HU Shurui: Project

The goal for this project is to understand whether deep learning for macro models can be sped up using gradients of the model statistics (and whether people are doing this currently). If this is not interesting, we will switch back to the conflict project.

Week 1: February 6th

(1) Read the following survey of ML for macroeconomists by Jesus Fernandez-Villaverde.
(2) For the listed models, present the macroeconomics solutions (variables, policy functions, steady-state) in three ways:
    (i) Closed-form: pen and paper formulas using first-order conditions,
    (ii) Bellman equations: pen and paper formulas using Bellman formulas (reach out if you need a good textbook for Bellman equations),
    (iii) Numeric policy function iteration,
    (iv) Polynomial policy function iteration,
    (v) One layer policy function,
    (vi) Multilayer policy function,
    (vii) Policy function that feeds closed-form model parameters/gradients into weight matrix optimization
(3) For (vii) understand the gains in terms of speed

Week 2: February 21st

(1) Read the following paper on structural reinforcement learning: https://benjaminmoll.com/wp-content/uploads/2025/12/SRL.pdf
(2) Start reverse-engineering what they are doing:
    a. Familiarize yourself with the benchmarks they are working with:
        i. Huggett, Krussell-Smith, TANK, HANK, etc
        ii. If helpful prepare a note that details the setup of every model and what known issues every model has in terms of solution methods
    b. Try to code up their method
        i. Here are some resources for a different type of methods: https://github.com/shade-econ/sequence-jacobian
        ii. I can point to more if necessary: essentially, people have tested various codes on standard models
(3) Think whether known jacobians with respect to model objects can speed up any steps or perhaps improve existing methods. (Say, adding those as a basis to the Chebyshev polynomials?)