Improving Simulations with Symmetry Control Neural Networks

The dynamics of physical systems is often constrained to lower dimensional sub-spaces due to the presence of conserved quantities. Here we propose a method to learn and exploit such symmetry constraints building upon Hamiltonian Neural Networks. By enforcing cyclic coordinates with appropriate loss functions, we find that we can achieve improved accuracy on simple classical dynamics tasks. By fitting analytic formulæ to the latent variables in our network we recover that our networks are utilizing conserved quantities such as (angular) momentum.

Main idea: using and exploiting conserved quantities to constrain motion of particles

- Scheme: Predicting motion of particles using physical bias (HNN and additional conserved quantities)
- Motion governed by Hamiltonian
- Hamiltonian Neural Network restricts motions to hyperplanes (in phase space) of constant energy
- Additional constraints due to other (unknown) conserved quantities (e.g. angular momentum)
- Using constraints from conserved quantities to improve the Hamiltonian
- Side effect: reveal symmetries (see also [Krippendorf, Svyatá 2020])

Theory of SCNN

- Hamiltonian Neural Network approach [Greydanus et al., 2019]: learn motion of particles by learning Hamiltonian; using Hamiltonian equations to compute time evolution (phyics bias):
  \[ \frac{dq_i}{dt} = \frac{\partial \mathcal{H}}{\partial p_i} = \{q_i, \mathcal{H}\}, \quad \frac{dp_i}{dt} = -\frac{\partial \mathcal{H}}{\partial q_i} = \{p_i, \mathcal{H}\} \]
- Problem: Rather unstable after longer time period, fails to conserve conserved quantities
- Solution: additional physics bias, use coordinate transformation to generalized coordinates \( (P, Q) \)
  \[ \mathcal{T}^*_p: (q,p) \mapsto (Q(q,p), P(q,p)) \]
  \[ \mathcal{T}^*_q: (P,Q) \mapsto (P(q,p), Q(q,p)) \]
  \( \mathcal{T}_q \) network for coordinate transformation, \( \mathcal{T}_p \) for Hamiltonian
  \( \mathcal{T}_p \) can be flexible (learning conserved quantities) or known conserved quantities can be enforced

How do additional constraints on the Hamiltonian arise from this bias?

- Conservation of generalized momenta, constrains the structure of the Hamiltonian
  \[ 0 = \dot{P}_\alpha = \frac{\partial \mathcal{H}}{\partial \dot{Q}_\alpha} = \{P_\alpha, \mathcal{H}\} \]
- Different coordinates have to satisfy Poisson algebra
  \[ \{Q_\alpha, P_\beta\} = \delta_{\alpha\beta} \{Q_\alpha, Q_\beta\} = \{P_\alpha, P_\beta\} = 0 \]

Architecture of the neural network

- Both neural networks consists of hidden layer with 200 neurons each and tanh-activation

Experiments

Does SCNN with and without domain knowledge perform better than HNN?

1) SCNN (k): SCNN with k known conserved coordinates
2) SCNN_constraint (k): SCNN with k known conserved coordinates (using domain knowledge)

How well are the trajectories?

Networks with varying number of conserved quantities

Architecture of SCNN: Canonical Transformation Networks

Sample trajectories

Gravitational 2-body

Neural Network and loss components

- Hamiltonian loss:
  \[ L_{\text{SCNN}} = \sum_i \left( \mathcal{L}(P_i, Q_i) - \delta \right) + \mathcal{L}(P, \dot{Q}) \]
- Poisson algebra loss:
  \[ L_{\text{HNN}} = \sum_i \left( \mathcal{L}(Q_i, Q_i) - \delta \right) \]
- Loss on cyclic coordinates:
  \[ L_{\text{SCNN}} = \sum_i \left( \mathcal{L}(Q_i, Q_i) - \delta \right) \]

Additional loss components force the neural network to find conserved quantities and use them to constrain Hamiltonian

- Complexity: Inference is comparable to HNN

Analytical formulae for conserved quantities

- Fitting of generalized coordinates (and specially conserved quantities) possible
- Improvement in prediction precision and computing efficiency
- Conserved quantities in the gravitational two body system (example):
  \[ \begin{align*}
  P_1 &= -4.2 q_1 - 4.2 p_1 + 1.3 q_3 - 1.3 p_3 \\
  P_2 &= -0.9 p_1 - 0.9 q_2 + 3.2 q_1 - 3.2 p_1 \\
  L &= 1 q_1 q_2 + 0.9 q_1 p_2 - 0.9 q_2 p_1 + 1.3 q_3 p_2 - 1.3 q_2 p_3 + 1.0 q_3 q_2 + 0.9 q_3 p_1 - 0.9 q_1 p_3 + 1.0 q_1 q_3 \\
  \end{align*} \]

Outlook and related work

- In the domain of the work of [Battaglia et al.]
- As for HNN, natural extensions to Graph Neural Network [Sanchez-Gonzalez et al., 2019] and in flows [Toth et al., 2019]
- Applications within molecular dynamics and astrophysical simulations

References