# Booster vos MLP


## Sevan YERLY

Travail de bachelor 2025

Informatique et systèmes de communication – Ingénierie des données

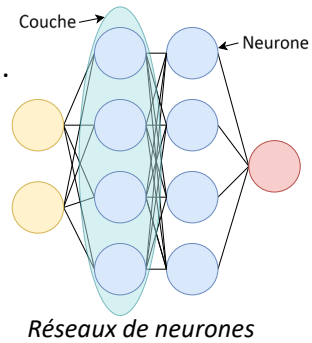
Professeur : Cédric BILAT

Expert : Bastien WERMEILLE


## Description

Un des défis majeurs des réseaux de neurones est de choisir le bon nombre de couches et de neurones.

- Trop peu, et le modèle manque de précision.
- Trop, et il devient lent à l'utilisation et surapprend les exemples d'entrainement.




*Réseaux de neurones*

L'enjeu est de trouver le bon équilibre entre complexité et efficacité.

L'Université de Genève a développé un algorithme permettant de calculer un indicateur aidant à sélectionner uniquement les variables les plus pertinentes afin d'améliorer la précision et la rapidité des prédictions.

Ce travail de bachelor se concentre sur l'optimisation de ce calcul sur carte graphique avec pour objectif de trouver la manière la plus rapide d'évaluer ce facteur clé, nommé Lambda Happy.

## Déroulement

Deux implémentations principales ont été réalisées pour accélérer le calcul :

- la première avec PyTorch, un framework Python de calcul sur tenseurs, servant de référence
- la seconde, plus complexe, avec CUDA, une plateforme de calcul parallèle développée par NVIDIA.

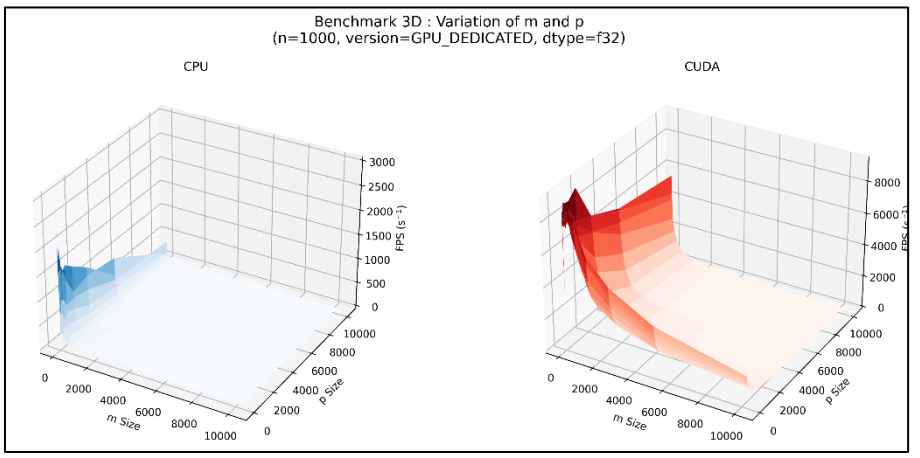
Une validation a ensuite été menée sur un modèle linéaire simple afin de confirmer la stabilité, la précision et la convergence du facteur Lambda Happy.

Pour finir, les codes C++ et CUDA développés ont été intégrés dans du code Python, ce qui a permis de publier un package Python directement utilisable sous Linux.

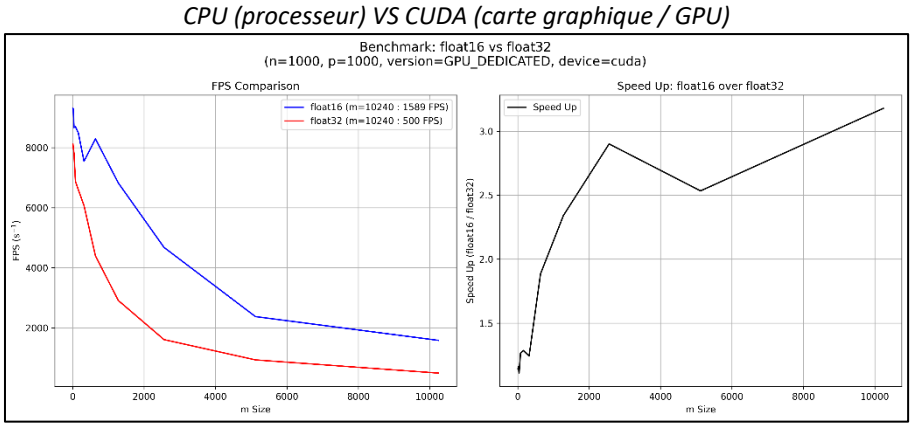
## Résultats

Les résultats obtenus sont prometteurs, passant de 0,4 FPS (nombre d'estimation par seconde) sur la première version utilisant le processeur à plus de 2'104 FPS, soit 5'260 fois plus rapidement.

Cette amélioration significative est obtenue grâce à notre implémentation parallélisant le calcul massivement sur plusieurs cartes graphiques. Cela permet de surpasser l'implémentation de référence PyTorch (1'215 FPS), elle aussi exécutée sur GPU.




*Comparaisons des performances matérielles :*
*CPU (processeur) VS CUDA (carte graphique / GPU)*




*Comparaisons des performances de type : float16 VS float32*

Au-delà du type de matériel utilisé, la manière de représenter les nombres impacte grandement les performances. Notre proposition de réduire de 32 à 16 bits permet de doubler les FPS, tout en préservant la précision requise pour l'application.

## Discussion : conclusions et perspectives

Les performances du calcul du facteur Lambda Happy ont été fortement améliorées. Des validations spécifiques ont confirmé sa justesse. Le package a alors été publié, afin de le rendre disponible à la communauté.

L'implémentation peut être intégrée par le mandant pour les réseaux de neurones sparse (plus légers et moins complexes). Un package alternatif, plus simple à maintenir et sans dépendance à CUDA, a également été fourni et publié.