Electronic Colloquium on Computational Complexity, Report No. 50 (2004)

Deterministic Clustering with Data Nets *

Michelle Effros Leonard J. Schulman

Abstract

We consider the K-clustering problem with the ℓ_2^2 distortion measure, also known as the problem of optimal fixed-rate vector quantizer design. We provide a deterministic approximation algorithm which works for all dimensions d and which, given a data set of size n, computes in time poly$(K)(d/\varepsilon)^{O(d)}n \log\log n + (d/\varepsilon)^{O(Kd)}$ a solution of distortion at most $1+\varepsilon$ times optimal. The key tool is construction of a new kind of representation called a "data net". A variety of applications of this object are discussed.

I Introduction

A striking lesson from the field of statistics is that important properties of a data set can be determined, to some precision, by examining a small, random subset of the data. In the field of clustering algorithms, specifically, a key technique is to choose a random subset of the input points, cluster that set, and then extend the clustering to the entire original input. This approach has enabled algorithms whose complexity scales very slowly as a function of the size of the data set. (See discussion and references in [7].) This approach can also be used to perform clustering on continuous distributions. For example, [19] applies a clustering optimized for n iid samples from distribution $p(x)$ to an independent sample drawn from the same distribution; results include a bound on the rate of convergence of the resulting expected performance to the optimal performance theoretically achievable on $p(x)$. The observation that we can learn about a complex distribution by examining a small random sample is important since large data sets characterize many clustering applications. However, algorithms based on sampling appear to be critically dependent on a source of random bits.

We study the K-clustering problem for ℓ_2^2 distortion, which is equivalently formulated as a problem in data compression. A fixed-rate vector quantizer of dimension d and rate $(\lg K)/d$ is a data compression system representing each vector in $\mathbb{R}^d$ by one of K possible vectors, known as "codewords" or "reproduction values." In the fixed-rate vector quantizer design problem with the ℓ_2^2 distortion measure, we are given a pdf $p(\mathbf{x})$ on $\mathbb{R}^d$, an integer $K \geq 1$, and the "squared error" distortion measure $\rho(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||^2$. An optimal vector quantizer is a set $\{\mu_1^\star, \ldots, \mu_K^\star\} \subset \mathbb{R}^d$ that achieves the minimal expected distortion $\int_{\mathbb{R}^d} p(\mathbf{x}) \min_{k\in\{1,\ldots,K\}} \rho(\mathbf{x}, \mu_k^\star) d\mathbf{x}$. While optimal scalar $(d = 1)$ fixed- and variable-rate quantizer design for discrete distributions can be accomplished in polynomial time [5, 25, 27, 28, 21] and algorithms for *locally* optimal design for arbitrary dimension d are well known (e.g., [18, 6]), *globally* optimal design for a distribution $p(\mathbf{x})$ that puts equal probability on every point in an n-point "data" or "training" set is NP-hard even for fixed-rate quantization with $K = 2$ [9]. We seek a deterministic ε-approximation algorithm for fixed-rate

*California Institute of Technology, Pasadena, CA 91125. The work of M. Effros (effros@caltech.edu) is partially supported by NSF Grant No. CCR-0220039 and a grant from Caltech's Lee Center for Advanced Networking. The work of L. J. Schulman (schulman@caltech.edu) is partially supported by NSF CAREER grant 0049092, the Charles Lee Powell Foundation, and a visiting membership at MSRI.

ISSN 1433-8092

vector quantizer design. Such an algorithm designs, for any $\varepsilon > 0$, a collection of K reproduction values yielding expected distortion within a factor of $(1+\varepsilon)$ of the optimal expected distortion.

Our algorithm works for any dimension d and finds a $(1+\varepsilon)$-optimal vector quantizer in time quasilinear in the size n of the training set. The algorithm also serves as an approximation algorithm for the d-dimensional, fixed-rate operational distortion-rate function. The approach is different from the random sampling approach described above in two key features. First, our approach is deterministic. Second, the key to our algorithm is in our reduction of the space of potential *outputs* ($\binom{\mathbb{R}^d}{K}$, the space of possible codebooks) rather than in a reduction of the algorithm's large *input* (the n-point training set). The list of potential codebooks we produce is short – its length is independent of the training set size – and is guaranteed to contain a near-optimal codebook.

There has been a large amount of work recently on various clustering tasks; this brief survey describes only the most directly related work, pertaining to ℓ_2^2 K-clustering. Randomized algorithms giving a $(1+\varepsilon)$-approximation to the ℓ_2 K-clustering problem for $d = 2$ in time $O(nKn^{O(1/\varepsilon)}\log n)$ and then extending to $d > 2$ in time $O(2^{O((1+\log(1/\varepsilon)/\varepsilon)^{d-1})}n(\log n)(\log K))$ appear in [2] and [17], respectively. Their problem differs from ours both in their assumption that the distortion measure is a metric (we treat the squared difference distortion measure) and in their requirement that the K codewords be training vectors. A poly-time randomized approximation algorithm for precisely our problem appears in [22], and a randomized solution running in time $O(\exp(\varepsilon^{-8}K^3(\ln K)(\ln\frac{1}{\varepsilon} + \ln K))n\log^K n)$ appears in [7]. (Similar results, though not for ℓ_2^2, appear in [3].)

Known deterministic algorithms for our problem are a factor of 2 approximation [9] running in time polynomial in n (and exponential in K and d), a poly-time constant-factor approximation even for the case of arbitrary K [16], and a $O(\varepsilon^{-2K^2d}n\log^K n)$-time ε-approximation [20].

The contribution of this paper is a fundamentally new approach to clustering problems. The key step is to identify a *finite* and *exhaustive* set of candidate codewords, which we call a "data net." "Finite" means that the size of this set is a function of K, ε, and d but *not* of n, the number of training vectors. "Exhaustive" means that only these points need to be considered as candidates for the locations of the codewords. Given such a set of candidate codewords, it is possible to directly test and compare all possible solutions in time linear in n or to find an approximation to the best codebook from the given set even more quickly.

While our first application of this new technique is to a family of ℓ_2^2 K-clustering problems, we believe that the new "net-point" approach has the potential to yield deterministic solutions for a much wider variety of clustering criteria than the narrowly-defined ℓ_2^2 criterion used in this paper. We limit our claims in this manuscript to the cases that have been proven. These cases include K-clustering and a variety of network vector quantization problems described in Section VII-B. All rely on the ℓ_2^2 distortion measure. The ℓ_2^2 version of K-clustering is widely motivated, in part because of applications involving mixtures of Gaussian sources, and in part due to metric embedding theorems [24]. Further, the ℓ_2^2 distortion measure remains the almost universal choice in practical applications.

A more precise description of the "net-point" approach follows.

Let Δ be the (unknown) expected distortion achievable by an optimal K-clustering, and define a *data net* to be a set $\mathcal{Z} \subset \mathbb{R}^d$, regions $\{A_{\mathbf{z}}\}_{\mathbf{z}\in\mathcal{Z}}$, and mapping $\zeta : \mathbb{R}^d \to \mathcal{Z}$ such that

1. *The additive property:* $\int_{A_{\mathbf{z}}} p(\mathbf{x})||\mathbf{x}-\mathbf{z}||^2 d\mathbf{x} \le \varepsilon\Delta/K$ for all $\mathbf{z} \in \mathcal{Z}$.

2. *The multiplicative property:* For any $\mu \in R^d$ and $\mathbf{x} \notin A_{\zeta(\mu)}$, $||\mathbf{x}-\zeta(\mu)||^2 \le (1+\varepsilon)||\mathbf{x}-\mu||^2$.

The additive property establishes that for subset $A_{\mathbf{z}}$ of $\mathbb{R}^d$, the cost of reproducing $\mathbf{x}$ by $\mathbf{z}$ rather than mapping the points in this region to their optimal codeword(s) increases the expected distortion by at most a small additive constant. The multiplicative property establishes that the cost of mapping $\mathbf{x}$ to $\zeta(\mu_k^\star)$ instead of optimal codeword $\mu_k^\star$ is bounded by a small multiplicative constant for all $\mathbf{x} \notin A_{\zeta(\mu_k^\star)}$.

Theorem 1 *The deterministic algorithm introduced in Section III-A designs a data net of size*

$$(1/\varepsilon)^{d+1} e^{d \log d + O(d)} \left(K^4 + K^2/\varepsilon^2\right)$$

within time

$$O\left(\left(K^2 e^{d\log d + O(d)} + Kd/\varepsilon\right) n \log\log n + (1/\varepsilon)^{d+1} e^{2d\log d + O(d)} \left(K^4 + K^2/\varepsilon^2\right)\right).$$

The algorithm can also perform efficient data net design (and therefore efficient vector quantizer design) for simply characterized continuous distributions, as discussed in Section VII-A.

Application of the data net to K-clustering yields the following theorem.

Theorem 2 *Given a data set of size n and a data net of size N, the deterministic K-clustering (VQ) algorithm of Section III-B finds a clustering of distortion $D \leq (1+\varepsilon)\Delta$ within time*

$$(1/\varepsilon)^{d+1} e^{O(d)} N^2 n + (1/\varepsilon)^{d+1} e^{O(d)} N^{K+2}.$$

Thus, by Theorem 1, the given algorithms perform data net construction followed by K-clustering in total time

$$\begin{aligned}\left(K^2 e^{d\log d + O(d)} + Kd/\varepsilon\right) n \log\log n + (1/\varepsilon)^{3(d+1)} e^{2d\log d + O(d)} (K^8 + K^4/\varepsilon^4) n \\ + e^{Kd \log(d/\varepsilon) + O(Kd + d\log d + K\log(K/\varepsilon))} \\ \leq \operatorname{poly}(K)(d/\varepsilon)^{O(d)} n \log\log n + (d/\varepsilon)^{O(Kd)}.\end{aligned}$$

Subsequent to codebook design, individual encodings can be performed in time $\log(K/\varepsilon)$ using [13].

The proofs of Theorems 1 and 2 appear in Sections III.

II Preliminaries

Given a distribution $p(\mathbf{x})$ on $\mathbb{R}^d$ and the squared error distortion measure $\rho(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||^2$, an optimal K-clustering is K points $\mu_1^\star, \ldots, \mu_K^\star$ such that

$$\{\mu_1^\star, \ldots, \mu_K^\star\} = \arg \min_{\{\mu_1,\ldots,\mu_K\}} \int_{\mathbb{R}^d} p(\mathbf{x}) \left[\min_{1\leq k\leq K} \rho(\mathbf{x}, \mu_k)\right] d\mathbf{x}.$$

We call each cluster center μ_k a *codeword*, each collection of K codewords a *codebook*, and each solution $\{\mu_1^\star, \ldots, \mu_K^\star\}$ to the above minimization an optimal codebook. Let

$$\Delta = \int_{\mathbb{R}^d} p(\mathbf{x}) \left[\min_{1\leq k\leq K} \rho(\mathbf{x}, \mu_k^\star)\right] d\mathbf{x}$$

denote the expected distortion of an optimal codebook. An optimal codebook for n points $\mathcal{T} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_n\}$ (known as a *data set* or *training set*) is simply an optimal codebook for the *empirical distribution* of $\mathcal{T}$ (which is the uniform distribution on the points of $\mathcal{T}$). The goal of an approximation algorithm is to find K points $\mu_1, ..., \mu_K$ such that

$$\int_{\mathbb{R}^d} p(\mathbf{x}) \left[\min_{1\leq k\leq K} \rho(\mathbf{x}, \mu_k)\right] d\mathbf{x} \leq (1+\varepsilon)\Delta$$

The following notation is useful in what follows. For a distribution p and a set of points S, let $\rho(p, S)$ be the integrated distortion of assigning p to nearest points in S. (For example, $\Delta = \min_{S=\{\mu_1,\ldots,\mu_k\}} \rho(\mathcal{T}, S)$.) When the distribution is understood, then for any $\mathcal{C} \subseteq \mathrm{I\!R}^d$, define

$$\rho(\mathcal{C}, \mathbf{z}) = \int_{\mathcal{C}} p(\mathbf{x})\rho(\mathbf{x}, \mathbf{z})d\mathbf{x} \qquad\qquad \mu(\mathcal{C}) = \arg\min_{\mathbf{z}} \rho(\mathcal{C}, \mathbf{z}).$$

For any points $\mathbf{t}, \mathbf{m} \in \mathrm{I\!R}^d$, let $\ell_\infty(\mathbf{t}, \mathbf{m})$ denote the maximal absolute coordinate of $\mathbf{t}$ with respect to coordinate axes centered at $\mathbf{m}$; more precisely, if $\mathbf{t} = (t_1, \ldots, t_d)$ and $\mathbf{m} = (m_1, \ldots, m_d)$, then $\ell_\infty(\mathbf{t}, \mathbf{m}) = \max\{|t_1 - m_1|, \ldots, |t_d - m_d|\}$. Finally, for any region $h \subseteq \mathrm{I\!R}^d$ and point $\mathbf{m} \in \mathrm{I\!R}^d$, let $\texttt{med}(\texttt{h}, \mathbf{m})$ be the median of $\{\ell_\infty(\mathbf{t}, \mathbf{m}) : \mathbf{t} \in h\}$ with respect to distribution

$$p_h(\mathbf{x}) = \begin{cases} \frac{p(\mathbf{x})}{\int_{\mathbf{x}\in h} p(\mathbf{x})} & \mathbf{x} \in h \\ 0 & \text{otherwise.} \end{cases}$$

III Algorithms

A The Data-Net Design Algorithm

1. Produce an estimate (denoted $\hat{\Delta}_0$) of Δ satisfying $\Delta \leq \hat{\Delta}_0 \leq 4n\Delta$, as described in Section IV-A.

2. Perform a binary search for the greatest integer i in the range $[0, 2 + \lg n]$ for which the space-partitioning algorithm of Section IV-B, run with argument $\hat{\Delta} = \hat{\Delta}_0/2^i$, does *not* report that $\hat{\Delta} < \Delta/2$.

   Let $\hat{\Delta}$, M, and the partition of space into regions $A_1, \ldots, A_M$ be as found by the space partitioning algorithm for the critical value of i.

3. Create a "data net" by placing several "clouds" of points around the centroid of each region A_m of the partition $\{A_1, \ldots, A_M\}$, as described in Section IV-C.

Proof of Theorem 1: Step 1 takes time $O(n \log K)$ by Lemma 1. Step 2 requires at most $\log \log n$ iterations, each of which takes time $O(n(Kd/\varepsilon + K^2 e^{d \log d + O(d)}))$ by Theorem 3. Step 3 takes time $O((1/\varepsilon)^{d+1} e^{d \log d + O(d)} (K^4 + K^2/\varepsilon^2))$ and yields a data net of size $(1/\varepsilon)^{d+1} e^{d \log d + O(d)} (K^4 + K^2/\varepsilon^2)$ by Theorem 4. The total runtime is $O((Kd/\varepsilon + K^2 e^{d \log d + O(d)}) n \log \log n + (1/\varepsilon)^{d+1} e^{d \log d + O(d)} (K^4 + K^2/\varepsilon^2))$. □

B The ε-Approximate K-Clustering (VQ) Algorithm

1. Design a data net of size N using the algorithm described in Section III-A.

2. Replace the input data $\mathcal{T}$ by a "reduced data set" $\tilde{\mathcal{T}}$ containing at most

$$N + (1/\varepsilon)^{d+1} e^{O(d)} N^2$$

   points, as described in Section V.

3. By exhaustive search, choose the best codebook for $\tilde{\mathcal{T}}$ using K codewords drawn from the data net.

Proof of Theorem 2: Given a data net $\mathcal{Z}$ for distribution p, the following argument shows that the best codebook from $\mathcal{Z}$ gives an ε-approximation for the optimal K-clustering for source p. Let $\{V_k^\star\}_{k=1}^K$ be the Voronoi cells for optimal codebook $\{\mu_k^\star\}_{k=1}^K$. Then optimal encoding with the best codebook $\{\mu_1, \ldots, \mu_K\} \subset \mathcal{Z}$ gives distortion

$$\begin{aligned}
D &= \int_{\mathbb{R}^d} p(\mathbf{x}) \min_{k\in\{1,\ldots,K\}} ||\mathbf{x} - \mu_k||^2 d\mathbf{x} \\
&\leq \int_{\mathbb{R}^d} p(\mathbf{x}) \min_{k\in\{1,\ldots,K\}} ||\mathbf{x} - \zeta(\mu_k^\star)||^2 d\mathbf{x} \\
&\leq \sum_{k=1}^K \int_{V_k^\star} p(\mathbf{x}) ||\mathbf{x} - \zeta(\mu_k^\star)||^2 d\mathbf{x} \\
&= \sum_{k=1}^K \left[\int_{V_k^\star \cap A_{\zeta(\mu_k^\star)}} p(\mathbf{x}) ||\mathbf{x} - \zeta(\mu_k^\star)||^2 d\mathbf{x} + \int_{V_k^\star - A_{\zeta(\mu_k^\star)}} p(\mathbf{x}) ||\mathbf{x} - \zeta(\mu_k^\star)||^2 d\mathbf{x} \right] \\
&\leq \sum_{k=1}^K \left[\frac{\varepsilon\Delta}{K} + (1+\varepsilon) \int_{V_k^\star - A_{\zeta(\mu_k^\star)}} p(\mathbf{x}) ||\mathbf{x} - \mu_k^\star||^2 d\mathbf{x} \right] \\
&\leq (1+2\varepsilon)\Delta.
\end{aligned}$$

The first inequality follows by definition since $\{\mu_1, \ldots, \mu_K\}$ is the best codebook from $\mathcal{Z}$; the second inequality follows since the Voronoi cells $V_1^\star, \ldots, V_K^\star$ for optimal codebook $\{\mu_1^\star, \ldots, \mu_K^\star\}$ may be suboptimal for codebook $\{\zeta(\mu_1^\star), \ldots, \zeta(\mu_K^\star)\}$; the third inequality follows from the additive and multiplicative properties of the data net; and the final inequality follows from the definition of Δ and the optimality of codebook $\{\mu_1^\star, \ldots, \mu_K^\star\}$.

Step 1 designs a data net of size $e^{d\log d + O(d)} M^4 (1/\varepsilon)^{3(d+1)}$ in time

$$O((Kd/\varepsilon + K^2 e^{d\log d+O(d)}) n \log\log n + (1/\varepsilon)^{d+1} e^{d\log d + O(d)} (K^4 + K^2/\varepsilon^2))$$

by Theorem 1. Given the size of the data net created in Step 1, Step 2 runs in time $O(n(Kd/\varepsilon + K^2 e^{d\log d + O(d)})^4 (1/\varepsilon)^{3(d+1)})$ and replaces the training set $\mathcal{T}$ of size n by the reduced training set $\tilde{\mathcal{T}}$ of size $\tilde{n} = (Kd/\varepsilon + K^2 e^{d\log d+O(d)})^4 (1/\varepsilon)^{3(d+1)}$. By Theorem 5, the optimal codebook $\{\mu_1, \ldots, \mu_K\} \subset \mathcal{Z}$ for $\tilde{\mathcal{T}}$ is an ε-approximation for the optimal codebook for $\mathcal{T}$. Step 3 uses an exhaustive search to find the best codebook $\{\mu_1, \ldots, \mu_K\} \subset \mathcal{Z}$ for $\tilde{\mathcal{T}}$ in time $O(K^{2K+4}(d/\varepsilon + Ke^{d\log d+O(d)})^{2K+4}(1/\varepsilon)^{(d+1)(K+3)})$. The total run time is

$$\begin{gathered}
\left(K^2 e^{d\log d + O(d)} + Kd/\varepsilon\right) n \log\log n + (1/\varepsilon)^{3(d+1)} e^{2d\log d + O(d)} (K^8 + K^4/\varepsilon^4) n \\
+ e^{Kd\log(d/\varepsilon) + O(Kd + d\log d + K\log(K/\varepsilon))}.
\end{gathered}$$

□

IV Deterministic Construction of a Data Net

This section details the steps used in designing the data net.

```
1. Find a hypercube $h$ such that $\rho(\mathbb{R} - h, \mu(\mathbb{R})) < \varepsilon\hat{\Delta}/K$.
2. Set $M = 1$. Run procedure split($h$).
```

Figure 1: The space partitioning algorithm.

A The Initial Estimate of Δ

Lemma 1 ([12, 15, 14, 11, 1]) *There is an algorithm that computes in time $O(n \log K)$ a number $\hat{\Delta}_0$ satisfying*

$$\Delta \leq \hat{\Delta}_0 \leq 4n\Delta.$$

Proof: The method is to approximate Δ by the square of the solution to the Euclidean K-center problem. In that problem, the goal is to find K codewords in $\mathbb{R}^d$ minimizing the greatest Euclidean distance from a data point to its nearest codeword. Equivalently, the problem is to find a smallest r and a set S so that $|S| = K$ and $\mathcal{T}$ is contained in the balls of radius r about members of S. Let $r^\star$ denote this smallest r. Observe that $(r^\star)^2/n \leq \Delta \leq (r^\star)^2$.

In [12] and [15, 14], Gonzalez and, independently, Hochbaum and Shmoys give an $O(nK)$ time algorithm that computes an S satisfying $r \leq 2r^\star$. Feder and Greene improve the run time to $O(n \log K)$ in [11] and show that beating factor 1.822 is NP-hard. A survey of these results and related work appears in [1]. Applying a 2-approximation for the K-center problem to give $r^\star \leq r \leq 2r^\star$ and setting $\hat{\Delta}_0 = r^2$ gives $\Delta \leq \hat{\Delta}_0 \leq 4n\Delta$. □

B The Space Partitioning Algorithm

Theorem 3 *The space partitioning algorithm either reports that $\hat{\Delta} < \Delta/2$ or creates a partition $\{A_1, \ldots, A_M\}$ of $\mathbb{R}^d$ with*

A. (Distortion within regions): $\max_{m \in \{1,\ldots,M\}} \rho(A_m, \mu(A_m)) \leq \varepsilon\hat{\Delta}/K$.

B. (Number of regions): $M \leq 2^{d+2}K/\varepsilon + K^2 e^{d \log d + O(d)}$.

C. (Runtime): The algorithm runs in time $O(n(Kd/\varepsilon + K^2 e^{d \log d + O(d)}))$.

Algorithm

The algorithm, given in Figure 1, finds a hypercube h outside of which the total distortion is small (step 1) and then recursively splits h (step 2). Let tree T describe the partitioning process. More precisely, set the root of T to $\mathbb{R}^d$ and let h and $\mathbb{R}^d - h$ be the children of $\mathbb{R}^d$. Each time a region h' is split, set the resulting subregions (split creates exactly 2^d of them) to be the children of h'. If the splitting procedure runs to completion (rather than reporting that $\hat{\Delta}$ is too small), then the leaves of the final tree T are the desired partition $\{A_1, \ldots, A_M\}$.

Procedure split, given in Figure 4, takes as its input a d-dimensional, axis-parallel hypercube. The routine recursively divides that hypercube into smaller regions until either all leaves of T meet the desired distortion constraint or the total number of leaves exceeds the bound on M. Global variable M, initialized to 1 on the first call of split, maintains a running count on the number of leaves in the current tree T. If T is not too large (step 1), then the division process begins by splitting hypercube h in half along each dimension to create 2^d equal sub-cubes (step 2). If one and

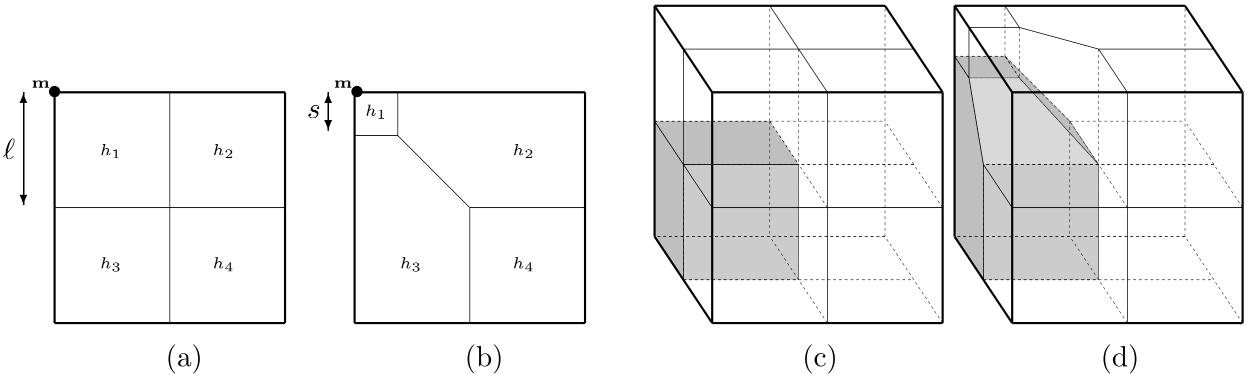

Figure 2: The region division and modification process in $\mathbb{R}^2$ and $\mathbb{R}^3$. Here (a) and (c) show an initial division, and (b) and (d) show the same division after modification. In (c) and (d), the bottom, rear subregion on the left is shaded to show its shape.

only one sub-cube, say h_1, has distortion exceeding $\varepsilon\hat{\Delta}/K$, then the algorithm shrinks the "heavy" region and grows its neighbors until at least one of those neighbors reaches the critical distortion or all regions become light (step 3).[1] The procedure then continues by recursively splitting any remaining heavy regions (step 4).

To describe step 3 in greater detail, let h_1 be the single heavy region in $\{h_1, \ldots, h_{2^d}\}$, $h_2, \ldots, h_{d+1}$ be the neighbors of h_1 (the sub-cubes that share faces with h_1), $\mathbf{m}$ be the outside corner of h_1 (the corner that is not shared with any other sub-cube h_m), and ℓ be the side length of h_1. (See Figure 2(a) and (c).) Shrinking h_1 to size $s < \ell$ is equivalent to replacing h_1 with the hypercube of side length s anchored at $\mathbf{m}$ ($h_1 := \{\mathbf{x} \in h : \ell_\infty(\mathbf{x}, \mathbf{m}) < s\}$) and expanding the neighbors $h_2, \ldots, h_{d+1}$ so that neighbor h_m, $2 \leq m \leq d+1$, acquires the convex span between the original face shared by h_1 and h_m and the new parallel face of h_1. (See Figure 2(b) and (d).)

Step 3 uses a binary search to iteratively narrow in on an optimal value for the side length s of h_1. Variables $\{(h_m, p_m, \mu_m, D_m)\}_{m=1}^{d+1}$ track the current knowledge of the regions and their probabilities ($p_m = \int_{h_m} p(\mathbf{x})d\mathbf{x}$), centroids ($\mu_m = \mu(h_m)$), and distortions ($D_m = \rho(h_m, \mu_m)$). Constants s_{in} and s_{out} track current bounds on the optimal value of s ($s \in [s_{\text{in}}, s_{\text{out}}]$). At each intermediate step, the algorithm maintains $h_1, \ldots, h_{d+1}$ at their smallest possible sizes given the current knowledge of s_{in} and s_{out}. That is, h_1 has sidelength s_{in} and $h_2, \ldots, h_{d+1}$ have length $2\ell - s_{\text{out}}$ in their longest dimension. Finally, $h_{\text{mid}} = \{\mathbf{x} \in h : \ell_\infty(\mathbf{x}, \mathbf{m}) \in (s_{\text{in}}, s_{\text{out}})\}$ describes the region between h_1 and $\cup_{m=2}^{d+1} h_m$. (See Figure 3.)

The initialization sets $h_{\text{mid}} := h_1$ (the initial hypercube of side length ℓ), $s_{\text{in}} := 0$, $s_{\text{out}} := \ell$, $(h_1, p_1, \mu_1, D_1) := (\{\}, 0, \mathbf{m}, 0)$, and $(h_m, p_m, \mu_m, D_m) := (h_m, \int_{h_m} p(\mathbf{x})d\mathbf{x}, \mu(h_m), \rho(h_m, \mu(h_m)))$ for all $m \in \{2, \ldots, d+1\}$. At iteration i, the algorithm splits h_{mid} at its median $s := \mathtt{med}(\mathtt{h}_{\text{mid}}, \mathbf{m})$, giving inner region $h_{\text{in}} := \{\mathbf{x} \in h_{\text{mid}} : \ell_\infty(\mathbf{x}, \mathbf{m}) < s\}$ and outer region $h_{\text{out}} := h_{\text{mid}} - h_{\text{in}}$. Let $h_{\text{out}}^{(m)}$ be the portion of h_{out} that lies in the convex span between the prior h_m and the hypercube of length s anchored at $\mathbf{m}$. (See Figure 3.) The algorithm calculates the distortions that would result if h_1 were grown to include h_{in} and $h_2, \ldots, h_{d+1}$ were grown to include h_{out}. More precisely, these are the distortions $\rho(h'_m, \mu(h'_m))$, where $h'_1 = h_1 \cup h_{\text{in}}$ and $h'_m = h_m \cup h_{\text{out}}^{(m)}$, $m \in \{2, \ldots, d+1\}$. If the maximal distortion over $h'_2, \ldots, h'_{d+1}$ is less than or equal to $\varepsilon\hat{\Delta}/K$, then for each $2 \leq m \leq d+1$, (h_m, p_m, μ_m, D_m) is updated to h'_m and its probability, centroid, and distortion, h_{mid} is updated

[1] This choice is important to the counting argument of Section VI-A since it keeps T relatively balanced.

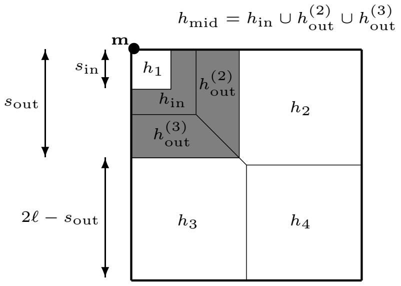

Figure 3: Searching for the optimal size of h_1. The shaded region is h_{mid}.

to h_{in}, and s_{out} is updated to s. Otherwise, (h_1, p_1, μ_1, D_1) is updated to h_1' and its probability, centroid, and distortion, h_{mid} is updated to h_{out}, and s_{in} is updated to s. The iterative procedure continues until finding a split location s where either $s_{\text{in}} = s_{\text{out}}$ or all regions are light or h_1 is heavy and its heaviest neighbor just reaches the threshold $\varepsilon\hat{\Delta}/K$. Since the probability of a training vector that lies on the boundary of two or more regions can be divided arbitrarily among those regions, $\max_{i \in \{2,\ldots,d+1\}} \rho(h_i', \mu(h_i'))$ is a continuous function of s and the proper choice of cell membership for training vectors lying on boundary $s_{\text{in}} = s_{\text{out}}$ yields regions meeting one of the previous two conditions (all regions light or at least two regions heavy).

While the above procedure creates some regions that are not hypercubes, the final choice of side length s allows non-hypercube regions to grow up to but not beyond distortion $\varepsilon\hat{\Delta}/K$; as a result, non-hypercube regions are never split, which is important since `split` requires a hypercube as its input.

Proof of Theorem 3:

A. (Distortion within regions): Property A follows immediately when the given construction runs through all splits (rather than stopping because the number of regions created exceeds the given bound on M).

B. (Number of regions): The proof of Section VI-A demonstrates that the number of regions required cannot exceed $2^{d+2}K/\varepsilon + K^2 e^{d \log d + O(d)}$ if $\hat{\Delta} \geq \Delta/2$.

C. (Runtime): Given a finite training set $\mathcal{T}$, let the bounding box h be the smallest box centered at $\mu(\mathbb{R}^d)$ that contains $\mathcal{T}$. The runtime for finding $\mu(\mathbb{R}^d)$ and the bounding box is $O(dn)$.

   The rest of the runtime equals the sum over calls to `split` of the runtime (excluding recursive calls) in each.

   Steps 1 and 2 of `split` together (including calculating the centroids and distortions of each of the 2^d regions) run in time $O(d2^d n)$.

   Step 3 involves many iterations, each with a new value of h_{mid}. The runtime of iteration i is proportional to $n_i d$, where n_i is the number of training vectors in h_{mid}. (Deterministic calculation of the median of n_i elements runs in time $O(n_i)$, specifically $2.96n_i + o(n_i)$ comparisons [4, 23, 8].) The number n_i halves in each iteration, so the total runtime of step 3 is $O(n'd)$ where n' is the number of points in the initial h_{mid}, which is h_1.

   The number of calls to `split` is the number of internal nodes in the tree T, which has degree 2^d and M leaves. The number of internal nodes, M', is related to M by $M' = (M-1)/(2^d-1) \leq$

```
1. If $M > 2^{d+2}K/\varepsilon + K^2 e^{d \log d + O(d)}$, then halt and report "$\hat{\Delta} < \Delta/2$"; else $M := M + 2^d - 1$
   endif.

2. Divide $h$ into $2^d$ axis-parallel hypercubes $h_1, \dots, h_{2^d}$ by cutting $h$ in half in each dimension.

3. If $\rho(h_m, \mu(h_m)) > \varepsilon\hat{\Delta}/K$, for exactly one $m \in \{1, \dots, 2^d\}$, let $h_1$ be that heavy region and
   $h_2, \dots, h_{d+1}$ be the neighbors of $h_1$.

   (a) Initialize: Set $h_{\text{mid}} := h_1$, $s_{\text{in}} := 0$, $s_{\text{out}} := \ell$, $p_{\text{mid}} := \int_{\mathbf{x} \in h_{\text{mid}}} p(\mathbf{x}) d\mathbf{x}$,
       $(h_1, p_1, \mu_1, D_m) = (\{\}, 0, \mathbf{m}, 0)$, and $\forall m \in \{2, \dots, d+1\}$,
       $(h_m, p_m, \mu_m, D_m) := (h_m, \int_{h_m} p(\mathbf{x}) d\mathbf{x}, \mu(h_m), \rho(h_m, \mu(h_m)))$.
   (b) Repeat:
             $s := \texttt{med}(\texttt{h}_{\text{mid}}, \mathbf{m})$
             $h_{\text{in}} := \{\mathbf{x} \in h_{\text{mid}} : \ell_\infty(\mathbf{x}, \mathbf{m}) < s\}$
             $h_{\text{out}} := h_{\text{mid}} - h_{\text{in}}$
             $\mu_{\text{in}} := \mu(h_{\text{in}})$
             $D_{\text{in}} := \rho(h_{\text{in}}, \mu_{\text{in}})$
             $p_{\text{mid}} := p_{\text{mid}}/2$
             $\mu'_1 := (p_1\mu_1 + p_{\text{mid}}\mu_{\text{in}})/(p_1 + p_{\text{mid}})$
             $D'_1 := D_1 + p_1\rho(\mu_1, \mu'_1) + D_{\text{in}} + p_{\text{mid}}\rho(\mu_{\text{in}}, \mu'_1)$
             for $m = 2$ to $d+1$,
                   $p^{(m)}_{\text{out}} := \int_{h^{(m)}_{\text{out}}} p(\mathbf{x}) d\mathbf{x}$
                   $\mu^{(m)}_{\text{out}} := \mu(h^{(m)}_{\text{out}})$
                   $D^{(m)}_{\text{out}} := \rho(h^{(m)}_{\text{out}}, \mu^{(m)}_{\text{out}})$
                   $\mu'_m := (p_m\mu_m + p^{(m)}_{\text{out}}\mu^{(m)}_{\text{out}})/(p_m + p^{(m)}_{\text{out}})$
                   $D'_m := D_m + p_m\rho(\mu_m, \mu'_m) + D^{(m)}_{\text{out}} + p^{(m)}_{\text{out}}\rho(\mu^{(m)}_{\text{out}}, \mu'_m)$
             end
             If $\max_{m \in \{2, \dots, d+1\}} D'_m \le \varepsilon\hat{\Delta}/K$,
             then
                   $s_{\text{out}} := s$,        $h_{\text{mid}} := h_{\text{in}}$
                   for $m = 2$ to $d+1$,
                         $(h_m, p_m, \mu_m, D_m) := (h_m \cup h^{(m)}_{\text{out}}, p_m + p^{(m)}_{\text{out}}, \mu'_m, D'_m)$
                   end
             else
                   $s_{\text{in}} := s$,        $h_{\text{mid}} := h_{\text{out}}$
                   $(h_1, p_1, \mu_1, D_1) := (h_1 \cup h_{\text{in}}, p_1 + p_{\text{mid}}, \mu'_1, D'_1)$
             endif
       until either  $s_{\text{in}} = s_{\text{out}}$
             or      $\max_{m \in \{1, \dots, d+1\}} D'_m \le \varepsilon\hat{\Delta}/K$
             or      $(D'_1 \ge \varepsilon\hat{\Delta}/K$ and $\max_{m \in \{2, \dots, d+1\}} D'_m = \varepsilon\hat{\Delta}/K)$.
       Set $h_1 := h_1 \cup h_{\text{in}}$, $h_m := h_m \cup h^{(m)}_{\text{out}}$ $\forall m \in \{2, \dots, d+1\}$.

4. For  $m = 1$ to $2^d$, if $\rho(h_i, \mu(h_i)) > \varepsilon\hat{\Delta}/K$, then $\texttt{split}(h_i)$, endif. end.
```

Figure 4: Procedure split(h).

1. Given an input set Z of size M, choose L to be a collection of at most $M-1$ distances such that every distance $||\mathbf{z}-\mathbf{z}'||$, $\mathbf{z},\mathbf{z}' \in Z$, is within a factor of $\sqrt{2}$ of some $\ell \in L$.
2. For each $(\mathbf{z},\ell) \in Z \times L$, create a "cloud" of net-points around $\mathbf{z}$; the inner radius of the cloud is $\ell\varepsilon$, the outer radius of the cloud is ℓ/ε, all net-points lie along lines of polar spacing ε, and the radial spacing is adjusted to match the polar spacing locally. See Figure 6. Output $C(Z)$ denotes the union of Z and all of the points in these clouds.

Figure 5: Cloud construction $C(Z)$.

$(2^{d+2}K/\varepsilon + K^2 e^{d\log d+O(d)} - 1)/(2^d-1) \le 8K/\varepsilon + K^2 e^{d\log d+O(d)}$. The total runtime of the procedure is $O(dn + d2^d + M'nd) \le O(n(Kd/\varepsilon + K^2 e^{d\log d+O(d)}))$.

□

C The Data Net Construction

Theorem 4 *Given a partition $\{A_1,\ldots,A_M\}$ satisfying the distortion constraints (property A) of Theorem 3, the following algorithm produces a data net of size at most*

$$M + (1/\varepsilon)^{d+1} e^{O(d)} M^2$$

in time

$$O\left(M + (1/\varepsilon)^{d+1} e^{O(d)} M^2\right).$$

The data net design algorithm relies on the cloud construction described below.

Cloud Construction

Given a set Z with M points from $\mathbb{R}^d$, the procedure creates M "clouds" of data net points around each of the M points in Z, giving M^2 clouds in total. Each cloud is parameterized by a constant $\ell \in L$. A cloud with parameter ℓ has inner radius $\varepsilon\ell$ and outer radius ℓ/ε. The set L is chosen to guarantee that for any pair of points $\mathbf{z},\mathbf{z}' \in Z$ there exists a cloud with parameter $\ell \in L$ within a constant factor of the distance $||\mathbf{z}-\mathbf{z}'||$. The notation $C(Z)$ designates the union of Z and the constructed cloud points. Figure 5 formalizes the cloud construction.

The following lemma is useful in restricting the size of L.

Lemma 1 *In an n-point metric space δ there are $n-1$ distances $\{\ell_i\}$ such that every distance $\delta(\mathbf{x},\mathbf{y})$ is within a factor of $\sqrt{2}$ of some ℓ_i.*

Proof: The proof begins with the complete undirected graph between points of the metric space and proceeds in steps $i = 1,...,n' \le n-1$ until no edges remain. In each step, select a shortest remaining edge, define L_i to be its length, and remove all remaining unselected edges of length at most $2L_i$. No combination of the selected edges can form a cycle, so at most $n-1$ edges can be selected before the graph is empty. Set $\ell_i = \sqrt{2}L_i$. □

It is interesting to notice that the bound of $n-1$ distances in Lemma 1 is optimal no matter how large the allowed constant factor.

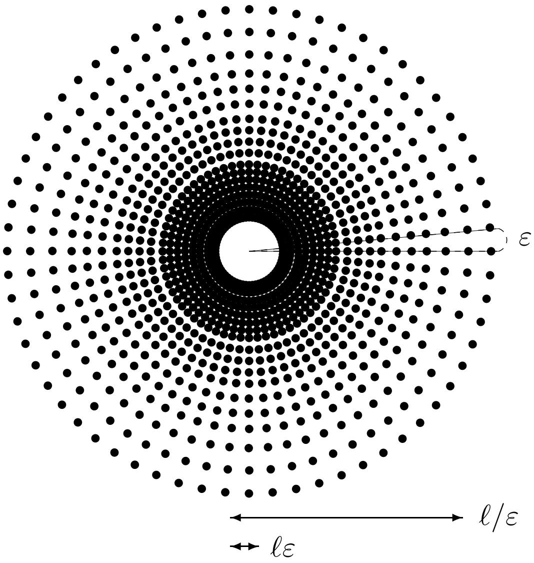

Figure 6: A sketch, for $d = 2$, of the net-point construction for a single value of $(\mathbf{z}, \ell)$.

Data Net Construction

For the data net construction, set $Z = \{\mu(A_1), \ldots, \mu(A_M)\}$, where $\{A_1, \ldots, A_M\}$ is the partition constructed in Section III-B. Apply the above cloud construction to get data net $\mathcal{Z} = C(Z)$.

Completing the data net description requires defining mapping $\zeta : \mathbb{R}^d \to \mathcal{Z}$ and regions $\{A_\mathbf{z}\}_{\mathbf{z}\in\mathcal{Z}}$.

Let $\zeta : \mathbb{R}^d \to \mathcal{Z}$ be defined as

$$\zeta(\mathbf{x}) = \arg\min_{\mathbf{z}\in\mathcal{Z}} \left[||\mathbf{x} - \mathbf{z}|| + \pi(\mathbf{z}) \right],$$

where penalty function $\pi(\mathbf{z}) = 0$ when $\mathbf{z} \in Z$ and $\pi(\mathbf{z}) = \varepsilon\ell$ for each $\mathbf{z} \in C(Z) - Z$ that belongs to a cloud with parameter ℓ.

Figure 7 illustrates the following definitions used to describe the regions $\{A_\mathbf{z}\}_{\mathbf{z}\in\mathcal{Z}}$. For any $\mathbf{z} \in \mathcal{Z}$, let $\mathbf{z}_1$ be a closest element of $Z - \{\mathbf{z}\}$ and let $\ell_1 = ||\mathbf{z} - \mathbf{z}_1||$. For any $\mathbf{x} \in \mathbb{R}^d$ and $r \in (0, \infty)$, $B(\mathbf{x}, r)$ denotes the closed ball of radius r about $\mathbf{x}$. Let $Z' = Z - B(\mathbf{z}_1, 2\varepsilon\ell_1)$. If $Z' \neq \phi$, let $\mathbf{z}_2$ be a closest element of Z' to $\mathbf{z}_1$ and let $\ell_2 = ||\mathbf{z}_1 - \mathbf{z}_2||$. Let H be the half-space of points closer to $\mathbf{z}$ than to $\mathbf{z}_1$. Define

$$A_\mathbf{z} = \begin{cases} B(\mathbf{z}, \ell_1/4) & \text{if } \mathbf{z} \in Z \\ H & \text{if } \mathbf{z} \in \mathcal{Z} - Z \text{ and } Z' = \phi \\ H \cap B(\mathbf{z}, \max\{\ell_1, \ell_2\}/4) & \text{if } \mathbf{z} \in \mathcal{Z} - Z \text{ and } Z' \neq \phi. \end{cases}$$

Figure 8 summarizes the data net construction.

Proof of Theorem 4: By Lemma 1, set L has size $M - 1$. Each cloud contains a number of points bounded above by $(1/\varepsilon)^{d+1} e^{O(d)}$. Thus the data net has size no greater than $M + (1/\varepsilon)^{d+1} e^{O(d)} M^2$. The proofs of the additive and multiplicative properties of $\mathcal{Z}$, ζ, and $\{A_\mathbf{z}\}_{\mathbf{z}\in\mathcal{Z}}$ are the topic of Section VI-B. □

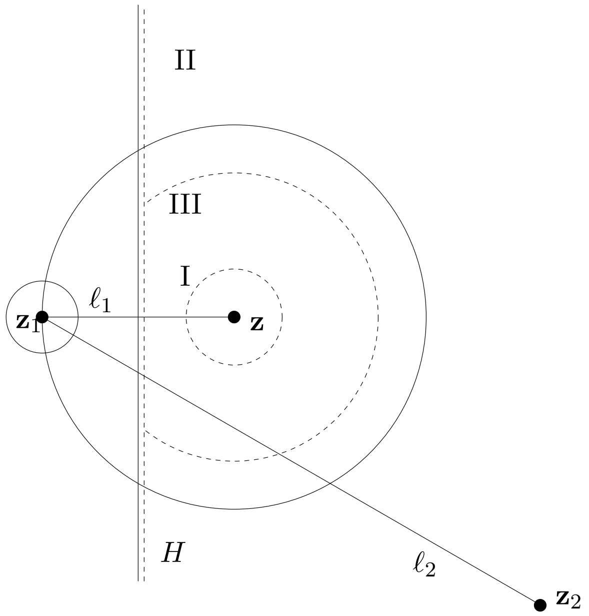

Figure 7: The definition of $A_{\mathbf{z}}$ under conditions I ($\mathbf{z} \in Z$), II ($\mathbf{z} \notin Z$ and $Z' = \phi$), and III ($\mathbf{z} \notin Z$ and $Z' \neq \phi$).

1. Let $Z = \{\mu(A_m)\}_{m=1}^{M}$ (assume that $\mu(A_i) \neq \mu(A_j)$ for all $i \neq j$).
2. Construct data net $\mathcal{Z} = C(Z)$, and define ζ and $\{A_{\mathbf{z}}\}_{\mathbf{z} \in \mathcal{Z}}$ as described in Section C.

Figure 8: Data net construction.

V Deterministic ε-Approximate K-Clustering

Recall that the K-clustering algorithm of Section III-B involves three steps: designing a data net, reducing the training set size, and then exhaustively finding the best codebook from the data net for the reduced training set. Only the second step, described below, requires further description.

Reducing the Data Set

In this section we show that an input distribution (or empirical distribution of a training set) p can, for the purpose of ε-approximate k-clustering, be replaced by a distribution $\tilde{p}$ of finite support (where "finite", as usual, means a size that depends on d, K, and ε, but is independent of the input distribution). Specifically, our construction, and our claim concerning it, are these:

Construction: *First form the set $Z = \{\mu(A_1), \ldots, \mu(A_M)\}$, where $\{A_1, \ldots, A_M\}$ is the partition whose design is described in Section III-B. Then form the data net $C(Z)$ as described in Section III-C. Then form the set $C(C(Z))$, and let π be the penalty function associated with the second stage of the construction (so $\pi(\mathbf{u}) = 0$ for $\mathbf{u} \in C(Z)$ and $\pi(\mathbf{u}) = \ell$ for all $\mathbf{u} \in C(C(Z)) - C(Z)$ constructed in a cloud of parameter ℓ). For $\mathbf{x} \in \mathbb{R}^d$, let $\eta(\mathbf{x})$ be a point $\mathbf{y} \in C(C(Z))$ minimizing $||\mathbf{x} - \mathbf{y}|| + \pi(\mathbf{y})$. Let*

$$\tilde{p}(\mathbf{y}) = \int_{\eta^{-1}(y)} p(\mathbf{x})\ d\mathbf{x}$$

Observe that the support of $\tilde{p}$ is of size at most $e^{O(d)} M^4 / \varepsilon^{3(d+1)}$.

The following theorem shows that for the purpose of ε-approximate K-clustering we can replace p by $\tilde{p}$.

Theorem 5 *Let $S = \{\mu_1, \ldots, \mu_k\} \subseteq C(Z)$, and let $\tilde{S} = \{\tilde{\mu}_1, \ldots, \tilde{\mu}_k\} \subseteq C(Z)$ be an ε-approximate best K-clustering of $\tilde{p}$ by $C(Z)$. Then $\rho(p, \tilde{S}) \leq (1 + O(\varepsilon))\rho(p, S)$.*

Proof: For any $\mathbf{x}$ we abbreviate notation by letting $\mathbf{y} = \eta(\mathbf{x})$, μ be the nearest point to $\mathbf{x}$ in S, and $\tilde{\mu}$ be the nearest point to $\mathbf{y}$ in $\tilde{S}$.

Let $L_1 = \int (||\mathbf{x} - \tilde{\mu}||^2 - ||\mathbf{x} - \mu||^2)\ d\mathbf{x}$. (We do not require that S be a best clustering for p using points of $C(Z)$, but if so, $0 \leq L_1$.)

Let $L_2 = \int (||\mathbf{y} - \tilde{\mu}||^2 - (1 + \varepsilon)||\mathbf{y} - \mu||^2)\ d\mathbf{x} \leq 0$. (We require that $\tilde{S}$ be an ε-approximate best clustering for $\tilde{p}$ using points of $C(Z)$, which is why $L_2 \leq 0$.)

What we need to show is that

$$L_1 \leq O(\varepsilon) \int ||\mathbf{x} - \mu||^2\ d\mathbf{x}. \tag{1}$$

It will suffice to show the evidently weaker

$$L_1 \leq O(\varepsilon) \int (||\mathbf{x} - \mu||^2 + ||\mathbf{x} - \tilde{\mu}||^2)\ d\mathbf{x}, \tag{2}$$

which is equivalent because $\int (||\mathbf{x} - \mu||^2 + ||x - \tilde{\mu}||^2)\ d\mathbf{x} = L_1 + 2 \int ||\mathbf{x} - \mu||^2\ d\mathbf{x}$, so equation 2 implies $L_1(1 - O(\varepsilon)) \leq O(\varepsilon) \int ||\mathbf{x} - \mu||^2\ d\mathbf{x}$ and in turn $L_1 \leq O(\varepsilon) \int ||\mathbf{x} - \mu||^2\ d\mathbf{x}$.

Next we note that since $L_2 \leq 0$, it suffices to show that

$$L_1 - L_2 \leq O(\varepsilon) \int (||\mathbf{x} - \mu||^2 + ||\mathbf{x} - \tilde{\mu}||^2)\ d\mathbf{x}.$$

It may seem that this makes our task needlessly harder, but the advantage comes in recasting the desired inequality as

$$\int (||\mathbf{x}-\tilde{\mu}||^2 - ||\mathbf{x}-\mu||^2 - ||\mathbf{y}-\tilde{\mu}||^2 + (1+\varepsilon)||\mathbf{y}-\mu||^2)\, d\mathbf{x} \le O(\varepsilon) \int (||\mathbf{x}-\mu||^2 + ||\mathbf{x}-\tilde{\mu}||^2)\, d\mathbf{x}.$$

in which we show that the inequality holds for each term $\mathbf{x}$.

We consider three cases according to the location of $\mathbf{x}$ relative to μ and $\tilde{\mu}$. Let $\ell = ||\mu - \tilde{\mu}||$.

Case 1: Either $||\mathbf{x}-\mu|| \le \varepsilon\ell$ or $||\mathbf{x}-\tilde{\mu}|| \le \varepsilon\ell$. Either implies $||\mathbf{x}-\mathbf{y}|| \le \varepsilon\ell$.

Case 1a: $||\mathbf{x}-\tilde{\mu}|| \le \varepsilon\ell$. Then $||\mathbf{x}-\mu|| \ge (1-\varepsilon)\ell$. The contribution of $\mathbf{x}$ to $L_1 - L_2$ is

$$(||\mathbf{x}-\tilde{\mu}||^2 - ||\mathbf{y}-\tilde{\mu}||^2) + ((1+\varepsilon)||\mathbf{y}-\mu||^2 - ||\mathbf{x}-\mu||^2) \le O(\varepsilon^2)\ell^2 + O(\varepsilon)\ell||\mathbf{x}-\mu|| \le O(\varepsilon||\mathbf{x}-\mu||^2).$$

Case 1b: $||\mathbf{x}-\mu|| \le \varepsilon\ell$. Then $||\mathbf{x}-\tilde{\mu}|| \ge (1-\varepsilon)\ell$. The contribution of $\mathbf{x}$ to $L_1 - L_2$ is

$$(||\mathbf{x}-\tilde{\mu}||^2 - ||\mathbf{y}-\tilde{\mu}||^2) + ((1+\varepsilon)||\mathbf{y}-\mu||^2 - ||\mathbf{x}-\mu||^2) \le ||\mathbf{x}-\tilde{\mu}||\ell O(\varepsilon) + O(\varepsilon^2)\ell^2 \le O(\varepsilon||\mathbf{x}-\tilde{\mu}||^2).$$

Case 2: Case 1 does not hold, and either $\varepsilon\ell < ||\mathbf{x}-\mu|| \le \ell/\varepsilon$ or $\varepsilon\ell < ||\mathbf{x}-\tilde{\mu}|| \le \ell/\varepsilon$.

This implies there is a $\mathbf{y}'$ for which

$$\pi(\mathbf{y}') + ||\mathbf{x}-\mathbf{y}'|| \le \varepsilon \min\{||\mathbf{x}-\mu||, ||\mathbf{x}-\tilde{\mu}||\},$$

either in the cloud about μ of radius (within $\sqrt{2}$ of) $||\mu-\tilde{\mu}||$, or in the cloud about $\tilde{\mu}$ of radius (within $\sqrt{2}$ of) $||\mu-\tilde{\mu}||$. The point $\mathbf{y}'$ may be different from $\mathbf{y}$, but its existence implies that $||\mathbf{x}-\mathbf{y}|| \le \varepsilon \min\{||\mathbf{x}-\mu||, ||\mathbf{x}-\tilde{\mu}||\}$. The contribution of $\mathbf{x}$ to $L_1 - L_2$ is

$$(||\mathbf{x}-\tilde{\mu}||^2 - ||\mathbf{y}-\tilde{\mu}||^2) + ((1+\varepsilon)||\mathbf{y}-\mu||^2 - ||\mathbf{x}-\mu||^2) \le (||\mathbf{x}-\tilde{\mu}||^2 + ||\mathbf{x}-\mu||^2)O(\varepsilon).$$

Case 3: $\ell/\varepsilon < ||\mathbf{x}-\mu||, ||\mathbf{x}-\tilde{\mu}||$. This implies that $\ell/\varepsilon < ||\mathbf{y}-\mu||, ||\mathbf{y}-\tilde{\mu}|| \le 2||\mathbf{x}-\mu||$. The contribution of $\mathbf{x}$ to $L_1 - L_2$ is

$$(||\mathbf{x}-\tilde{\mu}||^2 - ||\mathbf{x}-\mu||^2) + ((1+\varepsilon)||\mathbf{y}-\mu||^2 - ||\mathbf{y}-\tilde{\mu}||^2) \le (||\mathbf{x}-\mu||^2 + ||\mathbf{y}-\mu||^2)O(\varepsilon) \le ||\mathbf{x}-\mu||^2 O(\varepsilon).$$

VI Proofs

A The Partition Size Bound

This section aims bound the size M of the partition $\{A_1, \ldots, A_M\}$ designed in Section III-B provided that the input $\hat{\Delta}$ satisfies $\hat{\Delta} \ge \Delta/2$. Again, recall that tree T describes the complete partitioning process. The children of a node A of distortion greater than $\varepsilon\hat{\Delta}/K$ are the 2^d regions chosen to partition it. The tree leaves are the partition $A_1, \ldots, A_M$ of $\mathbb{R}^d$. Note that a node A is internal if and only if $\rho(A, \mu(A)) > \varepsilon\hat{\Delta}/K$. Let T' be the restriction of T to its internal nodes excluding the root. The bound on M follows from a bound on the number $M' = (M-1)/2^d$ of leaves in T'. The bound on M' follows by accounting for every leaf A' in one of two ways: "charge by volume" or "charge by distortion." The following notation is useful to that discussion.

Again let $B(\mathbf{x}, r)$ denote the closed ball of radius r about $\mathbf{x}$. $\partial B(\mathbf{x}, r)$ denote the boundary of $B(\mathbf{x}, r)$, $B^o(\mathbf{x}, r) = B(\mathbf{x}, r) - \partial B(\mathbf{x}, r)$ denote the corresponding open ball, and $B'(\mathbf{x}_1, \mathbf{x}_2) = B(\mathbf{x}_1, 10||\mathbf{x}_1 - \mathbf{x}_2||)$ describe a radius-$(10||\mathbf{x}_1 - \mathbf{x}_2||)$ ball around $\mathbf{x}_1$.

For a region A' that is to be charged by volume, the goal is to prove that there exist optimal codewords $\mu_i^\star$ and $\mu_j^\star$ such that $\mathrm{vol}(A' \cap B'(\mu_i^\star, \mu_j^\star))/\mathrm{vol}(B'(\mu_i^\star, \mu_j^\star)) > e^{-d\log d - O(d)}$; that is, A' occupies a constant fraction of $B'(\mu_i^\star, \mu_j^\star)$. At most $K^2 e^{d\log d + O(d)}$ regions can have this property. For a region A' that is to be charged by distortion, the goal is to prove that

$$\int_{A'} p(\mathbf{x}) \left[\min_{1\le k\le K} \rho(\mathbf{x}, \mu_k^\star)\right] d\mathbf{x} \ge \varepsilon\hat{\Delta}/(4K);$$

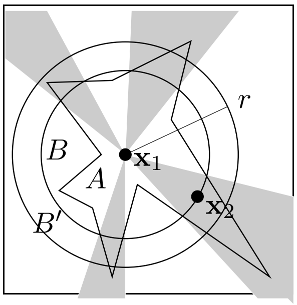

Figure 9: Region definitions used in the proof of Lemma 2. Here F is the intersection between A and $\partial(B)$. The shaded region is C (here bounded for display purposes), and C_B and $C_{B'}$ are the portions of the shaded region that lie in balls B and B', respectively.

that is, region A' contributes at least distortion $\varepsilon\hat{\Delta}/(4K)$ to the optimal performance. Since the optimal code achieves distortion Δ, at most $\Delta/(\varepsilon\hat{\Delta}/(4K))$ regions can have this property. The resulting bound is $M = 2^d M' + 1 \le 2^d(8K/\varepsilon + K^2 e^{d \log d + O(d)})$.

Let A' be some leaf of T'. The construction of A' implies that A' cannot be long and narrow – an important property for regions charged by volume. To make this observation precise, define the *fatness* of a region $A \subseteq \mathbb{R}^d$ [26], here denoted by fat(A), as follows

$$\mathrm{fat(A)} = \min_{\mathbf{x}_1,\mathbf{x}_2 \in \mathrm{A}} \frac{\mathrm{vol(A \cap B(}\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))}{\mathrm{vol(B(}\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))}.$$

Lemma 2 and Corollary 1 lead to the bound of leaf fatness fat(A′) given in Lemma 3. The specific form of the bound is not essential to the volume charging argument, which requires only that the lower bound be a positive function of the dimension d.

Lemma 2 shows that $\mathbf{x}_2$ can be chosen to lie on the boundary of A'.

Lemma 2 *Given any region A that is star convex relative to $\mathbf{x}_1$ and any $\mathbf{x}_2 \in A$,*

$$\frac{\mathrm{vol}(A \cap B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))}{\mathrm{vol}(B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))} \ge \frac{\mathrm{vol}(A \cap B(\mathbf{x}_1, r))}{\mathrm{vol}(B(\mathbf{x}_1, r))}$$

for all $r > ||\mathbf{x}_1 - \mathbf{x}_2||$.

Proof: Consider the following geometric quantities (illustrated in Figure 9)

$$\begin{aligned} B &= B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||) \qquad & C &= \{\mathbf{x} : \mathrm{ray}(\mathbf{x}_1, \mathbf{x}) \cap \mathrm{F} \ne \phi\} \text{ (shaded region)} \\ B' &= B(\mathbf{x}_1, r) & C_B &= C \cap B \\ F &= \partial(B) \cap A & C_{B'} &= C \cap B', \end{aligned}$$

where ray$(\mathbf{x}, \mathbf{y})$ denotes the ray originating at $\mathbf{x}$ and passing through $\mathbf{y}$. Then

$$\frac{\mathrm{vol}(A \cap B)}{\mathrm{vol}(B)} \ge \frac{\mathrm{vol}(C \cap B)}{\mathrm{vol}(B)} = \frac{\mathrm{area}(F)}{\mathrm{area}(\partial(B))} = \frac{\mathrm{vol}(C \cap (B' - B))}{\mathrm{vol}(B' - B)} \ge \frac{\mathrm{vol}(A \cap (B' - B))}{\mathrm{vol}(B' - B)},$$

where the final inequality follows from the star convexity of A. □

Corollary 1 *For any convex region A,*

$$\mathrm{fat(A)} = \frac{\mathrm{vol(A)}}{\mathrm{vol(B(\cdot, diam(A)))}}.$$

Proof: By Lemma 2, there exists an $(\mathbf{x}_1, \mathbf{x}_2) \in A^2$ such that $A \subseteq B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||)$ and

$$\frac{\mathrm{vol}(A \cap B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))}{\mathrm{vol}(B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))} = \mathrm{fat(A)}.$$

Thus any $(\mathbf{x}_1, \mathbf{x}_2) \in A^2$ that achieves the fatness fat(A) satisfies

$$\frac{\mathrm{vol}(A \cap B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))}{\mathrm{vol}(B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))} = \frac{\mathrm{vol}(A)}{\mathrm{vol}(B(\mathbf{x}_1, ||\mathbf{x}_1 - \mathbf{x}_2||))} \geq \frac{\mathrm{vol}(A)}{\mathrm{vol}(B(\mathbf{x}_1, \mathrm{diam}(A)))},$$

with equality if and only if $||\mathbf{x}_1 - \mathbf{x}_2|| = \mathrm{diam}(A)$. □

Corollary 1 leads to the bound on fat(A′) given in Lemma 3.

Lemma 3 *For any leaf A',*

$$\mathrm{fat(A')} \geq \mathrm{e^{-d \log d - O(d)}}.$$

Proof: By Corollary 1 and the construction of A', if d is even, then

$$\begin{aligned} \mathrm{fat(A')} &= \frac{\mathrm{vol}(A')}{\mathrm{vol}(B(\mathbf{0}, \mathrm{diam}(A')))} \\ &\geq \min_{t \in [0,1]} \frac{(d+1-t^d)/d}{\pi^{d/2}(d/2)!(d-1+(2-t)^2)^{d/2}} \\ &= \frac{d+1}{d\pi^{d/2}(d/2)!(d+3)^{d/2}} \end{aligned}$$

□

Use s to denote the "linear dimension" of region A', where the linear dimension of $A' \subseteq \mathrm{I\!R}^d$ is the side length of the largest axis-parallel, d-dimensional hypercube that lies within A'. Constant s serves as a simple approximation for the the diameter of A'; Lemma 4 shows their relationship more precisely.

Lemma 4 $\mathrm{diam}(A') \leq s\sqrt{d+3}$.

Proof: Given the tree construction, region A' is either a hypercube or a modified polyhedron of the type shown in Figure 2. The diameter of the region is maximized when the elongation is at its most extreme. In this case, the diameter equals the distance between the two furthest corners of an axis-parallel region with side-length s in $d-1$ dimensions and a side-length $2s$ in the last dimension, giving $\mathrm{diam}(A') \leq \sqrt{(d-1)s^2 + (2s)^2}$. □

The argument that follows uses a careful case analysis to decide whether to charge a particular A' by volume or by distortion. That case analysis relies on the following definitions:

$$\begin{aligned} \mu^\star(A') &= \arg \min_{1 \leq k \leq K} ||\mu(A') - \mu^\star_k|| \\ \ell &= ||\mu(A') - \mu^\star(A')|| \\ F &= B\left(\mu^\star(A'), \frac{\ell}{4}\right) - B^o(\mu(A'), \ell) \\ G &= B(\mu^\star(A'), 4\ell) - \left(B\left(\mu^\star(A'), \frac{\ell}{4}\right) \cup B(\mu(A'), \ell)\right) \\ H &= \mathrm{I\!R}^d - B(\mu^\star(A'), 4\ell). \end{aligned}$$

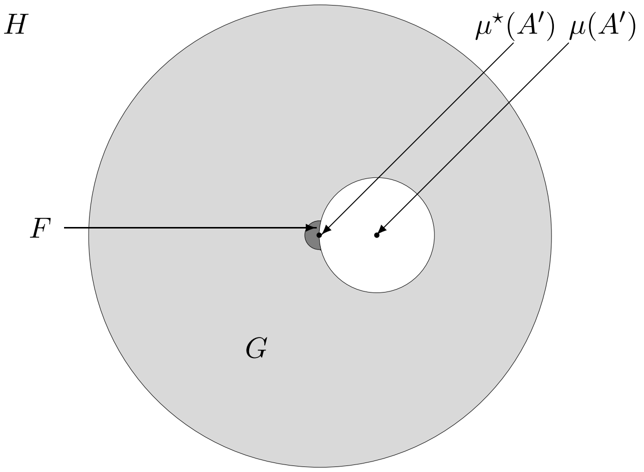

Figure 10: The regions F, G, and H used in the counting argument to estimate M.

Thus $\mu^\star(A')$ denotes the optimal codeword that is closest to centroid $\mu(A')$, ℓ denotes the distance of centroid $\mu(A')$ to its nearest optimal codeword, and F, G, and H are three non-overlapping regions in $\mathbb{R}^d$, as shown in Figure 10. Since F contains at least one optimal codeword, the following three values are well-defined.

$$\begin{aligned}
\mu^\star_F &= \arg \max_{\mu^\star \in \{\mu^\star_1,\ldots,\mu^\star_K\}\cap F} ||\mu^\star(A') - \mu^\star|| \\
\ell' &= ||\mu^\star(A') - \mu^\star_F|| \\
V &= \left\{ \mathbf{x} \in \mathbb{R}^d : \min_{\mu^\star \in \{\mu^\star_1,\ldots,\mu^\star_K\}\cap F} \rho(\mathbf{x}, \mu^\star) = \min_{\mu^\star \in \{\mu^\star_1,\ldots,\mu^\star_K\}} \rho(\mathbf{x}, \mu^\star) \right\}.
\end{aligned}$$

Case 1. $s \leq \ell/(2\sqrt{d+3})$**.** Charge A' by distortion. By Lemma 4 and the assumption, $\text{diam}(A') \leq s\sqrt{d+3} \leq \ell/2$. So no $\mathbf{x} \in A'$ can be closer than $\ell - \ell/2$, to its closest optimal reproduction, giving

$$\begin{aligned}
\int_{A'} p(\mathbf{x}) \left[\min_{1\leq k \leq K} \rho(\mathbf{x}, \mu^\star_k) \right] d\mathbf{x} &\geq \int_{A'} p(\mathbf{x}) \left(\frac{\ell}{2}\right)^2 d\mathbf{x} \\
&\geq \int_{A'} p(\mathbf{x}) \left(\text{diam}(A')\right)^2 d\mathbf{x} \geq \rho(A', \mu(A')) \geq \varepsilon\hat{\Delta}/K.
\end{aligned}$$

Case 2. $s > \ell/(2\sqrt{d+3})$ **and there is an optimal codeword** $\mu^\star_j$ **in** G**.** Charge A' by volume to the pair $\{\mu^\star(A'), \mu^\star_j\}$. The proof begins by showing that $B(\mu(A'), \ell) \subseteq B(\mu^\star(A'), \mu^\star_j)$ and that $B(\mu(A'), \ell)$ accounts for a constant fraction of the volume of $B'(\mu^\star(A'), \mu^\star_j)$. The argument goes on to show that A' accounts for a constant fraction of the volume of $B(\mu(A'), \ell)$.

By definition of G, $\ell/4 \leq ||\mu^\star(A') - \mu^\star_j|| \leq 4\ell$. Thus

$$\begin{aligned}
B(\mu(A'), \ell) &\subset B(\mu^\star(A'), 10\ell/4) \subset B'(\mu^\star(A'), \mu^\star_j) \\
\frac{\text{vol}(B(\mu(A'), \ell))}{\text{vol}(B'(\mu^\star(A'), \mu^\star_j))} &= \frac{\ell^d}{(10||\mu^\star(A') - \mu^\star_j||)^d} \geq \frac{\ell^d}{(40\ell)^d} = \frac{1}{(40^d)}.
\end{aligned}$$

If A' lies entirely within $B(\mu(A'), \ell)$, then $s > \ell/(2\sqrt{d+3})$ and the shape of A' imply

$$\frac{\text{vol}(A' \cap B(\mu(A'), \ell))}{\text{vol}(B(\mu(A'), \ell))} \geq \frac{s^d/d}{\pi^{d/2}\ell^d/(d/2)!} > \frac{c_2^{-d}}{d}$$

for some constant c_2 independent of A'. (The given form of the volume equation assumes d even. The equation is similar for d odd.) If A' extends outside of $B(\mu(A'),\ell)$, then there exists some point on $\mathbf{x} \in A'$ that lies on the outer boundary of $B(\mu(A'),\ell)$, and applying Lemma 3 with $\mathbf{x}_1 = \mu(A')$ and $||\mathbf{x}_1 - \mathbf{x}_2|| = \ell$ gives $\text{vol}(A' \cap B(\mu(A'),\ell))/\text{vol}(B(\mu(A'),\ell)) \geq c_1^{-d}/d$.

Case 3. $s > \ell/(2\sqrt{d+3})$**,** $\{\mu_1^\star, \ldots, \mu_K^\star\} \cap G = \phi$**, and** $A' \cap B(\mu^\star(A'), 3\ell') \neq \phi$**.** Charge A' by volume to $\{\mu^\star(A'), \mu_F^\star\}$. If $\ell' = 0$, then A' occupies the full volume of $B'(\mu^\star(A'), \mu_F^\star)$. Otherwise, apply Lemma 3 with $\mathbf{x}_1 \in B'(\mu^\star(A'), 3\ell') \cap A'$ and $\mathbf{x}_2 = \mu(A')$. Then $||\mathbf{x}_1 - \mathbf{x}_2|| \geq \ell - 3\ell' \geq \ell'$, and $\text{vol}(A' \cap B(\mathbf{x}_2, \ell'))/\text{vol}(B(\mathbf{x}_2, \ell')) \geq c_1^{-d}/d$. Since the radius of $B(\mathbf{x}_2, \ell')$ is 1/10 that of $B'(\mu^\star(A'), \mu_j^\star)$ and $B(\mathbf{x}_2, \ell') \subset B'(\mu^\star(A'), \mu_j^\star)$, the desired result is obtained.

Case 4. $s > \ell/(2\sqrt{d+3})$**,** $\{\mu_1^\star, \ldots, \mu_K^\star\} \cap G = \phi$**,** $A' \cap B(\mu^\star(A'), 3\ell') = \phi$**, and** $A' \subseteq V$**.** Charge A' by distortion. Bound

$$\begin{aligned} \int_{A'} \left[\min_{1 \leq k \leq K} \rho(\mathbf{x}, \mu_k^\star)\right] d\mathbf{x} &= \int_{A'} p(\mathbf{x}) \left[\min_{\mu^\star \in \{\mu_1^\star, \ldots, \mu_K^\star\} \cap F} \rho(\mathbf{x}, \mu^\star)\right] d\mathbf{x} \qquad (3) \\ &\geq \int_{A'} p(\mathbf{x}) \left[\frac{1}{4}\rho(\mathbf{x}, \mu^\star(A'))\right] d\mathbf{x} \geq \frac{1}{4}\rho(A', \mu(A')) \geq \frac{1}{4}\frac{\varepsilon\hat{\Delta}}{K}. \qquad (4) \end{aligned}$$

Here (3) follows from $A' \subseteq V$. The definition of ℓ' implies that all optimal codewords in F lie in a ball of radius ℓ' around $\mu^\star(A')$; since A' does not intersect the ball of radius $3\ell'$ around $\mu^\star(A')$, $(\min_{\mu^\star \in \{\mu_1^\star, \ldots, \mu_K^\star\} \cap F} ||\mathbf{x} - \mu^\star||)/||\mathbf{x} - \mu^\star(A')|| \geq 1/2$, giving the first inequality. The last two inequalities follow by definition of $\mu(A')$ and design of A', respectively.

Case 5. $s > \ell/(2\sqrt{d+3})$**,** $\{\mu_1^\star, \ldots, \mu_K^\star\} \cap G = \phi$**,** $A' \cap B(\mu^\star(A'), 3\ell') = \phi$**, and** $A' \not\subseteq V$**.** Here $A' \cap V \neq \phi$ by definition of $\mu^\star(A')$, and $A' \cap V^c \neq \phi$ by assumption. Thus there exist codewords $\mu_i^\star \in \{\mu_1^\star, \ldots, \mu_K^\star\} \cap F$ and $\mu_j^\star \in \{\mu_1^\star, \ldots, \mu_K^\star\} \cap H$ such that the intersection between the Voronoi regions for $\mu_i^\star$ and $\mu_j^\star$ runs through A'. Charge A' by volume to $\{\mu_i^\star, \mu_j^\star\}$. Let $\ell'' = ||\mu_i^\star - \mu_j^\star||$. By definition of F and H, $\ell'' \geq 15\ell/4$. Since the boundary between the Voronoi cells for $\mu_i^\star$ and $\mu_j^\star$ runs through A', $\max_{\mathbf{x} \in A'} ||\mu_i^\star - \mathbf{x}|| \geq \ell''/2$, giving $\max_{\mathbf{x} \in A'} ||\mu(A') - \mathbf{x}|| \geq \ell''/2 - 5\ell/4 \geq \ell''/6$. Applying Lemma 3 with $\mathbf{x}_1 = \mu(A')$ and $||\mathbf{x}_1 - \mathbf{x}_2|| \geq \ell''/6$ tells us that A' occupies some fraction $e^{-d\log d - O(d)}$ of the ball of radius $\ell'/6$ around $\mu(A')$. Since that ball falls entirely within the ball of radius $10\ell'$ around $\mu^\star(A')$ and occupies a constant fraction of its volume, the desired result is obtained.

B The Data Net Properties

The topic of this section is a proof that the choices of $\mathcal{Z}$, ζ, and $\{A_{\mathbf{z}}\}_{\mathbf{z} \in \mathcal{Z}}$ described in Section III-C meet the definition of a data net. Theorems 6 and 7 show that the data net satisfies the additive and multiplicative properties, respectively.

Theorem 6 (Additive analysis) *There exists a constant c such that $\rho(A_{\mathbf{z}}, \mathbf{z}) \leq c^d \varepsilon\hat{\Delta}/K$.*

Proof: The argument that follows shows that for every $\mathbf{z} \in \mathcal{Z}$, $A_{\mathbf{z}}$ intersects $M' \leq c^d$ of the regions in $\{A_1, \ldots, A_M\}$. Let $\{A_1, \ldots, A_{M'}\}$ be those intersecting regions. The definition of $A_{\mathbf{z}}$ gives $||\mathbf{x} - \mathbf{z}|| \leq (1+\varepsilon)||\mathbf{x} - \mu(A_m)||$ for all $m \in \{1, \ldots, M'\}$ and $\mathbf{x} \in A_{\mathbf{z}} \cap A_m$, and thus

$$\rho(A_{\mathbf{z}}, \mathbf{z}) \leq \sum_{m=1}^{M'} (1+\varepsilon)^2 \rho(A_{\mathbf{z}} \cap A_m, \mu(A_m)) \leq c^d \varepsilon\hat{\Delta}/K.$$

When $\mathbf{z} \in Z$, each A_m that intersects $A_{\mathbf{z}}$ must cross the annulus between $B(\mathbf{z}, \ell_1/4)$ and $B(\mathbf{z}, \ell_1)$. Thus by the shape of A_m, A_m has linear dimension at least $3\ell_1/(4\sqrt{d+3})$ and occupies an angular section of fraction at least c^{-d} of $\partial B(\mathbf{z}, 5\ell_1/8)$.

Let $P = \mathcal{Z} - Z$, the collection of points designed through the cloud construction. When $\mathbf{z} \in P$ and $Z' = \phi$, $Z \subseteq B(\mathbf{z}_1, 2\varepsilon\ell_1)$. Any A_m intersecting $A_\mathbf{z}$ has linear dimension $\geq (1/2 - 2\varepsilon)\ell_1/\sqrt{d+3}$ and occupies an angular section of fraction at least c^{-d} of $\partial B(\mathbf{z}_1, \ell_1/4)$.

When $\mathbf{z} \in P$ and $Z' \neq \phi$, let $Z_1 = Z \cap B(\mathbf{z}_1, 2\varepsilon\ell_1)$ and let $Z_2 = Z - Z_1$. For any A_m such that $\mu(A_m) \in Z_1$, the previous argument follows. For any A_m such that $\mu(A_m) \in Z_2$ and $A_m \cap \text{Near}(p) \neq \phi$, A_m intersects $B(\mathbf{z}, \max\{\ell_1, \ell_2\}/4)$ and $||\mu(A_m) - \mathbf{z}|| \geq \max\{\ell_1, \ell_2 - \ell_1\} \geq \max\{\ell_1, \ell_2/2\} \geq \max\{\ell_1, \ell_2\}/2$. Consequently, the linear dimension of A_m is at least $\max\{\ell_1, \ell_2\}/(2\sqrt{d+3})$, and A_m covers a constant fraction of $\partial B(\mathbf{z}, 3\max\{\ell_1, \ell_2\}/8)$. □

Theorem 7 proves that if $\mu_i^\star \in V(\mathbf{z})$, then $||\mathbf{x} - \mathbf{z}|| \leq ||\mathbf{x} - \mu^\star||/(1 - 4\varepsilon)$ for all $x \notin A_\mathbf{z}$, giving $\rho(\mathcal{C}_i^\star \cap A_\mathbf{z}^c, \mathbf{z}) \leq \rho(\mathcal{C}_i^\star \cap A_\mathbf{z}^c, \mu^\star)/(1 - \varepsilon)$. We first show Lemmas 2 and 3.

Lemma 2 *If $V(\mathbf{z}) \neq \emptyset$, then $\mathbf{z} \in V(\mathbf{z})$.*

Proof: Suppose $\mathbf{x} \in V(\mathbf{z})$. Then $||\mathbf{x} - \mathbf{z}|| + \pi(\mathbf{z}) \leq ||\mathbf{x} - \mathbf{z}||$ and $||\mathbf{x} - \mathbf{z}|| + \pi(\mathbf{z}) \leq ||\mathbf{x} - \mathbf{p}|| + \pi(\mathbf{p})$ for all $\mathbf{z} \in Z$ and $\mathbf{p} \in P$. By the triangle inequality $||\mathbf{x} - \mathbf{z}'|| - ||\mathbf{x} - \mathbf{z}|| \leq ||\mathbf{z} - \mathbf{z}'||$ for all $\mathbf{z}'$, so $\pi(\mathbf{z}) \leq ||\mathbf{z} - \mathbf{z}||$ and $\pi(\mathbf{z}) \leq ||\mathbf{z} - \mathbf{p}|| + \pi(\mathbf{p})$ for all $\mathbf{z} \in Z$ and $\mathbf{p} \in P$. □

Lemma 3 *If $\mathbf{z} \in Z$ generates a cloud of parameter ℓ, then for any $\mathbf{z} \in B(\mathbf{z}, \ell(1 - \varepsilon)/\varepsilon) \cap P$, $\text{diam}(V(\mathbf{z})) = O(\varepsilon \max\{\ell, ||\mathbf{z} - \mathbf{z}||\})$.*

Proof: Any $\mathbf{z} \in B(\mathbf{z}, (1 + \varepsilon)\varepsilon\ell)$ lies between $\mathbf{x}$ and a radius-$\varepsilon\ell$ sphere of net-points. Any other $\mathbf{z} \in B(\mathbf{z}, (1 - \varepsilon)(\ell/\varepsilon))$ lies among net-points spaced at order $\varepsilon||\mathbf{z} - \mathbf{z}||$ distances. □

Theorem 7 (Multiplicative analysis) *There exists a constant c such that for each $\mathbf{z} \in \mathcal{Z}$ and $\mathbf{x} \notin A_\mathbf{z}$, $B(\mathbf{x}, (1 - c\varepsilon)||\mathbf{x} - \mathbf{z}||) \cap V(\mathbf{z}) = \emptyset$.*

Proof: If $\mathbf{z} \in Z$, then $V(\mathbf{z}) \subseteq B(\mathbf{z}, \varepsilon\ell_1)$. So if $\mathbf{x} \notin A_\mathbf{z} = B(\mathbf{z}, \ell_1/4)$, then $||\mathbf{x} - \mathbf{z}|| - \varepsilon\ell_1 \geq ||\mathbf{x} - \mathbf{z}|| - \varepsilon(4||\mathbf{x} - \mathbf{z}||) = (1 - 4\varepsilon)||\mathbf{x} - \mathbf{z}||$, and $V(\mathbf{z})$ is disjoint from $B(\mathbf{x}, (1 - 4\varepsilon)||\mathbf{x} - \mathbf{z}||)$.

If $\mathbf{z} \in P$ and $Z' = \phi$, then $\mathbf{z}$ is created in a cloud about some $\mathbf{z} \in B(\mathbf{z}_1, 2\varepsilon\ell_1)$. Suppose that $V(\mathbf{z}) \subseteq L = \{\mathbf{y} \in \mathbb{R}^d : ||\mathbf{y} - \mathbf{z}|| \geq ||\mathbf{y} - \mathbf{z}|| + \ell_1(1 - \varepsilon)\}$ (a set with a hyperbolic boundary). The desired result follows by a geometric argument that shows that for $\mathbf{x} \notin H$, $B(\mathbf{x}, (1 - 4\varepsilon)||\mathbf{x} - \mathbf{z}||)$ does not intersect L. Suppose instead that $V(\mathbf{z})$ extends outside L; then the following argument gives a contradiction. If $V(\mathbf{z}) \cap L^c \neq \phi$, then $\pi(\mathbf{z}) \leq \varepsilon\ell_1$, point $\mathbf{z}$ is generated in a cloud of parameter $\ell \leq \ell_1$ around $\mathbf{z}$, and $\mathbf{z}$ is not on the inner sphere of the cloud in which it is generated. In this case, the cloud generating $\mathbf{z}$ has points on the sphere of radius $||\mathbf{z} - \mathbf{z}||$ around $\mathbf{z}$ and also on the sphere of radius $(1 - \varepsilon)||\mathbf{z} - \mathbf{z}||$ around $\mathbf{z}$. These points have the same penalty as $\mathbf{z}$ and are spaced regularly enough so $V(\mathbf{z})$ is contained in a wedge-shaped region entirely contained within L.

If $\mathbf{z} \in P$, $Z' \neq \phi$, and $\mathbf{z}$ is created by some $\mathbf{z} \in B(\mathbf{z}_1, 2\varepsilon\ell_1)$, then the reasoning from the previous case implies $V(\mathbf{z}) \subseteq L$. That handles the multiplicative distortion for $\mathbf{x} \notin H$; it remains to handle the multiplicative distortion for $\mathbf{x} \notin B(\mathbf{z}, \max\{\ell_1, \ell_2\}/4)$. Observe that $\mathbf{z} \in B(\mathbf{z}_1, \ell_2/(2\varepsilon))$; therefore, $\mathbf{z}$ is internal to the cloud of parameter ℓ_2 about $\mathbf{z}_1$. So by Lemma 3, $V(\mathbf{z})$ has diameter $O(\varepsilon \max\{\ell_1, \ell_2\})$ and so $B(\mathbf{x}, (1 - 4\varepsilon)||\mathbf{x} - \mathbf{z}||) \cap B(\mathbf{z}, \max\{\ell_1, \ell_2\}/4)$.

If $\mathbf{z} \in P$, $Z' \neq \phi$, $\mathbf{z}$ is created by some $\mathbf{z} \notin B(\mathbf{z}_1, 2\varepsilon\ell_1)$, and $\ell_2 \leq 4\ell_1$, then the existence of $\mathbf{z}_2$ again implies that $\mathbf{z} \in B(\mathbf{z}_1, \ell_2/(2\varepsilon))$ and is internal to the cloud of parameter ℓ_2 about $\mathbf{z}_1$; thus by the previous argument, $V(\mathbf{z})$ has diameter $O(\varepsilon \max\{\ell_1, \ell_2\}) = O(\varepsilon\ell_1)$. If $\mathbf{x} \notin H \cap B(\mathbf{z}, \max\{\ell_1, \ell_2\}/4)$, then in particular $||x - q|| \geq \ell_1/4$. So for a suitable c, dictated by the bound on $\text{diam}(V(q))$, $B(\mathbf{x}, (1 - c\varepsilon)||\mathbf{x} - \mathbf{z}||) \cap V(\mathbf{z}) = \emptyset$.

If $\mathbf{z} \in P$, $Z' \neq \phi$, and $\mathbf{z}$ is created by some $\mathbf{z} \notin B(\mathbf{z}_1, 2\varepsilon\ell_1)$, and $\ell_2 > 4\ell_1$ (so $\mathbf{z} \notin B(\mathbf{z}_1, 4\ell_1)$), then we want to show that $\text{diam}(V(\mathbf{z})) = O(\varepsilon\ell_1)$. This suffices since (always) $B(\mathbf{z}, \ell_1/4) \subseteq A_\mathbf{z}$.

Let ℓ be the parameter of the cloud that generated $\mathbf{z}$. Note $\ell/\varepsilon \geq \ell_2 - \ell_1 \geq 3\ell_2/4$, so $\ell \geq 3\varepsilon\ell_2/4$. If $\ell > \ell_1/\varepsilon$ then $\pi(\mathbf{z}) > \ell_1$ so by lemma 2, $V(\mathbf{z})$ is empty and we're done. It remains to consider the case $3\varepsilon\ell_2/4 \leq \ell \leq \ell_1/\varepsilon$. There is a cloud about $\mathbf{z}_1$ of parameter ℓ, hence of inradius at most ℓ_1 and outradius at least $3\ell_2/4$ which in turn is at least $3\ell_1$. This cloud's penalty is $\varepsilon\ell$, equal to the penalty of $\mathbf{z}$. If $\ell \leq \ell_1(1-\varepsilon)/\varepsilon$ then the inradius is $\leq \ell_1(1-\varepsilon)$, so $\mathbf{z}$ is surrounded in all directions, at a spacing of $\varepsilon\ell_1$, by points of penalty equal to its own; hence $\mathrm{diam}(V(\mathbf{z})) = O(\varepsilon\ell_1)$. On the other hand if $\ell_1(1-\varepsilon)/\varepsilon < \ell \leq \ell_1/\varepsilon$ then $\pi(\mathbf{z}) > \ell_1(1-\varepsilon)$ so $V(\mathbf{z})$ cannot extend past the hyperplane perpendicular to $\overline{\mathbf{z}_1\mathbf{z}}$, at distance $\varepsilon\ell_1$ from $\mathbf{z}$ and distance $(1-\varepsilon)\ell_1$ from $\mathbf{z}_1$; while outside of $B(z_1, \ell_1)$, $\mathbf{z}$ is surrounded in all directions, at a spacing of $\varepsilon\ell_1$, by points of penalty equal to its own. So in this case too, $\mathrm{diam}(V(\mathbf{z})) = O(\varepsilon\ell_1)$. □

VII Discussion

A Continuous Source Distributions

Our method does not rely in a very essential way on the finiteness of the data set. One of the design principles of our algorithm, which leads to its quasilinear runtime, is that the algorithm should access the data distribution only through some elementary types of queries, and that the runtime should be expressible in terms of the numbers of these queries. Those queries are: given a polyhedral region A and a point $\mathbf{x}$, compute $\mu(A)$, $\rho(A, \mu(A))$ and $\rho(A, \mathbf{x})$; and given a polyhedral region A, sweep some of the walls of A until $\rho(A, \mu(A))$ reaches some prescribed value. In addition the algorithm requires a coarse initial estimate $\hat{\Delta}_0$ on Δ and a suitable initial bounding box for the distribution, such that for the exterior A^c of that bounding box, $\rho(A^c, \mu(A))$ is less than $\varepsilon\Delta/K$. These are the only ingredients we need for the method.

B More Applications of Data Nets

Sections III–IV treat the data net design algorithm and its application to vector quantizer design. The vector quantizers introduced there are compression algorithms designed for networks where a single transmitter compresses information at a fixed rate of $\log K$ bits per vector, and the information is decoded by a single receiver. While the data net design algorithm originated from a search for a solution to that classic problem, it applies much more widely. We treat more applications in [10]. A summary of a few simple examples from [10] follows. These examples are of particular interest to the data compression community.

We again consider lossy compression problems with fixed-rate coding solutions. We move, however, from systems where a single transmitter sends information to a single receiver (so-called *point-to-point networks*) to more general network communication environments. In particular, we treat the following examples.

1. **Multiresolution vector quantization:** One transmitter sends a data description to L receivers. Receiver ℓ receives only the first $\sum_{i=1}^{\ell} \log K_i$ bits of the binary description. The goal in code design is to minimize the weighted sum of the distortions resulting from the L reconstructions. This problem is known as the *multiresolution* source coding problem.

2. **Multiple description vector quantization:** One transmitter sends L descriptions (also called *packets*) of a single source across an unreliable channel. Each packet is either received perfectly by the decoder or is entirely lost in transmission. The goal in code design is to make the expected distortion of the source reconstruction as small as possible; the expectation is taken with respect to a known distribution on the 2^L possible packet-loss scenarios.

3. **Side information vector quantization:** One transmitter describes a single source to a single receiver. The receiver observes side information unavailable to the decoder. The side information may be helpful in reconstructing the desired source. The goal in code design is to minimize the expected distortion of the reconstruction; the expectation is taken with respect to the joint distribution on the source and side information. Side information source coding is also called Wyner-Ziv source coding after the work of Wyner and Ziv [29] describing the optimal performance of a side information source code when the coding dimension d is allowed to grow without bound.

4. **Broadcast vector quantization:** A single transmitter describes multiple sources to a family of decoders. Each source is intended for a distinct subset of the receivers, and each component of the description is received by a distinct subset of the receivers. Each receiver uses all of its received descriptions to reconstruct all of its desired sources. The goal in code design is to minimize the weighted sum of the distortions of the reconstructions built by the receivers.

5. **Joint source-channel vector quantization:** A single transmitter describes a single source to a single receiver using a fixed-rate description of rate $\log K$ bits per vector. Due to channel noise, the source description index $i \in \{0, 1 \ldots, K-1\}$ may be corrupted during the transmission process and received as some distinct index j from the same alphabet. The goal in code design is to minimize the expected distortion in the source reproduction. Here the expectation is taken with respect to the known probability $p(i, j)$ of receiving index j when index i is transmitted.

6. **Joint source-channel vector quantization for the side-information network:** This scenario combines the side-information and joint source-channel vector quantizers, designing a code for the side-information coding environment in the presence of a noisy channel. The goal in code design is to minimize the expected distortion subject to both the joint distribution on the source and side information and the probability $p(i, j)$ that transmitted index i is received by the decoder as index j.[2]

7. **Remote source vector quantization:** An encoder observes a noisy copy of the true source and describes it to the decoder. The decoder reconstructs the true source as accurately as possible.

The simple examples described above can be combined to create an even richer set of coding scenarios (multiresolution multiple access codes, multiple description broadcast codes, and so on).

In each example, let S be the number of sources ($S = 1$ in all but the BCVQ), K be the maximal number of codewords that can be distinguished by any single decoder, T be the total number of single-source codewords in the code, and Δ be the optimal performance (usually an expected distortion over some distribution on the reconstructions at different receivers).

Theorems 8–14 *In each of the above scenarios, designing a data net of size*

$$N \leq (1/\varepsilon)^{d+1} e^{d \log d + O(d)} \left(K^4 + K^2/\varepsilon^2\right)$$

for each source, and using the best T codewords from those data nets gives a code with total distortion $D \leq (1+\varepsilon)\Delta$ within time

$$\leq S \mathrm{poly}(K) (d/\varepsilon)^{O(Td)} n \log\log n.$$

[2] The given notation assumes that the corruption probability $p(i, j)$ is independent of the source and side-information values x and y. While this assumption is likely appropriate for most coding environments, it is in now way critical for optimal code design using the proposed approach.

References

[1] P. K. Agarwal and M. Sharir. Efficient algorithms for geometric optimization. ACM Computing Surveys, 30:412–458, 1998.

[2] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for Euclidean k-medians and related problems. In *Annual ACM Symposium on Theory of Computing*, pages 106–113, Dallas, Texas, 1998.

[3] M. Badoiu, S. Har-Peled, and P. Indyk. Approximate clustering via core-sets. In *Proc. 34th ACM STOC*, pages 250–257, 2002.

[4] M. Blum, R. Floyd, V. Pratt, R. Rivest, and R. Tarjan. Time bounds for selection. *Journal of Computing and System Sciences*, 7:448–461, 1973.

[5] J. D. Bruce. *Optimum Quantization.* PhD thesis, M.I.T., Cambridge, MA, May 1964.

[6] P. A. Chou, T. Lookabaugh, and R. M. Gray. Optimal pruning with applications to tree structured source coding and modeling. *IEEE Transactions on Information Theory*, IT-35(2):299–315, March 1989.

[7] W. Fernandez de la Vega, M. Karpinski, C. Kenyon, and Y. Rabani. Approximation schemes for clustering problems. In *Proc. 35'th Ann. Symp. on Theory of Computing (STOC)*, 2003.

[8] D. Dor and U. Zwick. Selecting the median. *SIAM J. Comput.*, 28(5):1722–1758, 1999.

[9] P. Drineas, A. Frieze, R. Kannan, S. Vempala, and V. Vinay. Clustering in large graphs and matrices. In *Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1999.

[10] M. Effros and L. J. Schulman. Manuscript, 2004.

[11] T. Feder and D. H. Greene. Optimal algorithms for approximate clustering. In *Proc. 20th Ann. ACM Symp. Theory Comput.*, pages 434–444. ACM, 1988.

[12] T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. Theoretical Computer Science, 38:293–306, 1985.

[13] S. Har-Peled. A replacement for Voronoi diagrams of near linear size. In *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*, pages 94–103, Las Vegas, NV, October 2001.

[14] D. S. Hochbaum and D. B. Shmoys. A unified approach to approximation algorithms for bottleneck problems. J. Assoc. Comput. Mach., 33(3):533–550, 1986.

[15] D. S. Hochbaum and D. B. Smoys. A best possible heuristic for the k-center problem. Math. Oper. Res., 10:180–184, 1985.

[16] K. Jain and V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *J. ACM*, 48:274–296, 2001.

[17] S. G. Kolliopoulos and S. Rao. A nearly linear-time approximation scheme for the Euclidean k-median problem. In *Proc. European Symposium on Algorithms*, pages 378–389. Springer-Verlag LNCS, 1999.

[18] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28(1):84–95, January 1980.

[19] T. Linder, G. Lugosi, and K. Zeger. Rates of convergence in the source coding theorem, in empirical quantizer design, and in universal lossy source coding. *IEEE Transactions on Information Theory*, 40:1728–1740, November 1994.

[20] J. Matousek. On approximate geometric k-clustering. *Discrete & Computational Geometry*, 24:61–84, 2000.

[21] D. Muresan and M. Effros. Quantization as histogram segmentation: globally optimal scalar quantizer design in network systems. In *Proceedings of the Data Compression Conference*, pages 302–311, Snowbird, Utah, March 2002.

[22] R. Ostrovsky and Y. Rabani. Polynomial time approximation schemes for geometric clustering problems. *J. ACM*, 49(2):139–156, 2002.

[23] A. Schönhage, M. Paterson, and N. Pippenger. Finding the median. *J. Comput. System Sci*, 13:184–199, 1976.

[24] L. J. Schulman. Clustering for edge-cost minimization. In *Proc. 32'nd Ann. Symp. on Theory of Computing (STOC)*, pages 547–555, 2000.

[25] D. K. Sharma. Design of absolutely optimal quantizers for a wide class of distortion measures. *IEEE Transactions on Information Theory*, IT-24(6):693–702, November 1978.

[26] A. F. van der Stappen, D. Halpern, and M. H. Overmars. The complexity of the free space for a robot moving amidst fat obstacles. *Computational Geometry: Theory and Applications*, 3:353–373, 1993.

[27] X. Wu. *Algorithmic approach to mean-square quantization.* PhD thesis, University of Calgary, 1988.

[28] X. Wu and K. Zhang. Quantizer monotonicities and globally optimal scalar quantizer design. *IEEE Transactions on Information Theory*, IT-39(3):1049–1053, May 1993.

[29] A. D. Wyner and J. Ziv. The rate-distortion function for source coding with side information at the decoder. *IEEE Transactions on Information Theory*, IT-22(1):1–10, January 1976.