HELMHOLTZ AI

DEEP LEARNING AND PHYSICS

Student Projects

Stefan Kesselheim

Forschungszentrum Jülich & University of Cologne / 2025-01-27

Student Projects

Goals:

1. Deepen hands-on experience with ML.
2. Go through the typical ML workflow.
3. Understand an exciting use-case better.
4. Use limited ressource (30h per student) to solve a concrete problem.

Mode of operations

- You work in pairs.
- We jointly define a problem and a work plan.
- You work on the problem. Ping we with questions and problems on
- At the end, you create a report of 2-3 pages, and submit the it together with your code.
- It is absolutely fine, if your results are not great, but document what you did, even though it turns out it was not the most successful strategy.

Physics-informed Neural Networks

- Pick a PDE. For example: Heat equation.
- Construct a geometry with splines in 2D.
- Define initial and boundary conditions.
- Solve the PDE with a PINN using Physics-based losses.

Large Language Models

Comparing different models on Physics-related questions.

- Use the Benchmark MR-BEN [1] to inspire tasks.
- Evaluate different open source LLMs on questions such as:

Under ideal conditions, the electric and magnetic fields inside a superconductor are zero. Maxwell's equations imply that which of the following must be true just outside the surface of the superconductor?", "Options": "A: B = 0, B: B is perpendicular to the surface. C: B is tangential to the surface. D: B is time independent.

- Prompt the models to perform chain-of-thought reasoning.
- Check if LLMs can spot the error in a chain-of-thought.

[1] https://randolph-zeng.github.io/Mr-Ben.github.io/.

Neural Network Potentials 1

Calculate the heat capacity of a molecular gas with a NNP.

- Construct a small molecule in `ase`.
- Calculate energy with SchNet.
- Optimize the geometry.
- Calculate the Hessian of the energy.
- Decompose into normal models.
- Apply the formula for heat capacity to each normal model

$$C = \frac{\partial E}{\partial T} = \left(\frac{\hbar\omega}{T}\right)^2 \frac{e^{\hbar\omega/T}}{\left(e^{\hbar\omega/T} - 1\right)^2} \equiv \left[\frac{\hbar\omega/2T}{\sinh(\hbar\omega/2T)}\right]^2 .$$

- Compare to experimental data.

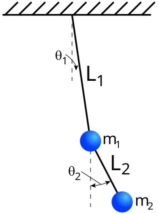
Neural Network Potentials 2

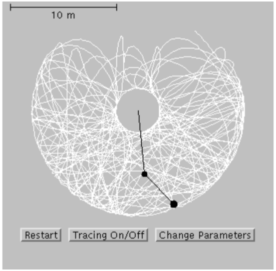
Calculate the chemistry of a chemical reaction.

- Construct educt and product state of a molecule in `ase`
- Use the nudged elastic band method [2] to create states between educt and product state.
- Use SchNet to evaluate the energy of every state.
- Let the method determine the reaction pathway.
- Interpret the result: Energy difference and activation energy.

[2] https://wiki.fysik.dtu.dk/ase/tutorials/dissociation.html

Inverse modeling of the double pendulum





- Simulate the double pendulum for a lot different parameters. generate an image of the trajectory.
- Train a Neural Network to predict the simulation parameters.
- Extend the Neural Network to a Mixture Density network, train with maximum likelihood.
- Evaluate the agreement of MDN and ground truth.

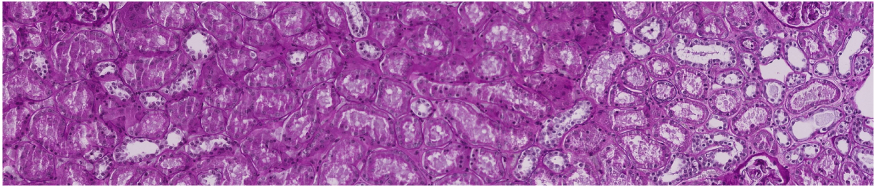
Small Angle Neutron Scattering

Small Angle Neutron Scattering (SANS) is a powerful experimental technique for probing the structure of materials at nanometer scales. Its applications range from materials science and biology to soft matter physics. The ability to accurately simulate SANS experiments using Monte Carlo methods is crucial for instrument design, optimization, and data interpretation. These simulations enable the study of diverse scattering models under various instrument configurations, providing insight into the relationships between experimental setups, model parameters, and resulting scattering patterns.

Jose Robledo

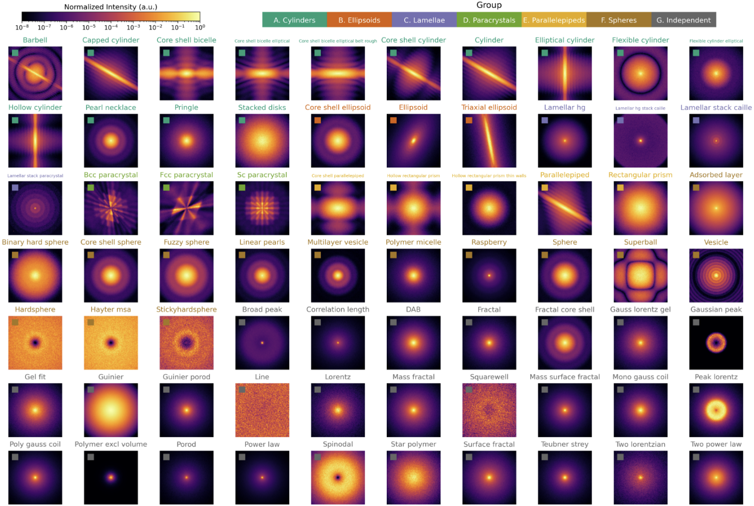
Semantic Segmentation of medical images

- Goal: Semantic Segmentation of images from the field of health.
- Example: Instance segmentation of microvascular structures from healthy human kidney tissue slides[3].



[3]https://www.kaggle.com/competitions/hubmap-hacking-the-human-vasculature/code

Small Angle Neutron Scattering



Small Angle Neutron Scattering 1

Train an ML model to classify the model ID based on the 2D SAS intensity arrays. Arrays are already normalized and a custom DataLoader, as well as the Train, val and test sets. Train a convolutional Neural Network for classification. Evaluate the model's accuracy on the test set.

- What is the classification accuracy of your model?
- Which models are the most challenging to classify, and why?

Small Angle Neutron Scattering 2

Train a neural network to predict the model parameters for a given 2D SAS intensity array. For this, scale model parameters, handle models with varying quantities (keep only those models with 12 parameters, there should be eight models). Define and train a regression model of your choice. Evaluate the model using the mean squared error (MSE).

- How well does your model predict the parameters for different models?
- What are the challenges when models have significantly different parameter spaces?

Small Angle Neutron Scattering 3

Train a generative model to reconstruct the 2D SAS intensity array from the input features of your choice.
For this, preprocess the data, define an train one (or both) of the following architectures: autoencoder or GAN. Evaluate the quality of the reconstructed scattering patterns using the mean squared error (MSE) metric.

- How accurately can your model reconstruct the scattering pattern?
- How does the reconstruction quality vary for different models and instrument configurations?

Overview

1. Physics-informed Neural Networks
2. Neural Network Potentials 1
3. Neural Network Potentials 2
4. Inverse modeling of the double pendulum
5. Semantic Segmentation of Images
6. Small Angle Neutron Scattering 1
7. Small Angle Neutron Scattering 2
8. Small Angle Neutron Scattering 3

References I