



Université Cadi Ayyad
Faculté des Sciences Semlalia, Marrakech
Département d'informatique
Master Sciences Données

Projet de fin de module.

Titre :

Identification des influenceurs dans un réseau complexe

Réalisé par :

MAGHRANE WAIL
BENMALOUK ILHAM

Sous la direction de :

Professeur Qaffou Issam

Année universitaire 2022/2023

MAGHRANE WAIL
BENMALOUK ILHAM

Décembre 2022

Résumé

La recherche sur l'influence dans les réseaux complexes, en particulier les réseaux sociaux, est un sujet d'intérêt majeur. La détection des utilisateurs influents dans un réseau est cruciale pour atteindre une diffusion de l'information à grande échelle et à faible coût, ce qui est très utile pour les campagnes de marketing ou les campagnes politiques.

Ce rapport présente un projet visant à détecter les nœuds influents dans un réseau complexe en utilisant la technique TOPSIS pour évaluer la pertinence des nœuds. Cette méthode est comparée aux méthodes classiques basées sur les mesures de centralité à l'aide d'un modèle Susceptible-Infecté (SI). Le but de ce projet est de montrer si l'utilisation de la méthode TOPSIS est plus efficace pour détecter les nœuds influents que les méthodes classiques.

La méthode TOPSIS est utilisée pour combiner plusieurs mesures de centralité et déterminer le score d'importance de chaque nœud. Les performances sont ensuite évaluées avec un modèle SI et la technique k-means est utilisée pour identifier les clusters, les résultats de la méthode TOPSIS sont comparés sur les clusters pour déterminer s'il est préférable d'identifier les nœuds influents sur de grands ensembles de données ou de les diviser d'abord.

TABLE DES MATIÈRES

Introduction Générale	1
1 Contexte général du projet	1
1.1 Réseaux Complexes	1
1.1.1 Définition	1
1.1.2 Exemples	1
1.2 Modélisation d'un réseau complexe	3
1.2.1 Définition	3
1.3 La théorie des graphes	3
1.3.1 Détection des noeuds influents	4
2 Méthodologie du travail et algorithme utilisés	5
2.1 Méthode MCDM	5
2.1.1 TOPSIS	5
2.1.1.1 Définition	5
2.1.1.2 Les étapes de TOPSIS	6
2.2 Les mesures de centralités	7
2.2.1 Betweenness Centrality :	8
2.2.2 Closeness Centrality :	8
2.2.3 Degree Centrality :	9
2.2.4 Eigenvector Centrality :	9
2.2.5 Le modèle SI (Susceptible-Infected)	10
2.3 Algorithme K-means :	12

3	Implémentation et résultats	13
3.0.1	Dataset utilisée :	13
3.1	Processus sur Facebook ego	14
3.1.1	Application de Topsis et W-Topsis :	14
3.1.1.1	Evaluation par SI	16
3.1.1.2	Discussion et interprétation des résultats :	17
3.1.2	Application de l'algorithme K-means :	17
3.1.2.1	1ère approche : K-means prédéfinie (distance euclidien) avec initialisation par le résultat de w-topsis :	17
3.1.2.2	2ème approche : K-means prédéfinie avec initialisation aléatoire :	20
3.1.2.3	3ème approche : K-means adapté avec initialisation par le résultat de w-topsis	22
3.2	Processus sur Football	25
3.2.1	Application de Topsis et W-Topsis :	25
3.2.2	Evaluation par SI :	26
3.2.2.1	Discussion et interprétation des résultats :	26
3.2.3	Application de l'algorithme K-means :	26
3.3	Processus sur Zachary	29
3.3.1	Application de Topsis et W-Topsis :	29
3.3.2	Evaluation par SI :	30
4	Conclusion générale et perspectives	32

TABLE DES FIGURES

1.1	exemple du réseau social	2
1.2	exemple réseau informatique	2
1.3	exemple réseau biologique	2
1.4	exemple réseau technologique	3
2.1	normalisation de la matrice X	6
2.2	normalisation de la matrice X	6
2.3	normalisation pondérée de la matrice	6
2.4	$V+, V-$	7
2.5	Distance Euclidienne	7
2.6	Degré de proximité	8
2.7	Betweenness Centrality	8
2.8	Betweenness Centrality	9
2.9	Betweenness Centrality	9
3.1	graphe de facebook	13
3.2	graphe de football	14
3.3	graphe de zachary	14
3.4	Matrice d'évaluatio	14
3.5	Les 10 premiers nœuds classés par TOPSIS ,DC, CC ,BC et EC	16
3.6	Les 10 premiers nœuds classés par W-TOPSIS ,DC, CC ,BC et EC	16

3.7	Résultat de la comparaison entre les quatre mesures de centralités ,Topsis et w-topsis	17
3.8	Comparaison entre les centroïdes du k-means et w-topsis et BC,CC et DC	18
3.9	Résultat de SI	18
3.10	Résultat w-topsis sur chaque cluster	19
3.11	Résultat du SI	19
3.12	Résultat du K-means avec initialisation aléatoire . .	20
3.13	Résultat SI	21
3.14	Résultat w-topsis sur chaque cluster	21
3.15	Résultat SI	22
3.16	Résultat de w-topsis sur chaque cluster	22
3.17	Résultat du SI	23
3.18	Résultat du SI	24
3.19	Les 10 premiers nœuds classés par TOPSIS ,DC, CC ,BC et EC	25
3.20	Les 10 premiers nœuds classés par W-TOPSIS ,DC, CC ,BC et EC	25
3.21	résultat du SI	26
3.22	Résultat du k-means	27
3.23	Résultat SI	27
3.24	résultat de w-topsis sur chaque cluster	28
3.25	résultat SI	28
3.26	les dix noeuds influents avec topsis	29
3.27	les dix noeuds influents avec <i>w_topsis</i>	30
3.28	Résultat SI	30

INTRODUCTION GÉNÉRALE

L'identification des nœuds influents dans les réseaux sociaux est un sujet très débattu dans la communauté des chercheurs. Un certain nombre de stratégies ont été proposées pour identifier les nœuds clés dans des réseaux complexes. Le contrôle de la diffusion des messages et des rumeurs sur les médias sociaux, le classement de la réputation des scientifiques et d'autres nouvelles possibilités d'application sont rendus possibles par la compréhension de la capacité de diffusion d'un nœud.

Dans ce rapport, nous proposons une méthode pour déterminer les nœuds les plus influents d'un réseau en utilisant une variété de critères (mesures de centralité). Les méthodes existantes se sont concentrées sur une seule mesure de centralité, mais ces mesures ont des limites. Nous montrerons que différentes mesures de centralité ont des performances différentes en termes d'identification des nœuds influents. Nous utilisons une technique de prise de décision multi-attributs pour remédier à cette inefficacité.

Le rapport est divisé en quatre chapitres. Le premier chapitre présente le contexte général. Le deuxième chapitre discute les méthodes et la démarche utilisées pour identifier les nœuds influents. Le troisième chapitre décrit la mise en oeuvre et la réalisation de notre projet.

CHAPITRE 1

CONTEXTE GÉNÉRAL DU PROJET

Introduction

La définition d'un réseau complexe et ses différentes variétés seront abordées dans ce premier chapitre. Nous étudierons également la théorie des graphes pour transformer un réseau en graphe, ainsi que la manière de reconnaître les noeuds influents dans un réseau social, afin de mieux apprécier le sujet central du projet.

1.1 Réseaux Complexes

1.1.1 Définition

Un réseau complexe est un type particulier de réseau qui présente un degré élevé de complexité et se distingue par un nombre important de nœuds (sommets) et une quantité importante d'interconnexions entre eux. Les nœuds d'un réseau complexe peuvent représenter un large éventail d'objets, notamment des individus dans des réseaux sociaux, des pages Internet et des cellules du cerveau. Afin de représenter et de comprendre une variété de systèmes et d'événements du monde réel, les réseaux complexes sont fréquemment étudiés dans les domaines de l'informatique, de la physique et de la biologie.

1.1.2 Exemples

Réseaux sociaux

Ces réseaux sont constitués de personnes ou d'organisations qui sont liées par des relations sociales, telles que l'amitié, la parenté, la collaboration ou l'influence.



FIGURE 1.1 – exemple du réseau social

Réseau informatique

L'exemple Classique de cette catégorie est le réseau internet qui relie plusieurs page web entre eux à l'aide des Canales on peut représenter les pages comme des nœuds et les Canales comme des arêtes.

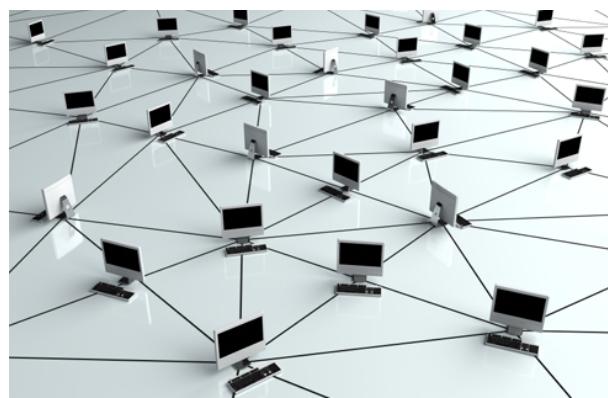


FIGURE 1.2 – exemple réseau informatique

Réseau biologique

Ces réseaux représentent les relations entre les différents éléments d'un système biologique, comme les gènes, les protéines, les cellules, les organes, etc.



FIGURE 1.3 – exemple réseau biologique

Réseau technologique

Ces réseaux regroupent les équipements et les technologies qui sont utilisées pour transférer de l'information ou de l'énergie. Les réseaux technologiques incluent par exemple les réseaux de télécommunication, les réseaux électriques, les réseaux de transport, etc



FIGURE 1.4 – exemple réseau technologique

1.2 Modélisation d'un réseau complexe

1.2.1 Définition

La modélisation des réseaux complexes consiste à utiliser des techniques mathématiques et informatiques pour comprendre comment ces réseaux fonctionnent et comment ils influencent le comportement des éléments qui les composent. Cela peut être utile pour prédire comment un réseau va évoluer dans le temps ou comment il va réagir à des perturbations externes. Il existe plusieurs approches pour modéliser des réseaux complexes, chacune ayant ses propres avantages et limitations. Certaines approches couramment utilisées incluent

- • Les réseaux de Petri
 - • Les réseaux de Bayésiens
 - • Les réseaux de graphes

1.3 La théorie des graphes

La théorie des graphes est un outil puissant pour la modélisation des réseaux complexes, car elle permet de représenter de manière claire et concise les relations entre les différents éléments d'un réseau. En utilisant des graphes, on peut modéliser de nombreux types de réseaux complexes, tels que les réseaux sociaux, les réseaux de communication, les réseaux de transport, les réseaux biologiques, etc.

Pour modéliser un réseau complexe en utilisant la théorie des graphes, il faut d'abord identifier les éléments du réseau et les relations qui existent entre eux. Ces éléments sont représentés par des sommets dans le graphe, et les relations entre eux sont représentées par des arêtes. Une fois que le graphe est construit, on peut utiliser différentes techniques de la théorie des graphes pour étudier les propriétés du réseau. Par exemple, on peut calculer le degré de chaque sommet pour mesurer son importance dans le réseau, la centralité de chaque sommet (c'est-à-dire son importance dans le réseau), etc.

1.3.1 Détection des noeuds influents

La détection des noeuds influents dans un réseau complexe consiste à identifier les noeuds qui ont une grande influence sur le reste du réseau. Cela peut être utile pour plusieurs raisons, notamment :

Améliorer la diffusion de l'information : en identifiant les noeuds influents, vous pouvez améliorer la diffusion de l'information dans le réseau en utilisant ces noeuds comme relais pour atteindre d'autres parties du réseau.

Identifier les points de contrôle clés : en connaissant les noeuds influents, vous pouvez identifier les points de contrôle clés dans le réseau et ainsi mieux comprendre comment vous pouvez influencer le réseau.

Identifier les points de faiblesse : en identifiant les noeuds influents, vous pouvez également identifier les points de faiblesse du réseau et ainsi mieux comprendre comment le réseau pourrait être perturbé ou modifié.

Conclusion

L'identification des noeuds influents dans un réseau complexe est cruciale pour comprendre comment le réseau fonctionne et comment il peut être influencé ou manipulé. Plusieurs mesures de centralité ont été proposées au fil des ans pour classer les noeuds d'un graphe en fonction de leur importance topologique.

CHAPITRE 2

MÉTHODOLOGIE DU TRAVAIL ET ALGORITHME UTILISÉS

Introduction

Dans ce chapitre, nous abordons la méthodologie qu'on va suivre dans ce projet ainsi que les différentes méthodes et les différents algorithmes qu'on va utiliser afin d'identifier les nœuds influents dans un réseau complexe

2.1 Méthode MCDM

Lorsque les critères de sélection sont nombreux et peuvent être en contradiction les uns avec les autres, des approches décisionnelles dites multicritères (MCDM) sont appliquées. Pour faire le meilleur choix, ces stratégies permettent de prendre en compte de nombreux facteurs et de les pondérer en fonction de leur importance relative. Il existe de nombreuses méthodes MCDM, notamment le Weighted Sum Model (WSM), la technique d'ordre de préférence par similitude à la solution idéale (TOPSIS), l'Analytic Hierarchy Process (AHP), etc...

2.1.1 TOPSIS

2.1.1.1 Définition

La méthode TOPSIS (Technique de préférence d'ordre par similarité à la solution idéale) est une méthode de prise de décision pour des problèmes à attributs multiples. Elle consiste à créer une solution idéale et une solution moins idéale, et utilise ces deux référentiels pour évaluer les projets réalisables. La solution idéale est la solution optimale qui atteint la meilleure valeur pour chaque attribut, tandis que la solution moins idéale

est la pire solution possible. Les projets alternatifs sont comparés à ces deux solutions pour déterminer le meilleur projet, celui qui est proche de la solution idéale et éloigné de la solution moins idéale.

2.1.1.2 Les étapes de TOPSIS

- Première étape :

Créer une matrice de décision standardisée A a m alternatives et n critères, et l'intersection de chaque alternative avec chaque critère est indiquée ,Cette matrice est généralement appelée « matrice d'évaluation ».

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{j=1}^k x_{ij}^2}}$$

FIGURE 2.1 – normalisation de la matrice X

- Deuxième étape : Normaliser la matrice de décision pour obtenir une nouvelle matrice R d'élément tel que :

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{j=1}^k x_{ij}^2}}$$

FIGURE 2.2 – normalisation de la matrice X

- troisième étape :

Calculer la matrice normalisée pondérée. Les poids sont donnés par les décideurs pour représenter leurs préférences entre les critères,avec :

$$v_{ij} = w_i \times r_{ij}$$

FIGURE 2.3 – normalisation pondérée de la matrice

-
- **Quatrième étape :** Dans cette étape on va calculer la solution idéale et la solution la moins idéale noté respectivement V^+ , V^-

$$V^+ = (best_j(v_{1j}), best_j(v_{2j}), \dots, best_j(v_{nj}))$$

$$V^- = (worst_j(v_{1j}), worst_j(v_{2j}), \dots, worst_j(v_{nj}))$$

Avec $best_j(v_{1j}) = \begin{cases} \max_j(v_{ij}) & \text{si } c_i \text{ est bénéficial} \\ \min_j(v_{ij}) & \text{si } c_i \text{n'est pas bénéfical} \end{cases}$

et $worst_j(v_{1j}) = \begin{cases} \min_j(v_{ij}) & \text{si } c_i \text{ est bénéficial} \\ \max_j(v_{ij}) & \text{si } c_i \text{n'est pas bénéfical} \end{cases}$

FIGURE 2.4 – V^+, V^-

- **Cinquième étape :** Calculer pour chaque alternative, la distance euclidienne entre la solution idéale et la solution la moins idéale, suivant à la formule :

$$d_j^+ = \sqrt{\sum_{i=1}^n (v_{ij} - v_i^+)^2}$$

FIGURE 2.5 – Distance Euclidienne

- **Sixième étape :** Calculer le degré de proximité au positif idéal V_j^+ . Plus V_j^+ est important, plus l'alternative j est proche de la solution idéale et loin de la solution moins idéale, avec :

2.2 Les mesures de centralités

Il existe de nombreuses mesures de centralité qui ont été développées pour classer les noeuds de réseau. Un noeud ayant un degré plus élevé, tel que le degré de centralité, est censé avoir une plus grande influence (par exemple, en tant que noeud initialement infecté, il est censé se propager plus rapidement et plus largement) qu'un noeud ayant un degré plus

$$D_j^+ = \frac{d_j^-}{d_j^- + d_j^+}$$

FIGURE 2.6 – Degré de proximité

faible. Cependant, le fait que cette méthode n'évalue qu'une petite quantité de données peut parfois l'empêcher d'identifier correctement les noeuds influents.

2.2.1 Betweenness Centrality :

Cette influence peut être déterminée à l'aide de la métrique de centralité d'interdépendance, qui quantifie l'impact d'un noeud sur le flux d'informations d'un graphe. Elle est largement utilisée pour localiser les noeuds qui relient deux zones différentes d'un réseau.

La centralité d'interférence du noeud v , représentée par le symbole $CB(v)$, pour un réseau.

$$CB(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

FIGURE 2.7 – Betweenness Centrality

$G = (V, E)$ avec $n = |V|$ noeuds et $m = |E|$ arêtes, est la suivante : où le nombre de plus courts chemins entre les noeuds s et t est , et le nombre de plus courts chemins entre les noeuds s et t qui passent par le noeud v est (v) .

2.2.2 Closeness Centrality :

Un moyen d'identifier les noeuds qui peuvent diffuser efficacement des informations dans un graphe est leur centralité de proximité. La distance moyenne (distance inverse) d'un noeud par rapport à tous les autres noeuds est mesurée par sa centralité de proximité. Les distances entre les noeuds

qui ont un score de proximité élevé sont les plus courtes.

$$C_C(v) = \frac{1}{\sum_{t \in V \setminus v} d_G(v, t)},$$

FIGURE 2.8 – Betweenness Centrality

où la distance euclidienne ((v, t)) entre v et t. Le nombre de propagation de l'information d'un nœud à d'autres nœuds assignables du réseau peut être considérée comme une mesure de proximité.

2.2.3 Degree Centrality :

La centralité de degré mesure le nombre de liens (ou arcs) qui entrent ou sortent d'un noeud dans un graphe. Plus précisément, elle mesure le nombre de voisins qu'un noeud a dans le graphe. Cette mesure est souvent utilisée pour identifier les noeuds les plus connectés ou les plus influents dans un réseau. Elle est simple à calculer et facile à interpréter. Cependant, elle ne prend pas en compte la structure globale du graphe, ce qui peut conduire à des résultats trompeurs dans certains cas.

2.2.4 Eigenvector Centrality :

La centralité de vecteur propre mesure l'importance d'un noeud dans un graphe en fonction de l'importance des noeuds auxquels il est relié. Cela signifie que si un noeud est connecté à d'autres noeuds importants, sa propre centralité sera élevée. Cette idée est basée sur l'hypothèse que les noeuds importants sont plus susceptibles d'être connectés à d'autres noeuds importants, donc une connexion à un noeud important contribuera davantage au score global d'un noeud qu'une connexion à un noeud moins important. Soit A une matrice de similarité de taille n*n. La entrée du vecteur propre normalisé relatif à la plus grande valeur propre de A est définie comme la centralité du vecteur propre xi du nœud i. n est le nombre de sommets, et la plus grande valeur propre de A est :

$$Ax = \lambda x, \quad x_i = \mu \sum_{j=1}^n a_{ij}x_j, \quad i = 1, \dots, n$$

FIGURE 2.9 – Betweenness Centrality

est proportionnel aux scores de similarité totaux de tous les nœuds qui lui sont connectés avec un facteur de proportionnalité de $\mu = 1 /$.

2.2.5 Le modèle SI (Susceptible-Infected)

L'équation $S + I = 1$ décrit la relation entre les nœuds sains (S) et infectés (I) dans un réseau. Elle indique que la somme des nœuds sains et infectés dans le réseau est égale à 1, ce qui signifie que tous les nœuds appartiennent à l'une de ces deux catégories.

Pour créer une équation théorique pour un réseau SI temporel, on examine la façon dont une infection se propage dans un réseau. Pour ce faire, on détermine la probabilité de chaque combinaison d'arêtes possible (SI, SS ou II) pour chaque pas de temps. Les combinaisons de nœuds SS et II sont classées dans la catégorie 0, car aucun nœud supplémentaire ne sera infecté, ce qui signifie qu'il n'y aura pas d'augmentation de la fraction de nœuds infectés. La dernière combinaison, SI, signifie qu'un nœud supplémentaire contractera l'infection, augmentant la proportion de nœuds infectés de $1/n$ (où n est le nombre total de nœuds dans le réseau).

Ainsi, en combinant ces informations sur les combinaisons d'arêtes et les probabilités correspondantes, on peut établir la valeur attendue du nombre de nœuds infectés au pas de temps suivant. Cette équation reflète théoriquement la façon dont l'infection se propage dans le réseau et peut être utilisée pour simuler la propagation de l'infection au fil du temps.

Le taux de variation du nombre de nœuds infectés est alors représenté comme suit :

$$\frac{dI}{dt} = \begin{cases} 0 & \text{avec probabilité } (1 - I)^2 + I^2 \\ \frac{1}{n} & \text{avec probabilité } 2I(1 - I) \end{cases}$$

La valeur attendue étant la somme de tous les résultats possibles multipliée par leurs probabilités, nous pouvons construire la fonction de valeur attendue pour le nombre de nœuds infectés à partir de ce point. Voici un schéma de l'évolution prévue des nœuds infectés :

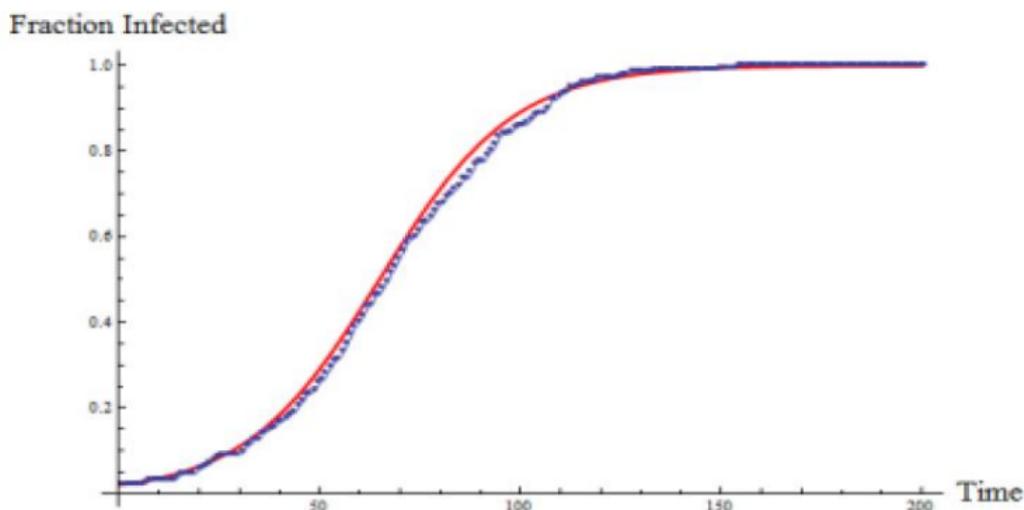
$$\frac{dI}{dt} = \frac{2}{n} I(1 - I)$$

Afin de développer une théorie universelle de champ moyen dépendant du nombre d'arêtes activées simultanément (w), nous suivons ensuite une dérivation similaire pour l'activation de deux bords à la fois, puis de trois.

$$\frac{dI}{dt} = \frac{2w}{n} I(1 - I)$$

Les arêtes qui partagent un nœud commun sont ignorées dans cette dérivation. Ensuite, en utilisant les mêmes présomptions que le modèle, nous simulons la transmission de l'infection sur un réseau temporel et comparons les résultats à notre équation théorique.

Courbes expérimentales basées sur les simulations effectuées. Notez que les deux correspondent bien.



La Figure montre que les résultats expérimentaux et théoriques sont similaires, ce qui est démontré par la similitude entre la médiane des courbes expérimentales et la courbe théorique. La médiane est utilisée plutôt que la moyenne car il y a des valeurs aberrantes qui pourraient fausser la moyenne, soit en la rendant plus élevée ou plus basse que ce qui est prévu. Cela montre que le réseau temporel est en bon accord avec les prédictions théoriques sur la façon dont la maladie se propage dans ce réseau temporel. En étudiant la théorie de champ moyen temporel, nous avons remarqué que certaines combinaisons d'arêtes pourraient partager des nœuds, ce qui pourrait permettre à des voisins de voisins d'un nœud infecté de devenir infectés en une seule étape de temps, si l'infection est autorisée à se propager d'un nœud infecté à un nœud sensible et à travers une autre connexion à un nœud sensible différent. Ainsi, nous allons examiner plusieurs cycles d'infection pour un réseau en conséquence.

2.3 Algorithme K-means :

L'algorithme est algorithme d'apprentissage non supervisé utilisé dans le clustering, k-means est une technique de regroupement populaire utilisée pour partitionner un ensemble de données en k clusters. Il est utilisé pour partitionner un réseau en k clusters en utilisant le placement optimal du centroïde pour chaque cluster.

Dans cet algorithme, le réseau est initialement divisé en k clusters, où chaque cluster est défini par un bus de référence (centroïde) qui minimise la somme des distances euclidiennes entre les nœuds du cluster et le centroïde. Les autres nœuds sont ensuite partitionnés et affectés aux clusters en fonction de leur proximité aux centroïdes respectifs.

Après chaque partition, les centroïdes sont recalculés pour devenir les nouveaux points de référence pour le prochain partitionnement. Ce processus itératif continue jusqu'à ce que les clusters atteignent un minimum local qui dépend de la sélection initiale des centroïdes. La mesure d'erreur utilisée est la somme des variances pour chaque cluster.

L'algorithme k-means est composé de plusieurs étapes :

1. Sélection des centroïdes initiaux de clusters : les clusters sont initialisés en sélectionnant k centroïdes dans le réseau.
2. Regroupement des nœuds en clusters : les nœuds sont regroupés en fonction de la distance euclidienne entre eux et les centroïdes.
3. Recalcule des positions des centroïdes : une fois tous les nœuds affectés aux clusters, les positions des centroïdes sont recalculées.
4. Évaluation de la fonction objective : la fonction potentielle est évaluée après chaque regroupement.
5. Itérations de l'algorithme : les étapes 2 à 4 sont répétées jusqu'à ce que les centroïdes ne changent plus de position.

Conclusion

Dans ce chapitre, nous avons introduit la méthode TOPSIS qui vise à repérer les nœuds influents dans un réseau. Nous avons défini les mesures de centralité nécessaires à sa mise en œuvre. Nous avons également présenté l'algorithme K-means qui permet de diviser les données d'un réseau complexe en sous-réseaux plus petits, ainsi que le modèle SI qu'on va utiliser pour évaluer les différentes méthodes.

CHAPITRE 3

IMPLÉMENTATION ET RÉSULTATS

Introduction

Ce chapitre se focalise sur la réalisation et l'implémentation des différentes approches qu'on a présenté précédemment en utilisant différents dataset pour comparer les résultats obtenues et déterminer la meilleure méthode pour identifier les nœuds influents dans un réseau complexe.

3.0.1 Dataset utilisée :

Afin d'avoir une meilleure compréhension de l'effet de chaque méthode, nous avons utilisé différents types de réseaux. Nous avons commencé par utiliser un réseau Facebook ego qui contient 4038 nœuds et 88234 arêtes. Ensuite, Nous avons également appliqué les méthodes sur des petits réseaux comme Football qui contient 115 nœuds et 613 arêtes, Zachary qui contient 34 nœuds et 78 arêtes. Tous ces réseaux sont stockés dans des fichiers txt sous forme de deux colonnes et des lignes, contenant les numéros des nœuds.

* [Facebook ego](#)

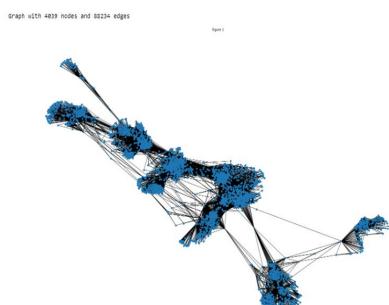


FIGURE 3.1 – graphe de facebook

* Football

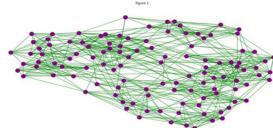


FIGURE 3.2 – graphe de football

* Zachary

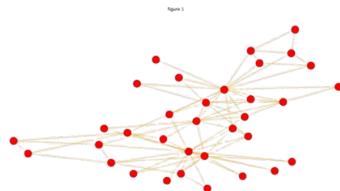


FIGURE 3.3 – graphe de zachary

3.1 Processus sur Facebook ego

3.1.1 Application de Topsis et W-Topsis :

Dans cette première partie, nous allons utiliser les mesures de centralité et la méthode TOPSIS pour comparer leur pertinence avec le modèle SI.

Élaborer la matrice d'évaluation.

	Node	DC	BC	CC	EC
0	0	0.085934	1.463059e-01	0.353343	3.391796e-05
1	1	0.004210	2.783274e-06	0.261376	6.045346e-07
2	2	0.002476	7.595021e-08	0.261258	2.233461e-07
3	3	0.004210	1.685066e-06	0.261376	6.635648e-07
4	4	0.002476	1.840332e-07	0.261258	2.236416e-07
...
4034	4034	0.000495	0.000000e+00	0.183989	2.951270e-10
4035	4035	0.000248	0.000000e+00	0.183980	2.912901e-10
4036	4036	0.000495	0.000000e+00	0.183989	2.931223e-10
4037	4037	0.000991	7.156847e-08	0.184005	2.989233e-10
4038	4038	0.002229	6.338922e-07	0.184047	8.915175e-10

FIGURE 3.4 – Matrice d'évaluatio

On a suivi toutes les étapes de l'algorithme de Topsis, et concernant l'étape où on doit attribuer des poids à chaque mesure de centralité, on a essayé avec plusieurs poids afin d'avoir des poids optimaux mais cette méthode est peu pratique, et pour éviter cela on a fait recourt à une autre méthode appelée W-Topsis.

On a réalisé les mêmes étapes de Topsis, la seule différence est de trouver une fonction qui peut calculer automatiquement les meilleurs poids au lieu de les donner aléatoirement et les évaluer à chaque fois avec le modèle SI. Pour ce faire, on a travaillé par la méthode d'entropie pour calculer ces poids.

la méthode d'entropie est la suivante : Tout d'abord, on a calculé la proportion de la valeur du critère de l'alternative () sous le critère () en utilisant la formule suivante :

$$p_{je} = \frac{X_{je}}{\sum_{je=1}^n X_{je}}$$

Deuxièmement, on a déterminé l'entropie "Ej" du critère (j) en utilisant l'équation suivante :

$$k = \frac{1}{\ln(n)}$$

$$e_j = -k \sum_{i=1}^n p_{ij} \ln(p_{ij})$$

$$r_j = 1 - e_j$$

$$w_j = \frac{r_j}{\sum_{j=1}^m r_j}$$

Apres d'appliquer topsis et w-topsis on a obtenir 10 noeud influent classé dans le tableau suivant :

Node	DC	BC	CC	EC	C
107	0.258791	0.480518	0.459699	2.606940e-04	0.645964
1912	0.186974	0.229295	0.350947	9.540696e-02	0.601426
1684	0.196137	0.337797	0.393606	7.164260e-06	0.533306
3437	0.135463	0.236115	0.314413	9.531613e-06	0.398863
2266	0.057949	0.001708	0.281708	8.698328e-02	0.343452
2206	0.052006	0.000012	0.263336	8.605239e-02	0.339017
2233	0.054978	0.000114	0.263543	8.517341e-02	0.337266
2142	0.054730	0.000090	0.263525	8.419312e-02	0.334613
2464	0.050025	0.000008	0.263199	8.427877e-02	0.333989
2218	0.050768	0.000015	0.263251	8.415574e-02	0.333790

FIGURE 3.5 – Les 10 premiers nœuds classés par TOPSIS ,DC, CC ,BC et EC

w_Node	DC	BC	CC	EC	C
107	0.258791	0.480518	0.459699	2.606940e-04	0.927913
1684	0.196137	0.337797	0.393606	7.164260e-06	0.696069
3437	0.135463	0.236115	0.314413	9.531613e-06	0.488526
1912	0.186974	0.229295	0.350947	9.540696e-02	0.480827
1085	0.016345	0.149015	0.357852	3.164082e-06	0.308478
0	0.085934	0.146306	0.353343	3.391796e-05	0.303198
698	0.016840	0.115330	0.271189	1.116876e-09	0.238868
567	0.015602	0.096310	0.328881	9.932295e-06	0.199523
58	0.002972	0.084360	0.397402	5.898120e-04	0.174775
428	0.028479	0.064309	0.394837	5.990065e-04	0.133340

FIGURE 3.6 – Les 10 premiers nœuds classés par W-TOPSIS ,DC, CC ,BC et EC

3.1.1.1 Evaluation par SI

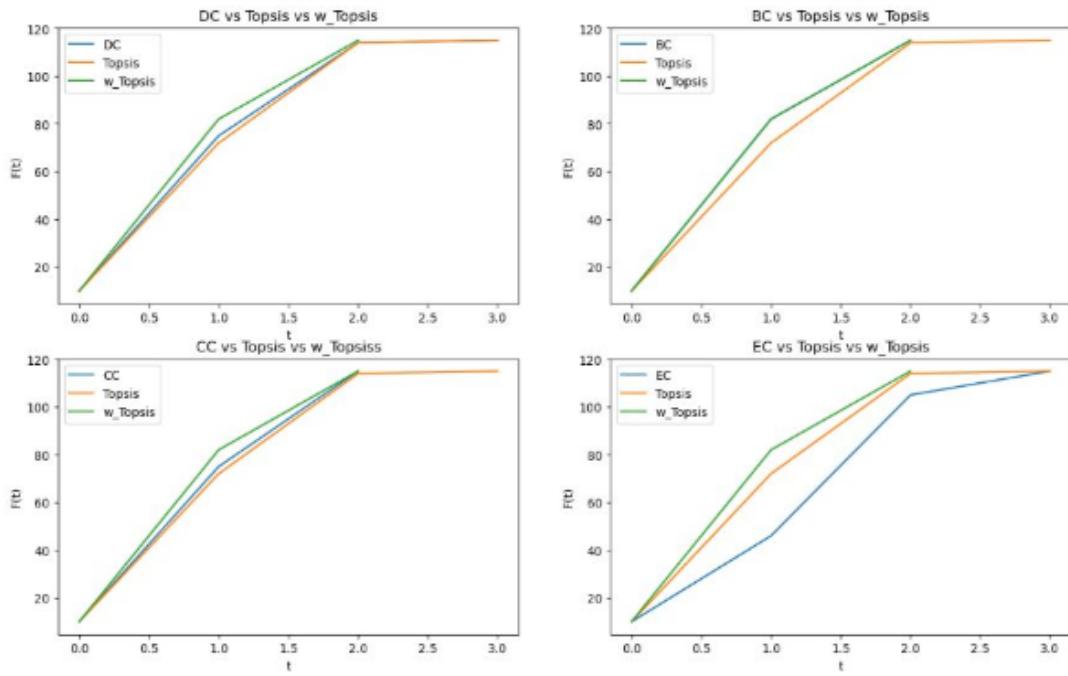


FIGURE 3.7 – Résultat de la comparaison entre les quatre mesures de centralités ,Topsis et w-topsis

3.1.1.2 Discussion et interprétation des résultats :

Les résultats de l’expérience sur le réseau Facebook montrent que notre méthode W-topsis est plus performante que les mesures de centralité traditionnelles telles que la centralité de proximité, la centralité d’interdépendance, et la centralité de degré. Elle est également presque aussi efficace que la centralité de vecteur propre. et elle plus efficace par rapport à Topsis.

3.1.2 Application de l’algorithme K-means :

Nous utiliserons la méthode Kmeans dans cette section du projet pour diviser notre ensemble de données en plus petits ensembles de données. Ensuite, nous comparons les deux résultats avec le modèle SI après avoir effectué la même procédure (w-topsis et mesures de centralité). Nous choisirons $k = 10$ comme nombre de clusters.

pour ce faire on a utilisé deux approches :

3.1.2.1 1^{ère} approche : K-means prédéfinie (distance euclidien) avec initialisation par le résultat de w-topsis :

On a initialisé les k centres de classes par les tops k qu’on a obtenu après la méthode de w-topsis ; on a choisi les 10 premiers nœuds influents et on a obtenu 10 clusters.

cluster	Node_before_KMeans	Node_after_KMeans	distance from center
0	0	107	0.000000e+00
1	1	1684	1.270549e-19
2	2	3437	6.950964e-20
3	3	1912	0.000000e+00
4	4	1085	1.425377e-02
5	5	0	3.541115e-02
6	6	698	1.678485e-02
7	7	567	8.161578e-04
8	8	58	2.899335e-03
9	9	428	1.700190e-04

FIGURE 3.8 – Comparaison entre les centroïdes du k-means et w-topsis et BC,CC et DC

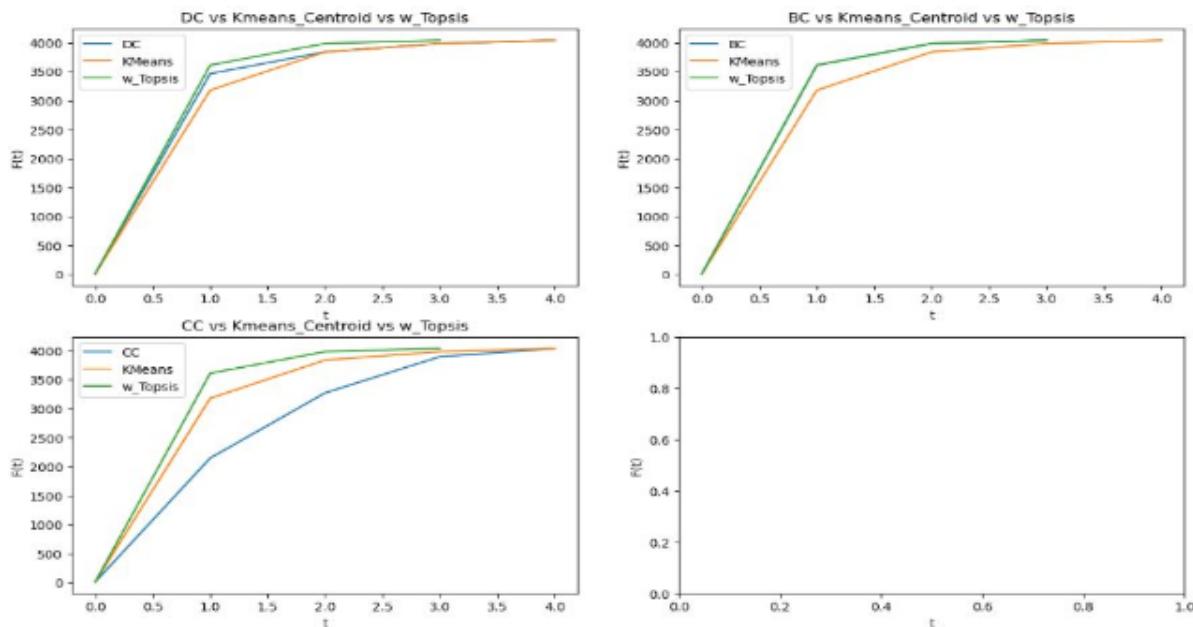


FIGURE 3.9 – Résultat de SI

Selon la figure présentée, on constate que l'algorithme k-means est plus meilleur que CC et il a le même résultat que BC . Cependant, DC a des performances supérieures à celles du k-means en termes de nombre moyen de noeuds infectés. Tandis que la méthode w-topsis est la plus pertinente par rapport à l'algorithme k-means et par rapport à BC,CC,et DC.

Application de K-means sur chaque cluster

Après le k-means on a pris chaque cluster et on l'a considéré comme un autre dataset et on a appliqué sur cette dernière la méthode W-TOPSIS afin d'identifier le premier nœud influent et comparer ensuite les dix nœuds influents des dix clusters avec le résultat de W-topsis appliquait sur toute la dataset

cluster 0 :								
Node	DC	BC	CC	EC				
0	107.0	0.258791	0.480518	0.459699	0.000261			
cluster 1 :								
Node	DC	BC	CC	EC				
1	1684.0	0.196137	0.337797	0.393696	0.000097			
cluster 2 :								
Node	DC	BC	CC	EC				
2	3437.0	0.135463	0.236115	0.314413	9.531613e-08			
cluster 3 :								
Node	DC	BC	CC	EC				
3	1912.0	0.186974	0.229295	0.350947	0.095487			
cluster 4 :								
Node	DC	BC	CC	C	S+	S-	C.1	
0	563	0.434783	0.524873	0.45098	0.990697	0.004338	0.472159	0.990697
cluster 5 :								
Node	DC	BC	CC	EC				
5	567.0	0.015602	0.09631	0.328881	0.00001			
cluster 6 :								

FIGURE 3.10 – Résultat w-topsis sur chaque cluster

Comparaison entre w-topsis et les mesures de centralité le k-means avec w-topsis sur chaque cluster

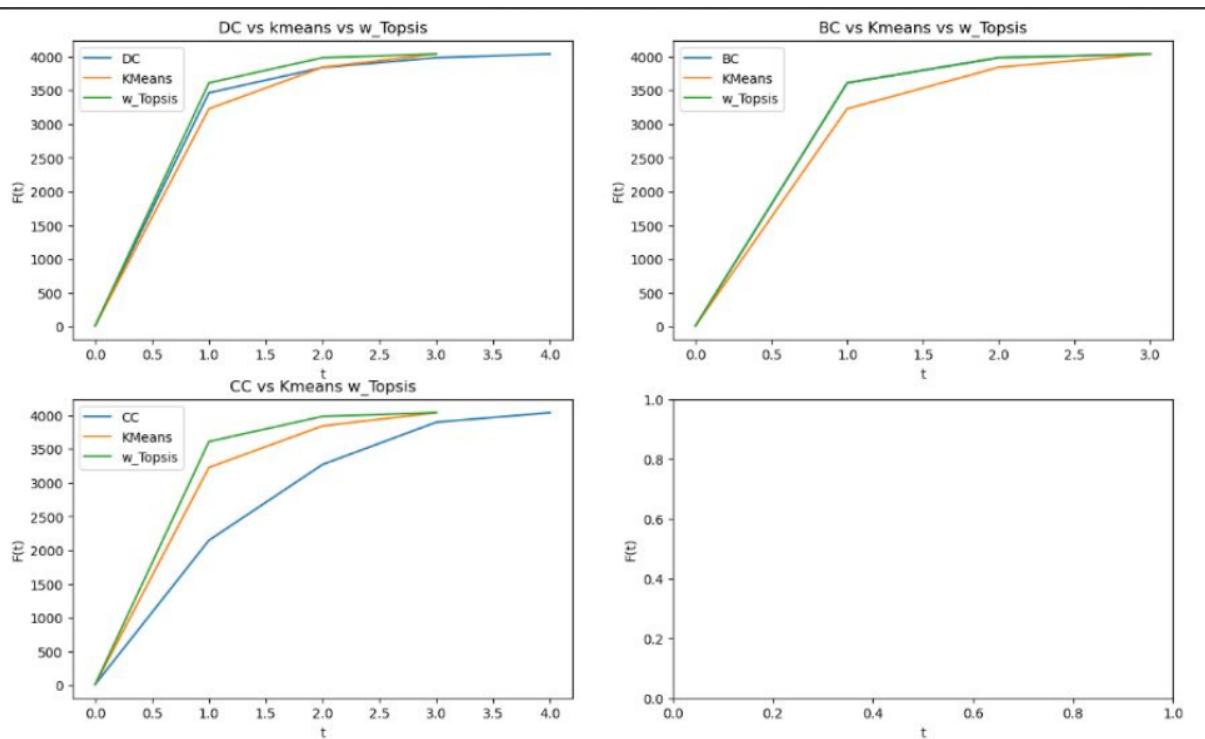


FIGURE 3.11 – Résultat du SI

On remarque toujours que le résultat de w-topsis est le meilleur par rapport à celui du k-means même si on a appliqué w-topsis sur chaque cluster. et la même remarque pour CC, BC et DC.

3.1.2.2 2ème approche : K-means prédéfinie avec initialisation aléatoire :

Dans cette partie on a suiv la même démarche de la 1ère approche on a divisé notre dataset sur des clusters en utilisant le k-means mais au lieu d'initialiser les premiers centroïdes avec le résultat de w-topsis on a fais une initialisation aléatoire, puis on a appliqué w-topsis sur chaque cluster comme précédemment .

cluster	Node_before_KMeans	Node_after_KMeans	distance from center
0	107	765	0.001527
1	1684	1277	0.002659
2	3437	2927	0.002203
3	1912	2112	0.000952
4	1085	2795	0.000216
5	0	303	0.001776
6	698	1684	0.032462
7	567	3662	0.002842
8	58	414	0.034669
9	428	1606	0.000416

FIGURE 3.12 – Résultat du K-means avec initialisation aléatoire

Comparaison entre les centroïdes obtenus et w-topsis

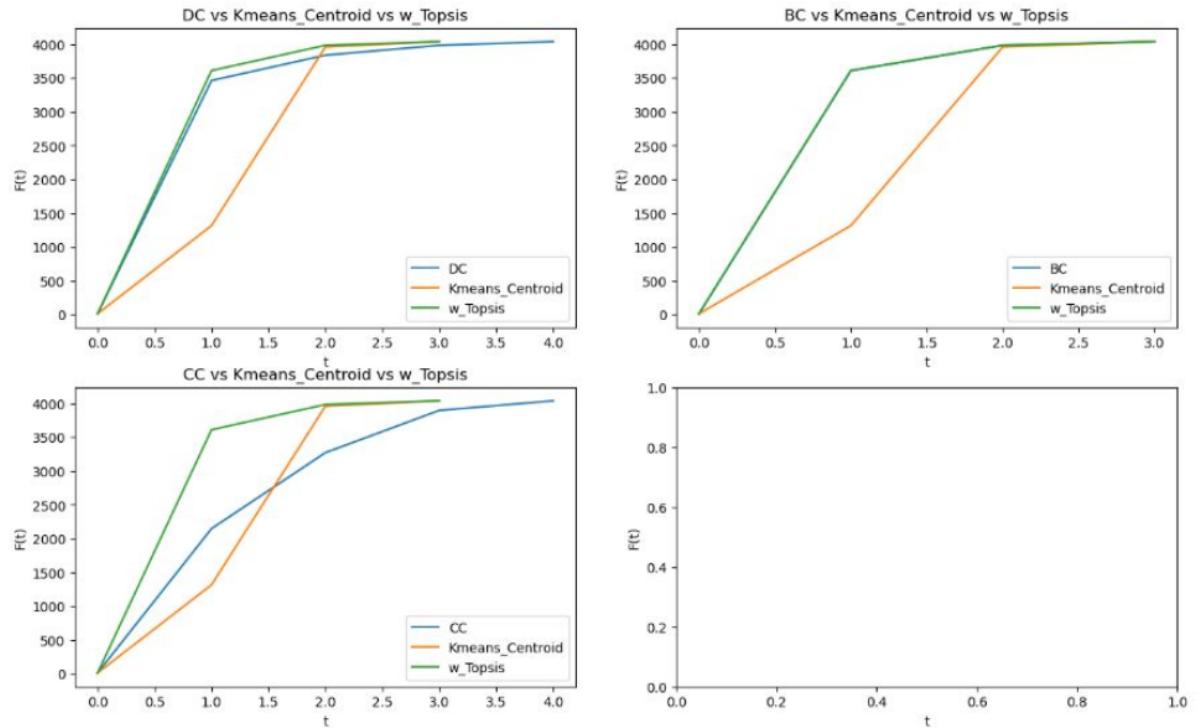


FIGURE 3.13 – Résultat SI

```

clusterS 0 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 705 0.288043 0.063698 0.391915 1.0 0.0 0.309258 1.0
clusterS 1 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 1086 0.588679 0.191038 0.708556 0.996565 0.00264 0.765971 0.996565
clusterS 2 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 2863 0.179487 0.04417 0.316873 0.991848 0.004004 0.487149 0.991848
clusterS 3 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 2266 0.985 0.00253 0.985222 1.0 0.0 0.106498 1.0
clusterS 4 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 2759 0.019074 0.06358 0.255089 0.967951 0.009941 0.30024 0.967951
clusterS 5 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 2313 0.093943 0.041322 0.199118 1.0 0.0 0.327652 1.0
clusterS 6 :

```

FIGURE 3.14 – Résultat w-topsis sur chaque cluster

Comparaison entre w-topsis et le k-means avec w-topsis sur chaque cluster

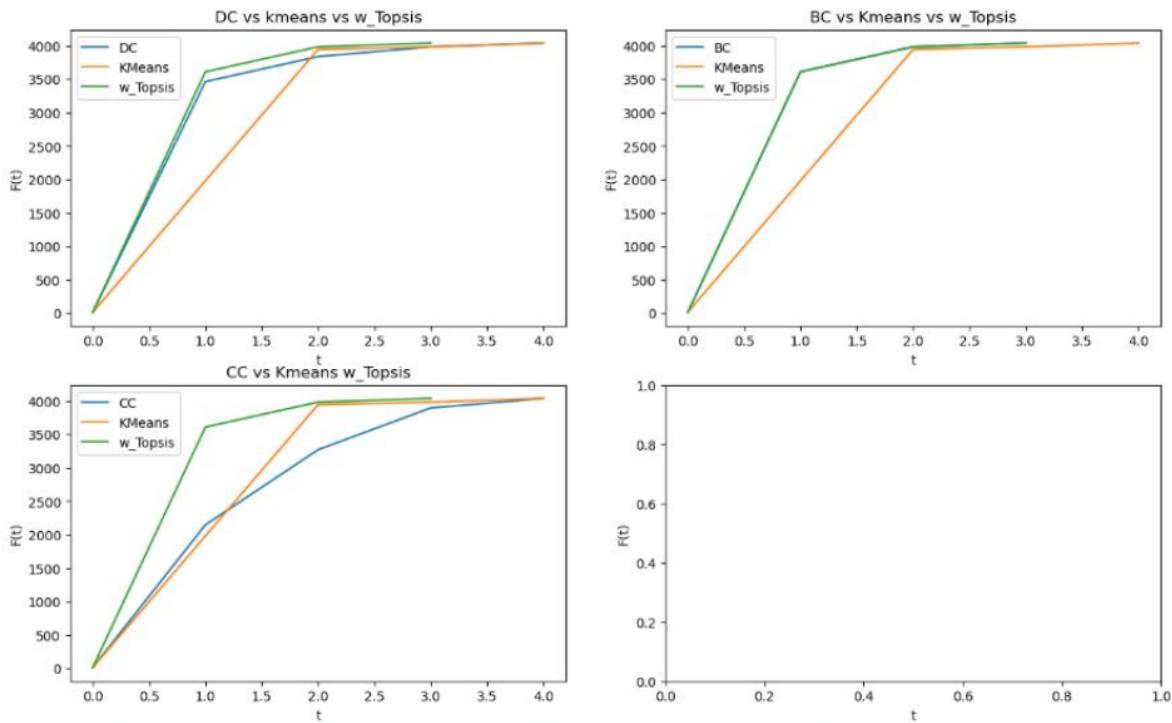


FIGURE 3.15 – Résultat SI

on constate que la méthode w-topsis est toujours la meilleure par rapport à l'algorithme du k-means et par rappor à CC ,BC et DC.

3.1.2.3 3ème approche : K-means adapté avec initialisation par le résultat de w-topsis

Dans cette partie on a implémenté l'algorithme du k-means en utilisant les scores pour réaliser les clusters ; on a affecté à chaque noeud un score ($S=0,5*BC +0,3*DC+0,2*EC$) et on a utilisé la closeness au lieu de la distance euclidienne. et après cette division on a appliqué de la même manière w-topsis sur chaque cluster et on a pris le premier noeud influent

```

clusters 0 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 1577  0.065714  0.145941  0.116312  0.994426  0.001844  0.328916  0.994426
clusters 1 :
    Node      DC      BC      CC      C      S+      S-      C.1
0  637  0.030986  0.024523  0.066355  0.889417  0.044462  0.35761  0.889417
clusters 2 :
    Node      DC      BC      CC      C      S+      S-      C.1
0  107  0.316716  0.172224  0.219941  1.0  0.0  0.662897  1.0
clusters 3 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 1666  0.014925  0.038264  0.069924  0.884799  0.037264  0.286201  0.884799
clusters 4 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 1632  0.048276  0.0694  0.101567  0.85525  0.0569  0.336192  0.85525
clusters 5 :
    Node      DC      BC      CC      C      S+      S-      C.1
0 1718  0.041667  0.087217  0.124808  0.972366  0.010984  0.38651  0.972366
clusters 6 .

```

FIGURE 3.16 – Résultat de w-topsis sur chaque cluster

Comparaison avec w-topsis et k-means modifié

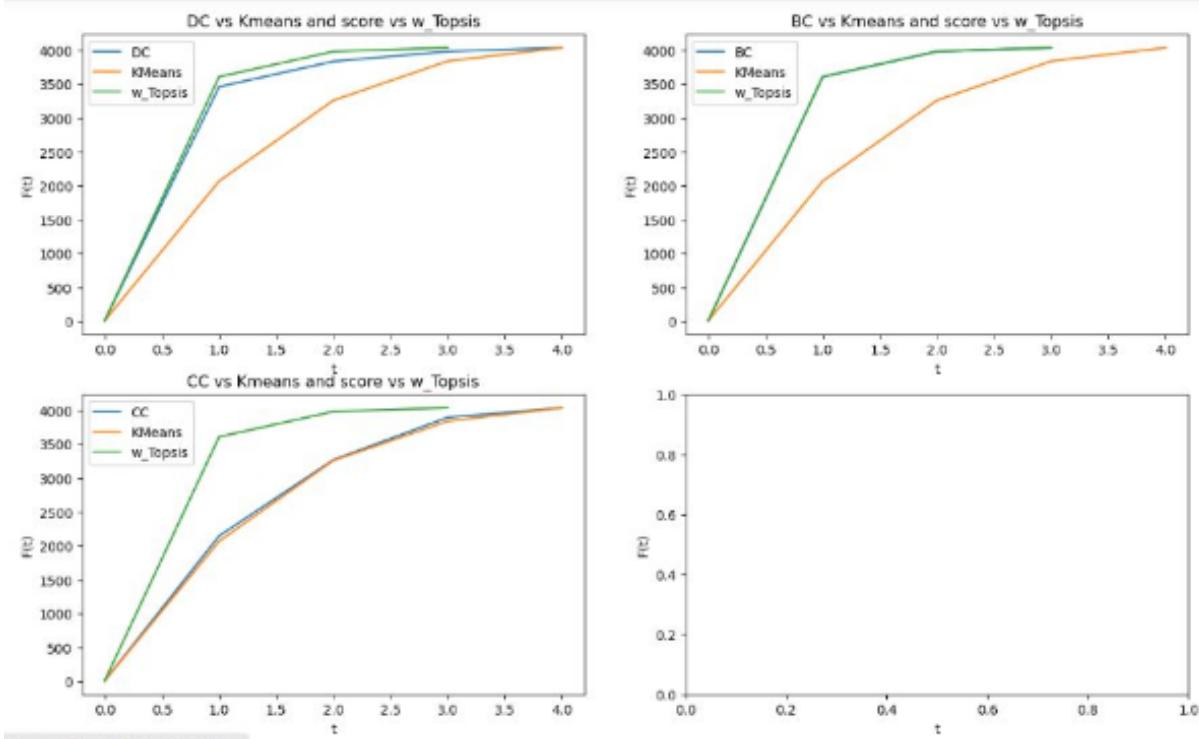


FIGURE 3.17 – Résultat du SI

on constate que la méthode w-topsis est toujours plus performante que l'algorithme du k-means même si on a modifié l'algorithme. et on a essayé d'utiliser les deux méthodes ELBOW et SILHOUETTE SCORE pour déterminer le nombre idéale k de cluster qu'on peut avoir de notre dataset et on a trouvé $k=7$ par elbow et $K=6$ par silhouette score et on a suit le même processus ; on a appliqué w-topsis sur les six clusters et puis sur les sept clusters mais on obtient toujours le même résultat du SI ; w-topsis est meilleur que l'algorithme du k-means. on a essayé avec $k=13$ qu'on a choisi aléatoirement et on a trouvé le résultat suivant :

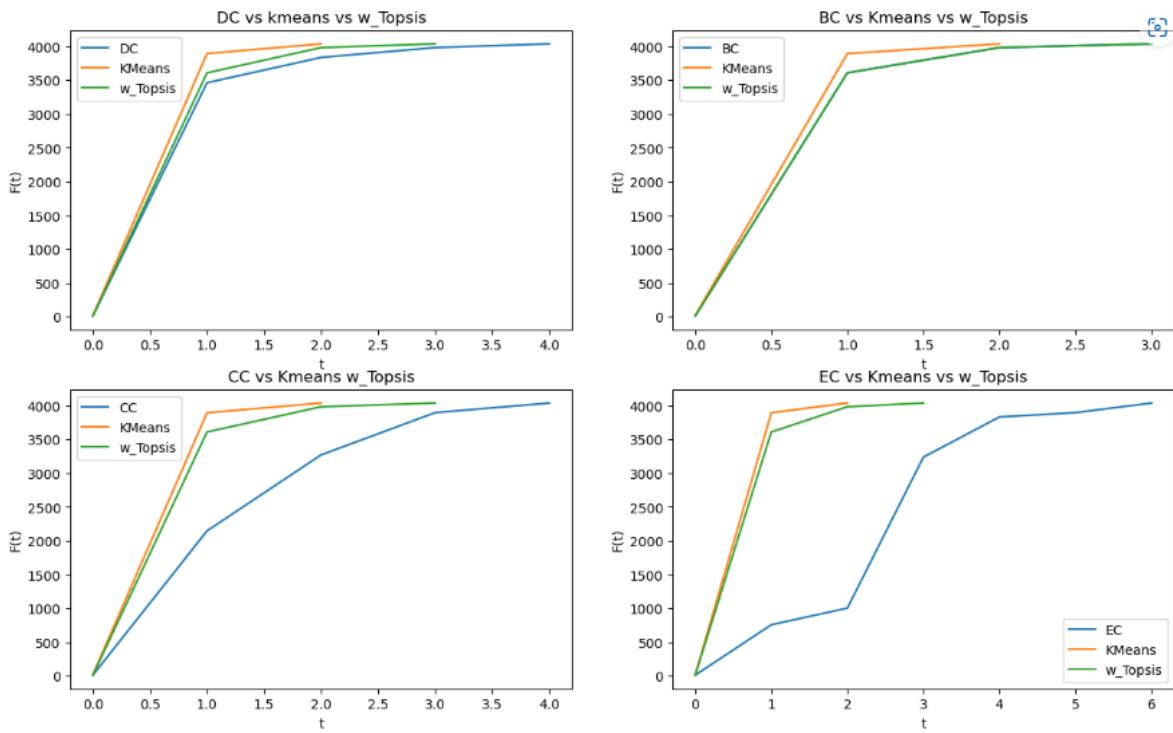


FIGURE 3.18 – Résultat du SI

là on remarque bien que l'algorithme du k-means nous a donné des résultats pertinents que celle de w-topsis.

Conclusion :

D'après les résultats précédents on peut conclure que la méthode w-topsis est la meilleure méthode pour identifier les noeuds influents dans ce type de réseaux car le résultat de l'algorithme du k-means avec ses différents approches n'est pas efficace ,même s'il est le meilleur après la division sur 13 cluster mais on peut généraliser car le choix de $k=13$ n'est qu'un choix aléatoire.

3.2 Processus sur Football

3.2.1 Application de Topsis et W-Topsis :

on a suit le même processus sur les données de Football et on a obtenu le résultat ci-dessous :

Node	DC	BC	CC	EC	C
1	0.105263	0.032490	0.423792	0.106503	0.782766
115	0.105263	0.020079	0.430189	0.121289	0.782345
29	0.105263	0.020895	0.423792	0.116303	0.765765
104	0.105263	0.016561	0.411552	0.122577	0.737870
65	0.096491	0.033533	0.422222	0.100915	0.731967
26	0.105263	0.019681	0.425373	0.112856	0.730856
13	0.105263	0.018819	0.422222	0.113703	0.725407
59	0.096491	0.022127	0.397213	0.109969	0.721082
25	0.105263	0.023070	0.420664	0.106250	0.715201
53	0.105263	0.014563	0.401408	0.120724	0.703854

FIGURE 3.19 – Les 10 premiers noeuds classés par TOPSIS ,DC, CC ,BC et EC

w_Node	DC	BC	CC	EC	C
65	0.096491	0.033533	0.422222	0.100915	0.976615
1	0.105263	0.032490	0.423792	0.106503	0.961193
107	0.096491	0.029161	0.435115	0.090415	0.853660
42	0.087719	0.028823	0.436782	0.079146	0.840744
75	0.096491	0.025187	0.404255	0.080489	0.724452
47	0.096491	0.024139	0.422222	0.089273	0.691313
81	0.096491	0.023836	0.423792	0.089921	0.681512
25	0.105263	0.023070	0.420664	0.106250	0.657666
84	0.096491	0.023046	0.402827	0.081215	0.655122
5	0.096491	0.022213	0.423792	0.103791	0.629590

FIGURE 3.20 – Les 10 premiers noeuds classés par W-TOPSIS ,DC, CC ,BC et EC

3.2.2 Evaluation par SI :

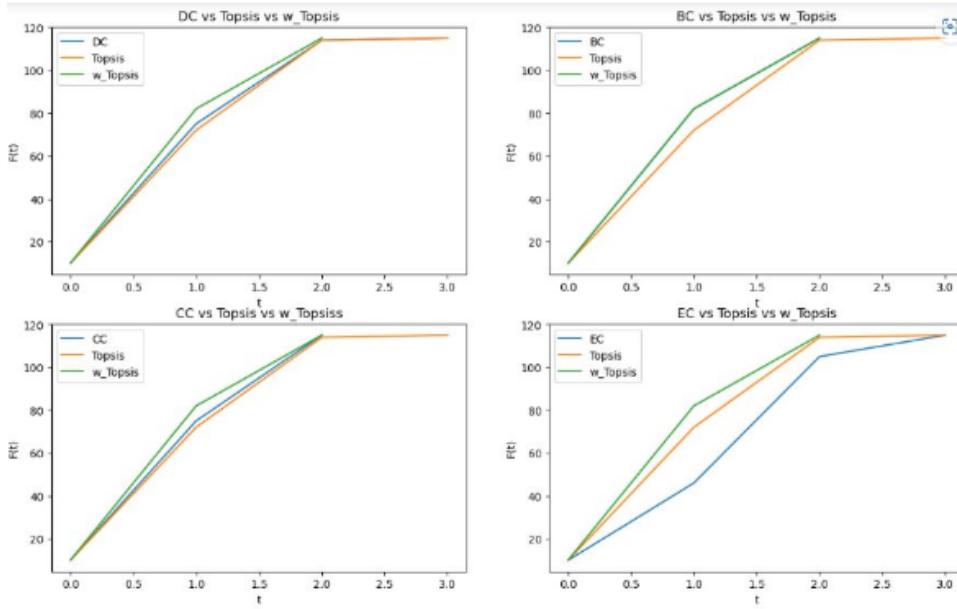


FIGURE 3.21 – résultat du SI

3.2.2.1 Discussion et interprétation des résultats :

Comme vous pouvez le constater, Les résultats de l'expérience sur le réseau Football montrent que notre méthode W-topsis est plus performante que les mesures de centralité traditionnelles telles que la centralité de proximité, la centralité d'interdépendance, et la centralité de degré. Elle est également presque aussi efficace que la centralité de vecteur propre. et elle plus efficace par rapport à Topsis.

3.2.3 Application de l'algorithme K-means :

Nous utiliserons la méthode K-means dans cette section du projet pour diviser notre ensemble de données en plus petits ensembles de données. Ensuite, nous comparons les deux résultats avec le modèle SI après avoir effectué la même procédure (w-topsis et mesures de centralité). Nous choisirons $k = 10$ comme nombre de clusters. et on a obtenu les résultats suivants :

cluster	Node_before_KMeans	Node_after_KMeans	distance from center
0	0	65	0.003756
1	1	1	0.002301
2	2	107	0.003754
3	3	42	0.007873
4	4	75	0.005705
5	5	47	0.005111
6	6	81	0.003046
7	7	25	0.007550
8	8	84	0.003952
9	9	5	0.004087

FIGURE 3.22 – Résultat du k-means

Comparaison entre les centroïdes du k-means et w-topsis et BC,CC et DC :

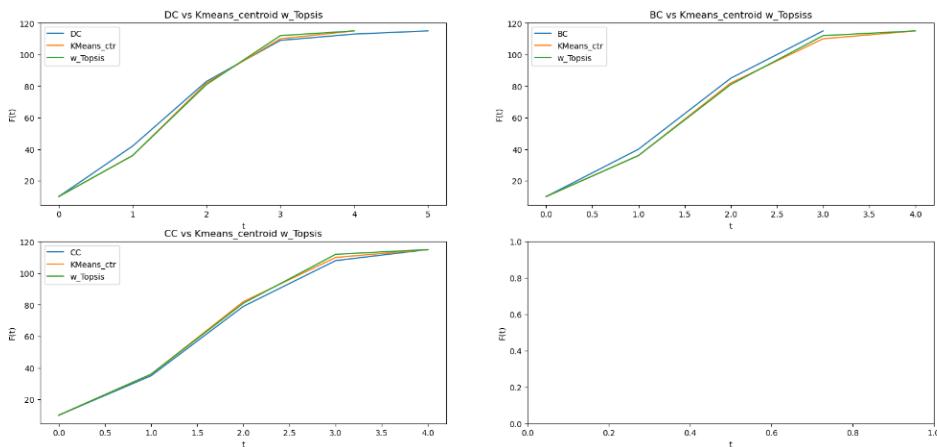


FIGURE 3.23 – Résultat SI

on remarque que BC et DC dépassent w-topsis et le k-means en termes de nombre moyen de noeuds infectés, tandis que cette dernière et le k-means sont peu meilleure que CC.

Application de K-means sur chaque cluster :

```

clusterS 0 :
    Node   DCN   BCN   CCN
    0     25     2     2     2
clusterS 1 :
    Node   DCN   BCN   CCN
    0     30     29    29    29
clusterS 2 :
    Node   DCN   BCN   CCN
    0    107    107   107   107
clusterS 3 :
    Node   DCN   BCN   CCN
    0     68     69    69    69
clusterS 4 :
    Node   DCN   BCN   CCN
    0     46     46    46    47
clusterS 5 :
    Node   DCN   BCN   CCN
    0    101    97   101   97
clusterS 6 :
    Node   DCN   BCN   CCN
    0     30     29    29    29

```

FIGURE 3.24 – résultatat de w-topsis sur chaque cluster

Comparaison entre w-topsis et le k-means avec w-topsis sur chaque cluster et BC,CC et DC

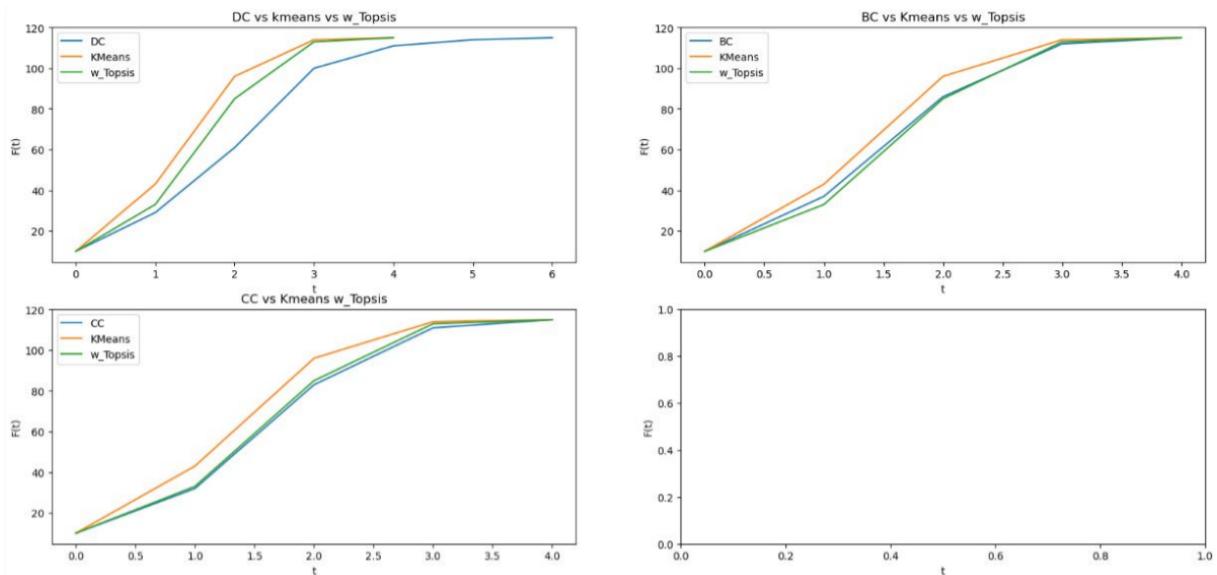


FIGURE 3.25 – résultatat SI

On constate bien que le k-means est plus efficace par rapport à w-topsis et par rapport à BC,CCet DC.

Conclusion : Pour ce types de réseaux on peut conclure que l'utilisation de l'algorithme du k-means est la meilleure méthode pour identifier les noeuds influents.

3.3 Processus sur Zachary

3.3.1 Application de Topsis et W-Topsis :

Pour ce réseau on a appliqué juste les méthode Topsis et W-topsis ; on a pas appliqué l'algorithme du k-means puisque le réseau est déjà petit donc on a pas besoin de le diviser.

Node	DC	BC	CC	EC	C
1	0.484848	0.437635	0.568966	0.355483	0.951239
34	0.515152	0.304075	0.550000	0.373371	0.840211
33	0.363636	0.145247	0.515625	0.308651	0.582125
3	0.303030	0.143657	0.559322	0.317189	0.529882
2	0.272727	0.053937	0.485294	0.265954	0.424332
32	0.181818	0.138276	0.540984	0.191036	0.345288
4	0.181818	0.011909	0.464789	0.211174	0.291307
9	0.151515	0.055927	0.515625	0.227405	0.291126
14	0.151515	0.045863	0.515625	0.226470	0.286220
24	0.151515	0.017614	0.392857	0.150123	0.223919

FIGURE 3.26 – les dix noeuds influents avec topsis

w_Node	DC	BC	CC	EC	C
1	0.484848	0.437635	0.568966	0.355483	0.991701
34	0.515152	0.304075	0.550000	0.373371	0.698990
33	0.363636	0.145247	0.515625	0.308651	0.341100
3	0.303030	0.143657	0.559322	0.317189	0.335060
32	0.181818	0.138276	0.540984	0.191036	0.316587
2	0.272727	0.053937	0.485294	0.265954	0.141261
9	0.151515	0.055927	0.515625	0.227405	0.134443
14	0.151515	0.045863	0.515625	0.226470	0.113237
20	0.090909	0.032475	0.500000	0.147911	0.077962
7	0.121212	0.029987	0.383721	0.079481	0.072268

FIGURE 3.27 – les dix noeuds influents avec w_{topsis}

3.3.2 Evaluation par SI :

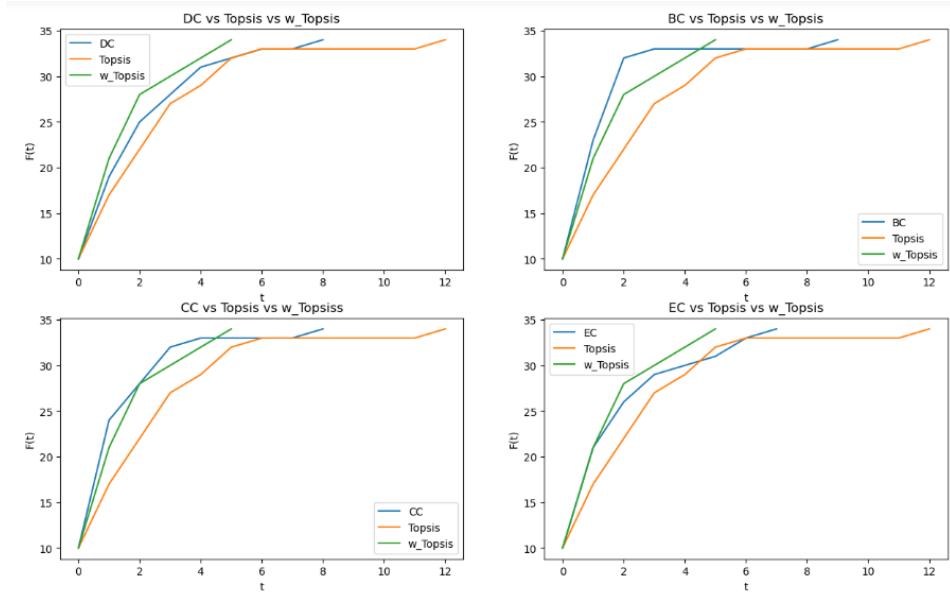


FIGURE 3.28 – Résultat SI

on remarque que w_{topsis} est plus efficace que $topsis$.

Conclusion :

on peut conclure que w-topsis est la méthode la plus efficace pour identifier les noeuds influents pour ce type de réseau.

Conclusion

D'après les deux exemples des réseaux qu'on a traité ,facebook et football, on peut conclure que pour identifier les noeuds influents dans les grands réseaux, la méthode w-topsis sera la bonne voie, ainsi que l'algorithme du k-means si on a bien choisi le nombre k des clusters. Cependant pour les réseaux de tailles moyennes ou les petits réseaux le k-means travaille bien et donne des résultats pertinents.

CHAPITRE 4

CONCLUSION GÉNÉRALE ET PERSPECTIVES

Pendant la période de réalisation de ce projet, nous avons eu la chance d'appliquer nos connaissances et nos compétences acquises pendant nos études de premier semestre du master.

Notre projet a pour but de détecter les influenceurs dans les réseaux complexes. On a analysé différents réseaux pour connaître l'efficacité des différents méthodes selon la taille du réseau.

La méthode TOPSIS a donné des résultats satisfaisants dans les grands réseaux, pour les petits réseaux, l'algorithme k-means est plus approprié ; les centroïdes qui en sont issus, peuvent mieux diffuser l'information au sein des clusters qui constituent le réseau.

En termes de perspectives, on pourra utiliser la closeness centrality au lieu de la distance euclidienne pour localiser les centroïdes dans la dataset . Comme nous avons l'intention d'utiliser des techniques d'apprentissage par renforcement qui visent à réduire l'intervention humaine.

Références

- <https://www.youtube.com/watch?v=usJ6RH8GCm0>
- D. Chen, L. Lü, M.-S. Shang, Y.-C. Zhang, T. Zhou, Identifying influential nodes in complex networks, *Physica A* 391 (2012) 1777–1787.
- Hu, J., Du, Y., Mo, H., Wei, D., Deng, Y. (2016). A modified weighted TOPSIS to identify influential nodes in complex networks. *Physica A : Statistical Mechanics and its Applications*, 444, 73-85.
- Bian, T., Hu, J., Deng, Y. (2017). Identifying influential nodes in complex networks based on AHP. *Physica A : Statistical Mechanics and its Applications*, 479, 422-436.
- Fei, L., Zhang, Q., Deng, Y. (2018). Identifying influential nodes in complex networks based on the inverse-square law. *Physica A : Statistical Mechanics and its Applications*, 512, 1044-1059.
- <https://www.sciencedirect.com/science/article/pii/S0378437113011552>
- <https://www.youtube.com/c/MachineLearnia>
- <https://github.com/MachineLearnia/Python-Machine-Learning>
- <https://www.mathweb.fr/euclide/les-graphes-en-python/>
- <https://tel.archives-ouvertes.fr/tel-00460708>
- <https://datascientest.com/apprentissage-non-supervise>
- <https://aclanthology.org/2018.jeptalnrecital-recital.9.pdf>
- <https://sites.google.com/a/uca.ma/sadgal/enseignement/enseiglpi-ia>
- <https://tel.archives-ouvertes.fr/tel-03640442/document>
- <https://github.com/ChekrounMohammed/Identification-of-top-k-nodes-using-TOPSISmethod- and-community-detection-in-complex-Networks>
- <https://github.com/achrafs758/A-dynamic-weighted-TOPSIS-method-for-identifyinginfluential-nodes-in-complex-networks/blob/main/RapportTopsisfinal.pdf>
- <https://snap.stanford.edu/data/ego-Facebook.html>
- <http://konect.cc/networks/>

