Régression Logistique

Descente de gradient stochastique - Programmation parallèle

Création d'un package pour R

Objectif du projet

- Création d'un package proposant
  1. La régression logistique binaire avec la méthode de la descente de gradient stochastique
  2. Avec la possibilité d'exploiter les capacités des processeurs multicœurs
- Package que l'on peut installer directement à partir de GitHub
- Le package intègre un fichier d'aide en anglais aux normes R. c.-à-d. description des fonctions, de leurs paramètres, des objets fournis en sortie, de la lecture des résultats, avec des exemples d'utilisation (voir par ex. ?glm du package stats)
- **A réaliser en groupes de 3 étudiants**

Cahier des charges - Obligatoire

- Implémentation de la régression logistique binaire avec pour algorithme la descente de gradient stochastique. Voici le prototype de la fonction

fit(formula, data, mode, batch_size, ncores, autres param. éventuels)

  - « formula » décrit le problème à résoudre et doit correspondre à la norme de description de R (« y ~ var_1 + var_2 » si on spécifie explicitement les explicatives, ou « y ~ . » si toutes variables disponibles)
  - « data » sont les données à traiter
  - « mode » décrit le mode de mise à jour des coefficients {« batch », « online », « mini_batch »}
  - « ncores » indique le nombre de cœurs à utiliser ; si invalide (<= 0 ou > nombre de cœurs disponibles), les capacités maximales de la machine hôte sont utilisées
  - « autres param. » , par ex. ceux relatifs à la règle d'arrêt
  - La fonction renvoie en sortie un objet de TYPE S3 pour lesquels les méthodes génériques « print » et « summary » au moins doivent être surchargées
  - Remarque :
    - Voir le mode opératoire de glm(.) de R pour caler les attendus de votre fonction
    - Toutes les explicatives sont censées être quantitatives (voir fonctionnalités optionnelles plus loin)

Cahier des charges - Obligatoire

- Implémentation de la fonction de prédiction predict(). Voici le prototype de la fonction

predict(objet Reg. Log., newdata, type)

  - « obj. Reg. Log » est un objet S3 fourni par la fonction fit(.)
  - « newdata » sont les données à traiter, attention un contrôle de cohérence doit être fait
  - « type » indique le type de prédiction {« class », « posterior »} (classe prédite ou probabilité d'appartenance aux classes)

Cahier des charges – Optionnel

A vous de voir, sans qu'il y ait une priorité dans les suggestions ci-dessous…

- Une sortie graphique qui permet d'observer au fil des itérations la décroissance de la fonction de coût. Si on peut l'observer en temps réel (durant le processus d'entraînement du modèle même), ce serait top.
- Préparation intégrée des variables, dont la standardisation des explicatives quantitatives et le codage 0/1 des explicatives qualitatives. Attention, dans ce cas, il faudra également prévoir le traitement adéquat pour la fonction predict(.)
- Introduire la régularisation (Ridge, Lasso, Elasticnet)
- Traitement de la régression logistique multiclasse (variable cible à plus de 2 modalités)
- Fournir un indicateur de pertinence des variables (avec peut-être une sortie graphique ?)
- Proposer une sélection automatique des variables
- Autres… ?

A rendre

- Un rapport en français au format PDF de présentation de votre travail. Il doit être rédigé en LaTeX (source .tex doit être fourni).
- Il doit indiquer les formules et stratégies utilisées pour produire les résultats.
- Il doit décrire également l'architecture de votre programme, modules R, fonctions, détail des objets générés, description de vos implémentations en pseudo-code.

- Le projet doit être hébergé sur GitHub.
- Un tutoriel (reproductible) en anglais montrant l'utilisation des fonctionnalités de votre package doit être disponible sur GitHub (ex. de l'année dernière ClustCheck ; ClusterAnalysis)
- Le package doit pouvoir être installé directement en ligne à partir de GitHub. Il doit comporter les jeux de données exemples utilisés dans le tutoriel.

- Le code source du package et les documents associés (aide, etc.).
- Une copie du package au format ZIP directement utilisable sous R (plan B au cas où l'installation en ligne est défaillante).

Critères d'évaluation

- Qualité et clarté du rapport (en français)
- Qualité de la documentation du package (en anglais)
- Qualité de la programmation – Commentaires / documentation du code source

- Qualité du package, notamment l'installation en ligne
- Rapidité d'exécution et appréhension des grandes volumétries
- Optimisation de l'exploitation des cœurs sollicités / disponibles
- Richesse fonctionnelle (au-delà du cahier des charges obligatoire)
- Utilisabilité (facilité pratique) des fonctions implémentées

Calendrier

- Diffusion du sujet : vendredi 15 octobre 2021
- Retour attendu : vendredi 26 novembre 2021 au soir
- Soutenances : semaine du 06 décembre 2021, à voir

- A faire :
  - Mettre votre projet complet (rapport, package, source, etc.) sur un drive quelconque
  - M'avertir par e-mail et m'envoyer le lien à l'adresse : ricco.rakotomalala@univ-lyon2.fr
  - Sujet : [SISE – Prog. R] Noms des étudiants