Presented at: Gaussian Process Round Table, Sheffield, 9 June 2005

Approximate Methods for GP Regression: A Survey and an Empirical Comparison

Chris Williams

School of Informatics, University of Edinburgh, UK

Overview

- Reduced-rank approximation of the Gram matrix
- Subset of Regressors
- Subset of Datapoints
- Projected Process Approximation
- Bayesian Committee Machine
- Iterative Solution of Linear Systems
- Empirical Comparison

Reduced-rank approximations of the Gram matrix

$$\tilde{K} = K_{nm} K_{mm}^{-1} K_{mn}$$

- Subset I (of size m) can be chosen randomly (Williams and Seeger), or greedily (Schölkopf and Smola)

- Drineas and Mahoney (YALEU/DCS/TR-1319, April 2005) suggest sampling the columns of K with replacement according to the distribution

$$p_i = K_{ii}^2 / \sum_j K_{jj}^2$$

to obtain the result

$$\|K - K_{nm} W_k^+ K_{mn}\| \leq \|K - K_k\| + \epsilon \sum_j K_{jj}^2$$

for 2-norm or Frobenius norm, by choosing $m = O(k/\epsilon^4)$ columns, both in expectation and with high probability. W_k is the best rank-k approximation to K_{mm}.

Gaussian Process Regression

Dataset $\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n$, Gaussian likelihood $p(y_i|f_i) \sim N(0, \sigma^2)$

$$\bar{f}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}, \mathbf{x}_i)$$

where

$$\boldsymbol{\alpha} = (K + \sigma^2 I)^{-1} \mathbf{y}$$

$$\text{var}(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}^T(\mathbf{x})(K + \sigma^2 I)^{-1} \mathbf{k}(\mathbf{x})$$

in time $O(n^3)$, with $\mathbf{k}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), \ldots, \mathbf{k}(\mathbf{x}, \mathbf{x}_n))^T$

Subset of Regressors

- Silverman (1985) showed that the *mean* GP predictor can be obtained from the finite-dimensional model

$$f(\mathbf{x}_*) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_*, \mathbf{x}_i)$$

with a prior $\boldsymbol{\alpha} \sim \mathcal{N}(\mathbf{0}, K^{-1})$

- A simple approximation to this model is to consider only a subset of regressors

$$f_{\mathsf{SR}}(\mathbf{x}_*) = \sum_{i=1}^{m} \alpha_i k(\mathbf{x}_*, \mathbf{x}_i), \qquad \text{with} \qquad \boldsymbol{\alpha}_m \sim \mathcal{N}(\mathbf{0}, K_{mm}^{-1})$$

$$\overline{f}_{\mathsf{SR}}(\mathbf{x}_*) = \mathbf{k}_m^T(\mathbf{x}_*)(K_{mn}K_{nm} + \sigma_n^2 K_{mm})^{-1} K_{mn}\mathbf{y},$$

$$\mathsf{var}_{\mathsf{SR}}(f(\mathbf{x}_*)) = \sigma_n^2 \mathbf{k}_m^T(\mathbf{x}_*)(K_{mn}K_{nm} + \sigma_n^2 K_{mm})^{-1} \mathbf{k}_m(\mathbf{x}_*).$$

Thus the posterior mean for $\boldsymbol{\alpha}_m$ is given by

$$\bar{\boldsymbol{\alpha}}_m = (K_{mn}K_{nm} + \sigma_n^2 K_{mm})^{-1} K_{mn}\mathbf{y}.$$

Under this approximation

$$\log P_{\mathsf{SR}}(\mathbf{y}|X) = -\frac{1}{2}\log|\tilde{K}+\sigma_n^2 I_n| - \frac{1}{2}\mathbf{y}^\top(\tilde{K}+\sigma_n^2 I_n)^{-1}\mathbf{y} - \frac{n}{2}\log(2\pi).$$

- Covariance function defined by the SR model has the form
  $\tilde{k}(\mathbf{x}, \mathbf{x}') = \mathbf{k}^\top(\mathbf{x}) K_{mm}^{-1} \mathbf{k}(\mathbf{x}')$

- Problems with predictive variance far from datapoints if kernels decay to zero

- Greedy selection: Luo and Wahba (1997) minimize RSS $|\mathbf{y} - K_{nm}\boldsymbol{\alpha}_m|^2$, Smola and Bartlett (2001) minimize

  $$\frac{1}{\sigma_n^2}|\mathbf{y} - K_{nm}\bar{\boldsymbol{\alpha}}_m|^2 + \bar{\boldsymbol{\alpha}}_m^\top K_{mm}\bar{\boldsymbol{\alpha}}_m = \mathbf{y}^\top(\tilde{K} + \sigma_n^2 I_n)^{-1}\mathbf{y},$$

  Quiñonero-Candela (2004) suggests using the approximate log marginal likelihood $\log P_{\mathsf{SR}}(\mathbf{y}|X)$

Nyström method

- Replaces K by $\tilde{K}$, but not $\mathbf{k}(\mathbf{x}_*)$

- Better to replace systematically, as in SR

Subset of Datapoints

- Simply keep m datapoints, discard the rest

- Greedy selection using differential entropy score (IVM; Lawrence, Seeger, Herbrich, 2003) or information gain score

Projected Process Approximation

- The SR method is unattractive as it is based on a *degenerate* GP

- The PP approximation is a non-degenerate process model but represents only $m < n$ latent function values explicitly

$$\mathbb{E}[\mathbf{f}_{n-m}|\mathbf{f}_m] = K_{(n-m)m}K_{mm}^{-1}\mathbf{f}_m$$

so that

$$Q(\mathbf{y}|\mathbf{f}_m) \sim \mathcal{N}(\mathbf{y}; K_{nm}K_{mm}^{-1}\mathbf{f}_m, \sigma_n^2 I),$$

- Combine $Q(\mathbf{y}|\mathbf{f}_m)$ and $P(\mathbf{f}_m)$ to obtain $Q(\mathbf{f}_m|\mathbf{y})$

- Predictive mean is the same as SR, but variance is never smaller than SR predictive variance

$$\mathbb{E}_Q[f(\mathbf{x}_*)] = \mathbf{k}_m^\top(\mathbf{x}_*)(\sigma_n^2 K_{mm} + K_{mn}K_{nm})^{-1}K_{mn}\mathbf{y},$$

$$\text{var}_Q(f(\mathbf{x}_*)) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_m^\top(\mathbf{x}_*)K_{mm}^{-1}\mathbf{k}_m(\mathbf{x}_*)$$

$$+ \sigma_n^2 \mathbf{k}_m^\top(\mathbf{x}_*)(\sigma_n^2 K_{mm} + K_{mn}K_{nm})^{-1}\mathbf{k}_m(\mathbf{x}_*).$$

- Csato and Opper (2002) use an online algorithm for determining the active set

- Seeger, Williams, Lawrence (2003) suggest a greedy algorithm using an approximation of the information gain

Bayesian Committee Machine

- Split the dataset into p parts and assume that $p(\mathcal{D}_1, \ldots, \mathcal{D}_p|\mathbf{f}_*) \simeq \prod_{i=1}^{p} p(\mathcal{D}_i|\mathbf{f}_*)$ (Tresp, 2000)

$$\mathbb{E}_q[\mathbf{f}_*|\mathcal{D}] = [\mathrm{cov}_q(\mathbf{f}_*|\mathcal{D})] \sum_{i=1}^{p} [\mathrm{cov}(\mathbf{f}_*|\mathcal{D}_i)]^{-1} \mathbb{E}[\mathbf{f}_*|\mathcal{D}_i],$$

$$[\mathrm{cov}_q(\mathbf{f}_*|\mathcal{D})]^{-1} = -(p-1)K_{**}^{-1} + \sum_{i=1}^{p} [\mathrm{cov}(\mathbf{f}_*|\mathcal{D}_i)]^{-1},$$

- Datapoints can be assigned to clusters randomly, or by using clustering

- Use $p = n/m$ and divide the test set into blocks of size m to ensure that all matrices are $m \times m$

- Note that BCM is *transductive*. Also, if n_* is small it may be useful to hallucinate some test points

Iterative Solution of Linear Systems

- Can solve $(K + \sigma_n^2 I)\mathbf{v} = \mathbf{y}$ by iterative methods, e.g. conjugate gradients (CG).

- However, this has $O(kn^2)$ scaling for k iterations

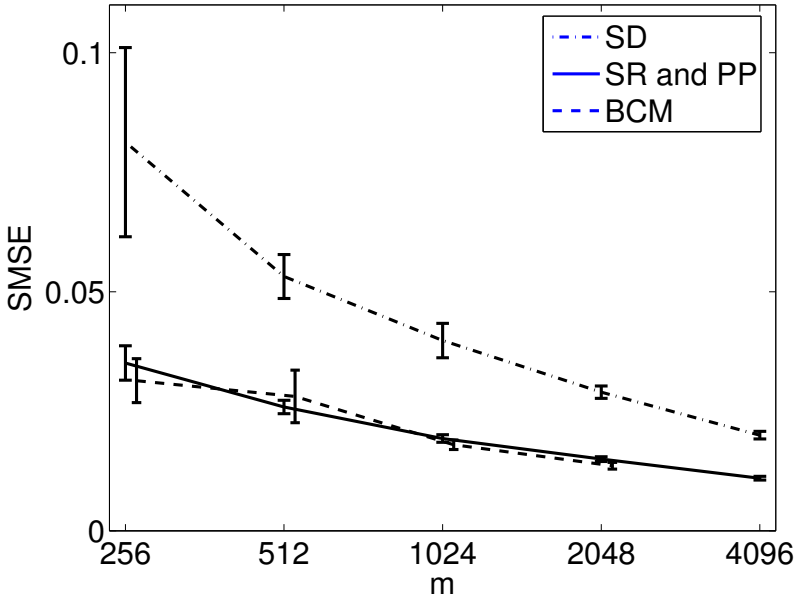
- Can be speeded up using approximate matrix-vector multiplication, e.g. Improved Fast Gauss Transform (Yang et al, 2005)

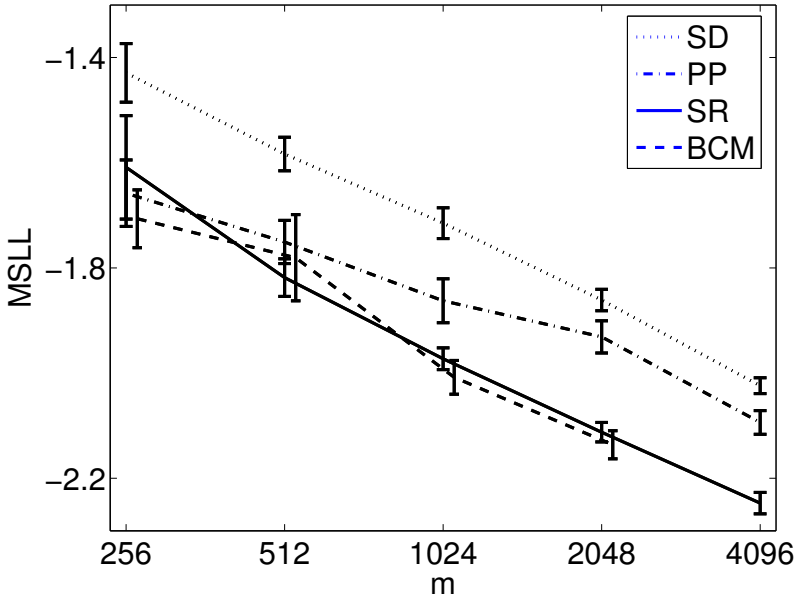
Complexity

Method	Storage	Initialization	Mean	Variance
SD	$O(m^2)$	$O(m^3)$	$O(m)$	$O(m^2)$
SR	$O(mn)$	$O(m^2n)$	$O(m)$	$O(m^2)$
PP	$O(mn)$	$O(m^2n)$	$O(m)$	$O(m^2)$
BCM	$O(mn)$		$O(mn)$	$O(mn)$

Empirical Comparison

- Robot arm problem, 44,484 training cases in 21-d, 4,449 test cases
- For SD method subset of size m was chosen at random, hyperparameters set by optimizing marginal likelihood (ARD). Repeated 10 times
- For SR, PP and BCM methods same subsets/hyperparameters were used (BCM: hyperparameters only)


SD
SR and PP
BCM
SMSE
0.1
0.05
0
256
512
1024
2048
4096
m


SD
PP
SR
BCM
MSLL
−1.4
−1.8
−2.2
256
512
1024
2048
4096
m

Method	m	SMSE	MSLL	mean runtime (s)
SD	256	0.0813 $\pm$ 0.0198	-1.4291 $\pm$ 0.0558	0.8
	512	0.0532 $\pm$ 0.0046	-1.5834 $\pm$ 0.0319	2.1
	1024	0.0398 $\pm$ 0.0036	-1.7149 $\pm$ 0.0293	6.5
	2048	0.0290 $\pm$ 0.0013	-1.8611 $\pm$ 0.0204	25.0
	4096	0.0200 $\pm$ 0.0008	-2.0241 $\pm$ 0.0151	100.7
SR	256	0.0351 $\pm$ 0.0036	-1.6088 $\pm$ 0.0984	11.0
	512	0.0259 $\pm$ 0.0014	-1.8185 $\pm$ 0.0357	27.0
	1024	0.0193 $\pm$ 0.0008	-1.9728 $\pm$ 0.0207	79.5
	2048	0.0150 $\pm$ 0.0005	-2.1126 $\pm$ 0.0185	284.8
	4096	0.0110 $\pm$ 0.0004	-2.2474 $\pm$ 0.0204	927.6
PP	256	0.0351 $\pm$ 0.0036	-1.6580 $\pm$ 0.0632	17.3
	512	0.0259 $\pm$ 0.0014	-1.7508 $\pm$ 0.0410	41.4
	1024	0.0193 $\pm$ 0.0008	-1.8625 $\pm$ 0.0417	95.1
	2048	0.0150 $\pm$ 0.0005	-1.9713 $\pm$ 0.0306	354.2
	4096	0.0110 $\pm$ 0.0004	-2.0940 $\pm$ 0.0226	964.5
BCM	256	0.0314 $\pm$ 0.0046	-1.7066 $\pm$ 0.0550	506.4
	512	0.0281 $\pm$ 0.0055	-1.7807 $\pm$ 0.0820	660.5
	1024	0.0180 $\pm$ 0.0010	-2.0081 $\pm$ 0.0321	1043.2
	2048	0.0136 $\pm$ 0.0007	-2.1364 $\pm$ 0.0266	1920.7

- For random subset selection, results suggest that BCM and SR perform best, and that SR is faster

- Some experiments using active selection for the SD method (IVM) and for the SR method did not lead to significant improvements in performance

- BCM using p-means clustering also did not lead to significant improvements in performance

- Cf Schwaighofer and Tresp (2003) who found advantage with BCM on KIN datasets

- For these experiments the hyperparameters were set using SD method. How would results compare if we, say, optimized the approximate marginal likelihood for each method?