AIDD Spring School – Lugano – 17/05/2022

Equivariant (G)NNs

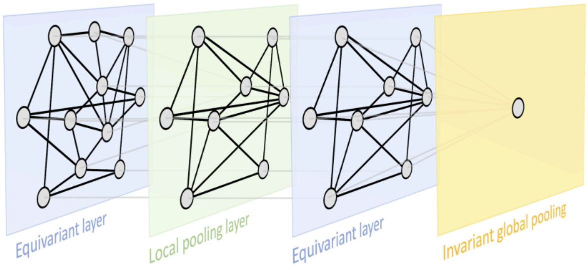
Marco Bertolini (Bayer AG)

Outline

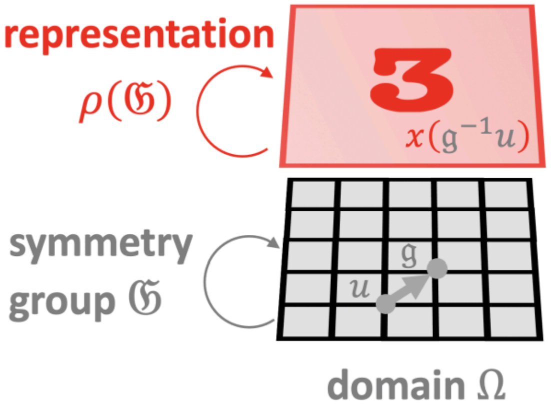
- Symmetry in data and models
- Basic concepts:
  - Groups and representations
  - Equivariance
- How to incorporate equivariance in neural networks
- Applications:
  - Unsupervised invariant representation learning
  - Electron density prediction
  - Hamiltonian matrix prediction

Symmetry in data

- ➢ Symmetry is often part of real world data
- ➢ Incorporating symmetries as inductive bias in neural networks has several advantages
  - Improve performance (NN can focus on actual problem)
  - Less data needed for training (no need of augmentation)
  - Common examples: CNNs, MPNNs



- ➢ Often: label/signal is group invariant but representation is not
  - Ordinary neural networks do not know what coordinates are.
    By default, coordinates are just numbers.



Bronstein et al., "Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges" https://arxiv.org/abs/2104.13478

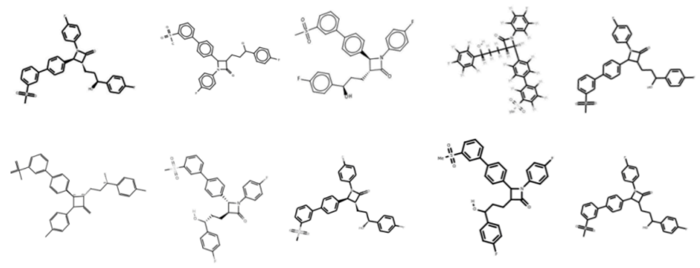
Symmetry-aware models

There are three main approaches to making the model “understand” the symmetry of the data

1. Data Augmentation

We provide to the model several instances of the same underlying signal

- ✗ Data inefficient
- ✗ Parameter inefficient
- ✗ Preprocessing necessary

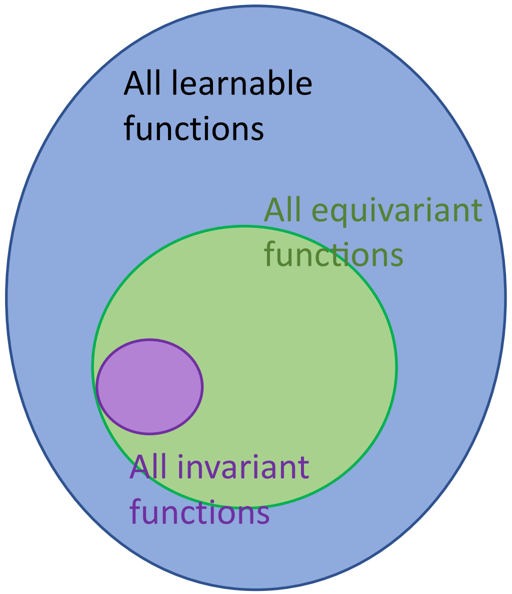


Different molecular depictions of the same structure

2. Invariant Models

We restrict to invariant models or invariant features

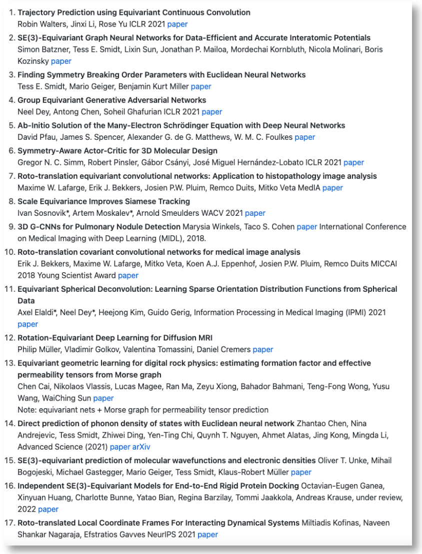
- ✓ Data efficient
- ✓ Parameter efficient
- ✗ Restrictive functions/features
- ✗ Often preprosessing necessary



3. Equivariant Models

Models are symmetry-aware but not restricted to be invariant.

- ✓ Data efficient
- ✓ Parameter efficient
- ✓ End-to-end learning easier
- ✓ Powerful function/features
- ✓ Can extract invariance (if needed)
- ✓ Very active (and successful) area of research



1. **Trajectory Prediction using Equivariant Continuous Convolution**
   Robin Walters, Jinxi Li, Rose Yu ICLR 2021 paper
2. **SE(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials**
   Simon Batzner, Tess E. Smidt, Lixin Sun, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Boris Kozinsky paper
3. **Finding Symmetry Breaking Order Parameters with Euclidean Neural Networks**
   Tess E. Smidt, Mario Geiger, Benjamin Kurt Miller paper
4. **Group Equivariant Generative Adversarial Networks**
   Neel Dey, Antong Chen, Soheil Ghafurian ICLR 2021 paper
5. **Ab-Initio Solution of the Many-Electron Schrödinger Equation with Deep Neural Networks**
   David Pfau, James S. Spencer, Alexander G. de G. Matthews, W. M. C. Foulkes paper
6. **Symmetry-Aware Actor-Critic for 3D Molecular Design**
   Gregor N. C. Simm, Robert Pinsler, Gábor Csányi, José Miguel Hernández-Lobato ICLR 2021 paper
7. **Roto-translation equivariant convolutional networks: Application to histopathology image analysis**
   Maxime W. Lafarge, Erik J. Bekkers, Josien P.W. Pluim, Remco Duits, Mitko Veta MedIA paper
8. **Scale Equivariance Improves Siamese Tracking**
   Ivan Sosnovik*, Artem Moskalev*, Arnold Smeulders WACV 2021 paper
9. **3D G-CNNs for Pulmonary Nodule Detection** Marysia Winkels, Taco S. Cohen paper International Conference on Medical Imaging with Deep Learning (MIDL), 2018.
10. **Roto-translation covariant convolutional networks for medical image analysis**
    Erik J. Bekkers, Maxime W. Lafarge, Mitko Veta, Koen A.J. Eppenhof, Josien P.W. Pluim, Remco Duits MICCAI 2018 Young Scientist Award paper
11. **Equivariant Spherical Deconvolution: Learning Sparse Orientation Distribution Functions from Spherical Data**
    Axel Elaldi*, Neel Dey*, Heejong Kim, Guido Gerig, Information Processing in Medical Imaging (IPMI) 2021 paper
12. **Rotation-Equivariant Deep Learning for Diffusion MRI**
    Philip Müller, Vladimir Golkov, Valentina Tomassini, Daniel Cremers paper
13. **Equivariant geometric learning for digital rock physics: estimating formation factor and effective permeability tensors from Morse graph**
    Chen Cai, Nikolaos Vlassis, Lucas Magee, Ran Ma, Zeyu Xiong, Bahador Bahmani, Teng-Fong Wong, Yusu Wang, WaiChing Sun paper
    Note: equivariant nets + Morse graph for permeability tensor prediction
14. **Direct prediction of phonon density of states with Euclidean neural network** Zhantao Chen, Nina Andrejevic, Tess Smidt, Zhiwei Ding, Yen-Ting Chi, Quynh T. Nguyen, Ahmet Alatas, Jing Kong, Mingda Li, Advanced Science (2021) paper arXiv
15. **SE(3)-equivariant prediction of molecular wavefunctions and electronic densities** Oliver T. Unke, Mihail Bogojeski, Michael Gastegger, Mario Geiger, Tess Smidt, Klaus-Robert Müller paper
16. **Independent SE(3)-Equivariant Models for End-to-End Rigid Protein Docking** Octavian-Eugen Ganea, Xinyuan Huang, Charlotte Bunne, Yatao Bian, Regina Barzilay, Tommi Jaakkola, Andreas Krause, under review, 2022 paper
17. **Roto-translated Local Coordinate Frames For Interacting Dynamical Systems** Miltiadis Kofinas, Naveen Shankar Nagaraja, Efstratios Gavves NeurIPS 2021 paper

REVIEW OF BASIC CONCEPTS

What is a group?

Definition

A group G is a **set** equipped with an **operation** $\cdot$, such that
1. (closure) $a \cdot b \in G$ for all $a, b \in G$;
2. (associativity) $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in G$;
3. (identity element) There exists a unique element e such that $\forall a \in G, a \cdot e = e \cdot a = a$;
4. (inverse element) $\forall a \in G$, there exists an element $a^{-1} \in G$ such that $a \cdot a^{-1} = a^{-1} \cdot a = e$.

Fundamental concept in mathematics to describe:

- Number systems
  - Integers $\mathbb{Z}$ $(0, \pm 1, \pm 2, \dots)$ with addition $(+)$
  - Real numbers $\mathbb{R}$ with addition
  - Real numbers $\mathbb{R}$ with multiplication ?? $\rightarrow \mathbb{R}_* = \mathbb{R}/\{0\}$

- Symmetries of an object/space
  - $SE(3)$: isometries of 3D Euclidean space

- Geometric transformations
  - Rotations in $\mathbb{R}^N$: $SO(N)$ = group of $n \times n$ orthogonal matrices $(O^T = O^{-1})$
  - Translations in $\mathbb{R}^N$: $T(N) = \mathbb{R}^N$

Representations

Groups are abstract objects, we are interested on their action on algrebraic/geometric spaces. A fundamental case is given by vector spaces, which generalize the concept of Eucliden spaces.

Definition (vector space)

A vector space over a field (of scalars) F is a set (of vectors) V equipped with two operations (vector addition and scalar multiplication) satisfying various axioms (additive, multiplicative, distributive axioms).

The bridge between the abstract group and its action on a vector space is given provided by the concept of representations:

Definition (representation)

A representation of a group G on a vector space V is a group homomorphism $\rho: G \to GL(W)$, that assigns to each $g \in G$ a linear ***map*** $\rho(g): V \to V$ such that

$$\rho(g_1 \cdot g_2) = \rho(g_1) \circ \rho(g_2)$$

Homomorphism = structure-preserving map

Examples of representations

Let us consider two triplets of real numbers:

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \in \mathbb{R}^3 \qquad \mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} \in \mathbb{R}^3$$

(e.g., energy, volume, …) (e.g., 3D coordinates, forces, velocities, …)

Both are a 3-tuple of real numbers. But they transform differently under, e.g., 3D rotations ($G = SO(3)$).
For example, under rotation $g \in SO(3)$ around x axis

$$\text{“}ga\text{”} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\rho_V(g)} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \qquad \text{“}g\mathbf{r}\text{”} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}}_{\rho_W(g)} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

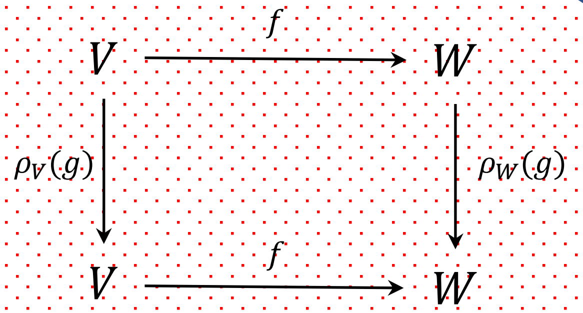
$\rho_V(g)$ ⟵ different representations ⟶ $\rho_W(g)$

Representations tell us:
- ➢ How to interpret the data with respect to G
- ➢ How the data transforms with respect to G

Equivariance

Definition (equivariance)

A map $f: V \to W$ is said to be G-equivariant with respect to the actions (representations) ρ_V, ρ_W if the diagram commutes for every $g \in G$.

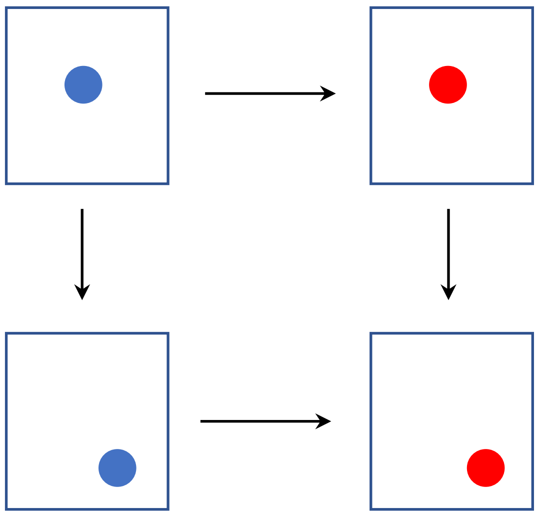


$$f(\rho_V(g)x) = \rho_W(g)f(x)$$

The function commutes with the group action

Examples

f =change from blue to red
G =translations in the plane



Let $\rho_V = \rho_W =$trivial representation,
i.e., $\rho_V(g)x = x,\ \forall x \in V,\ \forall g \in G$

$$f(\rho_V(g)x) = f(x)$$

$$\parallel$$

$$\rho_W(g)f(x) = f(x)$$

Invariance is a special case of equivariance!

f =translation by t
$G = \mathrm{SO}(3)$

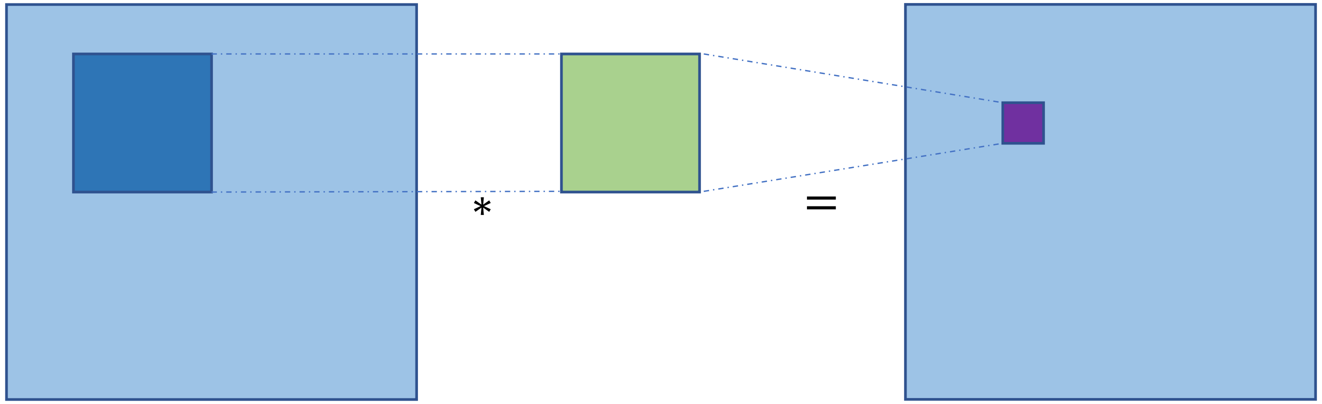
$$f(\rho_V(g)x) = f(Rx) = Rx + t$$

$$\nparallel$$

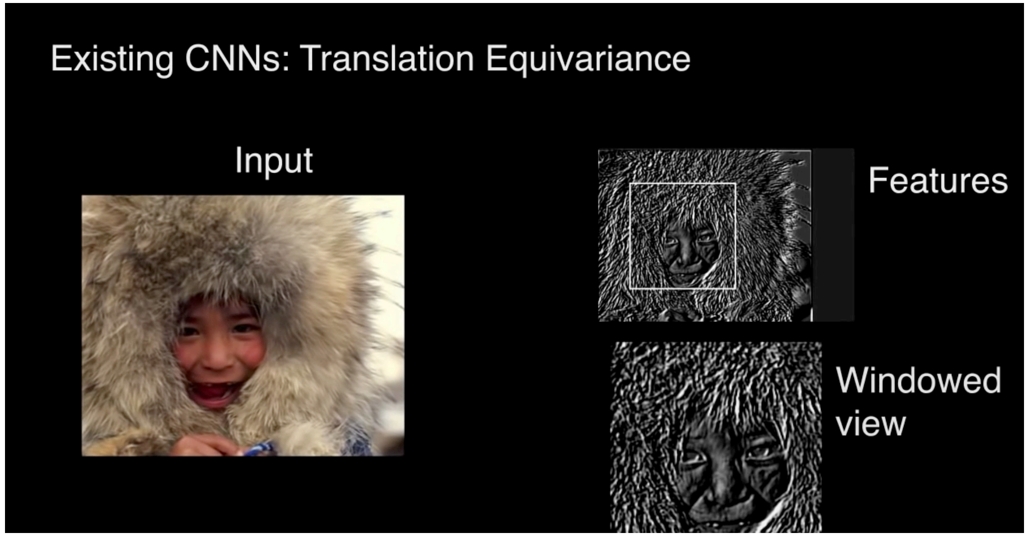
$$\rho_V(g)f(x) = \rho_V(g)(x+t) = Rx + Rt$$

Example 1: CNNs and translation equivariance

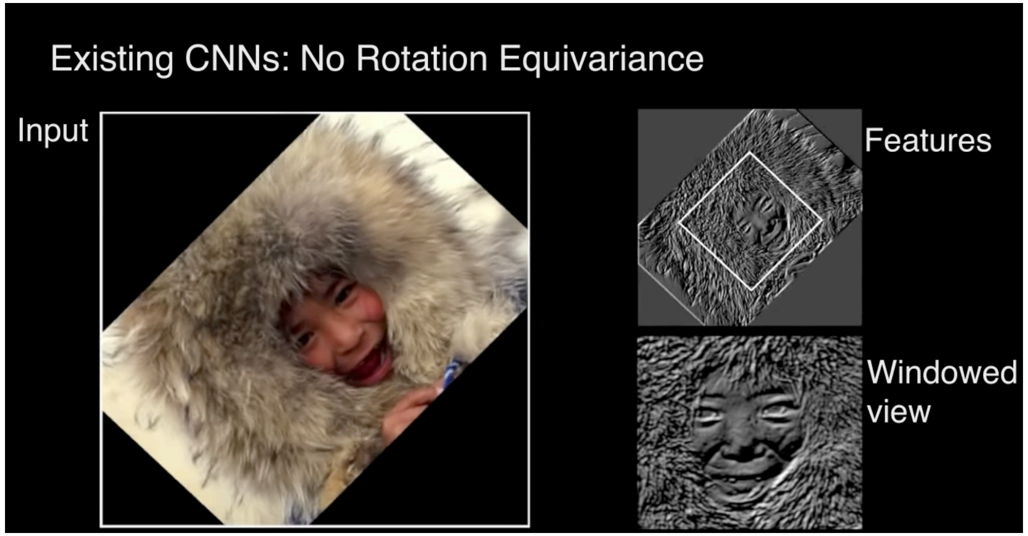
The success of CNNs strongly relies on their filter being equivatiant to translations.



We can check equivariance wrt translations



... and wrt rotations

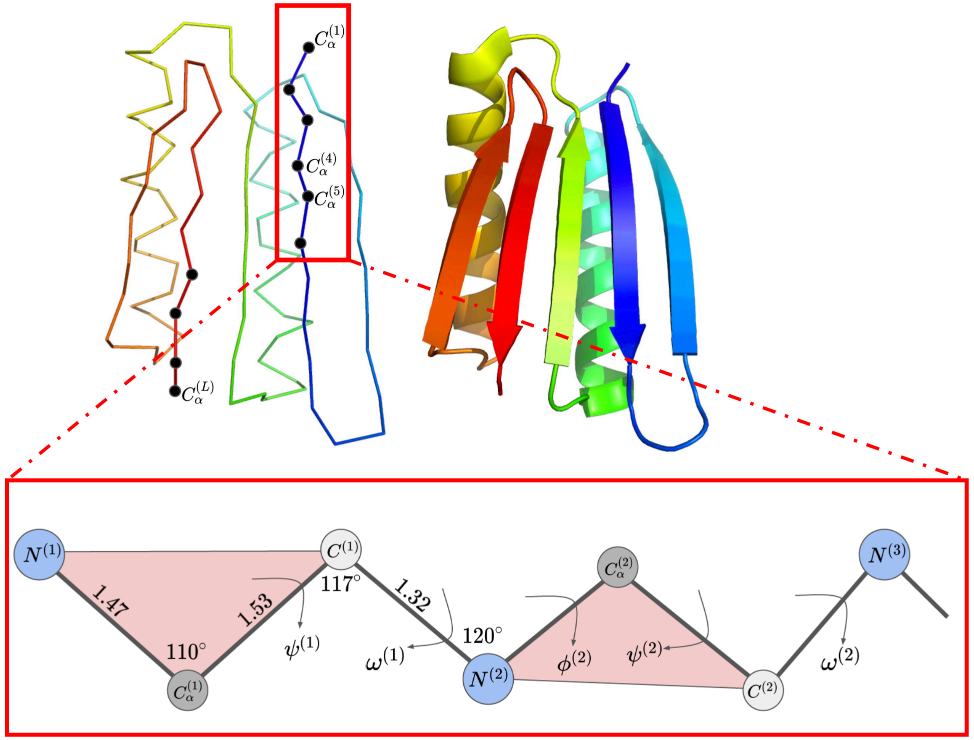


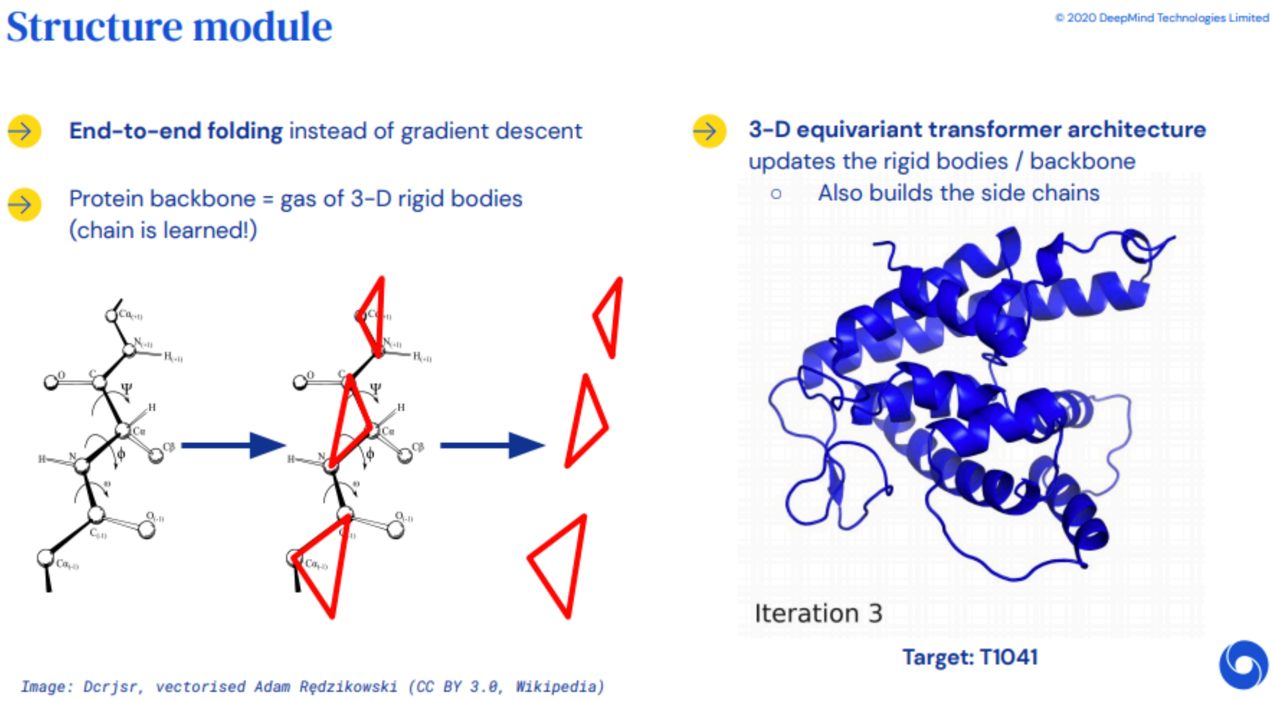
Harmonic Networks: Deep Translation and Rotation Equivariance (https://www.youtube.com/watch?v=qoWAFBYOtoU)

Example 2: AlphaFold2

CASP challenge: Amino Acid Sequence → 3D Folded Structure.

`The Structure module` uses a 3D equivariant transformer architecture to refine backbone coordinates and predict side chains.



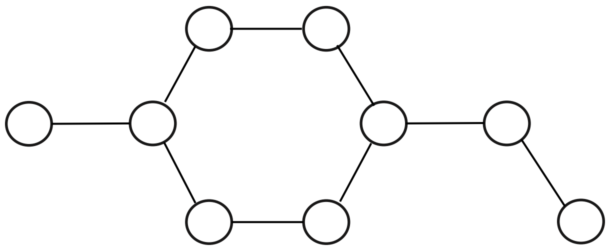


https://fabianfuchsml.github.io/alphafold2/

Beyond vectors

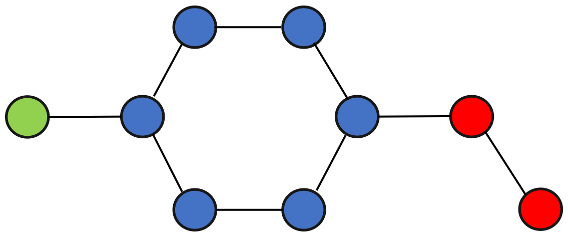
Topological Graph

Nodes and Edges.



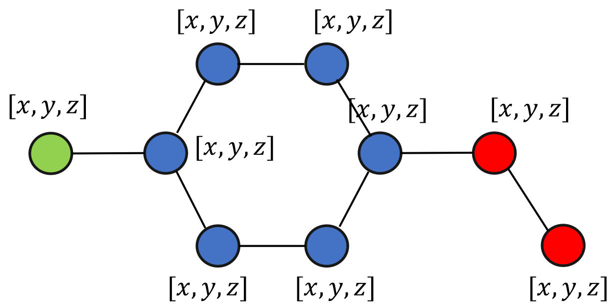
Topological Graph with features

Nodes (Edges) have features
(atom types, double/single bond, …)



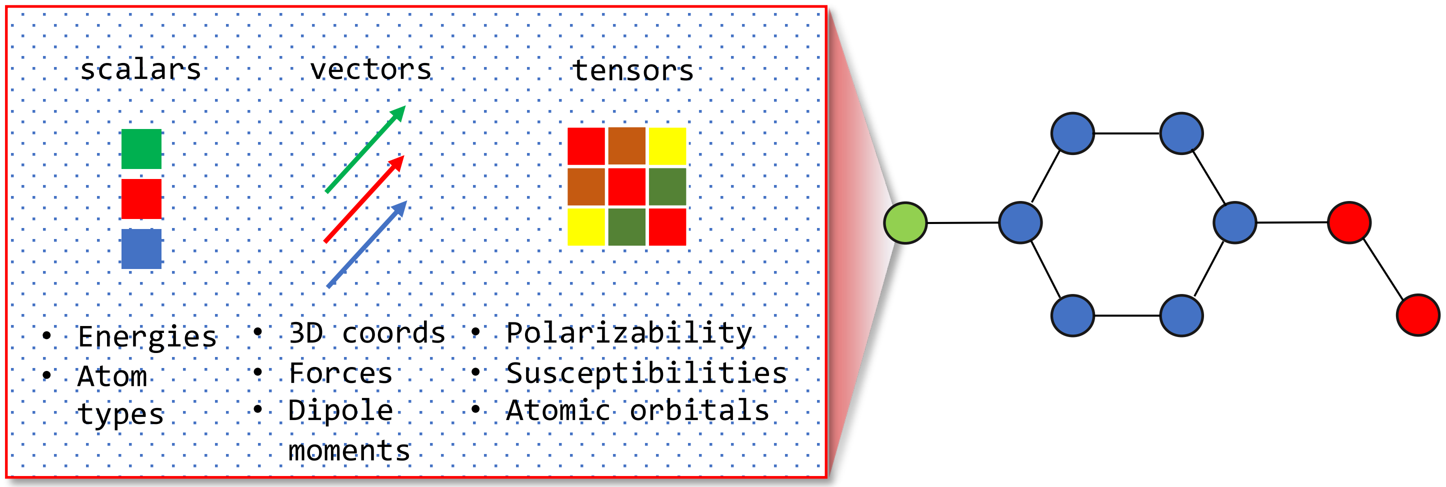
3D Graph with features

Nodes also have 3D coordinates



3D Graph with tensor features

Nodes also have geometric features

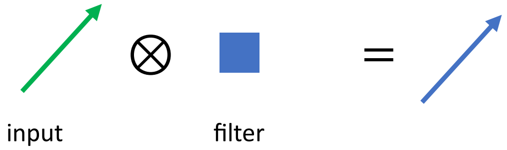


HOW TO INCORPORATE EQUIVARIANCE IN NNs

Approaches to equivariance

The various components of the network needs to be symmetry-aware as well

- Simplest choice: invariant (scalar) filters



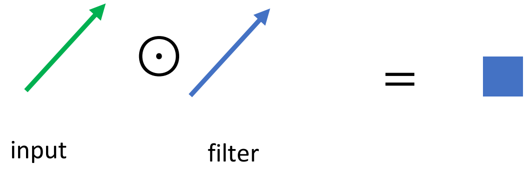
Example

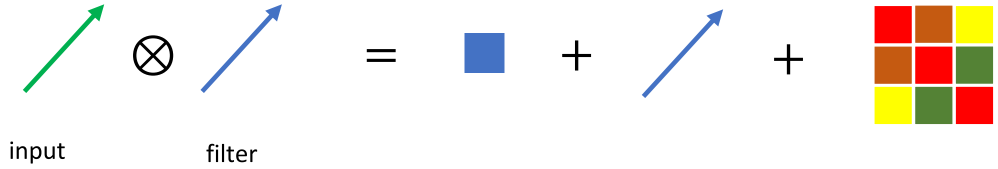
$$\vec{r} \otimes a = a\vec{r}$$

$$SO(3)\colon R(\vec{r} \otimes a) = R(a\vec{r}) = aR(\vec{r}) = R(\vec{r}) \otimes a$$

$$T(3)\colon \mathrm{T}(\vec{r} \otimes a) = (\vec{r} \otimes a) + t = a\vec{r} + t \neq a(\vec{r} + t) = T(\vec{r}) \otimes a$$

- Filters can carry non-trivial representations






```
In[248]: from e3nn import o3
    ...: rot_x = o3.matrix_x(torch.tensor(3.14 / 3.0)).squeeze()
    ...: rot_x
Out[248]:
tensor([[ 1.0000,  0.0000,  0.0000],
        [ 0.0000,  0.5005, -0.8658],
        [ 0.0000,  0.8658,  0.5005]])
In[249]: v1 = torch.tensor([[1.0,2.0,3.0]])
    ...: v2 = torch.tensor([[1.0,5.0,1.0]])
In[250]: v1_rot = v1 @ rot_x.T
    ...: v2_rot = v2 @ rot_x.T
In[251]: prod = torch.kron(v1, v2.T)
    ...: prod
Out[251]:
tensor([[ 1.,  2.,  3.],
        [ 5., 10., 15.],
        [ 1.,  2.,  3.]])
In[253]: prod_rot = torch.kron(v1_rot, v2_rot.T)
    ...: prod_rot
Out[253]:
tensor([[ 1.0000, -1.5964,  3.2329],
        [ 1.6365, -2.6125,  5.2908],
        [ 4.8293, -7.7092, 15.6125]])
In[254]: torch.trace(prod)
Out[254]: tensor(14.)
In[255]: torch.trace(prod_rot)
Out[255]: tensor(14.0000)
```

```
In[263]: def extract_vector(matrix):
    ...:     return torch.tensor(
    ...:         [matrix[1,2] - matrix[2,1],
    ...:          matrix[2,0] - matrix[0,2],
    ...:          matrix[0,1] - matrix[1,0]]
    ...:     )
    ...:
In[264]: extract_vector(prod)
Out[264]: tensor([13., -2., -3.])
In[265]: extract_vector(prod_rot)
Out[265]: tensor([13.0000,  1.5964, -3.2329])
In[266]: extract_vector(prod) @ rot_x.T
Out[266]: tensor([13.0000,  1.5964, -3.2329])
```

Approaches to equivariance - I

The are two main approaches to design an G-equivariant network **(focus on SO(3)/SE(3)).**

1. Design ad-hoc operations

$SO(3)$ –Equivariant NNs can use:

- Any (non-linear) function of scalars: $f(s)$
- Scaling of vectors: $s \circ \vec{v}$
- Linear combination of vectors: $W\vec{v}$
- Scalar products: $\langle \vec{v}_1, \vec{v}_2 \rangle$
- Vector products: $\vec{v}_1 \times \vec{v}_2$

} Filters, Transformations, Non-linearities, Message-passing, etc…

Convolution operation is translation equivariant:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau$$

(exercise: show it!)

DRAWBACKS:

- Extension to higher-order tensor very non-trivial
- Capacity limited by design (lack of generalization)

BENEFITS:

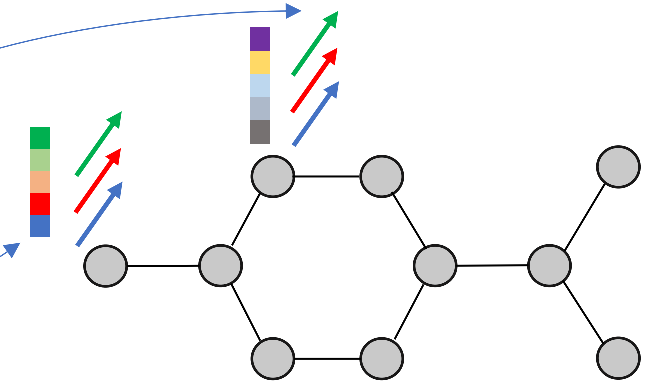
- "Low" computational cost
- "Easy" to implement for most groups
- Conceptually simple

Equivariant Graph NN for Molecular Property Prediction

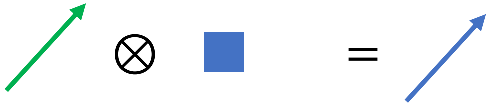
Consider a graph with both scalar and vector features
with respect to $G = SO(3)$

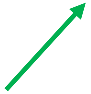
$$x_i = (s_i, \vec{v_i}) \in \mathbb{R}^{F_s} \times \mathbb{R}^{3xF_v}$$

scalar   vector



Even if no initial vector features, equivariant (vector) interactions are created via tensor product of relative positions with an invariant representation





$$= \vec{p}_{ij} = \frac{p_j - p_i}{\left\| p_j - p_i \right\|_2}$$

enables interaction
between type-1 features

Different strategies for $SE(3)$ −invariant embeddings:

Schütt, Kristof T. et al. "SchNet: A continuous-filter convolutional neural network for modeling quantum interactions." *NIPS* (2017).
Schütt, Kristof T .et al. "Equivariant message passing for the prediction of tensorial properties and molecular spectra." ICML (2021).
Le, Tuan et al. "Equivariant Graph Attention Networks for Molecular Property Prediction." ArXiv abs/2202.09891 (2022).

Approaches to equivariance- II

The are two main approaches to design an G-equivariant network

2. Design general symmetry-aware operations

Definition ((irr)reducible representation)

V is reducible if there exists $W \subseteq V$ such that W is a representation.
If V is not reducible it is said to be **irreducible.**

$$\rho_V(g)a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

$V = \mathbf{1}^{\oplus 3}$, $\mathbf{1}$ =trivial representation

$$\rho_W(g)\mathbf{r} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

W = **3** irreducible

Tensor product and complete reducibility

Definition (tensor product)

Let V_1, V_2 be two representation of the group G. Their tensor product is the tensor product of vector spaces $V_1 \otimes V_2$ with the linear action determined by

$$\rho_{V_1 \otimes V_2}(g)(v_1 \otimes v_2) = \rho_{V_1}(g) v_1 \otimes \rho_{V_2}(g) v_2$$

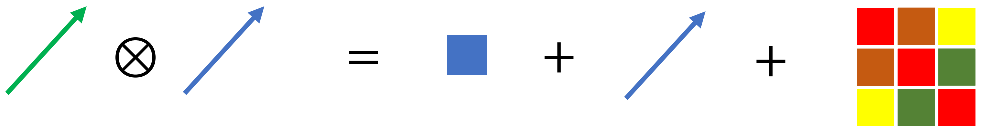
Proposition (complete reducibility)

Any representation is a direct sum of irreducible representations.

EX. $SO(3)$

$$\left(\mathbf{x}^{(l_1)} \otimes \mathbf{y}^{(l_2)}\right)^{l_3}_{m_3} = \sum_{m_1=-l_1}^{l_1} \sum_{m_2=-l_2}^{l_2} C^{l_3,l_2,l_1}_{m_3,m_2,m_1} x^{l_1}_{m_1} y^{l_2}_{m_2}$$

Clebsch-Gordan coefficients



Tensor-based NNs

Filters are constructed as a sum (collection) of different irreps

$$F = V_0^{a_1} \oplus V_1^{a_2} \oplus V_2^{a_3} \oplus \cdots$$

Features propagation is done with tensor product

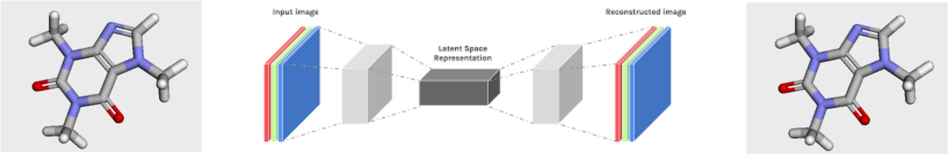
$$X^{(t)} \otimes F = X_0^{(t+1)} \oplus X_1^{(t+1)} \oplus X_2^{(t+1)} \oplus \cdots$$

- Extension to higher-order tensor "trivial"
- Higher computational cost
- Non-trivial to implement for most groups
- But frameworks available ($SE(3)$)
  - Tensor Field Networks (arXiv:1802.08219)
  - Clebsch-Gordan Nets (arXiv:1806.09231)
  - 3D Steerable CNNs (arXiv:1807.02547)
  - Cormorant (arXiv:1906.04015)
  - SE(3)-Transformers (arXiv:2006.10503)
  - e3nn (github.com/e3nn/e3nn)

APPLICATIONS

Unsupervised invariant representation learning

- In representation learning we wish to learn “the best possible” lower dimensional representation of the data
- Such representation can be used as powerful descriptors for downstream tasks/clustering/generative approaches, …
- Autoencoders are a powerful approach to unsupervised representation learning

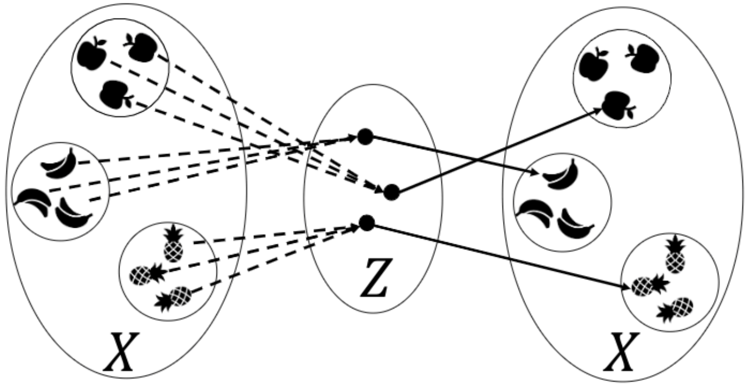


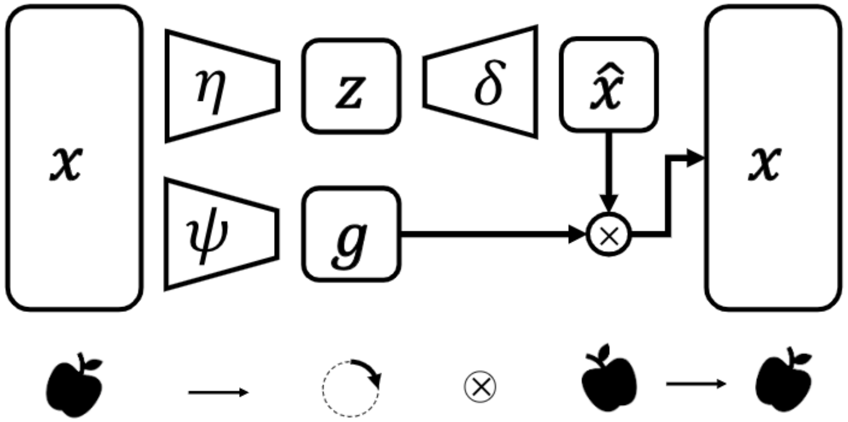
- Can we learn invariant representations with AE (from non-invariant data)?
- Problem is that, when data representation is not group invariant, the reconstruction loss will also not be!

$$\mathcal{L}_{\mathrm{rec}} = \mathcal{L}(x, \hat{x}) \neq \mathcal{L}(\rho(g)x, \hat{x})$$

- Node assignement problem: which permutation?
- Coordinate frame problem: which orientation?

Reconstruction through group action





- We want to learn the space Z as a bottleneck between **orbits** in X

$$O_x = \{\rho_X(g)x | \forall g \in G\}$$

- The decoder, at best, can map the embedding of the point x to a given element in its orbit

$$\delta(\eta(x)) = \hat{x} = \rho_X(\hat{g}_x)x$$

- We learn an additional **equivariant** map ψ, representing the group element mapping $\hat{x} \to x$

$$\mathcal{L}_{\text{rec}}(x) = \mathcal{L}(x, \rho(\psi(x))\hat{x})$$

- The loss is now G-invariant

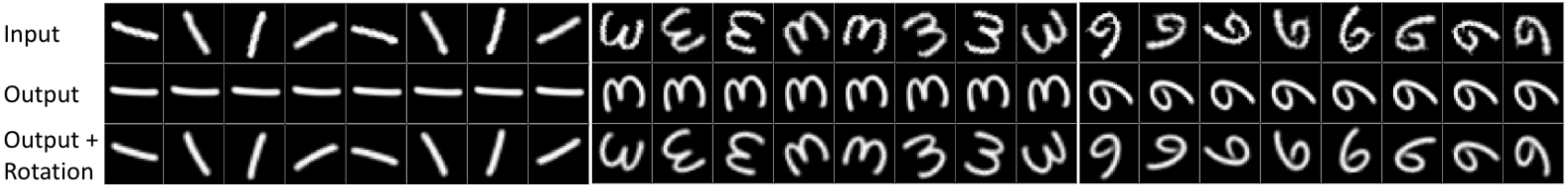
$$\begin{aligned}\mathcal{L}_{\text{rec}}(\rho(g)x) &= \mathcal{L}(\rho(g)x, \psi(\rho(g)x)\hat{x}) \\ &= \mathcal{L}(\rho(g)x, \rho(g)\psi(x)\hat{x}) = \mathcal{L}_{\text{rec}}(x)\end{aligned}$$

- We prove that such approach can be applied to any group G, both continuous and discrete
- We provide an explicit construction for any group G.

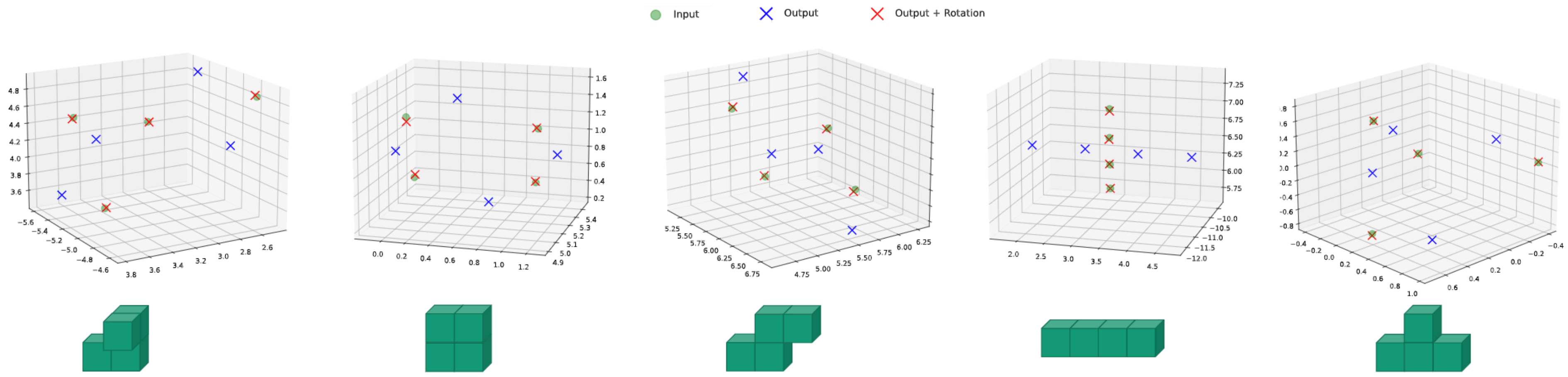
Winter, Robin et al. "Unsupervised Learning of Group Invariant and Equivariant Representations." ArXiv abs/2202.07559 (2022).

Examples

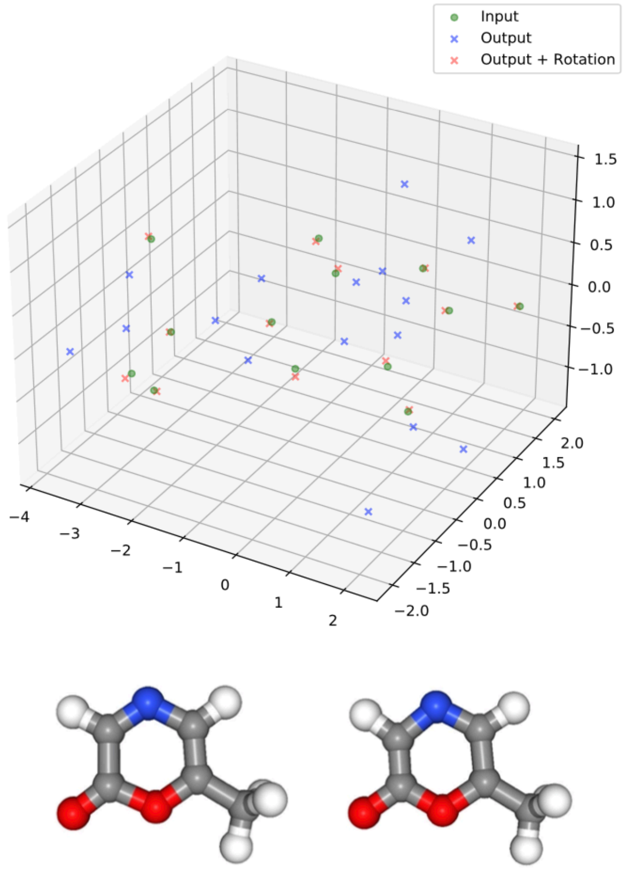
1. MNIST – G = SO(2)

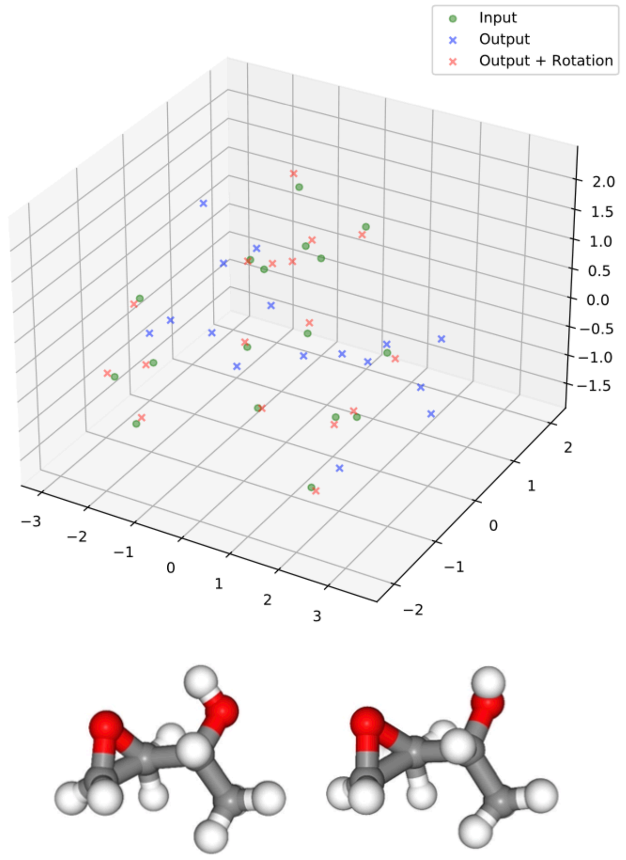


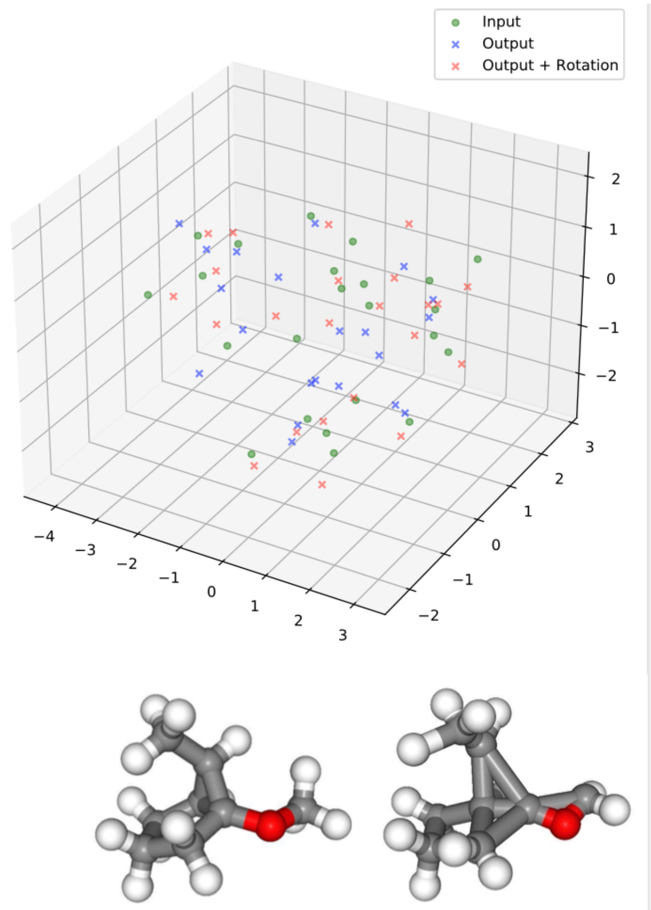
2. Point Cloud – G = SE(3) + S_n (Rotations + Translations + Permutations)



Visualization of Reconstructions: QM9





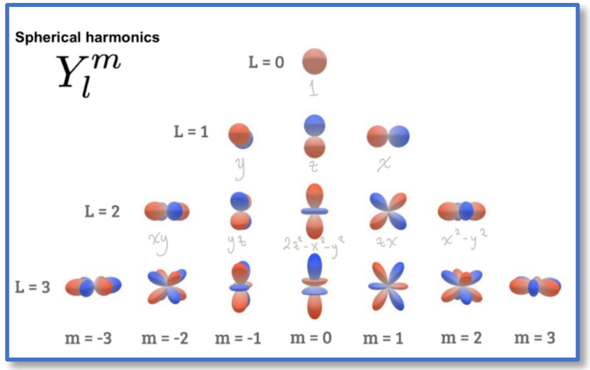


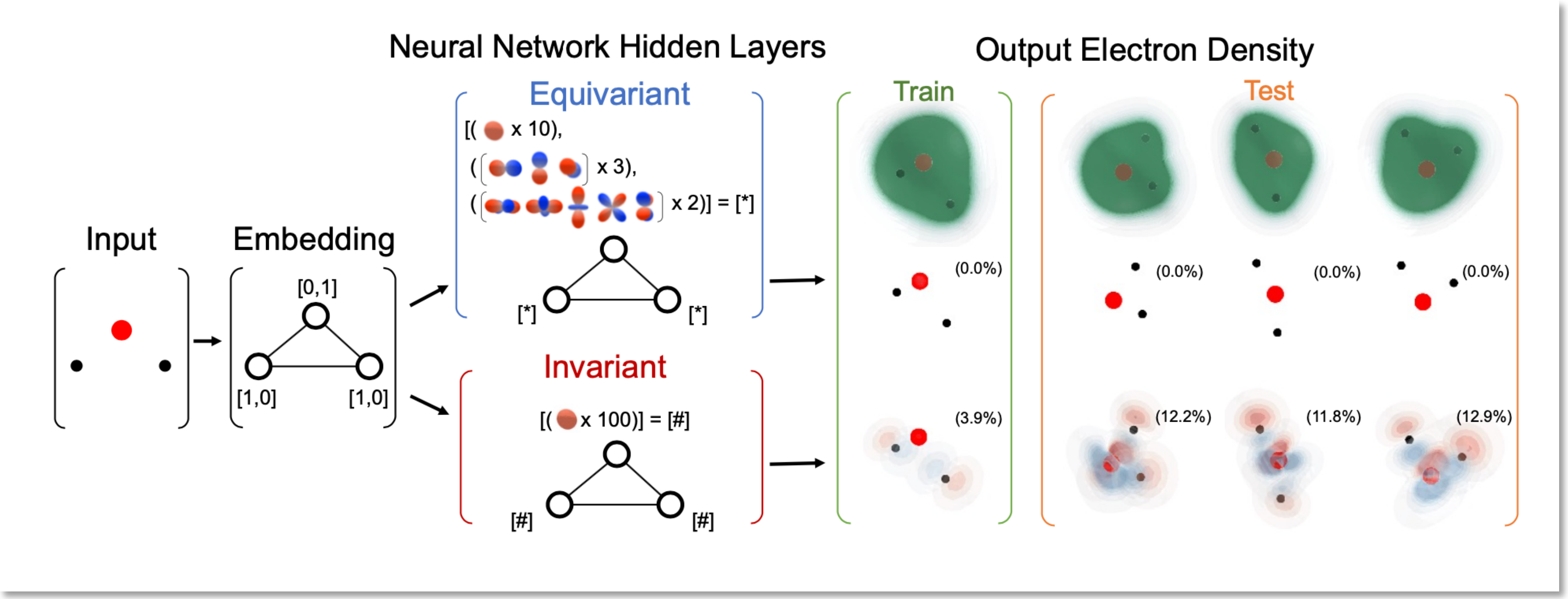
Electron density prediction

spherical harmonics

Electron density: $\rho(r) = \sum_i \sum_k \sum_l \sum_m C_{iklm} Y_{l,m} e^{-\alpha_{ikl}(r-r_i)^2}$

coefficients for each basis function on each atom





Rackers et al., Cracking the Quantum Scaling Limit with Machine Learned Electron Densities

Hamiltonian matrix prediction

$$\hat{H}_{\mathrm{el}}\Psi_{\mathrm{el}} = E_{\mathrm{el}}\Psi_{\mathrm{el}}$$

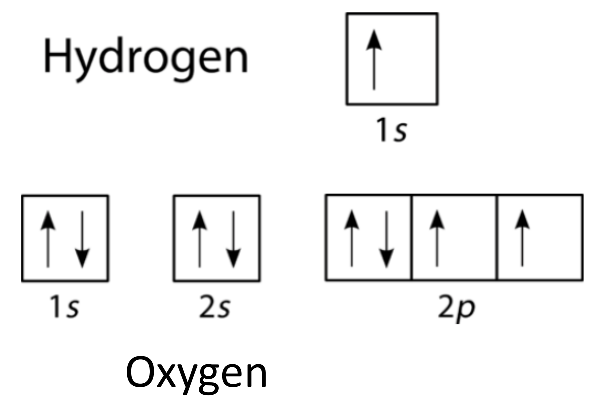
Hamiltonian operator

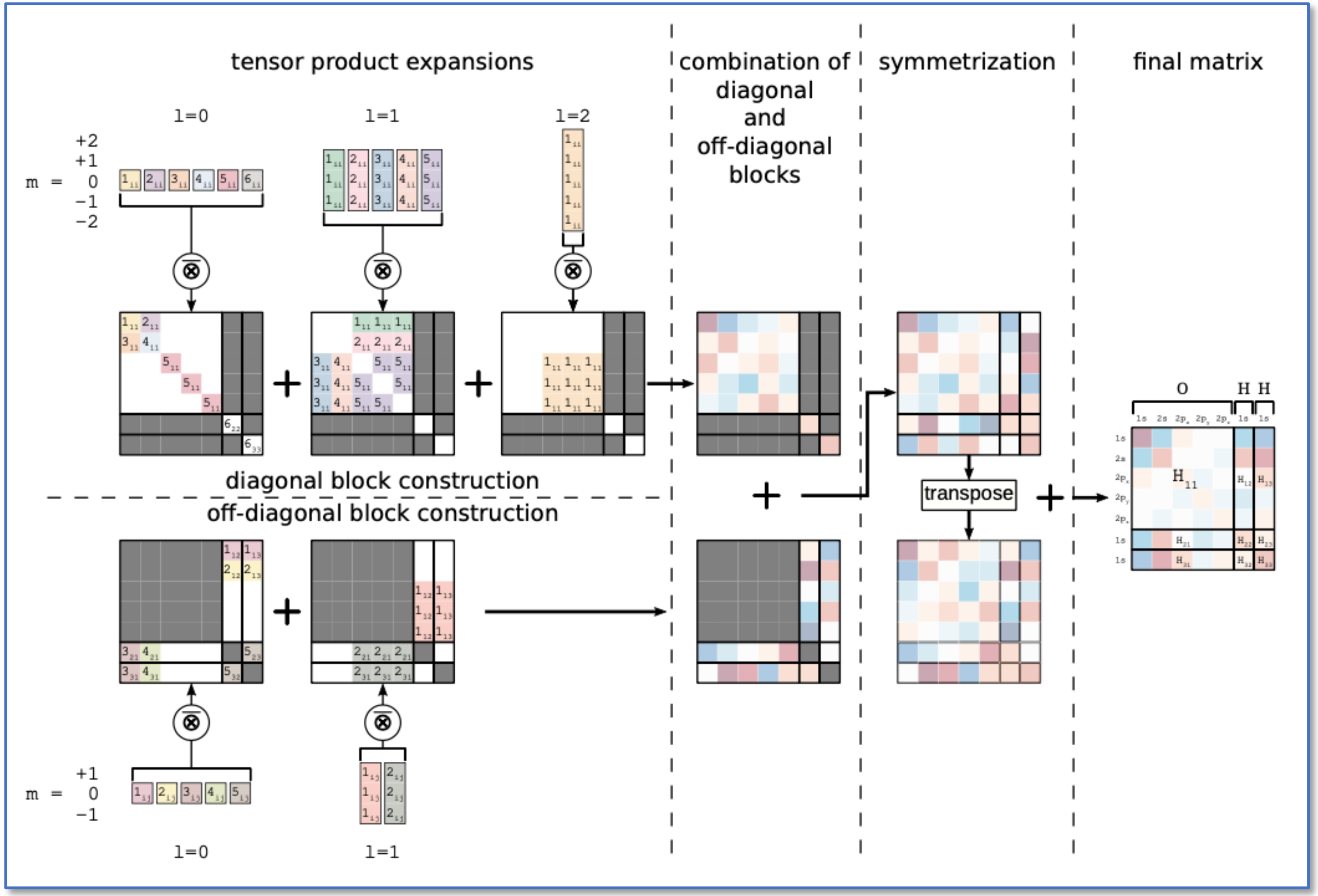
Energy eigenvalues

Electronic wavefunction

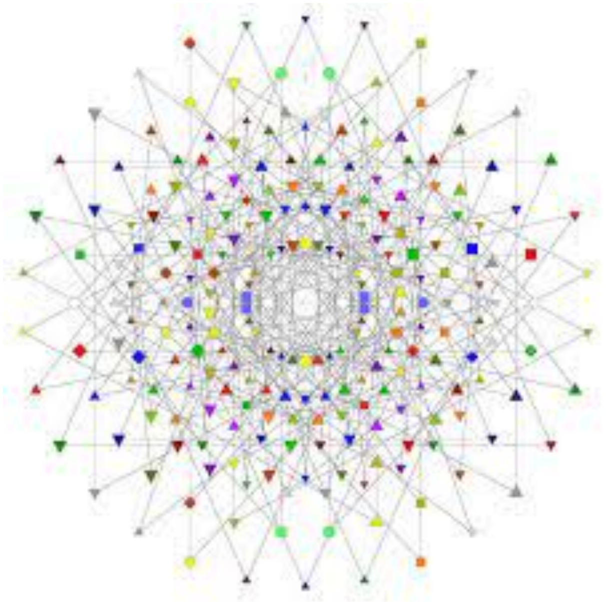
APPROX: $\psi_i = \sum_j C_{ij}\phi_j$

Linear combination of atomic orbitals (LCAO-MO method)





Unke et al., SE(3)-equivariant prediction of molecular wavefunctions and electron densities



THANK YOU FOR YOUR ATTENTION!