MAT-2930 Algèbre Linéaire Appliquée **TP 4 – (Pondération : 6 %)**
Automne 2020

TP 4 – Valeurs propres, vecteurs propres (6 %)

CONSIGNES

1. Les sections identifiées par **(Matlab)** doivent être développées dans les scripts ou fonctions Matlab.
2. Dans vos scripts, utilisez la commande % pour placez les commentaires nécessaires à la compréhension par les correcteurs de ce que vous aurez programmé. Placez aussi le numéro des questions auxquelles vous répondez.
3. **Affichage -** Toute demande d'affichage obligatoire dans les questions **(Matlab)** signifie un affichage dans la fenêtre commande. À cette fin, utiliser la commande *display* ou la commande *fprintf.*
4. N'oubliez pas d'écrire vos deux noms ainsi que le numéro de votre équipe au début des scripts. Sinon un malus de 30 points sera compté.
5. Les soumissions en retard auront la note zéro.
6. Les données numériques des deux problèmes sont disponibles dans les deux fichiers Excel **Scores_Sij** et **Matrice_A**.

CONTENU DE VOTRE SOUMISSION SUR MON PORTAIL ENA

Votre soumission doit être réalisée sous la forme d'un fichier compressé (e.g. format zip). Le nom de ce fichier doit être **Equipe_##** (pour l'équipe 12 par exemple, ça sera Equipe_12).

Ce fichier compressé doit contenir :

1. Script Matlab : **classement.m** pour répondre au Problème 1
2. Scriptz Matlab : **localiser.m** et **puissance.m** pour répondre au Problème 2

Problème 1 – (40 points) | Classement des équipes d'un championnat de football

Un championnat de football a eu lieu entre huit équipes. Les résultats sont consignés dans le tableau ci-dessous où S_{ij} représente le nombre de points marqués par l'équipe i contre l'équipe j.

Tableau des points marqués S_{ij}

Équipe j → i↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	0	14	3	17	24	0	35	2
2	7	0	14	31	45	2	29	28
3	7	31	0	42	7	17	7	34
4	12	10	34	0	20	31	12	14
5	27	28	35	27	0	14	15	20
6	3	24	41	7	41	0	13	35
7	38	23	27	13	31	17	0	49
8	3	16	30	14	13	8	35	0

L'objectif ici est d'établir le classement des équipes par ordre décroissant de compétitivité suite aux résultats du championnat. Pour ce faire, on associe une variable de classement x_i positive à chaque équipe i ($1 \leq i \leq 8$) telle que $x_i > x_j$ indique que l'équipe i est mieux classée que l'équipe j.

Afin d'établir un classement le plus juste possible, on impose que le classement de l'équipe i soit proportionnel à la somme pondérée des classements des sept équipes restantes :

$$x_i = k \sum_{j=1}^{8} r_{ij}\, x_j\ , \quad 1 \leq i \leq 8$$

k étant la constante de proportionnalité et r_{ij} sont les pondérations.

Méthode de classement 1

Il y a diverses façons possibles de définir les poids r_{ij}, la plus simple étant de fixer $r_{ij} = 1$ si l'équipe i gagne devant l'équipe j, sinon $r_{ij} = 0$.

a) **(Matlab).** Calculer les variables de classement, x_i.
b) **(Matlab).** Ranger les valeurs x_i de la plus grande à la plus faible.
c) **(Matlab).** Afficher le classement des huit équipes par un commentaire.

Méthode de classement 2

Le problème de la méthode de classement précédente est que l'équipe perdante n'en tire aucun bénéfice même si elle perd un match avec une différence de points minime, l'équipe gagnante n'en tire aucun bénéfice non plus même si elle gagne un match avec une différence de points importante. Une approche plus équitable serait donc de définir les poids r_{ij} en fonction des scores S_{ij}. Pour ce faire, on définit chaque poids r_{ij} comme le rapport des points marqués par l'équipe i contre l'équipe j (i.e. S_{ij}) au nombre de points total (i.e. $S_{ij} + S_{ji}$) marqués lors du match opposant les équipes i et j :

$$r_{ij} = \begin{cases} \dfrac{S_{ij}}{S_{ij}+S_{ji}} & , \quad i \neq j \\ 0, & \quad i = j \end{cases}$$

d) **(Matlab).** Calculer les variables de classement, x_i.
e) **(Matlab).** Ranger les valeurs x_i de la plus grande à la plus faible.
f) **(Matlab).** Afficher le nouveau classement des huit équipes par un commentaire.

Problème 2 – (60 points) **| Méthode numérique de la puissance inverse décalée**

Soit la matrice : $A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 2 & 5 & 0 & 0 \\ 1/2 & 0 & 3 & 1/2 \\ 0 & 0 & 3/4 & 7 \end{bmatrix}$

PARTIE I

1. **(Matlab).** Écrire un script ***localiser.m*** pour dessiner sur un même plan complexe tous les disques de Gerschgorin de la matrice A. Sauvegarder la figure.

2. **(Matlab).** Visualiser le graphe précédent pour cadrer le plus précisément possible les valeurs propres λ_i de A. Bien justifier les bornes pour chaque valeur propre. Répondre en affichant des commentaires.

PARTIE II

Écrire un script ***puissance.m*** pour répondre aux questions 3 à 5 ci-dessous.

3. **(Matlab).** Calculer les valeurs propres de la matrice A par la méthode itérative de la puissance inverse décalée. Pour ce faire, utiliser les résultats du point 2 pour donner une approximation initiale, α_i, à chaque valeur propre λ_i. Le calcul de chaque valeur propre doit s'arrêter automatiquement dès qu'une précision de 10^{-3} est atteinte. Afficher sous la forme d'un tableau les valeurs propres λ_i et les approximations initiales α_i.

4. **(Matlab).** Télécharger la figure sauvegardée au point 1) pour superposer les valeurs propres trouvées au point 3) aux disques de Gerschgorin de la matrice A.

5. **(Matlab).** Calculer les valeurs propres λ_i d'une autre façon dans Matlab. Afficher les sous la forme d'un tableau. Les comparer à celles trouvées par la méthode de la puissance inverse décalée. Commenter les éventuelles différences.
