Meeting : 30 May 2019

Kristyn Pantoja

5/30/2019

Criteria

Criteria

1. The total potential energy, which both algorithms aim to minimize:

$$\sum_{i \neq j} \frac{q(\mathbf{x}_i)q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)}$$

2. One-at-a-Time Algorithm criterion tries to minimize:

$$\left\{ \sum_{i \neq j} \left(\frac{q(\mathbf{x}_i)q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)} \right)^k \right\}^{1/k}$$

3. Fast Algorithm tries to minimize:

$$\max_{i \neq j} \frac{q(\mathbf{x}_i)q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)}$$

Tuning

Different Powers k and Stages K

- What happens when k gets large in the one-at-a-time algorithm's criterion:

$$\left\{ \sum_{i \neq j} \left(\frac{q(\mathbf{x}_i) q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)} \right)^k \right\}^{1/k}$$

  - the criterion itself is changed
- For the fast algorithm, for $k = 1, \ldots, K-1$ steps, we obtain each design point $\mathbf{x}_j^{(k+1)}$ of the next stage $\mathbf{D}_{k+1}$ by:

$$\mathbf{x}_j^{k+1} = \underset{\mathbf{x} \in \mathbf{C}_j^{k+1}}{\arg\min} \max_{i=1:(j-1)} \frac{q^{\gamma_k}(\mathbf{x}_i) q^{\gamma_k}(\mathbf{x})}{d(\mathbf{x}_i, \mathbf{x})}$$

What happens when we include more stages (larger K)?

- this isn't about a change in the criterion, but a change in the method for minimizing that criterion

In particular, we're interested in:

- the asymptotic behavior of the algorithms
- the limiting distributions of the design
- whether it improves the design's ability to minimize the critera (its own criterion as well sa the total potential energy criterion)

Parameters

- Parameters

```
mean_beta0 = 1 # slope of null model
mean_beta1 = 1 / 2 # slope of alternative model
var_mean = 0.001 # variance on beta
var_e = 0.01 # variance on error
```

- Settings for Fast and One-at-a-Time algorithms

```
N = 67
# for fast algorithm:
K = 20 # ceiling(4* sqrt(p))
numParameters = 1 # number of parameters (just slope!)
p = numParameters
# for one-at-a-time algorithm:
numCandidates = 10^5 # suggested 10^5
k = 4
```

Fast Algorithm (2018)

Fast Algorithm (2018)

In each of K stages, create a new design to iteratively minimize

$$\max_{i \neq j} \frac{q(\mathbf{x}_i)q(\mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_j)}$$

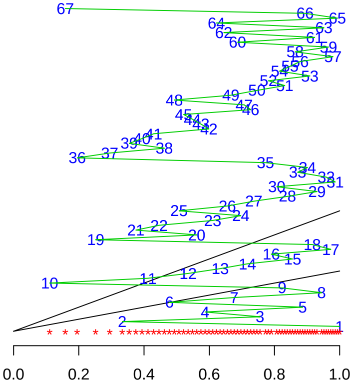
1. Initialize space-filling design $\mathbf{D}_1 = \{\mathbf{x}_1^{(1)} \dots \mathbf{x}_N^{(1)}\}$
2. For $k = 1, \dots, K-1$ steps, obtain each design point $\mathbf{x}_j^{(k+1)}$ of the next stage $\mathbf{D}_{k+1}$ by:

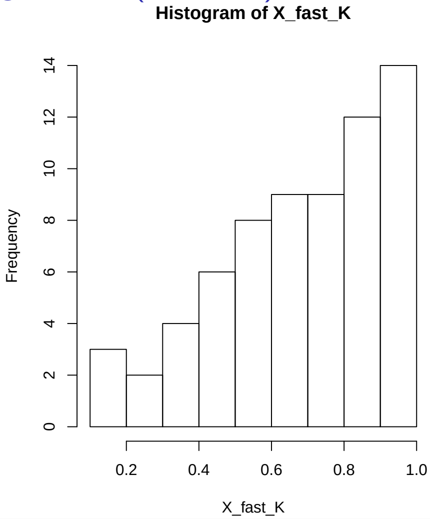
$$\begin{aligned}
\mathbf{x}_j^{k+1} &= \arg\min_{\mathbf{x} \in \mathbf{C}_j^{k+1}} \max_{i=1:(j-1)} \frac{1}{f^{\gamma_k}(\mathbf{x}_i) f^{\gamma_k}(\mathbf{x}) d^{(2p)}(\mathbf{x}_i, \mathbf{x})} \\
&= \arg\min_{\mathbf{x} \in \mathbf{C}_j^{k+1}} \max_{i=1:(j-1)} \frac{q^{\gamma_k}(\mathbf{x}_i) q^{\gamma_k}(\mathbf{x})}{d(\mathbf{x}_i, \mathbf{x})}
\end{aligned}$$

where $\gamma_k = k/(K-1)$ and $\mathbf{C}_j^{k+1}$ is the candidate set for design point $\mathbf{x}_j$ at stage $k+1$.

- points are no longer picked sequentially
- candidates are different for each design point

Design generated by Fast Algorithm (K = 20)



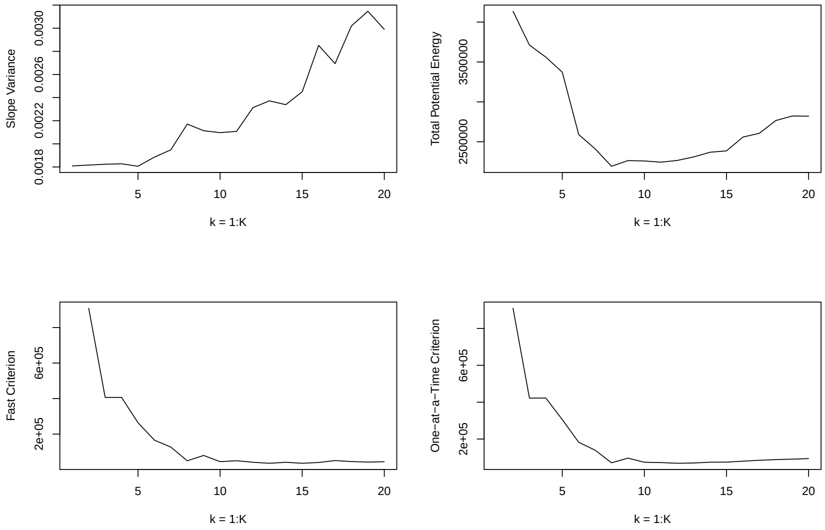



```
mean(X_fast_K); sd(X_fast_K)
```

```
## [1] 0.6836593
```

```
## [1] 0.2296987
```

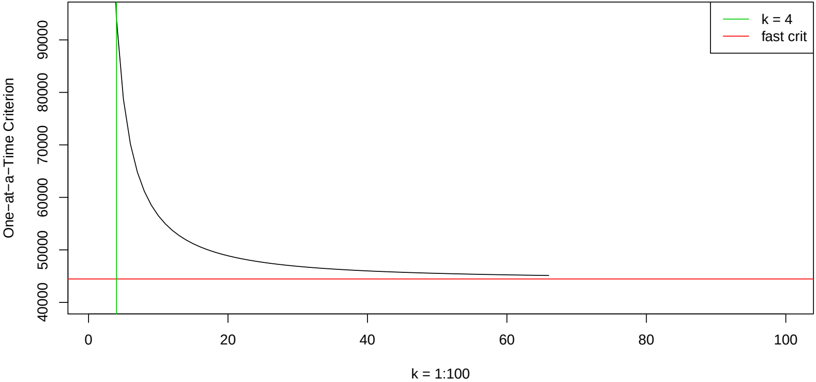
Over the K Designs



- Total Potential Energy criterion starts to increase as the design approaches `K = 20`. It could be that `K = 20` is too large
- Variance on Slope ($\hat{\beta}$) also increases as the stages move away from a space-filling design and toward one that minimizes the fast algorithm's criterion. Why?

One-at-a-Time Criterion, Different k Powers

- To see if it decreases as $k \to \infty$, since this algorithm is supposed to be the asymptotic result of the one-at-a-time algorithm, here are the results for `k = 1:100`.
- For the fast algorithm's design, the one-at-a-time algorithm's criterion approaches the fast algorithm's criterion as k gets large (until it becomes `Inf` - likely due to a computational issue.)



One-at-a-Time Algorithm

One-at-a-Time Algorithm (2015)

1. Obtain *numCandidates* candidate points, $\mathbf{x}$, in $[0, 1]$.
2. Initialize D_N by choosing $\mathbf{x}_j$ to be the candidate $\mathbf{x}$ which optimizes f, where $f(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$ and

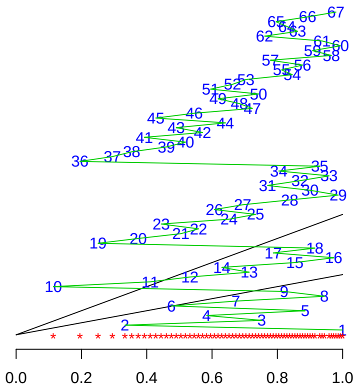
$$\phi_{0,\mathbf{x}} = N(\tilde{\beta}_0\mathbf{x}, \sigma^2_{\epsilon_0} + \mathbf{x}^2\sigma^2_{\beta_0}),$$
$$\phi_{1,\mathbf{x}} = N(\tilde{\beta}_1\mathbf{x}, \sigma^2_{\epsilon_1} + \mathbf{x}^2\sigma^2_{\beta_1})$$

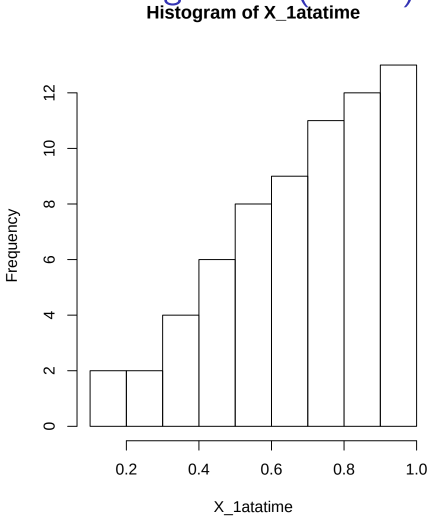
3. Choose the next point $\mathbf{x}_{j+1}$ by:

$$\mathbf{x}_{j+1} = \arg\min_{\mathbf{x}} \sum_{i=1}^{j} \left(\frac{q(\mathbf{x}_i)q(\mathbf{x})}{d(\mathbf{x}_i, \mathbf{x})} \right)^k$$

   where $q = 1/f^{(1/2p)}$, $d(x, y)$ is Euclidean distance and (suggested) $k = 4p$.

Design generated by One-at-a-Time Algorithm (k = 4)





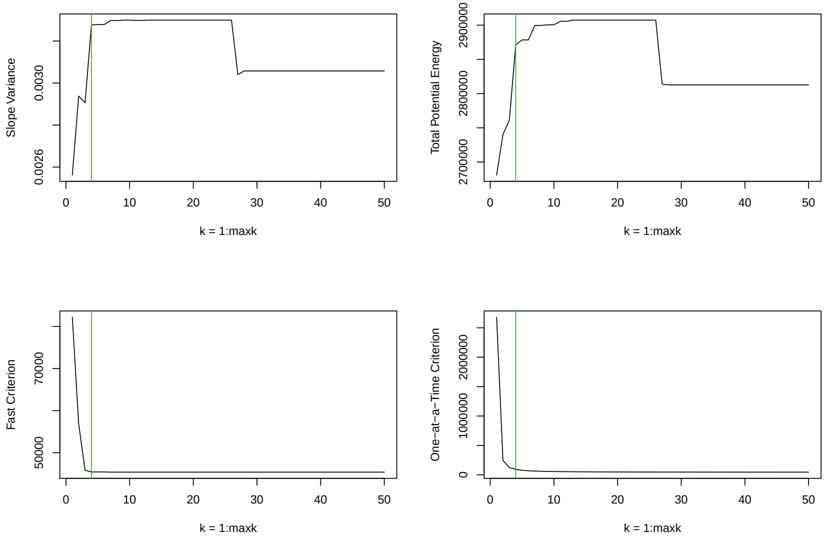

```
mean(X_1atatime)
```

```
## [1] 0.6886419
```

```
sd(X_1atatime)
```

```
## [1] 0.2187448
```

Robustness Across k Power



Comparing Evaluations for Criterion

- Each One-at-a-Time design from `k = 1:50`, evaluated at its corresponding one-at-a-time criterion.
- They appear to be approaching the `k = 4` design's evaluation of the fast algorithm's criterion.
- Also recall that at `k = 4` and after, the designs' evaluations of the fast algorithm's criterion are roughly the same.