الجـــــــــمـهـوريـــــة الجـــــــــزائريـة الديمـــــــــــقراطيـة الشـــــــعبيـة
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
وزارة التـــــــــــــعليـم العـــــــالي و البـــــــــحث العـــــــلمـي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
جـــــــــامعة غـــردايـــــة
Université de Ghardaia
كليـة العلـــوم و التكنولـوجيـا
Faculté des Sciences et de Technologie
قسم الرياضيات و الإعلام الآلي
Département des Mathématiques et Informatique

MÉMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Informatiques

Spécialité : Systèmes Intelligents pour l'Extraction de Connaissances

Par : BITOUR Ahmed & KETILA Mohammed Amine

Sujet

Identification des facteurs les plus influents dans les algorithmes de détection de communautés

Soutenu publiquement le 25/06/2018 devant le jury composé de :

M. MAHDJOUB Youcef	MAB	Univ. Ghardaia	Président
M. ADJILA Abderrahmane	MAB	Univ. Ghardaia	Examinateur
M. KERRACHE Chaker Abdelaziz	MAB	Univ. Ghardaia	Directeur de mémoire

Année Universitaire 2017/2018

Dédicace

Je dédie ce modeste travail à :

Mes chers Parents,

Qu'ils ont œuvré pour ma réussite, pour tous leurs sacrifices, leurs amours, leurs tendresses, leurs soutiens et leurs prières tout au long de mes études, peut être fières et trouvant ici le résultat de ces longues années.

Mes chers Frères,

Pour leurs soutiens moraux et leurs encouragements.

Mes grands Mères et Pères.

Toute ma Grande Famille,

Pour leurs soutiens tout au long de mon parcours d'étude.

Tous mes Collègues,

Merci pour les bons moments qu'on a passé ensemble.

Tous mes Amis,

Vous partagerez toujours une partie de ma vie.

BITOUR Ahmed

Dédicace

Je dédie ce modeste travail à :

Mes très chers Parents

En témoignage de ma reconnaissance envers le soutien, les sacrifies et tous les efforts qu'ils ont fait pour mon éducation ainsi que ma formation

Mes chers Frères, et mes chères Sœurs

Toute ma Famille, et mes Amis

Et à Tous ceux qui ont contribué de près ou de loin pour que ce projet soit possible,
je vous dis Merci.

KETILA Amine

Remerciements

Au terme de ce travail, nous remercions tout d'abord, **Allah**, qui nous a guider le chemin de savoir et a ouvert les portes des connaissance.

Nous aimions remercier tous ceux qui nous avons accompagné tout au long de la réalisation de ce mémoire.

Nous exprimons toute nous reconnaissance envers notre promoteur **M. KERRACHE Chaker Abdelaziz** pour son aide sans réserve, ces conseils, ces orientations et sa riche expérience en recherche, son esprit d'analyse critique, sa rigueur scientifique tout on conservant une humilité qui impose le respect.

Nous adressons également nos sincères remerciements au membre de jury pour leur disponibilité, d'avoir fait l'honneur d'être examinateurs de mémoire et surtout d'avoir accepté de juger notre travail.

Ensuite, nous remercions également tous **les enseignants** qui nous avons enseigné et qui par leurs compétences nous ont soutenu dans la poursuite de nos études.

À nos **familles** et **amis** qui en nous encourageant nous ont permis de surmonter tous les obstacles.

Enfin, nous remercions toutes **les personnes** qui, de près ou de loin, ont apporté leur contribution à ce travail.

BITOUR & KETILA

ملخص

يعد اكتشاف التجمع في الشبكات أحد الموضوعات الأكثر شيوعا في علم الشبكات الحديثة. عادة ما تكون التجمعات أو المجموعات عبارة عن مجموعات من العقد ذات احتمال أكبر بأن تكون مرتبطة ببعضها البعض من أعضاء مجموعات أخرى, تم اقتراح العديد من الخوارزميات للكشف عن هذه التجمعات, ولكن أداء هذه الخوارزميات ليس جيدا في جميع الحالات. لهذا استخدمنا طريقة تحدد العوامل الرئيسية المؤثرة في هذه الخوارزميات. تسمى هذه الطريقة بالتحليل العاملي 2^k والتي تسمح باستخراج تأثير كل عامل مع تفاعلاته بالعوامل الأخرى على النتائج. بعد النتائج التي تم الحصول عليها باستخدام تطبيقنا الذي يعتمد على التحليل العاملي 2^k, وجدنا أن العامل الذي يمثل معامل المزيج (μ) هو الأكثر تأثيرا من بين العوامل الأخرى بنسبة 95.28%.
لذلك اقرحنا على الأعمال المستقبلية التركيز على هذا العامل للحصول على أداء ممتاز.

كلمات مفتاحية :

تحديد التجمعات, الشبكات, التحليل العاملي 2^k, الخوارزميات, العوامل.

Résumé

La détection communautaire dans les réseaux est l'un des sujets les plus populaires de la science des réseaux modernes. Les communautés, ou groupes, sont généralement des groupes de nœuds ayant une probabilité plus élevée d'être connectés les uns aux autres qu'aux membres d'autres groupes. De nombreux algorithmes ont été proposés pour détecter ces communautés, mais la performance de ces algorithmes n'est pas bonne dans tous les cas. Pour cela nous avons utilisé une méthode qu'identifie les principaux facteurs influes dans ces algorithmes. Cette méthode appelée Analyse factorielle 2^k permet d'extraire l'influence de chaque facteur avec ces combinaisons. Après les résultats obtenues à l'aide de notre application qui base sur l'analyse factorielle 2^k, nous avons trouvé le facteur qui représente le paramètre de mélange (μ) est le plus influent que d'autres facteurs, avec un pourcentage de 95.28%.

Donc on propose pour les travaux futures la concentration sur ce facteur pour obtenir des excellentes performances.

Mots clés :

Détection des communautés, Réseaux, Analyse factorielle 2^k, Algorithmes, Facteurs.

Abstract

Community detection in networks is one of the most popular topics in the science of modern networks. Communities, or groups, are usually groups of vertices with a higher probability of being connected to each other than members of other groups. Many algorithms have been proposed to detect these communities, but the performance of these algorithms is not good in all cases.For this we used a method that identify the main factors in these algorithms. This method called 2^k Factor Analysis which allows to extract the influence of every factor with these combination, After the results obtained using our application based on the 2^k factor analysis, we found the factor that represents the mixing parameter (μ) is the most influential of the others, with a percentage of 95.28%.

Therefore we propose for future work the concentration on this factor to obtain excellent performance.

Keywords :
Community detection, Network, 2^k Factor Analysis, Algorithms, Factors.

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

AFE	**A**alyse **F**actorielle **E**xploratoire
AFC	**A**alyse **F**actorielle **C**onfirmatoire
ACP	**A**alyse en **C**omposantes **P**rincipales
AFC	**A**alyse **F**actorielle **C**rrespondance
AFM	**A**alyse **F**actorielle **M**ultiple
AFMH	**A**alyse **F**actorielle **M**ultiple **H**iérarchique
AFD	**A**alyse **F**actorielle **D**iscriminante
Var	**Var**iance
E	**E**spérance
χ^2	**K**hi-deux
SFI	**I**nstitut **S**anta **F**e
Web	**W**orld wid**e** we**b**e
CNM	**C**lauset **N**ewman **M**oore
HE	**HE**uristique de degré
HE'	**HE**uristique de degré accidentellement
HN	**H**euristique de **N**ombre

Introduction

La science des réseaux est une discipline moderne couvrant les sciences naturelles, sociales et informatiques, ainsi que l'ingénierie [9].

Les réseaux, ou graphiques, sont constitués des nœuds et d'arêtes. Un bord relie généralement une paire de nœuds. Les réseaux se produisent dans une grande variété de contextes ; l'exemple de Facebook, qui est un grand réseau social, où plus d'un milliard de personnes sont connectées via des connaissances virtuelles.

La plupart des réseaux d'intérêt affichent la structure de la communauté [3], les communautés, également appelées "clusters" ou modules, sont des ensembles de nœuds qui sont fortement liés entre eux, mais faiblement liés avec le reste du graphe, les groupes de nœuds partagent probablement des propriétés communes et / ou jouent des rôles similaires dans le graphe.

L'identification des communautés apporte un éclairage nouveau sur la structure du graphe qu'elle est importante dans de nombreux contextes ; exemple : les communautés pourraient représenter des protéines ayant une fonction similaire dans les réseaux d'interaction protéine-protéine, des groupes d'amis dans les réseaux sociaux, des sites Web sur des sujets similaires sur le graphique Web, ... etc.

Pour détecter des communautés, plusieurs documents de recherche ont proposés des méthodes et des algorithmes, parmi ceux-ci l'algorithme de Newman & Giravan [3], Blondel & al [7], et de Rosvall & al [8] ... etc.

Les algorithmes ont tendance à modifier beaucoup leur performance même si le réseau change peu, à cause de l'effet des facteurs qui sont étaient utilisées dans les conceptions de ces algorithmes.

Dans ce mémoire, nous travaillons sur la déterminisation des facteurs les plus influents sur la performance des algorithmes.

A cet effet, nous avons utilisé la méthode d'analyse factorielle 2^k [10] qui est une méthode statistique basée sur des variables numériques et catégoriques et les résultats des expériences appliquées sur ces variables.

Elle s'utilise pour Prévoir et améliorer les résultats avec les performances par la déterminisation des facteurs les plus influents.

Ce mémoire est organisé comme suit :

Dans le chapitre 1, nous introduisons l'analyse factorielle et les méthode d'analyse des données, on a donné une définition générale pour l'analyse factorielle et ces types, les méthodes utilisées, en décrivant chacune pour choisir la plus apte pour atteindre notre objectif.

Pour le chapitre 2 nous expliquons la détection de communautés et ces domaines d'application puis on a détaillé les algorithmes utilisées dans ce domaine.

Le chapitre 3 représente les outils de travail, l'application d'analyse factorielle et les facteurs utilisés, un exemple applicatif, ainsi les résultats finales avec la discussion.

En fin on a donné une conclusion générale pour ce travail avec des propositions sur des travaux futur.

1 Analyse Factorielle

1.1 Introduction

Découvrir la nature des relations entre les variables est une partie plus importante dans n'importe quel domaine scientifique. Pour les nouveaux domaines la sélection des variables est mal précisée, il n'y a pas beaucoup d'accord entre les scientifiques concernant quelles variables devraient être liées les unes aux autres, et la nature des relations entre les variables est moins clairement spécifiée.

L'analyse factorielle représente un corps de croissance rapide des méthodes statistiques qui sont considérées comme une grande valeur dans les sciences les moins développées.

Les méthodes d'analyse factorielles peuvent aider les scientifiques pour définir précisément ces variables et décident quelles variables devraient être étudies et se relier les uns aux autres dans la tentative de développer leurs sciences.

a cet effet, dans ce chapitre nous parlerons sur le comportement d'analyse factorielle (Définition, Principe, Caractéristique, Type, et les méthodes), ainsi que d'autres techniques utilisables dans le domaine de l'analyse des données.

1.2 Définition

L'analyse factorielle est une technique statistique crée par *Charles Spearman*. Le but de cette technique c'est pour réduire le nombre des variables par l'utilisation qui décrie la variabilité entre des variables observées au moyen des variables latentes (non observées). Cela ce fait par le calcul des variables latentes comme une combinaison linéaire des variables observées [11].

L'emphase dans l'analyse factorielle est l'identification des « facteurs » sous-jacents qui pourraient expliquer les dimensions associées à la variabilité des données.

Cette méthode est utilisée en informatique dans l'imagerie médicale, satellitaire, dans le web sémantique (ontologies). Aussi utilisée dans "data mining" pour l'extraction des connaissances, et même dans l'analyses statistiques (approximation / régression),$\cdots$ etc.

Et dans d'autres domaines d'utilisation en psychologie, en sciences humaines / sociales, et d'une façon plus générale dans toute discipline faisant face à des grandes quantités de données.

L'analyse factorielle est un outil utile pour étudier les relations variables pour des concepts complexes.

Il permet aux chercheurs d'étudier des concepts qui ne sont pas faciles à mesurer directement en regroupant un grand nombre de variables en quelques facteurs sous-jacents interprétables.

Le facteur : est un ensemble de variables observées qui ont des profils de réponse similaires. Ils sont associés à une variable cachée (appelée variable latente) qui n'est pas directement mesurée.

Les facteurs sont énumérés en fonction des facteurs de pondération ou de variation des données, qu'ils peuvent être exprimés.

1.3 Types d'analyse factorielle

Il existe deux types principaux d'analyse factorielle :

1.3.1 Analyse factorielle exploratoire (AFE)

L'analyse factorielle été traditionnellement utilisée pour explorer la structure des facteurs sous-jacente possible d'un ensemble des variables mesurées sans imposer une structure préconçue sur le résultat [12].

AFE est une technique de réduction variable qui identifie le nombre des constructions latentes et la structure factorielle sous-jacente d'un ensemble des variables.

Il existe des facteurs communs « latents » de cette hypothèse (AFE) pour découvrir dans l'ensemble des données.

Le but c'est de trouver les plus petits nombres de facteurs communs qui expliqueront les corrélations [13].

1.3.2 Analyse factorielle confirmatoire (AFC)

C'est une technique statistique utilisée pour vérifier la structure factorielle d'un ensemble de variables observées.

L'AFC commence par une hypothèse sur le nombre de facteurs et les éléments qui chargent sur quels facteurs.

La plupart des chercheurs commencent avec un modèle dans lequel les éléments ne se chargent que d'un seul facteur (structure simple).

1.3.3 Différence entre AFE et AFC

La différence entre l'AFE et AFC est comme suit :

AFE

- Dans l'analyse factorielle exploratoire (AFE), tous les éléments chargent sur tous les facteurs, figure 1.1.
- L'AFE est parfois utilisé par les chercheurs, même s'ils ont une idée bien développée de la structure des facteurs et veulent la confirmer.

AFC

- Dans l'analyse factorielle confirmatoire (AFC), la plupart des chercheurs commencent par un modèle dans lequel les éléments ne se chargent que d'un seul facteur (structure simple), figure 1.1.
- Les modèles AFC peuvent être modifiés si le modèle ne correspond pas bien.

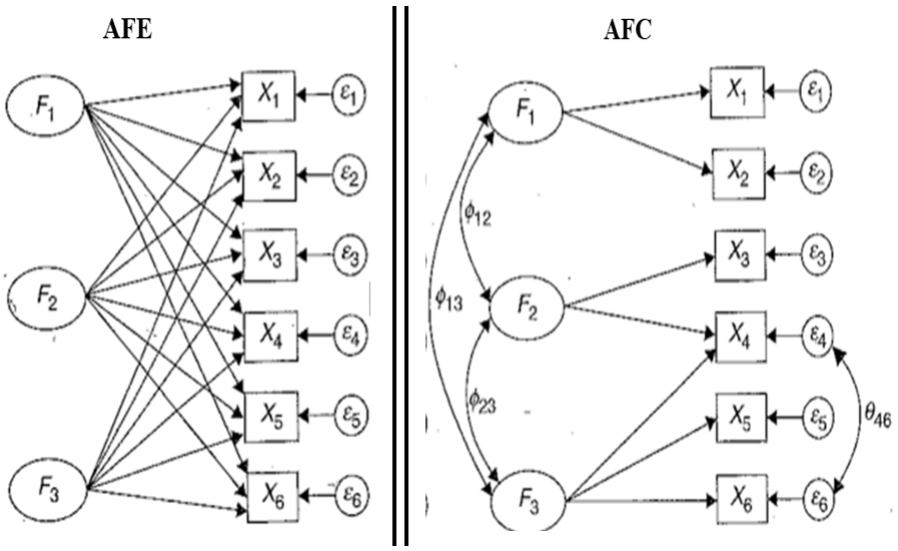


FIGURE 1.1 – AFE et AFC. Extrait de [1]

1.4 Méthodes d'analyse factorielles

Les méthodes d'analyse factorielle sont incontestablement des outils fondamentaux de l'analyse des tableaux de données qui visent essentiellement un but descriptif.

Il existe plusieurs méthodes pour faire l'analyse factorielle, chaque méthode consiste certains objectifs.

Nous représentons les principales méthodes de l'analyse factorielle :

1.4.1 Analyse en composantes principales (ACP)

L'analyse en composantes principales est une technique multivariée l'un des premières analyses factorielles qui fut conçue par Karl Pearson en 1901 [14].

L'ACP analyse le tableau de données dans lesquels où les observations sont décrites par plusieurs variables dépendantes quantitatives inter-corrélées.

Son but d'extraire les informations importantes de la table, de les représenter comme un ensemble de nouvelles variables orthogonales appelées composantes principales, et d'afficher le motif de similarité des observations et des variables comme des points dans les cartes.

On applique habituellement une ACP sur un ensemble de n variables aléatoires $X_1, \ldots, X_n$ connues à partir d'un échantillon de K réalisations conjointes de ces variables.

Cet échantillon de ces n variables aléatoires peut être structuré dans une matrice M à K lignes et n colonnes.

Chaque variable aléatoire $X_n = (X_{1,n}, \ldots, X_{k,n})$ a une moyenne $\bar{X_n}$ et un écart type σX_n.

On cherche une nouvelle matrice $\Pi(M)$ avec n nouvelles colonnes, chaque colonne représente une variable synthétique. Ce processus assure la minimisation de la corrélation entre les données.

L'ACP peut être généralisée en tant qu'analyse de correspondance (*AFC*) afin de gérer des variables qualitatives et en tant qu'analyse multi factorielles (AFM) afin de gérer des ensembles hétérogènes de variables.

1.4.2 Analyse factorielle des correspondances (AFC)

L'analyse factorielle des correspondances (AFC) est une méthode statistique d'analyse des données mise au point par Jean-Paul Benzecri, conçue pour les tableaux de contingence (appelés

aussi tableau de dépendance ou tableau croisé de Co-occurrence), Utilise pour analyser l'association entre deux variables qualitatives et déterminer qu'à hiérarchiser l'ensemble des dépendances entre les lignes et les colonnes du tableau.

Cette méthode est basée sur l'inertie, elle vise à rassembler en un nombre réduit de dimensions la plus grande partie de l'information initiale en s'attachant non pas aux valeurs absolues mais aux correspondances entre les variables [15].

On peut aussi appliquer l'AFC aux tableaux de mesures, aux tableaux des notes, des rangs et aux tableaux à valeurs logiques (0 ou 1) ... etc.

1.4.3 Analyse factorielle multiple (AFM)

L'Analyse factorielle multiple (AFM) utilisée pour analyser un ensemble d'observations décrites par plusieurs groupes de variables. Le nombre de variables dans chaque groupe peut être différèrent et la nature des variables (nominales ou quantitatives) peut varier d'un groupe à l'autre, mais les variables doivent être de même nature dans un groupe de données.

L'analyse dérive une image intégrée des observations et des relations entre les groupes de variables.

L'AFM cherche les structures communes présentées dans tout ou partie de ces ensembles. Le but de l'AFM est d'intégrer différents groupes de variables décrivant les mêmes observations.

L'AFM est effectuée en deux étapes :

1. L'ACP est effectuée sur chaque ensemble de données qui est ensuite "normalisé" en divisant tous ses éléments par la racine carrée de la première valeur propre obtenue à partir de son ACP.
2. Les ensembles de données normalisées sont fusionnées pour former une matrice unique et une ACP globale est effectuée sur cette matrice.

Les ensembles de données individuels sont ensuite projetés sur l'analyse globale pour analyser les points communs et les divergences [16].

L'AFM s'avère très utile pour analyser des enquêtes lorsque les questions peuvent être regroupées par thèmes, ou lorsque les mêmes questions sont posées à plusieurs intervalles de temps.

1.4.4 Analyse factorielle multiple hiérarchique (AFMH)

L'Analyse factorielle multiple hiérarchique (AFMH) c'est une méthode qui généralise l'AFM au cas où les variables sont structurées selon une hiérarchie.

Dans l'AFMH une succession d'AFM est appliquée à chaque nœud de la hiérarchie afin d'équilibrer les groupes de variables dans chaque nœud, en passant par l'arborescence hiérarchique de bas en haut.

L'AFMH fournit des sorties analogues à celles de l'AFM mais aussi des aides à l'interprétation spécifique, telle que la représentation de chacun des nœuds d'une part et la représentation des individus décrits par chacun des nœuds, d'autre part [17].

1.5 Analyse Factorielle Discriminante (AFD)

L'analyse factorielle discriminante une des nombreuses méthodes de l'analyse discriminante. AFD est une technique statistique qui vise à décrire, expliquer et prédire l'appartenance à des groupes prédéfinis (classes, modalités de la variable à prédire, etc) d'un ensemble d'observations (individus, exemples, ...) à partir d'une série de variables prédictives (descripteurs, variables exogènes, etc).

On peut distinguer très souvent deux grandes approches :

1.5.1 Approche descriptive

Est une technique de statistique exploratoire qui travaille sur un ensemble de n observations décrites par J variables, répartis en K groupes. Elle vise à produire un nouveau dispositif de représentation, constitué de combinaisons linéaires des variables initiales, qui sert à séparer au mieux les K catégories.

Elle est aussi une technique descriptive et explicative car elle propose une représentation graphique qui sert à visualiser les proximités entre les observations, appartenant au même groupe ou non. Et nous avons la possibilité d'interpréter les axes factoriels, combinaisons linéaires des variables initiales, et ainsi comprendre les caractéristiques qui distinguent les différents groupes.

1.5.2 Approche prédictive

Il s'agit dans ce cas de construire une fonction de classement qui sert à prédire le groupe d'appartenance d'un individu à partir des valeurs prises par les variables prédictives. Dans cette approche sont regroupées des méthodes de classification qui nécessitent une connaissance des classes préexistantes ; par exemple le domaine de la reconnaissance des formes c'est ce qui est appelé classification supervisée ou encore à apprentissage supervisé. Parmi ces méthodes peuvent être compter la régression logistique, les k-plus proches voisins, les arbres de décisions (souvent employées pour la segmentation) ou encore des méthodes issues de l'intelligence artificielle telles que le perceptron multicouche et les autres réseaux de neurones ... etc.

La plupart des méthodes qui ne sont pas issues de l'intelligence artificielle peuvent être décrites par deux étapes :

- L'étape de discrimination qui cherche à déterminer sur les données d'apprentissage une fonction qui discrimine au mieux les données.
- L'étape de classement qui cherche à affecter une nouvelle donnée à une classe, à l'aide de la fonction établie dans l'étape précédente.

L'arbre de décision est une méthode discrimination. La représentation sous forme d'arbre permet une interprétation rapide et aisée des résultats.

La construction de l'arbre (i.e. l'étape de discrimination) est effectuée sur les données d'apprentissage, puis l'étape de classement peut être réalisée pour de nouveaux individus.

L'idée de la construction est simple, et se décompose comme suit :

- Chercher la variable qui produit la meilleure division.
- Diviser en deux nœuds intermédiaires, les individus selon cette variable.
- Chercher les variables qui produisent les meilleures divisions des nœuds intermédiaires.
- Poursuivre ainsi jusqu'à n'obtenir que des nœuds terminaux.

L'AFD est utilisée dans des nombreux domaines ; en médecine pour la prédiction de maladies, en météorologie pour prédire un risque d'avalanche, en finance pour prédire un comportement boursier, en traitement d'images pour la reconnaissance des formes [18].

1.6 Régression

La régression est une méthode statistique utilisée pour analyser la relation entre une ou plusieurs variables indépendantes et une variable dépendante. Elle est l'une des méthodes statistiques les plus utilisées dans diverses sciences, car elle décrit la relation entre les variables sous une forme équivalente.

L'analyse de régression est un outil important pour la modélisation et l'analyse des données. Ici, nous ajustons une courbe / ligne aux points de données, de telle sorte que les différences entre les distances des points de données de la courbe ou de la ligne sont minimisées, figure1.2 .

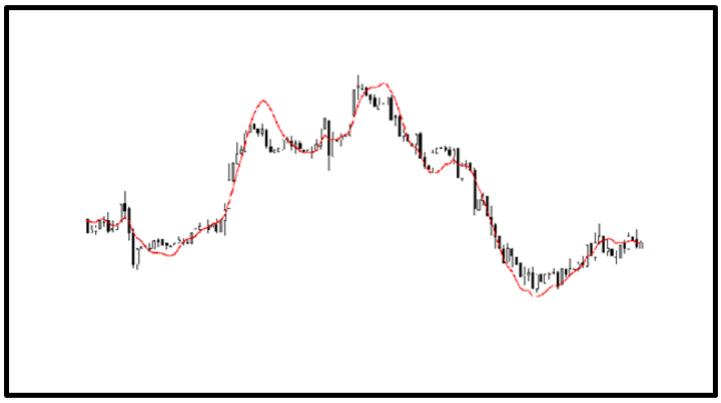


FIGURE 1.2 – Courbe de régression. Extrait de [2]

1.6.1 Utilisation de la régression

- Dans *la description des données* vous pouvez résumer et décrire un ensemble de données en trouvant une équation de régression qui décrit cette relation.
- Dans *la prédire* la réponse peut être évaluée et prédite pour aider à prendre des décisions.
- Dans *le contrôle* lorsque vous trouvez l'équation de régression qui décrit les données, il est possible de contrôler les valeurs de la variable dépendante en changeant les valeurs des variables indépendantes.

1.6.2 Types de régression

Il existe plusieurs type de régression, nous avons présenté deux types les plus utilisée :

Régression linéaire

C'est l'une des techniques de modélisation les plus connues. La régression linéaire est généralement parmi les premiers sujets que les gens choisissent, tout en apprenant la modélisation prédictive.

Dans cette technique, la variable dépendante est continue, les variables indépendantes peuvent être continues ou discrètes, et la nature de la droite de régression est linéaire.

La régression linéaire établit une relation entre la variable dépendante (Y) et une ou plusieurs variables indépendantes (X) en utilisant une ligne droite de meilleur ajustement (également appelée ligne de régression).

Il est représenté par une équation $Y = a + b * X + e$, où a est l'ordonnée à l'origine.

b est la pente de la droite et e est le terme d'erreur.

Cette équation peut être utilisée pour prédire la valeur de la variable ciblée en fonction d'une variable prédictive donnée 2.1.

Il existe deux type de régression linéaire :

- *La régression linéaire simple* du premier degré avec une variable indépendante.
- *La régression linéaire multiple* inclue plusieurs variables indépendantes.

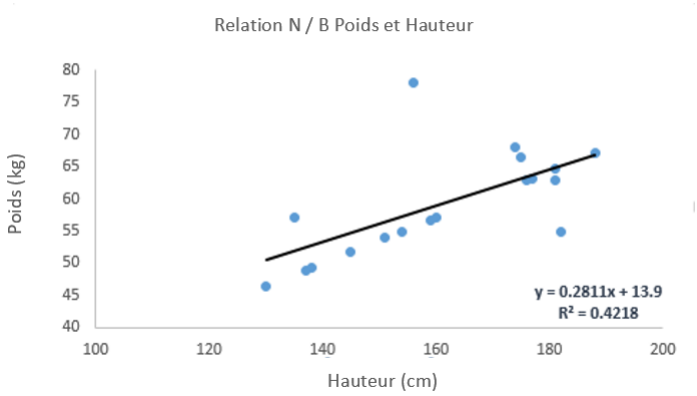


FIGURE 1.3 – Droite de régression . Extrait de [2]

Régression non linéaire

La régression non linéaire est une technique statistique qui aide à décrire les relations non linéaires dans les données expérimentales. Les modèles de régression non linéaire sont généralement supposées à être paramétriques. Le modèle étant décrit comme une équation non linéaire.

Typiquement, les méthodes d'apprentissage automatique sont utilisées pour la régression non-linéaire non-paramétrique.

La régression paramétrique non linéaire modélise la variable dépendante (appelée aussi réponse) en fonction d'une combinaison de paramètres non linéaires et d'une ou de plusieurs variables indépendantes (appelées prédicteurs). Le modèle peut être uni-varié (variable à réponse unique) ou multivarié (variables à réponses multiples).

Les paramètres peuvent prendre la forme d'une fonction exponentielle, trigonométrique, de puissance ou toute autre fonction non linéaire.

Pour déterminer les estimations de paramètres non linéaires, un algorithme itératif est généralement utilisé.

$$y = f(X, \beta) + \epsilon \tag{1.1}$$

Où β représente des estimations de paramètres non linéaires à calculer, et ϵ représente les termes d'erreurs.

1.7 Khi-deux

Est un test statistique où la statistique de test suit une loi du khi-deux dénote par(χ^2) sous l'hypothèse nulle, Ce test a été proposé par le statisticien Karl Pearson en 1900 [19].

La loi du khi-deux existe dans la théorie de probabilité et la statistique. Cette loi elle est centrée avec k-degré de liberté, tel que ce degré de liberté c'est le nombre de paramètre à estimer. En autre terme c'est la loi de la somme de carré de k lois normale centrées réduites indépendantes.

En utilisant cette loi sur les tests statistiques (test du χ^2) et généralement elle s'applique dans le test d'hypothèse à certains seuils (indépendance notamment).

On peut définir cette loi comme suit : Soient $X_1, \ldots, X_k$. de k variables aléatoires indépendantes de même loi (loi normale $N(0, 1)$).

Posons :

$$\chi^2 = \sum_{i=1}^{k} X_i^2 \tag{1.2}$$

Par définition, la variable aléatoire χ^2 suit une loi du khi-deux à k degrés de liberté on note cette loi $\chi^2(k)$.

Certaine propriété pour cette loi, l'Espérance (E) de la variable aléatoire χ^2 égale k on notera :

$$E[\chi^2] = K \tag{1.3}$$

Et la variance (Var) de variable aléatoire de χ^2 égale 2k on notera :

$$Var(\chi^2) = 2k \tag{1.4}$$

Alors le test de khi-deux (χ^2) c'est un test des hypothèses qui compare la loi de distribution observe de vos données à une loi attendue, et détermine si la différence entre deux distributions de fréquences est attribuable à l'erreur d'échantillonnage (le hasard) ou est suffisamment grande pour être statistiquement significative.

Si la différence entre les deux distributions est réduite, l'hypothèse nulle sera acceptée. Si la différence est grande, l'hypothèse nulle sera rejetée.

Dans ce dernier cas, on parlera d'une différence statistiquement significative parce que l'écart entre les deux distributions est trop important pour être expliqué par le hasard seulement, une différence réelle existe donc.

Il existe plusieurs types de test du khi-deux telle que le test d'ajustement (ou d'adéquation), test d'association, d'indépendance . . . etc.

Pour le test d'ajustement ce test est couramment utilisé pour tester l'association des variables dans les tables bidirectionnelles où le modèle d'indépendance présumé est évalué par rapport aux données observées.

En général, la statistique du test khi-deux est de forme :

$$\chi^2 = \sum \frac{(observe - attendu)^2}{attendu} \tag{1.5}$$

Si le calcul du test est grand, les valeurs observées et attendues ne sont pas proches et le modèle est un mauvais ajustement pour les données.

1.8 Analyse factorielle 2^k

L'analyse factorielle 2^k est utilisé pour déterminer l'effet des k facteurs, chacun d'entre eux à deux alternatives ou niveaux.

Cette classe d'analyse factorielle mérite une discussion spéciale car elle est facile à analyser et aide à trier les facteurs dans l'ordre d'impact. Au début d'une étude de performance, le nombre de facteurs et leurs niveaux est généralement élevé. Une analyse factorielle complète avec un si grand nombre de facteurs et de niveaux peut ne pas être la meilleure utilisation de l'effort disponible.

La première étape devrait consister à réduire le nombre de facteurs et à choisir les facteurs qui ont un impact significatif sur la performance [10].

Très souvent, l'effet d'un facteur est unidirectionnel, c'est-à-dire que la performance diminue continuellement ou augmente continuellement lorsque le facteur est augmenté du minimum au maximum.

Dans de tels cas, nous pouvons commencer par expérimenter au niveau minimum et maximum du facteur. Cela nous aidera à décider si la différence de performance est suffisamment importante pour justifier un examen détaillé.

Dans le chapitre 3, nous expliquerons les concepts d'analyse factorielle 2^k.

1.9 Conclusion

L'analyse factorielle fournit au chercheur une synthèse et une réduction des données. Il est couramment utilisé pour réduire les variables dans un ensemble plus petit pour gagner du temps et faciliter les interprétations plus faciles. l'objectif général de cette techniques est de résumer l'information.

Dans le chapitre suivant nous allons présenter le domaine de détection des communautés a laquel nous expérimentons l'analyse factorielle, et les algorithmes utilisés dans ce domaine.

2 Détection de Communauté

2.1 Introduction

La science des réseaux est une discipline moderne couvrant les sciences naturelles, sociales et informatiques, biologie et neuroscience, ainsi que l'ingénierie.

Cette science moderne est probablement le domaine le plus actif de la nouvelle science interdisciplinaire des systèmes complexes.

Beaucoup de systèmes complexes peuvent être représentés comme des réseaux, où les parties élémentaires d'un système et leurs interactions mutuelles sont des nœuds et des liens, respectivement.

Les réseaux se produisent dans une grande variété de contextes ; exemple, Facebook est un grand réseau social, où plus d'un milliard de personnes sont connectées via des connaissances virtuelles.

Les systèmes complexes sont généralement organisés en compartiments, qui ont leur propre rôle et fonction.

Dans la représentation en réseau, de tels compartiments apparaissent comme des groupes de nœuds ayant une probabilité plus élevée d'être connectés les uns aux autres qu'aux membres d'autres groupes, bien que d'autres modèles soient possible [4].

Ces sous-graphes sont appelés communautés, ou modules, et se produisent dans une grande variété de systèmes en réseau. Il s'agit d'une description qualitative et aucune définition mathématique commune n'a été convenue, comme l'illustre toute une sous-classe de méthodes qui fonctionnent avec les définitions opérationnelles de la communauté réseau.

La recherche de compartiments peut éclairer l'organisation de systèmes complexes et leur fonction. Par conséquent, la détection des communautés dans les réseaux est devenue un problème fondamental dans la science des réseaux.

Dans la figure 2.1, nous montrons un réseau de collaboration de scientifiques travaillant à l'Institut de Santa Fe (SFI), New Mexico, Les nœuds sont des scientifiques, les arêtes rejoignent les coauteurs, les arêtes sont concentrées dans des groupes de nœuds représentants des scientifiques travaillant sur le même sujet de recherche, où les collaborations sont plus naturelles.

De même, les communautés pourraient représenter des protéines ayant une fonction similaire dans les réseaux d'interaction protéine-protéine, des groupes d'amis dans les réseaux sociaux, des sites Web sur des sujets similaires sur le graphique Web,…etc.

L'identification des communautés peut donner un aperçu de la façon dont le réseau est organisé.

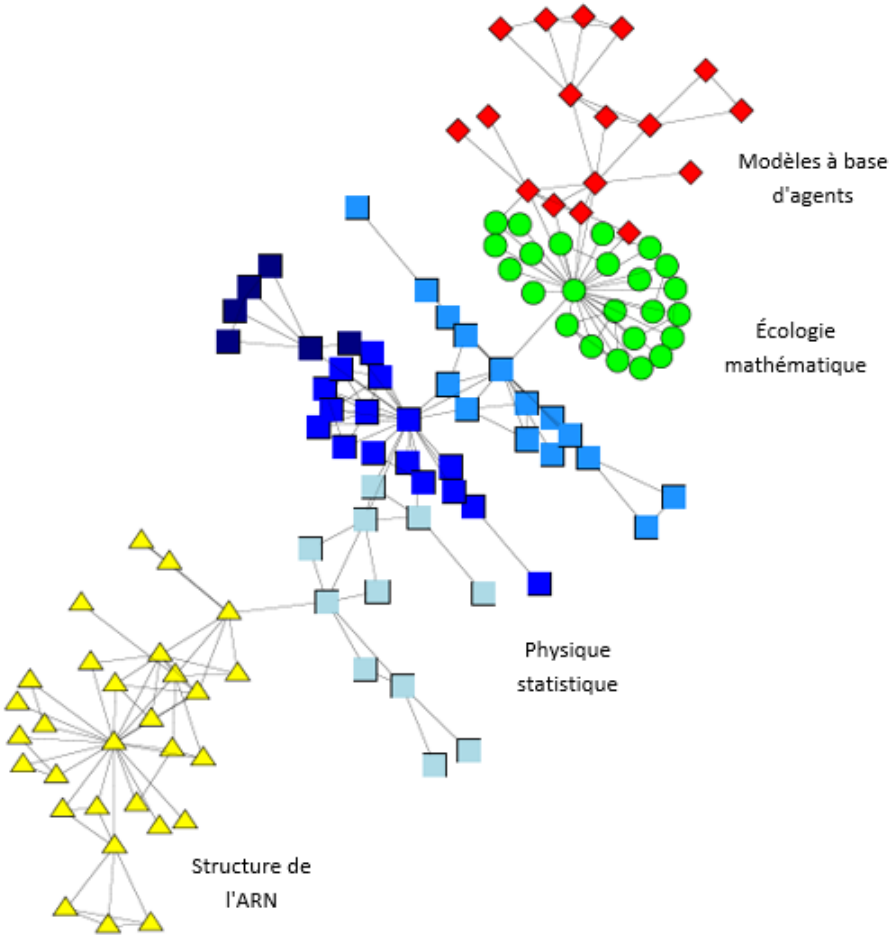


FIGURE 2.1 – Réseau de collaboration de scientifiques travaillant à l'Institut Santa Fe (SFI). Extrait de [3].

La détection communautaire dans les réseaux, également appelée "clustering" (regroupement) de graphe ou réseaux.

Le "clustering" est étroitement lié à l'apprentissage non supervisé dans les systèmes de reconnaissance des formes [20], est aussi un problème mal défini, il n'y a pas de définition universelle pour des objets que l'on devrait chercher, et il n'y a toujours pas d'accord entre les chercheurs sur ce à quoi ressemble un réseau avec les communautés.

De nombreuses méthodes ont été développées, en utilisant des outils et des techniques de disciplines telles que la physique, la biologie, les mathématiques appliquées, les sciences informatiques et sociales.

Cependant, il n'est pas encore clair quels algorithmes sont fiables et doivent être utilisés dans les applications. La question de la fiabilité elle-même est délicate, car elle nécessite des définitions partagées de la communauté et de la partition, qui sont actuellement manquantes.

Il existe plusieurs types de communauté chacun et sa définition, comme la communauté forte est un sous-graphe dont les nœuds ont une probabilité plus élevée d'être liés à chaque nœud du sous-graphe qu'à tout autre nœud du graphe. Et la communauté faible est un sous-graphe tel que la probabilité d'arête moyen de chaque nœud avec les autres membres du groupe dépasse la probabilité d'arête moyenne du nœud avec les nœuds de tout autre groupe.

La figure 2.2 représente une communauté forte et faible.

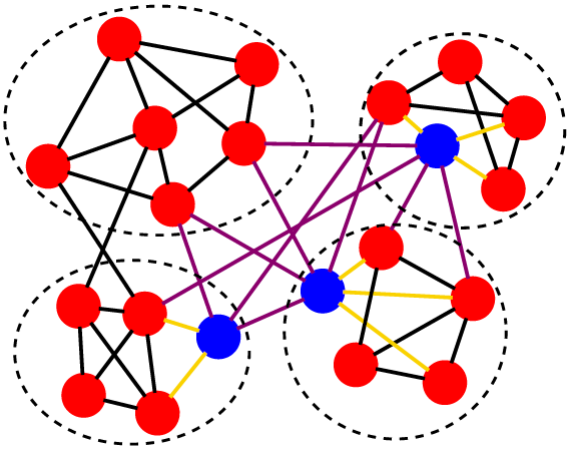


FIGURE 2.2 – Communautés fortes et faibles. Extrait de [4].

La figure 1.2 représente trois des sous-graphes qui ne sont pas des communautés fortes selon Radicchi et al [21], car certains nœuds (indiqués en bleu) ont un degré externe plus grand que leur degré interne (les bords interne et externe de ces nœuds sont colorés respectivement en jaune et en magenta).

2.2 Domaine d'application

Il existe de nombreuses applications pour la détection communautaire, parmi ces applications les réseaux d'information, dans n'importe quel réseau de communication, le "clustering" de graphe sert un outil d'analyse, de modélisation, de prédiction de la fonction, de l'utilisation et de l'évolution du réseau.

Les applications incluent l'analyse commerciale, l'amélioration de l'infrastructure et l'identification des utilisations anormales ...etc.

Dans les réseaux informatiques, le "clustering" peut être utilisé pour identifier les sous-structures pertinentes et analyser la connectivité à des fins de modélisation ou d'optimisation structurelle, dans le World Wide Web, le "clustering" des documents hypertextes, représentants chaque page Web par un nœud et chaque lien hypertexte par une arête, permet d'identifier des sujets et d'autres entités formées par plusieurs documents interconnectés.

Le "clustering" des pages Web aide à identifier les sujets et à regrouper des pages similaires, cela ouvre des applications dans la technologie des moteurs de recherche.

L'algorithme PageRank[22]utilisé par Google pour évaluer la qualité des sites Web assigne d'abord à chaque page Web la même valeur d'importance, qui est ensuite répartie itérativement uniformément entre tous ses voisins. Le but de l'algorithme PageRank est d'atteindre une distribution de valeur proche d'un état convergé stationnaire en utilisant relativement peu d'itérations. La quantité d'importance laissée à chaque nœud à la fin de l'itération devient le PageRank de ce nœud, plus le meilleur est élevé.

En ce qui concerne les services de téléphonie et de discussion internet comme WhatsApp, Skype, Messenger, des statistiques d'utilisation intéressantes pour optimiser les configurations matérielles et logicielles peuvent être obtenues en représentant chaque utilisateur comme un nœud et en plaçant des arêtes (pondérés) entre deux utilisateurs. Communiquer sur le système. Une analyse similaire peut aider les opérateurs de télécommunications traditionnels à identifier les "clusters" d'appels fréquents, i.e. groupes de personnes qui s'appellent tous principalement (comme des familles, des groupes d'amis ou des collègues), et donc mieux concevoir et cibler les offres largement répandues sur les tarifs spéciaux pour l'appel à un nombre limité de numéros de téléphone prédéfinis.

Le "clustering" des informations sur l'appelant peut également aider à identifier les changements dans le modèle de communication d'un client donné ; lorsque de longs appels sont effectués à l'extérieur du "cluster", le téléphone peut être volé.

Pour la détection de la fraude, la durée des appels et une intégration géographique serait très utiles pour déterminer ce qui forme le groupe de « destinations normales d'appels » pour un client spécifique et quels appels sont « hors de l'ordinaire ». Les algorithmes de "clustering" sont également utilisés dans la conception structurelle et le fonctionnement de réseaux [23]. Pour les réseaux avec une topologie dynamique, avec des changements fréquents dans la structure de périphérie, les méthodes de "cluster" local s'avèrent utiles, car les nœuds de réseau peuvent prendre des décisions locales pour modifier la classification afin de mieux refléter la topologie réseau actuelle [24].

Ulas & al [25] applique le "clustering" sur le graphique vidéo YouTube pour générer des "clusters" des vidéos nommés avec un contenu cohérent, ils considèrent le graphique YouTube, où chaque vidéo est un nœud et l'arête entre les noeuds capture leur similarité, qui pourrait être défini de plusieurs façons. Ils utilisent le graphique induit par le co-visionnement de vidéos par des utilisateurs dans des sessions d'utilisateurs YouTube anonymes. Deux vidéos ayant une valeur de co-visionnement élevées seront considérées comme similaires. Notez qu'ils construisent le graphique vidéo YouTube en fonction des statistiques de co-surveillance, mais utilisent également des fonctionnalités de texte pour affiner le "clustering".

Dans le domaine des Systèmes base de données, lors du stockage d'un grand ensemble de données, une question clé est de savoir comment regrouper les données dans des pages dans la mémoire physique. Bradley et al. [26] utilisent un algorithme itératif comme k-means pour déterminer un "clustering" pour une base de données volumineuse en une seule analyse en utilisant un tampon mémoire limité.

Pour les réseaux biologiques et sociologiques, dans le domaine de la bio-informatique, les tâches de classification de graphe traitent généralement la classification des données d'expression génique [27]. et les interactions protéiques figure 2.3. Une autre application biologique du "clustering" est la propagation épidémique.

Les applications de "clustering" local dans les réseaux sociaux comprennent l'identification des groupes d'individus « exposés » à l'influence d'un certain individu d'intérêt ; exemple, identifier des réseaux terroristes lorsqu'un membre est connu où localiser des personnes potentiellement infectées lorsqu'un individu infecté et contagieux est rencontré.

Dans la société de l'information actuelle, l'étude des réseaux sociaux tend à chevaucher l'étude des réseaux d'information, car la popularité et l'importance de la messagerie électronique sont devenues écrasantes.

D'autre domaine d'application comme l'analyse de "clustering" du réseau mondial de transport aérien donnée par Guimerà et al. [28]. Le "clustering" sert également dans la fabrication, où l'identification des "clusters" des pièces similaires aide à faciliter la ligne de production « appelé technologie de groupe ».

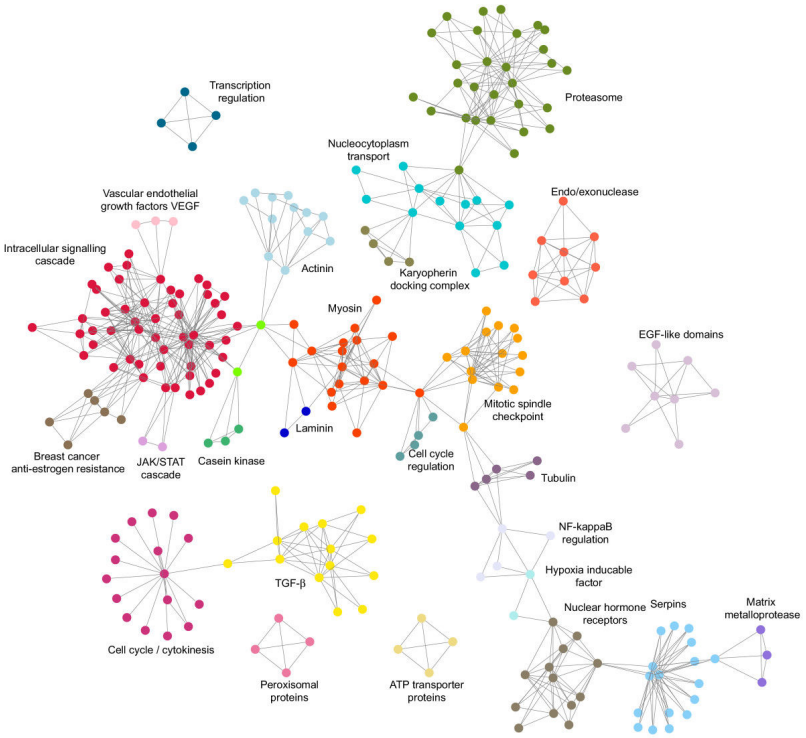


FIGURE 2.3 – Identifier les communautés de protéines par analyse de groupe. Extrait de [5].

2.3 Définition des Mots-clés

Le domaine de détection des communautés contient de nombreux mots-clés, on représente certains d'entre eux que nous avons besoin.

2.3.1 Arête (m)

Les arêtes qui dénote par (m) sont des liens entre les nœuds, chaque deux nœuds consiste un seul arête ; par exemple, en peut représenter l'amitié entre deux personnes par un arête ... etc.

2.3.2 Degré interne (k^{int})

1. Le degré interne d'un nœud i qui dénote par k_i^{int} en ce qui concerne un sous-graphe c, c'est le nombre d'arêtes se connectant aux nœuds de c. En peut exprimés par la matrice d'adjacence comme suit :

$$k_i^{int} = \sum_{j \in c} A_{ij} \tag{2.1}$$

2. Le degré interne d'un communauté C qui dénote par k_C^{int}, c'est la somme des degrés internes des nœuds de C. Il est égal au double du nombre m_C des arêtes internes, chaque arête contribuant à deux unités de degré.

   En peut exprimés par la matrice d'adjacence comme suit :

$$k_C^{int} = \sum_{i,j \in C} A_{ij} \tag{2.2}$$

2.3.3 Degré externe (k^{ext})

1. Le degré externe d'un nœud i qui dénote par k_i^{ext} en ce qui concerne un sous-graphe C, c'est le nombre d'arêtes se connectant aux nœuds de le reste du graphique.
   En peut exprimés par la matrice d'adjacence comme suit :

$$k_i^{ext} = \sum_{j \notin c} A_{ij} \tag{2.3}$$

2. Le degré externe d'une communauté c qui dénote par k_c^{ext}, c'est la somme des degrés externes des nœuds de C. Il donne le nombre d'arêtes externes du sous-graphe C.
   En peut exprimés par la matrice d'adjacence comme suit :

$$k_c^{ext} = \sum_{i \in c, j \notin c} A_{ij} \tag{2.4}$$

2.3.4 Degré total (k)

1. Le degré total d'un nœud i dénote par k_i, c'est la somme de degré interne et externe de nœud i.
2. Le degré total d'une communauté c dénote par k_c, c'est la somme de degré interne et externe de communauté c.

2.3.5 Conductance (C_c)

La conductance qui dénote par C_c, c'est le rapport entre le degré externe et le degré total de c :

$$C_c = \frac{k_C^{ext}}{k_C} \tag{2.5}$$

2.3.6 Modularité

Une classe populaire d'algorithmes de détection de communauté cherche à optimiser la soi-disant modularité de l'assignation de la communauté.

La modularité est une métrique proposée par Newman et Girvan [29]. Il quantifie la qualité d'une assignation communautaire en mesurant combien les connexions sont plus denses au sein des communautés par rapport à ce qu'elles seraient dans un type particulier de réseau aléatoire.

La modularité d'une partition est une valeur scalaire comprise entre -1 et 1 qui mesure la densité des liens au sein des communautés par rapport aux liens entre les communautés.

La définition mathématique de la modularité comme suit :

$$Q = \frac{1}{2m} \sum_{i,j} (A_{ij} - \frac{k_i k_j}{2m}) \delta(c_i, c_j) \tag{2.6}$$

$\delta(c_i, c_j)$ c'est 1 lorsque les nœuds i et j sont assignés à la même communauté et 0 sinon.

2.3.7 Information mutuelle normalisée

Tester un algorithme sur n'importe quel graphe avec une structure de communauté intégrée implique également de définir un critère quantitatif pour estimer la qualité de la réponse donnée par l'algorithme par rapport à la réponse réelle attendue. Cela peut être fait en utilisant des mesures de similarité appropriées.

Dans notre teste nous avons choisi une mesure dit *l'information mutuelle normalisée* qui adopté par Danon & al [30].

$$I_{norm}(X,Y) = \frac{2 \times I(X,Y)}{H(X) + H(Y)} \tag{2.7}$$

$X =$ étiquettes de classe
$Y =$ étiquettes de "cluster"
$H() =$ Entropie

$$H(X) = -\sum_{x} P(x) \log P(x) \tag{2.8}$$

$I(X,Y) =$ l'information mutuelle

$$I(X,Y) = H(X) - H(H|Y) \tag{2.9}$$

Ce qui est égal à 1 si les partitions sont identiques, alors qu'il a une valeur attendue de 0 si les partitions sont indépendantes.

L'information mutuelle normalisée est actuellement très souvent utilisée dans des tests d'algorithmes de détection de communauté.

2.4 Algorithmes de détection des communautés

Il existe plusieurs algorithmes dans la détection de communauté nous avons choisi cinq algorithmes selon la performance et l'utilisation et la disponibilité des codes sources.

2.4.1 "Edge betweenness"

C'est le premier algorithme de l'ère moderne de la détection communautaire dans les graphes. Crée par Newman & Girvan [3].

Dans cette méthode, ils considèrent une autre propriété, semble être commun à de nombreux réseaux, la propriété de la structure de la communauté, figure 2.4.

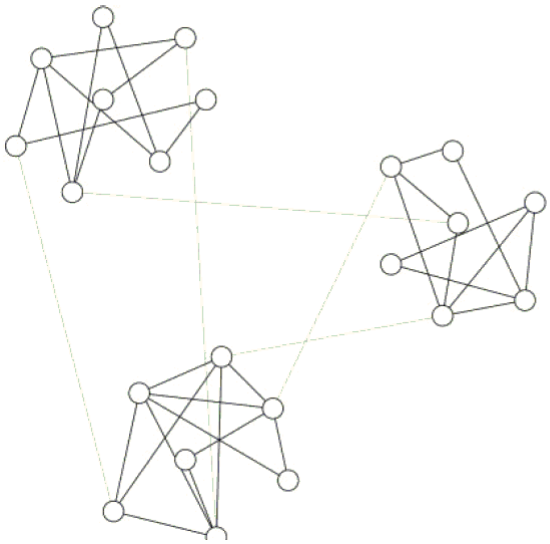


FIGURE 2.4 – Représentation schématique d'un réseau avec la structure de la communauté. Extrait de [3].

La figure 2.4 représente un réseau consiste trois communautés de nœuds densément connectés (cercles avec des lignes pleines), avec une densité des connexions beaucoup plus faible (lignes grises) entre eux.

Ils proposent une méthode qui détecte la structure de communauté, au lieu d'essayer de construire une mesure qui nous indique les arêtes les plus centrales des communautés, ils concentrent plutôt sur les arêtes qui sont les moins centrales, les arêtes qui sont le plus « betweenness » les communautés.

Plutôt que de construire des communautés en ajoutant les arêtes les plus fortes à un ensemble de nœuds initialement vide, ils construisent en supprimant progressivement les arêtes du graphe d'origine.

Freeman [31] propose une mesure de la centralité et de l'influence des nœuds dans les réseaux, qui s'appelle *nœud "betweenness"*, la centralité "betweenness" d'un nœud i est définie comme le nombre de chemins les plus courts entre des paires d'autres nœuds qui passent par i.

C'est une mesure de l'influence d'un nœud sur le flux d'informations entre les autres nœuds, en particulier dans les cas où le flux d'informations sur un réseau suit principalement le chemin le plus court disponible.

Pour trouver les arêtes d'un réseau qui se trouvent le plus entre d'autres paires de nœuds, nous généralisons la centralité "betweenness" de Freeman [31] pour les arêtes et définissons "betweenness" d'une arête comme le nombre de chemins les plus courts entre les paires de nœuds qui le longent.

Si un réseau contient des communautés ou des groupes qui ne sont que faiblement connectés par quelques arêtes intergroupes, alors tous les chemins les plus courts entre des communautés différentes doivent suivre l'une de ces quelques arêtes.

Ainsi, les arêtes reliant les communautés auront un arête "betweenness" plus élevée. En supprimant ces arêtes, nous séparons les groupes les uns des autres et révélons ainsi la structure de la communauté sous-jacente du graphe.

L'algorithme propose pour identifier les communautés est énoncé comme suit :

1. Calculez « *betweenness* » pour tous les arêtes du réseau.
2. Enlevez les arêtes avec « *betweenness* » le plus haut.
3. Recalculez « *betweenness* » pour tous les arêtes affectées par la suppression.
4. Répétez à partir de l'étape 2 jusqu'à ce qu'il ne reste plus d'arêtes.

L'algorithme s'exécute dans le temps $O(n^3)$ sur les graphes clairsemés, où n est le nombre de nœuds dans le réseau.

Avantages

— Cette méthode détecte les structures connues des communautés avec une sensibilité et une fiabilité élevées.
— Il détecte des divisions communautaires importantes et informatives dans les réseaux dont la structure de la communauté n'est pas bien connue.

inconvénients

— N'est pas pratique dans le cas où les très grands graphiques.

2.4.2 "Fast greedy modularity optimization"

L'algorithme d'analyse de communauté proposé par *Clauset, Newman et Moore (algorithme de CNM).* [32] trouve une structure de communauté.

Newman et Girvan 2.4.1 ont présenté un algorithme d'analyse de communauté avide qui optimise la modularité. Plus tard, *Clauset, Newman et Moore* ont proposé un algorithme plus efficace qui fonctionne en principe de la même manière que l'ancienne proposition, mais qui incorpore des structures de données sophistiquées.

L'algorithme CNM est une optimisation gloutonne ascendante qui trouve et fusionne continuellement une paire des communautés essayant de maximiser la modularité de la structure de la communauté.

L'algorithme commence à partir d'une situation totalement non-"cluster", où chaque nœud dans un graphique forme une communauté singleton. Puis calculé pour chaque paire de communautés, l'amélioration attendue de la modularité lorsqu'ils fusionnent :

$$\Delta Q^{C}_{c_i,c_j} = Q(G, C - c_i - c_j + (c_i \cup c_j)) - Q(G, C). \tag{2.10}$$

L'algorithme choisit à plusieurs reprises une paire des communautés qui donne la valeur ΔQ maximale et les fusionne dans une nouvelle communauté.

Pendant le processus de fusion, les valeurs ΔQ des communautés qui jouxtent la nouvelle communauté doivent être mises à jour, comme le nombre de paires de communautés dans le "cluster" diminue de façon monotone, l'algorithme s'arrête finalement lorsqu'il ne reste aucune paire de communautés à fusionner.

L'algorithme CNM utilise deux structures de données pour trouver une paire des communautés avec une valeur ΔQ maximale :

1. un arbre binaire équilibré de paires de communautés (c_i, c_j).
2. un tas maximum (ou tas de priorité) de paires des communautés triées par $\Delta Q^{c}_{c_i,c_j}$.

Pour chaque communauté, la paire de communautés avec la valeur ΔQ maximale est stockée dans un tas maximum à l'échelle du système.

En utilisant ces structures de données, la recherche de la paire de communautés avec la plus grande valeur ΔQ est effectuée en deux étapes :

Tout d'abord, chaque communauté recherche dans son tas maximal la paire avec le ΔQ le plus grand parmi ses paires de communautés et le stock dans un tas maximum à l'échelle du système qui est utilisé dans la deuxième étape.

Les éléments du tas max à l'échelle du système sont candidats à la paire de communautés qui a une valeur ΔQ maximale à l'échelle du système. Lorsque tous les candidats sont stockés dans le tas max du système, la paire avec la valeur maximale de l'ensemble du système ΔQ peut être facilement trouvée.

La complexité de cette algorithme est $O(nlog^2 n)$ sur les graphes clairsemés [6].

Avantages

- Cette méthode est essentiellement une implémentation rapide d'une technique antérieure proposée par Newman.
- Le temps de fonctionnement est linéaire.
- Nous permet d'étendre l'analyse de la structure de la communauté à des réseaux qui ont été jugés trop importants pour être traités.

inconvénients

- Elle construit un énorme dendegram déséquilibré, dégrade la performance.

2.4.3 Algorithme de Wakita & Tsurumi

L'algorithme de *Wakita & Tsurmi* [6], base sur l'algorithme précèdent (*Algorithme de CNM* 2.4.2), ils ont observé que la fusion de communautés de tailles déséquilibrées à un grand impact sur l'efficacité de calcul de *l'algorithme CNM*.

De cette observation, on s'attendait à ce que la fusion des communautés d'une manière équilibrée améliore l'efficacité de l'algorithme. Dans cet algorithme, ils introduisent la notion de ratio de consolidation, qui est une mesure de l'équilibre des paires de communautés, et l'utilisons ainsi que la modularité comme moyen de trouver la prochaine paire de communautés à fusionner en une plus grande.

Wakita & Tsurmi présentent une structure de donnée et trois types de ratio de consolidation. Trois modèles *d'algorithmes CNM*, chacun incorporant un de ces rapports de consolidation, ont été construits. Cela améliore considérablement l'efficacité de calcul de *l'algorithme CNM*.

Ils ont remplacé les arbres binaires équilibrés et les tas maximum, suggérés à l'origine dans 2.4.2 par une liste doublement chaînée triée dans l'ordre de l'identification de communauté.

Chaque communauté ci dans notre système a une structure de données pour stocker des références aux communautés voisines qui est représentée par une liste de paires de communautés. La liste est triée par ordre d'identification communautaire. Une paire de communautés a des références aux communautés auxquelles elle appartient, figure 2.5.

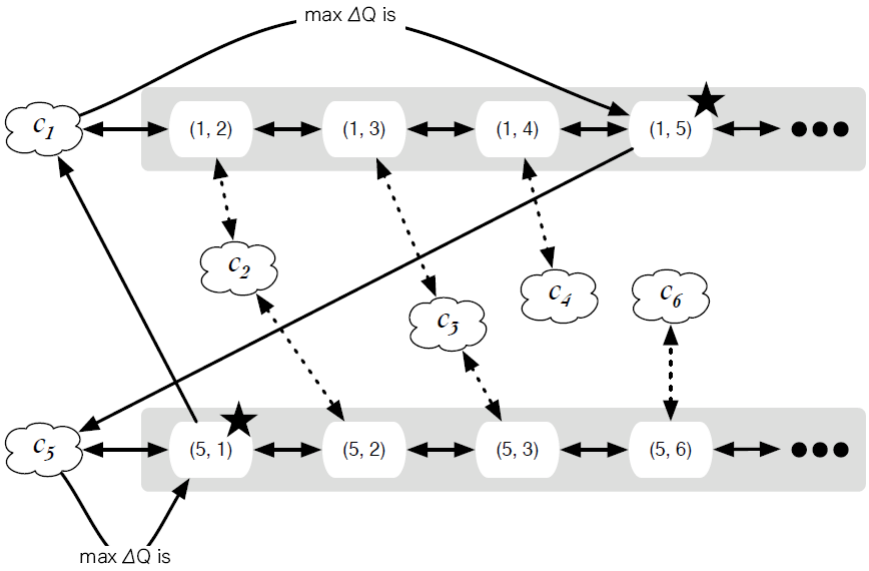


FIGURE 2.5 – Implémentation des communautés. Extrait de [6].

La figure 2.5 représente Une implémentation des communautés. La communauté $c1$ qui lie aux communautés $c_2, c_3, c_4, c_5 \cdots$ est représentée par un objet communautaire qui a une liste de paires de communautés $(1,2),(1,3),(1,4),(1,5)\cdots$

La paire de communautés (c_1, c_2) contient des liens pointant vers les communautés $c1$ et $c2$.

La figure représente la plus grande paire de communautés par des étoiles noires (*'s) et des liens vers les plus grandes paires de communautés par des « $\max \Delta Q$ is » liens.

Fusionner efficacement deux communautés est un processus de fusion de leurs paires de communautés, en éliminant les doublons et en mettant à jour leurs valeurs ΔQ.

Grâce à l'utilisation de listes triées, la fusion peut être effectuée dans un ordre linéaire par rapport au nombre de paires de communautés.

Comme dans 2.4.2, chaque communauté nomme sa plus grande paire de communautés (la paire dans sa liste de paires de communautés qui a la plus grande valeur ΔQ) à stocker dans le tas maximum à l'échelle du système.

Cette technique permet une récupération efficace de la paire de communautés maximales (la paire de communautés qui a la plus grande valeur ΔQ, à l'échelle du système).

À cette fin, chaque communauté maintient un lien vers la plus grande paire de communautés parmi les membres de sa liste. Lorsque deux communautés fusionnent, le lien « $\max \Delta Q$ » pour la nouvelle communauté peut simplement être trouvé car, de toute façon, nous devons analyser toutes les paires de communautés pour les fusionner, figure 2.6.

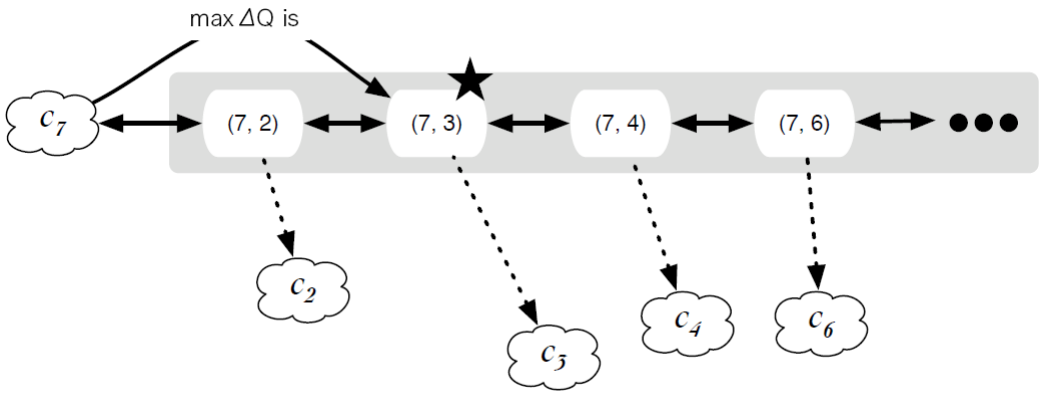


FIGURE 2.6 – Fusion de c_1 et c_5 dans la figure 3 a produit une nouvelle communauté c_7. Extrait de [6].

La figure 2.6 représente La fusion de c_1 et c_5 dans la figure 3 a produit une nouvelle communauté c_7. Pendant la fusion, les paires de communautés pour la fusion ont mis à jour leurs valeurs ΔQ.

L'idée de contrôler la croissance des communautés afin qu'elles se développent de manière équilibrée, vérifiée par le test de trois *algorithmes de CNM* qui intègrent des heuristiques basées sur trois types de ratio de consolidation.

La structure de l'algorithme reste la même que *algorithmes de CNM*. La seule différence réside dans la base d'évaluation des paires de communautés.

L'algorithme CNM utilise ΔQ alors qu'ils utilisent la combinaison de ΔQ et le ratio de consolidation $(ratio(c_i, c_j))$. Cette heuristique est conçue pour supprimer la fusion déséquilibrée des communautés et conduire à une croissance équilibrée des communautés.

La notion de ratio de consolidation de la fusion communautaire, définie comme suit :

$$ratio(c_i, c_j) = min(\frac{|c_i|}{|c_j|}, \frac{|c_j|}{|c_i|}) \tag{2.11}$$

Jusqu'à présent, il n'existe aucune définition de la façon de mesurer la taille de la communauté $(|c_i|)$. Ils ont défini trois différentes évaluations de la taille de la communauté et développé trois types d'heuristiques.

La première heuristique (HE) mesure la taille de la communauté en fonction de son degré.

La deuxième heuristique (HE') a été trouvée accidentellement quand ils essayaient de mettre en application HE. Comme ils l'avaient noté, le choix de la paire avec la plus grande valeur ΔQ est deux étapes.

La première étape (sélection d'une paire de communautés candidates), HE 'ignore la taille d'une communauté et se comporte donc comme l'algorithme CNM. D'autre part, pour la deuxième étape, où l'on cherche des paires de ΔQ maximum, on mesure la taille de la communauté en termes de degré, comme HE.

Cette heuristique bizarre, cependant, fonctionne plus vite que l'algorithme CNM et trouve également une meilleure classification en termes de modularité.

La dernière heuristique (HN) mesure la taille de la communauté en termes de nombre de ses membres.

2.4.4 "Fast modularity optimization"

Cet algorithme proposé par Blondel et al.[7] Qui extraire la structure de la communauté des grands réseaux. Base sur l'optimisation de la modularité.

Il trouve des partitions de modularité élevée de grands réseaux en peu de temps et qui déploie une structure de communauté hiérarchique complète pour le réseau, donnant ainsi accès à différentes résolutions de détection de communauté.

Cette algorithme est divisé en deux phases répétées de manière itérative. Supposons qu'ils commencent avec un réseau pondéré de n nœuds.

D'abord, ils assignent une communauté différente à chaque nœud du réseau. Ils parcourent chacun des nœuds du réseau, pour chaque nœud, ils considèrent le changement de modularité s'ils retirent le nœud de sa communauté actuelle et le plaçons dans la communauté de l'un de ses voisins.

Ils calculent le changement de modularité pour chacun des voisins du nœud. Si aucun de ces changements de modularité n'est positif, ils gardent le nœud dans sa communauté actuelle. Si certains des changements de modularité sont positifs, nous déplaçons le nœud dans la communauté pour laquelle le changement de modularité est le plus positif.

Ils répètent ce processus pour chaque nœud jusqu'à ce qu'un passage à travers tous les nœuds ne génère aucun changement d'affectation de communauté.

La deuxième phase de l'algorithme consiste à construire un nouveau réseau dont les nœuds sont maintenant les communautés trouvées lors de la première phase.

Pour ce faire, les poids des liens entre les nouveaux nœuds sont donnés par la somme du poids des liens entre nœuds dans les deux communautés correspondantes. Les liens entre les nœuds de la même communauté conduisent à des auto-boucles pour cette communauté dans le nouveau réseau.

Une fois cette deuxième phase terminée, il est alors possible de réappliquer la première phase de l'algorithme sur le réseau pondéré résultant et de l'itérer.

Le reste de l'algorithme consiste en une application répétée des étapes 1 et 2. Vous continuez ainsi jusqu'à ce qu'une application de l'étape 1 ne donne aucune réaffectation.

À ce point, l'application répétée des étapes 1 et 2 ne produira plus de changements d'optimisation de la modularité, de sorte que le processus est terminé. figure 2.8.

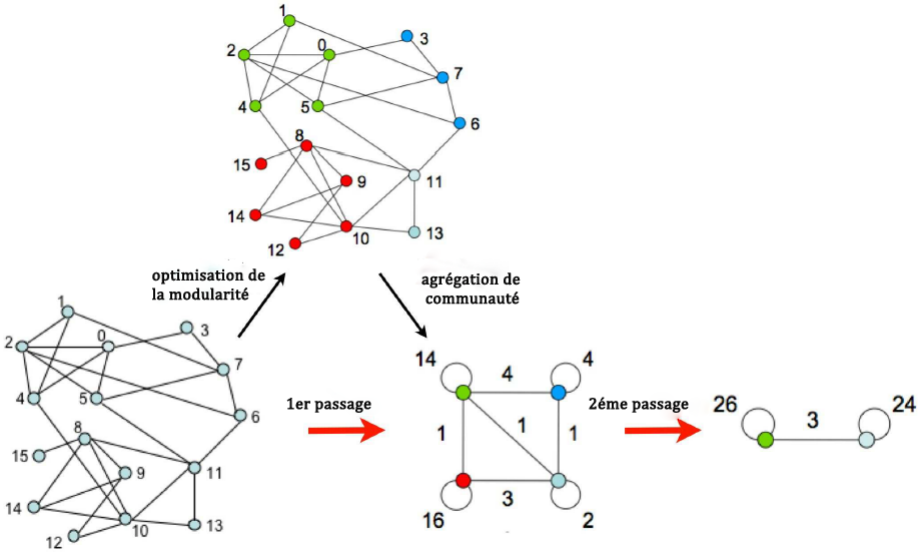


FIGURE 2.7 – Visualisation des étapes de Blondel & al algorithme. Extrait de [7].

Avantages

- Ses étapes sont intuitives et faciles à mettre en œuvre.
- L'algorithme est extrêmement rapide la complexité est linéaire sur des données typiques et éparses.
- la qualité des communautés détectées est très bonne, mesurée par la modularité.

inconvénients

- Les résultats préliminaires sur plusieurs cas de tests semblent indiquer que l'ordre des nœuds n'a pas d'influence significative sur la modularité obtenue. Cependant la commande peut influencer le temps de calcul.

2.4.5 Algorithme D'Infomap

Cet algorithme a été proposé par Rosvall et al [8]. Ici le problème de trouver la meilleure structure de "cluster" d'un graphe est transformé en problème de compression optimale de l'information d'un processus dynamique prenant place sur le graphe, à savoir une marche aléatoire.

Ils introduisent une approche théorique de l'information qui révèle la structure de la communauté dans les réseaux pondérés et dirigés.

Par cette approche ils peuvent mesurer avec quelle efficacité une carte représente la géographie sous-jacente, et nous pouvons mesurer la quantité de détails perdus dans le processus de simplification, ce qui nous permet de quantifier et de résoudre le compromis du cartographe.

Ils utilisent le flux de probabilité de marches aléatoires sur un réseau comme un proxy pour les flux d'information dans le système réel et décomposent le réseau en modules en compressant une description du flux de probabilité.

Le résultat est une carte qui à la fois simplifie et met en évidence les régularités dans la structure et leurs relations.

La figures 2.8 illustrent la détection des communautés en compressant la description des flux d'informations sur les réseaux.

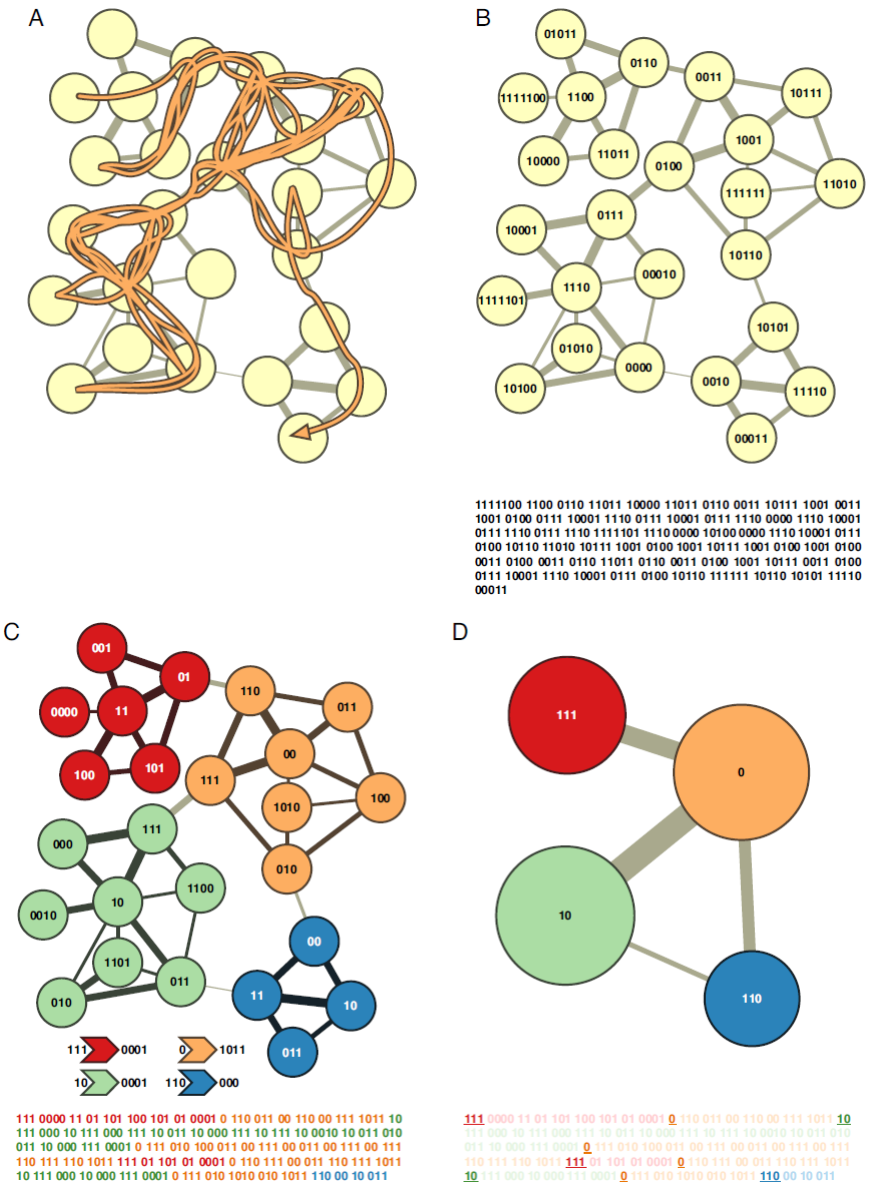


FIGURE 2.8 – Détecter les communautés en compressant la description des flux d'informations sur les réseaux. Extrait de [8].

La figure 2.8 représente la détection des communautés en compressant la description des flux d'informations sur les réseaux.

Dans la face (A) ils veulent décrire la trajectoire d'une marche aléatoire sur le réseau de sorte que les structures importantes aient des noms uniques. La ligne orange montre une trajectoire d'échantillon.

(B) Une approche de base consiste à donner un nom unique à chaque nœud du réseau. Le code Huffman illustré ici est un moyen efficace de le faire. Les 314 bits affichés sous le réseau décrivent la trajectoire de l'échantillon dans A, commençant par 1111100 pour le premier nœud de la promenade dans le coin supérieur gauche, 1100 pour le deuxième nœud,…etc. Et se terminant par 00011 pour le dernier nœud de la promenade dans le coin inférieur droit.

$C)$ Une description à deux niveaux de la marche aléatoire, dans laquelle les "clusters" majeurs reçoivent des noms uniques, mais les noms des nœuds dans les "clusters" sont réutilisés. Les codes identifiant les modules et les codes utilisés pour indiquer une sortie de chaque module sont respectivement affichés à gauche et à droite des flèches sous le réseau. En utilisant ce code, on peut décrire la marche en A par les 243 bits affichés sous le réseau en C. Les trois premiers bits 111 indiquent que la marche commence dans le module rouge, le code 0000 précise le premier

nœud sur la promenade,...etc.

(D) Le fait de ne rapporter que les noms des modules, et non les emplacements dans les modules, fournit un grainage grossier efficace du réseau.

Avantages

- *Infomap* et ses variantes renvoient généralement des partitions différentes des méthodes basées sur la structure (e.g. optimisation de la modularité).
- Les applications montrent que de cette manière, il est plus facile de récupérer les communautés qui se chevauchent, en particulier des chevauchements omniprésents, qui sont généralement hors de portée pour la plupart des algorithmes de "clustering".

inconvénients

- Si les communautés ont une densité des arêtes interne élevée et sont bien séparées les unes des autres, les marcheurs aléatoires seraient piégés dans chaque groupe pendant un certain temps, avant de trouver une issue et de migrer vers un autre groupe.
- Il ne peut pas être utilisé dans les réseaux non pondérés et non orientés.

2.5 Conclusion

Dans ce chapitre nous avons expliqué le domaine de détection des communautés ; les domaines d'application et les différents algorithmes utilisés avec des mesures évaluées la qualité de la partition.

Dans le chapitre suivant, nous allons utiliser et détailler la technique d'analyse factorielle 2^k et comment appliquer sur le domaine de détection des communautés pour trouver les facteurs les plus influents dans ce domaine.

3 Déterminisation du facteur le plus influent

3.1 Introduction

La détection des communautés est un sujet important, car il peut rencontré dans plusieurs domaines d'applications et des situations dans le monde réel.

La détection des communautés nous permet également de déterminer le rôle des différents acteurs au seins des communautés et dans le réseau dans sa globalité.

Par ailleurs, beaucoup de travaux portant sur la définition et à la détection de ces communautés qui ont été effectuées durant ces dernières années.

Dans ce chapitre nous allons proposé une méthode qui détecte le facteur le plus influent dans la détection de communauté ; afin de se concentrer sur ce facteur dans les travaux futurs pour optimiser les résultats finales et les performances.

Les facteurs représentent les paramètres utilisés dans les algorithmes de la détection des communautés, exemple : Le nombres des nœuds, des arêtes, le degré de nœud ... etc.

3.2 Application de l'analyse factorielle 2^k

Généralement, le processus de conception du "cluster" où la détection des communautés est basée sur certains facteurs, qui représentent les paramètres de réseau, exemples :

Les nombres des *nœuds*, des *arêtes*, *le degré de nœud* k_i, *le paramètre de mélange* μ qui donne le rapport entre le degré externe et le degré de nœud i ... etc.

Pour cela, nous avons besoin de déterminer ces facteurs et leurs impacts sur la performance de détection communautaire, ainsi de déterminer les plus influents pour concentrer les tests de recherches sur ces facteurs, et obtenir d'excellents résultats pour les processus des détections de la communauté et réduire le temps de mise en œuvre.

Notre application c'est de faire démontrer ces processus par la méthode d'analyse factorielle 2^k que nous avons expliqué dans le chapitre 1 section 1.8.

L'approche de base de cette méthode est basée sur la sélection d'un ensemble de k paramètres et la détermination de deux niveaux extrêmes (marqués avec -1 et 1).

Une expérience est exécutée pour toutes les $\binom{k}{2}$ combinaisons possibles des paramètres, dans chaque expérience nous pouvons également extraire les interactions $\binom{k}{2}$ à deux facteurs, les $\binom{k}{3}$ interactions à trois facteurs ... etc.

Ensuite, on construit la matrice de signes qui représente les expériences exécutées pour toutes les combinaisons.

On explique et on détaille cette approche dans l'exemple suivant :

3.2.1 Exemple applicatif

Supposons le problème de l'étude de l'impact du nombre des nœuds et les nombres des arêtes sur les performances de détection de communautés.

On doit choisir deux niveaux pour chacun de ces deux facteurs.

Dans cet exemple les performances de détection de communauté sont mesurées par seconde (s), qui sont répertoriées dans le tableau ci-dessous **??**.

TABLE 3.1 – Performance en seconde (s).

Nombre de nœuds (Nœuds)	Nombre d'arêtes 9k (arêtes)	Nombre d'arêtes 1M (arêtes)
24k	3.6	400
325k	45.3	5034

Définissons les deux variables x_A et x_B comme suit :

$$x_A = \begin{cases} \text{-1} & \text{si le nombre des nœuds =24k} \\ 1 & \text{si le nombre des nœuds =325k} \end{cases} \tag{3.1}$$

$$x_B = \begin{cases} \text{-1} & \text{si le nombre d'arêtes =9k} \\ 1 & \text{si le nombre d'arêtes =1M} \end{cases} \tag{3.2}$$

La performance y mesurée en seconde peut maintenant être régressée sur x_A et x_B en utilisant un modèle de régression non linéaire de la forme :

$$y = q_0 + q_A x_A + q_B x_B + q_{AB} x_A x_B \tag{3.3}$$

Tel que le q_A et q_B c'est le résultats de colonne A et B multiplie dans la colonne Y.

En substituant les quatre observations du modèle, nous obtenons les quatre équations suivantes :

$$3.6 = q_0 - q_A - q_B + q_{AB}$$
$$400 = q0 + q_A - q_B - q_{AB}$$
$$45.3 = q_0 - q_A + q_B + q_{AB}$$
$$5034 = q_0 + q_A + q_B + q_{AB}$$

Ces équations peuvent être résolues uniquement pour les quatre inconnues.

L'équation de régression est comme suit :

$$y = 1370.7 + 1346.3x_A + 4675.7x_B + 1170.7x_A x_B \tag{3.4}$$

Le résultat est interprété comme suit, la performance moyenne : $1370.7s$, l'effet du nombre de nœuds : $1346.3s$, l'effet du nombre d'arêtes est $4675.7s$, l'interaction entre le nombre de nœud et nombre d'arêtes : $1170.7s$.

TABLE 3.2 – Analyse pour la conception 2^2.

Expérience	A	B	y
1	- 1	- 1	y1
2	1	- 1	y2
3	- 1	1	y3
4	1	1	y4

En général n'importe quelle conception peut être analysée en utilisant la méthode de l'exemple 3.1.

Généralement, on suppose que ces y_1, y_2, y_3 et y_4 représentent les quatre réponses observées.

La correspondance entre les niveaux des facteurs et les réponses sont présentées au tableau 3.2.

Ce modèle de cette conception 2^2 est comme suit :

$$y = q_0 + q_A x_A + q_B x_B + q_{AB} x_A x_B$$

En substituant les quatre observations dans le modèle, nous obtenons :

$$y_1 = q_0 - q_A - q_B + q_{AB}$$
$$y_2 = q0 + q_A - q_B - q_{AB}$$
$$y_3 = q_0 - q_A + q_B + q_{AB}$$
$$y_4 = q_0 + q_A + q_B + q_{AB}$$

Pour résoudre ces équations de q_i, nous obtenons :

$$q_0 = \frac{1}{4}(y_1 + y_2 + y_3 + y_4)$$
$$q_A = \frac{1}{4}(-y_1 + y_2 - y_3 + y_4)$$
$$q_B = \frac{1}{4}(-y_1 - y_2 + y_3 + y_4)$$
$$q_{AB} = \frac{1}{4}(y_1 - y_2 - y_3 + y_4)$$

Notez que les expressions pour q_A, q_B et q_{AB} sont des combinaisons linéaires des réponses, telles que la somme des coefficients est nulle, de telles expressions sont appelées contrastes.

Notez également que les coefficients de y_i dans l'équation pour q_A sont identiques au niveaux de A énumérées dans le tableau **??**.

Ainsi, q_A peut être obtenue en multipliant les colonnes A et y dans la table.

Également, vrai pour q_B et q_{AB}, qui peuvent tous les deux être obtenues en multipliant les colonnes de niveaux A, B respectives avec la colonne de réponse y.

Ainsi, l'observation nous amène à la méthode de la table des signes pour calculer les effets, qui sont décrites ci-après.

TABLE 3.3 – Méthode de la table de signe pour calculer les effets des facteurs dans une conception 2^2.

I	A	B	AB	y
1	- 1	- 1	1	3.6
1	1	- 1	- 1	400
1	- 1	1	- 1	45.3
1	1	1	1	5034
5482.9	5385.1	4675.7	4592.3	Total
1370.7	1346.7	1168.9	1148.1	Total/4

Dans une analyse factorielle 2^k, en utilisant la méthode de la table des signes, nous pouvons obtenir les résultats et détecter les variations qui dépendent de la combinaison des facteurs.

Pour le modèle 2^2, les effets peuvent être calculés facilement en doit établir une matrice de signes 4×4 comme indiqué dans le tableau 3.3.

La première colonne de la matrice est étiquetée I, et tous ces éléments sont égaux à 1, les deux colonnes suivantes intitulées A et B, contiennent essentiellement toutes les combinaisons possibles de -1 et 1.

La quatrième colonne, étiquetée AB est le produit des entrées dans les colonnes A et B, les quatre observations sont répertoriées dans un vecteur colonne à coté de cette matrice.

Le vecteur colonne est étiqueté y et se compose des résultats correspondants aux niveaux des facteurs énumérés dans les colonnes A et B.

L'étape suivante consiste de multiplier les entrées de la colonne I par celles de la colonne y et de mettre leur somme sous la colonne I, les entrées de la colonne A sont maintenant multipliées par celles de la colonne y et la somme est inscrite sous la colonne A.

Cette opération de multiplication de colonnes est répétée pour les deux autres colonnes de la matrice, les sommes sous chaque colonne sont divisées par 4 pour donner les coefficients correspondants du modèle de régression.

L'importance d'un facteur dépend de la proportion de la variation totale métrique expliquée par le facteur.

La variation totale de y est également connue sous le nom de sommes des carrées totaux (SCT), qui peut être calculée comme suit :

$$\text{Variation totale de y} = SCT = \sum_{i=1}^{2^2} (y_i - \bar{y})^2. \tag{3.5}$$

$\bar{y}$ représente la moyenne des réponses des quatre expériences.

Pour une conception 2^2, la variation peut être divisée en trois parties :

$$SCT = 2^2 q_A^2 + 2^2 q_B^2 + 2^2 q_{AB}^2. \tag{3.6}$$

Ces parties peuvent être exprimées en fraction ; exemple :

$$\text{Fraction de variation expliquée par A} = \frac{SCA}{SCT} = \frac{2^2 q_A^2}{SCT}. \tag{3.7}$$

Tel que le SCA c'est la variation de facteur A.

Par conséquent, nous pouvons indiquer le pourcentage de variation de chaque métrique étudiée et expliquée par chaque facteur, plus le pourcentage de variation est élevé, plus ce facteur à d'impact sur la métrique de mesure.

Dans notre exemple, nous avons constaté que le nombre des nœuds représente $40,30\%(i.e.\ 2^2 \times 1346, 7^2/17987672, 9475)$ de la variation totale de temps d'exécution, le nombre d'arêtes représente $30.38\%(i.e.\ 2^2 \times 1168, 9^2/17987672, 9475)$, et leurs combinaisons représentent les $29.31\%(i.e.\ 2^2 \times 1148, 1^2/17987672, 9475)$ restants.

Par conséquent, dans notre exemple choisi ; le nombre des nœuds c'est le facteur le plus important qui affecte sur le temps d'exécution.

Le résultat d'analyse factorielle 2^k nous permet de trier les facteurs dans l'ordre d'impact. Au début de toute étude de performance, le nombre des facteurs et leurs niveaux peuvent généralement être importants.

Une conception factorielle complète avec un si grand nombre des facteurs et des niveaux peuvent ne pas être la meilleure utilisation de l'effort disponible. La première étape devrait consister à réduire le nombre des facteurs et de choisir les facteurs qui ont un impact significatif sur la performance.

3.2.2 Algorithmes implémentés

L'algorithme que nous allons utiliser pour calculer l'influence de chaque facteur basé sur la méthode d'analyse factorielle 2^k elle est décomposée en deux parties :

La première partie $(Algorithme1)$ construit la matrice des signes pour tous les facteurs et ces combinaisons, cette matrice peut données un nombre illimité des signes, mais l'architecture de l'ordinateur elle réserve juste une 2^{63} adresse mémoire.

Alors on remarque que le nombre dans le cas pratique existant bloqué à 63 facteurs.

La deuxième partie $(Algorithme2)$ faire les calculs pour trouvé l'influence de chaque facteur et les combinaisons.

Algorithm 1 Création la matrice des signes

```
Rec = MyRec;                                              ▷ Créer un nouveau type
MyRec = record;
val : integer;
suivant : Rec;                                            ▷ Fin de nouveau type
k, j, dimen, Prod : integer;
p0, p1, p2 : Rec;
Read k
Str ← Read String;
j ← 0;
t : tableau[dimen][dimen − 1]of integer;
List : tableau[k + 1]of Rec;
list[0] ← p0;
list[0].val ← 1;
list[0].in ← −1;
list[0].suivant ← null;
for p = 0 To p < k do
    p1 ← list[p];
    p0 = newRec();
    list[p + 1] ← p0;
    p2 ← p0;
    while p1 == null do
        for i = p1.ind + 1 To i < k do
            if i! = k then
                p2.suivant = p0;
                p2 ← p0;
                p2.ind ← i;
                p2.suivant ← null;
            j ← j + 1;
        p1 ← p1.suivant;
for d = 0 To d < dimen do
    j ← 0;
    for p = 0 To p < k do
        p1 ← list[p];
        p2 ← List[p + 1];
        while p1! = null do
            for i = p1.ind + 1 To i < k do
                Prod = (d >> i)&1;
                Prod ← −1^Prod;
                t[d][j] ← p1.val × Prod;
                if i! = k then
                    p2.val ← t[d][j];
                    p2 ← p2.suivant;
                j ← j + 1;
            p1 ← p1.suivant;
                                                          ▷ Fin la matrice de signes
```

Algorithm 2 Calcul Analyse factorielle 2^k

```
1:  y : table[2][dimen + 1] of integer;
2:  v : table[dimen] of double;
3:  textfile : file;
        Read textfile and put the values in table v[]
4:
5:  procedure CALCUL(k)
6:      y : table[4][dimen − 1] of double
7:      Calcul, Somme, Value, SST : double
8:      Calcul ← 0
9:      for m = 0 To m < dimen − 1 do
10:         Somme ← 0
11:         for i = 0 to i < dimen do
12:             Calcul ← t[i][m] × v[i]
13:             Somme ← Somme + Calcul
14:         y[0][m] ← Somme
15:     for r = 0 to r < dimen − 1 do
16:         y[1][r] ← y[0][r]/dimen
17:         Somme ← 0
18:         for i = 0 to i < dimen − 1 do
19:             Value ← y[1][i]
20:             SST ← Value^2
21:             y[2][i] ← dimen × SST
22:             Some ← Somme + y[2][i]
23:         for i = 0 to i < dimen − 1 do
24:             y[3][i] ← (y[2][i] × 100)/Somme
```

3.3 Résultat et Discussion

Le cas de notre étude consiste les éléments suivant :

3.3.1 Paramètre d'étude

Dans notre étude nous allons choisir trois paramètre pour réaliser le travail, On a trouvé des difficultés sur la disponibilité des paramètres, car chaque paramètre doit combiné avec les autres niveaux des paramètres.

Nombre de nœuds (n)

les nœuds qui sont dénotés par n représentent l'unité fondamentale sur laquelle les graphiques sont formées ; les nœuds sont traitées comme des objets indissociables et indivisibles, bien qu'ils puissent avoir une structure supplémentaire en fonction de l'application à partir de laquelle le graphe apparaît ; exemple : un réseau sémantique est un graphe dans lequel les nœuds représentent des concepts ou des classes d'objets.

Nous avons sélectionné deux niveaux pour les nombres de nœuds, un niveau minimal -1 qui prend la valeur 1000, et un niveau maximal 1 prend la valeur 5000.

Paramètre de mélange μ

Le paramètre de mélange d'un nœud i qui dénote par μ_i, c'est le rapport entre le degré externe et le degré de nœud i :

$$\mu_i = k_i^{ext}/k_i. \tag{3.8}$$

Pour cela nous avons choisi comme une valeur minimale 0 et une valeur maximale $0,8$.

Taille de communauté

La taille de communauté C représente le nombre de nœud pour chaque communauté.

Nous allons prendre deux niveaux, chaque niveau représente une intervalle ; pour le niveau minimal S l'intervalle des valeurs est $[10; 50]$, et pour le niveau maximal B l'intervalle est $[20; 100]$.

Table 3.4 – Les valeurs des facteurs considérés.

Indicateur	Facteur	Niveau (-1)	Niveau (1)
F1	Nombre de nœuds	1000	5000
F2	paramètre de mélange	0	0.8
F3	Taille de communauté	S	B

3.3.2 Données en entrée

Pour les données en entrée des paramètres utilisées, on a trouvé les résultats que nous avons besoin dans certains cas en fonction de l'information mutuelle normalisée 3.5.

Ces données appliquées sur des différentes algorithmes ; nous avons choisi trois algorithmes, Blondel & al 2.4.4, Infomap 2.4.5, CNM2.4.2.

Ces algorithmes sont les plus utilisés avec une excellente performance et avantage supplémentaire d'une bonne complexité de calcul, ce qui permet d'analyser des grands systèmes.[33] [34]

Le tableau suivant représente les observations des valeurs et les résultats.

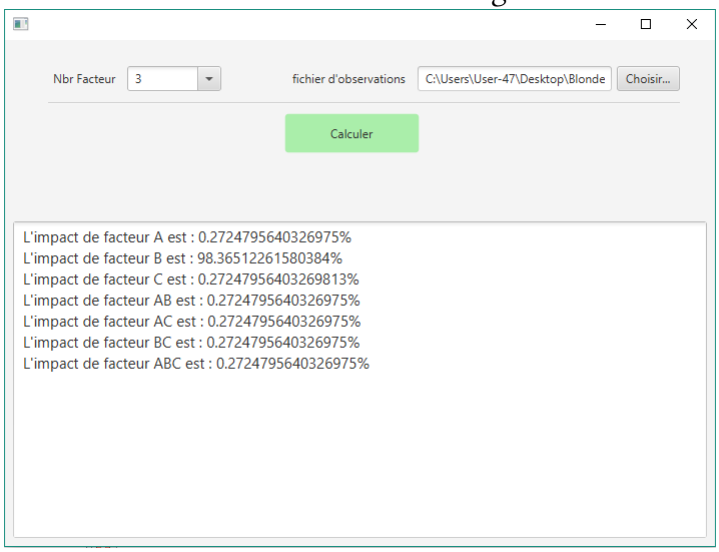
TABLE 3.5 – Observation pour les algorithmes Blondel & al, Infomap, CNM. en fonction de l'information mutuelle normalisée.

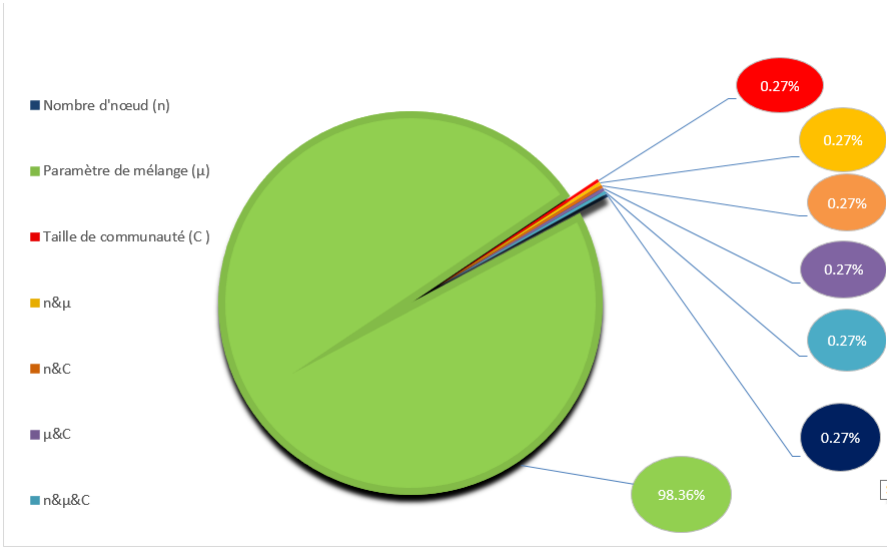
Données			Résultats		
			Blondel	infomap	CNM
1000	0	S	1	1	0,85
1000	0	B	1	1	0,9
1000	0,8	S	0	0	0
1000	0,8	B	0	0	0
5000	0	S	1	1	0,5
5000	0	B	1	1	0,7
5000	0,8	S	0.2	0,32	0
5000	0,8	B	0	0	0

3.3.3 Exécution

Dans l'algorithme de Blondel & al

FIGURE 3.1 – Influence des facteurs dans l'algorithme de Blondel & al.

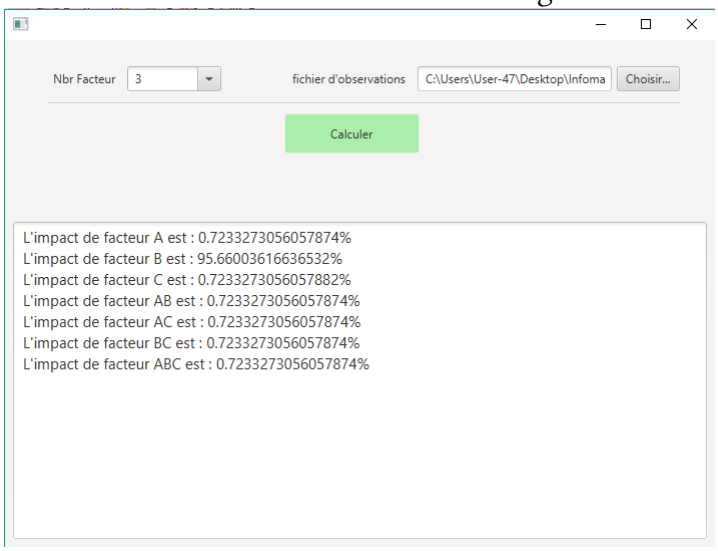


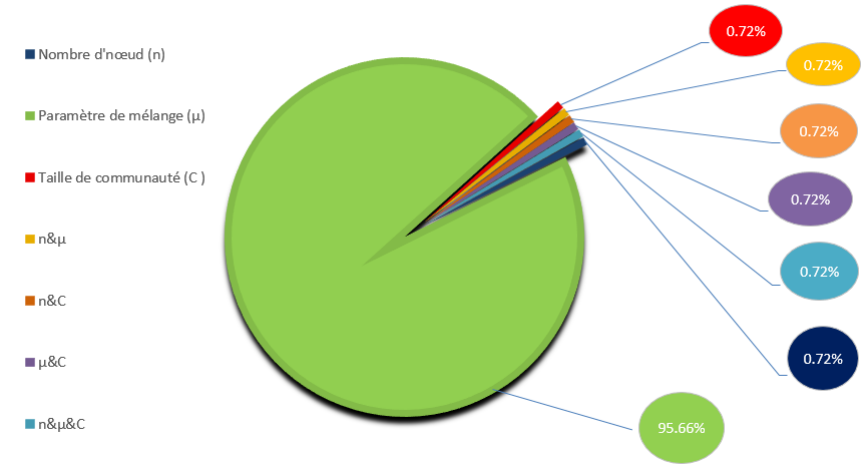


La figure 3.1 représente l'influence des facteurs de l'étude sur l'algorithme Blondel& al. Elle démontre que le paramètre de mélange (μ) 3.3.1 est le facteur le plus influent avec un pourcentage de 98.36%, suivi par les autres facteurs qui prennent le même pourcentage 0.27%.

Dans l'algorithme d'Infomap

FIGURE 3.2 – Influence des facteurs dans l'algorithme d'Infomap.

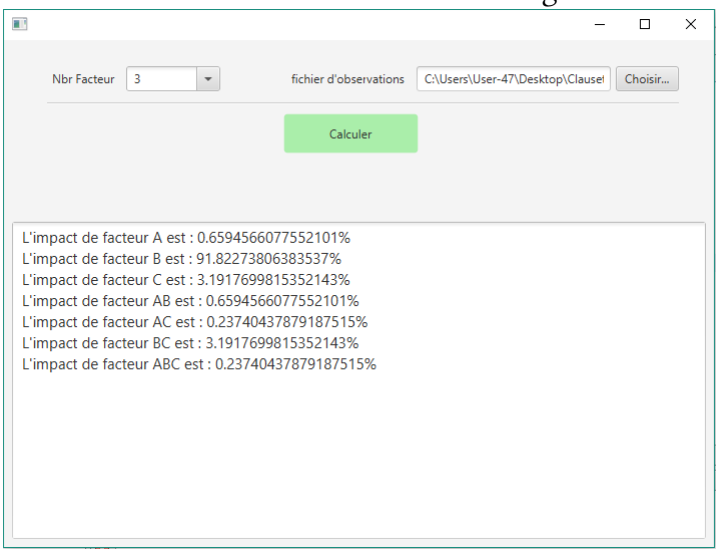


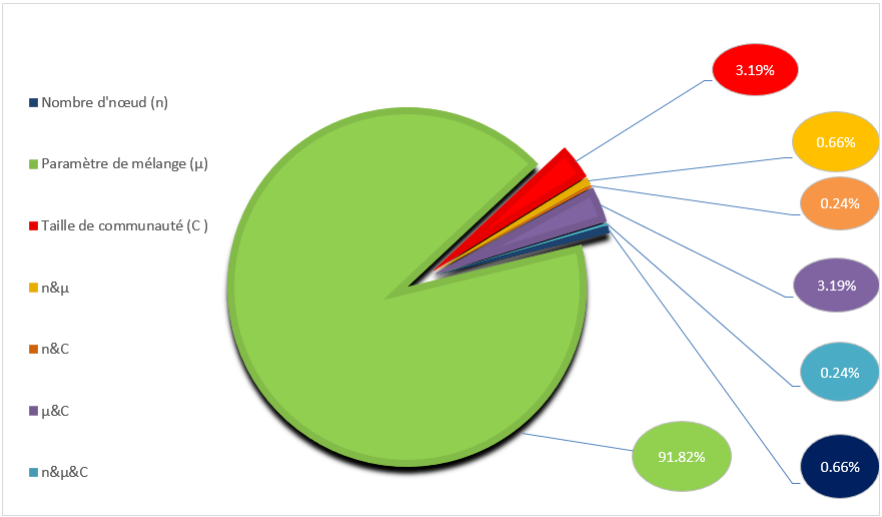


La figure 3.2 représente l'influence des facteurs de l'étude sur l'algorithme de CNM. Elle démontre que le paramètre de mélange μ 3.3.1 est le facteur le plus influent avec un pourcentage de 95.66%, suivi par les autres facteurs qui prennent le même pourcentage 0.72%.

Dans l'algorithme de CNM

FIGURE 3.3 – Influence des facteurs dans l'algorithme de CNM.





La figure 3.3 représente l'influence des facteurs de l'étude sur l'algorithme d'Infomap. Elle démontre que le paramètre de mélange μ 3.3.1 est le facteur le plus influent avec un pourcentage de 91.82%, suivi par les facteurs de taille de communauté et l'interaction entre le paramètre de mélange et la taille de communauté avec un pourcentage 3.19%; par la suite les facteurs de nombre des nœuds et l'interaction entre ce dernier et le μ avec un pourcentage de 0.66%, en fin les deux derniers facteurs l'interaction entre n, c et entre les trois facteurs principaux (n, μ, c) avec une pourcentage de 0.24%.

3.3.4 Discussion

Après l'étude que nous avons effectue sur les trois algorithmes (Blondel & al, Infomap, CNM) et les trois facteurs (nombre de nœuds n, paramètre de mélange μ, la taille de communauté C), avec sa combinaison, à l'aide de la méthode d'analyse factorielle 2^k.

Nous avons remarqué le résultat dans les trois algorithmes reportez-vous aux même facteur qui est le plus influent c'est le facteur de paramètre de mélange (μ) avec un pourcentage moyen de 95.28%, puis le facteur c (la taille de communauté) avec un pourcentage moyen de 1.39%, et en fin le facteur n (le nombre de nœuds) avec un pourcentage moyen de 0.55%.

Alors on peut dire, pour améliorer les travaux futur on recommande l'utilisation de la fonction objective suivante :

$$F(n, \mu, c) = 0.55n + 95.28\mu + 1.39c. \tag{3.9}$$

L'amélioration située sur les algorithmes de détection de communautés par la concentration sur certain(s) facteur(s) (μ) et réduit le temps de mis en ouvre, et trouvé des excellentes partition / performance .

3.4 Conclusion

Dans ce chapitre nous avons expliqué comment faire une analyse factorielle 2^k avec tous ses étapes, et appliqué sur notre application qui est basée sur cette méthode pour trouver les résultats finales qui représente le facteur le plus influent dans la détection des communautés et pour améliorer la qualité de partions et réduit le temps de mis en œuvre.

En guise de conclusion nous allons faire une conclusion générale de ce travail.

Conclusion générale

La détection de communautés est un domaine qui est encore dans une phase d'exploration, ce domaine est plus important dans la recherche et les applications actuelles.

Dans le cadre de ce mémoire, nous avons fait une étude sur les algorithmes de détection de communautés on a utilisé une méthode d'analyse factorielle qui s'appelle L'analyse factorielle 2^k, pour trouver le facteur le plus influent dans ces algorithmes.

La méthode que nous avons utilisée exige pour chaque facteur deux niveaux extrêmes (-1 et 1) et une expérience pour toutes les combinaison de facteur avec les autres.

Pour cela on a choisi trois facteurs (nombre des nœuds, paramètre de mélange, la taille de communauté) pour étudier le facteur le plus influent dans ces algorithmes.

Les résultats qu'on à trouvées ; le facteur qui représente le paramètre de mélange (μ) est le plus influent parmi les autres.

Donc, pour les travaux future on a proposé pour concevoir ou améliorer un algorithme de détection des communautés il faut concentré sur ce facteur, pour trouve des excellentes partions, performances, des communautés précises et réduit le temps de mis en œuvre.

Notre application peut être appliquée dans des autres domaines pour l'amélioration des résultats.

Bibliographie

[1] J. Newsom, "Exploratory and confirmatory factor analysis," *Spring*, vol. 36, 2017.

[2] S. Ray. (2015) Regression. [Online]. Available : https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression

[3] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[4] S. Fortunato and D. Hric, "Community detection in networks : A user guide," *Physics Reports*, vol. 659, pp. 1–44, 2016.

[5] E. J. North S, Gansner E. (1998) Graphviz. [Online]. Available : http://www.graphviz.org

[6] K. Wakita and T. Tsurumi, "Finding community structure in mega-scale social networks," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 1275–1276.

[7] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics : theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.

[8] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[9] M. Newman, *Networks : an introduction*. Oxford university press, 2010.

[10] R. K. Jain, *Art of Computer Systems Performance Analysis : Techniques for Experimental Design Measurements... Simulation and Modeling*. John Wiley, 2015.

[11] G. Saporta, *Probabilités, analyse des données et statistique*. Editions Technip, 2006.

[12] D. Child, *The essentials of factor analysis*. Cassell Educational, 1990.

[13] R. P. McDonald, *Factor analysis and related methods*. Psychology Press, 2014.

[14] S. Benzecri, *Leçons sur l'analyse factorielle et la reconnaissance des formes : annexes [au cours de 3e cycle, option : statistique mathématique]*. Institut de statistique de l'Université de Paris.

[15] (2010) Analyse en composantes principales. [Online]. Available : http://www.statelem.com/analyse_factorielle_des_correspondances.php

[16] H. Abdi, D. Valentin *et al.*, "Multiple factor analysis (mfa)," *Encyclopedia of Measurement and Statistics. Sage, Thousand Oaks (CA)*, pp. 657–663, 2007.

[17] S. Le Dien and J. Pagès, "Analyse factorielle multiple hiérarchique," *Revue de statistique appliquée*, vol. 51, no. 2, pp. 47–73, 2003.

[18] A. Martin, "L'analyse de données," 2004.

[19] S. M. Stigler *et al.*, "Karl pearson's theoretical errors and the advances they inspired," *Statistical Science*, vol. 23, no. 2, pp. 261–271, 2008.

[20] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.

[21] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 9, pp. 2658–2663, 2004.

[22] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[23] C. R. Lin and M. Gerla, "Adaptive clustering for mobile wireless networks," *IEEE Journal on Selected areas in Communications*, vol. 15, no. 7, pp. 1265–1275, 1997.

[24] S. E. Schaeffer, M. Sarela, S. Marinoni, and P. Nikander, "Dynamic local clustering for hierarchical ad hoc networks," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON'06. 2006 3rd Annual IEEE Communications Society on*, vol. 2. IEEE, 2006, pp. 667–672.

[25] U. Gargi, W. Lu, V. S. Mirrokni, and S. Yoon, "Large-scale community detection on youtube for topic discovery and exploration." in *ICWSM*, 2011.

[26] P. S. Bradley, U. M. Fayyad, C. Reina *et al.*, "Scaling clustering algorithms to large databases." in *KDD*, 1998, pp. 9–15.

[27] F. Boyer, A. Morgat, L. Labarre, J. Pothier, and A. Viari, "Syntons, metabolons and interactons : an exact graph-theoretical approach for exploring neighbourhood between genomic and functional data," *Bioinformatics*, vol. 21, no. 23, pp. 4209–4215, 2005.

[28] R. Guimera, S. Mossa, A. Turtschi, and L. N. Amaral, "The worldwide air transportation network : Anomalous centrality, community structure, and cities' global roles," *Proceedings of the National Academy of Sciences*, vol. 102, no. 22, pp. 7794–7799, 2005.

[29] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.

[30] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics : Theory and Experiment*, vol. 2005, no. 09, p. P09008, 2005.

[31] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.

[32] A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Physical review E*, vol. 70, no. 6, p. 066111, 2004.

[33] A. Lancichinetti and S. Fortunato, "Community detection algorithms : a comparative analysis," *Physical review E*, vol. 80, no. 5, p. 056117, 2009.

[34] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

A Annexe

A.1 Manuel d'utilisation

Notre application qui implémente la méthode d'analyse factorielle 2^k est programmée en langage Java. L'exécution se fait dans un ordinateur qui possède : CPU = Intel(R) 2.60GHZ, Cache = 3MB, RAM = 6GB et un système d'exploitation de Windows10.

Pour l'utilisation de cette application il faut suivre les étapes suivantes :

1. Déterminer le nombre de facteurs.
2. Choisir le fichier des données pour ces facteurs.
3. Cliquer sur le bouton "calculer" pour démarrer les calculs.
4. Visualiser l'écran des résultats.

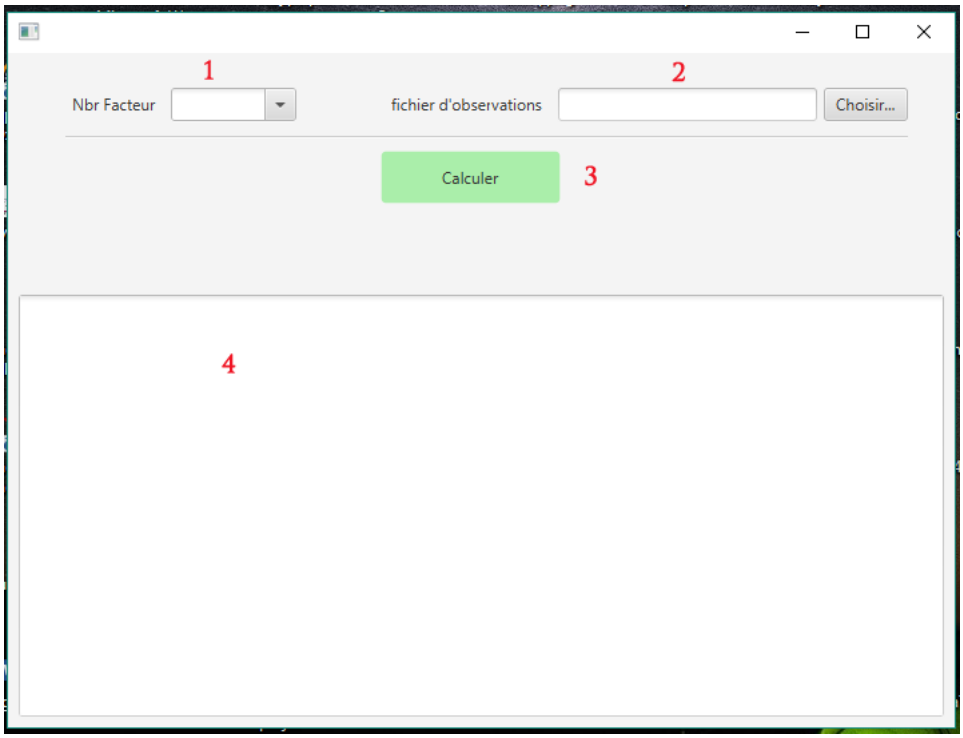


FIGURE A.1 – Interface de notre application

Pour le fichier des données qui contient les résultats des expériences pour les facteurs introduits avec les différentes combinaisons. Les données doivent être triées en ordre décroissant dans une matrice, et stockés chaque valeur d'expérience dans une ligne.

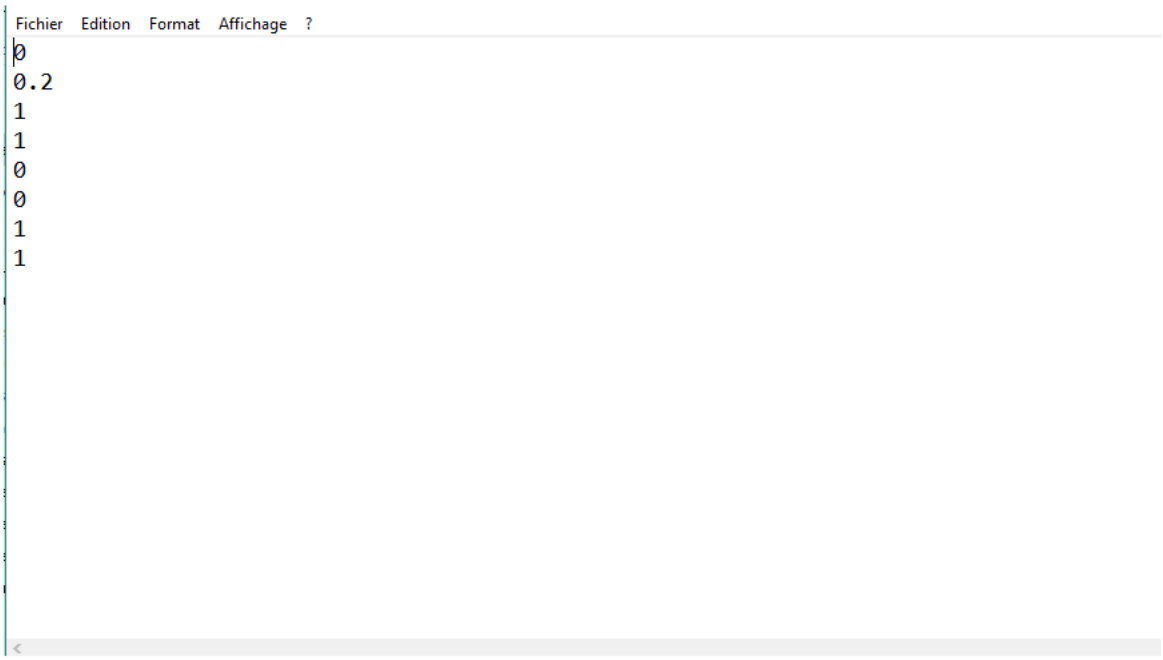


FIGURE A.2 – Fichier des données