UDK 681.5: 519.7

V. M. Kotsovsky, F. E. Geche, O. V. Mitsa

ARTIFICIAL COMPLEX NEURONS WITH HALF-PLANE-LIKE ACTIVATION FUNCTION

The paper deals with the problems of realization of Boolean functions on neural-like units with complex weight coefficients. The relation between classes of realizable function is considered for half-plane-like activation function. We also introduce the concept of sets separability, corresponding to our notion of neuron. The iterative online learning algorithm is proposed and sufficient conditions of its convergence are given.

Keywords: neuron, complex neuron, neural network, threshold unit, threshold function, activation function, learning, online learning.

Introduction

Artificial neural networks based on neural-like units have numerous applications in different areas, such as artificial intelligence, objects classification, pattern recognition, data compression, forecasting, approximation or extrapolation of functions of many variables and many others [1]. Different networks architectures and neuron kinds are described in [1, 2]. One of most important task in the theory of feed-forward neural networks with discrete activation functions is the one concerning the realization of a Boolean function on a single neuron. Its importance follows from the fact that for networks on the base of neurons with threshold-like activation function outputs of each network levels have two possible values (binary, bipolar, etc.). Minsky and Papert [3] proved that classical threshold units have enough weak capacity for recognition. Numerous improved models of neuron are proposed for overcome the mentioned limitations (see [1] for details).

In paper we deal with the one type of such extensions, namely complex neurons, which are introduced in [4]. There exists many way of complexification, e.g. [5].

Let $E_2 = \{-1,1\}$ be the bipolar set and E_2^n is an n-thCartesian power of E_2. A Boolean function in bipolar basis is a function mapping from E_2^n to E_2.

A Boolean function $f(x_1, \dots, x_n)$ on E_2^n is a Boolean threshold function if there exists a weight vector $(w_1, \dots, w_n) \in R^n$ and a threshold

ISSN 1562-9945

$-w_{n+1}$ such that

$$\text{for all } (x_1,\ldots,x_n)\in E_2^n \sum_{j=1}^{n} w_j x_j < -w_{n+1} \Leftrightarrow f(x_1,\ldots,x_n) = -1.$$

With intent of simplify notation we extend input and weight vectors dimension by introducing one new additional $(n+1)$-th coordinate. Let $\mathbf{x} = (x_1,\ldots,x_n,1)$, $\mathbf{w} = (w_1,\ldots,w_n,w_{n+1}) \in R^{n-1}$, $(\mathbf{w},\mathbf{x}) = \sum_{j=1}^{n} w_j x_j + w_{n+1}$ — inner product of vectors **w** and **x** (sometimes called a weighted sum). Thus, for any threshold function *f*: $f(\mathbf{x}) = \operatorname{sgn}(\mathbf{w},\mathbf{x})$, where $f(\mathbf{x}) = f(x_1,\ldots,x_n)$ and sgn is sign function given by

$$\operatorname{sgn} a = \begin{cases} -1 & \text{if } a < 0, \\ 1 & \text{if } a \ge 0. \end{cases}$$

Complex neurons

Now we extend the notion of threshold function to the complex domain. Let us consider Boolean function over alphabet $\{\alpha,\beta\}$ where α and β are complex number. Let l be an arbitrary line dividing the complex plane C on two half-plane C_+ and C_-. We may regard following sgn function

$$\operatorname{sgn}_l z = \begin{cases} -1 & \text{if } z\in C_-, \\ 1 & \text{if } z\in C_+ \cup l. \end{cases}$$

A Boolean function $f:\{\alpha,\beta\}^n \to \{\alpha,\beta\}$ is a complex Boolean threshold function (CBTF) in the alphabet $\{\alpha,\beta\}$ if there exists a complex weight vector $\mathbf{w}\in C^{n+1}$ and line l such that $f(\mathbf{x}) = \operatorname{sgn}_l(\mathbf{w},\bar{\mathbf{z}})$, where $\bar{\mathbf{z}}$ is a complex conjugate vector for **z** (here we used the definition of inner product in complex vector spaces).

Note that we do not use the notion of the threshold in our definition, because it is convenient to include the threshold in the weight vector.

It is easy to see that using rotation and fitting of the free term w_{n+1} we can restrict the class of possible sign function to the following function

$$\operatorname{Resgn} z = \begin{cases} -1 & \text{if } \operatorname{Re} z < 0, \\ 1 & \text{if } \operatorname{Re} z > 0. \end{cases}$$

Note that "small" change of term w_{n+1} allows avoiding the possibility that the weighted sum $(\mathbf{w}, \overline{\mathbf{z}})$ value lies on the division line.

Let $T_C(\alpha,\beta)$ be a class of all CBTF in alphabet $\{\alpha,\beta\}$. The question arises about relations existing among the classes of CBTF in different alphabets. The answer is given by the following proposition.

Proposition 1. *There exists an bijective correspondence between the classes* $T_C(\alpha,\beta)$ *and* $T_C(\gamma,\delta)$ *for arbitrary alphabets* $\{\alpha,\beta\}$, $\{\gamma,\delta\}$.

Proof. Let $f(z) \in T_C(\alpha,\beta)$. Then there exists $\mathbf{w} \in C^{n+1}$ such that for all $\mathbf{z} \in \{\alpha,\beta\}^n$ $f(\mathbf{z}) = \operatorname{Resgn}(\mathbf{w}, \overline{\mathbf{z}})$. The transformation $z' \to \frac{\beta-\alpha}{\delta-\gamma}(z'-\gamma)+\alpha$ is the one-one correspondence between sets $\{\gamma,\delta\}$ and $\{\alpha,\beta\}$. Then

$$(\mathbf{w}, \overline{\mathbf{z}}) = \sum_{j=1}^{n} w_j z_j + w_{n+1} = \frac{\beta-\alpha}{\delta-\gamma}\sum_{j=1}^{n} w_j z_j' + \left(\alpha - \frac{\beta-\alpha}{\delta-\gamma}\gamma\right)\sum_{j=1}^{n} w_j z_j' + w_{n+1} = (\mathbf{w}', \overline{\mathbf{z}'}),$$

where $w_j' = \frac{\beta-\alpha}{\delta-\gamma} w_j$, $(j = 1,2,\ldots n)$, $w_{n+1}' = \left(\alpha - \frac{\beta-\alpha}{\delta-\gamma}\gamma\right)\sum_{j=1}^{n} w_j + w_{n+1}$.

Let $g(\mathbf{z}')$ be a Boolean function in alphabet $\{\gamma,\delta\}$ realizable on the complex neuron with the weight vector $\mathbf{w}'$. It is easy to see that the correspondence $f \leftrightarrow g$ is bijective one between the functions from $T_C(\alpha,\beta)$ to $T_C(\gamma,\delta)$.

Note, in particular, that one cannot obtain the class of CBTF more powerful that $T_C(-1,1)$ by altering the alphabet.

The next question is how the cardinality of the class of CBTF changes if we restrict the set of possible value for weight vector coefficients. Let $T_D^n(\alpha,\beta)$ be the class of all CBTF of n variables realizable on neurons with weight vectors from the set D^{n+1}, $T_D(\alpha,\beta) = \bigcup_{n=0}^{\infty} T_D^n(\alpha,\beta)$, where $D \subseteq C$.

ISSN 1562-9945

Proposition 2. *If* $\mathrm{Re}\,\alpha \neq \mathrm{Re}\,\beta$, *then* $T_C(\alpha,\beta) = T_R(\alpha,\beta)$.

Proof. Let us proof that equality $T_C^n(\alpha,\beta) = T_R^n(\alpha,\beta)$ holds for all non-negative integer *n*. From proposition 1 it follows that $T_C^n(\alpha,\beta) \leftrightarrow T_C^n(\mathrm{Re}\,\alpha,\mathrm{Re}\,\beta)$. Let *f* be an arbitrary member of $T_C^n(\mathrm{Re}\,\alpha,\mathrm{Re}\,\beta)$, $\mathbf{z} \in \{\alpha,\beta\}^n \times \{1\}$, $z_j = x_j + iy_j$, $x_j, y_j \in R$ $(j = 1,\dots,n)$, $w_j = u_j + iv_j$, $u_j, v_j \in R$, $(j = 1,\dots,n+1)$. Then

$$f(\mathbf{z}) = \mathrm{Re}\left(\sum_{j=1}^{n} w_j x_j + w_{n+1}\right) = \sum_{j=1}^{n} u_j x_j + u_{n+1} = \mathrm{Re}\left(\sum_{j=1}^{n} u_j z_j + u_{n+1}\right).$$

It follows from the last equality that the classes $T_C^n(\mathrm{Re}\,\alpha,\mathrm{Re}\,\beta)$ and $T_R^n(\alpha,\beta)$ have the same cardinality. Then the same holds for classes $T_C(\alpha,\beta)$and $T_R(\alpha,\beta)$. Since $T_C(\mathrm{Re}\,\alpha,\mathrm{Re}\,\beta) \subseteq T_C(\alpha,\beta)$, these classes are equal.

Note that for the alphabet E_2 the last proposition is proved in [4].

From the previous proposition also follows that usage of neurons with weights belonging to the real line enable us to generate all CBTF. We will prove that similar fact is true for neurons with weights lying on any line in complex space.

Proposition 3. *If* $\gamma \in C$, $\gamma R = \{\gamma x \mid x \in R\}$ *and complex numbers* α,β,γ *satisfy conditions* $|\arg\gamma| < \frac{\pi}{2}$, $\mathrm{Re}(\alpha - \beta)\gamma \neq 0$, *then classes* $T_C(\alpha,\beta)$ and $T_{\gamma R}(\alpha,\beta)$ *coincide.*

Proof. Let us consider an arbitrary CBTF $f(\mathbf{z}) \in T_C(\alpha,\beta)$. Then there exists $\mathbf{w} \in C^{n+1}$ such that for each $\mathbf{z} \in \{\alpha,\beta\}^n$ equality $\mathrm{Resgn}(\mathbf{w},\overline{\mathbf{z}}) = f(\mathbf{z})$ is true, from which it follows that

$$(\mathbf{w},\overline{\mathbf{z}}) = \sum_{j=1}^{n} w_j z_j + w_{n+1} = \sum_{j=1}^{n} w_j \gamma^{-1} \gamma z_j + w_{n+1} = (\mathbf{w}',\overline{\mathbf{z}}'),$$

where $w'_j = w_j \gamma^{-1}$, $z'_j = \gamma z_j$ $(j = 1,\dots,n)$, $w'_{n+1} = w_{n+1}$. So, for all CBTF $f(\mathbf{z})$ inalphabet $\{\alpha,\beta\}$ there exists uniqueCBTF $g(\mathbf{z}')$ inalphabet $\{\gamma\alpha,\gamma\beta\}$ such thatfor each $\mathbf{z} \in \{\alpha,\beta\}^n$ equality $f(\mathbf{z}) = g(\mathbf{z}')$ holds. Using proposition 2 to function $g(\mathbf{z}')$ we obtain

$$\mathrm{Re}(\mathbf{w},\bar{\mathbf{z}}) = \mathrm{Re}(\mathbf{w}',\bar{\mathbf{z}}') = \mathrm{Re}\left(\sum_{j=1}^{n} u_j z_j' + u_{n-1}\right) = \mathrm{Re}\left(\sum_{j=1}^{n} u_j\gamma \cdot z_j + u_{n+1}\right) = \mathrm{Re}(\tilde{\mathbf{w}},\bar{\mathbf{z}}),$$

where $u_j \in R$, $\tilde{w}_j = u_j\gamma$, $(j = 1,\ldots,n)$, $\tilde{w}_{n+1} = \dfrac{u_{n-1}\gamma}{\mathrm{Re}\,\gamma}$. Thus, the Boolean function **f(z)** is realizable on complex neuron with weight vector $\tilde{\mathbf{w}} \in \gamma R^{n+1}$.

Learning algorithm

We have seen that $T_C(\alpha,\beta) = T_{\gamma R}(\alpha,\beta)$, and question how find some weight vector $\mathbf{w} \in T_{\gamma R}(\alpha,\beta)$, corresponding to given CBTF *f* naturally arises. That is, we need a learning algorithm for the class of CBTF.

Let A^-, A^- be two finite disjunctive subsets of vectors from the set $C^n \times \{\gamma\}$, $(\gamma \neq 0)$ (i.e. $A^+ \cap A^- = \varnothing$) and $A = A^+ \cup A^-$. We call sets A^+ and A^- γ-separable, if there exists vector $\mathbf{w} \in \gamma R^{n+1}$ such that for all $\mathbf{z} \in A$ following conditions hold

$$(\mathbf{w},\bar{\mathbf{z}}) > 0 \quad \text{if} \quad z \in A^+,$$
$$(\mathbf{w},\bar{\mathbf{z}}) < 0 \quad \text{if} \quad z \in A^-.$$

Next, we will suppose that there exists an angle ϕ and real number c such that

$$\forall \mathbf{z} \in A \quad \left|\mathrm{Re}\left(e^{i\phi}\mathbf{z}_j\right)\right| \geq c > 0 \quad (j = 1,\ldots n). \tag{1}$$

We will assume (1), without any loss of generality, because *A* is a finite set.

Let the training sample of vectors $\{\mathbf{z}^k\}$ satisfies following two conditions:

1) $\mathbf{z}^k \in A$, $k \in N$;

2) each element of the set *A* repeats in learning sample infinitely many times.

Without any loss of generalitywe will assume that $\gamma = e^{i\phi}$, where $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$. Let the initial weight vector be $\mathbf{w}^0 = (0,\ldots,0)$.Let us build the sequence of vectors $\{\mathbf{w}^k\}$ as follow:

$$\mathbf{w}^k = \mathbf{w}^{k-1} + t_k h_\phi(\mathbf{z}^k) e^{i\phi}, \tag{2}$$

where $h_\phi(\mathbf{z}) = \left(\mathrm{Re}(\overline{\mathbf{z}}_1 e^{-i\phi}),\ldots,\mathrm{Re}(\overline{\mathbf{z}}_n e^{-i\phi}),1\right)$, and a coefficient t_k in defined by

$$t_k = \begin{cases} 1 & \text{if } \mathrm{Re}\left(\mathbf{w}^{k-1}, \overline{\mathbf{z}^k}\right) \le 0 \text{ and } \mathbf{z} \in A_+, \\ -1 & \text{if } \mathrm{Re}\left(\mathbf{w}^{k-1}, \overline{\mathbf{z}^k}\right) \ge 0 \text{ and } \mathbf{z} \in A_-, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The algorithm of weights updating according to the rule (2)-(3) we call "the online learning algorithm" for the complex neural unit. The next proposition gives the sufficient condition for our learning algorithm to be convergent.

Proposition 4. *If finite sets* A^+ *and* A^- *are* γ*-separable, then there exists finite natural m such that the sequence* (2) *of weight vector, obtaining according to the rules* (2)-(3) *of online learning algorithmyield after m updates the weight vector* $\mathbf{w}^m$, *which separates sets* A^+ *and* A^-.

Proof. We do our proof by contradiction. Suppose that the opposite is true. We can assume that at each step of the learning algorithm the coefficients $t_k \neq 0$ (in opposite case we can simply throw awaysuch $\mathbf{z}^k$, for which $t_k = 0$, because weights are persistent on respective steps of the algorithm). Then $\mathbf{w}^{m+1} = t_1 h_\phi(\mathbf{z}^1)e^{i\phi} + \ldots + + t_m h_\phi(\mathbf{z}^m)e^{i\phi}$. Now find the inner product of both sides of the last equality by $\mathbf{w} \in R^{n+1}(\gamma)$, which separates sets A^+ and A^-. Without loss of generality we can assume there exists $d > 0$ such that $\forall \mathbf{z} \in A$ the following inequalityholds $\left|(\mathbf{w}, h_\phi(\mathbf{z}))\right| \ge d > 0$ (we always can satisfy it by changing in corresponding way the free term w_{n+1}). It follows from Cauchy-Schwartz inequality that

$$\|\mathbf{w}\|\cdot\|\mathbf{w}^{m+1}\|\geq\left|\left(\mathbf{w},\mathbf{w}^{m+1}\right)\right|\geq\sum_{k=1}^{m}\left|\left(\mathbf{w},h_\phi\left(\mathbf{z}^k\right)\right)\right|\geq md ,$$

and, hence,

$$\|\mathbf{w}^{m+1}\|^2\geq\frac{m^2d^2}{\|\mathbf{w}\|^2} . \tag{4}$$

In other way, if we square the both sides of (2), then we have that

$$\|\mathbf{w}^{k+1}\|^2=\|\mathbf{w}^k\|^2+2t_{k+1}\operatorname{Re}\left(\mathbf{w}^k,h_\phi\left(\mathbf{z}^{k+1}\right)e^{i\phi}\right)+\|h_\phi\left(\mathbf{z}^{k+1}\right)\|^2 .$$

Accordingly to the learning algorithm all vectors $\mathbf{w}^k$ satisfythe conditions $\mathbf{w}^k=e^{i\phi}\mathbf{u}^k$, where $\mathbf{u}^k\in\mathrm{R}^{n+1}$. Therefore,

$$\operatorname{Re}\left(\mathbf{w}^k,h_\phi\left(\mathbf{z}^{k+1}\right)e^{i\phi}\right)=\operatorname{Re}\sum_{j=1}^{n}u_j^ke^{i\phi}\operatorname{Re}\left(\overline{z_j^{k+1}}e^{-i\phi}\right)\cdot\overline{e^{i\phi}}+\operatorname{Re}\left(u_{n+1}^ke^{i\phi}\overline{e^{i\phi}}\right)=$$

$$=\operatorname{Re}\sum_{j=1}^{n}u_j^k\left(x_j^{k+1}\cos\phi-y_j^{k+1}\sin\phi\right)+u_{n+1}^k=\operatorname{Re}\left(\sum_{j=1}^{n}\operatorname{Re}\left(u_j^ke^{i\phi}\left(x_j^{k+1}+iy_j^{k+1}\right)\right)+u_{n+1}^ke^{i\phi}\overline{e^{i\phi}}\right)=$$

$$=\operatorname{Re}\left(\sum_{j=1}^{n}w_j^kz_j^{k+1}+w_{n+1}^k\overline{e^{i\phi}}\right)=\operatorname{Re}\left(\mathbf{w}^k,\overline{\mathbf{z}^{k+1}}\right) .$$

From (3) it followsthat $t_k\operatorname{Re}\left(\mathbf{w}^k,\overline{\mathbf{z}^{k+1}}\right)\leq 0$. Then, according to last equalities and condition (1) $\|\mathbf{w}^{k+1}\|^2-\|\mathbf{w}^k\|^2\leq\|h_\phi\left(\mathbf{z}^{k+1}\right)\|^2\leq nc^2+1,\ \ (k=0,1,\dots,m)$.

Let us sum the last equality by k from 0 to m. Then

$$\|\mathbf{w}^{m+1}\|^2\leq\sum_{k=0}^{m}\|h_\phi\left(\mathbf{z}^{k+1}\right)\|^2\leq(m+1)\left(nc^2+1\right) . \tag{5}$$

Inequalities (4) and (5) contradict for sufficiently large m. Hence, the learning process (2)-(3) cannot last infinitely long.

Conclusion

Artificial complex neurons with the half-plane surface of activation function are enough simple and powerful computational units. Main our results concerning complex neurons with Resgn activation function are following:

 ISSN 1562-9945

1. The choice of the alphabet of Boolean functions representation has no importance for representative power of class of respective realizable Boolean functions.

2. The restriction of possible weights to ones on an almost every line in complex plane does not shrink the class of respective complex Boolean threshold functions.

3. Neurons with restricted weights can be learned by using perceptron-like learning technique.

REFERENCES

1. Haykin S. Neural Networks. A Comprehensive Foundation, Upper Saddle River, NJ: Prentice Hall, Inc., 1999.
2. Руденко О. Г., Бодянський Є. В. Штучні нейронні мережі. — Харків: ТОВ «Компанія СМІТ», 2006. — 404 с.
3. Минский М., Пейперт С. Персептроны. — М.: Мир, 1971. — 262 с.
4. Aizenberg N.N. and Ivaskiv Yu.L. Multiple-Valued Threshold Logic. Naukova Dumka Publisher House, Kiev, 1977 (in Russian).
5. Aizenberg N.N., Aizenberg I.N., Krivosheev G.A. CNN based on Universal Binary Neurons: Learning algorithm with Error-Correction and Application to Impulsive-Noise Filtering on Gray-Scale Images // Proceedings of the Fourth International Workshop on Cellular Neural Networks and their Applications, Seville, Spain, June 24-26, 1996. - pp. 309-314.