MED for Model Selection : Slope

Kristyn Pantoja

5/30/2019

MED for Simple Linear Regression Model Selection

The Set-Up

- Assume $y_i = x_i\beta + \varepsilon_i$ with $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$ and $\beta \sim N(\tilde{\beta}, \sigma_\beta^2)$.
- $y_i|\beta \sim N(x_i\beta, \sigma_\varepsilon^2)$
- $y_i \sim N(x_i\tilde{\beta}, \sigma_\varepsilon^2 + x_i^2\sigma_\beta^2)$ after marginalizing out β (iterated expectation and variance)
- Here, we are assuming that the intercept is 0 (or known, in which case we can scale from 0).

Hypotheses

Suppose we suspect $\beta = \tilde{\beta}_0$ or $\beta = \tilde{\beta}_1$, i.e.

$$H_0 : \beta \sim N(\tilde{\beta}_0, \sigma_{\beta_0}^2)$$
$$H_1 : \beta \sim N(\tilde{\beta}_1, \sigma_{\beta_1}^2)$$

Goals

- Want to choose design $\mathbf{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ to gather data that will
  1. help distinguish these two hypotheses
  2. allow adequate estimation of β.

Evaluating the Designs

Evaluating Methods

- Regression Variance, i.e. Variance of the slope estimator, $\hat{\beta}$
- Expected Posterior Probabilities of H_ℓ, $\ell \in \{0, 1\}$ and Expected Bayes Factor
- Design Criteria:
  - Total Potential Energy
  - Criterion for One-at-a-Time Algorithm
  - Criterion for Fast Algorithm

Interpretations

- A design that is better for estimating β might have smaller regression variance.
- A design that is better for hypothesis testing will give larger expected values of BF_{01} for simulated data Y under H_0.

Regression Variance

In the case where $y_i = x_i\beta + \varepsilon_i$ (with a fixed β) with $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, we have:

$$\hat{\beta} \sim N\left(\beta, \frac{\sigma_\varepsilon^2}{\sum_{i=1}^N (x_i - \bar{x})^2}\right)$$

When $\beta \sim N(\tilde{\beta}, \sigma_\beta^2)$, we have:

$$\hat{\beta} \sim N\left(\beta, \frac{\sigma_\varepsilon^2}{\sum_{i=1}^N (x_i - \bar{x})^2} + \sigma_\beta^2 \frac{\sum_{i=1}^N x_i (x_i^2 - \bar{x})^2}{[\sum_{j=1}^N (x_j - \bar{x})^2]^2}\right)$$

Regression Variance (Derivation)

- Note that $\hat{\beta} = \frac{\sum_i (y_i - \bar{y})(x_i - \bar{x})}{\sum_i (x_i - \bar{x})} = \sum_i w_i y_i$, where $w_i = \frac{(x_i - \bar{x})}{\sum_j (x_j - \bar{x})^2}$.
- Hence, in the case where β is fixed:
  - $y_i \sim N(x_i \beta, \sigma_\varepsilon^2)$. Hence,

$Var[\hat{\beta}] = Var[\sum_i w_i y_i] = \sum_i w_i^2 Var[y_i] = \sum_i w_i^2 \sigma_\varepsilon^2 = \sigma_\varepsilon^2 \sum w_i^2 = \frac{\sigma_\varepsilon^2}{\sum_i (x_i - \bar{x})^2}$.

- It is similar when β is not fixed, i.e. $\beta \sim N(\tilde{\beta}, \sigma_\beta^2)$:
  - $y_i \sim N(x_i \tilde{\beta}, \sigma_\varepsilon^2 + x_i^2 \sigma_\beta^2)$ after marginalizing out β. Hence,

$Var[\hat{\beta}] = Var[\sum_i w_i y_i] = \sum_i w_i^2 Var[y_i] = \sum_i w_i^2 (\sigma_\varepsilon^2 + x_i^2 \sigma_\beta^2) = \frac{\sigma_\varepsilon^2}{\sum_i (x_i - \bar{x})^2} + \sigma_\beta^2 \frac{\sum_i x_i^2 (x_i - \bar{x})^2}{[\sum_j (x_j - \bar{x})^2]^2}$

Posterior Probabilities of Hypotheses and Bayes Factors

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

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

  where π_m is the prior on H_m (typically $\pi_m = \frac{1}{M}$), and $P(Y|H_m)$ is the model evidence.
- The posterior probability of hypotheses tells which hypothesis is more likely to give the correct model.
- If we have only 2 hypotheses, i.e. $M = 2$, we can also calculate the Bayes Factor, $BF_{01} = \frac{P(H_0|Y)}{P(H_1|Y)}$.

Expected Posterior Probabilities of Hypotheses

- We want to calculate the posterior probabilities of our hypotheses given a design, **D**.
- Since we don't have any data Y to calculate the model evidence, instead we estimate the *expected* model evidence $E_Y[P(Y|H_m)]$ from simulations under a chosen hypothesis.

Estimate Expected Posterior Probability of a Hypothesis

Estimate the expected posterior probability of hypothesis H_ℓ for data $Y = \{y_1, \ldots, y_N\}$ simulated under H_r (denoted Y_r):

1. Obtain design $\mathbf{D} = \{x_1, ..., x_N\}$.
2. For J simulations of Y under H_r, draw $y_i^{(j)} \sim N(\tilde{\beta}_r \mathbf{x}, \sigma^2_{\epsilon_r})$, $\forall \mathbf{x}_i \in \mathbf{D}$, \$j = 1, . . . , J.
3. $\forall m \in \{1, ..., M\}$, estimate model evidence $E_Y[P(Y|H_m)] \approx \frac{1}{J}\sum_{j=1}^{J} P(Y_r|H_m, \mathbf{D}) \approx \frac{1}{JN}\sum_{j=1}^{J}\sum_{i=1}^{N} P(y_i^{(j)}|H_m, \mathbf{x}_i)$
   - $P(y_i|H_m, \mathbf{x}_i)$ is the density of $N(\tilde{\beta_m}\mathbf{x}, \sigma^2_{\epsilon_m} + \mathbf{x}^2\sigma^2_{\beta_m})$ evaluated at y_i and $\mathbf{x_i}$.
4. Estimate the posterior probability of H_ℓ: $E_Y[P(H_\ell|Y_r)]$

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

MED 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:

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

which becomes the Total Potential Energy Criterion when $k = 1$.

3. Fast Algorithm tries to minimize:

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

One-at-a-Time Algorithm

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 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

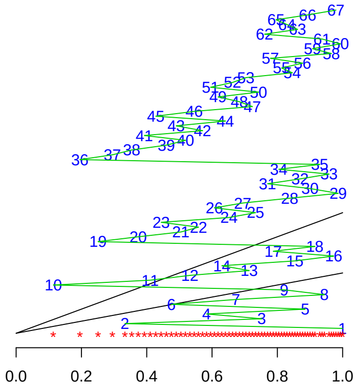
$$\phi_{0,\mathbf{x}} = N(\tilde{\beta}_0\mathbf{x}, \sigma^2_{\epsilon_0} + \mathbf{x}^2\sigma^2_{\beta_0}),$$
$$\phi_{1,\mathbf{x}} = N(\tilde{\beta}_1\mathbf{x}, \sigma^2_{\epsilon_1} + \mathbf{x}^2\sigma^2_{\beta_1})$$

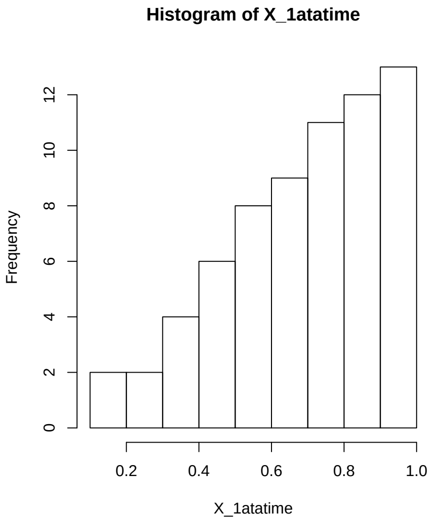
3. Choose the next point $\mathbf{x}_{j+1}$ by:

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

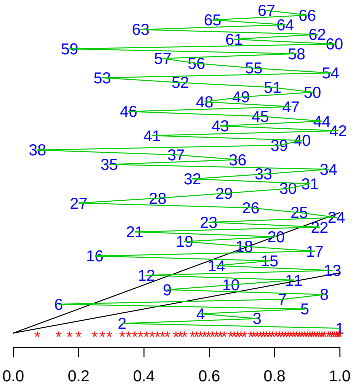
where $q = 1/f^{(1/2p)}$, $d(x, y)$ is Euclidean distance and (suggested) $k = 4p$.

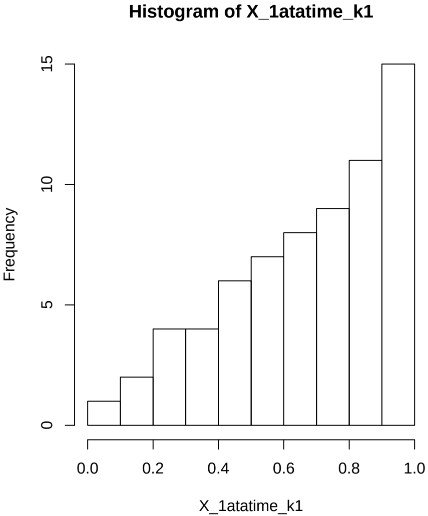
Design generated by One-at-a-Time Algorithm (k = 4)





Design generated by One-at-a-Time Algorithm (k = 1)





Fast Algorithm

Fast Algorithm (2018)

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

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

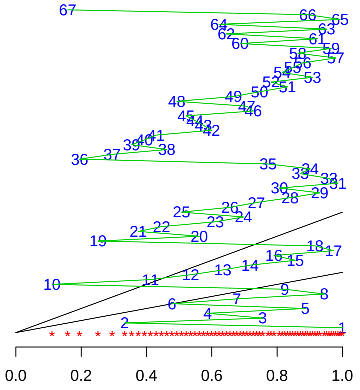
1. Initialize space-filling design $\mathbf{D}_1 = \{\mathbf{x}_1^{(1)} \ldots \mathbf{x}_N^{(1)}\}$
2. For $k = 1, \ldots, K-1$ steps, obtain each design point $\mathbf{x}_j^{(k+1)}$ of the next stage $\mathbf{D}_{k+1}$ by:

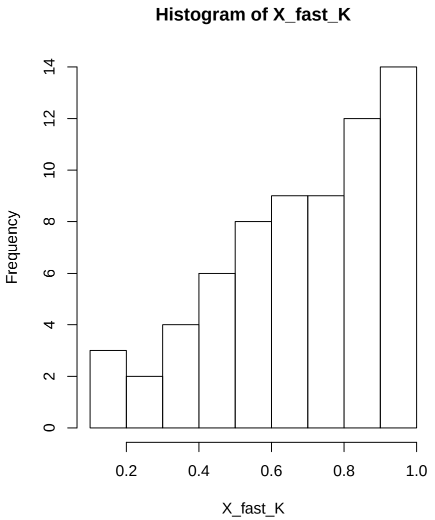
$$\begin{aligned}
\mathbf{x}_j^{k+1} &= \underset{\mathbf{x} \in \mathbf{C}_j^{k+1}}{\arg\min} \max_{i=1:(j-1)} \frac{1}{f^{\gamma_k}(\mathbf{x}_i) f^{\gamma_k}(\mathbf{x}) d^{(2p)}(\mathbf{x}_i, \mathbf{x})} \\
&= \underset{\mathbf{x} \in \mathbf{C}_j^{k+1}}{\arg\min} \max_{i=1:(j-1)} \frac{q^{\gamma_k}(\mathbf{x}_i)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

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)





Other Designs

Random Designs

10 simulated random designs ($\mathbf{x} \sim U([0,1]^p)$, $\forall \mathbf{x} \in \mathbf{D}_{\text{random}}$).

- There is large variability for the criteria in designs with randomly chosen design points.

```
# Mean Slope Variance
v_rand
```

```
## [1] 0.001926288
```

```
# Mean Total PE, Fast Alg Crit, One-at-a-Time Alg Crit
c(TPE_rand, crit1_rand, crit2_rand)
```

```
## [1] 7279551279 7046112509 7046949313
```

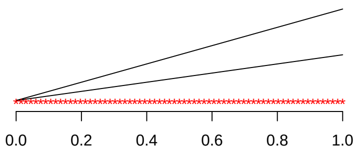
```
# SD Slope Variance
v_rand_sd
```

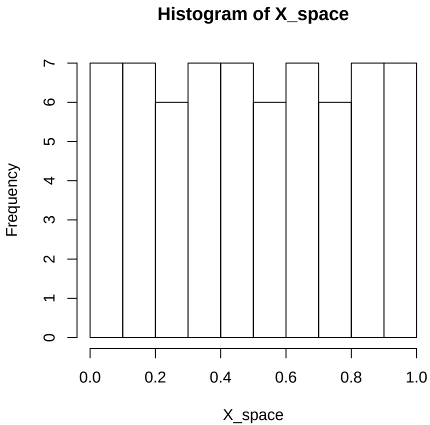
```
## [1] 0.000139415
```

```
# SD Total PE, Fast Alg Crit, One-at-a-Time Alg Crit
c(TPE_rand_sd, crit1_rand_sd, crit2_rand_sd)
```

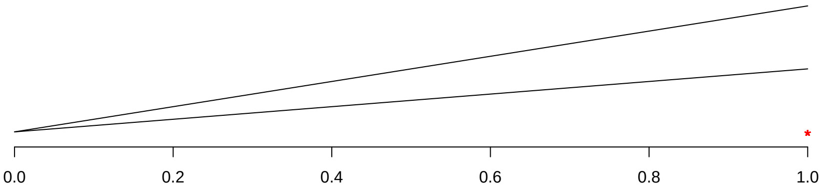
```
## [1] 13290670154 13228511528 13228052751
```

Space-Filling Design





Design at 1



D-Optimal Design

- The D-optimal design seeks to minimize the variance of the estimated regression coefficients (i.e. maximize $\det(X^T X)$, which occurs in the denominator of variance of each of the regression coefficients):

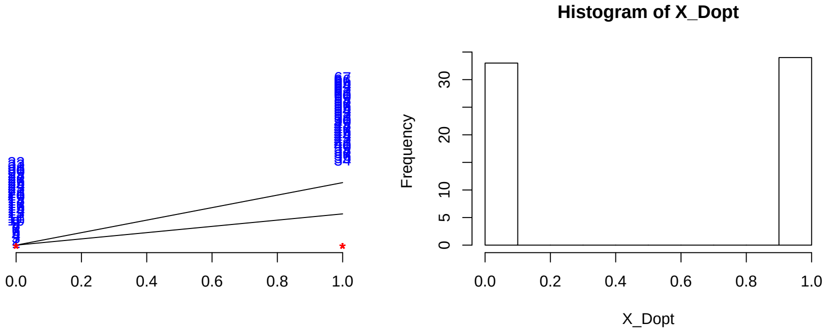
$$\det(X^T X)^{-1}) = 1/\det(X^T X)$$

  where X is the data matrix of independent variables.
- It is considered a sequential algorithm, since one can specify fixed points in the design while choosing the rest of the design points.
- It can be interpreted as minimizing the volume of the confidence ellipsoid of the regression estimates of the linear model parameters.
- It is model-dependent, which raises the question of robustness. Here, the model is assumed to be $y = \beta_0 + \beta_1 x_i$ hence why the points are approximately evenly split between 0 and 1 (since $X = [1\ D]$, i.e. $f(X) = (1, x)^T$ in the literature).

Design generated by D-Optimal Criterion

Using `AlgDesign` package (using Federov's exchange algorithm),

where the points are in no particular order. It is assumed that they will be randomized.




```r
mean(X_Dopt)
```

```
## [1] 0.5074578
```

```r
sd(X_Dopt)
```

```
## [1] 0.5035538
```

I-Optimal Design

- I-Optimal design seeks to minimize the average prediction variance over the entire design space. (In contrast, D-optimality focuses on reducing prediction variance at the design points.)
- The criterion:

$$\int_{\chi} f(x)^T (X^T X)^{-1} f(x)\, dx = \mathrm{tr}((X^T X)^{-1}) M$$

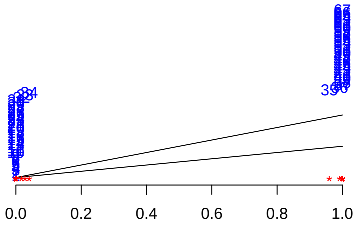
  where $M = \int_{\chi} f(x)^T f(x)\, dx$ and where row vector $f(x)^T$ consists of a 1 followed by the effects corresponding to the assumed model: here, $f(x)^T = (1, x)$.
- This can be approximated (and scaled) by

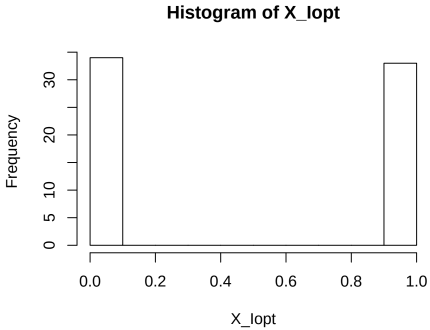
$$\frac{1}{M} \sum_{i \in \mathbf{C}} \mathbf{x}_i^T \frac{(X^T X)^{-1}}{N} \mathbf{x}_i$$

  where M is the number of candidate points in $\mathbf{C}$ and N is the number of design points in $\mathbf{D}$.
- It is more naturally applied when you know the form of the model and want "good" prediction over your design space
- I-Optimal design is also invariant under linear transformations

Design generated by I-Optimal Criterion

Using `AlgDesign` package (using Federov's exchange algorithm),






```
mean(X_Iopt)
```

```
## [1] 0.4931015
```

```
sd(X_Iopt)
```

```
## [1] 0.5016174
```

The Table

Results!

	Fast	1atT,k=4	1atT,k=1	Random	Space	X = 1	D-opt	I-opt
E[H0 | Y0]	0.875	0.883	0.861	0.732	0.737	0.998	0.67	0.67
E[H1 | Y0]	0.125	0.117	0.139	0.268	0.263	0.00181	0.33	0.33
E[BF01 | Y0]	7	7.58	6.18	2.77	2.8	550	2.03	2.03
E[H0 | Y1]	0.123	0.115	0.138	0.269	0.262	0.00193	0.33	0.33
E[H1 | Y1]	0.877	0.885	0.862	0.731	0.738	0.998	0.67	0.67
E[BF01 | Y1]	0.141	0.13	0.16	0.369	0.355	0.00194	0.492	0.492
Var Slope	0.00299	0.0033	0.00259	0.00193	0.00181	NaN	0.000627	0.000633
TPE	2820000	2870000	2760000	7.28e+09	Inf	Inf	Inf	Inf
Fast Crit	44500	43700	80400	7.05e+09	Inf	Inf	Inf	Inf
1atT Crit (k=4)	94100	92200	109000	7.05e+09	Inf	Inf	Inf	Inf
Mean(D)	0.684	0.689	0.674	NA	0.5	1	0.507	0.493
sd(D)	0.23	0.219	0.247	NA	0.295	0	0.504	0.502

- Design at $X = 1$ has highest expected Bayes Factor, and hence is best for testing hypotheses on slope.
- MEDs (designs from Fast & One-at-a-Time algorithms) have second expected Bayes Factors (when H_0 is true)
- Variances on slope, i.e. $Var[\hat{\beta}]$, are all fairly small except for that of design $X = 1$ which cannot be computed.
- `Inf` for the space-filling and D-optimal designs in the evaluations of each of the 3 criteria are from including `0`, which has gives as Wasserstein distance of 0 (in the denominator).

More Evaluations (D, De, A, I, Ge)

- D-efficiency, $De = \frac{\det(X^T X)^{(1/p)}}{N}$, is the relative number of runs (expressed as a percent) required by a hypothetical orthogonal design to achieve the same determinant value. It provides a way of comparing designs across different sample sizes.
  - When a design is orthogonal (all parameters can be estimated independently of each other), $De = 1$
  - De is proportional to the criterion of D-Optimal design, which seeks to maximize $\det(X^T X)$ (or minimize $\det((X^T X)^{-1})$). Hence, we want De close to 1.
- The A-Optimal design minimizes the average variance of the estimates of the regression coefficients: $\mathrm{tr}((X^T X)^{-1})/p$.
- The I-Optimal design seeks to minimize the average prediction variance over the design space.
- Ge, or G-efficiency, is available as a standard of design quality. It is good for minimizing the maximum variance of the predicted values.
  - Ge provides a lower bound on De for approximate theory: $De \geq exp(1 - \frac{1}{Ge})$. Hence, the closer to 1, the better.

When the model is given by $f(x) = (x)^T$,

	Fast	1atT,k=4	1atT,k=1	Space	X = 1	D-opt	I-opt
D	34.8	34.9	34.4	22.5	67	34	32.9
De	0.519	0.521	0.514	0.336	1	0.507	0.491
A	1.93	1.92	1.95	2.98	1	1.97	2.04
I	0.642	0.639	0.649	0.992	0.333	0.657	0.679
Ge	0.519	0.521	0.514	0.336	1	0.507	0.491

- Best performances:
  - Highest *De*: $X = 1$ design (MEDs were next best, but D-optimal design was close)
  - Lowest *A*: $X = 1$ design (MEDs were next best)
  - Lowest *I*: $X = 1$ design (MEDs were next best)
  - Highest *Ge*: $X = 1$ design (MEDs were next best)
- Note: Optimal designs were not optimized to this form.

When $f(x) = (1, x)^T$,

	Fast	1atT,k=4	1atT,k=1	Space	$X = 1$	D-opt	I-opt
D	233	212	269	385	0	1120	1110
De	0.228	0.217	0.245	0.293	0	0.5	0.498
A	14.6	16.1	12.6	7.78	NA	3.02	3.01
I	3.25	3.52	2.89	1.97	NA	1.33	1.34
Ge	0.2	0.181	0.233	0.511	NA	0.985	0.982

- Here, MEDs did not do so well, but they weren't optimized for this model.
  - Highest *De*: D-optimal design (MEDs > space-filling design)
  - Lowest *A*: I-optimal design (space-filling > MEDs)
  - Lowest *I*: D-optimal design (space-filling > MEDs)
  - Highest *Ge*: I-optimal design (space-filling > MEDs)
  - This may suggest that MED might not be very robust...
- Note: `NA`s for $X = 1$ design are due to invertibility of $X^T X$ for design matrix $X = (\mathbf{I}_N \ \mathbf{I}_N)$

The Set-Up with Unknown Intercept

The set-up is similar, but there are some slight differences:

- Assume $y_i = \beta_0 + x_i\beta_1 + \varepsilon_i$ with $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$ and $\beta \sim N(\tilde{\beta}, \sigma_\beta^2 I)$, where $\tilde{\beta} = (\tilde{\beta}_0, \tilde{\beta}_1)^T$.
- $y_i|\beta \sim N(\beta_0 + x_i\beta_1, \sigma_\varepsilon^2)$
- $y_i \sim N(\tilde{\beta}_0 + x_i\tilde{\beta}_1, \sigma_\varepsilon^2 + (x_i^2 + 1)\sigma_\beta^2)$ after marginalizing out β (iterated expectation and variance again)

Hypotheses

Suppose we suspect $\beta = \tilde{\beta}^{(0)}$ or $\beta = \tilde{\beta}^{(1)}$, i.e.

$$H_0 : \beta \sim N\left(\tilde{\beta}^{(0)}, \sigma_{\beta_0}^2 I\right)$$

$$H_1 : \beta \sim N\left(\tilde{\beta}^{(1)}, \sigma_{\beta_1}^2 I\right)$$

where $\tilde{\beta}^{(0)} = (\tilde{\beta}_0^{(0)}, \tilde{\beta}_1^{(0)})^T$ and $\tilde{\beta}^{(1)} = (\tilde{\beta}_0^{(1)}, \tilde{\beta}_1^{(1)})^T$.

Regression Variance

In the case where $y_i = x_i\beta + \varepsilon_i$ (with a fixed β) with $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, we have:

$$\hat{\beta}_0 \sim N\left(\beta_0, \sigma_\varepsilon^2 \frac{\sum_{i=1}^N x_i^2}{N\sum_{i=1}^N (x_i - \bar{x})^2}\right)$$

$$\hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma_\varepsilon^2}{\sum_{i=1}^N (x_i - \bar{x})^2}\right)$$

But when β is given a prior, i.e. $\beta \sim N(\tilde{\beta}, \sigma_\beta^2 I)$:

$$\hat{\beta}_0 \sim N\left(\beta_0, \frac{\sigma_\varepsilon^2 + (x_i^2 + 1)\sigma_\beta^2}{N} + Var[\hat{\beta}_1]\right)$$

$$\hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma_\varepsilon^2}{\sum_i (x_i - \bar{x})^2} + \sigma_\beta^2 \frac{\sum_i (x_i + 1)^2 (x_i - \bar{x})^2}{[\sum_j (x_j - \bar{x})^2]^2}\right)$$

Regression Variance (Derivation)

- Note that:
  - $\hat{\beta}_1 = \frac{\sum_i (y_i - \bar{y})(x_i - \bar{x})}{\sum_i (x_i - \bar{x})} = \sum_i w_i y_i$, where $w_i = \frac{(x_i - \bar{x})}{\sum_j (x_j - \bar{x})^2}$.
  - $\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{x} = \frac{1}{N} \sum_i y_i - \bar{x} \sum_i w_i y_i = \sum_i (\frac{1}{N} - \bar{x} w_i) y_i$.
- Hence, in the case where β is fixed:
  - $y_i \sim N(\beta_0 + x_i \beta_1, \sigma_\varepsilon^2)$.
  - $Var[\hat{\beta}_1]$ is as previously derived, and

$$Var[\hat{\beta}_0] = Var[\textstyle\sum_i (\frac{1}{N} - \bar{x} w_i) y_i] = \sum_i (\frac{1}{N} - \bar{x} w_i)^2 Var[y_i] = \sum_i (\frac{1}{N} - \bar{x} w_i)^2 \sigma_\varepsilon^2 = \cdots = \sigma_\varepsilon^2 \frac{\sum_{i=1}^N x_i^2}{N \sum_{i=1}^N (x_i - \bar{x})^2}$$

Regression Variance (Derivation, continued.)

- When β is not fixed, i.e. $\beta \sim N(\tilde{\beta}, \sigma_\beta^2 I)$:
  - $y_i \sim N(\tilde{\beta}_0 + x_i \tilde{\beta}_1, \sigma_\varepsilon^2 + (x_i^2 + 1)\sigma_\beta^2)$. Hence,

$$Var[\hat{\beta}_1] = Var[\textstyle\sum_i w_i y_i] = \sum_i w_i^2 Var[y_i] =$$
$$\textstyle\sum_i w_i^2(\sigma_\varepsilon^2 + (x_i^2 + 1)\sigma_\beta^2) = \frac{\sigma_\varepsilon^2}{\sum_i (x_i - \bar{x})^2} + \sigma_\beta^2 \frac{\sum_i (x_i+1)^2 (x_i - \bar{x})^2}{[\sum_j (x_j - \bar{x})^2]^2}$$

so only slightly different!

$$Var[\hat{\beta}_0] = Var[\textstyle\sum_i (\frac{1}{N} - \bar{x} w_i) y_i] = \sum_i (\frac{1}{N} - \bar{x} w_i)^2 Var[y_i] =$$
$$\textstyle\sum_i (\frac{1}{N} - \bar{x} w_i)^2 Var[y_i] = \sum_i (\frac{1}{N^2} - \frac{2}{N}\bar{x} w_i + \bar{x}^2 w_i^2) Var[y_i] = \dots$$

Notice that $\sum_i w_i = \sum_i \frac{(x_i - \bar{x})}{\sum_j (x_j - \bar{x})^2} = 0$. Hence,

$$\dots = \frac{1}{N} Var[y_i] - 0 + \bar{x}^2 \textstyle\sum_i w_i^2 Var[y_i] = \frac{\sigma_\varepsilon^2 + (x_i^2 + 1)\sigma_\beta^2}{N} + Var[\hat{\beta}_1]$$