ULaval **IFT&GLO**

Mini-Projet #1 (sur 20%)

à rendre le 24 juin 2020 à 23h55mn

Professeur Brahim Chaib-draa *GLO-7050 : Apprentissage machine en pratique*

Les projets du cours sont les projets des versions du cours COMP 551, cours donné par les collègues de McGill, un grand merci à eux pour nous avoir donné l'autorisation de les utiliser.

Préambule

- Ce mini-projet constitue un travail individuel. Toutefois, ceci nous vous empêche pas de discuter avec les autres étudiants qui suivent le cours. En aucun cas cependant vous ne devez reprendre le code et l'écrit d'autrui ; il vous est demandé d'élaborer les vôtres.
- Si vous "empruntez" des idées, méthodes, démarches ou autres, merci d'indiquer vos sources dans le rapport.
- Il vous est fortement suggéré d'argumenter et/ou justifier vos réponses.
- Après la date de remise, vous avez jusqu'à une semaine pour remettre votre travail avec une pénalité de 30%.
- Vous êtes libres d'utiliser les librairies telles que Numpy ou Scipy pour Python. Toutefois, vous ne devez pas utiliser des implémentations pré-existantes des algorithmes demandés, vous devez les implémenter par vous même.
- si vous avez des questions concernant le travail, merci de passer par le Forum, en posant clairement vos questions.

1 Sélection de modèle

Pour cette expérimentation, vous devez utiliser le Dataset-1 contenant des fichiers pour l'entraînement, la validation et le test. L'entrée et la sortie au niveau de ce dataset sont l'un et l'autre des scalaires à valeur réelle. Le dataset est généré à partir d'un polynôme de degré n et un léger bruit Gaussien est ajouté à la cible (target).

- Ajustez les données à un polynôme de degrée 20. Reportez alors l'erreur MSE (Mean-Square Error) pour l'entrainement et la validation. N'utilisez aucune régularisation. Visualisez l'ajustement modèle-données et commentez le.
- Ajoutez maitenant une régularisation $L2$ à votre modèle. Sachant que λ varie de 0 à 1 tracez alors pour différentes valeurs de λ le MSE pour l'entraînement et la validation. Choisissez alors la meilleure valeur de λ et donnez alors la performance-test pour le modèle considéré. Visualisez l'ajustement modèle-données et commentez le. item Quel est selon vous le degré du polynôme ? Peut-on l'inférer à partir de la visualisation de la question précédente ?

2 Descente de gradient pour la régression

Pour cette expérimentation, vous devez utiliser Dataset-2 contenant des fichiers pour l'entraînement, la validation et le test. L'entrée et la sortie au niveau de ce dataset sont l'une et l'autre des scalaires à valeur réelle.

- Mettre en place un modèle de régression linéaire pour ce jeu de données à l'aide de descente de gradient stochastique. Vous devez utiliser le online-SGD (Stochastic Gradient Descente) (i.e., SGD-avec un exemple à la fois). Utiliser une taille de pas (step size) de $1e-6$. Calculez le MSE pour l'ensemble de validation et ce pour chaque époque.
- Essayez différentes tailles de pas et choisir la meilleure en utilisant le fichier validation. Donnez alors le MSE pour le test.
- Visualisez l'ajustement de la régression pour chaque époque en donnant 5 visualisations qui montrent comment l'ajustement de la régression evolue durant le processus d'entraînement.

3 Dataset issu de la "vie réelle"

Pour cette question, vous devez utiliser le data set "Communities and Crime" qui peut être extrait de UCI : (`http://archive.ics.uci.edu/ml/datasets/Communities+and+Crime`).

- Le dataset étant issu de la vie réelle et comme tel il n'aurait probablement pas les "bonnes" propriétés auxquelles on pourrait s'attendre. Dès lors, votre premier travail consiste à rendre le dataset *utilisable* en complétant toutes les valeurs omises. Pour chacun des attributs manquants, utiliser la moyenne de l'échantillon de chaque colonne. Est-ce un bon choix ? Que pourriez vous utiliser d'autres ? Si vous avez une meilleure méthode, décrivez la et utilisez la pour compléter les données omises.
- Le dataset maintenant complété, mettez alors en place un modèle de régression. Reporter l'erreur de validation croisée à 5-parties (5-fold cross-validation error) à savoir : reporter le MSE (mean squared error) du meilleur ajustement, data-modèle, réalisé sur les données test, moyenné sur 5 différents partage des données 80-20 ; ainsi que les paramètres appris pour chacun des 5 modèles.
- Utilisez maintenant la régression Ridge avec le dataset ci-dessus. Répétez l'expérimentation pour différentes valeurs de λ et reportez le MSE pour chacune des valeurs, sur les données test, moyenné sur 5 différents partage des données 80-20 ; ainsi que les paramètres appris. Laquelle des valeurs de λ donne le meilleur ajustement ? Est-il possible d'utiliser l'information obtenue au cours de cet expérimentation pour sélectionner des caractéristiques ? Si oui, quel est le meilleur ajustement que vous pourriez réaliser avec un ensemble de caractéristiques réduit ?

Partage des données 80-20 : Quelques suggestions

1. Make 5 different 80-20 splits in the data and name them as *CandC-train<num>.csv* and *CandC-test<num>.csv*.

2. For all 5 datasets that you have generated, learn a regression model using the 80% data and test it using 20% data.
3. Report the average MSE over these 5 different runs.

Instructions pour soumettre le code

1. Submittez svp un simple folder zippé portant votre Nom.
2. En utilisant Python, la solution soumise devrait être du genre Jupyter Notebook.
3. Assurez vous que tous les fichiers permettant de faire rouler votre code, sont fournis avec les "bons" chemins d'accès. On devrait faire rouler votre code sans aucune modification.

Instructions pour soumettre le rapport

1. Votre rapport ne doit pas être verbeux, juste l'essentiel. Quand on vous demande des commentaires, vous ne devez pas aller au delà de 3 à 4 lignes pour chacun de ces commenatires.
2. Reportez toutes les visualisations dans la mesure du possible (courbes, graphes, ajustements, etc.).
3. Soit vous utilisez des couleurs pour séparer les différents graphes, soit vous utilisez différents symboles.