From: B.V. Ambrosiadou, S. Vadera, V. Shankararaman, D. Goulis, and G. Gogou, Decision Support Methods in Diabetic Patient Management by Insulin Administration. Neural Networks vs Induction Methods for Knowledge Classification Proc. 2nd ICSC Symposium on Neural Computation, Berlin, (2000).

Decision Support Methods in Diabetic Patient Management by Insulin Administration
Neural Network vs. Induction Methods for Knowledge Classification

B.V.Ambrosiadou[1], S. Vadera[2], V. Shankararaman[1], D. Goulis[3], [4]

[1]Department of Computer Science, University of Hertfordshire, England
[2]Department of Computer Science, University of Salford, England
[3] Clinical Research Fellow in Endocrinology & Diabetes, Imperial College School of Medicine, St. Mary's Hospital, London, UK
[4]Medical Informatics Laboratory, Aristotelion University of Thessaloniki

Abstract

Diabetes mellitus is now recognised as a major world-wide public health problem. At present, about 100 million people are registered as diabetic patients. Many clinical, social and economic problems occur as a consequence of insulin-dependent diabetes. Treatment attempts to prevent or delay complications by applying 'optimal' glycaemic control. Therefore, there is a continuous need for effective monitoring of the patient. Given the popularity of decision tree learning algorithms as well as neural networks for knowledge classification which is further used for decision support, this paper examines their relative merits by applying one algorithm from each family on a medical problem; that of recommending a particular diabetes regime. For the purposes of this study, OC1 a descendant of Quinlan's ID3 algorithm was chosen as decision tree learning algorithm and a generating shrinking algorithm for learning arbitrary classifications as a neural network algorithm. These systems were trained on 646 cases derived from two countries in Europe and were tested on 100 cases which were different from the original 646 cases.

1.0 Introduction

Diabetes is a chronic disease. The prevalence of diabetes mellitus is rising, especially in developing countries as they adopt a Western lifestyle [1]. Long term complications involving the eyes, the kidneys, the central and peripheral nervous system may appear to those patients that do not achieve an 'optimal' glycaemic control. Diabetic patient management may be achieved by appropriate diet, exercise and insulin administration for glycaemic control of Type I or Type II insulin dependent diabetic patients. There is divergence of opinions regarding the issue of insulin administration among the experts, which mainly depends on 'soft' information such as their educational and social background as well as their experience and gut feel. An effort has been made to systematise the process of insulin administration by developing decision support tools, which facilitate a consistent and objective decision making among specialists [2,3]. The knowledge in this field has been mainly acquired from a number of sources across Europe in order to alleviate the differences due to lifestyles and diet habits. A DELPHI approach was followed in order to arrive at consensus opinion in those cases where there was divergence [4]. In this paper two methods are compared for knowledge classification in insulin administration which may later be used for decision support purposes. These are the decision tree learning and the neural network algorithmic approaches. The following sections describe the domain under consideration, the two methods as well as the results obtained.

1.1 The Domain: Insulin Regime Prescription and Dose Adjustment

Insulin regimens and dose adjustment are prescribed by diabetologists depending on a number of factors such as diabetes type, patient age, activity during the day and control targets. There are three insulin types depending on the beginning and duration of their action. These are the fast, intermediate and long acting insulin types. Insulin regimens are insulin types in combinations administered in daily profiles. As insulin administration proceeds food intake by approximately half an hour, it may take place before breakfast, lunch, dinner and bed.

The most widely used insulin regimens are:

1. short-acting insulin mixed with intermediate-acting insulin, given twice daily before meals
2. short-acting insulin mixed with intermediate-acting insulin, given before breakfast AND short-acting insulin, given before evening meal AND intermediate-acting insulin, given at bedtime
3. short-acting insulin, given three times daily AND intermediate-acting insulin, given at bedtime
4. intermediate-acting insulin, given once daily

Regimen No	Pre-Breakfast	Pre-Lunch	Pre-Tea	Pre-Bed
1	Medium acting	-	-	-
2	Fast+Medium acting	-	Fast+Medium acting	-
3	Fast acting	Fast acting	Fast acting	Medium acting
4	Fast acting	Fast acting	Fast acting	Fast acting

Table 1. Most common regimens.

Step No	Subject	Action
1	group of experts in UK & Greece	*compilation of list :* most widely-used insulin regimens
2	authors	*compilation of questionnaire :* decision-making parameters for insulin regimen selection
3	group of experts in UK & Greece	*completion of questionnaire :* decision-making parameters for insulin regimen selection
4	authors	*software development :* Machine Learning (ML) & Neural Network (NN) for Regimen Adjustment
5	group of experts in UK & Greece	*completion of a table :* 646 diabetic cases complete with appropriate regimen
6	authors	*Software training :* training of ML & NN Regimen Adjustment software, using the 646 cases
7	authors	*Software evaluation :* evaluation of ML & NN Regimen Adjustment software, using 100 additional cases

Table 2. Study design.

Generally, the regimens that contain long acting insulin are not prescribed because they are considered outdated and they are used only out of necessity [6-8], especially in elderly patients. The intensive glucose control (regimen 3) gives the best results in terms of blood glucose control but, on the other hand, results in more hypoglycaemic episodes (low blood glucose) which is a, potentially, dangerous situation. The result of intensive glucose control is the most frequent appearance of hypoglycemia (low blood glucose) which is even more dangerous than hyperglycemia (high blood glucose values) [6], [7], [9].

Depending on special occasions other regimens can also be prescribed. These life circumstances include coexistence of another acute or chronic disease, short life expectancy, honeymoon period, psychosomatic problems due to injections, the environment and inability of the patient to understand the demands of a particular regimen [6-8].

2.0 The Comparative Study – An Overview

2.1 Study Design

The prototype systems described in this paper classify clinicians' knowledge in the domain of insulin administration. In this way, they may further support decision making in this field. The whole study design is presented in **Table 2**.

2.2 System Knowledge

For the purposes of the development of the ML & NN software for insulin regimen specification, a list of regimens was compiled by interviewing diabetes experts in the United Kingdom and Greece (Step 1). Subsequently, a questionnaire was prepared (Step 2) and was sent to three diabetic departments of UK and fourteen diabetological centres in Greece. The questionnaire was asking for the parameters that were necessary in order to decide for each specific insulin regimen (Step 3), among the ones that were proposed in Step 1.

The outcome of the above process was a choice of the main factors intuitively used by doctors for insulin regimen prescription. These are:

- Diabetes type (unknown, type I, type II)
- Patient age
- What the patient is used to taking (unknown, tablets, insulin)
- Special condition (pregnancy, surgery, infection)
- Dawn phenomenon (yes, no)
- Unstable ("brittle") diabetes (yes, no)
- Glucose profile (morning, afternoon, evening, night / unknown, normal, hyperglycemia, hypoglycemia)
- Physical activity (morning-noon, afternoon-evening, night / none-unknown, sedentary, light, heavy)
- Food intake (breakfast, lunch, tea, dinner)
- Desirable blood glucose control (fair, good, very good)

In the same way the main factors used by doctors for insulin dose adjustment are:

- Insulin regimen
- Current dose
- Glucose values
- Glucose profile (morning, afternoon, evening, night / unknown, normal, hyperglycemia, hypoglycemia).

Case No	**DM type**	**age**	**special**	**Previous Rx**	**target**	**dawn**	**unstable**	**BG-bre**	**BG-lun**	**BG-din**	**BG-bed**	**PA-mor**	**PA-aft**	**PA-nig**	**FI-bre**	**FI-lun**	**FI-din**	**FI-bed**	**Regimen No**
1	1	n	i	vg	n	n	n	hi	hi	hi	hi	l	-	y	y	y	y	**4**	
2	1	40	y	i	vg	n	n	hi	nl	nl	nl	s	-	-	y	y	y	n	**4**

Table 3. Parameters for insulin regimen prescription with two sample cases.

parameter	**values**	**explanation**
DM type	1 2	diabetes mellitus, type
age	number	age in years
special	yes (y) no (n)	special condition : pregnancy, surgery, infection
previous Rx	insulin (i) tablets (t)	patient's previous regime
target	fair (f) good (g) very good (vg)	desirable diabetes control
dawn	yes (y) no (n)	dawn phenomenon
unstable	yes (y) no (n)	unstable diabetes
BG-bre	normal (nl) hyperglycemia (hi) hypoglycemia (lo)	blood glucose - breakfast
BG-lun	normal (nl) hyperglycemia (hi) hypoglycemia (lo)	blood glucose - lunch
BG-din	normal (nl) hyperglycemia (hi) hypoglycemia (lo)	blood glucose - dinner

BG-bed	normal (nl) hyperglycemia (hi) hypoglycemia (lo)	blood glucose - bed
PA-mor	sedentary (s) light (l) heavy (h)	physical activity - morning
PA-aft	sedentary (s) light (l) heavy (h)	physical activity - afternoon
PA-nig	sedentary (s) light (l) heavy (h)	physical activity - night
FI-bre	yes (y) no (n)	food intake - breakfast
FI-lun	yes (y) no (n)	food intake - lunch
FI-din	yes (y) no (n)	food intake - dinner
FI-bed	yes (y) no (n)	food intake - bed

Legend for **Table 3**.

3.0 Methods

3.1 Machine Learning Algorithms

Since its initial development, the ID3 machine learning algorithm has provided the basis for several decision tree learning methods. Figure 1 shows some of the well known algorithms that have evolved to reduce deficiencies in ID3 and to tackle issues such as costs of carrying out tests.

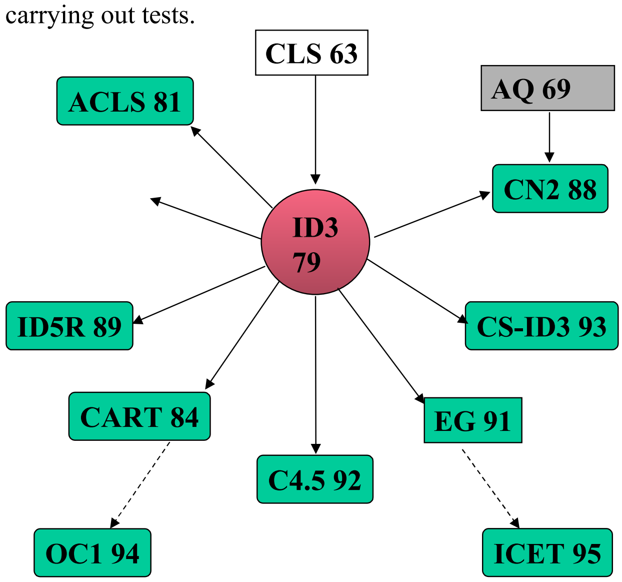

Figure 1: Some well known decision tree learning algorithms

These algorithms have been applied in various domains, and several experimental studies have evaluated their accuracy using benchmarking data

From the set of available algorithms, Murthy's OC1 [10] was chosen to represent the decision tree learning algorithms family, mainly because it constructs trees using a core algorithm that it inherits from ID3, and which is a characteristic of most decision tree learning algorithms. Section 3.1.1 gives an overview of the algorithm, and section 3.1.2 presents the results obtained.

3.1.1 Overview of the OC1 Algorithm

The main principle behind tree induction algorithms is to develop a tree that is consistent with the table of examples representing the training set. The core procedure to obtain such a tree can be summarised as follows:

If all the result values are the same, stop
Otherwise

- Select the root from the available attributes
- Partition the table so that each sub-table's Root column has the same value.
- Apply the algorithm recursively to find a decision tree for each of the sub-tables, ignoring the Root attribute.

The selection of the root is usually based on a metric that utilise information theoretic measures [11].

Numeric attributes are partitioned by considering all potentially useful axis parallel splits [12], or by adopting linear divisions [13]. The OC1 algorithm provides options for utilising a range of selection metrics, and allows the use of axis parallel splits, as well as a scheme for linear splits.

This type of process can produce deep trees that over trained, with terminal nodes based on only a few examples. There are various tree pruning methods that can be used to overcome this problem [14]. The central operation in post-pruning is to replace a subtree by a leaf node which takes the value of the majority class of the examples in the subtree provided it results in an improvement in accuracy.

OC1 adopts a techniques, known as the minimum cost-complexity pruning method [15]. This technique works in two stages. First it generates a series of pruned trees. This is done by considering the effect of replacing each subtree by a leaf and working out the reduction in error per leaf (α):

$$\alpha = \frac{R(t) - R(T_t)}{N_t - 1}$$

where $R(t)$is the expected error rate if the subtree is pruned, $R(T_t)$ is the expected error rate if the subtree is not pruned, and N_t is the number of leaf nodes in the subtree. The subtree with least value per leaf (smallest α) is pruned by replacing the subtree by its majority class, and the process repeated recursively to generate a series of increasingly pruned trees. Second, once a series of increasingly pruned trees have been generated, it uses an independent testing set to select a tree that is the most accurate (more precisely, it selects the smallest tree within one standard error of the most accurate tree)

3.1.2 Results

The OC1alogorithm was tested by applying it on the training dataset and testing it on the 100 additional cases. It was run with both an axis-parallel option set, when it is similar to C4.5, and with the use of linear divisions. The Information gain measure was used for the selection metric, and all other parameters were set to their default values (i.e., they were not tuned in any way).

The following table summarises the results in terms of accuracy and size of trees for the different variations.

In terms of comprehensibility, the simple splits adopted by utilising axis-parallel splits are easier to relate to the domain by an expert than the linear equations present when linear divisions are used. The use of linear divisions with pruning results in the smallest tree, although it is the only tree that is not 100% correct.

RESULTS BEFORE PRUNING			
	Max Depth	Leaves	Accuracy
Axis Parallel Version	14	42	100%
Linear Division Version	9	34	100%
RESULTS AFTER PRUNING			
Axis Parallel Version	11	32	100%
Linear Division	8	17	99%
Cat 1 = 100% (8/8) Cat 2 = 97.92% (47/48) Cat 3 = 100% (37/37) Cat 4 = 100% (7/7)			

Table 4 Results

3.2 Neural Network Algorithm

3.2.1. The Generating-Shrinking Algorithm –System Architecture

The description here is based on [16. The algorithm uses a three layer feed-forward neural network. The first, input, layer (IL) contains linear neurons, the second, hidden, layer (HL) contains sigma-pi neurons, and the third, output, layer (OL) contains linear threshold neurons. The number of neurons in IL and HL equals the number of training patterns, and the number of neurons in OL equals the number of classes (in our case insulin regimens) that the neural network has to distinguish.

The first layer has as input an n-dimensional pattern vector $p \in R^n$ to be classified, along with a fixed reference number $r \in R$. Together p and r they form an (n+1)-dimensional vector $(p,r) \in R^{n+1}$. The neurons' outputs in the third layer are binary, so when the output of the ith neuron is 1, the input pattern belongs to the ith class. In this sense, the network output is defined to be the ordinal number of the neuron in the output layer whose output is 1. The input-output associations of each layer are described by the following three equations:

Input Layer:

$$o_i^{IL} = \sum_{j=1}^{n+1} w_{ij}^{IL} p'_j \ , \ i = 1,2,...,n^{IL} \qquad (1)$$

where o_i^{IL} is the output of the ith neuron in the input layer, w_{ij}^{IL} is the weight of the connection between the jth input component and the ith neuron in the input layer, n^{IL} is the number of neurons in the input layer and p'_j is the jth component of p'.

Hidden Layer:

$$o_i^{HL} = \prod_{j=1 (j \neq i)}^{n^{IL}} f\left(w_{ij}^{HL} o_j^{IL} + w_{ii}^{HL} o_i^{IL}\right),$$

$$i = 1,2,...,n^{HL} \qquad (2)$$

where $f(x) = \begin{cases} 1, x > 0 \\ 0, x \leq 0 \end{cases}$,

o_i^{HL} is the output of the ith neuron in the hidden layer, w_{ij}^{HL} is the weight of the connection between the output of the jth neuron in the input layer and the ith neuron in the hidden layer and n^{HL} is the number of neurons in the hidden layer.

Output Layer:

$$o_i^{OL} = f\left(\sum_{j=1}^{n^{IL}} w_{ij}^{OL} o_j^{HL} - 0.5\right), \ i = 1,2,...,n^{OL} \qquad (3)$$

where o_i^{OL} is the output of the ith neuron in the output layer, w_{ij}^{OL} is the weight of the connection between the output of the jth neuron in the hidden layer and the ith neuron in the output layer and n^{OL} is the number of neurons in the output layer.

3.2.4. Neural Network Results

At first the Neural Network was not trained at all. The 646 cases were entered one by one as an input to the NN. If the Neural Network's output was the same answer as the expert's decision the next case was entered in the NN. If not, the case was used as a new training pattern for the NN before we go on to the next one. In this way the Neural Network had a little more "experience" before the processing of the next case.

In this way after the 1st pass of the 646 cases, the correct answers were 452 and the Neural Network had 194 training patterns (the wrong answers). The above process was repeated with the cases that were correctly answered to check if the patters that were added would change the Neural Network's decision, until the Neural Network was able to give the correct answer for all cases.

	1st pass	**2nd pass**	**3rd pass**	**4th pass**
Number of cases	646	452	412	409
Correct answers	452 (70,0 %)	412 (91,1 %)	409 (99,2 %)	409 (100,0 %)
Wrong answers	194 (30,0 %)	40 (8,9 %)	3 (0,8 %)	0 (0,0 %)
Training patterns	194	234	237	237

Table 5: Training results

Regimen	**Correct**	**Wrong**	**%**
1	4	8	33%
2	30	6	83%
3	22	16	58%
4	2	12	14%
Totals	**58**	**42**	**58%**

Table 6: Evaluation of the Neural Network

5. Conclusion

It was found that the decision tree learning algorithm out-performed the neural network in the evaluation phase. As a next step, the expert will comment on the outcome of the two algorithms, in order to determine why the neural network did not perform as expected.

Bibliography

1. Unger R, Foster D. (1992) Diabetes mellitus. In: Wilson J, Foster D (eds). *Williams textbook of endocrinology*, 8th ed. (Philadelphia, WB Saunders), 1255-1334.
2. Ambrosiadou V., Goulis D.G., Pappas C. (1996) "Clinical Evaluation of the DIABETES Expert System for Decision Support by Multiple Regimen Insulin Dose Adjustment". Computer Methods and Programs in Biomedicine, Vol. 49, 105-115.
3. Ambrosiadou V., Gogou G., Pappas C., Maglaveras N. (1996) "Decision Support for Insulin Regime Prescription Based on a Neural Network Approach". Journal of Medical Informatics, Taylor & Francis, Vol. 21, No.1, 23-34.
4. Ambrosiadou V., Goulis D. (in press) "The DELPHI Method as a Consensus and Knowledge Acquisition Tool for the Evaluation of the DIABETES System for Insulin Administration". Journal of Medical Informatics.
5. Luger G.F., Stubblefield, W.A. (1998) Artificial Intelligence Structures and Strategies for Complex Problem Solving, Third Edition, Addison Wesley Longman.
6. Foster D.W. (1987) "Diabetes mellitus", in Harrison's Principles of Internal Medicine, 11th ed, Mc Graw-Hill, 1778-1797.
7. Garber A.J. (1994) "Diabetes mellitus", in Stein J.H. (editor), Internal Medicine, 4th ed, Mosby, 1391-1430.
8. Group of European Politics IDDM. (1993) "Instructions for the Treatment of Insulin-Dependent (type I) Diabetes- Consensus Guidelines ".
9. Miller AS, BH Blott, TK Hames (1992) "Review of neural network applications in medical imaging and signal processing", Med & Biol Eng & Comput, vol. 30, 449-464.
10. S. Murthy and S. Kasif and S. Salzberg (1994), A System for Induction of Oblique Decision Trees, Journal of Artificial Intelligence Research, 2, 1-32
11. Vadera, S. and Nechab, S. (1994), ID3, its Children and their Safety }, BCS Specialist Group on Expert Systems Newsletter 31,11-21
12. Quinlan, J.R. (1992), C4.5 : A Program for Machine Learning, California:Morgan Kauffman
13. Vadera, S. and Nechab, S. (1996), Inducing Safer Trees, Proceedings of the International Conference on Knowledge Based Computing Systems, Narosa Publishing, pp205-216.
14. Breslow, L.A. and Aha, D.W. (1997), Simplifying decision trees: A survey, Knowledge Engineering Review, 12,1-40.
15. L. Breiman, L. and Friedman, J.H. and Olsen, R.A. and Stone, C. J. (1984), Classification and Regression Trees, Belmont: Wadsworth
16. Yan Qiu Chen, David W., Thomas, and Mark S. Nixon. (1994) "Generating-Shrinking Algorithm for Learning Arbitrary Classification". Neural Networks Vol. 7, No. 9, 1477-1489.