Fibre Sampling

Where Applied Statistics Meets Algebraic Ring Theory

Martin Hazelton <martin.hazelton@otago.ac.nz>
Department of Mathematics & Statistics, University of Otago

Australian Statistical Conference, Curtin University, 1–5 December 2025

Statistical Inverse Problems

Interest is in a random process that is observed only indirectly

Observations provide incomplete information about target variable

- Aggregated data
- Summarised data
- Corrupted data

Leads to statistical inverse problems

Problems of this sort are ubiquitous in science and engineering

Often arise from automated data collection

Linear Inverse Problems for Count Data

For count data, statistical linear inverse problems characterised by

$$\boldsymbol{y} = A\boldsymbol{x}$$

- $\boldsymbol{x} \in \mathbb{Z}^r_{\geq 0}$ is count vector of interest;
- $\boldsymbol{y} \in \mathbb{Z}^n_{\geq 0}$ is vector of observed counts.
- **Configuration matrix** A is $n \times r$ and non-negative integer elements (often binary).

Typically $r > n$ so linear system will be (heavily) underdetermined.

Aim is to perform inference for $\boldsymbol{x}$ and/or parameter vector $\boldsymbol{\theta}$ describing underlying distribution $f(\boldsymbol{x}|\boldsymbol{\theta})$.

- Prior information or auxiliary data often critical.

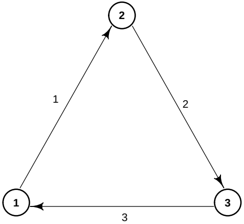
Network Tomography

$\boldsymbol{x}$ vector path traffic volumes; $\theta = \mathrm{E}[\boldsymbol{x}]$.

$\boldsymbol{y}$ traffic counts collected at various network locations.

Inference for $\boldsymbol{x}$ and/or θ is a standard engineering practice:

Example



- Assume travel possible between any of $r = 6$ node pairs by direct paths.
- Traffic counts $\boldsymbol{y} = (y_1, y_2, y_3)^{\mathsf{T}}$ observed on $n = 3$ links.
- Collect path volumes in vector $\boldsymbol{x}$.

$$\boldsymbol{y} = A\boldsymbol{x} \quad \text{where} \quad A = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Resampling Contingency Tables

$\boldsymbol{x}$ cell entries in table.

$\boldsymbol{y}$ marginal totals (or similar).

Resampling entries $\boldsymbol{x}$ conditional on $\boldsymbol{y}$ can be used to perform exact inference, generate confidentialized cross-tabulations of official statistics, etc.

Example (2×3 table)

	y_3	y_4	y_5
y_1	x_1	x_2	x_3
y_2	x_4	x_5	x_6

$$\Rightarrow \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}}_{\boldsymbol{x}}$$

Resampling Contingency Tables

$\boldsymbol{x}$ cell entries in table.

$\boldsymbol{y}$ marginal totals (or similar).

Resampling entries $\boldsymbol{x}$ conditional on $\boldsymbol{y}$ can be used to perform exact inference, generate confidentialized cross-tabulations of official statistics, etc.

Example (2×3 table)

	y_3	y_4	y_5
y_1	x_1	x_2	x_3
y_2	x_4	x_5	x_6

$$\Rightarrow \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}}_{\boldsymbol{y}} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}}_{\boldsymbol{x}}$$

Delete redundant row.

Other Applications

Capture-Recapture Studies in Ecology

- Data collected over a sequence of observational periods.
- $\boldsymbol{y}$ is vector of recorded counts classified by pattern of sightings.
  - E.g. y_{101} count of animals observed in periods 1 and 3 but not 2.
- True pattern of sightings $\boldsymbol{x}$ differs from $\boldsymbol{y}$ due to misidentifications.
- Related to multi-list problems in public health.

Biosecurity Surveillance

- Inspection schemes for mail items stratified by expected risk.
- Each item classified by unknown true compliance status, inclusion/exclusion and compliance assessment at each stage.
- Cross-classification generates table with cell counts $\boldsymbol{x}$, but we can observe only certain sums $\boldsymbol{y}$ of these entries.

The Conditional Distribution of $\boldsymbol{x}$

Inference for $\boldsymbol{x}$ based on conditional distribution $f(\boldsymbol{x}|\,\boldsymbol{y})$.

- Dependence of f on parameter θ suppressed for notational convenience.

Courtesy of fundamental equation $\boldsymbol{y} = A\boldsymbol{x}$,

$$f(\boldsymbol{x}|\,\boldsymbol{y}) = \frac{f(\boldsymbol{x})f(\boldsymbol{y}|\boldsymbol{x})}{f(\boldsymbol{y})} = \frac{f(\boldsymbol{x})I_{\{\boldsymbol{y}=A\boldsymbol{x}\}}}{f(\boldsymbol{y})}$$

Normalizing constant is $f(\boldsymbol{y}) = \sum_{\boldsymbol{x}\in\mathcal{F}_{A,\boldsymbol{y}}} f(\boldsymbol{x})$.

Here $\mathcal{F}_{A,\boldsymbol{y}} = \{\boldsymbol{x} \colon \boldsymbol{y} = A\boldsymbol{x}\} \cap \mathbb{Z}^r_{\geq 0}$.

This is solution set is called the ***y*-fibre**.

The Geometry of a $\boldsymbol{y}$-fibre

2×3 contingency table example

	2	4	2
3	x_1	x_2	x_3
5	x_4	x_5	x_6

$\boldsymbol{y} = (3, 5, 2, 4)^{\mathsf{T}}$. (Recall y_5 constraint redundant.)

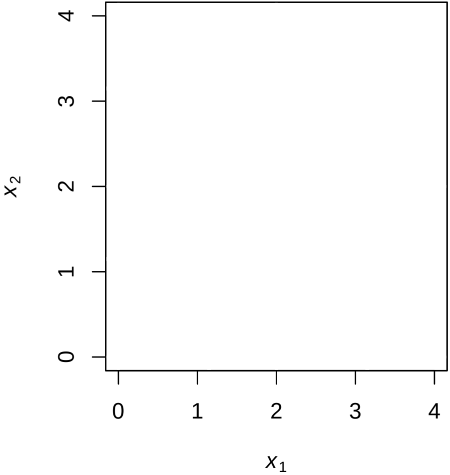
Set of feasible counts $\mathcal{F}_{A,\boldsymbol{y}} = \{\boldsymbol{x} \colon \boldsymbol{y} = A\boldsymbol{x}\} \cap \mathbb{Z}^r_{\geq 0}$ can be fully specified by values of x_1, x_2.

Constraints on these entries:

- $0 \leq x_1 \leq 2$
- $0 \leq x_2 \leq 4$
- $x_1 + x_2 \leq 3$
- $1 \leq x_1 + x_2$

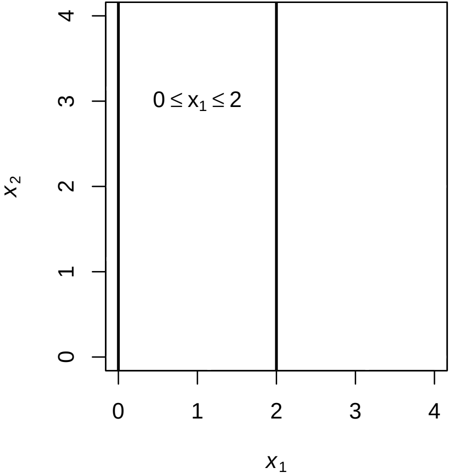
The Geometry of a *y*-fibre

Constructing the fibre for the 2×3 contingency table example



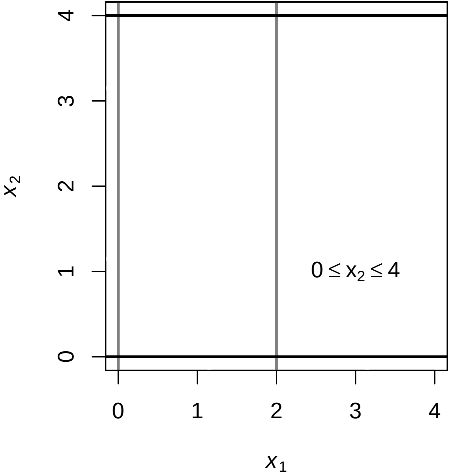
The Geometry of a *y*-fibre

Constructing the fibre for the 2×3 contingency table example



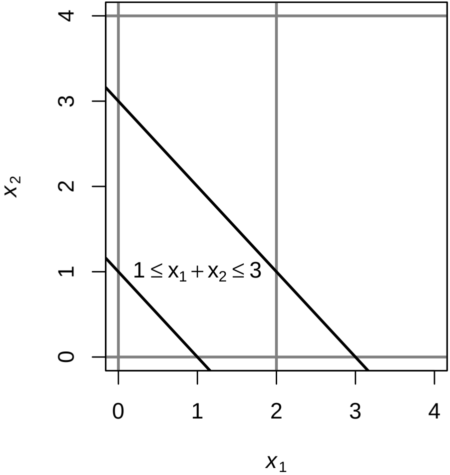
The Geometry of a *y*-fibre

Constructing the fibre for the 2×3 contingency table example



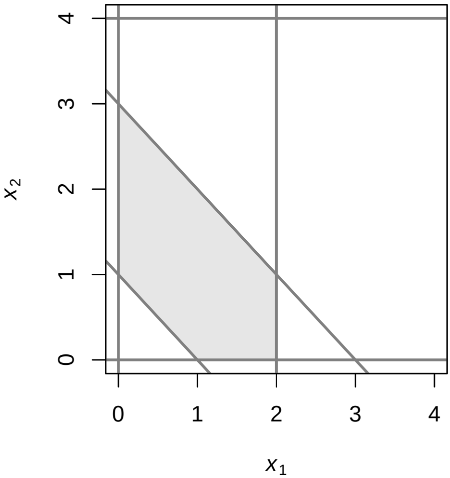
The Geometry of a *y*-fibre

Constructing the fibre for the 2×3 contingency table example



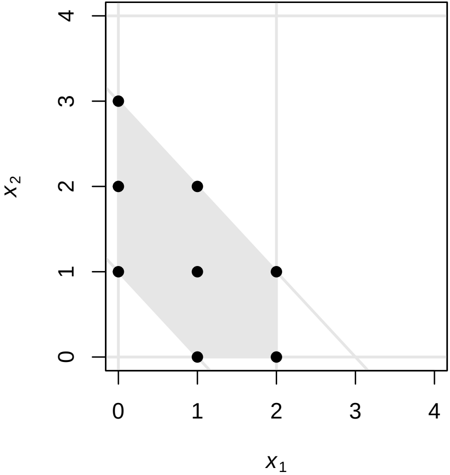
The Geometry of a y-fibre

Constructing the fibre for the 2×3 contingency table example



The Geometry of a *y*-fibre

Constructing the fibre for the 2×3 contingency table example



$\mathbb{Z}$-Polytopes

Continuous version of $\boldsymbol{y}$-fibre is $\{\boldsymbol{x} : \boldsymbol{y} = A\boldsymbol{x}, \boldsymbol{x} \geq \mathbf{0}\}$.

This is intersection of linear manifold $\{\boldsymbol{x} : \boldsymbol{y} = A\boldsymbol{x}\}$ with non-negative orthant $\{\boldsymbol{x} \geq \mathbf{0}\}$.

Hence $\{\boldsymbol{x} : \boldsymbol{y} = A\boldsymbol{x}, \boldsymbol{x} \geq \mathbf{0}\}$ is a convex polytope.

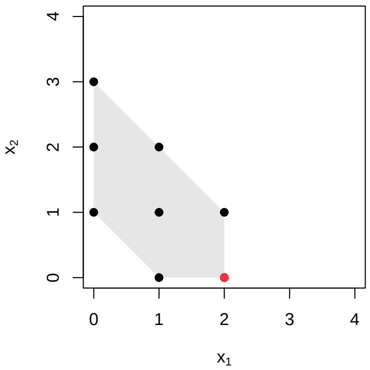
Follows that fibre $\mathcal{F}_{A,\boldsymbol{y}} = \{\boldsymbol{x} \colon \boldsymbol{y} = A\boldsymbol{x}\} \cap \mathbb{Z}^r_{\geq 0}$ is a **$\mathbb{Z}$-polytope**.

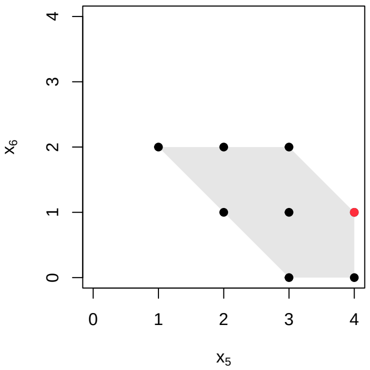
Assuming A of full rank, then $\mathcal{F}_{A,\boldsymbol{y}}$ is an $r - n$ dimensional object embedded in r-dimensional space.

Have flexibility in representation.

Different Projections of a Polytope

2×3 contingency table example: $r = 6$ and $r - n = 2$

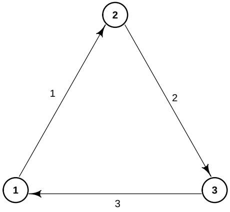




Red points correspond to table $\boldsymbol{x} = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 4 & 1 \end{bmatrix}$

Different Projections of a Polytope

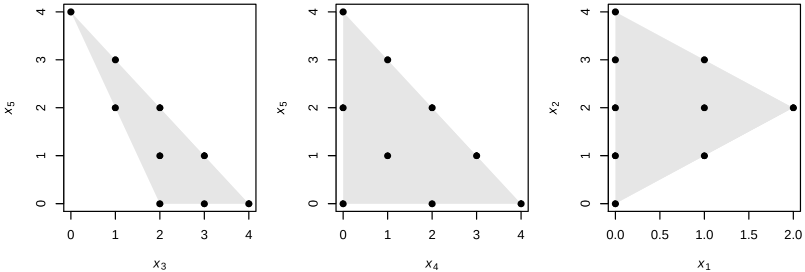
Circuit network example: $r = 5$ and $r - n = 2$



Like earlier example, but last route deleted.

Configuration matrix $A = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$

Traffic counts $\boldsymbol{y} = (4, 4, 4)^\mathsf{T}$ observed.



Inference

Likelihood is $L(\boldsymbol{\theta}) = f(\boldsymbol{y}|\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{F}_{A,\boldsymbol{y}}} f(\boldsymbol{x}|\boldsymbol{\theta})$

Hence direct resampling of $\boldsymbol{x}$ and likelihood-based inference for $\boldsymbol{\theta}$ both require knowledge of $\mathcal{F}_{A,\boldsymbol{y}}$...

... but fibres usually far too large to enumerate.

Example: how many tables on the same fibre?

	Hair				
Eyes	Black	Brunette	Red	Blond	Total
Brown	68	119	26	7	220
Blue	20	84	17	94	215
Hazel	15	54	14	10	93
Green	5	29	14	16	64
Total	108	286	71	127	592

Inference

Likelihood is $L(\boldsymbol{\theta}) = f(\boldsymbol{y}|\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{F}_{A,\boldsymbol{y}}} f(\boldsymbol{x}|\boldsymbol{\theta})$

Hence direct resampling of $\boldsymbol{x}$ and likelihood-based inference for $\boldsymbol{\theta}$ both require knowledge of $\mathcal{F}_{A,\boldsymbol{y}}$...

... but fibres usually far too large to enumerate.

Example: how many tables on the same fibre?

	Hair				
Eyes	Black	Brunette	Red	Blond	Total
Brown	68	119	26	7	220
Blue	20	84	17	94	215
Hazel	15	54	14	10	93
Green	5	29	14	16	64
Total	108	286	71	127	592

Answer: 1,225,914,276,276,768,514

MCMC Based Inference

Problem 1: Resampling $\boldsymbol{x}$ for fixed θ.

- Applications: contingency table resampling, stochastic EM algorithm

Problem 2: Posterior inference for θ.

- Sampling $f(\theta|\boldsymbol{x})$ typically straightforward by Gibbs, Metropolis-Hastings algorithms.
- Iterate sampling from $f(\boldsymbol{x}|\boldsymbol{y},\theta)$ with sampling from $f(\theta|\boldsymbol{x})$.
- Sampling $f(\boldsymbol{x}|\boldsymbol{y},\theta)$ is challenging step.

Random Walk Fibre Samplers

Hit and Run Algorithm

Want to sample $f(\boldsymbol{x}|\boldsymbol{y})$ (parameter dependence suppressed)

Recall that support of $f(\boldsymbol{x}|\boldsymbol{y})$ is $\mathbb{Z}$-polytope $\mathcal{F}_{A,\boldsymbol{y}}$.

Will adopt random walk Metropolis-Hastings sampler.

input

Current state $\boldsymbol{x}$

generate candidate $\boldsymbol{x}^\dagger$

Draw $\boldsymbol{z}$ from set $\mathcal{S} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_M\}$ of possible moves

Draw step size $b \in \mathbb{Z}$

Define candidate $\boldsymbol{x}^\dagger = \boldsymbol{x} + b\boldsymbol{z} \sim q(\cdot|\boldsymbol{x})$

return $\boldsymbol{x}^\dagger$

accept/reject

Compute $\alpha = \mathbf{1}_{\mathcal{F}_{A,\boldsymbol{y}}}(\boldsymbol{x}^\dagger) \min\left\{1, \frac{f(\boldsymbol{x}^\dagger|\boldsymbol{\theta})q(\boldsymbol{x}|\boldsymbol{x}^\dagger)}{f(\boldsymbol{x}|\boldsymbol{\theta})q(\boldsymbol{x}^\dagger|\boldsymbol{x})}\right\}$

Update $\boldsymbol{x} \leftarrow \boldsymbol{x}^\dagger$ with probability α

return $\boldsymbol{x}$

All the Right Moves

Focus for now on move directions; set move length $b = 1$.

Random walk sampler draws moves from set $\mathcal{S} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_M\}$.

If a move $\boldsymbol{z}$ is to have any chance of acceptance, require:

1. $A\boldsymbol{x}^\dagger = A(\boldsymbol{x} + \boldsymbol{z}) = \boldsymbol{y}$
   $\Rightarrow A\boldsymbol{z} = 0$.
   - That is, $\boldsymbol{z} \in \ker_{\mathbb{Z}}(A) = \ker(A) \cap \mathbb{Z}^r$.
2. $\boldsymbol{x} + \boldsymbol{z} \geq \boldsymbol{0}$.
   - Inequality interpreted elementwise (here and henceforth)

Lattice Bases

A set $\mathcal{S} = \{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M\}$ is **lattice basis** if every $\boldsymbol{z} \in \ker_{\mathbb{Z}}(A)$ can be written as a unique integer combination of the basis vectors.

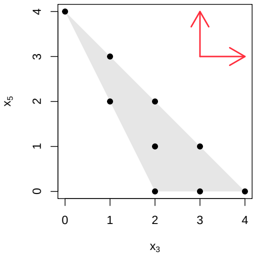
Example

Consider matrices

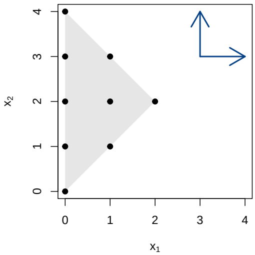
$$U_1 = \begin{bmatrix} -1 & -1 \\ -1 & 0 \\ 1 & 0 \\ 2 & 1 \\ 0 & 1 \end{bmatrix} \qquad U_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & -1 \\ -1 & -1 \\ -1 & 1 \end{bmatrix}$$

In each case, matrix columns form a lattice basis for $\mathcal{F}_{A,\boldsymbol{y}}$ in circuit network tomography problem whenever $\boldsymbol{y} > 0$.

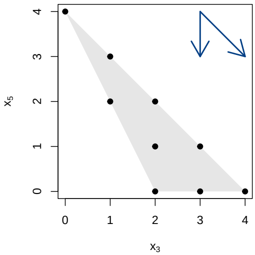
U_1 basis
x_3, x_5 coordinates



U_2 basis
x_1, x_2 coordinates



U_2 basis
x_3, x_5 coordinates



Partition Lattice Bases

Generated using Hermite normal form of A

Partition $A = [A_1|A_2]$ with $n \times n$ matrix A_1 invertible.

- Partition $\boldsymbol{x} = [\boldsymbol{x}_1|\boldsymbol{x}_2]$ likewise.

Define matrix

$$U = \begin{bmatrix} -A_1^{-1}A_2 \\ I_{r-n} \end{bmatrix}$$

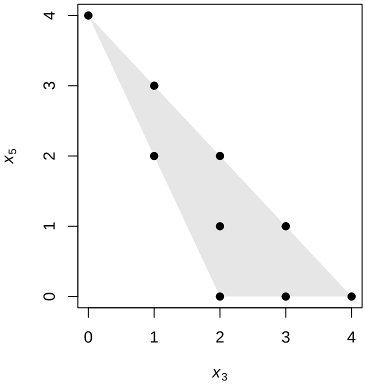
Then

$$AU = [A_1|A_2] \begin{bmatrix} -A_1^{-1}A_2 \\ I_{r-n} \end{bmatrix} = -A_2 + A_2 = 0$$

Hence columns $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{r-n} \in \ker_{\mathbb{Z}}(A)$ and so form a (partition) lattice basis if all components are integers; $|A_1| = \pm 1$ is sufficient.

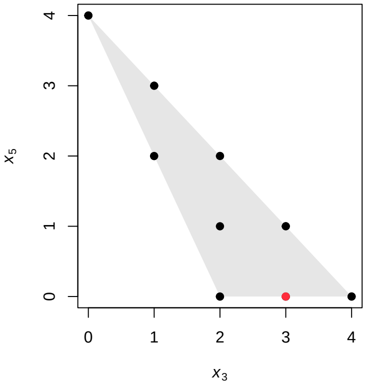
Moves $\pm \boldsymbol{u}_i$ correspond to steps in coordinate directions in polytope projection onto column space of A_2.

Application to Circuit Network Fibre Walk



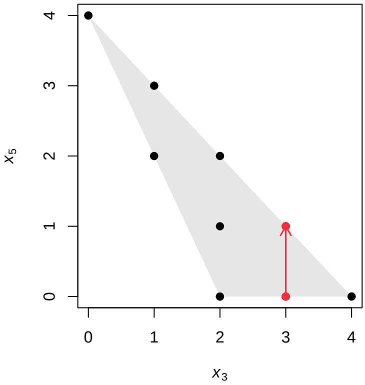
- Lattice basis U_1 comprises moves in coordinate directions.

Application to Circuit Network Fibre Walk



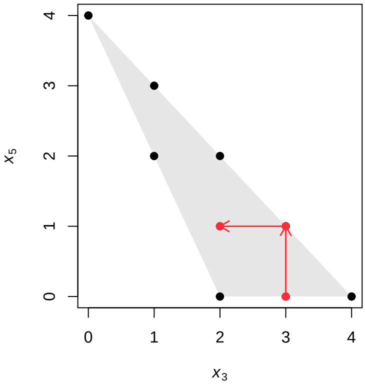
- Lattice basis U_1 comprises moves in coordinate directions.

Application to Circuit Network Fibre Walk



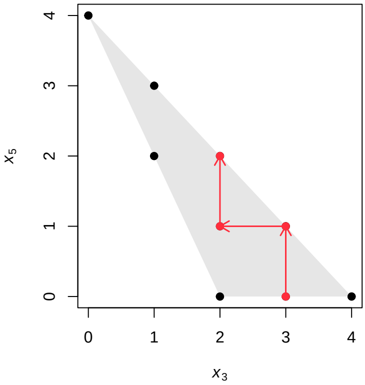
- Lattice basis U_1 comprises moves in coordinate directions.

Application to Circuit Network Fibre Walk



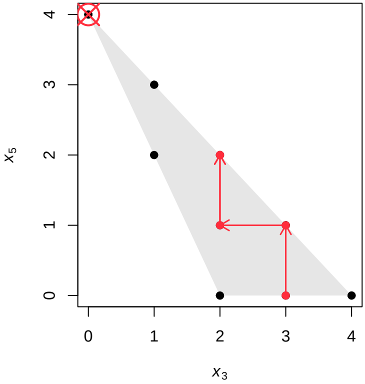
- Lattice basis U_1 comprises moves in coordinate directions.

Application to Circuit Network Fibre Walk



- Lattice basis U_1 comprises moves in coordinate directions.

Application to Circuit Network Fibre Walk



- Lattice basis U_1 comprises moves in coordinate directions.
- Random walk with directions U_1 cannot visit all points.

Connectedness

Irreducibility of random walk required for convergence to target posterior.

This requires that all elements of $\mathcal{F}_{A,\boldsymbol{y}}$ are accessible.

In other words, the MCMC sampler must be **connected**.

Connectedness can be very difficult to check in practice.

As we saw, lattice bases generally do not guarantee connectedness.

Markov Bases and Sub-Bases

Markov sub-basis

A set of moves $\mathcal{M}_{\mathcal{F}_{A,\boldsymbol{y}}} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_L\}$ is a Markov sub-basis if it guarantees existence of walk between any pair of points on $\mathcal{F}_{A,\boldsymbol{y}}$.

Markov basis

A set of moves $\mathcal{M}_A$ is a Markov basis if it guarantees existence of walk between any pair of points on any fibre (i.e. for all $\boldsymbol{y} \geq \mathbf{0}$).

Computing Markov (sub-)bases generally very difficult.

Most successful approach to date uses **algebraic statistics** (Diaconis and Sturmfels, 1998).

Idea is to represent $\boldsymbol{x} \geq \mathbf{0}$ by monomial:

$$T(\boldsymbol{x}) := \boldsymbol{t}^{\boldsymbol{x}} = t_1^{x_1} t_2^{x_2} \cdots t_r^{x_r}$$

A move $\boldsymbol{z}$ represented by monomial difference $\boldsymbol{t}^{\boldsymbol{z}^+} - \boldsymbol{t}^{\boldsymbol{z}^-}$ where $\boldsymbol{z}^+$ and $\boldsymbol{z}^-$ contain respectively positive and negative parts of $\boldsymbol{z}$.

Diaconis, P., & Sturmfels, B. (1998). *The Annals of Statistics*, **26(1)**, 363–397.

Circuit Network Example

$\boldsymbol{x} = (1, 3, 1, 0, 2)^{\mathsf{T}}$ represented by $t_1 t_2^3 t_3 t_5^2$.

Consider move $\boldsymbol{z} = (-1, 0, 0, 1, 1)^{\mathsf{T}}$

- $\boldsymbol{z}^+ = (0, 0, 0, 1, 1)^{\mathsf{T}}$
- $\boldsymbol{z}^- = (1, 0, 0, 0, 0)^{\mathsf{T}}$

So representation of $\boldsymbol{z}$ is $t_4 t_5 - t_1$.

Naturally $-\boldsymbol{z} = (1, 0, 0, -1, -1)^{\mathsf{T}}$ represented by $t_1 - t_4 t_5$.

WTF? (Why That Formulation?)

A move $\boldsymbol{z} \in \ker_{\mathbb{Z}}(A)$ is feasible from point $\boldsymbol{x}$ if $\boldsymbol{x} + \boldsymbol{z} \geq \mathbf{0}$...

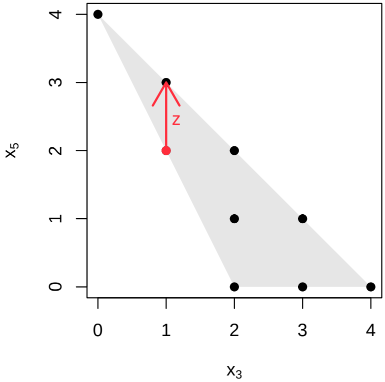
if $\boldsymbol{x} + \boldsymbol{z}^{+} - \boldsymbol{z}^{-} \geq \mathbf{0}$...

if $\boldsymbol{x} - \boldsymbol{z}^{-} \geq \mathbf{0}$...

if $\boldsymbol{t}^{\boldsymbol{x}}$ is divisible by $\boldsymbol{t}^{\boldsymbol{z}^{-}}$.

Similarly $-\boldsymbol{z}$ feasible if $\boldsymbol{t}^{\boldsymbol{x}}$ divisible by $\boldsymbol{t}^{\boldsymbol{z}^{+}}$.

Circuit Network Example Revisited



$\boldsymbol{x} = (1, 3, 1, 0, 2)^\mathsf{T}$
Represented by $t_1 t_2^3 t_3 t_5^2$

$\boldsymbol{z} = (-1, 0, 0, 1, 1)^\mathsf{T}$
Represented by $t_4 t_5 - t_1$

$t_1 t_2^3 t_3 t_5^2$ divisible by $\boldsymbol{t}^{\boldsymbol{z}^-} = t_1$ so move is feasible

$t_1 t_2^3 t_3 t_5^2$ not divisible by $\boldsymbol{t}^{\boldsymbol{z}^+} = t_4 t_5$ so move $-\boldsymbol{z}$ not feasible

Polynomial Rings and Gröbner Bases

Follows that walk between two points on fibre can be analysed as sequence of polynomial long divisions.

This mathematics is well understood through study of polynomial rings, particularly toric ideals.

In principle, can generate sufficient set of divisors using theory of **Gröbner bases**.

A polynomial Gröbner basis defines a Markov basis to ensure connectivity of our fibre samplers.

Implemented using e.g. `4ti2` software.

But that's far from the end of the story.

---

https://github.com/4ti2/4ti2

Mixing ~~Properties~~ Problems with Markov Bases

Samplers using full Markov bases often mix **very** poorly.

Full Markov bases can be huge.

- For given $\boldsymbol{y}$, Markov basis typically contains many useless moves.
- May wait a long time to select essential move.

Stanley, C., & Windisch, T. (2018) prove arbitrarily slow convergence to uniform target $f(\boldsymbol{x} \mid \boldsymbol{y})$ in large problems.

Situation more hopeful in applications where small Markov bases available (H et al., 2023).

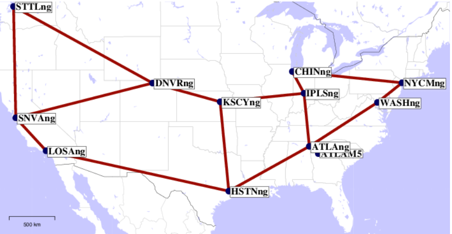
Stanley, C., & Windisch, T. (2018). *Advances in Applied Mathematics*, **92**, 122-143.
Hazelton, M., McVeagh, M., Tuffley, C., & van Brunt, B. (2023). *Bernoulli*, **30(4)**, 2676–2699.

Examples of unwieldy Markov bases

Two-Way Contingency Tables

- Markov basis for $I \times J$ table has $\frac{1}{4}IJ(I-1)(J-1)$ elements.
- So for 20×20 table, Markov basis has more than $35,000$ elements.

Abilene Network Tomography



- 12 nodes, $r = 132$ paths, $n = 42$ links.
- Using `4ti2`, took 9+ hours to find Markov basis with 10,705 vectors.

Dynamic Markov Bases

Idea is to avoid computing full Markov basis *ab initio*.

At each step, find a suitable set of 'local moves'.

So long as union of all such sets forms a Markov basis (in a sensible way), the resulting random walk should be connected.

Seminal work in this area by Dobra (2012) specific to contingency tables, and ignored geometry of polytopes.

We seek geometrically aware dynamic Markov basis using collections of lattice bases.

Dobra, A. (2012). *Journal of Computational and Graphical Statistics*, **21(2)**, 496–517.

Lattice Bases as Local Moves

In H et al. (2021) we propose using lattice bases to provide sets of local moves.

Recall lattice bases not unique.

Let π denote a partition of $\{1, \ldots, r\}$ into two subsets, K_1 and K_2, of size n and $r - n$ respectively.

Denote A_i^π submatrix of A formed by columns indexed by K_i, $i = 1, 2$.

Let $\Pi = \{\pi \colon |A_1^\pi| \neq 0\}$.

For $\pi \in \Pi$, lattice basis $\mathcal{B}_L^\pi$ defined by columns of

$$U^\pi = \begin{bmatrix} -(A_1^\pi)^{-1} A_2^\pi \\ I_{r-n} \end{bmatrix}$$

Have assumed pro tem that $|A_1^\pi| = \pm 1$.

Hazelton, M., McVeagh, M., and Van Brunt, B. (2021). *Biometrika*, **108(3)**, 609–626.

Recall:

- $\mathcal{B}_L^\pi$ is lattice basis corresponding to partition $\pi \in \Pi$
- Π set of partitions for which A_1^π is invertible.

Definition (Unimodular Matrix)

A matrix A is unimodular if every invertible maximal square submatrix of A has determinant ± 1.

Theorem

If A unimodular then $\bigcup_{\pi \in \Pi} \mathcal{B}_L^\pi$ *is a Markov basis.*

Designing a Dynamic Lattice Basis Sampler

Look at Markov process $\{(\boldsymbol{x}^t, \pi^t)\colon t = 1, 2, \ldots\}$.

Let conditional distribution of π^t depend on π^{t-1} but not $\boldsymbol{x}^{t-1}$, to avoid upsetting balance equations.

Connectedness of walk is assured if all $\pi \in \Pi$ have non-zero probability.

Naive approach: randomly select π from Π at each iteration. But...

1. Need to recalculate lattice bases from scratch unacceptably slow.
2. Does not take account of polytope geometry to facilitate mixing.

New Lattice Bases Via Single Column Updating

Update π be potentially exchanging swapping a pair of columns i and j between K_1 and K_2.

Then current lattice basis U can be updated to

$$\tilde{U} = \begin{bmatrix} -\tilde{C} \\ I \end{bmatrix}$$

where $C = A_1^{-1} A_2$, and updated version is

$$\tilde{C} = C - \frac{1}{c_{ij}} (\boldsymbol{c}_j - \boldsymbol{e}_i)(\boldsymbol{c}_i + \boldsymbol{e}_j)^{\mathsf{T}}$$

courtesy of the Sherman-Morrison formula.

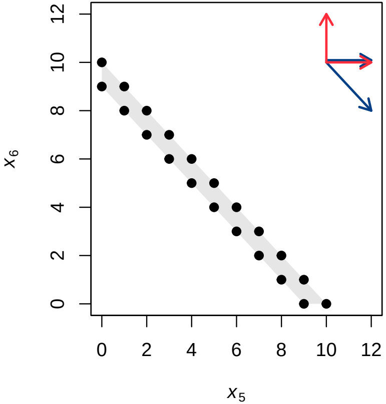
Note that the interchange of columns is feasible if and only if $c_{ij} \neq 0$ (required to ensure A_1 remains invertible).

Geometrically Aware Lattice Bases

	1	10	10
11	x_1	x_2	x_3
10	x_4	x_5	x_6

Colour identifies moves in two different lattice bases.

Choice of basis affects rate of mixing of sampler.

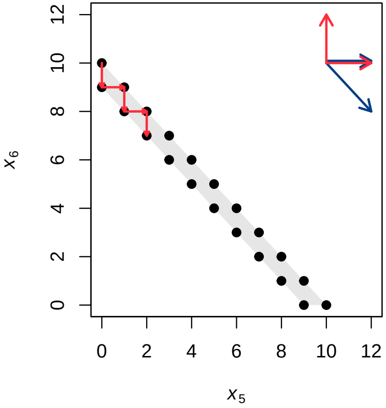


Geometrically Aware Lattice Bases

	1	10	10
11	x_1	x_2	x_3
10	x_4	x_5	x_6

Colour identifies moves in two different lattice bases.

Choice of basis affects rate of mixing of sampler.

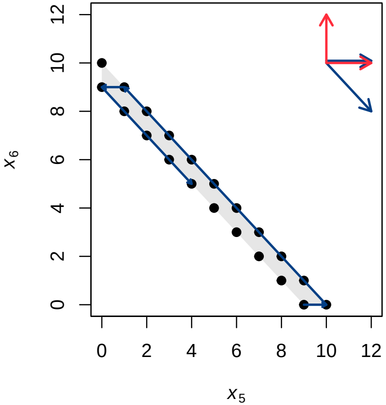


Geometrically Aware Lattice Bases

	1	10	10
11	x_1	x_2	x_3
10	x_4	x_5	x_6

Colour identifies moves in two different lattice bases.

Choice of basis affects rate of mixing of sampler.



Identifying Geometrically Advantageous Lattice Bases

Let $\boldsymbol{x} \in \mathcal{F}_{A,\boldsymbol{y}}$ be current state of sampler.

Consider sampling in direction $\boldsymbol{u} \in \mathcal{B}_L$.

For feasible $\boldsymbol{x}^\dagger = \boldsymbol{x} + b\boldsymbol{u}$, require $\boldsymbol{x} + b\boldsymbol{u} \geq \boldsymbol{0}$.

Extremes of sampling ray within $\mathcal{F}_{A,\boldsymbol{y}}$ are:

$b_{\min}(\boldsymbol{x}) = -\lfloor \min_{i:u_i>0}\{x_i/|u_i|\} \rfloor, \quad b_{\max}(\boldsymbol{x}) = \lfloor \min_{i:u_i<0}\{x_i/|u_i|\} \rfloor$

Best mixing when $b_{\max}(\boldsymbol{x}) - b_{\min}(\boldsymbol{x})$ is relatively large.

To optimize, choose partition so entries of $\boldsymbol{x}_1$ are relatively large.

Corresponding to maximizing slack in linear inequality $A_2\boldsymbol{x}_2 \leq \boldsymbol{y}$.

Sampling Partitions

What to assign high probability to partitions π with large $\boldsymbol{x}_1$.

Problem: sampling distribution of π should not depend on $\boldsymbol{x}$.

Resolution: use proxy for typical size of entries of $\boldsymbol{x}$.

Example: use unconditional mean $\boldsymbol{\mu} = \mathrm{E}[\boldsymbol{x}|\boldsymbol{\theta}]$.

Let $\phi \sim \mathrm{N}(\boldsymbol{\mu}, \alpha \mathrm{diag}(\boldsymbol{\mu}))$ be vector of fitnesses for columns of A.

Select fittest columns for A_1, subject to invertibility...

... but chance of selecting any column ordering ensures connectivity requirements for walk.

Tuning parameter α determines probability of visiting 'sub-optimal' lattice bases.

- $\alpha = 0$ only uses 'best basis' (irreducibility not assured)
- $\alpha = \infty$ ignores polytope geometry entirely

Sampling Partitions

Input

Current state $\boldsymbol{x}$

Current partition π and corresponding basis vectors U

begin

Draw $\boldsymbol{\phi} \sim \mathrm{N}(\boldsymbol{\mu}, \alpha \mathrm{diag}(\boldsymbol{\mu}))$

Sample $i^{\dagger}$ from discrete uniform distribution on K_1

Sample $j^{\dagger}$ from discrete uniform distribution on $\{j \in K_2 : c_{i^{\dagger} j} \neq 0\}$

if $\phi_{j^{\dagger}} \geq \phi_{i^{\dagger}}$ **then**

Update U

Update π by swapping $i^{\dagger}$ and $j^{\dagger}$ between K_1 and K_2

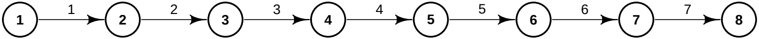
return π, U

Network Tomography Application

Section of A6, Leicester

Network Tomography Application

Nework model



Look at travel in one direction.

Paths connect each node with any subsequent node.

$n = 7$ links and $r = 28$ paths.

$\boldsymbol{y} = (1087, 1008, 1068, 1204, 1158, 1151, 1143)^{\mathsf{T}}$.

$\boldsymbol{x} \sim \mathrm{Pois}(\boldsymbol{\lambda})$ with $\boldsymbol{\lambda}^{\mathsf{T}} =$ (83.0, 25.0, 19.0, 89.0, 10.0, 9.0, 825.0, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 5.0, 1.0, 2.0, 74.0, 0.5, 36.0, 2.0, 105.0, 10.0, 0.1, 69.0, 5.0, 38.0, 15.0).

Chain initialized by solving integer programming problem.

Network Tomography Application

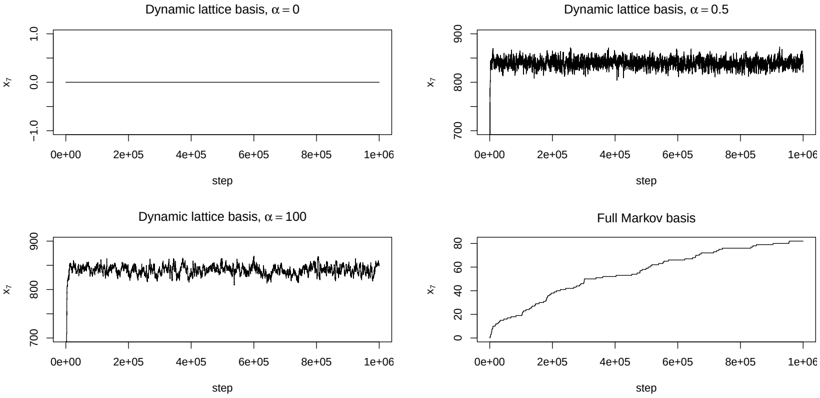
Configuration matrix

$$
A = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1
\end{bmatrix}
$$

Use default Markov basis found by `4ti2`.

Network Tomography Application

Sampler trace plots



Non-Unimodular Configuration Matrices

When A is not-unimodular, the union of lattice bases will still often be a Markov basis.

- In that case our dynamic fibre sampler can be applied directly.

Currently impossible to check whether that result holds in sizeable applications.

Can fix the theoretical hole by introducing occasional moves based on integer-weighted combinations of lattice basis vectors.

Sampler performance remains good in test cases.

Research Challenges

Theoretical analysis of mixing properties.

Adaptive choice of tuning parameter.

Algebraic identification of dynamic Markov bases in non-unimodular cases.

Acknowledgements

Collaborators

Andrew Robinson (U. Melbourne)
Bruce van Brunt (Massey U.)
Chris Tuffley (Massey U.)
Dominic Searles (U. Otago)
Linus Fromm (Current PhD, U. Otago)
Masoud Karimi (Islamic Azad University, Bojnourd Branch)
Mike McVeagh (PhD, Massey U.)
Timothy Bilton (AgResearch)

Funding

MARSDEN FUND
TE PŪTEA RANGAHAU A MARSDEN

ROYAL SOCIETY
TE APĀRANGI