# A Deeper Look into LieConv

An-Ya Olson and Jonathan Palafoutas, 2023

# LieConv refresher

- Finzi et. al (2020): “Generalizing Convolutional Neural Networks for Equivariance to Lie Groups on Arbitrary Continuous Data”
- Group convolutional layer with equivariance to transformations from any Lie group
    - 2D translations: T(2)
    - 2D rotations: SO(2)
- Generic network architecture applied to many different types of data
- Dynamical systems application: modeling equivariance in Hamiltonian physical systems → preservation of physical quantities

# File directory

- **dynamics.py** - defines a function which returns a time-series of momentum from a trajectory input for each system

$$z = (nb, n, D) \rightarrow \Sigma p_x, \Sigma p_y, \Sigma p_\Theta = (nb, n)$$

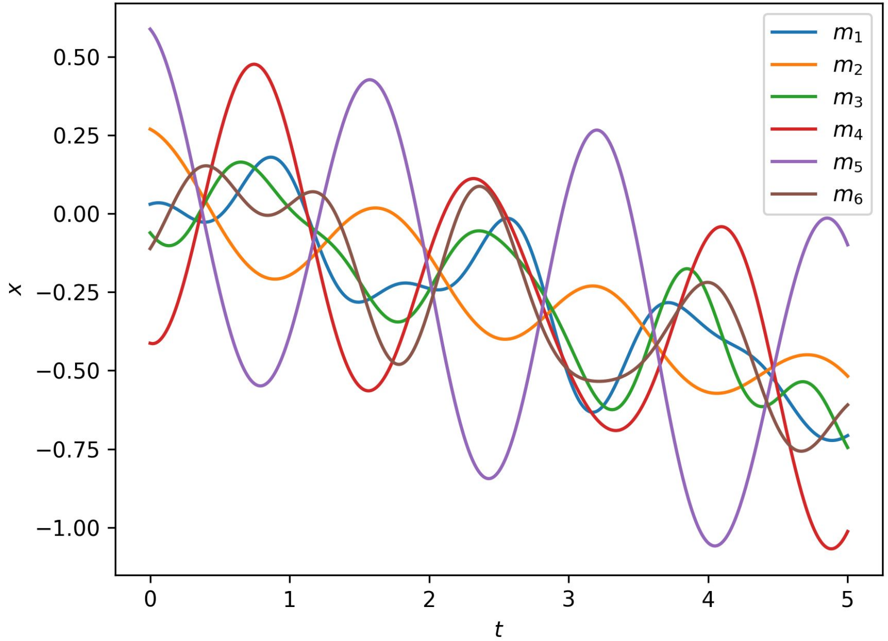
- **model_config.py** - defines model and training parameters
- **model_integrator.py** - defines a function which returns the predicted trajectory of a system provided a neural network as the dynamics function and an initial state

$$z0 = (nb, D), t = (nb, n) \rightarrow z = (nb, n, D)$$

# File directory continued

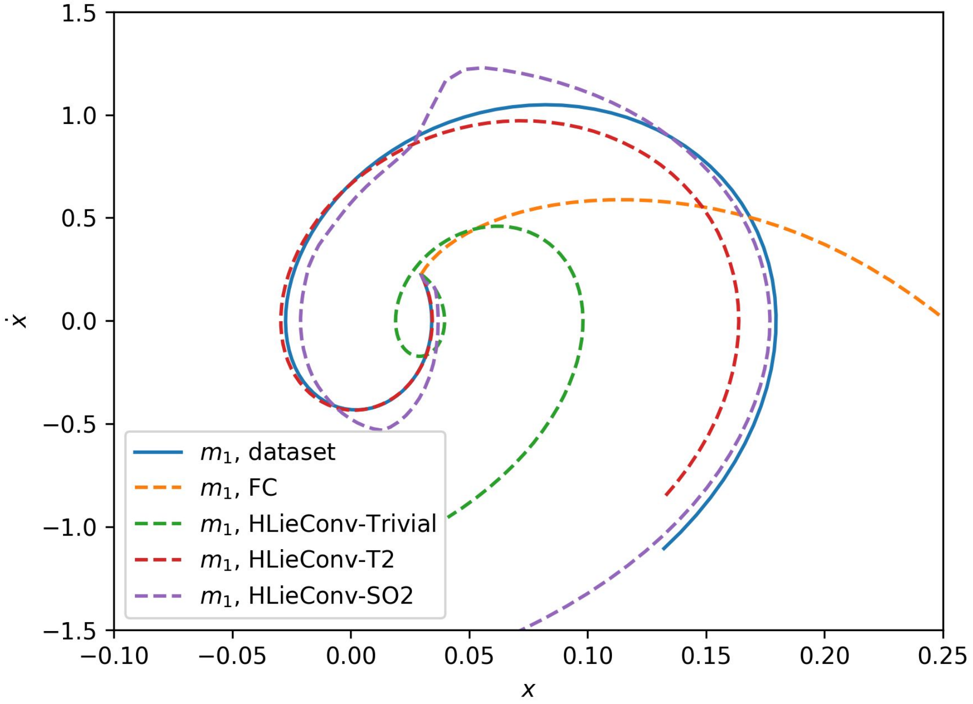
- **spring_trainer.py** - defines a model trainer which generates and saves spring-coupled mass data if it isn’t present
- **train_springs_FC.py** - trains and saves a fully connected network from the spring data
- **train_springs_HLieResNet.py** - trains and saves an HLieResNet from the spring data
- ☆**run.py** - choose a sample system from the dataset, plot the classically simulated ground truth, plot trajectory and momentum using the trained models’ dynamics predictions

# Spring-coupled mass dataset



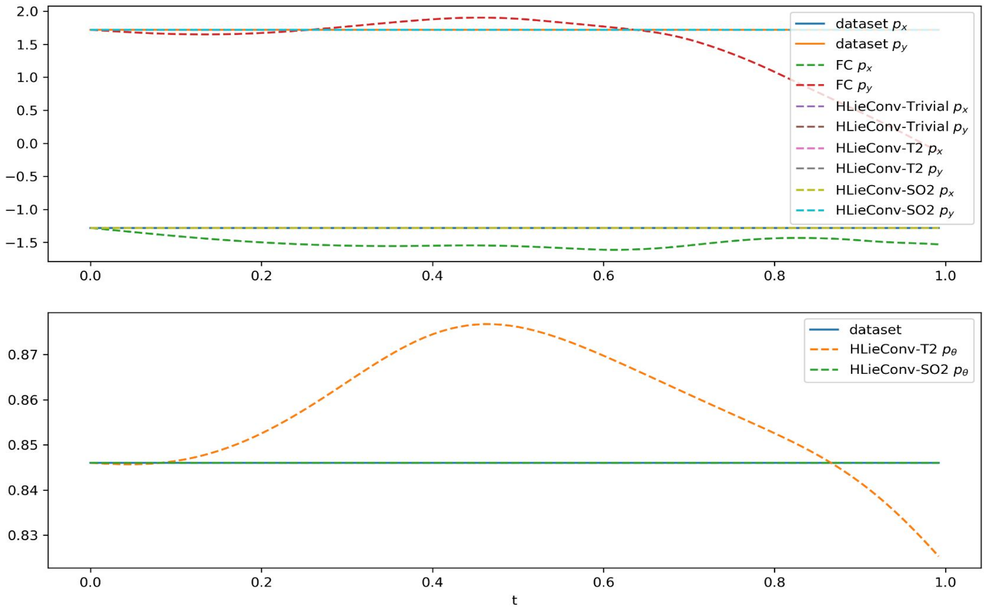

- 6 masses in two dimensions = 24-dimensional state vector
- Generated using **torchdiffeq.odeint**'s rk4 integration
- 10,000 such systems evolved for 5 seconds

# Trajectories



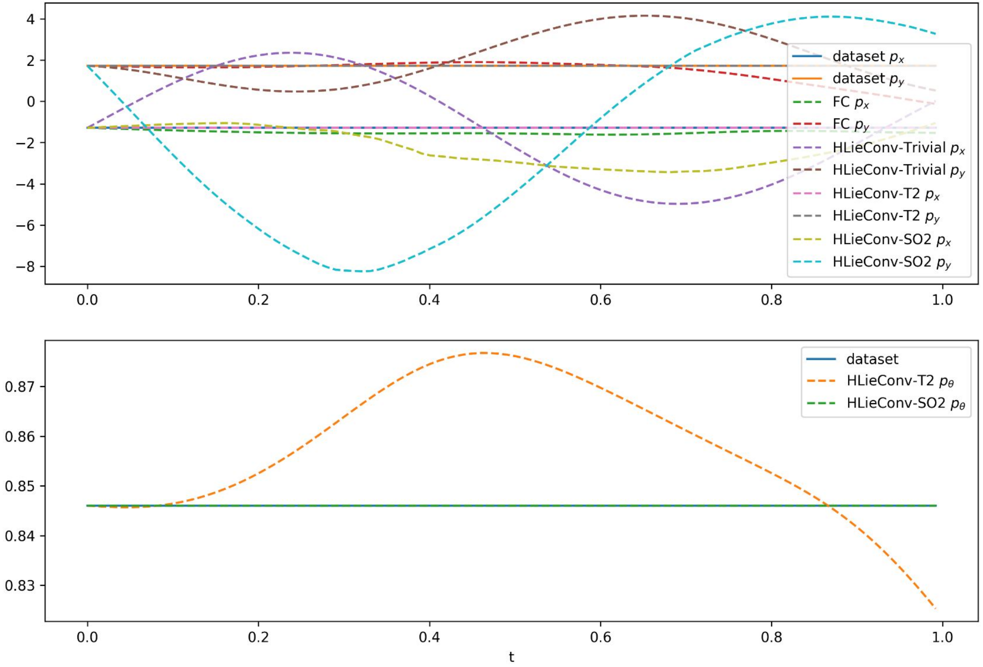

- Four models were trained with low-cost parameters
    - 2 layers
    - 150 nodes per layer
    - Batch size of 1000
    - 20 epochs for HLieResNet, 100 for FC

# Momentum



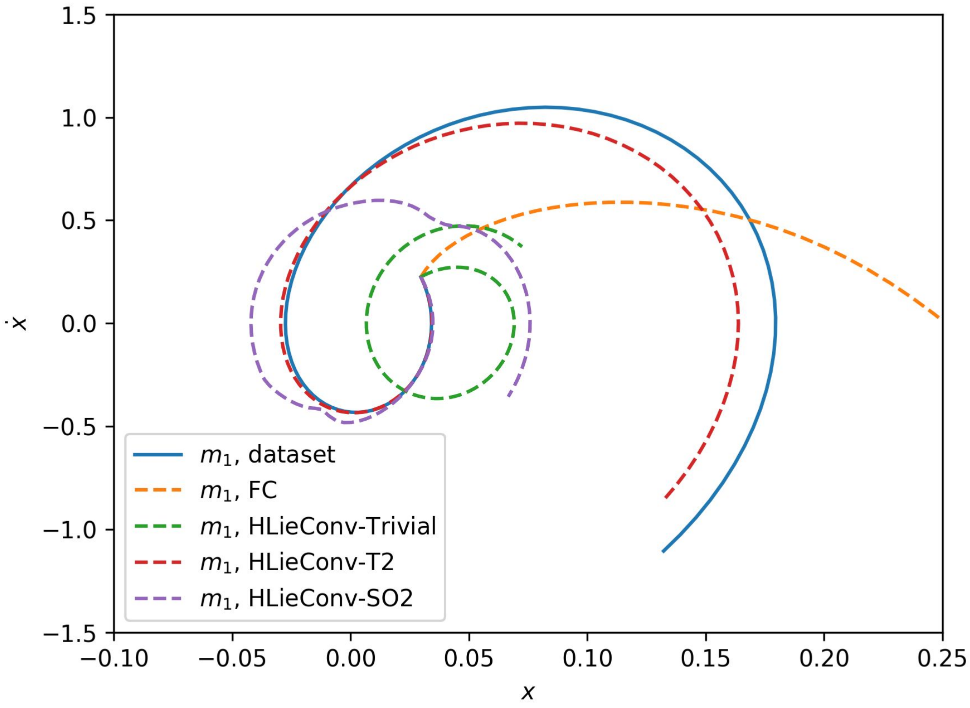

- All HLieConv models approximately conserve linear momentum in both directions
- Why? This is not the expected behavior
- Unsurprisingly, only the SO2-embedded model conserves angular momentum

# Momentum with HLieConv centering disabled




- Now only the T2-embedded model conserves linear momentum
- Disabling centering had no effect on the results for angular momentum

# Trajectories with HLieResNet centering disabled




- Only T2-embedded model should have translation equivariance and it yields the best results

# Conclusion

- Position centering causes a group convolution layer to be translation invariant regardless of whether translation is the convolution group (Finzi 2020).
- Translation invariance allows a Hamiltonian model to conserve linear momentum.
- For springs-mass system, translation equivariance still conserves linear momentum because the potential is already translation invariant.
- We showed this in slides 7 and 8.

# References

Finzi, M., Stanton, S., Izmailov, P., and Wilson, A. G. Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data, 2020.