PSU Random Walk

William Waldron, PhD
Mathematical Statistician
Division of Research and Methodology
National Center for Health Statistics

October 24, 2024
Innovations in Sample Design – From Theory to Practice
FCSM 2024 Conference

PSU Random Walk

Background

Background

Many federal nation-wide household surveys
are increasingly being used for **state-level** estimation.

The reliability of state-level estimates differs from national and domain estimation due to *lower degrees of freedom*.

Cross-sectional surveys are often **pooled** together into three-year datasets to improve state-level estimation.

- The sample sizes may be higher, but there are still diminishing returns.
- The *sample design* impacts the reliability of estimates from pooled datasets.

Sampling Background I

We consider cross-sectional surveys under **two-stage designs** when n PSUs are selected from N available PSUs.

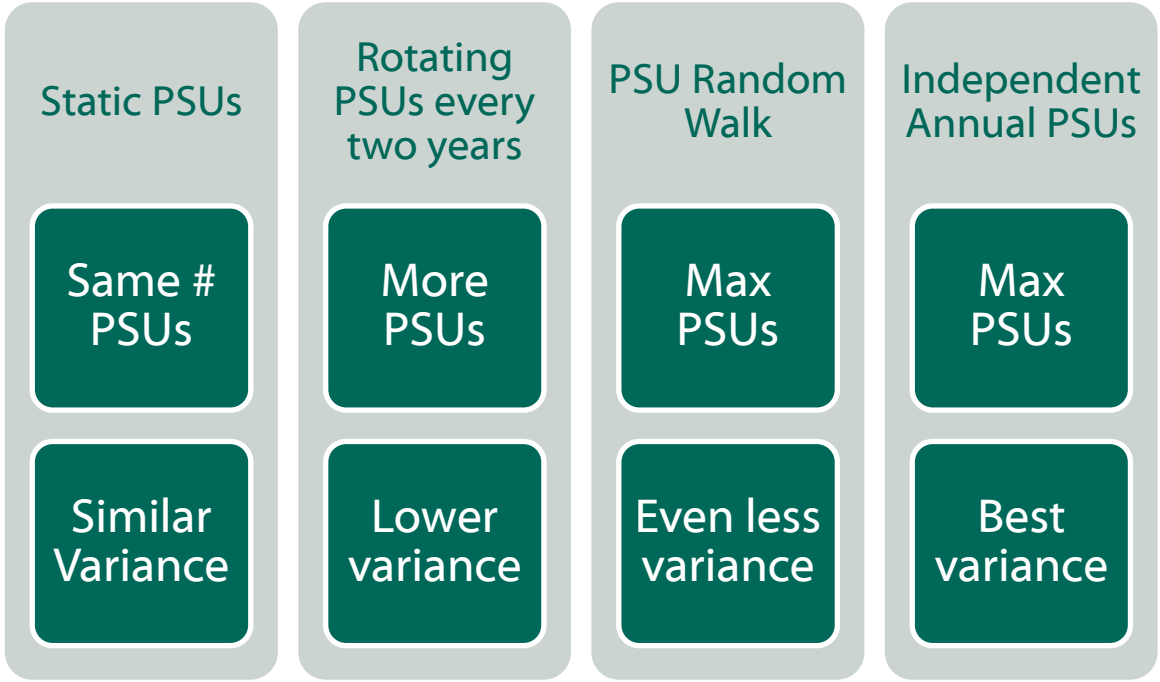
For many applications of federal surveys, PSUs comprise contiguous clusters of counties. This is done to improve sampling and facilitate data collection.

PSU sampling is costly, so PSUs may often repeat in successive years if there are cost savings.

- This does not adversely affect reliability in single-year data but reduces the reliability of survey estimates from the pooled dataset.

Sampling Background II

Comparative Reliability of Pooled Data based on Sample Design



Objective

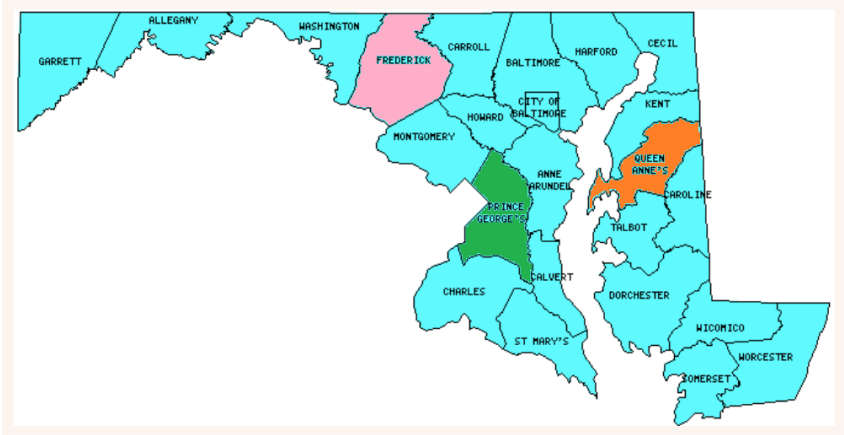
The PSU Random Walk can provide better gains in precision when there are still cost savings in moving PSUs to *adjacent* areas.

We build a *stochastic model* that allows PSUs to “walk” across a geography yet preserves the appropriate unconditional *probability of selection.*

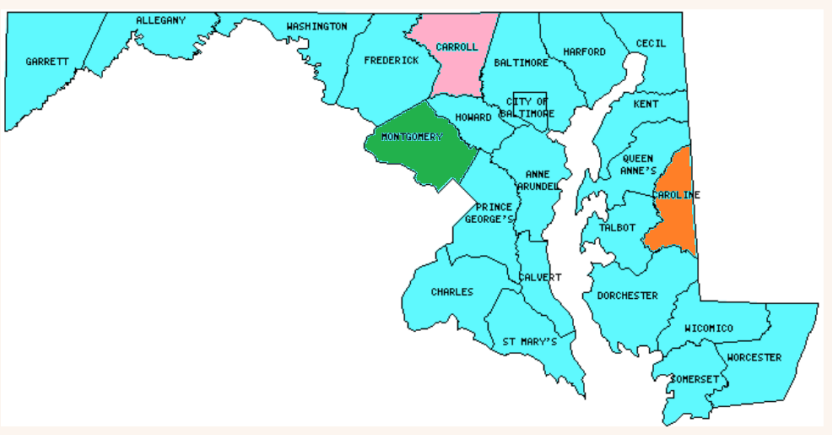
Can we define the *jump probabilities* in a manner that optimizes the variance from first-stage variance estimators? Is this better than just stratifying?

PSU Random Walk Example

Sample # 1

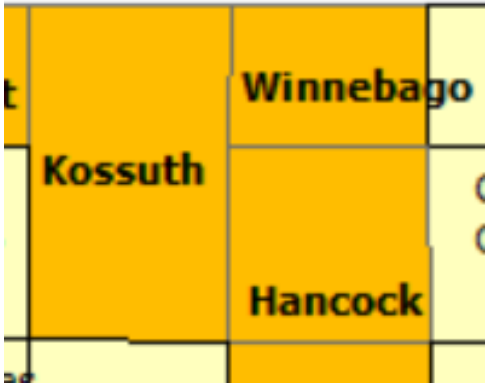


Sample # 2



Question: Can we define the iterative “jump” probabilities, so we are able preserve the appropriate probability of selection π_i in each step?

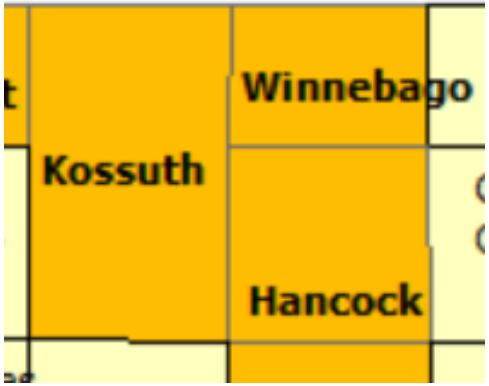
PSU Construction Efficiency



Number PSUs in Three-year data pool.	Sample Size per PSU (3-Year Pool)	Sample Size per PSU (Single Year)
1 Combined PSU (Winnebago/ Kossuth/Hancock	3m – Same PSU sampled three times	m – broader PSU definition (less clustered)
3 PSUs: Winnebago, Kossuth, Hancock	m – New PSU sampled every year	m – more narrow PSU geography (more clustered)

An aggressive design could select more and smaller county-based PSUs that were proximal instead of a single larger PSU comprising contiguous counties.

Jump (Transition) Probabilities



Next PSU Destination

Starting PSU

	Kossuth	Winn-ebago	Hancock
Kossuth	.5	.2	.3
Winn-ebago	.3	.5	.2
Hancock	.3	.2	.5

We have some flexibility when defining the jump probabilities. These conditional probabilities are less important than the unconditional probabilities.

Motivation

The *Method of Maximum Overlap* introduced a conditional probability of selection for PSUs based on exponential sampling (memoryless property) and permanent random numbers that:

1. Preserved the appropriate *unconditional* probability of selection.
2. Increased the likelihood of selecting many of the same PSUs.

PSU Random Walk

Definition

Definition

There are N PSUs in a stratum of which n need to be selected in each iteration. We have n << N and the PSUs are large enough that the ICC is not too high.

Let $X_1, X_2, \dots, X_N$ denote the population of PSUs. An initial sample of n PSUs is taken using standard methods.

Let x_{ij}^k denote the i^{th} PSU selected during the j^{th} sample iteration corresponding to the k^{th} PSU in the initial selection, $k = 1,2, \dots, n$.

Then n independent stochastic processes begin (strands), where we require that x_{ij}^k and $x_{h,j+1}^k$ share a geographical border, otherwise $\Pr\{x_{h,j+1}^k | x_{i,j}^k\} = 0$.

PSU Random Walk Implementation

Step 1. Initially select each PSU independently based on the relative measure of size as the probability of selection (POS).

Step 2. Determine the *transition*, or next step, probabilities of selection that move the active PSUs to an adjacent PSU. Use numerical methods, such as linear programming to preserve the original, or unconditional, probabilities of selection.

Step 3. Iterate each PSU according to the adjacent conditional probabilities of selection. If any two PSUs overlap, continue the iterative sequence on as many PSUs as needed until a without replacement sample is achieved.

Building the Transition Matrix

Suppose there are N PSUs. Let $\pi = \begin{bmatrix} \pi_1 \\ \pi_2 \\ \vdots \\ \pi_N \end{bmatrix}$ denote the vector of probabilities of selection.

We must find a nontrivial $\mathrm{N} \times \mathrm{N}$ matrix $A \neq I$ such that the following hold:

$$\pi' A = \pi'$$

$$0 \leq A_{ij} \leq 1 \; \forall \; 1 \leq i, j \leq n$$

$$\sum_{j=1}^{n} A_{ij} = 1 \; \forall \; i$$

The Probability Transition Matrix

Any matrix satisfying those conditions is called a *probability transition matrix*. We also add the requirement that $A_{ii} = 0$.

- This requirement states that the random walk cannot stay in the same PSU and must transition to another geographical area with each iteration.
- It also means that $\det(A) = 0$ and that A is a singular matrix.

The Probability Transition Matrix

Any matrix satisfying those conditions is called a *probability transition matrix*. We also add the requirement that $A_{ii} = 0$.

- This requirement states that the random walk cannot stay in the same PSU and must transition to another geographical area with each iteration.
- It also means that $\det(A) = 0$ and that A is a singular matrix.

It is not clear how to find such a matrix, in general, if one even exists for a given topology and vector t. Matrix problems usually have the form $Ax = b$, solve for x.

Since the PSU Random Walk will have a finite state space and because every state communicates, we expect to see $\lim_{n\to\infty} A^n = [t', \quad t', \quad t', \dots, t']'$.

Finding the Probability Transition Matrix

Unfurl the matrix A into a vector $v = (a_{11}, a_{12}, \ldots, a_{nn})$.

Create a solution vector b consisting of 1's and π_i's.

Build an appropriate matrix K that forces rows of A to sum to 1 and defines the property that $\pi' A = \pi'$. Thus, there are 2N restrictions.

Restrict the elements of the unknown vector v so that $0 \leq a_{ij} \leq 1$.

Solve the system $Kv = b$. Use a numerical method (e.g., LP) that minimizes the variance of the survey estimates. Reconstruct the matrix A using the solution vector v.

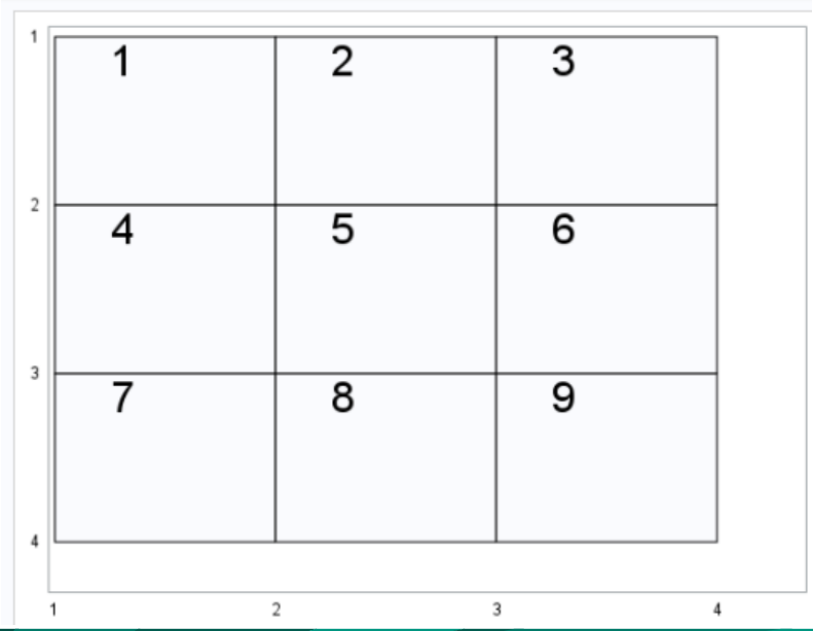
PSU Random Walk

Example

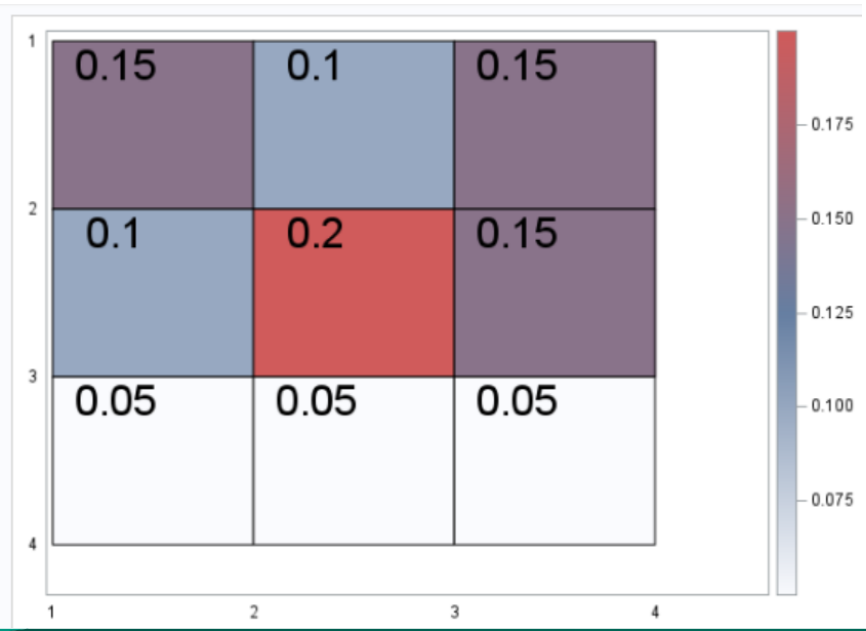
Example: 3×3 Square

Suppose there are nine PSUs arranged in a 3×3 grid with varying probabilities of selection. How to find a probability transition matrix?

PSU Topology



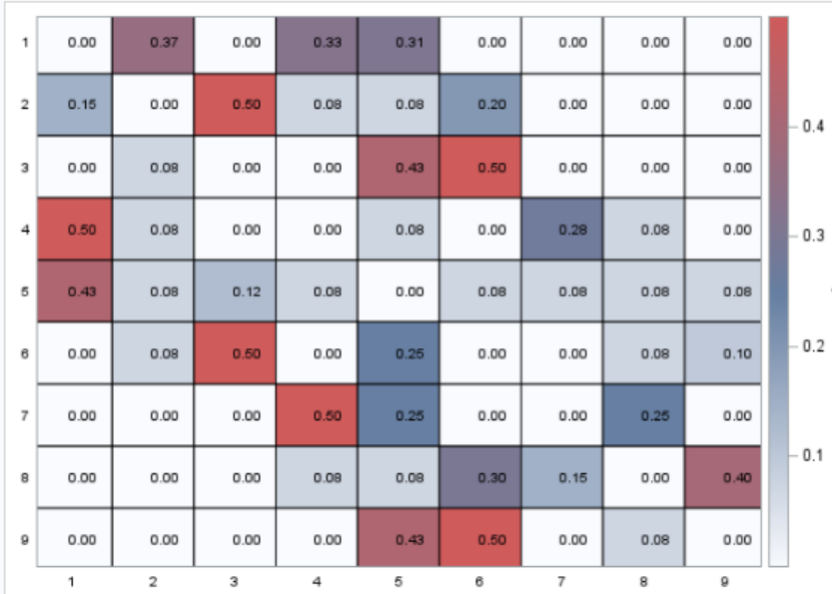
PSU POS



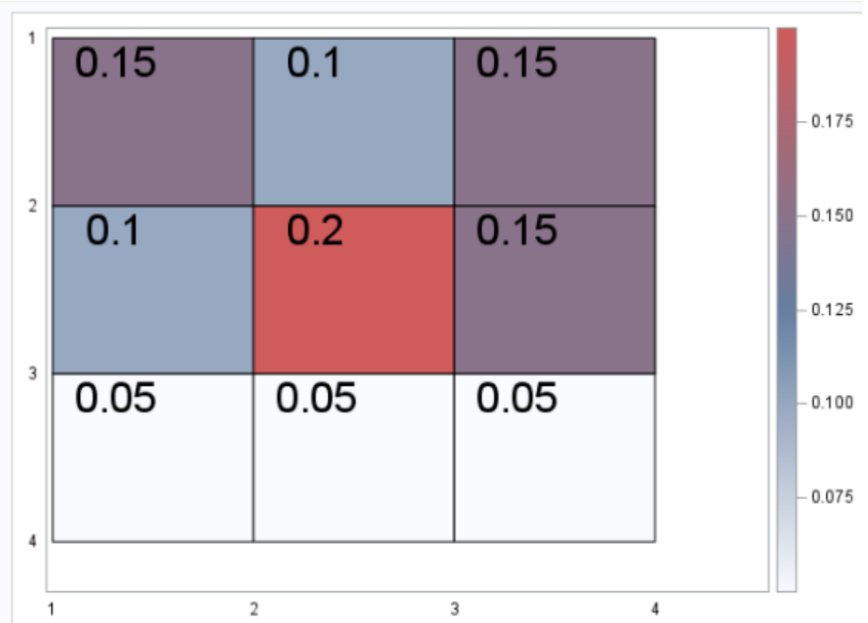
Example: 3×3 Square Solution

A linear programming algorithm was used to solve for the matrix with constraints: $0.075 \leq A_{ij} \leq 0.5$, $t'A = t'$, and $\sum_{j=1}^{n} A_{ij} = 1 \; \forall \; i$.

Transition Matrix



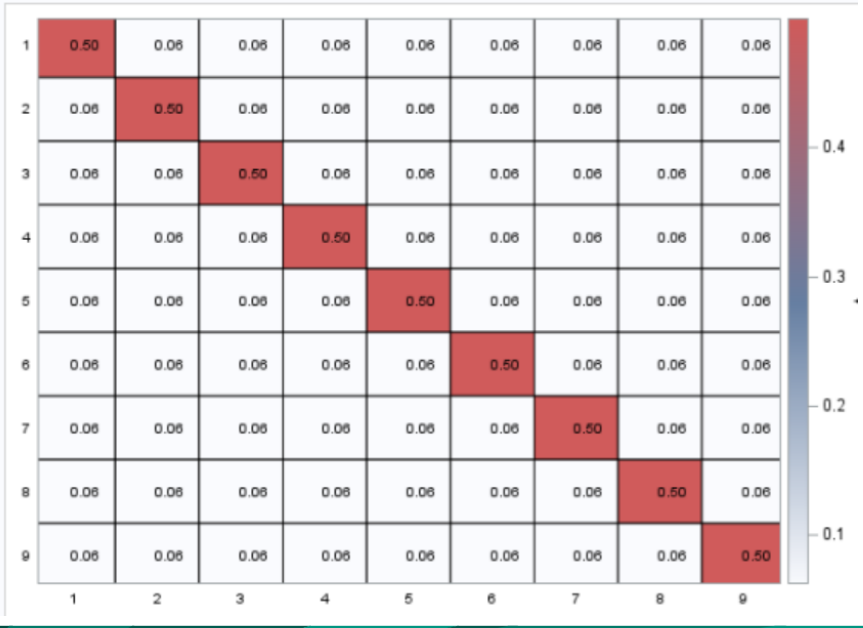
PSU POS



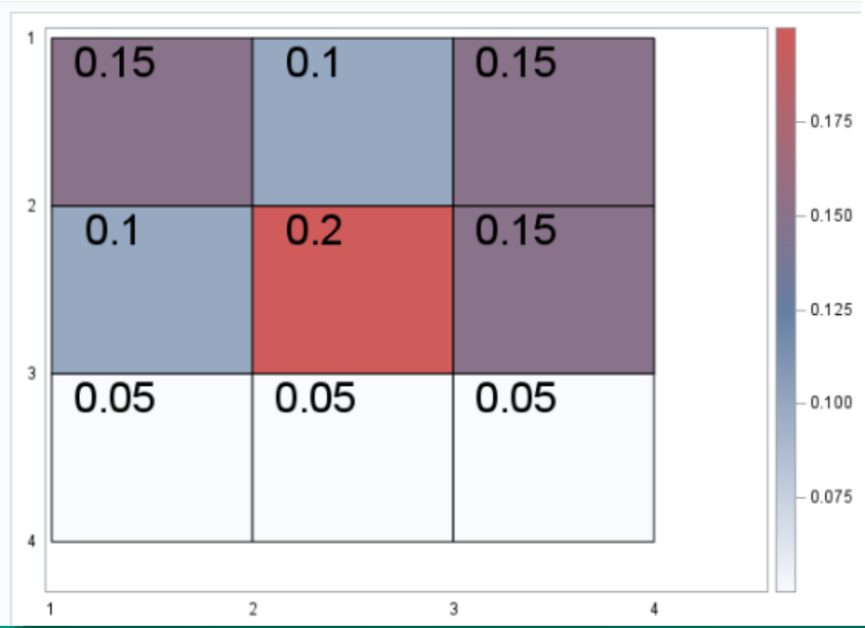
Example: 3×3 Maximum Overlap Solution

Some practitioners advocate for conditional probabilities that promote staying in the same PSU no matter the POS. Permanent random numbers are used to preserve POS.

Transition Matrix



PSU POS



Example: 3×3 Square Limiting Distribution

Transition Matrix A

0	0.3666667	0	0.325	0.3083333	0	0	0	0
0.15	0	0.5	0.075	0.075	0.2	0	0	0
0	0.075	0	0	0.425	0.5	0	0	0
0.5	0.075	0	0	0.075	0	0.275	0.075	0
0.425	0.075	0.125	0.075	0	0.075	0.075	0.075	0.075
0	0.075	0.5	0	0.25	0	0	0.075	0.1
0	0	0	0.5	0.25	0	0	0.25	0
0	0	0	0.075	0.075	0.3	0.15	0	0.4
0	0	0	0	0.425	0.5	0	0.075	0

Example: 3×3 Square POS Computation

POS Computation for Cell 1

$$0.15 = p_1 =$$

$$\sum P(X_{k+1}= 1|X_k = i)P(X_k = i) = \sum A_{i1}p_i = A_{21}p_2 + A_{41}p_4 + A_{51}p_5$$

$$= 0.15 * 0.1 + 0.5 * 0.1 + 0.425 * 0.2$$

$$= 0.15$$

Example: 3×3 Square Limiting Distribution

$$A^2$$

0.3485417	0.0475	0.221875	0.050625	0.051875	0.0964583	0.1125	0.0475	0.023125
0.069375	0.11875	0.109375	0.054375	0.314375	0.255625	0.02625	0.02625	0.025625
0.191875	0.069375	0.340625	0.0375	0.130625	0.046875	0.031875	0.069375	0.081875
0.043125	0.1889583	0.046875	0.316875	0.2341667	0.043125	0.016875	0.074375	0.035625
0.04875	0.1764583	0.075	0.186875	0.2704167	0.1375	0.031875	0.035625	0.0375
0.1175	0.05625	0.06875	0.03	0.26625	0.35625	0.03	0.02625	0.04875
0.35625	0.05625	0.03125	0.0375	0.05625	0.09375	0.19375	0.05625	0.11875
0.069375	0.03375	0.159375	0.080625	0.288125	0.205625	0.02625	0.10125	0.035625
0.180625	0.069375	0.303125	0.0375	0.130625	0.054375	0.043125	0.069375	0.111875

Example: 3×3 Square Limiting Distribution

$$A^4$$

0.2309452	0.0712701	0.1890951	0.0660934	0.1365952	0.1214131	0.0769727	0.0540823	0.0535331
0.1169125	0.1092526	0.1221094	0.1017801	0.2334103	0.1880641	0.0398953	0.0418664	0.0467094
0.1814967	0.0835702	0.217834	0.073818	0.1653189	0.1101915	0.0519949	0.0577363	0.0580395
0.0848883	0.1372927	0.104763	0.1676443	0.2438021	0.1346984	0.0321691	0.0542535	0.0404885
0.1016223	0.1248333	0.1121986	0.1342243	0.2382502	0.1599035	0.0371332	0.0472941	0.0445405
0.1354943	0.0956758	0.1214063	0.0866039	0.2222531	0.2046135	0.0451711	0.0406227	0.0481594
0.2438164	0.0690898	0.1592773	0.0637266	0.1318164	0.1291445	0.0919258	0.0526523	0.0585508
0.1215961	0.1033682	0.1406523	0.1073676	0.2218712	0.1643555	0.0397641	0.0510961	0.0499289
0.1806882	0.0835702	0.2125254	0.0736141	0.1663853	0.1127064	0.0532324	0.0575113	0.0597668

Example: 3×3 Square Limiting Distribution

$$A^8$$

0.1669468	0.0931729	0.1617988	0.0910585	0.1861368	0.1427301	0.0552105	0.0512645	0.051681
0.1430348	0.1025649	0.1439011	0.103016	0.206199	0.1550075	0.0479619	0.0490737	0.0492411
0.1594038	0.0959954	0.1603736	0.0944746	0.1915052	0.1434266	0.0522866	0.0512169	0.0513174
0.1336837	0.1073573	0.1382676	0.1106674	0.2128696	0.1544324	0.045259	0.0492795	0.0481835
0.1385514	0.1049024	0.1411391	0.1067566	0.2093923	0.1546621	0.0466891	0.0491792	0.0487277
0.1468286	0.1008113	0.1457448	0.1005112	0.2034327	0.1548311	0.0491718	0.0490885	0.0495799
0.1689923	0.092483	0.1608727	0.0902864	0.1849481	0.1433578	0.0562444	0.0511103	0.0517052
0.1445553	0.1020779	0.1464813	0.1026497	0.2044407	0.152432	0.0482585	0.0495954	0.0495092
0.1592719	0.0960465	0.1600005	0.0945379	0.191681	0.1437186	0.0522918	0.0511612	0.0512908

Example: 3×3 Square Limiting Distribution

$$A^{16}$$

0.150881	0.0996367	0.1506917	0.0995135	0.1992655	0.1495836	0.0502577	0.0500738	0.0500966
0.14963	0.1001516	0.1497008	0.1002018	0.2003114	0.1501861	0.0498927	0.0499668	0.0499588
0.1505421	0.0997765	0.1504448	0.0997006	0.1995433	0.149727	0.0501558	0.0500487	0.0500611
0.1491235	0.1003637	0.1493157	0.10049	0.2007281	0.1504024	0.0497438	0.0499289	0.0499038
0.1493857	0.100254	0.1495149	0.100341	0.2005123	0.1502903	0.0498209	0.0499486	0.0499323
0.149823	0.1000713	0.1498465	0.1000934	0.2001525	0.1501026	0.0499498	0.0499814	0.0499796
0.1509567	0.0996055	0.1507394	0.0994717	0.1992053	0.149558	0.0502815	0.0500781	0.0501038
0.149723	0.100114	0.1497834	0.1001524	0.2002309	0.150131	0.0499188	0.0499768	0.0499697
0.1505322	0.0997805	0.1504355	0.099706	0.1995519	0.1497332	0.0501531	0.0500476	0.0500599

Example: 3×3 Square Limiting Distribution

$$A^{24}$$

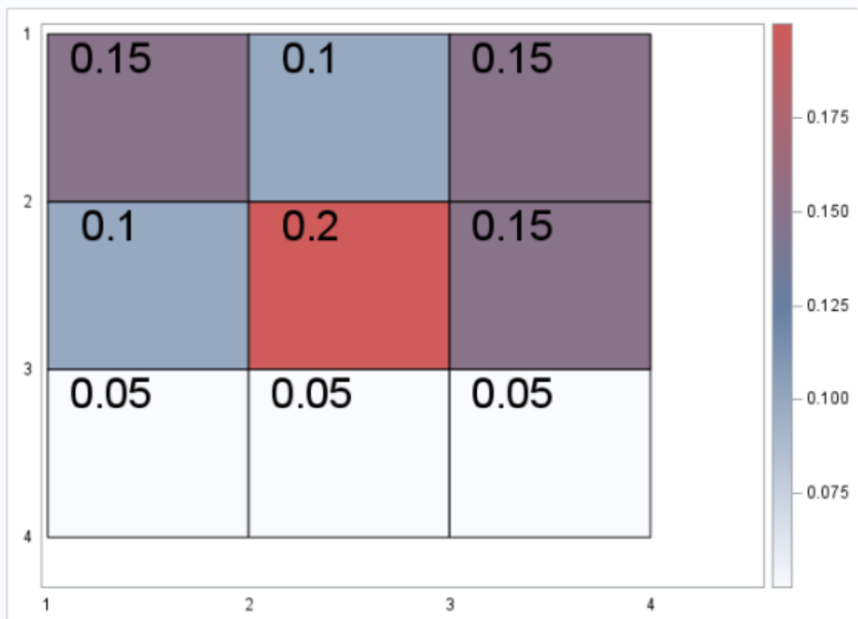
0.150047	0.0999806	0.1500373	0.099974	0.1999607	0.1499775	0.0500137	0.050004	0.0500052
0.1499802	0.1000082	0.1499842	0.1000109	0.2000166	0.1500096	0.0499942	0.0499983	0.0499978
0.1500292	0.099988	0.1500232	0.0999839	0.1999756	0.1499859	0.0500085	0.0500025	0.0500032
0.1499532	0.1000193	0.1499629	0.1000259	0.2000391	0.1500223	0.0499864	0.049996	0.0499948
0.1499672	0.1000135	0.149974	0.1000182	0.2000274	0.1500157	0.0499904	0.0499972	0.0499964
0.1499904	0.1000039	0.1499923	0.1000053	0.200008	0.1500047	0.0499972	0.0499992	0.0499989
0.1500509	0.099979	0.1500403	0.0999718	0.1999575	0.1499757	0.0500148	0.0500043	0.0500056
0.1499852	0.1000061	0.1499883	0.1000082	0.2000123	0.1500071	0.0499957	0.0499987	0.0499984
0.1500286	0.0999882	0.1500228	0.0999842	0.1999761	0.1499862	0.0500083	0.0500024	0.0500032

Example: 3×3 Square Limiting Distribution

A^{48}

0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05
0.15	0.1	0.15	0.1	0.2	0.15	0.05	0.05	0.05

PSU POS



Conclusions

An appropriate probability transition matrix can be constructed, provided the system has multiple options for PSUs to travel for each iteration.

The algorithm may not converge if the $\pi_i's$ are too close to zero or one. Guardrails may be needed to keep jump probabilities viable (not zero or one).

Variance can be improved by attempting to keep the transition matrix as close as possible to the transition matrix based on PPS or equally likely jumps.

The method will have higher degrees of freedom compared to a highly stratified design, meaning the variance estimates will be more stable.

Thank you!

For more information, contact CDC
1-800-CDC-INFO (232-4636)
TTY: 1-888-232-6348 www.cdc.gov