Orthogonal Forward Regression based on Directly Maximizing Model Generalization Capability

S. Chen† and X. Hong‡

† School of Electronics and Computer Science
University of Southampton, Southampton S017 1BJ
E-mail: sqc@ecs.soton.ac.uk

‡ Department of Cybernetics
University of Reading, Reading RG6 6AY
E-mail: x.hong@reading.ac.uk


Motivation

Modeling from data: generalization, interpretability, knowledge extraction → all depend on ability to construct appropriate sparse models

- Main engine or criterion in most of subset model selection algorithms: minimizing training mean square error
- It is highly desired to be able to construct sparse models by: directly maximizing model generalization capability
- Cross validation via delete-one approach:
  - leave-one-out (LOO) test score, a measure of generalization

Delete-1 Approach with Leave-One-Out Score

- Concept of delete-1 with associated leave-one-out test score
- For linear-in-the-parameter models, no need to sequentially splitting training data set and repeatedly estimating associated models
  - Even so and even with only incrementally minimizing LOO test score, complexity becomes prohibitive for a modest model set
- Adopting orthogonal forward regression, model construction using LOO test score becomes computationally affordable
- Proposed OLS: incrementally minimizing LOO test score (generalization error) using just one training data set
  - Original OLS: incrementally minimizing training error

Regression Model

\[ y(t) = \sum_{i=1}^{n_M} \theta_i \phi_i(t) + e(t) = \phi^T(t)\theta + e(t), \quad 1 \leq t \leq N \]

- \( y(t): \) target or desired output, \( e(t): \) model error, \( \theta_i: \) model weights and \( \theta = [\theta_1 \cdots \theta_{n_M}]^T, \phi_i(t): \) regressors and \( \phi(t) = [\phi_1(t) \cdots \phi_{n_M}(t)]^T, \) \( n_M: \) number of candidate regressors, and \( N: \) number of training samples.

- Defining

\[ \mathbf{y} = [y(1) \cdots y(N)]^T, \quad \mathbf{e} = [e(1) \cdots e(N)]^T, \quad \Phi = [\phi_1 \cdots \phi_{n_M}] \]

- with \( \phi_i = [\phi_i(1) \cdots \phi_i(N)]^T, \) leads to matrix form

\[ \mathbf{y} = \Phi \theta + \mathbf{e} \]

Note that \( \phi_i(t) \) is \( i \)-th row of \( \Phi \) and \( \phi_i \) is \( i \)-th column of \( \Phi \).
Orthogonalization

Orthogonal decomposition: \( \Phi = WA \), where

\[
A = \begin{bmatrix}
1 & a_{1,2} & \cdots & a_{1,n_M} \\
0 & 1 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots \\
0 & \cdots & \cdots & 1
\end{bmatrix}
\]

and \( W = [w_1 \cdots w_{n_M}] \) with orthogonal columns: \( w_i^T w_j = 0 \), if \( i \neq j \).

Let \( g = [g_1 \cdots g_{n_M}]^T \), satisfying \( A \theta = g \). Regression model becomes

\[
y = Wg + \epsilon
\]
or

\[
y(t) = \mathbf{w}^T(t)g + \epsilon(t), \quad 1 \leq t \leq N
\]

Note that \( \mathbf{w}(t) \) is \( t \)-th row of \( \mathbf{W} \) and \( w_i \) is \( i \)-th column of \( \mathbf{W} \).

Model Construction Algorithm

- At selection step \( k \), a model term is selected if it produces the smallest LOO test score \( J_k \) among the candidate model terms \( k \) to \( n_M \).

In this algorithm,

\[
J_k = \frac{1}{N} \sum_{t=1}^{N} \frac{e_k^2(t)}{\beta_k^2(t)}
\]

This should be compared with original OLS with

\[
J_k = \frac{1}{N} \sum_{t=1}^{N} e_k^2(t)
\]

The model construction process is fully automatic, and ends with a \( n_{\theta} \)-term model when

\[
\Delta J = J_{n_{\theta}+1} - J_{n_{\theta}} \geq 0
\]

User does not need to specify any separate termination criterion.

Leave-One-Out Generalization Error

Denoting \( k \)-term model error as \( e_k(t) \), then LOO error for \( k \)-term model is

\[
e_k^{(-t)}(t) = \frac{e_k(t)}{\beta_k(t)}
\]

where super-index \((-t)\) indicates that the model is obtained with \( t \)-th training sample removed, and LOO error weighting \( \beta_k(t) \) is computed recursively

\[
\beta_k(t) = \beta_{k-1}(t) - \frac{w_k^2(t)}{\mathbf{w}_k^T \mathbf{w}_k + \lambda}
\]

where \( \lambda \) is a regularization parameter. The LOO mean square error or test score is given by:

\[
J_k = E \left[ \left( e_k^{(-t)}(t) \right)^2 \right] = \frac{1}{N} \sum_{t=1}^{N} \frac{e_k^2(t)}{\beta_k^2(t)}
\]

A Simple Scalar Function Modelling

\[
f(x) = \frac{\sin(x)}{x}, \quad -10 \leq x \leq 10
\]

Give \( y = f(x) + \epsilon \) and \( x \). 400 \( x \) uniform distribution in \([-10, 10]\) and \( \epsilon \) zero mean Gaussian with variance 0.04. First 200 samples as training set, the other 200 as testing set. Additional test set with 200 noise-free \( f(x) \).

The RBF Gaussian kernel function with variance of 10.0. Each training data was considered as a candidate RBF center and \( n_M = 200 \). Regularization parameter fixed to \( \lambda = 0.001 \).

<table>
<thead>
<tr>
<th>Model terms</th>
<th>MSE (noisy training set)</th>
<th>LOO test score</th>
<th>MSE (noisy test set)</th>
<th>MSE (noise-free test set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.8 ± 0.6</td>
<td>0.037703 ± 0.0003708</td>
<td>0.040725 ± 0.003893</td>
<td>0.041692 ± 0.002458</td>
<td>0.001749 ± 0.000630</td>
</tr>
</tbody>
</table>
• Training MSE and LOO test score in log scale for a typical set of noisy training data. Note the algorithm terminated with a 7-term model when \( J_8 = 0.041589 \geq J_7 = 0.041589 \).

• The noisy training points \( y \) and the underlying function \( f(x) \) together with the mapping generated using this 7-term model identified.

**Engine Data Modelling**

System input \( u(t) \) and output \( y(t) \)

First 210 data points for modelling, last 200 points for testing

RBF model:
\[
\hat{y}(t) = \hat{f}_{RBF}(y(t-1), u(t-1), u(t-2))
\]

Gaussian kernel function variance 1.69. Regularization parameter fixed to \( 10^{-7} \)

**Modelling Results**

• Training MSE and LOO test score in log scale for engine data set. Note the algorithm terminated with a 23-term model when \( J_{24} = 0.000548 \geq J_{23} = 0.000548 \).

• Modelling accuracy for engine data set.

<table>
<thead>
<tr>
<th>model terms</th>
<th>MSE over training set</th>
<th>LOO test score</th>
<th>MSE over test set</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>0.000449</td>
<td>0.000548</td>
<td>0.000487</td>
</tr>
</tbody>
</table>

Modelling error \( y(t) - \hat{y}(t) \) by the constructed 23-term model:
Conclusions

- A fully automatic model construction algorithm for linear-in-the-parameters nonlinear models has been developed based directly on maximizing model generalization capability.

- The leave-one-out test score in the framework of regularized orthogonal least squares has been derived and, in particular, an efficient recursive computation formula for LOO errors has been presented.

- The proposed algorithm is based on orthogonal forward regression with LOO test score to optimize model structure without resorting to another validation data set for model assessment.