Victor Lemaître

Email : victor.lemaitre@dauphine.eu
Github
Mobile : +33 783219298
LinkedIn

Education

- **Paris Dauphine University** — Paris,France
  First year of Master in Computer Science and Mathematics (Master I2D) — *Aug. 2024 – now*
  Expected graduation date for the second year is may 2026
  - **Relevant Courses**: Machine Learning, Artificial Intelligence (GOFAI), Convex and combinatorial optimization, Game Theory, Decision Theory, Graph Theory
- **Paris Dauphine University** — Paris, France
  BSc in Computer Science and Mathematics; Graduated with honors — *Aug. 2022 – July 2024*
  Ranked 14th out of 201 in the first year and 10th out of 39 in the second, equivalent to 3.5 GPA
  - **Relevant Courses**: Data Analysis, Semi-structured Data, Functional programming, Linear Algebra, Probability Theory
- **Uppsala University** — Uppsala, Sweden
  Exchange program in Computer Science and Mathematics; Passed 4 out of 6 classes with highest honors — *Aug. 2023 – Jan. 2024*
  - **Relevant Courses**: Databases system, Differential equations

Additional Education

- **AI safety Sweden, AI safety fundamentals** — Uppsala, Sweden
  Studied technical aspects of AI safety and alignment. Conducted a research distillation project on shard theory. — *Sept. 2023 - Jan. 2024*
- **Andrej Karpathy, Neural Networks: Zero to Hero**
  13 hours of youtube videos giving in-depth explanations of pytorch's internals, backpropagation and transformer architecture. Reimplemented the multi-head attention layer from simple tensor operations. Used chain of thought to teach a small transformer the addition of two numbers — *Feb. 2024 - April 2024*

Experience

- **Summer research internship** — Paris Dauphine university, France
  Improved the neural network behind AstraZeneca's retrosynthesis tool Aizynthfinder by generating large amount of synthetic data — *May 2024 - Sept. 2024*
  - **Relevant skills**: TensorFlow, Numpy, Pandas

Personal project

- **Search engine for legal precedent**: Utilized parameter-efficient fine-tuning techniques like LoRA or NEAT to train a Bert model for embedding case laws. Preprocessed the data with GPT4o Mini's api before training the model using an unsupervised approach described in the SimCSE paper.
- **Interpreting an MLP trained on modular addition** : Investigated how a one-layer MLP computes modular addition using Fourier transforms inspired by Neel Nanda's Grokking work. Leveraged PyTorch hooks to investigate hidden layer activations and identified key frequency patterns used by neurons

Programming Skills

- **Languages** (From most to least proficient): Python, C, Haskell, Java, OCaml, SQL, R
- **Libraries & Frameworks**: Pytorch, Tensorflow, Numpy, Pandas, Matplotlib