


Stochastic Variants of EM: Monte Carlo, Quasi-Monte Carlo and More



Wolfgang Jank

Robert H. Smith School of Business

University of Maryland

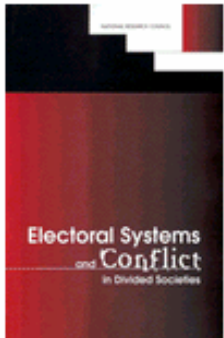
www.smith.umd.edu/faculty/wjank

An Old Method for Modern Statistical Problems: Examples

- A (personally biased) selection of MCEM applications:
 - Geostatistical model of online purchase behavior (Jank & Kannan, 2005)
 - Curve-clustering of functional databases for online auction price-dynamics (Jank, 2005)
 - Modeling flight departure delay distributions (Tu, Ball & Jank, 2005)

Online Purchase Behaviour

Online book publisher sells books in Print form and in PDF form



Publisher for THE NATIONAL ACADEMIES **nap.edu**

A PDF version is available of the book *Electoral Systems and Conflict in Divided Societies*. If you choose the PDF version, the full text will be available immediately. It can be downloaded to your computer chapter by chapter and read or printed out using [Adobe Acrobat](#).

Format	Price
PDF Version	\$14.40
Print Version	\$14.40

Estimated time to download the average chapter:

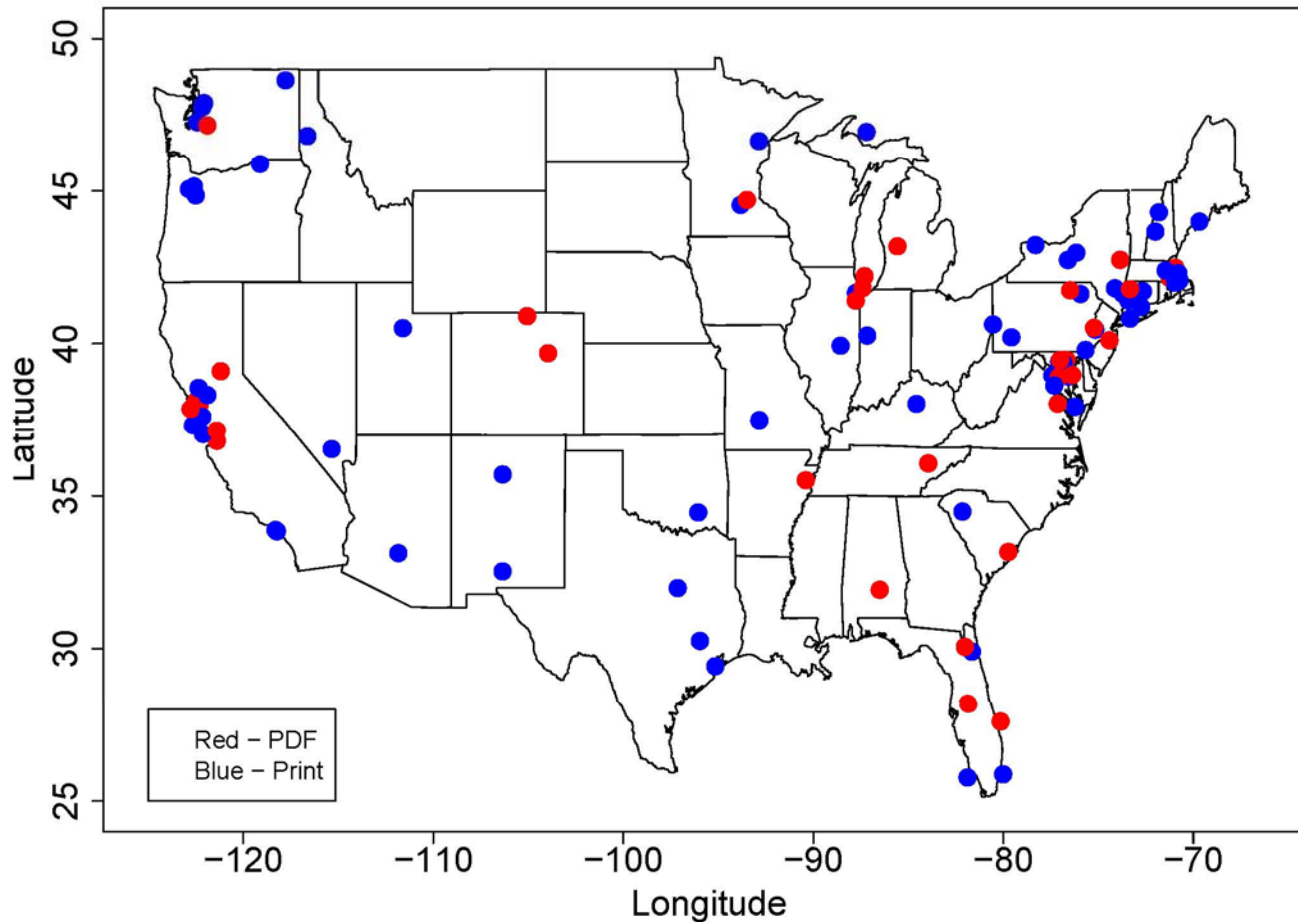
Connection Speed	Download Time
28.8 K	about 3-4 mins/chapter
56 K	about 2 mins/chapter
T1	< 1 minute/chapter

If you would like to get a better idea of the quality of the files, you may download a [free sample of the electronic version](#)

Geographically Varying Customer Preferences

- Customer's preference for either form varies due to geography
 - Areas with different broadband access
 - Varying shipping times
 - Technology readiness of customer
 - Availability of high-quality printers
 - Urban vs. rural areas
 - Etc.

Geographically Varying Customer Preferences



Model Spatial Correlation of Choices

- How can we model the (possible) spatial correlation of choices?
- For continuous data, we could use the multivariate normal distribution
- But choice (PDF vs. Print) is discrete (Bernoulli) data
- E.g. what is the correlation of the Bernoulli sequence $(0,0,1,0,1,1,0,1,0,0,0,0,1,1,0)$?

Modeling Correlated Discrete Data

- One way of modeling correlated discrete data is via
 - Mixed Models / Hierarchical Models
- General Principle
 - Let y_{ij} be j^{th} obs. from i^{th} group
 - Assume obs. **within** same group are correlated
 - We can model y_{ij} **conditional** on group i
 - $f(y_{ij} | \text{some effect due to group} = i)$
 - Assume that effect of group i is **unobserved** and varies **randomly**, modeled as r.v. u_i with distribution $g(u_i)$
 - This leads to the joint distribution of (y_{ij}, u_i)
 - $h(y_{ij}, u_i) = f(y_{ij} | u_i) g(u_i)$

Inference for Mixed Models

- For groups $i=1, \dots, n$ and obs. $j=1, \dots, m_i$ per group, we get the likelihood function
 - $L(\mathbf{y}, \mathbf{u}) = \prod_{ij} h(y_{ij}, u_i) = \prod_{ij} f(y_{ij} | u_i) g(u_i)$
- Problem: Maximum Likelihood / Maximum A Posteriori Analysis requires the **integrated-out** (or marginal) likelihood
 - $L(\mathbf{y}) = \int \prod_{ij} h(y_{ij}, u_i) du$
- Even bigger Problem: The integral $L(\mathbf{y})$ has typically no closed-form solution!

Approximating the Intractable Integral

- What can be done to approximate the intractable integral?
- Many different possibilities
 - Analytical (Taylor/Laplace Approximation)
 - E.g. SAS's Proc NLMIXED, R's nlme (library "nlme")
 - Quadrature
 - E.g. SAS's Proc NLMIXED (not sure about R)
 - Monte Carlo (i.e. simulation)
 - Some R packages, but only for very specific models...

Approximations with Monte Carlo

- There exist numerous Monte Carlo avenues:
 - “Approximate Integral First, Then Maximize”
 - Simulated Maximum Likelihood, Monte Carlo Maximum Likelihood
 - “Find Integral-Maximizer Iteratively”
 - Monte Carlo EM, Monte Carlo Newton-Raphson, Stochastic Approximation, Stochastic Approximation EM,...
 - “Ignore Integral, Do Fully Bayesian Analysis”
 - Markov-Chain Monte Carlo to simulate from the posterior distribution

When Should I Use Which Approach?

- All Monte Carlo approaches have their advantages and disadvantages!
 - Some are conceptually “easy”
 - Simulated Maximum Likelihood
 - Some have nice and stable algorithmic properties
 - Monte Carlo EM
 - Some are more appealing from an inferential/statistics-principle point-of-view
 - Bayesian inference? → MCMC

Why Use (Monte Carlo) EM?

- Is very stable
 - Choice of starting values
- Allows for significant analytical simplifications
 - Operates on log-scale
- Always gets one step closer to the goal
 - Likelihood Ascent Property
- Can handle huge databases
 - Many versions of EM in Machine Learning & Data Mining for Text Classification, Neural Nets, ..., with LOTS of data
- Can be used for online-learning
 - Delivers updates of parameters/predictions in real-time as new data arrives
- Appeals especially to statisticians
 - Missing data / Data Augmentation principle

Back to the Geostatistical Model of Online Purchase Behaviour

- Model the spatial preference via variant of Generalized Linear Mixed Models (GLMMs)
- z_i denotes **spatial coordinate/location** of i^{th} response y_i ; ($i=1, \dots, n$)
- Let (u_1, \dots, u_n) be vector of **random effects** with **correlation** structure that **decays with distance** between spatial locations
 - E.g. $\text{Corr}(u_i, u_j) = \exp[-\alpha \text{dist}(z_i, z_j)]$

Spatial GLMM of Online Purchases

- Then **Spatial GLMM** is of the form
 - Response: $y_i = 1$ if PDF, and $y_i = 0$ if Print
 - Conditional model for the observed data

$$y_i | u_i \sim \text{Binom}(n_i, p_i)$$

- where $\text{logit}(p_i) = \mathbf{x}_i \boldsymbol{\beta} + u_i$
- Model for the unobserved random effects

$$\mathbf{u} := (u_1, \dots, u_n) \sim \text{MVN}(0, \Sigma)$$

- Likelihood function intractable

$$L(\mathbf{y}) \propto \int \left(\prod_i \frac{\exp(\mathbf{x}_i \boldsymbol{\beta} + u_i)}{[1 + \exp(\mathbf{x}_i \boldsymbol{\beta} + u_i)]} \right) \exp\left(-\frac{1}{2} \mathbf{u}' \Sigma \mathbf{u}\right) d\mathbf{u}$$

Basic EM Algorithm

- Iterate between Expectation (E-) and Maximization (M-) steps

- E-step: Calculate conditional expectation of complete-data log-likelihood

$$Q(\theta | \theta^{(t)}) = E[\log h(\mathbf{y}, \mathbf{u}; \theta) | \mathbf{y}; \theta^{(t)}]$$

- M-step: Maximize Q-function wrt. θ

$$\theta^{(t+1)} = \arg \max(Q(\theta | \theta^{(t)}))$$

- Iterate until convergence
- Stationary point is the (local) maximizer of likelihood function

Monte Carlo EM for Spatial GLMM

- Problem: For Spatial GLMM, the Q-fct. equals

$$Q(\theta | \theta^{(t)}) \propto \int \left[\sum ((\mathbf{x}_i \boldsymbol{\beta} + u_i) - \log[1 + \exp(\mathbf{x}_i \boldsymbol{\beta} + u_i)]) - \frac{1}{2} \mathbf{u}' \boldsymbol{\Sigma} \mathbf{u} \right] f(\mathbf{u} | \mathbf{y}; \theta^{(t)}) d\mathbf{u}$$

which has no closed-form solution!

- Therefore: Approximate Q-fct. via Monte Carlo:

- Simulate $u_1, \dots, u_M \sim f(\mathbf{u} | \mathbf{y}; \theta^{(t)})$
- Approximate Q-fct. with empirical average

$$\hat{Q}(\theta | \theta^{(t)}) = \frac{1}{M} \sum_k \log h(\mathbf{y}, \mathbf{u}_k; \theta)$$

A few general remarks on Monte Carlo
EM, its Challenges and Problems, and
some proposed Solutions

Monte Carlo EM in its Most Basic Form

Simulation

Approximation

Maximization

Iteration

Convergence

1. Simulation:

- Simulate Monte Carlo sample from conditional distribution of missing data

2. Approximation:

- Approximate EM's Q-function via Monte Carlo average based on simulations

3. Maximization:

- Maximize approximated Q-function

➤ Iteration:

- Iterate through steps 1-3 until stopping rule kicks-in

➤ Convergence:

- Converges to local maximum

Problems and Challenges with the Basic Form: **Simulation**

Simulation

Approximation

Maximization

Iteration

Convergence

- Type of Simulation:
 - Independent vs. dependent (Importance Sampling vs. MCMC)
- Efficiency of Simulation:
 - Variance Reduction via Quasi-Monte Carlo (Jank, 2004)
- Amount of Simulation
 - Little simulation early, lots of simulation later
 - Increase simulation size intelligently
 - (Booth & Hobert, 1999; Levine & Casella, 2001; Levine & Fan, 2004; Caffo, Jank & Jones, 2005)

Problems and Challenges with the Basic Form: **Approximation**

- There are no actual **problems** with the approximation step
 - It is simply an average!
- There exist a few modifications to the basic approximation step:
 - Using importance re-weighting
 - More efficient use of all simulations (Quintana, Liu & delPino, 1999)
 - Using stochastic-approximation scheme
 - Convergence without simulation size increase (Delyon, Lavielle & Moulines, 1999)

Simulation

Approximation

Maximization

Iteration

Convergence

Problems and Challenges with the Basic Form: **Maximization**

- Problems same as for the deterministic EM algorithm
 - Closed form M-step often not available
 - Iterative methods necessary (e.g. Newton-Raphson, Gradient Search, etc)
 - Can be burdensome when parameter space is large!
 - Problem can be alleviated by appealing to EM's likelihood-ascent property
 - For example, using partial M-steps via Expectation-Conditional-Maximization (Meng & Rubin, 1993)

Simulation

Approximation

Maximization

Iteration

Convergence

Problems and Challenges with the Basic Form: Iteration

- How many iterations do we run MCEM?
 - MCEM is stochastic method; deterministic stopping rules do not apply
 - Any deterministic rule can be satisfied purely by chance!
 - Some of the proposed solutions are
 - To apply deterministic rules over several consecutive iterations (Booth & Hobert, 1999)
 - To measure improvement in likelihood function statistically (Gu & Zhu, 2001; Caffo, Jank & Jones, 2005)

Simulation

Approximation

Maximization

Iteration

Convergence

Problems and Challenges with the Basic Form: **Convergence**

Simulation

Approximation

Maximization

Iteration

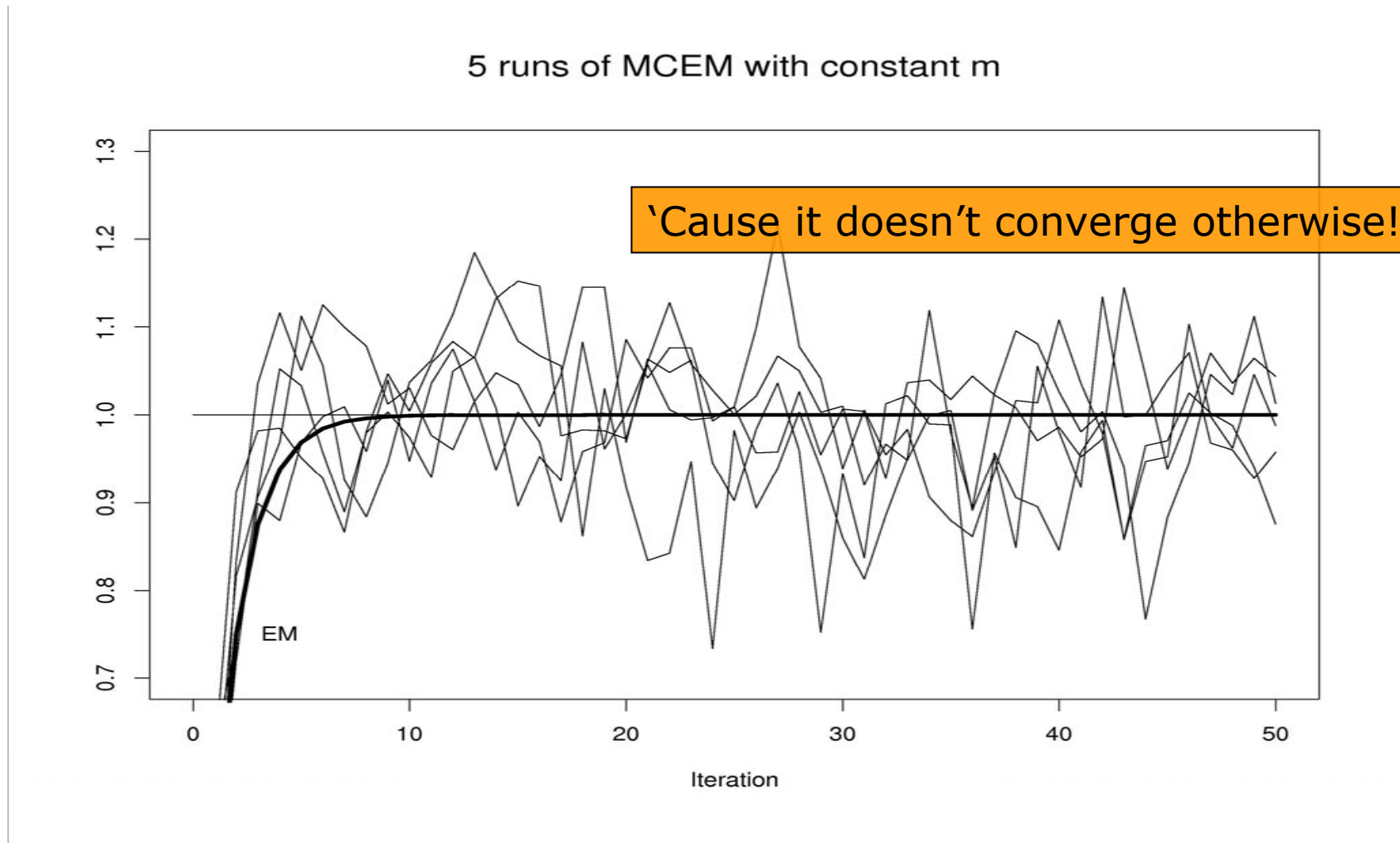
Convergence

- MCEM converges to **local maximum**
 - Can lead to sub-optimal solution if several local optima exist
 - For example in mixture-models, mixture-of-experts, neural nets
- Some Solutions
 - Initiate from different starting values
 - "Hit-and-Miss"
 - Combine MCEM with ideas of **Global Optimization**
 - E.g. MCEM with Genetic Algorithm (Tu, Ball & Jank, 2005; Jank, 2005)

A Closer Look at some of the Specific Challenges of MCEM

- Increasing the simulation size intelligently via Ascent-MCEM
- An alternative approach via Stochastic Approximation EM
- Efficient simulation via Quasi-Monte Carlo

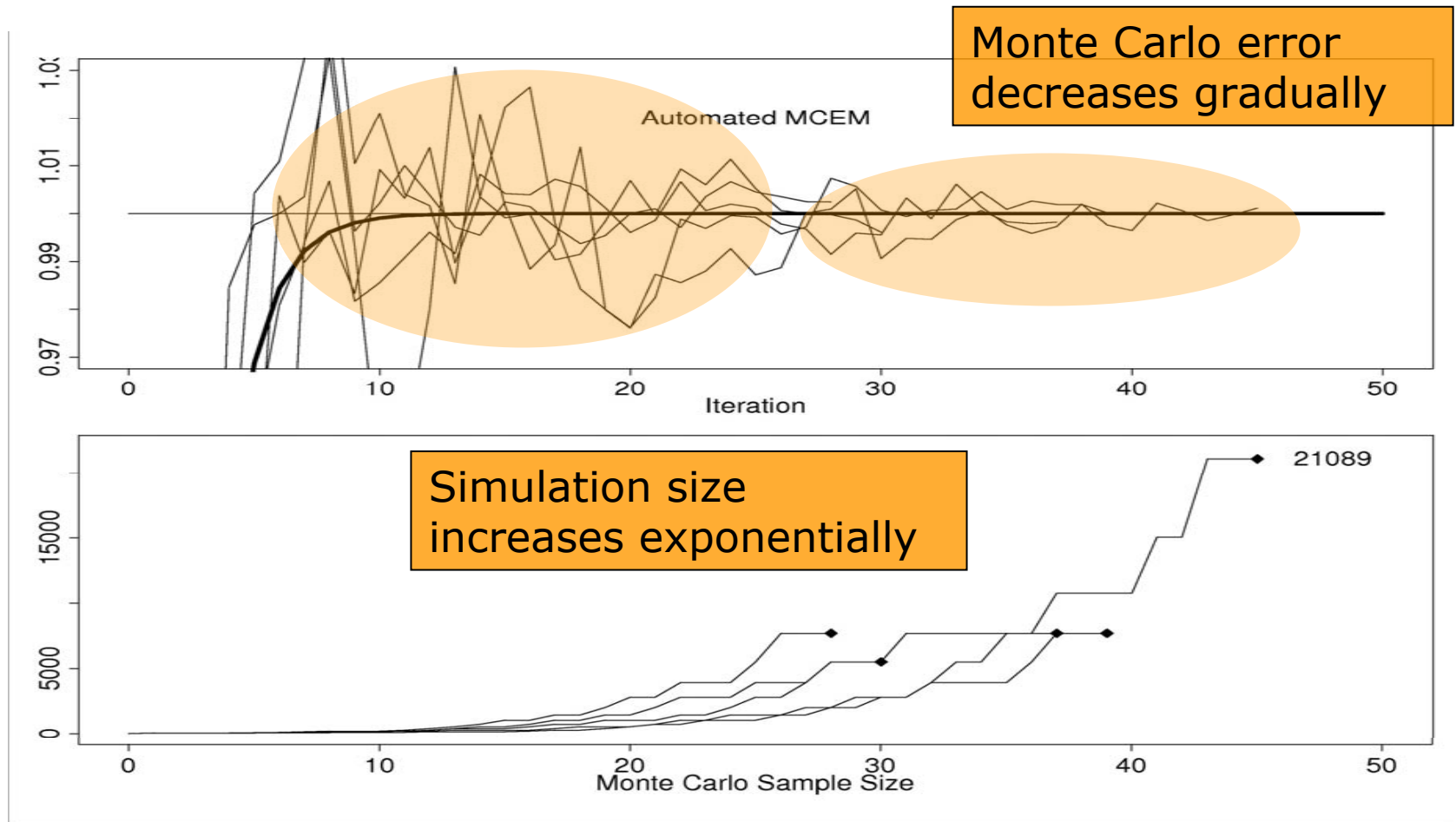
Why Increase Simulation Amount of MCEM?



Why Increase Amount Automatically?

- 'Cause different applications/models/data will require different amounts and distributions of the simulated data
 - Eventually, MCEM should become part of major software packages!!

Typical Iteration Path and Simulation Amount of an Automated MCEM



Ascent-MCEM for Intelligent Increase of Simulations

- Proposed by Caffo, Jank & Jones, 2005
- Idea based on Likelihood-Ascent Property:
 - Any update, for which the difference in the Q-functions is positive, increases the likelihood
- Basic Ascent-MCEM steps:
 - Estimate a level- α lower confidence bound for the difference in the Q-functions
 - If lower bound positive, then parameter update increases likelihood
 - Keep current simulation; move-on to next iteration!
 - If lower bound negative, then Monte Carlo error too large
 - Increase simulation-size; repeat iteration!

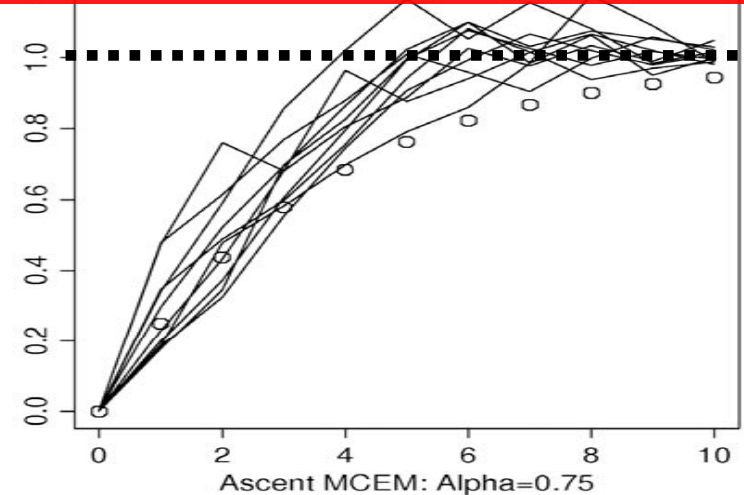
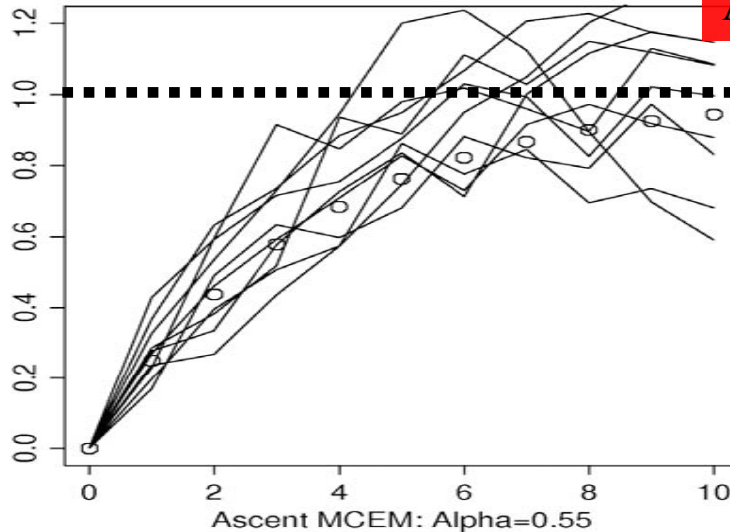


Ascent MCEM Details

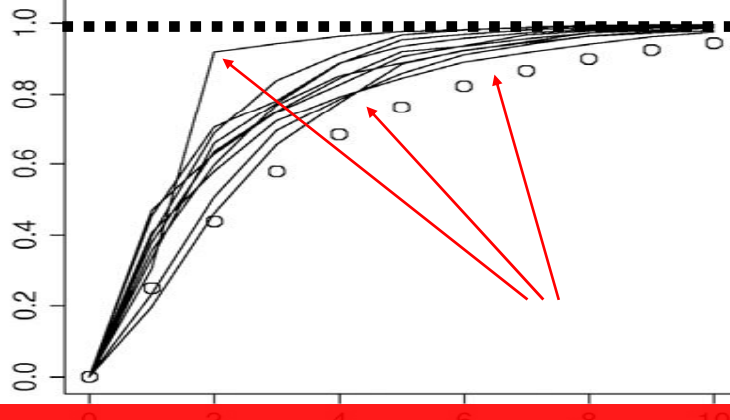
- Let $\theta^{(t)}$ denote the current parameter value and let θ^* denote a candidate for the parameter update
- If θ^* satisfies $DiffQ := Q(\theta^* | \theta^{(t)}) - Q(\theta^{(t)} | \theta^{(t)}) > 0$ then it increases the likelihood function, that is
$$L(\theta^* | \mathbf{y}) > L(\theta^{(t)} | \mathbf{y})$$
- Problem: Cannot calculate $DiffQ$ in closed form, so we estimate it via Monte Carlo
$$MC_DiffQ := \hat{Q}(\theta^* | \theta^{(t)}) - \hat{Q}(\theta^{(t)} | \theta^{(t)}) > 0$$
- We also estimate its standard error and derive a **lower** α -level confidence bound (LLCB) for $DiffQ$
- Notice: the candidate θ^* increases the likelihood (with $100(1 - \alpha)\%$ confidence), if LLCB > 0
- Rational for the **lower confidence bound**:
 - *We don't care if we under-estimate $DiffQ$; in fact that's nice! (Why?)*
 - *We only care if we over-estimate $DiffQ$; then, the update potentially does not increase the likelihood!*

Performance and α -Level

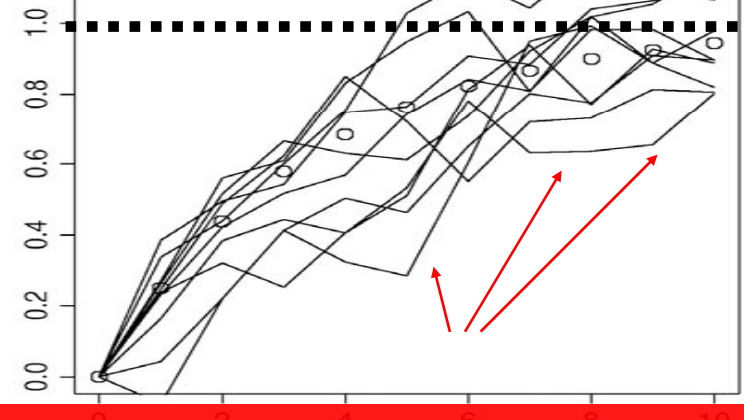
Ascent-MCEM with 75% Confidence Level



Ascent-MCEM with 95% Confidence Level



Benchmark: Booth&Hobert's Method



Moves faster TOWARDS the solution than EM; does not move AWAY

Exhibits frequent moves AWAY from the solution

Benefits of Ascent-MCEM

- Easier implementation for MCMC or Quasi-Monte Carlo since based on univariate quantity *DiffQ* rather than multivariate as in Booth & Hobert
- Counterproductive use of simulations rare since Likelihood-Ascent Property holds
- More stable estimates, especially of variance-covariance matrix, due to larger final-iteration simulation sizes
- Acceleration of deterministic EM possible/likely!

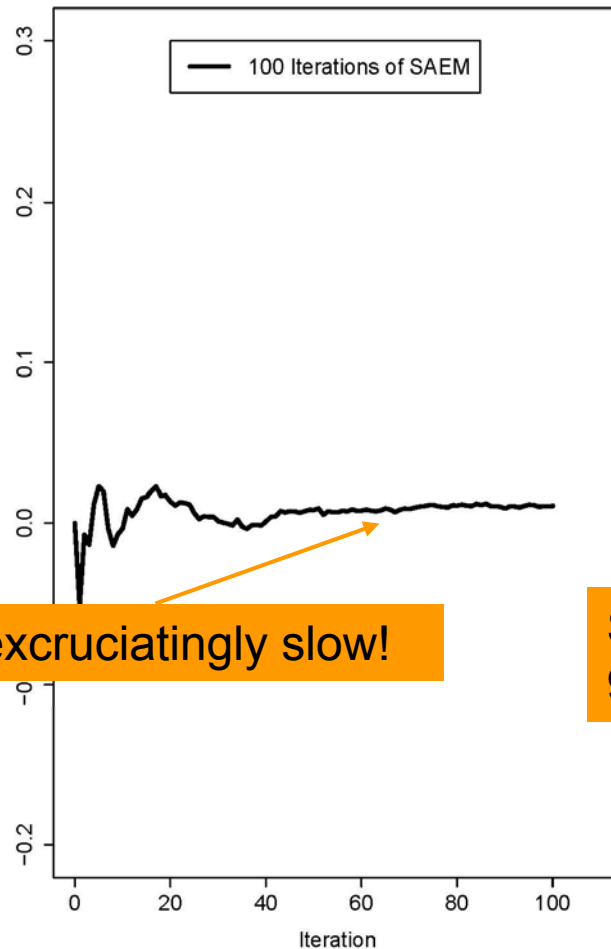
Alternatives to Increasing Simulations

- Automated/intelligent sample size rule make method harder to implement and to fine-tune!
 - Can we also converge with constant simulations?
 - Yes: using a stochastic approximation scheme
- The Stochastic-Approximation EM (SAEM) replaces the Q-fct. with the P-fct.

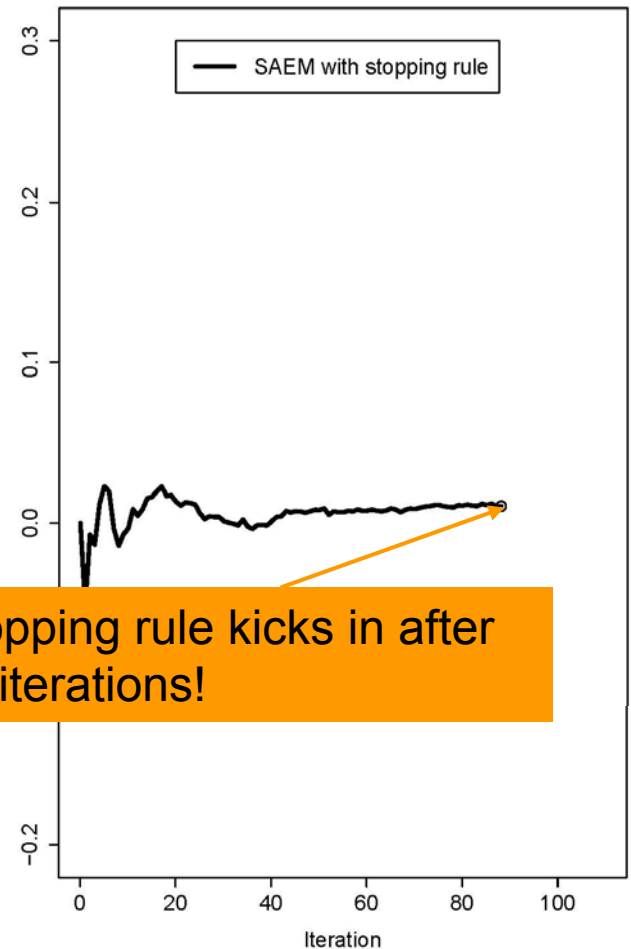
$$\hat{P}^{(t)}(\theta) = (1 - \gamma_t)\hat{P}^{(t-1)}(\theta) + \gamma_t\hat{Q}(\theta | \theta^{(t-1)})$$

- Discounting scheme to compute weighted average of Q-functions from current and all the previous iterations
- Advantage:
 - Converges with FIXED (and SMALL; in fact, ANY) sample-size!
 - Only need to select magnitude of **discounting factor** γ_t !
- Disadvantage:
 - Choice of discounting factor often arbitrary!
 - Choice strongly affects performance!

SAEM with Small Discounting Factor



Moves excruciatingly slow!

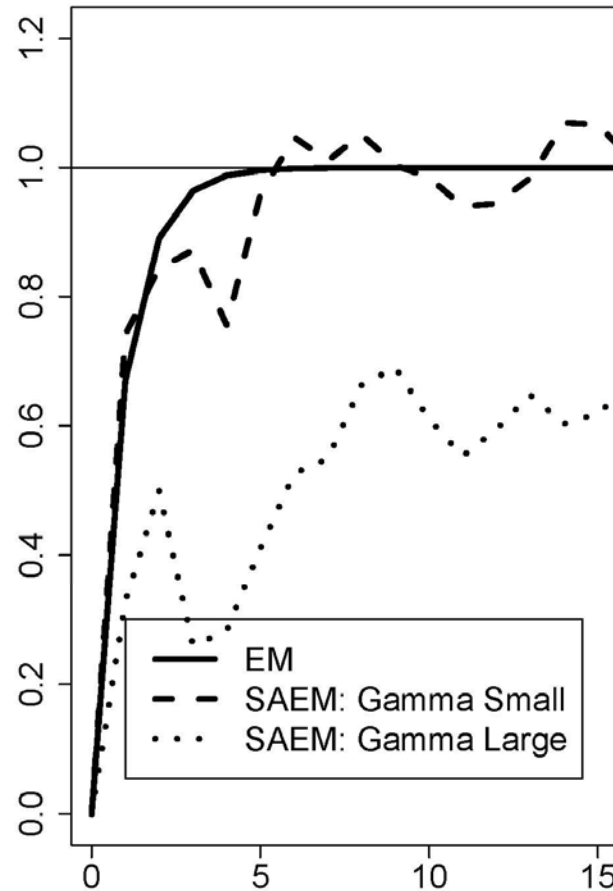


Stopping rule kicks in after 90 iterations!

But the true parameter estimate equals 1!!!!

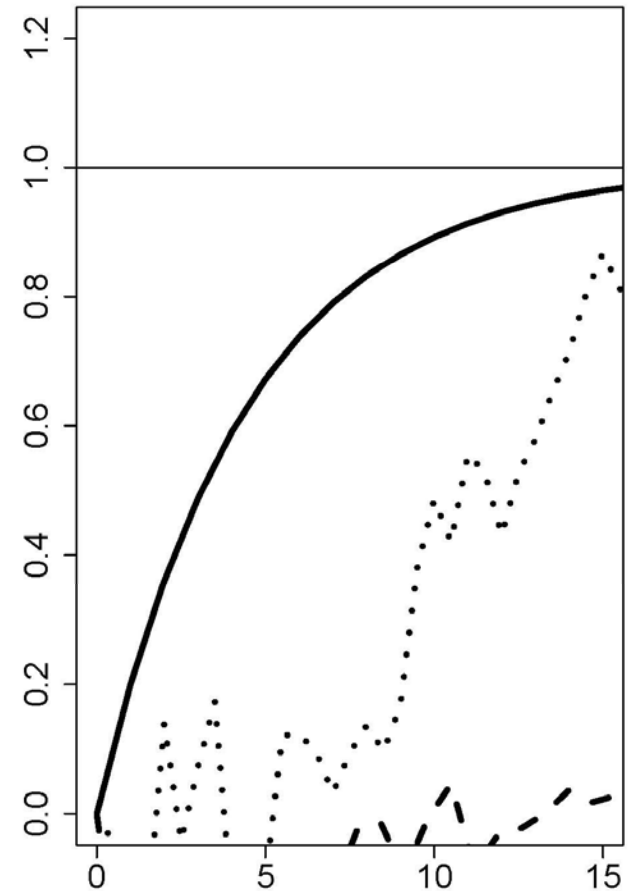
SAEM and Missing Information

Little Missing Info



Fast EM – Small Gamma!

Lots Of Missing Info



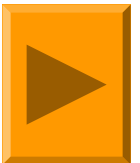
Slow EM – Large Gamma!

EM's conv. rate depends on amount of missing info

This should be taken into account when determining discounting factor!

Automated Choice of Discounting Factor

- Should depend on EM's convergence rate
 - I.e. the missing-to-complete information
- Jank (2004) suggests a way to link the two
 - But more analytical work possible/necessary on this topic!



Efficient Simulation via Quasi-Monte Carlo

X_0 in $[0,1]$

$X_1 = f(X_0)$

$X_2 = f(X_1)$

...

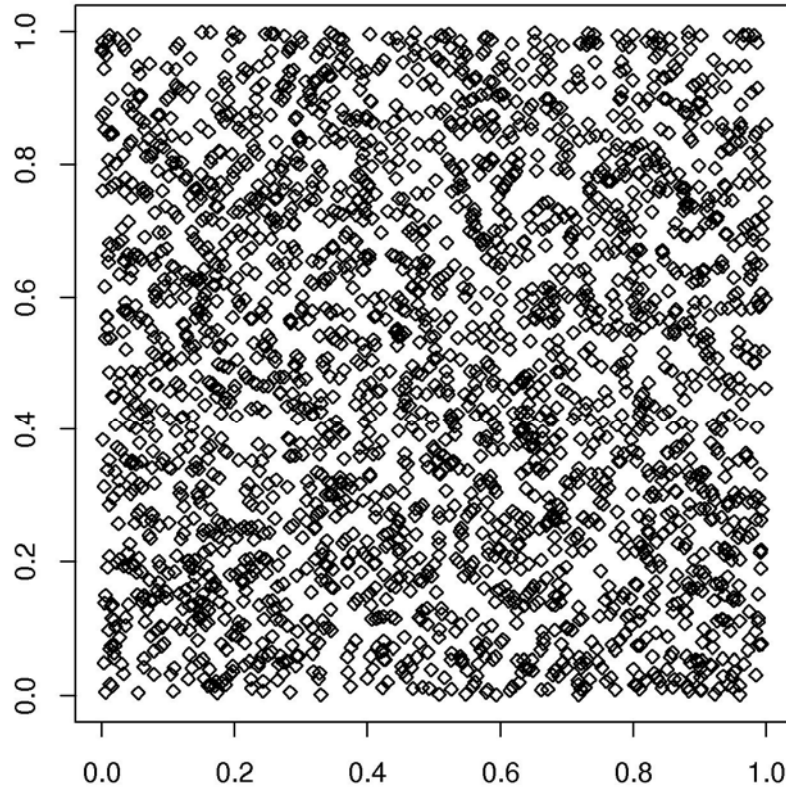
yields

$[X_0, X_1, \dots, X_n]$

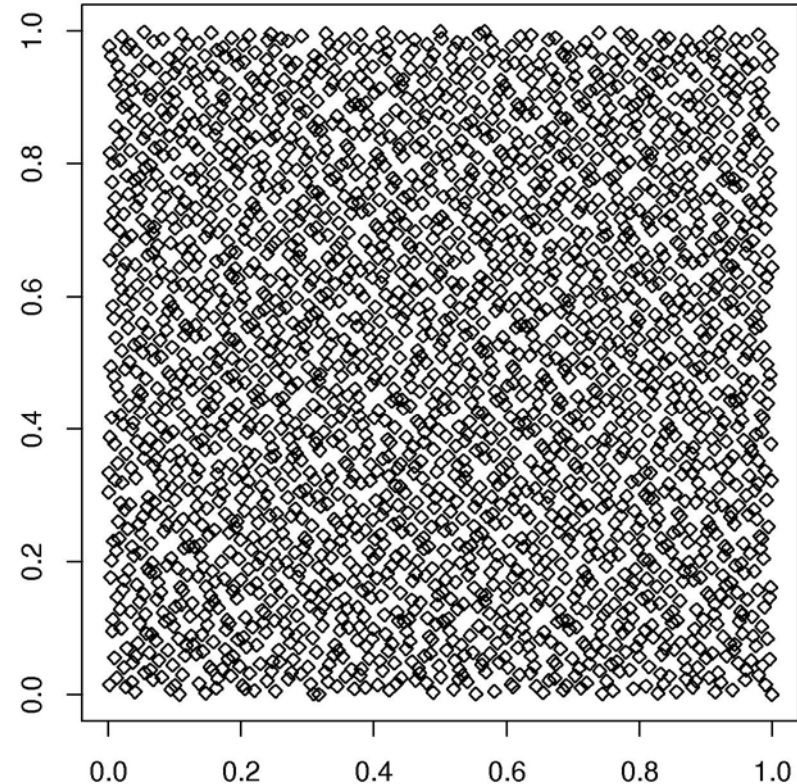
deterministic!

- Quasi-Monte Carlo is deterministic counterpart to classical Monte Carlo
- Does not select points randomly
- Chooses points deterministically, with best-possible spread in sampling space
 - Often called “Low-Discrepancy” Sequence
- Results in more efficient use of simulated data and reduced variance of the estimates

Monte Carlo vs. Quasi-Monte Carlo



Regular Monte Carlo



Quasi-Monte Carlo

Quasi-Monte Carlo points achieve a better spread of the sampling space

Randomized Quasi-Monte Carlo

$X_0 \sim \text{Uniform}[0,1]$

yields

$[X_0, X_1, \dots, X_n]$

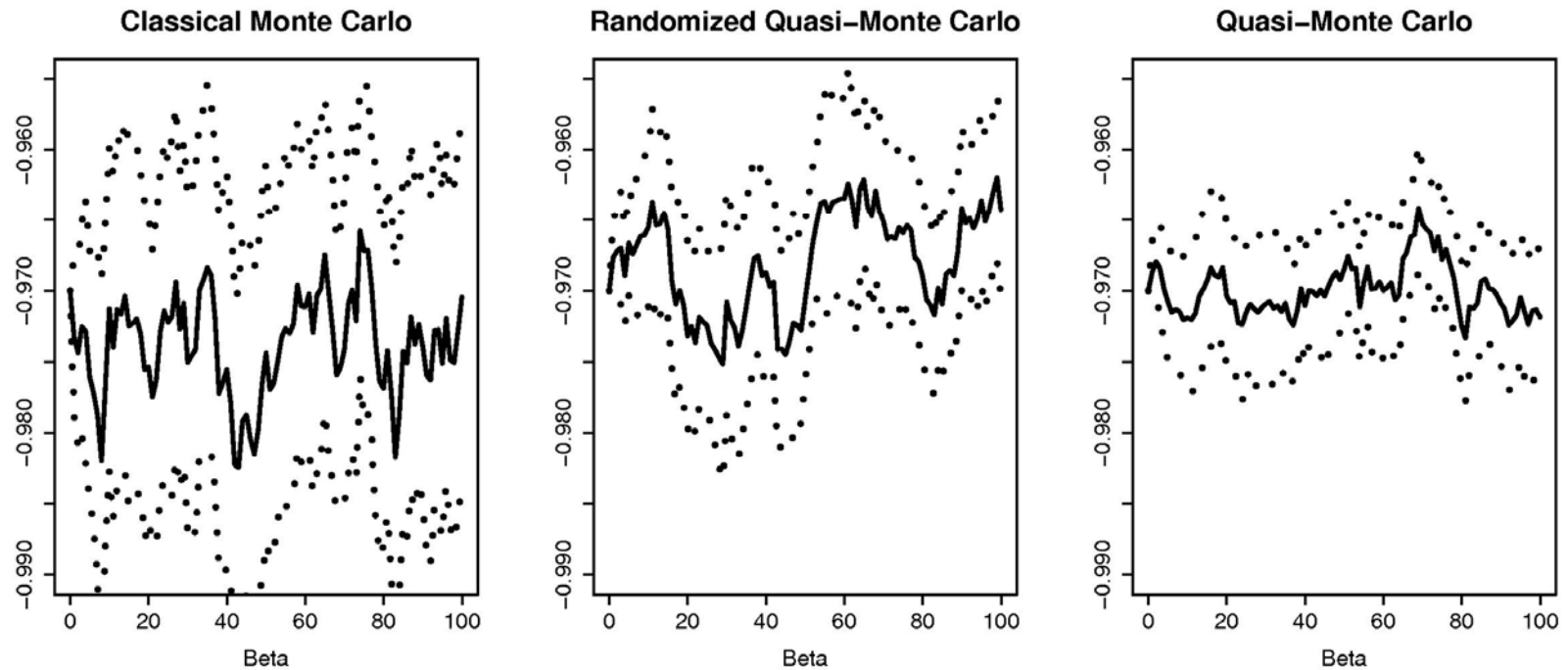
all uniformly
distributed

- Quasi-Monte Carlo deterministic
 - No statistical error estimation possible
- Randomized Quasi-Monte Carlo
 - Preserves low-discrepancy properties
 - Each point is uniformly distributed
 - Allows for statistical estimation of error!
 - See L'Ecuyer & Lemieux (2002) for overview; also Owen (2004).

Quasi-Monte Carlo EM for Geostatistical Model of Online Purchases

- Need to simulate high-dimensional vectors because of spatial correlation structure
 - Computational very intensive!
 - Efficient simulation very desirable!
- Quasi-Monte Carlo version of MCEM
 - Randomized Quasi-Monte Carlo for automated simulation-size selection
 - Results in huge reduction of simulation effort
 - See Jank (2004)

Noise of Geostatistical Parameter Estimates for different Monte Carlo Simulation Approaches



Using the SAME simulation-size, Quasi-Monte Carlo achieves the best variance-reduction!

Simulation Effort for Geostatistical Model

- Need to simulate vectors of dimension n ($=16$ in our case)
- In the t^{th} MCEM iteration, need to simulate m_t such vectors
- For a total of T iterations, need to simulate $N = \sum_t m_t$ vectors (= total simulation effort)

Total Simulation Effort for Monte Carlo vs. Quasi-Monte Carlo

Method	Simulation Effort
Monte Carlo	800,200
Quasi-Monte Carlo	21,000

A reduction of almost 98%!

Thanks for Listening

Papers available at
www.smith.umd.edu/faculty/wjank