INTERNATIONAL JOURNAL OF SYSTEMS SCIENCE, 2017
VOL. 48, NO. 10, 2195–2201
https://doi.org/10.1080/00207721.2017.1311383

Taylor & Francis
Taylor & Francis Group

# $l^1$-norm penalised orthogonal forward regression

Xia Hong[a], Sheng Chen[b,c], Yi Guo[d] and Junbin Gao[e]

[a]Department of Computer Science, School of Mathematical, Physical and Computational Sciences, University of Reading, Reading, UK; [b]Electronics and Computer Science, University of Southampton, Southampton, UK; [c]Department of Electrical and Computuer Engineering , Faculty of Engineering, King Abdulaziz University, Jeddah, Saudi Arabia; [d]CSIRO Mathematics and Information Sciences, North Ryde, Australia; [e]Discipline of Business Analytics, University of Sydney Business School, University of Sydney, Camperdown, Australia

**ABSTRACT**
A $l^1$-norm penalised orthogonal forward regression ($l^1$-POFR) algorithm is proposed based on the concept of leave-one-out mean square error (LOOMSE), by defining a new $l^1$-norm penalised cost function in the constructed orthogonal space and associating each orthogonal basis with an individually tunable regularisation parameter. Due to orthogonality, the LOOMSE can be analytically computed without actually splitting the data-set, and moreover a closed form of the optimal regularisation parameter is derived by greedily minimising the LOOMSE incrementally. We also propose a simple formula for adaptively detecting and removing regressors to an inactive set so that the computational cost of the algorithm is significantly reduced. Examples are included to demonstrate the effectiveness of this new $l^1$-POFR approach.

**ARTICLE HISTORY**
Received 13 May 2016
Accepted 20 March 2017

**KEYWORDS**
Cross-validation; forward regression; leave-one-out errors; regularisation

## 1. Introduction

One of the main aims in data modelling is good generalisation, i.e. the model's capability to approximate accurately the system output for unseen data. Sparse models can be constructed using the $l^1$-penalised cost function, e.g. the basis pursuit or least absolute shrinkage and selection operator (LASSO) (Chen, Donoho, & Saunders, 1998; Efron, Johnstone, Hastie, & Tibshirani, 2004; Tibshirani, 1996). Based on a fixed single $l^1$-penalised regularisation parameter, the LASSO can be configured as a standard quadratic programming optimisation problem. By exploiting piecewise linearity of the problem, the least angle regression procedure (Efron et al., 2004) was developed for solving the problem efficiently. Note that the computational efficiency in LASSO is facilitated by a *single* regularisation parameter setting. For more complicated constraints, e.g. multiple regularisers, the cross-validation by actually splitting data-sets as the means of evaluating model generalisation comes with considerably large overall computational overheads.

Fundamental to evaluate model generalisation capability is the concept of cross-validation (Rao, Fung, & Rosales, 2008; Stone, 1974), and one commonly used version of cross-validation is the leave-one-out (LOO) cross-validation. For the linear-in-the-parameter models, the LOO mean square error (LOOMSE) can be calculated without actually splitting the training data-set and estimating the associated models, by making use of Sherman–Morrison–Woodbury theorem (Sherman & Morrison, 1950). Using the LOOMSE as the model term selective criterion, an orthogonal forward regression (OFR) procedure was introduced in Hong, Sharkey, and Warwick (2003). Furthermore, the $l^2$-norm based regularisation techniques (MacKay, 1991; Orr, 1995) were incorporated into the orthogonal least squares (OLS) algorithm of Chen, Billings, and Luo (1989) to produce a regularised OLS algorithm that carries out model term selection while reduces the variance of parameter estimate simultaneously (Chen, Hong, & Harris, 2003). The optimisation of $l^1$-norm regulariser with respect to model generalisation analytically is however less studied (Ji, Xue, & Carin, 2008).

We propose a $l^1$-norm penalised OFR ($l^1$-POFR) algorithm to carry out the regulariser optimisation as well as model term selection and parameter estimation simultaneously in an OFR manner. The algorithm is based on a new $l^1$-norm penalised cost function with multiple $l^1$ regularisers, each of which is associated with an orthogonal basis vector, by orthogonal decomposition of the regression matrix of the selected model terms. We derive a closed form of the optimal regularisation parameter by greedily minimising the LOOMSE incrementally. To save computational costs, an inactive set is used along the OFR process by predicting whether any model terms will be unselectable in future regression steps.

## 2. Preliminaries

Consider the general nonlinear system represented by the nonlinear model (Chen & Billings, 1989):

$$y(k) = f(\boldsymbol{x}(k)) + \upsilon(k), \tag{1}$$

where $\boldsymbol{x}(k) = \left[x_1(k)\ x_2(k) \cdots x_m(k)\right]^{\mathrm{T}} \in \mathbb{R}^m$ denotes the input vector at sample time index $k$ and $y(k)$ is the system output variable, respectively, while $v(k)$ denotes the system white noise and $f(\bullet)$ is the unknown system mapping.

The unknown system (1) is to be identified based on an observation data-set $D_N = \{\boldsymbol{x}(k), y(k)\}_{k=1}^N$ using a linear-in-the-parameter model of the form:

**CONTACT** Xia Hong ✉ x.hong@reading.ac.uk

$$\widehat{y}^{(M)}(k) = f^{(M)}(\boldsymbol{x}(k)) = \sum_{i=1}^{M} \theta_i \phi_i(\boldsymbol{x}(k)), \tag{2}$$

where $\widehat{y}^{(M)}(k)$ is the model prediction output for $\boldsymbol{x}(k)$ based on the $M$-term regression model, and $M$ is the total number of nonlinear regressors, while $\theta_i$ are the model weights. While there exist many suitable choices for regressor, without loss of generality, we choose $\phi_i(\boldsymbol{x})$ to be Gaussian radial basis function (RBF)

$$\phi_i(\boldsymbol{x}) = e^{-\frac{\|\boldsymbol{x}-\boldsymbol{c}_i\|^2}{2\tau^2}} \tag{3}$$

in which $\boldsymbol{c}_i = [c_{1,i}\ c_{2,i} \cdots c_{m,i}]^{\mathrm{T}}$ is known as the centre vector of the $i$th RBF unit and $\tau$ is an RBF width parameter. We assume that each RBF unit is placed on a training data, namely, all the RBF centre vectors $\{\boldsymbol{c}_i\}_{i=1}^{M}$ are selected from the training data $\{\boldsymbol{x}(k)\}_{k=1}^{N}$, and the RBF width $\tau$ has been predetermined, for example, using cross-validation.

Let us denote $e^{(M)}(k) = y(k) - \widehat{y}^{(M)}(k)$ as the $M$-term modelling error for the input data $\boldsymbol{x}(k)$. Over the training data-set $D_N$, further denote $\boldsymbol{y} = [y(1)\ y(2) \cdots y(N)]^{\mathrm{T}}$, $\boldsymbol{e}^{(M)} = \left[e^{(M)}(1)\ e^{(M)}(2) \cdots e^{(M)}(N)\right]^{\mathrm{T}}$, and $\boldsymbol{\Phi}_M = \left[\boldsymbol{\phi}_1\ \boldsymbol{\phi}_2 \cdots \boldsymbol{\phi}_M\right]$ with $\boldsymbol{\phi}_n = \left[\phi_n(\boldsymbol{x}(1))\ \phi_n(\boldsymbol{x}(2)) \cdots \phi_n(\boldsymbol{x}(N))\right]^{\mathrm{T}}$, $1 \le n \le M$. We have the $M$-term model in the matrix form of

$$\boldsymbol{y} = \boldsymbol{\Phi}_M \boldsymbol{\theta}_M + \boldsymbol{e}^{(M)}, \tag{4}$$

where $\boldsymbol{\theta}_M = \left[\theta_1\ \theta_2 \cdots \theta_M\right]^{\mathrm{T}}$. Let an orthogonal decomposition of the regression matrix $\boldsymbol{\Phi}_M$ be

$$\boldsymbol{\Phi}_M = \boldsymbol{W}_M \boldsymbol{A}_M, \tag{5}$$

where

$$\boldsymbol{A}_M = \begin{bmatrix} 1 & a_{1,2} & \cdots & a_{1,M} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_{M-1,M} \\ 0 & \cdots & 0 & 1 \end{bmatrix} \tag{6}$$

and

$$\boldsymbol{W}_M = \left[\boldsymbol{w}_1\ \boldsymbol{w}_2 \cdots \boldsymbol{w}_M\right] \tag{7}$$

with columns satisfying $\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{w}_j = 0$, if $i \neq j$. The regression model (4) can alternatively be expressed as

$$\boldsymbol{y} = \boldsymbol{W}_M \boldsymbol{g}_M + \boldsymbol{e}^{(M)}, \tag{8}$$

where $\boldsymbol{g}_M = [g_1\ g_2 \cdots g_M]^{\mathrm{T}}$ satisfies the triangular system $\boldsymbol{A}_M \boldsymbol{\theta}_M = \boldsymbol{g}_M$, which can be used to determine the original model parameter vector $\boldsymbol{\theta}_M$, given $\boldsymbol{A}_M$ and $\boldsymbol{g}_M$. The space spanned by the original model bases $\boldsymbol{\phi}_n$, $1 \le n \le M$, is the same space spanned by the orthogonal model bases $\boldsymbol{w}_n$, $1 \le n \le M$. Also since only the $k$th row of $\boldsymbol{\Phi}_M$ depends on $\boldsymbol{x}(k)$, only the $k$th row of $\boldsymbol{W}_M$ depends on $\boldsymbol{x}(k)$.

Further consider the following weighted $l^1$-norm penalised OLS criterion for the model (8):

$$L_e\left(\boldsymbol{\Lambda}_M, \boldsymbol{g}_M\right) = \left\|\boldsymbol{y} - \boldsymbol{W}_M \boldsymbol{g}_M\right\|^2 + \sum_{i=1}^{M} \lambda_i |g_i|, \tag{9}$$

where $\boldsymbol{\Lambda}_M = \mathrm{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$, which contains the regularisation parameters $\lambda_i \ge \varepsilon$, $1 \le i \le M$, and $\varepsilon > 0$ is a predetermined lower bound for the regularisation parameters. Given $\boldsymbol{\Lambda}_M$, the solution for $\boldsymbol{g}_M$ can be obtained by setting the subderivative vector of $L_e$ to zero, i.e. $\frac{\partial L_e}{\partial \boldsymbol{g}_M} = \boldsymbol{0}$, yielding

$$g_i^{(\mathrm{olasso})} = \left(\left|g_i^{(\mathrm{LS})}\right| - \frac{\lambda_i/2}{\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{w}_i}\right)_+ \mathrm{sign}\left(g_i^{(\mathrm{LS})}\right) \tag{10}$$

for $1 \le i \le M$, with the usual least square solution given by $g_i^{(\mathrm{LS})} = \frac{\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{y}}{\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{w}_i}$, and the operator $(\ )_+$

$$z_+ = \begin{cases} z, & \text{if } z > 0, \\ 0, & \text{if } z \le 0. \end{cases} \tag{11}$$

Unlike the LASSO (Chen et al., 1998; Tibshirani, 1996), our objective $L_e\left(\boldsymbol{\Lambda}_M, \boldsymbol{g}_M\right)$ is constructed on the orthogonal space and the $l^1$-norm parameter constraints are associated with the orthogonal bases $\boldsymbol{w}_i$, $1 \le i \le M$. Since the cost function (9) contains sparsity inducing $l^1$-norm, some parameters $g_i^{(\mathrm{olasso})}$ will be returned as zeros, producing a sparse model in the orthogonal space spanned by the columns of $\boldsymbol{W}_M$, which corresponds to a sparse model in the original space spanned by the columns of $\boldsymbol{\Phi}_M$.

## 3. Regularisation parameter optimisation and model construction with LOOMSE

Each OFR stage involves the joint regularisation parameter optimisation, model term selection and parameter estimation. The regularisation parameters with respect to their associated candidate regressors are optimised using the approximate LOOMSE formula that is derived in Section 3.2, and the regressor with the smallest LOOMSE is selected. This OFR procedure is inherently suboptimal as it is based on greedy incremental optimisation.

### 3.1. Model representation and LOOMSE in nth stage OFR

Consider the OFR modelling process that has produced the $(n-1)$-term model. The model output vector of this $(n-1)$-term model is given by

$$\widehat{\boldsymbol{y}}^{(n-1)} = \sum_{i=1}^{n-1} g_i^{(\mathrm{olasso})} \boldsymbol{w}_i, \tag{12}$$

and we denote the corresponding modelling error vector by $\boldsymbol{e}^{(n-1)} = \boldsymbol{y} - \widehat{\boldsymbol{y}}^{(n-1)}$.

Consider the $n$th OFR stage where $n$ columns of regressors are constructed as $\boldsymbol{W}_n = [\boldsymbol{w}_1\ \boldsymbol{w}_2 \cdots \boldsymbol{w}_n)]$, with $\boldsymbol{w}_i = [w_i(1)\ w_i(2) \cdots w_i(N)]^{\mathrm{T}}$. $i = 1, \ldots, n$. Clearly, the $n$th OFR stage can be represented by

$$\boldsymbol{e}^{(n-1)} = g_n \boldsymbol{w}_n + \boldsymbol{e}^{(n)}. \tag{13}$$

The model form (13) illustrates the fact that the $n$th OFR stage is simply to fit a one-variable model using the current model residual produced after the $(n-1)$th stage as the desired system output. Since $\boldsymbol{w}_n^{\mathrm{T}} \widehat{\boldsymbol{y}}^{(n-1)} = 0$, it is easy to verify that $g_n^{(\mathrm{LS})} = \frac{\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{y}}{\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{w}_n} = \frac{\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{e}^{(n-1)}}{\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{w}_n}$.

The selection of one regressor from the candidate regressors involves initially generating candidate $\boldsymbol{w}_n$ by making each candidate regressor to be orthogonal to the $(n-1)$ orthogonal basis vectors, $\boldsymbol{w}_i$ for $1 \le i \le n-1$ obtained in the previous $(n-1)$ OFR stages, followed by evaluating their contributions. Consider the case of $2|\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{e}^{(n-1)}| > \varepsilon$. Applying Equation (10) into Equation (13), we note that clearly as $\lambda_n$ decreases away from $2|\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{e}^{(n-1)}|$ towards $\varepsilon$, $g_n^{(\mathrm{olasso})}$ increases its magnitude at a linear rate to $\lambda_n$, from zero to an upper bound $|g_n^{(\mathrm{B})}|$ with

$$g_n^{(\mathrm{B})} = \left( \left| g_n^{(\mathrm{LS})} \right| - \frac{\varepsilon}{2 \boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{w}_n} \right)_+ \operatorname{sign}\left( g_n^{(\mathrm{LS})} \right). \tag{14}$$

For any candidate regressor, it is vital that we evaluate its potential model generalisation performance using the most suitable value of $\lambda_n$. The optimisation of the LOOMSE with respect to $\lambda_n$ is detailed in Section 3.2, based on the idea of the LOO cross-validation outlined below.

Suppose that we sequentially set aside each data point in the estimation set $D_N$ in turn and estimate a model using the remaining $(N-1)$ data points. The prediction error is calculated on the data point that has not been used in estimation. That is, for $k = 1, 2, \ldots, N$, the models are estimated based on $D_N \setminus (\boldsymbol{x}(k), y(k))$, respectively, and the outputs are denoted as $\widehat{y}^{(n-1,-k)}(k, \lambda_n)$. Then, the LOO prediction error based on the $k$th data sample is calculated as

$$e^{(n,-k)}(k, \lambda_n) = y(k) - \widehat{y}^{(n-1,-k)}(k, \lambda_n). \tag{15}$$

The LOOMSE is defined as the average of all these prediction errors, given by $J(\lambda_n) = E[(e^{(n,-k)}(k, \lambda_n))^2]$. Thus, the optimal regularisation parameter for the $n$th stage is given by

$$\lambda_n^{\mathrm{opt}} = \arg\min_{\lambda_n} \left\{ J(\lambda_n) = \frac{1}{N} \sum_{k=1}^{N} \left( e^{(n,-k)}(k, \lambda_n) \right)^2 \right\}. \tag{16}$$

Evaluation of $J(\lambda_n)$ by directly splitting the data-set requires extensive computational efforts. We show in Section 3.2 that $J(\lambda_n)$ can be approximately calculated without actually sequentially splitting the estimation data-set. Furthermore, we also show that the optimal value $\lambda_n^{\mathrm{opt}}$ can be obtained in a closed-form expression in the orthogonal modelling space.

### 3.2. Optimal regularisation parameter estimate

We note from Equation (10) that $g_n^{(\mathrm{olasso})} = 0$ if $2\left|\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{e}^{(n-1)}\right| < \lambda_n$, and thus a sufficient condition that a given $\boldsymbol{w}_n$ may be excluded from the candidate pool without explicitly determining $\lambda_n$ is $2\left|\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{e}^{(n-1)}\right| < \varepsilon$, which is the regulariser's lower bound, a preset value indicating the correlation of the candidate regressor. Hence, in the following we assume that $2|\boldsymbol{w}_n^{\mathrm{T}} \boldsymbol{e}^{(n-1)}| > \varepsilon$, and we have

$$\boldsymbol{g}_n^{(\mathrm{olasso})} = \boldsymbol{H}_n^{-1} \left( \boldsymbol{W}_n^{\mathrm{T}} \boldsymbol{y} - \boldsymbol{\Lambda}_n \operatorname{sign}(\boldsymbol{g}_n^{(\mathrm{LS})})/2 \right), \tag{17}$$

where $\boldsymbol{g}_n^{(\mathrm{olasso})} = [g_1^{(\mathrm{olasso})}\, g_2^{(\mathrm{olasso})} \cdots g_n^{(\mathrm{olasso})}]^{\mathrm{T}}$, $\operatorname{sign}(\boldsymbol{g}_n) = [\operatorname{sign}(g_1)\ \operatorname{sign}(g_2) \cdots \operatorname{sign}(g_n)]^{\mathrm{T}}$, and $\boldsymbol{H}_n = \boldsymbol{W}_n^{\mathrm{T}} \boldsymbol{W}_n$. Note that Equation (17) is consistent to Equation (10) for all terms with nonzero $g_i$. In the OFR procedure, any candidate terms $\boldsymbol{w}_i$ producing zero $g_i^{(\mathrm{olasso})}$ will not be selected since they will not contribute to any reduction in the LOOMSE.

The model residual is defined by

$$\begin{aligned} e^{(n)}(k, \lambda_n) &= y(k) - (\boldsymbol{g}^{(\mathrm{olasso})})^{\mathrm{T}} \boldsymbol{w}(k) = y(k) \\ &\quad - (\boldsymbol{y}^{\mathrm{T}} \boldsymbol{W}_n - (\operatorname{sign}(\boldsymbol{g}^{(\mathrm{LS})}))^{\mathrm{T}} \boldsymbol{\Lambda}_n/2) \boldsymbol{H}_n^{-1} \boldsymbol{w}(k), \end{aligned} \tag{18}$$

where $\boldsymbol{w}(k)$ denotes the transpose of the $k$th row of $\boldsymbol{W}_n$. If the data sample indexed at $k$ is removed from the estimation data-set, the LOO parameter estimator obtained by using only the $(N-1)$ remaining data points is given by

$$\begin{aligned} \boldsymbol{g}_n^{(\mathrm{olasso},-k)} &= \left( \boldsymbol{H}_n^{(-k)} \right)^{-1} \\ &\quad \times \left( \left( \boldsymbol{W}_n^{(-k)} \right)^{\mathrm{T}} \boldsymbol{y}^{(-k)} - \boldsymbol{\Lambda}_n \operatorname{sign}\left( \boldsymbol{g}^{(\mathrm{LS},-k)} \right)/2 \right) \end{aligned} \tag{19}$$

where $\boldsymbol{H}_n^{(-k)} = \left( \boldsymbol{W}_n^{(-k)} \right)^{\mathrm{T}} \boldsymbol{W}_n^{(-k)}$, $\boldsymbol{W}_n^{(-k)}$ and $\boldsymbol{y}^{(-k)}$ are the resultant regression matrix and desired output vector, respectively, by removing $(\boldsymbol{x}(k), y(k))$, i.e. $(\boldsymbol{w}^{\mathrm{T}}(k), y(k))$, from $\boldsymbol{W}(k)$ and $\boldsymbol{y}(k)$. Thus, we have

$$\boldsymbol{H}_n^{(-k)} = \boldsymbol{H}_n - \boldsymbol{w}(k) \boldsymbol{w}^{\mathrm{T}}(k), \tag{20}$$

$$\left( \boldsymbol{y}^{(-k)} \right)^{\mathrm{T}} \boldsymbol{W}_n^{(-k)} = \boldsymbol{y}^{\mathrm{T}} \boldsymbol{W}_n - y(k) \boldsymbol{w}^{\mathrm{T}}(k). \tag{21}$$

The LOO error evaluated at $k$ is given by

$$\begin{aligned} e^{(n,-k)}(k, \lambda_n) &= y(k) - \left( \boldsymbol{g}^{(\mathrm{olasso},-k)} \right)^{\mathrm{T}} \boldsymbol{w}(k) \\ &= y(k) - \left( \left( \boldsymbol{y}^{(-k)} \right)^{\mathrm{T}} \boldsymbol{W}_n^{(-k)} \right. \\ &\quad \left. - \left( \operatorname{sign}\left( \boldsymbol{g}^{(\mathrm{LS},-k)} \right) \right)^{\mathrm{T}} \boldsymbol{\Lambda}_n/2 \right) \left( \boldsymbol{H}_n^{(-k)} \right)^{-1} \boldsymbol{w}(k). \end{aligned} \tag{22}$$

Applying the matrix inversion lemma to Equation (20) yields

$$\begin{aligned} \left( \boldsymbol{H}_n^{(-k)} \right)^{-1} &= \left( \boldsymbol{H}_n - \boldsymbol{w}(k) \boldsymbol{w}^{\mathrm{T}}(k) \right)^{-1} \\ &= \boldsymbol{H}_n^{-1} + \tfrac{\boldsymbol{H}_n^{-1} \boldsymbol{w}(k) \boldsymbol{w}^{\mathrm{T}}(k) \boldsymbol{H}_n^{-1}}{1 - \boldsymbol{w}^{\mathrm{T}}(k) \boldsymbol{H}_n^{-1} \boldsymbol{w}(k)} \end{aligned} \tag{23}$$

and

$$\left( \boldsymbol{H}_n^{(-k)} \right)^{-1} \boldsymbol{w}(k) = \frac{\boldsymbol{H}_n^{-1} \boldsymbol{w}(k)}{1 - \boldsymbol{w}^{\mathrm{T}}(k) \boldsymbol{H}_n^{-1} \boldsymbol{w}(k)}. \tag{24}$$

Substituting Equations (21) and (24) into Equation (22) yields

$$\begin{aligned} & e^{(n,-k)}(k, \lambda_n) \\ &= y(k) - (\boldsymbol{y}^{\mathrm{T}} \boldsymbol{W}_n - y(k) \boldsymbol{w}^{\mathrm{T}}(k) \\ &\quad - (\operatorname{sign}(\boldsymbol{g}^{(\mathrm{LS},-k)}))^{\mathrm{T}} \boldsymbol{\Lambda}_n/2) \frac{\boldsymbol{H}_n^{-1} \boldsymbol{w}(k)}{1 - \boldsymbol{w}^{\mathrm{T}}(k) \boldsymbol{H}_n^{-1} \boldsymbol{w}(k)} \\ &= \frac{y(k) - (\boldsymbol{y}^{\mathrm{T}} \boldsymbol{W}_n - (\operatorname{sign}(\boldsymbol{g}^{\mathrm{LS},-k)})^{\mathrm{T}} \boldsymbol{\Lambda}_n/2) \boldsymbol{H}_n^{-1} \boldsymbol{w}(k)}{1 - \boldsymbol{w}^{\mathrm{T}}(k) \boldsymbol{H}_n^{-1} \boldsymbol{w}(k)}. \end{aligned} \tag{25}$$

Assuming that $\text{sign}\big(\boldsymbol{g}_n^{(\text{LS},-k)}\big) = \text{sign}\big(\boldsymbol{g}_n^{(\text{LS})}\big)$ holds for most data samples in $D_N$, and applying Equation (18) into Equation (25), we have

$$e^{(n,-k)}(k,\lambda_n) = \gamma_n(k) e^{(n)}(k,\lambda_n), \tag{26}$$

where $\gamma_n(k) = \frac{1}{1-\sum_{i=1}^{n}\left(w_i(k)\right)^2 / \boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{w}_i} > 0$, and $w_i(k)$ is the $k$th element of $\boldsymbol{w}_i$. The LOOMSE can then be calculated as

$$J\big(\lambda_n\big) = \tfrac{1}{N}\textstyle\sum_{k=1}^{N} \gamma_n^2(k)\big(e^{(n)}(k,\lambda_n)\big)^2. \tag{27}$$

Note that for $\text{sign}\big(\boldsymbol{g}_n^{(\text{LS},-k)}\big)$ and $\text{sign}\big(\boldsymbol{g}_n^{(\text{LS})}\big)$ to be different, each element in $\boldsymbol{g}_n^{(\text{LS})}$ needs to be very close to zero, which is unlikely since only the model terms satisfying $\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big| > \varepsilon/2$ are considered. Hence, we can treat $J(\lambda_n)$ given in Equation (27) as the exact LOOMSE for any $\varepsilon$ that is not too small.

We further represent Equation (18) as

$$e^{(n)}(k,\lambda_n) = \eta(k) + \frac{\lambda_n}{2\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n} w_n(k)\,\text{sign}\big(g_n^{(\text{LS})}\big), \tag{28}$$

where $\eta(k) = e^{(n-1)}(k) - g_n^{(\text{LS})} w_n(k)$ is the model residual obtained based on the least square estimate at the $n$th step stage. By setting $\frac{\partial J(\lambda_n)}{\partial \lambda_n} = 0$, we obtain $\lambda_n$ in the form of the weighted least square estimate:

$$\lambda_n = -2\,\text{sign}\big(g_n^{(\text{LS})}\big)\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{\Gamma}^{(n)}\boldsymbol{\eta} \big/ \boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{\Gamma}^{(n)}\boldsymbol{w}_n, \tag{29}$$

where $\boldsymbol{\Gamma}^{(n)} = \text{diag}\big\{\gamma_n^2(1), \gamma_n^2(2), \ldots, \gamma_n^2(N)\big\}$ and $\boldsymbol{\eta} = \big[\eta(1)\ \eta(2) \cdots \eta(N)\big]^{\mathrm{T}} \in \mathbb{R}^N$. Finally, we calculate

$$\lambda_n^{\text{opt}} = \max\Big\{\min\big\{2\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big|, \lambda_n\big\}, \varepsilon\Big\}, \tag{30}$$

in order to satisfy the constraint that $\varepsilon \le \lambda_n^{\text{opt}} \le 2\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big|$. For $\lambda_n^{\text{opt}}$ obtained using Equation (30), we consider the following two cases:

(1) If $\lambda_n^{\text{opt}} = 2\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big|$, then $g_n^{(\text{olasso})} = 0$, and this candidate regressor will not be selected.
(2) If $\varepsilon \le \lambda_n^{\text{opt}} < 2\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big|$, then calculate $J\big(\lambda_n^{\text{opt}}\big)$ based on Equation (27) as the LOOMSE for this candidate regressor.

### 3.3. Moving unselectable regressors to the inactive set

From Section 3.2 we noted that a candidate regressor satisfying $2\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big| < \varepsilon$ does not need to be considered at the $n$th stage of selection. To save computational cost, we define the inactive set $\mathcal{S}$ as the index set of the unselectable regressors removed from the pool of candidates.

In the $n$th OFR stage, all the candidate regressors in the candidate pool are made orthogonal to the previously selected $(n-1)$ regressors, and the candidate with the smallest LOOMSE value is selected as the $n$th model term $\boldsymbol{w}_n$. Denote any other candidate regressor as $\boldsymbol{w}^{(-)}$.

*Main results*: If $\big\|\boldsymbol{w}^{(-)}\big\| \cdot \big\|\boldsymbol{e}^{(n-1)}\big\| < \frac{\varepsilon}{2}$, then this candidate regressor will never be selected in further regression stages, and hence it can be moved to $\mathcal{S}$.

*Proof*: At the $(n+1)$th OFR stage, consider making the regressor $\boldsymbol{w}^{(-)}$ orthogonal to $\boldsymbol{w}_n$, and define

$$\boldsymbol{w}^{(+)} = \boldsymbol{w}^{(-)} - \frac{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}^{(-)}}{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n}\boldsymbol{w}_n. \tag{31}$$

Clearly,

$$\begin{aligned}\big\|\boldsymbol{w}^{(+)}\big\|^2 &= \left(\boldsymbol{w}^{(-)} - \tfrac{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}^{(-)}}{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n}\boldsymbol{w}_n\right)^{\mathrm{T}}\left(\boldsymbol{w}^{(-)} - \tfrac{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}^{(-)}}{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n}\boldsymbol{w}_n\right) \\ &= \big\|\boldsymbol{w}^{(-)}\big\|^2 - \tfrac{\left(\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}^{(-)}\right)^2}{\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n} \le \big\|\boldsymbol{w}^{(-)}\big\|^2. \end{aligned} \tag{32}$$

The model residual vector after the selection of $\boldsymbol{w}_n$ is

$$\boldsymbol{e}^{(n)} = \boldsymbol{e}^{(n-1)} - g_n^{(\text{olasso})}\boldsymbol{w}_n, \tag{33}$$

where $g_n^{(\text{olasso})}$ can be written as

$$g_n^{(\text{olasso})} = \left(\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)} - \frac{\lambda_n}{2}\text{sign}\big(g_n^{(\text{LS})}\big)\right)\Big/\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n. \tag{34}$$

Thus, we have

$$\big\|\boldsymbol{e}^{(n)}\big\|^2 = \big\|\boldsymbol{e}^{(n-1)}\big\|^2 - 2g_n^{(\text{olasso})}\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)} + \big(g_n^{(\text{olasso})}\big)^2\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n, \tag{35}$$

$$\big(g_n^{(\text{olasso})}\big)^2\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n = \left(\big(\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big)^2 - \lambda_n\text{sign}\big(g_n^{\text{LS)}}\big)\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)} + \frac{\lambda_n^2}{4}\right)\Big/\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n, \tag{36}$$

and

$$\begin{aligned}&2g_n^{(\text{olasso})}\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)} \\ &= \left(2\big(\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big)^2 - \lambda_n\text{sign}\big(g_n^{(\text{LS})}\big)\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\right)\Big/\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n.\end{aligned} \tag{37}$$

Substituting Equations (36) and (37) into Equation (35) yields

$$\begin{aligned}\big\|\boldsymbol{e}^{(n)}\big\|^2 &= \big\|\boldsymbol{e}^{(n-1)}\big\|^2 - \left(\big(\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big)^2 - \tfrac{\lambda_n^2}{4}\right)\Big/\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n \\ &< \big\|\boldsymbol{e}^{(n-1)}\big\|^2,\end{aligned} \tag{38}$$

due to the fact that $\big|\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{e}^{(n-1)}\big| > \frac{\lambda_n}{2}$. From Equations (32) and (38), it can be concluded that

$$\big\|\boldsymbol{w}^{(+)}\big\| \cdot \big\|\boldsymbol{e}^{(n)}\big\| < \big\|\boldsymbol{w}^{(-)}\big\| \cdot \big\|\boldsymbol{e}^{(n-1)}\big\| < \frac{\varepsilon}{2}. \tag{39}$$

Since $\|\boldsymbol{w}^{(+)}\| \cdot \|\boldsymbol{e}^{(n)}\|$ is the upper bound of $|(\boldsymbol{w}^{(+)})^{\mathrm{T}}\boldsymbol{e}^{(n)}|$, this means that this regressor will not be selected at the $(n+1)$th stage. By induction, it will never be selected in further regression stages, and hence it can be moved to $\mathcal{S}$.

## 4. The proposed $l^1$-POFR algorithm

The proposed $l^1$-POFR algorithm integrates (1) the model regressor selection based on minimising the LOOMSE; (2) regularisation parameter optimisation also based on minimising the LOOMSE; and (3) the mechanism of removing unproductive candidate regressors during the OFR procedure. Define

$$\boldsymbol{\Phi}^{(n-1)} = \left[\boldsymbol{w}_1 \cdots \boldsymbol{w}_{n-1}\ \boldsymbol{\phi}_n^{(n-1)} \cdots \boldsymbol{\phi}_M^{(n-1)}\right] \in \mathbb{R}^{N\times M}, \quad (40)$$

with $\boldsymbol{\Phi}^{(0)} = \boldsymbol{\Phi}_M$. If some of the columns in $\boldsymbol{\Phi}^{(n-1)}$ have been interchanged, this will still be referred as $\boldsymbol{\Phi}^{(n-1)}$ for notational simplicity.

**Table 1.** The $n$th stage of the selection procedure.

For $\{n \le j \le M\} \cap \{j \notin \mathcal{S}\}$, denote the $k$th element of $\boldsymbol{\phi}_j^{(n-1)}$ as $\phi_j^{(n-1)}(k)$ and compute $\alpha_j = (\boldsymbol{\phi}_j^{(n-1)})^{\mathrm{T}}\boldsymbol{e}^{(n-1)}$, and $\beta_j = \|\boldsymbol{\phi}_j^{(n-1)}\| \cdot \|\boldsymbol{e}^{(n-1)}\|$.

Step (1): If $\beta_j < \varepsilon/2$, $\mathcal{S} = \mathcal{S} \cup j$; else if $|\alpha_j| < \varepsilon/2$, set $J_n^{(j)}$ as a very large positive number so that it will not be selected in Step (4). Otherwise goto step (2).

Step (2): Calculate

$$\kappa_n^{(j)} = (\boldsymbol{\phi}_j^{(n-1)})^{\mathrm{T}}\boldsymbol{\phi}_j^{(n-1)}, \quad (41)$$

$$g_n^{(\mathrm{LS},j)} = \frac{\alpha_j}{\kappa_n^{(j)}}, \quad (42)$$

$$\boldsymbol{\Gamma}^{(n,j)} = \mathrm{diag}\left\{\frac{1}{\left(\zeta^{(n-1)}(1) - (\phi_j^{(n-1)}(1))^2/\kappa_n^{(j)}\right)^2}, \frac{1}{\left(\zeta^{(n-1)}(2) - (\phi_j^{(n-1)}(2))^2/\kappa_n^{(j)}\right)^2}, \cdots, \frac{1}{\left(\zeta^{(n-1)}(N) - (\phi_j^{(n-1)}(N))^2/\kappa_n^{(j)}\right)^2}\right\} \in \mathbb{R}^{N\times N}, \quad (43)$$

$$\boldsymbol{\eta}^{(j)} = \boldsymbol{e}^{(n-1)} - g_n^{(\mathrm{LS},j)}\boldsymbol{\phi}_j^{(n-1)}, \quad (44)$$

$$\lambda_n^{(\mathrm{opt},j)} = \max\left\{\min\left\{2|\alpha_j|, -2\mathrm{sign}(g_n^{(\mathrm{LS},j)})\kappa_n^{(j)} (\boldsymbol{\phi}_j^{(n-1)})^{\mathrm{T}}\boldsymbol{\Gamma}^{(n,j)}\boldsymbol{\eta}^{(j)}/(\boldsymbol{\phi}_j^{(n-1)})^{\mathrm{T}}\boldsymbol{\Gamma}^{(n,j)}\boldsymbol{\phi}_j^{(n-1)}\right\}, \varepsilon\right\}. \quad (45)$$

Step (3): If $\lambda_n^{(\mathrm{opt},j)} = 2|\alpha_j|$, set $J_n^{(j)}$ as a very large positive number so that it will not be selected in Step (4); otherwise calculate

$$g_n^{(\mathrm{olasso},j)} = \left(|g_n^{(\mathrm{LS},j)}| - \frac{\lambda_n^{(\mathrm{opt},j)}/2}{\kappa_n^{(j)}}\right)_+ \mathrm{sign}(g_n^{(\mathrm{LS},j)}), \quad (46)$$

$$\boldsymbol{e}^{(n,j)} = \boldsymbol{e}^{(n-1)} - g_n^{(\mathrm{olasso},j)}\boldsymbol{\phi}_j^{(n-1)}, \quad (47)$$

$$J_n^{(j)} = (\boldsymbol{e}^{(n,j)})^{\mathrm{T}}\boldsymbol{\Gamma}^{(n,j)}\boldsymbol{e}^{(n,j)}/N. \quad (48)$$

Step (4): Find

$$J_n = J_n^{(j_n)} = \min\left\{J_n^{(j)},\ \{l \le j \le M\} \cap \{j \notin \mathcal{S}\}\right\}. \quad (49)$$

Then update $\boldsymbol{e}^{(n)}$ and $g_n^{(\mathrm{olasso})}$ as $\boldsymbol{e}^{(n,j_n)}$ and $g_n^{(\mathrm{olasso},j_n)}$, respectively. The $j_n$th and the $n$th columns of $\boldsymbol{\Phi}^{(n-1)}$ are interchanged, while the $j_n$th column and the $n$th column of $\boldsymbol{A}_M$ are interchanged up to the $(n-1)$th row. This effectively selects the $n$th regressor in the subset model. The modified Gram–Schmidt orthogonalisation procedure (Chen et al., 1989) then calculates the $n$th row of the matrix $\boldsymbol{A}_M$ and transfers $\boldsymbol{\Phi}^{(n-1)}$ into $\boldsymbol{\Phi}^{(n)}$ as follows:

$$\left.\begin{array}{l} \boldsymbol{w}_n = \boldsymbol{\phi}_n^{(n-1)}, \\ a_{n,j} = \boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{\phi}_j^{(n-1)}/\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n, \{n+1 \le j \le M\} \cap \{j \notin \mathcal{S}\}, \\ \boldsymbol{\phi}_j^{(n)} = \boldsymbol{\phi}_j^{(n-1)} - a_{n,j}\boldsymbol{w}_n, \{n+1 \le j \le M\} \cap \{j \notin \mathcal{S}\}. \end{array}\right\} \quad (50)$$

Then update $\zeta^{(n)}(k) = \zeta^{(n-1)}(k) - (w_n(k))^2/\boldsymbol{w}_n^{\mathrm{T}}\boldsymbol{w}_n$ for $1 \le k \le N$.

The initial conditions are as follows. Preset $\varepsilon > 0$ as a very small value. Set $\boldsymbol{e}^{(0)} = \boldsymbol{y}$, $\zeta^{(0)}(k) = 1$ for $1 \le k \le N$, and $\mathcal{S}$ as the empty set $\emptyset$. The $n$th stage of the selection procedure is listed in Table 1. The OFR procedure is automatically terminated at the $(n_s + 1)$th stage when the condition

$$J_{n_s+1} \ge J_{n_s} \quad (51)$$

is detected, yielding a subset model with $n_s$ significant regressors. It is worth emphasising that there always exists a model size $n_s$, and for $n \le n_s$, the LOOMSE $J_n$ decreases as $n$ increases, while the condition (51) holds (Chen, Hong, Harris, & Sharkey, 2004; Hong et al., 2003).

Note that the LOOMSE is used not only for deriving the closed form of the optimal regularisation parameter estimate $\lambda_n^{\mathrm{opt}}$ but also for selecting the most significant model regressor. Specifically, a regressor is selected as the one that produces the smallest LOOMSE value as well as offering the reduction in the LOOMSE. After the $n_s$ stage when there is no reduction in the LOOMSE criterion for a few consecutive OFR stages, the model construction procedure can be terminated. Thus, the $l^1$-POFR algorithm automatically constructs a sparse $n_s$-term model, where typically $n_s \ll M$.

Also note that it is assumed that $\varepsilon$ should not be too small such that the LOOMSE estimation formula can be considered to be accurate. This means that if $\varepsilon$ is set too low, many insignificant candidate regressors will have inaccurate LOOMSE values for competition. However, we emphasise that these terms with inaccurate LOOMSE values will not be selected as the winner to enter the model. Hence in practice we only need to make sure that $\varepsilon$ is not too large, which would introduce unnecessary bias to the model parameter estimates. Clearly, a relatively larger $\varepsilon$ will save computational costs by (1) resulting in a sparser model, and (2) producing a larger sized inactive set during the OFR process.

Finally, regarding the computational complexity of the $l^1$-POFR algorithm, if the unproductive regressors are not removed to the inactive set $\mathcal{S}$ during the OFR procedure, it is well known that the computational cost is in the order of $\mathrm{O}(N)$ for evaluating each candidate regressor (Chen et al., 2004). The total computational cost then needs to be scaled by the number of evaluations in forward regression, which is $M(M - n_s)/2$. By removing unproductive regressors to $\mathcal{S}$ during the OFR procedure, the computational cost can obviously be reduced significantly. It is not possible to exactly assess the computational cost-saving due to removing the unproductive regressors, as this is problem-dependent.

## 5. Simulation study

Example 5.1: This engine data-set (Billings, Chen, & Backhouse, 1989) contains the 410 data samples of the fuel rack position (the input $u(k)$) and the engine speed (the output $y(k)$), collected from a Leyland TL11 turbocharged, direct injection diesel engine which was operated at a low engine speed. The 410 input and output data points of the engine data-set are plotted in Figure 1 (a,b). The first 210 data samples were used in training

**Table 2.** Comparison of the modelling performance for engine data. The computational cost-saving is based on the same size of model without removing unproductive regressors in the $l^1$-POFR.

| Algorithm | MSE training set | MSE test set | Model size | Cost saving |
|---|---|---|---|---|
| LROLS-LOO (Chen et al., 2004) | 0.000453 | 0.000490 | 22 | NA |
| $\varepsilon$-SVM ($\tau = 3$) | 0.000502 | 0.000482 | 208 | NA |
| $\varepsilon$-SVM ($\tau = 2.5$) | 0.000480 | 0.000475 | 208 | NA |
| $\varepsilon$-SVM ($\tau = 2$) | 0.000461 | 0.000486 | 208 | NA |
| $\varepsilon$-SVM ($\tau = 1.5$) | 0.000415 | 0.000579 | 208 | NA |
| $\varepsilon$-SVM ($\tau = 1$) | 0.000370 | 0.000794 | 208 | NA |
| LASSO ($\tau = 1.5$) | 0.000923 | 0.001010 | 70 | NA |
| LASSO ($\tau = 1$) | 0.000708 | 0.000748 | 44 | NA |
| LASSO ($\tau = 0.5$) | 0.000706 | 0.000842 | 54 | NA |
| LASSO ($\tau = 0.2$) | 0.000565 | 0.000800 | 81 | NA |
| LASSO ($\tau = 0.1$) | 0.000644 | 0.001907 | 76 | NA |
| $l^1$-POFR ($\varepsilon = 10^{-4}$) | 0.000498 | 0.000502 | 20 | 27% |
| $l^1$-POFR ($\varepsilon = 10^{-5}$) | 0.000492 | 0.000480 | 20 | 18% |
| $l^1$-POFR ($\varepsilon = 10^{-6}$) | 0.000484 | 0.000485 | 20 | 8% |
| $l^1$-POFR ($\varepsilon = 10^{-7}$) | 0.000481 | 0.000476 | 20 | 3% |
| $l^1$-POFR ($\varepsilon = 0$) | 0.000452 | 0.000472 | 21 | 0% |

and the last 200 data samples for model testing. The previous study has shown that the data-set can be modelled adequately using the system input vector $\boldsymbol{x}(k) = [y(k-1)\ u(k-1)\ u(k-2)]^{\mathrm{T}}$, and the best Gaussian RBF model was provided by the $l^2$-norm local regularisation -assisted OLS (LROLS) algorithm based on the LOOMSE (LROLS-LOO) (Chen et al., 2004) which is quoted in Table 2 for comparison. The $\varepsilon$-SVM algorithm (Gun, 1998) and the LASSO were also experimented based on the Gaussian kernel with a common variance $\tau^2$. For the $\varepsilon$-SVM, the Matlab function *quadprog.m* was used with the algorithm option set as 'interior-point-convex'. The tuning parameters in the $\varepsilon$-SVM algorithm, such as soft margin parameter $C$ (Gun, 1998), were set empirically so that the best possible result was obtained after several trials. For the LASSO, the Matlab function *lasso.m* was used with 10-fold CV being used to select the associated regularisation parameter. For both the $\varepsilon$-SVM and LASSO, we list the results obtained for a range of kernel width $\tau$ values in Table 2, for comparison.

Similar to the LROLS-LOO algorithm (Chen et al., 2004), we also used the Gaussian RBF kernel (3) for the proposed $l^1$-POFR algorithm with an empirically set $\tau = 2.5$ and the RBF centres $\boldsymbol{c}_i$ were formed using all the training data samples. With a preset value of $\varepsilon$, a sparse model of size $n_s$ was automatically selected when the condition (51) was met. Figure 1(c) illustrates

**Table 3.** Comparison of the modelling performance for Boston House Data. The results were averaged over 100 realisations and given as mean ± standarddeviation.

| Algorithm | MSE training set | MSE test set | Model size |
|---|---|---|---|
| $\varepsilon$-SVM (Gun, 1998) | $6.80 \pm 0.44$ | $23.18 \pm 9.05$ | $243 \pm 5.3$ |
| LROLS-LOO (Chen et al., 2004) | $12.97 \pm 2.67$ | $17.42 \pm 4.67$ | $58.6 \pm 11.3$ |
| NonOFR-LOO (Chen et al., 2009) | $10.10 \pm 3.40$ | $14.07 \pm 3.62$ | $34.6 \pm 8.4$ |
| LASSO ($\tau = 2$) | $8.52 \pm 3.57$ | $14.37 \pm 8.15$ | $76.8 \pm 39.7$ |
| LASSO ($\tau = 3$) | $8.55 \pm 1.07$ | $13.31 \pm 6.65$ | $68.6 \pm 29.3$ |
| LASSO ($\tau = 5$) | $10.45 \pm 1.07$ | $15.05 \pm 8.37$ | $85.9 \pm 19.7$ |
| LASSO ($\tau = 10$) | $16.42 \pm 1.78$ | $19.39 \pm 8.31$ | $29.9 \pm 21.3$ |
| $l^1$-POFR ($\varepsilon = 0.01$) | $9.99 \pm 1.37$ | $14.47 \pm 7.47$ | $30.5 \pm 5.3$ |
| $l^1$-POFR ($\varepsilon = 0.001$) | $9.24 \pm 1.57$ | $14.10 \pm 7.02$ | $34.9 \pm 7.8$ |
| $l^1$-POFR ($\varepsilon = 0.0001$) | $9.07 \pm 1.64$ | $14.02 \pm 6.85$ | $36.6 \pm 9.3$ |
| $l^1$-POFR ($\varepsilon = 0.00001$) | $9.08 \pm 1.64$ | $13.95 \pm 6.76$ | $36.5 \pm 9.3$ |

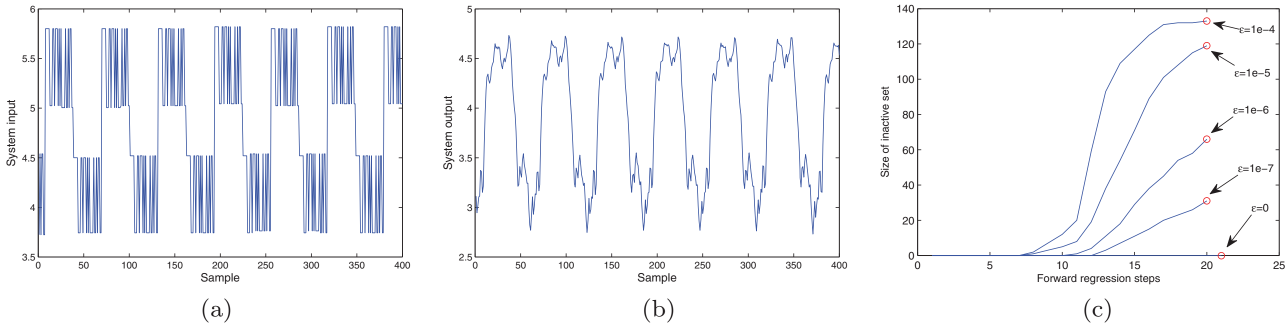

**Figure 1.** Engine data: (a) the system input $u(k)$, (b) the system output $y(k)$ and (c) the evolution of the size of $\mathcal{S}$ with respect to the chosen $\varepsilon$.

the evolution of the size of $\mathcal{S}$ with respect to a range of the preset $\varepsilon$ values. The test MSE values produced by the sparse models and the sizes of the models associated with the same range of $\varepsilon$ values are recorded in Table 2, which show that the excellent model generalisation capability of all the models generated by the proposed algorithm. Moreover, the $l^1$-POFR algorithm produces the sparsest model.

**Example 5.2** This regression benchmark data-set, Boston Housing Data, is available at the UCI repository (Frank & Asuncion, 2010). The data-set comprises 506 data points with 14 variables. The previous study (Chen et al., 2009) performed the task of predicting the median house value from the remaining 13 attributes using the $\varepsilon$-SVM (Gun, 1998), the LROLS-LOO (Chen et al., 2004) and the nonlinear OFR based on the LOOMSE (NonOFR-LOO) (Chen et al., 2009). The NonOFR-LOO algorithm (Chen et al., 2009) constructs a *nonlinear* RBF model in the OFR procedure, where each stage of the OFR determines one RBF node's centre vector and diagonal covariance matrix by minimising the LOOMSE. In the experiment study presented in Chen et al. (2009), 456 data points were randomly selected from the data-set for training and the remaining 50 data points were used to form the test set. Average results were given over 100 realisations. For each realisation, 13 input attributes were normalised so that each attribute had zero mean and standard deviation of one. We also experimented with the LASSO supplied by Matlab *lasso.m* with option set as 10-fold CV to select the associated regularisation parameter. For the LASSO, a common kernel width $\tau$ was set for constructing the kernel model from the 456 candidate regressors of each realisation, and a range of $\tau$ values were experimented.

For the $l^1$-POFR, $\tau = 15$ was empirically set for constructing 456 candidate Gaussian RBF regressors of each realisation. We experimented a range of the preset $\varepsilon$ values for the $l^1$-POFR algorithm, and the results obtained are as summarised in Table 3, in comparison with the results obtained by the $\varepsilon$-SVM and the LASSO, as well as the LROLS-LOO and NonOFR-LOO, which are quoted from the study (Chen et al., 2009).

## 6. Conclusions

We have developed an efficient data model algorithm, referred as the $l^1$-norm penalised orthogonal forward regression ($l^1$-POFR), for linear-in-the-parameternonlinear models based on a new $l^1$-norm penalised cost function defined in the constructed orthogonal modelling space. The LOOMSE is used for simultaneous model term selection and regularisation parameter estimation in a highly efficient OFR procedure. Additionally, we have proposed a lower bound of the regularisation parameters for robust LOOMSE estimation as well as detecting and removing insignificant regressors to an inactive set along the OFR process, further enhancing the efficiency of the OFR procedure. Numerical studies have been utilised to demonstrate the effectiveness of this new $l^1$-POFR approach.

## Disclosure statement

No potential conflict of interest was reported by the authors.

## References

Billings, S. A., Chen, S., & Backhouse, R. (1989). The identification of linear and nonlinear models of a turbocharged automative diesel engine. *Mechanical Systems and Signal Processings, 3*(2), 123–142.

Chen, S., & Billings, S. A. (1989). Representation of nonlinear systems: The NARMAX model. *International Journal of Control, 49*(3), 1013–1032.

Chen, S., Billings, S. A., & Luo, W. (1989). Orthogonal least squares methods and their applications to non-linear system identification. *International Journal of Control, 50*(5), 1873–1896.

Chen, S., Hong, X., & Harris, C. J. (2003). Sparse kernel regression modelling using combined locally regularised orthogonal least squares and D-optimality experimental design. *IEEE Transactions on Automatic Control, 48*(6), 1029–1036.

Chen, S., Hong, X., & Harris, C. J. (2009). Construction of tunable radial basis function networks using orthogoanl forward selection. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 39*(2), 457–466.

Chen, S., Hong, X., Harris, C. J., & Sharkey, P. M. (2004). Sparse modelling using orthogonal forward regression with PRESS statistic and regularization. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 34*(2), 898–911.

Chen, S. S., Donoho, D. L., & Saunders, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing, 20*(1), 33–61.

Efron, B., Johnstone, I., Hastie, T., & Tibshirani, R. (2004). Least angle regression. *Annals of Statistics, 32*, 407–451.

Frank, A., & Asuncion, A. (2010). UCI machine learning repository. Retrieved from archive.ics.uci.edu/ml/

Gun, S.R. (1998). *Support vector machines for classification and regression*. Southampton: ISIS Res. Group, Dept. Electron. Comput. Sci., Univ. Southampton.

Hong, X., Sharkey, P. M., & Warwick, K. (2003). Automatic nonlinear predictive model construction using forward regression and the PRESS statistic. *IEE Proceedings - Control Theory and Applications, 150*(3), 245–254.

Ji, S., Xue, Y., & Carin, L. (2008). Bayesian compressive sensing. *IEEE Transactions on Signal Processing, 56*(6), 2346–2356.

MacKay, D. J. C. (1991). *Bayesian methods for adaptive models* (PhD thesis). California Institute of Technology, CA.

Orr, M. J. L. (1995). Regularisation in the selection of radial basis function centers. *Neural Computation, 7*(3), 954–975.

Rao, R. B., Fung, G., & Rosales, R. (2008). On the dangers of cross-validation. An experimental evaluation. In C. Apte, H. Park, K. Wang, & M. J. Zaki (Eds.) *Proceedings of the SIAM Conference Data Mining* (pp. 588–596). Atlanta, GA: SIAM publishing.

Sherman, J., & Morrison, W. J. (1950). Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *Annals of Mathematical Statistics, 21*(1), 124–127.

Stone, M. (1974). Cross validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B, 36*(2), 111–147.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of Royal Statistical Society: Series B, 58*(1), 267–288.