Projet d'apprentissage non supervisé

Mathieu Pont et Lucas Rodrigues Pereira

Novembre 2019

Abstract

Nous analysons l'occupation des parkings de Birmingham dans l'objectif d'en tirer des profils principaux. Une telle classification aidera la prédiction de l'occupation des parkings et facilitera ainsi leur utilisation. Nous retrouvons principalement 3 profils se différenciant par leur activité la semaine et le week-end.

1 Introduction

Dans le cadre du Master 2 "Machine Learning for Data Science" (promotion 2019/2020) de l'université de Paris il nous a été demandé, lors du cours "Apprentissage non supervisé", d'étudier un jeu de données en utilisant les méthodes vues en cours.

Le jeu de données concerne le taux d'occupation des parkings de la ville de Birmingham. Nous avons des mesures pour 30 parkings différents, pour chaque mesure nous avons l'heure à laquelle elle a été prise ainsi que le nombre de places occupées pour le parking en question à ce moment.

2 Etude préliminaire du jeu de données

Le jeu de données de base est de dimension 35717×4. Chaque échantillon correspond à une mesure décrite par 4 variables, un nom (pour désigner le parking), la capacité totale du parking, le nombre de places occupées au moment de la mesure et enfin la date (avec heure) de la mesure.

En théorie, pour chaque parking les mesures vont du 4 Octobre au 19 Décembre pour l'année 2016 et pour chaque jour, les mesures sont espacées de 30 minutes de 8h à 16h30. Cela fait un total de 77 jours (11 semaines) et 18 mesures par jour donc 1386 mesures par parking. Malheureusement, certaines valeurs sont manquantes et nous n'avons donc pas toutes les mesures de chaque parking.

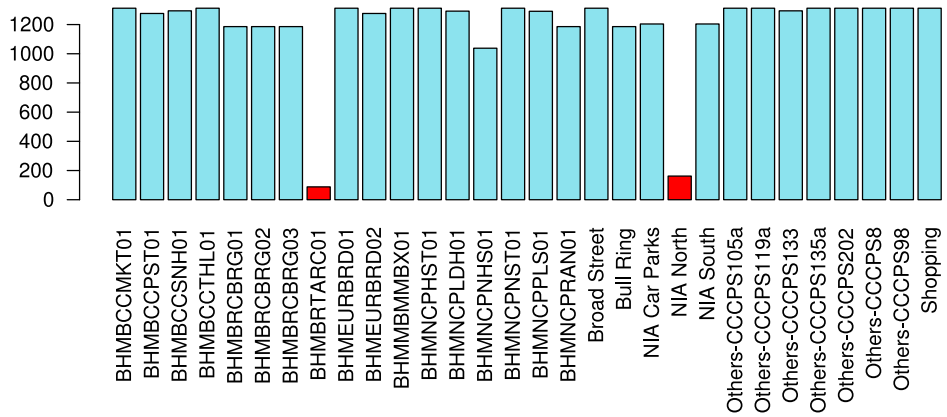


Figure 1: Nombre de mesures par parking.

Nettoyage et complétion des données. Nous voyons sur la figure 1 le nombre de mesures par parking. La distribution semble plus ou moins homogène, mais nous remarquons que le nombre de mesures est très faible pour deux parkings du jeu de données ("BHMBRTARC01" et "NIA North"). Nous avons donc décidé d'enlever ces deux parkings et de travailler sur les 28 restants.

Certaines valeurs semblent aberrantes ou érronées:

- Pour 373 mesures (échantillons) le nombre de places disponibles est supérieur à la capacité totale du parking, pour corriger ce problème nous ramenons la valeur aberrante/érronée à la capacité totale.
- Pour 12 mesures, le nombre de places disponibles est négatif, nous ramenons donc à 0 ces valeurs.
- 216 mesures sont des doublons et sont tout simplement retirées.

D'autres valeurs sont manquantes:

- Certains jours n'ont pas toutes leurs mesures. Pour combler les mesures manquantes nous faisons une moyenne du taux d'occupation des mesures avant et après. Si la mesure est au début ou en fin de journée nous répétons la mesure suivante ou précédente respectivement.
- Certains jours n'ont aucune mesure. Pour combler les jours manquants nous faisons une moyenne sur les mêmes jours du parking dans les autres semaines (par exemple si il nous manque un Jeudi, nous regardons la moyenne sur les autres Jeudis du parking pour chaque heure).

Pour trouver les mesures manquantes nous avons arrondi toutes les dates à la demi-heure près afin de faciliter la recherche desquelles nous manquaient exactement.

Après toutes ces étapes nous arrivons à un total de 38808 mesures (28 parkings, 77 jours par parking, 18 mesures par jour).

Jeux de données. Nous pouvons ensuite créer trois types de jeu de données différents:

- **Jeu de données des jours**: Chaque échantillon représente les mesures d'un jour. Dimension: 2156×18 ($28 \times 77 = 2156$ jours au total).
- **Jeu de données des semaines**: Chaque échantillon représente les mesures d'une semaine. Dimension: 308×126 ($28 \times 11 = 308$ semaines au total et $18 \times 7 = 126$ mesures par semaine).
- **Jeu de données des parkings**: Chaque échantillon représente les mesures d'un parking. Dimension: 28×1386.

Normalisation. Chacun de ces jeux de données peut être considéré comme une table de contingence croisant les mesures à certaines heures et les jours, semaines ou parkings respectivement pour chaque jeu. Cette propriété peut être exploitée en utilisant le critère du χ^2 qui est adapté pour les tables de contingence. Il permet de mesurer la différence entre la valeur observée liant les deux évènements et la valeur que l'on aurait si il y avait indépendance des évènements.

De ce critère nous pouvons en dériver une distance entre deux échantillons, pour une matrice $n \times p$ la distance du χ^2 peut être écrite:

$$d_{\chi^2}(i, i') = \sum_{j=1}^{p} \left(\frac{x_{ij}}{\sqrt{x_{.j}} x_{i.}} - \frac{x_{i'j}}{\sqrt{x_{.j}} x_{i'.}} \right)^2 \tag{1}$$

Nous pouvons, à la place d'utiliser cette distance, normaliser (en divisant) par $\sqrt{x_{.j}} x_{i.}$ pour chaque x_{ij} et ensuite utiliser la distance euclidienne. La normalisation a été choisie pour simplicité au niveau des expériences car les librairies utilisées n'offraient pas la possibilité d'utiliser cette distance.

3 Analyse descriptive

Dans un premier temps nous avons regardé la distribution des capacités de chaque parking afin de savoir si elle est homogène ou non.

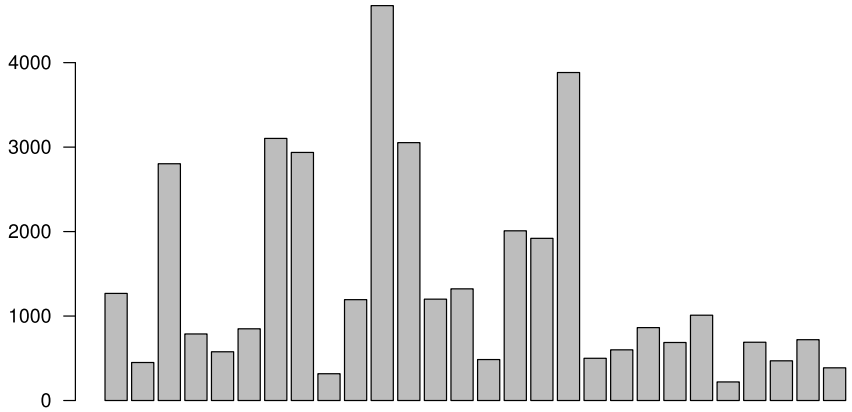


Figure 2: Distributions des capacités de chaque parking.

Du à la non homogéneité de la capacité des parkings, une normalisation par la capacité sera nécéssaire à certaines étapes afin de les mettre sur un même pied d'égalité.

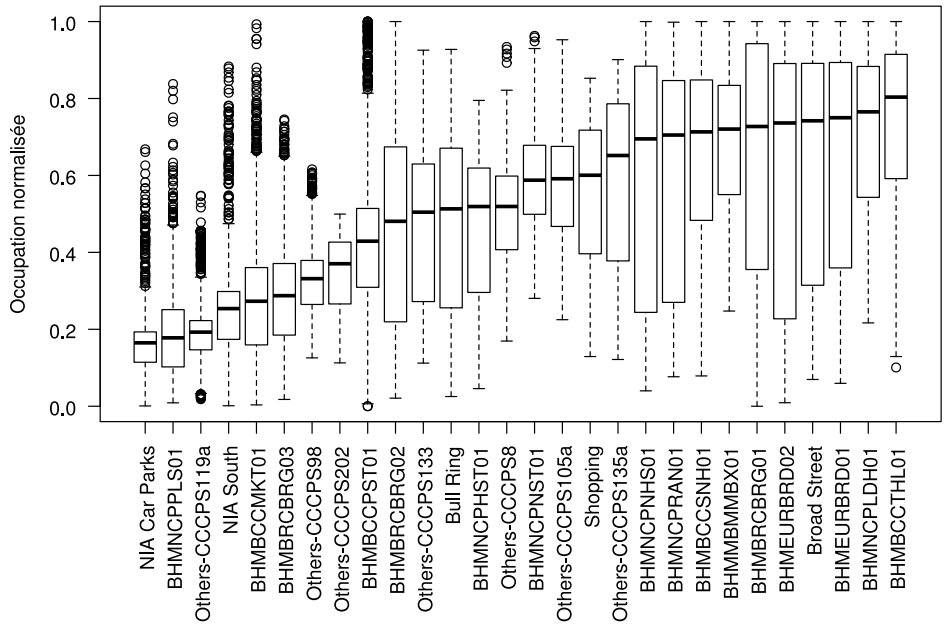


Figure 3: Box plot de l'occupation normalisée de chaque parking.

Sur la figure 3 nous avons affiché un boxplot pour chaque parking sur toutes les mesures (normalisées par la capacité) de celui-ci. Les boxplots ont été triés par la médiane et l'ordre des parkings de la figure 2 est le même. Nous pouvons déjà commencer à voir quelques groupes, on pourrait dire 2 ou 3 qui se distincterait par leur taux médian d'occupance.

Nous nous sommes ensuite intéressés à la manière dont les mesures évoluent de manière générale au long d'une journée. Pour prendre en compte tous les parkings nous avons normalisés les échantillons sur le **jeu de données des jours** (la normalisation est importante ici car tous les parkings n'ont pas la même capacité). Ici nous ne nous intéressons pas en détail à comment les parkings sont occupés les uns par rapport aux autres mais plutot comment leur occupation évolue de manière générale lors d'une journée.

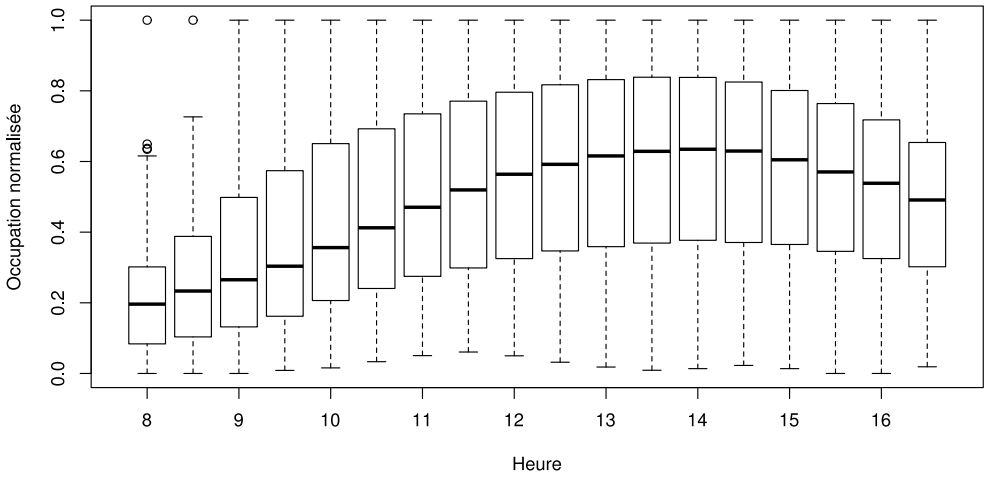


Figure 4: Evolution générale des mesures au long d'une journée.

Nous remarquons que, de manière générale, les parkings sont peu occupés en début de journée, l'occupation grimpe ensuite jusqu'à attendre un pic aux alentours de 14h pour ensuite redescendre en fin de journée.

4 Analyse du comportement hebdomadaire

4.1 Analyse préliminaire

Nous pouvons faire le même type de graphique que la figure 4 mais sur le **jeu de données des semaines** et ainsi voir comment évolue de manière générale les mesures tout au long d'une semaine pour tous les parkings.

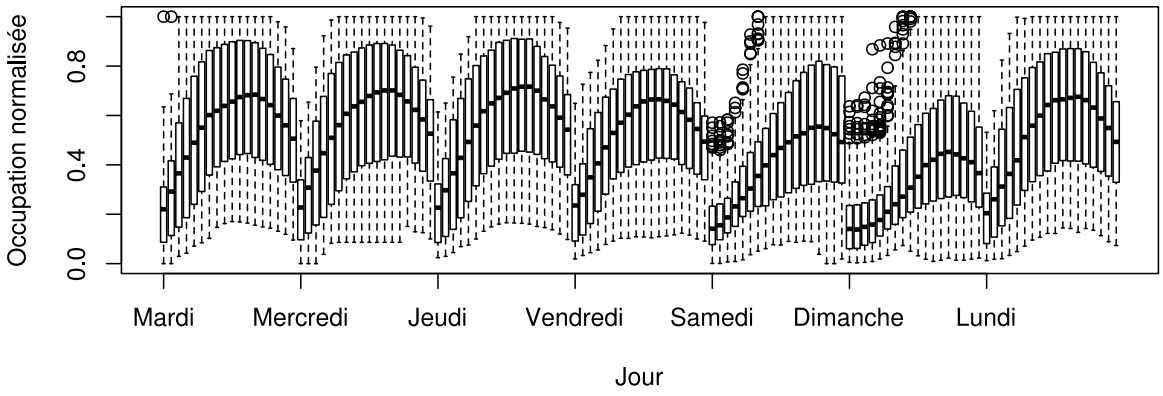


Figure 5: Evolution générale des mesures au long d'une semaine.

Nous remarquons, que de manière générale, l'évolution de l'occupation des parkings semble plus ou moins la même chaque jour du Lundi au Vendredi (et correspond plus ou moins à la figure 4). Comme dans [Stolfi et al., 2017] nous voyons que les parkings semblent être moins utilisés le week-end que les autres jours de la semaine. Le premier jour du graphique est Mardi car c'est le jour de la première date disponible (4 Octobre 2016).

4.2 Classification

Dans un premier temps, pour executer les algorithmes de classification, nous devons trouver un nombre de classe k pertinent pour notre problème. Plusieurs méthodes existent, notamment la méthode du coude consistant à executer l'algorithme pour plusieurs k et afficher l'inertie intra-classe pour chacun d'eux.

Nous avons aussi utilisé la librairie *NbClust* de R qui calcule une trentaine de critères ayant pour objectif de trouver le nombre de classes, ils tous de près ou de loin basés sur l'inertie. 15 critères ont répondu 3 classes, 8 pour 2 classes et enfin 4 pour 6 classes. Nous avons aussi réalisé la méthode du coude avec K-Medoids et K-Means et le nombre de classe adéquat semble être 3 avec cette méthode.

4.2.1 K-Medoids

En se basant sur le nombre de clusters indiqué par *NbClust*, nous lancions l'algorithme K-Medoids avec 3 classes pour le classement hebdomadaire (normalisée par la capacité et le χ^2). La distance est basée sur les ondelettes.

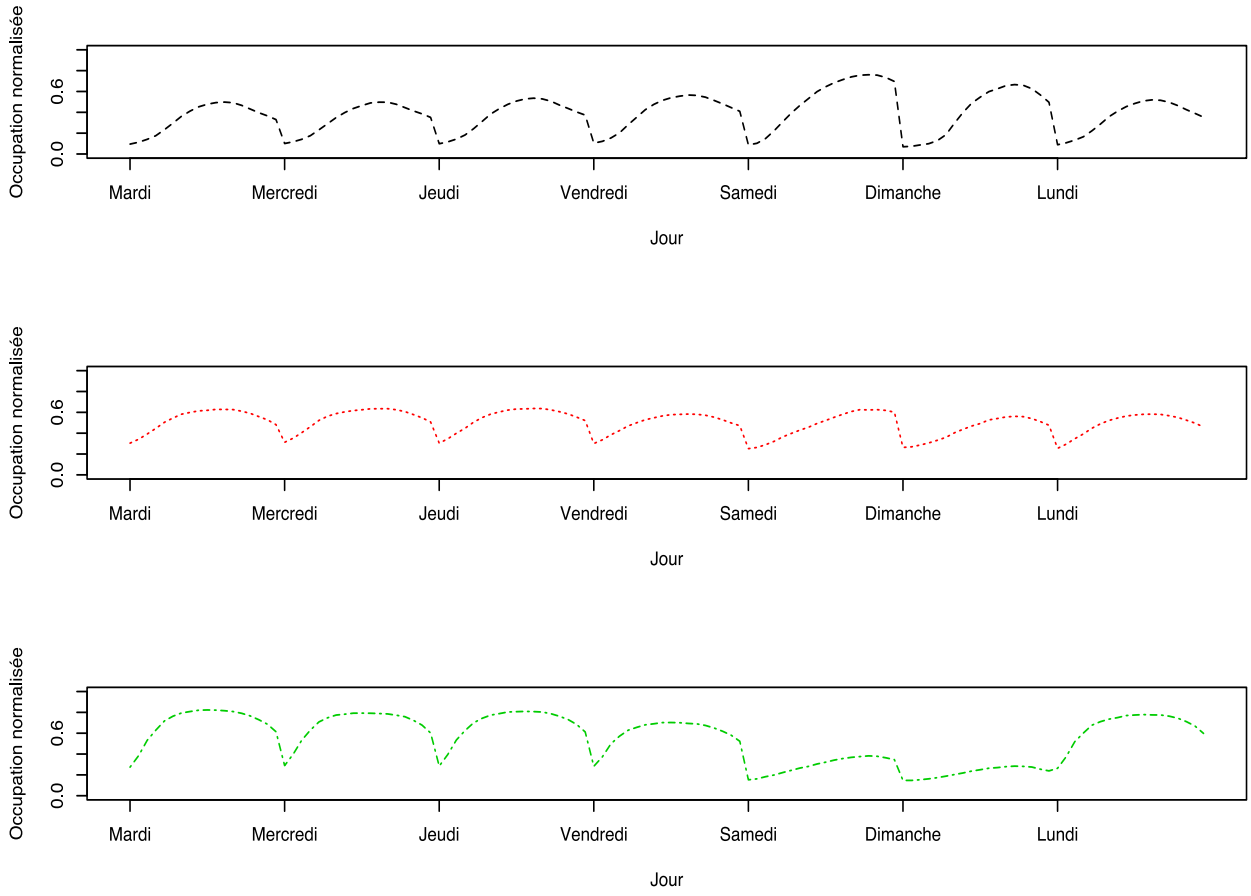


Figure 6: Boxplots pour chaque mesure de chaque semaine pour les 3 classes trouvés par K-Medoids (χ^2).

En analisant les classes trouvées par K-Medoids, nous observons que la différence se situe surtout dans les week-ends. On peut diviser les groupes par rapport aux critères suivants:

- Cluster noir (haut): activité moyenne au long de la semaine et forte le week-end.
- Cluster rouge (milieu): activité moyenne et constante au long de la semaine.
- Cluster vert (bas): faible activité le week-end et forte en semaine.

Le tableau suivant nous permet d'évaluer l'homogénéité des classes. Nous avons aussi ajouté le nombre de parkings dont les semaines affectées par chaque groupe appartient.

	Cluster haut	Cluster milieu	Cluster bas
Echantillons	99	91	118
Parkings concernés	12	11	13

Table 1: Caractéristiques par cluster.

Nous voyons que les classes ont une taille homogène bien que le dernier cluster (faible activité le week-end) semble prédominant. Nous voyons que la somme des parkings concernés ne correspond pas à notre nombre de parking (28), cela veut donc dire que certains parkings ont des semaines ayant des comportements pouvant se classer dans 2 clusters différents.

4.2.2 Self-Organizing Map

En utilisant la matrice normalisée par la capacité et le χ^2 nous avons exécuté une carte auto adaptative afin d'essayer de voir des groupes parmi les semaines de chaque parking. Nous utilisons une carte hexagonale de taille 10×10.

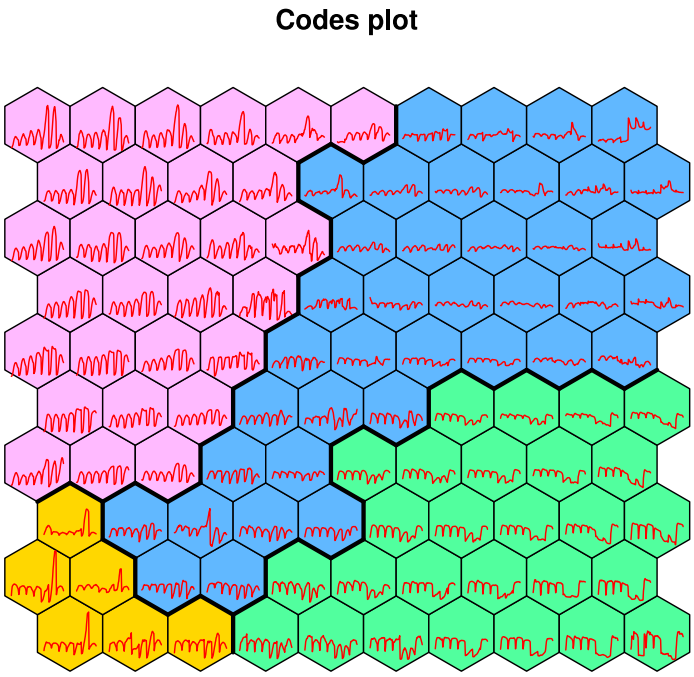


Figure 7: Résultat de la carte auto adaptative.

Nous avons regroupé les noeuds de la carte en groupe grâce à K-Means. Nous avons choisi 4 classes car nous avons pu déceler sur la carte, en amont de K-Means, 4 groupes prédominants. Les figures 7 et 8 nous ont aidées à établir la description de chaque cluster:

- Cluster rose (haut gauche): forte activité le week-end et moyenne en semaine.
- Cluster jaune (bas gauche): moyenne activité le dimanche et faible les autres jours.
- Cluster vert (bas droite): faible activité le week-end et forte en semaine.
- Cluster bleu (haut droite): activité moyenne et constante au long de la semaine.

Les clusters rose (haut gauche) et jaune (bas gauche) pourraient être rassemblés en un seul vu leur similarité, dans ce cas nous retombons sur les 3 classes proposées par *NbClust* et la méthode du coude . Mais il est tout de même intéressant de les analyser mis à part.

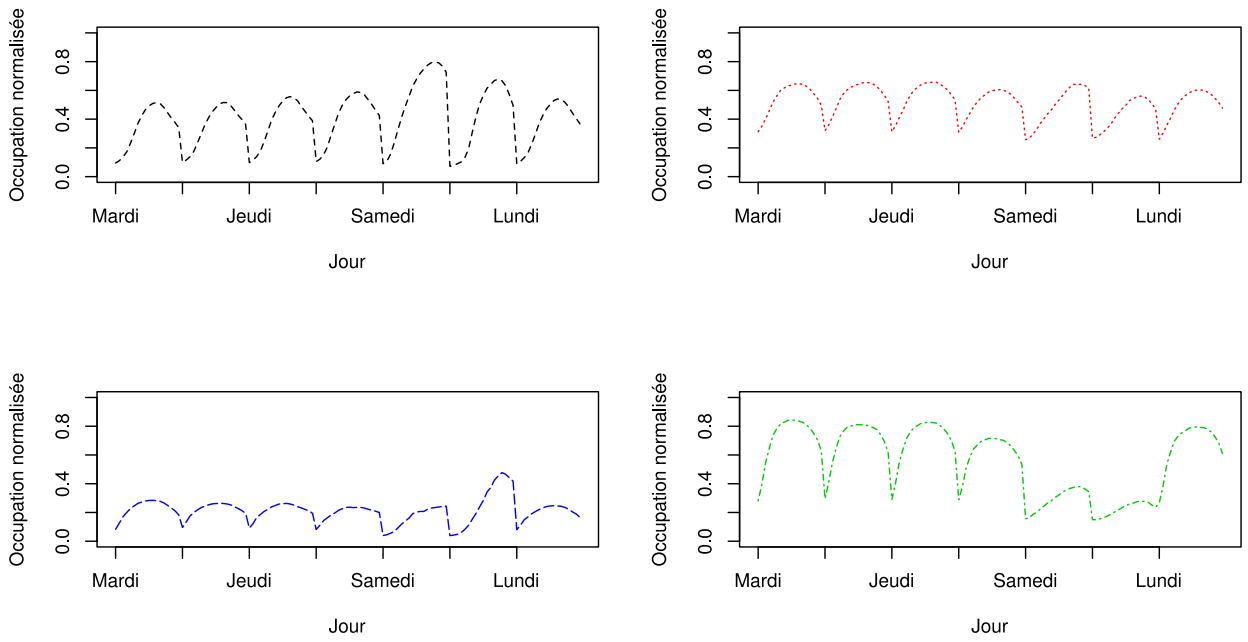
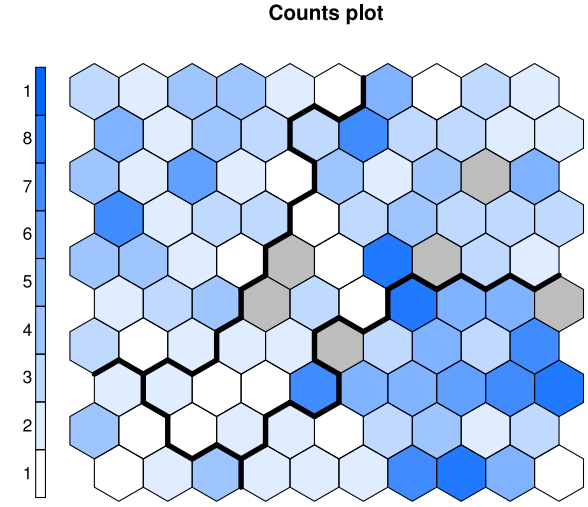


Figure 8: Courbe moyenne des mesures de la semaine pour chaque cluster.

Nous voyons sur la figure 8 la courbe moyenne pour chaque cluster et nous retrouvons les descriptions décrites dans les 4 points plus hauts. Il est à noté que, contrairement au clustering, pour l'affichage des courbes nous utilisons uniquement le jeu de données normalisée par la capacité (et non par le χ^2) afin de mieux interpréter les résultats.



Cluster	Echantillons	Parkings concernés
Haut gauche	86	11
Bas gauche	14	5
Haut droite	99	15
Bas droite	109	13

Table 2: Caractéristiques par cluster.

Figure 9: Densité de chaque noeud de la carte.

Dans la table 2 nous voyons certaines caractéristiques pour chaque cluster. Il y a une distribution des échantillons plus ou moins homogène entre les clusters sauf pour le bas gauche (concernant les semaines ayant une plus haute activité le dimanche que les autres jours). Finalement, si nous fusionnons les deux clusters concernant le week-end, comme dit plus haut, nous arrivons à une distribution homogène.

Il est intéressant de voir que K-Medoids n'a pas réussi à retrouver le cluster "en trop" (celui concernant une forte activité le dimanche) lorsque nous fixons 4 clusters. Donc la carte nous permet ici de trouver un groupe supplémentaire.

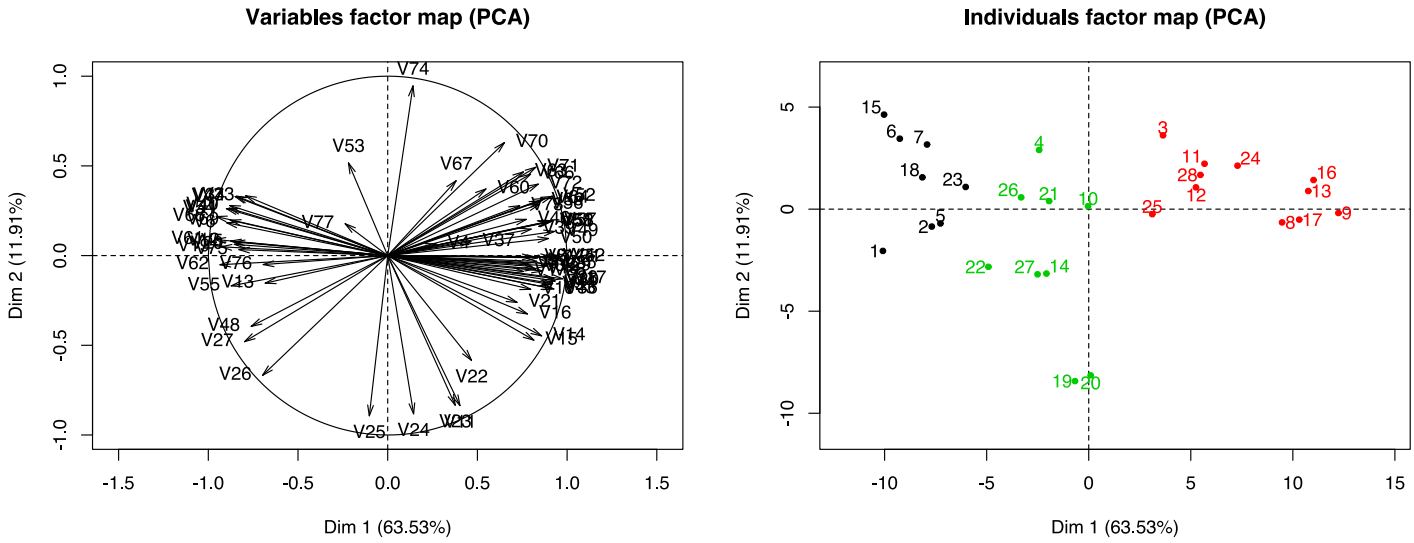
5 Analyse des parkings

5.1 Analyse préliminaire

Comme pour l'analyse du comportement hebdomadaire, nous devons trouver un nombre de classes adéquats. En utilisant les mêmes méthodes, *NbClust* nous indique majoritairement 3 classes.

5.1.1 Analyse en composantes principales (ACP)

D'abord, nous avons groupé les jours avec la moyenne des mesures normalisés (c'est à dire que nous gardions une seule mesure par jour, étant la moyenne des mesures qu'il contient). Nous avons fait une ACP pour trouver la corrélation entre les jours et avons colorés les points en utilisant le partitionnement obtenu avec K-Medoids (avec 3 classes) de la partie 5.2.1.



(a) Plan factoriel des variables.

(b) Plan factoriel des individus.

Figure 10: Résultats de l'ACP.

Il y a une corrélation négative entre les jours ouvrés et les week-ends. En effet, ce sont les variables associées aux jours ouvrés qui sont plutôt corrélées avec l'axe 1 tandis que celles associés aux jours du week-end sont corrélées négativement avec cet axe.

Nous pouvons donc penser que les parkings plus à droite ont une occupation forte dans les week-ends, alors que le groupe plus à gauche ont une plus forte occupation dans les jours ouvrés. Cette hypothèse se vérifie quand nous regardons la charge par jour de chaque cluster (voir 5.2.1), c'est la relation jour ouvré et week-end qui les différencie. C'est donc pour cela que nous avons des groupes bien séparés sur le plan factoriel des individus.

Nous remarquons que la partition est plus "correlée" avec l'axe 1. Nous interprétons cette relation comme l'importance du week-end pour la classification.

Nous voyons que les jours 22 à 25 semble corrélés négativement avec l'axe 2. Cela nous donne deux (presque) outliers sur le plan des individus. Ces jours correspondent à la quatrième semaine d'Octobre qui est une semaine de vacances scolaires à Birmingham, ce point sera discuté en détails en 5.2.2.

Il est important de garder en tête que l'ACP n'est pas une méthode ayant pour objectif de faire apparaître des clusters, l'idée est uniquement de trouver des axes maximisant la variance. Mais ici cela nous permet tout de même de retomber sur les classes trouvées par K-Medoids. C'est donc que dans notre cas les critères séparant les classes sont ceux maximisant la variance du jeu de données.

De plus, il est à noté, que comme nous pouvons assimiler notre matrice à une table de contingence, nous aurions pu utiliser ici l'analyse des correspondances.

5.2 Classification

5.2.1 K-Medoids

Nous analysons maintenant les résultats obtenus avec K-Medoids, toujours en normalisant par la capacité et le χ^2 et en utilisant la distance basée sur les ondelettes.

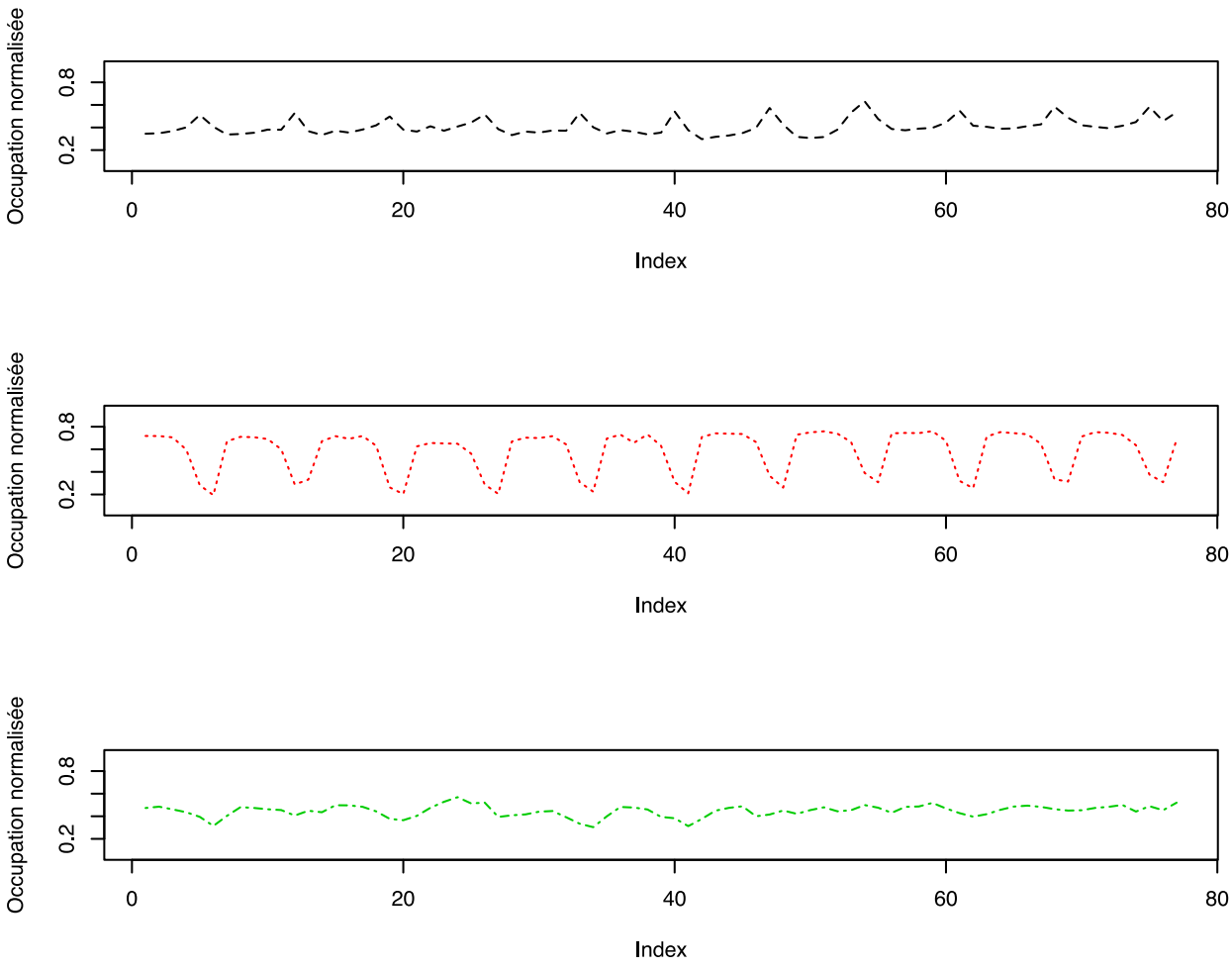


Figure 11: Courbe moyenne des mesures de la semaine pour chaque cluster.

Ci dessus, nous affichons les occupations normalisées par jour de chaque cluster. Nous voyons que les profils des clusters ressemblent fortement à ceux trouvés dans l'analyse du comportement hebdomadaire.

Nous avons un cluster ayant une plus forte activité le week-end, un autre pour lequel l'activité est forte pour les jours ouvrés et enfin le dernier ayant une activité plus ou moins constante.

Nous remarquons cependant une légère corrélation entre les deux derniers clusters, en effet, même si l'activité du troisième semble plus ou moins constante nous pouvons voir qu'elle diminue aussi le week-end, mais la différence est moindre que sur le deuxième.

Ce graphique confirme l'information que l'on avait découverte avec l'ACP qui nous indiquait la nature des clusters uniquement à l'aide des contributions des variables à chaque axe, sans même devoir afficher la courbe moyenne.

	Cluster 1	Cluster 2	Cluster 3
Nombre de parkings	8	11	9

Table 3: Distribution des parkings parmi les clusters.

Les clusters trouvés sont équilibrés par rapport au nombre de parkings dans chaque groupe.

5.2.2 Self-Organizing Map

Nous avons, comme pour l'analyse du comportement hebdomadaire, utilisé une carte auto adaptative sur nos données. Nous n'avons que 28 échantillons donc nous ne pouvions prendre une carte 10×10 comme précédemment (car sinon le nombre de noeuds de la carte serait supérieur au nombre d'échantillons). Nous avons donc décidé d'utiliser une carte hexagonale de taille 4×4.

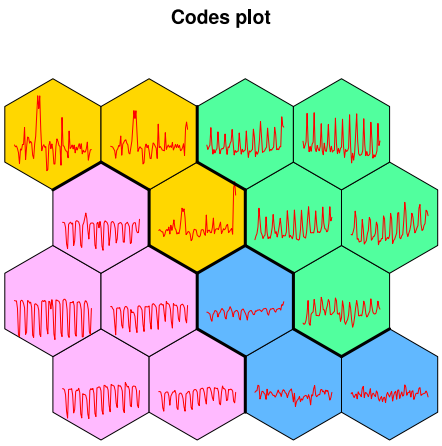


Figure 12: Résultat de la carte auto adaptative.

Nous avons dans un premier temps fait un partitionnement avec 3 clusters et nous avons retrouvés les clusters de la partie précédente. Nous avons utilisés ensuite 4 clusters, alors que nous retrouvions les 3 principaux profils, nous en avons vu un nouveau apparaître:

- Cluster vert (haut droite): faible activité la semaine et forte le week-end.
- Cluster rose (bas gauche): forte activité la semaine et faible le week-end.
- Cluster bleu (bas droite): activité moyenne et constante au long de la semaine.
- Cluster jaune (haut gauche): pic au niveau de la quatrième semaine.

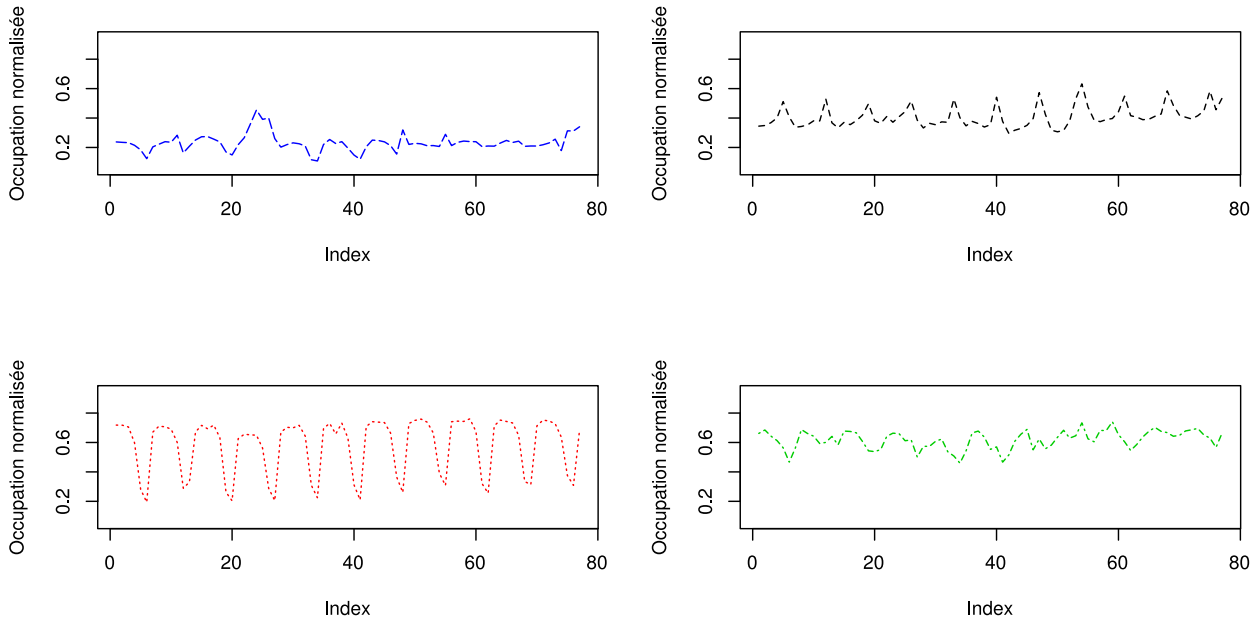
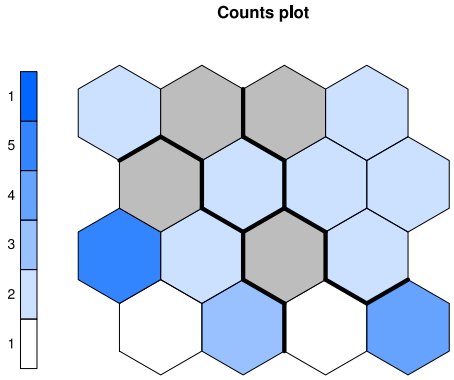


Figure 13: Courbe moyenne des mesures pour chaque cluster.

Les courbes moyennes ressemblent fortement à celles trouvées par K-Medoids. Nous retrouvons, encore une fois, les 3 profils principaux qui ressortent de cette étude.

Mais un nouveau profil est apparu où nous remarquons un pic aux alentours de la quatrième semaine (du 25 au 31 Octobre) qui correspond en fait à une période de vacances scolaires dans la ville de Birmingham. Cela nous indique donc que les vacances scolaires influent sur l'occupation, cette information pourrait être utilisée pour améliorer les prédictions d'occupation des parkings. Il est à noté que K-Medoids parvient lui aussi à trouver ce nouveau cluster lorsque nous l'exécutons avec $k = 4$.



Cluster	Parkings concernés
Haut gauche	4
Bas gauche	11
Haut droite	8
Bas droite	5

Table 4: Caractéristiques par cluster.

Figure 14: Densité de chaque noeud de la carte.

La répartition n'est pas homogène ici. Les groupes principaux sont ceux ayant une forte activité le week-end et ceux en ayant une faible à ce moment. Si nous mélangeons les deux petits clusters nous retombons sur le partitionnement de K-Medoids.

5.3 Analyse géographique

Nous utilisons ici la localisation des parkings afin de regarder sur la carte de Birmingham les alentours de chacun d'eux.

Nous utilisons les informations du site présentant le jeu de données[1] pour trouver les coordonnées GPS de chaque parking. Cependant, selon ce site, 10 des parkings ont des mêmes coordonnées pointant un endroit dans l'eau, loin de Birmingham, nous pensons donc que les coordonnées sont erronées. Pour afficher la carte nous utilisons la librairie *ggmap* de R.

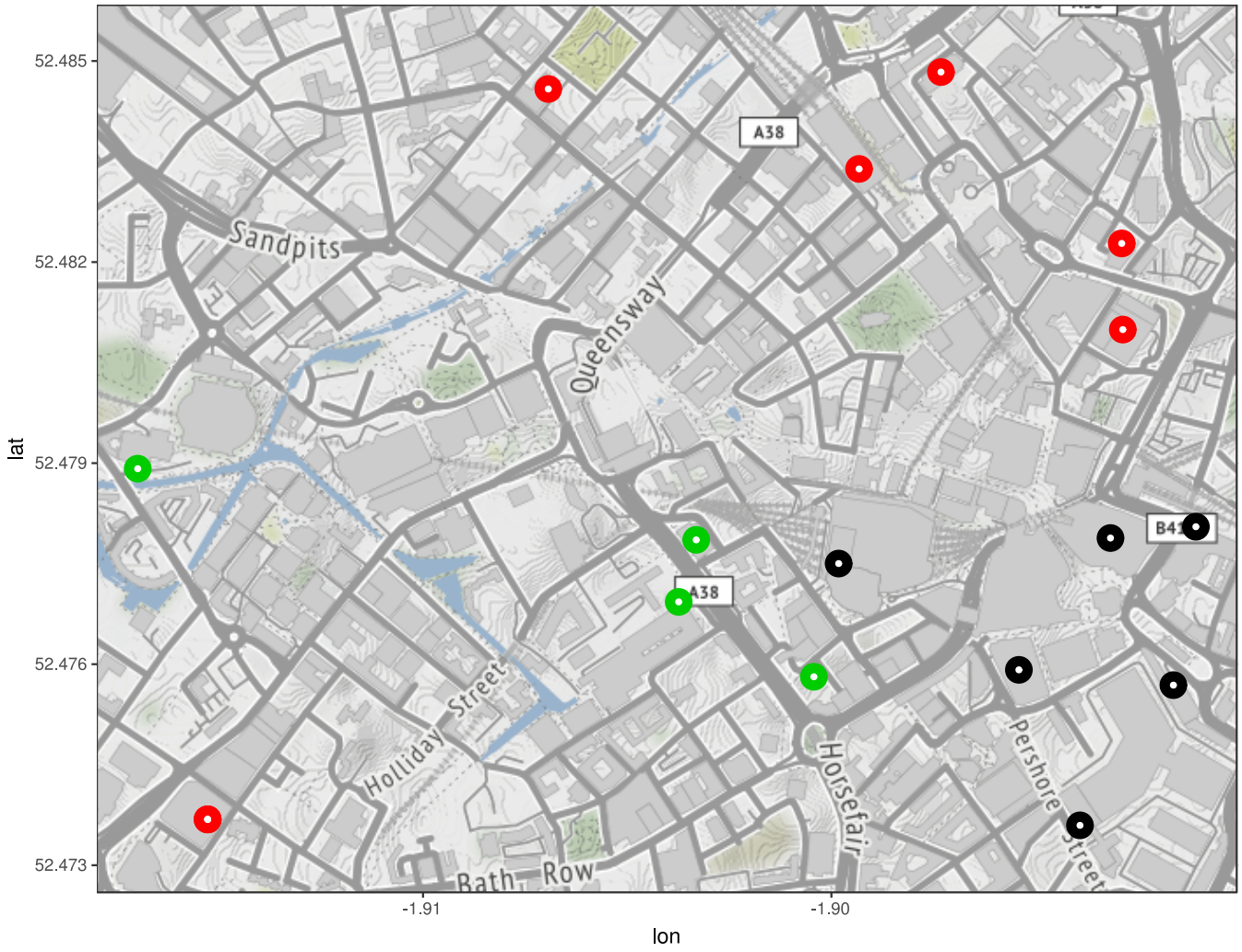


Figure 15: Carte de la ville de Birmingham centrée sur les parkings colorés par le partitionnement de K-Medoids.

Les points noirs (bas droite) correspondent au cluster ayant une forte activité le week-end. Ils semblent être dans une zone commerciale lorsque nous regardons autour d'eux sur un service de cartographie en ligne. Cela fait sens de voir que pour ce genre de zone l'activité est plus forte le week-end, en effet nous nous rendons plus souvent aux centres commerciaux le week-end qu'en semaine.

Les points rouges (haut droite et un isolé en bas à gauche) correspondent aux parkings ayant une forte activité les jours ouvrés. En utilisant un service de cartographie nous voyons que ces points sont proches d'un hôpital, de la cours des magistrats et du palais de justice de Birmingham. Ce n'est donc pas surprenant de voir que leur activité est forte en semaine et non le week-end car ce sont des lieux de travail généralement fermés le week-end.

[1] https://data.birmingham.gov.uk/dataset/birmingham-parking

Les points verts sont donc le groupe restant, celui ayant une activité plus ou moins homogène au cours du temps. Étant donné que, comme vu en 5.2.1, leur activité est légèrement corrélée avec le groupe noir, cela fait sens de les voir géographiquement proche de celui-ci.

6 Conclusions et futurs travaux

Nous voyons à travers ces différentes analyses que nous pouvons extraire différents types de profils que cela soit pour les semaines ou pour les parkings.

Il y a en général 3 profils principaux. Une activité "constante" tout au long de la semaine, une faible activité la semaine et forte en week-end et enfin l'inverse (forte activité la semaine et faible en week-end). Nous avons pu justifier ces différents comportements à l'aide d'une analyse géographique des parkings et de leurs alentours.

De plus nous avons vu certains cas particuliers en augmentant le nombre de cluster. Nous voyons d'une part que certaines semaines ont un profil lié aux fortes activités le dimanche et faible la semaine. D'autre part, pour les parkings, nous voyons que les périodes de vacances scolaire peuvent grandement influencer sur l'activité de certains parkings.

Nous pourrions penser qu'il y a d'autres cas particuliers comme décrit ci-dessus. Le troisième cluster (celui ayant une activité constante) pourrait en fait les contenir et une analyse approfondie de ce cluster pourrait révéler de nouveaux profils.

Nous pourrions aussi approfondir les expériences avec l'affichage temporel de la carte auto adaptative (appendice A). Ainsi que celles avec le co-clustering (appendice B).

Le répertoire github associé à ce projet et ouvert à tous et libre d'utilisation[2].

References

[Côme et al., 2010] Côme, E., Cottrell, M., Verleysen, M., and Lacaille, J. (2010). Aircraft engine health monitoring using Self-Organizing Maps. In Ed., P. P., editor, *10th Industrial Conference, ICDM 2010,*, volume 6171 of *LNAI*, pages 405–417, Berlin, Germany. Springer.

[Stolfi et al., 2017] Stolfi, D. H., Alba, E., and Yao, X. (2017). Predicting car park occupancy rates in smart cities. pages 107–117.

Appendices

A Self-Organizing Map: affichage temporel

A la manière de [Côme et al., 2010] nous avons essayé d'afficher l'évolution des mesures d'un cluster de parking sur la carte auto adaptative.

L'idée est d'utiliser le **jeu de données des parkings** pour faire un clustering et calculer le centroïde de chaque cluster (en faisant la moyenne sur chaque dimension). Ensuite nous utilisons le **jeu de données des jours** pour entraîner une carte auto adaptative. Pour finir, nous découpons le vecteur de chaque centroïde en plusieurs jours puis nous faisons une prédiction à l'aide de la carte pour savoir à quel noeud chaque jour appartient (la même procédure peut être faite avec le **jeu de données des semaines**).

Ensuite nous pouvons colorer chaque noeud prédit en fonction du temps, ici nous utilisons un dégradé du clair vers le foncé (jaune vers le bleu foncé en passant par le vert). En théorie nous devrions avoir 77 noeuds colorés pour les jours (car les mesures de chaque parking sont étalées sur 77 jours), mais la prédiction donne certaines fois le même noeud pour différents jours.

Nous pouvons voir sur les figures 16 et 18 que le tracé temporel de chaque cluster est plus ou moins bien séparé de celui des autres. Comme on pouvait s'y attendre, chaque cluster se partage une partie de la carte. Du moins cette conclusion l'est d'autant plus sur l'affichage avec les semaines qu'avec les jours.

[2] `https://github.com/MatPont/Birmingham_parking`

L'interprétation du profil des clusters est plus facile sur la carte des semaines, nous pouvons facilement voir lequel correspond aux parkings ayant une forte activité le week-end (gauche), ceux en ayant une faible (milieu) et ceux ayant une activité constante (droite).

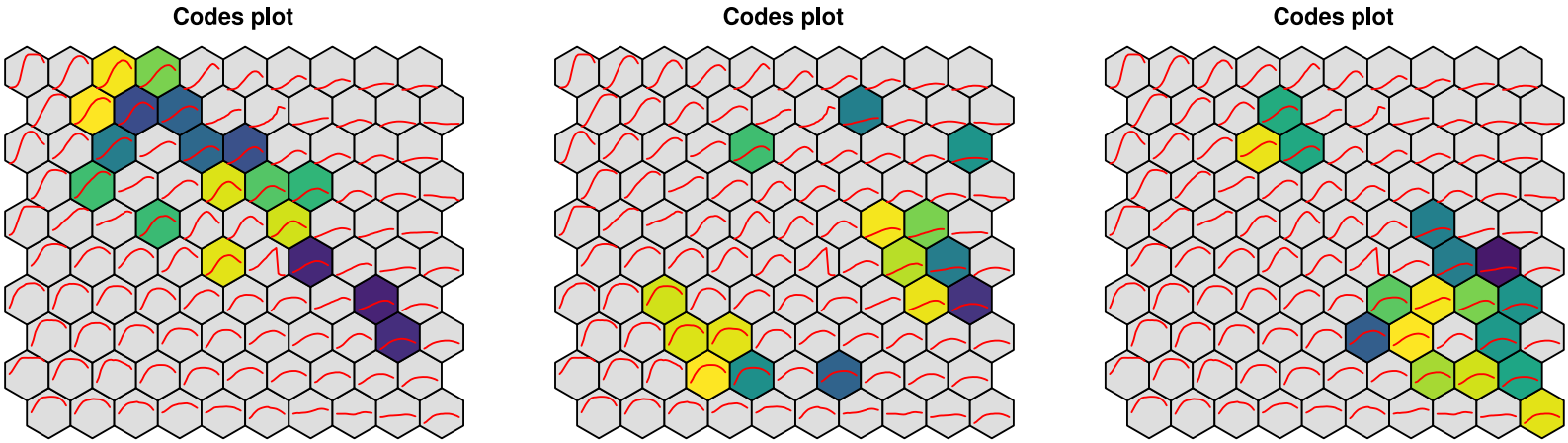


Figure 16: Affichage temporel du centroïde de chaque cluster pour chaque jour.

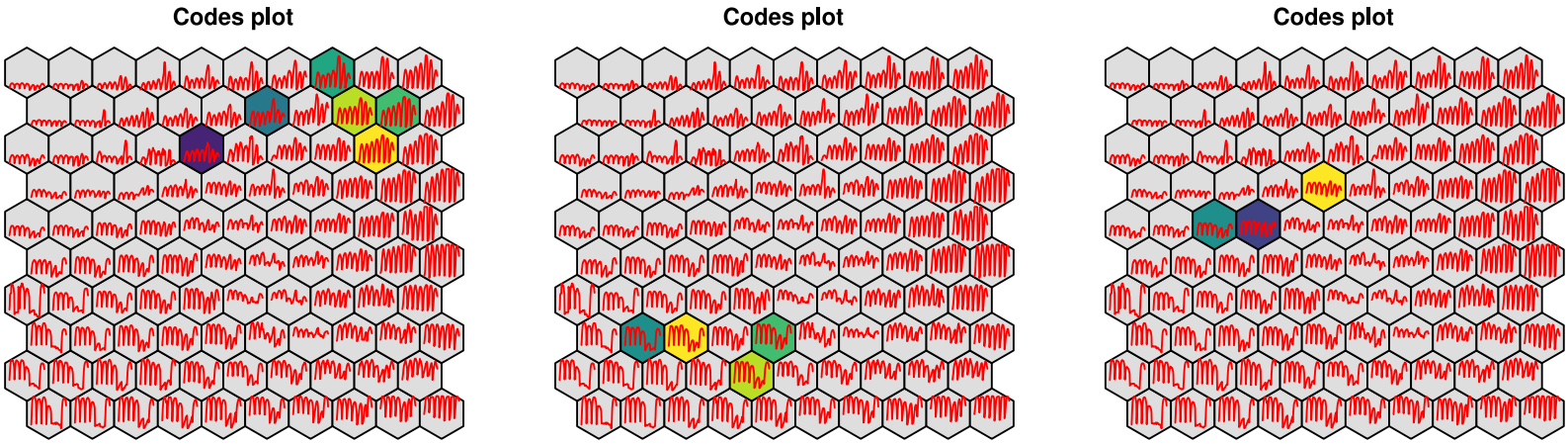


Figure 17: Affichage temporel du centroïde de chaque cluster pour chaque semaine.

B Co-clustering

Nous avons aussi essayé d'utiliser un modèle de mélange faisant du co-clustering afin de voir si l'ajout d'un clustering en colonnes pourrait apporter quelque chose. Nous n'avons malheureusement pas eu le temps d'approfondir.

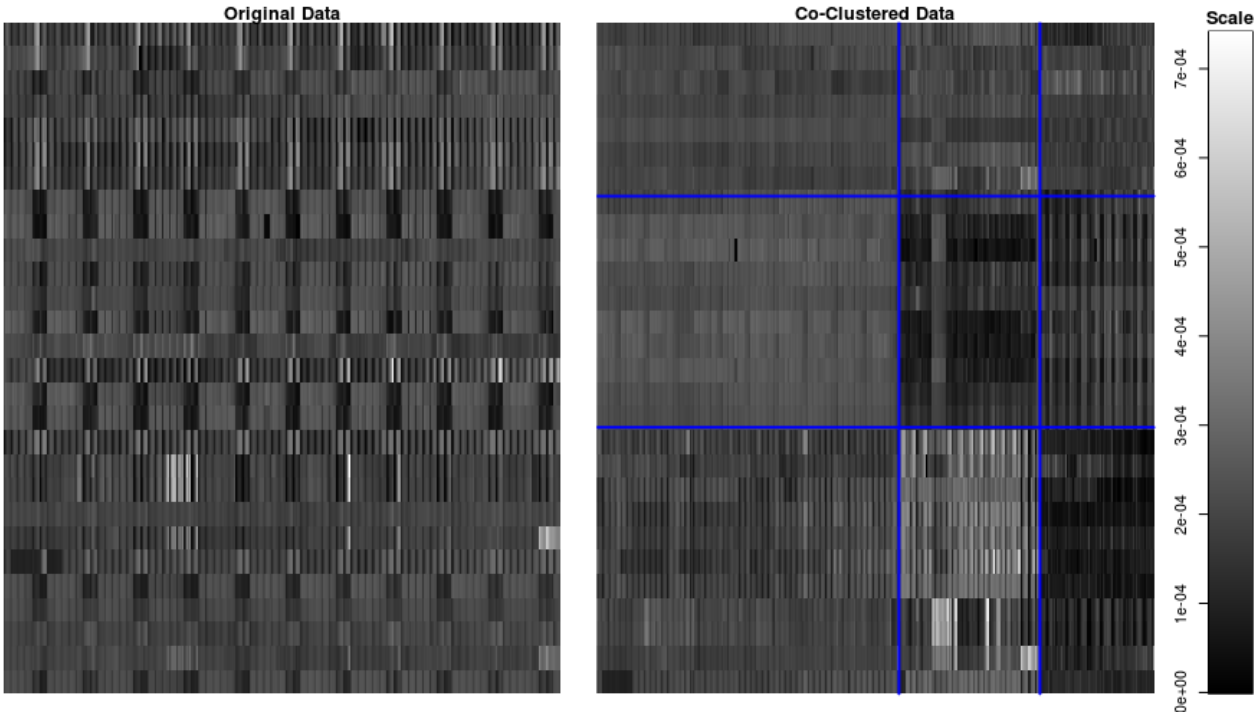


Figure 18: Réorganisation des données après le coclustering.

Nous retrouvons parmi les clusters lignes les 3 principaux profils que ceux trouvés précédemment. Mais en plus de cela nous obtenons des clusters colonnes ayant du sens, par exemple un des clusters colonnes contient des mesures uniquement de début de journée.

Il serait intéressant de voir l'impact sur le clustering en ligne, comme nous n'avons pas de réel label nous ne pouvons utiliser des métriques telles que la NMI ou la ARI pour évaluer le clustering. D'autres critères peuvent aider, ou sinon l'interprétation des résultats.