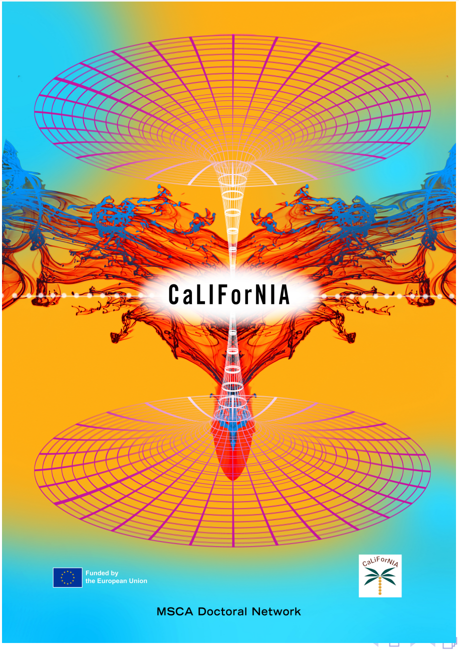
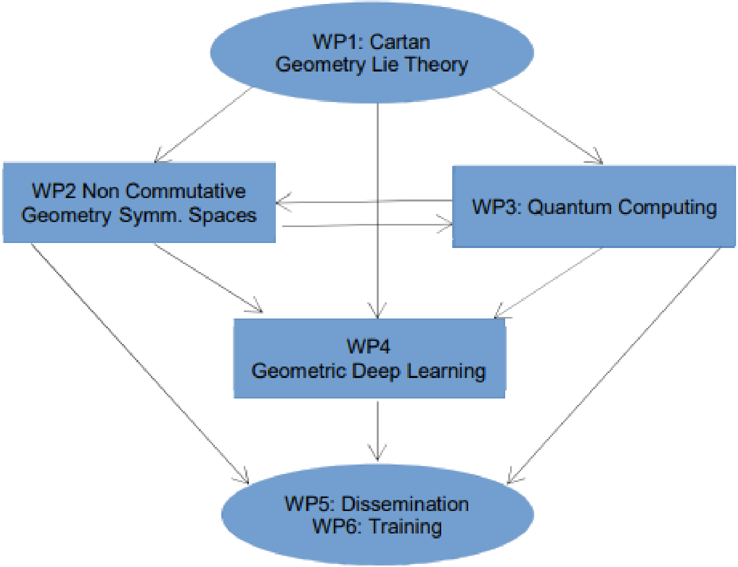
CaLiForNIA, Training School

Rita Fioresi, University of Bologna

January 27, 2025

CaLiForNIA Poster




WP1: Cartan
Geometry Lie Theory
WP2 Non Commutative
Geometry Symm. Spaces
WP3: Quantum Computing
WP4
Geometric Deep Learning
WP5: Dissemination
WP6: Training

1. **Lie, representation theory and Cartan Geometry. (WP Leader Prof. Slovak)**
   - **Task 1.1 Foundations of Cartan connections and representation theory.**
     DC1 Sprenger Georg (Neusser, Gover)
     Objectives: Investigate various aspects of rigid geometric structures using Cartan connections and methods from representation theory.
   - **Task 1.2 Geometric Control Theory.**
     DC2: Greenwood Steven (Slovak, Waldron)
     Objectives: Find new geometric techniques via Cartan geometry and tractor calculus for the geometric control theory problems, including singularities
   - **Task 1.3 Contact and SubRiemannian Geometry.**
     DC3: García Rivas Alejandro (Latini, Waldron)
     Objectives: Generalize, in terms of the curved orbit decomposition program, tractor calculus and the notion of defining densities for the study of conformal manifolds.
   - **Task 1.4 Palatini-Cartan Formalism.**
     DC11: Zaitseva Taisiia (Cattaneo, Latini)
     Objectives: Bring BV, BFV formalisms centerstage to develop a new approach to Palatini gravity

WP2: Non commutative geometry of symmetric spaces.
WP Leader Prof. R. O Buachalla.

- **Task 2.1 Quantum Flags and C*-Algebras.**
  DC4: Akhila Nelliyamkunnath Satheesen (Strung, Somberg)
  Objectives: extend known constructions of graph C*-algebra models for the C* algebras of quantum homogeneous spaces of Drinfeld–Jimbo quantum groups.
- **Task 2.2. Baum-Connes conjecture for quantum symmetric spaces.**
  DC5: Julius Benner (O Buachalla, Fioresi)
  Objectives: noncommutative geometry of the quantum flag manifolds from a geometric and algebraic point of view Noncommutative Kähler geometry of the Heckenberger–Kolb calculi of the irreducible quantum flag manifolds.
- **Task 2.3. Quantum Harmonic Superspace and Unitary Representations.**
  DC6: Giovanni Camilletti (Lledo, Fioresi)
  Objectives: Generalize the Mackey machine to supersymmetry (SUSY)

WP3: Quantum Computing and Quantum Information Geometry

- **DC7: Quantum Information Geometry.**
  Luca Ion (Perez-Canellas, Ercolessi)
  Objectives: Investigate the role of metrologic and associated metric concepts on the design and performance of noisy quantum algorithms. Research on systems of many interacting spins in quantum computing.
- **DC8: Quantum Algorithms**
  Marko Brnovic (Ercolessi, Perez-Canellas)
  Objectives: Exploit the geometry of quantum states to develop more efficient schemes for hybrid quantum-classical variational algorithms.

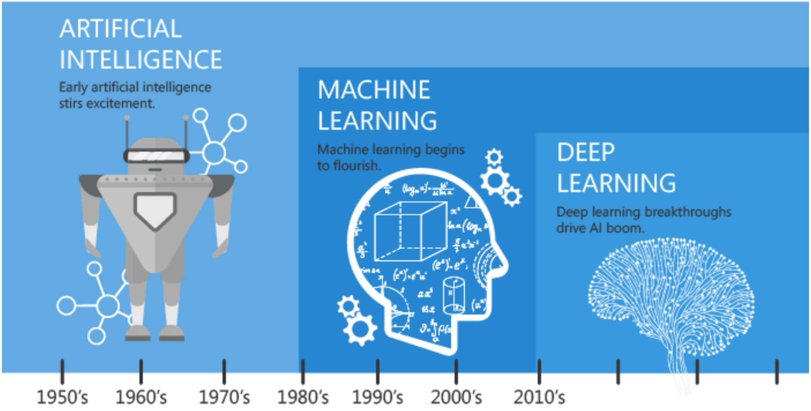
WP4: Geometric Deep Learning.

- **DC9: Foundations of Sheaf Neural Networks.**
  Jan-Willem Van Looy (Fioresi, Slovak)
  Objectives: Exploit sheaf theory and information geometry to understand parameter and data spaces in group equivariant graph neural network.
- **DC10: Geometric Deep Learning and Symmetric Spaces.**
  Olga Zaghen (Bekkers, Fioresi)
  Objectives:
  Understand group equivariant graph neural network.

	Main Training Events & Conferences	ECTS	Lead Institution	Action Month
1	**Training School: Cartan Geometry and Quantum Symmetric** Spaces [Slovak (MU), Buachalla (CU)]	5+5	CU	8, 24
2	**PhD course: Cartan Geometry, I, II** [Slovak, Neusser (MU)] *	5+5	MU	8, 18
3	**Quantum Groups I and II** PhD course [Ó Buachalla (CU)] *	5+5	CU	8, 18
4	**PhD course: Geometric Deep Learning I and II** [Fioresi (UNIBO)]*	5+5	UNIBO	12, 36
5	**Conference: Differential Geometry and its Applications** [Slovak (MU), Waldron (UC)]	5+5	MU	12, 36
6	**Training School: Geometric Deep Learning I, II** [Bekkers (UvA)]*	5+5	UvA	14, 38
7	**Conference: Quantum, Super Days in Bologna** [Latini, UNIBO]	2+2+2	UNIBO	16, 28, 40
8	**Midterm workshop to assess CaLiForNIA progress** *	2	CAS	24
9	**PhD course Quantum Computing** [Ercolessi (UNIBO) Perez-Canellas (UVEG)]	5	UNIBO	12
10	**Training School on Advanced Quantum Computing** [Perez-Canellas, UVEG, Ercolessi UNIBO] *	5	UVEG	24
11	**Escape room at Play fair** [Fioresi, UNIBO] *	1+1+1+1	UNIBO	12, 24
12	**Meme contest on Vision and Symmetry themes** [Cattabriga, UNIBO] *	1+1	UNIBO	12, 24
13	**Science in the garden** [Lledo, UVEG]	1+1	UVEG	12, 36
14	**Month of Science at the Library Delfini [Fioresi, UNIBO]**	1+1	UNIBO	18, 42
15	**Exhibition on Discrimination in Science: never again [Fioresi, UNIBO]** *	1	UNIBO	24
16	**Final Workshop to wrap up CaLiForNIA project***	2	CAS	46

1. Crash Course in Deep Learning

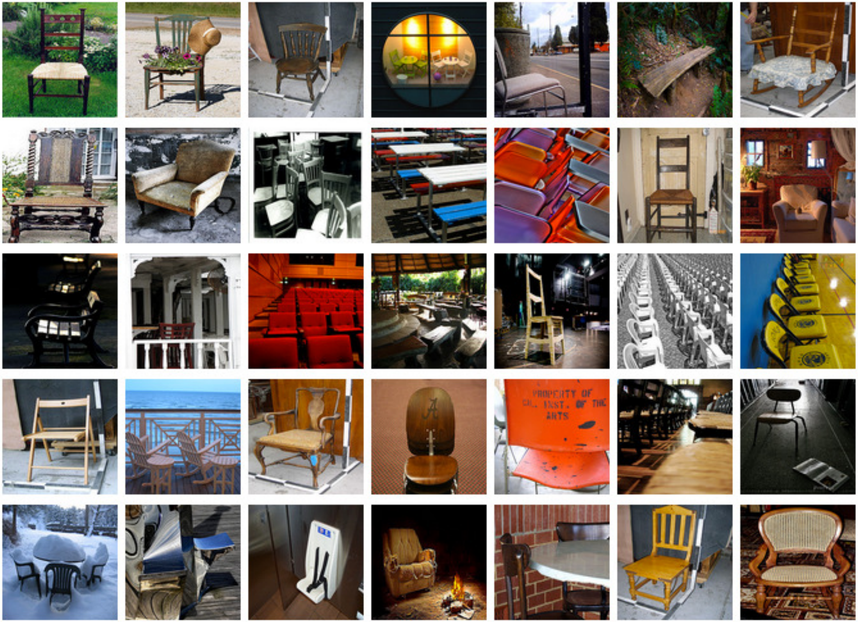
AI, Machine Learning and Deep Learning



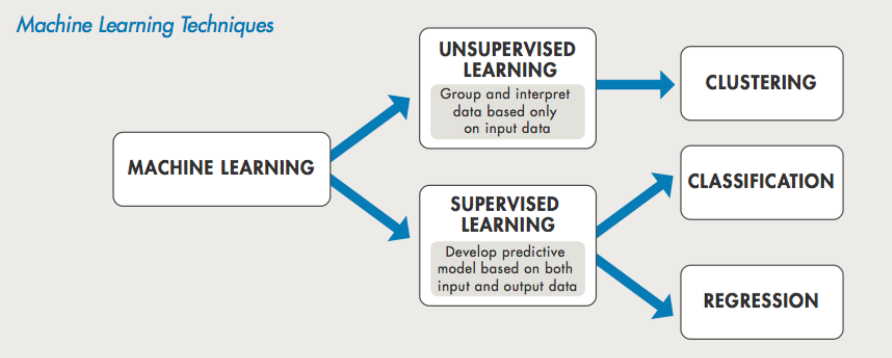
- **2010** 20000 immages, 20 categories: 25% error.
- **2011** 1 milion images, 1000 categories: 16% error.
- **2015** 1 milion di images, 1000 categories: 4% error.

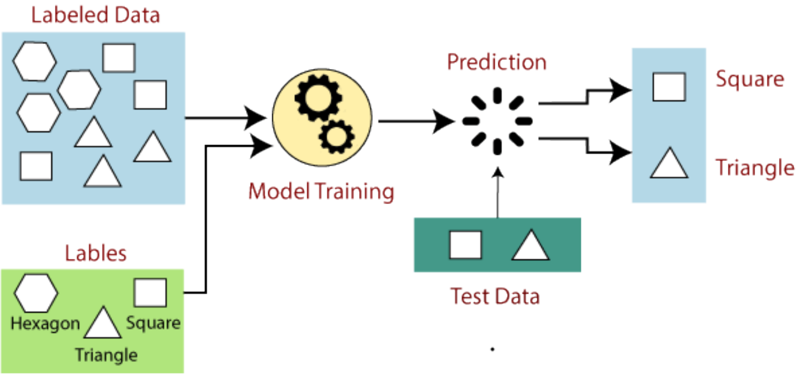
The Imagenet Challenge was declared won in 2017 by a Deep Learning algorithm.

Images in Imagenet category "chair"



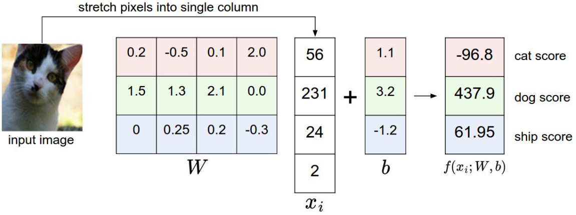
Supervised/Unsupervised Learning




Labeled Data
Lables
Hexagon
Square
Triangle
Model Training
Prediction
Test Data
Square
Triangle

Ingredients for Deep Learning

- **Score function**: it is a function of the weights w (es. linear classifier) It gives a *score* for a data x and weights w: e.g. $s(x, w) = \sum w_{ij} x_j$.



- **Loss function**: measures error
  (L_i datum i loss, y_i correct label)

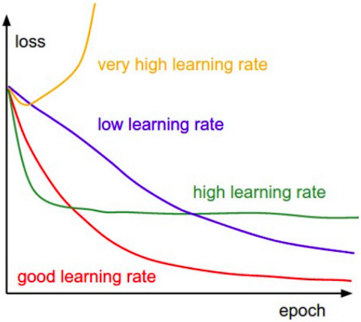
$$L_i = -log \frac{e^{f_{y_i}}}{\sum_j e^{f_j}} = -f_{y_i} + log \sum_j e^{f_j}, \qquad L = \sum_i L_i$$

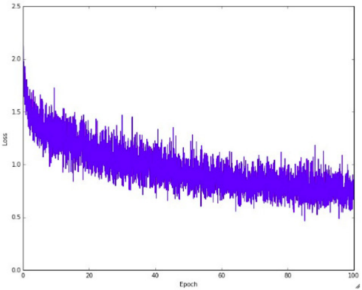
- **Optimizer**: for weights update "minimizes" the Loss

$$w_{ij}(t+1) = w_{ij}(t) - \alpha \nabla L_{\mathrm{stoc}}, \qquad \nabla L_{\mathrm{stoc}} = \sum_{i=1}^{32} \nabla L_{\mathrm{rand(i)}}$$

Divide the dataset (ex. CIFAR10):
80% Data for **training**
10% Data for **validation**
10% Data for **test** (ONCE)

1. **Learning**: determine weights **parameters**

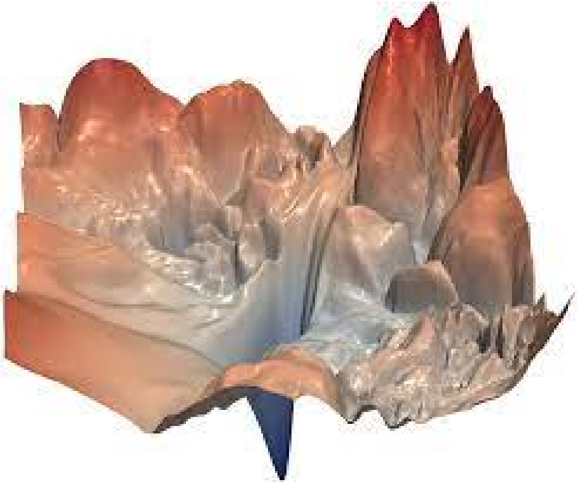




2. **Validation**: determine net structure.
   Example: choose loss function, number of layers, learning rate
   Goal: find best hyperparameters.
3. **Test**: once at the end.

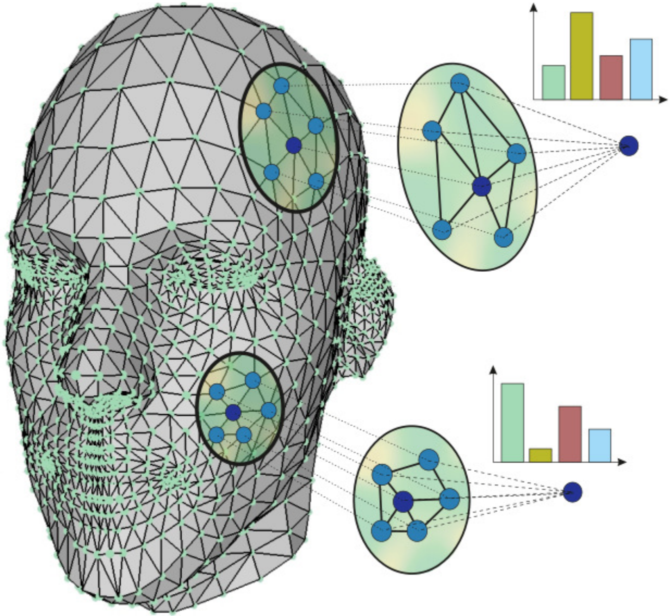
Accuracy: percentage of accurate predictions on tests set.

Loss Landscape

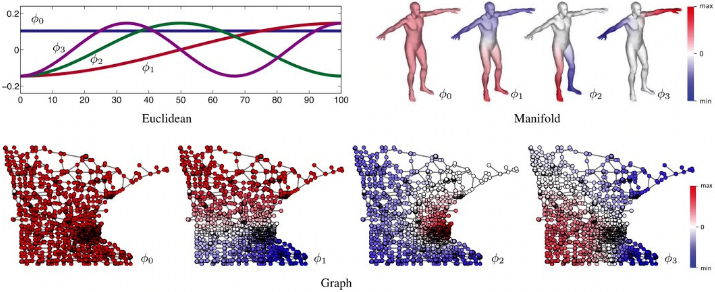


2. Crash Course in Geometric Deep Learning

Geometric Deep Learning: Graph Convolutional Networks



The eigenfunctions of the laplacian form the smoothest-possible basis function over a specific graph (they minimize the Dirichlet energy).

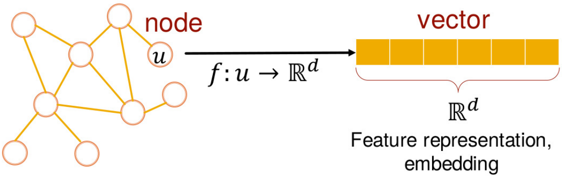


A GCN consists of the following steps:

- **Encoding:** realize a (low) dimensional embedding of the graph. Typically via a set of *learned* convolutional layers.
- **Decoding:** from the embedding we compute a **SCORE** Typically via a *learned* linear layer.
- **Loss function** (same idea as DL)
- **Optimizer** (same idea as DL)

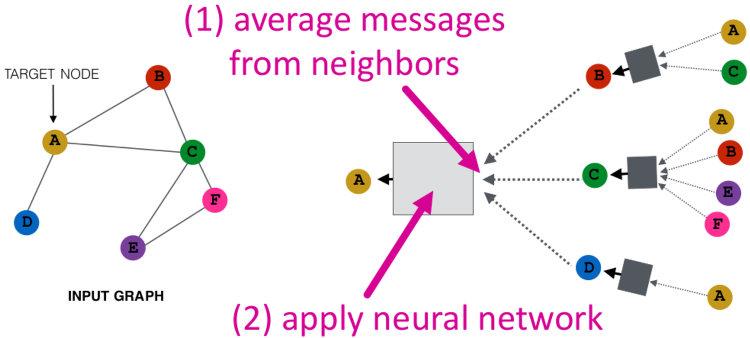
Once score, loss and optimizer are given, the training, validation and step take place in the same way as in Deep Learning algorithm.

Encoder-decoder framework:



3. Convolution on Graphs

The mechanism of **message passing** is an approximation of the mathematical operation of convolutions on graph (Spectral methods):



The operation of message passing in practice

Initial 0-th layer embeddings are equal to node features

$$h_v^0 = x_v$$

embedding of v at layer k

$$h_v^{(k+1)} = \sigma\Big(W_k \sum_{u \in N(v)} \frac{h_u^{(k)}}{|N(v)|} + B_k h_v^{(k)}\Big), \forall k \in \{0, \dots, K-1\}$$

Total number of layers

Average of neighbor's previous layer embeddings

$$z_v = h_v^{(K)}$$

Embedding after L layers of neighborhood aggregation

Non-linearity (e.g., ReLU)

Notice summation is a permutation invariant pooling/aggregation.

$h_v^{(k)}$: the hidden representation of node v at layer k
W_k: weight matrix for neighborhood aggregation (B_k: bias).
Important: Message passing and neighbor aggregation in graph convolution networks is **permutation equivariant**.

Laplacian on graphs

For ordinary geometry the Laplacian is the operator:

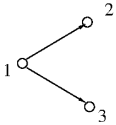
$$\Delta = \partial_1^2 + \cdots + \partial_n^2 = (\partial_1 + \cdots + \partial_n) \cdot (\partial_1 + \cdots + \partial_n) = \nabla^t \cdot \nabla$$

Definition. Let $G(V, E)$ be a directed graph.
$L = D - A$ is the Laplacian.
D degree matrix
A adjacency matrix

Proposition. $L = \nabla^t \cdot \nabla$ (obvious)
$\nabla f : E \longrightarrow \mathbb{R}$, $f(i,j) = f(j) - f(i)$ discrete gradient.

Example.

$$\nabla = \begin{pmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \quad \nabla^t \nabla = \begin{pmatrix} -1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$



Diffusion Laplacian

$$L_D := D^{-1}L = I - D^{-1}A$$

Heat equation

- Heat equation in its graph version (continuous time):

$$(\partial_t h)(v) = -(\Delta h)(v) = -h(v) + \sum_{u,(u,v)\,\text{edge}} \frac{h(u)}{d(u)}$$

- Heat equation in its time discrete version:

$$h_{t+1}(v) = h_t(v) - h_t(v) + \sum_{u,(u,v)\,\text{edge}} \frac{h_t(u)}{d(u)} = \sum_{u,(u,v)\,\text{edge}} \frac{h_t(u)}{d(u)}$$

Hence:

Message passing = heat equation for diffusion Laplacian

Definition. A First Order Differential Calculus (FODC) on an associative unital algebra A is a pair (Γ, d), where

i.) Γ is an A-bimodule.

ii.) $\mathrm{d}\colon A \to \Gamma$ is a k-linear map satisfying the Leibniz rule

$$\mathrm{d}(ab) = \mathrm{d}(a)b + a\mathrm{d}(b), \qquad a, b \in A$$

iii.) $A \otimes A \to \Gamma$, $a^i \otimes b^i \mapsto a^i \mathrm{d}(b^i)$ is a (left A-linear and) surjective map.

Example.

$$A = k[V] = \mathrm{span}\{\delta_x \mid x \in V\},$$

where $\partial_x(y) = 1$ if $x = y$ and zero otherwise. We define a FODC (Γ, d) on A:

$$\Gamma := kE = \mathrm{span}\{\omega_{x \to y} \,|\, (x, y) \in E\}$$

where the bimodule structure is given by:

$$f\omega_{x\to y} = f(x)\omega_{x\to y}, \quad \omega_{x\to y} f = \omega_{x\to y} f(y), \quad \mathrm{d}f = \sum_{x\to y \in E} (f(y) - f(x))\omega_{x\to y}, \qquad f \in k[V]$$

$$\mathrm{d}\delta_x = \sum_{y : y \to x} \omega_{y\to x} - \sum_{y : x \to y} \omega_{x \to y}, \quad \delta_x \mathrm{d}\delta_y = \begin{cases} -\sum_{z : x \to z} \omega_{x \to z} & x = y \\ \omega_{x \to y} & x \to y \\ 0 & \text{otherwise} \end{cases}$$

There is a one to one correspondence:

$$\text{FODC on } V \longleftrightarrow G = (V, E) \text{ directed graphs}$$

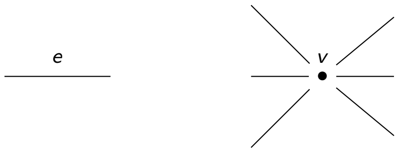
In this language we can define all the notions of Riemannian Geometry:

- Vector bundles and Differential operators
- Connections, Curvature and Cohomology

Most important we can define **sheaves** once we give a topology on G.

Base of open sets on G (Alexandroff topology):

- $U_e = \{e\}$, i.e. the edge e, without its vertices, for each $e \in E$.
- $U_v = \{e \in E \mid v \leq e\}$, that is the open star of v, for each vertex $v \in V$,



Definition. A **sheaf** on a directed graph $G = (E_G, V_G, h_G, t_G)$ is equivalent to a presheaf F on the base for the topology. This is the datum of:

- a vector space $F(v)$ for each vertex $v \in V_G$,
- a vector space $F(e)$ for each edge (with its endpoints) $e \in E_G$,
- linear maps (restriction maps) $F_{h_G(e) \leq e} : F(e) \to F(h_G(e))$, $F_{t_G(e) \leq e} : F(e) \to F(t_G(e))$ for each edge $e \in E_G$, where, we write $v \leq e$ to mean that v is a vertex of the edge e.

Notice: we have a natural notion of preorder on G, this leads to sheaves on preordered sets called **cellular sheaves**.

Definition. Let G be a directed graph. Let F be a sheaf on G of rank n. We define a *sheaf connection* as $\partial : C^0(G, F) := \oplus_{v \in V} F(U_v) \longrightarrow C^1(G, F) := \oplus_{e \in E} F(U_e)$ as

$$\partial(x)_e = F_{v \leq e} x_v - F_{u \leq e} x_u, \quad x_v \in F(U_v),\, x_u \in F(U_u),\; \partial(x)_e \in F(U_e)$$

Definition. Define the **sheaf Laplacian** $L_F : C^0(G, F) \to C^0(G, F)$ as

$$L_F := \partial^* \circ \partial$$

or, more explicitly (calculation):

$$L_F(x)_v = \sum_{u,v \leq e} F^*_{v \leq e}(F_{v \leq e} x_v - F_{u \leq e} x_u)$$

where:

$$\partial(x)_e = F_{v \leq e} x_v - F_{u \leq e} x_u, \quad x_v \in F(U_v),\ x_u \in F(U_u),\ \partial(x)_e \in F(U_e)$$

Note: if all the vector spaces involved in the definition of F are one dimensional, one recovers the usual graph Laplacian.

Conclusion and further work

- Go towards the notions of discrete manifolds, vector bundles, schemes, etc., you may want to apply to it appropriate networks invariant with respect to some geometric structure and you can use our framework to identify the "correct" constraints your network should satisfy (discrete symmetric spaces).
- Sheaf neural networks "learns the sheaf". A geometric bias on the data can constrain the learning (cocycle conditions/form of transition morphisms to learn).
- Different Grothendieck topologies (aka different notion of covering) give rise to different type of sheaves. We have also a noncommutative side of the story.

References

- Bronstein et al. Geometric deep learning: going beyond Euclidean data, https://arxiv.org/abs/1611.08097
- Kipf, Thomas N; Welling, Max (2016) Semi-supervised classification with graph convolutional networks". International Conference on Learning Representations. 5 (1): 61–80. arXiv:1609.02907
- Stanford course CS224w by Leskovec: http://cs224w.stanford.edu/
- Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, Yoshua Bengio Graph Attention Networks, 2018 https://arxiv.org/abs/1710.10903
- G. Godsil, G. Royce, Algebraic Graph Theory, GTM, Springer, 2001.
- E. Beggs, S. Majid, Quantum Riemannian Geometry, Springer 2020.
- R. Fioresi, F. Zanchetta, Deep Learning and Geometric Deep Learning, https://arxiv.org/abs/2305.05601
- R. Fioresi, A. Simonetti, F. Zanchetta, "Foundations of sheaf neural networks", preprint, 2025.

Join CaLISTA CA 21109!

https://site.unibo.it/calista/en

Action Chair: Rita Fioresi

- **Working group 1: Cap U. Vienna, Slovak U. Brno**
  Cartan Geometry and Representation Theory
- **Working group 2: Abenda U. Bologna, Tanzini Sissa**
  Integrable Systems and Supersymmetry
- **Working group 3: O Buachalla Charles U., Aschieri Uniupo**
  Noncommutative Geometry and Quantum Homogeneous Spaces
- **Working group 4: Angulo Mines Paris, Parton U. Pescara**
  Vision and Machine Learning
- **Working group 5: Lledo U. Valencia, Tekel U. Bratislava**
  Dissemination and Public Engagement

https://e-services.cost.eu/action/CA21109/working-groups/applications