

Préambule

On considère le problème des moindres carrés. Soit $A \in \mathbb{R}^{m \times n}$ avec $m \geq n$ et $b \in \mathbb{R}^m$. On cherche $x \in \mathbb{R}^n$ qui minimise

$$J(x) = \|Ax - b\|_2^2.$$

1. Montrez que x est solution du problème des moindres carrés ssi l'équation normale

$$A^T A x = A^T b$$

est vérifiée. Montrer ensuite que ce système admet toujours au moins une solution et qu'elle est unique si $\text{rang}(A) = n$.

2. Montrez comment utiliser judicieusement la décomposition QR de la matrice A pour résoudre le problème des moindres carrés dans le cas où $\text{rang}(A) = n$.
3. On considère m points du plan (x_i, y_i) , $i = 1, \dots, m$ de coordonnées x distinctes. Montrez comment trouver le polynôme

$$p_{n-1}(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1}$$

d'ordre $n - 1$ qui approxime au mieux ces m points au sens des moindres carrés.

Le devoir

Echéance

Le travail demandé est un travail individuel. Il consiste en (i) la remise d'un rapport et (ii) la remise d'un fichier unique compilable écrit en langage C standard.

Les implémentations et le rapport sont à soumettre sur le Moodle du cours d'Analyse Numérique pour le mardi 20 octobre 2020.

Le code

Les spécifications de ce code sont les suivantes.

- La seule librairie externe admise est la librairie BLAS.
- Le code sera composé d'un seul fichier avec l'extension `.c`. On pourra le compiler de la façon suivante

```
cc -o mon_executable devoir1.c -lblas -O3
```

Si vous n'utilisez pas BLAS, l'argument `-lblas` sera ignoré à la compilation.

- Le code prendra en entrée trois arguments :
`./mon_executable n fichier.in fichier.out`
 où n est le nombre de coefficients du polynome (qui est donc de degré $n - 1$). Le fichier d'entrée `fichier.in` contient le nombre de point m à la première ligne, puis ensuite la liste des paires de coordonnées $x_i y_i$, une paire par ligne, les deux nombres séparés par un espace. Le fichier de sortie `fichier.out` est le résultat du calcul. Il contient le nombre n des coefficients à la première ligne, puis la liste des coefficients de l'approximation polynomiale $a_0, a_1 \dots a_{n-1}$, un par ligne.
- Il est évidemment nécessaire d'implémenter un certain nombre de choses, dont la plus importante est la décomposition **QR** d'une matrice réelle. Il existe 3 méthodes pour calculer **QR** (Householder, Gram-Schmidt et rotations de Givens). On vous demande de *ne pas utiliser les réflecteurs de Householder*.

Toutes les implémentations seront soumises à un logiciel anti-plagiat.

Le rapport

Le rapport doit être réalisé avec \LaTeX , avec la `documentclass article [11pt]` en `pagestyle plain`. Le `.PDF` et le source `.TEX` de ce rapport sont également à remettre sur Moodle sous forme d'un dossier (`.zip`) compressé.

La longueur maximale du rapport est de 6 pages A4.

Le rapport devra contenir les éléments suivants

- La réponse aux 3 questions théoriques posées dans le Préambule.
- Une explication détaillée de l'algorithme **QR** choisi, ainsi que des autres fonctions nécessaires pour résoudre le problème des moindres carrés.
- Montrez numériquement l'exactitude de votre implémentation de la factorisation **QR**.
- Au moyen d'une expérience numérique que vous décrivez dans ce rapport, évaluez la complexité temporelle de factorisation **QR** appliquée à des matrices aléatoires de relativement grande taille ($m, n < 1000$).
- Illustrez graphiquement quelques résultats de l'approximation polynomiale appliqués, par exemple, à des données réelles et ce dans différents cas de figure ($m = n$, $m > n$, $m \gg n$).
- Bonus : Mettez en évidence le phénomène de Runge.

Les sources des rapports seront soumises à un logiciel anti-plagiat.