Embeddings of Metric Spaces induced by Permutation Groups

Radu Balan

Department of Mathematics, CSCAMM and NWC
University of Maryland, College Park, MD

May 18, 2021
Codex Seminar

Acknowledgments

"This material is based upon work partially supported by the National Science Foundation under grant no. DMS-1816608 and LTS under grant H9823013D00560049. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation."

Joint works with:
Naveed Haghani (UMD,APL)
Maneesh Singh (Verisk)
Efstratios Tsukanis (UMD)

Overview

In this talk, we discuss Euclidean embeddings of metric spaces induced by actions of the permutation group $\mathcal{S}_n$ on a linear space V.

Let $\Pi \in \mathcal{S}_n$, $X \in \mathbb{R}^{n\times d}$ and $A = A^T \in \mathbb{R}^{n\times n}$. Family of actions:

1. $V = \mathbb{R}^{n\times d}$, $X \mapsto \Pi X$
2. $V = Sym(n)$, $A \mapsto \Pi A \Pi^T$
3. $V = Sym(n) \times \mathbb{R}^{n\times d}$, $(A, X) \mapsto (\Pi A \Pi^T, \Pi X)$

Problem: Construct (bi)Lipschitz embeddings of the metric space $\hat{V} = V/\sim$ of co-orbits, $\alpha : \hat{V} \to \mathbb{R}^m$.

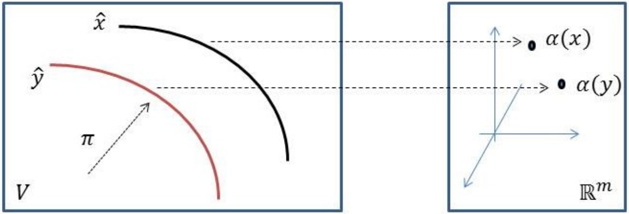


Table of Contents:

1. Motivation
2. Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n\times d}$
3. Polynomial Embeddings
4. Sorting based Embeddings
5. Towards Embeddings of $\hat{V}$ for $V = Sym(n)$
6. Numerical Examples

Table of Contents

1 Motivation

2 Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n\times d}$

3 Polynomial Embeddings

4 Sorting based Embeddings

5 Towards Embeddings of $\hat{V}$ for $V = Sym(n)$

6 Numerical Examples

Similarity of Matrices

Consider two symmetric matrices $A, B \in \mathrm{Sym}(n)$. When are they equivalent modulo an orthonormal change of coordinates?
Specificaly, is there an orthogonal matrix $U \in O(n)$ so that $B = UAU^T$?

An elementary derivation in linear algebra shows that $A \overset{O(n)}{\sim} B$ if and only if A and B have the same set of eigenvalues with exactly same multiplicities.

But what about other groups G? For instance what about the group of permutation matrices $\mathcal{S}_n$?
Find necessary and sufficient conditions so that $A \overset{\mathcal{S}_n}{\sim} B$.
Recall:

$$\mathcal{S}_n = \{P \in O(n) : P_{i,j} \in \{0,1\}\} = O(n) \cap \{W \in [0,1]^{n\times n} : W1 = 1, W^T 1 = 1\}$$

The Graph Isomorphism Problem

Consider two graphs $G = (\mathcal{V}, \mathcal{E})$ and $\tilde{G} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ with n nodes. The graph isomorphism problem is the computational problem of determining whether these graphs are identical after a relabeling of nodes.

If A and $\tilde{A}$ denote their adjacency matrices, these graphs are isomorphic if and only if $\tilde{A} = \Pi A \Pi^T$ for some permutation matrix $\Pi \in \mathcal{S}_n$.

Current state-of-the-art (Wikipedia): Babai (2015,2017) presented a quasi-polynomial algorithm with running time $2^{O((\log n)^c)}$, for some fixed $c > 0$. Helfgott (2017) claims that one can take $c = 3$.

Similar problem can be stated for weighted graphs: $A, \tilde{A} \in \mathrm{Sym}(n)$ with nonnegative entries, isomorphic if and only if $\tilde{A} = \Pi A \Pi^T$ for some $\Pi \in \mathcal{S}_n$.

Graph Alignment Problems

Consider two $n \times n$ symmetric matrices A, B. In the alignment problem for quadratic forms one seeks an orthogonal matrix $U \in O(n)$ that minimizes

$$\|UAU^T - B\|_F^2 := trace((UAU^T - B)^2) = \|A\|_F^2 + \|B\|_F^2 - 2trace(UAU^T B).$$

The solution is well-known and depends on the eigendecomposition of matrices A, B: if $A = U_1 D_1 U_1^T$, $B = U_2 D_2 U_2^T$ then

$$U_{opt} = U_2 U_1^T \quad , \quad \|U_{opt} A U_{opt}^T - B\|_F^2 = \sum_{k=1}^{n} |\lambda_k - \mu_k|^2,$$

where $D_1 = diag(\lambda_k)$ and $D_2 = diag(\mu_k)$ are diagonal matrices with eigenvalues ordered monotonically.

Quadratic Assignment Problem

The challenging case is when U is constrained to the permutation group as is the case in the *graph matching problem*. In this case, the optimization problem becomes

$$\min_{U\in\mathcal{S}_n} \|UAU^T - B\|_F$$

turns into a QAP:

$$\max_{U\in\mathcal{S}_n} trace(UAU^T B).$$

This is equivalent to computing the natural distance $d(\hat{A},\hat{B}) = \min_{P,Q\in\mathcal{S}_n} \|PAP^T - QBQ^T\|_F$ between the equivalence classes $\hat{A},\hat{B} \in \widehat{\mathrm{Sym}(n)}$ induced by the group action $\mathcal{S}_n \times \mathrm{Sym}(n) \to \mathrm{Sym}(n)$, $(\Pi, A) \mapsto \Pi A \Pi^T$.

Graph Learning Problems

Given a data graph (e.g., social network, transportation network, citation network, chemical network, protein network, biological networks):

- Graph adjacency or weight matrix, $A \in \mathbb{R}^{n\times n}$;
- Data matrix, $X \in \mathbb{R}^{n\times d}$, where each row corresponds to a feature vector per node.

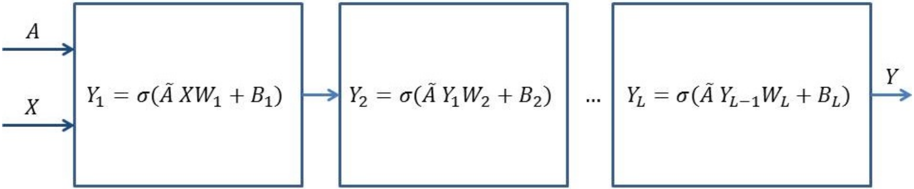
Contruct a map $f : (A, X) \to f(A, X)$ that performs:

1. classification: $f(A, X) \in \{1, 2, \cdots, c\}$
2. regression/prediction: $f(A, X) \in \mathbb{R}$.

Key observation: The outcome should be invariant to vertex permutation: $f(PAP^T, PX) = f(A, X)$, for every $P \in \mathcal{S}_n$.

Graph Convolutive Networks (GCN), Graph Neural Networks (GNN)

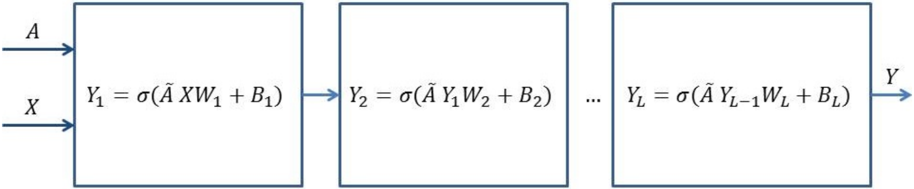
General architecture of a GCN/GNN



GCN (Kipf and Welling ('16)) choses $\tilde{A} = I + A$; GNN (Scarselli et.al. ('08), Bronstein et.al. ('16)) choses $\tilde{A} = p_l(A)$, a polynomial in adjacency matrix. L-layer GNN has parameters $(p_1, W_1, B_1, \cdots, p_L, W_L, B_L)$.

Graph Convolutive Networks (GCN), Graph Neural Networks (GNN)

General architecture of a GCN/GNN

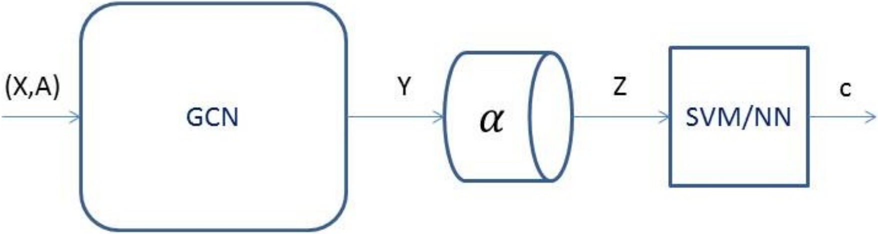


GCN (Kipf and Welling ('16)) choses $\tilde{A} = I + A$; GNN (Scarselli et.al. ('08), Bronstein et.al. ('16)) choses $\tilde{A} = p_l(A)$, a polynomial in adjacency matrix. L-layer GNN has parameters $(p_1, W_1, B_1, \cdots, p_L, W_L, B_L)$.

Note the *covariance (or, equivariance) property*: for any $P \in O(n)$ (including $\mathcal{S}_n$), if $(A, X) \mapsto (PAP^T, PX)$ and $B_i \mapsto PB_i$ then $Y \mapsto PY$.

Deep Learning with GCN

The approach for the two learning tasks (classification or regression) is based on the following scheme (see also Maron et.al. ('19)):



where α is a permutation invariant map (extractor), and SVM/NN is a single-layer or a deep neural network (Support Vector Machine or a Fully Connected Neural Network) trained on invariant representations.

The purpose of this talk is to analyze the α component.

Table of Contents

1. Motivation
2. Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n\times d}$
3. Polynomial Embeddings
4. Sorting based Embeddings
5. Towards Embeddings of $\hat{V}$ for $V = Sym(n)$
6. Numerical Examples

The metric space $\hat{V}$ when $V = \mathbb{R}^{n \times d}$

Recall the equivalence relation $\sim$ on $V = \mathbb{R}^{n \times d}$ induced by the group of permutation matrices $\mathcal{S}_n$ acting on V by left multiplication: for any $X, X' \in \mathbb{R}^{n \times d}$,

$$X \sim X' \quad \Leftrightarrow \quad X' = PX \ , \ \text{for some } P \in \mathcal{S}_n$$

Let $\widehat{\mathbb{R}^{n \times d}} = \mathbb{R}^{n \times d} / \sim$ be the quotient space endowed with the natural distance induced by Frobenius norm $\| \cdot \|_F$

$$d(\hat{X}_1, \hat{X}_2) = \min_{P \in S_n} \|X_1 - PX_2\|_F \quad , \quad \hat{X}_1, \hat{X}_2 \in \widehat{\mathbb{R}^{n \times d}}.$$

The metric space $\hat{V}$ when $V = \mathbb{R}^{n\times d}$

Recall the equivalence relation $\sim$ on $V = \mathbb{R}^{n\times d}$ induced by the group of permutation matrices $\mathcal{S}_n$ acting on V by left multiplication: for any $X, X' \in \mathbb{R}^{n\times d}$,

$$X \sim X' \quad \Leftrightarrow \quad X' = PX \ , \ \text{for some } P \in \mathcal{S}_n$$

Let $\widehat{\mathbb{R}^{n\times d}} = \mathbb{R}^{n\times d}/\sim$ be the quotient space endowed with the natural distance induced by Frobenius norm $\|\cdot\|_F$

$$d(\hat{X}_1, \hat{X}_2) = \min_{P\in S_n} \|X_1 - PX_2\|_F \quad , \quad \hat{X}_1, \hat{X}_2 \in \widehat{\mathbb{R}^{n\times d}}.$$

The computation of the minimum distance is performed by solving the Linear Assignment Problem (LAP) whose convex relaxation is exact:

$$\max_{P\in\mathcal{S}_n} trace(PX_2X_1^T) = \max_{W\in DS(n)} trace(WX_2X_1^T)$$

where $DS(n) = \{W \in [0,1]^{n\times n} : W1 = 1, W^T1 = 1\}$ is the convex set of doubly stochastic matrices.

The embedding problem

Problem 1: Construct a Lipschitz embedding $\hat{\alpha} : \widehat{\mathbb{R}^{n\times d}} \to \mathbb{R}^m$, i.e., an integer $m = m(n,d)$, a map $\alpha : \mathbb{R}^{n\times d} \to \mathbb{R}^m$ and a constant $L = L(\alpha) > 0$ so that for any $X, X' \in \mathbb{R}^{n\times d}$,

1. If $X \sim X'$ then $\alpha(X) = \alpha(X')$.
2. If $\alpha(X) = \alpha(X')$ then $X \sim X'$.
3. $\|\alpha(X) - \alpha(X')\|_2 \leq L \cdot d(\hat{X}, \hat{X}') = L \min_{P\in\mathcal{S}_n} \|X - PX'\|_F$.

Problem 2: Construct a bi-Lipschitz embedding, i.e., in addition to conditions 1-3 α should satisfy also

4. $\exists a > 0\ \forall X, X' \in \mathbb{R}^{n\times d}$, $a \cdot d(\hat{X}, \hat{X}') \leq \|\alpha(X) - \alpha(X')\|_2$.

The Universal Embedding

Consider the map

$$\mu : \widehat{\mathbb{R}^{n \times d}} \to \mathcal{P}(\mathbb{R}^d) \quad , \quad \mu(X)(x) = \frac{1}{n} \sum_{k=1}^{n} \delta(x - x_k)$$

where $\mathcal{P}(\mathbb{R}^d)$ denotes the convex set of probability measures over $\mathbb{R}^d$, and δ denotes the Dirac measure.
Clearly $\mu(X') = \mu(X)$ iff $X' = PX$ for some $P \in \mathcal{S}_n$.

Main drawback: $\mathcal{P}(\mathbb{R}^d)$ is infinite dimensional!

Finite Dimensional Embeddings

Architectures

Two classes of extractors [Zaheer et.al.17' -'Deep Sets']:

1. Pooling Map – based on Max pooling
2. Readout Map – based on Sum pooling

Finite Dimensional Embeddings

Architectures

Two classes of extractors [Zaheer et.al.17' -'Deep Sets']:

1. Pooling Map – based on Max pooling
2. Readout Map – based on Sum pooling

Intuition in the case $d = 1$:

Max pooling:

$$\downarrow: \mathbb{R}^n \to \mathbb{R}^n \quad , \quad \downarrow(x) = x^{\downarrow} := (x_{\pi(k)})_{k=1}^n \ , \ x_{\pi(1)} \geq x_{\pi(2)} \geq \cdots \geq x_{\pi(n)}$$

Finite Dimensional Embeddings

Architectures

Two classes of extractors [Zaheer et.al.17' -'Deep Sets']:

1. Pooling Map – based on Max pooling
2. Readout Map – based on Sum pooling

Intuition in the case $d = 1$:

Max pooling:

$$\downarrow: \mathbb{R}^n \to \mathbb{R}^n \quad , \quad \downarrow(x) = x^{\downarrow} := (x_{\pi(k)})_{k=1}^n \;,\; x_{\pi(1)} \geq x_{\pi(2)} \geq \cdots \geq x_{\pi(n)}$$

Sum pooling:

$$\sigma : \mathbb{R}^n \to \mathbb{R}^n \quad , \quad \sigma(x) = (y_k)_{k=1}^n \;,\; y_k = \sum_{j=1}^n \nu(a_k, x_j)$$

where kernel $\nu : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, e.g. $\nu(a,t) = e^{-(a-t)^2}$, or $\nu(a = k, t) = t^k$.

Pooling Mapping Approach

Fix a matrix $R \in \mathbb{R}^{d\times D}$. Consider the map:

$$\Lambda : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times D} \equiv \mathbb{R}^{nD} \quad , \quad \Lambda(X) = \downarrow(XR)$$

where $\downarrow$ acts columnwise (reorders monotonically decreasing each column). Since $\Lambda(\Pi X) = \Lambda(X)$, then $\Lambda : \widehat{\mathbb{R}^{n\times d}} \to \mathbb{R}^{n\times D}$. Let $R = [r_1, \cdots, r_D]$.

Theorem

The map Λ is Lipschitz with Lipschitz constant $L = \sum_{k=1}^{d} \|r_k\|_2$, i.e.

$$\| \downarrow(XR) - \downarrow(YR)\|_2 \leq L \min_{\Pi \in \mathcal{S}_n} \|X - \Pi Y\|_2$$

Proof For any $\Pi \in \mathcal{S}_n$,

$$\|\downarrow(XR) - \downarrow(YR)\| \leq \sum_{k=1}^{d} \|\downarrow(Xr_k) - \downarrow(Yr_k)\| \leq \sum_{k=1}^{d} \|Xr_k - \Pi Y r_k\| \leq \sum_{k=1}^{d} \|r_k\|_2 \|X - \Pi Y\|$$

Take the minimum over Π and the result follows.

Readout Mapping Approach

Kernel Sampling

Consider:

$$\Phi : \mathbb{R}^{n\times d} \to \mathbb{R}^m \quad , \quad (\Phi(X))_j = \sum_{k=1}^{n} \nu(a_j, x_k) \text{ or } (\Phi(X))_j = \prod_{k=1}^{n} \nu(a_j, x_k)$$

where $\nu : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a kernel, and $x_1, \cdots, x_n$ denote the rows of matrix X.

Known solutions: For $m = \infty$, the measure-valued representation is globally injective and stable. For $m < \infty$, one can construct Lipschitz embeddings of compacts.

The challenge is to construct ν so that: (1) the map is defined over entire metric space; (2) the map is bi-Lipschitz.

Readout Mapping Approach

The RKHS Point of View

Remark: If the kernel ν defines a Reproducing Kernel Hilberts Spaces (RKHSs), and a spectral theorem is applicable (e.g., Mercer's theorem) then:

$$(\Phi(X))_j = \sum_{p\geq 1} \sigma_p f_p(a_j) g_p(X)$$

This result suggests a tow-stage embedding:

$$X \mapsto \xi = (g_p(X))_{p\geq 1} \mapsto \Phi(X) = A\xi.$$

Special case: when $g_p(X)$ are monomials, then $\Phi(X)$ is a family of polynomials.

Table of Contents

1. Motivation
2. Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n \times d}$
3. Polynomial Embeddings
4. Sorting based Embeddings
5. Towards Embeddings of $\hat{V}$ for $V = Sym(n)$
6. Numerical Examples

Polynomial Expansions - Quadratics

In the case $d = 1$ recall Vieta's formulas, Newton-Girard identities

$$P(X) = \prod_{k=1}^{N}(X - x_k) \leftrightarrow (\sum_k x_k, \sum_k x_k^2, ..., \sum_k x_k^n)$$

Polynomial Expansions - Quadratics

In the case $d = 1$ recall Vieta's formulas, Newton-Girard identities

$$P(X) = \prod_{k=1}^{N}(X - x_k) \leftrightarrow (\sum_k x_k, \sum_k x_k^2, \ldots, \sum_k x_k^n)$$

For $d > 1$, consider the quadratic d-variate polynomial:

$$\begin{aligned} P(Z_1, \cdots, Z_d) &= \prod_{k=1}^{n}\left((Z_1 - x_{k,1})^2 + \cdots + (Z_d - x_{k,d})^2\right) \\ &= \sum_{p_1,\ldots,p_d=0}^{2n} a_{p_1,\ldots,p_d} Z_1^{p_1} \cdots Z_d^{p_d} \end{aligned}$$

Encoding complexity:

$$m = \binom{2n+d}{d} \sim (2n)^d.$$

Polynomial Expansions - Quadratics (2)

A more careful analysis of $P(Z_1, ..., Z_d)$ reveals a form:

$$P(Z_1, ..., Z_d) = t^n + Q_1(Z_1, ..., Z_d)t^{n-1} + \cdots + Q_{n-1}(Z_1, ..., Z_d)t + Q_n(Z_1, ..., Z_d)$$

where $t = Z_1^2 + \cdots + Z_d^2$ and each $Q_k(Z_1, ..., Z_d) \in \mathbb{R}_k[Z_1, ..., Z_d]$ is a (non-homogeneous) polynomial of degree k. Hence one needs to encode:

$$m = \binom{d+1}{1} + \binom{d+2}{2} + \cdots + \binom{d+n}{n} = \binom{d+n+1}{n} - 1$$

number of coefficients.

A significant drawback: Inversion is numerically unstable and embedding is not Lipschitz.

Readout Mapping Approach

Polynomial Expansion - Linear Forms

A stable embedding can be constructed as follows (see also Gobels' algorithm (1996) or [Derksen, Kemper '02]).
Consider the n linear forms $\lambda_k(Z_1, ..., Z_d) = x_{k,1}Z_1 + \cdots x_{k,d}Z_d$. Construct the polynomial in variable t with coefficients in $\mathbb{R}[Z_1, ..., Z_d]$:

$$P(t) = \prod_{k=1}^{n}(t-\lambda_k(Z_1, ..., Z_d)) = t^n - e_1(Z_1, .., Z_d)t^{n-1} + \cdots (-1)^n e_n(Z_1, ..., Z_d)$$

$$= t^n + \sum_{\substack{p_0, p_1, \cdots, p_d \geq 0 \\ p_0 + p_1 + \cdots + p_d = n \;,\; p_0 < n}} c_{p_0,p_1,\cdots,p_d} t^{p_0} Z_1^{p_1} \cdots Z_d^{p_d}$$

The elementary symmetric polynomials $(e_1, ..., e_n)$ are in 1-1 correspondence (Newton-Girard theorem) with the moments: $\mu_p = \sum_{k=1}^{n} \lambda_k^p(Z_1, ..., Z_d)$, $1 \leq p \leq n$.

Polynomial Expansions - Linear Forms (2)

Each μ_p is a homogeneous polynomial of degree p in d variables. Hence to encode each of them one needs $\left(\begin{array}{c} d+p-1 \\ p \end{array}\right)$ coefficients. Hence the embedding dimension is

$$m_0 = \left(\begin{array}{c} d \\ 1 \end{array}\right) + \left(\begin{array}{c} d+1 \\ 2 \end{array}\right) + \cdots + \left(\begin{array}{c} d+n-1 \\ n \end{array}\right) = \left(\begin{array}{c} d+n \\ n \end{array}\right) - 1$$

Polynomial Expansions - Linear Forms (2)

Each μ_p is a homogeneous polynomial of degree p in d variables. Hence to encode each of them one needs $\binom{d+p-1}{p}$ coefficients. Hence the embedding dimension is

$$m_0 = \binom{d}{1} + \binom{d+1}{2} + \cdots + \binom{d+n-1}{n} = \binom{d+n}{n} - 1$$

The map $\alpha_0 : \mathbb{R}^{n\times d} \to \mathbb{R}^{m_0}$, $X \mapsto (c_{p_0,p_1,\cdots,p_d})_{p_0,p_1,\cdots,p_d}$ is injective modulo $\mathcal{S}_n$ but it is not Lipschitz. However a simple modification as suggested by Cahill et.al. ('19) makes it Lipschitz.

Polynomial Lipschitz embedding

Denote by L_0 the Lipschitz constant of α_0 when restricted to the closed unit ball $B_1(\mathbb{R}^{n\times d}) : \{X \in \mathbb{R}^{n\times d}\ ,\ \|X\| \leq 1\}$ of $\mathbb{R}^{n\times d}$, i.e. $\|\alpha_0(X) - \alpha_0(Y)\| \leq L_0\|X - Y\|$ for any $X, Y \in \mathbb{R}^{n\times d}$ with $\|X\|, \|Y\| \leq 1$. Let $\varphi_0 : \mathbb{R} \to [0,1]$, $\varphi_0(x) = min(1, \frac{1}{x})$ be a Lipschitz monotone decreasing function with Lipschitz constant 1.

Theorem

The map:

$$\alpha_1 : \mathbb{R}^{n\times d} \to \mathbb{R}^m \ ,\ \ \alpha_1(X) = \left(\begin{array}{c} \alpha_0\Big(\varphi_0(\|X\|)X\Big) \\ \|X\| \end{array} \right),$$

with $m = \left(\begin{array}{c} n+d \\ d \end{array} \right) = m_0 + 1$ *lifts to an injective and globally Lipschitz map* $\hat{\alpha}_1 : \widehat{\mathbb{R}^{n\times d}} \to \mathbb{R}^m$ *with Lipschitz constant* $Lip(\hat{\alpha}_1) \leq \sqrt{1 + L_0^2}$.

Minimality

For $d = 1$, $m = n$ which is minimal.

For $d = 2$, $m = \frac{n^2+3n}{2}$. Is this minimal?

Algebraic Embedding

Encoding using Complex Roots

Idea: Consider the case $d = 2$. Then each $x_1, \cdots, x_n \in \mathbb{R}^2$ can be replaced by n complex numbers $z_1, \cdots, z_n \in \mathbb{C}$, $z_k = x_{k,1} + i x_{k,2}$.
Consider the complex polynomial:

$$Q(z) = \prod_{k=1}^{n} (z - z_k) = z^n + \sum_{k=1}^{n} \sigma_k z^{n-k}$$

which requires n complex numbers, or $2n$ real numbers.

Algebraic Embedding

Encoding using Complex Roots

Idea: Consider the case $d = 2$. Then each $x_1, \cdots, x_n \in \mathbb{R}^2$ can be replaced by n complex numbers $z_1, \cdots, z_n \in \mathbb{C}$, $z_k = x_{k,1} + i x_{k,2}$.
Consider the complex polynomial:

$$Q(z) = \prod_{k=1}^{n} (z - z_k) = z^n + \sum_{k=1}^{n} \sigma_k z^{n-k}$$

which requires n complex numbers, or $2n$ real numbers.

Open problem: Can this construction be extended to $d \geq 3$?
Remark: A drawback of polynomial (algebraic) embeddings: [Cahill'19] showed that polynomial embeddings of translation invariant spaces cannot be bi-Lipschitz.

Table of Contents

1. Motivation
2. Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n \times d}$
3. Polynomial Embeddings
4. Sorting based Embeddings
5. Towards Embeddings of $\hat{V}$ for $V = Sym(n)$
6. Numerical Examples

The Embedding Problem

Notations

Recall the equivalence relation, for $X, Y \in \mathbb{R}^{n\times d}$,

$$X \sim Y \quad \Leftrightarrow \quad \exists \Pi \in \mathcal{S}_n \ , \ \ Y = \Pi X$$

that induces a quotient space $\widehat{\mathbb{R}^{n\times d}} = \mathbb{R}^{n\times d} / \sim$ and the natural distance

$$d : \widehat{\mathbb{R}^{n\times d}} \times \widehat{\mathbb{R}^{n\times d}} \to \mathbb{R} \quad , \quad d(X,Y) = \min_{\Pi \in \mathcal{S}_n} \|X - \Pi Y\|_F$$

In the following we look for an Euclidean embedding of the form

$$\alpha : \widehat{\mathbb{R}^{n\times d}} \to \mathbb{R}^{n\times D} \quad , \quad \alpha(X) = \left[\ \downarrow (X) \quad , \quad \downarrow (XA)\ \right]$$

where $\downarrow (\cdot)$ sorts decreasingly each column of $\cdot$, independently.
The matrix $R = [I_d \ \ A] \in \mathbb{R}^{d\times D}$ is called the *key* of encoder α.

The Embedding Problem

Notations (2)

Definition

Fix $X \in \mathbb{R}^{n\times d}$. A matrix $A \in \mathbb{R}^{d\times D}$ is called admissible *for X if $\alpha^{-1}(\alpha(X)) = \hat{X}$. In other words, if $Y \in \mathbb{R}^{n\times d}$ so that $\downarrow(XA) = \downarrow(YA)$ then there is $\Pi \in \mathcal{S}_n$ sot that $Y = \Pi X$.*

We denote by $\mathcal{A}_{d,D}(X)$ (or $\mathcal{A}(X)$) the set of admissible keys for X.

Definition

Fix $A \in \mathbb{R}^{d\times D}$. A data matrix $X \in \mathbb{R}^{n\times d}$ is said separated by A *if $A \in \mathcal{A}(X)$.*

We let $\mathcal{S}(A)$ denote the set of data matrices separated by A.
A key A is said *universal* if $\mathcal{S}(A) = \mathbb{R}^{n\times d}$.
The Problem: Design universal keys.

Max pooling is isometric embedding when $d = 1$

Proposition

In the case $d = 1$, $\downarrow: \widehat{\mathbb{R}^n} \to \mathbb{R}^n$, $\hat{x} \mapsto \downarrow(x)$ is an isometric embedding:

$$\| \downarrow(x) - \downarrow(y)\| = \min_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| \;, \quad \text{for all} \quad x, y \in \mathbb{R}^n.$$

Proof

Claim is equivalent to: $\min_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| = \|x^{\downarrow} - y^{\downarrow}\|$.
First note:

$$\min_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| = \min_{\Pi \in \mathcal{S}_n} \|x^{\downarrow} - \Pi y^{\downarrow}\| \leq \|x^{\downarrow} - y^{\downarrow}\|$$

Hence $\downarrow$ is Lipschitz with constant 1.

Max pooling is isometric embedding when $d = 1$

Proposition

In the case $d = 1$, $\downarrow: \widehat{\mathbb{R}^n} \to \mathbb{R}^n$, $\hat{x} \mapsto \downarrow(x)$ is an isometric embedding:

$$\| \downarrow(x) - \downarrow(y) \| = \min_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| \ , \quad \text{for all} \quad x, y \in \mathbb{R}^n.$$

Proof

Claim is equivalent to: $\min_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| = \|x^{\downarrow} - y^{\downarrow}\|$.

First note:

$$\min_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| = \min_{\Pi \in \mathcal{S}_n} \|x^{\downarrow} - \Pi y^{\downarrow}\| \leq \|x^{\downarrow} - y^{\downarrow}\|$$

Hence $\downarrow$ is Lipschitz with constant 1.

WLOG: Assume $x = x^{\downarrow}$, $y = y^{\downarrow}$. Then

$$argmin_{\Pi \in \mathcal{S}_n} \|x - \Pi y\| = argmin_{\Pi \in \mathcal{S}_n} \|x - x_n \cdot 1 - \Pi(y - y_n \cdot 1)\|$$

Therefore assume $x_n = y_n = 0$ and $x, y \geq 0$. The conclusion follows by induction over n.

Genericity Results for $d \geq 2$

Admissible keys

Theorem

Let $X \in \mathbb{R}^{n\times d}$. For any $D \geq d+1$ the set $\mathcal{A}_{d,D}(X)$ of admissible keys for X is dense in $\mathbb{R}^{d\times D}$ with respect to Euclidean topology, and it is generic with respect to Zariski topology. In particular, $\mathbb{R}^{d\times D} \setminus \mathcal{A}_{d,D}(X)$ has Lebesgue measure 0, i.e., almost every key is admissible for X.

Proof

It is sufficient to consider the case $D = d+1$. Also, it is sufficient to analyze the case $A = [I_d \;\; b]$ and to show that a generic $b \in \mathbb{R}^d$ defines an admissible key. The vector $b \in \mathbb{R}^d$ does **not** define an admissible key if there are $\Xi, \Pi_1, \cdots, \Pi_d \in S_n$ so that for $Y = [\Pi_1 x_1, \cdots, \Pi_d x_d]$,

$$Yb = \Xi X b \;\; \text{but} \;\; Y - \Pi X \neq 0 \;, \;\; \forall \Pi \in \mathcal{S}_n$$

Define the linear operator

Genericity Results for $d \geq 2$

Admissible keys

Proof - cont'd

Let

$$\mathcal{P} = \left\{(\Pi_1, \cdots, \Pi_d) \in (\mathcal{S}_n)^d \ \ \forall \Pi \in \mathcal{S}_n, \exists k \in [d] \ s.t. \ (\Pi - \Pi_k)x_k \neq 0\right\}$$

Then

$$\{b \in \mathbb{R}^d : [I_d \ b] \text{ not admissible for } X\} = \bigcup_{(\Xi;\Pi_1,\cdots,\Pi_d)\in\mathcal{S}_n\times\mathcal{P}} \ker(B(\Xi;\Pi_1,\cdots,\Pi$$

It is now sufficient to show that each null space has dimension less than d. Indeed, the alternative would mean $B(\Xi;\Pi_1,\cdots,\Pi_d) = 0$ but this would imply $(\Pi_1,\cdots,\Pi_d) \notin \mathcal{P}$. $\square$

Non-Universality of vector keys

Insufficiency of a single vector key

The following is a no-go result, which shows that there is no universal single vector key for data matrices tall enough.

Proposition

If $d \geq 2$ and $n \geq 3$,

$$\bigcup_{X\in\mathbb{R}^{n\times d}} \{b \in \mathbb{R}^d : \ A = [I_d \ \ b] \text{ not admissible for} X\} = \mathbb{R}^d.$$

Consequently,

$$\bigcap_{X\in\mathbb{R}^{n\times d}} \mathcal{A}_{d,d+1}(X) = \emptyset.$$

On the other hand, for $n = 2$, $d = 2$, any vector $b \in \mathbb{R}^2$ with $b_1 b_2 \neq 0$ defines a universal key $A = [I_2 \ \ b]$.

Non-Universality of vector keys

Insufficiency of a single vector key - cont'd

Proof

To show the result, it is sufficient to consider a counterexample for $n = 3$, $d = 2$, with key $b = [1, 1]^T$.

$$X = \begin{bmatrix} 1 & -1 \\ -1 & 0 \\ 0 & 1 \end{bmatrix} \quad , \quad Y = \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix}$$

Then $Xb = [0, -1, 1]^T$ and $Yb = [1, 0, -1]^T$, yet $X \not\sim Y$. Thus $[I_2 \;\; b]$ is not admissible for X.

Then note if $a \in \mathbb{R}^d$ so that $[I_d \;\; a]$ is admissible for X then for any $P \in S_d$ and L an invertible $d \times d$ diagonal matrix, $L^{-1}P^TA \in \mathcal{A}_{d,1}(XPL)$. This shows how for any $b \in \mathbb{R}^2$, one can construct $X \in \mathbb{R}^{3\times 2}$ so that $b \notin \mathcal{A}_{2,1}(X)$.

For $n > 3$ or $d > 2$, proof follows by embedding this example.

Genericity Results for $d \geq 2$

Admissible Data Matrices

Theorem

Assume $a \in \mathbb{R}^d$ is a vector with non-vanishing entries, i.e., $a_1 a_2 \cdots a_d \neq 0$. Then for any $n \geq 1$, $\mathcal{S}([I_d \;\; a])$ is dense in $\mathbb{R}^{n \times d}$ and includes an open dense set with respect to Zariski topology. In particular, $\mathbb{R}^{n \times d} \setminus \mathcal{S}([I_d \;\; a])$ has Lebesgue measure 0, i.e., almost every data matrix X is separated by the vector key a.

Genericity Results for $d \geq 2$

Admissible Data Matrices

Theorem

Assume $a \in \mathbb{R}^d$ is a vector with non-vanishing entries, i.e., $a_1 a_2 \cdots a_d \neq 0$. Then for any $n \geq 1$, $\mathcal{S}([I_d\ \ a])$ is dense in $\mathbb{R}^{n\times d}$ and includes an open dense set with respect to Zariski topology. In particular, $\mathbb{R}^{n\times d} \setminus \mathcal{S}([I_d\ \ a])$ has Lebesgue measure 0, i.e., almost every data matrix X is separated by the vector key a.

Corollary

Assume $A \in \mathbb{R}^{d\times(D-d)}$ is a matrix such that at least one column has non-vanishing entries. Then for any $n \geq 1$, $\mathcal{S}([I_d\ \ A])$ is dense in $\mathbb{R}^{n\times d}$ and is generic with respect to Zariski topology. In particular, $\mathbb{R}^{n\times d} \setminus \mathcal{S}([I_d\ \ A])$ has Lebesgue measure 0, i.e., almost every data matrix X is separated by the matrix key $[I_d\ \ A]$.

Proof that $\mathcal{S}([I_d\ A])$ is generic

The case $D > d$

Assume $A \in \mathbb{R}^{d\times(D-d)}$ satisfies $A_{1,k}A_{2,k}\cdots A_{d,k} \neq 0$ for some $k \in [D-d]$. The set of non-separated data matrices $X \in \mathbb{R}^{n\times d}$ (i.e., the complement of $\mathcal{S}([I_d\ A])$) factors as follows:

$$\mathbb{R}^{n\times d}\setminus\mathcal{S}([I_d\ A]) = \bigcup_{(\Xi_1,\cdots,\Xi_{D-d};\Pi_1,\cdots,\Pi_d)\in(\mathcal{S}_n)^D} \Big(\ker L(\Xi_1,\cdots,\Xi_{D-d};\Pi_1,\cdots,\Pi_d;A$$

$$\setminus \bigcup_{\Pi\in\mathcal{S}_n} \ker\ M(\Pi,\Pi_1,\cdots,\Pi_d)\Big) \quad (*)$$

where, with $A = [a_1,\cdots,a_{D-d}]$, $X = [x_1,\cdots,x_d]$:

$$L(\Xi_1,\cdots,\Xi_{D-d};\Pi_1,\cdots,\Pi_d;A):\mathbb{R}^{n\times d}\to\mathbb{R}^{n\times D-d} \ , \ (L((...)X)_k=[(\Xi_k-\Pi_1)x_1,\cdots,(\Xi_k-\Pi_d)x_d]a_k \ , \ k\in[D-$$

$$M(\Pi,\Pi_1,\cdots,\Pi_d):\mathbb{R}^{n\times d}\to\mathbb{R}^{n\times d} \ , \ M(\Pi,\Pi_1,\cdots,\Pi_d)X=[(\Pi-\Pi_1)x_1,\cdots,(\Pi-\Pi_d)x_d]$$

Proof that $\mathcal{S}(A)$ is generic

cont'd

1. The outer union can be reduced by noting that on the "diagonal" Δ,

$$\Delta = \{(\Xi_1,\cdots,\Xi_{D-d};\Pi_1,\cdots,\Pi_d)\in(\mathcal{S}_n)^D \quad , \quad \Pi_1=\Pi_2=\cdots=\Pi_d\}$$

$$M(\Pi_1,\Pi_1,\cdots,\Pi_d)=0 \rightarrow \bigcup_{\Pi\in\mathcal{S}_n} \ker\, M(\Pi,\Pi_1,\cdots,\Pi_d)=\mathbb{R}^{n\times d}$$

2. If $(\Xi_1,\cdots,\Xi_{D-d};\Pi_1,\cdots,\Pi_d)\in(\mathcal{S}_n)^D\setminus\Delta$ then for every $k\in[D-d]$ there is $j\in[d]$ such that $\Xi_k-\Pi_j\neq 0$. In particular choose the k column of A that is non-vanishing. Let $x_j\in\mathbb{R}^n$ so that $(\Xi_k-\Pi_j)x_j\neq 0$. Consider the matrix $X=[0,\cdots,0,x_j,0,\cdots,0]$ where x_j is the only non identically 0 column. Claim: $X\notin \ker\, L(\Xi_1,...,\Pi_d;A)$. Indeed, the resulting k column of $L()X$ is $A_{j,k}(\Xi_k-\Pi_j)x_j\neq 0$. It follows that

$$\dim\ker\, L(\Xi_1,\cdots,\Xi_{D-d};\Pi_1,\cdots,\Pi_d;A)<nd$$

Hence $\mathbb{R}^{n\times d}\setminus\mathcal{S}([I_d\ \ A])$ is a finite union of subsets of closed linear spaces properly included in $\mathbb{R}^{n\times d}$. This proves the theorem. $\square$

Additional Relations

Note the following relationship and matrix representation of X when matrices are column-stacked:

$$M(\Pi, \Pi_1, \cdots, \Pi_d) = L(\Pi, \cdots, \Pi; \Pi_1, \cdots, \Pi_d; I)$$

$$L \equiv \begin{bmatrix} A_{1,1}(\Xi_1 - \Pi_1) & A_{2,1}(\Xi_1 - \Pi_2) & \cdots & A_{d,1}(\Xi_1 - \Pi_d) \\ A_{1,2}(\Xi_2 - \Pi_1) & A_{2,2}(\Xi_2 - \Pi_2) & \cdots & A_{d,2}(\Xi_2 - \Pi_d) \\ \vdots & \vdots & \ddots & \vdots \\ A_{1,D-d}(\Xi_{D-d} - \Pi_1) & A_{2,D-d}(\Xi_{D-d} - \Pi_2) & \cdots & A_{d,D-d}(\Xi_{D-d} - \Pi_d \end{bmatrix}$$

a $n(D-d) \times nd$ matrix.

Universal keys

Theorem

Consider the metric space $(\widehat{\mathbb{R}^{n\times d}}, d)$.
There exists a bi-Lipschitz map

$$\hat{\beta} : \widehat{\mathbb{R}^{n\times d}} \to \mathbb{R}^{n\times D} \sim \mathbb{R}^m$$

with $D = 1 + (d-1)n!$ and $m = (1 + (d-1)n!)n$. This map is given explicitly by $\hat{\beta}(\hat{X}) = \downarrow(XA)$ for any $A \in \mathbb{R}^{d\times(1+(d-1)n!)}$ whose columns form a full spark frame, and where $\downarrow$ acts column-wise.

Towards universal keys

Relation (*) from the proof of previous theorem provides an algorithm to check if a matrix A is a universal key. It is likely that if a universal key exists for a triple (n, d, D) then universal keys are generic in $\mathbb{R}^{d\times(D-d)}$.
Open Problem: Given (n, d) find the smallest dimension D (or $D-d$) so that there exists a universal key $A \in \mathbb{R}^{d\times(D-d)}$ for $\mathbb{R}^{n\times d}$.
So far we obtained:

n	d	D-d
2	2	1
3	2	2
4	2	2
5	2	?

Table of Contents

1. Motivation
2. Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n\times d}$
3. Polynomial Embeddings
4. Sorting based Embeddings
5. Towards Embeddings of $\hat{V}$ for $V = Sym(n)$
6. Numerical Examples

The metric space $\widehat{\mathrm{Sym}(n)}$

The real vector space $V = \mathrm{Sym}(n) = \{A = A^T \in \mathbb{R}^{n\times n}\}$ is of dimension $N = \frac{n(n+1)}{2}$. The permutation group $\mathcal{S}_n$ acts on V by the similarity transformation $(P, A) \in \mathcal{S}_n \times \mathrm{Sym}(n) \mapsto PAP^T$. The metric space $\widehat{\mathrm{Sym}(n)} = \mathrm{Sym}(n)/\sim$ of equivalence classes admits the natural metric:

$$d(\hat{A}, \hat{B}) = \min_{P\in\mathcal{S}_n} \|A - PBP^T\|_F$$

induuced by the Frobenius norm $\|\cdot\|_F$.

Problem: Construct a (bi)Lipschitz map $\hat{\beta} : (\widehat{\mathrm{Sym}(n)}, d) \to \mathbb{R}^m$.
Specifically, construct $\beta : \mathrm{Sym}(n) \to \mathbb{R}^m$, $a_0, b_0 > 0$ so that for any $A, B \in \mathrm{Sym}(n)$:

1. $\hat{A} = \hat{B}$ if and only if $\beta(A) = \beta(B)$;
2. $a_0 d(\hat{A}, \hat{B}) \leq \|\beta(A) - \beta(B)\|_2 \leq b_0 d(\hat{A}, \hat{B})$

Then $\hat{\beta}(\hat{A}) = \beta(A)$ and $\hat{\beta}$ lifts β to $\widehat{\mathrm{Sym}(n)}$.

The group representation point of view

The same action can be viewed as a representation of a subgroup of $\mathcal{S}_N$ acting on $\mathbb{R}^N$ where $N = n(n+1)/2$. Specifically, let $i : \mathrm{Sym}(n) \to \mathbb{R}^N$ and $j : \mathrm{Sym}(n) \to \mathbb{R}^{n^2}$ be the linear maps:

$$i(A) = (A_{1,1}, \cdots, A_{n,n}, A_{1,2}, \cdots, A_{1,n}, A_{2,3}, \cdots, A_{2,n}, \cdots, A_{n-1,n})^T$$

$$j(A) = vect(A) = (A_{1,1}, \cdots, A_{n,1}, A_{1,2}, \cdots, A_{n,2}, \cdots, A_{1,n}, \cdots, A_{n,n})^T$$

Note i is an isomorphism, whereas j is injective but not surjective. Let $E = Ran(j) \subset \mathbb{R}^{n^2} \sim \mathbb{R}^N$.

The action $A \mapsto PAP^T$ is implemented by the linear map $L_P : \mathbb{R}^{n^2} \to \mathbb{R}^{n^2}$, $L_P(\xi) = (P \otimes P)\xi$. Each subspace E is invariant to the action of L_P. This invariance induces a pull-back $T_P : \mathbb{R}^N \to \mathbb{R}^N$ which remains a permutation matrix on $\mathbb{R}^N$. Thus we obtain a linear representation of $\mathcal{S}_n$ seen as a subgroup of $\mathcal{S}_N$ acting on $\mathbb{R}^N$, via $(\Pi, v) \mapsto T_P v$.

Polynomial Invariants

Based on joint work with Efstratios Tsukanis.

The main task is to find a characterization of the algebra of invariant symmetric polynomials in n^2 variables $\mathbb{A} = \mathbb{R}[X_{1,1}, \cdots, X_{n,n}]^{\mathcal{S}_n}$. For easiness of notation, we shall collect into a matrix denoted by A the n^2 variables of these polynomials. Thus we are interested in finding polynomials $Q(A)$ in entries of A that satisfy:

1. $Q(A) = Q(A^T)$ for all $A \in \mathbb{R}^{n\times n}$;
2. $Q(\Pi A \Pi^T) = Q(A)$ for all $\Pi \in \mathcal{S}_n$.

The algebra $\mathbb{A}$ is graded: $\mathbb{A} = \oplus_{d\geq 0} H_d$, where each H_d denotes the vector space of homogeneous polynomials of degree d in $\mathbb{A}$.

Polynomial Invariants (2)

An homogeneous polynomial of degree d in entries of A can be compactly written as $Q(A) = \mathrm{trace}(W \cdot (A \otimes A \otimes \cdots \otimes A))$ for some $W \in \mathbb{R}^{nd \times nd}$.
Each invariant symmetric polynomial $Q \in H_d$ should satisfy:

$$W^T = W \quad , \quad W(\Pi \otimes \cdots \otimes \Pi) = (\Pi \otimes \cdots \otimes \Pi)W \ , \ \forall \Pi \in \mathcal{S}_n$$

Thus H_d can be identified with the self-adjoint elements of the commutant of the algebra generated by $\{\Pi^{\otimes d} \ , \ \Pi \in \mathcal{S}_n\}$. Let $\mathcal{C}_d = \{\Pi^{\otimes d} \ , \ \Pi \in \mathcal{S}_n\}'$ denote this commutant.

Proposition (see also Schneider et.al ('17))

For $d = 1$, $\dim \mathcal{C}_1 = 2$ *with a basis provided by* $W_1 = I_n$ *and* $W_2 = 11^T$.
Thus $\dim H_1 = 2$ *and a basis is provided by:*

$$Q_1(A) = trace(A) = \sum_i A_{i,i} \quad , \quad Q_2(A) = 1^T A 1 = \sum_{i,j} A_{i,j}$$

Polynomial Invariants (3)

Let $D = diag(A \cdot 1)$ be the diagonal matrix that collects the row-sums of A. Note the graph Laplacian is defined as $\Delta = D - A$, and, if $A' = \Pi A \Pi^T$ then $D' = \Pi D \Pi^T$. This covariance property provides us with a large class of invariant symmetric polynomials:

$$Q_{p_1,q_1,p_2,q_2,\cdots,p_L,q_L}(A) = \mathrm{trace}\left(A^{p_1} D^{q_1} A^{p_2} D^{q_2} \cdots A^{p_L} D^{q_L}\right).$$

With this notation, the previous basis for H_1 is provided by $\{Q_{1,0}, Q_{0,1}\}$.

A plausible conjecture: The system $Q_{p_1,q_1,p_2,q_2,\cdots,p_L,q_L}$ defines a complete system of invariant polynomials.

Polynomial Invariants (3)

Let $D = diag(A \cdot 1)$ be the diagonal matrix that collects the row-sums of A. Note the graph Laplacian is defined as $\Delta = D - A$, and, if $A' = \Pi A \Pi^T$ then $D' = \Pi D \Pi^T$. This covariance property provides us with a large class of invariant symmetric polynomials:

$$Q_{p_1,q_1,p_2,q_2,\cdots,p_L,q_L}(A) = \mathrm{trace}\left(A^{p_1} D^{q_1} A^{p_2} D^{q_2} \cdots A^{p_L} D^{q_L}\right).$$

With this notation, the previous basis for H_1 is provided by $\{Q_{1,0}, Q_{0,1}\}$.

A plausible conjecture: The system $Q_{p_1,q_1,p_2,q_2,\cdots,p_L,q_L}$ defines a complete system of invariant polynomials.

However this is not true: for $d = 2$ we obtained dim $H_2 = 7$, whereas only 3 generators are of this form $\{Q_{2,0}, Q_{1,1}, Q_{0,2}\}$.

Table of Contents

1. Motivation
2. Embeddings of $\hat{V}$ for $V = \mathbb{R}^{n\times d}$
3. Polynomial Embeddings
4. Sorting based Embeddings
5. Towards Embeddings of $\hat{V}$ for $V = Sym(n)$
6. Numerical Examples

The Protein Dataset

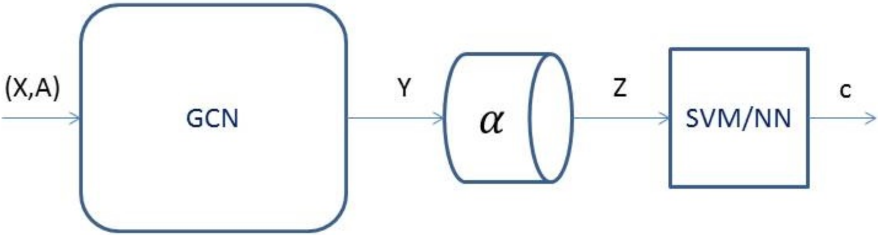
This section is based on joint work with Naveed Haghani and Maneesh Singh.

Protein Dataset: selection of 450 enzymes and 450 non-enzymes out of 1113 proteins. Each graph associated to one protein: nodes represent amino acids and edges represent the bonds between them. Number of nodes: varying between 10 and capped at 50.
Task: the task is classification of each protein into *enzyme* or *non-enzyme*.

The Deep Network Architecture

Architecture: ReLU activation and

- GCN with $L = 3$ layers and 29 input feature vectors, and 50 hidden nodes in each layer; no dropouts, no batch normalization. output of GCN: $d = 1, 10, 100$.
- Mid-layer component: α
- Fully connected NN with dense 3-layers and 150 internal units; no dropouts, with batch normalization.



The Network

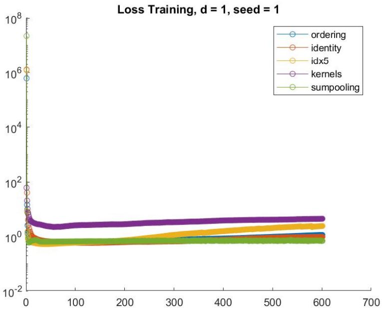
Training has been done over 3000 epochs with a batch size of 100. Loss function: cross-entropy.

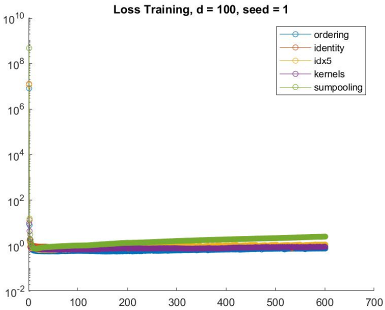
The following 5 α blocks have been tested:

1. Identity: $\alpha(X) = X$; no permutation invariance.
2. Identity $\times$ 5: $\alpha(X) = X$ BUT the training data set has been augmented with 4 random permutatons of each graph.
3. ordering: $\alpha(X) = \downarrow(XA)$, $A = [I\ 1]$
4. kernels: $\alpha(X) = (\sum_{k=1}^{n} exp(-\|x_k - a_j\|^2/\sigma_j^2))_{1\leq j\leq m=nd}$
5. sumpooling: $\alpha(X) = 1^T X$

Enzyme Classification Example

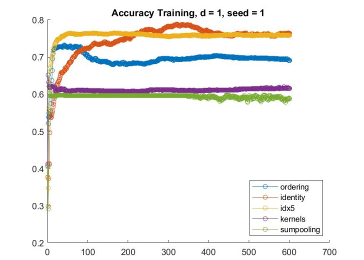
Training Loss: X Entropy

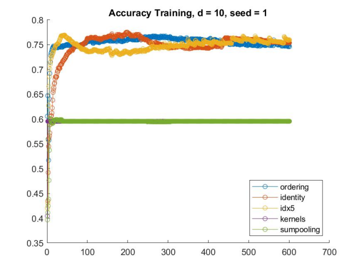


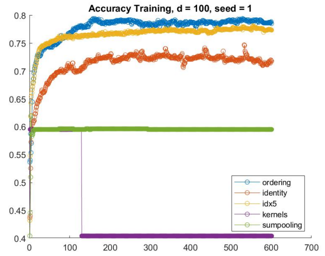


Enzyme Classification Example

Training Accuracy

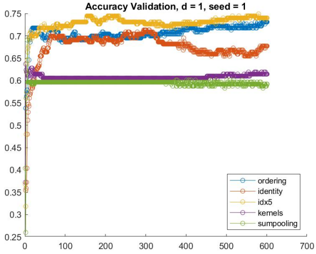


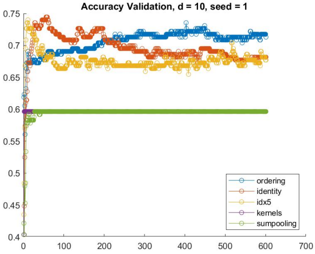


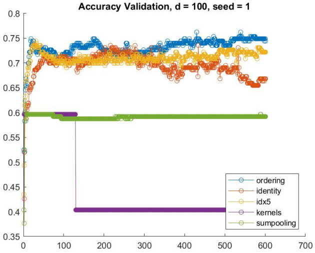


Enzyme Classification Example

Validation Accuracy

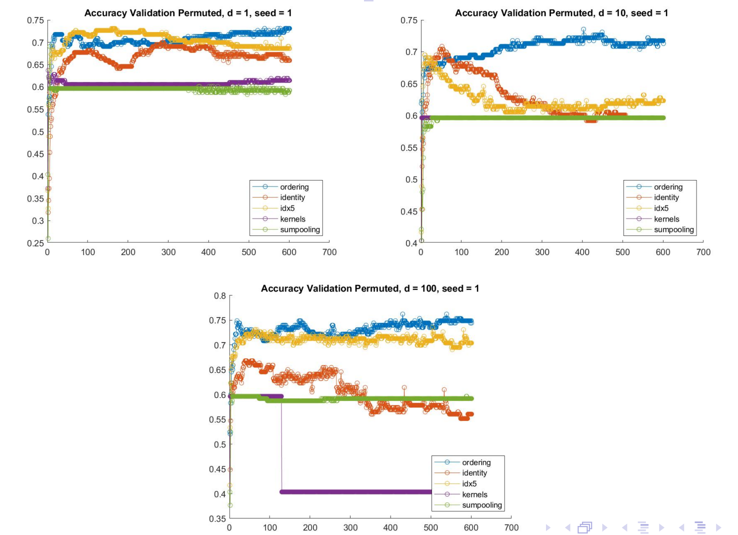






Enzyme Classification Example

Validation Accuracy with Random Permutations



The QM9 Dataset

Dataset: Consists of 134,000 isomers of organic molecules made up of CHONF, each containing 10-29 atoms. see http://quantum-machine.org/datasets/ Nodes corresponds to atoms; each feature vector containins geometry (x,y,z coordinates), partial charge per atom (Mulliken charge), and atom type.

Task: the task is regression: predict a physical feature (electron energy gap) computed for each molecule.

Architecture: ReLU activation and

- GCN with $L = 3$ layers and 50 hidden nodes in each layer; no dropouts, no batch normalization; zero padding to $m = 29$ number of rows. output of GCN: $d = 1, 10, 100$.
- Mid-layer component: α
- Fully connected NN with dense 3-layers and 150 internal units in each of the two hidden layers; no dropouts, with batch normalization.

The Network

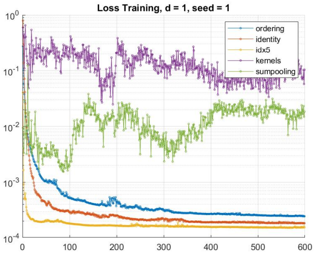
Training has been done over 3000 epochs with a batch size of 100. Loss function: Mean-Square Error (MSE).

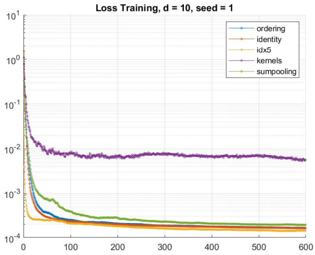
The following 5 α blocks have been tested:

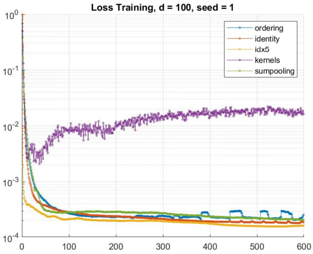
1. Identity: $\alpha(X) = X$; no permutation invariance.
2. Identity $\times$ 5: $\alpha(X) = X$ BUT the training data set has been augmented with 4 random permutatons of each graph.
3. ordering: $\alpha(X) = \downarrow(XA)$, $A = [I \ 1]$
4. kernels: $\alpha(X) = (\sum_{k=1}^{n} exp(-\|x_k - a_j\|^2/\sigma_j^2))_{1 \leq j \leq m=nd}$
5. sumpooling: $\alpha(X) = 1^T X$

QM9 Regression Example

Training MSE

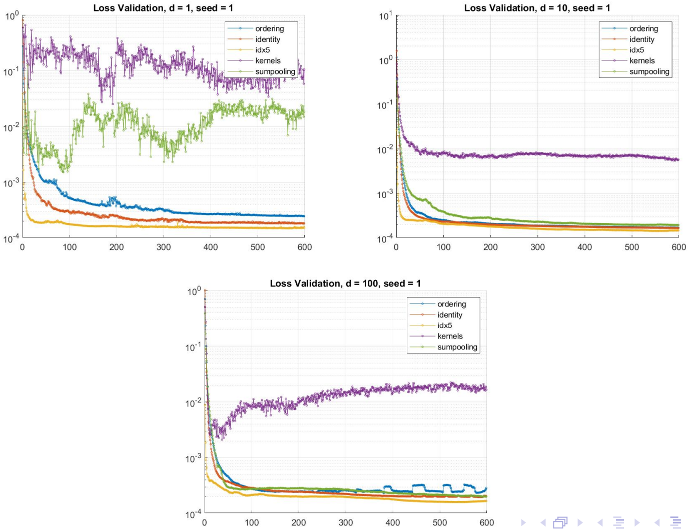






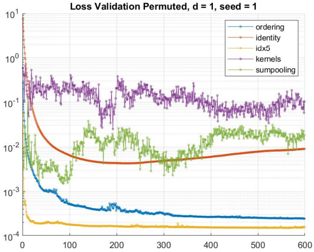
QM9 Regression Example

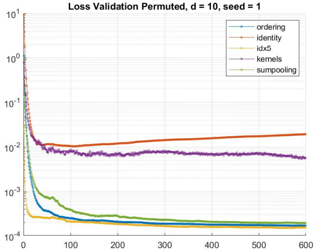
Validation MSE

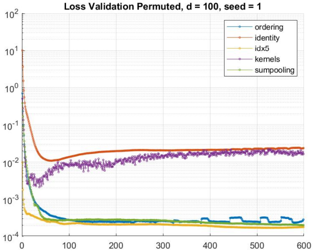


QM9 Regression Example

Validation MSE with Random Permutations







Bibliography

[1] Vinyals, O., Fortunato, M., and Jaitly, N., Pointer Networks, arXiv e-prints , arXiv:1506.03134 (Jun 2015).
[2] Sutskever, I., Vinyals, O., and Le, Q. V., Sequence to Sequence Learning with Neural Networks, arXiv e-prints , arXiv:1409.3215 (Sep 2014).
[3] Bello, I., Pham, H., Le, Q. V., Norouzi, M., and Bengio, S., Neural Combinatorial Optimization with Reinforcement Learning, arXiv e-prints , arXiv:1611.09940 (Nov 2016).
[4] Williams, R. J., Simple statistical gradient-following algorithms for connectionist reinforcement learning, Machine learning 8(3-4), 229-256 (1992).
[5] Kool, W., van Hoof, H., and Welling, M., Attention, Learn to Solve Routing Problems, arXiv e-prints , arXiv:1803.08475 (Mar 2018).

Bibliography

[6] Dai, H., Khalil, E. B., Zhang, Y., Dilkina, B., and Song, L., Learning Combinatorial Optimization Algorithms over Graphs, arXiv e-prints , arXiv:1704.01665 (Apr 2017).
[7] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al., Human-level control through deep reinforcement learning, Nature 518(7540), 529 (2015).
[8] Dai, H., Dai, B., and Song, L., Discriminative embeddings of latent variable models for structured data, in International conference on machine learning, 2702-2711 (2016).
[9] Nowak, A., Villar, S., Bandeira, A. S., and Bruna, J., Revised Note on Learning Algorithms for Quadratic Assignment with Graph Neural Networks, arXiv e-prints , arXiv:1706.07450 (Jun 2017).

Bibliography

[10] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G., The graph neural network model, IEEE Transactions on Neural Networks 20(1), 61-80 (2008).
[11] Li, Z., Chen, Q., and Koltun, V., Combinatorial Optimization with Graph Convolutional Networks and Guided Tree Search, arXiv e-prints , arXiv:1810.10659 (Oct 2018).
[12] Kipf, T. N. and Welling, M., Semi-Supervised Classification with Graph Convolutional Networks, arXiv e-prints , arXiv:1609.02907 (Sep 2016).
[13] Kingma, D. P. and Ba, J., Adam: A Method for Stochastic Optimization, arXiv e-prints , arXiv:1412.6980 (Dec 2014).
[14] H. Derksen, G. Kemper, Computational Invariant Theory, Springer 2002.

Bibliography

[15] J. Cahill, A. Contreras, A.C. Hip, Complete Set of translation Invariant Measurements with Lipschitz Bounds, arXiv:1903.02811 (2019).
[16] M. Zaheer, S. Kottur, S. Ravanbhakhsh, B. Poczos, R. Salakhutdinov, A.J. Smola, Deep Sets, arXiv:1703.06114
[17] H. Maron, E. Fetaya, N. Segol, Y. Lipman, On the Universality of Invariant Networks, arXiv:1901.09342 [cs.LG] (May 2019).
[18] M.M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric deep learning: going beyond euclidean data. CoRR, abs/1611.08097, 2016.
[19] S. Ravanbaksh, J. Schneider, B. Poczos, Equivariance through parameter sharing, ICML 2017.

Thank you!

Questions?