Applications of quasi-Monte Carlo methods in survey inference

Stas Kolenikov

Department of Statistics
University of Missouri-Columbia

SSC meeting, 2009, Vancouver
Setup and notation

- Finite population with $L$ strata and $N_h$ units in each
- Designs: sampling with replacement, $n_h$ units from stratum $h$
- Estimates: $\bar{y}_{st}$ for $\bar{Y}$, $g(\bar{y}_{st})$ for $g(\bar{Y})$, estimating equations
- Estimation of design variance
- “Golden standard”: linearization variance estimator
- Other available methods: replication/resampling
  - Produce $R$ subsamples of the data indexed by $r = 1, \ldots, R$
  - Estimate $\hat{\theta}(r)$ from $r$-th replicate
  - Estimate design variance by

$$v[\hat{\theta}] = \frac{\text{scale}}{R} \sum_{r=1}^{R} (\hat{\theta}(r) - \bar{\hat{\theta}(\cdot)})^2$$
Replication methods

- Balanced repeated replication (McCarthy 1969): use half-samples of the data, estimate, repeat $R$ times, combine results
  Features: $\forall h = 1, \ldots, L n_h = 2, R = 4([L/4] + 1)$ by using Hadamard matrices

- Jackknife (Kish & Frankel 1974, Krewski & Rao 1981): throw one PSU out, estimate, combine
  Features: $R = n$, closest to linearization estimator, inconsistent for non-smooth functions

- Bootstrap (Rao & Wu 1988): resample with replacement $m_h$ units from the available $n_h$ units in stratum $h$
  Features: need internal scaling — best with Rao, Wu & Yue’s (1992) weights; choice of $m_h$; choice of $R$
Pros and cons

+ Only need the software that does weighted estimation — no need for programming specific estimators for each model
+ No need to release the unit identifiers in public data sets
  - Computationally intensive
  - Non-response and post-stratification need to be performed on every set of weights
From BRR to bootstrap

- **Wu (1991), Sitter (1993)** — *first order balance*: each unit is resampled the same number of times
- **Second order balance**: each pair of units is resampled the same number of times
- **Gurney & Jewett (1975)**: orthogonal arrays for \( n_h = p \geq 2, R = p^n \)
- **Wu (1991)**: mixed orthogonal arrays
  Also: quantification of near-orthogonality and resulting biases in estimation
- **Sitter (1993)**: balanced orthogonal multi-arrays, \((p - 1)(L + 1) \leq R \leq (p - 1)(L + 4)\)
- **Nigam & Rao (1996)**: balanced bootstrap (Davison, Hinkley & Schechtman 1986, Graham, Hinkley, John & Shi 1990) for stratified samples — limited set of designs to which the idea is applicable
Which one looks nicer: this...
Resampling inference

Motivation: better balancing (?)

Quasi Monte-Carlo Implementation options

Why should this work?

Hansen-Tepping population simulations

References
Low discrepancy sequences


- For integers $n, b > 2$, if

\[ n = \sum_{j=0}^{\infty} a_j(n)b^j \]

then the \textit{radical inverse function in base $b$} is

\[ \phi_b(n) = \sum_{j=0}^{\infty} a_j(n)b^{-j-1} \in [0, 1) \]

- For an integer $b > 2$, the \textit{van der Corput sequence in base $b$} is

\[ \{\phi_b(n)\}_{n=0}^{\infty} \]

- A multivariate version is \textit{Halton sequence}:

\[ x_n = (\phi_{b_1}(n), \ldots, \phi_{b_s}(n)) \]
## Halton sequence

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_2(n)$</td>
<td>0</td>
<td>0.12 = $\frac{1}{2}$</td>
<td>0.012 = $\frac{1}{4}$</td>
<td>0.112 = $\frac{3}{4}$</td>
<td>0.0012 = $\frac{1}{8}$</td>
</tr>
<tr>
<td>$\phi_3(n)$</td>
<td>0</td>
<td>0.13 = $\frac{1}{3}$</td>
<td>0.23 = $\frac{2}{3}$</td>
<td>0.013 = $\frac{1}{9}$</td>
<td>0.113 = $\frac{4}{9}$</td>
</tr>
<tr>
<td>$\phi_5(n)$</td>
<td>0</td>
<td>0.15 = $\frac{1}{5}$</td>
<td>0.25 = $\frac{2}{5}$</td>
<td>0.35 = $\frac{3}{5}$</td>
<td>0.45 = $\frac{4}{5}$</td>
</tr>
</tbody>
</table>
Measures of discrepancy

For a sequence $P = \{x\}_{n=1}^{\mathcal{N}}$ and a family of sets $\mathcal{B}$

$$A(B, P) = \sum_{n=1}^{\mathcal{N}} I_B(x_n)$$

(number of hits by $P$),

$$D_N(\mathcal{B}; P) = \sup_{B \in \mathcal{B}} \left| \frac{A(B, P)}{\mathcal{N}} - \lambda(B) \right|,$$

$$D^*_N(P) = D_N(\mathcal{I}^*; P), \quad \mathcal{I}^* = \left\{ \prod_{i=1}^{s} [0, u_i), 0 \leq u_i \leq 1 \right\}$$

$$D_N(P) = D_N(\mathcal{I}; P), \quad \mathcal{I} = \left\{ \prod_{i=1}^{s} [u_i, v_i), 0 \leq u_i \leq v_i \leq 1 \right\}$$

$$D^*_N(P) \leq D_N(P) \leq 2^s D^*_N(P)$$

$D^*_N(P)$ is usually referred to as the star discrepancy, and $D_N(P)$, as the extreme discrepancy.
Low discrepancy sequences

- For van der Corput sequence,
  \[
  \lim_{\mathcal{N} \to \infty} \frac{\mathcal{N}}{\ln \mathcal{N}} D^*_\mathcal{N}(S_b) = \lim_{\mathcal{N} \to \infty} \frac{\mathcal{N}}{\ln \mathcal{N}} D_\mathcal{N}(S_b) \leq \frac{b}{4 \ln b}
  \]

- For Halton sequence in pairwise relatively prime bases \(b_1, \ldots, b_s\),
  \[
  D^*_\mathcal{N}(S) < \frac{s}{\mathcal{N}} + \frac{1}{\mathcal{N}} \prod_{i=1}^{s} \left( \frac{b_i - 1}{2 \ln b_i} \ln \mathcal{N} + \frac{b_i + 1}{2} \right) = A(b_1, \ldots, b_s) \mathcal{N}^{-1} \ln^{s} \mathcal{N} + O(\mathcal{N}^{-1} \ln^{s-1} \mathcal{N}) \quad (1)
  \]

- For regular Monte Carlo methods,
  \[D^*_\mathcal{N}(S_{MC}) = O_p(\mathcal{N}^{-1/2})\] which is asymptotically inferior
“Stratified” QMC

- Set up $L$-dimensional Halton sequence
  \[ \{x_k\}, \; k = 1, \ldots, \mathcal{N} \]
- For replication $r = (k - 1) \mod R + 1$, include unit $[n_h x_{hk} + 1]$ into $r$-th resample of $h$-th stratum
- Length of the sequence: $\mathcal{N} = Rm_h \propto b_1 \cdots b_L \Leftarrow$ first order balance
- Feature: $m_h = m \forall h$
- Dimensionality curse: the first few “good” numbers are $2, 6, 30, 210, 2310, 30030, \ldots$
- Option: shuffle dimensions independently
“Data matrix” QMC

- Think of the PSU × replications as a rectangular array: throw points by 2D Halton sequence
- Map $k$-th element of the sequence to replication number $[Rx_{1k} + 1]$ and unit number $[x_{2k}n + 1]$
- QMC balance condition: $N \propto 2 \cdot 3 = 6$
- First order balance condition:
  \[ N = R(m_1 + \ldots + m_L) = \alpha(n_1 + \ldots + n_L) \]
- Option: Force first-order balance by ordering the sequence on $x_2$ and assigning the first $Rm_1/n_1$ to unit 1 in stratum 1, etc. \( \Rightarrow R \propto \text{l.c.m. of } n_1, \ldots, n_h \)
- Another option: shuffle (one of) the dimensions
“Data matrix” QMC
Why should this work?

Simple extension of Wu’s (1991) measure of non-orthogonality/lack of balance, Method 1 (stratified QMC) implementation:

\[
\Delta_{hi} = \frac{\text{# times unit } hi \text{ is used}}{R} - \frac{m_h}{n_h},
\]

\[
\Delta_{hk,ij} = \frac{\text{# times units } hi \text{ and } kj \text{ are jointly used}}{R} - \frac{m_h m_k}{n_h n_k},
\]

\[
|\Delta_{hi}|, |\Delta_{hk,ij}| \leq A(b_1, \ldots, b_s)N^{-1}\ln^s N + O(\cdot)
\]

\[
\approx A(b_1, \ldots, b_s)\left(\frac{\ln R + \ln m_h}{m_h^2 R}\right)^s \rightarrow 0 \text{ as } R \rightarrow \infty
\]

and the estimator will converge to the standard \(v(\bar{y}_{st})\), thus consistent whenever the latter one is.
Really, why?

- For “stratified” QMC implementation, need \( N \gg \exp[s] \) — results in Shao (1996) are derived for
  \[
  R/n^{1+\tau/2} \to 0
  \]
  where \( \tau \) is the excess order above 2 in Liapounov condition for CLT
- For “data matrix” QMC, the discrepancies are still too big — need tighter bounds (Faure sequence)
Simulation set up

- Hansen-Tepping population 1
- 32 strata, $N_h = 10000$, $n_h = 5$
- Scale factors 5, 5
- Correlation 0.5
- Bivariate normal $|h|$
Resampling inference
Motivation: better balancing (?)
Quasi Monte-Carlo
Implementation options
Why should this work?
Hansen-Tepping population simulations
References

\begin{align*}
\text{Kernel density} & \\
& 0.06 \quad 0.08 \quad 0.1 \quad 0.12 \\
& \text{Linearized Jackknife} \\
& \text{Bootstrap, } m_h=1 \\
& \text{Balanced bootstrap, } m_h=1 \\
& \text{Bootstrap, } m_h=n_h-1 \\
& \text{Balanced bootstrap, } m_h=n_h-1
\end{align*}
Motivation:
better balancing (?)

Why should this work?

Hansen-Tepping population simulations

References

\[ \n\]
Resampling inference
Motivation: better balancing (?)
Quasi Monte-Carlo
Implementation options
Why should this work?
Hansen-Tepping population simulations
References
Resampling inference
Motivation: better balancing (?)
Quasi Monte-Carlo
Implementation options
Why should this work?
Hansen-Tepping population simulations
References
Resampling inference

Motivation: better balancing (?)

Quasi Monte-Carlo Implementation options

Why should this work?

Hansen-Tepping population simulations

References

---

**Motivation:**

Quasi Monte-Carlo Implementation options

**Why should this work?:**

Hansen-Tepping population simulations

**References:**
Motivation: better balancing (?)

References

- Hansen-Tepping population simulations

Graph showing kernel density plots for different implementations of Quasi Monte-Carlo (QMC) and resampling methods. The graph compares Linearized Jackknife, 2D QMC, 2D+balanced QMC, 2D+shuffled QMC, and 2D+shuffled+balanced QMC.
## Simulation: ratio

<table>
<thead>
<tr>
<th>Variance estimator</th>
<th>Coverage</th>
<th>Rel. bias</th>
<th>Rel. stab</th>
<th>Std. length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linearized, jknife</td>
<td>94.64</td>
<td>0.56</td>
<td>4.19</td>
<td>0.999</td>
</tr>
<tr>
<td>Bstrap, $m_h = 1$</td>
<td>94.54</td>
<td>2.15</td>
<td>4.82</td>
<td>1.004</td>
</tr>
<tr>
<td>Bal bstrap, $m_h = 1$</td>
<td>94.54</td>
<td>3.23</td>
<td>4.81</td>
<td>1.009</td>
</tr>
<tr>
<td>Bstrap, $m_h = n_h - 1$</td>
<td>94.35</td>
<td>3.79</td>
<td>4.87</td>
<td>1.012</td>
</tr>
<tr>
<td>Bal bstrap, $m_h = n_h - 1$</td>
<td>95.14</td>
<td>2.65</td>
<td>4.83</td>
<td>1.006</td>
</tr>
<tr>
<td>Strat QMC</td>
<td>94.25</td>
<td>1.86</td>
<td>5.16</td>
<td>1.000</td>
</tr>
<tr>
<td>Shuf strat QMC</td>
<td>94.84</td>
<td>3.11</td>
<td>4.80</td>
<td>1.009</td>
</tr>
<tr>
<td>Shuf strat QMC, $R = 600$</td>
<td>95.14</td>
<td>2.66</td>
<td>4.36</td>
<td>1.008</td>
</tr>
<tr>
<td>2D QMC</td>
<td>90.58</td>
<td>-19.59</td>
<td>5.13</td>
<td>0.885</td>
</tr>
<tr>
<td>Bal 2D QMC</td>
<td>88.89</td>
<td>-24.77</td>
<td>5.01</td>
<td>0.856</td>
</tr>
<tr>
<td>Shuf 2D QMC</td>
<td>96.92</td>
<td>27.83</td>
<td>5.20</td>
<td>1.124</td>
</tr>
<tr>
<td>Shuf+bal 2D QMC</td>
<td>95.24</td>
<td>2.18</td>
<td>4.73</td>
<td>1.005</td>
</tr>
</tbody>
</table>
## Simulation: correlation

<table>
<thead>
<tr>
<th>Variance estimator</th>
<th>Coverage</th>
<th>Rel. bias</th>
<th>Rel. stab</th>
<th>Std. length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linearized, jknife</td>
<td>93.35</td>
<td>-1.82</td>
<td>5.04</td>
<td>0.982</td>
</tr>
<tr>
<td>Bstrap, $m_h = 1$</td>
<td>93.35</td>
<td>-1.47</td>
<td>5.26</td>
<td>0.983</td>
</tr>
<tr>
<td>Bal bstrap, $m_h = 1$</td>
<td>93.06</td>
<td>-1.52</td>
<td>5.33</td>
<td>0.982</td>
</tr>
<tr>
<td>Bstrap, $m_h = n_h - 1$</td>
<td>93.75</td>
<td>2.12</td>
<td>5.43</td>
<td>1.000</td>
</tr>
<tr>
<td>Bal bstrap, $m_h = n_h - 1$</td>
<td>93.85</td>
<td>3.32</td>
<td>5.52</td>
<td>1.005</td>
</tr>
<tr>
<td>Strat QMC</td>
<td>93.85</td>
<td>-0.73</td>
<td>5.57</td>
<td>0.984</td>
</tr>
<tr>
<td>Shuf strat QMC</td>
<td>93.25</td>
<td>-1.91</td>
<td>5.29</td>
<td>0.980</td>
</tr>
<tr>
<td>Shuf strat QMC, $R = 600$</td>
<td>93.55</td>
<td>-1.31</td>
<td>4.99</td>
<td>0.985</td>
</tr>
<tr>
<td>2D QMC</td>
<td>90.28</td>
<td>-19.05</td>
<td>5.47</td>
<td>0.885</td>
</tr>
<tr>
<td>Bal 2D QMC</td>
<td>88.69</td>
<td>-23.82</td>
<td>5.42</td>
<td>0.857</td>
</tr>
<tr>
<td>Shuf 2D QMC</td>
<td>95.93</td>
<td>26.24</td>
<td>5.91</td>
<td>1.113</td>
</tr>
<tr>
<td>Shuf+bal 2D QMC</td>
<td>93.45</td>
<td>2.58</td>
<td>5.42</td>
<td>1.002</td>
</tr>
</tbody>
</table>
What I covered was...

1. Resampling inference
2. Motivation: better balancing (?)
3. Quasi Monte-Carlo
4. Implementation options
5. Why should this work?
6. Hansen-Tepping population simulations
7. References
References I


References II


References III


References IV