Farielberry Lab

Deep Learning for Genomic Prediction

FACULTAD DE INGENIERÍA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Agenda

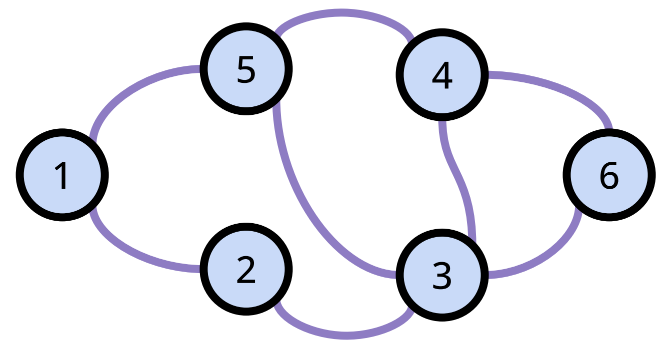
1. Grafos
2. Intuición grafos - CNN
3. Matrices de adyacencia
4. Graph Neural Networks
5. Algunos resultados

Diapositivas basadas en el trabajo de Federico Larroca, Paola Bemolen, Juan Elenter, Guillermo Etchebarne, Ignacio Hounie y Micaela Long.

Graphs

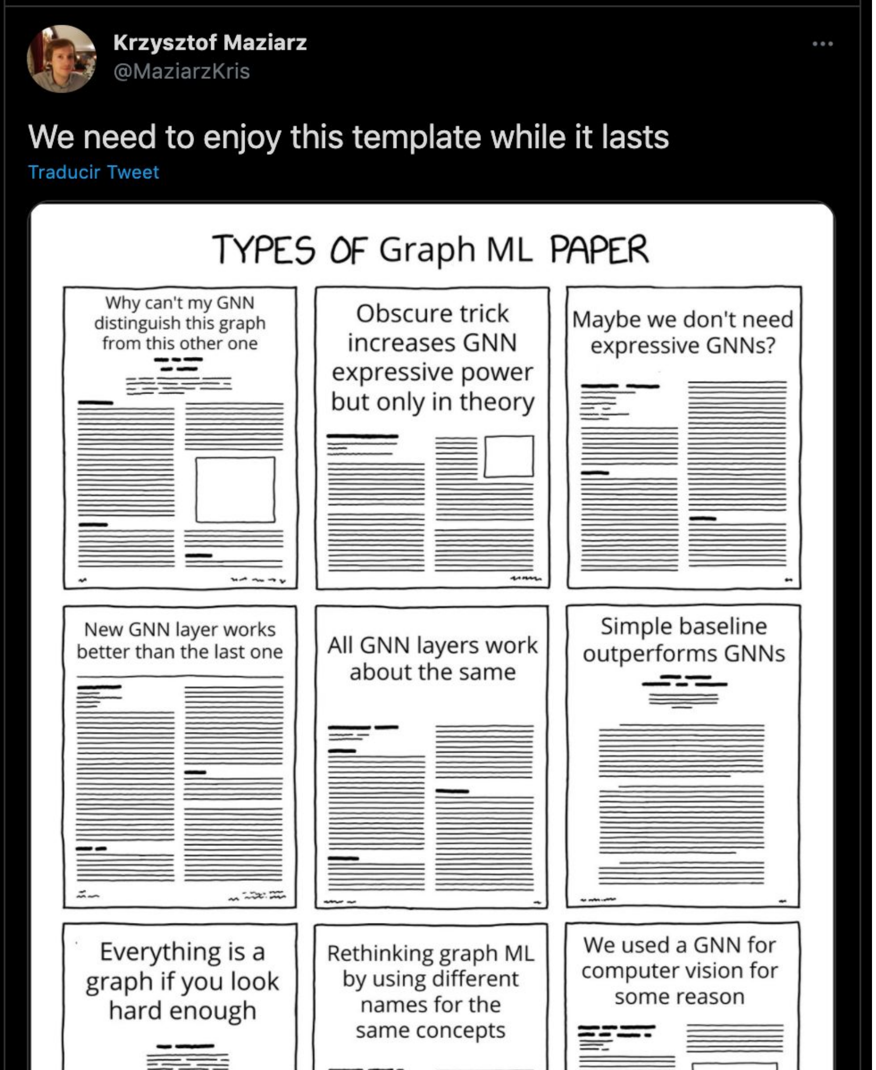
We call a graph $\mathcal{G}(\mathbb{V}, \mathbb{E})$ a set $\mathbb{V}$ of vertices or nodes connected by a set $\mathbb{E}$ of edges. The elements of $\mathbb{E} \subset \mathbb{V} \times \mathbb{V}$ are unordered pairs of distinct vertices $(u, v), u, v, \in \mathbb{V}$

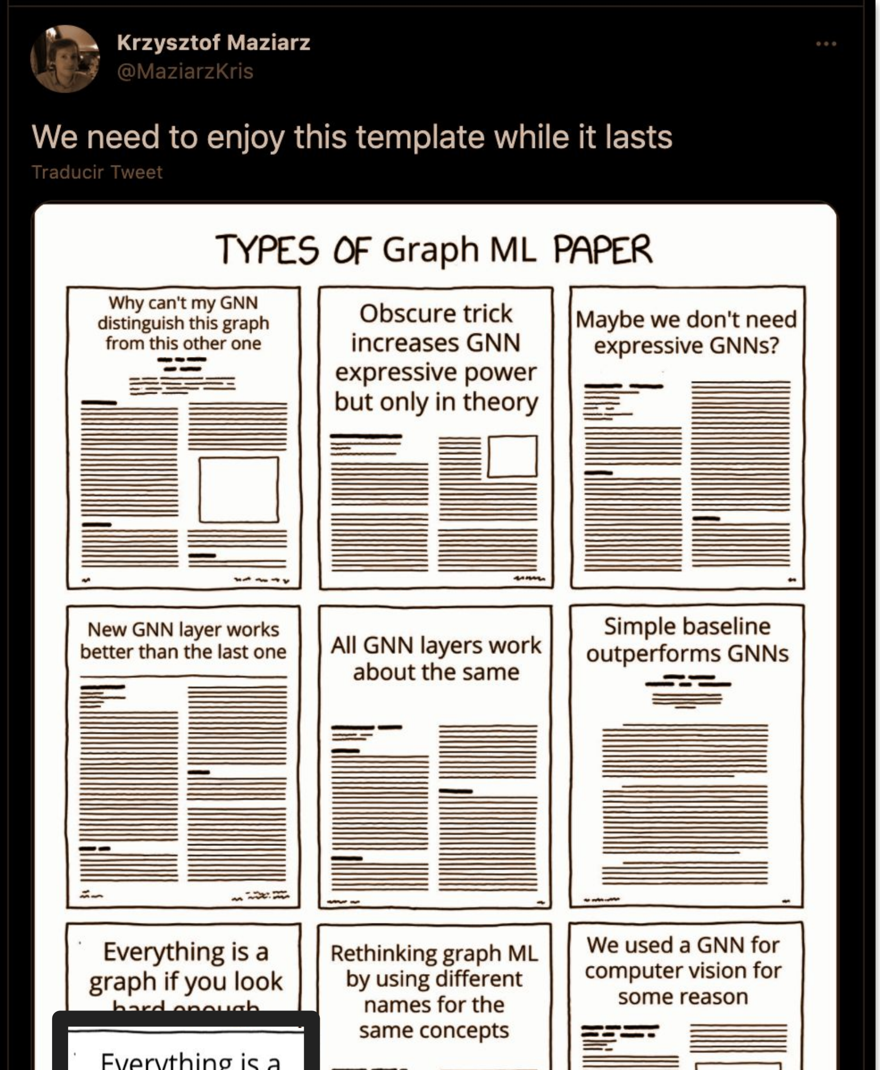
Usually a graph $\mathcal{G}$ has order$N_E = |\mathbb{E}|$ and size $N_V = |\mathbb{V}|$.

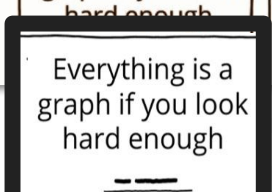


Vertices $\mathbb{V} = \{1, 2, 3, 4, 5, 6\}$

Edges $\mathbb{E} = \{(1, 2), (1, 5), (2, 3), (3, 4), (3, 5), (4, 5), (4, 6)\}$


Krzysztof Maziarz
@MaziarzKris
We need to enjoy this template while it lasts
Traducir Tweet
TYPES OF Graph ML PAPER
Why can't my GNN distinguish this graph from this other one
Obscure trick increases GNN expressive power but only in theory
Maybe we don't need expressive GNNs?
New GNN layer works better than the last one
All GNN layers work about the same
Simple baseline outperforms GNNs
Everything is a graph if you look hard enough
Rethinking graph ML by using different names for the same concepts
We used a GNN for computer vision for some reason





Todo es un grafo

Si mirás lo suficiente

Academic networks: co-cites between Nature's papers

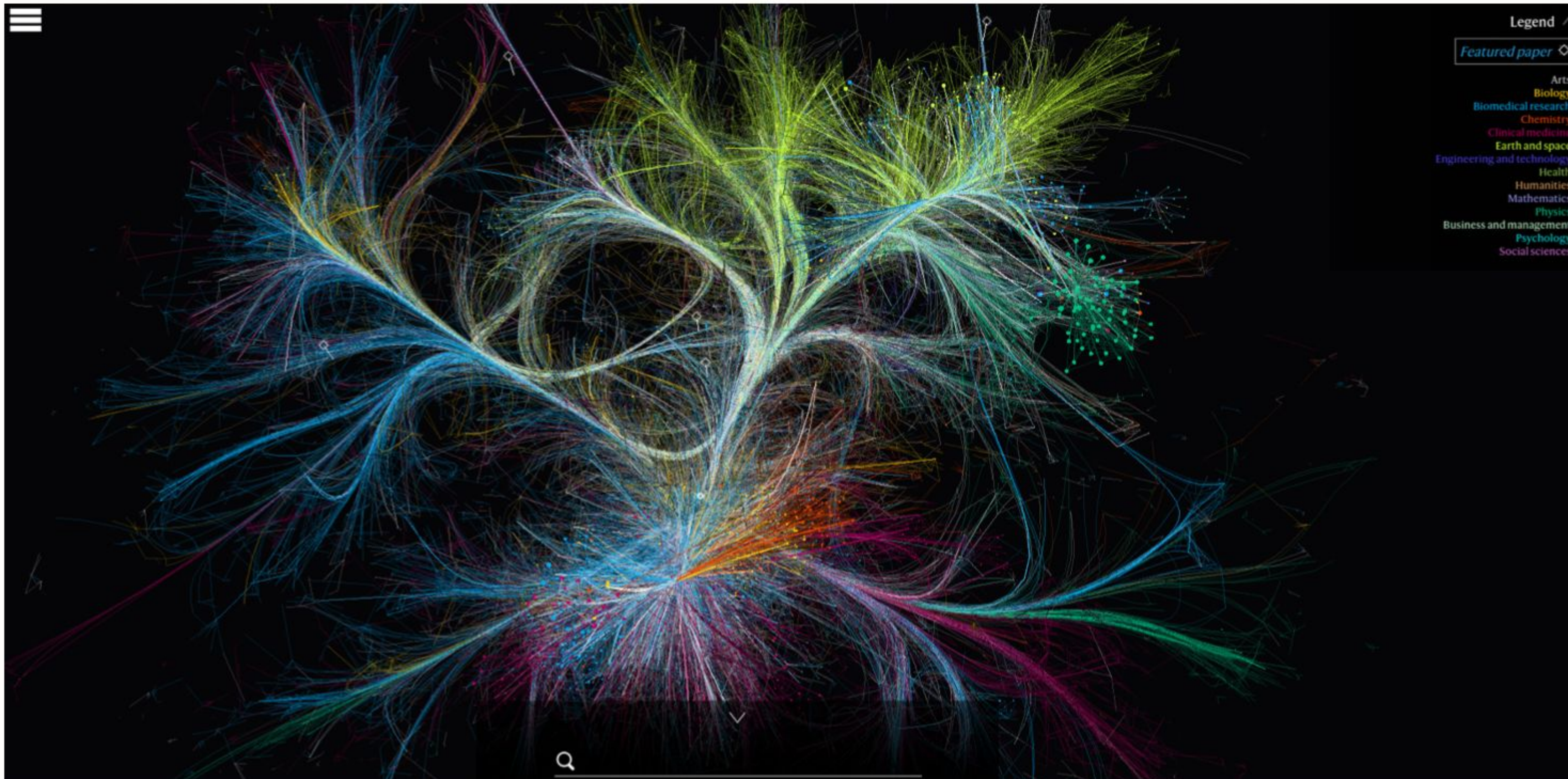


Figure: Grafo de co-citas entre artículos en Nature (extraído de
`https://www.nature.com/immersive/d41586-019-03165-4/index.html`

Ecologic networks (directed edges)

Redes ecológicas (aristas dirigidas)

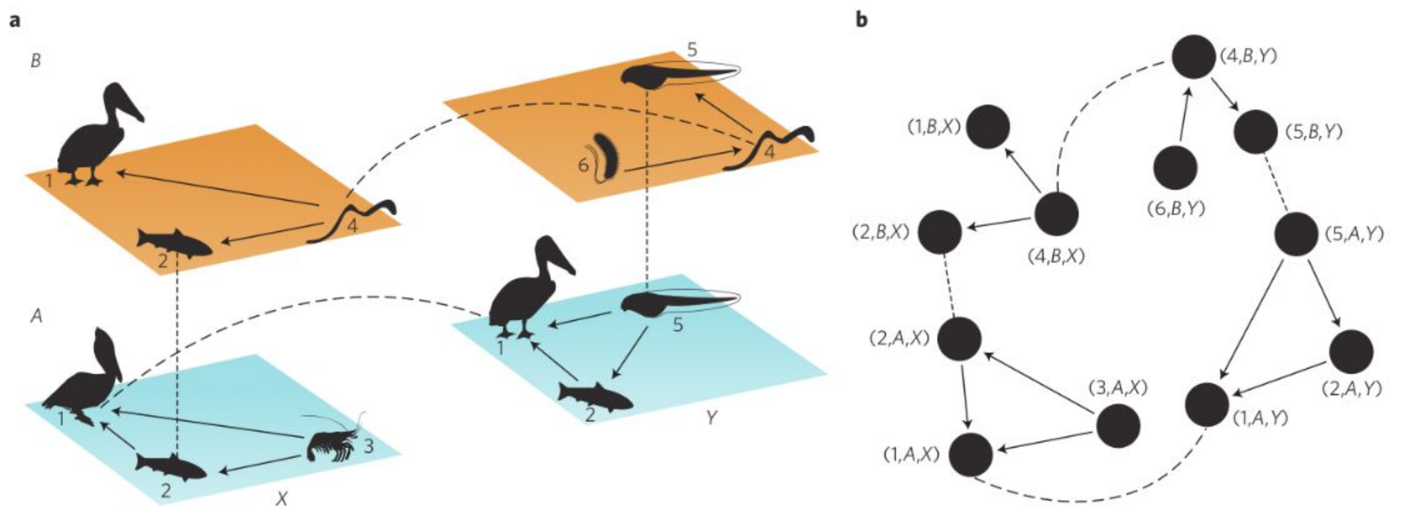
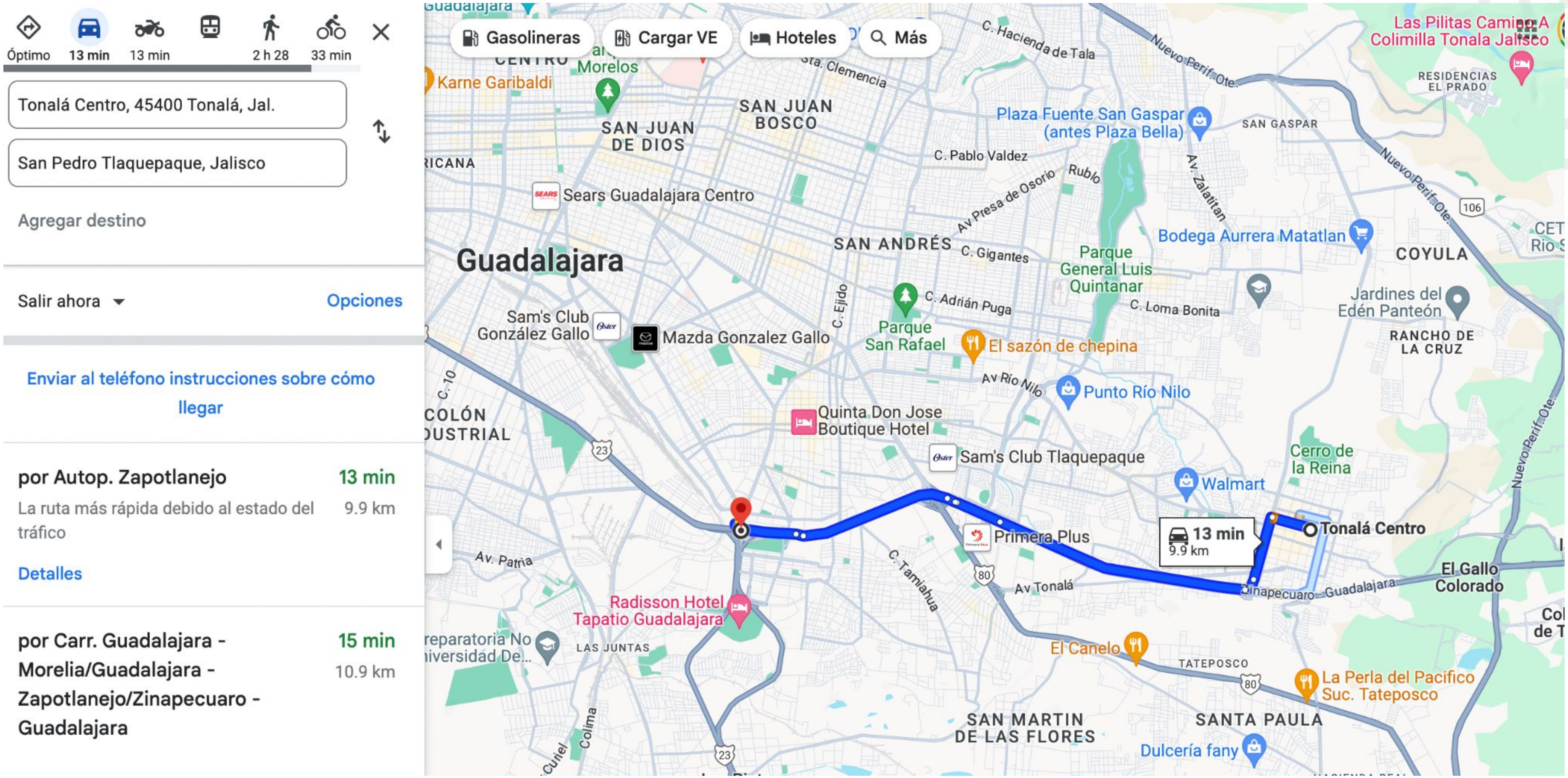
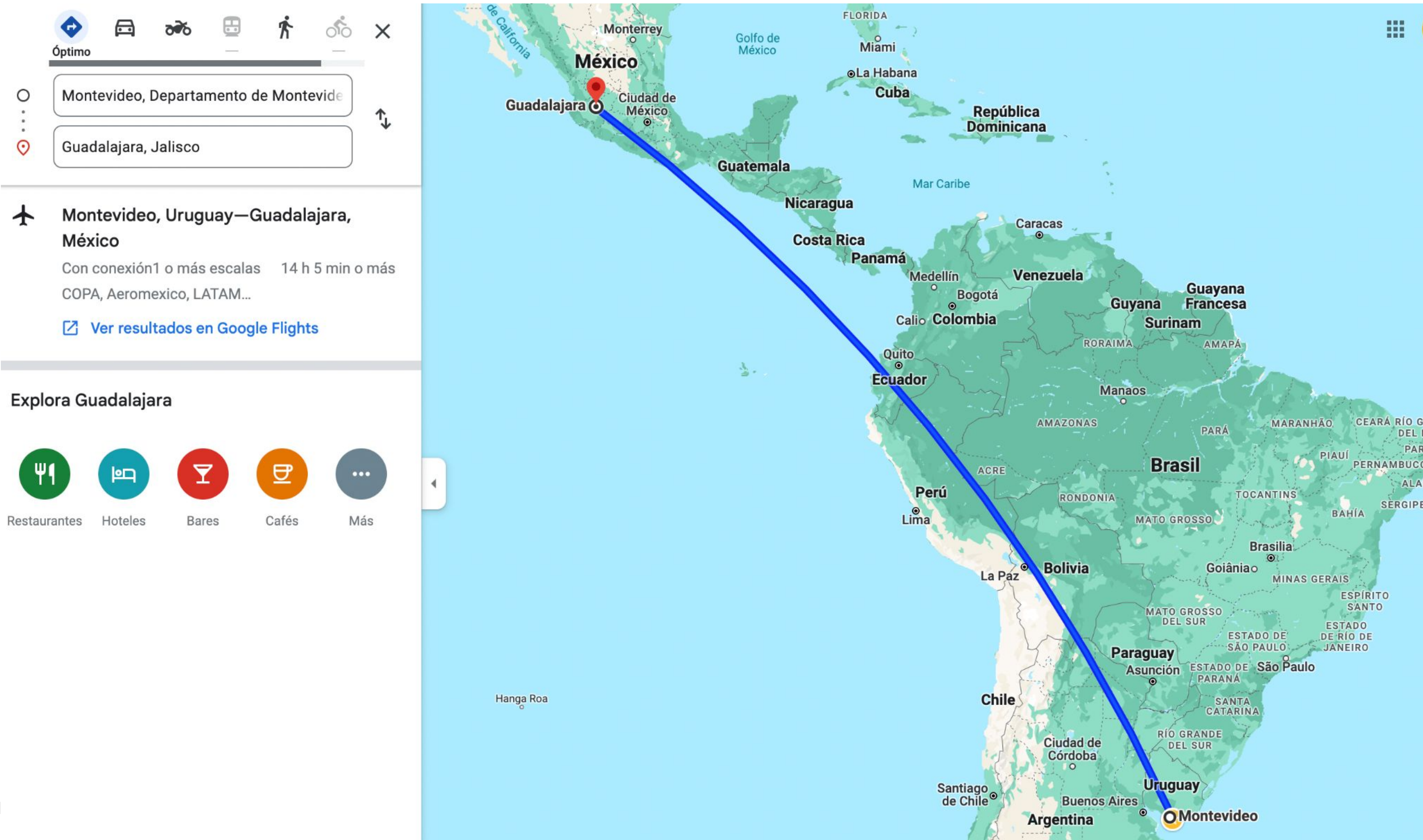


Figure: Extraído de S. Pilosof et al. “The multilayer nature of ecological networks”, Nature Ecology & evolution, 2017

Nodos: Lugares de una ciudad. Aristas: tiempo para llegar en ...



Nodos: Ciudades por el mundo. Aristas tiempo para llegar en ...

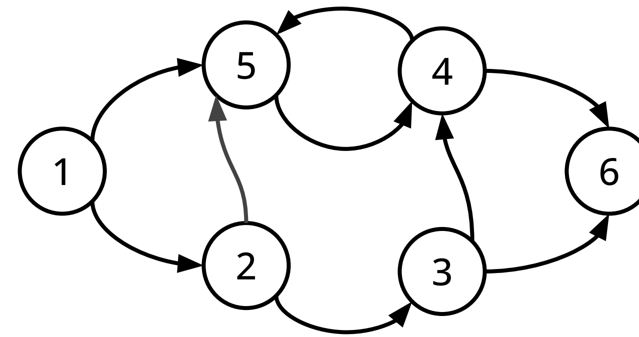


Vertices and edges in real networks

Network	Node	Edge
Internet	Computer/router	Cable/wireless link
WWW	Web page	Hyperlink
Food web	Species	Predation
Gene-regulatory web	Gene	Regulation of expression
Friendship web	Person	Friendship or acquaintance
Power grid	Substation	Transmission line
Affiliation web	Person or club	Membership
Citation web	Article/patent	Citation
Neural web	Neuron	Synapse

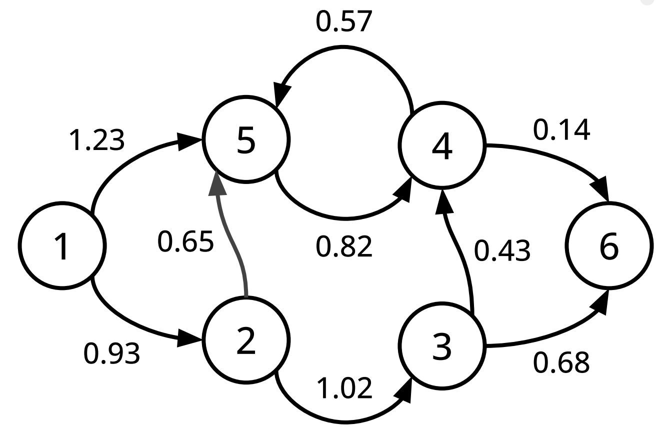
Directed graphs (digraph)

- In directed graphs, the elements of $\mathbb{E}$ are ordered pairs $(u, v), u, v, \in \mathbb{V}$
  - implies that the edge (u, v) is distinct from the edge (v, u)
  - the directed edges are called **semiarcs** or **arcs**
- Digraphs: directed graphs (u, v)
  - By convention, the arc leaves u and points to v (arcs are represented by arrows).
- Example: non-symmetric relationships such as followers on Twitter or citations in articles.



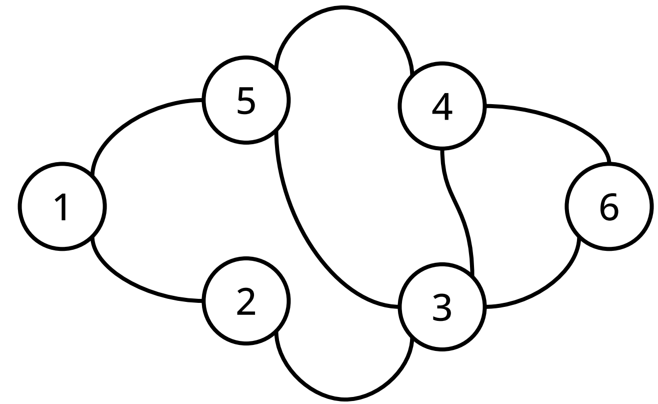
Weighted graphs

- Many times the vertices or edges have associated numerical values
  - such graphs are called weighted graphs.
- The values may correspond to measures of a defined process in the network.
  - Example: traffic on a link, number of cars on a certain route, node/person infected or not by COVID, etc.
- Multi-edges are generally encoded using weights.



Adjacency relation

- Necessary concept to study the connectivity of a graph.
- Adjacency is a very simple local concept.
  - two vertices are adjacent if they are connected by an edge.
  - two edges are connected if they share a vertex.

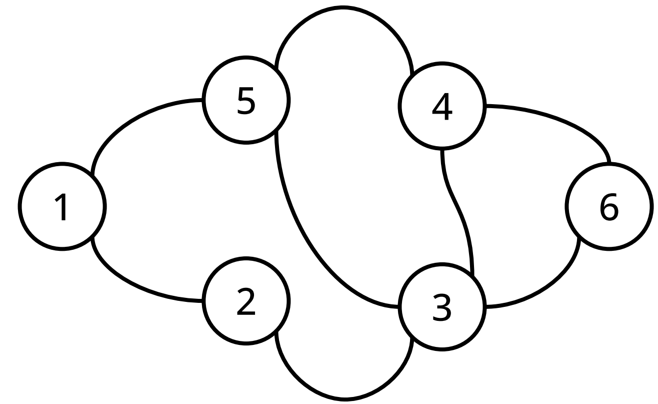


- Vertices 1 and 5 are adjacent.
- Vertices 2 and 4 are not adjacent.
- Edge (1,2) is adjacent to edge (1,5), but not adjacent to (4,6).

Degree of a vertex

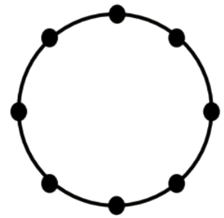
An edge (u,v) is incident to the vertices u and v.

The degree d_v of a vertex v of a graph is the number of edges that are incident to the vertex

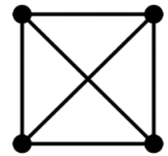


$$d_1 = 2 \quad d_4 = 3$$
$$d_2 = 2 \quad d_5 = 3$$
$$d_3 = 4 \quad d_6 = 2$$

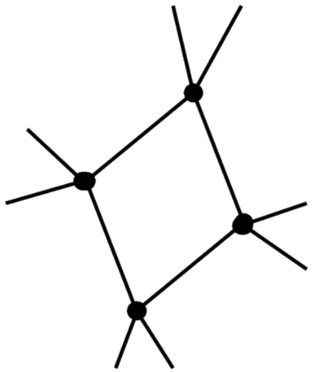
Regular graph



- A regular graph of order d has all its vertices of degree d
- The complete graph is (n-1)-regular.
  - Cycles are 2-regular (sub)-graphs.

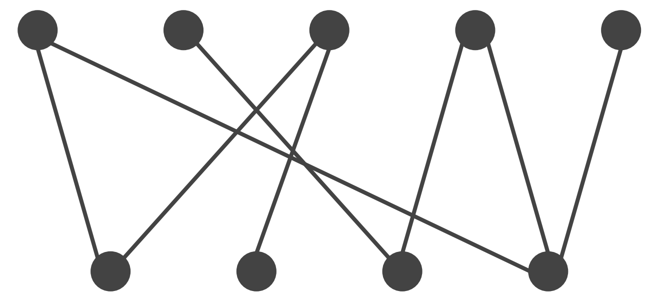


- Regular graphs appear frequently in:
  - physics and chemistry, in the study of crystal structures
  - geo-spatial references and pixel-models in image processing



Bipartite graph (bi-graph)

A bipartite graph also called a bi-graph, is a graph whose vertices can be divided into two disjoint and independent sets V1 and V2, that is, every edge connects a vertex in V1 to one in V2.



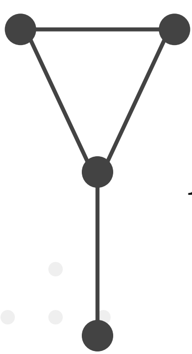
Adjacency matrix

- Algebraic graph theory
- How to represent a graph with a matrix? (Defines the "model")

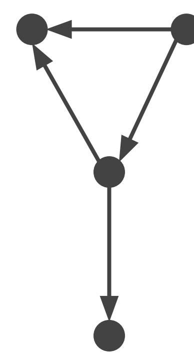
Given a graph G(V,E) the **adjacency matrix** is a binary matrix A such that

$$a_{ij} = \begin{cases} 1 & \text{if } (i,j) \in \mathbb{E} \\ 0 & \text{otherwise} \end{cases}.$$

- Symmetric for undirected graphs.
- Weighted graphs have $a_{ij} = \omega_{ij}$.

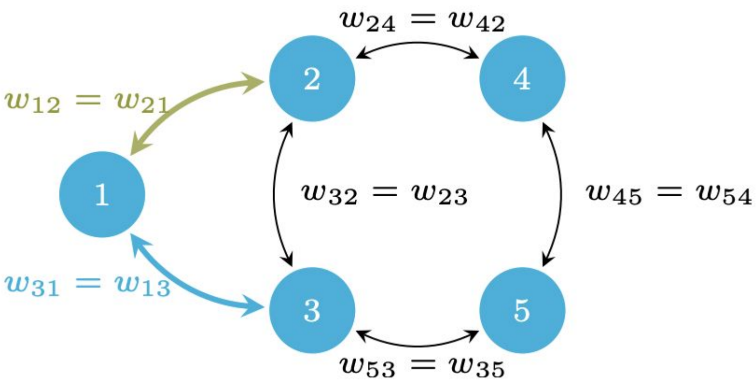


$$A_u = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

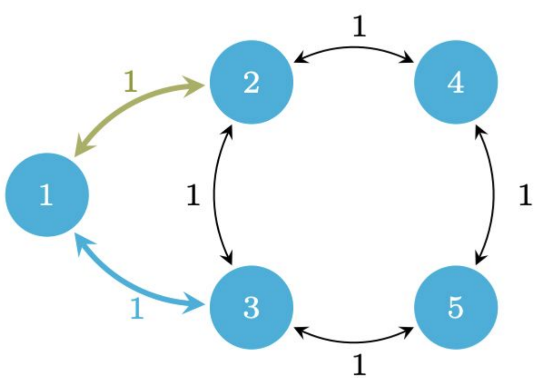


$$A_d = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

An Adjacency matrix, a model



$$A = \begin{bmatrix} 0 & w_{21} & w_{31} & 0 & 0 \\ w_{12} & 0 & w_{32} & w_{42} & 0 \\ w_{13} & w_{23} & 0 & 0 & w_{54} \\ 0 & 0 & w_{35} & w_{45} & 0 \end{bmatrix}$$

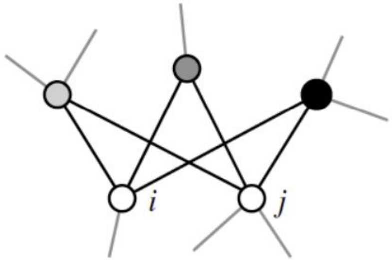


$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Cosine similarity

Number of neighbors in common

$$\eta_{ij} = \sum_k A_{ik}A_{kj} = \left(A^2\right)_{ij} = < A_{ik}, A_{kj} >$$



$$\eta_{ij} = 3$$

Cosine similarity between nodes:

$$\sigma_{ij} = \frac{\eta_{ij}}{\sqrt{d_i}\sqrt{d_j}}$$

$$\sigma_{ij} = \frac{3}{2\sqrt{5}}$$

Properties

- Symmetric
- It is the angle between normalized vectors
- $0 \leq \sigma_{ij} \leq 1 \forall i, j \in V$
- $\sigma_{ii} = 1$

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Pearson correlation

Pearson correlation coefficient
between rows.

$$r_{ij} = \text{corr}(a_{i\cdot}, a_{j\cdot}) = \frac{\sum_k (a_{ik} - \bar{a}_i) \sum_k (a_{jk} - \bar{a}_j)}{\sqrt{\sum_k (a_{ik} - \bar{a}_i)^2} \sqrt{\sum_k (a_{jk} - \bar{a}_j)^2}}$$

Properties

- Symmetric
- It is the angle between normalized vectors
- $-1 \leq r_{ij} \leq 1 \quad \forall i, j \in \mathbb{V}$
- $r_{ii} = 1$

$$A = \begin{bmatrix} 0 & w_{21} & w_{31} & 0 & 0 \\ w_{12} & 0 & w_{32} & w_{42} & 0 \\ w_{13} & w_{23} & 0 & 0 & w_{54} \\ 0 & 0 & w_{35} & w_{45} & 0 \end{bmatrix}$$

Adjacency matrix properties

- Number of edges

$$\sum_{i,j} a_{ij} = 2N_E$$

- Node degree

$$\sum_{j} a_{ij} = d_i$$

- For di-graphs A is not symmetric

$$\sum_{j} a_{ij} = d_i^{\text{out}}$$

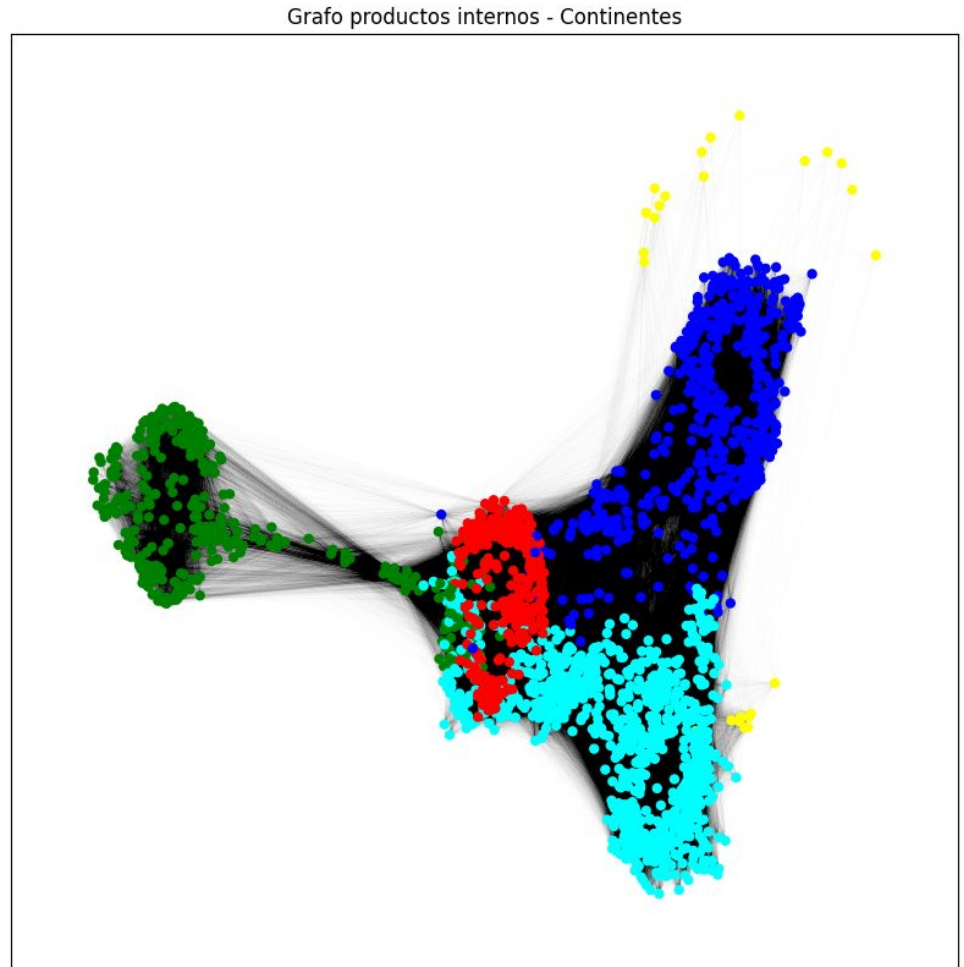
$$\sum_{i} a_{ij} = d_j^{\text{in}}$$

Embedding (reducción de dimensionalidad)

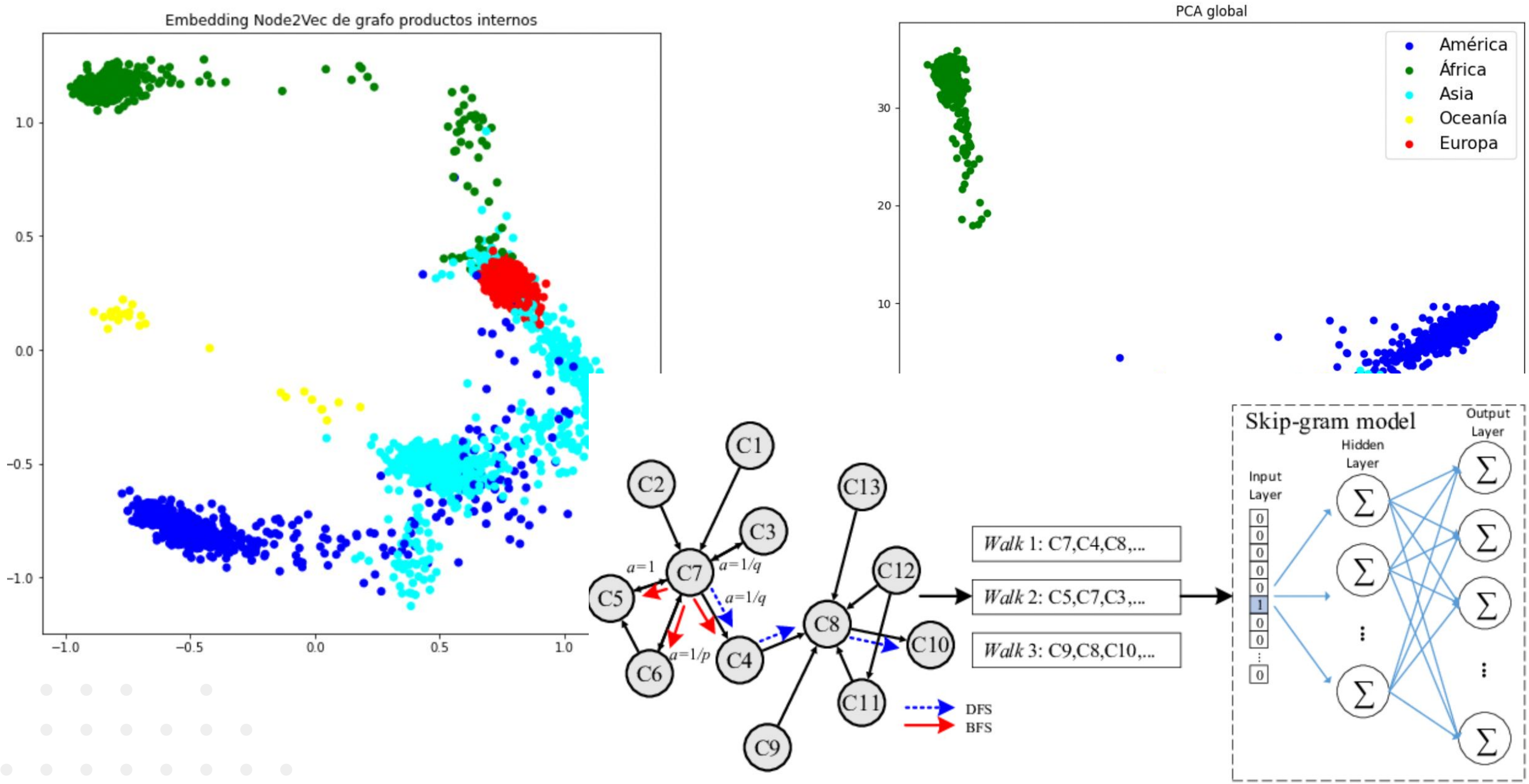
Dot product graph

Difficulty: Choose the threshold

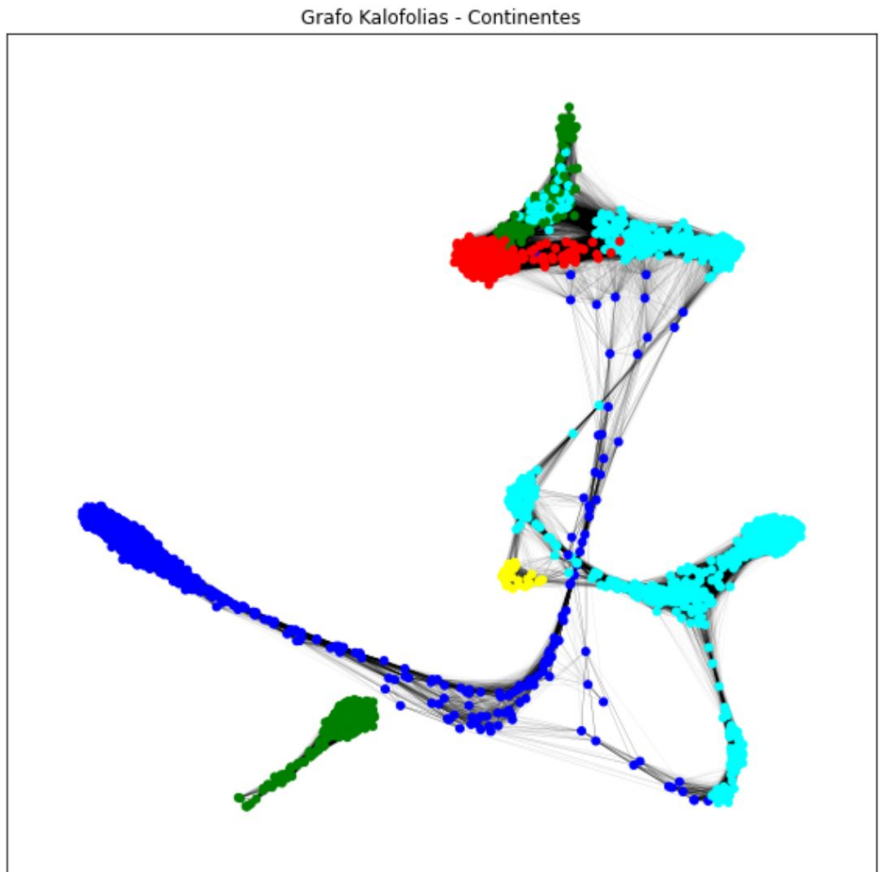
- Too high -> too sparse
- Too solow -> everybody is connected to everybody



Using node2vec (an autoencoder based on Word2vec)



Some art



More art

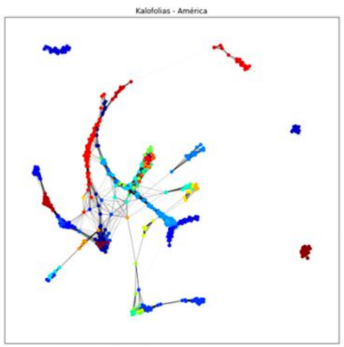


Figure: Kalofolias América

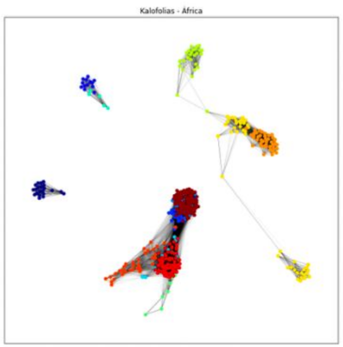


Figure: Kalofolias Africa

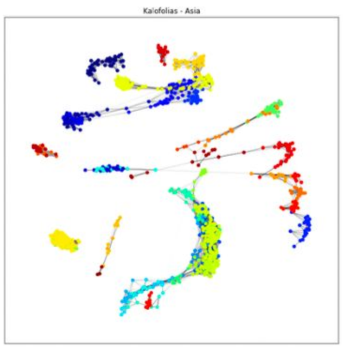


Figure: Kalofolias Africa

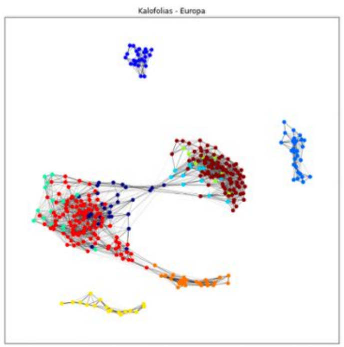
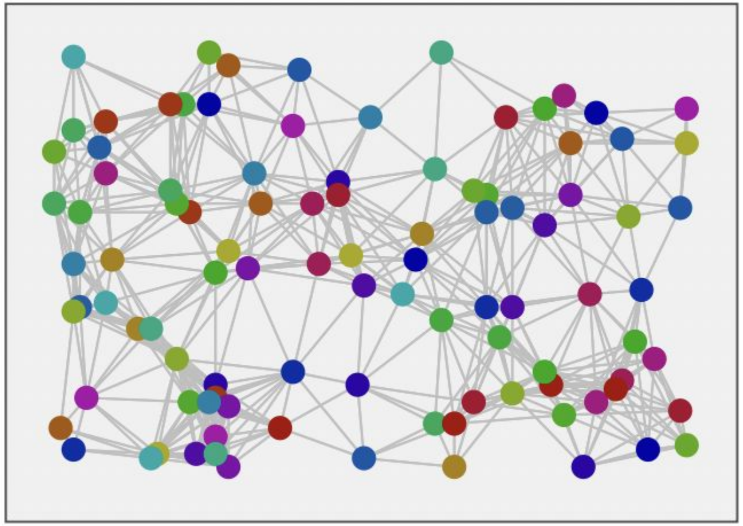
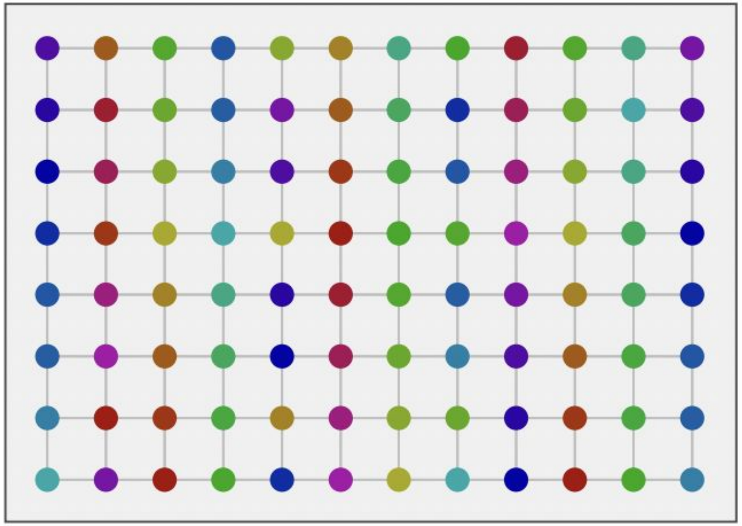


Figure: Kalofolias Africa

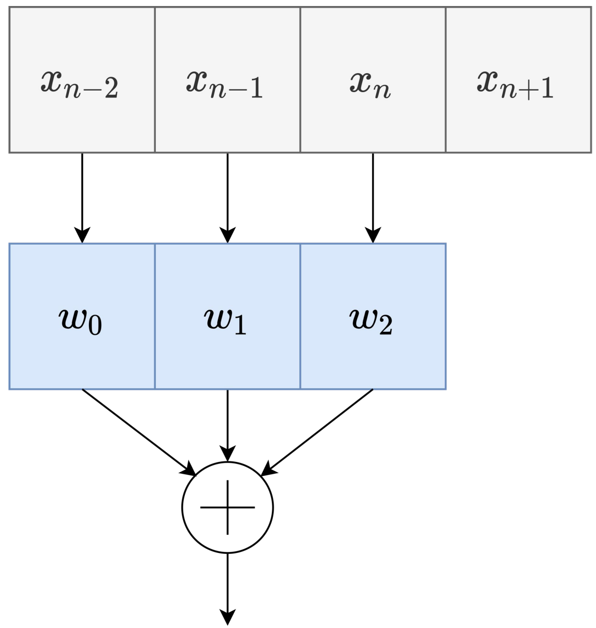
Graph Neural Networks (GNN)

CNNs: we know how to do it in "ordered poins"





Flashback: the convolution

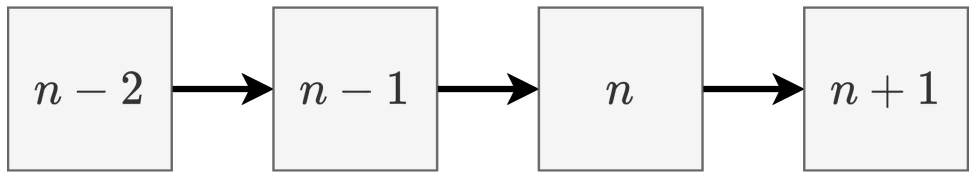


$$z_n = \sum_{m=-\infty}^{\infty} x_m w_{n-m}$$

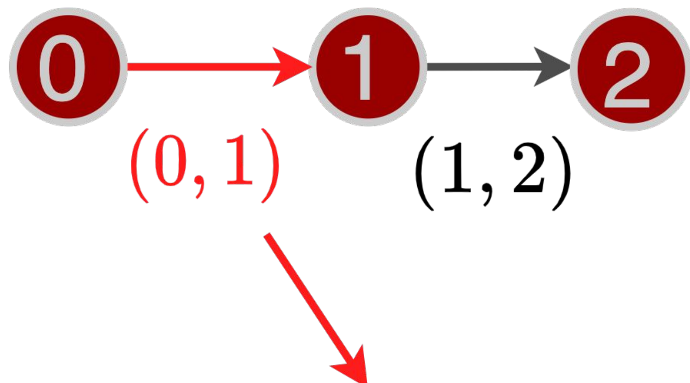
Convolution in time series

x_{n-2}	x_{n-1}	x_n	x_{n+1}

Convolution

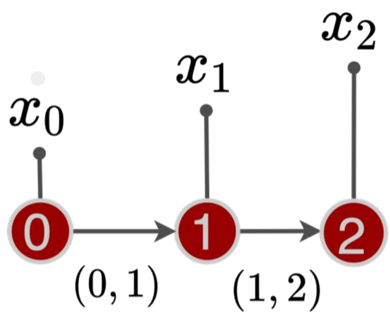


Matriz de Adyacencia

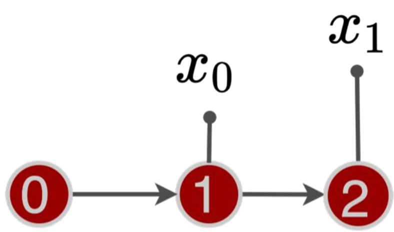


$$S_{ij} = \begin{cases} 1 & \text{si } (i,j) \in \mathcal{E} \\ 0 & \text{otro caso} \end{cases}$$

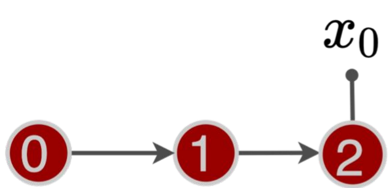
$$S = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$



$$S^0\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

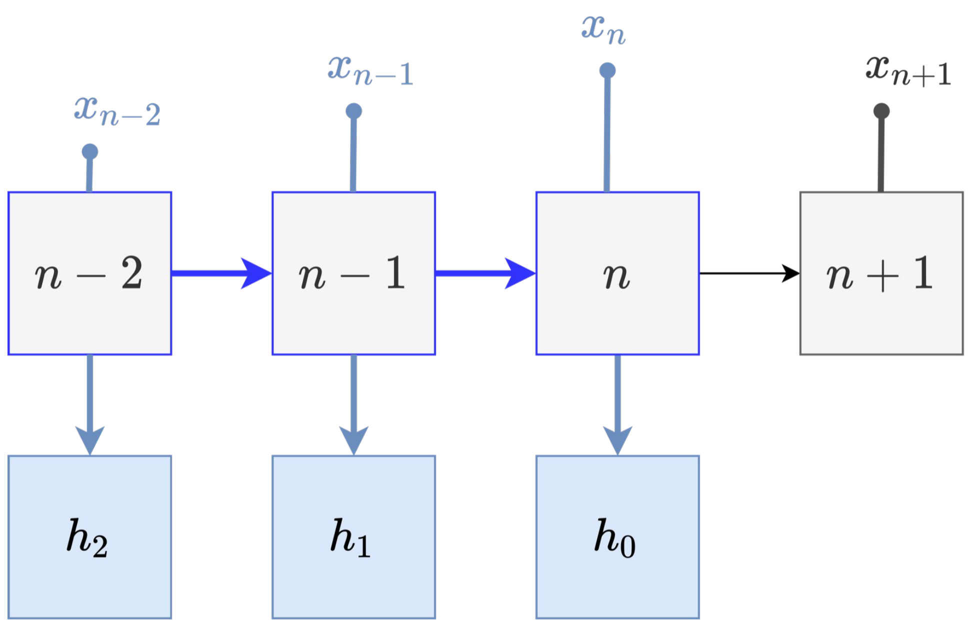


$$S^1\mathbf{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ x_0 \\ x_1 \end{bmatrix}$$



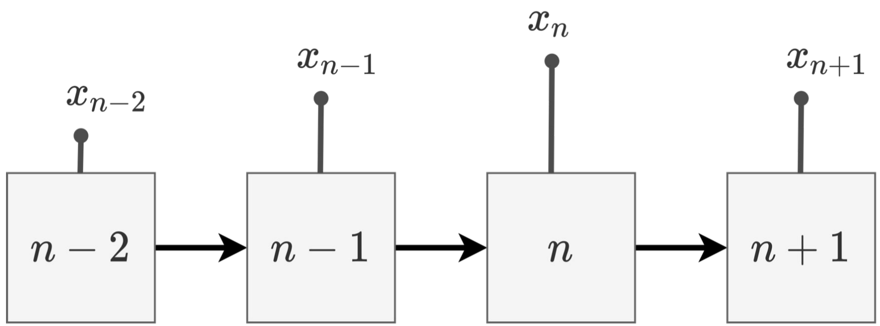
$$S^2\mathbf{x} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ x_0 \end{bmatrix}$$

Convolution in time series



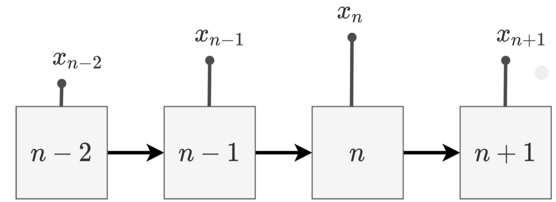
$$h_2 x_{n-2} + h_1 x_{n-1} + h_0 x_n$$

Convolution in time series



$$S = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

We can write the convolution as ...



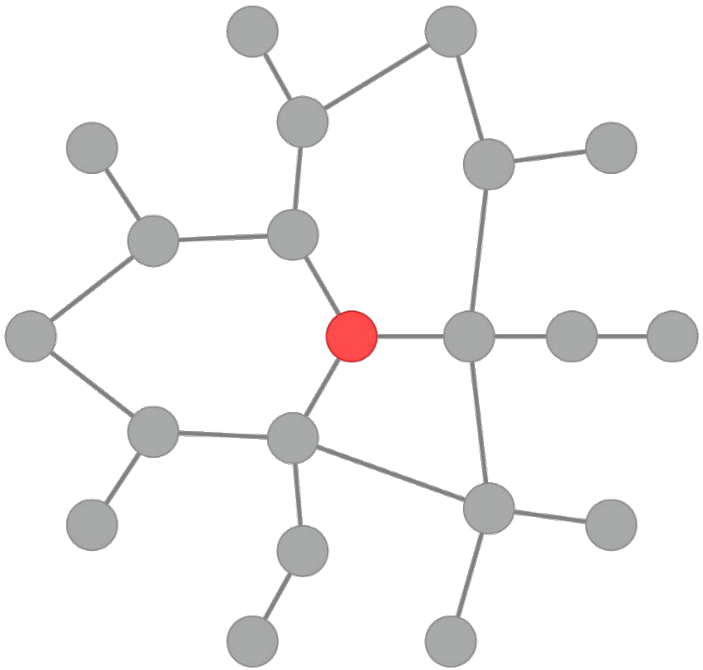
$$S^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad S^1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad S^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

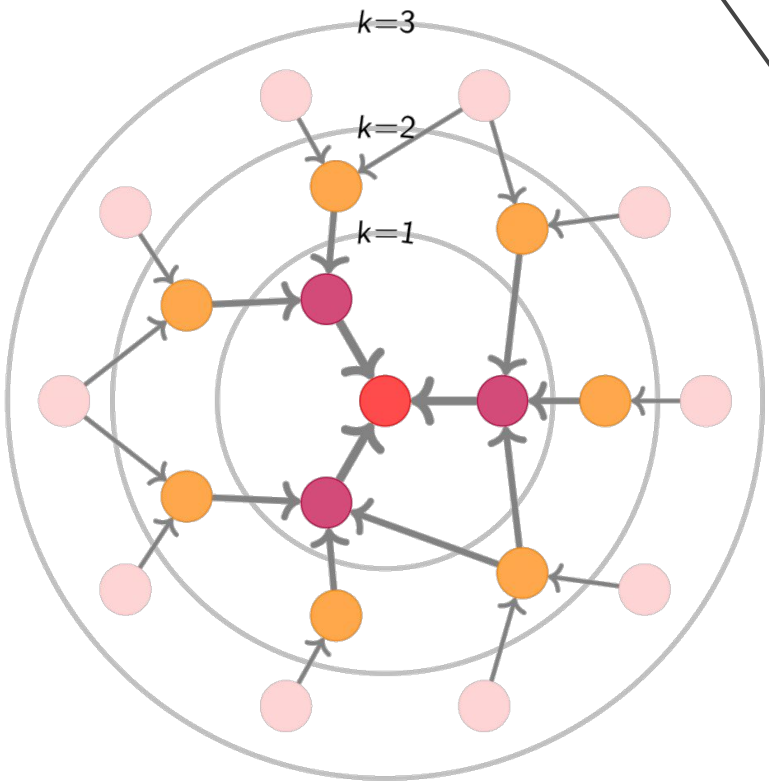
Given the filter $\mathbf{h} = [h_0, h_1, h_2]$

$$H(S) = \sum_{k=0}^{2} h_k S^k = \begin{bmatrix} h_0 & h_1 & h_2 & 0 \\ 0 & h_0 & h_1 & h_2 \\ 0 & 0 & h_0 & h_1 \\ 0 & 0 & 0 & h_0 \end{bmatrix}$$

$$: \mathbf{H}(\mathbf{S})\mathbf{x} = \begin{bmatrix} h_0x_0 + h_1x_1 + h_2x_2 \\ h_0x_1 + h_1x_2 + h_2x_3 \\ h_0x_2 + h_1x_3 \\ h_0x_3 \end{bmatrix}$$

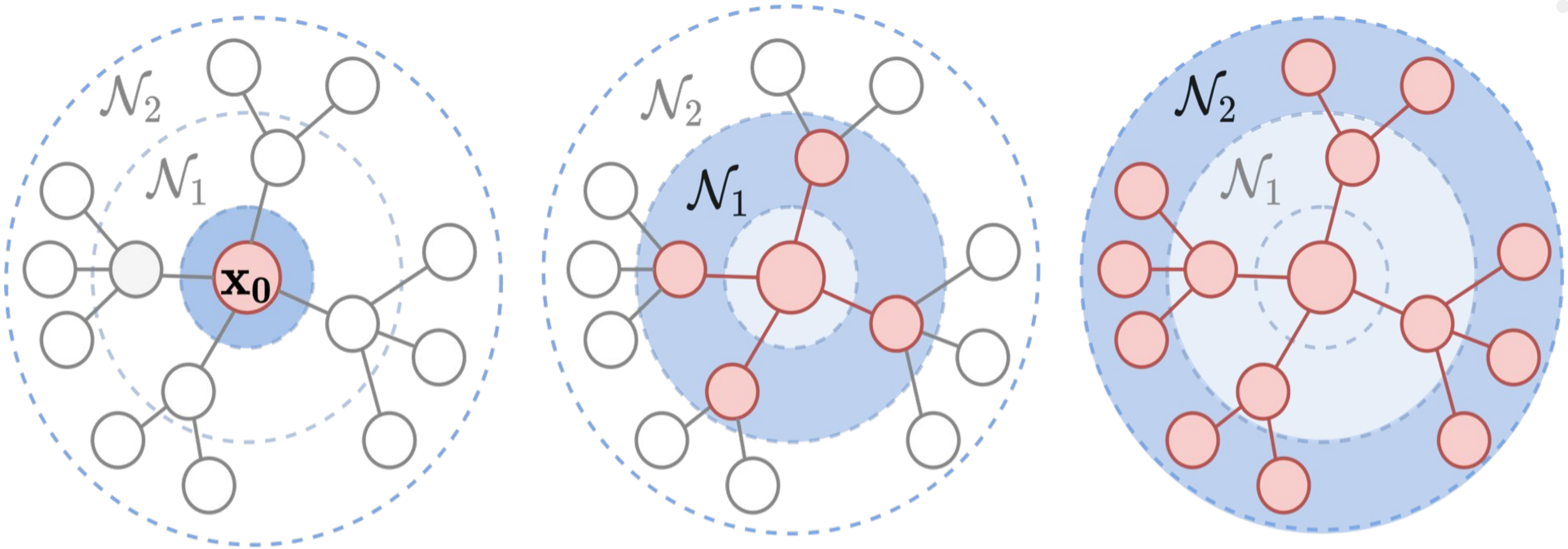
Graph' Convolution





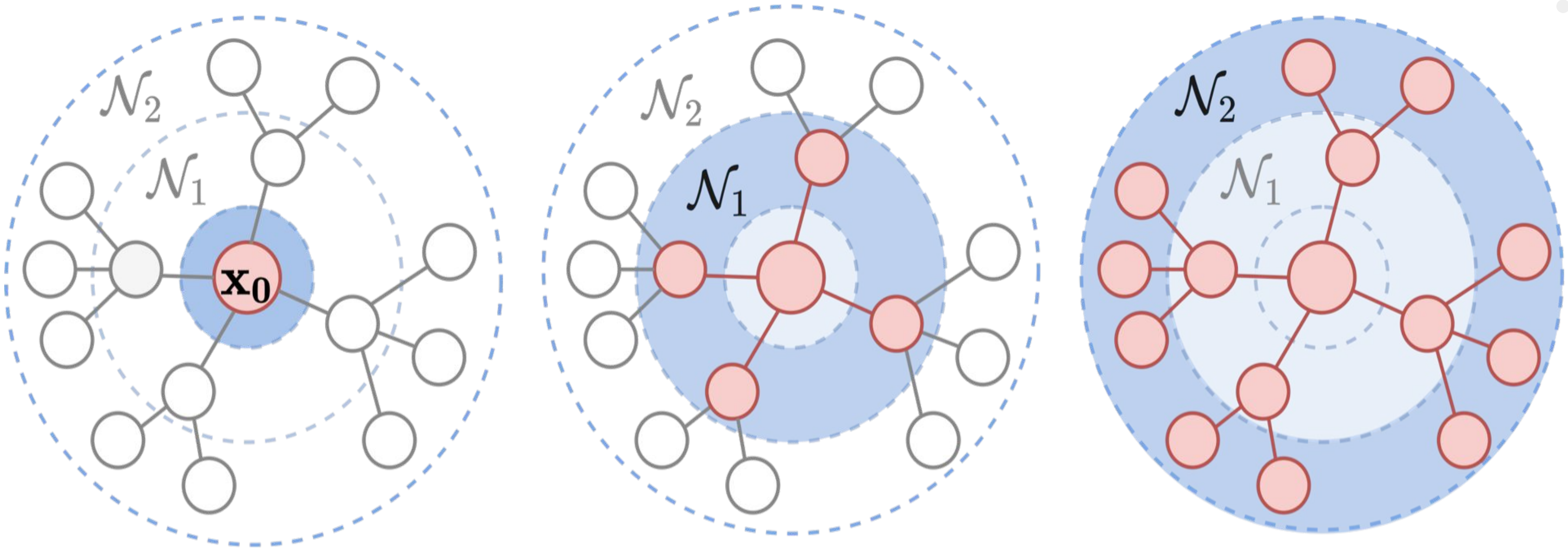
Neighborhoods (*k-hop*)

Graph' Convolution



$$h_0 \mathbf{x}_0 + h_1 \sum_{i \in \mathcal{N}_1} \mathbf{x}_i + h_2 \sum_{i \in \mathcal{N}_2} \mathbf{x}_i$$

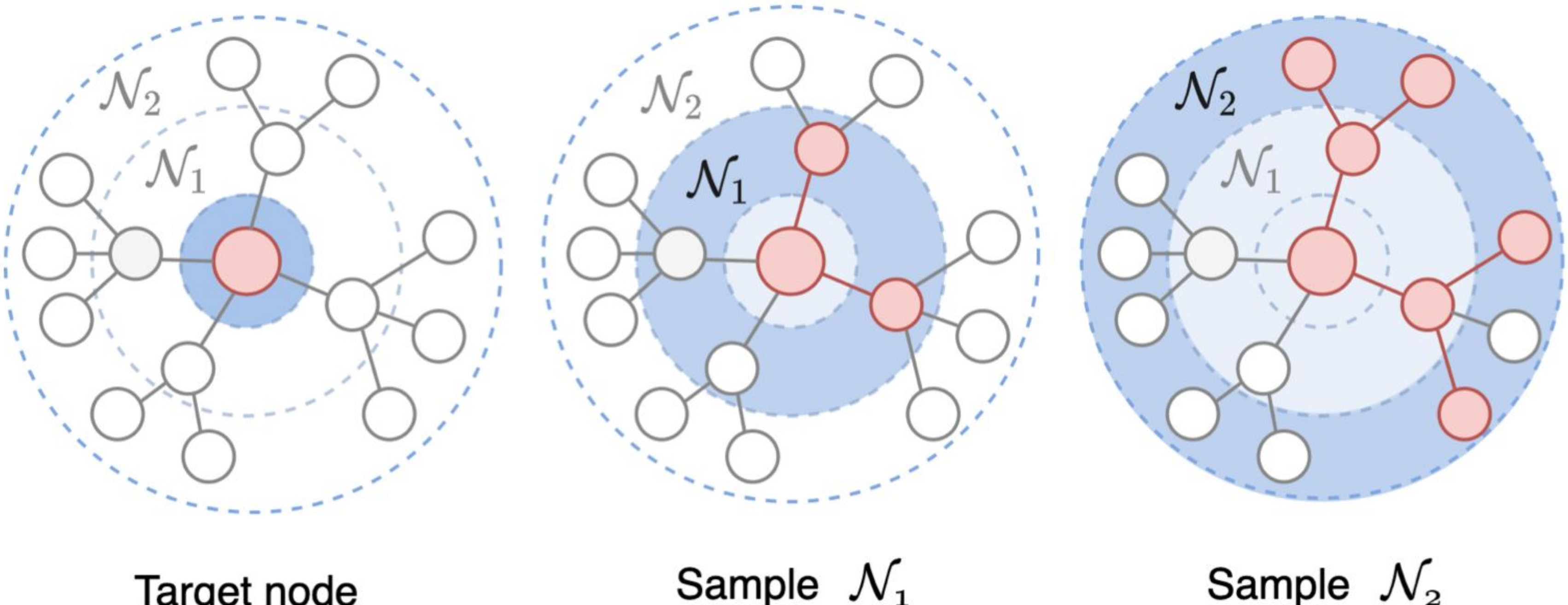
Graph Convolution



$$\mathbf{H}_0\mathbf{x_0} + \mathbf{H}_1\sum_{i\in\mathcal{N}_1}\mathbf{x_i} + \mathbf{H}_2\sum_{i\in\mathcal{N}_2}\mathbf{x_i}$$

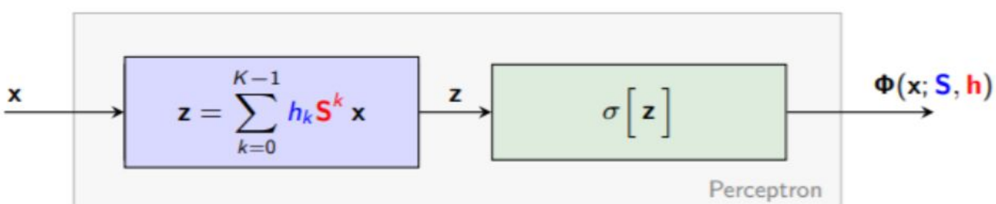
$$\Rightarrow \mathbf{H}_k \in \mathbb{R}^{\text{dim in}\times\text{dim out}}$$

We can also fix the number of neighbors

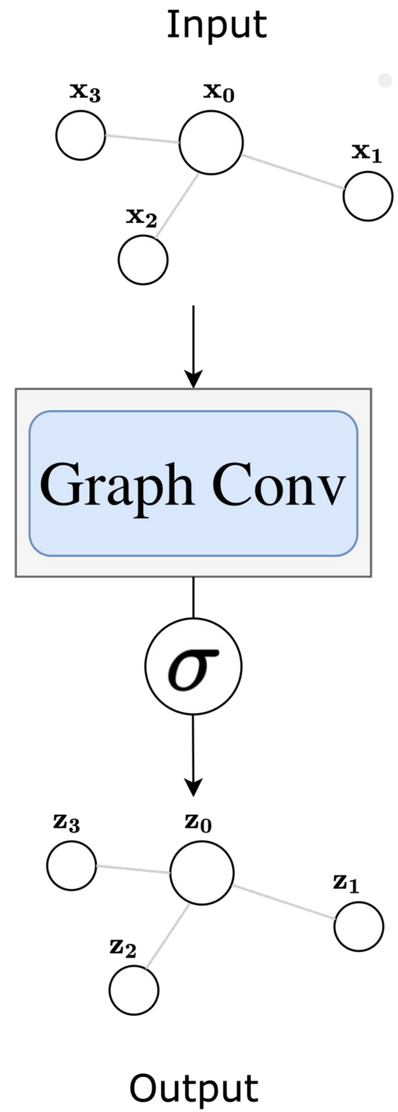


$$\mathbf{H}_0 \mathbf{x_0} + \mathbf{H}_1 \sum_{i \in \mathcal{N}_1} \mathbf{x_i} + \mathbf{H}_2 \sum_{i \in \mathcal{N}_2} \mathbf{x_i}$$

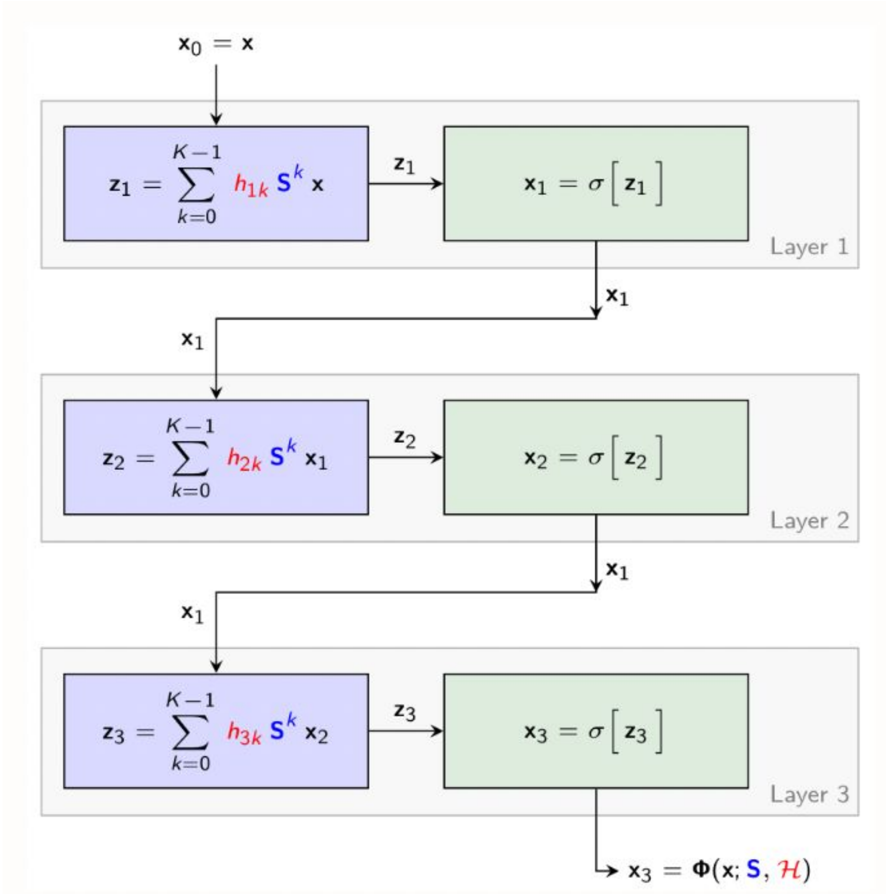
The Graph Perceptron



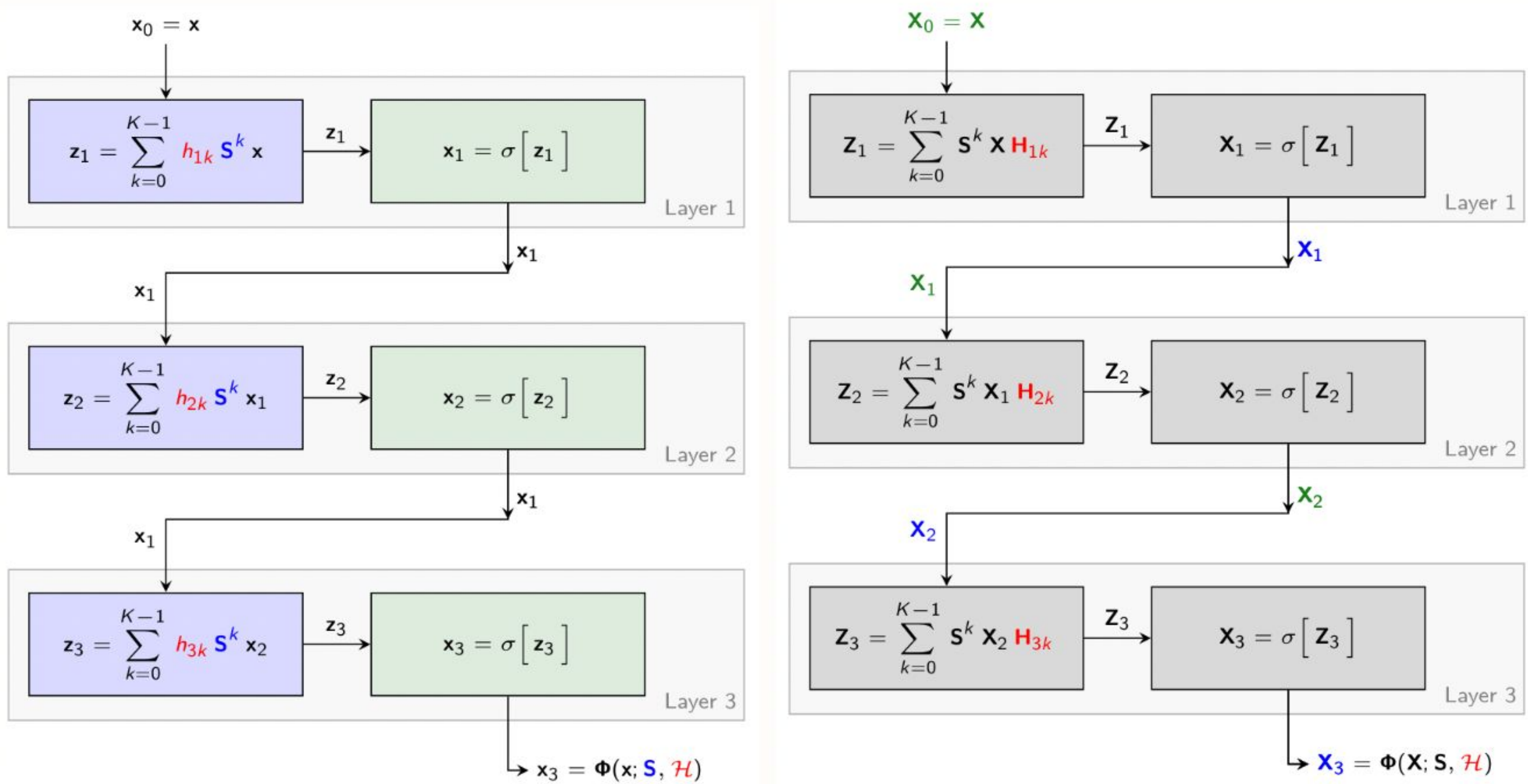
https://gnn.seas.upenn.edu/wp-content/uploads/2020/09/lecture_4_handout.pdf



A Graph Convolutional neural network with many layers



A Graph Convolutional neural network with several channels



Algorithm 2: Graph Node regression mini-batch SGD.

```
for epoch in 1, ..., n_epochs do
    for batch in 1, ..., n_batches do
        Sample m nodes without replacement
        for node in 1, ..., m do
            Sample k-hop neighbourhood
            Compute prediction for the target node
            Compute gradients
        end
        Gradient descent step
    end
end
```

GNNs examples

A cow, a node

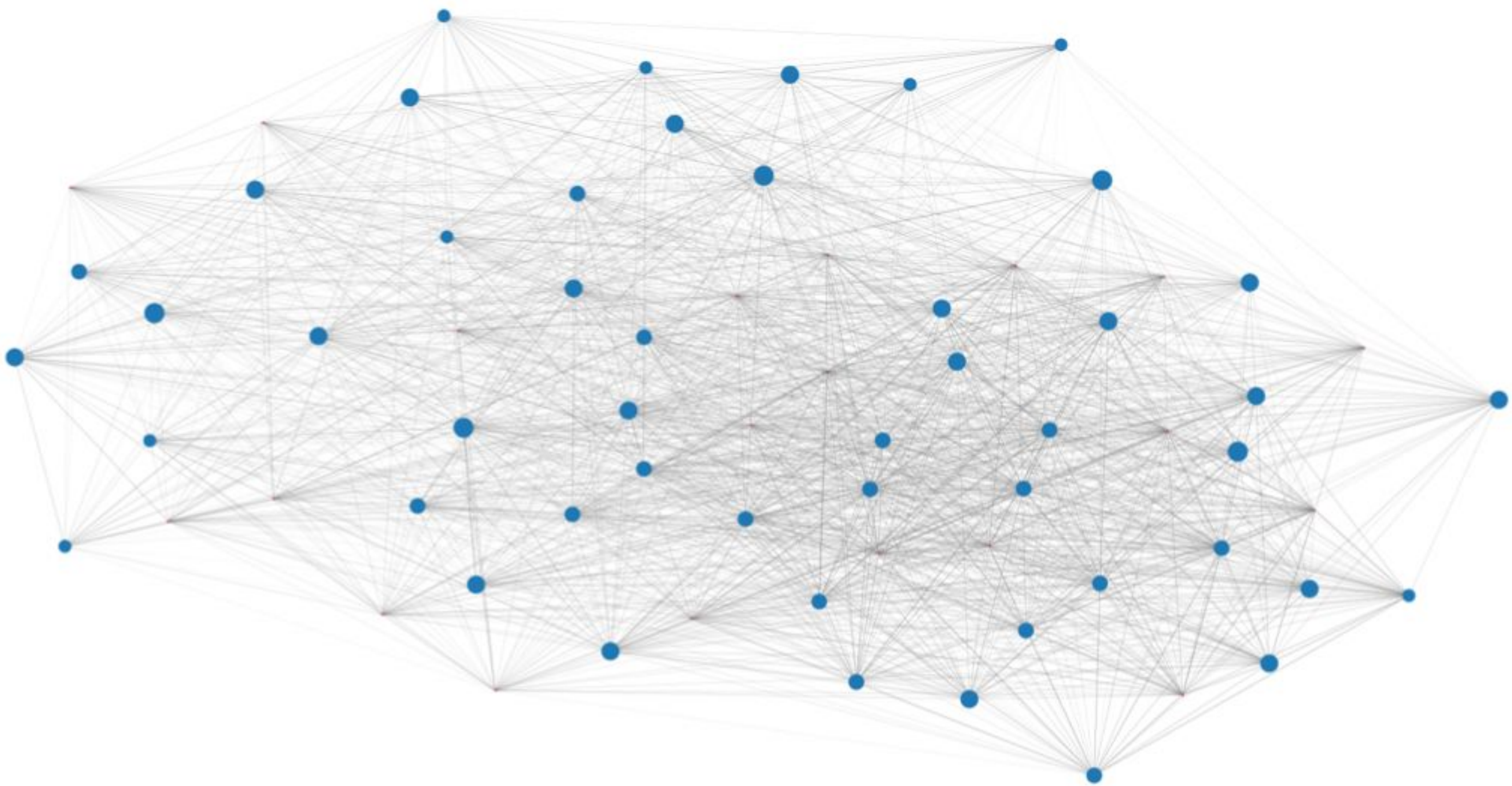
Another example: recommendation system

A user has rated several products.

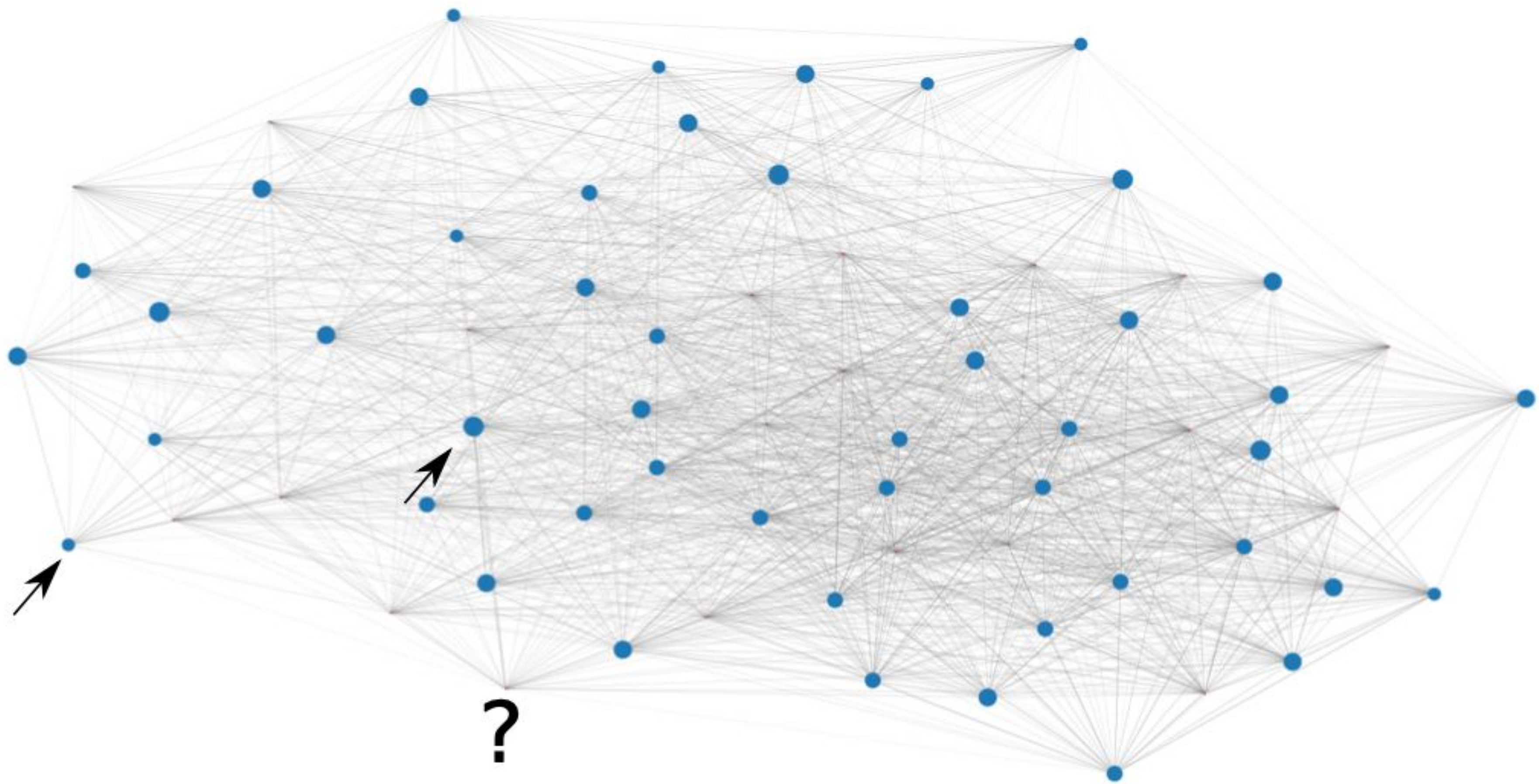
Goal: predict the rating of products that she did not rate

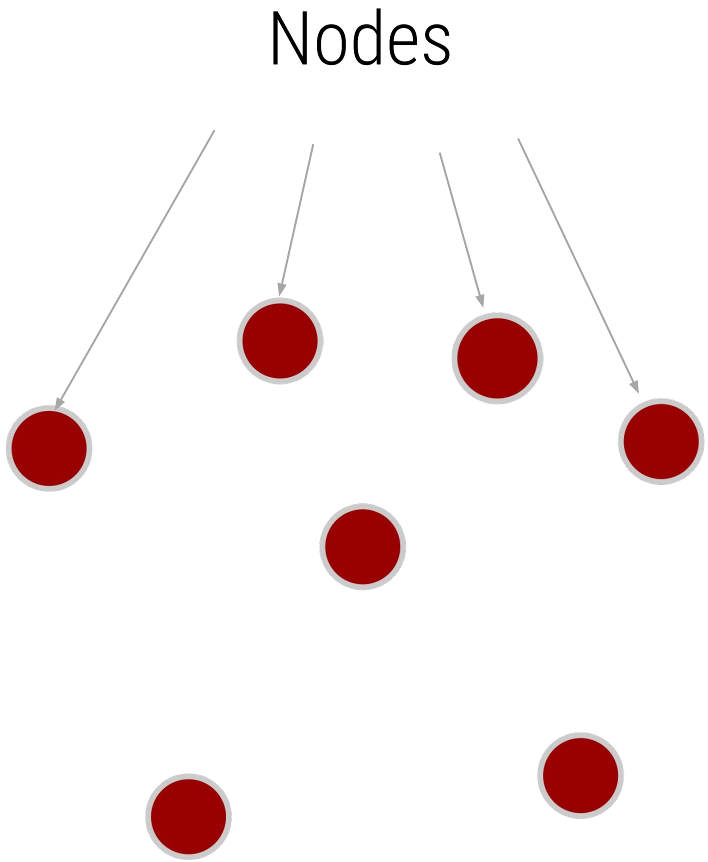
Sub-sample of the movie's graph based on the dataset Movielens-100k.

The signal of one user is represented in the graph

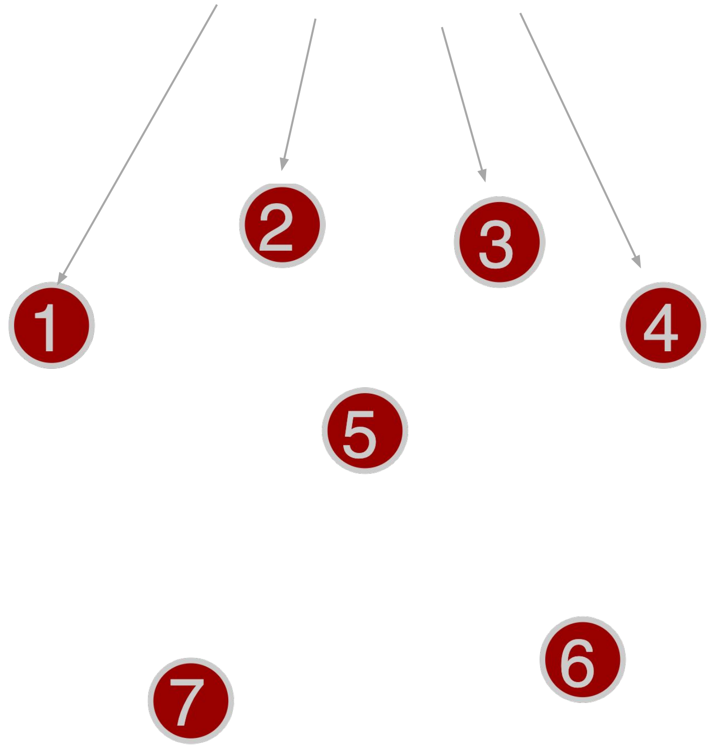


How would you rate this movie, base on the ratings that the user gave to other movies

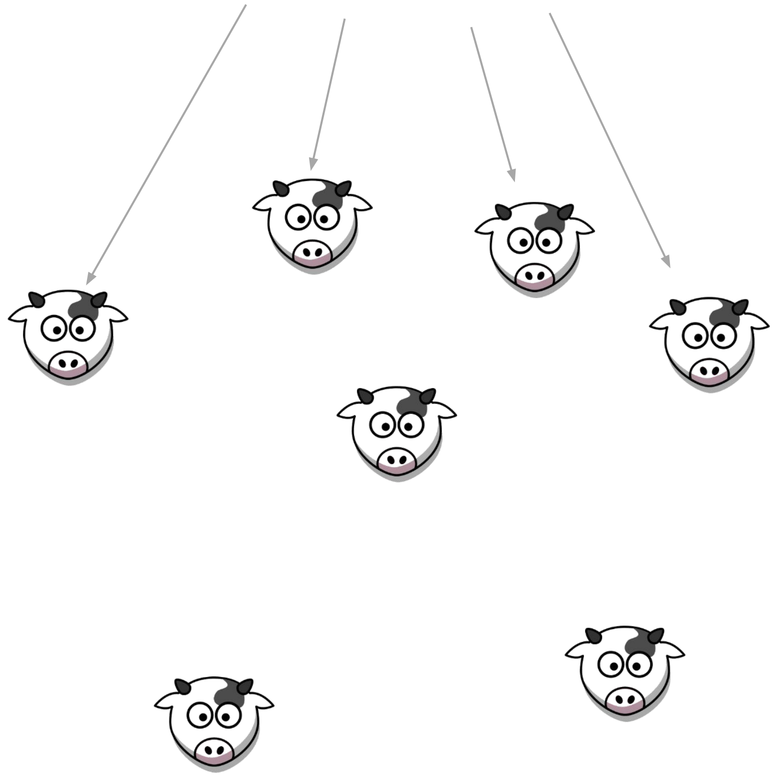



Nodes

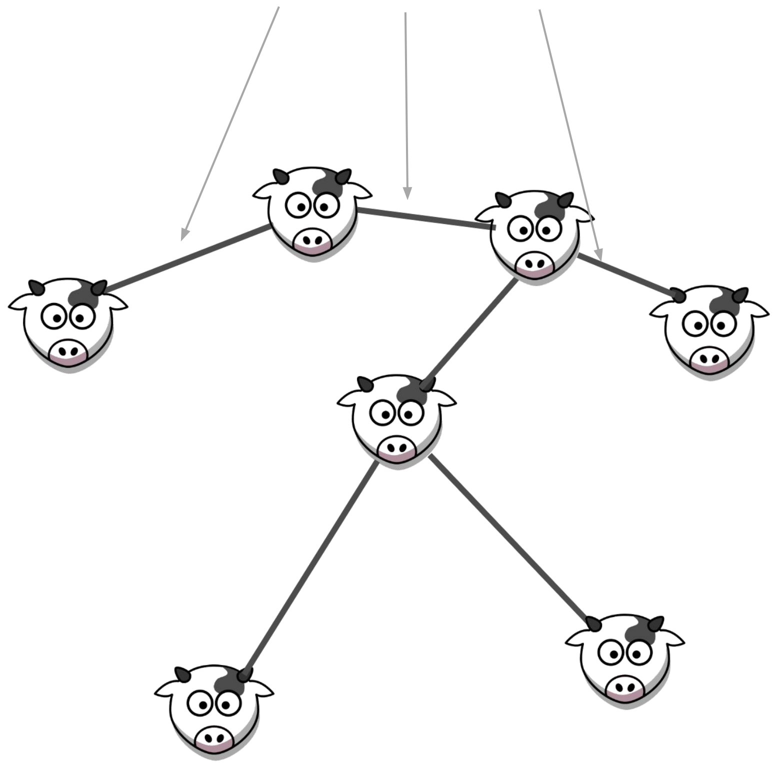
Nodes



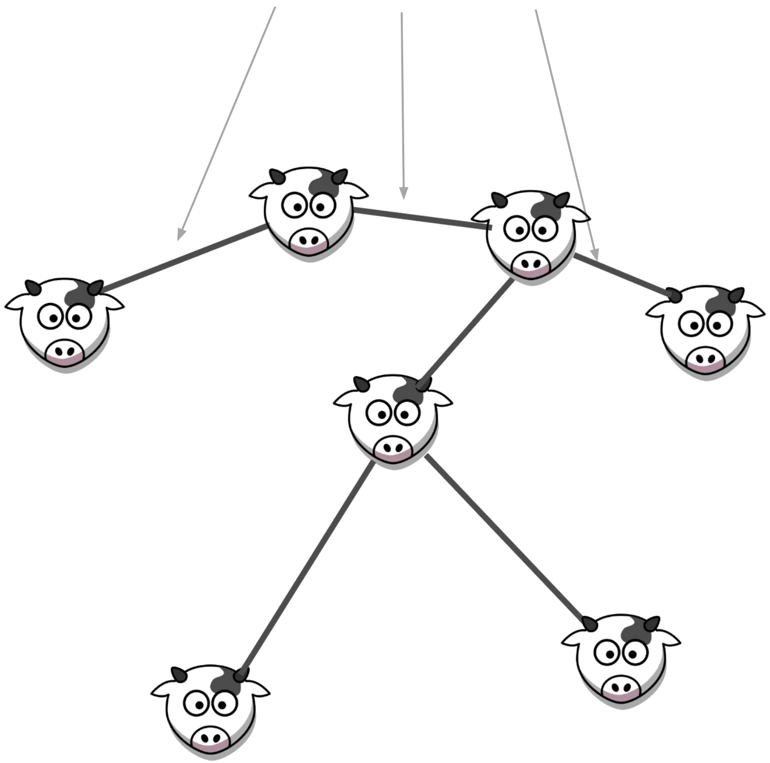
Nodes: cows



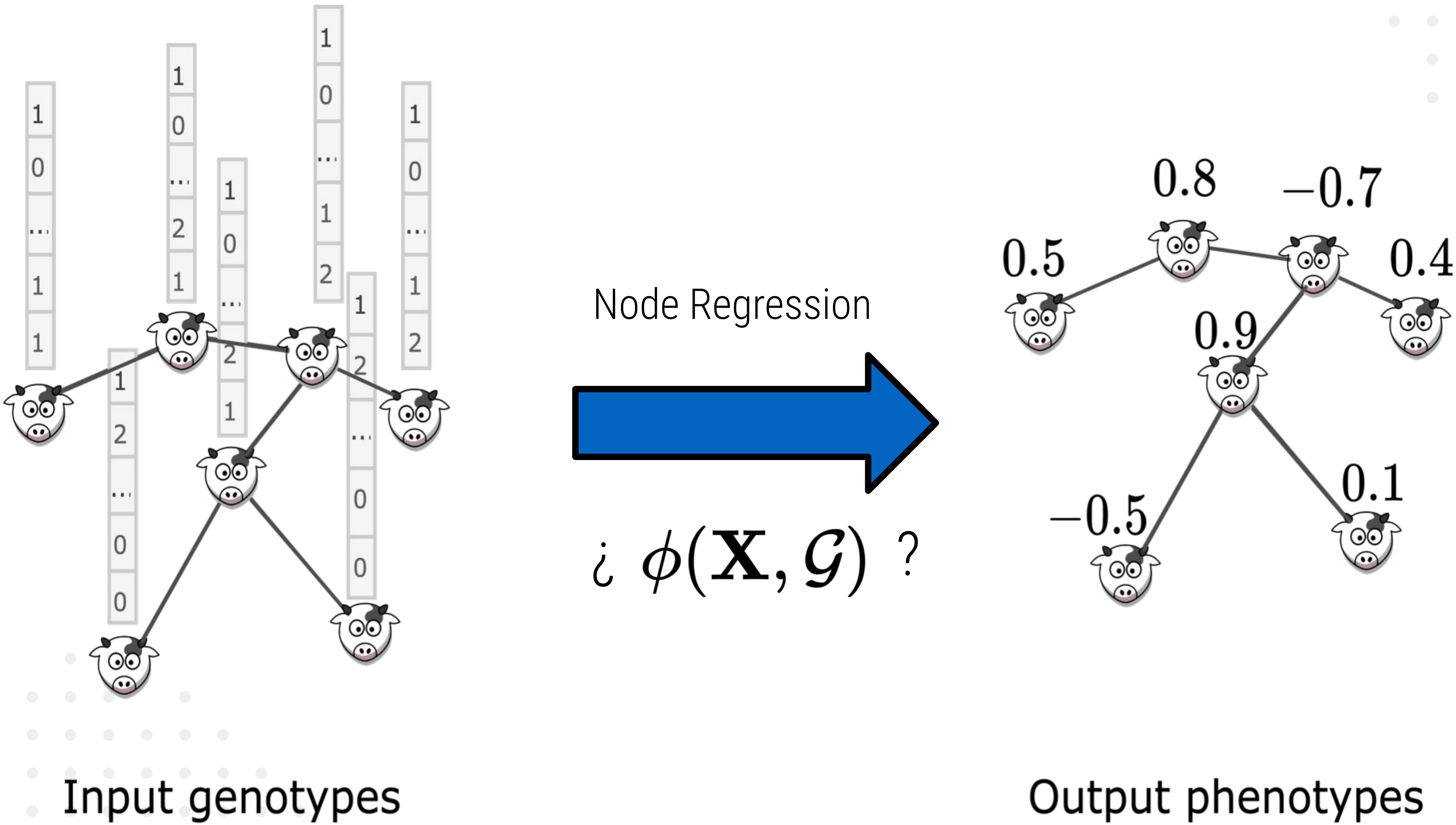
Edges



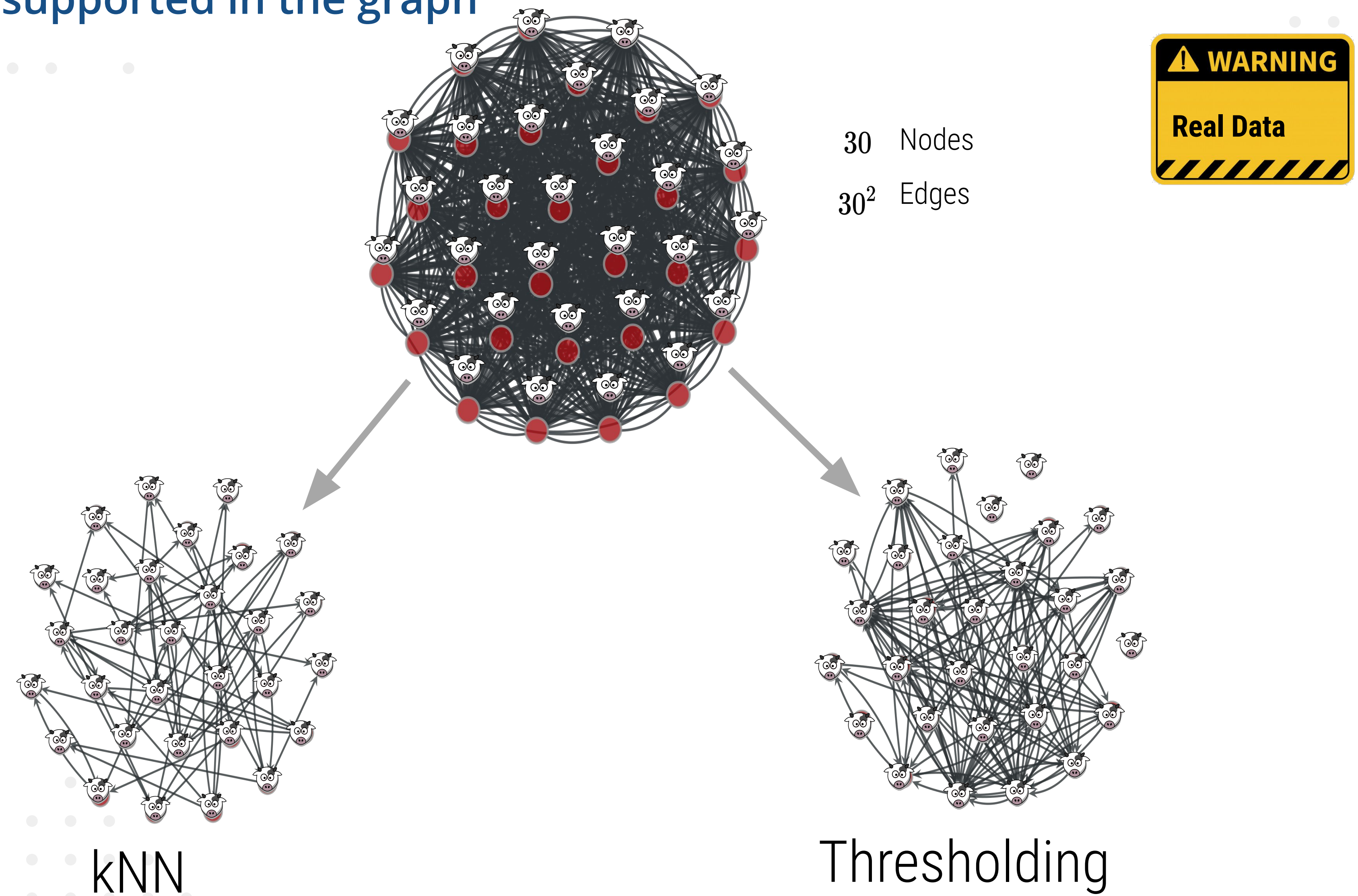
Aristas - Ej: Kinship



Signal supported in the graph

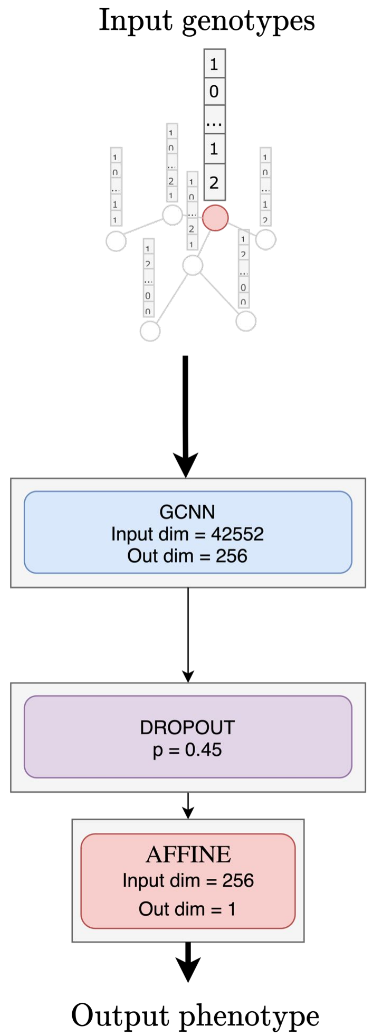


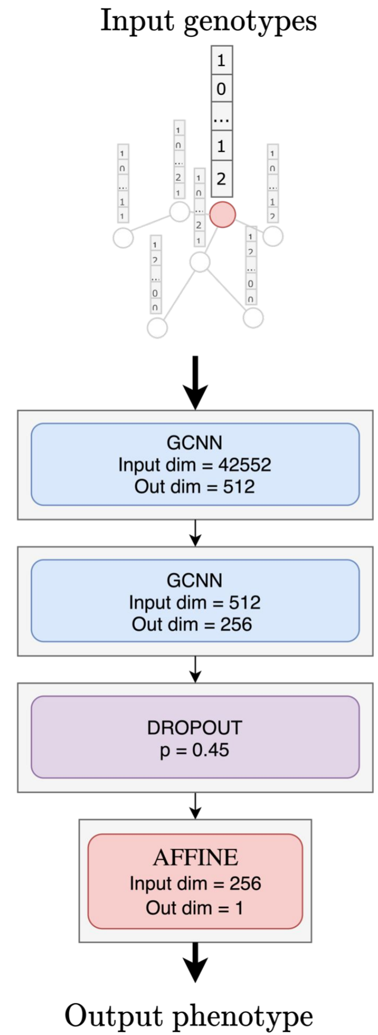
Signal supported in the graph



Examples with one or two layers

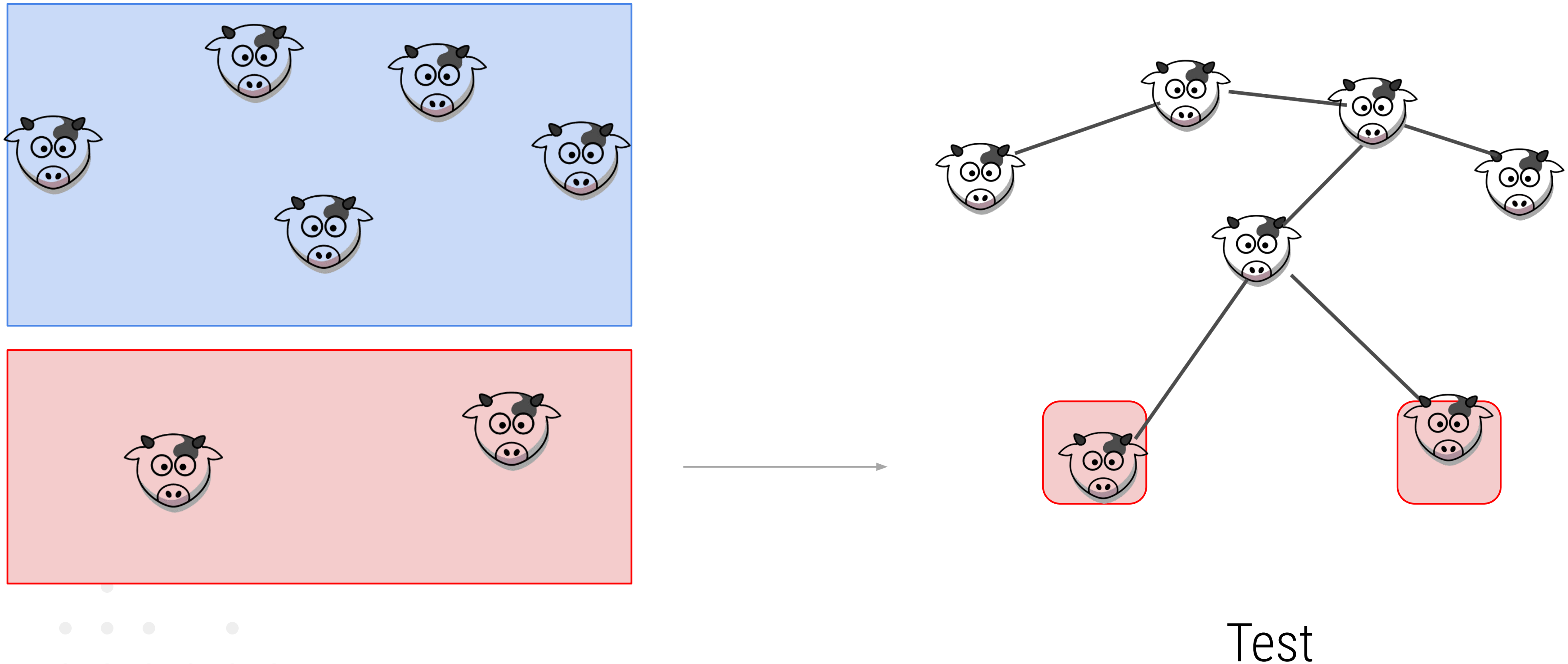
1 Layer



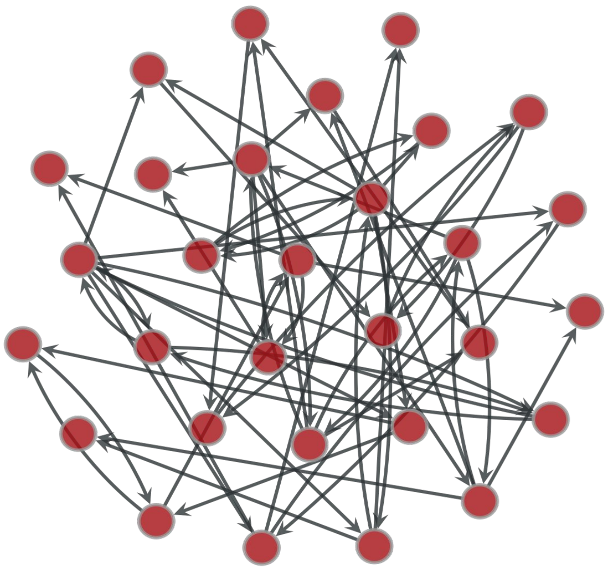


2 Layers

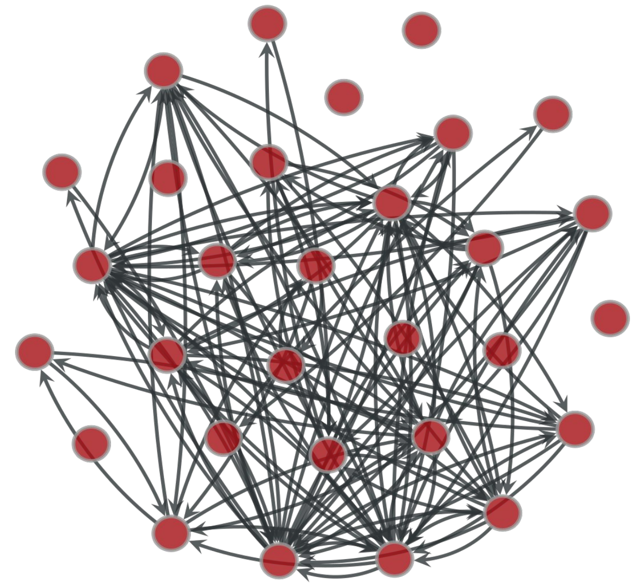
Training



Transferability

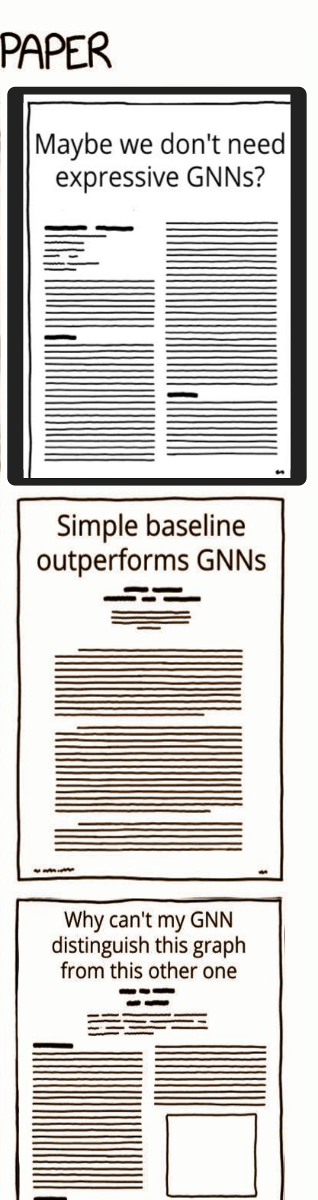


“Transferencia”



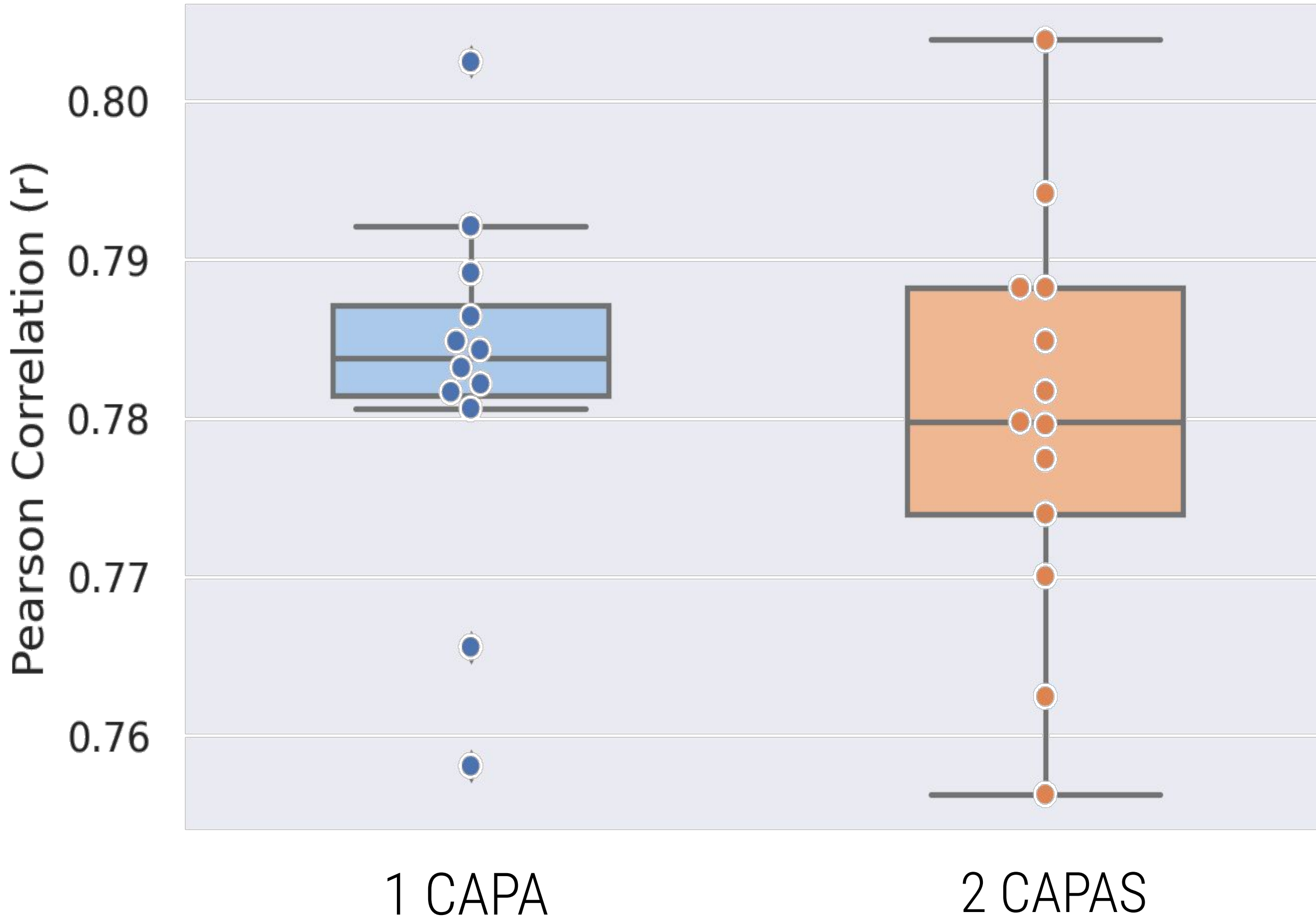
Entrenamiento

Test



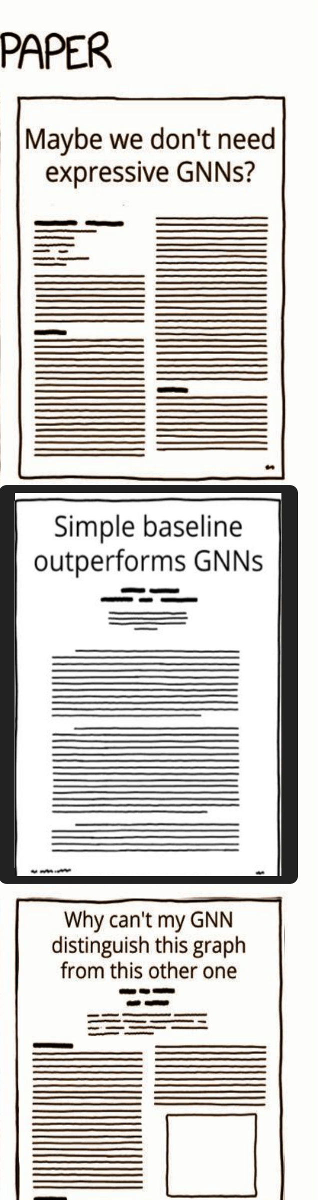
Capaz no necesitamos

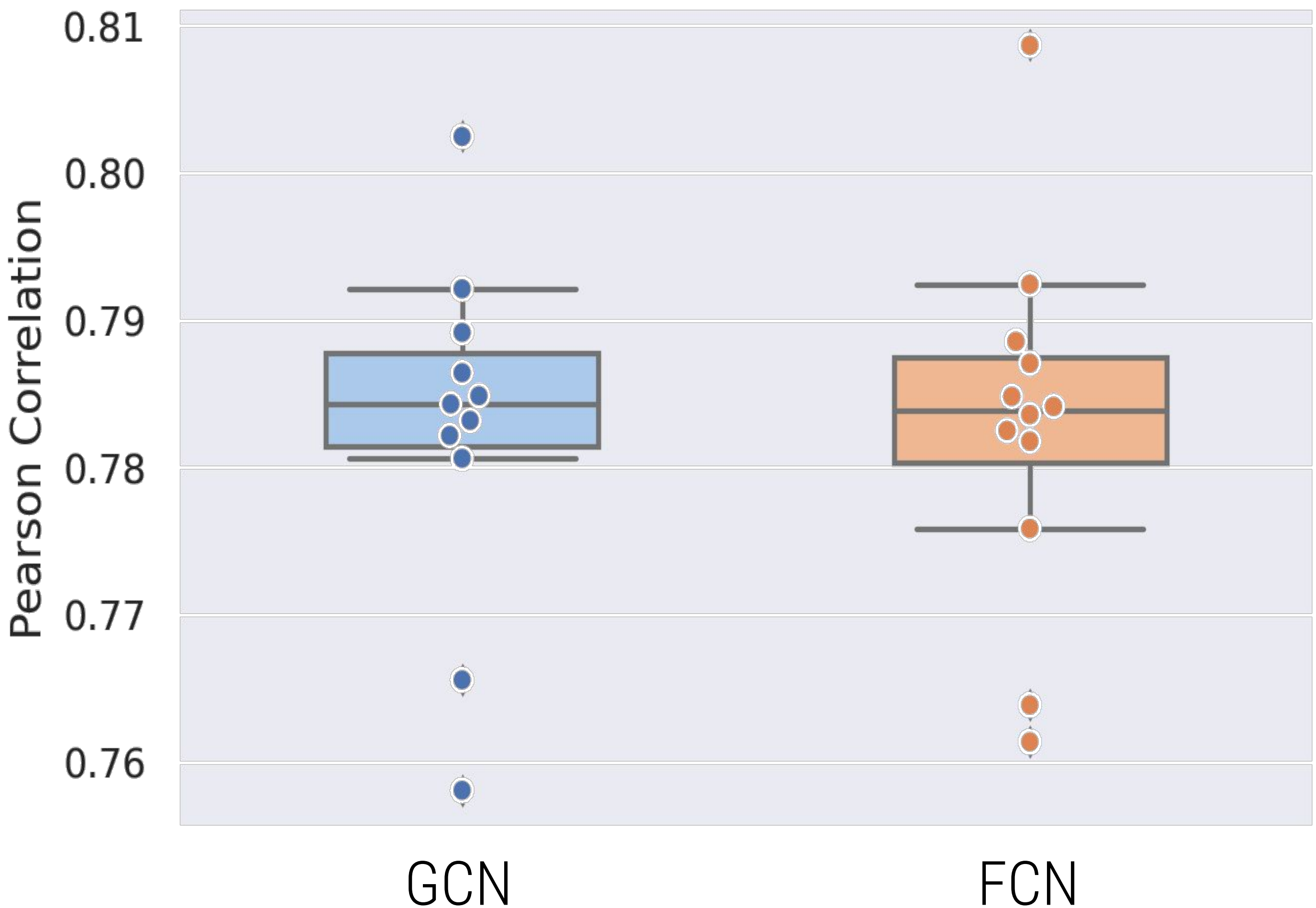
GNNs expresivas

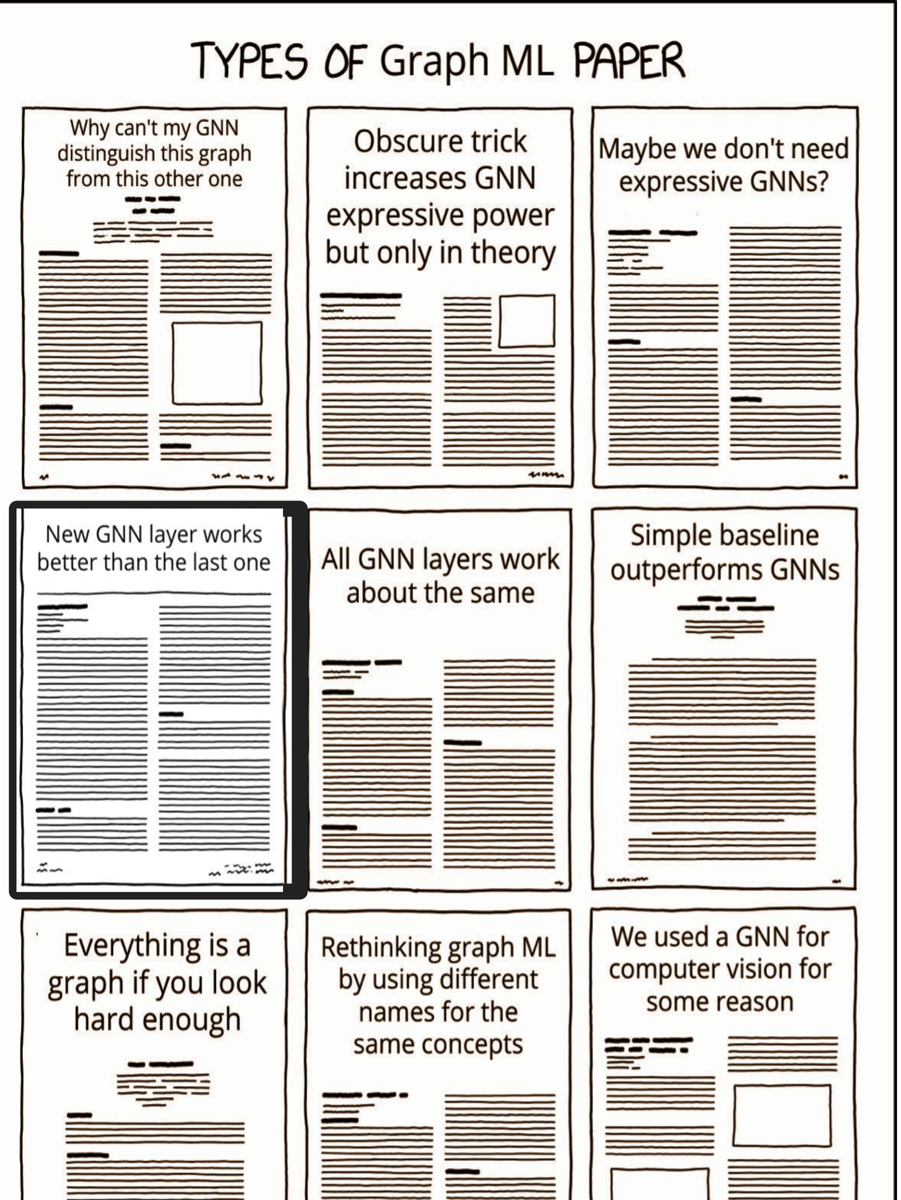


Un baseline Simple

Tiene desempeño similar a las GNNs



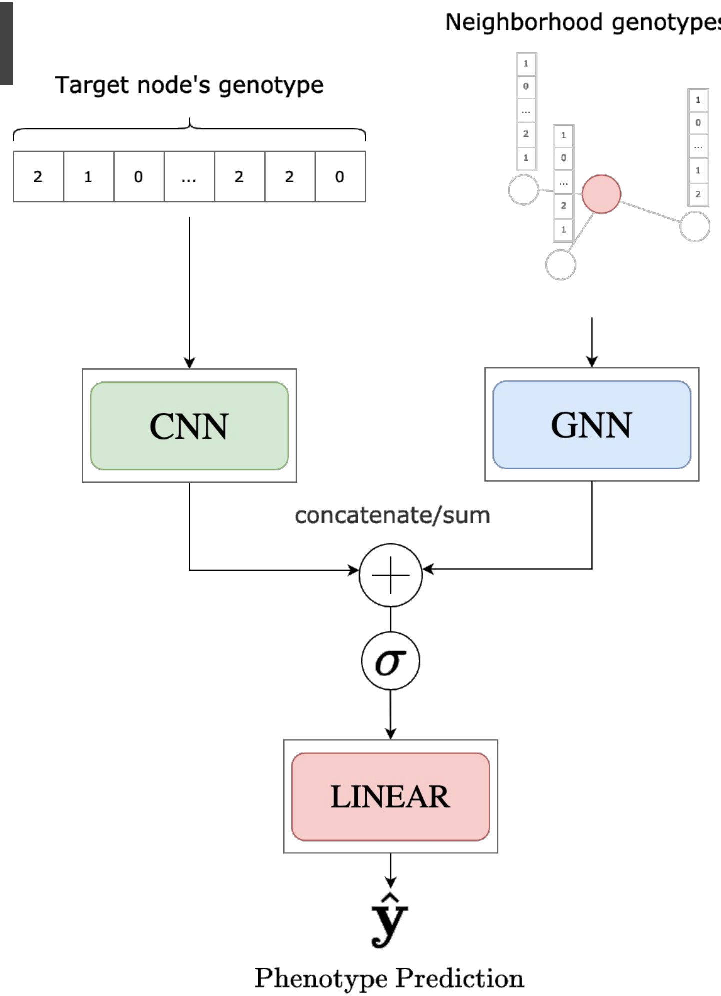




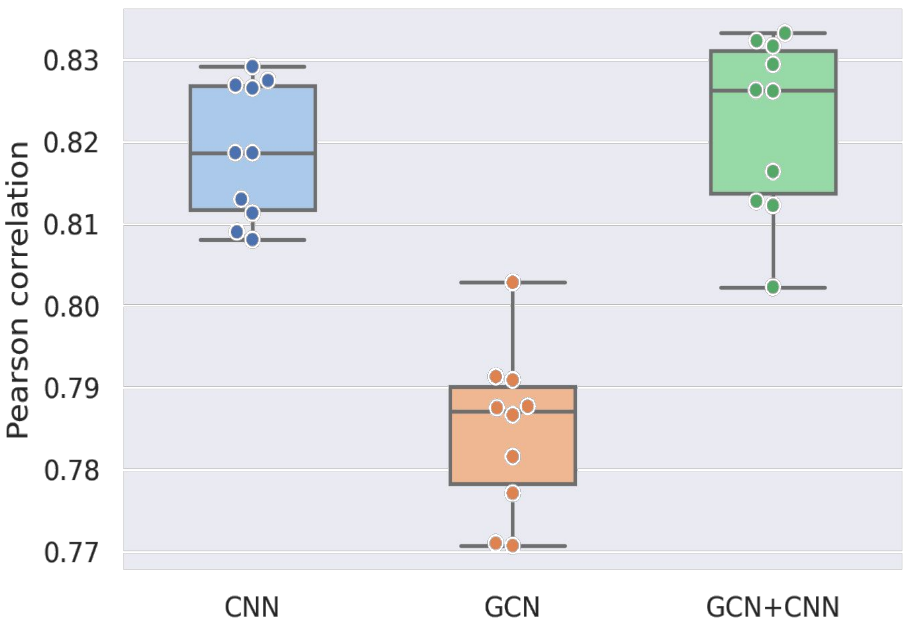
CNN+GNNs ?

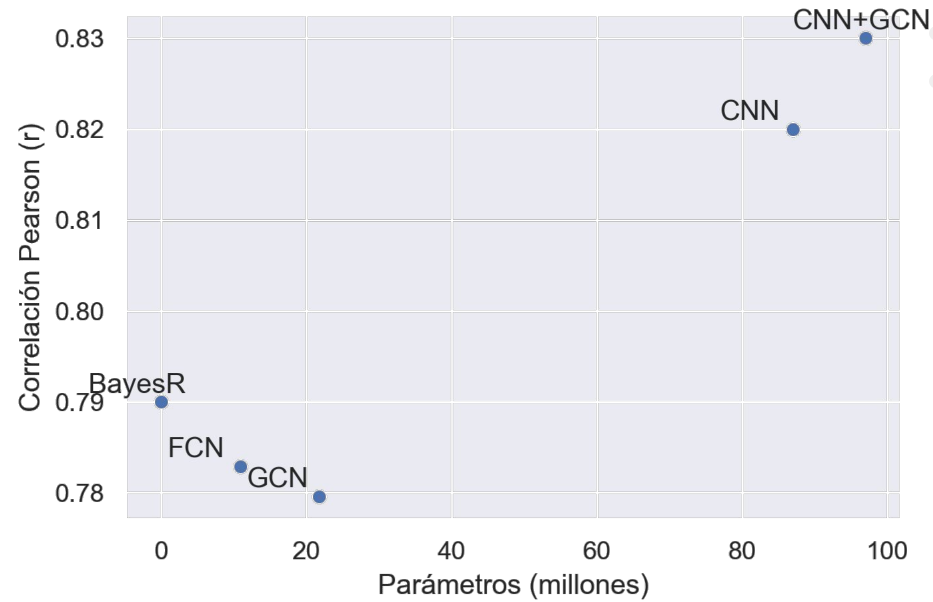
Lo mejor de los dos mundos (?)

ARQUITECTURA GCN+CNN

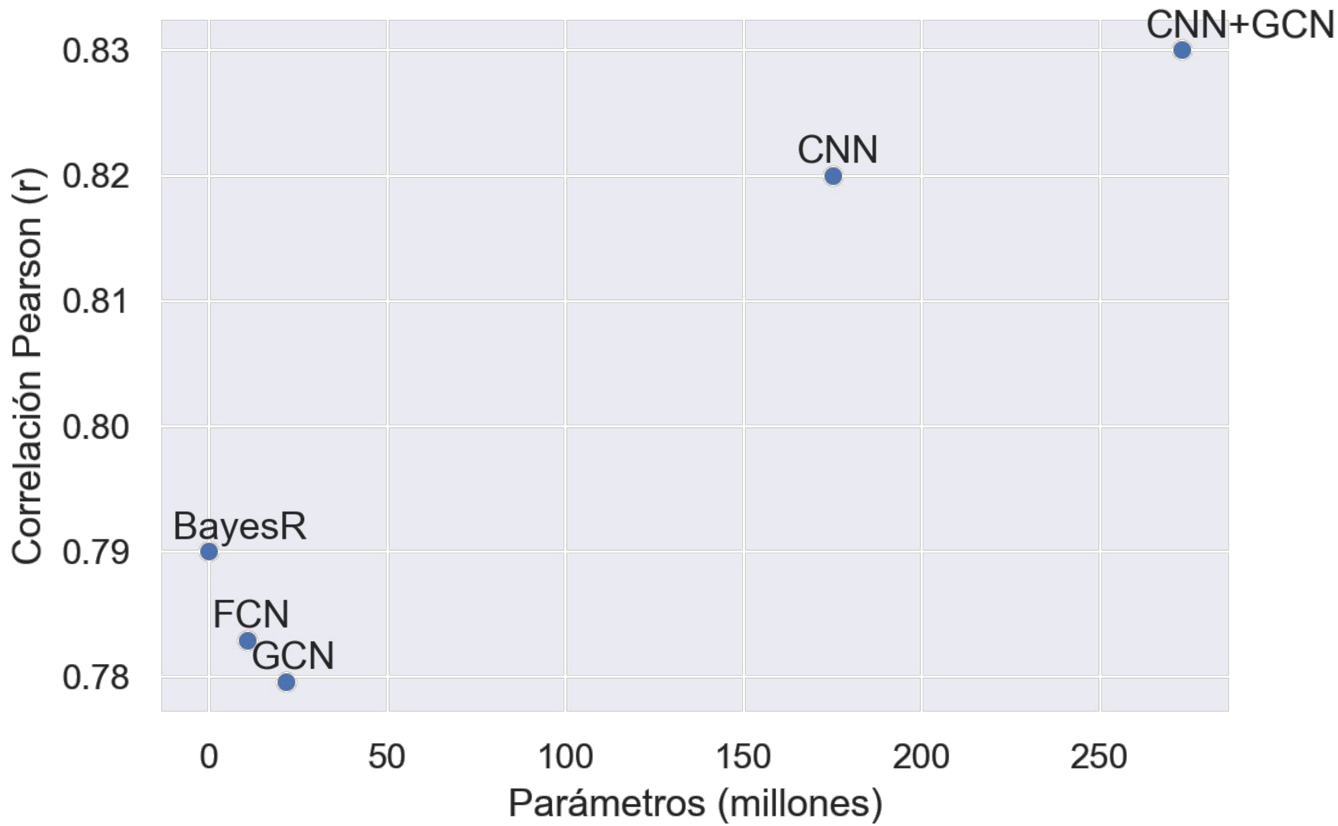


COMBINAR CNN Y GCN MEJORA EL DESEMPEÑO





SOBRE-PARAMETRIZACIÓN



2D-CNN

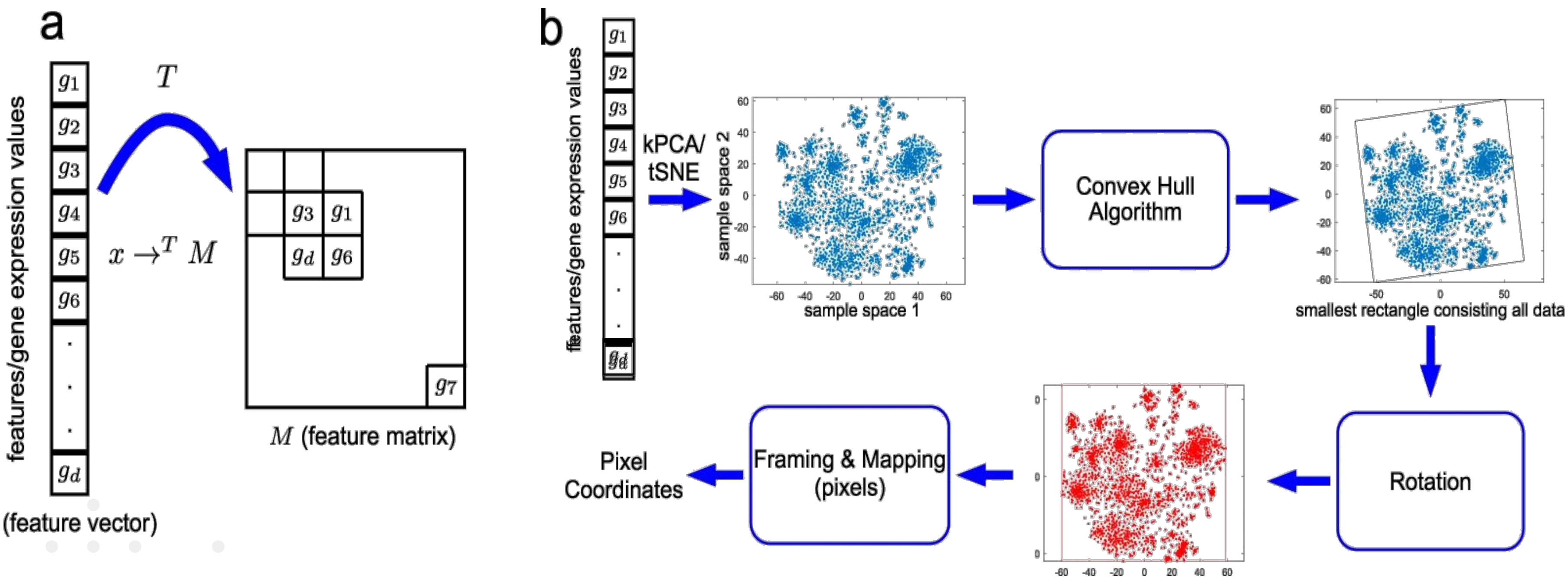
Un camino hacia GNNs en los SNPs?

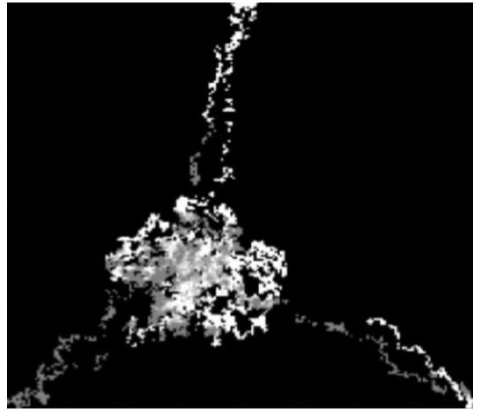
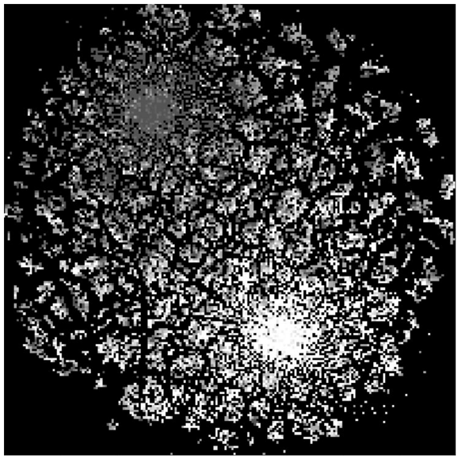
Una imagen vale más que mil SNPs?

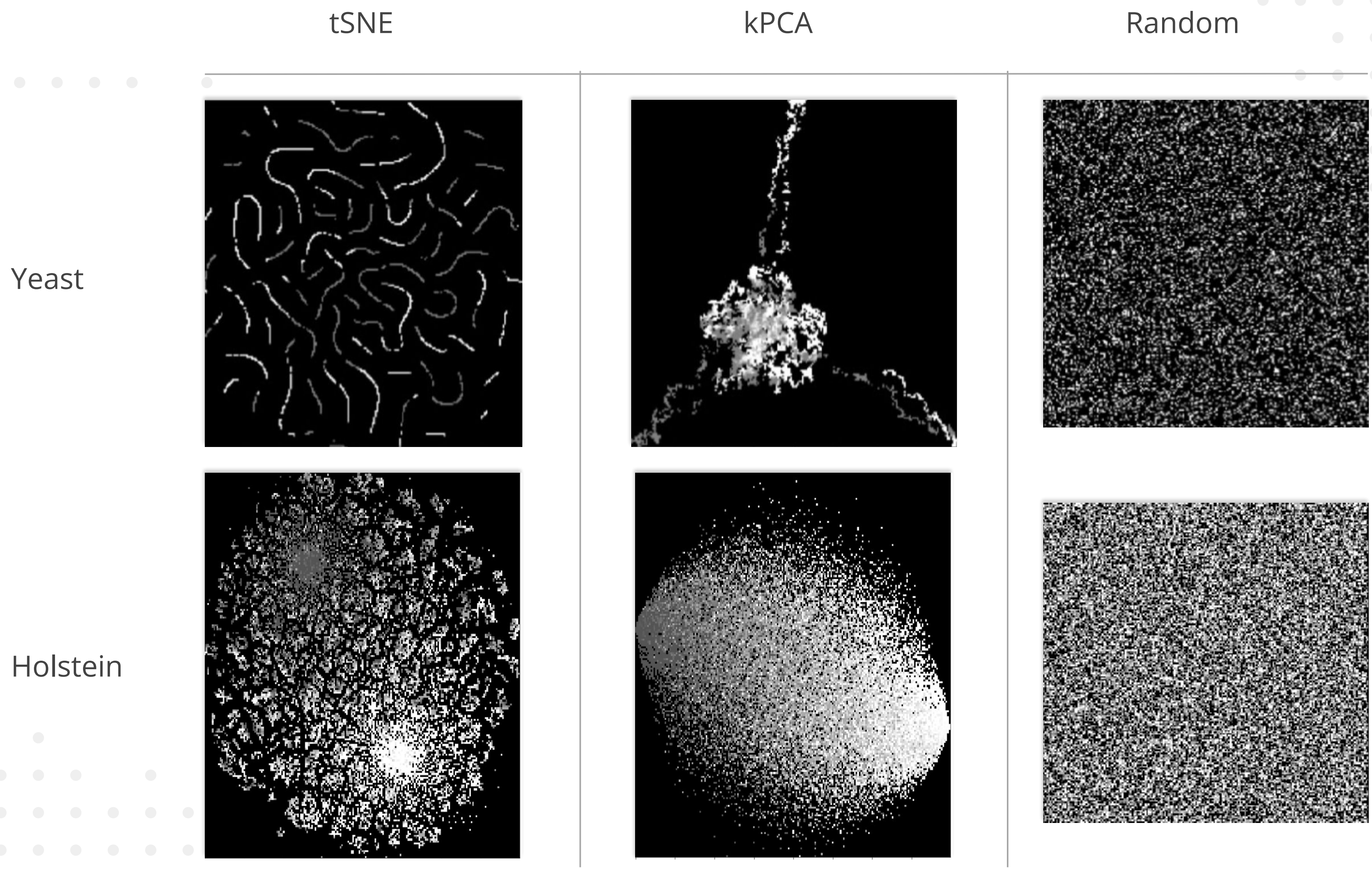
Article | Open Access | Published: 06 August 2019

DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture

Alok Sharma, Edwin Vans, Daichi Shigemizu, Keith A. Boroevich & Tatsuhiko Tsunoda

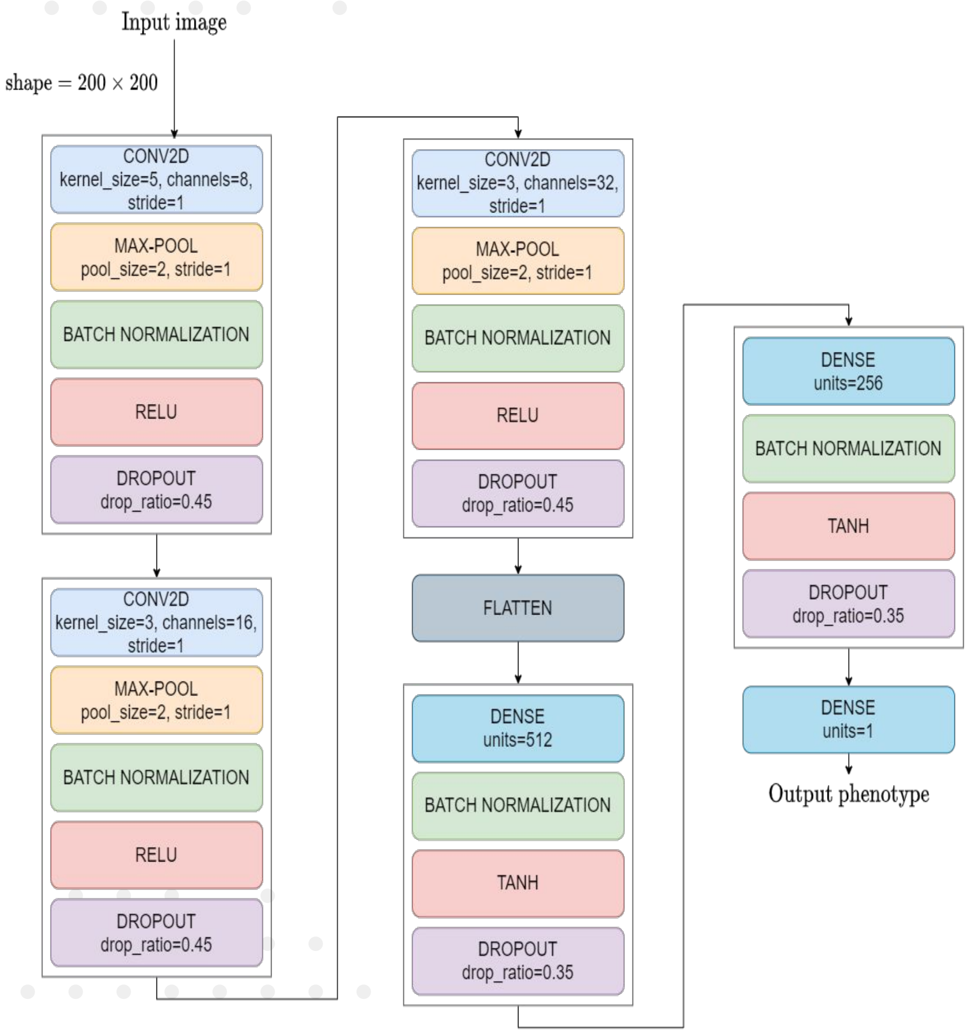


	tSNE	kPCA
Yeast		
Holstein		

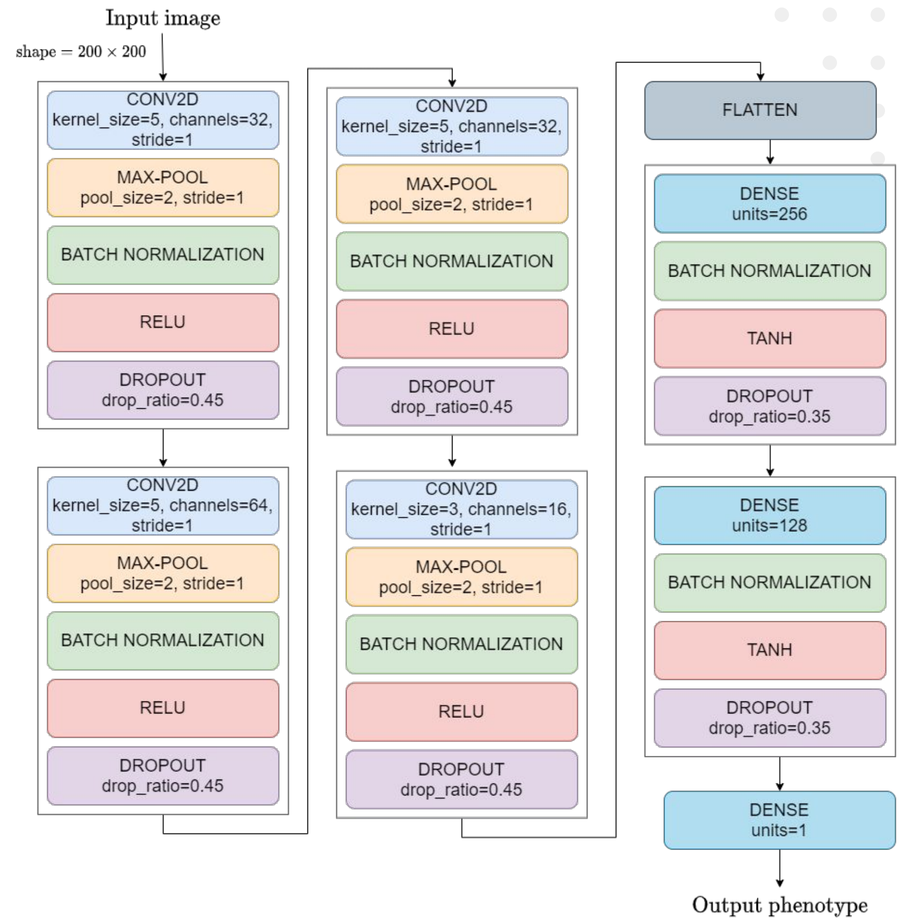


tSNE
kPCA
Random
Yeast
Holstein

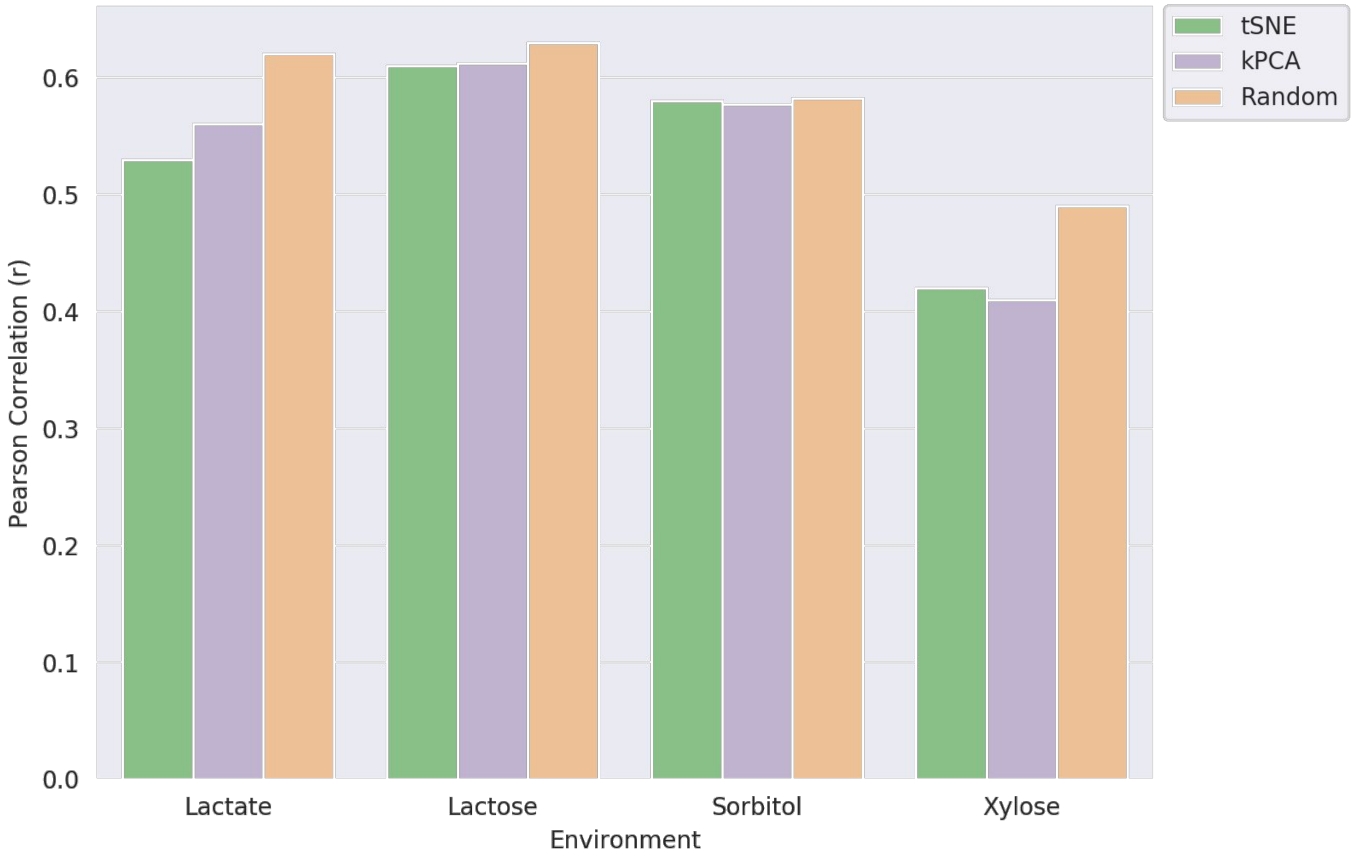
Yeast 2D CNN



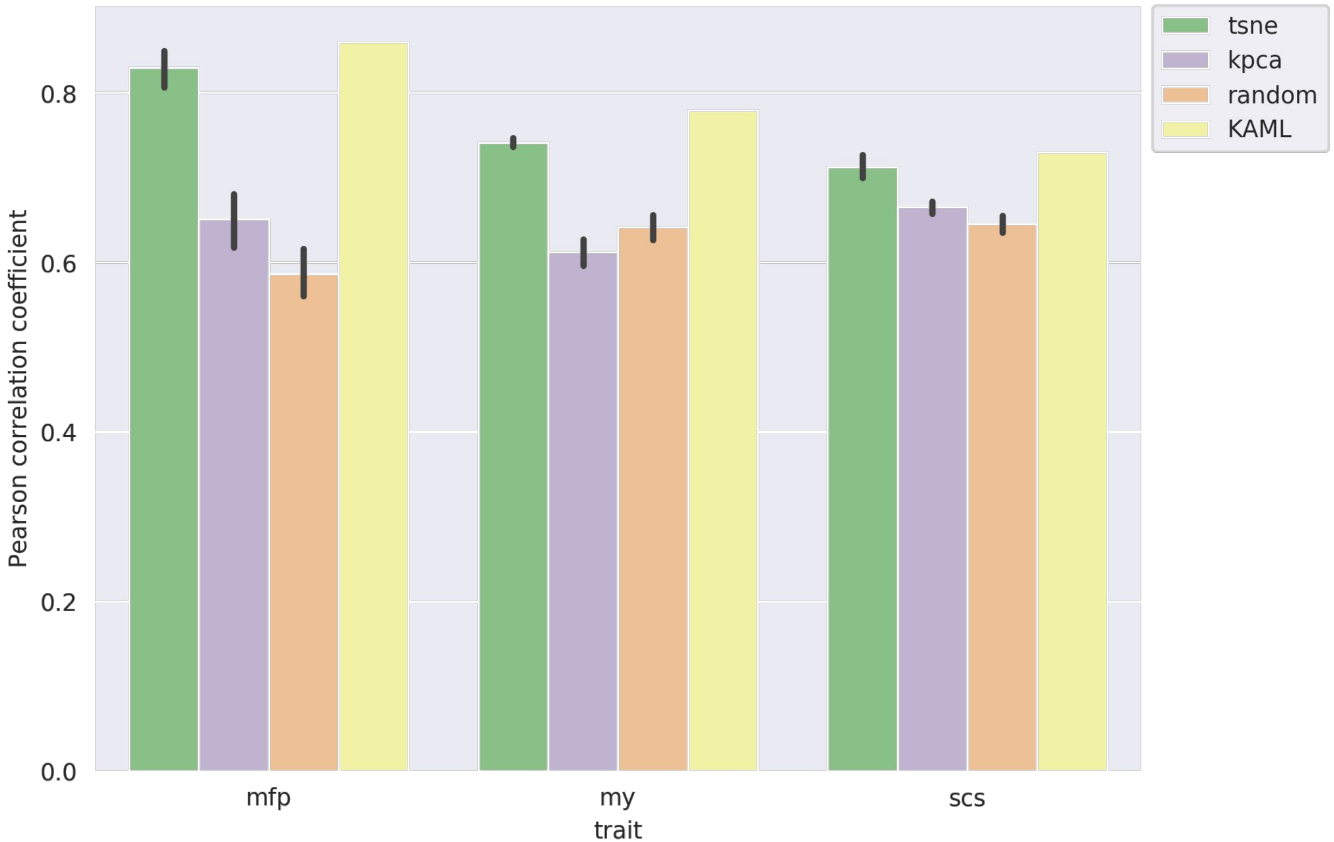
Holstein 2D CNN



Results: Yeast



Results: Holstein



Clustering analysis