Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Li'bia Réso

DEHAN, Laetitia

Award date:
2014

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 02. May. 2026



UNIVERSITE DE NAMUR

Faculté des Sciences

Li'bia Réso

Promoteur : Timotéo Carletti

Mémoire présenté pour l'obtention

du grade académique de master en Sciences mathématiques à finalité spécialisée

Laetitia DEHAN

Juin 2014

Résumé

Dans ce mémoire, nous analysons le comportement de la population de la ville de Namur utilisant le réseau "Li'bia Vélo" mis en place par la société JCDecaux dans la ville de Namur. Dans un premier temps, nous effectuons différentes analyses du réseau au fil du temps. Nous pouvons remarquer que toutes les stations ne sont pas utilisées de la même manière en fonction des jours et des heures qui sont considérées. Dans un second temps, nous appliquons plusieurs analyses statistiques sur nos données afin de déterminer les principaux facteurs qui influencent le comportement des utilisateurs. Nous constatons que le type d'abonnement que les utilisateurs possèdent ou tout simplement les saisons et les vacances scolaires ont un impact sur le comportement des utilisateurs du réseau. Différentes méthodes de classification nous permettrons finalement de regrouper ensemble des stations dont le comportement est similaire en fonction de l'utilisation journalière et/ou par tranche horaire.

Mots-clés *: Li'Bia vélos, réseaux dynamiques, empreintes digitales, villes intelligentes, classification, ...*

Abstract

In this master thesis, we analyze the behavior of the population of Namur city using the "Li'Bia Vélo" network of Namur set up by JCDecaux company. At first, we make different analysis of the network over the time. We can notice that all stations are not used in the same way according to days and hours which are considered. Secondly, we apply several statistical analysis on our data to see if some factors influence the behavior of the users. We notice that the type of subscriptions that the users have or simply the seasons and the school holidays have an impact on the behavior of the users of the network. Various methods of classifications will allow finally to group stations wich the behaviour is similar according to the daily use and/or by time part.

Keywords *: Li'Bia bikes, dynamic networks, digital footprints, smart-cities, classification, ...*

En préambule à ce mémoire, je souhaitais adresser mes remerciements aux personnes qui m'ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire.

En premier lieu, je tiens à remercier Monsieur T. Carletti, qui, en tant que promoteur, s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et principalement le temps qu'il a bien voulu me consacrer et sans qui ce mémoire n'aurait jamais vu le jour.

Mes remerciements s'adressent également à JCDecaux et à la ville de Namur pour leur collaboration et pour nous avoir fourni les données nécessaires aux analyses effectuées.

Je n'oublie pas mes parents et mon compagnon pour leur contribution, leur soutien et leur patience et tout particulièrement ma maman qui a eu la gentillesse de lire et corriger ce travail.

Enfin, j'adresse mes plus sincères remerciements à tous mes proches et amis, qui m'ont toujours soutenue et encouragée au cours de la réalisation de ce mémoire.

Merci à toutes et à tous.

Laetitia Dehan

Table des matières

Introduction **1**

1 Objectifs et outils **3**
1.1 Les réseaux 4
1.2 Les descriptions statistiques 9
1.3 La classification 10

2 Données et pré-traitement **13**
2.1 Les données 13
2.2 Modifications et nettoyage des données 14

3 Le réseau Li'Bia Vélo **17**
3.1 Visualisation globale du réseau 19
3.2 Construction 25

4 Descriptions statistiques **27**
4.1 Durées des locations 28
4.2 Utilisation du réseau au fil du temps 29
4.3 Utilisation du réseau en fonction des abonnements 36
4.4 Utilisation du réseau en fonction des stations 41

5 Analyse du réseau **45**
5.1 Utilisation du réseau au fil du temps 45
5.2 Utilisation du réseau en fonction des abonnements 50

6 La classification **53**
6.1 Méthodes de classifications standard 53
6.2 La similarité cosinus 55

Conclusion **59**

Table des figures **65**

Liste des tableaux **65**

Appendices

Annexe A Programmes Matlab **69**
A.1 Nettoyage . . . 69
A.2 Poids des liens . . . 71
A.3 Rassemblement des poids . . . 73
A.4 Histogramme des temps de locations . . . 75
A.5 Distribution des locations en fonction du temps . . . 77
A.5.1 Programme Principal . . . 77
A.5.2 Fonction pour différencier les jours de semaine . . . 79
A.6 Locations par rapport aux abonnements . . . 80
A.6.1 Programme principal . . . 80
A.6.2 Fonction de sélection . . . 82
A.7 Utilisation du réseau en fonction des dates . . . 83
A.8 Sélection en fonction des données temporelles . . . 85
A.8.1 Programme principal . . . 85
A.8.2 Sélection entre deux dates . . . 86
A.8.3 Sélection entre deux heures . . . 87
A.8.4 Sélection par rapport à un ou plusieurs jours de semaine . . . 88
A.9 Sélection par rapport à la borne de sortie ou d'entrée . . . 89
A.10 Courbes d'utilisation des vélos pour chaque station . . . 90
A.11 Similarité cosinus . . . 93
A.12 Calcul des distances entre les stations . . . 94

Introduction

Depuis toujours, les villes, sociétés, infrastructures urbaines ou autres récoltent des informations sur la population afin de déterminer les comportements ou besoins de celle-ci et d'améliorer les services destinés à ce public. L'évolution des réseaux routiers en est une bonne illustration. Ceux-ci ne cessent d'évoluer en fonction du nombre de voitures qui augmente constamment [8].

Afin d'étudier ces comportements, il faut, dans un premier temps, récolter diverses informations. Leurs récoltes s'effectuent à l'aide de sondages, de formulaires contenant des questions avec des réponses sous la forme de choix multiples, etc. Il faut ensuite les rassembler et les analyser. Ces démarches peuvent alors être très coûteuses en temps et en argent. Désormais, l'avancée technologique nous permet d'obtenir une quantité importante d'informations avec une grande facilité et rapidité. Internet, ainsi que la création des réseaux sans fils, ont été les acteurs principaux de cette avancée. Beaucoup d'infrastructures urbaines comme les bus, trams ou métros utilisent déjà ces informations, que l'on nomme *digital footprints*, afin d'améliorer et d'agrandir au mieux leurs services.

Le but premier de ces récoltes d'informations est d'établir des structures s'adaptant à la demande de la population, à la vie actuelle de la population mais aussi au futur. Cela est possible grâce à ces méthodes qui ont évolué mais également grâce aux améliorations des méthodes d'analyses et aux nouveaux outils de visualisation qui sont utilisés pour étudier les villes ou les entreprises.

Beaucoup de travaux ont déjà été réalisés grâce aux *digital footprints*. T. Rattenbury *et al.* [9] ainsi que Girardin *et al.* [6], par exemple, utilisent les données fournies par le site *Flickr* [15], site de partage de photos, afin de déterminer les flux de personnes dans les villes et les liens possibles avec des évènements ayant lieu à ces mêmes endroits. En effet, ce site de partage enregistre non seulement les photos, mais également les endroits où ont été prises ces photos, à l'aide de la longitude et de la latitude. T. Rattenbury *et al.* peuvent alors déterminer si ces flux de personnes sont des flux de touristes, d'habitants ou autre et faire un lien avec les endroits principaux de la ville. Sur la FIGURE 1, nous pouvons observer, grâce aux analyses effectuées ([6]), que lorsque des touristes se "taguent" sur des photos avec le mot "ruines", ils se trouvent au niveau du Coliseum et du Forum, qui sont des lieux très touristiques de la ville de Rome et réputés pour leurs ruines anciennes.



FIGURE 1 – La zone rose correspond aux coordonnées des photos provenant du site *Flickr* sur lesquelles le mot "ruines" a été "tagué" - [6]

Grâce à ces évolutions, nous avons aussi pu voir naître les smart-cities, ces villes intelligentes où tout est fait pour faciliter la vie des habitants et des entreprises. Ces villes utilisent des réseaux avec des capteurs, situés à différents endroits, qui captent les informations mais qui les transmettent également aux citoyens et aux infrastructures de manière interactive grâce, par exemple, à des panneaux d'informations situés dans les rues et qui affichent ces informations. Les smart-cities peuvent être évaluées selon différents critères :

- Smart Economy : basé sur la compétitivité régionale ;
- Smart Mobility : basé sur l'économie des transports et des technologies de l'information et de la communication ;
- Smart Environment : basé sur les ressources naturelles ;
- Smart People : basé sur les capitaux humains et sociaux ;
- Smart Living : basé sur la qualité de vie ;
- Smart Governance : basé sur la participation des citoyens à la vie démocratique de la ville.

Le but est donc d'améliorer les systèmes, à l'aide du flux d'informations récoltées en continu, afin d'augmenter le niveau de la ville dans ces différents domaines. Comme montré sur le site d'IBM [21], cela permettrait, dans le cas du trafic, de réguler le trafic des transports en commun. En effet, de nos jours, lorsqu'un transport est en retard, c'est tout le système qui est perturbé. Dans une smart-city, les heures de départ sont recalculées grâce aux informations récoltées en direct, ce qui permet de réguler le trafic.

Dans notre cas, ce sont les *digital footprints* fournies par le réseau de vélos libres de la ville de Namur fournies par JCDecaux qui vont nous servir à analyser ce réseau. Des études similaires ont déjà été publiées pour les villes de Lyon [2] et de Barcelone [4].

Ce document est composé de six chapitres. Le premier décrit les différents objectifs à atteindre ainsi que les outils qui permettront de réaliser ceux-ci. Les chapitres 2 et 5 nous fournissent des renseignements et des visualisations à propos du réseau sur lequel nous travaillons, une description complète des données est fournie dans le chapitre 2 afin de permettre la bonne compréhension des analyses effectuées par la suite. Le troisième chapitre nous fournit des analyses statistiques effectuées sur les données. Le dernier chapitre quant à lui concerne la classification des stations en fonction de leur utilisation.

Chapitre 1

Objectifs et outils

Durant ce mémoire, nous allons analyser le comportement de la population utilisant le réseau Li'bia Vélo mis en place par la société JCDecaux dans la ville de Namur. Le système étant informatisé, une quantité d'informations est disponible à propos des vélos : le lieu de départ parmi les différentes stations du réseau et donc un lieu géographique dans la ville de Namur, l'heure de départ, l'heure d'arrivée. La totalité des données fournies sont énoncées et décrites dans la section 2.1. À l'aide de ces informations, nous allons pouvoir étudier les différents mouvements des vélos et de leurs utilisateurs dans le temps mais également dans l'espace, suivant les informations obtenues. Quel est le trajet moyen ou le temps d'utilisation moyen d'un vélo ? Y a-t-il des stations plus souvent vides ou pleines que d'autres ? Quelle est la quantité d'emplacements libres ou occupés ? Y a-t-il « flux net » de déplacement de vélos d'un côté à l'autre de la ville ? La quantité de vélos utilisés dépend-elle du jour, de l'heure, des lieux d'intérêt à proximité ou des éventuels événements publics ? Il est alors possible, grâce aux *digital footprint*, à la construction d'un modèle statique et d'un modèle dynamique du réseau de vélos de trouver réponse à toutes ces questions.

Dans ce chapitre, nous présenterons une série d'objectifs sur lesquels il sera intéressant de travailler. La totalité de ces objectifs nous permettra d'effectuer des prédictions, ce qui peut être très intéressant afin d'optimiser le système. Nous pourrons prédire l'intensité d'utilisation des vélos et déterminer les heures durant lesquelles les vélos sont les plus utilisés ainsi que les endroits où ceux-ci manquent ou sont en trop. Le tout permettra une optimisation du système de "régularisation" du système qui consiste à remplir les stations vides et vider les stations pleines.

Nous n'auvons pas eu le temps d'arriver jusqu'à ce point mais l'idéal serait de pouvoir fournir l'utilisation du réseau de vélos en temps réel et en direct. En effet, cela est déjà d'actualité pour, par exemple, les villes de Londres et de New York, via le site [10]. Ce site nous permet également de voir l'utilisation au cours du temps sur plusieurs jours et nous pouvons, par exemple pour la ville de Londres, voir très clairement que les stations du centre de la ville sont fortement remplies la journée et commencent à se vider en début de soirée. En attendant, nous nous contenterons d'effectuer une visualisation sur une journée, comme cela a été fait pour la ville de Paris - [26].

1.1 Les réseaux

Dans un premier temps, la construction d'un réseau statique abstrait peut être une information utile pour clarifier les données ainsi que la visualisation du réseau des vélos en lui-même. Les noeuds de ce réseau seront les stations tandis que les liens correspondront à la quantité de vélos échangés entre chaque station. Pour cela, nous aurons besoin des lieux de départ et d'arrivée de chacun des vélos utilisés. Afin de représenter ce réseau, nous pourrons utiliser des logiciels tels que *Gephi* [18] et *Circos* [7], qui permettent d'obtenir de bons résultats graphiques.

Gephi

Gephi permet de représenter des graphes de réseaux pour lesquels différents types de visualisations sont possibles. Ce logiciel travaille à partir de deux tables, fournies sous le format `csv`, l'une contenant les informations sur les noeuds, l'autre contenant les informations sur les liens. La table des noeuds doit être composée des colonnes suivantes :

1. Id : numéro d'identification de la station.
2. Label : nom donné à la station.
3. Lat : latitude de la station.
4. Lng : longitude de la station.

Notons que les colonnes 3 et 4 ne sont nécessaires que si nous souhaitons représenter le graphe du réseau de manière réaliste, de telle façon que les stations soient positionnées en fonction de leur géolocalisation, ou sur une carte satellite, à l'aide de *Google Earth* [19] par exemple. En effet, après avoir construit la représentation du réseau via *Gephi*, ce logiciel stocke les informations (latitude, longitude, label et Id) dans les noeuds, ce qui permet d'importer toutes les informations dans *Google Earth*.

La table des liens, quant à elle, doit contenir les colonnes qui suivent :

1. Source : Id de la station de sortie.
2. Target : Id de la station de retour.
3. Weight : poids des liens entre les deux stations que le vélo relie.
4. SourceNom : nom de la station de sortie.
5. TargetNom : nom de la station de retour.

Sur les FIGURES 1.1 et 1.2[1] nous pouvons voir deux exemples de réseaux construits avec ce logiciel. La FIGURE 1.1 représente les collaborations présentes entre les différents départements des facultés de l'Université de Namur présents dans le fichier de données[2]. Les noeuds de ce graphe représentent les départements de l'UNamur et un lien est tracé entre deux noeuds à chaque fois que des membres appartenant à chaque noeud cosignent une publication. Notons que la taille des liens est proportionnelle au nombre d'articles cosignés par les membres des deux noeuds qu'il relie tandis que la taille de ces derniers est, à la fois, proportionnelle au nombre de personnes appartenant au département et au nombre de liens qui arrivent sur ceux-ci car le logiciel a regroupé les personnes en fonction du département auquel elles appartiennent.

1. Graphes construits par un groupe d'élèves de Master 1 en Mathématique dans le cadre du cours de Questions de probabilités durant l'année 2012-2013.

2. Données provenant du site Web Of Sciences [29] et fournies par Mr Lambiotte dans le cadre du cours de Questions de probabilités donné en Master 1 Mathématique durant l'année 2012-2013.

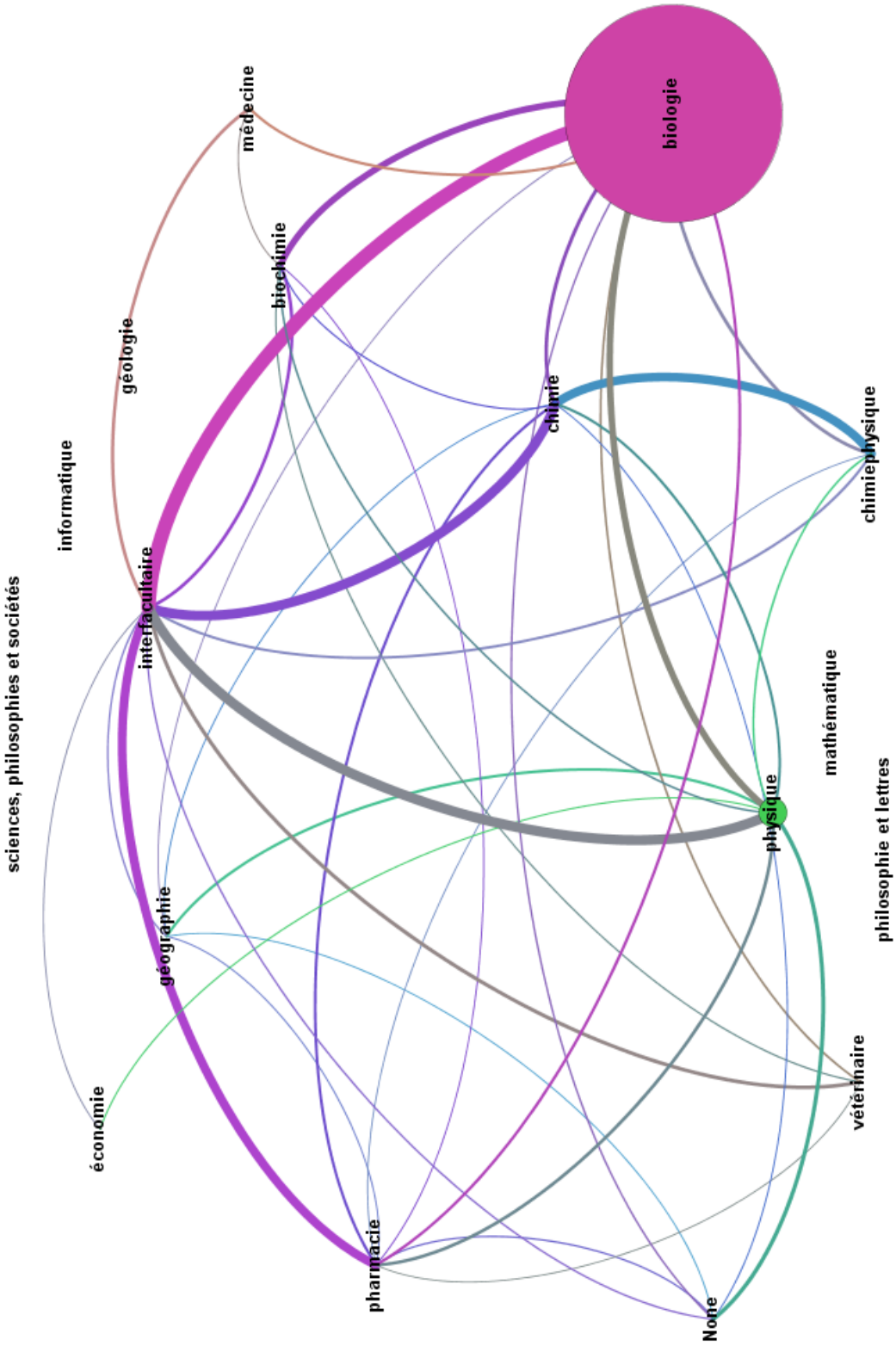


FIGURE 1.1 – Collaborations scientifiques entre les différents départements de l'Université de Namur.

Sur la FIGURE 1.2, nous pouvons observer le graphe représentant les transferts entre les différentes stations de vélos de la ville de Namur. Dans ce cas, les noeuds représentent les stations tandis que les liens représentent le nombre de vélos échangés entre deux stations. La taille ainsi que la couleur des différents noeuds sont proportionnelles au nombre de liens qui rejoignent ces noeuds (de 0 à 24 liens) tandis que la taille et la couleur des liens représentent la proportion du nombre de vélos échangés entre les stations que le lien relie. Le réseau de vélos en lui-même sera décrit en détails par la suite.

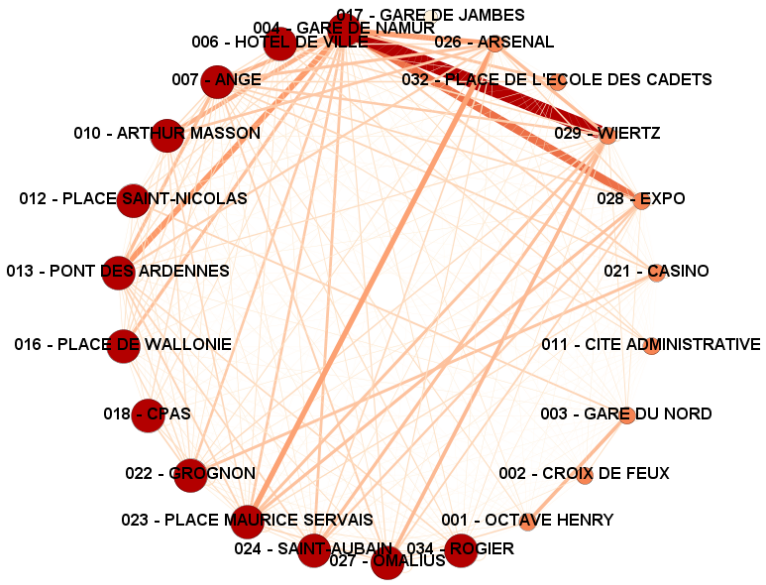


FIGURE 1.2 – Transferts de vélos entre les différentes stations de la ville de Namur - construite avec *Gephi*.

Grâce à ce programme, *Gephi*, nous pourrons construire un réseau similaire à celui décrit dans le paragraphe précédent mais avec, comme particularité, une notion de temps. Nous aurons, cette fois-ci, des liens dont le poidd est déterminé par le nombre de vélos échangés entre deux stations par tranche temporelle, c'est-à-dire, par intervalle de temps. Cela nous fournira donc le nombre de vélos échangés entre une station A et une station B sur une journée type, une semaine ou toute autre période de temps que nous choisirons.

Pour chaque vélo nous connaissons le moment de la prise du vélo (date, heures, minutes et secondes) et le moment de sa restitution. Pour représenter ces données nous devrons les agréger en tranches temporelles qui soient "significatives", c'est-à dire qui ne soient ni trop petites (pour éviter de ne pas avoir assez de données) ni trop grandes (pour ne pas perdre toute l'information temporelle).

Circos

Comme *Gephi*, *Circos* [11] est un logiciel de visualisation de données. Sa particularité est qu'il représente automatiquement les données sous la forme d'un cercle, ce qui est idéal pour analyser les similarités entre celles-ci. Afin de créer des graphes de type *Circos*, il faut soumettre les données que l'on souhaite analyser au site [12], dans un fichier `txt` et sous la forme d'une matrice. La première ligne et la première colonne de cette matrice doivent contenir les éléments entre lesquels nous souhaitons tracer les liens dont les poids sont les nombres qui composent le reste de la matrice. Afin de mieux comprendre ce type de visualisation, nous avons utilisé un exemple provenant directement du site [12].

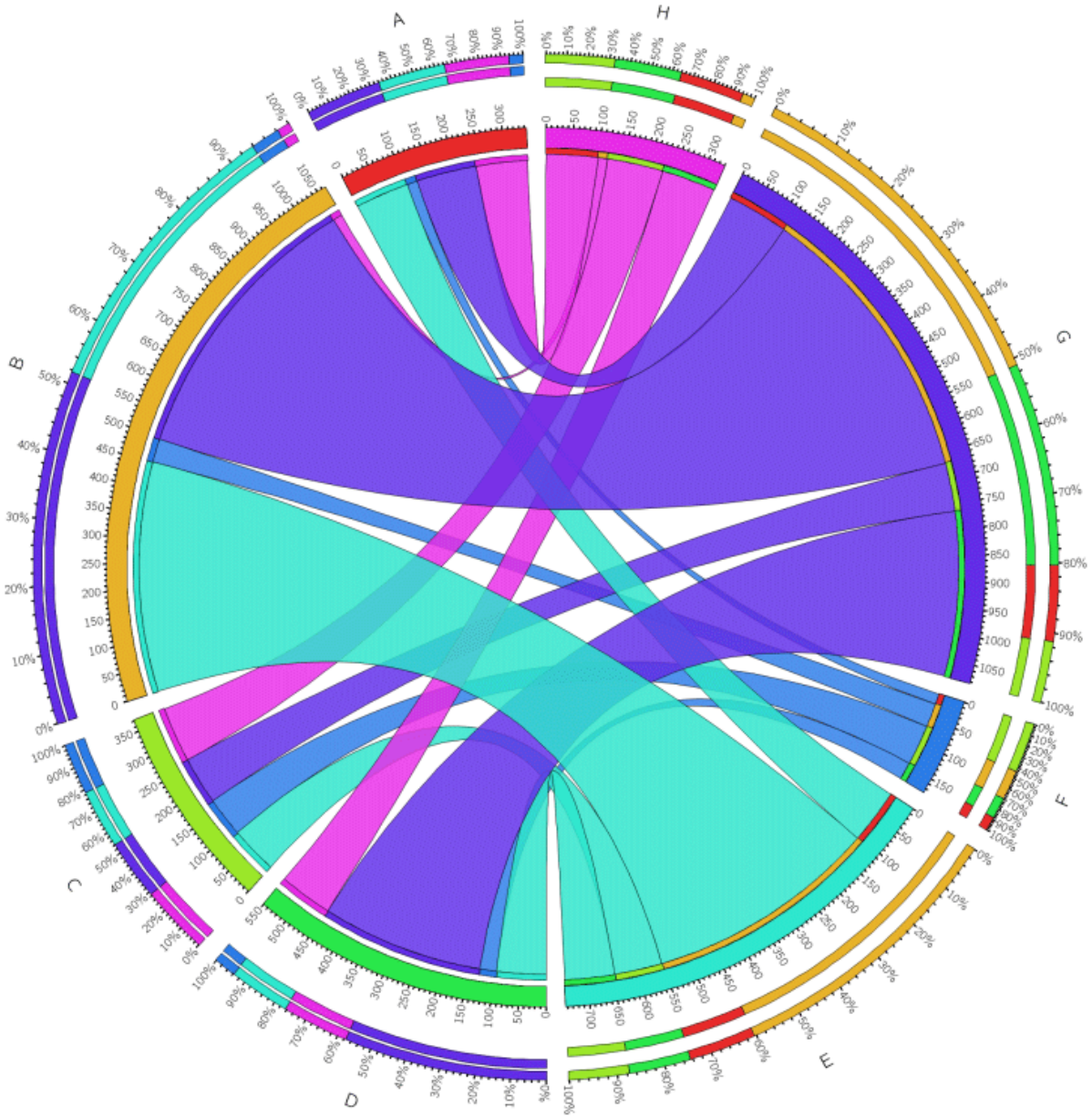
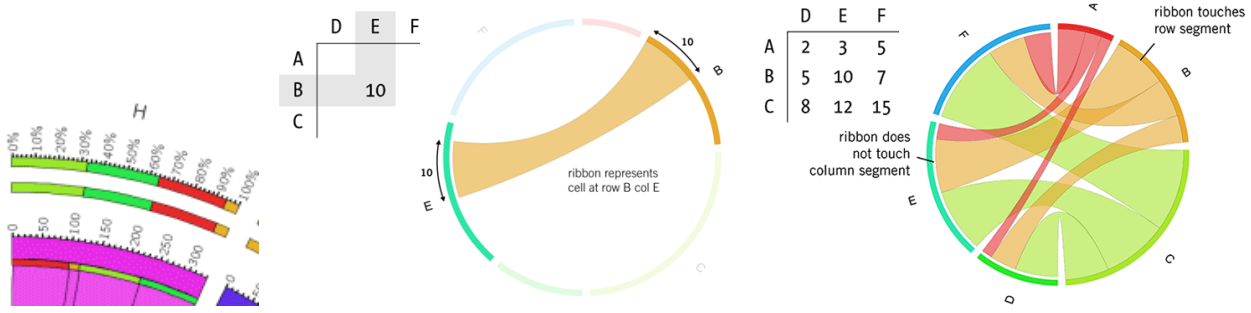


FIGURE 1.3 – Visualisation *Circos* des données de la TABLE 1.1 - [12].

La FIGURE 1.3 permet de visualiser les données de la TABLE 1.1. Nous pouvons voir que les éléments de la première ligne et de la première colonne de cette table sont représentés à l'extérieur du graphe par des sections d'arcs de cercles (exemple sur la FIGURE 1.4(a)). Ce sont nos noeuds auxquels on associe une couleur unique, qui sera également reprise pour les liens sortants de ce noeud, ce type de graphe peut donc être orienté grâce aux couleurs. Nous pouvons voir que nous avons trois arcs différents pour chaque noeud dont un plus épais, à l'intérieur. Cet arc permet d'évaluer le poids des liens échangés entre les noeuds et permet de différencier ces derniers à l'aide de la couleur qui leur est associée. Sur la FIGURE 1.4(a), par exemple, la couleur associée au noeud H est le rose. Les autres arcs représentent les proportions d'échanges entre les différents noeuds. 30% des liens sortants ou rentrants du noeud H par exemple, sont dirigés ou viennent du noeud portant la couleur vert clair qui est le noeud C lorsque l'on regarde sur la FIGURE 1.3. Les liens entre les différents noeuds sont représentés par des rubans dont la largeur est proportionnelle aux poids qui composent la matrice (voir FIGURE 1.4(b)). En effet, nous pouvons voir, sur la FIGURE 1.3, que le ruban le plus épais portant un poids de 553 relie bien les noeuds G et B. Rappelons que dans notre cas, les liens portent la couleur associée à leur station de départ. Notons également que les liens qui partent d'un noeud touchent son arc tandis que lorsqu'ils arrivent à un autre, il y a un espace entre l'arc du noeud d'arrivée et la fin des liens, comme nous pouvons le voir sur la FIGURE 1.4(c).

data	A	B	C	D
E	105	20	118	100
F	450	46	553	18
G	92	78	94	108
H	96	33	317	104

TABLE 1.1 – Exemple de table pour un graphe *Circos* - [12]



(a) (b) (c)

FIGURE 1.4 – Explications construction graphe *Circos* - [13]

Grâce à ce type de graphe, nous pourrons clairement visualiser les échanges entre les stations de notre réseau. Comme pour les représentations faites via *Gephi*, les noeuds représenteront les stations tandis que les liens représenteront la quantité de vélos échangés entre les stations. Nous aurions pu nous contenter du premier logiciel mais *Circos* fournit des graphes plus clairs et contenant plus d'informations. Nous pouvons clairement discerner les liens entrants et sortants ainsi que les plus importants. Cela nous permettra, dans le cas des vélos, d'avoir directement les quantité d'échanges entre les différentes stations du réseau pour avoir une idée, en un coup d'oeil, des transferts les plus importants et des stations les plus utilisées.

1.2 Les descriptions statistiques

Avec la même condition sur les intervalles de temps énoncée dans la section précédente, nous pourrons également déterminer la quantité totale de vélos utilisés par tranche horaire, les temps caractéristiques comme la moyenne et la médiane ou la distribution de l'utilisation des vélos, grâce à des logiciels statistiques tels que *R* [25], *Matlab* [23], *SAS* [27],...

Ce type d'analyse pourra également nous montrer l'impact de la gratuité des 30 premières minutes d'utilisation des vélos, comme le montrent P. Borgnat *et al.*, à la page 419 de leur article [2], en analysant le réseau de vélos libres de la ville de Lyon. Nous pouvons remarquer, à l'aide de la FIGURE 1.5 provenant de cet article, que les plus gros pourcentages regroupent des locations qui durent entre 0 et 25 minutes, ce qui montre clairement l'impact de la gratuité des 30 premières minutes. De plus, cela nous permettra d'avoir une idée du nombre de kilomètres effectués en vélo en 30 minutes et de voir si cette distance a une quelconque relation avec les caractéristiques géographiques du lieu.

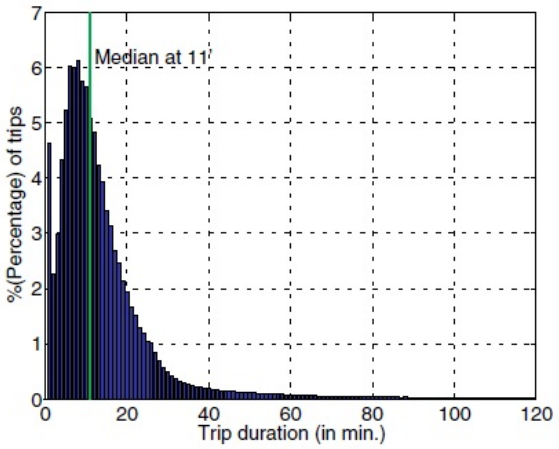


FIGURE 1.5 – Distribution des temps de location des vélos pour la ville de Lyon - [2]

Nous pourrons également déterminer les heures durant lesquelles les vélos sont les plus sollicités. Suite à ces résultats, nous pourrons distinguer une différence d'utilisation entre la semaine et le week-end, les vacances et les périodes scolaires, voir si certains évènements, ayant lieu dans la ville de Namur durant l'année, ont une influence sur l'utilisation du réseau de vélos libres, ou tout simplement l'effet des saisons.

Ces types de conclusions ont été réalisés par J. Froehlich *et al.* pour la ville de Barcelone [4]. Nous pouvons voir, sur la FIGURE 1.6 provenant de cet article, qu'il existe une nette différence d'utilisation des vélos au cours d'une journée ((a) et (b)) ou au cours de toute une semaine (c).

Nous pouvons remarquer sur l'image (a), représentant la quantité de vélos libres le week-end et un jour de semaine à la station 37, une nette différence d'utilisation des vélos entre la semaine et les week-ends. En effet, durant la matinée du jour de semaine, le nombre de vélos disponibles augmente pour ensuite diminuer fortement peu avant 16h. Le week-end, il semblerait que le nombre de vélos libres diminue à partir de minuit pour ensuite augmenter peu avant midi. L'utilisation durant le week-end est donc inverse à l'utilisation des vélos durant la semaine.

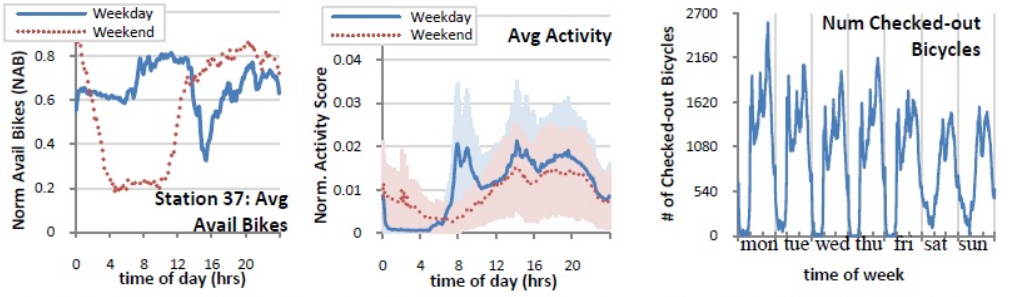


Figure (a) Normalized weekday and weekend DayViews of available bicycles at Station 37.

Figure (b) Normalized weekday and weekend DayViews of activity scores for all stations.

Figure (c) The total number of bicycles checked-out from all stations over a week.

FIGURE 1.6 – Utilisation des vélos sur une journée pour une station (a), pour toutes les stations (b) et au cours de la semaine (c) - [4]

Grâce à l'image (b), qui représente l'utilisation des vélos de toutes les stations sur une journée, et l'image (c), qui représente l'utilisation de tous les vélos sur une semaine, nous pouvons constater trois pics d'utilisation durant les jours de semaine. Ceux-ci ont une explication très logique : le premier correspond à l'utilisation matinale des vélos, après l'heure du petit-déjeuner, les habitants partent pour aller travailler ; le second, situé aux alentours de 14h, se situe à l'heure du déjeuner qui est un peu plus tardive pour les espagnols, dû à leur rythme de vie ; le dernier correspond aux heures de retour des employés ou des étudiants à leur domicile. Le week-end, nous pouvons voir que les vélos ne sont réellement utilisés qu'à partir de midi, nous n'avons donc pas la présence des trois pics d'utilisation que nous avons constatés auparavant. L'utilisation des vélos est donc fortement dépendante des jours de la semaine.

1.3 La classification

Certains chercheurs ont également effectué des classifications sur les stations de vélos réparties dans la ville sur laquelle ils ont travaillé, afin de comprendre, spatialement, comment était distribuée l'utilisation des vélos sur le réseau. J. Froehlich, par exemple dans son article [4], a classifié les stations de la ville de Barcelone en fonction de leur densité et de leurs périodes d'utilisation. Nous pouvons observer ces classifications sur la FIGURE 1.7.

La première visualisation du réseau, à gauche, ainsi que les graphes qui se situent juste en-dessous, sur la FIGURE 1.7, nous donnent la classification des stations en fonction de leur intensité d'utilisation. Les groupes fournis par cette classification sont notés de A1 à A5 et augmentent en intensité d'utilisation, comme nous pouvons le voir sur les graphes.

La seconde visualisation et les graphes de droite, eux, classifient les stations en fonction de leurs caractéristiques d'utilisation, c'est-à-dire en fonction du nombre de vélos qui sont disponibles ou en fonction des places qui sont libres. En effet, ici la classification nous fournit six classes avec trois types de comportements, les stations de sortie, qui correspondent aux classes B3 et B4 et pour lesquelles nous pouvons voir sur le graphique que le nombre de vélos disponibles diminue à partir de 8h, les stations de retour, qui composent les classes B5 et B6 et pour lesquelles nous pouvons voir que le nombre de vélos disponibles augmente à partir de 8h, et les

stations stables, qui correspondent aux classes B1 et B2 et pour lesquelles le nombre de vélos disponibles reste constant tout au long de la journée.

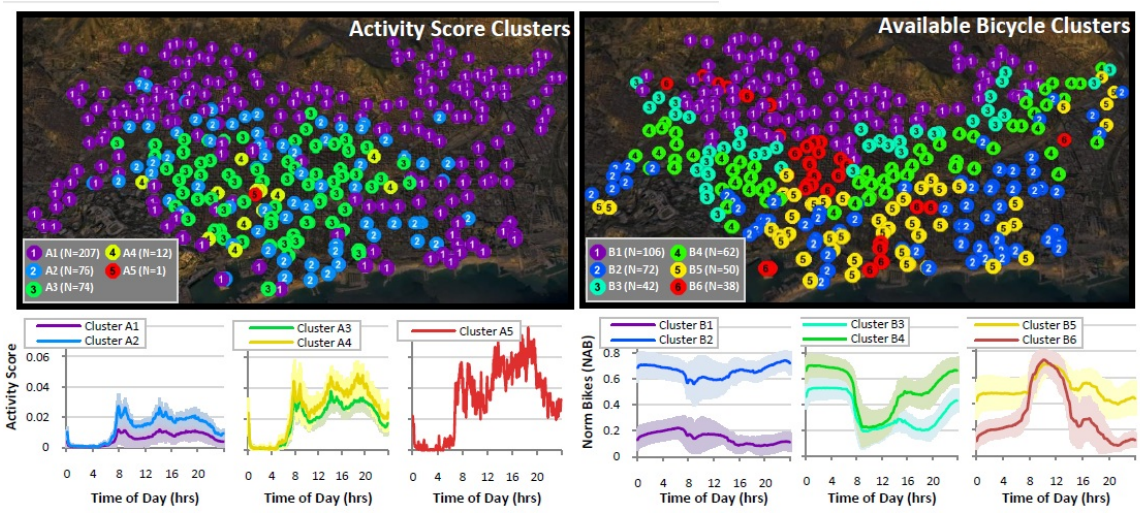


FIGURE 1.7 – Classification des différentes stations de la ville de Barcelone - [4]

Chapitre 2

Données et pré-traitement

Le but est d'analyser et de modéliser le réseau de vélos en libre service de la ville de Namur. Pour cela, une demande de collaboration a été introduite auprès de la Ville de Namur ainsi qu'auprès de JCDecaux qui nous a fourni les données utiles à ces analyses en avril 2013 et couvrant les douze mois précédents.

2.1 Les données

Les données nous ont été transmises sous la forme de deux tables `xls`, contenant près de 80 000 données qui couvrent un intervalle de temps compris entre le mois d'avril 2012 et le moment où nous avons reçu les données, ce qui correspond à une année.

Première table

La première table fournie contient les coordonnées GPS des différentes stations qui composent le réseau. C'est grâce à cette table que nous allons pouvoir représenter les différents graphes du réseau sur des cartes réelles. Les colonnes qui nous intéressent sont organisées de la manière suivante.

1. Adresse de station : rue et, si nécessaire, numéro du lieu où se situe la station.
2. Marker/lat : latitude de la station.
3. Maker/lng : longitude de la station.
4. Nom de station : nom sous lequel la station est répertoriée dans les fichiers de données.
5. Numéro de station : numéro sous lequel la station est répertoriée dans les fichiers de données, plus communément appelé l'Id de la station. Il faut savoir que nous avons 24 stations mais que ces stations ne sont pas numérotées de 1 à 24, les nombres peuvent aller au-delà de cet intervalle. La station dont l'Id est le numéro maximal, par exemple, est la station 34 portant le nom Rogier.

Seconde table

La seconde table, qui est plus conséquente du point de vue de la taille, nous fournit les informations concernant les transferts de vélos entre les différentes stations. Chaque ligne correspond donc au trajet d'un vélo. Cette table est construite de la manière suivante.

1. Borne de sortie : station de départ du vélo. Toutes les informations de cette station sont contenues dans la première table (Id, situation, nom).

2. Bornette de sortie : point d'attache auquel le vélo a été retiré. Les stations sont toutes composées d'un certain nombre de points d'attache qui sont également numérotés.
3. Types de trajet : type de trajet pour lequel le vélo a été retiré de la station. Cinq types de sorties apparaissent dans cette colonne : la sortie pour location, la sortie pour régulation (vider les stations pleines et remplir les stations vides), la sortie pour stockage, la sortie pour réparation en atelier et la sortie pour maintenance.
4. Date et heure de sortie : date, ainsi que l'heure, auxquelles le vélo a été retiré de la station de sortie.
5. Type d'abonnement : nous informe du type d'abonnement que possède la personne qui loue le vélo. Il existe 4 types d'abonnements : l'abonnement de longue durée, pour les personnes ayant un abonnement de location d'un an ; le CD-1 jour, pour les personnes qui ont acheté un ticket de location valable une journée ; le CD-1 semaine, pour les personnes qui ont acheté un ticket de location valable une semaine ; l'administrateur, qui correspond aux personnes régulant le réseau ou qui stockent les vélos pour réparations ou autres, souvent associé à une sortie pour régulation, stockage ou réparation.
6. Numéro du vélo : numéro du vélo qui est loué.
7. Flag retour : confirme si le vélo a bien été rapporté après sa sortie.
8. Date et heure de retour : date et heure auxquelles le vélo a été rapporté à la station d'arrivée.
9. Borne de retour : station d'arrivée du vélo, dont toutes les informations sont également répertoriées dans la première table décrite auparavant.

2.2 Modifications et nettoyage des données

Afin de pouvoir utiliser la seconde table de données de manière complète, nous avons dû y ajouter quelques modifications.

Dans un premier temps, nous avons rassemblé toutes les données et avons remplacé les colonnes Borne de Sortie et Borne de Retour par deux colonnes qui correspondent uniquement aux numéros des stations de sortie et de retour. De cette manière, nous n'avons plus de texte. Pour ce faire, nous avons été rechercher, dans la première table de données et à l'aide de la fonction `RECHERCHEV` disponible dans le logiciel *Microsoft Excel*, à quel Id (ou numéro de station) correspondait chacun des noms des stations de départ et d'arrivée, respectivement appelées les stations de sortie et de retour. Lorsque la station n'est pas répertoriée parmi celles que nous avons, nous avons mis 0. Ensuite, nous avons décidé de supprimer les colonnes Bornette de sortie et Numéro du vélo, car celles-ci ne nous fournissaient pas d'informations intéressantes lors des différentes analyses effectuées. À l'aide des fonctions `MOIS` et `JOURSEM` de *Microsoft Excel*, nous avons également ajouté les colonnes Mois de Sortie/Retour et Jours de Sortie/Retour. Nous verrons l'utilité de ces dernières dans la section 4.2. Après ces modifications, nous obtenons une table identique à la Table 2.1.

Afin de pouvoir manipuler les données qui sont sous le format de chaîne de caractères plus facilement, nous avons utilisé le logiciel *SAS* dans le but de supprimer les données pour lesquelles le Flag de retour était non. En effet, lorsque le Flag correspond à non, cela signifie que le vélo n'a jamais été retourné à une station, il n'est donc pas utile d'analyser les temps de location ou les trajets de ces vélos. Cette sélection nous a permis, par exemple, de supprimer

une location de plus d'un an.

Num Sortie	1
Types de trajet	Sortie pour location
Date de sortie	21/04/2012 09 : 41 : 24
Mois de sortie	4
Jour de sortie	6
Date de retour	21/04/2012 09 : 42 : 15
Mois de retour	4
Jour de retour	6
Flag retour	Oui
Num Retour	1
Type d'abonnement	Longue Durée

TABLE 2.1 – Exemple représentant une entrée de la seconde table avec ajout des numéros des stations de départ et d'arrivée, des jours et des mois.

La notion du temps

Comme nous l'avons vu dans la section 2.1, les données fournies nous permettent d'introduire une notion temporelle dans nos analyses. En effet, pour chaque vélo utilisé nous avons les heures et les dates, de sorties et de retours, nous avons donc choisi de supprimer les locations de moins de 60 secondes. Nous avons estimé qu'une location inférieure à une minute ne correspondait pas à une location destinée à un trajet entre deux stations mais plutôt à un détachement suivit d'une remise immédiate du vélo à la même station, ce n'est donc pas pertinent d'analyser ces déplacements. Cependant, le format sous lequel les données ont été encodées ne nous permet pas d'effectuer des calculs et des analyses directement car les données temporelles sont fournies sous un format "Jour/Mois/Année Heures : Minutes : Secondes" et celui-ci est considéré comme un format texte par *Matlab*, et non numérique, ce qui n'est pas adapté à des calculs numériques ou statistiques.

Grâce à *Matlab* et à la fonction `datenum`, il est possible de transformer des données temporelles en *serial date numbers*. Ce type de nombre correspond à un numéro de série propre à chacune des dates et/ou heures. Le numéro de référence, qui est le numéro 1, est le *serial date number* correspondant au 1/01/9000 00 : 00 : 00. Toutes les opérations arithmétiques sont alors possibles vu que nous avons des données numériques. Lorsque l'on soustrait les *serial date numbers* de deux dates par exemple, cela revient à soustraire les deux dates elles-mêmes, ce qui nous donne la différence entre celles-ci ou plus précisément le *serial date number* de cette différence, qui est donc un numéro de série. Ce nombre peut être réexprimé sous le format "Jour/Mois/Année Heures : Minutes : Secondes" grâce à la fonction `datestr`.

Dans un premier temps, nous avons directement transformé et utilisé les données telles qu'elles étaient. Les programmes construits au fur et à mesure utilisaient les *serial date number* des données enregistrées sous le format "Jour/Mois/Année Heures : Minutes : Secondes" qui étaient dans les colonnes Date de sortie et Date de retour. Mais par la suite, afin de faciliter les

différentes manipulations qui allaient être effectuées sur ces colonnes, nous les avons séparées en trois colonnes de formats différents :

- Date de sortie/retour sous le format "Jour/Mois/Année" ;
- Heure de sortie/retour sous le format "Heures : Minutes : Secondes" ;
- Heure de sortie/retour2 sous le format "0/01/1900 Heures : Minutes : Secondes".

Num Sortie	1
Types de trajet	Sortie pour location
Date de sortie	21/04/2012
Mois de sortie	4
Jour de sortie	6
Heure de sortie	09 : 41 : 24
Heure de sortie2	0/01/1900 09 : 41 : 24
Date de retour	21/04/2012
Mois de retour	4
Jour de retour	6
Heure de retour	09 : 42 : 15
Heure de retour2	0/01/1900 09 : 42 : 15
Flag retour	Oui
Num Retour	1
Type d'abonnement	Longue Durée
Date_Heure sortie	21/04/2012 09 : 41 : 24
Date_Heure retour	21/04/2012 09 : 42 : 15

TABLE 2.2 – Exemple représentant une entrée de la seconde table avec séparations des colonnes des données temporelles.

La TABLE 2.2 nous permet de voir les changements effectués. Chacun de ces trois formats va être enregistré de manière différente dans *Matlab*. En effet, la première et la troisième colonne, respectivement, vont être enregistrées sous un format texte de la forme "Jour/Mois/Année" et "Heures : Minutes : Secondes", respectivement. Ces données seront donc transformées à l'aide de la fonction `datenum` pour pouvoir les utiliser étant donné qu'elles sont sous un format texte au départ. La seconde colonne, quant à elle, sera enregistrée directement sous un format numérique qui représente la fraction de la journée à laquelle correspond l'heure importée. Si nous importons 12 : 00 : 00 par exemple, *Matlab* associera ce moment au nombre 0,5 et non à $7,3560.10^5$, qui est le *serial date number* donné lorsque nous appliquons la fonction `datenum` au texte "12 : 00 : 00". Nous verrons, tout au long des analyses, qu'il est important de garder les deux formats suivant les analyses effectuées.

Après ces différentes modifications, nous avons pu construire le programme se trouvant à l'annexe A.1. Celui-ci permet de transformer les dates en *serial date number* et calcule les temps de location des vélos en secondes. Les données et les résultats sont sortis sous un fichier de données *Matlab*, c'est à dire un fichier `mat`. Ce type de fichier permet d'utiliser les données plus facilement étant donné qu'elles seront toujours enregistrées sous le même format. De plus, l'importation d'un fichier `mat` est plus rapide que celle d'un `xls` en *Matlab*, surtout pour des fichiers de taille importante. Notons qu'au départ nous avions 89 916 données et qu'après les différentes modifications apportées nous avons conservé 77 199 données.

Chapitre 3

Le réseau Li'Bia Vélo

Le réseau des Li'bia Vélos a été mis en place par JCDecaux dans le courant de l'année 2012. Ce réseau recouvre la ville de Namur mais aussi les villes de Jambes, Saint-Servais, Salzinnes et Bomel, à l'aide de 24 stations, espacées en moyenne de 1,11 km (le programme calculant la matrice des distances entre les stations est disponible à l'annexe A.12), et de 240 vélos. Ce système étant informatisé, il peut être utilisé au moyen d'une carte bancaire ou d'une carte d'abonnement d'un an, 24h sur 24, 7 jours sur 7.

Concernant le coût de l'utilisation de ces services, il existe un abonnement d'un an, pour les utilisateurs réguliers, mais également des tickets de 1 ou 7 jours, pour les utilisateurs occasionnels. Ensuite, la première demi-heure d'utilisation des vélos est gratuite, tandis que le reste du temps d'utilisation est payant (voir FIGURE 3.1 pour les différents tarifs).

	Abonnement 1 an	Ticket 1 jour	Ticket 7 jours
Prix de base	30 €	1 €	3 €
Nombre de déplacements	Illimité	Illimité	Illimité
De 0 à 30 minutes	Gratuit	Gratuit	Gratuit
1ère demi-heure supplémentaire	+0,50 €	+0,50 €	+0,50 €
2ème demi-heure supplémentaire	+1 €	+1 €	+1 €
A partir de la 3ème 1/2h suppl et par 1/2h supplémentaire	+2 €	+2 €	+2 €
> 2h par 30 mn supplémentaire	+2 €	+2 €	+2 €

FIGURE 3.1 – Tarifs des locations des Li'Bia vélos - [22]

Chaque vélo peut être pris à une station et déposé à une autre, il n'est donc pas nécessaire de revenir à son point de départ. Grâce au système informatique, l'heure d'arrivée et de départ ainsi que la station d'arrivée et de départ sont automatiquement enregistrées. C'est grâce à ces données que nous allons pouvoir faire nos analyses.

La disponibilité des vélos ainsi que les places libres pour restituer les vélos sont disponibles à chaque borne, située aux différentes stations, ainsi que sur le site Li'Bia Vélo [22], à l'aide d'une visualisation sur une carte provenant de *Google Map* [20]. Depuis peu, ces informations sont également disponibles via l'application AllBikesNow, qui est gratuite.

La FIGURE 3.2 représente la carte de la ville de Namur sur laquelle nous avons représenté les différentes stations du réseau Li'Bia Vélo à l'aide de points rouges et blancs. Chaque point porte un libellé qui correspond à l'Id de la station qu'il représente. Cette représentation est possible grâce aux latitudes et longitudes des différentes stations fournies par JCDecaux dans le premier fichier `xls`, dont les détails sont donnés dans la section 2.1. Afin de pouvoir utiliser *Gephi*, il a tout d'abord fallu transformer notre fichier `xls` en un fichier au format `csv` pour pouvoir l'utiliser comme table de noeuds dans ce logiciel. De cette manière, le logiciel a pu construire un noeud, représenté ici sous la forme d'un point rouge, pour chacune des stations. Finalement, afin d'avoir cette représentation sur carte, nous avons transporté les noeuds, correspondant aux stations et construits par *Gephi*, ainsi que leurs données (Id, label, latitude et longitude) dans *Google Earth*, à l'aide d'un fichier de format `kmz`. Grâce aux latitudes et longitudes des différentes stations, *Google Earth* a pu les replacer de manière précise sur la carte de la ville de Namur. Chacune des stations est représentée par un point rouge mise à part la station 15, qui est représentée par un point blanc car celle-ci est en cours de construction. Comme dit précédemment, les stations sont annotées par leur numéro d'identification (Id), et non pas par leur nom complet, cela dans le but d'avoir une meilleure visibilité.

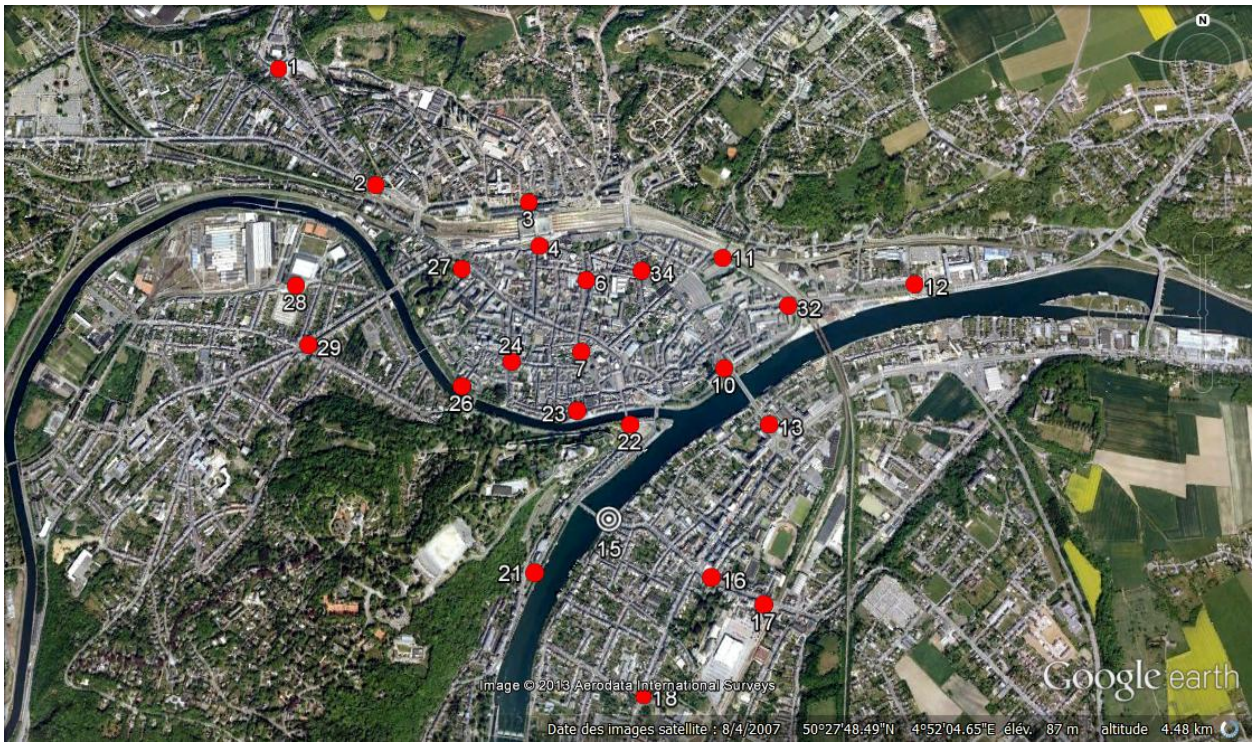


FIGURE 3.2 – Carte des stations des vélos libres dans la ville de Namur construite à l'aide de *Gephi* et de *Google Earth*

3.1 Visualisation globale du réseau

Après avoir placé les stations sur une carte, nous nous sommes demandé à quoi ressemblait le trafic des vélos entre les différentes stations. Quelles stations sont les plus prisées comme point de départ ou comme point d'arrivée ? Entre quelles stations y a-t-il le plus d'échanges de vélos ? Afin de répondre à ces questions, nous avons établi trois visualisations globales du réseau.

La première visualisation, sous la forme d'un réseau *Circos*, permet d'analyser la quantité des transferts entre les différentes stations. Ce type de graphe contient à la fois les informations concernant les entrées et les sorties. Nous sommes directement partis des données qui ont été nettoyées, comme expliqué dans la section 2.2, ce qui signifie que nous avons agrégé et représenté tous les déplacements des vélos sur un an mais d'une durée de plus de 60 secondes. Les différentes sections, qui correspondent à des noeuds et possèdent chacune sa propre couleur, représentent les stations tandis que les liens représentent les échanges de vélos entre les stations qu'ils relient. Notons que les liens portent la couleur du noeud à partir duquel ils partent et que nous n'avons visualisé que 25% des liens, ceux possédant les poids les plus importants, afin d'avoir une représentation plus facile à lire.

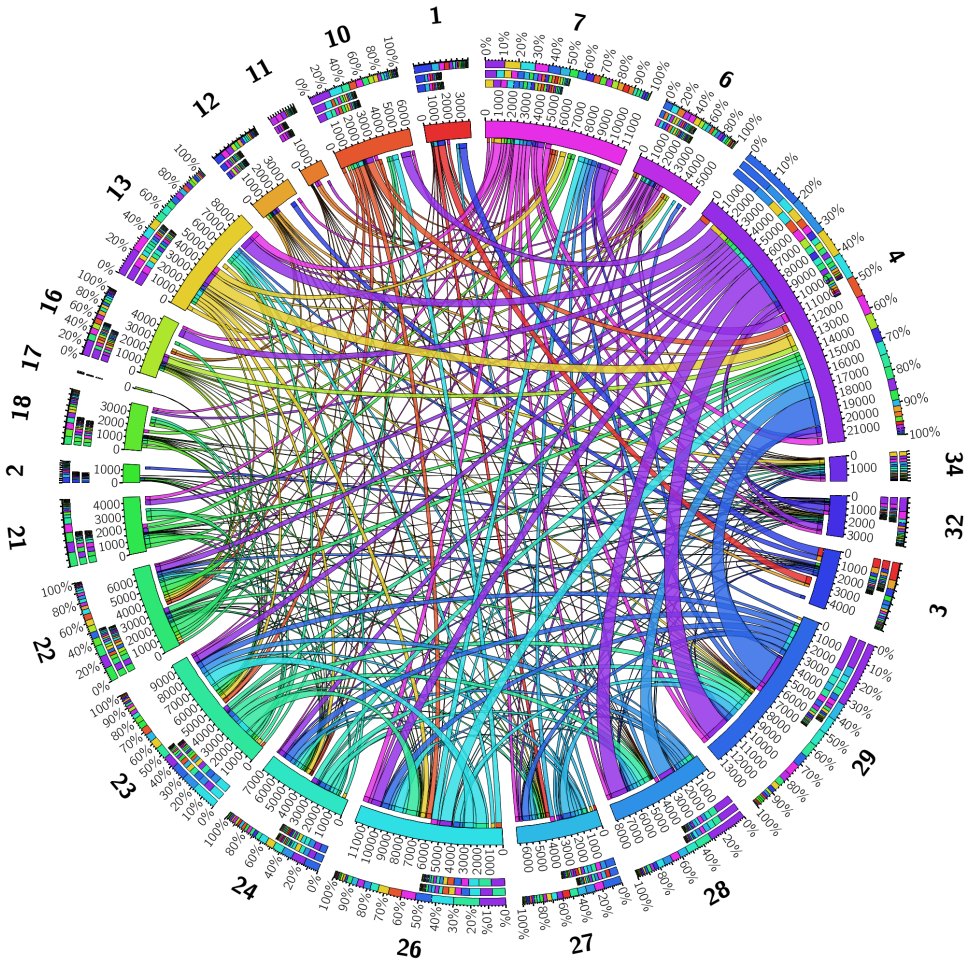


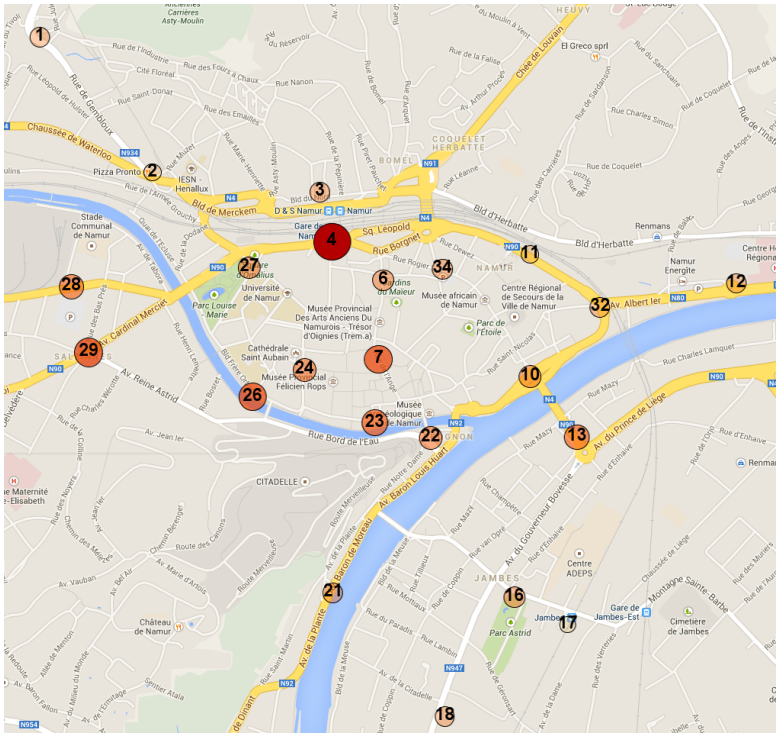
FIGURE 3.3 – Visualisation des liaisons entre les stations de vélos - construite à l'aide du site *Circos*

Ce type de visualisation se trouve sur la FIGURE 3.3. Nous pouvons voir que c'est la station 4, station située en face de la gare de Namur, qui possèdent la section la plus importante, c'est donc la station du réseau pour laquelle il y a le plus de transferts aussi bien entrants que sortants. Cela semble logique vu que la gare est un lieu constant de passages, reliant plusieurs modes de transports en commun différents (Trains et Bus). Nous allons nous concentrer sur cette station afin de décrire l'analyse de ce type de graphique. Comme nous l'avons dit auparavant, les liens portent la couleur de leur station de départ. Dans notre cas, la station 4 correspond à la couleur mauve, les liens correspondant aux sorties sont donc de couleur mauve et sont "collés" à l'arc mauve tandis que les liens représentant les entrées à cette station portent la couleur de leur station de départ et sont légèrement écartés de l'arc mauve. Nous pouvons voir que cet arc est gradué, cela permet d'évaluer la quantité de vélos qui ont transité par la station 4. Au total, 21 861 transferts de vélos sont passés par cette station. Ensuite nous avons trois arcs plus fins, qui sont quantifiés par des pourcentages. Le premier arc, se trouvant à l'intérieur, évalue le pourcentage des sorties tandis que le second évalue celui des entrées, le tout en fonction des différentes stations. Le dernier arc, à l'extérieur du graphe, évalue les transferts avec les autres stations, sorties et entrées confondues. Nous pouvons dire, par exemple, que la station avec laquelle la station 4 a le plus de transferts (20% des transferts) est la station 29, la couleur correspondant à celle-ci est un certain type de bleu.

Les deux autres types de visualisation représentent les stations sur une carte en fonction, pour la première, des retours des vélos et pour la seconde, des sorties. Ce type de visualisation nous permet d'avoir une information supplémentaire qui est la localisation spatiale des stations car celles-ci sont placées les unes par rapport aux autres grâce à leur latitude et leur longitude.

La FIGURE 3.4 est basée sur les retours de vélos. Les stations qui réceptionnent le plus de vélos correspondent aux noeuds les plus gros et les plus rouges tandis que les stations qui réceptionnent un nombre moindre de vélos sont représentées par les noeuds de taille plus petite dont la couleur tend vers le blanc. En règle générale, nous pouvons voir que les stations réceptionnant un nombre important de vélos sont les stations 4, 29, 26 et 27 tandis que celle de la gare de Jambes (17) en réceptionne moins au vu de la taille et de la couleur du noeud qui lui correspond. Nous pouvons remarquer que ces observations collent bien avec le graphe *Circos* (FIGURE 3.3), les stations 4, 29, 26 et 7 sont bien les stations pour lesquelles les proportions de rentrées de vélos sont les plus élevées tandis que la station 17 possèdent les proportions les plus faibles, aussi bien pour ses entrées que pour ses sorties. Notons que la station 4 a réceptionné 10 521 vélos tandis que la station 17 n'en a réceptionné que 80.

Le graphe présenté à la FIGURE 3.5 est quant à lui basé sur les sorties des vélos. Dans ce cas-ci, les noeuds les plus gros et les plus rouges correspondent aux stations pour lesquelles le nombre de vélos qui en sort est le plus important tandis que les noeuds les plus petits et blancs correspondent aux stations dont le nombre de vélos qui en sort est faible. Nous pouvons alors noter les mêmes observations qu'auparavant. Dans ce cas, la station 4 ressort 11 340 vélos tandis que la 17 en ressort 58. La station Gare de Namur reste donc parmi les stations principales du réseau.


Chaussée de Waterloo
Bld de Merckem
Bld d'Herbatte
Sq. Léopold
Rue Borgnet
Av. Cardinal Mercier
Av. Reine Astrid
Rue Bord de l'Eau
Av. Albert Ier
Av. du Prince de Liège
Av. Baron Louis Huart
Av. de la Plante
Av. Baron de Moreau
Rue de Gembloux
Chée de Louvain
Rue de Coquelet
Rue Charles Lamquet
Rue Mazy
Rue d'Enhaive
Av. du Gouverneur Bovesse
Montagne Sainte-Barbe
Rue des Viviers
Rue de Dinant
Rue de la Croix
Rue Saint-Nicolas
Rue Rogier
Rue Dewez
Rue Léanne
Rue de la Pagnerelle
Rue Pont d'Avroy
Rue de Bonel
Rue de la Fusée
Rue Nanon
Rue des Fours à Chaux
Rue Saint-Donat
Rue des Emaillets
Rue Leopold de Hulster
Rue de l'Industrie
Cité Floréal
Rue du Réservoir
Rue Asty-Moulin
Rue Marie-Henriette
Rue de la Dodaine
Rue de l'Armée Grouchy
Rue Muzet
Rue de Bas Prés
Rue Henri Lemaître
Rue Bouvier
Rue de la Croix
Rue de Tabora
Rue de l'Ecluse
Rue de la Plante
Rue de la Montagne
Rue Fumal
Rue Notre Dame
Rue Tillieux
Rue Monnaux
Rue de Coupin
Rue Lambin
Rue du Parabis
Rue de la Meuse
Rue Saint-Martin
Rue de la Citadelle
Rue de Cesnuet
Rue de la Chaine
Rue van Opre
Rue Chanoine
Rue d'Ernage
Rue des Nayers
Rue de Loyauté
Rue de Fer
Rue de la Colline
Rue de la Croix
Route des Canons
Route Merveilleuse
Chemin Berenger
Chemin des Marais
Av. Vaudan
Av. Sel Air
Av. Marie d'Arlois
Av. Jean Ier
Av. du Milieu du Monde
Av. de l'Ermitage
Av. Baron Fallon
Allée de Mercure
Sentier Atala
Av. de la Dame
Av. du Gouverneur Bovesse
Av. de la Citadelle
Av. du Prince de Liège
Rue d'Enhaive

Nous pouvons conclure que c'est la station Gare de Namur qui est la plus utilisée du réseau et ce, grâce aux différents lieux de départs et de passages des transports en commun situés à proximité de celle-ci. En effet, $14,69\%$ des vélos sortants sortent de cette station et $13,63\%$ des retours se dirigent vers cette station. Nous pouvons également remarquer que pour les derniers graphes, la station 15 n'a pas été prise en compte car celle-ci ne possède aucun lien entrant ou sortant, vu qu'elle est en cours de construction, et posait des problèmes au niveau de la visualisation des tailles des noeuds. En effet, nous n'avions pas beaucoup de différence entre les tailles des stations les plus importantes (stations rouges) car nous avions un écart trop important entre le nombre maximum et le nombre minimum de vélos échangés, dû au fait que la station 15 possède 0 liens avec les autres stations. Nous avons donc retiré cette station du graphe afin de réduire l'écart entre le maximum et le minimum de vélos échangés.

Afin de confirmer le faible changement entre les nombres de vélos sortants et entrants, nous avons tracé l'histogramme de ces nombres en fonction des stations (voir FIGURE 3.6). Grâce à cet histogramme nous pouvons affirmer que les taux de sorties et d'entrées de vélos sont équivalents pour la majorité des stations. Certes il y a des différences, mais celles-ci ne sont pas énormes. Afin de les voir plus en détails, nous avons tracé l'histogramme se situant à la FIGURE 3.7. Les barres qui sont dans la partie positive signifient que le nombre de vélos sortis de la station est supérieur à celui des vélos y entrant. Dès lors, nous pouvons voir que seules huit stations ont eu plus de sorties que de retours, ce qui représente un tiers du réseau. Nous remarquons également que les trois plus grandes différences sont également positives. Il y a donc un déséquilibre, certaines stations seront plus vite pleines ou vides que d'autres et vice-versa.

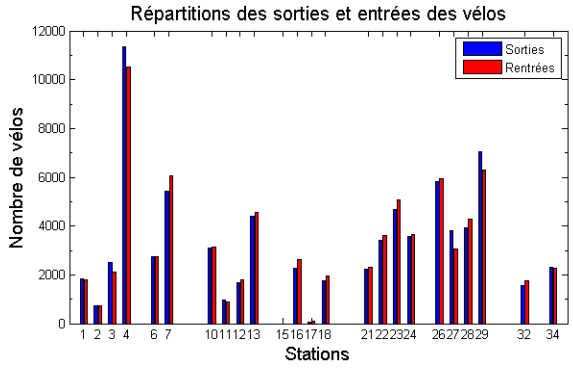


FIGURE 3.6 – Répartition des locations en fonction des stations pour les entrées et les sorties de vélos en fonction des stations - construite à l'aide de *Matlab*

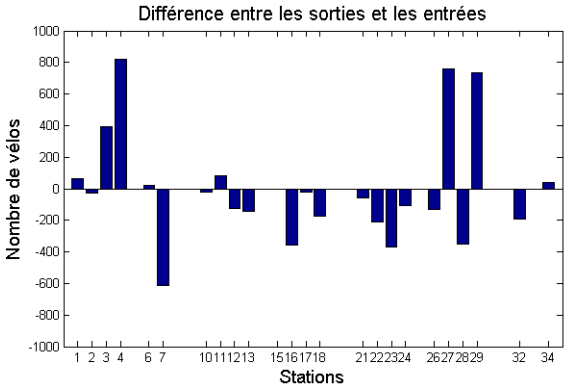


FIGURE 3.7 – Fluctuations entre le nombre de vélos entrants et sortants en fonction des stations : $(N_{sorties} - N_{entrees})$ - construite avec *Matlab*

À l'aide de la FIGURE 3.8, nous pouvons visualiser les différents liens entre les stations. Sur ce type de graphique, nous n'avons pas d'autre information concernant les noeuds que leur géolocalisation étant donné qu'ils ont tous la même taille, nous nous focalisons sur les liens, contrairement à avant. Afin d'y voir plus clair, seuls 20% des liens sont représentés, nous avons gardé uniquement les liens ayant les poids les plus importants. L'épaisseur des liens est donc proportionnelle à leur poids, plus il y a eu de transferts entre deux stations, plus le poids est important et plus le lien est épais. Notons également que nous avons un graphe qui est orienté, la couleur des liens correspond à la couleur de leur noeud de départ.

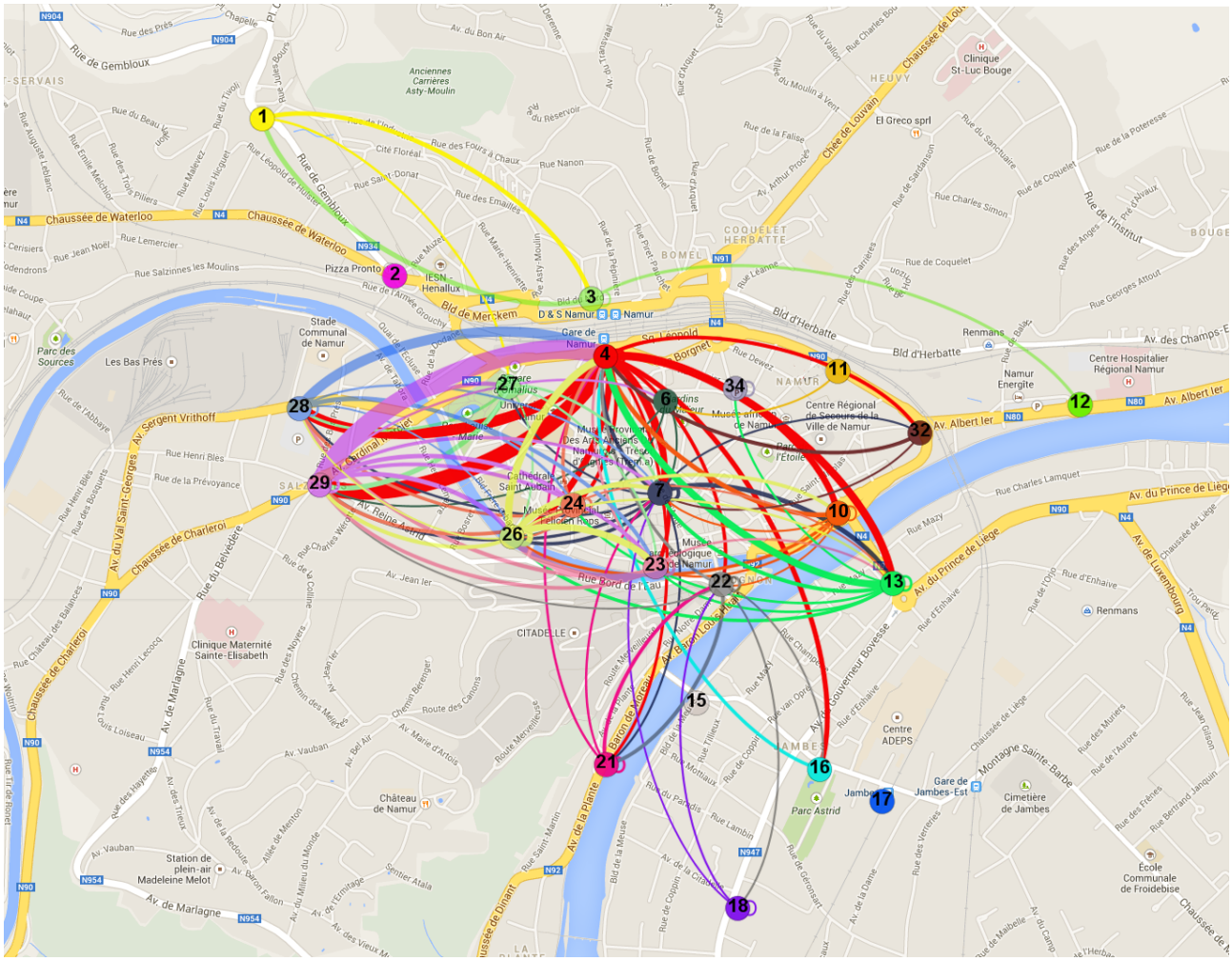


FIGURE 3.8 – Représentation des liens entre les stations - construite à l'aide des logiciels *Gephi* et *Gimp*

Dans le but d'avoir une visualisation encore plus claire, nous pouvons rassembler les sorties et les entrées. De cette manière, nous n'aurons plus qu'un seul lien qui relie les stations entre-elles, représentant le transfert total de vélos entre deux stations, aussi bien entrants que sortants. Cette représentation du réseau, en rassemblant les liens pour les sorties et les entrées entre chaque station, se trouve sur la FIGURE 3.9 tandis que le programme qui calcule les nouveaux poids des liens se trouve en annexe A.3. Nous pouvons voir que dans ce cas, nous avons une visualisation plus claire et plus lisible tout en sachant que désormais, 40% des liens sont visibles, nous avons donc plus d'informations qu'auparavant. Notons que les noeuds possèdent tous la même couleur étant donné que nous ne devons plus différencier l'arrivée et le départ des liens. La couleur des liens est également différente, avant elle ne dépendait que du noeud de départ tandis qu'ici elle dépend du poids du lien, plus le poids d'un lien est élevé, plus il est épais et de couleur foncée. Nous pouvons voir que l'épaisseur des liens est plus importante vu qu'ils regroupent à chaque fois deux liens, les entrées et les sorties, que l'on voyait séparément sur la figure précédente.

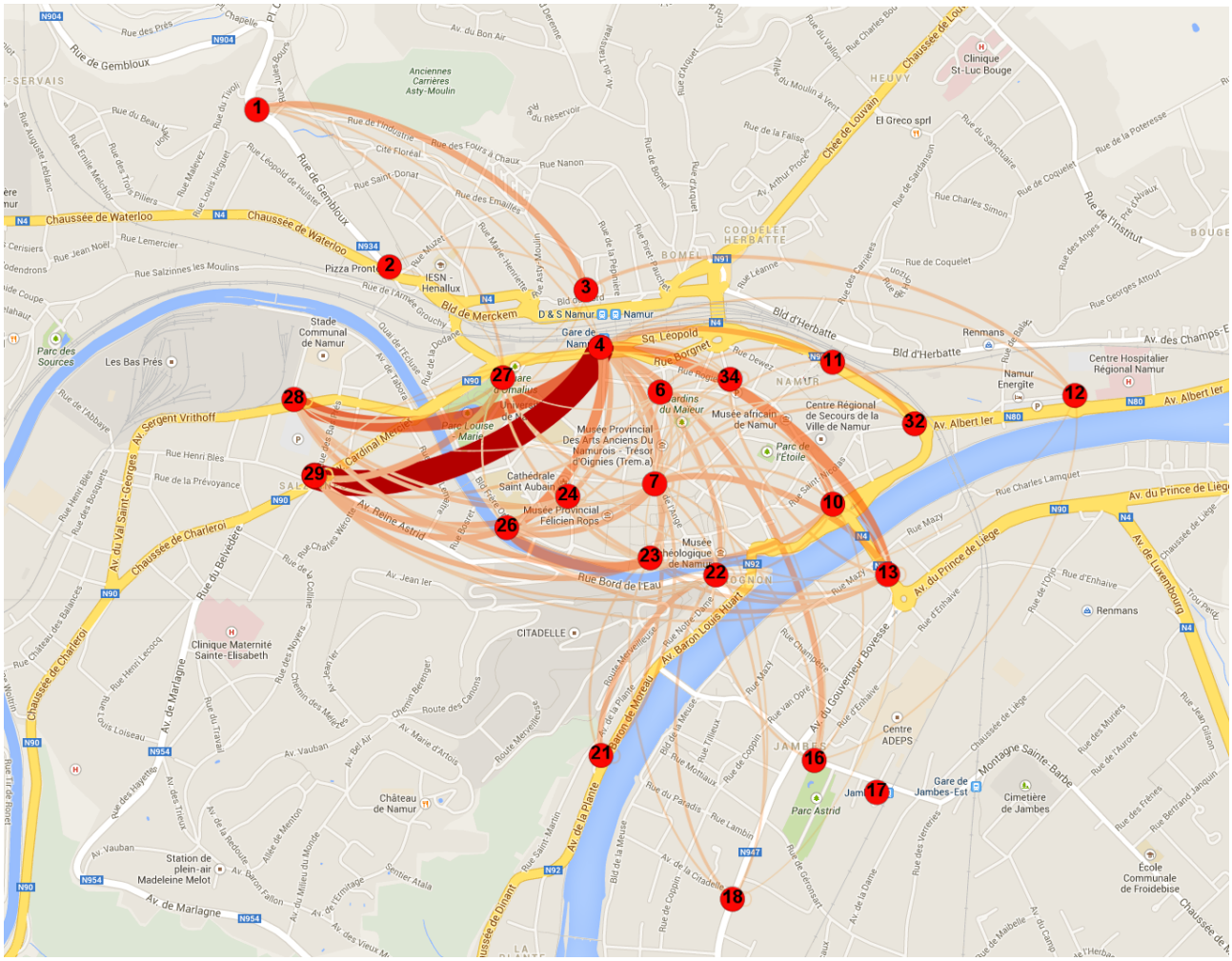


FIGURE 3.9 – Représentation des liens entre les stations avec le rassemblement des entrées et sorties - construite à l'aide des logiciels *Gephi* et *Gimp*

Grâce aux FIGURES 3.8 et 3.9, il est possible de clairement confirmer que la station 4, située en face de la gare, est un lieu essentiel du réseau. En effet, les liens les plus importants sont liés au noeud de cette station dont notamment les deux plus épais, qui relient la station 4 aux stations 28 et 29 situées à Salzinnes.

Par la suite, nous construirons ces mêmes visualisations mais pour une seule journée, de la semaine et du week-end, afin de voir, dans ces cas-là, quelles sont les stations les plus prisées. Nous construirons également ces graphes en fonction des heures, par intervalles, nous pourrons de cette manière, déterminer si la station Gare de Namur, par exemple, est fortement utilisée toute la journée ou uniquement le matin et en fin d'après-midi. Une condition importante est d'avoir des tranches de temps qui ne sont ni trop petites, ni trop grandes, pour contenir une quantité d'informations raisonnable et avoir une représentation claire et précise. Dans l'idéal, nous aurons besoin des heures de départ et d'arrivée des vélos ainsi que de leurs dates d'utilisation. Nous pourrons alors déterminer le nombre de vélos échangés entre deux stations toutes les heures ou chaque jour de la semaine.

3.2 Construction

La construction des graphes contenus dans la section précédente est plus complexe et c'est ici que l'ajout des numéros d'identification des stations dans le second fichier de données intervient (voir section 2.2).

Tout d'abord, nous sommes partis du fichier de données triées et nous avons conservé uniquement les colonnes contenant les numéros d'identification des stations d'entrée et de sortie liées à des locations de vélos. Dès lors, nous nous sommes retrouvés avec un fichier de deux colonnes, l'une étant le numéro de la station de sortie et l'autre étant le numéro de la station de retour.

Ensuite, nous avons établi un programme *Matlab* qui calcule, en trois phases, les poids des liens entre les différentes stations, c'est-à dire le nombre de vélos échangés entre deux stations. Ce programme se trouve en annexe A.2. La première phase consiste à construire une matrice de manière à ce que l'élément $a_{i,j}$, avec $i, j = 1, \ldots, 24$, corresponde au nombre de vélos sortis de la station i et rentrés à la station j, comme le montre la matrice suivante.

$$\begin{pmatrix} a_{1,1} & \cdots & a_{1,j} & \cdots & a_{1,24} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i,1} & \cdots & a_{i,j} & \cdots & a_{i,24} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{24,1} & \cdots & a_{24,j} & \cdots & a_{24,24} \end{pmatrix}$$

Nous avons alors calculé chacun des éléments de notre matrice à partir de la table à deux colonnes construite juste avant. Afin d'y voir plus clair, illustrons ces démarches par un exemple. Imaginons que notre table, composée de deux colonnes, soit la TABLE 3.1.

Num Sortie	Num entrée
1	4
3	2
4	3
1	2

TABLE 3.1 – Exemple de table

Par la construction choisie, nous obtenons la matrice suivante.

$$\begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

La seconde phase consiste à transformer notre matrice en une table composée de trois colonnes, correspondant respectivement au numéro d'identification de la station de sortie, au numéro d'identification de la station de retour et au nombre total de vélos qui ont voyagé de cette

station de sortie à cette station de retour, noté *weight* (TABLE 3.2).

Num Sortie	Num entrée	Weight
i	j	$w_{i,j}$

TABLE 3.2 – Forme de la seconde table

avec $i, j = 1, \ldots, 24$.

Lors de la troisième et dernière phase, nous avons supprimé les stations entre lesquelles zéro vélos ont circulé. Nous avons donc, comme table finale pour l'exemple choisi précédemment, la TABLE 3.3.

Num Sortie	Num entrée	Weight
1	2	1
1	4	1
3	2	1
4	3	1

TABLE 3.3 – Table finale de l'exemple

Nous avons sorti cette table sous le format `csv` afin de pouvoir la modifier et la mettre sous une forme correcte (forme expliquée dans la partie explicative sur *Gephi* dans la section 1.1) pour pouvoir l'utiliser dans le logiciel *Gephi*. Nous avons alors introduit cette table, dans le logiciel de représentation, comme table de liens entre les différents noeuds avec la première table contenant les informations sur les stations, transformée sous le format `csv`, comme table de noeuds.

Chapitre 4

Descriptions statistiques

Comme nous l'avons vu, il existe cinq types de sorties possibles pour les vélos. Grâce à la TABLE 4.1, nous pouvons voir la répartition des vélos en fonction de ces types de sorties. La FIGURE 4.1 illustre cette table sous la forme d'un histogramme, ce qui permet d'avoir une meilleure visualisation des proportions.

Type de trajet	Nombre de vélos	Pourcentage
Sortie pour location	78 955	87,82
Sortie pour maintenance	485	0,54
Sortie pour régulation	9 296	10,34
Sortie pour stockage (conforme)	606	0,67
Sortie réparation en atelier	563	0,63

TABLE 4.1 – Table de contingence en fonction du type de sorties possibles

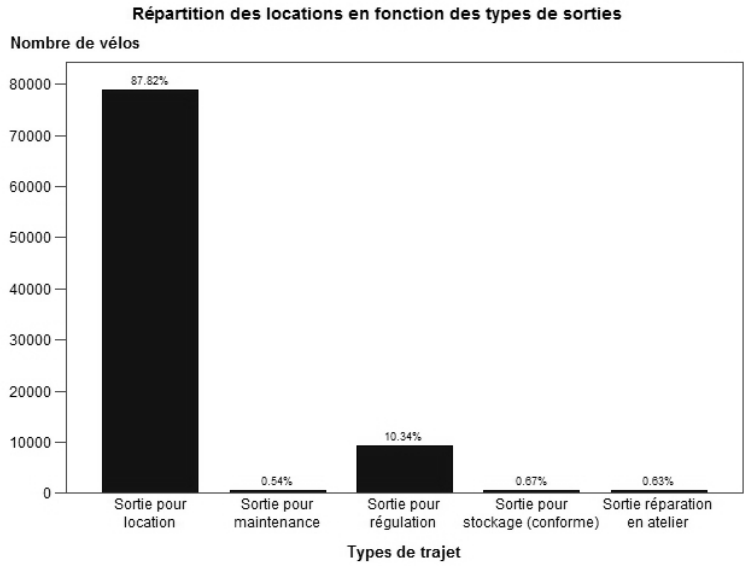


FIGURE 4.1 – Répartition des vélos en fonction du type de sorties - construite avec *SAS*

Notons que pour construire le graphe de la FIGURE 4.1, nous avons travaillé sur les données brutes. Nous remarquons évidemment qu'une grande partie des vélos est destinée à la location. En effet, 78 955 entrées dans la liste sur 89 916, ce qui correspond à $87,82\%$, sont des locations de vélos. Ensuite, nous pouvons voir que nous avons $10,34\%$ de sorties destinées à la régulation, 9 296 vélos ont donc été déplacés afin de remplir les stations vides et vider les stations pleines. Les autres pourcentages sont, quant à eux, minimes. Lorsque nous nettoyons nos données, comme expliqué dans la section 2.2 nous obtenons, sans grand étonnement 100% de sorties destinées à des locations.

4.1 Durées des locations

Grâce au programme de l'annexe A.1, nous avons calculé les temps de locations des données nettoyées qui correspondent à la différence entre chaque date-heure de sortie et de retour. Cela va alors nous permettre de calculer, par exemple, la moyenne du temps d'utilisation des vélos. Nous n'avons pas de location de moins de 60 secondes, vu que nous travaillons uniquement sur les données nettoyées, afin d'analyser uniquement les vélos qui ont été loués et non pas stockés, réparés ou autres.

Voici quelques résultats statistiques concernant les locations de vélos que nous avons déjà pu faire jusque maintenant.

- La moyenne des temps de locations des vélos est de 36 minutes et 54 secondes.
- La médiane des temps de locations des vélos est de 6 minutes et 50 secondes.
- Le temps maximal durant lequel a été loué un vélo est de 14 jours, 6 heures, 1 minutes et 43 secondes.

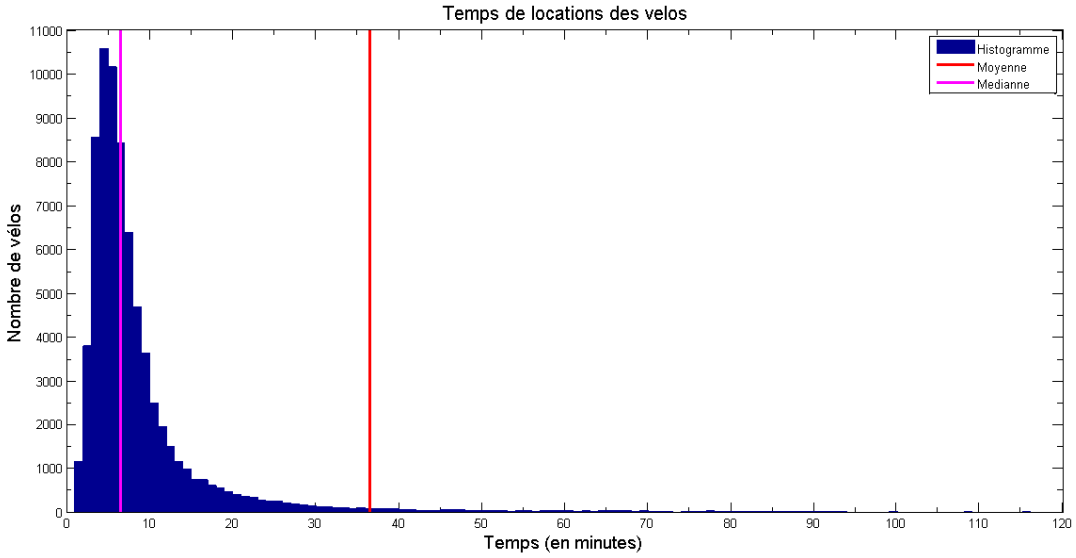


FIGURE 4.2 – Distribution des temps de locations des vélos en fonction des minutes - construite avec *Matlab*

Sur la FIGURE 4.2, fournie par le programme de l'annexe A.4, nous pouvons apercevoir la distribution des locations de vélos en fonction du temps, en minutes. Notons que pour faire l'histogramme, nous avons analysé les données en secondes, pour garder de l'information, mais que pour une meilleure compréhension nous avons modifié l'axe en minutes. Nous avons limité l'abscisse à 120 minutes pour une meilleure visualisation. Les données, dont les temps de location dépassent 120 minutes, représentent $2,45\%$ des locations, cela signifie que nous avons plus de 97% de données qui apparaissent sur le graphe, c'est un choix représentatif. En effet, la valeur maximale amène l'abscisse à aller jusque 1 231 303 secondes (14 jours, 6 heures, 1 minute et 43 secondes), ce qui nous donne un graphe illisible dont nous ne pouvons tirer aucune conclusion quant à l'utilisation des vélos si nous ne limitons pas l'abscisse.

Lorsque nous calculons le mode, qui correspond à la valeur, le temps de location dans ce cas-ci, qui apparait le plus, nous obtenons une valeur d'un peu plus de 4 minutes, ce qui correspond bien au graphique (FIGURE 4.2). En effet, nous pouvons voir que la barre la plus élevée est située à 4 minutes. Lorsque nous calculons la médiane, qui correspond à la valeur qui permet de couper l'ensemble des observations en deux parties égales, nous obtenons une valeur de plus ou moins 7 minutes. Cela signifie que la moitié des vélos ont été loués pour moins de 7 minutes, ce que nous pouvons très bien apercevoir sur le graphe. Pour finir, nous pouvons conclure que, vu la distribution asymétrique des temps de locations, la moyenne n'a pas vraiment de sens car celle-ci s'élève à 37 minutes, ce qui n'est pas du tout visible sur le graphe.

Lorsque nous comparons les histogrammes des FIGURES 4.2 et 1.5, nous pouvons voir qu'ils sont très semblables. Ils ont la même allure alors que les données correspondent à des villes différentes. Nous pouvons donc dire que les personnes adoptent le même comportement vis à vis du temps d'utilisation, la gratuité des 30 premières minutes influencent le comportement des consommateurs, qu'ils soient namurois ou lyonnais.

4.2 Utilisation du réseau au fil du temps

Afin d'avoir une idée globale de la manière dont est utilisé le réseau, nous avons décidé d'analyser nos données en fonction des données temporelles que nous possédons en différenciant les différents jours de locations. Pour ce faire, nous avons inséré une colonne qui nous informe du jour de la semaine, et non pas de la date exacte, durant lequel la location a été effectuée, comme expliqué dans la section 2.2. Les jours sont renseignés sous format numérique, 1 correspondant au lundi et 7 au dimanche. Nous savons également que *Matlab* convertit les heures en fractions d'une journée, ce qui nous a permis, avec les informations concernant le jour, de tracer différents graphes. Le programme permettant de construire ces types de graphiques se trouve en annexe A.5.

Pour construire la FIGURE 4.3, nous avons tout simplement séparé les données en deux groupes, un premier pour les jours du week-end, dont le jour était égal à 6 ou 7, et un second avec le reste des données pour les jours de la semaine. Dès lors, nous avons tracé les locations relatives à nos deux groupes en fonction des heures de location par intervalles de 15 minutes afin de ne pas avoir trop de fluctuations. Ce type de graphique permet de visualiser la densité des locations au fil des heures, nous ne nous sommes pas occupés des mois, nous avons agrégé toutes les données quelle que soit la date ou le mois afin de créer une journée "moyenne"

d'utilisation du réseau, une pour la semaine et une pour le week-end. Nous pouvons voir que les locations ne se déroulent pas de la même manière entre les jours de la semaine (du lundi au vendredi) et ceux du week-end (samedi et dimanche). En effet, pour la semaine nous avons trois pics d'utilisation visibles tandis que le week-end, nous n'en avons que deux. En ce qui concerne les trois pics d'utilisation, nous en avons un le matin, dans les alentours de 8h, un le midi, et le plus important le soir, dans les alentours de 18h. Ces pics collent donc parfaitement avec la vie des usagers : ils partent travailler le matin, vont manger le midi et retournent chez eux le soir. Pour ce qui est du week-end, nous pouvons voir que l'utilisation du réseau est tout à fait différente. En effet, la fréquence d'utilisation des vélos commence à augmenter plus tardivement dans la journée, comparée aux jours de semaine et le réseau est utilisé de manière plus importante l'après-midi plutôt que le matin. Pour confirmer ces différences, nous pouvons voir que le pic maximal pour un jour de semaine se situe dans les alentours de 18h tandis que pour le week-end, il se situe entre 12h et 16h. Notons également que durant la nuit, l'utilisation du week-end dépasse celle de la semaine, ce qui signifie qu'en soirée et durant la nuit, le réseau est sollicité un peu plus le week-end que la semaine.

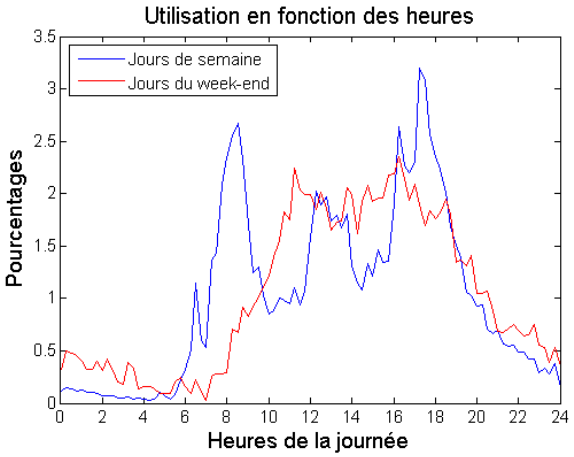


FIGURE 4.3 – Activité du réseau pour toutes les stations pour les jours de semaine et du week-end - construite avec *Matlab*

Après avoir étudié une journée moyenne, nous nous sommes intéressés à l'analyse de la semaine moyenne à l'aide des résultats repris dans la FIGURE 4.4. Pour obtenir cette figure, nous avons travaillé différemment. Ce graphe nous permet de visualiser la densité des locations au fil des heures et pour chaque jour de la semaine de manière séparée. En effet, afin d'avoir une différenciation entre les jours de la semaine, nous avons ajouté le "numéro du jour" -1 à notre fraction correspondant à l'heure afin de pouvoir faire commencer notre graphe à l'origine, comme nous le montre cette partie simplifiée du code correspondant situé en annexe A.5.2 :

```
n=nombre de locations
BOUCLE i=1:n
SI (Jour(i)==2) ALORS fraction_Journée(i)+=1
    SINON SI (Jour(i)==3) ALORS fraction_Journée(i)+=2
        ...
        END
    END
END
```

La première location, par exemple, a été effectuée à $14:37:47$ ($= 0,6096$ en fraction de jour) le 21/04/2012 ($= 6$ car c'était un samedi)., ous avons donc fait $0,6096 + (6-1)$ afin d'avoir une fraction de journée dépendante du jour de la semaine durant lequel a eu lieu la location. Mais encore une fois, nous avons agrégé toutes les données par jour de la semaine. Pour la partie du lundi par exemple, la courbe nous est donnée pour toutes les locations du fichier ayant eu lieu un lundi.

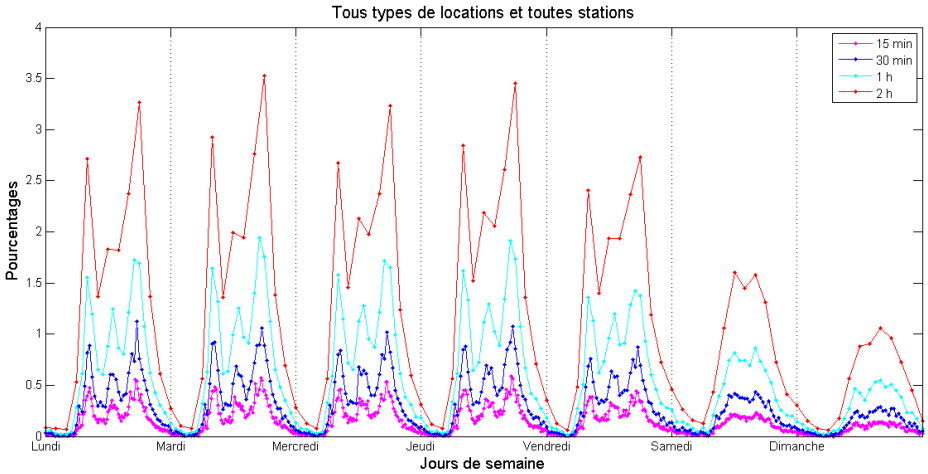


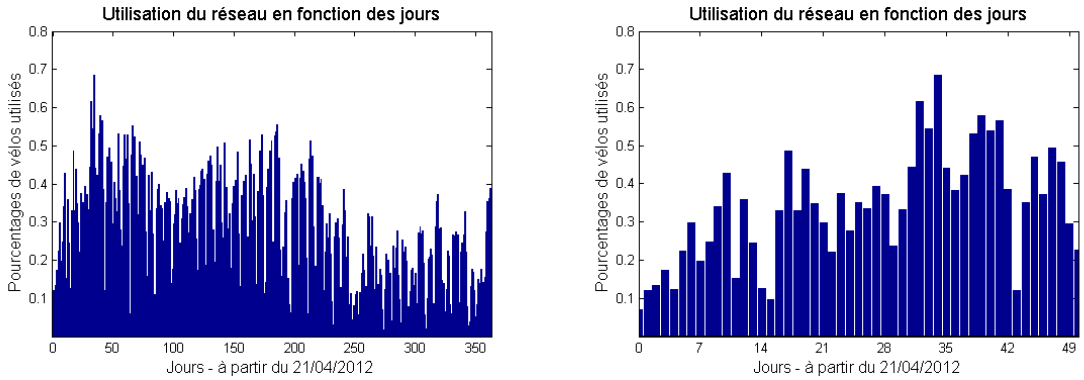
FIGURE 4.4 – Distribution des locations des vélos en fonction du temps - construit avec *Matlab*

Pour ce graphe, contrairement au précédent, quatre courbes ont été tracées, pour des intervalles de 15 minutes, 30 minutes, 1 heure et 2 heures. Pour ce faire, nous avons défini un vecteur de bins allant de 0 à 1 propre à chacun des intervalles étant donné que les heures sont enregistrées comme une fraction d'une journée. Pour l'intervalle de 15 minutes par exemple, qui a été choisi pour tracer les deux courbes du graphe précédent, nous avons établi le vecteur avec un pas de $\frac{1}{96} = \frac{1}{4\times 24}$ étant donné que prendre un intervalle de 15 minutes signifie que l'on divise les heures de la journée en 4. Nous avons choisi ces intervalles de temps afin d'avoir une échelle assez fine pour ne pas obtenir de grosses moyennes. Cependant, il ne faut pas choisir des intervalles trop petits pour ne pas avoir trop d'oscillations sur une journée. Sur la FIGURE 4.3 le choix d'intervalles de 15 minutes convient très bien, nous avons une bonne visualisation, mais pour ce graphique-ci, ce choix n'est pas pertinent, il y a trop d'oscillations. Avec l'intervalle de deux heures, nous pouvons voir que nous sommes dans le cas contraire, nous perdons le pic de midi, ce n'est pas assez précis, ce qui engendre une perte d'informations. Ceux d'une heure ou de 30 minutes semblent être les plus intéressants car ils nous permettent d'avoir une idée des densités de locations claire et suffisamment représentative. Lorsque nous analysons la FIGURE 4.4, nous pouvons remarquer que, quel que soit le type d'intervalle, les locations ne se déroulent pas de la même manière entre les jours de la semaine (du lundi au vendredi) et ceux du week-end (samedi et dimanche), ce qui confirme les résultats de la figure précédente. En effet, en semaine nous avons toujours les trois pics d'utilisation visibles tandis que le week-end, il n'y en a que deux.

Lorsque nous comparons les FIGURES 4.3 et 4.4 aux graphes de la FIGURE 1.6 pour la ville de Barcelone, nous pouvons voir que les allures des courbes sont similaires. Nous avons également trois pics d'utilisation pour les jours de la semaine et deux plus faibles pour le week-

end. Comme pour ce qui avait dit lors de la comparaison avec la ville de Lyon auparavant, les personnes qui utilisent ce système de vélos ont tendance à l'utiliser de la même manière, qu'ils soient belges, espagnols ou français.

Finalement, nous allons étudier les données sur toute la période d'un an. Ce qui signifie que nous allons nous intéresser à l'utilisation du réseau en fonction des différentes dates que nous avons dans le fichier. Entre la première location, le 21 avril 2012 et la dernière, le 18 avril 2013, il y a 363 jours qui se sont écoulés. Contrairement à l'analyse précédente, nous différencions chaque jour et nous ne regroupons pas les données en fonction du jour de la semaine ou autre. Ici, chaque date est différente et nous allons garder cette information.



(a) Histrogramme (b) Zoom

FIGURE 4.5 – Distribution de l'utilisation du réseau pour chaque date contenue dans le fichier - construites avec *Matlab*

Dans un premier temps, nous avons décidé de tracer l'histogramme du nombre de vélos utilisés en fonction des dates que nous avons. Pour ce faire, nous avons comparé chaque date-heure de location à la première date-heure en soustrayant leur *serial date number* comme nous le montre le pseudo code suivant :

```
n = nombre de locations
% Numero de serie de la premiere date
serie(1)=datenum(dateheure_location(1),'dd/mm/yy HH:MM:SS')
BOUCLE i=2:n
   serie(i)=datenum(dateheure_location(i),'dd/mm/yy HH:MM:SS')
   %Difference par rapport au premier
   Diff(i-1)=serie(i)-serie(1)
END
```

Cette soustraction nous fournit un nombre décimal qui représente l'écart entre les deux données. Si notre première location possède la série numérique 5000,5 et la seconde le série 5003 par exemple, la différence entre les deux nous donne 2,5 ce qui signifie qu'il y a 2 jours et 12h de différence entre les deux locations. Cela nous permet, encore une fois, de faire commencer notre graphe à l'origine.

Lorsque nous analysons la FIGURE 4.5(a) et que nous zoomons sur le début de l'histogramme (FIGURE 4.5(b)), nous pouvons clairement voir une différence d'utilisation entre les jours du

week-end et ceux de la semaine. En effet, l'abscisse en 0 correspond au samedi 21/04/2012, donc chaque multiple de 7 correspond à un samedi et chaque abscisse qui suit les précédentes correspond à un dimanche. Chacune des barres se situant à ces endroits diminue par rapport aux cinq barres précédentes, qui correspondent à des jours de la semaine. Le programme qui construit ces figures se trouve en annexe A.7.

En revenant à la FIGURE 4.5(a), nous pouvons voir que le réseau a été très vite utilisé de manière régulière mais qu'après le 244ème jour, qui correspond au vendredi 21 décembre 2012, le réseau a été moins sollicité. C'est grâce au zoom se situant à la FIGURE 4.6 et quelques recherches que nous avons pu définir la cause de cette baisse d'utilisation. Nous pouvons voir sur la figure que, comparé aux autres jours présents sur celle-ci, du samedi 22 décembre 2012 au dimanche 6 janvier 2013 le réseau a été beaucoup moins utilisé et ce, en particulier pour le 25 décembre et le 1 janvier. La raison de cette baisse est sans nul doute que ces dates correspondent exactement aux vacances de Noël de l'année 2012 – 2013 et que le 25 et le 1 sont des jours fériés. L'utilisation du réseau est donc dépendante des périodes de vacances car après le 6 janvier, nous pouvons voir que la fréquence d'utilisation du réseau redevient similaire à l'utilisation qui précédait les vacances, élevée en semaine et assez faible le week-end. Après quelques recherches, nous pouvons également affirmer que la météo, durant ces vacances, a été assez pluvieuse ce qui a donc fait diminuer le nombre de locations : "*Les moyennes régionales des précipitations mensuelles furent toutes supérieures aux valeurs normales*", [16]. Notons tout de même que l'utilisation durant la seconde semaine des vacances (du 31 au 6) est légèrement plus élevée que celle de la première semaine, débutant le 24. Cette réflexion nous permet de discerner également les vacances d'été qui, sur la FIGURE 4.5(a), se trouvent entre les jours 70 et 130. Nous pouvons remarquer que le réseau a été légèrement moins utilisé durant cette période sans pour autant avoir une baisse aussi importante que celle qui a eu lieu pendant les vacances de Noël. Nous pouvons donc supposer que les saisons influencent les utilisateurs du réseau. Nous referons cette analyse à l'aide d'une distribution cumulative afin d'avoir une vision plus claire de la diminution d'utilisation du 21 décembre 2012.

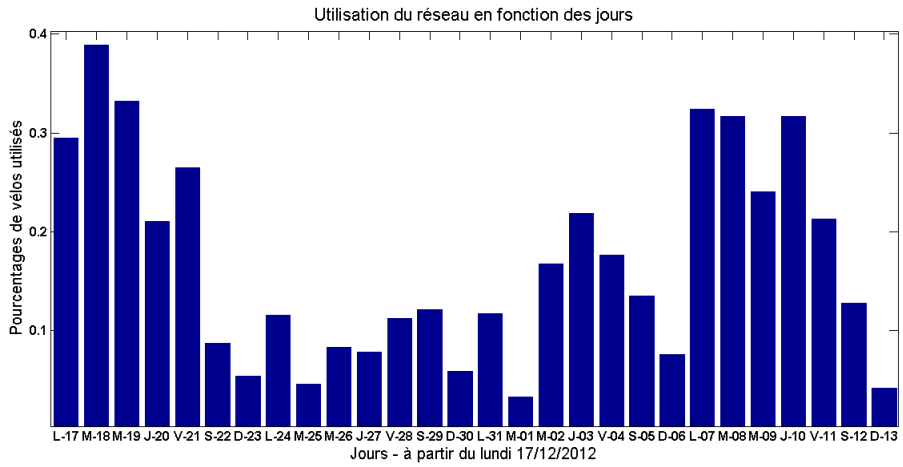
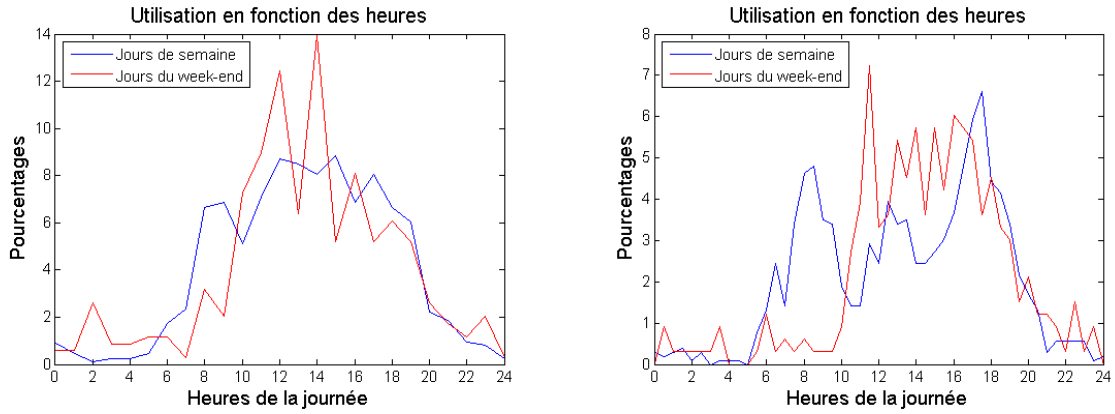


FIGURE 4.6 – Zoom 244ème jour - construite avec *Matlab*

Dans le but d'analyser l'utilisation du réseau lors des vacances de Noël 2012, comme décrit dans le paragraphe précédent, nous avons sélectionné les données dont les dates de locations

étaient situées entre le 24 décembre 2012 et le 6 janvier 2013 afin de tracer le graphe représentant l'activité du réseau pour toutes les stations pour les jours de semaine et du week-end sélectionnés. Pour ce faire, nous avons d'abord créé un programme qui sélectionnait les données souhaitées en fonction de deux dates. Ce programme se trouve dans l'annexe A.8 et permet également de sélectionner les données en fonction des heures ou du jour, ce qui nous servira par la suite. Après avoir trié nos données, nous avons donc tracé la FIGURE 4.7(a). Nous pouvons voir que ce graphe n'est pas similaire à celui de la FIGURE 4.3, qui avait été fait pour toutes les données agrégées sur une journée moyenne. Dans ce cas, nous n'avons pas les trois pics d'utilisation du matin, du midi et du soir pour les jours de semaine. Nous pouvons constater qu'il n'y a pas de différence importante entre les locations qui ont lieu un jour de semaine ou le week-end, les comportements en semaine se sont donc convertis en comportements typiques du week-end.

Nous avons sélectionné les données pour les autres vacances et avons tracé le même graphe mais dans ces cas, l'utilisation en semaine reste différente de celle du week-end, les trois pics sont présents, comme nous pouvons le voir sur la FIGURE 4.7(b) pour les vacances de Pâques. Il y a certes une légère diminution d'utilisation du réseau durant ces périodes mais la différence entre les jours de semaine et du week-end reste visible. Nous pouvons donc conclure que les vacances ou les saisons influencent l'intensité avec laquelle est utilisé le réseau mais qu'il n'y a que pour les vacances de Noël que le réseau a été utilisé différemment. Pour les autres périodes de vacances, les différences entre la semaine et le week-end sont moins importantes mais restent tout de même présentes.



(a) Vacances de Noël (b) Vacances de Pâques

FIGURE 4.7 – Utilisation du réseau pendant les vacances - construites avec *Matlab*

Traçons maintenant la courbe cumulative correspondant à la FIGURE 4.5(a) afin d'avoir une idée de l'évolution des locations. La "dérivée" de la courbe cumulative nous donne le nombre de vélos loués à chaque moment et en utilisant ce type de courbe, l'importance des petites fluctuations présentes dans les données est réduite, ce qui permet d'analyser la tendance globale. Lorsque nous traçons la courbe cumulative (voir FIGURE 4.8) nous pouvons voir, comme dit précédemment, qu'aux alentours du $230^{\text{ème}}$ jour la courbe a tendance à avoir une pente moins élevée qu'au début. Afin de confirmer cette hypothèse, nous avons approximé cette courbe avec deux fonctions linéaires, avant et après ce jour, à l'aide de la fonction `polyfit` disponible par *Matlab*. Les droites représentant ces approximations sont tracées sur la même figure à l'aide de

traits et nous pouvons confirmer que la pente n'est pas la même suivant la période considérée. Ces approximations nous permettent d'avoir le taux moyen de vélos loués par jour qui est de 265 vélos par jour pour les 230 premiers jours, c'est à dire pour les huit premiers mois, et de 130 vélos par jour pour les quatre derniers mois. Nous pouvons constater que le taux de locations est divisé par deux après les huit premiers mois.

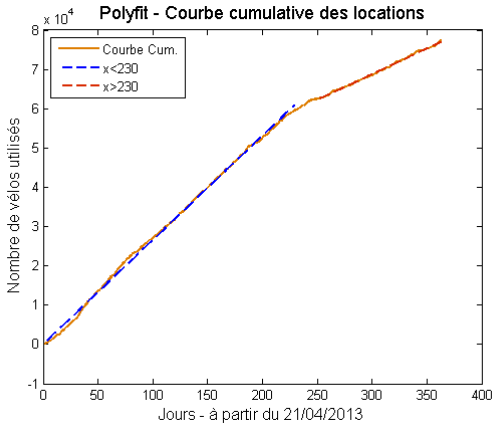


FIGURE 4.8 – Courbe cumulative des locations de vélos - construite avec *Matlab*

Grâce aux taux moyens de locations obtenus, nous avons décidé d'établir une représentation des fluctuations des locations en fonction des jours ainsi que l'histogramme correspondant dans le but de voir s'il existe des phénomènes "rares" correspondant à des fluctuations importantes. Pour ce faire, nous avons soustrait les taux moyens des locations aux taux réels pour les deux périodes séparément afin de voir les différences entre les jours. La FIGURE 4.9 représente ces fluctuations et nous pouvons voir que celles-ci sont plus importantes pour la première période que pour la seconde.

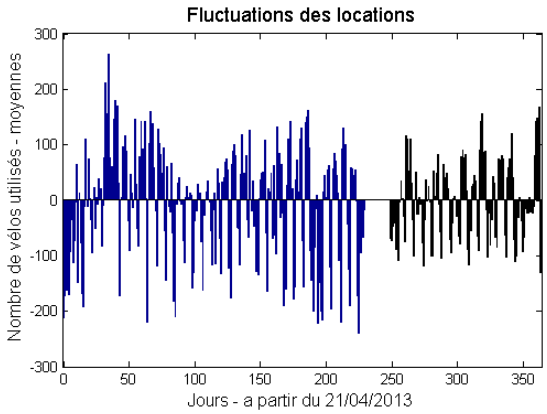


FIGURE 4.9 – Fluctuations des quantités de locations - construite avec *Matlab*

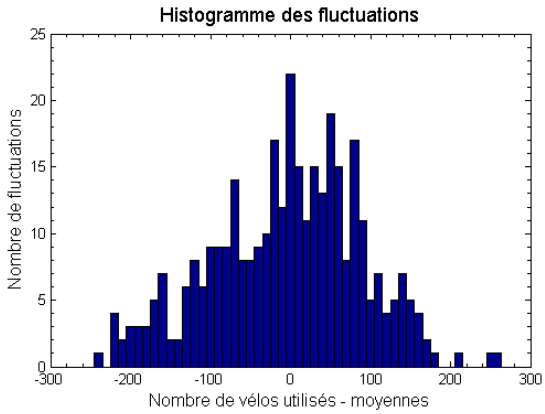


FIGURE 4.10 – Histogramme des fluctuations des locations - construite avec *Matlab*

L'histogramme des fluctuations, situé sur la FIGURE 4.10, rassemble les données des deux périodes concernées auparavant. Nous pouvons voir que nous avons des barres plus élevées dans les fluctuations positives mais qu'il y en a plus, même si elles sont plus petites, dans les négatifs.

Lorsque nous cherchons le jour où la différence entre la fréquence de locations et le taux moyen est maximale positive dans la première période, nous tombons sur le jour 34, qui correspond au 25 mai 2012, avec une variation de 263 vélos loués en plus de la moyenne journalière, ce qui signifie que par rapport aux autres jours, deux fois plus de vélos ont été loués. Cette date tombe un vendredi durant le week-end de la Pentecôte et a bénéficié d'un temps totalement estival, ce qui peut en partie expliquer cette différence importante par rapport à la moyenne journalière de la première période. Dans la seconde période, c'est le 17 avril 2013 qui possède la variation positive maximale. Nous supposons que cela peut être influencé par la météo : "*Après quelques jours à nouveau plus mitigés, nous connaîtrons à nouveau une journée aux relents printaniers le 17 avril, avant un nouveau refroidissement jusqu'au 22 avril.*, [17]. Les minimas quant à eux tombent le 2 décembre 2012 et le 18 avril 2013, en notant que pour ce dernier, nous n'avons des données allant que jusque 23h50. Le programme permettant de tracer les derniers graphiques de cette section est à l'annexe A.7.

4.3 Utilisation du réseau en fonction des abonnements

Comme expliqué dans la section 2.1, le fichier de données contient une information concernant le type d'abonnement de l'utilisateur, nous pouvons donc nous demander si l'utilisation du réseau est différente suivant le type d'abonnement que l'utilisateur détient. Afin de trouver des réponses, nous avons trié et stocké nos données dans différents fichiers suivant les différents abonnements possibles, à l'aide de la fonction de situant à l'annexe A.6.2, et nous avons tracé des graphes similaires aux FIGURES 4.4, 4.5(a) et 4.8. Rappelons que nous ne prenons en compte que les abonnements des utilisateurs et non des administrateurs vu que nous travaillons sur les données qui ont été nettoyées et qui ne contiennent plus que des locations.

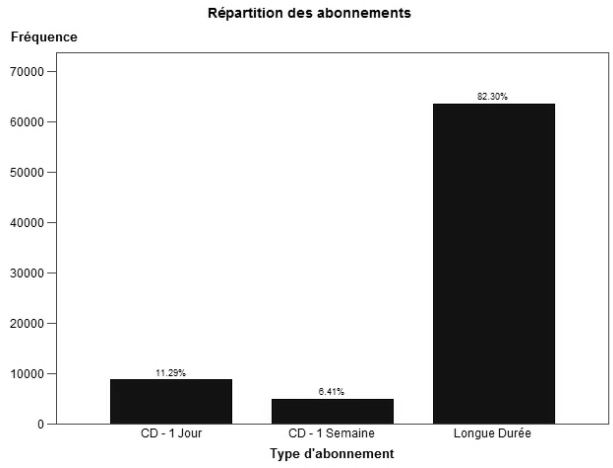


FIGURE 4.11 – Répartition des locations en fonction de l'abonnement - construite avec *SAS*

Dans un premier temps, nous avons tracé un histogramme qui permet de se faire une idée de la répartition des différents abonnements dans nos données. Cet histogramme se trouve sur la FIGURE 4.11 et nous pouvons voir que l'abonnement le plus utilisé est l'abonnement annuel, couvrant 82% des locations. Moins cher et gratuit durant les six premiers mois lors de la mise en service du réseau, cet abonnement est idéal pour des utilisations régulières du réseau, il est donc logique qu'il soit présent en majorité dans nos données.

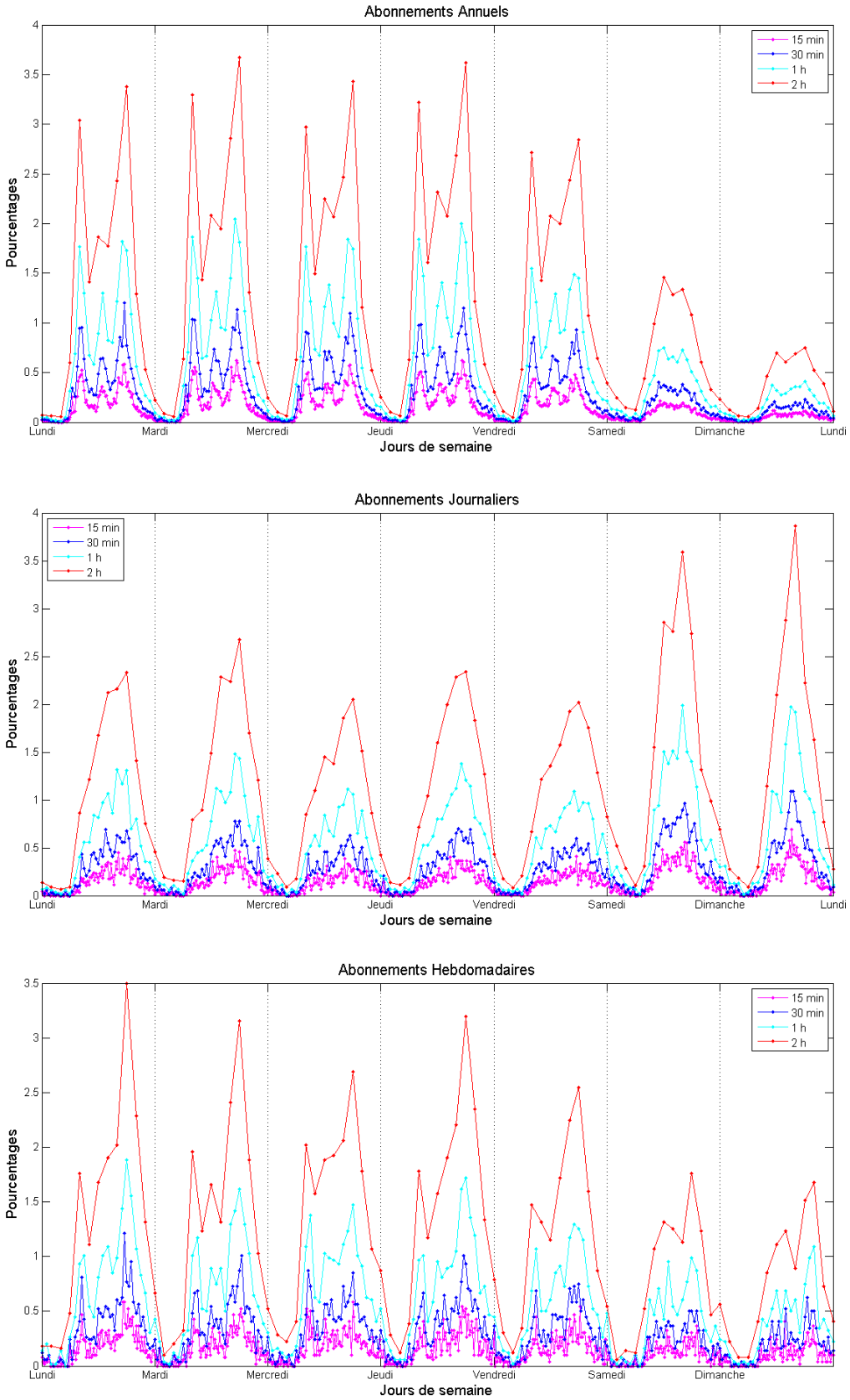
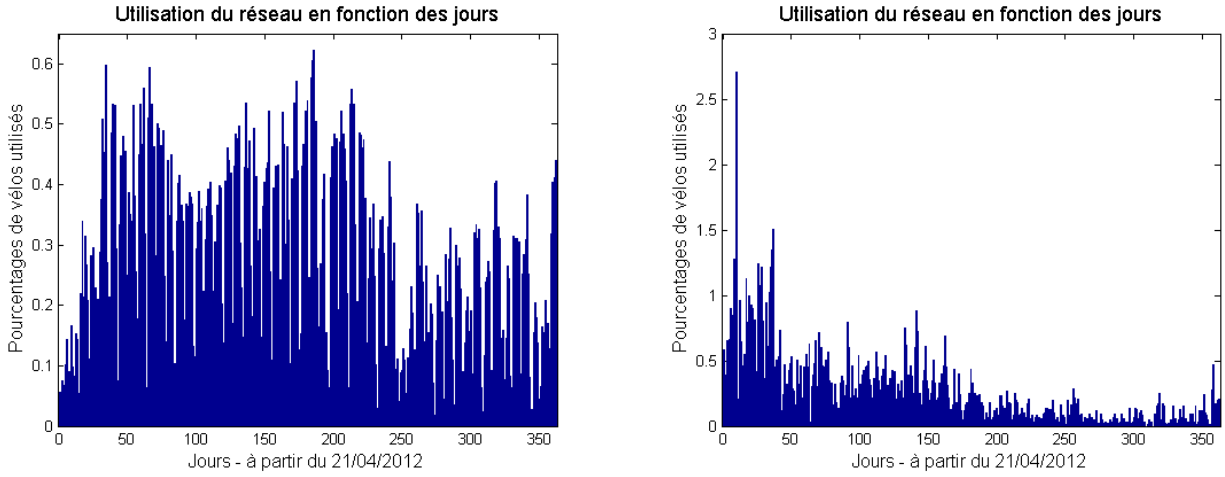


FIGURE 4.12 – Distribution des locations des vélos en fonction du temps pour les différents types d'abonnement - construites avec *Matlab*

Ensuite, nous nous sommes penchés sur l'analyse des données agrégées sur une semaine avec les graphes de la FIGURE 4.12. Ces graphiques nous permettent de différencier l'utilisation du réseau par rapport aux abonnements. En effet, nous voyons très clairement que les courbes correspondant aux abonnements de longue durée ou d'un jour ont des allures complètement opposées ; le pourcentage d'utilisation pour les abonnements journaliers est plus élevé le week-end alors que c'est en semaine qu'il est le plus élevé pour les abonnements annuels. Nous pouvons également remarquer que, pour les abonnements d'une journée, les trois pics d'utilisation ne sont pas présents alors que sur les deux autres graphiques ils y sont. Lorsque nous analysons le graphique des locations hebdomadaires, nous pouvons voir que celui-ci se comporte un peu comme celui des locations annuelles mis à part le fait que les pics pour le week-end et les soirées sont plus imposants comparés aux autres pics de la journée. Nous pouvons donc considérer que les utilisateurs possédant un abonnement d'une semaine ont tendance à utiliser le réseau le soir, plutôt qu'en journée, et les week-ends tandis que les utilisateurs possédant un abonnement annuel vont principalement utiliser le réseau en semaine. Étant donné cette similitude assez proche entre les abonnements hebdomadaires et annuels, pour la suite nous n'analyserons plus les abonnements d'une semaine pour ne pas alourdir la lecture étant donné que les résultats seront similaires.

Après ces différentes observations, nous nous sommes demandés quelle allure pouvait avoir les histogrammes, non plus en fonction des jours de la semaine mais en fonction des dates lorsque l'on sépare les données en fonction du type d'abonnement que les utilisateurs possèdent. Les histogrammes de la FIGURE 4.13 représentent les abonnements annuels et journaliers, nous pouvons voir qu'ils ne sont pas du tout semblables.



(a) Abonnements annuels

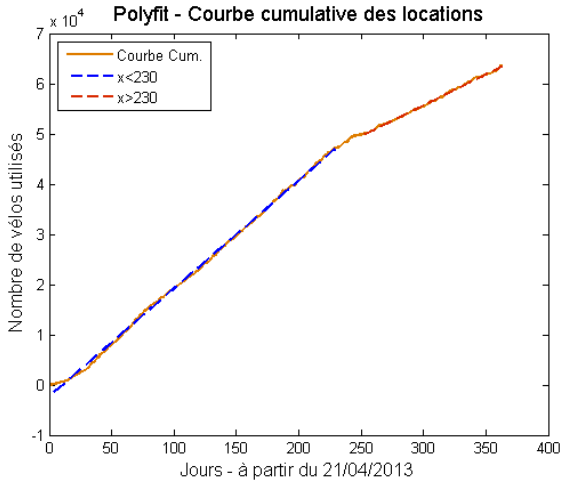
(b) Abonnement d'un jour

FIGURE 4.13 – Distribution de l'utilisation du réseau en fonction des abonnements - construites avec *Matlab*

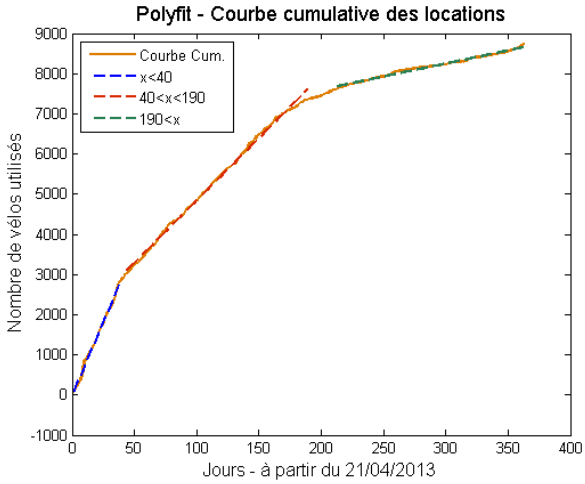
Pour les abonnements annuels, sur la FIGURE 4.13(a), les creux liés aux vacances sont plus prononcés tandis que pour les abonnements journaliers, sur la FIGURE 4.13(b), nous n'avons pas de baisse d'utilisation particulière liée aux vacances mais bien un pic d'utilisation qui a eu lieu le $10^{\text{ème}}$ jour, c'est à dire le 1^{er} mai, qui, en 2012, a bénéficié d'un temps plutôt clément. Nous pouvons remarquer que pour les abonnements annuels, un peu après le $230^{\text{ème}}$ jour, le nombre

journaliers de vélos loués a tendance à être inférieure à ceux des jours précédents, comme nous avons pu le remarquer dans la section précédente. En ce qui concerne les abonnements journaliers, nous remarquons qu'au fil des jours, le nombre d'utilisations décroit, ce qui signifie que de moins en moins de personnes utilisent ce type d'abonnement. Mise à part cette diminution, nous ne savons rien discerner d'autre, nous allons donc tracer les courbes cumulatives afin d'en savoir un peu plus sur l'évolution des quantités de locations.

Lorsque nous traçons les courbes cumulatives liées aux histogrammes précédents, nous pouvons voir que les deux courbes ne se comportent pas du tout de la même manière. Sur la FIGURE 4.14(a), nous pouvons voir la courbe cumulative des abonnements annuels qui, sans grand étonnement, ressemble à celle que nous avons analysé dans la section précédente. Cependant, nous pouvons voir qu'au début de la courbe, la croissance n'est pas similaire à celle observée jusqu'ici, elle est sub-linéaire, ce qui signifie que lors des premiers jours, il n'y avait pas beaucoup d'abonnés de 1 an. Cela peut signifier que les utilisateurs ont testé le service avant de prendre un abonnement annuel. Passons maintenant à la FIGURE 4.14(b), liée aux abonnements d'un jour. Nous pouvons voir que dans ce cas, la courbe peut être séparée en trois parties : une pour les 40 premiers jours (jusqu'au 31/05/2012), une pour les jours se situant entre le 31 mai et le 28 octobre ($190^{\text{ème}}$ jour) et une pour les derniers jours. Les taux moyens de location associés aux pentes sont respectivement de 73, 30 et 6, ce qui implique que le succès des locations avec des abonnements d'un jour a fortement diminué depuis la mise en place du réseau. De fait, celui-ci a été divisé par plus de 10 tandis que celui des locations via l'abonnement annuel n'a été divisé que de 2.



(a) Abonnements annuels



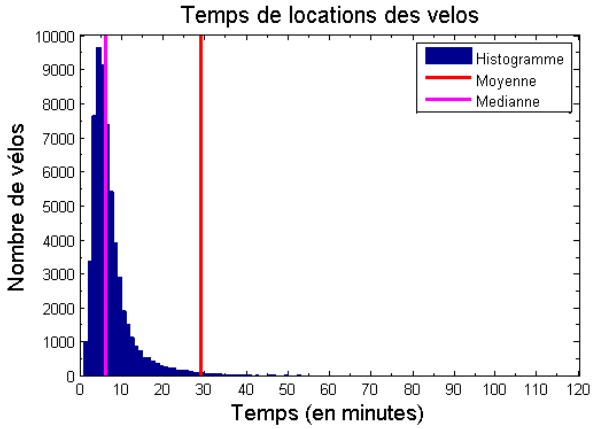
(b) Abonnement d'un jour

FIGURE 4.14 – Courbes cumulatives des locations de vélos - construites avec *Matlab*

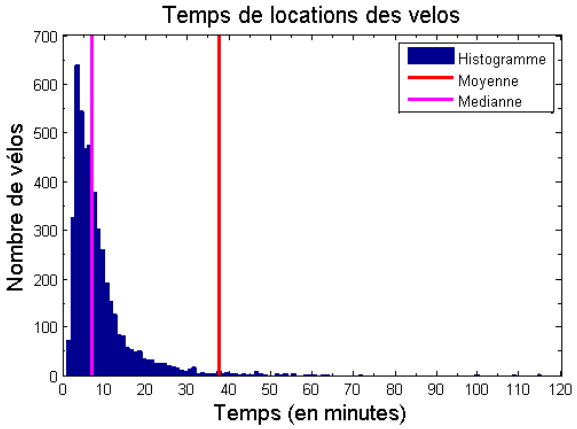
Après ces différents résultats et sans grand étonnement, nous pouvons affirmer que le type d'abonnement que l'utilisateur possède influence considérablement la manière dont il va utiliser le réseau. Les personnes possédant des abonnements annuels ont tendance à se déplacer à l'aide des vélos durant la semaine et aux trois moments principaux de la journée, matin midi et début de soirée, tandis que les détenteurs d'un abonnement d'une journée utilisent les vélos de préférence le week-end et ce à partir de midi. Nous pouvons également constater qu'en règle générale, le taux d'utilisation des vélos diminue au fil du temps et en particulier pour les abon-

nements journaliers.

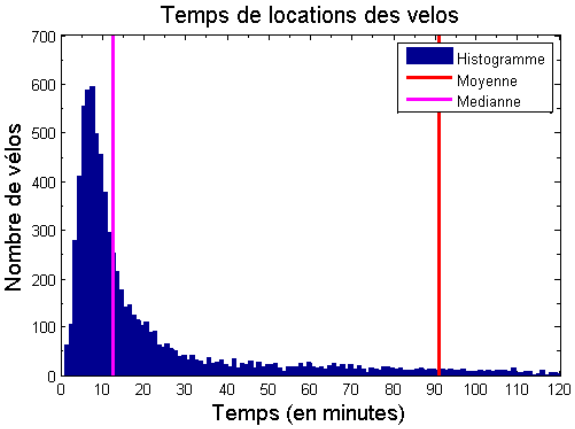
Cependant, le fait de posséder un abonnement annuel, hebdomadaire ou journalier n'influence pas les durées des locations. En effet, comme nous pouvons le voir sur la FIGURE 4.15, les graphes liés aux trois types d'abonnements sont similaires. La médiane ne varie que très légèrement et nous observons le même profil pour les trois graphes, les locations durent en grande partie moins de 30 minutes, ce qui nous montre encore une fois l'impact de la gratuité de ces dernières.



(a) Abonnements annuels



(b) Abonnement hebdomadaire



(c) Abonnement d'un jour

FIGURE 4.15 – Histogramme des durées des locations en fonctions des trois abonnements - construites avec *Matlab*

4.4 Utilisation du réseau en fonction des stations

Afin d'observer le comportement des utilisateurs en fonction des différentes stations qui composent le réseau, nous avons tracé les courbes représentant le pourcentage de vélos utilisés en fonction des heures pour chaque station séparément afin de comparer l'utilisation de chaque station en fonction de l'heure de la journée et du jour de la semaine. Pour faire cela, nous avons dû, dans un premier temps, trier les données en fonction des stations de sortie, ou d'entrée, à l'aide du programme qui se trouve en annexe A.9. Celui-ci permet donc de construire deux fichiers pour chaque station, l'un contenant les données liées aux vélos qui sont sortis de cette station et l'autre contenant les données des vélos qui sont rentrés à cette station après leur trajet. Ensuite, nous avons créé le programme se trouvant en annexe A.10, qui permet de tracer les figures désirées en fonction des sorties, celui pour les entrées est presque identique, nous ne l'avons donc pas mis en annexe.

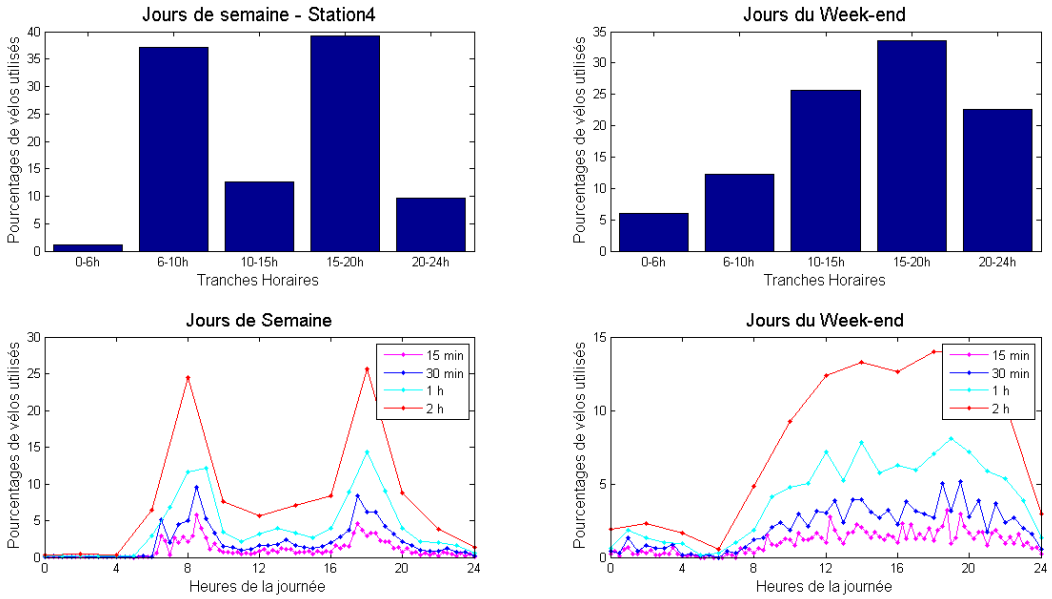


FIGURE 4.16 – Distribution des locations des vélos pour les sorties de la station 4 - construit avec *Matlab*

La FIGURE 4.16 représente deux types de visualisation pour les jours de la semaine et pour ceux du week-end. La première, avec les histogrammes, représente les pourcentages de vélos sortant de la station choisie pour cinq tranches horaires recouvrant une journée de 24h. La seconde est similaire aux précédentes, les courbes nous permettent de visualiser les sorties de vélos de la station sélectionnée au fil d'une journée par intervalles de 15, 30 minutes, 1 et 2 heures. Tous ces graphiques ont été tracés ensembles afin de pouvoir facilement les comparer les uns avec les autres et pour pouvoir les manipuler plus facilement. En effet, si nous avions tracé chaque graphique de manière séparée, nous aurions eu 96 (4×24 stations) graphiques à visualiser, ce qui n'était pas possible. Nous avons tracé ce type de graphique pour chaque station qui compose le réseau et nous avons pu voir qu'elles possédaient toutes des graphiques similaires. Nous avons donc choisi, à titre d'exemple, de vous montrer celui de la station 4 qui est la station située en face de la gare.

Lorsque nous analysons l'histogramme pour les jours de la semaine, nous pouvons voir sans grand étonnement que la plupart des vélos sont utilisés pendant la journée et que la tranche horaire la plus importante est celle du soir, celle qui recouvre les sorties de 15 à 20h. Nous remarquons également que la tranche horaire de midi est plus petite que celle la précédant, ce qui correspond bien au graphe avec les courbes qui est juste en dessous.

Si nous jetons un oeil à l'histogramme pour le week-end, nous pouvons voir que nous avons un comportement totalement différent. Dans ce cas, les pourcentages sont croissants jusque 20h, il y a donc dans ce cas plus de sorties à midi qu'au matin (de 6h à 10h). Cependant, la tranche de 15 à 20h reste la plus importante, les sorties de vélos sont donc plus tardives le week-end qu'en semaine, ce qui confirme bien les différentes observations réalisées dans les sections précédentes.

Passons maintenant à l'analyse des rentrées de vélos en fonction de chaque station à l'aide des graphiques des FIGURES 4.17 et 4.18. Dans ce cas, nous avons mis deux représentations, pour les stations 1 et 4, car elles ne sont pas identiques. En effet, en visionnant les différentes représentations, nous avons pu remarquer que celles des stations 1 et 2 sont différentes des autres, qui elles sont similaires à celle de la station 4.

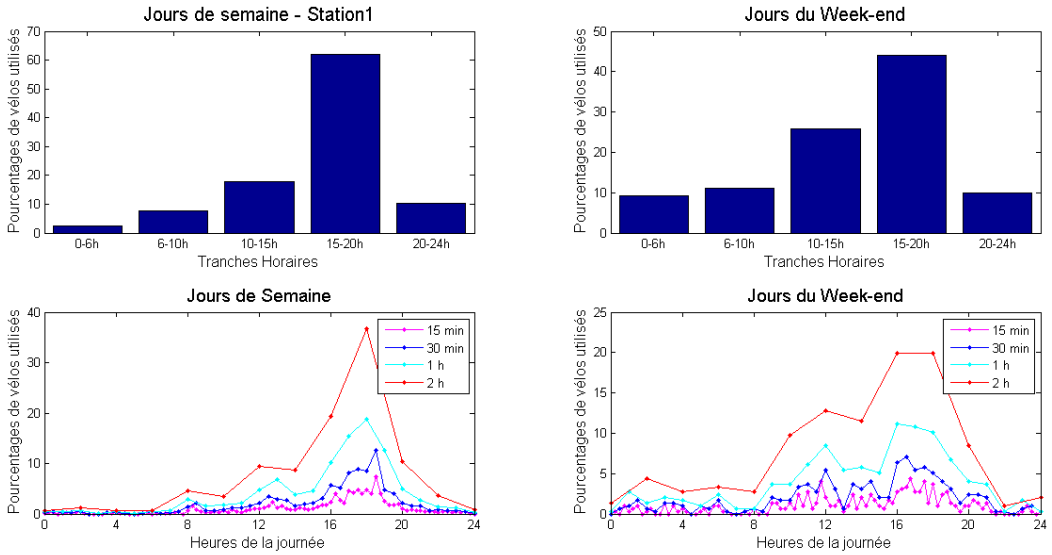


FIGURE 4.17 – Distribution des locations des vélos pour les rentrées de la station 1 - construites avec *Matlab*

Lorsque nous analysons les graphes de la FIGURE 4.17, nous pouvons voir que ceux pour les jours de la semaine sont similaires aux graphes représentant les jours du week-end. En effet, les barres des histogrammes croient de la même manière, celle de la matinée, de 6h à 10h, est plus petite que celle de midi qui est elle plus petite que celle de la soirée de 15 à 20h. Nous n'avons donc pas les trois pics pour le graphe des jours de semaine avec les courbes, il n'y a qu'un pic qui ressort, celui de la soirée comme observé pour le week-end.

Pour le reste des stations, à l'aide de la FIGURE 4.18, nous notons les mêmes observations que pour les sorties, nous avons trois pics durant la journée dont celui de midi qui est inférieur

aux deux autres, du matin et du soir. Ce dernier reste comme toujours le plus important, le plus élevé.

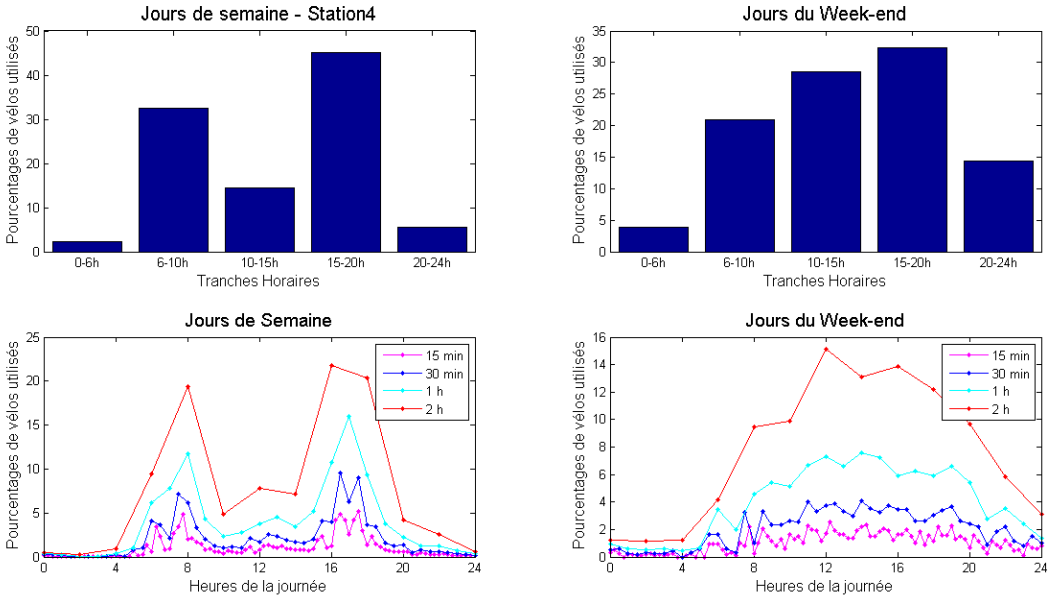


FIGURE 4.18 – Distribution des locations des vélos pour les rentrées de la station 4 - construit avec *Matlab*

Chapitre 5

Analyse du réseau

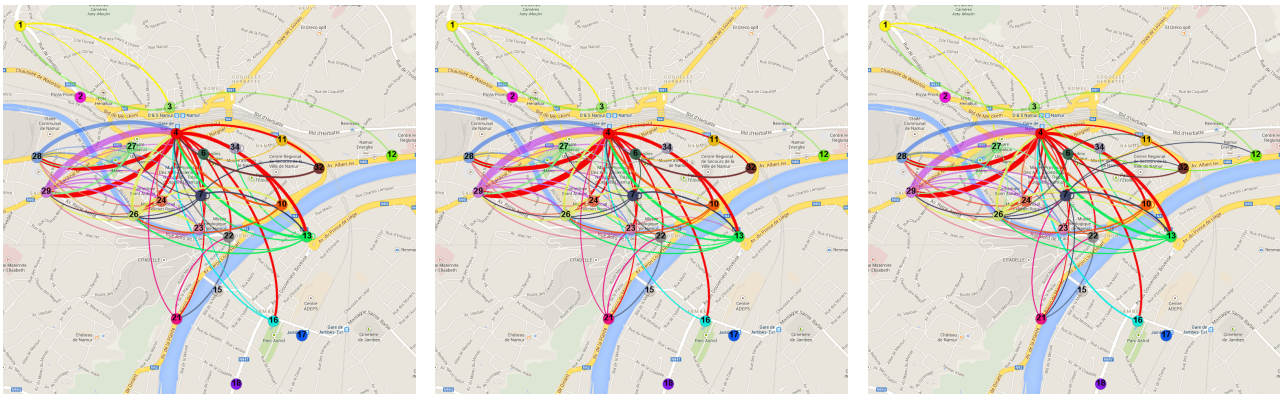
Après avoir effectué quelques analyses statistiques, nous allons désormais analyser des visualisations plus précises du réseau, en rapport avec les résultats découlant des analyses statistiques et similaires aux visualisations présentes dans la section 3.1. Comme nous avons pu le voir dans cette section, les transferts les plus importants ont lieu entre la station située à la gare de Namur et les stations situées à Salzinnes. Nous allons maintenant voir si c'est toujours le cas lorsque nous trions nos données en fonction des abonnements, des heures ou des jours, comme cela a été fait pour le chapitre 4. Notons que la structure de ce chapitre sera similaire à celle du précédent afin de pouvoir faire les liens entre les représentations du réseau et les analyses effectuées auparavant plus facilement et que nous travaillons toujours sur les données nettoyées.

5.1 Utilisation du réseau au fil du temps

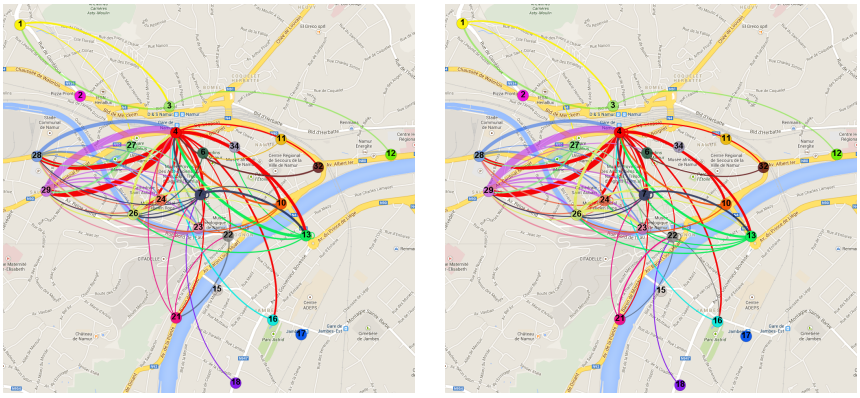
Dans cette partie, nous avons trié nos données, dans un premier temps, en fonction des jours de la semaine et, ensuite, en fonction des heures en séparant les données par tranches horaires. Le tout a pu être effectué grâce aux programmes de l'annexe A.8.

Les graphes présents sur la FIGURE 5.1 permettent de visualiser les transferts des vélos entre les différentes stations du réseau au fil des jours de la semaine et du week-end. Dans ce cas, tous les abonnements et toutes les dates sont rassemblés, nous avons trié nos données uniquement par rapport au jour durant lequel la location a été effectuée et cela grâce à la colonne ajoutée pour différencier ces jours (voir section 2.2). Comme auparavant, les noeuds possèdent tous la même taille, ce sont uniquement les liens qui changent et possèdent la couleur de leur noeud de départ. Notons que dans ce cas, seuls 15% des liens ont été tracés afin de pouvoir discerner les plus importants.

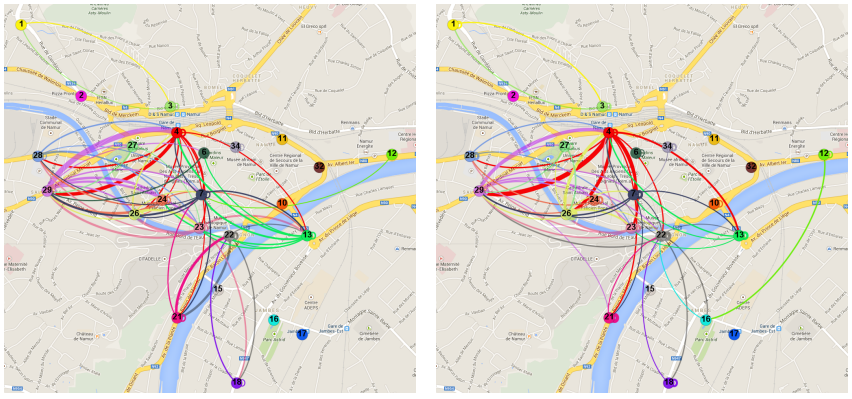
Nous pouvons voir que du lundi au vendredi, les transferts les plus importants ont lieu entre les stations 4 et 28, 29, 10, 13 et 32. Les personnes ont donc tendance à utiliser le réseau pour voyager entre la station 4, qui est la station de la gare, et les stations qui sont situées en périphérie et non pas dans le centre même de la ville. Lorsque nous observons les graphes pour le week-end, nous pouvons voir que les stations 28, 10 et 32 sont moins utilisées au profit de la 18. Aussi bien pour le week-end que la semaine, le transfert de vélos le plus important a lieu de la station 4 à la 29, et la quantité de trajets entre les stations 1 et 3 reste similaire au fil des jours.



(a) Lundi (b) Mardi (c) Mercredi



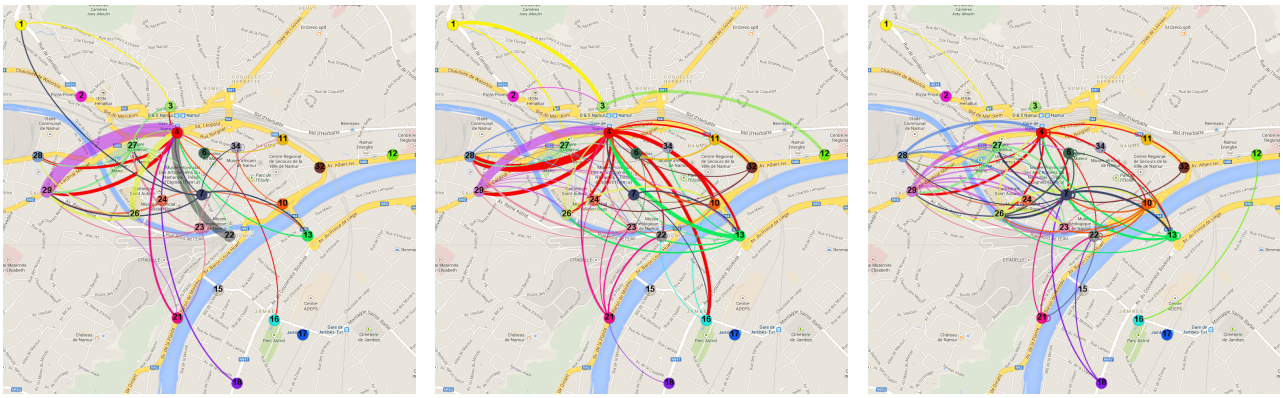
(d) Jeudi (e) Vendredi



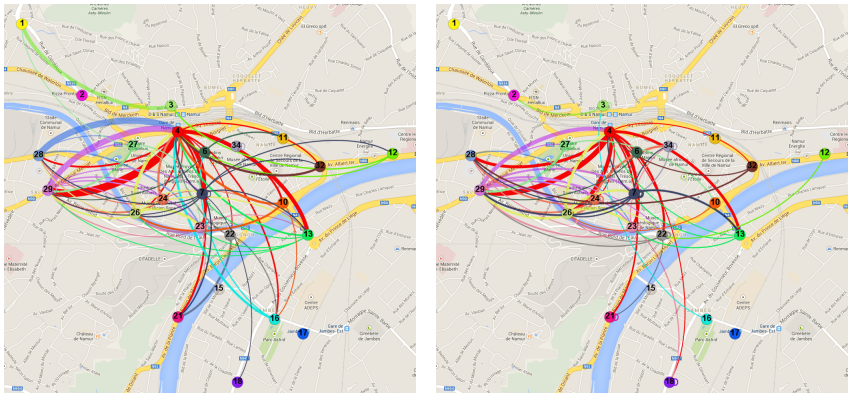
(f) Samedi (g) Dimanche

FIGURE 5.1 – Transferts des vélos entre les stations en fonction des jours de la semaine et du week-end - construites avec *Gephi* et *Gimp*

Passons maintenant à la comparaison par tranches horaires. Dans ce cas, nous avons séparé les données en deux : les données relatives aux jours de semaine (du lundi au vendredi) et celles qui ont eu lieu le week-end. En effet, nous avons pu constater que l'allure du réseau est similaire du lundi au vendredi, nous avons donc rassemblé les données relatives à ces jours afin de réduire le nombre de graphiques à analyser. Ensuite, nous avons séparé nos données en fonction des cinq tranches horaires qui ont déjà été utilisées auparavant. Le tout a encore une fois été réalisé grâce aux programmes de l'annexe A.8.



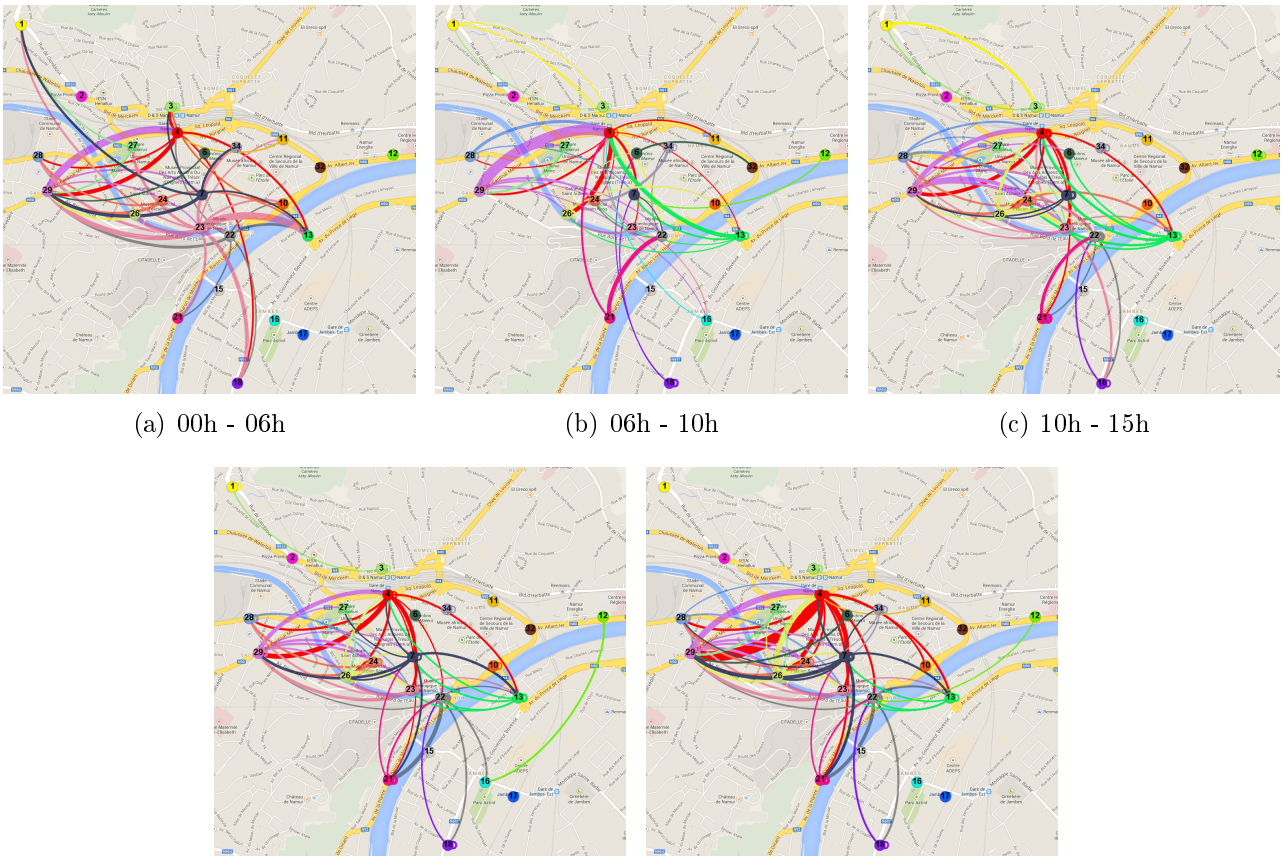
(a) 00h - 06h (b) 06h - 10h (c) 10h - 15h



(d) 15h - 20h (e) 20h - 24h

FIGURE 5.2 – Transferts des vélos entre les stations en fonction des tranches horaires du lundi au vendredi - construites avec *Gephi* et *Gimp*

Lorsque nous analysons les visualisations de la FIGURE 5.2, qui représentent le réseau du lundi au vendredi par tranche horaire, nous pouvons voir que de minuit à six heures, le réseau est utilisé de manière tout à fait particulière, différente de ce qui a été constaté jusqu'à présent. En effet, la quantité de transferts de la station 29 à la 4 est toujours importante mais nous avons en plus de celle-ci, une quantité de transferts tout aussi importante de la station 22, située au bas de la citadelle, à la 4 en face de la gare. Nous constatons également pour cette tranche horaire qu'il y a moins de transferts à partir de la gare (station 4), nous avons beaucoup moins de liens partant de cette station. Notons ici que seuls 15% des liens ont été représentés afin de pouvoir discerner correctement les particularités. Ensuite, dès 6h, nous pouvons voir que le réseau "s'active", les liens partant de la station 4 deviennent plus nombreux. En effet, si nous comparons les visualisations des tranches comprises entre 6h et 24h avec les visualisations de la FIGURE 5.1 pour les jours allant du lundi au vendredi, nous pouvons voir qu'elles sont semblables et en particulier la FIGURE 5.2(d) représentant la tranche horaire 15-20h qui est la tranche horaire durant laquelle nous avons un pic d'utilisation du lundi au vendredi, c'est à ce moment que le nombre de trajets est le plus élevé (voir section 4.2).



(a) 00h - 06h (b) 06h - 10h (c) 10h - 15h

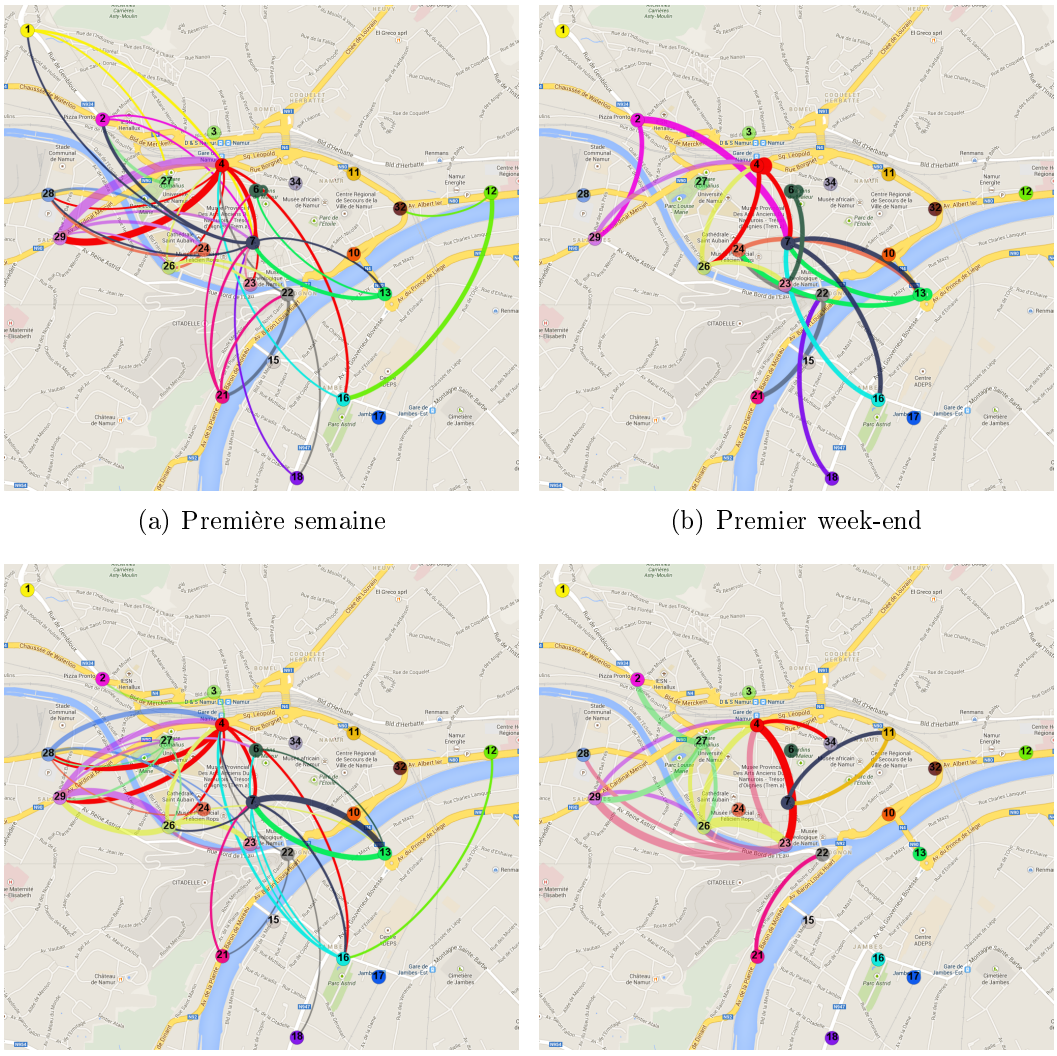
(d) 15h - 20h (e) 20h - 24h

FIGURE 5.3 – Transferts des vélos entre les stations en fonction des tranches horaires du week-end - construites avec *Gephi* et *Gimp*

Lorsque nous analysons les tranches horaires pour le week-end à l'aide des cartes de la FIGURE 5.3, nous pouvons voir que durant la première tranche, de minuit à 6h, ce sont dans ce cas les stations 23 et 7 qui sont les plus utilisées. Ces stations sont proches l'une de l'autre et sont toutes les deux situées dans le centre de la ville, elles sont donc plus proches des bars et cafés qui composent le "vieux Namur", qui est un quartier connu de la ville en grande partie semi-piéton. De 6h à 10h, nous pouvons voir que nous avons un flux important partant de la station 29 et allant vers la 4 alors que de 20h à minuit ce flux est en sens inverse. De 10h à 20h nous pouvons remarquer que les transferts sont similaires et que les stations situées à l'est sont délaissées en faveur des stations situées à l'ouest de la ville.

Comme nous l'avons remarqué dans la section 4.2, nous avons eu une baisse d'utilisation des vélos durant les vacances de Noël de l'année 2012. Cette différence nous a incité à visualiser l'utilisation du réseau durant ces deux semaines afin de voir s'il y a une différence avec les visualisations globales en fonction des jours, regroupant toutes les données triées, se trouvant sur la FIGURE 5.1. Pour ce faire, nous avons décidé de sélectionner les données relatives aux deux semaines de vacances de manière séparée, pour la première et la deuxième semaine, que nous avons ensuite re-divisées en fonction des jours de semaine et du week-end. Nous avons donc obtenu les quatre graphiques se situant sur la FIGURE 5.4. Dans ce cas, nous avons conservé 25% des liens étant donné que nous travaillons sur une tranche temporelle plus petite, nous

avons donc moins de données. À l'aide de ces figures, nous pouvons voir que pendant la semaine, le réseau est utilisé comme à l'habitude, nous avons des transferts importants entre les stations situées à Salzinnes et à la gare et dans le centre. Pour ce qui est du week-end par contre, nous constatons que les représentations ne sont pas du tout comparables aux autres et même entre-elles. Durant la première semaine, le réseau a été particulièrement utilisé au centre, à l'ouest et au sud de la ville tandis que durant le second week-end, seul le centre et l'ouest ont été fréquentés. Encore une fois, nous pouvons voir que les stations situées à l'est de la ville ne sont pas beaucoup utilisées mise à part la station 12 qui a été utilisée plus fréquemment durant les jours de semaine.



(a) Première semaine

(b) Premier week-end

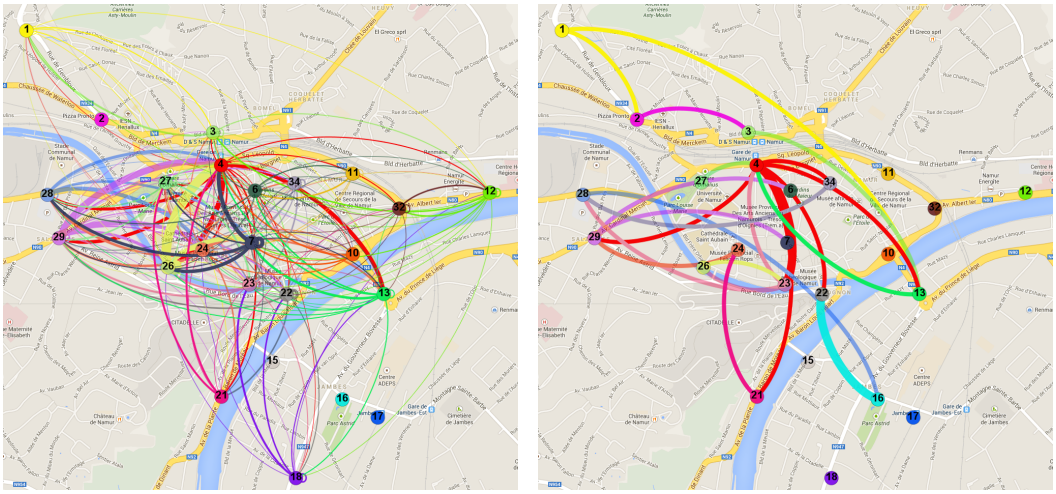
(c) Seconde semaine

(d) Second week-end

FIGURE 5.4 – Transferts des vélos entre les stations en fonction des jours pour les deux semaines de vacances de Noël 2012 - construites avec *Gephi* et *Gimp*

Par la suite, nous avons également analysé les fluctuations des taux de locations. Grâce à cela, nous avons pu déterminer les jours où le taux de locations était le plus élevé et le plus faible. Ces jours correspondaient respectivement au 25 mai 2012, avec 519 locations, et au 2 décembre 2012, avec 25 locations. Nous avons visualisé le réseau en ces jours (FIGURE 5.5) afin

de voir la manière dont celui-ci a été utilisé. Dans le but d'avoir toutes les informations, nous avons tracé 100% des liens, de cette manière nous pouvons discerner les stations qui n'ont pas été du tout utilisées durant ces jours. Dès lors, nous pouvons voir que les taux de transferts les plus élevés ne se trouvent pas aux mêmes endroits. En effet, pour le jour où le taux d'utilisation a été le plus élevé, le transfert le plus important a eu lieu de la station 26 à la 4 tandis que pour le jour où l'utilisation a été minimale le transfert le plus important a eu lieu de la station 26 à la 22 et de la 4 à la 23.



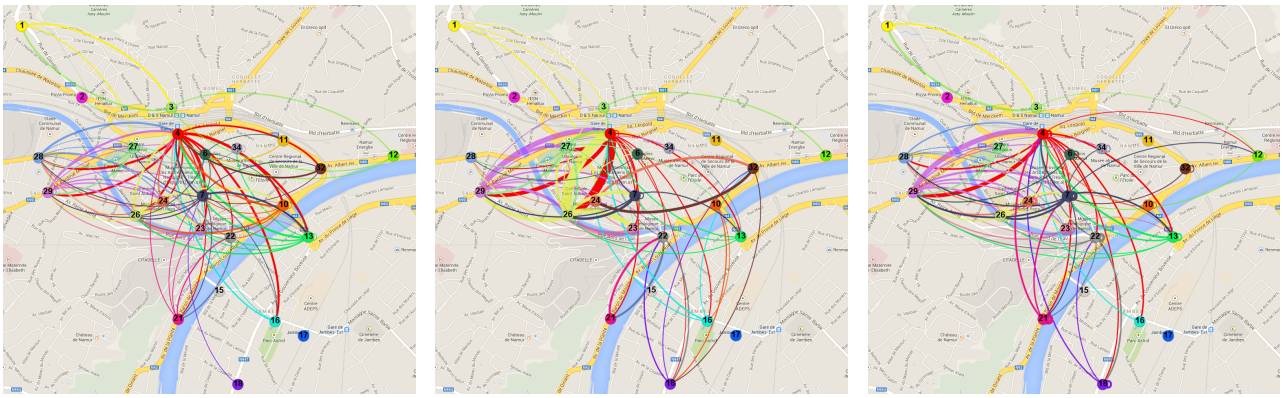
(a) Jour max - 25 Mai 2012 (b) Jour min - 2 décembre 2012

FIGURE 5.5 – Transferts des vélos entre les stations en fonction des abonnements pour les jours d'utilisation maximale et minimale - construites avec *Gephi* et *Gimp*

5.2 Utilisation du réseau en fonction des abonnements

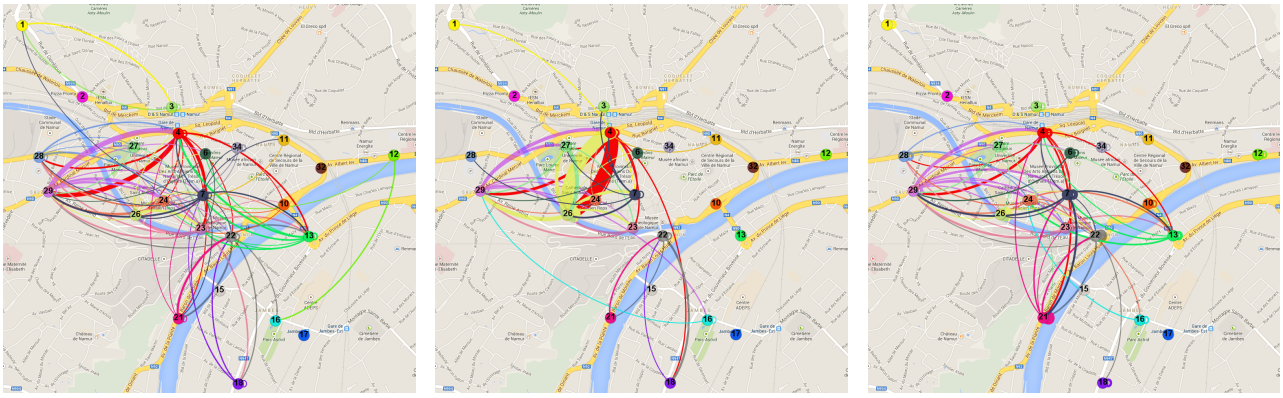
Comme nous avons pu le voir dans la section 4.3, le comportement des utilisateurs du système varie suivant le type d'abonnement qu'ils possèdent. Nous allons maintenant voir si l'utilisation des différentes stations qui composent le réseau varie ou non en fonction de l'abonnement détenu par les utilisateurs. Pour ce faire, nous avons travaillé sur nos données séparées en fonction des trois types d'abonnements (journalier, hebdomadaire et annuel) et des jours de semaine ou du week-end, nous avons donc deux visualisations pour chaque abonnement, une pour la semaine et une pour le week-end.

Ces visualisations se trouvent sur la FIGURE 5.6 pour les jours de semaine et sur la FIGURE 5.7 pour le week-end. Nous pouvons voir que les visualisations pour les abonnements annuels et journaliers sont assez similaires aussi bien pour la semaine que le week-end. La seule différence particulière concerne principalement les abonnements d'une semaine car nous pouvons voir que pour la semaine entière, c'est entre la station 4 et la station 26, qui se situe au niveau de la passerelle, qu'ont lieu les transferts les plus importants de vélos. Donc, mis à part pour les abonnements hebdomadaires, nous pouvons conclure que les utilisateurs du réseau y voyagent de la même manière qu'ils possèdent un abonnement annuel ou journalier.



(a) Abonnements annuels (b) Abonnements hebdomadaires (c) Abonnements journaliers

FIGURE 5.6 – Transferts des vélos entre les stations en fonction des abonnements pour les jours en semaine - construites avec *Gephi* et *Gimp*



(a) Abonnements annuels (b) Abonnements hebdomadaires (c) Abonnements journaliers

FIGURE 5.7 – Transferts des vélos entre les stations en fonction des abonnements pour les jours du week-end - construites avec *Gephi*

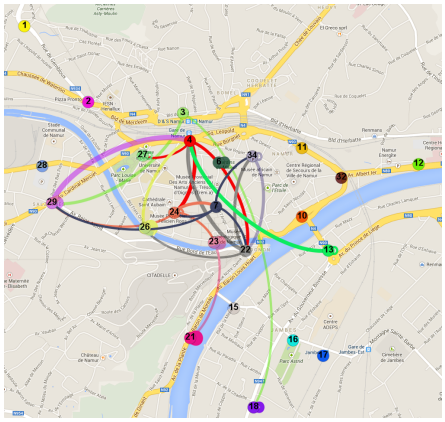


FIGURE 5.8 – Transferts des vélos entre les stations le 1er mai 2012 - construite avec *Gephi*

Dans cette même section, nous avons constaté un pic d'utilisation le 1er mai 2012 pour les personnes possédant un abonnement journalier. Nous avons donc visualisé les trajets entre les différentes stations pour ce jour afin de voir si l'utilisation a été différente de l'utilisation moyenne des personnes détenant un abonnement d'un jour, visible sur les FIGURES 5.6(c) et 5.7(c). Cette visualisation se trouve sur la FIGURE 5.8 et nous permet de remarquer que le réseau n'est pas similaire aux figures énoncées précédemment, il ne peut donc pas être considéré comme un jour de semaine ou du week-end habituel pour les abonnés d'un jour. Cependant, si nous comparons cette visualisation aux FIGURES 5.4(b) et 5.4(d) nous pouvons voir qu'elles ont plus de ressemblances. Ce qui confirme que les utilisateurs ont un comportement particulier durant les week-ends des vacances ou d'un jour férié.

Chapitre 6

La classification

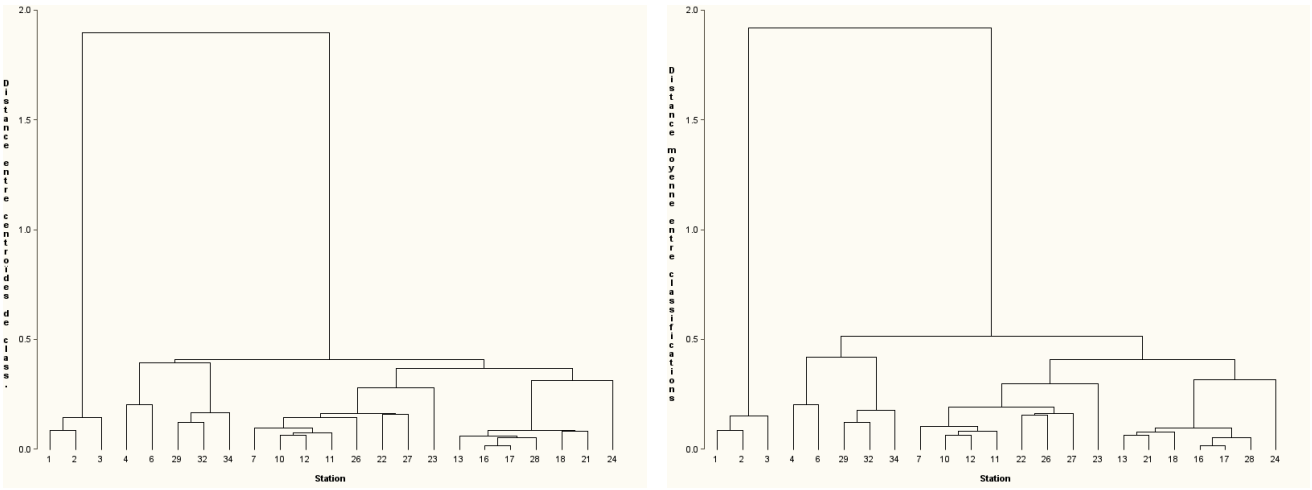
Après les différentes analyses effectuées pour observer l'utilisation des vélos en fonction du temps, nous allons essayer de voir s'il existe des similitudes entre certaines stations. Afin d'effectuer les différentes méthodes de classification, nous avons créé quatre matrices $n \times m$ où n est le nombre de stations et m le nombre de tranches horaires considérées. Dans un premier temps, nous avons considéré $m = 4$ car nous avons choisi de ne conserver que les quatre dernières tranches horaires parmi celles considérées auparavant. Par la suite, nous avons travaillé avec des tranches horaires plus fines, d'une heure, afin d'avoir des classifications plus précises. Les deux premières matrices contiennent la quantité de vélos sortis par station en fonction de chaque tranche horaire pour les jours de semaine et pour ceux du week-end, tandis que les deux autres contiennent la quantité de vélos rentrants par station et par tranche horaire pour les jours de semaine et du week-end de manière séparée. Dans notre cas, nous allons effectuer nos analyses uniquement sur les matrices concernant les sorties.

6.1 Méthodes de classifications standard

Afin d'avoir une première idée d'une éventuelle classification, nous avons appliqué les méthodes classiques qui sont le lien moyen, les centroïdes et la méthode de Ward. Dans ce cas, nous obtenons pour toutes les méthodes une classification en deux classes. En effet, la FIGURE 6.1 contient les résultats de la classification à l'aide du dendrogramme pour la méthode des centroïdes et du lien moyen par rapport aux fichiers de sorties des vélos pour les jours du week-end et nous pouvons voir que les deux dendrogrammes nous suggèrent cette classification. Les résultats des autres méthodes, appliquées sur les deux matrices considérées, ressortent tous la même chose. Toutes les classifications nous suggèrent de séparer nos données en deux classes principales, la première contenant les stations 1, 2 et 3, situées géographiquement à Belgrade et Bomel, et la seconde les autres stations composant le réseau, qui se situent dans le centre de Namur, à Jambes et à Salzinnes. Les graphiques de la FIGURE 6.2 nous permettent de comprendre pourquoi cette classification ressort autant. En effet, nous pouvons voir qu'aussi bien pour les jours de la semaine que pour le week-end, le nombre de locations qui sortent des trois premières stations est toujours en dessous de celui des autres stations.

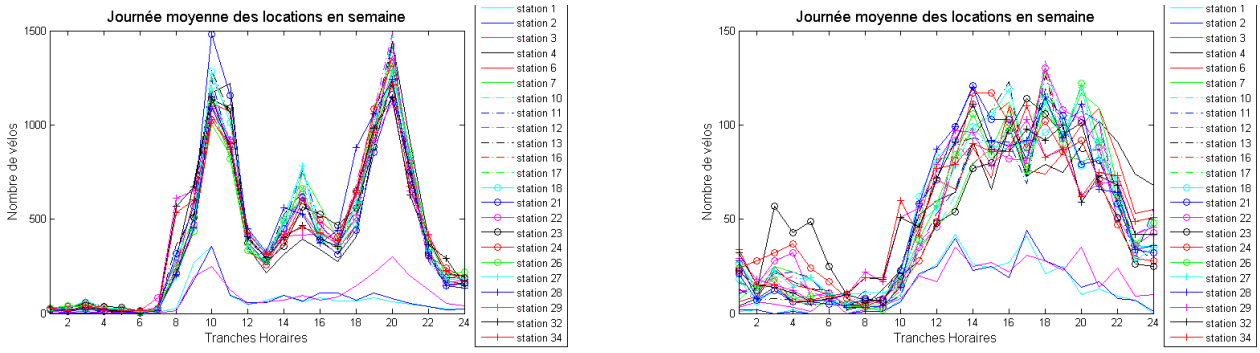
Cependant, nous pouvons voir sur les dendrogrammes que, pour les jours du week-end, lorsque l'on regarde la classification pour un nombre plus élevé de classes, celle-ci n'est pas identique suivant la méthode appliquée. les stations 13 et 21 par exemple vont être mises

directement dans la même classe lors de la première classification de la méthode du lien moyen tandis que pour les centroïdes, elles seront mises ensembles après que la 13 ait été mise avec les stations 16, 17 et 28.



(a) Méthode des centroïdes (b) Méthode du lien moyen

FIGURE 6.1 – Dendrogramme de la méthode des centroïdes et du lien moyen appliquées à la matrice relatives aux sorties des vélos le week-end - construites avec *SAS*



(a) Jours de Semaine (b) Jours du weel-end

FIGURE 6.2 – Nombre de locations en fonction en fonction des heures pour chaque station - construites avec *Matlab*

Rappelons que ces deux méthodes sont des méthodes de classification ascendantes hiérarchique, elles vont donc partir du nombre de classes maximum pour ensuite diminuer celui-ci. L'indice d'agrégation du lien moyen consiste à prendre la moyenne des distances entre les paires d'observations de deux groupes différents. La méthode du lien moyen aura donc tendance à regrouper des groupes possédant une petite variance. La méthode des centroïdes elle, va chercher à construire des classes de types sphériques étant donné que son indice d'agrégation consiste à calculer la distance entre les différents centres des classes et de rassembler les classes dont les centres de gravité sont les plus proches.

6.2 La similarité cosinus

Afin de pouvoir classifier nos stations de manière plus fine, nous avons décidé, dans un premier temps, d'utiliser la similarité cosinus. Cette méthode permet de calculer la similarité entre deux vecteurs de même dimension en calculant le cosinus de l'angle entre eux grâce à la formule

$$\cos\theta = \frac{A \cdot B}{\|A\| \cdot \|B\|}.$$

Le résultat de cette fraction est compris en -1 et 1. Dans le cas où $\cos\theta$ vaut 0, cela signifie que les vecteurs sont orthogonaux, ils ne sont donc pas similaires, dans le cas où $\cos\theta$ vaut 1 ou -1, les vecteurs sont similaires. Notons que nous n'aurons jamais de $\cos\theta$ dont la valeur est négative car nos vecteurs ont uniquement des entrées positives. Il est possible de calculer la similarité cosinus via *Matlab* grâce à la fonction `pdist` qui ressort

$$d(A,B) = 1 - \cos\theta = 1 - \frac{A \cdot B}{\|A\| \cdot \|B\|}.$$

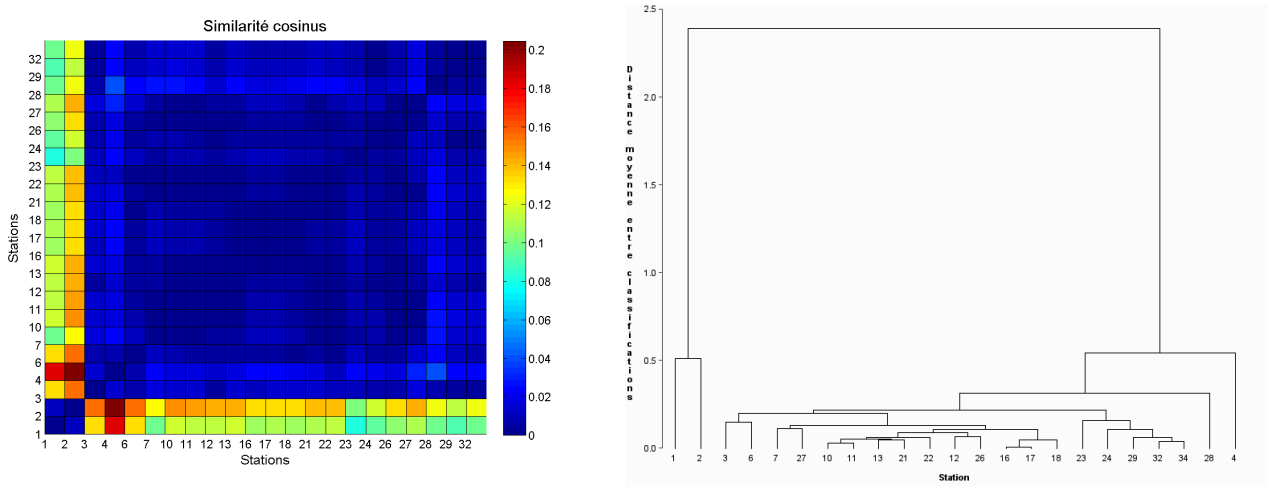
Cela signifie que nos interprétations sont inversées par rapport à celles de départ ; lorsque $d(A,B)$ vaut 0 cela signifie que nos vecteurs sont similaires tandis que lorsqu'elle vaut 1 cela implique qu'ils sont orthogonaux.

Nous avons appliqué cette méthode aux matrices des sorties des vélos, du week-end et de la semaine, contenant les données en fonction de chaque tranche horaire mais encore une fois, nous obtenons la même classification en deux classes principales. Dès lors, au lieu de séparer nos données par tranche horaire et de comparer toutes les données de toutes les tranches entre-elles directement, nous avons décidé de séparer nos données en fonction des heures, pour ensuite effectuer des classifications sur les 8 premières heures, les six dernières, etc. Nous n'avons donc plus quatre colonnes, liées aux quatre tranches horaires considérées au départ (6-10h,10-15h,...) mais bien 24, une pour chaque heure. Après cette séparation des données, nous avons re-calculé la matrice de similarité cosinus, pour toutes les paires de stations, propre à chaque tranche horaire que l'on souhaite analyser de manière séparée pour ensuite ré-appliquer la méthode classique du lien moyen pour observer les différentes classifications qu'elle nous suggère en fonction des différentes tranches horaires. Notons que nous avons uniquement appliqué cette méthode de classification afin de ne pas alourdir le travail.

Nous avons choisi d'analyser les tranches horaires de 00h à 8h, de 8h à 14h, de 14h à 19h et de 19h à 24h. Nous avons donc appliqué la similarité cosinus à chacune de nos quatre tranches horaires de manières séparées, ce qui implique que nous avons obtenu cette fois-ci huit matrices différentes (2×4 , pour la semaine et le week-end), contenant les valeurs ressorties par la fonction `pdist`, qui calcule $1 - \cos\theta$ entre chaque vecteur, pour chaque station et pour les heures correspondant à la tranche horaire choisie. La représentation de ce type de matrice pour les données relatives à la tranche horaire 14h-19h en semaine est visible sur la FIGURE 6.3(a).

Nous pouvons très facilement voir que pour cette tranche horaire, nous avons deux groupes qui ressortent grâce aux deux groupes de carrés bleus, un premier avec les deux premières stations et un second avec les autres. Cependant, nous aimerions analyser la classification que nous pourrions établir parmi les stations lorsque nous excluons les stations 1 et 2. Étant donné

que les nuances de bleus sur le graphique ne sont pas assez différentes que pour établir d'autres groupes parmi le principal, nous avons ré-appliquer la méthode des liens moyens sur notre matrice de similarité cosinus afin d'obtenir le dendrogramme se trouvant à la FIGURE 6.3(b).



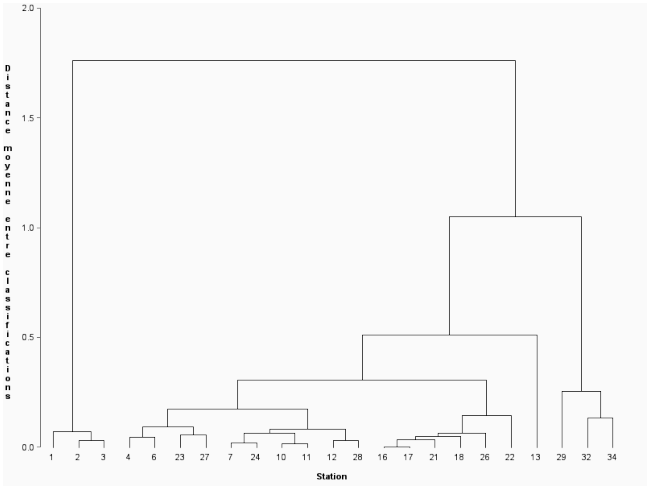
(a) Matrice similarité cosinus : $1 - \cos\theta$

(b) Dendrogramme de la méthode du lien moyen

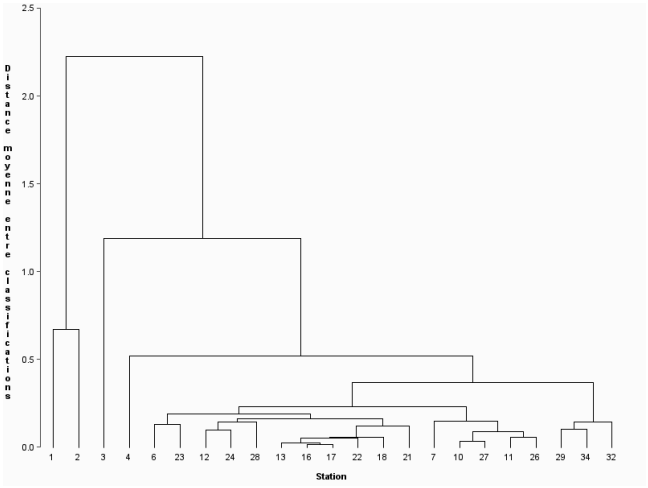
FIGURE 6.3 – Analyse sur les sorties des vélos de 14 à 19h en semaine - construites avec *Matlab* et *SAS*

Si nous analysons la classification en trois classes, nous pouvons voir que la station 4 est isolée du groupe principale pour former une classe à elle seule, ce qui concorde bien avec la FIGURE 6.3(a). En effet, nous pouvons voir sur ce graphe que l'intersection entre les stations 4, 1 et 2 est de couleur bordeaux, ce qui signifie qu'elles ont tendance à ne pas être similaires. De plus, si nous observons le grand carré bleu formé par les intersections entre les autres stations, nous pouvons voir que la ligne correspondant à la station 4 est plus claire, ce qui implique donc que cette station sera moins similaires aux autres. Nous pouvons faire la même observation pour la station 28 lorsque nous analysons la classification en quatre classes.

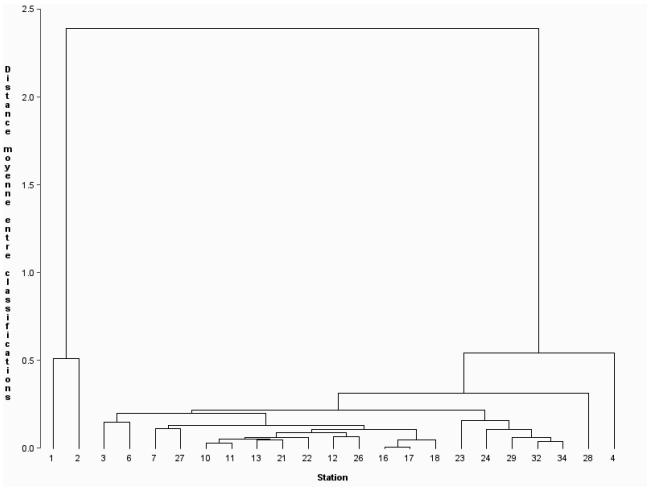
Nous pouvons observer les dendrogrammes obtenus pour les données en semaine pour chaque tranche horaire sur la FIGURE 6.4 tandis que ceux pour le week-end se trouvent sur la FIGURE 6.5. Nous pouvons tout d'abord voir, aussi bien en semaine que le week-end, que les classifications proposées ne sont pas totalement similaires en fonction des différentes tranches horaires. En effet, les stations 1, 2 et 3 ne sont ensembles que pour les données de 00h à 8h en semaine, le reste du temps nous avons juste une première classe qui contient les stations 1 et 2, sauf pour les données de la semaine de 19 à 24h. Par contre nous pouvons voir que les stations 16 et 17 sont toujours mises ensembles dès la première classification sauf pour le week-end de 00h à 8h. Remarquons également le rassemblement rapide des stations 29, 32 et 34 à chaque fois sauf pour le week-end de 19h à 24h.



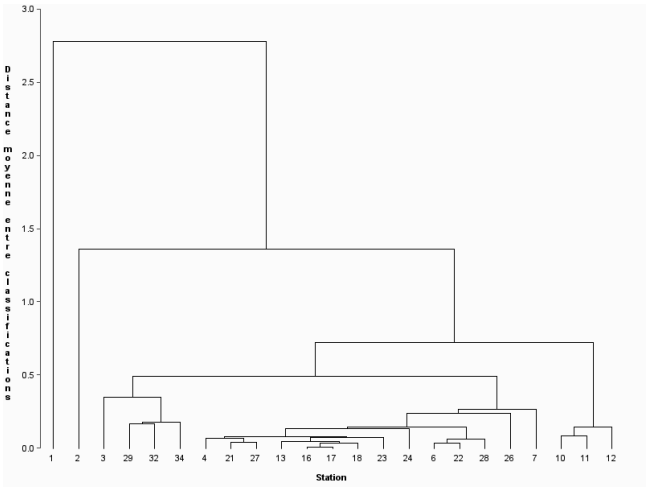
(a) 00-8h



(b) 8h-14h

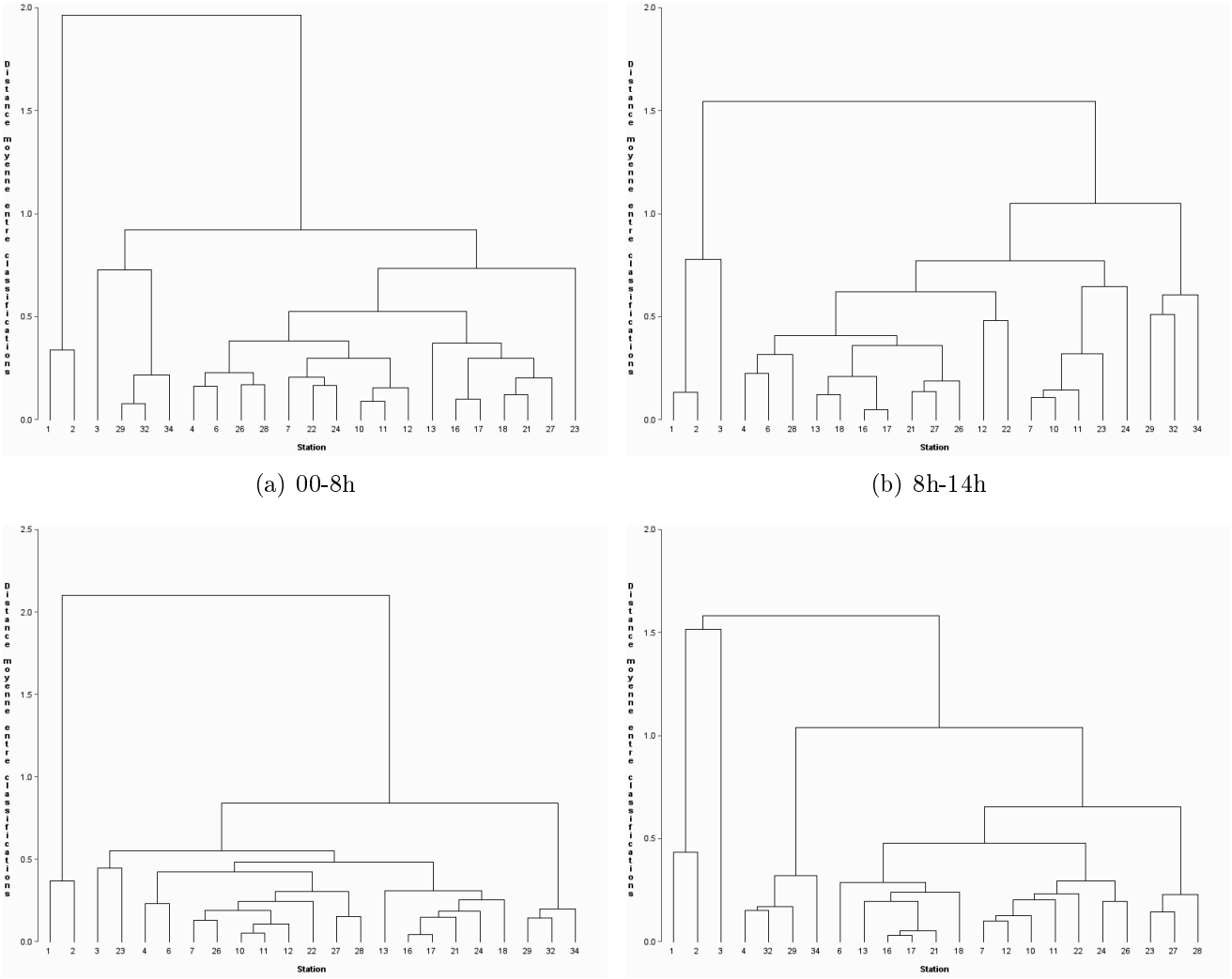


(c) 14h-19h



(d) 19h-24h

FIGURE 6.4 – Dendrogramme du lien moyen pour les jours de la semaine (du lundi au vendredi) - construites avec *Matlab*



(a) 00-8h

(b) 8h-14h

(c) 14h-19h

(d) 19h-24h

FIGURE 6.5 – Dendrogramme du lien moyen pour les jours du week-end - construites avec *Matlab*

Rappelons que nous n'avons analysé que les sorties, les données pour les rentrées peuvent également être intéressantes à analyser. De plus, nous avons effectué une classification basée sur les volumes échangés entre les stations. Il peut être également intéressant de normaliser le nombre de vélos pour chaque tranche horaire de chaque station par le nombre total de vélos sortis (ou rentrés) de la station, ce qui permettrai donc de faire une classification sur l'utilisation temporelle des vélos.

Conclusion

Grâce aux différentes analyses effectuées durant ce mémoire, nous avons pu découvrir les différents facteurs influençant le comportement des utilisateurs du réseau de vélos en libre service de la ville de Namur.Nous avons pu mettre en évidence certaines choses auxquelles nous n'aurions pas pensé à première vue.

Tout d'abord, nous avons pu voir, dans le chapitre 3, que la station 4 Gare de Namur était la station la plus utilisée aussi bien comme point de départ que d'arrivée. Située près du centre du réseau et donnant accès aux différents moyens de transports en commun possibles, cette gare est un lieu essentiel du système. Cependant, nous avons pu constater que le nombre de vélos sortant de cette station est plus élevé que le nombre de vélos y entrant, il y a donc un déséquilibre qui implique que cette station va plus souvent être vide que pleine de vélos. Dans le cas contraire, la station 17, située en face de la gare de Jambes, est celle qui a eu le moins d'entrées et de sorties de vélos dans notre fichier de données. Les quantités de vélos échangés les plus importantes se situaient entre la station de la gare de Namur et les stations situées à Salzinnes. Nous pouvons donc conclure que les utilisateurs ont tendance à emprunter les vélos pour rejoindre les stations de Salzinnes plutôt que les stations situées dans le centre.

Ensuite, les analyses statistiques du chapitre 4 nous ont permis de mettre en évidence l'impact de la gratuité des 30 premières minutes d'utilisation. La majorité des personnes utilisent les vélos pour des trajets de moins de 20 minutes qu'importe l'abonnement qu'ils possèdent, ils utilisent les vélos pour effectuer des trajets de courte durée. En plus de cela, nous avons pu voir qu'il y avait trois pics d'utilisation du lundi au vendredi alors que durant le week-end, les personnes ont tendance à utiliser le réseau plus tardivement dans la journée et durant deux moments en particuliers. La semaine, le pic d'utilisation le plus important se situe aux alentours de 18h, ce qui correspond aux retours des personnes à leur domicile après leur journée de travail, tandis que le week-end, celui-ci se situe entre 12h et 16h, l'utilisation en fonction des jours n'est donc pas du tout similaire.

Dans ce même chapitre, nous avons constaté que le type d'abonnement détenu par les utilisateurs, la météo ainsi que les périodes de vacances ou les jours fériés avaient une influence considérable sur le pourcentage journalier de vélos utilisés. Les détenteurs d'un abonnement annuel auront tendance à utiliser le réseau en semaine vers 8h, 12h et 18h alors que ceux possédant un abonnement d'un jour utiliseront plutôt les vélos le week-end à partir de midi. Nous avons pu remarquer que durant les vacances, le réseau est moins sollicité par les personnes possédant un abonnement annuel alors que les utilisateurs possédant un abonnement d'un jour auront tendance à l'utiliser durant les jours fériers.

Le chapitre 5 nous a permis quant à lui de montrer que les utilisateurs du réseau y voyagent de la même manière quel que soit le type d'abonnement qu'ils possèdent. En effet, aussi bien pour la semaine que le week-end, les transferts de vélos les plus importants ont toujours lieu entre la station 4 et la station 29, située à Salzinnes.

Finalement, nous avons pu classifier nos stations par rapport aux nombres de vélos sortants de celles-ci. Les différentes classifications obtenues diffèrent suivant les jours ou les heures considérées, mais nous avons pu voir que les stations de Saint-Servais et Bomel avaient tendance à être regroupées ensembles tandis que toutes les autres stations étaient placées dans d'autres groupes différents au fil des jours et des heures. Pour compléter ce mémoire, des classifications plus approfondies et en fonction des rentrées des vélos auraient également pu être effectuées.

D'un point de vue spatial, il est également possible de déterminer le parcours moyen des vélos à condition d'avoir différentes positions GPS au cours du temps. Les vélos ne sont pas dotés de GPS, nous n'avons donc pas leurs positions dans les données fournies. Il faudra alors récolter ces informations en utilisant les vélos et en se munissant d'un GPS. Ces reconstitutions de parcours permettront de montrer l'uniformité de la distribution des différentes stations dans la Ville de Namur. À partir de cela, il sera également possible de construire un réseau abstrait et déterminer s'il existe une corrélation avec les différents lieux d'intérêts situés à proximité des stations. Cependant, même sans GPS nous pourrions essayer d'estimer le parcours le plus probable en connaissant la station de départ, la station d'arrivée (et donc les distances qui les relient en suivant les différents chemins possibles), le temps du parcours et la vitesse moyenne.

De plus, cela permettra de chercher le parcours le plus probable en fonction du temps d'utilisation, d'une vitesse moyenne et du réseau routier de la ville de Namur.

Bibliographie

[1] A. Hardy, *Aspects statistiques de la classification*, FUNDP, 2012-2013.

[2] P. Borgnat, P. Abry, P. Flandrin, C. Robardet,J-B. Rouquier & E. Fleury, *Shared bicycles in a city : a signal processing and data analysis perspective*, Lyon, Juin 2010.

[3] J. Froehlich, J. Neumann & N. Oliver, *Measuring the pulse of the city through Shared Bicycle Programs*, `http://sensorlab.cs.dartmouth.edu/urbansensing/papers/froehlich_urbansense08.pdf`, 2008, consulté en Janvier 2013.

[4] J. Froehlich, J. Neumann & N. Oliver, *Sensing and Predicting the Pulse of the City Throught Shared Bicycle Programs*, `http://www.nuriaoliver.com/bicing/IJCAI09_Bicing.pdf`, 2009, consulté en Janvier 2013.

[5] F. Girardin, A. Vaccari, A. Gerber, A. Biderman & C. Ratti, *Quantifying urban attractiveness from the distribution and density of digital footprints*, `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.172.9964&rep=rep1&type=pdf`, 2009, consulté en Janvier 2013.

[6] F. Girardin, F. Calabrese, F. Dal Fiore, C. Ratti, J. Blat, Digital footprinting : Uncovering tourists with user-generated content, *IEEE Pervasive Computing*, Volume 7, Issue 4 Octobre - Décembre 2008, p. 36-43 .

[7] Krzywinski, M. et al., *Circos : an Information Aesthetic for Comparative*, `http://genome.cshlp.org/content/early/2009/06/15/gr.092759.109.abstract`,Genomics.Genome Res, 2009, 19 :1639-1645.

[8] Service publique de wallonie, *Recensement circulation 2010*, `http://routes.wallonie.be/download.do?action=2&shortId=4211&comment=T%E91%E9charger%20%ABRecensement%20de%20la%20circulation%202010%BB`, 2010, consulté en Avril 2014.

[9] T. Rattenbury, N. Good, M. & M. Naaman, *Towards automatic extraction of event and place semantics from flickr tags.*, `http://cens.ucla.edu/~mhr/cs219/BackEndSystem/rattenbury07.pdf`, 2007, consulté en Avril 2013.

[10] Bake share map, *Paris*, `http://bikes.oobrien.com/paris/timeline/`, consulté en Février 2013.

[11] Circos, `http://circos.ca/`, consulté en Avril 2014.

[12] Circos table viewer, `http://mkweb.bcgsc.ca/tableviewer/visualize/`, consulté en Avril 2014.

[13] Circos vistable, `http://circos.ca/presentations/articles/vis_tables1/#vistabular`, consulté en Mai 2014.

[14] European smart cities, http://www.smart-cities.eu/why-smart-cities.html, consulté en Avril 2013.

[15] Flickr, http://www.flickr.com/, consulté en Avril 2013.

[16] Climat IRM décembre 2012, http://www.meteo.be/meteo/view/fr/9278931-Decembre+2012.html, consulté en mai 2014.

[17] Climat IRM avril 2013, http://www.meteobelgique.be/article/85-annee-2013/1889-resume-avril-2013.html, consulté en mai 2014.

[18] Gephi, https://gephi.org/, 2008, consulté en Avril 2013.

[19] Google Earth, http://www.google.fr/intl/fr/earth/index.html, consulté en Mai 2013.

[20] Google Map, http://maps.google.com/, consulté en Mai 2013.

[21] IBM, *the smart city traffic*, http://www-03.ibm.com/innovation/us/thesmartercity/traffic/index.html#!/2, consulté en Avril 2013.

[22] Li Bia vélos, http://www.libiavelo.be/, consulté en Janvier 2013.

[23] MathWorks, *Matlab*, http://www.mathworks.nl/, 1994-2013, consulté en Avril 2013.

[24] Machine Learning : Cosine Similarity for Vector Space Models, http://pyevolve.sourceforge.net/wordpress/?p=2497, consulté en Mai 2014.

[25] The R Project for Statistical Computing, http://www.r-project.org/, consulté en Avril 2013.

[26] Revealing Paris Through Velib' Data, http://www.girardin.org/fabien/tracing/velib/, consulté en Avril 2013.

[27] SAS, http://www.sas.com/, 2012, consulté en Avril 2013.

[28] Smart City, http://en.wikipedia.org/wiki/Smart_city, consulté en Mars 2013.

[29] Thomson Reuters, *Web of science*, http://thomsonreuters.com/products_services/science/science_products/a-z/web_of_science/, consulté en Avril 2013.

Table des figures

1 La zone rose correspond aux coordonnées des photos provenant du site *Flickr* sur lesquelles le mot "ruines" a été "tagué" - [6] 2

1.1 Collaborations scientifiques entre les différents départements de l'Université de Namur. 5
1.2 Transferts de vélos entre les différentes stations de la ville de Namur - construite avec *Gephi*. 6
1.3 Visualisation *Circos* des données de la TABLE 1.1 - [12]. 7
1.4 Explications construction graphe *Circos* - [13] 8
1.5 Distribution des temps de location des vélos pour la ville de Lyon - [2] 9
1.6 Utilisation des vélos sur une journée pour une station (a), pour toutes les stations (b) et au cours de la semaine (c) - [4] 10
1.7 Classification des différentes stations de la ville de Barcelone - [4] 11

3.1 Tarifs des locations des Li'Bia vélos - [22] 17
3.2 Carte des stations des vélos libres dans la ville de Namur construite à l'aide de *Gephi* et de *Google Earth* 18
3.3 Visualisation des liaisons entre les stations de vélos - construite à l'aide du site *Circos* 19
3.4 Réseau des stations de vélos libres dans la ville de Namur en fonction des rentrées de vélos - construite à l'aide des logiciels *Gephi* et *Gimp* 21
3.5 Réseau des stations de vélos libres dans la ville de Namur en fonction des sorties de vélos - construite à l'aide des logiciels *Gephi* et *Gimp* 21
3.6 Répartition des locations en fonction des stations pour les entrées et les sorties de vélos en fonction des stations - construite à l'aide de *Matlab* 22
3.7 Fluctuations entre le nombre de vélos entrants et sortants en fonction des stations : $(N_{sorties} - N_{entrees})$ - construite avec *Matlab* 22
3.8 Représentation des liens entre les stations - construite à l'aide des logiciels *Gephi* et *Gimp* 23
3.9 Représentation des liens entre les stations avec le rassemblement des entrées et sorties - construite à l'aide des logiciels *Gephi* et *Gimp* 24

4.1 Répartition des vélos en fonction du type de sorties - construite avec *SAS* 27
4.2 Distribution des temps de locations des vélos en fonction des minutes - construite avec *Matlab* 28
4.3 Activité du réseau pour toutes les stations pour les jours de semaine et du week-end - construite avec *Matlab* 30
4.4 Distribution des locations des vélos en fonction du temps - construit avec *Matlab* 31

4.5 Distribution de l'utilisation du réseau pour chaque date contenue dans le fichier - construites avec *Matlab* . . . 32
4.6 Zoom 244ème jour - construite avec *Matlab* . . . 33
4.7 Utilisation du réseau pendant les vacances - construites avec *Matlab* . . . 34
4.8 Courbe cumulative des locations de vélos - construite avec *Matlab* . . . 35
4.9 Fluctuations des quantités de locations - construite avec *Matlab* . . . 35
4.10 Histogramme des fluctuations des locations - construite avec *Matlab* . . . 35
4.11 Répartition des locations en fonction de l'abonnement - construite avec *SAS* . . . 36
4.12 Distribution des locations des vélos en fonction du temps pour les différents types d'abonnement - construites avec *Matlab* . . . 37
4.13 Distribution de l'utilisation du réseau en fonction des abonnements - construites avec *Matlab* . . . 38
4.14 Courbes cumulatives des locations de vélos - construites avec *Matlab* . . . 39
4.15 Histogramme des durées des locations en fonctions des trois abonnements - construites avec *Matlab* . . . 40
4.16 Distribution des locations des vélos pour les sorties de la station 4 - construit avec *Matlab* . . . 41
4.17 Distribution des locations des vélos pour les rentrées de la station 1 - construites avec *Matlab* . . . 42
4.18 Distribution des locations des vélos pour les rentrées de la station 4 - construit avec *Matlab* . . . 43

5.1 Transferts des vélos entre les stations en fonction des jours de la semaine et du week-end - construites avec *Gephi* et *Gimp* . . . 46
5.2 Transferts des vélos entre les stations en fonction des tranches horaires du lundi au vendredi - construites avec *Gephi* et *Gimp* . . . 47
5.3 Transferts des vélos entre les stations en fonction des tranches horaires du week-end - construites avec *Gephi* et *Gimp* . . . 48
5.4 Transferts des vélos entre les stations en fonction des jours pour les deux semaines de vacances de Noël 2012 - construites avec *Gephi* et *Gimp* . . . 49
5.5 Transferts des vélos entre les stations en fonction des abonnements pour les jours d'utilisation maximale et minimale - construites avec *Gephi* et *Gimp* . . . 50
5.6 Transferts des vélos entre les stations en fonction des abonnements pour les jours en semaine - construites avec *Gephi* et *Gimp* . . . 51
5.7 Transferts des vélos entre les stations en fonction des abonnements pour les jours du week-end - construites avec *Gephi* . . . 51
5.8 Transferts des vélos entre les stations le 1er mai 2012 - construite avec *Gephi* . . . 51

6.1 Dendrogramme de la méthode des centroïdes et du lien moyen appliquées à la matrice relatives aux sorties des vélos le week-end - construites avec *SAS* . . . 54
6.2 Nombre de locations en fonction en fonction des heures pour chaque station - construites avec *Matlab* . . . 54
6.3 Analyse sur les sorties des vélos de 14 à 19h en semaine - construites avec *Matlab* et *SAS* . . . 56
6.4 Dendrogramme du lien moyen pour les jours de la semaine (du lundi au vendredi) - construites avec *Matlab* . . . 57
6.5 Dendrogramme du lien moyen pour les jours du week-end - construites avec *Matlab* 58

Liste des tableaux

1.1 Exemple de table pour un graphe *Circos* - [12] 8

2.1 Exemple représentant une entrée de la seconde table avec ajout des numéros des stations de départ et d'arrivée, des jours et des mois. 15
2.2 Exemple représentant une entrée de la seconde table avec séparations des colonnes des données temporelles. 16

3.1 Exemple de table 25
3.2 Forme de la seconde table 26
3.3 Table finale de l'exemple 26

4.1 Table de contingence en fonction du type de sorties possibles 27

Appendices

Annexe A

Programmes Matlab

A.1 Nettoyage

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : September 2013                                                  %
% But : Calcule les temps de locations en secondes et supprime les       %
%       locations de moins de 60 secondes                                %
% Attention : Importer le fichier locations_Flag_oui.xls                 %
%                                                                        %
%************************************************************************%

format long e;
[m,n]=size(textdata);   % Calcul de la taille du fichier

for i=1:m-1
    % Calcul de la serie numerique de chaque date d entree et de
    % sortie
    serie(i,1)=datenum(textdata(i+1,3),'dd/mm/yy');
    serie(i,2)=datenum(textdata(i+1,8),'dd/mm/yy');

    % Calcul des temps d utilisation
    Temps_Ut_Date(i,1) = serie(i,2)-serie(i,1);
    Temps_Ut_Heure(i,1)=abs(data(i,11)-data(i,6));

    % Reexprime les différences sous la forme de vecteur
    % Car commence le 1 Janvier donc il compte un mois en trop quand la
    % difference depasse un mois
     if month(Temps_Ut_Date(i,1))~=0
         Diff(i,1:6)=[year(Temps_Ut_Date(i,1)) ...
             (month(Temps_Ut_Date(i,1))-1) day(Temps_Ut_Date(i,1)) ...
             hour(Temps_Ut_Heure(i,1)) minute(Temps_Ut_Heure(i,1)) ...
             second(Temps_Ut_Heure(i,1))];
```

```
    else
        Diff(i,1:6)=[year(Temps_Ut_Date(i,1)) month(Temps_Ut_Date(i,1)) ...
            day(Temps_Ut_Date(i,1)) hour(Temps_Ut_Heure(i,1)) ...
            minute(Temps_Ut_Heure(i,1)) second(Temps_Ut_Heure(i,1))];
    end
end

% Verification pas de location de plus de 1 mois ou 1 an car s il y en a
% il faut les supprimer
 for i=1:m-1
    if Diff(i,1)~=0
        i
    end
    if Diff(i,2)~=0
        i
    end
 end

 j=1;
 for i=1:m-1
    %Reexprime le temps en secondes
    Temps_Seconde(i,1)=(day(Temps_Ut_Date(i,1))*60*60*24)+...
        (hour(Temps_Ut_Heure(i,1))*60*60)+...
        (minute(Temps_Ut_Heure(i,1))*60)+second(Temps_Ut_Heure(i,1));

    %Suppression des locations inferieures à 60 secondes
     if Temps_Seconde(i,1)>60
         New_Diff(j,:)=Diff(i,:);
         New_Temps_Ut_Date(j,:)=Temps_Ut_Date(i,:);
         New_Temps_Ut_Heure(j,:)=Temps_Ut_Heure(i,:);
         New_Temps_Seconde(j,:)=Temps_Seconde(i,:);
         New_serie(j,:)=serie(i,:);
         New_textdata(j,:)=textdata(i+1,:);
         New_Data(j,:)=data(i,:);
         j=j+1;
     end
 end

 %Nettoyage des variables
 clear Diff
 clear Temps_Ut_Heure
 clear Temps_Ut_Date
 clear Temps_Seconde
 clear serie
 clear textdata
 clear j
```

```
clear i
clear m
clear n
clear data

data=New_Data;
textdata=New_textdata;
Diff=New_Diff;
serie=New_serie;
Temps_Ut_Heure=New_Temps_Ut_Heure;
Temps_Ut_Date=New_Temps_Ut_Date;
Temps_Seconde=New_Temps_Seconde;

% Sauvegarde des variables sous format matlab pour faciliter l utilisation
save Locations_Triees_TEST textdata data Diff serie Temps_Ut_Heure ...
    Temps_Ut_Date Temps_Seconde
```

A.2 Poids des liens

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Septembre 2013                                                  %
% But : calculer le poids de liens entre les différentes stations.       %
%                                                                        %
%************************************************************************%
clc;
close all;
clear all;
format long e;

load Locations_Triees textdata data Temps_Seconde  % Importation données
%fid = fopen('Lien_Matlab_Pourcents_test.csv','wt');
load Numero_Station   % Fichier avec les numéros des stations uniquement
[m,n]=size(textdata); % Calcul taille des données

% Insertion des numéros des stations dans la premiere ligne et la premiere
% colonne
% Sortie = colonne verticale
% Entree = colonne horizontale
tab=zeros(25,25);
for i=1:24
   tab(1,i+1)=Num(i);
   tab(i+1,1)=Num(i);
end
```

```
% Creation variable plus explicites avec les numéros des station d'entrée
% et de sortie pour chaque donnée
Num_Sortie(:,1)=data(:,1);
Num_Retour(:,1)=data(:,14);

% Boucle sur toutes les données
for i=1:m
    j=1;
    while ( j<26 && Num_Sortie(i)~=tab(j,1) )
        j=j+1;
    end
    k=1;
    while ( k<26 && Num_Retour(i)~=tab(1,k) )
        k=k+1;
    end

    if (k<26 && j<26)
        tab(j,k)=tab(j,k)+1;
    end
end

% Creation fichier txt avec la matrice pour circos
fid2 = fopen('test.txt','w');
for i=1:24
    fprintf(fid2,'%i\t',tab(i,:));
    fprintf(fid2,'\n')
end
fclose(fid2)

%Premiere colonne
for j=2:25
   for i=1:24
      New(i+((j-2)*24),1)=tab(j,1);
   end
end

%Deuxieme et 3eme colonne
 for j=1:24
     for i=2:25
         New((i-1)+((j-1)*24),2)=tab(1,i);
         New((i-1)+((j-1)*24),3)=tab(j+1,i);
     end
 end

 %Suppression des zeros
 k=0;
```

```
 for i=1:length(New)
    if New(i,3)~=0
        k=k+1;
        final(k,:)=New(i,:);
    end
 end
xlswrite('Lien_Matlab.xls',final)

Total=sum(tab(2:25,2:25))
Total=sum(Total)

final(:,3)=(final(:,3)./Total)*100;

% Ecriture dans un csv pour Gephi
if (fid == -1); return; end
fprintf(fid, '%s;%s;%s;%s;%s\n','Source','Target','Weight','SourceNom','TargetNom');
for i=1:size(final)
fprintf(fid,'%i;%i;%f;\n',final(i,:));
end
fclose(fid);
```

A.3 Rassemblement des poids

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Septembre 2013                                                  %
% But : calculer le poids de liens entre les différentes stations en     %
%       sommant les entrées et les sorties entre 2 stations, on aura plus%
%       que un lien entre deux stations.                                 %
%                                                                        %
%************************************************************************%
clc;
close all;
clear all;
format long e;

load Locations_Triees textdata data % Importation données
fid = fopen('Lien_Matlab_Pourcents_test.csv','wt');
load Numero_Station   % Fichier avec les numéros des stations uniquement
[m,n]=size(textdata); % Calcul taille des données
% Insertion des numéros des stations dans la premiere ligne et la premiere
% colonne
%Sortie = colonne verticale
%Entree = colonne horizontale
tab=zeros(25,25);
```

```
for i=1:24
   tab(1,i+1)=Num(i)
   tab(i+1,1)=Num(i)
end
% Creation variable plus explicites avec les numéros des station d'entrée
% et de sortie pour chaque donnée
Num_Sortie(:,1)=data(:,1);
Num_Retour(:,1)=data(:,14);
tab2=tab;

% Boucle sur toutes les données
for i=1:m
    j=1;
    while ( j<26 && Num_Sortie(i)~=tab(j,1) )
        j=j+1;
    end
    k=1;
    while ( k<26 && Num_Retour(i)~=tab(1,k) )
        k=k+1;
    end
    if (k<26 && j<26)
        tab(j,k)=tab(j,k)+1;
    end
end

% Somme
for i=2:25
    for j=2:25
        if (j>i)
        tab2(i,j)=tab(i,j)+tab(j,i);
        end
        if i==j
            tab2(i,j)=tab(i,j);
        end
    end
end

%Premiere colonne
for j=2:25
   for i=1:24
      New(i+((j-2)*24),1)=tab2(j,1);
   end
end

%Deuxieme et 3eme colonne
 for j=1:24
```

```
     for i=2:25
         New((i-1)+((j-1)*24),2)=tab2(1,i);
         New((i-1)+((j-1)*24),3)=tab2(j+1,i);
     end
 end

 %Suppression des zeros
 k=0;
 for i=1:length(New)
    if New(i,3)~=0
        k=k+1;
        final(k,:)=New(i,:);
    end
 end

Total=sum(tab(2:25,2:25))
Total=sum(Total)
%xlswrite('Lien_Matlab.xls',final)
final(:,3)=(final(:,3)./Total)*100;

% Ecriture dans un csv pour Gephi
if (fid == -1); return; end
fprintf(fid, '%s;%s;%s;%s;%s\n','Source','Target','Weight','SourceNom','TargetNom');
for i=1:size(final)
fprintf(fid,'%i;%i;%f;\n',final(i,:));
end
fclose(fid);
```

A.4 Histogramme des temps de locations

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Novembre 2013                                                   %
% But : Construit l histogramme des temps de locations (en secondes)     %
%       avec la moyenne, la medianne, le minimum, maximum et le mode     %
%                                                                        %
%************************************************************************%
close all;
clear all;
clc;

load Selection_Abonnements/Locations_Semaine_Triees  Temps_Seconde    % Importation des
taille=size(Temps_Seconde)   % Calcul de la taille du fichier
nbins=max(Temps_Seconde)/60  % Nombre de barres, une par minute
[n,x]=hist(Temps_Seconde,nbins)
```

```
figure
hist(Temps_Seconde,nbins,'EdgeColor','b') % Histogramme
axis([0 7200 0 700])
set(gca,'XTick',0:600:7200) % Limitation des axes à 120 minutes
set(gca,'XTickLabel',{'0','10','20','30','40','50','60','70','80','90',...
    '100','110','120'})
xlabel('Temps (en minutes)','FontSize',14)
ylabel('Nombre de vélos','FontSize',14)
title('Temps de locations des velos','FontSize',15)
hold on
% Moyenne et Medianne
plot([mean(Temps_Seconde) mean(Temps_Seconde)],[0 20000],'r','LineWidth',2)
plot([median(Temps_Seconde) median(Temps_Seconde)],[0 20000],'m',...
    'LineWidth',2)
legend('Histogramme','Moyenne','Medianne','FontSize',14)

figure
loglog(x./60,n)
axis([10^0 10^3 10^0 10^5])
title('Abonnements hebdomadaires')
%set(gca,'XTick',10^0:10^3)
%set(gca,'XTickLabel',{'10^0','10^1','10^2','10^3'})
Moyenne=mean(Temps_Seconde)
Medianne=median(Temps_Seconde)
Variance=var(Temps_Seconde)
Minimum=min(Temps_Seconde)
Maximum=max(Temps_Seconde)
Mode=mode(Temps_Seconde)
Pourcentage_plus120=0;
% Calcul du pourcentage de donnees superieures a 120 minutes
 for i=120:20521
     Pourcentage_plus120=Pourcentage_plus120+n(i);
 end
 Pourcentage_plus120/taille(1)*100
```

A.5 Distribution des locations en fonction du temps

A.5.1 Programme Principal

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Novembre 2013                                                   %
% But : Programme permettant d'évaluer la quantité de locations en       %
%       fonction des minutes par intervalles de 15, 30 minutes, 1 ou     %
%       2 heures toutes stations confondues pour les jours de semaine    %
%       et du week-end                                                   %
%                                                                        %
%************************************************************************%
clc;
close all;
clear all;

load Selection_Paques data   % Importation des donnees
[m,n]=size(data);            % Calcul taille des donnees

% Separation semaine ou week-end
% ------------------------------
bin=[0:1/(24*2):1];
j=1;                  % Initialisation des indices
k=1;
for i=1:m
    if (data(i,5)==6) ||  (data(i,5)==7)
        Week(j,:)=data(i,:);
        j=j+1;
    else
        Semaine(k,:)=data(i,:);
        k=k+1;
    end
end
%taille des vecteurs
Taille_Sem=size(Semaine);
Taille_Week=size(Week);

% Semaine
[n1,x1]=hist(Semaine(:,6),bin);
figure
plot(x1,n1/sum(n1)*100,'b')

% Week end
[n1,x1]=hist(Week(:,6),bin);
hold on
```

```
plot(x1,n1/sum(n1)*100,' r')
set(gca,'XTick',0:(1/24)*2:1)
set(gca,'XTickLabel',{'0','2','4','6','8','10','12','14','16','18','20',...
    '22','24'},'FontSize',11)
xlabel('Heures de la journée','FontSize',14)
ylabel('Pourcentages','FontSize',14)
legend('Jours de semaine','Jours du week-end','FontSize',14)
title('Utilisation en fonction des heures','FontSize',15)
hold off


% Graphe avec jours séparés
%--------------------------
% Differenciation des jours
data=Ajout_Jours(data,m);

bin15=[0:1/96:7];   % Bins pour intervalles de 15 minutes
bin30=[0:1/48:7];   % Bins pour intervalles de 30 minutes
bin1=[0:1/24:7];    % Bins pour intervalles de 1 heure
bin2=[0:1/12:7];    % Bins pour intervalles de 2 heures

% Calculs des frequences
[n15,x15]=hist(data(:,6),bin15);
[n30,x30]=hist(data(:,6),bin30);
[n1,x1]=hist(data(:,6),bin1);
[n2,x2]=hist(data(:,6),bin2);

% Graphique
figure
plot(x15,n15/sum(n15)*100,'.- m')
hold on
plot(x30,n30/sum(n30)*100,'.-')
plot(x1,n1/sum(n1)*100,'.- c')
plot(x2,n2/sum(n2)*100,'.- r')
set(gca,'XTick',0:1:7)
set(gca,'XTickLabel',{'Lundi','Mardi','Mercredi','Jeudi','Vendredi',...
    'Samedi','Dimanche'},'FontSize',11)
xlabel('Jours de semaine','FontSize',14)
ylabel('Pourcentages','FontSize',14)
legend('15 min','30 min','1 h','2 h','FontSize',14)
title('Tous types de locations et toutes stations','FontSize',15)
set(gca,'XGrid','on')
hold off
```

A.5.2 Fonction pour différencier les jours de semaine

```
function [X]=Ajout_Jours(data,m)
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Decembre 2013                                                   %
% But : Fonction qui différencie les jours de la semaine                 %
% Entree : data : donnees à analyser                                     %
%          m    : taille des données                                     %
% Sortie : X : les données auxquelles on a ajouté les jours aux fractions%
%              de journées                                               %
%                                                                        %
%************************************************************************%
for i=1:m
    if (data(i,5)==2)
        data(i,6)=data(i,6)+1;
    else if (data(i,5)==3)
        data(i,6)=data(i,6)+2;
        else if (data(i,5)==4)
                data(i,6)=data(i,6)+3;
            else if (data(i,5)==5)
                    data(i,6)=data(i,6)+4;
                else if (data(i,5)==6)
                        data(i,6)=data(i,6)+5;
                    else if (data(i,5)==7)
                            data(i,6)=data(i,6)+6;
                        end
                    end
                end
            end
        end
    end
end
X=data
end
```

A.6 Locations par rapport aux abonnements

A.6.1 Programme principal

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Novembre 2013                                                   %
% But : Programme permettant d'évaluer la quantité de locations en       %
%       fonction des minutes par intervalles de 15, 30 minutes, 1 ou     %
%       2 heures toutes stations confondues pour un certain type         %
%       d abonnement.                                                    %
%                                                                        %
%************************************************************************%
clc;
close all;
clear all;

%----------%
% Variable %
%----------%
%'Longue Durée'
%'CD - 1 Jour'
%'CD - 1 Semaine'
Texte = 'CD - 1 Semaine';

% Appel de la fonction pour selectionner les donnees.
Pourcent=Selection_Abonnements(Texte);

% Importation des donnees
if strcmp(Texte,'Longue Durée')
    load Locations_Annuelles_Triees data
else if strcmp(Texte,'CD - 1 Jour')
        load Locations_Jour_Triees data
    else if strcmp(Texte,'CD - 1 Semaine')
            load Locations_Semaine_Triees data
        end
    end
end

% Taille des donnees
[m,n]=size(data);

% Differenciation des jours
data=Ajout_Jours(data,m);

bin15=[0:1/96:7];   % Bins pour intervalles de 15 minutes
```

```
bin30=[0:1/48:7];   % Bins pour intervalles de 30 minutes
bin1=[0:1/24:7];    % Bins pour intervalles de 1 heure
bin2=[0:1/12:7];    % Bins pour intervalles de 2 heures

% Calculs des frequences
[n15,x15]=hist(data(:,6),bin15);
[n30,x30]=hist(data(:,6),bin30);
[n1,x1]=hist(data(:,6),bin1);
[n2,x2]=hist(data(:,6),bin2);

% Graphique
figure
plot(x15,n15/sum(n15)*100,'.- m')
hold on
plot(x30,n30/sum(n30)*100,'.-')
plot(x1,n1/sum(n1)*100,'.- c')
plot(x2,n2/sum(n2)*100,'.- r')
set(gca,'XTick',0:1:7)
set(gca,'XTickLabel',{'Lundi','Mardi','Mercredi','Jeudi','Vendredi',...
    'Samedi','Dimanche'},'FontSize',11)
xlabel('Jours de semaine','FontSize',14)
ylabel('Pourcentages','FontSize',14)
legend('15 min','30 min','1 h','2 h','FontSize',14)
if strcmp(Texte,'Longue Durée')
    title('Abonnements Annuels','FontSize',15)
else if strcmp(Texte,'CD - 1 Jour')
        title('Abonnements Journaliers','FontSize',15)
    else if strcmp(Texte,'CD - 1 Semaine')
            title('Abonnements Hebdomadaires','FontSize',15)
        end
    end
end
set(gca,'XGrid','on')
hold off
```

A.6.2 Fonction de sélection

```
function [Pourcent]=Selection_Abonnements(Texte)
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Decembre 2013                                                   %
% But : Fonction qui selectionne les donnees en fonction de l'abonnement %
%       choisi. Il enregistre les donnees desirees dans un fichier mat   %
% Entree : Texte : Nom de l'abonnement                                   %
% Sortie : Pourcentage : Pourcentage que représente le type d'abonnement %
%                        choisi parmis toutes les donnees                %
%                                                                        %
%************************************************************************%
% Ouverture du fichier de départ
load ../Locations_Triees data textdata Temps_Seconde serie
[m,n]=size(textdata);

j=1;
 for i=1:m
     % Comparaison entre deux variables textes
    if strcmp(textdata(i,15),Texte)
        New_textdata(j,:)=textdata(i,:);
        New_data(j,:)=data(i,:);
        New_Temps(j,:)=Temps_Seconde(i,:);
        New_serie(j,:)=serie(i,:);
        j=j+1;
    end
 end

clear textdata data Temps_Seconde serie
textdata=New_textdata;
data = New_data;
Temps_Seconde=New_Temps;
serie=New_serie;
% Enregistrement des donnees
if strcmp(Texte,'Longue Durée')
    save Locations_Annuelles_Triees data textdata Temps_Seconde serie
else if strcmp(Texte,'CD - 1 Jour')
        save Locations_Jour_Triees data textdata Temps_Seconde serie
    else if strcmp(Texte,'CD - 1 Semaine')
            save Locations_Semaine_Triees data textdata Temps_Seconde serie
        end
    end
end

% Calcul du pourcentage que representent les donnees selectionnees parmi la
```

```
% totalite des donnees
Pourcent=size(data)/m;
end
```

A.7 Utilisation du réseau en fonction des dates

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Novembre 2013                                                   %
% But : Programme permettant d'évaluer la fréquence des locations pour   %
%       chaque date du fichier de donnees                                %
%                                                                        %
%************************************************************************%
clc;
close all;
clear all;
format long e;

load Locations_Annuelles_Triees textdata   % Importe les donnees
[m,n]=size(textdata);            % Calcul la taille

% Numero de serie de la premiere date
serie(1,1)=datenum(textdata(1,16),'dd/mm/yy HH:MM:SS');
for i=2:m
   serie(i,1)=datenum(textdata(i,16),'dd/mm/yy HH:MM:SS');
   %Difference par rapport au premier
   Diff(i-1,1)=serie(i,1)-serie(1,1);
end

% Histogramme
%-------------
bin=[0:1:363];
[n,x]=hist(Diff,bin); %n= nbr de velos
bar(x,n/sum(n)*100)
%set(gca,'XTick',0:10:365)
axis([-1 363 0 0.6])
xlabel('Jours - à partir du 21/04/2012','FontSize',12)
ylabel('Pourcentages de vélos utilisés','FontSize',12)
title('Utilisation du réseau en fonction des jours','FontSize',13);

% Courbe cumulative
%------------------
figure
B=cumsum(n);
plot(x,B);
```

```
xlabel('Jours - à partir du 21/04/2013','FontSize',12)
ylabel('Nombre de vélos utilisés','FontSize',12)
title('Courbe cumulative des locations','FontSize',13)

Temp=find(x<230); % On prend tous les jours précédents le 230 ième
% Retire les valeurs des p aux n correspondant et voir les difference et analyser avec l

figure
% Premiere periode
p = polyfit(x(Temp),B(Temp),1)
plot(x,B)  % Courbe cumulative de départ
hold on
plot(x(Temp),p(1)*x(Temp)+p(2),'c --');

% Seconde periode
p2= polyfit(x(Temp(end)+20:end),B(Temp(end)+20:end),1)
plot(x(Temp(end)+20:end),p2(1)*x(Temp(end)+20:end)+p2(2),'r --');
legend('Courbe Cum.','x<230','x>230','NorthWest')
xlabel('Jours - à partir du 21/04/2013','FontSize',12)
ylabel('Nombre de vélos utilisés','FontSize',12)
title('Polyfit - Courbe cumulative des locations','FontSize',13)
hold off

% Fluctuations
%--------------
% Soustraction du taux moyen de locations pour la premiere periode
Temp1=n(Temp)-p(1);
% Soustraction du taux moyen de locations pour la seconde periode
Temp2=n(Temp(end)+20:end)-p2(1);
figure
bar(x(Temp),Temp1)
hold on
bar(x(Temp(end)+20:end),Temp2)
xlabel('Jours - a partir du 21/04/2013','FontSize',12)
ylabel('Nombre de vélos utilisés - moyennes','FontSize',12)
title('Fluctuations des locations','FontSize',13)
hold off

% Maximum et minimum
[y,ii]=max(Temp1)
[y2,ii2]=max(Temp2)
[y3,ii3]=min(Temp1)
[y4,ii4]=min(Temp2)

% Histogrammes des fluctuations
%------------------------------
```

```
 figure
 hist([Temp1 Temp2],-250:10:250)
xlabel('Nombre de vélos utilisés - moyennes','FontSize',12)
ylabel('Nombre de fluctuations','FontSize',12)
title('Histogramme des fluctuations','FontSize',13)
```

A.8 Sélection en fonction des données temporelles

A.8.1 Programme principal

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date   : Septembre 2013                                                %
% But    : Programme principal qui permet de créé un fichier contenant   %
%          des donnees sélectionnées suivant les mois, les jours ou les  %
%          heures                                                        %
%                                                                        %
%************************************************************************%
clc;
clear all;
close all;
format long e;

load Locations_Triees textdata data Temps_Seconde serie  % Importation

% Selection par rapport à des dates ou des mois.
%-----------------------------------------------
% Dates entre lesquels doivent se situer les donnees
borne1=datenum('01/12/2012','dd/mm/yy');
borne2=datenum('15/12/2012','dd/mm/yy');
% Appel de la fonction
[New_textdata,New_Data,New_Temps_Seconde]=Selection_Date(textdata,...
    data,Temps_Seconde,serie,borne1,borne2);

% Selection par rapport a des jours de la semaine
%-----------------------------------------------
Jour=[6 7];    % Jours désirés
% Appel de la fonction
%[New_textdata,New_Data,New_Temps_Seconde]=Selection_Jour(textdata,...
    %data,Temps_Seconde,Jour);

%Selection par rapport à des heures
%---------------------------------
% heures entre lesquelles doivent se situer les donnees
borne3=datenum('00:00:00','HH:MM:SS');
```

```
borne4=datenum('06:00:00','HH:MM:SS');
% Appel de la fonction
%[New_textdata,New_Data,New_Temps_Seconde]=Selection_Heure(textdata,...
    %data,Temps_Seconde,borne3,borne4);

clear textdata data Temps_Seconde

textdata=New_textdata;
data=New_Data;
Temps_Seconde=New_Temps_Seconde;

% Sauvegarde des donnees
save Selection textdata data Temps_Seconde
```

A.8.2 Sélection entre deux dates

```
function [New_textdata,New_Data,New_Temps_Seconde]=Selection_Date(textdata,...
data,Temps_Seconde,serie,borne1,borne2)
%************************************************************************%
% Author : Dehan Laetitia                                                 %
% Date   : Septembre 2013                                                 %
% But    : Fonction qui Sert a selectionner les donnees comprises entre   %
%          deux dates choisies uniquement par rapport à la date de sortie%
% Entree : textdata & data : donnees de depart                            %
%          Temps_secondes  : durée des locations du fichier de départ en %
%                            secondes                                     %
%          serie           : serial date number des données de départ    %
%          borne1 & borne2 : bornes entre lesquels on veut sélectionner  %
%                            les donneées                                 %
% Sortie : New_textdata,New_Data & New_Temps_Seconde : contiennent les    %
%                            donnees souhaitées                           %
%************************************************************************%

[m,n]=size(textdata);   % Calcul de la taille des donnees

j=1;
for i=1:m
    % On verie si la date est comprises entre celles choisies
    if  (serie(i,1)>=borne1) && (serie(i,1)<=borne2)
        New_textdata(j,:)=textdata(i,:);
        New_Data(j,:)=data(i,:);
        New_Temps_Seconde(j,:)=Temps_Seconde(i,:);
        j=j+1;
    end
end
```

```
% Dans le cas où les nouvelles variables sont vides
if j==1
    New_textdata=[];
    New_Data=0;
    New_Temps_Seconde=0;
end
end
```

A.8.3 Sélection entre deux heures

```
function [New_textdata,New_Data,New_Temps_Seconde]=Selection_Heure(textdata,...
data,Temps_Seconde,borne1,borne2)
%*********************************************************************%
% Author : Dehan Laetitia                                             %
% Date   : Septembre 2013                                             %
% But    : Fonction qui sert a selectionner les donnees comprises entre  %
%          deux heures choisies uniquement par rapport à l'heure de      %
%          sortie                                                        %
% Entree : textdata & data : donnees de depart                         %
%          Temps_secondes  : durée des locations du fichier de départ en %
%                            secondes                                    %
%          serie           : serial date number des données de départ    %
%          borne1 & borne2 : bornes entre lesquels on veut sélectionner  %
%                            les donneées                                %
% Sortie : New_textdata,New_Data & New_Temps_Seconde : contiennent les   %
%                            donnees souhaitées                          %
%*********************************************************************%
[m,n]=size(textdata);  % Calcul de la taille des donnees

j=1;
for i=1:m
   %Calcul de la serie numerique de chaque heure d'entrees
    Temp=datenum(textdata(i,7),'HH:MM:SS');
   if  (Temp>=borne1) && (Temp<borne2)
       %On garde les lignes de textdata dont les heures sont comprises
       % entre celles choisies.
        New_textdata(j,:)=textdata(i,:);
        New_Data(j,:)=data(i,:);
        New_Temps_Seconde(j,:)=Temps_Seconde(i,:);
        j=j+1;
   end
end

% Dans le cas où les nouvelles variables sont vides
if j==1
    New_textdata=[];
```

```
    New_Data=0;
    New_Temps_Seconde=0;
end
end
```

A.8.4 Sélection par rapport à un ou plusieurs jours de semaine

```
function [New_textdata,New_Data,New_Temps_Seconde]=Selection_Jour(textdata,...
data,Temps_Seconde,Jour)
%*********************************************************************%
% Author : Dehan Laetitia                                             %
% Date   : Septembre 2013                                             %
% But    : Fonction qui sert a selectionner les donnees par rapport au   %
%          jour de la semaine, sélection uniquement par rapport au jour  %
%          de sortie                                                  %
% Entree : textdata & data : donnees de depart                        %
%          Temps_secondes  : durée des locations du fichier de départ en %
%                            secondes                                 %
%          Jour : jour pendant lesquels on veut sélectionner les données %
% Sortie : New_textdata,New_Data & New_Temps_Seconde : contiennent les   %
%                            donnees souhaitées                       %
%*********************************************************************%
[m,n]=size(textdata); % Calcul de la taille des donnees
[o,p]=size(Jour)
j=1;
for i=1:m
    % Verification du jour
    for k=1:p
        if  (data(i,5)==Jour(k))
            New_textdata(j,:)=textdata(i,:);
            New_Data(j,:)=data(i,:);
            New_Temps_Seconde(j,:)=Temps_Seconde(i,:);
            j=j+1;
        end
    end
end
% Dans le cas où les nouvelles variables sont vides
if j==1
    New_textdata=[];
    New_Data=0;
    New_Temps_Seconde=0;
end
end
```

A.9 Sélection par rapport à la borne de sortie ou d'entrée

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Novembre 2013                                                   %
% But : Sélectionne les données en fonction de la borne de sortie ou     %
%       d'entrée. Stocke dans fichier .mat propre à chaque borne         %
%                                                                        %
%************************************************************************%
clc;
clear all;
close all;
format long e;

load ../Locations_Triees data textdata   % Importation des données
load ../Numero_Station.mat Num % Vecteur avec les numéros des bornes
Num(11)=[]; % Suppression de la station 15
borne=Num';
% Calculs des tailles
[m,n]=size(textdata);
taille_borne=size(borne);

% Selection
% ---------
for ii=1:taille_borne(2)   % Boucle sur chaque numéro de borne
    j=1;
    k=1;
    for i=1:m              % Boucle sur chaque ligne des données de départ
        % par rapport à la borne de sortie
        if data(i,1)==borne(ii)
            textdata_Sort(j,:)=textdata(i,:);
            data_Sort(j,:)=data(i,:);
            j=j+1;
        end

        %par rapport à la borne de retour
        if data(i,14)==borne(ii)
            textdata_Rentr(k,:)=textdata(i,:);
            data_Rentr(k,:)=data(i,:);
            k=k+1;
        end
    end
    clear i j k
    % Enregistre les sorties
    nom_fich=strcat('Borne',num2str(borne(ii)),'_Sorties.mat');
```

```
    save(nom_fich,'data_Sort','textdata_Sort','-mat') % Enregistrement donnees
     % Enregistre les entrées
    nom_fich=strcat('Borne',num2str(borne(ii)),'_Rentrees.mat');
    save(nom_fich,'data_Rentr','textdata_Rentr','-mat') % Enregistrement donnees
    clear textdata_Sort data_Sort textdata_Rentr data_Rentr
end
 clear m n ii
```

A.10 Courbes d'utilisation des vélos pour chaque station

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Novembre 2013                                                   %
% But : Programme permettant d'évaluer la quantité de locations en       %
%       fonction des des minutes par intervalles de 15, 30 minutes, 1 ou %
%       2 h pour les jours de semaine et le week-end pour les rentrées   %
%       Ce programme construit également la matrice pour la              %
%       classification par rapport à la station de rentrée, jours de     %
%       semaine et week-end                                              %
%                                                                        %
%************************************************************************%
clc;
close all;
clear all;

load Numero_Station       % Importation des numeros de stations
Num(11)=[];               % Suppression de la station 15

%Initialisation des matrices
tab_Semaine=zeros(23,5);
tab_Week=zeros(23,5);
for i=1:23
   tab_Semaine(i,1)=Num(i);
   tab_Week(i,1)=Num(i);
end

% Boucle pour appliquer le programme a toutes les stations
%for kk=1:23
kk=1
    nom_fich=strcat('Borne',num2str(Num(kk)),'_Rentrees');
    load(nom_fich,'-mat') % Importation du fichier de donnees

    [m,n]=size(data_Rentr);     % Taille du fichier de donnees

    % Separation semaine ou week-end
```

```
% ----------------------------
j=1;                  % Initialisation des indices
k=1;
for i=1:m
    if (data_Rentr(i,10)==6) ||  (data_Rentr(i,10)==7)
        Week(j,:)=data_Rentr(i,:);
        j=j+1;
    else
        Semaine(k,:)=data_Rentr(i,:);
        k=k+1;
    end
end

%taille des vecteurs
Taille_Sem=size(Semaine);
Taille_Week=size(Week);

% Construction des graphes
% ------------------------
bin=[0,6/24,10/24,15/24,20/24,1]; % Intervalles de temps
bin15=[0:1/96:1];               % Intervalle de 15 minutes
bin30=[0:1/48:1];               % Intervalle de 30 minutes
bin1=[0:1/24:1];                % Intervalle d 1 heure
bin2=[0:1/12:1];                % Intervalle de 2 heures

% Semaine
% -------
[n15,x15]=hist(Semaine(:,11),bin15);
[n30,x30]=hist(Semaine(:,11),bin30);
[n1,x1]=hist(Semaine(:,11),bin1);
[n2,x2]=hist(Semaine(:,11),bin2);

% Construction histogramme
for i=1:5
    % Regarde dans quel intervalle la donnée se trouve
    ii=find(bin(i)<=x1 & x1<bin(i+1));
    N(i)=sum(n1(ii));
end
tab_Semaine(kk,2:5)=N(2:5);     % Ajout dans la matrice
% Changement du titre de la figure en fonction de la station
Titre=strcat('Jours de semaine - Station ',num2str(Num(kk)));

h=figure('Units','Normalized','Outerposition',[0 0 1 1]);
subplot(2,2,1)
bar(N/sum(N)*100)                   % Histogramme
xlabel('Tranches Horaires','FontSize',12)
```

```
ylabel('Pourcentages de vélos utilisés','FontSize',12)
set(gca,'XTickLabel',{'0-6h','6-10h','10-15h','15-20h','20-24h'})
title(Titre,'FontSize',15)
% Construction courbes
subplot(2,2,3)
plot(x15,n15/sum(n15)*100,'.- m')
hold on
plot(x30,n30/sum(n30)*100,'.-')
plot(x1,n1/sum(n1)*100,'.- c')
plot(x2,n2/sum(n2)*100,'.- r')
set(gca,'XTick',0:(1/24)*4:1)
set(gca,'XTickLabel',{'0','4','8','12','16','20','24'})
xlabel('Heures de la journée','FontSize',12)
ylabel('Pourcentages de vélos utilisés','FontSize',12)
legend('15 min','30 min','1 h','2 h')
title('Jours de Semaine','FontSize',14)
hold off

% Week-end
% --------
[n15,x15]=hist(Week(:,11),bin15);
[n30,x30]=hist(Week(:,11),bin30);
[n1,x1]=hist(Week(:,11),bin1);
[n2,x2]=hist(Week(:,11),bin2);

for i=1:5
    % Regarde dans quel intervalle la donnée se trouve
    ii=find(bin(i)<=x1 & x1<bin(i+1));
    N(i)=sum(n1(ii));
end
tab_Week(kk,2:5)=N(2:5);     % Ajout dans la matrice

subplot(2,2,2)
bar(N/sum(N)*100)                % Histogramme
set(gca,'XTickLabel',{'0-6h','6-10h','10-15h','15-20h','20-24h'})
xlabel('Tranches Horaires','FontSize',12)
 ylabel('Pourcentages de vélos utilisés','FontSize',12)
title('Jours du Week-end','FontSize',15)
% Construction courbes
subplot(2,2,4)
plot(x15,n15/sum(n15)*100,'.- m')
hold on
plot(x30,n30/sum(n30)*100,'.-')
plot(x1,n1/sum(n1)*100,'.- c')
plot(x2,n2/sum(n2)*100,'.- r')
set(gca,'XTick',0:(1/24)*4:1)
```

```
    set(gca,'XTickLabel',{'0','4','8','12','16','20','24'})
    xlabel('Heures de la journée','FontSize',12)
     ylabel('Pourcentages de vélos utilisés','FontSize',12)
    legend('15 min','30 min','1 h','2 h')
    title('Jours du Week-end','FontSize',14)
    hold off
    % Sauvegarde de la figure
    saveName = (['Stations_Entrees', num2str(Num(kk))]);
    saveas(h, saveName, 'fig');
end
% Sauvegarde de la matrice
save Matrice_Entrees tab_Semaine tab_Week
```

A.11 Similarité cosinus

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Mai 2014                                                        %
% But : Programme permettant de calculer la matrice de similarité cosinus%
%                                                                        %
%************************************************************************%
clc;
clear all;
close all;
load Sorties/Classif_Sorties % Importation des données
A=tab_Semaine(:,2:end)
D=pdist(A(:,1:8),'cosine'); % selection de la tranche horaire et calcul
% des 1-cos(theta)
Z=squareform(D);            % Mise sous la forme de matrice
h=figure('Units','Normalized');
pcolor(Z);
set(gca,'XTick',1:1:22)
set(gca,'XTickLabel',{'1','2','3','4','6',' 7',' 10',' 11',' 12','13',...
    '16','17','18','21','22','23','24','26','27','28','29','32','34'})
set(gca,'YTick',1:1:22)
set(gca,'YTickLabel',{'1','2','3','4','6',' 7',' 10',' 11',' 12','13',...
    '16','17','18','21','22','23','24','26','27','28','29','32','34'})
xlabel('Stations','FontSize',12)
ylabel('Stations','FontSize',12)
title('Similarité cosinus','FontSize',14)
colorbar
% Enregistrement des la figure
saveName = (['Semaine_08h']);
saveas(h, saveName, 'png');
```

A.12 Calcul des distances entre les stations

```
%************************************************************************%
%                                                                        %
% Author : Dehan Laetitia                                                %
% Date : Mais 2014                                                       %
% But : Programme permettant de calculer la matrice des distances entre  %
%       les différentes stations du réseau                               %
%Attention : Importee le xls avec les coordonnées GPS                    %
%************************************************************************%
[n,m]=size(data);
dist=zeros(n,n); % Initialisationd de la matrice
for i=1:n
    for j=1:n
        lat1=data(i,6);
        lat2=data(j,6);
        lon1=data(i,7);
        lon2=data(j,7);
        if (lat1 == lat2) && ( lon1 == lon2)
            dist(i,j)=0
        else
        rlat1 = (pi * lat1)/180.0;
        rlat2 = (pi * lat2)/180.0;
        rlon1 = (pi * lon1)/180.0;
        rlon2 = (pi * lon2)/180.0;

        theta = lon1-lon2;
        rtheta = pi * theta/180.0;

        dist_Temps = sin(rlat1) * sin(rlat2) + cos(rlat1) * cos(rlat2) * cos(rtheta);
        dist_Temps = acos(dist_Temps);
        dist_Temps = dist_Temps * 180/pi;
        dist_Temps = dist_Temps * 60 * 1.1515;

        dist_Temps = dist_Temps * 1.609344;
        dist(i,j)=dist_Temps;
        end
    end
end
```