Title: Machine Learning of Conserved Quantities and Symmetry Invariants

Speakers: Sebastian Wetzel

Collection: Machine Learning for Quantum Many-Body Systems

Date: June 15, 2023 - 4:00 PM

URL: https://pirsa.org/23060101

Machine Learning of Conserved Quantities and Symmetry Invariants

Machine Learning for Quantum Many-Body Systems

Perimeter Institute, Waterloo, June 22, 2023

Sebastian Johann Wetzel

PERIMETER INSTITUTE PIQUIL HOMES+

Overview

- ✗ Neural Solution to Learning Symmetry Invariants and Conserved Quantities
- ✗ Siamese Neural Networks
- ✗ Examples
  - Central Potential
  - Minkowski Spacetime
  - Electromagnetic Fields
- ✗ Recent Developments

Wetzel, et al, PRR 2020

Ha, Jeong, PRR 2021

Liu, Madhavan, Tegmark, PRE 2022

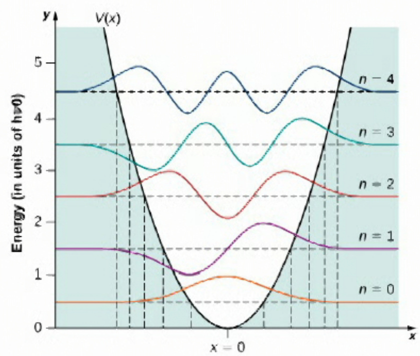
Zhu, Zhang, Kevrekidis, arXiv 2023

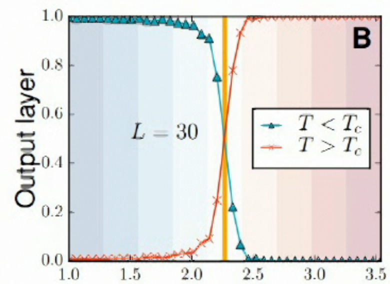
Central Physical Concepts

Wave Functions

Phase Diagrams

Partition Function





$$Z = \int \mathcal{D}\phi \; e^{iS[\phi]}$$

Neural Network Quantum States

Neural Network Tomography

Classifying Simulated / Experimental Snapshots

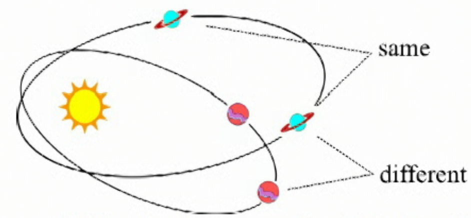
ML assisted MC

Normalizing Flows

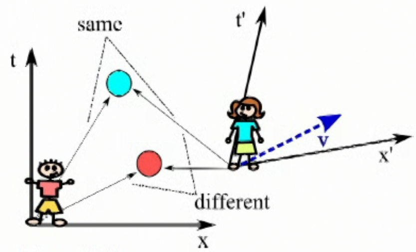
Symmetry Invariants and Conserved Quantities

Problem:

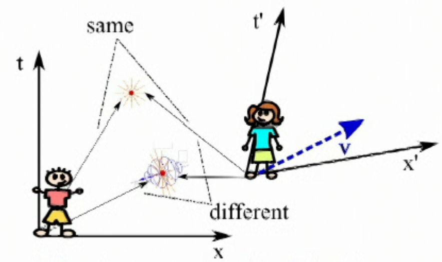
- Given multiple observations from different space-time coordinates of a thing and other things, what are the symmetry invariants and conserved quantities ?



Objects in a central Potential



Euclidean space vs Minkowski spacetime



Electromagnetic Fields

Overview

Interpretable conservation law estimation by deriving the symmetries of dynamics from trained deep neural networks

Yoh-ichi Mototake*
The Institute of Statistical Mathematics, Tachikawa, Tokyo 190-8562, Japan

(Received 19 April 2020; accepted 8 February 2021; published 18 March 2021)

Conservation laws and spin system modeling through principal component analysis ☆

David Yevick

Department of Physics, University of Waterloo, Waterloo, ON N2L 3G7, Canada

Received 18 May 2020, Revised 17 November 2020, Accepted 6 January 2021, Available online 14 January 2021, Version of Record 27 January 2021.

Discovering invariants via machine learning

Seungwoong Ha and Hawoong Jeong*
Department of Physics, Korea Advanced Institute of Science and Technology, Daejeon 34141, Korea

(Received 8 February 2021; revised 7 June 2021; accepted 4 November 2021; published 9 December 2021)

AI Poincaré: Machine Learning Conservation Laws from Trajectories

Ziming Liu and Max Tegmark
Department of Physics, Massachusetts Institute of Technology, Cambridge, USA
(Dated: April 27, 2021)

AI Poincaré 2.0: Machine Learning Conservation Laws from Differential Equations

Ziming Liu,[1] Varun Madhavan,[2] and Max Tegmark[1]
[1]*Department of Physics, Massachusetts Institute of Technology, Cambridge, USA*
[2]*Indian Institute of Technology Kharagpur, India*
(Dated: November 1, 2022)

Noether Networks: Meta-Learning Useful Conserved Quantities

Ferran Alet*[1], Dylan Doblar*[1], Allan Zhou[2], Joshua Tenenbaum[1], Kenji Kawaguchi[3], Chelsea Finn[2]
[1]MIT, [2]Stanford University, [3]National University of Singapore
{alet,ddoblar}@mit.edu

Discovering Conservation Laws using Optimal Transport and Manifold Learning

Peter Y. Lu,[1,2,*] Rumen Dangovski,[3] and Marin Soljačić[2]
[1]*Data Science Institute, University of Chicago, Chicago, IL 60637, USA*
[2]*Department of Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*
[3]*Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*
(Dated: September 1, 2022)

Machine learning independent conservation laws through neural deflation

Wei Zhu,[1] Hong-Kun Zhang,[1] and P. G. Kevrekidis[1]
[1]*Department of Mathematics and Statistics, University of Massachusetts Amherst, Amherst, MA 01003-4515, USA*
(Dated: March 29, 2023)

Discovering New Interpretable Conservation Laws as Sparse Invariants

Ziming Liu,[1] Patrick Obin Sturm,[2] Saketh Bharadwaj,[3] Sam J. Silva,[2] and Max Tegmark[1]
[1]*Department of Physics, Institute of Artificial Intelligence and Fundamental Interactions, Massachusetts Institute of Technology, Cambridge, USA*
[2]*Department of Earth Sciences, University of Southern California, Los Angeles, USA*
[3]*Department of Chemical Engineering, Indian Institute of Technology, Hyderabad, India*
(Dated: June 7, 2023)

Symmetry Invariants and Conserved Quantities

A function f is a symmetry invariant of a transformation

$$U : D \to D$$

if

$$f(x) = f(Ux) \qquad \forall x \in D$$

A symmetry invariant f is a conserved quantitiy of a differential equation if $x(t) = U_t x$ is a solution of

$$\frac{d}{dt}x = K(x,t)$$

If the dynamics of $x(t) = U_t x$ is described by a Hamiltonian H

$$\frac{d}{dt}f = \frac{\partial}{\partial t}f + \{H, f\} = 0$$

is fulfilled by conserved quantities.

Symmetry Invariants and Conserved Quantities

A function f is a symmetry invariant of a transformation

$$U : D \to D$$

if

$$f(x) = f(Ux) \qquad \forall x \in D$$

A symmetry invariant f is a conserved quantitiy of a differential equation if $x(t) = U_t x$ is a solution of

$$\frac{d}{dt} x = K(x, t)$$

If the dynamics of $x(t) = U_t x$ is described by a Hamiltonian H

$$\frac{d}{dt} f = \frac{\partial}{\partial t} f + \{H, f\} = 0$$

is fulfilled by conserved quantities.

More knowledge

Machine Learning of Symmetry Invariants and Conserved Quantities

- Given a dataset of observations

$$X = \{(x_1^1, ..., x_n^1), ..., (x_1^k, ..., x_m^k)\}$$

- How do we build a Neural Network f that becomes a symmetry invariant / conserved quantity?

Machine Learning of Symmetry Invariants and Conserved Quantities

- Given a dataset of observations

$$X = \{(x_1^1, ..., x_n^1), ..., (x_1^k, ..., x_m^k)\}$$

- How do we build a Neural Network f that becomes a symmetry invariant / conserved quantity?
- For all data points where $\exists U :$ $x' = Ux$ to learn

$$f(x) = f(x')$$

Machine Learning of Symmetry Invariants and Conserved Quantities

- Given a dataset of observations

$$X = \{(x_1^1, ..., x_n^1), ..., (x_1^k, ..., x_m^k)\}$$

- How do we build a Neural Network f that becomes a symmetry invariant / conserved quantity?
- For all data points where $\exists U: \quad x' = Ux$ to learn

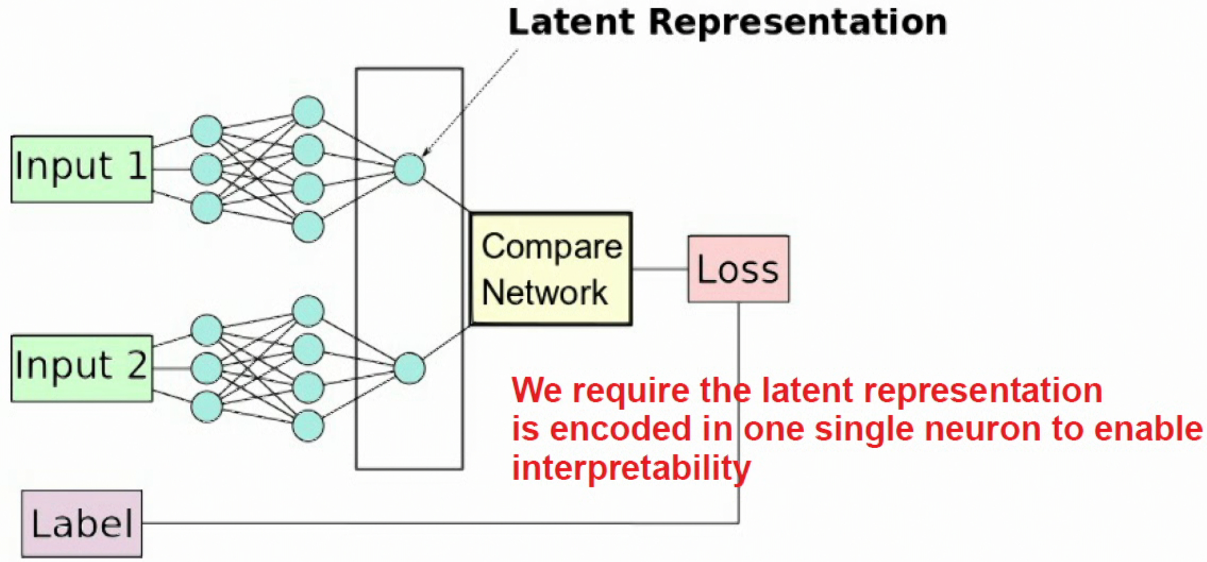
$$f(x) = f(x')$$

Problem 1: We need a network architecture that is applied to 2 data points at the same time

Problem 2: Enforcing this property leads to

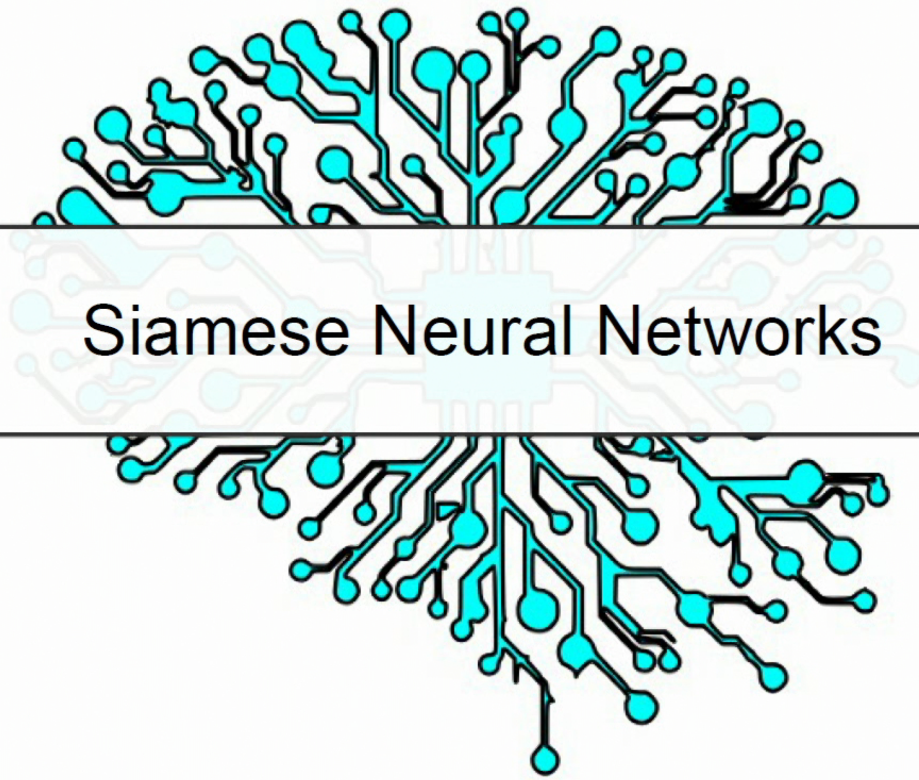
$$f(x) = \text{const}$$

Siamese Neural Networks

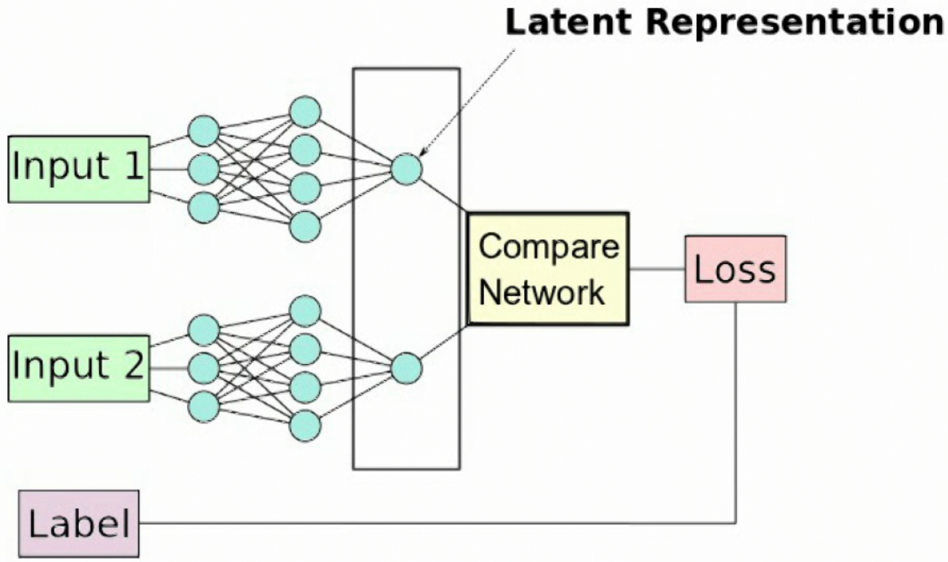


- Input : Pair of Physical Configurations
- Label : same / different
- Network pair contains identical neural networks with shared weights
- Different architectures: triplet, direct latent comparison

Siamese Neural Networks



Siamese Neural Networks



- Input : Pair of Images
- Label : same / different
- Network pair contains identical neural networks with shared weights
- Different architectures: triplet, direct latent comparison

Siamese Neural Networks

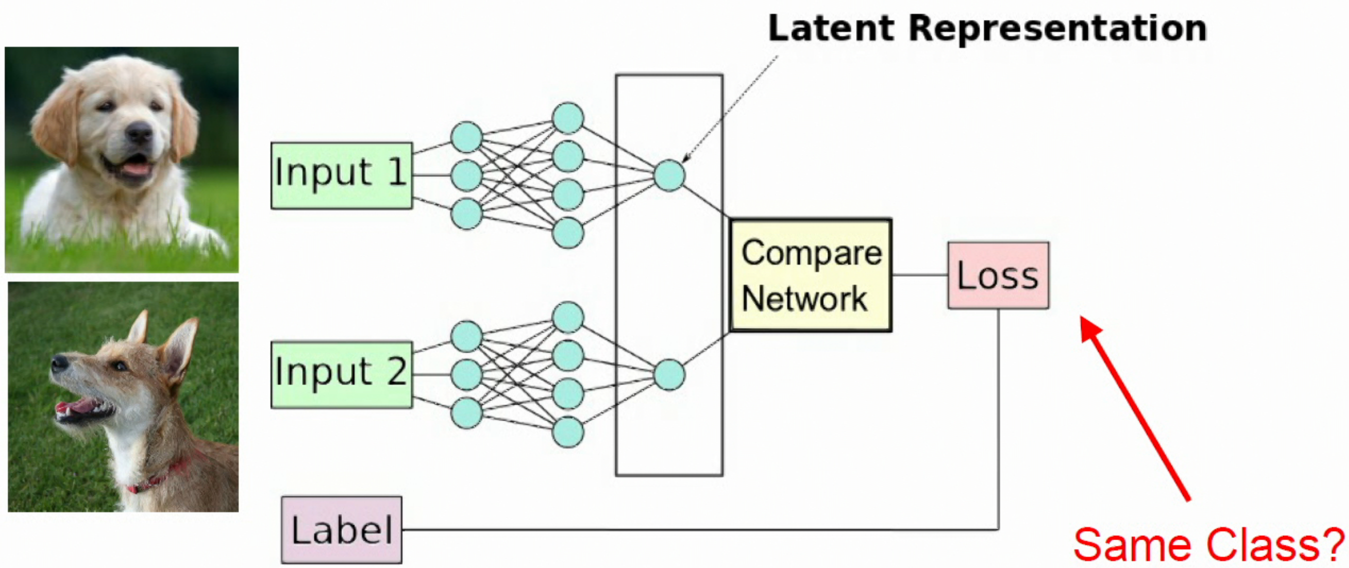
- 1994: Introduction of Siamese Networks
  - Fingerprint identification — *Baldi, Chauvin, Neural Computation 1993*
  - Signature verification — *Bromley, Guyon, LeCun, Säckinger Shah, NIPS 1994*
- 2014: DeepFace solves face recognition problem with human level accuracy of 97%

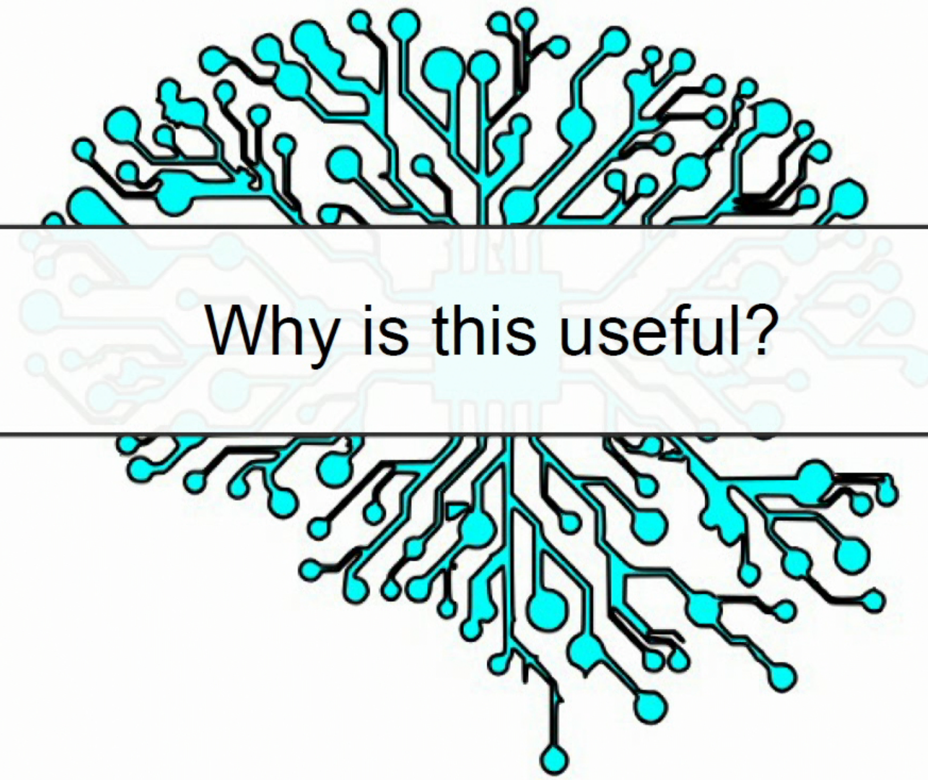
  Taigman, Yang, Ranzato, Wolf
  Facebook Research 2014

- 2015: Facebook starts using the technology to identify its users (exception: EU due to data privacy laws).

Similarity Learning

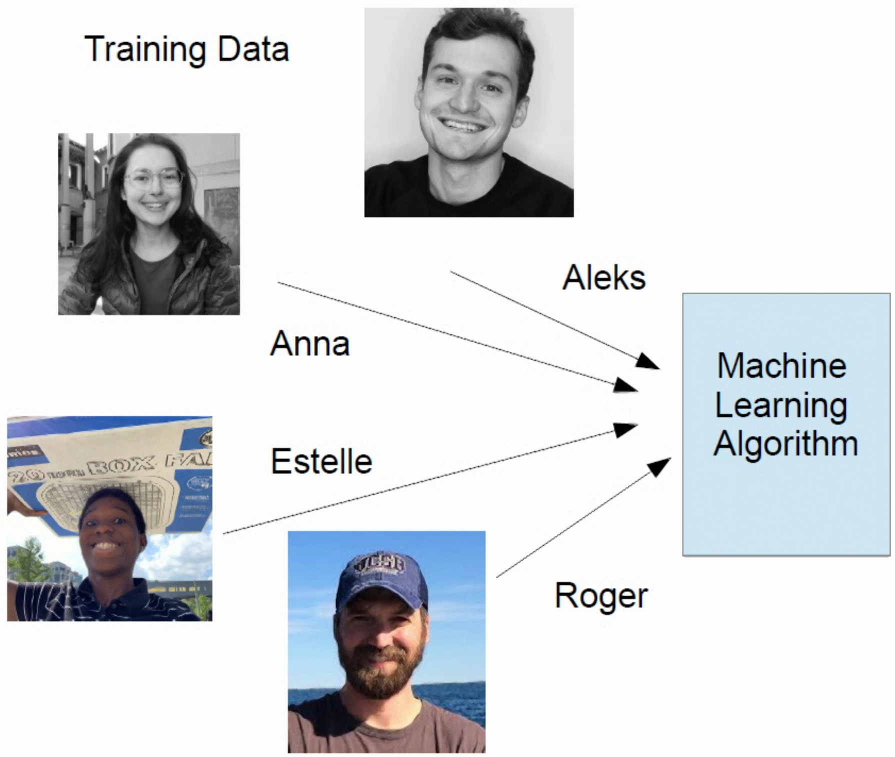
(Class-) Similarity learning *is a machine learning technique used to learn the general features of a dataset* ***with labels*** *by teaching the model which data points belong to the same or to different classes.*



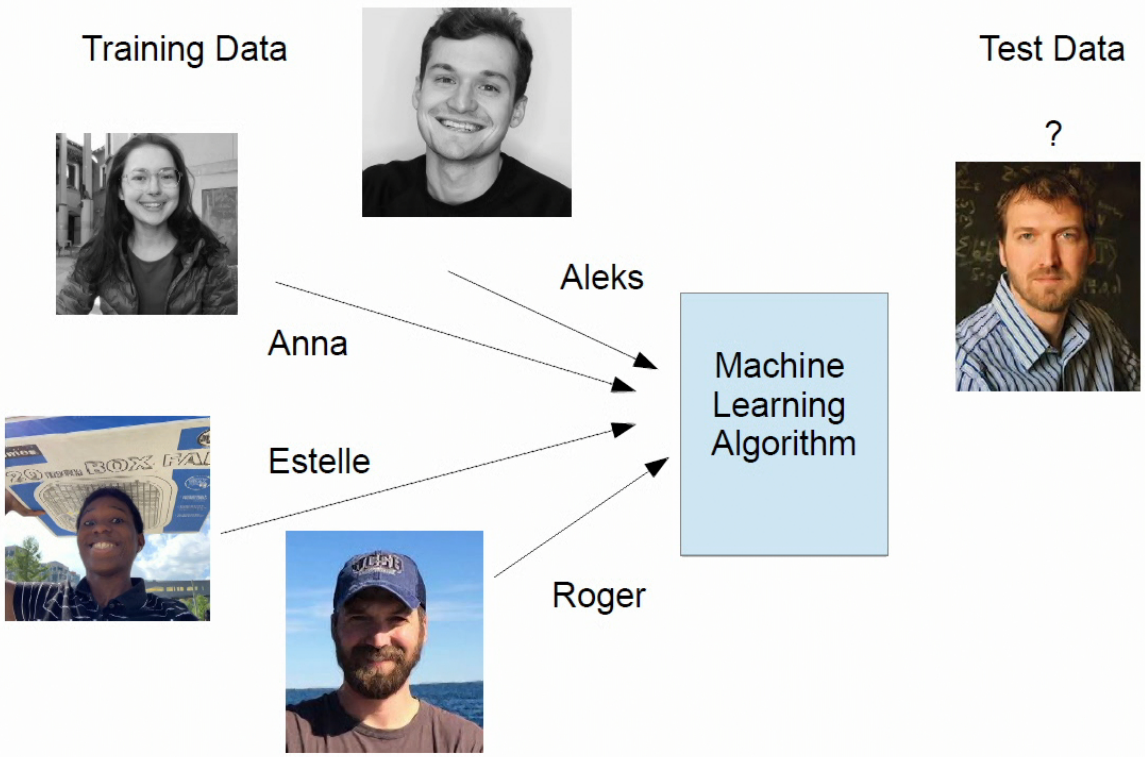


Why is this useful?

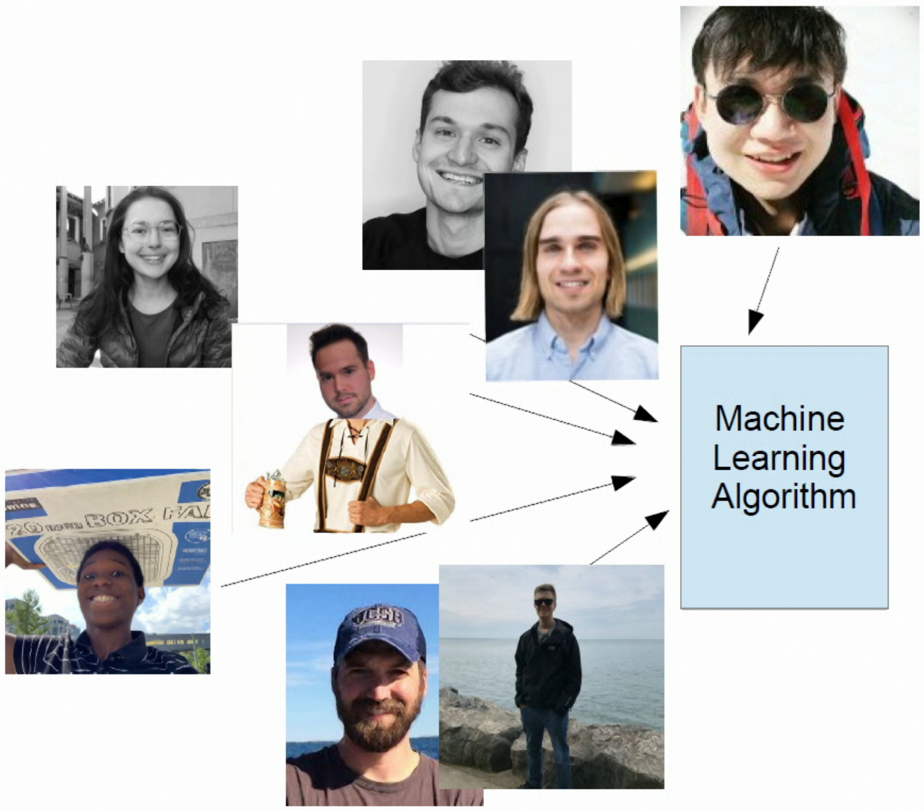
Machine Learning
Multi Class Classification



Machine Learning
Multi Class Classification



Machine Learning
Infinite Class Classification



Problems:

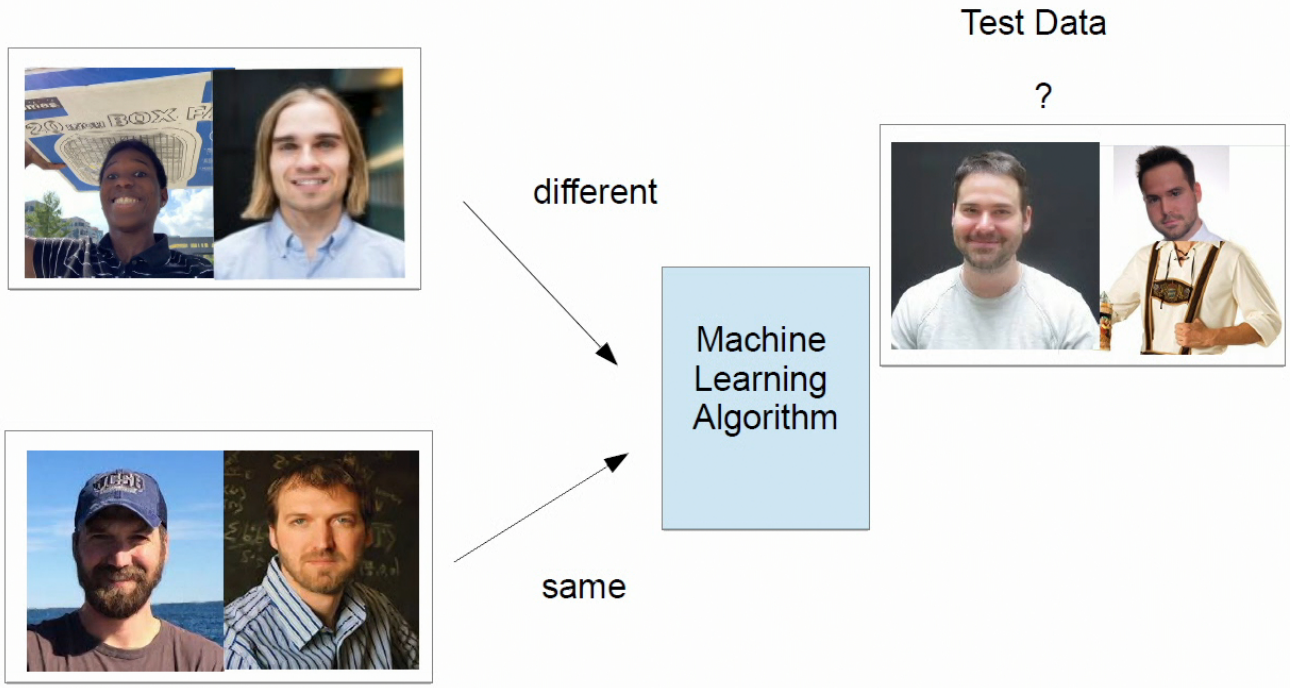
- ✗ New classes
- ✗ Class imbalance
- ✗ Classes without training data.

Can this be solved smarter?

Machine Learning
Infinite Class Classification

Reformulation of the Problem:

- Teach a maching learning algorithm if two pictures show the same class.

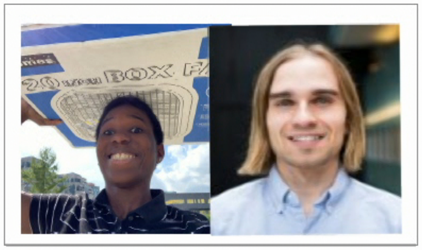


Machine Learning
Infinite Class Classification

Reformulation of the Problem:

- Teach a maching learning algorithm if two pictures show the same class.

Test Data



different

same

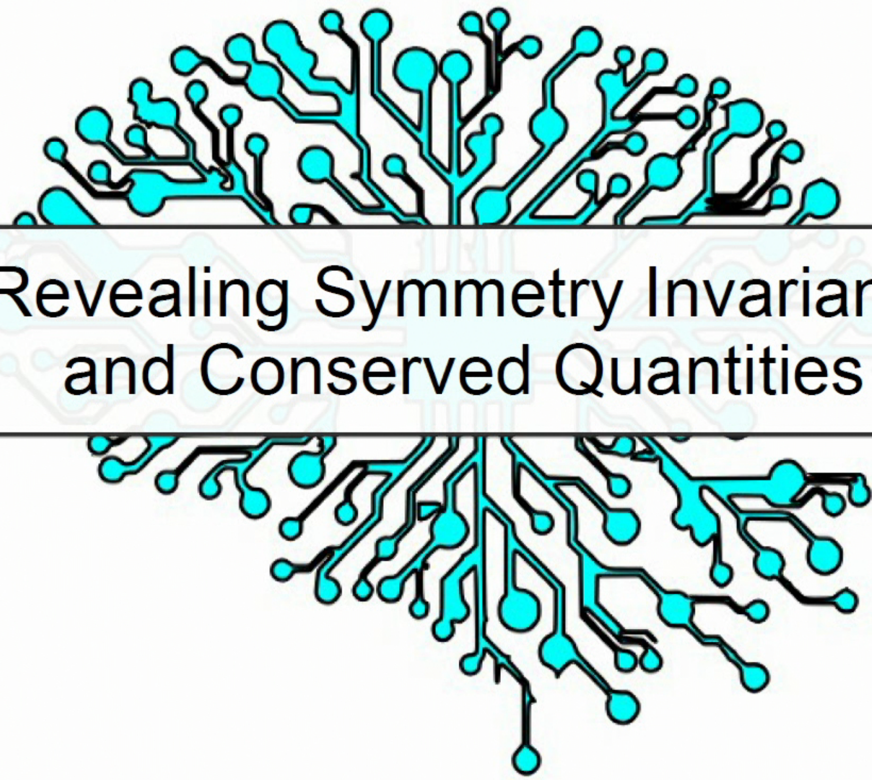
Machine Learning Algorithm

?

same

Reference Picture *Test Picture*

(Does not need to be in training set)



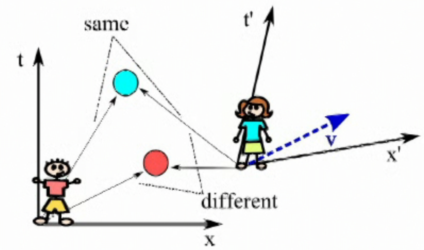
Revealing Symmetry Invariants and Conserved Quantities

Wetzel, et al, PRR 2020

Lorentz Transformation in Special Relativity

Problem:

- Given two observations, do they belong to the same event in Minkowski spacetime?



SNN Solution:

- Prepare Dataset of positive data with datapoints satisfying

$$X^{\mu} = (\mathbf{x}^{\mu}, \mathbf{x}'^{\mu} = \Lambda_{\nu}{}^{\mu}\mathbf{x}^{\nu})$$

- Prepare Negative Dataset by permuting positive dataset
- Train SNN to distinguish between positive and negative pairs

Lorentz Transformation in Special Relativity

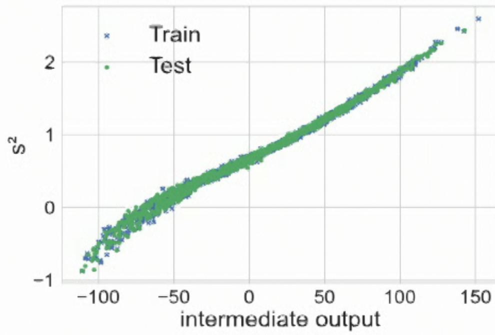
Results:

Training accuracy : 94%

Test accuracy : 92%

- Interpretation by polynomial regression on latent representation:

$$
\begin{aligned}
f(\mathbf{x}) \approx & -87.41t^2 - 60.48 - 0.11x \\
& - 0.10yz + 0.04ty + 0.06z \\
& + 0.07y + 0.10tx + 0.12tz \\
& + 0.15xz + 0.21xy + 2.50t \\
& + 88.10z^2 + 88.61y^2 + 88.63x^2 \\
\approx & 88\underbrace{\left(-t^2 + x^2 + y^2 + z^2\right)}_{=s^2} - 60
\end{aligned}
$$

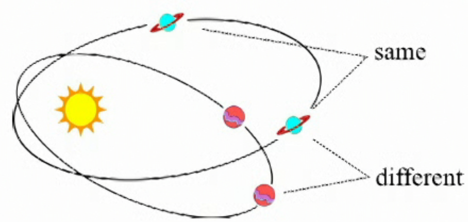


- Network has learned the space time interval to infer its prediction

Motion of Particle in Gravitational Potential

Problem:

- Given two observations of positions and velocities, do they belong to the same particle trajectory?



SNN Solution:

- Prepare Dataset of positive data where the pair is connected by solving the equations of motion

$$((x, y, v_x, v_y), (x', y', v'_x, v'_y))$$

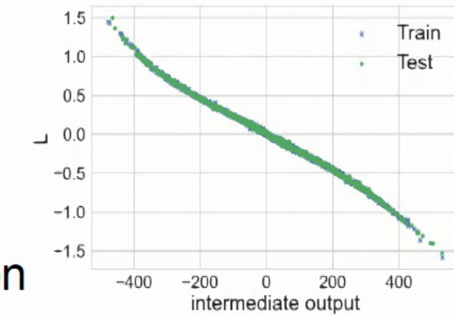
- Prepare Negative Dataset by permuting positive dataset
- Train SNN to distinguish between positive and negative pairs

Motion of Particle in Gravitational Potential

Results:

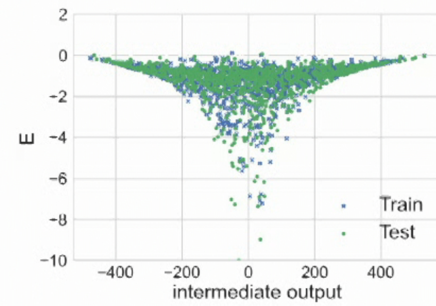
Training accuracy : 98%

Test accuracy : 97%



- Interpretation by polynomial regression on latent representation:

$$
\begin{aligned}
f(\mathbf{x}) \approx & -403.71 x v_y - 4.85 x - 0.58 x y \\
& - 0.17 x v_x - 0.02 v_y^2 - 0.01 v_x v_y \\
& + 0.00 v_y^2 + 0.01 v_y + 0.02 v_x \\
& + 0.45 x^2 + 0.66 y^2 + 0.74 \\
& + 0.99 y v_y + 1.24 y + 402.44 y v_x \\
\approx & -403 \underbrace{(x v_y - y v_x)}_{= L_z}
\end{aligned}
$$



- Network has learned the angular momentum to infer its prediction.

Motion of Particle in Gravitational Potential

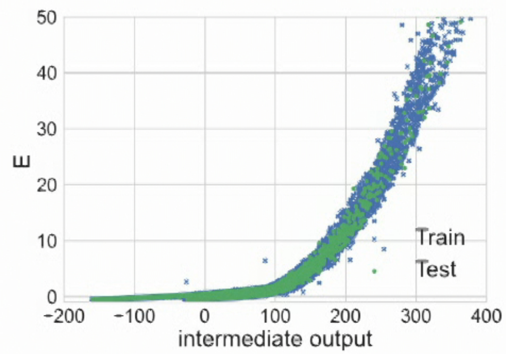
Results for fixed angular momentum:

Training accuracy : 99%

Test accuracy : 99%

- Interpretation by polynomial regression on latent representation:

$$f(\mathbf{x}) \approx -174.57 - 88.28\frac{1}{r} - 87.39 y v_x$$
$$- 1.43\frac{1}{r^2} + \ldots + 1.27\frac{x}{r}$$
$$+ 46.22 v_x^2 + 46.53 v_y^2 + 87.18 x v_y$$
$$\approx -175 + 87\underbrace{(x v_y - y v_x)}_{=L_z=\text{const}} + 90\underbrace{(\frac{1}{2}v_x^2 + \frac{1}{2}v_y^2 - \frac{1}{r})}_{=E}$$

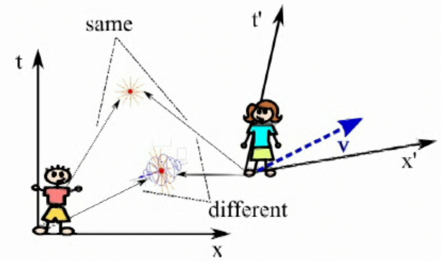


- Network has learned the energy to infer its prediction.

Lorentz Transformation of Electromagnetic Fields

Problem:

- Given two field configurations, can they be transformed into each other by a Lorentz transformation?



SNN Solution:

- Prepare Dataset of positive data where the pair is connected by a Lorentz Transformation

$$((E_x, E_y, E_z, B_x, B_y, B_z), (E'_x, E'_y, E'_z, B'_x, B'_y, B'_z))$$

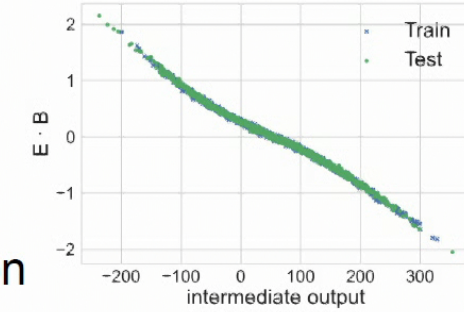
- Prepare Negative Dataset by permuting positive dataset
- Train SNN to distinguish between positive and negative pairs

Lorentz Transformation of Electromagnetic Fields

Results:

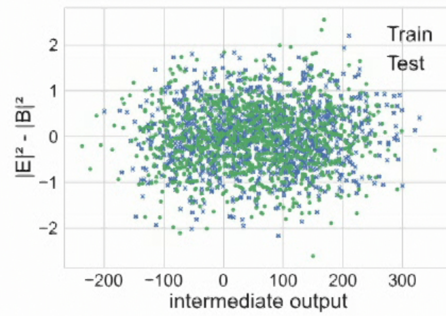
Training accuracy : 95%

Test accuracy : 94%



- Interpretation by polynomial regression on latent representation:

$$f(\mathbf{x}) \approx -170.53E_2B_2 - 170.22E_1B_1 - 170.20E_3B_3 - 4.13B_3^2 + \ldots + 4.92E_2^2 + 53.43 \approx -170\underbrace{(E_1B_1 + E_2B_2 + E_3B_3)}_{=E\cdot B} + 53$$



- Network has learned the determinant of the field strength tensor to infer its prediction.

Machine Learning of Symmetry Invariants and Conserved Quantities

Problem 1

- Any function that only depends on an invariant f is itself an invariant
- Thus there is no need for the NN to learn a human readable version of the invariant.

Solution

- Give up human readability
- As we have seen above NN tend to learn the simplest form of invariants that humans are used to.
- More efficient symbolic regression scheme

Machine Learning of Symmetry Invariants and Conserved Quantities

Problem 2

- Finding Multiple Invariants is not scalable (according to previous slides)

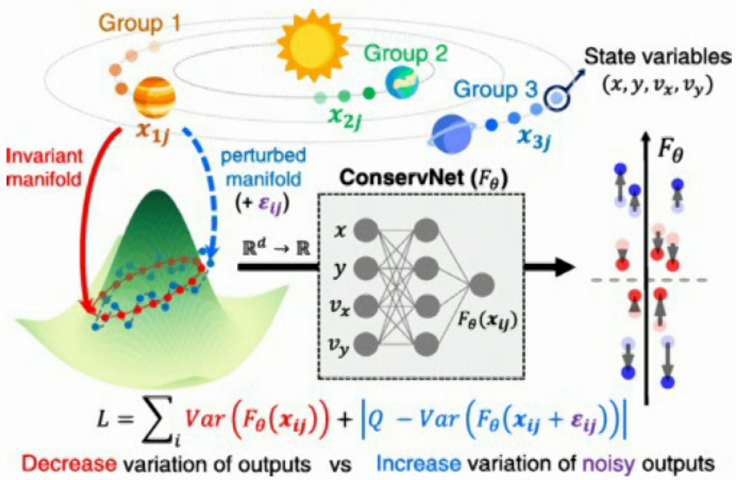
Solution

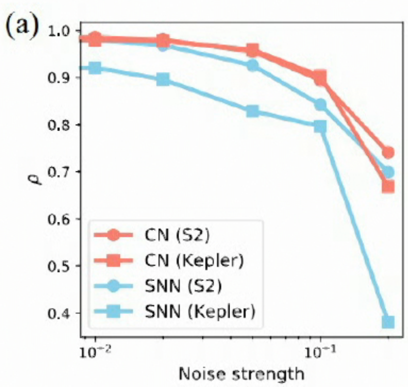
- Enforce Independence in loss function
- See following slides

Recent Developments

Discovering conservation laws from trajectories via machine learning

- Data: Trajectories
- Create negative data pairs through perturbing trajectories
- More accurate than SNN





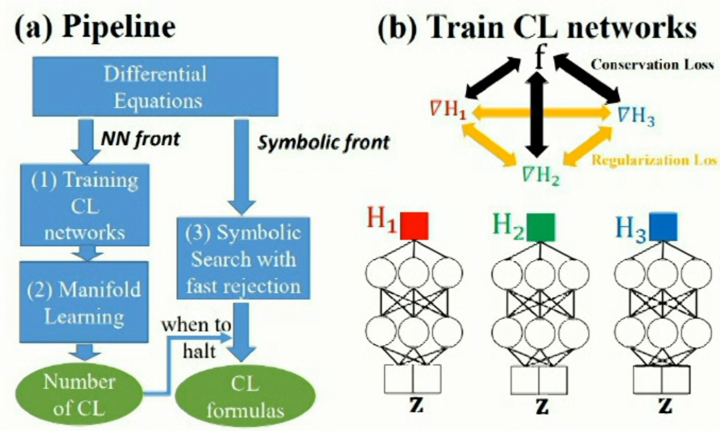
Ha, Jeong, PRR 2021

AI Poincare 2.0: Machine Learning Conservation Laws from Differential Equations

- Data: Trajectories / Differential Equations

- Pipeline involving symbolic regression

- Enforce linearly independent conserved quantities



$$R(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \equiv \frac{1}{P} \sum_{i=1}^{P} \left| \widehat{\nabla H_1}(\mathbf{z}^{(i)}; \boldsymbol{\theta}_1) \cdot \widehat{\nabla H_2}(\mathbf{z}^{(i)}; \boldsymbol{\theta}_2) \right|^2$$

Liu, Tegmark, PRL 2022

Liu, Madhavan, Tegmark, PRE 2022

Liu et al, arxiv 2023

Machine learning independent conservation laws through neural deflation

- A maximal set of functionally independent, Poisson-commuting conservation laws
- Require Hamiltonian system

$$\frac{d}{dt}x = h(x),\ \ h(x) = J(x)\nabla H(x)$$

- In addition to linear independence enforce

$$\{I_j, I_k\} = 0$$

Zhu, Zhang, Kevrekidis, arXiv 2023

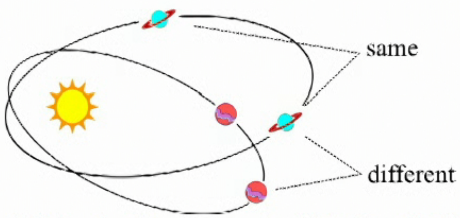
Conclusion

Neural Networks can Learn symmetry invariants and conserved quantities

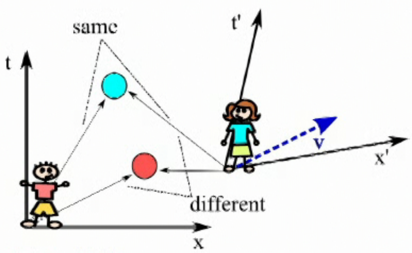
- Interpretable through symbolic regression
- If interpretability is not required:

  use NN as invariant itself

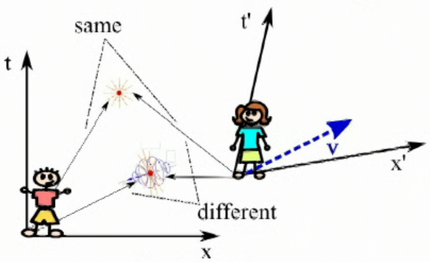
- Full set, independent, commuting



Objects in a central Potential
→ Energy, Angular Momentum



Euclidean space vs
Minkowski spacetime
→space time interval



Electromagnetic Fields
→ Determinant/Contraction of Field Strength Tensor