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*
  Ranked 2/34 in the first semester. 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
- **MPSI-MP* Janson de Sailly** Paris, France
  Aug. 2020 – Sept 2022
- **Lycée Notre Dame les Oiseaux** Verneuil-sur-Seine, France
  Baccalauréat Général mention très bien *2020*

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*

Experience

- **Summer research internship** Paris Dauphine university, France
  Under Tristan Cazenave's supervision I 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
- **Summer research internship** LAMSADE, France
  Currently pursuing a research internship under Yann Chevaleyre's supervision on improving the robustness of LLM for roleplay. *May 2025 - Sept. 2025*

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
- **ENS-Challenge data on Parkinson Disease**: Ranked 2nd on the public dataset as of May 26th. The challenge's aim is to predict an unbiased estimate of the patients "OFF score" which describes the severity of their motor symptoms when the effects of treatment wear off. I experimented with several popular ML technique like XGBoost and batch-ensemble before selecting a Transformer architecture combined with deep ensembling.

Programming Skills

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