Data-Independent Space Partitionings for Summaries

Graham Cormode
University of Warwick
g.cormode@warwick.ac.uk

Minos Garofalakis
ATHENA Research Center
minos@athenarc.gr

Michael Shekelyan*
King's College London
michael.shekelyan@kcl.ac.uk

ABSTRACT

Histograms are a standard tool in data management for describing multidimensional data. It is often convenient or even necessary to define data independent histograms, to partition space in advance without observing the data itself. Specific motivations arise in managing data when it is not suitable to frequently change the boundaries between histogram cells. For example, when the data is subject to many insertions and deletions; when data is distributed across multiple systems; or when producing a privacy-preserving representation of the data. The baseline approach is to consider an equiwidth histogram, i.e., a regular grid over the space. However, this is not optimal for the objective of splitting the multidimensional space into (possibly overlapping) bins, such that each box can be rebuilt using a set of non-overlapping bins with minimal excess (or deficit) of volume. Thus, we investigate how to split the space into bins and identify novel solutions that offer a good balance of desirable properties. As many data processing tools require a dataset as an input, we propose efficient methods how to obtain synthetic point sets that match the histograms over the overlapping bins.

CCS CONCEPTS

• **Information systems** → **Multidimensional range search**; **Summarization**; • **Theory of computation** → **Data structures design and analysis**.

KEYWORDS

Multidimensional;Space Partitioning;Histograms;Equiwidth

ACM Reference Format:
Graham Cormode, Minos Garofalakis, and Michael Shekelyan. 2021. Data-Independent Space Partitionings for Summaries. In *Proceedings of the 40th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS '21), June 20–25, 2021, Virtual Event, China.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3452021.3458316

1 INTRODUCTION

Aggregate range queries are a crucial primitive for data analytics. These entail computing standard aggregates (such as SUM, COUNT, MIN or MAX) over values that meet a selection criterion corresponding to a geometric range. Typically, these are *box ranges*, specified by the intersection of one-dimensional range queries (e.g., "18 ≤ AGE ≤ 65") on each of d dimensions. Many applications require us to answer such queries quickly and accurately based on a summary of the data, without requiring a complete scan of the full input. Of particular relevance to this work are scenarios where the data may change dynamically (subject to insertions and deletions of records), and when the data is considered sensitive, and we wish to protect the privacy of the individuals corresponding to individual tuples. The canonical approach to this problem is to design and maintain histograms over the data, recording the data density (and other statistics) of items falling within the buckets, or bins, of the histogram. Even in one-dimension, there are multiple different histograms: equi-width (divide the domain into equal-length portions); equi-depth (choose bins so that an equal fraction of the input weight falls in each); and "optimal" histograms, which minimize the squared variation of weights within each bucket [20], to name but a few. In multiple dimensions, finding an optimal partitioning of the space based on the data becomes NP-hard [22], and many heuristics or approximations are proposed instead [15]. For a more comprehensive overview, we recommend surveys of this topic [7, 15].

Our focus in this work is the notion of a *data-independent histogram*, where the bins are chosen and fixed without first examining the data. There are a number of attractive features for this paradigm. We can give guarantees that are robust to arbitrary data and query distributions: the error, expressed in terms of the volume of the region of uncertainty can be precisely bounded in terms of the dimensionality of the space and the number of bins allocated to the histogram. Data-independent histograms are straightforward to update in the presence of dynamic data, precisely because their bin boundaries never alter. Last, they are highly suited to privacy-preserving publishing of data, as discussed below.

Much prior work which adopts data independent histograms appears to begin with the assumption that the best data-independent partitioning of a space is to simply take a single, regular grid. The starting point of our study is the observation that, while simple, this approach is not optimal. Instead, we will show that approaches based on keeping multiple histograms with different shaped bins yield improved accuracy for the same space budget. We begin by posing a problem, which we refer to as the continuous binning problem. This problem is how to pick a small set of preselected ranges such that any query range can be approximately composed by them. This is for instance useful to maintain statistics for each preselected range, which can then be combined to approximate statistics for all query ranges. In order to facilitate the combination of statistics, the composition is limited to be additive, i.e., the ranges used to approximate a query range are not allowed to overlap each other.

We make progress on these questions as follows. First, we introduce a set of definitions to evaluate the quality of a data-independent histogram. The quality of a set of preselected ranges can be judged by its size, how well they allow to approximate each query range

[*] This work was done while the author was at the University of Warwick.

PODS '21, June 20–25, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8381-3/21/06.
https://doi.org/10.1145/3452021.3458316

in terms of diverging (hyper)volume, how many preselected ranges are needed to compose query ranges and how much overlap exists between preselected ranges. Our main focus is on schemes which ensure that any query from a family of queries can be answered with bounded error in terms of the volume of space that the query occupies. We refer to these as “α-binnings”.

We study data independent histograms under choices such as the amount of overlap between bins in the histogram. In the strict case where there is no overlap between the preselected ranges, we call it a flat binning. For a fixed approximation error, we prove a tight lower bound for the size of flat binnings. As one might expect here, the optimal flat binning is a regular grid, although the number of bins must be very large in order to offer a fixed accuracy guarantee, exponential in the data dimension, d. Nevertheless, we show that using exotic tilings one could not improve more than by constant factors.

The bulk of our work is in studying the case when the preselected ranges can overlap. Here, we can obtain considerably improved results. We study existing and novel approaches to data independent histograms, and analyze their properties. The approaches we consider draw preselected ranges from *multiple* grids, where each grid offers a different trade-off between the precision along spatial dimensions. A natural approach is to collect many grids of all shapes and sizes — based, say, on having cell dimensions of powers of two. This “complete dyadic” approach improves on a uniform grid, but not dramatically so. We obtain stronger results by identifying a more restricted set of grids, inspired by the discrepancy theory literature. Here, we collect grids with resolutions that are powers of two, but restricted to those where each bin has the same fixed area. For instance, in this “elementary dyadic” scheme, one could have the grids with dimensions 1×16, 2×8, 4×4, 8×2 and 16×1. The advantage of the elementary dyadic scheme is that it requires fewer preselected ranges to achieve small approximation error. One downside is that each data point is contained in many preselected ranges, which can lead to larger update times and more noise in case of privacy-preservation. To address these shortcomings, we propose different ways to use a small number of grids and analyse the properties in that case.

Our last algorithmic result is a novel binning scheme “varywidth”, with a simple structure: we take a uniform grid, and create d copies, each of which refines the gridding in one of the d dimensions. This has a worst case cost (number of bins) in which we approximately halve the exponent for the uniform grid, with a dependence of $(d + 1)/2$ as opposed to d. We plot the analytical comparisons of the different schemes, which demonstrate that the novel methods of elementary dyadic and varywidth are preferable to the more familiar uniform and complete dyadic schemes.

We further evaluate these data independent histograms by considering applications and extensions. While a binning can be a useful summary of a large point set, it is sometimes important to also recreate a point set back from a binning. For a “flat” binning, where all bins are disjoint, this is a trivial task. We discuss the more complex case when bins overlap, and provide some candidate approaches. Last, we consider applications for data-independent schemes in the context of dynamic and privacy-sensitive data and draw parallels to data-dependent indexing and summary schemes.

Table 1: Aggregators in the semigroup model (query answers can be constructed from unions of disjoint fragments), and aggregators in the group model (query answers can be constructed by adding/subtracting fragments).

	semigroup	group
Count / Sum	yes	yes [34]
Diff.-Priv.-Count/Sum	yes	yes
Average / Variance	yes	yes [34]
Min. / Max. / Top-k	yes	no
Approximate Min./Max.	yes	yes
Approximate Distinct	yes	yes
Random sample	yes	no
Approximate Quantiles	yes [1]	no
F_2 AMS / CM / ℓ_1 sketches	yes [3, 8, 12, 26]	no
Heavy hitters	yes [1]	no
HyperLogLog	yes [14]	no
Exact Quantiles and Min/Max	no	no

2 PRELIMINARIES

In this section, we introduce the necessary definitions to describe different histogram methodologies, and use these to describe existing approaches.

2.1 Formal Framework

Definition 2.1 (data space). The d-dimensional data space is a unit cube in the d-dimensional Euclidean space.

Definition 2.2 (region). A *region* is a connected set of points in the data space.

Definition 2.3 (bins and binning). A *binning* is a (possibly overlapping) set of regions *(“bins”)* whose union covers the whole space.

Hence, each point in the data space is contained in at least one bin. In this work, we think of binnings as a collection of regions (bins) that, through composition of the bins, can be used to answer (approximately) aggregate range queries. For any query region, we can seek a (maximal) bin-aligned contained region and a (minimal) bin-aligned containing region (completely covering the query region). As a simple example, we can define a binning via a regular grid with some fixed cell size. Then the grid’s cells can be thought of as the bins, and for any query region we can find both the minimal set of cells whose union contains the query region and the maximal set of cells that are fully contained in the query region.

A binning can be used to approximately answer standard aggregates such as SUM, COUNT, MAX and MIN. We just prepare a query result for each bin (the appropriate weight of data points within the bin), and apply the corresponding aggregate over the weights in the bin aligned region for a query Q: take the SUM of the weights for the overall sum, or MAX for the overall max, etc. More generally, we can apply any aggregator that has the semi-group property to combine partial results per bin: see Table 1 for a list and references.

Clearly, the better the bin-aligned regions of a binning match the query region, the more precise we can expect the approximate answers to be. We assume that although the data density may vary

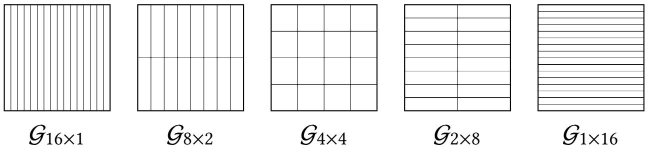

$\mathcal{G}_{16\times 1}$
$\mathcal{G}_{8\times 2}$
$\mathcal{G}_{4\times 4}$
$\mathcal{G}_{2\times 8}$
$\mathcal{G}_{1\times 16}$

Figure 1: The elementary binning $\mathcal{L}_4^d$ in $d = 2$ dimensions is the union of grids $\mathcal{G}_{16\times 1} \cup \mathcal{G}_{8\times 2} \cup \mathcal{G}_{4\times 4} \cup \mathcal{G}_{2\times 8} \cup \mathcal{G}_{1\times 16}$.

over the whole data space, locally it is more uniform, so that queries are answered better provided the uncertainty in volume from the binning is not too large.

Definition 2.4 (bin height). For a binning, we say that its *bin height* is h if any intersection of more than h bins is empty. We say that a binning is *flat*, iff it has bin height 1, in which case all bins are disjoint.

2.2 Example Data-Independent Binning Schemes

This section introduces data-independent binning techniques with precedent in the literature.

Grids. The simplest example of a data-independent binning scheme is given by a grid division of space.

Definition 2.5 (grid). A uniform grid in d dimensions with parameters $\ell_1, \ldots \ell_d$ is given by the following collection of bins:

$$\mathcal{G}_{\ell_1\times\ell_2\times\ldots\times\ell_d} = \bigcup_{j_1=1}^{\ell_1} \bigcup_{j_2=1}^{\ell_2} \cdots \bigcup_{j_d=1}^{\ell_d} \left(\left[\frac{j_1-1}{\ell_1}, \frac{j_1}{\ell_1} \right] \times \left[\frac{j_2-1}{\ell_2}, \frac{j_2}{\ell_2} \right] \times \ldots \times \left[\frac{j_d-1}{\ell_d}, \frac{j_d}{\ell_d} \right] \right).$$

A grid $\mathcal{G}_{\ell_1\times\ell_2\times\ldots\times\ell_d}$ is comprised of all cells of a regular grid with ℓ_i equi-width grid divisions in dimension i. Each of the $\prod_{i=1}^d \ell_i$ cells has the same volume, $1/\prod_{i=1}^d \ell_i$. A few example grids are shown in Figure 1.

Equiwidth and Marginal Binnings. While a grid itself provides a binning, we will use grids in the following primarily as building blocks for binnings that treat all dimensions the same way. A special case of a a grid that has the same number of divisions in all dimensions is an *equiwidth binning*:

Definition 2.6 (equiwidth binning). An equiwidth binning $\mathcal{W}_\ell^d$ with parameter ℓ is the grid $\mathcal{G}^d_{\ell\times\ell\times\ldots\times\ell}$.

We will also make use of a collection of grids where we only divide one dimension at a time, called *marginal binnings*:

Definition 2.7 (marginal binning).

$$\mathcal{M}_\ell^d = \mathcal{G}^d_{\ell\times 1\times\ldots\times 1} \cup \mathcal{G}^d_{1\times\ell\times\ldots\times 1} \cup \mathcal{G}^d_{1\times 1\times\ldots\times\ell}$$

As the purpose of a binning is to define a subset of queries that can be combined to make representatives for all queries, an equiwidth binning is limited to shifted (hyper-)cubes of the same size and a marginal binning to shifted slabs of the same size.

Dyadic Binnings. A more diverse set of shapes can be covered if we consider a greater range of edge lengths. While there are a very large number of such grids if we consider all possible edge lengths, we can obtain a useful selection if we restrict ourselves to grids based on powers of two. We refer to binnings with this restriction to powers of two as dyadic binnings:

Definition 2.8 (complete dyadic binning). The (complete) dyadic binning with parameter m is given by the union

$$\mathcal{D}_m^d = \bigcup_{\ell_1,\ell_2,\ldots,\ell_d \in \{2^1, 2^2, \ldots, 2^m\}} \mathcal{G}_{\ell_1\times\ell_2\times\ldots\times\ell_d}$$

The complete dyadic binning $\mathcal{D}_m^d$ is the union of m^d grids, resulting in $|\mathcal{D}_m^d| = (2^{m+1}-1)^d$ bins, where each bin is the cross product of d dyadic intervals of the form $[\frac{j}{2^n}, \frac{j+1}{2^n}]$ for non-negative integers $j \leq 2^m$ and $n \leq m$. In one dimension, it is therefore equivalent to the set of dyadic intervals. A subset of the grids of $\mathcal{D}_m^d$ is shown in Figure 1.

This concept has been widely used in related settings. For example, it has been combined with “sketch” data structures in order to answer multidimensional range queries, where the approach is referred to as “dyadic decompositions” [7]. Here, a sketch is built for each of the grids and each cell is treated like a value. In computational geometry, a data-dependent variant of this idea is used to approximate $O(n^2)$ box-shaped range queries by $O(n)$ canonical subsets (corresponding to bins of a dyadic binning), where n is the number of points. Consider a range tree over a set of points where each point is contained in a distinct cell of a regular grid with widths $2^m \times 2^m \times \ldots \times 2^m$. In this case the range tree implicitly operates on a dyadic binning, i.e., each node will contain a set of points that are contained in a set of cells whose union is a different bin from $\mathcal{D}_m^d$ and the total number of nodes will be $|\mathcal{D}_m^d|$.

Elementary Dyadic Binnings. Complete dyadic binnings have lots of bins that are unions of other bins. In order to reduce the level of redundancy, one can select only the bins with volume $\frac{1}{2^m}$, which will result in the “elementary binning”:

Definition 2.9 (elementary dyadic binning).

$$\mathcal{L}_m^d = \bigcup_{p_1+p_2+\ldots+p_d = m} \mathcal{G}_{2^{p_1}\times 2^{p_2}\times\ldots\times 2^{p_d}}$$

Elementary dyadic binnings $\mathcal{L}_m^d$ are formed by the union of all grids $\mathcal{G}_{2^{p_1}\times 2^{p_2}\times\ldots\times 2^{p_d}}$ with non-negative integers $p_1, p_2, \ldots, p_d$ that sum up to m. As there are $\binom{m+d-1}{d-1}$ distinct sequences of such integers and each grid has 2^m bins, an elementary binning $\mathcal{L}_m^d$ contains $|\mathcal{L}_m^d| = 2^m\binom{m+d-1}{d-1}$ bins. In one dimension, an elementary dyadic binning reduces to an equiwidth binning.

For instance, $\mathcal{L}_4^2$, where each grid has $2^m = 16$ bins, is depicted in Figure 1. It is comprised of bins along grids with resolutions 16 × 1, 8 × 2, 4 × 4, 2 × 8 and 1 × 16.

In discrepancy theory, the same concept as elementary binnings appears under the term elementary intervals to generate near-uniformly spread points, with applications in numerical integration (quasi-Monte Carlo). The objective (in our terminology) is to generate a set of d dimensional points within a data space such that the number of points in any box minimizes the difference to a continuous uniform distribution, which is referred to as the discrepancy. It was shown that choosing a set of boxes that corresponds to our notion of an elementary binning $\mathcal{L}_m^d$ has (in low dimensionality) a significantly lower discrepancy than random points. This connection is discussed further in Section 3.2. In the streaming literature,

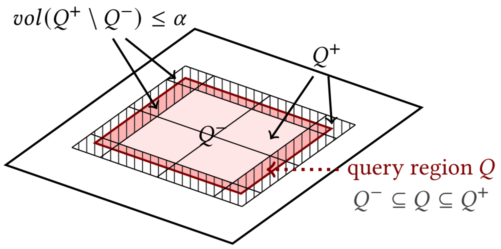

$vol(Q^+ \setminus Q^-) \leq \alpha$
Q^+
Q^-
query region Q
$Q^- \subseteq Q \subseteq Q^+$

Figure 2: For α-binnings, the volume of any alignment region $Q^+ \setminus Q^-$ (hatched region) is at most α.

a data-dependent variant of elementary binnings is used as a summary structure for data points that can be constructed in d stream passes [32], which was refined for two dimensions in [36].

3 α-BINNINGS

This section introduces a special class of binnings, namely, α-binnings, that for any query region provide a set of query-answering bins whose union differs in at most volume α from the query region. Table 2 summarizes the binnings that have appeared previously in the literature.

3.1 Definitions

In order to formally define the class of α-binnings, we introduce definitions based on volumes of regions and their differences. First, we denote the volume of a region A as $\text{vol}(A)$. Then,

Definition 3.1. A pair of regions A, B are α-similar if $\text{vol}(A \cup B) - \text{vol}(A \cap B) \leq \alpha$.

Intuitively, two regions are α-similar if they "differ" in at most an α fraction of the data space.

Definition 3.2 (α-binning). For any supported query region $Q \in \boldsymbol{Q}$, an α-binning allows us to find two regions Q^+ and Q^-, such that Q^+ and Q^- are α-similar. Q^- is the "contained region" for Q, so that $Q^- = (a_1 \cup a_2 \cup \ldots \cup a_n) \subseteq Q$ while Q^+ is the "containing region" $Q^+ = (Q^- \cup b_1 \cup b_2 \cup \ldots \cup b_m) \supseteq Q$ where $\{a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m\}$ are disjoint bins of the binning.

Thus, an α-binning bounds any supported query region $Q \in \boldsymbol{Q}$ by a pair of regions $Q^- \subseteq Q \subseteq Q^+$ that are α-similar to each other (and hence also to Q), each formed as a union of bins from the binning (cf. Figure 2). This allows query answering for aggregators that have semi-group semantics. For instance, if a maximum aggregate is stored for each bin, then Q^- implies a lower and Q^+ an upper bound for the maximum aggregate over Q. Table 1 enumerates many other supported aggregators such as sketches, distinct value estimators, or sum/average/variance aggregators. We abstract the process of finding the bins to answer a given query as an alignment mechanism.

Definition 3.3 (alignment mechanism and answering bins). An *alignment mechanism* $\mathcal{A}$ for a binning maps any supported query region $Q \in \boldsymbol{Q}$ to a set of *answering bins*, i.e., a set of disjoint bins $\mathcal{A}(Q) = \{a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m\}$ that satisfies $Q^- \subseteq Q \subseteq Q^+$ for $Q^- = (a_1 \cup a_2 \cup \ldots \cup a_n)$ and $Q^+ = (a_1 \cup a_2 \cup \ldots \cup a_n \cup b_1 \cup b_2 \cup \ldots \cup b_m)$.

Definition 3.4 (bin-aligned region and alignment region). For a query region Q, a binning's *bin-aligned region* $Q^- \subseteq Q$ is the union of all answering bins that are completely contained in Q. Q's *alignment region* $Q^+ \setminus Q^-$ is the union of all answering bins that cross the border of Q.

Fact 1. *A binning is an α-binning if there exists an alignment mechanism where the volume of the alignment region is at most α.*

For d-dimensional space, we focus in this work primarily on the set of queries that we refer to as box ranges:

Definition 3.5 ($\mathcal{R}^d$). The set of d-dimensional *box ranges* $\mathcal{R}^d$ is comprised of all axis-aligned (hyper-)boxes in the d-dimensional euclidean space.

Canonical worst-case query for our binnings. For many of our constructions, we can reason about the worst case error based on a single query that occupies almost the entire space. Specifically, for the class of α-binnings supporting d-dimensional box ranges that are formed from the union of uniform grids, the query box (parametrized by r) $Q^{\max} = [\frac{1}{2r}, 1 - \frac{1}{2r}]^d$ will provide a worst case. We can observe that $Q^{\max}$ has the largest alignment region for any individual grid because the number of answering bins is proportional to the query volume and as a result more bins can be crossed along the borders of the space. The proximity to the space border is chosen as $\frac{1}{2r}$ to ensure that grid cells at the border will definitely be crossed. If the number of bins crossed by the query box is maximised for all individual grids, it is also maximised for their union, because answering bins are disjoint.

3.2 Discrepancy theory and α-binnings

As α-binnings can be thought of as space-partitionings for infinitely many uniformly spread points, it is natural that there should be some connection to the notions of geometric discrepancy and low-discrepancy point sets [16, 28, 30, 35], which define a measure of how uniformly points are spread and aim to minimise it (with applications in numerical integration, see Section 6 for more details on discrepancy theory). In this context, a relevant notion is Niederreiter's $(t, m, s) - nets$, which are s-dimensional point sets that contain exactly 2^t points in each of 2^m boxes. For (t, m, s)-nets over base 2 (also known as digital nets) these boxes coincide with bins of an elementary binning $\mathcal{L}_m^d$ as discussed in Section 2.2. We can indeed generalise this notion in the following theorem to all α-binnings that have equal-volume bins :

Theorem 3.6. *Let t, m be non-negative reals and P be a set of d-dimensional points. If an α-binning supports queries $\boldsymbol{Q}$, each of its bins has the same volume and contains exactly 2^t points from P, then the discrepancy of P is at most* $\max_{Q \in \boldsymbol{Q}} \mid |P \cap Q| - |P| \text{vol}(Q) \mid \leq 2^t \alpha |P|$.

Proof. Let Q be some query in $\boldsymbol{Q}$. Let A be the alignment region with $\text{vol}(A) \leq \alpha$ and $(Q \setminus A) \subseteq Q \subseteq (Q \cup A)$. Let v be the bin volume. The binning requires $2^t \text{vol}(Q \setminus A)/v$ points to be contained in $(Q \setminus A)$, because it is aligned with the bins and there should exist $\text{vol}(Q \setminus A)/v$ bins in that region, with each bin containing 2^t points. For similar reasons, it also requires $2^t \text{vol}(A)/v$ points to be contained in A. Thus, the number of points in Q lies between $2^t \text{vol}(Q \setminus A)/v$ and $2^t \text{vol}(Q \setminus A)/v + 2^t \text{vol}(A)/v$. From $\text{vol}(Q) - \alpha \leq$

Table 2: Binnings supporting box queries that appear in the literature

binning	bins	height	number of answering bins	type
equiwidth $\mathcal{W}_\ell^d$	ℓ^d	1	ℓ^d	grid, equal-volume bins
marginals $\mathcal{M}_\ell^d$	$d\ell$	d	ℓ	union of grids, equal-volume bins
multiresolution [13] $\mathcal{U}_m^d$	2^{m+1}	m	$2^d(m-2)$	union of grids
complete dyadic [4, 5, 7, 31] $\mathcal{D}_m^d$	$(2^{m+1}-1)^d$	m^d	$2^d(m-2)^d$	union of grids
elementary dyadic [28, 29, 32] $\mathcal{L}_m^d$	$\binom{m+d-1}{d-1}2^m$	$\binom{m+d-1}{d-1} = O(m^{d-1})$	2^m	union of grids, equal-volume bins

$\mathrm{vol}(Q \setminus A) \leq \mathrm{vol}(Q)$ it then follows that the number of points in Q lies between $2^t \frac{\mathrm{vol}(Q)-\alpha}{v}$ and $2^t \frac{\mathrm{vol}(Q)+\alpha}{v}$. As the binning requires the total number of points to be $2^t n = \frac{2^t}{v}$, it thus follows that $|\mathrm{vol}(Q)n - x| \leq 2^t \alpha n$ where x is between $(\mathrm{vol}(Q) - \alpha)2^t n$ and $(\mathrm{vol}(Q) + \alpha)2^t n$. □

3.3 Lower bounds for supporting $\mathcal{R}^d$

This section proves lower bounds on the number of bins necessary to provide an α-binning supporting box-shaped ranges using any type of regions as bins.

Given a subset of overlapping bins, we define their intersection volume as the volume of their mutual intersection. Our subsequent results show that if a binning supports box ranges (boxes, for short), then a subset of all boxes (equivalent to the bins of an elementary dyadic binning) force the binning to have at least a certain amount of bins, due to the limited intersection volume of such boxes. In order to not require one separate bin for each of these boxes, a bin region has to be contained in the intersection of multiple boxes, because a bin can only contribute to the contained bin-aligned region of a box if it is fully contained in the box (Definition 3.2).

First, we show that for this special subset of boxes, the intersection volume of multiple boxes can be tightly bounded. This will be used to form a collection of queries as a hard instance that any α-binning must be able to handle.

Lemma 3.7. *The intersection volume of $\binom{k+d-1}{d-1}$ or more bins, drawn from an elementary dyadic binning $\mathcal{L}_m^d$, cannot be larger than $\frac{1}{2^{m+k}}$. Conversely, in order to achieve intersection volume of $\frac{1}{2^{m+k}}$, we cannot intersect more than $\binom{k+d-1}{d-1}$ bins from $\mathcal{L}_m^d$.*

Proof. Recall that the elementary dyadic binning is formed in a highly structured way, so that every bin from any grid in $\mathcal{L}_m^d$ has the same volume, 2^{-m}. There are $h = \binom{m+d-1}{d-1}$ grids. We will use d-dimensional coordinate notation of the form $R = [r_1, r_2, \ldots r_d]$ to refer to the grid with resolutions $2^{r_1} \times 2^{r_2} \times \ldots 2^{r_d}$ for each of the resolutions in turn. We write $|R|$ to denote $\sum_{i=1}^d r_i$—note that the volume of every cell in the grid R is $2^{-|R|}$. We define the intersection of grids to be the collection of new non-empty regions formed by intersecting all pairs of bins from the two grids. Observe that if we intersect two (dyadic) grids with resolutions R and S, we obtain a new grid whose resolution is $(R \cap S) := [\max(r_1, s_1), \max(r_2, s_2), \ldots \max(r_d, s_d)]$. Moreover, this intersection operation is associative, $(R \cap S \cap T) = (R \cap S) \cap T = R \cap (S \cap T)$. Hence, if we take the intersection of all the grids in the elementary binning $\mathcal{L}_m^d$, we obtain the resolution $[m, m, \ldots m]$, and so the volume of the intersection of h cells is at most 2^{-md}.

In order for a set of grids $\mathcal{X} \subset \mathcal{L}_m^d$ to have an intersection grid with resolution T s.t. each grid cell has volume $\frac{1}{2^{k+m}}$ then we must have $|T| = m + k$. Then T can be formed by the intersection of all grids R such that $|R| = m$ with $R[i] \leq T[i]$ for all $1 \leq i \leq d$. Then we have $|T - R| = |T| - |R| = k$. That is, the difference between the resolution vectors is also a d-vector whose (non-negative integer) entries sum to k. We can bound the number of such R as $\binom{k+d-1}{d-1}$.

From this, we have our claimed results: if we intersect more than $\binom{k+d-1}{d-1}$, then we will end up with a T such that $|T| > k + m$. Moreover, this is the largest volume that can be obtained through intersection of no more than this many bins. □

Theorem 3.8 (Bound for α-binnings supporting box-queries). *An α-binning supporting $\mathcal{R}^d$ has at least $\Omega(\frac{1}{2^d}\frac{1}{\alpha}\log^{d-1}\frac{1}{\alpha})$ bins.*

Proof. In order to show a lower bound, we will consider a family of queries and argue that any binning that can support all these queries with the required accuracy has at least the claimed number of bins. For this query family, we will make use of the set of query boxes corresponding to one of the binning approaches we have already desribed. We use the bins of an elementary binning $\mathcal{L}_m^d$ with $m = \lfloor \log_2(\frac{1}{2\alpha}) \rfloor$, such that each bin has at least volume 2α. Recall from Section 2.2 that the total number of bins in this elementary binning $N = |\mathcal{L}_m^d| = 2^m\binom{m+d-1}{d-1} = \Omega(\frac{1}{\alpha}\log^{d-1}\frac{1}{\alpha})$. We refer to each of these bins as an elementary box. We have that the height of the elementary binning is $h = \binom{m+d-1}{d-1}$.

We now consider an arbitrary α-binning which we will use to answer a query that is an elementary box. For that box, some bins of the binning may be used to form the contained bin-aligned region. Note that the contained bin-aligned region for each elementary box must be non-empty, since its volume is 2α, and so the contained bin-aligned region must have volume at least $(2\alpha - \alpha) = \alpha$.

We say that a bin of our binning *contributes* to an elementary box if it is fully contained in that box (and hence can be part of the bin-aligned region Q^-). If a bin contributes to an elementary box, the elementary box *receives* the volume of the bin.

Summing this over all N boxes, we conclude that summed over all bins the elementary boxes must receive a total volume of $N\alpha$ from the bins.

We now analyze the tradeoff between the number of boxes that a bin contributes to, and the total volume received by the boxes.

By Lemma 3.7, we have that for any $0 \leq k \leq m(d-1)$ and for $d \geq 2$, the intersection volume of $\binom{k+d-1}{d-1}$ elementary boxes is $\leq \frac{1}{2^{m+k}}$. The larger the intersection volume and the fewer the number of intersected boxes, the larger is the received volume per bin.

In order for a bin to contribute to x elementary boxes, it needs to be contained in the intersection of all x boxes. If a bin contributes to $\binom{k+d-1}{d-1}$ elementary boxes, each box can receive from the bin at most $1/(2^{k+m})$ volume. Hence, the received volume from each bin is at most $\binom{k+d-1}{d-1}/2^{k+m}$.

This means the number of bins needed is at least

$$\frac{2^{k+m}}{\binom{k+d-1}{d-1}} N\alpha \geq \frac{2^{k+m}}{\binom{k+d-1}{d-1}} N \frac{1}{2^{m+2}} = \frac{2^k}{\binom{k+d-1}{d-1}} \frac{N}{4}$$

Now, it is left to minimise the term $\frac{2^k}{\binom{k+d-1}{d-1}}$ over all valid choices of k. Consider the ratio of this term as we increase from k to $k+1$: it is $2(k+1)/(k+d)$. For k small, this ratio is below 1, leading to a smaller value. The cross-over value is when $2(k+1)/(k+d) = 1$, which is achieved exactly when $k = d-2$. Hence, we minimize this term by choosing $k = d - 1$[1]. The term is then bounded by

$$\frac{2^{d-1}}{\binom{2d-2}{d-1}} \geq \frac{2^{d-1}}{4^{d-1}} = \frac{1}{2^{d-1}}.$$

It therefore follows that the minimal number of bins needed is at least $\frac{1}{2^{d-1}} \frac{N}{4} = \frac{1}{2^d} \frac{N}{2} = \Omega(\frac{1}{2^d} \frac{1}{\alpha} \log^{d-1} \frac{1}{\alpha})$. □

A similar argument also provides a lower bound for flat binnings, i.e., binnings which are restricted to a disjoint set of bins:

Theorem 3.9 (Bound for flat α-binnings supporting box queries). *A flat α-binning supporting $\mathcal{R}^d$ has at least $\Omega(\frac{1}{\alpha^d})$ bins.*

Proof. We will again find a set of query boxes that any (flat) α-binning must be able to answer. These query boxes will be derived from a particular binning scheme. Consider the marginal binning comprised of the d grids $\mathcal{G}_{\ell\times 1\times\ldots\times 1}, \mathcal{G}_{1\times\ell\times\ldots\times 1}, \ldots, \mathcal{G}_{1\times 1\times\ldots\times\ell}$ with $\ell = \lfloor\frac{1}{2\alpha}\rfloor$, s.t. each bin has volume $\frac{1}{\ell} \geq 2\alpha$. We refer to the bins of this binning as marginal boxes, and use them as our query set.

In order for a flat binning to be an α-binning for all these marginal boxes (which are a subset of all boxes), the total contribution to the containing bin-aligned regions of marginal boxes (Q^+) needs to be at least $d\ell\frac{1}{\ell} = d$. However, if each bin contributes to just one marginal box, the total contribution to Q^+ cannot be larger than the volume of the unit cube, i.e., 1. In order to possibly increase the total contribution from 1 to d, each bin needs to intersect at least d marginal boxes. Note, that at most d marginal boxes intersect each other. Going forward, we can therefore assume that every bin of the flat binning intersects exactly d marginal boxes.

Furthermore, an α-binning needs to contribute in total at least $d\ell\alpha = d\lfloor\frac{1}{2\alpha}\rfloor\alpha$ to the contained bin-aligned regions of marginal boxes (Q^-). In order for a bin to contribute to a contained bin-aligned region of a box, it needs to be contained in the box, which means in order to intersect d marginal boxes, it needs to lie in the intersection of d boxes. This follows that a bin has a volume that is at most as large as the intersection of d marginal boxes, which is $\frac{1}{\ell^d}$. As a result, the maximal contribution per bin is $\frac{d}{\ell^d}$. In conclusion, the binning needs at least $d\lfloor\frac{1}{2\alpha}\rfloor\alpha\frac{\ell^d}{d} \geq \frac{\ell^d}{2} = \Omega(\frac{1}{\alpha^d})$ bins. □

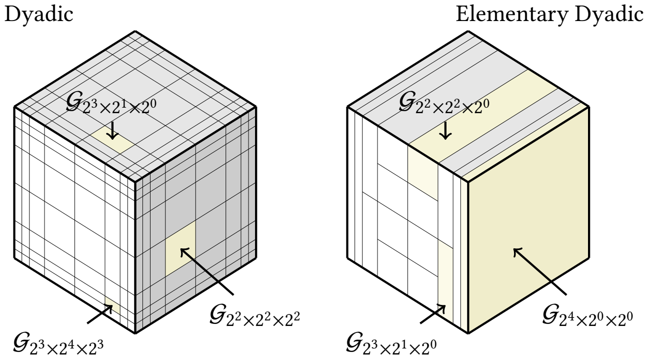

Dyadic
Elementary Dyadic
$\mathcal{G}_{2^3\times 2^1\times 2^0}$
$\mathcal{G}_{2^2\times 2^2\times 2^2}$
$\mathcal{G}_{2^3\times 2^4\times 2^3}$
$\mathcal{G}_{2^2\times 2^2\times 2^0}$
$\mathcal{G}_{2^4\times 2^0\times 2^0}$
$\mathcal{G}_{2^3\times 2^1\times 2^0}$

Figure 3: Fragmentation of a cube-shaped query box into dyadic boxes (on the left) and equal-volume elementary dyadic boxes (on the right) , with some indicated origin grids of dyadic boxes.

As an aside, we conjecture that data-dependent binnings for multisets of points of size n have matching size lower bounds for counts with additive error εn, because the α-binning setting is essentially the same as summarising infinitely many uniformly spread points. Thus, asymptotic bounds where ε basically goes to 0 implicitly forces n to go to infinity and the points being uniformly spread is one of many possible data distributions. For instance, for a uniformly distributed data set, a multidimensional equi-depth histogram reduces to an histogram along an equi-width binning.

3.4 Upper bounds for supporting $\mathcal{R}^d$

Family of Subdyadic Binnings. A binning is subdyadic if it is a union of grids whose resolutions are powers of two. We call them subdyadic binnings, because each such binning with maximal grid resolution m is a subset of the (complete) dyadic binning $\mathcal{D}_m^d$. A couple of examples are highlighted in Figure 4, which uses tables to show which subsets of the complete dydadic binning are materialized. We can hence express a subdyadic binning as a selection of cells in a multidimensional table, where each cell corresponds to one dyadic grid. For instance, the complete dyadic binning picks all dyadic grids (up to a certain resolution), whereas the elementary binning picks the grids corresponding to the leading diagonal in the figure (where all resolutions sum up to the same number, i.e., the volume of the bins is equal).

A general method to query subdyadic binnings is to first split the query into dyadic intervals along each dimension and then assign each cross product of dyadic intervals (which we will call dyadic boxes) over to the grid that has a cell with a matching region. Figure 3 shows the three-dimensional dyadic boxes of the worst-case query for $m = 4$. This suffices to answer queries for a (complete) dyadic binning. However, in a subdyadic binning, the assigned grid may not be present, and so we need to decide how to reassign the dyadic box to a grid which is present.

If the dyadic boxes from one grid are reassigned to a finer grid, the dyadic boxes are split into the cells of the finer grid, which increases the number of query-answering bins. To keep this number as low as possible, dyadic boxes of missing coarser grids should be passed

[1] Note that it holds that $2^{d-2}/\binom{2d-3}{d-1} = 2^{d-1}/\binom{2d-2}{d-1}$.

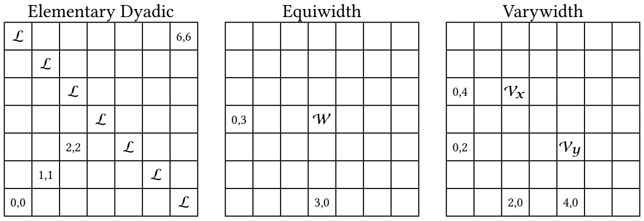


Figure 4: Subdyadic binnings (elementary dyadic $\mathcal{L}$, equiwidth $\mathcal{W}$ and varywidth $\mathcal{V}$) select different sets of grids from a d-dimensional table (depicted $d = 2$) where each coordinate $a, b, \ldots, z$ contains the grid $\mathcal{G}_{2^a \times 2^b \times \ldots \times 2^z}$.

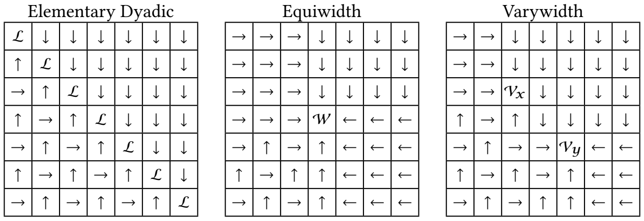


Figure 5: Querying hand-off rules for subdyadic binnings depicted by paths from missing grids to selected grids (note, that only the source and the target of the paths matter and some path segments are arbitrary).

on to the closest selected grid in terms of L_1 distance along the grid. While there are often multiple grids at the same distance, at least w.r.t. to the worst-case query it does not make a difference which one is chosen.

While this covers how to redirect from coarser to finer grids, it does not answer how to redirect from finer to coarser grids. We leave the former as an open problem and only answer it for certain subdyadic binnings. Figure 5 gives a pictorial encoding of how we might progressively reassign dyadic boxes for different subdyadic binnings so that we can eventually answer the query.

Equiwidth. An equiwidth binning is a subdyadic binning that assigns dyadic boxes to the grid $\mathcal{G}[\frac{m}{d}, \frac{m}{d}, \ldots, \frac{m}{d}]$. It is an asymptotically optimal α-binning (treating d as a constant), if the bins are not allowed to overlap each other:

Lemma 3.10 (upper bound for flat binnings). *There exists a flat α-binning supporting $\mathcal{R}^d$ with $\Theta((\frac{2d}{\alpha})^d)$ bins.*

Proof. Each equiwidth binning with $\ell \geq 2$ grid divisions per dimension and ℓ^d bins is an α-binning for some α. Note, that such a binning is only subdyadic in case ℓ is a power of two.

As an equiwidth binning is a (uniform) grid, it follows from the discussion in Section 3.1 that the largest alignment region is for a query that almost touches the border of the space. This means the maximal alignment region volume α is the number of border cells divided by total number of grid cells. The number of border cells is the number of all cells ℓ^d, less the number of non-border cells $(\ell - 2)^d$. By ignoring some double-counting of cells, one can obtain an upper bound by multiplying the number of cells ℓ^{d-1} on each side by the number of sides $2d$. From that, it follows $\alpha = \frac{\ell^d - (\ell-2)^d}{\ell^d} < \frac{(2d)\ell^{d-1}}{\ell^d}$. Solving for ℓ to obtain the target α results in $\ell < \frac{2d}{\alpha}$, and so the number of bins is less than $\frac{(2d)^d}{\alpha^d}$. □

Elementary dyadic. For an elementary dyadic binning $\mathcal{L}^d_m$ we can define the assignment that for each missing grid $\mathcal{G}[a, b, \ldots, z]$ with $a + b + \ldots + z > m$, all its dyadic boxes are handled by $\mathcal{G}[a, F_m(a, b), F_m(a, b, c), \ldots, F_m(a, b, c, \ldots, z)]$ where $F_m(a, b) = \min\{b, m - a\}$ and $F_m(a, b, \ldots, y, z) = \min\{z, m - F(a, b \ldots, y)$. Such a rule simply dictates that we greedily increase the resolutions, giving preference to the dimensions in order of appearance. Elementary dyadic binnings are asymptotically the best-known approach, if the bin height is unlimited:

Lemma 3.11 (upper bound for arbitrary binnings). *There exists an α-binning supporting $\mathcal{R}^d$ with $\tilde{O}(\frac{1}{\alpha}\log^{2d-2}\frac{2^d}{\alpha})$ bins and height $\tilde{O}(\log^{d-1}\frac{2^d}{\alpha})$ where $\tilde{O}(\ldots)$ omits* poly$(\log\log\frac{1}{\alpha})$ *factors.*

Proof. As an elementary dyadic binning is the union of (uniform) grids, it follows from the discussion in Section 3.1 that such a constrained binning will always have the largest alignment region volume α for a "worst-case query" that almost touches the border of the space. Thus, one can derive a sufficient number of bins to achieve a certain α.

Each query is split into $4 + 2(m-2)$ fragments, where 4 fragments are bins along one dimension and cannot be further split, while $2(m - 2)$ fragments can be further split into more parts in the next dimensions (illustrated in Figure 3). In the last dimension, each fragment is partially intersected by the worst-case query in at most two bins. When $d = 1$, the number of bins that are partially intersected by the worst-case query is $f_d(m) = 2 = \Theta(1)$. For $d > 1$ it is equal to $f_d(m) = 4 + 2\sum_{n=1}^{m-2} f_{d-1}(n) = \Theta(m^{d-1})$ (unless $m \leq 2$, in which case $f_d(m) = 2^m$. As each bin has volume $\frac{1}{2^m}$, the maximal alignment region volume $\alpha = \frac{f_d(m)}{2^m} = O(\frac{m^{d-1}}{2^m})$.

Provided that d divides m, a dyadic binning will always contain an equiwidth grid with $\Theta(2^{m/d})$ grid divisions per dimension, and so it follows that $\frac{2d}{2^{m/d}} \geq \alpha$ and $m \leq \log_2(\frac{2d}{\alpha}^d) = d\log_2\frac{2d}{\alpha}$. We use this upper bound for m to substitute for the numerator in the expression for α, but keep m in the denominator and solve for m.

This yields $\alpha = O(\frac{m^{d-1}}{2^m}) \in O(\frac{d^{d-1}\log^{d-1}\frac{2d}{\alpha}}{2^m})$. Solving for m results in $m = O(\log\frac{d^{d-1}\log^{d-1}\frac{2d}{\alpha}}{\alpha})$. Thus, after some simplifications we obtain that the number of bins is $O\left(\log^{d-1}\left(\frac{\log^{d-1}\frac{2^d}{\alpha}}{\alpha}\right)\frac{\log^{d-1}\frac{2^d}{\alpha}}{\alpha}\right)$ □

3.5 Varywidth Binning Scheme

We now introduce a simple novel binning strategy with bounded height, by seeking to remedy the deficiencies of the simple gridding approach. It can be observed that an equiwidth binning with ℓ cells per dimension accumulates all of its bin-alignment error along the border of the query box. This query can be assumed to be of maximal size, as any other box is simply the query box of maximal size for a smaller grid. The border of the query box is comprised of $3^d - 1$ faces, of which $2^{d-k}\binom{d}{k}$ are k-dimensional faces (e.g., corners are 0-dimensional faces, edges are 1-dimensional faces and

Table 3: Comparison of different α-binnings. $\tilde{O}(\ldots)$ hides any $\mathrm{poly}(\log\log\frac{1}{\alpha})$ terms.

binning scheme supporting $\mathcal{R}^d$	number of bins	height h	number of query-answering bins
lower bound for flat binnings	$\Omega(\frac{1}{\alpha^d})$	1	$\Omega(\frac{1}{\alpha^d})$
equiwidth	$O(\frac{(2d)^d}{\alpha^d})$	1	$O(\frac{(2d)^d}{\alpha^d})$
lower bound for arbitrary binnings	$\Omega_d(\frac{1}{\alpha}\log^{d-1}\frac{1}{\alpha})$	≥ 1	-
varywidth	$O_d(\frac{1}{\alpha^{(d+1)/2}})$	d	$O_d(\frac{1}{\alpha^{(d+1)/2}})$
elementary dyadic	$\tilde{O}_d(\frac{1}{\alpha}\log^{2d-2}\frac{1}{\alpha})$	$\tilde{O}_d(\log^{d-1}\frac{1}{\alpha})$	$\tilde{O}_d(\frac{1}{\alpha}\log^{d-1}\frac{1}{\alpha}))$
dyadic	$O_d(\frac{1}{\alpha^d})$	$\tilde{O}_d(\log^d\frac{1}{\alpha})$	$\tilde{O}_d(\log^d\frac{1}{\alpha})$

sides are $d-1$ dimensional faces). There lie $(\ell-2)^k$ grid cells on each k-dimensional face. Thus, if $\ell \gg d$, most grid cells lie on the sides of the box (i.e., $d-1$-dimensional faces) and the question of whether a point is contained in the box is solely dependent on one dimension (orthogonal to the side of the box). We can make use of this by introducing bins that are "fat" in this one dimension, and "skinny" in the remaining dimensions. That is, we further split bins individually along each dimension into C parts, which results in an $C\ell\times\ell\times\ldots\times\ell$ grid, a $\ell\times C\ell\times\ldots\times\ell,\ldots,$ and a $\ell\times\ell\times\ldots\times C\ell$ grid. Thus, there are $dC\ell^d$ bins in total with d bin overlaps, although it has almost the same bin-alignment error as an equiwidth histogram with $(C\ell)^d$ bins.

Lemma 3.12 (upper bound for height d). *There exists an α-binning for d-dimensional box ranges with $O\left(d^{d+2}(\frac{2}{\alpha})^{(d+1)/2}\right)$ bins and bin height d.*

Proof. A varywidth binning has two parameters ℓ and C. It has $dC\ell^d$ bins along grids $G_1,\ldots,G_d$ where each grid G_i has ℓC grid divisions in dimension i, and ℓ divisions in the other dimensions.

A varywidth binning can be thought of as subdividing an equiwidth grid with ℓ^d "big" cells. Each such "big" cell is subdivided in each dimension with a sub-grid having C grid divisions along that dimension (and no grid divisions in the other dimensions). As with the equiwidth binning, the worst case for bin-alignment error is a query that covers almost the whole space, but does not touch the border of the space. The bin-alignment error is therefore accumulated along the borders of the data space, which is a cube. A cube has 3^d-1 faces, of which $2^{d-k}\binom{d}{k}$ are k-dimensional faces. Along each k-dimensional face, there are $(\ell-2)^k$ "big" cells, each having a volume of $\frac{1}{\ell^d}$. All subcells of a "big" cell on the border can be partially intersected, except if the "big" cell lies on the border only along one dimension, i.e., on one of the $2d$ sides of the hypercube which are $(d-1)$-dimensional faces. In this exceptional case only one subcell is partially intersected, which has a volume of $\frac{1}{\ell^d C}$. The reason only one subcell is partially intersected is because in each "big cell" that lies on the sides, the worst case query extends beyond the cell in all but one dimension and for that dimension there are C divisions available through the "big cell"'s subgrid of that dimension. Thus, the maximum volume of the alignment region is

$$\begin{aligned}\frac{\sum_{k=0}^{d-2}2^{d-k}\binom{d}{k}(\ell-2)^k}{\ell^d}&+\frac{2d(\ell-2)^{d-1}}{\ell^d C}\\ &<\frac{2d(d-1)\ell^{d-2}}{\ell^d}+\frac{2d(\ell-2)^{d-1}}{\ell^d C}\\ &=O\left(\frac{2d(d-1)}{\ell^2}+\frac{d}{\ell C}\right).\end{aligned}$$

For $C=\frac{\ell}{2(d-1)}$, the maximal volume of the alignment region is $\alpha = O(\frac{2d(d-1)}{\ell^2}) \leq O(\frac{2d^2}{\ell^2})$. Expressing ℓ through α yields $\ell = O\left(d\sqrt{\frac{2}{\alpha}}\right)$ and the number of bins is $d\ell^{d+1} = O(d^{d+2}(\frac{2}{\alpha})^{(d+1)/2})$. □

4 SAMPLING

The previous sections have discussed different data independent histogram constructions, and their ability to give upper and lower bounds for range queries via an alignment mechanism. This enables them to be used flexibly to answer a variety of different query types. However, there are many data analysis problems which do not immediately reduce to a collection of query regions. Consider, for example, clustering algorithms, which are defined to take as input a point set. For this reason, it is often useful to be able to extract a representative point set from a histogram representation that stores counts in each bin. We do not expect the output points to match the input exactly (since the point of a histogram is to act as a form of lossy compression), but we would like to ensure that they are consistent with the description of the spatial distribution given by the histogram. In this section, we describe two approaches that can be applied to the histograms we consider. First, we consider a random sampling approach to draw from the density distribution implied by the histogram. Second, we adopted the sampling approach to find point sets matchings the histogram counts exactly.

4.1 Sampling from distributions over binnings

Any histogram over a flat binning can be intepreted as a probability distribution over the data space where bin counts are normalized to sum up to one. Our challenge in this section is be able to sample in accordance with multiple distributions, originating from histograms for each flat binning. While the histograms cannot contradict each other, they offer different pieces of information that have to be pieced together to a coherent picture. In this context, it is helpful to introduce the concept of atoms of a binning, which are intersections of bins that are contained in all bin regions that intersect it, i.e., each bin is a union of atoms and each bin either fully contains an atom or does not intersect it. The coherent picture is a distribution over the atoms. If one could find probabilities for atoms, s.t., the sum of atom probabilities matches all bin probabilities, one could first draw a random atom and then uniformly draw a point from that atom. A challenge of such an approach is the sheer number of atoms, which can be orders of magnitude larger than the number of bins and in addition to that it is a challenging combinatorial problem. Thus, to completely avoid dealing with atoms directly, we instead exploit simple “intersection hierarchies” observed in the binnings such as equi-width, marginal binnings, varywidth and (two-dimensional) complete dyadic and elementary dyadic binnings.

The approach is not applicable to complete dyadic and elementary dyadic binnings in more than two dimensions, because their hierarchies become too complicated. This also mirrors the increased difficulty of obtaining low-discrepancy point sets in more than two dimensions, as that requires generating point sets that count one point in each bin of a dyadic binning. Thus, we leave this as an open problem.

In case of a marginal binning, one can draw a random bin from each flat binning and then intersect the result. We generalise this idea by the following “intersection sampling” algorithm:

(1) Split the binning into a “root” binning and multiple “branch” binnings according to the rules in Definition 4.2.
(2) Draw a random bin from the “root” binning according to its bin probabilities.
(3) Remove all “branch” bins that do not intersect the selected bin.
(4) Apply this sampling algorithm (in parallel) recursively to each “branch”, reduced to the bins that intersect the selected bin(s).
(5) Return the intersection of the random bin region with the returned region from each recursive call per “branch”.
(6) If this is the end of the recursion, uniformly draw a point from the returned region.

In step one, the binning is split into a *root* and multiple *branches* where the root is a flat binning and the branches are disjunct sets of non-root bins. In the remaining steps, a root bin is sampled, then a bin from each branch that intersects the root bin is sampled and at the end a point is sampled inside the intersection between the root and branch bins. In order to obtain the selected bin from the branches, the sampling approach is recursively applied.

In order for the approach to work, the choice of the root bin has to adhere to the probabilities of the branch bins and the choice of the branch bins have to be independent from each other. For the discussion of these properties it is helpful to introduce the concept of *super regions*:

Definition 4.1 (super region). A *super region* of a set of bins is a union of disjoint bins that contains all bin regions that intersect it.

Super regions of multiple flat binnings are the regions for which they require the same sum of probabilities, as there exists a union of bins in each of those flat binnings that equals a super region. Now, we can formalise the required property for the root-branch splitting of the binning as follows:

Definition 4.2 (intersection hierarchy rules). A split of a binning into a flat root binning R and multiple branch binnings $A, \ldots, Z$ is valid, iff it follows the following two rules:

(i) A bin from a branch binning has to intersect any root bin that has the same super region, where the super region is defined only over the bins from the root and that branch and not the other branches.
(ii) A bin from a branch has to intersect any bin from another branch that intersects the same root bin.

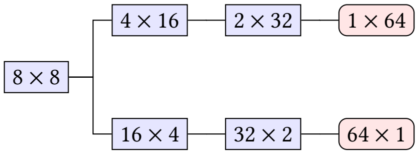


Figure 6: Recursive intersection hierarchy

Before we show how these properties ensure consistency, we will take a look at how such rules can be satisfied with grids. The grid with the highest minimal resolution in all dimensions can be picked as the root and each branch can contain grids that have a lower resolution in a distinct dimension.s For instance, suppose the binning is comprised of the equi-width grids $\{8 \times 8, 16 \times 4, 4 \times 16, 32 \times 2, 2 \times 32, 64 \times 1, 1 \times 64\}$, then the bins from the grid 8×8 can be used as a root and $\{16 \times 4, 32 \times 2, 64 \times 1\}$ as one branch and $\{4 \times 16, 2 \times 32, 1 \times 64\}$ as the other branch. This satisfies the intersection hierarchy rules, because each branch has a unique dimension in which it has a lower resolution, which after fixing a root bin becomes irrelevant, such that each branch specialises in the other dimension, i.e., it has in that dimension a higher resolution than the other branches. Recursively applying this approach results in the root grid choices of Figure 6. First a random bin is drawn from the 8×8 grid and then the approach recursively continues for each branch, e.g., the first branch $\{16 \times 4, 32 \times 2, 64 \times 1\}$, any bins that do not intersect the selected root bin are removed, which results in subgrids with local resolution $\{2 \times 1, 4 \times 1, 8 \times 1\}$, where the 2×1 grid can then subsequently be used as a root and $\{4 \times 1, 8 \times 1\}$ form a single branch.

In the following theorem, we now show how these properties guarantee that the intersection sampling algorithm operates according to all distributions over the flat binnings.

Theorem 4.3. *If there is a joint distribution over the space consistent with the bin probabilities and the intersection sampling algorithm is applicable, the algorithm samples according to such a consistent joint distribution.*

Proof. The sampling algorithm splits the binning in each step into root and branch bins as described in Definition 4.2. As root bins are sampled directly, they are guaranteed to be sampled according to their probabilities.

The second condition of Definition 4.2 ensures, that the selection of branch bins is conditionally independent from each other upon selection of a root bin, because any branch bin choices would intersect each other anyway and the algorithm can only exclude choices thats do not intersect previous choices.

The first condition from Definition 4.2 means that one could first select a super region summing probabilities over the unions of bins they are comprised of and the choice of branch bins is conditionally independent from the root bin choice upon selection of a super region, because root bins and branch bins sharing the same super region have to intersect each other. Furthermore, the choice of the super region follows the same probabilities as the branch bins, as the probability with which a super region has to be selected is by definition equal to the sum of probabilities of the branch bins it is composed of. The sampling algorithm is then recursively applied to the branches, to ensure that not only the summed probabilities for super regions are satisfied, but also of individual branch bins. □

4.2 Reconstructing point sets from histograms over binnings

The previous section allows us to sample a point that is consistent with every distribution implied by a stored binning. By repeatedly sampling, we can build up a set of points. However, each point is sampled independently, meaning that the point set is not guaranteed to agree with the stored bin counts. By contrast, in this section we seek to build a point set of the same size as the original, which will agree exactly with the stored counts for each bin. We modify the previous approach to adjust the sampling probabilities after each sample. Due to the intersection hierarchy rules followed by the sampling approach, the only modification necessary is to adjust bin probabilities as we go.

Theorem 4.4. *If there exists a probability distribution over the atoms of a binning that are consistent with the distributions over the flat binnings and the intersection hierarchy rules can be applied to a binning, one can construct a set of points that is consistent with a histogram over the binning using the intersection sampling algorithm.*

Proof. After each generated point the count of the containing bin in each flat binning can be increased, such that once the bin is "full" upon reaching the correct count it can be removed from further consideration. The only way to not produce a consistent point set would be if a non-full bin becomes unselectable due to other full bins. As root bins are directly sampled, they can be selected as long as they are not full. The selection of root bins and other branch bins cannot influence the selection of a branch bin, because of the conditional independence properties outlined in the proof of Theorem 4.3. Thus, if a branch bin can no longer be selected, this means either that the branch bin is full or that all root bins in the super region are full, in which case the branch bin is also full. □

This discussion assumes that the bin counts are all mutually consistent with some underlying assignment of points to atoms. This might not be the case if, for example, there is some noise in the bin counts. We can address this situation by harmonising the counts in a way that does not increase their variance. This is described in more detail in Section A.2, where noise is deliberately introduced to bin counts in order to ensure privacy.

5 APPLICATIONS

As binnings are data-independent they are a great tool for dynamic data and privacy preservation applications, where any data-dependent information can breach privacy. For the differential privacy setting, the binnings can either serve as a basis for histograms or as a means to obtain a sample.

5.1 Histograms over dynamic data

Highly dynamic data makes it very challenging to maintain data-dependent partitionings or even samples, as data removals require an additional sample over the removals. As an alternative, it is therefore common to utilise an equi-width binning.

The log-log plot in Figure 7 shows that varywidth and elementary dyadic binnings can achieve more precision (quantified through the maximal alignment error α) using fewer bins.

Equiwidth only does best for a low number of bins, whereas elementary dyadic does best for large number of bins and varywidth sits in between. A downside of elementary dyadic binnings, is that the binning height is very large in comparison, which can be undesirable in some cases.

As each update requires to modify one count in each flat binning, the update costs are proportional to the binning height. While an equi-width binning always has height 1 and a d-dimensional varywidth binning has height d, a d-dimensional elementary dyadic binning has a height dependent on the number of bins, i.e,. for $\binom{m+d+1}{d-1}2^m$ bins it has height $\binom{m+d+1}{d-1}$ where m is a positive integer. For a thousand bins, the elementary dyadic binning has at least height 8 in two dimensions (21 in three and 35 in four dimensions). For a million bins, the elementary dyadic binning has at least height 16 in two dimensions (105 in three and 364 in four dimensions). For a billion (10^9) bins, the elementary dyadic binning has at least height 26 in two dimensions (253 in three and 1540 in four dimensions). In conclusion, an elementary dyadic binning is more precise with more bins, but requires larger update costs. Varywidth appears to be a good compromise in that regard.

5.2 Differential Privacy

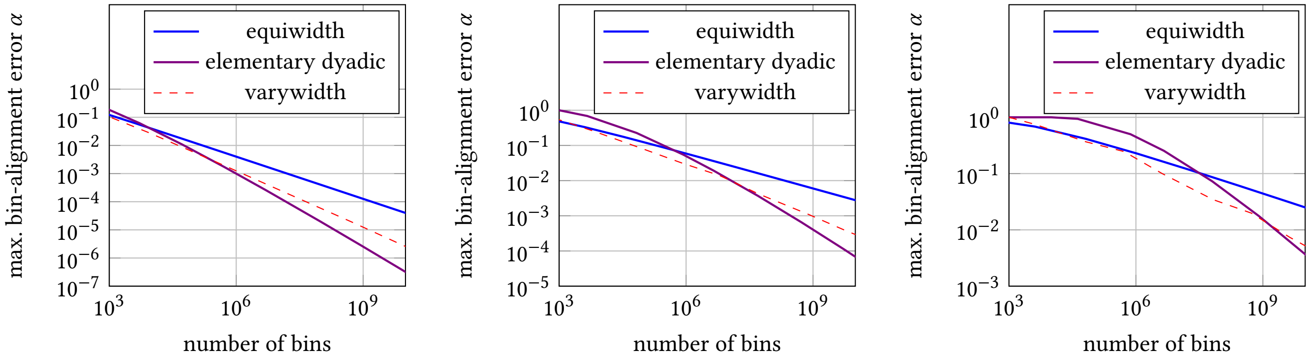
Differential privacy deals with sensitive data from individuals. Attackers that aim to breach the privacy of individuals might possess all but one record. In such a case, a released statistic over the data makes it easy to discern if an individual participated in it. To prevent such breaches, a random statistic is published skewed towards the correct statistic and the presence or absence of an individual changes the probabilities only slightly. For histograms, the the random statistics are counts with added random noise, usually from a Laplace distribution. For more details we refer the reader to [11].

Using existing techniques, the noisy bin counts can be harmonised to generate a sample according to the bin counts. One can then assess the spatial precision of the sample, i.e., how far points deviate from their correct positions quantified by the maximal alignment region volume, and the count precision of the sample, i.e., how many noise points are added and how many original points are removed quantified by the variance of aggregates.

The plot in Figure 8 (cf. Appendix) shows on the y-axis the achieved spatial precision as a function of counting precision on the x-axis. In this setting, binning techniques do best that require few bins, but also have a small bin height. This is achieved by consistent varywidth (varywidth with an additional grid that contains the super regions of the other grids) achieves both a better spatial as well as a better counting precision than other binnings. An extensive discussion can be found in the Appendix that details how to harmonise bin counts and which properties the sample inherits from the binning.

6 RELATED WORK

In the literature, most works focus on flat binnings, e.g., tilings. In this is shown that a regular grid (equiwidth) is asymptotically the best flat binning for box queries, which means that using alternative



(a) dimensionality $d = 2$

(b) dimensionality $d = 3$

(c) dimensionality $d = 4$

Figure 7: Number of bins of different schemes for box ranges

type of tilings such as hexagons can only lead to constant-factor improvements (that can depend on the number of dimensions).

Index structures share a lot of commonalities with binnings on a more abstract level. The goals of reducing querying and update times can be compared to that of reducing the number of answering bins of queries and the height of a binning. Indexing schemes [19, 24, 29] strip the indices down to the most bare-bone parts to better reason about lower bounds for querying and storage costs. Lower Bounds for Orthogonal Range Searching [2, 6] have some links to lower/upper bounds for α-binnings, as we can think of α-binnings as summaries over infinitely large uniformly spread datasets.

Subpavings [10, 21, 23] approximate arbitrary regions by a union of freely chosen boxes that contains the region and one that is contained in the region. They have many applications to set-inversion and other non-linear problems. One can think of subpavings as how to build query regions from a binning with infinitely many box-shaped bins, in way that reduces number of answering bins and the alignment error α.

Geometric discrepancy [27] is closely linked to α-binnings as it also deals with uniformity and L_∞-norms. We identify the dyadic boxes in (t,m,s)-nets [28] as the elementary binning and show that if an equal-volume α-binning sees the same number of points across the bins, that has implications for the uniform spread of the point sets. We thereby generalise the notion of (t,m,s)-nets to arbitrary binnings and utilise the α-binning property to derive discrepancy upper bounds of the point set. An open question is if lower bounds for the discrepancy of a point set have any implication on the number of bins of α-binnings, or vice versa. ε-approximations [17, 27] summarise a set of points through a subset that behaves almost identically for the ranges of interest. While α-binnings do not directly construct point sets, they do implicitly describe a point set via the bin regions and binning structure.

Dyadic boxes that form cross products over dyadic intervals can be found in almost any field that aims to reason over a continuous space, e.g, dyadic decompositions for sketches [7] and wavelets. Range trees over space partitions [33, 41] can also be thought of in this way, as each node of the tree will correspond to a dyadic box. A data-dependent analogue can be found in the summary literature [32, 37], that can be seen as a set of equi-depth histograms (each bucket containing the same number of points) where each one has the same number of space divisions, but the divisions are spread differently across dimensions, or as a range tree where we strip away all the lower-levels, so that each node contains the same number of points and then only keep the space partitioning.

In this work binnings are also used to obtain differentially private versions of a dataset (see extensive discussion in the Appendix). There are many works that take steps towards non-parametric differentially-private synthetic data generation [9, 25, 38–40, 42], but unlike this work they cannot guarantee a limited variance of aggregates along bin-aligned regions. To achieve these guarantees a sample is drawn from histograms over α-binnings, where the noise counts are harmonised [18] such that the leaf level mirrors the small variance of higher-level hierarchies.

7 CONCLUSION AND FUTURE WORK

This work revisits binnings, shifting the focus away from data to supported queries and how they can be approximated. Apart from establishing some lower bounds and identifying existing upper bounds in the literature, a novel type of binning termed varywidth is analysed and determined to offer excellent properties in the differential privacy setting. It is shown that a (consistent) varywidth binning offers the best trade-off between spatial precision and variance accumulated over differentially private aggregates. In future work, non-box queries (e.g., half-space queries) could be prioritised and the group model (allowing subtracting fragments) could be explored. Another aspect this work touches upon are the family of subdyadic binnings that share a universal querying algorithm, which starts from the dyadic decomposition and hands off the dyadic boxes to a select subset of the grids. Finding optimal subdyadic binnings, generating points from subdyadic binnings and how to optimally hand-off dyadic boxes are still open problems.

Acknowledgements. We thank the anonymous reviewers for their helpful remarks. This work was supported by European Research Council grant ERC-2014-CoG 647557, Leverhulme Trust grant RPG-2019-399, and EU RIA grant H2020-ICT-2018-2 825070 (INFORE).

REFERENCES

[1] Pankaj K. Agarwal, Graham Cormode, Zengfeng Huang, Jeff M. Phillips, Zhewei Wei, and Ke Yi. 2012. Mergeable summaries. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2012, Scottsdale, AZ, USA, May 20-24, 2012.* 23–34. https://doi.org/10.1145/2213556.2213562

[2] Pankaj K Agarwal, Jeff Erickson, et al. 1999. Geometric range searching and its relatives. *Contemp. Math.* 223 (1999), 1–56.

[3] Noga Alon, Yossi Matias, and Mario Szegedy. 1999. The Space Complexity of Approximating the Frequency Moments. *J. Comput. Syst. Sci.* 58, 1 (1999), 137–147. https://doi.org/10.1006/jcss.1997.1545

[4] Jon Louis Bentley. 1979. Decomposable Searching Problems. *Inf. Process. Lett.* 8, 5 (1979), 244–251. https://doi.org/10.1016/0020-0190(79)90117-0

[5] Hans-Joachim Bungartz and Michael Griebel. 2004. Sparse grids. *Acta numerica* 13 (2004), 147–269.

[6] Bernard Chazelle. 1990. Lower Bounds for Orthogonal Range Searching II. The Arithmetic Model. *J. ACM* 37, 3 (1990), 439–463. https://doi.org/10.1145/79147.79149

[7] Graham Cormode, Minos N. Garofalakis, Peter J. Haas, and Chris Jermaine. 2012. Synopses for Massive Data: Samples, Histograms, Wavelets, Sketches. *Foundations and Trends in Databases* 4, 1-3 (2012), 1–294. https://doi.org/10.1561/1900000004

[8] Graham Cormode and S. Muthukrishnan. 2005. An improved data stream summary: the count-min sketch and its applications. *J. Algorithms* 55, 1 (2005), 58–75. https://doi.org/10.1016/j.jalgor.2003.12.001

[9] Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, Entong Shen, and Ting Yu. 2012. Differentially Private Spatial Decompositions. In *IEEE 28th International Conference on Data Engineering (ICDE 2012), Washington, DC, USA (Arlington, Virginia), 1-5 April, 2012.* 20–31. https://doi.org/10.1109/ICDE.2012.16

[10] Nicolas Delanoue, Luc Jaulin, and Bertrand Cottenceau. 2006. Using interval arithmetic to prove that a set is path-connected. *Theor. Comput. Sci.* 351, 1 (2006), 119–128. https://doi.org/10.1016/j.tcs.2005.09.055

[11] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3-4 (2014), 211–407. https://doi.org/10.1561/0400000042

[12] Joan Feigenbaum, Sampath Kannan, Martin Strauss, and Mahesh Viswanathan. 2002. An Approximate L1-Difference Algorithm for Massive Data Streams. *SIAM J. Comput.* 32, 1 (2002), 131–151. https://doi.org/10.1137/S0097539799361701

[13] Raphael A. Finkel and Jon Louis Bentley. 1974. Quad Trees: A Data Structure for Retrieval on Composite Keys. *Acta Inf.* 4 (1974), 1–9. https://doi.org/10.1007/BF00288933

[14] Philippe Flajolet, Éric Fusy, Olivier Gandouet, and Frédéric Meunier. 2007. Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm. In *Discrete Mathematics and Theoretical Computer Science*. Discrete Mathematics and Theoretical Computer Science, 137–156.

[15] Phillip B. Gibbons, Yossi Matias, and Viswanath Poosala. 1997. Fast Incremental Maintenance of Approximate Histograms. In *International Conference on Very Large Data Bases.* 466–475. http://www.vldb.org/conf/1997/P466.PDF

[16] J. H. Halton. 1964. Algorithm 247: Radical-inverse Quasi-random Point Sequence. *Commun. ACM* 7, 12 (Dec. 1964), 701–702. https://doi.org/10.1145/355588.365104

[17] Sariel Har-Peled and Micha Sharir. 2011. Relative (p, ε)-approximations in geometry. *Discrete & Computational Geometry* 45, 3 (2011), 462–496.

[18] Michael Hay, Vibhor Rastogi, Gerome Miklau, and Dan Suciu. 2010. Boosting the Accuracy of Differentially Private Histograms Through Consistency. *PVLDB* 3, 1 (2010), 1021–1032. https://doi.org/10.14778/1920841.1920970

[19] Joseph M. Hellerstein, Elias Koutsoupias, and Christos H. Papadimitriou. 1997. On the Analysis of Indexing Schemes. In *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, May 12-14, 1997, Tucson, Arizona, USA.* 249–256. https://doi.org/10.1145/263661.263688

[20] H. V. Jagadish, Nick Koudas, S. Muthukrishnan, Viswanath Poosala, Kenneth C. Sevcik, and Torsten Suel. 1998. Optimal Histograms with Quality Guarantees. In *Proceedings of 24rd International Conference on Very Large Data Bases.* 275–286. http://www.vldb.org/conf/1998/p275.pdf

[21] Luc Jaulin and Eric Walter. 1993. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica* 29, 4 (1993), 1053–1064. https://doi.org/10.1016/0005-1098(93)90106-4

[22] Sanjeev Khanna, S. Muthukrishnan, and Mike Paterson. 1998. On Approximating Rectangle Tiling and Packing. In *ACM-SIAM Symposium on Discrete Algorithms.* 384–393. http://dl.acm.org/citation.cfm?id=314613.314768

[23] Michel Kieffer, Isabelle Braems, Eric Walter, and Luc Jaulin. 2001. Guaranteed set computation with subpavings. In *Scientific Computing, Validated Numerics, Interval Methods*. Springer, 167–178.

[24] Elias Koutsoupias and David Scot Taylor. 1998. Tight Bounds for 2-Dimensional Indexing Schemes. In *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 1-3, 1998, Seattle, Washington, USA.* 52–58. https://doi.org/10.1145/275487.275494

[25] Yang D. Li, Zhenjie Zhang, Marianne Winslett, and Yin Yang. 2011. Compressive mechanism: utilizing sparse representation in differential privacy. In *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society, WPES 2011, Chicago, IL, USA, October 17, 2011.* 177–182. https://doi.org/10.1145/2046556.2046581

[26] Amit Manjhi, Vladislav Shkapenyuk, Kedar Dhamdhere, and Christopher Olston. 2005. Finding (Recently) Frequent Items in Distributed Data Streams. In *Proceedings of the 21st International Conference on Data Engineering, ICDE 2005, 5-8 April 2005, Tokyo, Japan.* 767–778. https://doi.org/10.1109/ICDE.2005.68

[27] Jiri Matousek. 2009. *Geometric discrepancy: An illustrated guide.* Vol. 18. Springer.

[28] Harald Niederreiter. 1987. Point sets and sequences with small discrepancy. *Monatshefte für Mathematik* 104, 4 (1987), 273–337.

[29] Vasilis Samoladas and Daniel P. Miranker. 1998. A Lower Bound Theorem for Indexing Schemes and Its Application to Multidimensional Range Queries. In *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 1-3, 1998, Seattle, Washington, USA.* 44–51. https://doi.org/10.1145/275487.275493

[30] I.M Sobol. 1967. Distribution of points in a cube and approximate evaluation of integrals. *U.S.S.R Comput. Maths. Math. Phys* 7 (1967), 86–112.

[31] Eric J. Stollnitz, Tony DeRose, and David Salesin. 1996. *Wavelets for computer graphics - theory and applications.* Morgan Kaufmann.

[32] Subhash Suri, Csaba D. Tóth, and Yunhong Zhou. 2006. Range Counting over Multidimensional Data Streams. *Discrete & Computational Geometry* 36, 4 (2006), 633–655. https://doi.org/10.1007/s00454-006-1269-4

[33] Yufei Tao, George Kollios, Jeffrey Considine, Feifei Li, and Dimitris Papadias. 2004. Spatio-Temporal Aggregation Using Sketches. In *Proceedings of the 20th International Conference on Data Engineering, ICDE 2004, 30 March - 2 April 2004, Boston, MA, USA.* 214–225. https://doi.org/10.1109/ICDE.2004.1319998

[34] Ernesto Tapia. 2011. A note on the computation of high-dimensional integral images. *Pattern Recognition Letters* 32, 2 (2011), 197–201. https://doi.org/10.1016/j.patrec.2010.10.007

[35] J. G. van der Corput. 1935. Verteilungsfunktionen. I. Mitt. *Proc. Akad. Wet. Amsterdam* 38 (1935), 813–821.

[36] Zhewei Wei and Ke Yi. 2013. The Space Complexity of 2-Dimensional Approximate Range Counting. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013.* 252–264. https://doi.org/10.1137/1.9781611973105.19

[37] Zhewei Wei and Ke Yi. 2018. Tight Space Bounds for Two-Dimensional Approximate Range Counting. *ACM Trans. Algorithms* 14, 2 (2018), 23:1–23:17. https://doi.org/10.1145/3205454

[38] Xiaokui Xiao, Guozhang Wang, and Johannes Gehrke. 2011. Differential Privacy via Wavelet Transforms. *IEEE Trans. Knowl. Data Eng.* 23, 8 (2011), 1200–1214. https://doi.org/10.1109/TKDE.2010.247

[39] Yonghui Xiao, Li Xiong, Liyue Fan, Slawomir Goryczka, and Haoran Li. 2014. DPCube: Differentially Private Histogram Release through Multidimensional Partitioning. *Trans. Data Privacy* 7, 3 (2014), 195–222. http://www.tdp.cat/issues11/abs.a136a13.php

[40] Yonghui Xiao, Li Xiong, and Chun Yuan. 2010. Differentially Private Data Release through Multidimensional Partitioning. In *Secure Data Management, 7th VLDB Workshop, SDM 2010, Singapore, September 17, 2010. Proceedings.* 150–168. https://doi.org/10.1007/978-3-642-15546-8_11

[41] Ke Yi, Lu Wang, and Zhewei Wei. 2014. Indexing for summary queries: Theory and practice. *ACM Trans. Database Syst.* 39, 1 (2014), 2:1–2:39. https://doi.org/10.1145/2508702

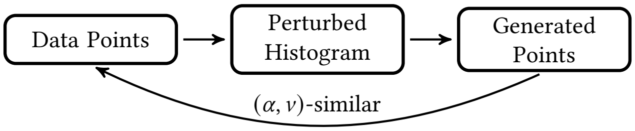
[42] Shuheng Zhou, Katrina Ligett, and Larry A. Wasserman. 2009. Differential privacy with compression. In *IEEE International Symposium on Information Theory, ISIT 2009, June 28 - July 3, 2009, Seoul, Korea, Proceedings.* 2718–2722. https://doi.org/10.1109/ISIT.2009.5205863

A APPLICATION: PRIVATE DATA PUBLISHING

In this appendix, we consider an application of data independent histograms for differentially private data publishing. Here, the aim is to publish a modified version of sensitive data from which meets a statistical privacy guarantee, while providing guarantees on the accuracy. For privacy preservation we utilise the widely used notion of differential privacy and mostly adapt existing tools, rather than introducing new techniques. For data preservation we extend ideas from α-binnings introduced in the previous sections:

Definition A.1 ((α, ν)-similarity). A generated set of points G is (α, ν)-similar to the original multiset of points O, if for each box in the space there exists an α-similar box s.t. the count of G in that box can be used as an unbiased estimator of the count of O with at most variance v.

In this section, we adopt the following workflow:



Achieving both small space discretisation error α and count estimation error v appears challenging if not impossible, but our goal here is to explore the trade-offs between both.

Differential privacy has arisen as the most popular notion for privacy-preserving data publication, because it offers a very strong notion of privacy. The intent is that from differentially-private statistics over a dataset it should not be possible to infer with confidence whether an individual participated in the dataset or not. A core use of differential privacy is to release statistics over histograms (typically counts). In this setting, data-independent histograms are of particular value, since they ensure that the focus can be on maximizing the accuracy of the counts, without requiring extra treatment to ensure that the description of the binning meets the privacy bounds.

The canonical approach to achieve differential privacy is the Laplace mechanism, which for histograms replaces counts with random variables following a Laplacian distribution. The variance of these random variables is chosen large enough to obscure an individual participating in a count.

In this section, we first deal with how to allocate privacy budget between overlapping bins, as the noise added to the counts has to match how much information is revealed for individual points. While one could naively split the privacy equally between all bins, it is not optimal with regards to v. Then we tackle how to ensure that the added noise does not cause inconsistencies by pooling multiple random variables. At the end, we compare different binnings for which trade-offs between α and v they are guaranteed to achieve for any dataset.

For a flat binning, each data point participates only in one bin count, but in arbitrary binnings, it contributes to multiple bin counts, which has to be accounted for when adding artificial noise to the counts. As overlapping grids allow better results for α, v, we study the problems of how to split the privacy budget between them when optimising for the worst-case, how to keep the counts consistent and how to generate points that match their counts.

A.1 Privacy Budget Allocation

As histograms with overlapping bins expose data points multiple times, the privacy budget needs to be allocated between overlapping bins, making use of sequential composition results.

Definition A.2. The Laplacian histogram mechanism with privacy budget allocation function μ replaces the count of each bin a with the random variable $\mathrm{Lap}(\mathrm{count}(a), \frac{1}{\mu(a)})$.

Definition A.3. A binning has DP-aggregate variance v if there exists a privacy allocation function μ that maps each bin to a real in $(0, 1]$ s.t. each set of intersecting bins S satisfy $\sum_{s \in S} \mu(s) \leq 1$ and for each query Q and answering bins $\mathcal{A}(Q)$ it holds that $\sum_{a \in \mathcal{A}(Q)} \frac{2}{\mu(a)^2} \leq v$.

Fact 2. *Let $L_0, L_1, \ldots, L_k$ with $k \geq 1$ be i.i.d. random variables with $L_j \sim \mathrm{Lap}(0, \sqrt{\frac{\lambda}{2}})$ and $X \sim \sum_{i=0}^{k} L_i$. Then $\mathrm{Var}(X) = k\lambda$.*

This fact follows since, for independent variables $X_1, \ldots, X_n$, it holds that $\mathrm{Var}(\sum_{i=1}^{n} X_i) = \sum_{i=1}^{n} \mathrm{Var}(X_i)$ and $\mathrm{Var}(L_i) = \lambda$.

The Laplace mechanism ensures differential privacy by adding a Laplacian random variable to each count. As range queries are composed by multiple bins whose counts have to be summed up, the count of range queries is the sum of random variables distributed by a Laplace distribution. As the random variables are independent of each other, the variance over the sum is simply the sum over the individual variances.

Fact 3. *Any binning with height h and at most β answering bins has DP-aggregate variance $v \leq 2h^2\beta$*

The result follows simply by setting $\mu(x) = \frac{1}{h}$ for every bin x. A better result can be achieved, if we take into account from which grid (or more generally flat binning) most answering bins come from, so that we can allot more privacy budget to that grid:

Definition A.4 (answering dimensions). Let $h \in \mathbb{N}$. A binning has answering dimensions $\{w_1, w_2, \ldots, w_h\}$ if there exists a set of flat binnings $\{F_1, F_2, \ldots, F_h\}$, s.t. for each query Q and answering bins $\mathcal{A}$ it is satisfied that $|\mathcal{A} \cap F_1| \leq w_1$, $|\mathcal{A} \cap F_2| \leq w_2, \ldots$, and $|\mathcal{A} \cap F_h| \leq w_h$.

Intuitively, the answering dimensions is a histogram that tells us how many answering bins come from distinct bins without telling us where exactly it comes from. This is sufficient to determine a worst-case guarantee for the aggregate variance:

Lemma A.5. *Any binning with answering dimensions $w_1, \ldots, w_h$ has DP-aggregate variance $v \leq 2(\sqrt[3]{w_1} + \sqrt[3]{w_2} + \ldots + \sqrt[3]{w_h})^3$.*

Proof. Find a set of flat binnings $\{F_1, F_2, \ldots, F_h\}$ s.t. for each query Q there is a set of answering bins $\mathcal{A}$ that satisfies $|\mathcal{A} \cap F_1| \leq w_1$, $|\mathcal{A} \cap F_2| \leq w_2, \ldots$, and $|\mathcal{A} \cap F_h| \leq w_h$. Letting μ_i denote the privacy allocation to buckets of flat binning F_i, we aim to minimize the resulting DP-aggregate variance; that is, minimize $\sum_i \frac{2w_i}{\mu_i^2}$ subject to the constraint $\sum_i \mu_i \leq 1$. Forming the Lagrangean:

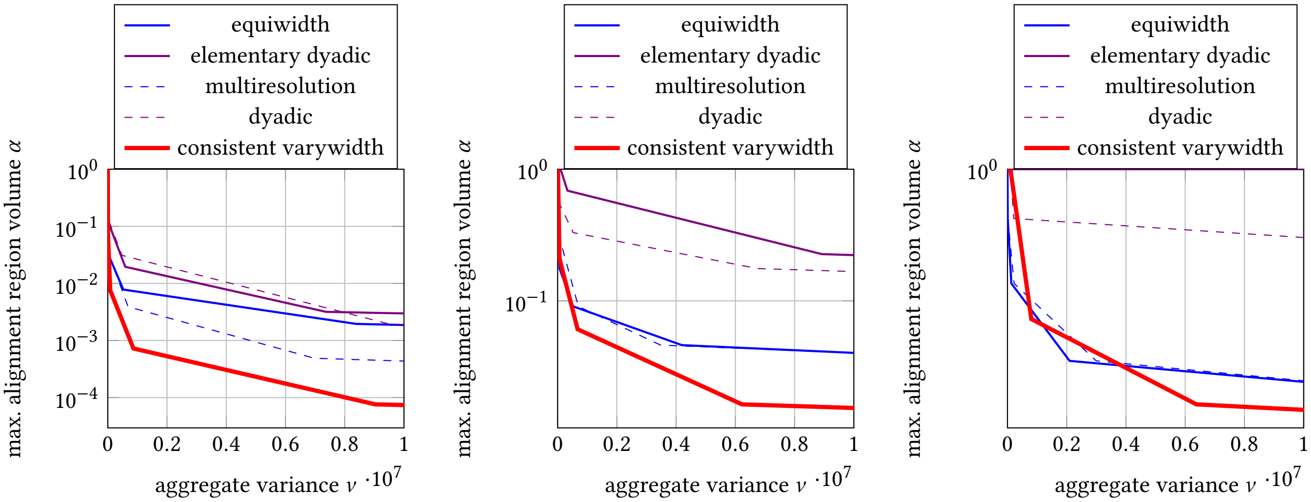
$$L(\mu_1, \ldots, \mu_h, \lambda) = \sum_i \frac{2w_i}{\mu_i^2} + \lambda(\sum_i \mu_i - 1),$$

and setting the partial derivatives $\frac{\partial L}{\partial \mu_i} = 0$, we find that the optimal privacy budget allocation is given by $\mu_i = \frac{\sqrt[3]{w_i}}{\sqrt[3]{w_1} + \sqrt[3]{w_2} + \ldots + \sqrt[3]{w_h}}$ for $i = 1, \ldots, h$. □

Adding noise to counts of binnings that are not flat can introduce inconsistencies. In the next section we show how noise can be added in a consistent way that can even reduce the overall variance.

A.2 Harmonised Bin Counts over Hierarchies

In this section we show how existing techniques can be used to achieve both consistent bin counts as well as not increasing the variance. For simplicity, we restrict ourselves to bins that follow a tree hierarchy:



(a) dimensionality $d = 2$

(b) dimensionality $d = 3$

(c) dimensionality $d = 4$

Figure 8: Differentially Private (DP) Aggregate variance of different α-binnings

Definition A.6 (tree binning). A tree binning is a binning that allows us to order the bins within a hierarchy, such that the bin corresponding to each node is the union of the bins at the children nodes.

Surprisingly, although the previously discussed binnings are very structured, only marginal binnings, multiresolution and equiwidth are tree binnings, whereas dyadic elementary and varywidth are not. However, we can convert varywidth to a tree binning by adding the coarser grid shared by the grids:

Definition A.7 (Consistent varywidth). A consistent varywidth binning is comprised of the usual d finer grids $\mathcal{G}_{C\ell\times\ell\ldots\times\ell}$, $\mathcal{G}_{\ell\times C\ell\ldots\times\ell}$, $\ldots$, $\mathcal{G}_{\ell\times\ell\ldots\times C\ell}$ along with the coarser grid $\mathcal{G}_{\ell\times\ell\ldots\times\ell}$,

In order to make bin counts from a tree binning consistent, we adapt the least-squares minimization from [18]. This effectively makes the counts $L_1, L_2, \ldots, L_k$ of k smaller bins consistent with the count L_0 over the larger bin (i.e., the union of the smaller bins), by subtracting the average over $L_1, L_2, \ldots, L_k$ and adding $\frac{L_0}{k}$, s.t., the sum over the new counts $L_1^*, L_2^*, \ldots, L_k^*$ will equal to L_0. Applying this concept to our setting, we similarly pool multiple noise terms together. We can then show, with a mild assumption on the privacy budget splitting, that the sum of variances does not increase, i.e., if $\mathrm{Var}(L_0)$ is at most k times larger than $\mathrm{Var}(L_j)$, it holds that $\mathrm{Var}(L_j^*) \leq \mathrm{Var}(L_j)$:

Lemma A.8. *Let $L_0, L_1, \ldots, L_k$ with $k \geq 1$ be a set of of i.i.d. random variables with $L_j \sim \mathrm{Lap}(0, \sqrt{\frac{\lambda}{2}})$ and $L_0 \sim \mathrm{Lap}(0, \sqrt{\frac{m\lambda}{2}})$ where $m \leq k$.*

For $L_j^* = L_j + (\frac{L_0 - \sum_{i=1}^k L_i}{k})$, the expected values remain the same, i.e., $\mathrm{E}[L_j^*] = \mathrm{E}[L_j]$ and $\mathrm{E}[\sum_{i=1}^k L_j^*] = \mathrm{E}[L_0]$, and the variances do not increase, i.e., $\mathrm{Var}(L_j^*) \leq \mathrm{Var}(L_j)$ and $\mathrm{Var}(\sum_{i=1}^k L_j^*) = \mathrm{Var}(L_0)$.

Proof. For independent variables $X_1, \ldots, X_n$ it holds that

$$\mathrm{Var}(\sum_{i=1}^n a_i X_i) = \sum_{i=1}^n a_i^2 \,\mathrm{Var}(X_i).$$

From $\mathrm{Var}(L_i) = \lambda$, it then follows that

$$\mathrm{Var}\left(L_j + \left(\frac{L_0 - \sum_{i=1}^k L_i}{k}\right)\right) = \mathrm{Var}\left(L_j \frac{k-1}{k} + \frac{L_0 - \sum_{i=1}^{j-1} L_i - \sum_{i=j+1}^k L_i}{k}\right)$$

$$= \left(\frac{k-1}{k}\right)^2 \underbrace{\lambda}_{\mathrm{Var}(L_j)} + \left(\frac{1}{k}\right)^2 \underbrace{(m\lambda)}_{\mathrm{Var}(L_0)} + (k-1)\left(\frac{1}{k}\right)^2 \underbrace{\lambda}_{\mathrm{Var}(L_i)}$$

$$= \frac{(k-1)^2 + (k-1) + m}{k^2}\lambda$$

$$= \frac{k(k-1) + m}{k^2}\lambda \leq \frac{k^2 - k + k}{k^2}\lambda \leq \lambda$$

□

A.3 Error tradeoff of binning schemes

To better understand the tradeoffs between volume errors (captured by α), and privacy noise (captured by the aggregate variance of the binning, as derived above), we plot these values for different binnings as we vary the number of bins. The log-log plot in Figure 8 shows the achieved spatial precision (on the y-axis) against the aggregate variance (on the x-axis).

In this setting, the binning techniques that do best require few bins, and also have a small height. This is best exemplified by consistent varywidth (varywidth with the additional grid containing the super regions of the other grids), which achieves both a better spatial accuracy as well as a better counting precision than other binnings. This achieves orders of magnitude better results than the standard dyadic and uniform grid approaches from the literature in 2 or 3 dimensions. The second choice method, “multiresolution”, is the subdyadic scheme that generalizes quadtrees.