ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 1-6

A Local Search Method for Pattern Classification*

A. Albrecht[1,4], M. Loomes[1], K. Steinhöfel[2], M. Taupitz[3], and C.K. Wong[4]

[1] Univ. of Hertfordshire, Dept. of CS, Hatfield, Herts AL10 9AB, UK
[2] GMD–National IT Res. Center, Kekuléstr. 7, 12489 Berlin, Germany
[3] Inst. of Radiology, Humboldt Univ., 10117 Berlin, Germany
[4] The Chinese University of Hong Kong, Shatin, N.T., Hong Kong

Abstract. We present a new method to compute depth-three threshold circuits for pattern classification problems. The first layer of the circuits is calculated from a sample set S of the classification problem by a local search strategy that minimises the error on S for each individual gate. The local search is based on simulated annealing with the logarithmic cooling schedule $c(k) = \Gamma/\ln(k+2)$. The parameter Γ depends on S and the neighbourhood relation is determined by the classical Perceptron algorithm. The approach is applied to the recognition of focal liver tumours.

1 Introduction

The paper describes method of computing depth-three threshold circuits for pattern recognition purposes. The approach is applied to focal liver tumour recognition, where the CT images are classified by the threshold circuit without any pre-processing. From a general point of view, the threshold circuits are designed for binary classifications of points from an n-dimensional space. This problem has been studied for a long time and is closely related to algorithms solving systems of linear inequalities.

AGMON proposed in 1954 [2] a simple iteration procedure to find solutions of linear inequalities $l^j(\vec{z}) = \vec{a}^j \cdot \vec{z} + b^j \geq 0$, $j = 1, \ldots, m$. In pattern recognition, AGMON's method became popular as the classical Perceptron algorithm [16]. For sets S of n-dimensional vectors $\vec{x}$ that are separable by a linear threshold function into "positive" and "negative" examples, MINSKY and PAPERT [13] proved the following convergence property: If $\vec{w}^*$ is a unit vector solution to the separation problem, then the Perceptron algorithm converges in at most $1/\sigma^2$ iterations, where $\sigma := \min_{[\vec{x},\eta]\in S} |\vec{w}^* \cdot \vec{x}|$, $\eta \in \{+,-\}$. The parameter σ can be exponentially small in terms of the dimension n.

In general, the simple Perceptron algorithm performs well even if the sample set is not consistent with any weight vector $\vec{w}$ of linear threshold functions,

*Research partially supported by the Strategic Research Programme at The Chinese University of Hong Kong under Grant No. SRP 9505, by a Hong Kong Government RGC Earmarked Grant, Ref. No. CUHK 4010/98E, and by the AIF Research Programme under Grant No. FKV 0352401N7.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 1-6

see [9]. For our problem of CT-image classification, one can hardly assume that positive and negative examples are separable by a single linear threshold function. In order to reduce the classification error, we try to compute a bounded-depth circuit consisting of linear threshold functions. The threshold functions, in particular the gates of the first level, are determined by a learning procedure from positive and negative examples S of the classification problem.

HÖFFGEN [12] has shown that finding a linear threshold function that minimises the number of misclassified examples is NP-hard in the case of arbitrary sample sets. In our approach, we utilise a combination of logarithmic simulated annealing and the Perceptron algorithm for this computationally hard minimisation problem. The combination has been studied in [6] for samples generated by non-linear threshold functions. The approach belongs to the class of local search methods [1].

To our knowledge, the first paper on learning-based methods applied to X-ray diagnosis was published by ASADA ET AL. [7]. Since then, the research has been concentrating on using commercially available neural networks for medical image classification [8, 10, 14, 15, 18].

In a number of papers, feature extraction is used in learning-based classification methods [14, 17]. In [14], for example, a high classification rate of nearly 98% is reported, where the Wisconsin breast cancer diagnosis (WBCD) database of 683 cases is taken for learning and testing. The approach is based on feature extraction from image data and uses nine visually assessed characteristics for learning and testing. Among the characteristics are the uniformity of cell size, the uniformity of cell shape, and the clump thickness.

The paper continues the research from [3]. In the present paper, we describe the computation of depth-three threshold circuits from positive and negative examples that are designed to recognise focal liver tumours. The input are fragments of CT images of size 119×119 with an 8 bit grey scale in DICOM standard format [11]. Therefore, the input size is $n = 14161$ and the input values range from 0 to 255. For the learning procedure, we used 400 positive (focal liver tumours) and 400 negative (normal liver tissue) examples. The circuits were tested on $100 + 100$ examples (different from the learning set), and we obtained a correct classification of about 94%.

2 A Simulated Annealing-Based Heuristic

We assume that rational numbers are represented by pairs of binary tuples of length d and denote the set of linear threshold functions by

$$\mathcal{F} := \bigcup_{n \geq 1} \mathcal{F}_n, \quad \text{where } \mathcal{F}_n = \Big\{ f(\vec{x}) \,:\, f(\vec{x}) = \sum_{i=1}^{n} w_i \cdot x_i \geq \vartheta_f \Big\},$$

where w_i and x_i are equal to $\pm(p_i, q_i)$ for p_i, $q_i \in \{0,1\}^d$.

The functions from $\mathcal{F}$ are used to design single-output circuits $\mathcal{C}$ of threshold functions: A circuit $\mathcal{C}$ is defined by the underlying acyclic directed graph $\mathcal{G} = [E, V]$, $E \subset V \times V$. The graph $\mathcal{G}$ has n input nodes labelled by variables x_1,

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 1-6

$\cdots$, x_n, and $|V| - n$ nodes v_f labelled by threshold functions $f \in \mathcal{F}$, where the number of incoming edges of v_f has to be consistent with the number of variables of f. Finally, one v_f is chosen as the output v_{out} of $\mathcal{C}$.

The depth of $\mathcal{C}$ is the maximum number of edges on a path from an input node x_i to the output node v_{out}. The nodes that are not input nodes are called gates. The function $F(\mathcal{C})$ computed by $\mathcal{C}$ is defined as follows: The gates of the first level output 1 or 0 depending on whether or not $\sum_{i=1}^{n} w_i \cdot x_i \geq \vartheta_f$. In the same way, the gates at higher levels have Boolean outputs only. Therefore, when all paths from input nodes to v_{out} are of the same length, the gates at level 2, 3,.. do compute Boolean threshold functions. Thus, we have $F(\mathcal{C}) : \{0,1\}^{n \cdot d} \rightarrow \{0,1\}$.

In the present paper, the maximum depth of $\mathcal{C}$ is three; circuits of depth one are simply the elements of $\mathcal{F}_n$, and in Section 3 we consider circuits of depth two and three, respectively. In our application, each of the threshold functions from the first level is equally important for the overall classification result of the depth-three circuit. Therefore, the weights at the input lines of second level functions (gates) can be normalised to the value 1 and only the threshold values depend on the sample set S.

For a given sample set S, we assume $S = \{ [\vec{x}, \eta] \}$ for $\eta \in \{+, -\}$ and $\vec{x} = (x_1, ..., x_n)$ where $x_i = (p_i, q_i)$, $p_i, q_i \in \{0,1\}^d$. Furthermore, we consider a particular number n of variables only and we take the set $\mathcal{F} := \mathcal{F}_n$ as the configuration space.

The objective of our optimisation procedure is to minimise the number $|S\Delta f|$ of misclassified examples, $S\Delta f := \{ [\vec{x}, \eta] : f(\vec{x}) < 0 \,\&\, \eta = + \text{ or } f(\vec{x}) > 0 \,\&\, \eta = - \}$. The objective function is defined by $\mathcal{Z}(f) := |S\Delta f|$, and $\mathcal{F}_{\min}(S)$ denotes the set of minimum-error solutions.

Given $f = \sum_{i=1}^{n} w_i \cdot x_i \geq \vartheta_f$, the neighbourhood relation $\mathcal{N}_f$ is suggested by the Perceptron algorithm [16] and defined by

$$w_i(f') \quad := \quad w_i \; - \; y_j \cdot x_{ij} / \sqrt{\sum_{i=1}^{n} w_i^2}, \; j \in \{1, 2, \ldots, m\}, \tag{1}$$

for all i simultaneously and for a specified j that maximises $|y_j - \vartheta_f|$, where $y_j = \sum_{i=1}^{n} w_i \cdot x_{ij}$. The threshold $\vartheta_{f'}$ is equal to $\vartheta_f + y_j / \sqrt{\sum_{i=1}^{n} w_i^2}$.

Given a pair $[f, f']$, $f' \in \mathcal{N}_f$, we denote by $G[f, f']$ the probability of generating f' from f and by $A[f, f']$ the probability of accepting f' once it has been generated from f.

To speed up the local search for minimum error solutions, we take a non-uniform generation probability where the transitions are forced into the direction of the maximum deviation. We used this approach before [4] in the context of equilibrium computations. The non-uniform generation probability

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 1-6

is derived from the Perceptron algorithm: For the current hypothesis f we set:

$$U(\vec{x}) \quad := \quad \begin{cases} -f(\vec{x}), & \text{if } f(\vec{x}) < \vartheta_f \text{ and } \eta(\vec{x}) = +, \\ f(\vec{x}), & \text{if } f(\vec{x}) \geq \vartheta_f \text{ and } \eta(\vec{x}) = -, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

For $f' \in \mathcal{N}_f$ we set $G[f,\, f'] := U(\vec{x}) / \sum_{\vec{x} \in S\Delta f} U(\vec{x})$. Thus, preference is given to the neighbours that maximise the deviation. Now, our heuristic can be summarised in the following way:

1. The initial hypothesis is defined by $w_i = 1$, $i = 1, 2, \ldots n$ and $\vartheta = 0$.
2. For the current hypothesis, the probabilities $U(\vec{x})$ are calculated; see (2).
3. To determine the next hypothesis f_k, a random choice is made among the elements of $\mathcal{N}_{f_{k-1}}$ according to the definition of $G[f,\, f']$.
4. When $\mathcal{Z}(f_k) \leq \mathcal{Z}(f_{k-1})$, we set $A[f_{k-1}, f_k] := 1$.
5. When $\mathcal{Z}(f_k) > \mathcal{Z}(f_{k-1})$, a random number $\rho \in [0, 1]$ is drawn uniformly.
6. If $A[f_{k-1}, f_k] := e^{-(\mathcal{Z}(f_k) - \mathcal{Z}(f_{k-1}))/c(k)} \geq \rho$, the function f_k is the new hypothesis. Otherwise, we return to 3 with f_{k-1}.
7. The computation is terminated after a predefined number of steps K.

Hence, instead of following unrestricted increases of the objective function, our heuristic tries to find another "initial" hypothesis when the difference of the number of misclassified examples is too large.

The crucial parameter $c(k)$ is defined by $c(k) = \Gamma / \ln(k+2)$, $k = 0, 1, \ldots$. The parameter Γ depends on the underlying energy landscape. When Γ is larger than or equal to the maximum value of the minimum escape depth from local minima, one can prove the convergence to minimum-error solutions for a more general neighbourhood relation that provides the reversibility of the configuration space. In this case, the convergence analysis from [5] indicates a time complexity of roughly $n^{\Gamma + O(1)}$, i.e., after $n^{\Gamma} + \log^{O(1)}(1/\delta)$ transitions the confidence that a minimum-error threshold function has been computed is larger than $1 - \delta$.

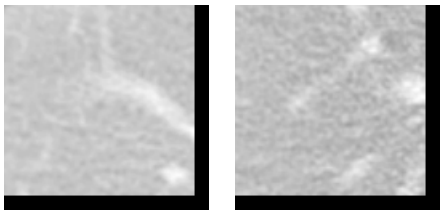
3 Implementation and Results

The heuristic was implemented in C^{++} and we performed computational experiments on SUN Ultra 5/333 workstations with 128 MB RAM. In the experiments, we used fragments of CT images of size 119×119 with 8 bit grey levels. From 400 positive (with focal liver tumours) and 400 negative examples (normal tissue) several independent hypotheses of the type $w_1 \cdot x_1 + \cdots + w_n \cdot x_n \geq \vartheta$ were calculated for $n = 14161$. We tested the hypotheses simultaneously on 100 positive and 100 negative examples. The test examples were not presented to the algorithm during the training phase.

In Figure 1 and Figure 2, typical representatives of negative and positive examples are shown. It is important to note that the CT examinations were

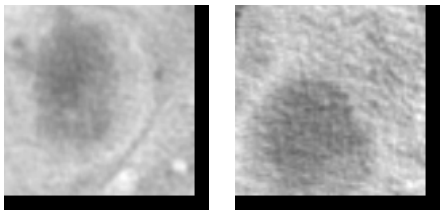
ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 1-6

performed with comparable imaging parameters, as can be seen from the brightness and contrast of the examples.



Normal liver tissue: Negative examples.

Figure 1



Focal liver tumour: Positive examples.

Figure 2

Table 1 summarises typical results for circuits of depth 1, ..., 3 Due to the long run-time, only a few experiments have been performed so far, and therefore it is difficult to present average values). Each function (gate) from the first level was trained on a random choice of 100+100 examples out of 400+400 examples. The examples were learned with zero error after about 25000 to 35000 changes of the hypothesis for a single linear threshold function (gate from the first level). The results are for $\Gamma = 60$.

The depth-three circuit consists of three sub-circuits of depth two, where each depth-two circuit has 11 threshold functions at the first layer. The output gate of the depth-three circuit is a simple majority function. Thus, the depth-three circuit consists of $3 \cdot (11 + 1) + 1 = 37$ linear threshold functions (gates). Each of the threshold functions of the first level (i.e., each input gate) has $n = 14161$ inputs, i.e., the total number of input lines that are connected to the 14161 input nodes (pixel values) is $3 \cdot 11 \cdot 14161 = 467313$.

Depth of Circuits	Learning Run-Time	Errors on POS	Errors on NEG	Errors on T_POS	Errors on T_NEG	Percentage of Errors
1	$\approx$ 275 min	0	0	32	36	34%
2	$\approx$ 3000 min	0	0	13	17	15%
3	$\approx$ 9000 min	0	0	4	8	6%

Table 1

The test on a single image from the $100 + 100$ test examples is performed within a few seconds. The parameter settings "$\Gamma = 60$", "100 out of 400", and "11" for the number of gates at the first layer of depth-two (sub-)circuits were determined by computational experiments.

We think that one reason for the good classification rate of 94% is the homogeneous imaging technique that was used to obtain the training and test material.

Acknowledgement

The authors would like to thank Eike Hein and Daniela Melzer (HUB, Institute of Radiology) for preparing the image material.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 1-6

References

[1] E.H.L. Aarts. *Local Search in Combinatorial Optimization.* Wiley & Sons, 1998.

[2] S. Agmon. The Relaxation Method for Linear Inequalities. *Canadian J. of Mathematics*, 6(3):382 – 392, 1954.

[3] A. Albrecht. On Threshold Circuit Depth. In: M. Verleysen, ed., *Proc. 3rd European Symp. on Artificial Neural Networks, ESANN'95*, pp. 211–216, Brussels, 1995.

[4] A. Albrecht, S.K. Cheung, K.S. Leung, and C.K. Wong. Stochastic Simulations of Two-Dimensional Composite Packings. *J. of Comput. Physics*, 136(2):559 – 579, 1997.

[5] A. Albrecht and C.K. Wong. On Logarithmic Simulated Annealing. In: J. van Leeuwen, O. Watanabe, M. Hagiya, P.D. Mosses, T. Ito, eds., *Theoretical Computer Science: Exploring New Frontiers of Theoretical Informatics*, pp. 301 – 314, LNCS Series, vol. 1872, 2000.

[6] A. Albrecht and C.K. Wong. Combining the Perceptron Algorithm with Logarithmic Simulated Annealing. To appear in: *Neural Processing Letters.*

[7] N. Asada, K. Doi, H. McMahon, S. Montner, M.L. Giger, C. Abe, Y.C. Wu. Neural Network Approach for Differential Diagnosis of Interstitial Lung Diseases: A Pilot Study. *Radiology*, 177:857 – 860, 1990.

[8] D.B. Fogel, E.C. Wasson III, E.M. Boughton and V.W. Porto. Evolving Artificial Neural Networks for Screening Features from Mammograms. *Artificial Intelligence in Medicine*, 14(3):317, 1998.

[9] S.I. Gallant. Perceptron-Based Learning Algorithms. *IEEE Trans. on Neural Networks*, 1(2):179 – 191, 1990.

[10] H. Handels, Th. Roß, J. Kreusch, H.H. Wolff and S.J. Pöppl. Feature Selection for Optimized Skin Tumour Recognition Using Genetic Algorithms. *Artificial Intelligence in Medicine*, 16(3):283 – 297, 1999.

[11] R. Hindel. *Implementation of the DICOM 3.0 Standard.* RSNA Handbook, 1994.

[12] K.-U. Höffgen. Computational Limitations on Training Sigmoid Neural Networks. *Information Processing Letters*, 46(6):269 – 274, 1993.

[13] M.L. Minsky and S.A. Papert. *Perceptrons.* MIT Press, Cambridge, Mass., 1969.

[14] C.A. Pea-Reyes and M. Sipper. A Fuzzy-genetic Approach to Breast Cancer Diagnosis. *Artificial Intelligence in Medicine*, 17(2):131 – 155, 1999.

[15] A.L. Ronco. Use of Artificial Neural Networks in Modeling Associations of Discriminant Factors: Towards an Intelligent Selective Breast Cancer Screening. *Artificial Intelligence in Medicine*, 16(3):299 – 309, 1999.

[16] F. Rosenblatt. *Principles of Neurodynamics.* Spartan Books, New York, 1962.

[17] C. Roßmanith, H. Handels, S.J. Pöppel, E. Rinast, and H.D. Weiss. Computer-Assisted Diagnosis of Brain Tumors Using Fractals, Texture and Morphological Image Analysis. In: H.U. Lemke, ed., *Proc. Computer-Assisted Radiology*, pp. 375 – 380, 1995.

[18] R. Tawel, T. Dong, B. Zheng, W. Qian, and L.P. Clarke. Neuroprocessor Hardware Card for Real-time Microcalcification Detection at Digital Mammography. In: *Proc. Meeting of the Radiological Society of North America*, p. 172, 1994.