Particle Swarm Optimization Assisted Markov Chain Monte Carlo

Matthew Simpson

Department of Statistics, University of Missouri

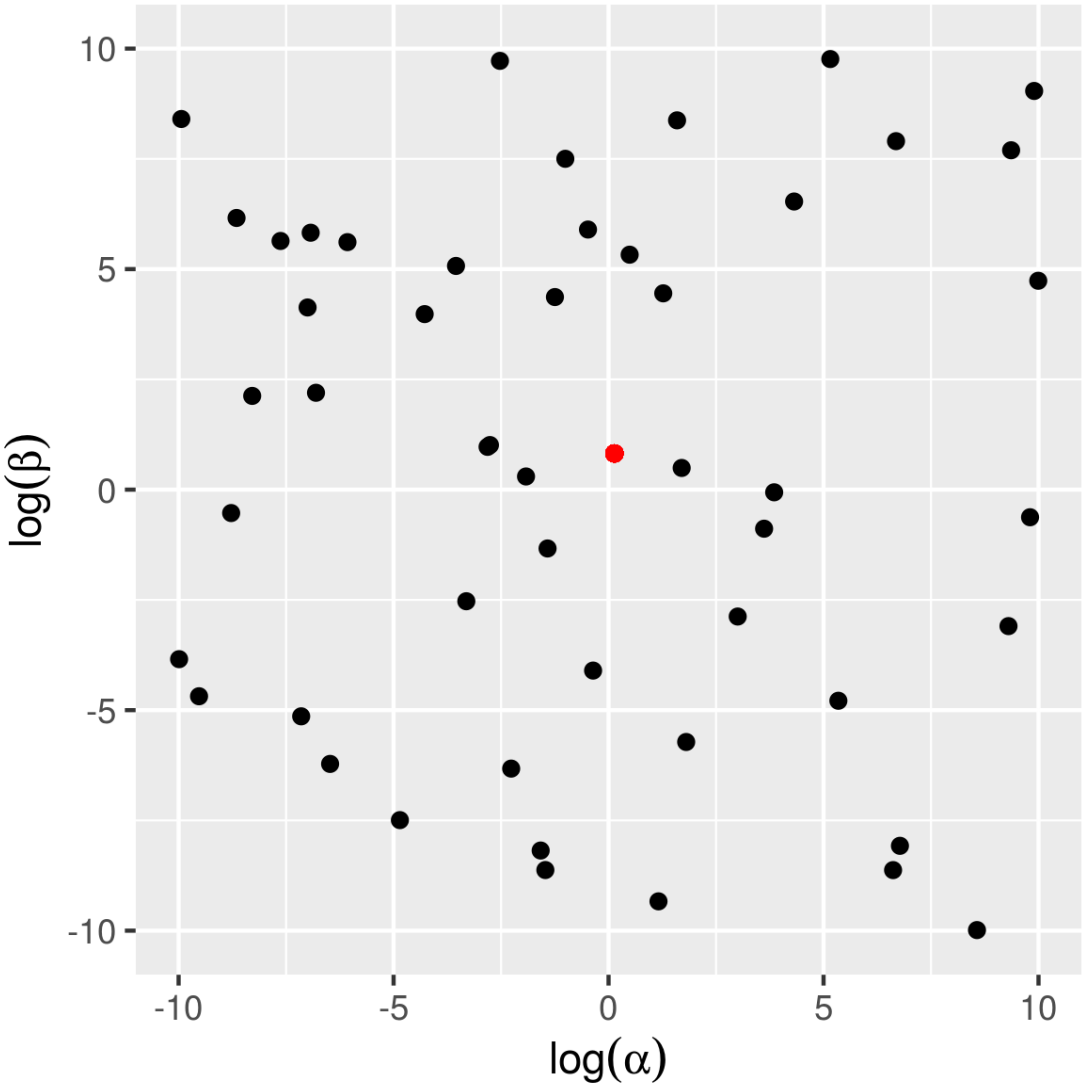
August 3, 2016

Joint work with Christopher Wikle and Scott H. Holan

Research supported by the NSF-Census Research Network

Particle Swarm Optimization

Finding MLE for an *iid* beta model. Red point = true MLE.



Particle Swarm Optimization

Goal: maximize $Q(\boldsymbol{\theta}) : \Theta \to \Re$; $\Theta \subseteq \Re^D$.

Populate Θ with n_{part} particles. Define particle i in period t by:

- a location $\boldsymbol{\theta}_i(t) \in \Theta$;
- a velocity $\boldsymbol{v}_i(t) \in \Theta$;
- a personal best location $\boldsymbol{p}_i(t) \in \Theta$: $Q(\boldsymbol{p}_i(t)) \geq Q(\boldsymbol{\theta}_i(s))$ for $s \leq t$;
- a group best location $\boldsymbol{g}_i(t) \in \Theta$: $Q(\boldsymbol{g}_i(t)) \geq Q(\boldsymbol{\theta}_i(s))$ for $s \leq t$.

Update particle i from t to $t+1$ via: for $j = 1, 2, \ldots, D$

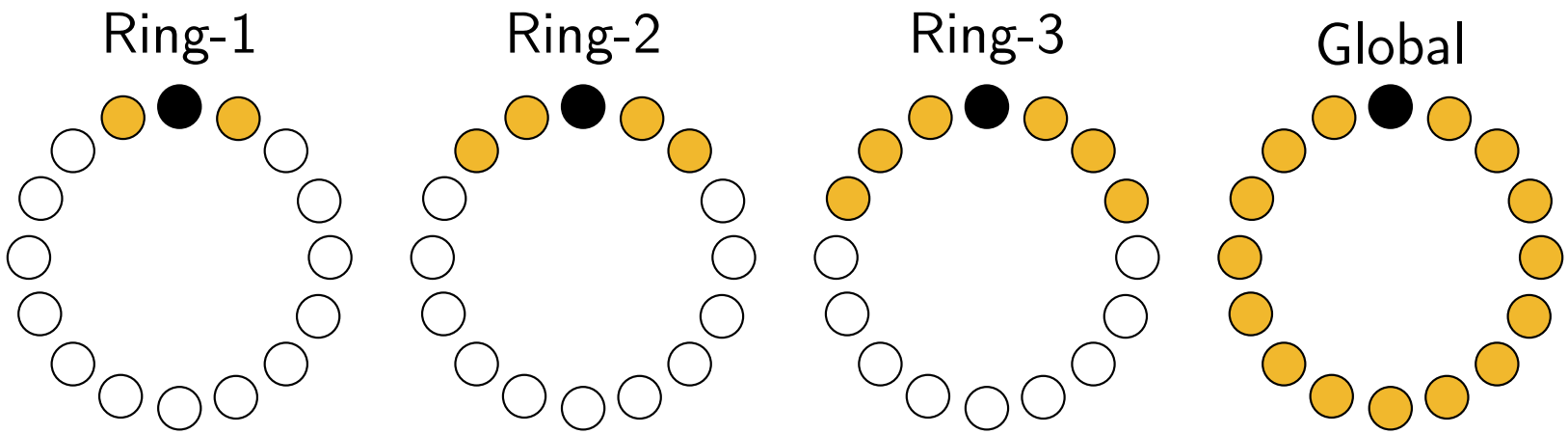
$$\begin{aligned} v_{ij}(t+1) &= \omega v_{ij}(t) + U(0, \phi_1) \times \{p_{ij}(t) - \theta_{ij}(t)\} \\ &\quad + U(0, \phi_2) \times \{g_{ij}(t) - \theta_{ij}(t)\} \\ &= \text{inertia} + \text{cognitive} + \text{social}, \\ \theta_{ij}(t+1) &= \theta_{ij}(t) + v_{ij}(t+1), \end{aligned}$$

Default choices: $\omega = 0.7298$, $\phi_1 = \phi_2 = 1.496$
(Clerc and Kennedy, 2002; Blum and Li, 2008)

PSO — Neighborhood Topologies

Sometimes it is useful to restrict the flow of information across the swarm — e.g. complicated objective functions with many local optima.

Ring-k neighborhood topology: arrange particles in a ring; each particle has k neighbors to the left and k to the right.



Filled in gold particles are neighbors of the filled in black particle.

Adaptively Tuning PSO

In PSO larger $\omega \implies$ more exploration, smaller $\omega \implies$ more exploitation.

Idea: slowly decrease $\omega(t)$ over time (Eberhart and Shi, 2000).

AT-PSO: tune $\omega(t)$ like in adaptive MCMC (Andrieu and Thoms, 2008).

- Define the swarm's improvement rate:

$$R(t) = \frac{|\{i : Q[\boldsymbol{p}_i(t)] > Q[\boldsymbol{p}_i(t-1)]\}|}{n}.$$

  and let R^* be the target rate.
- Then update: $\log \omega(t+1) = \log \omega(t) + c \times \text{sgn}\{R(t) - R^*\}$.

Good choices: $c = 0.1$ and $R^* = 0.5$.

Tuning $\omega(t)$ allows the swarm to adjust the exploration / exploitation tradeoff based on local conditions.

- This has a tendency to speed up convergence.
- ...but convergence may be premature in multi-modal problems.

Laplace Approximation to the Posterior

Posterior distribution: $p(\boldsymbol{\theta}|\boldsymbol{y}_{1:n})$. Let $\boldsymbol{\theta}_n^*$ be the posterior mode and let $\boldsymbol{H}_n(\boldsymbol{\theta}_n^*)$ denote the Hessian of $\log p(\boldsymbol{\theta}|\boldsymbol{y}_{1:n})$ evaluated at $\boldsymbol{\theta}_n^*$.

Laplace approximation (Schervish, 1997, Sections 7.4.2 and 7.4.3):

$$\boldsymbol{\theta}|\boldsymbol{y}_{1:n} \overset{a}{\sim} N(\boldsymbol{\theta}_n^*, -\boldsymbol{H}_n^{-1}(\boldsymbol{\theta}_n^*)) \qquad \text{for large } n.$$

Standard usage: proposal for independent Metropolis-Hastings (IMH).

- Use t_{df} kernel to ensure the proposal's tails dominate the posterior's.
- Use PSO (or some other optimization algorithm) to find $\boldsymbol{\theta}_n^*$.

Laplace Approximation for IMH within Gibbs

Let $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ and $\boldsymbol{\Sigma}^* = [-\boldsymbol{H}^*(\boldsymbol{\theta}_n^*)]^{-1} = \begin{bmatrix} \boldsymbol{\Sigma}_{11}^* & \boldsymbol{\Sigma}_{12}^* \\ \boldsymbol{\Sigma}_{21}^* & \boldsymbol{\Sigma}_{22}^* \end{bmatrix}$.

Suppose $p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1, \boldsymbol{y}_{1:n})$ is easy to draw from and $p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2, \boldsymbol{y}_{1:n})$ is approximately Gaussian. Then IMH within Gibbs (IMHwG) may work well:

1. Draw $\boldsymbol{\theta}_2^{(t+1)} \sim p(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1^{(t)}, \boldsymbol{y}_{1:n})$.
2. Metropolis step with proposal $\boldsymbol{\theta}_1^{prop} \sim t_{df}(\widetilde{\boldsymbol{\theta}}_1, \widetilde{\boldsymbol{\Sigma}}_{11})$, where

$$\widetilde{\boldsymbol{\theta}}_1 = \boldsymbol{\theta}_1^* + \boldsymbol{\Sigma}_{12}^*(\boldsymbol{\Sigma}_{22}^*)^{-1}(\boldsymbol{\theta}_2^{(t+1)} - \boldsymbol{\theta}_2^*),$$
$$\widetilde{\boldsymbol{\Sigma}}_{11} = \boldsymbol{\Sigma}_{11}^* - \boldsymbol{\Sigma}_{12}^*(\boldsymbol{\Sigma}_{22}^*)^{-1}\boldsymbol{\Sigma}_{21}^*.$$

Uses a global Laplace approx to construct an approx for $p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2, \boldsymbol{y}_{1:n})$.

- Worse than directly approximating $p(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2, \boldsymbol{y}_{1:n})$.
- But for MCMC, it is much cheaper to do the optimization once rather than every iteration.

Spatial Models of County Population Estimates

The American Community Survey (ACS) provides 5-year period estimates of county populations as recently as 2014.

In 2014 there were $n = 3,142$ counties in the United States, including the District of Columbia, Alaska, and Hawaii.

Let Z_i denote the ACS estimate of the population of county i and Y_i denote the true population. Data models:

1. Poisson: $Z_i | \boldsymbol{Y}_{1:n} \sim Pois(Y_i)$.

Process model:

$$\log \boldsymbol{Y}_{1:n} = \boldsymbol{X}\beta + \boldsymbol{S}\delta = \text{fixed effects} + \text{random effects},$$

where $\boldsymbol{X}$ is a $n \times p$ matrix of covariates and $\boldsymbol{S}$ is a $n \times r$ matrix of spatial basis functions.

Spatial Models of County Population Estimates

Process model details

$$\log \boldsymbol{Y}_{1:n} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{S}\boldsymbol{\delta}$$

Define $\boldsymbol{G} = (\boldsymbol{I}_n - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}')\boldsymbol{A}(\boldsymbol{I}_n - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}')$.

$\boldsymbol{A}$ is the binary adjacency matrix: $a_{ij} = 1$ if counties i and j are neighbors, $a_{ij} = 0$ otherwise, and $a_{ii} = 0$ along the diagonal.

$\boldsymbol{S}$ is the r eigenvectors corresponding to the largest r eigenvalues of $\boldsymbol{G}$.

- Forces the spatial model onto the residual variation of $\log \boldsymbol{Y}_{1:n}$.
- Known as the truncated Moran's I basis set.
- Hughes and Haran (2013); Porter et al. (2015); Bradley et al. (2015).

Choose $r \ll n$ so the model is reduced rank (use sensitivity analysis).

Random effects models:

1. iid random effects: $\delta_i \overset{iid}{\sim} N(0, \sigma^2)$,
2. fully correlated random effects: $\boldsymbol{\delta} \sim N(\boldsymbol{0}, \boldsymbol{\Sigma})$.

IMH AND IMHwG acceptance rates

IMH; iid random effects, r = 10

n_{iter}	PSO	DI-PSO	BBPSO	AT-BB	AT-PSO
100	0.28	0.28	0.37	0.33	0.65
500	0.88	0.89	0.27	0.23	0.89
1000	0.89	0.89	0.32	0.28	0.89
1500	0.89	0.89	0.29	0.29	0.88
2000	0.89	0.89	0.27	0.31	0.89

IMHwG; iid random effects, r = 10

n_{iter}	PSO	DI-PSO	BBPSO	AT-BB	AT-PSO
100	0.45	0.03	0.04	0.04	0.21
500	0.97	0.97	0.05	0.02	0.97
1000	0.97	0.96	0.09	0.05	0.97
1500	0.96	0.97	0.10	0.10	0.97
2000	0.96	0.96	0.01	0.01	0.97

IMH; fully correlated random effects, r = 5

n_{iter}	PSO	DI-PSO	BBPSO	AT-BB	AT-PSO
100	0.04	0.03	0.07	0.04	0.02
500	0.04	0.20	0.01	0.08	0.18
1000	0.35	0.29	0.04	0.03	0.45
1500	0.33	0.40	0.01	0.03	0.34
2000	0.24	0.37	0.06	0.05	0.40

IMHwG; fully correlated random effects, r = 5

n_{iter}	PSO	DI-PSO	BBPSO	AT-BB	AT-PSO
100	0.24	0.35	0.25	0.23	0.48
500	0.31	0.71	0.31	0.31	0.86
1000	0.85	0.97	0.36	0.54	0.98
1500	0.97	0.98	0.29	0.63	0.98
2000	0.97	0.97	0.47	0.33	0.97

Spatial Models of County Population Estimates

MCMC Simulations

Effective sample size (n_{eff})

ranef	r	IMH	IMHwG	RWwG	B-RWwG	Stan
iid	10	23170	46177	8072	1168	34682
	20	16958	43005	5739	646	50000
	30	30237	39739	4440	404	50000
full	5	32	47240	8089	2070	28662
	7	37	42459	7811	1743	35298
	9	9	717	8298	1417	30805

Time (seconds) per 10,000 n_{eff}

ranef	r	IMH	IMHwG	RWwG	B-RWwG	Stan
iid	10	24	23	201	146	1911
	20	27	26	506	215	1106
	30	16	24	790	483	980
full	5	14433	21	131	170	2634
	7	11188	20	145	167	1419
	9	32197	876	126	153	1143

Takeaways

- Particle swarm optimization is a useful class of optimization algorithms — robust and easy to use.
- Adaptively tuned PSO algorithms are a new class of PSO algorithms that are competitive with other PSO algorithms, and often outperform them.
- PSO works well to find the posterior mode for the Laplace approximation to a posterior distribution.
- Using the global Laplace approximation to construct an approximation to a conditional posterior for an IMH within Gibbs sampler will often work well under a wider variety of circumstances.

Thank you!

References I

Andrieu, C. and Thoms, J. (2008). A tutorial on adaptive MCMC. *Statistics and Computing*, 18(4):343–373.

Blum, C. and Li, X. (2008). Swarm intelligence in optimization. In Blum, C. and Merkle, D., editors, *Swarm Intelligence: Introduction and Applications*. Springer.

Bradley, J. R., Holan, S. H., and Wikle, C. K. (2015). Multivariate spatio-temporal models for high-dimensional areal data with application to longitudinal employer-household dynamics. *Annals of Applied Statistics*, 9(4):1761–1791.

Clerc, M. and Kennedy, J. (2002). The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *Evolutionary Computation, IEEE Transactions on*, 6(1):58–73.

References II

Eberhart, R. C. and Shi, Y. (2000). Comparing inertia weights and constriction factors in particle swarm optimization. In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, volume 1, pages 84–88. IEEE.

Hughes, J. and Haran, M. (2013). Dimension reduction and alleviation of confounding for spatial generalized linear mixed models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(1):139–159.

Porter, A. T., Holan, S. H., and Wikle, C. K. (2015). Bayesian semiparametric hierarchical empirical likelihood spatial models. *Journal of Statistical Planning and Inference*, 165:78–90.

Schervish, M. J. (1997). *Theory of statistics*. Springer Science & Business Media.