Sequential Knot Selection in Sparse Gaussian Processes

Jarad Niemi, Nate Garton, and Alicia Carriquiry

Iowa State University

January 10, 2020

Funded, in part, by

- the Center for Statistics and Applications in Forensic Evidence (CSAFE) and
- the Iowa State University Presidential Interdisciplinary Research Initiative on C-CHANGE: Science for a Changing Agriculture.

Outline

- Basics of Gaussian Processes (GPs)
- Sparse GPs using knots
- One-at-a-time (OAT) selection
- Applications

Non-parametric regression

Suppose we have the model

$$y_i = f(x_i) + \epsilon_i$$

where

- response $y_i \in \mathbb{R}$ (for simplicity)
- input $x_i \in \mathcal{X} \subset \mathbb{R}^d$
- noise $\epsilon_i \overset{ind}{\sim} N(0, \tau^2)$
- unknown $f : \mathcal{X} \to \mathbb{R}$

We observe pairs $\left(y_i, x_i^\top\right)$ for $i = 1, \ldots, N$ and we are interested in inference on the unknown $f(\cdot)$.

Gaussian Process

Assume a Gaussian Process (GP) prior for f:

$$f(x) \sim \mathcal{GP}\left(m(x), k_\theta(x, x')\right)$$

which assumes, for any finite subset,

$$f_x = \left[\begin{array}{c} f(x_1) \\ \vdots \\ f(x_M) \end{array}\right] \sim \mathcal{N}_M(m_x, \Sigma_{xx})$$

where $m_x = [m(x_1), \ldots, m(x_M)]^\top$ and

$$\Sigma_{xx}(i,j) = k_\theta(x_i, x_j)$$

for some kernel (covariance function) $k_\theta : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

[Rasmussen and Williams, 2006]

Kernel

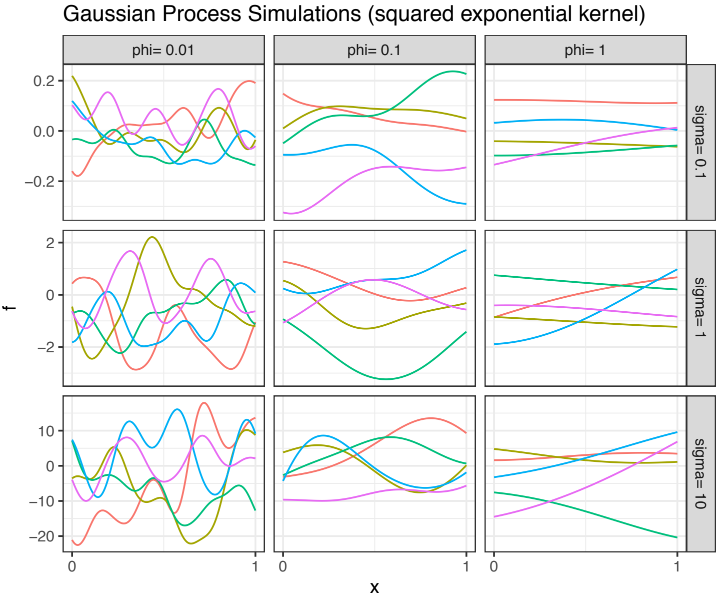
Kernel controls how smooth the process is both by determining:

- function differentiability and
- function wiggliness.

As an example, the squared exponential (or Gaussian) kernel is

$$k_\theta(x_i, x_j) = \sigma^2 \exp\left(-\frac{1}{2}\frac{(x_i - x_j)^\top (x_i - x_j)}{\phi}\right)$$

which provides infinitely differentiable GP realizations. The parameter σ^2 is the variance that controls the overall magnitude of the function and ϕ is the length-scale that controls how wiggly the function is.


Gaussian Process Simulations (squared exponential kernel)
phi= 0.01
phi= 0.1
phi= 1
sigma= 0.1
sigma= 1
sigma= 10
f
x
0.2
0.0
−0.2
2
0
−2
10
0
−10
−20
0
1
0
1
0
1

Training a GP

Find the maximum likelihood estimator (MLE) for $\theta = (\tau^2, \sigma^2, \phi)$,

$$\hat{\theta} = \mathsf{argmax}_\theta \, p(y|\theta) = \mathsf{argmax}_\theta \, N\left(y; m(x), \tau^2 \mathrm{I} + \Sigma(\theta)\right)$$

where $y = (y_1, \ldots, y_N)$. The log-likelihood is

$$\begin{aligned} \log \mathcal{N}(y; m(x), \tau^2 \mathrm{I} + \Sigma_\theta) = \quad & C \\ & -\tfrac{1}{2} \log |\tau^2 \mathrm{I} + \Sigma(\theta)| \\ & -\tfrac{1}{2} (y - m_x)^\top [\tau^2 \mathrm{I} + \Sigma(\theta)]^{-1} (y - m_x) \end{aligned}$$

Predicting from a GP

Function estimation (prediction) from a GP is based on the following joint distribution:

$$\begin{array}{c} y \\ f_{x^*} \end{array} \Bigg| \hat{\theta} \sim \left(\left[\begin{array}{c} m_x \\ m_{x^*} \end{array} \right], \left[\begin{array}{cc} \hat{\Sigma}_{xx} + \hat{\tau}^2 \mathrm{I} & \hat{\Sigma}_{xx^*} \\ \hat{\Sigma}_{x^*x} & \hat{\Sigma}_{x^*x^*} \end{array} \right] \right)$$

where

- $x^* = (x_1^*, \ldots, x_{N^*}^*)$ represents a set of prediction locations,
- $f_{x^*} = (f(x_1^*), \ldots, f(x_{N^*}^*))^\top$ represents a set of prediction values,
- $m_{x^*} = (m(x_1^*), \ldots, m(x_{N^*}^*))^\top$,
- $\Sigma_{x^*x^*}(i,j) = k_{\hat{\theta}}(x_i^*, x_j^*)$, and
- $\Sigma_{xx^*}(i,j) = k_{\hat{\theta}}(x_i, x_j^*)$.

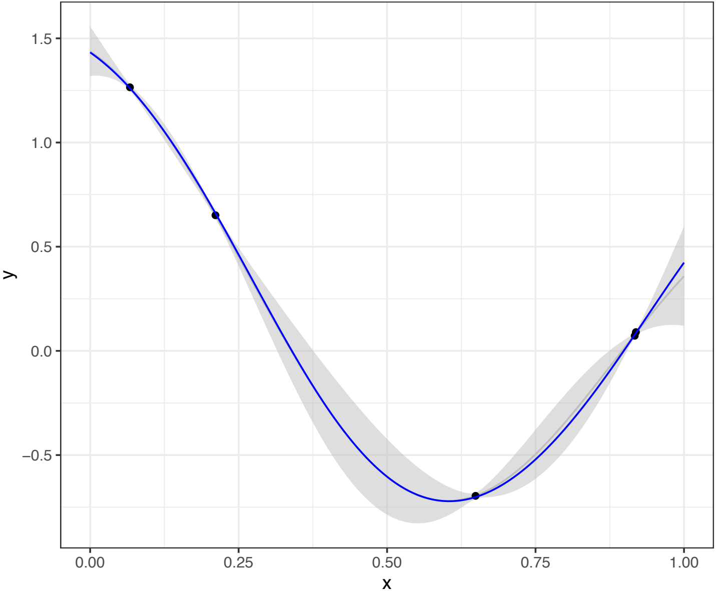
Thus, the desired conditional distribution is

$$f_{x^*} | y, \hat{\theta} \sim \mathcal{N}(\hat{m}_{x^*}, \hat{\Sigma}_{x^*x^*})$$

where

$$\begin{aligned} \hat{m}_{x^*} &= m_{x^*} + \hat{\Sigma}_{x^*x} \left[\hat{\tau}^2 \mathrm{I} + \hat{\Sigma}_{xx} \right]^{-1} (y - m_x) \\ \hat{\Sigma}_{x^*x^*} &= \hat{\Sigma}_{x^*x^*} - \hat{\Sigma}_{x^*x} \left[\hat{\tau}^2 \mathrm{I} + \hat{\Sigma}_{xx} \right]^{-1} \hat{\Sigma}_{xx^*}. \end{aligned}$$

Graphical representation



Training a GP - revisited

Find the maximum likelihood estimator (MLE) for $\theta = (\tau^2, \sigma^2, \phi)$,

$$\hat{\theta} = \text{argmax}_\theta\, p(y|\theta) = \text{argmax}_\theta\, N\left(y; m_x, \tau^2 \mathrm{I} + \Sigma(\theta)\right)$$

where $y = (y_1, \ldots, y_N)$. The log-likelihood is

$$\begin{aligned} \log \mathcal{N}(y; m_x, \tau^2 \mathrm{I} + \Sigma_\theta) = \quad & C \\ & -\tfrac{1}{2} \log |\tau^2 \mathrm{I} + \Sigma(\theta)| \\ & -\tfrac{1}{2} (y - m_x)^\top [\tau^2 \mathrm{I} + \Sigma(\theta)]^{-1} (y - m_x) \end{aligned}$$

If there are N observations, $\Sigma(\theta)$ is an $N \times N$ covariance matrix and thus the computational time scales as $\mathcal{O}(N^3)$.

This is doable if $N \approx 1,000$ but not when you start getting larger and larger data sets.

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$, such that

$$p(f_x, f_{x^\dagger}|\theta) = p(f_x|f_{x^\dagger}, \theta)p(f_{x^\dagger}|\theta).$$

where

$$\begin{aligned} f_x|f_{x^\dagger}, \theta &\sim \mathcal{N}\left(m_x + \Sigma_{xx^\dagger}\Sigma_{x^\dagger x^\dagger}^{-1}(f_{x^\dagger} - m_{x^\dagger}), \Lambda\right) \\ f_{x^\dagger}|\theta &\sim \mathcal{N}(m_{x^\dagger}, \Sigma_{x^\dagger x^\dagger}) \end{aligned}$$

with $\Lambda = \text{diag}\left(\Sigma_{xx} - \Sigma_{xx^\dagger}\Sigma_{x^\dagger x^\dagger}^{-1}\Sigma_{x^\dagger x}\right)$.

This joint implies the following marginal distribution for f_x:

$$f_x|\theta \sim \mathcal{N}(m_x, \Lambda + \Sigma_{xx^\dagger}\Sigma_{x^\dagger x^\dagger}^{-1}\Sigma_{x^\dagger x})$$

which has the correct marginal means and variances, but the covariances are controlled by the knots.

[Seeger et al., 2003, Quiñonero-Candela and Rasmussen, 2005, Snelson and Ghahramani, 2006, Banerjee et al., 2008, Finley et al., 2009, Titsias, 2009, Cao et al., 2013]

Train FIC Model

Let $\Psi_{xx} \equiv \Lambda(\theta) + \Sigma_{xx^\dagger}(\theta)\Sigma_{x^\dagger x^\dagger}(\theta)^{-1}\Sigma_{x^\dagger x}(\theta)$, then

$$Y|x^\dagger, \theta \sim \mathcal{N}(m_x, \tau^2\mathrm{I} + \Psi_{xx}).$$

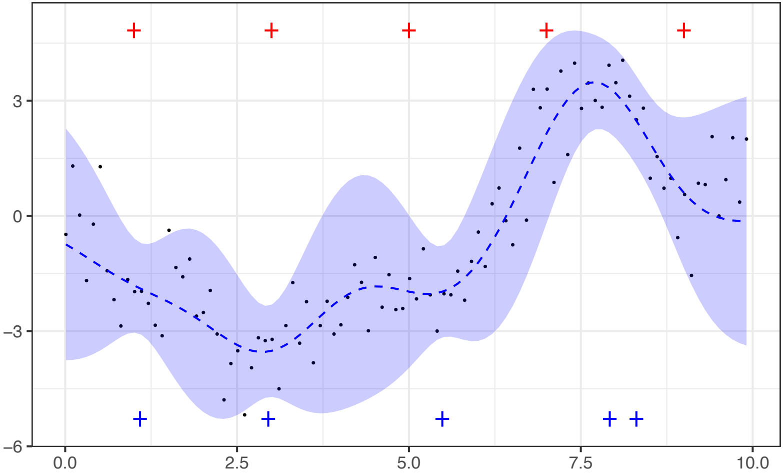
Train the model by finding

$$\hat{x}^\dagger, \hat{\theta} = \mathsf{argmax}_{x^\dagger,\theta}\mathcal{N}(y; m_x, \tau^2 I + \Psi_{xx}).$$

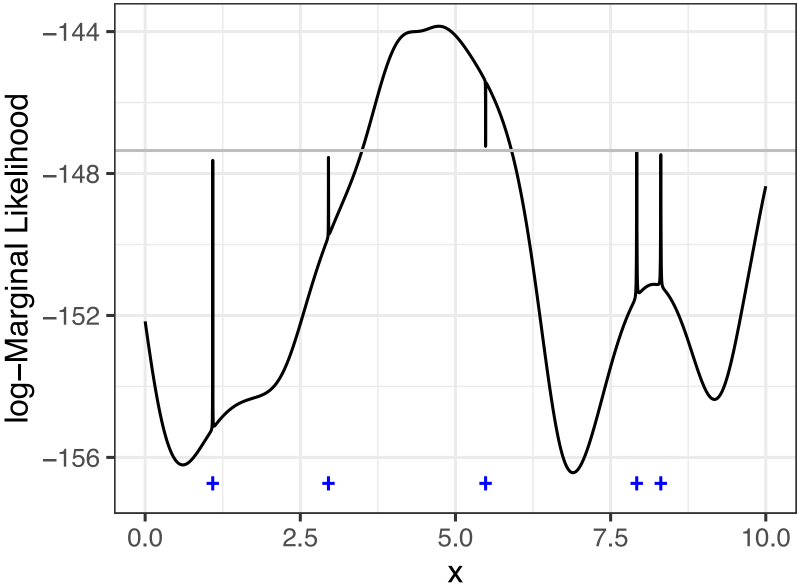
Appealing due to similarity with Full GP MLE approach, but there are a number of questions:

- how many knots are needed?
- where should we initialize the knots?
- when do we stop our iterative optimization algorithm?

Simultaneous knot optimization



Adding another knot



Knot selection algorithm

Algorithm 1. OAT knot selection algorithm. Convergence in the repeat loop is declared when the change in the objective function, the log-marginal likelihood, falls below a threshold. Set initial number of knots (K_I).

```
**Initialize:** $x^\dagger = \{x_i^\dagger\}_{i=1}^{K_I}$ ;
$\hat\theta = \text{argmax}_\theta p(y|x, x^\dagger, \theta)$ ;
**repeat**
    propose new knot $x^{\dagger^*} \leftarrow J(y, x, x^\dagger, \hat\theta)$ ;
    $(\hat{x}^{\dagger^*}, \hat\theta) = \text{argmax}_{(x^{\dagger^*}, \theta)} p(y|x, \{x^\dagger, x^{\dagger^*}\}, \theta)$ ;
    $x^\dagger = \{x^\dagger, \hat{x}^{\dagger^*}\}$ ;
**until** $|x^\dagger| = K_{max}$ *or convergence*;
```

Bayesian optimization

Let

- $w_{1:t-1}$ be the vector of log-marginal likelihood values at the candidates for the knot proposal which have thus far been explored at time t
- $w^+ = \max(w_{1:t-1})$

Let $W(z)$ be the unknown marginal likelihood at input location z, then expected improvement is

$$\begin{aligned}\alpha\left(z; w_{1:t-1}, \left\{x_1^\dagger, \ldots, x_{t-1}^\dagger\right\}\right) = \quad & \left(E\left[W(z)|w_{1:t-1}\right] - w^+\right)\Phi\left(\frac{E\left[W(z)|w_{1:t-1}\right]-w^+}{\sqrt{V\left[W(z)|w_{1:t-1}\right]}}\right) \\ & + \sqrt{V\left[W(z)|w_{1:t-1}\right]}\phi\left(\frac{E\left[W(z)|w_{1:t-1}\right]-w^+}{\sqrt{V\left[W(z)|w_{1:t-1}\right]}}\right).\end{aligned}$$

where ϕ and Φ are the pdf and cdf of a standard normal, respectively.

We will model the unknown marginal likelihood $W(z)$ using a meta GP.

[Jones, 2001, Shahriari et al., 2016]

Knot proposal algorithm

Algorithm 2. Knot proposal algorithm. Set the minimum (T_{min}) and maximum (T_{max}) number of marginal likelihood evaluations.

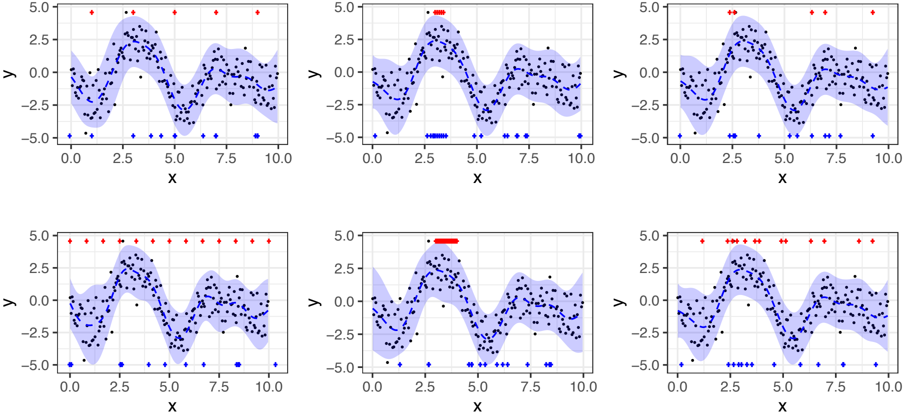
```
set the mean of the meta GP equal to $\log p\left(y \middle| x, \left\{x^\dagger, \cdot\right\}, \hat{\theta}\right)$ ;
sample $x_1^\dagger, ..., x_{T_{min}}^\dagger$ without replacement from $x$ ;
augment known marginal likelihood values $w_j = \log p\left(y \middle| x, \left\{x^\dagger, x_j^\dagger\right\}, \hat{\theta}\right)$ for
  $j = 1, \ldots, k$ with evaluations of the marginal likelihood at the new knots, that is
  $w_{k+j} = \log p\left(y \middle| x, \left\{x^\dagger, x_j^\dagger\right\}, \hat{\theta}\right)$ for $j = 1, ..., T_{min}$ ;
for $t = T_{min} + 1, ..., T_{max}$ do
    update covariance parameters in meta GP ;
    $x_t^* = \text{argmax}_{z \in x \setminus \{x_l^\dagger\}_{l=1}^{t-1}} \alpha\left(z; w, \left\{x_1^\dagger, \ldots, x_{t-1}^\dagger\right\}\right)$ ;
    $w_t = \log p\left(y \middle| x, \left\{x^\dagger, x_t^*\right\}, \hat{\theta}\right)$ ;
end
return $x_j^*$ such that $j = \text{argmax}_t\, w_t$
```

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm. Convergence in the repeat loop is declared when the change in the objective function, the log-marginal likelihood, falls below a threshold. Set initial number of knots (K_I).

```
**Initialize:** $x^\dagger = \{x_i^\dagger\}_{i=1}^{K_I}$ ;
$\hat\theta = \text{argmax}_\theta p(y|x, x^\dagger, \theta)$ ;
**repeat**
    propose new knot $x^{\dagger^*} \leftarrow J(y, x, x^\dagger, \hat\theta)$ ;
    $(\hat{x}^{\dagger^*}, \hat\theta) = \text{argmax}_{(x^{\dagger^*}, \theta)} p(y|x, \{x^\dagger, x^{\dagger^*}\}, \theta)$ ;
    $x^\dagger = \{x^\dagger, \hat{x}^{\dagger^*}\}$ ;
**until** $|x^\dagger| = K_{max}$ *or convergence*;
```

One-D Gaussian data



Starting locations: evenly spaced (left), adversarial (middle), random (right)
Algorithm: OAT (top) and simultaneous (bottom)

Computational results

Method	Initialization	K	RMSE	Runtime	GA Steps	log-Likelihood
Full GP	–	–	0.192	–	–	-311.720
OAT	Uniform	13	0.180	50	464	-308.120
OAT	Adversarial	22	0.228	96	669	-308.587
OAT	Random	13	0.228	50	470	-308.225
Simult.	Uniform	13	0.220	140	212	-306.852
Simult.	Adversarial	22	0.196	700	529	-308.398
Simult.	Random	13	0.247	88	140	-308.071

Performance metrics

All data models:

$$MNLP = \mathsf{median}_{i \in 1,\ldots,N_{test}} \{-\log p(\tilde{y}_i | x^\dagger, \hat{\theta}, y)\}.$$

$$AUKL = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \int p_{full}(f(\tilde{x}_i)|\hat{\theta}, y) \log \frac{p_{full}(f(\tilde{x}_i)|\hat{\theta}, y)}{p_{sparse}(f(\tilde{x}_i)|x^\dagger, \hat{\theta}, y)} df(\tilde{x}_i).$$

Gaussian:

$$SRMSE = \sigma_{\tilde{y}}^{-1} \sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (E[f(\tilde{x}_i)|Y] - \tilde{y}_i)^2},$$

where $\sigma_{\tilde{y}}^2 = \frac{1}{N_{test}-1} \sum_{i=1}^{N_{test}} (\tilde{y}_i - \underline{\tilde{y}})^2$, $\underline{\tilde{y}} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \tilde{y}_i$, and $\tilde{y}$ is the vector of test set target values.

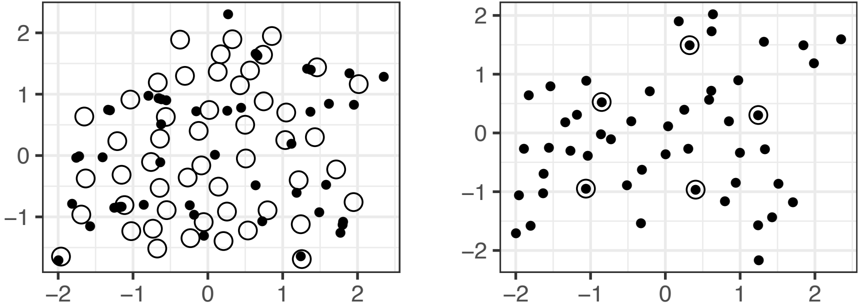
Boston Housing

490 observations (random 80% for training) with $d = 3$

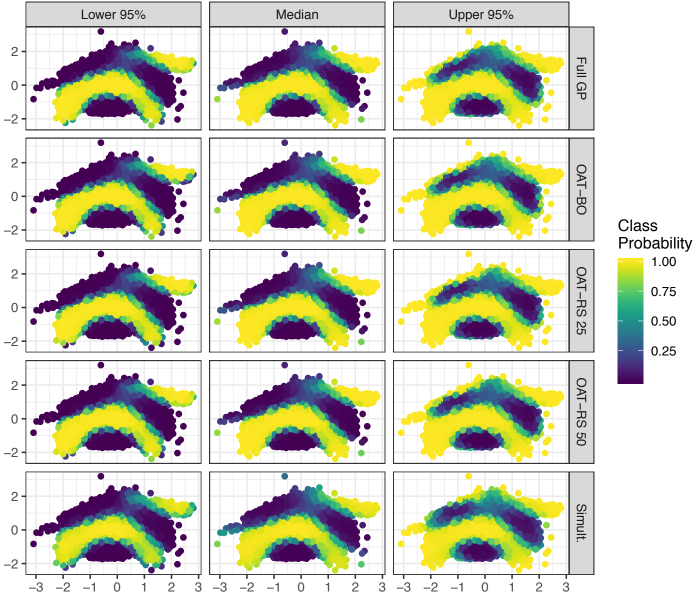
Method	Runtime	K	K/Tmax	SRMSE	MNLP	AUKL
Full	394	–	–	0.359	2.500	0.000
OAT-BO	545	13	25	0.366	2.466	0.045
OAT-RS	356	12	25	0.366	2.464	0.039
OAT-RS	339	15	50	0.364	2.469	0.047
Simult.	25831	50	–	0.378	2.291	0.356
Simult.	3945	13	–	0.356	2.313	0.242

Banana data

Binary lassification: 5300 observations (random 10% as training) with $d = 3$



Locations of initialized and estimated knots for the simultaneously optimized model with 50 knots (left) and for the OAT-BO model (right). Open circles are initial knots and solid points are estimated knots.


Lower 95%
Median
Upper 95%
Full GP
OAT–BO
OAT–RS 25
OAT–RS 50
Simult.
2
0
−2
−3
−2
−1
0
1
2
3
Class
Probability
1.00
0.75
0.50
0.25

Method	Runtime	Tmax	K	MNLP	AUKL
Full	26795	–	–	0.038	0.000
OAT-BO	3150	25	50	0.037	0.061
OAT-RS	2954	25	50	0.038	0.051
OAT-RS	3471	50	50	0.038	0.039
Simult.	6219	–	50	0.069	3.265

Summary

One-at-a-time (OAT) knot selection

- Similar predictive performance to simultaneous knot selection
- Better represents full GP compared to simultaneous knot selection
- Better computational efficiency than simultaneous knot selection

This slides are available

- `https://github.com/jarad/SFU2020`
- `http://www.jarad.me/research/presentations.html`

Thank you!

Other links:

- http://www.jaradniemi.com/
- https://www.youtube.com/jaradniemi
- https://twitter.com/jaradniemi

Reference

Sudipto Banerjee, Alan E. Gelfand, Andrew O. Finley, and Huiyan Sang. Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70 (4):825–848, 2008.

Yanshuai Cao, Marcus A Brubaker, David J Fleet, and Aaron Hertzmann. Efficient optimization for sparse gaussian process regression. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2013.

Andrew O. Finley, Huiyan Sang, Sudipto Banerjee, and Alan E. Gelfand. Improving the performance of predictive process modeling for large datasets. *Computational Statistics & Data Analysis*, 53(8):2873–2884, June 2009. doi: https://doi.org/10.1016/j.csda.2008.09.008.

Donald R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21(4):345–383, 2001.

Joaquin Quiñonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 2005.

Carl Edward Rasmussen and Christopher K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Matthias Seeger, Christopher Williams, and Neil Lawrence. Fast forward selection to speed up sparse gaussian process regression. In *Artificial Intelligence and Statistics*, 2003.

B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

Edward Snelson and Zoubin Ghahramani. Sparse gaussian processes using pseudo-inputs. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1257–1264. MIT Press, 2006. URL http://papers.nips.cc/paper/2857-sparse-gaussian-processes-using-pseudo-inputs.pdf.

Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Artificial Intelligence and Statistics*, pages 567–574, 2009.