Radial Basis Function Classifier Construction Using Particle Swarm Optimisation Aided Orthogonal Forward Regression

Sheng Chen[1], Xia Hong[2], Chris J. Harris[1]

[1]School of Electronics and Computer Science
University of Southampton, Southampton SO17 1BJ, UK
E-mails: {sqc,cjh}@ecs.soton.ac.uk

[2]School of Systems Engineering
University of Reading, Reading RG6 6AY, UK
E-mail: x.hong@reading.ac.uk

International Joint Conference on Neural Networks 2010

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Existing RBF Classifiers

- **Nonlinear** optimisation ⇒ optimise all RBF classifier's parameters: centres, variances or covariances and weights
  - Very "sparse" (small size), but all problems associated with "nonlinear" optimisation
- **Linear** optimisation ⇒ set RBF centres to training data and fix a variance: seek a "linear" subset classifier
  - Orthogonal least squares forward selection:
    - Sparse, good performance, and efficient construction
    - Need to specify RBF variance (via cross validation)
  - Sparse kernel modelling methods:
    - Sparse (though not as sparse as OLS), good performance
    - Need to specify kernel variance and other hyperparameters (via cross validation)

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Combined Linear/Nonlinear Learning

- Linear approach ⇒ state-of-the-art efficient ROLS-LOO, but fixed bases and a common RBF variance
- Nonlinear approach ⇒ optimise all parameters, but a too large and complex nonlinear optimisation
- Combined linear/nonlinear approach:
  - Retain advantage of linear optimisation → use orthogonal forward regression to add RBF bases one by one
  - Have tunable RBF bases for enhanced modelling capability → use nonlinear optimisation
- Each stage of OFR, optimise one tunable base, i.e. determine RBF base's centre and covariance
  - **How efficient** this combined RBF classifier modelling, in comparison with state-of-the-art ROLS-LOO**?**

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Tunable-RBF Classifier

- **Two-class** training set $D_N = \{\mathbf{x}_k, y_k\}_{k=1}^{N}$, where $\mathbf{x}_k \in \mathcal{R}^m$ is pattern vector and $y_k \in \{\pm 1\}$ class label
- Construct **RBF classifier** as linear combiner of RBF bases $\{g_i(\mathbf{x}_k)\}_{i=1}^{M}$

$$\hat{y}_k = f^{[M]}(\mathbf{x}_k) = \sum_{i=1}^{M} w_i g_i(\mathbf{x}_k)$$

where w_i are weights, with estimated class label

$$\tilde{y}_k = \mathrm{sgn}(\hat{y}_k)$$

- Generic **RBF base** is given by

$$g_i(\mathbf{x}) = K\left(\sqrt{(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)}\right)$$

where $\boldsymbol{\mu}_i$: ith centre vector, $\boldsymbol{\Sigma}_i$: ith diagonal covariance matrix, and $K(\bullet)$: chosen basis function

Orthogonal Decomposition

- **Regression** model on training set D_N: $\mathbf{y} = \mathbf{G}_M \mathbf{w}_M + \mathbf{e}$
- **Orthogonal decomposition** of regression matrix, $\mathbf{G}_M = \mathbf{P}_M \mathbf{A}_M$:

$$\mathbf{A}_M = \begin{bmatrix} 1 & \alpha_{1,2} & \cdots & \alpha_{1,M} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \alpha_{M-1,M} \\ 0 & \cdots & 0 & 1 \end{bmatrix}$$

$\mathbf{P}_M = [\mathbf{p}_1 \ \mathbf{p}_2 \cdots \mathbf{p}_M]$ is orthogonal, $\mathbf{A}_M \mathbf{w}_M = \boldsymbol{\theta}_M$, and equivalently:

$$\mathbf{y} = \mathbf{G_M w_M} + \mathbf{e} \Leftrightarrow \mathbf{y} = \mathbf{P_M} \boldsymbol{\theta}_\mathbf{M} + \mathbf{e}$$

- After nth stage of OFR, n bases are constructed $\mathbf{G}_n = [\mathbf{g}_1 \cdots \mathbf{g}_n]$ with corresponding $\mathbf{P}_n = [\mathbf{p}_1 \cdots \mathbf{p}_n]$ and $\mathbf{A}_n$, while kth row of $\mathbf{P}_n$ is denoted as $[p_1(k) \cdots p_n(k)]$

LOO Classification

- Define **leave-one-out** *n*-term classifier's output

$$\hat{y}_k^{[n,-k]} = f^{[n,-k]}(\mathbf{x}_k)$$

- LOO **signed decision variable** $s_k^{[n,-k]} = y_k \hat{y}_k^{[n,-k]} = \phi_k^{[n]} / \eta_k^{[n]}$ with

$$\eta_k^{[n]} = \eta_k^{[n-1]} - p_n^2(k) / \left(\mathbf{p}_n^T \mathbf{p}_n + \lambda\right)$$

$$\phi_k^{[n]} = \phi_k^{[n-1]} + y_k \, \theta_n \, p_n(k) - p_n^2(k) / \left(\mathbf{p}_n^T \mathbf{p}_n + \lambda\right)$$

where λ is a regularisation parameter

- LOO **misclassification rate** can then be computed efficiently

$$J_n = \frac{1}{N} \sum_{k=1}^{N} \mathcal{I}_d\left(s_k^{[n,-k]}\right)$$

where indicator $\mathcal{I}_d(y) = 1$ if $y \leq 0$ and $\mathcal{I}_d(y) = 0$ if $y > 0$

Nonlinear Optimisation in OFR

- At nth stage of OFR, determine nth RBF base, i.e. its nonlinear parameters $\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n$, by solving nonlinear optimisation

$$\min_{\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n} J_n\left(\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n\right)$$

- For LOO criterion J_n, there exists an "optimal" model size M: for $n \leq M$, J_n decreases as model size n increases while

$$J_M \leq J_{M+1}$$

  Thus, OFR construction procedure is automatically terminated when above condition holds, yielding an M-base model

- We propose to use particle swarm optimisation,

  A population based stochastic optimisation method inspired by social behaviour of bird flocks or fish schools (Swarm Intelligence)

Outline

1 Motivations
- Existing Approaches
- Our Novelty

2 PSO Aided OFR Based RBF Classifier
- Tunable RBF Modelling
- PSO Aided OFR Algorithm

3 Experimental Results
- Breast Cancer Data
- Diabetes Data
- Thyroid Data

4 Conclusions

Particle Swarm Optimisation

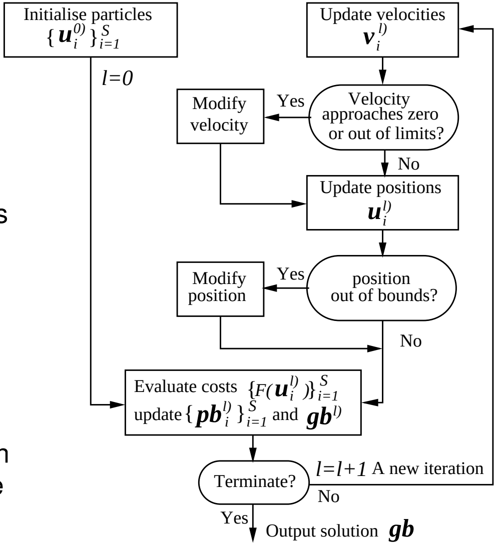
- Solving generic optimisation

$$\mathbf{u}_{\text{opt}} = \arg \min_{\mathbf{u} \in \prod_{j=1}^{m'} \mathsf{P}_j} F(\mathbf{u})$$

$\mathbf{u} = [u_1 \cdots u_{m'}]^T$ is parameter vector to be optimised, $F(\bullet)$ is cost, and search space

$$\prod_{j=1}^{m'} \mathsf{P}_j = \prod_{j=1}^{m'} [P_{j,\min}, \; P_{j,\max}]$$

- A **swarm** of **particles**, $\{\mathbf{u}_i^{(l)}\}_{i=1}^{S}$, are evolved in search space, where S is swarm size and l denotes iteration index



PSO Algorithm Adopted

- Each particle remembers its best position visited – *cognitive information*, $\mathbf{pb}_i^{l)}$, $1 \leq i \leq S$
- Every particle knows best position visited among entire swarm – *social information*, $\mathbf{gb}^{l)}$
- Each particle has a **velocity** $\mathbf{v}_i^{l)}$ to direct its "flying", and

$$\mathbf{v}_i^{l)} \in \prod_{j=1}^{m'} \mathrm{V}_j = \prod_{j=1}^{m'} [-V_{j,\max}, \;\; V_{j,\max}]$$

- In our application, $m' = 2m$, each $\mathbf{u}_i^{l)}$ contains a candidate solution for $(\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$, and cost function $F(\mathbf{u}) = J_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

PSO Procedure

- **a) Swarm initialisation**: Set iteration index $l = 0$ and randomly generate $\{\mathbf{u}_i^{l)}\}_{i=1}^{S}$ in search space $\prod_{j=1}^{m'} \mathrm{P}_j$
- **b) Swarm evaluation**: Particle $\mathbf{u}_i^{l)}$ has cost $F(\mathbf{u}_i^{l)})$, based on which $\mathbf{pb}_i^{l)}$, $1 \le i \le S$, and $\mathbf{gb}^{l)}$ are updated
- **c) Swarm update**: Velocities and positions are updated

$$\mathbf{v}_i^{l+1)} = w_{\mathrm{I}} * \mathbf{v}_i^{l)} + rand() * c_1 * (\mathbf{pb}_i^{l)} - \mathbf{u}_i^{l)}) + rand() * c_2 * (\mathbf{gb}^{l)} - \mathbf{u}_i^{l)})$$

$$\mathbf{u}_i^{l+1)} = \mathbf{u}_i^{l)} + \mathbf{v}_i^{l+1)}$$

- **d) Termination**: If maximum number of iterations $l_{\max}$ is reached, terminate with solution $\mathbf{gb}^{l_{\max})}$; otherwise, $l = l + 1$ and goto **b)**

PSO Algorithmic Parameters

- **Inertial weight** $w_{\mathrm{I}} = rand()$, other alternative is $w_{\mathrm{I}} = 0$ or w_{I} set to a small positive constant
- **Time varying acceleration coefficients**

  $$c_1 = (0.5 - 2.5) * l/l_{\max} + 2.5, \;\; c_2 = (2.5 - 0.5) * l/l_{\max} + 0.5$$

  - Initially, large cognitive component and small social component help particles to exploit better search space
  - Later, small cognitive component and large social component help particles to converge quickly to a minimum
- $S = 10$ to 20 appropriate for small to medium size problems, and empirical results suggest $I_{\max} = 20$ is often sufficient
- **Search space** is specified by problem, **velocity space** can be determined with $V_{j,\max} = 0.5 * (P_{j,\max} - P_{j,\min})$

Computational Complexity

- Let complexity of evaluating cost function once be $C_{\text{single}} \Rightarrow$ total complexity in determining one RBF node is

$$C_{\text{total}} = I_{\max} \times S \times C_{\text{single}}$$

  - Complexity of one LOO cost evaluation and associated column orthogonalisation is order of $N \Rightarrow C_{\text{single}} = \mathcal{O}(N)$

- Complexity of **PSO-aided OFR** in constructing M tunable-bases

$$C_{\text{PSO-OFR}} = (M + 1) \times I_{\max} \times S \times \mathcal{O}(N)$$

- Complexity of **ROLS-LOO** in selecting M' fixed-bases from N-candidate set is

$$C_{\text{ROLS}} = \left(M' + 1\right) \times N \times \mathcal{O}(N)$$

- PSO-aided OFR is generally simpler for large data set:
  $M < M'$, typically $I_{\max} \times S \leq 400$: when $N \geq 400$, $C_{\text{PSO-OFR}} < C_{\text{ROLS}}$

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Breast Cancer Data Set

Average classification test error rate in % over 100 realizations

method	RBF type	test error rate	model size
RBF-Network	tunable	27.64 ± 4.71	5
AdaBoost RBF-Network	tunable	30.36 ± 4.73	5
LP-Reg-AdaBoost (-"-)	tunable	26.79 ± 6.08	5
QP-Reg-AdaBoost (-"-)	tunable	25.91 ± 4.61	5
AdaBoost-Reg (-"-)	tunable	26.51 ± 4.47	5
SVM with RBF-Kernel	fixed	26.04 ± 4.74	unavailable
Kernel Fisher Discriminant	fixed	24.77 ± 4.63	200
ROLS-LOO	fixed	25.74 ± 5.00	6.0 ± 2.0
PSO OFR-LOO	tunable	23.04 ± 3.41	2.8 ± 0.9

Data and first 7 results from:
`http://ida.first.fhg.de/projects/bench/benchmarks.htm`

PSO OFR-LOO: $S = 10$ and $I_{\max} = 20$ with complexity of $760 \cdot \mathcal{O}(200)$

ROLS-LOO: with complexity of $1400 \cdot \mathcal{O}(200)$, given RBF variance

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Diabetis Data Set

Average classification test error rate in % over 100 realizations

method	RBF type	test error rate	model size
RBF-Network	tunable	24.29 ± 1.88	15
AdaBoost RBF-Network	tunable	26.47 ± 2.29	15
LP-Reg-AdaBoost (-"-)	tunable	24.11 ± 1.90	15
QP-Reg-AdaBoost (-"-)	tunable	25.39 ± 2.20	15
AdaBoost-Reg (-"-)	tunable	23.79 ± 1.80	15
SVM with RBF-Kernel	fixed	23.53 ± 1.73	unavailable
Kernel Fisher Discriminant	fixed	23.21 ± 1.63	468
ROLS-LOO	fixed	23.00 ± 1.70	6.0 ± 1.0
PSO OFR-LOO	tunable	21.87 ± 1.24	3.5 ± 1.4

Data and first 7 results from:
`http://ida.first.fhg.de/projects/bench/benchmarks.htm`

PSO OFR-LOO: $S = 10$ and $I_{\max} = 20$ with complexity $900 \cdot \mathcal{O}(468)$

ROLS-LOO: with complexity $3276 \cdot \mathcal{O}(468)$, given RBF variance

Outline

1. Motivations
   - Existing Approaches
   - Our Novelty
2. PSO Aided OFR Based RBF Classifier
   - Tunable RBF Modelling
   - PSO Aided OFR Algorithm
3. Experimental Results
   - Breast Cancer Data
   - Diabetes Data
   - Thyroid Data
4. Conclusions

Thyroid Data Set

Average classification test error rate in % over 100 realizations

method	RBF type	test error rate	model size
RBF-Network	tunable	4.52 ± 2.12	8
AdaBoost RBF-Network	tunable	4.40 ± 2.18	8
LP-Reg-AdaBoost (-"-)	tunable	4.59 ± 2.22	8
QP-Reg-AdaBoost (-"-)	tunable	4.35 ± 2.18	8
AdaBoost-Reg (-"-)	tunable	4.55 ± 2.19	8
SVM with RBF-Kernel	fixed	4.80 ± 2.19	unavailable
Kernel Fisher Discriminant	fixed	4.20 ± 2.07	140
ROLS-LOO	fixed	4.80 ± 2.20	4.6 ± 1.0
PSO OFR-LOO	tunable	2.48 ± 1.41	3.5 ± 0.8

Data and first 7 results from:
`http://ida.first.fhg.de/projects/bench/benchmarks.htm`

PSO OFR-LOO: $S = 20$ and $I_{\max} = 20$ with complexity $1800 \cdot \mathcal{O}(140)$

ROLS-LOO: with complexity $784 \cdot \mathcal{O}(140)$, given RBF variance

Conclusions

- We have developed a PSO aided OFR based algorithm for constructing tunable RBF classifiers, which combines
  - advantages of “linear” learning (orthogonal forward regression selects RBF bases one by one), and
  - advantages of “nonlinear” learning (particle swarm optimisation optimises one base at each OFR stage)
- Compared with best ROLS-LOO algorithm for selecting subset RBF model from full fixed-base candidate set, the proposed method offers:
  - better test performance, smaller classifier size, and lower complexity in classifier construction process