Applying Data Mining Techniques to Implement the Clinical Guidelines for the Management of the Patients with Type 2 diabetes: Medication Dose Adjustments

Gulay CICEK¹, Kadir TUFAN¹, Halil EROL²
¹Computer Engineering Department, Fatih University, Turkey
²Department of Internal Medicine, Fatih University School of Medicine Hospital, Turkey

Abstract: Diabetes is an essential community health problem which is approaching epidemic proportions globally. The prevalence of chronic, non-infectious disease is increasing at alarming rates worldwide. Every year 18 million people die from cardiovascular illness, for which diabetes mellitus and hypertension are the main predisposing elements. Many patients with type 2 diabetes meet with health care providers every 3-4 months to evaluate medication usage, blood sugar, and insulin dosing. The primary treatment for type 2 diabetes is weight reduction. Physicians prescribe dietary restriction and exercise to patients with type 2 diabetes. When these measures are unsuccessful for controlling high blood sugar, oral medication are used. Before adjusting oral medication, the physician examines some biochemical characteristics of diabetic patient such as blood and urine tests. Blood and urine analysis by experts is very time consuming. Methods to automate the interpretation of diabetic data by minimizing human efforts are critical for adjusting medication dosages. Data mining can be a solution to this problem by producing rules from enormous datasets which can be used in analyzing diabetic data. Our aim is to create a tool that could be used to adjust medication dosage. A medication dosage algorithm was developed using data mining techniques. Input parameters consisted of age, gender, alanine aminotransferase, aspartate transaminase, glycosylated hemoglobin (HbA1c), HDL cholesterol, creatinine, LDL cholesterol, microalbuminuria, triglyceride and urea. The dosage of the drug Metformin was adjusted according to the results of the algorithm.

Keywords: Diabetes Mellitus, medication dose adjustment, data mining, classification

Received April 22, 2015; accepted December 27, 2015

1. Introduction

Diabetes is a significant public health challenge that has reached epidemic proportions in recent years [24]. The overall prevalence of non-communicable disease has increased at a tremendous rate and as many as 18 million people succumb to cardiovascular disease each year; diabetes and hypertension are major risk factors for morbidity and mortality associated with heart disease. Current estimates suggest that as many as 312 million adults are obese while 1.7 billion are significantly overweight throughout the world [9]. Furthermore, 155 million children may be overweight or obese. These statistics suggest that a worldwide diabetes epidemic is already underway. The International Diabetes Federation estimates that the relative prevalence of the disease was 8% in 2007 and will reach 7.3% by 2025 [15]. A total of 246 million people are affected by diabetes (46% of whom are between 40 and 59 years of age) and as many as 380 million may have active diabetes by 2025. There are widespread concerns regarding this increase in type 2 diabetes. Many of people with diabetes in developed countries are above the retirement age.

In those countries, many people are diagnosed with diabetes during middle age, between 30 and 65 years, which is the most productive phase of life [9]. According to estimation of International Diabetes Federation, every year 7 million human beings suffer from diabetes. Incidence of type 2 diabetes mellitus has increased dramatically [14]. Lifestyle changes have been demonstrated to produce significant improvements in overall health among diabetes patients. Many patients with type 2 diabetes meet with health care providers every 3-4 months to evaluate medication usage, blood sugar, and insulin dosing. The primary treatment for type 2 diabetes is weight reduction. Physicians prescribe dietary restriction and exercise to patients with type 2 diabetes. When these measures are unsuccessful for controlling high blood sugar, oral medication are used. Before adjusting oral medication, the physician examines some biochemical characteristics of diabetic patient such as blood and urine tests. As a result of these tests physicians give appropriate drugs and adjust dosages [10]. Blood and urine analysis by experts is very time consuming and there is a shortage of experts possessing knowledge on the analysis of diabetic data. Methods to automate the
interpretation of diabetic data by minimizing human efforts are critical for adjusting medication dosages. Data mining is the practice of applying computation science to large data sets and extracting statistical inferences [12]. Data mining can be a solution to this problem by producing rules from those enormous datasets which can be used in analyzing diabetic data. Data mining has become a significant component of health care delivery and evaluation [25]. Data mining includes classification, clustering, prediction and association of independent datasets. It also spans other disciplines like Data Warehousing, Statistics, Machine Learning and Artificial Intelligence [2-10].

Data mining techniques can be useful in healthcare [4]. They are successfully employed in diagnosis, prognosis and evaluating the quality of some treatments [25]. Healthcare organizations that effectively implement data mining are able to better meet long term needs [15]. Data can be a tremendous asset for healthcare organizations; however datasets must be adapted for digital manipulation [8].

To estimate the result of a specific dosage of a medication is one of the most interesting applications of data mining. A recently developed resource, the Knowledge Discovery in Databases (KDD), is a widely used web application that enables the medical investigator to determine medication dosage based on large number of historical cases stored in a digital archive. A number of different classification methods can be used to evaluate this type of data, including Support Vector Machine [25-28], Neural Networks [11-23], Decision Trees, [30-12] K Nearest Neighbor, Genetic and Rule based induction.

1.1. Statement of The Problem

The International Diabetes Federation (IDF) reports that diabetes mellitus affects 370 million individuals worldwide, among whom are 130 million individuals who are unaware of their illness. These statistics represent an increase over the 366 million estimated cases in 2011. Up to 48 million people will die as a result of the complications of diabetes each year, with patients under the age of 60 accounting for 50% of deaths [17].

Diabetes has been traditionally characterized as a disease of affluent countries. However, 60% of diabetes patients reside in low and middle income communities. Approximately 3.6 billion adults reside in high-risk low- and middle- income communities as of 2011. By comparison, 75 million cases of adult diabetes originate from high-income countries [1]. Type 2 diabetes accounts for 85% of diabetes cases worldwide. Many patients with type 2 diabetes meet with health care providers every 3-4 months to evaluate medication usage, blood sugar, and insulin dosing. Our aim is to create a tool that could be used in adjusting dosage of medication. Medication dosage is adjusted using data mining techniques.

2. Research Methodology

In this study, the aim is to develop an estimation model that can predict the dosage of certain drugs based on the given datasets. The overall approach involved the use of the KDD methodology, the aim of which is to extract specific knowledge from data in the context of large databases [30].

a. To understand the application domain

To define the problem and specify medical goals, I have worked closely with Dr. Erol, who is an internist at Fatih University School of Medicine. The discussions with this specialist have helped me clarify specific challenges and become familiar with the current solutions to those problems. A key sub goal in this technique is to determine the data mining success criteria. Our aim is to translate medical goals into data mining goals.

b. Selecting and creating the target dataset

We used data which were collected by Fatih University School of Medicine and include patients with type 2 diabetes. There were 63 patient records in the dataset. This data set was imbalanced [21]. To make balance our dataset, we use smooth technique in weka. So Number of instances become 560. To identify the most important features while creating the training data, I took advice from the medical expert. My medical mentor selected the most important features as inclusion criteria in this dataset, which is called the “training data”.

c. Pre-processing and cleansing

Data cleaning is the step where noise and irrelevant data are removed from the large data set. In a given dataset, there can be duplicate records, unnecessary fields, and missing values. To increase the quality of the selected data, these problems must be addressed. In our dataset, we have some missing values and duplicate records. Missing values were replaced with the most probable value as determined by regression. To solve duplicate records in dataset, all records were reviewed and duplicate records were manually eliminated.

d. Data transformation

The goal of data transformation is to reduce the number of effective variables to include only the most useful attributes which can address the specific goals of the task. Data is converted to the most appropriate format for data mining applications. In that case, after taking advice from the medical expert, a few
transformations were required to make the data more suitable for data mining algorithms.

In this study, we used two transformation methods which are called discretization and attribute selection.

To reduce the number of attributes in a dataset, attribute selection was applied. Attribute selection was needed to decrease the number of features as a classification algorithm to be examined and reduce errors resulting from inconsequential attributes. We have used the ranker search method to select the most appropriate attributes from 11 features that were available in the dataset. Discretization was applied for converting continuous values variables to discrete values.

This data transformation method was applied to reduce the number of distinct values of continuous variables by allowing a limited number of labels to demonstrate the original variables.

e. Selecting Suitable Data Mining Task

In this part, our aim is to determine what type of data mining modelling should be used. Choosing a suitable model based on the purpose of the study, and we aimed to adjust drug dosages using data mining techniques. There are different types of algorithms for creating these models, such as Neural Network, Decision Tree, Support Vector Machine, and Bayesian Classification [20]. To select the best model, we looked the performance of each algorithm.

f. Employing Data Mining Algorithm

Classification algorithm experiments were planned and conducted on a full training dataset including 63 instances. In all of the experiments two scenarios were considered, one containing all 11 attributes and the other containing only 8 selected attributes. A method of 10-fold cross validation was adopted for conducting random sampling of the training and test data sets.

g. Evaluation

In this step, models that are created will be evaluated according to their performance. The performance of individual models was evaluated using a series of algorithms including classification accuracy and a confusion matrix.

h. Tools

We need to use data mining algorithms for adjusting the drug dosage. Weka™ workbench, a collection of machine learning algorithms for data mining tasks, was used to determine drug dosage from the available datasets [6].

3. Algorithms Used for Model Building and Performance Measures

In this study, our aim is to estimate medication dosage for a given patient with type 2 diabetes. An attempt was made to construct a prediction model using Decision Tree and Neural Network methods. After building the models, performance of each of the models were evaluated, and their performances were compared to each other. In this part, the algorithms used to build the models and the matrices used for performance measures and comparisons are discussed in detail.

3.1. Decision Trees

A decision tree is a classification algorithm with recursive or iterative partition of the instance space; it consists of a series of nodes forming a rooted tree in which the root has no incoming edge. Nodes with outgoing edges are frequently referred to as internal or test nodes. Terminal nodes are referred to as “leaves” or decision nodes. Internal nodes in a decision tree are divided into two or more subspaces as a result of a specific discrete function of the input attribute. In the most basic sense, each test evaluates a single attribute, and the instance space is separated according to attribute values. A range is often applied for numeric attributes. Individual leaves on a decision tree are directed at one class and indicate the most appropriate target value. Circles represent internal nodes while leaves are indicated by triangles. Multiple branches may be drawn from each internal node. Each node and its branches correspond to a certain value range. This range specifies a partition within the possible values of a given attribute.

Individual instances are categorized by navigation from the root node to the terminal leaf, applying test characteristics at each intermediate node. A decision tree contains both nominal (categorical) and numeric features. Nodes are characterized by the attribute tested and branches are labeled according to specific value partitions. Similar data mining algorithms are widely used in medicine, molecular biology, manufacturing, and financial analysis [10].

Many human analysts find decision tree systems easy to understand. The discrete steps of classification within a decision tree occur quickly and simply. Decision tree algorithms may achieve high levels of accuracy; however this will be dependent upon the amount and quality of the data available. As in many areas, Decision tree algorithms are used such as medicine, production and manufacturing and so on.
3.1.1. J48 Classifier Algorithm

J. Ross Quinlan, a machine learning researcher, has developed an improved decision tree algorithm known as Iterative Dichotomiser (ID3). A later successor, C4.5, improved upon the Iterative Dichotomiser (ID3).

This algorithm is implemented in a serial manner. A pruning method used in C4.5 replaces internal nodes with leaf nodes, reducing the overall error rate. The ID3 algorithm functions as a non-categorical feature. However the C4.5 algorithm can utilize both continuous and categorical features when generating a decision tree. Both ID3 and C4.5 utilize a gain ratio to determine the optimal splitting attribute.

Entropy reduction is used to generate the optimal splitting attribute in C4.5

- Given probabilities \( p_1, p_2, \ldots, p_s \) \((p_i \text{ is the probability of class } s_i)\), whose sum is 1. Entropy is defined as:

\[
H(p_1, p_2, \ldots, p_s) = \sum_{i=1}^{s} p_i \log \left( \frac{1}{p_i} \right)
\]  

(1)

Given candidate split S, which partitions the training dataset \( T \) into several subsets, \( T_1, T_2, \ldots, T_k \) the mean information requirement can then be calculated as the weighed sum of the entropies for the individual subsets, as follows:

\[
H(S) = \sum_{i=1}^{k} p_i H_i(T_i)
\]  

(2)

Gain can subsequently be calculated using the following formula:

\[
Gain(D, S) = H(D) - \sum_{i=1}^{k} \left( p(D) H_i(D) \right)
\]  

(3)

Information gain tells us which attribute can be a top part. At each successive decision node, C4.5 determines the optimal split based on maximized information gain. J48 implements the C4.5 algorithm to generate both pruned and un-pruned C4.5 decision trees. Decision trees generated by J48 can be used for accurate classification.

3.1.2. Neural Network

Neural networks are interconnected nodes with directional links. Individual nodes represent processing units, and the links between nodes reflect causal links between these processes. Neural network systems were originally generated to mimic the neurophysiology of the human brain using simple computational elements (neurons) in a highly interconnected system [2].

Neural network architectures contain at least two layers, an input and an output layer. Additional layers can be inserted between the input and output. The input layer may contain multiple nodes, each of which represents a unique predictor variable. Nodes are the foundation of the neural network. All input nodes can be linked to other input nodes within the hidden layer, to other layers, or to the output. Multiple nodes in the output layer represent response variables. Each node receives a set of inputs which are multiplied by connection weight \( W_{ij} \) (e.g., the weight from node 1 to 3 is \( W_{i3} \)) and added together (Two Cross Corporation, 1999). The activation function is then applied to the output before passing the function to the next layer.

Activation function is applied as following:

- Multiplies each input by its weight.
- Applies an activation function to the sum of results.

3.1.2.1. Multilayer Perception

Multilayer perception (MLP) is a type of neural network that has become the most widely used in data mining. Multilayer perception includes multi-layer nodes arranged in a directed graph. All nodes in a layer are linked to a subsequent layer.

Neurons, or nodes within layers, have non-linear function except for the input layer. A supervised training technique known as back propagation trains the network in multilayer perception.

The forward feed of the network propagates from the input to the output without iteration. A feed
forward neural network does not include connections between nodes in the same layer; output nodes in a specific layer are linked to specific input nodes in succeeding layers. This modular design is generally preferred, with nodes in a shared layer sharing some functionality or generating comparable abstractions [2].

4. Performance Measures

Cross-Validation is a statistical method of evaluating and comparing learning algorithms by dividing data into two segments: one used to learn or train a model and another used to validate the model [4]. The primary method of statistical cross-validation is the K-fold method. In K-fold cross-validation, data are randomly divided into mutually exclusive subsets or ‘folds’ \( D_1, D_2, \ldots, D_k \) each with similar size. Testing and training are implemented \( k \) times [7].

There are two primary reasons to use cross validation statistics:

- To generate algorithm performance data associated with a learned model based on existing data
- To compare performance among multiple algorithms using a given dataset

4.1. 10-Fold Cross Validation

In 10-fold CV each dataset is randomly separated into 10 reciprocally exclusive subsets of comparable size. The constructed model is trained and tested 10 times. The algorithm is trained in the first iteration and tested in the final iteration.

In order to perform 10 fold CV, a set of general principles may be followed:
1. Dataset is divided into 10 equal parts. Each part is known as a fold.
2. Create a model using all records in each fold except one fold. The excepted fold is used for the testing purpose.
3. Step 2 is replicated 10 times and average accuracy is calculated.

4.2. Confusion Matrix

Many different mechanisms are used to measure performance in classification problems. If there are \( k \) classes, the size of the row and column of a confusion matrix is equal to \( k \). Size of the table is \( k \) by \( k \). If instance is positive and this is classified as positive; it is counted as TP; if it is classified as negative, it is counted as a False Negative (FN). If the instance is negative and it is classified as negative, it is counted as a False Negative (FN). If the instance is negative and it is classified as positive, it is counted as True Negative (TN); positive instances are considered False Positive (FP)”.

Accurate identification of true/false positive and true/false negative results is critical to determining the accuracy of the algorithm.

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>True Positive</td>
</tr>
<tr>
<td>C2</td>
<td>False Negative</td>
</tr>
<tr>
<td>C2</td>
<td>False Positive</td>
</tr>
<tr>
<td>C1</td>
<td>True Negative</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROC Area</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9 - 1.0</td>
<td>Excellent (A)</td>
</tr>
<tr>
<td>0.8 - 0.9</td>
<td>Good (B)</td>
</tr>
<tr>
<td>0.7 - 0.8</td>
<td>Fair (C)</td>
</tr>
<tr>
<td>0.6 - 0.7</td>
<td>Poor (D)</td>
</tr>
<tr>
<td>0.5 - 0.6</td>
<td>Fair (F)</td>
</tr>
</tbody>
</table>

5. Experimentation

As the aim of this study is to adjust medication dosage using data mining techniques, a classification technique was adopted to improve a predictive model. The models were built with two separate supervised
machines, i.e., Decision Tree and Neural Network using Weka 3.6.4 machine learning software.

5.1. Experimental Setup

For this study, experiments were handled and every experiments two scenarios were considered, one including all the 11 features and the other including 9 selected attributes. So with each experiment and eight different scenarios a total of eight models were built. First Experiments were conducted on a full training dataset including 63 instances and 10-Fold Cross Validation was adopted for randomly sampling the training and test sets.

While performing the experiments all parameters were set to their default setting for each algorithm except for J48 classifier where the parameter “Unpruned” which had a default value “False” was changed to “True” for the first experiment to observe the performance of J48 unpruned tree.

In this study, the performance of models were evaluated using standard metrics of accuracy, precision, recall and F-measure which were computed using predictive classification table, known as Confusion Matrix. For comparing the performance measures, Receiver Operating Characteristic was also used.

5.1.1. Model Building Using J48 Decision Tree

In J48 Decision Tree Classifier, Two experiments were conducted. We have designed two experiments for investigating:

- The effect of attribute selection on classification accuracy on unpruned J48 Decision Tree Classifiers.
- The effect of attribute selection on classification accuracy on pruned J48 Decision Tree Classifiers.

5.1.1.1. Experiment 1

The first experiment was designed to evaluate the performance of a J48 classifier unpruned tree in predicting dosage of medication and investigate the effect selection on the performance of the model. In this experiment two scenarios were considered, one including all 11 attributes and the other including the selected 9 attributes.

One first scenario the algorithm was run on a full training set including 560 (number of instances is increased because of imbalanced dataset) instances with 11 attributes. On the second scenario the algorithm was run on a cross validation set including 560 instances with selected 9 attributes.

The model built with J48 unpruned tree with all attributes correctly classified 503 (89.8214%) instances while 57 (10.1786%) of the instances were classified incorrectly. The overall accuracy rate of the model is highly successful.

The second model was built with J48 unpruned tree with selected 9 attributes correctly classified 501 (89.4643%) instances while 59 (%10.5357) of the instances were classified incorrectly. Like the experiment the overall accuracy rate of the model is highly successful.

5.1.1.2. Experiment 2

The second experiment was designed to investigate:

- The performance of a J48 classifier pruned tree in estimating dosage of medication.
- The effect of attribute selection on the performance of a J48 classifier pruned tree model.
- The effect of the tree pruning methods when building a J48 decision model.

Similar to experiment 1, in this experiment two scenarios were considered, one containing all 11 attributes and other containing the selected 9 attributes. On the first scenario the algorithm was run on a full training set including 560 instance with all attributes. On the second scenario the algorithm was run on a full training set including 560 instances with only 9
selected attributes. Shown in Table 4. Confusion matrices for Experiment 2 and Table 5. Detailed Performance Measures for Experiment 2.

Table 4. Confusion matrixes for Experiment 2.

<table>
<thead>
<tr>
<th>Predicted</th>
<th>500</th>
<th>1000</th>
<th>1500</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>J48 pruned with all attributes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>9</td>
<td>5</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>140</td>
<td>1</td>
<td>17</td>
<td>500</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>81</td>
<td>7</td>
<td>1000</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>3</td>
<td>124</td>
<td>1500</td>
</tr>
</tbody>
</table>

| J48 pruned with selected attributes |
| 158 | 9 | 5 | 4 | 0 |
| 2 | 141 | 1 | 16 | 500 |
| 5 | 3 | 78 | 10 | 1000 |
| 0 | 2 | 4 | 122 | 1500 |

Table 5. Detailed Performance Measures for Experiment 2.

<table>
<thead>
<tr>
<th>Model: J48 unpruned with all attributes</th>
<th>Test options: Cross Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>87.14 %</td>
</tr>
<tr>
<td>TP Rate</td>
<td>0.87 %</td>
</tr>
<tr>
<td>FP Rate</td>
<td>0.04 %</td>
</tr>
<tr>
<td>Precision</td>
<td>0.88 %</td>
</tr>
<tr>
<td>F-Measure</td>
<td>0.87 %</td>
</tr>
<tr>
<td>ROC Curve</td>
<td>0.94 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model: J48 unpruned with selected attributes</th>
<th>Test options: Cross Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>87.15 %</td>
</tr>
<tr>
<td>TP Rate</td>
<td>0.70 %</td>
</tr>
<tr>
<td>FP Rate</td>
<td>0.04 %</td>
</tr>
<tr>
<td>Precision</td>
<td>0.88 %</td>
</tr>
<tr>
<td>F-Measure</td>
<td>0.87 %</td>
</tr>
<tr>
<td>ROC Curve</td>
<td>0.94 %</td>
</tr>
</tbody>
</table>

5.1.2. Model Building Using Neural Network

This experiment was designed to explore the ability of Neural Network in estimating dosage of medication. From Neural Network Algorithms Multilayer perception was selected to conduct in experiment. In this experiment like in all of the experiment two scenario was conducted one containing all 11 attributes and other containing the selected 9 attributes.

5.1.2.1. Experiment 3

On the first scenario the algorithm was run on a cross validation set containing 560 instances with 11 attributes. On the second scenario the algorithm was run on a cross validation set containing 560 instances with 9 selected attributes. Shown in Table 6. Confusion Matrixes for Experiment 3 and Table 7. Detailed Performance Measures for Experiment 3.

Table 6. Confusion matrixes for experiment 3.

<table>
<thead>
<tr>
<th>Predicted</th>
<th>0</th>
<th>500</th>
<th>1000</th>
<th>1500</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neural Network with all attributes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>7</td>
<td>7</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>70</td>
<td>10</td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>126</td>
<td>1500</td>
<td></td>
</tr>
</tbody>
</table>

| Neural Network with selected attributes |
| 164 | 5 | 7 | 0 | 0 |
| 12 | 4 | 71 | 9 | 1000 |
| 0 | 1 | 1 | 125 | 1500 |

Table 7. Detailed performance measures for experiment 3.

<table>
<thead>
<tr>
<th>Model: Test options:</th>
<th>Neural Network with all attributes</th>
<th>Cross Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>87.14 %</td>
<td></td>
</tr>
<tr>
<td>TP Rate</td>
<td>0.87 %</td>
<td></td>
</tr>
<tr>
<td>FP Rate</td>
<td>0.04 %</td>
<td></td>
</tr>
<tr>
<td>Precision</td>
<td>0.88 %</td>
<td></td>
</tr>
<tr>
<td>F-Measure</td>
<td>0.87 %</td>
<td></td>
</tr>
<tr>
<td>ROC Curve</td>
<td>0.94 %</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model: Test options:</th>
<th>Neural Network with selected attributes</th>
<th>Cross Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>87.85 %</td>
<td></td>
</tr>
<tr>
<td>TP Rate</td>
<td>0.87 %</td>
<td></td>
</tr>
<tr>
<td>FP Rate</td>
<td>0.04 %</td>
<td></td>
</tr>
<tr>
<td>Precision</td>
<td>0.88 %</td>
<td></td>
</tr>
<tr>
<td>F-Measure</td>
<td>0.87 %</td>
<td></td>
</tr>
<tr>
<td>ROC Curve</td>
<td>0.94 %</td>
<td></td>
</tr>
</tbody>
</table>

The first Neural Network model built on all 11 attributes correctly classified 488 (87, 1429 %) instances while 72(12, 8571 %) of the instances were classified incorrectly. The overall accuracy rate of the model is very high. But it is not better compared to J48 pruned and unpruned model.

The second Neural Network model built on 8 selected attributes correctly classified 492 (87, 1229%) instances while 62(12, 8771 %) of the instances were classified incorrectly.

6. Conclusions

In this study, the aim was to design a predictive model for adjusting dosage of medication using data mining techniques. Data collected by Fatih University School of Medicine from the year including 63 instances was selected and preprocessed for this study. The models were built on the preprocessed Metformin Dataset with two different supervised machine learning algorithms i.e., J48 Classifier and Multilayer Perception using Weka 3.6.4 machine learning software.

The performance of the models were evaluated using the standard metrics of accuracy, precision, recall and F-measure. 10 – Fold Cross Validation was adopted for randomly sampling the training and test data samples. All six models performed well in
predicting dosage of medication. The most effective model to be J48 classifier implemented on selected attributes and all attributes with a classification accuracy of 91.76%.

From a total of 11 attributes that were available, all of them are highly relevant in estimating medication dosage of medication from Metformin dataset were selected. Because when we remove two feature from dataset, we can acquire low performance.

This study showed that data mining techniques can be used efficiently to model and predict dosage of medication. The outcome of this study can be used as assistant tool by physicians.

References


Gulay CİCEK received her diplomas in Computer Engineering from Fatih University where she received both a BSc (2012) and the MSc (2014) degrees. She also received a BS degree in Management in 2011 at Anadolu University. She is a PhD student in biomedical engineering at Fatih University in Istanbul, Turkey.

Kadir TUFAN got B.Sc. degree from Electrical and Electronics Department, Middle East Technical University in 1995. He completed his M.Sc. in 2003 at the Biomedical Engineering Institute of Bosphorus University. He earned the PhD degree from Electrical and Electronics Department of Fatih University in 2011. He is currently faculty member and chairman of the Computer Engineering Department of Fatih University. Dr. Tufan’s research interests are biomedical signal processing, image processing, data mining and pattern recognition.

Halil EROL is an Assistant Professor of Clinical Medicine at Fatih University Hospital in Istanbul. He received his MD degree in 1997 from Hacettepe University, Ankara. He completed his Internal Medicine residency at Hacettepe University Hospital and did his subspeciality training in Endocrinology at Ankara Numune Hospital. He completed a 3-year postdoctoral research fellowship at the University of Miami in Florida, USA in the field of leptin deficiency and obesity.