Modifying MED for Model Selection

Kristyn Pantoja

11/06/2019

MED Overview

Minimum Energy Design

Design $\mathbf{D} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ is a MED if it minimizes the total potential energy, given by:

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

Theorem: If $q = \frac{1}{f^{1/2p}}$, the **limiting distribution**[1] of the design points is target distribution, f.

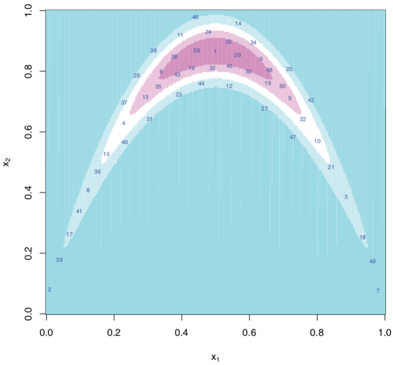


Figure 1: Sampling the "Banana" function

[1] "Sequential Exploration of Complex Surfaces Using Minimum Energy Designs," Joseph et. al. 2015, Result 1

MED for Model Selection

A design $\mathbf{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ to gather data that will

1. help distinguish these two slopes
2. allow adequate estimation of β

Define q in terms of $f_D(x)$, a normalized Wasserstein distance between $y|H_0, X$ and $y|H_1, X$, assuming a bounded design space.

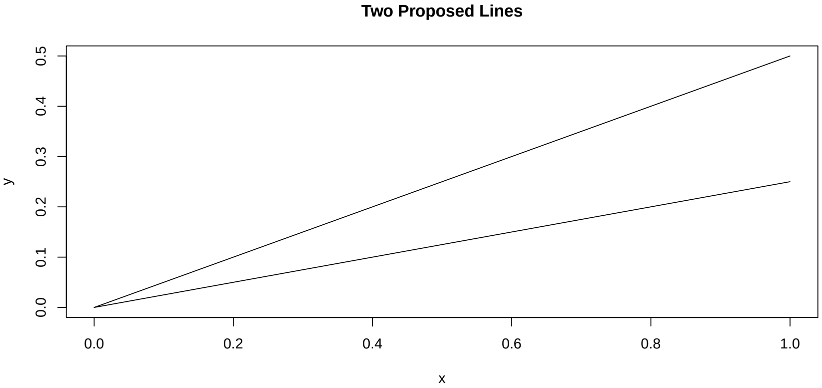
Modified Objective

$$q = \frac{1}{f_D^{1/2p}}$$

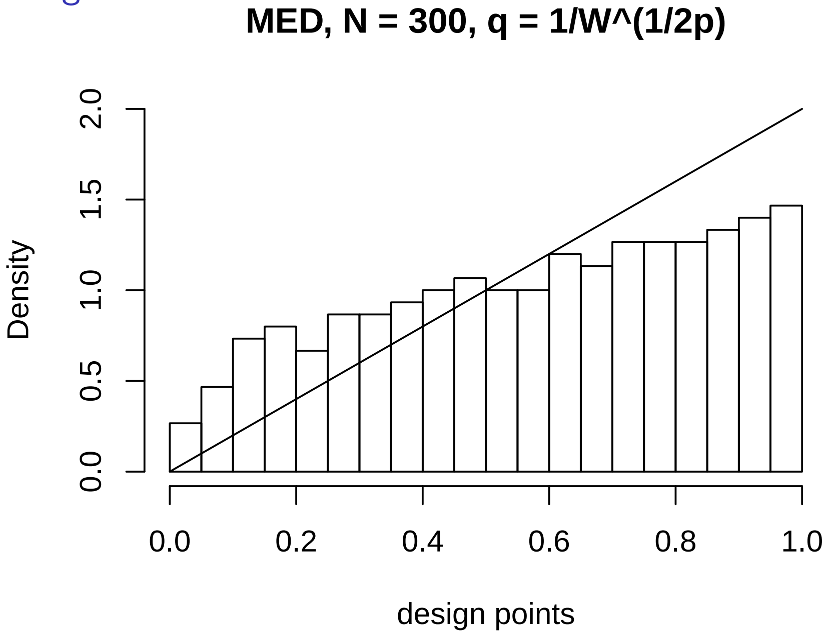
where $f_D(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$,

- Here, the regions that are important for distinguishing the two models have high density.
- A tuning parameter α adjusts the space-filling aspect: $q_\alpha = 1/f_D^{\alpha/2p}$

Original Motivating Example



Limiting Distribution

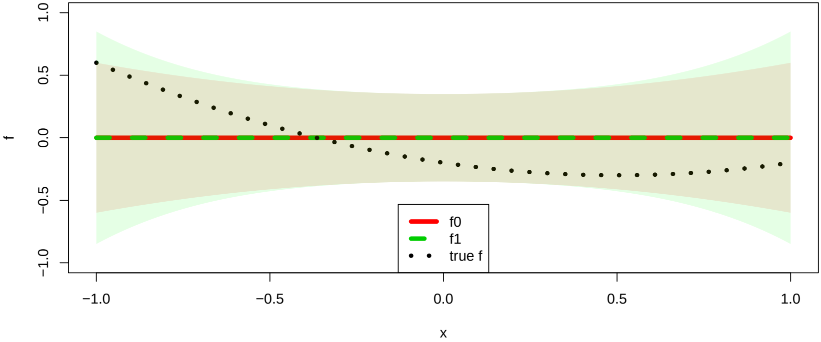


Cautionary Example

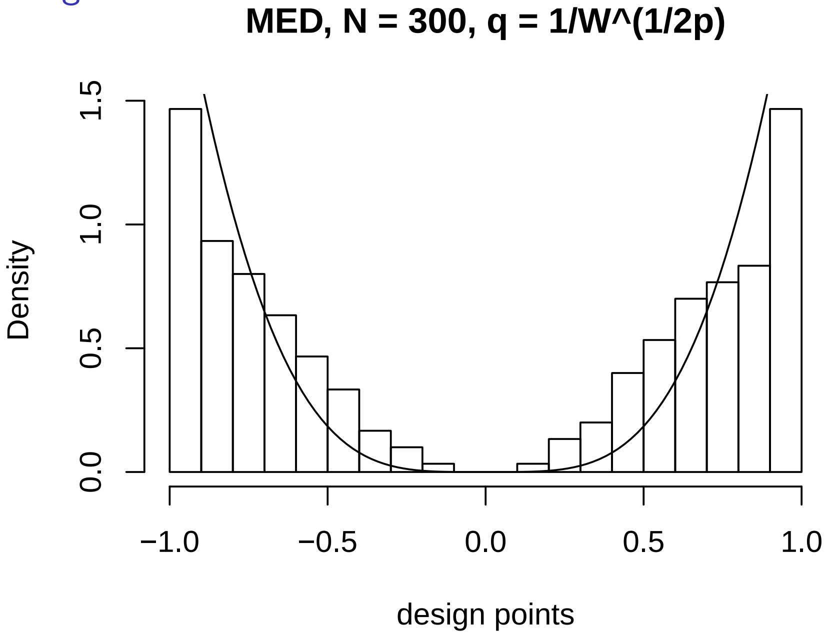
Suppose we want to consider a linear model and quadratic model:

$$H_0 : \beta \sim N((0,0)^T, \nu^2 I_2)$$
$$H_1 : \beta \sim N((0,0,0)^T, \nu^2 I_3)$$

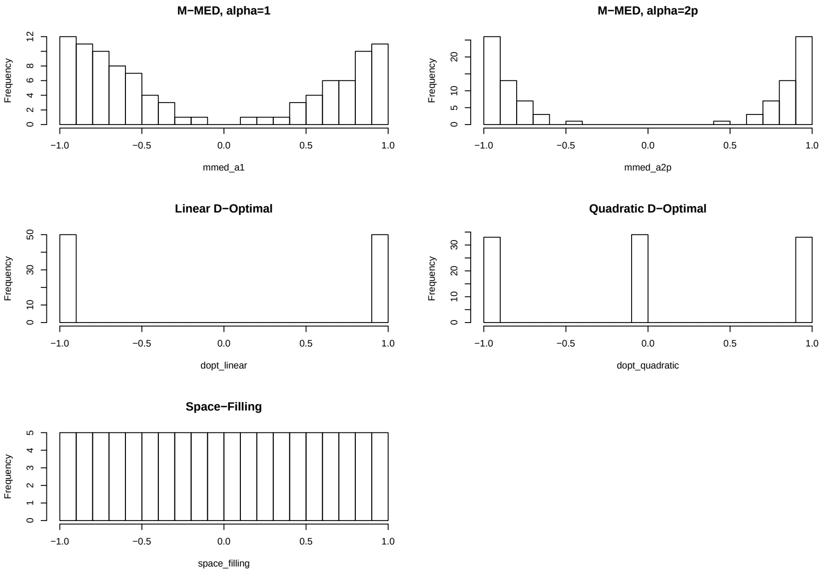
Consider the case where the true model is quadratic:
$\beta_T = (-0.2, -0.4, 0.4)$



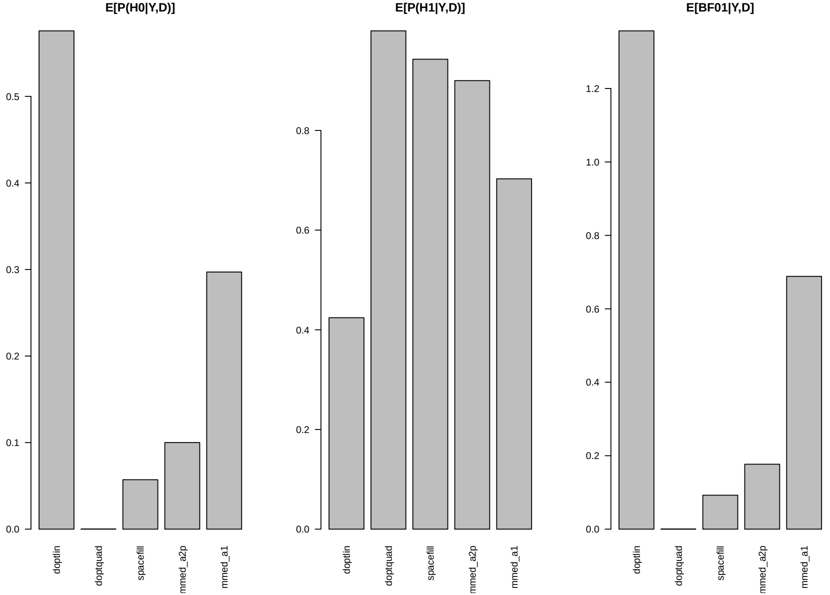
Limiting Distribution



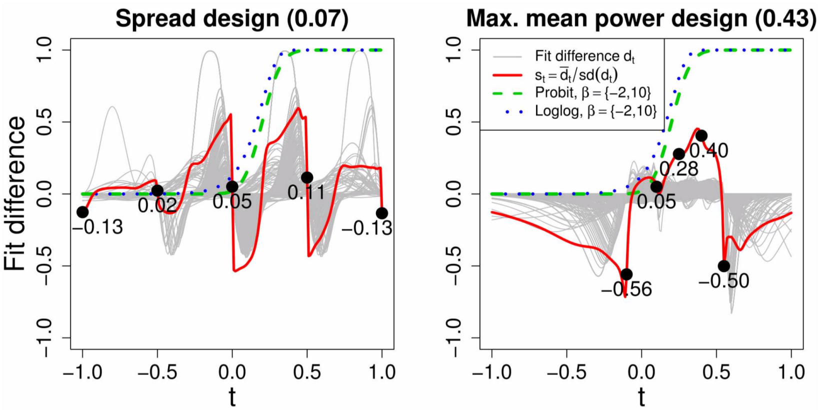
D-Optimal and Space-filling Designs



Posterior Probabilities



Points for Estimation



Points in the middle do not show large difference between the two models, but are importaint for constraining the models to be distinguished[2]

[2] "Designing Test Information and Test Information in Design", Jones & Meng

Sequential Modified MED

Sequential Design

If an experiment setting allows for data to be gathered sequentially, the modified MED (M-MED) can be adjusted to take into account data from previous experiments.

Currently, we have $q_\alpha = 1/f_D^{\alpha/2p}$, where $f_D(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$

- M-MED: $\phi_{\ell,\mathbf{x}}$ is the marginal distribution of $y|H_\ell, X$

Taking data into account

- Sequential M-MED: $\phi_{\ell,\mathbf{x}}$ is the posterior predictive distribution[3] of $y|H_\ell, X$
- In addition, we can sequentially adjust α:
  1. Start the sequence at $\alpha = 0$, a space-filling design, to help determine the models that we would like to select from.
  2. Incrementally adjust α to focus more on distinguishing models, while still allowing some space-filling for robustness.[4]

[3] See Appendix A

[4] See Appendix A for more details

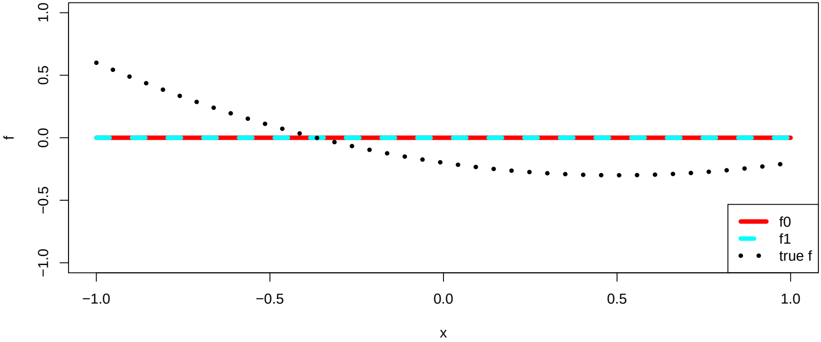
Case 1: Quadratic true model

Hypothesized and True Models

Suppose we want to consider a linear model and quadratic model:

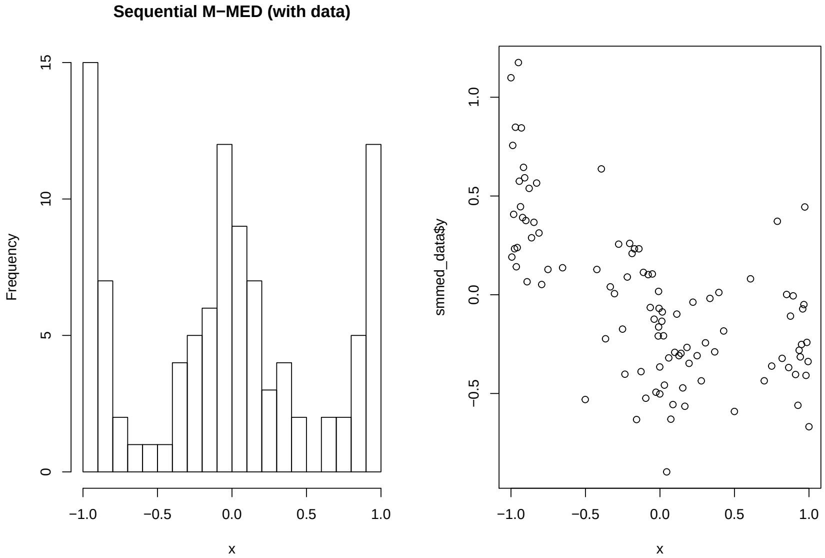
$$H_0 : \beta \sim N((0,0)^T, \nu^2 I_2)$$
$$H_1 : \beta \sim N((0,0,0)^T, \nu^2 I_3)$$

Consider the case where the true model is quadratic:
$\beta_T = (-0.2, -0.4, 0.4)$



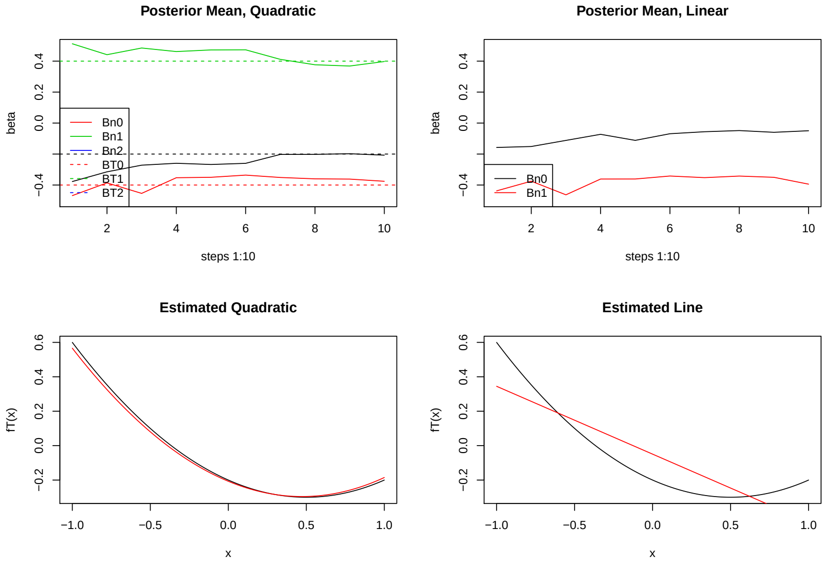
Sequential M-MED (using data)

A sequence of 10 steps, generating 10 points in each step, resulting in 100 points:

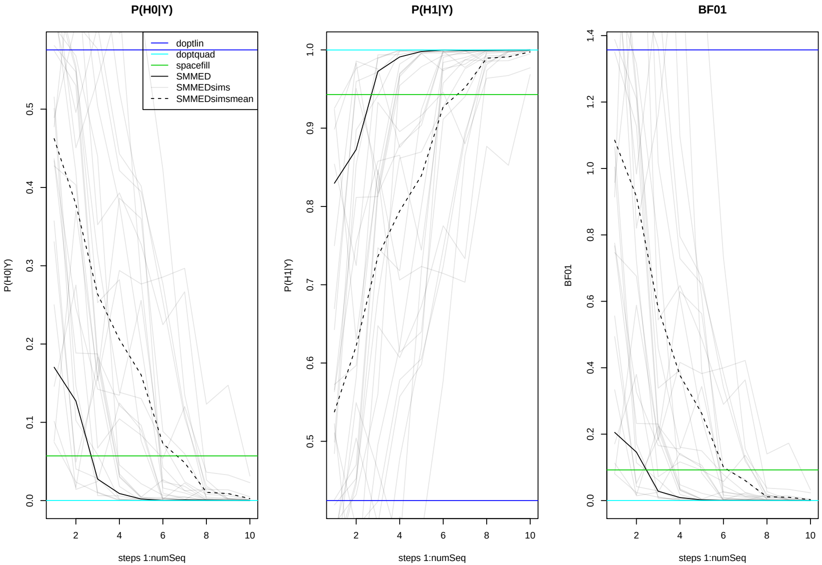


Competing Designs

Linear and Quadratic Fits



Hypothesis Testing



T-Optimal Designs

Comparing linear model with fixed parameters against the quadratic model parameters allowed to vary

```
res
```

```
##
## ################################################################################
## Models:
## [[1]]
## function(x, theta0)
##   theta0[1] + theta0[2] * x
## <bytecode: 0x7fb507b3fd78>
##
## [[2]]
## function(x, theta1)
##   theta1[1] + theta1[2] * x + theta1[3] * x^2
## <bytecode: 0x7fb5078beee0>
##
## Fixed parameters:
## [[1]]
## [1] -0.06492809 -0.39745204
##
## [[2]]
## [1] -0.1988117 -0.3974520  0.3936974
##
##
## ################################################################################
## Design:
##         [,1] [,2]      [,3]
## x -1.0000000  0.0 1.0000000
## w  0.2500026  0.5 0.2499974
##
## ################################################################################
```

T-Optimal Designs

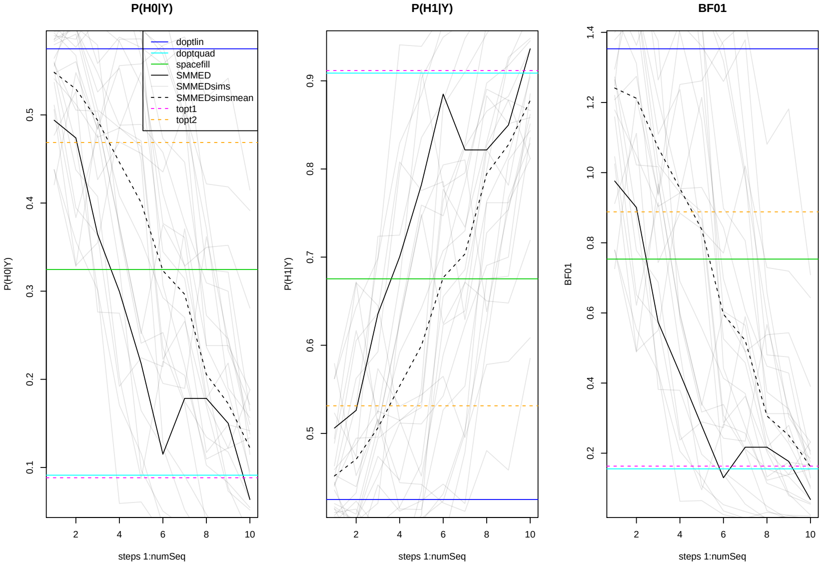
Comparing quadratic model with fixed parameters against the linear model parameters allowed to vary

```
res2
```

```
##
## ################################################################################
## Models:
## [[1]]
## function(x, theta0)
##   theta0[1] + theta0[2] * x
## <bytecode: 0x7fb507b3fd78>
##
## [[2]]
## function(x, theta1)
##   theta1[1] + theta1[2] * x + theta1[3] * x^2
## <bytecode: 0x7fb5078beee0>
##
## Fixed parameters:
## [[1]]
## [1] -0.06492809 -0.39745204
##
## [[2]]
## [1] -0.1988117 -0.3974520  0.3936974
##
##
## ################################################################################
## Design:
##         [,1]      [,2]
## x -1.0000000 1.0000000
## w  0.6516207 0.3483793
##
## ################################################################################
```

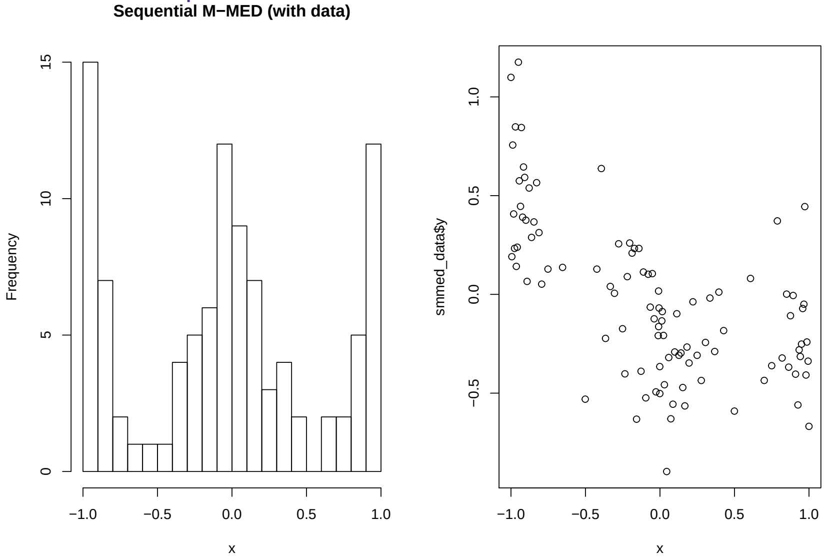
E[P(Hi|Y,D)] with T-Optimal Designs

`sigmasq01 = 0.3` instead of `sigmasq01 = 0.1` for clarity

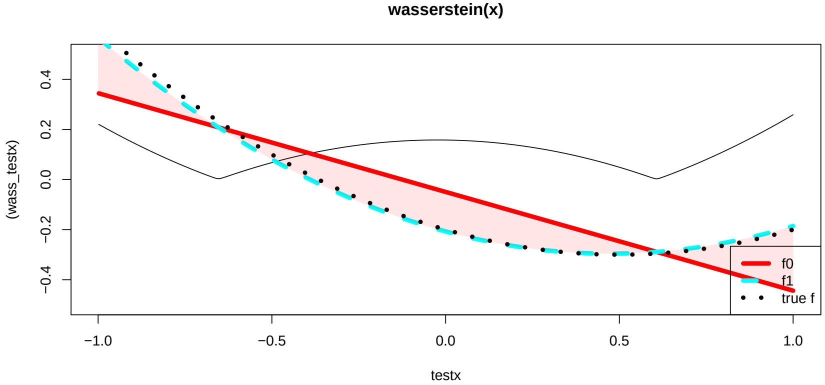


Understanding SMMED

Recall the Sequential MED

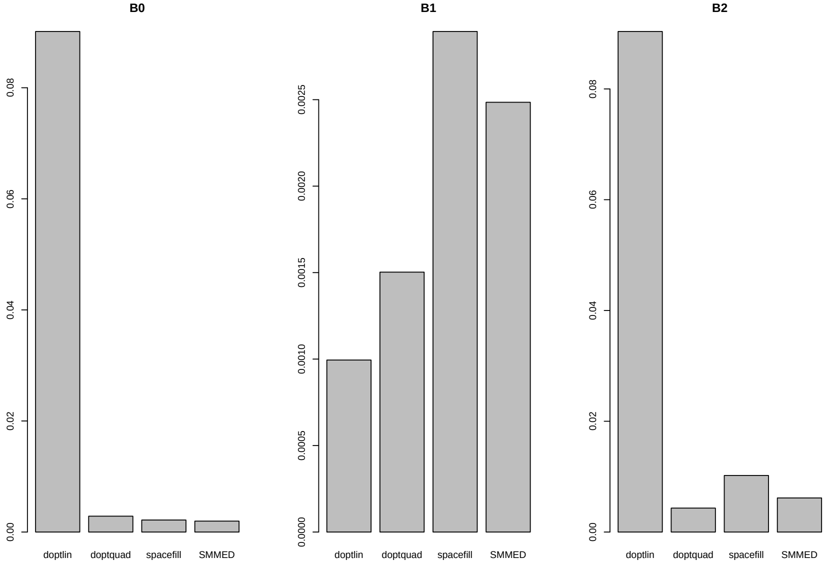


High Density Areas

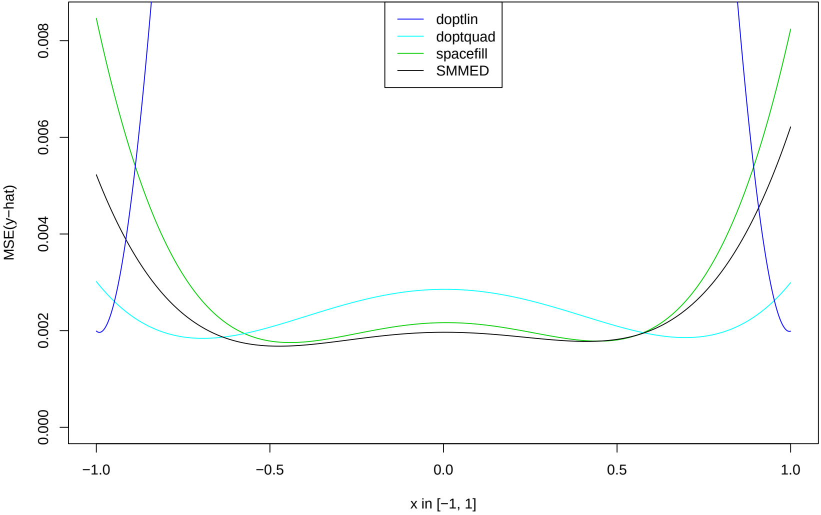


- Since both the linear and quadratic models are trying to capture the data from sequential experiment design, they will intersect in such a way that regions near $-1, 0, 1$ are given high density.
- Why not use quadratic D-optimal design?
  - D-optimal designs are not robust to model misspecification.

Parameter Estimation: MSE(Bn)



Prediction: MSE(y-hat)



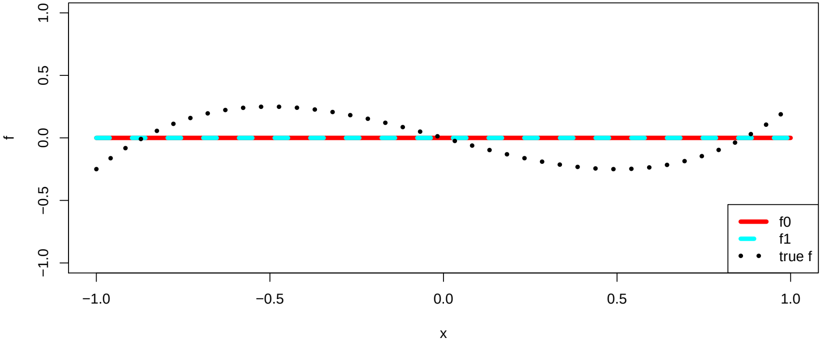
Case 2: Cubic

f0, f1, true f

Suppose we want to consider a linear model and quadratic model:

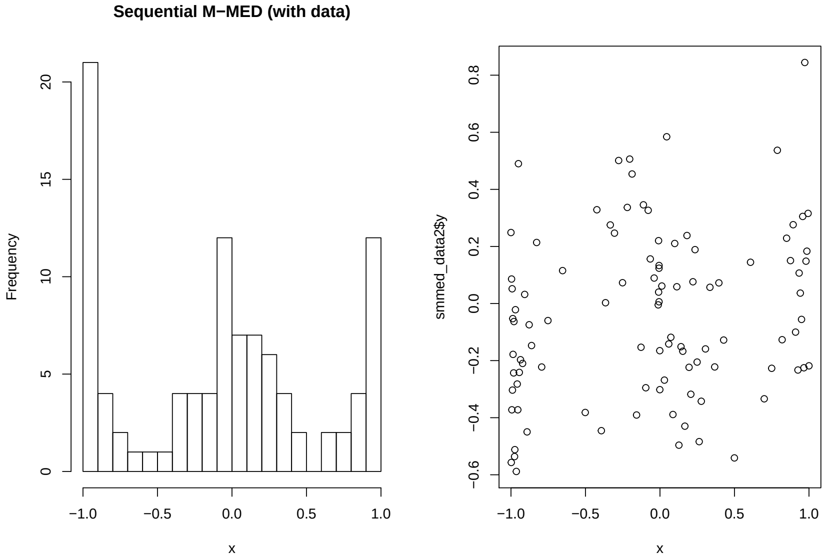
$$H_0 : \beta \sim N((0,0)^T, V_0)$$
$$H_1 : \beta \sim N((0,0,0)^T, V_0)$$

and suppose $\beta_T = (0, -0.75, 0, 1)$

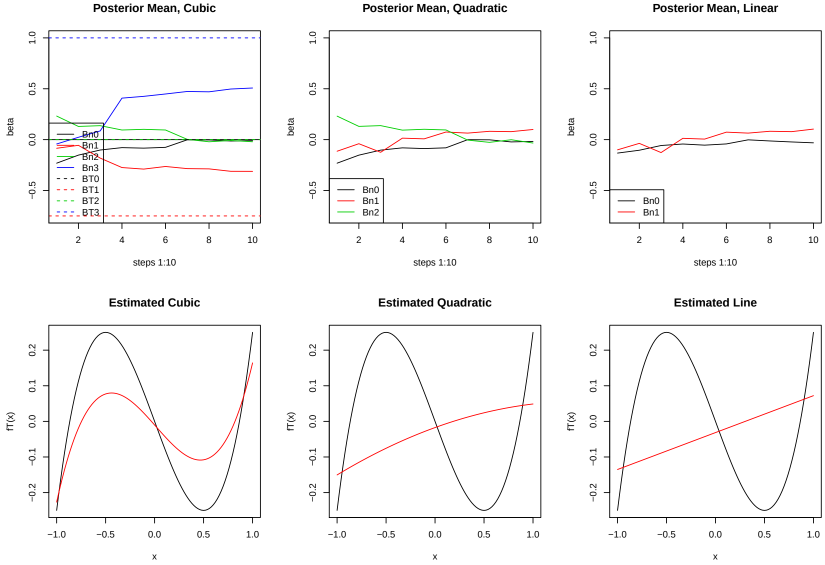


Sequential M-MED With Data

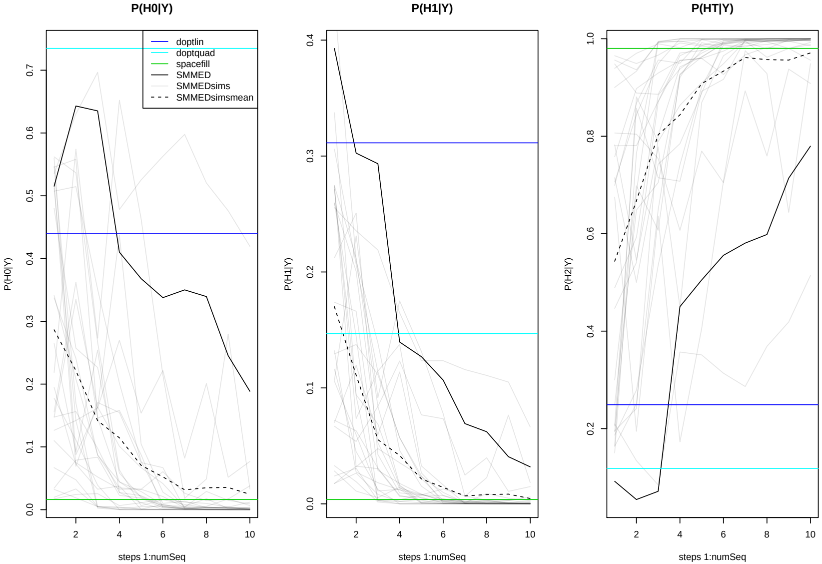
A sequence of 10 steps, generating 10 points in each step, resulting in 100 points:



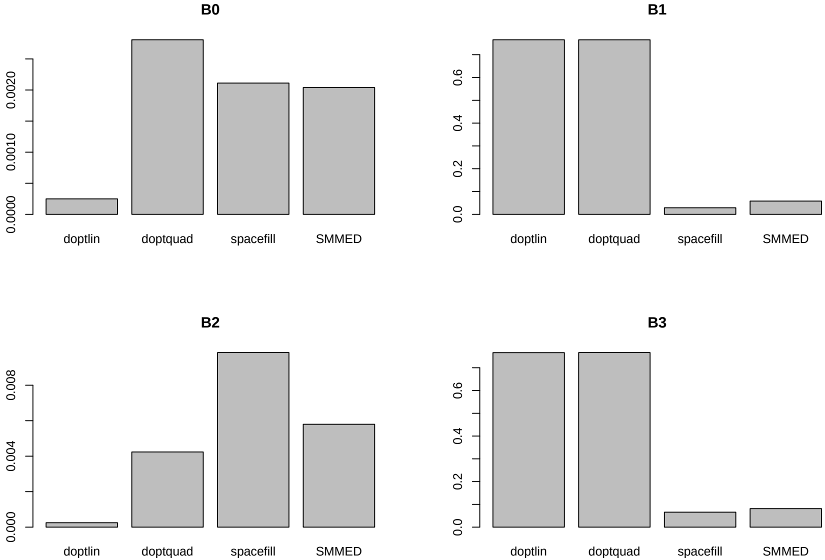
Linear, Quadratic, Cubic Fits



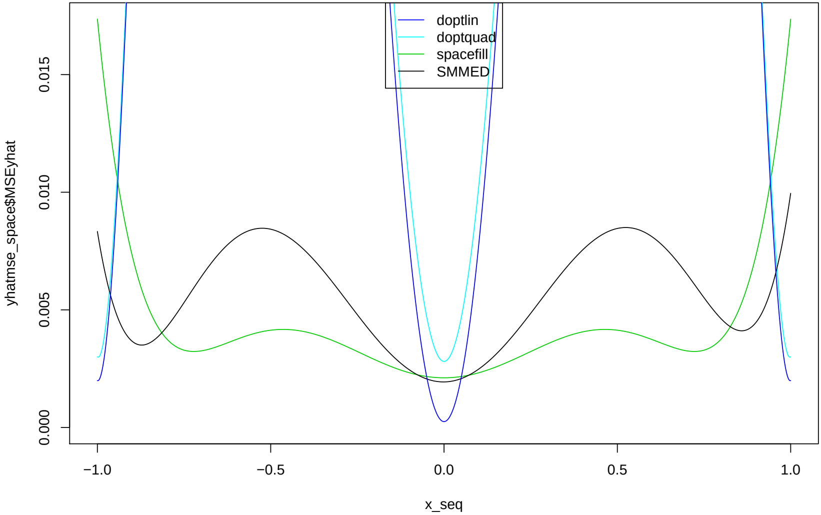
Hypothesis Testing



Parameter Estimation: MSE(Bn)



Prediction: MSE(y-hat)



Gaussian Process Application

Applying MED to Gaussian Process Model Selection

- When there are two Gaussian Process Models that can be used to estimate a function, e.g. Matern vs. Squared Exponential covariance functions[5]
  - Squared Exponential: infinitely differentiable, standard choice
  - Matern: more reasonable smoothness assumptions

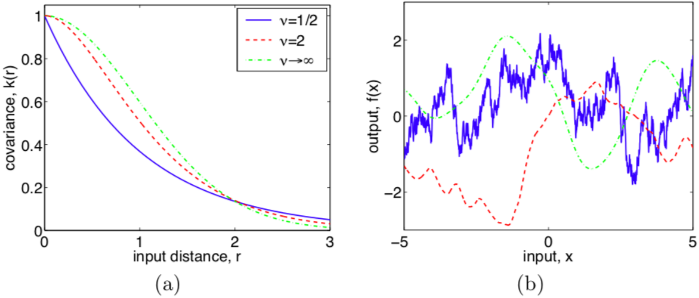


Figure 4.1: Panel (a): covariance functions, and (b): random functions drawn from Gaussian processes with Matérn covariance functions, eq. (4.14), for different values of ν, with $\ell = 1$. The sample functions on the right were obtained using a discretization of the x-axis of 2000 equally-spaced points.

[5] "Gaussian Processes for Machine Learning" Rasmussen et. al. 2005

Applying MED to Gaussian Process Model Selection

- Goal: Choose a design that will distinguish the two gaussian process models.
- Distinguishing functions vs. distributions over functions:
  - For regression models, we use $f_D(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$. What is the distance function now? What are $\phi_{0,\mathbf{x}}, \phi_{0,\mathbf{x}}$?
  - Key Question: Do we need to consider the predictive distribution for each GP model?
    - Doing so would give us an option for $\phi_{0,\mathbf{x}}, \phi_{0,\mathbf{x}}$.
    - However, we will need data (and possibly need to choose new points one at a time).

One-at-a-Time Algorithm (2015) Review

Steps to obtain MED using One-at-a-Time algorithm:

1. Obtain *numCandidates* candidate points, $\mathbf{x}$, in $[0, 1]$ to form candidate set C.
2. Initialize D_N by choosing $\mathbf{x}_1$ to be the candidate $\mathbf{x}$ which optimizes f, where $f(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$ and

$$\phi_{0,\mathbf{x}} = N(\mu_0 \mathbf{x}, \sigma_0^2 + \mathbf{x}^2 \nu_0^2),$$
$$\phi_{1,\mathbf{x}} = N(\mu_1 \mathbf{x}, \sigma_1^2 + \mathbf{x}^2 \nu_1^2)$$

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

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

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

One-at-a-Time Algorithm for GP?

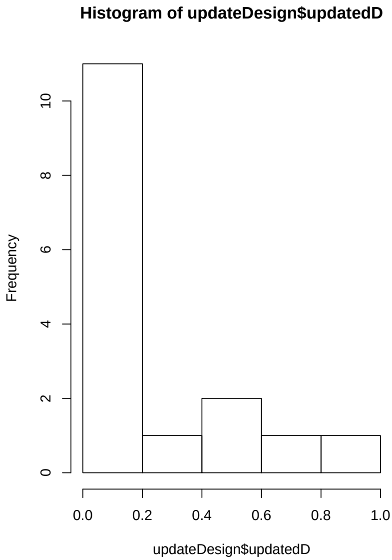
Suppose you have training data $\mathcal{T} = \{(\mathbf{x}_k, y_k)\}_{k=1}^{N_1}$.

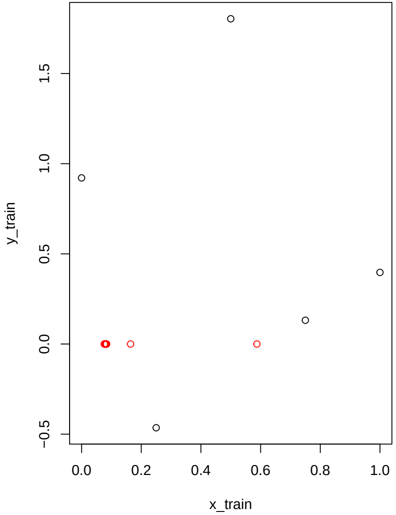
1. Obtain candidate set C
2. Initialize the new set of design points $\mathbf{D}$ as the candidate point $\mathbf{x}_*$ that maximizes $f(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$, where, here, $\phi_{\ell,\mathbf{x}}$ is the predictive distribution $f_*|\mathbf{x}_*, X, f \sim N(k_*^T(K+\tau^2 I)^{-1}Y, k(\mathbf{x},\mathbf{x}) - k_*^T(K+\tau^2 I)^{-1}k_*)$, where $k_* = k(\mathbf{x}, X), K = K(X, X)$, and k and K are determined by the hypothesis ℓ.
3. For subsequent design points, choose:

$$\mathbf{x}_{j+1} = \underset{\mathbf{x}\in C}{\arg\min} \sum_{\mathbf{x}_i\in\mathbf{D}}^{j} \left(\frac{q(\mathbf{x}_i)q(\mathbf{x})}{d(\mathbf{x}_i,\mathbf{x})}\right)^k + \sum_{\mathbf{x}_i\in\mathcal{T}} \left(\frac{q(\mathbf{x}_i)q(\mathbf{x})}{d(\mathbf{x}_i,\mathbf{x})}\right)^k$$

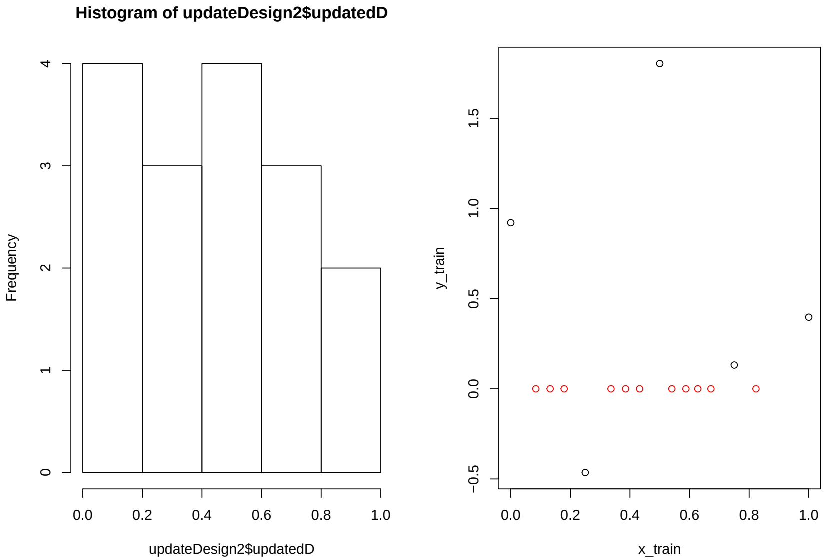
What is the data for previously added design points, $\{(\mathbf{x}_i) | i = 1:j\}$?

Including Data's Points in TPE





Excluding Data's Points in TPE



Appendix A: MED Algorithms

Posterior Predictive Distribution of y

$[\tilde{y}|\tilde{x}, X, y, \sigma_\varepsilon^2, H_i, V_i]$ for brevity, call it $\tilde{y}|y$

$$f(\tilde{y}|y) = \int f(\tilde{y}|\beta; \tilde{x}, \sigma_\varepsilon^2) f(\beta|y, X, V_i, \sigma_\varepsilon^2) d\beta$$

where $f(\tilde{y}|\beta; \tilde{x}, \sigma_\varepsilon^2)$ is the pdf of $N(\tilde{x}^T\beta, \sigma_\varepsilon^2)$ and $f(\beta|y, X, V_i, \sigma_\varepsilon^2)$ is the posterior distribution of β; we denote the posterior mean and variance β_n and Σ_n, respectively.

Integrating out β leads to a normal distribution with mean

$$E[\tilde{y}|y] = E[E[\tilde{y}|\beta, y]] = E[\tilde{x}^T\beta|y] = \tilde{x}^T\beta_n$$

and with variance

$$\begin{aligned} \text{Var}[\tilde{y}|y] &= E[\text{Var}[\tilde{y}|\beta, y]] + \text{Var}[E[\tilde{y}|\beta, y]] \\ &= \sigma_\varepsilon^2 + \text{Var}[\tilde{x}^T\beta|y] = \sigma_\varepsilon^2 + \tilde{x}^T\Sigma_n\tilde{x} \end{aligned}$$

One-at-a-Time Algorithm (2015)

Steps to obtain MED using One-at-a-Time algorithm:

1. Obtain *numCandidates* candidate points, $\mathbf{x}$, in $[0, 1]$.
2. Initialize $\mathbf{D}_N$ by choosing $\mathbf{x}_1$ to be the candidate $\mathbf{x}$ which optimizes f, where $f(\mathbf{x}) = \text{Wasserstein}(\phi_{0,\mathbf{x}}, \phi_{1,\mathbf{x}})$ and

$$\phi_{0,\mathbf{x}} = N(\mu_0 \mathbf{x}, \sigma_0^2 + \mathbf{x}^2 \nu_0^2),$$
$$\phi_{1,\mathbf{x}} = N(\mu_1 \mathbf{x}, \sigma_1^2 + \mathbf{x}^2 \nu_1^2)$$

3. For $j = 1, \ldots, N$, 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 $k = 4p$.

- This is a greedy algorithm for choosing points one at a time

Fast Algorithm (2018)

In each of S 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 $s = 1, \dots, S-1$ stages, obtain each design point $\mathbf{x}_j^{(s+1)} \in \mathbf{D}_{s+1}$ by:

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

where $\gamma_s = s/(S-1)$ and $\mathbf{C}_j^{s+1}$ is the candidate set for $\mathbf{x}_j^{(s+1)}$

- Points migrate to more optimal locations in each stage

Appendix B: Evaluations

Posterior Probabilities of Hypotheses

- Posterior Probability of model $H_\ell, \ell \in 1, ..., M$:

$$P(H_\ell|y, X) = \frac{\pi_\ell f(y|H_\ell, X)}{\sum_{m=1}^{M} \pi_m f(y|H_m, X)}$$

  where π_m is the prior on H_m (typically $\pi_m = \frac{1}{M}$), and $f(y|H_m, X)$ is the model evidence, i.e. density of $N_N(X\mu_\ell, \sigma_\varepsilon^2 I + XV_\ell X^T)$ evaluated at a given y and design **D** with N design points.
- $P(H_\ell|y, X)$ tells which hypothesis is more likely to give the correct model.
- $E[P(H_\ell|y, X)|H_r, X]$ may be estimated using MC approximation from simulated responses y.
- $E[P(H_\ell|y, \mathbf{D})|H_r, \mathbf{D}]$ can be used to evaluate a design **D**'s ability to distinguish hypotheses

Estimate Expected Posterior Probability of a Hypothesis

Estimate the expected posterior probability of hypothesis H_ℓ for J simulations of Y under H_r, given design $\mathbf{D} = \{x_1, ..., x_N\}$:

1. For $j = 1, \ldots, J$:
   1.1 Draw $y_i^{(j)} | \mathbf{x}_i \sim N(\mathbf{x}_i^T \beta_T, \sigma_\varepsilon^2)$, $\forall \mathbf{x}_i \in \mathbf{D}$, so $y^{(j)} \in R^N$.
   1.2 $\forall m = \{0, 1\}$, calculate model evidences $f(y|H_m, \mathbf{D})$
   1.3 Calculate the posterior probability of H_ℓ, $P(H_\ell | y^{(j)}, \mathbf{D})$, from simulation j

$$P(H_\ell | y^{(j)}, \mathbf{D}) = \frac{f(y^{(j)} | H_\ell, X)}{f(y^{(j)} | H_0, X) + f(y^{(j)} | H_1, X)}$$

2. Average the estimated posterior probabilities of H_ℓ over $\forall j$ to obtain MC estimate of $E[P(H_\ell | y, \mathbf{D}) | H_r, \mathbf{D}]$

Note that $y^{(j)}$ are generated from $N_N(X\beta_T, \sigma_\varepsilon^2 I)$ and are independent, while the model evidence for H_m marginalizes out β and evaluates $y^{(j)}$ using $f(y|H_m, \mathbf{D})$, the density of $N_N(X\mu_m, \sigma_\varepsilon^2 I + XV_m X^T)$, in which they are no longer assumed to be independent.

Closed Form MSE of Posterior Mean

For notation, call $E[\beta|Y] = \beta_n$.

$$\begin{aligned} MSE(\beta_n) &= Var[\beta_n] + (E[\beta_n] - \beta_T)^2 \\ &= Var[\beta_n] + (E[\beta_n])^2 - 2\beta_T E[\beta_n] + \beta_T^2 \end{aligned}$$

where

$$\begin{aligned} Var[\beta_n] &= Var[\frac{1}{\sigma^2}\Sigma_B(X^T y + \sigma^2 V^{-1}\mu)] = Var[\frac{1}{\sigma^2}\Sigma_B X^T y] \\ &= (\frac{1}{\sigma^2})^2 \Sigma_B X^T Var[y] X \Sigma_B = (\frac{1}{\sigma^2})^2 \Sigma_B X^T (\sigma^2 I) X \Sigma_B \\ &= \frac{1}{\sigma^2}\Sigma_B X^T X \Sigma_B \\ E[\beta_n] &= E[\frac{1}{\sigma^2}\Sigma_B(X^T y + \sigma^2 V^{-1}\mu)] = \frac{1}{\sigma^2}\Sigma_B(X^T E[y] + \sigma^2 V^{-1}\mu) \\ &= \frac{1}{\sigma^2}\Sigma_B(X^T X\beta_T + \sigma^2 V^{-1}\mu) = \frac{1}{\sigma^2}\Sigma_B X^T X \beta_T + \Sigma_B V^{-1}\mu \end{aligned}$$

where $\Sigma_B = Var[\beta|y] = \sigma^2(X^T X + \sigma^2 V^{-1})^{-1}$ and
$y \sim N(X\beta_T, \sigma^2 I)$

Closed Form MSE of y-hat

For an unseen point $\mathbf{x}_*$, its predicted response $\hat{y} = \mathbf{x}_*^T \beta_n$, where β_n is the posterior mean of β.

$$
\begin{aligned}
MSE(\hat{y}) &= Var[\hat{y}] + Bias^2(\hat{y}) \\
&= Var[\mathbf{x}_*^T \beta_n] + E[\hat{y} - y_T]^2 \\
&= \mathbf{x}_*^T Var[\beta_n]\mathbf{x}_* + E[\mathbf{x}_*^T \beta_n] - \mathbf{x}_*^T \beta_T \\
&= \mathbf{x}_*^T Var[\beta_n]\mathbf{x}_* + \mathbf{x}_*^T E[\beta_n] - \mathbf{x}_*^T \beta_T
\end{aligned}
$$

where $E[\beta_n]$ and $Var[\beta_n]$ were calculated in the previous slide.