DATAMINING AND NETWORKS NEURONAL

EXTRACTING KNOWLEDGE FROM HIGH PRESSURE DATA PATIENTS.

Mbuyi Mukendi Eugène[1], Kafunda Katalayi Pierre[2], Mbuyi Badibanga Steve[3], Mbuyi Mukendi Didier[4]

[1] Professor University of Kinshasa, Department of Computer Sciences, DR Congo
[2] University of Kinshasa, Department of Computer Sciences, DR Congo
[3] University of Kinshasa, Department of Computer Sciences, DR Congo
[3] University of Kinshasa, Department of Computer Sciences, DR Congo

Kinshasa Computer science Laboratory.

Summary

Data mining is a set of methods and techniques for exploring and analyzing automatically or semi-automatically databases in order to detect rules, associations, unknown or hidden trends, specific structures that restore most of the useful information while reducing the amount of data.This is a process of extracting valid and tractable knowledge from large amount of data.In this paper, we present a contribution on the extraction of useful knowledge from databases on patients with high blood pressure from one of the hospital in Kinshasa (RD Congo), using multi-layerneural networks.

Key words: Datamining, extraction of knowledge, database, Neural Network, algorithm, high blood pressure.

I. INTRODUCTION [1], [2], [3], [4], [5], [7], [8], [9], [10]

Learning can be seen as a problem of updating connection weights within a network in order to achieve the requested task. The learning rule allows network to evolve over time taking into account prior experiences. The connection weights are modified according to previous results for finding the best model in comparison to the given examples.

Neural networks is divided in two main classes, that is, supervised learning networks and unsupervised learning. Another class is called hybrid learning network.

In this work however, we focus on supervised learning. We present some theoretical concepts on learning algorithms: the back-propagation algorithm, used for extracting knowledge from high blood pressure data.

.

This algorithm is applied to implement neural networks done by Microsoft, i.e., Microsoft Neural Networks used in chapter II of this work.

I.1. Back-propagation algorithm

A. Definition :

1) A function $\sigma_k(x)$ is said sigmoid of parameter $k > 0$, if it is defined as follows $\sigma_k(x) = \frac{e^{kx}}{1+e^{kx}} = \frac{1}{1+e^{-kx}}$ (1)

This is an infinitely differentiable approximation of the Heaviside function threshold. The approximation is better when k is large. In this work, we take k = 1. Therefore,

$$\sigma(x) = \frac{e^x}{e^x+1} = \frac{1}{1+e^{-x}} \quad (2)$$

The derivative of this function will be used in the rule for updating the weights by the back-propagation algorithm.

$$\sigma'(x) = \frac{e^x}{(1+e^x)^2} = \sigma(x).(1-\sigma(x) \quad (3)$$

2) A n-input real cell unitis a real $\vec{x} = (x_1, x_2, ... x_n)$ defined by the synaptic weight $\vec{w} = (w_1, w_2 ..... w_n)$ and the outputα, $\alpha\left(\vec{x}\right)$ is computed with the following formula :

$$\alpha(x) = \frac{1}{1+e^{-y}} \quad with \quad y = \vec{x}.\vec{w} = \sum_{i=1}^{n} w_i\, x_i$$

A multilayer perceptron (MLP) is a neural network with hidden layers with well defined unit cells.

B. Algorithm principle

- As with the linear perceptron, the principle is to minimize error function. The next step is to calculate the contribution to the error of each of the synaptic weights.
- Let a MLP defined by a n-input architecture and p outcomes; $\vec{w}$ vector synaptic weights associated with

$$\left(\vec{x}^s, \vec{c}^s\right)$$

all network links. The MLP error on a learning sample S of examplesis defined by :

$$E\left(\vec{w}\right)=\frac{1}{2}\sum_{\left(\vec{X}^{s},\vec{C}^{s}\right)\varepsilon S}\ \sum_{k=1}^{P}\left(C_k^s-\alpha_k^s\right)^2$$

(4)

with α_k^s the kth component of the output $\vec{\alpha}^s$ computed by the MLP given the input $\vec{x}^s$.

The error thus measures the difference between the expected and calculated outputs on the full sample. Let's assume S is fixed, the problem is how to determine a vector $\vec{w}$ which minimizes $E\left(\vec{w}\right)$.

In the other hand, in the same way as for the perceptron with the Widrow-Hoff rule, rather than seeking to minimize the overall error on the full sample, we try to minimize the error on each individual example. Then the error error for an example is:

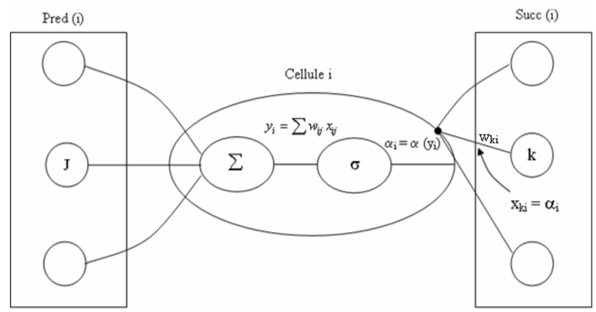
$$E_{\left(\vec{x},\vec{c}\right)}\left(\vec{w}\right)=\frac{1}{2}\sum_{k=1}^{P}(C_k-\alpha_k)^2$$

(5)

C. Notations

For simplification reasons, we adopt the following notations:

1) $E=E_{\left(\vec{x},\vec{c}\right)}$ synoptic weight function
2) The network has p output cells ;
3) If i is the index of a cell output, $c_i$is the expected output for that cell for the input $\vec{x}$;
4) $w_{ij}$is the synaptic weight associated with the link between cell j to i, implying that they are on two successive layers of the architecture, given the definition of the architecture;
5) $X_{ij}$is the input associated to the link between cell j to i ;
6) Pred (i) is a set of cells whose output is an input of the cell i ; this implies that the cell is not an entry cell and all elements of Pred(i) belong to the previous layer of which cell I belongs;
7) y_i is the total input of cell i, that is $y_i=\sum_{j\,\varepsilon\,\mathrm{Pred}(i)} w_{ij}\,x_{ij}$ (6)
8) $\alpha_i$is the output of the cell i, that is$\alpha_i=\sigma(y_i)$ (7)
9) Succ (i) is a set of cells having as input the output of cell i, this implies that this cell is not an output cell and elements of Succ (i) belong to the layer following the one containing i.



10) We note $\dfrac{\partial E\left(\vec{w}\right)}{\partial w_{ij}}=\dfrac{\partial E}{\partial w_{ij}}$ (8)

D. Remarks

We know that $\dfrac{\partial E}{\partial w_{ij}}=\dfrac{\partial E}{\partial y_i}.\dfrac{\partial y_i}{\partial w_{ij}}=\dfrac{\partial E}{\partial y_i}x_{ij}$ (9)

We need therefore to evaluate $\dfrac{\partial E}{\partial y_i}$. To do so, two cases have to be pointed out:

Cell i is an output cell or it is an intern cell.

a) <u>cell i is an output cell</u>

it follows that, $y_i$can only influence the output through the calculation ofα_i. Therefore $\dfrac{\partial E}{\partial y_i}=\dfrac{\partial E}{\partial \alpha_i}.\dfrac{\partial \alpha_i}{\partial y_i}$

Let us compute the two partial derivatives $\dfrac{\partial E}{\partial \alpha_i}$ et $\dfrac{\partial \alpha_i}{\partial y_i}$

$$\frac{\partial E}{\partial \alpha_i}=\frac{\partial}{\partial \alpha_i}.\left[\frac{1}{2}\sum_{i=1}^{P}\left(C_k-\alpha_k\right)^2\right]$$

The term corresponding to k = i has no null derivative, implying

$$\frac{\partial E}{\partial \alpha_i}=\frac{\partial}{\partial \alpha_i}\left[\frac{1}{2}\left(C_i-\alpha_i\right)^2\right]=-(C_i-\alpha_i) \qquad (10)$$

For second factor $\dfrac{\partial \alpha_i}{\partial y_i}$, we use the computation of the sigmoid function and the definition of the calculation of cell unit output.

$$\frac{\partial \alpha_i}{\partial y_i}=\frac{\partial \sigma(y_i)}{\partial y_i}=\sigma'(y_i)=\sigma(y_i)(1-\sigma(y_i))=\alpha_i(1-\alpha_i)$$

(11)

By replacing the two derivatives by their corresponding values, we obtain:

$$\frac{\partial E}{\partial y_i}=-(C_i-\alpha_i)\alpha_i(1-\alpha_i) \qquad (12)$$

b) <u>cell i is internal</u>

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

y_i will influence the network by all the calculations of the cells belonging to Succ(i).

$$\frac{\partial E}{\partial y_i} = \sum_{k \varepsilon Succ(i)} \frac{\partial E}{\partial yk} \cdot \frac{\partial y_k}{\partial y_i} = \sum_{k \varepsilon Succ(i)} \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial \alpha_i} \cdot \frac{\partial \alpha_i}{\partial y_i} = \sum_{k \varepsilon Succ(i)} \frac{\partial E}{\partial y_k} . w_{ki} . \alpha_i (1-\alpha_i)$$

$$\frac{\partial E}{\partial y_i} = \alpha_i (1-\alpha_i) \sum_{k \varepsilon Succ(i)} \frac{\partial E}{\partial y_k} . w_{ki} \qquad (13)$$

The study of the preceding two cases provide two equations (12) and (13) allowing to calculate partial derivatives $\frac{\partial E}{\partial y_i}$ for all cell i.

The calculation needs to be done for the output cells, then for the cell before the last layer until the first layer cells. This is the reason we speak about back-propagation. According to the equation (9), all the partial derivatives $\frac{\partial E\left(\vec{w}\right)}{\partial w_{ij}}$ can be computed using equation (12). To deduce the changes to bring on the synoptic weight, it is necessary to recall the gradient method

$$\Delta w_{ij} = -\varepsilon \frac{\partial E\left(\vec{w}\right)}{\partial w_{ij}} \qquad (14)$$

All tools are available for the back-propagation algorithm.

E. Remarks

1. Let's define $\delta_i = -\frac{\partial E}{\partial y_i}$, from equations (11), (12), (13) and (14) we obtain for output cell i:

   $\delta_i = \alpha_i (1-\alpha_i)(C_i - \alpha_i)$ (15) using (12)

   Pour une cellule interne :

   $\delta_i = \alpha_i (1-\alpha_i) \sum_{k \varepsilon Succ(i)} \delta_k w_{ki}$ (16) using (13)

   The modification of w_{ij} becomes :

   $\Delta w_{ij} = \varepsilon x_{ij} . \delta_i$ (17)

2. The modification rule for the linear perceptron weight is :

   $w_i \rightarrow w_i + \varepsilon (c - \alpha) x_i$.

   When dealing with MLP, this rule becomes:

   $w_{ij} \rightarrow w_{ij} + \varepsilon \delta_i x_{ij}$

   These two rules are very similar. The error $c - \alpha$ is replaced by a more complex term δ_i.

3. Constats :
   a. For an output cell i, the quantity δ_i corresponds to the usual error $c_i - \alpha_i$ multiplied by the derivative of the sigmoid function.
   b. For an intern i cell, le computation of δ_i depends on the weighted sum of the next (or following) layer errors.
   c. After presenting the input $\vec{x}$ and computing the output $\vec{\alpha}$, the calculation of errors δ_i will be done from the output layer to the input layer.

I.5. Back-propagation algorithm

A. Algorithm

Input : a sample S of $R^n \times R^P$; εa MLP with input layer c_o, $c_1, c_2, c_3, \ldots\ldots c_q-1$, q - 1 hidden layers ; an output layer c_q, n cells.

Random initialisation of the weight w_i in confidence interval [-0, 5 ; 0,5] for

$1 \leq i \leq n$

Repeat

Take an example $\left(\vec{x}, \vec{c}\right)$ of S and compute $\vec{\alpha}$;by back-propagation, compute δ_i.

For all output cell i, $\delta_i \leftarrow \alpha_i (1 - \alpha_i) (c_i - \alpha_i)$

[end For] Foreach layer from q – 1 to 1

Foreach cell i on the present layer

$$\delta_i = \alpha_i (1-\alpha_i) \sum_{k \in succ(i)} \delta_k w_{ki}$$

[end For]

[end For]

Update the weights

Forall weight $w_{ij} \leftarrow w_{ij} + \varepsilon \delta_i x_{ij}$ [end For]

[end Repeat]

Output : a MLP defined by a chosen initial structure and weigths w_{ij}

Automatic classifier

A. Definition: A classifier is a procedure (algorithm) that from a set of examples produces a prediction of any given class.

Given a set E of N pairs {(Xi, Yi)}, $1 \leq i \leq N$, $X \in R$, $Y \in \varepsilon^p$. Fw build a network capable of matching these forms, that is to say, such as: Fw (Xi) = Yi. This requires first of all by the choice of network architecture, to our case, it is a network with one hidden layer, then adjusted the synaptic weights W allowing the architecture to perform the task requested associative.

B. Assignment rule

We are in the case of a classification problem in p classes where one tries to learn to associate the network, each $X \in IRn$ form a training set E of cardinality N, $Y \in$ a form $\{-1.1\}^p$ such that Yik = 1 if X is of class k and Yik = -1 otherwise

II. APPLICATION OF THE METHOD OF NEURAL NETWORK TO HIGH PRESSURE BLOOD DATA[6], [12], [13], [14], [15], [16]

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

In this work we are interested in three complications of high blood pressure, said: Stroke, acute renal failure and others heart disease.

After extensive interviews with experts in the medical field and reading papers, we identified several factors about high blood pressure complication. We remember that the concept factor also refers to symptoms that may present a complication in a patientwith high blood pressure.

Following factors were considered:

- Age (age)
- BMI (IMC)
- Gender (sexe)
- HBP (chiffre)
- Headache (Yes / no) (presence des maux de tête)
- The presence of syncope (presence de syncope)
- The presence of vertigo (presence de vertige)
- The presence of eye disorder (presence de trouble oculaire)
- The presence of respiratory distress (presence de gene respiratoire)
- The presence of chest pain (angiine de poitrine)
- The presence of anuria (presence d'anurie)
- The presence of polyuria (presence de ployurie)
- The presence of urinary frequency (presence de pollakiurie)
- The presence of history for one of the complications (presence d'antécédents pour l'une des complications)
- The presence of seizures (presence de convulsion)
- The presence of fatigue (presence de fatigue)

II.1. LEARNING SAMPLE PRESENTATION

To build the learning sample size for neural network algorithm, we use the data for patients with high blood pressure from a hospital. This dataset is stored in SourceHTA which is a SQL Server 2008 relational database.

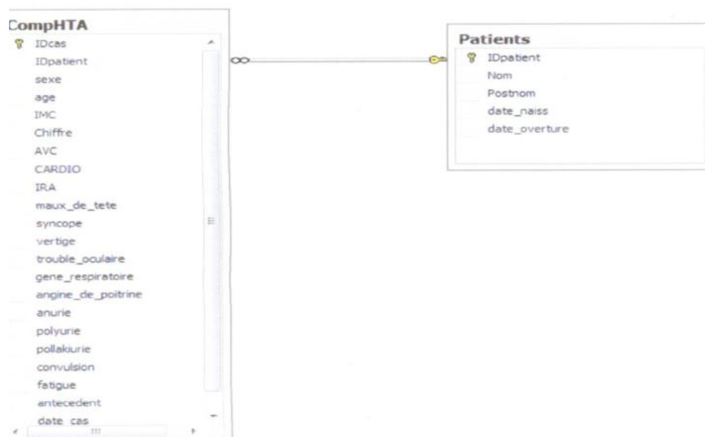
This database contains two tables;the first contains information on patients in orderto facilitateinformation searching information. The second table CompHTA contains information on complications due to high blood pressure. This table also contains symptoms or factors observed with high blood pressure.

We have observed that a patient can experiment more complications or no complications.Each complication for the patient is considered as a new complication.

TheConceptual Data Model (CDM)of that relational database is given in figure 1.

Figure 1 : MCD of SourceHTA database

Applying rules for passing from MCD to MLD, the relation « Développer » being of type « one to more » implies that ComHTA inherits the primary key of Patients table.



In this work, we used the data contained in ComHTA. The structure o

Figure 2 : SourceHTA MLD

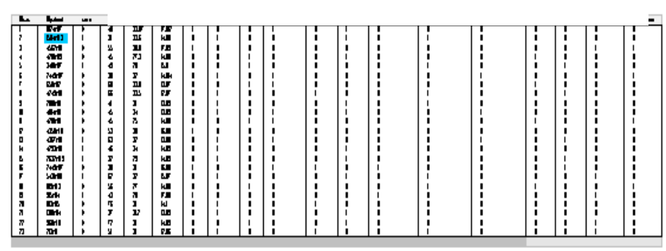


Figure 3 : structure of CompHTA

II.2. APPLICATION OF NEURAL NETWORK METHODS

Two important concepts will be presented before applying the neural network method on the database. This will give a better understanding to the reader.

II.2.1. The data exploration structure

The data exploration structure defines the domain of exploration for a given problem. It contains a list of columns for the data and their types. These columns are linked to a source of data. The data exploration structure also contains eventually attributes on the way data are modelized. It also contains operative data models.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

II.2.2. Data Operative model

A data operative model is an application of the algorithm on the data contained in the data exploration structure.

The definition of the data operative model contains an algorithm with its parameters, and a list of columns for the data exploration structure of the data.

Each data operative model within a structure can use different algorithm or a subset of different operative columns. For each of the columns in the model, one can determine its use (Predict, Predictonly, Input, Ignore, Key).

There are two type of data operative models: relationnal or OLAP. In this paper we have constructed a relational operative model.

II.2.3. Neuronal network model

Our work is based on Microsoft Business Intelligency Studio 2008 structure. It incorporates the Analysis Services module which allows performingData mining tasks.
The Analysis services module contains a lot of Data mining algorithms such us Microsoft Neuronal Network, suggested by Microsoft for Neuronal network methods.
We began by creating anAnalysis Services project and selected, as source of data, the database.

Figure 4 : Data source project choice

Then we created a data view and a data exploration structure containing the learning data for the algorithm.

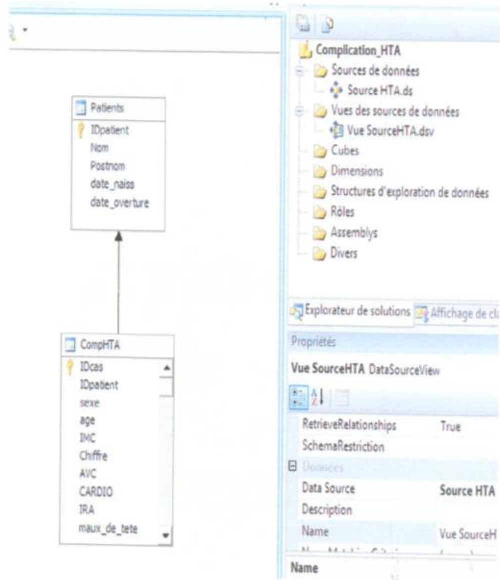
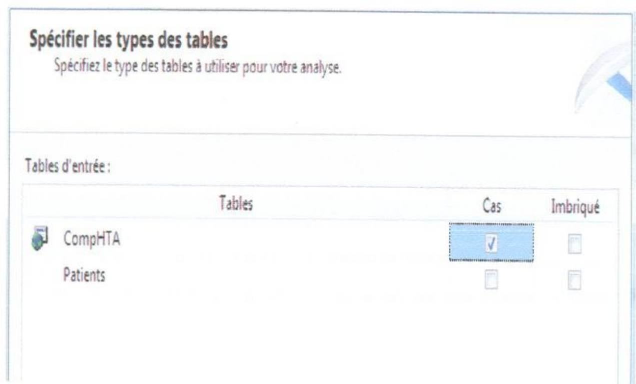
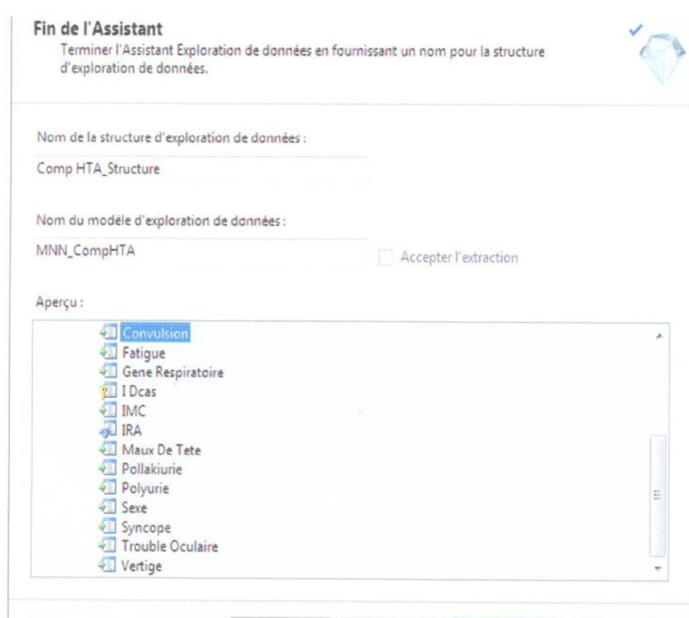


Figure 5: Data view creation

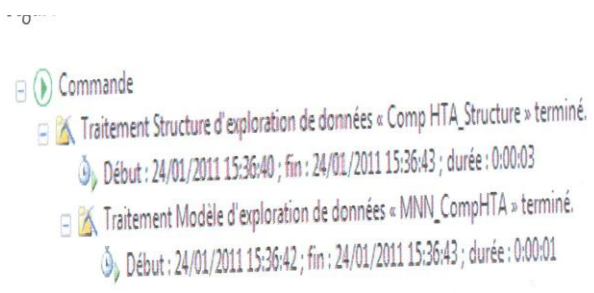
The creation of data exploration model was done by choosing Microsoft Neuronal Network as learning algorithm.





Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

Figure 7: End of structure creation and operative model



In order to know the importance of factors individually with regard to the disease complications of high blood pressure, we have applied the data structure algorithm.

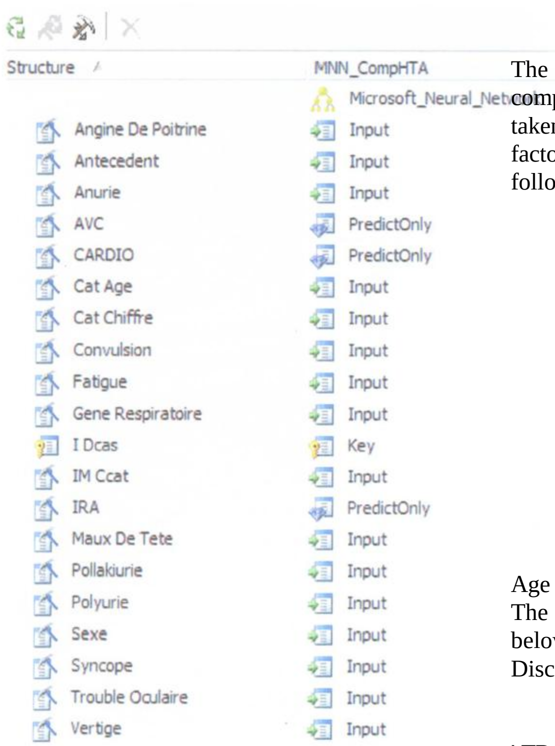


Figure 8 : Exploration model

Figure 9 : Results of Exploration model

II.3. RESULT: TABLE ON THE IMPORTANCE OF FACTORS ON THE EXHIBITION of HBP complications

The importance of each factor is the probability of a given complication of high blood pressure according to the value taken by the factor. In our case we considered only binary factors except for age, BMI and HBP which were discretized as follows:

Age → cat_Age

- 20≤Age<30 cat_Age="JeuneAdulte", 30≤Age≤65 Cat_Age="Adulte", Age>65 Cat_Age="TroisiemeAge"

IMC→Cat_IMC

- 18≤IMC≤25 Cat_IMC="Normal", 25<IMC≤30 Cat_IMC="Surpoids", Cat_IMC <18 Cat_IMC="Maigre", IMC>30 Cat_IMC="Obese"

Chiffre →chiffre> à la moyenne SUP_AVG, chiffre≤ à la moyenne INF_AVG

Age discretization
The SQL queries we used to discretize these variables are given below:
Discretization of the age column

```
SELECT (CASE WHEN (CompHTA.age > 65) THEN ' TROISIEMEAGE'
WHEN (CompHTA.age >= 20 AND CompHTA.age < 30) THEN 'JEUNEADULTE'

WHEN (CompHTA.age >=30 AND CompHTA.age <=65) THEN 'ADULTE'
ELSE 'PASCADULTE' END) AS Cat_Age FROM ComHTA
```

BMIdiscretization

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

```
SELECT (CASE WHEN (CompHTA.IMC > 30) THE 'OBESE'
WHEN (CompHTA.IMC >= 18 AND CompHTA.IMC <=25) THEN 'NORMAL'
WHEN (CompHTA.IMC >25 AND CompHTA.IMC <=30) THEN 'SURPOIDS'
ELSE 'MAIGRE' END) AS Cat_IMC FROM CompHTA
```

HBP discretization

```
SELECT (CASE WHEN (Chiffre > (SELECT AVG (Chiffre)
FROM CompHTA)) THEN 'SUP_AVG'
ELSE 'INF_AVG' END) AS Cat_Chiffre
```

In accordance with the probabilities derived by applying the algorithm, the percentageimportance in percentage terms of each factor in relation to each of the three types of complications that we considered is shown in the following table:

Nom facteur	AVG	Cardiopathie	IRA	Score
Cat_Age : jeuneAdulte	1,05	2,43	2,81	70
Cat_Age : Adulte	20,65	53,50	0,52	70
Cat_Age : TroisièmeAge	16,67	59,90	89,87	70
Cat_IMC : Normale	10,95	14,60	40,87	70
Cat_IMC : Surpoids	55,60	7,84	42,77	75
Cat_IMC : Obese	22,56	74,26	7,67	70
Cat_Chiffre : SUP_ABG	87,80	9,39	7,87	70
Cat_Chiffre : INF_AVG	1,32	17,40	52,01	70
Sexe : F	10,52	58,22	30,15	68
Sexe : M	15,70	49,12	13,25	68
Angine de poitrine	23,35	67,24	18,72	78
Antécédent	29,50	69,43	10,91	78
Anurie	2,70	6,19	83,20	90
Convulsion	56,35	58,20	37,20	90
Fatigue	4,37	90,34	2,97	78
Gêne respiratoire	3,18	47,28	4,28	85
Maux de tête	38,27	16,82	11,62	75
Pollakiurie	37,22	59,76	26,43	90
Polyurie	2,27	76,56	18,34	78
Syncope	4,56	90,43	5,56	78
Trouble oculaire	44,22	1,76	26,43	78
Vertige	7,40	37,04	29,62	80

Figure 10 : **Results found by datamining**

The last column gives the score obtained by the model operating data relating to the scoreof an ideal model which is actually a theoretical model.

II.4. Analysis of results provided by the neural networks method.

After applying the algorithm of Microsoft Neural Network on the data, the following results were obtained:

• Patients whose age ranges from 40 to 60 years are more likely to develop stroke than others.
• Patients aged 65 years and more are more likely to have heart disease or kidney failure.
• Patients with a BMI indicating overweight are more likely to have kidney failure or stroke, while those with a BMI indicating obesity are more likely to have heart diseases.
• A heart disease is more likely when an elderly has a blood pressure below the average.
• Women with high blood pressure are more likely to develop heart disease than other forms of complications of high blood pressure.
• Patients under 40 years are very few and are the least likely to present any of the complications. And most of the cases have a BMI indicatingobesity.
• Angina pectoris is a syndrome harbinger of heart disease in a hypertensive person.
• Men are more likely to develop complications due to high blood pressure.
• Males are more likely to develop stroke.
• Patients with a history of complications were 69.43% risk of developing heart disease.
• Patients with HBP (systolic and diastolic) above the average are more than 50% risk of developing a stroke.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.

CONCLUSION

• The exploration and analysis of the database about complications of high blood pressure in a hospital by multilayer neural networks, using the back-propagation algorithm, provided very interesting results in making medical decisions.

• We have shown that data mining helps to confirm a behavior or an hypothesis, by checking it through the application of artificial intelligence methods. In addition, data was searched to discover previously unknown relationships.

• From these results system can be put in place to assess risksfor a patient with high blood pressure to develop one of three types of complications studied.

• With regard to neural networks, we confirmed the ability to model complex structures andirregular data, as well as very different problems.

References

[1] **AUPETIT A** : *les réseaux de neurones artificiels ; Laboratoire d'Algorithmique 2004*.

[2] **BISHOP C.M**: *Neuronal Network for Pathern recogrition, Clarendon Press, oxford, 1995*.

[3] **CHAMBO J** : *Concepts d'informatique décisionnelle ; cours université de Provence, 2010.*

[4] **LEMAIRE V***les réseaux de neurones multicouches, cours université de Reune, 2007*.

[5] **KASENDE TSHITUTA I.M** : *Applications du Data Mining, Mémoire de Licence en Sciences Informatiques, Kinshasa, 2010*.

[6] **MACLENNAN J, TANG, CRIVAT B**, : *Data mining with Microsoft SQL Server 2008, Willey 2009.*

[7] **MBUYI MUKENDI E** : *Intelligence artificielle approfondie, Cours Université de Kinshasa, 2010*.

[8] **PEREZ-URIBE A** : *Réseau de neurones artificiels et apprentissage non-supervisé, Hieg-vd 2009*.

[9] **RAKOTOMALALA R** : *Introduction au datamining et arbre de décision, Revue Modulad n°33, 2005*.

[10] **RAKOTOMALALA R** : *Introduction au datamining cours Rouen 2007*.

[11] **SAPORTA G** : *Datamaning : une nouvelle façon de faire des statistiques, CNAM, 2002*.

[12] **SCHNEDER R. GIBSOND** : *Microsoft SQL Server 2008 for Domnies ; Willey 2008*.

[13] **SERNA ENICAS M** : *Entrepôt de données pour l'aide à la décision médicale : conception et expérimentation, Thèse de Doctorat, Université Joseph Fourier, 2005*.

[14] **TRUFFEYS** : *Data mining et Scornig ; édition Dunod, 2002*.

[15] **TRUFFEYS** : *Data mining et statistique décisionnelle, édition Technip 2007*.

[16] **VOLLEM** : *Le système informatique d'aide à la décision, Paris 2001*.

Copyright (c) 2012 International Journal of Computer Science Issues. All Rights Reserved.