Neural Injective Functions for Multisets, Measures and Graphs via a Finite Witness Theorem

Tal Amir[1] Steven J. Gortler[2]
Ilai Avni[1] Ravina Ravina[1] Nadav Dym[1]

[1] Technion – Israel Institute of Technology, Haifa, Israel
[2] Harvard University, Cambridge, USA

13 Nov. 2023

Introduction

Multisets: Like sets, but allow repetitions.

Introduction

Multisets: Like sets, but allow repetitions.

They are the natural way to represent:

- 3D point-clouds

Introduction

Multisets: Like sets, but allow repetitions.

They are the natural way to represent:

- 3D point-clouds
- Neighborhoods of vertices in a graph

Introduction

Multisets: Like sets, but allow repetitions.

They are the natural way to represent:

- 3D point-clouds
- Neighborhoods of vertices in a graph
- Any data structure with an intrinsic order that is irrelevant to the problem.

Introduction

Multisets: Like sets, but allow repetitions.

They are the natural way to represent:

- 3D point-clouds
- Neighborhoods of vertices in a graph
- Any data structure with an intrinsic order that is irrelevant to the problem.

The collection of all multisets with at most n elements that come from a fixed set $\Omega \subseteq \mathbb{R}^d$:

$$\mathcal{S}_{\leq n}(\Omega) = \{ \{\!\{ \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \}\!\} \mid \boldsymbol{x}_i \in \Omega,\ k \leq n \}$$

Introduction

Multisets: Like sets, but allow repetitions.

They are the natural way to represent:

- 3D point-clouds
- Neighborhoods of vertices in a graph
- Any data structure with an intrinsic order that is irrelevant to the problem.

The collection of all multisets with at most n elements that come from a fixed set $\Omega \subseteq \mathbb{R}^d$:

$$\mathcal{S}_{\leq n}(\Omega) = \{\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\} \mid \boldsymbol{x}_i \in \Omega,\ k \leq n\}$$

We refer to Ω as an *alphabet*.

Introduction

Goals:

1. Develop an efficient method to represent multisets by an *embedding*

$$F : \mathcal{S}_{\leq n}(\mathbb{R}^d) \rightarrow \mathbb{R}^m.$$

Introduction

Goals:

1. Develop an efficient method to represent multisets by an *embedding*

$$F : \mathcal{S}_{\leq n}(\mathbb{R}^d) \to \mathbb{R}^m.$$

Desired properties of F:

Introduction

Goals:

1. Develop an efficient method to represent multisets by an *embedding*

$$F : \mathcal{S}_{\leq n}(\mathbb{R}^d) \rightarrow \mathbb{R}^m.$$

Desired properties of F:

a. Injective

Introduction

Goals:

1. Develop an efficient method to represent multisets by an *embedding*

$$F : \mathcal{S}_{\leq n}(\mathbb{R}^d) \to \mathbb{R}^m.$$

Desired properties of F:

a. Injective
b. Permutation invariant

Introduction

Goals:

1. Develop an efficient method to represent multisets by an *embedding*

$$F : \mathcal{S}_{\leq n}(\mathbb{R}^d) \rightarrow \mathbb{R}^m.$$

Desired properties of F:

a. Injective
b. Permutation invariant
c. Have a low output-dimension m

Introduction

Goals:

1. Develop an efficient method to represent multisets by an *embedding*

$$F : \mathcal{S}_{\leq n}(\mathbb{R}^d) \to \mathbb{R}^m.$$

   Desired properties of F:

   a. Injective
   b. Permutation invariant
   c. Have a low output-dimension m

2. Approximate *any* function on $\mathcal{S}_{\leq n}(\mathbb{R}^d)$, by composing F with existing architectures.

A popular approach: Moment functions

Any $f : \Omega \to \mathbb{R}^m$ induces a *moment function* $\hat{f} : \mathcal{S}_{\leq n}(\Omega) \to \mathbb{R}^m$:

$$\hat{f}\left(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}\right) = \sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

A popular approach: Moment functions

$$\hat{f}\left(\{\!\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_k\}\!\}\right)=\sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

A popular approach: Moment functions

$$\hat{f}\left(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}\right) = \sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

Studied in theory: Polynomial moments

Example. For $d = 1$, $n = 2$,

$$\hat{f}(\{\!\{x_1, x_2\}\!\}) = \left(x_1 + x_2, x_1^2 + x_2^2\right)$$

is injective on $\mathcal{S}_{\leq 2}(\mathbb{R})$.

A popular approach: Moment functions

$$\hat{f}\left(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}\right) = \sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

Studied in theory: Polynomial moments

Example. For $d = 1$, $n = 2$,

$$\hat{f}(\{\!\{x_1, x_2\}\!\}) = \left(x_1 + x_2, x_1^2 + x_2^2\right)$$

is injective on $\mathcal{S}_{\leq 2}(\mathbb{R})$.

- For $d > 1$, previous works require m that is exponential or high-polynomial in d, n (Balan, Haghani, and Singh 2022; Maron et al. 2019; Segol and Lipman 2019; Wang et al. 2023).

A popular approach: Moment functions

$$\hat{f}\left(\{\!\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_k\}\!\}\right)=\sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

Studied in theory: Polynomial moments

Example. For $d = 1$, $n = 2$,

$$\hat{f}(\{\!\{x_1, x_2\}\!\}) = \left(x_1 + x_2, x_1^2 + x_2^2\right)$$

is injective on $\mathcal{S}_{\leq 2}(\mathbb{R})$.

- For $d > 1$, previous works require m that is exponential or high-polynomial in d, n (Balan, Haghani, and Singh 2022; Maron et al. 2019; Segol and Lipman 2019; Wang et al. 2023).
- Recently $m = 2nd + 1$ was achieved using polynomials with random coefficients (Dym and Gortler 2022).

A popular approach: Moment functions

$$\hat{f}\left(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}\right) = \sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

Used in practice: Neural moments

(Zaheer et al. 2017)

$$\hat{f}(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) = \sum_{i=1}^{k} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})$$

A popular approach: Moment functions

$$\hat{f}\left(\{\!\{ \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \}\!\}\right) = \sum_{i=1}^{k} f(\boldsymbol{x}_i)$$

Used in practice: Neural moments

(Zaheer et al. 2017)

$$\hat{f}(\{\!\{ \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \}\!\}) = \sum_{i=1}^{k} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})$$

$\rightarrow$ Not known to be injective.

Main Result

Theorem. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an analytic non-polynomial function. Let $m \geq 2nd + 1$. Then for almost any $\boldsymbol{A} \in \mathbb{R}^{m \times d}$, $\boldsymbol{b} \in \mathbb{R}^m$, the shallow network

$$\hat{f}(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) = \sum_{i=1}^{n} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})$$

is injective on $\mathcal{S}_{\leq n}(\mathbb{R}^d)$.

Main Result

Theorem. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an analytic non-polynomial function. Let $m \geq 2nd + 1$. Then for almost any $\boldsymbol{A} \in \mathbb{R}^{m \times d}$, $\boldsymbol{b} \in \mathbb{R}^m$, the shallow network

$$\hat{f}(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) = \sum_{i=1}^{n} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})$$

is injective on $\mathcal{S}_{\leq n}(\mathbb{R}^d)$.
For injectivity on $\mathcal{S}_{\leq n}(\Omega)$ with a countable Ω, it suffices to take $m \geq 1$.

Main Result

Theorem. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an analytic non-polynomial function. Let $m \geq 2nd + 1$. Then for almost any $\boldsymbol{A} \in \mathbb{R}^{m \times d}$, $\boldsymbol{b} \in \mathbb{R}^m$, the shallow network

$$\hat{f}(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) = \sum_{i=1}^{n} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})$$

is injective on $\mathcal{S}_{\leq n}(\mathbb{R}^d)$.
For injectivity on $\mathcal{S}_{\leq n}(\Omega)$ with a countable Ω, it suffices to take $m \geq 1$.

- More generally, $m = 2D + 1$ is required, where D is the *intrinsic dimension* of the input space $\mathcal{S}_{\leq n}(\Omega)$.

Main Result

Theorem. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an analytic non-polynomial function. Let $m \geq 2nd + 1$. Then for almost any $\boldsymbol{A} \in \mathbb{R}^{m \times d}$, $\boldsymbol{b} \in \mathbb{R}^m$, the shallow network

$$\hat{f}(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) = \sum_{i=1}^{n} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})$$

is injective on $\mathcal{S}_{\leq n}(\mathbb{R}^d)$.
For injectivity on $\mathcal{S}_{\leq n}(\Omega)$ with a countable Ω, it suffices to take $m \geq 1$.

- More generally, $m = 2D + 1$ is required, where D is the *intrinsic dimension* of the input space $\mathcal{S}_{\leq n}(\Omega)$.
- This required size is near-optimal (essentially up to a multiplicative factor of 2).

Applications

Applications

1. Universal approximation of functions on multisets

Corollary. Let $K \subseteq \mathbb{R}^d$ be compact. Let σ be analytic and non-polynomial. Set $m = 2nd + 1$. Then for almost all $\boldsymbol{A} \in \mathbb{R}^{m \times d}$, $\boldsymbol{b} \in \mathbb{R}^m$, any continuous $f : \mathcal{S}_{\leq n}(K) \to \mathbb{R}$ can be approximated by functions of the form

$$f(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) \approx F\left(\sum_{i=1}^{k} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})\right), \quad \text{with } F \text{ being an MLP.}$$

Applications

1. Universal approximation of functions on multisets

Corollary. Let $K \subseteq \mathbb{R}^d$ be compact. Let σ be analytic and non-polynomial. Set $m = 2nd + 1$. Then for almost all $\boldsymbol{A} \in \mathbb{R}^{m \times d}$, $\boldsymbol{b} \in \mathbb{R}^m$, any continuous $f : \mathcal{S}_{\leq n}(K) \to \mathbb{R}$ can be approximated by functions of the form

$$f(\{\!\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}\!\}) \approx F\left(\sum_{i=1}^{k} \sigma(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b})\right), \quad \text{with } F \text{ being an MLP.}$$

Previous works use MLPs to *approximate* the injective function fed to F (Maron et al. 2019; Xu et al. 2018; Zaheer et al. 2017). The number of neurons required for injectivity was not known, and in some cases is infinite.

Applications

2. Weisfeiler-Leman equivalent MPNNs with neural aggregators

Corollary. Consider an MPNN with random weights, analytic non-polynomial activations, and one hidden feature in $\mathbb{R}$ per vertex. Such MPNN, when run for T iterations, returns different outputs for any two graphs that can be separated by T iterations of 1-WL.

Applications

2. Weisfeiler-Leman equivalent MPNNs with neural aggregators

Corollary. Consider an MPNN with random weights, analytic non-polynomial activations, and one hidden feature in $\mathbb{R}$ per vertex. Such MPNN, when run for T iterations, returns different outputs for any two graphs that can be separated by T iterations of 1-WL.

$\rightarrow$ Can support vertex features in $\mathbb{R}^d$. Requires hidden dimension $m \geq 2nd + 1$.

Applications

2. Weisfeiler-Leman equivalent MPNNs with neural aggregators

Corollary. Consider an MPNN with random weights, analytic non-polynomial activations, and one hidden feature in $\mathbb{R}$ per vertex. Such MPNN, when run for T iterations, returns different outputs for any two graphs that can be separated by T iterations of 1-WL.

$\rightarrow$ Can support vertex features in $\mathbb{R}^d$. Requires hidden dimension $m \geq 2nd + 1$.

Previous works use multiset functions that are not MLPs (Xu et al. 2018), or require a number of parameters and node features that depends polylogarithmically on the graph size (Aamand et al. 2022; Morris, Ritzert, et al. 2019).

Applications

2. Weisfeiler-Leman equivalent MPNNs with neural aggregators

Corollary. Consider an MPNN with random weights, analytic non-polynomial activations, and one hidden feature in $\mathbb{R}$ per vertex. Such MPNN, when run for T iterations, returns different outputs for any two graphs that can be separated by T iterations of 1-WL.

$\rightarrow$ Can support vertex features in $\mathbb{R}^d$. Requires hidden dimension $m \geq 2nd + 1$.

Previous works use multiset functions that are not MLPs (Xu et al. 2018), or require a number of parameters and node features that depends polylogarithmically on the graph size (Aamand et al. 2022; Morris, Ritzert, et al. 2019).

Our work uses a single node feature and a constant number of parameters.

Negative results: Limitations of moment functions

Negative results: Limitations of moment functions

1. Moments of neural networks with piecewise-linear activations (e.g. ReLU, leaky ReLU, HardTanh) cannot be injective:

Proposition. Let $\Omega \subseteq \mathbb{R}^d$ be an open set, and let $n \geq 2$. If $\psi : \mathbb{R}^d \to \mathbb{R}^m$ is piecewise linear, then its moment $\hat{\psi} : \mathcal{S}_{\leq n}(\Omega) \to \mathbb{R}^m$ is not injective.

Negative results: Limitations of moment functions

1. Moments of neural networks with piecewise-linear activations (e.g. ReLU, leaky ReLU, HardTanh) cannot be injective:

Proposition. Let $\Omega \subseteq \mathbb{R}^d$ be an open set, and let $n \geq 2$. If $\psi : \mathbb{R}^d \to \mathbb{R}^m$ is piecewise linear, then its moment $\hat{\psi} : \mathcal{S}_{\leq n}(\Omega) \to \mathbb{R}^m$ is not injective.

2. Even when moment functions are injective, they can never be bi-Lipschitz:

Proposition. Let $n \geq 2$, and let $f : \mathbb{R}^d \to \mathbb{R}^m$ be differentiable at some $\boldsymbol{x}_0 \in \mathbb{R}^d$. Then the induced moment function $\hat{f} : \mathcal{S}_{\leq n}(\mathbb{R}^d) \to \mathbb{R}^m$ is not bi-Lipschitz.

Numerical Demonstration

Hidden Dimension	Analytic			Piecewise Linear		
	Tanh	SiLU	Sin	HardTanh	ReLU	Leaky ReLU
1	0	0	0	7	17	7
10	0	0	0	3	7	7
50	0	0	0	4	5	5
100	0	0	0	1	0	0

Table: Number of non-isomorphic pairs of graphs not separated by MPNN, out of the 600 pairs in the TUDataset (Morris, Kriege, et al. 2020)

Finite Witness Theorem

Our injectivity results are based on a novel theorem, which enables reducing an infinite family of analytic equality constraints

$$\{F(\boldsymbol{x};\boldsymbol{\theta}) = 0 \mid \boldsymbol{\theta} \in \mathbb{W}\}$$

to a finite subset with random parameters:

$$\{F\left(\boldsymbol{x};\boldsymbol{\theta}^{(i)}\right) = 0 \mid i = 1, \dots, m\}.$$

Finite Witness Theorem

Finite Witness Theorem

Theorem. Let $\mathbb{M} \subseteq \mathbb{R}^p$ be an admissible set (see below) of dimension D, and let $\mathbb{W} \subseteq \mathbb{R}^q$ be open and connected. Let $F : \mathbb{M} \times \mathbb{W} \to \mathbb{R}$ be an analytic function. Let $\mathcal{N}$ be the set

$$\mathcal{N} = \{\boldsymbol{x} \in \mathbb{M} \mid F(\boldsymbol{x}; \boldsymbol{\theta}) = 0, \ \forall \boldsymbol{\theta} \in \mathbb{W}\}.$$

Then for almost any $\left(\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(D+1)}\right) \in \mathbb{W}^{D+1}$,

$$\mathcal{N} = \{\boldsymbol{x} \in \mathbb{M} \mid F(\boldsymbol{x}; \boldsymbol{\theta}^{(i)}) = 0, \ \forall i = 1, \ldots D + 1\}.$$

Finite Witness Theorem

Finite Witness Theorem

Theorem. Let $\mathbb{M} \subseteq \mathbb{R}^p$ be an admissible set (see below) of dimension D, and let $\mathbb{W} \subseteq \mathbb{R}^q$ be open and connected. Let $F : \mathbb{M} \times \mathbb{W} \to \mathbb{R}$ be an analytic function. Let $\mathcal{N}$ be the set

$$\mathcal{N} = \{\boldsymbol{x} \in \mathbb{M} \mid F(\boldsymbol{x}; \boldsymbol{\theta}) = 0, \ \forall \boldsymbol{\theta} \in \mathbb{W}\}.$$

Then for almost any $\left(\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(D+1)}\right) \in \mathbb{W}^{D+1}$,

$$\mathcal{N} = \{\boldsymbol{x} \in \mathbb{M} \mid F(\boldsymbol{x}; \boldsymbol{\theta}^{(i)}) = 0, \ \forall i = 1, \ldots D+1\}.$$

- The class of sets admissible as $\mathbb{M}$ is vast: It includes all open sets, closed ℓ_2-balls, polygons, as well as countable unions and finite intersections thereof.

Finite Witness Theorem

Finite Witness Theorem

Theorem. Let $\mathbb{M} \subseteq \mathbb{R}^p$ be an admissible set (see below) of dimension D, and let $\mathbb{W} \subseteq \mathbb{R}^q$ be open and connected. Let $F : \mathbb{M} \times \mathbb{W} \to \mathbb{R}$ be an analytic function. Let $\mathcal{N}$ be the set

$$\mathcal{N} = \{\boldsymbol{x} \in \mathbb{M} \mid F(\boldsymbol{x}; \boldsymbol{\theta}) = 0, \ \forall \boldsymbol{\theta} \in \mathbb{W}\}.$$

Then for almost any $\left(\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(D+1)}\right) \in \mathbb{W}^{D+1}$,

$$\mathcal{N} = \{\boldsymbol{x} \in \mathbb{M} \mid F(\boldsymbol{x}; \boldsymbol{\theta}^{(i)}) = 0, \ \forall i = 1, \ldots D+1\}.$$

- The class of sets admissible as $\mathbb{M}$ is vast: It includes all open sets, closed ℓ_2-balls, polygons, as well as countable unions and finite intersections thereof.
- The full version of the theorem admits a wider class of functions, which in particular includes all semialgebraic functions.

Generalizing to measures

Our results can be generalized to **signed measures**:

$$\mathcal{M}_{\leq n}(\Omega) = \{\sum_{i=1}^{n} w_i \delta_{\boldsymbol{x}_i} \mid \boldsymbol{x}_i \in \Omega,\ w_i \in \mathbb{R},\ k \leq n\}.$$

Generalizing to measures

Our results can be generalized to **signed measures**:

$$\mathcal{M}_{\leq n}(\Omega) = \{\sum_{i=1}^{n} w_i \delta_{\boldsymbol{x}_i} \mid \boldsymbol{x}_i \in \Omega,\ w_i \in \mathbb{R},\ k \leq n\}.$$

- Can represent weighted point-clouds and vertex-neighborhoods in weighted graphs.

Generalizing to measures

Our results can be generalized to **signed measures**:

$$\mathcal{M}_{\leq n}(\Omega) = \{\sum_{i=1}^{n} w_i \delta_{\boldsymbol{x}_i} \mid \boldsymbol{x}_i \in \Omega,\ w_i \in \mathbb{R},\ k \leq n\}.$$

- Can represent weighted point-clouds and vertex-neighborhoods in weighted graphs.
- Can approximately represent any signed measure in $\mathbb{R}^d$.

For more information, see our paper:

Tal Amir, Steven J. Gortler, Ilai Avni, Ravina Ravina, and Nadav Dym (2023). "Neural Injective Functions for Multisets, Measures and Graphs via a Finite Witness Theorem". In: *Advances in Neural Information Processing Systems*

For more information, see our paper:

Tal Amir, Steven J. Gortler, Ilai Avni, Ravina Ravina, and Nadav Dym (2023). "Neural Injective Functions for Multisets, Measures and Graphs via a Finite Witness Theorem". In: *Advances in Neural Information Processing Systems*

Thanks for watching

Acknowledgements

N.D. is partially funded by a Horev Fellowship. T.A, R.R. and N.D. are partially funded by ISF grant 272/23.

References

Aamand, Anders et al. (2022). "Exponentially Improving the Complexity of Simulating the Weisfeiler-Lehman Test with Graph Neural Networks". In: *Advances in Neural Information Processing Systems* 35, pp. 27333–27346.

Balan, Radu, Naveed Haghani, and Maneesh Singh (2022). "Permutation invariant representations with applications to graph deep learning". In: *arXiv preprint arXiv:2203.07546.*

Dym, Nadav and Steven J Gortler (2022). "Low Dimensional Invariant Embeddings for Universal Geometric Learning". In: *arXiv preprint arXiv:2205.02956.*

Maron, Haggai et al. (2019). "Provably powerful graph networks". In: *Advances in neural information processing systems* 32.

Morris, Christopher, Nils M. Kriege, et al. (2020). "TUDataset: A collection of benchmark datasets for learning with graphs". In: *ICML 2020 Workshop on Graph Representation Learning and Beyond (GRL+ 2020)*. arXiv: 2007.08663. URL: www.graphlearning.io.

Morris, Christopher, Martin Ritzert, et al. (2019). "Weisfeiler and leman go neural: Higher-order graph neural networks". In: *Proceedings of the AAAI conference on*