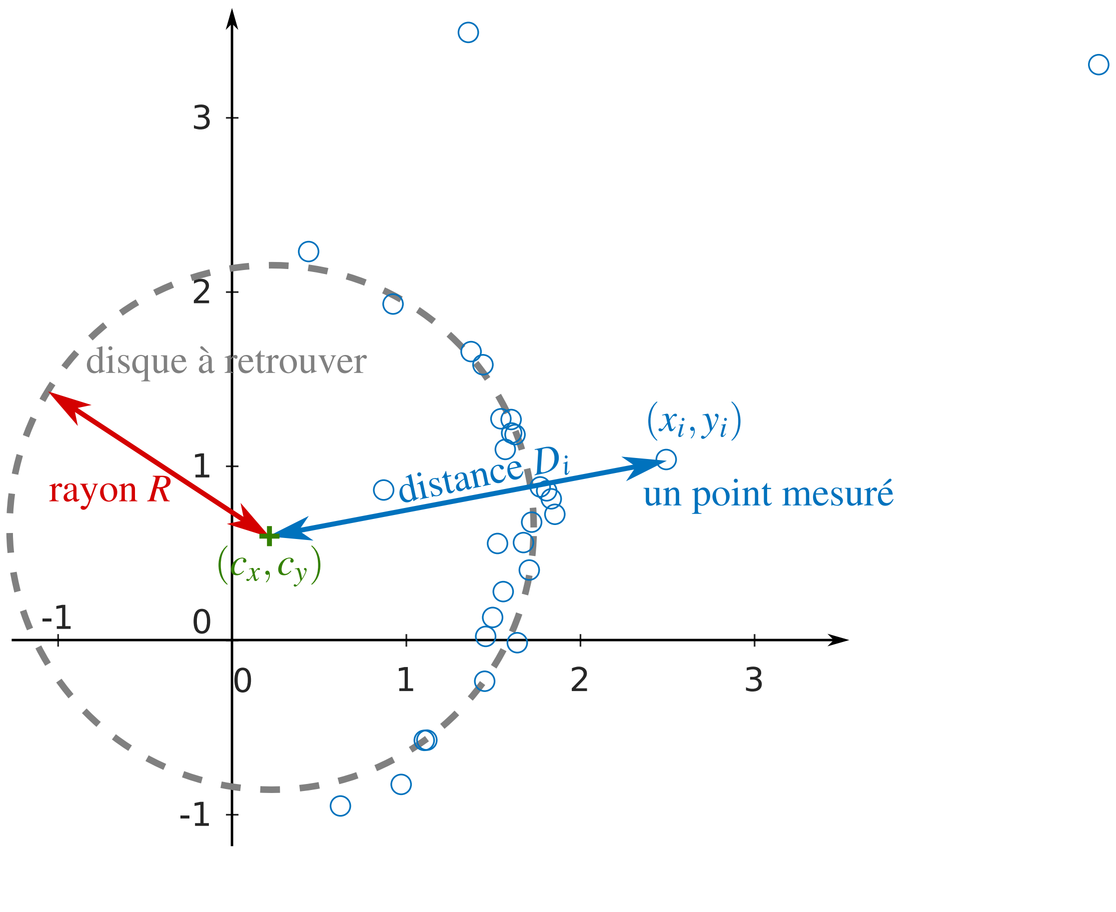
FISE2 – 2022/2023

Projet d'optimisation continue
Estimation robuste du centre d'une pièce circulaire

Le but de ce projet est d'appliquer les méthodes d'optimisation continue à un cas concret: le problème de l'*estimation robuste*, c'est à dire l'estimation en présence de points aberrants (outliers). Nous nous intéressons à une classe d'estimateurs appelés *M-estimateurs* qui sont définis sous la forme d'un problème d'optimisation continue.

1 Description du problème

On souhaite estimer le plus précisément possible le centre d'une pièce mécanique circulaire de rayon $R = 1.5$cm connu à partir de mesures réalisées le long de la pièce. Chaque point de mesure (x_1, y_1) à (x_n, y_n) est entaché d'une erreur. Certains points sont aberrants (fausses détections de la méthode qui extrait les points (x_i, y_i) à partir des images du système de contrôle).



Pour estimer les coordonnées (c_x, c_y) du centre du cercle, nous allons essayer de minimiser les écarts entre les distances $D_i = \sqrt{(x_i - c_x)^2 + (y_i - c_y)^2}$ (distance entre le i-ème point et le centre du cercle) et le rayon R du cercle: en effet, si tous les points mesurés appartenaient au cercle de centre (c_x, c_y) et de rayon R alors les distances D_i seraient toutes égales à R.

2 Estimation au sens des moindres distances carrées (total least squares)

Le premier estimateur du centre (c_x, c_y) du cercle que nous allons étudier est défini par:

$$\hat{\theta} = \begin{pmatrix} \widehat{c_x} \\ \widehat{c_y} \end{pmatrix} = \arg\min_{c_x,\, c_y} \quad \sum_{i=1}^{n} (\underbrace{\sqrt{(x_i - c_x)^2 + (y_i - c_y)^2}}_{D_i} - R)^2 \,. \tag{1}$$

Il est donc calculé en trouvant le minimum d'une fonction de coût $\mathscr{C}_{\mathrm{TLS}}(c_x, c_y)$ représentant la somme du carré de l'écart entre D_i (la distance entre chaque point de mesure et le cercle) et R (le rayon du cercle).

Q1: Représentez la fonction de coût $\mathscr{C}_{\mathrm{TLS}}(c_x, c_y)$ dans l'espace des paramètres (c_x, c_y) pour les intervalles $[-1,\, 1] \times [-1,\, 2]$ puis $[-1,\, 4] \times [-1,\, 4]$. Que remarquez-vous?

Q2: Déterminez par échantillonnage régulier des paramètres c_x et c_y une solution approchée au problème pour chacun des deux domaines représentés à la question 1. Représentez les cercles correspondants ainsi que le nuage des points mesurés (on pourra utiliser la fonction Matlab `viscircle()`). Combien d'évaluations de la fonction de coût sont-elles nécessaires pour déterminer c_x et c_y à 10^{-4} près par cette méthode d'échantillonnage régulier? Si nous devions également estimer R, comment ce nombre d'évaluations serait-il modifié (on supposera $R \in [0.5\text{cm},\, 2.5\text{cm}]$ et visera la même précision à 10^{-4} près)?

Pour estimer précisément les coordonnées (c_x, c_y) de façon beaucoup plus rapide nous allons utiliser des algorithmes d'optimisation continue (cf. chapitre 3 du cours): les algorithmes de plus forte pente, puis de quasi-Newton. Pour cela, nous aurons besoin de pouvoir évaluer la valeur du gradient de la fonction de coût $\mathscr{C}_{\mathrm{TLS}}(c_x, c_y)$.

Q3: Calculez sur le papier l'expression du gradient de la fonction de coût $\mathscr{C}_{\mathrm{TLS}}(c_x, c_y)$.

Q4: Programmez une fonction renvoyant la valeur du gradient de la fonction de coût pour un couple (c_x, c_y) arbitraire grâce à l'expression trouvée à la question précédente. Vérifiez numériquement votre calcul en comparant les valeurs obtenues pour différents couples (c_x, c_y) avec votre fonction et en calculant les dérivées par différences finies.

Q5: Calculez le gradient pour des couples (c_x, c_y) régulièrement échantillonnés puis représentez-le comme un champ de vecteurs avec la fonction Matlab `quiver` ainsi que les lignes de niveaux de la fonction de coût. Vérifiez que le gradient est orthogonal aux lignes de niveau (important: il faut utiliser un repère orthonormé avec la commande `axis equal`).

Q6: Implémentez la méthode des plus fortes pentes en réalisant l'étape de recherche linéaire (ligne `arg min` de l'algorithme décrit en page 13 du chapitre 3) avec l'algorithme de Fletcher-Lemaréchal. On affichera:

1. la suite des itérés dans le plan (c_x, c_y)
2. l'évolution de la fonction de coût au cours des itérations
3. l'évolution de la norme du vecteur gradient au cours des itérations
4. la distance entre deux itérés successifs (c'est à dire entre $(c_x^{(k)}, c_y^{(k)})$, à l'itération k, et $(c_x^{(k+1)}, c_y^{(k+1)})$, à l'itération suivante).
5. la distance à la solution (approximée par le résultat de l'algorithme pour un nombre très élevé d'itérations, par exemple 1000 itérations)

Q7: Répétez cette étude pour d'autres points de départ (valeur initiale des paramètres c_x et c_y). Interprétez.

Q8: Appliquez la méthode de quasi-Newton à cette fonction de coût. Représentez, comme à la question 6, les courbes illustrant la convergence de la méthode. Commentez et comparez aux résultats obtenus avec la méthode des plus fortes pentes.

Afin d'améliorer la qualité de l'estimation du centre du cercle, on propose de réaliser la pénalisation de l'écart entre D_i et R par une fonction à croissance lente: $\frac{1}{2}\log(1+\frac{\bullet^2}{\sigma^2})$ au lieu de la fonction carré: $\bullet^2$.

Q9: Représentez les valeurs de cette nouvelle fonction de coût en fonction de c_x et c_y pour $\sigma = 10^{-3}$, $\sigma = 0.1$ et $\sigma = 10$. Représentez dans chaque cas le cercle correspondant au minimum approché de cette fonction de coût.

Q10: Reprendre les questions 3 à 7 pour cette nouvelle fonction de coût.