Simplicial Neural Networks

Stefania Ebli
Laboratory for Topology and Neuroscience
EPFL, Lausanne, Switzerland
stefania.ebli@epfl.ch

Michaël Defferrard
Signal Processing Laboratory (LTS2)
EPFL, Lausanne, Switzerland
michael.defferrard@epfl.ch

Gard Spreemann*
Laboratory for Topology and Neuroscience
EPFL, Lausanne, Switzerland
gspr@nonempty.org

Abstract

We present simplicial neural networks (SNNs), a generalization of graph neural networks to data that live on a class of topological spaces called simplicial complexes. These are natural multi-dimensional extensions of graphs that encode not only pairwise relationships but also higher-order interactions between vertices—allowing us to consider richer data, including vector fields and n-fold collaboration networks. We define an appropriate notion of convolution that we leverage to construct the desired convolutional neural networks. We test the SNNs on the task of imputing missing data on coauthorship complexes. Code and data are available at https://github.com/stefaniaebli/simplicial_neural_networks.

1 Introduction

The key to convolutional neural networks (CNNs) lies in the way they employ convolution as a local and shift-invariant operation on Euclidean spaces, e.g. $\mathbb{R}$ for audio or $\mathbb{R}^2$ for images. Recently, the concept of CNNs has been extended to more general spaces to exploit different structures that may underlie the data: This includes spherical convolutions for rotationally invariant data [1, 2, 3], more general convolutions on homogeneous spaces [4, 5, 6], or convolutions on graphs [7, 8].

Graph neural networks (GNNs) have proven to be an effective tool that can take into account irregular graphs to better learn interactions in the data [9, 10]. Although graphs are useful in describing complex systems of irregular relations in a variety of settings, they are intrinsically limited to modeling pairwise relationships. The advance of topological methods in machine learning [11, 12, 13], and the earlier establishment of *topological data analysis (TDA)* [14, 15, 16, 17] as a field in its own right, have confirmed the usefulness of viewing data as topological spaces in general, or in particular as simplicial complexes. The latter can be thought of as a higher-dimensional analog of graphs [18, 19]. We here take the view that structure is encoded in *simplicial complexes*, and that these represent n-fold interactions. In this setting, we present *simplicial neural networks (SNNs)*, a neural network framework that take into account locality of data living over a simplicial complex in the same way a GNN does for graphs or a conventional CNN does for grids.

Higher-order relational learning methods, of which hypergraph neural networks [20] and motif-based GNNs [21] are examples, have already proven useful in some applications, e.g. protein interactions [22]. However the mathematical theory underneath the notion of convolution in these

*Work done while at the EPFL. Present affiliation: Telenor Research, Fornebu, Norway.

Topological Data Analysis and Beyond Workshop at the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

approaches does not have clear connections with the global topological structure of the space in question. This leads us to believe that our method, motivated by Hodge–de Rham theory, is far better suited for situations where topological structure is relevant, such as perhaps in the processing of data that exists naturally as vector fields or data that is sensitive to the space's global structure.

2 Proposed method

Simplicial complexes. A simplicial complex is a collection of finite sets closed under taking subsets. We call a member of a simplicial complex K a *simplex* of *dimension* p if it has cardinality $p+1$, and denote the set of all such p-simplices K_p. A p-simplex has $p+1$ *faces* of dimension $p-1$, namely the subsets omitting one element. We denote these $[v_0, \ldots, \hat{v}_i, \ldots, v_p]$ when omitting the i'th element. If a simplex σ is a face of τ, we say that τ is a *coface* of σ. While this definition is entirely combinatorial, there is a geometric interpretation, and it will make sense to refer to and think of 0-simplices as *vertices*, 1-simplices as *edges*, 2-simplices as *triangles*, 3-simplices as *tetrahedra*, and so forth (see Figure 1, (b)).

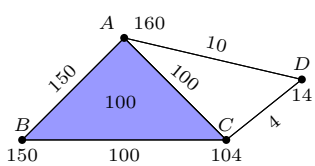
Let $C^p(K)$ be the set of functions $K_p \to \mathbb{R}$, with the obvious vector space structure. These *p-cochains* will encode our data. Define the linear *coboundary* maps $\delta^p : C^p(K) \to C^{p+1}(K)$ by

$$\delta^p(f)([v_0, \ldots, v_{p+1}]) = \sum_{i=0}^{p+1} (-1)^i f([v_0, \ldots, \hat{v}_i, \ldots, v_{p+1}]).$$

Observe that this definition can be thought of in geometric terms: The support of $\delta^p(f)$ is contained in the set of $(p+1)$-simplices that are cofaces of the p-simplices that make up the support of f.

Papers	Authors	Citations
Paper I	A, B, C	100
Paper II	A, B	50
Paper III	A, D	10
Paper IV	C, D	4

(a)



(b)

$$\begin{array}{c} \begin{array}{ccccc} \text{AB} & \text{AC} & \text{AD} & \text{BC} & \text{CD} \end{array} \\ \begin{pmatrix} 3 & 0 & 1 & 0 & 0 \\ 0 & 3 & 1 & 0 & -1 \\ 1 & 1 & 2 & 0 & 1 \\ 0 & 0 & 0 & 3 & -1 \\ 0 & -1 & 1 & -1 & 2 \end{pmatrix} \end{array} \begin{array}{c} \\ \text{AB} \\ \text{AC} \\ \text{AD} \\ \text{BC} \\ \text{CD} \end{array}$$

(c)

Figure 1: Constructing a simplicial complex from data. (a) Coauthorship data. (b) Coauthorship complex with corresponding cochains from the data. (c) Degree-1 Laplacian L_1 of the coauthorship complex.

Simplicial Laplacians. We are in this paper concerned with finite abstract simplicial complexes, although our method is applicable to a much broader setting, e.g. CW-complexes. In analogy with Hodge–de Rham theory [23], we define the *degree-i simplicial Laplacian* of a simplicial complex K as the linear map

$$\begin{aligned} &\mathcal{L}_i : C^i(K) \to C^i(K) \\ &\mathcal{L}_i = \mathcal{L}_i^{\text{up}} + \mathcal{L}_i^{\text{down}} = \delta^{i*} \circ \delta^i + \delta^{i-1} \circ \delta^{i-1*}, \end{aligned}$$

where δ^{i*} is the adjoint of the coboundary with respect to the inner product (typically the one making the indicator function basis orthonormal). In most practical applications, the coboundary can be represented as a sparse matrix B_i and the Laplacians can be efficiently computed as $L_i = B_i^\intercal B_i + B_{i-1} B_{i-1}^\intercal$. The matrices L_0 and B_0 are the classic graph Laplacian and incidence matrix. Note that the Laplacians carry valuable topological information about the complex: The kernel of the k-Laplacian is isomorphic to the k-(co)homology of its associated simplicial complex [24, 25][2].

Simplicial convolution. A convolutional layer is of the form $\psi \circ (f * \varphi_W)$, where $*$ denotes convolution, φ_W is a function *with small support* parameterized by learnable weights W, and ψ is some nonlinearity and bias. This formulation of CNNs lends itself to a spectral interpretation that we exploit to extend CNNs to a much more general setting.

[2] In other words, the number of zero-eigenvalues of the k-Laplacian corresponds to the number of k-dimensional holes in the simplicial complex.

Following [8] and motivated by the fact that the discrete Fourier transform of a real-valued function on an n-dimensional cubical grid coincides with its decomposition into a linear combination of the eigenfunctions of the graph Laplacian for that grid, we define the Fourier transform of real p-cochains on a simplicial complex with Laplacians $\mathcal{L}_p$ as

$$\begin{aligned}&\mathcal{F}_p : C^p(K) \to \mathbb{R}^{|K_p|}\\&\mathcal{F}_p(c) = \left(\langle c, e_1\rangle_p, \langle c, e_2\rangle_p, \ldots, \langle c, e_{|K_p|}\rangle_p\right),\end{aligned}$$

where the e_i's are the eigencochains of $\mathcal{L}_p$ ordered by eigenvalues $\lambda_1 \leq \cdots \leq \lambda_{|K_p|}$. The function $\mathcal{F}_p$ is invertible since $\mathcal{L}_p$ is diagonalizable; explicitly, if we write $U \operatorname{diag}(\Lambda) U^\intercal$ for a normalized eigendecomposition, the orthonormal matrices U and $U^\intercal$ represent $\mathcal{F}_p^{-1}$ and $\mathcal{F}_p$, respectively. This is the foundation for Barbarossa's development of signal processing on simplicial complexes [26].

Recall that on the function classes for which it is defined, the classical Fourier transform satisfies $\mathcal{F}(f * g) = \mathcal{F}(f)\mathcal{F}(g)$, where the right-hand side denotes pointwise multiplication. This will be our definition of convolution in the simplicial setting. Indeed, for cochains $c, c' \in C^p(K)$ we *define* their convolution as the cochain

$$c *_p c' = \mathcal{F}_p^{-1}\left(\mathcal{F}_p(c)\mathcal{F}_p(c')\right).$$

Within this framework, we are led to define a *simplicial convolutional layer* with input p-cochain c and weights W as being of the form

$$\psi \circ \left(\mathcal{F}_p^{-1}(\varphi_W) *_p c\right)$$

for some as of yet unspecified $\varphi_W \in \mathbb{R}^{|K_p|}$. To ensure the central property that a convolutional layer be localizing, we demand that φ_W be a low-degree polynomial in $\Lambda = (\lambda_1, \ldots, \lambda_{|K_p|})$, namely

$$\varphi_W = \sum_{i=0}^{N} W_i \Lambda^i = \sum_{i=0}^{N} W_i (\lambda_1^i, \lambda_2^i, \ldots, \lambda_{|K_p|}^i),$$

for small N. In signal processing parlance, one would say that such a convolutional layer *learns filters that are low-degree polynomials in the frequency domain*.

The reason for restricting the filters to be these low-degree polynomials is best appreciated when writing out the convolutional layer in a basis. Let L_p^i denote the i'th power of the matrix for $\mathcal{L}_p$ in, say, the standard basis for $C^p(K)$, and similarly for c. Then (ignoring ψ),

$$\mathcal{F}_p^{-1}(\varphi_W) *_p c = \sum_{i=0}^{N} W_i U \operatorname{diag}(\Lambda^i) U^\intercal c = \sum_{i=0}^{N} W_i \left(U \operatorname{diag}(\Lambda) U^\intercal\right)^i c = \sum_{i=0}^{N} W_i L_p^i c.$$

This is important for three reasons, like for traditional CNNs. First, the convolution can be efficiently implemented by N sparse matrix-vector multiplications: This reduces the computational complexity from $\mathcal{O}(|K_p|^2)$ to $\mathcal{O}(\xi|K_p|)$ where ξ is the density factor. Second, the number of weights to be learned is reduced from $\mathcal{O}(|K_p|)$ to $\mathcal{O}(1)$. Third, the operation is N-localizing in the sense that if two simplices σ, τ are more than N hops apart, then a degree-N convolutional layer does not cause interaction between $c(\sigma)$ and $c(\tau)$ in its output (see the supplementary material). Those local interactions (in the spatial domain) can be interpreted as message-passing between simplices.

3 Experimental results

As many real-world datasets contain missing values, missing data imputation is an important problem in statistics and machine learning [27, 28]. Leveraging the structure underlying the data, GNNs have recently proved to be a powerful tool for this task [29]. Extending this view to higher-dimensional structure, we evaluate the performance of SNNs in imputing missing data over simplicial complexes.

Data. A *coauthorship complex (CC)* [19] is a simplicial complex where a paper with k authors is represented by a $(k-1)$-simplex. The added subsimplices of the $(k-1)$-simplex are interpreted as collaborations among subsets of authors—a natural hierarchical representation that would be missed by the hypergraph representation of papers as hyperedges between authors. In general, a simplicial

complex representing n-fold interactions (e.g. between authors) can be constructed as the one-mode projection of a multipartite graph (e.g. a paper-author bipartite graph). The $(k-1)$-cochains are given by the number of citations attributed to the given collaborations of k authors. See Figure 1 and 4, and the supplementary material for details. We sampled (see the supplementary material) two coauthorship complexes—CC1 and CC2, see Table 2 for statistics—from the Semantic Scholar Open Research Corpus [30], a dataset of over 39 million research papers with authors and citations.

Method. We evaluated the performance of the SNNs on the task of imputing missing citations on the k-cochains (for $k = 0, 1, 2$) of the extracted coauthorship complexes. As in a typical pipeline for this task [28], missing values are artificially introduced by replacing a portion of the values with a constant. Specifically, given a fixed coauthorship complex, missing data is introduced at random on the k-cochains at 5 rates: $10\%, 20\%, 30\%, 40\%$, and 50%. The SNN is given as input the k-cochains on which missing citations are substituted by the median of known citations (as a reasonable first guess) and is trained to minimize the L_1 norm over known citations. We trained SNNs made of 3 layers with 30 convolutional filters of degree $N = 5$ with Leaky ReLu for 1000 iterations with the Adam optimizer and a learning rate of 10^{-3}.

Results. Figure 2 shows the mean accuracy and absolute error distribution (see the supplementary material for definitions) of the SNN in inputing missing citations on CC1. Observe that the distribution of the prediction error accumulates close to zero.

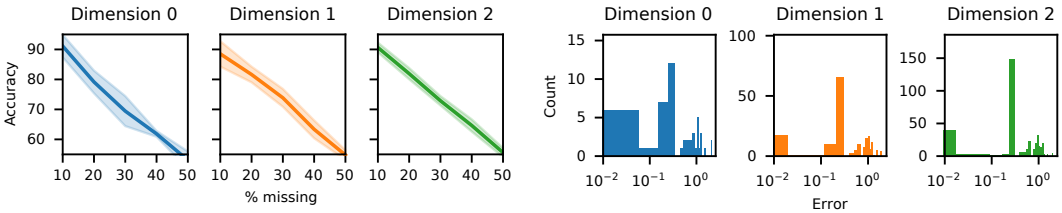


Figure 2: Performance of SNNs. Left: Mean accuracy $\pm$ standard deviation over 5 samples in imputing missing citations on CC1. Right: Absolute error distribution over 1 sample for 40% missing citations on CC1.

Table 1 shows the performance of two baselines: missing values inferred as (i) the mean or median of all known values, and (ii) the mean of the $(k-1)$ and $(k+1)$ neighboring simplices. SNNs well outperform these baselines. Comparison with stronger imputation algorithms is left for future work.

To demonstrate that our filters transfer across complexes, we evaluated how accurately an SNN trained on one coauthorship complex can impute missing citations on a different complex. We found that when imputing citations on CC1, a SNN trained on CC2 is almost as good as one trained on CC1 (compare Figures 2 and 3). We expect this result as coauthorship complexes share a similar structure, and the same process underlies the generation of citations across coauthorship complexes.

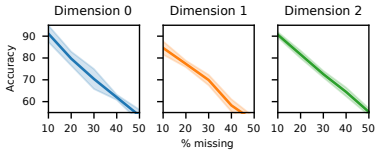


Figure 3: Performance on CC1 with an SNN trained on CC2.

Method	Dimension 0	Dimension 1	Dimension 2
Global Mean	3.30 ± 0.82	5.75 ± 1.28	2.96 ± 0.49
Global Median	7.78 ± 2.70	10.44 ± 1.00	12.50 ± 0.63
Neighbors Mean	11.88 ± 5.29	24.15 ± 1.85	27.38 ± 1.18

Table 1: Performance of baselines: mean accuracy $\pm$ standard deviation over 5 samples for 30% missing citations on CC1.

4 Conclusion and future work

We introduced a mathematical framework to design neural networks for data that live on simplicial complexes and provided preliminary results on their ability to impute missing data. Future work might include: (i) comparing SNNs with state-of-the-art imputation algorithms, (ii) using SNNs to solve vector field problems, (iii) generalizing coarsening and pooling to simplicial complexes, (iv) using boundaries and coboundaries to mix data structured by relationships of different dimensions, and (v) studying the expressive power of SNNs.

Unrelated to the simplicial nature of this work, we would like to emphasize how the spectral language was key to developing and even formulating our method. On homogeneous spaces, convolutions are defined as inner-products with filters shifted by the actions of a symmetry group of the space. They are the most general shift-invariant linear operators. On non-homogeneous spaces however, the spectral language yields generalized convolutions which are inner-products with *localized* filters [31, Sec. 2.4]. Those too are invariant to any symmetry the space might have. Convolutions exploit the space's structure to reduce learning complexity by sharing learnable weights through shifts and localizations of filters.

Acknowledgments and disclosure of funding

S.E. and G.S. were supported by the Swiss National Science Foundation grant number 200021_172636, and would like to thank K. Hess for valuable discussions.

References

[1] Taco S Cohen et al. "Spherical CNNs". In: *International Conference on Learning Representations (ICLR)*. 2018. arXiv: 1801.10130.

[2] Carlos Esteves et al. "Learning SO(3) Equivariant Representations with Spherical CNNs". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018. arXiv: 1711.06721.

[3] Michaël Defferrard et al. "DeepSphere: a graph-based spherical CNN". In: *International Conference on Learning Representations (ICLR)*. 2020. URL: https://openreview.net/forum?id=B1e3OlStPB.

[4] Taco Cohen and Max Welling. "Group Equivariant Convolutional Networks". In: vol. 48. Proceedings of Machine Learning Research. 2016, pp. 2990–2999.

[5] Risi Kondor and Shubhendu Trivedi. "On the Generalization of Equivariance and Convolution in Neural Networks to the Action of Compact Groups". In: (2018), pp. 2747–2755. arXiv: 1802.03690.

[6] Daniel E Worrall et al. "Harmonic networks: Deep translation and rotation equivariance". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 5028–5037.

[7] Joan Bruna et al. "Spectral Networks and Locally Connected Networks on Graphs". In: *International Conference on Learning Representations (ICLR)*. 2014. arXiv: 1312.6203.

[8] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. "Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering". In: *Advances in Neural Information Processing Systems 29*. 2016, pp. 3844–3852.

[9] M. M. Bronstein et al. "Geometric Deep Learning: Going beyond Euclidean data". In: *IEEE Signal Processing Magazine* 34.4 (2017), pp. 18–42.

[10] Z. Wu et al. "A Comprehensive Survey on Graph Neural Networks". In: *IEEE Transactions on Neural Networks and Learning Systems* (2020), pp. 1–21.

[11] Rickard Brüel Gabrielsson et al. "A Topology Layer for Machine Learning". In: vol. 108. Proceedings of Machine Learning Research. 2020, pp. 1553–1563.

[12] C. Hofer, R. Kwitt, and M. Niethammer. "Learning Representations of Persistence Barcodes". In: *J. Mach. Learn. Res.* 20 (2019), 126:1–126:45.

[13] Bastian Rieck et al. "Neural Persistence: A Complexity Measure for Deep Neural Networks Using Algebraic Topology". In: International Conference on Learning Representations. 2019.

[14] Gunnar Carlsson. "Topology and Data". In: *Bull. Amer. Math. Soc.* 46.2 (2009), pp. 255–308.

[15] Frederic Chazal and Bertrand Michel. *An introduction to Topological Data Analysis: fundamental and practical aspects for data scientists*. 2017. arXiv: 1710.04019.

[16] Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Soc., 2010.

[17] Robert Ghrist. "Barcodes: the persistent topology of data". In: *Bulletin of the American Mathematical Society* 45.1 (2008), pp. 61–75.

[18] Terrence J. Moore et al. "Analyzing collaboration networks using simplicial complexes: A case study". In: *Proceedings IEEE INFOCOM Workshops* (2012), pp. 238–243.

[19] Alice Patania, Giovanni Petri, and Francesco Vaccarino. "The shape of collaborations". In: *EPJ Data Science* 6.1 (2017).

[20] Yifan Feng et al. "Hypergraph Neural Networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019.

[21] F. Monti, K. Otness, and M. M. Bronstein. "MotifNet: a motif-based Graph Convolutional Network for directed graphs". In: *2018 IEEE Data Science Workshop (DSW)*. 2018, pp. 225–228.

[22] Ze Xiao and Yue Deng. "Graph embedding-based novel protein interaction prediction via higher-order graph convolutional network". In: *PLoS One* 15 (2020).

[23] Ib Madsen and Jørgen Tornehave. *From calculus to cohomology: de Rham cohomology and characteristic classes*. Cambridge University Press, 1997.

[24] Beno Eckmann. "Harmonische Funktionen und Randwertaufgaben in einem Komplex". In: *Commentarii Mathematici Helvetici* 17.1 (1944), pp. 240–255.

[25] Danijela Horak and Jürgen Jost. "Spectra of combinatorial Laplace operators on simplicial complexes". In: *Advances in Mathematics* 244 (2013), pp. 303–336.

[26] Sergio Barbarossa, Stefania Sardellitti, and Elena Ceci. "Learning from signals defined over simplicial complexes". In: *2018 IEEE Data Science Workshop (DSW)*. IEEE. 2018, pp. 51–55.

[27] Roderick J A Little and Donald B Rubin. *Statistical Analysis with Missing Data*. USA: John Wiley & Sons, Inc., 1986. ISBN: 0471802549.

[28] Fulufhelo Nelwamondo, Shakir Mohamed, and Tshilidzi Marwala. "Missing Data: A Comparison of Neural Network and Expectation Maximisation Techniques". In: *Current Science* 93 (May 2007).

[29] Indro Spinelli, Simone Scardapane, and Aurelio Uncini. "Missing data imputation with adversarially-trained graph convolutional networks". In: *Neural Networks* 129 (2020), pp. 249–260.

[30] Waleed Ammar et al. "Construction of the Literature Graph in Semantic Scholar". In: *NAACL*. 2018. URL: https://www.semanticscholar.org/paper/09e3cf5704bcb16e6657f6ceed70e93373a54618.

[31] Nathanaël Perraudin et al. "DeepSphere: Efficient spherical convolutional neural network with HEALPix sampling for cosmological applications". In: *Astronomy and Computing* 27 (2019), pp. 130–146.

A Supplementary material

Simplicial distance and the localizing property of the Laplacian. Suppose that σ and τ are p-simplices for which $(\nu_0, \nu_1, \ldots, \nu_d)$ is the shortest sequence of p-simplices with the property that $\nu_0 = \sigma$, $\nu_d = \tau$, and each ν_i shares a face or a coface with ν_{i-1}, and a face or a coface with ν_{i+1}. We say that d is the *simplicial distance* between σ and τ. Then for all $N < d$, the entry of L_p^N corresponding to σ and τ is 0, and so the filter does not cause interaction between $c(\sigma)$ and $c(\tau)$. This is analogous to a size-d ordinary CNN layer not distributing information between pixels that are more than d pixels apart. We will refer to N as the *degree* of the convolutional layer, but one may well wish to keep in mind the notion of *size* from traditional CNNs.

Simplicial complexes as the projections of bipartite graphs. Given a bipartite graph X-Y, the simplicial projection on Y is the simplicial complex whose $(k-1)$-simplices are the sets of k vertices in Y that have at least one common neighbor in X. Cochains on the simplicial projection are naturally given by weights on X: Given any $(k-1)$-simplex $[y_1, \ldots, y_k]$ and its neighboring vertices $\{x_1, \ldots, x_j\} \subseteq X$, one can define a $(k-1)$-cochain as $\phi(\{x_1, \ldots, x_j\})$, for any function $\phi : \mathcal{P}(X) \to \mathbb{R}$. In our coauthorship application, ϕ is the sum and the weight of a paper is the number of times it is cited. See Figure 4.

Dimension	0	1	2	3	4	5	6	7	8	9	10
CC1	352	1474	3285	5019	5559	4547	2732	1175	343	61	5
CC2	1126	5059	11840	18822	21472	17896	10847	4673	1357	238	19

Table 2: Number of simplices of the two coauthorship complexes sampled from Semantic Scholar.

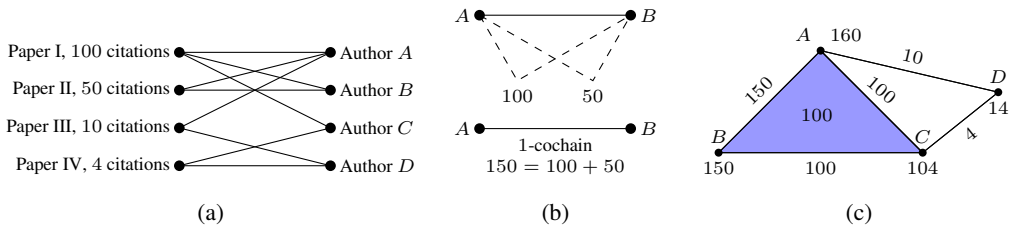


Figure 4: Constructing a simplicial complex and its cochain from a bipartite graph. (a) Paper-author bipartite graph (same data as in Figure 1). (b) The 1-simplex $[A, B]$ is included in the coauthorship complex since authors A and B collaborated on papers I and II. The 1-cochain on $[A, B]$ is given by the sum of their common papers' citations. (c) Resulting coauthorship complex with cochains.

Sampling papers. From the Semantic Scholar Open Research Corpus [30], we excluded papers with fewer than 5 citations or more than 10 authors. To sample a CC, we sampled 80 papers (corresponding to maximal simplices in the CC) by performing a random walk (of length 80, from a randomly chosen starting paper) on the graph whose vertices represent papers and edges connect papers sharing at least one author.

Mean accuracy and absolute error. A missing value is considered to be correctly imputed if the imputed value differs by at most 10% from the true value. The *accuracy* is the percentage of missing values that has been correctly imputed and the *absolute error* is the magnitude of the difference between the imputed and true value. For each rate of missing values, we compute the *mean accuracy* $\pm$ standard deviation over 5 samples with randomly damaged portions.