# Radial basis function

Radial functions are a special class of function. Their characteristic feature is that their response decreases (or increases) monotonically with distance from a central point. The center, the distance scale, and the precise shape of the radial function are parameters of the model, all fixed if it is linear.

A typical radial function is the Gaussian which, in the case of a scalar input, is

$$h(x) = \exp\left(-\frac{(x-c)^2}{r^2}\right)$$

Its parameters are its center $c$ and its radius $r$. Figure 1 illustrates a Gaussian RBF with center $c = 0$ and radius $r = 1$

A Gaussian RBF monotonically decreases with distance from the center. In contrast, a multiquadric RBF which, in the case of scalar input, is

$$h(x) = \frac{\sqrt{r^2 + (x-c)^2}}{r},$$

monotonically increases with distance from the center (see Figure 1). Gaussian-like RBFs are local (give a significant response only in a neighborhood near the center) and are more commonly used than multiquadric-type RBFs which have a global response. They are also more biologically plausible because their response is finite.

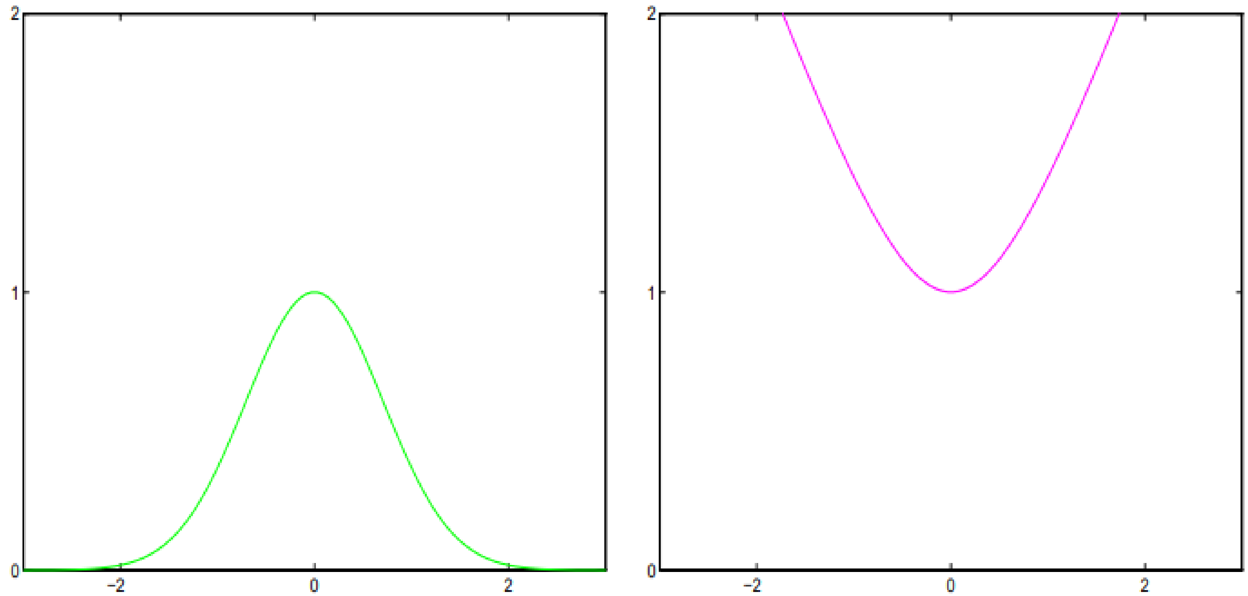


*Figure 1:2: Gaussian an (left) and multiquadric RBFs.*

## Radial Basis Function Networks

Radial functions are simply a class of functions. In principle, they could be employed in any sort of model (linear or nonlinear) and any sort of network (single-layer or multi-layer). However, since Broomhead and Lowe's 1988 seminal paper, radial basis function networks (RBF networks) have traditionally been associated with radial functions in a single-layer network such as shown in figure 2.

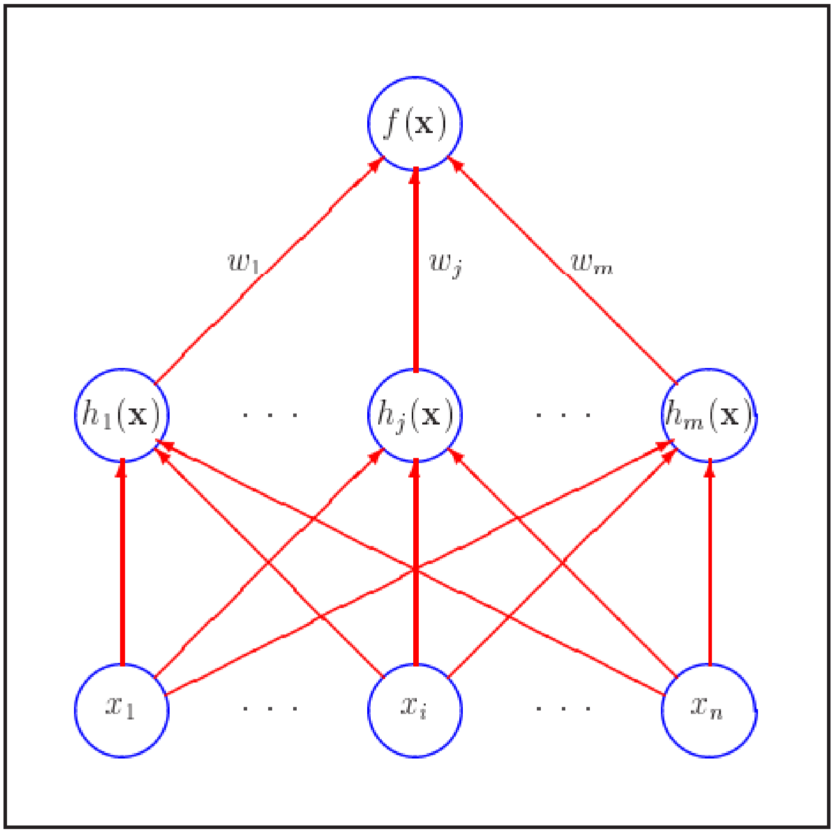


*Figure 2: The traditional radial basis function network. Each of $n$ components of the input vector $x$ feeds forward to $m$ basis functions whose outputs are linearly combined with weights $\{w_j\}_{j=1}^{m}$ into the network output $f(x)$.*

An RBF network is nonlinear if the basis functions can move or change size or if there is more than one hidden layer. Here we focus on single-layer networks with functions which are fixed in position and size.

# Adaptive Resonance Theory

Adaptive Resonance Theory, or ART, is a cognitive and neural theory of how the brain autonomously learns to categorize, recognize, and predict objects and events in a changing world. The problem of learning makes the unity of conscious experience hard to understand, if only because humans are able to rapidly learn enormous amounts of new information, on their own, throughout life. How do humans integrate all this information into unified experiences that cohere into a sense of self? One has only to see an exciting movie just once to marvel at this capacity, since we can then tell our friends many details about it later on, even though the individual scenes flashed by very quickly. More generally, we can quickly learn about new environments, even if no one tells us how the rules of each environment differ. To a remarkable degree, humans can rapidly learn new facts without being forced to just as rapidly forget what they already know. As a result, we can confidently go out into the world without fearing that, in learning to recognize a new friend's face, we will suddenly forget the faces of our family and friends. This is sometimes called the problem of catastrophic forgetting.

Many contemporary learning algorithms do experience catastrophic forgetting, particularly when they try to learn quickly in response to a changing world. These include the competitive learning, self-organizing map, back propagation, support vector machine, regularization, and Bayesian models. The brain solves a challenging problem that many current biological and technological learning models have not yet solved: It is a self-organizing system that is capable of rapid, yet stable, autonomous learning in real time of huge amounts of data from a changing environment that can be filled with unexpected events. Discovering the brain's solution to this key problem is as important for understanding ourselves as it is for developing new pattern recognition and prediction applications in technology.

Grossberg (1980) has called the problem whereby the brain learns quickly and stably without catastrophically forgetting its past knowledge the stability–plasticity dilemma (stability refers to their nature of memorizing the learning and plasticity refers to the fact that they are flexible to gain new information.). The stability–plasticity dilemma must be solved by every brain system that needs to rapidly and adaptively respond to the flood of signals that subserves even the most ordinary experiences.

Carpenter and Grossberg's Adaptive Resonance Theory models (ART1, ART2, and ARTMap) were developed in an attempt to overcome this dilemma." The network has a sufficient supply of output units, but they are not used until deemed necessary. A unit is said to be committed (uncommitted) if it is (is not) being used. The learning algorithm updates the stored prototypes of a category only if the input vector is sufficiently similar to them. An input vector and a stored prototype are said to resonate when they are sufficiently similar. The extent of similarity is controlled by a vigilance parameter, $p$, with $0 < p < 1$, which also determines the number of categories. When the input vector is not sufficiently similar to any existing prototype in the network, a new category is created, and an uncommitted unit is assigned to it with the input vector as the initial prototype. If no such uncommitted unit exists, a novel input generates no response.

We present only ART1, which takes binary (0/1) input to illustrate the model. Figure 3 shows a simplified diagram of the ARTl architecture. It consists of two layers of fully connected units. A top-down weight vector $\boldsymbol{w}_j$, is associated with unit $j$ in the input layer, and a bottom-up weight vector $\overline{\boldsymbol{w}_\iota}$ is associated with output unit $i$; $\overline{\boldsymbol{w}_\iota}$ is the normalized version of $\boldsymbol{w}_i$,

$$\overline{\boldsymbol{w}_\iota} = \frac{\boldsymbol{w}_i}{\varepsilon + \sum_j w_{ji}}$$

where $\varepsilon$ is a small number used to break the ties in selecting the winner. The top-down weight vectors $\boldsymbol{w}_j$'s store cluster prototypes. The role of normalization is to prevent prototype with a long vector length from dominating prototypes with a short one. Given an $n$-bit input vector $\boldsymbol{x}$, the output of the auxiliary unit $A$ is given by

$$A = Sgn_{0/1}\left(\sum_j x_j - n\sum_i O_i - 0.5\right)$$

where $Sgn_{\frac{0}{1}}(x)$ is the signum function that produces $+1$ if $x \geq 0$ and 0 otherwise, and the output of an input unit is given by

$$V_j = Sgn_{0/1}\left(x_j + \sum_i w_{ji}O_i + A - 1.5\right)$$

$$= \begin{cases} x_{j,} & if\ no\ output\ O_i\ is\ "on" \\ x_j \wedge \sum_i w_{ji}O_i, & otherwise \end{cases}$$

A reset signal $R$ is generated only when the similarity is less than the vigilance level. The ARTl model can create new categories and reject an input pattern when the network reaches its capacity. However, the number of categories discovered in the input data by ARTl is sensitive to the vigilance parameter.

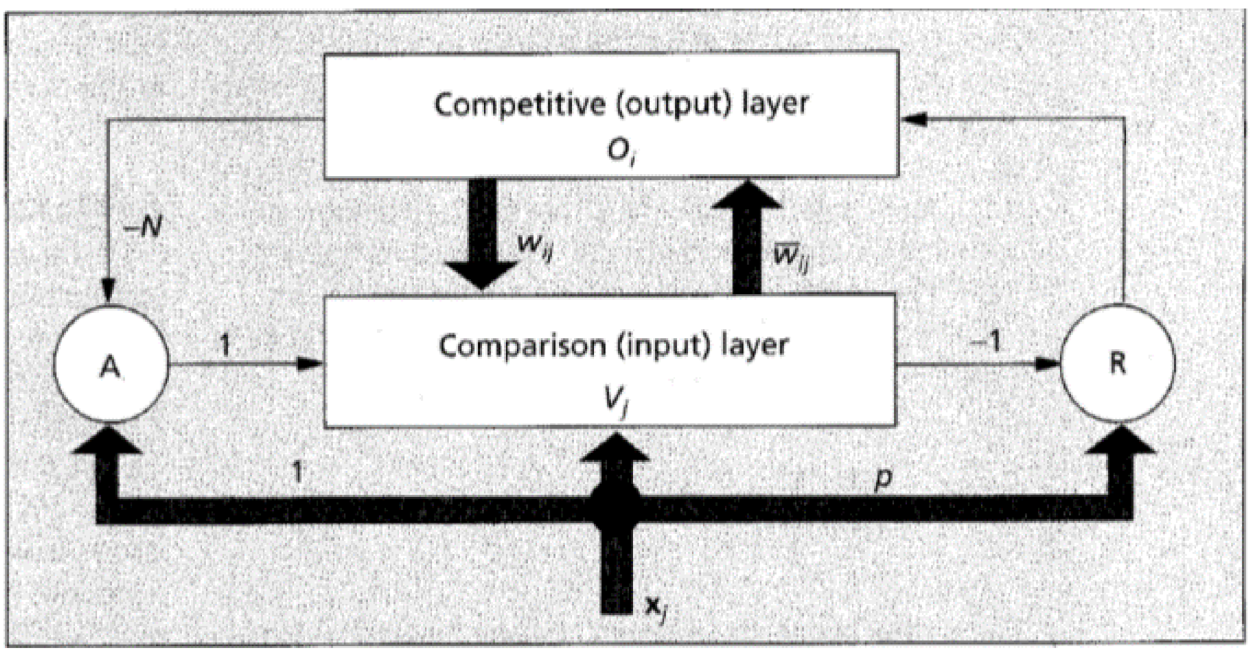


*Figure 3: ART1 network*

ART1 learning algorithm

1. Initialize $w_{ij} = 1$, for all $i, j$. Enable all output units.
2. Present a new pattern $\boldsymbol{x}$.
3. Find the winner unit $i^*$ among the enabled output units

   $$\overline{\boldsymbol{w}_{i^*}}\boldsymbol{x} \geq \overline{\boldsymbol{w}_i}\boldsymbol{x}, \qquad \forall\, i$$

4. Perform vigilance test

   $$r = \frac{\boldsymbol{w}_{i^*}\boldsymbol{x}}{\sum_j \boldsymbol{x_j}}$$

   If $r \geq p$ (resonance), go to step 5. Otherwise, disable unit $i^*$ and go to step 3 (until all the output units are disabled).

5. Update the winning weight vector $\boldsymbol{w}_i$ enable all the output units, and go to step 2

   $$\Delta w_{j\,i^*} = \eta(V_j - w_{j\,i^*})$$

6. If all the output units are disabled, select one of the uncommitted output units and set its weight vector to $x$. If there is no uncommitted output unit (capacity is reached), the network rejects the input pattern.

## References and further reading:

D.S. Broomhead and D. Lowe. Multivariate functional interpolation and adaptive networks. Complex Systems, 2:321-355, 1988.

Grossberg, S. (1980). How does a brain build a cognitive code? Psychological Review, 87, 1–51.

Grossberg, S. (2013). Adaptive Resonance Theory: How a brain learns to consciously attend, learn, and recognize a changing world. Neural Networks, 37, 1-47.

Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial. Computer, 29(3), 31-44.

Orr, M. J. (1996). Introduction to radial basis function networks.

https://www.geeksforgeeks.org/adaptive-resonance-theory-art/ [visited in February 2020]