# Statistical Regular Pavings for Bayesian Nonparametric Density Estimation

**Raaz**esh Sainudiin

joint work with: Jennifer Harlow, Dominic Lee, Carey Priebe, Gloria Teng and Warwick Tucker

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

September 24, 2014,

Department of Statistical Sciences Seminar, Cornell University, Ithaca, NY, USA

Section 1

# Non-parametric Multi-dimensional Density Estimation

# Massive Metric Data Streams – Introduction

- A massive metric data stream is:

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

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$
- Most estimators of $f$ grind to a halt on such data streams

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$
- Most estimators of $f$ grind to a halt on such data streams
- Need a multi-dimensional metric data-structure that is:

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$
- Most estimators of $f$ grind to a halt on such data streams
- Need a multi-dimensional metric data-structure that is:
  1. Computationally Efficient

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$
- Most estimators of $f$ grind to a halt on such data streams
- Need a multi-dimensional metric data-structure that is:
  1. Computationally Efficient
  2. Statistically Consistent

# 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 Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$
- Most estimators of $f$ grind to a halt on such data streams
- Need a multi-dimensional metric data-structure that is:
  1. Computationally Efficient
  2. Statistically Consistent
  3. Data-adaptive

# Massive Metric Data Streams – Introduction

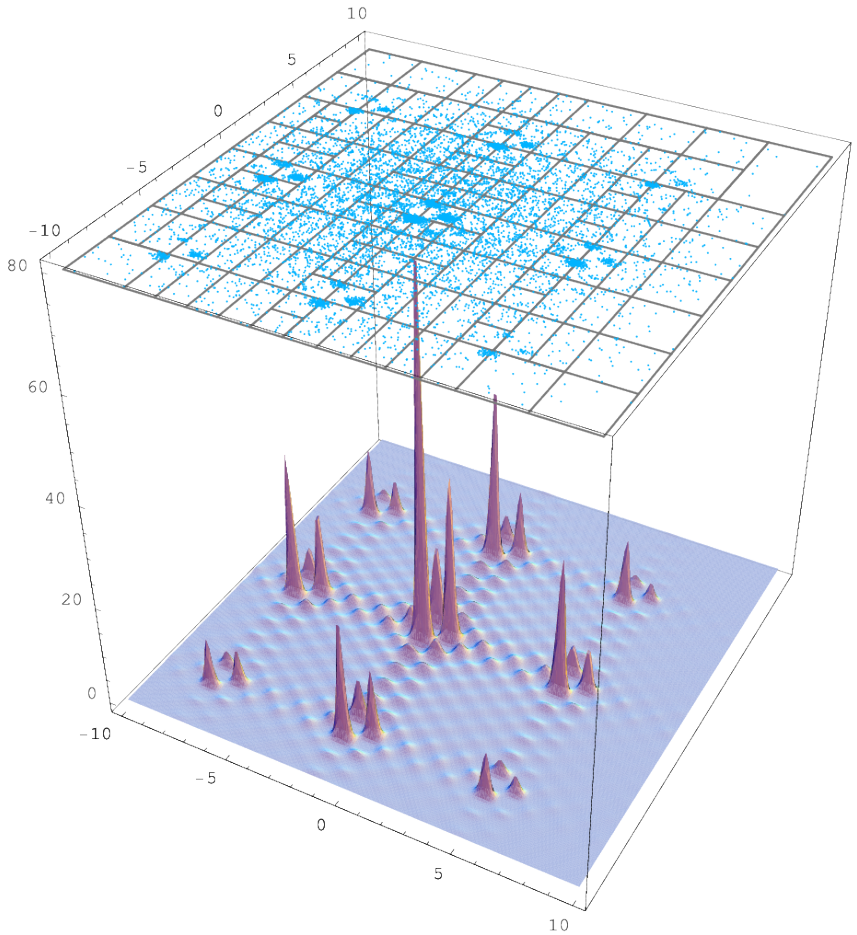
- A massive metric data stream is:

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

- Large Effective Dimension:
  $1 \leq d \leq 1000$ (unstructured $f$), $1 \leq d \leq 10$ (highly structured $f$)
- Huge Observations: $10^4 \leq n \leq 10^{10}$
- Most estimators of $f$ grind to a halt on such data streams
- Need a multi-dimensional metric data-structure that is:
  1. Computationally Efficient
  2. Statistically Consistent
  3. Data-adaptive
  4. Bayesian Non-parametric

# Non-parametric Density Estimation – Problem

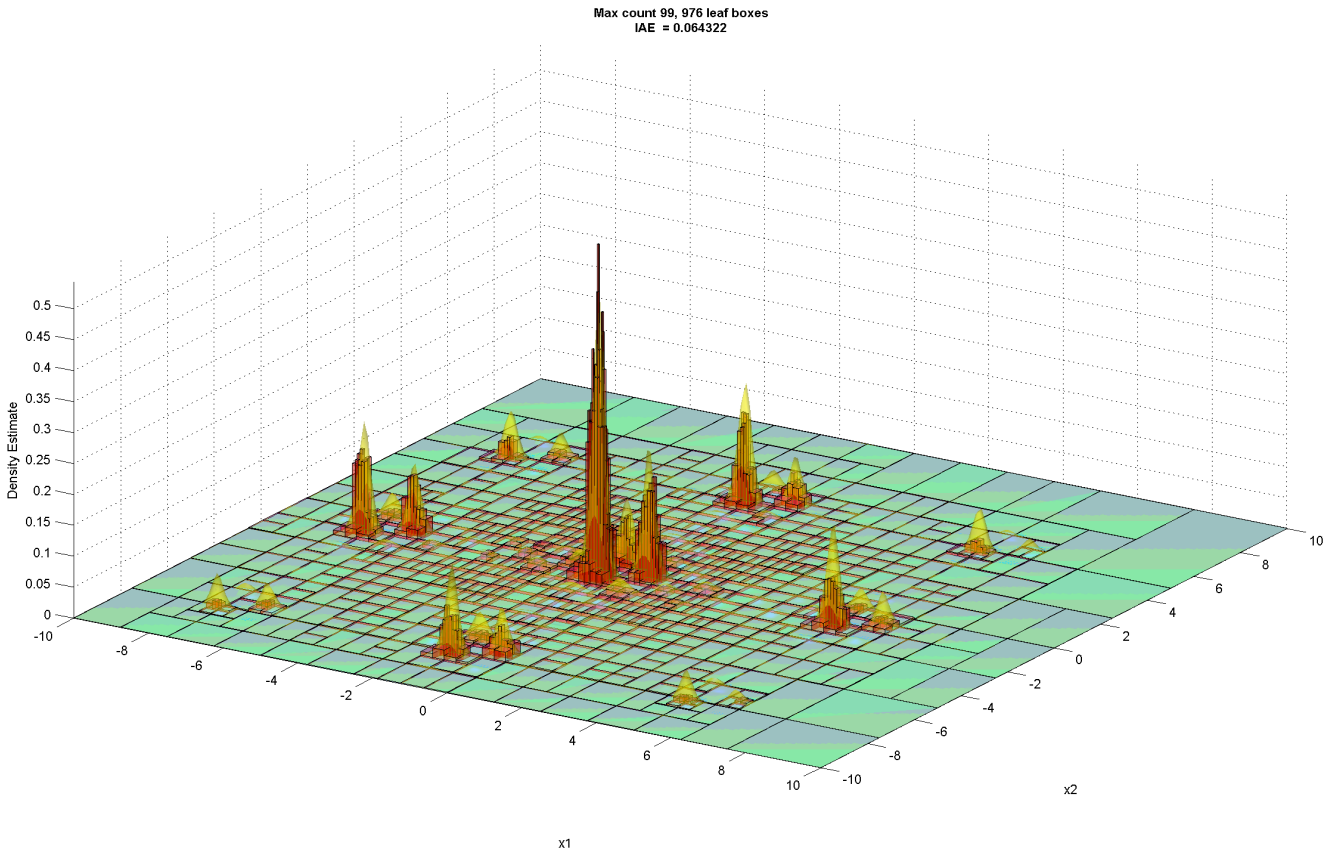
Take $X_1, X_2, \ldots, X_n$ IID samples from unknown density $f$




S & York, An auto-validating trans-dimensional universal rejection sampler for locally Lipschitz arithmetical expressions, Reliable Computing, 2013

# Non-parametric Density Estimation – Problem

and give a consistent estimator $f_n$ of $f$, i.e., $f_n : \left(\mathbb{R}^d\right)^n \times \mathbb{R}^d \to \mathbb{R}$




S, Teng, Harlow & Lee, Posterior Expectation of Regularly Paved Random Histograms, ACM Trans.on Model. and Comp. Simul., Special Issue on Monte Carlo Methods in Statistics, 2013

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)
4. Conditional Density Regression in Business Analytics (Industrial R&D Grant, Datamine Ltd., 2014/15)

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)
4. Conditional Density Regression in Business Analytics (Industrial R&D Grant, Datamine Ltd., 2014/15)
5. Anomaly Detection in Graph-valued Time Series

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)
4. Conditional Density Regression in Business Analytics (Industrial R&D Grant, Datamine Ltd., 2014/15)
5. Anomaly Detection in Graph-valued Time Series

But, even the fastest dual-tree algorithms are limited:

1. sample size $n < 10^6$ and

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)
4. Conditional Density Regression in Business Analytics (Industrial R&D Grant, Datamine Ltd., 2014/15)
5. Anomaly Detection in Graph-valued Time Series

But, even the fastest dual-tree algorithms are limited:

1. sample size $n < 10^6$ and
2. dimension $d < 10$ using space partitioning kd-trees and

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)
4. Conditional Density Regression in Business Analytics (Industrial R&D Grant, Datamine Ltd., 2014/15)
5. Anomaly Detection in Graph-valued Time Series

But, even the fastest dual-tree algorithms are limited:

1. sample size $n < 10^6$ and
2. dimension $d < 10$ using space partitioning kd-trees and
3. dimension $d < 200$ for structured data using ball trees

# Non-parametric Density Estimation – Ad/Dis-advantages

Non-parametric Density Estimation is a Fundamental Task in:

1. Classification (Air-traffic Management, 2012)
2. Self-exciting Point Process in Crime Risk Prediction (Industrial Contract, Wynyard Group, 2014)
3. Probablistic Collaborative Filtering (MBIE Capability Fellowships, adScale GmbH, 2013/14)
4. Conditional Density Regression in Business Analytics (Industrial R&D Grant, Datamine Ltd., 2014/15)
5. Anomaly Detection in Graph-valued Time Series

But, even the fastest dual-tree algorithms are limited:

1. sample size $n < 10^6$ and
2. dimension $d < 10$ using space partitioning kd-trees and
3. dimension $d < 200$ for structured data using ball trees
4. Existing tree-based algorithms are incapable of arithmetic

# The Fundamental Histogram Arithmetic Problem

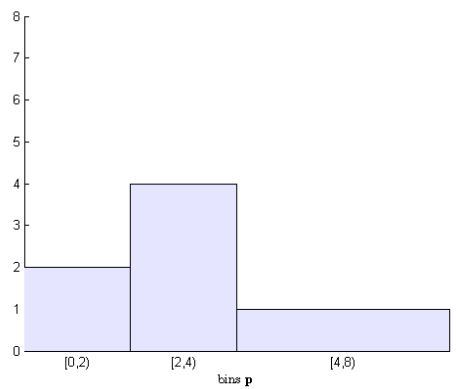
The Bayesian, non-parametric, $L_2$-loss minimizing, posterior mean estimate **needs** histogram arithmetic $(+, \cdot)$.

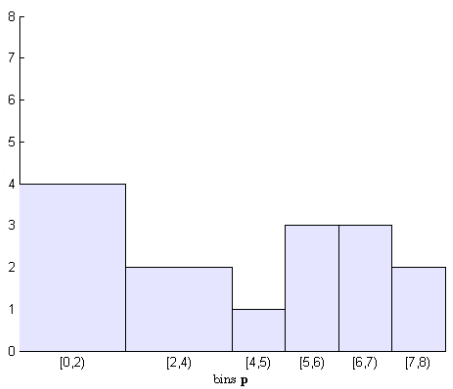
# The Fundamental Histogram Arithmetic Problem

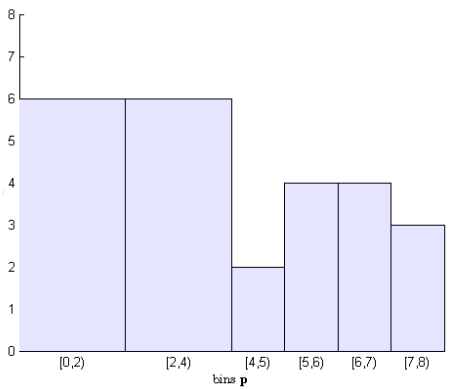
The Bayesian, non-parametric, $L_2$-loss minimizing, posterior mean estimate **needs** histogram arithmetic $(+, \cdot)$.
Arithmetic ?: How to average (add & scalar multiply by $1/2$) two histograms with different partitions in any dimension?

# The Fundamental Histogram Arithmetic Problem

The Bayesian, non-parametric, $L_2$-loss minimizing, posterior mean estimate **needs** histogram arithmetic $(+, \cdot)$.
Arithmetic ?: How to average (add & scalar multiply by $1/2$) two histograms with different partitions in any dimension?

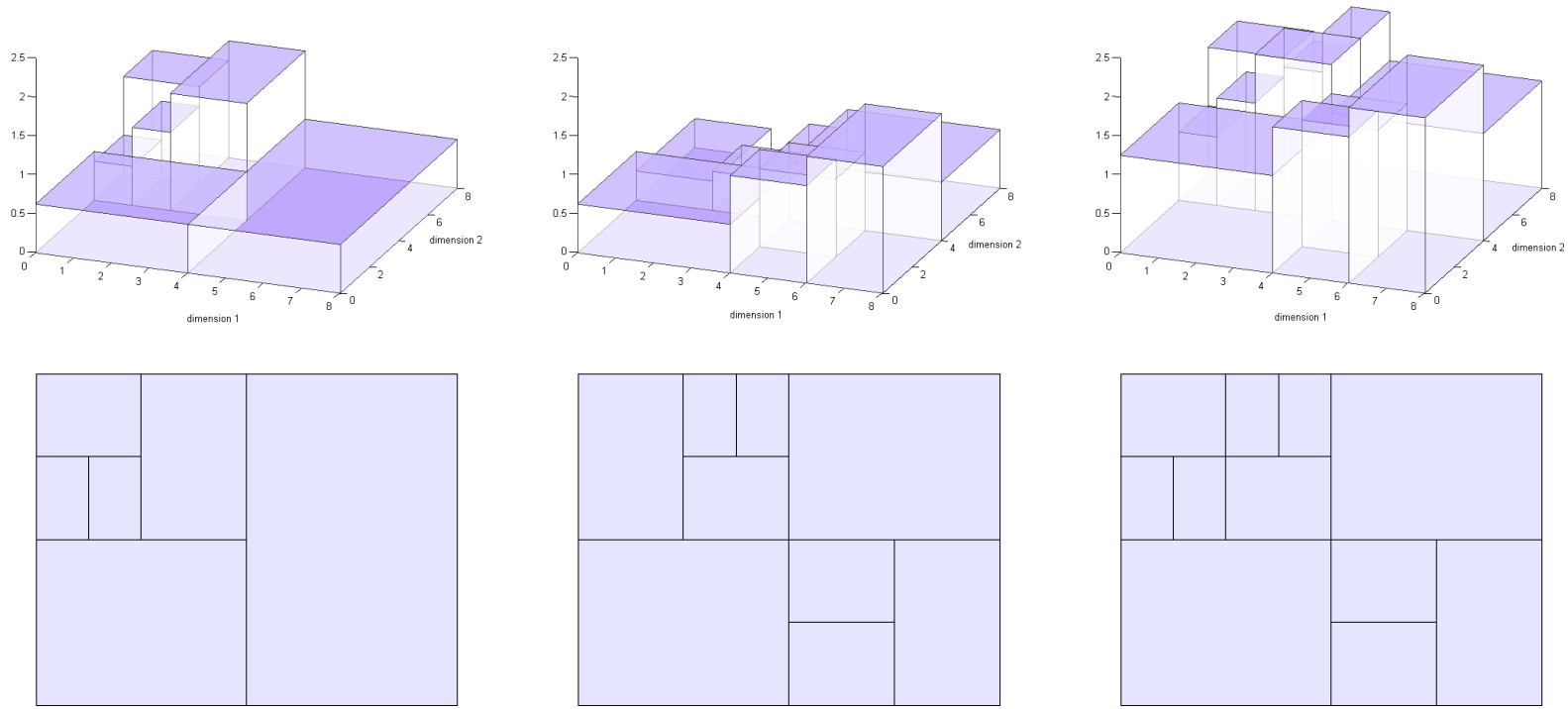


# The Fundamental Histogram Arithmetic Problem

The Bayesian, non-parametric, $L_2$-loss minimizing, posterior mean estimate **needs** histogram arithmetic $(+, \cdot)$.
Arithmetic ?: How to average (add & scalar multiply by $1/2$) two histograms with different partitions in any dimension?



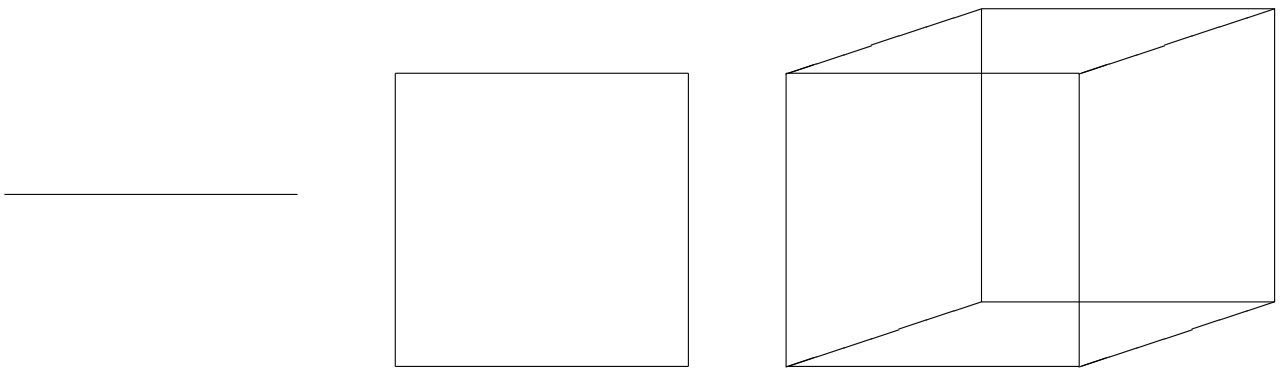
Section 2

# Arithmetic & Algebra of Rooted Recursive Plane Binary Trees (RRPBT)

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

*Intervals* and *Boxes* as interval vectors:

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



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

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

# An RP tree of a root box $\boldsymbol{x}_\rho \in \mathbb{IR}^d$

The regularly paved boxes of $\boldsymbol{x}_\rho$ can be represented by nodes of rooted recursive plane binary (RRPBT) trees or just regular paving (RP) trees

An operation of bisection on a box is equivalent to performing the operation on its corresponding node in the tree:

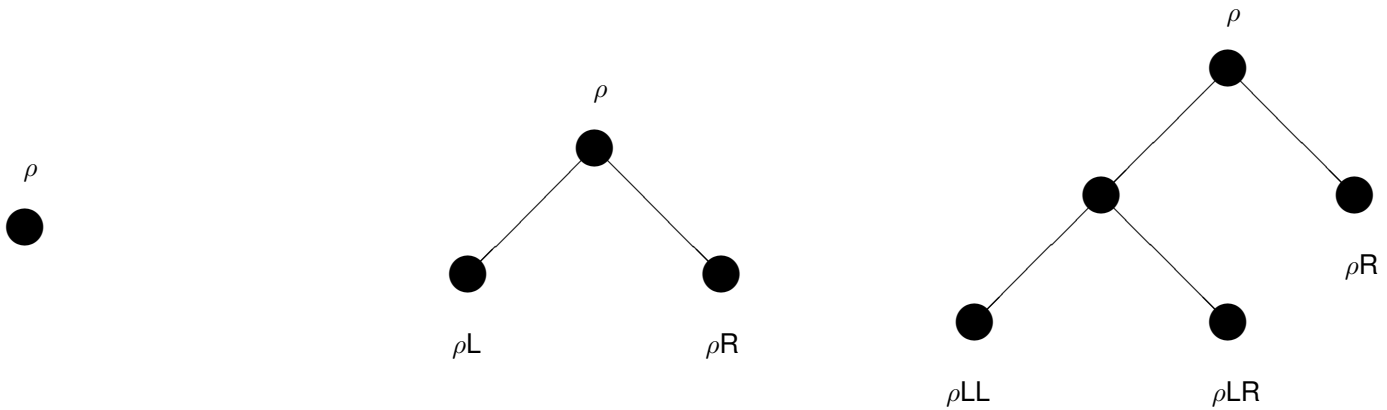
# An RP tree of a root box $\boldsymbol{x}_\rho \in \mathbb{IR}^d$

The regularly paved boxes of $\boldsymbol{x}_\rho$ can be represented by nodes of rooted recursive plane binary (RRPBT) trees or just regular paving (RP) trees

An operation of bisection on a box is equivalent to performing the operation on its corresponding node in the tree:

Leaf boxes of RP tree partition the root interval $\boldsymbol{x}_\rho \in \mathbb{IR}^1$

Bisect at the midpoint of the chosen leaf interval



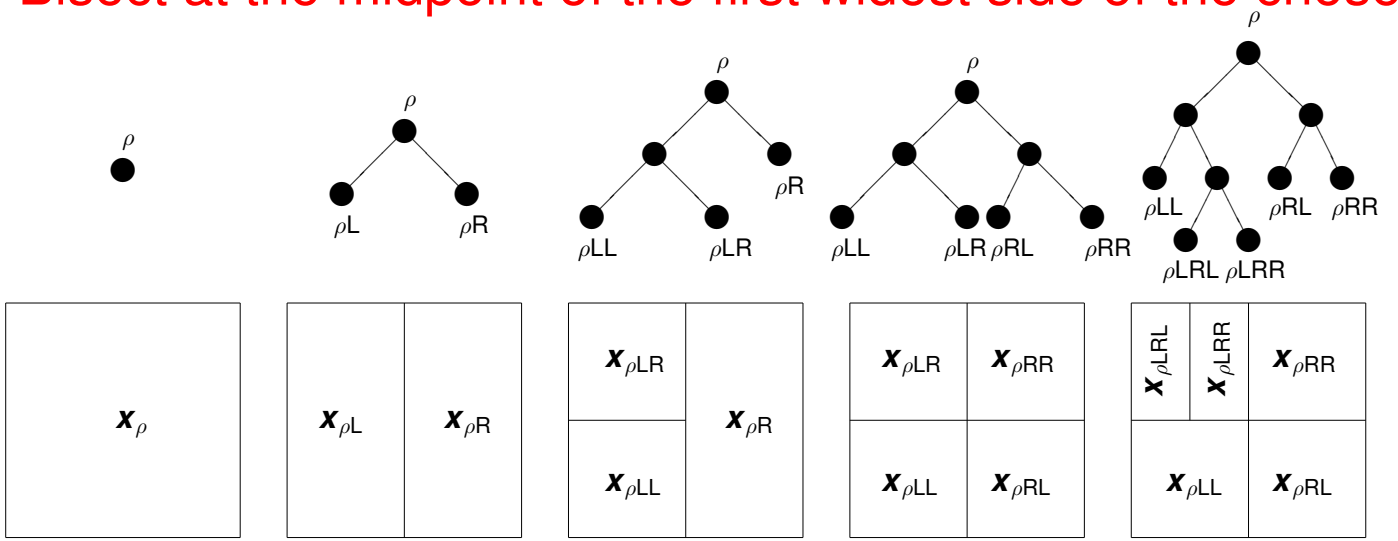
# An RP tree of a root box $\boldsymbol{x}_\rho \in \mathbb{IR}^d$

The regularly paved boxes of $\boldsymbol{x}_\rho$ can be represented by nodes of rooted recursive plane binary (RRPBT) trees or just regular paving (RP) trees

An operation of bisection on a box is equivalent to performing the operation on its corresponding node in the tree:

Leaf boxes of RP tree partition the root interval $\boldsymbol{x}_\rho \in \mathbb{IR}^2$

Bisect at the midpoint of the first widest side of the chosen leaf box



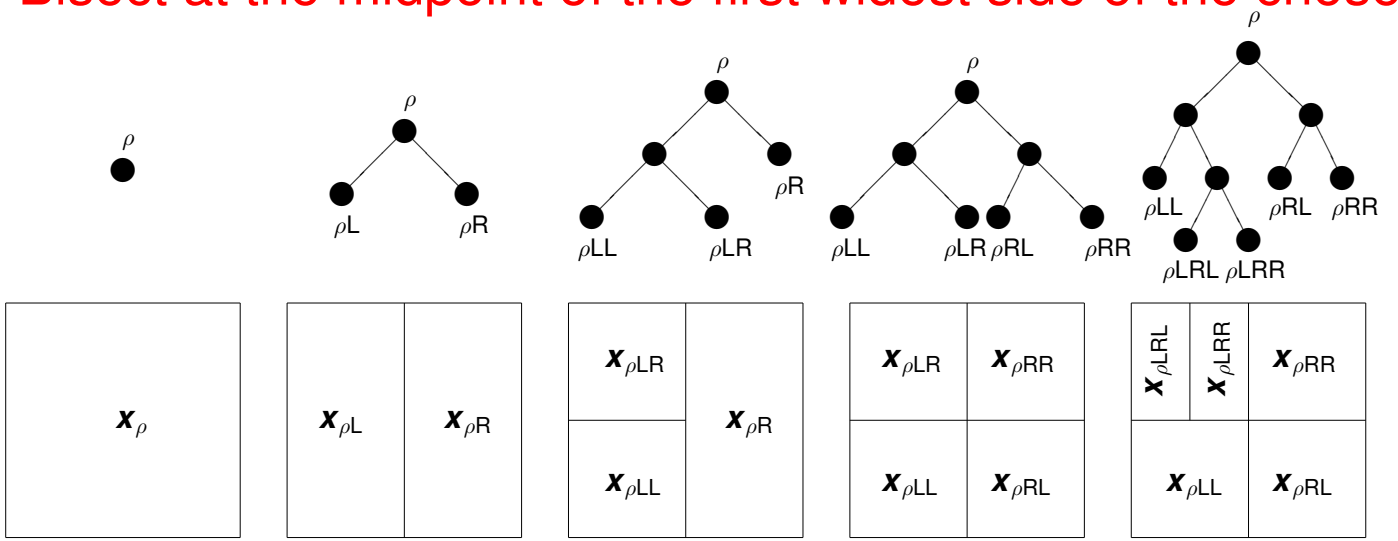
# An RP tree of a root box $\boldsymbol{x}_\rho \in \mathbb{IR}^d$

The regularly paved boxes of $\boldsymbol{x}_\rho$ can be represented by nodes of rooted recursive plane binary (RRPBT) trees or just regular paving (RP) trees

An operation of bisection on a box is equivalent to performing the operation on its corresponding node in the tree:

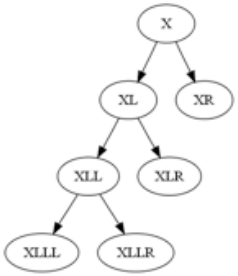
Leaf boxes of RP tree partition the root interval $\boldsymbol{x}_\rho \in \mathbb{IR}^2$

Bisect at the midpoint of the first widest side of the chosen leaf box

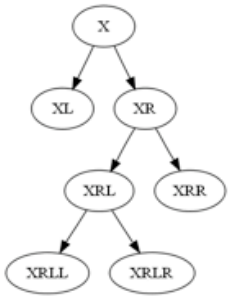


# Algebraic Structure and Combinatorics of RPs

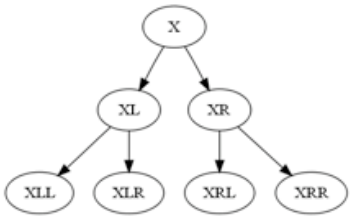
## Leaf-depth encoded RPs



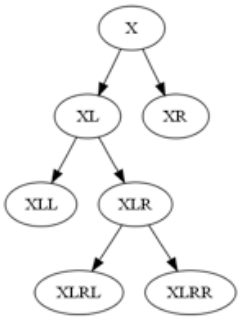

(3, 3, 2, 1)



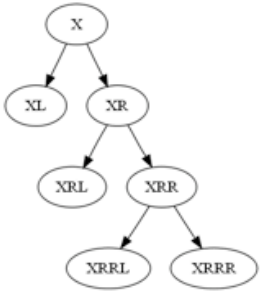

(1, 3, 3, 2)




(2, 2, 2, 2)

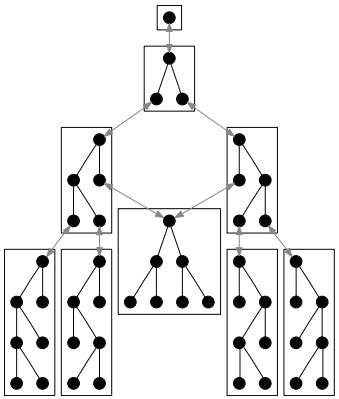



(2, 3, 3, 1)

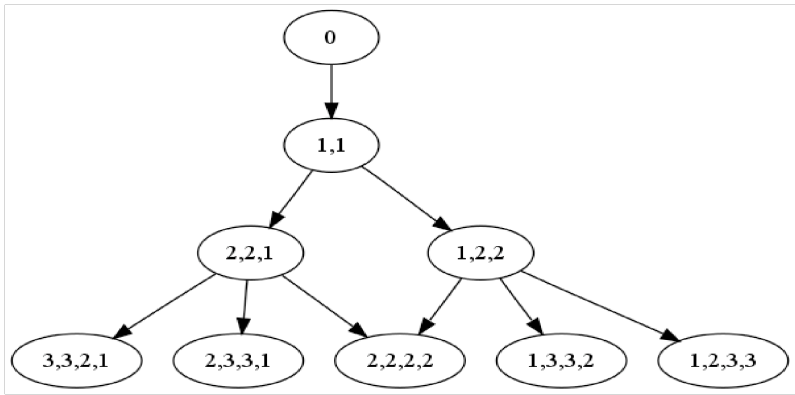



(1, 2, 3, 3)

There are $C_k$ RPs with $k$ splits

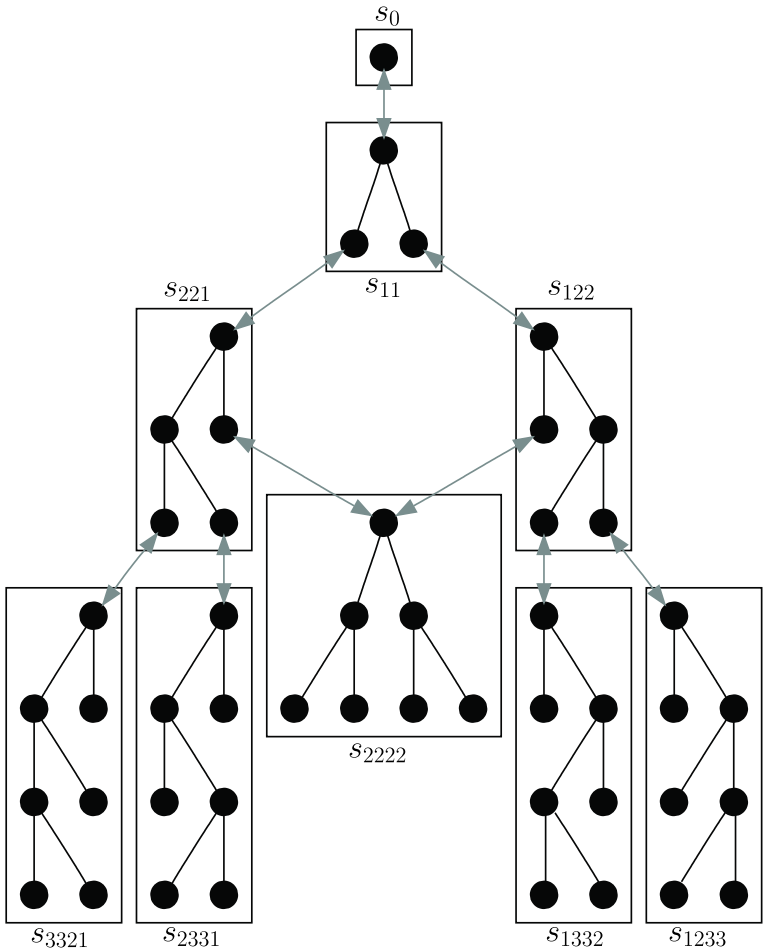


$$
\begin{aligned}
C_0 &= 1 \\
C_1 &= 1 \\
C_2 &= 2 \\
C_3 &= 5 \\
C_4 &= 14 \\
C_5 &= 42 \\
&\dots \\
C_k &= \frac{(2k)!}{(k+1)!k!} \\
&\dots \\
C_{15} &= 9694845 \\
&\dots \\
C_{20} &= 6564120420
\end{aligned}
$$



# Hasse (transition) Diagram of Regular Pavings

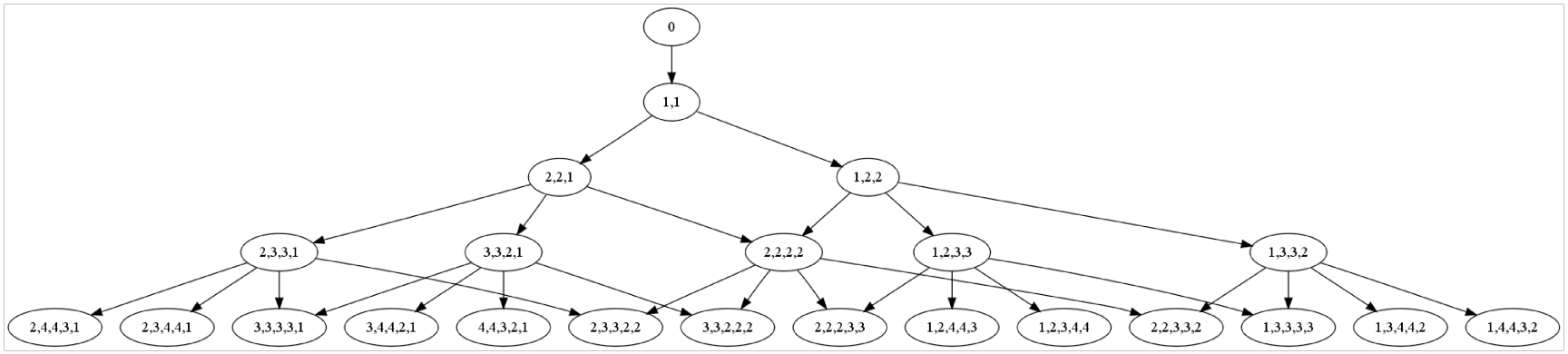
## Transition diagram over $\mathbb{S}_{0:3}$ with split/reunion operations




S, Taylor & Teng, Catalan Coefficients, Sequence A185155 in The On-Line Encyclop. of Integer Sequences, 2012

# Hasse (transition) Diagram of Regular Pavings

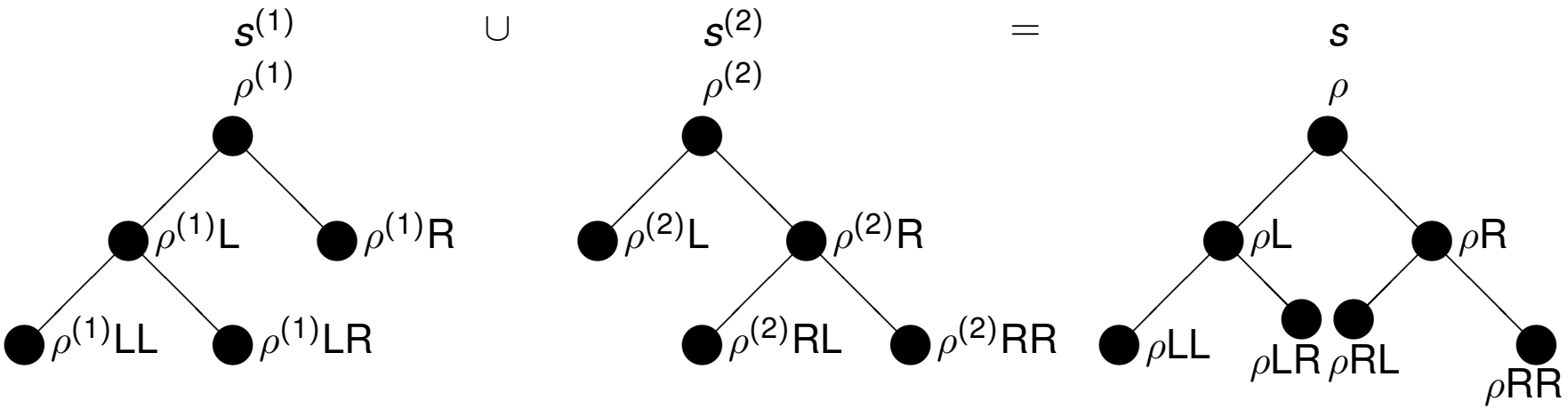
Transition diagram over $\mathbb{S}_{0:4}$ with split/reunion operations

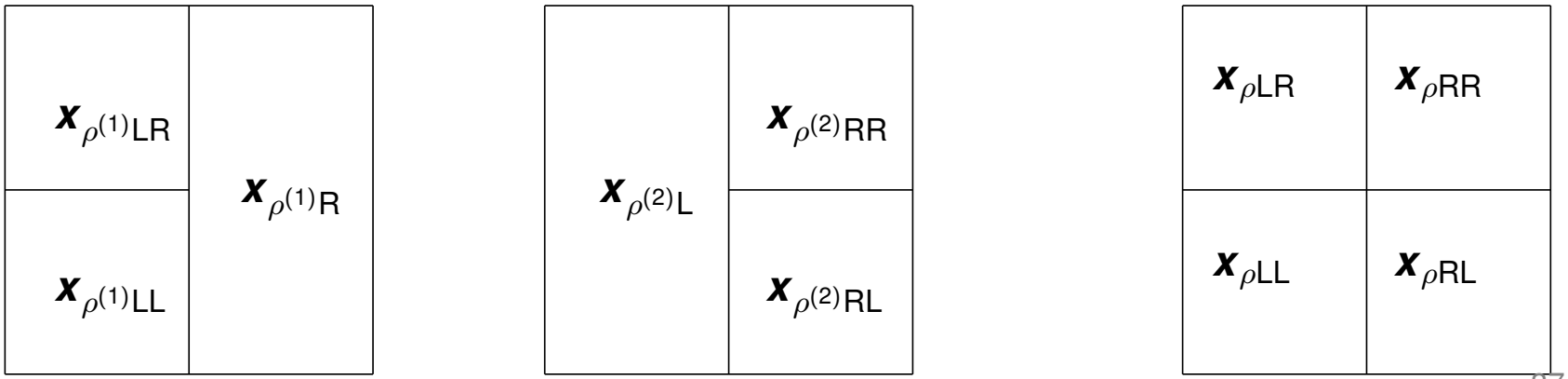



1. The above state space is denoted by $\mathbb{S}_{0:4}$
2. Number of RPs with $k$ splits is the Catalan number $C_k$
3. There is more than one way to reach a RP by $k$ splits
4. Randomized algorithms are Markov chains on $\mathbb{S}_{0:\infty}$

# RPs are closed under union operations

$s^{(1)} \cup s^{(2)} = s$ is union of two RPs $s^{(1)}$ and $s^{(2)}$ of $\boldsymbol{x}_\rho \in \mathbb{IR}^2$.



# RPs are closed under union operations

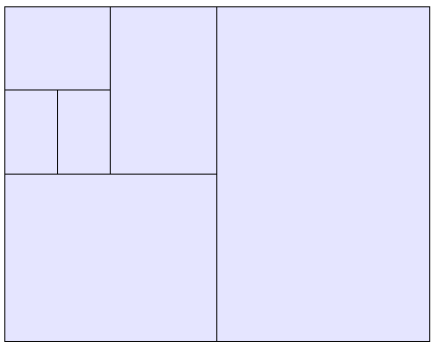
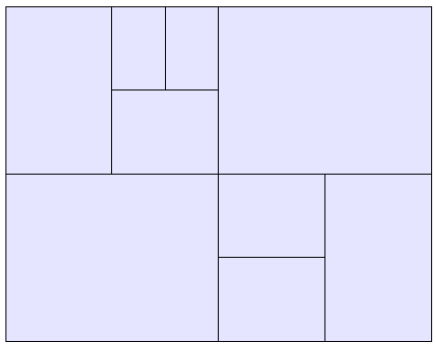
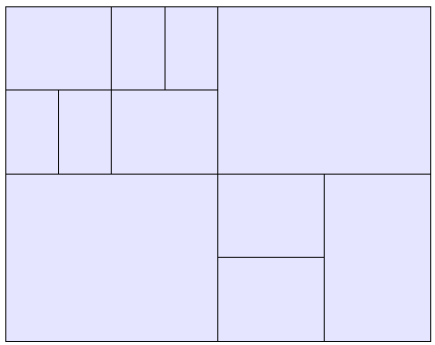
**Lemma 1:** The algebraic structure of RP trees (a.k.a. frb-trees underlying Thompson's group in geometric group theory) is closed under union operations.

# RPs are closed under union operations

**Lemma 1:** The algebraic structure of RP trees (a.k.a. frb-trees underlying Thompson's group in geometric group theory) is closed under union operations.

**Proof:** by a "transparency overlay process" argument (cf. Meier 2008).

$s^{(1)} \cup s^{(2)} = s$ is union of two RPs $s^{(1)}$ and $s^{(2)}$ of $\boldsymbol{x}_\rho \in \mathbb{R}^2$.

## Algorithm 1: `RPUnion`$(\rho^{(1)}, \rho^{(2)})$

**input** : Root nodes $\rho^{(1)}$ and $\rho^{(2)}$ of RPs $s^{(1)}$ and $s^{(2)}$, respectively, with root box $\boldsymbol{x}_{\rho^{(1)}} = \boldsymbol{x}_{\rho^{(2)}}$

**output** : Root node $\rho$ of RP $s = s^{(1)} \cup s^{(2)}$

**if** `IsLeaf`$(\rho^{(1)})$ & `IsLeaf`$(\rho^{(2)})$ **then**
  $\rho \leftarrow$ `Copy`$(\rho^{(1)})$
  **return** $\rho$
**end**

**else if** !`IsLeaf`$(\rho^{(1)})$ & `IsLeaf`$(\rho^{(2)})$ **then**
  $\rho \leftarrow$ `Copy`$(\rho^{(1)})$
  **return** $\rho$
**end**

**else if** `IsLeaf`$(\rho^{(1)})$ & !`IsLeaf`$(\rho^{(2)})$ **then**
  $\rho \leftarrow$ `Copy`$(\rho^{(2)})$
  **return** $\rho$
**end**

**else**
  !`IsLeaf`$(\rho^{(1)})$ & !`IsLeaf`$(\rho^{(2)})$
**end**

Make $\rho$ as a node with $\boldsymbol{x}_\rho \leftarrow \boldsymbol{x}_{\rho^{(1)}}$

Graft onto $\rho$ as left child the node `RPUnion`$(\rho^{(1)}\text{L}, \rho^{(2)}\text{L})$

Graft onto $\rho$ as right child the node `RPUnion`$(\rho^{(1)}\text{R}, \rho^{(2)}\text{R})$

**return** $\rho$

Harlow, S & Tucker, Mapped Regular Pavings, Reliable Computing, 2012

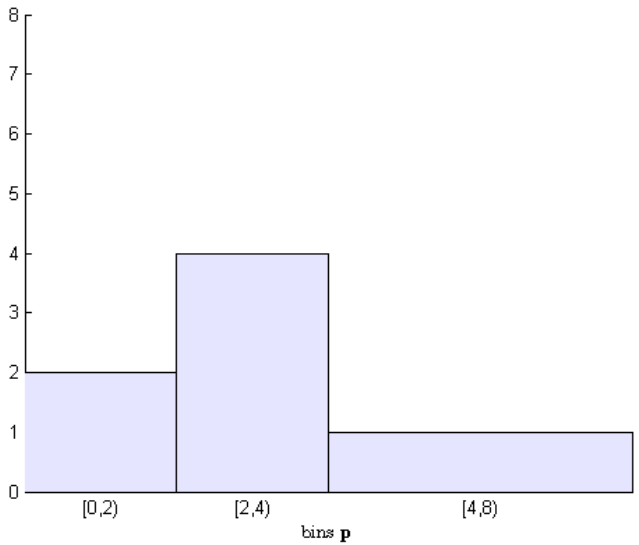
# Dfn: Real Mapped Regular Paving ($\mathbb{R}$-MRP)

## Simple functions over an RP tree partition

Let $f : \mathbb{V}(s) \to \mathbb{R}$ map each node of RP $s$ to a real number:

$$\{\rho v \mapsto f_{\rho v} : \rho v \in \mathbb{V}(s), f_{\rho v} \in \mathbb{R}\} \ .$$

$\mathbb{R}$-MRP over $s_{221}$ with $x_\rho = [0, 8]$



# $\mathbb{R}$-MRP Arithmetic

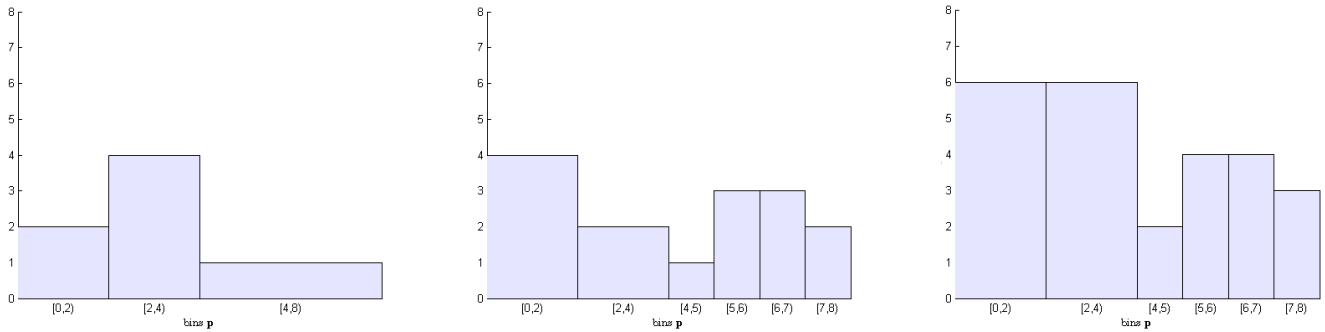
If $\star : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ then we can extend $\star$ point-wise to two $\mathbb{R}$-MRPs *f* and *g* with root nodes $\rho^{(1)}$ and $\rho^{(2)}$ via `MRPOperate`$(\rho^{(1)}, \rho^{(2)}, \star)$.

# $\mathbb{R}$-MRP Arithmetic

If $\star : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ then we can extend $\star$ point-wise to two $\mathbb{R}$-MRPs $f$ and $g$ with root nodes $\rho^{(1)}$ and $\rho^{(2)}$ via $\texttt{MRPOperate}(\rho^{(1)}, \rho^{(2)}, \star)$.

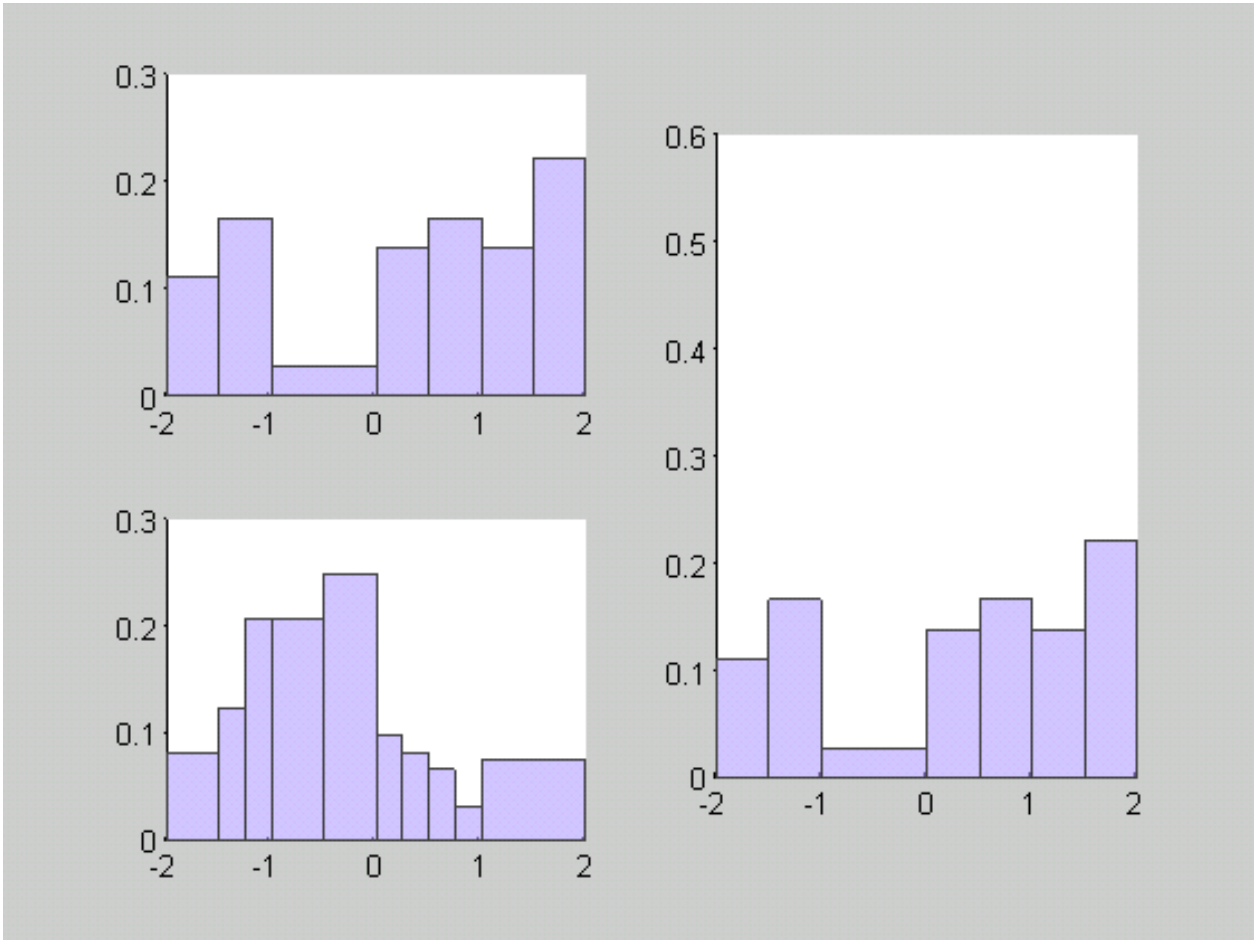
The addition below is done using $\texttt{MRPOperate}(\rho^{(1)}, \rho^{(2)}, +)$

$f$ $g$ $f + g$



# $\mathbb{R}$-MRP Addition by $\texttt{MRPOperate}(\rho^{(1)}, \rho^{(2)}, +)$

adding two piece-wise constant functions or $\mathbb{R}$-MRPs



# Algorithm 2: `MRPOperate`$(\rho^{(1)}, \rho^{(2)}, \star)$

**input** : two root nodes $\rho^{(1)}$ and $\rho^{(2)}$ with same root box $\boldsymbol{x}_{\rho^{(1)}} = \boldsymbol{x}_{\rho^{(2)}}$ and binary operation $\star$.

**output** : the root node $\rho$ of $\mathbb{R}$-MRP $h = f \star g$.

Make a new node $\rho$ with box and image
$\boldsymbol{x}_\rho \leftarrow \boldsymbol{x}_{\rho^{(1)}}$; $h_\rho \leftarrow f_{\rho^{(1)}} \star g_{\rho^{(2)}}$

**if** `IsLeaf`$(\rho^{(1)})$ & !`IsLeaf`$(\rho^{(2)})$ **then**
    Make temporary nodes L′, R′
    $\boldsymbol{x}_{\mathsf{L}'} \leftarrow \boldsymbol{x}_{\rho^{(1)}\mathsf{L}}$; $\boldsymbol{x}_{\mathsf{R}'} \leftarrow \boldsymbol{x}_{\rho^{(1)}\mathsf{R}}$
    $f_{\mathsf{L}'} \leftarrow f_{\rho^{(1)}}$, $f_{\mathsf{R}'} \leftarrow f_{\rho^{(1)}}$
    Graft onto $\rho$ as left child the node `MRPOperate`(L′, $\rho^{(2)}$L, $\star$)
    Graft onto $\rho$ as right child the node `MRPOperate`(R′, $\rho^{(2)}$R, $\star$)
**end**

**else if** !`IsLeaf`$(\rho^{(1)})$ & `IsLeaf`$(\rho^{(2)})$ **then**
    Make temporary nodes L′, R′
    $\boldsymbol{x}_{\mathsf{L}'} \leftarrow \boldsymbol{x}_{\rho^{(2)}\mathsf{L}}$; $\boldsymbol{x}_{\mathsf{R}'} \leftarrow \boldsymbol{x}_{\rho^{(2)}\mathsf{R}}$
    $g_{\mathsf{L}'} \leftarrow g_{\rho^{(2)}}$, $g_{\mathsf{R}'} \leftarrow g_{\rho^{(2)}}$
    Graft onto $\rho$ as left child the node `MRPOperate`($\rho^{(1)}$L, L′, $\star$)
    Graft onto $\rho$ as right child the node `MRPOperate`($\rho^{(1)}$R, R′, $\star$)
**end**

**else if** !`IsLeaf`$(\rho^{(1)})$ & !`IsLeaf`$(\rho^{(2)})$ **then**
    Graft onto $\rho$ as left child the node `MRPOperate`($\rho^{(1)}$L, $\rho^{(2)}$L, $\star$)
    Graft onto $\rho$ as right child the node `MRPOperate`($\rho^{(1)}$R, $\rho^{(2)}$R, $\star$)
**end**
**return** $\rho$

# Unary transformations are easy too

Let $\texttt{MRPTransform}(\rho, \tau)$ apply the unary transformation $\tau : \mathbb{R} \to \mathbb{R}$ to a given $\mathbb{R}$-MRP $f$ with root node $\rho$ as follows:

- copy $f$ to $g$
- recursively set $f_{\rho v} = \tau(f_{\rho v})$ for each node $\rho v$ in $g$
- return $g$ as $\tau(f)$

# Arithmetic and Algebra of $\mathbb{R}$-MRPs

Thus, we can obtain any $\mathbb{R}$-MRP arithmetical expression that is specified by finitely many sub-expressions involving:

1. constant $\mathbb{R}$-MRP,

# Arithmetic and Algebra of $\mathbb{R}$-MRPs

Thus, we can obtain any $\mathbb{R}$-MRP arithmetical expression that is specified by finitely many sub-expressions involving:

1. constant $\mathbb{R}$-MRP,
2. binary arithmetic operation $\star \in \{+, -, \cdot, /\}$ over two $\mathbb{R}$-MRPs,

# Arithmetic and Algebra of $\mathbb{R}$-MRPs

Thus, we can obtain any $\mathbb{R}$-MRP arithmetical expression that is specified by finitely many sub-expressions involving:

1. constant $\mathbb{R}$-MRP,
2. binary arithmetic operation $\star \in \{+, -, \cdot, /\}$ over two $\mathbb{R}$-MRPs,
3. standard transformations of $\mathbb{R}$-MRPs by elements of $\mathfrak{S} := \{\exp, \sin, \cos, \tan, \ldots\}$ and

# Arithmetic and Algebra of $\mathbb{R}$-MRPs

Thus, we can obtain any $\mathbb{R}$-MRP arithmetical expression that is specified by finitely many sub-expressions involving:

1. constant $\mathbb{R}$-MRP,
2. binary arithmetic operation $\star \in \{+, -, \cdot, /\}$ over two $\mathbb{R}$-MRPs,
3. standard transformations of $\mathbb{R}$-MRPs by elements of $\mathfrak{S} := \{\exp, \sin, \cos, \tan, \ldots\}$ and
4. their compositions.

# Stone-Wierstrass Theorem: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

## Theorem (Harlow, S & Tucker, 2012)

*Let $\mathcal{F}$ be the class of $\mathbb{R}$-MRPs with the same root box $\boldsymbol{x}_\rho$. Then $\mathcal{F}$ is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$, the algebra of real-valued continuous functions on $\boldsymbol{x}_\rho$.*

# Stone-Wierstrass Theorem: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

## Theorem (Harlow, S & Tucker, 2012)

*Let $\mathcal{F}$ be the class of $\mathbb{R}$-MRPs with the same root box $\boldsymbol{x}_\rho$. Then $\mathcal{F}$ is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$, the algebra of real-valued continuous functions on $\boldsymbol{x}_\rho$.*

**Proof:**

Since $\boldsymbol{x}_\rho \in \mathbb{IR}^d$ is a compact Hausdorff space, by the Stone-Weierstrass theorem we can establish that $\mathcal{F}$ is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$ with the topology of uniform convergence, provided that $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ that separates points in $\boldsymbol{x}_\rho$ and which contains a non-zero constant function.

# Stone-Wierstrass Theorem: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

## Theorem (Harlow, S & Tucker, 2012)

*Let $\mathcal{F}$ be the class of $\mathbb{R}$-MRPs with the same root box $\boldsymbol{x}_\rho$. Then $\mathcal{F}$ is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$, the algebra of real-valued continuous functions on $\boldsymbol{x}_\rho$.*

**Proof:**

Since $\boldsymbol{x}_\rho \in \mathbb{IR}^d$ is a compact Hausdorff space, by the Stone-Weierstrass theorem we can establish that $\mathcal{F}$ is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$ with the topology of uniform convergence, provided that $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ that separates points in $\boldsymbol{x}_\rho$ and which contains a non-zero constant function.

We will show all these conditions are satisfied by $\mathcal{F}$

# Stone-Wierstrass Theorem Contd.: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

- $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ since it is closed under addition and scalar multiplication.

# Stone-Wierstrass Theorem Contd.: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

- $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ since it is closed under addition and scalar multiplication.
- $\mathcal{F}$ contains non-zero constant functions

# Stone-Wierstrass Theorem Contd.: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

- $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ since it is closed under addition and scalar multiplication.
- $\mathcal{F}$ contains non-zero constant functions
- Finally, RPs can clearly separate distinct points $x, x' \in \boldsymbol{x}_\rho$ into distinct leaf boxes by splitting deeply enough.

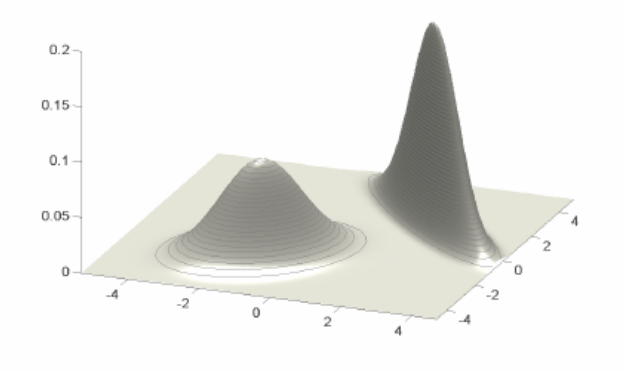
# Stone-Wierstrass Theorem Contd.: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

- $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ since it is closed under addition and scalar multiplication.
- $\mathcal{F}$ contains non-zero constant functions
- Finally, RPs can clearly separate distinct points $x, x' \in \boldsymbol{x}_\rho$ into distinct leaf boxes by splitting deeply enough.
- Thus, $\mathcal{F}$, the class of $\mathbb{R}$-MRPs with the same root box $\boldsymbol{x}_\rho$, is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$, the algebra of real-valued continuous functions on $\boldsymbol{x}_\rho$.

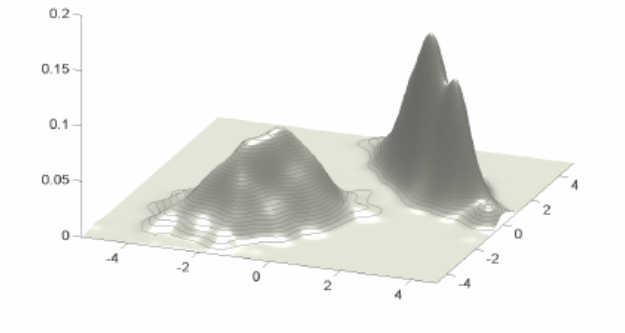
# Stone-Wierstrass Theorem Contd.: $\mathbb{R}$-MRPs Dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$

- $\mathcal{F}$ is a sub-algebra of $C(\boldsymbol{x}_\rho, \mathbb{R})$ since it is closed under addition and scalar multiplication.
- $\mathcal{F}$ contains non-zero constant functions
- Finally, RPs can clearly separate distinct points $x, x' \in \boldsymbol{x}_\rho$ into distinct leaf boxes by splitting deeply enough.
- Thus, $\mathcal{F}$, the class of $\mathbb{R}$-MRPs with the same root box $\boldsymbol{x}_\rho$, is dense in $C(\boldsymbol{x}_\rho, \mathbb{R})$, the algebra of real-valued continuous functions on $\boldsymbol{x}_\rho$.
- Q.E.D.

# Kernel Density Estimate (visualization of a procedure)

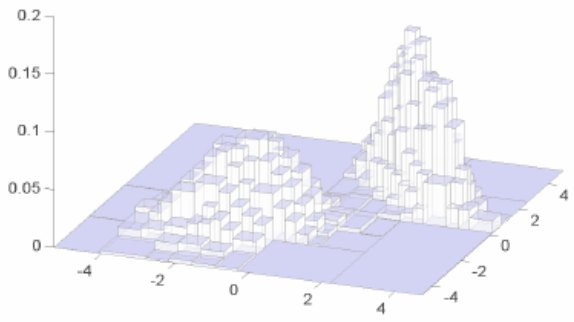



(a) True density.

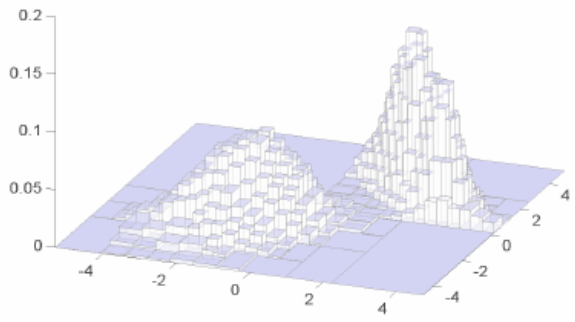



(c) MCMC bandwidth KDE.

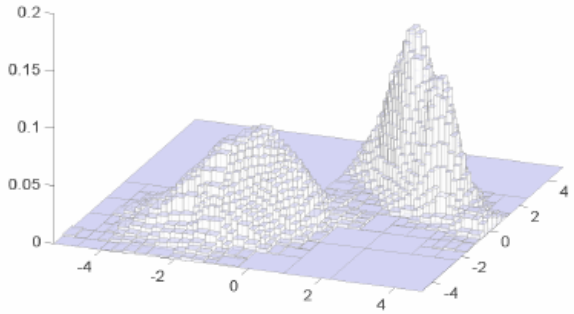
# Approximating Kernel Density Estimates by $\mathbb{R}$-MRPs



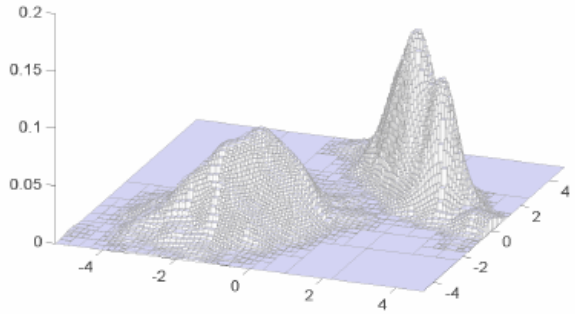

(a) $\overline{\psi} = 0.001$ (187 leaves).




(b) $\overline{\psi} = 0.005$ (316 leaves).




(c) $\overline{\psi} = 0.0001$ (919 leaves).




(d) $\overline{\psi} = 0.00001$ (4420 leaves).

# Approximating Kernel Density Estimates by $\mathbb{R}$-MRPs

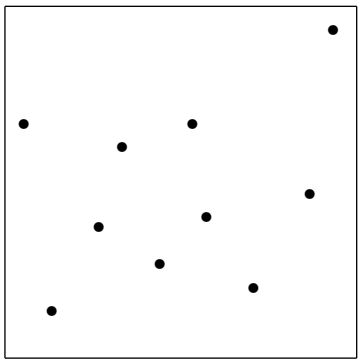
Table J.4: 5-$d$ case: estimated errors for KDE and RMRP-KDE approximations.

| | $\hat{d}_{KL}$ | $\hat{L}_1$ error | Time (s) | Leaves |
|---|---|---|---|---|
| KDE ($n_K = 2,000$) | 0.41 | 0.66 | 7,350–8,880 | $n/a$ |
| RMRP-KDE approximations | | | | |
| $\overline{\psi} = 0.0001$ | 5.06 | 0.96 | 1.0 | 2,363 |
| $\overline{\psi} = 0.00005$ | 4.85 | 0.91 | 2.3 | 4,639 |
| $\overline{\psi} = 0.00001$ | 4.51 | 0.85 | 8.7 | 17,759 |
| $\overline{\psi} = 0.000005$ | 4.49 | 0.84 | 17.2 | 31,335 |
| $\overline{\psi} = 0.000001$ | 3.33 | 0.76 | 66.1 | 133,493 |
| $\overline{\psi} = 0.0000005$ | 3.31 | 0.75 | 131.0 | 237,561 |
| $\overline{\psi} = 0.0000001$ | 3.54 | 0.74 | 470.0 | 895,012 |

# Statistical Regular Pavings (SRPs)

- Extended from the RP;
- 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 covariance matrix;
- Done in Dual-tree KDE of Gray & Moore (2003)
- Fisher (1920) already notes this (pers. commn. Lauritzen)

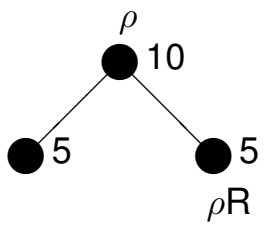
Caching the sample count in each node (or box).

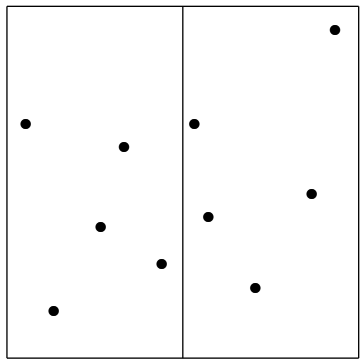



# Statistical Regular Pavings (SRPs)

- Extended from the RP;
- 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 covariance matrix;
- Done in Dual-tree KDE of Gray & Moore (2003)
- Fisher (1920) already notes this (pers. commn. Lauritzen)

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

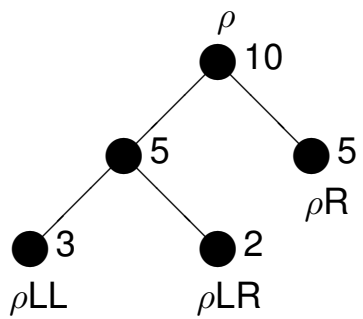


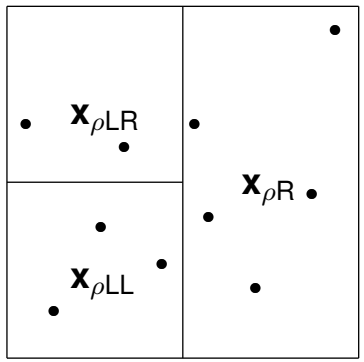



# Statistical Regular Pavings (SRPs)

- Extended from the RP;
- 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 covariance matrix;
- Done in Dual-tree KDE of Gray & Moore (2003)
- Fisher (1920) already notes this (pers. commn. Lauritzen)

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



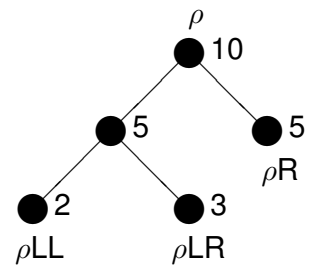
# SRPs as Adaptive Histograms

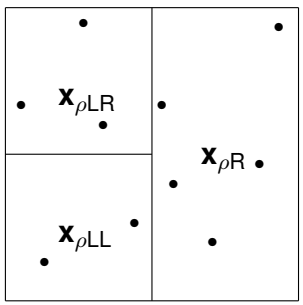
SRP estimate of $f$ from random vectors $X_1, X_2, \ldots, X_n \overset{iid}{\sim} f$ is

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

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

Figure : A SRP as a histogram estimate.



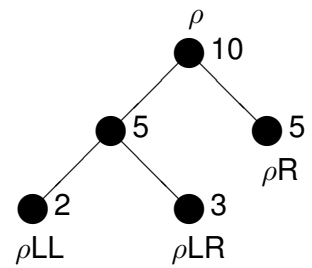
# SRPs as Adaptive Histograms

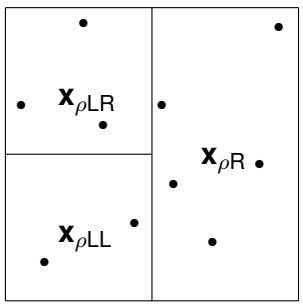
SRP estimate of $f$ from random vectors $X_1, X_2, \ldots, X_n \overset{iid}{\sim} f$ is

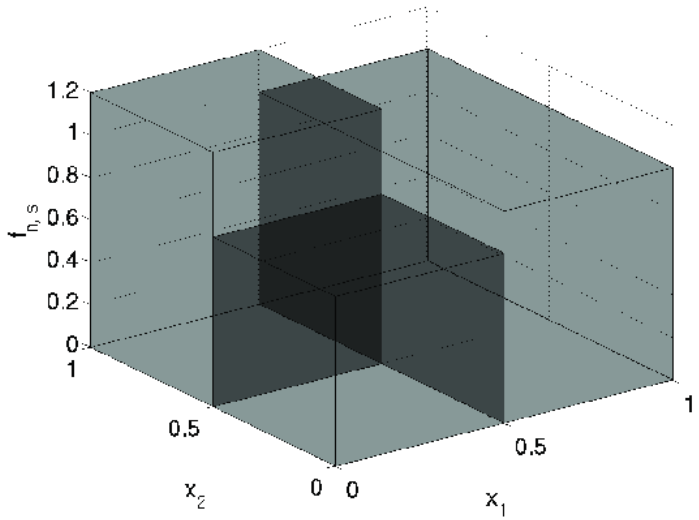
$$f_{n,\dot{s}}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{\mathbb{1}\left(x_i \in \mathbf{x}(x)\right)}{\mathrm{vol}\left(\mathbf{x}(x)\right)} \quad ,$$

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

Figure : A SRP as a histogram estimate.



Section 3

# Bayesian Smoothing by Averaging – MCMC

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

- The most likely histogram with partition $\ell(s)$ given by the leaves of RP $s$ with $k$ splits and $k+1$ leaves in $\mathbb{S}_k$ is:

$$f_{n,s}(x) = \begin{cases} \frac{\mu_n(\mathbf{x}(x))}{\mathrm{vol}\,(\mathbf{x}(x))} = \frac{\#\mathbf{x}(x)/n}{\mathrm{vol}\,(\mathbf{x}(x))} & \text{if } \mathrm{vol}\,(\mathbf{x}(x)) < \infty, \mathbf{x}(x) \in \ell(s), \\ 0 & \text{otherwise.} \end{cases}$$

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

- The most likely histogram with partition $\ell(s)$ given by the leaves of RP $s$ with $k$ splits and $k+1$ leaves in $\mathbb{S}_k$ is:

$$f_{n,s}(x) = \begin{cases} \frac{\mu_n(\mathbf{x}(x))}{\mathrm{vol}\,(\mathbf{x}(x))} = \frac{\#\mathbf{x}(x)/n}{\mathrm{vol}\,(\mathbf{x}(x))} & \text{if } \mathrm{vol}\,(\mathbf{x}(x)) < \infty, \mathbf{x}(x) \in \ell(s), \\ 0 & \text{otherwise.} \end{cases}$$

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

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

- The most likely histogram with partition $\ell(s)$ given by the leaves of RP $s$ with $k$ splits and $k+1$ leaves in $\mathbb{S}_k$ is:

$$f_{n,s}(x) = \begin{cases} \frac{\mu_n(\mathbf{x}(x))}{\mathrm{vol}\,(\mathbf{x}(x))} = \frac{\#\mathbf{x}(x)/n}{\mathrm{vol}\,(\mathbf{x}(x))} & \text{if } \mathrm{vol}\,(\mathbf{x}(x)) < \infty, \mathbf{x}(x) \in \ell(s), \\ 0 & \text{otherwise.} \end{cases}$$

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

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

- The most likely histogram with partition $\ell(s)$ given by the leaves of RP $s$ with $k$ splits and $k+1$ leaves in $\mathbb{S}_k$ is:

$$f_{n,s}(x) = \begin{cases} \frac{\mu_n(\mathbf{x}(x))}{\mathrm{vol}\,(\mathbf{x}(x))} = \frac{\#\mathbf{x}(x)/n}{\mathrm{vol}\,(\mathbf{x}(x))} & \text{if } \mathrm{vol}\,(\mathbf{x}(x)) < \infty, \mathbf{x}(x) \in \ell(s), \\ 0 & \text{otherwise.} \end{cases}$$

- Let the prior probability be $P(s) \propto \frac{1}{C_k^2}$, $s \in \mathbb{S}_k$
- Then the posterior density of histogram $f_{n,s}$ with k splits is

$$P(f_{n,s}|X_{1:n}) \propto P(X_{1:n}|s)P(s) = \prod_{\mathbf{x}_{\rho v} \in \ell(s)} \left( \frac{\#\mathbf{x}_{\rho v}}{n\ \mathrm{vol}\,(\mathbf{x}_{\rho v})} \right)^{\#\mathbf{x}_{\rho v}} \frac{1}{C_k^2}$$

# Averaging SRP Histograms for Posterior Mean

## Adding $m$ SRP histograms is fast & recursive as $\mathbb{R}$-MRPs

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

# Averaging SRP Histograms for Posterior Mean

## Adding $m$ SRP histograms is fast & recursive as $\mathbb{R}$-MRPs

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

## Averaging gives the posterior sample mean histogram:

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

# Averaging SRP Histograms for Posterior Mean

## Adding $m$ SRP histograms is fast & recursive as $\mathbb{R}$-MRPs

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

## Averaging gives the posterior sample mean histogram:

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

provided, $f_{n,s^{(1)}}, f_{n,s^{(2)}}, \ldots, f_{n,s^{(m)}} \sim P(f_{n,s^{(\cdot)}}|X_{1:n}), s^{(\cdot)} \in \mathbb{S}_{0:\infty}$

# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$

# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$
- We propose uniformly at random to split a leaf node or merge a cherry (sub-terminal) node of current state $s^{(i)}$

# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$
- We propose uniformly at random to split a leaf node or merge a cherry (sub-terminal) node of current state $s^{(i)}$
- **Repeat**

# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$
- We propose uniformly at random to split a leaf node or merge a cherry (sub-terminal) node of current state $s^{(i)}$
- **Repeat**
  - **Draw** $u \sim U(0,1)$

# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$
- We propose uniformly at random to split a leaf node or merge a cherry (sub-terminal) node of current state $s^{(i)}$
- **Repeat**
  - **Draw** $u \sim U(0,1)$
  - **If** $u < \frac{P(f_{n,s'}|X_{1:n})}{P(f_{n,s^{(i)}}|X_{1:n})} \frac{q(s^{(i)}|s')}{q(s'|s^{(i)})}$ **then** $s^{(i+1)} \leftarrow s'$

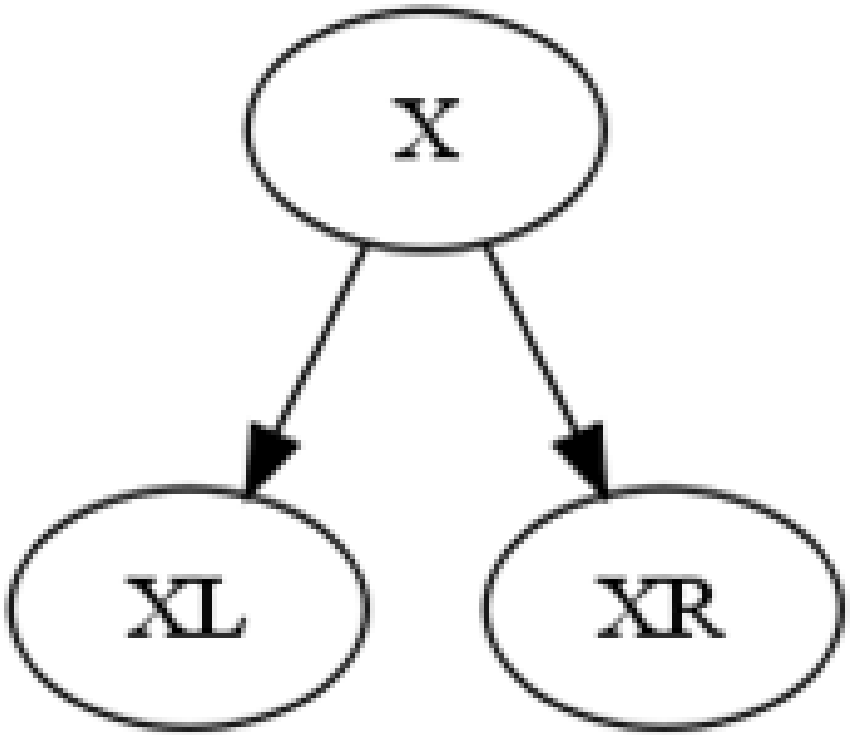
# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$
- We propose uniformly at random to split a leaf node or merge a cherry (sub-terminal) node of current state $s^{(i)}$
- **Repeat**
  - **Draw** $u \sim U(0,1)$
  - **If** $u < \frac{P(f_{n,s'}|X_{1:n})}{P(f_{n,s^{(i)}}|X_{1:n})} \frac{q(s^{(i)}|s')}{q(s'|s^{(i)})}$ **then** $s^{(i+1)} \leftarrow s'$
  - **else** $s^{(i+1)} \leftarrow s^{(i)}$

# Metropolis-Hastings Algorithm

- Use a proposal density $q(s'|s^{(i)})$ which depends on current state $s^{(i)}$, to generate a new proposed state $s'$
- We propose uniformly at random to split a leaf node or merge a cherry (sub-terminal) node of current state $s^{(i)}$
- **Repeat**
  - **Draw** $u \sim U(0,1)$
  - **If** $u < \frac{P(f_{n,s'}|X_{1:n})}{P(f_{n,s^{(i)}}|X_{1:n})} \frac{q(s^{(i)}|s')}{q(s'|s^{(i)})}$ **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 $P(f_{n,s}|X_{1:n})$ over $\mathbb{S}_{0:\infty}$

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



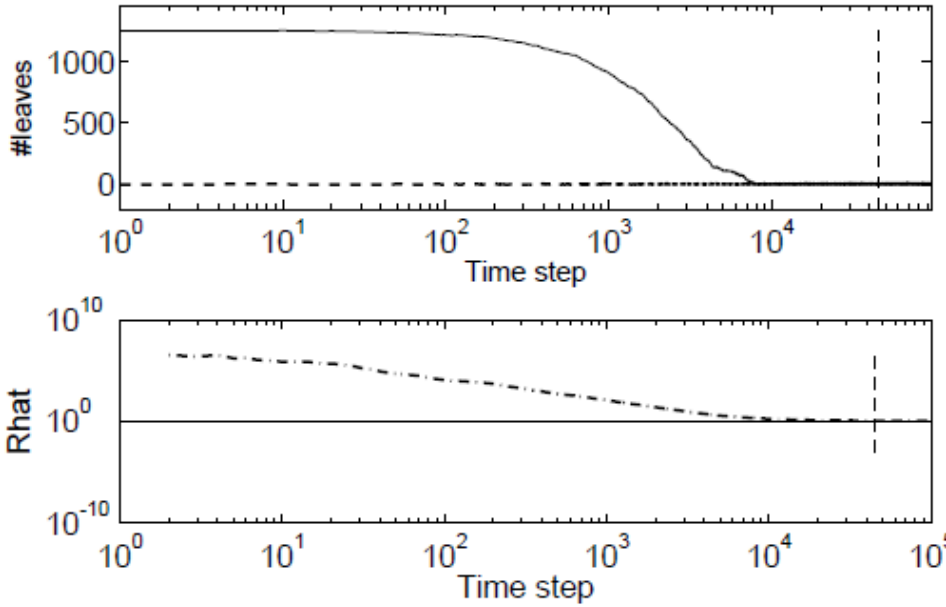
Section 4

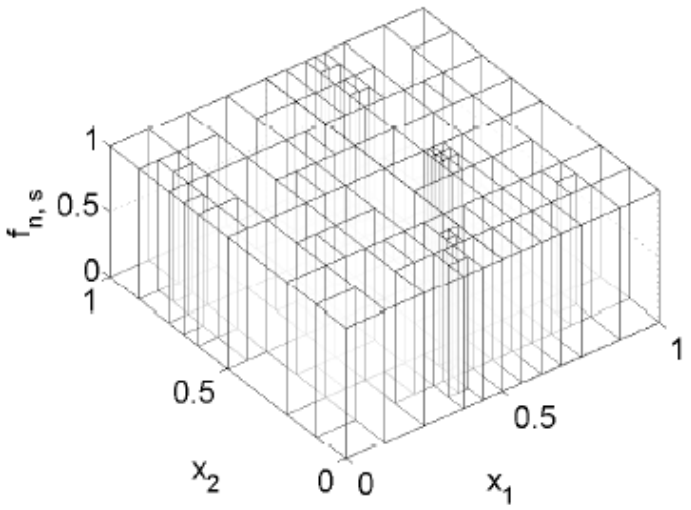
# Examples - good, bad and ugly

# Good: $f \sim \text{Uniform}(0,1)^D$ – The Curse of Dimensions $\not\exists$ for Unstructured $f$

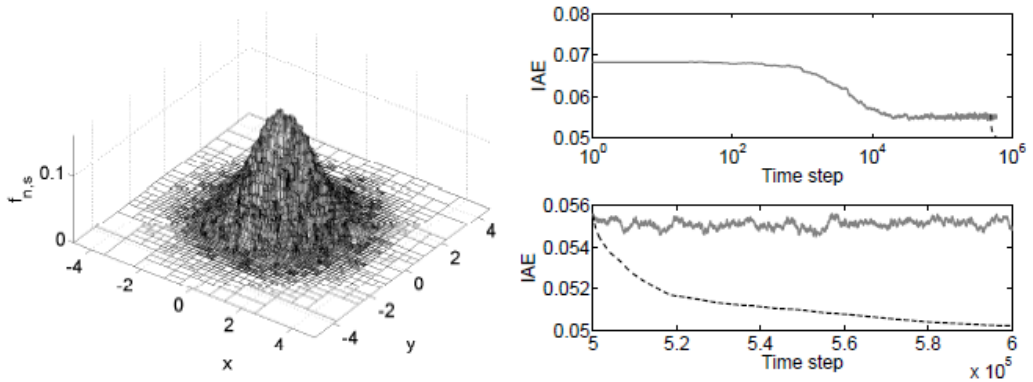
MIAE (std. err.) for $n$ samples from uniform density in various dimensions (CPU Times < $O$(1minute)).

| $n$ | 1D | 2D | 10D | 100D | 1000D |
|---|---|---|---|---|---|
| $10^2$ | 0.1112 (0.0707) | 0.1425 (0.0882) | 0.1170 (0.0723) | 0.0958 (0.0605) | 0.1111 (0.0524) |
| $10^3$ | 0.0366 (0.0192) | 0.0363 (0.0219) | 0.0442 (0.0275) | 0.0413 (0.0196) | 0.0305 (0.0195) |
| $10^4$ | 0.0164 (0.0095) | 0.0124 (0.0073) | 0.0115 (0.0070) | 0.0111 (0.0083) | 0.0089 (0.0065) |
| $10^5$ | 0.0041 (0.0020) | 0.0040 (0.0026) | 0.0041 (0.0028) | 0.0050 (0.0030) | 0.0043 (0.0025) |
| $10^6$ | 0.0011 (0.0005) | 0.0016 (0.0007) | 0.0010 (0.0006) | 0.0012 (0.0001) | 0.0010 (0.0004) |
| $10^7$ | 0.0004 (0.0003) | 0.0003 (0.0002) | 0.0003 (0.0002) | 0.0002 (0.0001) | - |
| $10^8$ | 0.0001 (0.0009) | 0.0002 (0.0002) | 0.0001 (0.0001) | - | - |



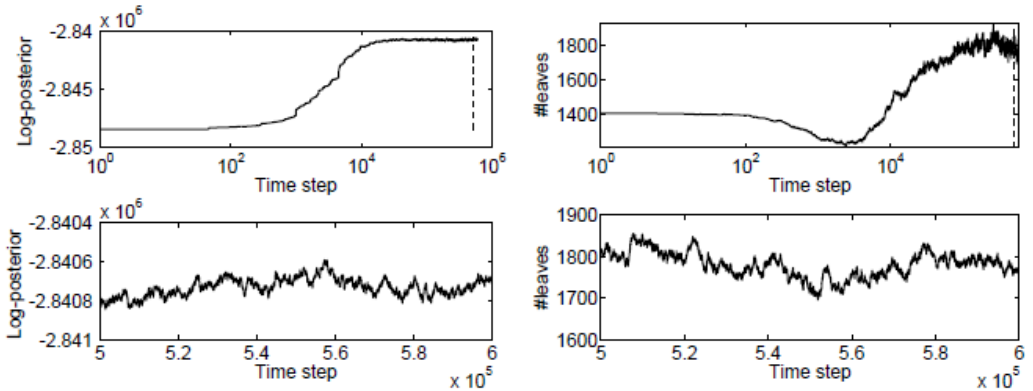



# Bad: $f \sim$ 2D Gaussian – The Curse of Dimensions $\exists$ for Structured $f$




(a) Posterior mean histogram estimate.

(b) IAEs for current (gray) and averaged states (black).



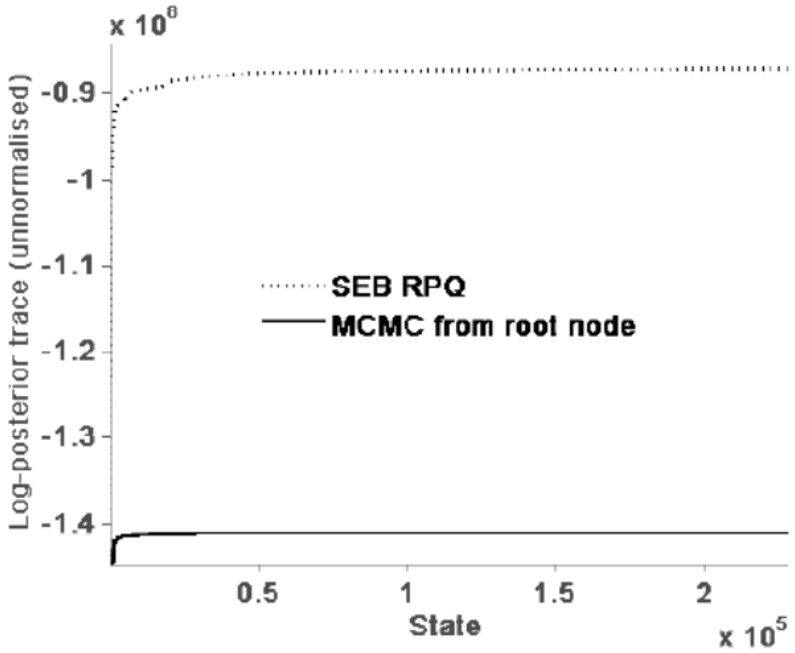

(c) Trace plots for log-posterior.

(d) Trace plots for number of leaves.

Need to initialize MCMC from states with many leaves for structured $f$

# Ugly: $f \sim$ 6D Gaussian – MCMC Can Get Stuck at Local Maxima!

Log-posterior traces of MCMC started at root node & at a good state




(a) Initial SEB phase compared with MCMC from root node.

**Solution**: Start from highest log-posterior states by a Randomized Priority Queue

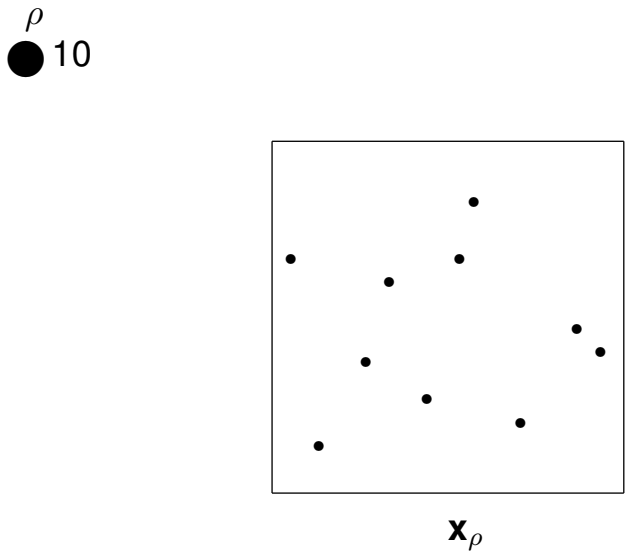
Section 5

# Randomized Priority Queue Markov chain

# A Prioritized Queue based Algorithm (for $L_1$ Consistent Initialization)

## Algorithm `SplitMostCounts`

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

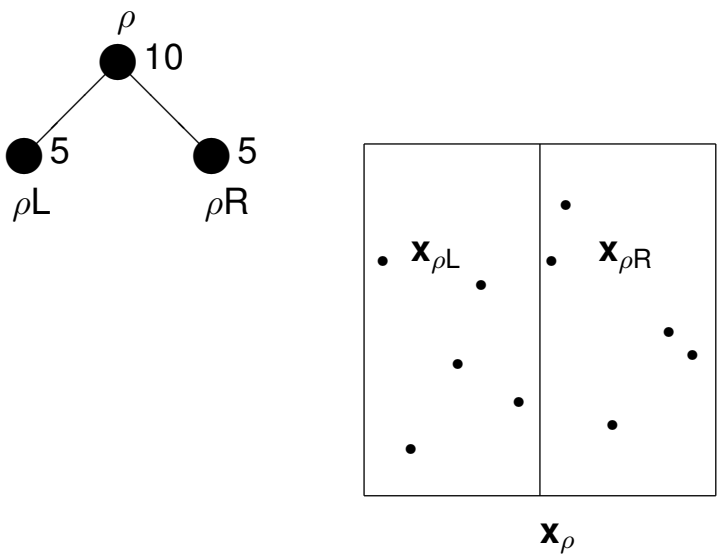


# A Prioritized Queue based Algorithm (for $L_1$ Consistent Initialization)

## Algorithm `SplitMostCounts`

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

Split the root box.

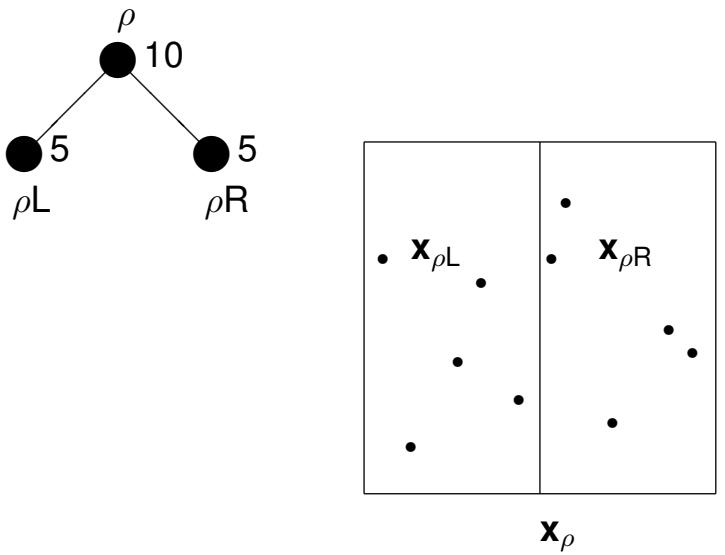


# A Prioritized Queue based Algorithm (for $L_1$ Consistent Initialization)

## Algorithm `SplitMostCounts`

As data arrives, order the leaf boxes of the SRP 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?

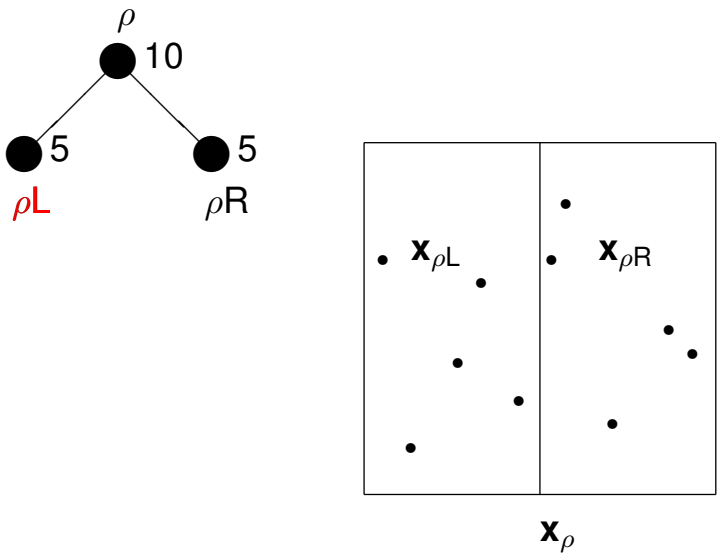


# A Prioritized Queue based Algorithm (for $L_1$ Consistent Initialization)

## Algorithm `SplitMostCounts`

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

Break such ties by randomising the next bisection.

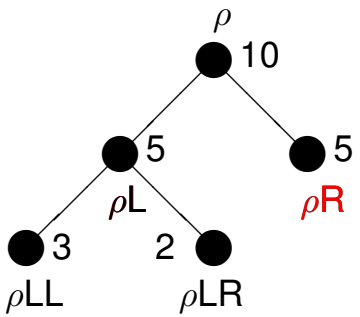


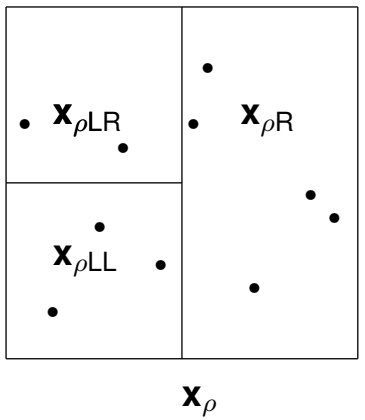
# A Prioritized Queue based Algorithm (for $L_1$ Consistent Initialization)

## Algorithm `SplitMostCounts`

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

Bisect until each box has $\leq \overline{k}_n$ points (let $\overline{k}_n = 3$ here).



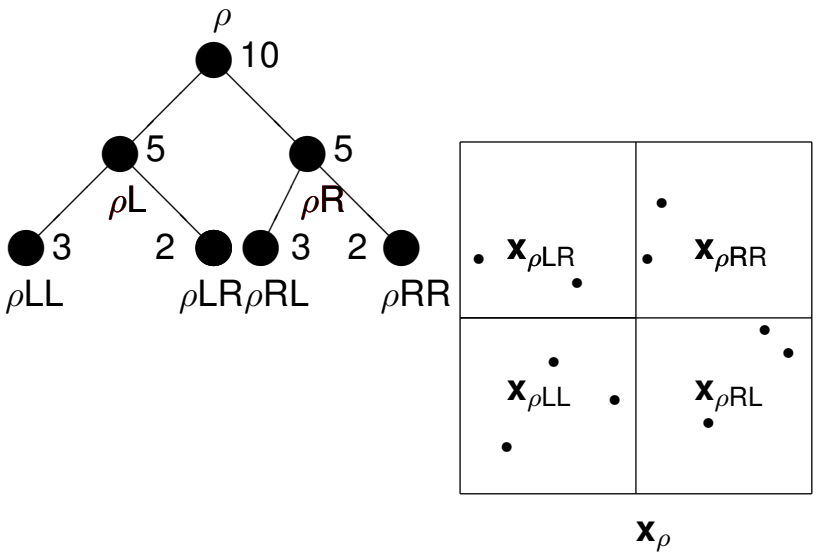



# A Prioritized Queue based Algorithm (for $L_1$ Consistent Initialization)

## Algorithm `SplitMostCounts`

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

Final state



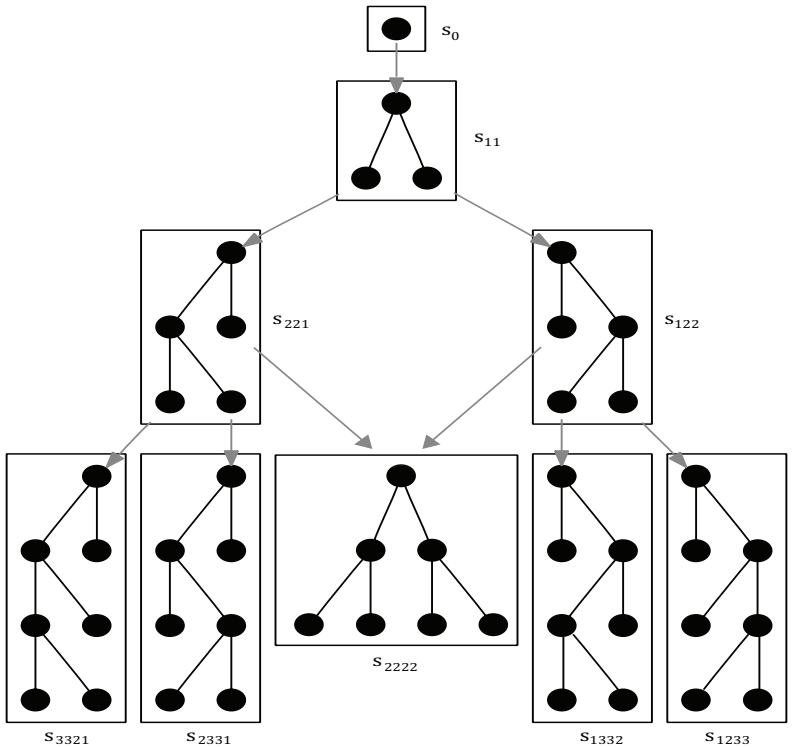
# The `SplitMostCounts` Algorithm

**Input:** (i) data: $x_1, \ldots, x_n \subseteq \mathbb{R}^d$; (ii) root box: $\mathbf{x}_\rho$;
(iii) SEB max: $\overline{k}_n$; (iv) maximum partition size: $\overline{m}_n$.
**Output:** histogram estimate $f_{n,s}$

initialize $i \leftarrow 1$; $s \leftarrow \mathbf{x}_\rho$;
repeat until
$\#\mathbf{x}_{\rho v} \leq \overline{k}_n$ for each $\mathbf{x}_{\rho v} \in \ell(s)$ and $i \leq \overline{m}_n$ // $\ell(s) = \{$`leaf boxes`$\}$
$\mathbf{x}_{\rho v} \leftarrow \text{Uniform}(\hat{\ell}(s))$ // `randomized PQ on leaf boxes`
$s \leftarrow$ `bisect`$(s, \mathbf{x}_{\rho v})$ // `bisect leaf box` $\mathbf{x}_{\rho v}$ `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 RPs of $\mathbf{x}_\rho$ made of $i$ splits and for $i, j \in \mathbb{N}$ with $i \leq j$, let $\mathbb{S}_{i:j}$ be the set of RPs with $k$ splits, $i \leq k \leq j$.




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

# $L_1$-Consistency of `SplitMostCounts` Markov chain

## Theorem (S & Teng, 2012)

*Let $X_1, X_2, \ldots$ be independent and identical random vectors in $\mathbb{R}^d$ whose common distribution $\mu$ has a non-atomic density $f$, i.e., $f \ll \lambda^d$. Let $\{S_n(i)\}_{i=0}^{J}$ 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 $\mathcal{L}_n$.*

# $L_1$-Consistency of `SplitMostCounts` Markov chain

## Theorem (S & Teng, 2012)

*Let $X_1, X_2, \ldots$ be independent and identical random vectors in $\mathbb{R}^d$ whose common distribution $\mu$ has a non-atomic density $f$, i.e., $f \ll \lambda^d$. Let $\{S_n(i)\}_{i=0}^{J}$ 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 $\mathcal{L}_n$.*
*Then, as $n \to \infty$, if $\overline{k}_n \to \infty$, $\overline{k}_n/n \to 0$, $\overline{m}_n \geq n/\overline{k}_n$, 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) $\mu(x : \mathrm{diam}(\mathbf{x}(x)) > \gamma) \to 0$ with probability 1 for every $\gamma > 0$,

are satisfied. Then by Theorem 1 of Lugosi & Nobel (Ann. Stats., 1996) our $f_{n,\dot{s}}$ is strongly consistent in $L_1$.

# 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) $\mu(x : \mathrm{diam}(\mathbf{x}(x)) > \gamma) \to 0$ with probability 1 for every $\gamma > 0$,

are satisfied. Then by Theorem 1 of Lugosi & Nobel (Ann. Stats., 1996) our $f_{n,\dot{s}}$ 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

# (a) Sub-linear Growth of the Number of Leaf Boxes

Let $\{S_n(i)\}_{i=0}^{J}$ on $\mathbb{S}_{0:\infty}$ be the Markov chain formed using `SplitMostCounts`. The Markov chain terminates at some state $\dot{s}$ with partition $\ell(\dot{s})$.

# (a) Sub-linear Growth of the Number of Leaf Boxes

Let $\{S_n(i)\}_{i=0}^{J}$ on $\mathbb{S}_{0:\infty}$ be the Markov chain formed using `SplitMostCounts`. The Markov chain terminates at some state $\dot{s}$ with partition $\ell(\dot{s})$. Associated with the Markov Chain is a fixed, non-random collection of partitions

$$\mathcal{L}_n := \{\ell(\dot{s}) : \dot{s} \in \mathbb{S}_{0:\infty}, P(S(J) = \dot{s}) > 0\} \subseteq \mathbb{S}_{0:\overline{m}_n - 1} \quad .$$

# (a) Sub-linear Growth of the Number of Leaf Boxes

Let $\{S_n(i)\}_{i=0}^{J}$ on $\mathbb{S}_{0:\infty}$ be the Markov chain formed using `SplitMostCounts`. The Markov chain terminates at some state $\dot{s}$ with partition $\ell(\dot{s})$. Associated with the Markov Chain is a fixed, non-random collection of partitions

$$\mathcal{L}_n := \{\ell(\dot{s}) : \dot{s} \in \mathbb{S}_{0:\infty}, P(S(J) = \dot{s}) > 0\} \subseteq \mathbb{S}_{0:\overline{m}_n - 1} \ .$$

The size of the largest partition $\ell(\dot{s})$ in $\mathcal{L}_n$ is given by

$$m(\mathcal{L}_n) := \sup_{\ell(\dot{s}) \in \mathcal{L}_n} |\ell(\dot{s})| \leq \overline{m}_n \ .$$

# (a) Sub-linear Growth of the Number of Leaf Boxes

Let $\{S_n(i)\}_{i=0}^{J}$ on $\mathbb{S}_{0:\infty}$ be the Markov chain formed using `SplitMostCounts`. The Markov chain terminates at some state $\dot{s}$ with partition $\ell(\dot{s})$. Associated with the Markov Chain is a fixed, non-random collection of partitions

$$\mathcal{L}_n := \{\ell(\dot{s}) : \dot{s} \in \mathbb{S}_{0:\infty}, P(S(J) = \dot{s}) > 0\} \subseteq \mathbb{S}_{0:\overline{m}_n - 1} \ .$$

The size of the largest partition $\ell(\dot{s})$ in $\mathcal{L}_n$ is given by

$$m(\mathcal{L}_n) := \sup_{\ell(\dot{s}) \in \mathcal{L}_n} |\ell(\dot{s})| \leq \overline{m}_n \ .$$

Thus, (a) is satisfied by assumption that $\overline{m}_n / n \to 0$.

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971).

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$.

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$. Let $\Delta(\mathcal{L}_n, B)$ be the number of distinct partitions of the finite set $B$ that are induced by partitions $\ell(\dot{s}) \in \mathcal{L}_n$:

$$\Delta(\mathcal{L}_n, B) := \left| \left\{ \left\{ \mathbf{x}_v \cap B : \mathbf{x}_v \in \ell(\dot{s}) \right\} : \ell(\dot{s}) \in \mathcal{L}_n \right\} \right| .$$

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$. Let $\Delta(\mathcal{L}_n, B)$ be the number of distinct partitions of the finite set $B$ that are induced by partitions $\ell(\dot{s}) \in \mathcal{L}_n$:

$$\Delta(\mathcal{L}_n, B) := \left| \left\{ \left\{ \mathbf{x}_v \cap B : \mathbf{x}_v \in \ell(\dot{s}) \right\} : \ell(\dot{s}) \in \mathcal{L}_n \right\} \right| \quad .$$

Define the growth function of $\mathcal{L}_n$ as

$$\Delta^*(\mathcal{L}_n, B) := \max_{B \in \left(\mathbb{R}^d\right)^n} \Delta(\mathcal{L}_n, B)$$

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$. Let $\Delta(\mathcal{L}_n, B)$ be the number of distinct partitions of the finite set $B$ that are induced by partitions $\ell(\dot{s}) \in \mathcal{L}_n$:

$$\Delta(\mathcal{L}_n, B) := \left| \left\{ \left\{ \mathbf{x}_v \cap B : \mathbf{x}_v \in \ell(\dot{s}) \right\} : \ell(\dot{s}) \in \mathcal{L}_n \right\} \right| \quad .$$

Define the growth function of $\mathcal{L}_n$ as

$$\Delta^*(\mathcal{L}_n, B) := \max_{B \in \left(\mathbb{R}^d\right)^n} \Delta(\mathcal{L}_n, B) \leq |\mathcal{L}_n|$$

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$. Let $\Delta(\mathcal{L}_n, B)$ be the number of distinct partitions of the finite set $B$ that are induced by partitions $\ell(\dot{s}) \in \mathcal{L}_n$:

$$\Delta(\mathcal{L}_n, B) := \left| \left\{ \left\{ \mathbf{x}_v \cap B : \mathbf{x}_v \in \ell(\dot{s}) \right\} : \ell(\dot{s}) \in \mathcal{L}_n \right\} \right| \quad .$$

Define the growth function of $\mathcal{L}_n$ as

$$\Delta^*(\mathcal{L}_n, B) := \max_{B \in \left(\mathbb{R}^d\right)^n} \Delta(\mathcal{L}_n, B) \leq |\mathcal{L}_n| \leq \sum_{k=0}^{\overline{m}_n} C_k$$

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$. Let $\Delta(\mathcal{L}_n, B)$ be the number of distinct partitions of the finite set $B$ that are induced by partitions $\ell(\dot{s}) \in \mathcal{L}_n$:

$$\Delta(\mathcal{L}_n, B) := \left| \left\{ \left\{ \mathbf{x}_v \cap B : \mathbf{x}_v \in \ell(\dot{s}) \right\} : \ell(\dot{s}) \in \mathcal{L}_n \right\} \right| \quad .$$

Define the growth function of $\mathcal{L}_n$ as

$$\Delta^*(\mathcal{L}_n, B) := \max_{B \in \left(\mathbb{R}^d\right)^n} \Delta(\mathcal{L}_n, B) \leq |\mathcal{L}_n| \leq \sum_{k=0}^{\overline{m}_n} C_k \approxeq \frac{4^{\overline{m}_n+1}}{3\overline{m}_n\sqrt{(\pi\overline{m}_n)}}$$

where $\approxeq$ is a known partial Catalan sum result (Mattarei, 2010).

# (b) Sub-exponential Growth of the Partition

The complexity of $\mathcal{L}_n$ will be measured by a combinatorial quantity similar to the growth function for classes of sets proposed by Vapnik and Chervonenkis (1971). Fix $n$ points $x_1, \ldots, x_n \in \mathbb{R}^d$ and let $B = \{x_1, \ldots, x_n\} \in \left(\mathbb{R}^d\right)^n$. Let $\Delta(\mathcal{L}_n, B)$ be the number of distinct partitions of the finite set $B$ that are induced by partitions $\ell(\dot{s}) \in \mathcal{L}_n$:

$$\Delta(\mathcal{L}_n, B) := \left| \left\{ \left\{ \mathbf{x}_v \cap B : \mathbf{x}_v \in \ell(\dot{s}) \right\} : \ell(\dot{s}) \in \mathcal{L}_n \right\} \right| \quad .$$

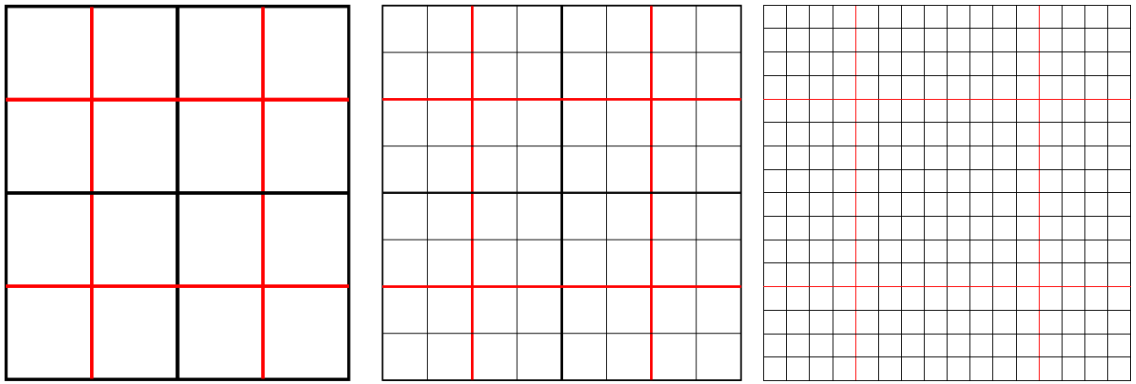
Define the growth function of $\mathcal{L}_n$ as

$$\Delta^*(\mathcal{L}_n, B) := \max_{B \in \left(\mathbb{R}^d\right)^n} \Delta(\mathcal{L}_n, B) \leq |\mathcal{L}_n| \leq \sum_{k=0}^{\overline{m}_n} C_k \approxeq \frac{4^{\overline{m}_n+1}}{3\overline{m}_n\sqrt{(\pi\overline{m}_n)}}$$

where $\approxeq$ is a known partial Catalan sum result (Mattarei, 2010). This ensures condition (b) is satisfied, i.e. $n^{-1} \log \Delta_n^*(\mathcal{L}_n) \to 0$.

# Shrinking Cells

$$\text{diam}(\mathbf{x}) = \sqrt{\sum_{i=1}^{d} (\overline{x}_i - \underline{x}_i)^2}, \mathbf{x} = [\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_d, \overline{x}_d]$$



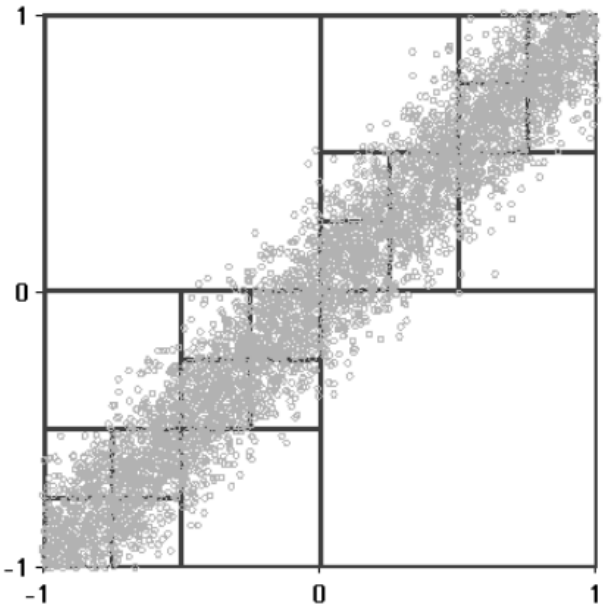
Basically find large enough box $[-M, +M]^d$ with almost all $\mu$ measure and diadically chop it to upper bound number of boxes with diameter $> \gamma$ and using VC $\neq$ to boxes in $\mathbb{R}^d$ to show:

(c) $\mu(x : \text{diam}(\mathbf{x}(x)) > \gamma) \to 0$ with probability 1 for every $\gamma > 0$.

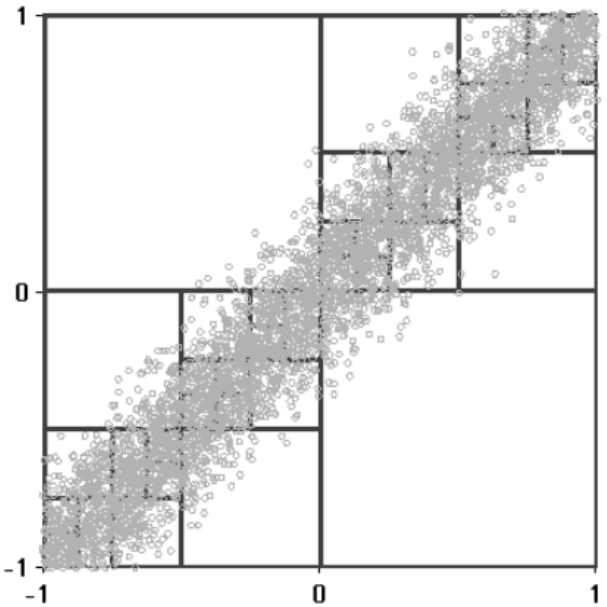
Q.E.D.

# Complementary PQ to "carve out" Support – A Trick

`SplitMostCounts` uses priority = $\mu_n(\mathbf{x}_{\rho v})$.



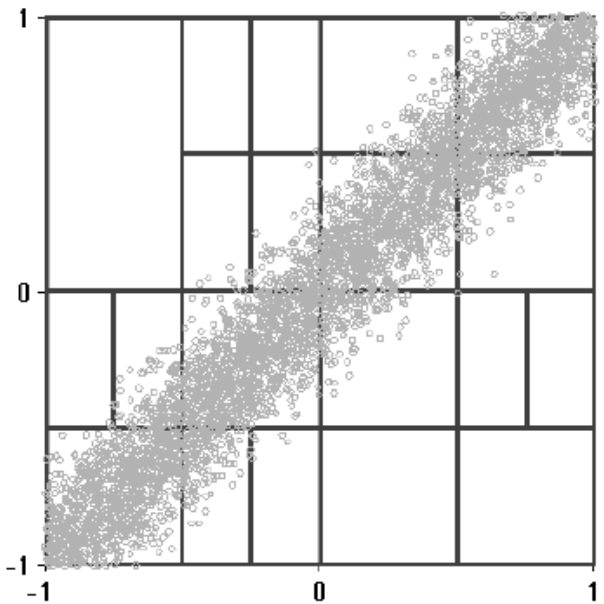

(a) 20 leaves.



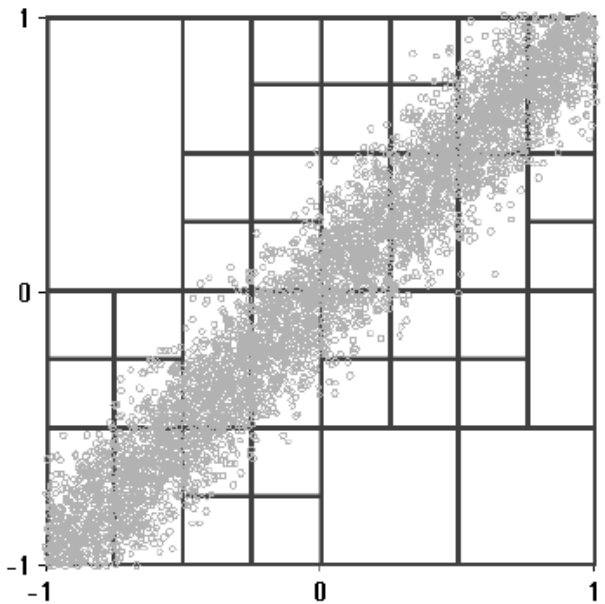

(b) 40 leaves.

# Complementary PQ to "carve out" Support – A Trick

`SupportCarver` uses priority = $(1 - \mu_n(\mathbf{x}_{\rho v}))\mathrm{vol}(\mathbf{x}_{\rho v})$.




(a) 20 leaves.

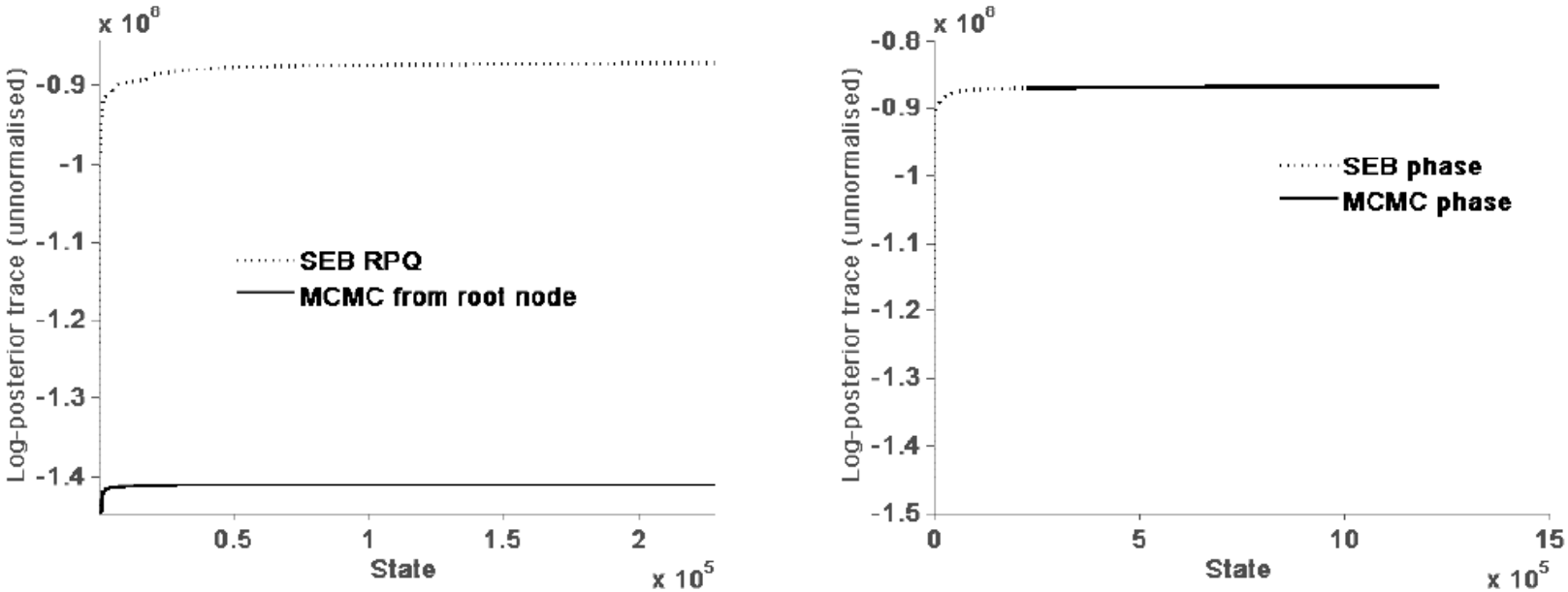



(b) 40 leaves.

Necessary to use `SupportCarver` for high-dimensional structured densities before using `SplitMostCounts`

# Ugly Revisited: $f \sim$ 6D Gaussian – Initialize MCMC by RPQ

Log-posterior traces of SEB RPQ Vs. MCMC started from root node



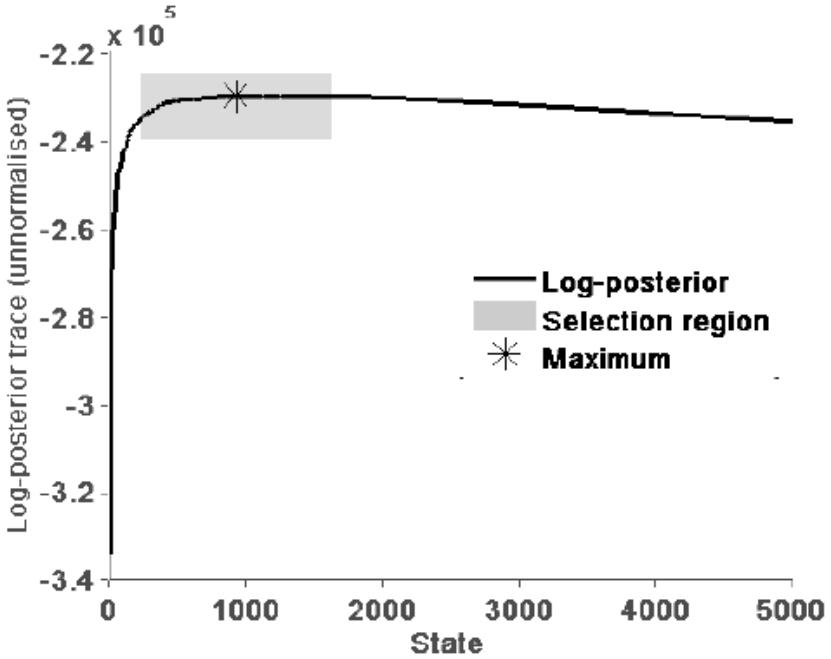

(a) Initial SEB phase compared with MCMC from root node.

(b) Combined log-posterior trace to $t = 1,000,000$.

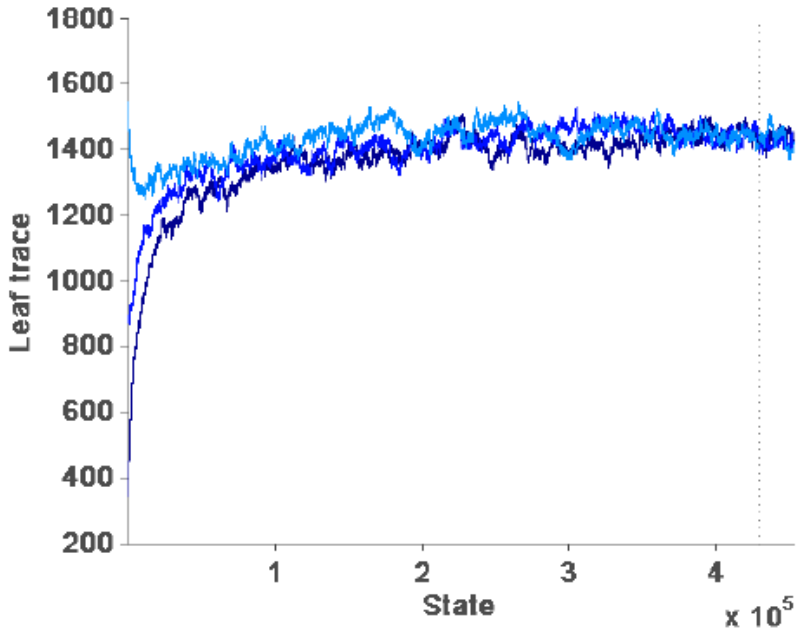
(data drawn from 6D Gaussian Density) – Initialize from highest log-posterior states visited by RPQ

# Ugly Revisited: $f \sim$ 6D Gaussian – Initialize MCMC by RPQ

Multiple MCMC chains started from high log-posterior region




(a) Selection region.

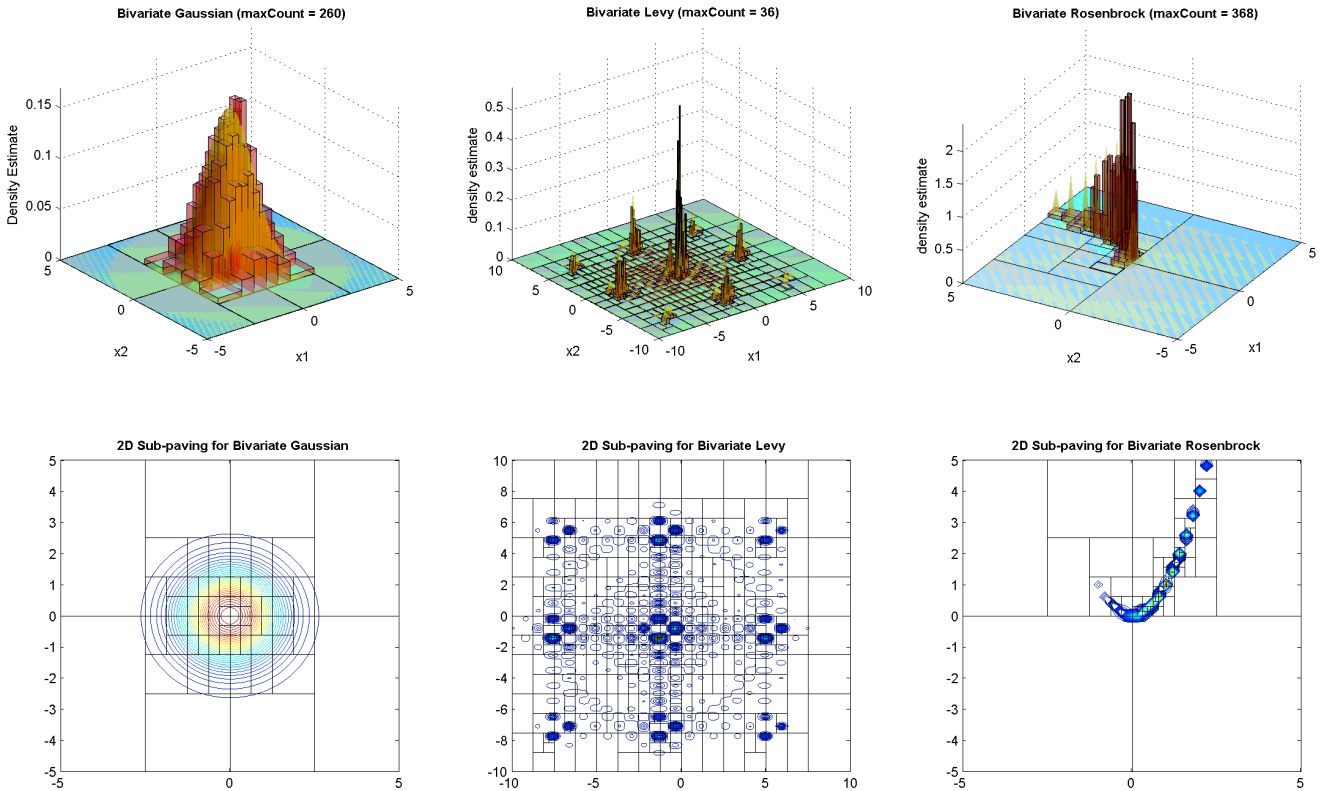



(b) Leaf trace for three chains.

(data drawn from mixture of two 3D Gaussian Densities)

# Some More Examples

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



# Simulations for MCMC and `SplitMostCounts` PQ

MIAE (std. err.) for $n$ samples from approximated 1D-, 2D- and 10D-Gaussian densities, and 2D- and 10D-Rosenbrock densities ($L_1$-minimal Simple function approximation in $\mathbb{S}_\Lambda$).

| | | Standard Gaussian densities | | | Rosenbrock densities | |
|---|---|---|---|---|---|---|
| $\Lambda$ | n | 1D | 2D | 10D | 2D | 10D |
| $10^2$ | $10^2$ | 0.2665 (0.0415) | 0.4856 (0.0491) | 0.1192 (0.0662) | 0.5089 (0.0924) | 0.0323 (0.0511) |
| | $10^3$ | 0.1390 (0.0192) | 0.2558 (0.0127) | 0.0543 (0.0172) | 0.1712 (0.0224) | 0.0095 (0.0191) |
| | $10^4$ | 0.0620 (0.0047) | 0.0992 (0.0067) | 0.0382 (0.0036) | 0.0498 (0.0081) | 0.0025 (0.0050) |
| | $10^5$ | 0.0262 (0.0016) | 0.0279 (0.0019) | 0.0259 (0.0017) | 0.0143 (0.0025) | 0.0009 (0.0015) |
| | $10^6$ | 0.0099 (0.0008) | 0.0086 (0.0006) | 0.0073 (0.0009) | 0.0045 (0.0005) | 0.0004 (0.0005) |
| | $10^7$ | 0.0026 (0.0002) | 0.0027 (0.0003) | 0.0025 (0.0004) | 0.0017 (0.0010) | 0.0001 (0.0003) |
| $10^3$ | $10^2$ | 0.2946 (0.0678) | 0.6046 (0.1299) | 0.1702 (0.0907) | 1.0027 (0.0437) | 0.0323 (0.0511) |
| | $10^3$ | 0.1418 (0.0226) | 0.2973 (0.0174) | 0.0739 (0.0183) | 0.4747 (0.0191) | 0.0039 (0.0075) |
| | $10^4$ | 0.0648 (0.0052) | 0.1586 (0.0067) | 0.0555 (0.0045) | 0.2139 (0.0054) | 0.0013 (0.0028) |
| | $10^5$ | 0.0292 (0.0014) | 0.0768 (0.0016) | 0.0295 (0.0020) | 0.0789 (0.0023) | 0.0004 (0.0006) |
| | $10^6$ | 0.0136 (0.0006) | 0.0297 (0.0006) | 0.0108 (0.0005) | 0.0267 (0.0058) | 0.0001 (0.0002) |
| | $10^7$ | 0.0061 (0.0002) | 0.0091 (0.0003) | 0.0045 (0.0003) | 0.0082 (0.0011) | 0.0001 (0.0002) |
| $10^4$ | $10^2$ | 0.2864 (0.0487) | 0.5508 (0.0590) | 0.5210 (0.0799) | 1.1391 (0.0545) | 0.1941 (0.0820) |
| | $10^3$ | 0.1380 (0.0152) | 0.3301 (0.0120) | 0.2719 (0.0251) | 0.6018 (0.0139) | 0.0791 (0.0223) |
| | $10^4$ | 0.0664 (0.0062) | 0.1736 (0.0038) | 0.1157 (0.0047) | 0.3163 (0.0047) | 0.0391 (0.0041) |
| | $10^5$ | 0.0293 (0.0017) | 0.0957 (0.0014) | 0.0870 (0.0014) | 0.1691 (0.0053) | 0.0209 (0.0021) |
| | $10^6$ | 0.0138 (0.0005) | 0.0495 (0.0005) | 0.0788 (0.0009) | 0.0882 (0.0048) | 0.0123 (0.0012) |
| | $10^7$ | 0.0063 (0.0001) | 0.0244 (0.0008) | 0.0563 (0.0018) | 0.0479 (0.0057) | 0.0096 (0.0017) |

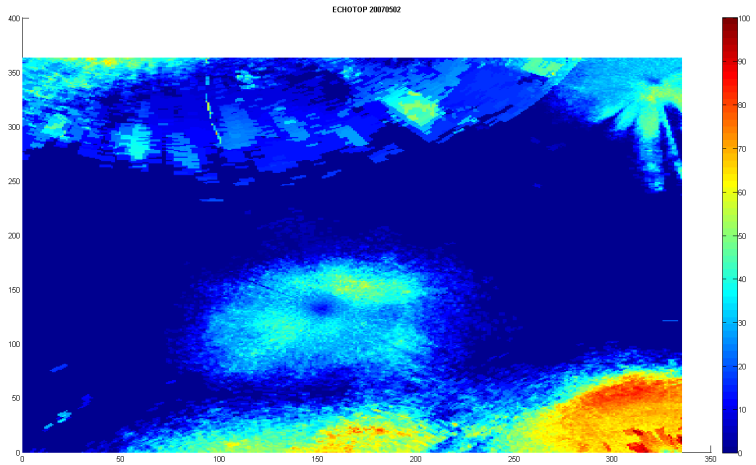
Section 6

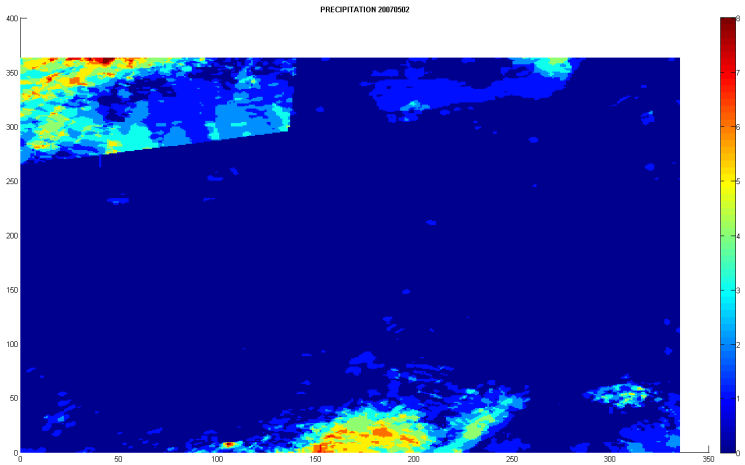
# Real-world Applications

# 12 GBz Data for 59-days of Weather & Air Traffic

## Weather Data over Atlanta, GA – Cloud height & Precipitation

Measured every 1/2-hour and predicted every 5 minutes



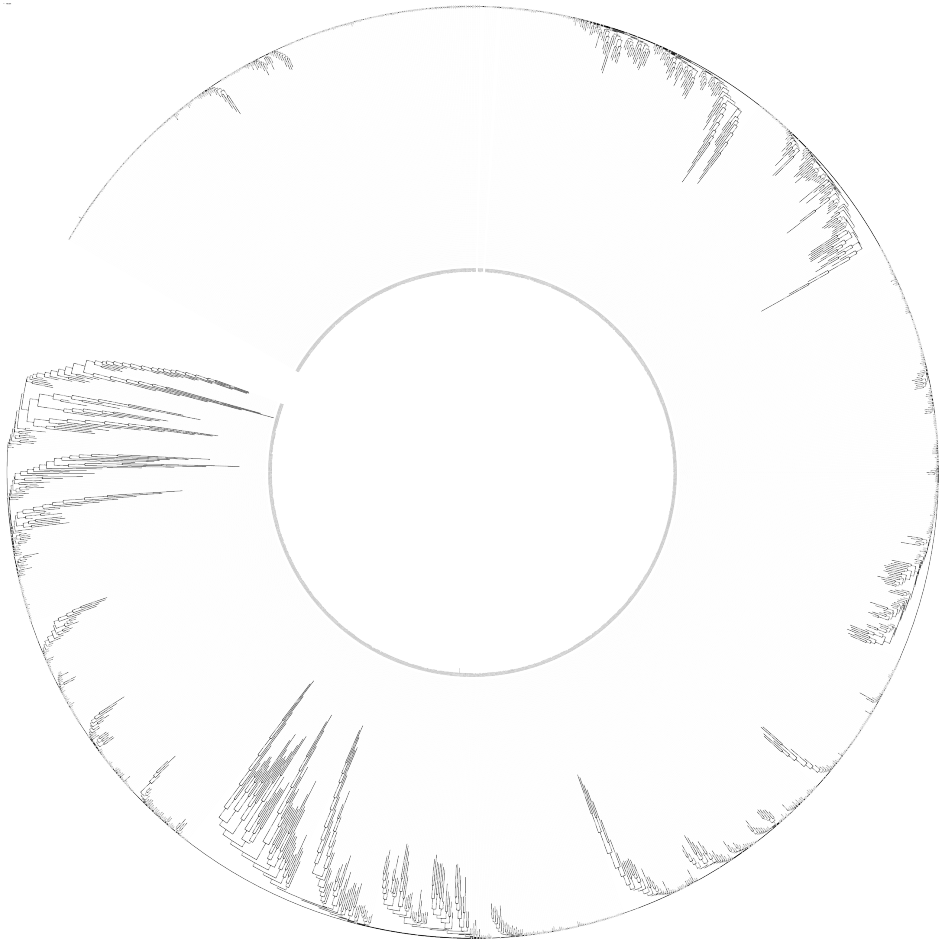




There are $45 \times 59 = 2655$ half-an-hour blocks of weather data.

# 12 GBz Data for 59-days of Weather & Air Traffic

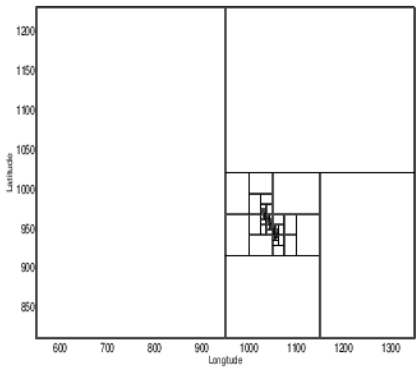
## Weather Tree – Leaves are Time-blocks over 65 days

Neighbor-Joining Tree from Pairwise L1 Distances between Cloud height & Precipitation

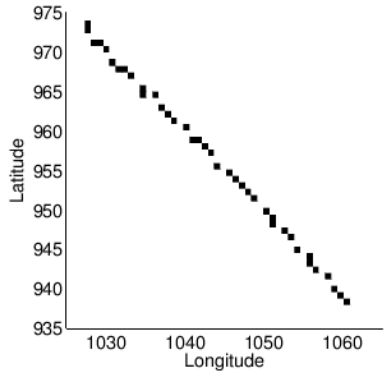


# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

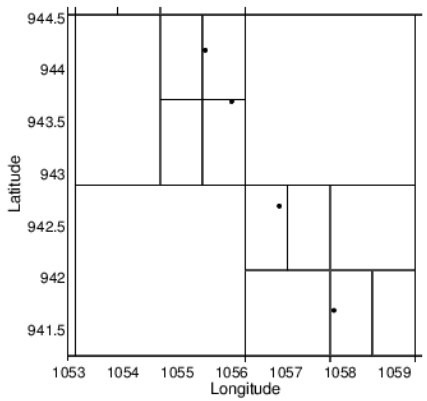
$\mathbb{Z}$-MRP of an aircraft trajectory and its tree
(every 4-6 sec. position data from radar sweep)



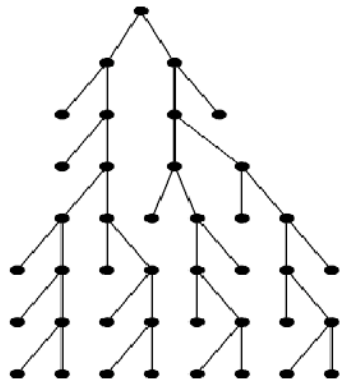

(a) SRP trajectory for aircraft position data.




(b) Shaded boxes in the SRP trajectory.



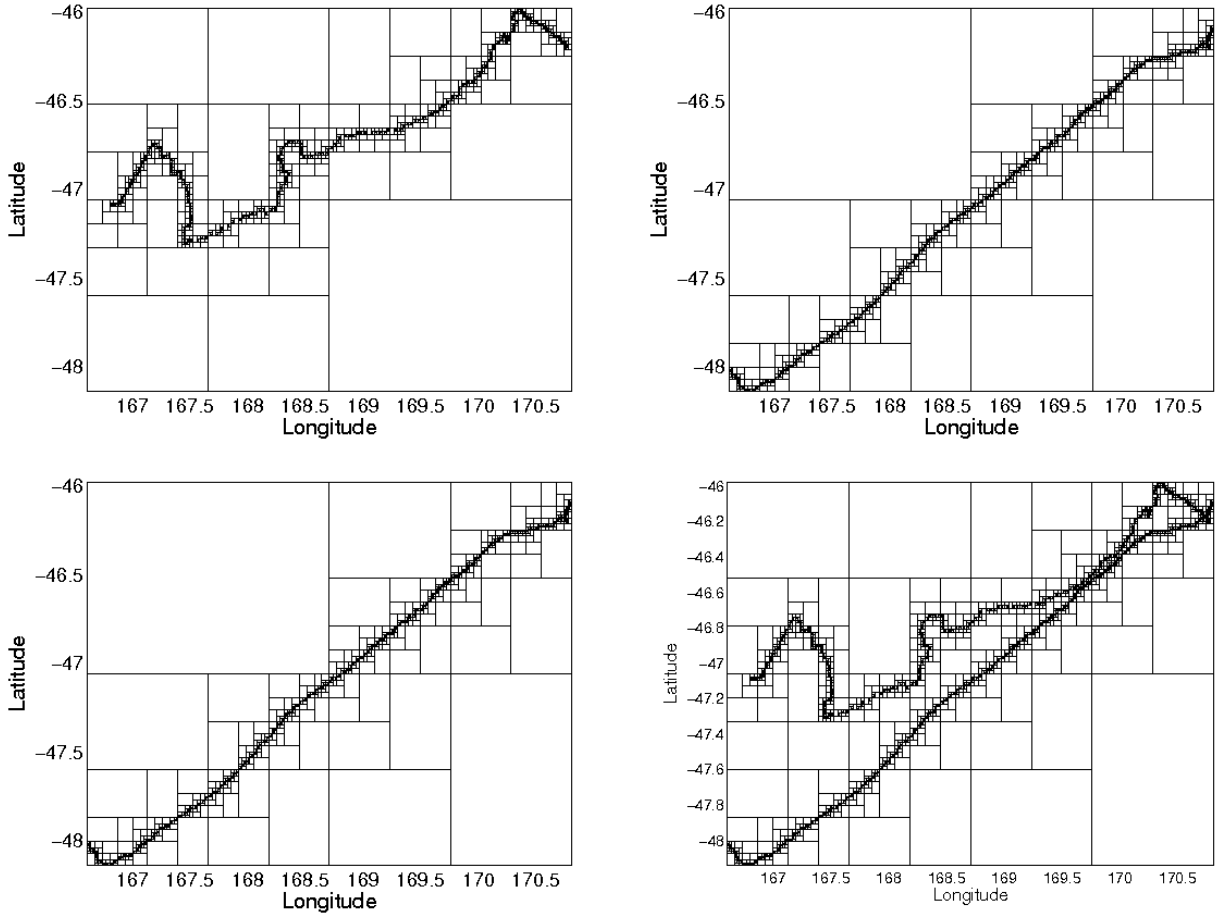

(c) Aircraft positions enclosed by boxes.



(d) The tree corresponding to (c).

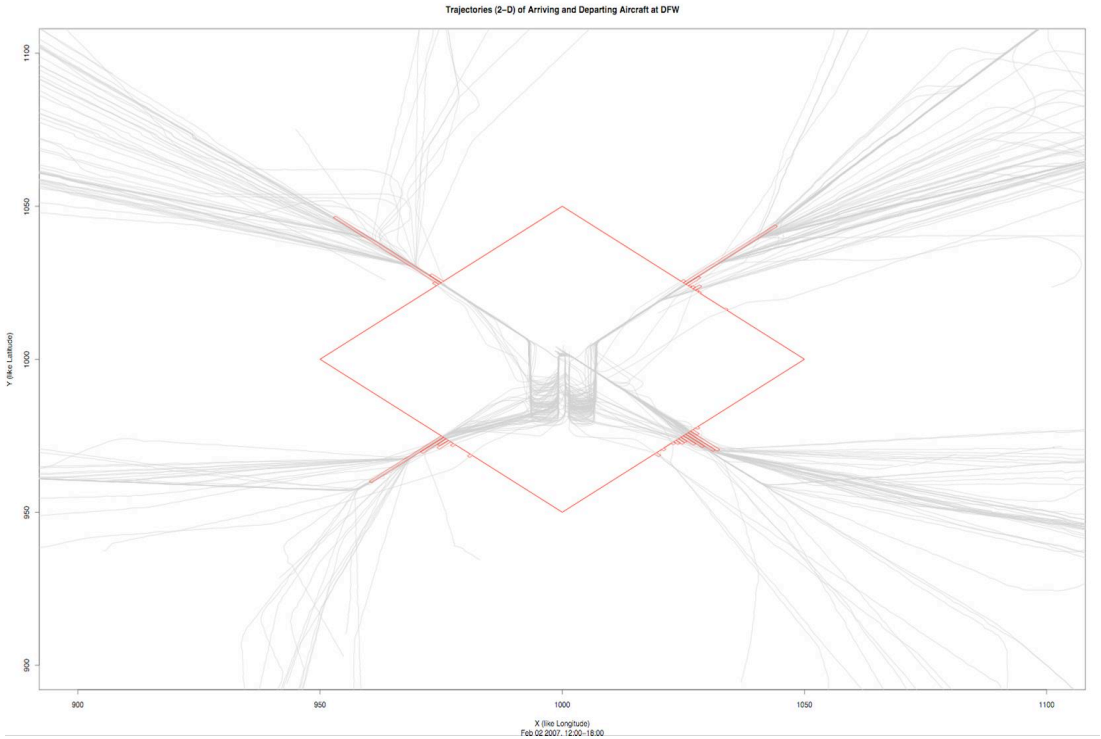
# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

Three individual trajectories and their sum as $\mathbb{Z}$-MRPs



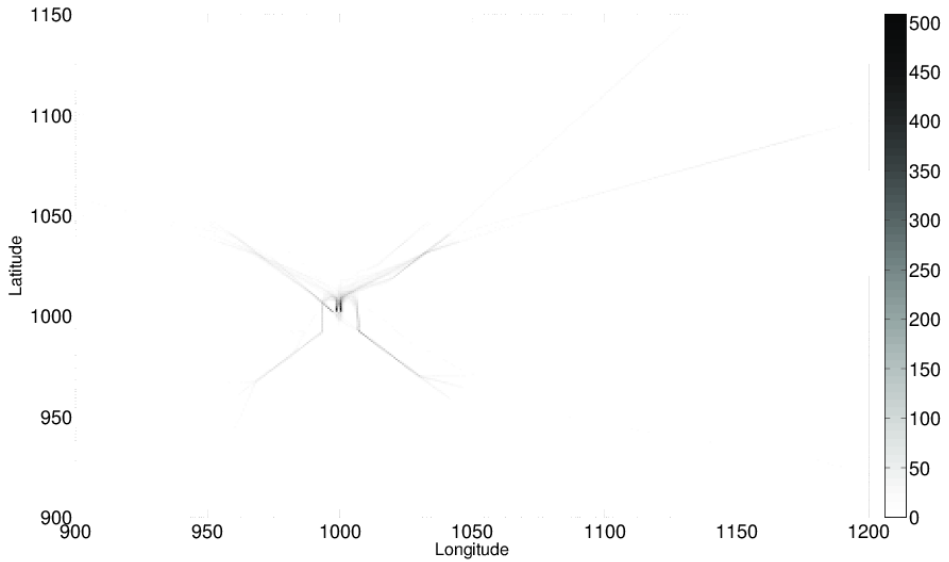
# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

## On a Sunny Day over Atlanta, GA



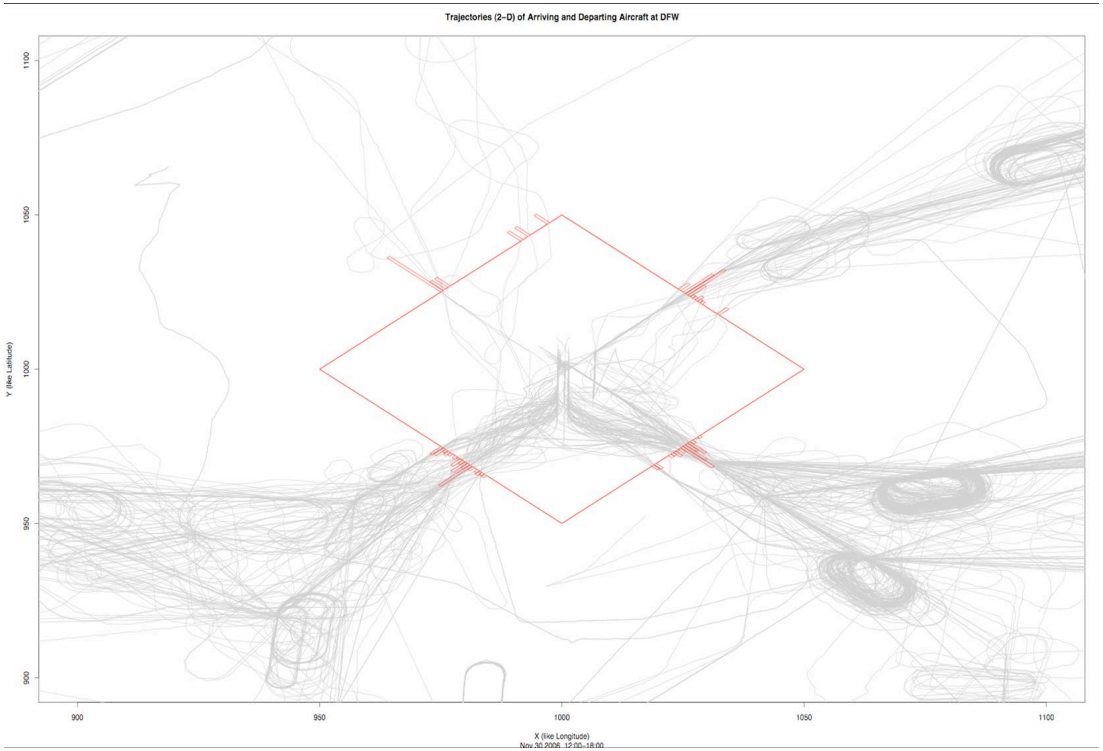
# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

## $\mathbb{Z}$-MRP On this Sunny Day



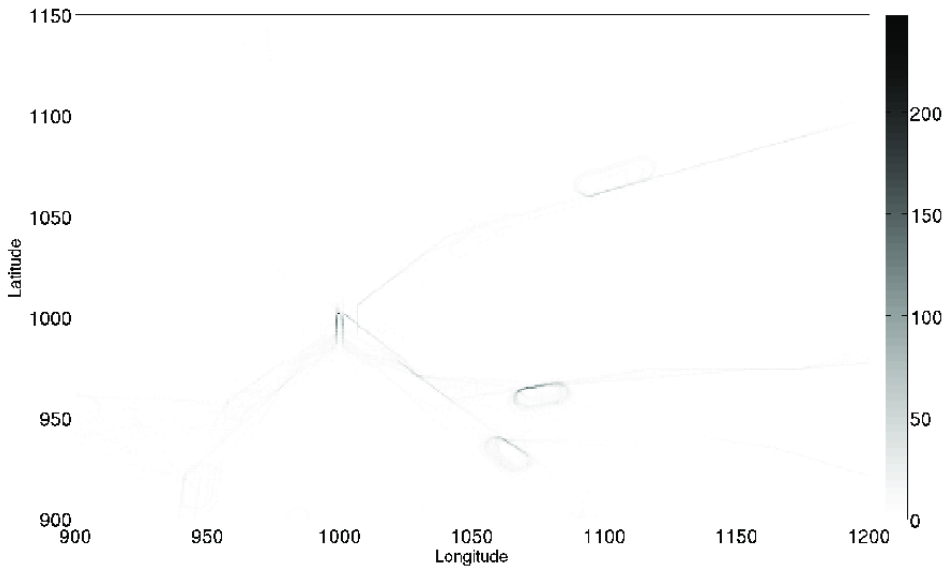
# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

## On a Stormy Day over Altanta, GA



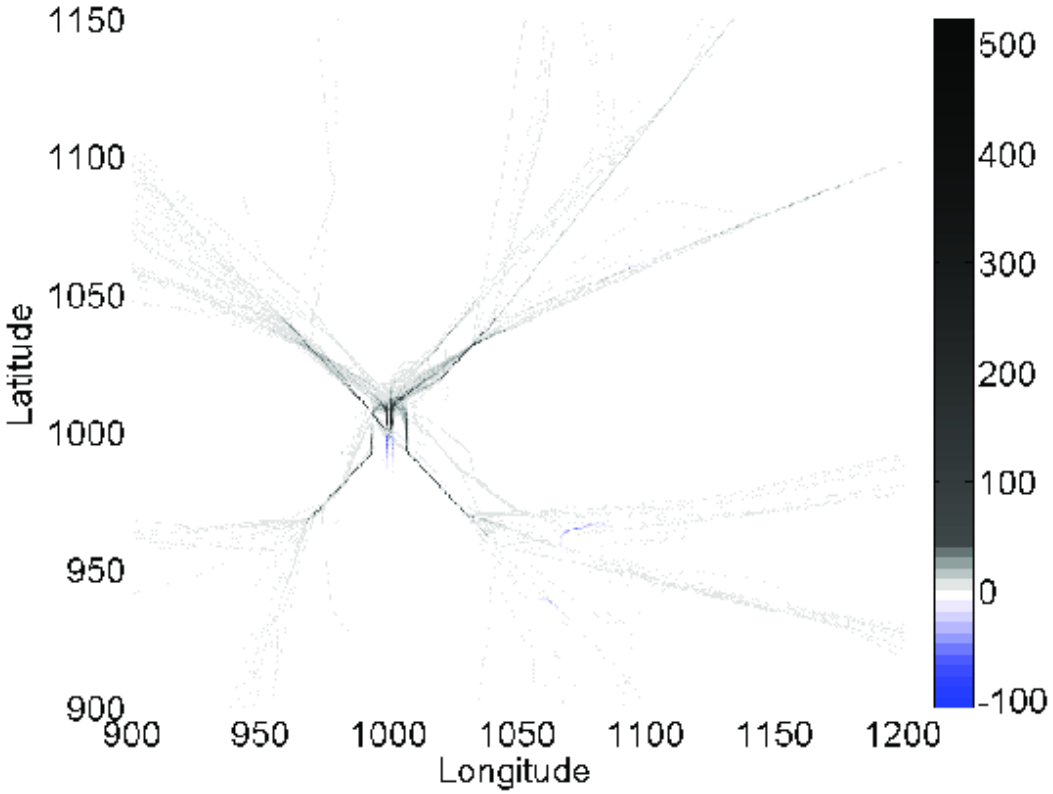
# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

## $\mathbb{Z}$-MRP On this Stormy Day



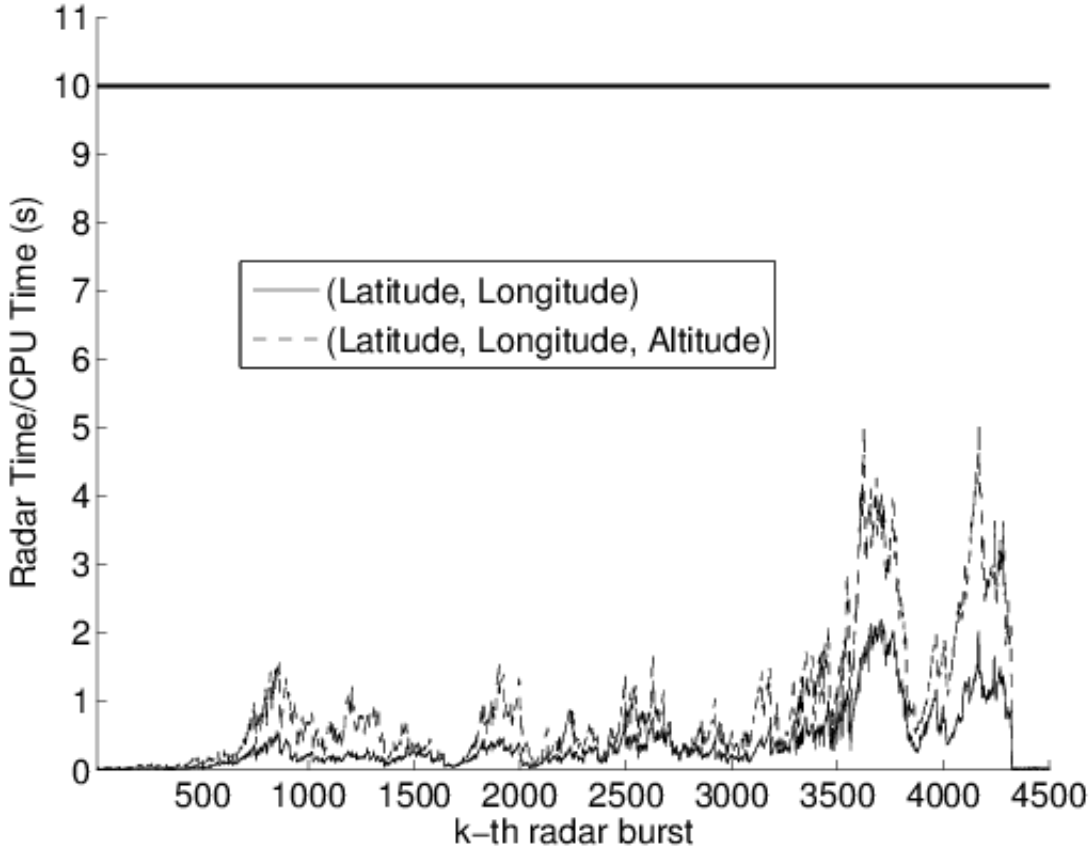
# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

## $\mathbb{Z}$-MRP pattern for Sunny Day $-$ Stormy Day

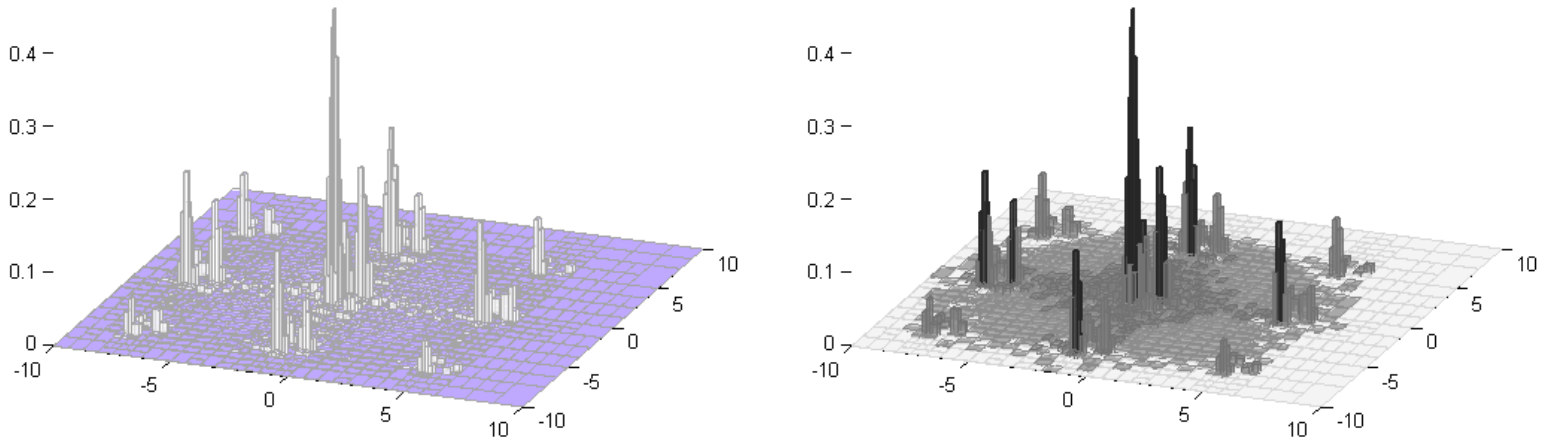


# Dynamic $\mathbb{Z}$-MRPs for 12GB-z (59-days of Air Traffic)

## $\mathbb{Z}$-MRP Dynamic Trees Can be Created in Real-time

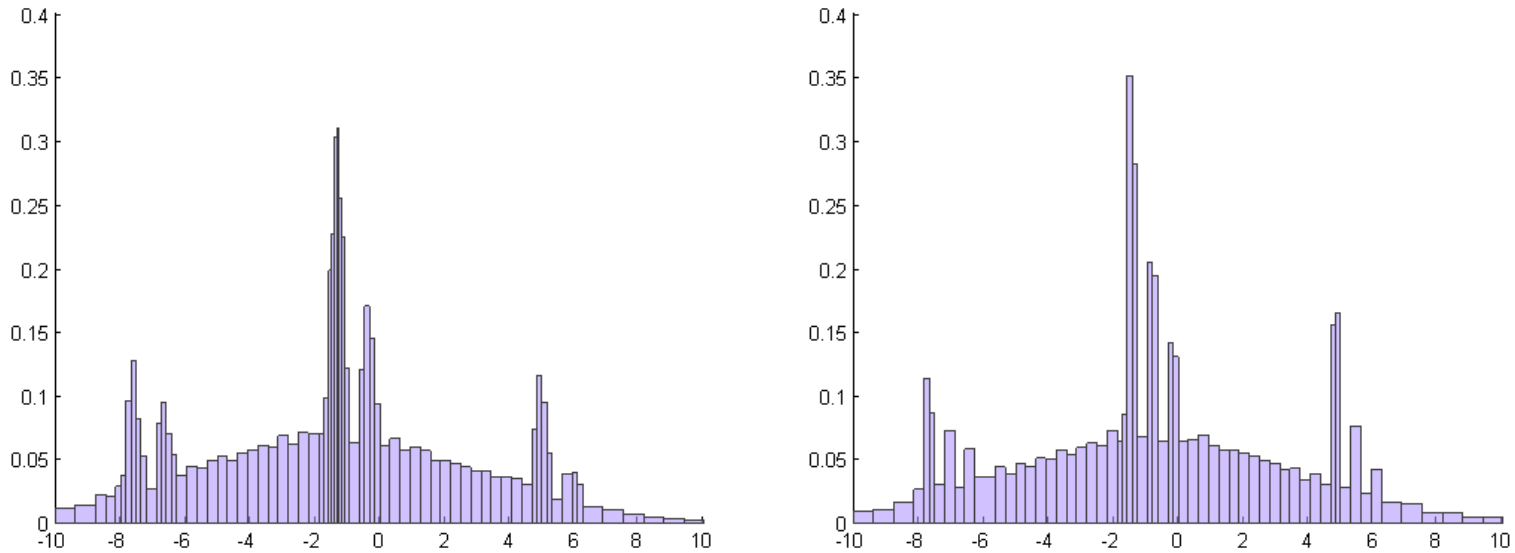


# Coverage, Marginal & Slice Operators of $\mathbb{R}$-MRP



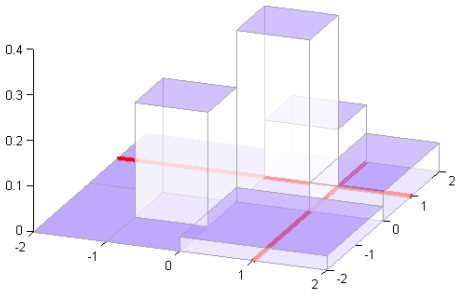

$\mathbb{R}$-MRP approximation to Levy density and its coverage regions with $\alpha = 0.9$ (light gray), $\alpha = 0.5$ (dark gray) and $\alpha = 0.1$ (black)

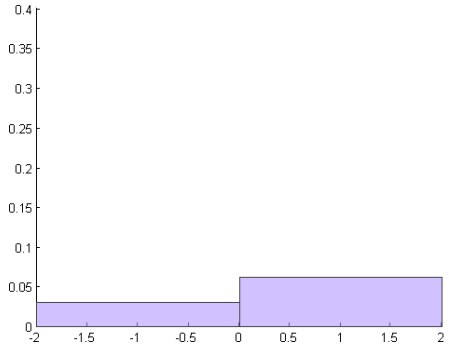
# Coverage, Marginal & Slice Operators of $\mathbb{R}$-MRP

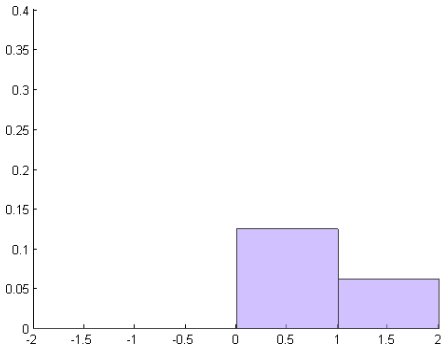



Marginal densities $f^{\{1\}}(x_1)$ and $f^{\{2\}}(x_2)$ along each coordinate of $\mathbb{R}$-MRP approximation (as a tree algorithm in kD)

# Coverage, Marginal & Slice Operators of $\mathbb{R}$-MRP




The slices of a simple $\mathbb{R}$-MRP in 2D

(as a tree algorithm in kD)

# 2014/15 R&D Grant $\longrightarrow$ Python bindings for our C++ Library

MRS 1.0: A C++ Class Library for Statistical Set Processing, Harlow, S & York, 2013

MRS 1.0 is GNU auto-confiscated, Doxygenized, GPL-licensed (needs GNU Sci. Lib., C-XSC & Boost) and has:

54 directories, 432 files

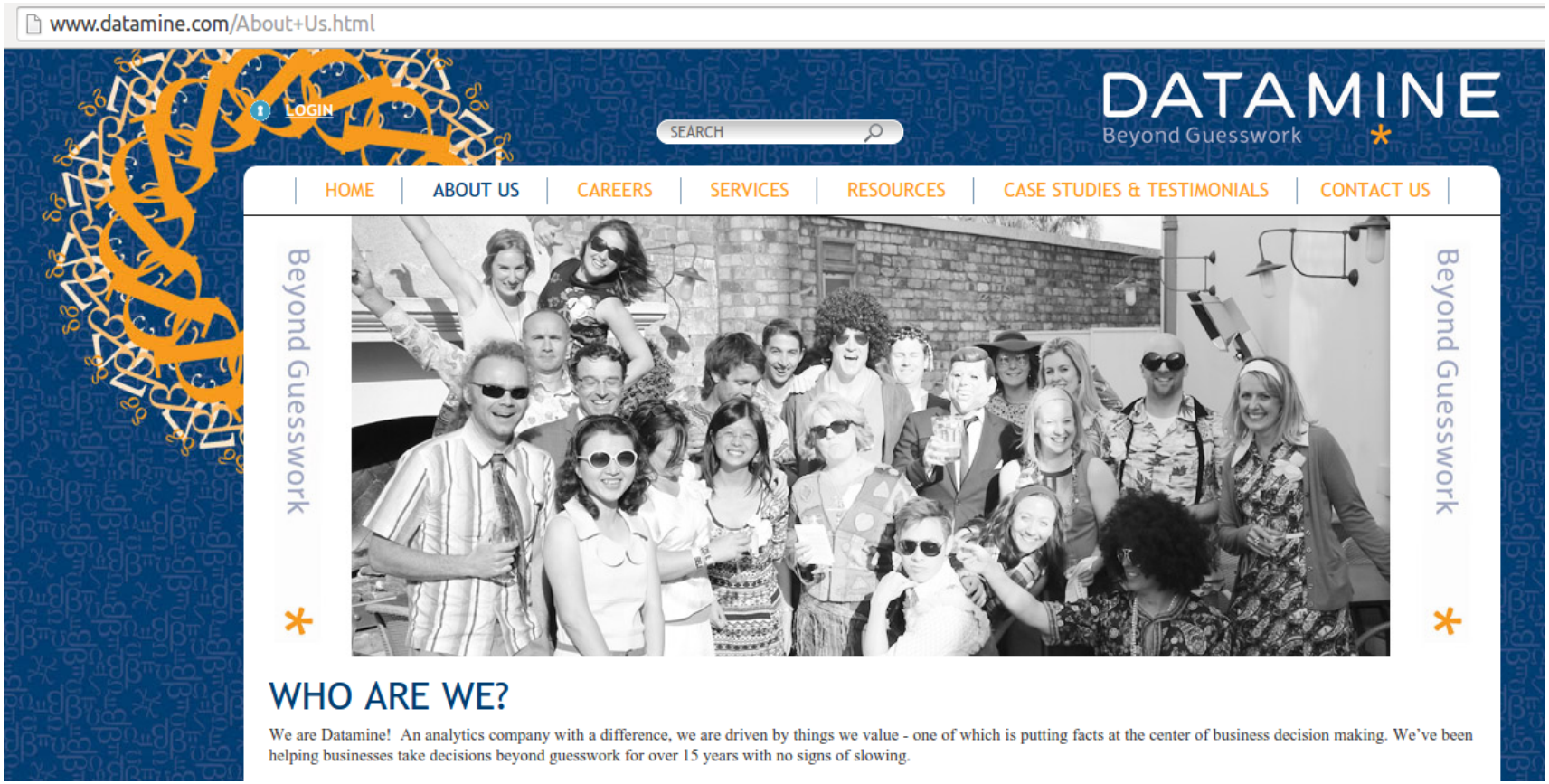
| Language | files | blank | comment | code |
|---|---|---|---|---|
| C++ | 148 | 13274 | 10488 | 38738 |
| C/C++ Header | 123 | 8366 | 19634 | 9012 |
| Bourne Shell | 6 | 1137 | 1191 | 7872 |
| MATLAB | 14 | 522 | 103 | 1585 |
| m4 | 3 | 149 | 20 | 1237 |
| CSS | 1 | 173 | 55 | 721 |
| make | 51 | 106 | 72 | 493 |
| HTML | 2 | 1 | 18 | 46 |
| Bourne Again Shell | 2 | 0 | 2 | 4 |
| SUM: | 350 | 23728 | 31583 | 59708 |

# 2014/15 R&D Grant $\longrightarrow$ Python bindings for our C++ Library

MRS 1.0: A C++ Class Library for Statistical Set Processing, Harlow, S & York, 2013

Callaghan Innovation Award (NZ Ministry of Buisness Innovation & Employment)

with Datamine Ltd. (winner of NZ Innovators Award 2014) that is focussed on *high-end data analytics*



# Choice of $k_n$ – Prior Selection by CV

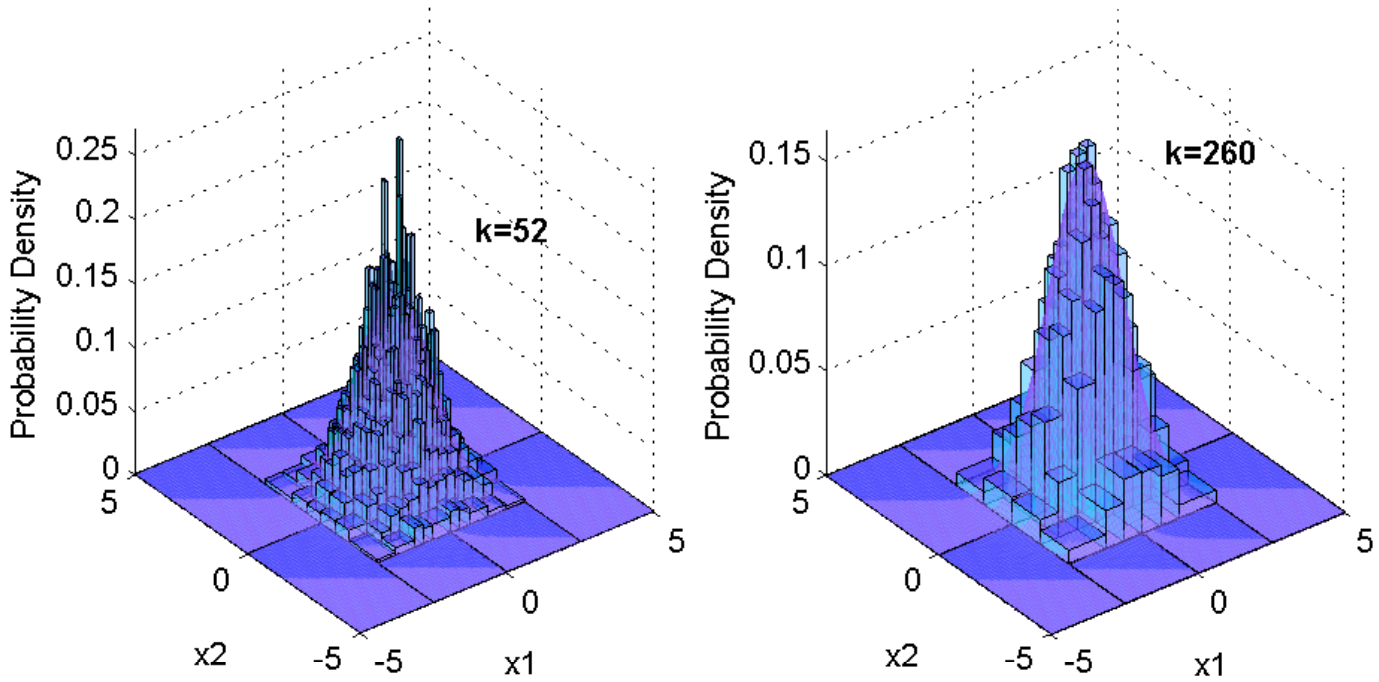



Figure : Two histogram density estimates for the standard bivariate gaussian density with different choices of $k_n$. The histogram is under-smoothed when $k_n$ is relatively smaller than $n$ and over-smoothed when $k_n$ is relatively larger.

# Finding image of $\mathbb{R}$-MRP is by fast look-ups

**Algorithm 3:** `PointWiseImage`$(\rho, x)$

**input** : $\rho$ with box $\boldsymbol{x}_\rho$, the root node of $\mathbb{R}$-MRP $f$ with RP $s$, and a point $x \in \boldsymbol{x}_\rho$.
**output** : Return $f_{\eta(x)}$ at the leaf node $\eta(x)$ that is associated with the box $\boldsymbol{x}_{\eta(x)}$ containing $x$.

```
if IsLeaf(ρ) then
    return f_ρ
end
else
    if x ∈ x_ρR then
        PointWiseImage(ρR, x)
    end
    else
        PointWiseImage(ρL, x)
    end
end
```

# Finding image of $\mathbb{R}$-MRP is by fast look-ups

**Algorithm 4:** `PointWiseImage`$(\rho, x)$

**input** : $\rho$ with box $\boldsymbol{x}_\rho$, the root node of $\mathbb{R}$-MRP $f$ with RP $s$, and a point $x \in \boldsymbol{x}_\rho$.
**output** : Return $f_{\eta(x)}$ at the leaf node $\eta(x)$ that is associated with the box $\boldsymbol{x}_{\eta(x)}$ containing $x$.

```
if IsLeaf(ρ) then
    return f_ρ
end
else
    if x ∈ x_ρR then
        PointWiseImage(ρR, x)
    end
    else
        PointWiseImage(ρL, x)
    end
end
```

- Cost of KDE image $\sim O(n)$ **KFLOPs** (FLOPs for kernel evaluation procedure)

# Finding image of $\mathbb{R}$-MRP is by fast look-ups

**Algorithm 5:** `PointWiseImage`$(\rho, x)$

**input** : $\rho$ with box $\boldsymbol{x}_\rho$, the root node of $\mathbb{R}$-MRP $f$ with RP $s$, and a point $x \in \boldsymbol{x}_\rho$.
**output** : Return $f_{\eta(x)}$ at the leaf node $\eta(x)$ that is associated with the box $\boldsymbol{x}_{\eta(x)}$ containing $x$.

```
if IsLeaf(ρ) then
    return f_ρ
end
else
    if x ∈ x_ρR then
        PointWiseImage(ρR, x)
    end
    else
        PointWiseImage(ρL, x)
    end
end
```

- Cost of KDE image $\sim O(n)$ **KFLOPs** (FLOPs for kernel evaluation procedure)
- 10-fold CV cost $\sim 10 \times O\left(\frac{1}{10}n\frac{9}{10}n\right) = O(n^2)$ **KFLOPs**

# Finding image of $\mathbb{R}$-MRP is by fast look-ups

**Algorithm 6:** `PointWiseImage`$(\rho, x)$

**input** : $\rho$ with box $\boldsymbol{x}_\rho$, the root node of $\mathbb{R}$-MRP $f$ with RP $s$, and a point $x \in \boldsymbol{x}_\rho$.
**output** : Return $f_{\eta(x)}$ at the leaf node $\eta(x)$ that is associated with the box $\boldsymbol{x}_{\eta(x)}$ containing $x$.

```
if IsLeaf(ρ) then
    return f_ρ
end
else
    if x ∈ x_ρR then
        PointWiseImage(ρR, x)
    end
    else
        PointWiseImage(ρL, x)
    end
end
```

- Cost of KDE image $\sim O(n)$ **KFLOPs** (FLOPs for kernel evaluation procedure)
- 10-fold CV cost $\sim 10 \times O\left(\frac{1}{10}n\frac{9}{10}n\right) = O(n^2)$ **KFLOPs**
- But using $\mathbb{R}$-MRP approximation to KDE requires $10 \times O\left(\frac{1}{10}n \lg\left(\frac{9}{10}n\right)\right) = O(n \lg(n))$ **tree-look-ups**

# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

Consider the following two block-specific edge probability matrices:

$$A(\alpha) = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \end{bmatrix} \quad , \quad B(\alpha) = \begin{bmatrix} 0.42 & 0.42 \\ 0.42 & \alpha \end{bmatrix}$$

and the following two prior block probabilities:

$$\pi = (0.6, 0.4) \quad \text{and} \quad \pi' = (0.4, 0.6)$$

# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

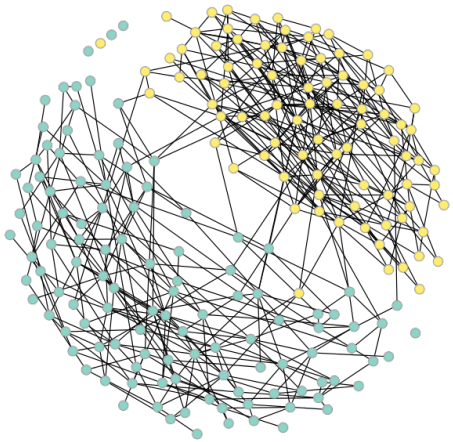
Consider the following two block-specific edge probability matrices:

$$A(\alpha) = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \end{bmatrix} \quad , \quad B(\alpha) = \begin{bmatrix} 0.42 & 0.42 \\ 0.42 & \alpha \end{bmatrix}$$

and the following two prior block probabilities:

$$\pi = (0.6, 0.4) \quad \text{and} \quad \pi' = (0.4, 0.6)$$

Consider 9 random graph bursts from $SBM(A(0.999), \pi, 200)$ but the 5-th anomalous burst from $SBM(B(0.4), \pi, 200)$ – easy to eye-ball anomaly



# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

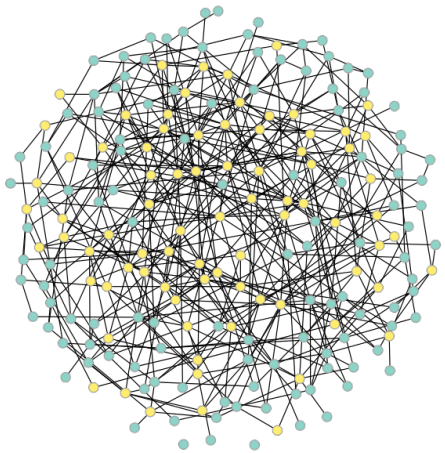
Consider the following two block-specific edge probability matrices:

$$A(\alpha) = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \end{bmatrix} \quad , \quad B(\alpha) = \begin{bmatrix} 0.42 & 0.42 \\ 0.42 & \alpha \end{bmatrix}$$

and the following two prior block probabilities:

$$\pi = (0.6, 0.4) \quad \text{and} \quad \pi' = (0.4, 0.6)$$

Consider 9 random graph bursts from $SBM(B(0.5), \pi, 200)$ but the 5-th anomalous burst from $SBM(B(0.4), \pi, 200)$ – not easy to eye-ball anomaly



# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

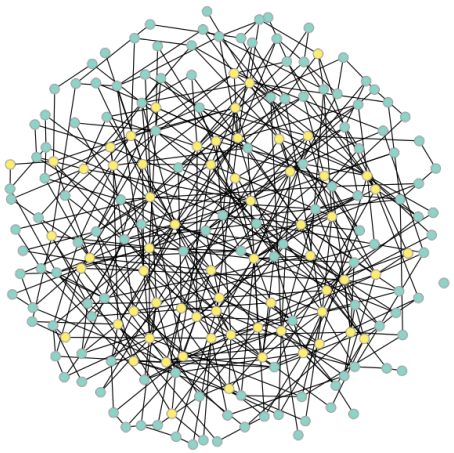
Consider the following two block-specific edge probability matrices:

$$A(\alpha) = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \end{bmatrix} \quad , \quad B(\alpha) = \begin{bmatrix} 0.42 & 0.42 \\ 0.42 & \alpha \end{bmatrix}$$

and the following two prior block probabilities:

$$\pi = (0.6, 0.4) \quad \text{and} \quad \pi' = (0.4, 0.6)$$

Consider 9 random graph bursts from $SBM(B(0.5), \pi, 200)$ but the 5-th anomalous burst from $SBM(B(0.5), \pi', 200)$ – not easy to eye-ball anomaly



# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

We use $\mathbb{R}$-MRP based (10-fold CV) "prior selection" $\pi(s) \propto \exp(-t \times \#\textit{leaves})$ to estimate densities from a low-dimensional point-cloud obtained from the Eigen decomposition of the adjacency matrix of each graph.

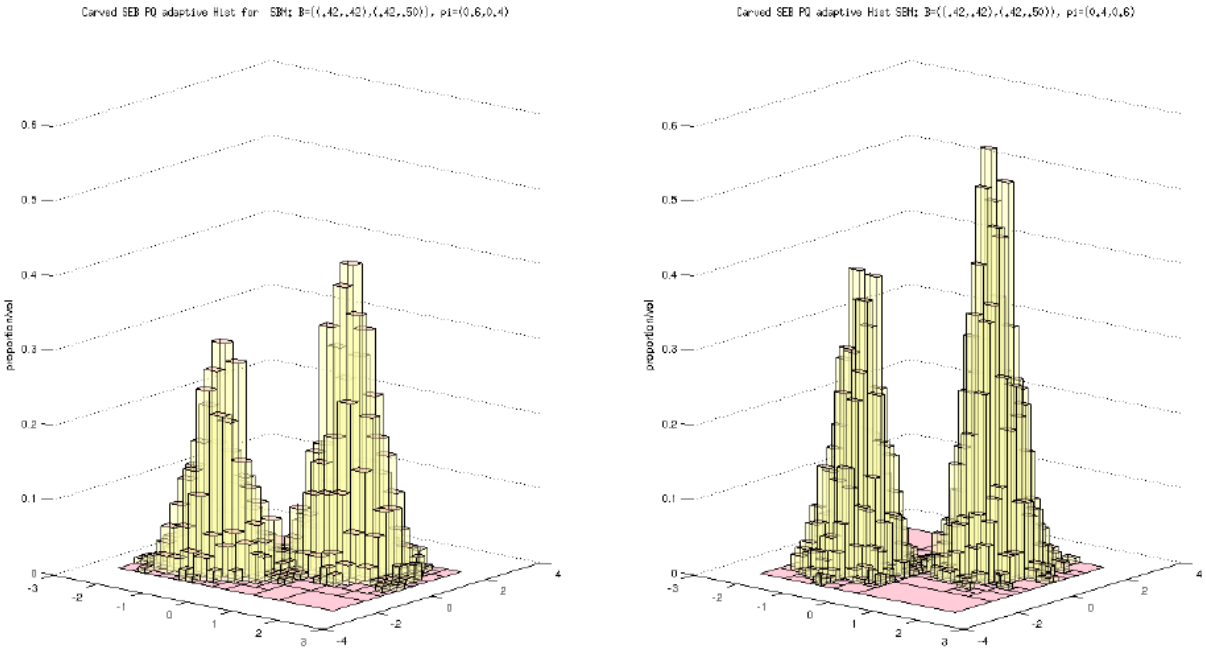



Fig. 1. Density estimates of the embedding of $SBM(B, \pi, n = 3000)$ and $SBM(B, \pi', n = 3000)$.

# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

We use $\mathbb{R}$-MRP based $L_1$ computations between all pairs of graph densities.

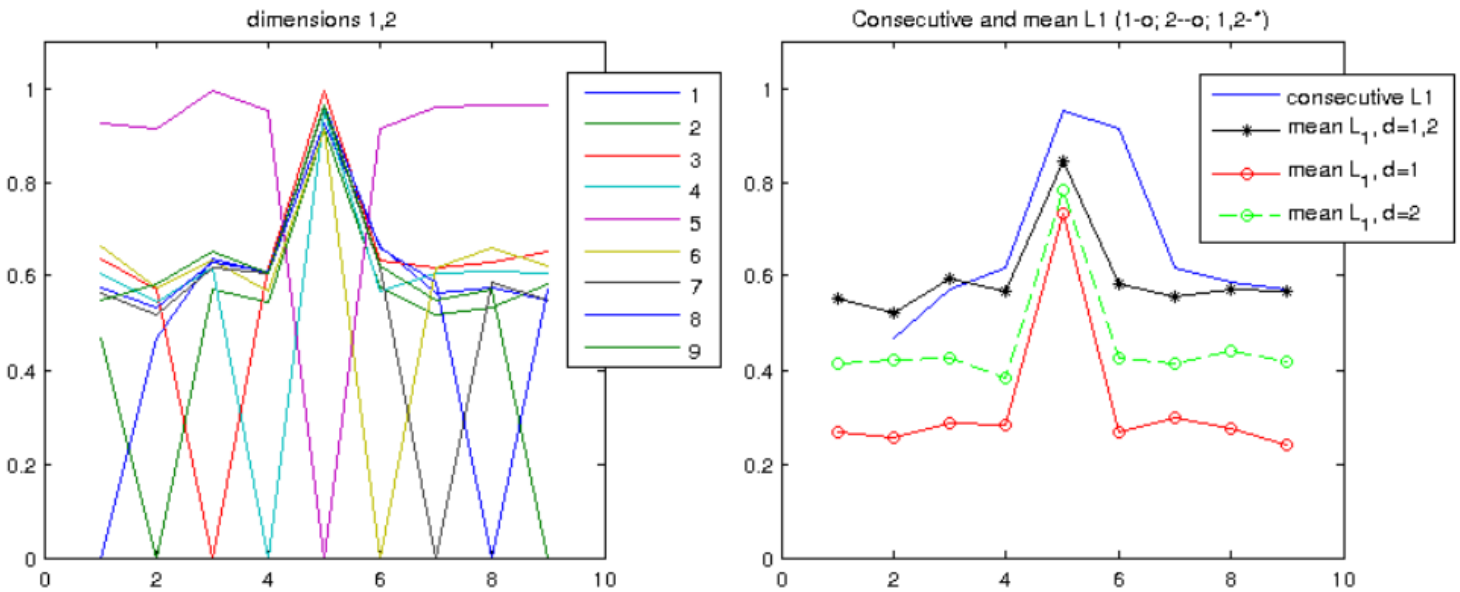
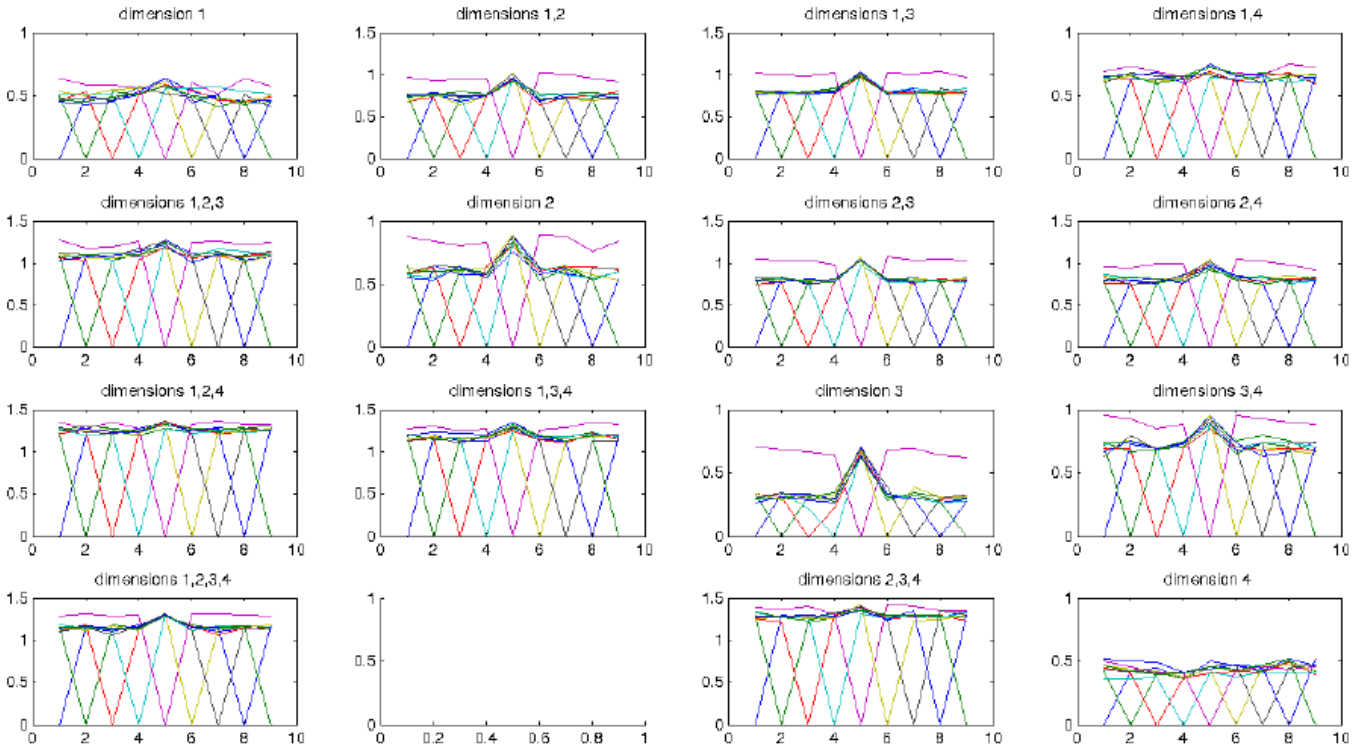



Fig. 2. Pairwise $L_1$ distances (left) and mean and consecutive $L_1$ distances (right) between the nine bursts (dimensions 1 and 2) in Scenario C5 with $n = 5000, d = K = 2, m = 500$.

# Anomaly Detection in Graph Time Series (joint with Carey E. Priebe)

We use $\mathbb{R}$-MRP based $L_1$ computations between all marginal densities of each pair of joint densities (4 blocks).



Section 7

# Conclusions and References

# Conclusions

- Statistical Regular Paving (SRP) is a stat. sufficient data-adaptive structure for density estimation

# Conclusions

- Statistical Regular Paving (SRP) is a stat. sufficient data-adaptive structure for density estimation
- Arithmetic is efficiently extended through $\mathbb{R}$-MRPs

# Conclusions

- Statistical Regular Paving (SRP) is a stat. sufficient data-adaptive structure for density estimation
- Arithmetic is efficiently extended through $\mathbb{R}$-MRPs
- Combining PQ-based ($L_1$-consistent) initialization + Bayesian MCMC is powerful

# Conclusions

- Statistical Regular Paving (SRP) is a stat. sufficient data-adaptive structure for density estimation
- Arithmetic is efficiently extended through $\mathbb{R}$-MRPs
- Combining PQ-based ($L_1$-consistent) initialization + Bayesian MCMC is powerful
- Further decisions can be made with appropriate $\mathbb{R}$-MRP *arithmetic* (regression, anomaly detection, etc.)

# Conclusions

- Statistical Regular Paving (SRP) is a stat. sufficient data-adaptive structure for density estimation
- Arithmetic is efficiently extended through $\mathbb{R}$-MRPs
- Combining PQ-based ($L_1$-consistent) initialization + Bayesian MCMC is powerful
- Further decisions can be made with appropriate $\mathbb{R}$-MRP *arithmetic* (regression, anomaly detection, etc.)
- Future 1: MDE over Yatracoss Classes of SRP histograms

($L_1$ School's Universal Performance Guarantees)

# Conclusions

- Statistical Regular Paving (SRP) is a stat. sufficient data-adaptive structure for density estimation
- Arithmetic is efficiently extended through $\mathbb{R}$-MRPs
- Combining PQ-based ($L_1$-consistent) initialization + Bayesian MCMC is powerful
- Further decisions can be made with appropriate $\mathbb{R}$-MRP *arithmetic* (regression, anomaly detection, etc.)
- Future 1: MDE over Yatracoss Classes of SRP histograms

  ($L_1$ School's Universal Performance Guarantees)
- Future 2: Arithmetic over data-partitioning Ball Trees (to

  overcome the curse of space-partitioning trees for structured data = high-dim. data on low-dim. manifolds)

# References

Devroye, L., and Lugosi, G. (2001). *Combinatorial methods in density estim.*, Springer.

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

Lavine, M. (1992). Some aspects of Polya tree distributions for statistical modelling. *Ann. Stat.*. **20**, 1222–1235.

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

Gray, A. G. & Moore, A. W. (2003). Nonparametric density estimation: toward computational tractability. In *Proc. of the SIAM Intl. Conf. on Data Mining*.

Harlow, J., Sainudiin, R. & Tucker, W. (2012). Mapped Regular Pavings, *Reliable Computing*, **16**, 252–282.

Teng G., Kuhn, K. and Sainudiin, R. (2012). Statistical regular pavings to analyze massive data of aircraft trajectories, *J. of Aerospace Comp. Inf. Commun.*, **9**:1, 14–25.

Sainudiin, R., Teng G., Harlow, J., and Lee D. (2013). Posterior expectation of regularly paved random histograms, *ACM Trans. on Model. Comp. Simul.*, **23**: 1, Article 6, 20 pp.

Sainudiin, R. & York, T. (2013). An auto-validating trans-dimensional universal rejection sampler for locally Lipschitz arithmetical expressions, *Reliable Computing*, **18**, 15–54.

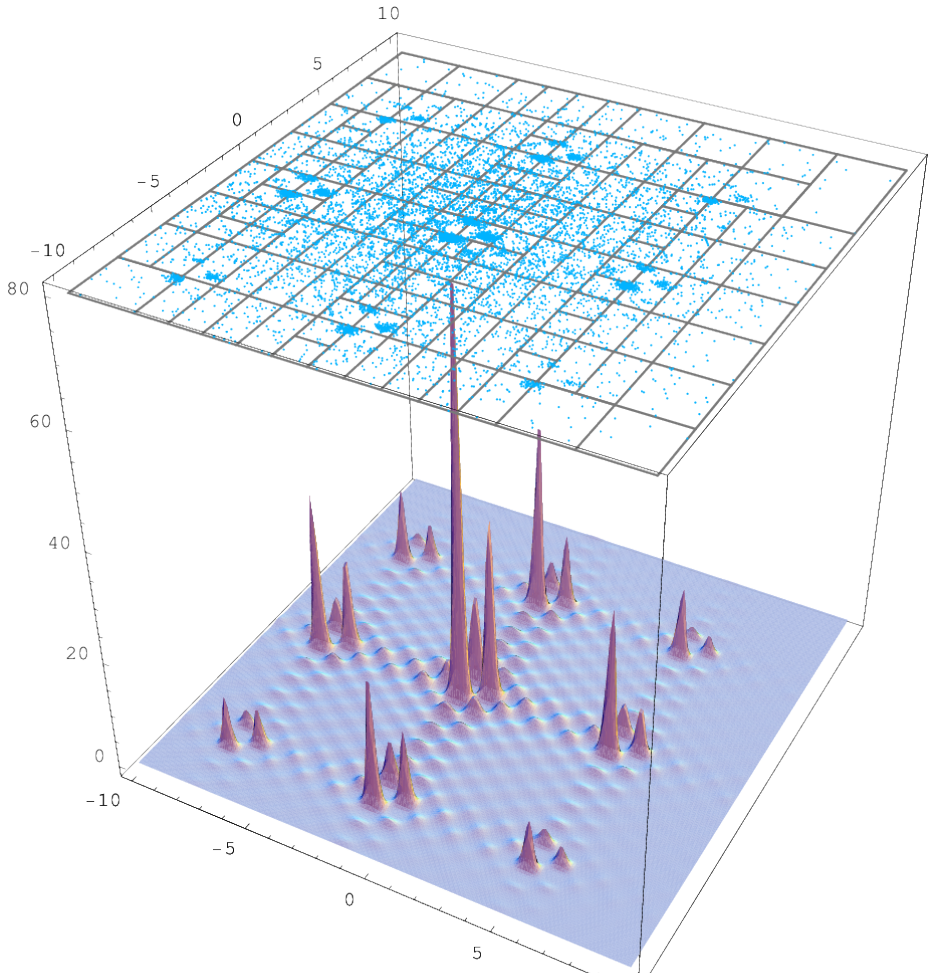
# Acknowledgements

- RS's external consulting revenues from the New Zealand Ministry of Tourism
- WT's Swedish Research Council Grant 2008-7510 that enabled RS's visits to Uppsala in 2006, 2009, 2012
- Erskine grant from University of Canterbury that enabled WT's visit to Christchurch in 2011 & 2014
- University of Canterbury MSc Scholarship to JH.
- *Correctness by Construction*, 7th Framework Prog. of the EU, Marie Curie Actions-People, International Research Staff Exchange Scheme (IRSES), 2014 - 2017 (counter-part funds from Royal Soc. of NZ)

Thank you!

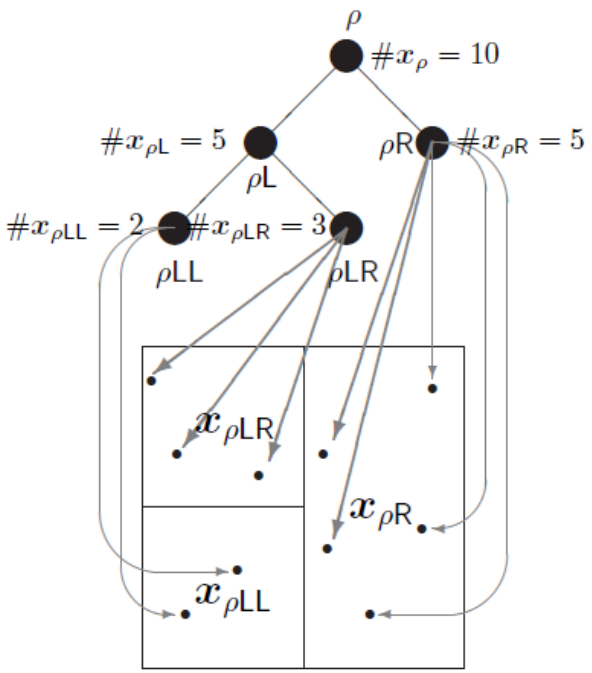
# Nonparametric Density Estimation – Our Approach

Problem: Take samples from an unknown density $f$ and consistently reconstruct $f$

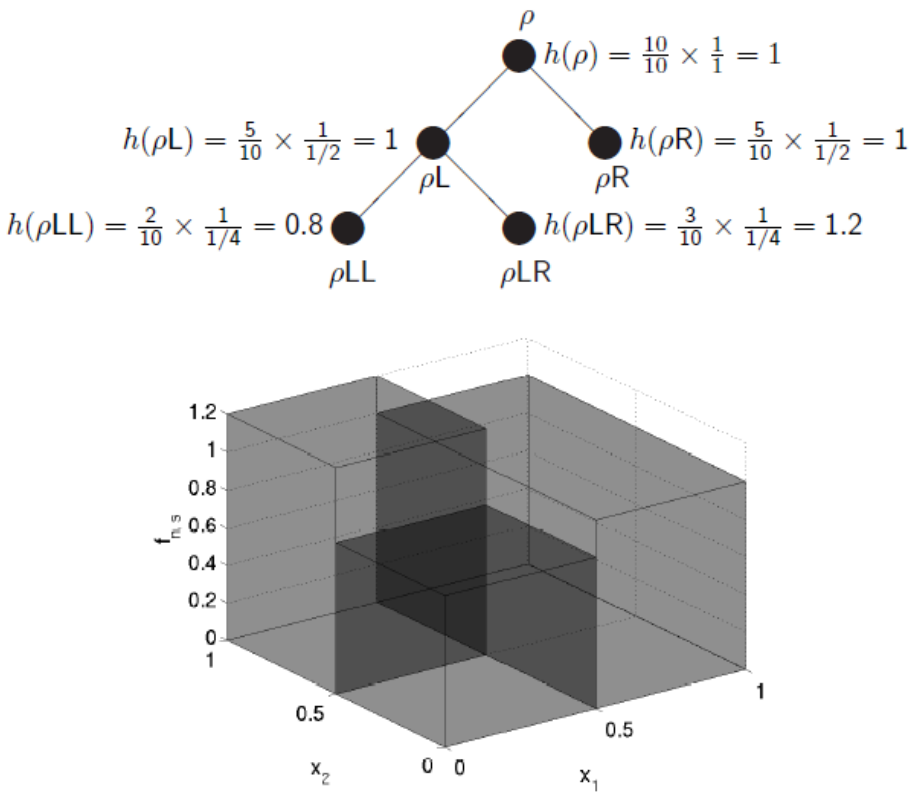


# Nonparametric Density Estimation – Our Approach

Use statistical regular paving to get $\mathbb{R}$-MRP data-adaptive histogram




(a) An SRP tree and its constituents.



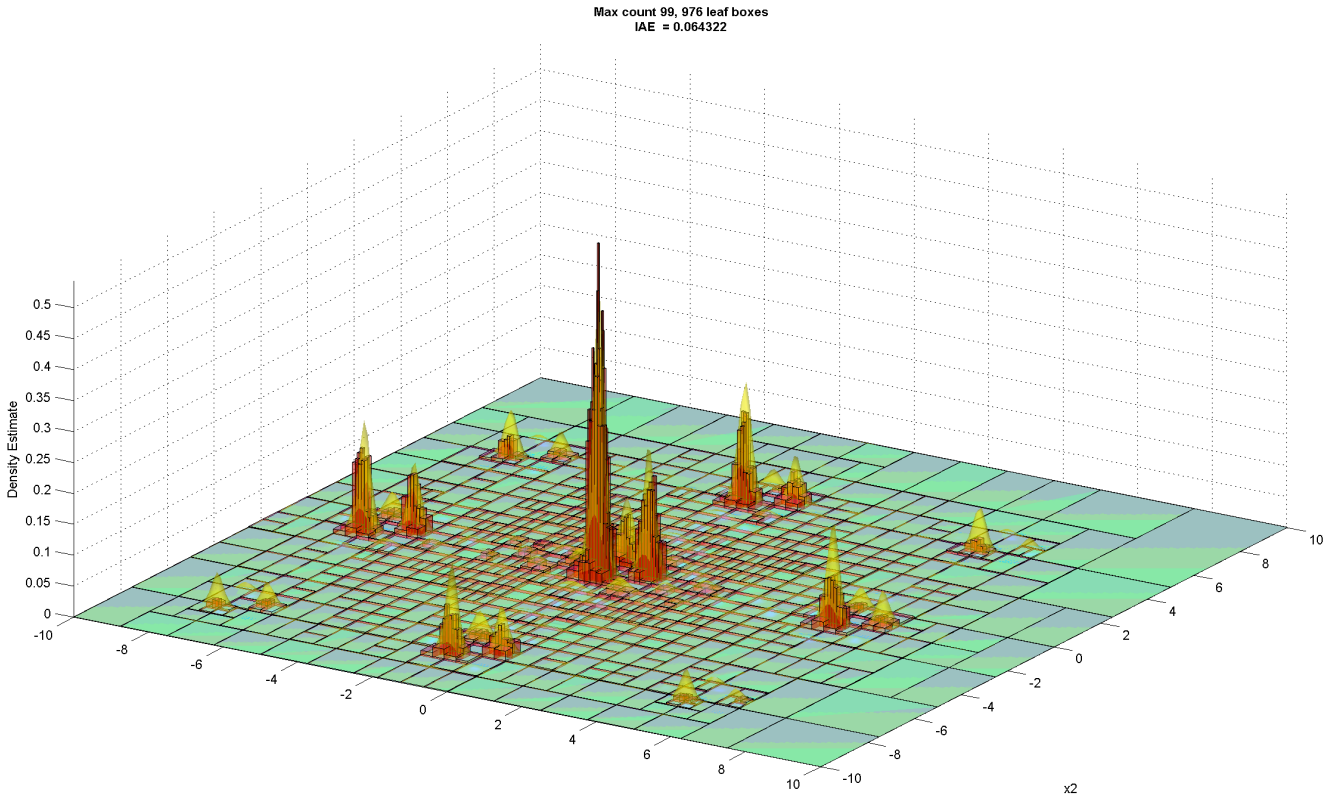

(b) An SRP histogram and its tree.

# Nonparametric Density Estimation – Our Approach

$\mathbb{R}$-MRP histogram averaging produces Bayesian posterior mean estimate of $f$ when initialized using SEB Randomized PQ

Structured $f$ in up to 10 D with $n \in [10^4, 10^7]$

(Teng, Harlow, Lee and S., *ACM Trans. Mod. & Comp. Sim.*, 2013)



# KDE (diagonal badwidth) Vs. SRP MCMC

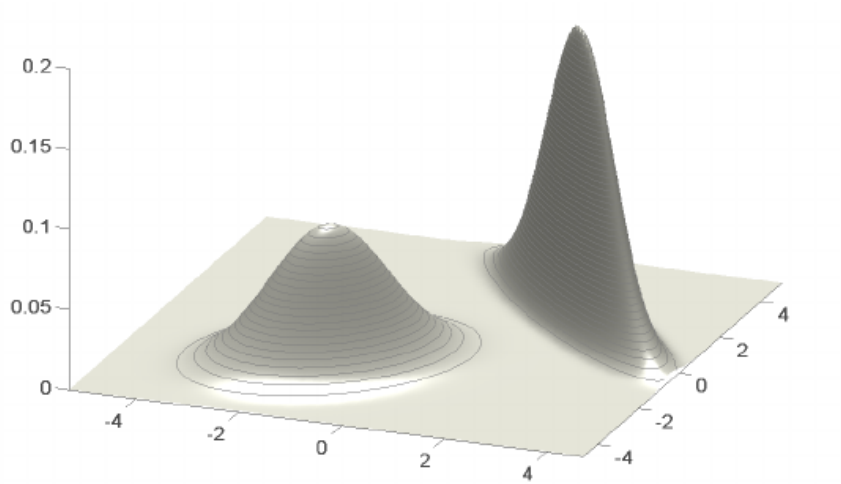



Figure B.2: Density II, $d = 2$.

When $d = 2$ Density II is a mixture of two bivariate Normal densities and is the same as Density A studied in Zhang et al. (2006):

$$\mu_a = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \quad \Sigma_a = \begin{pmatrix} 1 & -0.9 \\ -0.9 & 1 \end{pmatrix}, \quad \mu_b = \begin{pmatrix} -1.5 \\ -1.5 \end{pmatrix}, \quad \Sigma_b = \begin{pmatrix} 1 & 0.3 \\ 0.3 & 1 \end{pmatrix}.$$

# KDE (diagonal badwidth) Vs. SRP MCMC

Density II is a mixture of two multivariate Normal densities for $x \in \mathbb{R}^d$. Density II has high correlation between data coordinates and high bimodality:

$$f_{II}(x \,|\, \mu_a, \Sigma_a, \mu_b, \Sigma_b) = \frac{1}{2}\varphi(x \,|\, \mu_a, \Sigma_a) + \frac{1}{2}\varphi(x \,|\, \mu_b, \Sigma_b),$$

where $\varphi(x \,|\, \mu, \Sigma)$ is the multivariate Normal density with mean $\mu \in \mathbb{R}^d$ and $d \times d$ variance-covariance matrix $\Sigma$, and

$$\mu_a = \begin{pmatrix} 2.0 \\ \vdots \\ 2.0 \end{pmatrix}, \quad \Sigma_a = \begin{pmatrix} \sigma_a(x_1, x_1) & \sigma_a(x_1, x_2) & \cdots & \sigma_a(x_1, x_d) \\ \sigma_a(x_2, x_1) & \sigma_a(x_2, x_2) & \cdots & \sigma_a(x_2, x_d) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_a(x_d, x_1) & \sigma_a(x_d, x_2) & \cdots & \sigma_a(x_d, x_d) \end{pmatrix},$$

$$\mu_b = \begin{pmatrix} -1.5 \\ \vdots \\ -1.5 \end{pmatrix}, \quad \Sigma_b = \begin{pmatrix} \sigma_b(x_1, x_1) & \sigma_b(x_1, x_2) & \cdots & \sigma_b(x_1, x_d) \\ \sigma_b(x_2, x_1) & \sigma_b(x_2, x_2) & \cdots & \sigma_b(x_2, x_d) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_b(x_d, x_1) & \sigma_b(x_d, x_2) & \cdots & \sigma_b(x_d, x_d) \end{pmatrix},$$

and

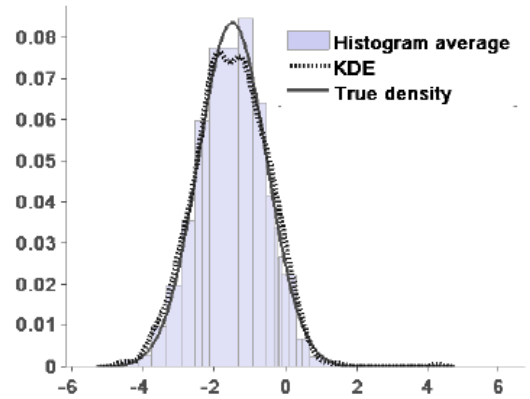
$$\sigma_a(x_i, x_j) = \begin{cases} 1 & \text{if } i = j, \\ -0.9^{|i-j|} & \text{if } i \neq j, \end{cases}, \quad \sigma_b(x_i, x_j) = \begin{cases} 1 & \text{if } i = j, \\ 0.3^{|i-j|} & \text{if } i \neq j, \end{cases}.$$

# KDE (diagonal badwidth) Vs. SRP MCMC

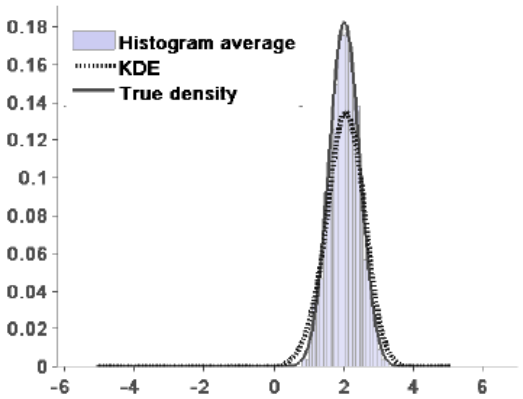
Table 7.2: Estimated errors for KDE and averaged SRP histogram RMRP.

| | $\hat{d}_{KL}$ | $\hat{L}_1$ error | Time (s) min. | Time (s) max. | Leaves min. | Leaves max. |
|---|---|---|---|---|---|---|
| 2-$d$ | | | | | | |
| KDE ($n_K = 2,000$) | 0.04 | 0.20 | 5,000 | 7,200 | $n/a$ | |
| Averaged RMRP histogram | | | | | | |
| $n = 10,000$ | 0.06 | 0.22 | 2 | 13 | 811 | 902 |
| $n = 50,000$ | 0.03 | 0.15 | 15 | 2,168 | 1,546 | 1,719 |
| 3-$d$ | | | | | | |
| KDE ($n_K = 2,000$) | 0.13 | 0.35 | 5,600 | 7,200 | $n/a$ | |
| Averaged RMRP histogram | | | | | | |
| $n = 10,000$ | 0.24 | 0.41 | 21 | 451 | 1,573 | 1,718 |
| $n = 50,000$ | 0.12 | 0.30 | 295 | 27,832 | 3,507 | 3,783 |
| 4-$d$ | | | | | | |
| KDE ($n_K = 2,000$) | 0.25 | 0.51 | 7,200 | 8,050 | $n/a$ | |
| Averaged RMRP histogram | | | | | | |
| $n = 50,000$ | 0.32 | 0.47 | 2,524 | 53,190 | 6,241 | 6,570 |
| $n = 100,000$ | 0.25 | 0.42 | 10,382 | 82,684 | 9,431 | 9,775 |
| 5-$d$ | | | | | | |
| KDE ($n_K = 2,000$) | 0.41 | 0.66 | 7,350 | 8,880 | $n/a$ | |
| Averaged RMRP histogram | | | | | | |
| $n = 50,000$ | 0.65 | 0.67 | 28,841 | 277,071 | 9,342 | 9,803 |
| $n = 100,000$ | 0.53 | 0.60 | 24,244 | 399,016 | 15,160 | 15,563 |

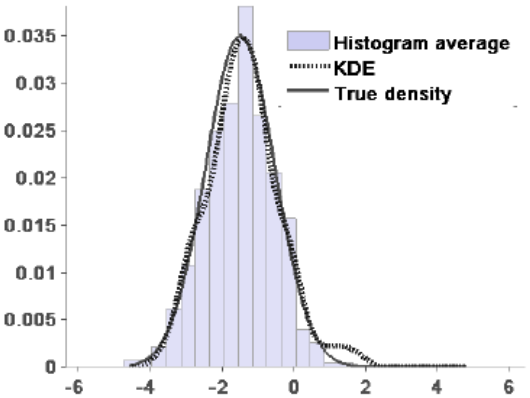
# KDE (diagonal badwidth) Vs. SRP MCMC



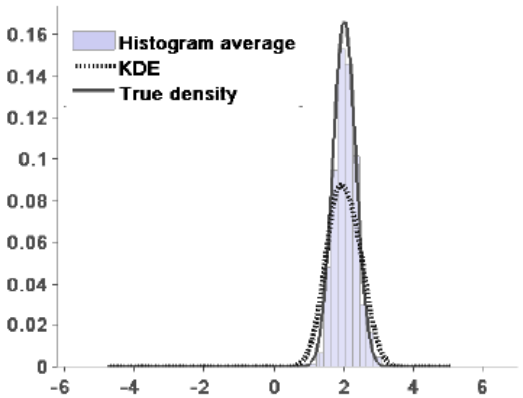

(a) $d = 2$, $x_2 = -1.5$, $n = 50,000$.




(b) $d = 2$, $x_1 = 2.0$, $n = 50,000$.




(c) $d = 3$, $x_2 = x_3 = -1.5$, $n = 50,000$.




(d) $d = 3$, $x_1 = x_3 = 2.0$, $n = 50,000$.

Figure 7.5: Density II, KDE and averaged RMRP histogram slice.