Fast and scalable spike and slab variable selection in high-dimensional Gaussian processes

Hugh Dance[1] and Brooks Paige[1]

March 2022

[1]University College London: hughwdance@gmail.com, b.paige@ucl.ac.uk

Setting

- Samples $(x_i, y_i)_{i=1}^n : x \in \mathcal{X}^d \subseteq \mathbb{R}^d, y \in \mathcal{Y} = \mathbb{R}$

Setting

- Samples $(x_i, y_i)_{i=1}^n : x \in \mathcal{X}^d \subseteq \mathbb{R}^d, y \in \mathcal{Y} = \mathbb{R}$
- Want to learn/approximate $f(\cdot) : \mathcal{X}^d \to \mathcal{Y}$

Setting

- Samples $(x_i, y_i)_{i=1}^n : x \in \mathcal{X}^d \subseteq \mathbb{R}^d, y \in \mathcal{Y} = \mathbb{R}$
- Want to learn/approximate $f(\cdot) : \mathcal{X}^d \to \mathcal{Y}$
- High-dimensional inputs - which are relevant to $f(\cdot)$?

Setting

- Samples $(x_i, y_i)_{i=1}^n : x \in \mathcal{X}^d \subseteq \mathbb{R}^d, y \in \mathcal{Y} = \mathbb{R}$
- Want to learn/approximate $f(\cdot) : \mathcal{X}^d \to \mathcal{Y}$
- High-dimensional inputs - which are relevant to $f(\cdot)$?
- Variable selection:
  1. improve predictive accuracy
  2. reduce downstream data collection costs
  3. understand 'meaningful' relationships

Gaussian process regression (GPR)

$$
\begin{aligned}
y &= f(x) + \epsilon : \epsilon \sim \mathcal{N}(0, \sigma^2) \\
f(x) &\sim \mathcal{GP}(m(x), k(x, x'))
\end{aligned}
$$

Gaussian process regression (GPR)

$$y = f(x) + \epsilon : \epsilon \sim \mathcal{N}(0, \sigma^2)$$
$$f(x) \sim \mathcal{GP}(m(x), k(x, x'))$$

- A Gaussian process $f(x)$ is a random function where any $f(x_1), ..., f(x_n)$ are MVN
- Properties determined by mean $m(x)$, covariance (kernel) $k_\alpha(x, x')$

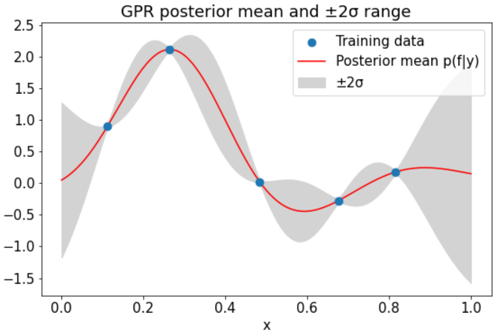
Gaussian process regression (GPR)

1. $\alpha^* = \text{argmax}_\alpha \{ \underbrace{\int p(y|f)p(f|\alpha)df}_{\text{marginal likelihood}} \}$

Gaussian process regression (GPR)

1. $\alpha^* = \text{argmax}_\alpha \{ \underbrace{\int p(y|f)p(f|\alpha)df}_{\text{marginal likelihood}} \}$

2. $\underbrace{p(f(\cdot)|y)}_{\text{posterior}} = \frac{p(y|f(\cdot))p(f(\cdot)|\alpha)}{p(y)}$



GPR variable selection with automatic relevance determination (ARD)[3]

1. Use 'automatic relevance determination' (ARD) kernel[2]

$$k_{ARD}(x, x) = k(\theta \odot x, \theta \odot x') : \theta \in \mathbb{R}_+^d$$

[2] e.g. stationary isotropic monotone $k(\cdot, \cdot) : \frac{\partial}{\partial \theta_j} \mathbb{E}[(f(x + he_j) - f(x))^2] > 0 \implies \theta$ as relevance measure

[3] MacKay (1996); Rasmussen and Williams (2006)

GPR variable selection with automatic relevance determination (ARD)[3]

1. Use 'automatic relevance determination' (ARD) kernel[2]

$$k_{ARD}(x, x) = k(\theta \odot x, \theta \odot x') : \theta \in \mathbb{R}_+^d$$

2. Optimise log marginal likelihood (ML-II):

$$\theta^* = \mathrm{argmax}_\theta \{\log p(y|\theta)\}$$

---

[2] e.g. stationary isotropic monotone $k(\cdot, \cdot) : \frac{\partial}{\partial \theta_j}\mathbb{E}[(f(x + he_j) - f(x))^2] > 0 \implies \theta$ as relevance measure

[3] MacKay (1996); Rasmussen and Williams (2006)

GPR variable selection with automatic relevance determination (ARD)[3]

1. Use 'automatic relevance determination' (ARD) kernel[2]

$$k_{ARD}(x, x) = k(\theta \odot x, \theta \odot x') : \theta \in \mathbb{R}_+^d$$

2. Optimise log marginal likelihood (ML-II):

$$\theta^* = \text{argmax}_\theta \{\log p(y|\theta)\}$$

3. Hard threshold θ:

$$\theta_j^* \leftarrow \theta_j^* \mathbb{I}(\theta_j^* \geq \beta) : \beta \in \mathbb{R}_+$$

---

[2] e.g. stationary isotropic monotone $k(\cdot, \cdot) : \frac{\partial}{\partial \theta_j} \mathbb{E}[(f(x + he_j) - f(x))^2] > 0 \implies \theta$ as relevance measure

[3] MacKay (1996); Rasmussen and Williams (2006)

GPR variable selection with automatic relevance determination (ARD)[3]

1. Use 'automatic relevance determination' (ARD) kernel[2]

$$k_{ARD}(x, x) = k(\theta \odot x, \theta \odot x') : \theta \in \mathbb{R}_+^d$$

2. Optimise log marginal likelihood (ML-II):

$$\theta^* = \mathrm{argmax}_\theta \{\log p(y|\theta)\}$$

3. Hard threshold θ:

$$\theta_j^* \leftarrow \theta_j^* \mathbb{I}(\theta_j^* \geq \beta) : \beta \in \mathbb{R}_+$$

ML-II complexity penalty $\implies$ irrelevant $\theta_j \to 0$

---

[2] e.g. stationary isotropic monotone $k(\cdot, \cdot) : \frac{\partial}{\partial \theta_j} \mathbb{E}[(f(x + he_j) - f(x))^2] > 0 \implies \theta$ as relevance measure

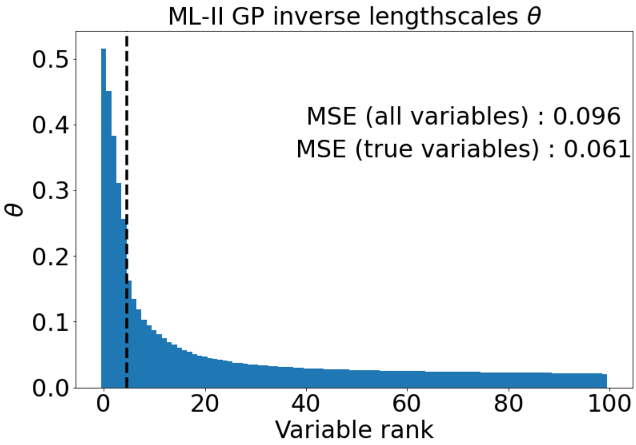
[3] MacKay (1996); Rasmussen and Williams (2006)

ARD limitations[4] - toy example

[4] Cawley and Talbot (2010); Mohammed and Cawley (2017); Ober et al. (2021)

ARD limitations[4] - toy example

$x \sim \mathcal{N}_{100}(0, I)$, $y \sim \mathcal{N}(\sum_{j=1}^{5} sin(a_j x_j)), \sigma^2), \sigma^2 = \frac{\sigma_y^2}{20}$



[4] Cawley and Talbot (2010); Mohammed and Cawley (2017); Ober et al. (2021)

An alternative: spike and slab priors

An alternative: spike and slab priors

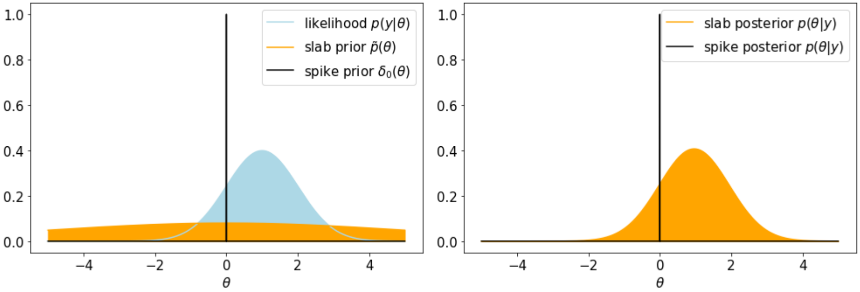
1. Place spike and slab prior on inverse lengthscales θ:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\tilde{p}(\theta_j)}_{\text{slab}} + (1-\gamma_j)\underbrace{\delta_0(\theta_j)}_{\text{spike}} \quad , \quad p(\gamma_j) = \mathcal{B}ern(\gamma_j|\pi)$$

An alternative: spike and slab priors

1. Place spike and slab prior on inverse lengthscales θ:

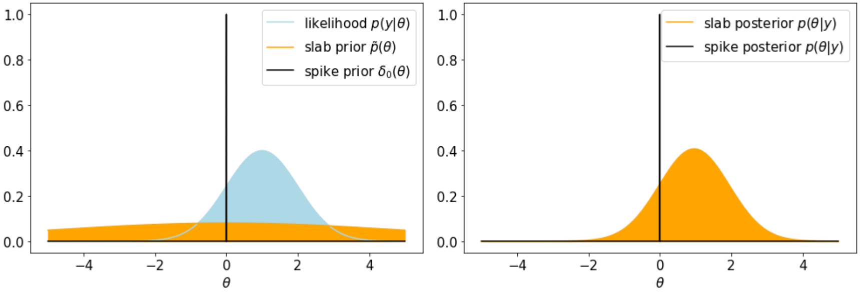
$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\tilde{p}(\theta_j)}_{\text{slab}} + (1-\gamma_j) \underbrace{\delta_0(\theta_j)}_{\text{spike}} \quad , \quad p(\gamma_j) = \mathcal{B}ern(\gamma_j|\pi)$$



An alternative: spike and slab priors

1. Place spike and slab prior on inverse lengthscales θ:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\tilde{p}(\theta_j)}_{\text{slab}} + (1-\gamma_j) \underbrace{\delta_0(\theta_j)}_{\text{spike}} \quad , \quad p(\gamma_j) = \mathcal{B}ern(\gamma_j|\pi)$$

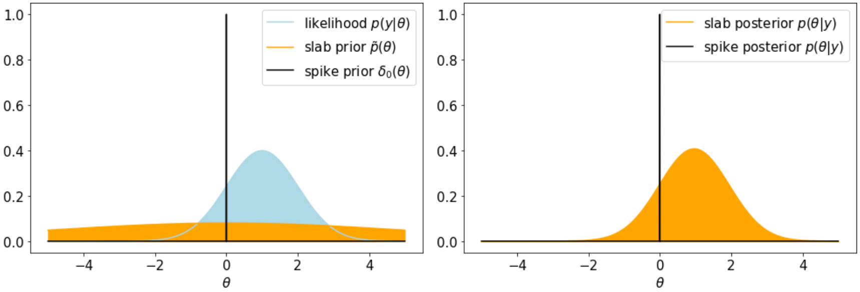


2. Recover posterior approximation: $q(\theta,\gamma) \approx p(\theta,\gamma|y) \propto p(y|\theta)p(\theta|\gamma)p(\gamma)$

An alternative: spike and slab priors

1. Place spike and slab prior on inverse lengthscales θ:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\tilde{p}(\theta_j)}_{\text{slab}} + (1-\gamma_j)\underbrace{\delta_0(\theta_j)}_{\text{spike}} \quad , \quad p(\gamma_j) = \mathcal{B}ern(\gamma_j|\pi)$$



2. Recover posterior approximation: $q(\theta, \gamma) \approx p(\theta, \gamma|y) \propto p(y|\theta)p(\theta|\gamma)p(\gamma)$
3. Variable select based on $q(\gamma_j = 1|y)$ or $\text{argmax}_\gamma\{q(\gamma)\}$.

An alternative: spike and slab priors

Existing implementations

- **Supervised GPR**[5]: MCMC based $\implies$ costly in high-dimensions (2^d search $+$ no HMC)

[5] Savitsky et al. (2011); Linkletter et al. (2006); Qamar and Tokdar (2014)
[6] Dai et al. (2015)

An alternative: spike and slab priors

Existing implementations

- **Supervised GPR**[5]: MCMC based $\implies$ costly in high-dimensions (2^d search + no HMC)
- **Unsupervised GP-LVM**[6]: Variational inference $\implies$ fast but intractable in supervised GPR

[5] Savitsky et al. (2011); Linkletter et al. (2006); Qamar and Tokdar (2014)
[6] Dai et al. (2015)

An alternative: spike and slab priors

Existing implementations

- **Supervised GPR**[5]: MCMC based $\implies$ costly in high-dimensions (2^d search + no HMC)
- **Unsupervised GP-LVM**[6]: Variational inference $\implies$ fast but intractable in supervised GPR

can we develop fast and scalable VI scheme for spike and slab priors in GPR?

[5] Savitsky et al. (2011); Linkletter et al. (2006); Qamar and Tokdar (2014)
[6] Dai et al. (2015)

An alternative: spike and slab priors

Existing implementations

- **Supervised GPR**[5]: MCMC based $\implies$ costly in high-dimensions (2^d search + no HMC)
- **Unsupervised GP-LVM**[6]: Variational inference $\implies$ fast but intractable in supervised GPR

can we develop fast and scalable VI scheme for spike and slab priors in GPR?

Paananen et al. (2019): better relevance measure than ARD, but thresholding still challenging in high-dimensions

[5] Savitsky et al. (2011); Linkletter et al. (2006); Qamar and Tokdar (2014)
[6] Dai et al. (2015)

An alternative: spike and slab priors

Existing implementations

- **Supervised GPR**[5]: MCMC based $\implies$ costly in high-dimensions (2^d search + no HMC)
- **Unsupervised GP-LVM**[6]: Variational inference $\implies$ fast but intractable in supervised GPR

can we develop fast and scalable VI scheme for spike and slab priors in GPR?

Paananen et al. (2019): better relevance measure than ARD, but thresholding still challenging in high-dimensions

[5] Savitsky et al. (2011); Linkletter et al. (2006); Qamar and Tokdar (2014)
[6] Dai et al. (2015)

Variational inference for spike and slab GPR

- Want $q^*(\theta, \gamma) = \text{argmin}_{q \in \mathcal{Q}}\{KL[q(\theta, \gamma)|p(\theta, \gamma|y)]\}$

Variational inference for spike and slab GPR

- Want $q^*(\theta, \gamma) = \text{argmin}_{q \in \mathcal{Q}}\{KL[q(\theta, \gamma)|p(\theta, \gamma|y)]\}$
- Equivalent to maximising free energy / evidence lower bound:

$$\mathcal{F} = \langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} - KL[q(\theta, \gamma)||p(\theta, \gamma)]\}$$

Variational inference (challenges) for spike and slab GPR

$$\mathcal{F} = \langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} - KL[q(\theta,\gamma)||p(\theta,\gamma)]$$

Variational inference (challenges) for spike and slab GPR

$$\mathcal{F} = \langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} - KL[q(\theta,\gamma)||p(\theta,\gamma)]$$

1. $\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)}$ intractable:

$$\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} = -\frac{1}{2} \int \left(\log |K(\theta)| + y^T K(\theta)^{-1} y \right) q(\theta,\gamma) d\theta d\gamma + ...$$

Variational inference (challenges) for spike and slab GPR

$$\mathcal{F} = \langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} - KL[q(\theta,\gamma)||p(\theta,\gamma)]$$

1. $\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)}$ intractable:

$$\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} = -\frac{1}{2} \int \left(\log|K(\theta)| + y^T K(\theta)^{-1} y \right) q(\theta,\gamma) d\theta d\gamma + \dots$$

$\implies$ need $q(\theta)q(\gamma) = q(\theta,\gamma)$ for (unbiased) reparameterisation gradients

Variational inference (challenges) for spike and slab GPR

$$\mathcal{F} = \langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} - KL[q(\theta,\gamma)||p(\theta,\gamma)]$$

1. $\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)}$ intractable:

$$\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} = -\frac{1}{2} \int \left(\log |K(\theta)| + y^T K(\theta)^{-1} y \right) q(\theta,\gamma) d\theta d\gamma + \dots$$

$\implies$ need $q(\theta)q(\gamma) = q(\theta,\gamma)$ for (unbiased) reparameterisation gradients

2. ....but then $KL[q(\theta)q(\gamma)||p(\theta,\gamma)]$ undefined:

Variational inference (challenges) for spike and slab GPR

$$\mathcal{F} = \langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} - KL[q(\theta,\gamma)||p(\theta,\gamma)]$$

1. $\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)}$ intractable:

$$\langle \log p(y|\theta) \rangle_{q(\theta,\gamma)} = -\frac{1}{2} \int \left(\log |K(\theta)| + y^T K(\theta)^{-1} y \right) q(\theta,\gamma) d\theta d\gamma + \ldots$$

$\implies$ need $q(\theta)q(\gamma) = q(\theta,\gamma)$ for (unbiased) reparameterisation gradients

2. ....but then $KL[q(\theta)q(\gamma)||p(\theta,\gamma)]$ undefined:

$$\langle \log \delta_0(\theta_j) \rangle_{q(\theta_j)} = -\infty \quad \forall q(\cdot) \neq \delta_0(\cdot)$$

Our variational inference strategy

1. Gaussian approximation to the Dirac spike:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\mathcal{N}(0, \sigma_1^2)}_{\text{slab}} + (1 - \gamma_j) \underbrace{\mathcal{N}(0, \sigma_0^2)}_{\text{spike}} \quad : \quad \sigma_0^2 \ll 1 \ll \sigma_1^2$$

Our variational inference strategy

1. Gaussian approximation to the Dirac spike:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\mathcal{N}(0, \sigma_1^2)}_{\text{slab}} + (1-\gamma_j)\underbrace{\mathcal{N}(0, \sigma_0^2)}_{\text{spike}} \quad : \quad \sigma_0^2 \ll 1 \ll \sigma_1^2$$

$\implies$ $KL[q(\theta)q(\gamma)||p(\theta, \gamma)]$ is defined.

Our variational inference strategy

1. Gaussian approximation to the Dirac spike:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\mathcal{N}(0, \sigma_1^2)}_{\text{slab}} + (1-\gamma_j) \underbrace{\mathcal{N}(0, \sigma_0^2)}_{\text{spike}} \quad : \quad \sigma_0^2 \ll 1 \ll \sigma_1^2$$

$\implies KL[q(\theta)q(\gamma)||p(\theta,\gamma)]$ is defined.

2. $q(\theta,\gamma) = q(\theta)q(\gamma)$ with reparameterisable $q_\psi(\theta)$

Our variational inference strategy

1. Gaussian approximation to the Dirac spike:

$$p(\theta_j|\gamma_j) = \gamma_j \underbrace{\mathcal{N}(0, \sigma_1^2)}_{\text{slab}} + (1-\gamma_j) \underbrace{\mathcal{N}(0, \sigma_0^2)}_{\text{spike}} \quad : \quad \sigma_0^2 \ll 1 \ll \sigma_1^2$$

$\implies$ $KL[q(\theta)q(\gamma)||p(\theta,\gamma)]$ is defined.

2. $q(\theta,\gamma) = q(\theta)q(\gamma)$ with reparameterisable $q_\psi(\theta)$

$\implies$ fast approximate co-ordinate ascent strategy available

aCAVI: approximate coordinate ascent variational inference (CAVI)

(Exact) CAVI update to $q(\gamma)$:

$$q(\gamma) \propto exp\{\langle logp(\theta|\gamma)\rangle_{q(\theta)}\}p(\gamma) = \prod_j \mathcal{B}ern(\gamma_j|\lambda_j)$$

$\mathcal{O}(d)$ cost

aCAVI: approximate coordinate ascent variational inference (CAVI)

(Exact) CAVI update to $q(\gamma)$:

$$q(\gamma) \propto exp\{\langle logp(\theta|\gamma)\rangle_{q(\theta)}\}p(\gamma) = \prod_j \mathcal{B}ern(\gamma_j|\lambda_j)$$

$\mathcal{O}(d)$ cost

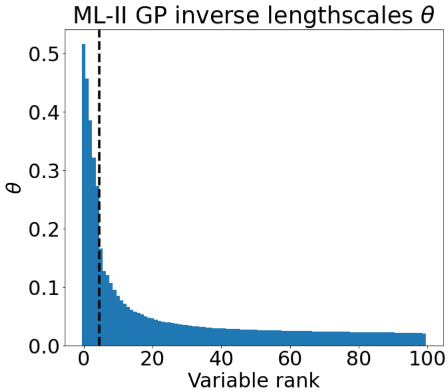
(Approximate) CAVI update to $q_\psi(\theta)$ using rep-grad SVI:

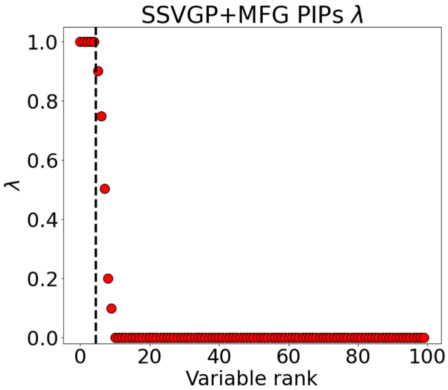
For $t = 1, \ldots T$:

$$\psi \leftarrow \psi + \eta \odot \hat{\nabla}_\psi \mathcal{F}$$

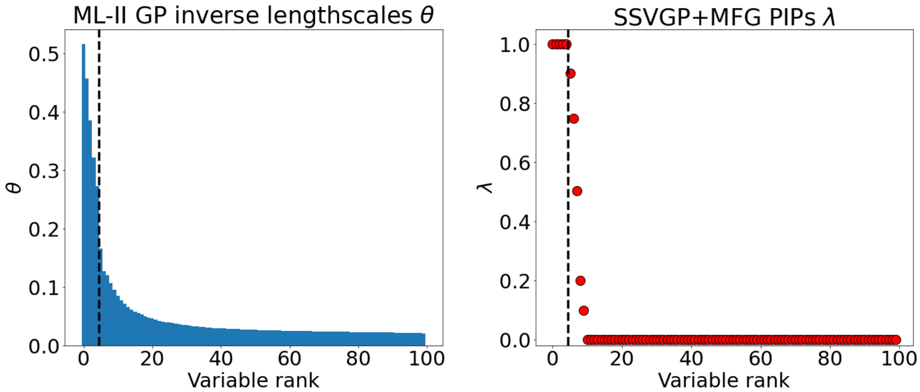
$\mathcal{O}(sn^2 d) + \mathcal{O}(sn^3)$ cost

Toy example results





Toy example results



Method	MSE	Runtime (s)
ML-II GP	0.096 ± 0.014	$\mathbf{5.0} \pm 0.7$
SSVGP+MFG	$\mathbf{0.068} \pm 0.011$	27.8 ± 0.5

Addressing hyperparameter sensitivity

$$v = \frac{1}{\sigma_0^2}, c = \frac{\sigma_0^2}{\sigma_1^2}$$

Addressing hyperparameter sensitivity

$$v = \frac{1}{\sigma_0^2}, c = \frac{\sigma_0^2}{\sigma_1^2}$$

v	10^2	10^3	10^4	10^5	10^6
$\bar{\lambda}$ (toy example)	0	0.05	0.07	0.34	1

Bayesian model averaging

$$\mathcal{M}_k = \{v_k, c, \pi\}$$

$$\begin{tikzcd}
q(\theta, \gamma|\mathcal{M}_1) \arrow[dr] & q(\theta, \gamma|\mathcal{M}_2) \arrow[d] & \cdots & q(\theta, \gamma|\mathcal{M}_K) \arrow[dll] \\
& \boxed{\sum_k^K p(\mathcal{M}_k|y)(\cdot)} \arrow[d] & & \\
& q(\theta, \gamma) & &
\end{tikzcd}$$

Bayesian model averaging

$\mathcal{M}_k = \{v_k, c, \pi\}$

$$
\begin{tikzcd}
q(\theta, \gamma | \mathcal{M}_1) \arrow[dr] & q(\theta, \gamma | \mathcal{M}_2) \arrow[d] & \cdots & q(\theta, \gamma | \mathcal{M}_K) \arrow[dll] \\
& \sum_k^K p(\mathcal{M}_k | y)(\cdot) \arrow[d] & & \\
& q(\theta, \gamma) & &
\end{tikzcd}
$$

... but expensive and $p(\mathcal{M}_k | y)$ intractable

Speeding up the algorithm

Speeding up the algorithm

1. We zero-temperature restrict $q(\theta_j) = \delta_{\mu_j}(\theta_j)$
    - Exact $\nabla_\mu \mathcal{F}$ and posterior predictive distribution

Speeding up the algorithm

1. We zero-temperature restrict $q(\theta_j) = \delta_{\mu_j}(\theta_j)$
   - Exact $\nabla_\mu \mathcal{F}$ and posterior predictive distribution
2. We prune low PIP variables during training: $\mu_j \leftarrow \mu_j \mathbb{I}(\lambda_j > \epsilon) : \epsilon \in (0, 1)$
   - Reduces complexity from $\mathcal{O}(d)$ to $\mathcal{O}(q)$: $q = \{\#\lambda > \epsilon\}$

Speeding up the algorithm

1. We zero-temperature restrict $q(\theta_j) = \delta_{\mu_j}(\theta_j)$
   - Exact $\nabla_\mu \mathcal{F}$ and posterior predictive distribution
2. We prune low PIP variables during training: $\mu_j \leftarrow \mu_j \mathbb{I}(\lambda_j > \epsilon) : \epsilon \in (0, 1)$
   - Reduces complexity from $\mathcal{O}(d)$ to $\mathcal{O}(q)$: $q = \{\#\lambda > \epsilon\}$
   - Under certain conditions $\mu_j \to N_\delta(0)$ if $\lambda_j \leq \epsilon$ during a-CAVI

Using the leave-one-out predictive density to approximate posterior weights

$$\text{LOO-PD} = \prod_i p(y_i | y_{\neg i}, \mathcal{M}_k)$$

Using the leave-one-out predictive density to approximate posterior weights

$$\text{LOO-PD} = \prod_i p(y_i | y_{\neg i}, \mathcal{M}_k)$$

Under uniform prior $p(\mathcal{M}_k) \propto 1$:

$$p(\mathcal{M}_k | y) \propto p(y | \mathcal{M}_k) = \prod_i p(y_i | y_{<i}, \mathcal{M}_k) \approx \prod_i p(y_i | y_{\neg i}, \mathcal{M}_k)$$

Using the leave-one-out predictive density to approximate posterior weights

$$\text{LOO-PD} = \prod_i p(y_i | y_{\neg i}, \mathcal{M}_k)$$

Under uniform prior $p(\mathcal{M}_k) \propto 1$:

$$p(\mathcal{M}_k | y) \propto p(y | \mathcal{M}_k) = \prod_i p(y_i | y_{<i}, \mathcal{M}_k) \approx \prod_i p(y_i | y_{\neg i}, \mathcal{M}_k)$$

- Under ZT approximation:

$$\text{LOO-PD} = \prod_i \underbrace{p(y_i | y_{\neg i}, \theta = \mu_k)}_{\text{standard GPR posterior}}$$

$$\implies \mathcal{O}(n^3) \text{ using Bürkner et al. (2021)}$$

Nearest neighbour truncations for large-n scalability[7]

Marginal likelihood:

$$\log p(y|\theta) \approx \frac{n}{m} \log p(y_i, y_{NN(i)}|\theta)$$

Predictive distribution:

$$\log p(y_i|y_{\neg i}, \mathcal{M}_k) \approx logp(y_i|y_{NN(i)}, \mathcal{M}_k)$$

[7] Used previously in Chen et al. (2020); Jankowiak and Pleiss (2021).

Nearest neighbour truncations for large-n scalability[7]

Marginal likelihood:

$$\log p(y|\theta) \approx \frac{n}{m} \log p(y_i, y_{NN(i)}|\theta)$$

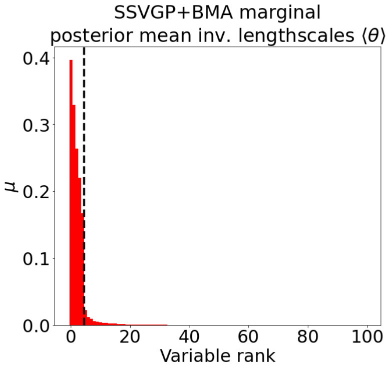
Predictive distribution:

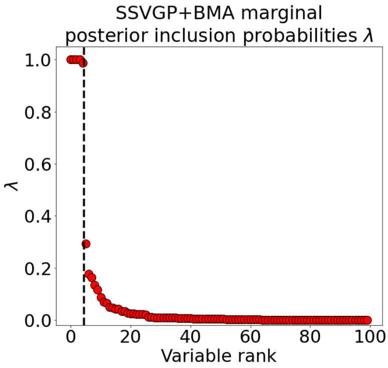
$$\log p(y_i|y_{\neg i}, \mathcal{M}_k) \approx logp(y_i|y_{NN(i)}, \mathcal{M}_k)$$

For m-nearest neighbours we get $\mathcal{O}(n^3) \to \mathcal{O}(n \log n + m^3)$

[7] Used previously in Chen et al. (2020); Jankowiak and Pleiss (2021).

Returning to the toy example





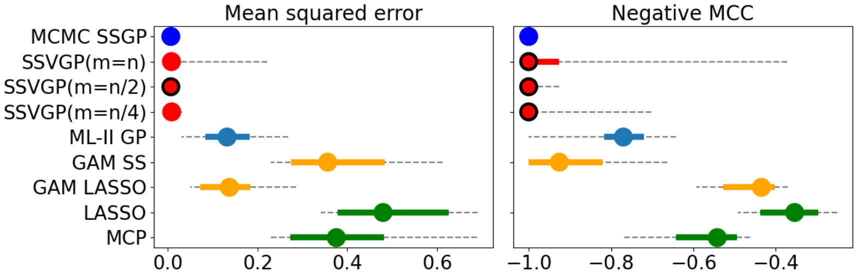
Method	MSE	Runtime[8] (s)
ML-II GP	0.096 ± 0.014	$\mathbf{5.0} \pm 0.7$
SSVGP original	0.068 ± 0.011	27.8 ± 0.5
SSVGP+BMA	$\mathbf{0.064} \pm 0.01$	14.6 ± 0.4

[8] nearest neighbour truncation with m=n/4 neighbours used for BMA

"Ground truth" simulation comparison

Savitsky et al. (2011) experiment:

- Draw $n = 100$ samples of $x \sim Unif[0,1]^{1000}$
- Set $y = x_1 + x_2 + x_3 + x_4 + sin(3x_5) + sin(5x_6) + \epsilon$ for $\epsilon \sim \mathcal{N}(0, 0.05^2)$.



Method:	Savitsky et al. (2011)	SSVGP(n)	SSVGP(n/2)	SSVGP(n/4)	ML-II
Runtime:	10224s	20.3s	11.0s	8.4s	3.9s

Large-scale dataset results

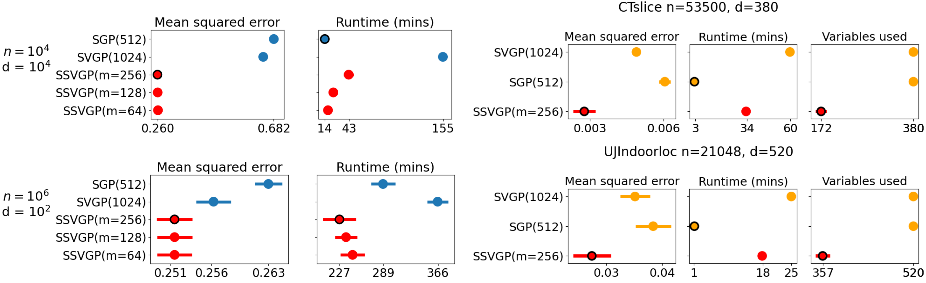


Figure: LHS: synthetic datasets, RHS: real datasets from UCI repository

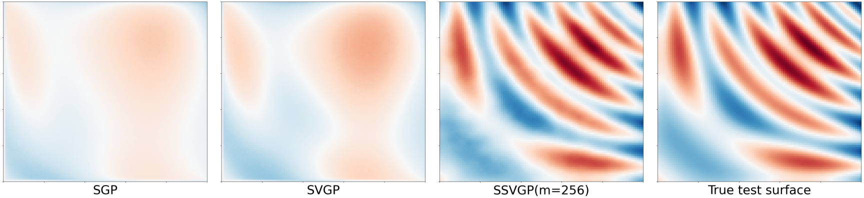


Figure: Synthetic experiment: average prediction surfaces for $n = d = 10^4$.

Summary

- Method for variable selection in Gaussian process regression using spike and slab priors

Summary

- Method for variable selection in Gaussian process regression using spike and slab priors
- (very) fast runtimes on high-dimensional datasets and $\mathcal{O}(n \log n)$ scalability

Summary

- Method for variable selection in Gaussian process regression using spike and slab priors
- (very) fast runtimes on high-dimensional datasets and $\mathcal{O}(n \log n)$ scalability
- BMA crucial for adaptive sparsity

Summary

- Method for variable selection in Gaussian process regression using spike and slab priors
- (very) fast runtimes on high-dimensional datasets and $\mathcal{O}(n \log n)$ scalability
- BMA crucial for adaptive sparsity
- Can compete with spike and slab MCMC but orders of magnitude faster

Summary

- Method for variable selection in Gaussian process regression using spike and slab priors
- (very) fast runtimes on high-dimensional datasets and $\mathcal{O}(n \log n)$ scalability
- BMA crucial for adaptive sparsity
- Can compete with spike and slab MCMC but orders of magnitude faster
- Consistently outperformed standard GPR and scalable approximations in high-dimensional, especially sparse settings.

Summary

- Method for variable selection in Gaussian process regression using spike and slab priors
- (very) fast runtimes on high-dimensional datasets and $\mathcal{O}(n \log n)$ scalability
- BMA crucial for adaptive sparsity
- Can compete with spike and slab MCMC but orders of magnitude faster
- Consistently outperformed standard GPR and scalable approximations in high-dimensional, especially sparse settings.

https://github.com/HWDance/SSVGP

Bürkner, P.-C., Gabry, J., and Vehtari, A. (2021). Efficient leave-one-out cross-validation for bayesian non-factorized normal and student-t models. *Computational Statistics*, 36(2):1243–1261.

Cawley, G. C. and Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, 11:2079–2107.

Chen, H., Zheng, L., Al Kontar, R., and Raskutti, G. (2020). Stochastic gradient descent in correlated settings: A study on gaussian processes. *Advances in Neural Information Processing Systems*, 33.

Dai, Z., Hensman, J., and Lawrence, N. (2015). Spike and slab gaussian process latent variable models. *arXiv preprint arXiv:1505.02434*.

Jankowiak, M. and Pleiss, G. (2021). Scalable cross validation losses for gaussian process models. *arXiv preprint arXiv:2105.11535*.

Linkletter, C., Bingham, D., Hengartner, N., Higdon, D., and Ye, K. Q. (2006). Variable selection for gaussian process models in computer experiments. *Technometrics*, 48(4):478–490.

MacKay, D. J. (1996). Bayesian methods for backpropagation networks. In *Models of neural networks III*, pages 211–254. Springer.

Mohammed, R. O. and Cawley, G. C. (2017). Over-fitting in model selection with gaussian process regression. In *International Conference on Machine Learning and Data Mining in Pattern Recognition*, pages 192–205. Springer.

Ober, S. W., Rasmussen, C. E., and van der Wilk, M. (2021). The promises and pitfalls of deep kernel learning. *arXiv preprint arXiv:2102.12108*.

Paananen, T., Piironen, J., Andersen, M. R., and Vehtari, A. (2019). Variable selection for gaussian processes via sensitivity analysis of the posterior predictive distribution. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1743–1752. PMLR.

Qamar, S. and Tokdar, S. T. (2014). Additive gaussian process regression. *arXiv preprint arXiv:1411.7009*.

Rasmussen, C. E. and Williams, C. K. (2006). Rasmussen and christopher ki williams. gaussian processes for machine learning. *MIT Press*, 211:212.

Savitsky, T., Vannucci, M., and Sha, N. (2011). Variable selection for nonparametric gaussian process priors: Models and computational strategies. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 26(1):130.