T.C
BAHÇEŞEHİR ÜNİVERSİTESİ

DEVELOPING AN EXPERT-SYSTEM FOR DIABETICS BY SUPPORTING WITH ANFIS

Master Thesis

ALİ KARA

İSTANBUL, 2008

T.C

BAHÇEŞEHİR ÜNİVERSİTESİ

INSTITUTE OF SCIENCE

COMPUTER ENGINEERING

DEVELOPING AN EXPERT-SYSTEM FOR DIABETICS BY SUPPORTING WITH ANFIS

Master Thesis

Ali KARA

Supervisor: ASSOC.PROF.DR. ADEM KARAHOCA

İSTANBUL, 2008

T.C

BAHÇEŞEHİR ÜNİVERSİTESİ

INSTITUTE OF SCIENCE

COMPUTER ENGINEERING

Name of the thesis: Developing an Expert-System for Diabetics by supporting with ANFIS
Name/Last Name of the Student: Ali Kara
Date of Thesis Defense: Jun .09. 2008

The thesis has been approved by the Institute of Science.

Prof. Dr. A. Bülent ÖZGÜLER
Director

____

I certify that this thesis meets all the requirements as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Adem KARAHOCA
Program Coordinator

____

This is to certify that we have read this thesis and that we find it fully adequate in scope, quality and content, as a thesis for the degree of Master of Science.

Examining Committee Members	Signature
Assoc.Prof.Dr. Adem KARAHOCA	____________________
Prof.Dr. Nizamettin AYDIN	____________________
Asst.Prof.Dr. Yalçın ÇEKİÇ	____________________

To my father

ACKNOWLEDGEMENTS

This thesis is dedicated to **my father** for being a role model in front of my educational life.

I would like to express my gratitude to **Assoc. Prof. Dr. Adem Karahoca**, for not only being such a great supervisor but also encouraging and challenging me throughout my academic program.

I also wish to thank **R. Tolga Şen**, for supplying various real data of diabetes to be processed during development, and Academic Hospital, for allowing me to use their medical forms.

And I would finally like to thank to **my spouse** for her endless patience.

Ali Kara

ABSTRACT

DEVELOPING AN EXPERT-SYSTEM FOR DIABETICS

BY SUPPORTING WITH ANFIS

Kara, Ali

M.S. Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Adem Karahoca

June 2008, 40 pages

Medical researches and questionnaires declare that there are approximately 5 million diabetic patients in Turkey. Unfortunately majority of them don’t realize that they are in danger of diabetes. It is thought difficult to visit a doctor and examine the results of their insulin measurement. We intend to develop an expert system, which both examines the medical results of potential patients and leads the patients during all their lives. While developing the system, the main aim is to reach as many patients as we can. So web technologies and development tools were used to create our expert system.

After developing a web based expert system, it’s needed to benchmark of data mining techniques using socio-demographic data of diabetic patients, in order to reveal diabetes map of Turkey, to find association rules among the social-demographic data and to apply Adaptive Neuro Fuzzy Inference System (ANFIS). Via benchmarking ANFIS with multinomial logistic regression (MLR), it’s seen that ANFIS is more effective than MLR using fuzzy diabetes data.

Key words: expert system, Prolog Server Pages (PSP), diabetes diagnosis, WEKA, data mining, adaptive neuro fuzzy inference system (ANFIS), logistic regression

ÖZET

DİYABET HASTALIĞININ TANI VE TEDAVİSİ

İÇİN

ANFIS DESTEKLİ UZMAN SİSTEM GELİŞTİRİLMESİ

Kara, Ali

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi: Doç. Dr. Adem Karahoca

Haziran 2008, 40 sayfa

Tıbbi araştırmalar ve anketler, Türkiye'de 5 milyondan fazla diyabet (şeker) hastası bulunduğunu ortaya koymaktadır. Ancak bu hastaların büyük çoğunluğu maalesef diyabet tehlikesinde olduklarının farkında değildirler. Uzman bir doktoru ziyaret etmek, muayene olmak ve insülin tedavisinde dozajı ayarlamak için doktorla görüşmek hastalara zor gelmektedir. Hem potansiyel hastaların risk oranını belirlemek, hem de diyabetlilerin tedavileri boyunca yol gösterici bir uzman sistem geliştirmek istendi. Böyle bir sistemi geliştirmedeki ana amaç, ulaşabilindiği kadar fazla diyabet hastasına ulaşabilmekti. Bu nedenle, Internet üzerinden de kullanılabilen bir sistem olması için, web teknolojilerini ve yazılım geliştirme araçları kullanıldı.

Web tabanlı uzman sistemi geliştirdikten sonra, diyabet hastalarının sosyo-demografik verilerini kullanarak veri madenciliği tekniklerinin karşılaştırılması istendi. Bu amaçla, diyabet hastalarının sahip olduğu sosyo-demografik veriler arasında birliktelik kurallarının çıkarılması sağlandı ve ANFIS yardımı ile kestirim yapıldı. Son olarak ANFIS' in lojistik regresyon ile kıyaslanması ile, ANFIS' in daha etkili bir öğrenme ve kestirim aracı olduğu görüldü.

Anahtar Kelimeler: uzman sistem, Prolog Server Pages (PSP), diyabet tanı ve tedavisi, WEKA, veri madenciliği, ANFIS, benchmarking, lojistik regresyon.

TABLE OF CONTENTS

TABLES

FIGURES

ABBREVIATIONS

Artificial Intelligence	:	AI
Adaptive Neuro Fuzzy Inference System	:	ANFIS
Body Mass Index	:	BMI
High Density Lipoprotein	:	HDL
Low Density Lipoprotein	:	LDL
Multinomial Logistic Regression	:	MLR
Personal Home Pages	:	PHP
Prolog Server Pages	:	PSP
Relational Database Management System	:	RDBMS
Receiver Operating Characteristic	:	ROC

1. INTRODUCTION

Medical researches and questionnaires declare that there are approximately 5 million diabetic patients in Turkey. But unfortunately most of diabetic patients either don't visit physician regularly or don't know he is already diabetic. Our starting point to develop a diabetes expert system is to help these kinds of patients.

We would like to serve an expert system to diabetic patients or the people who suspect if they have diabetes risk. Our main purpose is to help patients and physicians during medical treatment for dosage planning stage.

The thesis contains 4 main sections. In the first section after introducing the problem and proposed system, infrastructure of the system and its database structure are mentioned about. In addition to this, the 3rd party solutions that work integrated to the diabetes expert system are shortly mentioned about.

The functionalities of the modules are exhibited in section 2. Both functionality and usage are defined in the section supported with screen-shots.

The third section has capability of integration to Weka and Matlab. Some benchmarks are executed in the section in order to examine performances of data mining techniques and statistical methods.

In the final section, the results that we have reached, and the conclusions that the thesis has given us are shared.

1.1 Problem Definition

Life is difficult to diabetic patients. They must measure their glucose rate, inject insulin regularly, visit physician and examine the results. An expert system may help them to minimize measurement time for detecting glucose level. By this way, insulin dosage may be planned effectively.

Once diabetes expert systems that are already used are examined, we saw that they are not web-based application, but founded in 2-tier architecture. Then they carry all disadvantaged of 2-tier architecture, such as difficulties in maintenance, difficulties in upgrade, limited access (you must have enough capacity on workstations) and so on.

Another problem that must be solved by a diabetes expert system is the diabetic map of Turkey. It's needed to work on data mining with a knowledge based of diabetic patients. However the knowledge base must be pre-processed before applying data mining techniques. The diabetes expert system should solve the pre-processing problem and apply data mining techniques such as classification and association rules and apply Neuro-Fuzzy Inference System like ANFIS (Adaptive Neuro-Fuzzy Inference System) (Polat & Gunes 2006).

1.2 Related Works

Any researcher could find expert systems, when he searches about expert systems on Internet or scientific libraries. Some related works about diabetes and experts system are found in the literature research, and mentioned about them in the following section.

When we investigated them, we saw they must be categorized and cannot meet our requirements and problems defined in the Section 1.1.

Related Diabetes Expert Systems

An expert system's central goal is to help professional in the process of shifting from old implementation to modern approaches, based on latest technologies. An expert system assists the human designer by efficient encoding of expert knowledge and by reusing the available systems.

The use of expert systems in the speed-up of human professional work has been in two orders of magnitude with resulting increases in human productivity and financial returns. Last decade shows that a growing number of organizations shift their informational systems towards a knowledge-based approach. This fact generates the need for new tools and environments that intelligently port the legacy systems in modern, extensible and scalable knowledge-integrated systems (Pop & Negru 2003).

The most popular technique of knowledge acquisition is still done with an interaction with a human expert. A knowledge engineer, a person acquiring knowledge, interacts with an expert either by observation of the expert in action or by interview. As a result, rules are produced, first in plain English, later on in the coded form accepted by a computer. It is the responsibility of the knowledge engineer to acquire knowledge in such a way that the knowledge base is as complete as possible (Dobroslawa et al. 1995). The process of working with an expert to map what he or she knows into a form suitable for an expert system to use has come to be known as knowledge engineering. We refer to the process of mapping an expert's knowledge into a program's knowledge base as knowledge engineering.

For the representation of knowledge in expert systems, a number of forms are used, such as: rules set (production rules, association rules, rules with exceptions), decision tables, classification and regression trees, instance-based representations, and clusters. Each representation has its advantages and drawbacks (Pop & Negru 2003).

The knowledge needed to drive the pioneering expert systems was codified through protracted interaction between a domain specialist and a knowledge engineer. While the typical rate of knowledge elucidation by this method is a few rules per man day, an expert system for a complex task may require hundreds or even thousands of such rules (Quinlan 1985).

To avoid drawbacks of the knowledge-based systems, in this thesis, learning-based methodology is used. At this point, to be clear on the framework structure, summarized comparison of the knowledge-based and learning-based approaches is needed.

As searching literature, some expert systems for allergy, cholera, thyroid disease or chronic hepatitis-B are encountered; and also some expert systems for general medical requirements are come across (Pontow et al. 2007, pp. 308–326; Guler & Ubeyli 2006; Gutierrez-Estrada et al. 2006, pp. 110–125; Hwang et al. 2006, pp. 299–308; Karagiannis et al. 2006, pp. 1305-2403).

In addition to this, we encountered earlier works on diabetes expert systems. In this research, on diabetes disease, this is a very common and important disease using principal component analysis (PCA) and adaptive neuro-fuzzy inference system (ANFIS). The aim of this study is to improve the diagnostic accuracy of diabetes disease combining PCA and ANFIS. The proposed system has two stages. Initially, dimension of diabetes disease dataset that has 8 features is reduced to 4 features using principal component analysis. And then, diagnosis of diabetes disease is conducted via adaptive neuro-fuzzy inference system classifier. The obtained classification accuracy of the system was 89.47 percent (Polat & Gunes 2006).

Most of them work on client-side and carrying all advantages-disadvantages of 2-tier architecture.

Consequently, there couldn't be found any expert system which satisfying all needed below:

i. A web-based system for easy upgrade, management and maintenance.
ii. An AI engine developed by an AI language.
iii. A system having purpose for diabetes diagnosis and treatment.
iv. Integration to 3rd Party solutions for data mining and statistics.

Since there is non-existence of any expert system that satisfies all requirements mentioned above, we have developed a web based diabetes expert system supported by ANFIS and Weka (http://www.cs.waikato.ac.nz/ml/weka 2008).

2. STRUCTURE OF THE EXPERT SYSTEM

The Diabetes Expert System is fully integrated and has a flexible architecture so that runs on different operating systems such as MS Windows, Linux etc. It's obliged its flexibility to the architecture defined below.

2.1 The System Architecture

The Diabetes Expert System is originated by multi sub-systems. An expert system was aimed to run on web environment, in order users to reach the system on Internet. Architecture of the diabetes expert system is illustrated on Figure 2.1 below.

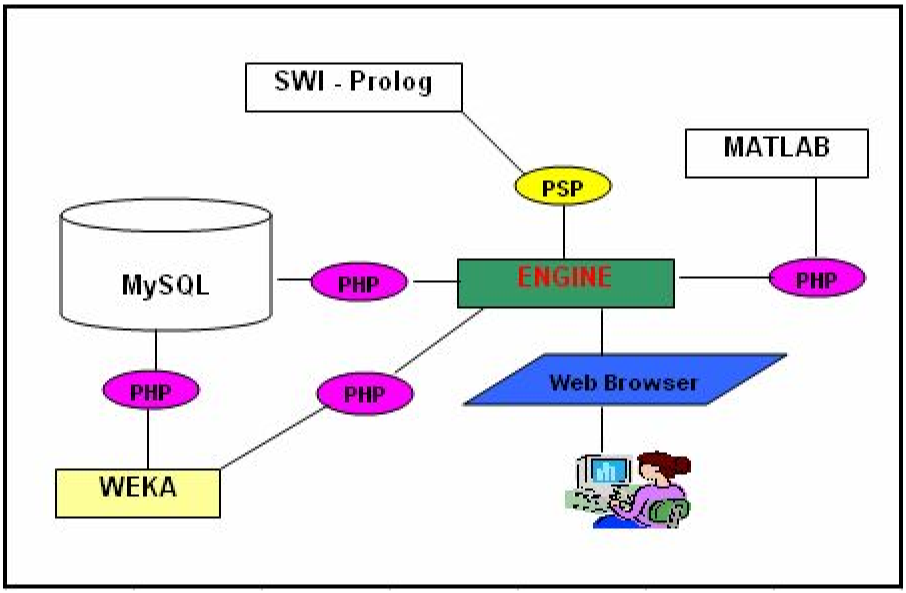

Figure 2.1 : Architecture of the system

We benefited from Personal Home Page (PHP) (http://www.php.net 2008), which is a web programming language. Because PHP has a lot of functions and a large library in order to install bridge between the system and the 3rd party solutions. JavaScript helped the PHP while forming the user interface.

Apache was installed (http://httpd.apache.org 2008) as a web server, because Apache is free and saves clear Access Log that is useful for web mining.

If we aim to develop an AI engine, we wouldn't have lots of alternatives to use AI language. Prolog or LISP could be chosen, and the choice was Prolog. Because Prolog has a web version using SWI-Prolog engine, which was developed by University of Amsterdam and has GNU Public Licence (http://www.swi-prolog.org 2008), and it supplies high easiness to developers who aim to develop a web based expert system. The name of web version is PSP (http://www.prologonlinereference.org/psp.psp 2007) and it's developed by Benjamin Johnston from University of Technology, Sydney.

In order to save users' data, we needed an RDBMS. The database server should be relational and have interface to web services. Then we determined to use MySQL (http://www.mysql.com 2008), because it has all features we need, and is totally free as well.

After developing the kernel of the system, we needed to develop a bridge to reach data mining solutions such as Weka and Matlab (http://www.mathworks.com 2008). All the connections between kernel of the diabetes expert system and the 3rd parties were developed by PHP system functions. The system components are summarized in Table 2.1 below.

Table 2.1 : System components

Kernel of Expert System	SWI-Prolog v5.6.30 + Prolog Server Pages v0.4
Scripting	PHP v5.1.2 and JavaScript
Web Server	Apache v2.0.61
Database Server	MySQL v5.0.19-nt
3rd Party Solutions	Weka v3.5.6, MATLAB v7.1

2.2 Structure of the Database

As mentioned above, it's needed to use a relational database to save social-demographic and personal data of the users. Tables of the database, the fields and their types are shown in Figure 2.2 below.

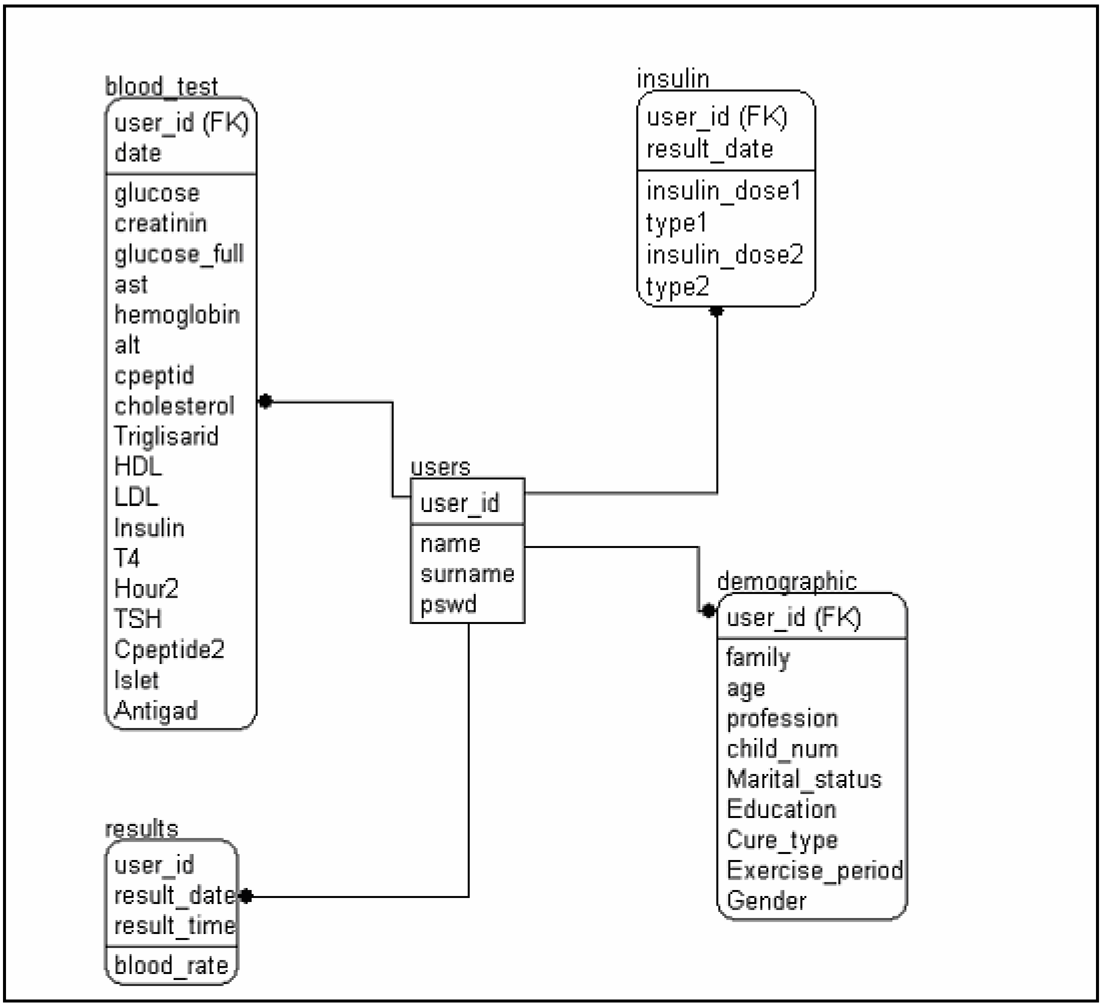

Figure 2.2 : Entity relationship diagram of the database

Tables and purpose of their creation are explained below.

Users:

Table Users saves primary data of all users, such as names, IDs and passwords. Primary Key is users in Users.

Insulin:

Diabetic patients daily need to save insulin type and dosage in table Insulin. Primary keys are user_id and result_date.

Results:

The system uses daily glucose rates of diabetic patients by saving results of the measurements in table Results. Primary keys are user_id, result_date and result_time.

Blood_Test:

The patients measure blood values per 3 or 4 months. They also need to measure blood value in initial procedure of treatment. Table blood_test saves these values. Primary keys are user_id and date.

Demographic:

Social-demographic data of the diabetic patients are saved in table Demographic to be used in ANFIS module. Primary key is user_id of the table.

3. A WEB-BASED EXPERT SYSTEM FOR DIABETES DIAGNOSIS

The expert systems are a branch of applied artificial intelligence (AI) and were developed by the AI community in the mind-1960s. The basic idea of the development expert systems is that expertise of domain expert is transferred to a computer. The computer (diabetes expert system) can make inferences and arrive at a specific conclusion (Nammuni et al. 2004, pp.121-129).

The diagnosis of diabetes is an important problem in the medical science. Therefore, the design an automatic control system of diagnosis diabetes with the use of expert system would be a very basic tool in the service of doctor.

3.1 Glucose & Insulin Rate Tracking

The main data collector suite of the web-based expert system is Glucose and Insulin Tracker. Any user who uses the system can save and retrieve the daily and periodical rates of glucose and insulin affecting it. Figure 3.1 summarizes usage of the module.

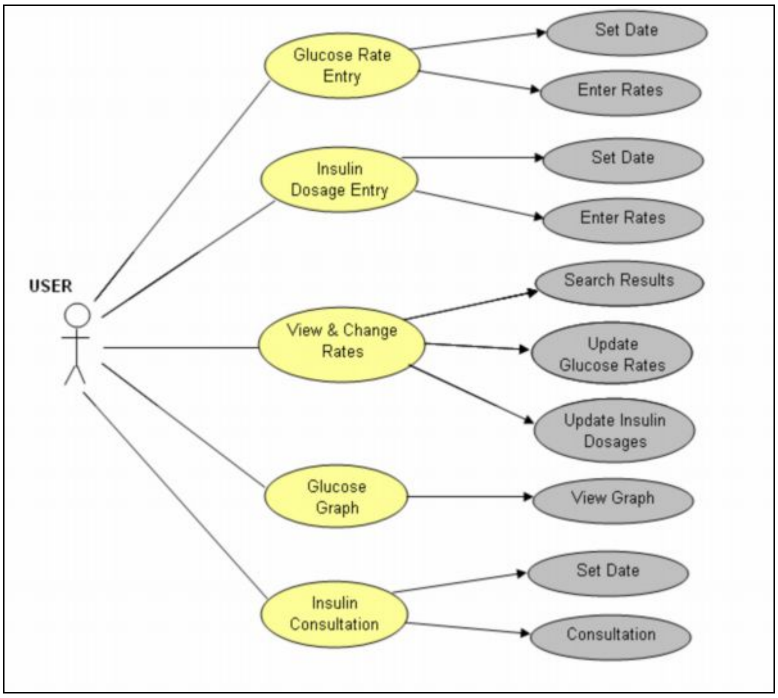

Figure 3.1 : Use case diagram of glucose & insulin system

Glucose Rate Entry

Diabetic patient periodically measures their glucose levels 3 or 4 times per day. He writes down the rates, and then asks a physician how much insulin to be injected in next day in order to control glucose level.

The Diabetic Expert System has a module to save daily glucose rates. (It can be seen in Figure 3.2)

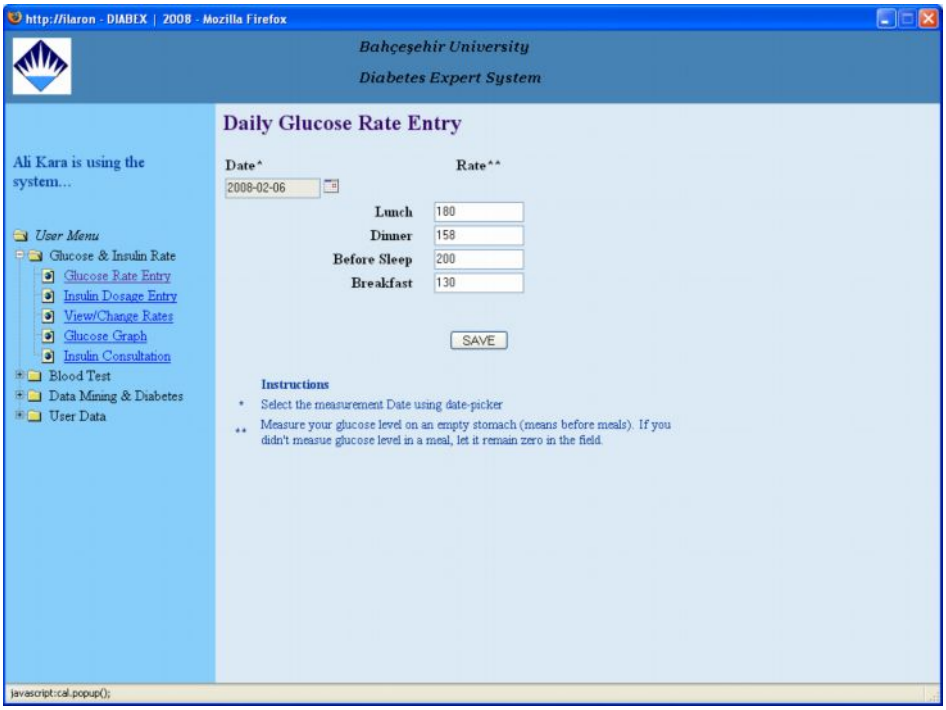

Figure 3.2 : Daily glucose rate entry

Patient (or user of the system) enters glucose rates, as frequent as he measures. Periods are:

i. Lunch
ii. Dinner
iii. Before Sleep
iv. Breakfast

The expert system saves daily results in order to show information or supply data for data mining in future.

Insulin Dosage Entry

User generally injects insulin 1 or 2 times per day. The insulin types differ according to affectivity duration, strength etc. The Diabetes Expert System contains 3 insulin classes:

1. Rapid-Acting is strong and effective for 2 hours (e.g. Humalog/lispro, Novorapid/aspart, Humulin-R)
2. Long-Acting is effective longer than Rapid (almost 8-10 hours)(e.g. Humulin-U, Ultralente, Lantus)
3. Pre-Mixed is a mixture of Intermediate and Short acting insulin. (e.g. Humalog Mix25, Humulin-N/L, Novolin-ge, Humalog/Mix25, Novolog)

Figure 3.3 : Daily insulin rate entry

User selects insulin type and enters how many units he daily injects. (It can be seen in Figure 3.3)

View & Changes Rates

User would like to see and change daily insulin rates and glucose dosages. Then the Diabetes Expert System supplies change glucose rate and/or insulin dosage and type. (It can be seen in Figure 3.4 and Figure 3.5)

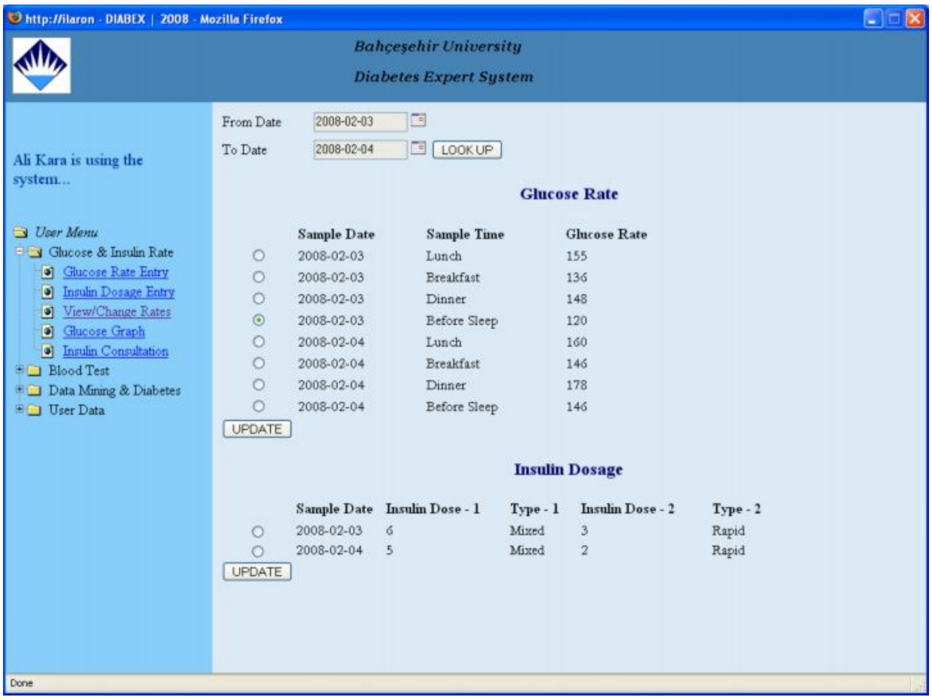

Figure 3.4 : Daily glucose rate update

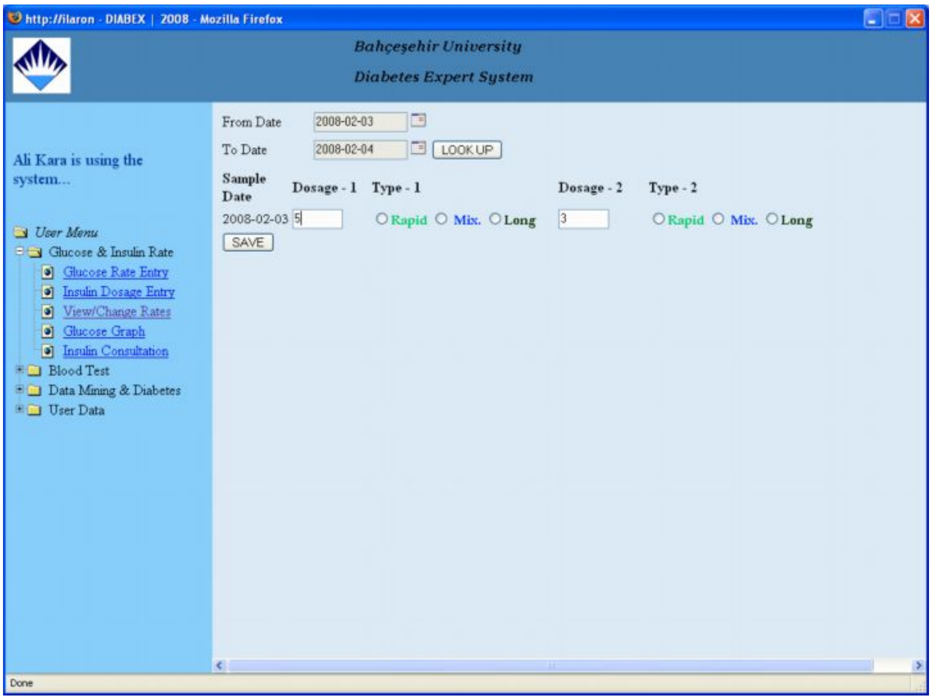

Figure 3.5 : Daily insulin rate update

Glucose Graph

A patient who uses The Diabetes Expert System would like to see graph of glucose level history. The Diabetes Expert System supplies a graph that illustrates daily glucose levels depends on user glucose entries. (It can be seen in Figure 3.6)

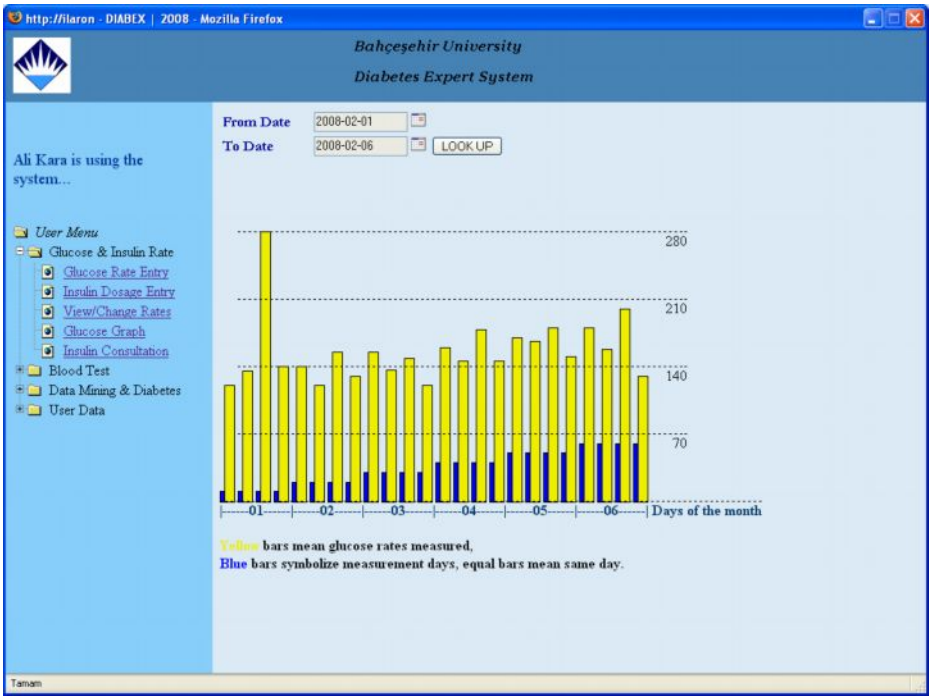

Figure 3.6 : Glucose graph

Physicians or users reach and examine the level of glucose and when it reaches the critical border.

3.2 Periodical Blood Tests

Result of periodical blood tests is a useful parameter supporting decision of physician. He anyway applies the result of periodical blood test in order to examine situation of diabetes. Figure 3.7 summarizes usage of the module.

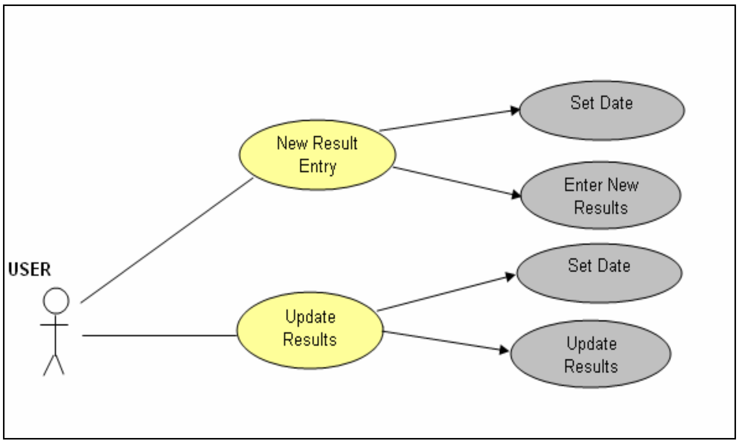

Figure 3.7 : Use case diagram of blood test

New Result Entry

Main parameters effecting the decision of physician are

i. Glucose
ii. Insulin
iii. Insulin (2 Hour)
iv. Triglisarid
v. Cholesterol (HDL – LDL and Total)
vi. C Peptide

The diabetes expert system supplies a module to save the results and physician can examine the results periodically. (It can be seen in Figure 3.8)

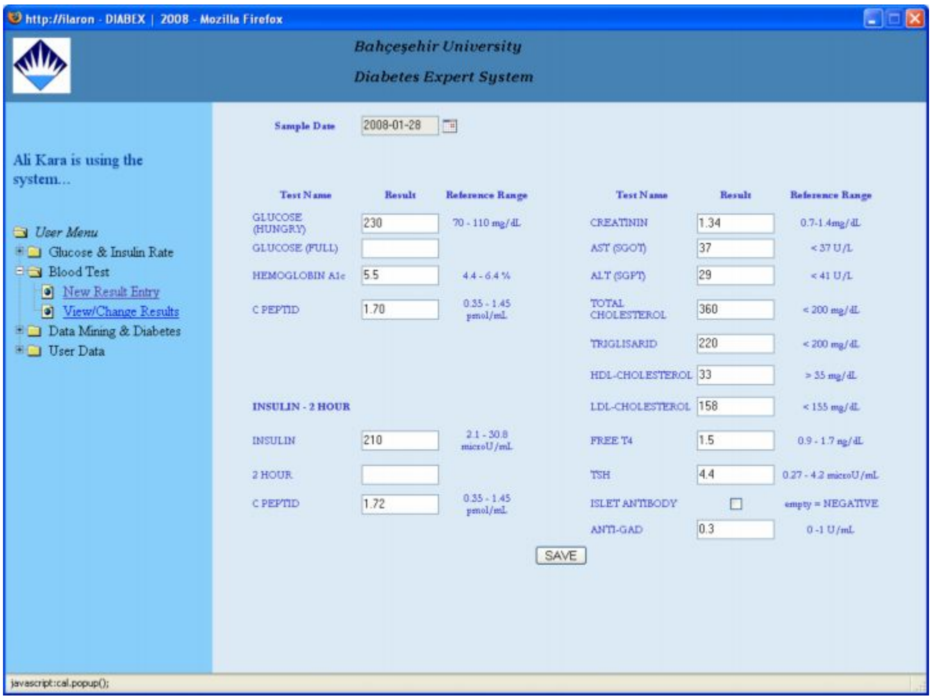

Figure 3.8 : Periodical blood test

User may make mistake while entering the results. Then he is able to find and change the result in the Diabetes Expert System.

3.3 Insulin Consultation

Physicians examine how much insulin need to be injected to the patient in early 7 or 10 days of medical treatment. They use glucose rates and offer to increase or decrease insulin unit.

The Diabetes Expert System, which is a diabetes expert system, also consults insulin dosage that will be injected the day after.

The consultation process works so:

i. Patient measure daily glucose rates, and then enters them into the Diabetes Expert System.

ii. Patient also enters insulin type and unit that is suggested by physician as starting.
iii. Insulin Consultation module examines insulin type and units, and also the glucose rates. Then the system suggests to increase or decrease insulin unit.

The result page behaves as a physician and mixes medical experiences with trial and error method.

After 7 or 10 days, the Diabetes Expert System reaches the best insulin dosage for the patient. (It can be seen in Figure 3.9)

Figure 3.9 : Insulin consultation

3.4 User Data & Diabetes Oracle

One of the main purposes of the thesis is to reveal diabetes map of Turkey and find association rules among the social-demographic data of the users, which are also diabetic patients. To maintain the purpose, the system needs users' social-demographic data. Figure 3.10 summarizes usage of the module.

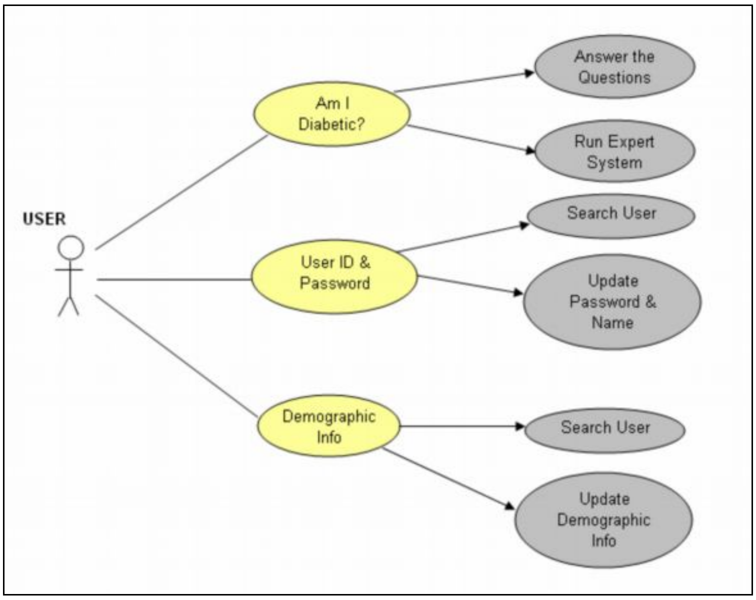

Figure 3.10 : Use case diagram of user data

User Data

During the signing-up process, every user must enter his social-demographic data. The data contains:

i. Age
ii. Gender
iii. Family type
iv. Profession
v. Education
vi. Cure Type
vii. Marital Status
viii. Exercise Period
ix. Child Number

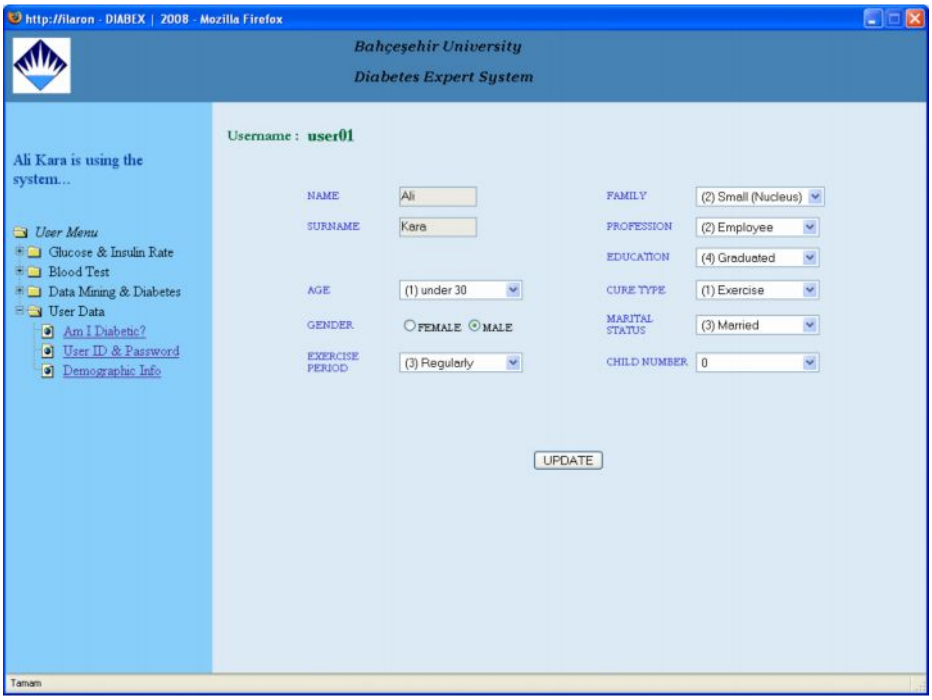

Figure 3.11 : Social-demographic data

As it can be seen in Figure 3.11, any user is able to update his current social-demographic data. The data will be used in data mining techniques, which will be mentioned in the section 4.

Diabetes Oracle

The Diabetes Expert System is designed not only for diabetic patients, but also for the people who suppose that they are healthy. If a person suspects that it has risk of diabetes, the Diabetes Expert System consults the answers from him and responds a result containing the BMI (Body Mass Index) and diabetes risk.

The parameters are information such as height, weight etc. and some questions shown in Figure 3.12.

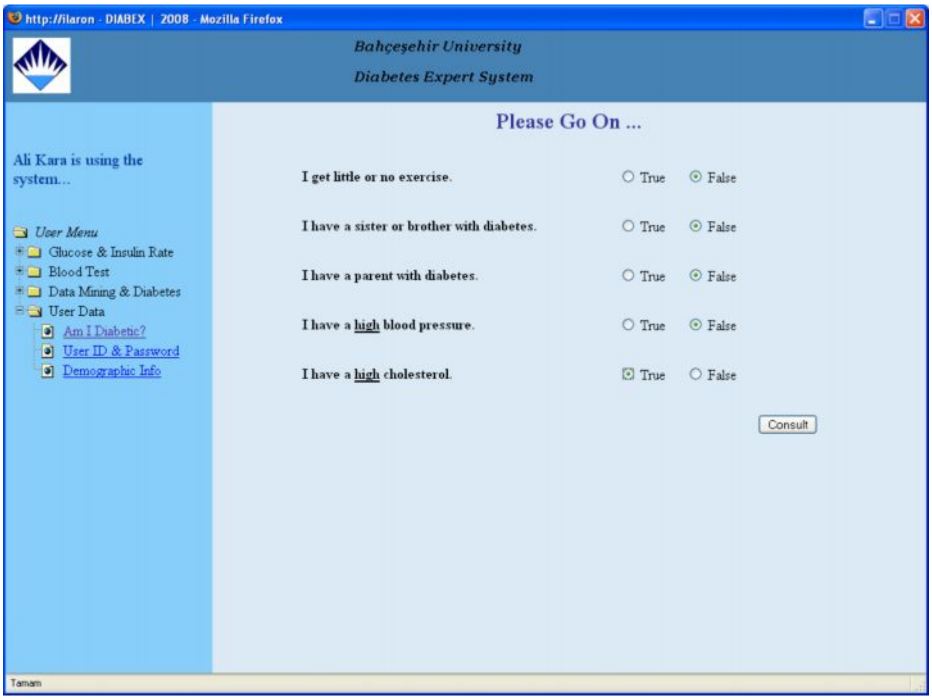

Figure 3.12 : Diabetes oracle

Result of consultation determines the diabetes risk before the patient being diabetic. That's why we called it oracle. (it can be seen in Figure 3.13)

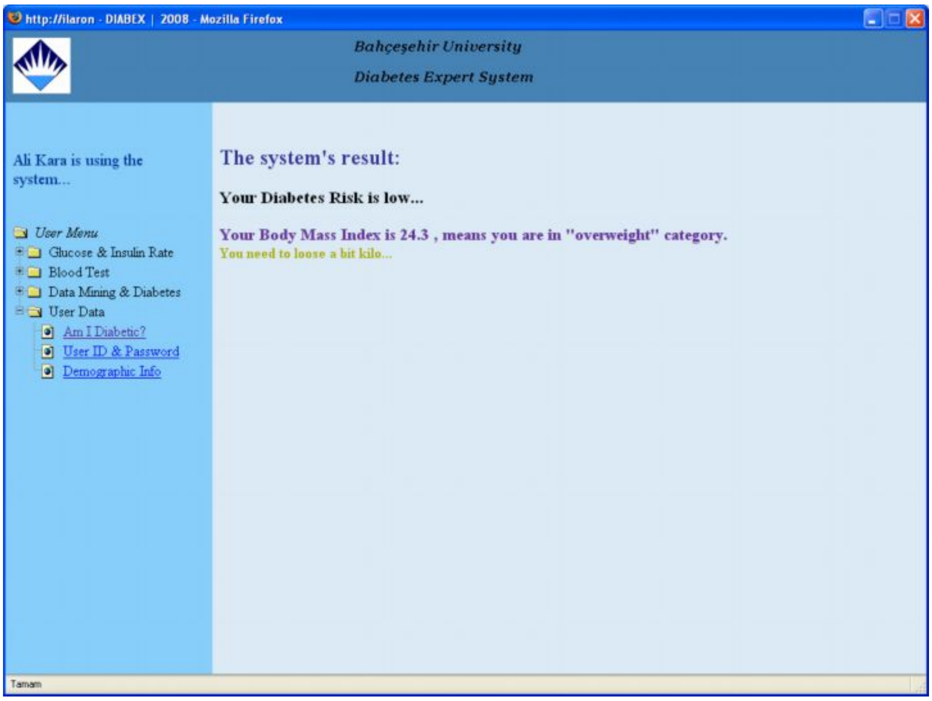

Figure 3.13 : Diabetes oracle results

4. DATA MINING IN THE EXPERT SYSTEM

Data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

In our thesis, some data mining methods on social-demographic data of the users were applied. Correction and actuality of the data is very important for data mining for diabetes.

4.1 Data Mining by WEKA Engine

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

Weka is developed by the University of Waikato. In our thesis we used the Weka as data mining engine, and made a bridge between the Diabetes Expert System interface and Weka. No user needs to install Weka in his workstation, but it do already enough to be installed on server machine. In addition to this, the detail information is given in the section 2.

Refreshing Dataset

Weka can directly reach the database and fetch the data from there. Or we need to prepare a dataset to be processed by Weka engine. We chose the second way, because we don’t need the whole data (all columns of the table), and it’s better to prepare an ARFF file, which is processed by Weka.

The Diabetes Expert System shows inclusion of the ARFF file that is prepared, after choosing some attributes of social-demographic data or all of them. (It can be seen in Figure 4.1)

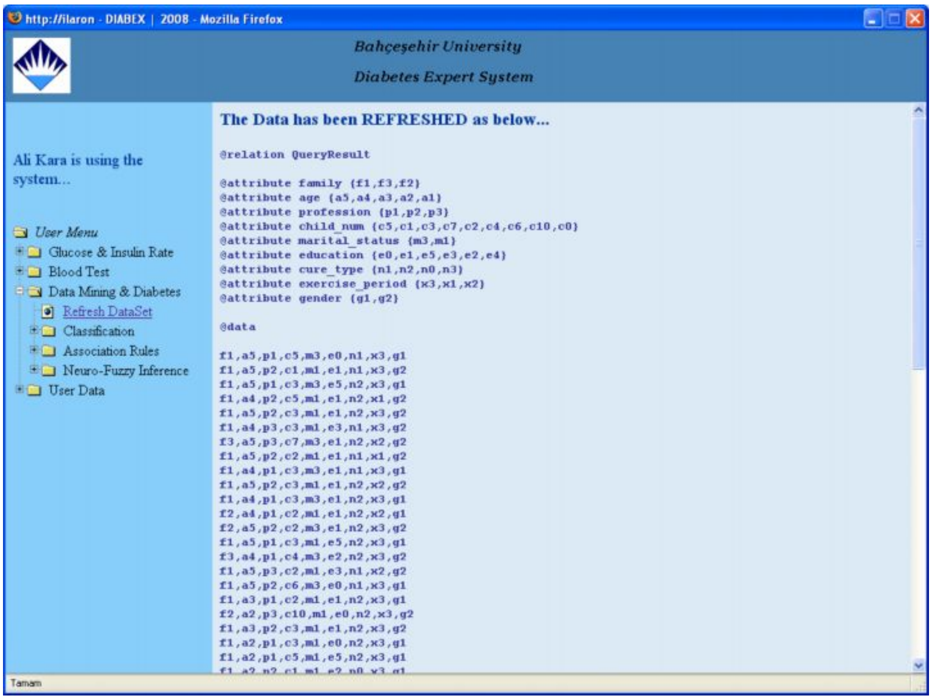

Figure 4.1 : Refreshing dataset

4.1.1 Classification

Classification is a data mining (machine learning) technique used to predict group membership for data instances. For example, you may wish to use classification to predict whether the weather on a particular day will be “sunny”, “rainy” or “cloudy”. Popular classification techniques include decision trees and neural networks.

Bayes Network Classifier

Bayes Network learning uses various searching algorithms and quality measures. This is the base module for a Bayes Network Classifier, and also provides data structures (network structure, conditional probability distributions, etc.) and facilities common to Bayes Network learning algorithms like K2 and B.

When we applied Bayes Network onto refreshed data, we got the results shown as below on Figure 4.2

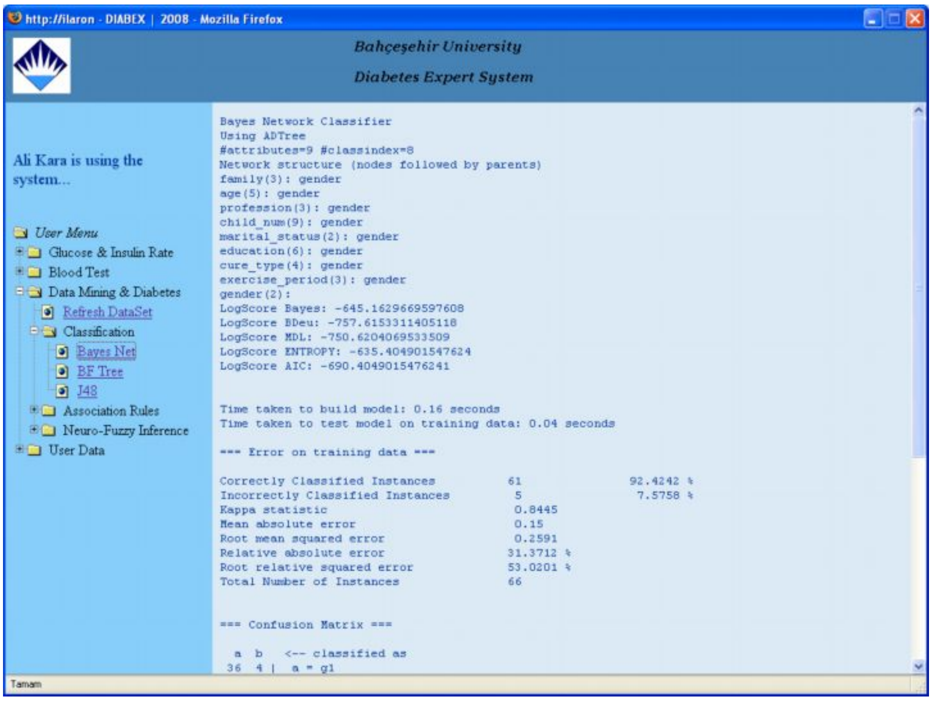

Figure 4.2 : Bayes network classifier

BF Tree Decision Tree

This is a module for building a best-first decision tree classifier. This class uses binary split for both nominal and numeric attributes. For missing values, the method of 'fractional' instances is used.

When we applied BF Tree onto refreshed data, we got the results shown as below on Figure 4.3.

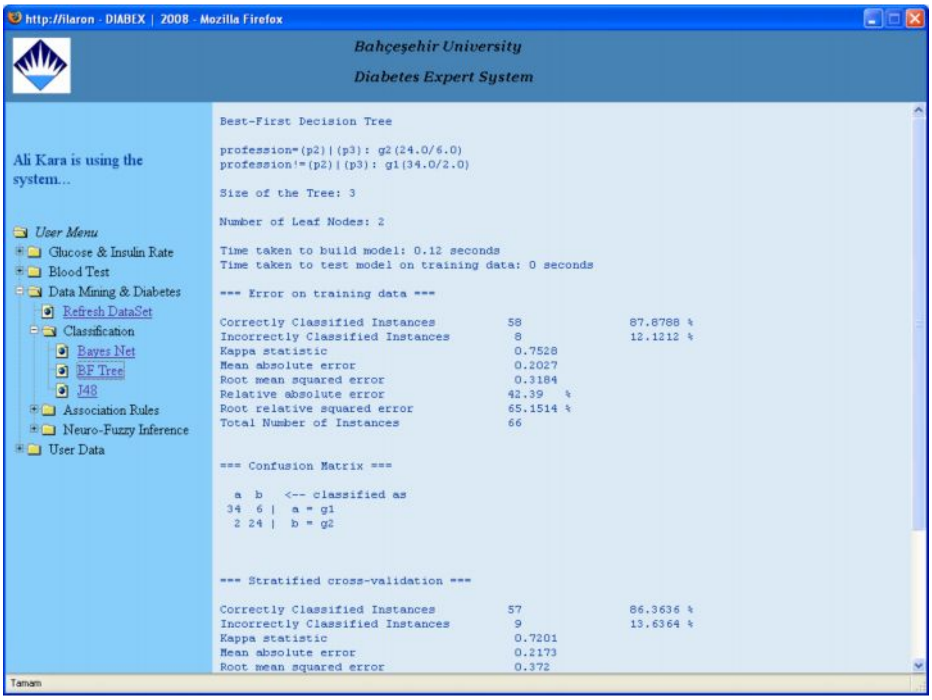

Figure 4.3 : Best-first decision tree

J48 Pruned Tree

J48 is a module for generating a pruned or unpruned C4.5 decision tree. When we applied J48 onto refreshed data, we got the results shown as below on Figure 4.4.

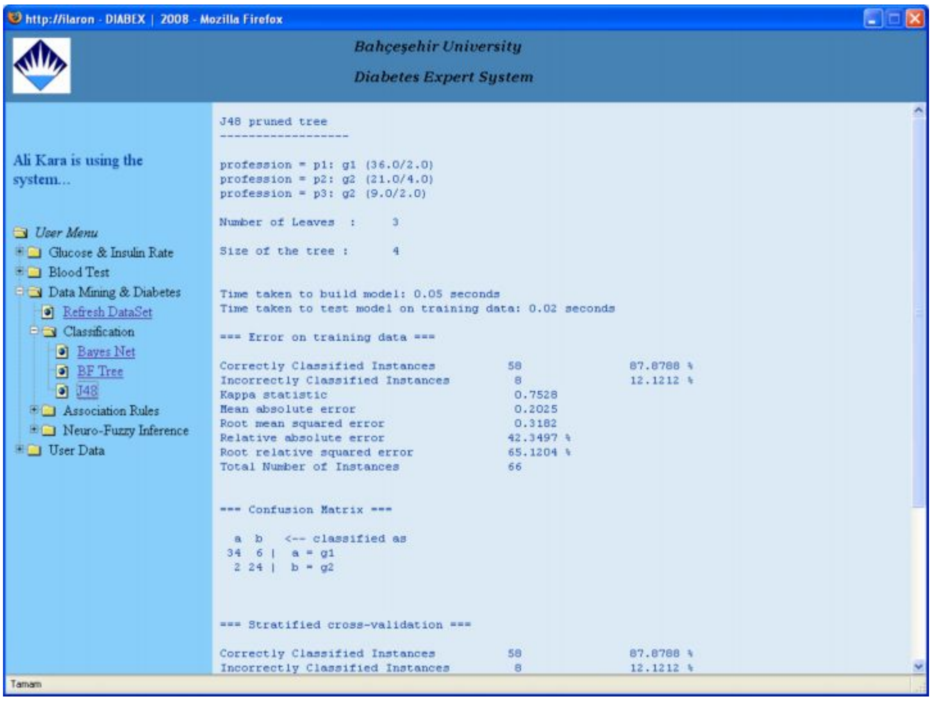

Figure 4.4 : J48 pruned tree

4.1.2 Association Rules

One of the reasons behind maintaining any database is to enable the user to find interesting patterns and trends in the data. For example, in a supermarket, the user can figure out which items are being sold most frequently. But this is not the only type of `trend' which one can possibly think of. The goal of database mining is to automate this process of finding interesting patterns and trends. Once this information is available, we can perhaps get rid of the original database. The output of the data-mining process should be a "summary" of the database. This goal is difficult to achieve due to the vagueness associated with the term `interesting'. The solution is to define various types of trends and to look for only those trends in the database. One such type constitutes the association rule (Edward & Omiecinski 2003, pp. 57-69).

In the rest of the discussion, we shall assume the supermarket example, where each record or ordered list consists of the items of a single purchase. However the concepts are applicable in a large number of situations.

In the present context, an association rule tells us about the association between two or more items. For example: In 80 percent of the cases when people buy bread, they also buy milk. This tells us of the association between bread and milk. We represent it as

Bread => milk | 80 percent

This should be read as - "Bread means or implies milk, 80 percent of the time." Here 80 percent is the "confidence factor" of the rule.

Association rules can be between more than 2 items. For example -

Bread, milk => jam | 60 percent

Bread => milk, jam | 40 percent

Given any rule, we can easily find its confidence. For example, for the rule

Bread, milk => jam

We count the number say n_1, of records that contain bread and milk. Of these, how many contain jam as well? Let this be n_2. Then required confidence is n_2/n_1.

This means that the user has to guess which rule is interesting and ask for its confidence. But our goal was to "automatically" find all interesting rules. This is going to be difficult because the database is bound to be very large. We might have to go through the entire database many times to find all interesting rules.

Apriori

Apriori is a module implementing an Apriori-type algorithm. Iteratively reduces the minimum support until it finds the required number of rules with the given minimum confidence. The algorithm has an option to mine class association rules. It is adapted as explained in the second reference.

Apriori is a classic algorithm for learning association rules. Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions (Winepi and Minepi), or having no timestamps (DNA sequencing) (Agrawal et al. 1993, pp. 207-16).

As is common in association rule mining, given a set of itemsets (for instance, sets of retail transactions, each listing individual items purchased), the algorithm attempts to find subsets which are common to at least a minimum number C (the cutoff, or confidence threshold) of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found (Agrawal et al. 1994).

When we applied Apriori onto refreshed data, we got the results shown as below on Figure 4.5.

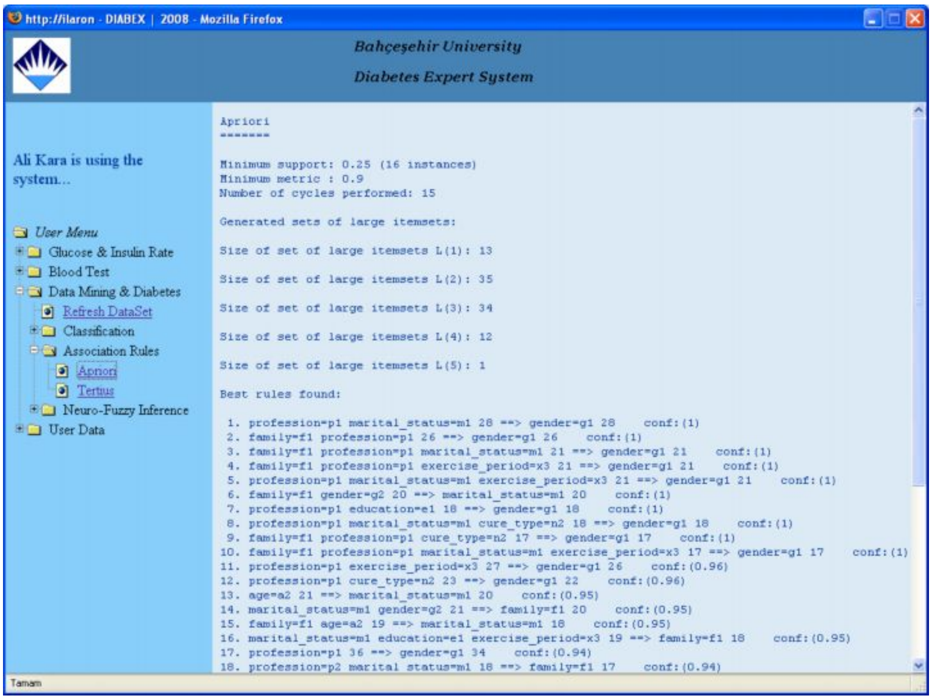

Figure 4.5 : Apriori results

Tertius

This finds rules according to confirmation measure (Tertius-type algorithm). The Tertius system implements a top-down rule discovery system employing the confirmation measure. Tertius uses a first-order logic representation. Such a representation allows it to deal with several kinds of data and, moreover, allows the user to choose the most convenient or the most comprehensible representation among several possible representations (Flach & Lachiche 1999, pp.61-95).

When we applied Tertius onto refreshed data, we got the results shown as below on Figure 4.6.

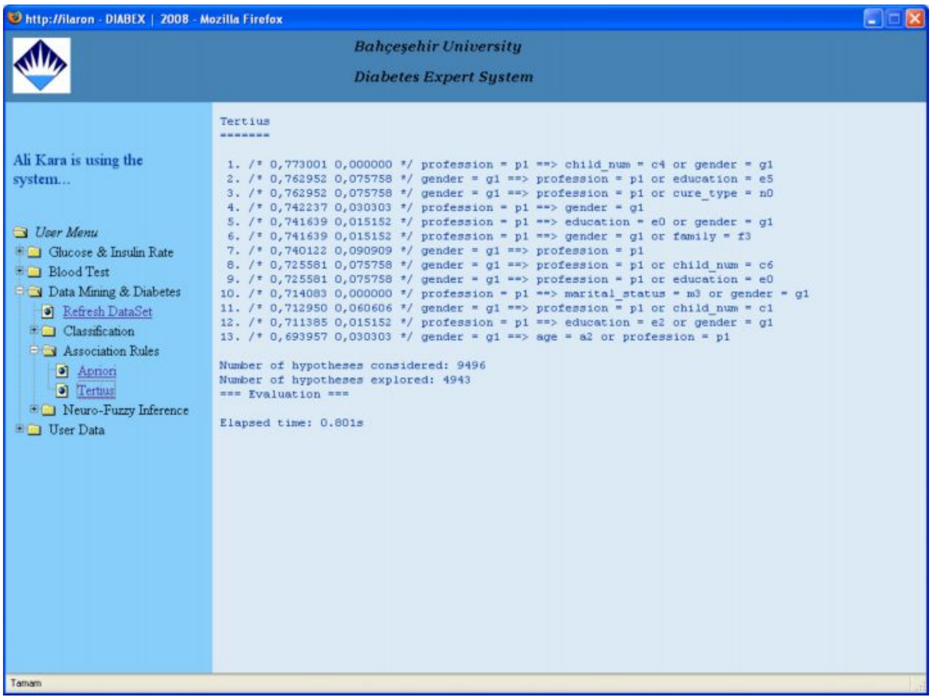

Figure 4.6 : Tertius results

4.1.3 Benchmarking of Association Rule Methods

2 association rule methods (Apriori and Tertius) were applied onto social-demographic data of diabetic patients, and then found useful relations between attributes of social-demographic data.

Before processing Weka for association rules, it's useful to refresh dataset and choose 3 attributes instead of the all. In our experiment we have 66 instances to be processed. And our attributes are **profession, age** and **family.**

Apriori results are shown in Figure 4.7 and Tertius results are shown in Figure 4.8.

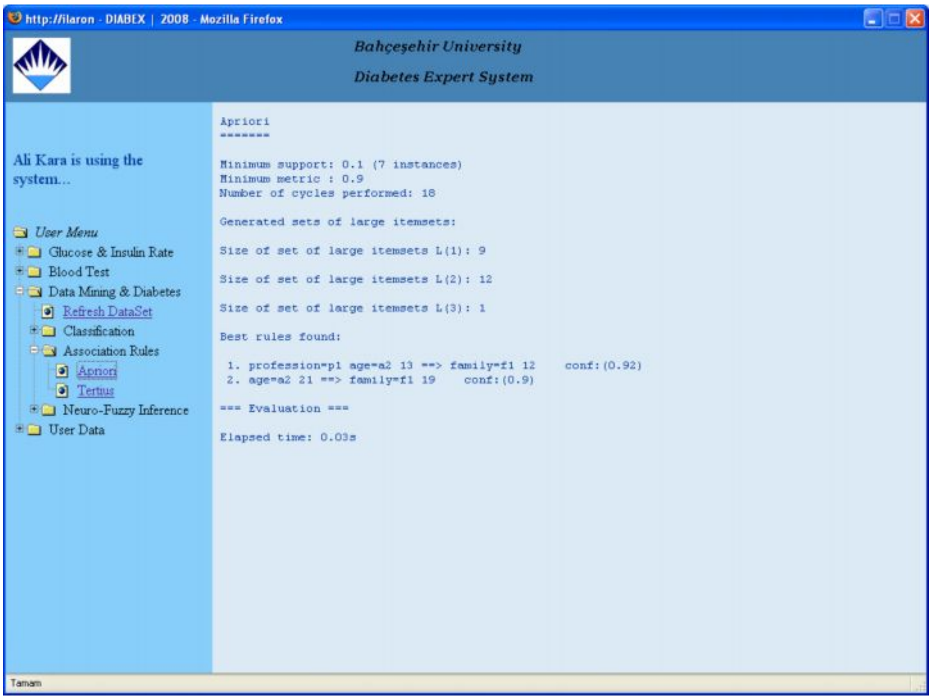

Figure 4.7 : Apriori results for 3 attributes

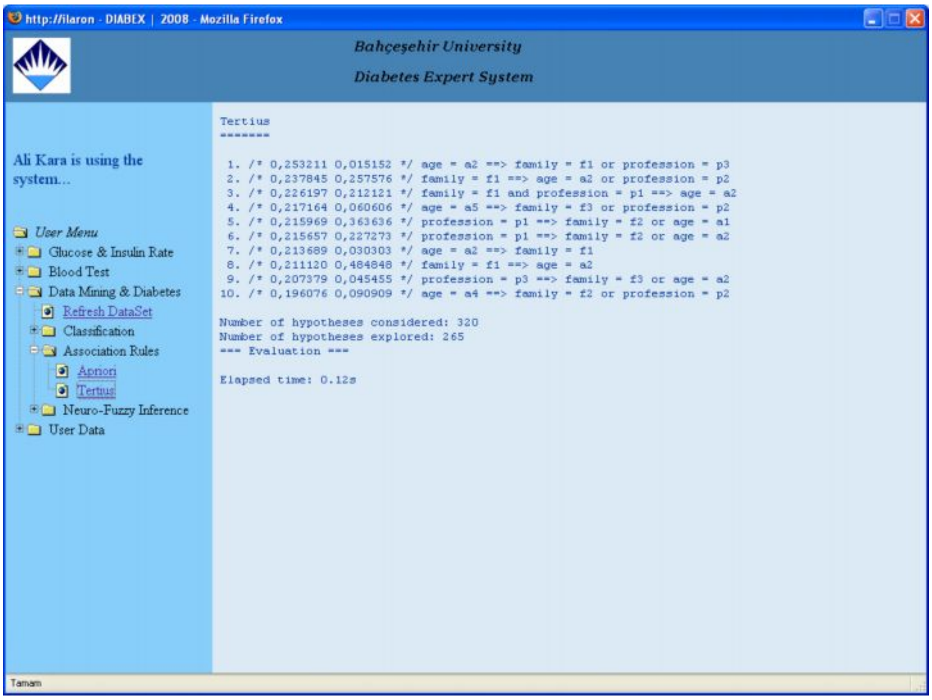

Figure 4.8 : Tertius results for 3 attributes

4.2 ANFIS vs. Logistic Regression

Artificial Neural Networks (ANNs) and Fuzzy Logic (FL) have been increasingly in use in many engineering fields since their introduction as mathematical aids by McCulloch and Pitts, 1943, and Zadeh, 1965, respectively. Being branches of Artificial Intelligence (AI), both emulate the human way of using past experiences, adapting itself accordingly and generalizing. While the former have the capability of learning by means of parallel connected units, called neurons, which process inputs in accordance with their adaptable weights usually in a recursive manner for approximation; the latter can handle imperfect information through linguistic variables, which are arguments of their corresponding membership functions.

Although the fundamentals of ANNs and FL go back as early as 1940s and 1960s, respectively, significant advancements in applications took place around 1980s. After the introduction of back-propagation algorithm for training multi-layer networks by Rumelhart and McClelland, 1986, ANNs has found many applications in numerous inter-disciplinary areas (Patterson 1994; Rumelhart & McCelland 1986; McCelland & other 1986, pp.216-271). On the other hand, FL made a great advance in the mid 1970s with some successful results of laboratory experiments by Mamdani and Assilian (1975, pp.1-13). In 1985, Takagi and Sugeno (1985, pp.116-132) contributed FL with a new rule-based modeling technique. Operating with linguistic expressions, fuzzy logic can use the experiences of a human expert and also compensate for inadequate and uncertain knowledge about the system. On the other hand, ANNs have proven superior learning and generalizing capabilities even on completely unknown systems that can only be described by its input-output characteristics. By combining these features, more versatile and robust models, called “neuro-fuzzy” architectures have been developed (Culliere et al. 1995, pp.2009-2016).

In a control system the plant displaying nonlinearities has to be described accurately in order to design an effective controller. In obtaining the model, the designer has to follow one of two ways. The first one is using the knowledge of physics, chemistry, biology and the other sciences to describe an equation of motion with Newton’s laws, or electric circuits and motors with Ohm’s, Kirchhoff’ s or Lentz’s laws depending on the plant of interest. This is generally referred to as mathematical modeling. The second way requires the experimental data obtained by exciting the plant, and measuring its response. This is called system identification and is preferred in the cases where the plant or process involves extremely complex physical phenomena or exhibits strong nonlinearities.

Conventional control methods rely upon strong mathematical modeling, analysis, and synthesis. In case where mathematical models are available conventional control theory acts as a powerful tool for controlling even complex systems. On the other hand obtaining a mathematical model for a system can be rather complex and time consuming as it often requires some assumptions such as defining an operating point and doing linearization about that point and ignoring some system parameters, etc. This fact has recently led the researchers to exploit the neural and fuzzy techniques in modeling and control of complex systems.

Although fuzzy logic allows one to model (control) a system using human knowledge and experience with if-then rules, it is not always adequate on its own. This is also true for ANNs, which only deal with numbers rather than linguistic expressions. This deficiency can be overcome by combining the superior features of the two methods, as is performed in ANFIS architecture introduced by Jang (1993, pp.665-685). ANFIS architecture which was used in here as the controller of the dynamic system is generally encountered in the areas of function approximation, fault detection, medical diagnosis and control, (Gonzalez-Andujar et al. 2006, pp.115-123; Turner et al. 2006; Kim et al. 2006).

Adaptive Neuro Fuzzy Inference System was used, as an estimation method which has fuzzy input and output parameters. Then standard errors of ANFIS were benchmarked with Multinomial Logistic Regression as a non-linear regression method, and it’s seen that ANFIS is more efficient than Multinomial Logistic Regression with fuzzy diabetes dataset.

Material & Method

After researching ANFIS theoretically, glucose rates and affecting factors to glucose rate were evaluated with ANFIS. As a comparison, estimation results and standard error percentages of ANFIS were compared with Multinomial Logistic Regression (MLR).

Having searched the literature, it's seen that works were dealing with binary results (1=healthy, 0=diabetic). In this work it's desired to make a step beyond, and worked on fuzzy dependent variable.

The dataset that was used in the thesis consist of 4 variables of 470 subjects who were interviewed in a clinic. All subjects are known as diabetic and all of them are under diabetes treatment. In our work it's tried to find any relation between diabetes risk and age, gender, total cholesterol and a ratio that is called frame. The waist/hip ratio (Frame) may be a predictor in diabetes.

The dependent variable in the dataset is *Glucose rate* and independent variables are:

i. Age
ii. Gender
iii. Frame (Waist/Hip ratio)
iv. Total Cholesterol

Some variables of the dataset had already fuzzy values, such as frame. However some of them didn't have fuzzy values such as age, glucose. Variables of the dataset used in the thesis are shown in Table 4.1.

Table 4.1 : Variables and descriptions of dataset

Name	Description
Glucose	Diabetes indicator, glucose level in blood (Dependent variable)
Age	Age of diabetic patients
Gender	Gender of diabetic patients
Frame	Waist / hip ratio of diabetic patients.
Cholesterol	Total Cholesterol values of diabetic patients

The main purpose of preprocessing data was to make fuzzy the variable in order to use them in ANFIS as a fuzzy inference system. Variables and their fuzzy values are shown in Table 4.2.

Table 4.2 : Fuzzy values of variables used

Age (1)	Gender (2)	Frame (3)	Cholesterol (4)	Glucose (Out)
1- (0-24)	1-Male	1-Small	1- (<200)	1- (<60)
2- (25-49)	2-Female	2-Medium	2- (201-240)	2- (60-89)
3- (50-74)		3-Large	3- (over 240)	3- (90-120)
4- (75-99)				4- (121-300)
5- (over 99)				5- (over 300)

Since the purpose is same the other estimation method, and it's to define a relationship between dependent variable and independent variables by using minimum variables; then it's needed to decrease the number of inputs to the least meaningful number.

The function *exhsrch* in MATLAB performs an exhaustive search within the available inputs to select the set of inputs that most influence the diabetes diagnosis. The first parameter to the function specifies the number of input combinations to be tried during the search.

Table 4.3 : Results of preprocessing

INPUT			Train Error	Check Error
1	2	3	0.7154	0.7921
1	2	4	0.7139	0.7940
1	3	4	0.7122	0.8033
2	3	4	0.7346	0.7974

The results that are shown in Table 4.3 indicate that it's better to make a set of Input 1, 3 and 4. Because all other alternative sets of inputs have more standard error of training data. The purpose is to have estimation with the least standard error, so it's better to use the set with the least error. Essentially, *exhsrch* builds an ANFIS model for each combination and trains it for one epoch and reports the performance achieved.

Adaptive Neuro Fuzzy Inference System (ANFIS) as an Estimation Method

As a neural-fuzzy system, ANFIS is a combination of neural networks and fuzzy systems in such a way that neural networks or neural networks algorithms are used to determine parameters of fuzzy system. This means that the main intention of neural-fuzzy approach is to create or improve a fuzzy system automatically by means of neural network methods. Adaptive neuro fuzzy inference system basically has 5 layer architectures and each of the function is explained in detail below (Sojda 2007; Seising 2006):

Layer 1 Every mode in this layer is an adaptive node with a node function where x (or y) is the input to node I and A_i (or B_{i-2}) is a linguistic label and Oi^1 is the membership grade of fuzzy set A (= A_1, A_2, B_1 or B_2) and it specifies the degree to which the given input x (or y) satisfies the quantifier A. The membership function for A can be parameterized membership function as given in equation 1 or normally known as Bell function and {a_i, b_i, c_i} is the parameter set

$$\mu_{Ai}(x)=\frac{1}{1+\left\{((x-c_i)/a_i)^2\right\}^{b_i}} \quad , \qquad (1)$$

$$O_i^1 = \mu_{Ai}(x), \quad \mathrm{i} = 1,2,$$
$$O_i^1 = \mu_{Bi-2}(y), \ \mathrm{i} = 3,4, \tag{2}$$

Layer 2 Every node in this layer is a fixed node labeled M, whose output is the product of all the incoming signals each node output represents the firing strength of a rule.

$$O_i^2 = \omega_i = \mu_{Ai}(x)_{Bi}(y), \quad \mathrm{i} = 1,2, \tag{3}$$

Layer 3 Every node in this layer is a fixed node labeled N. The ith node calculates the ratio of the ith rule's firing strength to the sum of all rules' firing strengths. Outputs of this layer are called normalized

$$O_i^3 = \varpi_i = \frac{\omega_i}{\omega_1 + \omega_2}, \qquad \mathrm{i} = 1,2, \tag{4}$$

Layer 4 Every node I in this layer is an adaptive node with a node function. Where wi is a normalized firing strength from layer 3 and {*pi, qi, ri*} is the parameter set of this node. Parameters in this layer are referred to as consequent parameters.

$$O_i^4 = \varpi_i f_i = \varpi_i(\mathrm{p_i x + q_i y + r_i}), \qquad \mathrm{i} = 1,2, \tag{5}$$

Layer 5 The single node in this layer is a fixed node labeled ∑, which computes the overall output as the summation of all incoming signals. Overall output:

$$O_i^4 = \sum_{i=1}^{2} \varpi_i f_i = \frac{\sum_{i=1}^{2} \omega_i f_i}{\omega_1 + \omega_2}, \tag{6}$$

For simplicity, we assume that the fuzzy inference system under consideration has two input x and y and one output z. For a first-order Sugeno fuzzy model, a common rule set with two fuzzy if-then rules is the following:

Rule 1: If x is A_1 and y is B_1, the f1 = p1 x + q1 y + r1,
Rule 2: If x is A_2 and y is B_2, the f2 = p2 x + q2 y + r2.

The corresponding equivalent ANFIS architecture is as shown in Figure 4.9, where nodes of the same layer have similar functions. ANFIS has hybrid learning capability which compromised of back propagation and least square method.

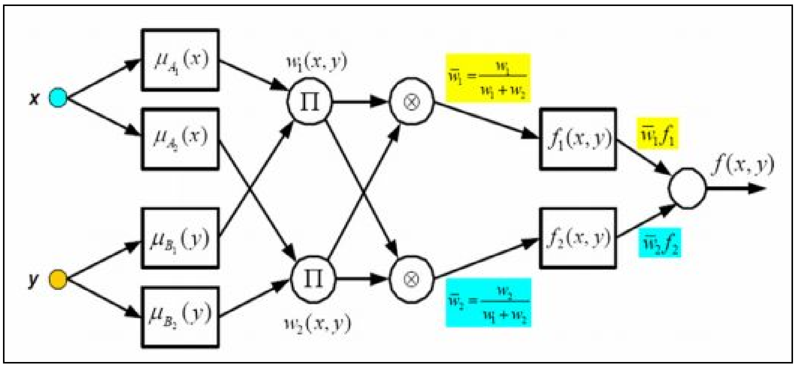

Figure 4.9 : ANFIS architecture

Multinomial Logistic Regression as an Estimation Method

Multinomial logistic regression is used when the dependent variable in question is nominal and consists of more than two categories. In our work, multinomial logistic regression would be appropriate, because we are trying to determine how factors to predict glucose level causing diabetes disease.

The multinomial logistic model assumes that data are case specific; that is, each independent variable has a single value for each case. The multinomial logistic model also assumes that the dependent variable cannot be perfectly predicted from the independent variables for any case.

$$\Pr(y_i = j) = \frac{\exp(X_i \beta_j)}{1 + \sum_j^J \exp(X_i \beta_j)}, \qquad (7)$$

$$\Pr(y_i = 0) = \frac{1}{1 + \sum_j^J \exp(X_i \beta_j)}, \qquad (8)$$

According to multinomial logistic regression model, which is defined in (7) and (8), the ith is individual, y_i is the observed outcome and X_i is a vector off explanatory variables. The unknown parameters β_j are typically estimated by maximum likelihood (Keles & Keles 2006).

Findings

In this section, the performance of ANFIS was evaluated, by compared with the performance of Multinomial Logistic Regression.

After preparation of dataset explained in *material and method* section, ANFIS module of the diabetes expert system was run for training and checking datasets. Before running ANFIS, user can determine which inputs will be used as shown in Figure 4.10.

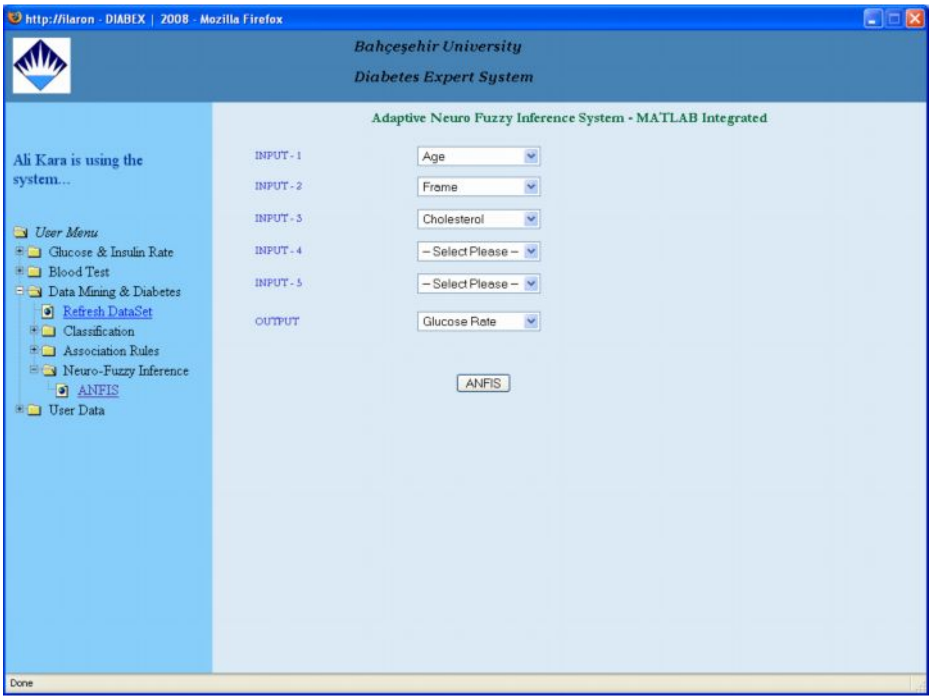

Figure 4.10 : Input and output selection for ANFIS

Matlab integration was used for ANFIS. The result of ANFIS is read from Matlab Logs and shown in the diabetes expert system page, as shown by Figure 4.11. Although we used 300 instances for training and 90 ones for checking, ANFIS reached the results at just epoch 2.

Figure 4.11 : ANFIS results

After training and checking the data, estimation module by ANFIS could be run. The estimation module uses the rules found by Matlab engine, and estimates a glucose rate using input variables. The sample page of the diabetes expert system is shown in Figure 4.12.

Figure 4.12 : Estimation by ANFIS

Same training and checking datasets were run by multinomial logistic regression (MLR) model in SPSS. And then the results shown as Table 4.4 were reached.

Table 4.4 : Results of benchmarking

Method	epoch	Standard Error	Error percentage	Data Type
ANFIS	2	0.7095	**0.1418**	Train
ANFIS	2	0.8725	**0.1745**	Check
MLR	300	0.7083	**0.1417**	Train
MLR	90	1.1713	**0.2343**	Check

I would like to have your attention to 2 important points while evaluating the results. It's clear to see from Figure 4.13; standard errors of *training* datasets for both of the methods are very similar. However same parameter of *checking* datasets shows that multinomial logistic regression has much bigger standard error than ANFIS.

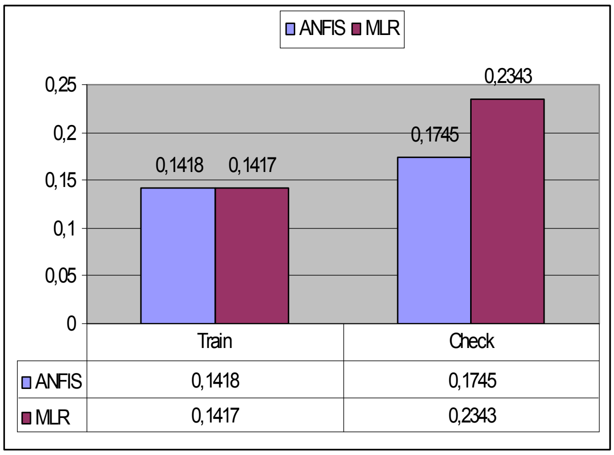

Figure 4.13 : Standard error percentages

Another important point is difference between learning durations of the methods. As seen from Table 4.4, the learning duration of ANFIS is shorter than multinomial logistic regression. ANFIS training could be completed at epoch 2; but MLR should evaluate the whole dataset.

Many medical tests have issues with **sensitivity** and **specificity**. Before dealing with the terms, it's better to exlain what they mean.

Sensitivity: is the ability to correctly detect diabetes disease.

Specificity: is the ability to avoid calling Normal things as diabetes disease.

A perfect diabetes test would have 100 percent sensitivity and 100 percent specificity. It would positively identify all the true cases of disease, and it would never mislabel anything normal as disease. When a diabetes test is imperfect, the expert system should try to strike a balance between sensitivity and specificity. To do this, a chart might be plotted which is formed by sensitivity and 1-specificity on a graph, called a "ROC curve". ROC means Receiver Operator Characteristic.

ROC curve is drawn in Figure 4.14, according to the Check Data of ANFIS and MLR.

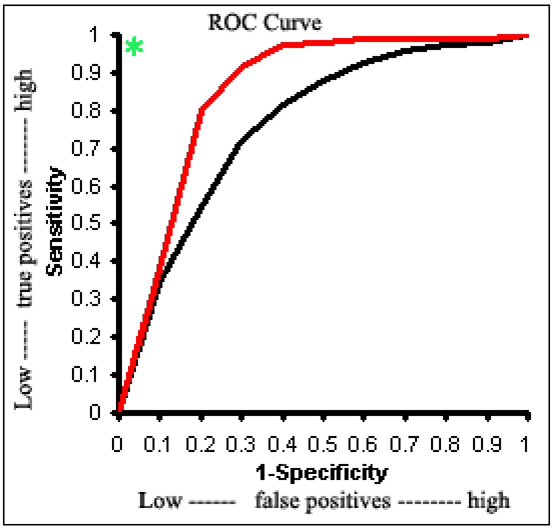

Figure 4.14 : ROC curve of check data

The **Red** line (ANFIS) is a better medical test than the **Black** line (MLR), because the curve of the Red line comes closer to the **Green** asterisk. To pick the best point along the ROC curve, it's generally looked for the shortest distance from the **Green** asterisk, to the **Red** line. In this case of the thesis, ANFIS gives **0.81** sensitivity (81 percent) and 0.21 false positive fraction (**79 percent** specificity), and MLR gives **0.77** sensitivity (77 percent) and 0.24 false positive fraction (**76 percent** specificity).

5. RESULTS & DISCUSSIONS

An expert system is developed for diabetic patients. The main purpose of the system is to guide diabetic patients during the disease. Diabetic patients could benefit from the diabetes expert system by entering their daily glucoses rate and insulin dosages; producing a graph from insulin history; consulting their insulin dosage for next day. The diabetes expert system is not only for diabetic patient, but also for the people who suspect if they are diabetic. For this aim, we have developed a diabetes estimation module which is developed by prolog server pages (PSP) as a web-based artificial intelligence language.

It's also tried to determine an estimation method to predict glucose rate in blood which indicates diabetes risk.

Continuous values are initially designated in the dataset, and then converted to fuzzy values. Glucose rates (dependent variable) were made fuzzy, instead of binary. Binary values have high accuracy, but don't have enough information about diabetes risk. After preprocessing dataset, we run ANFIS method in Matlab and Multinomial Logistic Regression method in SPSS. Table 3 summarizes the results of benchmark we made between ANFIS and multinomial logistic regression.

It's found out that learning duration of ANFIS is much shorter than MLR's duration. When a more sophisticated system with a huge data is imagined, the use of ANFIS instead of multinomial logistic regression would be more useful to overcome faster the complexity of the problem.

In training of the data, ANFIS and MLR gave quite similar results with standard error. However, when the trained parameters were applied to checking data, standard error of ANFIS is smaller than that of MLR. This shows that ANFIS is a better and faster learning method than multinomial logistic regression. Consequently it could be said, if we have a system which contains fuzzy inputs and output, ANFIS is better system than MLR for diabetes diagnosis.

REFERENCES

Books

Polat, K., & Gunes, S., (2006), *An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease*, Elsevier, Digital Signal Processing.

Agrawal R., & Srikant R., (1994), *Fast Algorithms for Mining Association Rules*, VLDB. Chile, 487-99, ISBN 1-55860-153-8

Chong, Q., Marwadi, A., Supekar, K. & Lee, Y., (2005*), Ontology Based Metadata Management in Medical Domains*, Published by School of Interdisciplinary Computing and Engineering University of Missouri – Kansas City

Dobroslawa, M., Grzymala-Busse, M., & Grzymala-Busse, W, (1995), *On The Usefulness Of Machine Learning Approach To Knowledge Acquisition, Department of Electrical Engineering and Computer Science*, University of Kansas, Lawrence, KS 66045, USA

Karagiannis, S., Dounis, A.I., Chalastras, T., Tiropanis, P. & Papachristos, D., (2006), *Design of Expert System for Search Allergy and Selection of the Skin Tests using CLIPS*, INTERNATIONAL JOURNAL OF INFORMATION TECHNOLOGY VOLUME 3 NUMBER 2 ISSN, 1305-2403.

McCelland, J.L., Rumelhart, D.E. & PDP Research Group (Eds), (1986), *Parallel Distrubuted Processing: Exploratios in The Microstructeres of Cognition*, pp.216-271. Cambridge, MA: Bradford / MIT Press.

Patterson, D. W., *Artificial Neural Networks* – Theory and Applications, Prentice

Pop, D. & Negru, V., (2003), *An Extensible Environment for Expert System Development,* Department of Computer Science, University of the West from Timi_oara 4 V. Parvan Street, RO-1900 Timioara, Romania.

Quinlan, J.R., (1985), *Induction of Decision Trees, Centre for Advanced Computing Sciences,* New South Wales Institute of Technology, Sydney 2007, Australia

Articles

Agrawal, R., Imielinski T., & Swami AN., (June 1993), *Mining Association Rules between Sets of Items in Large Databases.* SIGMOD, **22**(2):207-16.

Culliere, T., Titli, A., & Corrieu, J., (1995), *Neuro-fuzzy modelling of nonlinear systems for control purposes,* In Proc. IEEE INT. Conf. on Fuzzy Systems, pp 2009-2016, Yokohama.

Edoura-Gaena, R.B., Allais, I., Gros, J.B., & Trystram., G., (2006), *A decision support system to control the aeration of sponge finger batters*, Elsevier, Food Control **17**, 585–596.

Edoura-Gaena, R.B., Allais, I., Gros, J.B., & Trystram., G., (2006), *A decision support system to control the aeration of sponge finger batters"*, Elsevier, Food Control **17**, 585–596.

Edward R., & Omiecinski., (2003), *Alternative interest measures for mining associations in databases.* IEEE Transactions on Knowledge and Data Engineering, **15**(1):57-69.

Flach P.A., & N. Lachiche, (1999), *Confirmation-Guided Discovery of first-order rules with Tertius. Machine Learning,* **42**, 61-95.

Fleming, G., van der Merwe, M., & McFerren, G., (2007), *Fuzzy expert systems and GIS for cholera health risk prediction in southern Africa*, Elsevier, Environmental Modelling & Software **22**, 442 – 448.

Gonzalez-Andujar, J.L., Fernandez-Quintanilla, C., Izquierdo, J., & Urbano, J.M., (2006), *SIMCE: An expert system for seedling weed identification in cereals*, Elsevier, Computers and Electronics in Agriculture **54**, 115–123.

Guler, I., & Ubeyli, E. D., (2006), *Expert systems for time-varying biomedical signals using eigenvector methods,* Elsevier, Expert Systems with Applications.

Gutierrez-Estrada, J.C., De Pedro Sanz, E., Lopez-Luque, R,. & Pulido-Calvo, I., (2006), *SEDPA, an expert system for disease diagnosis in eel rearing systems*, Elsevier, Aquacultural Engineering **33**, 110–125.

Huang, M.J., & Chen, M.Y., (2007), *Integrated design of the intelligent web-based Chinese Medical Diagnostic System (CMDS) – Systematic development for digestive health,* Elsevier, Expert Systems with Applications **32**, 658–673.

Hwang, G.H., Chen, J.M., Hwang, G.J., & Chu, H.C, (2006), *A time scale-oriented approach for building medical expert systems,* Elsevier, Expert Systems with Applications **31**, 299–308.

Jang, J. (1993), *ANFIS: Adaptive-Network Based Fuzzy Inference System*. IEEE Trans. On Systems, Man and Cybernetics, Vol. **23**. No. 3 pp. 665-685.

Keles, A., & Keles, A., (2006), ESTDD: *Expert system for thyroid diseases diagnosis*, Elsevier, Expert Systems with Applications.

Kim, J.S., (2006*), Development of a user-friendly expert system for composite laminate design*, Elsevier, Composite Structures.

Mamdani, E.H., & Assilian, S., (1975), *An Experiment in Linguistic Synthesis with a Logic Controller,* Int. J. Man – Machine Studies **8**, pp. 1 – 13.

Mangalampalli, A., Mangalampalli, S. M., Chakravarthy, R., & Jain, A. K., (2006), *A neural network based clinical decision-support system for efficient diagnosis and fuzzy-based prescription of gynecological diseases using homoeopathic medicinal system,* Elsevier, Expert Systems with Applications **30**, 109–116.

Nammuni, K., Pickering, C., Modgil, S., Montgomery, A., Hammond, P., Wyatt, J.C., Altman, D.G., Dunlop, R., & Potts, H.W.W., (2004), *Design-a-trial: a rule-based decision support system for clinical trial design*, Elsevier, Knowledge-Based Systems **17**, 121–129.

Nauck, D., (1994), *Fuzzy neuro systems: An overview*; R., Gebhardt J., & R. Palm, R., (1994), *Fuzzy Systems in Computer Science*, pp 91-107, Vieweg, Braunschweig,.

Piasecki, M., & Sas, J., (2005*), Partial Parsing Method Applied to Rules Acquisition for Medical Expert System,* Computer Science Department of Wroclaw University of Technology ul. Wybrzeze Wyspianskiego **27**, 50-370, Wroclaw, Poland.

Polat, K. & Gunes, S., (2006), *An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease*, Elsevier, Digital Signal Processing.

Pontow, C., Dazinger, F. & Schubert, R., (2007), *A framework in prolog for computing structural relationship*, Elsevier, Data & Knowledge Engineering **62**, 308–326.

Prasad, R., Ranjan, K. R. & Sinha, A.K., (2006), *AMRAPALIKA: An expert system for the diagnosis of pests, diseases, and disorders in Indian mango*, Elsevier, Knowledge-Based Systems **19**, 9–21.

Seising, R., (2006), *From vagueness in medical thought to the foundations of fuzzy reasoning in medical diagnosis*, Elsevier, Artificial Intelligence in Medicine.

Shaalan, K., Rafea, M. & Rafea, A., (1998), *KROL: a knowledge presentation object language on top of Prolog, Pergamon*, Expert Systems with Applications **15**, 33 – 46.

Sojda, R.S., (2007), *Empirical evaluation of decision support systems: Needs, definitions, potential methods, and an example pertaining to waterfowl management,* Elsevier, Environmental Modelling & Software **22**, 269 - 277.

Staszewski, W.J., & Worden, K., (1997), *Classification of Faults in Gearboxes Preprocessing Algorithms and Neural Networks*, Neural Computing and Applications **5**(3): 160-183.

Takagi, S., & Sugeno, M, (1985), *Fuzzy identification of fuzzy systems and it'sapplication to modelling and control,* IEEE Trans. Systems Man Cybern., **15** pp 116-132.

Turner, C., Bishay, H., Bastien, G., Peng, B. & Phillips, R.C., (2006), *Configuring policies in public health applications*, Elsevier, Expert Systems with Applications.

VITAE

Ali Kara was born in Burhaniye of Balikesir. He received his B.Sc. degree in Industrial Engineering from Yildiz Technical University in 2003. Since then he has been an ERP consultant over Oracle – JD Edwards solution. His main areas of interest are machine learning, bio-medical and Artificial Intelligence.