Automne 2021

Faculté des sciences
Département d'informatique

INFO4301 – Apprentissage machine
TP1 *(10pts)*

Instructions importantes – Lire avant de commencer le travail

- Ce travail est **individuel**.
- Vous devez utiliser le notebook **Jupyter** pour réaliser le travail.
- Le notebook **Jupyter** servira de rapport (ne pas effacer vos résultats), donc il faut : soigner sa **qualité,** diviser en **sections** avec **titres,** faire une **analyse détaillée des résultats**.
- La note tient compte du respect des instructions, qualité du livrable (code, commentaires, texte, etc.), résultats, démonstration du travail/code (fonctionnel sous **Google Colab**) et réponses aux questions.
- Soignez la qualité de votre texte (vous pouvez utiliser un correcteur orthographique[1] et les services de la CAF[2]).
- Vous devez lire tout le texte et effectuez correctement tout le travail demandé pour obtenir une note complète.

- **Important : Pour les exercices, il est important d'utiliser ce que vous avez appris lors du cours sur les données, caractéristiques, etc. pour s'assurer d'obtenir un bon modèle**

[1] https://support.office.com/fr-fr/article/choisir-les-options-de-correction-automatique-pour-la-mise-en-majuscules-l-orthographe-et-les-symboles-e7433b94-f3de-4532-9dc8-b29063a96e1f
[2] http://www.umoncton.ca/umcm-caf/node/61

1. Régression linéaire multivariée (Python) - 5 pts

L'objectif de cette partie est de programmer une régression linéaire multivariée en utilisant le langage **Python** et **Jupyter** Notebooks.

Pour résoudre le problème de régression multivariée (non régularisée), il faut programmer les fonctions ci-dessous pour trouver les paramètres θ_i :

- **CalculHypothèse** : $h_\Theta(x) = \Theta^T x$
- **FonctionDeCout** : $J(\theta_i)$
- **DescenteGradient** : $\theta_i = \theta_i - \alpha \ \partial/\partial\theta_i \ (J(\theta_i))$

1. Développer les fonctions ci-dessus et développer l'algorithme de régression linéaire multivariée.
2. Utilisez tous les données de **data1.csv** (1ere et 2ème colonne contiennent les variables x_i. La dernière colonne c'est l'étiquette y).
3. Pour le taux d'apprentissage :
    a. Il faut tester plusieurs taux 0.3, 0.1, 0.03, 0.01, 0.003, 0.001 avec seulement 50 itérations.
    b. Une fois le meilleur taux obtenu (celui qui a le coût le plus bas), utilisez-le avec une itération de 1500 fois.
    c. Utilisez une boucle et un code bien structuré qui fait ces tests (ne pas copier le même code plusieurs fois).
4. Affichez la courbe de coût $J(\theta_i)$ en fonction des 1500 itérations.
5. Affichez les meilleurs paramètres θ_i.

2. Régression avancée (Scikit-Learn) - 5 pts

Dans cette partie avec les données **data2_...csv** nous allons utiliser des approches de régression avancés pour faire des prédictions. Les colonnes $x1$-$x12$ sont les variables x_i et la dernière colonne y est l'étiquette/prédiction y).

Pour cela nous allons utiliser la régression par forêts aléatoires (Random Forest regressor) et les arbres extra (Extra Trees Regressor) qui utilisent une stratégie de combinaison de plusieurs arbres (différente des forêts aléatoires) pour prendre une décision.

Lisez le fonctionnement de ces deux techniques dans la documentation de Scikit-Learn et voir comment les utiliser dans les exemples fournis.

1. Implémenter ces deux techniques avec les paramètres par défauts et comparer le coût total de chaque technique pour les données d'entrainement (**data2_train.csv**) et ensuite pour les données de test qui ne sont pas utilisés lors de l'entrainement (**data2_test.csv**).
2. Analysez et discutez vos résultats.

3. Livrables - Important

- Le notebook avec le nom suivant : **NI_VotreNom_TP1.ipynb** (la partie en jaune à modifier avec vos informations)
- Les remises doivent être fait sur le site du cours dans la boîte de dépôt correspondante au travail. Je n'accepte aucun autre moyen d'envoi (~~courriel~~, ~~partage~~, etc.)
- Seule la remise en 1 seul fichier est acceptée. Chaque fichier soumis, écrase-le précèdent.
- Le notebook Jupyter servira de rapport pour le travail, il doit être :
    - Divisé en sections avec des titres (**cellules texte**) ;
    - **Code commenté** ;
    - **Sections** d'analyse (**cellules texte**) dans chacun des fichiers ;
    - Etc.
- Il faut que vous soyez prêt à démontrer votre code et l'expliquer au besoin (la note y tient compte).
- Citez **vos sources** dans la dernière cellule texte du Notebook Jupyter. En cas de **plagiat** la **note** est **0** pour tout le travail.

3. Date de remise et retards

Date de remise : Voir sur la boîte de dépôt du TP 1 sur CLIC

Pénalités de retard : -2.5pt par jour de retard supplémentaire (heure de référence 23h30).