Machine Learning for Beyond The Standard Model Physics

27.10.2022, ACAT 2022 Bari

Sven Krippendorf (sven.krippendorf@physik.uni-muenchen.de, @krippendorfsven)

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Our purpose in theoretical physics is not to describe the world as we find it, but to explain - in terms of a few fundamental principles - why the world is the way it is.

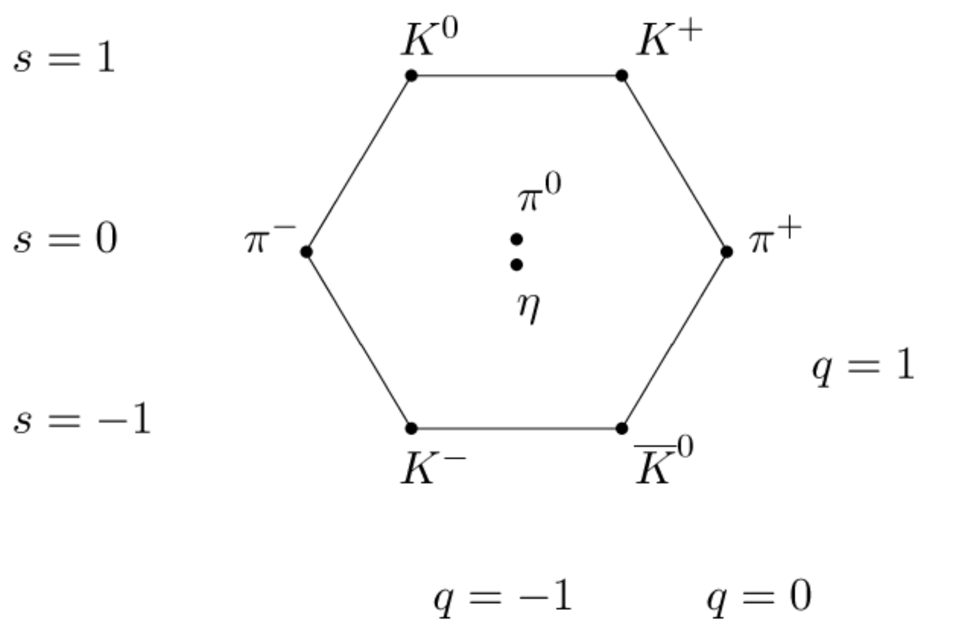
Steven Weinberg

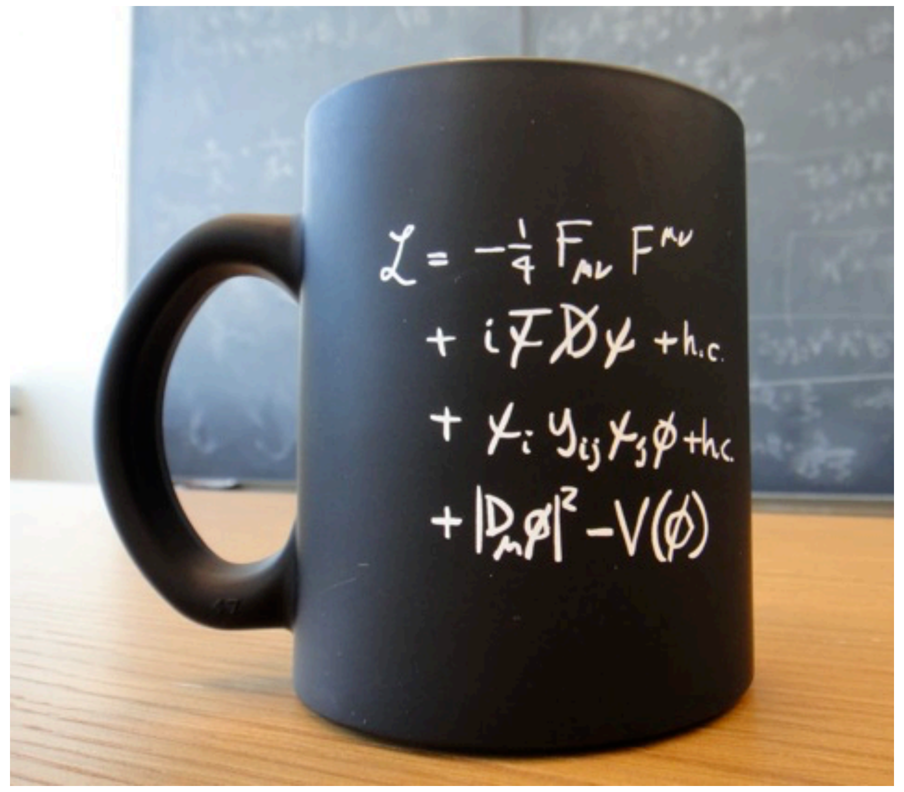
What are these fundamental principles lying beyond our Standard Models?

Can ML reveal them?

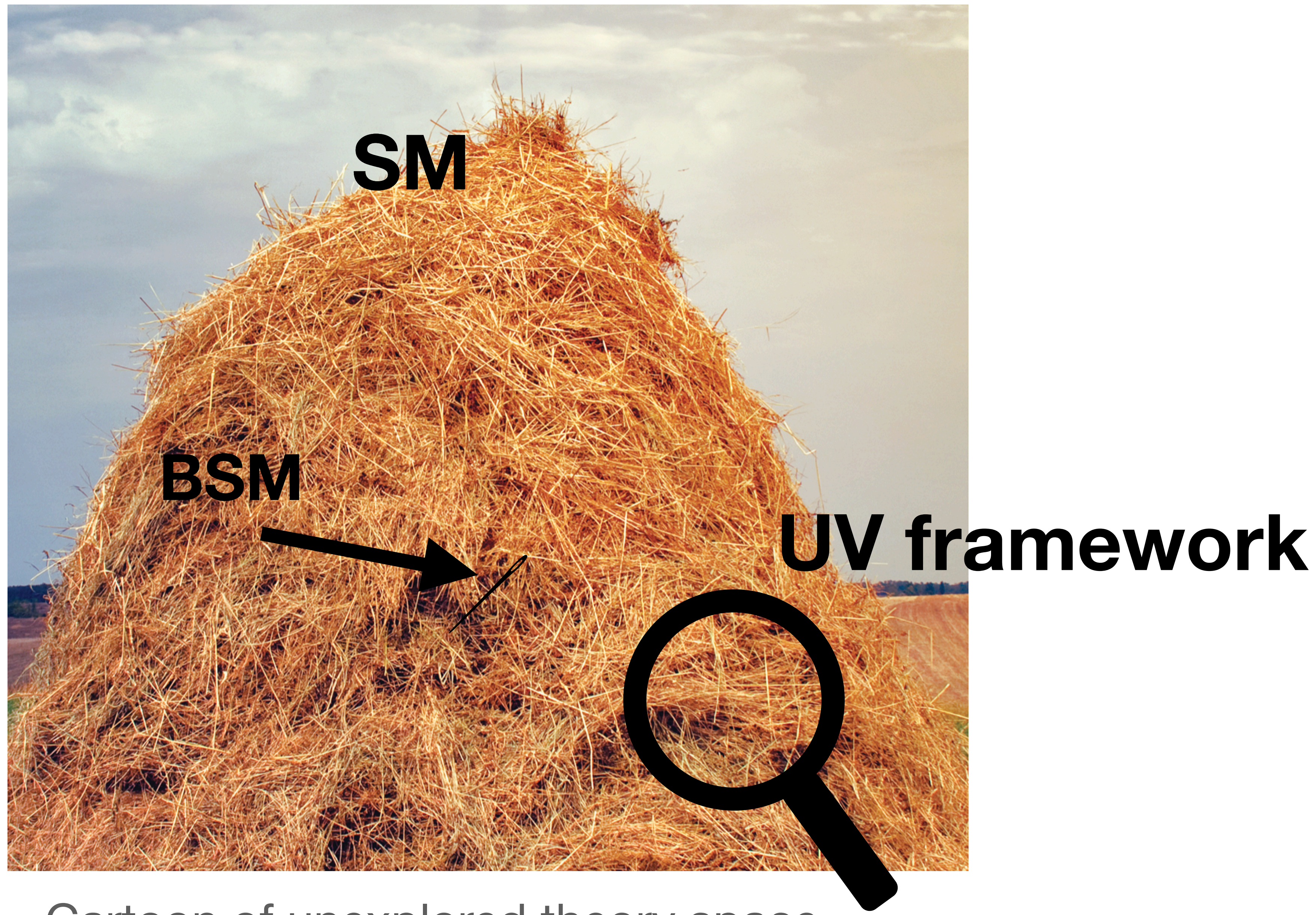
Understanding BSM physics with ML

- Finding where to look for BSM physics, e.g. via: Goodness of fit (cf. Wulzer's talk), Anomaly Detection (cf. Kasieczka's talk)
- Beyond knowing where to look, we would like to understand which Lagrangian is describing our new physics.
  What are the building blocks (mathematical structures) of BSM physics?
- This has been at the heart of theorists' work over decades, the development of the Standard Model being the prime example. Still the theory parameter space is widely unexplored.
  Problem: HUGE search space





Examples of humanly identified building blocks

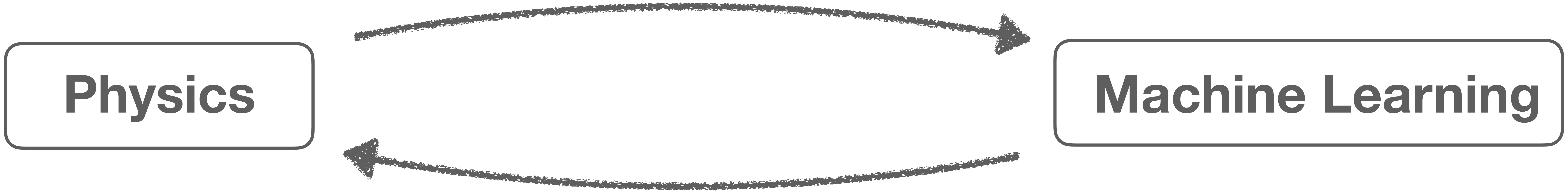


Cartoon of unexplored theory space

Physics ∩ ML
Finding structures in the wider perspective

<u>emerging field:</u>
Gur-Ari et al., Minerva Undergrad physics
Polu et al., Undergrad maths
Charton et al., Maths with Transformers
…

If we have true artificial intelligence, it needs to be able to do theoretical physics and mathematics. What is needed to build such a system? *It does not work out of the box, dedicated design necessary!*



Algorithms for identifying pattern/structure in huge search spaces (e.g. image, text generation)

Searching for (new) structures

General pipeline

- Finding new/unknown structures is not a supervised learning problem.
- Supervised problems can only help for the actual unsupervised problem.
- Defining the optimisation problem is problem specific at this stage. Nevertheless there are already general lessons.
- **Four steps:**
    1. Defining optimisation problem
    2. Selecting the right data for solving the optimisation problem
    3. Selecting a suitable architecture
    4. Evaluating the result and connecting with other pipelines
- This approach is not limited to mathematical structures but also applies for phenomenological models.
- Advantage in mathematical data: no noise and detector effects

Optimisation problem

Data selection

Architecture selection

Evaluation

Content

Examples of identifying mathematical structures with ML

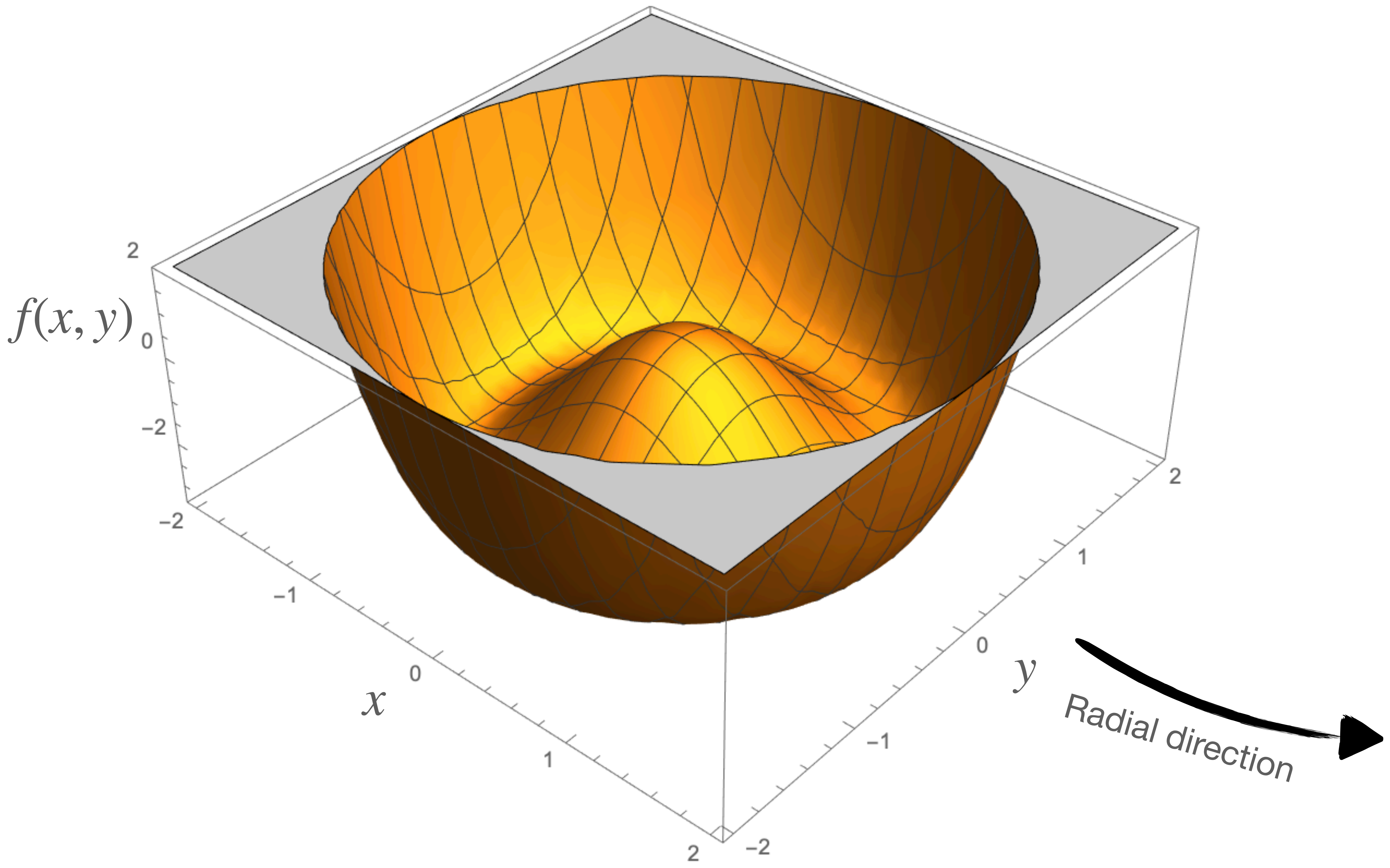
- Today's focus: unsupervised ML to look for finding symmetries and integrability in physical systems as a warm-up
  - No direct optimisation available: Symmetries from embedding layer [arXiv:2003.13679]
  - Symmetries from samples of phase space [arXiv:2104.14444]
  - Towards new physics applications: integrability from samples of phase space [arXiv:2103.07475]

Symmetries from embedding layer

Krippendorf, Syvaeri 2020

How to search for symmetries?

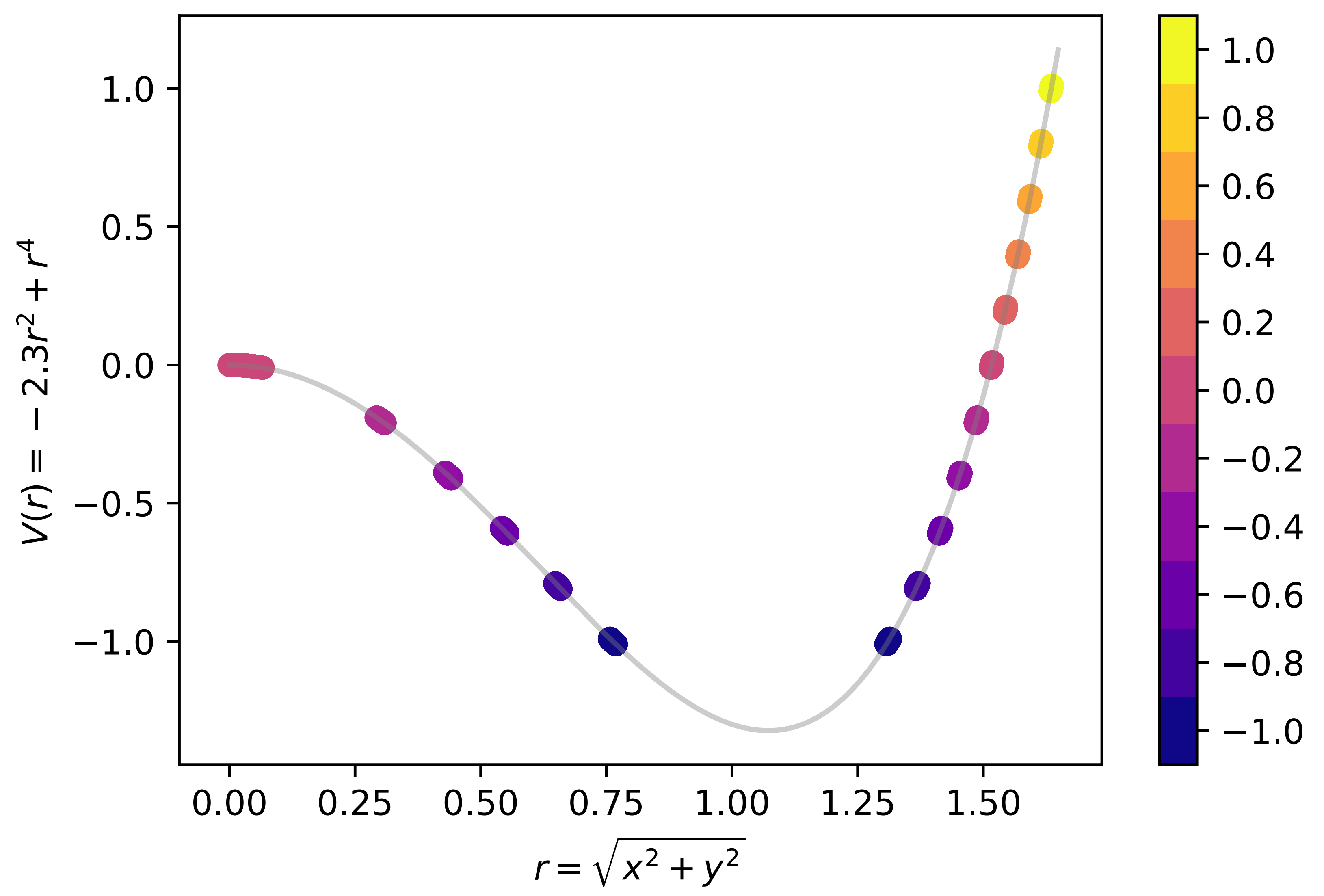
The problem



1. How to find invariances?

$$f(\phi) = f(\tilde{\phi})$$

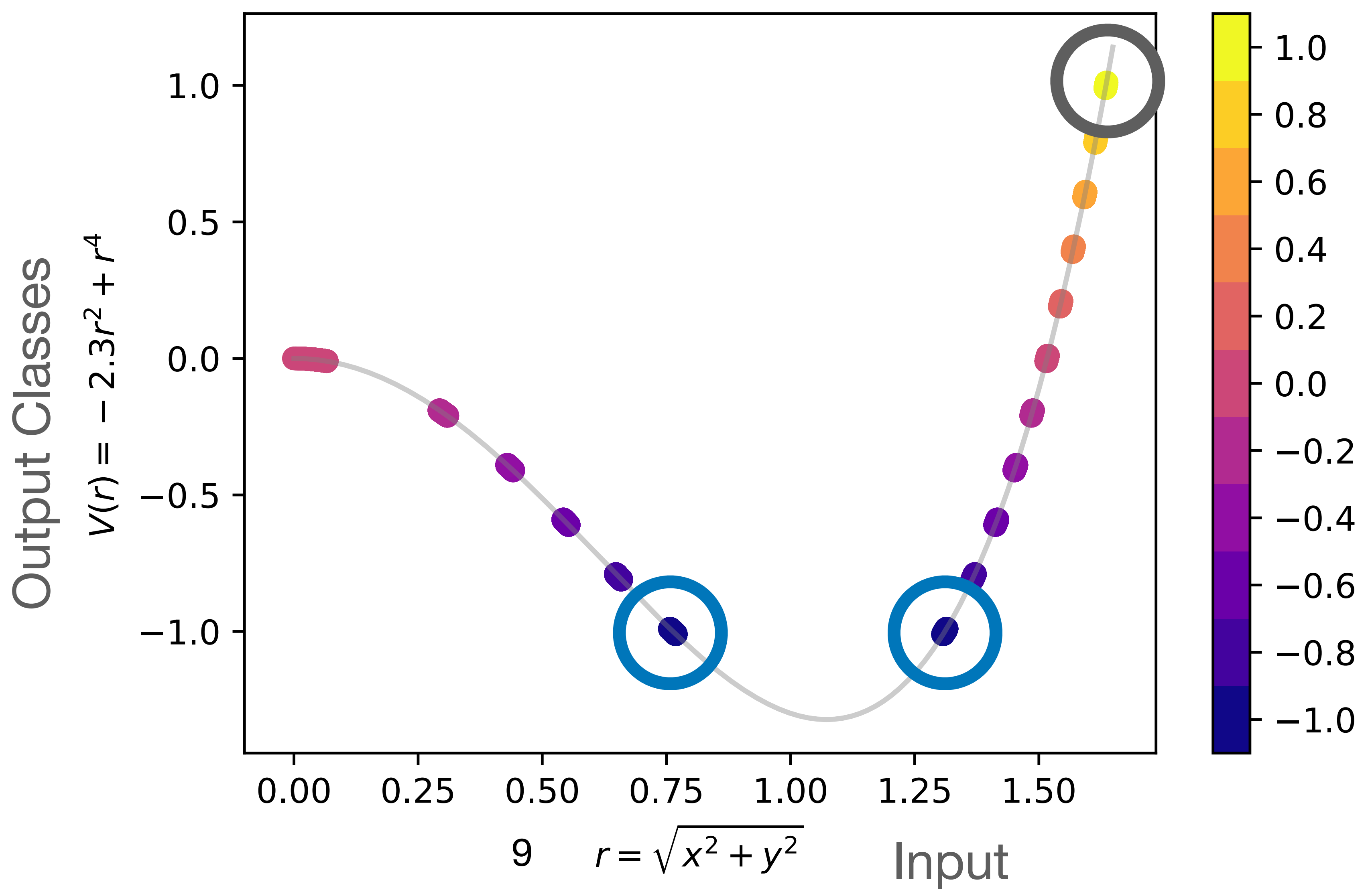
2. Which symmetry is behind such an invariance?



How to search for symmetries?

No direct optimisation available: embedding in deep layer

We need: group input with the same meaning together

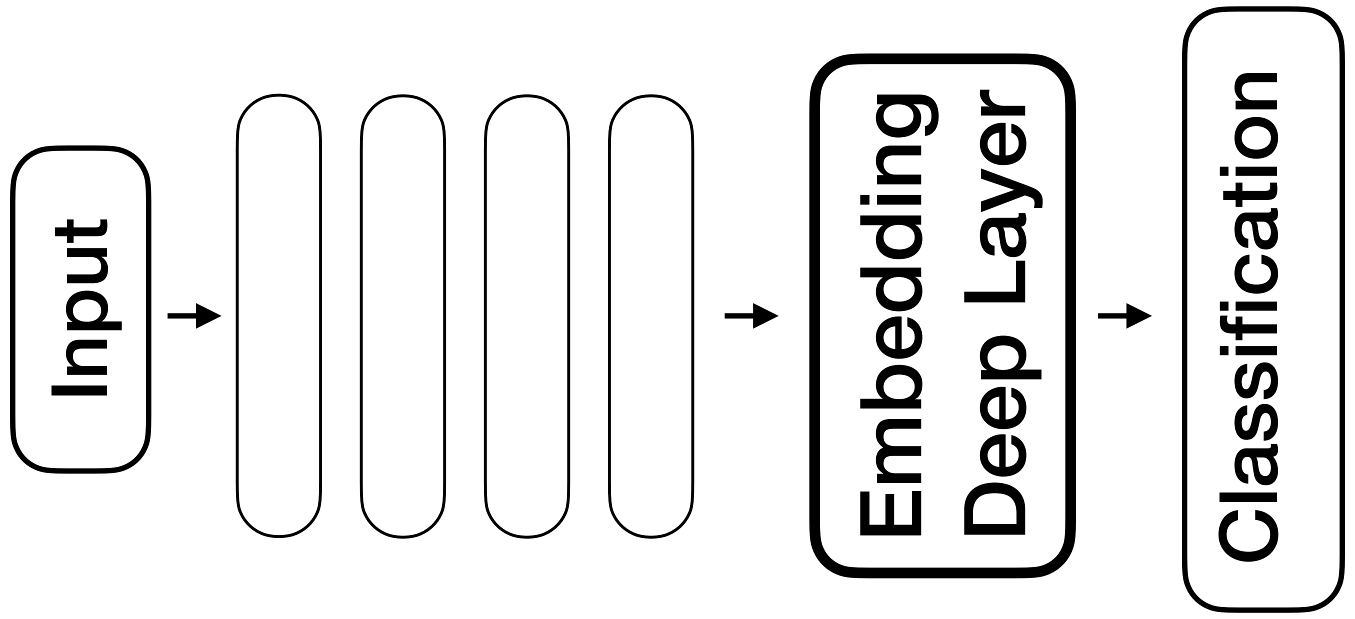


Krippendorf, Syvaeri 2020

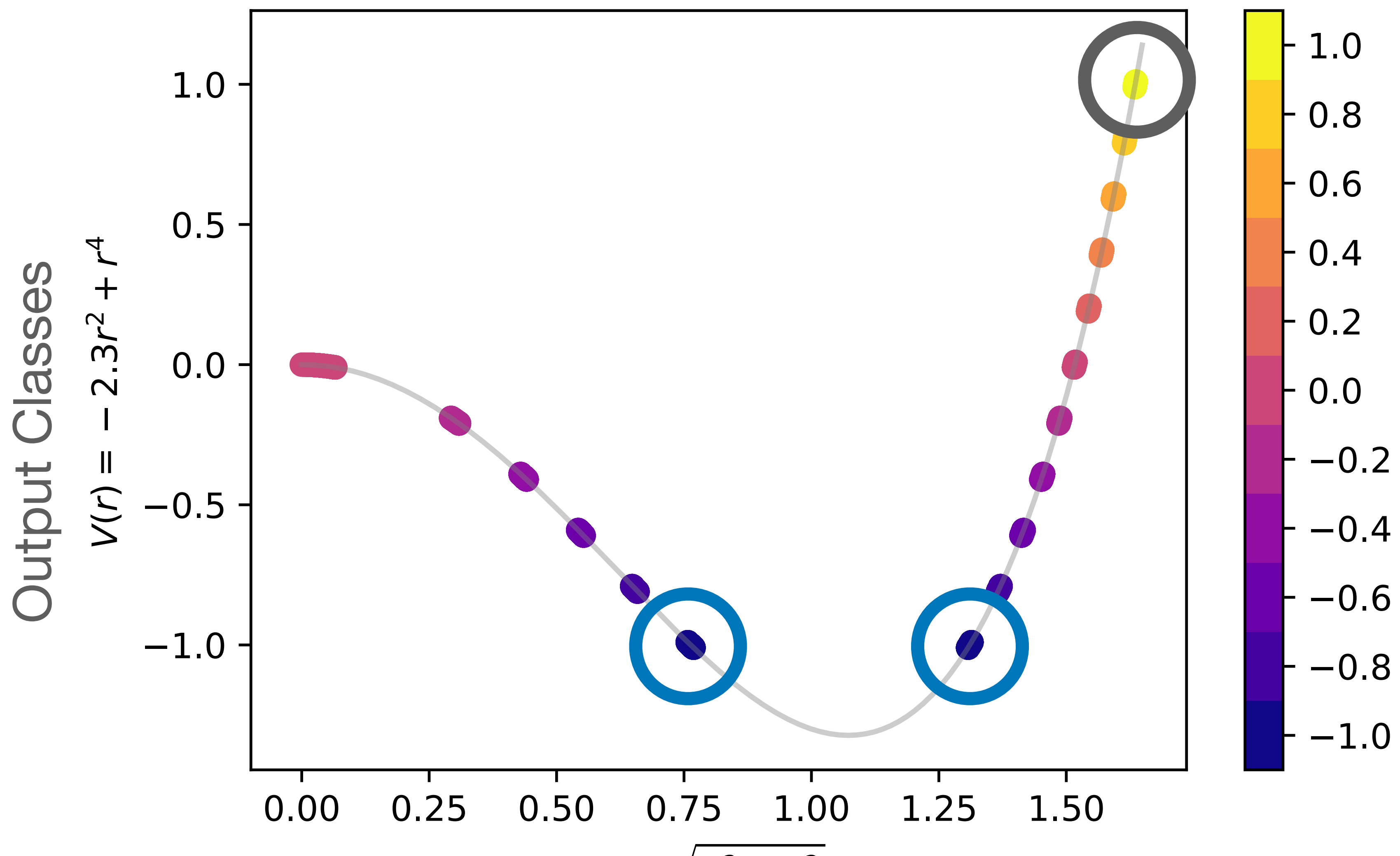
How to search for symmetries?

No direct optimisation available: embedding in deep layer

Feed-forward network



We need: group input with the same meaning together

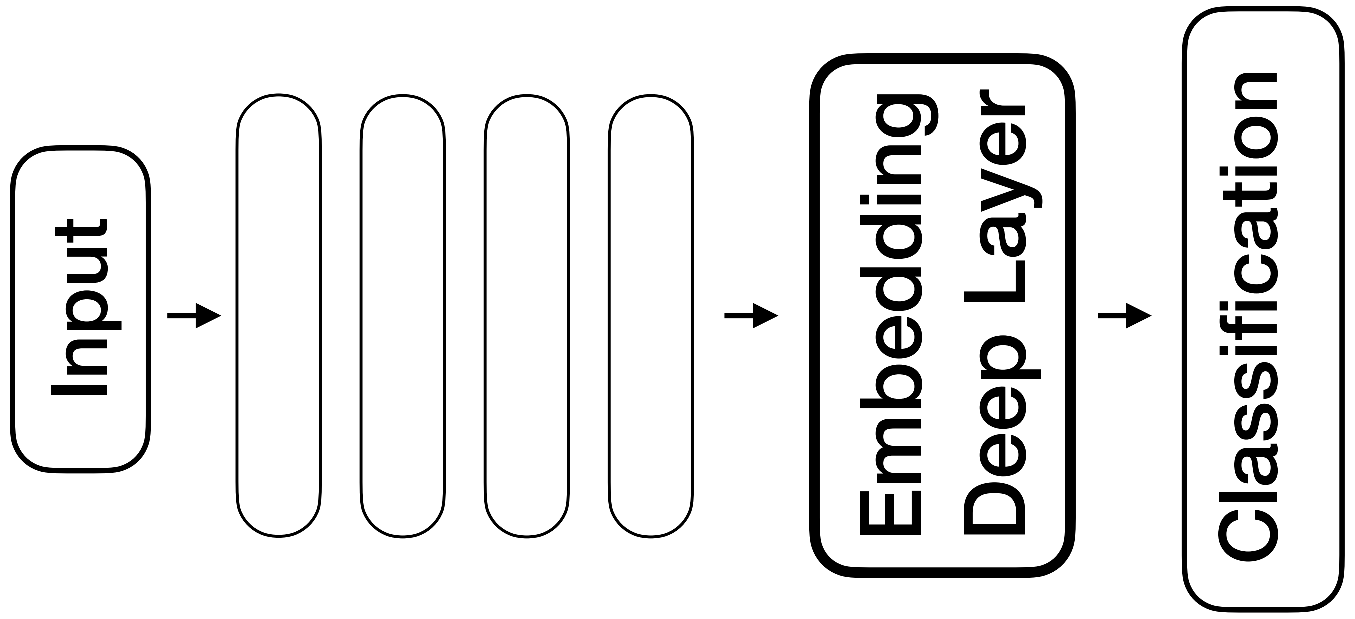


Krippendorf, Syvaeri 2020

How to search for symmetries?

No direct optimisation available: embedding in deep layer

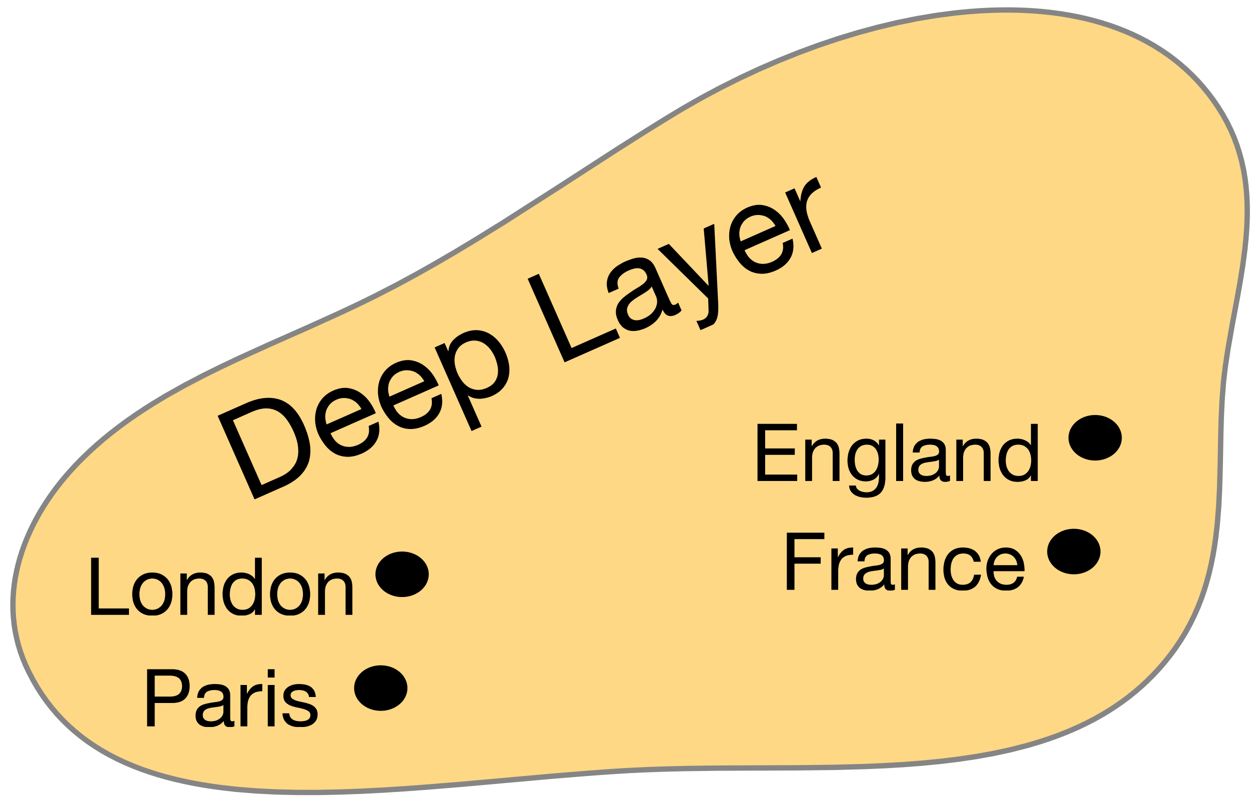
Feed-forward network

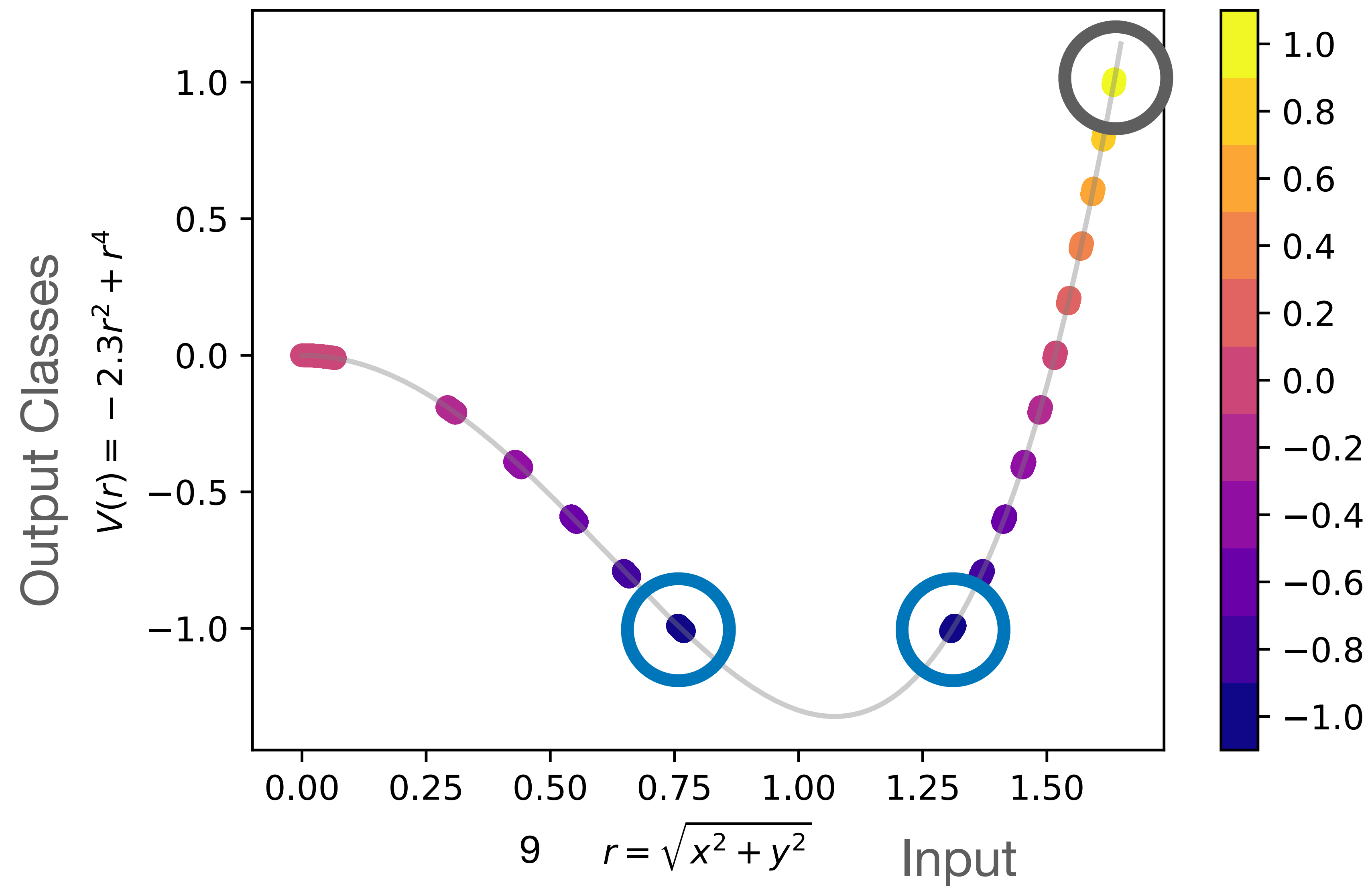


We need: group input with the same meaning together

Word2Vec does it:
(England - London = Paris - France)

[1301.3781, used for re-discovering periodic table 1807.05617, classifying scents of molecules 1910.10685]



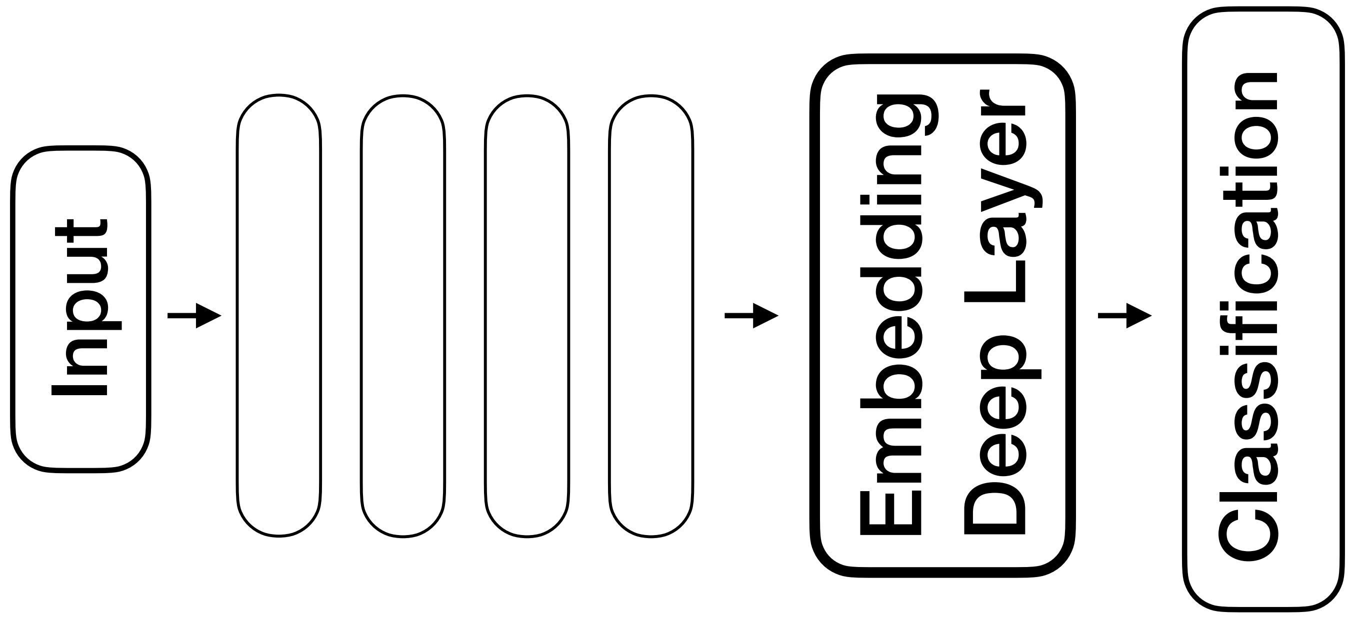


Krippendorf, Syvaeri 2020

How to search for symmetries?

No direct optimisation available: embedding in deep layer

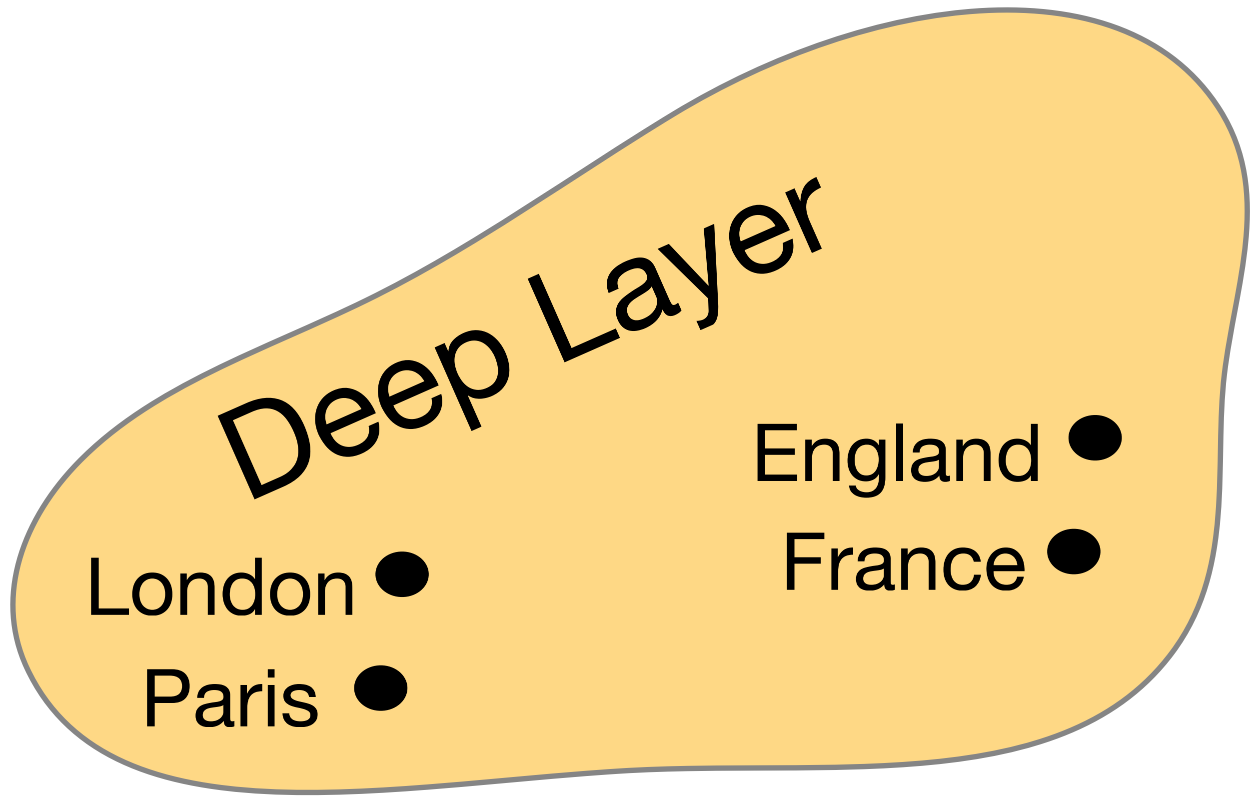
Feed-forward network



We need: group input with the same meaning together

Word2Vec does it:
(England - London = Paris - France)

[1301.3781, used for re-discovering periodic table 1807.05617, classifying scents of molecules 1910.10685]

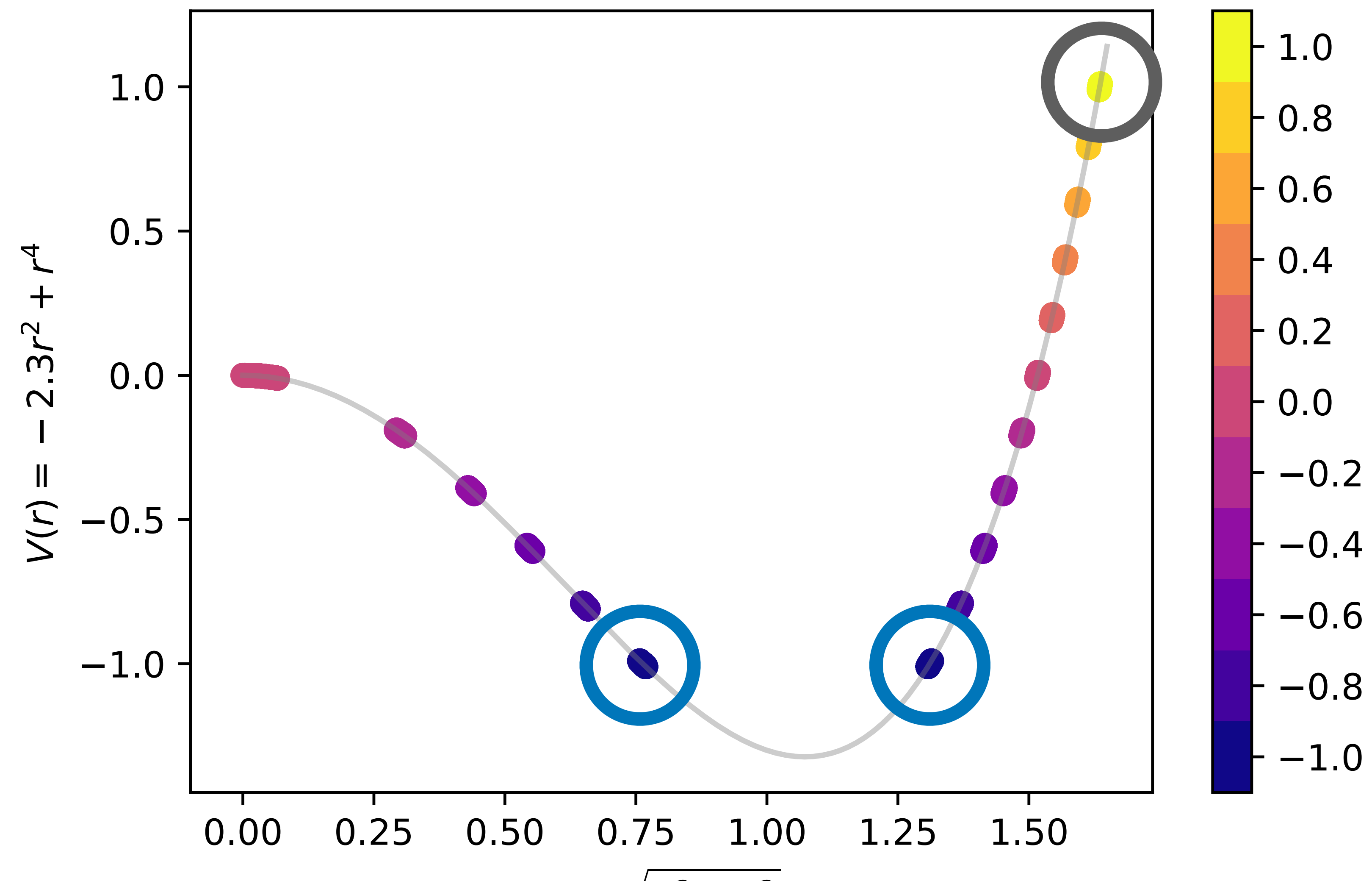


Can we search for symmetries in this way?

Yes!

Examples: SO(2), SU(2), discrete symmetries (CICY)

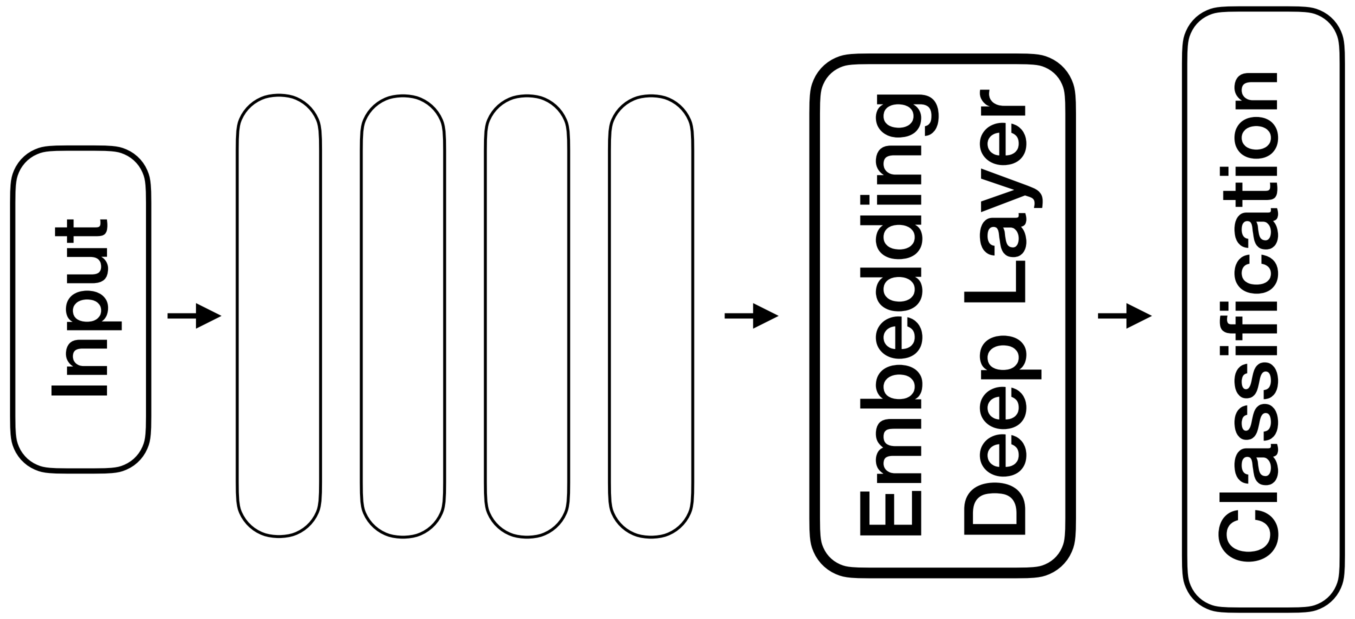
Krippendorf, Syvaeri 2020



How to search for symmetries?

No direct optimisation available: embedding in deep layer

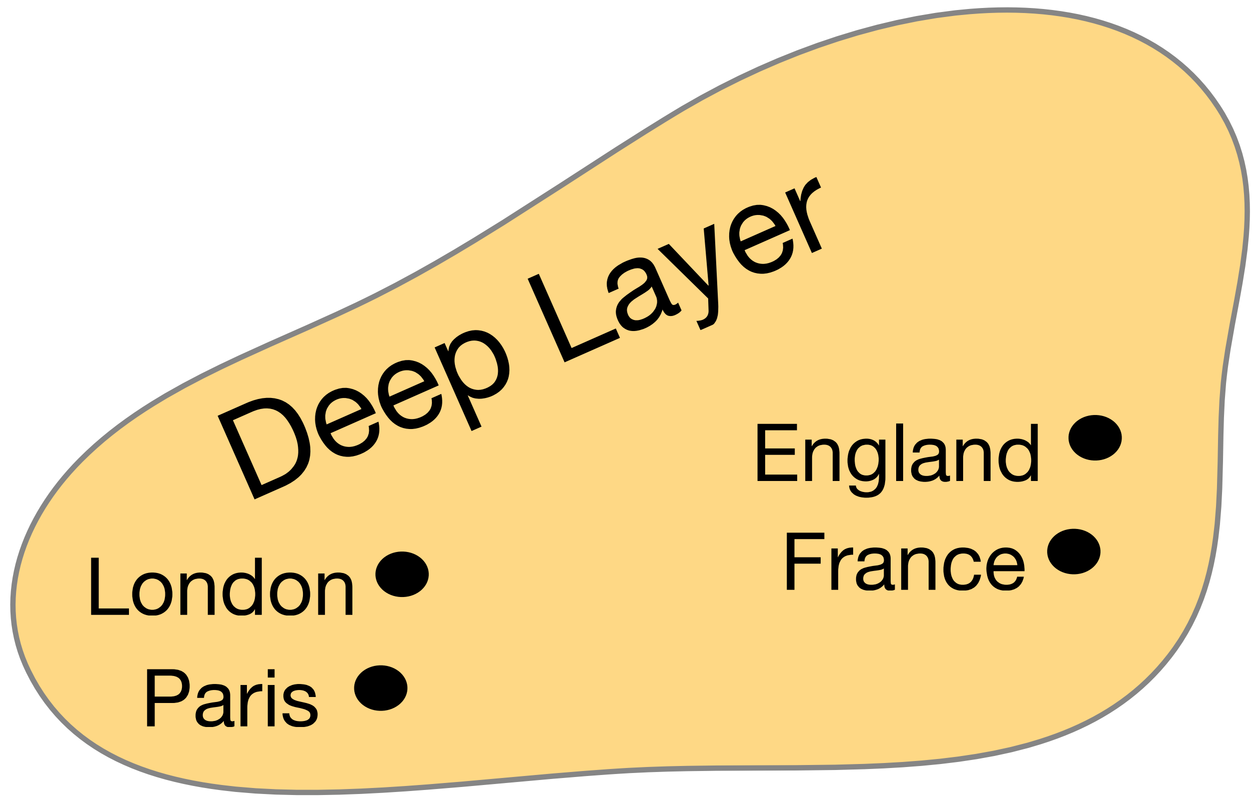
Feed-forward network



We need: group input with the same meaning together

Word2Vec does it:
(England - London = Paris - France)

[1301.3781, used for re-discovering periodic table 1807.05617, classifying scents of molecules 1910.10685]

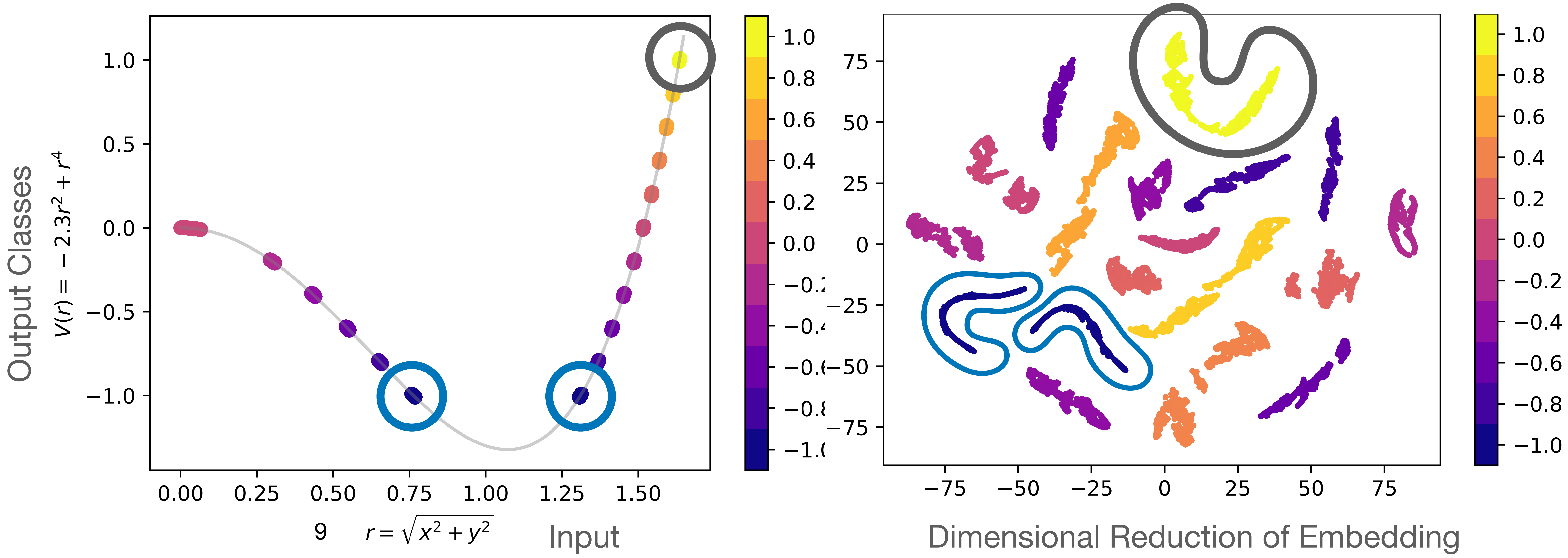


Can we search for symmetries in this way?

Yes!

Examples: SO(2), SU(2), discrete symmetries (CICY)

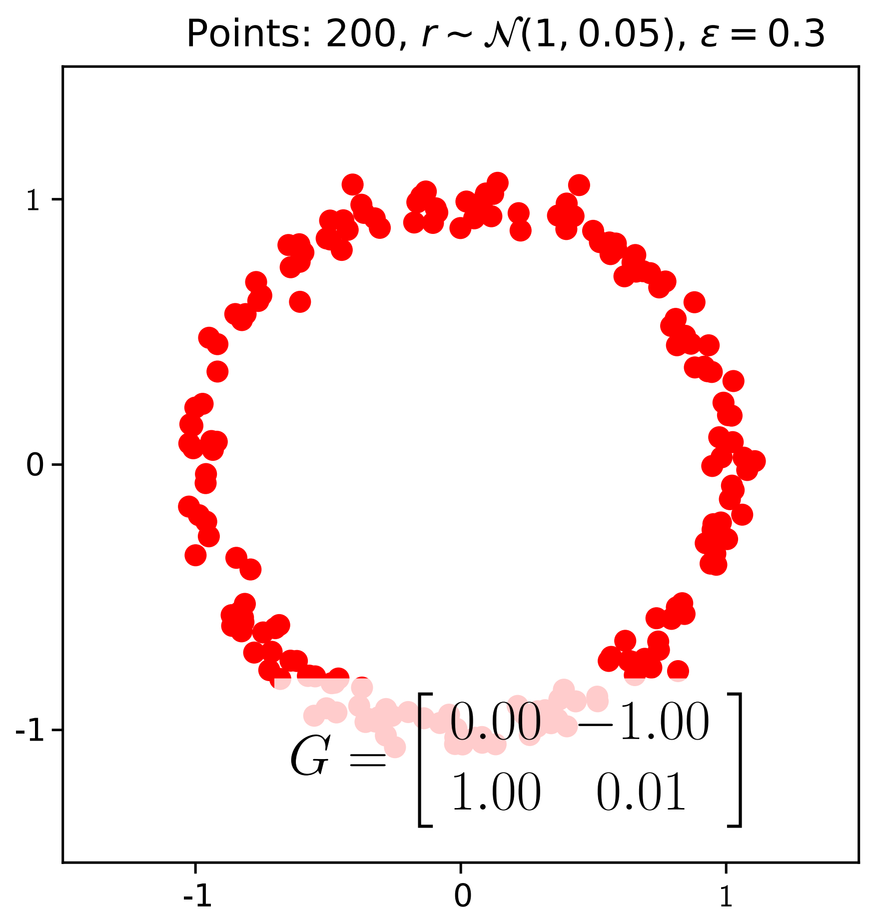
Krippendorf, Syvaeri 2020



How to determine the symmetry?

Connected points in input space:

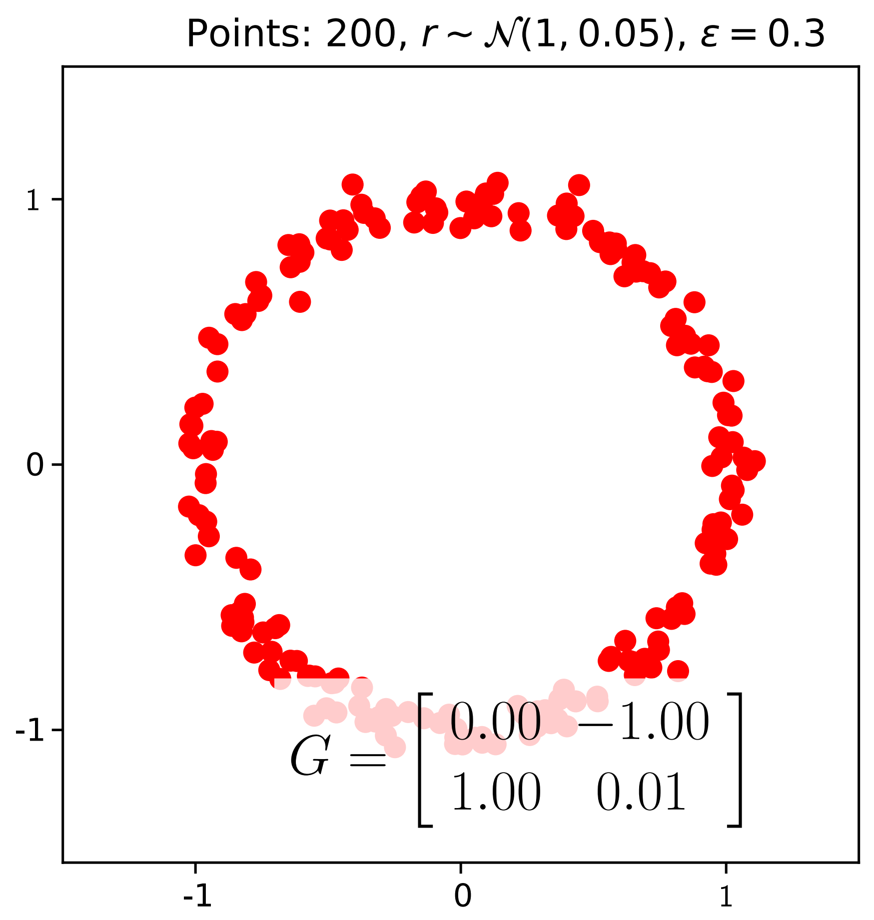
Which symmetry?



Krippendorf, Syvaeri 2020

How to determine the symmetry?

Connected points in input space:



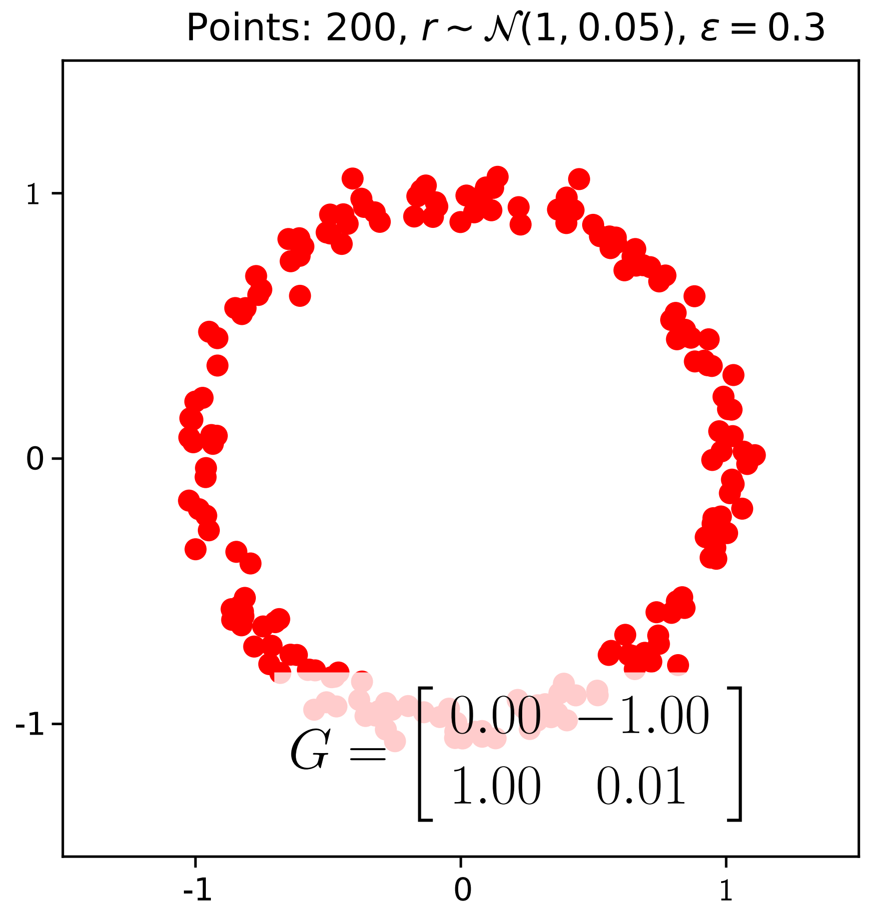
Which symmetry?

Determine generator connecting points in (sub)-space:

Krippendorf, Syvaeri 2020

How to determine the symmetry?

Connected points in input space:



Which symmetry?

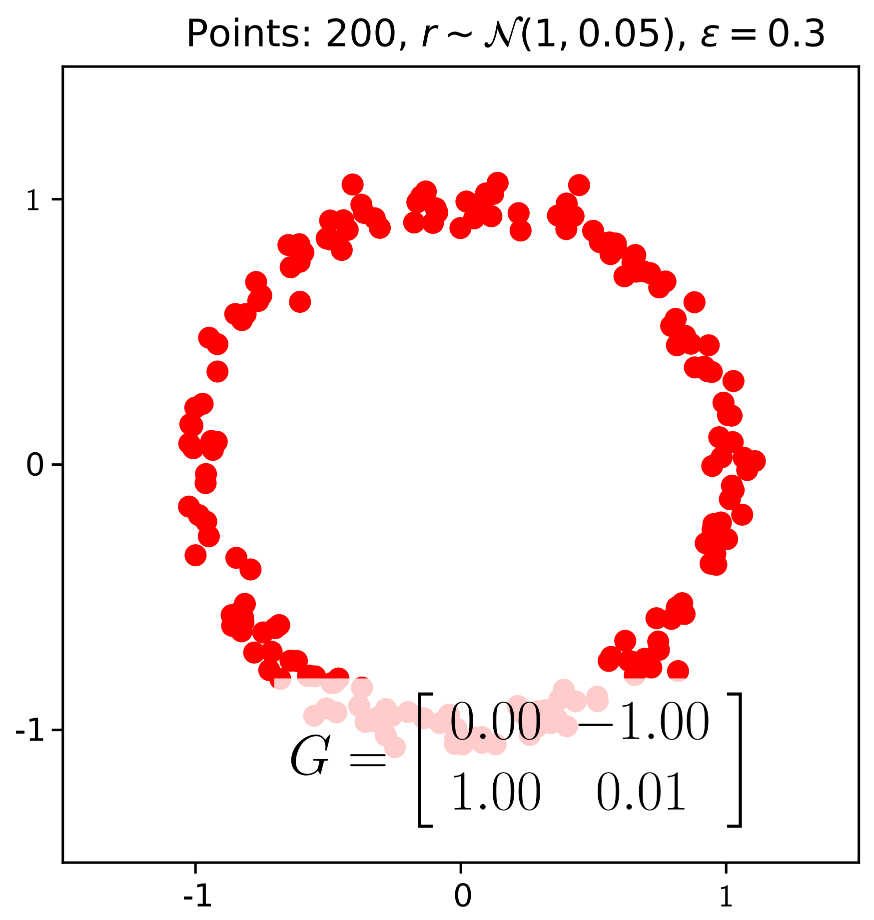
Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

Krippendorf, Syvaeri 2020

How to determine the symmetry?

Connected points in input space:



Which symmetry?

Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

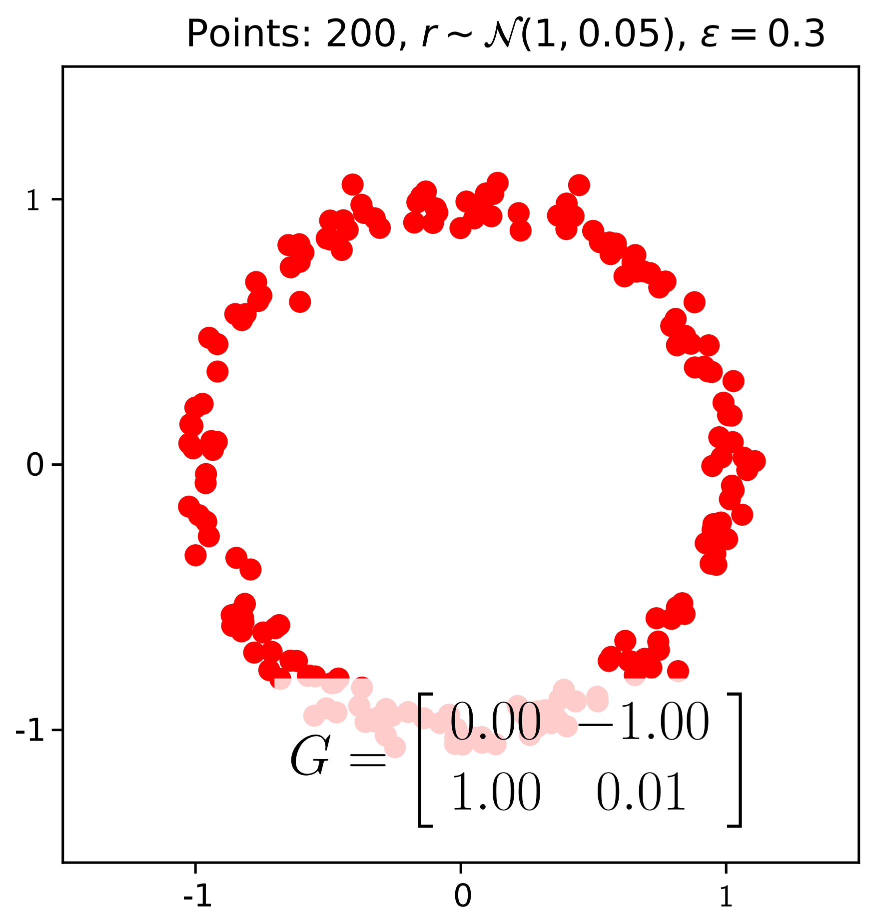
Repeat multiple times (covering all sub-spaces) and perform PCA on generators:

Krippendorf, Syvaeri 2020

How to determine the symmetry?

Connected points in input space:

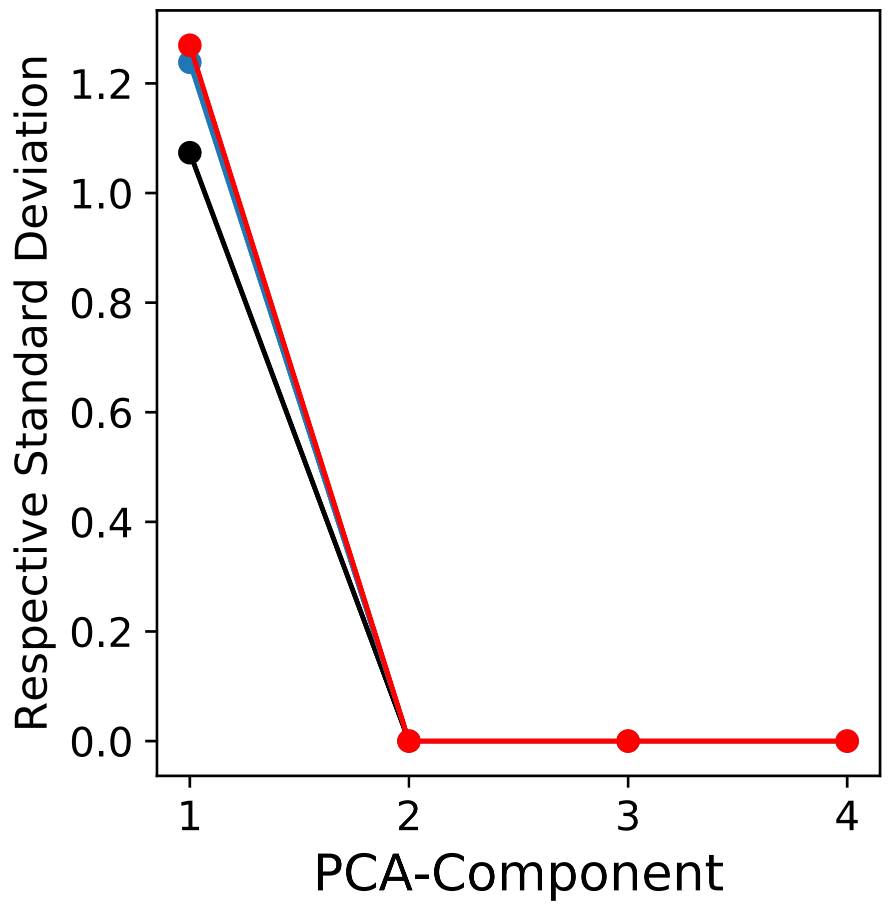
Which symmetry?



Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

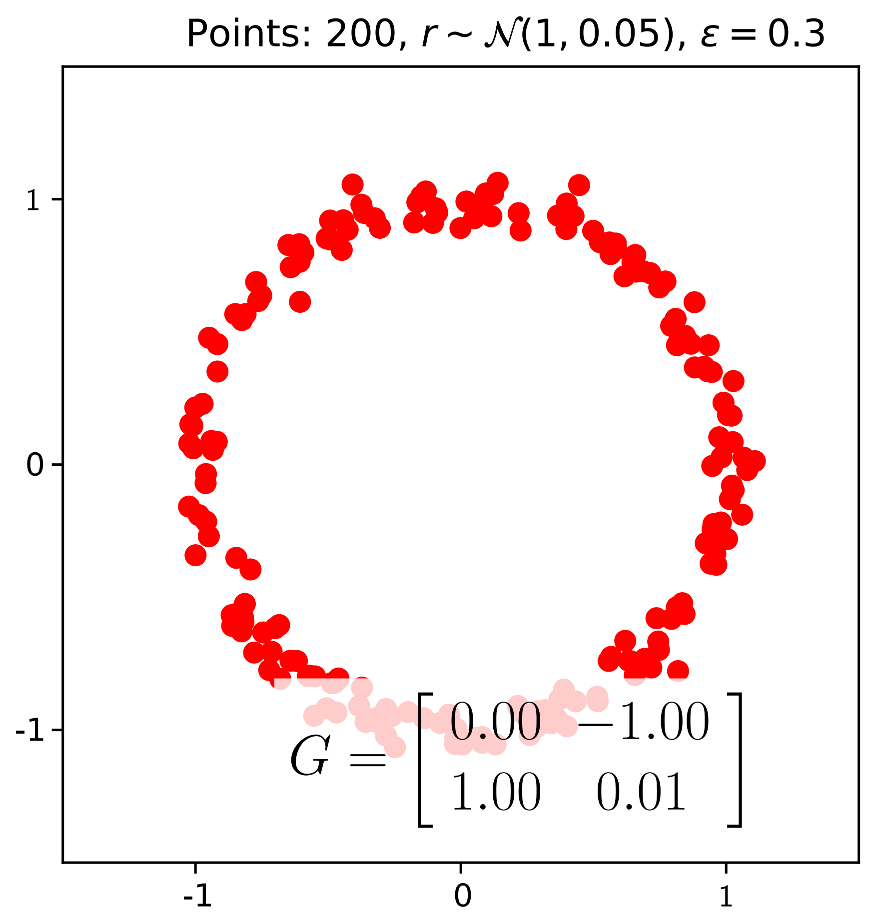
Repeat multiple times (covering all sub-spaces) and perform PCA on generators:



Krippendorf, Syvaeri 2020

How to determine the symmetry?

Connected points in input space:



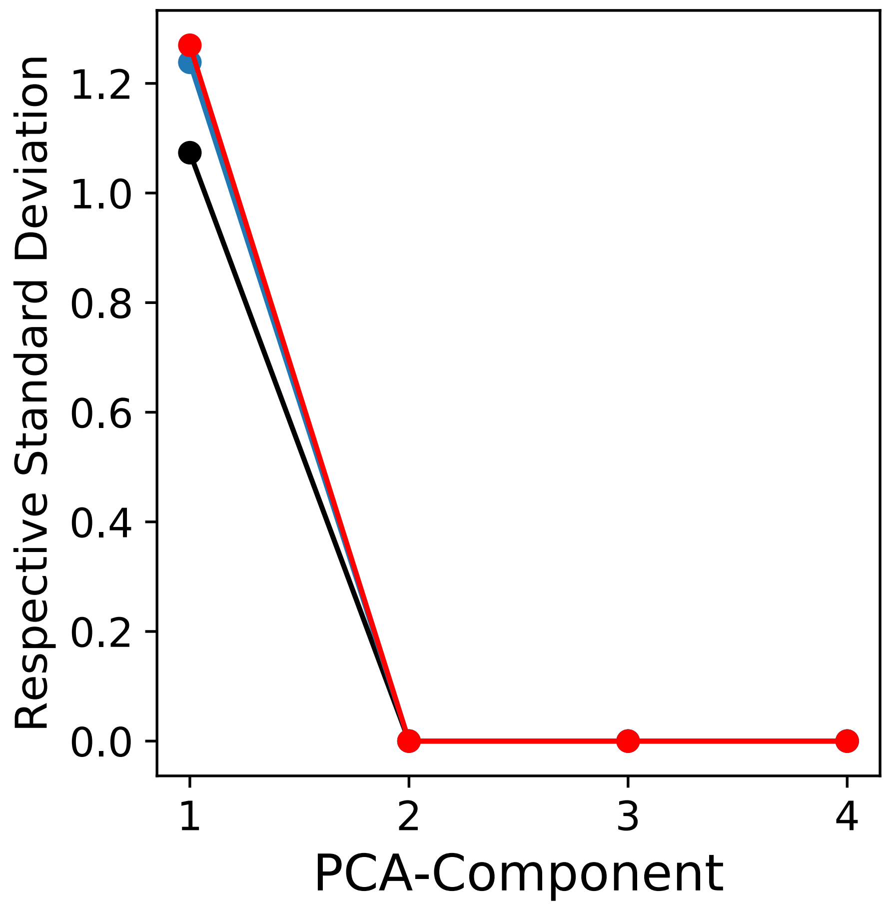
Which symmetry?

Other Examples?

Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

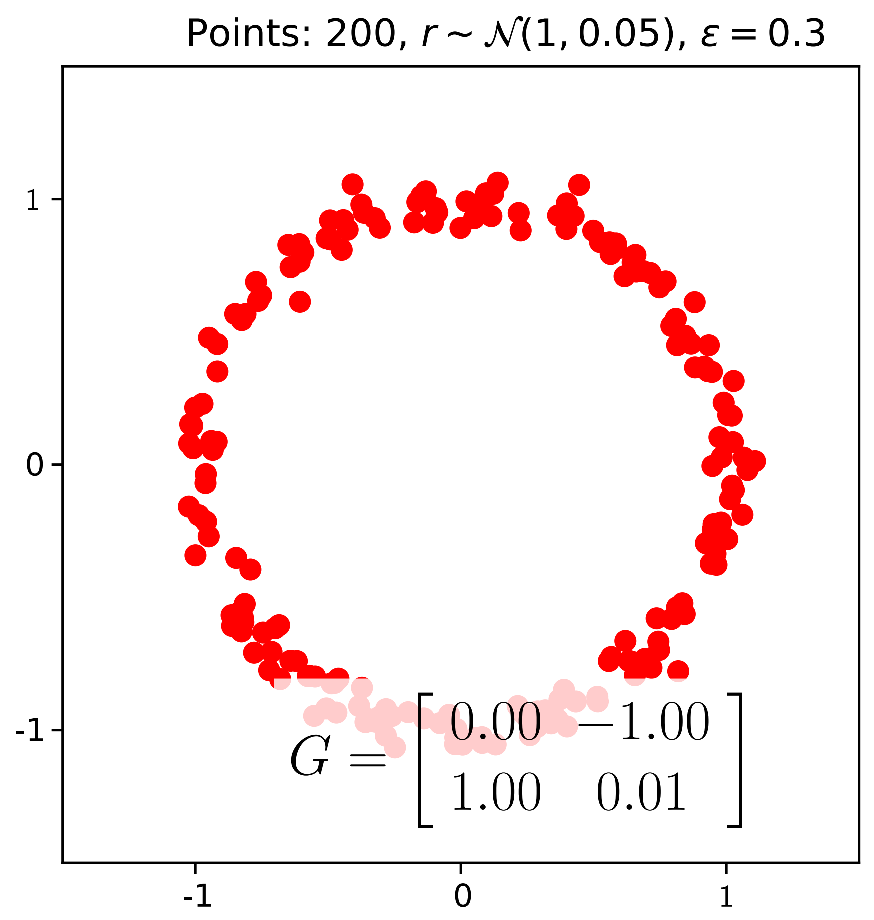
Repeat multiple times (covering all sub-spaces) and perform PCA on generators:



Krippendorf, Syvaeri 2020

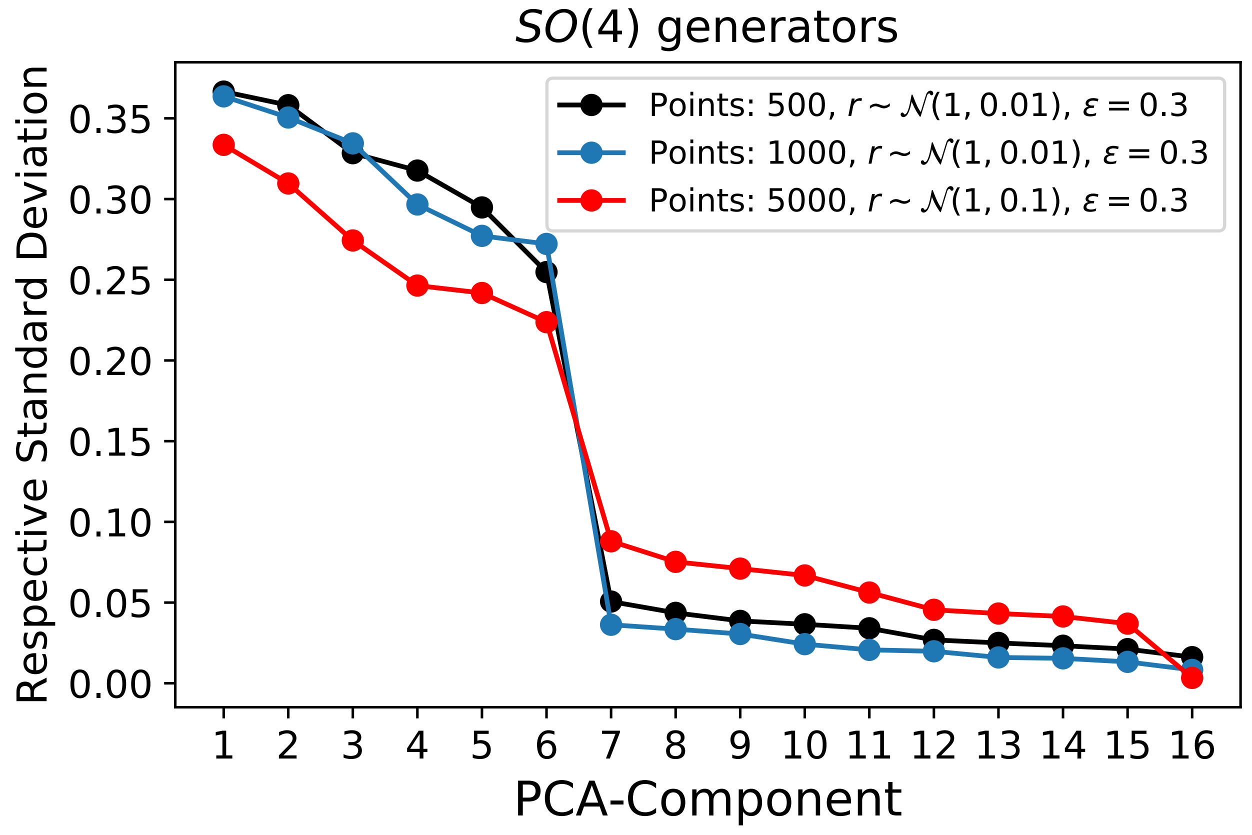
How to determine the symmetry?

Connected points in input space:



Which symmetry?

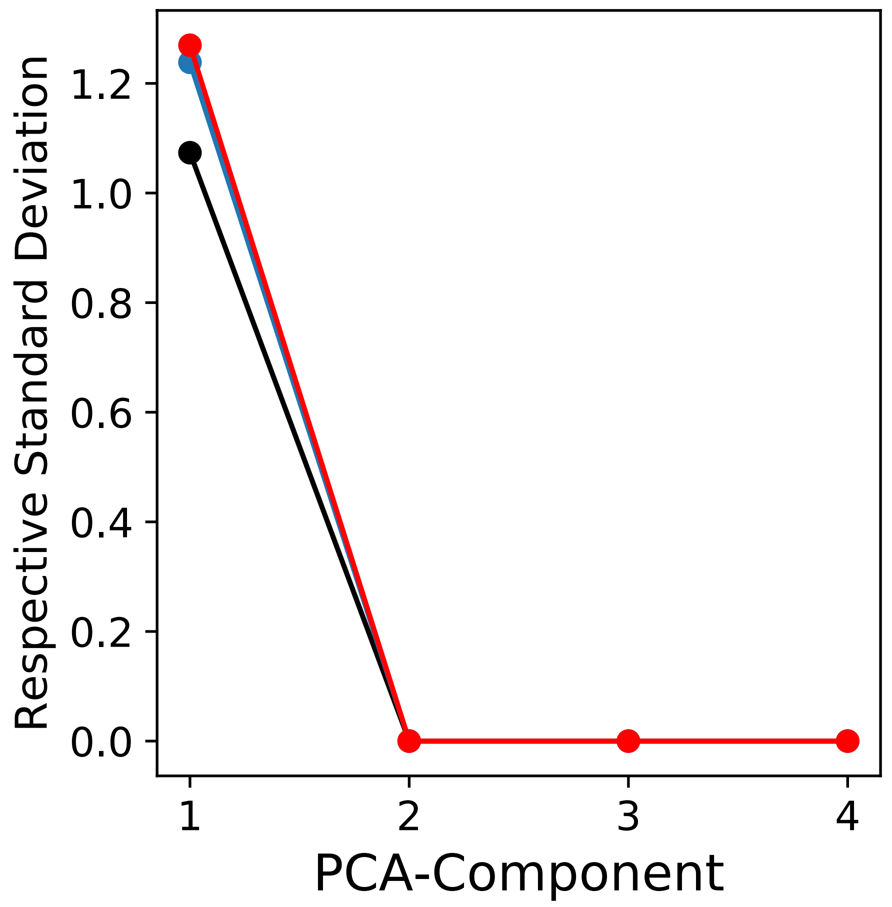
Other Examples?



Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

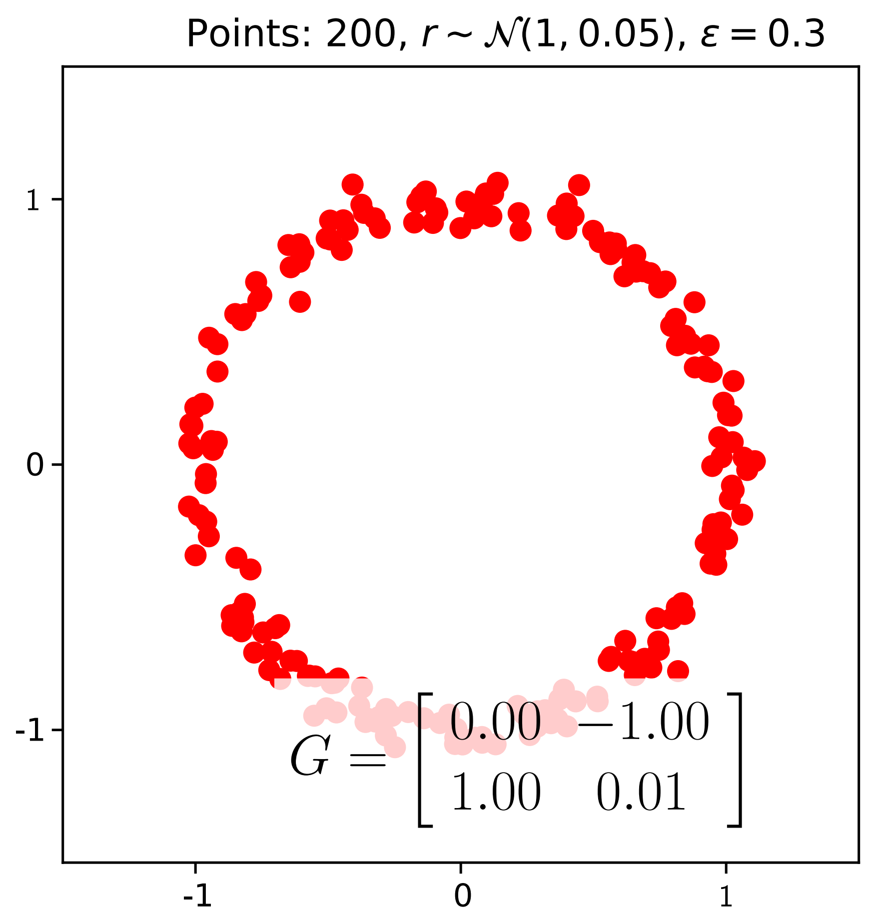
Repeat multiple times (covering all sub-spaces) and perform PCA on generators:



Krippendorf, Syvaeri 2020

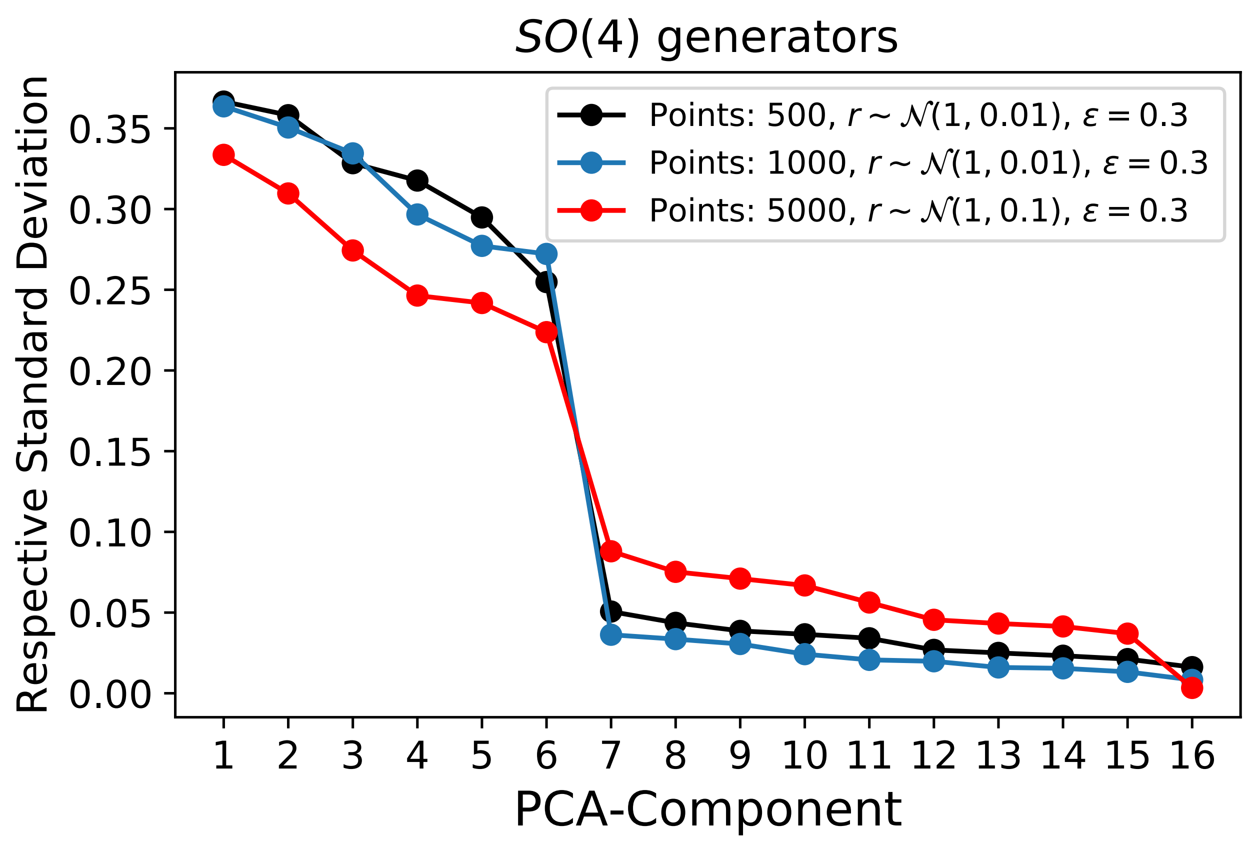
How to determine the symmetry?

Connected points in input space:



Which symmetry?

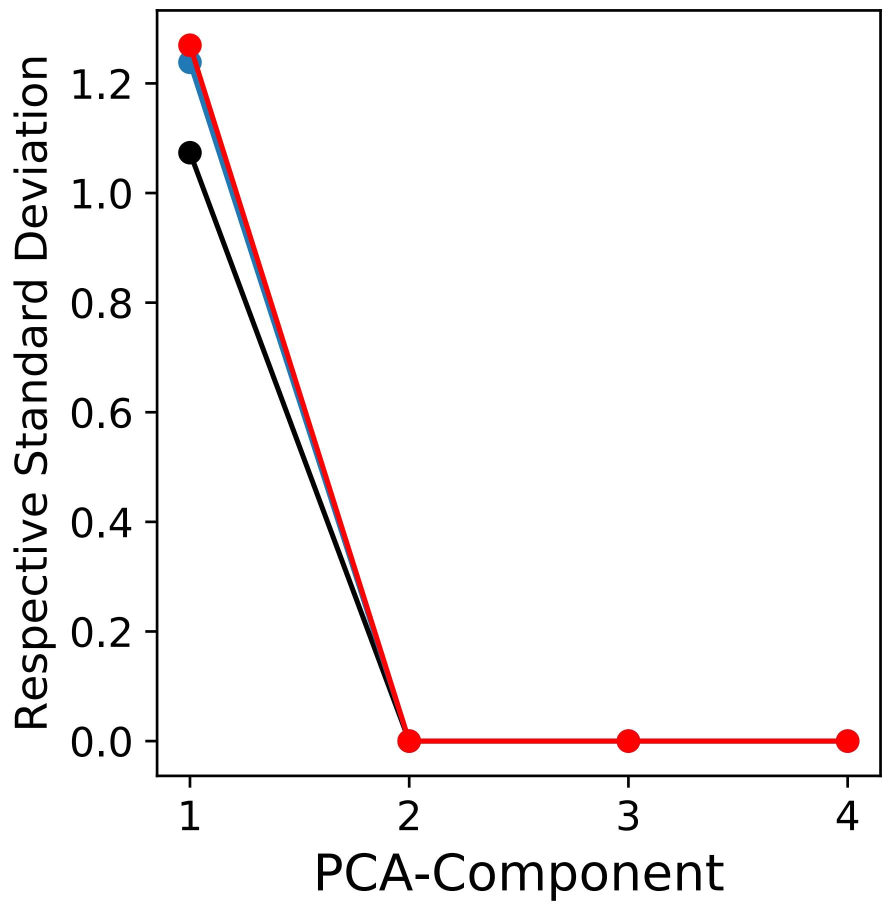
Other Examples?

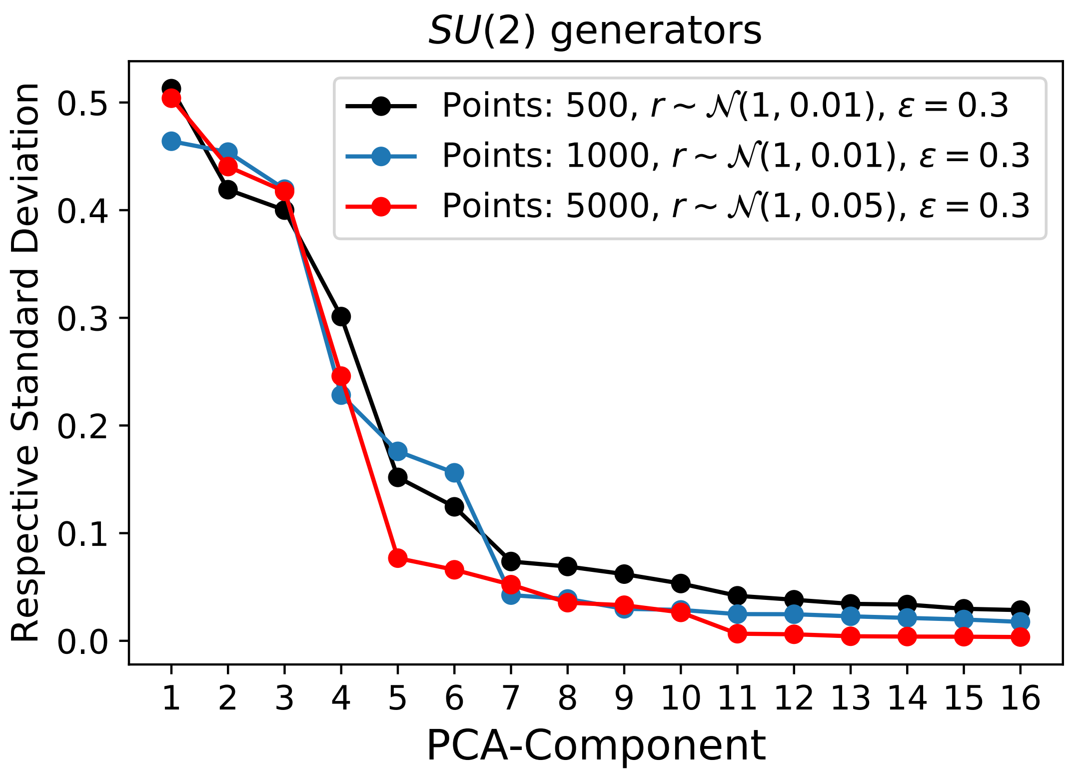


Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

Repeat multiple times (covering all sub-spaces) and perform PCA on generators:

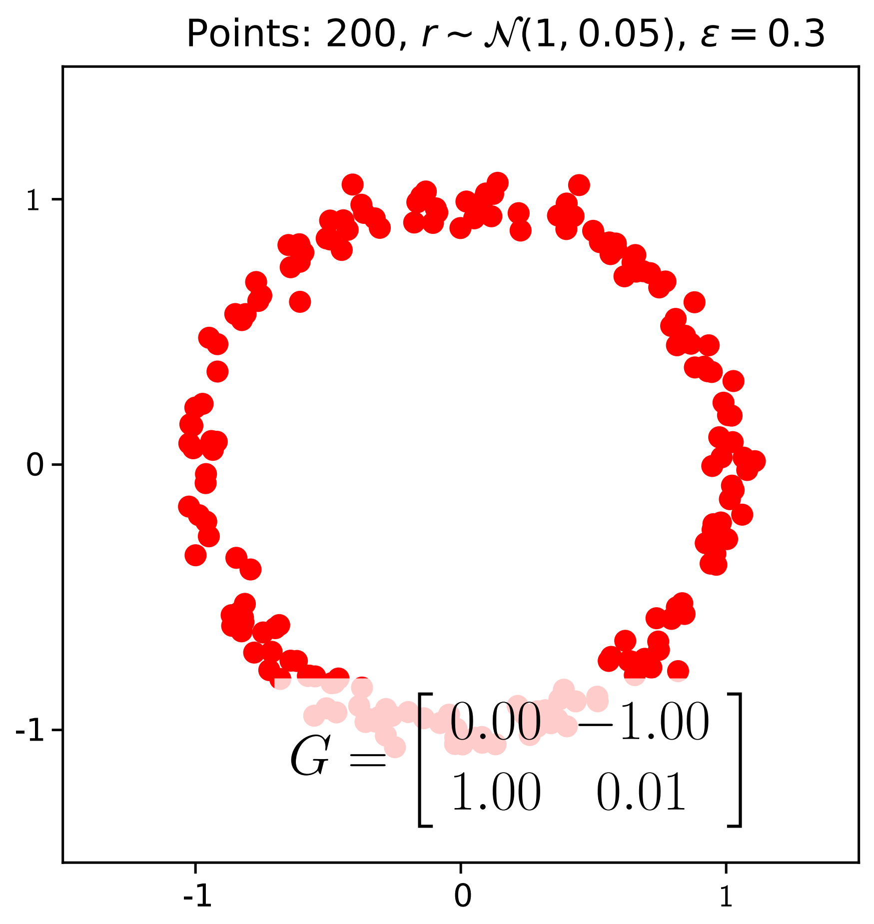




Krippendorf, Syvaeri 2020

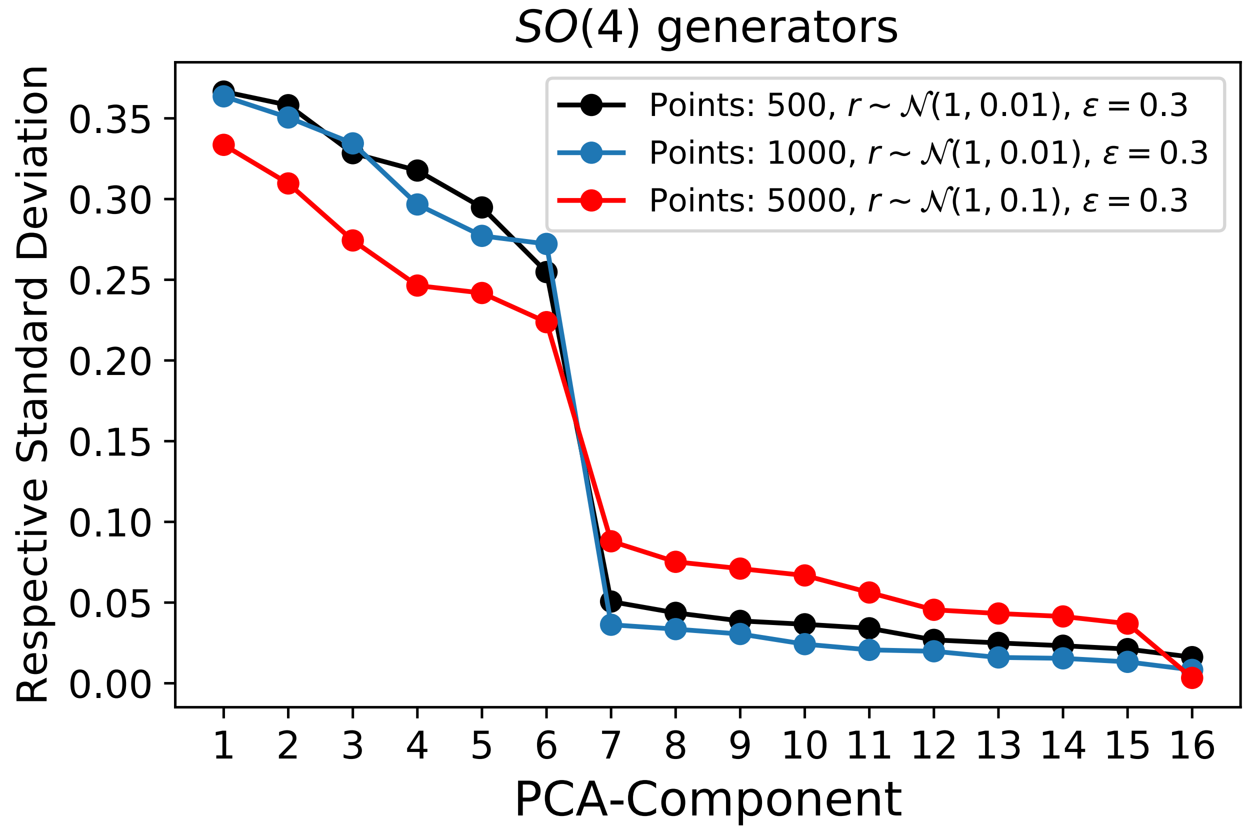
How to determine the symmetry?

Connected points in input space:



Which symmetry?

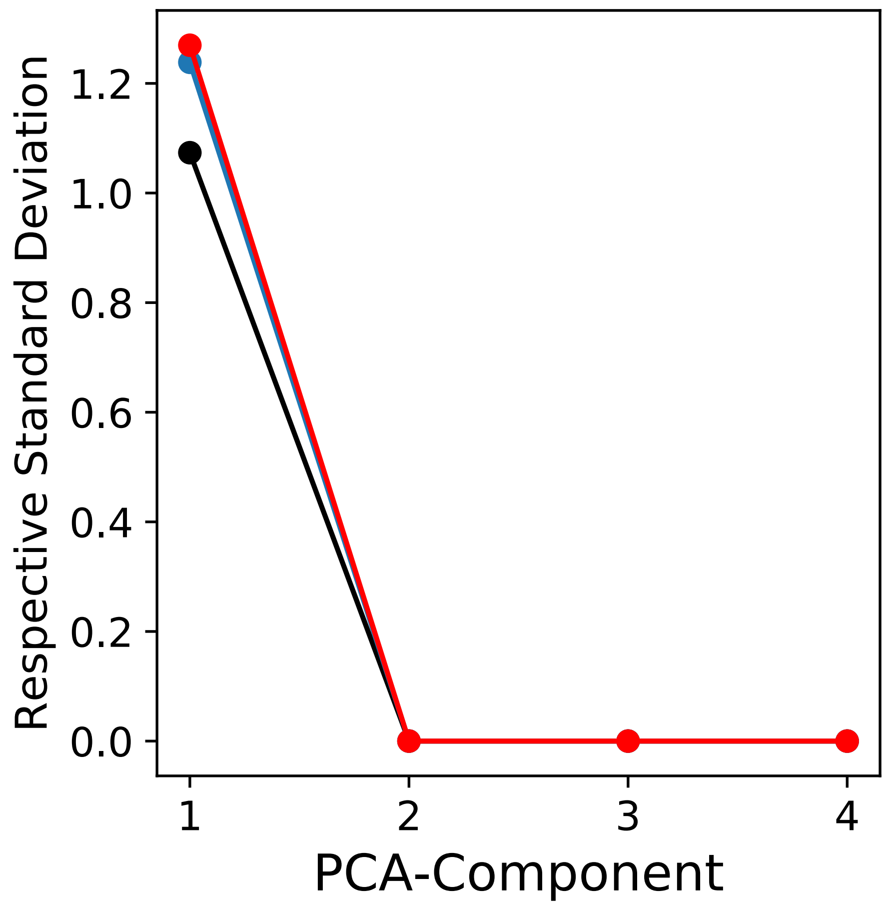
Other Examples?

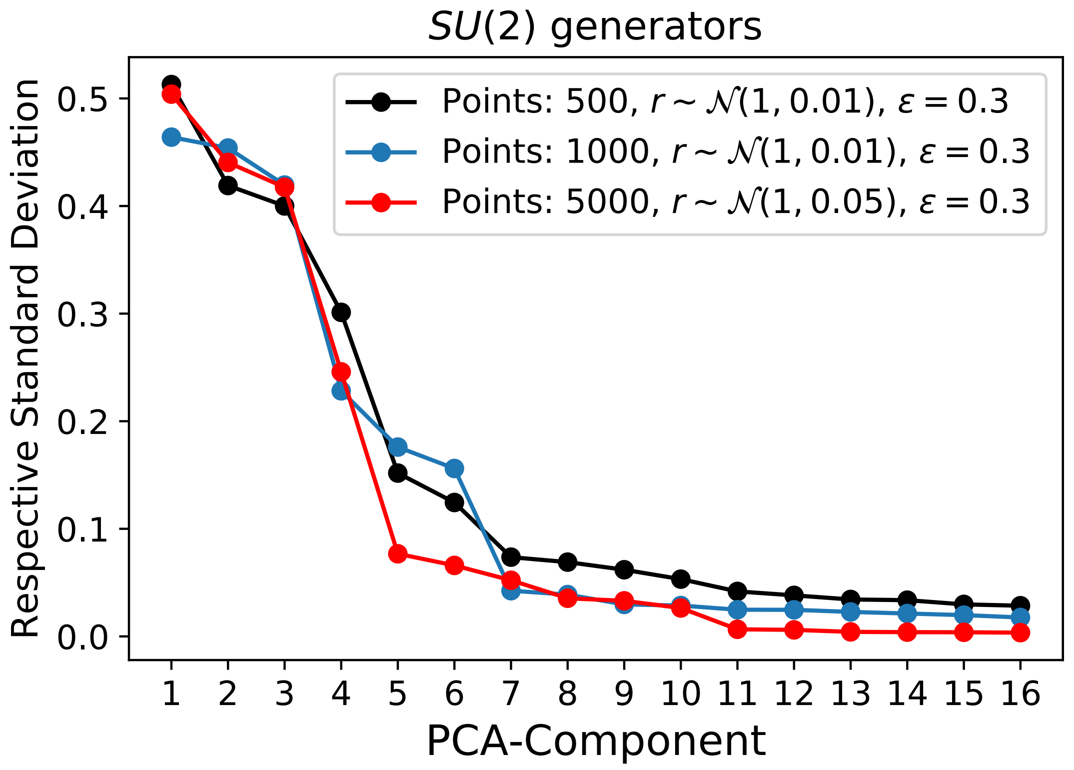


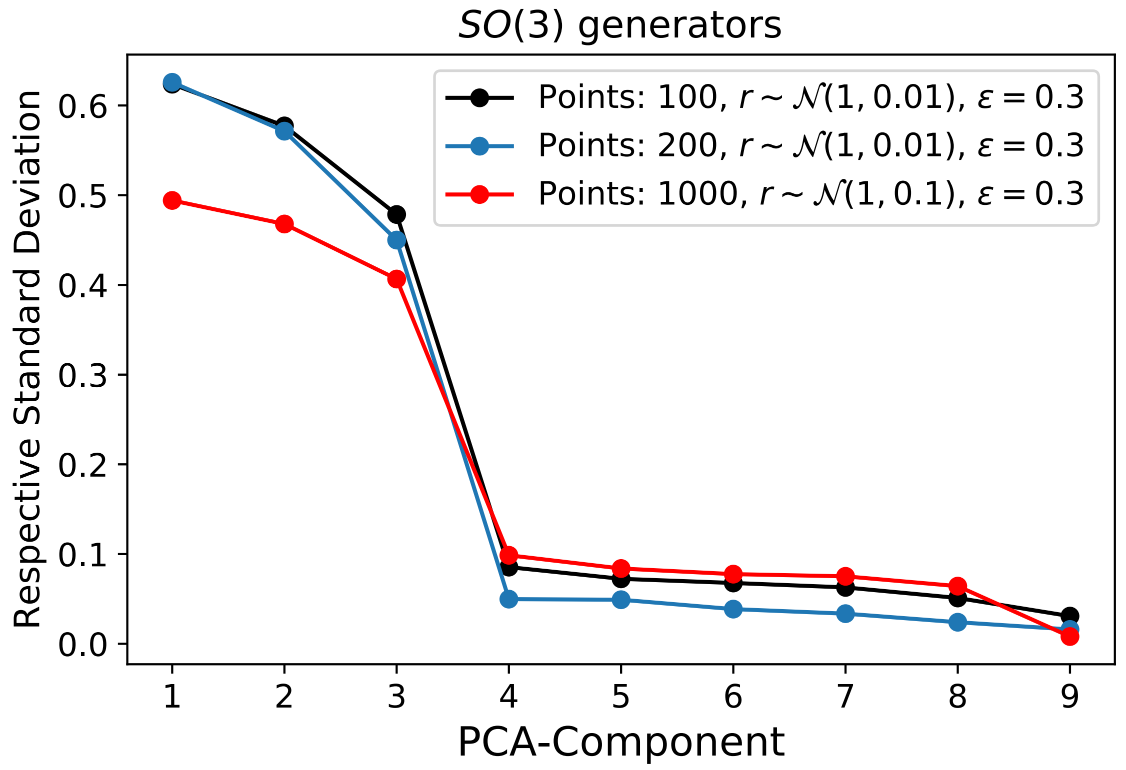
Determine generator connecting points in (sub)-space:

$$p' = p + \epsilon_a T^a p$$

Repeat multiple times (covering all sub-spaces) and perform PCA on generators:







Krippendorf, Syvaeri 2020

Symmetries from data (samples of phase space)

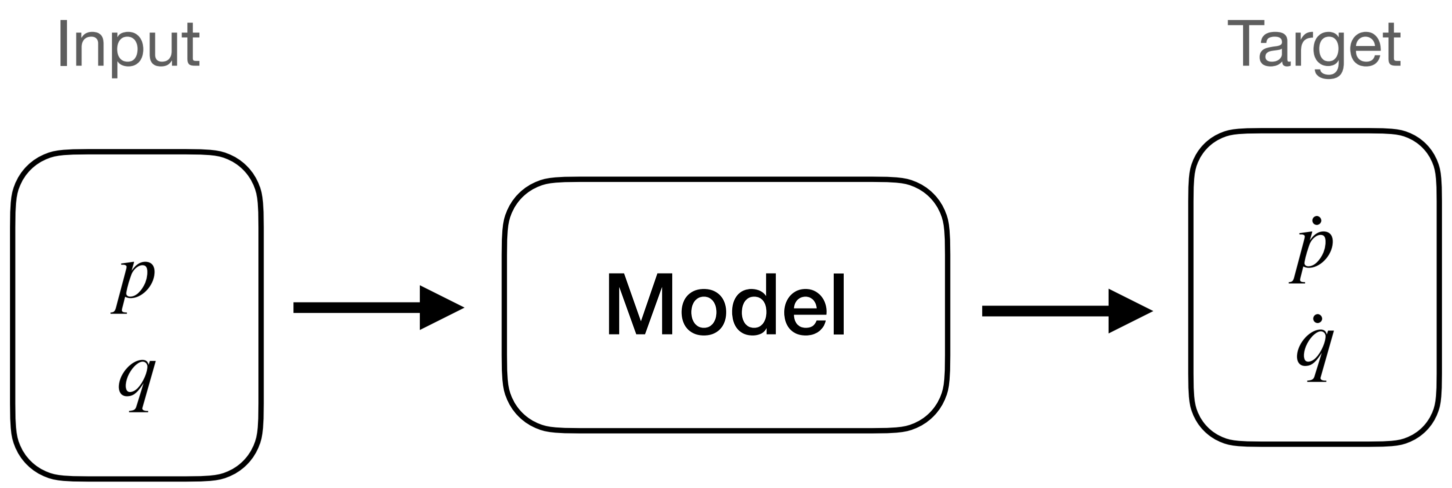
Krippendorf, Syvaeri (ICLR simDL workshop, 2104.14444)

Defining optimisation

Predicting trajectories

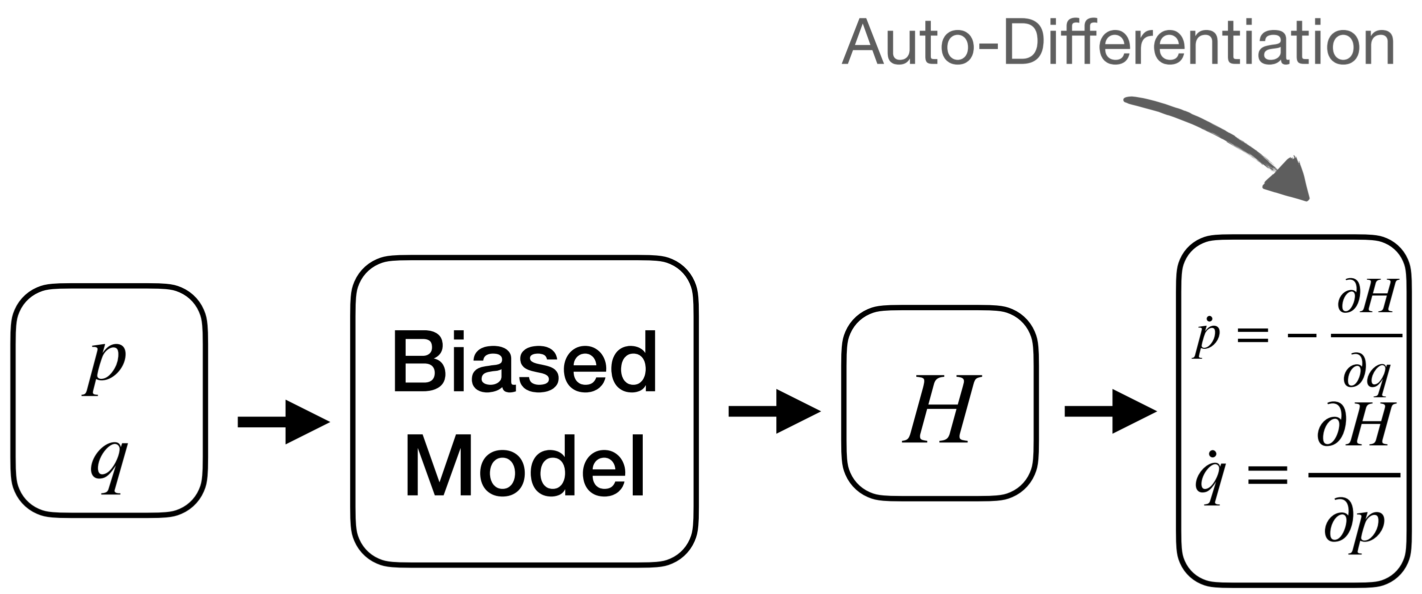
Which problem are we interested in?

- How can we define an energy function whose minima result in appropriate models?
- From current particle position and momentum predict the next time step/change of position and momentum

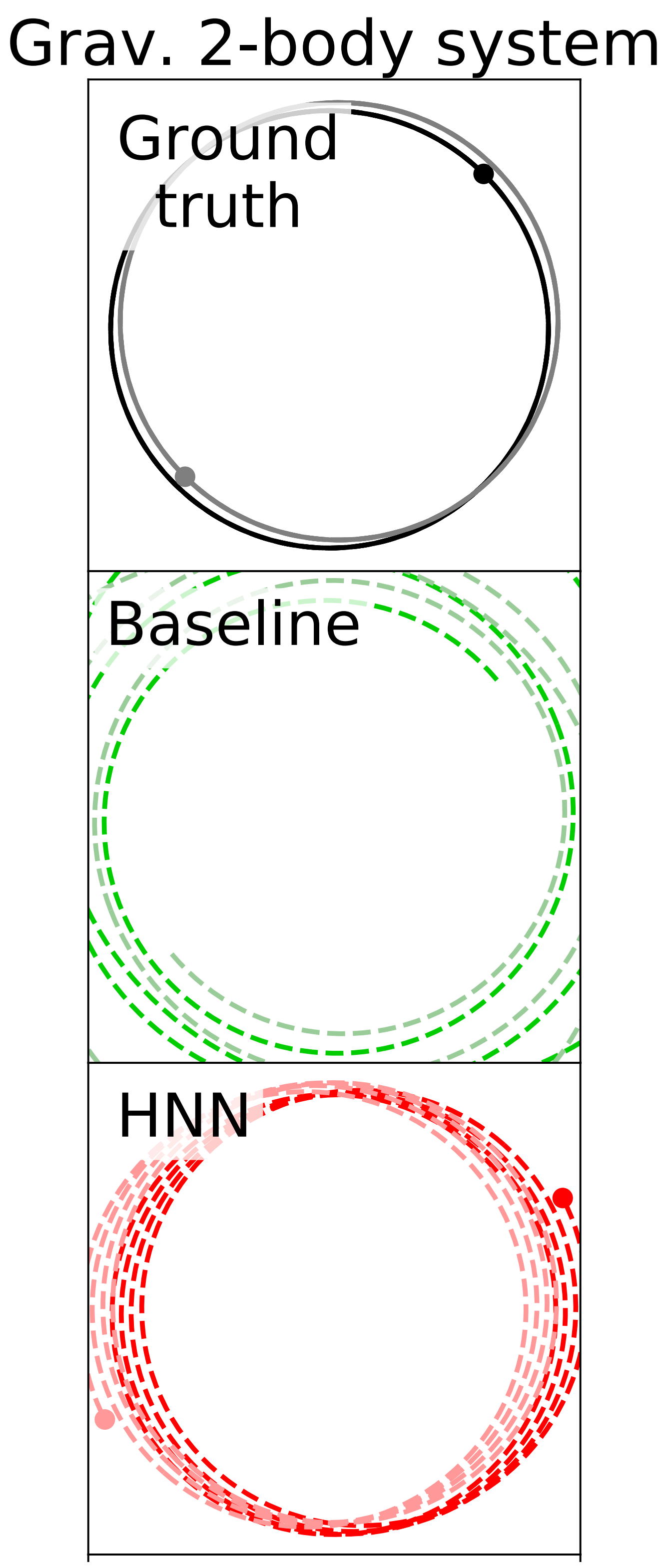


- Option 1: predict directly $\dot{p},\ \dot{q}$
- Option 2 (domain knowledge): predict Hamiltonian and use auto-differentiation for $\dot{p},\ \dot{q}$

- Predicting Hamiltonian ensures physics bias of energy conservation



- Note: this optimisation problem is predicting a Hamiltonian without knowing the Hamiltonian in advance.



Greydanus, Dzamba, Yosinski 1906.01563

Can we learn more structures from samples of phase space?

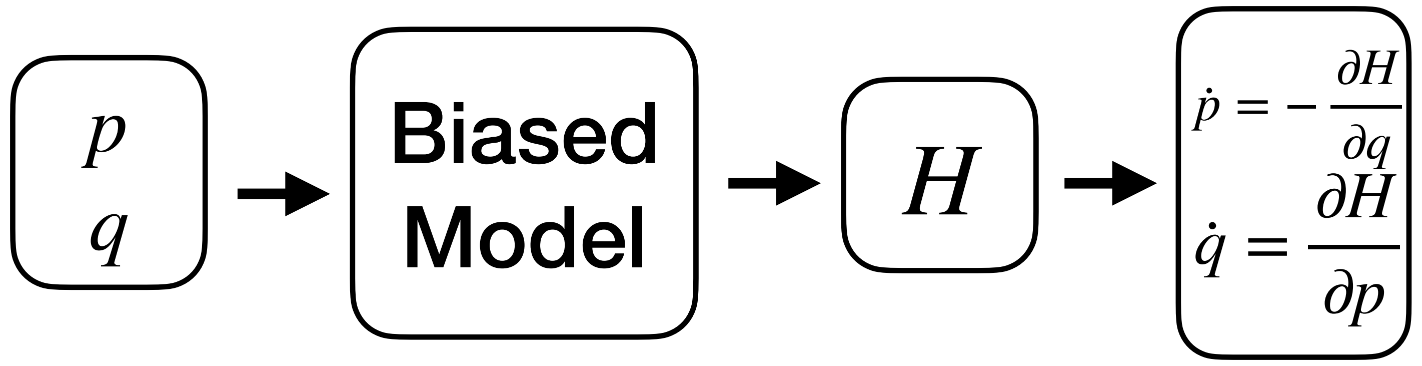
More structures from neural networks?

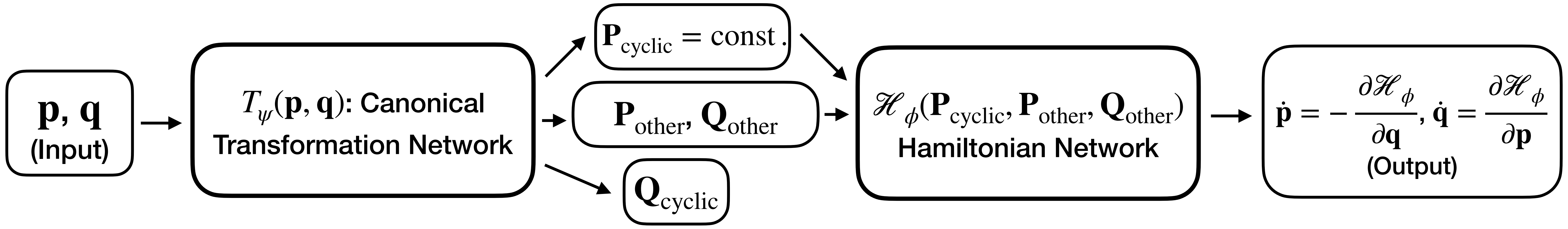
- If we can train NNs to find the Hamiltonian of a system, can we use it to learn other interesting structures?
- Symmetries of the system? E.g. via canonical transformations (cyclic coordinates reveal conserved quantities)
- How does this work? 2 key steps:
    1. Formulate your physics search problem as an optimisation problem.
    2. Make sure it's learnable for your architecture.
- Good news for analytic understanding of numerical approximations: most physics functions are simple
- Interesting **side effect**: quantify how much these structures help in predicting dynamics

AI for Simulations — Symmetries

Introducing physicists’ bias

SCNNs: We cannot only learn the Hamiltonian but also the symmetries by enforcing canonical coordinates

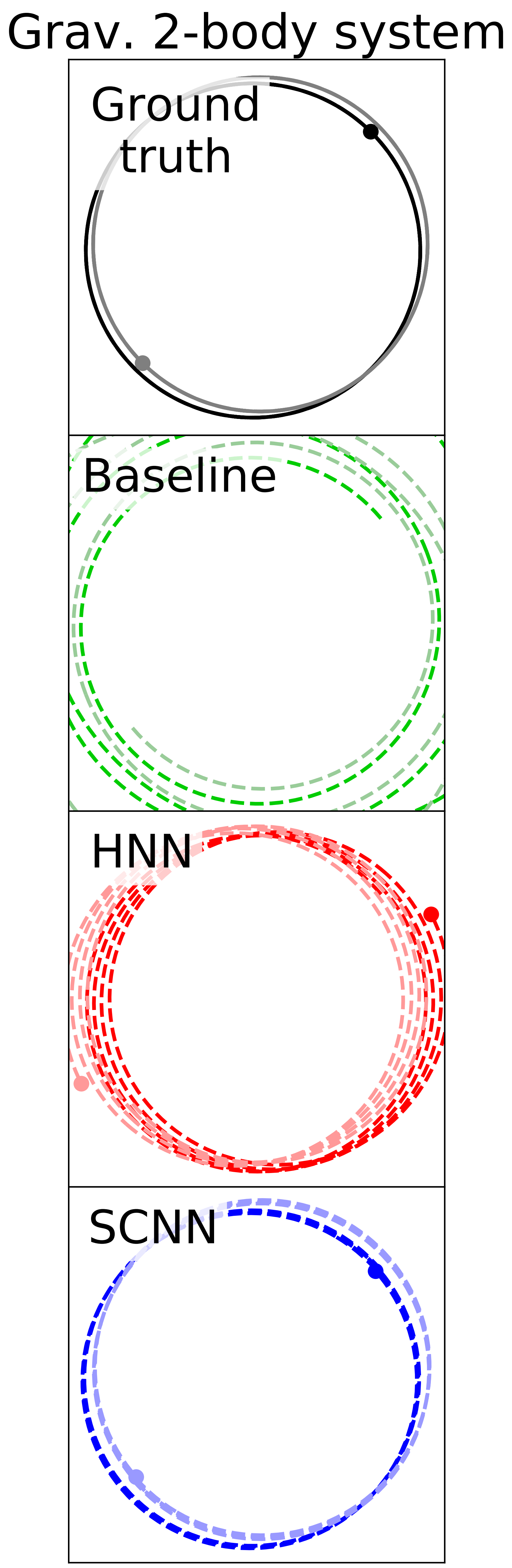




Modified Losses:

$$0 = \dot{F}_k(p, q) = \{H(p, q), F_k(p, q)\}$$

Additional constraint on motion (not just energy conservation), i.e. motion takes place on hyper-surface in phase space

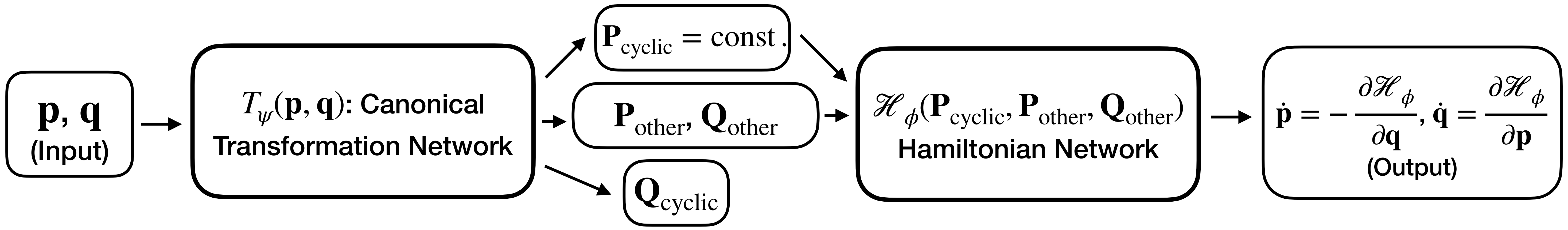


Krippendorf, Syvaeri 2104.14444

AI for Simulations — Symmetries

Introducing physicists' bias

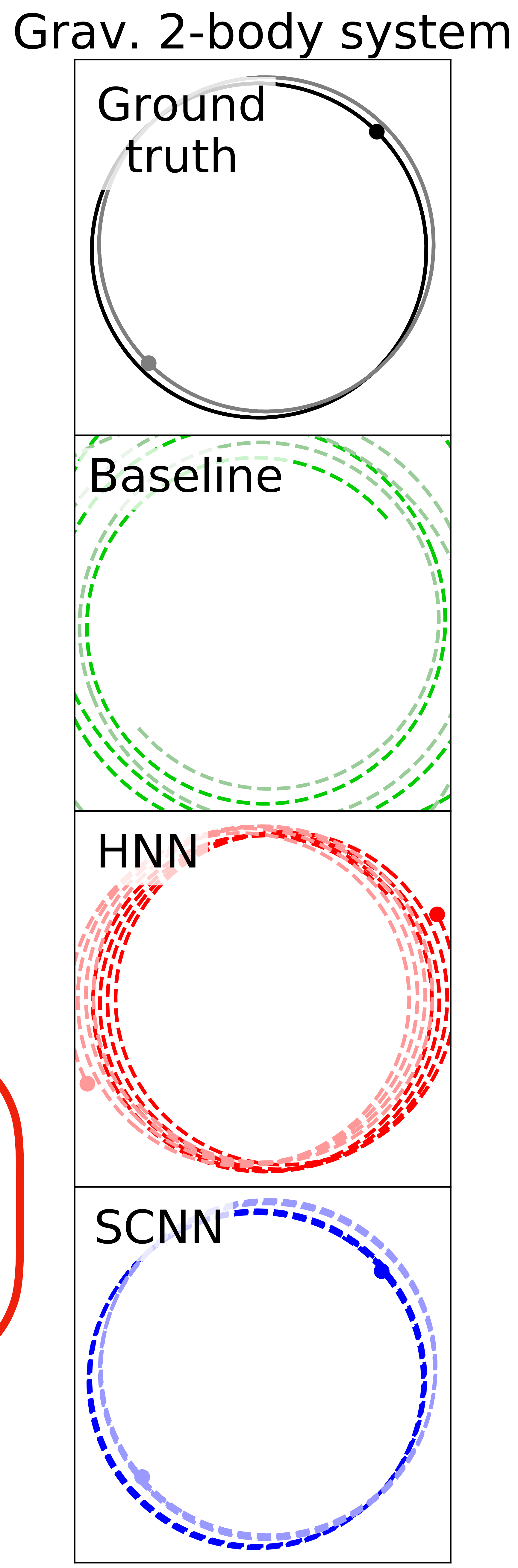
SCNNs: We cannot only learn the Hamiltonian but also the symmetries by enforcing canonical coordinates



Modified Losses for canonical coordinates:

- Hamilton equations: $\dot{P}_i(p,q) = -\dfrac{\partial H(p,q)}{\partial Q_i(p,q)} = 0$ and $\dot{Q}_i(p,q) = \dfrac{\partial H(p,q)}{\partial P_i(p,q)}$
- Poisson algebra: $\{P_i, Q_j\} = \delta_{ij}$ and $\{P_i, P_j\} = \{Q_i, Q_j\} = 0$

Additional Loss terms



Krippendorf, Syvaeri 2104.14444

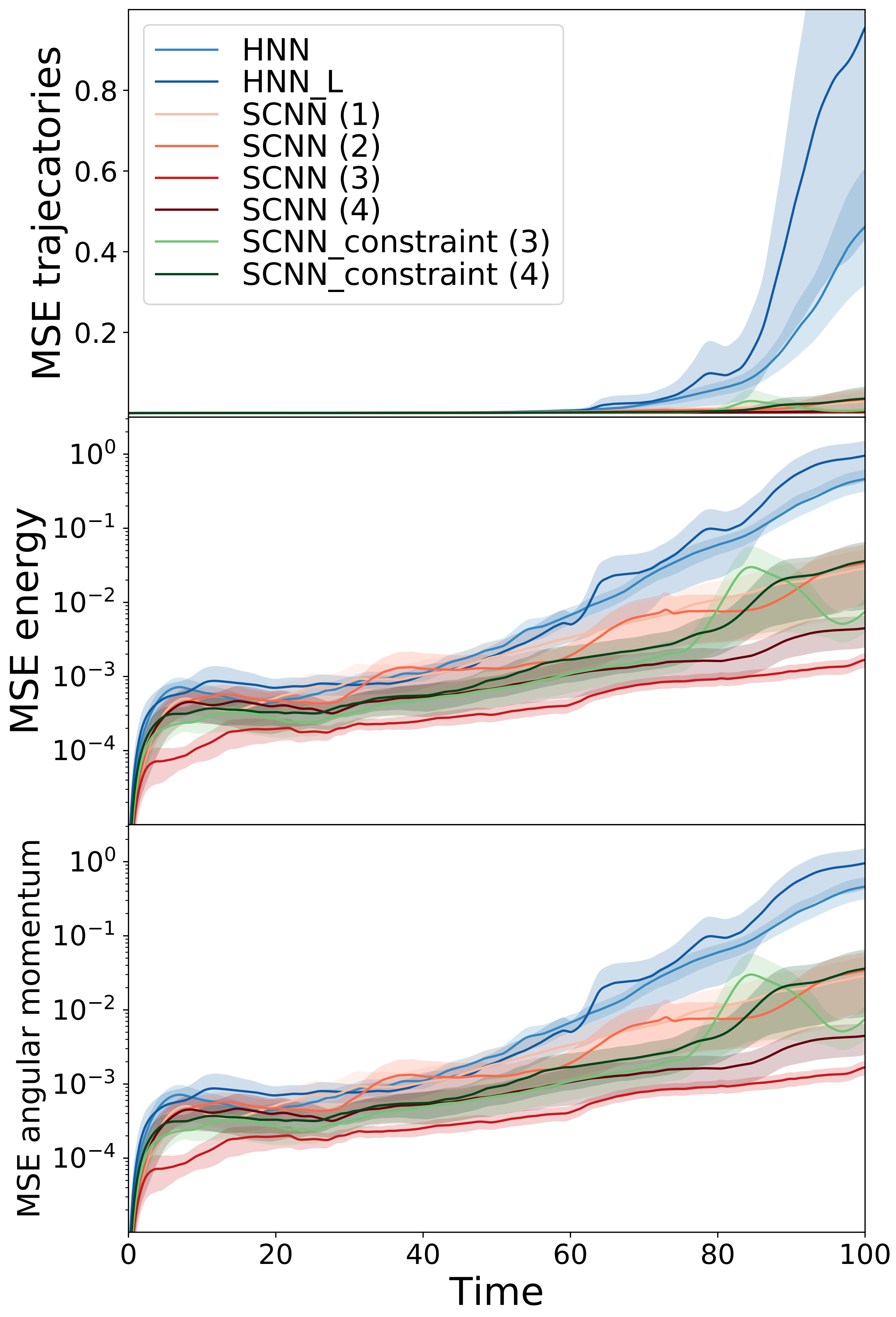
Benefits from Physicists' Bias

- Conserved quantities interpretable:

$$P_{c_1} = -4.2p_{x_1} - 4.2p_{x_2} - 1.3p_{y_1} - 1.3p_{y_2}\ ,\ P_{c_2} = -0.9p_{x_1} - 0.9p_{x_2} - 3.2p_{y_1} - 3.2p_{y_2}$$

$$L = -1.1q_{x_1}p_{y_1} + 0.9q_{x_1}p_{y_2} + 0.9q_{x_2}p_{y_1} - 1.0q_{x_2}p_{y_2} + 1.0q_{y_1}p_{x_1} - 0.9q_{y_1}p_{x_2} - 0.9q_{y_2}p_{x_1} + 1.0q_{y_2}p_{x_2}$$

- Using learned conserved quantities helps in predicting trajectories



Can we search for new mathematical/physical structures?

Symmetries → Integrability

Krippendorf, Lüst, Syvaeri 2021

Krippendorf, Lüst, Syvaeri 2021

Integrability

A lightning overview

- Additional constraint F_k on motion:
$$0 = \dot{F}_k = \{H, F_k\}$$
How many F_k can there be?

- **System** (2n dimensional) **integrable** iff: n independent, everywhere differentiable integrals of motion F_k (in involution).

- Alternatively search for **Lax pair**:
$$\dot{L} = [L, M]$$
s.t. eom are satisfied. Conserved quantities via:
$$F_k = \mathrm{tr}(L^k)$$
(additional condition for $\{F_k, F_j\} = 0$)

Example: Harmonic Oscillator

- Hamiltonian and EOM:
$$H = \frac{1}{2}p^2 + \frac{\omega^2}{2}q^2\,; \quad \dot{q} = p, \dot{p} = -\omega^2 q$$

- Lax pair:
$$L = a\begin{pmatrix} p & b\omega q \\ \frac{\omega}{b}q & -p \end{pmatrix}, \quad M = \begin{pmatrix} 0 & \frac{b}{2}\omega \\ -\frac{\omega}{2b} & 0 \end{pmatrix}$$

- Conserved quantities:
$$F_1 = 2\lambda$$
$$F_2 = 2\lambda^2 + 4H$$
$$F_3 = 2\lambda^3 + 12\lambda H$$
$$\ldots$$

$\lambda\ldots$ spectral parameter

Krippendorf, Lüst, Syvaeri 2021

Integrability

A search problem with many examples and unexplored theory space

Having a Lax pair formulation of integrability is very convenient, but

- inspiration is needed to find it,
- its structure is hardly transparent,
- it is not at all unique,
- the size of the matrices is not immediately related to the dimensionality of the system.

Therefore, the concept of Lax pairs does not provide a means to decide whether any given system is integrable (unless one is lucky to find a sufficiently large Lax pair).

Beisert: Lecture Notes on Integrability (p17)

We need some *deus ex machina* moment…

Nonlinear Sciences > Exactly Solvable and Integrable Systems

[Submitted on 12 Mar 2021]

Integrability ex machina

Sven Krippendorf, Dieter Lust, Marc Syvaeri

Applications:

- Classical mechanics (e.g. planetary motion)
- Classical field theories (1+1 dimensions)
- Spin Chain Models
- D=4 N=4 SYM in the planar limit
- ...

Formulating the search as optimisation

Formulating the search as optimisation

- **Aim: Method to find new Lax pairs with unsupervised learning (i.e. not requiring prior knowledge of a Lax pair)**

Formulating the search as optimisation

- **Aim: Method to find new Lax pairs with unsupervised learning (i.e. not requiring prior knowledge of a Lax pair)**
- Lax equation as loss:

$$\dot{L} = [L, M] \rightarrow \mathscr{L}_{\text{Lax}} = \left| \dot{L} - [L, M] \right|^2$$

Formulating the search as optimisation

- **Aim: Method to find new Lax pairs with unsupervised learning (i.e. not requiring prior knowledge of a Lax pair)**
- Lax equation as loss:

$$\dot{L} = [L, M] \rightarrow \mathscr{L}_{\text{Lax}} = \left| \dot{L} - [L, M] \right|^2$$

- Equivalence to EOM (e.g. $\dot{x}_i = f_i\left(x_i, \partial x_i, \ldots\right)$): L has to include x_i in some component (LHS of EOM), $[L, M]$ has to include RHS of EOM

$$\mathscr{L}_{\text{L}} = \sum_{i,j} \min_k \left(||c_{ijk}\dot{L} - \dot{x}_k||^2, ||\dot{L}_{ij}||^2 \right) + \sum_k \min_{ij} \left(||c_{ijk}\dot{L}_{ij} - \dot{x}_k||^2 \right), \quad c_{ijk} = \frac{\sum_{batch} \dot{L}_{ij}}{\sum_{batch} \dot{x}_k}$$

$$\mathscr{L}_{\text{LM}} = \sum_{i,j} \min_k \left(||\tilde{c}_{ijk}[L, M]_{ij} - f_k||^2, ||[L, M]_{ij}||^2 \right) + \sum_k \min_{ij} \left(||\tilde{c}_{ijk}[L, M]_{ij} - f_k||^2 \right), \quad \tilde{c}_{ijk} = \frac{\sum_{batch} [L, M]_{ij}}{\sum_{batch} f_k}$$

only fixed up to proportionality (loss function independent of refactor)

Formulating the search as optimisation

- **Aim: Method to find new Lax pairs with unsupervised learning (i.e. not requiring prior knowledge of a Lax pair)**
- Lax equation as loss:

$$\dot{L} = [L, M] \rightarrow \mathcal{L}_{\text{Lax}} = \left| \dot{L} - [L, M] \right|^2$$

- Equivalence to EOM (e.g. $\dot{x}_i = f_i\left(x_i, \partial x_i, \ldots\right)$): L has to include x_i in some component (LHS of EOM), $[L, M]$ has to include RHS of EOM

$$\mathcal{L}_{\text{L}} = \sum_{i,j} \min_k \left(||c_{ijk}\dot{L} - \dot{x}_k||^2, ||\dot{L}_{ij}||^2 \right) + \sum_k \min_{ij} \left(||c_{ijk}\dot{L}_{ij} - \dot{x}_k||^2 \right), \quad c_{ijk} = \frac{\sum_{batch} \dot{L}_{ij}}{\sum_{batch} \dot{x}_k}$$

$$\mathcal{L}_{\text{LM}} = \sum_{i,j} \min_k \left(||\tilde{c}_{ijk}[L, M]_{ij} - f_k||^2, ||[L, M]_{ij}||^2 \right) + \sum_k \min_{ij} \left(||\tilde{c}_{ijk}[L, M]_{ij} - f_k||^2 \right), \quad \tilde{c}_{ijk} = \frac{\sum_{batch} [L, M]_{ij}}{\sum_{batch} f_k}$$

only fixed up to proportionality (loss function independent of refactor)

- Avoiding mode collapse:

$$\mathcal{L}_{\text{MC}} = \max\left(1 - \sum \left| A_{ij} \right|, 0\right)$$

- Total loss:

$$\mathcal{L}_{\text{Lax-pair}} = \alpha_1 \mathcal{L}_{\text{Lax}} + \alpha_2 \mathcal{L}_{\text{L}} + \alpha_3 \mathcal{L}_{\text{LM}} + \alpha_4 \mathcal{L}_{\text{MC}}$$

Applications

Harmonic Oscillator

- Harmonic Oscillator:

$$H = \frac{1}{2}p^2 + \frac{\omega^2}{2}q^2; \qquad \dot{q} = p \ , \qquad \dot{p} = -\omega^2 q$$

- Lax Pair:

$$L = \begin{pmatrix} 0.437\ q & -0.073\ p \\ -0.666\ p & -0.437\ q \end{pmatrix}, \quad M = \begin{pmatrix} 0.001 & 0.329 \\ -3.043 & -0.001 \end{pmatrix}$$

- Consistency check:

$$\frac{dL}{dt} = \begin{pmatrix} 0.437\ \dot{q} & -0.073\ \dot{p} \\ -0.666\ \dot{p} & -0.437\ \dot{q} \end{pmatrix} = \begin{pmatrix} 0.441\ p & 0.288\ q \\ 2.660\ q & -0.441\ p \end{pmatrix} = [L, M]$$

- Conserved quantities:

$$L^2 = \begin{pmatrix} 0.048618p^2 + 0.190969q^2 & 0 \\ 0 & 0.048618p^2 + 0.190969q^2 \end{pmatrix} \Rightarrow \mathrm{tr}L^2 \approx 0.2\ H$$

Applications

Further systems

- Korteweg-de Vries (waves in shallow water):

$$\dot{\phi}(x,t) + \phi'''(x,t) + 6\phi(x,t)\phi'(x,t) = 0$$

$$A_x = \begin{pmatrix} -1.7\phi & 1.7\phi + 1.0 \\ 1.7\phi + 1.0 & -1.7\phi \end{pmatrix},$$

$$A_t = \begin{pmatrix} 5.0\phi^2 + 1.7\phi'' & -5.0\phi^2 - 1.7\phi'' - 0.5 \\ -5.0\phi^2 - 1.7\phi'' - 0.5 & 5.0\phi^2 + 1.7\phi'' \end{pmatrix}$$

- Heisenberg magnet:

$$H = \frac{1}{2}\int dx\, \overrightarrow{S}^2(x)\ , \quad \overrightarrow{S} \in S^2;\text{ constraint:}$$

$$\{S_a(x), S_b(y)\} = \epsilon_{abc} S_c(x)\delta(x-y)$$

$$A_x = -\ \mathrm{i}\ \vec{\sigma}\vec{S} + 0.3\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

$$A_t = \begin{pmatrix} 2\ \mathrm{i}\ S_z & 2\ \mathrm{i}\ S_x + 2S_y \\ 2\ \mathrm{i}\ S_x - 2S_y & -\ \mathrm{i}\ S_z \end{pmatrix} + \begin{pmatrix} \mathrm{i}\ S'_y S_x - \ \mathrm{i}\ S'_x S_y & -S'_z S_x + S'_x S_z + \ \mathrm{i}\ (S'_z S_y - S'_y S_x) \\ +S'_z S_x - S'_x S_z + \ \mathrm{i}\ (S'_z S_y - S'_y S_x) & -\ \mathrm{i}\ S'_y S_x + \ \mathrm{i}\ S'_x S_y \end{pmatrix}$$
$$= 2\ \mathrm{i}\ \vec{\sigma}\vec{S} + \ \mathrm{i}\ \epsilon_{ijk}\sigma_i S_j S'_k\ ,$$

- O(N) non-linear sigma models (Sine-Gordon equation and principal chiral model):

$$\mathscr{L} = -\,\mathrm{Tr}(J_\mu J^\mu)\ , \quad J_\mu = (\partial_\mu g)g^{-1}\ , \quad \mu = 0,1.$$

Perturbations on integrable systems

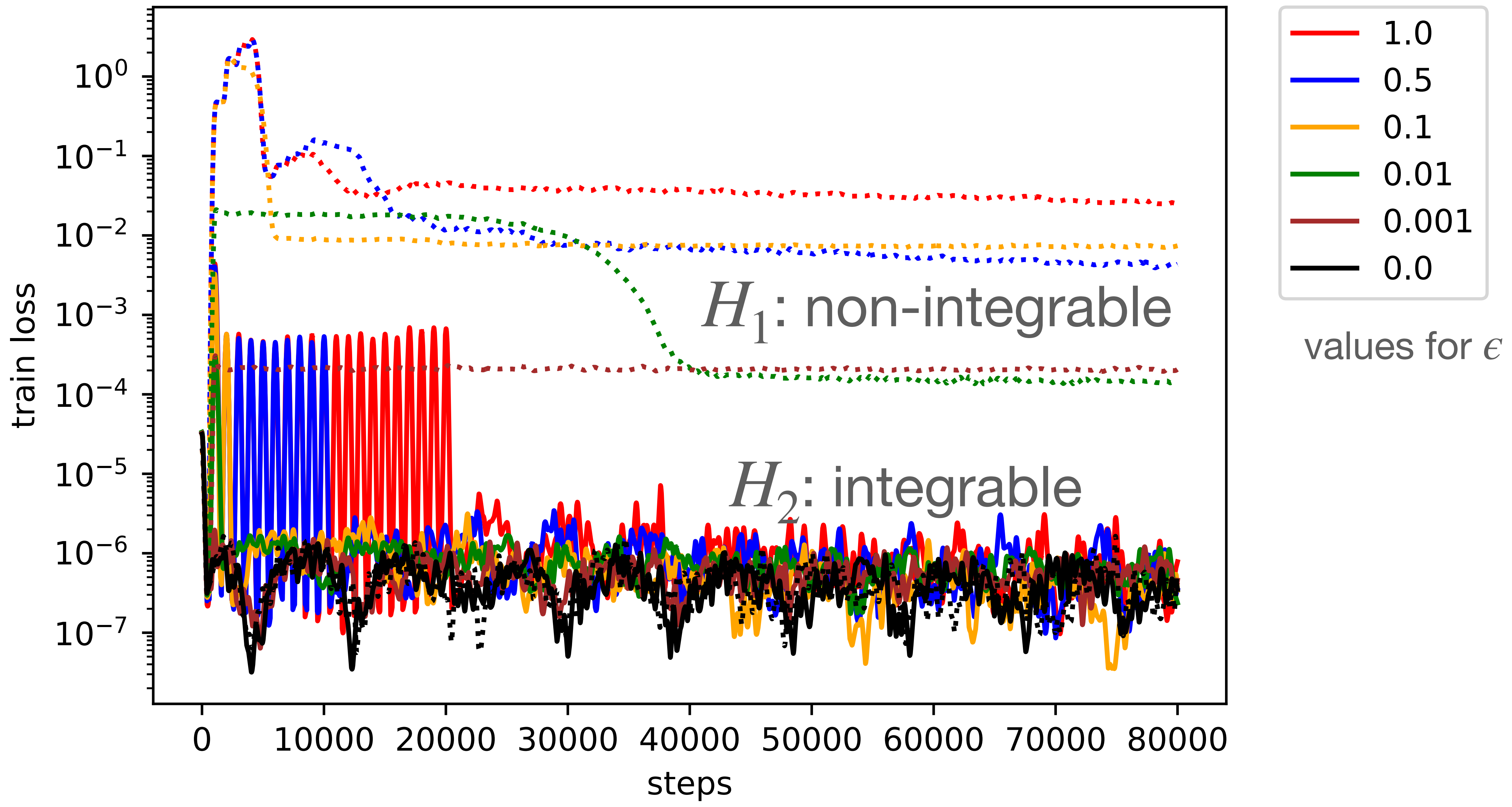
- Harmonic Oscillator:

$$H_0 = \frac{p_x^2 + p_y^2}{2m} + \omega^2 \left(q_x^2 + q_y^2\right)$$

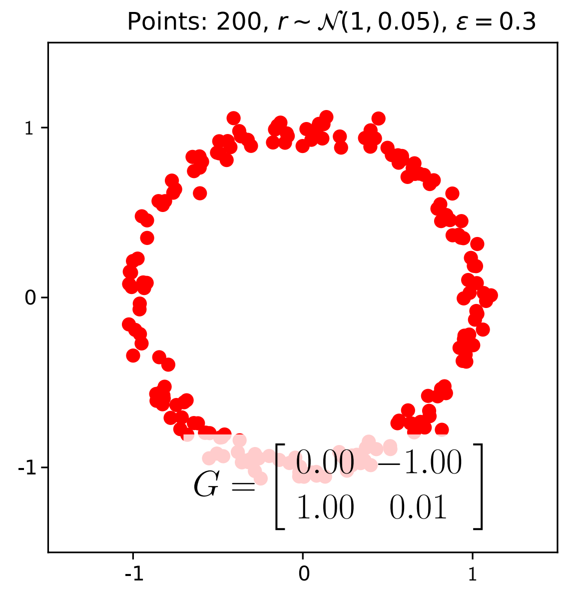
- Are the following perturbations integrable:

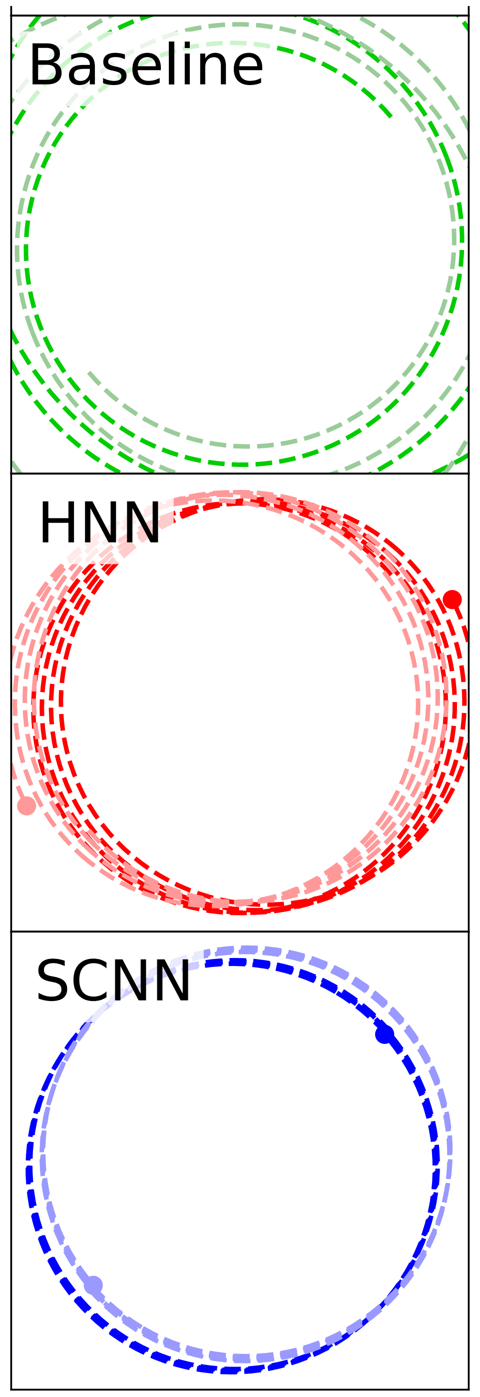
$$H_1 = \epsilon q_x^2 q_y^2, \quad H_2 = \epsilon q_x q_y$$

- Initialise network at known solution for unperturbed system and see how it reacts to samples from perturbed system



Conclusions





- ML search for new types of BSM physics requires search and identification of mathematical structures.
- ML to search for mathematical structures requires careful setting up of learning problem (loss, data, architecture, evaluation/integration)
- We can use ML to identify symmetries in an unsupervised way: embedding layer [no direct optimisation], phase space samples [energy functionals using classical mechanics knowledge]
- Road to making these ML search strategies useful: learning Lax pairs to identify integrability of system. Enables search for integrable perturbations.
- Symmetries are one example, talk to me for our work on extra-dimensional metrics or search for dualities.

Thank you!

2104.14444: Simulations with Symmetry Control Neural Networks
2103.07475: Integrability
2003.13679: Symmetries from Embedding Layer

For talks at the interface of physics and ML: physicsmeetsml.org