11-30-2010

The Unit RBF Network: Experiments and Preliminary Results

Peter G. Anderson
Rochester Institute of Technology

Recommended Citation
Peter G. Anderson. The Unit RBF network: Experiments and preliminary results. Cybernetics and Systems, 33, 4, 379-390, Nov 2010.

The Unit RBF Network: Experiments and Preliminary Results

Peter G. Anderson
Computer Science Department
Rochester Institute of Technology

URBF, the *unit radial basis function network* is an RBF neural network with all second layer weights set to ± 1. The URBF models functions or physical phenomena by sampling their behaviors at various probe points, and correcting the model, more and more delicately (i.e., using Gaussian functions with ever narrower spread), when discrepancies are discovered. The probe points—input space positions to test and adjust the network—are *linear pixel shuffling* points, used for their highly uniform sampling property.

We demonstrate the network's performance on several examples. It shows its power via good extrapolation behavior: for smooth-boundary discriminations, very few new hidden units need to be added for a large number of probe points.

Definitions

- $\tau = (1 + \sqrt{5})/2 \approx 1.618034$ is the golden mean. τ is the positive root of $\tau^2 = \tau + 1$.
- $\alpha \approx 1.4655712$ is the positive real root of $\alpha^3 = \alpha^2 + 1$.
- $\beta = \alpha(\alpha - 1)$.
- $\{z\} = z - [z]$ denotes the fractional part of z.
- $c_k = \{k\tau\}$, for $k = 1, 2, 3, \cdots$. This is a well-known good sequence of probe points in $I = (0, 1)$ [1, 8].

- $\vec{c}_k = (\{k\alpha\}, \{k\beta\})$, for $k = 1, 2, 3, \cdots$, is a sequence of points which uniformly probes the unit square, I^2. This sequence of points arises from algorithms in the theory of *linear pixel shuffling (LPS)* [3]. Higher dimensional analogues have been investigated to probe I^n; for the present discussion we limit the dimensions to $n = 1$ or 2.
- $e^{-||\vec{x}||^2}$ is the Gaussian ("bump") function centered at $\vec{0}$. A set of scaled translates of this function make up the hidden layer of *radial basis function (RBF) neural networks.*
- *psup* $\mathcal{D} = \{\vec{x} \mid \mathcal{D}(\vec{x}) > 0\}$ is the *positive support* of $\mathcal{D}$. $\mathcal{D}$ is some arbitrary real-valued function whose purpose here is to divide I^n, into two sets according to the sign of $\mathcal{D}(\vec{x})$.

Introduction

We present a method to determine the coefficients, d_k, of a function

$$\mathcal{F}(\vec{x}) = \sum_{k=1}^{N} d_k e^{-k||\vec{x}-\vec{c}_k||^2}$$

where the number of terms, N, is a positive integer and $d_k \in \{-1, 0, 1\}$, such that the positive support of $\mathcal{F}$ closely approximates that of $\mathcal{D}$.

We determine the coefficients d_k as follows. d_1 is the sign of $\mathcal{D}(\vec{c}_1)$; for $k > 1$,

$$\begin{aligned} \text{if} \quad & \mathcal{D}(\vec{c}_k) \cdot \textstyle\sum_{i=1}^{k-1} d_i e^{-i||\vec{c}_k-\vec{c}_i||^2} \leq 0 \\ \text{then} \quad & d_k \leftarrow sign\ \mathcal{D}(\vec{c}_k) \\ \text{else} \quad & d_k \leftarrow 0 \end{aligned}$$

That is, $d_k = 0$ in case the $(k-1)$-term function produces the correct sign at $\vec{c}_k$; otherwise, d_k is nonzero, to correct the function. In all our experiments, that choice of an additional term did the trick; i.e., we never had $|\sum_{i=1}^{k-1} d_i e^{-i||\vec{c}_k-\vec{c}_i||^2}|$ exceed 1.0.

Linear pixel shuffling

Knuth [8] pointed out the the *mod-1* multiples of the golden mean (the sequence c_k) is an effective way to probe the unit interval. A discrete analog

of this, the multiples of a Fibonacci number modulo the following Fibonacci number, is a similarly excellent shuffling of a list of integers: values numerically close to each other are distant in position on the list. For instance, the *mod-21* multiples of 13 are:

0, 13, 5, 18, 10, 2, 15, 7, 20, 12, 4,
17, 9, 1, 14, 6, 19, 11, 3, 16, 8

This property was exploited in [2] for a scan-line ordering for computer graphics in order to get a useful visual preview of the product of time-consuming computer graphics.

In [3] this phenomenon was generalized to higher dimensions: *mod-1* multiples of appropriate irrational vectors probed I^n smoothly, giving a useful sequence of points for Monte Carlo integration; a discrete analogy provided the means to visit the coordinates of a large matrix—for example, the pixels of a large image—in a jumbled order. The discrete pixel shuffling can be very useful for progressive image rendering and image processing operations, such as morphology [6] and digital half tone production [10]. Fig. 1 shows how 100 and 1000 *mod-1* multiples of $(\alpha, \beta) = (1.465571, 0.682328)$ fill I^2.

The three-dimensional analog of this sequence was used in [5] to select the first layer of weights for MLPs with a single hidden layer with transfer function *tanh*.

The discrete version (truly *linear pixel shuffling*) was used in [4] to select pixels for computing features for a hand printed character recognition system.

Experimental results

One dimensional experiments

A one-point discrimination

Our first experiment involves modeling the positive support of the function $y = x - 0.5$ with a function of the form

$$\mathcal{F}(x) = \sum_{k=1}^{N} d_k e^{-k(x-c_k)^2}$$

For $N = 100$ there were only four non-zero coefficients: $d_1 = 1$, $d_2 = -1$, $d_4 = -1$, and $d_9 = 1$. The graph of $y = \mathcal{F}(x)$ is shown in Fig. 2. For $N =$

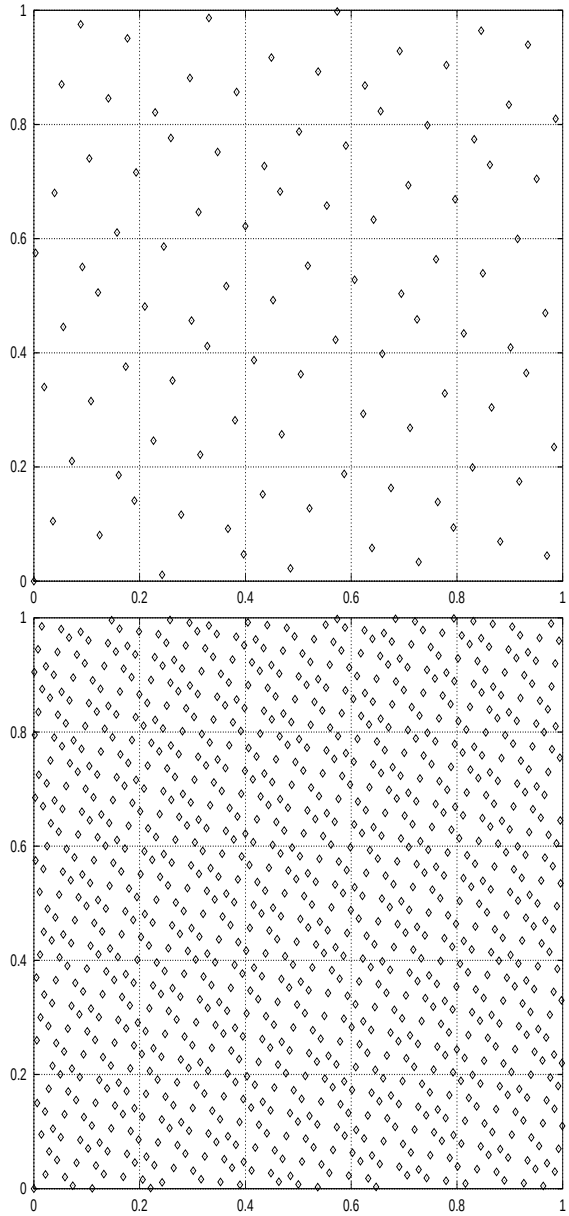

Figure 1: The first 100 and 1000 points in the sequence $\vec{c}_k$.

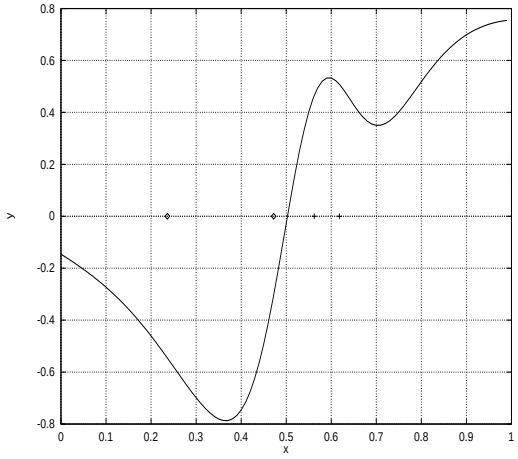

Figure 2: A function $y = \mathcal{F}(x)$ constructed to have sign approximately the same as that of $y = x - 0.5$. The centers of the four non-zero terms are marked in the x axis.

10000 there are only three more non-zero coefficients: $d_{305} = 1$, $d_{449} = -1$, and $d_{682} = -1$.

Two dimensional experiments

A linear discriminator

Our first two dimensional example is also simple linear discrimination; the unit square is partitioned according to the sign of $5x + 3y - 4$. Our model

$$\mathcal{F}(\vec{x}) = \sum_{k=1}^{3000} d_k e^{-k||\vec{x}-\vec{c}_k||^2}$$

uses only 47 non-zero d_k's. The graph of $\mathcal{F}_N(\vec{x})$ is shown in Fig. 3. Surprisingly, a large fraction of the function values in this domain have absolute value close to 1. The centers of the 47 bump functions (corresponding to the nonzero d_k's) and the positive support of $\mathcal{F}$ are shown in Fig. 4

Nested spirals

Our second two-dimensional experiment is concerned with the famous *two spirals problem* [7]. This problem requires a neural network or other machine

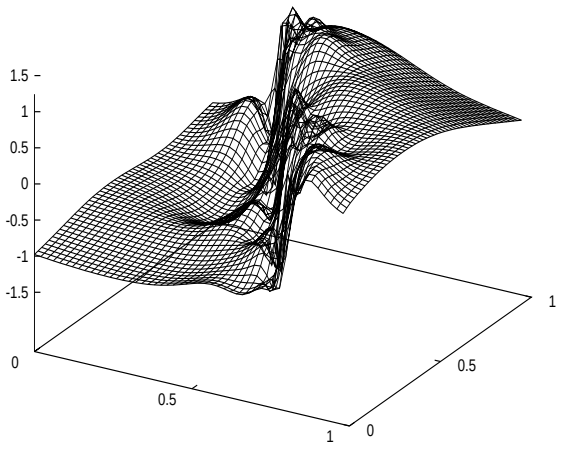

Figure 3: A function $z = \mathcal{F}(\vec{x})$ constructed to have positive support approximately the same as that of $5x + 3y - 4$.

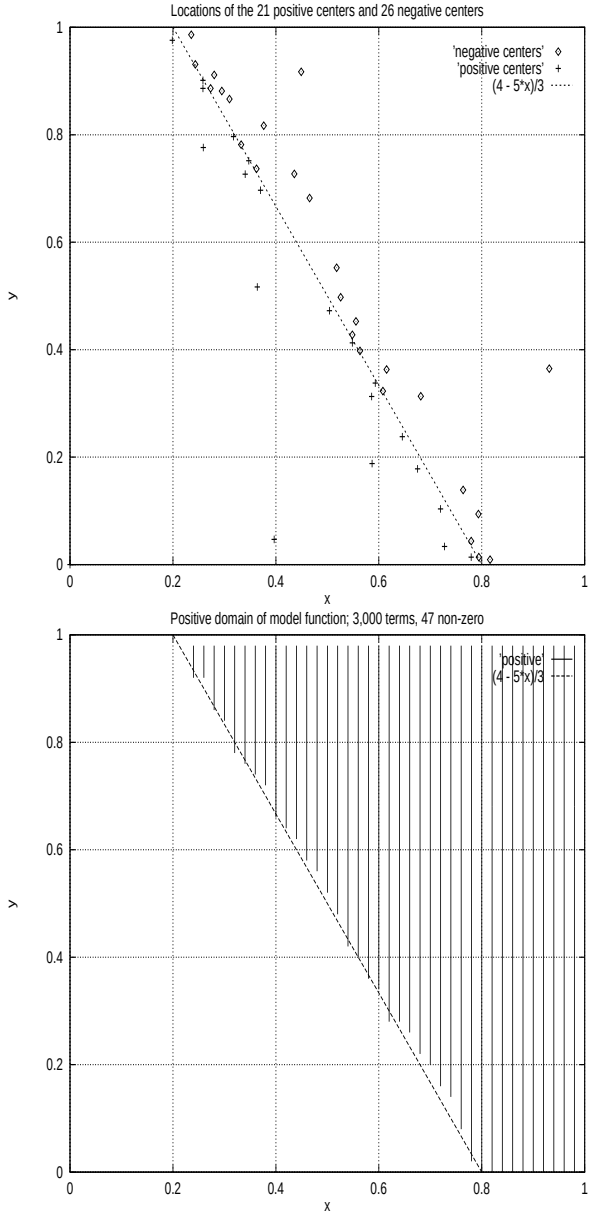

Figure 4: Left: the placement of the 47 bump functions' centers. Notice the clustering of the centers near the boundary. The behavior of the probe sequence suggests that the present pattern is representative of zoomed images at more general, but smooth boundary curves. Right: the positive support of the model function $\mathcal{F}$.

learning system to distinguish between two sets of $M \approx 100$ points of the form

$$\begin{aligned} A &= \{(r_k \cos\theta_k, r_k \sin\theta_k)\} \\ B &= \{(-r_k \cos\theta_k, -r_k \sin\theta_k)\} \end{aligned}$$

where, for $k = 1, \cdots, M$,

$$\begin{aligned} r_k &= \frac{k}{M} \\ \theta_k &= \frac{6\pi k}{M} \end{aligned}$$

A and B are points on two nested, three-revolution spirals.

The neural network in the present study does not use training sets such as A and B; instead we use a function $\mathcal{D}$ whose positive and negative supports contain a pair of nested spirals in their interiors:

$$\mathcal{D}(r, \theta) = \sin(\theta + 7.5r)$$

The positive support of this $\mathcal{D}$ and those of three successive $\mathcal{F}$ models are shown in Fig. 5.

A graph showing the number of centers for 50–1000 probes is shown in Fig. 6.

Preliminary analysis

How many of the N coefficients, $\{d_k\}$ are non-zero? Consider the simplest nontrivial two class discrimination in the $n = 2$ dimensional cube, given by the positive support of the function $z = \mathcal{F}(x, y) = x - 0.5$.

The probe point-sequence $\{\vec{c}_k\} \subset I^n$ was chosen so that the pattern of points visiting subcubes of I^n appears the same as the pattern visiting the whole of I^n. Thus, for two-class discrimination functions with piecewise smooth boundaries, we may cover I^n with small subcubes of side s and consider the probe points in the subcubes that cover the boundary. Suppose that the cubes are each oriented so that the boundary segment each one covers nearly bisects each cube with a nearly plane surface parallel to a cube side. If μ is the $(n-1)$-dimensional volume of the decision boundary, in the limit,

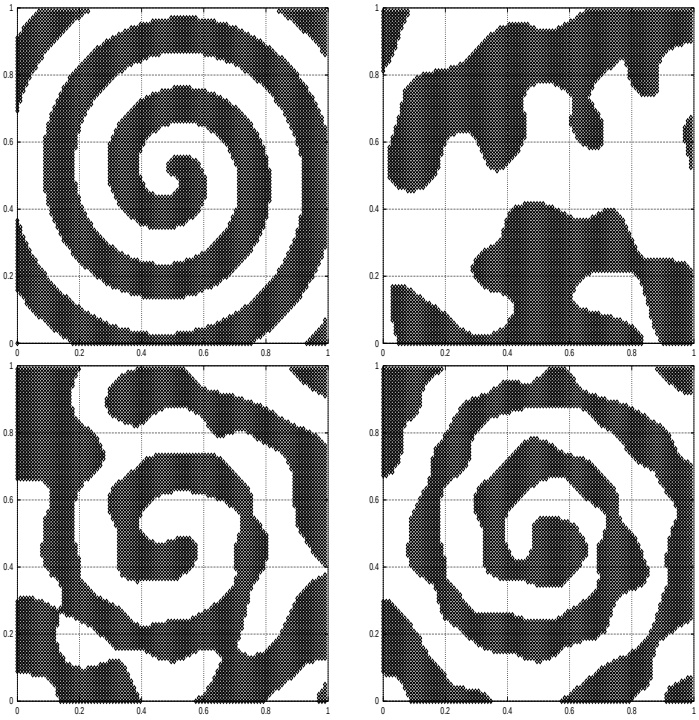

Figure 5: The upper left picture shows the target set. The remaining three show the response of the URBF network after it was trained with 100, 200, and 300 probes. The number of hidden units in these three nets is 57, 108, and 138, respectively.

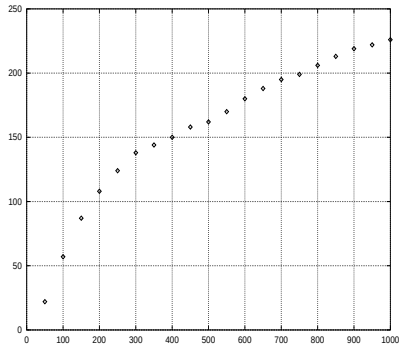

Figure 6: The number of centers used in the model function for 50–1000 probes.

that boundary will be covered by μ/s^{n-1} cubes with a total n-dimensional volume of μs. Consequently, approximately $\mu s N$ of the N probe points meet these subcubes. We expect the fraction of the probe points meeting each of the subcubes that corresponds to a non-zero d_k to be approximately the same as the fraction of N probes in the simple problem.

Experiments show this to be overly pessimistic: the networks we have constructed employ many fewer hidden nodes than this analysis predicts.

Discussion

Economy of representation

The number (or ratio) of non-zero coefficients, d_k, appears to depend on the length of the boundary of the target function's positive support.

This technique would be most appropriate for modeling generally smooth boundaries; we expect it to be most uneconomical for fractals.

For smooth boundaries, we may approximate small portions of the boundaries as nearly straight lines. We expect that our linear experiment 1 is representative of the behavior of these systems. Behavior of the probe sequence within small neighborhoods is analogous to its behavior in the entire I^2. Thus, a square neighborhood bisected by a nearly straight line, should contain the centers of bump functions corresponding to roughly 1.5% non-zero d_k's.

Neural networks

A feed-forward neural network with one hidden layer and two layers of synaptic weights is shown in Fig. 7. In case the hidden layer transfer function is a translation of a dilated Gaussian, this is known as a *radial basis function (RBF)* network [9]. Our URBF is of this form, however the second layer of synaptic weights all have value ± 1.

Traditionally, such networks determine their weights through a process of supervised training. This requires two sets of training exemplars, i.e., input-output pairs of the form $\langle \vec{x}, z \rangle$. One set (the *training set*) is used to determine the weights w_{ij} and v_k, and the other set (the *testing set*) is used to determine the resulting network's performance. If that performance is less than satisfactory, one or more actions have to be taken: add more

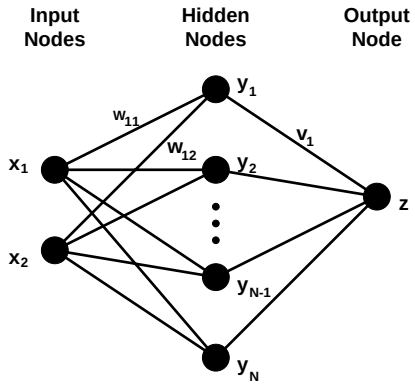

Figure 7: A "2 feeds N feeds 1" *URBF* network. The y_k's emit a Gaussian function of the difference between the weights w_{ij} and the the inputs x_j; the second layer of weights, satisfies $v_i = \pm 1$.

hidden layers to the network, increase the training set, modify the training algorithm, etc.

In contrast, our URBF requires a function or physical phenomenon which it interrogates, adding hidden nodes as required. Instead of a testing set, the network's performance may be judged by the number of probes it used and the generally small number of hidden units it adds.

Analogy to wavelet analysis

Our bump functions summed to model a subset discrimination are reminiscent of wavelet decomposition of continuous-scale image. In the latter, the gray value is approximated by a linear combination of functions, where each function is a translate and a dilation of a given, fixed function (the "mother wavelet"). We note the following distinctions:

- Our analog of the mother wavelet, $e^{-||\vec{x}||^2}$ does not have a zero integral.
- Our goal is to model a function's positive domain, not its values.
- We only employ coefficient values in $\{-1, 0, 1\}$.

Conclusions

Our URBF consists of a novel modification of the RBF neural network architecture for two-class discrimination problems. Like a scientist, testing the predictive power of a theory, it learns by interrogating the phenomenon at points chosen to be far from the points it has already used for training (points that would be especially suitable for Monte Carlo integration).

The neural network models that this approach develops are quite small. This attests to the appropriateness of three aspects of the system:

- the smoothness and simplicity of the functions used for training,
- the uniformity of the LPS point sequence, and
- the sequence of Gaussian functions with decreasing spreads.

References

[1] Peter G. Anderson. Multidimensional golden means. In *Applications of Fibonacci Numbers*, University of Saint Andrews, Scotland, July 1992.

[2] Peter G. Anderson. Fast rendering. *Computer Language*, Feb 1993.

[3] Peter G. Anderson. Advances in linear pixel shuffling. In G. E. Bergum, A. N. Philippou, and A. F. Horodam, editors, *Applications of Fibonacci Numbers*, pages 1–21, Boston, MA, 1994. Klewer Academic Publishers.

[4] Peter G. Anderson and Roger S. Gaborski. The polynomial method augmented by supervised training for hand printed character recognition. In R. F. Albrecht, C. R. Reves, and N. C. Steele, editors, *Artificial Neural Networks and Genetic Algorithms, Proceedings of the International Conference*, pages 417–422, Innsbruck, Austria, 1993.

[5] Peter G. Anderson, Roger S. Gaborski, Sanjay Raghavendra, Ming Ge, and Mei-ling Lung. Using quasirandom numbers in neural networks. In *ICSC International Symposium on Fuzzy Logic*, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 1995.

[6] Mark deRoller. Implementing image morphology operations using linear pixel shuffling. Master's project, Rochester Institute of Technology, 1998.

[7] Scott E. Fahlman and Christian Lebiere. The cascade correlation learning architecture. Technical report, Carnegie-Mellon University, Pittsburgh, PA, 1990.

[8] Donald E. Knuth. *The Art of Computer Programming, Volume 3/Sorting and Searching.* Addison-Wesley, Reading, MA, 1977.

[9] Mark J. L. Orr. Introduction to radial basis function networks. Technical report, Center for Cognitive Science, University of Edinburgh, Scotland, 1996. `http://www.cns.ed.ac.uk/ people/mark.html`.

[10] John Szybist. An error diffusion algorithm based on linear pixel shuffling. Computer science master's project, Rochester Institute of Technology, 1997.