Meeting Update

Gaussian Process Covariance Function Selection Using Minimum Energy Designs

Kristyn Pantoja

Department of Statistics
Texas AM University

21 March 2019

Outline

Last Time

Last Meeting Recap

What happened last time

1. Review of Joseph et. al. 2015: One-at-a-Time Algorithm for SMED
2. One-at-a-Time SMED Algorithm for (Bayesian) Linear Model Selection
3. Review of Joseph et. al. 2018: Fast Algorithm for SMED
4. Fast SMED Algorithm for (Bayesian) Linear Model Selection

Corrections to the algorithm

1. Using max instead of sum in the criterion
2. Candidate sets for each design $k = 1, ..., K$

A Fast Algorithm for Linear Model Selection

Generate N-point MED $\mathbf{D}_N$ to distinguish linear models H_0 & H_1:

1. Sample β_i from each prior: $\beta_i \sim N(\tilde{\beta}_i, \sigma^2_{\beta_i}), i = 0, 1$
2. Obtain N candidate points, $\mathbf{D}_1$, in $[0, 1]$ using a space-filling design. These points will be $\tilde{\mathbf{D}}_1$, which are used to obtain the candidate set, $\mathbf{C}_2^j$, for the next design, $k = 2$.
3. For $k = 1, ..., K-1$, get next design $\mathbf{D}_{k+1}$: for $j = 1, ..., N$, Determine L_{jk} and use it to obtain N candidate points, $\tilde{\mathbf{D}}_k^j$ to get candidate set $\mathbf{C}_{k+1}^j$ for selection of design point $\mathbf{x}_j^{k+1}$, in the next design, $\mathbf{D}_{k+1}$ (details in next slide).
   Find the design point $\mathbf{x}_j^{k+1}$ that minimizes

$$\max_{i \neq j} \frac{1}{f^{\gamma_k}(\mathbf{x}_i) f^{\gamma_k}(\mathbf{x}_j) d(\mathbf{x}_i, \mathbf{x}_j)} \tag{1}$$

   where $\gamma_k = k/(K-1)$, $d(\cdot, \cdot)$ is Euclidean distance, $f(\mathbf{x}) = \{\text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})\}^{(1/2p)}$, and where

$$\phi_{0,\mathbf{x}} = N(\tilde{\beta}_0 \mathbf{x}, \sigma^2_{\epsilon_0} + \mathbf{x}^2 \sigma^2_{\beta_0}),$$

$$\phi_{1,\mathbf{x}} = N(\tilde{\beta}_1 \mathbf{x}, \sigma^2_{\epsilon_1} + \mathbf{x}^2 \sigma^2_{\beta_1})$$

A Fast Algorithm, continued...

Updating $\mathbf{x}_j^{k+1}$, jth design point of design $\mathbf{D}_{k+1}$:

3. To find design point $\mathbf{x}_j^{k+1}$, use a method similar to Greedy Algorithm (max, not sum):

$$\mathbf{x}_j^{k+1} = \arg\min_{\mathbf{x}\in\mathbf{C}_{k+1}^j} \max_{i=1:(j-1)} \frac{1}{f^{\gamma_k}(\mathbf{x}_i)f^{\gamma_k}(\mathbf{x})d(\mathbf{x}_i,\mathbf{x})} \qquad (2)$$

Section 3: What is the Candidate Set, then?

For now (not including linear combinations of adjacent points):

1. For each $j = 1, \ldots, N$, the space-filling design over L_{jk} gives the n candidate points $\tilde{\mathbf{D}}_{k+1}^{j}$. We obtain the corresponding $\mathbf{x}_j^{k+1}$ by optimizing over the set $\mathbf{C}_{k+1}^{j} = \mathbf{C}_k^{j} \cup \tilde{\mathbf{D}}_{k+1}^{j}$. Hence, for each of the n design points in design $\mathbf{D}_k$, n candidate points are created from the space-filling design and combined with $\mathbf{C}_1^{j}$ for a total of $2n$ candidate points for $\mathbf{x}_j^k$ to be picked from.
2. For $k = 1$, $\mathbf{C}_1^{j} = \mathbf{D}_1$, where $\mathbf{D}_1$ is from a space-filling design over the support $[0, 1]^p$.
3. Then for $k = 1, \ldots, K-1$, to get next design $\mathbf{D}_{k+1}$, the first design point is $\mathbf{x}_1^{k+1} = \underset{\mathbf{x} \in \mathbf{C}_{k+1}^{1}}{\arg\max} \log f(\mathbf{x})$, and for the next $j = 2, \ldots, n$ design points,

$$\mathbf{x}_j^{k+1} = \underset{\mathbf{x} \in \mathbf{C}_{k+1}^{j}}{\arg\max} \min_{i=1:(j-1)} \gamma_k \log f(\mathbf{x}) + \gamma_k \log f(\mathbf{x}_i^{k+1}) + 2p \log d(\mathbf{x}, \mathbf{x}_i^{k+1}) \tag{3}$$

(p. 9 of Joseph 2018).

Section 3: What is the Candidate Set, then? continued...

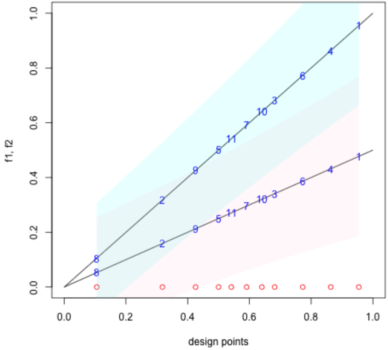
Just to further illustrate this point (because I will be confused again):

- For $k = 1$, $\mathbf{C}_1 = \text{Lattice}([0,1]^p, N)$ and the resulting design is $\mathbf{D}_1$.
- For $k = 2$ and $j = 1, ..., N$, $\mathbf{C}_2^j = \mathbf{C}_1 \cup \text{Lattice}(L_{j2}, N)$ $= \text{Lattice}([0,1]^p, N) \cup \text{Lattice}(L_{j2}, N)$
- For $k = 3$ and $j = 1, ..., N$, $\mathbf{C}_3^j = \mathbf{C}_2^j \cup \text{Lattice}(L_{j3}, N)$ $= \text{Lattice}([0,1]^p, N) \cup \text{Lattice}(L_{j2}, N) \cup \text{Lattice}(L_{j3}, N)$ $= \text{Lattice}([0,1]^p, N) \cup_{m=1}^{3} \text{Lattice}(L_{jm}, N)$
- ...
- Hence, for $k = 2, ..., K$ and $j = 1, ..., N$, $\mathbf{C}_k^j = \text{Lattice}([0,1]^p, N) \cup_{m=1}^{k} \text{Lattice}(L_{jm}, N)$

Results

The 2 Models

- $f_0(\mathbf{x}) = \mathbf{x}\beta_0$ with $\tilde{\beta}_0 = 1$
- $f_1(\mathbf{x}) = \mathbf{x}\beta_1$ with $\tilde{\beta}_1 = 1/2$
- $\sigma_{\epsilon_0} = \sigma_{\epsilon_1} = 0.1, \sigma_\mu = 0.05$
- $K = \lceil 4\sqrt{(p)} \rceil = 4$ sequential designs
- $N = 11$, selected arbitrarily
- number of candidate points is largest prime number less than $100 + 5p = 103$



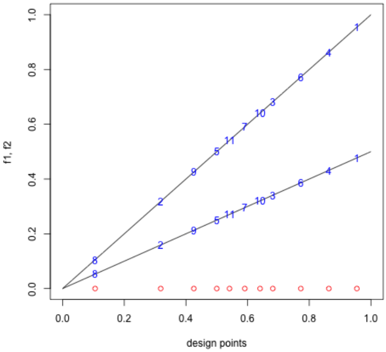
Questions

- Is N supposed to be equal to the number of candidate points?
- Interpreting errors on marginal $y|H_i$?

Compare Results

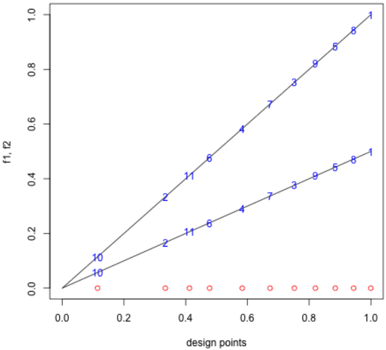
Compare this to the one-at-a-time algorithm (as in Joseph et. al. 2015) with 11 sequentially picked design points, 1000 candidate points, and a power of $k = 4$:

Fast Algorithm



Points are more concentrated at middle of support.

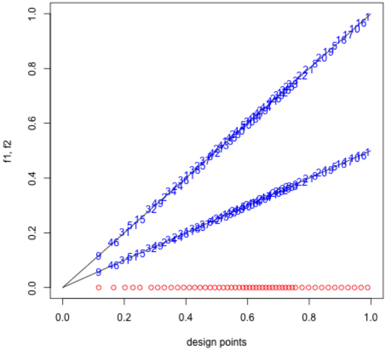
One-at-a-Time Greedy Algorithm



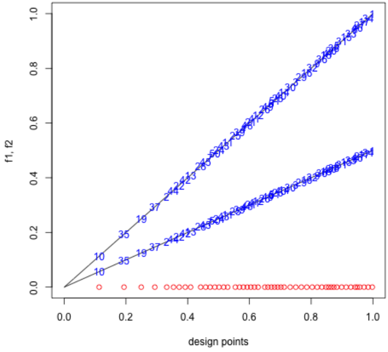
Compare Results, continued

What about for higher N$(= 51)$? Not quite the same...

Fast Algorithm



One-at-a-Time Greedy Algorithm



Questions/Goals from Last Time

1. ~~Solve problem of too-near points (candidate set)~~
2. ~~Show the spread at each design point.~~
3. Evaluate the design.
4. For the one-at-a-time algorithm (Joseph 2015), does large k recover the asymptotic result in the 2018 paper?
5. Asymptotic properties of SMED for (Bayesian) Linear Model Selection
6. Distinguishing models with Gaussian Process Kernels
   - 2 Different Kernels: Matern vs. Squared Exponential
   - Same Kernel, different Parameters: Matern, scaled differently

Evaluate the Design

Ideas for Evaluating the Design

- Expected Posterior Probability of a model

$$E_Y[P(H_\ell|X,Y)], \ell \in \{1,2\}$$

- "Optimal Discrimination Designs" (Dette & Titoff 2009)
  - Their goal: Design for model discrimination (not parameter estimation)
  - Our goal: Get parameter estimates and error, in addition to being able to adequately discriminate between the two models. So, both!
  - Come up with a criterion for measuring "goodness" of design for parameter estimates <u>and</u> ability to discriminate between models
  - Compare criterions
  - Compare designs

Expected Posterior Probability of Model

- Posterior probability of a model given by $H_\ell, \ell = 1, ..., M$ to see which model is more likely to be the correct one.
- Posterior Probability is

$$P(H_\ell|Y) = \frac{\pi_\ell P(Y|H_\ell)}{\sum_{m=1}^{M} \pi_m P(Y|H_m)} \tag{4}$$

  where π_m is the prior on the model given by H_m, and $P(Y|H_m)$ is the model evidence.
- Consider the linear model,

$$y = f(\mathbf{x}) + \epsilon$$
$$f(x) = \mathbf{x}\beta$$

  where $\epsilon \sim N(0, \sigma_\epsilon^2)$ and $\beta \sim N(\tilde{\beta}, \sigma_\beta^2)$.
- For model H_ℓ with $\epsilon_\ell \sim N(0, \sigma_{\epsilon_\ell}^2)$ and $\beta_\ell \sim N(\tilde{\beta}_\ell, \sigma_{\beta_\ell}^2)$, we have $y|H_\ell \sim N(\tilde{\beta}_\ell \mathbf{x}, \sigma_{\epsilon_\ell}^2 + \mathbf{x}^2\sigma_{\beta_\ell}^2)$ where the mean and variance are computed by iterated expectation and variance.
- Since we don't have Y to calculate the model evidence, instead calculate the expected model evidence $E_Y[P(Y|H_m)]$

Estimate Expected Posterior Probability of Model

1. Obtain design $\mathbf{D} = \{x_1, ..., x_N\}$ from the Fast Algorithm for Linear Model Selection.
2. Draw $y_i^{(j)} \sim N(\tilde{\beta}_\ell \mathbf{x}, \sigma^2_{\epsilon_\ell} + \mathbf{x}^2 \sigma^2_{\beta_\ell}), \forall \mathbf{x}_i \in \mathbf{D}, j = 1, ..., J$ to obtain J simulations of $Y = \{y_1, ..., y_N\}$ for model H_ℓ, which gives the linear model:

$$y = f(\mathbf{x}) + \epsilon$$
$$f(x) = \mathbf{x}\beta_\ell$$

   where $\epsilon \sim N(0, \sigma^2_{\epsilon_\ell})$ and $\beta_\ell \sim N(\tilde{\beta}_\ell, \sigma^2_{\beta_\ell})$.
3. Estimate $E_y[P(Y|H_m)] \approx \frac{1}{J} \sum_{j=1}^{J} P(Y|H_m), \forall m \in \{1, ..., M\}$, where $P(Y|H_m)$ is the pdf of the distribution $N(\tilde{\beta}_m \mathbf{x}, \sigma^2_{\epsilon_m} + \mathbf{x}^2 \sigma^2_{\beta_m})$.
4. Then estimate $E_Y[P(H_\ell|Y)]$ by

$$E_Y[P(H_\ell|Y)] = \frac{\pi_\ell E_Y[P(Y|H_\ell)]}{\sum_{m=1}^{M} \pi_m E_Y[P(Y|H_m)]} \tag{5}$$

Questions about Step 3

- How to get a single number out of this computation?
- Where to put the design points $\mathbf{D} = \{x_1, ..., x_N\}$?
- What do we do we do with $y_i^{(j)}$ from simulations?
- Is there some kind of likelihood for linear regression? Some sources talk about Laplace approximation to marginal Y... did I compute that wrong? I averaged over β, am I supposed to somehow average over $\mathbf{D}$ as well?
- Need to get rid of $\mathbf{x}$'s somehow...

What I Ended Up Doing

1. Obtain design $\mathbf{D}$ from the Fast Algorithm for Linear Model Selection.
2. Draw $y_i^{(j)} \sim N(\tilde{\beta}_\ell \mathbf{x}, \sigma^2_{\epsilon_\ell} + \mathbf{x}^2 \sigma^2_{\beta_\ell}), \forall \mathbf{x}_i \in \mathbf{D}, j = 1, ..., J$ for model H_ℓ. For now, $M = 2$.
3. Estimate $E_y[P(Y|H_m)] \approx \frac{1}{JN} \sum_{j=1}^{J} \sum_{i=1}^{N} P(y_i|H_m, \mathbf{x}_i)$, $\forall m \in \{1, ..., M\}$, where $P(y_i|H_m, \mathbf{x}_i)$ is the pdf of $N(\tilde{\beta}_m \mathbf{x}, \sigma^2_{\epsilon_m} + \mathbf{x}^2 \sigma^2_{\beta_m})$.
4. Assume $\pi_m = \frac{1}{M}, \forall m = 1, ..., M$ and estimate $E_Y[P(H_\ell|Y)]$ by

$$E_Y[P(H_\ell|Y)] \approx \frac{E_Y[P(Y|H_\ell)]}{\sum_{m=1}^{M} \pi_m E_Y[P(Y|H_m)]} \tag{6}$$

5. Since $M = 2$, can also compute the Bayes Factor,

$$BF_{01} \approx \frac{E_Y[P(Y|H_0)]}{E_Y[P(Y|H_1)]} \tag{7}$$

Results

For both H_0, H_1, $\sigma_\epsilon = 0.01, \sigma_\beta = 0.001, N = 51$.
$\tilde{\beta}_0 = 1, \tilde{\beta}_0 = 1/2$.

For generating Y under the null model given by H_0,

- $E_Y[P(H_0|Y)] \approx 0.8211184$
- $E_Y[P(H_1|Y)] \approx 0.1788816$
- $BF_{01} \approx 4.590291$: supports H_0

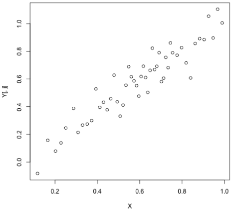
For generating Y under the alternative model given by H_1,

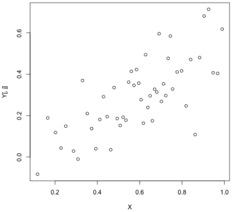
- $E_Y[P(H_0|Y)] \approx 0.1801486$
- $E_Y[P(H_1|Y)] \approx 0.8198514$
- $BF_{01} \approx 0.2197332$: supports H_1

Results, continued...

For both H_0, H_1, $\sigma_\epsilon = 0.01, \sigma_\beta = 0.001, N = 51$.
$\tilde{\beta}_0 = 1, \tilde{\beta}_0 = 1/2$.

Example Simulation from H_0 Model,





Interpreting Bayes Factors (Kass & Raftery 1995):

BF_{01}	Evidence for H_0
< 1	Supports H_1
1 to 3	not worth mentioning
3 to 20	positive
20 to 150	Strong
> 150	v strong

“Optimal Discrimination Designs” (Dette & Titoff 2009)

- Under Construction!

Large k in Joseph et. al. 2015

Recap of Stated Asymptotic Behavior of MED

Generalized Version of MED (Joseph et. al. 2015)

Choose the design $\mathbf{D} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ given by

$$\min_{\mathbf{D}} \mathrm{GE}_k = \left\{ \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(\frac{q(\mathbf{x}_i) q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)} \right)^k \right\}^{1/k} \tag{8}$$

$k \in [1, \infty)$ ((4) in Joseph 2015)

Criterion when $k \to \infty$ (p.5 in Joseph 2015)

$$\max_{\mathbf{D}} \min_{i,j} \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{q(\mathbf{x}_i) q(\mathbf{x}_j)} \tag{9}$$

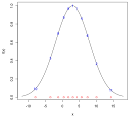
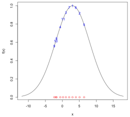
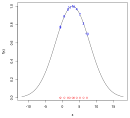
Criterion when $k \to \infty$ (p.5 in Joseph 2018)

$$E(\mathbf{D}) = \max_{i \neq j} \frac{q(\mathbf{x}_i) q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)} \tag{10}$$

What We See

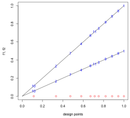
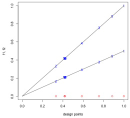
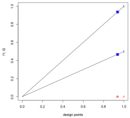
One-At-A-Time SMED

Concentrated at the mode and sometimes to one side.

k	Design
$k = 4$	
$k = 5000$	
$k = 5000$	

One-At-A-Time SMED LM

Also get stuck at one place, but sooner and more severe.

k	Design
$k = 4$	
$k = 200$	
$k = 500$	

Asymptotic properties of SMED for Linear Model Selection

Asymptotic properties of SMED for Linear Model Selection

Asymptotic properties of SMED for Linear Model Selection

- a distribution?
- a distance?

First, take a look how the asymptotic properties of SMED were made.

- Chapter 5 of Yan Wang's thesis
- Joseph et al 2015 & Supplementary Material
- Joseph et al 2018 & Supplementary Material

Chapter 5 of Yan Wang's Thesis

Main Points

Chapter 5 of "Asymptotic Theory for Decentralized Sequential Hypothesis Testing Problems and SMED Algorithm," thesis by Yan Wang is on the asymptotic properties of the SMED Algorithm (an unpublished version).

- "... good design should be able to estimate the global optimum accurately with as few design points as possible."
- A comprehensive review of efficient designs for low-dimensional design space (e.g. fractional factorial), and space-filling designs for dimension $/approx 10$ design spaces (e.g. Latin hypercube), semi-sequential designs (PI and EI algorithms)
- Motivates sequential design that incorporate information from previous evaluated points into subsequent choice of test points - which is what SMED does.
- Some theorems on denseness and asymptotic distribution of design

Chapter 5 Theorems on Denseness

Notation

- Let $\mathcal{E} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots\}$ be the infinite sequence of design points obtained from running the algorithm without stopping.
- Assume $q(x) \in [0, 1]$ and call lower and upper bounds of $q(x)$ by $\underline{q} = \min_\chi q(x)$ and $\bar{q} = \max_\chi q(x)$ so that $0 \leq \underline{q} \leq \bar{q} \leq 1$.

Theorems on Denseness

- 5.3.1 If $\beta > m$, then $\bar{\mathcal{E}} = \chi$
- 5.3.2 If $\bar{q} > 0$ and $\beta = m = 1$, then $\bar{\mathcal{E}} = \chi$

Chapter 5 Heuristic Arguments for Asymptotic Distribution

Assumptions

- The set $\mathcal{E}$ of SMED design points has a positive density function $\rho(x)$ s.t. $\forall B \subset \chi$,

$$\lim_{n\to\infty} \frac{\sum_{i=1}^{n} I\{x_i \in B\}}{n} = \int_B \rho(x)dx$$

Properties with Heuristic Arguments

- When $\underline{q} > 0$ and $\beta > m$, $\rho(x) \propto (1 - \alpha p(x))^{-\frac{2m\gamma}{\beta}}$.
- the set $\mathcal{E}$ is dense in χ when $\beta = m > 1$. Furthermore, $\rho(x) \sim q(x)^2$.

The rest of Chapter 5 is on a simplification of the SMED algorithm, called "adjusted SMED."

Joseph et al 2015, 2.2 Limiting Dist & Charge Fn

Definitions & Notation

- The index for design $\mathbf{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, denoted by $IN(\mathbf{D})$, is the number of pairs $(\mathbf{x}_l, \mathbf{x}_m)$ with the smallest value of $d(\mathbf{x}_i, \mathbf{x}_j)/q(\mathbf{x}_i)q(\mathbf{x}_j)$ over all $i \neq j$.
- Let $\mathbf{D}^* = \{\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*\}$ be the MED with the smallest index.
- Let $\mathcal{B}$ be the Borel σ-algebra over $\chi = [0,1]^p$.
- Define the probability measures on $(\chi, \mathcal{B})$:

$$\mathcal{P}_n(A) = \frac{\#\{\mathbf{x}_i^* : 1 \leq i \leq n, \mathbf{x}_i^* \in A\}}{n}, \forall A \in \mathcal{B} \qquad (11)$$

Theorems & Results

- Theorem 1. If $q(x) \equiv 1$, $\mathcal{P}_n \xrightarrow{d} Uniform(\chi)$
- Result 1. If $1/q$ is differentiable over χ and bounded away from 0, $\exists \mathcal{P}$ with density f s.t. $\mathcal{P}_n$ converges to $\mathcal{P}$ and $f(x) \propto 1/q^{2p}(x)$

Joseph et al 2015 Appendix & Supplementary Material

Appendix

- Lemma 1 & Lemma 2 for proving Theorem 1.

Supplementary Materials

- Proofs of Theorem 1, Result 1, and the two lemmas in the Appendix (in addition to some figures).

Joseph et al 2018

Definitions

- Generalized distance:

$$d_s(\mathbf{u}, \mathbf{v}) = (\frac{1}{p}\sum_{l=1}^{p} |u_l - v_l|^s)^{1/s} \tag{12}$$

  where $s > 0$.

- MinED criterion using generalized distance:

$$\max_{\mathbf{D}} \min_{i \neq j} f^{1/(2p)}(\mathbf{x}_i) f^{1/(2p)}(\mathbf{x}_j) d_s(\mathbf{x}_i, \mathbf{x}_j) \tag{13}$$

  Denote this design by $MinED_s$

Joseph et al 2018

Theorem 1

Suppose the charge function $q(\cdot)$ is Lipschitz continuous, i.e. $|q(\mathbf{u} - q(\mathbf{v})| \leq LD(\mathbf{u}, \mathbf{v})$, for $\mathbf{u}, \mathbf{v} \in \chi$ and a constant $L > 0$.
Let $\mathbf{D}^*$ be the n-point MED design using 13 with the smallest index and $\mathcal{B}$ be the Borel σ-algebra of χ.
Define the following probability measures on $(\chi, \mathcal{B})$:

$$\mathcal{P}_n(A) = \frac{card\{\mathbf{x}_i^* : 1 \leq i \leq n, \mathbf{x}_i^* \in A\}}{n}, \forall A \in \mathcal{B} \tag{14}$$

Then there $\exists \mathcal{P}$ with density f s.t. $\mathcal{P}_n$ converges weakly to $\mathcal{P}$ and $f(x) \propto 1/q^{2p}(x)$ over χ.
When $s \to 0$, the criterion becomes

$$\max_{\mathbf{D}} \min_{i \neq j} f^{1/(2p)}(\mathbf{x}_i) f^{1/(2p)}(\mathbf{x}_j) \prod_{l=1}^{p} |\mathbf{x}_{il}, \mathbf{x}_{jl}|^{1/p} \tag{15}$$

???

Joseph et al 2018 Supplementary Material

Theorem 1 shows that the limiting distribution of $MinED_s$ is $f(\mathbf{x})$ irrespective of the value of $s \in (0, \infty)$. The proof is given in the Supplementary material. Requires measure theory. Big yikes.

Distinguishing Gaussian Processes

Distinguishing Gaussian Processes

Distinguishing models with Gaussian Process Kernels

- 2 Different Kernels, e.g. Matern vs. Squared Exponential
- Same Kernel, different Parameters, e.g. Squared Exponential with different scale

The general model: $y = \eta(x, \theta) + \epsilon$

Questions about GP Set-Up

Distinguishing between Functions vs. Distributions over Functions

- Where to begin?
- Distance?
  - 2-Wasserstein distance for GP in "Learning from uncertain curves: The 2-Wasserstein metric for Gaussian processes" (Mallasto &Feragen 2017)
  - Wasserstein distance between the posteriors
- How to choose design points?
- How to evaluate them? What's $\phi_{0,\mathbf{x}}$ and $\phi_{1,\mathbf{x}}$ now?
- Are we trying to evaluate each GP's ability to model some thing?
- As opposed to evaluating a model's ability to estimate parameters?

Reminder of Gaussian Process Stuff

(Gaussian Processes for Machine Learning, Rasmussen 2006)

1. GP Prior: $GP(\mathbf{0}, K(\cdot,\cdot))$, where $\mathbf{0}$ is the mean function, $\mu(\mathbf{x})$ and $K(\cdot,\cdot))$ is the covariance function.
2. The null distribution is
$$f_* \sim N(\mathbf{0}, K(\mathbf{X}_*, \mathbf{X}_*)) \tag{16}$$
where $\mathbf{X}_*$ is a vector of design points, $\{\mathbf{x}_1, ..., \mathbf{x}_n\}$, which basically amounts to a draw from the Gaussian Process (i.e. a draw from a GP is MVN).
3. To make predictions for design points $\{\mathbf{x_1}^*, \ldots, \mathbf{x_{n_*}}^*\}$ given noise-free observations $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n_0}$,
$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim N\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} K(\mathbf{X},\mathbf{X}) & K(\mathbf{X},\mathbf{X}_*) \\ K(\mathbf{X}_*,\mathbf{X}) & K(\mathbf{X}_*,\mathbf{X}_*) \end{pmatrix}\right] \tag{17}$$
This is the posterior predictive distribution.
Note: since it's noise-free, $f_i = y_i, \forall i$

Idea: Wasserstein b/w Post. Predictives

Generate N-point MED $\mathbf{D}_N$ to distinguish GPs H_0 & H_1:

1. Obtain N candidate points, $\mathbf{x}$, in $[0, 1]$ using a space-filling design and obtain f_* from the null distribution of H_0 & H_1 each:
2. For $k = 1, ..., K - 1$, get next design $\mathbf{D}_{k+1}$: for $j = 1, ..., N$, Determine L_{jk}, obtain N candidate points, $\tilde{\mathbf{D}}_k^j$ to get candidate set $\mathbf{C}_{k+1}^j$ for selection of design point $\mathbf{x}_j^{k+1}$. Find the design point $\mathbf{x}_j^{k+1}$ that minimizes

$$\max_{i=1:(j-1)} \frac{1}{f^{\gamma_k}(\mathbf{x}_i) f^{\gamma_k}(\mathbf{x}_j) d(\mathbf{x}_i, \mathbf{x}_j)} \tag{18}$$

   where $\gamma_k = k/(K-1)$, $d(\cdot, \cdot)$ is Euclidean distance, $f(\mathbf{x}) = \{\text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})\}^{(1/2p)}$, and where $\phi_{0,\mathbf{x}}$ and $\phi_{1,\mathbf{x}}$ are the posterior predictive distributions from each of H_0 and H_1 (see next slide).

A Fast Algorithm, continued...

Posterior Predictives from each of H_0 and H_1

$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim N\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} K_0(\mathbf{X},\mathbf{X}) & K_0(\mathbf{X},\mathbf{X}_*) \\ K_0(\mathbf{X}_*,\mathbf{X}) & K_0(\mathbf{X}_*,\mathbf{X}_*) \end{pmatrix}\right]$$
$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim N\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} K_1(\mathbf{X},\mathbf{X}) & K_1(\mathbf{X},\mathbf{X}_*) \\ K_1(\mathbf{X}_*,\mathbf{X}) & K_1(\mathbf{X}_*,\mathbf{X}_*) \end{pmatrix}\right]$$

Which are MVN so Wasserstein has a closed form for each.

Updating $\mathbf{x}_j^{k+1}$, jth design point of design $\mathbf{D}_k$:

1. To find design point $\mathbf{x}_j^{k+1}$, use a method similar to Greedy Algorithm (max, not sum):

$$\mathbf{x}_j^k = \arg\min_{\mathbf{x}\in\mathbf{C}_{k+1}^j} \max_{i=1:(j-1)} \frac{1}{f^{\gamma_k}(\mathbf{x}_i) f^{\gamma_k}(\mathbf{x}) d(\mathbf{x}_i,\mathbf{x})} \tag{19}$$

...

Some Thoughts

- Difference between initializing with candidate set from space-filling design and initializing with candidate set from $\mathit{Uniform}([0,1]^p)$ (is there a difference, like there was in seminar?)
- Asymptotic properties of other model-type comparisons, besides linear
- We hope that this is more meaningful than points concentrated in the middle and somewhat spread out from there... order matters, too, though - "sequential." But how is 2018 sequential?
- Optimize code for Fast Algorithm for Linear Model Selection, so that we can use N = 103 in linear model selection case (rather than N = 51 in estimates for expected posterior probabilities of models).