Performances of non-parameterised radial basis functions in pattern recognition applications

S Tavaen1,*, R Viriyapong2 and S Kaennakham1,3

1School of Mathematics, Institute of Science, Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand.
2Department of Mathematics, Faculty of Science, Naresuan University, Phitsanulok, 65000, Thailand.
3Centre of Excellence in Mathematics, Bangkok, 10400, Thailand.

*E-mail: sunisat58@nu.ac.th

Abstract. Pattern recognition appears in many applications with most popular scheme are those involved the so-called ‘Radial Basis Function (RBF)’. It is known that the shape parameter contained in some RBFs used has great influence on the final quality of prediction. This study focusses on RBFs which contains no parameters where three data patterns are used for performance validation. With a good choice of number of centres, it is clearly possible to obtain satisfactory results with no burden on choosing the suitable or optimal shape. This can well shed more light into applications with more complexity with less user’s judgment and be more automatic in the process.

1. Introduction

Pattern recognition is the process of differentiating and dividing the data according to certain criteria or by general components, which are performed by special algorithms. Because pattern recognition helps to classification and prediction, it is one of the important components of machine learning technology [1]. Pattern recognition is applied to image processing [2], industry [3], and medical [4] (see references therein).

The task of pattern recognition is to construct the model with unknown input-output mapping pattern. It is to construct the best model, if any, from the train data with some mapping functions and expect this model to best represent the rest of the data, called ‘training data’. Both sets of the data can be of the following form;

\[ D = \{ (x_i, y_i) | x_i \in \mathbb{R}^d, y_i \in \mathbb{R}, i = 1, 2, \ldots, n \} \]  

where \( x_i \) are inputs with the corresponding \( y_i \) are outputs. The main task is to find a mapping \( D \) from the \( d \) – dimensional input space to \( 1 \) – dimensional output space. Over the decade, there have been several models designed to tackle the problem and some are statistical model, structural model, template matching model, neural network based model, fuzzy based model, and hybrid model. Amongst these, very often that radial basis functions (RBF) are involved where the crucial factor is the shape parameter, mostly contained within the RBF used. The most popular choice for RBF is the famous Gaussian type and its performance is certainly determined by its shape parameter. Therefore, the main objective of this
work is to investigate the capability of other RBFs containing no shape parameter for the same problem of pattern recognition.

Section 2 provides the brief concept of dealing pattern recognition by using radial basis functions before three non-parameterised RBFs under investigation in this work are shortly presented in Section 3. The experiments are demonstrated numerically with the results are presented in Section 4 and the main findings are listed in Section 5.

2. Pattern Recognition by Radial Basis Functions
Model structure of radial basis function is given data set \( \{(x_i, y_i)\}_{i=1}^{n} \) and output estimate \( \hat{y} \) for input vector \( x \) is represented by functional form:

\[
\hat{y} = f(x) = \sum_{j=1}^{w} w_j \phi_j(x) = \sum_{j=1}^{w} w_j \phi_j(\|x - \mu_j\|_2)
\]

(2)

where \( \|\cdot\|_2 \) is the Euclidean distance norm, \( \phi(\cdot) \) is a basis function and \( m \ll n, \mu_j, w_{j} \) are the width of \( j \)-th basis functions, the number of basis function, the centre of \( j \)-th basis functions and the weight associated with the \( j \)-th basis function, respectively.

The Gaussian is the most popular basis function because it has attractive mathematical properties defined as:

\[
\phi(r) = \exp\left(-r^2/2\sigma^2\right) = \exp\left(-\|x\|_2^2/2\sigma^2\right)
\]

(3)

Gaussian RBF model is dependent on the width (or shape parameter) \( \sigma \) of basis function. Schemes involving this methods is that presented in [5], orthogonal least squares [6], and more can be found in [7]. In this work, RC algorithm is paid with attention and is to be given with more detail in Section 4.

3. The Non-Parameterised Radial Basis Functions
With the non-straight forward way to pinpoint the optimal choice for Gaussian RBF uses, the main attention has now turned to alternative forms of RBF. In this work, it focuses on non-parameter form of radial basis functions and with this purpose, those proposed by Buhmann [8] are to be explored and they are:

- Noted as ‘CS-RBF1’ and defined as: \( \phi(r) = \frac{1}{3} + r^2 - \frac{4}{3} r^3 + 2r^2 \log(r) \).
- Noted as ‘CS1’ and defined as: \( \phi(r) = \frac{112}{45} r^2 + \frac{16}{3} r^2 - 7r^4 - \frac{14}{15} r^2 + \frac{1}{9} \).
- Noted as ‘CS2’ and defined as: \( \phi(r) = \frac{1}{18} - r^2 + \frac{4}{9} r^3 + \frac{1}{2} r^4 - \frac{4}{3} r^3 \log(r) \).

What appears to be interesting about these forms is that they do not depend on any user’s input information making it more convenient when in use.

4. Numerical Experiments

4.1. RC Algorithm
RC algorithm as proposed by Shin and Park [9] is a method to drag out information of interpolation matrix when RBF is in use. For given input data \( \{x_i, y_i\}_{i=1}^{n} \), the algorithm contains the following steps.

**Step 1**: Select a value for width (\( \sigma \)) and effect of noise (\( \delta \)) which \( \delta \) is usually taken to be 0.1% to 1.0% and construct the interpolation matrix \( G \). For example, when Gaussian basis function is used, so that
\[ G = \begin{bmatrix} g_{i1} & g_{i2} & \cdots & g_{in} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1} & g_{n2} & \cdots & g_{nn} \end{bmatrix} \]  

(4)

where \( g_{ij} = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right) \) for \( i, j = 1, 2, \ldots, n \).

**Step 2:** Determine the number of basis function \((m)\) by applied singular value decomposition of the interpolation matrix \(G\). This yields a diagonal matrix of singular values \( s_1 \geq s_2 \geq \cdots \geq s_n \geq 0 \). From these, \( m \) can be determined from the following:

\[
m = \max \left\{ i \mid s_{i+1} \leq s_1 \times \frac{\delta}{100} \right\}.
\]

(5)

**Step 3:** Determine the centres of basis function \((\mu)\). Partition matrix \(V\) from singular value decomposition of the interpolation matrix \(G\) as:

\[
V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1m} \\ v_{21} & v_{22} & \cdots & v_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & \cdots & v_{mm} \end{bmatrix}^{m-n}
\]

(6)

Next, generate matrix \(V' = \begin{bmatrix} v'_{11} & v'_{12} \\ v'_{21} \end{bmatrix}^{m-n}\) and apply QR factorization with column pivoting of matrix \(V'\). And then compute \(X^TP\) and choose the first \(m\) elements in \(X^TP\) be the centre of basis function which are:

\[
\mu = \{\mu_j\}_{j=1}^{m}
\]

(7)

**Step 4:** Compute the weight parameters from the basis function \((w)\). Consider,

\[
\phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1m} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mm} \end{bmatrix}
\]

(8)

where \( \phi_{ij} = \exp\left(-\frac{||x_i - \mu_j||^2}{2\sigma^2}\right) \) for \( i = 1, 2, \ldots, n \) and \( j = 1, 2, \ldots, m \).

Compute the \(m\) weights with

\[
w = \phi^+ y
\]

(9)

where \( \phi^+ \) denoted the pseudo inverse of \( \phi \).

For result validation, the mean square error (MSE) defined as follows is used:

\[
MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - \tilde{y}_i)^2
\]

(10)

With being the number of data points involved in each case.

4.2. Test 1: Linear interpolation

In this section, a linear case is studied and the simple line expressed as followed is considered:

\[ y = 2x + 1 \]

(11)

The total number of 100 data points from above function with \( x \) in \([0, 2\pi]\) are generated. The data points are generated by \( x_i = 2\pi \left(\frac{i-1}{100}\right), i = 1, 2, \ldots, 100 \), with node distribution and its graph are in figure 1.
With using RC Algorithm, the parameters $\sigma$ and $m$ are optioned for both training and validation cases are shown in table 1.

Table 1. Training and validation errors for candidate models for linear trend case.

| RBF      | $\sigma$ | $m$ | MSE          |
|----------|----------|-----|--------------|
|          |          |     | Training | Validation |
| CS RBF1  | -        | 29  | 1.6659 | 3.1688 |
| CS1      | -        | 13  | 1.9633 | 2.849 |
| CS2      | -        | 22  | 1.853  | 3.0211 |
| Gaussian I | 0.2  | 9   | 2.2658 | 2.4408 |
| Gaussian II | 0.3 | 7   | 2.2941 | 2.4161 |
| Gaussian III | 0.4 | 5   | 2.3514 | 2.3643 |
| Gaussian IV | 0.5 | 5   | 2.3458 | 2.3603 |
| Gaussian V  | 0.6  | 4   | 2.3444 | 2.3585 |
| Gaussian VI | 0.7  | 4   | 2.3438 | 2.3574 |

Thus, based on RC Algorithm, by using Gaussian III with $\sigma = 0.4$ and $m = 5$ for this data set. When using the same numbers of centres, table 2 and table 3 provide the main results of this case.

Table 2. Results comparison when using different numbers of centres and RBFs for linear trend case.

| RBF      | $m = 9, \sigma = 0.2$ | $m = 7, \sigma = 0.3$ | $m = 5, \sigma = 0.4$ | $m = 4, \sigma = 0.7$ |
|----------|-----------------------|-----------------------|-----------------------|-----------------------|
|          | Training Error | Validation Error | Training Error | Validation Error | Training Error | Validation Error | Training Error | Validation Error |
| CS RBF1  | 2.1988     | 2.5176 | 2.1796 | 2.4946 | 2.2539 | 2.6201 | 2.4811 | 2.5307 |
| CS1      | 2.2469     | 2.4538 | 2.2802 | 2.4502 | 2.6874 | 2.6801 | 2.3129 | 2.5212 |
| CS2      | 2.2698     | 2.4799 | 2.3149 | 2.4290 | 2.6218 | 2.6326 | 2.6478 | 2.8352 |
| Gaussian | 2.2658     | 2.4408 | 2.2941 | 2.4161 | 2.3514 | 2.3643 | 2.3438 | 2.3574 |
### Table 3. Listing of basis function centres for linear trend case.

| $m$ | RBF     | $\mu_1$  | $\mu_2$  | $\mu_3$  | $\mu_4$  | $\mu_5$  | $\mu_6$  | $\mu_7$  | $\mu_8$  | $\mu_9$  |
|-----|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 9   | CS-RBF1 | 0.6768   | 0.9697   | 0.0303   | 0.1818   | 0.9293   | 0.8586   | 0.6465   | 0.2525   | 0.8990   |
|     | CS1     | 0.0707   | 0.6667   | 0.5859   | 0.9293   | 0.8485   | 0.2424   | 1.0000   | 0        | 0.4141   |
|     | CS2     | 0.7172   | 0.1919   | 0.2323   | 0.0404   | 0.6667   | 1.0000   | 0.9091   | 0.7677   | 0.1414   |
|     | Gaussian| 0        | 1.0000   | 0.0808   | 0.9192   | 0.2121   | 0.7879   | 0.3535   | 0.6465   | 0.5051   |
| 7   | CS-RBF1 | 0.2828   | 0.9697   | 0.5051   | 0.7475   | 0.9293   | 0.1818   | 0.8990   |          |          |
|     | CS1     | 0.2424   | 0.4141   | 0.9293   | 0        | 0.3333   | 1.0000   | 0.5859   |          |          |
|     | CS2     | 0.7172   | 0.5253   | 0.1919   | 0.6162   | 1.0000   | 0.0404   | 0.2828   |          |          |
|     | Gaussian| 0        | 1.0000   | 0.1010   | 0.8889   | 0.7071   | 0.2929   | 0.4949   |          |          |
| 4   | CS-RBF1 | 0.7172   | 0.9293   | 0.3535   | 0.1818   |          |          |          |          |          |
|     | CS1     | 0.7576   | 0.8485   | 1.0000   | 0.0707   |          |          |          |          |          |
|     | CS2     | 0.4242   | 0.5758   | 1.0000   | 0.7677   |          |          |          |          |          |
|     | Gaussian| 1.0000   | 0        | 0.2727   | 0.7071   |          |          |          |          |          |

### 4.3. Test 2: Parabola function

The second case to investigate is a parabola described as the following equation.

$$y = \frac{x^2}{6} - x + 4$$  \hspace{1cm} (12)

The data points are generated similarly to the previous example and figure 2 depicts the graph with nodes.

![Figure 2. Data with 100 points for parabolas trend case.](image)

With using RC Algorithm, the parameters $\sigma$ and $m$ are optioned for both training and validation cases are shown in table 4.

### Table 4. Training and validation errors for candidate models for parabolas trend case.

| RBF      | $\sigma$ | $m$ | Training MSE | Validation MSE |
|----------|----------|-----|--------------|----------------|
| CS-RBF1  | -        | 29  | 0.17637      | 0.52294        |
| CS1      | -        | 13  | 0.22069      | 0.44781        |
| CS2      | -        | 22  | 0.18575      | 0.51042        |
| Gaussian I | 0.2  | 9   | 0.24256      | 0.3985         |
| Gaussian II | 0.3 | 7   | 0.24649      | 0.3960         |
| Gaussian III | 0.4 | 5   | 0.24742      | 0.39345        |
| Gaussian IV | 0.5 | 5   | 0.24736      | 0.39365        |
| Gaussian V | 0.6  | 4   | 0.25371      | 0.39465        |
| Gaussian VI | 0.7 | 4   | 0.25203      | 0.39386        |
Thus, based on RC Algorithm, by using Gaussian V with $\sigma = 0.6$ and $m = 4$ for this data set. When using the same numbers of centres, table 5 and table 6 provide the main results of this case. Figure 3 illustrates the predicted trend with all the testing data points.

**Table 5. Results comparison when using different numbers of centres and RBFs for parabolas trend case.**

| RBF       | $m = 9, \sigma = 0.2$ | $m = 7, \sigma = 0.3$ | $m = 5, \sigma = 0.4$ | $m = 4, \sigma = 0.6$ |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|
|           | Training Error | Validation Error | Training Error | Validation Error | Training Error | Validation Error | Training Error | Validation Error |
| CS-RBF1   | 0.2432     | 0.41758           | 0.24342          | 0.40239           | 0.26359       | 0.3919           | 0.25937       | 0.39388           |
| CS1       | 0.22953    | 0.43281           | 0.24674          | 0.39482           | 0.26100       | 0.3992           | 0.25813       | 0.40699           |
| CS2       | 0.24407    | 0.40677           | 0.25483          | 0.40454           | 0.25456       | 0.39304          | 0.24952       | 0.40052           |
| Gaussian  | 0.24256    | 0.3985            | 0.24649          | 0.3960            | 0.24742       | 0.39345          | 0.25371       | 0.39465           |

**Table 6. Listing of basis function centres for parabolas trend case.**

| $m$ | RBF     | $\mu_1$ | $\mu_2$ | $\mu_3$ | $\mu_4$ | $\mu_5$ | $\mu_6$ | $\mu_7$ | $\mu_8$ |
|-----|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 9   | CS-RBF1 | 0.9293  | 0.8182  | 0.1414  | 0.2525  | 0.5354  | 0.7172  | 0.0707  | 0.3232  |
|     | CS1     | 0.5051  | 0.1616  | 0.0707  | 0.8485  | 0.9293  | 0.7576  | 0.4141  | 0.6667  | 1.0000  |
|     | CS2     | 0.4747  | 1.0000  | 0.4242  | 0.5253  | 0.2323  | 0.6162  | 0.0909  | 0.7677  | 0.9596  |
|     | Gaussian | 0       | 1.0000  | 0.0808  | 0.9192  | 0.2121  | 0.7879  | 0.3535  | 0.6465  | 0.5051  |
| 5   | CS-RBF1 | 0.4646  | 1.0000  | 0.7172  | 0.0330  | 0.9697  |        |        |        |
|     | CS1     | 0.7576  | 0.0707  | 0.6667  | 0.8485  |        |        |        |        |
|     | CS2     | 0       | 0.7172  | 1.0000  | 0.3838  | 0.4747  |        |        |        |
|     | Gaussian | 1.0000  | 0       | 0.8081  | 0.2020  | 0.5051  |        |        |        |
| 4   | CS-RBF1 | 0.9697  | 0.7879  | 0.6465  | 0        |        |        |        |        |
|     | CS1     | 0.1616  | 0       | 1.0000  | 0.4141  |        |        |        |        |
|     | CS2     | 1.0000  | 0.4747  | 0.0404  | 0        |        |        |        |        |
|     | Gaussian | 1.0000  | 0       | 0.7172  | 0.3030  |        |        |        |        |

**Figure 3.** Predicted training trend produced (a) and validation trend produced (b) by using 4 centres for parabolas trend case.

4.4. **Test 3: Sine interpolation**

In the final case study, a non-linear sine function expressed below is studied.

$$y = \sin(x) + \varepsilon$$

(13)
A set of 100 data on $[0, 2\pi]$ is generated and the noise is set to be a Gaussian distribution with mean zero and standard deviation 0.5, as depicted in figure 4.

![Figure 4. Data with 100 points for sine trend case.](image)

With using RC Algorithm, the parameters $\sigma$ and $m$ are optioned for both training and validation cases are shown in table 7.

**Table 7.** Training and validation errors for candidate models for sine trend case.

| RBF     | $\sigma$ | $m$ | MSE  
|---------|----------|-----|------
|         |          |     | Training | Validation |
| CS RBF1 | -        | 29  | 0.22723  | 0.78921  |
| CS1     | -        | 13  | 0.30183  | 0.64521  |
| CS2     | -        | 22  | 0.24992  | 0.73249  |
| Gaussian I | 0.2   | 9   | 0.36463  | 0.53588  |
| Gaussian II | 0.3   | 7   | 0.36596  | 0.53268  |
| Gaussian III | 0.4  | 5   | 0.37101  | 0.52852  |
| Gaussian IV | 0.5   | 5   | 0.37275  | 0.52933  |
| Gaussian V  | 0.6    | 4   | 0.37927  | 0.52375  |
| Gaussian VI | 0.7    | 4   | 0.37985  | 0.52492  |

Thus, based on RC Algorithm, by using Gaussian V with $\sigma = 0.6$ and $m = 4$ for this data set. When using the same numbers of centres, table 8 and table 9 provide the main results of this case. Figure 5 illustrates the predicted trend with all the testing data points.

**Table 8.** Results comparison when using different numbers of centres and RBFs for sine trend case.

| RBF     | $m = 9, \sigma = 0.2$ | $m = 7, \sigma = 0.3$ | $m = 5, \sigma = 0.5$ | $m = 4, \sigma = 0.6$ |
|---------|----------------------|----------------------|----------------------|----------------------|
|         | Training Error | Validation Error | Training Error | Validation Error | Training Error | Validation Error | Training Error | Validation Error |
| CS RBF1 | 0.34868    | 0.55688   | 0.37794    | 0.53255   | 0.45835    | 0.66754   | 0.46907    | 0.63571   |
| CS1     | 0.35783    | 0.54659   | 0.34087    | 0.57013   | 0.47041    | 0.66921   | 0.38884    | 0.53472   |
| CS2     | 0.33763    | 0.58504   | 0.36135    | 0.56636   | 0.41765    | 0.55269   | 0.52803    | 0.80572   |
| Gaussian| 0.36463    | 0.53588   | 0.36596    | 0.53268   | 0.37275    | 0.52933   | 0.37927    | 0.52375   |
Table 9. Listing of basis function centres for sine trend case.

| m | RBF   | $\mu_1$ | $\mu_2$ | $\mu_3$ | $\mu_4$ | $\mu_5$ | $\mu_6$ | $\mu_7$ | $\mu_8$ |
|---|-------|---------|---------|---------|---------|---------|---------|---------|---------|
| 7 | CS-RBF1 | 0.8586  | 0.0303  | 1.0000  | 0.1414  | 0       | 0.2828  | 0.5051  |
|   | CS1    | 0.3333  | 0.5051  | 0.2424  | 0.7576  | 0.8485  | 0.6667  | 0.5859  |
|   | CS2    | 0.5253  | 0.1919  | 0.2828  | 0.7172  | 0.3838  | 0.5758  | 0.8586  |
|   | Gaussian | 0       | 1.0000  | 0.1010  | 0.8889  | 0.7071  | 0.2929  | 0.4949  |
| 5 | CS-RBF1 | 0.6768  | 0.2828  | 0.7172  | 0.4242  | 0       |         |         |
|   | CS1    | 0.7576  | 0.4141  | 0.2424  | 0.1616  | 0       |         |         |
|   | CS2    | 0.6162  | 0.2323  | 0.2828  | 0.9091  | 0.1919  |         |         |
|   | Gaussian | 1.0000  | 0       | 0.1818  | 0.8081  | 0.4949  |         |         |
| 4 | CS-RBF1 | 0.7475  | 0.4242  | 0.6768  | 0.2121  |         |         |         |
|   | CS1    | 0.7576  | 0       | 0.3333  | 0.8485  |         |         |         |
|   | CS2    | 0.7172  | 0.1919  | 0.4747  | 0.0404  |         |         |         |
|   | Gaussian | 1.0000  | 0       | 0.7172  | 0.3030  |         |         |         |

Figure 5. Predicted training trend produced (a) and validation trend produced (b) by using 4 centres for sine trend case.

5. Conclusion

The investigation begins with the observation that RBF-pattern recognition problem relies highly on the choice of what is called ‘shape parameter’ which requires a user’s pre-judgement. It then comes to attention an alternative way to avoid this difficulty by using RBFs that contain no-parameter. For this, three non-parameterized RBF have been explored numerically. For the comparison purpose, the RC algorithm normally used with Gaussian RBF is utilized for suggestion for the number of suitable centres. Three trends of data patterns are tested with the scheme and the main findings are;

- With AC-algorithm, it is found that CS-RBF1, CS1, and CS2 have appeared to be slightly overfitting. This figure is to be further investigated.

- With appropriate choice of number of centres, these selected non-parameterized RBFs are found to perform equally well when compared to the famous Gaussian.

The next step of this study is to tackle problems in more dimensions and with more complexity.

Acknowledgement

The authors would like to express our sincere appreciation to the Centre of Excellence in Mathematics, Thailand, for their kind support.
References

[1] Holmstrom L and Koistinen P 2010 Pattern recognition Inc. WIREs Comp Stat 2 404–13
[2] Bhowmik M K, Bhattacharjee D, Nasipuri M, Basu D K and Kundu M 2009 Classification of fused images using radial basis function neural network for human face recognition 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC) (Coimbatore, India. : IEEE)
[3] Escobar C A and Menendez R M 2017 Machine Learning and Pattern Recognition Techniques for Information Extraction to Improve Production Control and Design Decisions Advances in Data Mining Conference: Industrial Conference on Data Mining
[4] Zarbakhsh P and Addeh A 2018 Breast cancer tumor type recognition using graph feature selection technique and radial basis function neural network with optimal structure. J. Cancer Res. Ther. 14 625-33
[5] Moody J and Darken C J 1989 Fast Learning in Networks of Locally-tuned Processing Units Neural Comput. 1 281–94
[6] Chen S, Cowan C F N and Grant P M 1991 Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks IEEE Transactions on Neural Networks 2 302–9.
[7] Shin M 1998 Design and Evaluation of Radial Basis Function Model for Function Approximation Ph.D Dissertation, Syracuse University
[8] Buhmann M D 1998 Radial functions on compact support, in Proceedings of the Edinburgh Mathematical society (Series 2) 41 33–46
[9] Shin M and Park C 2000 A Radial Basis Function Approach to Pattern Recognition and Its Applications ETRI Journal 22