Emulation of Agricultural Production Systems sIMulator (APSIM)

Jarad Niemi and Luis Damiano

Iowa State University

September 23, 2020

Emulation of Agricultural Production Systems sIMulator (APSIM)

Jarad Niemi and Luis Damiano

Iowa State University

September 23, 2020

Funded, in part, by

- the Iowa State University Presidential Interdisciplinary Research Initiative on C-CHANGE: Science for a Changing Agriculture
- USDA NIFA CAP: Consortium for Cultivating Human And Naturally reGenerative Enterprises (C-CHANGE)
- Foundation for Food and Agriculture Research: Prairie Strips for Healthy Soils and Thriving Farms

C-CHANGE: Science for a changing agriculture

http://agchange.org

Iowa Agricultural Production

https://www.iadg.com/iowa-advantages/target-industries/

> *Iowa is the largest producer of corn, pork and eggs in the United States and second in soybeans and red meat production.*





Loss of prairie

85% of Iowa was covered with prairie;

Loss of prairie

85% of Iowa was covered with prairie; today less than 0.1% remains

https://www.facebook.com/NealSmithNWR/photos/a.363116150433606/3093501587395035

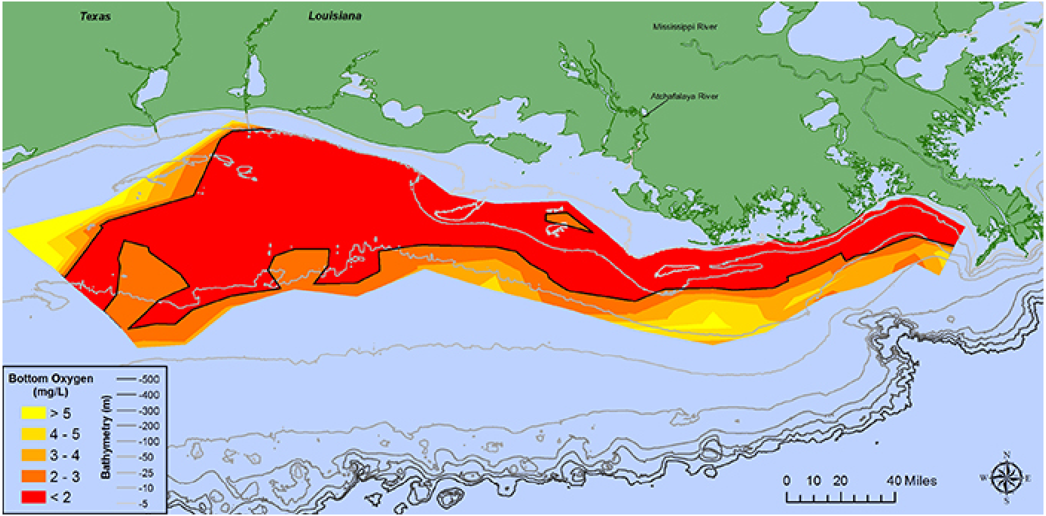
https://dissolve.com/stock-photo/field-young-soybean-plants-showing-corn-stalks-residue-royalty-free-image/101-D869-14-341

Soil loss

Iowa loses $1,000,000,000/year in soil

https://www.desmoinesregister.com/story/money/agriculture/2014/05/03/erosion-estimated-cost-iowa-billion-yield/8682651/

Gulf of Mexico Dead Zone



https://www.noaa.gov/media-release/gulf-of-mexico-dead-zone-is-largest-ever-measured

Des Moines Water Works Lawsuit

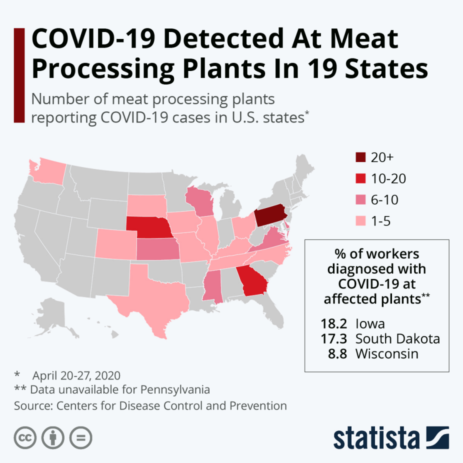
https://www.lwvumrr.org/blog/des-moines-water-works-lawsuit-update

Tedesco Environmental Learning Corridor

Manure lagoons

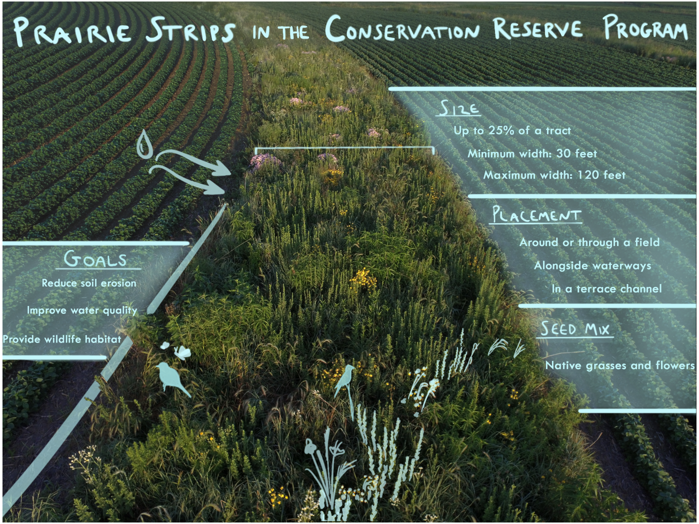
https://www.npr.org/2018/09/22/650698240/hurricane-s-aftermath-floods-hog-lagoons-in-north-carolina

COVID-19 in Meat Packing Plants



https://www.statista.com/chart/21585/meat-processing-plants-reporting-coronavirus-cases/

Prairie STRIPS



http://prairiestrips.org

Prairie STRIPS

PNAS Proceedings of the National Academy of Sciences of the United States of America

Keyword, Author, or D

Home Articles Front Matter News Podcasts Authors

NEW RESEARCH IN Physical Sciences Social Sciences

RESEARCH ARTICLE

Prairie strips improve biodiversity and the delivery of multiple ecosystem services from corn–soybean croplands

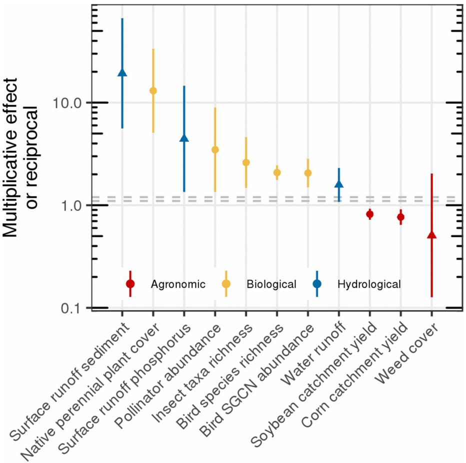
Lisa A. Schulte, Jarad Niemi, Matthew J. Helmers, Matt Liebman, J. Gordon Arbuckle, David E. James, Randall K. Kolka, Matthew E. O'Neal, Mark D. Tomer, John C. Tyndall, Heidi Asbjornsen, Pauline Drobney, Jeri Neal, Gary Van Ryswyk, and Chris Witte

PNAS October 17, 2017 114 (42) 11247-11252; first published October 2, 2017 https://doi.org/10.1073/pnas.1620229114

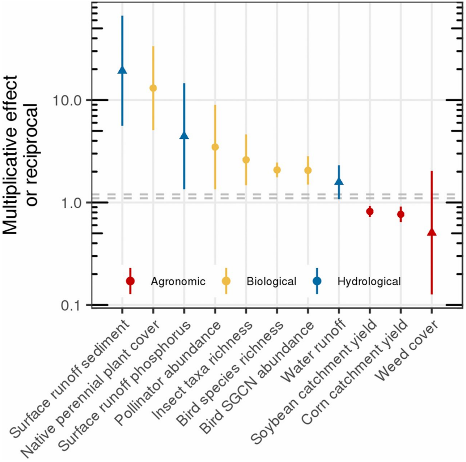
Edited by David Tilman, University of Minnesota, St. Paul, MN, and approved August 1, 2017 (received for review December 9, 2016)

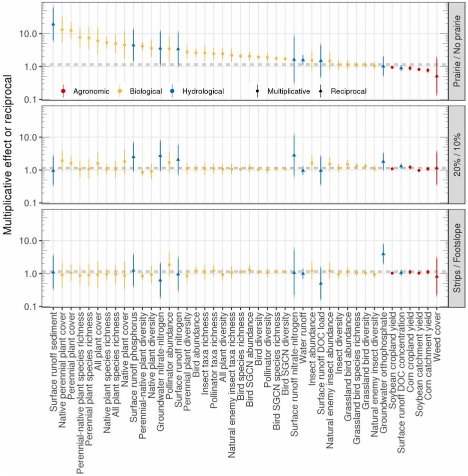
`https://www.pnas.org/content/114/42/11247.short`

Prairie STRIPS results



Prairie STRIPS results





https://www.pnas.org/content/114/42/11247/tab-figures-data

USDA NIFA - Biogas production from manure and herbaceous biomass

NIFA PRESS RELEASE

NATIONAL INSTITUTE OF FOOD AND AGRICULTURE

USDA Scientific Research Program Funds Sustainable Agricultural Systems Projects

USDA NIFA - Biogas production from manure and herbaceous biomass

NIFA PRESS RELEASE

USDA

NATIONAL INSTITUTE OF FOOD AND AGRICULTURE

USDA Scientific Research Program Funds Sustainable Agricultural Systems Projects

Feedstock Production | Anaerobic Digestion (AD) | Product Processing | Distribution and Markets | Society

Fig. 1. The agricultural value chain developed through C-CHANGE.

Computer models

- Water Erosion Prediction Project (WEPP)

Computer models

- Water Erosion Prediction Project (WEPP)
  - Daily Erosion Project (DEP)

Computer models

- Water Erosion Prediction Project (WEPP)
  - Daily Erosion Project (DEP)
- Agricultural Production Systems sIMulator (APSIM)

Computer models

- Water Erosion Prediction Project (WEPP)
  - Daily Erosion Project (DEP)
- Agricultural Production Systems sIMulator (APSIM)
  - Foresite

Computer models

- Water Erosion Prediction Project (WEPP)
  - Daily Erosion Project (DEP)
- Agricultural Production Systems sIMulator (APSIM)
  - Foresite
- Cycles

Computer models

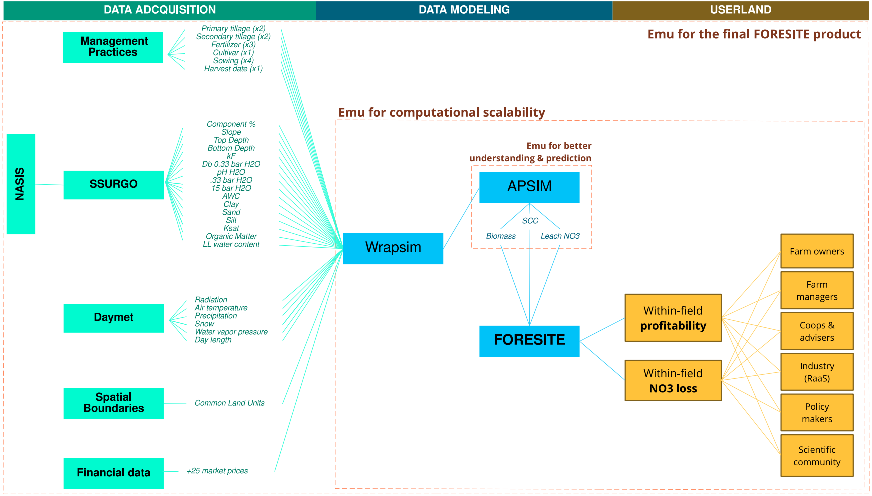
- Water Erosion Prediction Project (WEPP)
  - Daily Erosion Project (DEP)
- Agricultural Production Systems sIMulator (APSIM)
  - Foresite
- Cycles
- Iowa Biogas Assessment Model (IBAM)

Computer models

- Water Erosion Prediction Project (WEPP)
  - Daily Erosion Project (DEP)
- Agricultural Production Systems sIMulator (APSIM)
  - Foresite
- Cycles
- Iowa Biogas Assessment Model (IBAM)
- others...

APSIM

Role in the Foresite project

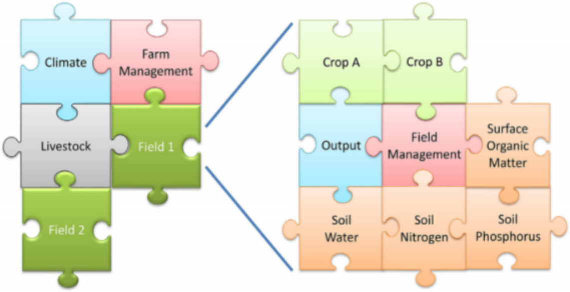


APSIM

Overview

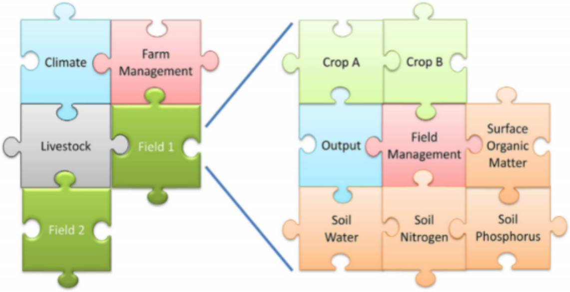
APSIM

Overview



APSIM

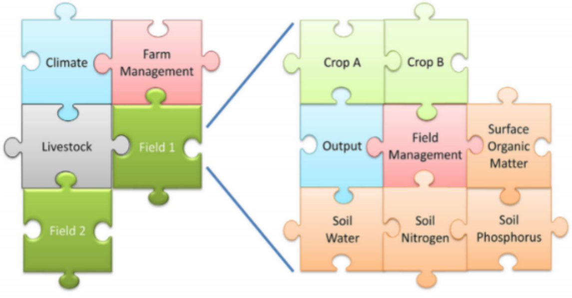
Overview



- Physical process based

APSIM

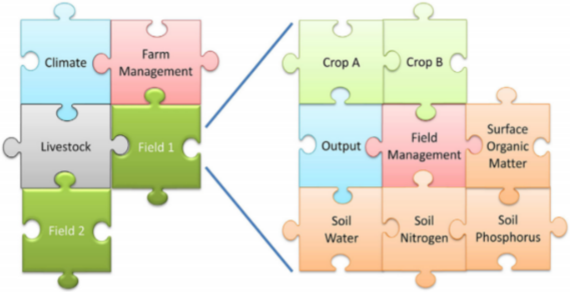
Overview



- Physical process based
- Peer-reviewed

APSIM

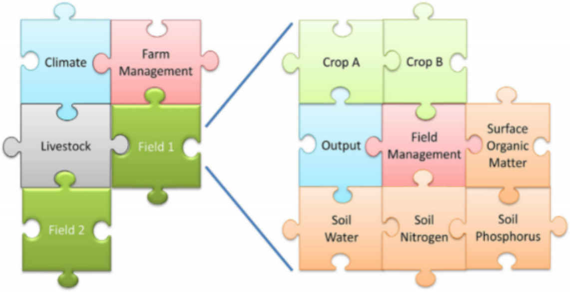
Overview



- Physical process based
- Peer-reviewed
- Flexible

APSIM

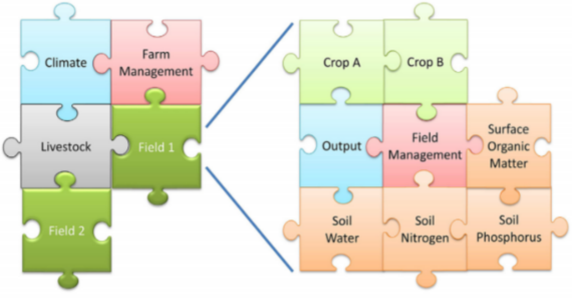
Overview



- Physical process based
- Peer-reviewed
- Flexible
- Calibrated for many different climates and countries

APSIM

Overview



- Physical process based
- Peer-reviewed
- Flexible
- Calibrated for many different climates and countries
- Dedicated, funded team of software engineers working to improve it

APSIM

Emulation goals

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.
- Model calibration: APSIM assumes a flat, uniform soil. TWI.

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.
- Model calibration: APSIM assumes a flat, uniform soil. TWI.
- Identify deficiencies: does APSIM emphasize on soil too much?

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.
- Model calibration: APSIM assumes a flat, uniform soil. TWI.
- Identify deficiencies: does APSIM emphasize on soil too much?
- Uncertainty quantification: APSIM is deterministic, real life is not.

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.
- Model calibration: APSIM assumes a flat, uniform soil. TWI.
- Identify deficiencies: does APSIM emphasize on soil too much?
- Uncertainty quantification: APSIM is deterministic, real life is not.
- Inform the development & calibration of new crop modules (e.g. cover crops).

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.
- Model calibration: APSIM assumes a flat, uniform soil. TWI.
- Identify deficiencies: does APSIM emphasize on soil too much?
- Uncertainty quantification: APSIM is deterministic, real life is not.
- Inform the development & calibration of new crop modules (e.g. cover crops).
- Portability: web interface back end for on-line prediction.

APSIM

Emulation goals

- Variable selection: to guide data collection by field scientists.
- Model calibration: APSIM assumes a flat, uniform soil. TWI.
- Identify deficiencies: does APSIM emphasize on soil too much?
- Uncertainty quantification: APSIM is deterministic, real life is not.
- Inform the development & calibration of new crop modules (e.g. cover crops).
- Portability: web interface back end for on-line prediction.
- Assist scientists in studying agronomical hypothesis.

APSIM
Input space

- 2,160 run-specific input values

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.
  - Climate dynamics 4 functionals x 365 daily values, 2 scalars.

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.
  - Climate dynamics 4 functionals x 365 daily values, 2 scalars.
  - Land management practices, mostly categorical scalars.

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.
  - Climate dynamics 4 functionals x 365 daily values, 2 scalars.
  - Land management practices, mostly categorical scalars.

- The input space is...

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.
  - Climate dynamics 4 functionals x 365 daily values, 2 scalars.
  - Land management practices, mostly categorical scalars.

- The input space is...
  - High dimensional: computational and modeling challenging.

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.
  - Climate dynamics 4 functionals x 365 daily values, 2 scalars.
  - Land management practices, mostly categorical scalars.

- The input space is...
  - High dimensional: computational and modeling challenging.
  - Structured: complex to capture hierarchy.

APSIM

Input space

- 2,160 run-specific input values
  - Soil properties 22 functionals x 16 layers, 2 scalars.
  - Climate dynamics 4 functionals x 365 daily values, 2 scalars.
  - Land management practices, mostly categorical scalars.

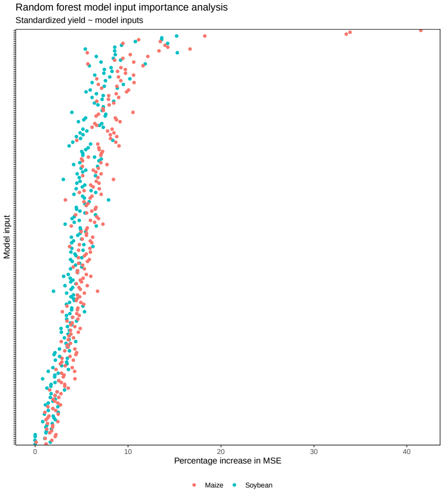
- The input space is...
  - High dimensional: computational and modeling challenging.
  - Structured: complex to capture hierarchy.
  - Vast: large number of runs to explore it.

APSIM

Exploratory Analyses

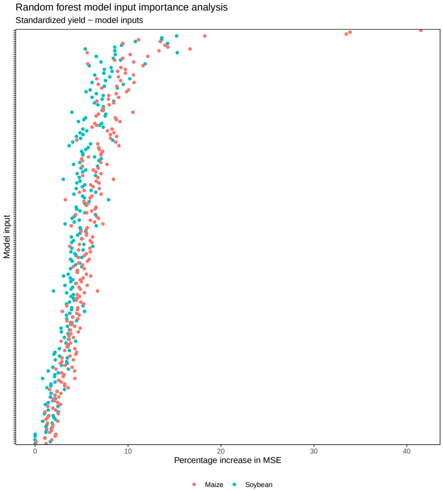
APSIM

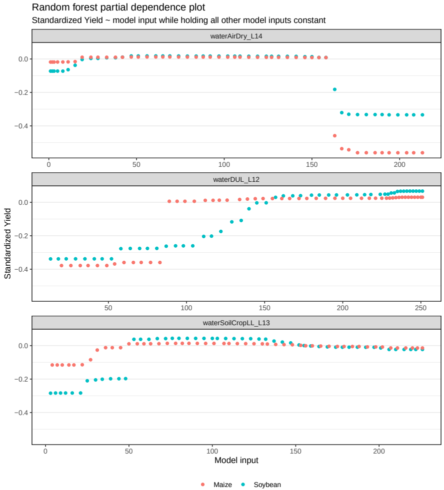
Exploratory Analyses



APSIM

Exploratory Analyses





One-at-a-time knot selection

Training a GP

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

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

One-at-a-time knot selection

Training a GP

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

One-at-a-time knot selection

Training a GP

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.

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$ with $K << N$,

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

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

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

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

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

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

where

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

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

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

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

where

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

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$ with $K << N$,
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 $\Lambda = \text{diag}\left(\Sigma_{xx} - \Sigma_{xx^\dagger}\Sigma_{x^\dagger x^\dagger}^{-1}\Sigma_{x^\dagger x}\right)$.

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$ with $K << N$,
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:

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

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$ with $K << N$,
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,

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$ with $K << N$,
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.

One-at-a-time knot selection

Fully Independent Conditional (FIC) Approximation

Introduce a set of knots $x^\dagger = \left\{x_1^\dagger, \ldots, x_K^\dagger\right\}$ with $K << N$, 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]

One-at-a-time knot selection

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm.

One-at-a-time knot selection

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm.

Initialize: $x^{\dagger} = \{x_i^{\dagger}\}_{i=1}^{K_I}$

One-at-a-time knot selection

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm.

Initialize: $x^\dagger = \{x_i^\dagger\}_{i=1}^{K_I}$;

$\hat{\theta} = \text{argmax}_\theta p(y|x, x^\dagger, \theta)$

One-at-a-time knot selection

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm.

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 $x^{\dagger^*} \leftarrow J(y, x, x^{\dagger}, \hat{\theta})$

One-at-a-time knot selection

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm.

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 $x^{\dagger^*} \leftarrow J(y, x, x^{\dagger}, \hat{\theta})$;

  $(\hat{x}^{\dagger^*}, \hat{\theta}) = \text{argmax}_{(x^{\dagger^*}, \theta)} p(y|x, \{x^{\dagger}, x^{\dagger^*}\}, \theta)$

One-at-a-time knot selection

Knot selection algorithm

Algorithm 1. OAT knot selection algorithm.

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*;

[Garton et al., Garton]

One-at-a-time knot selection

Knot proposal algorithm

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

One-at-a-time knot selection

Knot proposal algorithm

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

---

1 set the mean of the meta GP equal to $\log p\left(y \middle| x, \left\{x^{\dagger}, \cdot\right\}, \hat{\theta}\right)$

One-at-a-time knot selection

Knot proposal algorithm

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

---

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

One-at-a-time knot selection

Knot proposal algorithm

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

---

set the mean of the meta GP equal to $\log p\left(y \middle| x, \left\{x^\dagger, \cdot\right\}, \hat{\theta}\right)$;

sample $x_1^\dagger, \ldots, x_{T_{min}}^\dagger$ 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 $j = 1, \ldots, T_{min}$

One-at-a-time knot selection

Knot proposal algorithm

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

---

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

One-at-a-time knot selection

Knot proposal algorithm

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

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 ;

  $x_t^* = \text{argmax}_{z \in x \setminus \{x_l^\dagger\}_{l=1}^{t-1}} \alpha \left(z; w, \left\{ x_1^\dagger, \ldots, x_{t-1}^\dagger \right\} \right)$

One-at-a-time knot selection

Knot proposal algorithm

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

```
set the mean of the meta GP equal to log p(y | x, {x†, ·}, θ̂) ;
sample x†_1, ..., x†_{T_min} without replacement from x ;
augment known marginal likelihood values w_j = log p(y | x, {x†, x†_j}, θ̂) for j = 1, ..., k with evaluations
  of the marginal likelihood at the new knots, that is w_{k+j} = log p(y | x, {x†, x†_j}, θ̂) for j = 1, ..., T_min ;
for t = T_min + 1, ..., T_max do
    update covariance parameters in meta GP ;
    x*_t = argmax_{z ∈ x \ {x†_l}_{l=1}^{t-1}} α(z; w, {x†_1, ..., x†_{t-1}}) ;
    w_t = log p(y | x, {x†, x*_t}, θ̂)
```

set the mean of the meta GP equal to $\log p\left(y \,\middle|\, x, \left\{x^\dagger, \cdot\right\}, \hat\theta\right)$;

sample $x^\dagger_1, ..., x^\dagger_{T_{min}}$ without replacement from x ;

augment known marginal likelihood values $w_j = \log p\left(y \,\middle|\, x, \left\{x^\dagger, x^\dagger_j\right\}, \hat\theta\right)$ for $j = 1, \ldots, k$ with evaluations of the marginal likelihood at the new knots, that is $w_{k+j} = \log p\left(y \,\middle|\, x, \left\{x^\dagger, x^\dagger_j\right\}, \hat\theta\right)$ for $j = 1, ..., T_{min}$;

for $t = T_{min} + 1, ..., T_{max}$ **do**

- update covariance parameters in meta GP ;
- $x^*_t = \text{argmax}_{z \in x \setminus \{x^\dagger_l\}_{l=1}^{t-1}} \alpha\left(z; w, \left\{x^\dagger_1, \ldots, x^\dagger_{t-1}\right\}\right)$;
- $w_t = \log p\left(y \,\middle|\, x, \left\{x^\dagger, x^*_t\right\}, \hat\theta\right)$

One-at-a-time knot selection

Knot proposal algorithm

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

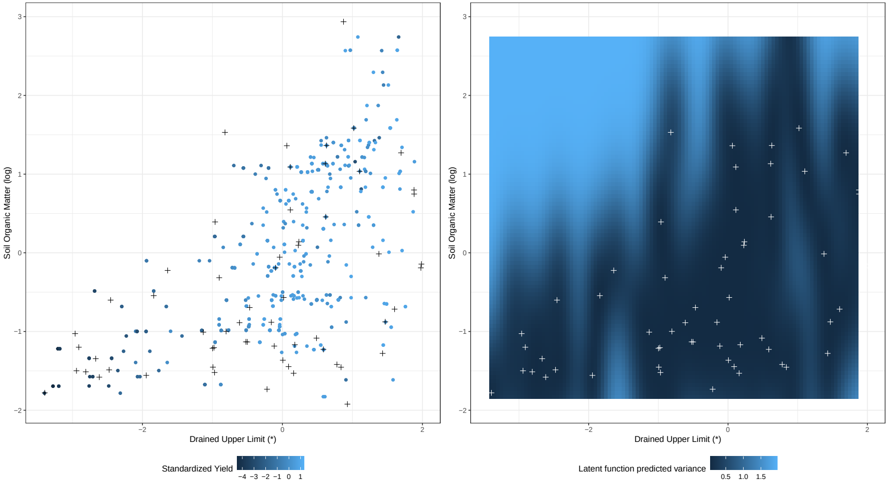
---

set the mean of the meta GP equal to $\log p\left(y \middle| x, \left\{x^\dagger, \cdot\right\}, \hat{\theta}\right)$;
sample $x^\dagger_1, ..., x^\dagger_{T_{min}}$ without replacement from x ;
augment known marginal likelihood values $w_j = \log p\left(y \middle| x, \left\{x^\dagger, x^\dagger_j\right\}, \hat{\theta}\right)$ for $j = 1, \ldots, k$ with evaluations of the marginal likelihood at the new knots, that is $w_{k+j} = \log p\left(y \middle| x, \left\{x^\dagger, x^\dagger_j\right\}, \hat{\theta}\right)$ for $j = 1, ..., T_{min}$;
for $t = T_{min} + 1, ..., T_{max}$ **do**
    update covariance parameters in meta GP ;
    $x^*_t = \text{argmax}_{z \in x \setminus \{x^\dagger_l\}_{l=1}^{t-1}} \alpha\left(z; w, \left\{x^\dagger_1, \ldots, x^\dagger_{t-1}\right\}\right)$;
    $w_t = \log p\left(y \middle| x, \left\{x^\dagger, x^*_t\right\}, \hat{\theta}\right)$;
end
return x^*_j such that $j = \text{argmax}_t\, w_t$

[Garton et al., Garton]

OAT Knot selection

Visualization



(*) Content of water retained after gravitational flow.

Future Efforts

Extending Morris' correlation distance for unknown weight function

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$
- Assume $Y = f(\mathbf{X})$

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$
- Assume $Y = f(\mathbf{X})$
- Assume $Y \sim GP(\mu, \sigma^2)$, $\mu \in \mathbb{R}$ and $\sigma^2 > 0$

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$
- Assume $Y = f(\mathbf{X})$
- Assume $Y \sim GP(\mu, \sigma^2)$, $\mu \in \mathbb{R}$ and $\sigma^2 > 0$
- Recall the SE Kernel with ARD

$$
\begin{aligned}
k(\mathbf{X_i}, \mathbf{X_j}) &= \sigma^2_{ard} e^{-\frac{1}{2} D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w})} \\
D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w}) &= \sum_{d=1}^{D} w_d (X_{i,d} - X_{j,d})^2
\end{aligned}
$$

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$
- Assume $Y = f(\mathbf{X})$
- Assume $Y \sim GP(\mu, \sigma^2)$, $\mu \in \mathbb{R}$ and $\sigma^2 > 0$
- Recall the SE Kernel with ARD

$$k(\mathbf{X_i}, \mathbf{X_j}) = \sigma^2_{ard} e^{-\frac{1}{2} D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w})}$$

$$D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w}) = \sum_{d=1}^{D} w_d (X_{i,d} - X_{j,d})^2$$

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$
- Assume $Y = f(\mathbf{X})$
- Assume $Y \sim GP(\mu, \sigma^2)$, $\mu \in \mathbb{R}$ and $\sigma^2 > 0$
- Recall the SE Kernel with ARD

$$k(\mathbf{X_i}, \mathbf{X_j}) = \sigma^2_{ard} e^{-\frac{1}{2} D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w})}$$

$$D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w}) = \sum_{d=1}^{D} w_d (X_{i,d} - X_{j,d})^2$$

For a continuous index $t \in [0, T], T \in \mathbb{R}^+$

$$D(\cdot) = \int_0^t w(t-s)(X_i(s) - X_j(s))^2 ds$$

$$w : \mathbb{R}^+ \to \mathbb{R}^+$$

Future Efforts

Extending Morris' correlation distance for unknown weight function

- Let $Y \in \mathbb{R}$ be a scalar-valued output
- Let $\mathbf{X}$ be a D-sized input vector, $D \in \mathbb{N}$
- Assume $Y = f(\mathbf{X})$
- Assume $Y \sim GP(\mu, \sigma^2)$, $\mu \in \mathbb{R}$ and $\sigma^2 > 0$
- Recall the SE Kernel with ARD

$$k(\mathbf{X_i}, \mathbf{X_j}) = \sigma^2_{ard} e^{-\frac{1}{2} D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w})}$$

$$D(\mathbf{X_i}, \mathbf{X_j}, \mathbf{w}) = \sum_{d=1}^{D} w_d (X_{i,d} - X_{j,d})^2$$

For a continuous index $t \in [0, T], T \in \mathbb{R}^+$

$$D(\cdot) = \int_0^t w(t-s)(X_i(s) - X_j(s))^2 ds$$

$$w : \mathbb{R}^+ \rightarrow \mathbb{R}^+$$

The model now becomes as follows

$$\begin{aligned} \mu_w &\sim \pi(\mu_w) \\ \sigma^2_w &\sim \pi(\sigma_w) \\ \log w(t) &\sim GP(\mu_w, \sigma^2_w) \\ Y &\sim GP(\mu_Y, \sigma^2_Y) \end{aligned}$$

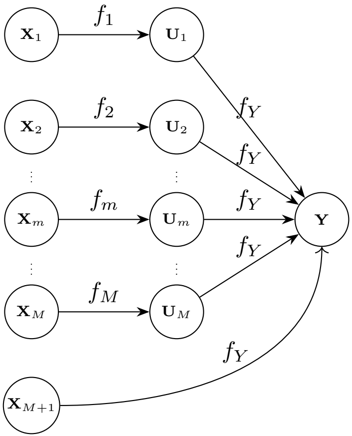
for some priors $\pi(\mu_w)$, $\pi(\sigma_w)$

Future Efforts

Deep Gaussian Process

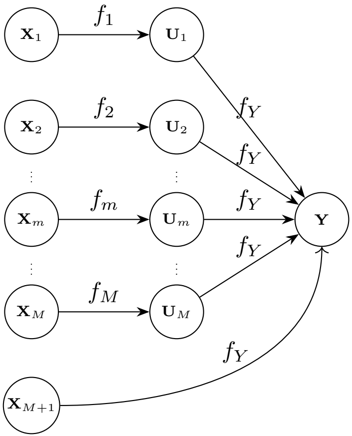
Future Efforts

Deep Gaussian Process



Future Efforts

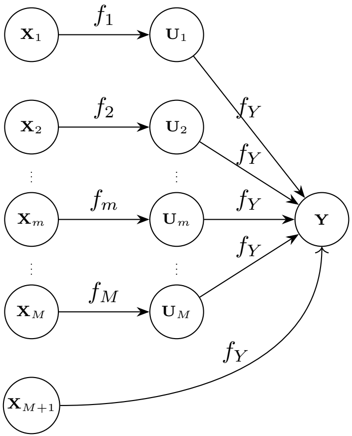
Deep Gaussian Process



- High-dimensional input vector $\mathbf{X}$ with $M \in \mathbb{N}$ functional inputs.

Future Efforts

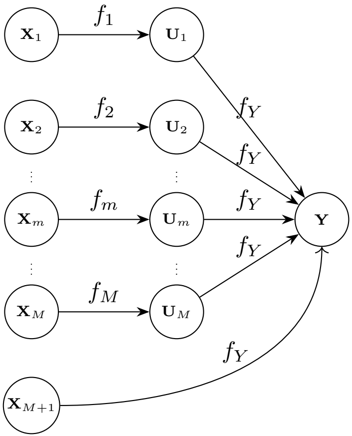
Deep Gaussian Process



- High-dimensional input vector $\mathbf{X}$ with $M \in \mathbb{N}$ functional inputs.
- Let $\mathbf{X} \in \mathbb{R}^{D_1} \times \cdots \times \mathbb{R}^{D_M} \times \mathbb{R}^{D_{M+1}}$, where $D_1, \ldots, D_M, D_{M+1} \in \mathbb{N}$, be a $D = \sum_{m=1}^{M} D_m + D_{M+1}$ dimensional vector.

Future Efforts

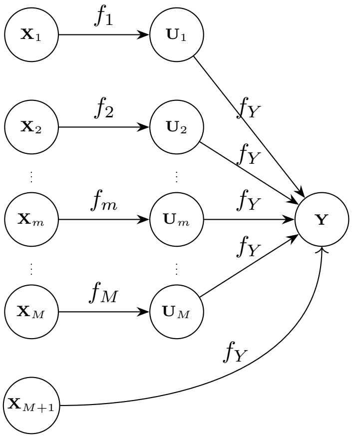
Deep Gaussian Process



- High-dimensional input vector $\mathbf{X}$ with $M \in \mathbb{N}$ functional inputs.
- Let $\mathbf{X} \in \mathbb{R}^{D_1} \times \cdots \times \mathbb{R}^{D_M} \times \mathbb{R}^{D_{M+1}}$, where $D_1, \ldots, D_M, D_{M+1} \in \mathbb{N}$, be a $D = \sum_{m=1}^{M} D_m + D_{M+1}$ dimensional vector.
- For each functional $m \in \{1, \ldots, M\}$, let $\mathbf{U_m} = \mathbf{f_m}(\mathbf{X_m}) + \varepsilon_m$ and assume $\mathbf{f_m} \sim GP(\mu_\mathbf{m}, \sigma_\mathbf{m})$ for $\mu_\mathbf{m} \in \mathbb{R}^{D_m}$

Future Efforts

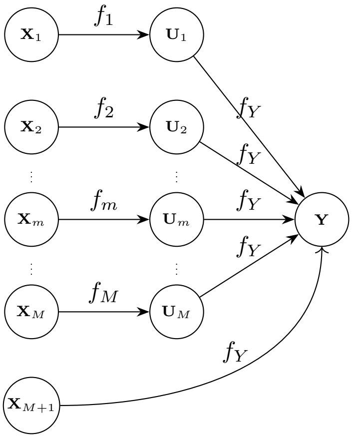
Deep Gaussian Process



- High-dimensional input vector $\mathbf{X}$ with $M \in \mathbb{N}$ functional inputs.
- Let $\mathbf{X} \in \mathbb{R}^{D_1} \times \cdots \times \mathbb{R}^{D_M} \times \mathbb{R}^{D_{M+1}}$, where $D_1, \ldots, D_M, D_{M+1} \in \mathbb{N}$, be a $D = \sum_{m=1}^{M} D_m + D_{M+1}$ dimensional vector.
- For each functional $m \in \{1, \ldots, M\}$, let $\mathbf{U_m} = \mathbf{f_m}(\mathbf{X_m}) + \varepsilon_m$ and assume $\mathbf{f_m} \sim GP(\mu_\mathbf{m}, \sigma_\mathbf{m})$ for $\mu_\mathbf{m} \in \mathbb{R}^{D_m}$
- $\tilde{\mathbf{X}} = (\mathbf{U_1}, \ldots, \mathbf{U_M}, \mathbf{X_{M+1}})$

Future Efforts

Deep Gaussian Process



- High-dimensional input vector $\mathbf{X}$ with $M \in \mathbb{N}$ functional inputs.
- Let $\mathbf{X} \in \mathbb{R}^{D_1} \times \cdots \times \mathbb{R}^{D_M} \times \mathbb{R}^{D_{M+1}}$, where $D_1, \ldots, D_M, D_{M+1} \in \mathbb{N}$, be a $D = \sum_{m=1}^{M} D_m + D_{M+1}$ dimensional vector.
- For each functional $m \in \{1, \ldots, M\}$, let $\mathbf{U_m} = \mathbf{f_m}(\mathbf{X_m}) + \varepsilon_m$ and assume $\mathbf{f_m} \sim GP(\mu_\mathbf{m}, \sigma_\mathbf{m})$ for $\mu_\mathbf{m} \in \mathbb{R}^{D_m}$
- $\tilde{\mathbf{X}} = (\mathbf{U_1}, \ldots, \mathbf{U_M}, \mathbf{X_{M+1}})$
- $Y = f_Y(\tilde{\mathbf{X}}) + \varepsilon_y$ and $f_Y \sim GP(\mu_Y, \sigma_Y)$

Summary

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

- Automatically selects the number of knots
- Similar predictive performance to simultaneous knot selection
- Better represents full GP compared to simultaneous knot selection
- Reduced runtimes compared to simultaneous knot selection

Future work

- Extending the functional input correlation function for unknown weights
- Deep Gaussian Process for the emulation of functional

These slides are available at

- `https://github.com/jarad/LANL2020`
- `http://www.jarad.me/research/presentations.html`

Thank you!

Other links:

- `http://www.jarad.me/`
- `https://luisdamiano.github.io/`

References I

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.

Nathaniel Garton. Score-based likelihood ratios and sparse gaussian processes. doi: 10/ghbt2c. URL https://lib.dr.iastate.edu/etd/17938.

Nathaniel Garton, Jarad Niemi, and Alicia Carriquiry. Knot selection in sparse gaussian processes with a variational objective function. 13 (4):324–336. ISSN 1932-1872. doi: 10/ghbt2d. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/sam.11459. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/sam.11459.

Max D. Morris. Gaussian surrogates for computer models with time-varying inputs and outputs. 54(1):42–50. ISSN 0040-1706. doi: 10/ghbnds. URL https://doi.org/10.1080/00401706.2012.648870. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/00401706.2012.648870.

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

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

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.