Travaux pratiques

TP1 : Méthode de gradient projecté

I- Écrire une fonction Python `DGPFK`(f, P_K, X_0, τ, N_{max}, ε) qui renvoie la valeur de X_n générée par la méthode de résolution.

Argument	Description
f	Fonction à optimiser
X_0	Valeur initiale
P_K	Opérateur de projection sur K
τ	Pas de l'algorithme
N_{max}	Nombre maximal d'itérations
ε	Précision

II- Etude numérique : Soient

$$f(x,y) = x^2 + y^2; \quad K = \{(x,y) \in \mathbb{R} \times \mathbb{R} / x + y = 1\}.$$

1) Montrer que f admet une seul solution seul un point minimum X^* sur K.
2) Montrer par le théorème de Lagrange que le problème de minimisation admet une seule solution $X^* = \left(\frac{1}{2}, \frac{1}{2}\right)^t$.
3) Verifier que l'opérateur de projection orthogonal P_K est bien défini, puis déterminer le.
4) Montrer que l'algorithme du gradient à pas fixe avec projection appliqué à notre problème converge en précisant des bons choix théoriques du pas
5) Implémentation :

- Tracer les lignes de niveau de f dans le rectangle $[-3, 3] \times [-3, 3]$.
- Implémenter la méthode de gradient à pas fixe avec projection pour résoudre le problème $\arg\min\limits_{x \in K} f(x)$.
- Proposer au moins deux critères d'arrêt pour l'algorithme et discuter de leur pertinence.
- Tracer graphiquement les itérées de la méthode de gradient à pas fixe avec différent points initials et valeurs du pas.
- Interpréter les résultats.

6) Discuter les résultas numériques si en remplace f par t $h(x,y) = \frac{1}{2}x^2 + \frac{1}{2}\beta y^2$ (où $\beta > 1$) selon les valeurs du β

TP 2 : Méthode de pénalisation

I- Écrire une fonction Python `PENALISATION`(f, g, X_0, τ, N,$\mathcal{N}$, λ, ε) qui résout le problème de minimisation en renvoyant la valeur de X_n générée par la méthode de pénalisation

Argument	**Description**
f	Fonction objectif à minimiser
g	Fonction de contrainte
X_0	Valeur initiale
τ	Pas de l'algorithme
N_{max}	Nombre maximal d'itérations
$\mathcal{N}$	fonction de pénalisation extérieure
λ	Paramètre de pénalisation
ε	Précision

II- Réalisez une étude numérique similaire à celle effectuée dans le TP1, appliquée au même problème traité. Cette étude devra inclure :

- Une analyse de la méthode choisie.
- L'évaluation des performances numériques en fonction des paramètres (comme le pas de l'algorithme, la précision, etc.).
- Une comparaison des résultats avec les théories précédemment abordées, accompagnée de graphiques ou de tableaux.

TP3 : Méthode d'Uzawa (Méthode des multiplicateurs de Lagrange)

I- Écrire une fonction Python `UZWA`(f, g, X_0, τ, N, ε) qui résout un problème d'optimisation différentiable de sous contrainte, en renvoyant la valeur de X_n générée par la méthode d'UZwa

Argument	**Description**
f	Fonction objectif à minimiser
g	Fonction de contrainte
X_0	Valeur initiale
τ	Pas de l'algorithme
N_{max}	Nombre maximal d'itérations
ε	Précision

II- Réalisez une étude numérique similaire à celle effectuée dans le TP1, appliquée au même problème traité. Cette étude devra inclure :

- Une analyse de la méthode choisie.
- L'évaluation des performances numériques en fonction des paramètres (comme le pas de l'algorithme, la précision, etc.).
- Une comparaison des résultats avec les théories précédemment abordées, accompagnée de graphiques ou de tableaux.