Exercice 1 : Quelques commandes Matlab

a) Construire les vecteurs

$$u = (5, 8, 7, 9, 30),\ \ v = (2,\ 2.6,\ \cdots,\ \underbrace{2 + 0.6i}_{\text{élément i}}\ ,\cdots,\ 17),\ \ w = \underbrace{(2,\ 2, \cdots, 2,\ 2)}_{\text{100 éléments}}.$$

b) Calculer la norme euclidienne de ces vecteurs.

c) Soient $u = (1, 10, 15)$ et $v = (4, 8, 3)$ deux vecteurs. Ecrire une instruction qui permet de calculer le produit scalaire $u \cdot v$. On utilisera $\cdot *$ et *sum*.

d) Construire la matrice dont les colonnes sont

$$(1, 2, 3), \quad (7, 8, 9) \text{ et } (4, 5, 6).$$

Exercice 2 : Orthogonalisation de Gram-Schmidt

a) Rappeler les différentes étapes de l'orthogonalisation de Gram-Schmidt.

b) Ecrire une fonction *gram_schmidt* qui prend en argument d'entrée une matrice M dont les colonnes sont les vecteurs à orthonormaliser. Cette fonction renvoie la matrice Q dont les colonnes sont les vecteurs orthonormalisés.

c) Tester la fonction *gram_schmidt* sur les vecteurs :

$$u_1 = (1, 2, 3)^t,$$
$$u_2 = (2, 2, 3)^t,$$
$$u_3 = (2, 2, 1)^t.$$

Exercice 3 : Instabilité numérique de l'orthogonalisation de Gram-Schmidt

a) Utiliser votre fonction *gram_schmidt* pour orthonormaliser les vecteurs $(1, 10^{-10}, 0, 0)$, $(1, 0, 10^{-10}, 0)$, $(1, 0, 0, 10^{-10})$.

b) Les vecteurs obtenus sont-ils orthogonaux ? Donner une explication.

Exercice 4 : Orthogonalisation de Gram-Schmidt modifié

Les questions **a)**, **b)** et **c)** se traitent sur le papier.

a) Soient a et b deux vecteurs de même taille. Montrer que

$$< a, b > b = bb^t a.$$

b) Soient $(v_1, \cdots, v_n)$, une famille de n vecteurs orthonormés. Montrer que

$$(Id - v_n v_n^t)(Id - v_{n-1}v_{n-1}^t)\cdots(Id - v_1 v_1^t) = Id - v_n v_n^t - v_{n-1}v_{n-1}^t \cdots - v_1 v_1^t.$$

c) Réecrire le procédé de Gram-Schmidt en utilisant les deux résultats précédents.

d) Programmer cette nouvelle version du procédé de Gram-Schmidt.

e) Tester la nouvelle fonction sur l'exemple de l'exercice 3 et conclure.