On Space Folds by Neural Networks

Michal Lewandowski

16.10.2025

Co-authors

Hamid
(Meta)

Raphael
(SCCH)

Bernhard
(SCCH)

Bernhard
(JKU & SCCH)

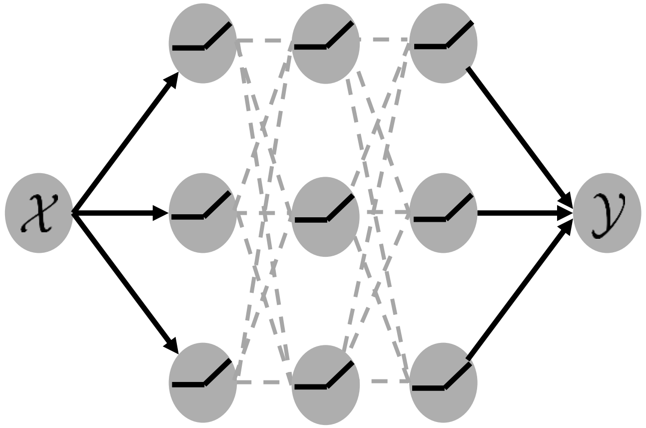
- **SCCH**: a Research Center in Hagenberg, Austria, focused on applied AI and software sciences
- **S3AI**: a research project focused on geometry of neural networks (2020-2024)

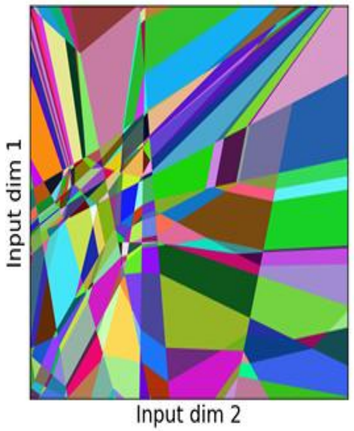
scch { }
software competence
center hagenberg

s3AI

Introduction

- **Focus**: Analysis of the **phenomena of folding** through a novel measure
- **How:** We focus on fully connected neural networks (MLPs)
- **Goal:** Gaining theoretical insights – not optimizing real-world performance
    - ✓ Universal approximation and exponential expressiveness
    - ✓ Piece-wise linear functions - inherit a geometrical interpretation



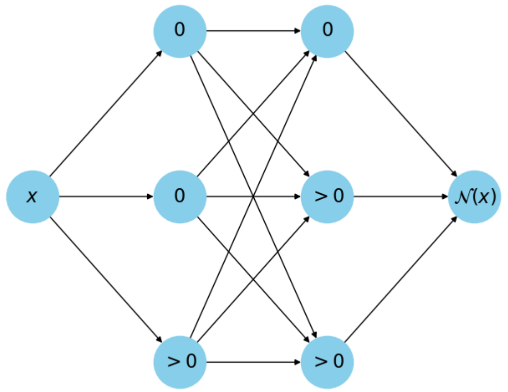


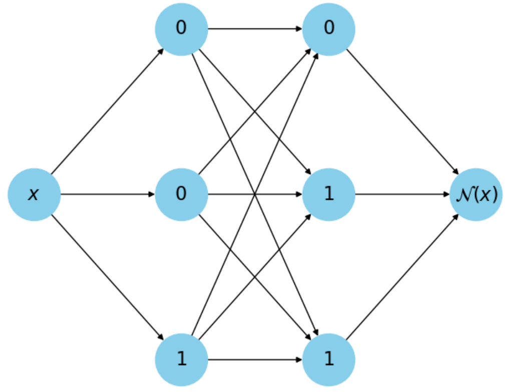
(Reproduced from [1])

[1] H. Boris, and D. Rolnick. (2019). *Deep relu networks have surprisingly few activation patterns*. NeurIPS.

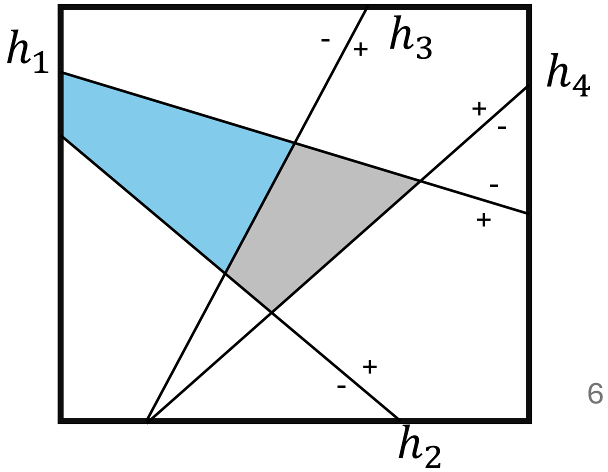
Hamming Activation Space

- A ReLU NN (denoted $\mathcal{N}$) is an alternating composition of the ReLU ($\sigma(x) \coloneqq \max(x, 0)$) and affine functions $g_i(\mathbf{x}) \coloneqq W_i\mathbf{x} + b_i$
- Fix $(W_i, b_i)_{i=1}^{L}$; push an input $\mathbf{x}$ through $\mathcal{N}$; at every layer we obtain a vector of activation values





- Activation pattern: $\pi_1 = (001011)$
- For any $\mathbf{x}_i \in \mathcal{X}$, there is an associated activation pattern π_i
- By construction, each (observable) π_i is a solution $\{\mathbf{x}: h_1(\mathbf{x}) \geq 0 \;\&\; \ldots \;\&\; h_4(\mathbf{x}) \geq 0\}$
- In geometry: *a polytope* [1], in ML: *a linear region* [2]

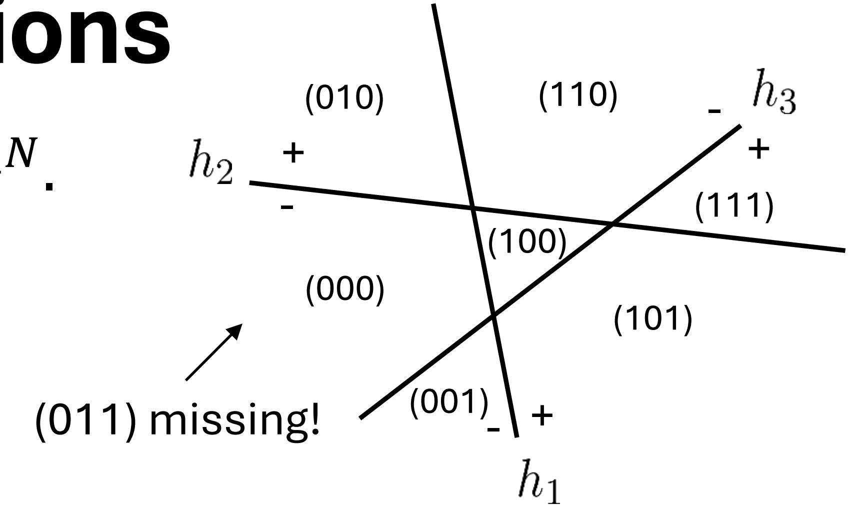


- Hamming distance d_H quantifies difference between $\pi_i, \pi_j \in \{0,1\}^N : d_H(\pi_i, \pi_j) \coloneqq |\{i: \pi_{i,k} \neq \pi_{j,k}\}|$
- $(\{0,1\}^N, d_H)$ is a metric space *(The Hamming Activation Space)*

[1] Ziegler, G. M. (2000). *Lectures on 0/1-polytopes*. Polytopes—combinatorics and computation.
[2] Montufar et al. (2014). *On the Number of Linear Regions of Deep Neural Networks*. NeurIPS.

Analysis of ReLU-Induced Tessellations

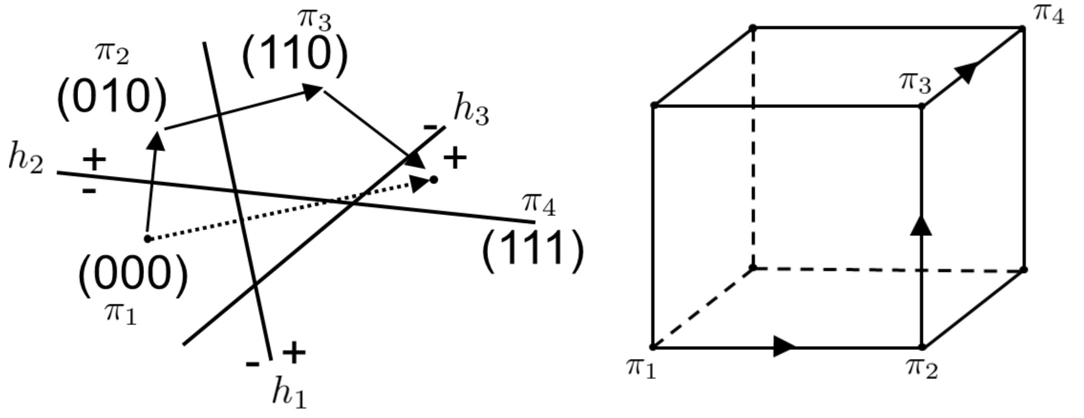
The Hamming activation space contains every π ∈ $\{0,1\}^N$.
BUT not every π is a pre-image of a linear region!



Impacts the definition of convexity in the Hamming space [1, 2]

$$(\mathcal{X}, d_E) \longrightarrow (\{0,1\}^N, d_H)$$

$\downarrow$ Convexity ✓ $\qquad$ $\downarrow$ Convexity ?

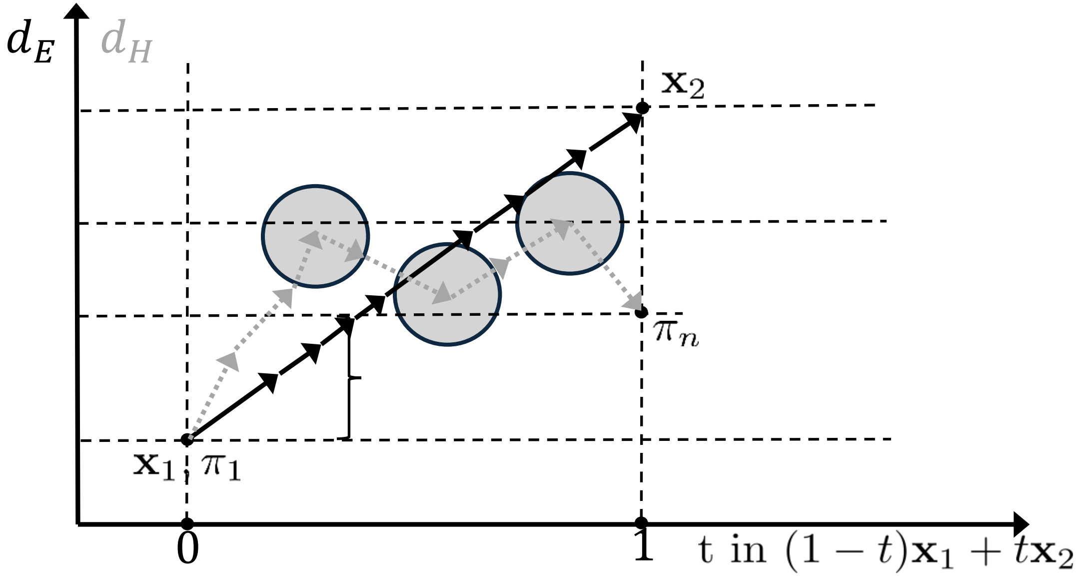


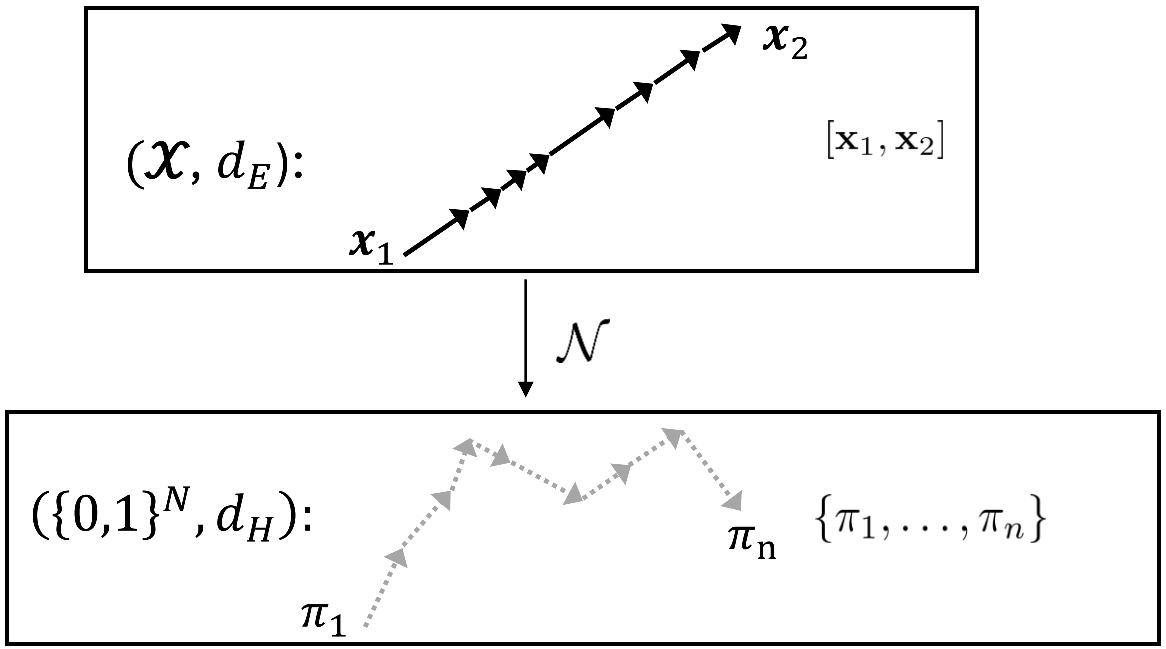
Definition 2. A subset S of the Hamming cube H^N is convex if, for every pair of points $\pi_i, \pi_j \in S$, all (observable) points on every shortest path between π_i, π_j are also in S.

Lemma 1: Given a space partition into regions $R_{\pi_1}, \dots, R_{\pi_r}$ labelled by $A = \{\pi_1, \dots, \pi_r\}$, a union $R = \bigcup_{\pi \in S \subset A} R_\pi$ is convex in R^n iff A is convex in H^N.

[1] B.A.Moser et al. (2022). *Tessellation Filtering ReLU Networks*. IJCAI.
[2] M.Lewandowski et al. (2025). *On Space Folds of ReLU Neural Networks*. TMLR.

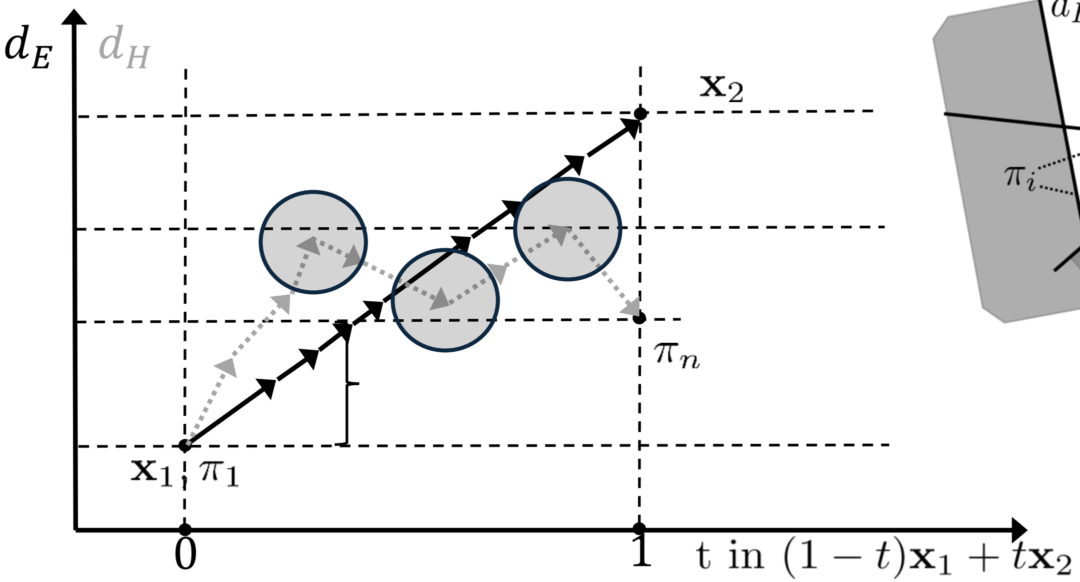
Mappings of Paths: Connection to Convexity

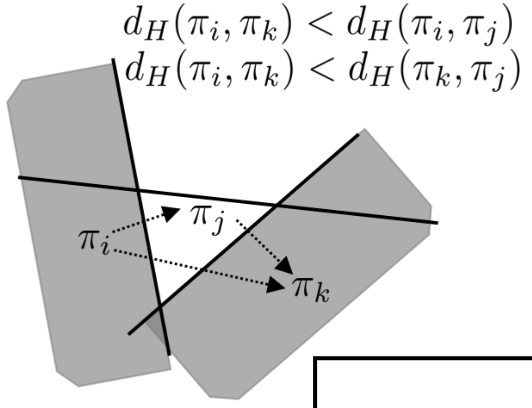


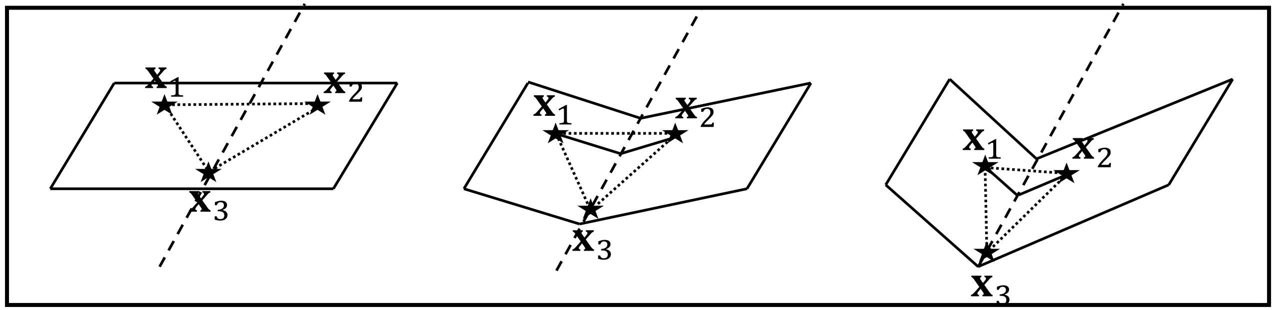


- $[\mathbf{x}_1, \mathbf{x}_2]$ - straight line in $(\mathcal{X}, d_E)$ – the smallest convex "set"
- We *map* $[\mathbf{x}_1, \mathbf{x}_2]$ from $(\mathcal{X}, d_E)$ to $(\{0,1\}^N, d_H)$ obtaining $\{\pi_1, \ldots, \pi_\mathrm{n}\}$
- But in $(\{0,1\}^N, d_H)$ not *every* shortest path γ (under d_H) between π_1 and π_n contains all $\{\pi_1, \ldots, \pi_n\}$, i.e., $\gamma \neq \{\pi_1, \ldots, \pi_n\} \Rightarrow \{\pi_1, \ldots, \pi_n\}$ is not convex (cf. Def 2)
- By investigating γ we can measure *deviations* from convexity between $\mathbf{x}_1$ and $\mathbf{x}_2$

Mappings of Paths: Connection to Folding







- On a path $\Gamma = \{\pi_1, \ldots, \pi_n\}$, we monitor two *range measures:*
  - r_1: Max change in d_H at each step i wrt π_1: $r_1(\Gamma) \coloneqq \max_i d_H(\pi_1, \pi_i)$
  - r_2: Total travelled distance on the hypercube: $r_2(\Gamma) \coloneqq \sum_{i=1}^{n-1} d_H(\pi_i, \pi_{i+1})$

Local Folding

$$\chi(\Gamma) \coloneqq 1 - \frac{r_1(\Gamma)}{r_2(\Gamma)} \in [0,1]$$

Global Folding

$$\Phi_{\mathcal{N}} \coloneqq median(\{\chi(\Gamma) : \chi(\Gamma) > 0\})$$

Alternatively, map $\sum_{i,j} \alpha_j \mathbf{x_i}$ through $\mathcal{N}$ for $\forall i, j$

Space Folding Measure – Properties

ICLR

Association for the Advancement of Artificial Intelligence

Edge cases:
- $\chi(\Gamma) = 0$ if $d_H(\pi_1, \pi_i)$ increases monotonically.
- $\chi(\Gamma) = 1$ for a looped path: $\max_i d_H(\pi_1, \pi_i) = c$, while $\sum_{i=1}^{n-1} d_H(\pi_i, \pi_{i+1}) \to \infty$.

The space folding measure has the following properties:

1. **[Stability]** Multiple steps in the same linear region do not influence its values.
2. **[Asymmetry]** The folding measure is sensitive to the direction of path traversal, i.e., $\chi(\Gamma) \neq \chi(-\Gamma)$, where $-\Gamma = \{\pi_n, \dots, \pi_1\}$.
3. **[Flatness Invariance]** $\chi(\Gamma) = 0$ if and only if $\chi(-\Gamma) = 0$.
4. **[Non-additivity]** Is neither sub- nor super-additive.

Let $\Gamma = \Gamma(\mathbf{x}_1, \mathbf{x}_2)$ be a path spanned between the edge points $\mathbf{x}_1, \mathbf{x}_2$.
Proposition. Let d_χ be a symmetrized space folding measure,

$$d_\chi(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{2}\Big(\chi\big(\Gamma(\mathbf{x}_1, \mathbf{x}_2)\big) + \chi\big(\Gamma(\mathbf{x}_2, \mathbf{x}_1)\big)\Big).$$

Then, d_χ is a pseudo-metric: **(1) Positivity**: from bounds on χ; **(2) Symmetry**: from construction; **(3) Triangle inequality**: from the triangle inequality of d_H

Algorithm & Complexity

Algorithm 1: Computation of the Space Folding Measure

Input: Two input samples $\mathbf{x}_1$, $\mathbf{x}_2$, the number of intermediate points n, the total number of hidden neurons N, cost of running the network in the inference mode $O(\mathtt{C})$

Output: Space Folding $\chi(\Gamma)$

Step 1: Linearly interpolate $\mathbf{x}_1$ and $\mathbf{x}_2$, sampling n points; `// Sampling Complexity:` $O(n)$

Step 2: For each sampled point:

begin

  Compute the binarisation; `// Binarization Complexity Per Point:` $O(\mathtt{C})$

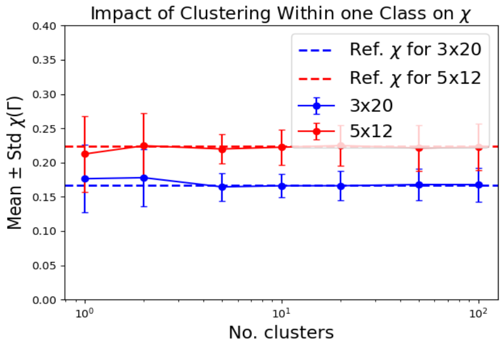
`// Total Binarization Complexity:` $O(n \cdot \mathtt{C})$

Step 3: Compute the maximal (from the starting point) and total Hamming distances between intermediate points; `// Computation of Range Measures Complexity:` $O(n \cdot N)$

return Space Folding $\chi(\Gamma)$; `// Total Algorithm Complexity:` $O(n \cdot (N + \mathtt{C}))$

Computational complexity: $O(n \cdot (N + Cost\ of\ Sample\ Propagation) \cdot |C_1| \cdot |C_2|)$

1. For a given the dataset, $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m}$
2. Cluster data samples $\mathbf{x}_i$ of the same label
3. Use clusters' centroids for computation of χ



Space Folding - Caveats

Association for the Advancement of Artificial Intelligence

- For an activation function f, we considered its pre-image thresholded at 0, i.e., $f^{(-1)}((-\infty, 0]) \to 0$ and $f^{(-1)}((0, \infty)) \to 1$
- Why thresholding at 0?
  - consider thresholds $a, b \in (-\infty, \infty)$ in the range of the activation function f such that $|a| < |b|$,
  - $\#a$-induced regions $\geq \#b$-induced regions $\Rightarrow$ a-induced folding measure $\geq b$-induced folding measure;
  - for ReLU $a = 0$ is "more informative" than any $b > 0$

Space Folding - Beyond ReLU

- Space Folding computation is based on a walk traversing linear regions
- The computation can be extended to a walk traversing equivalence classes [1,2]:
$$\mathbf{x_1} \sim_{\mathcal{N}} \mathbf{x_2} \Leftrightarrow d_H(\pi_1, \pi_2) = 0$$
- For ReLU NN, $[\mathbf{x}]_{\mathcal{N}} \coloneqq \{\mathbf{z} : \mathbf{z} \sim_{\mathcal{N}} \mathbf{x}\}$ correspond to linear regions
- For non-monotonous f equivalence classes may be topologically disconnected, but the construction applies as is

[1] N. Shepeleva, et al. (2020). *Relu code space: a basis for rating network quality besides accuracy*. ICLR W.
[2] M. Lewandowski, et al. (2025). The Space Between: On Folding, Symmetries and Sampling. ICLR W.

Summary

- Same output function yet different architectures?
  - Studied in [1] based on CantorNet [2]; higher folding values for less Kolmogorov-complex representation

The question:

Given a space folding value $\chi(\Gamma) = \tau \in [0, 1]$ for some path Γ, what can we learn?

- χ can be seen as a feature of the network
- χ is upper- and lower-bounded - a reference point across neural networks
- **Interpretation**:
  - Higher χ $\rightarrow$ indicates a more compact representation
  - Lower χ $\rightarrow$ indicates potential architecture improvements for the task

[1] M.Lewandowski et al. (2025). On Space Folds of Neural Networks. TMLR.
[2] M.Lewandowski et al. (2024). *CantorNet: A Sandbox For Testing Topological and Geometrical Complexity Measures*. NeurIPS Workshop.

Space Folding and Generalization

- Folding values increase with the depth of the network *conditioned on* high validation accuracy
- On CIFAR100, we observed a steady increase in global folding during the training process

Beyond fully connected networks

- Consider a **subset of layers** – different underlying tessellation
- **Skip Connections** – applies as is
- **Transformer** – fcnn after the attention head
- **Convolutional layers** – after convolution and pooling

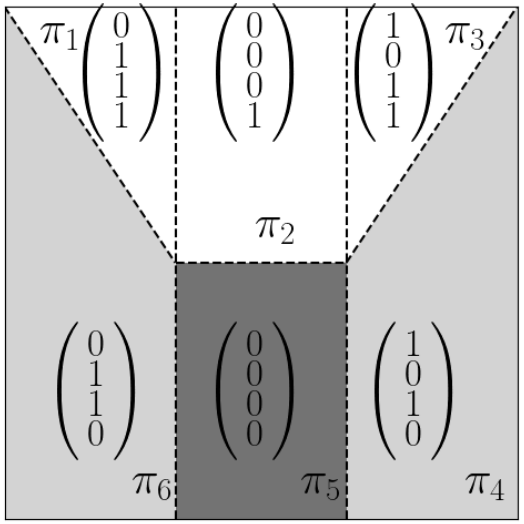
Low sensitivity to batch-norm, dropout

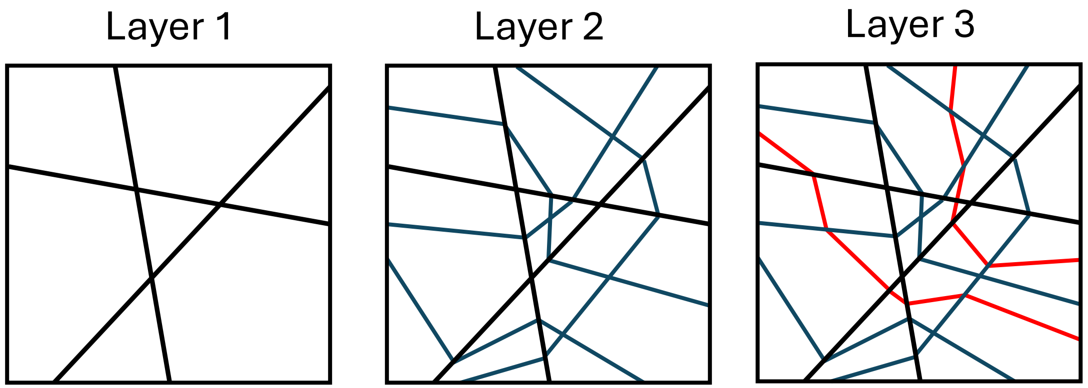
Convexity in the Hamming Activation Space

Example 1. Consider activation patterns $\pi_1 = (0111), \pi_2 = (0001), \pi_3 = (1011)$.
For a walk (1) $\pi_1 \to \pi_2$, (2) $\pi_2 \to \pi_3$, (3) $\pi_1 \to \pi_3$, there are intermediate, non-observable activation patterns π_{inter} that we traverse:

$$(1)\ \pi_{\text{inter}} = \{(0011), (0101)\}$$
$$(2)\ \pi_{\text{inter}} = \{(0011), (1001)\}$$
$$(3)\ \pi_{\text{inter}} = \{(0011), (1111)\}$$

Thus, the activation patterns $\{\pi_1, \pi_2, \pi_3\}$ form a convex set in the Hamming cube sense.





(Adapted from [1])

[1] M. Raighu et al. (2017). *On the expressive power of deep neural networks*. ICML.
[2] M.Lewandowski et al. (2024). *CantorNet: A Sandbox for Testing Topological and Geometrical Complexity Measures*. NeurIPS Workshop.

Interaction Coefficient and Adversarial Examples

χ is neither sub- nor super-additive - define *interaction coefficient*

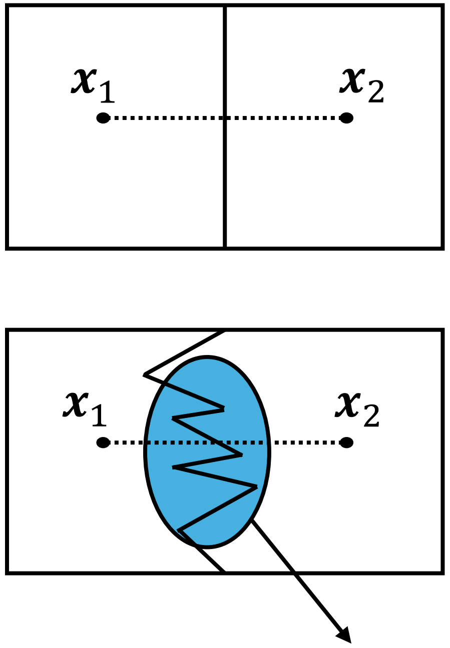
$$I(k) := |\chi(\Gamma_1 \oplus \Gamma_2) - \chi(\Gamma_1) - \chi(\Gamma_2)|,$$

where $k \in \mathrm{N}$ is the connecting index of paths Γ_1 and Γ_2 defined as

$$\Gamma_1 = \{\pi_1, \dots, \pi_k\}, \Gamma_2 = \{\pi_k, \dots, \pi_n\}, \Gamma_1 \oplus \Gamma_2 = \{\pi_1, \dots, \pi_n\}.$$

- I computed using digits 5 and 7 from the MNIST test set; $*$ denotes adversarial perturbation
- I is consistently higher for unperturbed samples

Intuition: folding and adversarial geometry

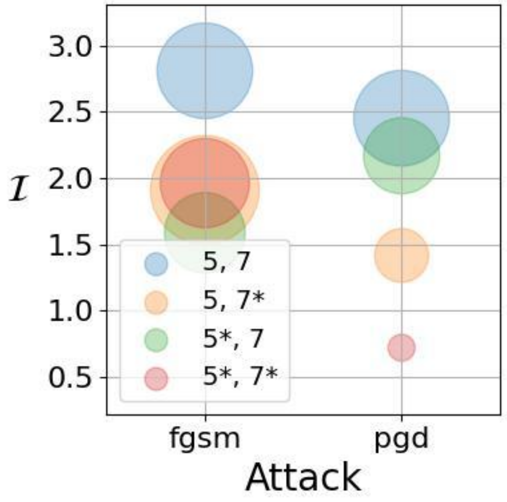


We can measure this!

Algorithm 1: $\mathcal{I}$ and Adversarial Attacks

Input: Dataset $\{(\mathbf{x}_i, y_i)\}_{i \in I}$
Output: Mean non-zero $\mathcal{I}$

Step 1: Adversarially perturb the input $\mathbf{x}$ obtaining $\mathbf{x}^*$.
Step 2: Assert that $\mathcal{N}(\mathbf{x}) \neq \mathcal{N}(\mathbf{x}^*)$.
Step 3: Compute $\mathcal{I}$ on a path spanned between $[\mathbf{x}, \mathbf{x}^*]$
(a) check the linear combination of $\mathbf{x}$ and $\mathbf{x}^*$ for varying connecting index k.
(b) store k s.t. $\mathcal{I}$ increases step-wise
return $\frac{1}{n-k+1} \sum_{i>k} \mathcal{I}_i$



Equivalence of Convexity Notions: Sketch of the Proof

Convexity in R_n $\Rightarrow$ Convexity in the Haming space

- Consider convex $R = \bigcup_{\pi \in A} R_\pi$ in R^n.
- Connectivity of A: take π_i, π_j and some points $P \in R_{\pi_i}; Q \in R_{\pi_j}$
- $R = \bigcup_{\pi \in A} R_\pi$ is convex $\Rightarrow$ $[P, Q]$ lies entirely within R.
  - Along $[P, Q]$ we cross $h_1, \dots, h_l$ flipping 1-bit in activation patterns at a time.
  - Sequence of flips forms a shortest path in the Hamming space from π_i to π_j.
  - If A wasn't convex, there would exist γ connecting π_i with π_j and leaving A.
  - But $[P, Q]$ is a straight line - it corresponds to the minimal sequence of bit flips.