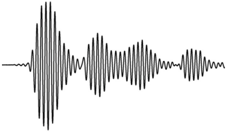
# Deep Learning & Applied AI

## Geometric deep learning

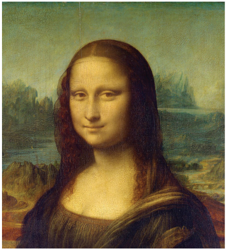
Emanuele Rodolà
rodola@di.uniroma1.it

SAPIENZA
UNIVERSITÀ DI ROMA

2nd semester a.y. 2022/2023 · May 03, 2023

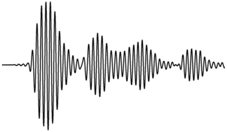


Audio signals

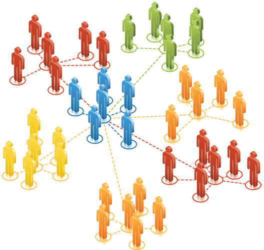


Images

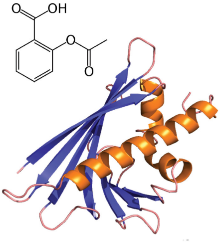
Geometric DL si occupa di estendere le capacità del DL ad altri domini non piatti come quelli qui mostrati



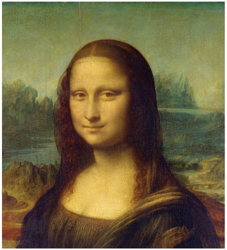
Audio signals



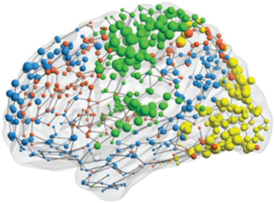
Social networks



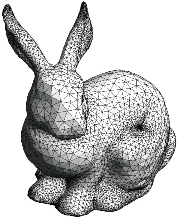

Molecules



Images

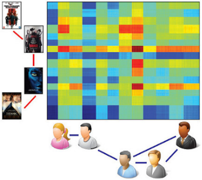


Functional networks

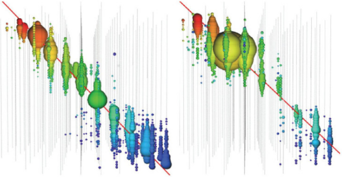


3D shapes

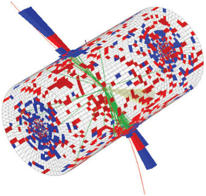
# Applications of geometric deep learning



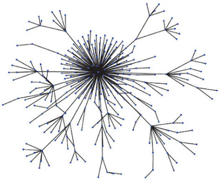
Recommender system



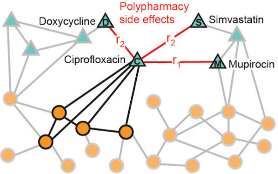
Neutrino detection



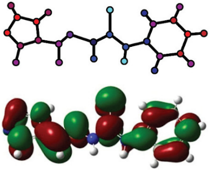
LHC



Fake news detection



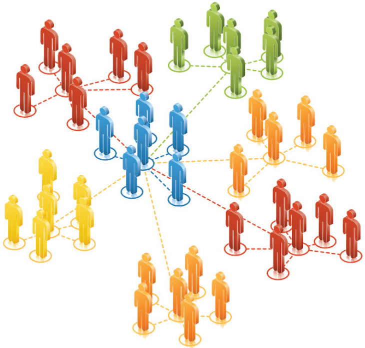

Drug repurposing



Chemistry

# Prototypical non-Euclidean objects

Manifolds

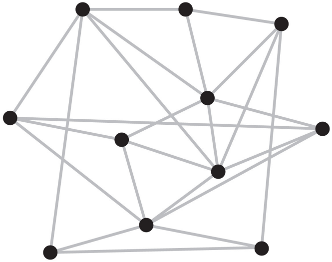


Graphs

Ci sono anche altri esempi ma questi due
sono i campi in cui il geometric DL è più utilizzato

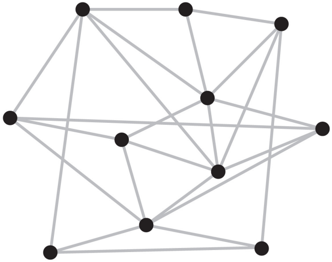
# Domain structure vs Data on a domain

# Domain structure vs Data on a domain

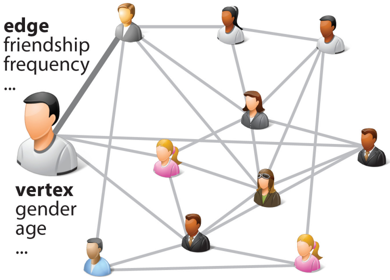


Domain structure

# Domain structure vs Data on a domain

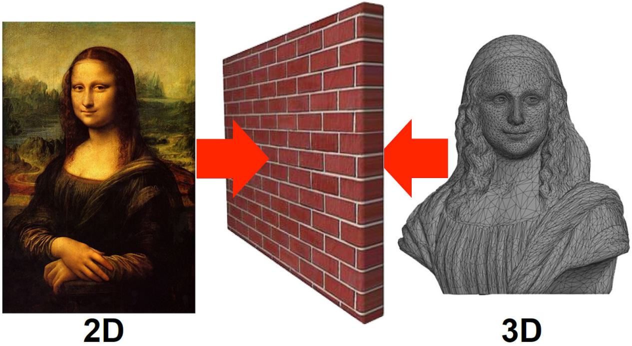


Domain structure




Data on a domain

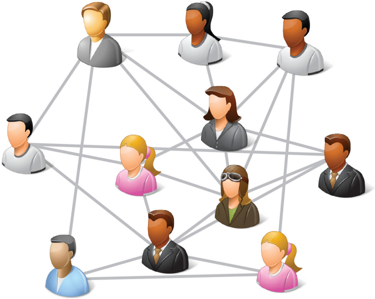
# Domain structure vs Data on a domain




only data | only structure

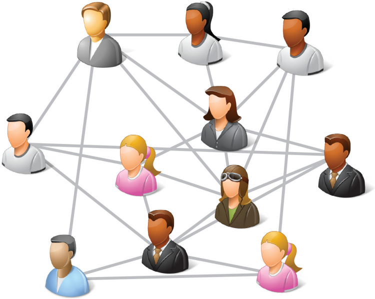
Con dati 2D spesso si hanno solo informazioni riguardanti le feature. D'altra parte, spesso con dati 3D si hanno solo informazioni riguardanti la struttura. Quindi il passaggio tra questi due tipi di dato non è immediato e richiede una rimodulazione delle tecniche di DL

# Fixed vs different domain

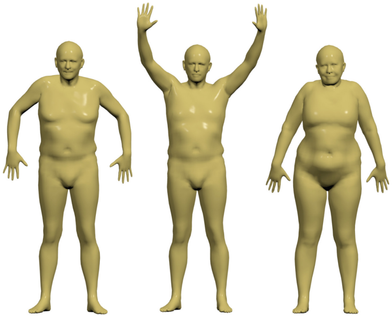


Social network
(fixed graph)

# Fixed vs different domain



Social network
(fixed graph)



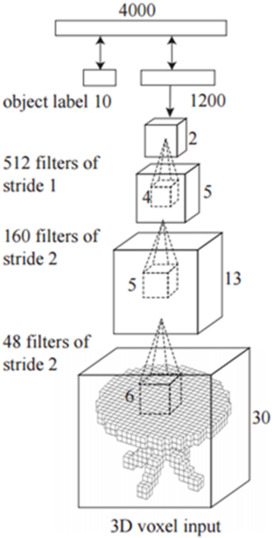
3D shapes
(different manifolds)

(idea semplificata) in un grafo il dominio
è sempre lo stesso, come in un'immagine...

... mentre con le forme 3D il dominio cambia
(non esiste un dominio fisso in grado di descrivere
tutte le possibili varianti di un oggetto 3D, perché
la sua struttura cambia)

# 3D ShapeNets

- Volumetric representation (shape = binary voxels on 3D grid)



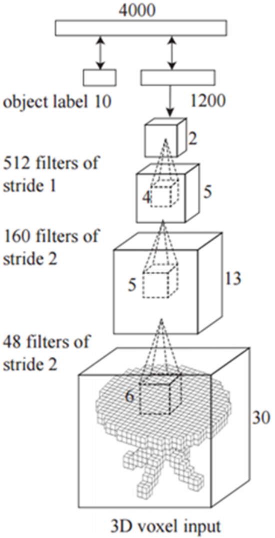

Convolutional deep belief network

Wu et al, "3D ShapeNets: A Deep Representation for Volumetric Shapes" 2015

# 3D ShapeNets

- Volumetric representation (shape = binary voxels on 3D grid)
- 3D convolutional network

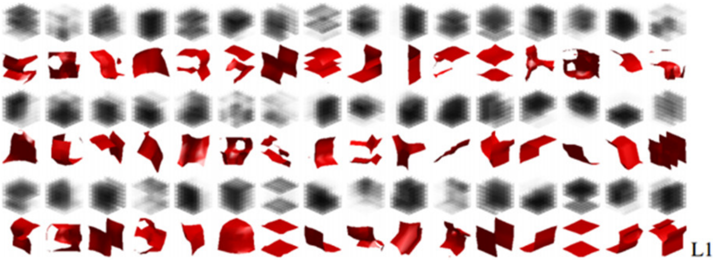
In pratica, è possibile realizzare l'operatore di convoluzione anche a forme 3D, adattando il kernel alla dimensione




Convolutional deep belief network

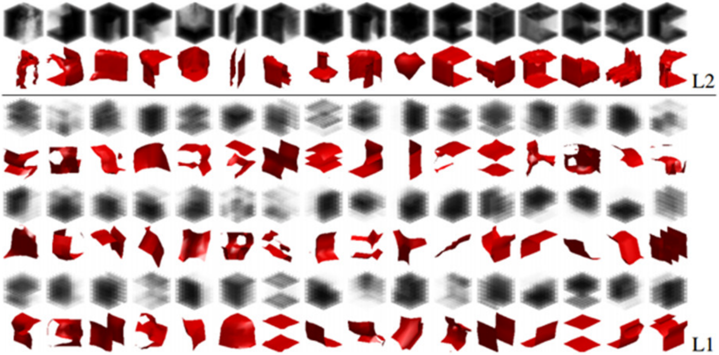
Wu et al, "3D ShapeNets: A Deep Representation for Volumetric Shapes" 2015

# Learned features: 3D primitives




Wu et al, "3D ShapeNets: A Deep Representation for Volumetric Shapes" 2015

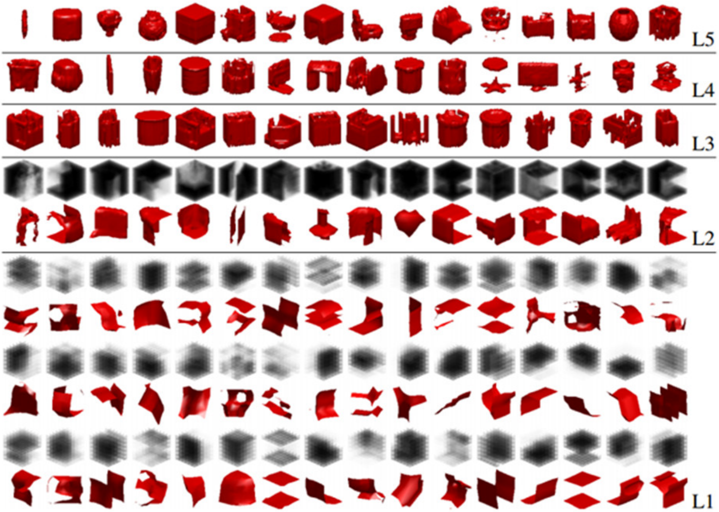
# Learned features: 3D primitives




Wu et al, "3D ShapeNets: A Deep Representation for Volumetric Shapes" 2015

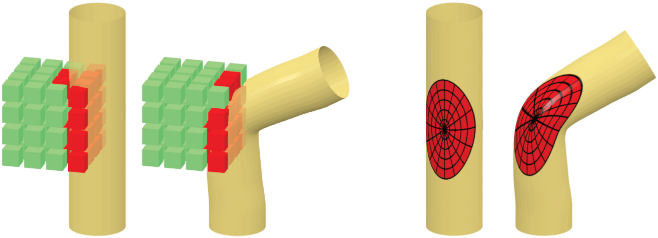
# Learned features: 3D primitives

Così come in 2D, anche in 3D vengono catturate le feature su diversi piani di complessità




Wu et al, "3D ShapeNets: A Deep Representation for Volumetric Shapes" 2015

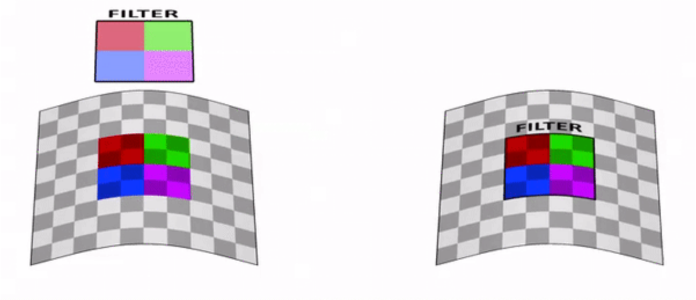
# Challenges of geometric deep learning



Extrinsic Intrinsic

Così come per il 2D, anche per il 3D vogliamo che le operazioni siano invarianti rispetto ad alcune trasformazioni. Tuttavia, in 3D sbucano fuori nuove sfide, come questa.
Nel caso di oggetti 3D deformabili (con quelli NON deformabili bastano le convoluzioni 3D),vogliamo che il kernel subisca le stesse trasformazioni/acquisisca le stesse proprietà dell'oggetto sul quale viene applicato

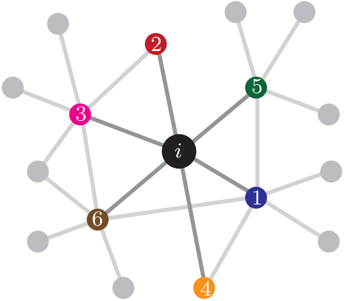
# Extrinsic vs Intrinsic




**Extrinsic** **Intrinsic**

# Local ambiguity

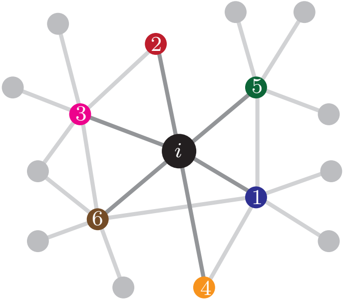
Unlike images, there is **no** canonical ordering of the domain points.



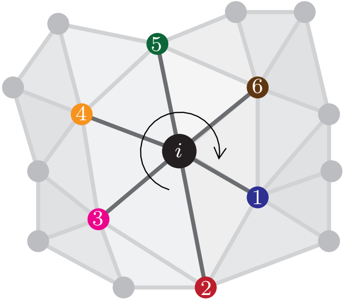

Graph (permutation)

# Local ambiguity

Unlike images, there is **no** canonical ordering of the domain points.




Graph (permutation)

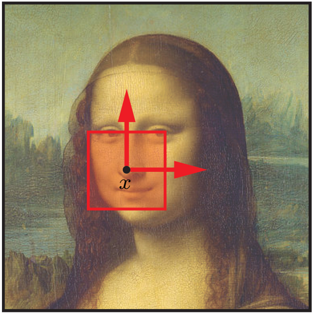



Mesh (rotation)

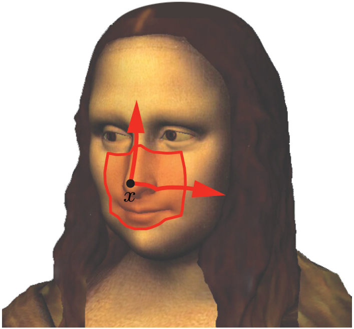
In un grafo 2D posso riordinare i nodi come voglio e così creare ogni volta un grafo diverso

in un grafo 3D non posso riordinare i nodi perché il grafo possiede già una forma, ma posso cambiare il nodo iniziale e questo genera diverse orientamenti dello stesso grafo

# Non-Euclidean convolution?



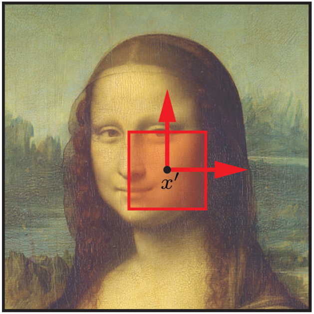

Euclidean



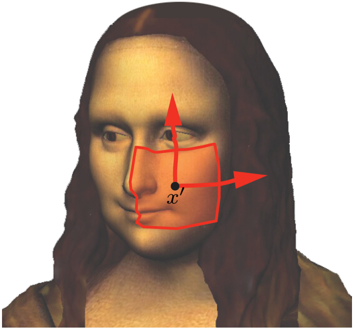

Non-Euclidean

Le coordinate 2D non funzionano su un oggetto 3D, perché un movimento base in 2D non si può tradurre in maniera universale in 3D. Ancora peggio se...

# Non-Euclidean convolution?

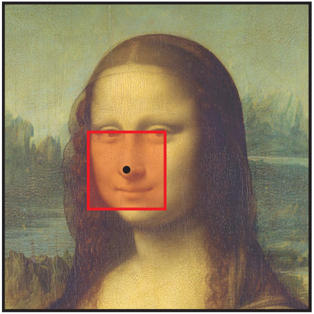



Euclidean

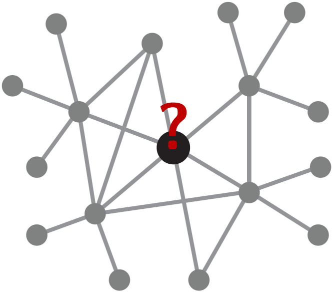



Non-Euclidean

# Non-Euclidean convolution?



Image





Graph

... parliamo di grafi

# Suggested reading

Bronstein et al, 2016

"Geometric deep learning: going beyond Euclidean data"

https://arxiv.org/abs/1611.08097