Sparse Data Modelling Using Combined Locally Regularized Orthogonal Least Squares and D-Optimality Design

S. Chen†, X. Hong‡ and C.J. Harris†

† Department of Electronics and Computer Science
University of Southampton, Southampton SO17 1BJ, U.K.
E-mail: sqc@ecs.soton.ac.uk

‡ Department of Cybernetics
University of Reading, Reading RG6 6AY, U.K.

Presented at CACSCUK'2002, Beijing, China, September 20-21, 2002

Motivation

Modelling from data: *generalization*, *interpretability*, *knowledge extraction* $\Longrightarrow$ all depend on ability to construct appropriate sparse models

- ◯ Parsimonious principle: subset model selection
  - ⋆ OLS: significance of individual selected terms
- ◯ Bayesian learning: maximum $a\ posteriori$ (MAP)
  - ⋆ Bayesian framework: hyperparameters/regularization to enforce sparsity
- ◯ Optimal experimental designs: optimizing model robustness
  - ⋆ D-optimality design: maximizing determinant of design matrix
- ● **OLS with individual regularization and D-optimality design**

Regression Model

$$y(k) = \hat{y}(k) + e(k) = \sum_{i=1}^{n_M} \theta_i \phi_i(k) + e(k),\ 1 \le k \le N$$

$y(k)$: target or desired output, $e(k) = y(k) - \hat{y}(k)$, $\hat{y}(k)$: model output, θ_i: model weights, $\phi_i(k)$: regressors, n_M: number of candidate regressors, N: number of training samples.

Defining

$$\mathbf{y} = [y(1) \cdots y(N)]^T,\quad \mathbf{e} = [e(1) \cdots e(N)]^T,\quad \boldsymbol{\theta} = [\theta_1 \cdots \theta_{n_M}]^T$$

$$\boldsymbol{\Phi} = [\boldsymbol{\phi}_1 \cdots \boldsymbol{\phi}_{n_M}] \quad \text{with} \quad \boldsymbol{\phi}_i = [\phi_i(1) \cdots \phi_i(N)]^T$$

leads to matrix form

$$\mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\theta} + \mathbf{e}$$

Orthogonalization

Orthogonal decomposition: $\boldsymbol{\Phi} = \mathbf{W}\mathbf{A}$, where

$$\mathbf{A} = \begin{bmatrix} 1 & a_{1,2} & \cdots & a_{1,n_M} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_{n_M-1,n_M} \\ 0 & \cdots & 0 & 1 \end{bmatrix}$$

and $\mathbf{W} = [\mathbf{w}_1 \cdots \mathbf{w}_{n_M}]$ with orthogonal columns: $\mathbf{w}_i^T \mathbf{w}_j = 0$, if $i \ne j$.

Regression model becomes

$$\mathbf{y} = \mathbf{W}\mathbf{g} + \mathbf{e}$$

with orthogonal weight vector $\mathbf{g} = [g_1 \cdots g_{n_M}]^T$ satisfying

$$\mathbf{A}\boldsymbol{\theta} = \mathbf{g}$$

LROLS Regression with D-Optimality Design

Given regularization parameter vector $\boldsymbol{\lambda} = [\lambda_1 \cdots \lambda_{n_M}]^T$ and denoting $\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, \cdots, \lambda_{n_M}\}$, and D-optimality weighting β, combined error criterion:

$$J_C(\mathbf{g}, \boldsymbol{\lambda}, \beta) = \mathbf{e}^T\mathbf{e} + \mathbf{g}^T\boldsymbol{\Lambda}\mathbf{g} - \beta \log\det\left(\mathbf{W}^T\mathbf{W}\right)$$

$$= \mathbf{y}^T\mathbf{y} - \sum_{i=1}^{n_M} \left(\left(\mathbf{w}_i^T\mathbf{w}_i + \lambda_i\right) g_i^2 + \beta \log\left(\mathbf{w}_i^T\mathbf{w}_i\right)\right)$$

• Forward-regression procedure selects significant regressors according to combined error reduction ratio due to each regressor $\mathbf{w}_i$

$$[\text{cerr}]_i = \frac{\left(\mathbf{w}_i^T\mathbf{w}_i + \lambda_i\right) g_i^2 + \beta \log\left(\mathbf{w}_i^T\mathbf{w}_i\right)}{\mathbf{y}^T\mathbf{y}}$$

Selection terminated with n_s-term sub-model at the n_s-th stage when

$$[\text{cerr}]_l \leq 0 \quad \text{for} \quad n_s + 1 \leq l \leq n_M$$

Regularization Parameter Update

Bayesian evidence procedure for updating regularization parameters:

$$\lambda_i^{\text{new}} = \frac{\gamma_i^{\text{old}}}{N - \gamma^{\text{old}}} \frac{\mathbf{e}^T\mathbf{e}}{g_i^2}, \quad 1 \leq i \leq n_M$$

$$\gamma_i = \frac{\mathbf{w}_i^T\mathbf{w}_i}{\lambda_i + \mathbf{w}_i^T\mathbf{w}_i} \quad \text{and} \quad \gamma = \sum_{i=1}^{n_M} \gamma_i$$

Iterative Procedure

Initialization. Set all λ_i to same small positive value (e.g. 0.001). Set $\beta > 0$.

Step 1. Given current $\boldsymbol{\lambda}$, orthogonal forward procedure selects n_s-term subset model.

Step 2. Update $\boldsymbol{\lambda}$. If $\boldsymbol{\lambda}$ remains sufficiently unchanged in two successive iterations or a pre-set maximum iteration number is reached, stop; otherwise go to *Step 1*.

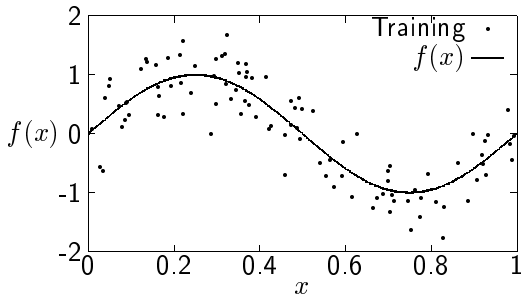
• Step 1 termination automatically, insensitive to β value

• Very sparse models with excellent generalization, without costly cross validation

A Simple Scalar Function Modelling

Modelling $f(x)$ given $y = f(x) + \epsilon$ and x. 100 x uniform distribution in $(0,\ 1)$ and ϵ zero mean Gaussian with variance 0.16.

The RBF Gaussian kernel function with variance of 0.04. Each training data was considered as a candidate RBF center and $n_M = 100$.



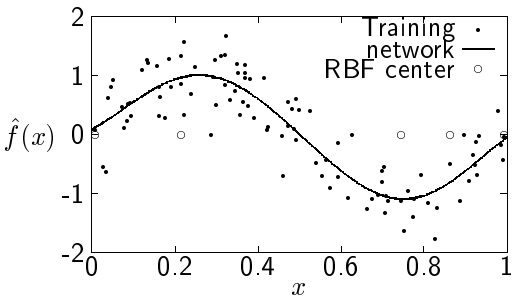
Modelling Using LROLS with D-Optimality Design

D-optimality weighting β	number of terms	variance over noise training data	variance over noise-free testing data
10^{-8}	6	0.15766	0.00168
10^{-7}	6	0.15766	0.00168
10^{-6}	6	0.15823	0.00202
10^{-5}	5	0.15705	0.00194
10^{-4}	5	0.15826	0.00246
10^{-3}	5	0.15705	0.00194
10^{-2}	5	0.15705	0.00194
10^{-1}	5	0.15911	0.00223

• Insensitive to D-optimality cost weighting β

• Sparser model with equally good generalization performance, compared with using LROLS alone (6 terms)

Simple Scalar Function Modelling Result



5-term model mapping (curve) produced by the combined LROLS and D-optimality algorithm with $\beta = 10^{-5}$ for simple scalar function modelling problem. Dots indicate noisy training data y and circles the RBF centers.

Modelling Using LROLS with D-Optimality Design

RBF model $\hat{y}_k = f_{RBF}(y_{k-1}, y_{k-2})$ with Gaussian kernel function of variance 0.81.

D-optimality weighting β	number of terms	variance over training data	variance over testing data
10^{-6}	19	0.09275	0.09635
10^{-4}	13	0.09311	0.09607
10^{-2}	13	0.09338	0.09750
10^{0}	13	0.09395	0.09667

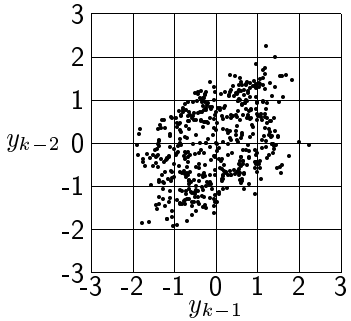
- Insensitive to D-optimality cost weighting β

- Sparser model with equally good generalization performance, compared with using LROLS alone (18 terms, 0.09264, 0.09678)

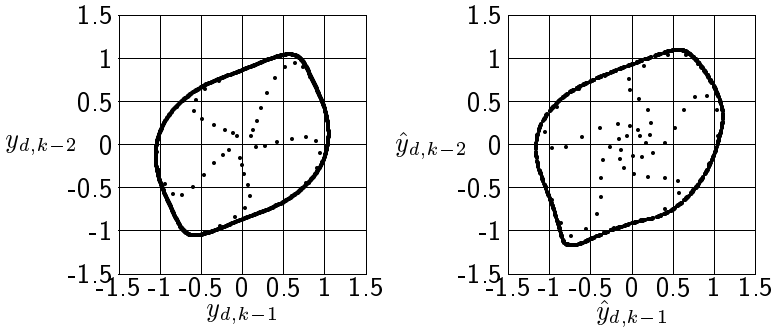
Simulated Nonlinear Time Series Modelling

$$y_k = \left(0.8 - 0.5\exp(-y_{k-1}^2)\right) y_{k-1} - \left(0.3 + 0.9\exp(-y_{k-1}^2)\right) y_{k-2} + 0.1\sin(\pi y_{k-1}) + \epsilon_k$$

Noise ϵ_k Gaussian with zero mean and variance 0.09. 1000 samples, first 500 for training (figure below), last 500 for testing.



Nonlinear Time Series Modelling Result



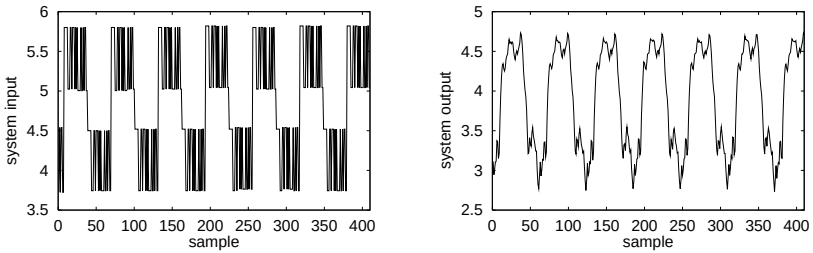
Comparison of underlying noise-free system $y_{d,k}$ and iterative RBF model output

$$\hat{y}_{d,k} = f_{RBF}(\hat{y}_{d,k-1}, \hat{y}_{d,k-2})$$

13-term model produced by combined LROLS and D-optimality algorithm with $\beta = 10^{-4}$

Engine Data Modelling

System input u_k and output y_k



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

RBF one-step prediction: $\hat{y}_k = f_{RBF}(y_{k-1}, u_{k-1}, u_{k-2})$, Gaussian kernel function variance 1.69

RBF iterative model output: $\hat{y}_{d,k} = f_{RBF}(\hat{y}_{d,k-1}, u_{k-1}, u_{k-2})$

Modelling Using LROLS with D-Optimality Design

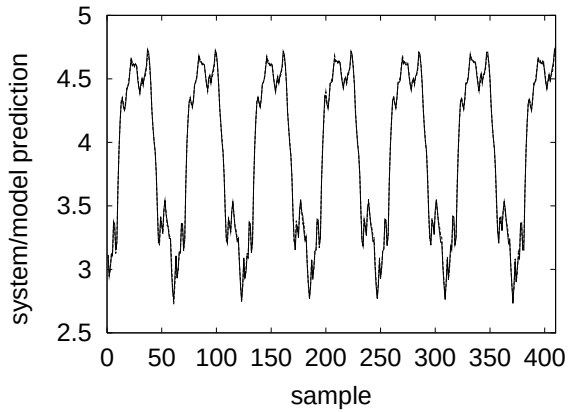
D-optimality weighting β	number of terms	variance over training data	variance over testing data
10^{-8}	22	0.000459	0.000488
10^{-7}	27	0.000442	0.000484
10^{-6}	25	0.000441	0.000479
10^{-5}	22	0.000452	0.000499
10^{-4}	20	0.000586	0.000606
10^{-3}	20	0.000478	0.000501
10^{-2}	16	0.000884	0.000982
10^{-1}	12	0.004951	0.005050

- Insensitive to a wide range values of D-optimality cost weighting β

- Sparser model with equally good generalization performance, compared with using LROLS alone (34 terms, 0.000435, 0.000487)

Engine Data Modelling Result

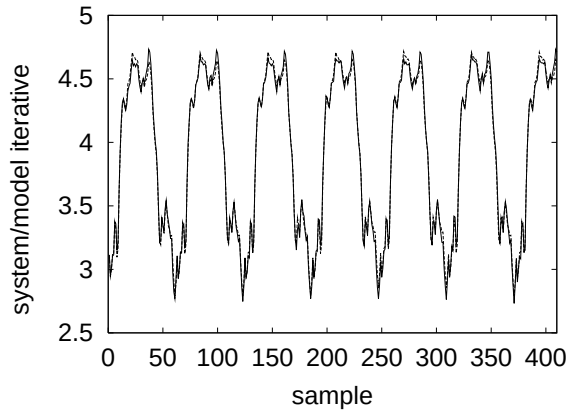
y_k: solid $\hat{y}_k$: dashed



Comparison of system output y_k and model one-step prediction $\hat{y}_k$. 22-term model produced by combined LROLS and D-optimality algorithm with $\beta = 10^{-5}$

Engine Data Modelling Result

y_k: solid $\hat{y}_{d,k}$: dashed



Comparison of system output y_k and model iterative output $\hat{y}_{d,k}$. 22-term model produced by combined LROLS and D-optimality algorithm with $\beta = 10^{-5}$

Conclusions

Combining locally regularized orthogonal least squares with D-optimality experimental design — a state of art model construction algorithm

- Efficiency ensured as usual by orthogonal forward regression
- Coupling effects of local regularization and D-optimality design further enhance each other, and combined algorithm is capable of producing small-size models that generalize well
- User only needs to specify D-optimality cost weighting β, and model construction is automatic, without need of costly cross validation

  Value of β does not critically influence performance, and it can be chosen with ease from a large range of values