Ensimag 2A
Modélisation Géométrique

TP : Approximation aux Moindres Carrés sous contraintes

- Le TP est à effectuer par chaque élève **individuellement**.
- Les réponses aux questions 1,3,5 doivent être incluses dans le code python au début du fichier.
- Remise du code python nommé *votreNOM.py* sur TEIDE au plus tard le **mercredi 22/11/2023**
- Un code Python contenant toutes les routines nécessaires pour la lecture des données, l'évaluation des surface de Bézier tensorielles ainsi que la visualisation est fourni. Il vous est demandé d'implémenter les deux méthodes d'approximation au sense des MC sans et avec contraintes.
- Chaque ligne ajoutée à ce code initial doit être strictement commentée et justifiée.

On considère ici un problème d'approximation aux moindres carrés en dimension deux dans le cas explicite, d'une part sans contrainte (Partie 1) puis avec des contraintes de type égalité (Partie 2).

Partie 1 : Approximation aux Moindres carrés sans contrainte

Dans cette partie on considère le cas sans contraintes. Précisément, on se place dans l'espace $\mathbb{R}^3$ et on considère les données

$$(x_k, y_k, z_k) \quad \text{avec} \;\; (x_k, y_k) \in [0,1]^2,\; z_k \in \mathbb{R},\; 1 \leq k \leq N,$$

où N est un entier suffisamment grand. La valeur z_k est la mesure "*au dessus*" du point de coordonnées (x_k, y_k), de sorte qu'il s'agit d'un ensemble de mesures en des points du carré $[0,1] \times [0,1]$. On admet que ces mesures *échantillonnent l'ensemble* du carré $[0,1]^2$ comme montré sur la figure ci-dessous à droite.

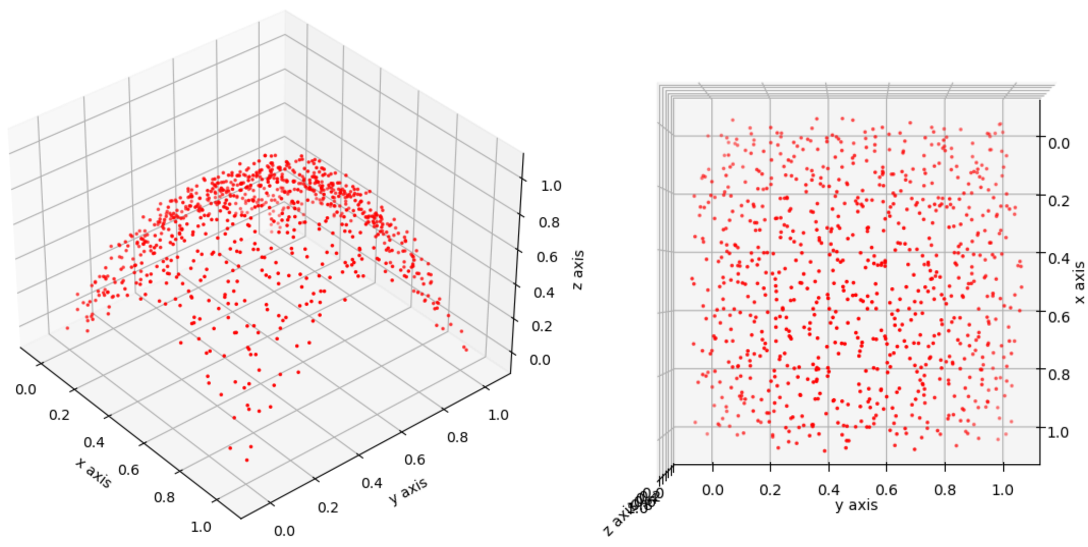

Figure 1 : nuage de points des données 3D – vue "de dessus" à droite

On cherche à approximer ces données par une fonction $h : [0,1] \times [0,1] \to \mathbb{R}$ de la forme

$$h(x,y) = \sum_{i=0}^{n} \sum_{j=0}^{m} \alpha_{ij}\, H_i^n(x)\, H_j^m(y)$$

où les coefficients α_{ij} sont les paramètres réels à déterminer et où pour $p \in \mathbb{N}^*$ et $0 \leq i \leq p$, les fonctions $H_i^p(t)$ sont les polynômes de Bernstein de degré p :

$$H_i^p(t) = \binom{p}{i}\, t^i\, (1-t)^{p-i}$$

Le problème à résoudre est le suivant :

$$\min_{\alpha_{ij}} \quad \sum_{k=1}^{N} \left[h(x_k, y_k) - z_k \right]^2$$

1. Question préliminaire. – On nous dit que le nombre de données N doit être suffisamment grand. Préciser la valeur minimale N_0 acceptable pour cet entier N. Quelle est alors la situation dans le cas $N = N_0$?

2. Récupérer le script python `ApproxTensorBezierSurfaceStudents.py` et déterminer la solution pour les données fournies dans les fichiers `dataSailX.txt, dataSailY.txt, dataSailZ.txt` pour des surfaces Bézier tensorielles de bi-degré (n,m).

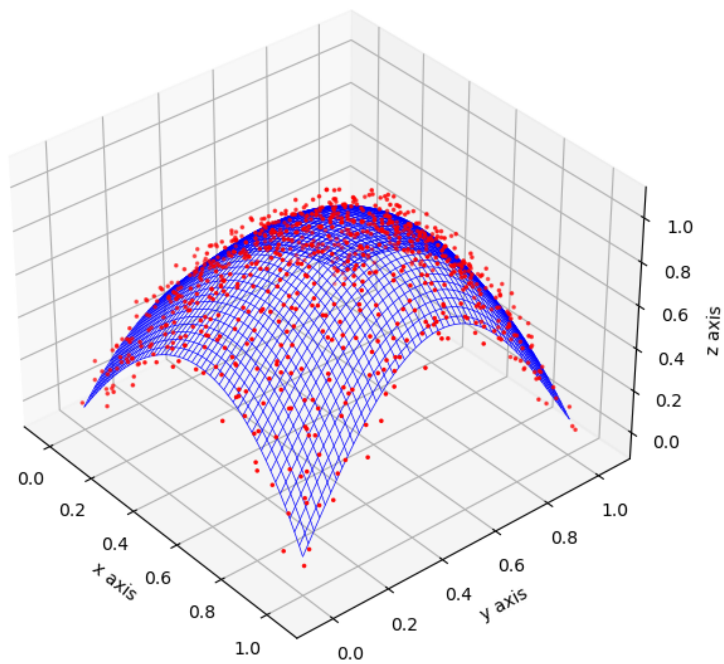

Figure 2 : approximation par une surface de bi-degré (3,4)

Partie 2 : Approximation aux Moindres carrés avec contraintes

On considère maintenant le problème d'approximation aux moindres carrés précédent auquel on ajoute des contraintes d'interpolation aux 4 coins.

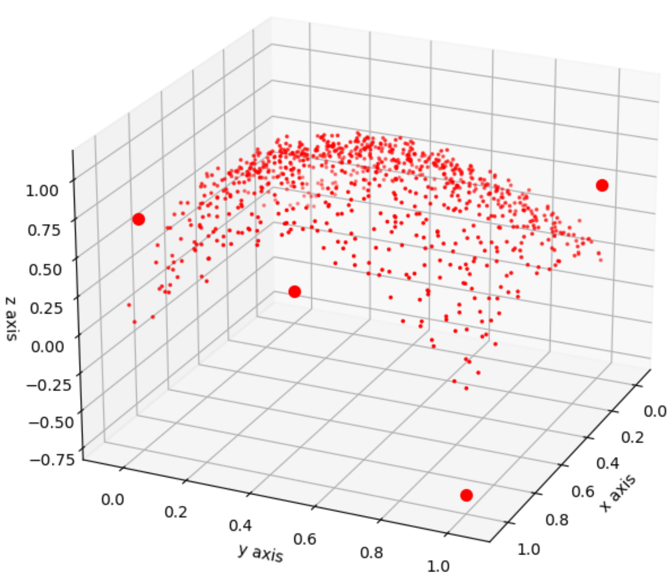

Figure 3 : données avec contraintes aux 4 coins

Précisément, le problème à résoudre est le suivant :

$$\min_{\alpha_{ij}} \quad \sum_{k=1}^{N} \Big[h(x_k, y_k) - z_k \Big]^2$$

sous condition “d’interpolation aux 4 coins”, c’est-à-dire

$$h(0,0) = z_{0,0}\,, \quad h(0,1) = z_{0,1}\,, \quad h(1,0) = z_{1,0}\,, \quad h(1,1) = z_{1,1}.$$

où les valeurs $z_{0,0}$, $z_{0,1}$, $z_{1,0}$, $z_{1,1}$ sont fixées.

3. Question préliminaire. – Préciser dans cette nouvelle situation la valeur minimale N_1 acceptable pour le nombre de données N. Quelle est alors la situation dans le cas $N = N_1$?

4. Compléter votre script python précédent avec les données fournies pour l’interpolation aux coins afin de déterminer l’approximation sous contraintes pour des surfaces Bézier tensorielles de bi-degré (n,m).

5. Préciser la taille du système linéaire à résoudre pour la question précédente (en fonction de N, n et m). Dans ce cas particulier, peut-on envisager une méthode de résolution conduisant à un système linéaire de taille inférieure ?

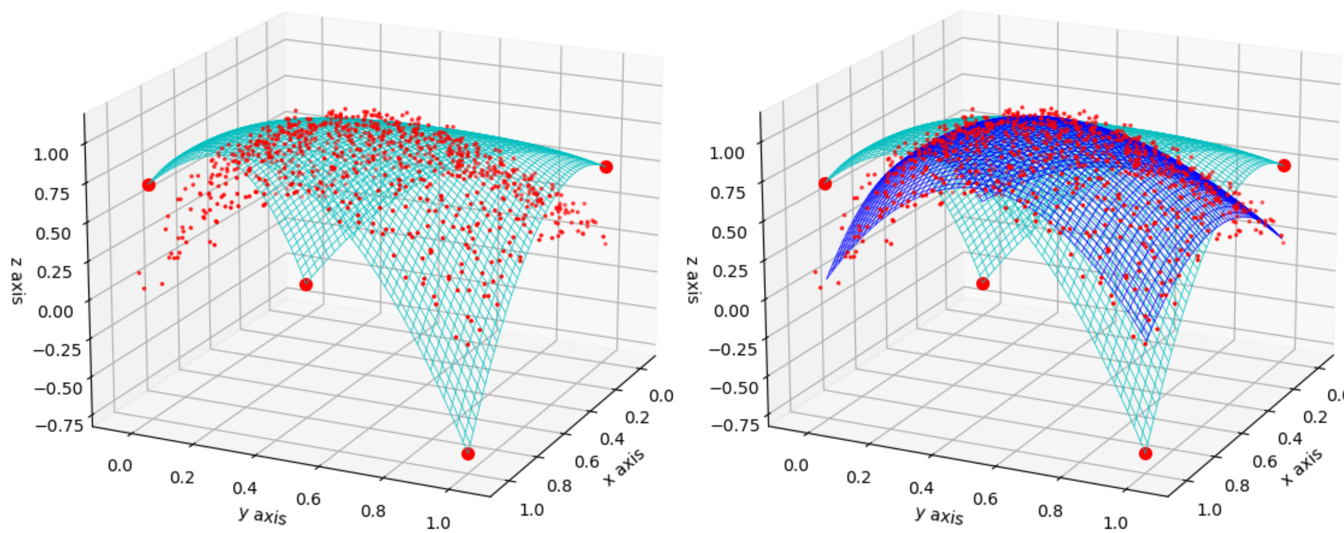

Figure 4 : approximation sous contraintes d’interpolation aux 4 coins par une surface de bi-degré (2,2) — A droite on a ajouté l’approximation sans contraintes en bleu.

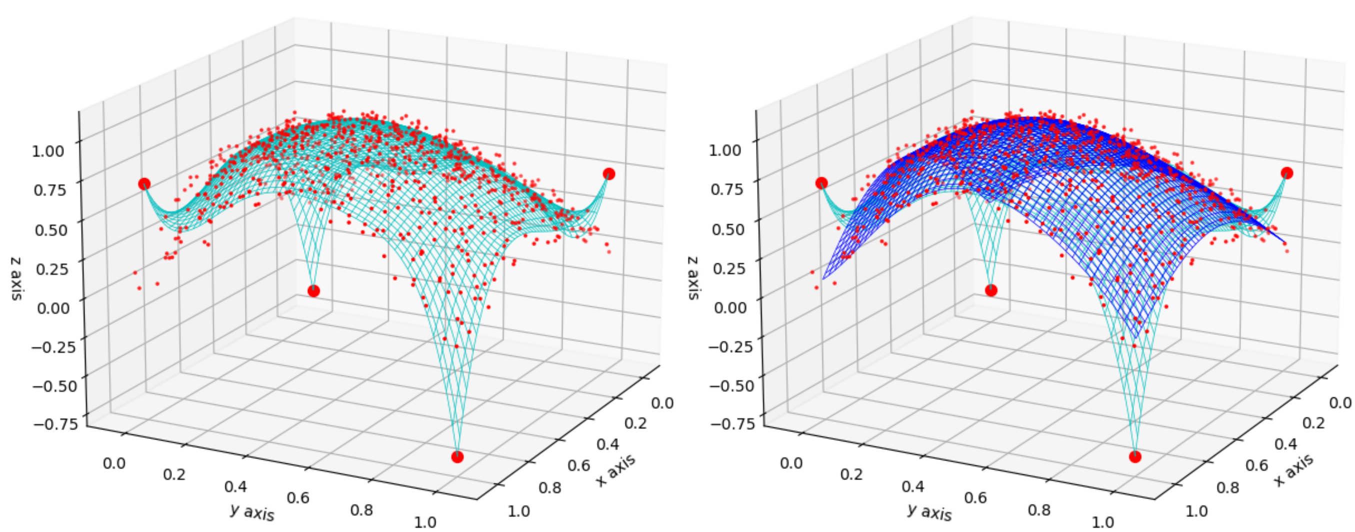

Figure 5 : approximation sous contraintes d’interpolation aux coins par une surface de bi-degré (4,5) — A droite on a ajouté l’approximation sans contraintes en bleu.