate self-reported patient data to enhance health care delivery within the context of SBIRT.

Some limitations accompany the analysis and interpretation of data in this study. First, data collected for this study were based on self-reports from patients in primary care clinics and may not necessarily reflect actual alcohol or drug use problems. In addition, patients' self-reported data are subject to recall bias and social desirability bias. Second, because this study was based on a cross-sectional design, we could not make causal inferences about the relationships examined in the ML/DL models. Third, although our GAIN method for data imputation was successful, this study was limited because the medical records we used for analysis contained many null values. Fourth, generalizability may be limited in this study because we collected data from only three medical facilities in western Alabama.

In conclusion, this study used ML/DL approaches to understand better alcohol and drug use problems among patients seen by primary care providers participating in a federally funded SBIRT program. We concluded that the GAIN method for data imputation coupled with mixed effects prediction models are best suited for predicting AUDIT and DAST scores. Future research should consider fine-tuning the ML/DL model developed in this study, especially with more complete data, and developing and testing the utility of a clinical tool translated from the ML/DL algorithm.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] T.F. Babor, F. Del Boca, J.W. Bray, Screening, brief intervention and referral to treatment: implications of SAMHSA's SBIRT initiative for substance abuse policy and practice, Addiction 112 (2017) 110–117.

[2] J.W. Bray, F.K. Del Boca, B.G. McRee, S.W. Hayashi, T.F. Babor, Screening, Brief Intervention and Referral to Treatment (SBIRT): rationale, program overview and cross-site evaluation, Addiction 112 (2017) 3–11.

[3] D.-H. Han, S. Lee, D.-C. Seo, Using machine learning to predict opioid misuse among US adolescents, Prev. Med. 130 (2020) 105886.

[4] M. Afshar, A. Phillips, N. Karnik, J. Mueller, D. To, R. Gonzalez, R. Price, R. Cooper, C. Joyce, D. Dligach, Natural language processing and machine learning to identify alcohol misuse from the electronic health record in trauma patients: development and internal validation, J. Am. Med. Inform. Assoc. 26 (3) (2019) 254–261.

[5] M. Afshar, B. Sharma, S. Bhalla, H.M. Thompson, D. Dligach, R.A. Boley, E. Kishen, A. Simmons, K. Perticone, N.S. Karnik, External validation of an opioid misuse machine learning classifier in hospitalized adult patients, Addict. Sci. Clin. Pract. 16 (1) (2021) 1–11.

[6] W.S. John, H. Zhu, P. Mannelli, R.P. Schwartz, G.A. Subramaniam, L.-T. Wu, Prevalence, patterns, and correlates of multiple substance use disorders among adult primary care patients, Drug Alcohol Depend. 187 (2018) 79–87.

[7] K.K. Mak, K. Lee, C. Park, Applications of machine learning in addiction studies: A systematic review, Psychiatry Res. 275 (2019) 53–60.

[8] B.A. Smothers, H.T. Yahr, Alcohol use disorder and illicit drug use in admissions to general hospitals in the United States, Am. J. Addict. 14 (3) (2005) 256–267.

[9] M.A. Prince, B.T. Conner, S.R. Davis, R.C. Swaim, L.R. Stanley, Risk and protective factors of current opioid use among youth living on or near American Indian reservations: An application of machine learning, Transl. Issues Psychol. Sci. (2021).

[10] J.A. Sidey-Gibbons, C.J. Sidey-Gibbons, Machine learning in medicine: a practical introduction, BMC Med. Res. Methodol. 19 (1) (2019) 1–18.

[11] E.J. Ha, J.H. Baek, Applications of machine learning and deep learning to thyroid imaging: where do we stand? Ultrasonography 40 (1) (2021) 23.

[12] Q. Bi, K.E. Goodman, J. Kaminsky, J. Lessler, What is machine learning? A primer for the epidemiologist, Am. J. Epidemiol. 188 (12) (2019) 2222–2239.

[13] A. Karatzoglou, D. Meyer, K. Hornik, Support vector machines in R, J. Stat. Softw. 15 (1) (2006) 1–28.

[14] S. Lalmuanawma, J. Hussain, L. Chhakchhuak, Applications of machine learning and artificial intelligence for Covid-19 (SARS-CoV-2) pandemic: A review, Chaos Solitons Fractals 139 (2020) 110059.

[15] V.K.R. Chimmula, L. Zhang, Time series forecasting of COVID-19 transmission in Canada using lstm networks, Chaos Solitons Fractals 135 (2020) 109864.

[16] T. Chakraborty, I. Ghosh, Real-time forecasts and risk assessment of novel coronavirus (COVID-19) cases: A data-driven analysis, Chaos Solitons Fractals 135 (2020) 109850.

[17] S.C. Lemon, J. Roy, M.A. Clark, P.D. Friedmann, W. Rakowski, Classification and regression tree analysis in public health: methodological review and comparison with logistic regression, Ann. Behav. Med. 26 (3) (2003) 172–181.

[18] I.H. Witten, E. Frank, Data mining: practical machine learning tools and techniques with java implementations, ACM Sigmod Rec. 31 (1) (2002) 76–77.

[19] D.L. Albright, L. Holmes, M. Lawson, J. McDaniel, K. Laha-Walsh, S. McIntosh, Veteran-nonveteran differences in alcohol and drug misuse by tobacco use status in Alabama SBIRT, Journal of Social Work Practice in the Addictions 20 (1) (2020) 46–58.

[20] C. Dunn, L. Deroo, F.P. Rivara, The use of brief interventions adapted from motivational interviewing across behavioral domains: a systematic review, Addiction 96 (12) (2001) 1725–1742.

[21] SAMHSA, Substance abuse and mental health services administration. Integrated change therapy: Brief treatment for adults with substance use and co-occurring mental health disorders, 2021, Retrieved from: https://health.uconn.edu/sbirtinstitute/wp-content/uploads/sites/101/2018/03/SAMHSA-Brief-Treatment-Guide.pdf. (Last accessed 1 Jan 2022).

[22] J.B. Saunders, O.G. Aasland, T.F. Babor, J.R. De la Fuente, M. Grant, Development of the alcohol use disorders identification test (AUDIT): WHO collaborative project on early detection of persons with harmful alcohol consumption-II, Addiction 88 (6) (1993) 791–804.

[23] E. Yudko, O. Lozhkina, A. Fouts, A comprehensive review of the psychometric properties of the Drug Abuse Screening Test, J. Subst. Abuse Treat. 32 (2) (2007) 189–198.

[24] S.A. Maisto, M.P. Carey, K.B. Carey, C.M. Gordon, J.R. Gleason, Use of the AUDIT and the DAST-10 to identify alcohol and drug use disorders among adults with a severe and persistent mental illness, Psychol. Assess. 12 (2) (2000) 186.

[25] Z.B. Sahri, U.T. Malaysia, et al., Support vector machine-based fault diagnosis of power transformer using k nearest-neighbor imputed DGA dataset, J. Comput. Commun. 2 (09) (2014) 22.

[26] M.J. Azur, E.A. Stuart, C. Frangakis, P.J. Leaf, Multiple imputation by chained equations: what is it and how does it work? Int. J. Methods Psychiatric Res. 20 (1) (2011) 40–49.

[27] H.-m. Lu, G. Perrone, J. Unpingco, Multiple imputation with denoising autoencoder using metamorphic truth and imputation feedback, 2020, arXiv preprint arXiv:2002.08338.

[28] F. Biessmann, T. Rukat, P. Schmidt, P. Naidu, S. Schelter, A. Taptunov, D. Lange, D. Salinas, Datawig: Missing value imputation for tables, J. Mach. Learn. Res. 20 (2019) 1–6.

[29] J. Yoon, J. Jordon, M. Schaar, Gain: Missing data imputation using generative adversarial nets, in: International Conference on Machine Learning, PMLR, 2018, pp. 5689–5698.

[30] R.J. Little, A test of missing completely at random for multivariate data with missing values, J. Amer. Statist. Assoc. 83 (404) (1988) 1198–1202.

[31] E.-L. Silva-Ramírez, R. Pino-Mejías, M. López-Coello, M.-D. Cubiles-de-la Vega, Missing value imputation on missing completely at random data using multilayer perceptrons, Neural Netw. 24 (1) (2011) 121–129.

[32] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. Artificial Intelligence Res. 16 (2002) 321–357.

[33] E.C. Gök, M.O. Olgun, SMOTE-NC and gradient boosting imputation based random forest classifier for predicting severity level of covid-19 patients with blood samples, Neural Comput. Appl. (2021) 1–15.

[34] J. Starkweather, A.K. Moske, Multinomial logistic regression, 2011.

[35] I. Steinwart, A. Christmann, Support Vector Machines, Springer Science & Business Media, 2008.

[36] J.R. Quinlan, Induction of decision trees, Mach. Learn. 1 (1) (1986) 81–106.

[37] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.

[38] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, Adv. Neural Inf. Process. Syst. 30 (2017) 3146–3154.

[39] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F.E. Alsaadi, A survey of deep neural network architectures and their applications, Neurocomputing 234 (2017) 11–26.

[40] A.L. Caterini, A. Doucet, D. Sejdinovic, Hamiltonian variational auto-encoder, 2018, arXiv preprint arXiv:1805.11328.

[41] M. Zhang, Z. Cui, M. Neumann, Y. Chen, An end-to-end deep learning architecture for graph classification, in: Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[42] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.

[43] C.J. Petrucci, A primer for social worker researchers on how to conduct a multinomial logistic regression, J. Soc. Serv. Res. 35 (2) (2009) 193–205.

[44] T. Yamashita, K. Yamashita, R. Kamimura, A stepwise AIC method for variable selection in linear regression, Comm. Statist. Theory Methods 36 (13) (2007) 2395–2403.

[45] H. An, L. Gu, Fast stepwise procedures of selection of variables by using AIC and BIC criteria, Acta Math. Appl. Sin. 5 (1) (1989) 60–67.

[46] L.B. Sheiner, T.H. Grasela, An introduction to mixed effect modeling: concepts, definitions, and justification, J. Pharmacokinet. Biopharm. 19 (3) (1991) S11–S24.

[47] D. Hedeker, A mixed-effects multinomial logistic regression model, Stat. Med. 22 (9) (2003) 1433–1446.

[48] A. Hajjem, F. Bellavance, D. Larocque, Mixed-effects random forest for clustered data, J. Stat. Comput. Simul. 84 (6) (2014) 1313–1328.

[49] D. Chicco, G. Jurman, The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation, BMC Genomics 21 (1) (2020) 1–13.

[50] N. Wongpakaran, T. Wongpakaran, D. Wedding, K.L. Gwet, A comparison of Cohen's Kappa and Gwet's AC1 when calculating inter-rater reliability coefficients: a study conducted with personality disorder samples, BMC Med. Res. Methodol. 13 (1) (2013) 1–7.

[51] D.A. Newcombe, R.E. Humeniuk, R. Ali, Validation of the world health organization alcohol, smoking and substance involvement screening test (ASSIST): report of results from the Australian site, Drug Alcohol Rev. 24 (3) (2005) 217–226.

[52] A. Mottini, R. Acuna-Agost, Relative label encoding for the prediction of airline passenger nationality, in: 2016 IEEE 16th International Conference on Data Mining Workshops, ICDMW, IEEE, 2016, pp. 671–676.

[53] N. Molinari, Free knot splines for supervised classification, Journal of classification 24 (2) (2007) 221–234.

[54] Y. Bengio, Y. Grandvalet, No unbiased estimator of the variance of k-fold cross-validation, J. Mach. Learn. Res. 5 (Sep) (2004) 1089–1105.

[55] J. Gareth, W. Daniela, H. Trevor, T. Robert, An introduction to statistical learning: with applications in R, Spinger, 2013.

[56] C. Data61, Stellargraph machine learning library, GitHub Repository (2018) https://github.com/stellargraph/stellargraph. (Last accessed 1 Jan 2022).

[57] J.T. Schwartz, M. Gao, E.A. Geng, K.S. Mody, C.M. Mikhail, S.K. Cho, Applications of machine learning using electronic medical records in spine surgery, Neurospine 16 (4) (2019) 643.

[58] D.J. Park, M.W. Park, H. Lee, Y.-J. Kim, Y. Kim, Y.H. Park, Development of machine learning model for diagnostic disease prediction based on laboratory tests, Sci. Rep. 11 (1) (2021) 1–11.

[59] A. Mechelli, S. Vieira, From models to tools: clinical translation of machine learning studies in psychosis, Npj Schizophrenia 6 (1) (2020) 1–3.