Three-phase Sequential Design for Sensitivity Experiments

C.F. Jeff Wu
Georgia Institute of Technology

- Sensitivity testing : problem formulation
- Review of existing procedures for sequential sensitivity testing
- Proposed procedure: 3-phase optimal design (dubbed 3pod for its steady performance)
- Simulation comparisons with existing procedures
- Conclusions and further work

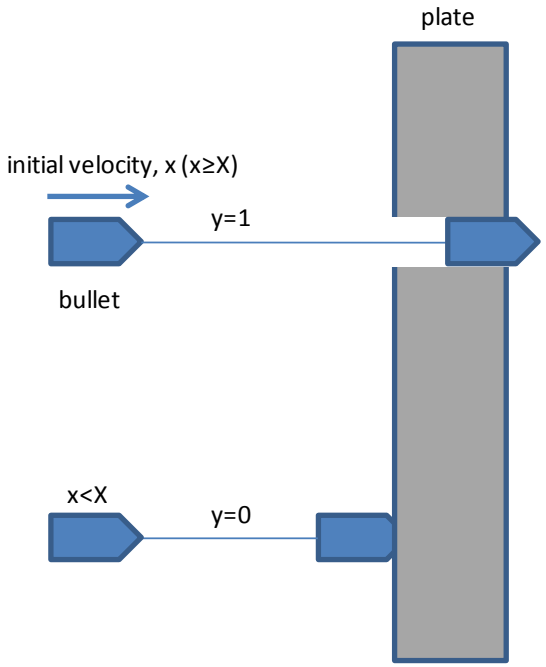
(Joint work with Yubin Tian)

Georgia Institute of Technology

ISyE INDUSTRIAL & SYSTEMS ENGINEERING

Sensitivity testing

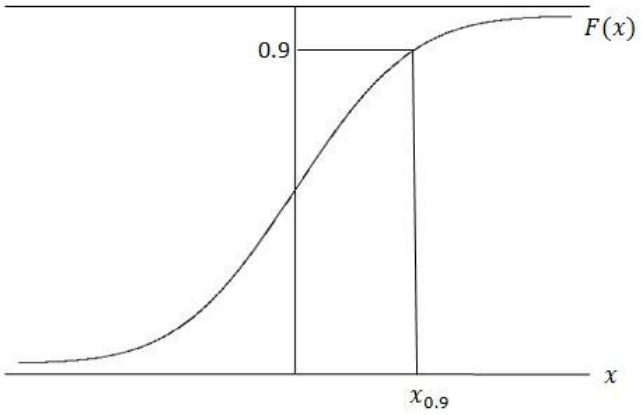
- Stress/stimulus level x: launching velocity, drop height
- Response/nonresponse y = 1 or 0: penetrate, explode



X=unknown critical level (a random quantity)

Quantal response curve

- Quantal response curve $F(x)$ = prob ($y = 1 \mid x$); interested in estimating the p-th quantile x_p with $F(x_p) = p$, p typically high, e.g., p = 0.9, 0.99, 0.999. Useful for certification or quantification of test items. Common in military and heavy industry applications



- Choice of F: probit, logit, or skewed version
- Problem/challenge: find a sequential design procedure to estimate x_p efficiently and for small samples

Review of existing procedures

- Up-and-down method (Dixon-Mood, 1948):

$$x_{i+1} = \begin{cases} x_i + \Delta, \text{if } y_i = 0 \\ x_i - \Delta, \text{if } y_i = 1 \end{cases},$$

  easy to use, not efficient, only for median $x_{0.5}$

- Stochastic approximation (Robbins-Monro, 1951):

$$x_{i+1} = x_i - \frac{c}{i}(y_i - p),$$

  optimal $c = \hat{\beta}_i^{-1}$, $\hat{\beta}_i$ regression slope based on $\{x_n, y_n\}$ (Lai-Robbins, 1979). For binary data y, use of linear regression slope is not efficient, i.e., not the best exploitation of data

Modification of Robbins-Monro procedure

- Recognizing the deficiency for binary data, Joseph (2004) proposed a modification: retain binary structure btn y and x, assume $\theta\ (=x_p) \sim N(x_1, \tau_i^2)$. Consider the scheme

$$x_{i+1} = x_i - a_i(y_i - b_i).$$

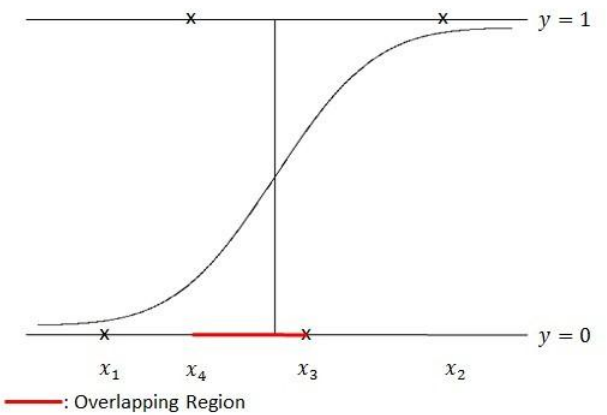
- Let $Z_i = x_i - \theta$, and choose a_i, b_i to minimize $E(Z_{i+1}^2)$ under $E(Z_{i+1}) = 0$. Assuming $Z_{i+1} \sim N(0, \tau_i^2)$, the solution is

$$b_i = \Phi\left\{\frac{\Phi^{-1}(p)}{(1+\beta^2\tau_i^2)^{1/2}}\right\}, \quad a_i = \frac{1}{b_i(1-b_i)}\frac{\beta\tau_i^2}{(1+\beta^2\tau_i^2)^{1/2}}\phi\left\{\frac{\Phi^{-1}(p)}{(1+\beta^2\tau_i^2)^{1/2}}\right\},$$
$$\tau_{i+1}^2 = \tau_i^2 - b_i(1-b_i)a_i^2, \quad \beta = \frac{G'(G^{-1}(p))}{\phi(\Phi^{-1}(p))}\cdot\frac{1}{\sigma},$$

- We call it the Robbins-Monro-Joseph (RMJ) procedure.

Logit-MLE procedure

- Wu (1985): relating stochastic approximation to likelihood estimation to take advantage of the latter's estimation efficiency. Assume a parametric model $F(x|\gamma)$, $\gamma=(\mu,\sigma)$, like logit or probit. At the ith run, $\hat{\gamma}_i$ = MLE of γ. Then, choose x_{i+1} so that $F(x_{i+1}|\hat{\gamma}_i) = p$
- However, existence of MLE requires an *overlapping data pattern*



- Wu did not incorporate overlapping data pattern in its design procedure. Bayesian modification (Joseph-Tian-Wu, 2007)

D-optimality based procedure

- Neyer's method (1994) has three parts: use guessed value σ_g
  (i) use a modified binary search to generate $y = 1$ and $y = 0$,
  and to "close-in" on design region of interest; update σ_g
  (ii) use *D*-optimality criterion based on σ_g to generate overlapping pattern;
  (iii) Assume a parametric model like probit or logit for $F(x|\theta)$; same procedure as in (ii) except that the MLE $\hat{\theta}$ is used in the *D*-criterion. This step for estimation efficiency
- *First* to incorporate the achieving of overlapping pattern in the design procedure

Challenges

- For this problem with a long history, is there a consensus on best procedure? No! Why?
- Up-and-down for its simplicity appeal is still misused by less sophisticated users; lately Neyer has become popular among well informed users; Joseph's modification of RM for binary data has received scant attention; some military in-house procedure like Langlie (1962) has been used but is *ad hoc*, and no good theoretical justification
- There is a still room for improvement, thus our work ☺

Three-phase optimal design

- A trilogy of search-estimate-approximate:
  I. (*search*) to generate $y = 1$ and $y = 0$, to "close-in" on region of interest and to obtain overlapping data pattern; similar to Neyer's parts 1-2, details differ
  II. (*estimate*) use *D*-optimality criterion evaluated at MLE $\hat{\theta}$ to generate design points; *spread out* design points (same as Neyer's part 3)
  III. (*approximate*) Taking $\hat{\mu} + F^{-1}(p)\hat{\sigma}$, where $\hat{\mu}, \hat{\sigma}$ are MLE of μ,σ based on data in I-II, as the starting value, use the Robbins-Monro-Joseph (RMJ) procedure to generate design points
- 3-phase optimal design, dubbed as 3pod (for its steady performance ☺)

Georgia Institute of Technology

ISyE INDUSTRIAL & SYSTEMS ENGINEERING

Phase I of 3pod

- It has three stages I1, I2, I3
- I1. (quickly obtain y = 1 and y = 0). Choose (μ_{min}, μ_{max}) for location parameter μ and σ_g as guessed value of scale parameter σ and μ_{max}- $\mu_{min} \geq 6\sigma_g$. Take y_1 and y_2 at $x_1 = \frac{3}{4}\mu_{min} + \frac{1}{4}\mu_{max}$, $x_2 = \frac{1}{4}\mu_{min} + \frac{3}{4}\mu_{max}$.
  Four cases result:
  (i) $y_1 = y_2 = 0$ ⟶ x_1, x_2 to the left of μ; take $x_3 = \mu_{max} + 1.5\sigma_g$. If $y_3 = 1$, move to I2. If $y_3 = 0$, take $x_4 = \mu_{max} + 3\sigma_g$. If $y_4 = 1$, move to I2. If $y_4 = 0$, range not large; increase x by $1.5\sigma_g$ until y=1.

Phase I of 3pod (continued)

(ii) $y_1 = y_2 = 1$, do the mirror image of (i)
(iii) $y_1 = 0$, $y_2 = 1$: good! Move to I2
(iv) $y_1 = 1$, $y_2 = 0$: range too narrow around μ,

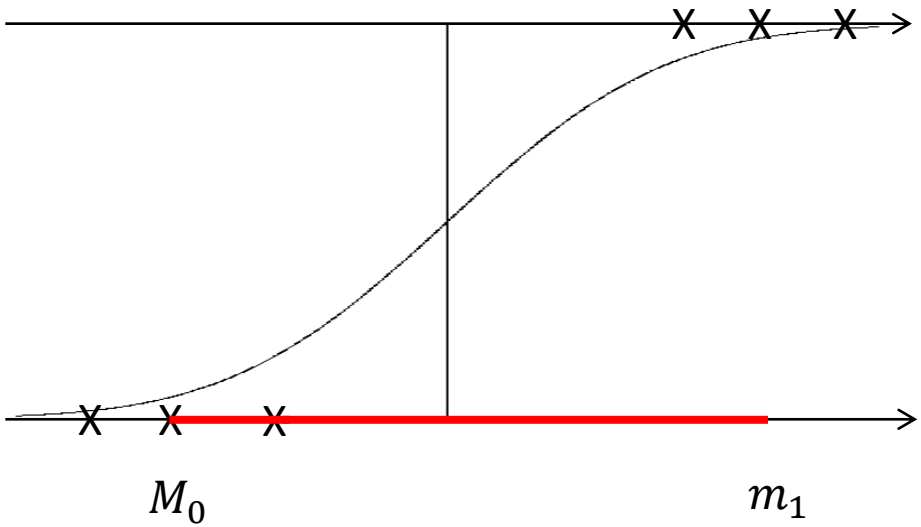
expand it by taking $x_3 = \mu_{min} - 3\sigma_g$,

$x_4 = \mu_{max} + 3\sigma_g$; move to I2

- Note: I1 is like "dose ranging" in dose-response studies

Trapped in separation?

- Let M_0 = largest x value with $y = 0$, m_1= smallest x value with $y = 1$. Overlapping iff $M_0 > m_1$, separation iff $M_0 \leq m_1$
- Running test within the separation interval $[M_0, m_1]$ will forever be *trapped in separation* ☹. ➡ When the interval is small, get out to avoid logjam



I2: stage 2 of phase I

- If overlapping in data from I1, move to I3. Otherwise, take next level at $\hat{\mu}$ (=MLE assuming probit and σ_g); if overlapping, move to I3. If no overlapping, update M_0, m_1 , $\hat{\mu}$, take next level at $\hat{\mu}$ until m_1-M_0<1.5 σ_g. Then choose x levels *outside* the separation interval [M_0, m_1]. See next.
- Take next run at m_1+0.3σ_g; if y = 0, overlapping, move to I3. If y = 1, next run at M_0-0.3σ_g; if y = 1, overlapping, move to I3. Otherwise it suggests σ_g is too large, *reduce* it to $\frac{2}{3}\sigma_g$, repeat I2 until seeing overlapping.

I3: stage 3 of phase I

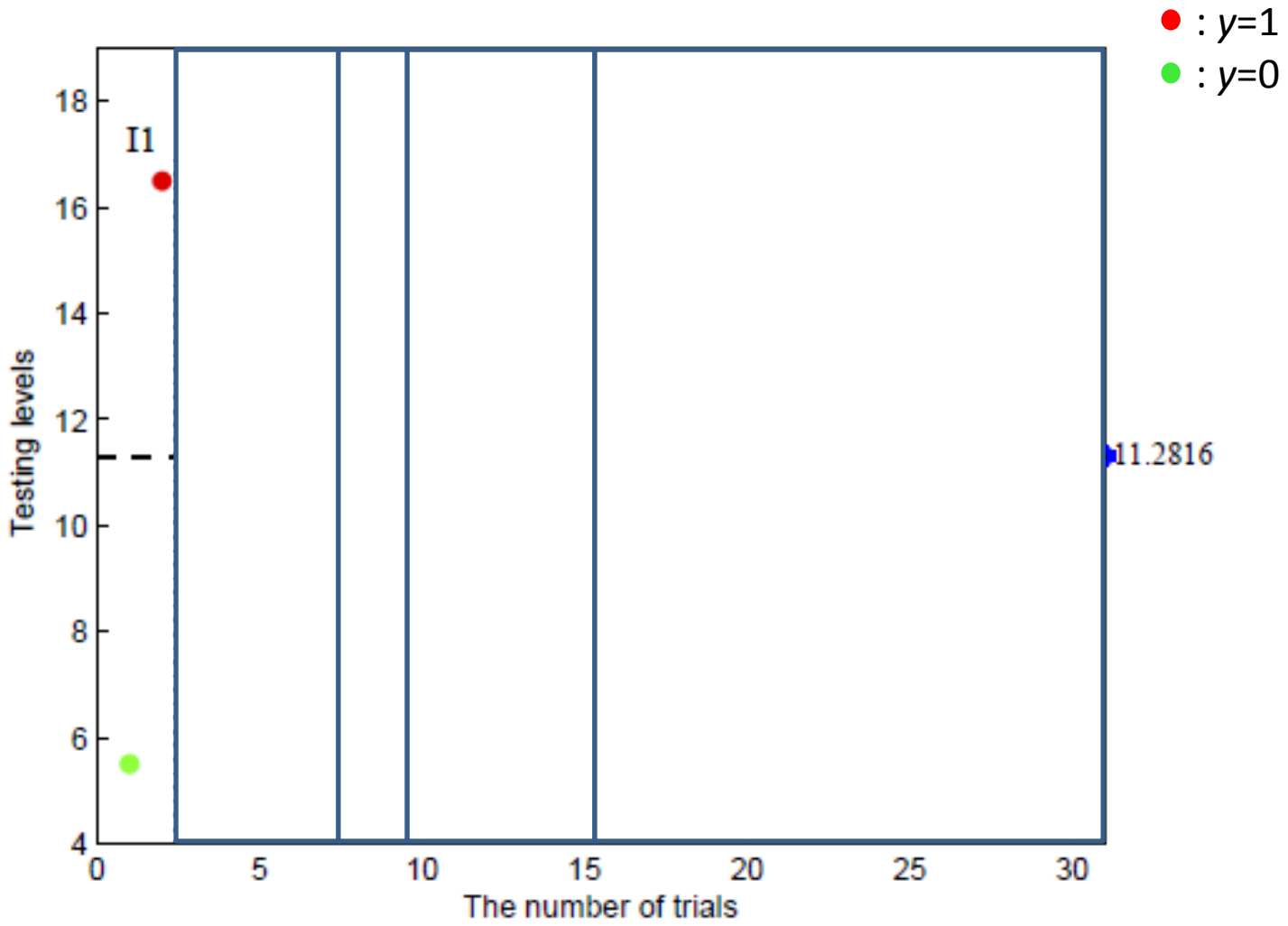
- To enhance overlapping pattern
- If M_0-$m_1 \geq \sigma_g$, take one more run at $(M_0+m_1)/2$;

  if M_0-$m_1 < \sigma_g$, take two runs at

  $(M_0+m_1)/2 \pm 0.5\sigma_g$. Then move to phase II.

Illustrative Example

(0,22), probit, $\mu=10$, $\sigma=1$, $\sigma_g=3$, $x_{0.99}=11.2816$



Comparison in terms of # of wasted (separating) runs in order to generate 1000 successful (overlapping)runs

(i) n=40 (n_1=25, n_2=15 for 3pod)

method	$\mu_g = 9 \sim 11$				
	$\sigma_g = 0.5$	$\sigma_g = 1.0$	$\sigma_g = 2.0$	$\sigma_g = 3.0$	$\sigma_g = 4.0$
Up-and-Down	$1 \sim 2$	$32 \sim 40$	$106 \sim 1775$	$2158 \sim 37681$	$45595 \sim 1530915$
Neyer	$23 \sim 34$	$74 \sim 84$	$414 \sim 528$	$498 \sim 1103$	$2142 \sim 2411$
3pod	0	$0 \sim 1$	$0 \sim 4$	$6 \sim 16$	$14 \sim 30$

Comparison in terms of # of wasted runs (in order to generate 1000 successful runs) (ii) n=60 (n_1=30, n_2=30 for 3pod)

method	$\mu_g = 9 \sim 11$				
	$\sigma_g = 0.5$	$\sigma_g = 1.0$	$\sigma_g = 2.0$	$\sigma_g = 3.0$	$\sigma_g = 4.0$
Up-and-Down	$0 \sim 1$	$1 \sim 7$	$22 \sim 1052$	$1202 \sim 24934$	$29157 \sim 1020277$
Neyer	$21 \sim 33$	$68 \sim 77$	$408 \sim 526$	$494 \sim 1041$	$2029 \sim 2334$
3pod	0	0	$0 \sim 1$	$0 \sim 2$	$0 \sim 3$

Comparison in terms of # of wasted runs (in order to generate 1000 successful runs) (iii) n=80 (n_1=35, n_2=45 for 3pod)

method	$\mu_g = 9 \sim 11$				
	$\sigma_g = 0.5$	$\sigma_g = 1.0$	$\sigma_g = 2.0$	$\sigma_g = 3.0$	$\sigma_g = 4.0$
Up-and-Down	$0 \sim 1$	$1 \sim 2$	$2 \sim 662$	$787 \sim 18519$	$21396 \sim 764958$
Neyer	$16 \sim 32$	$62 \sim 74$	$393 \sim 521$	$482 \sim 993$	$1971 \sim 2289$
3pod	0	0	0	0	$0 \sim 1$

Summary of results

- As σ_g increases, # of wasted runs gets bigger. Larger σ_g indicates more fluctuating behavior
- Up-and-down is the worst, dropped in further comparisons
- 3pod *consistently outperforms* Neyer, especially for large σ_g because its phase I has a more elaborate search to reach overlapping than Neyer's parts 1-2 (binary search, *D*-optimal)
- For 3pod, by increasing n_1 (= sample size of phase I) by 5, # of wasted runs decreases to nearly zero

RMSE for estimation of $x_{0.9}$, n=40, (n_1=25, n_2=15 for 3pod) , true distribution=normal

		$\sigma_g = 0.5$	$\sigma_g = 1.0$	$\sigma_g = 2.0$	$\sigma_g = 3.0$	$\sigma_g = 4.0$
$\mu_g = 9$	Neyer	0.4798	0.4957	0.5095	0.4675	0.5268
	3pod	0.4284	0.4534	0.4686	0.4472	0.4606
	RMJ	0.3109	0.2605	0.3035	0.3529	0.3929
$\mu_g = 10$	Neyer	0.4596	0.4644	0.4958	0.4817	0.4626
	3pod	0.4505	0.4520	0.4897	0.4423	0.4498
	RMJ	0.2967	0.2632	0.3065	0.3595	0.4046
$\mu_g = 11$	Neyer	0.5681	0.5001	0.5005	0.6202	0.7446
	3pod	0.4436	0.4480	0.4780	0.4583	0.4439
	RMJ	0.3054	0.2730	0.3147	0.3605	0.5139

RMJ $\succ$ 3pod $\succ$ Neyer

except for μ_g =11, σ_g=4, 3pod is the best (RMJ deteriorates from σ_g=3 to σ_g=4).

RMSE for estimation of $x_{0.99}$, n=60, (n_1=n_2=30 for 3pod), true distribution=normal

		$\sigma_g = 0.5$	$\sigma_g = 1.0$	$\sigma_g = 2.0$	$\sigma_g = 3.0$	$\sigma_g = 4.0$
$\mu_g = 9$	Neyer	0.7160	0.6310	0.7445	0.6834	0.5309
	3pod	0.5574	0.5532	0.5737	0.5542	0.5619
	RMJ	0.4633	0.4005	0.5064	1.3509	3.9470
$\mu_g = 10$	Neyer	0.6225	0.6270	0.6987	0.6678	0.4409
	3pod	0.6033	0.5859	0.6078	0.5213	0.5580
	RMJ	0.4537	0.4128	0.4752	2.3509	4.9470
$\mu_g = 11$	Neyer	0.8675	0.6616	0.7621	0.8921	0.8699
	3pod	0.5765	0.5789	0.5892	0.5752	0.5525
	RMJ	0.4629	0.4172	0.7808	3.3509	5.9470

For yellow entries, 3pod $\succ$ Neyer $\succ$ RMJ

For others, RMJ $\succ$ 3pod $\succ$ Neyer;

Sudden deterioration of RMJ when σ_g increases.

Values of x_1-$x_{0.99}$, x_1=starting value, $x_{0.99}$=true value under normal

	σ_g=0.5	σ_g=1.0	σ_g=2.0	σ_g=3.0	σ_g=4.0
μ_g=9	-2.1632	-1	1.3263	3.6527	5.979
μ_g=10	-1.1632	0	2.3263	4.6527	6.979
μ_g=11	-0.1632	1	3.3263	5.6527	7.979

Explanation for the poor performance of RMJ:

- Poor starting value x_1 (i.e., large x_1-$x_{0.99}$)
- Large $\sigma_g \Rightarrow$ small a_i in the RMJ iterations $x_{i+1}=x_i-a_i(y_i-b_i)$, small steps in iterations
- Small σ_g does not suffer because larger iteration steps can compensate for poor start

Poor performance of RMJ: a case study

- Take n = 60, μ_g =10, σ_g=4, RMSE = 4.947 = Bias in simulations. So all the errors are due to the bias term. Here x_1 = 19.3054, much large than $x_{0.99}$ = 12.3263. x_2 = 19.228, x_3 = 19.1548, and $y_1 = y_2 = y_2 = 1$. Each of the following 58 iterations make tiny steps (due to large σ_g) toward 12.3263 and their y values are equal to 1. When it terminates, x_{61} = 17.2733, still far away from 12.3263, with bias = 4.947

RMSE for estimation of $x_{0.999}$, n=80, (n_1=35, n_2=45 for 3pod), true distribution=normal

		$\sigma_g = 0.5$	$\sigma_g = 1.0$	$\sigma_g = 2.0$	$\sigma_g = 3.0$	$\sigma_g = 4.0$
$\mu_g = 9$	Neyer	0.8149	0.7850	0.8817	0.8398	0.5460
	3pod	0.8371	0.7959	0.7713	0.7277	0.7679
	RMJ	0.6970	0.5841	0.5282	4.0104	7.4440
$\mu_g = 10$	Neyer	0.6793	0.7627	0.8758	0.7850	0.4917
	3pod	0.8168	0.8323	0.7909	0.7220	0.7347
	RMJ	0.6835	0.5618	1.4748	5.0104	8.4440
$\mu_g = 11$	Neyer	0.9837	0.7928	0.9245	1.0653	0.9861
	3pod	0.8311	0.7950	0.7800	0.7423	0.7555
	RMJ	0.7087	0.6622	2.4748	6.0103	9.4440

Same conclusion as in the case of n=60

Values of x_1-$x_{0.999}$, x_1=starting value, $x_{0.999}$=true value under normal

	σ_g=0.5	σ_g=1.0	σ_g=2.0	σ_g=3.0	σ_g=4.0
μ_g=9	-2.5451	-1	2.0902	5.1805	8.2707
μ_g=10	-1.5451	0	3.0902	6.1805	9.2707
μ_g=11	-0.5451	1	4.0902	7.1805	10.2707

- Same explanation as in the case of n=60 for the poor performance of RMJ
- For logistic *F*, conclusions are qualitatively the same

Conclusions and further work

- 3pod outperforms Neyer uniformly
- 3pod and Robbins-Monro-Joseph (RMJ) are the two winners; 3pod performs more steadily but RMJ excels when it does not deteriorate
- How to choose between the two?
- Further improvement for each procedure
- 3pod has four "modules": I1 (dose ranging), I2 (overlapping search), II (opt estimate), III (approxim). These modules can be reassembled for other purposes, or be deployed to improve other procedure like RMJ with I1