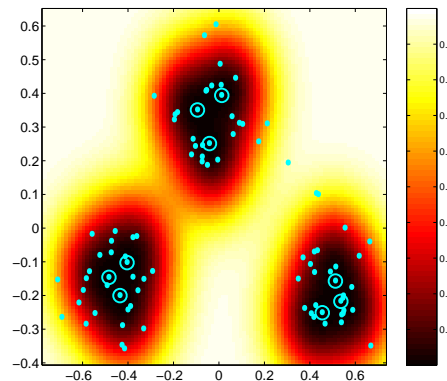


Bayesian Inference: Principles and Practice

4. Sparse Bayesian Models: Analysis, Optimisation and Applications

Mike Tipping



Microsoft
Research Cambridge, UK

Lecture 4: Overview

- Further analysis of the sparse Bayesian marginal likelihood function
- Based on this, an improved optimisation algorithm
- Extensions and applications of (sparse) Bayesian models

The Marginal Likelihood Function

- We integrated out weights \mathbf{w} to obtain *marginal likelihood*:

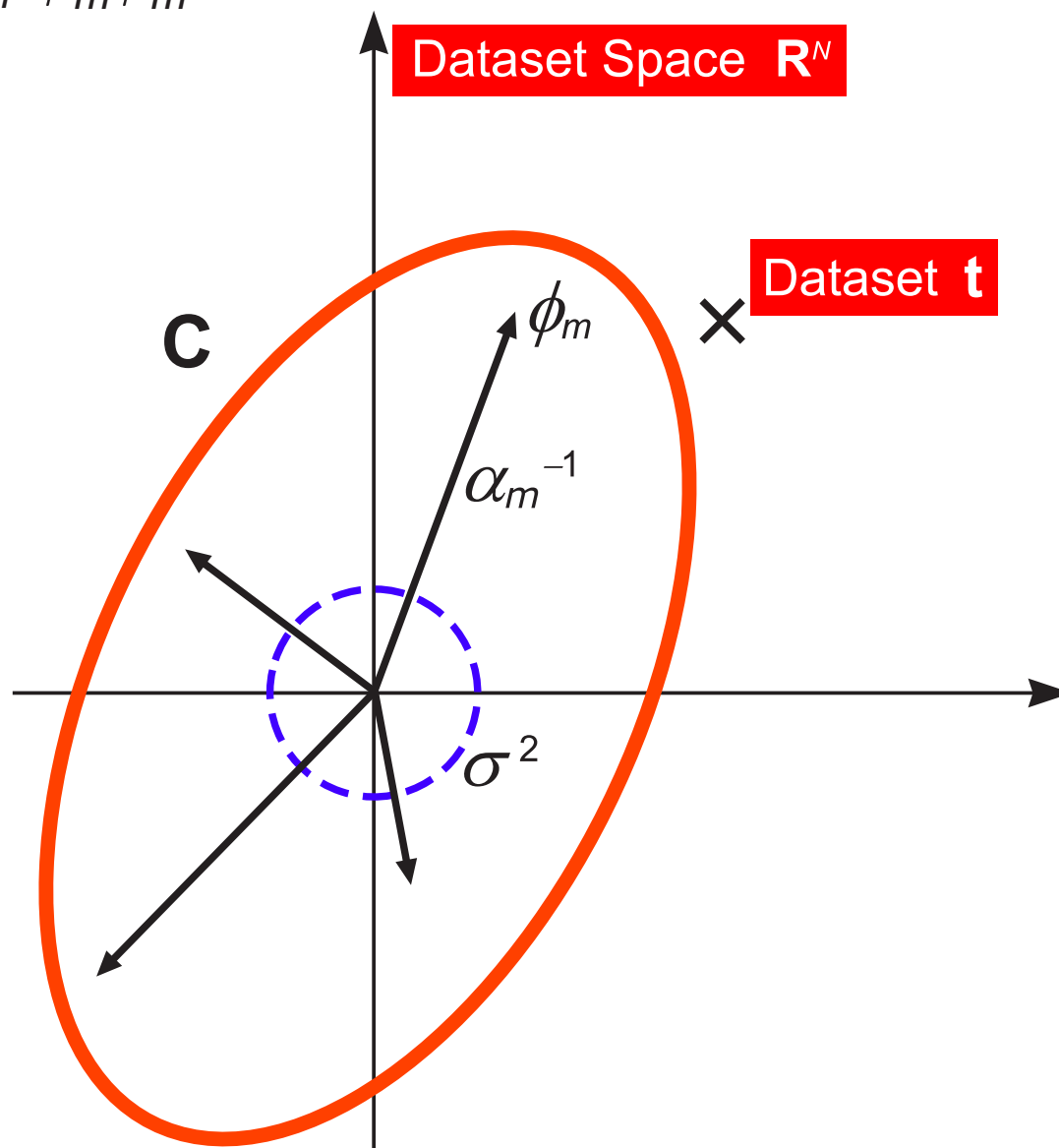
$$\begin{aligned} p(\mathbf{t}|\alpha, \sigma^2) &= \int p(\mathbf{t}|\mathbf{w}, \sigma^2) p(\mathbf{w}|\alpha) d\mathbf{w}, \\ &= (2\pi)^{-N/2} |\mathbf{C}|^{-1/2} \exp \left\{ -\frac{1}{2} \mathbf{t}^\top \mathbf{C}^{-1} \mathbf{t} \right\} \end{aligned}$$

with $\mathbf{C} = \sigma^2 \mathbf{I} + \sum_m \alpha_m^{-1} \phi_m \phi_m^\top$

- Further integration over α intractable
- We maximise $p(\alpha, \sigma^2|\mathbf{t})$ to find α_{MP} and σ_{MP}^2
- For uniform hyperpriors, equivalent to maximising $p(\mathbf{t}|\alpha, \sigma^2)$

That Picture Again...

$$\mathbf{C} = \sigma^2 \mathbf{I} + \sum_m \alpha_m^{-1} \phi_m \phi_m^\top$$



Dependence on a Single Hyperparameter (1)

- Our objective is to maximise:

$$\log p(\mathbf{t}|\alpha, \sigma^2) = -\frac{1}{2} \left[\log |\mathbf{C}| + \mathbf{t}^\top \mathbf{C}^{-1} \mathbf{t} \right] + \text{constant terms}$$

- Decompose:

$$\begin{aligned} \mathbf{C} &= \sigma^2 \mathbf{I} + \sum_{m \neq i} \alpha_m^{-1} \phi_m \phi_m^\top + \alpha_i^{-1} \phi_i \phi_i^\top \\ &= \mathbf{C}_{-i} + \alpha_i^{-1} \phi_i \phi_i^\top \end{aligned}$$

- Now we exploit some established matrix identities:

$$\begin{aligned} |\mathbf{C}| &= |\mathbf{C}_{-i}| |1 + \alpha_i^{-1} \phi_i^\top \mathbf{C}_{-i}^{-1} \phi_i| \\ \mathbf{C}^{-1} &= \mathbf{C}_{-i}^{-1} - \frac{\mathbf{C}_{-i}^{-1} \phi_i \phi_i^\top \mathbf{C}_{-i}^{-1}}{\alpha_i + \phi_i^\top \mathbf{C}_{-i}^{-1} \phi_i} \end{aligned}$$

Dependence on a Single Hyperparameter (2)

- $\log p(\mathbf{t}|\alpha, \sigma^2)$ can then be written in the form:

$$\log p(\mathbf{t}|\alpha_{-i}, \sigma^2) + \frac{1}{2} \left[\log \alpha_i - \log (\alpha_i + s_i) + \frac{q_i^2}{\alpha_i + s_i} \right]$$

where $\log p(\mathbf{t}|\alpha_{-i}, \sigma^2)$ is independent of α_i

- For convenience, “quality” and “sparsity” terms have been defined:

$$q_i = \phi_i^\top \mathbf{C}_{-i}^{-1} \mathbf{t}$$

$$s_i = \phi_i^\top \mathbf{C}_{-i}^{-1} \phi_i$$

- Note these terms are independent of α_i (but depend on all other α_{-i})

Maxima of the Marginal Likelihood

- Dependence of marginal likelihood on single hyperparameter α_j is captured by:

$$\ell(\alpha_j) = \log \alpha_j - \log (\alpha_j + s_j) + \frac{q_j^2}{\alpha_j + s_j}$$

- Setting $\partial \ell(\alpha_j) / \partial \alpha_j = 0$ gives analytic solutions:

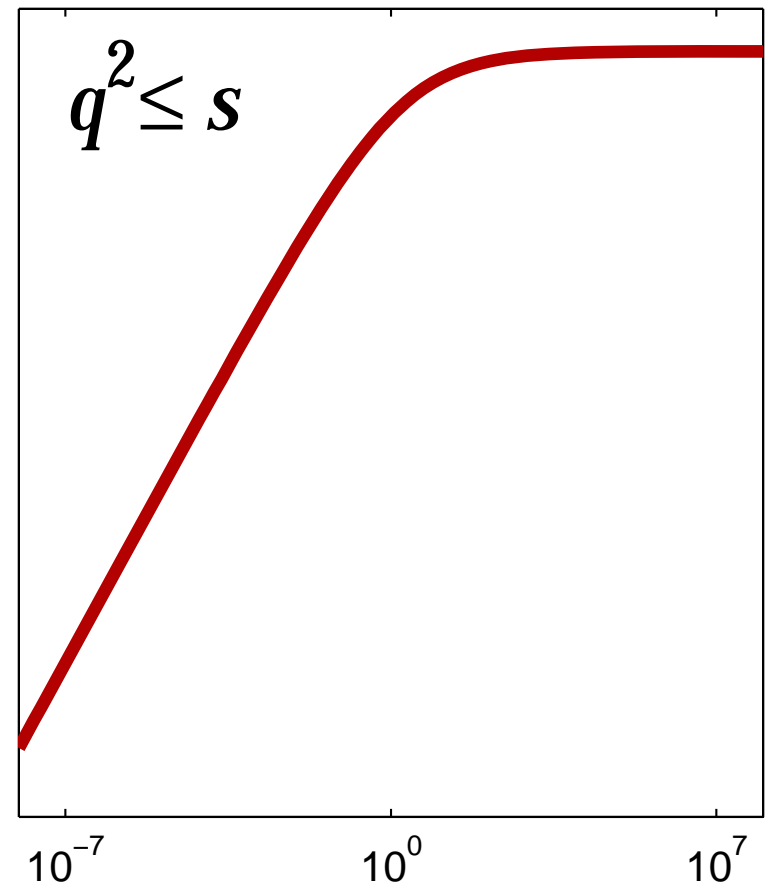
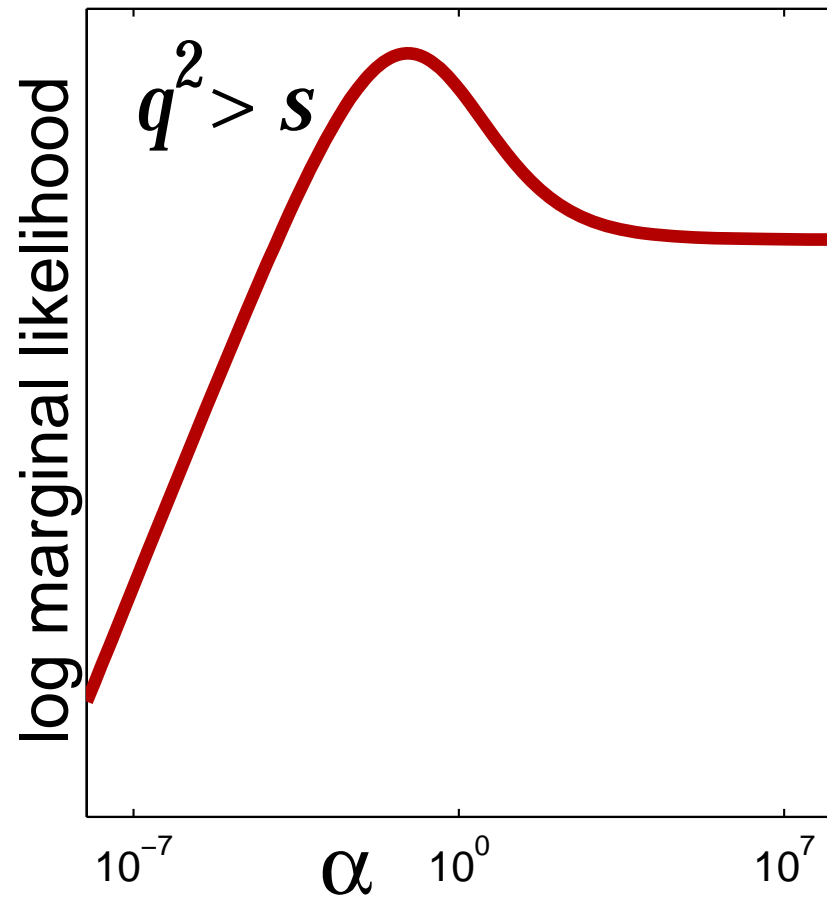
- If $q_j^2 > s_j$:

$$\alpha_j^{\text{opt}} = \frac{s_j^2}{q_j^2 - s_j}$$

- If $q_j^2 \leq s_j$:

$$\alpha_j^{\text{opt}} = \infty$$

Maxima Visualised



Optimisation Operations

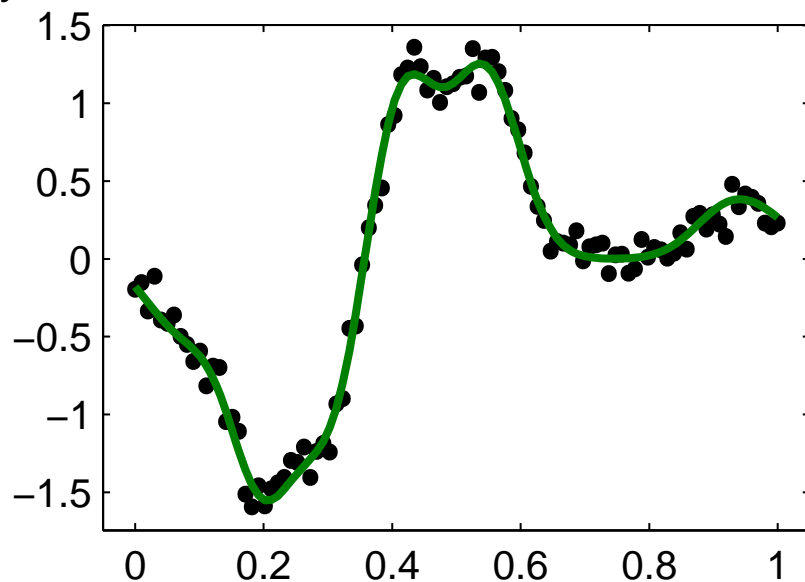
- For any given basis function $\phi_j(\mathbf{x})$ and associated hyperparameter α_j we can compute the quantities s_j and q_j^2 (true even if $\alpha_j = \infty$)
- Depending on the criterion $q_j^2 > s_j$ and the value of α_j we can then perform the following updates, all of which will increase $p(\mathbf{t}|\alpha, \sigma^2)$:

	“In model”: $\alpha_j < \infty$	“Out of model”: $\alpha_j = \infty$
$q_j^2 > s_j$	<i>re-estimation of α_j</i>	<i>addition of $\phi_j(\mathbf{x})$</i>
$q_j^2 \leq s_j$	<i>deletion of $\phi_j(\mathbf{x})$</i>	—

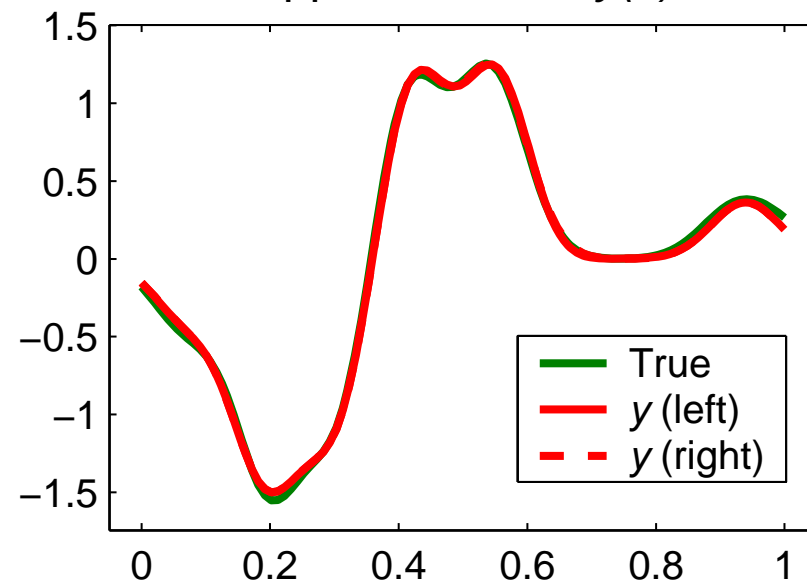
Optimisation Algorithm Sketch

- ① Initialise σ^2 sensibly and all $\alpha_m = \infty$ (i.e. the ‘empty’ model)
- ② Select a function $\phi_i(\mathbf{x})$ from the set of all M
- ③ Compute “relevance” $\mathcal{R}_i \triangleq q_i^2 - s_i$
 - ┃ If $\mathcal{R}_i > 0$ and $\alpha_i < \infty$: **re-estimate** α_i
 - ┃ If $\mathcal{R}_i > 0$ and $\alpha_i = \infty$: **add** ϕ_i to the model with updated α_i
 - ┃ If $\mathcal{R}_i \leq 0$ and $\alpha_i < \infty$: **delete** ϕ_i from the model and set $\alpha_i = \infty$
- ④ If estimating the noise level, update σ^2
- ⑤ Recalculate all q_m and s_m
- ⑥ If converged terminate, otherwise goto ②

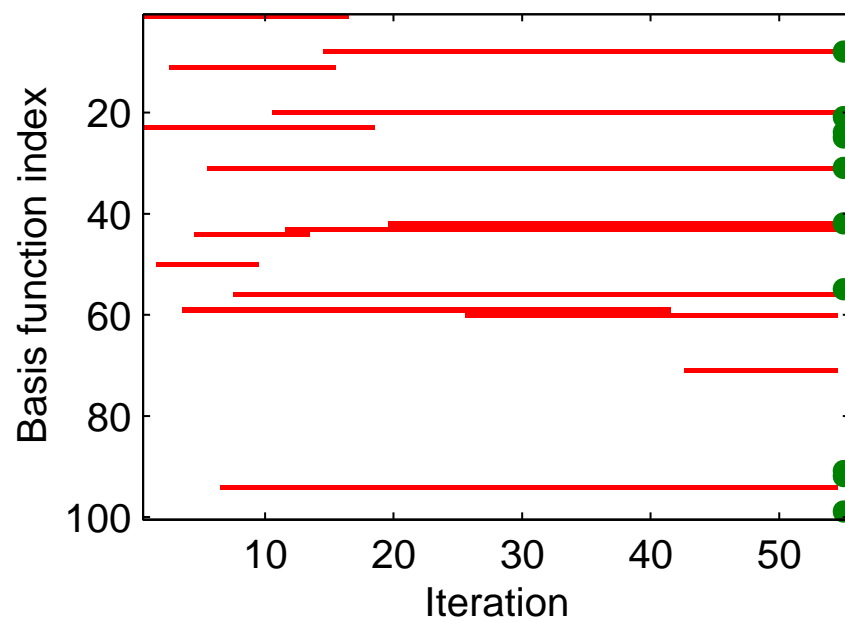
Synthetic data from size 10 basis, noise 0.100



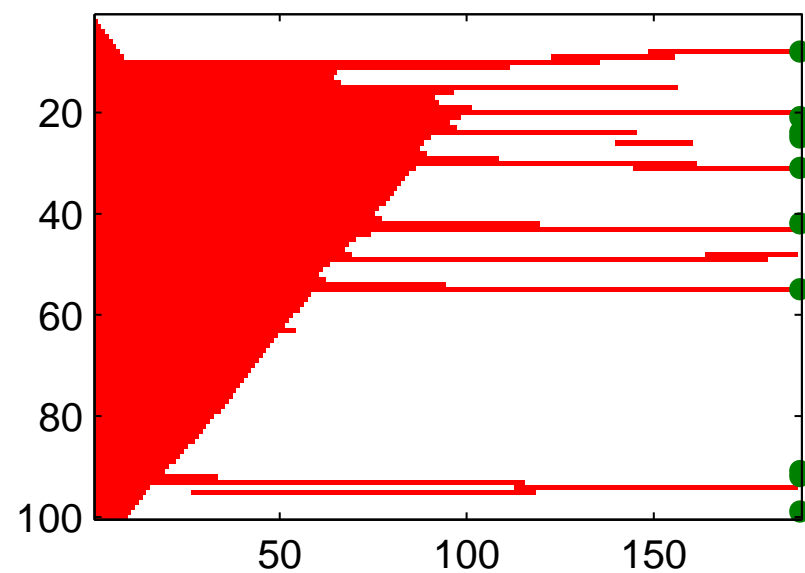
Approximations $y(x)$



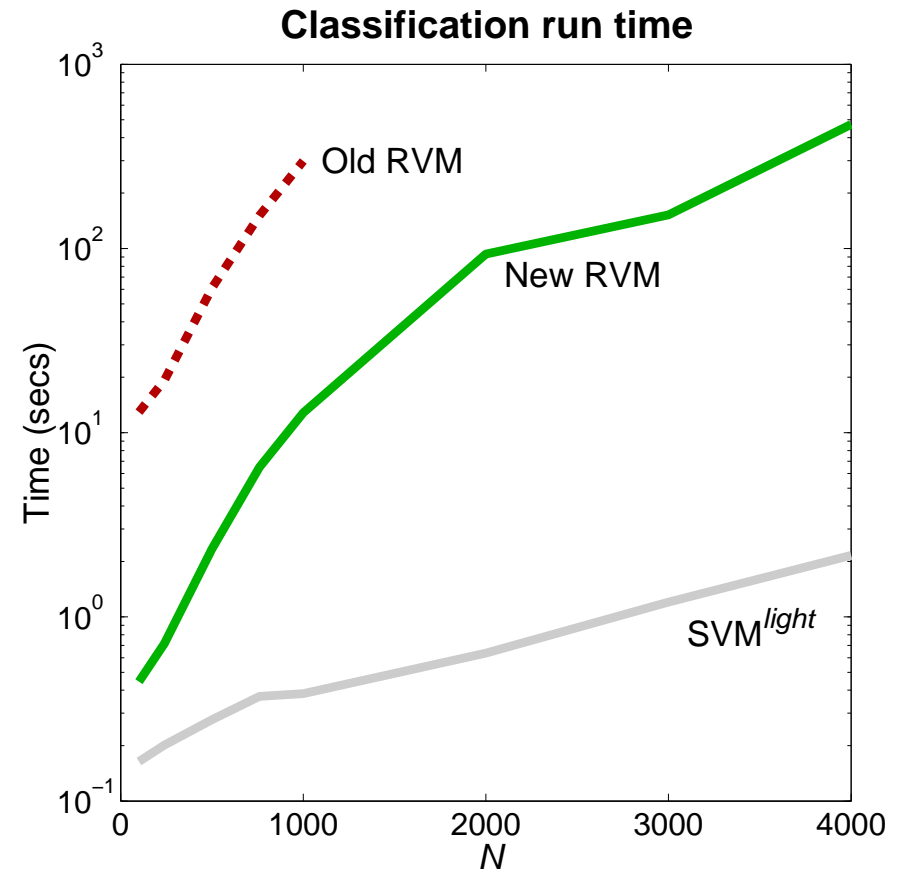
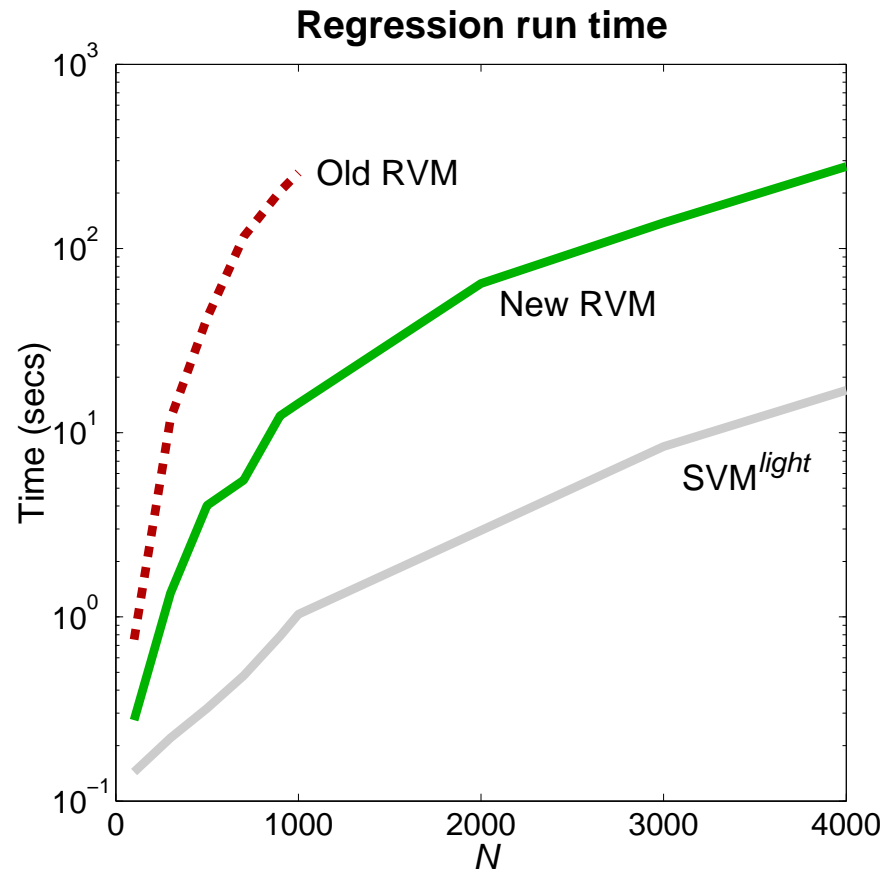
Basis=9, Error 0.031, Noise 0.088



Basis=7, Error 0.030, Noise 0.088



Performance Illustration: run time



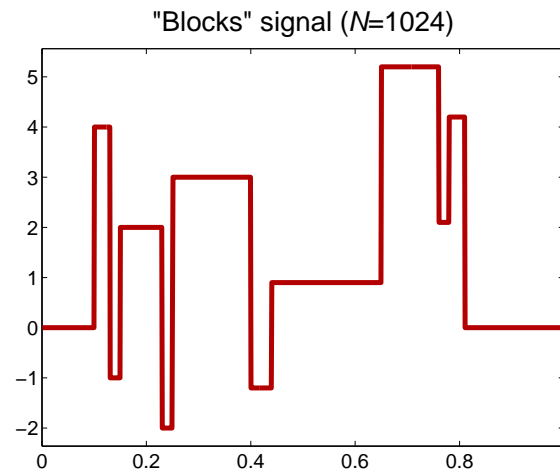
Performance Illustration: example timing

- Comparing at $N = 1000$ we have:

	Regression	Classification
Old RVM	4 mins 17 secs	4 mins 58 secs
New RVM	14.42 secs	12.84 secs
SVM ^{light}	1.03 secs	0.38 secs

Greediness?

- Agglomerative algorithms (e.g. “matching pursuit”) are often *greedy* — *i.e.* “early” additions can be significantly sub-optimal
- Demonstration: a popular signal processing test data set



- Approximate with a basis comprising:
 - “heaviside” step functions (easy)
 - “heaviside” *and* Gaussians (hard?)

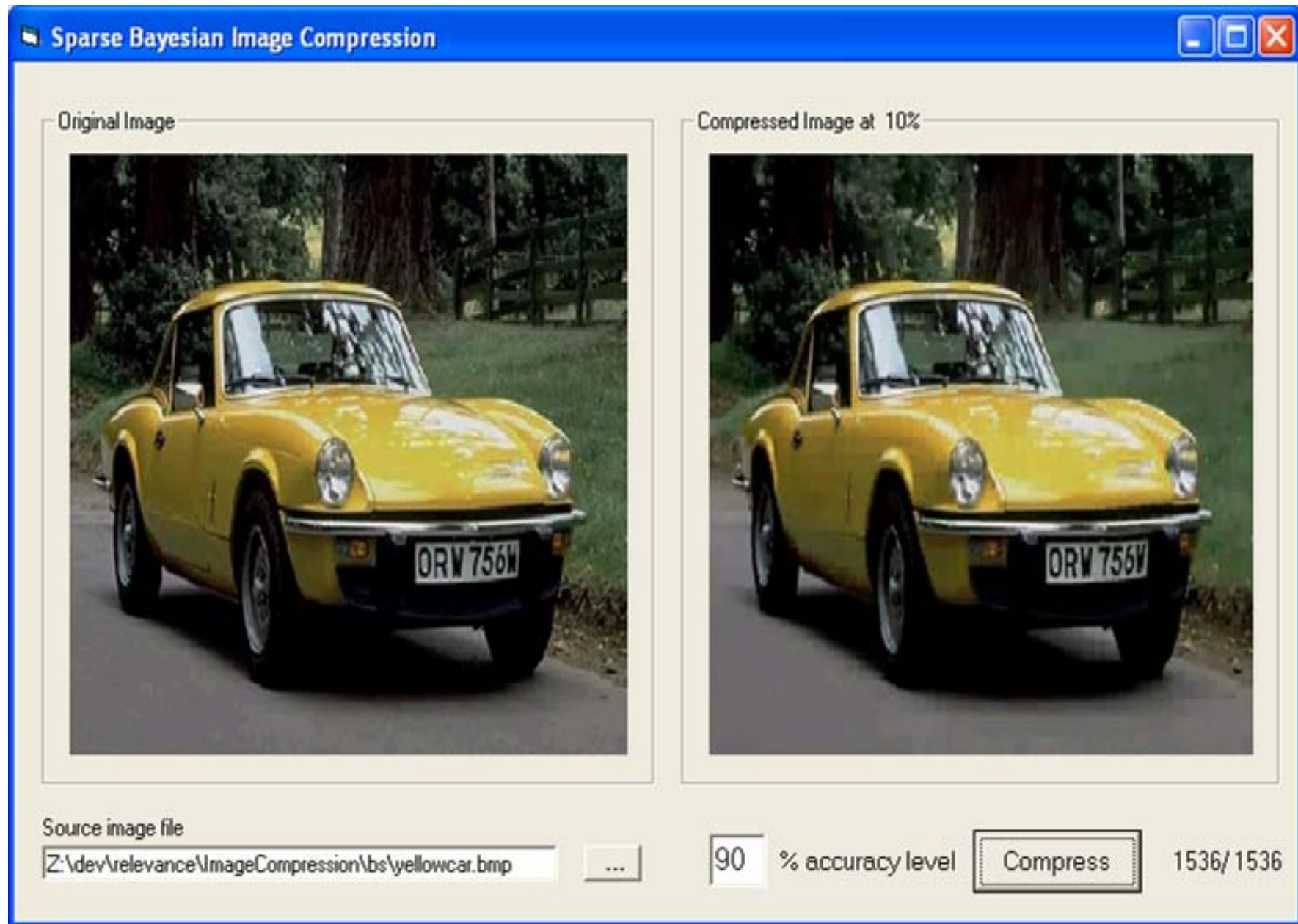
“Blocks” Data Results Summary

	<i>Heaviside</i>		<i>Heaviside + Gauss</i>	
	Bayes	ORMP	Bayes	ORMP
M	1024	1024	5120	5120
\widehat{M}	12	12	12	82
Iterations	21	11	224	82
Additions	11	11	107	82
Deletions	0	–	96	–
Re-estimates	10	–	21	–
Time	1.34s	1.19s	43.3s	24.6s

Applications: approximation (1)

- Assume the target is noise-free and is to be approximated more ‘cheaply’, e.g. an image which is to be compressed
- Choose some appropriate basis set (e.g. Gabor wavelets)
- Fix σ^2 as desired
- Run the sparse Bayes regression algorithm
- Interpretation of σ^2 has changed — it now models the approximation error, not the noise process

Applications: image compression



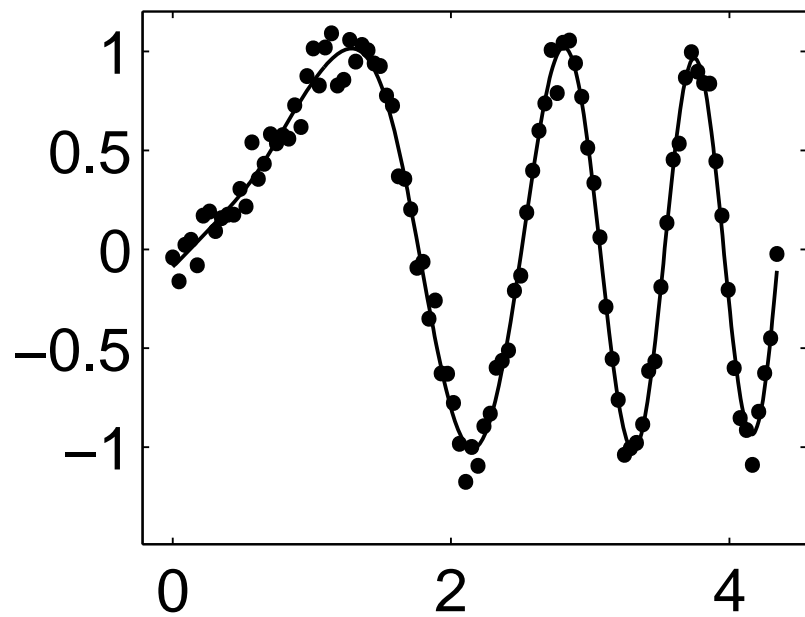
Applications: approximation (2)

- Can approximate *functions* $f(\mathbf{x})$:

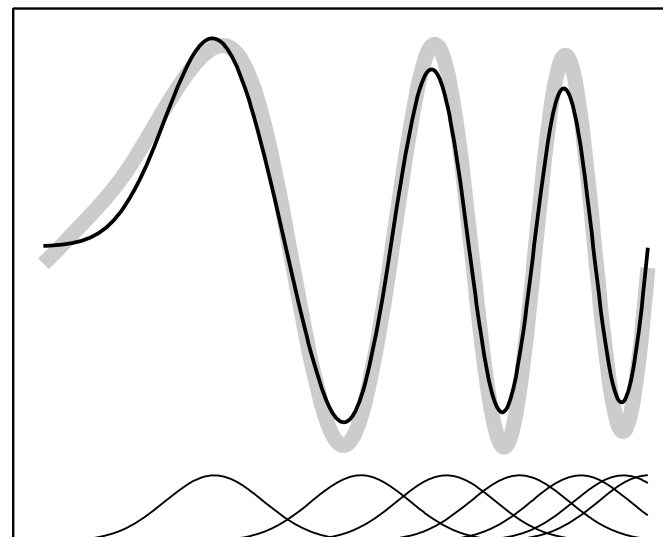
$$\text{Likelihood} \propto \exp \left\{ -\frac{1}{2\sigma^2} \int \|y(\mathbf{x}; \mathbf{w}) - f(\mathbf{x})\|^2 d\mathbf{x} \right\}$$

- Condition: we need to compute all $\int \phi_i(\mathbf{x})f(\mathbf{x}) d\mathbf{x}$ and $\int \phi_i(\mathbf{x})\phi_j(\mathbf{x}) d\mathbf{x}$
- Practical example: $f(\mathbf{x}) = \sum_j v_j \psi_j(\mathbf{x})$ with ψ_j Gaussian
- Potential target functions: Gaussian process, SVM, kernel density estimator *etc*

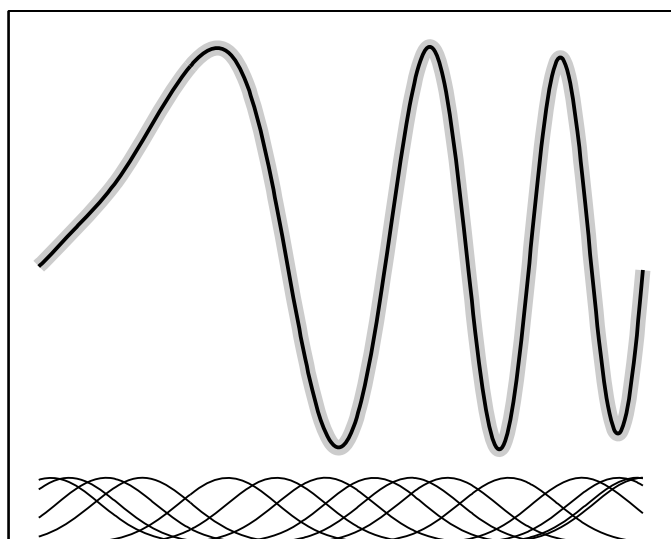
Mean Gaussian Process Predictor



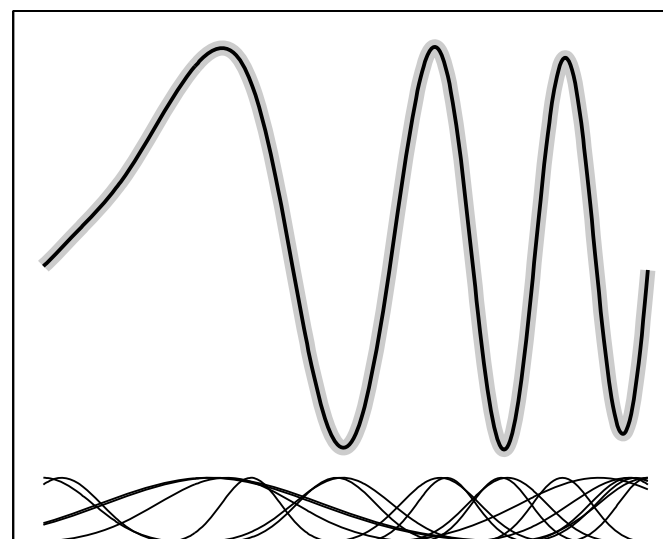
(a) $\sigma=0.100$, $M=7/100$



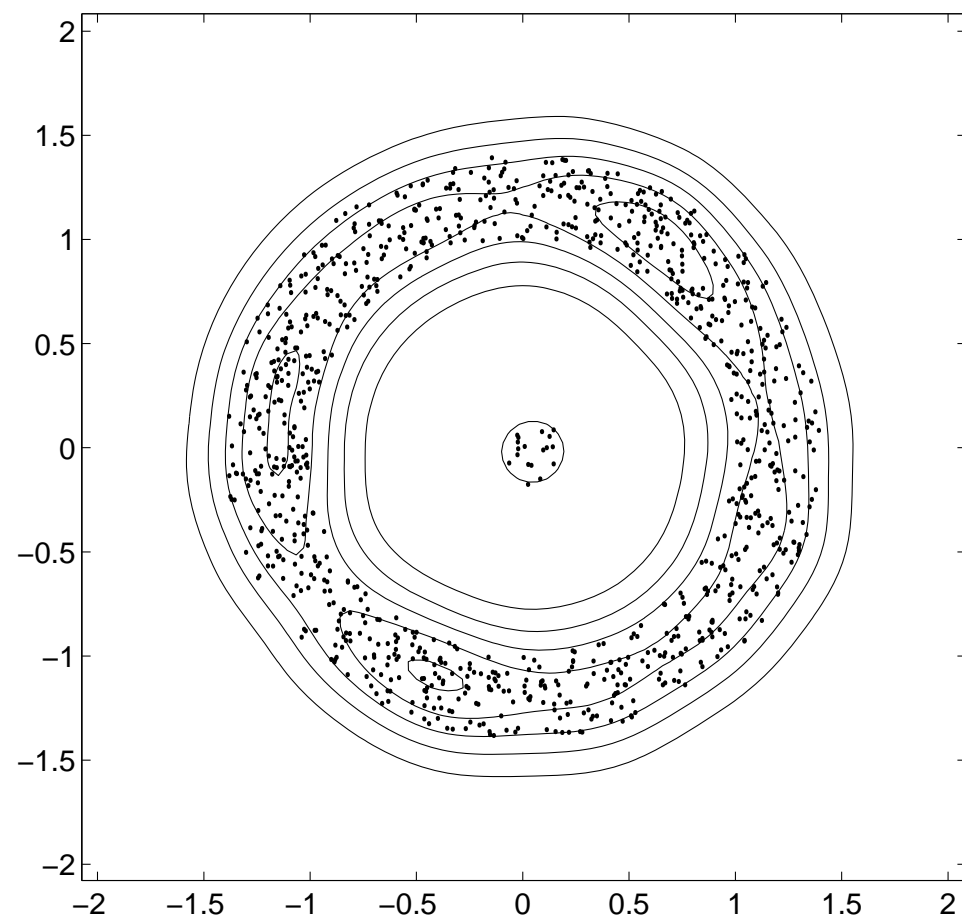
(b) $\sigma=0.001$, $M=15/100$



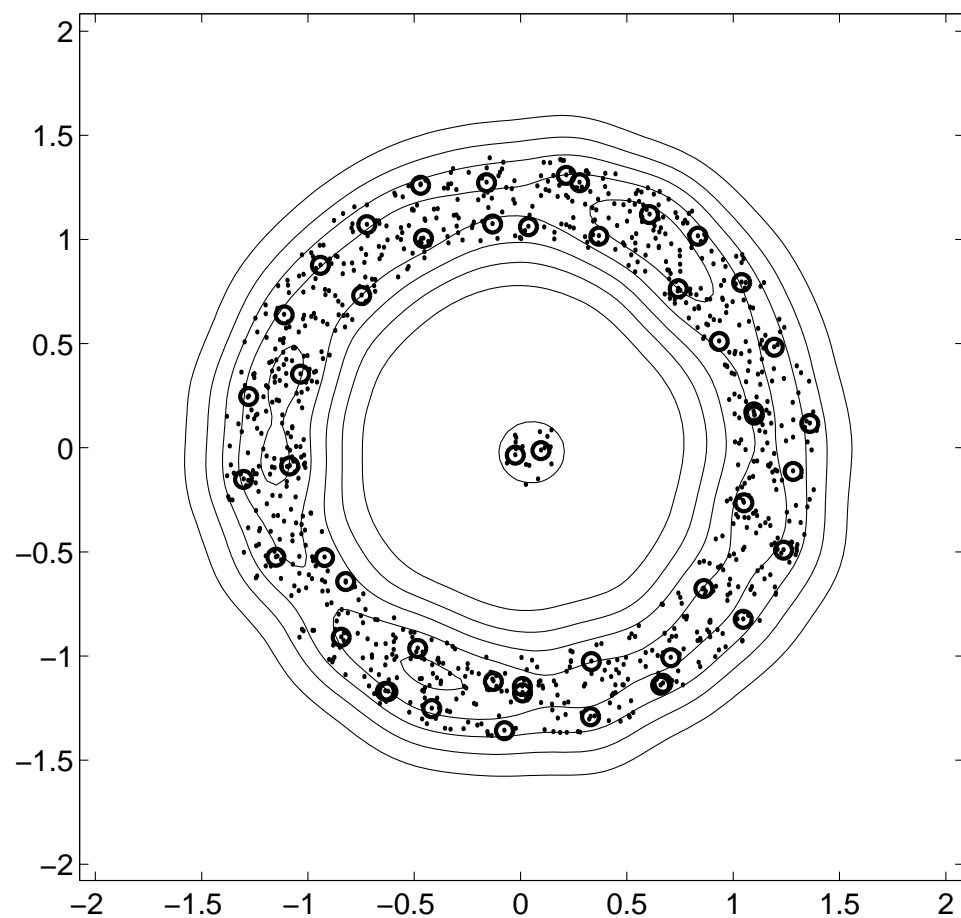
(c) $\sigma=0.001$, $M=20/2000$



Kernel density estimate with 1000 Gaussians

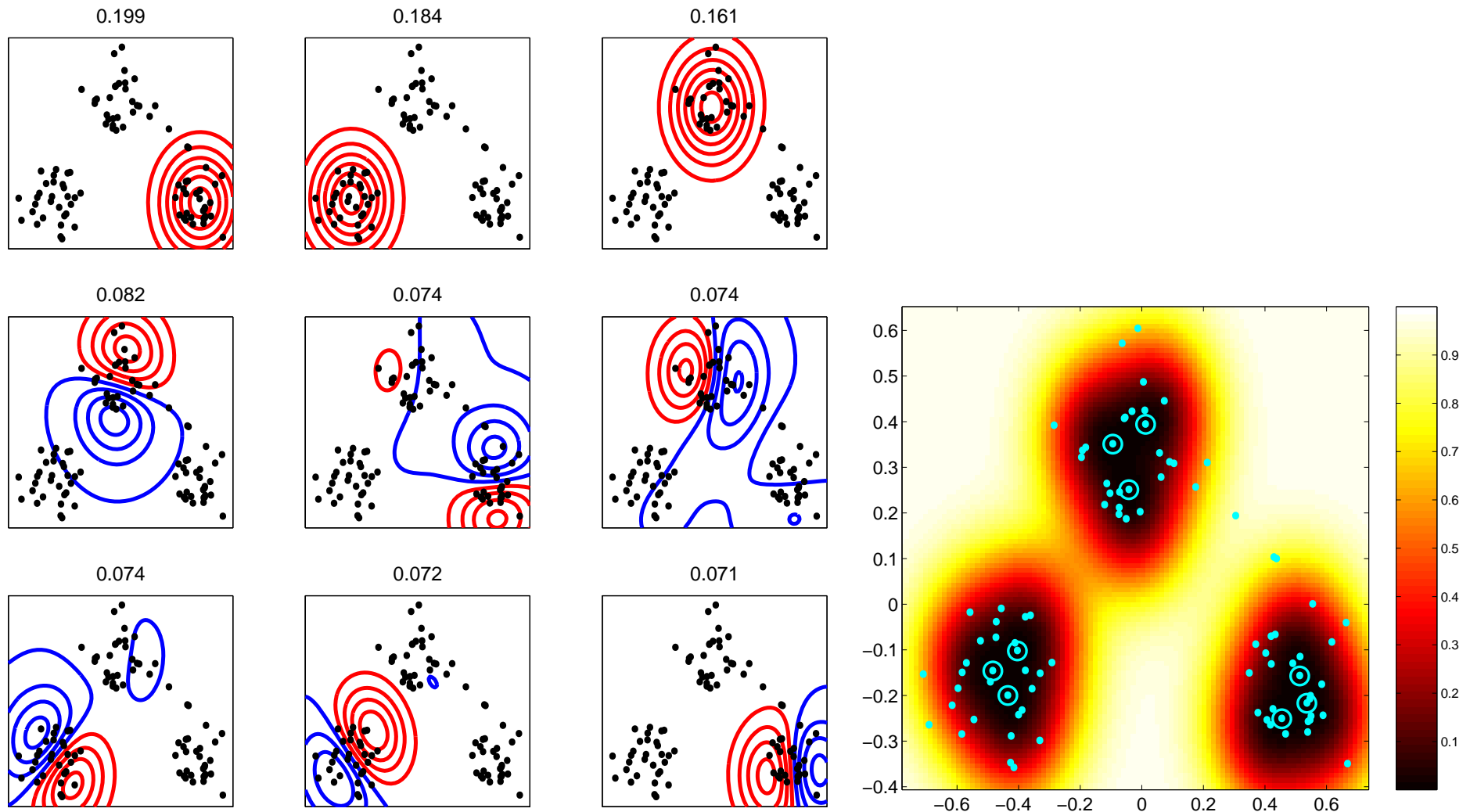


Approximation with 49 Gaussians



Applications: sparse kernel PCA

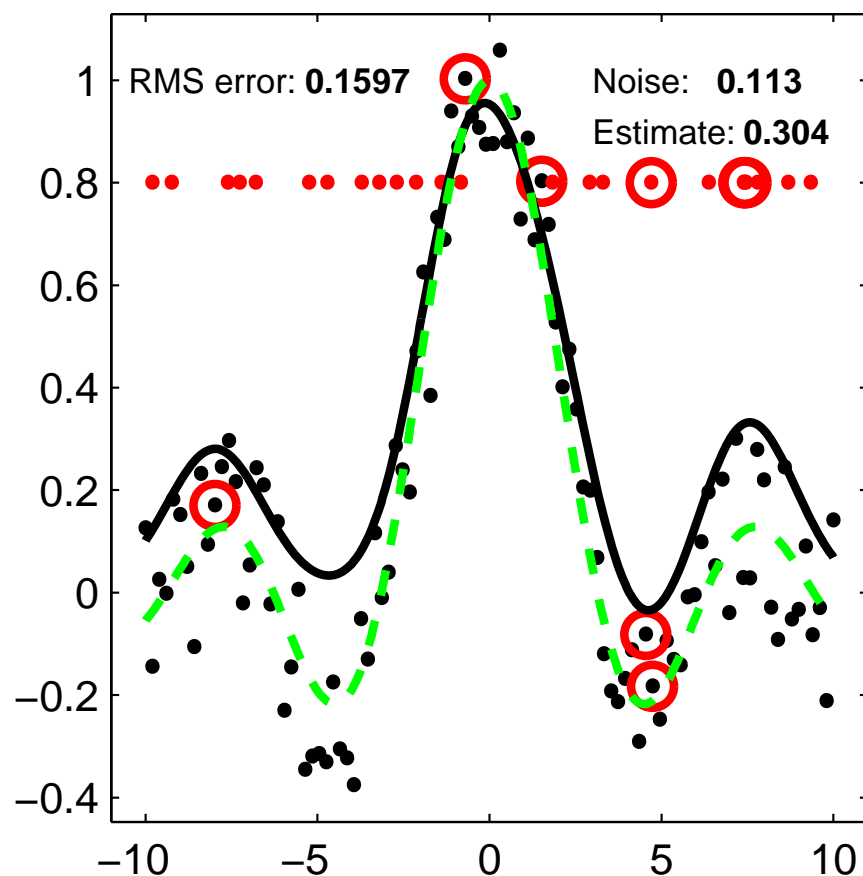
■ Work directly with $\mathbf{C} = \sum_{n=1}^N \alpha_n^{-1} \phi_n \phi_n^\top + \sigma^2 \mathbf{I}$



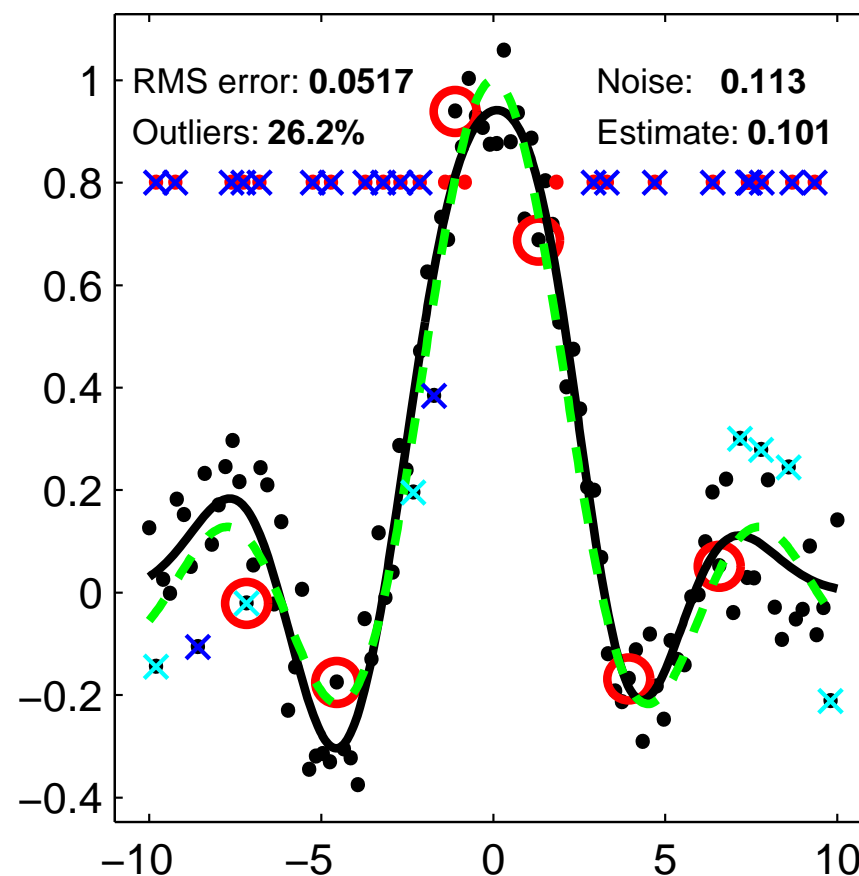
Applications: robust regression

- Exploit *variational* formalism to incorporate outlier distribution. *i.e.* ‘mixture’ likelihood: $p(\mathbf{t}|\mathbf{w}) = \theta \cdot p_{\text{data}}(\mathbf{t}|\mathbf{w}) + (1 - \theta) \cdot p_{\text{outlier}}(\mathbf{t})$

Standard RV regression



RV regression with outlier ‘detection’



Applications: Image Super-Resolution

Exploits marginalisation over the unknown high-resolution image to optimise registration parameters

Low-resolution image (1 of 16)



Low-resolution image (2 of 16)



4x Super-resolved image (Bayesian)



More Information

The screenshot shows a Microsoft Internet Explorer browser window displaying the website for the Machine Learning and Perception Group's Relevance Vector Machine (RVM) project. The browser's address bar shows the URL: <http://www.research.microsoft.com/mlp/RVM/default.htm>. The website has a blue header with the title "The Relevance Vector Machine" in yellow. On the left side, there is a vertical navigation menu with the text "machine learning & perception" and "mlp pages". Below this, there are links for "Home & Projects" and "Seminars". Further down, there is a section titled "mlp people" which lists the names of the group members: Chris Bishop, Andrew Blake, Antonio Criminisi, Michel Gangnet, Julian Gold, Thore Graepel, Ralf Herbrich, Patrick Perez, Martin Szummer, Michael Taylor, Mike Tipping, Phil Torr, and Hugo Zaragoza. The main content area on the right is titled "Sparse Bayesian Learning and the Relevance Vector Machine" and includes an "Intro" section. The intro text states: "Sparse Bayesian learning is the application of Bayesian *automatic relevance determination* (ARD) to models linear in their parameters. The 'relevance vector machine' is a special case of the technique, applied to linear kernel models of the same form as the popular 'support vector machine'." Below the intro, there is a note: "This page is continually in the process of being updated ...". The "Papers" section lists a comprehensive paper on sparse Bayesian learning from the *Journal of Machine Learning Research*, along with two early conference publications on the RVM. The bottom of the page features the Microsoft Research logo.

Machine Learning and Perception Group: The Relevance Vector Machine - Microsoft Internet Explorer

File Edit View Favorites Tools Help Links BBC News CAMWEB Google Hotmail Microsoft MLP Multimap

Back Search Favorites Media Go

Address <http://www.research.microsoft.com/mlp/RVM/default.htm>

machine learning & perception

mlp pages

Home & Projects
Seminars

mlp people

Chris Bishop
Andrew Blake
Antonio Criminisi
Michel Gangnet
Julian Gold
Thore Graepel
Ralf Herbrich
Patrick Perez
Martin Szummer
Michael Taylor
Mike Tipping
Phil Torr
Hugo Zaragoza

Microsoft Research

The Relevance Vector Machine

Sparse Bayesian Learning and the Relevance Vector Machine

Intro

Sparse Bayesian learning is the application of Bayesian *automatic relevance determination* (ARD) to models linear in their parameters. The "relevance vector machine" is a special case of the technique, applied to linear kernel models of the same form as the popular "support vector machine".

This page is continually in the process of being updated ...

Papers

A comprehensive paper on sparse Bayesian learning from the *Journal of Machine Learning Research*:

- **Sparse Bayesian learning and the relevance vector machine.** *Journal of Machine Learning Research* **1**, 211-244. [[Abstract](#)] [[Available from JMLR](#)]

There are a couple of (very minor) typos in the above [[corrections here](#)]

Two early conference publications on the *Relevance Vector Machine*:

- **The Relevance Vector Machine.** In S. A. Solla, T. K. Leen, and K.-R. Müller (Eds.), *Advances in Neural Information Processing Systems 12*, pp. 652-658. Cambridge, Mass: MIT Press. [[Abstract](#)] [[gzipped PostScript](#)]
- **Variational relevance vector machines.** In *Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence*. Accepted to appear. [[Abstract](#)] [[gzipped PostScript](#)]

Exploiting the sparse Bayes methodology to realise "sparse kernel PCA":

- Tipping, M. E. (2001). **Sparse kernel principal component analysis.** In *Advances in Neural Information Processing Systems 13*. MIT Press. [[Abstract](#)] [[gzipped PostScript](#)]

Internet

<http://www.research.microsoft.com/mlp/RVM/>