Mathematical Foundations of Machine Learning – Spring 2023

Summer Project list

General comments

A. Use publicly available datasets such as the UCI machine learning repository (http://archive.ics.uci.edu/ml/index.php), Kaggle (https://www.kaggle.com/datasets), Pytorch (https://pytorch.org/vision/stable/datasets.html)
B. In all experiments try to use 5 fold cross validation (split the dataset into 5 equal parts, where each fifth serves as testing in each iteration. Then present average results over the 5 iterations).
C. For regression problems provide average error and std of error statistics (statistical significance is important!).
D. For classification problems provide accuracy (TP+TN)/(P+N), precision TP/(TP+FP) and recall TP/(TP+FN) statistics.
E. Perform hyper-parameter search and try to explain the logic of the best configuration.
F. "Debug" your results: look at confusion matrices, investigate your false positives and negatives. Try to understand where your models fail and try to fix them.
G. For ML problems - compare your results to the results using standard models from Scikit-Learn, etc.
H. For DL problems - **try to start with small datasets and small architectures and work your way from there**. Some computations such as the SparsityProbe on deep & wide networks are relatively heavy. There needs to be a certain ratio between the size of the training dataset and the dimension of the layers, so as the samples are not too sparse in the high dimensional representation.
I. Weights and Biases: In the following files, please insert your wandb credentials (wandb.login + wandb.init), or your scripts won't run:
    1. MFOML_CourseExamples/VisionSparsityProbeExperiments/train/train.py
    2. MFOML_CourseExamples/NeuralCollapse/Vision/init_loader.py
    3. MFOML_CourseExamples/NeuralCollapse/NLP/train_glue.py
    4. MFOML_CourseExamples/NLPSparsityProbeExperiments/train_glue_without_trainer.py
J. Try to come up with other ideas beyond the basic project description.
K. If you have a new research idea using the tools of this course, feel free to propose it. Please note that any project needs to have a "mathematical foundational" component to it. This means, some form of mathematical analysis of the model being built for the task at hand.

Dates:

L. Project selection & team formation deadline: 10th July 2023.
M. Submission Deadline: Friday 8th September 2023.
N. Presentation day: Sunday 10th September 2023

AWS credentials: To be provided by Ido+Yuval personally. Please keep confidential and do not(!) distribute (or mine bitcoin with :-))

Course Image: CourseImageUpdated

a. Create via simulations a dataset of waves with time [0,500] and different source locations. Use as domain a square with a grid of 128x128.
b. Train a regression DL network on a dataset of images at time 500 to predict (x,y) source location.
c. Train a regression DL network on a dataset of images at various times [250,500] to predict source location.

Advanced Projects

11. Understanding the effects of backbone architecture on intermediate NCC accuracy in SSL
    a. Choose a dataset (e.g. CIFAR100, CIFAR10, but not exclusively!)
    b. Choose an SSL algorithm (e.g. VICReg [15]), and check the intermediate layer NCC with respect to different hierarchies. Try to recreate the results of [16].
    c. How does the NCC change with respect to the arch? Check this thoroughly and explain.

12. Understanding the effects of SSL algorithm (VICReg, SimCLR, DINO) on the intermediate NCC
    a. In [16], we showed that SimCLR and VICReg act quite similarly. Does this hold for a wide array of SSL algorithms? Try to incorporate as many different algorithm(also different in theme)
    b. Can you think of a better algorithm? Show us!

13. Phase Retrieval using DL - Follow [13] and apply a DL approach to solve the phase retrieval problem
    a. Use the MNIST and fashion-MNIST datasets
    b. Construct an initial simple network with shallow layers that are fully connected and then deeper layers based on convolutions.
    c. Try to implement a PR network based on the encoder-decoder approach of [13] with the Haar wavelet transform as the encoder-decoder.

14. Spectral physics aware neural networks - Follow [17] and try to use a neural network architecture inspired by spectral methods
    a. You should test linear and nonlinear equations, but you may restrict your experiments to the unit interval.
    b. Reproduce the architectures of [17] that can take as input samples from the initial condition, a point on the unit interval, a time step and output an approximation to the solution.

References

[1] O. Elisha and S. Dekel, Wavelet decompositions of Random Forests - smoothness analysis, sparse approximation and applications, JMLR 17 (2016).

[2] O. Morgan, O. Elisha and S. Dekel, Wavelet decomposition of Gradient Boosting, preprint.

[3] O. Elisha and S. Dekel, Function space analysis of deep learning representation layers, preprint.

[4] H. Kaiming, Z. Xiangyu, R. Shaoqing and SD Jian, Residual Learning for Image Recognition, proceedings of CVPR 2016.