

Le projet consiste à programmer en Python l'algorithme de calcul des splines lissantes décrit dans le support de cours, l'appliquer sur des exemples, approfondir des questions plus poussées et à synthétiser l'ensemble dans un rapport. Il doit être réalisé en binôme en utilisant un dépôt `git`.

## Dépôt `git`

Le dépôt `git` doit être situé sur `gitlab.univ-lille.fr`. Il ne doit y avoir qu'un seul dépôt par binôme. Le membre du binôme qui crée le dépôt doit inviter, avec le statut `Maintainer`, l'autre membre du binôme et les trois enseignants suivants : `Francois.Boulier`, `Francois.Lemaire`, `Adrien.Poteaux@univ-lille.fr`.

Pour `git clone`, choisissez `Clone with https`. Pour éviter de trop entrer votre identifiant et votre mot de passe, utilisez : `git config --global credential.helper cache`.

À la fin du projet, le dépôt doit avoir une structure simple et claire. Il doit contenir un rapport au format PDF, vos codes Python et vos fichiers d'exemples. Il doit être propre (pas de fichiers superflus). Évitez les lettres accentuées et les espaces dans les noms de fichiers.

## Le rapport

La note porte essentiellement sur le rapport : rédigez-le pour qu'il donne envie de vous attribuer une bonne note.

Il doit donc montrer clairement le travail que vous avez réalisé et à quoi on voit que les résultats sont corrects. Les analyses doivent montrer que vous avez compris ce que vous faites. Les figures doivent porter une légende : qu'est-ce que la figure nous montre ? Que représentent les axes ? Chaque trace de calcul doit être accompagnée d'instructions permettant au correcteur de reproduire ce calcul avec votre code (pensez à structurer votre code en conséquence).

Dès l'introduction, dites ce que vous avez fait de bien : avez-vous structuré votre code Python avec des techniques sophistiquées ? avez-vous implémenté en détail les différents algorithmes numériques ? avez-vous traité des exemples particulièrement intéressants ? un grand nombre d'exemples ? avez-vous creusé certaines questions plus poussées ?

Pour la technique de rédaction, appliquez la règle dite du « un septième » : *tout texte doit être précédé d'un texte sept fois plus court qui le résume et cela, jusqu'à la phrase initiale*. Commencez donc par écrire une phrase qui indique tout ce qui est important. Reprenez ensuite approximativement sept expressions de cette phrase qui méritent des explications et écrivez une nouvelle phrase pour chacune d'elles. Recommencez si nécessaire pour chacune des sept nouvelles phrases. Vous aurez alors une excellente introduction. Le reste du rapport ne sert plus qu'à justifier les affirmations faites dans l'introduction. Son plan devrait apparaître très naturellement.

Le rapport doit comporter une introduction, une conclusion, une table des matières et des pages numérotées. Le « baratin inutile » (recopie de l'énoncé, du cours etc) sera sanctionné.

## Le code Python

Il doit être écrit dans un fichier texte (les codes écrits sur des *Notebook* ne seront pas corrigés) en utilisant Python 3.

Les bibliothèques Python autorisées sont : `numpy`, `scipy` et `matplotlib`. L'utilisation de toute autre bibliothèque doit faire l'objet d'une autorisation.

Le projet est un projet de calcul numérique. N'hésitez pas à écrire différentes versions des algorithmes : les premières peuvent utiliser les fonctionnalités de haut niveau offertes par `scipy` ; dans les suivantes, entrez dans le détail des calculs en utilisant les algorithmes étudiés en troisième année. Mais n'allez pas jusqu'à implanter les opérations élémentaires de calcul matriciel.

## Les exemples

Commencez par des exemples élémentaires sur lesquels il est facile de vérifier que vos codes sont corrects.

Vous pouvez ensuite traiter des exemples plus compliqués, obtenus sur le site <https://web.stanford.edu/~hastie/ElemStatLearn>. Pensez à décrire les pré-traitements nécessaires pour ces exemples. Le livre [1] peut fournir d'autres sources d'inspiration.

## Remarques diverses

C'est vous qui choisissez la valeur du paramètre de lissage  $p$ . Pensez à expliquer son rôle dans votre rapport.

Les écarts-types  $\sigma_i$  peuvent tous être mis à 1 (au moins dans un premier temps).

Quelques questions supplémentaires que vous pouvez approfondir : qu'est-ce qu'un multiplicateur de Lagrange (trouver un exemple simple avec une seule contrainte d'égalité) ? quelle suite définie par récurrence et condition initiale permet de calculer  $p$  en fonction de  $S$  ? avez-vous réussi à modifier votre implantation pour paramétrer le par le degré de liberté ? parvenez-vous à calculer le paramètre optimal ?

## Références

- [1] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning. Data Mining, Inference and Prediction*. Springer Series in Statistics. Springer, 2nd edition, 2009. Available at <https://web.stanford.edu/~hastie/Papers/ESLII.pdf>.