Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2014) 22: 690 – 707

doi:10.3906/elk-1208-60

An urgent precaution system to detect students at risk of substance abuse through classification algorithms

Faruk BULUT[1], **İhsan Ömür BUCAK**[2,*]
[1]Department of Computer Engineering, Faculty of Engineering, Fatih University, İstanbul, Turkey
[2]Department of Electrical and Electronics Engineering, Faculty of Engineering and Architecture, Melikşah University, Kayseri, Turkey

Received: 17.08.2012 • **Accepted:** 19.09.2012 • **Published Online:** 21.03.2014 • **Printed:** 18.04.2014

Abstract: In recent years, the use of addictive drugs and substances has turned out to be a challenging social problem worldwide. The illicit use of these types of drugs and substances appears to be increasing among elementary and high school students. After becoming addicted to drugs, life becomes unbearable and gets even worse for their users. Scientific studies show that it becomes extremely difficult for an individual to break this habit after being a user. Hence, preventing teenagers from addiction becomes an important issue. This study focuses on an urgent precaution system that helps families and educators prevent teenagers from developing this type of addiction. The aim of this study is to detect a teenager's probability of being a drug abuser using classification algorithms in machine learning and data mining. The objective is not to test the classifiers theoretically on the benchmark datasets, but rather to use this study as a basis for advanced and detailed studies in this field in the future. This paper not only uses a special dataset but also focuses on psychometrics and statistics. The findings of this study show that if there is a computed high risk for a teenager, some precautions, if necessary, may be taken by educators and parents to keep the teenager away from those substances.

Key words: Substance abuse, risk assessment, data mining, machine learning, classification algorithms

1. Introduction

Substance abuse among youth is a very common problem worldwide. There is a gradual increase in the use of illicit drugs among elementary and high school students. Children as young as the age of 10 are becoming drug abusers. Preventing teenagers from becoming addicted to drugs is an important issue. Hence, some precautions have to be taken to prevent teenagers from abusing drugs.

The purpose of this study is to detect a risk rate for teenagers who are at high risk. It does not focus on addicts. If the risk rate is higher than normal, it is strongly recommended that families and educators take some precautions for their children to stay away from drugs. Hence, we call this project the "Urgent Precaution System" for those teenagers who are at high risk. With the aid of this study, it will be possible to take some precautions to keep them drug-free.

The flow chart in Figure 1 shows how the system works. As is seen, after administering the questionnaire, some extra information, such as grade point average (GPA), smoking habit, alcoholism, and misbehavior, is to be obtained from educators and families. These data are collected and provided to the classifier algorithms in order to find out the risk rate for each teenager in terms of percentage.

*Correspondence: iobucak@meliksah.edu.tr

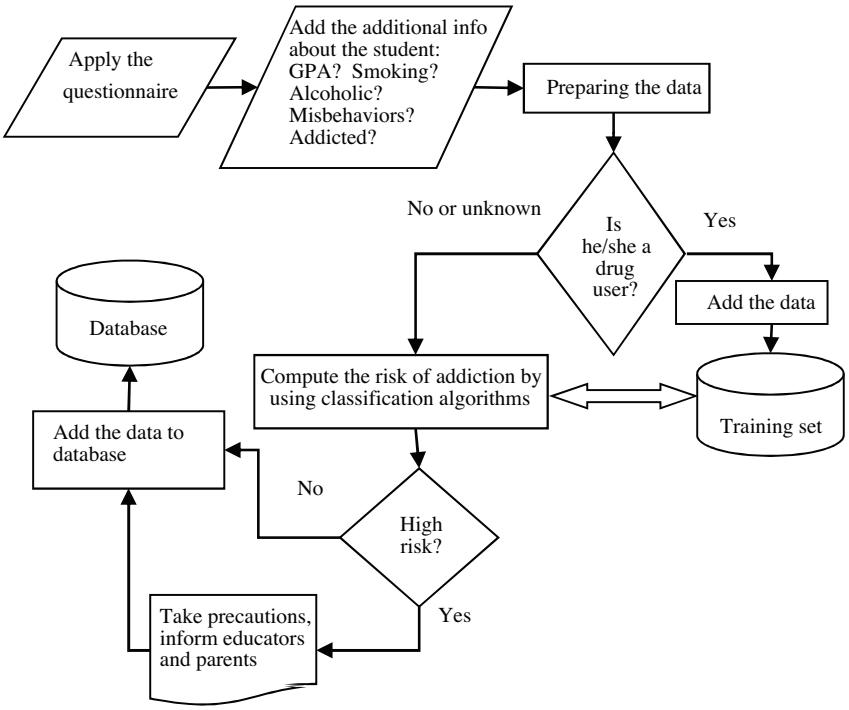

Figure 1. A flowchart of the system.

2. Questionnaire validation procedure

In order to find the risk value percentage for each student, the reasons and factors leading to addiction should be identified. A questionnaire, which aims to find out the reasons and factors of addiction, ought to be utilized so as to gather relevant data to conduct this study.

There are many harmful effects of drug abuse, including changes in the user's brain, body, and spirit. This work does not deal with the results, but rather deals with the reasons for being dependent. There are some reasons leading adolescents to addiction. The first is about family-related issues, such as dissatisfaction with family relationships, antisocial family members, stress in the family, poverty or welfare usage in the family, illiterate parents, divorced parents, loss of one or both parents, and lack of people who could be a positive role model for the adolescent [1]. The second is physical/sexual abuse or violence. The third is related to some genetic factors, birth-related problems, and physical or psychological problems [2]. The fourth is related to social problems. Social variables can be peer pressure, cultural effects, acceptance of substance use in society, low socioeconomic status, and unemployment [3,4].

In general, adolescents experiment with alcohol or cigarettes before heroin or other hard substances [5]. These possible risk factors in adolescence are the predictors of substance use.

3. Dataset preparation procedure

In order to find the risk value percentage for each teenager and gather relevant data to conduct a study, the causes and factors leading to addiction were collected in a questionnaire comprising 25 questions that was presented for students at schools and in-patient facilities (i.e. drug abusers) to fill out [6]. The answers in the questionnaire are seen as the attributes of the records in the database. The questions are derived from scientific articles, theses, and books in the related field [7–13]. Each of the questions tries to determine an effect leading

the teenager to be a drug abuser. For example, smoking cigarettes or using tobacco products, being a member of a problematic family, and being a friend of drug abusers might have higher effects on the individual in using those substances. The first 20 questions are asked directly to the teenagers and the last 5 are asked directly to their educators because the students try to avoid answering those private and individual questions and never want to reveal their bad habits. The 25 questions in the questionnaire, which were checked and examined by psychological, counseling, and guidance experts before being used, are listed in Table 1.

Questions about abuse symptoms are not asked in the questionnaire since the objective is not to figure out whether the person is presently a drug abuser or not. Therefore, these answers do not show the leading reasons, but rather the results of addiction. With official permission obtained from the Ministry of National Education, the questionnaire forms were distributed to 671 students to fill out. In addition, with official permission obtained from the Ministry of Health, the same questionnaire forms were distributed to 35 Research, Treatment, and Training Center for Alcohol and Substance Addiction (AMATEM) inpatients. AMATEM is a health center for drug-abusing youth in the hospital. Therefore, in our database, there are a total of 706 records.

All of the classifiers used in this study produce outputs for each record in the database. The output indicates the risk rate for an individual under consideration. There are 5 risk classes as outputs, from Classes 1 through 5. The risk rate in Class 1 is between 0% and 20%. Since it is the lowest risk section, there is no need to take any precautions. Students in this section are not at risk at all. The risk rate in Class 2 is between 20% and 40%. Students in this class are not potential drug abusers. However, it does not mean that there is no risk at all. The risk rate in Class 3 is between 40% and 60%. Students in this section tend to be potential future drug abusers. It is better to inform the students about the harmful effects of drugs and take some precautions. The risk rate in Class 4 is between 60% and 80%. There is a very big risk for the students in this section. It is compulsory to take urgent precautions. The risk rate in Class 5 is between 80% and 100%. Since the risk rate is the highest in this section, some precautions must be taken immediately. Educators and families have to be very careful about their students. If he/she is a drug abuser, it is compulsory to consult a clinic immediately.

All of the approaches to performing classification require a predefined training dataset, which is used to develop the specific parameters required by the approach and consists of sample input and output data, as well as the classification assignment for the data [14]. The training dataset was diligently prepared using the suggestions and propositions of the experts and staff of the Psychology Department at Fatih University in İstanbul, Turkey, and it was well-defined before applying the algorithms. In this study, there are 110 tuples in the predefined dataset, of which 37 are absolutely 100% drug abusers; 35 of the 37 are from the AMATEM clinic and the remaining 2 are from the high schools. A total of 63 students were chosen by the educators, where 43 of these students were at a risk rate ranging from 10% to 80% and 20 of them were at the risk rate of 0%. The former and latter groups of students were put into these risk intervals on the basis of reasoning for addiction.

4. Classification algorithms

In this study, 5 types of classification algorithms are applied to categorize the incoming records. These algorithms are k-nearest neighbor (k-NN) as a distance-based algorithm, the naïve Bayes classifier as a statistical based algorithm, ID3 and C4.5 as decision tree-based algorithms, naïve Bayes/decision tree (NBTree) as a hybrid approach, and, finally, one-attribute-rule (OneR) and projective adaptive resonance theory (PART) as rule-based algorithms. All 5 types of classification methods are implemented in Weka software and the C programming language in order to get results, discuss, and criticize. The definitions of some notations used in these methods are as follows:

Table 1. The questionnaire for elementary and high school students.

1. Your age?
2. Your sex? (a) Male. (b) Female.
3. Are your parents alive? (a) Yes, both. (b) Only my mother. (c) Only my father. (d) No, neither.
4. Whom are you staying with or where are you staying at? (a) My mother (parents are separated). (b) My father (parents are separated). (c) Both parents. (d) Another relative of mine. (e) An orphanage. (f) A dormitory.
5. Monthly income ($USD) of your family? (a) Less than $600. (b) Between $600 and $1750. (c) Between $1750 and $3500. (d) More than $3500.
6. At what frequency do you visit or meet with your relatives (aunt, uncle, grandfather, grandmother, cousins, etc.)? (a) Once a week. (b) Once a month. (c) Once a year. (d) No visits.
7. Do you play an instrument? (a) Yes. (b) No.
8. Do you exercise regularly? (a) Yes. (b) No.

Table 1. Continued.

9. At what frequency do you read books?
   (a) Once a week.
   (b) Once a month.
   (c) Once a year.
   (d) No reading at all.

10. What kind of music do you like?
   (a) Rock – heavy metal.
   (b) Rap.
   (c) Pop.
   (d) Folk songs.
   (e) Do not like music.

11. At what frequency do you go to movies?
   (a) Once a week.
   (b) Once a month.
   (c) Once a year.
   (d) Almost never.

12. How many hours a day do you spend on the Internet?
   (a) Less than 1 hour, sometimes none.
   (b) 1–2 hours.
   (c) At least 3 hours.

13. In the case of an offer by the friend you like most of something that you know is harmful, what would you do?
   (a) I would accept one time.
   (b) I would certainly reject it.
   (c) Not sure.

14. Do your parents take care of you when you run into a problem?
   (a) Only my mother does.
   (b) Only my father does.
   (c) Both do.
   (d) They do not intervene much.

15. How do you think your life in the future will be compared to today?
   (a) Worse.
   (b) The same.
   (c) Better.
   (d) No idea.

Table 1. Continued.

16. Who is the person you feel is the closest to you that you can talk to in case of a problem you run into?
    (a) My mother and father.
    (b) My brother and/or sister.
    (c) My relative.
    (d) My teacher.
    (e) My friend.
    (f) I do not feel close to anyone.

17. With whom do you spend most of your leisure time?
    (a) Someone from my family.
    (b) One of my relatives.
    (c) My friends.
    (d) My computer.
    (e) No one.

18. When you face a situation that is really tough and you must overcome it,
    (a) I do not make any attempts to overcome if I know it is hard
    (b) I make an attempt. If it does not work, then I leave it alone.
    (c) I do my best if I need to overcome something, no matter how hard it is.

19. Do you feel that you lack confidence in yourself?
    (a) Yes, I have no confidence in myself.
    (b) No, I have confidence in myself.
    (c) I do not know.

20. Who do you like most out of your friends at your school? Please write down their names including their last names.
    (a) . . . . . .
    (b) . . . . . .
    (c) . . . . . .
    (d) . . . . . .
    (e) . . . . . .

21. Grade point average of the student for the previous semester according to a 5-point grading system?

22. Whether or not the student has been given any disciplinary punishment so far?

23. Whether or not the student is an alcoholic?

24. Whether or not the student smokes cigarettes or uses tobacco products?

25. Whether or not the student is a substance/drug addict?

$D = \{t_1, t_2, t_3, \ldots t_n\}$ and $T = \{t_1, t_2, t_3, \ldots t_n\}$ represent the database and tuples, respectively. For example, t_i is the ith tuple in the database. C is the set of classes including the 5 types, $C = \{$*Class 1*, *Class 2*, *Class 3*, *Class 4*, *Class 5*$\}$, and is defined in Section 3. $f : D \rightarrow C$ is the mapping function or the classifier. Each tuple in the database is assigned to exactly one class by the classifier algorithms, such that $C_j = \{t_i | f(t_i) = C_j, 1 \leq I \leq N$, and $t_i \in D\}$.

4.1. k-NN algorithm

The k-NN algorithm is a common method for classifying new tuples based on the closest training examples in the feature space [15]. A k-NN classifier determines the class label of a record by looking at the labels of its k nearest neighbors in the training dataset and puts the record into the class that most of its neighbors belong to [16]. The neighbors are taken from a set of objects for which the correct classification is known. The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and the class labels of the training samples.

k is a positive integer, typically small. In the classification phase, k is a user-defined constant and an unlabeled vector (a query or test point). It is classified by assigning the label that is most frequent among the k training samples nearest to that query point. Usually, Euclidian distance is used as a distance metric. Determining the k value in this method usually seems to be problematic for users [17,18]. The accuracy of the k-NN method can be degraded by the presence of irrelevant features or by inconsistent feature scales. Much research effort has been put into selecting or scaling features to improve classification [17]. In this study, 5 kinds of k-NN algorithms are applied: simple k-NN, absolute distance k-NN, Euclidian distance k-NN, distance-weighted k-NN, and, finally, distance- and attribute-weighted k-NN. These algorithms are listed below from the least to the most efficient.

4.1.1. Simple k-NN

This type of approach is the simplest and easiest among the existing k-NN algorithms. Only the average of the closest tuples in the training set to a new record is sufficient. For instance, x as the new entry to be placed into a class and y in the training set are considered as 2 tuples composed of N features or attributes, such that:

$$\begin{aligned} x &= \{x_1, x_2, x_3, \ldots x_N\}, \\ y &= \{y_1, y_2, y_3, \ldots y_N\}. \end{aligned} \tag{1}$$

In order to implement the simple k-NN, it is sufficient to check whether there is a difference between the values of each corresponding attribute of these 2 tuples. If there is a difference between these 2 values of the corresponding attributes, it is treated as 1. If the values of the attributes are the same, it is treated as 0.

4.1.2. Absolute distance k-NN

This version is obviously better than the simple k-NN approach. We can compute the absolute distances between 2 tuples using the absolute distance function $d(x, y)$. x as the new tuple to be put into a class and y as one of the tuples in the training dataset are composed of N features (attributes), such that:

$$\begin{aligned} x &= \{x_1, x_2, x_3, \ldots x_N\}, \\ y &= \{y_1, y_2, y_3, \ldots y_N\}, \end{aligned} \tag{2}$$

where N is 25. The distance function based on an absolute distance measuring is given by:

$$d_A(x,y) = \sum_{i=1}^{N} |x_i - y_i|. \tag{3}$$

4.1.3. Euclidian distance k-NN

Euclidian distance is used to measure the real distance in N-dimensional space between 2 tuples as follows:

$$d_E(x,y) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \tag{4}$$

where N is the number of features or attributes in the tuple and is equal to 25 in this study. d_E is the Euclidian distance between tuples x and y. Euclidian distance measuring produces better results when it is compared to the other measuring techniques mentioned above.

4.1.4. Distance-weighted k-NN

One obvious refinement to the k-NN algorithm is to weigh the contribution of each k neighbor according to their inversely squared distance to the query point x_{new}, which gives a greater weight to closer neighbors [19]:

$$f(x_{new}) = \arg\max v \in V \sum_{i=1}^{N} w_i \delta(v, f(x_i)), \tag{5}$$

where

$$w_i = \frac{1}{d^2} = \frac{1}{d(x_{new}, x_i)^2}. \tag{6}$$

Here, d is the distance between tuples x_i and x_{new}. N is the number of tuples in the training set. f is the function used to find the class of the new entry.

To accommodate the case where the query point x_{new} exactly matches one of the training instances x_i, which causes the denominator $d(x_{new}, x_i)^2$ to become zero, $f(x_{new})$ is assigned to $f(x_i)$. If there are several such training examples, we assign the majority class among them. We can distance-weigh the instances for real-valued target functions in a similar way, replacing the final line of the algorithm as follows [19]:

$$f(x_{new}) = \frac{\sum_{i=1}^{N} w_i f(x_i)}{\sum_{i=1}^{N} w_i}. \tag{7}$$

There is no need to try to detect the appropriate value of k, which is usually difficult in this approach. The k value can usually be chosen as smaller than half of the N value. Moreover, it is observed during the experiments that the larger k values do not produce any feasible solutions. In this experiment, the k value is omitted and all of the tuples in the dataset are contaminated to the calculation. Usually, contaminating some tuples that are far from the new record produces inefficient results. As a result, each of the N tuples in the k area has a contribution in the calculations. In other words, each record in the training set has either a low or high effect on

the calculations. In particular, the tuples closer to the new record according to their distances produce higher effects, whereas the tuples farther away produce lower effects in the calculations. Hence, this seems to be the best of all of the examined methods so far.

In this approach, each tuple whose distance is less than or equal to k has an effect on the new record, which makes it better. It is robust and quite effective when there is a large training dataset.

4.1.5. Distance- and attribute-weighted k-NN

This k-NN version is the most advanced and efficient one of them all. In the previous versions of k-NN methods, one problem was the equal effect of all of the attributes in calculating the distance between the new tuple and the available tuples in the training set. Some of the attributes of a tuple should be less important to the classification and some of them should be more important. However, this might mislead the classification process and decrease the accuracy of the classification algorithm. A major approach to deal with this problem is to weigh each of the attributes differently when calculating the distance between 2 records. In this approach, a combined method is used to improve the accuracy of k-NN [20]. For example, smoking cigarettes, being a heavy drinker, and being a member of a problematic family are the main factors leading the youth to addiction. Therefore, these attributes are more effective than others for classifying the upcoming records of the attribute weights in Table 2. Eq. (8) figures out the distance, *dist*, between tuples x and y:

$$dist(x,y) = \sqrt{\sum_{i=1}^{N} c_i * (x_i - y_i)^2}, \tag{8}$$

where $x = \{x_1,\ x_2,\ x_3,\ \ldots x_N\}$ is the new entry to be placed into class $y = \{y_1,\ y_2,\ y_3,\ldots y_N\}$ in the training set, and x and y are composed of N features (attributes). $C = \{c_1,\ c_2,\ c_3,\ \ldots,c_N\}$ identifies the weights (cost) of the N attributes of each record in the training set.

Table 2. Ranked attributes.

0.4635	Discipline
0.4337	Tobacco
0.3379	Parental care
0.3346	Internet
0.2957	Whom are you staying with?
0.2904	Book
0.2768	Sport
0.2524	Self-confidence
0.2498	Leisure time
0.2396	Alcohol
0.2022	Music
0.1826	Age
0.1749	Need to overcome
0.1317	Visiting relatives
0.118	Friend offering
0.1135	Parent
0.095	Instrument
0.0897	Income (level)
0.0724	Movies
0.0721	Problems sharing
0	Life in the future

In the previous versions of k-NN, all of the weights of the N attributes were equal to one another ($c_1 = c_2 = c_3 = \ldots. = c_N$). However, in this version, some attributes produce higher effects on the others and some produce lower. The Euclidian distance and attribute weighted k-NN equation to find the effect of y on x is given as follows:

$$f(x) = \arg\max v \in V \sum_{i=1}^{N} w_i \delta(v, f(y_i)), \tag{9}$$

where

$$w_i = \frac{1}{dist^2(x_i, y_i)} = \frac{1}{\left(\sqrt{\sum_{i=1}^{N} c_i * (x_i - y_i)^2}\right)^2}. \tag{10}$$

Here, the f function is used to find the class of the new entry, which is the same as in Eq. (7) [19]:

$$f(y_i) = \frac{\sum_{i=1}^{N} w_i f(x_i)}{\sum_{i=1}^{N} w_i}. \tag{11}$$

We use the following distance equation along with Eq. (10) for the distance- and attribute-weighted k-NN algorithm:

$$dist_w(x, y) = \sum_{i=1}^{N} w_i (x_i - y_i)^2. \tag{12}$$

There may be 2 options to determine the weights: in the first one, the weights are provided by experts, and in the second, the machine learning approach is used. In this study, the former option is preferred.

The main steps of the distance- and attribute-weighted k-NN algorithm are given in a pseudocode, as follows:

1. Let $\boldsymbol{D}$ be the set of all points in the training set,
2. Let $\boldsymbol{C} = \{C_1, C_2, C_3, \ldots, Cn\}$ be the set of classes,
3. Let $\boldsymbol{f}$ be the function to find the class for a tuple,
4. Let $\boldsymbol{c} = \{c_1, c_2, c_3, \ldots, c_n\}$ be the weights (costs) of each of the attributes,
5. Given a query tuple $\boldsymbol{x}$ to be classified **do**
6. **compute** the Euclidian distances between $\boldsymbol{x}$ and $\boldsymbol{D}$
7. **sort** the computed distances
8. **select** the k number of nearest points
9. for each k-nearest point **do**
10. let $\boldsymbol{w}$ be the distance weights of each point
11. **compute** the weight $\boldsymbol{w}$ according to all attributes $\boldsymbol{c}$

12. **compute** the function f
13. **return** C_i

In this section, we considered a new classification version of k-NN in order to increase the classification accuracy. In this algorithm, each of the attributes is given a different weight in the calculation. Each attribute that is more weighted has more effect on the distance between 2 records and each attribute that is less weighted has less effect on the distance. The practical test shows that the suggested approach is more accurate than the previous approaches presented above. The results of the other approaches are compared with the results of this combinatory approach. This version of k-NN seems to be the best of all. It yields feasible solutions.

4.1.6. Attribute weights

The weights of the attributes in the training dataset are detected with the aid of their information gains. Their information gain rates are found using Weka [21]. The most effective attributes as the leading factors of addiction for youth are shown at the top of the list in Table 2. The less effective ones are at the bottom. Hence, the attributes at the top have higher weights.

In our experiments, 5 of the less effective attributes are not contaminated into the risk calculations in order to test the accuracy and performance of the system. At the end, we see that the computational performance has increased and the accuracy of the system has changed slightly. It is understood that there is no need to contaminate some attributes that have a low information gain value.

4.2. Naïve Bayes classifier

The naïve Bayes classifier is another classification approach known as Bayesian classification, which is based on Bayes' rule of conditional probability. In this method, each of the attributes has a part in the calculation [22]. The naïve Bayes classifier is particularly vigorous in its irrelevant attributes and classification allows for evidence from many attributes to make the final decision [23].

4.3. Decision tree algorithm

A decision tree is used to classify new instances by sorting them down the tree from the root to some leaf nodes. In other words, a decision tree is a tree in which each branch node represents a choice between a number of alternatives, and each leaf node at the bottom represents a predefined class as a final decision. Decision trees are commonly used for classifying new tuples according to a predefined training dataset. Moreover, they are used for gaining information for the purpose of decision making. A decision tree starts with a root node. From this node, each node is split recursively according to the answer of the question in the current node. The final result is a decision tree in which each branch represents a possible scenario of a decision and its outcome. When reaching a leaf node at the end, this will be the class of the final decision for the new tuple [24].

In this section, the C4.5 and ID3 algorithms are implemented in order to construct a decision tree for classifying the upcoming tuples [25,26].

4.4. A naïve Bayes/decision tree hybrid algorithm

Naïve Bayes classifiers exhibit their weakness when their requirement of making strong independence assumptions is violated. In this case, the desired accuracy is obtained at an early stage and will not improve much as the size of the database increases [23].

On the downside of decision tree classifiers, as each split is made, the data are split based on the test and at 2 dozen levels, there are usually very few data left to make decisions [23].

NBTree proposes a hybrid approach to take advantage of both the decision tree (i.e. segmentation) and naïve Bayes (evidence accumulation from multiple attributes). NBTree is particularly used to analyze the large volume of data, of which intrusion detection is one popular area [27]. In this study, a decision tree is developed through univariate (single attribute) splits at each node like regular decision trees, but naïve Bayes classifiers are employed at the leaves [23].

4.5. Rule-based algorithms

OneR (or shortly 1R) is a simple and attribute-based classification algorithm proposed by Holte in [28]. Only one attribute (that is why it is called 1R) is assumed sufficient to classify the new incoming records. The basic idea in this approach is to find the best attribute to perform the classification with the lowest errors as based on the training data. The 'best' is defined by counting the number of errors for each attribute. In other words, the one with the least total error of all of the values will be the 'best' attribute for the OneR algorithm. Therefore, OneR selects the rule with the lowest error rate. In this study, another rule-based algorithm, which is known as PART, is also applied. PART is a partial decision tree algorithm, which is the developed form of the C4.5 algorithm. The chief strength of PART over C4.5 is that it does not necessitate performing global optimization to produce the proper rules [29].

4.6. Graph theoretical approach

Peer pressure plays a great role in leading a person to addiction [2–4]. In the questionnaires, the students are asked to list their best friends for the purpose of knowing the friendships among them. To illustrate that, the relations among students are shown as an example in a graph in Figure 2. It is an undirected and unweighted graph. In this graph, each node with a number in it represents the ID number of a record in the database. The edge between 2 nodes represents the adjacency; in other words, it indicates the friendship between these 2 friends.

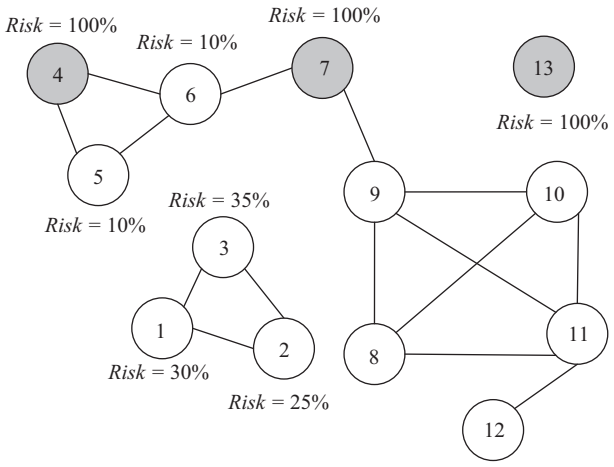

Figure 2. Graph representation of friendships.

As seen Figure 2, the 4th, 7th, and 13th nodes are drug abusers. Although the 13th individual is absolutely a drug abuser, he/she has no bad effect on the others because he/she has no relationship with the

other students. The 1st, 2nd, and 3rd students have a group of friends. Their risks are 30%, 25%, and 30%, respectively. None of them are drug abusers. Thus, there is not a big risk for these students. The 5th, 6th, and 9th students are at high risk because they have friendships with the addicted students. The 8th, 9th, 10th, and 11th students have an adjacency among them. Only the 9th student is at risk because of the 7th student. The 9th student has a higher risk of addiction than the others.

The relationships among the students represented in Figure 2 are put into a 2-dimensional array. In order to keep the graph with its nodes and edges, 2-dimensional arrays are used to store the nodes and adjacencies to implement the breadth-first search algorithm [30].

5. Experimental results

The C programming language is used in order to implement the classifier algorithms, and the Weka program is used in order to check the results and measure the accuracies of the algorithms. Different versions of k-NN algorithms, particularly the last 2 versions, distance-weighted k-NN and attribute- and distance-weighted k-NN, are implemented in C. Programming in C also provides us with flexibility and scalability. In [6], the C and Weka outputs were provided on pages 61–70 and the C codes were given in part on pages 78–83, and a CD of the entire code was uploaded to tez2.yok.gov.tr as a compressed RAR file.

In this study, 7 different classification algorithms are implemented. Interpretations and comparisons about these algorithms over the output files are made below. Figure 3 compares the accuracy of these implemented algorithms according to the training set.

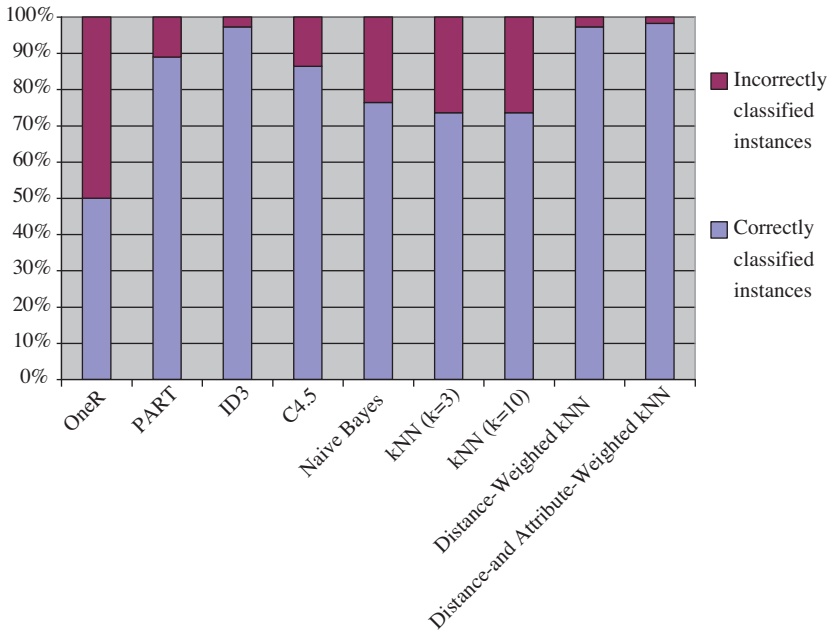

Figure 3. Accuracy of the implemented algorithms.

Two of the most commonly used algorithms, ID3 and distance-weighted k-NN, end up classifying the data with the same 97.3% accuracy rate, as shown in Figure 3.

The naïve Bayes classifier can be used for both binary and multiclass problems. It is easy to implement, fast in running, and offers highly scalable model building and scoring. However, if the training set is not rich enough, some of the attribute probabilities in the training set could be zero. This is a big barrier in

the calculations. In this study, the naïve Bayes classifier produces 22.8% incorrectly classified outputs in the experiments. In other words, 22.8% of the students are put into the wrong class. The naïve Bayes classifier cannot produce satisfactory results in the implementations since it holds all of the attributes equally. However, some attributes end up producing higher effects than others. For example, the *smoking* attribute brings about a higher effect than that of the *going to the cinema occasionally* attribute. In addition, this method requires a well-prepared training dataset. All of the probabilities and predictions in this method depend on the training set. It requires more than 2000 tuples in the training set in order to get satisfactory and feasible results.

k-NN is the most flexible and customizable algorithm for our needs. In particular, the distance- and attribute-weighted k-NN produces the best results of all. This method can cause us to weigh some particular attributes and customize the calculations in C. Moreover, it can also produce instances within the entire range from 0% to 100%. For instance, it is expected to see outputs such as 14.56%, 47.59%, and 70.44%, whereas other methods can only output the class names, such as Classes 1 through 5. The k-NN performances for various k values are shown in Table 3 and Figure 4.

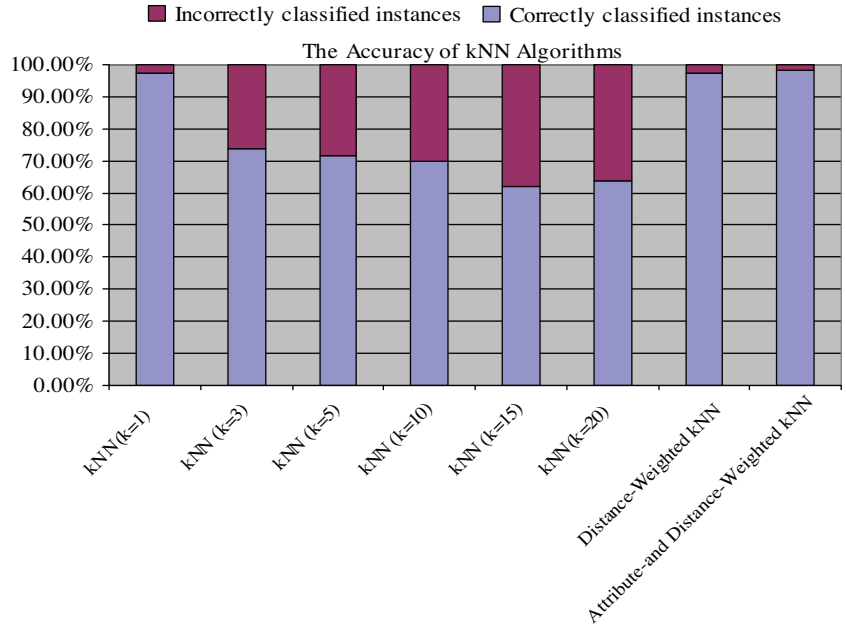

Figure 4. Comparing the accuracies of the k-NN algorithms.

As the k value gets higher, the accuracy goes down according to the results, as shown in Figure 4. The best solution is obtained by the distance- and attribute-weighted k-NN algorithm.

The ID3 decision tree and likewise the naïve Bayes classifier depend on the training data. First, the decision tree is built by the training set. Next, the decision tree is applied to each tuple in the database. However, building the decision tree requires a lot of calculations. The accuracy of the decision tree is not as good as that of others, even though classifying the new tuples is done very easily.

NBTree significantly improves on both C4.5 and naïve Bayes. Although it surpasses both C4.5 and naïve Bayes, it runs longer than C4.5 and naïve Bayes alone. The number of nodes built with NBTree is in many cases significantly smaller than that of C4.5.

Table 3. Comparing the accuracies of the k-NN algorithms.

	Correctly classified instances (%)	Incorrectly classified instances (%)
k-NN (k = 1)	97.3	2.7
k-NN (k = 3)	73.6	26.4
k-NN (k = 5)	71.8	28.2
k-NN (k = 10)	70.0	30.0
k-NN (k = 15)	61.8	38.2
k-NN (k = 20)	63.6	36.4
Distance-weighted k-NN	97.3	2.7
Distance- and attribute-weighted k-NN	98.2	1.8

Generating a rule based solely on a single attribute among many may not be seen as sufficient for this study. As is known, there are many factors leading an individual to addiction. Therefore, the OneR and PART algorithms cannot be accepted as applicable to this study, as both depend only upon a single attribute. In the experiments, OneR gives us just 2 possible classes, Classes 1 and 5, by checking the *Discipline* attribute, and ignores Classes 2, 3, and 4.

All of the tuples in the database within the 5 classes and the percentages of the classes calculated for each algorithm are shown in Table 4.

Table 4. The percentages of the classes for each algorithm.

	Class 1 (%)	Class 2 (%)	Class 3 (%)	Class 4 (%)	Class 5 (%)
C4.5	66.0	11.4	9.8	3.2	9.5
ID3	66.2	8.9	8.8	3.4	12.8
Naïve Bayes classifier	56.8	14.8	16.2	4.2	8.2
NBTree	82.6	6.7	2.4	3.5	4.9
OneR	94.8	0.0	0.0	0.0	5.2
PART	69.5	6.8	10.2	6.0	7.5
k-NN (k = 5)	82.6	2.8	2.5	7.7	5.2
Distance-weighted k-NN (k = 5)	76.3	4.6	5.7	5.8	7.5
Distance- and attribute-weighted k-NN (k = 5)	73.4	16.3	8.5	0.8	1.1

In Table 5, all of the records in the dataset are put into the classes against different versions of the k-NN algorithms and are shown in terms of their percentages. The more reliable version of the k-NN algorithms seems to be attribute- and distance-weighted k-NN.

Table 5. Output percentages of the k-NN algorithms.

	Class 1 (%)	Class 2 (%)	Class 3 (%)	Class 4 (%)	Class 5 (%)
Attribute- and distance-weighted k-NN	73.4	16.3	8.5	0.8	1.1
Distance-weighted k-NN	76.3	4.6	5.7	5.8	7.5
k-NN (k = 1)	65.7	11.5	9.2	4.5	9.1
k-NN (k = 3)	69.5	3.1	4.2	5.7	3.7
k-NN (k = 5)	82.6	2.8	2.5	7.7	5.2
k-NN (k = 10)	86.2	1.1	1.8	7.7	3.2
k-NN (k = 15)	86.3	0.2	1.8	8.0	3.7

6. Conclusion

As is widely accepted, digitalizing the whole psychological status of a human being is very difficult. Converting someone's emotional status, habits, attitudes, and other specifications to numeric values seems nearly impossible. In this study, we tried to overcome this hardship and observed that some advanced scientific studies could also be done on this subject. In order to get more feasible and efficient results out of these kinds of similar studies, there are some suggestions proposed below about, for example, the scientific committee, the applied questionnaire, the training set, and the employed algorithms.

First, there should be a scientific committee strictly studying this project. In the committee, there should be some psychologists, sociologists, psychiatrists, academicians, educators, computer engineers, IT programmers, and some relevant specialists. In the application of the distance- and attribute-weighted k-NN method, this committee can assign different weights to each attribute in the training set. Only this kind of a committee will have the opportunity to find feasible and reliable solutions.

The questions should preferably be selected by the committee. Although 20 or 25 questions in the questionnaire are seen as sufficient in the initial steps, in further levels, there should be at least 50 questions aimed at deciphering the complete nature of a human being for a more strict and serious application.

If the training set is larger than the one used in this project, we could have better outputs. In this study, the number of tuples in the training set is 110. There should be at least 2000 different entries in the predefined training set. Each of the entries should be different from the others in the set and assigned a different risk value. The training set should contain a lot of different entries in order to protect the system from bias among the output classes. All of the probabilities of the attributes in that phase should have a value that is different from zero. If the conditional probability used in the naïve Bayes classifier is zero, classifying the new records becomes difficult, sometimes impossible. For this reason, there should be a variety of tuples in the training set. For the training dataset, we prepare 110 tuples with different risk rates, ranging from 0% to 100%. If an individual smokes cigarettes, his/her risk rate cannot be lower than 50%. Additionally, if the individual both smokes and misbehaves, his/her risk rate cannot be lower than 70%.

On the other hand, there is another approach in the way of defining the training dataset, which is to put only drug abusers and those who are not drug abusers into the training set. Therefore, the risk values for the tuples can either be 0% or 100%. Hence, there is no need to give different risk rates, from 0% and 100%, to the tuples in the training set. By applying the classifiers to some particular new records, some different risk rates can be calculated. These particular records, whose risk values range from 0% to 100%, can then be added to the training set in order to enrich it.

As is seen in the experiments, the attribute-weighted k-NN and distance- and attribute-weighted k-NN methods yield the best results of all. Aside from these applied methods, more advanced and effective classifier algorithms can also be used in this study. In particular, in the distance- and attribute-weighted k-NN method, different weights can be assigned to each question (attribute) in the training set by the scientific committee. By applying the other classifying techniques, similar results can also be obtained. Therefore, it is possible to compare these algorithms for the purpose of determining the most efficient one. The weaknesses and strengths of the implemented algorithms can also be compared at the same time.

This study puts emphasis on psychometrics and statistics, as well as the use of its special dataset in the classification algorithms. The objective is not to test the classifiers theoretically on the benchmark datasets, but rather to use this study as a basis for more advanced and detailed studies in this field in the future.

This study deals with an interesting and beneficial study of youth drug dependency. It focuses on a common worldwide problem in the new generation using a new and original method. Moreover, it tries to find

a feasible solution as an urgent precaution system to those youth problems. In conclusion, it can be said that this study could be a blueprint for further advanced steps.

As future work, some ensembles like bagging, boosting, random subspaces, and rotation forest algorithms, as collective classifiers, can be implemented on this specific dataset in order to increase the accuracy of the output.

References

[1] K. Ögel, Characteristics of Adolescent Volatile-substance Abusers - UMATEM Data, İstanbul, Bakırköy Ruh ve Sinir Hastalıkları Hastanesi, 2004.

[2] S. Kırcan, The Relationship Between Peer Pressure, Internal versus External Locus of Control and Adolescent Substance Use, MSc, Boğaziçi University, İstanbul, 2006.

[3] D.C. Kimmel, I.B. Weiner, Adolescence: A Developmental Transition, 2nd ed., New York, Wiley, 1995.

[4] M. Windle, R.C. Windle, Alcohol and other substance use and abuse, In: G.R. Adams, M.D. Berzonsky, M. Windle, R.C. Windle, editors, Blackwell Handbook of Adolescence, Malden, MA, USA, Blackwell Publishing, 2003.

[5] O.G. Bukstein, Adolescent Substance Abuse: Assessment, Prevention and Treatment, New York, Wiley, 1995.

[6] F. Bulut, Detecting Students at Risk of Substance Abuse by Using Data Mining Classification Algorithms, MSc, Fatih University Graduate School of Technical Sciences, İstanbul, 2010.

[7] B. Gülkan, Personality and Socio-demographic Traits of Heroin Addicts, MSc, İstanbul University, İstanbul, 1994.

[8] I. Seyman, Dimensions of the Narcotics Issue in Turkey, MSc, Ankara University, Ankara, 2000.

[9] C. Zor, Views of Student Families for Secondary Education about the Risks of Drug Use and the Ways of Protection, MSc, Ankara University, Ankara, 2005.

[10] G. Erdem, C.Y. Eke, K. Ögel, S. Tanver, "Peer characteristics and substance use among high school students", Journal of Dependence, Vol. 7, pp. 111–116, 2006.

[11] C. Aydın, "A socio-demographic evaluation of cases applying to a child and adolescent dependency centre during a period of two years attending", Journal of Dependence, Vol. 7, pp. 31–37, 2006.

[12] M.S. Can, Substance Dependence Habits Observed in the Second Grades of Primary Students, MSc, Sakarya University, Sakarya, Turkey, 2007.

[13] L. Tuncer, An Assay on the Role and Importance of Domestic Safety and National Ethics Factors in Struggle against Substance Addiction from Republic Era to Date, MSc, Fırat University, Elazığ, Turkey, 2007.

[14] M.H. Dunham, Data Mining: Introductory and Advanced Topics, 2nd ed., Upper Saddle River, NJ, USA, Prentice Hall, 2005.

[15] L. Jiang, H. Zhang, Z. Cai, Dynamic K-Nearest-Neighbor Naive Bayes with Attribute Weighted, Berlin, Springer, 2006.

[16] K. Chen, L. Liu, A survey of multiplicative perturbation for privacy-preserving data mining, in: C.C. Aggarwal, P.S. Yu, editors, Privacy-Preserving Data Mining: Models and Algorithms, New York, Springer, pp. 157–181, 2008.

[17] F. Nigsch, A. Bender, B. Buuren, J. Tissen, E. Nigsch, J.B.O. Mitchell, "Melting point prediction employing k-nearest neighbor algorithms and genetic parameter optimization", Journal of Chemical Information and Modeling, Vol. 46, pp. 2412–2422, 2006.

[18] P. Hall, B.U. Park, R.J. Samworth, "Choice of neighbor order in nearest-neighbor classification", Annals of Statistics, Vol. 36, pp. 2135–2152, 2008.

[19] T.M. Mitchell, Machine Learning, New York, McGraw-Hill, 1997.

[20] M. Moradian, A. Barani, "KNNBA: k-nearest-neighbor-based association algorithm", Journal of Theoretical and Applied Information Technology, Vol. 6, pp. 123–130, 2009.

[21] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, P.I. Witten, "The WEKA data mining software: an update", ACM SIGKDD Explorations Newsletter, Vol. 11, pp. 10–18, 2009.

[22] T. Hill, P. Lewicki, Naïve Bayes Classifier Introductory Overview, STATISTICS Methods and Applications, StatSoft, Tulsa, OK, USA, 2007.

[23] R. Kohavi, "Scaling up the accuracy of naïve-Bayes classifiers: a decision-tree hybrid", Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Vol. 7, pp. 202–207, 1996.

[24] W. Peng, J. Chen, H. Zhou, An Implementation of ID3 - Decision Tree Learning Algorithm, Sydney, Australia, University of New South Wales, 2010.

[25] J.R. Quinlan, "Induction of decision trees", Machine Learning, Vol. 1, pp. 81–106, 1986.

[26] J.R. Quinlan, C4.5: Programs for Machine Learning, San Mateo, CA, USA, Morgan Kaufmann Publishing, 1993.

[27] D.M. Farid, H. Nouria, M.Z. Rahman, "Combining naïve Bayes and decision tree for adaptive intrusion detection", International Journal of Network Security & Its Applications, Vol. 2, pp. 12–25, 2010.

[28] R.C. Holte, "Very simple classification rules perform well on most commonly used datasets", Machine Learning, Vol. 11, pp. 63–91, 1993.

[29] E. Frank, I.H. Witten, "Generating accurate rule sets without global optimisation", Proceedings of the 15th International Conference on Machine Learning, 1988.

[30] S. Russel, P. Norvig, Artificial Intelligence: A Modern Approach, 2nd ed., Upper Saddle River, NJ, USA, Prentice Hall, 2003.