On the Power of Boolean Computations in Generalized RBF Neural Networks

Frauke Friedrichs
Institut für Neuroinformatik
Ruhr-Universität Bochum, D–44780 Bochum, Germany
Frauke.Friedrichs@neuroinformatik.ruhr-uni-bochum.de

Michael Schmitt
Lehrstuhl Mathematik und Informatik, Fakultät für Mathematik
Ruhr-Universität Bochum, D–44780 Bochum, Germany
mschmitt@lmi.ruhr-uni-bochum.de

Abstract

Generalized radial basis function (RBF) neurons are extensions of the RBF neuron model where the Euclidean norm is replaced by a weighted norm. We study binary-valued variants of generalized RBF neurons and compare their computational power in the Boolean domain with linear threshold neurons. As one of the main results, we show that generalized binary RBF neurons with any weighted norm can compute every Boolean function that is computed by a linear threshold neuron. While this inclusion turns into an equality if the RBF neuron uses the Euclidean norm, we exhibit a weighted norm where the inclusion is proper. Applications of the results yield bounds on the Vapnik-Chervonenkis (VC) dimension of RBF neural networks with binary inputs.

Keywords: radial basis functions, feedforward neural networks, Boolean functions, computational power, Vapnik-Chervonenkis dimension

1 Introduction

As far as the history of neuron models can be traced back, the use of binary values for the representation of neural activity has played a fundamental role (see, e.g., [8, 7, 11, 9]). One of many reasons for this is certainly that assuming a simple neural code might be most helpful when trying to understand the computational capabilities of nervous systems. Consequently, the question of which

Boolean functions can be computed by neurons has always been a major subject of theoretical research. For the linear threshold neuron, a model that originated from the work by McCulloch and Pitts [8], comprehensive answers had been given already by the end of the 1960s [10]. Since then, a considerable number of new neuron models has arisen. Among these, the radial basis function (RBF) neuron has become one of the standard models in artificial neural networks (see, e.g., [4, 6]).

We continue the research on the Boolean computational power of neuron models by studying binary-valued variants of the RBF neuron. The generally real-valued RBF neuron is turned into a model for computing binary-valued functions by comparing the output value with some threshold. In this way, we introduce the Euclidean binary RBF neuron and the generalized binary RBF neuron. The latter is an extension of the former where the Euclidean norm is replaced by a weighted norm, defined in terms of some arbitrary symmetric, positive definite matrix. The generalized binary RBF neuron can be considered as the counterpart of the linear threshold neuron, which is the binary-valued relative of the sigmoidal neuron. While the continuous RBF neuron and the sigmoidal neuron are in the real-valued input domain quite different in nature, this is not obvious when both input and output values are binary.

In this article, we ask how the generalized binary RBF neuron is related to the linear threshold neuron with regard to the computation of Boolean functions. We present three results concerning this question. At first, we show that the generalized binary RBF neuron is at least as powerful as the linear threshold neuron: Every Boolean function computed by the linear threshold neuron can also be computed by the generalized binary RBF neuron, regardless of the norm employed by the latter. While this fact might be evident for some weighted norms, the emphasis of this result lies in showing that it is valid for any given norm. Moreover, the proof is constructive by providing a procedure for calculating the parameters of the RBF neuron from the parameters of the linear threshold neuron. Secondly, we exhibit a specific Boolean function that is computed by the generalized binary RBF neuron, but cannot be computed by the linear threshold neuron. (In the two-dimensional case, this function is the parity function.) Thus, generalized binary RBF neurons exist that are strictly more powerful than linear threshold neurons. A necessary condition for this to hold is revealed by the third result: If the matrix represents the Euclidean norm (that is, if it is the identity matrix), both neuron models compute the same class of Boolean functions. These results are given in Section 3.

As an immediate consequence, it follows that in any neural network, any linear threshold neuron that receives only binary inputs can be replaced by a generalized binary RBF neuron having any norm, without decreasing the computational power of the network. In Section 4 we exploit this fact for deriving bounds on the Vapnik-Chervonenkis (VC) dimension of RBF neural networks with binary inputs. This dimension, introduced by Vapnik and Chervonenkis

[13], is a combinatorial characterization of the diversity of functions that can be computed by a given neural architecture. It is of major interest in connection with computational theories of learning [1, 3] where it can be used to obtain estimates for the number of examples that must be presented to a learning algorithm in order for the outcome to have good generalization capabilities. We study the Boolean VC dimension of RBF networks, that is, the VC dimension of RBF networks with binary inputs. The results of Section 3 imply that the Boolean VC dimension of a neural network remains unchanged when linear threshold neurons are replaced by Euclidean binary RBF neurons, and vice versa. Further, when linear threshold neurons are replaced by generalized binary RBF neurons, the Boolean VC dimension does not decrease. As a consequence of this fact and a previous bound for networks of linear threshold neurons due to Bartlett [2], we obtain that the standard RBF architecture with n input nodes, k generalized binary RBF neurons in one hidden layer, and one linear threshold neuron as output node has Boolean VC dimension no less than $nk+1$, that is, it is at least linear in the number of network parameters.

Finally, we show in Section 5 that the results on the Boolean computational power of generalized binary RBF neurons can be used to derive bounds on the Boolean VC dimension of networks with standard, that is, real-valued, RBF neurons, albeit in a more indirect way. Based on ideas from Bartlett [2], we provide an explicit construction showing that RBF neural networks with n inputs, k generalized Gaussian RBF neurons with any weighted norm in a single hidden layer, and one linear threshold neuron as output node have Boolean VC dimension at least $nk+1$. This is a new bound for generalized RBF networks with Boolean inputs, see [12] for bounds concerning Euclidean RBF networks with unconstrained inputs. In Section 2 we give the definitions of the mathematical concepts used in this article, that is, the formal neuron models, the network architectures, and the Boolean VC dimension.

2 Network Architecture, Neuron Types, and Boolean VC Dimension

We consider feedforward networks with one hidden layer that are fully connected between adjacent layers (see Fig. 1). The output layer of these networks is always assumed to consist of a single linear threshold neuron. In the following, we define the neuron types that are the candidates for the nodes in the hidden layer. We use $\{0,1\}$ to denote the Boolean values.

- **Linear threshold neuron**

  The *linear threshold neuron* with n inputs $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ has as parameters a weight vector $w = (w_1, \ldots, w_n) \in \mathbb{R}^n$ and a threshold $\theta \in \mathbb{R}$.

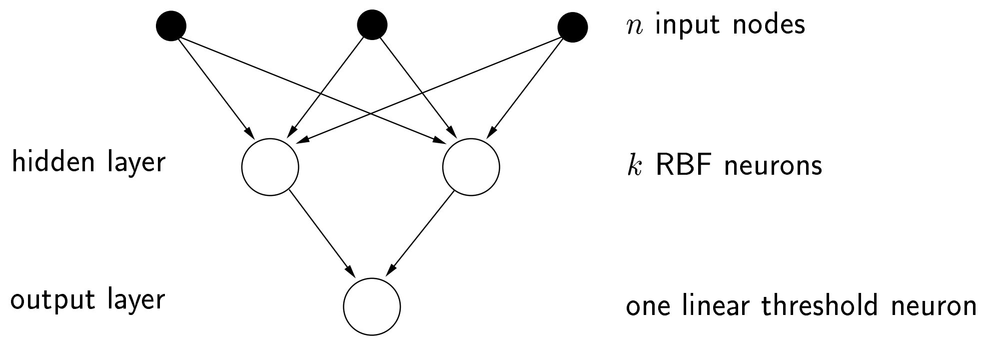

Figure 1: A standard RBF neural network architecture.

It computes functions $f_{w,\theta} : \mathbb{R}^n \to \{0,1\}$ of the form

$$f_{w,\theta}(x) \quad = \quad \operatorname{sgn}(x^T w - \theta),$$

where $\operatorname{sgn} : \mathbb{R} \to \{0,1\}$ satisfies $\operatorname{sgn}(y) = 0$ if $y < 0$, and $\operatorname{sgn}(y) = 1$ otherwise.

- **Generalized binary radial basis function neuron**

  The parameters of the *generalized binary radial basis function (RBF) neuron* are the center $c = (c_1, \ldots, c_n) \in \mathbb{R}^n$ and the radius $r \geq 0$. The functions $g_{c,r} : \mathbb{R}^n \to \{0,1\}$ computed by this neuron[1] are defined by

  $$g_{c,r}(x) \quad = \quad \begin{cases} 1 & \text{if } \|x - c\|_A \leq r, \\ 0 & \text{otherwise,} \end{cases}$$

  where A is some arbitrary real matrix that is symmetric and positive definite, and $\|\cdot\|_A$ is the weighted norm $\|y\|_A = (y^T A y)^{1/2}$. If A is the identity matrix, the neuron is referred to as the *Euclidean binary RBF neuron*.

- **Generalized Gaussian RBF neuron**

  The *generalized Gaussian RBF neuron* has a center $c = (c_1, \ldots, c_n) \in \mathbb{R}^n$ and a width $\sigma > 0$ as parameters. It computes functions $h_{c,\sigma} : \mathbb{R}^n \to (0,1]$ with

  $$h_{c,\sigma}(x) = \exp\left(-\frac{\|x - c\|_A^2}{\sigma^2}\right).$$

[1] In calling this neuron an RBF neuron we follow the widely accepted definition that radial basis functions are radially symmetric, that is, their value depends only on the (weighted) distance of the input vector from some center.

In the neuron models that employ the weighted norm $\|\cdot\|_A$, we consider the entries of the matrix A as fixed and not as part of the adjustable parameters.

The notion of shattering provides the basis for the definition of the VC dimension.

Definition 1. *A class of binary-valued functions $\mathcal{F}$ is said* to shatter *a set $S \subseteq \mathbb{R}^n$ if every dichotomy of S is induced by $\mathcal{F}$, that is, if for every (S^-, S^+), where $S^- \cap S^+ = \emptyset$ and $S^- \cup S^+ = S$, there is some $f \in \mathcal{F}$ such that $f(S^-) \subseteq \{0\}$ and $f(S^+) \subseteq \{1\}$.*

Definition 2. *The* Vapnik-Chervonenkis (VC) dimension *of a class $\mathcal{F}$ of binary-valued functions is the cardinality of the largest set that is shattered by $\mathcal{F}$. If there is no such set, the VC dimension is infinite.*

We define the Boolean VC dimension of a neural network as the VC dimension of the class of Boolean functions that is computed by the network.

Definition 3. *Let $\mathcal{N}$ be a neural network that computes binary-valued functions. The* Boolean VC dimension *of $\mathcal{N}$, denoted* $\mathrm{BVCdim}(\mathcal{N})$*, is the VC dimension of the class of functions that is computed by $\mathcal{N}$ and restricted to Boolean inputs.*

3 Linear Threshold Neurons vs. Generalized Binary RBF Neurons

In the following, we compare generalized binary RBF neurons with linear threshold neurons. We show first that the generalized binary RBF neuron is able to compute all Boolean functions that are computed by the linear threshold neuron. This holds for any arbitrary weighted norm. We emphasize that the proof is constructive, that is, it provides a method for computing the parameters of the RBF neuron from the parameters of the linear threshold neuron.

Theorem 1. *Let $f : \{0,1\}^n \to \{0,1\}$ be some arbitrary Boolean function and A some arbitrary symmetric, positive definite $(n \times n)$-matrix. If f is computed by a linear threshold neuron then f is also computed by a generalized binary RBF neuron with norm $\|\cdot\|_A$.*

Proof. The proof is based on geometric arguments. Let $f : \{0,1\}^n \to \{0,1\}$ be some Boolean function that is computed by a linear threshold neuron. Let $P, N \subseteq \{0,1\}^n$ be defined as

$$\begin{aligned} P &= \{x \in \{0,1\}^n : f(x) = 1\}, \\ N &= \{x \in \{0,1\}^n : f(x) = 0\}. \end{aligned}$$

Since f is computed by a linear threshold neuron, there is a decision boundary in form of a hyperplane that separates the sets P and N. Without loss of generality we may assume that no elements of $\{0,1\}^n$ lie on this hyperplane. (Otherwise we consider a hyperplane that is slightly shifted in the direction of the halfspace that contains N.) Let the neuron have weight vector $w \in \mathbb{R}^n$ and threshold θ, such that the hyperplane H is given by

$$H = \{x : x^T w - \theta = 0\}.$$

Then, with

$$\begin{aligned} H_1 &= \{x \in \mathbb{R}^n : x^T w - \theta > 0\} \\ H_0 &= \{x \in \mathbb{R}^n : x^T w - \theta < 0\} \end{aligned}$$

we have that

$$P \subset H_1, \; N \subset H_0.$$

Suppose that we have a generalized binary RBF neuron with symmetric and positive definite matrix A. We must show that there is a center $c \in \mathbb{R}^n$ and a radius $r \geq 0$ so that all elements of P are contained in the ellipsoid specified by A (including the boundary of the ellipsoid), and all elements of N lie outside of it.

First, we transform $\mathbb{R}^n$ using an affine mapping. It is well known that affine transformations map hyperplanes to hyperplanes, halfspaces to halfspaces, and spheres and ellipsoids to ellipsoids. We choose the transformation such that the following two conditions are met:

- The hyperplane H is mapped to a hyperplane that contains the origin of the coordinate system.
- Any ellipsoid defined by the matrix A is mapped to a sphere.

Since A is symmetric and positive definite, $A^{1/2}$ exists and is invertible (see, e.g., [5]). Let $a \in H$ be some arbitrary point of the hyperplane (see Fig. 2). The affine transformation

$$T_1(x) = A^{1/2}x - A^{1/2}a$$

satisfies both conditions, because we have $T_1(a) = 0$, and the ellipsoid given by

$$\{x : (x-c)^T A (x-c) \leq r^2\}$$

with center c and radius r is mapped to the sphere with center $-A^{1/2}(a-c)$ and radius r,

$$\begin{aligned} &T_1(\{x : (x-c)^T A(x-c) \leq r^2\}) \\ &\quad = \{y : (A^{-1/2}y + a - c)^T A (A^{-1/2}y + a - c) \leq r^2\} \\ &\quad = \{y : (y + A^{1/2}(a-c))^T (y + A^{1/2}(a-c)) \leq r^2\}. \end{aligned}$$

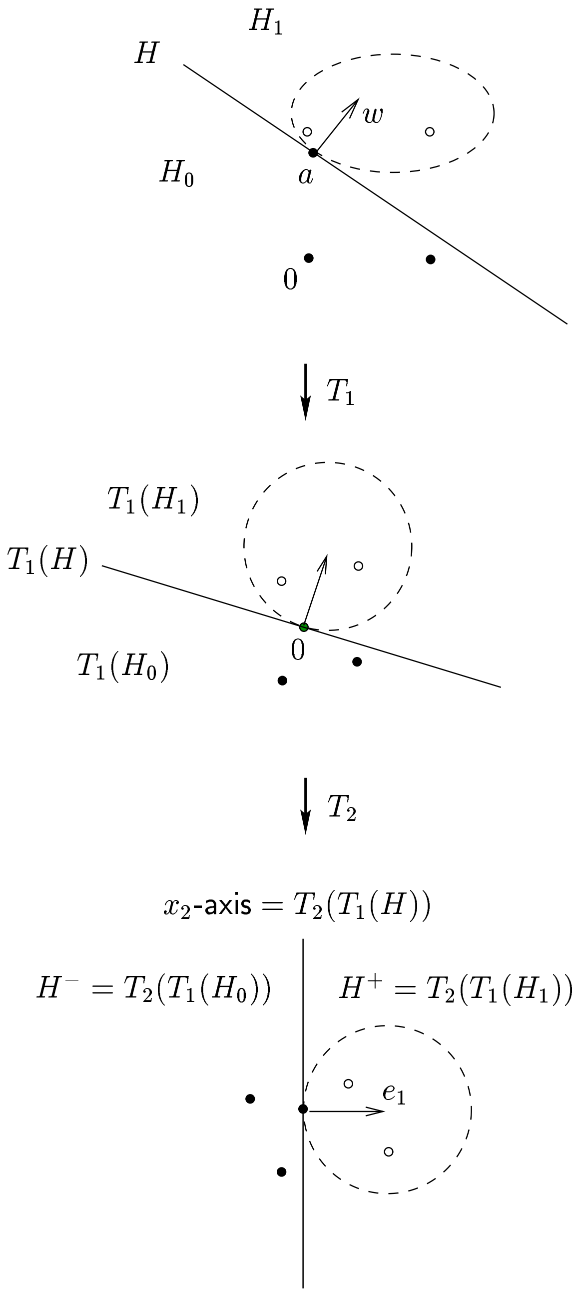

Figure 2: The transformations T_1 and T_2 used in the proof of Theorem 1 for the case $n = 2$. The white dots in the top part represent the set P, the black dots the set N. See the text for further explanations.

Here, we have used the substitution $y = T_1(x)$ in the first and the fact that $A^{1/2}$ is symmetric in the second equation.

To simplify things further, we consider a second affine transformation T_2 that maps the halfspace $T_1(H_1)$ to the halfspace

$$H^+ = \{(x_1, \ldots, x_n) : x_1 > 0\}.$$

We choose T_2 as a rotation that maps the normal vector of the hyperplane $T_1(H)$ which points to the interior of the halfspace $T_1(H_1)$ (a simple computation shows that this is the vector $A^{-1/2}w$) to the unit vector $e_1 = (1, 0, \ldots, 0)$ (see the lower part of Fig. 2). The affine transformation T_2 is an orthogonal transformation that maps spheres to spheres.

We are interested in the image $T_2(T_1(P)) \subset H^+$. We aim at finding a sphere that contains $T_2(T_1(P))$ but does not intersect the halfspace $H^- = \{x : x_1 < 0\}$. For this purpose we consider spheres with center $(r, 0, \ldots, 0)$ and radius r. Obviously, these spheres do not intersect H^-. Moreover, an element $x \in H^+$ lies inside these spheres when we select a radius r greater than $\|x\|_2^2/(2x_1)$:

$$\begin{aligned}
\|x - (r, 0, \ldots, 0)\|_2^2 < r^2 &\iff \|(x_1 - r, x_2, \ldots, x_n)\|_2^2 < r^2 \\
&\iff r^2 - 2x_1 r + \|x\|_2^2 < r^2 \\
&\overset{x_1>0}{\iff} r > \frac{\|x\|_2^2}{2x_1}.
\end{aligned}$$

Since $T_2(T_1(P)) \subset H^+$ is finite, there is a radius r_P such that this set is contained in the sphere with center $(r_P, 0, \ldots, 0)$ and radius r_P. The preimage of this sphere with regard to the affine transformation $T_2 \circ T_1$ is an ellipsoid given by the matrix A that contains P and does not intersect the halfspace H_0. In particular, it does not contain N. In other words, the function of the generalized binary RBF neuron that maps this ellipsoid to 1 and all elements outside the ellipsoid to 0 computes the Boolean function f when restricted to $\{0, 1\}^n$. □

The previous result raises the question whether the established inclusion is strict. In fact, in the following we exhibit a Boolean function that is computed by a generalized binary RBF neuron, but cannot be computed by a linear threshold neuron. It follows that generalized binary RBF neurons can be more powerful than linear threshold neurons.

Lemma 2. *Let $n \geq 2$ and $e_1, \ldots, e_n \subseteq \mathbb{R}^n$ be the unit vectors in $\mathbb{R}^n$ (where e_i has 1 in the i-th component and 0 elsewhere). Consider the Boolean function $f : \{0, 1\}^n \to \{0, 1\}$ that is 1 on every e_i, and 0 on every element of $\{0, 1\}^n \setminus \{e_1, \ldots, e_n\}$. There is a generalized binary RBF neuron that computes f, but f cannot be computed by a linear threshold neuron.*

Proof. First, we specify a generalized binary RBF neuron that computes the function f. Let the matrix A be defined by

$$A = B + C = \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & & \vdots \\ 1 & \cdots & 1 \end{pmatrix} + \begin{pmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{pmatrix}.$$

Obviously, A is symmetric and it is also positive definite, because for every $x \neq 0$,

$$x^T A x \;=\; x^T B x + x^T C x \;=\; \Big(\sum_{i=1}^{n} x_i\Big)^2 + \sum_{i=1}^{n} x_i^2 \;>\; 0.$$

A generalized binary RBF neuron with this matrix computes the function f when choosing $c = (1/n, \ldots, 1/n)$ as center and $r = 1$ as radius. In order to see this, let $v \in \{0,1\}^n$ and k be the number of non-zero entries of v. Then, we get

$$\begin{aligned}
\|v - c\|_A^2 &= \|v - c\|_B^2 + \|v - c\|_C^2 \\
&= \Big(\sum_{i=1}^{n} \Big(v_i - \frac{1}{n}\Big)\Big)^2 + \sum_{i=1}^{n} \Big(v_i - \frac{1}{n}\Big)^2 \\
&= \Big(k\Big(1 - \frac{1}{n}\Big) - (n-k)\frac{1}{n}\Big)^2 + k\Big(1 - \frac{1}{n}\Big)^2 + (n-k)\frac{1}{n^2} \\
&= \Big(k - \frac{k}{n} - 1 + \frac{k}{n}\Big)^2 + k - \frac{2k}{n} + \frac{k}{n^2} + \frac{1}{n} - \frac{k}{n^2} \\
&= (k-1)^2 + k\Big(1 - \frac{2}{n}\Big) + \frac{1}{n}.
\end{aligned}$$

For all unit vectors $v = e_i, 1 \leq i \leq n$, we have $k = 1$. Thus, they satisfy

$$\|e_i - c\|_A^2 \;=\; \Big(1 - \frac{2}{n}\Big) + \frac{1}{n} \;=\; 1 - \frac{1}{n} \;<\; 1.$$

Hence, they are mapped to 1 by the generalized binary RBF neuron with the parameters defined above. For all vectors $v \in \{0,1\}^n \setminus \{e_1, \ldots, e_n\}$ we have due to $k \neq 1$,

$$\|x - c\|_A^2 \;=\; \underbrace{(k-1)^2}_{\geq 1} + \underbrace{k\Big(1 - \frac{2}{n}\Big)}_{\geq 0} + \frac{1}{n} \;\geq\; 1 + \frac{1}{n} \;>\; 1.$$

That is, they are mapped to 0. Thus, the generalized binary RBF neuron with matrix A computes the Boolean function f.

Finally, we show that a linear threshold neuron cannot compute f. Assume that the opposite holds. The decision boundary that corresponds to the appropriate linear threshold function is a hyperplane that divides $\mathbb{R}^n$ into two disjoint halfspaces H_0 and H_1. The halfspaces are convex, that is, for all $x, y \in H_0$ we have

$$\{tx + (1-t)y \mid 0 \leq t \leq 1\} \subseteq H_0,$$

and the same for H_1. The vectors $\bar{0} = (0, \ldots, 0)$ and $\bar{1} = (1, \ldots, 1)$ are in H_0. The convexity of H_0 implies that

$$\left(1 - \frac{1}{n}\right)\bar{0} + \frac{1}{n}\bar{1} = \left(\frac{1}{n}, \ldots, \frac{1}{n}\right) \in H_0.$$

Since H_1 is convex and $e_1, \ldots, e_n \in H_1$, we have

$$\frac{1}{n}e_1 + \cdots + \frac{1}{n}e_n = \left(\frac{1}{n}, \ldots, \frac{1}{n}\right) \in H_1,$$

a contradiction. □

We show next that a function such as provided by the previous lemma can exist only if the norm of the generalized binary RBF neuron is different from the Euclidean norm. In other words, the Euclidean binary RBF neuron and the linear threshold neuron are equivalent with respect to the computation of Boolean functions. To derive this equivalence we first establish a result that holds for more general input domains that are real-valued and may even have infinitely many elements: We show that linear threshold neurons and generalized binary RBF neurons compute on elliptical surfaces the same class of functions.

Theorem 3. *Let A be a symmetric and positive definite $(n \times n)$-matrix. Further, let $m \in \mathbb{R}^n$, $\rho > 0$, and $E(m, \rho) = \{x \in \mathbb{R}^n : \|x - m\|_A = \rho\}$. Suppose that $M \subseteq E(m, \rho)$. Then for every $f : M \to \{0, 1\}$, f can be computed by a linear threshold neuron if and only if f can be computed by a generalized binary RBF neuron with norm $\|\cdot\|_A$.*

Proof. Assume first that $f : M \to \{0, 1\}$ is computed by a linear threshold neuron with weight vector $w \in \mathbb{R}^n$ and threshold $\theta \in \mathbb{R}$, that is, for every $x \in M$,

$$f(x) \;=\; \mathrm{sgn}(x^T w - \theta).$$

Let $x \in M \subseteq E(m, \rho)$. In the following, we use the fact that every symmetric, positive definite matrix A has an inverse A^{-1} (see, e.g., [5]). Then, we have

$$
\begin{aligned}
& f(x) = 1 \\
\iff \quad & x^T w \;\geq\; \theta \\
\iff \quad & -2x^T w \;\leq\; -2\theta \\
\overset{x\in E(m,\rho)}{\iff} \quad & \|x-m\|_A^2 - 2x^T w + 2m^T w + \|A^{-1}w\|_A^2 \\
& \qquad\qquad\qquad \leq\; \rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 \\
\iff \quad & (x-m)^T A(x-m) - 2(x-m)^T w + (A^{-1}w)^T A A^{-1} w \\
& \qquad\qquad\qquad \leq\; \rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 \\
\iff \quad & (x-m)^T A(x-m) - 2(x-m)^T A A^{-1} w + (A^{-1}w)^T A A^{-1} w \\
& \qquad\qquad\qquad \leq\; \rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 \\
\overset{A \text{ symmetric}}{\iff} \quad & (x-m-A^{-1}w)^T A(x-m-A^{-1}w) \\
& \qquad\qquad\qquad \leq\; \rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 \\
\iff \quad & \|x-(m+A^{-1}w)\|_A^2 \;\leq\; \rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 .
\end{aligned}
$$

Suppose that the right-hand side of the last inequality satisfies $\rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 \geq 0$. Then we choose the square root of this term as the radius for the generalized binary RBF neuron. In this case, the function f is computed by a generalized binary RBF neuron with center $c = m + A^{-1}w$ and radius $r = (\rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2)^{1/2}$.

On the other hand, if $\rho^2 - 2\theta + 2m^T w + \|A^{-1}w\|_A^2 < 0$, the function f is always zero and we define the center by $c = m$ and the radius by some $r < \rho$.

Now, let $f : M \to \{0,1\}$ be computed by a generalized binary RBF neuron with norm $\|\cdot\|_A$ having center $c \in \mathbb{R}^n$ and radius $r \geq 0$ such that for every $x \in M$,

$$
f(x) \;=\; \begin{cases} 1 & \text{if } \|x-c\|_A \leq r, \\ 0 & \text{otherwise.} \end{cases}
$$

Let $x \in M \subseteq E(m,\rho)$. Here, we have

$$
\begin{aligned}
&f(x) = 1 \\
&\iff \quad \|x-c\|_A \ \le\ r \\
&\overset{r\ge 0}{\iff} \quad \|x-c\|_A^2 \ \le\ r^2 \\
&\overset{x\in E(m,\rho)}{\iff} \quad \|x-c\|_A^2 - \|x-m\|_A^2 \ \le\ r^2-\rho^2 \\
&\iff \quad (x-c)^T A(x-c) - (x-m)^T A(x-m) \ \le\ r^2-\rho^2 \\
&\overset{A \text{ symmetric}}{\iff} \quad x^T Ax - 2x^T Ac + c^T Ac - x^T Ax + 2x^T Am - m^T Am \ \le\ r^2-\rho^2 \\
&\iff \quad -2x^T Ac + 2x^T Am \ \le\ r^2-\rho^2-\|c\|_A^2+\|m\|_A^2 \\
&\iff \quad -2x^T A(c-m) \ \le\ r^2-\rho^2-\|c\|_A^2+\|m\|_A^2 \\
&\iff \quad x^T A(c-m) \ \ge\ \frac{-r^2+\rho^2+\|c\|_A^2-\|m\|_A^2}{2}.
\end{aligned}
$$

Thus, the function f is computed by the linear threshold neuron with weight vector $w = A(c-m)$ and threshold $\theta = (-r^2+\rho^2+\|c\|_A^2-\|m\|_A^2)/2$. □

As a consequence of this theorem, we obtain that linear threshold neurons and Euclidean binary RBF neurons compute the same class of Boolean functions.

Corollary 4. *Let $f : \{0,1\}^n \to \{0,1\}$ be some arbitrary Boolean function. Then f is computed by a linear threshold neuron if and only if f is computed by a Euclidean binary RBF neuron.*

Proof. We employ Theorem 3. The Euclidean binary RBF neuron is obtained from the generalized binary RBF neuron by choosing the identity matrix for A. The elements of $\{0,1\}^n$ are all located on the sphere with center $m = (1/2,\dots,1/2)$ and radius $\rho = \sqrt{n}/2$. □

4 On the Boolean VC Dimension of Binary RBF Networks

In this section we consider neural networks that have generalized binary RBF neurons as hidden nodes. We establish bounds on the Boolean VC dimension of these networks taking advantage of the fact that such bounds are already known for networks with linear threshold neurons instead of RBF neurons. The results of the previous section immediately lead to the following relations concerning the Boolean VC dimension of these neural network types.

Theorem 5. *Let $\mathcal{N}$, $\mathcal{N}'$, and $\mathcal{N}''$ be feedforward neural networks with the same architecture and one linear threshold neuron as output node. Assume that the* *hidden nodes of $\mathcal{N}$, $\mathcal{N}'$, and $\mathcal{N}''$ are linear threshold neurons, Euclidean binary RBF neurons, and generalized binary RBF neurons with arbitrary, fixed, symmetric, and positive definite matrices, respectively. Then*

$$\mathrm{BVCdim}(\mathcal{N}) \;=\; \mathrm{BVCdim}(\mathcal{N}') \;\leq\; \mathrm{BVCdim}(\mathcal{N}'').$$

Proof. By Corollary 4, linear threshold neurons compute the same class of Boolean functions as Euclidean binary RBF neurons. Hence, the equality in the statement follows. Theorem 1 shows that a generalized RBF neuron computes every Boolean function a linear threshold gate can compute. This implies the inequality. □

As a consequence of this result, we obtain a lower bound on the Boolean VC dimension for neural networks with generalized binary RBF neurons in terms of the number of parameters. Note that this statement includes Euclidean binary RBF neurons as a special case of the generalized binary RBF neurons.

Corollary 6. *Let $\mathcal{N}$ be a neural network with n input nodes, one linear threshold neuron as output neuron, and k generalized binary RBF neurons in a single hidden layer. Assume that each RBF neuron has an arbitrary, fixed matrix that is symmetric and positive definite. Suppose that $n \geq 3$ and $k \leq 2^{n+1}/(n^2+n+2)$. Then*

$$\mathrm{BVCdim}(\mathcal{N}) \geq nk + 1.$$

Proof. The statement follows from Theorem 5 and the lower bound $nk+1$ for networks of linear threshold neurons with one hidden layer provided by Theorem 8 in [2] (see also Theorem 6.2 in [1]). □

5 On the Boolean VC Dimension of Generalized Gaussian RBF Networks

Now, we use an explicit construction based on linear threshold neurons and a result from Section 3 to show that networks with generalized Gaussian RBF neurons satisfy the lower bound that was established in the previous section for networks with binary RBF neurons.

Theorem 7. *Let $\mathcal{N}$ be a neural network with n input nodes, one linear threshold neuron as output node, and k generalized Gaussian RBF neurons in a single hidden layer. Assume that each RBF neuron has an arbitrary, fixed matrix that is symmetric and positive definite. Suppose that $n \geq 3$ and $k \leq 2^{n+1}/(n^2+n+2)$. Then*

$$\mathrm{BVCdim}(\mathcal{N}) \geq nk + 1.$$

Proof. We show that the network shatters the set defined in the proof of Theorem 8 in [2] (see also Theorem 6.2 in [1]). There, a set $M \subseteq \{0,1\}^n$ of cardinality at least $2^{n+1}/(n^2+n+2)$ is constructed such that any two different elements of M differ in at least three components. Let $\{m_1, \ldots, m_k\} \subseteq M$ and, for $i = 1, \ldots, k$, let S_i be the set of elements in $\{0,1\}^n$ that differ from m_i in exactly one component. (Note that $k \leq 2^{n+1}/(n^2+n+2)$.) In the following, we show that $\mathcal{N}$ shatters the set

$$S \;=\; \{m_1\} \cup \bigcup_{i=1}^{k} S_i,$$

which has cardinality $nk+1$.

The proof is in two parts, where we first consider dichotomies that all classify m_1 as positive. Let (S^-, S^+) be an arbitrary dichotomy of S with $m_1 \in S^+$. It induces dichotomies of the sets S_i, $1 \leq i \leq k$, into

$$S_i^- = S_i \cap S^- \quad \text{and} \quad S_i^+ = S_i \cap S^+.$$

The proof of Theorem 8 in [2] shows that, for $i = 2, \ldots, k$, a linear threshold neuron can be chosen that classifies all elements in S_i^+ as positive and all elements in $S \setminus S_i^+$ as negative. Further, there is a linear threshold neuron that classifies all elements in $S_1^+ \cup \{m_1\}$ as positive and all elements in $S \setminus (S_1^+ \cup \{m_1\})$ as negative. Theorem 1 implies that every Boolean function computed by a linear threshold neuron can also be computed by a generalized binary RBF neuron with any arbitrary symmetric and positive definite matrix. Let A_i, for $i = 1, \ldots, k$, be the given matrix of the i-th generalized Gaussian RBF neuron. Then by Theorem 1, for the generalized binary RBF neuron with matrix A_i, there is a center c_i and a radius r_i such that this neuron induces on S the dichotomy $(S \setminus S_i^+, S_i^+)$ for $i = 2, \ldots, k$, and the dichotomy $(S \setminus (S_1^+ \cup \{m_1\}), S_1^+ \cup \{m_1\})$ for $i = 1$.

This means that we have k ellipsoids such that each element of S^+ is contained in at least one ellipsoid, whereas each element of S^- is not contained in any of them. Let δ be the smallest distance of any element in S^- from any center of these ellipsoids, that is,

$$\delta \;=\; \min_i \min_{x \in S^-} \|x - c_i\|_{A_i}.$$

Then, we have $\delta > r_i$ for $i = 1, \ldots, k$. This implies that we can choose some $\varepsilon > 0$ satisfying

$$\varepsilon \;<\; \min_i \frac{\delta^2/r_i^2 - 1}{\ln(2k)}.$$

We use c_i, for $i = 1, \ldots, k$, as the centers of the generalized Gaussian RBF neurons and define their widths by

$$\sigma_i^2 \;=\; \varepsilon r_i^2,$$

for $i = 1, \ldots, k$. Finally, the weights of the output neuron are set to 1 and the threshold gets the value $\exp(-1/\varepsilon)$.

To see that this network induces the dichotomy as claimed, consider first some arbitrary $s \in S^-$. As s is not contained in any of the ellipsoids, the output of the i-th RBF neuron satisfies

$$\exp\left(-\frac{\|s - c_i\|^2_{A_i}}{\sigma_i^2}\right) \quad \leq \quad \exp\left(-\frac{\delta^2}{\varepsilon r_i^2}\right).$$

For the right-hand side, we get

$$\begin{aligned}
\exp\left(-\frac{\delta^2}{\varepsilon r_i^2}\right) \quad &= \quad \exp\left(-\frac{\delta^2 - r_i^2}{\varepsilon r_i^2}\right) \cdot \exp\left(-\frac{1}{\varepsilon}\right) \\
&< \quad \exp\left(-\frac{\ln(2k)(\delta^2 - r_i^2)}{(\delta^2/r_i^2 - 1)r_i^2}\right) \cdot \exp\left(-\frac{1}{\varepsilon}\right) \\
&< \quad \frac{1}{k} \cdot \exp\left(-\frac{1}{\varepsilon}\right).
\end{aligned}$$

This implies that each of the k RBF neurons outputs a value smaller than the last term. Thus, their sum, weighted by the output weights, is less than the threshold, which is $\exp(-1/\varepsilon)$. Thus, s is classified as negative.

On the other hand, if $s \in S^+$ then $s \in S_i^+$ for some $i \in \{1, \ldots, k\}$, or $s = m_1$. In this case, s is contained in the ellipsoid with center c_i and radius r_i (where $i = 1$ if $s = m_1$), and we have for the output of the i-th RBF neuron that

$$\exp\left(-\frac{\|s - c_i\|^2_{A_i}}{\sigma_i^2}\right) \quad \geq \quad \exp\left(-\frac{r_i^2}{\varepsilon r_i^2}\right).$$

Hence, this output value is at least as large as the output threshold and, since all RBF neurons yield positive values, s is classified correctly.

In the second part, it remains to show that $\mathcal{N}$ is able to compute all possible classifications of S with $m_1 \in S^-$. Let (S^-, S^+) be such a dichotomy and (S_i^-, S_i^+) the induced dichotomies of the sets S_i, for $i = 1, \ldots, k$. By the same reasoning as above, there exists, for $i = 2, \ldots, k$, a linear threshold neuron that classifies all elements of S_i^- as positive and all of $S \setminus S_i^-$ as negative; and there is a linear threshold neuron that classifies all element of $S_1^- \cup \{m_1\}$ as positive and all of $S \setminus (S_1^- \cup \{m_1\})$ as negative. Again by Theorem 1, the functions of these threshold neurons can be computed by generalized binary RBF neurons with any arbitrary symmetric and positive definite matrix. Thus, there are k ellipsoids such that each element of S^- is contained in at least one ellipsoid, and each element of S^+ is not contained in any of them.

We choose the parameter values of the generalized Gaussian RBF neurons exactly as in the first part, with the only difference that in the definition of δ we use S^+ in place of S^-. The weights of the output neuron are set to -1 and the threshold to $-\exp(-1/\varepsilon)$. Then, following the arguments above, if $s \in S^-$ then at least one RBF neuron has an output value greater than or equal to $\exp(-1/\varepsilon)$. Since all other neurons output positive values, this yields a weighted sum that is less than the threshold. On the other hand, if $s \in S^+$ then all RBF neurons output values less than $\exp(-1/\varepsilon)/k$. Hence, the weighted sum exceeds the threshold. This completes the second part of the proof. □

6 Conclusions

We have presented results on the Boolean computational power of generalized binary RBF neurons. A comparison with the linear threshold neuron yielded that they are always at least as powerful as the latter model. Moreover, there are cases when generalized binary RBF neurons can do more than the linear threshold neuron. These results are the first steps into the investigation of the logics of RBF neuron models. Beyond, there are some interesting theoretical issues that are raised by this work, such as, for instance, the task to construct a Boolean function that cannot be computed using a given weighted norm (different from the Euclidean norm) or to count the number of Boolean functions that are computable with that norm.

As a consequence of these results, we have obtained bounds on the VC dimension of generalized RBF neural networks with binary inputs. In particular, the Boolean VC dimension of the standard architecture with generalized (binary or Gaussian) RBF neurons was shown to grow at least in proportion to the number of network parameters, a fact that has been known for other neural network types for quite a while. The matrices of the RBF neurons were considered as fixed and not as adjustable parameters. Therefore, a question for further studies is to calculate the (Boolean or real) VC dimension of RBF networks where the norms of the neurons are learned.

Acknowledgments

Part of this work was done while one of the authors (M. S.) was with the Universität Osnabrück. This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

References

[1] M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations.* Cambridge University Press, Cambridge, 1999.

[2] P. L. Bartlett. Lower bounds on the Vapnik-Chervonenkis dimension of multi-layer threshold networks. In *Proceedings of the Sixth Annual ACM Conference on Computational Learning Theory*, pages 144–150, ACM Press, New York, 1993.

[3] P. L. Bartlett and W. Maass. Vapnik-Chervonenkis dimension of neural nets. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 1188–1192. MIT Press, Cambridge, MA, second edition, 2003.

[4] S. Haykin. *Neural Networks: A Comprehensive Foundation.* Prentice Hall, Upper Saddle River, NJ, second edition, 1999.

[5] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, Cambridge, 1985.

[6] V. Kecman. *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models.* The MIT Press, Cambridge, Mass., 2001.

[7] S. C. Kleene. Representation of events in nerve nets and finite automata. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, volume 34 of *Annals of Mathematics Studies*, pages 3–41. Princeton University Press, Princeton, NJ, 1956.

[8] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

[9] M. L. Minsky and S. A. Papert. *Perceptrons: An Introduction to Computational Geometry.* The MIT Press, Cambridge, Mass., expanded edition, 1988.

[10] S. Muroga. *Threshold Logic and Its Applications.* John Wiley & Sons, 1971.

[11] F. Rosenblatt. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.

[12] M. Schmitt. Neural networks with local receptive fields and superlinear VC dimension. *Neural Computation*, 14:919–956, 2002.

[13] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16:264–280, 1971.