Embedding Dimensions of Box-Convex Codes

Kevin Zhou

Advisor: Amzi Jeffs

Abstract

We study box-convex codes, which are combinatorial codes that record how a collection of axis-parallel boxes intersect and cover one another in Euclidean space. We provide an explicit algorithm for enumerating all possible box-convex codes on any number of indices, which we use to classify codes on 3 and 4 indices. Furthermore, we prove that box-convex codes can in general have embedding dimension at least $\binom{n}{\lfloor n/2 \rfloor}/2$. We also apply our methods to inductively pierced codes, proving that their realizations are "inflatable", and proving that they are box-convex.

1 Introduction

The motivation for studying convex codes arose from modeling place cells [5]. Suppose I am in $\mathbb{R}^2$ and have 4 neurons that help my brain determine my position. Neuron i fires within a convex set U_i. An example of a family of sets $\mathcal{U}$ in $\mathbb{R}^2$ is shown below.

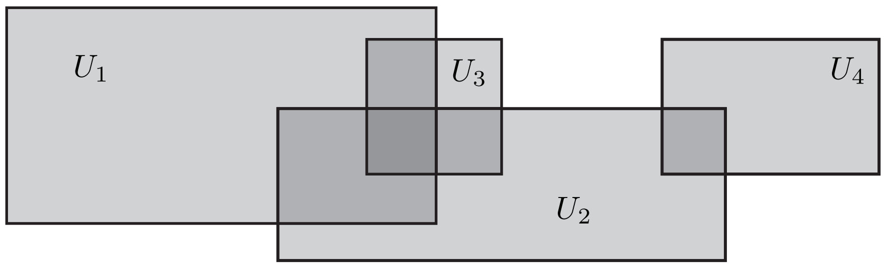


Figure 1: A family of sets U_1, U_2, U_3, U_4 in $\mathbb{R}^2$

We can write a combinatorial code that shows how the sets in $\mathcal{U}$ intersect and overlap one another. For a family of n sets $\mathcal{U}$ with each set living in $\mathbb{R}^d$, we define

$$\operatorname{code}(\mathcal{U}) \stackrel{\text{def}}{=} \{\sigma \subseteq [n] : \text{ There exists } p \in \mathbb{R}^d \text{ where } p \in U_i \iff i \in \sigma\}$$

For example, in the figure above, $\operatorname{code}(\mathcal{U}) = \{123, 24, 12, 13, 23, 1, 2, 3, 4, \emptyset\}$. A difficult question posed by Curto, Itskov, Veliz-Cuba, and Youngs [5] is, given a code $\mathcal{C}$, whether there exists a family of, say, open convex sets $\mathcal{U}$ in some dimension d such that $\operatorname{code}(\mathcal{U}) = \mathcal{C}$. Such a family of sets $\mathcal{U}$ is known as a *realization* of $\mathcal{C}$.

To shed some light on this question, we will restrict to determining whether for a given $\mathcal{C}$, there exists a family of open *boxes* $\mathcal{U}$ in some dimension d such that $\operatorname{code}(\mathcal{U}) = \mathcal{C}$. This problem has a much more combinatorial flavor, and we will see it is rich with interesting structure.

1.1 Definitions for boxes

Definition 1 (Box)**.** *An* open box *in $\mathbb{R}^d$ is the Cartesian product of d open intervals $(a_1, b_1) \times \ldots \times (a_d, b_d)$. Similarly a* closed box *is the Cartesian product of closed intervals. Note that boxes must be axis-parallel; they cannot be slanted.*

Definition 2 (Box embedding dimension)**.** *The* box embedding dimension *of a code $\mathcal{C}$ is the minimum dimension d such that $\mathcal{C}$ has a realization by boxes in $\mathbb{R}^d$. We will abbreviate this as* $\operatorname{bdim}(\mathcal{C})$.

We can similarly define the open embedding dimension $\operatorname{odim}(\mathcal{C})$ and the closed embedding dimension $\operatorname{cdim}(\mathcal{C})$ as the minimum dimension d such that $\mathcal{C}$ has a realization by *open convex sets* (respectively, closed convex sets) in $\mathbb{R}^d$.

If a code $\mathcal{C}$ has a realization by boxes, then we say $\mathcal{C}$ is *box-convex*. Some codes don't have a realization in any dimension; in this case we say the embedding dimension is ∞.

Since our codes are defined on the index set $[n]$, we refer to elements of $[n]$ as indices. Other papers may refer to indices as "neurons" instead.

1.2 The product theorem

We now introduce a very important tool for studying box-convex codes, which was first proved in collaboration with Benitez, Chen, Han, Jeffs, Paguyo during a summer 2022 SEMS project at CMU [1].

Definition 3 (Intersection product)**.** *The* intersection product *of two codes $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[n]}$ is the code*

$$\mathcal{C} \boxtimes \mathcal{D} \stackrel{\text{def}}{=} \{c_1 \cap c_2 \mid c_1 \in \mathcal{C} \text{ and } c_2 \in \mathcal{D}\}.$$

Theorem 1 (Product theorem)**.** *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ and $\mathcal{V} = \{V_1, \ldots, V_n\}$ be collections of sets in $\mathbb{R}^{d_1}$ and $\mathbb{R}^{d_2}$ respectively. Define $\mathcal{W} \stackrel{\text{def}}{=} \{U_1 \times V_1, U_2 \times V_2, \ldots, U_n \times V_n\}$. Then*

$$\operatorname{code}(\mathcal{W}) = \operatorname{code}(\mathcal{U}) \boxtimes \operatorname{code}(\mathcal{V}).$$

For example, let $\mathcal{C}_1 = \{123, 12, 124, \emptyset\}$ and $\mathcal{C}_2 = \{134, 1234, \emptyset\}$. Then $\mathcal{C}_1 \boxtimes \mathcal{C}_2 = \{123, 124, 12, 13, 14, 1, \emptyset\}$; let this code be $\mathcal{C}$. We can visualize this as follows, with a realization of $\mathcal{C}_1$ along the horizontal axis and a realization of $\mathcal{C}_2$ along the vertical axis. Then the Cartesian products form a realization of $\mathcal{C}$.

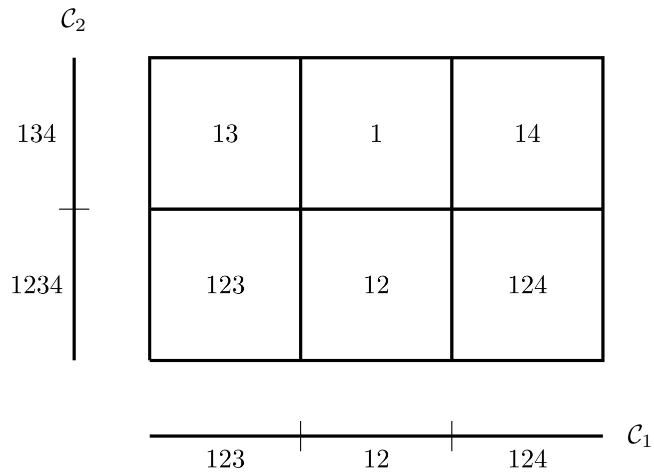


Figure 2: Realizations of $\mathcal{C}_1$ and $\mathcal{C}_2$ along the horizontal and vertical axes, and a realization of $\mathcal{C}_1 \boxtimes \mathcal{C}_2$ in the plane. For instance, the top left region corresponds to the codeword 13, and indeed $134 \cap 123 = 13$.

1.3 Codes in 1 dimension

Codes in 1 dimension are important in our study of box-convex codes because every box is a Cartesian product of 1-dimensional intervals. We have the following theorems for 1-dimensional codes:

Theorem 2 (Integer coordinates [1])**.** *If a code $\mathcal{C}$ on n indices is realizable by closed intervals in $\mathbb{R}$, then there exists a realization $\mathcal{U} = \{U_1, \ldots, U_n\}$ of $\mathcal{C}$ in $\mathbb{R}$ such that each U_i is a closed interval with integer endpoints between 1 and $2n$.*

Theorem 3 (Open, closed doesn't matter in 1 dimension [1])**.** *A code $\mathcal{C}$ is realizable in $\mathbb{R}$ by closed intervals iff $\mathcal{C}$ is realizable in $\mathbb{R}$ by open intervals.*

We can extend Theorem 2 to realizing codes with boxes in $\mathbb{R}^d$ for any positive integer d. Here's a quick sketch of the proof: Suppose a code $\mathcal{C}$ on n indices is realizable by a family of closed boxes $\mathcal{U}$ in $\mathbb{R}^d$. Then we can decompose $\mathcal{U}$ into d families of closed boxes, $\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_d$, each in $\mathbb{R}$. Then, using Theorem 2, for each $\mathcal{U}_i$ we can create a realization $\mathcal{V}_i$ with integer coordinates between 1 and $2n$. Finally we note that $\mathcal{V}_1 \times \mathcal{V}_2 \times \ldots \times \mathcal{V}_d$ is a realization of $\mathcal{C}$, where each coordinate of each box is between 1 and $2n$.

Similarly, Theorem 3 tells us that codes realizable by open boxes in $\mathbb{R}^d$ are also realizable by closed boxes in $\mathbb{R}^d$, and vice versa.

We will frequently use the following lemma from [1] to show that a code is not realizable in $\mathbb{R}$.

Lemma 4. *Let $\mathcal{C}$ be a code with codewords $\sigma, \tau_1, \tau_2, \tau_3$ such that $\tau_i \subset \sigma$ and $\tau_i \not\subset \tau_j$ for all $i \neq j$. Then $\mathcal{C}$ is not realizable in $\mathbb{R}$.*

For example, the code $\{123, 12, 13, 23, 1, 2, 3, \emptyset\}$ is not realizable in $\mathbb{R}$ because the codewords $12, 13, 23$ do not contain each other, and are all subsets of 123.

As a quick sketch of why the lemma is true in general, note a realization of $\mathcal{C}$ in $\mathbb{R}$ would have σ in the middle, and two of the codewords τ_1, τ_2, τ_3 both to the left (or both to the right) of σ. But then the codeword farther away from σ must be a subset of the codeword closer to σ.

There is also the following theorem proven by Rosen and Zhang, which only works in 1 dimension and uses complicated techniques involving consecutive-ones matrices and PQ-trees:

Theorem 5 (Algorithm for 1D codes [9])**.** *There exists an algorithm which, given a code $\mathcal{C}$ on n indices, outputs whether $\mathcal{C}$ is realizable in $\mathbb{R}$ in $O(n \log n)$ time.*

1.4 Restrictions on box-convex codes

There are many codes which we can immediately rule out as not convex (and thus not box-convex), which will help us reduce casework when analyzing codes on small numbers of indices.

Curto et al. [3] classified *mandatory codewords*, which must be present in any convex code. To use mandatory codewords in our study of codes on small numbers of indices, we first define maximal codewords, the simplicial complex, and the link of a code. These definitions make sense by considering codewords as elements in the poset $(\{0,1\}^n, \subseteq)$.

Definition 4 (Maximal codeword)**.** *σ is a maximal codeword of $\mathcal{C}$ if $\sigma \not\subseteq \tau$ for any other codeword $\tau \in \mathcal{C}$.*

Definition 5 (Simplicial complex)**.** *The simplicial complex $\Delta(\mathcal{C})$ is $\{\tau : \tau \subseteq \sigma$ for some $\sigma \in \mathcal{C}\}$.*

For example, we can bold the maximal codewords of the code $\mathcal{C}$ from Section 1.2 to get $\{\mathbf{123}, \mathbf{124}, 12, 13, 14, 1, \emptyset\}$. The simplicial complex needs to include all codewords that are a subset of **123** and **124**, so $\Delta(\mathcal{C}) = \{\mathbf{123}, \mathbf{124}, 12, 13, 14, 23, 24, 1, 2, 3, 4, \emptyset\}$.

Definition 6 (Link)**.** *Let Δ be a simplicial complex, and $\sigma \in \Delta$. The link of σ inside Δ is the set of codewords*

$$Lk_\sigma(\Delta) \stackrel{\text{def}}{=} \{\tau : \tau \cup \sigma \in \Delta \text{ and } \tau \cap \sigma = \emptyset\}$$

For example, let $\mathcal{C} = \{12, 13, \emptyset\}$. Then $\Delta(\mathcal{C}) = \{12, 13, 1, 2, 3, \emptyset\}$, and $\text{Lk}_1(\Delta(\mathcal{C})) = \{2, 3\}$.

Definition 7 (Mandatory codewords)**.** *Let $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$. If $Lk_\sigma(\Delta(\mathcal{C}))$ is not contractible, then $\mathcal{C}$ is not an open convex code (nor a closed convex code). We call σ a mandatory codeword.*

Continuing the example $\mathcal{C} = \{12, 13, \emptyset\}$, note that $\text{Lk}_1(\Delta(\mathcal{C})) = \{2, 3\}$ is not contractible to a point, as $\{2, 3\}$ is two separate, disconnected points. Therefore, to turn $\mathcal{C}$ into a convex code, we need to add the mandatory codeword 1.

There is also a theorem which only applies in the box setting. This theorem rules out scenarios such as $\mathcal{C} = \{12, 13, 23, 1, 2, 3, \emptyset\}$, where the codewords $12, 13, 23$ all have nonempty pairwise intersection, but 123 is not in $\mathcal{C}$.

Theorem 6. *[6] Let $B_1, B_2, \cdots, B_n$ be boxes such that every pair of boxes has nonempty intersection. Then $\bigcap_{i=1}^n B_i$ is nonempty.*

2 Box-convex codes on 4 indices and computational results

The appendix of [1] gives a complete treatment on box-convex codes on 3 indices. In general, we can manually determine whether a specific code $\mathcal{C}$ on n indices is box-convex with the following technique:

- First, determine whether the code is realizable in $\mathbb{R}$ (if so, we are done). This is possible in $O(n \log n)$ time by [9]. An alternative is to generate all possible sets of n intervals to determine all possible codes on n indices in $\mathbb{R}$, and check whether $\mathcal{C}$ is among these codes.
- If $\mathcal{C}$ is not realizable in $\mathbb{R}$, then we attempt to write it as $\mathcal{C}_1 \boxtimes \mathcal{C}_2$. When $\mathcal{C}$ is indeed box-convex, it may take trial and error to find a valid $\mathcal{C}_1, \mathcal{C}_2$. When $\mathcal{C}$ is not box-convex, sometimes casework is necessary to prove $\mathcal{C}$ is not box-convex.

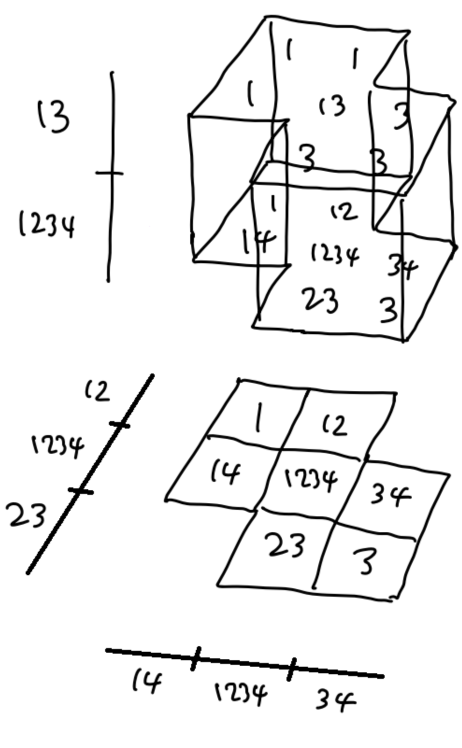
Example of a box-convex code on 4 neurons: Consider the code $\{1234, 12, 13, 14, 23, 34, 1, 3, \emptyset\}$. We can write this as

$$\begin{aligned} &\{1234, 12, 23, 34, 14, 1, 3, \emptyset\} \\ \boxtimes &\{1234, 13, \emptyset\} \end{aligned}$$

Then we can further decompose this into three codes, which we'll call "factors":

$$\begin{aligned} &\{12, 1234, 23, \emptyset\} \\ \boxtimes &\{14, 1234, 34, \emptyset\} \\ \boxtimes &\{1234, 13, \emptyset\} \end{aligned}$$

Each factor is realizable in $\mathbb{R}$. We put one realization on each of the coordinate axes, and take Cartesian products to get a realization of the original code in $\mathbb{R}^3$:



Example of a non-box-convex code on 4 neurons: Consider the code $\mathcal{C} = \{1234, 12, 23, 34, \emptyset\}$. $\mathcal{C}$ is not realizable in $\mathbb{R}$ because of Lemma 4. So, we attempt to write this code as $\mathcal{C}_1 \boxtimes \mathcal{C}_2$, and note that the maximal codeword 1234 must be in both:

$$\begin{array}{r} \{1234, \ldots\} \\ \boxtimes \{1234, \ldots\} \end{array}$$

We note that extra codewords $\sigma \subset 1234$ may not be in either code, because then $\sigma \cap 1234 = \sigma$ would appear in the intersection product.

Without loss of generality we place 12 in the first code.

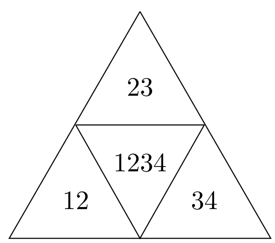
$$\begin{array}{l} \;\;\,\{1234, 12, \ldots\} \\ \boxtimes \{1234, \ldots\} \end{array}$$

23 must be placed in the first code also, since otherwise $12 \cap 23 = 2$ which does not exist in $\mathcal{C}$. Similarly, 34 must be placed in the first code since $23 \cap 34 = 3$ which does not exist in $\mathcal{C}$.

$$\begin{array}{l} \;\;\,\{1234, 12, 23, 34, \ldots\} \\ \boxtimes \{1234, \ldots\} \end{array}$$

Putting $\emptyset$ into the first code (as we assume the realizations are bounded), we have that $\mathcal{C} = \mathcal{C}_1$. This is problematic: if $\mathcal{C}$ is realizable by boxes (that is, $\mathrm{bdim}(\mathcal{C})$ is finite), then $\mathrm{bdim}(\mathcal{C}_1) < \mathrm{bdim}(\mathcal{C})$ because we decomposed $\mathcal{C} = \mathcal{C}_1 \boxtimes \mathcal{C}_2$. However, $\mathrm{bdim}(\mathcal{C}_1) = \mathrm{bdim}(\mathcal{C})$ because $\mathcal{C} = \mathcal{C}_1$.

Interestingly, $\mathcal{C}$ is realizable by convex sets in $\mathbb{R}^2$ (in fact, any code with exactly 1 maximal codeword is realizable by convex sets in $\mathbb{R}^2$ [3]), as shown below:



In manually determining whether a code $\mathcal{C}$ is box-convex, a classification of subcodes helps greatly, in a way similar to dynamic programming. For example, consider the code $\{1234, 12, 23, 34, 1, \emptyset\}$. If we wrote this as $\mathcal{C}_1 \boxtimes \mathcal{C}_2$, the exact same analysis as above tells us that $\mathcal{C}_1 = \{1234, 12, 23, 34, \ldots\}$. Now, using our newly gained knowledge that $\{1234, 12, 23, 34, \emptyset\}$ is not box-convex, this forces $\mathcal{C}_1 = \{1234, 12, 23, 34, 1, \emptyset\}$, a contradiction.

2.1 Algorithm details

To study box-convex codes on 4 indices, I both studied codes manually as above, and used a computer program to generate all possible box-convex codes. The brute-force algorithm generates all possible sets of intervals, and finds the associated codes. Here are details about the algorithm, which we will break down into several different functions, as well as analysis of its running time.

1. `unique` function: Given a list of length m, this takes in a comparison function and does $O(m \log m)$ comparisons to sort the elements and remove duplicates.

2. Lexicographic codeword comparison: This takes in two codewords and lexicographically compares them. We prioritize codewords by size (largest size first), and then by lower-numbered indices. The running time is $O(n)$ where n is the number of indices.

3. Lexicographic code comparison: This takes in two codes, and prioritizes codes with the most codewords, and then the lexicographically-earlier codewords. The running time is $O(n \cdot k)$ where n is the number of indices, and k is the number of codewords. k can be as large as 2^n although it is usually smaller.

4. Checking whether two codes are isomorphic: For every permutation of $[n]$, replace all the codewords under that permutation: $O(n)$ for replacing a codeword, k codewords, $k \log k$ to sort the codewords after replacing, $O(n \cdot k)$ for comparing two codes, $n!$ permutations. All in all this process is in $O(n^2 \cdot n! \cdot k)$.

   Given a set of c codes, removing all isomorphic codes naively (current implementation) would be in $O(c^2 \cdot n^2 \cdot n! \cdot k)$. A far more efficient implementation would be to `collect` the elements into groups (e.g. same number of codewords, codewords have the same lengths), and then remove isomorphisms within those groups.

5. Generating all possible sets of n intervals: The coordinates can range from 0 to $2n$. However, I can cut corners and let the coordinates from 0 to $2n-1$ (which may remove the empty set from some codes, but we add it in later). The number of possible intervals is $\frac{2n(2n-1)}{2}$, which is roughly $2n^2$. Thus the number of sets of intervals is roughly $(2n^2)^n$, which is $2^n \cdot n^{2n}$. Here is a table showing how fast this grows:

n	# of interval sets
3	3375
4	614656
5	184528125
6	$8.2653950016 \cdot 10^{10}$
7	$5.1676101936 \cdot 10^{13}$

   Narrowing down on the interval sets we need to consider would be super beneficial!

6. Turning a set of intervals into a code: This process is known as "interval stabbing" and can be done in $O(n \log n)$. So turning all such sets of intervals into codes would take $O(2^n \cdot n^{2n} \cdot (n \log n))$.

7. Getting rid of duplicate codes using `unique`: Each call to `unique` takes $O(n \cdot k)$. Also, note $\log \left(2n^2\right)^n = n \log 2n^2$. So a bound on the time would be $O(2^n \cdot n^{2n} \cdot n\,(1 + 2\log n) \cdot n \cdot k)$.

8. Getting rid of isomorphic codes: $O(c^2 \cdot n^2 \cdot n! \cdot k)$ as discussed above, or roughly $O(c \cdot n^2 \cdot n! \cdot k)$ after optimization.

   Here c is the number of codes after getting rid of all duplicates; it would be nice to get an asymptotic (even if very rough) estimate of this.

The above determines all codes on n indices in 1 dimension. Here is the amount of time the (unoptimized) algorithm takes to run:

n	Time
3	$<$ 0.1 seconds
4	about 30 seconds
5	???

Now, using the product theorem, we can find all codes on n indices and d dimensions:

1. Solve (or load in) a solution to lower dimensional cases.
2. Intersection of two codewords: Since the codewords are sorted, this takes $O(n)$ time.
3. Intersection product of two codes: We need to find all intersections. If the codes have k_1 and k_2 codewords, then this is $O(n \cdot k_1 \cdot k_2)$. The result is probably not a valid code (the duplicates need to be removed), which would take roughly $O(k_1 \cdot k_2 \cdot \log(k_1 \cdot k_2))$ time.
4. All possible intersection products: To get from dimension d_1, d_2 to dimension $d_1 + d_2$, we intersection product all possible codes together. But notably the codes from one of the dimensions should **include isomorphic codes**! Otherwise, some codes may not show up. For example $\{1234, 12, 34, \emptyset\} \mathbin{\cap\!\!\!\!\!\times} \{1234, 13, 24, \emptyset\}$ is meaningful, even though the two codes being intersected are isomorphic.

   Suppose we have c_1 codes (including all isomorphic codes) of dimension d_1 and c_2 codes (removing isomorphic codes) of dimension d_2. Then we'll need to do $c_1 \cdot c_2$ intersection products, so this is in $O(c_1 \cdot c_2 \cdot k_1 \cdot k_2 \cdot \log(k_1 \cdot k_2))$. This could possibly be very slow.

In the case $n = 4$, here is the runtime of the algorithm for various values of d. We will see that beyond $d = 3$, there are no new codes on four indices.

(n,d)	Time
(4,1)	30 seconds
(4,2)	About 5 minutes
(4,3)	About 12 minutes

2.2 Algorithm results

For the programs (written in Standard ML) and raw data used in this analysis, see https://github.com/kevinazhou150/box-convex-codes.

The number of different codes on n indices in d dimensions (up to isomorphism) is given by the following table:

n	Dimension			
	1	2	3	4
1	2	2	2	2
2	5	5	5	5
3	25	31	31	31
4	185	600	607	607

For example, there are 600 different codes on 4 indices and 2 dimensions. Of particular interest are the 7 codes which are realizable with boxes in $\mathbb{R}^3$ but not $\mathbb{R}^2$, which we list below:

1. $\{1234, 134, 234, 12, 13, 14, 23, 24, 34, 1, 2, 3, 4, \emptyset\}$
2. $\{1234, 134, 234, 12, 13, 14, 23, 24, 1, 2, 3, 4, \emptyset\}$
3. $\{1234, 234, 12, 13, 14, 23, 24, 34, 1, 2, 3, 4, \emptyset\}$
4. $\{1234, 234, 12, 13, 14, 24, 34, 1, 2, 3, 4, \emptyset\}$
5. $\{1234, 234, 13, 14, 23, 24, 34, 1, 2, 3, 4, \emptyset\}$
6. $\{1234, 12, 13, 14, 23, 24, 34, 1, 2, 3, 4, \emptyset\}$
7. $\{1234, 13, 14, 23, 24, 34, 1, 2, 3, 4, \emptyset\}$

Note that at some point, adding more dimensions does not increase the number of box-convex codes that can be created. We can make this observation precise:

Theorem 7. *Let $\#Box(n, d)$ be the number of box-convex codes on n indices and d dimensions, up to isomorphism. If $\#Box(n, d) = \#Box(n, d + 1)$, then $\#Box(n, d')$ stays constant for all $d' \geq d$.*

Proof: Any code that is realizable in $\mathbb{R}^d$ by boxes is realizable in $\mathbb{R}^{d+1}$ by boxes. Suppose for contradiction that $\#\mathrm{Box}(n, d) = \#\mathrm{Box}(n, d + 1) < \#\mathrm{Box}(n, d + 2)$. Then there must be a code $\mathcal{C}$ on n indices with $\mathrm{bdim}(\mathcal{C}) = d + 2$. Let $\mathcal{U}$ be a realization of $\mathcal{C}$ in $\mathbb{R}^{d+2}$. We take the first $d + 1$ coordinates to get $\mathcal{U}_1$, and the last coordinate to get $\mathcal{U}_2$.

Since $\mathrm{code}(\mathcal{U}_1)$ is realizable in $d + 1$ dimensions, by assumption there is a realization of $\mathrm{code}(\mathcal{U}_1)$ (call it $\mathcal{U}_1'$) in d dimensions. Now we just take the Cartesian product of $\mathcal{U}_1'$ and $\mathcal{U}_2$, which is a realization of $\mathrm{code}(\mathcal{U})$ in $d + 1$ dimensions, a contradiction. $\square$

Remark: One open question is whether Theorem 7 holds for convex codes, rather than box-convex codes.

This leads to a natural question of what the maximum non-infinite embedding dimension for box-convex codes is. For example, for $n = 4$, this maximum embedding dimension is 3. This leads into the next section, where we prove box-convex codes can have embedding dimension at least $\binom{n}{\lfloor n/2 \rfloor}/2$. This lines up with the fact that $\binom{4}{2}/2 = 3$.

3 Box-convex codes with embedding dimension $\binom{n}{\lfloor n/2 \rfloor}/2$

Here, we make progress toward answering what the maximum finite embedding dimension of a code is. Currently known bounds are:

$$\binom{n-1}{\lfloor (n-1)/2 \rfloor} \leq \max_{\mathcal{C} \text{ on } n \text{ indices}} \operatorname{odim}(\mathcal{C}) \leq \binom{n}{\lfloor n/2 \rfloor}$$
$$n-1 \leq \max_{\mathcal{C} \text{ on } n \text{ indices}} \operatorname{cdim}(\mathcal{C}) \leq 2^n$$

We will prove the following in this section, by providing a construction of a code.

Theorem 8 (Lower bound on box-embedding dimension)**.**

$$\binom{n}{\lfloor n/2 \rfloor}/2 \leq \max_{\mathcal{C} \text{ on } n \text{ indices}} \operatorname{bdim}(\mathcal{C})$$

Proof: I claim that the following code has bdim $\binom{n}{\lfloor n/2 \rfloor}/2$:

$$\mathcal{C} := [n] \cup \left\{\sigma \mid \sigma \text{ has size } \leq \frac{n}{2}\right\}$$

For example the code $\{1234, 12, 13, 14, 23, 24, 34, 1, 2, 3, 4\}$ has bdim 3 since $\binom{4}{2}/2 = 3$.

Let's write $\mathcal{C} = \mathcal{C}_1 \boxtimes \mathcal{C}_2 \boxtimes \cdots \boxtimes \mathcal{C}_d$, where each $\mathcal{C}_i$ has bdim 1, and d is minimal. Therefore, d is the embedding dimension. Let's first show $d \geq \binom{n}{n/2}/2$. Note that $[n]$ is in each $\mathcal{C}_i$. Therefore, $\mathcal{C}_i \subseteq \mathcal{C}$ for all i, so codewords of size strictly greater than $\frac{n}{2}$ can't be in any $\mathcal{C}_i$.

We claim that each $\mathcal{C}_i$ has at most two codewords of size $\frac{n}{2}$. For arbitrary i, suppose that $\mathcal{C}_i$ has three distinct codewords of size $\frac{n}{2}$: $\sigma_1, \sigma_2, \sigma_3$. Each is a subset of $[n]$, and the codewords do not contain each other. So, by Lemma 4, $\mathcal{C}_i$ is not realizable in $\mathbb{R}$.

There are $\binom{n}{n/2}$ codewords of size $\frac{n}{2}$. Since each of the codewords of size exactly $\frac{n}{2}$ must appear in at least one of the $\mathcal{C}_i$'s, and each $\mathcal{C}_i$ has at most 2 codewords of size $\frac{n}{2}$, this shows that $d \geq \binom{n}{n/2}/2$.

Now, the following construction shows that $d \leq \binom{n}{n/2}/2$. Label the $\binom{n}{n/2}$ codewords $\sigma_0, \sigma_1, \sigma_2, \sigma_3, \cdots$ such that $\sigma_{2i}, \sigma_{2i+1}$ are disjoint sets. For example, for $n = 6$, this would look like

$$\begin{array}{ccccc} 123, 456, & 124, 356, & 125, 346, & 126, 345, & 134, 256, \\ 135, 246, & 136, 245, & 145, 236, & 146, 235, & 156, 234 \end{array}$$

Now we take $\mathcal{C}_i = \{[n], \sigma_{2i}, \sigma_{2i+1}, \emptyset\}$. Each $\mathcal{C}_i$ is realizable in $\mathbb{R}$ by having $[n]$ in the middle and the codewords $\sigma_{2i}, \sigma_{2i+1}$ on either side. Then, for any codeword τ of size at most $\frac{n}{2}$, we may write it as

$$(\tau \cup \tau_1) \cap (\tau \cup \tau_2)$$

where τ_1, τ_2 are disjoint, and $|\tau \cup \tau_1| = \frac{n}{2}, |\tau \cup \tau_2| = \frac{n}{2}$. Unless $\tau = \emptyset$ (which is always in $\mathcal{C}$), note that $\tau \cup \tau_1$ and $\tau \cup \tau_2$ are not disjoint. Therefore they live in different $\mathcal{C}_i$'s, so we may take the intersection of them (and for the rest of the intersections, use $[n]$). Now,

$$\begin{aligned} (\tau \cup \tau_1) \cap (\tau \cup \tau_2) &= \tau \cup (\tau_1 \cap \tau_2) \\ &= \tau && \text{(Since } \tau_1, \tau_2 \text{ disjoint by assumption)} \end{aligned}$$

Thus, $\tau \in \mathcal{C}$ for any τ with size at most $\frac{n}{2}$, as desired. $\square$

It remains open whether this is the best upper bound. We can show a silly upper bound on $\mathrm{bdim}(\mathcal{C})$, which is the number of codes on n indices, which is bounded by 2^{2^n}.

Theorem 9. $\mathrm{bdim}(\mathcal{C}) \leq$ *the number of codes on n indices. Therefore,*

$$\max_{\mathcal{C} \text{ on } n \text{ indices}} \mathrm{bdim}(\mathcal{C}) \leq 2^{2^n}$$

Proof: Let $\mathcal{C}$ be box-convex, with $\mathrm{bdim}(\mathcal{C}) = k$. By the product theorem we may write $\mathcal{C} = \mathcal{C}_1 \boxtimes \mathcal{C}_2 \boxtimes \ldots \boxtimes \mathcal{C}_k$. Furthermore define $\mathcal{D}_i = \mathcal{C}_1 \boxtimes \mathcal{C}_2 \boxtimes \ldots \boxtimes \mathcal{C}_i$. If $\mathcal{D}_i = \mathcal{D}_j$ for some $1 \leq i < j \leq k$, then we can write

$$\begin{aligned}
\mathcal{C} &= (\mathcal{C}_1 \boxtimes \mathcal{C}_2 \boxtimes \ldots \boxtimes \mathcal{C}_j) \boxtimes (\mathcal{C}_{j+1} \boxtimes \mathcal{C}_{j+2} \ldots \boxtimes \mathcal{C}_k) \\
&= (\mathcal{C}_1 \boxtimes \mathcal{C}_2 \boxtimes \ldots \boxtimes \mathcal{C}_i) \boxtimes (\mathcal{C}_{j+1} \boxtimes \mathcal{C}_{j+2} \ldots \boxtimes \mathcal{C}_k)
\end{aligned}$$

This gives us a realization in a smaller dimension, which contradicts k being minimal. Therefore by the pigeonhole principle, $k \leq 2^{2^n}$, the total number of codes on n indices. $\square$

It is currently unknown whether we can get a better upper bound on the box-embedding dimension. One idea is to consider each codeword separately: If $\sigma \in \mathcal{C}$ and $\sigma \in \mathcal{D}_i$, then σ will continue to be in $\mathcal{D}_i'$ for $i' > i$. Furthermore, if $\tau \in \Delta(\mathcal{C}) \setminus \mathcal{C}$, then τ can never appear in any $\mathcal{D}_i$ (because $\tau \subseteq \sigma$ for some $\sigma \in \mathcal{C}$, and σ must be a subset of some codeword in $\mathcal{D}_i$ because of intersection products.) However, we do not know anything about codewords that are a superset of codewords in $\mathcal{C}$.

4 Inductively pierced codes

We first provide a self-contained introduction to inductively pierced codes, with information from Curto, Itskov, Veliz-Cuba, and Youngs'[5] explanation of the neural ring. We also use Curry, Jeffs, Youngs, and Zhao's[2] treatment of inductively pierced codes.

4.1 Neural Rings

The receptive field structure (RF structure) of a code $\mathcal{C}$ with realization $\mathcal{U}$ is the set of relations of the following form, where σ, τ are disjoint.

$$\bigcap_{i\in\sigma} U_i \subseteq \bigcup_{j\in\tau} U_j$$

For a code $\mathcal{C}$ on n indices, a codeword $\sigma \in \mathcal{C}$ can be viewed as a bitvector in $\{0,1\}^n$. This can further be associated with the polynomial $\prod_{i\in\sigma} x_i \prod_{j\notin\sigma}(1-x_j)$, which evaluates to 1 on σ and 0 on any other codeword. With this, we define the ideal $J_{\mathcal{C}}$, which is the ideal generated from the characteristic functions of all *non-codewords* of $\mathcal{C}$.

$$J_{\mathcal{C}} = \langle\ \{\prod_{i\in\sigma} x_i \prod_{j\notin\sigma}(1-x_j)\ \mid \sigma \notin \mathcal{C}\}\ \rangle$$

For example, let $\mathcal{C} = \{123, 12, 1, \emptyset\}$. Then

$$\begin{aligned} J_{\mathcal{C}} &= \langle\ \{x_2x_3(1-x_1), \quad x_2(1-x_1)(1-x_3), \quad x_1x_3(1-x_2), \quad x_3(1-x_1)(1-x_2)\ \}\rangle \\ &= \langle\ \{x_2(1-x_1), \quad x_3(1-x_2)\}\ \rangle \end{aligned}$$

We define pseudo-monomials as polynomials of the form $\prod_{i\in\sigma} x_i \prod_{j\in\tau}(1-x_j)$, where σ, τ are disjoint. The canonical form $CF(J_{\mathcal{C}})$ is the set of all minimal pseudo-monomials in $J_{\mathcal{C}}$, that is, all pseudo-monomials which are not divisible by another pseudo-monomial in $J_{\mathcal{C}}$.

Curto, Itskov, Veliz-Cuba, and Youngs proved that the pseudo-monomials in $CF(J_{\mathcal{C}})$ correspond to the following RF relations:

- $\prod_{i\in\sigma} x_i \in CF(J_{\mathcal{C}})$ means $U_\sigma = \emptyset$, and:
  - $U_{\sigma'}$ is nonempty for any $\sigma' \subset \sigma$.
- $\prod_{i\in\sigma} x_i \prod_{j\in\tau}(1-x_j) \in CF(J_{\mathcal{C}})$ means $\bigcap_{i\in\sigma} U_i \subseteq \bigcup_{j\in\tau} U_j$, and:
  - If $\sigma' \supset \sigma$ then $\bigcap_{i\in\sigma'} U_i \nsubseteq \bigcup_{j\in\tau} U_j$.
  - If $\tau' \subset \tau$ then $\bigcap_{i\in\sigma} U_i \nsubseteq \bigcup_{j\in\tau'} U_j$.
- $\prod_{j\in\tau}(1-x_j) \in CF(J_{\mathcal{C}})$ means $\mathbb{R}^d = \bigcup_{j\in\tau} U_j$, and τ is a minimal codeword with this property. We won't consider these pseudo-monomials since we generally require $\emptyset \in \mathcal{C}$.

4.2 Introduction to inductively pierced codes

First, we define the interval between σ, τ to be $[\sigma, \tau] := \{\gamma : \sigma \subseteq \gamma \subseteq \tau\}$. The rank of $[\sigma, \tau]$ is defined to be $|\tau \setminus \sigma|$. A k-inductively pierced code on $[n]$ is:

- $\{\emptyset\}$, or
- $\mathcal{C}$, where $\mathcal{C} = (\mathcal{C} \setminus i) \cup [\sigma \cup \{i\}, \tau \cup \{i\}]$, and $[\sigma, \tau]$ is an interval in $\mathcal{C} \setminus i$ with rank k, and $\mathcal{C} \setminus i$ is k-inductively pierced.
  - We define $\mathcal{C} \setminus i$ to be the code $\{\sigma \setminus \{i\} : \sigma \in \mathcal{C}\}$.

An inductively pierced code is just a k-inductively pierced code for some value of k.

Gross, Obatake, and Youngs proved in [7] that for an inductively pierced code $\mathcal{C}$, the canonical form $CF(J_{\mathcal{C}})$ only has pseudo-monomials of degree 2, which are $x_i x_j$ (for some $i, j \in [n]$) and $x_i(1 - x_j)$.

One important property that a code $\mathcal{C}$ can have is being *intersection-complete*: for every $\sigma, \tau \in \mathcal{C}$, the intersection $\sigma \cap \tau \in \mathcal{C}$. Curto, Gross, Jeffries, Morrison, Rosen, Shiu, and Youngs [4] proved that a code $\mathcal{C}$ is intersection-complete if $CF(J_{\mathcal{C}})$ only has monomials and pseudo-monomials of the form $x_i(1 - x_j)$. Therefore, inductively pierced codes are intersection-complete.

Furthermore, Jeffs [8] proved that $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$ for intersection complete codes $\mathcal{C}$.

4.3 Inflatable codes

The goal in this section is to show $\operatorname{cdim}(\mathcal{C}) \geq \operatorname{odim}(\mathcal{C})$. We do this by proving that every closed realization of an inductively pierced code can be "inflated" by making every set larger by some ε and taking interiors. First, we provide some conditions on the RF relations of "inflatable" codes.

Definition 8 (Inflatable)**.** *We say a code $\mathcal{C}$ is inflatable if the following implication holds for all $\sigma, \tau \subseteq [n]$:*

$$\emptyset \neq \bigcap_{i \in \sigma} U_i \subseteq \bigcup_{j \in \tau} U_j \Longrightarrow \exists i \in \sigma, U_i \subseteq \bigcup_{j \in \tau} U_j$$

Theorem 10. *A code $\mathcal{C}$ satisfies (Inflatable) iff $CF(J_{\mathcal{C}})$ only has monomials $\prod_{i \in \sigma} x_i$ and pseudomonimals of the form $x_i \prod_{j \in \tau}(1 - x_j)$.*

Proof of $\Longrightarrow$: Let $\sigma, \tau \in \mathcal{C}$. By definition,

$$\bigcap_{i \in \sigma} U_i \subseteq \bigcup_{j \in \tau} U_j \Longleftrightarrow \prod_{i \in \sigma} x_i \prod_{j \in \tau}(1 - x_j) \in J_{\mathcal{C}}$$

In the case $\bigcap_{i \in \sigma} U_i = \emptyset$, then the monomial $\prod_{i \in \sigma} x_i$ vanishes, so it's in $J_{\mathcal{C}}$. If $\bigcap_{i \in \sigma} U_i$ is nonempty, the inflatable condition gives us the corresponding RF relation $x_i \prod_{j \in \tau}(1 - x_j)$ for some i.

Proof of $\Longleftarrow$: Suppose that $\bigcap_{i \in \sigma} U_i \subseteq \bigcup_{j \in \tau} U_j$. If this corresponds to the monomial $\prod_{i \in \sigma} x_i$, that means $\bigcap_{i \in \sigma} U_i = \emptyset$ so there is nothing to prove. If it instead corresponds to a pseudomonomial $x_i \prod_{j \in \tau}(1 - x_j)$, then $U_i \subseteq \bigcup_{j \in \tau} U_j$ as desired. $\square$

Theorem 10 shows that inductively pierced codes, which only have pseudo-monomials of the form $x_i x_j$ and $x_i(1 - x_j)$, are inflatable. Now we show that inflatable codes are, well, inflatable: we can make each set of a realization slightly larger and take its interior, without changing the realized code.

Theorem 11. *A closed convex code $\mathcal{C}$ that satisfies (Inflatable) is also an open convex code.*

Proof: Let $\mathcal{C}$ have a realization $\{C_1, C_2, \cdots, C_n\}$, where each C_i is a closed convex set. Let D_i be the Minkowski sum of C_i with an ε-ball, where ε will be chosen later. We want to show that the RF relations $\bigcap_{i \in \sigma} C_i \subseteq \bigcup_{j \in \tau} C_j$ stay the same. We break this into three cases:

1. $\left(\bigcap_{i \in \sigma} C_i = \emptyset\right) \Longrightarrow \left(\bigcap_{i \in \sigma} D_i = \emptyset\right)$
2. $\left(\bigcap_{i \in \sigma} C_i \not\subseteq \bigcup_{j \in \tau} C_j\right) \Longrightarrow \left(\bigcap_{i \in \sigma} D_i \not\subseteq \bigcup_{j \in \tau} D_j\right)$
3. $\left(\emptyset \neq \bigcap_{i \in \sigma} C_i \subseteq \bigcup_{j \in \tau} C_j\right) \Longrightarrow \left(\bigcap_{i \in \sigma} D_i \subseteq \bigcup_{j \in \tau} D_j\right)$ This is the only case requiring the (Inflatable) condition.

Case 1. Assume $\bigcap_{i\in\sigma} C_i$ is empty. For all points $p \in \mathbb{R}^d$, we can define a continuous function $f : \mathbb{R}^d \to \mathbb{R}$:

$$f(p) = \sum_{i\in\sigma} \operatorname{dist}(p, C_i)$$

If $p \notin C_i$, then since $\mathbb{R}^d \setminus C_i$ is open, we can find a ball centered at p with positive radius contained in $\mathbb{R}^d \setminus C_i$. Therefore $\operatorname{dist}(p, C_i) > 0$. As a result $f(p)$ is never 0. Now choosing $\varepsilon < \frac{f(p)}{|\sigma|}$, then for all points $p \in \mathbb{R}^d$:

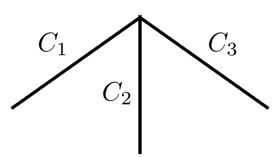
$$\begin{aligned}\sum_{i\in\sigma} \operatorname{dist}(p, D_i) &\geq \sum_{i\in\sigma} (\operatorname{dist}(p, C_i) - \varepsilon) \\ &= f(p) - \varepsilon|\sigma| \\ &> 0\end{aligned}$$

If $p \in \bigcap_{i\in\sigma} D_i$, then $\sum_{i\in\sigma} \operatorname{dist}(p, D_i) = 0$, contradicting the above. Therefore $\bigcap_{i\in\sigma} D_i$ is empty as desired.

Case 2. Suppose $\bigcap_{i\in\sigma} C_i \not\subseteq \bigcup_{j\in\tau} C_j$. So there is some point $p \in \bigcap_{i\in\sigma} C_i \setminus \bigcup_{j\in\tau} C_j$. As $\bigcup_{j\in\tau} C_j$ is closed, p has some positive distance d from $\bigcup_{j\in\tau} C_j$. Choosing $\varepsilon < \frac{d}{2}$, then $p \in \bigcap_{i\in\sigma} D_i$ since $C_i \subseteq D_i$. On the other hand, $p \notin \bigcup_{j\in\tau} D_j$ because $\operatorname{dist}(p, \bigcup_{j\in\tau} D_j) > \frac{d}{2}$. Therefore $\bigcap_{i\in\sigma} D_i \not\subseteq \bigcup_{j\in\tau} D_j$.

Case 3. Suppose $\emptyset \neq \bigcap_{i\in\sigma} C_i \subseteq \bigcup_{j\in\tau} C_j$. By (Inflatable), there exists $k \in \sigma$ such that $C_k \subseteq \bigcup_{j\in\tau} C_j$. If a point $p \in D_k$, then $p = x + y$ where $x \in C_k, y \in B(0, \varepsilon)$. Therefore $x \in C_j$ for some $j \in \tau$, so $x + y \in \bigcup_{j\in\tau} D_j$. Therefore $\bigcap_{i\in\sigma} D_i \subseteq D_k \subseteq \bigcup_{j\in\tau} D_j$. □

To demonstrate why (Inflatable) is necessary, consider the code $\{123, 1, 2, 3, \emptyset\}$ with the following realization.



Note that $C_1 \cap C_2$ is the intersection point, which is a subset of C_3. However if we inflate each set, there may be a point in $(C_1 \cap C_2) \setminus C_3$.

Corollary 12. *If $\mathcal{C}$ is an inductively pierced code, then* $\operatorname{cdim}(\mathcal{C}) = \operatorname{odim}(\mathcal{C})$.

Proof: From above, $\mathcal{C}$ is inflatable. Therefore, given a closed convex realization of $\mathcal{C}$, Theorem 11 gives us an open convex realization in the same dimension. Thus $\operatorname{cdim}(\mathcal{C}) \geq \operatorname{odim}(\mathcal{C})$. Furthermore, $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$ for intersection-complete codes, so $\operatorname{cdim}(\mathcal{C}) = \operatorname{odim}(\mathcal{C})$. □

4.4 Inductively pierced codes are box-convex

One natural question is whether inductively pierced codes are realizable by boxes. The answer turns out to be yes, and in not too many dimensions either.

Theorem 13. *If $\mathcal{C}$ is an inductively pierced code on n indices, then $\mathcal{C}$ is realizable by boxes in dimension at most $n - 1$.*

Proof: We prove by induction on n, the number of indices.

Base case: The inductively pierced codes for $n = 1, 2$ are all realizable in $\mathbb{R}$.

Inductive step: Let $\mathcal{C}$ on $2^{[n]}$ be an inductively pierced code. Then $\mathcal{C} = (\mathcal{C} \setminus n) \cup [\sigma \cup \{n\}, \tau \cup \{n\}]$. By the induction hypothesis, $\mathcal{C} \setminus n$ has a realization by boxes $\mathcal{U}$ in $\mathbb{R}^d$ where $d \leq n - 2$.

Consider $\bigcap_{i \in \sigma} U_i$. This is a box as it's the intersection of finitely many boxes. Now let's create a new realization $\mathcal{V}$ in $\mathbb{R}^{d+1}$, defined as follows:

$$\begin{cases} V_i = U_i \times (0,1) & i \notin \tau, i \neq n \\ V_i = U_i \times (0,3) & i \in \tau \\ V_n = \left(\bigcap_{i \in \sigma} U_i\right) \times (2,3) \end{cases}$$

Within the strip where the $(d+1)$th coordinate is in $(0,1)$, we exactly have the codewords in $\mathcal{C} \setminus n$.

Within $\left(\bigcap_{i \in \sigma} U_i\right) \times (2,3)$, only the codewords that are a superset of $\sigma \cup \{n\}$ appear; also, the codewords here are subsets of $\tau \cup \{n\}$. Also, all the codewords in $[\sigma \cup \{n\}, \tau \cup \{n\}]$ appear, since the interval $[\sigma, \tau]$ is in $\bigcap_{i \in \sigma} U_i$.

Now we must show there are no extra codewords. Consider the strip where the $(d+1)$th coordinate is in $[1,3)$. If there is a new codeword c in this strip, it must be a subset of τ. Pick a point p which witnesses the codeword c. We can project the $(d+1)$th coordinate of p to 0.5 to get a new point that witnesses a codeword c'. Note that $c' \cap \tau = c$. So, $c' \in \mathcal{C} \setminus n$ and $\tau \in \mathcal{C} \setminus n$. Therefore, as inductively pierced codes are intersection-complete, $c' \cap \tau \in \mathcal{C} \setminus n$ as well. $\square$

To illustrate the construction of $\mathcal{V}$, consider the inductively pierced code $\mathcal{C} = \{123, 12, 23, 1, 2, \emptyset\} \cup \{234, 24\}$. That is, the interval $[2, 23]$ was most recently pierced. Starting with a realization of $\{123, 12, 23, 1, 2, \emptyset\}$ in $\mathbb{R}$, we thicken the boxes in $\mathbb{R}^2$ to get the sets V_1, V_2, V_3. Then we add the set V_4 slicing through the top part of V_2.

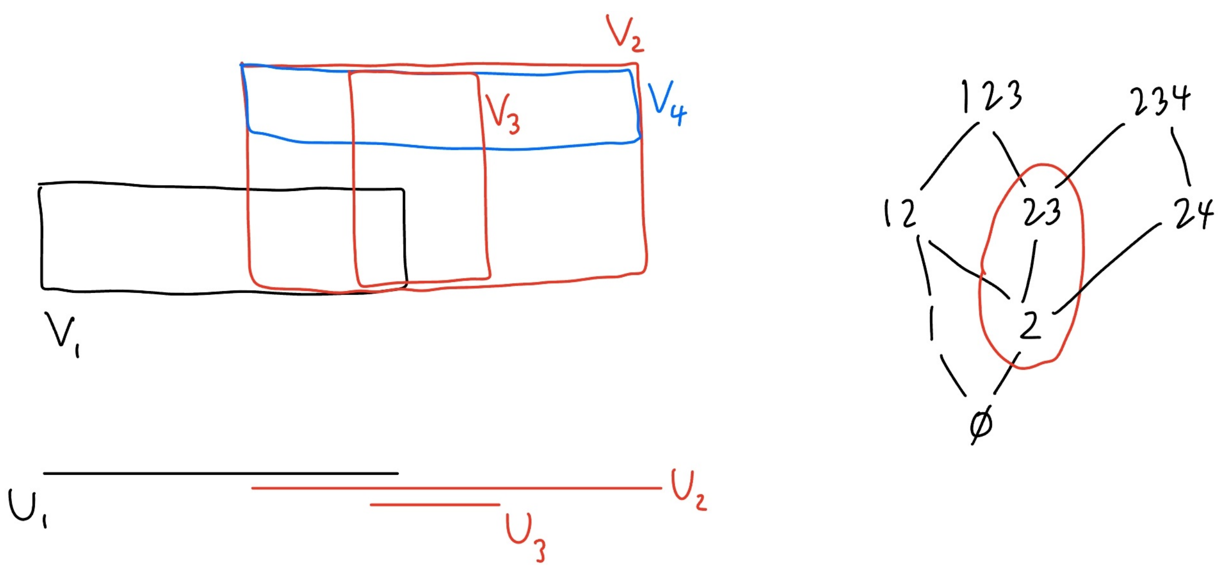


Figure 3: Left: A realization of the code $\{123, 12, 23, 1, 2, \emptyset\}$ in $\mathbb{R}$, and the construction of $\mathcal{V}$ which realizes the code $\{123, 234, 12, 23, 24, 1, 2, \emptyset\}$. Right: The codewords of $\mathcal{C}$ visualized using a poset under inclusion. The circled codewords is the pierced interval $[2, 23]$.

5 Open questions

Here are several questions inspired by the above work on embedding dimension and box-convex codes.

1. Does Theorem 7 hold for convex codes, rather than just box-convex codes?
2. Is there an asymptotically tighter upper bound on the maximum box-embedding dimension than 2^{2^n}? One idea is, given a code $\mathcal{C}$, to consider a set of codewords where no codeword contains another (that is, an antichain in the poset under inclusion: this has size at most $\binom{n}{n/2}$ by Dilworth's theorem.)
3. For codes $\mathcal{C}, \mathcal{D}$, what does $\mathrm{CF}(J_{\mathcal{C} \boxtimes \mathcal{D}})$ look like when compared to $\mathrm{CF}(J_{\mathcal{C}})$ and $\mathrm{CF}(J_{\mathcal{D}})$?

6 Acknowledgements

Thanks to Amzi Jeffs for introducing me to this spectacular problem in the intersection of combinatorics and convex geometry, and giving valuable feedback on my first drafts. Furthermore, thanks to Miguel Benitez, Siran Chen, Tiffany Han, and Kinapal Paguyo for a great collaboration experience on [1].

References

[1] Miguel Benitez, Siran Chen, Tiffany Han, Amzi Jeffs, Kinapal Paguyo, and Kevin Zhou. Realizing convex codes with axis-parallel boxes.

[2] Ryan Curry, R. Amzi Jeffs, Nora Youngs, and Ziyu Zhao. Recognizing and realizing inductively pierced codes. *arXiv e-prints:2207.06266*, 2022.

[3] Carina Curto, Elizabeth Gross, Jack Jeffries, Katherine Morrison, Mohamed Omar, Zvi Rosen, Anne Shiu, and Nora Youngs. What makes a neural code convex? *SIAM J. Appl. Algebra Geom.*, 1(1):222–238, 2017.

[4] Carina Curto, Elizabeth Gross, Jack Jeffries, Katherine Morrison, Zvi Rosen, Anne Shiu, and Nora Youngs. Algebraic signatures of convex and non-convex codes. *Journal of Pure and Applied Algebra*, 223(9):3919–3940, 2019.

[5] Carina Curto, Vladimir Itskov, Alan Veliz-Cuba, and Nora Youngs. The neural ring: an algebraic tool for analyzing the intrinsic structure of neural codes. *Bulletin of Mathematical Biology*, 75(9):1571–1611, 2013.

[6] Jürgen Eckhoff. Intersection properties of boxes. part i: An upper-bound theorem. *Israel Journal of Mathematics*, 62(3):283–301, 1988.

[7] Elizabeth Gross, Kazi Obatake Nida, and Nora Youngs. Neural ideals and stimulus space visualization. *Advances in Applied Mathematics*, 95:65–95, 2016.

[8] R. Amzi Jeffs. Embedding dimension phenomena in intersection complete codes. *Selecta Mathematica*, 28, 2021.

[9] Zvi Rosen and Yan Zhang. Convex neural codes in dimension 1.