

Compte rendu 9 décembre.

Avancement de la semaine du 2 décembre.

- Mise en ordre du GitHub. Unification des bases de données et des notebook Jupyter. Suppression des dossiers redondants et obsolètes.
Problèmes rencontrés: 1) Problèmes de synchronisation entre dossiers locaux et en ligne. Nous n'avions pas pu faire des *push* et *pull* à partir de machines différentes à cause de conflits.
2) Problèmes de compatibilité entre notebooks après une mise en commun.
Message "is not a JSON file".
Solutions apportées: 1) Utilisation de la commande *stash* pour les changements que l'on ne souhaite pas intégrer au dossier Git.
2) Analyse du code source avec un éditeur texte (VSCode) pour régler conflits de versions de JSON.
- Création de l'algorithme C++ pour la génération de la base de données. Mise en commun sur GitHub.
Difficulté rencontrée : problème d'importation de librairie C++ d'utilisation de base de données SDL.
Solution apportée : stockage des données dans un fichier texte en séparant les éléments avec des ";" et les lignes avec des "!" et création d'une routine python permettant de lire ce fichier texte et de l'exporter dans la base de donnée au format .db.
- Réalisation d'un code de test pour la base de données par le procédé suivant: génération de valeurs aléatoires de alpha et de dx comprises dans les intervalles utilisés pour l'entraînement, visualisation des résultats pour alpha fixe et dx fixe. Mise en commun sur le nom "Test_resaux_diff.ipynb".

Point d'avancement sur la diffusion 1D.

Présents dans la réunion: le groupe complet, tuteur Alexis Giaque.

- Description de la base de données avant et après filtrage.
 - Remarque sur la stabilité de chaque point de la base de données. Les points dits "stables" par l'algorithme de génération ne vérifient pas la condition de stabilité du schéma. Une cause possible est le fait que pour certains points (alpha,dx,dt) l'algorithme n'atteint pas l'instant où le calcul plante.
 - Une solution possible est d'ajouter un filtre qui enlève les points pour lesquels la condition de stabilité n'est pas atteinte. Une autre serait de changer le test de stabilité dans le code de génération de la base de données, mais cela sera fait lorsque cela ne représente pas un obstacle pour l'étape suivante.
- Description du réseau et des résultats de l'entraînement.

- La corrélation et le R^2 obtenus avec l'entraînement sont très élevés, c'est-à-dire que le réseau est un modèle vraisemblable par rapport à la base d'entraînement.
- Malgré cela, pour des valeurs alpha et dx générées aléatoirement on obtient des dt au-dessus de la borne de stabilité théorique. Cela est probablement dû, d'un côté, au fait qu'il y ait des données instables dans la base de données, et d'un autre côté, à l'absence d'une pénalisation, intégrée dans l'entraînement du réseau, pour les prédictions qui ne vérifient pas la condition de stabilité.
- Pour les valeurs stables, il y a un écart relativement petit entre la valeur théorique et celle fournie par le réseau.
- Tâches à réaliser.
 - Rédaction d'un rapport de synthèse pour le lot 1, avec le travail réalisé et les résultats obtenus.
 - Prise en main de la discrétisation de l'équation d'advection 2D en lisant les documents mis sur le GitHub et en transcrivant le code SciLab en langage Python.
 - Réalisation des essais pour le schéma centré ($q_1=0, q_2=0$).