# Deep Learning Projects / Spring 2023

## PROJECT DEADLINES AND REPORT GUIDELINES

1. Project description deadline: **03/10 (Friday) 6pm on Gradescope**
2. **Project description guidelines (project descriptions are part of the project grade):**
   (a) Project description: exactly 2-pages in ICLR 2023 format:
   `https://github.com/ICLR/Master-Template/raw/master/iclr2023.zip`
   (b) Description must contain a title and student name + Purdue ID
   (c) Section "Introduction": ~1/2 page to describe objective, clearly define the task (statistically, e.g., learn $p(x)$ , learn $p(y|x)$ where train and test data have the same distribution, etc.)
   (d) Section "Dataset": ~1/4 page to describe dataset used and why it aligns with the task
   (e) Section "Deep learning method used": ~1/2 page to describe deep learning methods that will be used (or tried) and why it aligns with the task
   (f) Section "Related work": ~1/2 page to describe the related work and how it connects to the proposed work
   (g) Section "Expected results": ~1/4 page describing the expected results and metrics of success (how can we tell your project failed or has been successful?)
3. **Final project report due: 04/15 (Saturday) 6pm**
   (a) Final report must contain between 4 and 6 pages in ICLR 2023 format:
   `https://github.com/ICLR/Master-Template/raw/master/iclr2023.zip`
   (b) Description must contain a title and student name + Purdue ID
   (c) Section "Introduction": ~1 page to describe objective, clearly define the task (statistically, e.g., learn $p(x)$ , learn $p(y|x)$ where train and test data have the same distribution, etc.)
   (d) Section "Dataset": ~1/4 page to describe dataset used and why it aligns with the task
   (e) Section "Proposed Approach": 2+ pages to describe deep learning methods that will be used (or tried); formally write down the **objective function**, describe why objective aligns with objective described in the introduction.
   (f) Section "Related work": ~1/2 page to describe the related work and how it connects to the proposed work
   (g) Section "Results": 1+ page describing the results and metrics of success.

**SOME SUGGESTED TOPICS (FOR STUDENTS WITHOUT A TOPIC)**

1. OOD robustness in graph neural network methods
   (a) Choose an inductive graph task or a task that uses GNNs (like neural algorithm reasoning)
   (b) Design a shift between the train and test distributions
      i. A simple shift is graph size. Test graphs are larger than training graphs.
      ii. Changing the distribution of degrees, triangles, etc. is also relatively simple
      iii. Shifts driven by applications are better and easier to justify.
   (c) Instructor will send relevant papers to the ones interested.
2. Contrastive learning on graphs
   (a) Connections to OOD robustness
   (b) Connection with counterfactually-invariant representations
3. Stable diffusion generative methods for sets and graphs
   (a) Stable diffusion is used mostly for ordered sequences (e.g., images, text)
   (b) Design a generative method with some invariance or equivariance, such as a method for sets or graphs (you can make simple modifications to existing methods. Will later list some papers)
4. Stable diffusion for aiding supervised learning via self-supervision
5. Predicting links in knowledge graphs using structural (pairwise) representation methods (GNN-type methods)
   (a) Open problem.
6. Conditional invariances for more robust supervised learning. The choice of invariance changes depending on the input data.
7. Causal representation learning vs contrastive learning
   (a) Connect/contrast causal representation learning with contrastive learning
   (b) Let $T$ be a data-augmentation operator. And do(T) be a counterfactual intervention on the training data that impose this transformation. Is contrastive learning learning a counterfactually-invariant representation? Yes, no? Why?
   (c) Would contrastive learning work if the transformations we must be invariant to are not lumpable (Definition 2 in Mouli & Ribeiro, "Asymmetry Learning", ICLR 2022)