

# IFT 780 : Devoir 3

## Travail par équipe de 2 ou de 3

Le but de ce devoir est de vous familiariser avec des architectures plus complexes comme les autoencodeurs et les réseaux récurrents. Bien qu'il soit possible d'entraîner un réseau sur CPU, il est beaucoup plus rapide et utile d'utiliser un GPU (processeur de carte graphique). Si vous n'avez pas accès à un GPU permettant d'effectuer des calculs scientifiques (souvent de marque NVidia), vous pourrez utiliser gratuitement les services de « Google Colab » comme pour le TP2.

Dans ce travail pratique, vous aurez très peu de programmation à faire. Plutôt, vous devrez répondre aux questions présentes dans les trois *ipython notebooks* mis à votre disposition. Vous devez inscrire vos réponses dans le fichier « réponses » associé (voir si web du cours). Pour ce qui est des *notebooks*, ils sont disponibles dans le dossier `tutorials/` des fichiers fournis.

1. **[3 point]** Répondez aux questions dans le notebook `mnist_autoencoders.ipynb`. À noter que certaines cellules permettant la visualisation de l'espace latent des autoencodeurs peuvent ne pas fonctionner sur Google Colab. Vous devrez donc exécuter le notebook localement. Si l'entraînement des modèles n'est pas une option sur votre machine ou celles des laboratoires, vous pouvez entraîner sur Google Colab, sauvegarder les poids des modèles, et les charger localement. La documentation de PyTorch sur la sauvegarde et le chargement de paramètres est disponible à cette adresse : [https://pytorch.org/tutorials/beginner/saving\\_loading\\_models.html](https://pytorch.org/tutorials/beginner/saving_loading_models.html).
2. **[3 point]** Répondez aux questions dans le notebook `cardiac_autoencoders.ipynb`. Encore une fois, certaines cellules permettant de visualiser l'espace latent de l'autoencodeur ne fonctionnent possiblement pas sur Google Colab. Vous pourrez appliquer la stratégie indiquée à la question précédente.
3. **[4 points]** Répondez aux questions dans le notebook `rnn_image_captioning.ipynb`. Ce notebook est plus demandant en ressources et il est **fortement** recommandé de l'exécuter sur Google Colab. De plus, pour vous sauver du temps, l'entraînement complet du modèle a déjà été effectué et les paramètres de celui-ci sont disponibles dans le dossier `weights`.