598sml — Projects

- **Goal: deep dive into a specific aspect of SciML**
  - **Need not be PINNs**
  - **Must be SciML (i.e., we're not interested in image training)**
  - **HPC, Physics, UQ, error estimation, PINN+X, DGM, turbulence, etc all good topics**
- **Guidelines**
  - **Solo or in pairs. If in pairs, then you are agreeing to equal effort on coding, writing, ~~or~~ and presenting**
  - **Each Monday we will be doing updates. Each Wednesday we will be covering special topics.**
  - **Presentation as the final. Details forthcoming (see schedule).**
- **Scope**
  - **Pair with your research**
  - **Define something specific**
  - **Data workflows.**

Steps

- **Steps:**
  - `prj00`: **selecting a topic**
    - **1. check in short project description**
    - **2. peer feedback**
  - `prj01`: **1/2 page - 1page description of your topic and the steps**
    - **identify at least one reference**
  - `prj02`: **Project goals and workflow**
  - `prj03`: **Project setup and initial results**
  - `prj04`: **Summarize model, loss, and training results**
  - `prj05`: **Peer feedback**
  - `prj06`: **Slides draft (1-3 slides)**
  - `prj07`: **Final slides**

- **Ideas**
  - Error bounds and Sobolev theory
  - Inverse problems in materials (stress, strain, etc)
    - Open question: access to experimental data
  - Meta-material design
    - Parameter optimization (reduced order modeling?)
    - Open question: converge on a specific applicaiton
  - Comparing stability, accuracy, **cost** with a conventional method (FD, FE, etc etc)
  - Conventional: discretize -> A x = b. **Can we** train to find A^{-1}?
    - Recast: can we learn Green's functions?
    - Recast: neural operator learning — a mapping between boundary data/solution
  - NN without full use of automatic differentiation
    - Use of adjoint equations, for example

- **Ideas**
  - Adaptive activation functions. What are they and how/when do they work?
  - Causality
  - Pick a method and a new problem
    - Add/remove different “tricks” — which matter? Initialization? Optimizer? Layers? Activation?
  - Other networks
    - GNNs (graphs)
    - LSTMs, transformers
  - Connection between layers and frequency
    - Single layer, arbitrarily wide — universal approximation
  - Global structure vs local structure
    - Can NN identify both?
  - Long time evolution

- **Today**
  - **`prj01`: 1/2 page - 1page description of your topic and the steps**
    - **identify at least one reference**

  - **`prj02`:**
    - **`netid.md`:** (if a group project, then note “joint with **`othernetid.md`**”)
      1. **Establish goals**
      2. **Identify your data**
      3. **Anticipate storage needs and location**
      4. **Commit to a naming scheme**
      5. **Outline your verification plan**
      6. **Map out your computing/resource needs**

- **1. Goals**
- Break the project into a sequence of goals
- This should include at least
  - A quick goal, that can be accomplished with certainty.
  - A middle goal that is doable, yet interesting.
  - A stretch goal that is more ambitious.
- **2. Data**
  - Training data?
  - Testing data?
  - Verification data?
  - Experimental data?

- **3. Storage**
  - Where will you run things?
  - Where will you store things?
  - `598sml-f23` is not your working directory
  - Create your own private repo and add `lukeo@illinois.edu` and `mwest@illinois.edu`
- **4. Naming**
  - `run.py` is insufficient
  - separate training from testing from visualization
  - clearly identify steps
    (example: https://lagrange.mechse.illinois.edu/latex_quick_ref/)

- **5. Verification (and validation)**
  - Do you have analytical solutions?
  - Are there limiting cases to consider?
  - Is there a comparison with other codes (or someone else's example)?
  - Do you have experimental data/results to validate against?
- **6. Compute resources**
  - Where will you run your code?
  - Do you need special machine access?
  - How many nodes and how many GPUs do you anticipate?

- **Today**
  - **Take aways so far:**
    - Projects are too ambitious!
    - Lack of clear reference (to reproduce, to follow)
    - Exciting collection of applications!
  - **`prj02:`**
    1. **Establish goals**
    2. **Identify your data**
    3. **Anticipate storage needs and location**
    4. **Commit to a naming scheme**
    5. **Outline your verification plan**
    6. **Map out your computing/resource needs**

- **Today**
  - **Take aways so far:**
    - Projects are too ambitious!
  - **`prj03:`**
    1. **What model are you training?**
    2. **Where are you getting training data (if you need any data)?**
    3. **Which cost function are you optimizing?**
    4. **Do you have any preliminary results already?**
    5. **What problems have you encountered or what do you need help with?**
    6. **What are your next steps and goals for next week?**