Randomized Algorithms over Dense Trees of Trees

Raazesh Sainudiin

joint work with: Jennifer Harlow, Dominic Lee and Gloria Teng

Department of Mathematics and Statistics, University of Canterbury,
Christchurch, New Zealand

Plenary Talk at ICMSIC 2010,

International congress of Mathematicians Satellite International Conference on Probability and Statistics,
Sambalpur University, Jyoti Vihar, Orissa, India 768019
September 1-3, 2010

Massive Metric Data Streams – Introduction

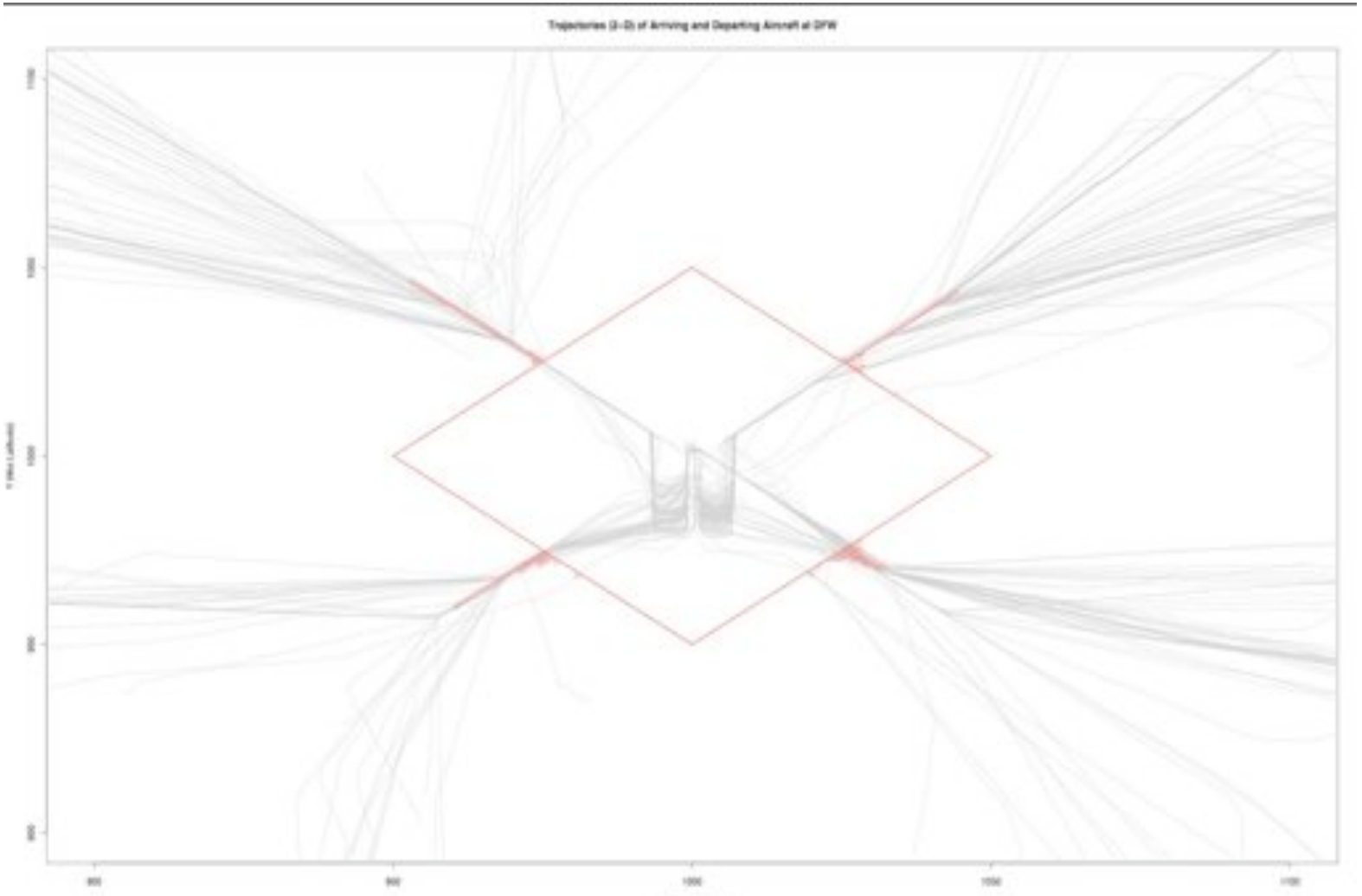
- A massive metric data stream is:

$$\ldots, X_{-3}, X_{-2}, X_{-1}, X_0, \mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}, \mathbf{X_n}, X_{n+1}, \ldots \sim F, \quad X_i \in \mathbb{R}^d.$$

- Large Dimension: $1 \leq d \leq 1000$
- Huge Observations: $10^6 \leq n \leq 10^{10}$
- Need an **efficient** and **sufficient** multi-dimensional metric data-structure for non-parametric inference that is capable of:
  1. L_1-consistent density estimation – adaptive histograms
  2. Extend Arithmetic over a dense class of Lipschitz $\mathbb{M}$-valued maps: $\{g : \mathbb{R}^d \to \mathbb{M}\}$ – functional estimation when $\mathbb{M} = \mathbb{R}$

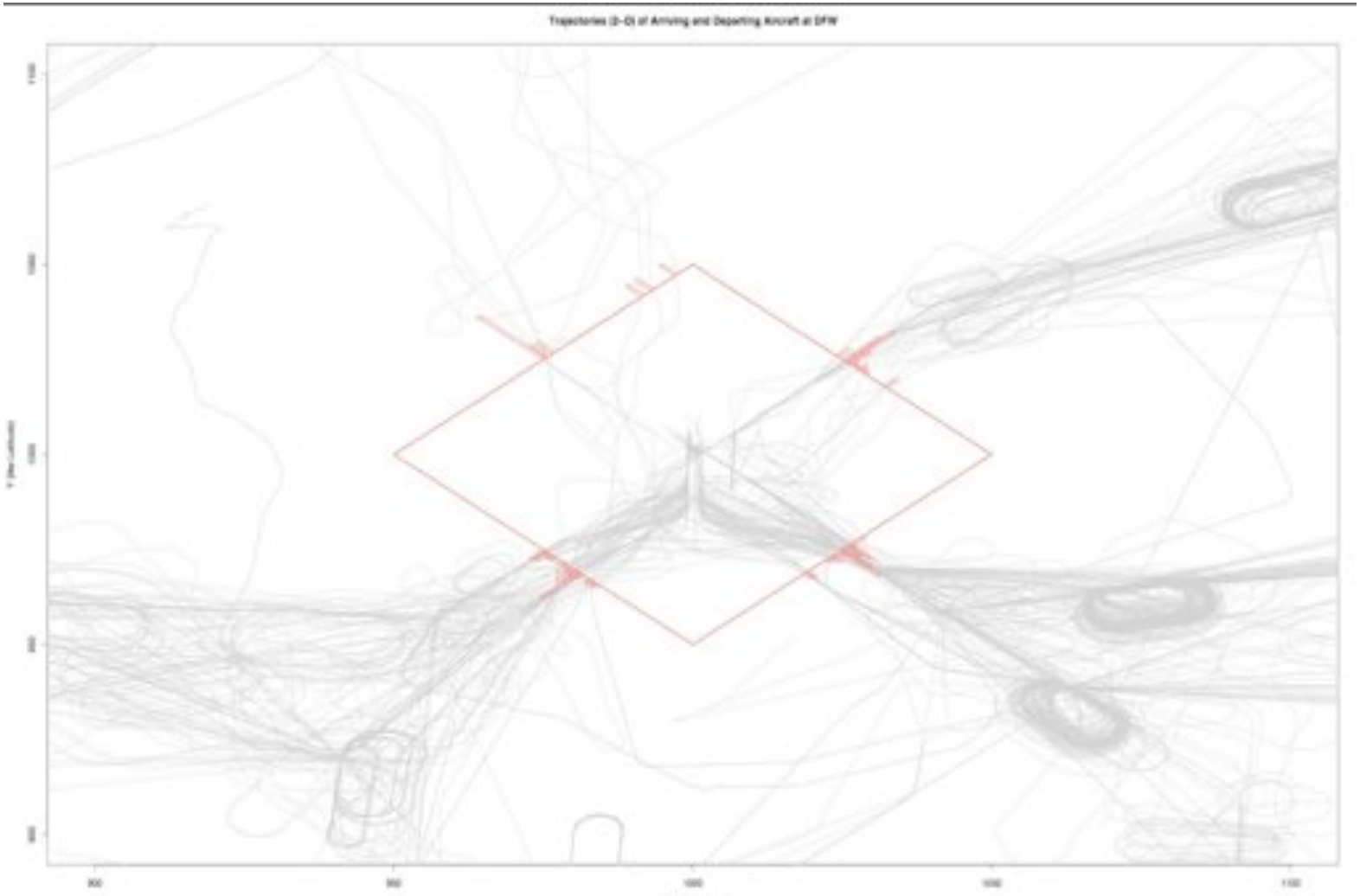
Massive Metric Data Streams – Air Traffic Example

On a Sunny Day



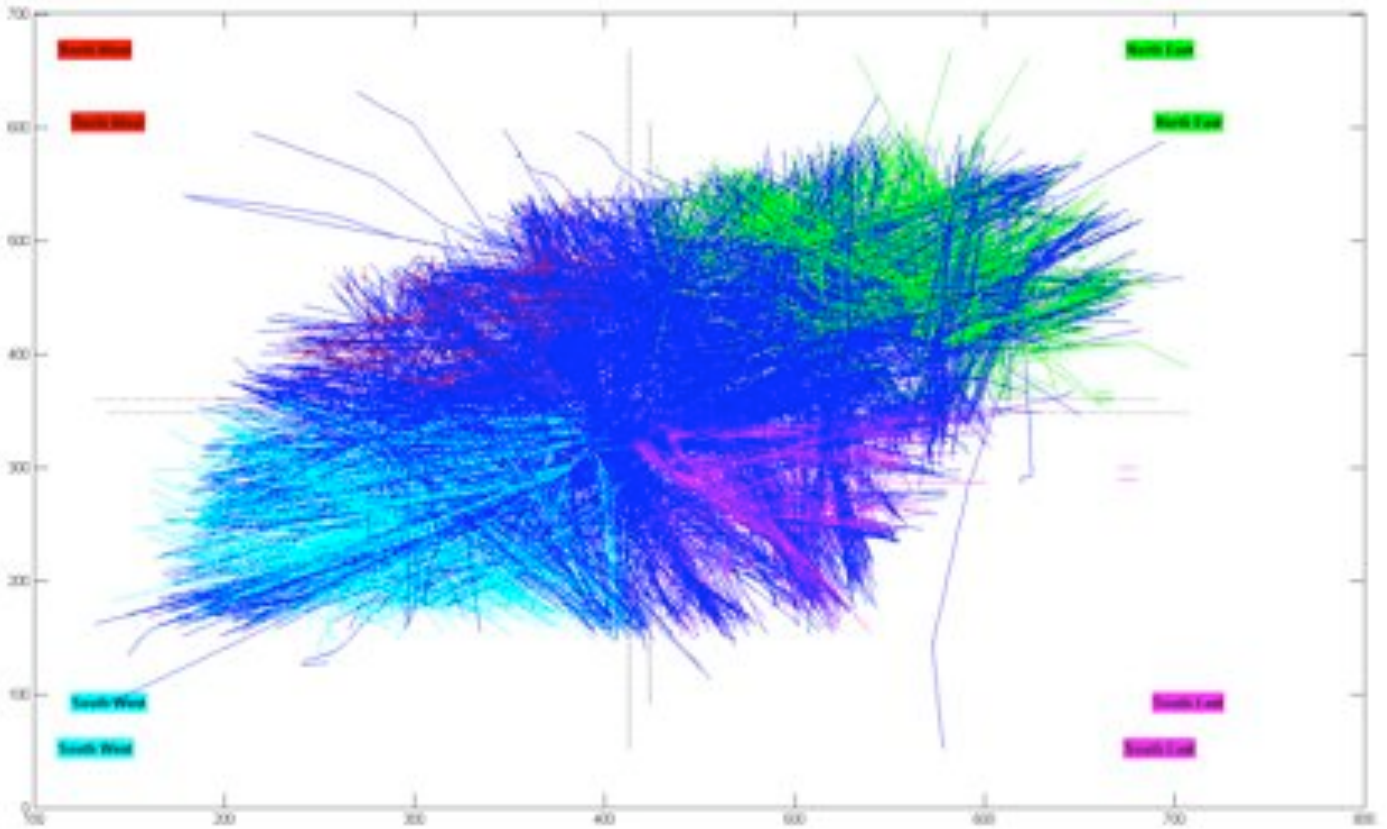
Massive Metric Data Streams – Air Traffic Example

On a Rainy Day



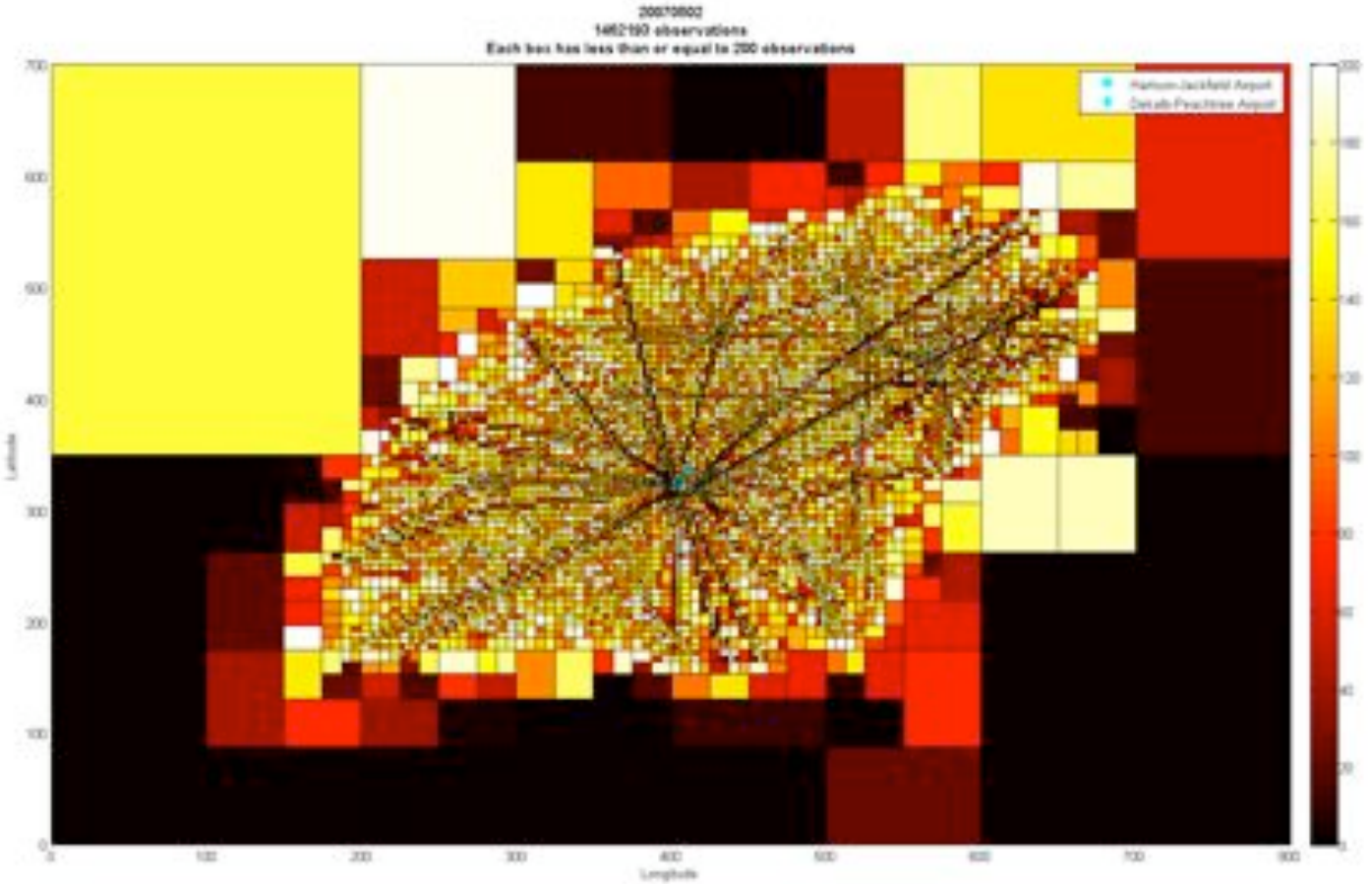
Massive Metric Data Streams – Air Traffic Example

We want to make sense of trajectories like these



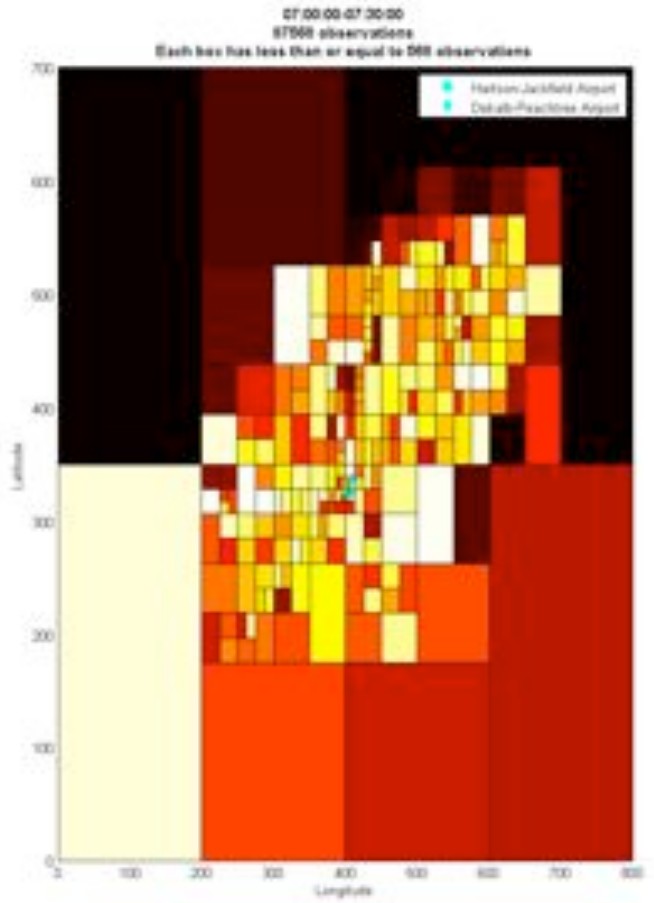
Massive Metric Data Streams – Air Traffic Example

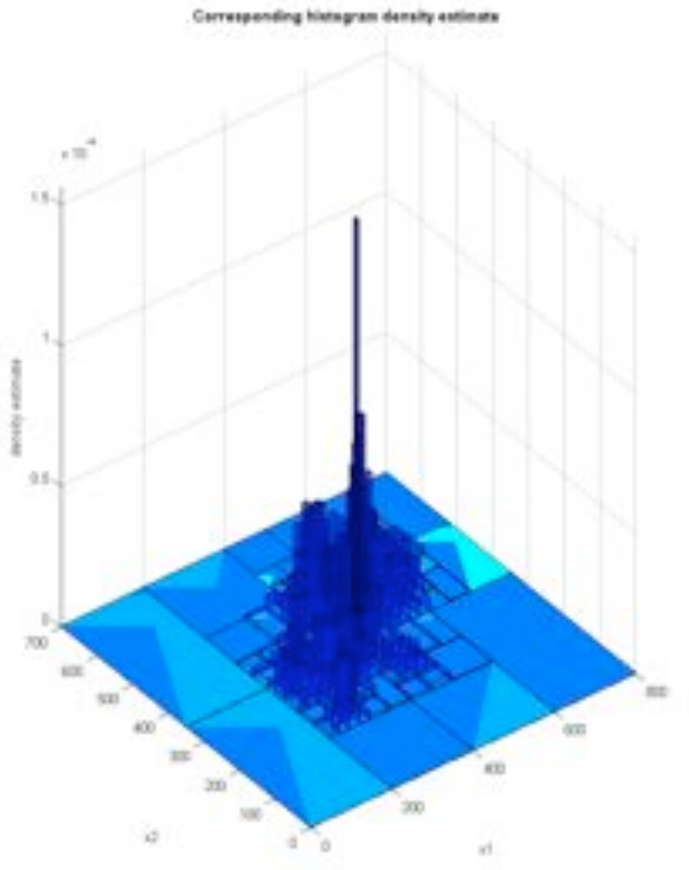
using a picture like this



Massive Metric Data Streams – Air Traffic Example

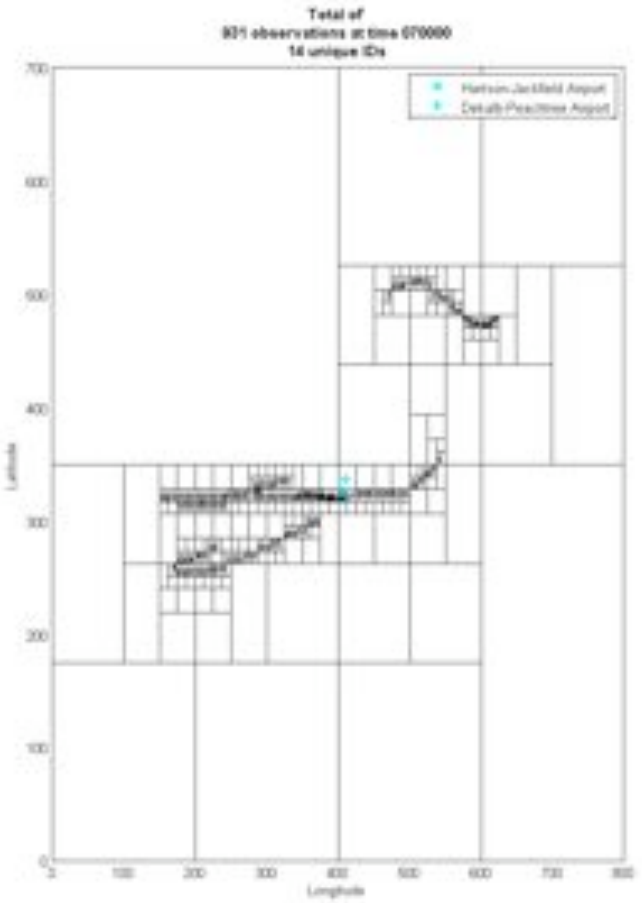
A Histogram Estimate of Air-traffic between 0700 – 0730 hours

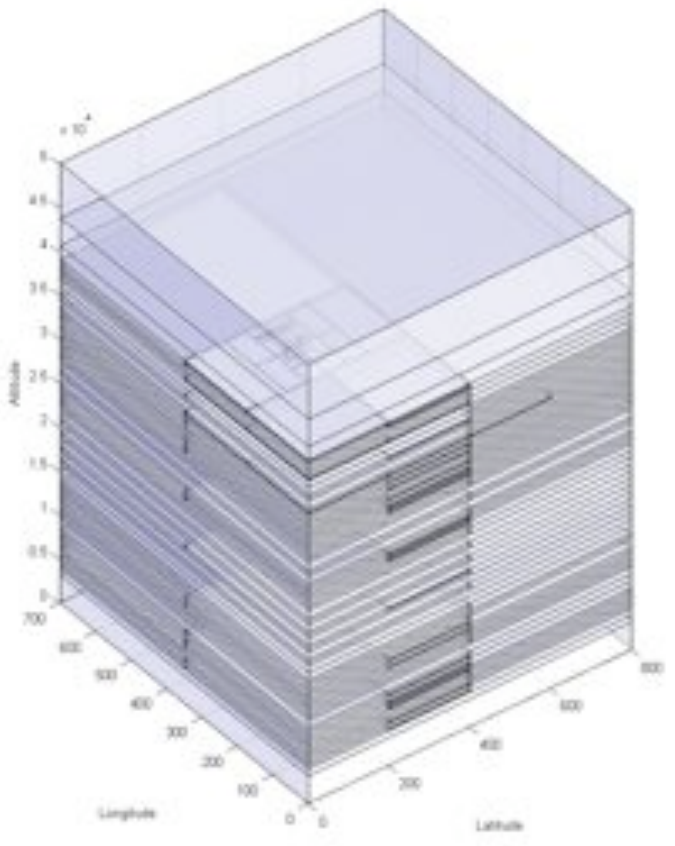




Massive Metric Data Streams – Air Traffic Example

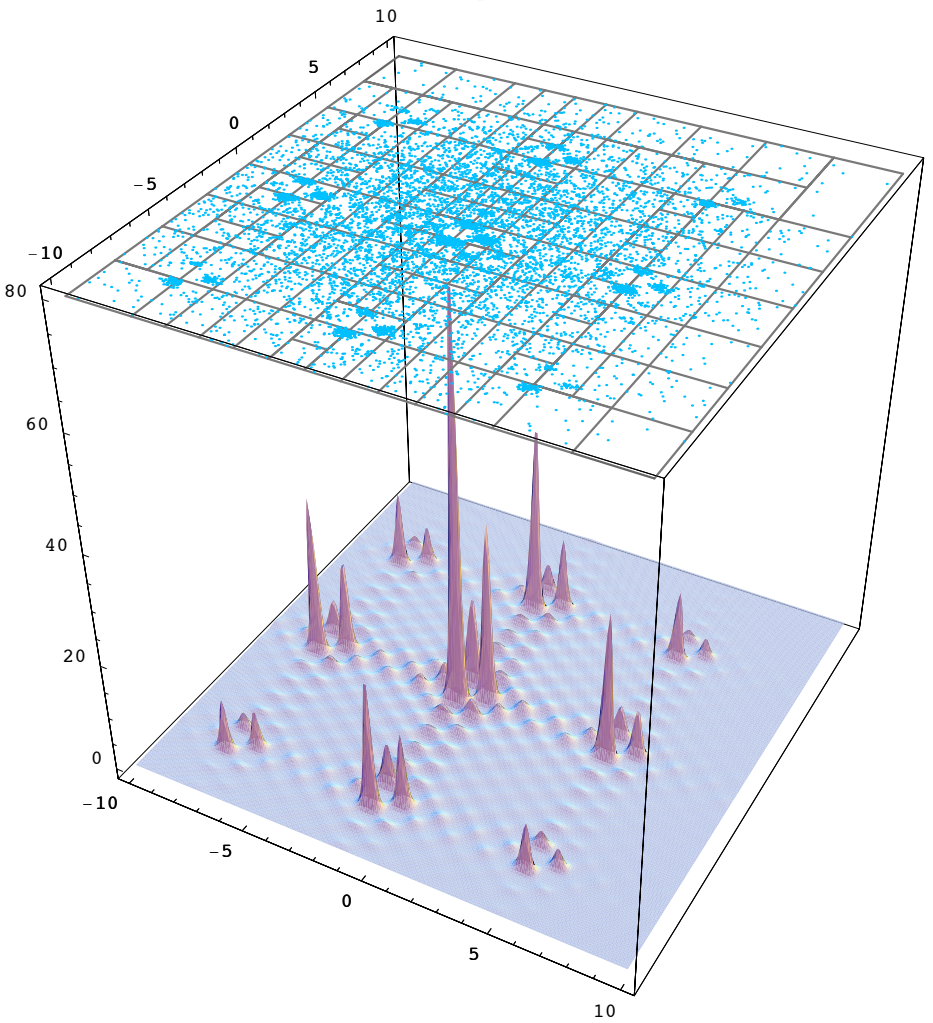
Add the pavings of 14 flight trajectories like this





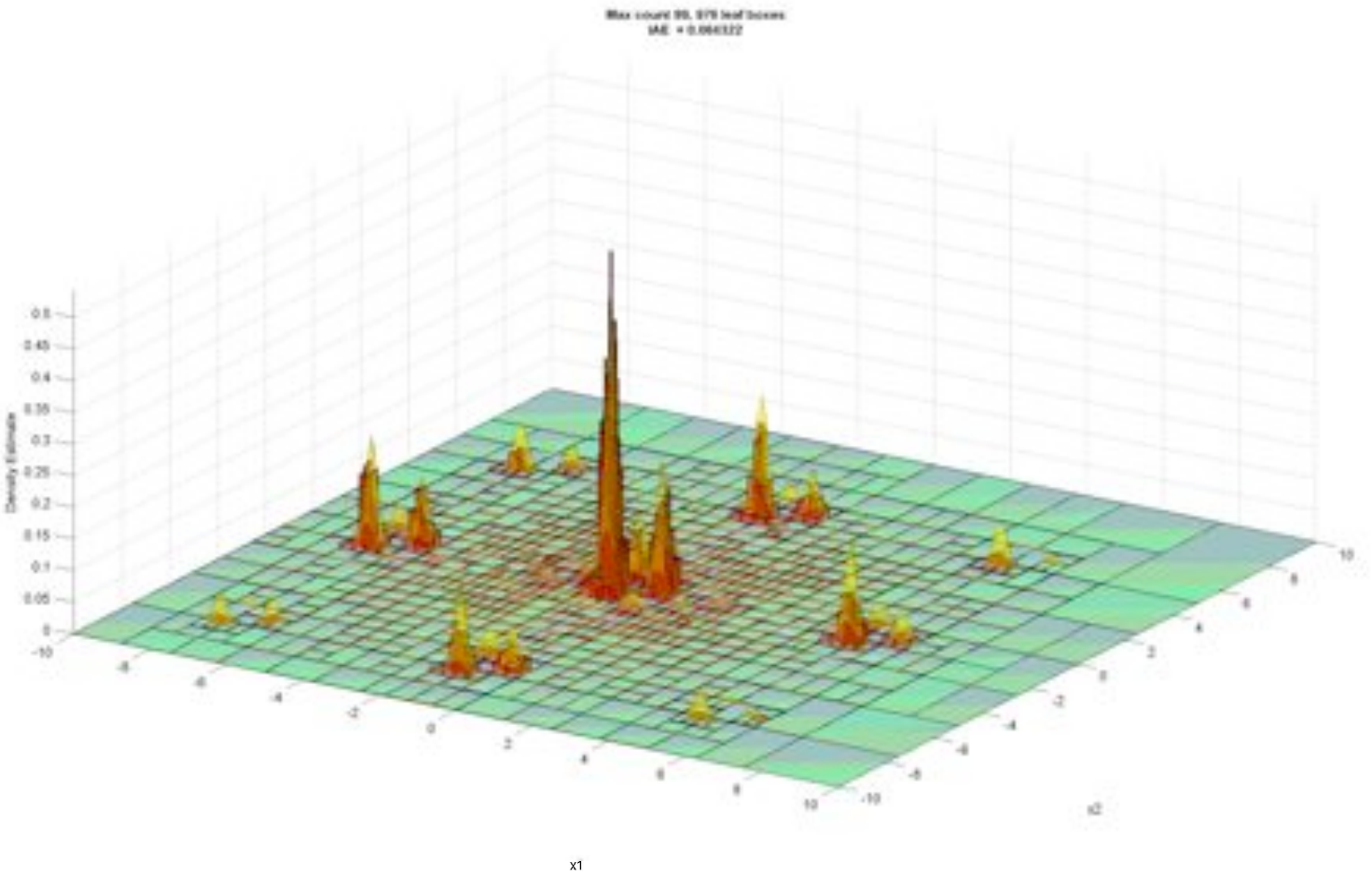
Massive Metric Data Streams – Synthetic Examples

Take millions of realizations of a possibly ‘challenging’ density



Massive Metric Data Streams – Synthetic Examples

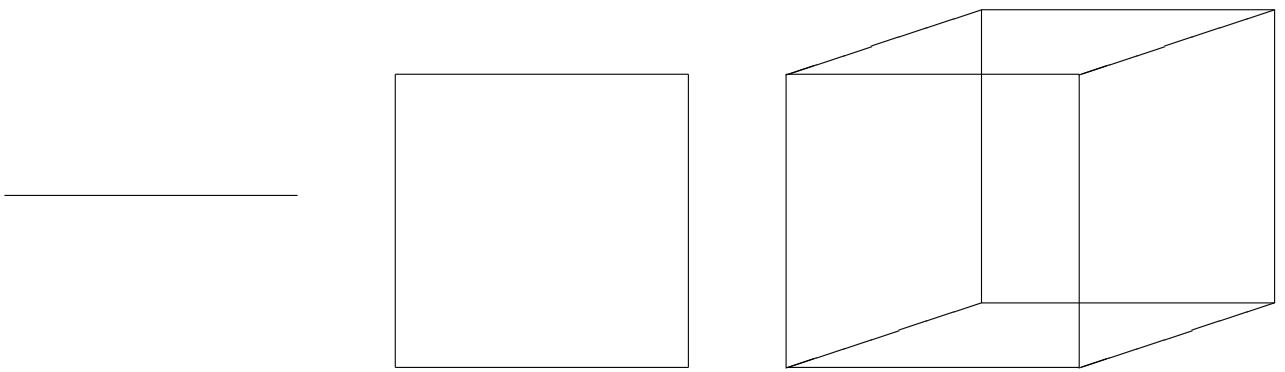
and produce a consistent estimate of the density



Intervals and Boxes in $\mathbb{R}^d$

Intervals and *Boxes* as interval vectors:

$$\boldsymbol{x} = [\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2] \times \ldots \times [\underline{x}_d, \overline{x}_d] \,, \; \underline{x}_i \leq \overline{x}_i \;\; .$$



1-dim. 2-dim. 3-dim.

Figure: Boxes in 1D, 2D, and 3D.

Binary Tree Representation

These boxes can also be represented by ordered binary trees. An operation of bisection on a box is equivalent to performing the operation on its corresponding node in the tree, i.e.:

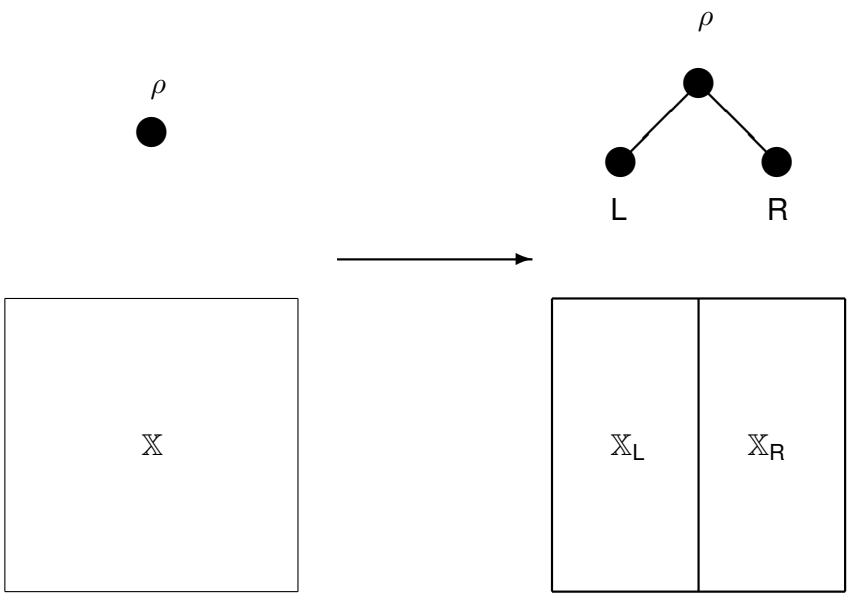
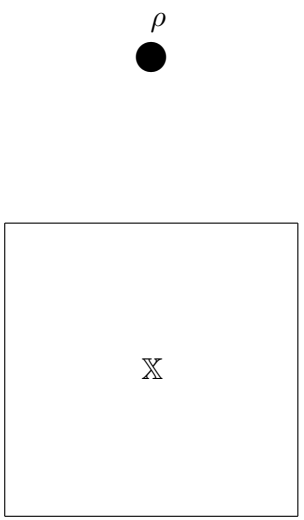


Figure: Bisecting a box or its equivalent node.

Regular Sub-pavings (RSPs)

- A sequence of bisections of boxes;
- Start from the root box;
- Along the first widest dimension.

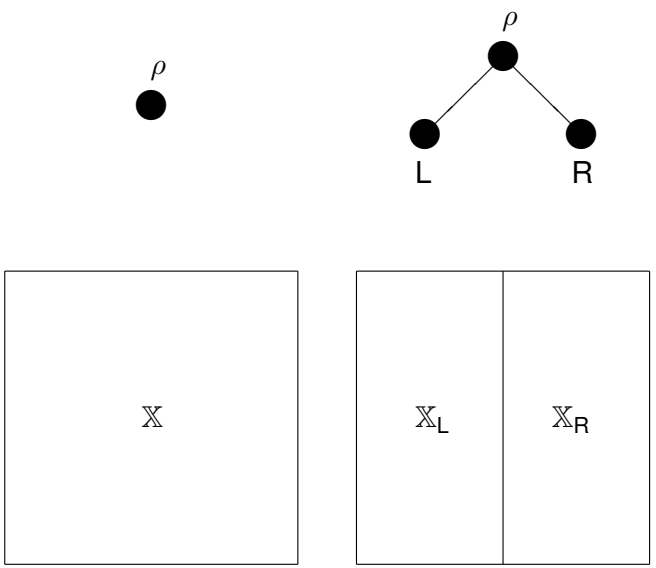
A sequence of bisections on root box $\mathbb{X}$ to get a 4-leaved RSP.



Regular Sub-pavings (RSPs)

- A sequence of bisections of boxes;
- Start from the root box;
- Along the first widest dimension.

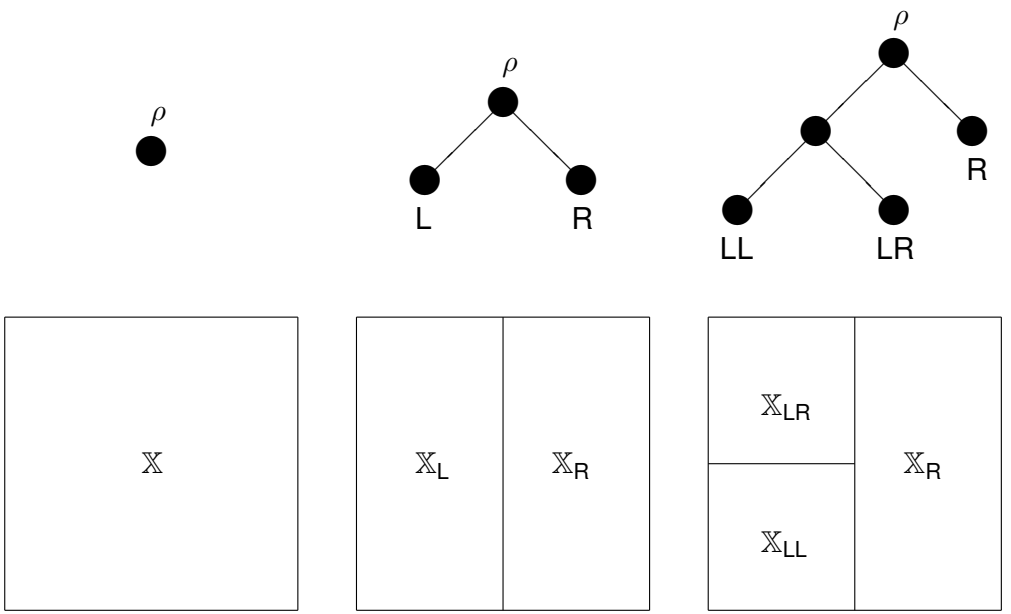
A sequence of bisections on root box $\mathbb{X}$ to get a 4-leaved RSP.



Regular Sub-pavings (RSPs)

- A sequence of bisections of boxes;
- Start from the root box;
- Along the first widest dimension.

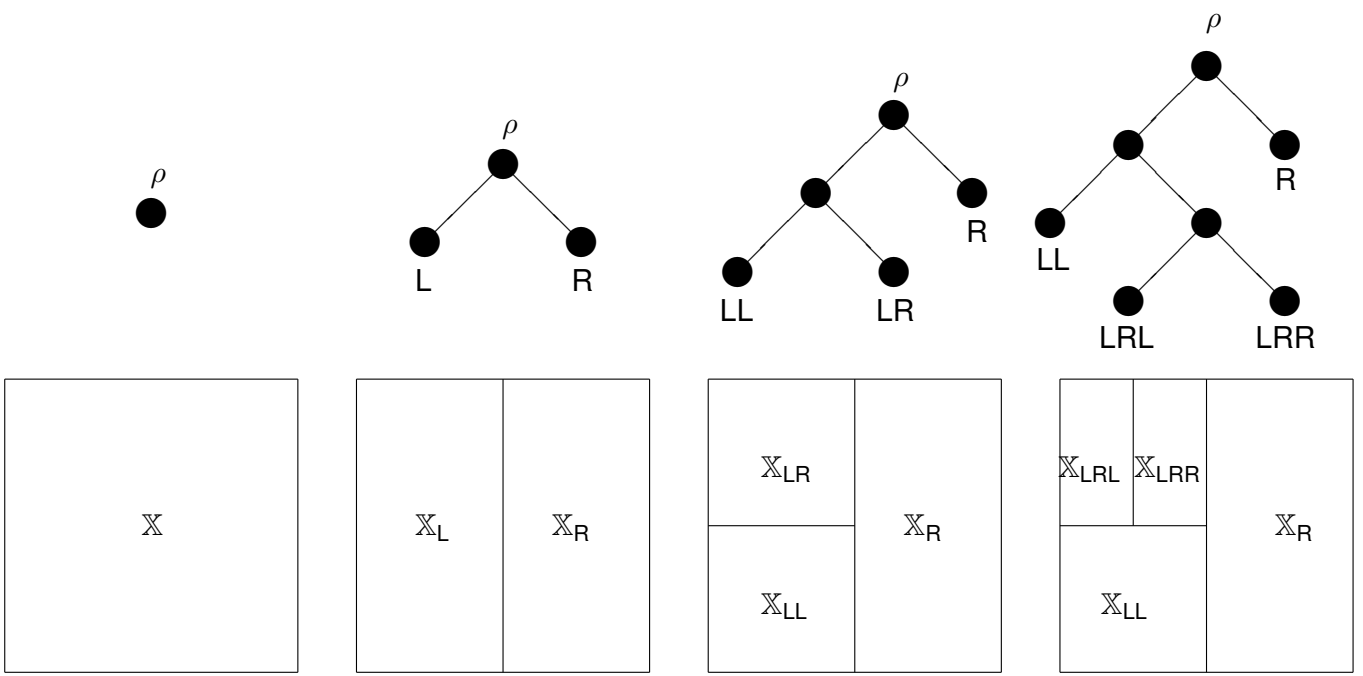
A sequence of bisections on root box $\mathbb{X}$ to get a 4-leaved RSP.



Regular Sub-pavings (RSPs)

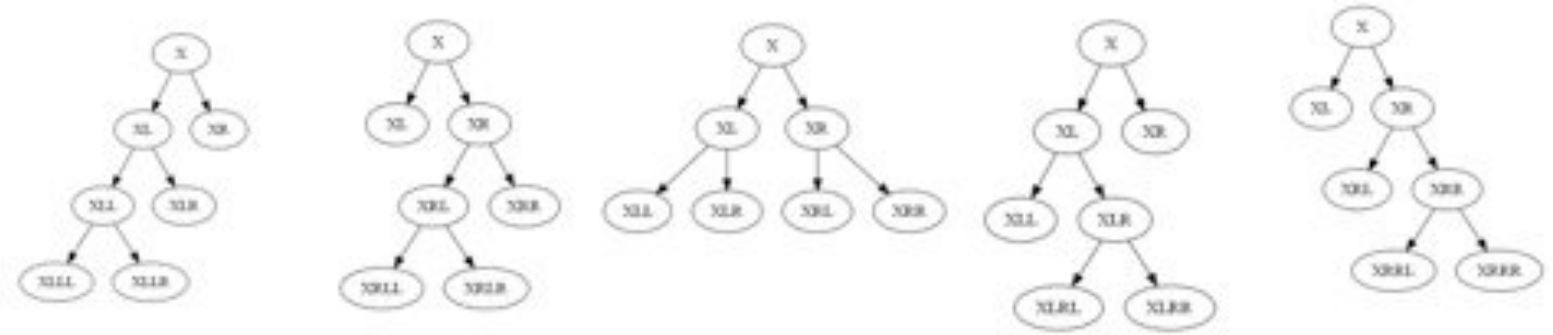
- A sequence of bisections of boxes;
- Start from the root box;
- Along the first widest dimension.

A sequence of bisections on root box $\mathbb{X}$ to get a 4-leaved RSP.

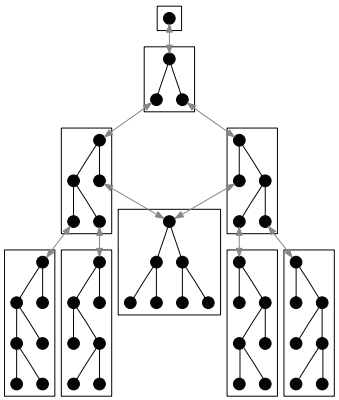


State Space of Regular Sub-pavings

Leaf-depth encoded RSPs

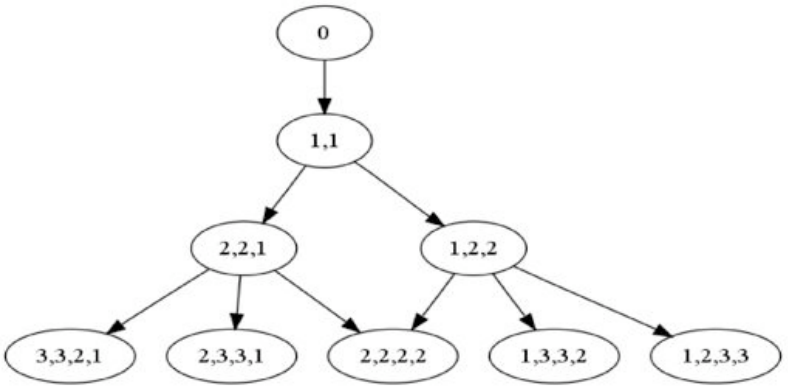


(3, 3, 2, 1) (1, 3, 3, 2) (2, 2, 2, 2) (2, 3, 3, 1) (1, 2, 3, 3)



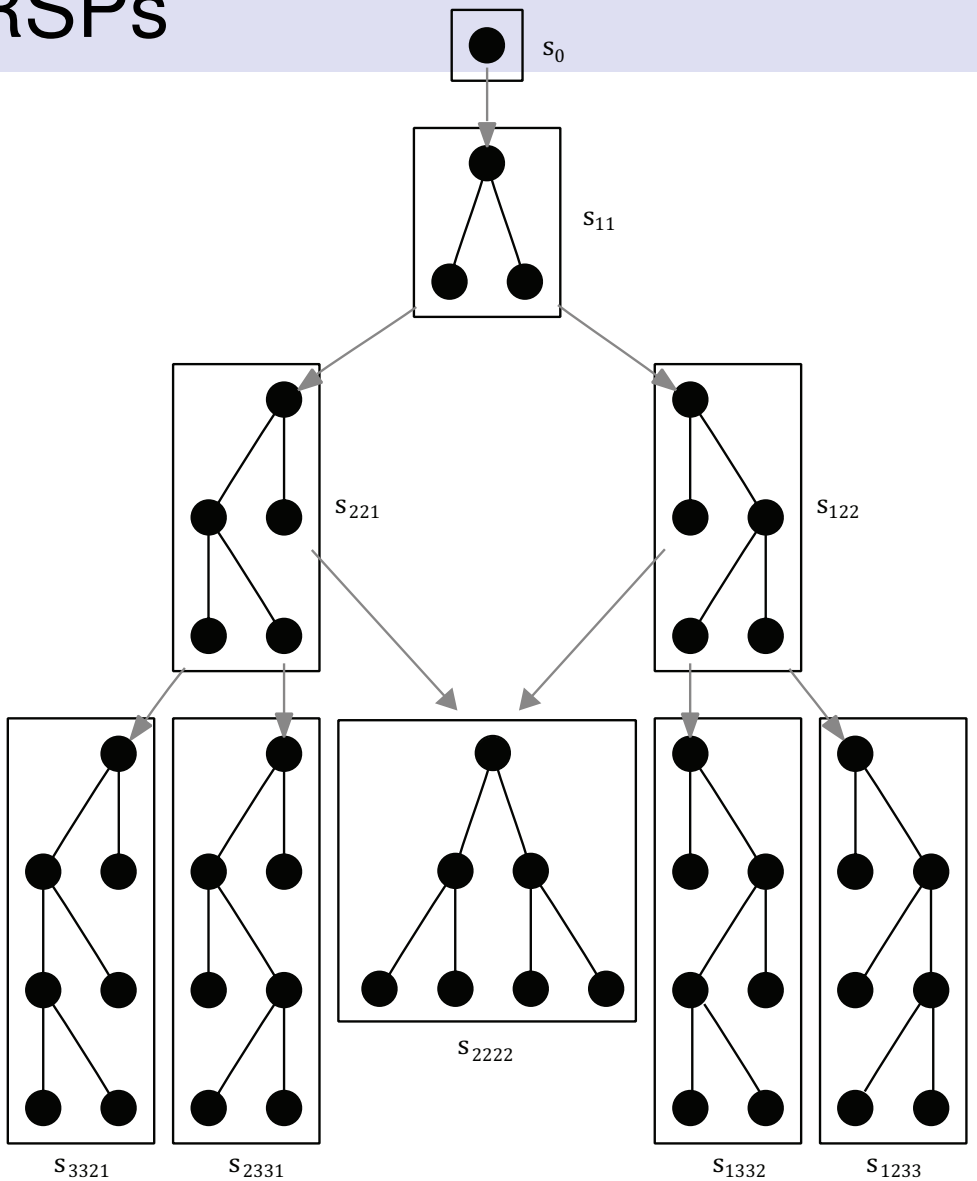
C_k RSPs wit k splits

$$
\begin{aligned}
C_0 &= 1 \\
C_1 &= 1 \\
C_2 &= 2 \\
C_3 &= 5 \\
C_4 &= 14 \\
C_5 &= 42 \\
C_k &= \frac{(2k)!}{(k+1)!k!}
\end{aligned}
$$



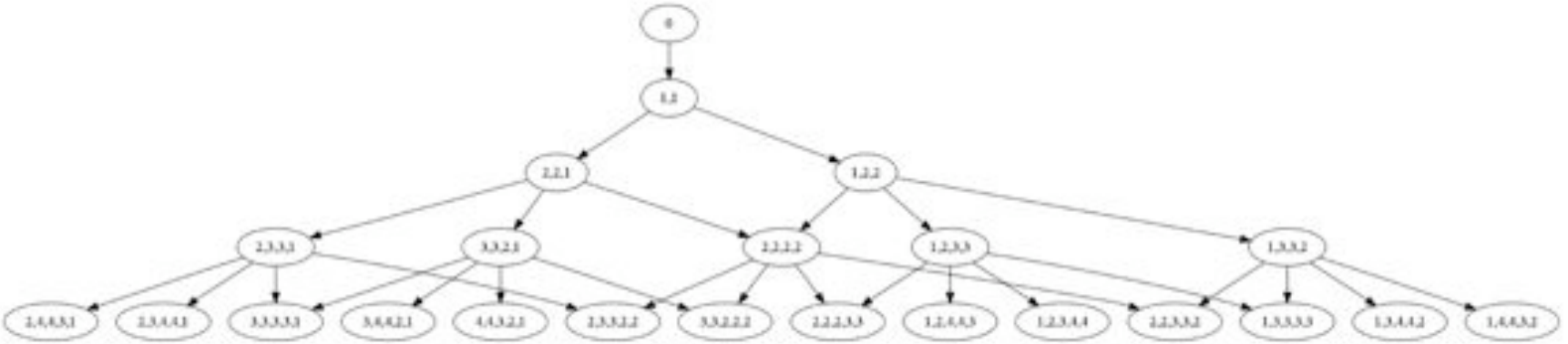
The Space of All Possible RSPs

- Let $\mathbb{S}_i$ be the set of all RSPs of $\mathbb{X}$ made of i splits and
- Let $\mathbb{S}_{i:j}$ be the set of RSPs with k splits where $k \in \{i, i+1, \ldots, j\}$
- The space of all RSPs is $\mathbb{S}_{0:\infty} := \lim_{i \to \infty} \mathbb{S}_{0:i}$
- RSPs are closed under pair-wise union, intersection and overlay (non-minimal union) operations
- can get as m_∞-close as desired to any subset of $\mathbb{X}$



State Transition Diagram of Regular Sub-pavings

State Transition Diagram of RSPs with 0, 1, 2, 3 and 4 splits.

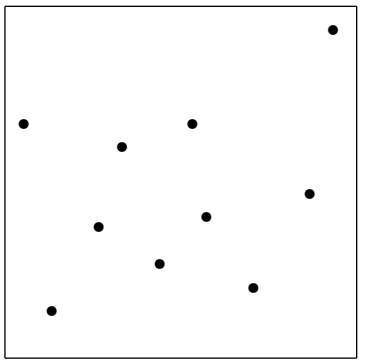


1. The above state space is denoted by $\mathbb{S}_{0:4}$
2. Number of RSPs with k splits is the Catalan number C_k
3. There is more than one way to reach a RSP by k splits
4. Randomized algorithms of interest are often Markov chains on $\mathbb{S}_{0:\infty}$

Statistical Regular Sub-pavings (SRSPs)

- Extended from the RSP;
- Caches recursively computable statistics at each box or node as data falls through;
- These statistics include:
  - the sample count;
  - the sample mean vector;
  - the sample variance-covariance matrix;
  - and the volume of the box.

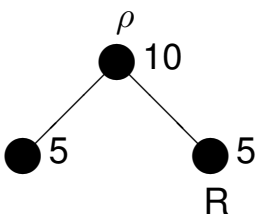
Caching the sample count in each node (or box).

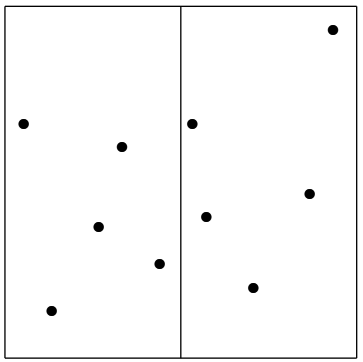


Statistical Regular Sub-pavings (SRSPs)

- Extended from the RSP;
- Caches recursively computable statistics at each box or node as data falls through;
- These statistics include:
  - the sample count;
  - the sample mean vector;
  - the sample variance-covariance matrix;
  - and the volume of the box.

Caching the sample count in each node (or box).

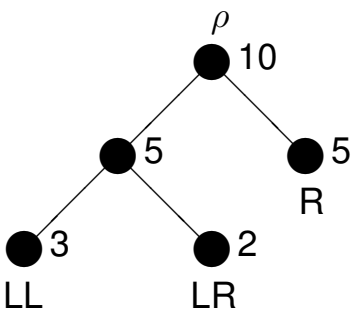


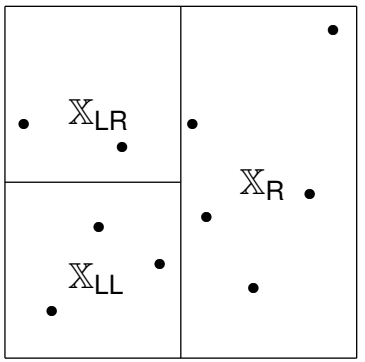


Statistical Regular Sub-pavings (SRSPs)

- Extended from the RSP;
- Caches recursively computable statistics at each box or node as data falls through;
- These statistics include:
  - the sample count;
  - the sample mean vector;
  - the sample variance-covariance matrix;
  - and the volume of the box.

Caching the sample count in each node (or box).





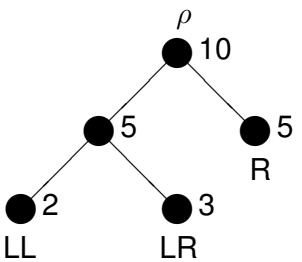
SRSPs as Adaptive Histograms

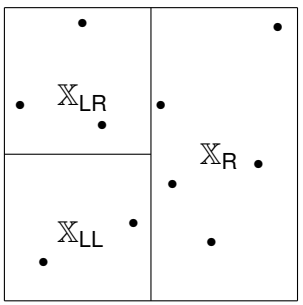
SRSP estimate of f from random vectors $X_1, X_2, \ldots, X_n \overset{iid}{\sim} f$ is

$$f_{n,\dot{s}}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{\mathbb{1}\left(x_i \in \mathbf{x}(x)\right)}{\mathrm{vol}\left(\mathbf{x}(x)\right)} \quad ,$$

$\boldsymbol{x}(x) \in \ell(\dot{s})$ is the leaf box containing x with volume $\mathrm{vol}\left(\boldsymbol{x}(x)\right)$

Figure: A SRSP as a histogram estimate.





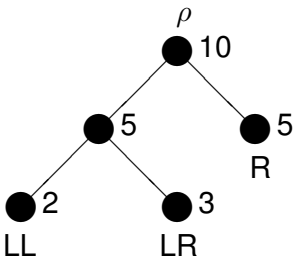
SRSPs as Adaptive Histograms

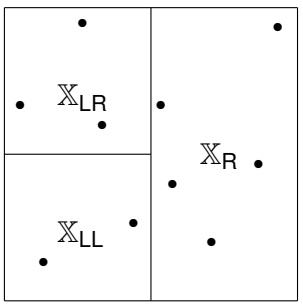
SRSP estimate of f from random vectors $X_1, X_2, \ldots, X_n \overset{iid}{\sim} f$ is

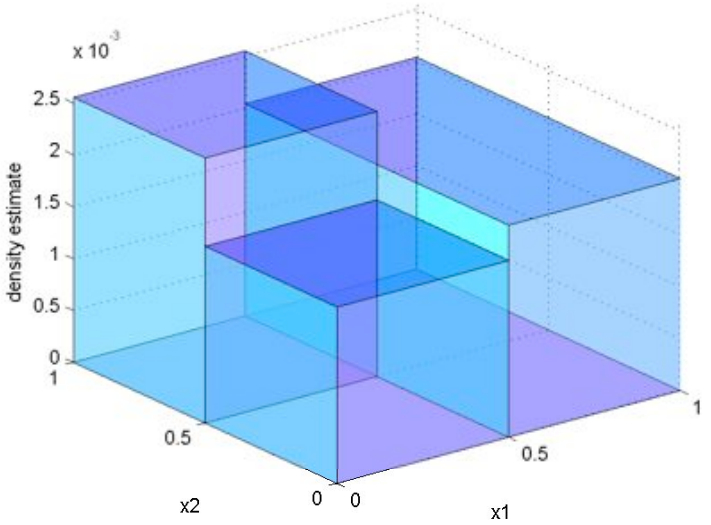
$$f_{n,\dot{s}}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{\mathbb{1}\left(x_i \in \mathbf{x}(x)\right)}{\mathrm{vol}\left(\mathbf{x}(x)\right)} \quad ,$$

$\boldsymbol{x}(x) \in \ell(\dot{s})$ is the leaf box containing x with volume $\mathrm{vol}\left(\boldsymbol{x}(x)\right)$

Figure: A SRSP as a histogram estimate.



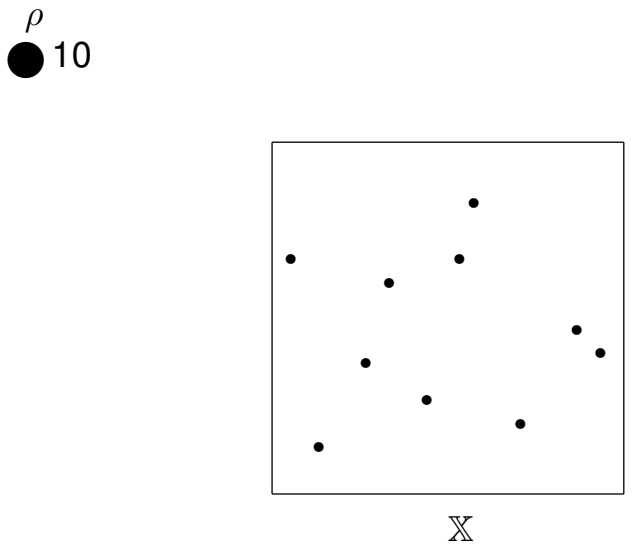




A Prioritized Queue based Algorithm

Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRSP so that the leaf box with **the most number of points** will be chosen for the next bisection.

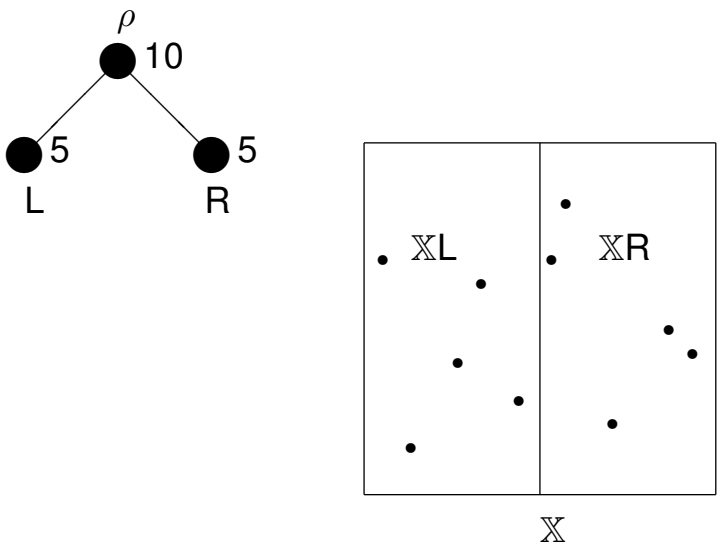


A Prioritized Queue based Algorithm

Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRSP so that the leaf box with **the most number of points** will be chosen for the next bisection.

Split the root box.

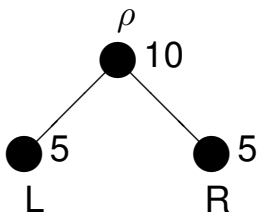


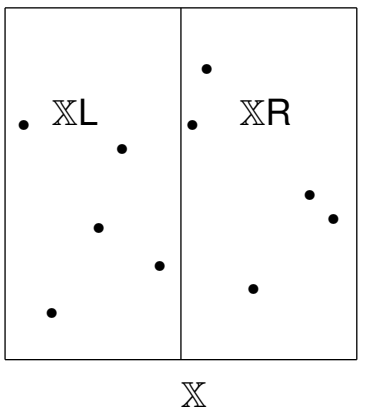
A Prioritized Queue based Algorithm

Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRSP so that the leaf box with **the most number of points** will be chosen for the next bisection.

Two or more boxes with the most number of points?



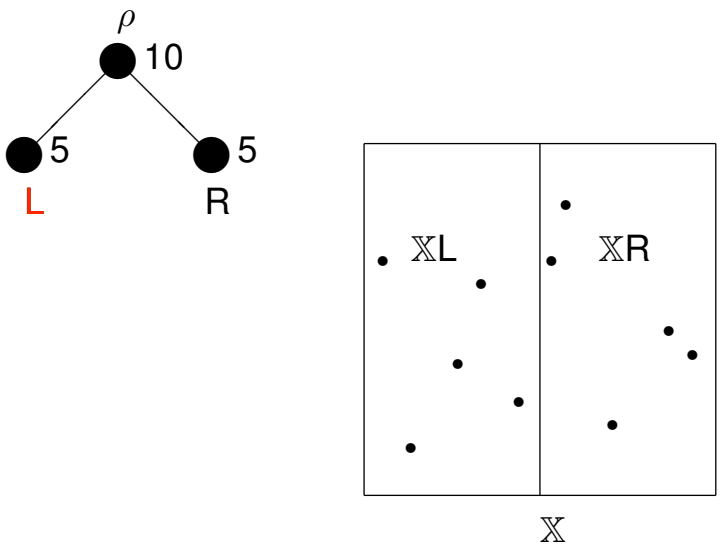


A Prioritized Queue based Algorithm

Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRSP so that the leaf box with **the most number of points** will be chosen for the next bisection.

Break such ties by randmising the next bisection.

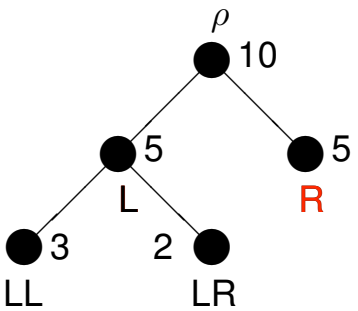


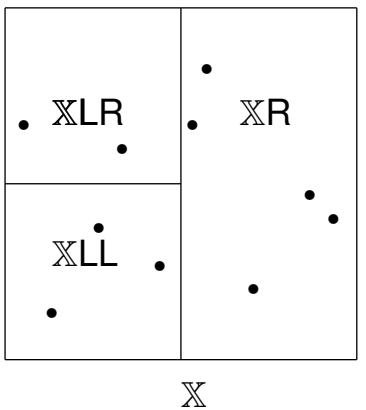
A Prioritized Queue based Algorithm

Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRSP so that the leaf box with **the most number of points** will be chosen for the next bisection.

Bisect until each box has $\leq k_n$ points (let $k_n = 3$ here).



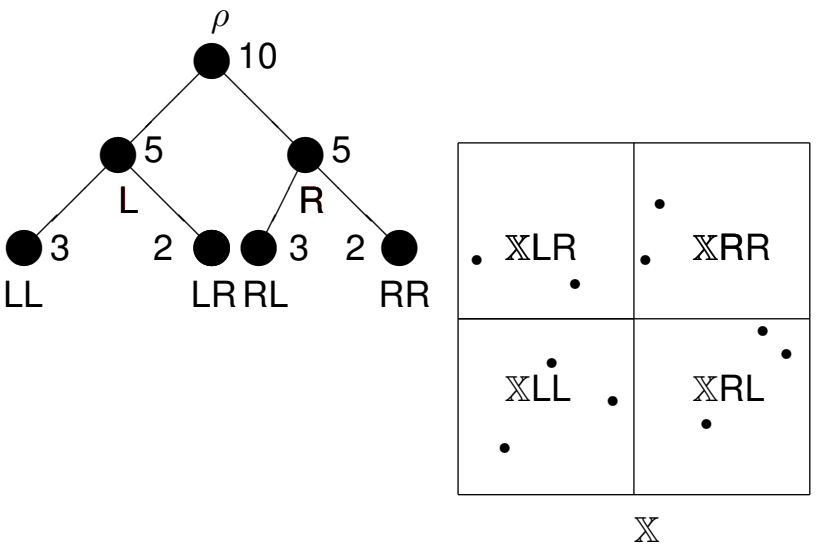


A Prioritized Queue based Algorithm

Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRSP so that the leaf box with **the most number of points** will be chosen for the next bisection.

Final state



The `SplitMostCounts` Algorithm

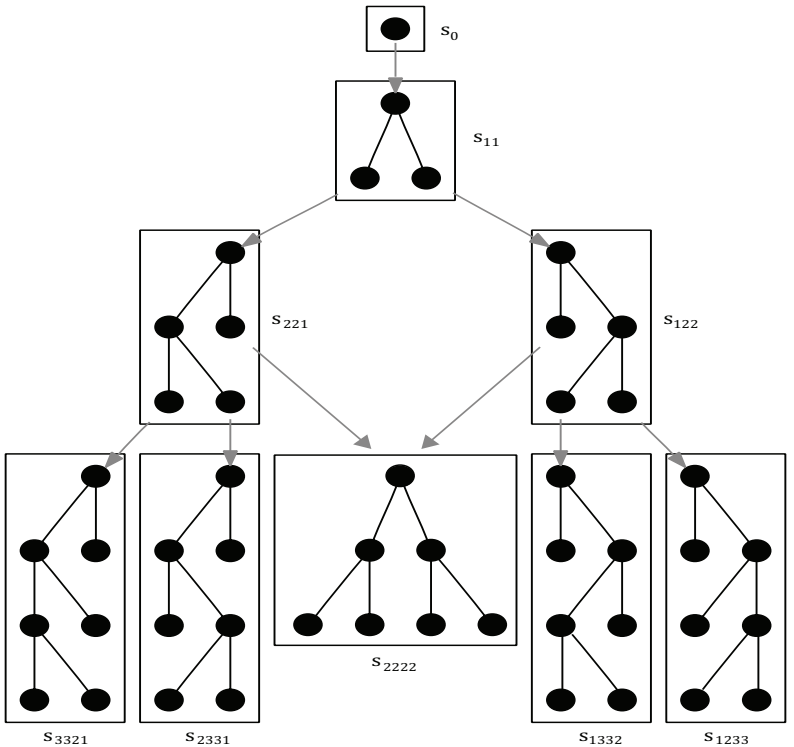
Input: (i) data: $x_1, \ldots, x_n \subseteq \mathbb{R}^d$; (ii) root box: $\mathbb{X}$ // `optional`;
(iii) padding to handle pulsed data: $\psi \geq 0$ // `optional`;
(iv) S.E.B. max: $\overline{k}_n$; (v) maximum partition size: $\overline{m}_n$.

Output: histogram estimate $f_{n,s}$.

initialize $i \leftarrow 1$; $s \leftarrow \mathbb{X} + \psi$;
repeat until
  $\#\boldsymbol{x} \leq \overline{k}_n$ for each $\boldsymbol{x} \in \ell(s)$ and $i \leq \overline{m}_n$ // $\ell(s) = \{$`leaf boxes`$\}$
  $\boldsymbol{x} \leftarrow \text{Uniform}(\hat{\ell}(s))$ // `randomized PQ on leaf boxes`
  $s \leftarrow$ `bisect`$(s, \boldsymbol{x})$ // `bisect leaf box` $\boldsymbol{x}$ `of` s
  recursively update counts in s;
  $i \leftarrow i + 1$;
return $f_{n,s}$.

Transition Diagram of Randomized PQ Markov chain

Let $\mathbb{S}_i$ be the set of all RSPs of $\mathbb{X}$ made of i splits and for $i, j \in \mathbb{N}$ with $i \leq j$, let $\mathbb{S}_{i:j}$ be the set of RSPs with k splits, $i \leq k \leq j$.



All possible RSP partitions in $\mathbb{S}_{0:4}$.

Proposition: L_1-Consistency of Histogram Estimates from `SplitMostCounts`

Let $X_1, X_2, \ldots$ be independent and identical random vectors in $\mathbb{R}^d$ whose common distribution μ has a non-atomic density f, i.e., $f \ll \lambda^d$. Let $\{S_n(i)\}_{i=0}^{\dot{l}}$ on $\mathbb{S}_{0:\infty}$ be the Markov chain formed using `SplitMostCounts` with terminal state $\dot{s}$ and histogram estimate $f_{n,\dot{s}}$ over the collection of partitions $\mathcal{L}_n$.
As $n \to \infty$, if $\overline{k}_n \to \infty$, $n^{-1}\overline{k}_n \to 0$, $\overline{m}_n \geq n/\overline{k}_n$, and $\overline{m}_n/n \to 0$ then the density estimate $f_{n,\dot{s}}$ is strongly consistent in L_1, i.e.

$$\int |f(x) - f_{n,\dot{s}}(x)|dx \to 0 \text{ with probability 1.}$$

Proof Sketch

We will assume that $\overline{k}_n \to \infty$, $n^{-1}\overline{k}_n \to 0$, $\overline{m}_n \geq n/\overline{k}_n$, and $\overline{m}_n/n \to 0$, as $n \to \infty$, and show that the three conditions:

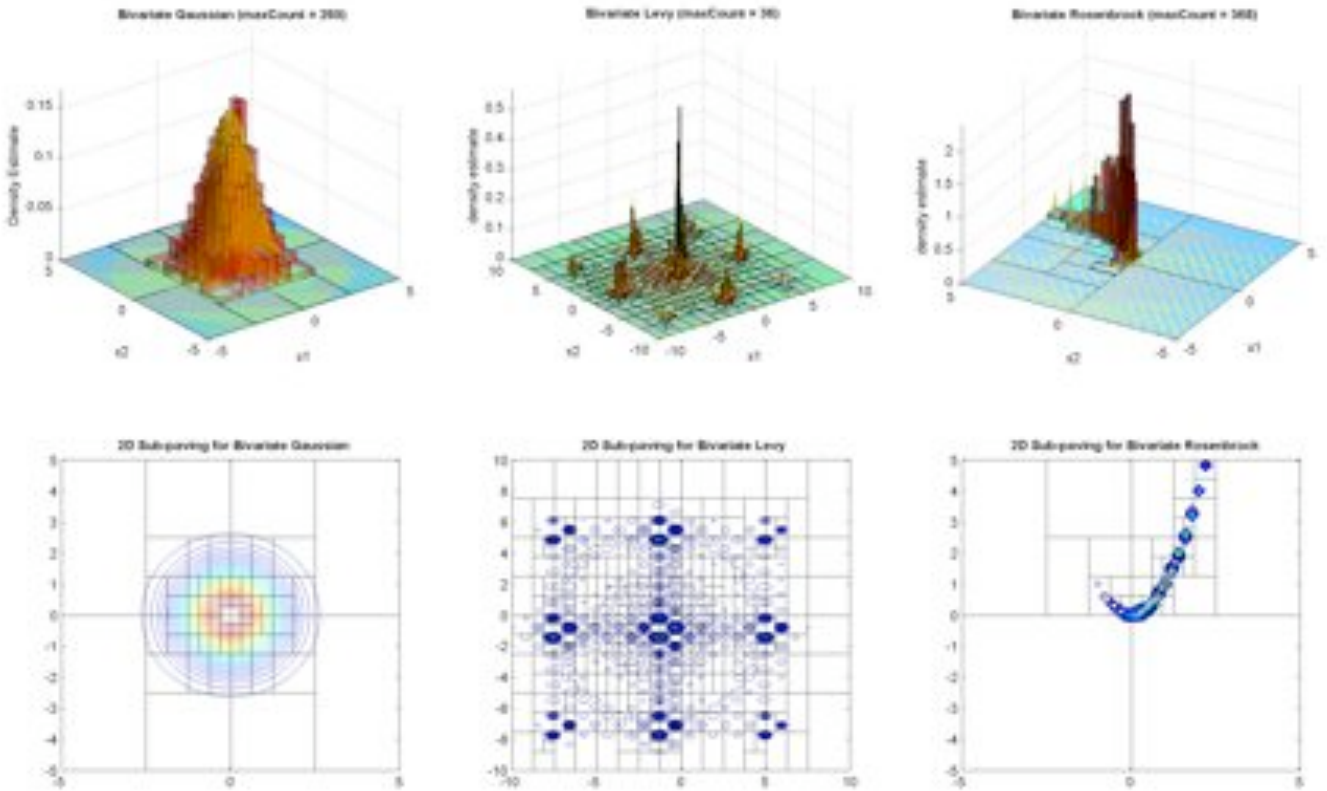
(a) $n^{-1} m(\mathcal{L}_n) \to 0$,
(b) $n^{-1} \log \Delta_n^*(\mathcal{L}_n) \to 0$, and
(c) $\mu(x : \mathrm{diam}(\boldsymbol{x}(x)) > \gamma) \to 0$ with probability 1 for every $\gamma > 0$,

are satisfied. Then by Theorem 1 of Lugosi and Nobel, 1996 our density estimate $f_{n,\dot{s}}$ is strongly consistent in L_1.
These conditions mean:

(a) sub-linear growth of the number of leaf boxes
(b) sub-exponential growth of a combinatorial complexity measure of the growth of the partition
(c) shrinking leaf boxes in the partition

Some Examples

Figure: Histogram density estimates their corresponding sub-pavings for the bivariate Gaussian, Levy and Rosenbrock densities.



Choice of k_n

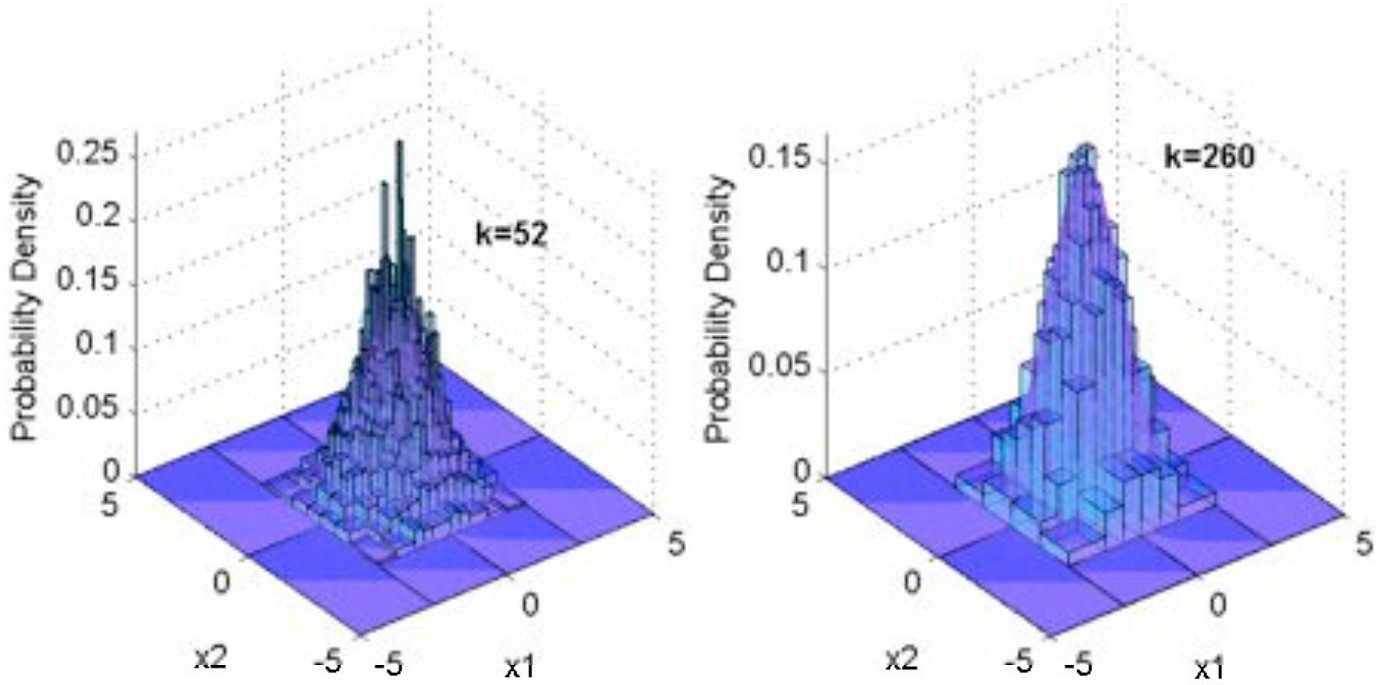
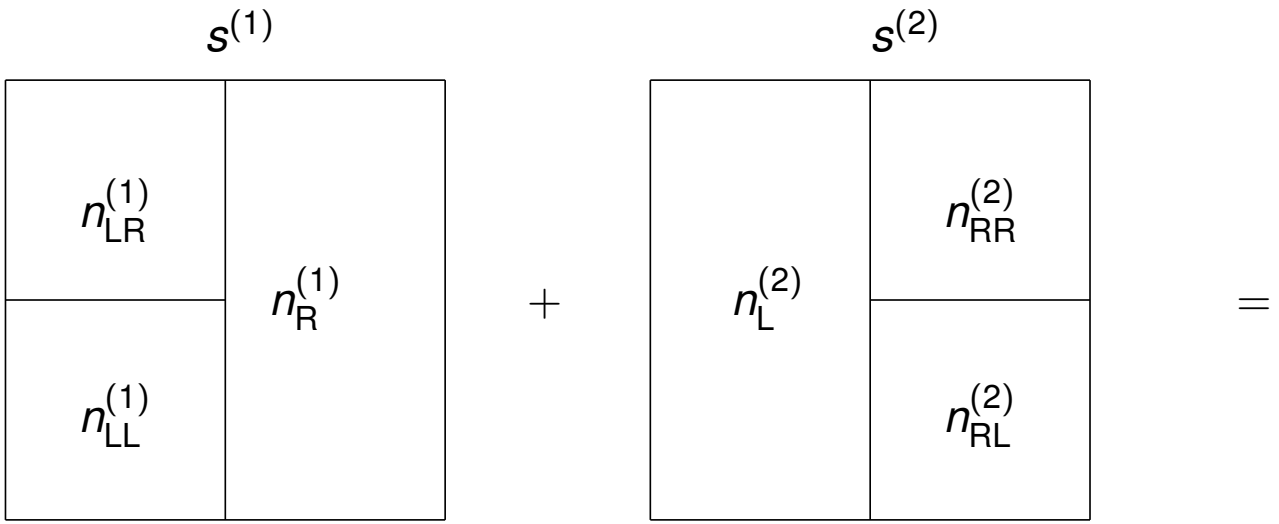


Figure: Two histogram density estimates for the standard bivariate gaussian density with different choices of k_n. The histogram is under-smoothed when k_n is relatively smaller than n and over-smoothed when k_n is relatively larger.

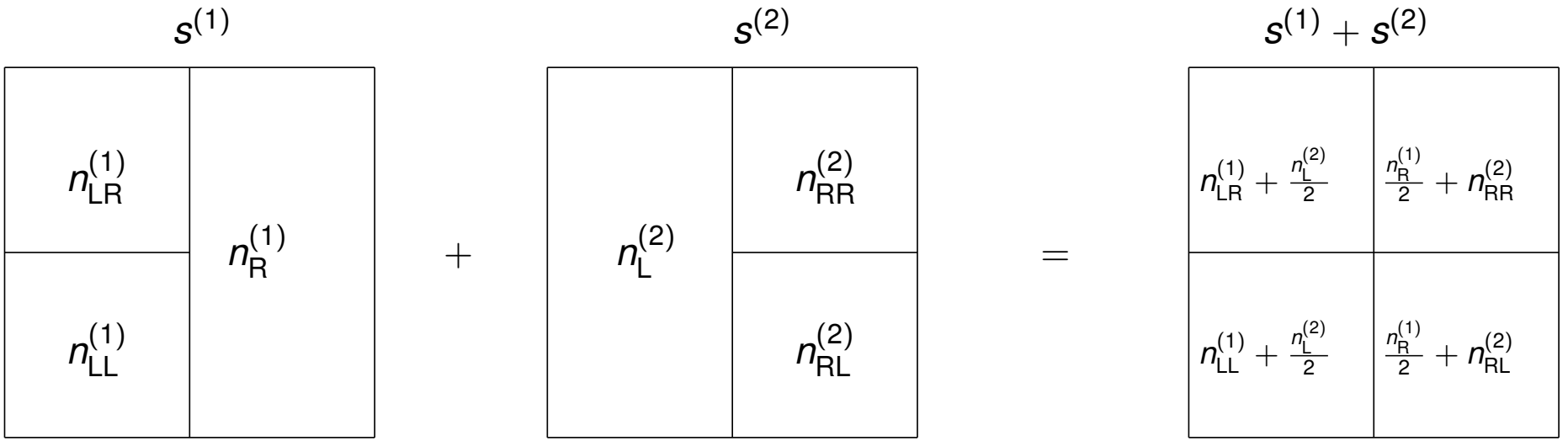
Adding and Averaging SRSPs

Do a non-minimal union (or overlay) operation of $s^{(1)}$ and $s^{(2)}$ and adjust counts:



Adding and Averaging SRSPs

Do a non-minimal union (or overlay) operation of $s^{(1)}$ and $s^{(2)}$ and adjust counts:



Adding and Averaging SRSPs

Adding m SRSP histogram density estimates

$$\sum_{i=1}^{m} f_{n,s^{(i)}} = f_{n,s^{(1)}} + f_{n,s^{(2)}} + f_{n,s^{(3)}} + \ldots + f_{n,s^{(m)}}$$

$$= \left(\left(\left(f_{n,s^{(1)}} + f_{n,s^{(2)}}\right) + f_{n,s^{(3)}}\right) + \ldots + f_{n,s^{(m)}}\right) \quad .$$

Adding and Averaging SRSPs

Adding m SRSP histogram density estimates

$$\sum_{i=1}^{m} f_{n,s^{(i)}} = f_{n,s^{(1)}} + f_{n,s^{(2)}} + f_{n,s^{(3)}} + \ldots + f_{n,s^{(m)}}$$
$$= \left(\left(\left(f_{n,s^{(1)}} + f_{n,s^{(2)}}\right) + f_{n,s^{(3)}}\right) + \ldots + f_{n,s^{(m)}}\right) \quad .$$

Averaging m SRSP histogram density estimates recursively yields the sample mean SRSP histogram

$$\bar{f}_{n,m} = \frac{1}{m} \sum_{i=1}^{m} f_{n,s^{(i)}} \quad .$$

An Example

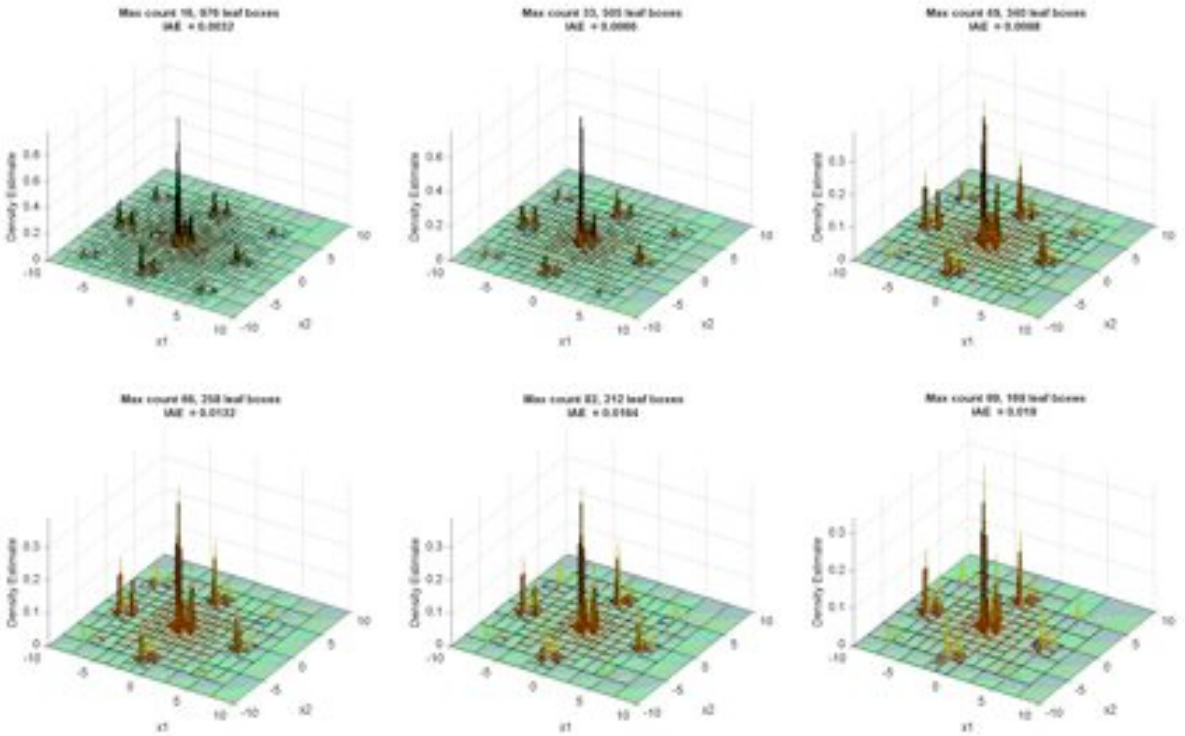


Figure: Histogram density estimates of the bivariate Levy using different values of k_n.

An Example

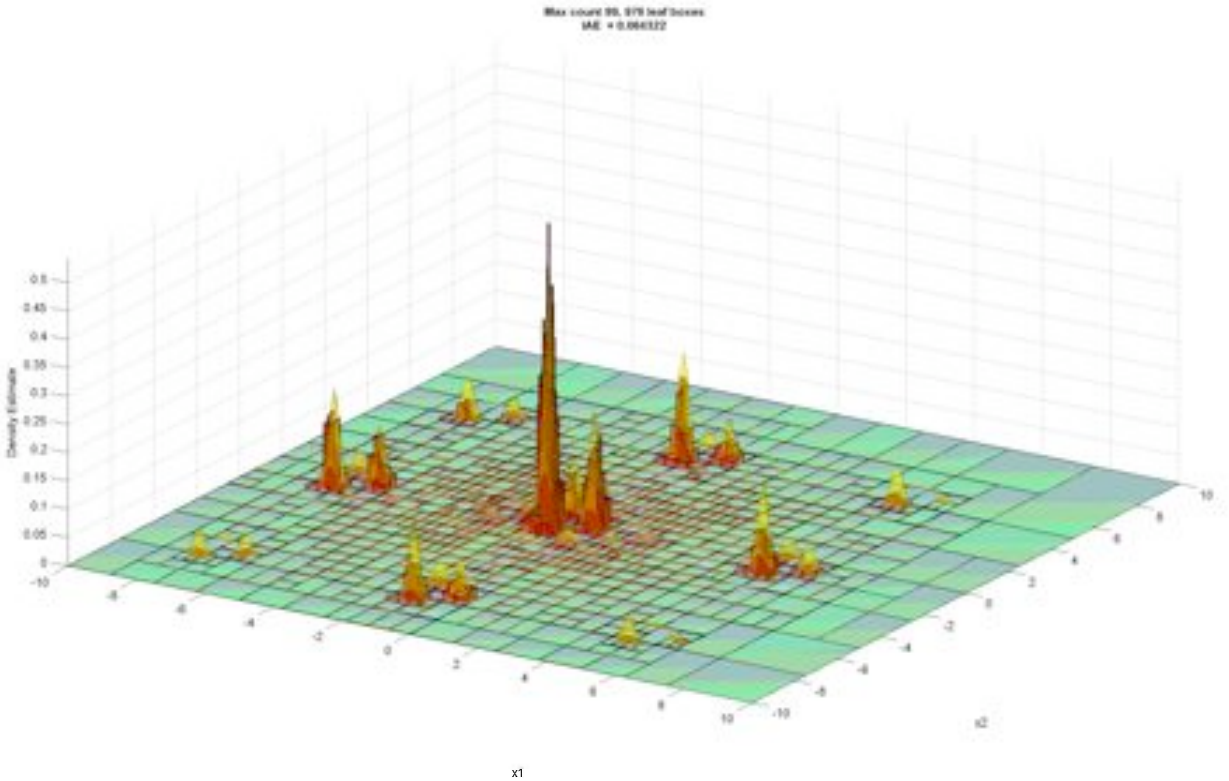


Figure: The averaged histogram density estimate.

Posterior Distribution over Histograms in $\mathbb{S}_{0:\infty}$

- Let $\hat{f}_s$ be a histogram with partition $\ell(s)$ given by the leaves of RSP s with k splits and $k+1$ leaves in $\mathbb{S}_k$

Posterior Distribution over Histograms in $\mathbb{S}_{0:\infty}$

- Let $\hat{f}_s$ be a histogram with partition $\ell(s)$ given by the leaves of RSP s with k splits and $k+1$ leaves in $\mathbb{S}_k$
- Then for this partition, the most likely histogram estimate is

$$\hat{f}_s(x; \text{data}) = \frac{1}{n}\hat{f}_s(x; X_{1:n}) = \sum_{i=1}^{n} \frac{\mathbb{1}\left(x_i \in \mathbf{x}(x)\right)}{vol(\mathbf{x}(x))}$$

Posterior Distribution over Histograms in $\mathbb{S}_{0:\infty}$

- Let $\hat{f}_s$ be a histogram with partition $\ell(s)$ given by the leaves of RSP s with k splits and $k+1$ leaves in $\mathbb{S}_k$
- Then for this partition, the most likely histogram estimate is

$$\hat{f}_s(x; \text{data}) = \frac{1}{n}\hat{f}_s(x; X_{1:n}) = \sum_{i=1}^{n} \frac{\mathbb{1}\left(x_i \in \mathbf{x}(x)\right)}{vol(\mathbf{x}(x))}$$

- Let the prior probability be $P(s) \propto \frac{1}{C_k^2}$, $s \in \mathbb{S}_{0:\infty}$

Posterior Distribution over Histograms in $\mathbb{S}_{0:\infty}$

- Let $\hat{f}_s$ be a histogram with partition $\ell(s)$ given by the leaves of RSP s with k splits and $k+1$ leaves in $\mathbb{S}_k$
- Then for this partition, the most likely histogram estimate is

$$\hat{f}_s(x; \text{data}) = \frac{1}{n}\hat{f}_s(x; X_{1:n}) = \sum_{i=1}^{n} \frac{\mathbb{1}\left(x_i \in \mathbf{x}(x)\right)}{vol(\mathbf{x}(x))}$$

- Let the prior probability be $P(s) \propto \frac{1}{C_k^2}$, $s \in \mathbb{S}_{0:\infty}$
- Then the posterior density of histogram $\hat{f}_s$ with k splits is

$$P(\hat{f}_s|X_{1:n}) \propto P(X_{1:n}|s)P(s) = \prod_{\mathbf{x}\in\ell(s)} \left(\frac{n_{\mathbf{x}}}{n \text{ vol}(\mathbf{x})}\right)^{n_{\mathbf{x}}} \frac{1}{C_k^2}$$

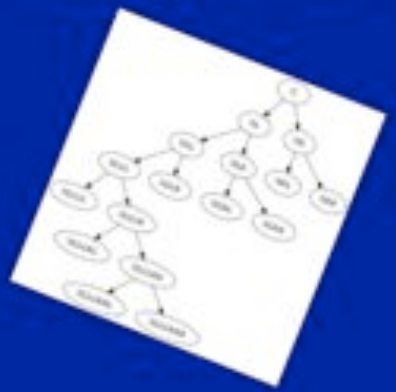
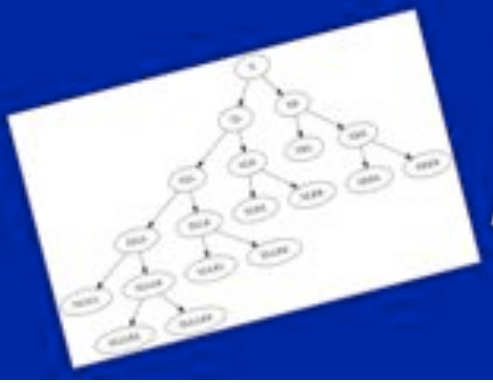
Metropolis-Hastings Algorithm

- Use a proposal density $q(s^{prime}|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state s'
- We propose uniformly at random to split a leaf or merge a cherry of current SRSP state $s^{(i)}$
- **Repeat**
  - **Draw** $u \sim U(0,1)$
  - **If** $u < \frac{P(\hat{f}_{s'}|X_{1:n})}{P(\hat{f}_{s^{(i)}}|X_{1:n})} \frac{q(s^{(i)}|s')}{q(s^{(prime)}|s^{(i)})}$ **then** $s^{(i+1)} \leftarrow s'$
  - **else** $s^{(i+1)} \leftarrow s^{(i)}$
- With a "long enough" burn-in time, this Markov chain will be at the desired stationary distribution $P(\hat{f}_s|X_{1:n})$ over $\mathbb{S}_{0:\infty}$

Metropolis-Hastings Algorithm

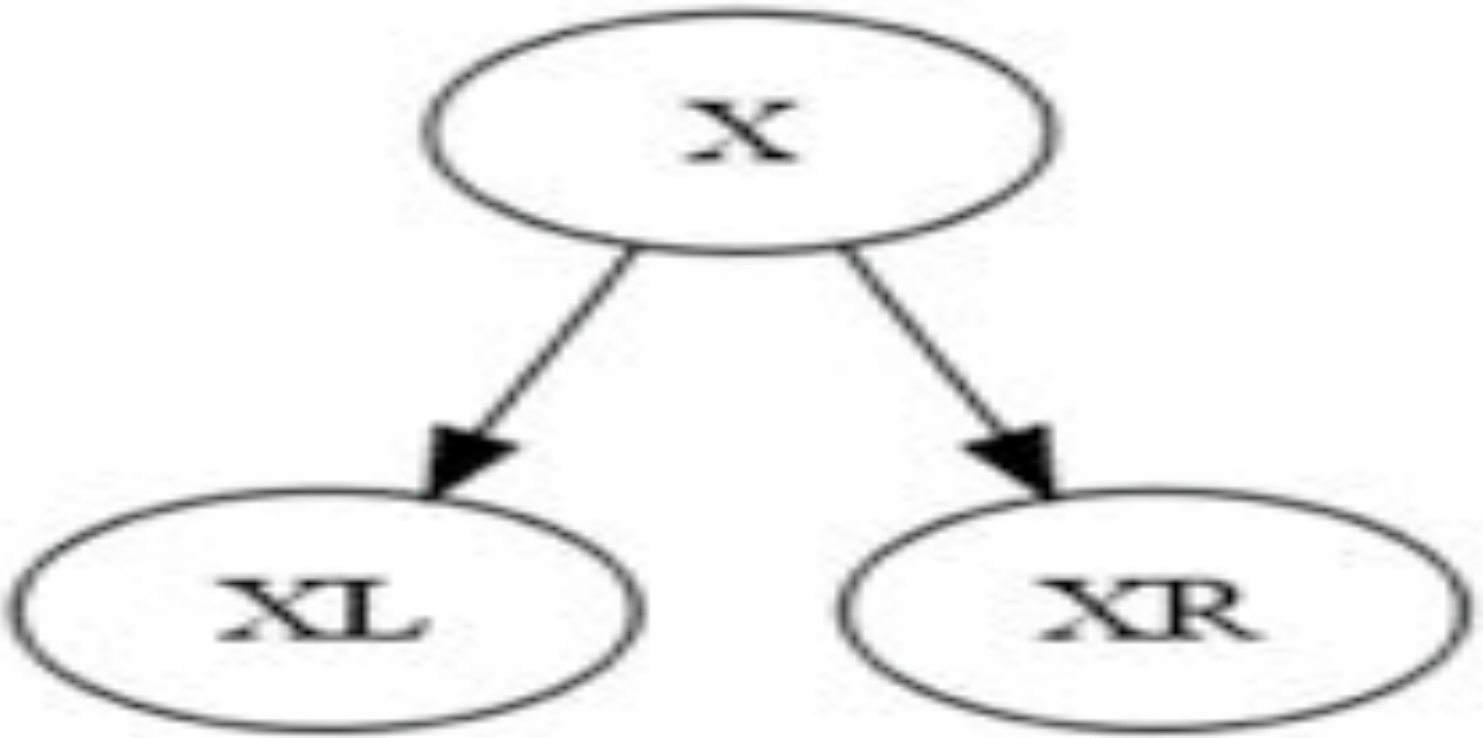
- Start from some initial state m^0

- Burn-in: run until initial state is 'forgotten'
- States after burn-in are sample histograms

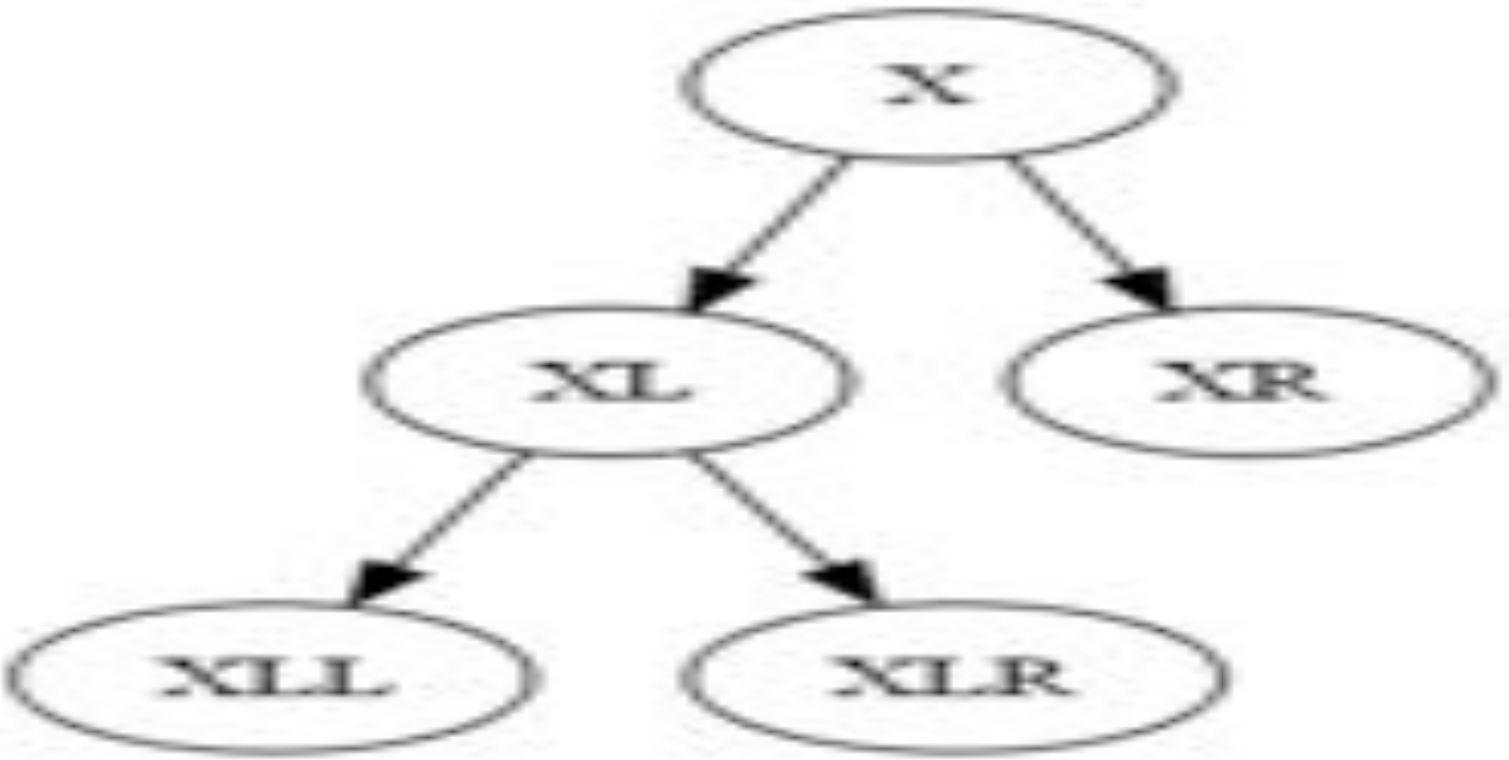




etc...

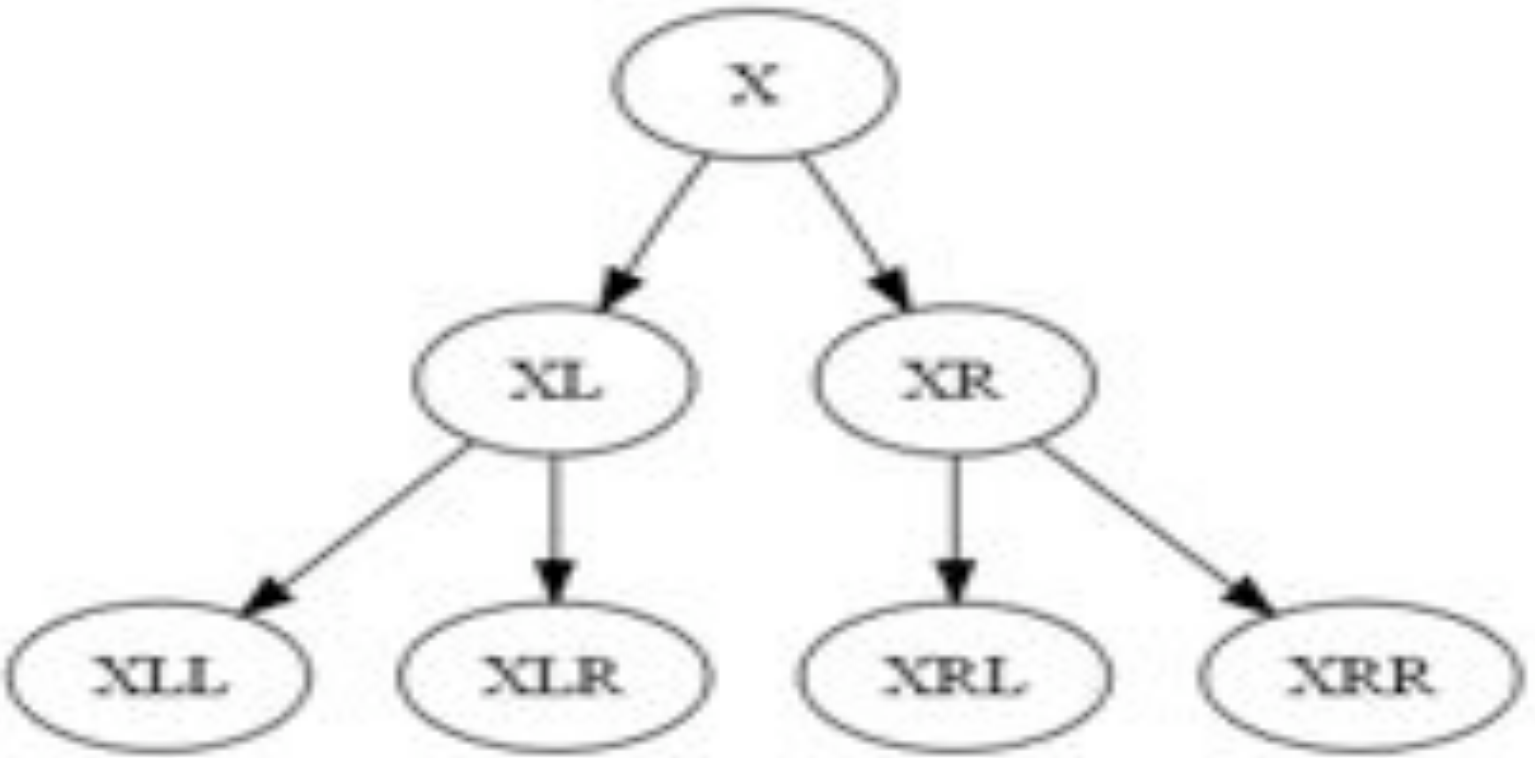
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



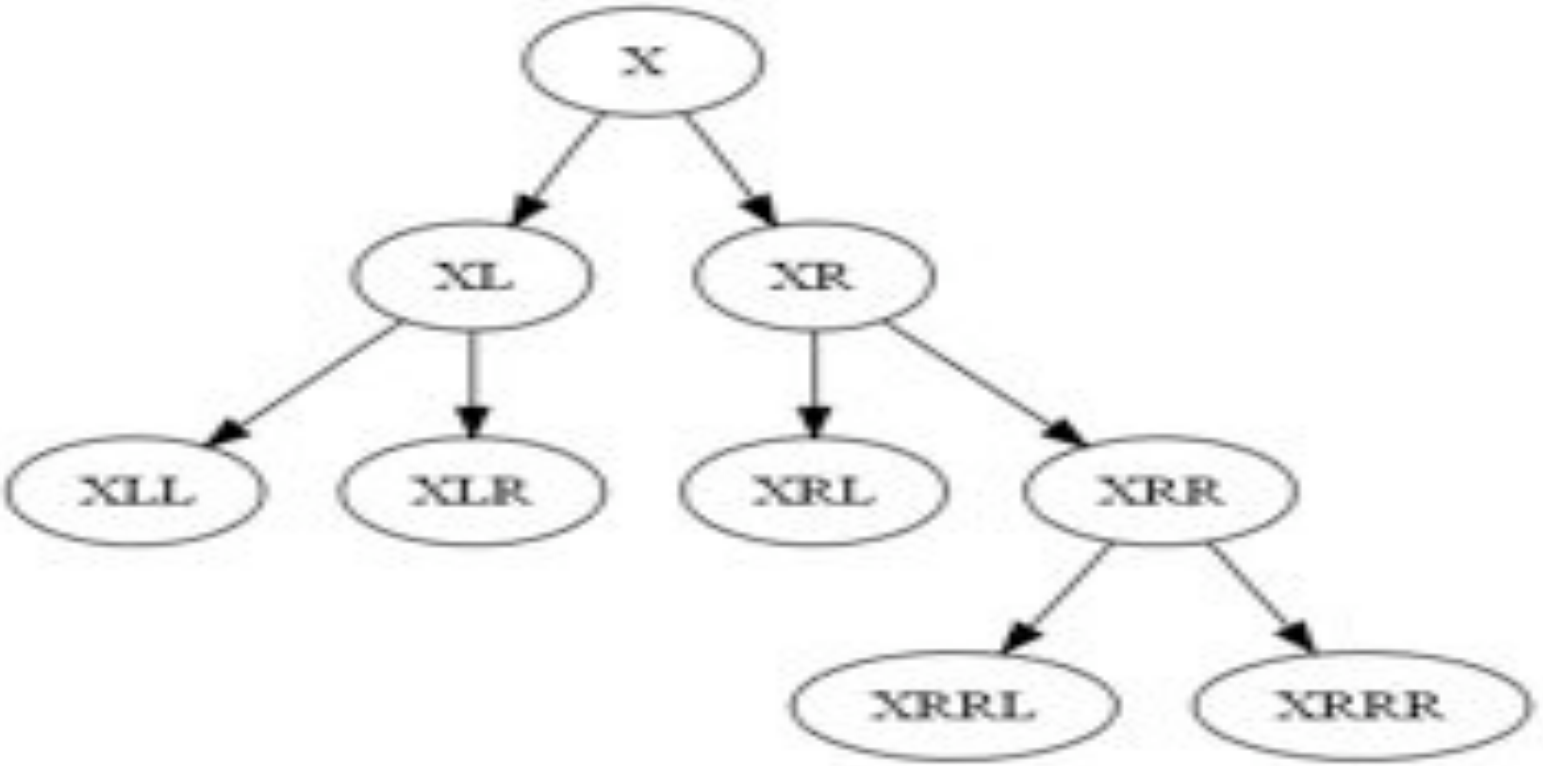
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



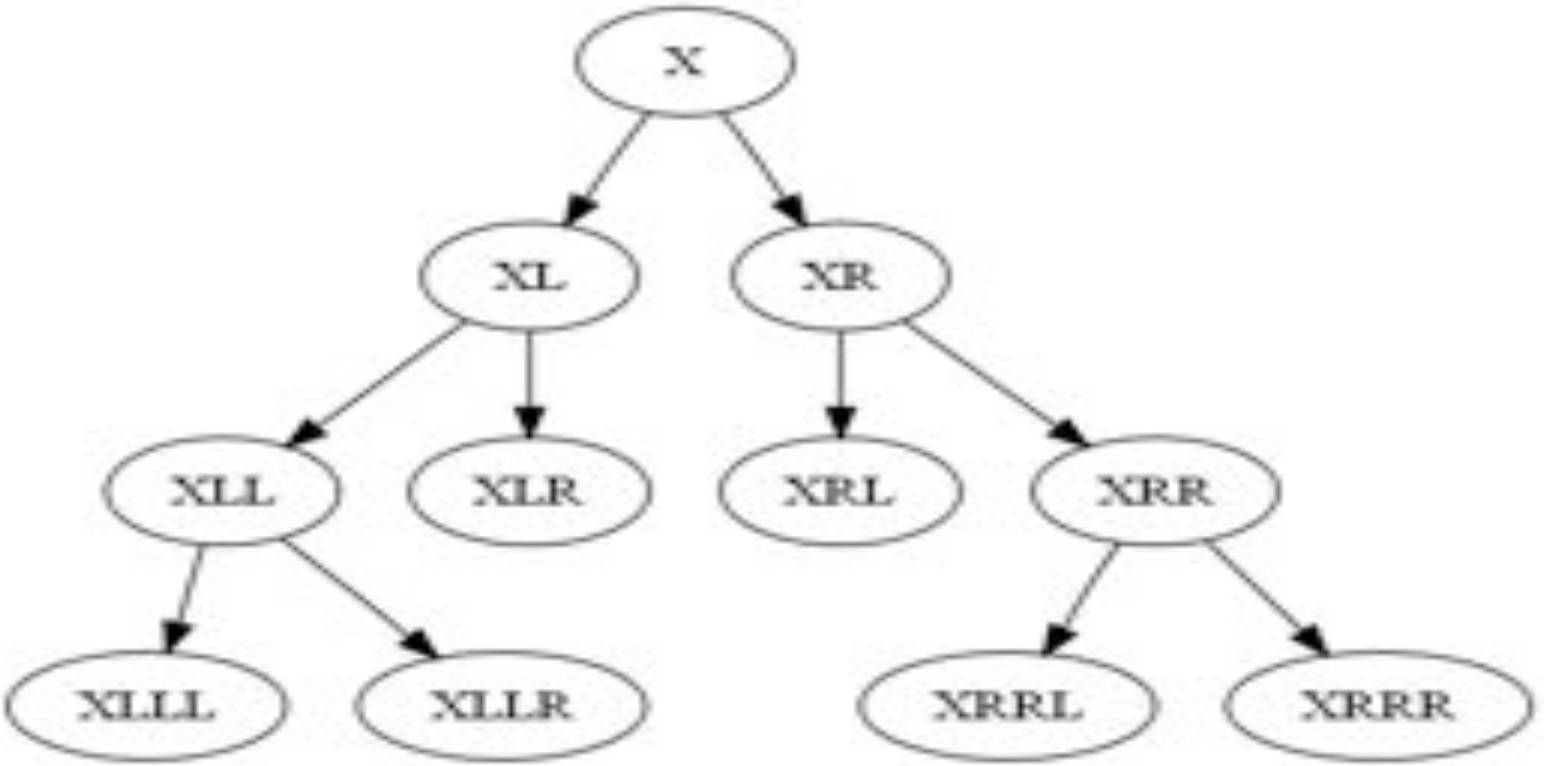
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



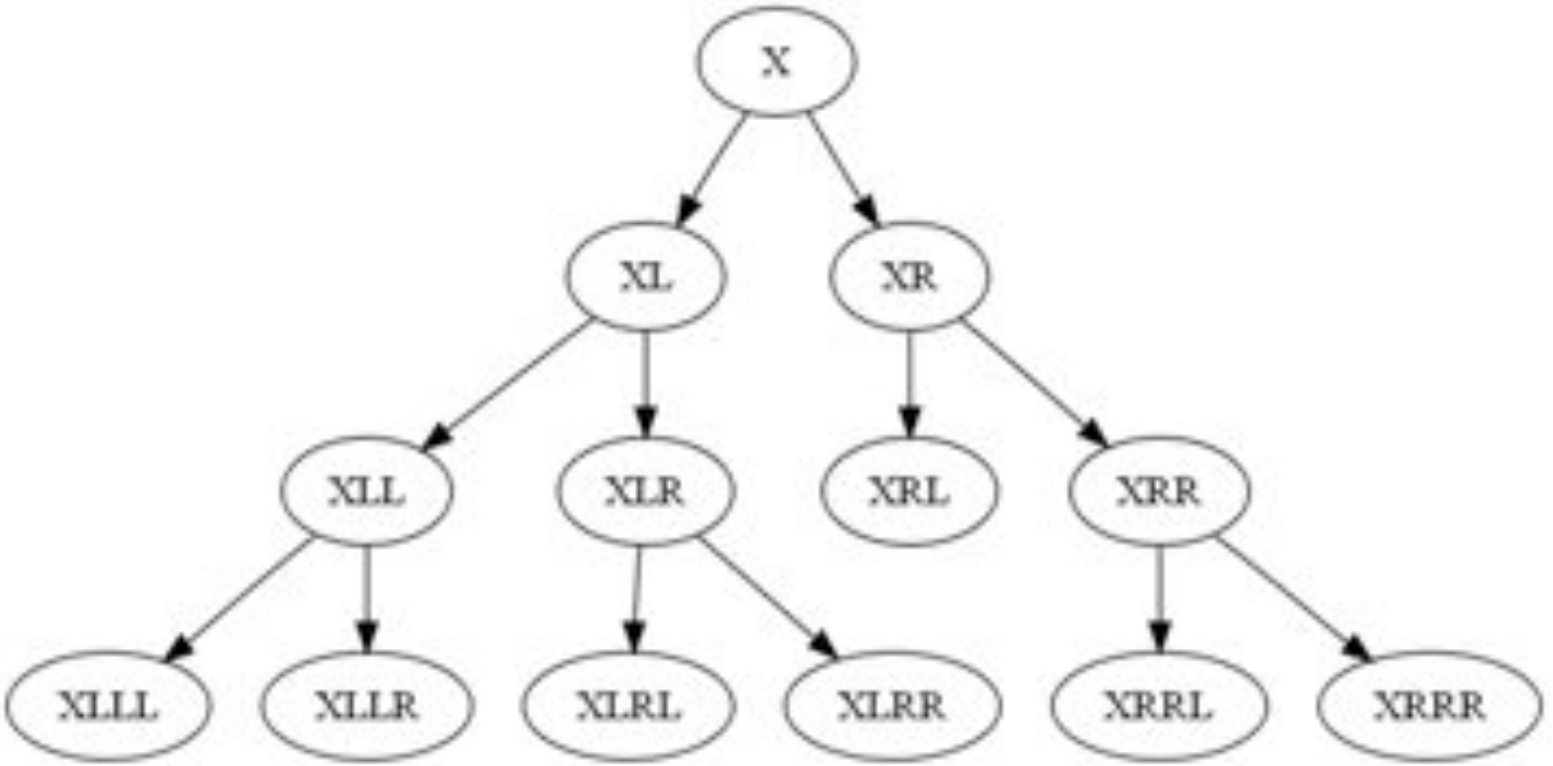
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



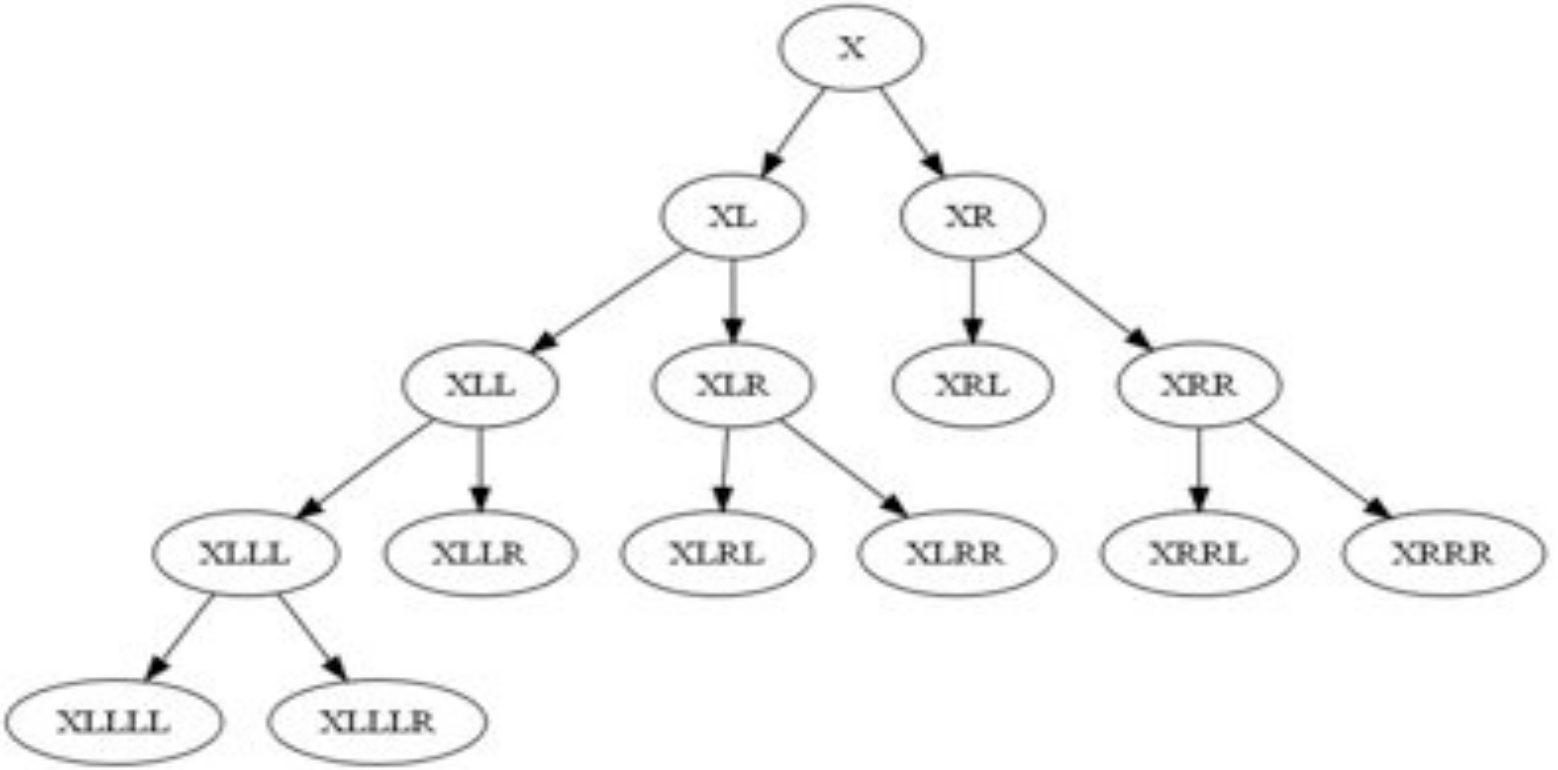
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



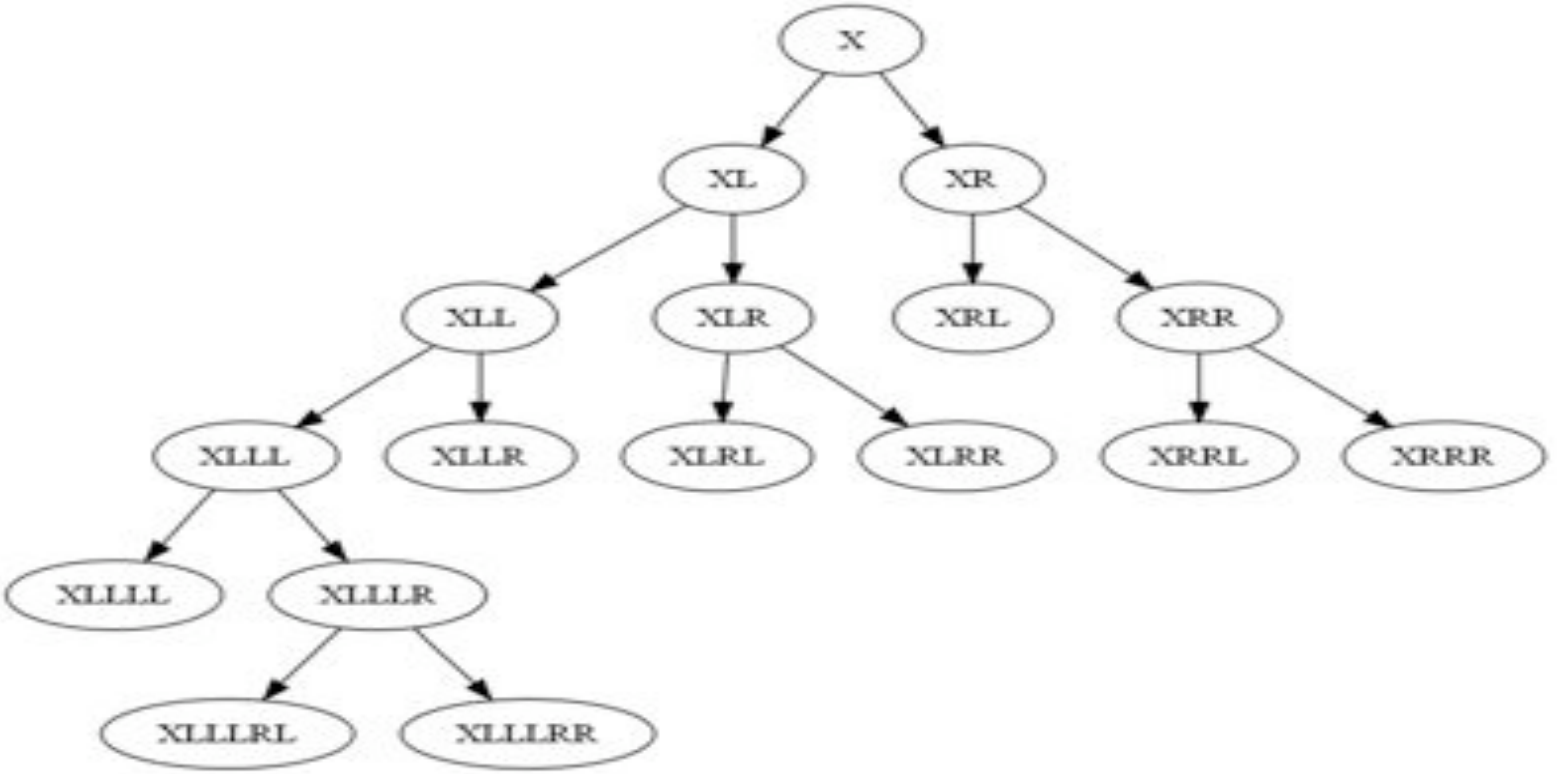
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



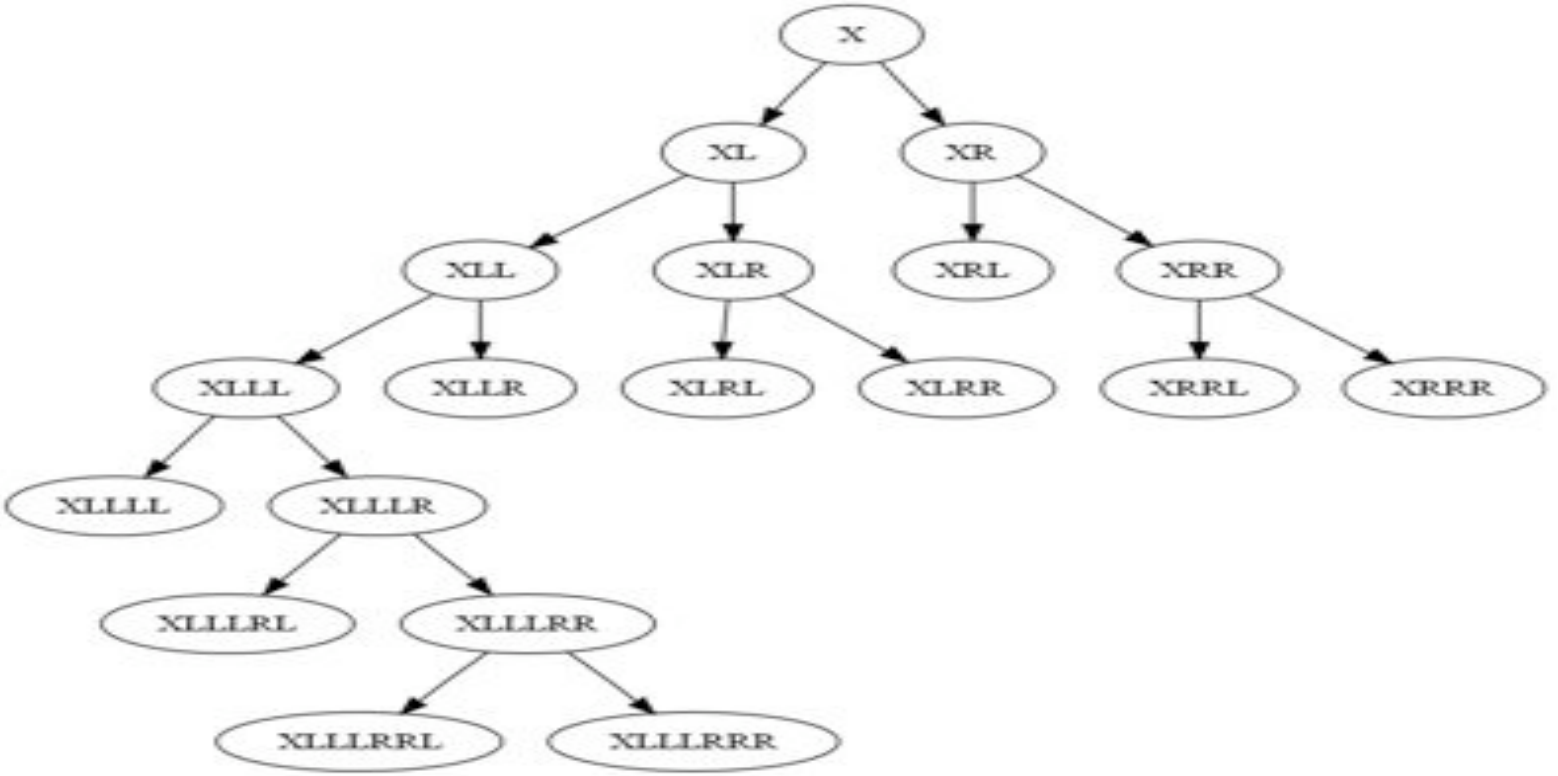
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



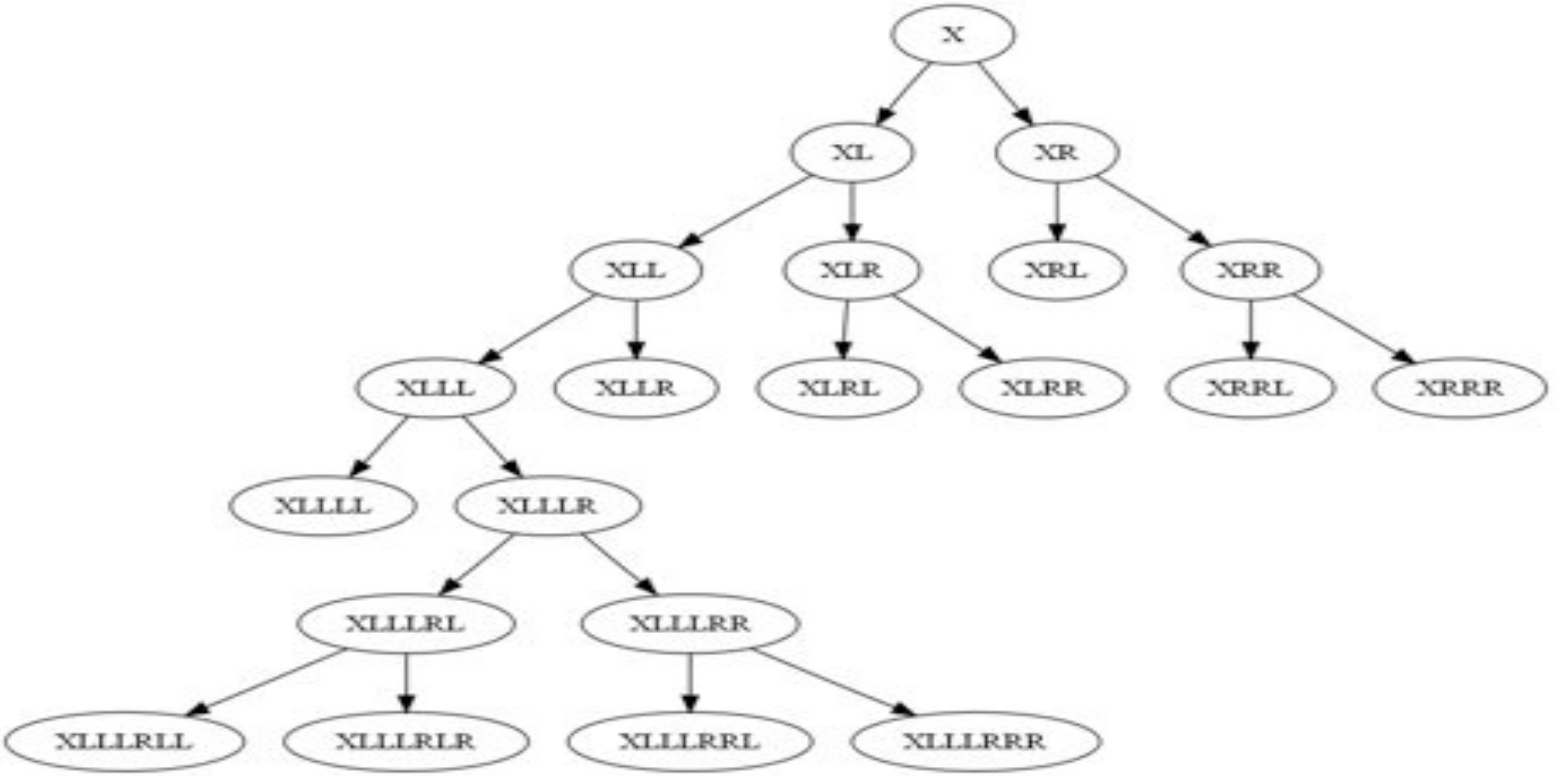
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



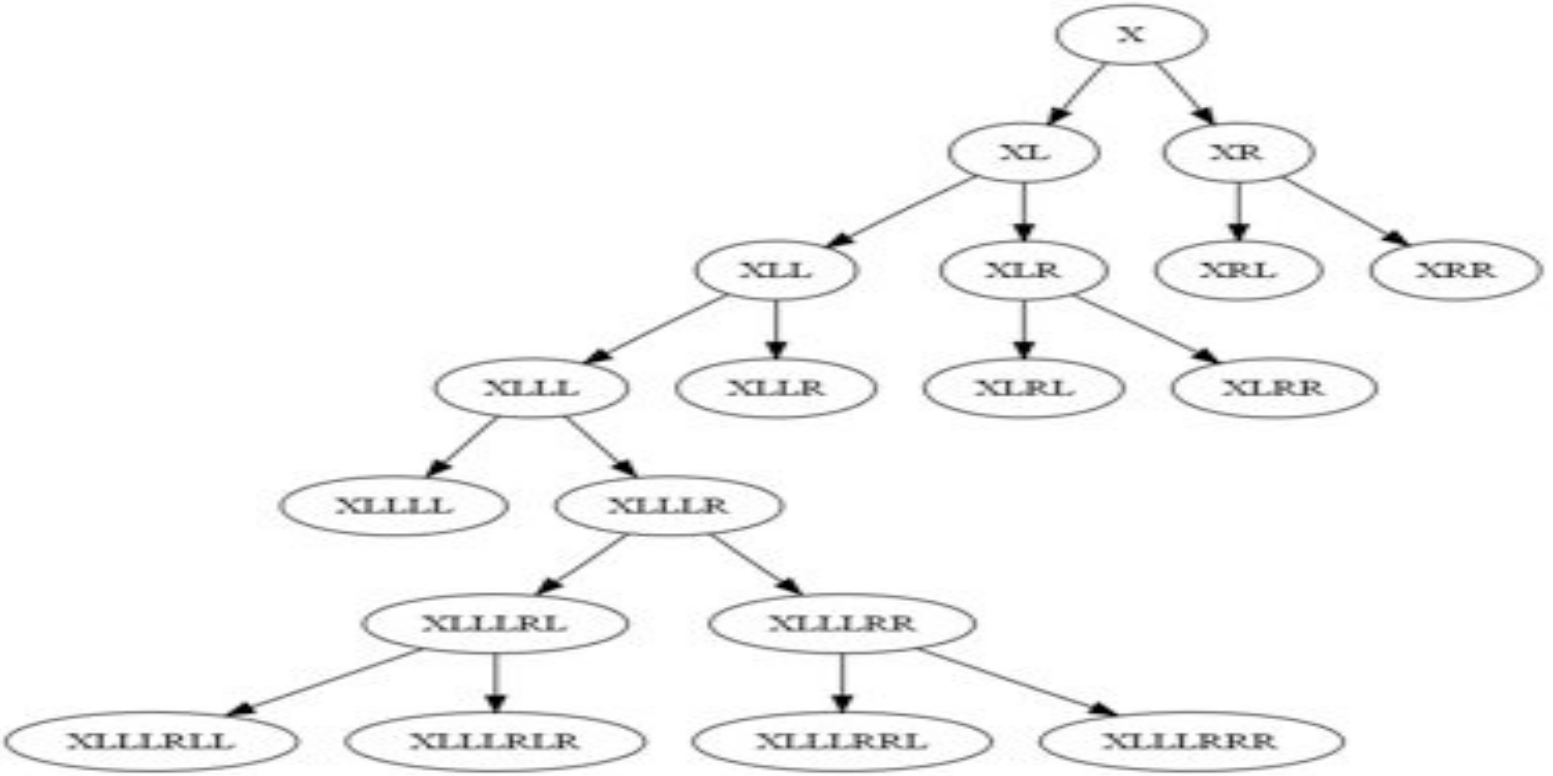
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



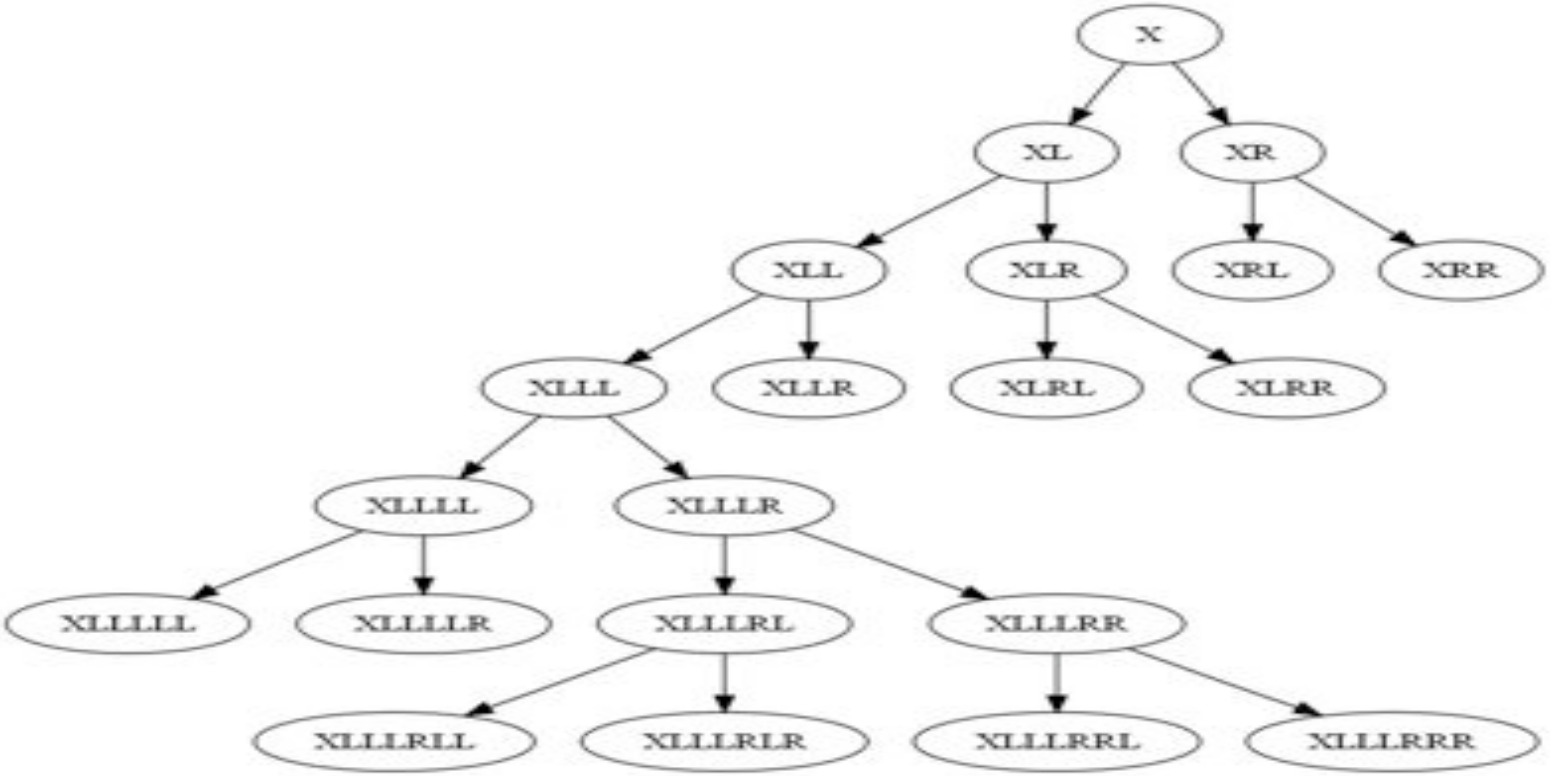
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



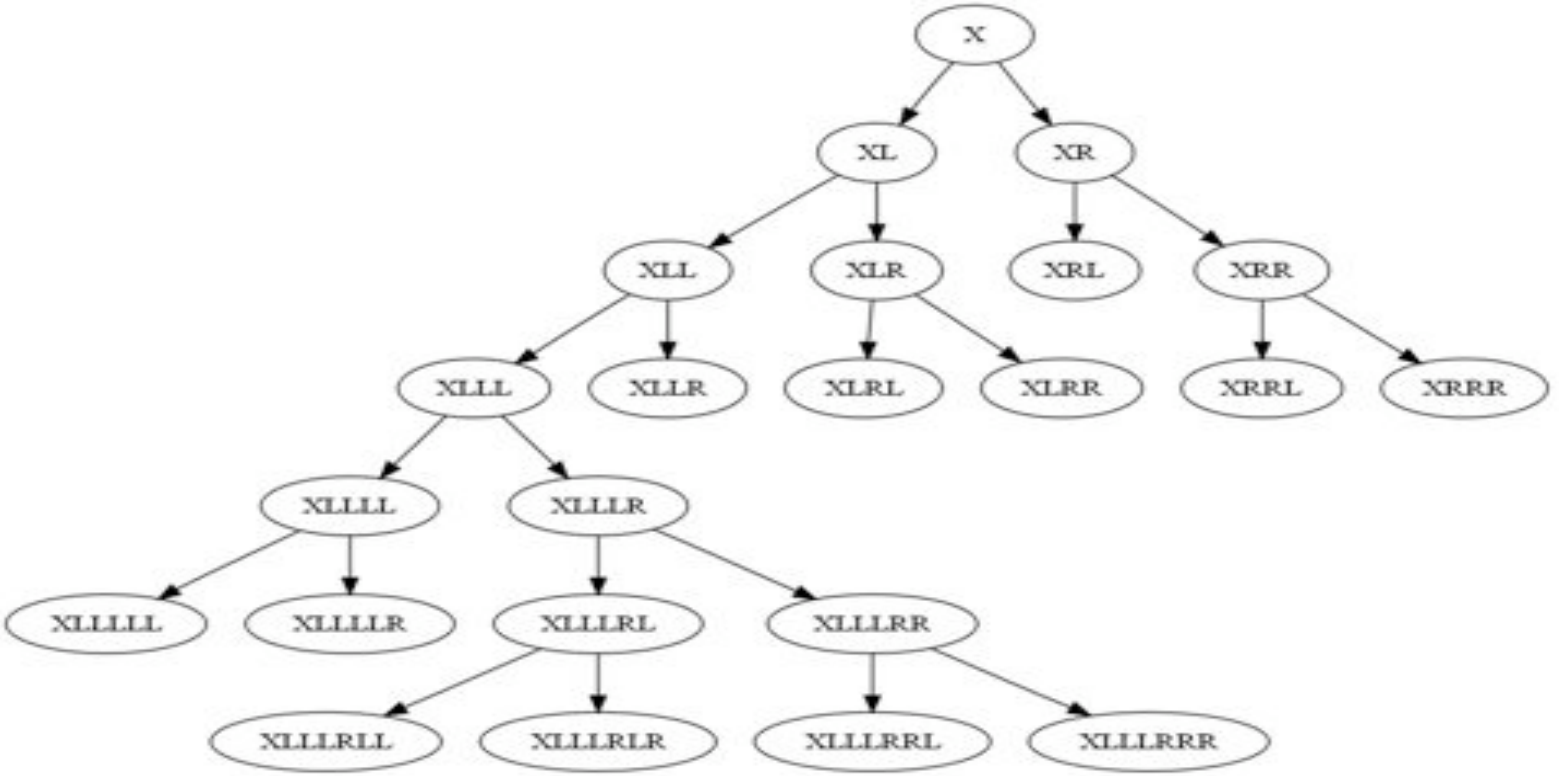
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



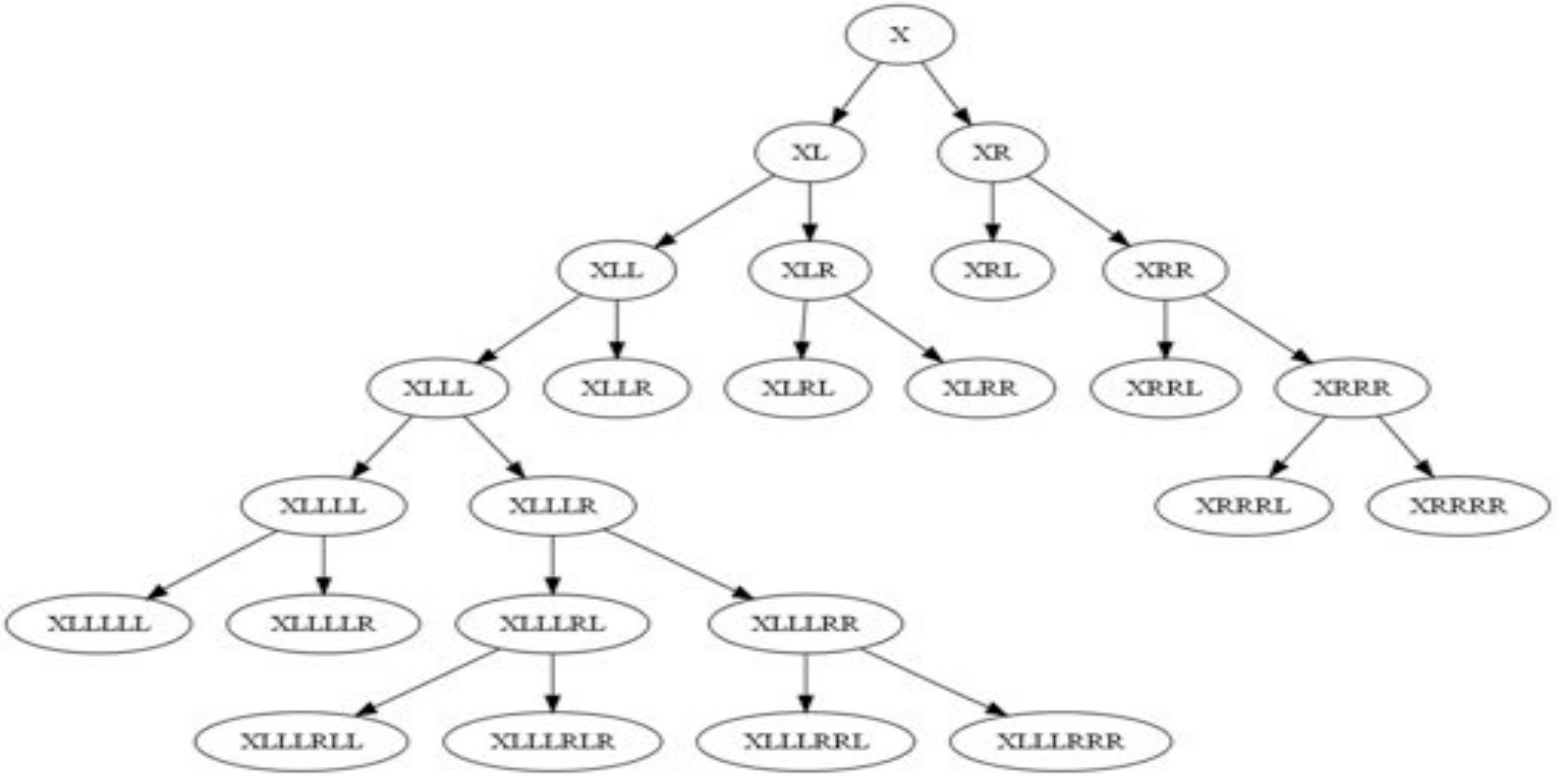
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



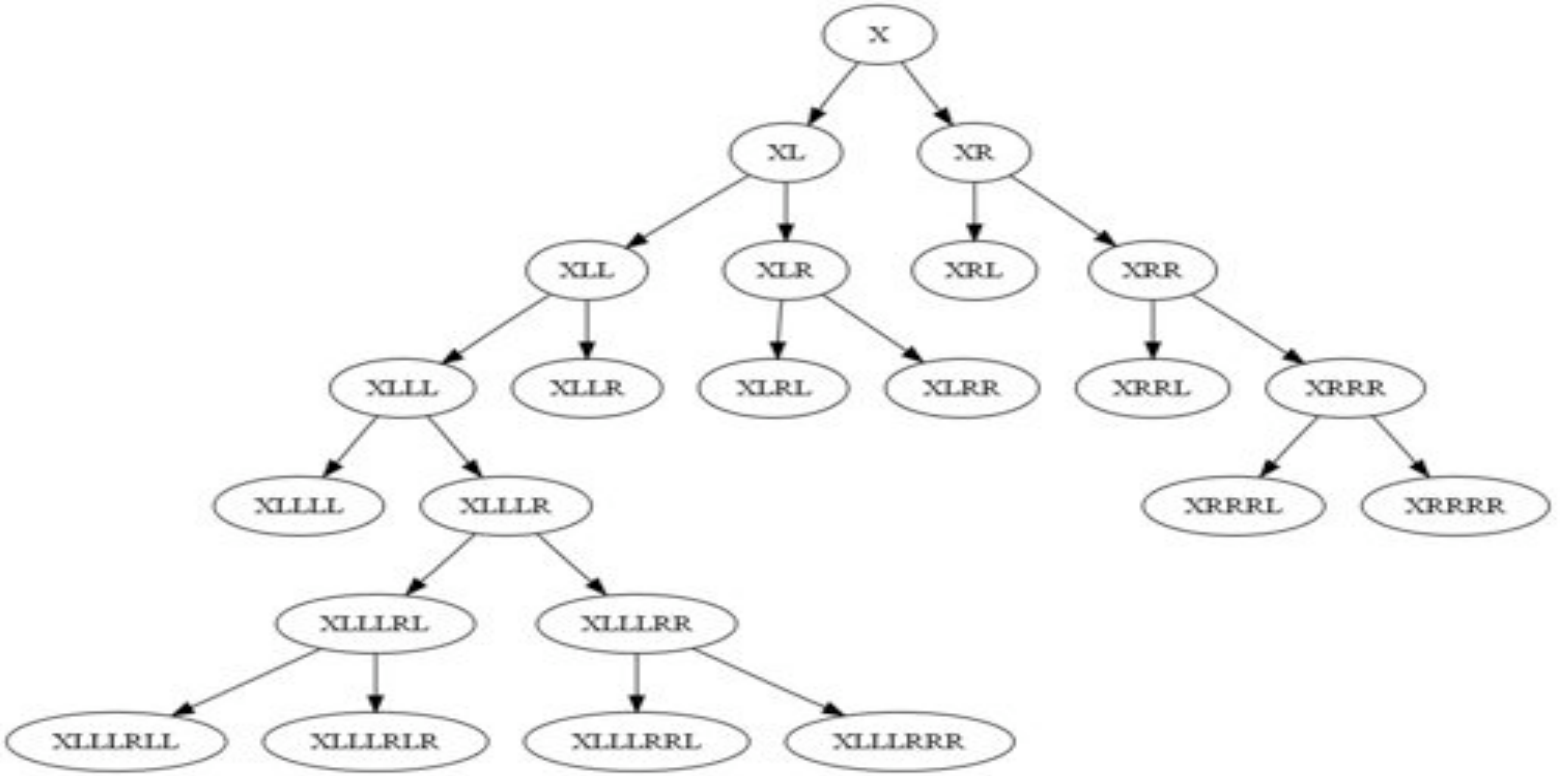
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



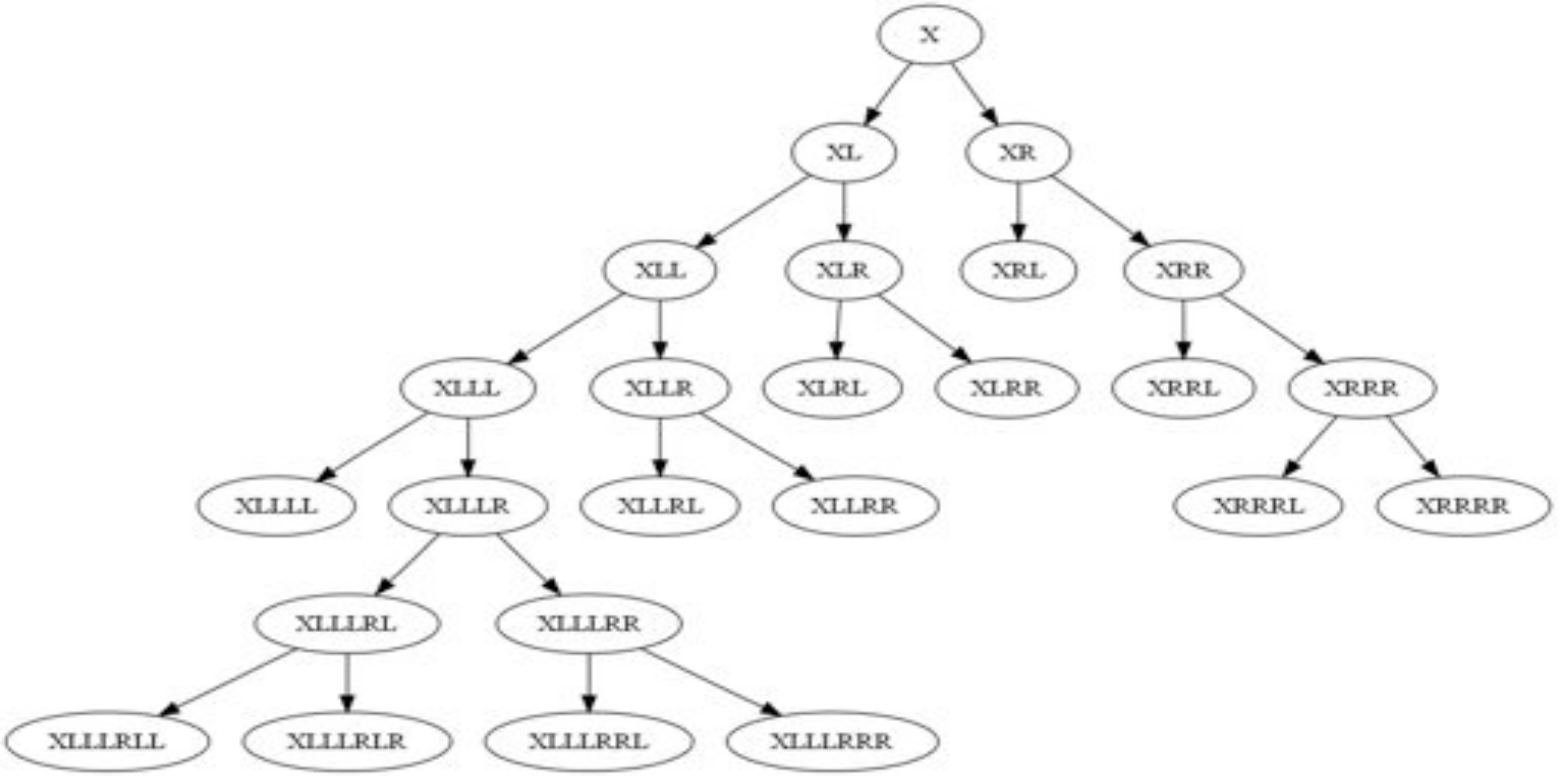
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



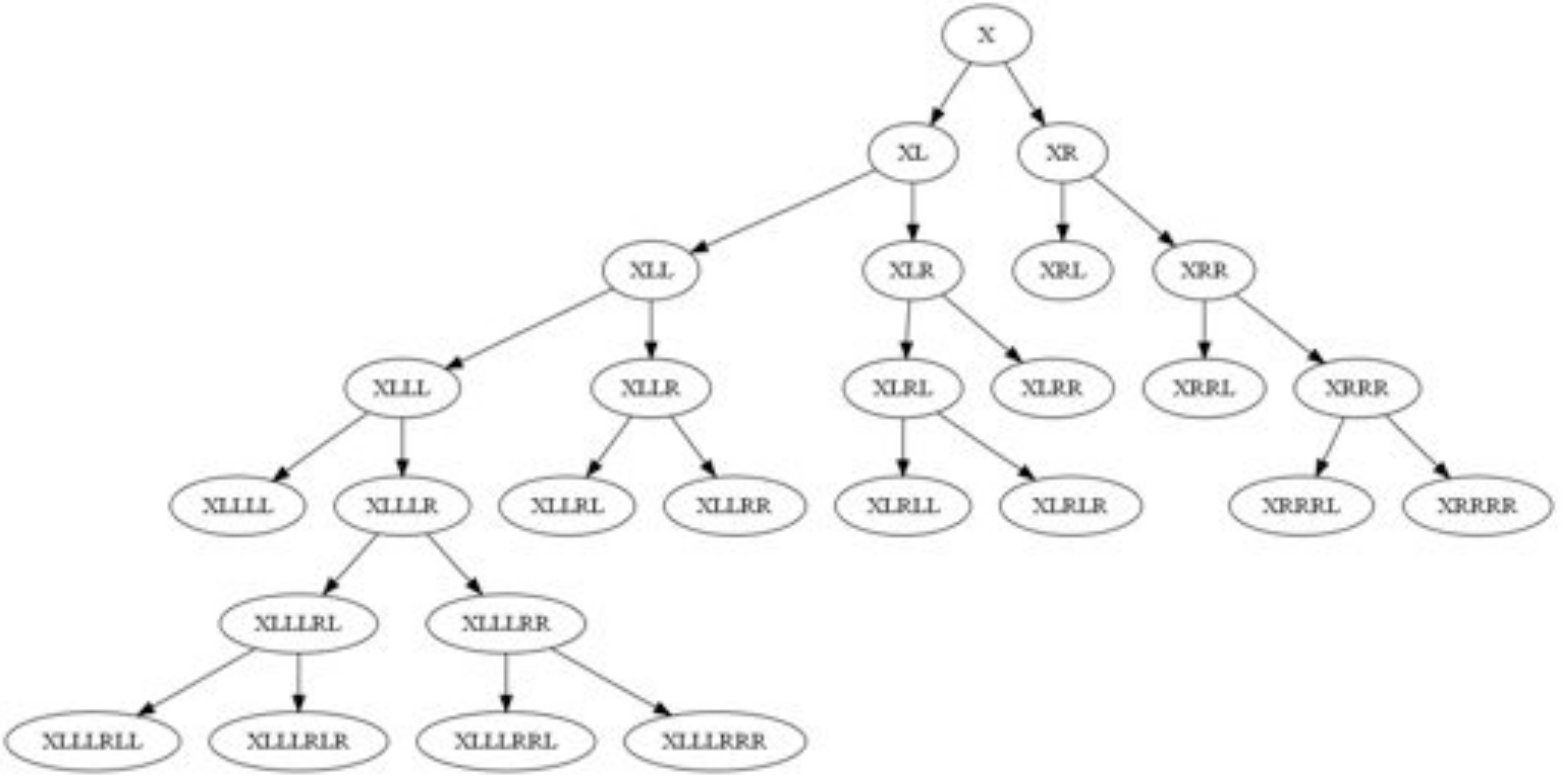
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



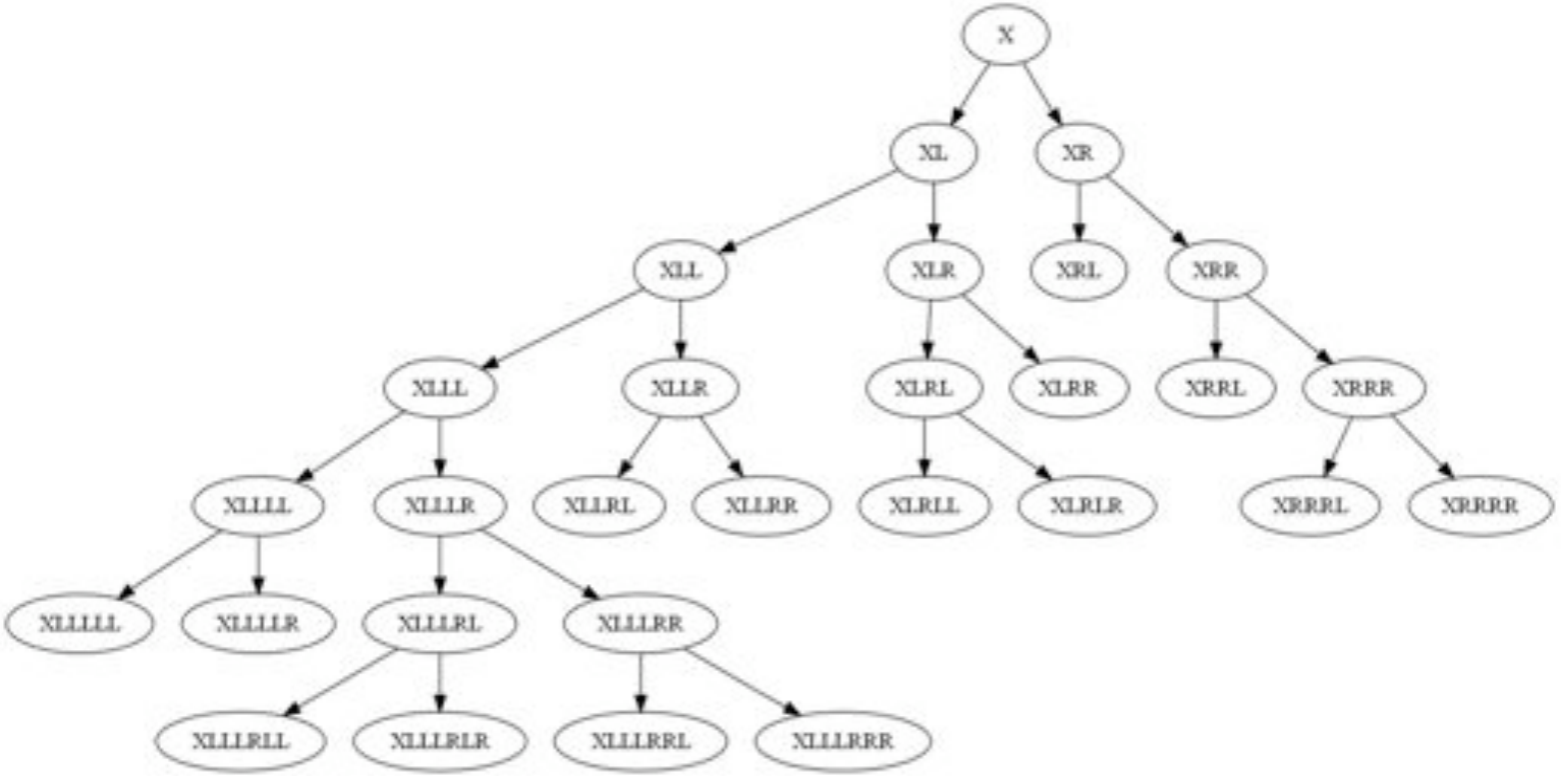
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



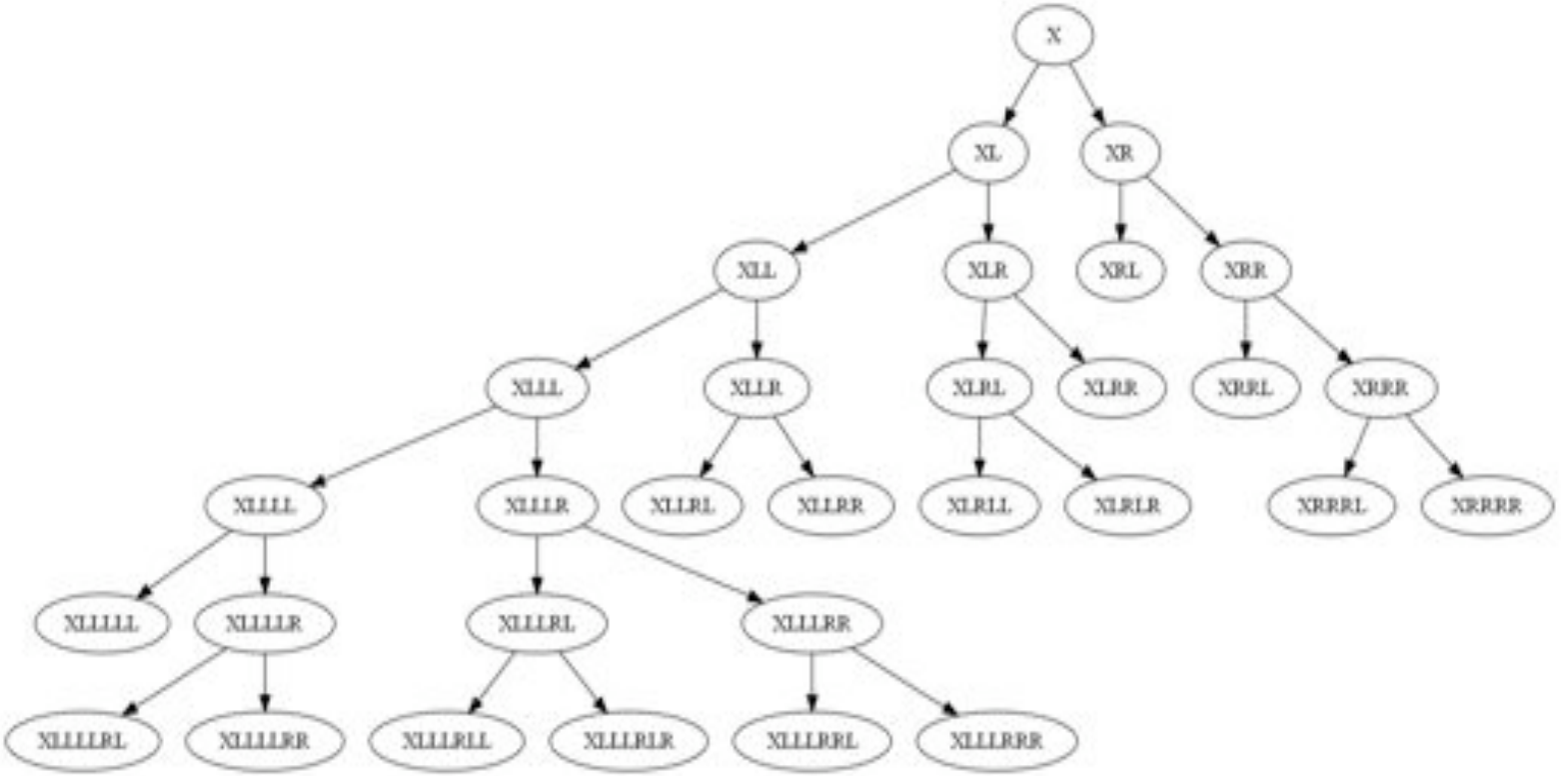
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



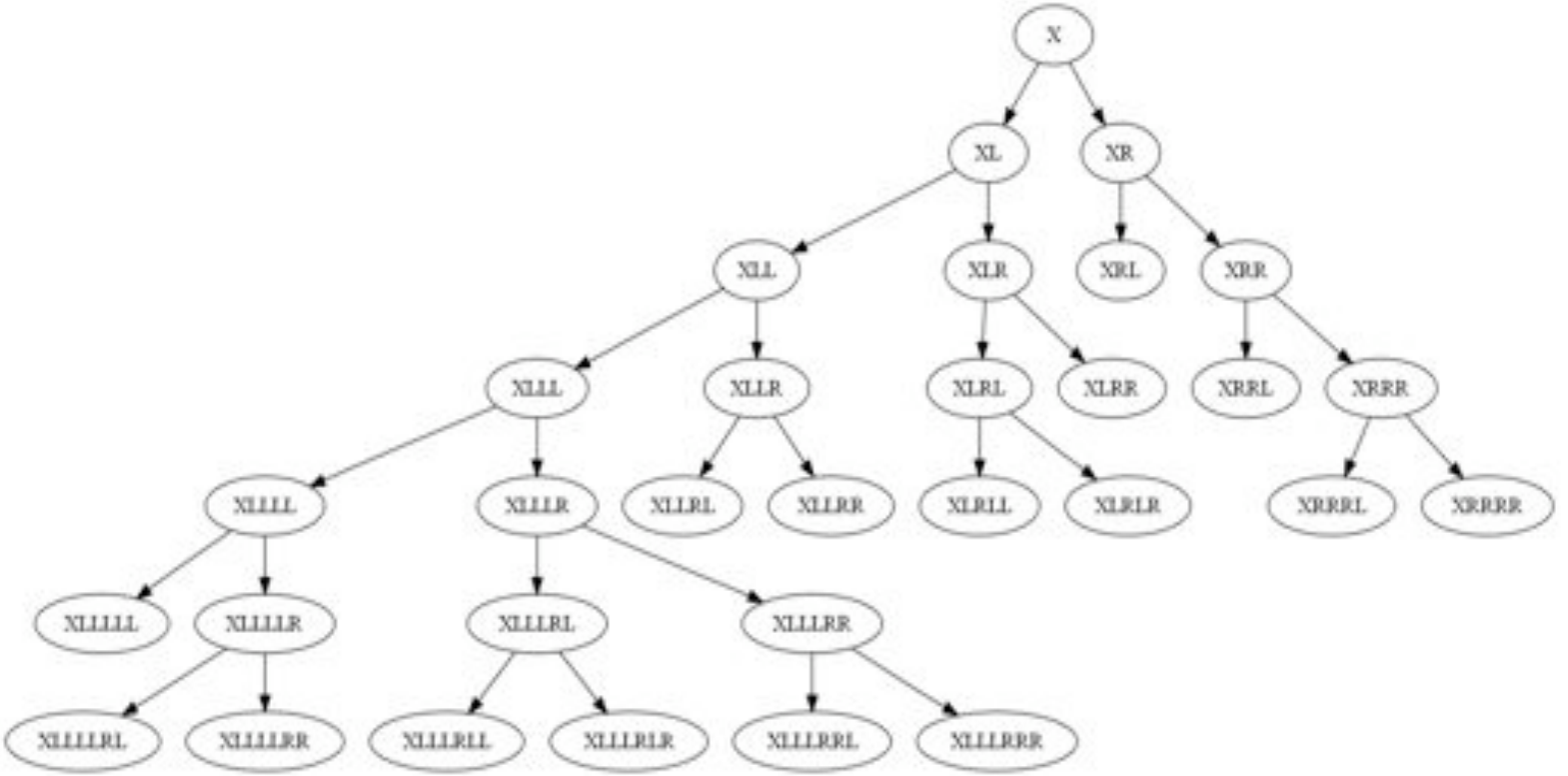
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



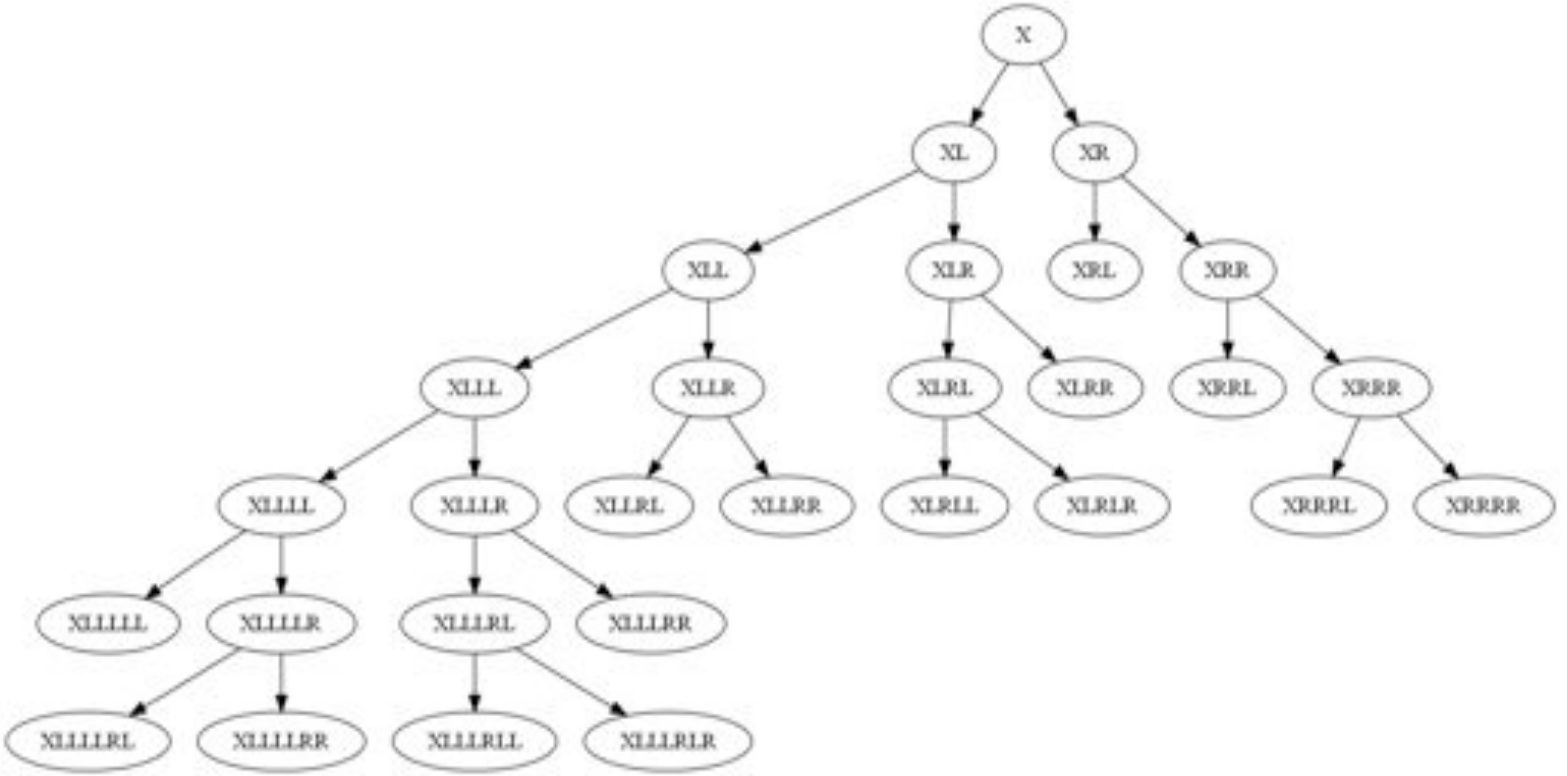
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



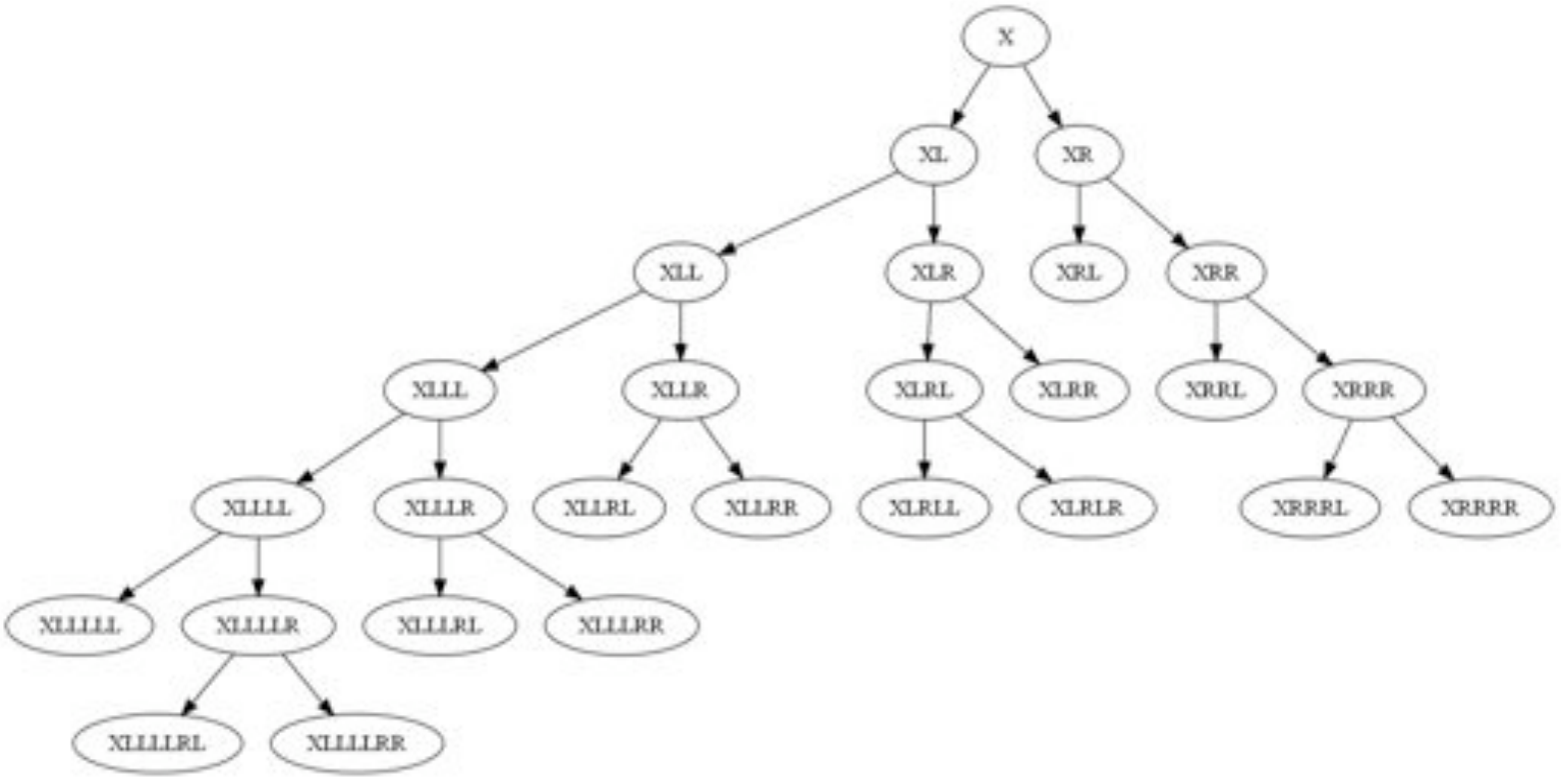
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



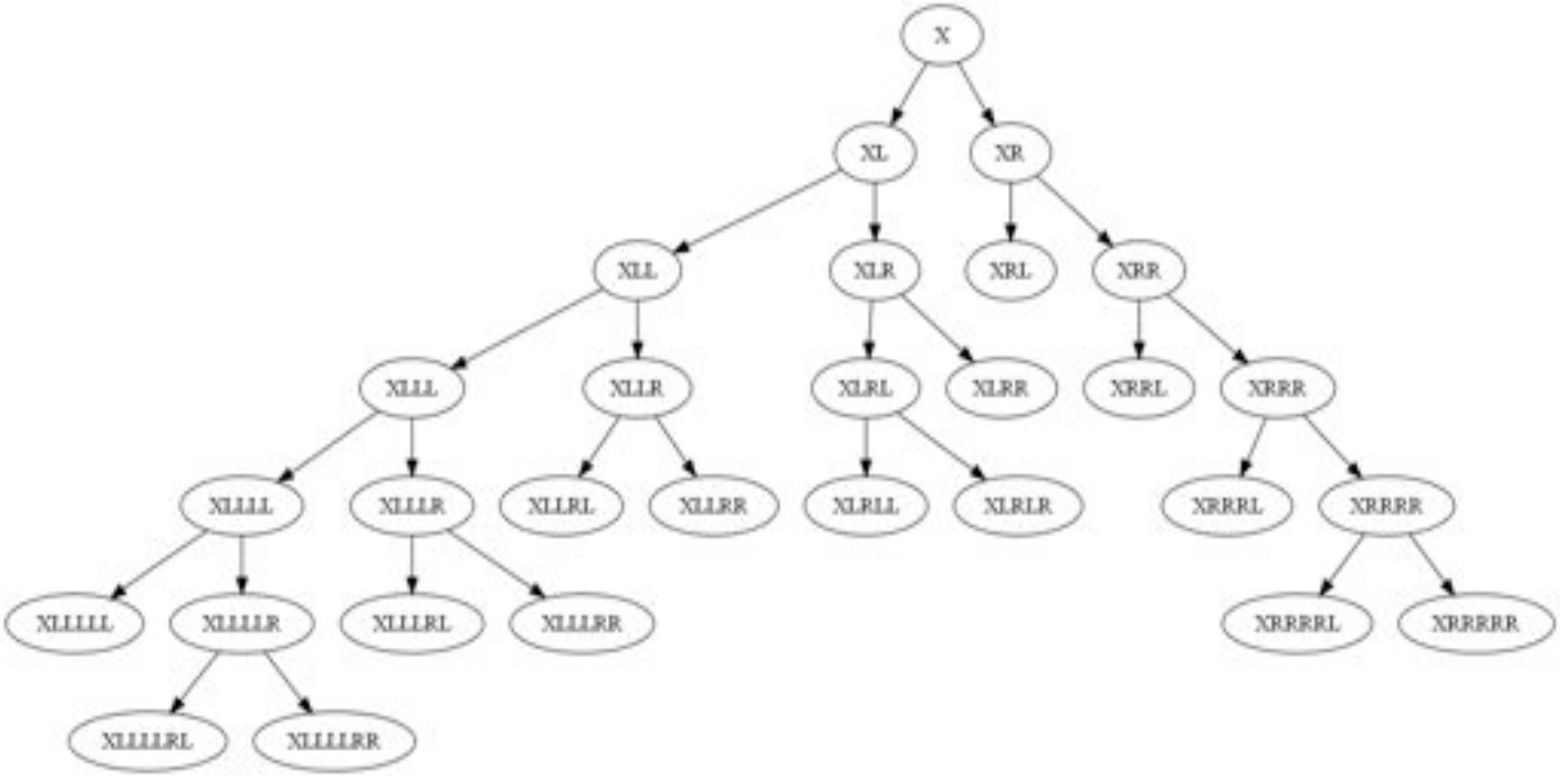
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



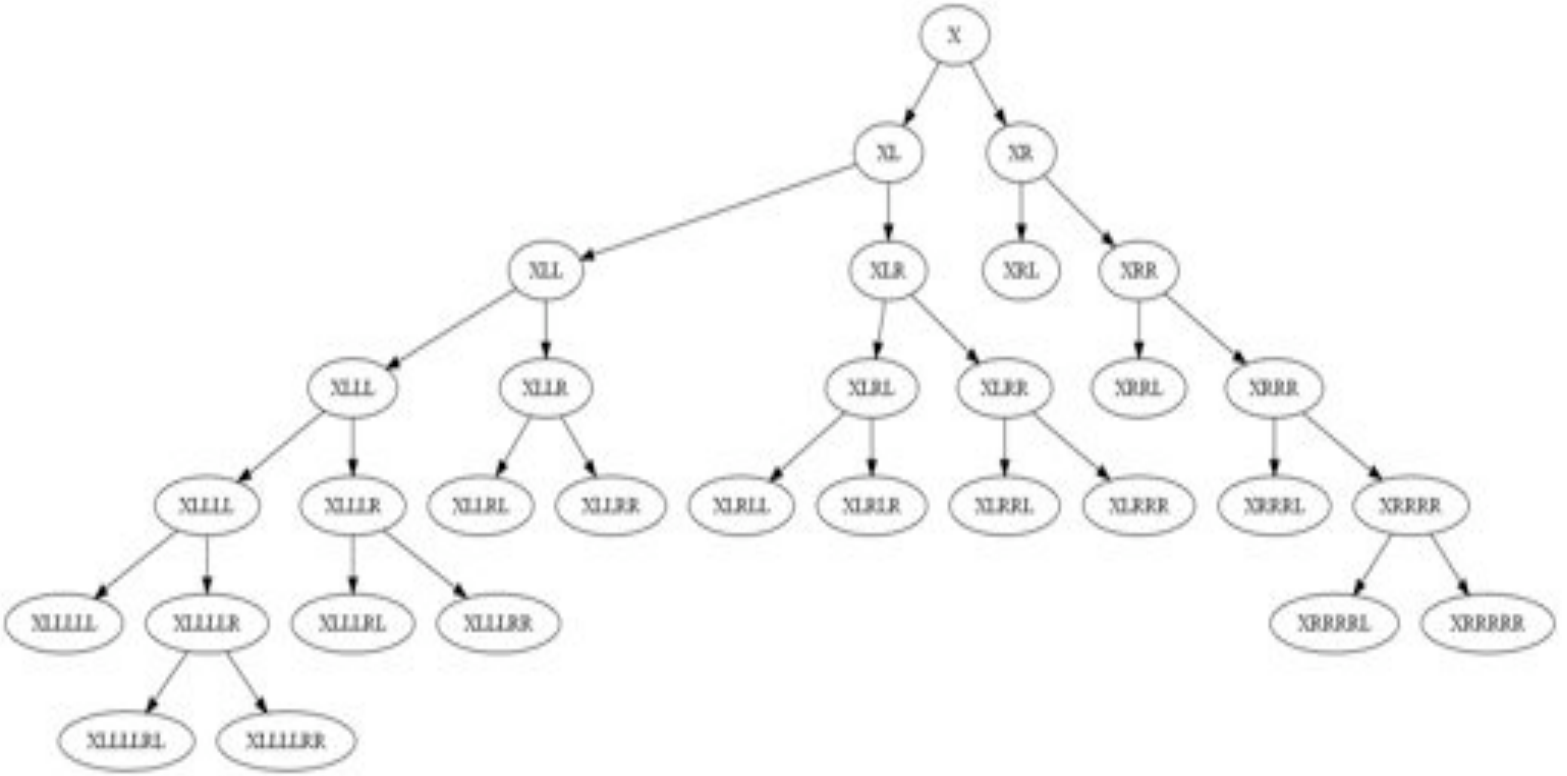
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



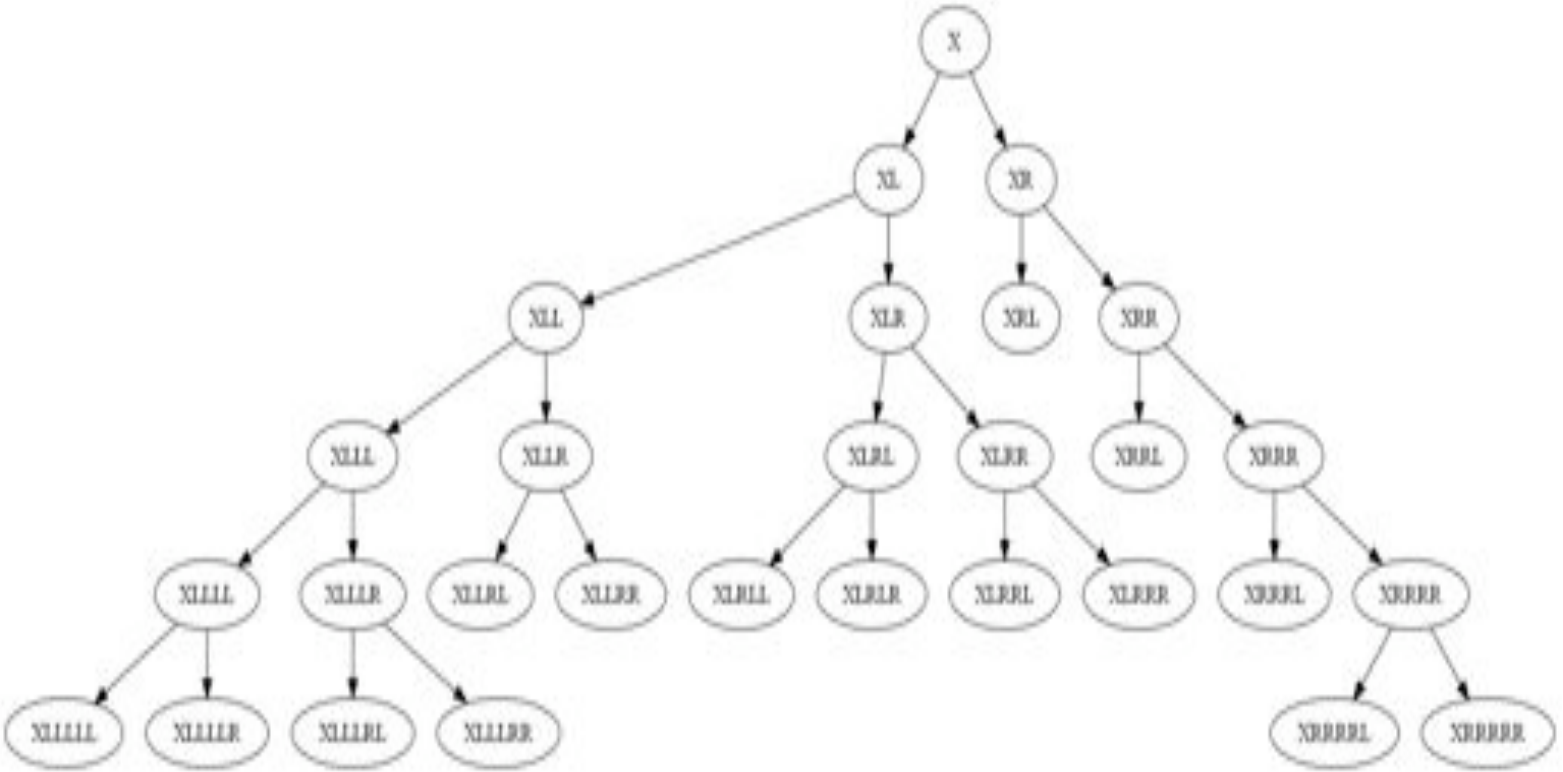
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



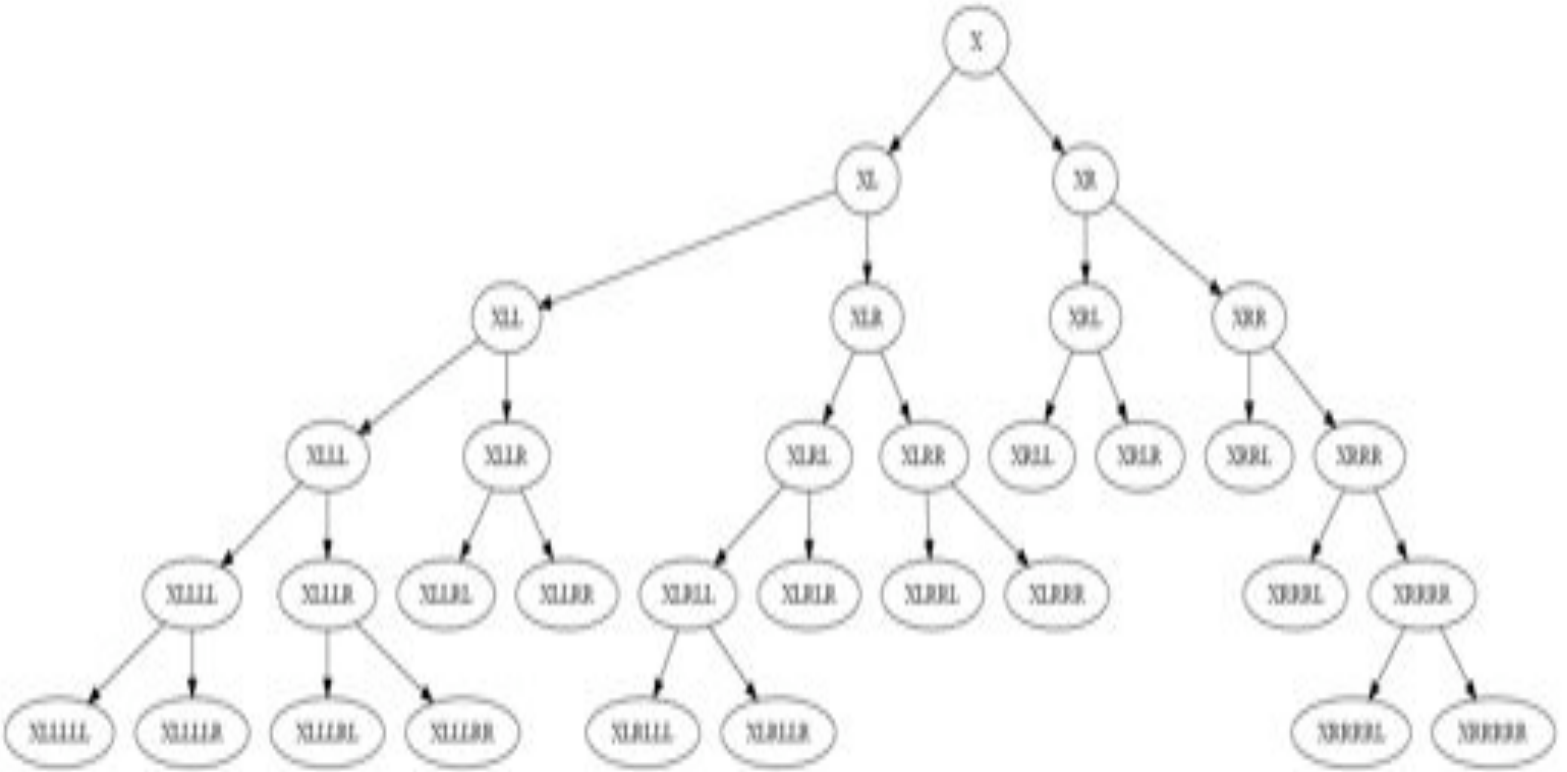
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



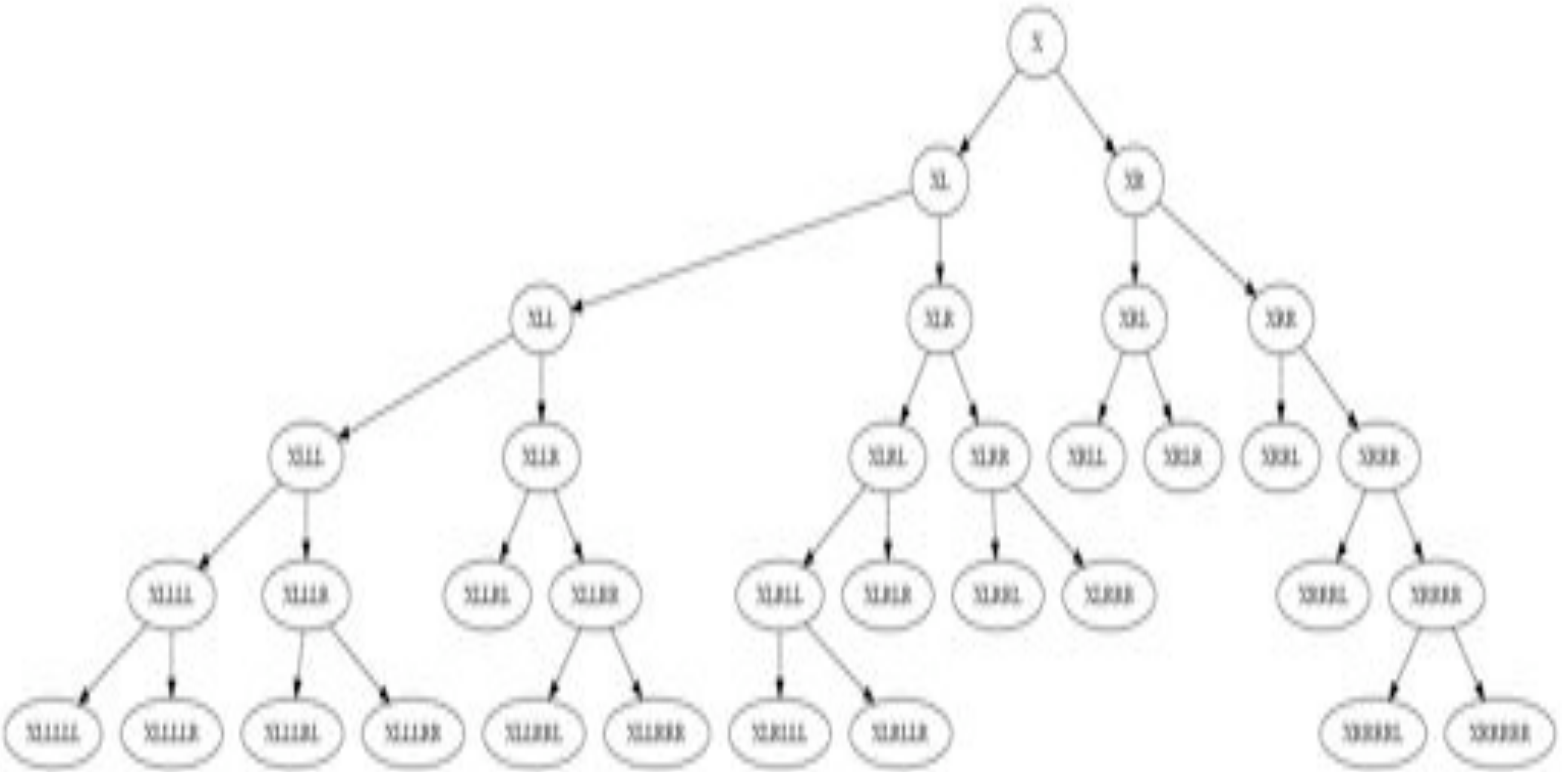
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



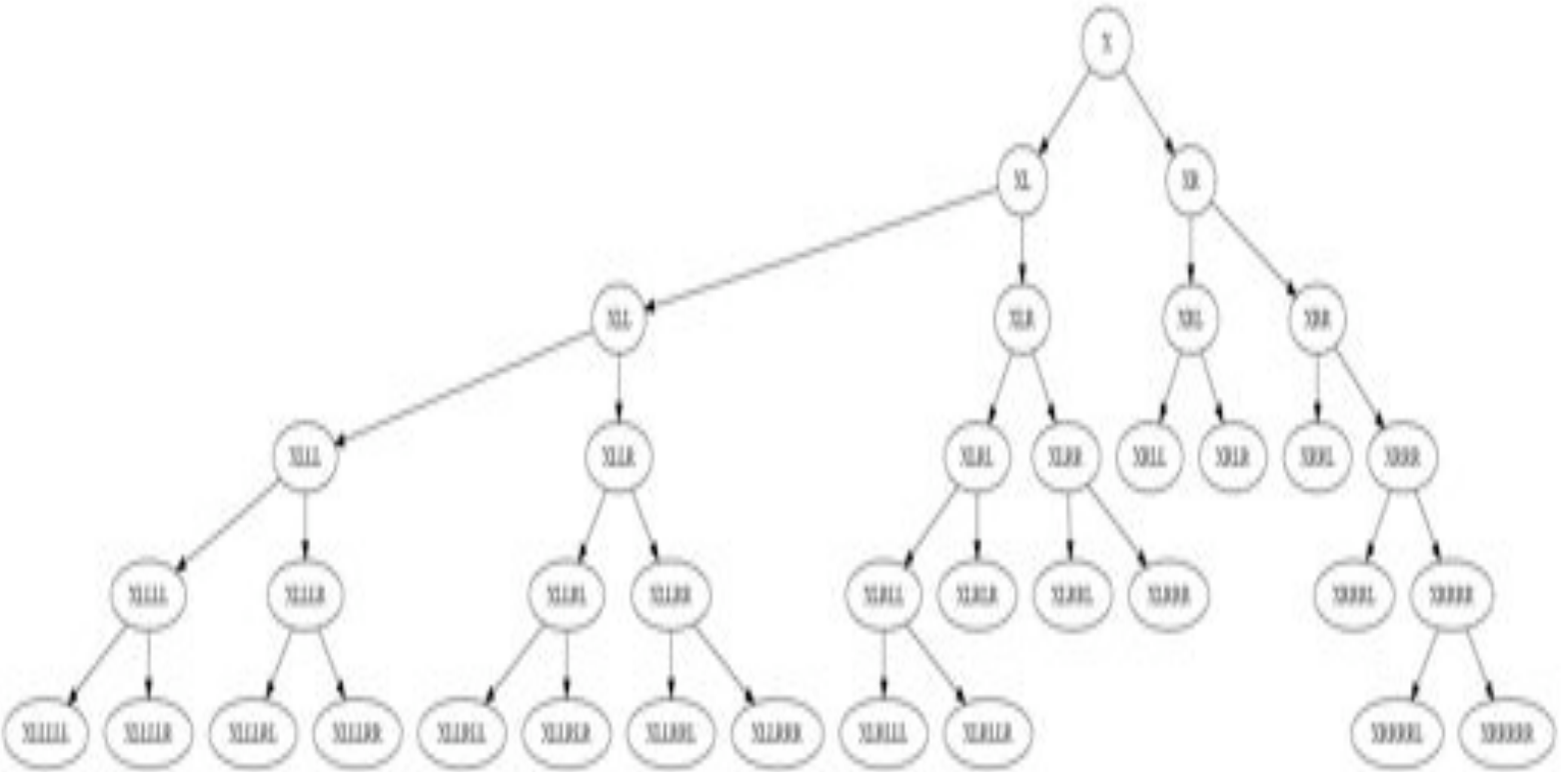
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



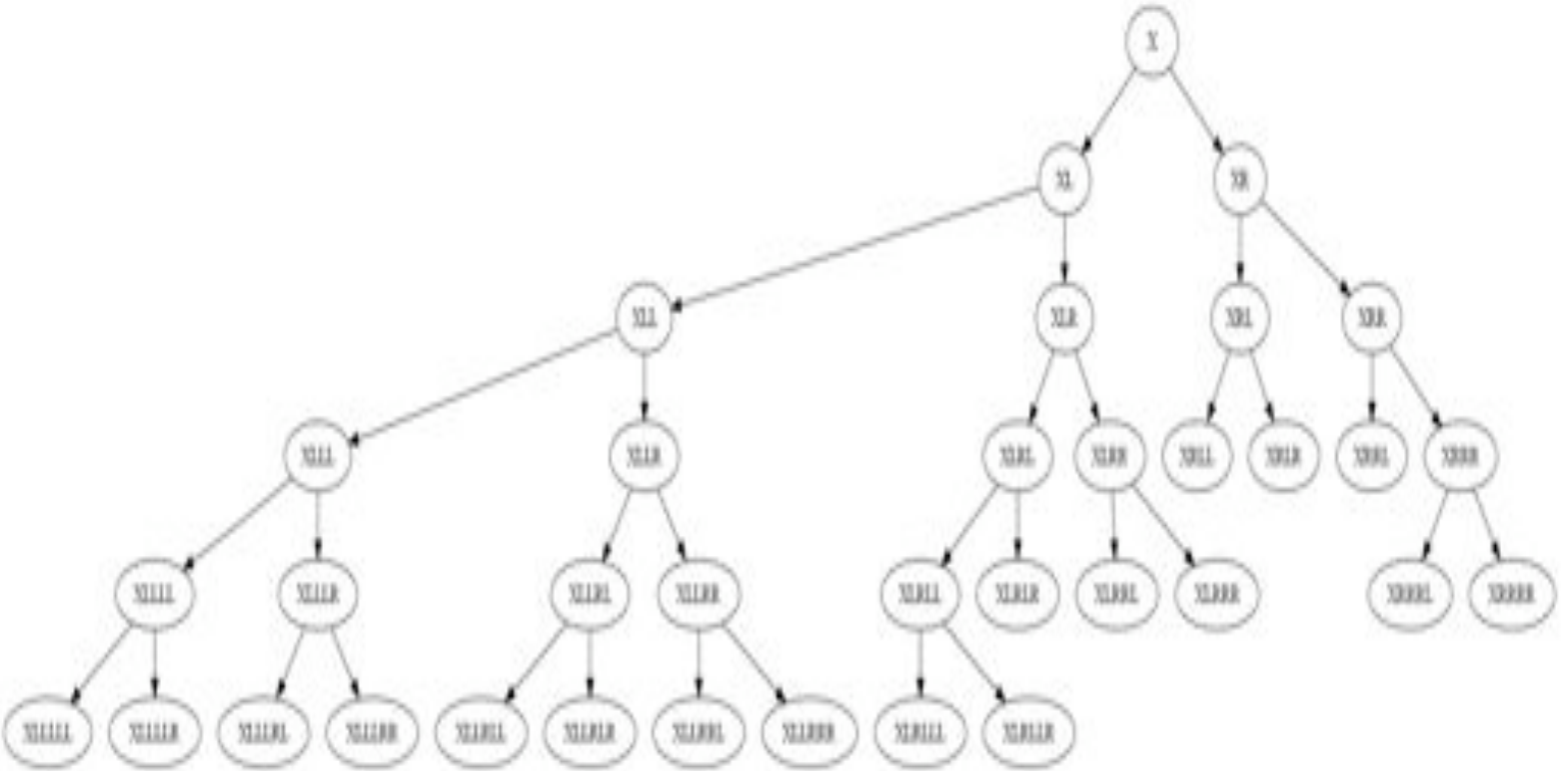
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



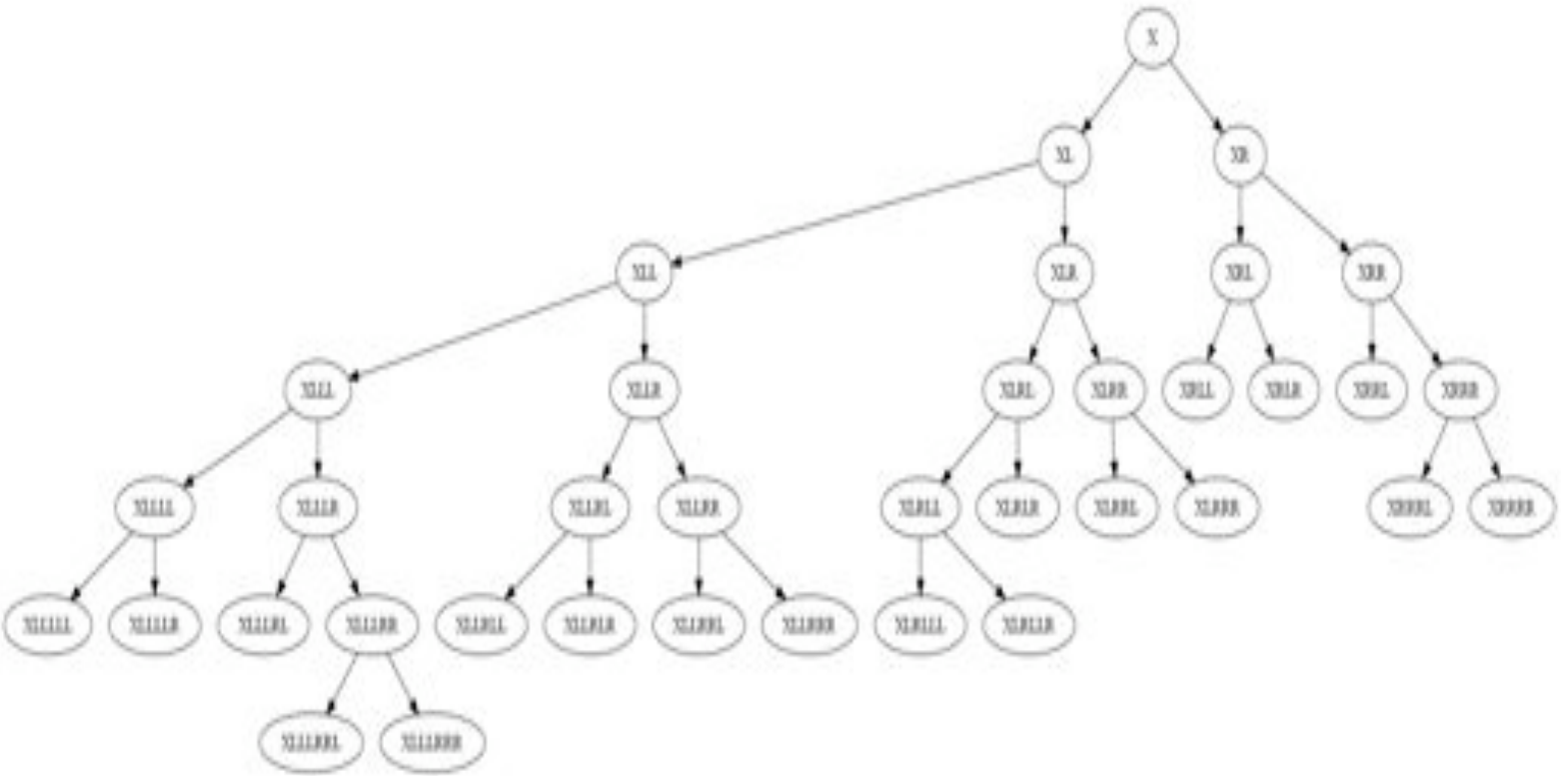
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



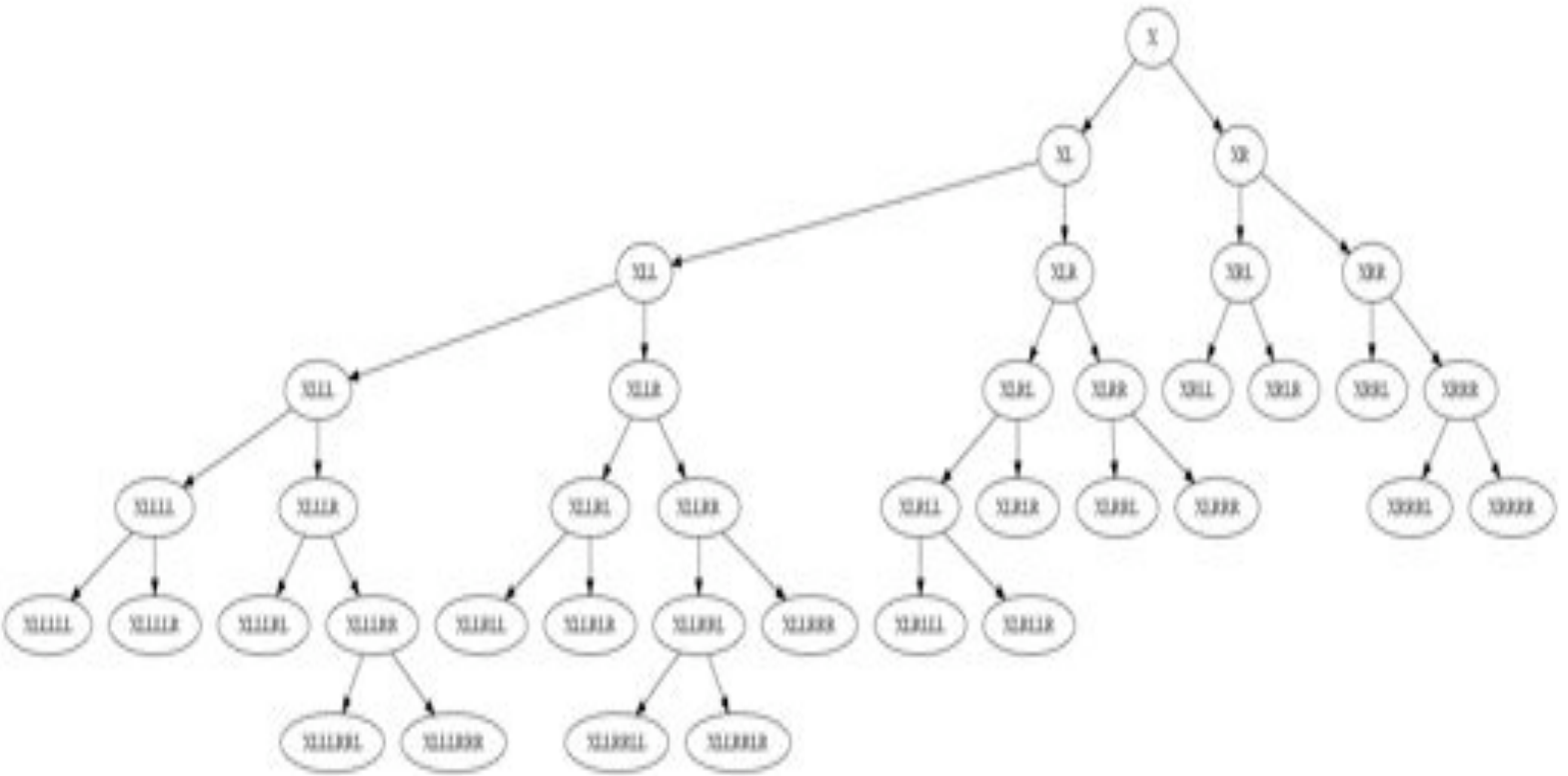
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



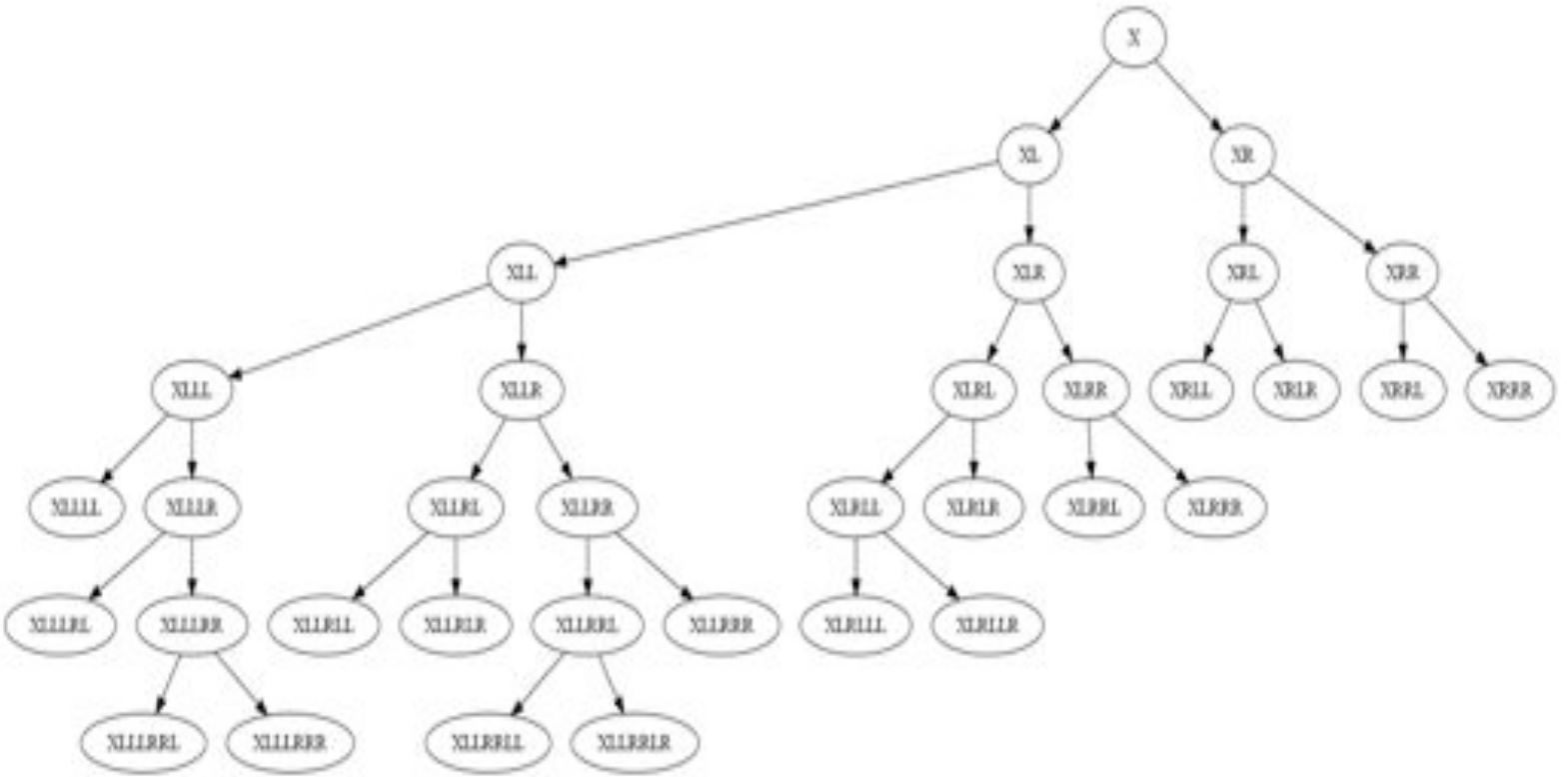
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



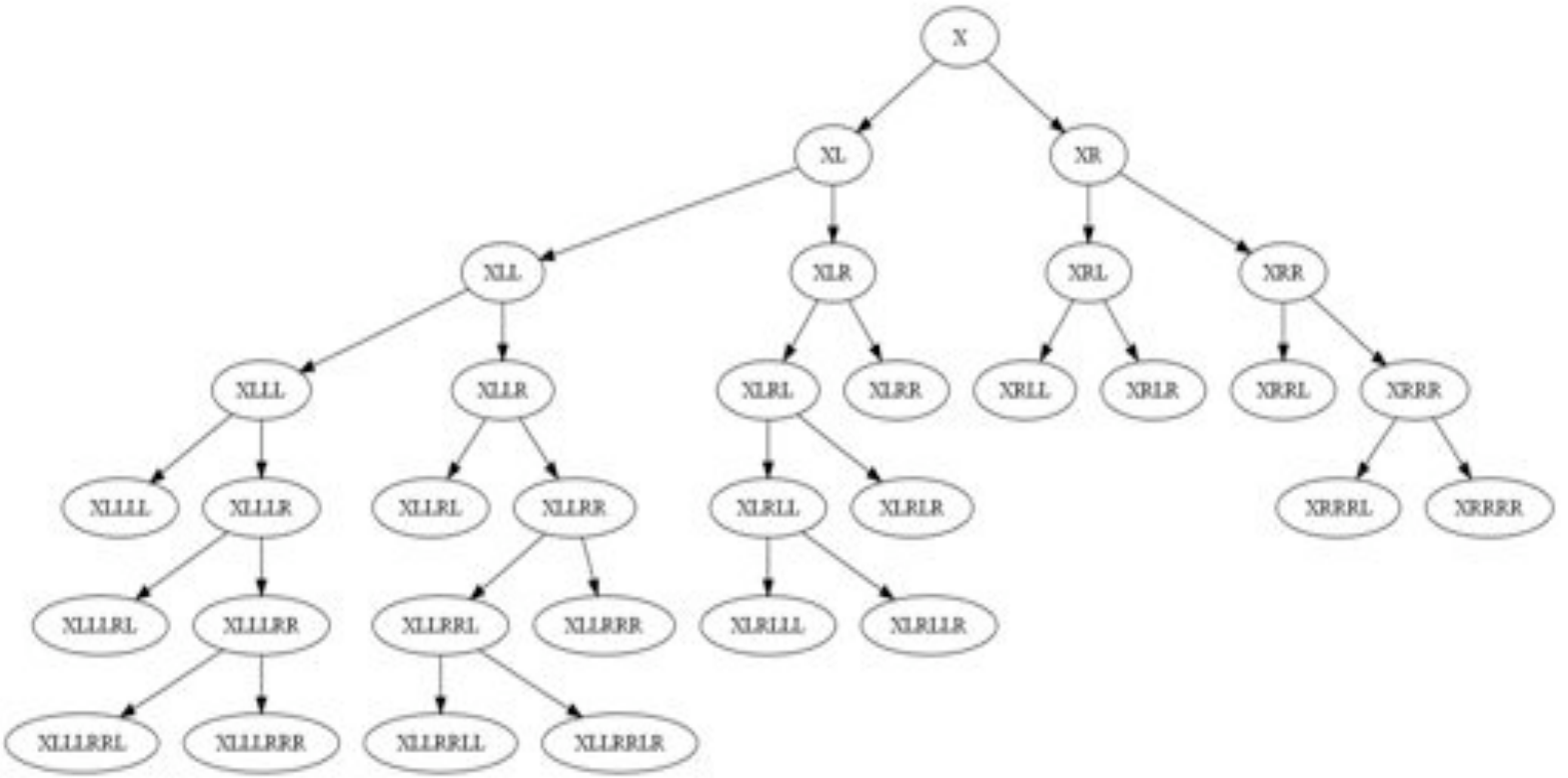
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



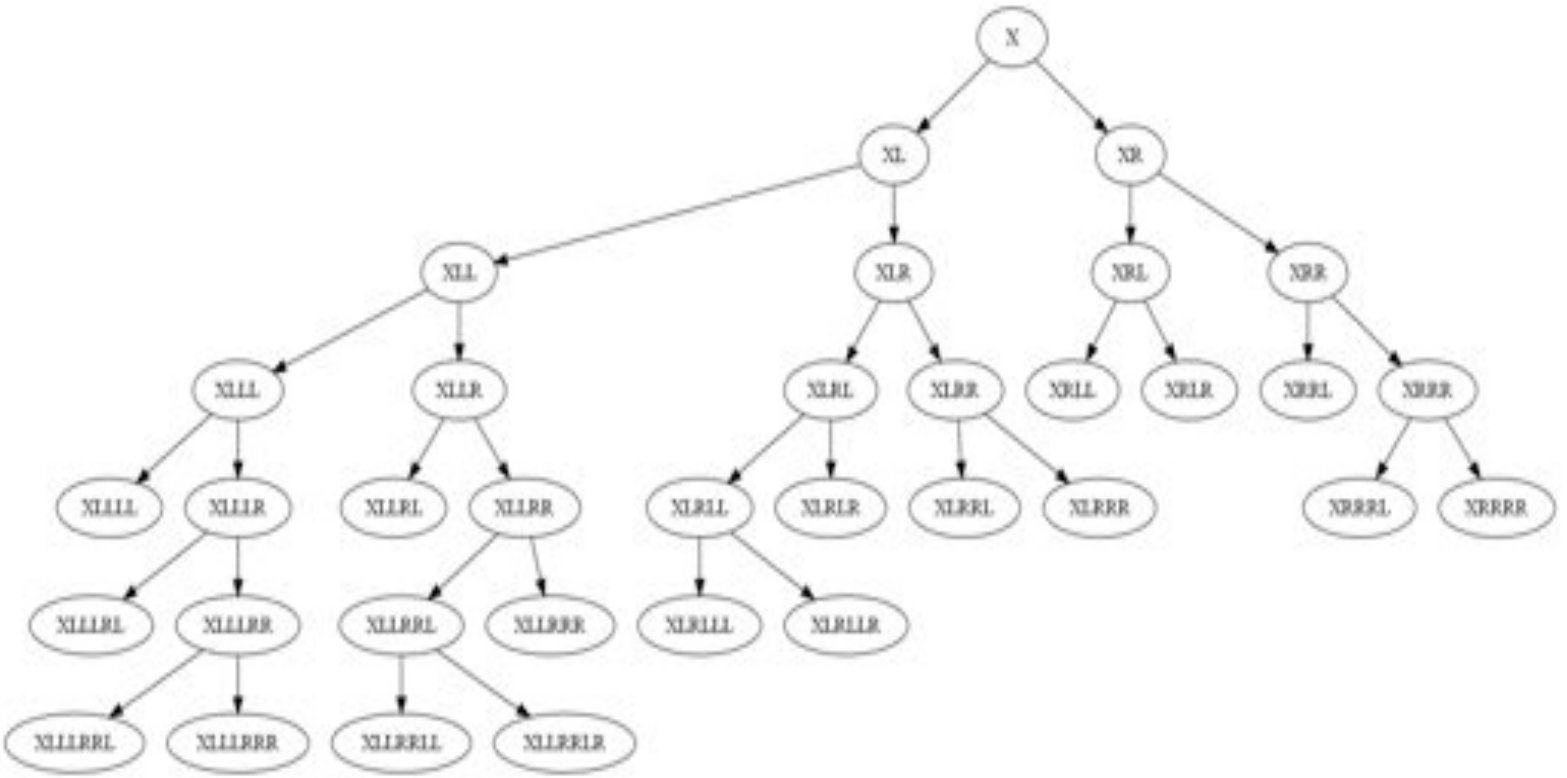
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



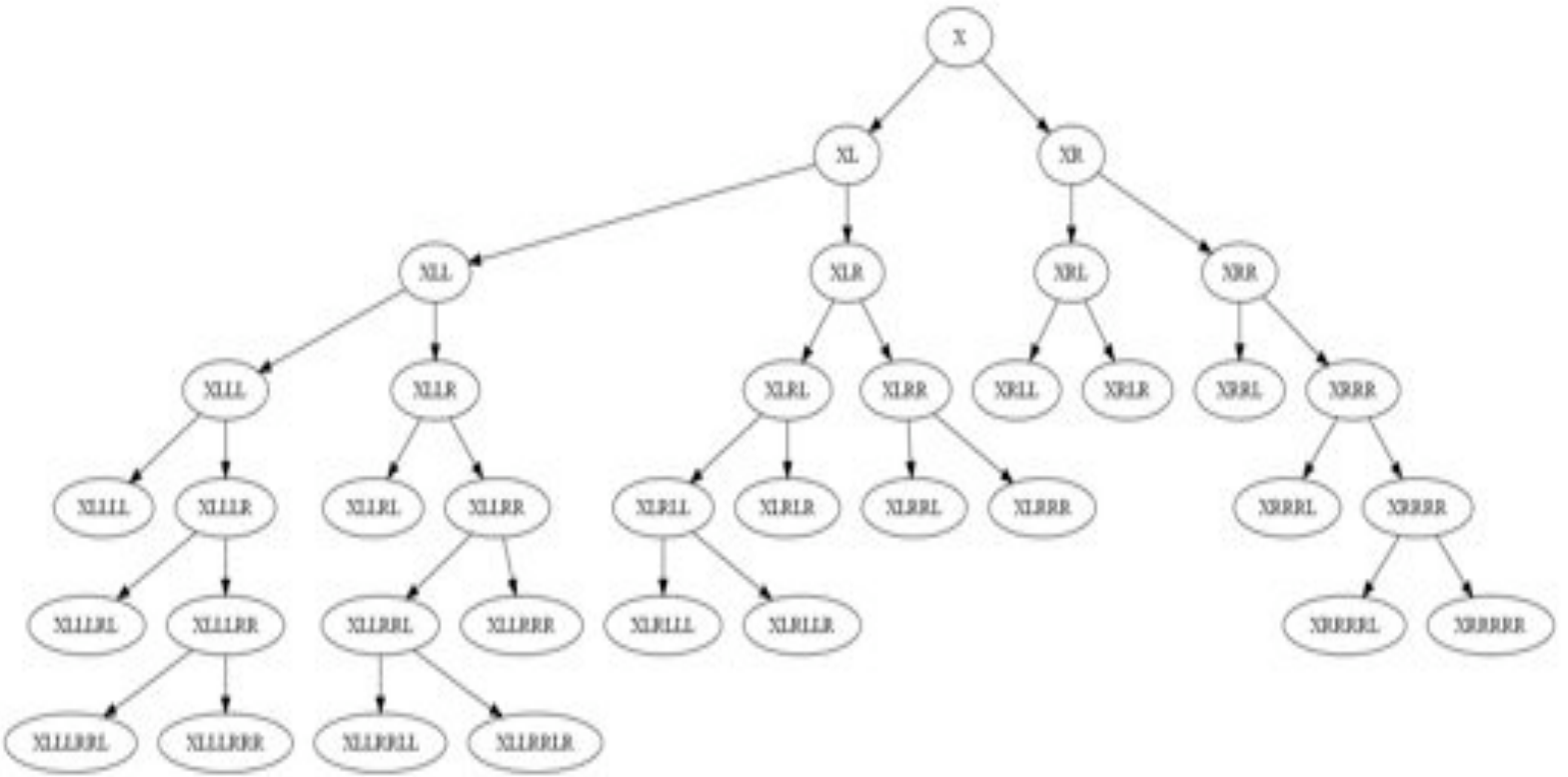
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



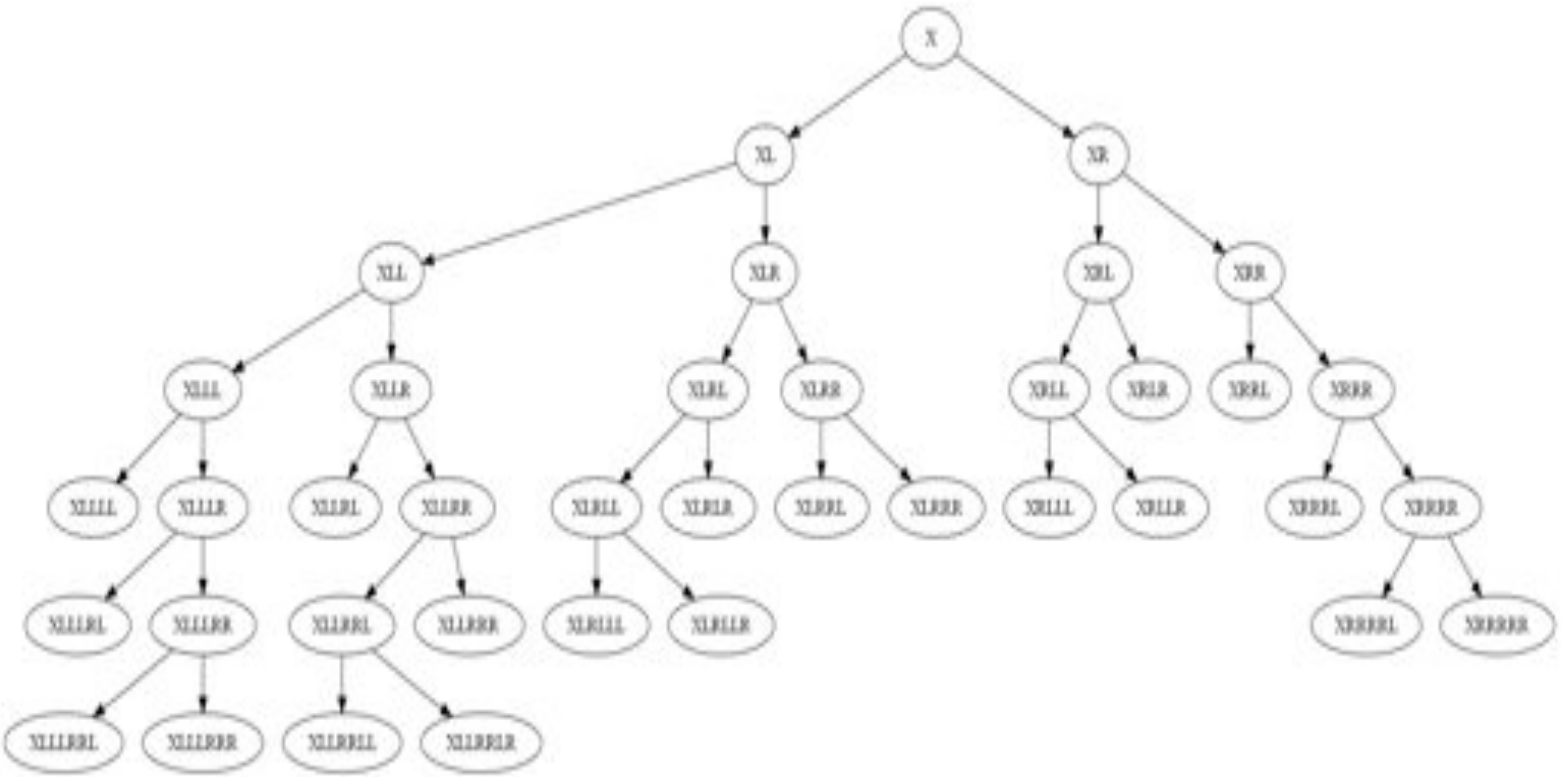
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



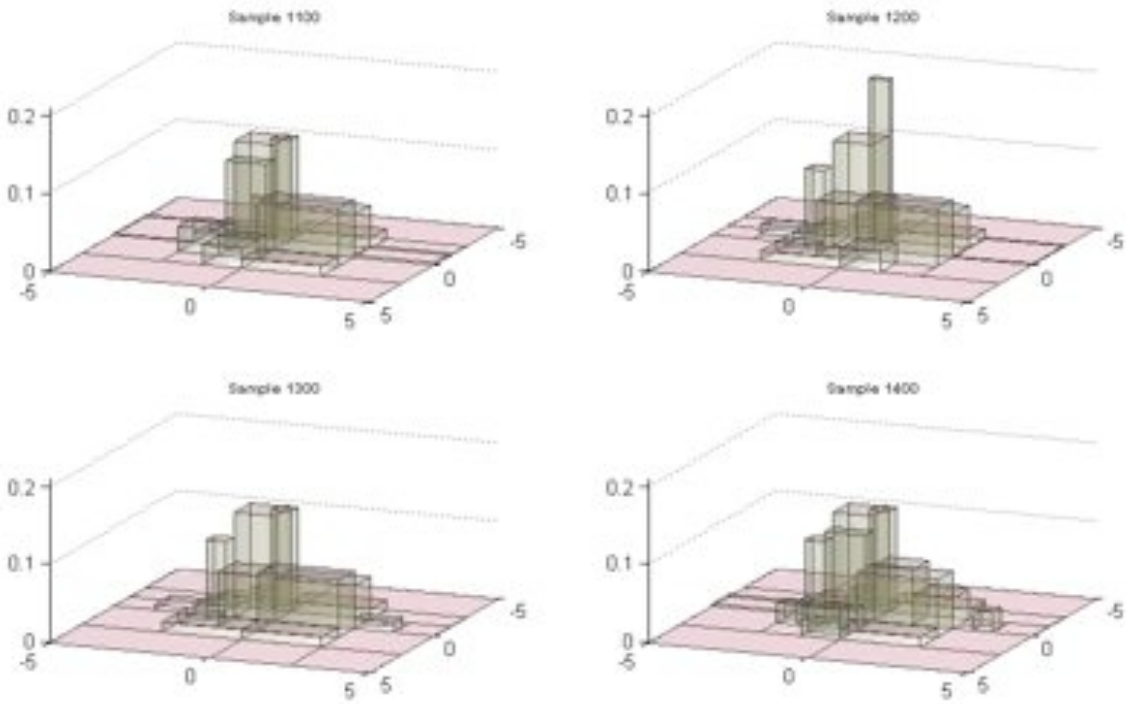
Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$



Monte Carlo Markov Chain over Histograms in $\mathbb{S}_{0:\infty}$

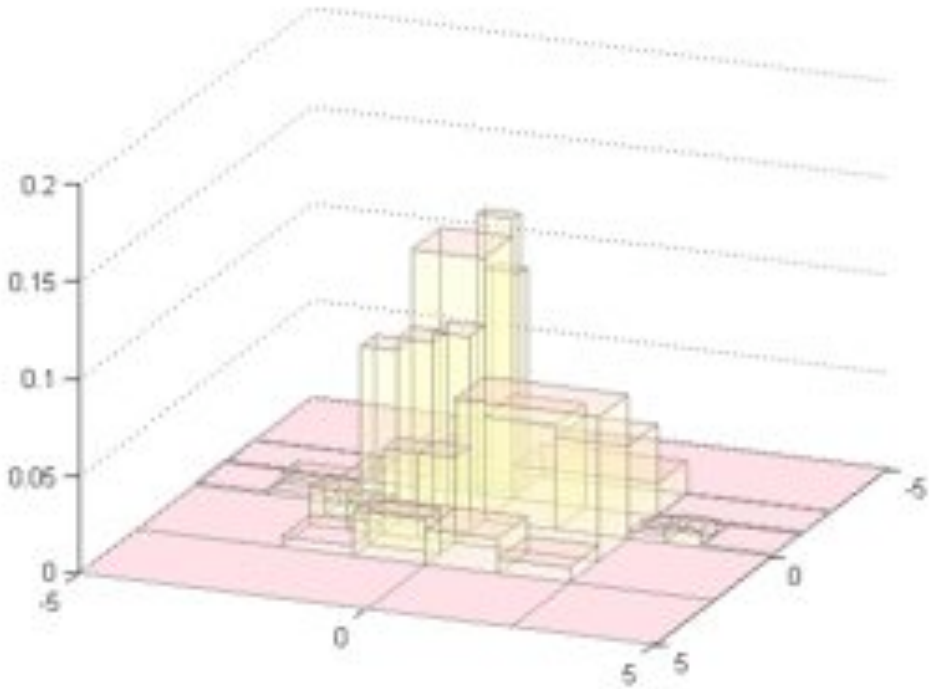


Histogram Estimates - Standard Bivariate Gaussian



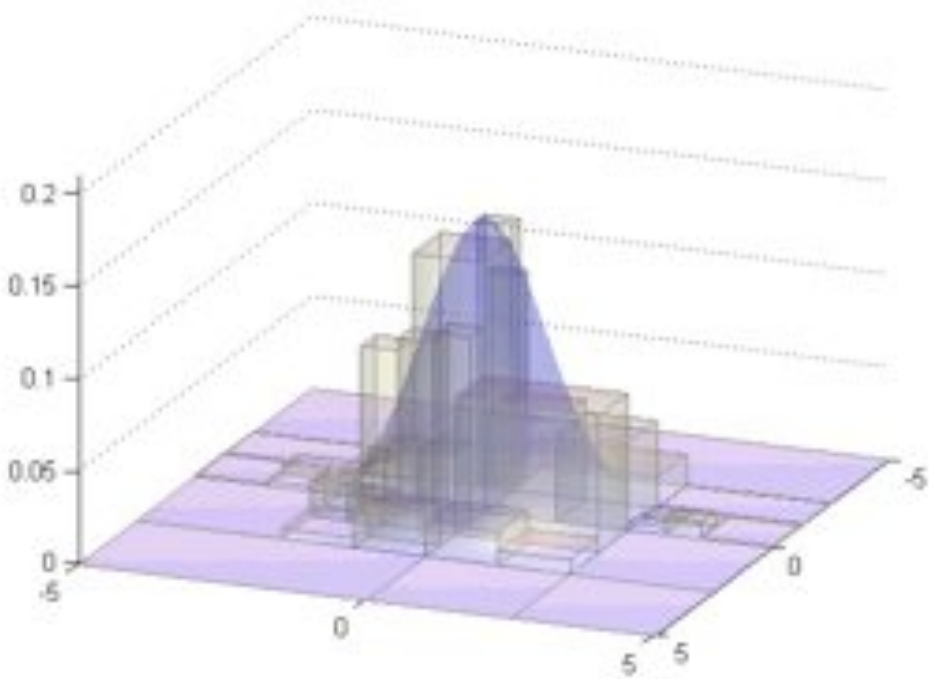
Four sample histograms

Histogram Estimates - Standard Bivariate Gaussian



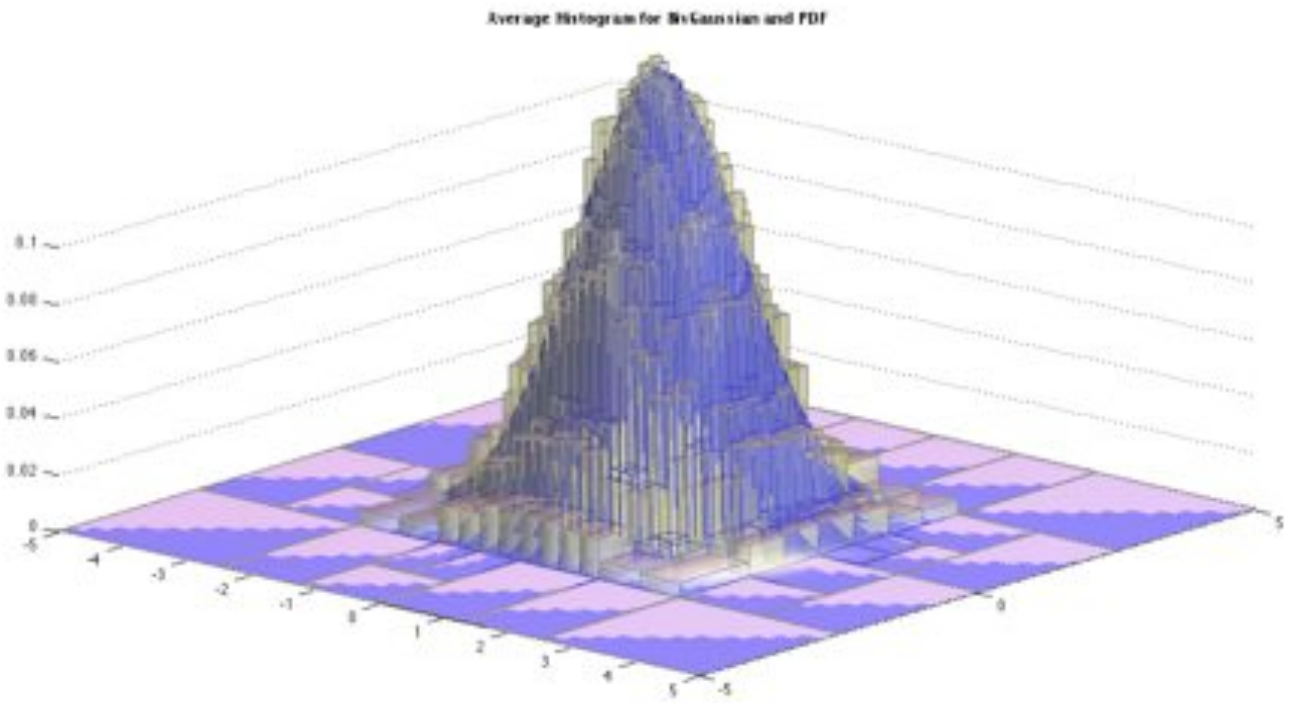
Average of the four sampled histograms

Histogram Estimates - Standard Bivariate Gaussian



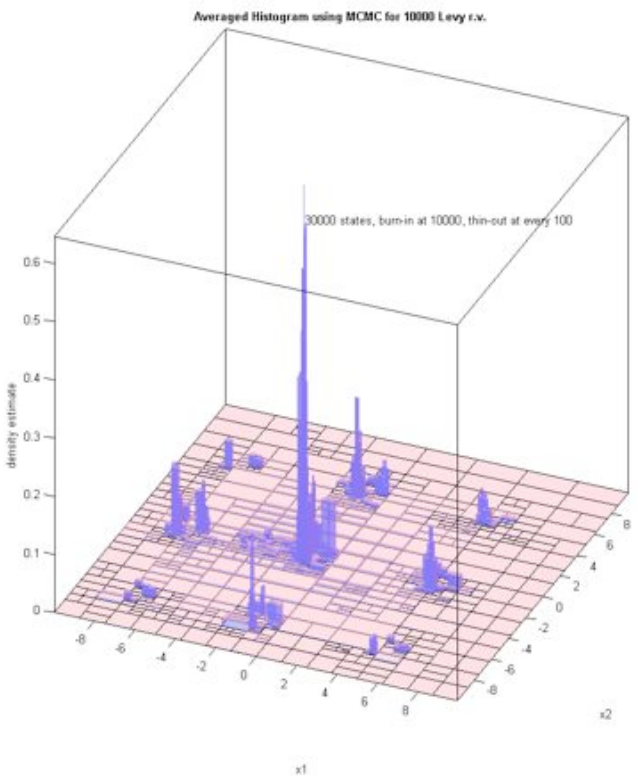
Average of the four sampled histograms with Gaussian PDF

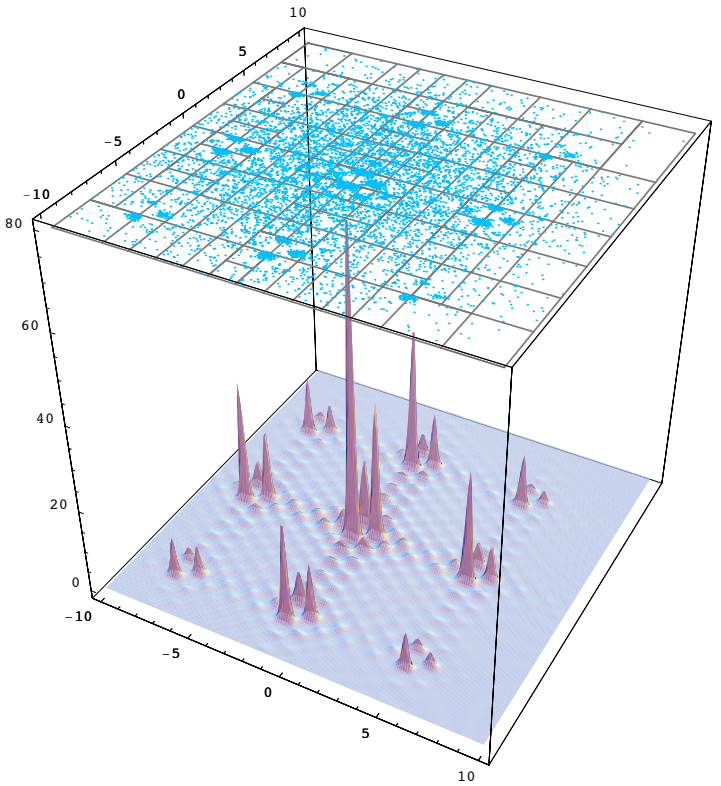
Histogram Estimates - Standard Bivariate Gaussian



A much better estimate

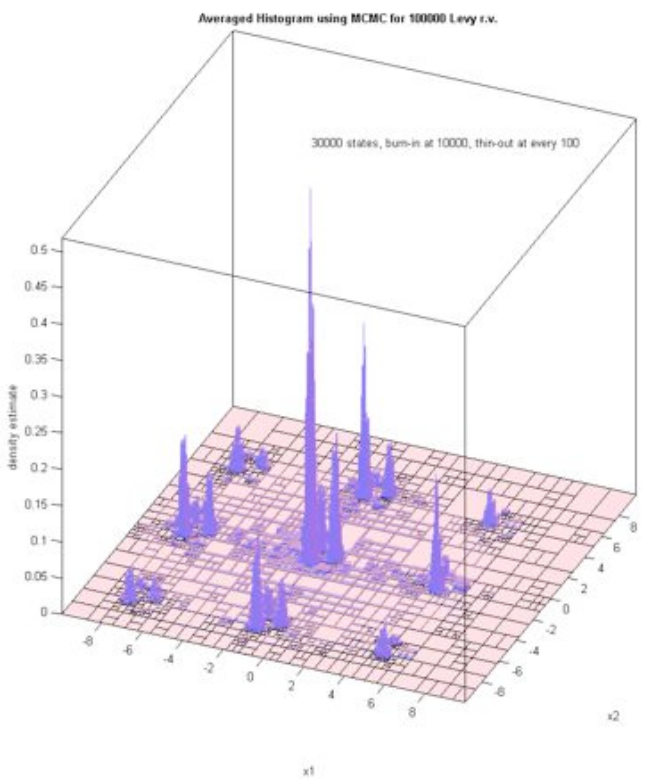
Histogram Estimates - Bivariate Levy Density

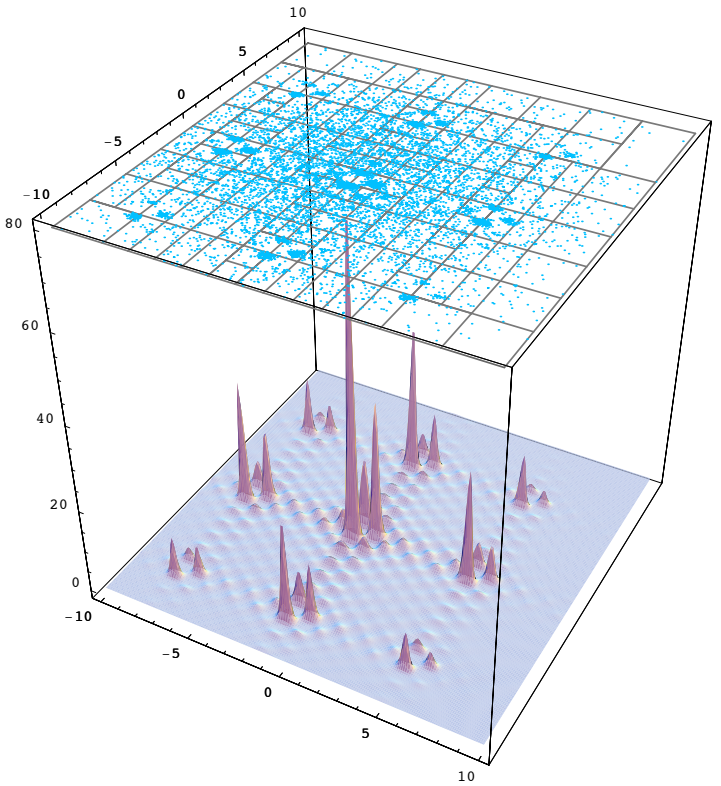




Data points = 10000, Number of states = 30000, Burn-in = 10000, Thin-out = 100, Averaged over 201 states, Time taken = 14.16s

Histogram Estimates - Bivariate Levy Density

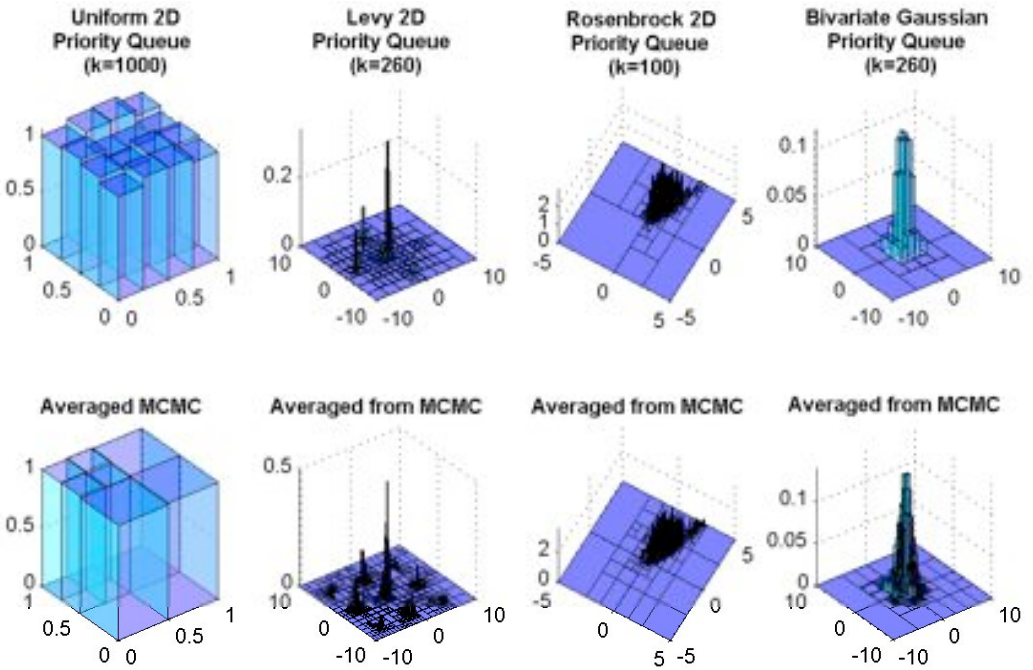




Data points = 100000, Number of states = 30000, Burn-in = 10000, Thin-out = 100, Averaged over 201 states, Time taken = 50.59s

Simulations for MCMC and `SplitMostCounts` PQ

Density	Time (s)	MIAE (root box)	MIAE (PQ)	Density	Time (s)	MIAE (root box)	MIAE (PQ)
U(0,1) 1D	5.2940	0.0112	0.0115	U(0,1) 2D	4.96	0.0125	0.0123
N(0,1) 1D	0.4857	0.0663	0.0651	Rosen. 10D	2.2900	NA	NA
Gaussian 2D	0.6206	0.2444	0.2702	U(0,1) 10D	14.775	0.0127	0.0119
Levy 2D	4.3200	0.4187	0.3272	U(0,1) 100D	107.3963	0.0108	0.0116
Rosen. 2D	9.5672	0.3273	0.4307	U(0,1) 1000D	970.4471	0.0117	0.0108



Examples of Application

- Web Log Data (Link to SAGE server)
- Air Traffic Data (Link to SAGE server)
- Earthquake Data (Link to SAGE server)

Conclusions

- Statistical Regular Sub-paving (SRSP) is a sufficient statistical data-structure for density estimation and many decisions in massive IID experiments.

Conclusions

- Statistical Regular Sub-paving (SRSP) is a sufficient statistical data-structure for density estimation and many decisions in massive IID experiments.
- We can quickly grow or prune the SRSP tree data-adaptively

Conclusions

- Statistical Regular Sub-paving (SRSP) is a sufficient statistical data-structure for density estimation and many decisions in massive IID experiments.
- We can quickly grow or prune the SRSP tree data-adaptively
- Arithmetic can be efficiently extended to SRSPs - averaging histograms and other *arithmetics* on $\mathbb{S}_{0:\infty}$.

Conclusions

- Statistical Regular Sub-paving (SRSP) is a sufficient statistical data-structure for density estimation and many decisions in massive IID experiments.
- We can quickly grow or prune the SRSP tree data-adaptively
- Arithmetic can be efficiently extended to SRSPs - averaging histograms and other *arithmetics* on $\mathbb{S}_{0:\infty}$.
- Smoother posterior mean from MCMC samples on the space of adaptive multi-variate histograms with partitions in $\mathbb{S}_{0:\infty}$. NFL: MCMC convergence issues exist!

Conclusions

- Statistical Regular Sub-paving (SRSP) is a sufficient statistical data-structure for density estimation and many decisions in massive IID experiments.
- We can quickly grow or prune the SRSP tree data-adaptively
- Arithmetic can be efficiently extended to SRSPs - averaging histograms and other *arithmetics* on $\mathbb{S}_{0:\infty}$.
- Smoother posterior mean from MCMC samples on the space of adaptive multi-variate histograms with partitions in $\mathbb{S}_{0:\infty}$. NFL: MCMC convergence issues exist!
- Higher (1000) dimensional densities can be estimated fast and rough (but L_1-consistent) with the approach especially with `SplitMostCounts` PQ and further decisions can be done with appropriate *mapped RSP arithmetic* over $\mathbb{S}_{0:\infty}$.

References

Jaulin, L., Kieffer, M., Didrit, O. & Walter, E. (2001). *Applied interval analysis*. London: Springer-Verlag.

Lugosi, G. and Nobel, A. (1996). Consistency of data-driven histogram methods for density estimation and classification. *The Annals of Statistics* **24** 687–706.

Sainudiin, R. and York, T. L. (2005). *An Auto-validating Rejection Sampler*. BSCB Dept. Technical Report BU-1661-M, Cornell University, Ithaca, New York.

Vapnik, V. N. and Chervonenkis, A. Y. (1971). On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.* **16** 264–280.