ISSN: 1309 – 6575
Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi
Journal of Measurement and Evaluation in Education and Psychology
2019; 10(1);80-95

An Analysis Program Used in Data Mining: WEKA

Gökhan AKSU* Nuri DOĞAN**

Abstract
In this study, it is aimed to introduce one of the data mining methods which is very popular in recent years and commonly used in this area. For this purpose, the WEKA program and the decision trees, which is one of the methods used to estimate the dependent variable through independent variables, will be introduced. In today's age of technology, the amount of information at hand is constantly increasing and the derivation of meaningful results from this information is seen as a valuable field of study. Data mining aims to reveal the information that is hidden in a large amount of data after a series of operations, which is very useful for researchers. Regarding this approach that is mostly based on estimation and classification, there is a lot of new and unvalidated software that has not yet been fully tested. In this study, we discuss WEKA software, which is one of the programs in the field of data mining, how to run the program and the content of the analyzes and output files. The study also contains some suggestions for the practitioners who want to use this program about the superior aspects of the software and what kind of analysis can be done with it.

Key Words: Data mining, WEKA, Classification, Prediction, Algorithm

INTRODUCTION

Thanks to the Internet, a major revolution has been occurred in accessing and using information over the last decade (Jain, 2015). At this stage, researchers and scientists focused on storing, recalling and using data when needed. From time to time, the data at hand is likened to a gold mine for conducting research and development in a particular area. Data mining is a process that defines the data obtained as input and output information (Weiss and Davison, 2010). Fayyad, Piatetsky-Shapiro, and Smyth (1996), one of the most cited researchers in the field, describe data mining as the execution of certain algorithms to elicit certain patterns from the available data.

Simple structures of different types are formed on the data set, in a way that they can be easily displayed. For example, in a data set, there may be examples where one property works very well, whereas the others are unrelated or unnecessary. In one data set, there may be examples where the properties contribute independently and equally to the output variable. In another data set, there may be a number of properties that have a simple logical structure and that are obtained by the decision tree. In a different data set, there may be several independent sets of rules that assign examples to different classes. In another data set, interdependencies between different subsets of properties can be examined. In a data set, linear dependence between weighted sums of numerical properties determined by appropriate weighting methods can be examined, whereas in another it is possible to place these examples on specific regions of the sample space based on the distances between the examples. In a different set of data, there may be no class value as seen in the algorithms where learning is uncontrolled (Witten and Frank, 2005). There are different examples where different structures and different data mining tools are used in the infinite equality of possible data sets. In these examples, there are also algorithms that may completely overlook the appropriateness of different types and make correct classifications for only one class (Holte, 1993).

Classification rules are alternatives to decision trees. The given rules are first executed for the first line, and then for the second and the last line respectively. The lines setting the rules are generally

*Dr., Adnan Menderes University, Aydın Vocational School, Aydın-Turkey, e-mail: gokhanaksu1983@hotmail.com ORCID ID: 0000-0003-2563-6112
** Prof. Dr., Hacettepe University, Education Faculty, Ankara-Turkey, e-mail: nuridogan2004@gmail.com, ORCID ID: 0000-0001-6274-2016

To cite this article:
Aksu, G., & Doğan, N. (2010). An analysis program used in data mining: WEKA. *Journal of Measurement and Evaluation in Education and Psychology, 10*(1), 80-95. DOI: 10.21031/epod.399832

Received: 28.02.2018
Accepted: 29.11.2018

named as "Decision List". These lists show the rules required for the correct classification of the examples in the decision table (Chadha and Singh, 2012).

The rules of association are not different from the classification rules except that they are not limited with classification, they can also estimate any property. In this approach in which the patterns, relationships and causal structures in the data set are revealed, different combinations of properties can be estimated as well. In addition, the association rules do not require the use of a set of sets of rules, as in the rules of classification (Han, Kamber and Pei, 2000). A large number of different association rules can be derived from a small data set. For this reason, applying it to a large number of examples is not needed. The rules of association are generally formed by "if…. then" cycle. For example, "if there is no wind and the game is not played, the humidity is high" association rule indicates that the humidity should be high if the game was not played when there is no wind (Witten and Frank, 2005)

Clustering rules take the form of a diagram showing how the examples in the data set fall into specified clusters. The results obtained in this method, where the clusters' belonging is determined, are sometimes given as a dendrogram and sometimes as a table. The cluster, to which each example in the data set belongs, is shown according to its characteristics regarding the similarity or dissimilarity criteria with the other examples. On the dendrogram, each cluster is depicted on a sub-level, as decomposed into its subsets (Karypis, Han, and Kumar, 1999).

In the prediction procedure, numerical properties are considered instead of categorical variables. In data mining, even though categorical variables are considered in decision trees or association rules, linear regression models are used to estimate the numerical value of the property (Padmavathi, 2012). Apart from classical regression methods, a more accurate and consistent estimation can be made by logistic regression method. In this approach, where input variables are weighted according to their contribution to the model, a large number of logistic equations are obtained instead of a single regression equation (Perlich and Provost, 2002).

Decision Trees

The problem of creating a decision tree is expressed self-repetitively. First of all, a property is determined to be set for the root node (a node without ancestor node and therefore the node at the top) and a branch is created for each possible value (Fayyad and Irani, 1992). This process breaks the data set into subsets for each property value. The process is repeated successively for each branch by using only the examples that reach this branch. If, in any case, all examples in a node belong to the same class, then the development of that branch of the tree (node) is stopped. Because from this point on there will be no decomposition into different classes (Quinlan, 1993). The only thing left to make a decision is to determine the way of dividing each property when a series of different classes is given. Below are the results of the game played for the general view and temperature characteristics of the weather. There are 2 possible (Yes-No) alternative for each branch and they are divided into classes at the top, as shown in Figure 1.

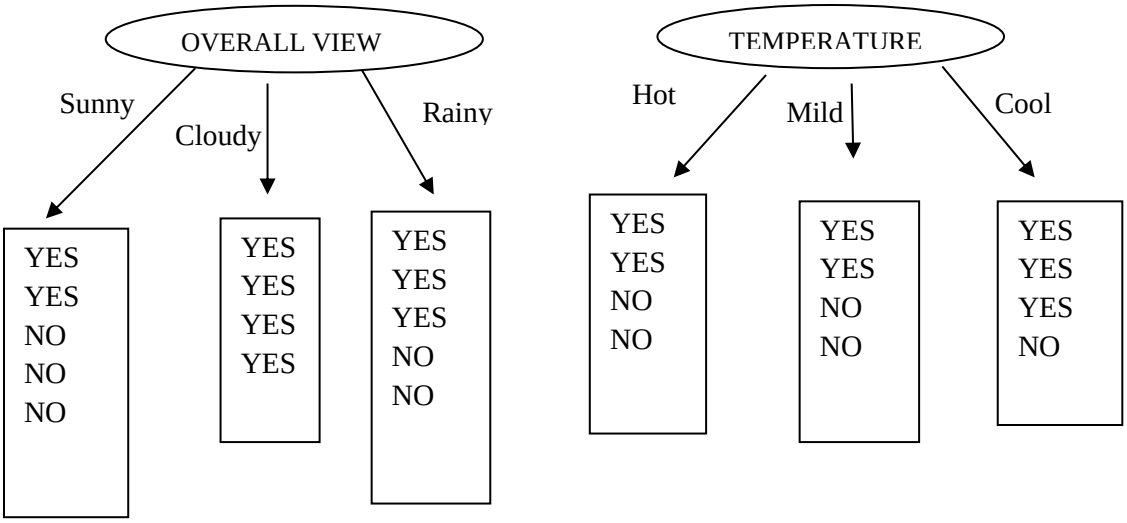

Figure 1. Tree Structure for Weather Data (Witten and Frank, 2005)

We can decide which of the branches are the best choices for the tree structure given in the figure regarding the class of the leaves shown in the rectangular shape. The numbers of yes and no classes are shown on the leaves. If only a single class is created on the sheet (Yes or No), there will be no need to divide again and the iterative branching process will end.

We want this process to be as short as possible, thus we examine small trees. If we measured the purity of each node, we would have to choose the properties that produced the purest child nodes. Now take some time and consider which feature will be the best.

The measure of purity that we will use is called information and it is measured by units designated as bits. The bit represents the amount of information required for a new Yes (played) or No (not played) classification related to a node of the tree. Of course, there is nothing special about these numbers and there is a similar relationship between them, regardless of their actual value. Thus, we can add another criterion to the list. The information obtained must be in accordance with the multi-stage property shown previously. Remarkably, it appears that there is only one function that meets all these characteristics, and this function is called the amount of information or entropy (Rokach and Maimon, 2008). The following equation shows how the amount of information is obtained mathematically.

Entropy (p_1, p_2, …, p_n) = - p_1.logp_1 – p_2.logp_2, … , - p_n.logp_n

The negative signs (-) here are due to the rules of the logarithm coming from the fractions p_1, p_2, … , p_n, while calculating the logarithm of the fractions in the form of A / B, the base remains unchanged and a minus sign is placed before the expression in the denominator ($\log_a {}^{A}/_{B} = \log_a A - \log_a B$). In spite of the minus sign in the expression, in reality, the amount of information will be positive.

Usually, logarithms are given at the base of 2 ($\log_2 x$) and in this case the amount of information called entropy is defined as bit. These bits are a generic type of bits used in the computer (1-0).

p_1,p_2,…,p_n expressions of the entropy indicates fractions that sum up to 1. For example;

info ([2,3,4]) = entropy (2/9, 3/9, 4/9)

As shown in the above formula, the sum of 2/9, 3/9 and 4/9 fractions specified as p_1, p_2, p_3 will be equal to 1. Therefore, decisions regarding multi-stage characteristics can be obtained by the general formula given below.

entropy (p,q,r) = entropy (p,q+r) + (q+r). entropy ($\frac{q}{q+r}$, $\frac{r}{q+r}$)

It should be kept in mind that p+q+r = 1 in the given formula.

info ([2,3,4]) = $-\frac{2}{9}$ x log $\left(\frac{2}{9}\right)$ - $\frac{3}{9}$ x log $\left(\frac{3}{9}\right)$ - $\frac{4}{9}$ x log $\left(\frac{4}{9}\right)$ = $-\frac{2}{9}$ x (log2 – log9) - $\frac{3}{9}$ x (log3 – log9) - $\frac{4}{9}$ x (log4 – log9)

$$= \frac{2\log2+2\log9}{9} - \frac{3\log3+3\log9}{9} - \frac{4\log4+4\log9}{9}$$

$$= \frac{-2\log2-3\log3-4\log4-9\log9}{9}$$

The above formula is a formula that shows how the amount of information is calculated in practice.

Reliability: Evaluation of What We Learn

Evaluation plays a key role in achieving real progress in data mining. There are many different ways to extract patterns from the data file and to make deductions from these patterns. However, while determining which method should be used for a particular problem, we should have systematic ways of evaluating and comparing different approaches in which different methods are used. The evaluation process is not as simple as it seems.

To see which of the two different methods works better, training and test sets are needed. But in fact, learning the performance in the training set will not be a good indicator of performance in an independent test set. We will need ways to estimate performance limits based on the experimental studies to be conducted on the data sets (Breiman and Friedman, 1984). More clearly, training the learning method through a data set and then test this method over the same data set would not be an approach that reflects reality. Therefore, the error rate on the training set may not be a good indicator of actual performance. Because, since the classifier learned from the same training data, any calculation process based on this data will be optimistic (Witten & Frank, 2005).

Our concern in data mining will be the future performance of the method based on the new data rather than past performance based on the historical data. The class of each example in the training set is already known. Even though we are not interested in the classification, and our aim is to clear the data rather than making a prediction, the classifications should also be taken care of. To estimate the performance of a classifier on the new data, we need to evaluate the error rate on a different data set than the data set from which the classifier is obtained. In the studies, we should assume that both training data and test data are representative examples of the problem that we are investigating (Sumathi & Sivanandam, 2006).

In some cases, the test data may differ from the training data in nature. For example, suppose we have examples of credit risk problems. Let us assume, for example, that the bank has received training data from branch offices in New York and Florida, and wants to learn how well the classifier to be obtained through this training data will perform at a new branch in Mexico. It would probably be appropriate to use New York data as the test data to evaluate Florida classifier and to use Florida data to evaluate New York classifier. However, if the data sets were combined in the process of training, the performance in the test data would probably not be a good indicator of the performance of data obtained from a different state in the future (Witten, Frank and Hall, 2016). In the studies conducted in the field of education, it has been investigated whether a different learning method is effective on the courses that students attend school. In a study by Lopez et al. (2012), it was investigated whether the forum usage data in the Moodle learning management system is a significant indicator of the success of the course. In the study, it was tested whether the same result can be achieved with clustering algorithms in cases where the class variable (course achievement) was not known.

In most cases, the training data should be classified by hand and, of course, the test data should also be classified likewise in order to obtain the error rate. This limits the amount of data that can be used for training, verification, and testing, and how to achieve the best performance with this limited data becomes a question. The answer is, a certain amount (20% - 30%) of the data set is kept for testing, which is called the retention procedure, and then the remaining amount is used for training. However,

if necessary, a part of the data to be used for training can also be separated as the validation data (Mitchell, 1997).

Cross-validation

What will you do if you have limited amounts of data for training and testing? In the retention method, a portion of the available data is used for testing and the rest is used for training. However, a part of the data allocated to training can be used for verification. In practice, allocating one-third of the available data for testing and the remaining two-thirds for education is a very common method (66% = Education, 34% = Test data).

Of course, the sample group that you use for training (or test) may not be a good representative of the universe. In general, you cannot directly understand whether a sample is a good representative of the universe or not. But there is a very simple control that can be significant for you. In this approach, each class of your universe, in which all the data is included, should be represented in the training and test data with an accurate ratio. Only this will increase the representation power of the universe. However, if you are unlucky and you have lost data in all examples of a class in your sample, you cannot expect the classifier, which will be obtained through this data, to be able to perform well on the test data. In this case, the results will be worsened as the data in the test set is over-represented, since none of these data is included in the training set. Instead, you must ensure that random sampling was carried out to ensure that each class in the universe is properly represented in the training and test data. This method is called the stratification or stratified retention approach. Although it is often noteworthy to apply this method, this is seen only as primitive protection to eliminate an uneven or unbalanced representation in the training and test data (Witten, Frank, & Hall, 2016).

A more generic way of reducing the bias arising from a particular sample selection is repeating the whole process several times for training and test data across different sample groups. In each repetition process, a certain proportion of the data (for example two-thirds for the training) can be used for training (if possible, in a stratified and randomly selected way), whereas the rest can be used for testing. A more general error rate can be obtained by taking the average of error rates from different repetitions. This is called the repetitive retention method for calculating the error rate.

In cases where only one retention method is applied, you may think that the training and test data exchange roles. In other words, you can train the system with the test data while testing the system with the training data. This way, you reduce the problem of unequal representation in training and test data by averaging the two results you get.

Unfortunately, this only makes sense for 50% - 50% splitting of training and test data, which is not generally considered as a suitable split. Instead, using more than half of the available data for training is a better approach. On the other hand, there is an important statistical technique called cross-validation. In cross-validation, we set a constant folding or splitting number for the data, for instance, let's set this constant number as 3. The data is then divided into three equal parts and each of them is used for testing while the remaining portion is used for training. That is to say, two-thirds of the available data are used for training while one-third of the data at hand is used for testing. When this process is repeated, each example will be used once for testing. This process is called the triple cross-validation and if stratification is carried out, it is called the stratified triple cross-validation (Kohavi, 1995).

When there is a single and constant data available, a 10-fold cross-validation approach is applied to the standard method used to estimate the error rate of the learning technique. In this approach, the universe of the study, which includes all of the data, is randomly divided into 10 classes and each class must be represented in the whole data set at approximately the same rate. 10-fold cross-validation process is illustrated in Figure 2 in order to make it easier to understand.

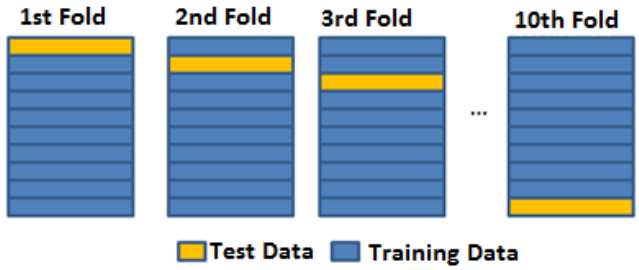

Figure 2. Ten-fold cross-validation

In this process, each fold is held separately for the test, while the learning process is performed through the remaining nine folds and the error rate is calculated from the 1/10 data held separately. Thus, the learning method is repeated for a total of 10 times over different training data (each with many common points). Finally, a general error rate is obtained by averaging 10 different error rates (Refaeilzadeh, Tang and Liu, 2007).

Bootstrap Method

This method is based on statistical procedures that can be defined as repeated sampling. A sample is taken from the current data set for training or testing beforehand and these examples are excluded. That is, once an example is selected, it cannot be selected again. This approach is similar to building teams for a football game. Just as you cannot choose a player for two different teams, in the Bootstrap method the same example cannot be selected twice. However, the examples in the data set are not like humans. Most learning methods use the same example twice, and the results will differ if it takes place twice in the training set. Mathematically speaking, in case of having more than one image of the same object, it is not meaningful to mention the groups defined as "cluster" (Ibarguren et al., 2014).

The logic of the Bootstrap method is to be able to sample the data set at hand by relocation in order to be able to create a training set. This stage will show the mysterious Bootstrap number (0.632) and how it is obtained. For this, a data set consisting of n examples is relocated so that another sample consisting of n examples is sampled. Since some of the examples from this second data set will be repeated, the original data set should contain examples used in somewhere else, to be used as test samples

But what is the probability that a particular example is not included in the training set? The probability of being included in the training set is $\frac{1}{n}$ and the probability of not taking part in this group is calculated as $(1 - \frac{1}{n})$. The multiplication of the probabilities of not being in the given cluster is calculated with the help of the equation given below.

$$(1-\frac{1}{n})n \approx e^{-1} = 0{,}368$$

In the equation given above, the symbol shown by e is the base of the natural logarithm and it is equal to 2.7168. However, you should keep in mind that this display is not equal to the error rate. This equation shows the possibility of not being selected in any way for a particular example. This way, the test set will cover approximately 36.80% of the data for a fairly large data set and therefore the training set will cover 63.20% of the data at hand. So, we learned where the number of Bootstrap 0.632, which was previously defined as a mysterious number, is coming from. Some examples are re-used in the training set to reach the total number of n in the original data set (Kushary, 2012).

This numerical value obtained by training a learning system through the training set and the error value obtained from the test set will be a pessimistic estimate of the actual error. Because even though the size of the training set is n, it still covers 63% of the samples and considering that 90% of the data is used in 10-fold cross-validation, it can be seen that this is not a larger proportion. To compensate this, the error obtained from the relocation process through the examples in the training set is combined

with the error rate obtained from the test set. The error value obtained as a result of the relocation should not be used as the error term alone because it will be a very optimistic estimate of the actual error value. However, the Bootstrap method generates the error value in the combination of the errors obtained from the training and test data and the value is shown below is calculated as the final error value (Bradley and Tibshirani, 1993).

$$e = 0{,}632 \times e_{\text{test data}} + 0{,}368 \times e_{\text{training data}}$$

The entire Bootstrap method is then repeated several times by relocating different examples for the training set and the results are averaged.

The Bootstrap method can be considered as the best method for estimating the error rate for small data sets. However, just as in the method of excluding one group, it is possible to show its disadvantage through a completely artificial and special situation. In fact, assume that the data set that we previously considered had been randomly divided into two classes. The actual error rate of any prediction rule is assumed to be 50%. But let's consider the training set; as a result of the relocation, a draft learning method will give 100% success with zero error (e = 0). In this case, the 0.632 Bootstrap method will weigh this with 0.368 and accordingly the overall error term will be calculated as 31,60% (0,632 × 50% + 0,368 × 0%), which will be misleadingly optimistic.

Power of Estimation: Considering the Magnitude of the Error

The evaluation process performed according to the accuracy of the classification implicitly assumes that different errors do not have the same meaning. An example of this can be seen in credit debts. The mistake of giving a loan to someone who did not pay his/her previous loan is much bigger than the mistake made by not giving credit to someone who has never taken a loan.

For the cases where there are two classes in the form of yes-no, borrowing-not borrowing money, giving-not giving a loan, classifying a suspicious part as scrap or not, etc., there are 4 different outcomes, as shown in Table 1. The true positive (TP) and the true negative (TN) are the correct classification results. TP are the cases when the test result is positive when the actual situation is positive, whereas TN are the cases when the test result is negative when the actual situation is negative. False positive (FP) is predicting the output incorrectly as positive (YES) while the result is actually negative (NO). False negative (FN) is predicting the output incorrectly as negative (NO) while the result is actually positive (YES). True positive ratio (TPR) is obtained by dividing positive cases to all cases [TP/(TP+FN)], whereas false positive ratio is obtained by dividing false positives to all negative cases [FP/(FP+TN)] (Powers, 2011).

Table 1. Different Results Regarding a Two-class Estimate

		REAL CASE		
		POSITIVE	NEGATIVE	TOTAL
TEST RESULT	POSITIVE	TRUE POSITIVE (TP)	FALSE POSITIVE (FP)	TP+FP
	NEGATIVE	FALSE NEGATIVE (FN)	TRUE NEGATIVE (TN)	FN+TN
	TOTAL	TP+FN	FP+TN	TP+FP+FN+TN

The overall success rate after computing the ratios of the outcomes is the division of the number of correct classifications to the total number of classifications.

$$\text{Overall Success} = \frac{TP+TN}{TP+TN+FP+FN}$$

The error rate for this classification is calculated by subtracting the overall success rate from 1. Sensitivity and selectivity concepts are also important to know. Sensitivity is the ability of the test to

distinguish positive cases from actual positive cases. At the same time, the true positive rate is defined as sensitivity.

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

Selectivity: It is the ability of the test to distinguish negative cases from actual negative cases.

$$\text{Selectivity} = \frac{TN}{TN+FP}$$

The sensitivity and selectivity values described above define how well the test differentiates non-relevant and relevant cases (Johari, 2016). False positive rate is considered as Type I Error and False negative rate is considered as Type II error.

In a multi-class prediction, the results obtained from the test are shown on a table called an error matrix, having rows and columns for each two-dimensional class. The matrix is defined in a way that each column represents the actual value of an example, whereas each line represents predicted value (test result) of the example (In some cases the rows show the actual value, and the columns the test results). Good results are obtained when the numbers on the main diagonal (the line drawn from the upper left corner to the lower right corner) are quite large and the non-diagonal elements are small (ideally zero).

The curves which are known as ROC (Receiver Operating Characteristic) curves and trying to select test samples with high positive rate attempt to describe the performance of a classifier regardless of the type of error and class distribution. In these graphs, the number of positive examples corresponding to the negative examples is shown in the horizontal axis (x-axis) as a percentage, whereas the number of negative examples corresponding to the positive examples is shown in the vertical axis as a percentage (Sinha & May, 2005). The Kappa statistic or Kappa value is a numerical value in which the expected and observed values are compared. Besides, it is less misleading because it takes the chance factor into account. Expected and observed accuracy calculation is used simultaneously in Kappa statistics and it can be easily determined through the confusion matrix. In this method, the values in the rows show the actual values, while the values in the columns show the estimated values; the expected accuracy rate is subtracted from the actual accuracy rate and this value is divided to (1-expected accuracy rate). Although there is no definitive standard interpretation of the Kappa statistic, in general, the values in the range of 0.00-0.20 are considered as low, 0.21-0.40 as notable; 0.41-0.60 as mediocre; 0.61-0.80 as important and 0.81-1.00 as excellent (Landis and Koch, 1977). There are different error criteria in the evaluation of numerical estimation in data mining. Mean square error, root mean squared error, mean absolute error, squared relative error, root relative squared error and relative error are the error criteria used to determine how accurate numerical estimation is (Witten and Frank, 2005). Relative error values attempt to compensate the fundamental predictability or unpredictability of the output variable, whereas square and square root criteria of the errors allow reducing the errors to the same size. The F-criterion used in data mining is obtained from certainty and recall values. Precision, which is of great importance particularly in the medicine and medical field, shows the success in a condition predicted as true (positive). Recall is more important in marketing and marketing research and shows how successfully positive situations are predicted (Schwenke and Schering, 2007)

DESCRIPTION OF THE PROGRAM, ANALYSIS, EXAMPLES OF ESTIMATION AND CLASSIFICATION

Explorer, which is the main interface of the WEKA program allows easy access to all operations by providing menu selection and form filling options. As shown in Figure 3, the window on the WEKA main screen displays the different mining tasks that the WEKA program supports under six different tabs. Knowledge Flow is a feature of WEKA that allows multiple uses of the features across a screen. Even though only one operation can be performed on the Explorer screen, Knowledge Flow is a feature that allows the tasks to run repeatedly over a number of different operations. Under the Explorer

screen, everything is automatic and ready. In Knowledge Flow, these processes should be created by the user (Şeker, 2016).

Figure 3. WEKA Main Screen

Imagine that you have some data and you want to obtain a decision tree from this data. First, you need to prepare your data, then run the explorer and upload the data to the program. Then choose a decision tree creation method, create a tree, and interpret the output you get. It is easy to repeat this process with a different decision tree algorithm and a different evaluation method. Under the Explorer menu, you can make back and forth transitions between the outputs you get, you can evaluate the models built on different data sets and you can graphically visualize both the models and the datasets including classification errors made by the models.

Below and in Figure 4 & 5, six different tabs at the top of the explorer window are briefly defined. Each of these tabs shows different actions that you can perform with the data at hand.

1. PREPROCESS: allows you to select the data set and edit it in different ways.
2. CLASSIFY: allows you to train the learning method that will classify or predict and evaluate them
3. CLUSTER: allows you to learn clusters of the data sets
4. ASSOCIATE: allows you to learn the rules of association for your data set and evaluate them
5. SELECT ATTRIBUTES: allows you to select relevant properties in your data set
6. VISUALIZE: allows you to see different two-dimensional graphs of your data set and to determine the interaction between them.

Each option provides access to a range of possibilities. Up to now only preprocess and classification options have been considered superficially. For the researchers who want to conduct further analysis, it is recommended to examine the options of cluster, associative, select attributes and visualization.

Preparing Data for Analysis

Data is usually presented in the form of tables or databases. However, the data storage method of the WEKA program contains the aggregated list of examples in ARFF file format, with the data you have entered in the table in ARFF format, in which attribute values specified for each example are shown as separated by commas. Most tables and database programs allow you to convert your data file into .csv (comma-separated value) type. In this file type, the data is stored by inserting a comma between the values in the data file. Once you have done this, it will be sufficient to upload your file to a text

document or to the software. Then save the relations with the "@relation" extension, attribute information with "@attribute" extension, and your data file with "@data" extension as a raw text document. For example, an Excel data file for the PISA data that will be mentioned later is saved differently with a .csv extension and this file is opened from the program and converted to a WEKA data file with .arff extension. However, you don't really need to follow these steps to create the ARFF file yourself, because the explorer can directly read the files with CSV extension.

Introducing Data to the Program

Now, let's upload the data you have at hand to the explorer and try to analyze it. Start the WEKA program to access the screen shown in Figure 4. Then select the explorer, which is one of the five different interfaces under the applications heading.

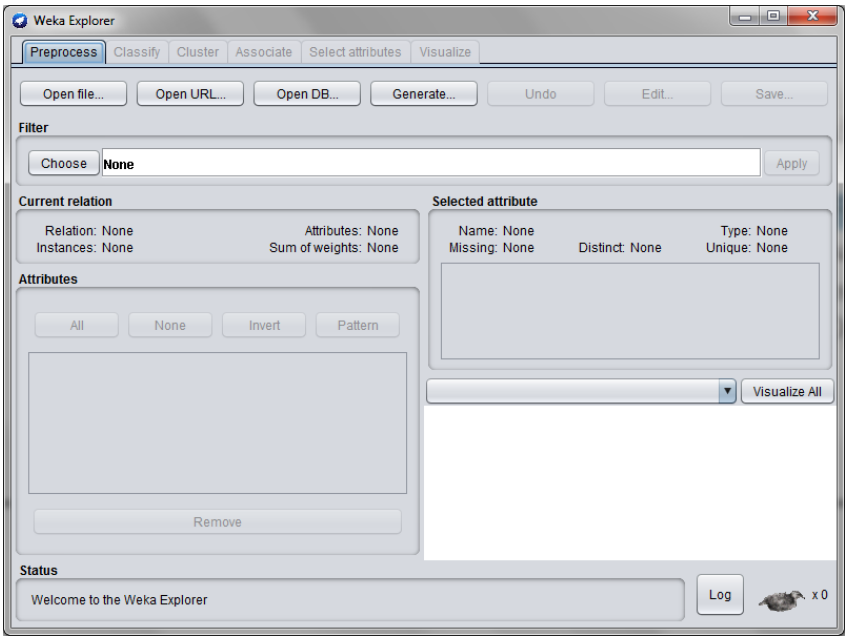

Figure 4. Data Preparation Screen

Then, you can see one of your previous files by clicking the Open File button. By defining your file saved on your computer, you will show your data to the program. If your data is not in ARFF format, which is the basic data file extension of the WEKA program, you will need to select the file type as CSV. When you specify a file with a .csv extension in the program, this file will be automatically converted to a file type with ARFF extension. The screen that you'll face after uploading the file provides you with information about the data set you have.

Performance of Analysis: Creating the Decision Tree

C4.5 decision tree learning method which is one of the most used algorithms in data mining works with J.48 algorithm and this algorithm is a version of the WEKA program that can be used by everyone before the launch of C5.0 application (Kaur and Chhabra, 2014). When you click on the CLASSIFY button shown in Figure 5 (a) and then click on the CHOOSE button from the screen, the screen shown in Figure 5 (b) will appear. Since there is no analysis on the screen, there is no result in the output window in the lower right corner. Once the algorithm and the test type to be used for the classification are set, all you have to do is click the START button.

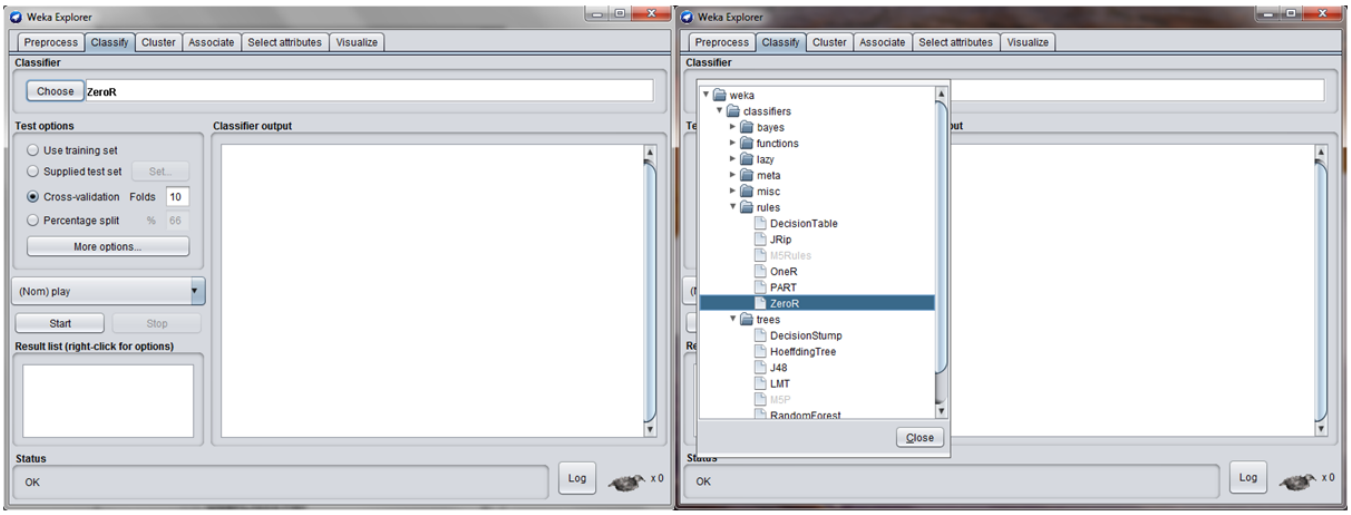

Figure 5. Startup Screen: (a) Choose Window and (b) Classify Window

In order to perform the analysis first, click the choose button located at the top left, then click on Trees section from the hierarchically listed menu and choose J.48 method. At the moment, all you have to do is to choose the method or algorithm that is always at the lowest level from the methods presented sequentially. When you select the algorithm that you will use, you will see the parameters for the method or algorithm you selected in the row next to the choose button. If you double click on this line, J4.8 classifier editor will be opened, the parameters and the numerical values corresponding to the desired values accepted by the program will be displayed. Of course, these default values are determined to make more precise measurements.

After selecting the classifier, you can run the program by clicking the Start button. The WEKA program has a very fast processing capability and it can perform analysis in a short time and while the analysis is in progress a bird will move in the lower right corner of the main screen shown in Figure 5 (a).

As an example, the screenshot obtained after uploading the variables covered in the PISA 2015 student questionnaire, namely the duration of science learning (smins), the total learning time at school (tmins) and the socioeconomic status index of the student (escs) and the input and output variables used in the process of estimating science literacy levels (pv1scie) to WEKA is shown in Figure 6.

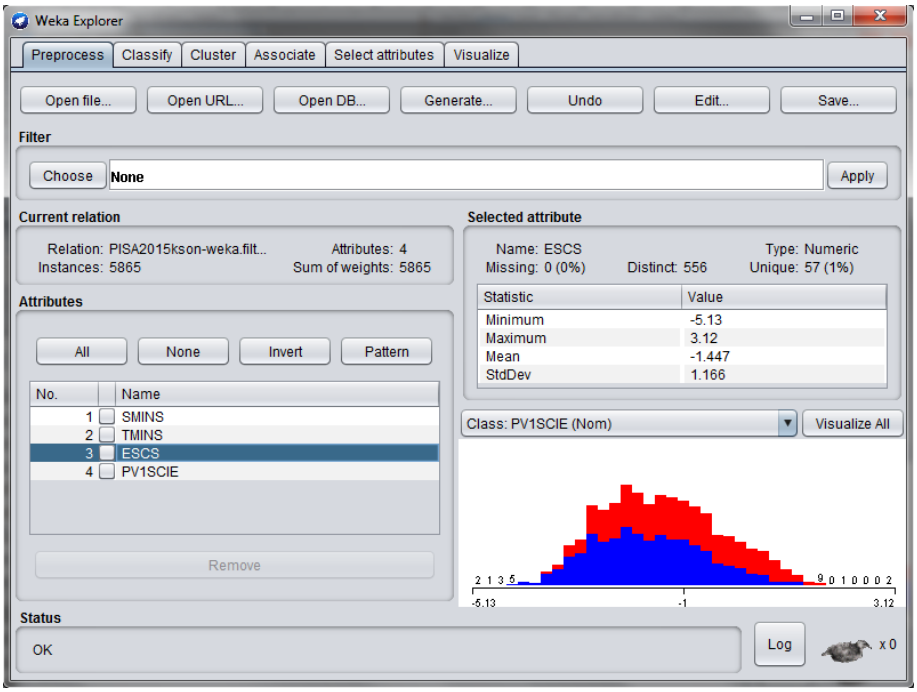

Figure 6. Uploading Input and Output Variables to the Program

After this process, Classify interface is opened, the window shown in Figure 7 is reached by clicking the Choose button in the Classifier window and one of the decision trees under the Trees tab is chosen. As an example, analyzes were performed by selecting J.48 method.

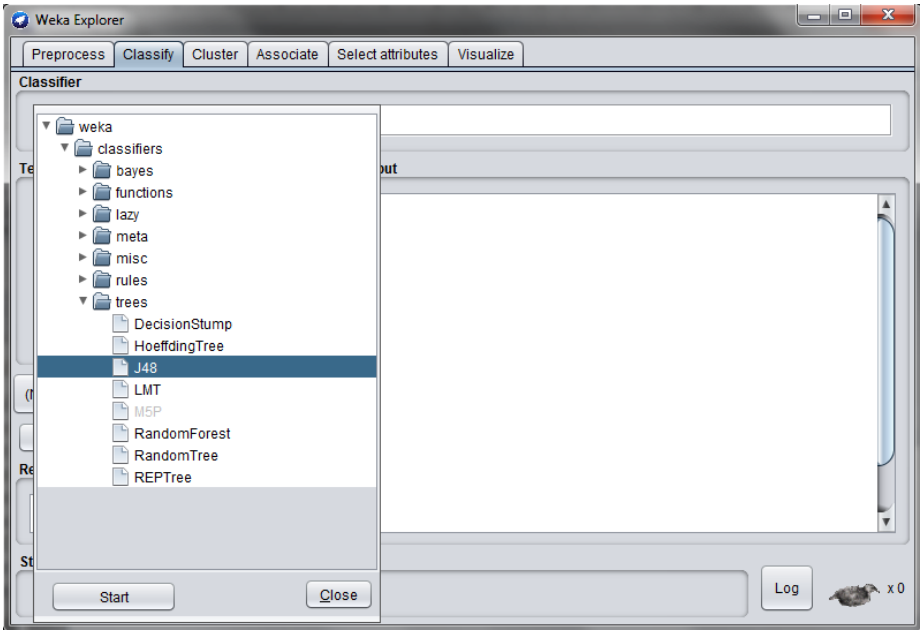

Figure 7. Determination of Decision Tree Method

The analysis process is completed by clicking the Start button on the screen shown in Figure 7. If researchers want to use a different method instead of a 10-fold cross validation method, one of four different validation types can be selected in the test options window. After this process, the window shown in Figure 8 is reached by clicking the Start button.

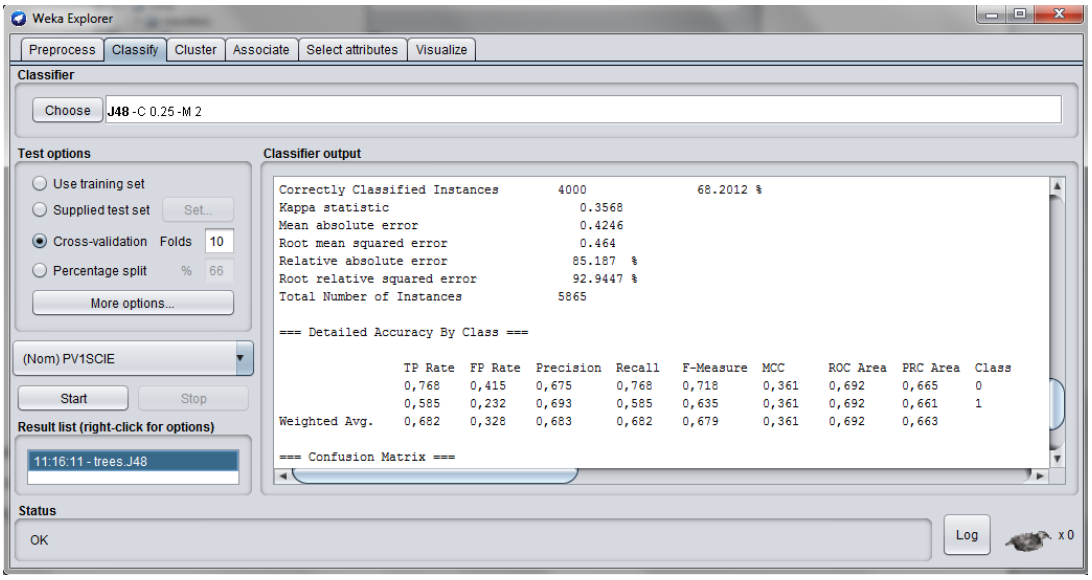

Figure 8. Decision Tree Analysis Outputs

In the classifier output window shown in Figure 8, the number of correctly classified students, statistics of the classification process and confusion matrix are given. Other outputs of the classification process can be seen by moving the cursor down. In order to see the decision tree, researchers should go over the trees.J48 line in the Results list window in the lower left corner of the screen shown in Figure 8 and activate Visualize tree option by right clicking it.

Evaluation of the Results

Within the scope of the study, duration of science learning (smins), total time of learning in school (tmins) and student's socio-economic status index (escs) variables covered in PISA 2015 student survey are defined as input, whereas science literacy level (*pv1scie*) is defined as the output variable, then the decision tree for estimating the output variable with input variables is shown in Figure 9.

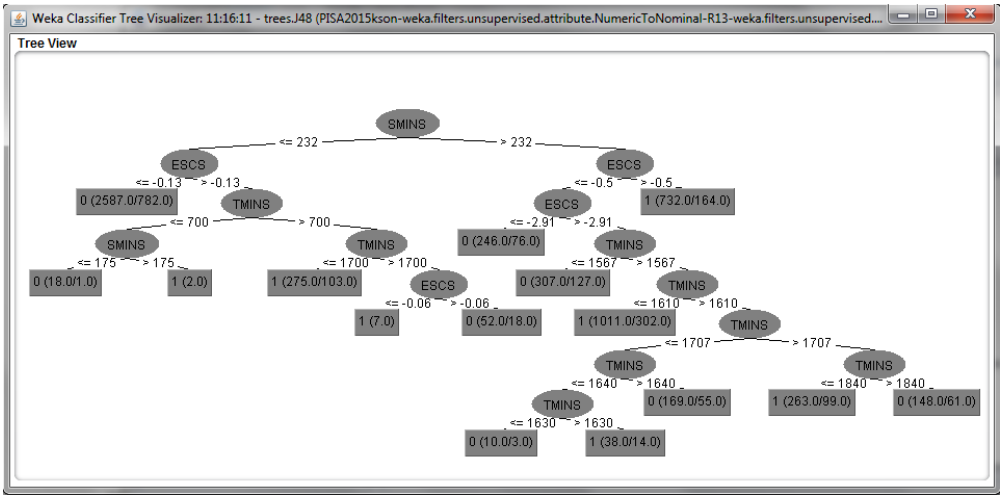

Figure 9. Decision Tree Obtained Through J.48 Method

As can be seen in Figure 9, it is found that duration of science learning (smins) variable has the most significant effect among the three input variables which are covered in the study to classify students in terms of PISA Science literacy. Students are divided into two classes according to the cutting point of the duration of science learning, which is 232. Regarding the students whose science learning time is below 232, a socio-economic status index has the most important effect in the classification of students followed by total learning time variable. At the second level of tree branching, the cut-off point according to socio-economic status index is determined as 0.13 and students below this value are classified as Failed (0). On the same branch, the total learning time of those with a socio-economic status index over 0.13 is checked and the cut-off point for this property is set at 700.00.

In the output file, the classification error of the learning method, and the evaluation results of the tree classification process, including Kappa statistics, the mean absolute error, and the root mean squared error are reported. The root mean squared error is the squared average of the second order loss function. The mean absolute error is calculated by taking the absolute value of the differences instead of taking the errors. Moreover, the output also includes relative error values calculated based on the a priori probability distributions. These statistics are obtained using the ZeroR learning method. Lastly, when the output file is examined, it is seen that detailed accuracy, precision, recall, and F-criterion values are reported for each class.

RESULTS AND DISCUSSION

WEKA program, which is one of many different programs that allows drawing meaningful results from the existing patterns in a data file, seems to be more popular than others because of having a user-friendly interface and being an open source software (Zupan and Demsar, 2008). Especially in quantitative estimation, LMT algorithms based on J4.8, ID3, M5P, and logistic model offer you many options besides Bayesian methods. In addition, it is possible to make estimations with a high level of accuracy and precision with random tree structures such as Random Trees, Raptree and Random Forest. To move one step further than the classical methods of estimating the dependent variable using independent variables, logistic regression model, which has different equations for each branch, is used instead of a single regression equation for the estimation with higher accuracy (Robu and Hora, 2012). Again, offering more than 20 different options under the rules and functions menu is one of the important advantages of the program. In addition, another advantage of the program is being able to read data files with .csv extension directly, without needing any converters.

Although the decision trees for Hoefding tree, J.48, Logistic model, Reptree and Random Tree algorithms can be easily created by the program, the decision trees are not given for the other algorithms, which can be considered as one of the limitations of the program. In addition, the numerical properties with a negative sign and a decimal value cannot be directly read by the program, which is also seen as a limitation.

Data mining is mainly based on classification and prediction algorithms. Although you have obtained a categorical variable when you divide a numeric property into two classes according to a certain threshold value, there is a significant difference between these two. In the classification properties, all information about the property is used in the branching process, whereas in numeric properties you continue to use information about the property in consecutive nondisjunction. In other words, sequential branching in digital properties will continue to produce new information. A classified property can only be tested once, from the root of the tree to a certain leaf of the tree, whereas a numerical feature can be tested several times. Therefore, trees may be more complex and more difficult to understand. Because the tests for a property are not performed together and there may be dispersion along the way. The way to create a tree that is more readable but more difficult to achieve is allowing a multidimensional test for the specified property and testing a few constants on a single node of the tree. For this reason, it is more useful to continue analyzing numeric properties instead of classification. However, at this point, the WEKA program is, unfortunately, unable to predict what will be the numerical value of the dependent variable through independent variables. Although it is possible to obtain the result variable of the relevant example by defining the data related to a single property in

the model with the help of the logistic model established by the learning method using the necessary codes, failing to estimate the dependent variable numerically for all examples is considered to be an important limitation in data mining where very large data is used.

Researchers who are going to study about data mining are recommended to review one of the latest versions of the free program offered by Waikato University by downloading from https://www.cs.waikato.ac.nz/ml/weka/downloading.html, within the framework of the superiorities and limitations mentioned above. In particular, they are advised to repeat the analyzes on the sample data files that the program provides and to interpret obtained results. Besides, Select Attributes feature based on the cross-validation method is thought to be useful for the researchers in reducing the number of variables, especially if there are too many variables at hand. Although the program is open source, researchers who study on the literature are advised to compare the results that they obtained with the command files that they have written with the results of other software, preferably with the assistance of a programmer.

REFERENCES

Bradley, E., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. Boca Raton, USA: Chapman and Hall.

Breiman L., Friedman J. H., Olsen E. A., & Stone C. J. (1984). *Classification and regression trees*. Belmont, CA: Wadsworth International Group.

Chadha, P., & Singh, G. N. (2012). Classification rules and genetic algorithm in data mining. *Global Journal of Computer Science and Technology Software & Data Engineering*, *12*(15), 50-54.

Fayyad, U., & Irani, K. (1992) On the handling of continuous-valued attributes in decision tree generation. *Machine Learning*, *8*, 87–102. doi: 10.1007/BF00994007

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). The KDD process of extracting useful knowledge from volumes of data. *Communications of the. ACM*, *39*(11), 27–34. doi: 10.1145/240455.240464

Han J., Kamber, M., & Pei, J. (2000). *Data mining: Concepts and techniques*. Massachusetts: Morgan Kaufmann.

Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning 11*, 63–91. doi: 10.1023/A:102263111

Ibarguren, I., Perez, J. M., Muguerza, J., Gurrutxaga, I., & Arbelaitz, O. (2014). *An update of the J48Consolidated WEKA's class: CTC algorithm enhanced with the notion of coverage*, (Technical Report EHU-KAT-IK-02-14), Spain: University of the Basque Country

Jain, A. K. (2015). *Indian ethnobotany: Emerging Trends*. Jodhpur: Scientific Publisher.

Johari, R. (2016). MS&E 226: "Small" data lecture 8: Classification problems, lucture notes, Retrived from: http://web.stanford.edu/~rjohari/teaching/notes.html

Karypis, G., Han, E. H., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Institute of Electrical and Electronics Engineers Computer Society*, *32*, 68-74. doi: 10.1109/2.781637

Kohavi, R. (1995, August) *A study of cross-validation and bootstrap for accuracy estimation and model selection*. Paper presented at the In Proceedings of International Joint Conference on AI. Montreal, Canada.

Kushary, D. (2012). Bootstrap Methods and Their Application, *Technometrics*, *42*(2), 216-217. doi: 10.1080/00401706.2000.10486018

Landis, J. R.,& Koch, G. G. (1977) The measurement of observer aggreement of categorical data. *Biometrics*, *31*(3), 159-174.

Lopez, M. I., Luna, J. M., Romero, C., & Ventura, S. (2012). *Classification via clustering for predicting final marks based on student participation in forums*. Paper presented at the 5th International Conference on Educational Data Mining, Chania, Greece.

Mitchell, T. M. (1997). *Machine learning*. New York: McGraw-Hill.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning.* San Mateo, CA: Morgan Kaufmann.

Padmavathi, J. (2012). Logistic regression in feature selection in data mining. *International Journal of Scientific & Engineering Research*, *3*(8), 1-4.

Perlich, C., Provost, F., & Simonoff, J. S. (2002). Tree induction vs. logistic Regression: A learning-curve analysis. The Journal of Machine Learning Research, *4*(1), 211-255. doi: 10.1162/153244304322972694

Powers, D. M. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, *2*(1), 37–63. doi: 10.9735/2229-3981

Refaeilzadeh P., Tang L., & Liu, H. (2007). *On comparison of feature selection algorithms*. In Proc. AAAI-07 Workshop on Evaluation Methods in Machine Learing II. Vancouver, British Columbia, Canada, July 2007.

Robu, R., & Hora, C. C. (2012, June). *Medical data mining with extended WEKA*, Paper presented at the IEEE 16th International Conference on Intelligent Engineering Systems (INES), Lisbon, Portugal.

Rokach, L., & Maimon, O. Z. (2008). *Data mining with decision trees: Theory and applications*. Singapore: World Scientific Publishing Co., Inc.

Schwenke, C., & Schering, A. (2007). *True positives, true negatives, false positives, false negatives.* New Jersey, USA: Wiley Encyclopedia of Clinical Trials.

Sinha, A. P., & May, J. H. (2005). Evaluating and tuning predictive data mining models using receiver operating characteristic curves. *Journal of Management Information Systems*, *21*(3), 249-280.

Sumathi, S., & Sivanandam, S. N. (2006). *Introduction to data mining principles*. Berlin: Springer-Verlag.

Şeker, S. E. (2016). *Weka ve veri madenciliği*, Retrieved from https://www.dr.com.tr/ekitap/weka-ile-veri-madenciligi

Weiss, G. M., & Davison, B. (2010). Data mining, In H. Bidgoli (Ed.), *The handbook of technology management* (pp.2-17), New Jersey: John Wiley and Sons.

Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques*. San Francisco: Morgan Kaufmann.

Witten, I. H., Frank, E., & Hall, M. (2016). *Data minig: Practical machine learning tools and techniques*. San Francisco: Morgan Kaufmann.

Zupan, B., & Demsar, J. (2008). Open-source tools for data mining, *Clinics in Laboratory Medicine, 28(1)*, 37-54. doi: 10.1016/j.cll.2007.10.002