

Université de
Sherbrooke

Département d'informatique
IFT 603 / IFT 712 – Techniques d'apprentissage
Plan d'activité pédagogique
Hiver 2024

Enseignant

Pierre-Marc Jodoin

Courriel : Pierre-Marc.Jodoin@USherbrooke.ca

Local : D4-1016-1

Téléphone : +1 819 821-8000 x62025

Disponibilités : Du lundi au vendredi de 9h00 à 17h00

Responsable(s) : Djemel Ziou, Martin Vallieres, Pierre-Marc Jodoin, Shengrui Wang

Site web du cours : <https://jodoin.github.io/cours/ift603/index.html>

Horaire

Exposé magistral :	Mardi	8h30 à 10h20	salle À venir
	Jeudi	10h30 à 11h20	salle À venir

Description officielle de l'activité pédagogique¹

Cibles de formation : S'initier aux techniques d'apprentissage automatique en intelligence artificielle.

Contenu : Concepts fondamentaux de l'apprentissage automatique, formulation mathématique probabiliste, régression linéaire, classification linéaire, astuce du noyau, machine à vecteurs de support, apprentissage bayésien, modèle de mélange de gaussiennes, algorithme EM, réduction de dimensionnalité et combinaison de modèles.

Crédits 3

Organisation 3 heures d'exposé magistral par semaine
6 heures de travail personnel par semaine

Préalable IFT 436, STT 418

Particularités Aucune

¹<https://www.usherbrooke.ca/admission/fiches-cours/ift603>

1 Présentation

Cette section présente les cibles de formation spécifiques et le contenu détaillé de l'activité pédagogique. Cette section, non modifiable sans l'approbation du comité de programme du Département d'informatique, constitue la version officielle.

1.1 Mise en contexte

L'apprentissage automatique ou l'apprentissage par machine (Machine Learning) s'intéresse à la conception, l'analyse, l'implémentation et l'application de programmes d'ordinateur capables de s'améliorer, au fil du temps, soit sur la base de leur propre expérience, soit à partir des données antérieures fournies par d'autres programmes. De nos jours, l'apprentissage automatique joue un rôle essentiel dans de nombreux domaines d'applications, tels que la vision par ordinateur, le traitement automatique du langage, la reconnaissance vocale, les systèmes tutoriels intelligents, la modélisation de l'utilisateur, la robotique, la bio-informatique, les finances, le marketing, les jeux vidéo, la télédétection, etc. En fait, la plupart des programmes de l'intelligence artificielle contiennent un module d'apprentissage. Presque tous les systèmes de reconnaissance de formes sont basés sur des techniques d'apprentissage.

1.2 Cibles de formation spécifiques

Ce cours vise à introduire les concepts fondamentaux de l'apprentissage automatique et à présenter une vaste gamme d'approches différentes utilisées dans des systèmes d'apprentissage automatique. À la fin de cette activité pédagogique, l'étudiante ou l'étudiant sera capable de :

1. Comprendre les notions fondamentales de l'apprentissage automatique ;
2. Comprendre les formulations probabilistes de l'apprentissage automatique ;
3. Maîtriser des algorithmes de régression et classification en apprentissage supervisé ;
4. Comprendre comment utiliser l'astuce du noyau ;
5. Maîtriser les fondements des réseaux de neurones et leur apprentissage par rétro propagation ;
6. Maîtriser des algorithmes de combinaison de modèles.

1.3 Contenu détaillé

Thème	Contenu	Nbr. d'heures	Objectifs	Travaux	Lectures
1	Mathématiques pour l'I.A. : algèbre linéaire de base ; loi gaussienne multivariée, marginale et conditionnelle d'une gaussienne, distance de Mahalanobis, matrice de covariance, valeurs propres et vecteur propres, espérance mathématique ; distribution de vraisemblance, a posteriori, a priori et évidence ; lagrangien.	4	1	✓	[1] sections 1.2, 1.2.1, 1.2.4, 2.3, 2.3.1, 2.3.2 et annexes B, C et E
2	Concepts fondamentaux : apprentissage supervisé vs non-supervisé ; performance d'entraînement vs généralisation, sur- et sous-entraînement ; régularisation ; sélection d'hyper-paramètres et validation croisée ; minimisation d'une fonction de perte, argmin ; exemple de la régression polynomiale ; malédiction de la dimensionalité.	3	1	✓	[1] sections 1.0, 1.1, 1.2.4, 1.3, 2.1, 2.3 et appendice C
3	Formulation probabiliste : théorie des probabilités (cas discret et continu) ; théorie de l'information (entropie, information mutuelle, divergence de Kullback-Leibler) ; bayes et probabilités jointes, conditionnelles, a priori ; introduction au maximum de vraisemblance ; introduction au maximum a posteriori.	4	1 et 2	✓	[1] sections 1.2, 1.2.1, 1.2.2, 1.2.4, 1.2.5, 1.6 et 1.6.1
4	Régression linéaire : formulation probabiliste de la régression polynomiale ; apprentissage par maximum de vraisemblance ; apprentissage par maximum a posteriori.	3	3	✓	[1] sections 3.1, 3.1.1, 3.1.4, 3.1.5 et 3.2
5	Classification linéaire : classification comme un problème de régression ; méthode de Fisher 2 classes ; méthodes génératives ; perceptron binaire vs multiclasse ; régression logistique binaire vs multiclasse ; descente de gradient ; sigmoid vs softmax ; distribution de Bernoulli vs entropie croisée.	6	1, 3 et 5	✓	[1] sections 4.1, 4.1.2, 4.1.3, 4.1.4, 4.2 et 4.3
6	Méthode à noyau : représentation duale et régression à noyau ; construction de noyau ; régression à noyau.	3	4	✓	[1] sections 6.1 et 6.2
7	Méthode à noyau parcimonieuse : classifieur à marge maximale ; machine à vecteurs de support (SVM).	3	3 et 4	✓	[1] sections 7.0, 7.1, 7.1.1 et 7.1.2
8	Réseaux de neurones multi-couches : approche linéaire vs non linéaire ; classification binaire et multiclasse ; fonction d'activation ; couche d'entrée vs couche cachée vs couche de sortie ; dérivée en chaîne, rétro-propagation, problème avec les fonctions d'activation non centrées à zéro, utilisation de mini-batches, disparition du gradient, <i>Dropout</i> ; initialisation d'un réseau ; <i>sanity checks</i> et vectorisation.	6	5	✓	[1] sections 5.1, 5.2, 5.2.1, 5.2.4, 5.3, 5.3.1, 5.3.2, 5.5
9	Combinaison de modèles : <i>bootstrap</i> , <i>boosting</i> .	3	6		[1] sections 14.0, 14.2, 14.3, 14.3.1
10	Théorie de la décision : décomposition biais-variance ; théorie de la décision.	3	1, 2 et 3		[1] sections 1.5.5 et 3.2
11	Mélange de gaussiennes (si le temps le permet) : partitionnement de données (<i>clustering</i>) ; algorithme EM ; fenêtres de Parzen ; EM Spectral Clustering.	5	2		[1] sections 9.2 et 9.4

1. Le cours doit comprendre au moins quatre travaux pratiques couvrant tous les sujets marqués «✓» dans le tableau.
2. Les lectures indiquées ne sont là qu'à titre indicatif. L'enseignant est libre de choisir un autre document de référence.

2 Organisation

Cette section propre à l'approche pédagogique de chaque enseignante ou enseignant présente la méthode pédagogique, le calendrier, le barème et la procédure d'évaluation ainsi que l'échéancier des travaux. Cette section doit être cohérente avec le contenu de la section précédente.

2.1 Méthode pédagogique

Une semaine comprend une à trois heures de cours magistral, ainsi qu'une à trois heures de **vidéo devant être regardées à la maison**. Il est à noter que les vidéos contiennent certaines des parties les plus importantes du cours. Il est donc fondamentalement important que les étudiantes et étudiants suivent les vidéos au rythme mentionné sur le site web du cours. Les présentations magistrales auront pour objectif de clarifier le contenu mathématique des notions introduites dans les vidéos, donner des illustrations concrètes et faire des exercices. Les liens vers les vidéos et les diapositives y étant associées (format PDF) sont disponibles sur le site web du cours. Tout au long de la session, les élèves devront implémenter les notions vues en classe à travers quatre travaux pratiques, ainsi qu'un projet de session pour les étudiantes et étudiants d'IFT712. Des instructions particulières seront données pour chacun des travaux et les périodes de consultation seront déterminées en classe. En ce qui concerne les examens, **seules des notes manuscrites non photocopées sont admises**, tout manuel et notes de cours imprimées étant interdits. L'examen final est récapitulatif.

2.2 Calendrier

Semaine	Date	Thème	Matière vue en classe
1	2024-01-08	1 et 2	Présentation, notation vecteurs-colonnes vs vecteurs-lignes, illustration des probabilités, espérance mathématique et gaussienne, loi gaussienne multivariée, distance de Mahalanobis, matrice de covariance
2	2024-01-15	1	argmin, argmax, entraînement, validation, généralisation, conditionnelle, preuve que l'espérance mathématique d'une gaussienne est μ que sa variance est σ^2 et Σ . Jointe, marginale, Bayes, exemple concret (math vs info), présentation du projet (IFT712)
3	2024-01-22	1 et 3	Jointe, marginale, Bayes, exemple concret (math vs info)(suite). Maximum de vraisemblance (MV) et a posteriori (MAP). Lien avec MV - bruit gaussien et régression L2. Malédiction de la dimensionnalité, preuve que $E[ax+by] = aE[x] + bE[y]$ dans le cas de variables indépendantes.
4	2024-01-29	1 et 4	Lien entre MAP - Bruit et a priori gaussien et régression Ridge, calcul de la dérivée d'une fonction par un vecteur. Illustration de la classification linéaire
5	2024-02-05	3 et 4	Théorie de l'information, entropie, divergence KL, Lagrangien
6	2024-02-12	3 et 5	(suite) Lagrangien et Critère de Fisher. Régression logistique + entropie croisée : lien avec la divergence KL
7	2024-02-19	6 et 7	Révision et autres exemples de MV vs MAP
8	2024-02-26	Examen périodique	
9	2024-03-04	Relâche	

10	2024-03-11	8	Preuve que W est une normale, notion de distance signée, pré-traitement des données, dérivée en chaîne
11	2024-03-18	8	Dérivée en chaîne (suite), rétro-propagation, problèmes avec les fonctions d'activation non centrées à zéro et disparition du gradient.
12	2024-03-25	9	Gradient de la fonction Softmax, initialisation de Xavier, MC Dropout (mesure de confiance) et apprentissage actif
13	2024-04-01	10	Combinaison de modèles + analyse théorique d'Adaboost et de bagging
14	2024-04-08	11	Révision, Algorithmes K-Means (nuées dynamiques) et expectation maximisation
15	2024-04-15	Examen final	
16	2024-04-22	Examen final	

2.3 Évaluation

Travaux pratiques (4)	40 %
Examen intra	20 %
Examen final	40 %

Le cours contient également **quatre (4) travaux pratiques** d'une pondération égale. La pondération de toutes les évaluations est donc la suivante :

IFT603

Travaux pratiques : 40% (10% par TP)

Examen intra : 20%

Examen final : 40%

IFT712

Travaux pratiques : 30% (7.5% par TP)

Examen intra : 15%

Examen final : 30%

Projet : 25%

Détails à l'intention de tout le monde :

TP1 : régression linéaire (Thème principal 4)

TP2 : classification linéaire (Thème principal 5)

TP3 : méthodes à noyau – classification non-linéaire (Thèmes principaux 6,7)

TP4 : réseaux de neurones (Thème principal 8)

IMPORTANT : afin d'éviter toute discrimination, les équipes des travaux pratiques et du projet de session (IFT712) 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.

IMPORTANT : toute personne dont l'implication dans l'**agenda d'équipe** et dans **Git** est jugée trop faible peut se voir **être pénalisée** et même se voir attribuée la **note de 0**.

En toute circonstance :

- Les devoirs doivent être effectués par équipe de **deux ou de trois personnes** ;
- Le code rédigé dans le cadre de ce cours doit se faire dans le langage de programmation **Python**. Le code soumis doit être compatible avec, c'est-à-dire exécutable sous, la version de Python installée dans les laboratoires, sous Linux - Ubuntu ;
- Il est fortement recommandé de configurer votre **ordinateur personnel** et de vous en servir pour effectuer les travaux pratiques et le projet. Si vous avez **Windows**, il est suggéré d'installer **VMware** (gratuits pour les étudiantes et étudiants) et une **VM Ubuntu** (disponible dans le répertoire public).

- La qualité du français et de la présentation peut être considérée lors de l'évaluation des travaux ;
- Prendre note que des erreurs dans le "**style de programmation**" peuvent mener à des pertes de points. Par exemple, un manque de commentaires, du code dupliqué, une fonction inutilement complexe, de code en commentaire, des fonctions définies, mais non utilisées, etc.

- Une mauvaise utilisation de git pourra entraîner une **perte de points aux TP3 et TP4 ainsi que pour le projet (IFT 712)**.

Projet de session (IFT712)

Les étudiantes et étudiants inscrits au cours IFT 712 sont tenus de faire un projet de session en python par équipe de deux ou trois personnes. Le projet a pour objectif de tester au moins **six méthodes différentes de classification** sur une base de données Kaggle (www.kaggle.com) avec la bibliothèque Sklearn (scikit-learn.org). Les équipes sont libres de choisir la base de données de leur choix, mais une option simple est celle du challenge de classification de feuilles d'arbres (kaggle.com/c/leaf-classification). Pour ce projet, on s'attend à ce que les bonnes pratiques de validation croisée et de recherche d'hyperparamètres soient mises de l'avant pour identifier la meilleure solution possible pour résoudre le problème. Au final, on s'attend à ce que vous fassiez preuve d'une bonne démarche scientifique en analysant bien vos résultats, en testant différentes approches de pré-traitement des données et de post-traitement des résultats. La pire chose que vous pourriez faire serait d'appliquer à l'aveugle les fonctions de sklearn.

Le barème de correction est le suivant :

- Qualité du code - Commentaires /10
- Choix de design /10
- Gestion de projet (Git) /10
- Rapport
 - Démarche scientifique /50
 - Analyse des résultats /20

Qualité du code et commentaires

On vous demande du code rédigé 100 % en anglais ou en français, mais **pas de français** ! Votre code doit être bien documenté, respecter le standard pep8 et respecter un standard uniforme pour la nomenclature des variables, des noms de fonctions et des noms de classes. Évitez également les variables « hard codées » empêchant l'utilisation de votre programme sur un autre ordinateur que le vôtre.

Choix de design

Vous devez organiser votre code de façon professionnelle. Pour ce faire, on s'attend à une hiérarchie de classes cohérente, pas seulement une panoplie de fonctions disparates. Aussi, du code dans un script « qui fait tout » se verra automatiquement attribuer la note de zéro. Bien que non requis, on vous encourage à faire un design de classes avant de commencer à coder. Aussi, le code, les données et la documentation doivent être organisés suivant une bonne structure de répertoires. Pour vous aider, vous pouvez utiliser le projet « cookie cutter » (<https://github.com/audreyr/cookiecutter>). La solution proposée doit aussi être facile à utiliser.

Gestion de projet

Comme tout projet qui se respecte, vous devez utiliser un gestionnaire de version de code. On vous demande d'utiliser « git » via la plateforme « gitLab ». On s'attend également à ce que vous fassiez une bonne utilisation de git. Par exemple : évitez de « pousser » du code dans le master sans « merge », éviter les « méga » commits, etc. L'utilisation d'une plateforme de type « trello.com » est recommandée mais non obligatoire.

Démarche scientifique

Pour ce volet, vous devez vous poser les questions suivantes :

- avez-vous bien « cross-validé » vos méthodes ?
- Avez-vous bien fait votre recherche d'hyperparamètres ?
- Avez-vous entraîné et testé vos méthodes sur les mêmes données ?
- Avez-vous testé différentes méthodes de pré- et de post-traitement ?
- * Êtes-vous en mesure d'expliquer pourquoi une méthode est meilleure qu'une autre ?
- Est-ce que tout cela transparait dans le rapport ?

2.3.1 Qualité de la langue et de la présentation

Conformément à l'article 17 du règlement facultaire d'évaluation des apprentissages² l'enseignante ou l'enseignant peut retourner à l'étudiante ou à l'étudiant tout travail non conforme aux exigences quant à la qualité de la langue et aux normes de présentation.

²https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants_actuels/Etudiants_actuels/Informations_academiques_et_reglements/2017-10-27_Reglement_facultaire_-_evaluation_des_apprentissages.pdf

2.3.2 Plagiat

Le plagiat consiste à utiliser des résultats obtenus par d'autres personnes afin de les faire passer pour sien et dans le dessein de tromper l'enseignante ou l'enseignant. Vous trouverez en annexe un document d'information relatif à l'intégrité intellectuelle qui fait état de l'article 9.4.1 du Règlement des études³. Lors de la correction de tout travail individuel ou de groupe une attention spéciale sera portée au plagiat. Si une preuve de plagiat est attestée, elle sera traitée en conformité, entre autres, avec l'article 9.4.1 du Règlement des études de l'Université de Sherbrooke. L'étudiante ou l'étudiant peut s'exposer à de graves sanctions qui peuvent être soit l'attribution de la note E ou de la note zéro (0) pour un travail, un examen ou une activité évaluée, soit de reprendre un travail, un examen ou une activité pédagogique. Tout travail suspecté de plagiat sera transmis au Secrétaire de la Faculté des sciences. Ceci n'indique pas que vous n'avez pas le droit de coopérer entre deux équipes, tant que la rédaction finale des documents et la création du programme restent le fait de votre équipe. En cas de doute de plagiat, l'enseignante ou l'enseignant peut demander à l'équipe d'expliquer les notions ou le fonctionnement du code qu'elle ou qu'il considère comme étant plagié. En cas d'incertitude, ne pas hésiter à demander conseil et assistance à l'enseignante ou l'enseignant afin d'éviter toute situation délicate par la suite.

2.4 Échéancier des travaux

Toute soumission en retard implique une perte de **10 points sur 100 par jour** (incluant samedi et dimanche), sauf celles motivées par des raisons valables et conformes au règlement des études (par exemple, maladie avec attestation d'un médecin).

2.5 Utilisation d'appareils électroniques et du courriel

Selon le règlement complémentaire des études, section 4.2.3⁴, l'utilisation d'ordinateurs, de cellulaires ou de tablettes pendant une prestation est interdite à condition que leur usage soit explicitement permise dans le plan de cours.

Dans ce cours, l'usage de téléphones cellulaires, de tablettes ou d'ordinateurs est autorisées. Cette permission peut être retirée en tout temps si leur usage entraîne des abus.

Tel qu'indiqué dans le règlement universitaire des études, section 4.2.3⁵, toute utilisation d'appareils de captation de la voix ou de l'image exige la permission de la personne enseignante.

Note : Je ne réponds à aucun courriel.

3 Matériel nécessaire pour l'activité pédagogique

Voir page web du cours

4 Références

- [1] BISHOP, CHRISTOPHER M. : *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.
- [2] DUDA, RICHARD O. AND HART, PETER E. AND STORK, DAVID G. : *Pattern Classification*. Wiley, New York, 2^e édition, 2001.
- [3] MURPHY, KEVIN P. : *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013.

³<https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/>

⁴https://www.usherbrooke.ca/sciences/fileadmin/sites/sciences/documents/Etudiants_actuels/Etudiants_actuels/Informations_academiques_et_reglements/Sciences_Reglement_complementaire.pdf

⁵<https://www.usherbrooke.ca/registraire/droits-et-responsabilites/reglement-des-etudes/>

L'intégrité intellectuelle passe, notamment, par la reconnaissance des sources utilisées. À l'Université de Sherbrooke, on y veille!

Extrait du Règlement des études (Règlement 2575-009)

9.4.1 DÉLITS RELATIFS AUX ÉTUDES

Un délit relatif aux études désigne tout acte trompeur ou toute tentative de commettre un tel acte, quant au rendement scolaire ou une exigence relative à une activité pédagogique, à un programme ou à un parcours libre.

Sont notamment considérés comme un délit relatif aux études les faits suivants :

- a) commettre un plagiat, soit faire passer ou tenter de faire passer pour sien, dans une production évaluée, le travail d'une autre personne ou des passages ou des idées tirés de l'œuvre d'autrui (ce qui inclut notamment le fait de ne pas indiquer la source d'une production, d'un passage ou d'une idée tirée de l'œuvre d'autrui);
 - b) commettre un autoplagiat, soit soumettre, sans autorisation préalable, une même production, en tout ou en partie, à plus d'une activité pédagogique ou dans une même activité pédagogique (notamment en cas de reprise);
 - c) usurper l'identité d'une autre personne ou procéder à une substitution de personne lors d'une production évaluée ou de toute autre prestation obligatoire;
 - d) fournir ou obtenir toute aide non autorisée, qu'elle soit collective ou individuelle, pour une production faisant l'objet d'une évaluation;
 - e) obtenir par vol ou toute autre manœuvre frauduleuse, posséder ou utiliser du matériel de toute forme (incluant le numérique) non autorisé avant ou pendant une production faisant l'objet d'une évaluation;
 - f) copier, contrefaire ou falsifier un document pour l'évaluation d'une activité pédagogique;
- [...]

Par plagiat, on entend notamment :

- Copier intégralement une phrase ou un passage d'un livre, d'un article de journal ou de revue, d'une page Web ou de tout autre document en omettant d'en mentionner la source ou de le mettre entre guillemets;
- reproduire des présentations, des dessins, des photographies, des graphiques, des données... sans en préciser la provenance et, dans certains cas, sans en avoir obtenu la permission de reproduire;
- utiliser, en tout ou en partie, du matériel sonore, graphique ou visuel, des pages Internet, du code de programme informatique ou des éléments de logiciel, des données ou résultats d'expérimentation ou toute autre information en provenance d'autrui en le faisant passer pour sien ou sans en citer les sources;
- résumer ou paraphraser l'idée d'un auteur sans en indiquer la source;
- traduire en partie ou en totalité un texte en omettant d'en mentionner la source ou de le mettre entre guillemets ;
- utiliser le travail d'un autre et le présenter comme sien (et ce, même si cette personne a donné son accord);
- acheter un travail sur le Web ou ailleurs et le faire passer pour sien;
- utiliser sans autorisation le même travail pour deux activités différentes (autoplagiat).

Autrement dit : mentionnez vos sources
