Sparse Kernel Density Estimation Technique Based on Zero-Norm Constraint

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

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

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

International Joint Conference on Neural Networks 2010

Outline

1. Motivations
   - Existing Regularisation Approaches
   - Our Contributions
2. Proposed Sparse Kernel Density Estimator
   - Problem Formulation
   - Approximate Zero-Norm Regularisation
   - *D*-Optimality Based Subset Selection
3. Numerical Examples
   - Experimental Set Up
   - Experimental Results
4. Conclusions

Outline

Regularisation Methods

- **Two-norm** of weight vector
  - Naturally combined with quadratic main cost function, and computationally efficient implementation
  - Only drive many weights to small near-zero values
- **One-norm** of weight vector
  - Can drive many weights to zero, and hence should achieve sparser results than two-norm based method
  - Harder to minimise and higher complexity implementation
- **Zero-norm** of weight vector
  - Ultimate model sparsity and generalisation performance
  - Intractable in implementation, and even with approximation, very difficult to minimise and impose very high complexity

Two-norm and one-norm based regularisations have been combined with OLS algorithm, with the former approach providing highly efficient sparse kernel modelling

Outline

Our Contributions

- We incorporate an effective **approximate zero-norm** regularisation into **sparse kernel density** estimation
  - Approximate zero-norm naturally merges into underlying **constrained nonnegative quadratic programming**
  - Various SVM algorithms can readily be applied to obtain SKD estimate efficiently
- Proposed sparse kernel density estimator:
  - use *D*-optimality OLS subset selection to select a small number of significant kernels, in terms of kernel eigenvalues
  - then solve final SKD estimate from associate subset constrained nonnegative quadratic programming

Outline

1. Motivations
   - Existing Regularisation Approaches
   - Our Contributions
2. Proposed Sparse Kernel Density Estimator
   - Problem Formulation
   - Approximate Zero-Norm Regularisation
   - *D*-Optimality Based Subset Selection
3. Numerical Examples
   - Experimental Set Up
   - Experimental Results
4. Conclusions

Kernel Density Estimation

- Give finite data set $D_N = \{\mathbf{x}_k\}_{k=1}^N$, drawn from unknown **density** $p(\mathbf{x})$, where $\mathbf{x}_k \in \mathcal{R}^m$
- Infer $p(\mathbf{x})$ based on D_N using **kernel density estimate**

$$\hat{p}(\mathbf{x}; \boldsymbol{\beta}_N, \rho) \quad = \quad \sum_{k=1}^{N} \beta_k K_\rho(\mathbf{x}, \mathbf{x}_k)$$

$$\text{s.t.} \quad \beta_k \geq 0,\ 1 \leq k \leq N,\ \boldsymbol{\beta}_N^T \mathbf{1}_N = 1$$

- Here $\boldsymbol{\beta}_N = [\beta_1\ \beta_2 \cdots \beta_N]^T$: kernel weight vector, $\mathbf{1}_N$: the vector of ones with dimension N, and $K_\rho(\bullet, \bullet)$: chosen kernel function with **kernel width** ρ
- **Unsupervised** density estimation $\Rightarrow$ "**supervised**" regression
    - using **Parzen window** estimate as "desired response"

Regression Formulation

- For $\mathbf{x}_k \in D_N$, denote $\hat{y}_k = \hat{p}(\mathbf{x}_k; \boldsymbol{\beta}_N, \rho)$, y_k as Parzen window estimate at $\mathbf{x}_k$, and $\varepsilon_k = y_k - \hat{y}_k \Rightarrow$ **regression** formulation

$$y_k = \hat{y}_k + \varepsilon_k = \phi_N^T(k)\boldsymbol{\beta}_N + \varepsilon_k$$

  or over D_N

$$\mathbf{y} = \boldsymbol{\Phi}_N \boldsymbol{\beta}_N + \boldsymbol{\varepsilon}$$

- Associated **constrained nonnegative quadratic programming**

$$\min_{\boldsymbol{\beta}_N} \left\{ \frac{1}{2} \boldsymbol{\beta}_N^T \mathbf{B}_N \boldsymbol{\beta}_N - \mathbf{v}_N^T \boldsymbol{\beta}_N \right\}$$
$$\text{s.t. } \boldsymbol{\beta}_N^T \mathbf{1}_N = 1 \text{ and } \beta_i \geq 0, 1 \leq i \leq N$$

  where $\mathbf{B}_N = \boldsymbol{\Phi}_N^T \boldsymbol{\Phi}_N$ is the design matrix and $\mathbf{v}_N = \boldsymbol{\Phi}_N^T \mathbf{y}$

- This is **not** using kernel density estimate to fit Parzen window estimate !

Outline

Zero-Norm Constraint

- Given $\alpha > 0$, an approximation to **zero norm** $\|\boldsymbol{\beta}_N\|_0$ is

$$\|\boldsymbol{\beta}_N\|_0 \approx \sum_{i=1}^{N} \left(1 - e^{-\alpha|\beta_i|}\right)$$

- Combining this zero-norm constraint with constrained NNQP

$$\min_{\boldsymbol{\beta}_N} \left\{ \frac{1}{2}\boldsymbol{\beta}_N^T \mathbf{B}_N \boldsymbol{\beta}_N - \mathbf{v}_N^T \boldsymbol{\beta}_N + \lambda \sum_{i=1}^{N} \left(1 - e^{-\alpha|\beta_i|}\right) \right\}$$
$$\text{s.t. } \boldsymbol{\beta}_N^T \mathbf{1}_N = 1 \text{ and } \beta_i \geq 0, 1 \leq i \leq N$$

  with $\lambda > 0$ a small "regularisation" parameter

- With 2nd order **Taylor series expansion** for $e^{-\alpha|\beta_i|}$

$$e^{-\alpha|\beta_i|} \approx 1 - \alpha|\beta_i| + \frac{\alpha^2\beta_i^2}{2} \quad \Rightarrow$$

$$\sum_{i=1}^{N} \left(1 - e^{-\alpha|\beta_i|}\right) \approx \alpha \sum_{i=1}^{N} |\beta_i| - \frac{\alpha^2}{2} \sum_{i=1}^{N} \beta_i^2$$

Constrained NNQP

- Hence, "new" constrained NNQP

$$\min_{\boldsymbol{\beta}_N} \left\{ \frac{1}{2} \boldsymbol{\beta}_N^T \mathbf{A}_N \boldsymbol{\beta}_N - \mathbf{v}_N^T \boldsymbol{\beta}_N \right\}$$
$$\text{s.t. } \boldsymbol{\beta}_N^T \mathbf{1}_N = 1 \text{ and } \beta_i \geq 0, 1 \leq i \leq N$$

$\mathbf{A}_N = \mathbf{B}_N - \delta \mathbf{I}_N$ and $\delta = \lambda \alpha^2$ predetermined small parameter

- **Remark**: Under convexity constraint on $\boldsymbol{\beta}_N$, **minimisation** of approximate **zero norm** $\Leftrightarrow$ **maximisation** of **two norm** $\boldsymbol{\beta}_N^T \mathbf{I}_N \boldsymbol{\beta}_N$
- Design matrix $\mathbf{B}_N$ should **positive definite**, and δ bounded by smallest **eigenvalue** of $\mathbf{B}_N$ so that $\mathbf{A}_N$ also positive definite
  - Common for $\mathbf{B}_N$ of large data set to be ill-conditioned
  - Approach most **effective** when it is applied following some model **subset selection** preprocessing

Outline

D-Optimality Design

- **Least squares** estimate $\hat{\beta}_N = \mathbf{B}_N^{-1}\mathbf{\Phi}_N^T\mathbf{y}$ is unbiased and covariance matrix of estimate $\mathrm{Cov}\left[\hat{\beta}_N\right] \propto \mathbf{B}_N^{-1}$
  - Estimation accurate depends on **condition number**

$$C = \frac{\max\{\sigma_i, 1 \le i \le N\}}{\min\{\sigma_i, 1 \le i \le N\}}$$

  where σ_i, $1 \le i \le N$, are eigenvalues of $\mathbf{B}_N$

- D-optimality design maximises **determinant** of design matrix
  - Selected subset model $\mathbf{\Phi}_{N_s}$ maximises

$$\det\left(\mathbf{\Phi}_{N_s}^T\mathbf{\Phi}_{N_s}\right) = \det\left(\mathbf{B}_{N_s}\right)$$

  - Prevent oversized ill-posed model and high estimate variances

- "**Unsupervised**" D-optimality design particularly suitable for determining structure of kernel density estimate

OFR Aided Algorithm

- **Orthogonal forward regression** selects $\mathbf{\Phi}_{N_s}$ of N_s significant kernels based on D-optimality criterion
  - Complexity of this **preprocessing** no more than $\mathcal{O}(N^2)$
- This preprocessing results in subset constrained NNQP

$$\min_{\boldsymbol{\beta}_{N_s}} \left\{ \frac{1}{2} \boldsymbol{\beta}_{N_s}^T \mathbf{A}_{N_s} \boldsymbol{\beta}_{N_s} - \mathbf{v}_{N_s}^T \boldsymbol{\beta}_{N_s} \right\}$$
$$\text{s.t. } \boldsymbol{\beta}_{N_s}^T \mathbf{1}_{N_s} = 1 \text{ and } \beta_i \geq 0, 1 \leq i \leq N_s$$

with $\mathbf{v}_{N_s} = \mathbf{\Phi}_{N_s}^T \mathbf{y}$, $\mathbf{A}_{N_s} = \mathbf{B}_{N_s} - \delta \mathbf{I}_{N_s}$, $\mathbf{B}_{N_s} = \mathbf{\Phi}_{N_s}^T \mathbf{\Phi}_{N_s}$, $\delta < \mathbf{w}_{N_s}^T \mathbf{w}_{N_s}$

  - Various **SVM** algorithms can be used to solve this problem
- As N_s is very small and $\mathbf{A}_{N_s}$ is well-conditioned, we use simple **multiplicative** nonnegative quadratic programming algorithm
  - Complexity of which is negligible, in comparison with $\mathcal{O}(N^2)$ of D-optimality based OFR preprocessing

Outline

Experimental Setup

- **Training** set had N randomly drawn samples, while **test** set of $N_{\text{test}} = 10,000$ samples for calculating L_1 test error

$$L_1 = \frac{1}{N_{\text{test}}} \sum_{k=1}^{N_{\text{test}}} \left| p(\mathbf{x}_k) - \hat{p}(\mathbf{x}_k; \boldsymbol{\beta}_N, \rho) \right|$$

  between true density $p(\mathbf{x})$ and estimate $\hat{p}(\mathbf{x}_k; \boldsymbol{\beta}_N, \rho)$

- Numerical approximation of Kullback-Leibler **divergence** (KLD)

$$D_{\text{KL}}(p|\hat{p}) = \int_{\mathcal{R}^m} p(\mathbf{x}) \log \frac{p(\mathbf{x})}{\hat{p}(\mathbf{x}; \boldsymbol{\beta}_N, \rho)} \, d\mathbf{x}$$

  also used for testing in 2-D case

- Proposed SKD estimator compared with **PW** estimator, our **previous** SKD estimator and reduced set density estimator (**RSDE**), as well as Gaussian mixture model (**GMM**) estimator

Outline

First 2-D Example

- True density: **mixture** of Gaussian and Laplacian distributions

$$p(x_1, x_2) = \frac{1}{4\pi} e^{-\frac{(x_1-2)^2}{2}} e^{-\frac{(x_2-2)^2}{2}} + \frac{0.35}{8} e^{-0.7|x_1+2|} e^{-0.5|x_2+2|}$$

  $N = 500$, and experiment repeated $N_{\text{run}} = 100$ times

- Performance comparison, $N = 500$ and average over 100 runs

estimator	PW	previous SKD	RSDE	GMM	**proposed SKD**
kernel	$\rho^{\text{Par}} = 0.42$	$\rho = 1.1$	$\rho = 1.2$	tunable	$\rho = 1.1$
$L_1 \times 10^3$	4.04 ± 0.69	3.84 ± 0.78	4.05 ± 0.45	3.47 ± 0.99	3.56 ± 0.69
KLC $\times 10$	1.47 ± 0.23	1.40 ± 0.53	0.90 ± 0.41	0.61 ± 0.17	1.30 ± 0.31
kernel no.	500	15.3 ± 3.9	16.2 ± 3.4	11	11.0 ± 1.5
maximum	500	25	24	11	14
minimum	500	8	9	11	8

- **Similar** test performance to existing kernel density estimators, but **sparser** estimate

Second 2-D Example

- True density: **mixture** of five Gaussian distributions

$$p(x,y) = \sum_{i=1}^{5} \frac{1}{10\pi} e^{-\frac{(x-\mu_{i,1})^2}{2}} e^{-\frac{(y-\mu_{i,2})^2}{2}}$$

  Five means of Gaussian distributions: $[0.0 \ -4.0]$, $[0.0 \ -2.0]$, $[0.0 \ 0.0]$, $[-2.0 \ 0.0]$, and $[-4.0 \ 0.0]$

- Performance comparison, $N = 500$ and average over 100 runs

estimator	PW	previous SKD	RSDE	GMM	**proposed SKD**
kernel	$\rho^{\mathrm{Par}} = 0.5$	$\rho = 1.1$	$\rho = 1.2$	tunable	$\rho = 1.0$
$L_1 \times 10^3$	3.62 ± 0.44	3.61 ± 0.50	3.63 ± 0.36	3.68 ± 0.67	3.32 ± 0.63
KLC $\times 10^2$	3.42 ± 0.55	3.67 ± 0.92	3.54 ± 0.49	3.39 ± 0.87	2.90 ± 1.09
kernel no.	500	13.2 ± 2.9	13.2 ± 3.0	8	7.8 ± 1.3
maximum	500	22	21	8	11
minimum	500	8	6	8	5

- **Similar** test performance to existing kernel density estimators, but **sparser** estimate

6-D Example

- True density: **mixture** of three Gaussian distributions

$$p(\mathbf{x}) = \frac{1}{3}\sum_{i=1}^{3} \frac{1}{(2\pi)^{6/2}} \frac{1}{\det^{1/2}|\boldsymbol{\Gamma}_i|} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Gamma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)}$$

with

$$\boldsymbol{\mu}_1 = [1.0\ 1.0\ 1.0\ 1.0\ 1.0\ 1.0]^T$$
$$\boldsymbol{\Gamma}_1 = \mathrm{diag}\{1.0, 2.0, 1.0, 2.0, 1.0, 2.0\}$$

$$\boldsymbol{\mu}_2 = [-1.0\ \ -1.0\ \ -1.0\ \ -1.0\ \ -1.0\ \ -1.0]^T$$
$$\boldsymbol{\Gamma}_2 = \mathrm{diag}\{2.0, 1.0, 2.0, 1.0, 2.0, 1.0\}$$

$$\boldsymbol{\mu}_3 = [0.0\ 0.0\ 0.0\ 0.0\ 0.0\ 0.0]^T$$
$$\boldsymbol{\Gamma}_3 = \mathrm{diag}\{2.0, 1.0, 2.0, 1.0, 2.0, 1.0\}$$

- Estimation set had $N = 600$ samples, and experiment was repeated $N_{\text{run}} = 100$ times

6-D Example Results

- Performance comparison, $N = 600$ and average over 100 runs

estimator	PW	previous SKD	RSDE	GMM	**proposed SKD**
kernel	$\rho^{\text{Par}} = 0.65$	$\rho = 1.2$	$\rho = 1.2$	tunable	$\rho = 1.2$
$L_1 \times 10^5$	3.52 ± 0.16	3.11 ± 0.53	2.74 ± 0.50	1.74 ± 0.29	2.77 ± 0.24
kernel no.	600	9.4 ± 1.9	14.2 ± 3.6	8	7.9 ± 1.3
maximum	600	16	25	8	12
minimum	600	7	8	8	5

- **Similar** test performance to existing kernel density estimators, but **sparser** estimate

Conclusions

- We have integrated zero-norm regularisation naturally into construction of sparse kernel density estimator
    - Classical Parzen window estimate as “desired response”
    - Convexity constraint with zero-norm approximation turns problem into tractable nonnegative quadratic programming
    - *D*-optimality preprocessing selects small significant kernel subset to ensure well-conditioned solution
    - Complexity compares favourably with existing sparse kernel density estimators
- Zero-norm regularisation and *D*-optimality aided estimator offers an efficient means
    - for selecting very sparse kernel density estimates with excellent generalisation performance