# Sequential Monte Carlo Algorithms for Bayesian Sequential Design

Dr Chris Drovandi
Queensland University of Technology
c.drovandi@qut.edu.au

Collaborators: James McGree, Tony Pettitt, Gentry White
Acknowledgements: Australian Research Council Discovery Grant and organisers of MCMski

January 6, 2014

# Sequential Experimental Design

- Adaptive decisions as new data are collected
- More robust to parameter and model uncertainty
- Natural to use Bayesian framework. Posterior becomes new prior
- Next decision obtained by looking forward to all future decisions (backward induction)
- Simplified by myopic design (one-at-a-time)
- Next design point $d_{t+1} = \arg\max U(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$. $\boldsymbol{y_{1:t}}$ collected data at design $\boldsymbol{d_{1:t}}$. $U$ is utility function

# Why SMC for Bayesian sequential design?

- Much more efficient than MCMC. Simple re-weighting step to incorporate new information.
- Parallel implementation possible (e.g. GPU)
- Increase in efficiency allows comparisons of utility functions
- Convenient estimation of important Bayesian utility functions (e.g. mutual information)
- Decisions in real time?

# SMC For One static Model $m$

- Sample from sequence of targets
- Data annealing here

$$\pi_t(\boldsymbol{\theta_m}|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = f(\boldsymbol{y_{1:t}}|\boldsymbol{\theta_m}, \boldsymbol{d_{1:t}})\pi(\boldsymbol{\theta_m})/Z_{m,t}, \text{ for } t = 1, \ldots, T.$$

$\boldsymbol{y_{1:t}}$ (independent) data up to $t$, $\boldsymbol{d_{1:t}}$ design points up to $t$, $\boldsymbol{\theta_m}$ parameter for model $m$. $f$ is likelihood, $\pi$ prior, $\pi_t$ posterior

$$f(\boldsymbol{y_{1:t}}|m, \boldsymbol{d_{1:t}}) = Z_{m,t} = \int_{\boldsymbol{\theta_m}} f(\boldsymbol{y_{1:t}}|\boldsymbol{\theta_m}, \boldsymbol{d_{1:t}})\pi(\boldsymbol{\theta_m})d\boldsymbol{\theta_m}.$$

- SMC: Generate a weighted sample (particles) for each target in the sequence via steps
  - Reweight: particles as data comes in (efficient)
  - Resample: when ESS small
  - Mutation: diversify duplicated particles (can be efficient)

# SMC For One STATIC Model $m$ (Algorithm) Chopin (2002)

- Have current particles $\{W_t^i, \boldsymbol{\theta_t^i}\}_{i=1}^N$ based on data $\boldsymbol{y_{1:t}}$
- Re-weight step to included $y_{t+1}$

$$W_{t+1}^i \propto W_t^i f(y_{t+1}|\boldsymbol{\theta_t^i}, d_{t+1}).$$

- Check effective sample size: $\text{ESS} = 1/\sum_{i=1}^N (W_{t+1}^i)^2$
- If ESS $> E$ (e.g. $E = N/2$) go back to re-weight step for next observation
- If ESS $< E$ do the following
- Resample proportional to weights. Duplicates good particles
- Mutation: Move all particles via MCMC kernel say $R$ times (adaptive proposal)

# SMC Estimate of Evidence Del Moral et al (2006)

- It can be shown

$$Z_{t+1}/Z_t = f(y_{t+1}|\boldsymbol{y_{1:t}}, d_{t+1}) = \int_{\boldsymbol{\theta}} f(y_{t+1}|\boldsymbol{\theta}, d_{t+1})\pi(\boldsymbol{\theta}|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})d\boldsymbol{\theta}.$$

- Using SMC particles to approximate posterior at $t$ gives estimator

$$Z_{t+1}/Z_t \approx \sum_{i=1}^{N} W_t^i f(y_{t+1}|\boldsymbol{\theta_t^i}, d_{t+1}).$$

- Can then obtain approximation of $Z_{t+1}$ through

$$\frac{Z_{t+1}}{Z_0} = \frac{Z_{t+1}}{Z_t}\frac{Z_t}{Z_{t-1}}\cdots\frac{Z_1}{Z_0}.$$

- Also gives estimate of posterior predictive probability of $y_{t+1}$

# Advantage 1: Efficiently comparing utilities (Drovandi et al 2013)

- Discrete data (binary) example
- Need to compute utility for all possible $d$ (then for all $t = 1, \ldots, T$)

$$U(d|\boldsymbol{y}_{1:t}, \boldsymbol{d}_{1:t}) = \sum_{z \in \{0,1\}} f(z|\boldsymbol{y}_{1:t}, \boldsymbol{d}_{1:t}, d) U(d, z|\boldsymbol{y}_{1:t}, \boldsymbol{d}_{1:t}).$$

- The whole process requires the computation (sampling) of many many posterior distributions
- SMC (IS) to rescue. Pretend $z$ is the 'next' observation collected at design $d$. Simple re-weight to incorporate this observation.
- Use weighted sample to estimate $U(d, z)$. SMC also provides estimate of posterior predictive.
- There may be different choices for $U(d, z)$, want to compare.
- Need to simulate the design many times.

# The Design Problem

Estimating Maximum tolerated dose, minimum effective dose (clinical trials)

$$E[Y_t] = g^{-1}(\eta_t) \text{ where}$$

$$\eta_t = \theta_0 + \theta_1 \frac{d_t^\lambda - 1}{\lambda},$$

where $d_t$ is the dose assigned to the $t$th subject.
$Y_t \sim Binary(E[Y_t])$. Uninformative prior

$$\pi(\theta_0, \theta_1, \lambda) \propto N(\theta_0; 0, 100) N(\theta_1; 0, 100) U(\lambda; 0, 1) 1\left(\lambda \frac{\eta^* - \theta_0}{\theta_1} + 1 > 0\right),$$

Objective is precise estimation of

$$d^* = \left(\lambda^T \frac{\eta^* - \theta_0^T}{\theta_1^T} + 1\right)^{1/\lambda^T}.$$

Let $\boldsymbol{\theta} = (\theta_0, \theta_1, \lambda)$.

# The Utility Functions

Possible choices for $U(d, z|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$

1. Posterior precision of $d^*$ (Natural choice, but posterior of $d^*$ can be unstable when little information is available)
2. Kullback-Leibler Divergence between prior and posterior for $\boldsymbol{\theta}$
3. Determinant of Posterior Covariance matrix of $\boldsymbol{\theta}$
4. Hybrid utility. Utility 3 for 10 subjects. Utility 1 thereafter.

# Some Results

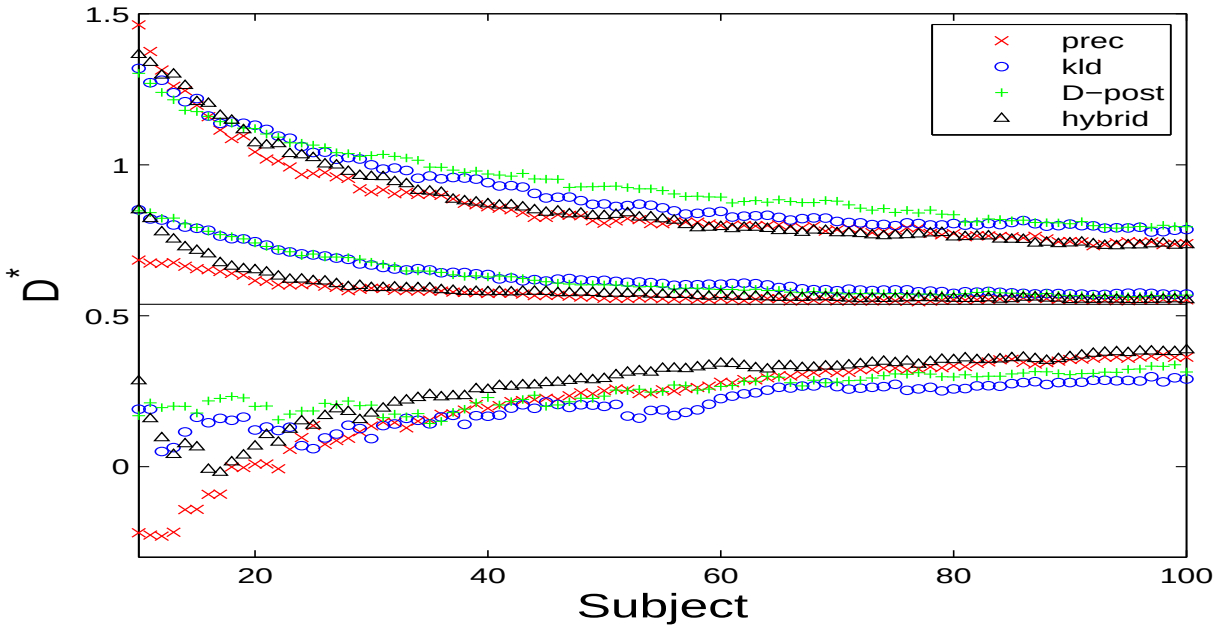


Figure: Distributions of the estimated target stimulus over 10 to 100 subjects for the true parameter configuration of $\boldsymbol{\theta}^{\boldsymbol{T}} = (0,3,1)$ producing $d^* = 0.538$. Solid horizontal line is the true $d^*$. Shown are the 2.5%, 50% and 97.5% quantiles over the 500 runs for each utility function.

# Advantage 2: Estimating Difficult Utilities

- E.g. Mutual Information for Model Discrimination (Drovandi et al 2014)
- Have set of $K$ proposed models $M = 1, \ldots, K$. Select design to maximise ability to discriminate between models
- Consider mutual information between model indicator $M$ and predicted observation $Z$ for $y_{t+1}$ Box and Hill (1967).

$$I(M; Z|\boldsymbol{y_{1:t}}, d) = H(M|\boldsymbol{y_{1:t}}) - H(M|Z; \boldsymbol{y_{1:t}}, d).$$

- Therefore $U(d|\boldsymbol{y_{1:t}}) = -H(M|Z; \boldsymbol{y_{1:t}}, d)$ which is equal to

$$U(d|\boldsymbol{y_{1:t}}) = \sum_{m=1}^{K} \pi(m|\boldsymbol{y_{1:t}}) \int f(z|m, \boldsymbol{y_{1:t}}, d) \log \pi(m|\boldsymbol{y_{1:t}}, z, d) d\mu(z),$$

# SMC for multiple models

- Effectively run an SMC algorithm for each model $m = 1, \ldots, K$
- Have set of $N$ particles for each model $\{W_{m,t}^i, \boldsymbol{\theta}_{\boldsymbol{m,t}}^{\boldsymbol{i}}\}_{i=1}^N$.
- ESS for each model $m$
- resampling and within-model updates when required
- Design part: use data up to $t$, $\boldsymbol{y}_{\boldsymbol{1:t}}$, and particles of all models to compute the next design $d_{t+1}$

# Estimating the Utility

$$U(d|\boldsymbol{y_{1:t}}) = \sum_{m=1}^{K} \pi(m|\boldsymbol{y_{1:t}}) \int f(z|m, \boldsymbol{y_{1:t}}, d) \log \pi(m|\boldsymbol{y_{1:t}}, z, d) d\mu(z),$$

- Borth (1975) notes difficult computation
- SMC to the rescue.
- Potential observation $z$ at potential design point $d$.
- Pretend this the observation for $y_{t+1}$.

# Estimating the Utility (cont...)

- Estimate predictive probability using weights

$$w^i_{m,t}(d,z) = W^i_{m,t} f(z|\boldsymbol{\theta}^{\boldsymbol{i}}_{\boldsymbol{m,t}}, d),$$

$$\hat{f}(z|m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d) = \sum_{i=1}^{N} w^i_{m,t}(d,z).$$

- $Z_{m,t}$ denotes current evidence for model $m$, which integrates out posterior of $\boldsymbol{\theta}$ at $t$.
- Estimate evidence including $(d,z)$ $Z_{m,t}(d,z)$ using

$$\log \hat{Z}_{m,t}(d,z) = \log \hat{Z}_{m,t} + \log \hat{f}(z|m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d).$$

- Convert to $\hat{\pi}(m|\boldsymbol{y_{1:t}}, d, z)$

# Estimating the Utility (cont...)

- Therefore estimate of utility for discrete $z$ is

$$\hat{U}(d|\boldsymbol{y}_{1:t}) = \sum_{m=1}^{K} \hat{\pi}(m|\boldsymbol{y}_{1:t}) \sum_{z \in \mathcal{S}} \hat{f}(z|m, \boldsymbol{y}_{1:t}, d) \log \hat{\pi}(m|\boldsymbol{y}_{1:t}, z, d).$$

- In continuous case approximate integral via MC integration. Draw $z^i_{m,t} \sim f(z|m, \theta^{\mathbf{i}}_{\mathbf{m},\mathbf{t}}, d)$ for $i = 1, \ldots, N$. Weighted sample $\{W^i_{m,t}, \boldsymbol{\theta}^i_{m,t}, z^i_{m,t}\}$ from joint $p(z, \boldsymbol{\theta}|d, m, \mathbf{y_{1:t}}, \mathbf{d_{1:t}})$. Therefore estimate of utility for continuous $z$ is

$$\hat{U}(d|\mathbf{y_{1:t}}, \mathbf{d_{1:t}}) = \sum_{m=1}^{K} \hat{\pi}(m|\mathbf{y_{1:t}}, \mathbf{d_{1:t}}) \sum_{i=1}^{N} W^i_{m,t} \log \hat{\pi}(m|\mathbf{y_{1:t}}, \mathbf{d_{1:t}}, z^i_{m,t}, d).$$

Could also be used for large counts.

# Chemical Engineering Example Masoumi et al 2013

Consider chemical reaction $A \rightarrow B$. Four competing models for the fraction of $A$ remaining after time $t$ minutes at temperature $T$

$$\text{Model 1: } \mu_1 = \exp\left(-\theta_1 t \exp\left(-\frac{\theta_2}{T}\right)\right)$$

$$\text{Model 2: } \mu_2 = \left[1 + \theta_1 t \exp\left(-\frac{\theta_2}{T}\right)\right]^{-1}$$

$$\text{Model 3: } \mu_3 = \left[1 + 2\theta_1 t \exp\left(-\frac{\theta_2}{T}\right)\right]^{-1/2}$$

$$\text{Model 4: } \mu_4 = \left[1 + 3\theta_1 t \exp\left(-\frac{\theta_2}{T}\right)\right]^{-1/3}$$

$y|m \sim N(\mu_m, \sigma^2)$ Two design variables $\boldsymbol{d} = (t, T)$:
$t \in \{0, 25, 50, 75, 100, 125, 150\}$ and $T \in \{450, 475, 525, 575, 600\}$ yielding 35 possible choices.

# Chemical Example Continued

- 15 independent observations
- Four cases. Each model allowed to be true with parameter $(\theta_1, \theta_2, \sigma) = (400, 5000, 0.1)$
- Prior distribution

$$\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} \sim N\left(\begin{bmatrix} 400 \\ 5000 \end{bmatrix}, \begin{bmatrix} 70 & 0 \\ 0 & 500 \end{bmatrix}\right) 1(\theta_1 > 0)1(\theta_2 > 0).$$

$$\sigma \sim IG(10, 1)$$

- Comparing 3 different utility functions: Random design, mutual information and total separation (Masoumi et al 2013)

# Chemical Reaction Results (model 1 true)

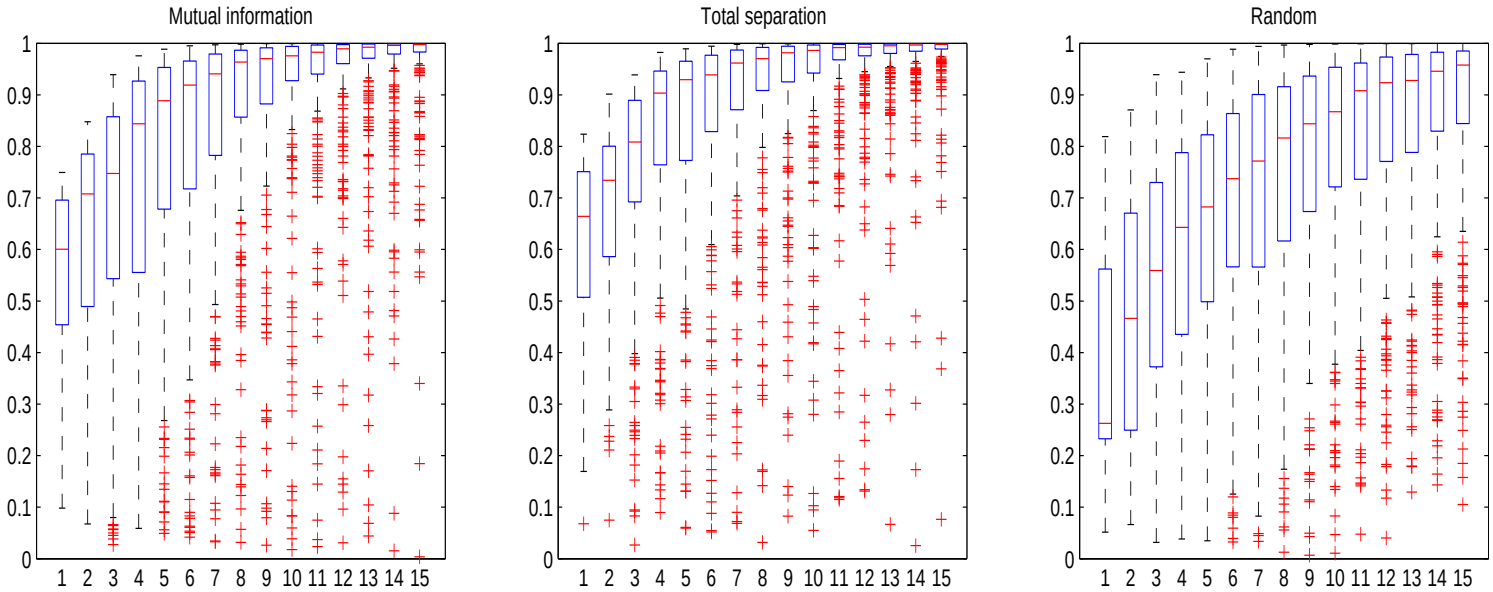


Figure: First order reaction as true.

# Chemical Reaction Results (model 2 true)

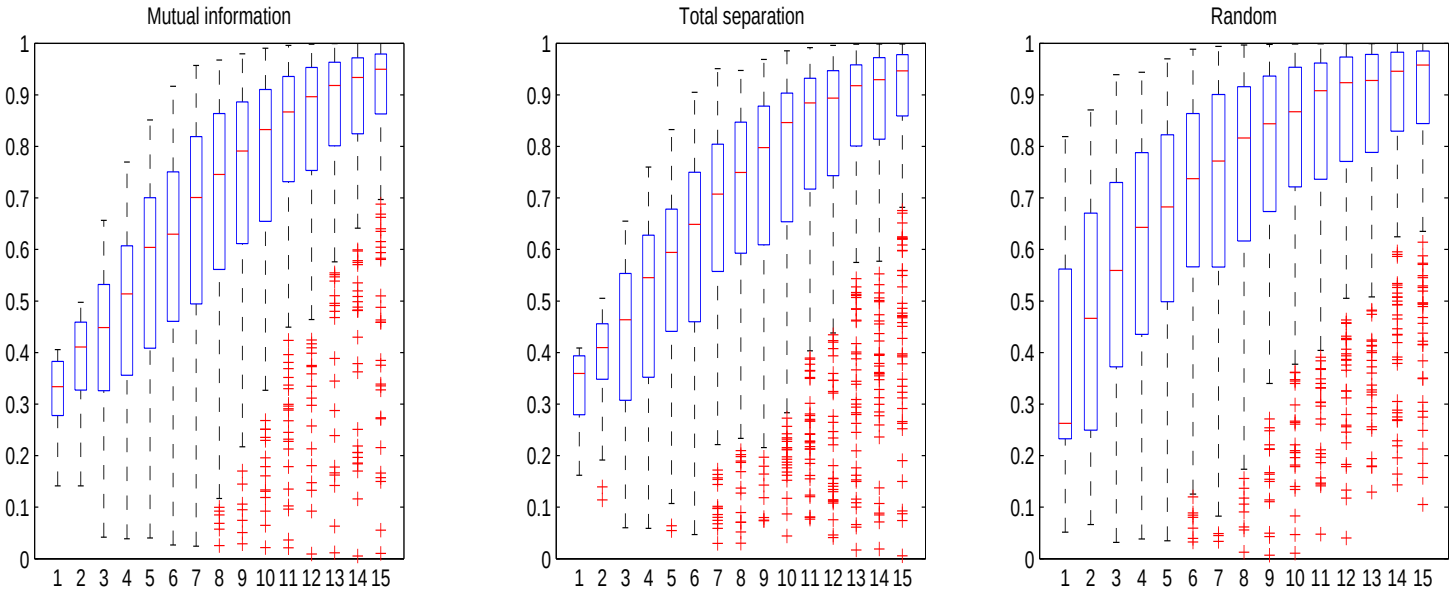


Figure: Second order reaction as true.

# Chemical Reaction Results (model 3 true)

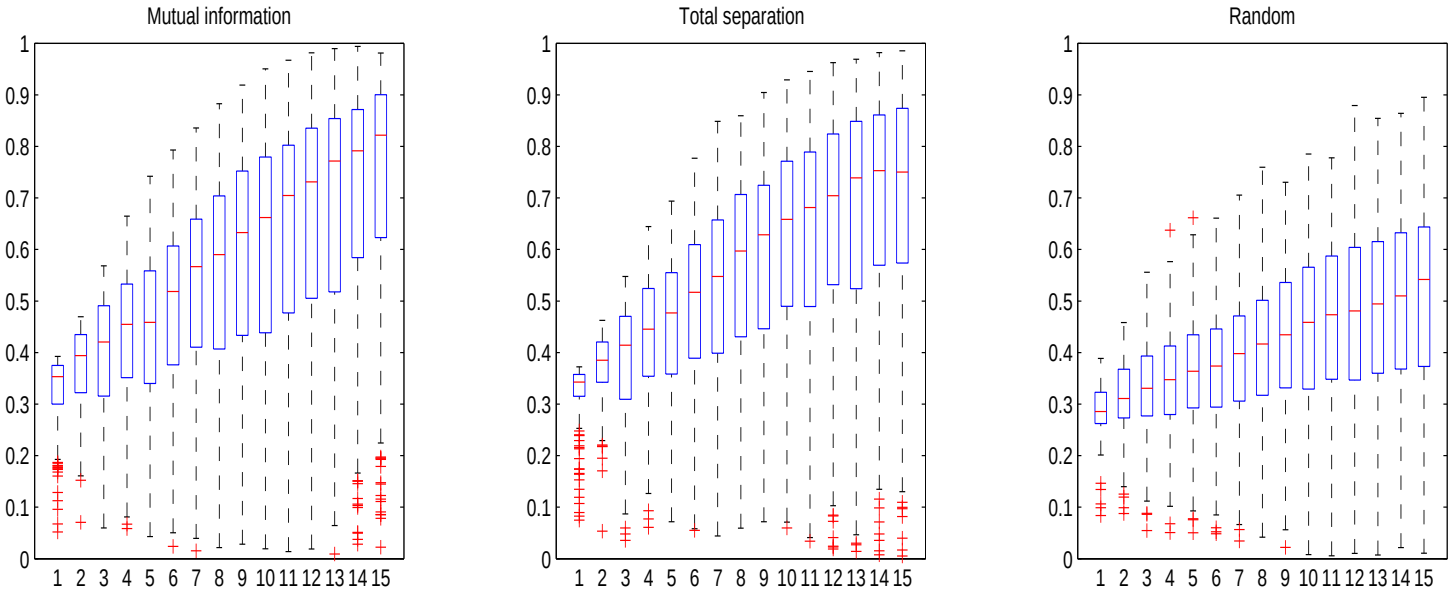


Figure: Third order reaction as true.

# Chemical Reaction Results (model 4 true)

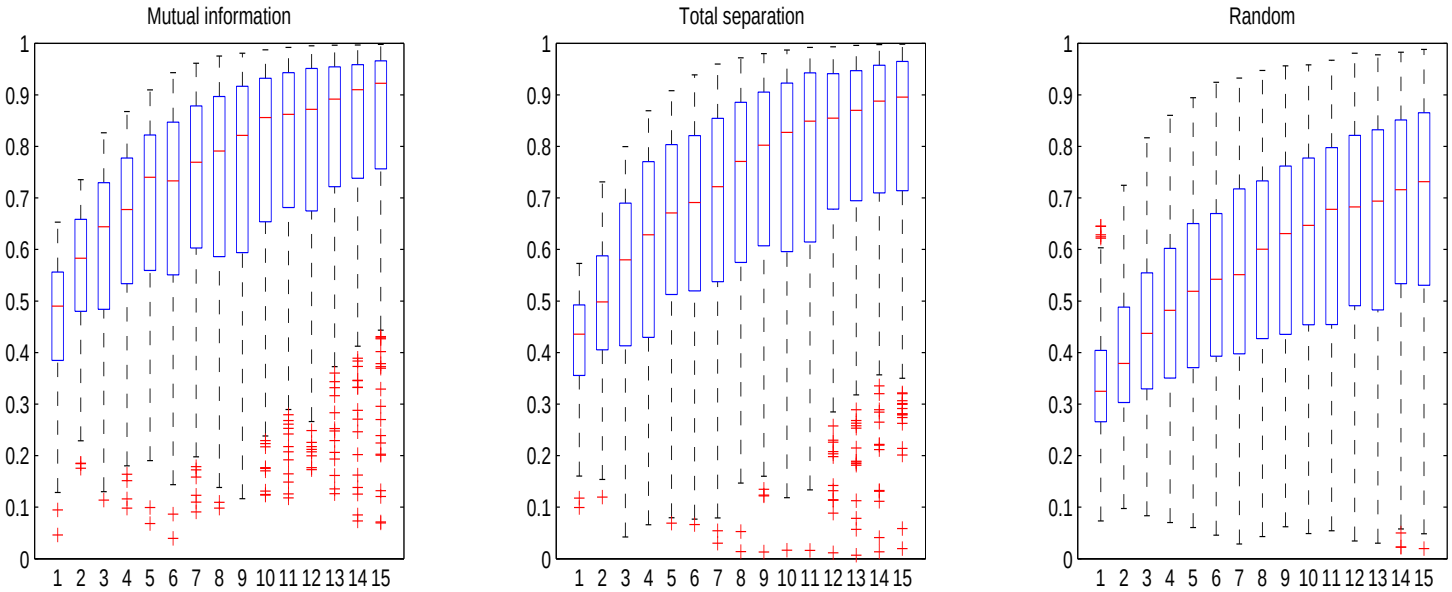


Figure: Fourth order reaction as true.

# Advantage 3: Embarrisingly Parallel

- Extend to design in presence of random effects model
- Pharmacokinetics Example. For subject $t$ the model for samples collected at design $\boldsymbol{d}_t = (d_{1t}, d_{2t})$ is:

$$y_t \sim MVN(g(\beta_t, \boldsymbol{d}_t), \delta_0 \boldsymbol{I}),$$
$$\beta_t \sim MVN(\mu, \boldsymbol{\Omega}),$$

- Here $\beta_t$ is random effect for $t$th subject

$$g(\beta_t, \boldsymbol{d}_t) = \frac{100}{\exp(\beta_{2t})} \exp\left(-\frac{\exp(\beta_{1t})}{\exp(\beta_{2t})} \boldsymbol{d}_t\right),$$

- Priors:

$$\mu \sim MVN(\mathbf{0}, \boldsymbol{\Sigma}), \text{ for } \boldsymbol{\Sigma} \text{ known.}$$
$$\boldsymbol{\Omega} \sim InvWish(\boldsymbol{\Psi}, \nu), \text{ for } \boldsymbol{\Psi} \text{ and } \nu \text{ known}$$
$$\delta_0 \sim U(a, b), \quad 0 < a \leq b < \infty,$$

- Design objective is to learn about parameters: $\boldsymbol{\theta} = (\mu, \boldsymbol{\Omega}, \delta_0)$.

# Exact-Approximate SMC

- The (marginal) likelihood

$$f(\boldsymbol{y}_t|\boldsymbol{\theta}, \boldsymbol{d}_t) = \int f(\boldsymbol{y}_t|\boldsymbol{\beta}_t, \delta_0, \boldsymbol{d}_t)\pi(\boldsymbol{\beta}_t|\boldsymbol{\mu}, \boldsymbol{\Omega})d\boldsymbol{\beta}_t$$

- Can be estimated unbiasedly. For each particle $\boldsymbol{\theta}^{(i)}$

$$f(\boldsymbol{y}_t|\boldsymbol{\theta}^{(i)}, \boldsymbol{d}_t) = \frac{1}{M}\sum_{j=1}^{M} f(\boldsymbol{y}_t|\boldsymbol{\beta}_t^{(j)}, \delta_0^{(i)}, \boldsymbol{d}_t) \quad (1)$$

  where $\boldsymbol{\beta}_t^{(j)} \sim MVN(\boldsymbol{\mu}^{(i)}, \boldsymbol{\Omega}^{(i)}), \;\; j = 1, \ldots, M.$

- SMC with unbiased estimate of likelihood $\rightarrow$ an exact-approximate algorithm! (Duan and Fulop 2013)
- Caveat: SMC can degenerate quicker compared to using 'exact' likelihood (need large enough $M$)

# Speeding things up on the GPU

- Serial implementation of SMC design too slow here
- Many ways to parallelise algorithm (over particles, $M$ and/or design choices $d$)
- Here we chose calculating the likelihood over particles in parallel on GPU (required in all aspects of algorithm)
- Order of magnitude speed improvement over C implementation of likelihood
- Work still in progress...

# Key References


- Box, G. E. P. and Hill, W. J. (1967). Discrimination among mechanistic models. Technometrics, 9(1):57-71.
- Drovandi, C. C. et al. (2014). A Sequential Monte Carlo Algorithm to Incorporate Model Uncertainty in Bayesian Sequential Design. JCGS To Appear.
- Drovandi, C. C. et al. (2013). Sequential Monte Carlo for Bayesian sequentially designed experiments for discrete data. CSDA, 57(1):320-335.
- Chopin, N. (2002). A sequential particle filter method for static models. Biometrika, 89(3):539-551.
- Del Moral, P et al (2006). Sequential Monte Carlo samplers. JRSS: Series B, 68(3):411-436.