

Réseaux neuronaux  
**IFT-780**

Présentation

Par  
Pierre-Marc Jodoin

1

## Présentation

- **Professeur** : Pierre-Marc Jodoin
- **Courriel** : pierre-marc.jodoin@usherbrooke.ca
- **Page web perso** : [jodoin.github.io](https://jodoin.github.io)
- **Page web cours** : [jodoin.github.io/cours/ift780](https://jodoin.github.io/cours/ift780)
- **Local** : D4-1016-1, pavillon des sciences
- **Période de disponibilités** : [de 9h30 à 17h00](#)

2

2

## Horaire

**Sujet à modification!**

- Cours magistraux
  - voir ici: <http://horaire.dinf.usherbrooke.ca/>
- Travaux dirigés (*parfois, à voir en classe*)

3

3

## Évaluation

- Examen intra : 20%
  - Examen final : 25%
- } Seule les notes manuscrites seront admises
- Travaux pratiques: 45%
    - TP1 : 10%
    - TP2 : 15%
    - TP3 : 5%
    - TP4 : 15% (rapport à remettre)
  - Présentation orale : 10%

4

4

## Évaluation

- Les travaux pratiques se font en **python**.
- Correctrice et soutien technique
  - **Voir site web du cours**

5

5

## Évaluation (suite)

- Les examens intra et final se font **seul**.
- L'examen final portera sur toute la matière vue en classe
- Vous avez droit à des feuilles **manuscrites** pour toute documentation.
- Votre présence aux séances magistrales est fortement recommandée mais non obligatoire.

6

6

## Évaluation (suite)

- Les travaux pratiques se font **PAR ÉQUIPE DE TROIS**.
  - **Pas d'équipe solo!**
- La remise du code et des exercices théoriques (lorsqu'il y en a) se fait par le système **turninWeb**  
(<https://turnin.dinf.usherbrooke.ca/>)
- 10 points de pénalité par jour de retard
- 0 après 5 jours de retard
- Une erreur de remise **peut entraîner une note de zéro**.
- **PAS D'EXCEPTION!**

7

7

## Évaluation (suite)

- Avec le travail à distance, il est **obligatoire** d'utiliser un gestionnaire de code source « git ». Afin de simplifier les choses, veuillez utiliser le gitlab de l'UdeS:

[depot.dinf.usherbrooke.ca](https://depot.dinf.usherbrooke.ca)

- Pas de code envoyé par courriel!
- Une mauvaise utilisation de git peut entraîner une **perte de points**.
- Vous ne connaissez pas git?

[www.tutorialspoint.com/git/index.htm](http://www.tutorialspoint.com/git/index.htm)

8

8

## Évaluation (suite)

Les présentations se font aussi **PAR ÉQUIPE DE TROIS**.

Soumettre le sujet de votre présentation avant la **date limite (voir plan de cours)**.

La présentation doit porter sur un domaine de l'apprentissage profond et d'un article en particulier

- Présentation du problème
- Revue de littérature
- Présentation d'une solution en particulier

Un gabarit vous sera soumis au cours de la session.

9

9

## Évaluation (suite)

- **IMPORTANT** : afin d'éviter toute discrimination, les équipes des travaux pratiques seront formées par l'enseignant.
- **IMPORTANT** : en plus de votre code, veuillez soumettre un fichier "gitlab.txt" dans lequel vous donnez le lien vers votre dépôt gitlab. Il est obligatoire d'utiliser gitLab (gitHub, Bitbucket, etc. sont interdits.)
- **IMPORTANT** : en plus de vos documents de travail, vous devez remplir, signer et joindre à votre travail le **formulaire d'intégrité** disponible sur le site web du cours.
- **IMPORTANT** : en plus de vos documents de travail, vous devez remplir et joindre à votre travail l'**agenda d'équipe** disponible sur le site web du cours.

10

10

## Évaluation (suite)

Une personne ayant peu contribué au travail d'équipe (**gitLab** et **agenda d'équipe**) s'expose à une **perte de points** et, à la limite, à avoir une note de **ZÉRO**.

**Rencontres hebdomadaires d'équipe obligatoires** (agenda d'équipe).

11

11

## Recommandations

- **N'attendez pas à la dernière minute pour faire les TP**
- Faites 100% des TP et non 50%-50%.
- Jamais une bonne idée de **plagier**
- Feedbacks en temps réel.
- Pas de programmation en groupe sur un ordinateur.
- Pénalité de 10% par jour de retard.
- En tout temps, il revient à vous de ne **pas faire d'erreur** avec le système de remise « **turninWeb** ».

12

12

# NE PLAGIEZ PAS!

(voir dernière page du plan de cours)

13

1er Plagiat = *black list* facultaire

2<sup>e</sup> Plagiat = expulsion

14

ChatGPT = plagiat  
Code en ligne = plagiat  
Code identique dans 2 travaux = plagiat  
Travailler fort + plagiat = plagiat  
Petit plagiat = plagiat

15

## À partir de maintenant

- Au cours de la 2<sup>e</sup> semaine du trimestre, **les équipes seront formées par l'enseignant.**
- Attention! advenant un **conflit** dans une équipe (mauvaise attitude, incompetence, manque d'ardeur au travail, etc.)
  - **Perte de points à des co-équipiers.ères fautifs.ives.**
- Une personne ayant **trop peu contribué** à un travail s'expose à obtenir la **note de 0**
- **Rencontres d'équipe hebdomadaires obligatoires** (agenda d'équipe).

16

16



## À partir de maintenant

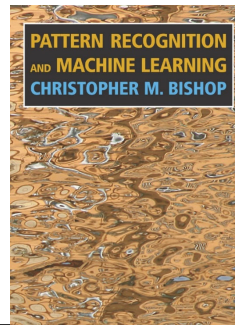
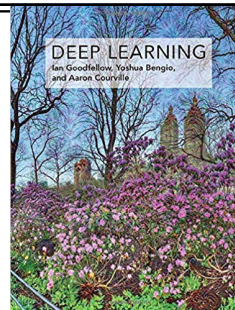
- Révision (ou apprentissage) de python (voir tutoriel sur le site du cours)
- Révision des bases en math (voir vidéos en ligne)
- Révision des notions de techniques d'apprentissage (voir vidéos du cours ift603 ou prendre le cours ift603)

17

17

## Déroulement

- Site web du cours
  - [jodoin.github.io/cours/ift780/index.html](http://jodoin.github.io/cours/ift780/index.html)
- Livres (non obligatoires)
  - *Deep Learning*  
Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press 2016
  - *Pattern Recognition and Machine Learning*  
Christopher Bishop, Springer, 2007
- Où trouver les livres?
  - Amazon
    - [www.amazon.com/Pattern-Recognition-Learning-Information-Statistics/dp/0387310738](http://www.amazon.com/Pattern-Recognition-Learning-Information-Statistics/dp/0387310738)
    - <https://www.amazon.ca/-/fr/Ian-Goodfellow/dp/0262035618/>
  - Bibliothèque de science et de génie
  - Versions gratuites en ligne!!  
[jodoin.github.io/cours/ift780/](http://jodoin.github.io/cours/ift780/)



18

# D'ici la semaine prochaine

## Révision

- Programmation Python 3.x (*Spyder, Pycharm*)
  - Tutoriel Stanford : [cs231n.github.io/python-numpy-tutorial/](https://cs231n.github.io/python-numpy-tutorial/)
  - Tutoriel approfondi : <https://docs.python.org/3/tutorial/>
  - Etc.
- Se familiariser avec Linux (recommandé mais pas obligatoire)
- Se familiariser avec git si vous ne connaissez pas... **ça urge!**
  - Confession d'un gros employeur « *Un élève qui ne maîtrise pas git ne mérite pas de travailler en informatique.* »
- Se mettre à niveau avec les concepts de base en **techniques d'apprentissage** et en **probabilités**

19