Chapter 1

Near Neighbor Search in ℓ_∞

In this chapter, we deal with near neighbor search (NNS) under the distance $d(p,q) = \|p-q\|_\infty = \max_{i\in[d]} |p_i - q_i|$, called the ℓ_∞ norm. See §**??** for background on near-neighbor search, and, in particular, for a discussion of this important metric.

The structure of the ℓ_∞ space is intriguingly different, and more mysterious than other natural spaces, such as the ℓ_1 and ℓ_2 norms. In fact, there is precisely one data structure for NNS under ℓ_∞ with provable guarantees. In FOCS'98, Indyk [Ind01] described an NNS algorithm for d-dimensional ℓ_∞ with approximation $4\lceil\log_\rho \log_2 4d\rceil + 1$, which required space $dn^\rho \lg^{O(1)} n$ and query time $d \cdot \lg^{O(1)} n$, for any $\rho > 1$. For 3-approximation, Indyk also gives a solution with storage $O(n^{\log_2 d+1})$. Note that in the regime of polynomial space, the algorithm achieves an uncommon approximation factor of $O(\log\log d)$.

In this chapter, we begin by describing Indyk's data structure in a manner that is conceptually different from the original description. Our view relies on an information-theoretic understanding of the algorithm, which we feel explains its behavior much more clearly.

Inspired by this understanding, we are able to prove a lower bound for the asymmetric communication complexity of c-approximate NNS in ℓ_∞:

Theorem 1.1. *Assume Alice holds a point $q \in \{0,\dots,m\}^d$, and Bob holds a database $D \subset \{-m,\dots m\}^d$ of n points. They communicate via a deterministic protocol to output:*
"1" *if there exists some $p \in D$ such that $\|q-p\|_\infty \le 1$;*
"0" *if, for all $p \in D$, we have $\|q-p\|_\infty \ge c$.*

Fix $\delta, \varepsilon > 0$; assume the dimension d satisfies $\Omega(\lg^{1+\varepsilon} n) \le d \le o(n)$, and the approximation ratio satisfies $3 < c \le O(\log\log d)$. Further define $\rho = \frac{1}{2}(\frac{\varepsilon}{2}\log d)^{1/c} > 10$.

Then, either Alice sends $\Omega(\delta\rho\log n)$ bits, or Bob sends $\Omega(n^{1-\delta})$ bits.

Note that this result is tight in the communication model, suggesting the Indyk's unusual approximation is in fact inherent to NNS in ℓ_∞. As explained in Chapter **??**, this lower bound on asymmetric communication complexity immediately implies the following corollaries for data structures:

Corollary 1.2. *Let $\delta > 0$ be constant, and assume $\Omega(\lg^{1+\delta} n) \le d \le o(n)$. Consider any cell-probe data structure solving d-dimensional NNS under ℓ_∞ with approximation $c =$*

$O(\log_\rho \log d)$. If the word size is $w = n^{1-\delta}$ and the query complexity is t, the data structure requires space $n^{\Omega(\rho/t)}$.

Corollary 1.3. *Let $\delta > 0$ be constant, and assume $\Omega(\lg^{1+\delta} n) \le d \le o(n)$. A decision tree of depth $n^{1-2\delta}$ with predicate size n^δ that solves d-dimensional near-neighbor search under ℓ_∞ with approximation $c = O(\log_\rho \log d)$, must have size $n^{\Omega(\rho)}$.*

As with all known lower bounds for large space, Corollary 1.2 is primarily interesting for constant query time, and degrades exponentially with t. On the other hand, the lower bound for decision trees holds even for extremely high running time (depth) of $n^{1-\delta}$. A decision tree with depth n and predicate size $O(d \log M)$ is trivial: simply test all database points.

Indyk's result is a deterministic decision tree with depth $d \cdot \log^{O(1)} n$ and predicate size $O(\lg d + \lg M)$. Thus, we show an optimal trade-off between space and approximation, at least in the decision tree model. In particular, for polynomial space, the approximation factor of $\Theta(\lg \lg d)$ is intrinsic to NNS under ℓ_∞.

1.1 Review of Indyk's Upper Bound

Decision trees. Due to the decomposability of ℓ_∞ as a maximum over coordinates, a natural idea is to solve NNS by a decision tree in which every node is a coordinate comparison. A node v is reached for some set $Q_v \subseteq \mathbb{Z}^d$ of queries. If the node compares coordinate $i \in [d]$ with a "separator" x, its two children will be reached for queries in $Q_\ell = Q_v \cap \{q \mid q_i < x\}$, respectively in $Q_r = Q_v \cap \{q \mid q_i > x\}$ (assume x is non-integral to avoid ties).

Define $[x, y]_i = \{p \mid p_i \in [x, y]\}$. Then, $Q_\ell = Q_v \cap [-\infty, x]_i$ and $Q_r = Q_v \cap [x, \infty]_i$.

If the query is known to lie in some Q_v, the set of database points that could still be a near neighbor is $N_v = D \cap \left(Q_v + [-1, 1]^d\right)$, i.e. the points inside the Minkowski sum of the query set with the ℓ_∞ "ball" of radius one. For our example node comparing coordinate $i \in [d]$ with x, the children nodes have $N_\ell = N_v \cap [-\infty, x + 1]_i$, respectively $N_r = N_v \cap [x - 1, +\infty]_i$.

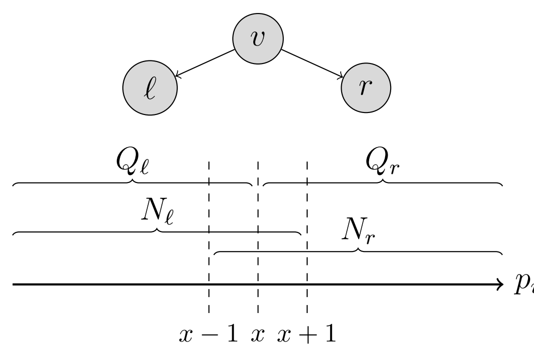

Figure 1-1: A separator x on coordinate i.

Observe that $N_\ell \cap N_r = N_v \cap [x - 1, x + 1]_i$. In some sense, the database points in this slab are being "replicated," since both the left and right subtrees must consider them as potential near neighbors. This recursive replication of database points is the cause of superlinear space. The contribution of Indyk [Ind01] is an intriguing scheme for choosing a separator that guarantees a good bound on this recursive growth.

Information progress. Our first goal is to get a handle on the growth of the decision tree, as database points are replicated recursively. Imagine, for now, that queries come from some distribution μ. The reader who enjoys worst-case algorithms need not worry: μ is just an analysis gimmick, and the algorithm will be deterministic.

We can easily bound the tree size in terms of the measure of the smallest Q_v ever reached: there can be at most $1/\min_v \Pr_\mu[Q_v]$ distinct leaves in the decision tree, since different leaves are reached for disjoint Q_v's. Let $I_Q(v) = \log_2 \frac{1}{\Pr_\mu[Q_v]}$; this can be understood as the information learned about the query, when computation reaches node v. We can now rewrite the space bound as $O\big(2^{\max_v I_Q(v)}\big)$.

Another quantity that can track the behavior of the decision tree is $H_N(v) = \log_2 |N_v|$. Essentially, this is the "entropy" of the identity of the near neighbor, assuming that all database points are equally likely neighbors.

At the root λ, we have $I_Q(\lambda) = 0$ and $H_N(\lambda) = \lg n$. Decision nodes must reduce the entropy of the near neighbor until H_N reaches zero ($|N_v| = 1$). Then, the algorithm can simply read the single remaining candidate, and test whether it is a near neighbor of the query. Unfortunately, decision nodes also increase I_Q along the way, increasing the space bound. The key to the algorithm is to balance this tension between reducing the entropy of the answer, H_D, and not increasing the information about the query, I_Q, too much.

In this information-theoretic view, Indyk's algorithm shows that we can (essentially) always find a separator that decreases H_N by some δ but does not increase I_Q by more than $\rho \cdot \delta$. Thus, H_D can be pushed from $\lg n$ down to 0, without ever increasing I_Q by more than $\rho \lg n$. That is, space $O(n^\rho)$ is achieved.

Searching for separators. At the root λ, we let $i \in [d]$ be an arbitrary coordinate, and search for a good separator x on that coordinate. Let π be the frequency distribution (the empirical probability distribution) of the projection on coordinate i of all points in the database. To simplify expressions, let $\pi(x : y) = \sum_{j=x}^{y} \pi(j)$.

If x is chosen as a separator at the root, the entropy of the near neighbor in the two child nodes is reduced by:

$$\begin{aligned} H_N(\lambda) - H_N(\ell) &= \log_2 \frac{|N_\lambda|}{|N_\ell|} = \log_2 \frac{|D|}{|D \cap [-\infty, x+1]_i|} = \log_2 \frac{1}{\pi(-\infty : x+1)} \\ H_N(\lambda) - H_N(r) &= \log_2 \frac{1}{\pi(x-1 : \infty)} \end{aligned}$$

Remember that we have not yet defined μ, the assumed probability distribution on the query. From the point of view of the root, it only matters what probability μ assigns to Q_ℓ and Q_r. Let us reason, heuristically, about what assignments are needed for these probabilities in order to generate difficult problem instances. If we understand the most difficult instance, we can use that setting of probabilities to obtain an upper bound for all instances.

First, it seems that in a hard instance, the query needs to be close to some database point (at least with decent probability). Let us simply assume that the query is always *planted* in the neighborhood of a database point; the problem remains to find this near neighbor.

Assume by symmetry that $H_N(\ell) \geq H_N(r)$, i.e. the right side is smaller. Under our heuristic assumption that the query is planted next to a random database point, we can lower bound $\Pr_\mu[Q_r] \geq \pi(x+1,\infty)$. Indeed, whenever the query is planted next to a point in $[x+1,\infty]_i$, it cannot escape from $Q_r = [x,\infty]_i$. Remember that our space guarantee blows up when the information about Q_v increases quickly (i.e. the probability of Q_v decreases). Thus, the worst case seems to be when $\Pr_\mu[Q_r]$ is as low as possible, namely equal to the lower bound.

To summarize, we have convinced ourselves that it's reasonable to define μ such that:

$$\Pr_\mu[Q_\ell] = \pi(-\infty : x+1); \qquad \Pr_\mu[Q_r] = \pi(x+1,\infty) \tag{1.1}$$

We apply similar conditions at all nodes of the decision tree. Note that there exists a μ safisfying all these conditions: the space of queries is partitioned recursively between the left and right subtrees, so defining the probability of the left and right subspace at all nodes is an (incomplete) definition of μ.

From (1.1), we can compute the information revealed about the query:

$$\begin{aligned} I_Q(\ell) - I_Q(\lambda) &= \log_2 \frac{\Pr[Q_\lambda]}{\Pr[Q_\ell]} = \log_2 \frac{1}{\pi(-\infty : x+1)} \\ I_Q(r) - I_Q(\lambda) &= \log_2 \frac{1}{\pi(x+1:\infty)} \end{aligned}$$

Remember that our rule for a good separator was "$\Delta I_Q \leq \rho \cdot \Delta H_N$." On the left side, $I_Q(\ell) - I_Q(\lambda) = H_N(\lambda) - H_N(\ell)$, so the rule is trivially satisfied. On the right, the rule asks that: $\log_2 \frac{1}{\pi(x+1:\infty)} \leq \rho \cdot \log_2 \frac{1}{\pi(x-1:\infty)}$. Thus, x is a good separator iff $\pi(x+1:\infty) \geq \big[\pi(x-1:\infty)\big]^\rho$.

Finale. As defined above, any good separator satisfies the bound on the information progress, and guarantees the desired space bound of $O(n^\rho)$. We now ask what happens when no good separator exists.

We may assume by translation that the median of π is 0, so $\pi([1:\infty]) \leq \frac{1}{2}$. If $x = 1\frac{1}{2}$ is not a good separator, it means that $\pi(3:\infty) < \big[\pi(1:\infty)\big]^\rho \leq 2^{-\rho}$. If $x = 3\frac{1}{2}$ is not a good separator, then $\pi(5:\infty) < \big[\pi(3:\infty)\big]^\rho \leq 2^{-\rho^2}$. By induction, the lack of a good separator implies that $\pi(2j+1:\infty) < 2^{-\rho^j}$. The reasoning works simmetrically to negative values, so $\pi(-\infty : -2j-1) < 2^{-\rho^j}$.

Thus, if no good separator exists on coordinate i, the distribution of the values on that coordinate is very concentrated around the median. In particular, only a fraction of $\frac{1}{2d}$ of the database points can have $|x_i| > R = 2\log_\rho \log_2 4d$. Since there is no good separator on any coordinate, it follows that less than $d \cdot \frac{n}{2d} = \frac{n}{2}$ points have *some* coordinate exceeding R. Let $D^\star$ be the set of such database points.

To handle the case when no good separator exists, we can introduce a different type of node in the decision tree. This node tests whether the query lies in an ℓ_∞ ball of radius $R+1$ (which is equivalent to d coordinate comparisons). If it does, the decision tree simply outputs any point in $D \setminus D^\star$. Such a point must be within distance $2R+1$ of the query, so it is an $O(\log_\rho \log d)$ approximation.

If the query is outside the ball of radius $R+1$, a near neighbor must be outside the ball of radius R, i.e. must be in $D^\star$. We continue with the recursive construction of a decision tree for point set $D^\star$. Since $|D^\star| \leq |D|/2$, we get a one-bit reduction in the entropy of the answer for free. (Formally, our μ just assigns probability one to the query being outside the ball of radius $R+1$, because in the "inside" case the query algorithm terminates immediately.)

1.2 Lower Bound

Armed with this information-theoretic understanding of Indyk's algorithm, the path to a lower bound is more intuitive. We will define a distribution on coordinates decaying roughly like $2^{-\rho^x}$, since we know that more probability in the tail gives the algorithm an advantage. Database points will be independent and identically distributed, with each coordinate drawn independently from this distribution.

In the communication view, Alice's message sends a certain amount of information restricting the query space to some Q. The entropy of the answer is given by the measure of $N(Q) = Q + [-1,1]^d$, since the expected number of points in this space is just $n \cdot \Pr[N(Q)]$. The question that must be answered is: fixing $\Pr[Q]$, how small can $\Pr\big[N(Q)\big]$ be?

We will show an isoperimetric inequality proving that the least expanding sets are exactly the ones generated by Indyk's algorithm: intersections of coordinate cuts $[x,\infty]_i$. Note that $\Pr\big[[x,\infty]_i\big] \approx 2^{-\rho^x}$, and $N\big([x,\infty]_i\big) = [x-1,\infty]_i$. Thus, the set expands to measure $\Pr\big[x-1,\infty]_i\big] \approx 2^{-\rho^{x-1}} \approx \Pr\big[[x,\infty]_i\big]^{1/\rho}$. Our isoperimetric inequality will show that for any Q, its neighborhood has measure $\Pr[N(Q)] \geq \Pr[Q]^{1/\rho}$.

Then, if Alice's message has $o(\rho \lg n)$ bits of information, the entropy of the near neighbor decreases by only $o(\lg n)$ bits. In other words, $n^{1-o(1)}$ of the points are still candidate near neighbors, and we can use this to lower bound the message that Bob must send.

The crux of the lower bound is not the analysis of the communication protocol (which is standard), but proving the isoperimetric inequality. Of course, the key to the isopermitetric inequality is the initial conceptual step of defining an appropriate biased distribution, in which the right inequality is possible. The proof is rather non-standard for an isoperimetric inequality, because we are dealing with a very particular measure on a very particular space. Fortunately, a few mathematical tricks save it from being too technical.

Formal details. We denote the communication problem, c-approximate NNS, by the partial function F. Let the domain of $\bar{F}$ be $X \times Y$, where $X = \{0,1,\ldots m\}^d$ and $Y = \big(\{0,1,\ldots m\}^d\big)^n$. Complete the function F by setting $\bar{F}(q,D) = F(q,D)$ whenever $F(q,D)$ is defined (in the "0" or "1" instances), and $\bar{F}(q,D) = \star$ otherwise.

As explained already, our lower bound only applies to deterministic protocols, but it requires conceptual use of distributions on the input domains X and Y. First define a measure π over the set $\{0,1,\ldots m\}$: for $i \geq 1$, let $\pi(\{i\}) = \pi_i = 2^{-(2\rho)^i}$; and let $\pi_0 = 1 - \sum_{i\geq 1} \pi_i \geq \frac{1}{2}$. Here ρ is a parameter to be determined.

Now, define a measure μ over points by generating each coordinate independently according to π: $\mu(x_1, x_2, \dots, x_d)\}) = \pi_{x_1} \cdots \pi_{x_d}$. Finally, define a measure η over the database by generating each point independently according to μ.

First, we show that $\bar{F}$ is zero with probability $\Omega(1)$:

Claim 1.4. *If $d \geq \lg^{1+\varepsilon} n$ and $\rho \leq \frac{1}{2}(\frac{\varepsilon}{2}\lg d)^{1/c}$, then* $\Pr_{q\leftarrow\mu, D\leftarrow\eta}[\bar{F}(q, D) \neq 0] \leq \frac{1}{2}$.

Proof. Consider q and some $p \in D$: their jth coordinates differ by c or more with probability at least $2\pi_0\pi_c \geq \pi_c$. Thus,

$$\Pr[\|q-p\|_\infty < c] \leq (1-\pi_c)^d \leq e^{-\pi_c d} \leq e^{-2^{-(\varepsilon/2)\lg d}\cdot d} \leq e^{-d^{1-\varepsilon/2}} \leq e^{-(\lg n)^{(1+\varepsilon)(1-\varepsilon/2)}} \leq e^{-(\lg n)^{1+\varepsilon/4}}$$

By a union bound over all $p \in D$, we get that q has no neighbor at distance less than c with probability at least $1 - n \cdot \exp(-(\lg n)^{1+\varepsilon/4}) = 1 - o(1)$. □

Claim 1.5. *If Alice sends a bits and Bob sends b bits, there exists a combinatorial rectangle $\mathcal{Q} \times \mathcal{D} \subseteq X \times Y$ of measure $\mu(\mathcal{Q}) \geq 2^{-O(a)}$ and $\eta(\mathcal{D}) \geq 2^{-O(a+b)}$, on which $\bar{F}$ only takes values in $\{0, \star\}$.*

Proof. This is just the deterministic richness lemma (Lemma **??**) in disguise. Let $F' : X \times Y \to \{0, 1\}$ be the output of the protocol. We have $F'(q, D) = \bar{F}(q, D)$ whenever $\bar{F}(q, D) \neq \star$. Since $\Pr[F'(q, D) = 0] \geq \frac{1}{2}$, F' is rich: half of the columns are at least half zero (in the weighted sense). Thus, we can find a rectangle of size $\mu(\mathcal{Q}) \geq 2^{-O(a)}$ and $\eta(\mathcal{D}) \geq 2^{-O(a+b)}$, on which F' is identically zero. Since the protocol is always correct, this means that $\bar{F}$ is 0 or $\star$ on the rectangle. □

To obtain a lower bound, we show that any big rectangle must contain at least a value of "1." This will follow from the following isoperimetric inequality in our measure space, shown in §1.3:

Theorem 1.6. *Consider any set $S \subseteq \{0, 1, \dots m\}^d$, and let $N(S)$ be the set of points at distance at most 1 from S under ℓ_∞: $N(S) = \{p \mid \exists s \in S : \|p - s\|_\infty \leq 1\}$. Then, $\mu(N(S)) \geq \big(\mu(S)\big)^{1/\rho}$.*

Claim 1.7. *Consider any rectangle $\mathcal{Q} \times \mathcal{D} \subseteq X \times Y$ of size $\mu(\mathcal{Q}) \geq 2^{-\delta\rho\lg n}$ and $\eta(\mathcal{D}) \geq 2^{-O(n^{1-\delta})}$. Then, there exists some $(q, D) \in \mathcal{Q} \times \mathcal{D}$ such that $\bar{F}(q, D) = 1$.*

Proof. By isoperimetry, $\mu(N(\mathcal{Q})) \geq \big(\mu_d(\mathcal{Q})\big)^{1/\rho} \geq 1/n^\delta$. All we need to show is that there exists a set $D \in \mathcal{D}$ that intersects with $N(\mathcal{Q})$.

For $D \in Y$, let $\sigma(D) = |D \cap N(\mathcal{Q})|$. The proof uses a standard concentration trick on σ. Suppose D is chosen randomly according to η, i.e. not restricted to $\mathcal{D}$. Then $\mathbf{E}[\sigma(D)] = n \cdot \Pr_\mu[N(\mathcal{Q})] \geq n^{1-\delta}$. Furthermore, $\sigma(D)$ is tightly concentrated around this mean, by the Chernoff bound. In particular, $\Pr[\sigma(D) = 0] \leq e^{-\Omega(n^{1-\delta})}$.

This probability is so small, that it remains small even is we restrict to $\mathcal{D}$. We have $\Pr[\sigma(D) = 0 \mid D \in \mathcal{D}] \leq \frac{\Pr[\sigma(D)=0]}{\Pr[D\in\mathcal{D}]} \leq e^{-\Omega(n^{1-\delta})}/\eta(\mathcal{D})$. Thus, if $\eta(\mathcal{D}) \geq 2^{-\gamma\cdot n^{1-\delta}}$ for some small enough constant γ, we have $\Pr[\sigma(D) = 0 \mid D \in \mathcal{D}] = o(1)$. In other words, there exists some $D \in \mathcal{D}$ such that $N(\mathcal{Q}) \cap D \neq \emptyset$, and thus, there exists an instance in the rectangle on which $\bar{F} = 1$. □

Combining Claims 1.5 and 1.7, we immediately conclude that either Alice sends $a = \Omega(\delta\rho \log n)$ bits or Bob sends $b = \Omega(n^{1-\delta})$ bits. This concludes the proof of Theorem 1.1.

1.3 An Isoperimetric Inequality

This section proves the inequality of Theorem 1.6: for any S, $\mu(N(S)) \geq \big(\mu(S)\big)^{1/\rho}$. As with most isoperimetric inequalities, the proof is by induction on the dimensions. In our case, the inductive step is provided by the following inequality, whose proof is deferred to §1.4:

Lemma 1.8. *Let $\rho \geq 10$ be an integer, and define $\pi_i = 2^{-(2\rho)^i}$ for all $i \in \{1, \dots, m\}$, and $\pi_0 = 1 - \sum_{i=1}^m \pi_i$. For any $\beta_0, \dots, \beta_m \in \mathbb{R}_+$ satisfying $\sum_{i=0}^m \pi_i \beta_i^\rho = 1$, the following inequality holds (where we interpret β_{-1} and β_{m+1} as zero):*

$$\sum_{i=0}^{m} \pi_i \cdot \max\{\beta_{i-1},\ \beta_i,\ \beta_{i+1}\} \ \geq\ 1 \tag{1.2}$$

To proceed to our inductive proof, let μ_d be the d-dimensional variant of our distribution. The base case is $d = 0$. This space has exactly one point, and $\mu_0(S)$ is either 0 or 1. We have $N(S) = S$, so $\mu_0(N(S)) = \mu_0(S) = \big(\mu_0(S)\big)^{1/\rho}$.

Now consider the induction step for $d-1$ to d dimensions. Given a set $S \subset \{0, 1, \dots m\}^d$, let $S_{[i]}$ be the set of points in S whose first coordinate is i, i.e. $S_{[i]} = \{(s_2, \dots, s_d) \mid (i, s_2, \dots, s_d) \in S\}$. Define:

$$\beta_i = \left(\frac{\mu_{d-1}(S_{[i]})}{\mu_d(S)}\right)^{1/\rho} \quad \Rightarrow \quad \sum_{i=0}^{m} \pi_i \beta_i^\rho \ = \ \sum_{i=0}^{m} \pi_i \cdot \frac{\mu_{d-1}(S_{[i]})}{\mu_d(S)} \ = \ \frac{1}{\mu_d(S)} \sum_{i=0}^{m} \pi_i \mu_{d-1}(S_{[i]}) \ = \ 1$$

We have $N(S)_{[i]} = N(S_{[i-1]}) \ \cup \ N(S_{[i]}) \ \cup \ N(S_{[i+1]})$. Thus, we can lower bound:

$$\mu_d(N(S)) \ = \ \sum_{i=0}^{m} \pi_i \cdot \mu_{d-1}\big(N(S)_{[i]}\big) \ \geq \ \sum_{i=0}^{m} \pi_i \cdot \max\big\{\mu_{d-1}(N(S_{[i-1]})), \mu_{d-1}(N(S_{[i]})), \mu_{d-1}(N(S_{[i+1]}))\big\}$$

But the inductive hypothesis assures us that $\mu_{d-1}(N(S_{[i]})) \ \geq \ \big(\mu_{d-1}(S_{[i]})\big)^{1/\rho} \ = \ \beta_i \cdot \big(\mu_d(S)\big)^{1/\rho}$. Thus:

$$\mu_d(N(S)) \ \geq \ \big(\mu_d(S)\big)^{1/\rho} \cdot \sum_{i=0}^{m} \pi_i \cdot \max\big\{\beta_{i-1}, \beta_i, \beta_{i+1}\big\} \ \geq \ \big(\mu_d(S)\big)^{1/\rho},$$

where we have used inequality (1.2) in the last step. This finishes the proof of Theorem 1.6.

1.4 Expansion in One Dimension

Let $\Gamma = \left\{(\beta_0,\ldots,\beta_m) \in \mathbb{R}^{m+1} \mid \sum_{i=0}^{m} \pi_i \beta_i^{\rho} = 1\right\}$, and denote by $f(\beta_0,\ldots,\beta_m)$ the left hand side of (1.2). Since f is a continuous function on the compact set $\Gamma \subset \mathbb{R}^{m+1}$, it achieves its minimum. Call an $(m+1)$-tuple $(\beta_0,\ldots,\beta_m) \in \Gamma$ *optimal* if $f(\beta_0,\ldots,\beta_m)$ is minimal. Our proof strategy will be to show that if $(\beta_0,\ldots,\beta_m)$ is optimal, then $\beta_i = 1$.

We consider several possible configurations for sizes of β_i's in an optimal β, and rule them out in three separate lemmas. We then prove the inequality by showing that these configurations are all the configurations that we need to consider.

Lemma 1.9. *If there exists an index $i \in \{1,\ldots,m-1\}$ such that $\beta_{i-1} > \beta_i < \beta_{i+1}$, then $\bar{\beta} = (\beta_0,\ldots,\beta_m)$ is not optimal.*

Proof. Define a new vector $\bar{\beta}' = (\beta_0,\ldots,\beta_{i-2},\beta_{i-1}-\epsilon,\beta_i+\delta,\beta_{i+1}-\epsilon,\beta_{i+2},\ldots,\beta_m)$, where $\epsilon,\ \delta > 0$ are chosen suitably so that $\bar{\beta}' \in \Gamma$, and $\beta_{i-1}-\epsilon > \beta_i+\delta < \beta_{i+1}-\epsilon$. It's easy to see that $f\left(\bar{\beta}\right) > f\left(\bar{\beta}'\right)$, which contradicts the optimality of $\bar{\beta}$. □

Lemma 1.10. *If there exists an index $i \in \{1,\ldots,m\}$ such that $\beta_{i-1} > \beta_i \geq \beta_{i+1}$, then $\bar{\beta} = (\beta_0,\ldots,\beta_m)$ is not optimal.*

Proof. Let $\beta = \left(\frac{\pi_{i-1}\beta_{i-1}^{\rho}+\pi_i\beta_i^{\rho}}{\pi_{i-1}+\pi_i}\right)^{1/\rho}$ and define $\bar{\beta}' = (\beta_0,\ldots,\beta_{i-2},\beta,\beta,\beta_{i+1},\ldots\beta_m)$. Then $\bar{\beta}' \in \Gamma$, and $\beta_{i-1} > \beta > \beta_i$.

We claim that $f(\bar{\beta}) > f(\bar{\beta}')$. Comparing the expressions for $f\left(\bar{\beta}\right)$ and $f\left(\bar{\beta}'\right)$ term by term, we see that it's enough to check that:

$$\pi_i \max\left\{\beta_{i-1},\beta_i,\beta_{i+1}\right\} + \pi_{i+1}\max\left\{\beta_i,\beta_{i+1},\beta_{i+2}\right\} \;>\; \pi_i\max\left\{\beta,\beta_{i+1}\right\} + \pi_{i+1}\max\left\{\beta,\beta_{i+1},\beta_{i+2}\right\}$$

where the terms involving π_{i+1} are ignored when $i = m$. For $i = m$, the inequality becomes $\beta_{i-1} > \beta$ which holds by assumption. For $i = 1,\ldots,m-1$, this inequality is equivalent to:

$$\pi_i(\beta_{i-1}-\beta) > \pi_{i+1}\cdot(\max\{\beta,\beta_{i+2}\} - \max\{\beta_i,\beta_{i+2}\})$$

which, in its strongest form (when $\beta_i \geq \beta_{i+2}$), is equivalent to $\pi_i(\beta_{i-1}-\beta) > \pi_{i+1}(\beta-\beta_i)$. But this is equivalent to:

$$\left(\frac{\pi_i\beta_{i-1}+\pi_{i+1}\beta_i}{\pi_i+\pi_{i+1}}\right)^{\rho} > \frac{\pi_{i-1}\beta_{i-1}^{\rho}+\pi_i\beta_i^{\rho}}{\pi_{i-1}+\pi_i}$$

which we can rewrite as:

$$\left(\frac{c_i+t}{c_i+1}\right)^{\rho} - \frac{c_{i-1}+t^{\rho}}{c_{i-1}+1} > 0, \tag{1.3}$$

letting $t = \frac{\beta_i}{\beta_{i-1}} \in [0,1)$, and $c_i = \frac{\pi_i}{\pi_{i+1}} \geq 2^{(2\rho)^{i+1}-(2\rho)^i}$ (for $i > 0$ this is an equality; only for $i = 0$ is this a strict inequality, because p is large).

We are now left to prove (1.3). Let $F(t)$ denote the left hand side of this inequality, and note that $F(0) > 0$, because:

$$\left(\frac{c_i}{c_i+1}\right)^\rho \;=\; \left(1-\frac{1}{c_i+1}\right)^\rho \;\geq\; 1-\frac{\rho}{c_i+1} \;>\; 1-\frac{1}{c_{i-1}+1} \;=\; \frac{c_{i-1}}{c_{i-1}+1}$$

Here we used Bernoulli's inequality: $(1-x)^n \geq 1-nx$ for $0 < x < 1/n$. Then, we observed that $c_i + 1 > 2^{(2\rho)^{i+1}-(2\rho)^i} > \rho \cdot (2^{(2\rho)^i}+1) = \rho(\frac{1}{\pi_{i-1}}c_{i-1}+1) > \rho(c_{i-1}+1)$.

Now we let $t \in (0,1)$ and write $F(t) = F(0) + t^\rho G(t)$, where:

$$G(t) \;=\; \tfrac{1}{(c_i+1)^\rho}\left(\tbinom{\rho}{1}c_i^{\rho-1}\tfrac{1}{t} + \tbinom{\rho}{2}c_i^{\rho-2}\tfrac{1}{t^2} + \cdots + \tbinom{\rho}{\rho-1}c_i\tfrac{1}{t^{\rho-1}}\right) + \left(\tfrac{1}{(c_i+1)^\rho} - \tfrac{1}{c_{i-1}+1}\right).$$

If $G(t) \geq 0$, then clearly $F(t) \geq F(0) > 0$, so we are done. Otherwise, $G(t) < 0$, and in this case it easily follows that $G(1) < G(t) < 0$, hence $F(t) = F(0)+t^\rho G(t) > F(0)+G(1) = F(1) = 0$, as desired. This concludes the proof of the lemma. □

Lemma 1.11. *If there is an index $i \in \{0, 1\ldots, m-1\}$ such that $\beta_{i-1} \leq \beta_i < \beta_{i+1}$, then $\beta = (\beta_0, \beta_1, \ldots, \beta_m)$ is not optimal.*

Proof. We proceed as in the previous lemma. Let $\beta = \left(\frac{\pi_i\beta_i^\rho+\pi_{i+1}\beta_{i+1}^\rho}{\pi_i+\pi_{i+1}}\right)^{1/\rho}$, and define $\bar\beta' = (\beta_0, \ldots, \beta_{i-1}, \beta, \beta, \beta_{i+2}, \ldots, \beta_m)$. As before, $\bar\beta' \in \Gamma$ and $\beta_i < \beta < \beta_{i+1}$. We claim that $f(\bar\beta) > f(\bar\beta')$. Comparing the expressions for $f\left(\bar\beta\right)$ and $f\left(\bar\beta'\right)$ term by term, we see that it's enough to check that

$$\pi_{i-1}\cdot\max\{\beta_{i-2}, \beta_{i-1}, \beta_i\}+\pi_i\cdot\max\{\beta_{i-1}, \beta_i, \beta_{i+1}\} \;>\; \pi_{i-1}\cdot\max\{\beta_{i-2}, \beta_{i-1}, \beta\}+\pi_i\cdot\max\{\beta_{i-1}, \beta, \beta\}$$

where the terms involving π_{i-1} appear unless $i = 0$. If $i = 0$, the above inequality becomes $\beta_{i+1} > \beta$ and we are done. For $i = 1, \ldots m-1$, the inequality is equivalent to

$$\pi_i(\beta_{i+1} - \beta) > \pi_{i-1} \cdot (\max\{\beta, \beta_{i-2}\} - \max\{\beta_i, \beta_{i-2}\})$$

which, in its strongest form (when $\beta_i \geq \beta_{i-2}$) is equivalent to $\pi_i(\beta_{i+1} - \beta) > \pi_{i-1}(\beta - \beta_i)$. The latter inequality is equivalent to

$$\left(\frac{\pi_i\beta_{i+1} + \pi_{i-1}\beta_i}{\pi_i + \pi_{i-1}}\right)^\rho > \frac{\pi_{i+1}\beta_{i+1}^\rho + \pi_i\beta_i^\rho}{\pi_{i+1} + \pi_i}$$

which we can rewrite as

$$\left(\frac{c_{i-1}t + 1}{c_{i-1} + 1}\right)^\rho - \frac{c_i t^\rho + 1}{c_i + 1} > 0, \tag{1.4}$$

where $c_i = \pi_i/\pi_{i+1}$ as before, and $t = \beta_i/\beta_{i+1} \in [0, 1)$.

We are left to prove (1.4). Let $F(t)$ denote the left hand side of this inequality, and note that $F(0) > 0$, because:

$$\left(\frac{1}{c_{i-1}+1}\right)^{\rho} > \frac{1}{(2c_{i-1})^{\rho}} = \frac{1}{\pi_{i-1}^{\rho}} \cdot 2^{-\rho\cdot(2\rho)^i-\rho} > 2^{-\rho\cdot(2\rho)^i-\rho} \geq 2^{(2\rho)^i-(2\rho)^{i+1}} = \frac{1}{c_i} > \frac{1}{c_i+1}$$

Now we let $t \in (0,1)$ and write $F(t) = F(0) + t^{\rho}G(t)$, where

$$G(t) = \tfrac{1}{(c_{i-1}+1)^{\rho}}\left(\tbinom{\rho}{1}c_{i-1}\tfrac{1}{t} + \tbinom{\rho}{2}c_{i-1}^2\tfrac{1}{t^2} + \cdots + \tbinom{\rho}{\rho-1}c_{i-1}^{\rho-1}\tfrac{1}{t^{\rho-1}}\right) + \left(\left(\tfrac{c_{i-1}}{c_{i-1}+1}\right)^{\rho} - \tfrac{c_i}{c_{i-1}+1}\right).$$

If $G(t) \geq 0$, then clearly $F(t) \geq F(0) > 0$, so we are done. Otherwise, $G(t) < 0$, in which case it easily follows that $G(1) < G(t) < 0$, hence $F(t) = F(0)+t^{\rho}G(t) > F(0)+G(1) = F(1) = 0$, as desired. This concludes the proof of the lemma. □

To prove Lemma 1.8, assume $\bar{\beta} = (\beta_0, \ldots, \beta_m) \in \Gamma$ is optimal. By Lemmas 1.9 and 1.10, it follows that $\beta_0 \leq \beta_1 \leq \cdots \leq \beta_m$. Now Lemma 1.11 implies that $\beta_0 = \beta_1 = \cdots = \beta_m$. Since $\bar{\beta} \in \Gamma$, we have $\beta_i = 1$, and hence the minimal value of f over Γ is $f(1,1,\ldots,1) = 1$.

This concludes the proof of Lemma 1.8.