Neural Bonanza III

The Final Bonanza, Pt. II

Brianna Gambacini, University of Connecticut

Sam Macdonald, Willamette University

Definition

A code C is **max-intersection complete** if all the intersections of its facets are in C. If a code does not contain all of its facets' intersections then it is **max-intersection incomplete**.

Definition

For a neural code C on n vertices, the **simplicial complex** $\Delta(C)$ is a subset of $2^{[n]}$ that is closed under taking subsets, where $[n] := \{1, 2, \ldots, n\}$ is the population of neurons. More specifically:

$$\Delta(C) := \{\sigma \subseteq [n] : \sigma \subseteq \alpha \text{ for some } \alpha \in C\}.$$

Definition

Let Δ be a simplicial complex on n vertices and $\sigma \in \Delta$. Then the **link** of σ in Δ is:

$$\mathrm{Lk}_\sigma(\Delta) := \{\tau \subseteq [n]\backslash\sigma : \sigma \cup \tau \in \Delta\}.$$

Theorem

Let C be a locally good neural code on n neurons. If no intersection of facets contains more than one neuron, then C is max-intersection complete.

Theorem

Let C be a locally good neural code on n neurons. If no intersection of facets contains more than one neuron, then C is max-intersection complete.

Proof.

Let C be a locally good neural code on n neurons with distinct facets $M_1, M_2, \ldots, M_n$ such that no intersection of facets contains more than one neuron.

Theorem

Let C be a locally good neural code on n neurons. If no intersection of facets contains more than one neuron, then C is max-intersection complete.

Proof.

Let C be a locally good neural code on n neurons with distinct facets $M_1, M_2, \dots, M_n$ such that no intersection of facets contains more than one neuron.

Suppose by way of contradiction that C is max intersection incomplete. Thus there must exist some neuron $\sigma \notin C$ and $M_i, M_j \in C$ such that $M_i \cap M_j = \sigma$.

Theorem

Let C be a locally good neural code on n neurons. If no intersection of facets contains more than one neuron, then C is max-intersection complete.

Proof.

Let C be a locally good neural code on n neurons with distinct facets $M_1, M_2, \ldots, M_n$ such that no intersection of facets contains more than one neuron.

Suppose by way of contradiction that C is max intersection incomplete. Thus there must exist some neuron $\sigma \notin C$ and $M_i, M_j \in C$ such that $M_i \cap M_j = \sigma$.

As M_i and M_j are distinct, there must exist α, β such that $M_i = \sigma\alpha$ and $M_j = \sigma\beta$.

Proof.

Consider $\mathrm{Lk}_\sigma(\Delta)$. Recall that $\mathrm{Lk}_\sigma(\Delta) := \{\tau \subseteq [n]\backslash\sigma : \sigma \cup \tau \in \Delta\}$.

Proof.

Consider $\mathrm{Lk}_\sigma(\Delta)$. Recall that $\mathrm{Lk}_\sigma(\Delta) := \{\tau \subseteq [n] \backslash \sigma : \sigma \cup \tau \in \Delta\}$.

As $\sigma \cup \alpha$ and $\sigma \cup \beta \in \Delta(C)$, it must be the case that $\alpha, \beta \in \mathrm{Lk}_\sigma(\Delta)$.

Complete to the Max

Proof.

Consider $\mathrm{Lk}_\sigma(\Delta)$. Recall that $\mathrm{Lk}_\sigma(\Delta) := \{\tau \subseteq [n] \backslash \sigma : \sigma \cup \tau \in \Delta\}$.

As $\sigma \cup \alpha$ and $\sigma \cup \beta \in \Delta(C)$, it must be the case that $\alpha, \beta \in \mathrm{Lk}_\sigma(\Delta)$.

Thus, as it stands, $\mathrm{Lk}_\sigma(\Delta)$ is the following:

which is not contractible. There are three ways to make this link contactable, and we will show how each leads to a contradiction.

Case I: $\sigma\alpha\beta \in C$

Proof.

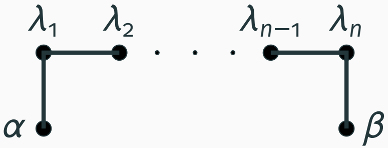
This would introduce $\sigma\alpha\beta$ to the code. This is either a facet of C or a subset of some facet in C. Either way, the intersection of this facet with M_i is $\sigma\alpha$, a contradiction.

Case II: There Exists Exactly One λ

Proof.

This would introduce $\sigma\alpha\lambda$ and $\sigma\beta\lambda$ to the code. These codewords are either facets of C or subsets of other facets in C. Either way, the intersection of these facets is $\sigma\lambda$, a contradiction.

Proof.



This would introduce quite a few things to the code. However, just focusing on α, λ_1, and λ_2, we see that both $\sigma\alpha\lambda_1$ and $\sigma\lambda_1\lambda_2$ are in the code, meaning this case also results in contradiction. Thus, C must be max intersection complete. $\square$

Three for the Price of Three!

Theorem

Let C be a 3-sparse locally good max intersection incomplete code. Then there must be at least three codewords of length three.

Three for the Price of Three!

Theorem

Let C be a 3-sparse locally good max intersection incomplete code. Then there must be at least three codewords of length three.

Proof.

Let C be a 3-sparse locally good max intersection incomplete code with distinct facets $M_1, M_2, \ldots, M_n$.

Three for the Price of Three!

Theorem

Let C be a 3-sparse locally good max intersection incomplete code. Then there must be at least three codewords of length three.

Proof.

Let C be a 3-sparse locally good max intersection incomplete code with distinct facets $M_1, M_2, \ldots, M_n$.

Then there must exist some neuron $\sigma \notin C$ and $M_i, M_j \in C$ such that $M_i \cap M_j = \sigma$.

Three for the Price of Three!

Theorem

Let C be a 3-sparse locally good max intersection incomplete code. Then there must be at least three codewords of length three.

Proof.

Let C be a 3-sparse locally good max intersection incomplete code with distinct facets $M_1, M_2, \ldots, M_n$.

Then there must exist some neuron $\sigma \notin C$ and $M_i, M_j \in C$ such that $M_i \cap M_j = \sigma$.

The facets M_i and M_j must be of at least length two to remain distinct, given their shared σ.

Three for the Price of Three!

Theorem

Let C be a 3-sparse locally good max intersection incomplete code. Then there must be at least three codewords of length three.

Proof.

Let C be a 3-sparse locally good max intersection incomplete code with distinct facets $M_1, M_2, \ldots, M_n$.

Then there must exist some neuron $\sigma \notin C$ and $M_i, M_j \in C$ such that $M_i \cap M_j = \sigma$.

The facets M_i and M_j must be of at least length two to remain distinct, given their shared σ.

However, if M_i and M_j were of length two, then σ would be a mandatory codeword.

Proof.

So M_i and M_j must be of length three.

Proof.

So M_i and M_j must be of length three.

However, this could not be the entire code, as this would make σ a mandatory codeword. As C is 3-sparse, there must exist some other facet $M_k \in C$ such that $M_i \cap M_j \cap M_k = \sigma$.

Proof.

So M_i and M_j must be of length three.

However, this could not be the entire code, as this would make σ a mandatory codeword. As C is 3-sparse, there must exist some other facet $M_k \in C$ such that $M_i \cap M_j \cap M_k = \sigma$.

To remain distinct from M_i and M_j, M_k must contain some neuron $\tau \notin M_i, M_j$. However, if both $M_i \cap M_k = \sigma$ and $M_j \cap M_k = \sigma$, then there would exist a local obstruction at σ.

Proof.

So M_i and M_j must be of length three.

However, this could not be the entire code, as this would make σ a mandatory codeword. As C is 3-sparse, there must exist some other facet $M_k \in C$ such that $M_i \cap M_j \cap M_k = \sigma$.

To remain distinct from M_i and M_j, M_k must contain some neuron $\tau \notin M_i, M_j$. However, if both $M_i \cap M_k = \sigma$ and $M_j \cap M_k = \sigma$, then there would exist a local obstruction at σ.

Thus, without loss of generality, there must exist some α such that $M_i \cap M_k = \sigma\alpha$. Therefore, as C is 3-sparse we know that $M_k = \sigma\alpha\tau$, completing the proof. □

Street C_{red}

Definition

For a 3-sparse neural code, the **reduced** code of C, denoted C_{red}, is the code containing all length three codewords of C and their subsets that are also in C.

Definition

For a 3-sparse neural code, the **reduced** code of C, denoted C_{red}, is the code containing all length three codewords of C and their subsets that are also in C.

Example

Consider the following neural code:

$$C = \{123, 134, 145, 13, 14, 26, 27, 29, 35, 37, 38, 46, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \emptyset\}.$$

Definition

For a 3-sparse neural code, the **reduced** code of C, denoted C_{red}, is the code containing all length three codewords of C and their subsets that are also in C.

Example

Consider the following neural code:

$$C = \{123, 134, 145, 13, 14, 26, 27, 29, 35, 37, 38, 46, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \varnothing\}.$$

$$C_{red} = \{123, 134, 145, 13, 14, \varnothing\}$$

Theorem

Let C be a 3-sparse neural code on n neurons. If there exists a closed convex cover $U = \{U_i\}_{i=1}^n$ in $\mathbb{R}^d$ of C_{red} such that every set in U can be realized as fully $\mathbb{R}^{d-1}$ or higher, then C is open convex.

Theorem

Let C be a 3-sparse neural code on n neurons. If there exists a closed convex cover $U = \{U_i\}_{i=1}^n$ in $\mathbb{R}^d$ of C_{red} such that every set in U can be realized as fully $\mathbb{R}^{d-1}$ or higher, then C is open convex.

Proof.

Let C be a 3-sparse locally good neural code on n neurons. Suppose that there exists some fully dimensional closed cover of C_{red}, denoted $U = \{U_i\}_{i=1}^n$ in $\mathbb{R}^d$. We will construct an open cover of C using U.

Proof.

$C =$
$\{123, 134, 145, 13, 14, 26, 27, 29, 35, 37, 38, 46, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \varnothing\}.$

U_{123}	U_{13}	U_{134}	U_{14}	U_{145}

Step One: Intersections of Neurons in C_{red}

Proof.

$C =$

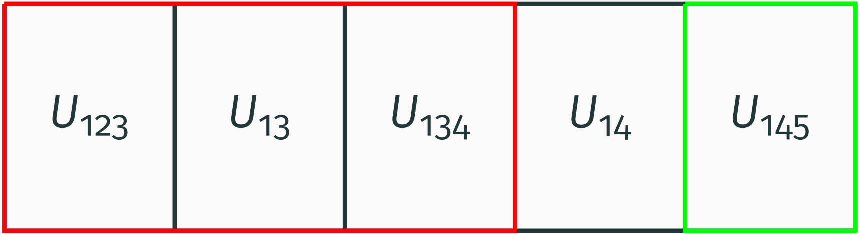
$\{123, 134, 145, 13, 14, 26, 27, 29, \mathbf{35}, 37, 38, 46, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \emptyset\}.$

U_{123}	U_{13}	U_{134}	U_{14}	U_{145}

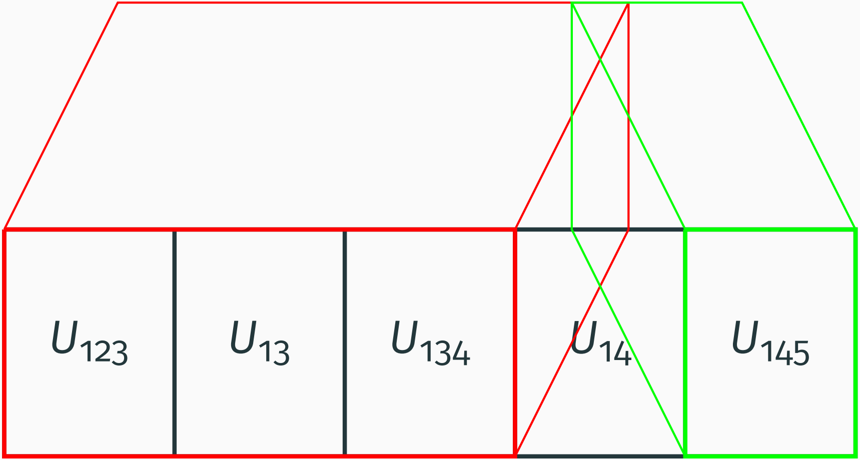
Proof.

$C =$

$\{123, 134, 145, 13, 14, 26, 27, 29, \mathbf{35}, 37, 38, 46, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \emptyset\}.$

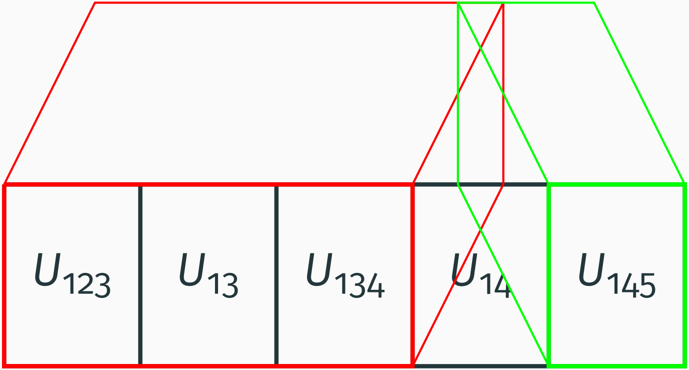


Proof.



Proof.

Using the same epsilonic procedure as was used in Theorem 4.3, we can make this new realization fully dimensional.



Proof.

The only neurons missing from U are the ones not involved in any triple-wise intersection. Let $A = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subset C$ denote the set of these neurons.

Proof.

The only neurons missing from U are the ones not involved in any triple-wise intersection. Let $A = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subset C$ denote the set of these neurons.

Begin with some α_i such that α_i fires with some neuron $\beta_1 \in C_{red}$.

Proof.

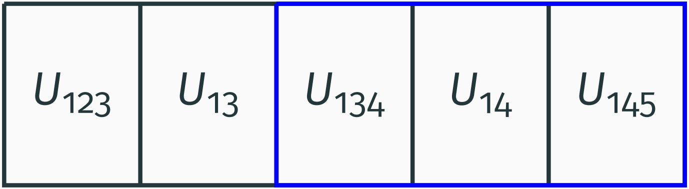
The only neurons missing from U are the ones not involved in any triple-wise intersection. Let $A = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subset C$ denote the set of these neurons.

Begin with some α_i such that α_i fires with some neuron $\beta_1 \in C_{red}$.

U_{123}	U_{13}	U_{134}	U_{14}	U_{145}

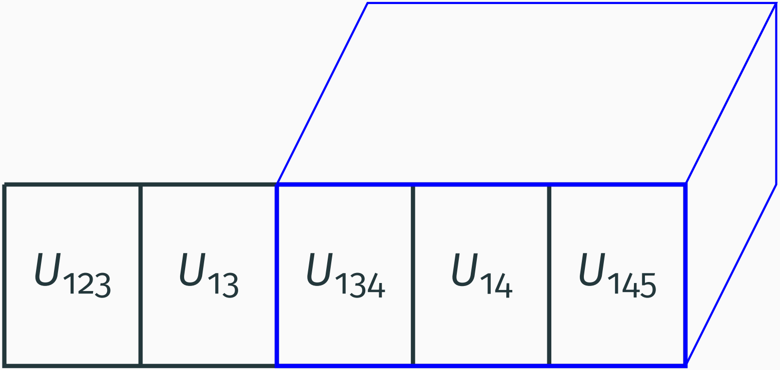
Proof.

$C =$
$\{123, 134, 145, 13, 14, 26, 27, 29, 35, 37, 38, \mathbf{46}, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \varnothing\}.$



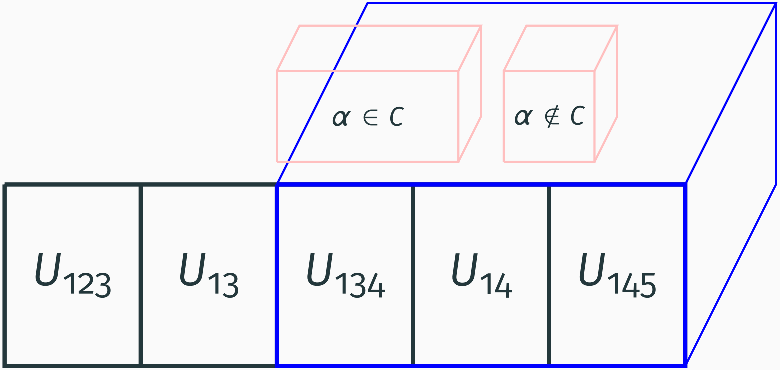
Proof.

$C =$
$\{123, 134, 145, 13, 14, 26, 27, 29, 35, 37, 38, \mathbf{46}, 48, 49, 58, 67, 79, 89, 2, 3, 4, 5, 6, 7, 8, 9, \emptyset\}.$



Proof.

In general, either $\alpha_i \in C$ or it isn't. If not, draw it as a subset of β_1. If so, draw it so that it overlaps with β_1.



Step Two: Neurons in C but not C_{red}

Proof.

Repeat this process for each $1 \leq i \leq n$, each time selecting a codeword that contains a neuron already existing in the realization. This provides us with a fully dimensional closed realization of C.

Proof.

Repeat this process for each $1 \leq i \leq n$, each time selecting a codeword that contains a neuron already existing in the realization. This provides us with a fully dimensional closed realization of C.

Thus, by Theorem 4.3, C is convex. □

Conjecture

Let C be a closed convex neural code on n neurons. Let $U = \{U_i\}_{i=1}^{n}$ in $\mathbb{R}^d$ be an arbitrary open convex cover of C. If filling in the boundary of each $U_i \in U$ will always create a set that can only be realized in $\mathbb{R}^{d-2}$ or below, then C is not open convex.

Conjecture

Let C be a closed convex neural code on n neurons. Let $U = \{U_i\}_{i=1}^n$ in $\mathbb{R}^d$ be an arbitrary open convex cover of C. If filling in the boundary of each $U_i \in U$ will always create a set that can only be realized in $\mathbb{R}^{d-2}$ or below, then C is not open convex.

Conjecture

Let C be a locally good neural code on n neurons. If C is not open convex, then any convex realization of C in $\mathbb{R}^d$ must contain a set that can only be realized in $\mathbb{R}^{d-2}$ or below.

Conjecture

Let C be a closed convex neural code on n neurons. Let $U = \{U_i\}_{i=1}^n$ in $\mathbb{R}^d$ be an arbitrary open convex cover of C. If filling in the boundary of each $U_i \in U$ will always create a set that can only be realized in $\mathbb{R}^{d-2}$ or below, then C is not open convex.

Conjecture

Let C be a locally good neural code on n neurons. If C is not open convex, then any convex realization of C in $\mathbb{R}^d$ must contain a set that can only be realized in $\mathbb{R}^{d-2}$ or below.

Conjecture

Let C be a locally good neural code on n neurons. If $n \leq 7$, then C must be either open or closed convex.

References

- Carina Curto, Elizabeth Gross, Jack Jeffries, Katherine Morrison, Mohamed Omar, Zvi Rosen, Anne Shiu, and Nora Youngs. What makes a neural code convex? *SIAM Journal on Applied Algebra and Geometry*, 1(1):222-238, 2017.
- Chad Giusti and Vladimir Itskov. A no-go theorem for one-layer feedforward networks. *Neural computation*, 26(11):2527-2540, 2014.
- Sarah Ayman Goldrup and Kaitlyn Phillipson. Classification of open and closed convex codes on five neurons, 2014.