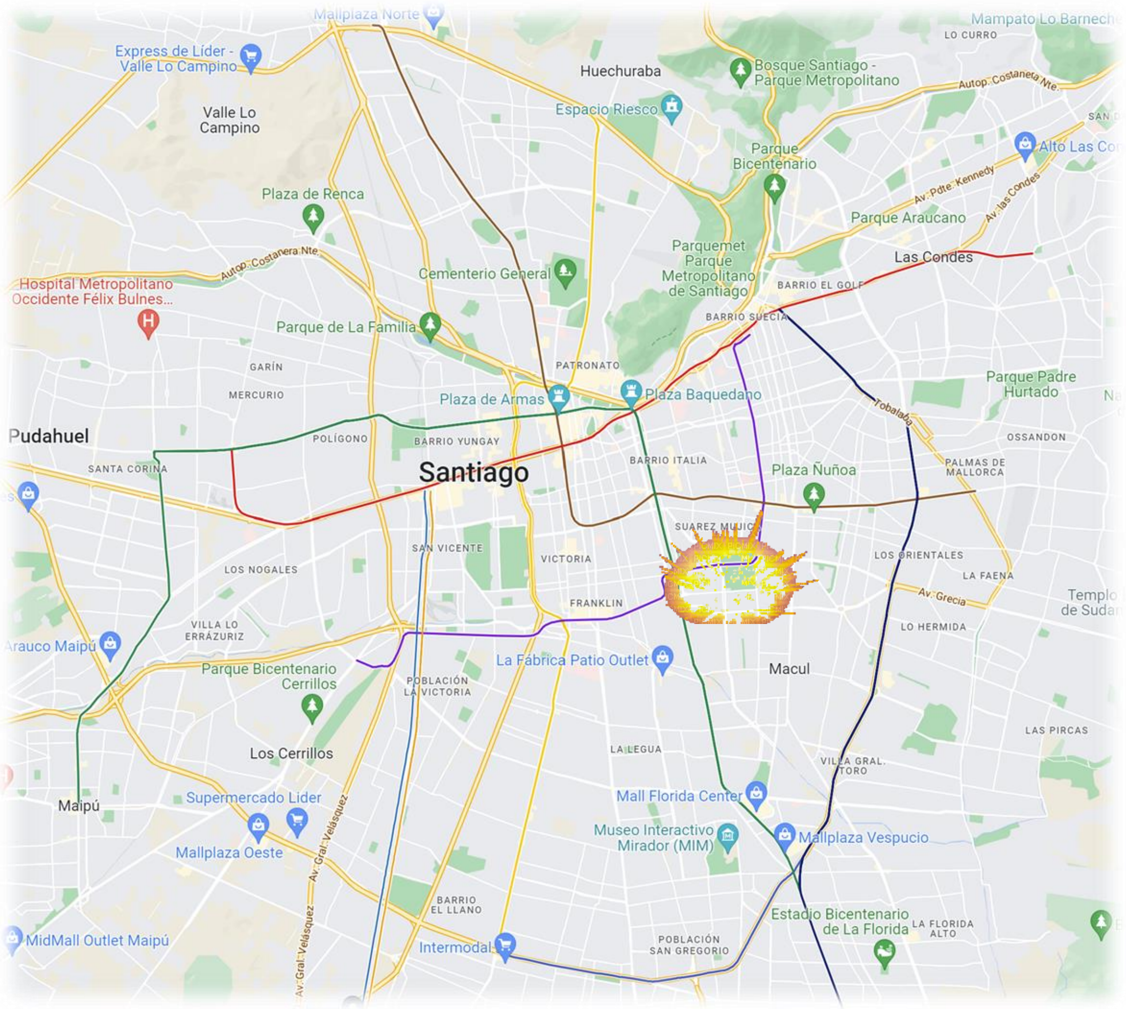
Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

Sistemas Urbanos Inteligentes

Redes para redes

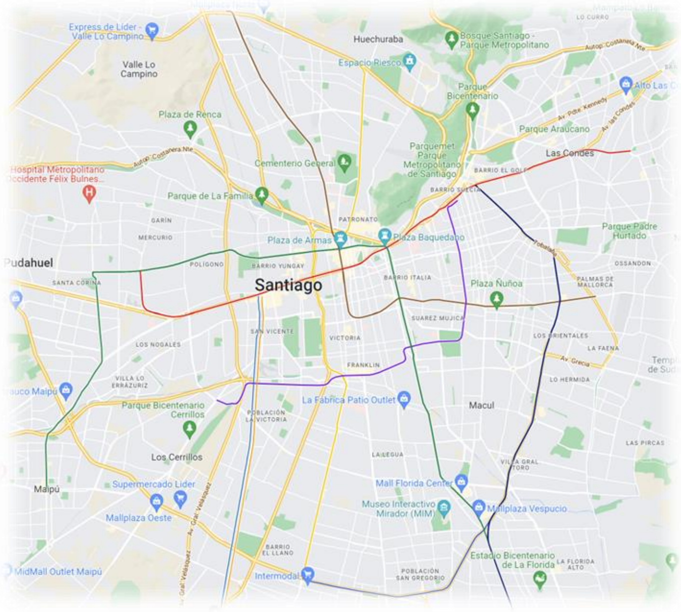
Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación

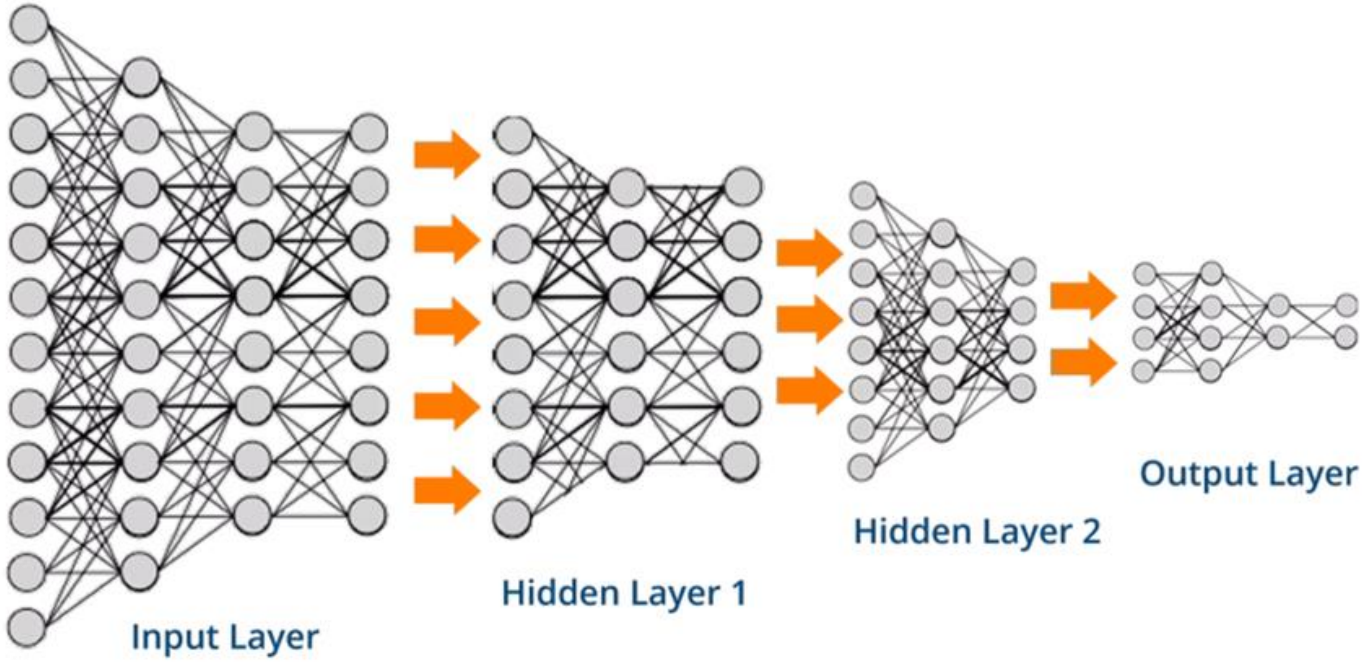
¿Cómo podemos predecir el efecto que genera un evento mayor en el tráfico?



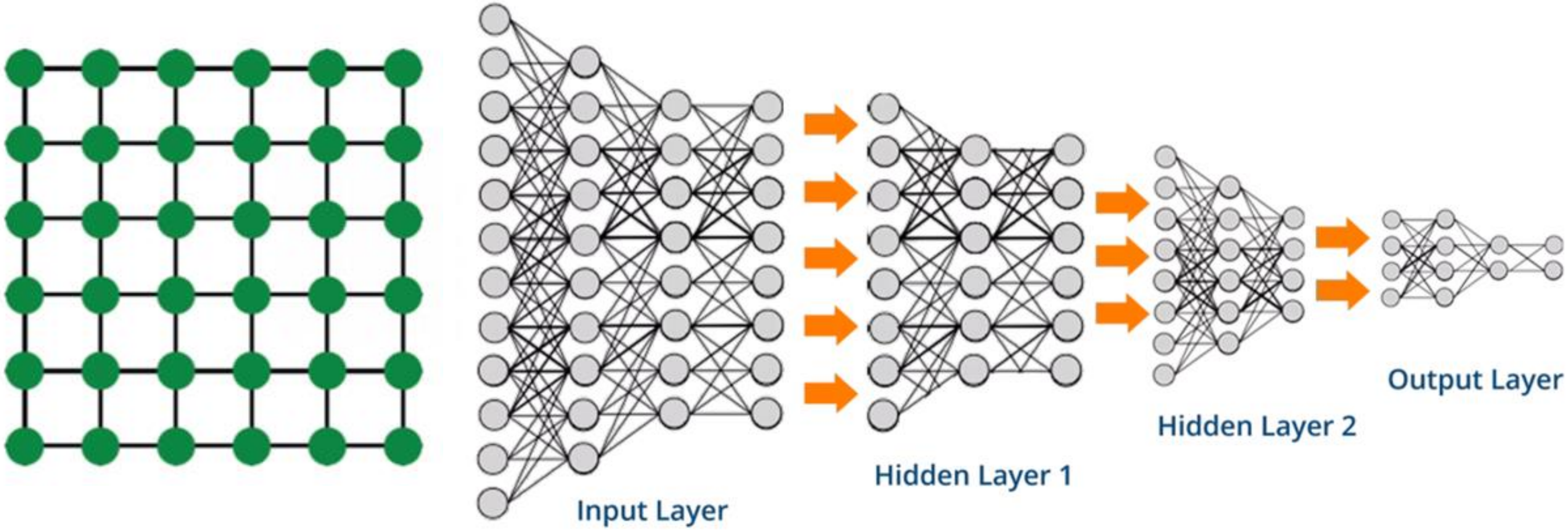
En base a lo que hemos visto hasta ahora, ¿cómo podríamos usar redes neuronales en este contexto?

Dependiendo de lo que se quiere hacer, no necesitamos nada muy sofisticado



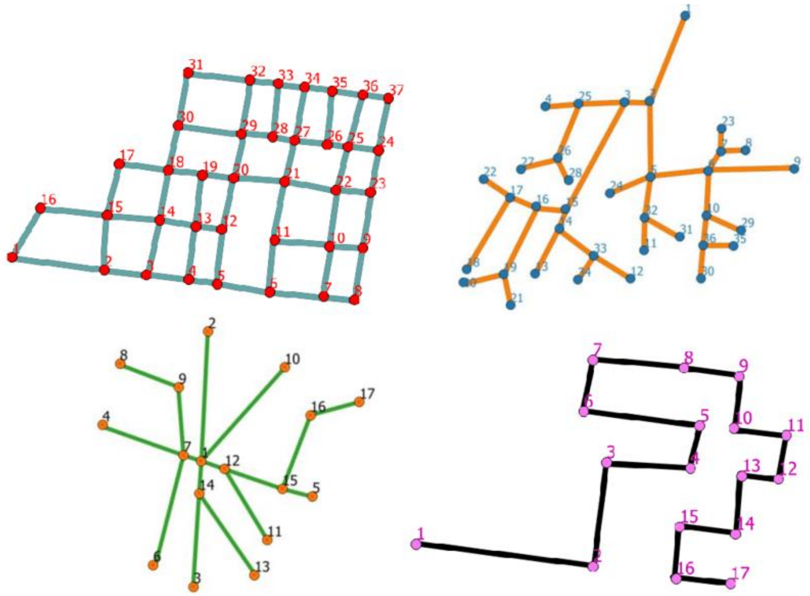


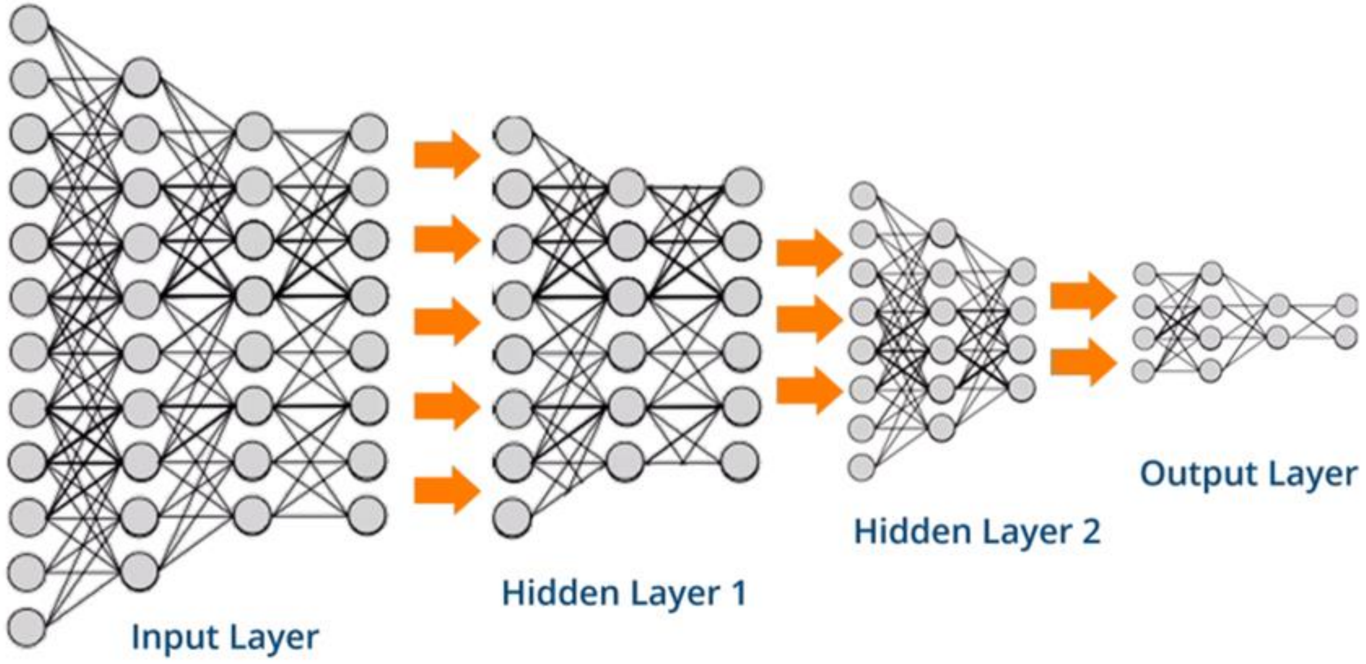
Pero si lo pensamos bien, este enfoque solo funciona para “grillas”



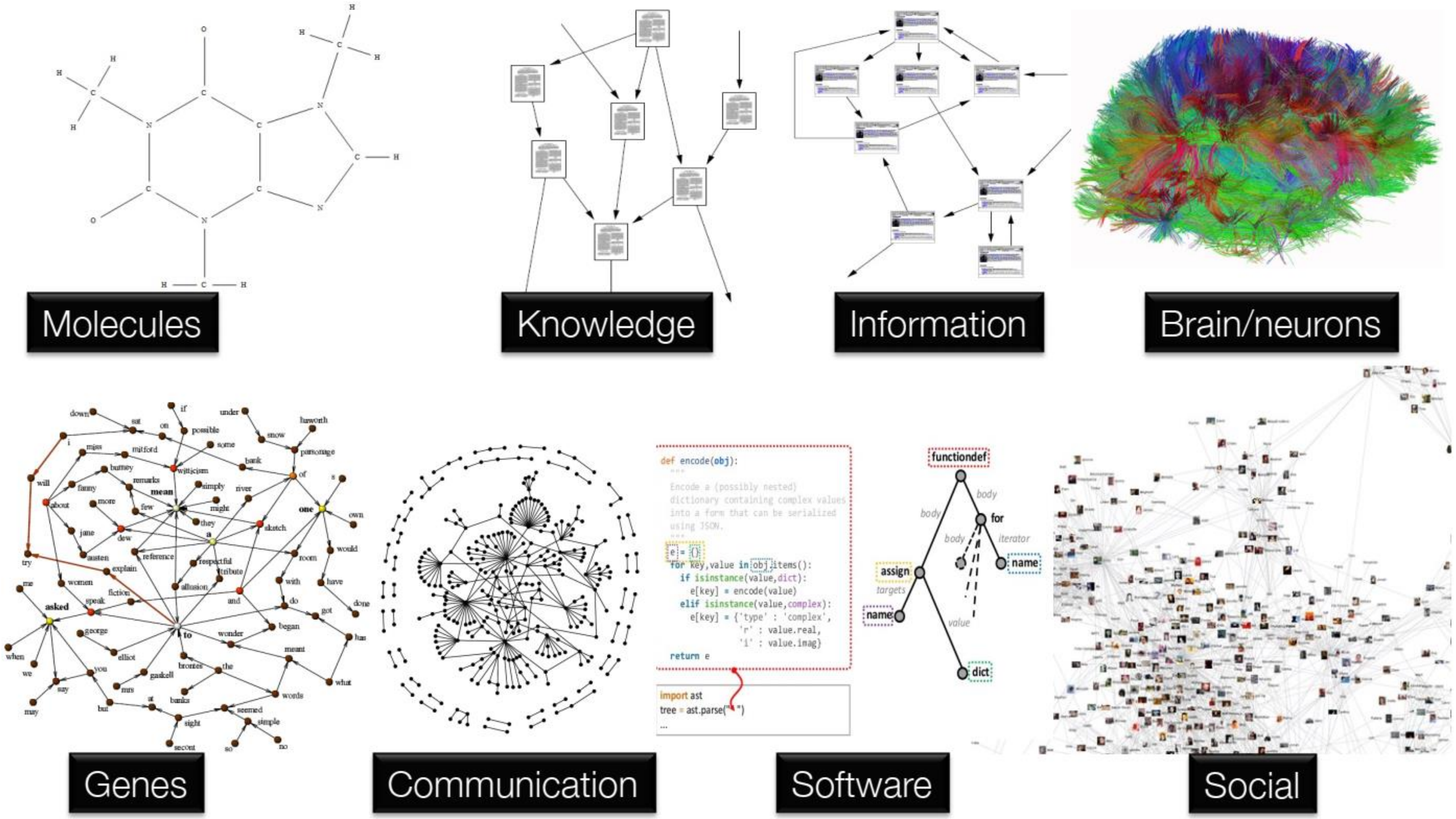
¿Qué limitación trae esto al uso de redes neuronales?

En realidad, lo que queremos es procesar datos estructurados de manera más general

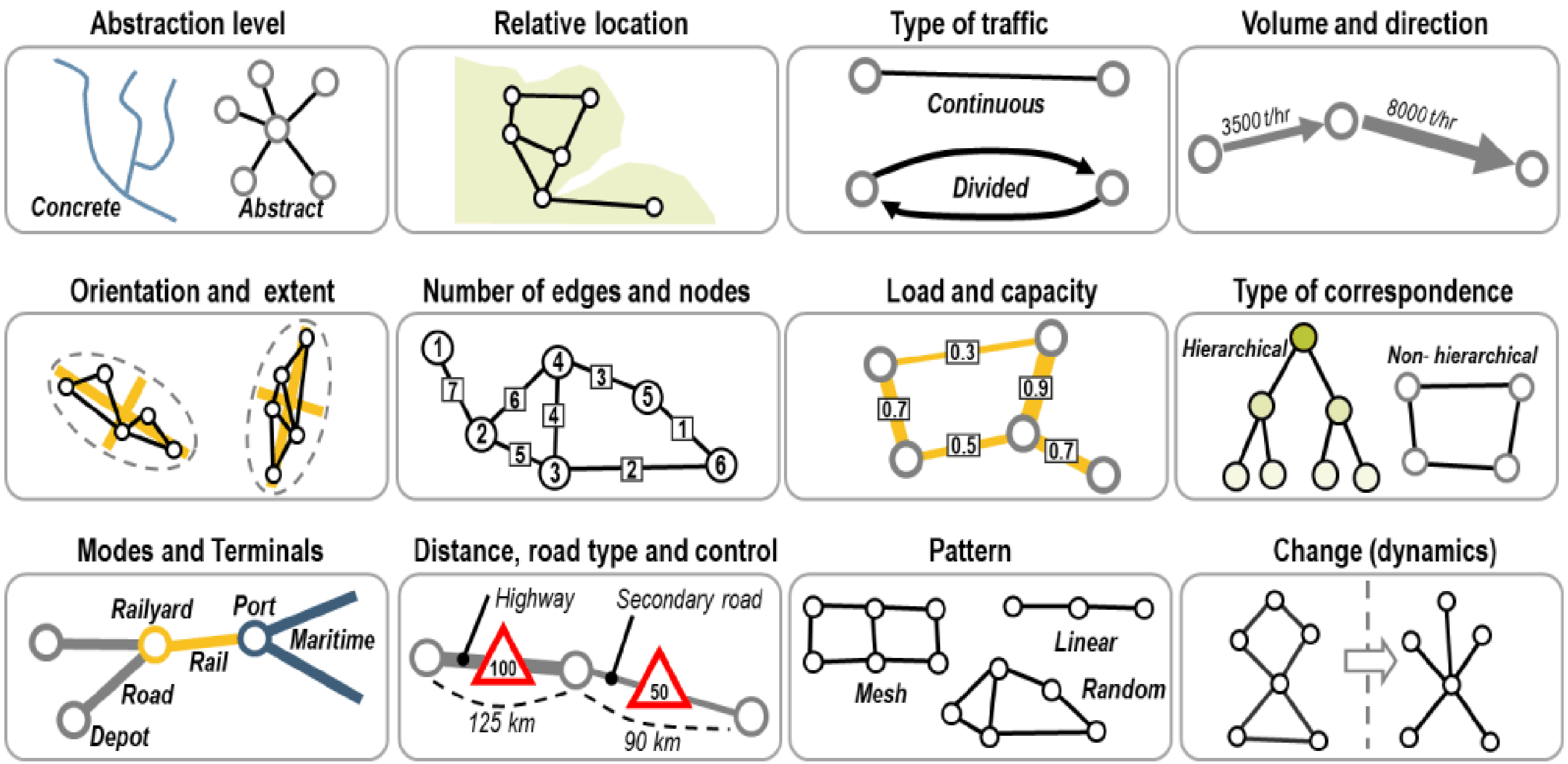




¿Cómo procesamos entonces datos/problemas que son modelados de forma natural como grafos?

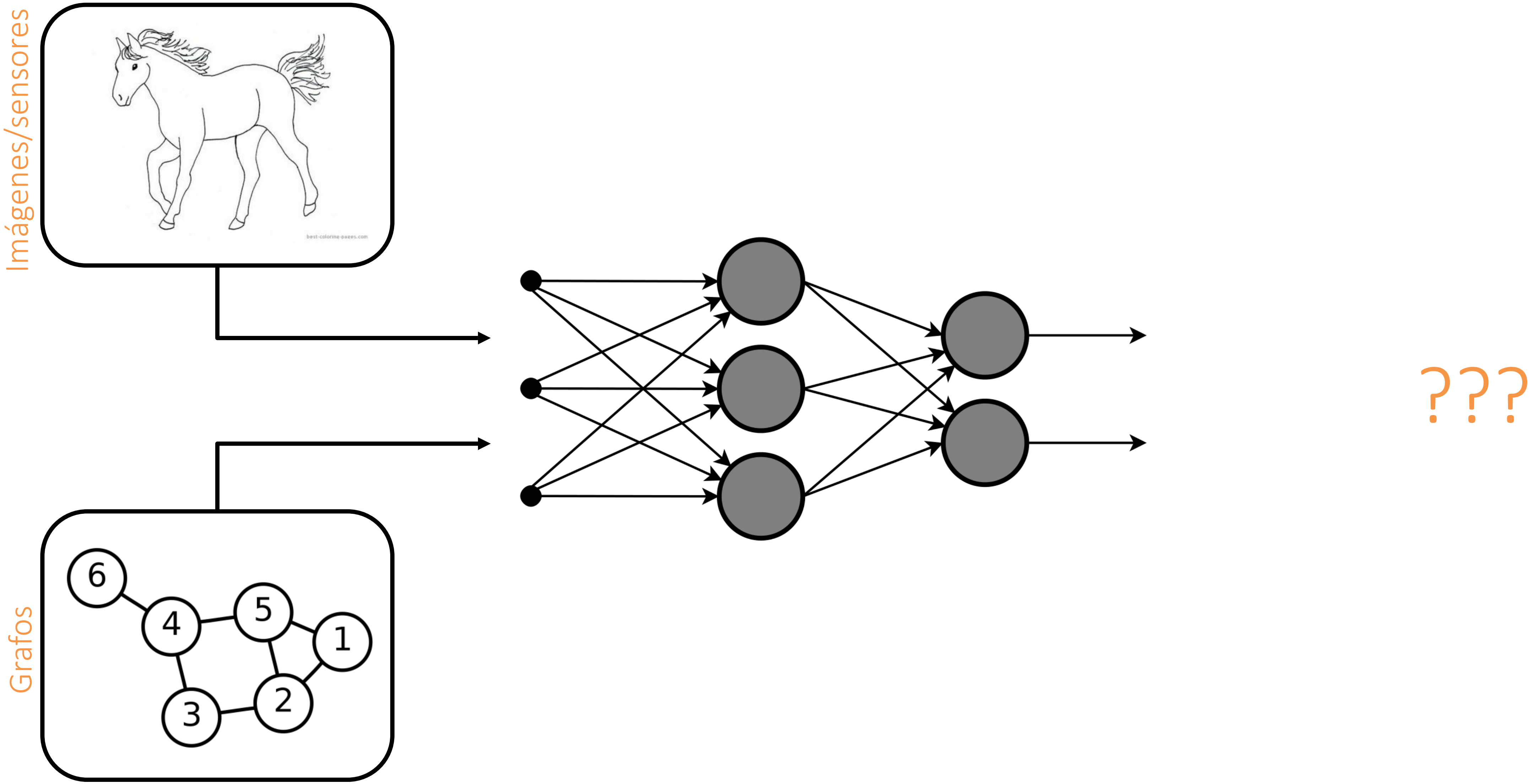


¿Cómo procesamos entonces datos/problemas que son modelados de forma natural como redes?

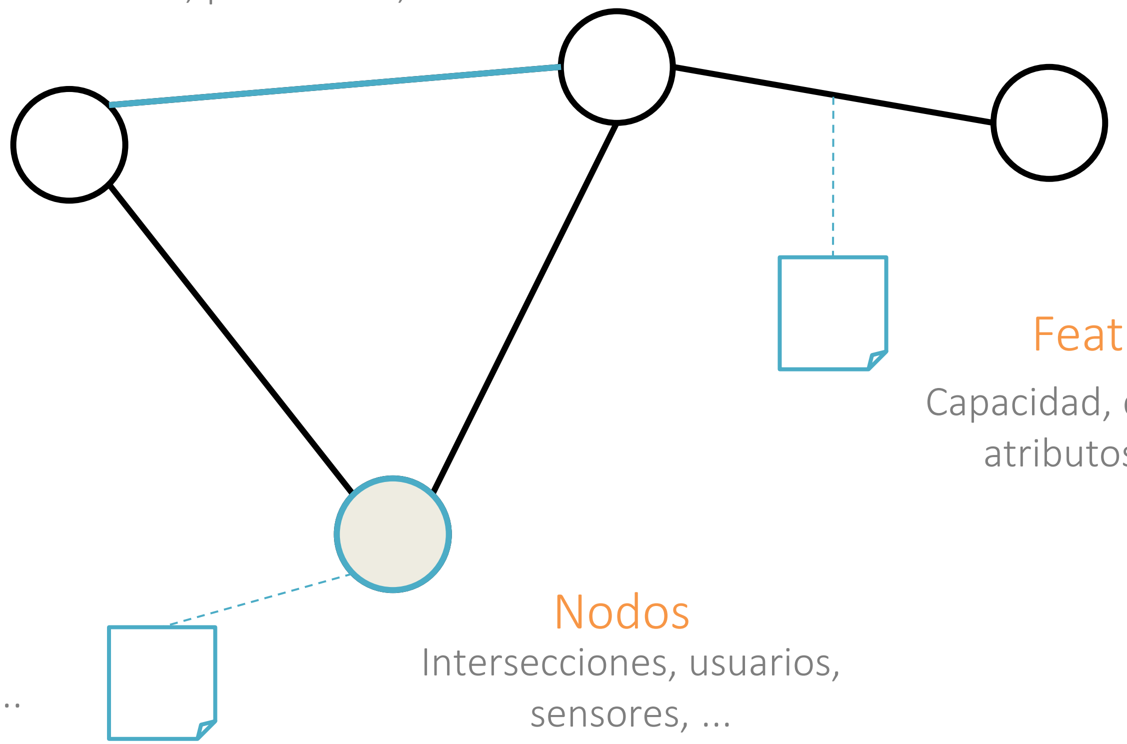


https://transportgeography.org/?page_id=719

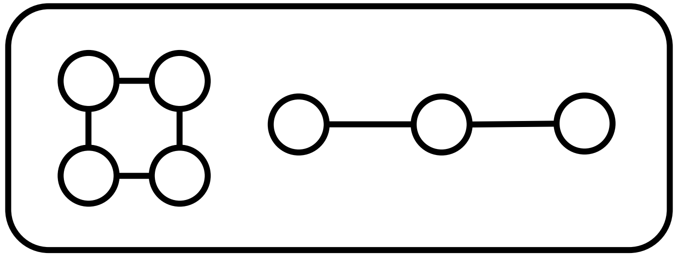
Estructura con menos orden hace que no sea evidente cómo hacerlo



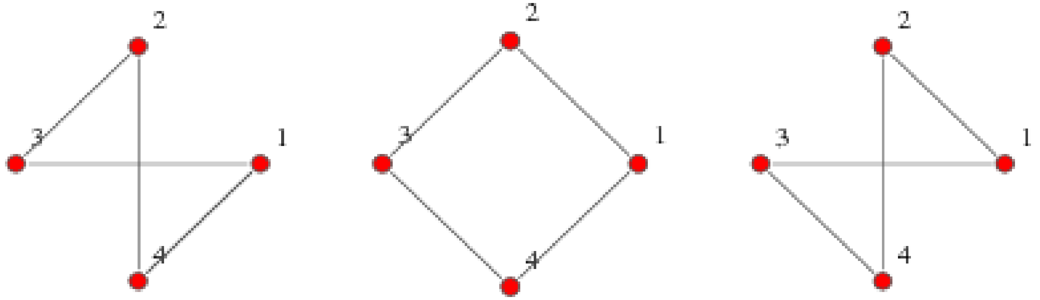
Partamos por lo básico, ¿qué es un grafo en este contexto?



Otros grafos



Para aprovechar lo ya cubierto en el curso, representaremos los grafos como matrices



$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Matrix de adyacencia

$$\mathbf{A} \in \mathbb{R}^{n \times n}$$

n nodos en el grafo

Feature de nodos

$$\mathbf{X} \in \mathbb{R}^{n \times d}$$

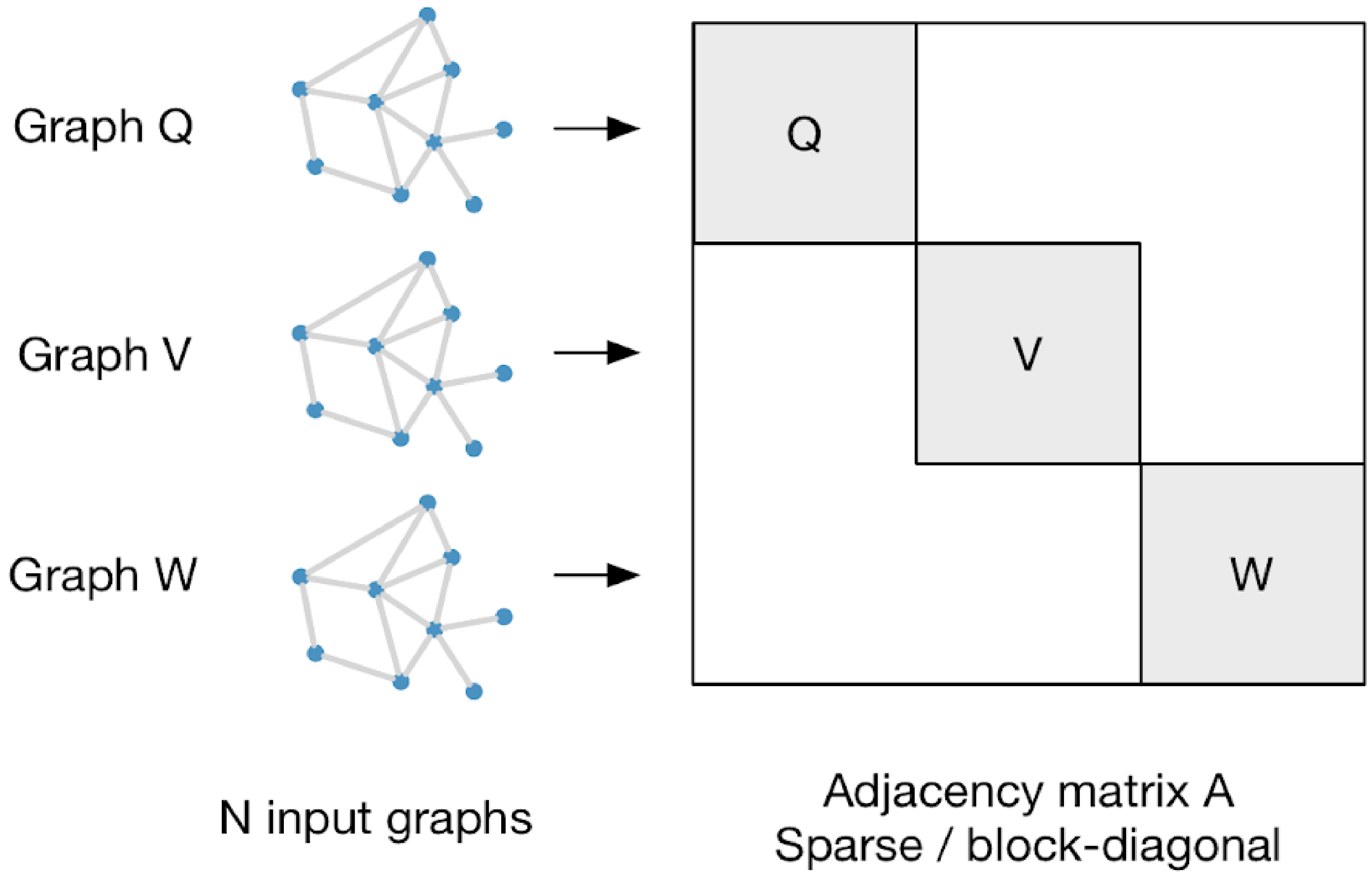
cada nodo tiene *d* features

Features de arcos

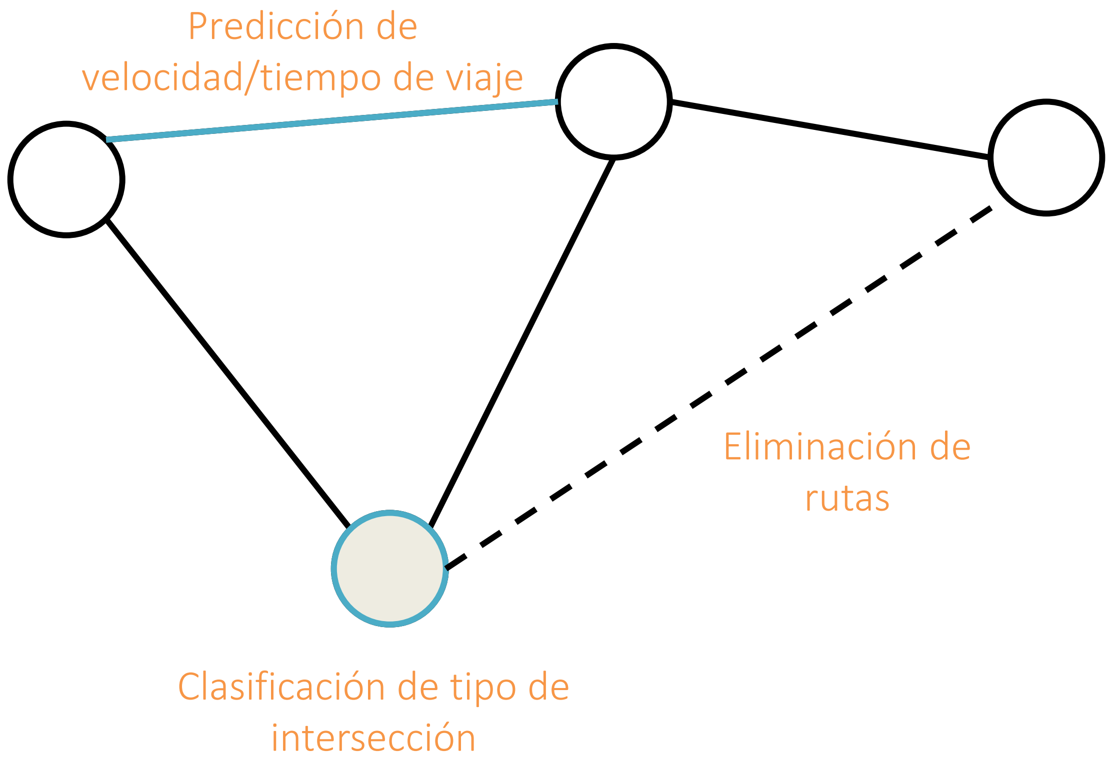
$$\mathbf{E} \in \mathbb{R}^{e \times f}$$

Cada arco tiene *f* features

Para aprovechar lo ya cubierto en el curso, representaremos los grafos como matrices



¿Qué podría interesarnos aprender sobre redes/grafos?

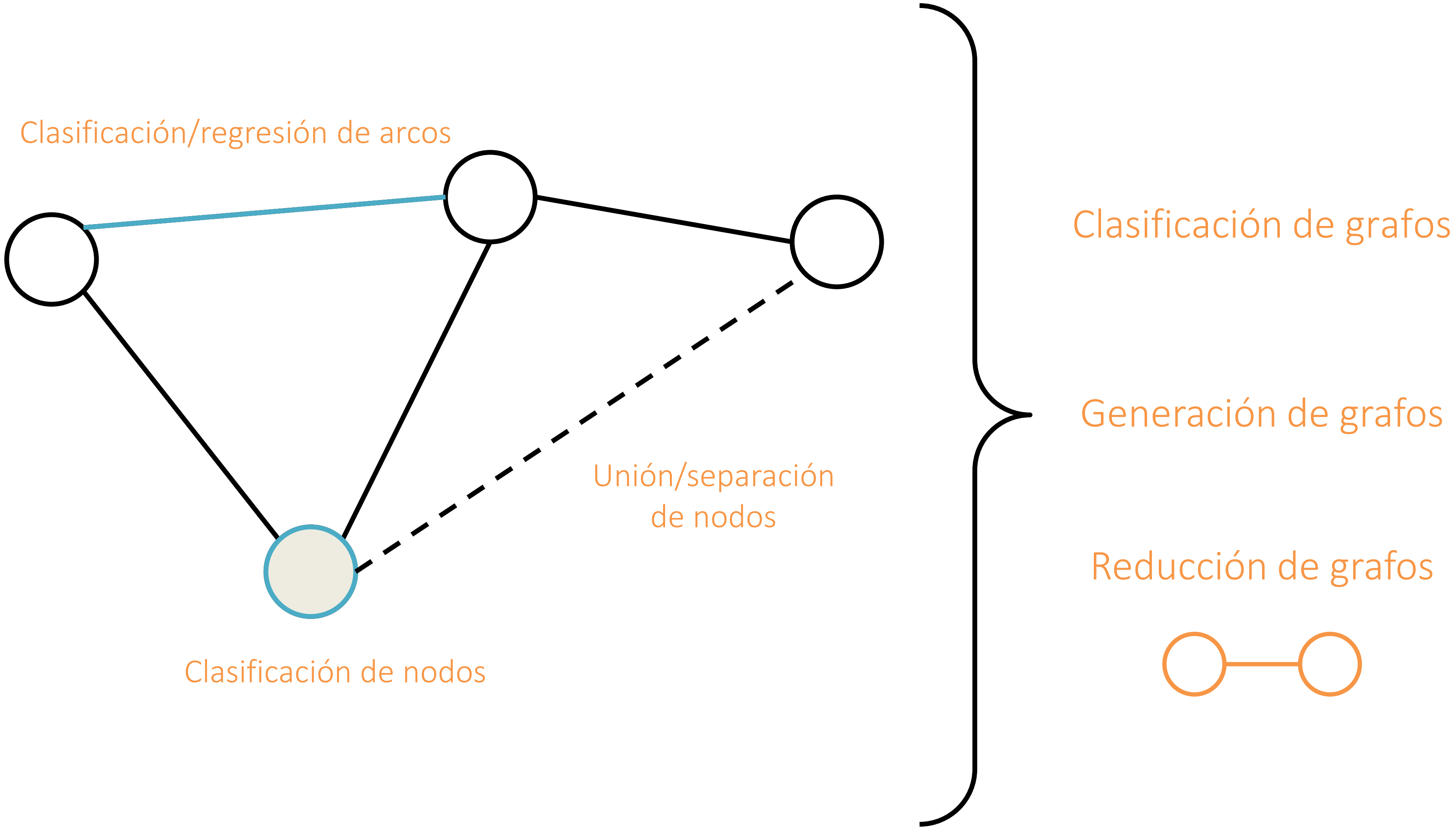


Predicción del estado del tráfico

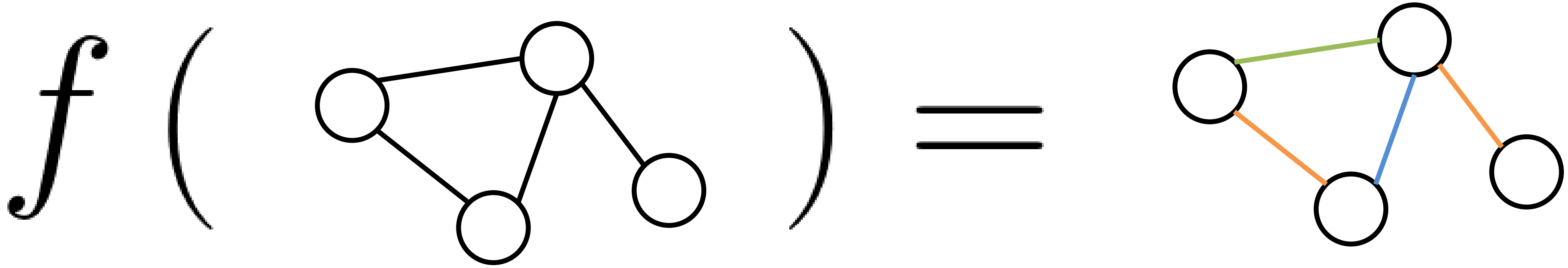
Generación de rutas

Reducción de redes

¿Qué podría interesarnos aprender sobre redes/grafos?

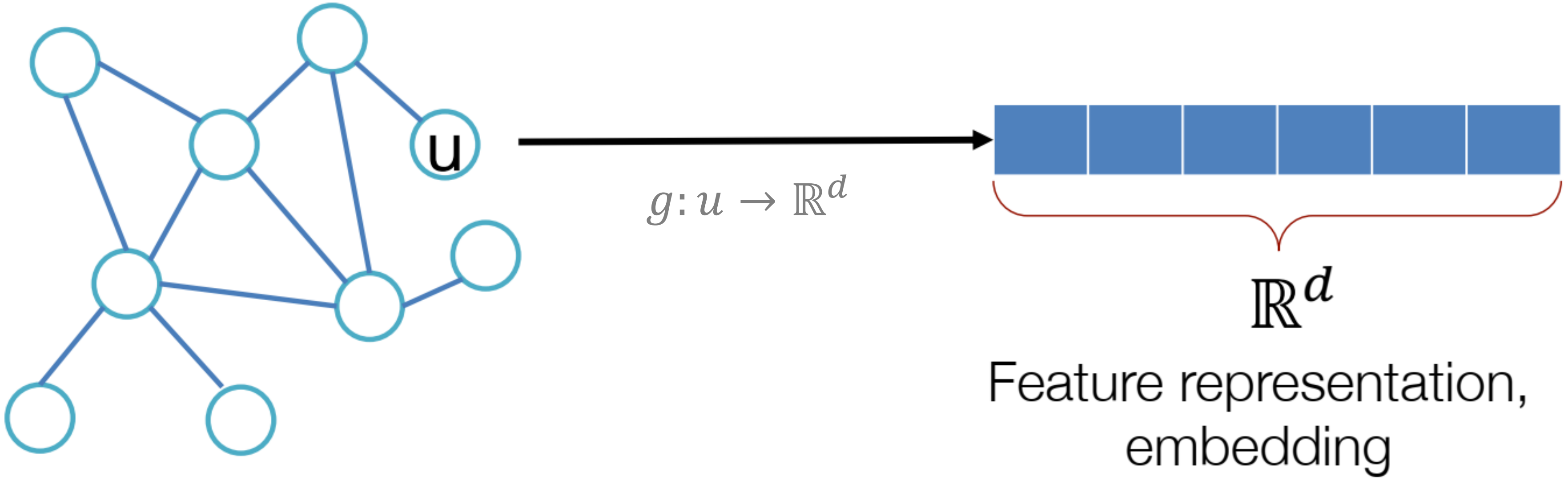


¿Y cómo podemos aprender en/sobre un grafo con redes neuronales?



Si queremos utilizar redes neuronales para modelar f, necesitamos que esta sea diferenciable, componible y escalable.

Al igual que antes, la clave está en el aprendizaje de representaciones



Podemos ver también esto como el aprendizaje de un mapeo de nodos a un espacio de embedding, donde nodos similares sean cercanos en esta espacio

Redes neuronales para grafos son un área de estudio extremadamente reciente

The Graph Neural Network Model

Franco Scarselli, Marco Gori, *Fellow, IEEE*, Ah Chung Tsoi, Markus Hagenbuchner, *Member, IEEE*, and Gabriele Monfardini

***Abstract*—Many underlying relationships among data in several areas of science and engineering, e.g., computer vision, molecular chemistry, molecular biology, pattern recognition, and data mining, can be represented in terms of graphs. In this paper, we propose a new neural network model, called graph neural network (GNN) model, that extends existing neural network methods for processing the data represented in graph domains. This GNN model, which can directly process most of the practically useful types of graphs, e.g., acyclic, cyclic, directed, and undirected, implements a function $\tau(\boldsymbol{G}, n) \in \mathbb{R}^m$ that maps a graph $\boldsymbol{G}$ and one of its nodes n into an m-dimensional Euclidean space. A supervised learning algorithm is derived to estimate the parameters of the proposed GNN model. The computational cost of the proposed algorithm is also considered. Some experimental results are shown to validate the proposed learning algorithm, and to demonstrate its generalization capabilities.**

ples a function τ that maps a graph G and one of its nodes n to a vector of reals[1]: $\tau(\boldsymbol{G}, n) \in \mathbb{R}^m$. Applications to a graphical domain can generally be divided into two broad classes, called *graph-focused* and *node-focused* applications, respectively, in this paper. In *graph-focused* applications, the function τ is independent of the node n and implements a classifier or a regressor on a graph structured data set. For example, a chemical compound can be modeled by a graph $\boldsymbol{G}$, the nodes of which stand for atoms (or chemical groups) and the edges of which represent chemical bonds [see Fig. 1(a)] linking together some of the atoms. The mapping $\tau(\boldsymbol{G})$ may be used to estimate the probability that the chemical compound causes a certain disease [13]. In Fig. 1(b), an image is represented by a region adjacency graph where nodes denote homogeneous regions of intensity of

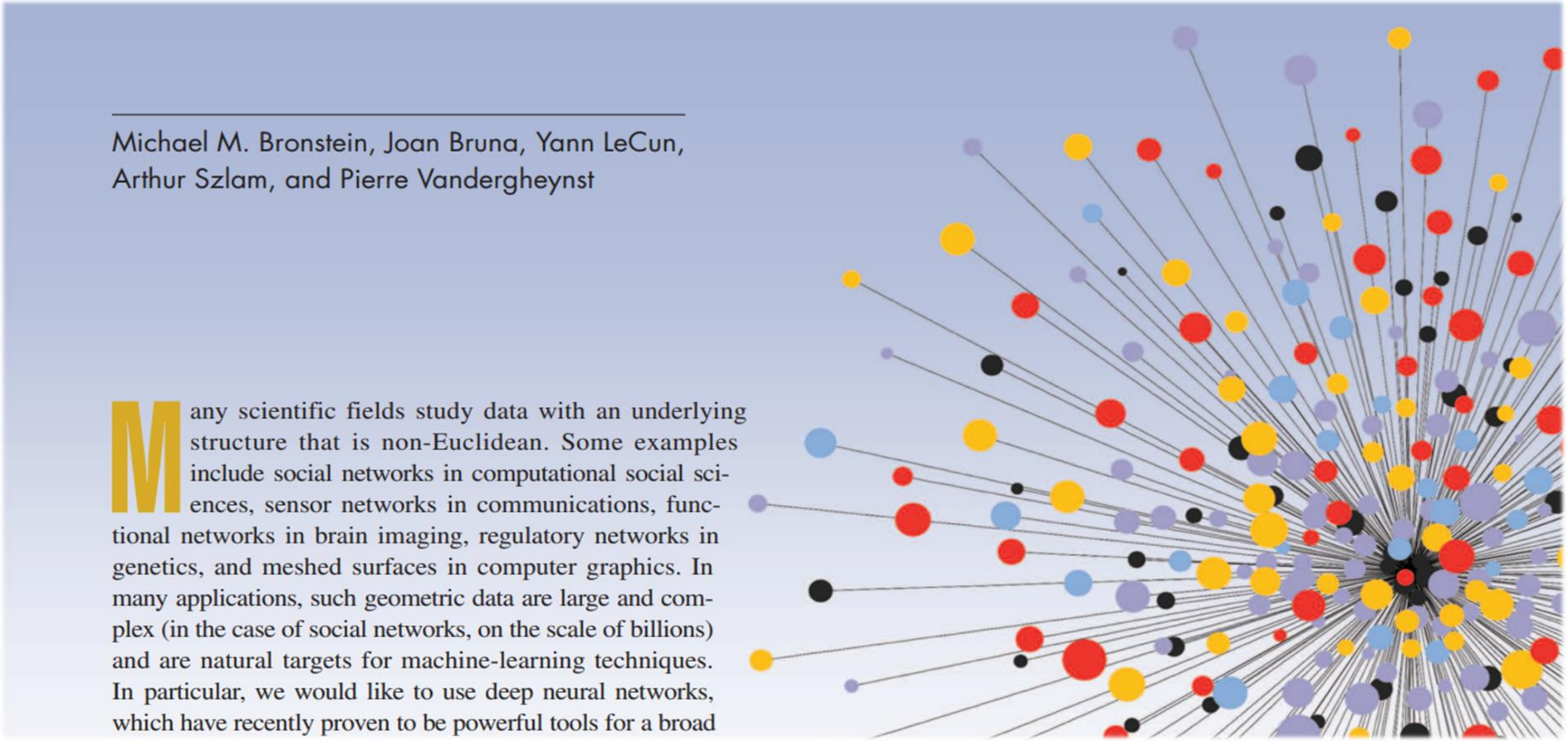
A pesar de lo reciente, es un área de gran actividad

TABLE III: Summary of RecGNNs and ConvGNNs. Missing values ("-") in pooling and readout layers indicate that the method only experiments on node-level/edge-level tasks.

Approach	Category	Inputs	Pooling	Readout	Time Complexity
GNN* (2009) [15]	RecGNN	A, X, X^e	-	a dummy super node	$O(m)$
GraphESN (2010) [16]	RecGNN	A, X	-	mean	$O(m)$
GGNN (2015) [17]	RecGNN	A, X	-	attention sum	$O(m)$
SSE (2018) [18]	RecGNN	A, X	-	-	-
Spectral CNN (2014) [19]	Spectral-based ConvGNN	A, X	spectral clustering+max pooling	max	$O(n^3)$
Henaff et al. (2015) [20]	Spectral-based ConvGNN	A, X	spectral clustering+max pooling		$O(n^3)$
ChebNet (2016) [21]	Spectral-based ConvGNN	A, X	efficient pooling	sum	$O(m)$
GCN (2017) [22]	Spectral-based ConvGNN	A, X	-	-	$O(m)$
CayleyNet (2017) [23]	Spectral-based ConvGNN	A, X	mean/graclus pooling	-	$O(m)$
AGCN (2018) [40]	Spectral-based ConvGNN	A, X	max pooling	sum	$O(n^2)$
DualGCN (2018) [41]	Spectral-based ConvGNN	A, X	-	-	$O(m)$
NN4G (2009) [24]	Spatial-based ConvGNN	A, X	-	sum/mean	$O(m)$
DCNN (2016) [25]	Spatial-based ConvGNN	A, X	-	mean	$O(n^2)$

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C. and Yu, P.S., 2019. *A comprehensive survey on graph neural networks*.

Gran parte del atractivo del enfoque se basa en su generalidad

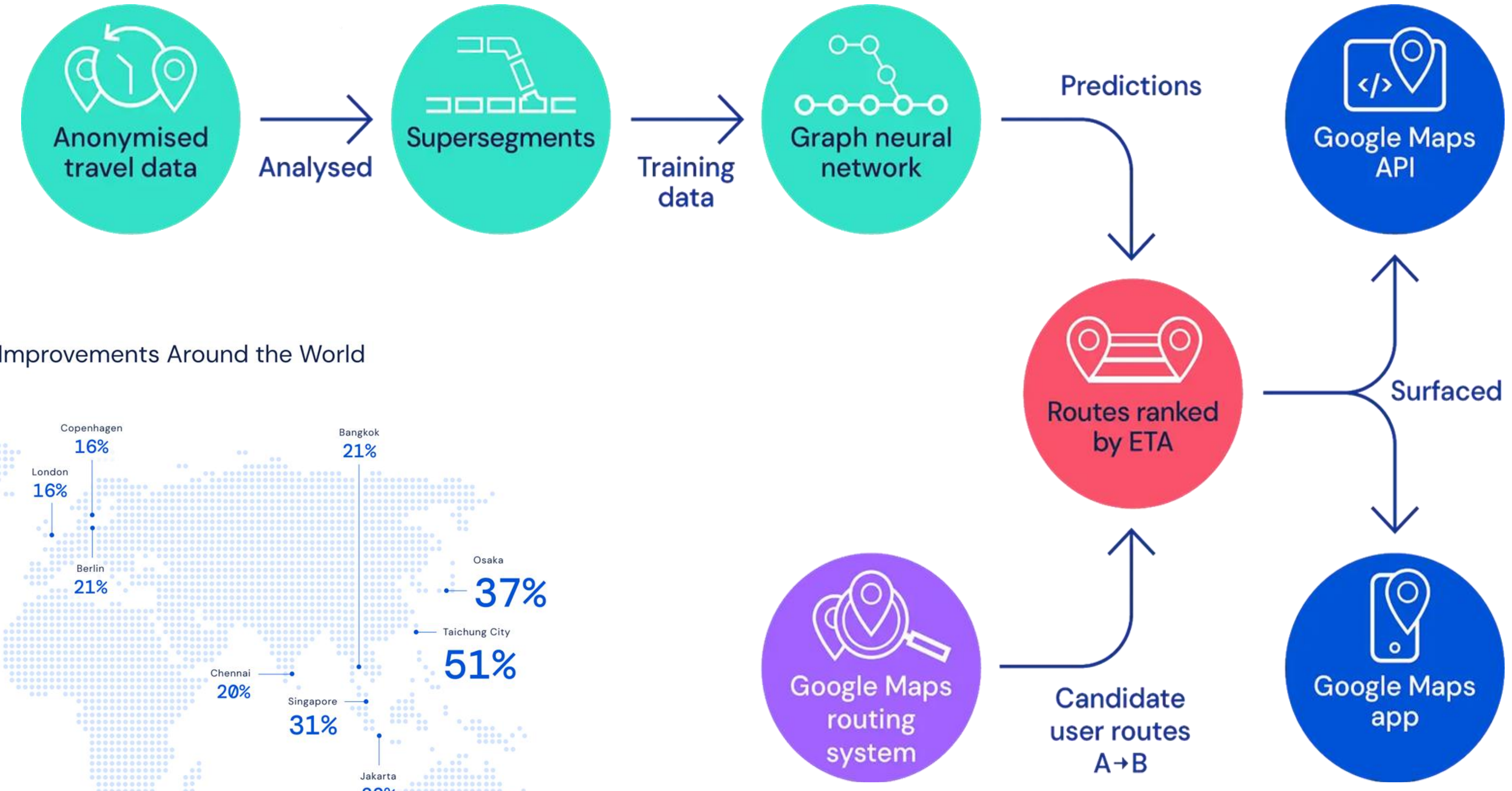


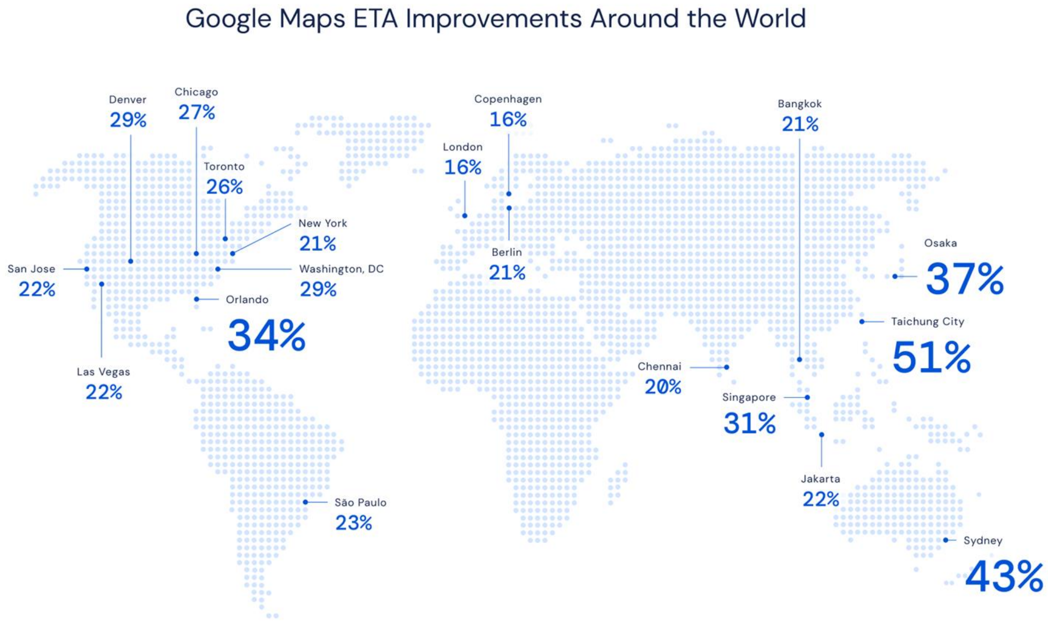
Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst

Many scientific fields study data with an underlying structure that is non-Euclidean. Some examples include social networks in computational social sciences, sensor networks in communications, functional networks in brain imaging, regulatory networks in genetics, and meshed surfaces in computer graphics. In many applications, such geometric data are large and complex (in the case of social networks, on the scale of billions) and are natural targets for machine-learning techniques. In particular, we would like to use deep neural networks, which have recently proven to be powerful tools for a broad

Bronstein, M.M. et al., 2017. *Geometric deep learning: going beyond Euclidean data*. IEEE Signal Processing Magazine, 34(4), pp.18-42.

Algunas aplicaciones: predicción de tráfico





Traffic prediction with advanced Graph Neural Networks
(https://deepmind.com/blog/article/traffic-prediction-with-advanced-graph-neural-networks)

Algunas aplicaciones: predicción de tráfico

Traffic prediction with advanced Graph Neural Networks
(https://deepmind.com/blog/article/traffic-prediction-with-advanced-graph-neural-networks)

Algunas aplicaciones: detección de fake news

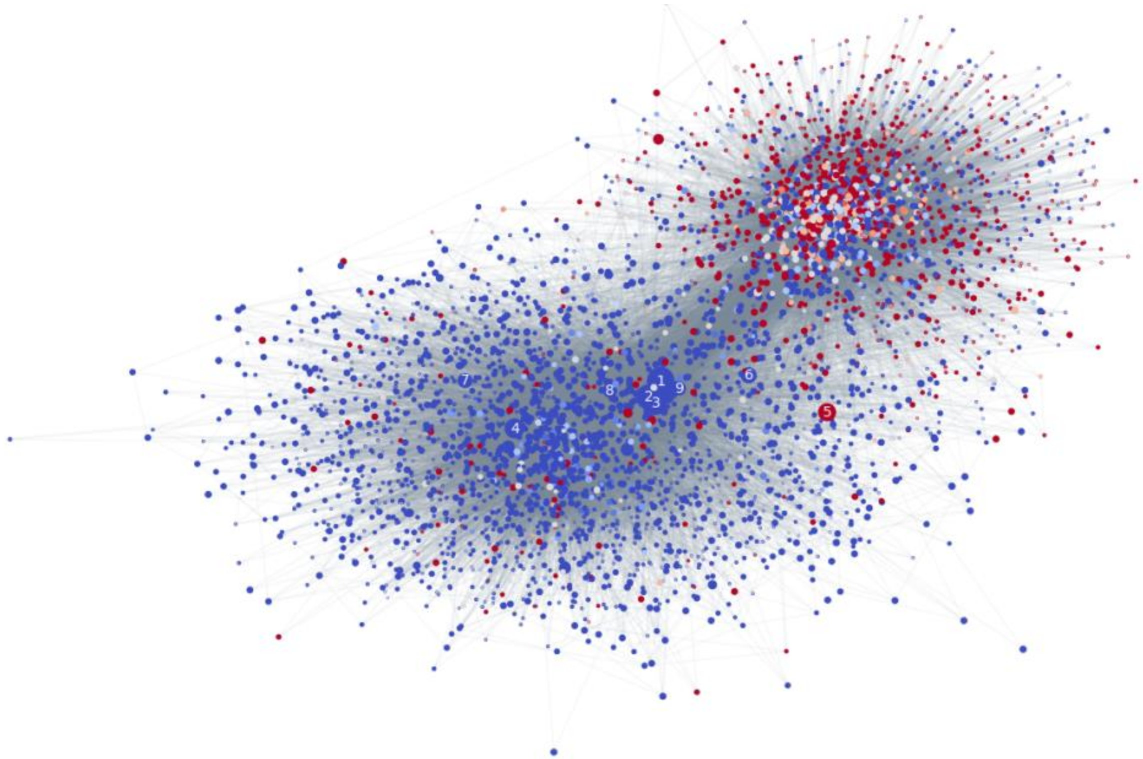
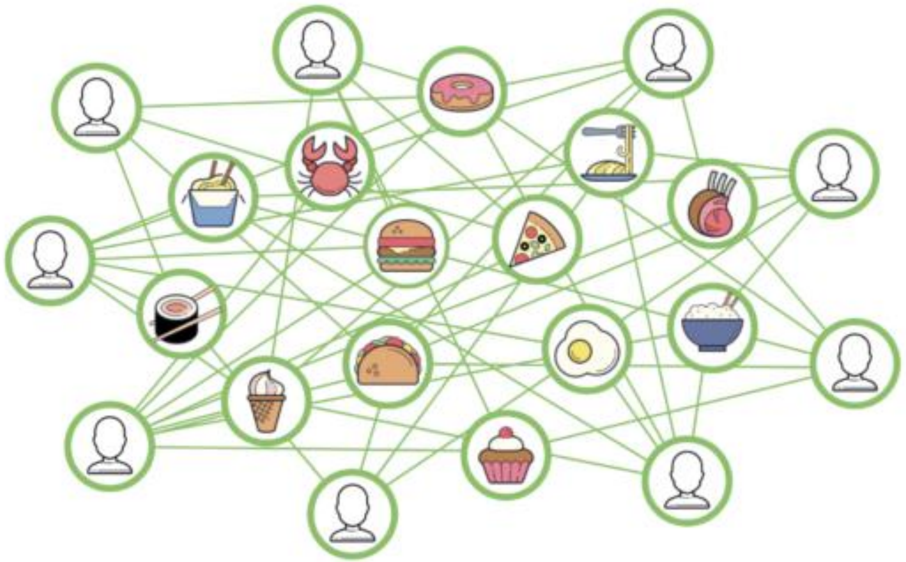
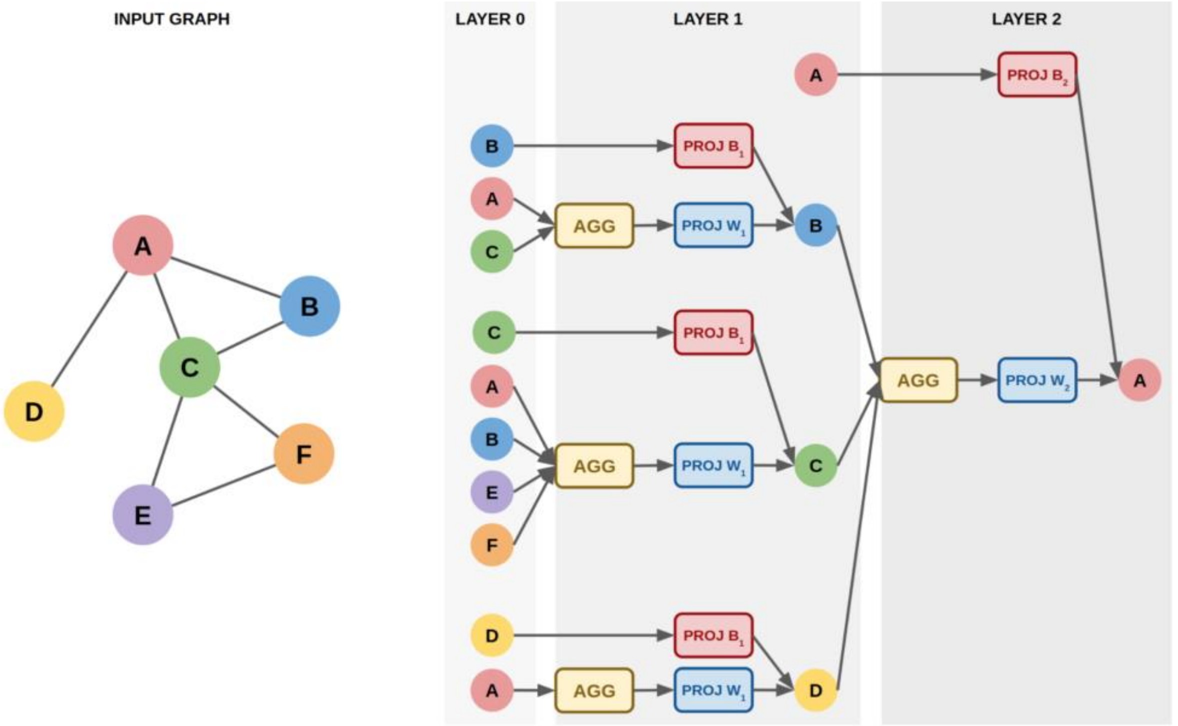


Figure 4: Subset of the Twitter network used in our study with estimated user credibility. Vertices represent users, gray edges the social connections. Vertex color and size encode the user credibility (blue = reliable, red = unreliable) and number of followers of each user, respectively. Numbers 1 to 9 represent the nine users with most followers.

Monti, F., Frasca, F., Eynard, D., Mannion, D. and Bronstein, M.M., 2019.
Fake News Detection on Social Media using Geometric Deep Learning.

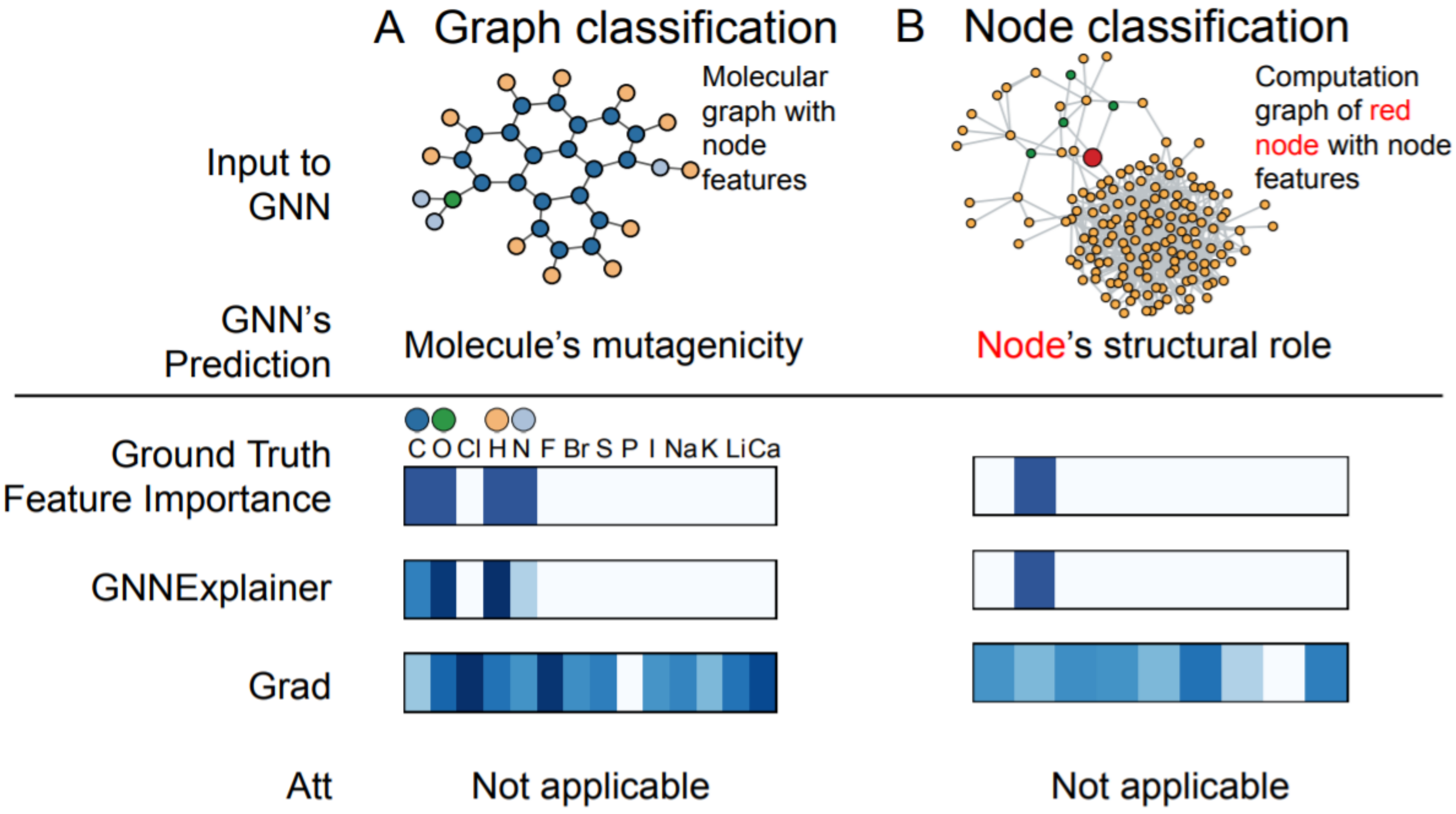
Algunas aplicaciones: sistemas de recomendación para Uber Eats





Ankit Jain, Isaac Liu, Ankur Sarda, and Piero Molino (2020). *Food Discovery with Uber Eats: Using Graph Learning to Power Recommendations*

Interpretabilidad se ve beneficiada por estructura



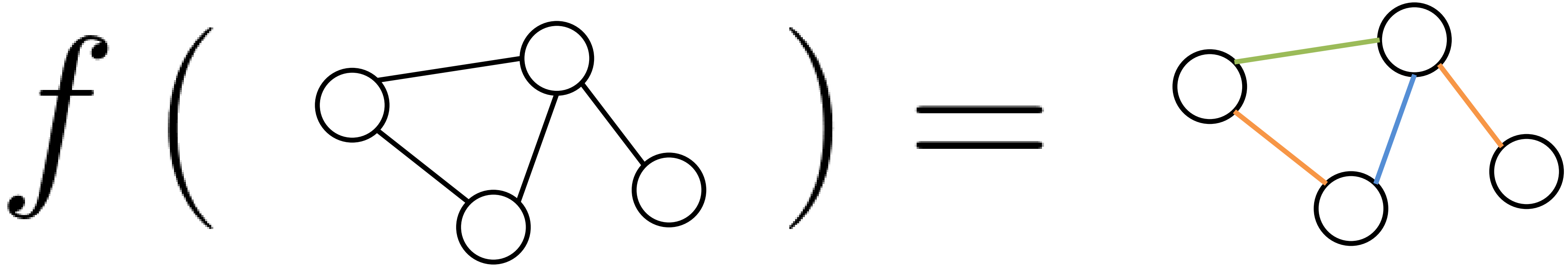
Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, Jure Leskovec (2019).
GNNExplainer: Generating Explanations for Graph Neural Networks.

Al igual que para las arquitecturas anteriores, existen multiples implementaciones disponibles

DGL
DEEP GRAPH LIBRARY

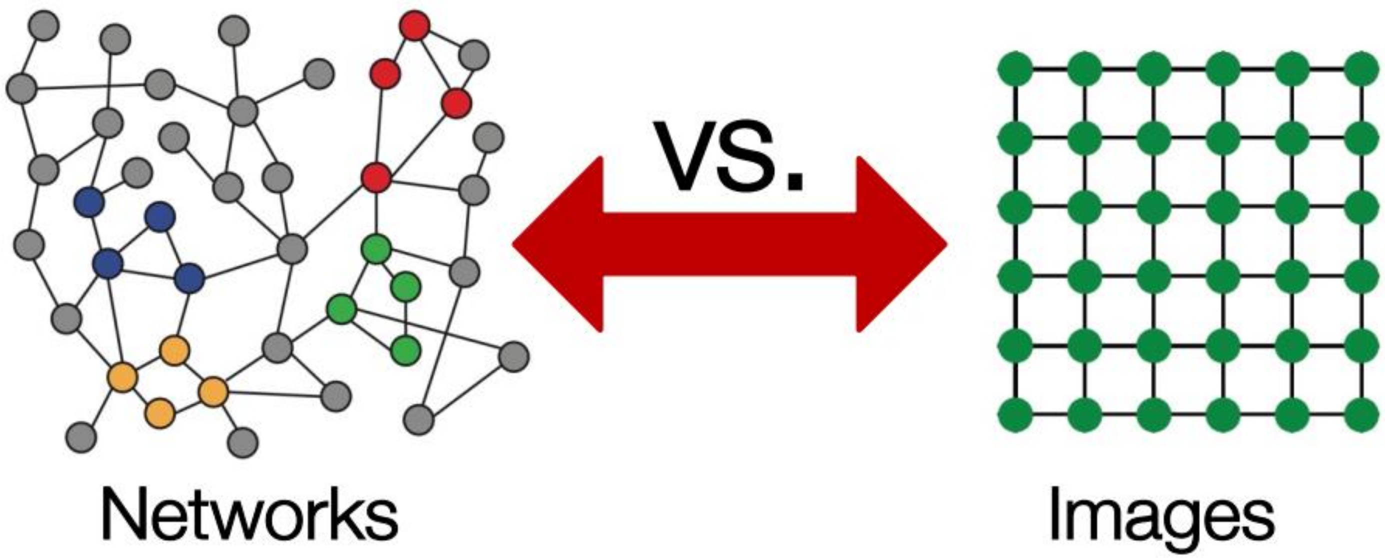
PyG

No olvidemos lo que buscamos



Si queremos utilizar redes neuronales para parametrizar f, necesitamos que esta sea differenciable, componible y escalable.

Ni por qué es difícil

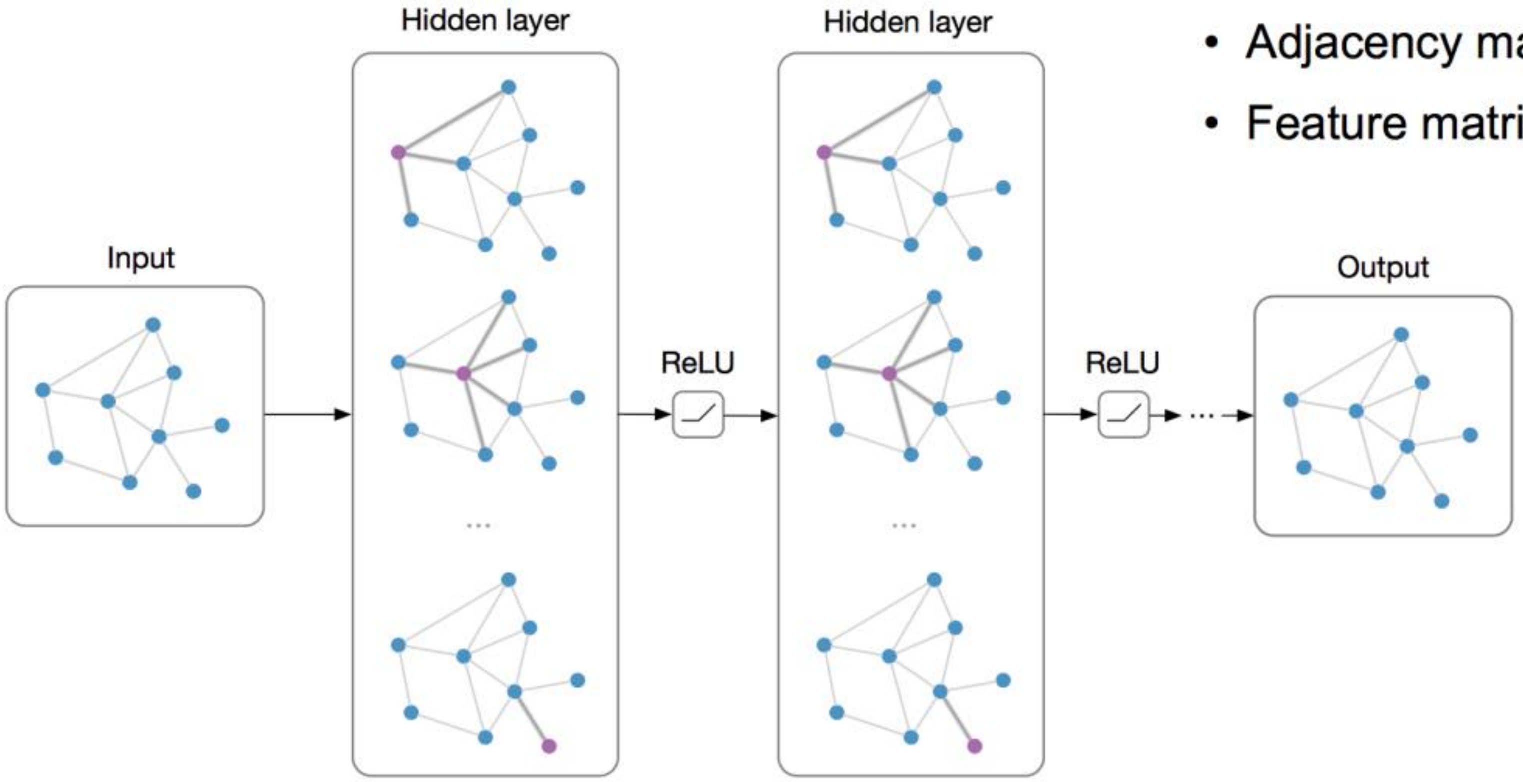


Los grafos son estructuras complejas: tamaño arbitrario, no existe el concepto de orden ni de punto de referencia, dinámicos, etc.

Hacia dónde vamos

Notation: $\mathcal{G} = (\mathbf{A}, \mathbf{X})$

- Adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$
- Feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$



Idea principal: pasar mensajes entre nodos y combinarlos

Otra perspectiva más ML: pasar mensajes entre nodos para refinar la representación

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

Sistemas Urbanos Inteligentes

Redes para redes

Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación