A Summary of “Sequential Optimization through Locally Important Dimensions” (SOLID)

Winkel et al. [2021]

Miles Woollacott

Spring 2025

Background and Construction

- Want to find *maximum* (χ) of a multidimensional function $f(x)$.
  - f is slow to optimize over.
- Idea: Sequential optimization of f that only considers a subset of variables at each iteration.
- $\underline{y} = f(\underline{x}) + \underline{\epsilon}$.
  - $f(\underline{x}) \sim GP(\mu \underline{1}_n, \Sigma(x, x'))$.
  - $\underline{\epsilon}$: noise term.
- $K(\underline{x}, \underline{x}') = \exp\left\{-\sum_{k=1}^{p} \gamma_k (x_k - x_k')^2\right\}$.
  - γ_k are unknown!
- **Globally Inactive Variable**: $\gamma_k = 0$.
- **Globally Active Variable**: $\gamma_k > 0$.

How can we identify potentially globally active variables?

Identifying Globally Active Variables

How do we determine if a variable is globally (in)active?

$$\begin{aligned} \gamma_k =& u_k b_k \\ u_k \sim& \text{Gamma}(a_u, b_u) \\ b_k \sim& \text{Ber}(\theta) \\ \theta \sim& \text{Beta}(a_\theta, b_\theta). \end{aligned}$$

If $\hat{b}_k := P(\gamma_k > 0 | \underline{y}) < g$ for some $g \in (0,1)$, then we declare variable k as globally inactive.

GVS

Global Variable Selection: Sequentially remove globally inactive variables from model.

- Dimension reduction by only considering globally active variables.
- Variables declared globally inactive are never considered after removal.
- Acquisition function: **Augmented EI Criterion (AEI)**.

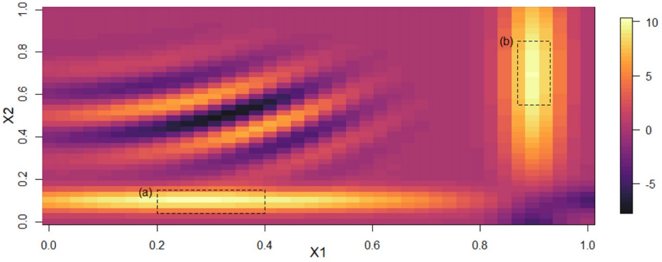
After identifying globally active variables:

1. Sample M posterior draws $\boldsymbol{\Omega}_1, \ldots, \boldsymbol{\Omega}_M$, and calculate $\hat{f}_t = \hat{f}(\cdot|\boldsymbol{\Omega}_t)$.
2. Calculate the estimated global maximizer $\hat{\chi} = \arg\max_{\underline{x}} \left\{ \frac{1}{M} \sum_t \hat{f}_t \right\}$.

Locally Active/Inactive Variables

Want to find reasonable maximum in as few runs as possible. Global variable selection might still be too slow!

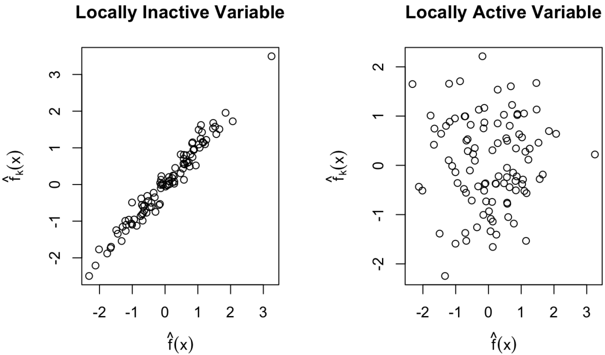
Locally Inactive Variable: The variable is not important in a subset of the input space.



Identifying Locally Active/Inactive Variables

How do we determine if a variable is locally (in)active?

Idea: Compare local predicted surfaces with and without $\gamma_k \overset{\text{set}}{=} 0$.



Identifying Locally Active/Inactive Variables

How do we determine if a variable is locally (in)active?

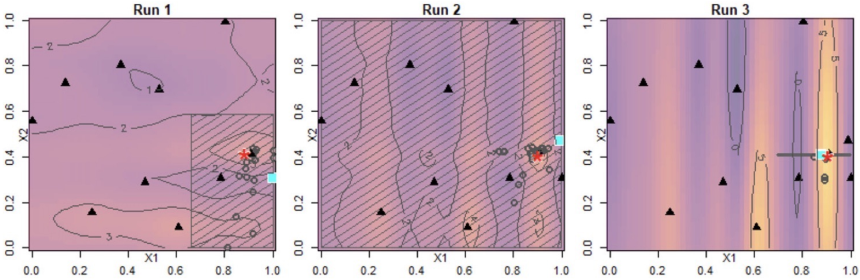
Idea: Compare local predicted surfaces with and without $\gamma_k \overset{\text{set}}{=} 0$.

1. Define local space $\mathcal{F}_t$, centered at $\hat{\boldsymbol{\chi}}_t$ with size δ.
2. Sample points in $\mathcal{F}_t$, denoted as $\boldsymbol{Q}_t \sim TN_{[0,1]^p}(\hat{\boldsymbol{\chi}}_t, \delta \boldsymbol{I})$.
3. Calculate $R^2_{kt} = \text{Corr}\left(\hat{f}_t(\boldsymbol{Q}_t), \hat{f}^k_t(\boldsymbol{Q}_t)\right)^2$.
4. Calculate $L_k = 1 - \frac{1}{M}\sum_{i=1}^{M} R^2_{ki}$, and declare variable k locally inactive if $L_k < \rho$ for some $\rho \in (0,1)$.

SOLID

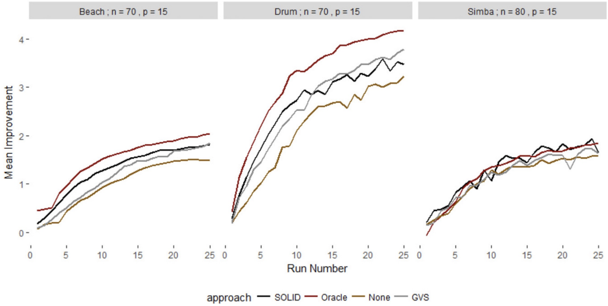
Idea: Sequentially optimize over globally and locally active variables.

1. GVS.
2. Local variable selection (LVS).
3. AEI in LVS space for $\mathcal{R}^{\boldsymbol{A}}$ vs. $\mathcal{R}^{\delta}$.



$\circ = \hat{\chi}_t$, $\blacktriangle$ = design points, $\blacksquare$ = AEI maximizer, $*$ = $\hat{\chi}$,
Shaded area = the region which has the AEI maximizer.

Comparing SOLID



- **Mean Improvement**: $\frac{1}{n}\sum_{i=1}^{n} f(\hat{\chi}_n) - f(\hat{\chi}_0)$.
- SOLID initially improves faster than GVS and None.
  - Other methods tend to "catch up" as number of runs increases.

Area of Improvement: Global Variable Selection

- If a variable is declared globally inactive, it is never considered again.
- If $\hat{b}_k := P(\gamma_k > 0|\underline{y}) < g$ for some $g \in (0, 1)$, then we declare variable k as globally inactive.

References I

M. A. Winkel, J. W. Stallrich, C. B. Storlie, and B. J. R. and. Sequential optimization in locally important dimensions. *Technometrics*, 63(2): 236–248, 2021. doi: 10.1080/00401706.2020.1714738. URL `https://doi.org/10.1080/00401706.2020.1714738`.

The End

Questions?

Appendix A.1: Identifying Locally Inactive Variables

Algorithm Identifying Locally Active Variables [Winkel et al., 2021]

1: Initialize ρ and δ. Also randomly sample $m \leq M$ posterior draws.

- ρ: A cutoff value in the interval $(0, 1)$ that will determine if a variable is locally active/inactive.
- δ: Controls how far the prediction points are spread from $\hat{\boldsymbol{\chi}} = \arg\max_x \hat{f}$.
- M: The number of posterior draws from $\boldsymbol{\Omega}$, the vector of GP parameters.

Appendix A.1: Identifying Locally Inactive Variables

Algorithm Identifying Locally Active Variables [Winkel et al., 2021]

1: Initialize ρ and δ. Also randomly sample $m \leq M$ posterior draws.
2: **for** $t \in \{1, \ldots, m\}$ **do**
3: Estimate $\hat{\boldsymbol{\chi}}_t$ using $\boldsymbol{\Omega}_t$ and all globally active variables.
4: Construct q prediction points $\boldsymbol{Q}_t$ centered around $\boldsymbol{\chi}_t$.
5: Determine baseline predictions $\hat{f}$ at $\boldsymbol{Q}_t$ using $\boldsymbol{\Omega}_t$.
6: **for** variable $k \in \{1, \ldots, p\}$ **do**
7: Make alternative predictions $\hat{f}_t^k$ at $\boldsymbol{Q}_t$ and calculate R^2_{kt}.

- $\boldsymbol{Q}_t \sim TN_{[0,1]^p}(\hat{\boldsymbol{\chi}}_{\boldsymbol{t}}, \delta \boldsymbol{I})$.
- $R^2_{kt} = \text{Corr}\left(\hat{f}_t(\boldsymbol{Q}_t), \hat{f}_t^k(\boldsymbol{Q}_t)\right)^2$.

Appendix A.1: Identifying Locally Inactive Variables

Algorithm Identifying Locally Active Variables [Winkel et al., 2021]

```
Initialize ρ and δ. Also randomly sample m ≤ M posterior draws.
for t ∈ {1, ..., m} do
    Estimate χ̂_t using Ω_t and all globally active variables.
    Construct q prediction points Q_t centered around χ_t.
    Determine baseline predictions f̂ at Q_t using Ω_t.
    for variable k ∈ {1, ..., p} do
        Make alternative predictions f̂_t^k at Q_t and calculate R²_kt.
    end for
end for
Calculate L_k = 1 − R̄²_k.
return A = {k : L_k ≥ ρ|χ̂}, the set of locally active variables. =0
```

Appendix A.2: SOLID

Algorithm SOLID [Winkel et al., 2021]

Set n_0 and N (the maximum number of evaluations). Also set g, δ, ρ, M, m, and c.
Create an initial maximin LHS(n_0, p) design $\boldsymbol{X}$.
Generate $\underline{y}$ from $Y(\boldsymbol{X})$.
for $i \in \{0, \dots, N\}$ **do**
    Obtain M posterior draws of $\boldsymbol{\Omega}_t$ and $\boldsymbol{\chi}_t$, and calculate $\hat{f}$ and $\hat{\boldsymbol{\chi}}$.
    Global Variable Selection: Remove variables with $\hat{b}_k < g$ from $\boldsymbol{X}$. If variables are removed, repeat the previous step with the new $\boldsymbol{X}$.
    Local Variable Selection: Obtain $\boldsymbol{A}$ form previous algorithm.
    Define restricted $\mathcal{R}^{\delta}$ and unrestricted $\mathcal{R}^{\boldsymbol{A}}$ search spaces.
    Localized Optimum Estimation: Update estimate $\hat{\boldsymbol{\chi}}$ in $\mathcal{R}^{\boldsymbol{A}}$ using $\hat{f}$. Store as $\hat{\boldsymbol{\chi}}^{i}$.
    Create maximin LHS designs $\boldsymbol{C}_{\delta} \subset \mathcal{R}^{\delta}$ and $\boldsymbol{C}_{\boldsymbol{A}} \subset \mathcal{R}^{\boldsymbol{A}}$.
    Evaluate AEI in $\boldsymbol{C}_{\delta}$ and $\boldsymbol{C}_{\boldsymbol{A}}$. Define the set with the largest AEI as $\boldsymbol{C}$.

- $AEI(\underline{x}) = \mathbb{E}\left[\max\left\{\hat{f}(\underline{x}) - \hat{f}(\underline{x}^*), 0\right\}\right]\left(1 - \frac{\tau}{\sqrt{s^2(\underline{x}) + \tau^2}}\right)$, where $\underline{x}^* = \arg\max_{\underline{x}_i \in \boldsymbol{X}}\left\{\hat{f}(\underline{x}_i) - \nu \cdot s(\underline{x}_i)\right\}$ for some $\nu \geq 0$, and τ is the nugget term.

Appendix A.2: SOLID

Algorithm SOLID [Winkel et al., 2021]

1: Set n_0 and N (the maximum number of evaluations). Also set g, δ, ρ, M, m, and c.
2: Create an initial maximin LHS(n_0, p) design $\boldsymbol{X}$.
3: Generate $\underline{y}$ from $Y(\boldsymbol{X})$.
4: **for** $i \in \{0, \dots, N\}$ **do**
5: Obtain M posterior draws of $\boldsymbol{\Omega}_t$ and χ_t, and calculate $\hat{f}$ and $\hat{\chi}$.
6: **Global Variable Selection**: Remove variables with $\hat{b}_k < g$ from $\boldsymbol{X}$. If variables are removed, repeat the previous step with the new $\boldsymbol{X}$.
7: **Local Variable Selection**: Obtain $\boldsymbol{A}$ form previous algorithm.
8: Define restricted $\mathcal{R}^{\delta}$ and unrestricted $\mathcal{R}^{\boldsymbol{A}}$ search spaces.
9: **Localized Optimum Estimation**: Update estimate $\hat{\chi}$ in $\mathcal{R}^{\boldsymbol{A}}$ using $\hat{f}$. Store as $\hat{\chi}^i$.
10: Create maximin LHS designs $\boldsymbol{C}_{\delta} \subset \mathcal{R}^{\delta}$ and $\boldsymbol{C}_{\boldsymbol{A}} \subset \mathcal{R}^{\boldsymbol{A}}$.
11: Evaluate AEI in $\boldsymbol{C}_{\delta}$ and $\boldsymbol{C}_{\boldsymbol{A}}$. Define the set with the largest AEI as $\boldsymbol{C}$.
12: **Localized AEI Estimation**: Perform line search optimization to identify $\underline{x}^* = \arg\max_{\underline{x} \in \boldsymbol{C}} AEI(\underline{x})$.
13: Augment $\underline{x}^*$ to $\boldsymbol{X}$. Generate $Y(\underline{x}^*)$ and add to $\underline{y}$.
14: **end for**
15: **return** $\{\hat{\chi}^0, \dots, \hat{\chi}^N\}$. =0