DATS 598: Data Science Capstone

Journal Entries and Progress Documentation

Hybrid Symbolic-Generative AI Physics-Informed Neural Networks for Generating First-Principle Physics-Based Simulations

Eric Crisp
University of Pennsylvania
`ecrisp@upenn.edu`

Tuesday 2$^{\text{nd}}$ September, 2025

Contents

Journal Entry - Week 1

Date: Tuesday 2^{nd} September, 2025
Duration: 6 hours

Objective

To begin the planning and synthesizing part of the project including literature review, data location, and beginning to define scope.

Activities Completed

- Conducted initial literature review on state of physics-informed neural networks
- Began populating database for retrevial of thermodynamic training data using CoolProp/NIST
- Set up repositiory, python environment, and docker in case deploying
- Developed fundamental tools and data structures and began validating against open source utilities

Key Findings

We want to be able to predict the state of a fluid (i.e., water) when undergoing physical processes like expansion, contraction, and hopefully phase change. To do so, gathering either formula or dataset of points is required. This is tabulated to a SQLite database using grid population from CoolProp.

Challenges Encountered

- Understanding the technical feasibility of the process
- Ensuring the current literature review does not already contain results regarding this process
- Understanding and determining data storage, access, and compute limitations

Next Steps

1. Finish literature review and summarize
2. Leverage open source but understand where limitations are likely to occur
3. Develop timeline and process chart for ensuring progress and hitting milestones

Reflections

While this project is intentionally open ended, define clarity on the goal. We aim to achieve the following:

1. Can a neural network architecture, when combined and trained with first-principle physics, develop physically valid equations?
2. Leveraging LLM, can the physically valid equations be converted into syntactically and mathematically correct code?
3. Can the process take timeseries data (i.e., sensor data) and derive a simple model (simulation) that captures the neccesary physics?

A Code Repository Structure

```
pinn_dats598/
 .gitignore
 documents/
 journals/
 lectures/
 refs/
 results/
 notebooks/
 src/
    __init__.py
    activation.py
    neural_network.py
    opensource.py
    perceptron.py
 tests/
    __init__.py
    test_perceptron.py
 data/
 docs/
```