



Titre: Développement et visualisation d'indicateurs de performance d'une entreprise d'autopartage
Title: entreprise d'autopartage

Auteur: Amadou Mbagnick Sarr
Author:

Date: 2023

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Sarr, A. M. (2023). Développement et visualisation d'indicateurs de performance d'une entreprise d'autopartage [Mémoire de maîtrise, Polytechnique Montréal].
Citation: PolyPublie. <https://publications.polymtl.ca/55116/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/55116/>
PolyPublie URL:

Directeurs de recherche: Martin Trépanier
Advisors:

Programme: Maîtrise recherche en génie industriel
Program:

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

**Développement et visualisation d'indicateurs de performance d'une entreprise
d'autopartage**

AMADOU MBAGNICK SARR

Département de mathématiques et de génie industriel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

Génie industriel

Août 2023

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé :

Développement et visualisation d'indicateurs de performance d'une entreprise d'autopartage

présenté par **Amadou Mbagnick SARR**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

a été dûment accepté par le jury d'examen constitué de :

Jean-Marc FRAYRET, président

Martin TRÉPANIER, directeur de recherche

Delphine VAILLANCOURT, membre

REMERCIEMENTS

Je tiens à exprimer mes sincères remerciements à toutes les personnes qui ont contribué à la réalisation de ce mémoire.

Tout d'abord, je souhaite remercier chaleureusement mon directeur de recherche, Martin Trépanier, pour m'avoir offert cette opportunité et pour son encadrement précieux tout au long de ce projet. Ses conseils avisés, ses idées stimulantes et son soutien constant ont grandement enrichi cette expérience de recherche.

Je tiens également à exprimer ma gratitude envers l'équipe de Communauto, en particulier Marco Viviani et Delphine Vaillancourt, pour leur collaboration, leur disponibilité et leur contribution à la réalisation de ce mémoire. Leur expertise et leur intérêt pour le sujet ont été essentiels pour la collecte des données et la compréhension du contexte de l'autopartage.

Mes remerciements vont également au Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) pour leur soutien financier qui a rendu possible la réalisation de ce projet de recherche.

J'adresse mes remerciements aussi à Amaury Philippe pour sa précieuse collaboration avec les données et pour ses suggestions éclairées qui ont contribué à améliorer la qualité de ce travail.

Je souhaite exprimer ma profonde gratitude envers ma famille, mon père, ma mère, Yaye Touty et Djiby, pour leur amour, leur soutien indéfectible et leurs encouragements tout au long de mes études. Je suis également reconnaissant envers la famille de ma tante, Tata Khady, Michel, Yakham et Jeanne, qui m'a accueilli et m'a apporté leur soutien lors de mon séjour ici.

Enfin, je voudrais remercier sincèrement Adama Niass et Ndeye Sira Thiam, qui sont devenus une partie précieuse de ma famille ici.

À tous ceux qui ont contribué de près ou de loin à la réalisation de ce mémoire, je vous exprime ma profonde gratitude. Vos encouragements, votre soutien et votre confiance ont été des moteurs essentiels dans la concrétisation de ce travail. Merci du fond du cœur pour tout !

RÉSUMÉ

L'autopartage s'est imposé ces dernières années comme une solution viable et durable dans le domaine de la mobilité urbaine, offrant une alternative économique aux modes de transport traditionnels et transformant ainsi notre approche de la mobilité. Face à l'évolution rapide et à l'adoption croissante de l'autopartage, il est essentiel de disposer d'indicateurs de performance fiables et d'outils de visualisation adaptés pour optimiser la gestion des opérations d'autopartage et anticiper les besoins futurs.

Ce mémoire s'attache à examiner l'importance de l'autopartage, son évolution et la nécessité d'avoir des indicateurs de performance pertinents, ainsi que des outils pour visualiser ces indicateurs. À travers une recherche approfondie dans la littérature, un tableau exhaustif des indicateurs de performance a été établi, incluant leurs sources, les objets qu'ils concernent, leur description détaillée ainsi que leur façon de les calculer. Cette étude a permis d'identifier les principales tendances et les enjeux liés à l'autopartage, ainsi que de bonnes pratiques pour mesurer et évaluer la performance des entreprises d'autopartage.

En outre, ce travail présente une valorisation des fichiers transactionnels représentant les réservations de véhicules Communauto, à partir desquels deux fichiers ont été créés. Le premier fichier décrit l'état du réseau par intervalle de 15 minutes, tandis que le second fichier présente les circonstances de chacune des réservations par saut de 15 minutes. Les circonstances englobent le nombre de véhicules disponibles dans des rayons de 250, 500 et 750 mètres, avec la possibilité de ne compter que les véhicules électriques ou ceux en libre-service intégral.

Afin de faciliter la compréhension et l'utilisation des indicateurs de performance identifiés, des tableaux de bord ont été développés. Ces tableaux de bord permettent de visualiser un grand nombre d'indicateurs développés, ainsi que pour mettre en valeur les deux fichiers créés. Ces tableaux de bord se composent de courbes et de graphiques, offrant une représentation claire et compréhensible des tendances et des performances. Dans cet outil de visualisation de données, nous avons développé plusieurs interfaces pour faciliter la compréhension de l'usage des objets :

Outil de visualisation des indicateurs de performance des objets : Cette interface affiche différents KPI liés à l'usage des véhicules ou des utilisateurs. Ces KPI, calculés à partir d'un jeu de données, peuvent inclure des distances d'utilisation ou des durées d'utilisation, selon les paramètres spécifiés par l'utilisateur.

Outil de visualisation de la carte de l'état du réseau : Cet outil affiche une carte indiquant l'emplacement des véhicules disponibles sur le réseau, selon la date et le type de service choisis par l'utilisateur. Cela donne un aperçu en temps réel de l'état du réseau.

Outil de visualisation des circonstances : Cette interface présente une carte montrant le nombre de véhicules disponibles par type dans un rayon spécifique. Les utilisateurs peuvent filtrer les données en fonction de leurs préférences, ce qui facilite une utilisation optimale des ressources et une meilleure compréhension des besoins des utilisateurs.

Outil de visualisation d'indicateurs sur les stations : Cette interface fournit une vue d'ensemble des indicateurs de performance cumulés pour chaque station, aidant à comprendre comment les ressources sont utilisées à chaque station et favorisant une gestion efficace des ressources.

En somme, ce mémoire souligne l'importance croissante de l'autopartage dans le contexte actuel de la mobilité urbaine et met en évidence la nécessité d'avoir des indicateurs de performance pertinents et des outils de visualisation adaptés. Grâce à ces outils, les entreprises d'autopartage pourront mieux gérer leur flotte de véhicules, améliorer leurs services et contribuer efficacement à la transition vers des modes de transport plus durables et responsables. Les résultats obtenus, principalement les deux fichiers produits dans cette étude pourraient servir de base pour de futures recherches et applications dans le domaine de l'autopartage, en favorisant l'adoption et le développement de pratiques de gestion et de planification optimales.

ABSTRACT

In recent years, car-sharing has emerged as a viable and sustainable solution in the realm of urban mobility, offering an economical alternative to traditional transportation modes and thus transforming our approach to mobility. With the rapid evolution and increasing adoption of car-sharing, it is essential to have reliable performance indicators and suitable visualization tools to optimize the management of car-sharing operations and anticipate future needs.

This end-of-study thesis focuses on examining the importance of car-sharing, its evolution, and the need for relevant performance indicators, as well as tools to visualize these indicators. Through an in-depth literature review, a comprehensive table of performance indicators has been established, including their sources, the objects they concern, and their detailed description. This study allowed the identification of main trends and challenges related to car-sharing, as well as good practices for measuring and evaluating the performance of car-sharing companies.

Moreover, this work presents a valorization of transactional files representing vehicle reservations from Communauto, from which two files were created. The first file describes the state of the network at 15-minute intervals, while the second file presents the circumstances of reservations at 15-minute jumps. Circumstances encompass the number of vehicles available within radii of 250, 500, and 750 metres, with the option to count only electric vehicles or those in flex mode.

To facilitate the understanding and use of the identified performance indicators, dashboards have been developed. These dashboards allow for the visualization of a large number of indicators present in the table, as well as highlighting the two created files. The dashboards presenting indicators from the table of indicators consist of curves and graphs, providing a clear and understandable representation of trends and performances. Those presenting the two files of network status and reservation circumstances are based on a map, allowing for a detailed and intuitive geographic representation.

In conclusion, this end-of-study thesis underlines the growing importance of car-sharing in the current context of urban mobility and highlights the need for relevant performance indicators and suitable visualization tools. With these tools, car-sharing companies will be able to better manage their fleet of vehicles, improve their services, and effectively contribute to the transition towards more sustainable and responsible transportation modes. The findings obtained in this study could

serve as a basis for future research and applications in the field of car-sharing, fostering the adoption and development of optimal management and planning practices.

TABLE DES MATIÈRES

REMERCIEMENTS	III
RÉSUMÉ.....	IV
ABSTRACT	VI
TABLE DES MATIÈRES	VIII
LISTE DES TABLEAUX.....	XI
LISTE DES FIGURES.....	XII
LISTE DES SIGLES ET ABRÉVIATIONS	XIII
LISTE DES ANNEXES.....	XIV
CHAPITRE 1 INTRODUCTION.....	1
1.1 Mise en contexte.....	1
1.2 Objectifs	2
1.3 Structure du mémoire	2
CHAPITRE 2 REVUE DE LITTÉRATURE	4
2.1 Définition de l'autopartage.....	4
2.2 Objets de l'autopartage.....	5
2.3 Indicateurs de performance	7
CHAPITRE 3 MÉTHODOLOGIE.....	11
3.1 Fichiers bruts de Communauto	12
Fichier des événements	14
Fichier de l'état du réseau	15
3.1.1 Enrichissement du fichier de l'état du réseau.....	17
3.2 Fichier Circonstances	24
CHAPITRE 4 TABLEAU DES INDICATEURS	27

4.1	Définitions.....	27
4.2	Tableau.....	28
CHAPITRE 5 VISUALISATION		36
5.1	Présentation de Panel Holoviz.....	36
5.2	Étapes de réalisation du tableau de bord de visualisation des indicateurs	37
5.3	Visualisation de quelques indicateurs de performances des objets.....	38
5.3.1	Besoins de visualisation	38
5.3.2	Données requises en entrée et formalisation	39
5.3.3	Réalisation de la plateforme sur <i>Panel</i>	41
5.3.3.1	Récupération et prétraitement des données	41
5.3.3.2	Ajout des <i>widgets</i> de paramétrage.....	41
5.3.3.3	Création de l'application	43
5.4	Visualisation de la carte de l'état du réseau	47
5.4.1	Besoins de visualisation	47
5.4.2	Données requises en entrée et formalisation	48
5.4.3	Réalisation de la plateforme sur Panel	48
5.4.3.1	Récupération et prétraitement des données	48
5.4.3.2	Ajout des <i>widgets</i> de paramétrage.....	49
5.4.3.3	Création de l'application	49
5.5	Visualisation des circonstances	53
5.5.1	Besoins de visualisation	53
5.5.2	Données requises en entrée et formalisation	54
5.5.3	Réalisation de la plateforme sur Panel	54
5.5.3.1	Récupération et prétraitement des données	54

5.5.3.2	Ajout des widgets de paramétrage	54
5.5.3.3	Création de l'application	55
5.6	Visualisation d'indicateurs sur les stations	58
5.6.1	Besoins de visualisation	58
5.6.2	Données requises en entrée et formalisation	59
5.6.3	Réalisation de la plateforme sur Panel	59
5.6.3.1	Récupération et prétraitement des données	59
5.6.3.2	Ajout des widgets de paramétrage	59
5.6.3.3	Création de l'application	60
CHAPITRE 6	CONCLUSION ET RECOMMANDATIONS	64
6.1	Contributions	64
6.2	Limitations	64
6.3	Perspectives	65
RÉFÉRENCES		66
ANNEXES		70

LISTE DES TABLEAUX

Tableau 2.1 Différents types d'autopartage	5
Tableau 3.1 : Structure du fichier des réservations	13
Tableau 3.2 : Champs du fichier des événements	15
Tableau 3.3 : Structure du fichier de l'état du réseau	20
Tableau 3.4 Champs du fichier des circonstances.....	25
Tableau 4.1 : Tableau des indicateurs	29
Tableau 4.2 : Références des indicateurs du tableau.....	31

LISTE DES FIGURES

Figure 3.1: Méthodologie générale du traitement informatique	12
Figure 3.2: Organigramme de création du fichier de l'état du réseau	16
Figure 3.3 : Évolution du nombre de lignes sans position connue.....	21
Figure 3.4 : Proportion des lignes sans position connue sur le nombre possible	22
Figure 5.1 : <i>Widget</i> des paramètres	43
Figure 5.2: Interface de visualisation des indicateurs sous forme de graphe	44
Figure 5.3 : Distances d'utilisation des usagers du service basé station	45
Figure 5.4 : Maximum des durées d'utilisation des usagers flex	45
Figure 5.5 : Maximum des durées d'utilisation des usagers dans le district Ville-Marie	46
Figure 5.6: Paramètres du tableau de bord de l'état du réseau	49
Figure 5.7: Interface de visualisation cartographique de l'état du réseau.....	50
Figure 5.8 : Extrait de la carte de l'état du réseau	51
Figure 5.9 : Vue éclatée des véhicules disponibles dans l'extrait de carte	52
Figure 5.10: Paramètres du tableau de bord des circonstances	55
Figure 5.11 : Interface de visualisation cartographique des circonstances liées aux réservations	56
Figure 5.12 : Vue agrandie de la page des circonstances de réservation du tableau de bord.....	57
Figure 5.13 : Page des profils cumulés du tableau de bord.....	61
Figure 5.14 : Page du profil longitudinal du nombre de véhicules libres du tableau de bord.....	62
Figure 5.15 : Page du nombre de stations vides du tableau de bord	63
Figure B.1 : Aperçu du fichier de l'état du réseau.....	76
Figure C.1 : Aperçu du fichier des circonstances.....	77

LISTE DES SIGLES ET ABRÉVIATIONS

CSV Comma-Separated Values, fichier séparé par des virgules

FFCS Free-Floating Car Sharing, autopartage en libre-service

KPI Key Performance Indicator, indicateur clé de performance

SBCS Station-Based Car Sharing, autopartage basé station

SOC State Of Charge, état de charge de la batterie du véhicule électrique

CR Circonstances de réservation

ER État du réseau

LISTE DES ANNEXES

Annexe A	Code python de la création des fichiers	70
Annexe B	Aperçu du fichier de l'état du réseau	76
Annexe C	Aperçu du fichier des circonstances.....	77

CHAPITRE 1 INTRODUCTION

1.1 Mise en contexte

Ces dernières années, l'avènement de l'autopartage comme alternative viable de moyen de transport fait passer la mobilité privée de la propriété à l'utilisation de services partagés. L'autopartage fait référence à l'utilisation temporaire d'une voiture sans les responsabilités de propriété, avec des paiements liés aux frais d'utilisation ou d'abonnement. Contrairement à la location de voitures traditionnelle, l'autopartage s'appuie sur la médiation d'une plateforme pour identifier les correspondances entre les ressources du fournisseur et des utilisateurs pour faciliter les échanges (Eckhardt et al, 2019). Différents types d'autopartage existent et ceux-ci diffèrent par leur mode et leur tarification (Liao et al, 2018). L'autopartage est apparu au Canada en 1994 avec la coopérative Auto-Com devenue Communauto. C'est devenu un moyen de transport de plus en plus utilisé, dû à ses avantages au niveau du coût et de son impact sur l'environnement. Ainsi, les utilisateurs de l'autopartage devraient atteindre les 36 millions et partageraient 427000 véhicules à l'horizon 2025 (Neumann, 2021). Avec l'inquiétude grandissante sur la congestion de la circulation et de la propriété des véhicules (Steininger et al, 1996), l'autopartage apparaît comme une solution. En plus des raisons environnementales, les utilisateurs sont aussi motivés par des raisons telles que la commodité ou l'économie de coût (Orski, 2001).

Compte tenu de l'aspect nouveau de ce mode de transport, il est difficile de l'intégrer dans les modèles de planification actuels, et il est difficile d'appréhender son rôle à travers, par exemple, des bases de données communes issues des enquêtes origine-destination. Des travaux confirment le potentiel d'analyse et de modélisation des systèmes d'autopartage et la demande en outils avancés afin que les entreprises d'autopartage répondent à la croissance (Tuominen et al, 2019). Un des domaines qui pourraient bénéficier de ces analyses est la gestion de la flotte de véhicules. Les caractéristiques d'utilisation de la flotte d'autopartage sont différentes de celles des autres flottes de véhicules. Des milliers d'utilisateurs utilisent des véhicules dans un large éventail de paramètres. Il devient alors important d'identifier des indicateurs sur cette utilisation et de les mettre en pratique grâce à l'analyse de données opérationnelles brutes. Avoir une vision claire sur ces indicateurs de performance permettrait alors à un opérateur d'autopartage de mieux gérer sa flotte de véhicules.

1.2 Objectifs

L'objectif global de ce mémoire est de contribuer à l'amélioration du fonctionnement d'une entreprise d'autopartage par le développement et la visualisation d'indicateurs de performance pertinents. Afin d'atteindre cet objectif, les sous-objectifs suivants ont été identifiés :

- Contribution à l'organisation des données opérationnelles de notre partenaire industriel (Communauto).
 - Requier le montage d'une nouvelle base de données à partir des transactions et des événements recueillis chez l'opérateur Communauto.
 - Requier la création de deux nouveaux fichiers : le fichier de l'état du réseau et le fichier des circonstances.
- Identification d'indicateurs de performance pertinents dans la littérature et création/adaptation de nouveaux indicateurs
 - Passe par une revue de littérature sur l'autopartage et les indicateurs de performance des flottes des véhicules d'autopartage et l'analyse des données transformées.
- Développement d'un prototype d'interface de visualisation de ces indicateurs
 - Demande la réalisation d'une plateforme de visualisation permettant d'avoir une vision en tableau de bord et intégration des indicateurs pertinents.

À noter que l'objectif du mémoire n'est pas d'analyser les indicateurs développés, mais bien de fournir les outils à ceux qui auront à le faire, notamment les planificateurs de notre partenaire de recherche.

1.3 Structure du mémoire

Ce mémoire est composé de six chapitres incluant le chapitre introductif. Après celui-ci suit un consacré à une revue de littérature sur l'autopartage et ses indicateurs de performances. Cette revue a pour but de faire l'état des travaux sur l'autopartage, ensuite plus précisément sur des indicateurs de performance identifiés. Le troisième chapitre présente la méthodologie suivie pour cette étude. Il consiste à un recensement d'indicateurs, puis à la création de deux fichiers avec des informations importantes sur l'état du réseau d'autopartage et sur les circonstances de réservation d'un véhicule.

Dans ce chapitre, les données utilisées seront décrites, tout comme les méthodes de transformation pour aboutir aux deux fichiers. Dans le chapitre 4, les résultats seront présentés avec un tableau d'indicateurs recensant les différents indicateurs trouvés dans la littérature. Les deux fichiers créés y seront aussi analysés. Le chapitre 5 est celui consacré à la mise en place d'une plateforme de visualisation de nos indicateurs. Enfin, le dernier chapitre sera celui de la conclusion où seront montrées les contributions et les limites de notre mémoire.

CHAPITRE 2 REVUE DE LITTÉRATURE

Dans ce chapitre, nous présentons une revue de la littérature d'abord axée sur l'autopartage en mettant l'accent sur ses définitions, ses origines, mais aussi ses différentes déclinaisons. Ensuite, nous parlerons de travaux touchant à des indicateurs de performance pour un système d'autopartage.

2.1 Définition de l'autopartage

L'autopartage est un mode de transport dans lequel un véhicule est partagé entre plusieurs personnes. Cette pratique a vu le jour pour la première fois à Zurich en 1948, avec une motivation économique (Shaheen et al., 1999). D'autres tentatives ont également eu lieu en Europe par le passé, telles que "Procotip" à Montpellier en France ou "Witkar" à Amsterdam aux Pays-Bas. Cependant, ces expériences ont échoué (Doherty et al., 1987).

De nos jours, l'autopartage s'est répandu un peu partout dans le monde, notamment dans les villes à forte densité, où les coûts et l'encombrement liés à la possession d'un véhicule sont plus élevés. Les véhicules sont généralement gérés par une organisation qui s'occupe de les déployer dans la ville et les utilisateurs doivent adhérer à cette organisation pour pouvoir utiliser le service. Les frais de carburant et d'entretien sont pris en charge par l'organisation et les utilisateurs doivent payer une redevance à chaque utilisation du service.

Selon le type, la fonction et le public ciblés, l'autopartage peut être divisé en trois grandes catégories. Tout d'abord, il y a l'autopartage en station, qui peut être en aller-retour ou en aller simple. Dans ce type d'autopartage, les utilisateurs doivent récupérer le véhicule à une station dédiée et le ramener au même endroit ou à une autre station désignée. Ensuite, il y a l'autopartage entre particuliers, où des particuliers peuvent partager leur propre voiture avec d'autres utilisateurs moyennant une rémunération. Ce type d'autopartage est souvent géré par des plateformes en ligne et est de plus en plus populaire. Enfin, il y a l'autopartage en libre-service intégral, dans lequel les utilisateurs peuvent trouver et réserver un véhicule via une application mobile, et utiliser le véhicule pour un trajet en aller simple, avant de le garer dans une zone désignée.

Il est important de noter que l'autopartage présente de nombreux avantages, notamment la réduction des coûts liés à la possession d'un véhicule, la diminution de la congestion routière, la réduction de

la pollution atmosphérique et la promotion de l'utilisation de modes de transport plus durables (Martin et Shaheen, 2011). Cependant, comme pour tout mode de transport, il y a aussi des défis à relever, tels que la gestion de la flotte de véhicules, l'optimisation des services pour répondre aux besoins des utilisateurs et la garantie de la sécurité des utilisateurs.

Tableau 2.1 Différents types d'autopartage

	Autres appellations	Définitions
Autopartage en station	• Autopartage en boucle • Autopartage traditionnel	Une compagnie privée offre à ses clients des véhicules positionnés à des stations déterminées moyennant une compensation financière. Après utilisation, ces véhicules doivent être retournés à leur station d'origine. (Münzel et al, 2019)
	• Autopartage unilatéral avec réservation	Il est similaire au mode en boucle, mais l'individu n'est pas obligé de ramener le véhicule à la même station et peut se garer dans des endroits désignés, qui peuvent être dispersés dans une ville ou une région. (Shaheen et al, 2015)
Autopartage entre particuliers	• Prêt entre personnes	Des particuliers louent leur véhicule à des automobilistes pour une courte durée et moyennant une compensation financière. Les modalités de restitution sont préalablement définies (Shaheen et al., 2012)
Autopartage en libre-service intégral	• Autopartage en trace directe • Autopartage serviciel	Une compagnie privée offre à ses clients des véhicules positionnés dans une zone prédéterminée. Après utilisation, ces véhicules peuvent être retournés à n'importe quelle place de stationnement autorisée dans cette même zone. (Wielinski et al. 2015)

2.2 Objets de l'autopartage

Morency et al. (2008) ont identifié plusieurs objets clés dans leur analyse orientée objet du système d'autopartage. Ces objets comprenaient notamment les véhicules, les utilisateurs, les stations et les

réservations (transactions). L'utilisation de la modélisation orientée objet dans le système d'autopartage leur a permis de normaliser et de structurer le calcul des différents indicateurs de performance, tels que la dispersion spatiale et la disponibilité des voitures.

Les véhicules constituent le cœur du système d'autopartage, car ils sont mis à la disposition des utilisateurs pour être utilisés à des fins de déplacement. Ils sont équipés de technologies de suivi et de communication pour permettre aux utilisateurs de les trouver, de les réserver et de les utiliser. Les données relatives aux véhicules, telles que leur emplacement, leur état et leur utilisation, sont collectées et analysées pour améliorer l'efficacité et la performance du système.

Les utilisateurs sont les personnes qui utilisent les véhicules dans le système d'autopartage. Ils peuvent être des particuliers ou des entreprises qui utilisent les véhicules pour les déplacements de leur personnel. Les utilisateurs peuvent s'inscrire pour obtenir un compte auprès de l'opérateur du système d'autopartage, donc en devenir membres, et ils peuvent alors accéder aux véhicules en utilisant une application mobile ou en se rendant à une station. Les données relatives aux utilisateurs, telles que leur emplacement, leur utilisation des véhicules et leur comportement de paiement, sont collectées et analysées pour mieux comprendre les besoins des utilisateurs et améliorer l'offre de service.

Les stations sont des endroits où les utilisateurs peuvent prendre et déposer les véhicules. Les stations peuvent être situées à des emplacements stratégiques tels que les gares, les aéroports ou les centres-villes. Elles peuvent être équipées de bornes de recharge pour les véhicules électriques et de systèmes de communication pour permettre aux utilisateurs de trouver et de réserver les véhicules. Les données relatives aux stations, telles que leur emplacement, leur utilisation et leur état, peuvent être collectées et analysées pour optimiser la disponibilité des véhicules et améliorer l'expérience utilisateur.

Les réservations sont des transactions clés dans le système d'autopartage. Les utilisateurs peuvent réserver un véhicule à partir de leur compte en ligne ou via l'application mobile de l'opérateur. Les réservations permettent aux utilisateurs de planifier leur voyage à l'avance et de s'assurer que le véhicule dont ils ont besoin sera disponible à la station de leur choix. Les données relatives aux réservations, telles que leur durée, leur fréquence et leur annulation, peuvent être collectées et analysées pour mieux comprendre les besoins des utilisateurs et améliorer l'offre de service.

En conclusion, les véhicules, les utilisateurs, les stations et les réservations sont les quatre éléments clés d'un système d'autopartage. Chacun d'entre eux joue un rôle important dans le fonctionnement du système, et les données collectées sur ces éléments peuvent être utilisées pour améliorer l'efficacité et la performance du système d'autopartage dans son ensemble

2.3 Indicateurs de performance

D'un point de vue méthodologique, les études sur l'analyse des services d'autopartage se sont faites sous forme d'enquêtes, de modèles de simulation ou d'analyse de données. Elles peuvent avoir différents objectifs comme la caractérisation du système, la proposition d'améliorations du service ou des considérations environnementales. Les études sur la caractérisation du système visent à avoir un portrait du service autant du point de vue de l'utilisateur que du point de vue de l'opérateur. Dans ce contexte, beaucoup d'indicateurs renseignant sur les caractéristiques du système ont eu à être présentés. (Martin, 2007) a effectué un travail portant sur la caractérisation du système de l'autopartage dans l'agglomération de Montréal et sur une analyse des objets de ce système que sont les usagers, les stationnements et les véhicules. En prenant l'autopartage comme un système orienté objet, il les définit de façon formelle avant de construire des indicateurs permettant de les analyser sur des données opérationnelles. Ainsi des indicateurs, de nature statistique ou spatiale, relatifs aux abonnés, aux véhicules ou encore aux stations ont été développés.

Un problème crucial dans l'industrie de l'autopartage est la description des comportements d'utilisation des clients. En fonction de leurs activités d'utilisation, les utilisateurs peuvent être divisés selon leurs attributs. (Morency et al., 2007) ont identifié des types d'utilisateurs en utilisant l'algorithme des k-means. Huit groupes ont été trouvés et pour chacun, il y avait un jour préféré pour utiliser le service d'autopartage. Ils ont aussi utilisé d'autres caractéristiques des utilisateurs tels que la fréquence d'utilisation pour classer en utilisateur occasionnel et fréquent ou encore la longueur du voyage pour différencier les utilisateurs de longue distance et ceux de courte distance. Un modèle économétrique du comportement des utilisateurs a été proposé par (Habib et al., 2012). Ce modèle peut prévoir la durée d'utilisation continue des utilisateurs, identifier le mois au cours duquel les utilisateurs deviennent le plus actifs et déterminer le nombre de membres actifs par mois. Dans le mode d'autopartage en libre-service intégral, (Niels et al., 2017) ont montré que la distance entre l'utilisateur et le véhicule a une influence sur la possibilité de sélectionner un véhicule.

En ce qui concerne l'amélioration du service, plusieurs travaux s'y sont consacrés. (Barth et Todd ,1999) développent un modèle de simulation des opérations d'autopartage avec des entrées et des mesures d'efficacité qui permettent l'analyse de scénarios. Ils concluent qu'une taille de flotte suffisante pour satisfaire les clients est de 3 à 6 véhicules pour 100 trajets, mais que 18 à 24 véhicules pour 100 trajets sont nécessaires pour minimiser les coûts de relocalisation. (Kek et al. ,2009) et (Correia et Antunes, 2012) ont proposé deux modèles de programmation en nombres entiers mixtes qui ont pour but d'optimiser les opérations dans un système d'autopartage. Kek et al. , pour le cas d'une société d'autopartage à Singapour, ont conclu que l'optimisation de la main-d'œuvre peut réduire les dépenses de personnel jusqu'à 50 % et le temps avec zéro véhicule (durée de pénurie de véhicules dans les stations) jusqu'à 13 %. (Correia et Antunes, 2012), quant à eux, se concentrent sur la taille de la flotte, le nombre de déplacements de véhicules, la taille du dépôt et l'emplacement des stations de stationnement potentielles. L'étude conclut qu'un ratio véhicules/déplacement de 22,7 pour 100 déplacements est la composition optimale de la flotte, ce qui confirme la conclusion de (Barth et Todd ,1999).

Dans le but d'analyser les performances de leur modèle de réallocation de véhicules dans un système d'autopartage, (Alfian et al. ,2017) ont utilisé les indicateurs sur le taux d'utilisation des véhicules et sur le taux d'acceptation des réservations.

Récemment, la recherche sur la distance parcourue a attiré l'attention dans les domaines du transport et de l'urbanisme (Ding et al., 2017). Malgré cela, la distance parcourue par les membres de l'autopartage n'a pas été un objectif principal de recherche et n'a été utilisée que pour des analyses descriptives et préliminaires. Cependant, quelques études ont approfondi l'analyse de la distance parcourue par les membres de l'autopartage, indiquant qu'il s'agit d'un facteur essentiel pour comprendre le comportement des utilisateurs de l'autopartage.

Par exemple (Baumgarte et al., 2021) ont constaté que la distance parcourue par les utilisateurs d'autopartage n'est pas seulement le plus important pour les revenus de l'opérateur, mais également le meilleur indicateur pour révéler le comportement d'utilisation de ses utilisateurs. Ils ont utilisé une approche de regroupement pour étudier les données d'utilisation de plus de 4 200 utilisateurs actifs et les ont divisés en trois groupes : fréquents, à long terme et longue distance. Leurs résultats ont révélé que les utilisateurs fréquents avaient une longueur médiane de trajet de 70 km, tandis que 98 % des utilisateurs de vacances appartiennent aux utilisateurs de longue durée ou longue distance avec une longueur moyenne de trajet de 230 km et ont duré plus de 24 h.

Dans une autre étude, (Baumgarte et al.,2022) ont utilisé les mêmes données avec différentes méthodologies d'apprentissage automatique. La distance de trajet a de nouveau été utilisée comme principal indicateur pour analyser le comportement de déplacement des utilisateurs d'autopartage en fonction de leurs besoins personnels, l'heure, la voiture et les caractéristiques liées à l'environnement. Leurs résultats ont montré que les caractéristiques liées au temps avaient une plus grande influence sur la distance parcourue dans tous les groupes. Par exemple, commencer tard ou tôt le matin, se terminer le soir ou la nuit et commencer le week-end entraîne une plus grande distance parcourue.

De même, (Mishra et al.,2015) ont examiné le comportement de déplacement des utilisateurs de l'autopartage à l'aide de l'enquête California Household Travel Survey 2010-2012. Ils ont constaté que les répondants qui vivent et travaillent dans les centres-villes ne possèdent pas de voiture personnelle et parcourent de courtes distances. (Costain et al. ,2012) ont étudié les schémas de mobilité complets des membres de l'autopartage à l'aide d'ensembles de données de transactions du programme d'autopartage dominant de Toronto. Leurs résultats montrent que la plupart des trajets en autopartage étaient de courtes distances et que la distance totale parcourue par 64 % des membres était inférieure à 40 km par an, ce qui indique que l'autopartage à Toronto contribue à l'augmentation des trajets urbains en voiture sur de courtes distances.

Une autre étude menée par (Kopp et al. ,2015) a examiné en quoi le comportement de déplacement des membres de l'autopartage diffère de celui de la population générale. Ils ont constaté que la distance de déplacement quotidienne moyenne des membres de l'autopartage était plus longue (50,8 km) que celle de la population générale (46,8 km). Ils ont en outre constaté que la distance moyenne du trajet pour les membres de l'autopartage était plus élevée dans les domaines de l'éducation, du magasinage, des loisirs et des affaires que pour la population générale et que le vendredi et le dimanche, les membres voyageaient beaucoup plus longtemps. Les auteurs de cette étude ont affirmé que, globalement, les schémas de mobilité des membres de l'autopartage semblent être plus orientés vers un objectif et plus flexibles, car ils organisent leurs activités de manière différente de ceux n'utilisant pas l'autopartage et utilisent donc le service de manière plus consciente.

En outre, (Luca & Di Pace ,2014) ont représenté la distance comme une impédance spatiale, car, à mesure que la distance du trajet augmente, la probabilité d'être intéressé par les services d'autopartage diminue, quel que soit le niveau de service fourni par les modes de transport disponibles. En outre, les utilisateurs réguliers qui parcourent de plus longues distances constituent

un segment difficile à attirer pour les fournisseurs de services d'autopartage, car l'augmentation de la distance hebdomadaire des trajets réduit la propension à un système d'autopartage.

CHAPITRE 3 MÉTHODOLOGIE

Le but de ce chapitre est de présenter la méthodologie utilisée dans ce mémoire. Pour cela, la Figure 2.1 représente le traitement informatique des données. Dans un premier temps, nous avons travaillé à partir d'une base de données brute provenant de Communauto (A). Nous avons ensuite obtenu des informations sur l'état du réseau (B) grâce à ces données brutes. Ce fichier renseigne sur la disponibilité des véhicules ainsi que leur position par intervalle de 15 minutes.

Par la suite, nous avons enrichi ce fichier (B) avec un autre fichier brut provenant de Communauto (C). Ce fichier a permis d'ajouter des informations sur la dernière position connue de chaque véhicule. Cela nous a permis d'avoir une vision plus précise de la position des véhicules dans le réseau grâce aux informations sur le dernier arrêt connu.

Un autre fichier important pour notre travail est celui qui fournit des éclaircissements sur les circonstances de la réservation (E). Ce fichier nous donne des informations sur le nombre de véhicules disponibles ainsi que leurs attributs dans chaque rayon fixé autour de la réservation. Ce fichier a été produit à partir du fichier de l'état du réseau enrichi (D).

Enfin, pour avoir une vision claire sur ces fichiers et les différents indicateurs évoqués, nous avons développé une plateforme de visualisation (F). Cette plateforme nous permet de mieux comprendre les données et les indicateurs que nous avons collectés. Elle facilite également l'analyse des données en les présentant de manière plus claire et intuitive.

Ces différentes étapes ont été décrites plus en détail dans les sections suivantes. Grâce à cette méthodologie, nous avons pu collecter des données pertinentes sur le réseau de Communauto et réaliser notre travail de manière efficace.

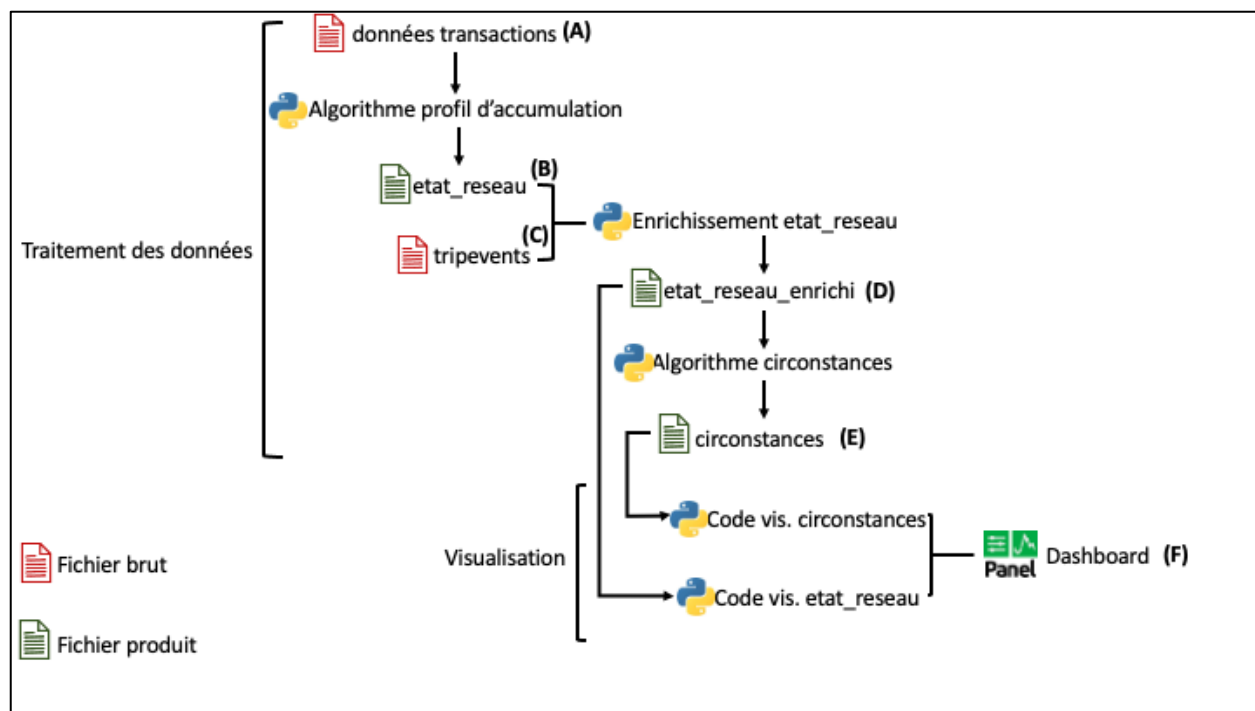


Figure 3.1: Méthodologie générale du traitement informatique

3.1 Fichiers bruts de Communauto

Pour mener à bien notre étude, nous avons commencé par collecter les fichiers bruts de Communauto. Le premier fichier important que nous avons obtenu est celui contenant les informations sur les réservations de véhicules. Ce fichier renseigne sur les différentes transactions effectuées par les utilisateurs de Communauto à Montréal, incluant la date, l'heure et la durée de la réservation, ainsi que la station de départ et la station d'arrivée. En plus de ces informations, ce fichier contient également des données géographiques importantes, telles que la longitude et la latitude des stations.

Fichier des réservations

Le fichier contenant des informations brutes sur les transactions a été fourni sous format « comma-separated values » (CSV) et s'étale sur une durée de 3 mois, de janvier 2021 à mars 2021. Il est le résultat de la fusion de différentes tables de la base de données de Communauto et contient les colonnes présentes au Tableau 3.1.

Tableau 3.1 : Structure du fichier des réservations

Nom de colonne	Définition	Type
st_id	Identifiant unique de la table station	Entier
st_lon	Longitude de la station	Décimal
st_lat	Latitude de la station	Décimal
v_id	Identifiant unique de la table véhicule	Entier
v_type_modele	Type de voiture (essence, hybride ou électrique)	Chaîne de caractères
u_customer_id	Identifiant unique de l'utilisateur	Entier
res_dist_price	Prix calculé en fonction de la distance parcourue lors de la réservation	Décimal
res_time_price	Prix calculé en fonction de la durée de la réservation	Décimal
res_total_price	Prix total de la réservation	Décimal
res_number_of_trips	Nombre de trajets effectués lors de la réservation	Entier
st_no	Identifiant de la station	Entier
v_brand	Marque de la voiture	Chaîne de caractères
v_model	Identifiant du modèle de la voiture	Entier
v_nom_modele	Modèle de la voiture	Chaîne de caractères
v_no_serie	Numéro de série de la voiture	Chaîne de caractères
v_annee	Année du véhicule	Entier
u_gender_nom	Genre de l'utilisateur	Chaîne de caractères
u_lang_nom	Langue de l'utilisateur	Chaîne de caractères
u_type	Type d'utilisateur	Entier
u_naissance_date	Date de naissance de l'utilisateur	Date
u_abonnement_date	Date d'abonnement de l'utilisateur	Date
a_id	Identifiant de l'utilisateur	Entier
a_city_id	Identifiant de la ville de l'utilisateur	Entier
a_postal_code	Code postal de l'utilisateur	Chaîne de caractères
a_long	Longitude de l'adresse (Code postal)	Décimal

Tableau 3.1 : Structure du fichier des réservations (suite et fin)

Nom de colonne	Définition	Type
a_lat	Latitude de l'adresse (Code postal)	Décimal
a_district	Arrondissement de l'utilisateur	Chaîne de caractères
res_id	Identifiant unique de la table réservation	Entier
res_trip_start_datetime	Horodatage du début de l'emprunt	Date
res_trip_end_datetime	Horodatage de la fin de l'emprunt	Date
res_trip_start_date	Jour du début de l'emprunt	Date
res_trip_end_date	Jour de fin de l'emprunt	Date
res_tarif_no	Tarif appliqué	Flottant
res_trip_dist_km	Distance parcourue en km	Flottant
res_trip_length_hr	Durée de l'emprunt en heure	Flottant
res_type_service	Description du type de service	Chaîne de caractères
TripEventID_1	Identifiant unique de la table TripEvents, correspondant au premier événement d'une réservation	Entier
TripEventType_1	Type du premier événement	Entier
Latitude_1	Latitude du premier événement	Flottant
Longitude_1	Longitude du premier événement	Flottant
TripEventDate_1	Horodatage du premier événement	Date
TripEventID_2	Identifiant unique de la table TripEvents, correspondant au dernier événement d'une réservation	Date
TripEventType_2	Type du dernier événement	Chaîne de caractères
Latitude_2	Latitude du dernier événement	Flottant
Longitude_2	Longitude du dernier événement	Flottant
TripEventDate_2	Horodatage du dernier événement	Date

Fichier des événements

C'est un fichier sous format CSV nous provenant de Communauto et qui contient les différents arrêts moteur des véhicules lors de leurs réservations. Il contient des événements pour la période de novembre 2020 à décembre 2021 et a pour colonnes les éléments présentés au tableau suivant.

Tableau 3.2 : Champs du fichier des événements

Colonne	Description	Type
TripEventID	Identifiant unique de l'événement	Entier
CustomerID	Identifiant unique du client	Entier
CarID	Identifiant unique du véhicule, correspond à v_id du fichier des réservations	Entier
ReservationID	Identifiant unique de la réservation, correspond à res_id du fichier des réservations	Entier
TripEventDate	Date de l'événement	Date
TripEventType	Type de l'événement	Chaîne de caractères
Latitude	Latitude de l'événement	Flottant
Longitude	Longitude de l'événement	Flottant
SOC	État de charge de la batterie du véhicule (si électrique) lors de l'événement	Flottant
FuelGauge	Niveau de carburant du véhicule (si à essence) lors de l'événement	Flottant

Fichier de l'état du réseau

Nous allons ici présenter le procédé qui nous a permis d'arriver au fichier d'état du réseau.

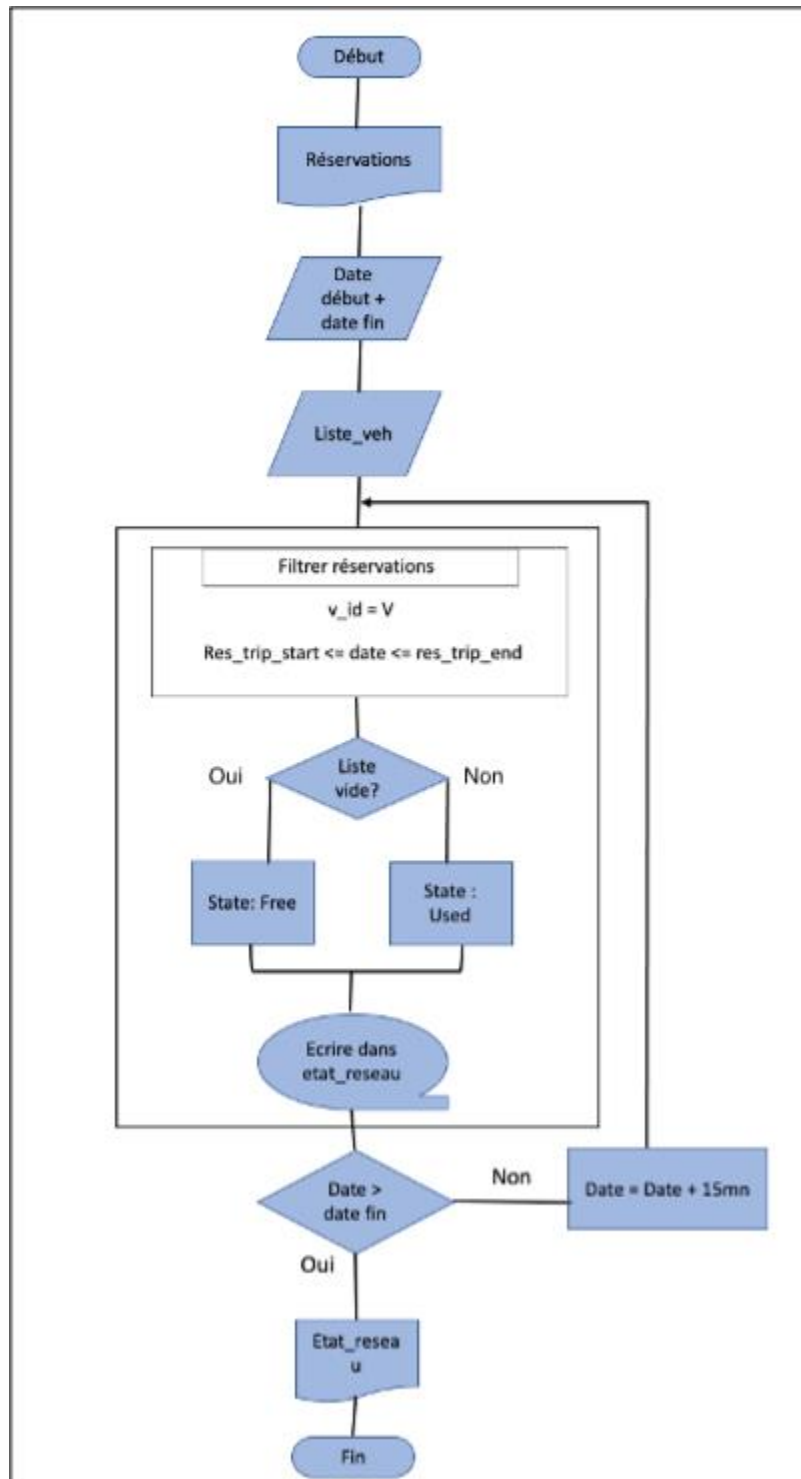


Figure 3.2: Organigramme de création du fichier de l'état du réseau

Dans cette partie, nous décrivons en détail l'algorithme de création du fichier d'état du réseau, présenté à la Figure 3.2. Cet algorithme permet de générer un fichier qui renseigne sur la disponibilité des véhicules à intervalles de 15 minutes, pour une période de temps donnée. Pour cela, l'algorithme prend en entrée le fichier multi contenant toutes les réservations, la liste des véhicules de la flotte et les dates de début et de fin de la période d'intérêt.

L'algorithme procède ensuite par étapes en se basant sur les données d'entrée. Tout d'abord, il parcourt la liste des véhicules et filtre les réservations ne concernant que ce véhicule. Ensuite, il teste s'il y a une réservation dont la plage de réservation contient la date à laquelle nous nous situons. Si aucune occurrence n'est trouvée, cela signifie que le véhicule est disponible à cette date, et le champ "state" prend la valeur "Free". Les champs "longitude_start" et "latitude_start" prennent alors la valeur des coordonnées de la station dans le cas d'une voiture de station, tandis que les champs sont laissés libres si c'est en libre-service intégral.

En revanche, si une occurrence est trouvée, cela signifie que le véhicule est en cours d'utilisation, et le champ "state" prend la valeur "Used". Les champs sur les coordonnées sont alors remplis directement à partir des coordonnées de la réservation. Enfin, une ligne complète avec les différents champs est inscrite dans le document pour chaque véhicule.

L'algorithme répète ensuite ce traitement pour tous les véhicules de la flotte, pour chaque intervalle de 15 minutes et jusqu'à la date de fin spécifiée en entrée. À la fin du traitement, nous obtenons un fichier d'état du réseau, qui fournit des informations sur la disponibilité des véhicules et leur position à intervalles de 15 minutes pour la période de temps donnée. Ce fichier est une étape cruciale pour notre analyse et est utilisé pour produire d'autres fichiers importants pour notre travail, tels que celui fournissant des éclaircissements sur les circonstances de la réservation.

3.1.1 Enrichissement du fichier de l'état du réseau

Le fichier de l'état du réseau contient des informations importantes sur l'état des véhicules en autopartage, mais pour obtenir des informations plus précises, nous avons enrichi ce fichier en y ajoutant les informations sur la dernière position connue de chaque véhicule dans le système. Ces informations incluent les champs "last_longitude", "last_latitude", "last_TripEventDate" et "last_event".

Pour ce faire, nous avons développé un algorithme qui prend en entrée le fichier "etat_reseau" et le fichier "tripevents". Il lit chaque ligne du fichier de l'état du réseau et effectue des traitements en fonction de l'état du véhicule.

L'algorithme recherche la dernière position connue de chaque véhicule dans le fichier "tripevents". Pour y arriver, nous recherchons la dernière réservation du véhicule avant la date courante à l'aide d'une requête. Le résultat attendu est un extrait du fichier des événements avec toutes les lignes d'événements antérieurs liées au véhicule. L'extrait est ensuite trié, et la première ligne correspond à l'événement le plus récent. Toutefois, il est possible qu'aucun résultat ne soit trouvé en raison du fichier "tripevents" incomplet ou si l'on est au début de l'année.

Si aucun résultat n'est trouvé, les informations à écrire dans le fichier seront différentes en fonction du type d'autopartage du véhicule.

Lorsqu'il s'agit d'un véhicule de station, nous pouvons considérer qu'il était en station avec les coordonnées de la station dans les champs "last_longitude" et "last_latitude", et le champ "last_TripEventDate" portera la mention "Unknown". Lorsqu'il s'agit d'un véhicule en libre-service intégral, les champs sur la position et la date de l'événement seront laissés vides.

Si un résultat est trouvé, l'algorithme extrait les informations sur la dernière position connue du véhicule à partir de la première ligne. Une différence sera notée dans le champ "last_event" en fonction du type d'autopartage et de l'état du véhicule pour plus de précision. Si le véhicule est de type libre-service intégral et non utilisé, le champ sera mis à "Parked". S'il s'agit d'un véhicule de type station et libre d'utilisation, le champ sera mis à « Station ». Lorsque le véhicule est utilisé, il est souvent impossible de déterminer la nature exacte de l'arrêt, le champ sera alors mis à "trip_event".

Ci-dessous se trouve l'algorithme expliquant le processus :

Début

Pour chaque ligne du fichier "etat_reseau" faire

Si le véhicule est "en service"

 Chercher la dernière position du véhicule dans le fichier "tripevents"

Si une position est trouvée alors

 Extraire les informations de la première ligne du résultat de la recherche

Si le véhicule est de type "libre-service intégral" et "non-utilisé" alors

 Mettre le champ "last_event" à "Parked"

 Mettre les coordonnées de "tripevents" dans "last_longitude", "last_latitude"

Sinon si le véhicule est de type "station" et "non-utilisé" alors

 Mettre le champ "last_event" à "Station"

 Mettre les coordonnées de la station dans "last_longitude", "last_latitude"

Sinon si le véhicule est "en déplacement" alors

 Mettre le champ "last_event" à "Trip_event"

 Mettre les coordonnées de "tripevents" dans "last_longitude", "last_latitude"

Fin Si

 Écrire les informations du véhicule enrichies dans le fichier de sortie

Sinon si aucune position n'est trouvée alors

Si le véhicule est de type "libre-service intégral" alors

 Laisser les champs "last_longitude", "last_latitude" et "last_TripEventDate" vides

 Mettre le champ "last_event" à "Unknown"

Sinon si le véhicule est de type "station" alors

 Utiliser les coordonnées de la station comme dernière position

 Mettre le champ "last_TripEventDate" à "Unknown"

 Mettre le champ "last_event" à "Station"

Fin Si

 Écrire les informations du véhicule enrichies dans le fichier de sortie

Fin Si

Fin Si

Fin Pour

Fin

Tableau 3.3 : Structure du fichier de l'état du réseau

Nom de colonne	Définition	Type
<i>datetime</i>	Horodatage de la réservation	Date
<i>res_id</i>	Identifiant unique de la table	Entier
<i>state</i>	État du véhicule (« Free » ou « Used »)	Chaîne de caractères
<i>st_id</i>	Identifiant unique de la table station	Entier
<i>longitude_start</i>	Longitude de départ	Flottant
<i>latitude_start</i>	Latitude de départ	Flottant
<i>res_type_service</i>	Description du type de service	Chaîne de caractères
<i>v_id</i>	Identifiant unique de la table véhicule	Entier
<i>v_type_modele</i>	Type de voiture (essence, hybride ou électrique)	Chaîne de caractères
<i>res_id</i>	Identifiant unique de la réservation	Entier
<i>res_type_service</i>	Description du type de service	Chaîne de caractères
<i>last_longitude</i>	Dernière longitude connue à cette date	Flottant
<i>last_latitude</i>	Dernière latitude connue à cette date	Flottant
<i>last_TripEventDate</i>	Date du dernier événement connu	Date
<i>last_event</i>	Nature du dernier événement	Chaîne de caractères

Le fichier de l'état du réseau enrichi qui a été créé contient un nombre de 17 858 880 lignes pour la période de 3 mois. La taille du fichier reflète la quantité importante d'informations enregistrées, avec des points de données capturés toutes les 15 minutes du 1er janvier au 31 mars. Une telle quantité de données présente à la fois des opportunités et des défis pour l'analyse. D'une part, le riche ensemble de données permet une exploration approfondie des modèles d'utilisation des véhicules et des taux d'utilisation au fil du temps, fournissant des informations précieuses sur l'efficacité de la gestion de la flotte. D'autre part, le volume considérable de données nécessite des outils et des techniques analytiques sophistiqués pour extraire des informations significatives et éviter le risque de paralysie de l'analyse.

Dans le contexte de l'étude de la flotte de véhicules d'une entreprise d'autopartage, il est important de prendre en compte toutes les informations disponibles pour pouvoir tirer des conclusions fiables et pertinentes. Cependant, comme cela est souvent le cas avec les données réelles, il peut y avoir des erreurs, des incohérences ou des données manquantes dans le jeu de données.

Dans le cas présent, une analyse plus détaillée du fichier de l'état du réseau a révélé qu'il y a des valeurs manquantes pour la dernière position connue de certains véhicules. Plus précisément, ces

valeurs manquantes concernent les véhicules en libre-service intégral pour lesquels aucune information n'est disponible quant à leur dernière position connue. Cette situation est due à un manque d'informations dans le fichier des événements qui enregistre les mouvements des véhicules.

Il est important de prendre en compte cette information dans l'analyse des données, car elle peut affecter l'exactitude et la validité des résultats obtenus. En particulier, si une partie importante des données manque, il peut être difficile de tirer des conclusions fiables sur les habitudes d'utilisation des véhicules, les emplacements les plus fréquentés, etc. Cela peut avoir des implications importantes pour la prise de décision dans la gestion de la flotte, par exemple en ce qui concerne le dimensionnement du parc automobile, l'optimisation des trajets, etc.

La qualité des données est cruciale dans l'analyse des résultats, en identifiant les données manquantes et en évaluant leur impact potentiel sur les conclusions. Dans le cas présent, la visualisation graphique des valeurs manquantes pour la dernière position des véhicules peut aider à identifier les zones qui nécessitent une enquête et une correction supplémentaires. Cela peut également aider à développer des stratégies pour améliorer la qualité des données à l'avenir, par exemple en renforçant la surveillance et la collecte de données pour les véhicules en libre-service intégral.

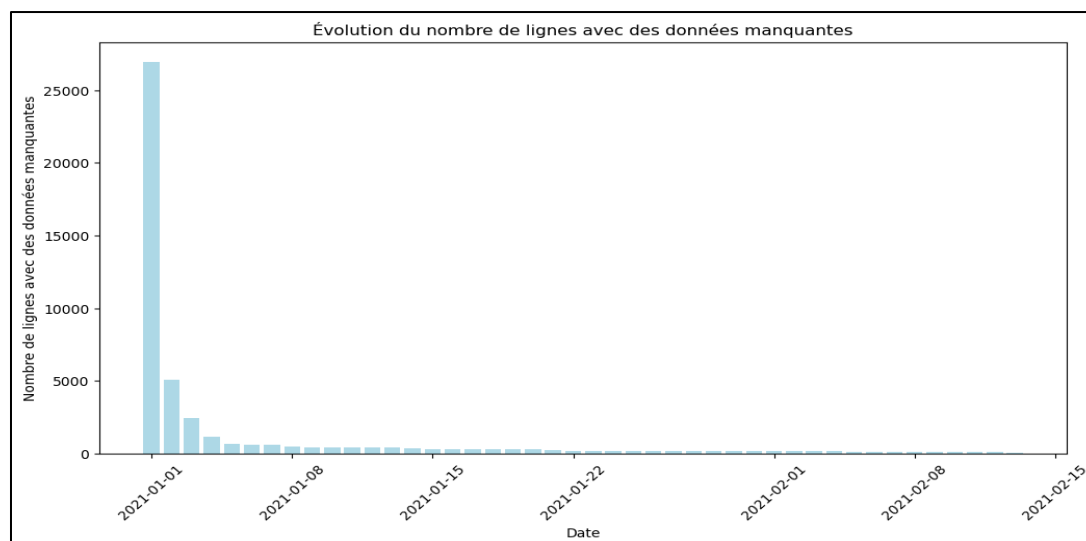


Figure 3.3 : Évolution du nombre de lignes sans position connue

L'analyse des données relatives aux lignes de véhicules avec une dernière position inconnue révèle une tendance significative sur la période allant du 1er au 15 janvier. En effet, le nombre de lignes

concernées a diminué de manière régulière sur cette période, passant de plus de 30 000 le 1er janvier à presque zéro le 15 janvier.

Cette tendance à la baisse pourrait être interprétée de différentes manières. Tout d'abord, elle pourrait indiquer une amélioration de la qualité des données sur cette période, avec une réduction des erreurs ou des lacunes dans les informations collectées. Cela pourrait être dû à des améliorations apportées au processus de collecte et de stockage des données ou à une meilleure formation des employés impliqués dans ce processus.

Une autre explication possible pourrait être liée à la saisonnalité. La période allant du 1er au 15 janvier correspond à la période de retour à la normale après les vacances de fin d'année. Il est donc possible que cette baisse du nombre de lignes concernées soit due à une diminution du nombre d'utilisateurs de la flotte de véhicules pendant cette période, ce qui entraînerait une réduction des données manquantes.

Enfin, il est important de noter que cette tendance ne doit pas être considérée isolément et que d'autres facteurs pourraient influencer le nombre de lignes avec une dernière position inconnue. Une analyse plus approfondie des données et de l'environnement dans lequel elles ont été collectées serait nécessaire pour confirmer ou infirmer ces hypothèses.

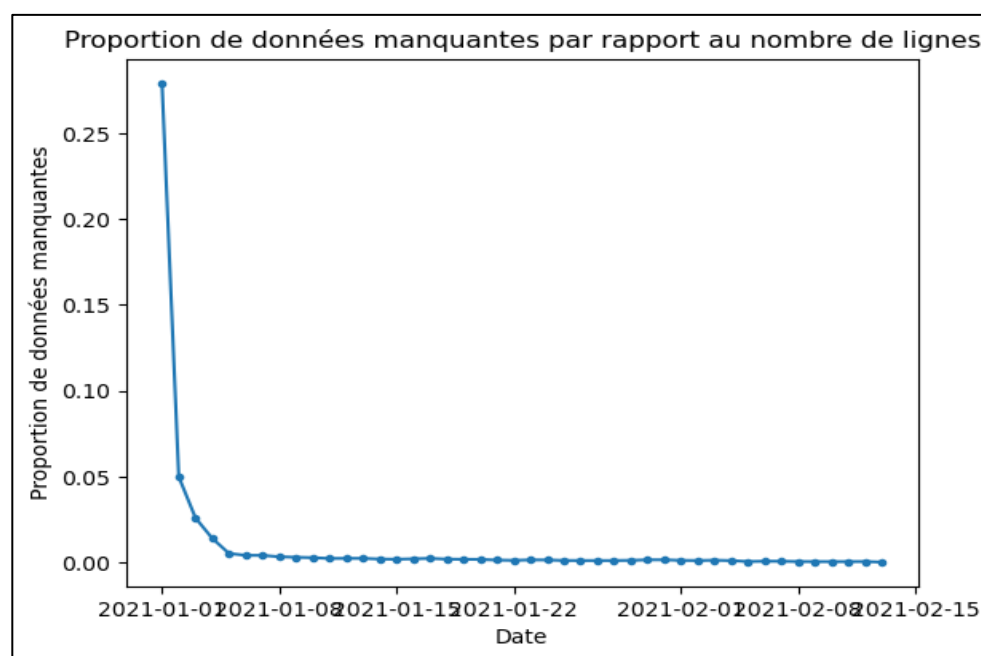


Figure 3.4 : Proportion des lignes sans position connue sur le nombre possible

Il est noté que la proportion des lignes où il n'y a pas d'informations sur la position du véhicule lorsque libre est importante le 1er janvier et baisse drastiquement le jour suivant. Elle continue de baisser jusqu'à être nulle à partir du 15 janvier et il en est ainsi jusqu'à la fin du mois de mars qui coïncide avec la fin de notre fichier.

3.2 Fichier Circonstances

Comme évoqué précédemment, à partir du fichier sur l'état du réseau, un autre a été produit avec des informations sur les circonstances des différentes réservations par intervalle.

Pour cela, nous avons calculé la distance séparant chaque véhicule au moment de sa réservation aux autres véhicules libres à la même date et heure. Vu que la base de données est d'une taille importante, les dates pour lesquelles on veut calculer les distances sont définies et un filtre est appliqué sur la base de données pour ne garder que les réservations de la date courante. On procède ici aussi par intervalles de 15 minutes pour y appliquer le traitement.

Pour chaque date, on parcourt de façon séquentielle l'extrait de la base de données contenant l'état du réseau. À chaque première apparition d'un identifiant de réservation, on filtre pour ne garder que les autres réservations libres.

Il en résulte alors un extrait avec la réservation et tous les autres véhicules libres à la même date avec leurs informations. À partir de cet extrait, on rajoute de façon temporaire une colonne « distance » qui contiendra la distance entre le lieu de la réservation et l'emplacement de chaque véhicule libre à la même date. La distance a été calculée en utilisant la **formule de Haversine** en prenant pour hypothèse que la Terre a une forme sphérique; celle-ci permet de calculer la plus petite distance entre deux points sur une sphère.

$$d = 2r \sin^{-1} \sqrt{\sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + \cos\varphi_1 \cos\varphi_2 \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}$$

d : distance entre les deux points

r : rayon de la sphère

φ_1 , φ_2 : latitude du point 1, latitude du point 2

λ_1 , λ_2 : longitude du point 1, longitude du point 2

Il est important de préciser que les angles sont exprimés en radians. Pour appliquer cette formule à toutes lignes de l'extrait, elle a été transformée en une fonction python déclenchée à chaque itération.

Après avoir calculé les distances entre chaque réservation et les véhicules libres à la même date, il est possible d'obtenir le nombre de véhicules par rayon fixé. Ainsi le nombre de véhicules disponibles a pu être obtenu pour les rayons de 250, 500 et 750 mètres. À partir des informations sur les véhicules, nous avons pu faire une distinction sur le type de véhicule (essence, électrique) mais aussi le type de service (ffcs ou sbcs). Ce traitement est fait pour chaque réservation dans le fichier d'état du réseau. Les colonnes suivantes sont ajoutées.

Tableau 3.4 Champs du fichier des circonstances

Nom de colonne	Définition
<i>v_id</i>	Identifiant unique de la table véhicule
<i>v_type_modele</i>	Type de voiture (essence, hybride ou électrique)
<i>res_id</i>	Identifiant unique de la table
<i>res_type_service</i>	Description du type de service
<i>datetime</i>	Horodatage
<i>nb_veh_avail</i>	Nombre de véhicules disponibles dans le réseau à cette date
<i>nb_ffcs_avail_250</i>	Nombre de véhicules en libre-service intégral disponibles dans un rayon de 250 m autour du lieu de réservation
<i>nb_sbcs_avail_250</i>	Nombre de véhicules en station disponibles dans un rayon de 250 m autour du lieu de réservation
<i>nb_ffcs_avail_500</i>	Nombre de véhicules en libre-service intégral disponibles dans un rayon de 500m autour du lieu de réservation
<i>nb_sbcs_avail_500</i>	Nombre de véhicules en station disponibles dans un rayon de 500 m autour du lieu de réservation
<i>nb_ffcs_avail_750</i>	Nombre de véhicules en libre-service intégral disponibles dans un rayon de 750 m autour du lieu de réservation
<i>nb_sbcs_avail_750</i>	Nombre de véhicules en station disponibles dans un rayon de 750 m autour du lieu de réservation
<i>nb_EV_250</i>	Nombre de véhicules électriques disponibles dans un rayon de 250 m autour du lieu de réservation
<i>nb_EV_500</i>	Nombre de véhicules électriques disponibles dans un rayon de 500 m autour du lieu de réservation
<i>nb_EV_750</i>	Nombre de véhicules électriques disponibles dans un rayon de 750 m autour du lieu de réservation

L'algorithme suivant explique le processus :

Algorithme : Création du fichier des circonstances

Définir une liste de rayons [250, 500, 750]

Pour chaque date **par intervalle de 15 minutes** dans la journée :

Filtrer la base de données pour ne garder que les réservations de la date

Pour chaque réservation ayant débuté à la date ou dans les 14 minutes suivantes

Calculer la distance entre le lieu de la réservation et chaque véhicule libre à la même date, pour chaque rayon de la liste de rayons

Pour chaque distance calculée :

Si la distance est inférieure ou égale au rayon en cours, ajouter le véhicule libre correspondant à la liste des véhicules disponibles pour ce rayon

Pour chaque réservation dans l'intervalle :

Pour chaque rayon de la liste de rayons :

Compter le nombre de véhicules disponibles pour ce rayon selon le type de véhicule (essence, électrique) et le type de service (ffcs ou sbcs)

Ajouter une colonne dans le fichier de réservations pour le nombre de véhicules disponibles pour ce rayon selon le type de véhicule et le type de service

Notons que certains véhicules en libre-service intégral ne seront pas comptés dans le calcul des effectifs par rayon, car leur position n'a pu être obtenue à partir du fichier des événements

CHAPITRE 4 TABLEAU DES INDICATEURS

Dans ce chapitre, nous présentons le tableau des indicateurs qui a été développé et dont la plupart ont été intégrés à l'outil de visualisation.

4.1 Définitions

À partir des travaux de Martin (2007) nous avons pu définir les termes utilisés dans la suite :

- Abonné : membre du service Communauto qui est autorisé à conduire les véhicules mis à sa disposition par l'entreprise. L'abonné principal est l'abonné répondant et une seconde personne peut être associée au compte en tant que co-abonné. Ces personnes doivent demeurer au même domicile.
- Réservation : Une plage horaire réservée par un abonné pour un véhicule choisi. Le statut d'une réservation peut être actif ou annulé. Tant et aussi longtemps qu'elle est active, la voiture n'est pas disponible pour un autre abonné à la même période. Les réservations peuvent seulement débuter à l'heure pile ou aux 15 minutes. La plage horaire doit être de 30 minutes minimalement et elle peut s'échelonner jusqu'à une durée de 30 jours.
- Transaction : Une réservation active ou annulée qui apparaît sur la facture mensuelle d'un abonné.
- Utilisation : Une réservation active qui affiche un kilométrage non nul. Ce qui signifie que la voiture a été utilisée par l'abonné.
- Véhicule : Une automobile appartenant à l'opérateur d'autopartage.
- Station : Lieu où un véhicule appartenant à Communauto doit être récupéré et retourné. Chaque véhicule est assigné à une seule station.

On dira que l'indicateur a une portée opérationnelle lorsque la période d'observation sera plutôt courte comme pour le cas d'une journée par exemple. Il permet donc de se corriger ou de s'améliorer au fur et à mesure qu'on avance dans notre processus de production d'un service. C'est un indicateur important pour l'équipe, au point de l'influencer dans ses actions quotidiennes.

On dira que l'indicateur est tactique lorsqu'on l'utilise pour évaluer des performances sur le moyen terme et qu'il a une portée stratégique lorsqu'on l'utilise pour évaluer des performances sur le long

terme. Les indicateurs de performance stratégiques permettent de vérifier que l'organisation a atteint les objectifs fixés par la direction.

À partir des termes définis, nous avons pu formaliser à l'aide de ces différents ensembles :

$\mathbb{R} = \{u\}$: l'ensemble des réservations enregistrées sur la période considérée.

$\mathbb{U} = \{u \mid d > 0\}$: l'ensemble des utilisations faites sur la période considérée.

$U_a = \{u_a\}$: l'ensemble des utilisations faites par l'abonné a

$U_v = \{u_v\}$: l'ensemble des utilisations faites avec le véhicule v

$U_s = \{u_s\}$: l'ensemble des utilisations faites à partir de la station s

$A_r = \{a \mid \exists U_a \in \mathbb{R}\}$: l'ensemble des abonnés ayant fait au moins une réservation sur la période observée.

$A = \{a \mid \exists U_a \in \mathbb{U}\}$: l'ensemble des abonnés ayant fait au moins une utilisation sur la période observée.

$V = \{v \mid \exists U_v \in \mathbb{U}\}$: l'ensemble des véhicules ayant été utilisés sur la période observée.

$S = \{s \mid \exists U_s \in \mathbb{U}\}$: l'ensemble des stations ayant été utilisées.

4.2 Tableau

Voici donc le tableau des indicateurs qui a été créé. Les sources du tableau sont présentées à la suite.

* : Présent dans l'outil de visualisation des indicateurs

ER : Pouvant être calculé avec le fichier d'état du réseau

CR : Pouvant être calculé avec le fichier des circonstances

Tableau 4.1 : Tableau des indicateurs

Indicateur	TRAD	SUS	LSI	Formule	Usager	Vehicule	Station	Réservation	Source
Durée totale des utilisations*, CR	X			$T_a = \sum_{U_a} T_u$	X				[1]
Durée moyenne des utilisations*, CR	X			$t_{moy} = \frac{T_a}{U_a}$	X				[1], [18]
Durée minimum des utilisations*, CR	X			$Min(T_u)$	X				[1]
Durée maximum des utilisations*, CR	X			$Max(T_u)$	X				[1]
Distance totale des utilisations*, CR	X			$D_a = \sum_{U_a} D_u$	X				[1], [12], [14], [15], [16], [17]
Distance moyenne des utilisations*, CR	X			$d_{moy} = \frac{D_a}{U_a}$	X				[1], [12], [17], [18], [19]
Distance minimum des utilisations*, CR	X			$Min(D_u)$	X				[1]
Distance maximum des utilisations*, CR	X			$Max(D_u)$	X				[1]
Fréquence d'utilisation du service par l'utilisateur*, R	X			$f_{ua} = \frac{U_a}{nj}$	X				[1] [12]
Fidélité à une voiture	X			$\frac{U_a}{V_a}$	X	X			[1]
Fidélité à une station	X			$\frac{U_a}{S_a}$	X		X		[1]
Disponibilité des véhicules autour de l'abonné	X				X	X			[1]
Durée totale des utilisations*, CR	X			$T_v = \sum_{U_v} T_u$		X			[1] [18]
Durée moyenne des utilisations*, CR	X			$t_{moyv} = \frac{T_v}{U_v}$		X			[1] [18]
Durée minimum des utilisations*, CR	X			$Min(T_u)$		X			[1]
Durée maximum des utilisations*, CR	X			$Max(T_u)$		X			[1]
Distance totale des utilisations*, CR	X			$D_v = \sum_{U_v} D_u$		X			[1]

Table 4.1 : Tableau des indicateurs (suite)

Indicateur	TRAD	SUS	LSI	Formule	Usager	Véhicule	Station	Réservation	Source
Distance moyenne des utilisations *, CR	X			$d_{moy} = \frac{D_v}{U_v}$		X			[1]
Distance minimum des utilisations *, CR	X			$Min(D_u)$		X			[1]
Distance maximum des utilisations*, CR	X			$Max(D_u)$		X			[1]
Fréquence d'utilisation du véhicule*, ER	X			$f_{uv} = \frac{U_v}{nj}$		X			[1]
Capacité opérationnelle du véhicule	X			$C_v = 24nj$		X			[1]
Taux d'utilisation du véhicule*, ER	X			$T_v = \frac{T_v}{C_v}$		X			[1], [18], [10]; [11]
Nombre d'utilisations à la station*,ER	X			U_s			X		[1]
Nombre d'abonnés différents ayant utilisé la station	X			A_s	X		X		[1]
Capacité observée de la station	X			V_s	X		X		[1]
Temps total des utilisations faites à partir de la station*,ER	X			$T_s = \sum_{v \in S} T_v v$	X		X		[1]
Distance totale des utilisations faites à partir de la station*,CR	X			$D_s = \sum_{v \in S} D_v v$	X		X		[1]
Ratio abonné par véhicule à la station	X			$R_{av} = \frac{A_s}{V_s}$	X		X		[1]
Fidélité des usagers à la station	X				X		X		[1]
Dispersion spatiale des utilisateurs autour de la station	X				X		X		[1]
Capacité opérationnelle de la station	X			$C_s = \sum_{v \in S} C_v v$		X	X		[1]

Table 4.1 : Tableau des indicateurs (Suite et fin)

Indicateur	TRAD	SUS	LSI	Fornule	Usager	Véhicule	Station	Réservation	Source
Taux d'utilisation de véhicule de la station	X			$\mathcal{T}_v = \frac{T_s}{C_s}$	X		X		[1]
taille de la flotte active (par jour, mois, année)		X				X			[2], [3]
Véhicules par station (jour, mois, année)		X				X	X		[2], [3], [12]
Taux d'acceptation (jour, mois, année)		X							[4],[5]
Utilisation Moyenne de la flotte (jour, mois, année) *, ER		X				X	X		[6], [7], [13]
ratio véhicules par réservations (par jour)	X								[8]
Temps avec zéro véhicule, ER	X						X		[9]
Temps de réservation	X							X	[8]

Tableau 4.2 : Références des indicateurs du tableau

Référence	Auteurs
1	Martin, 2007
2	Correia and Antunes, 2012
3	Correia et al., 2014
4	Alfian et al., 2017
5	Nourinejad et al., 2015
6	Alfian, Rhee, Kang, & Yoon, 2015
7	El Fassi et al., 2012
8	Barth and Todd, 1999
9	Kek et al., 2009
10	Alfian et al., 2015
11	Nourinejad & Roorda, 2014
12	Morency, Trépanier & Agard, 2011
13	Morency, Trépanier & Martin, 2008
14	Baumgarte et al., 2022
15	Mishra et al., 2015

Table 4.2 : Références des indicateurs du tableau (suite et fin)

Référence	Auteurs
16	Costain et al., 2012
17	Kopp et al., 2015
18	Habibi et al, 2017
19	Schmidt et al, 2015

Le tableau 4.2 fournit une liste d'indicateurs pour évaluer différents aspects de l'utilisation d'un service d'autopartage, avec des colonnes pour le nom de chaque indicateur, sa définition (si nécessaire), et les différentes variables nécessaires pour calculer l'indicateur. Les variables incluent des mesures telles que la durée et la distance totales des utilisations, la fréquence d'utilisation par l'utilisateur, la fidélité à une voiture ou une station, la disponibilité des véhicules et la capacité opérationnelle. Les colonnes indiquent également si les variables sont liées à l'utilisateur, au véhicule ou à la station.

Les indicateurs comprennent des mesures de temps et de distance pour les utilisations totales, moyennes, minimales et maximales, ainsi que des mesures de la fréquence d'utilisation et de la fidélité à une voiture ou une station. Des indicateurs spécifiques à la station incluent le nombre d'utilisations, le nombre d'abonnés différents ayant utilisé la station, la capacité observée de la station, le temps total des utilisations à partir de la station, la distance totale des utilisations à partir de la station, le ratio d'abonnés par véhicule à la station, la fidélité des usagers à la station, la dispersion spatiale des utilisateurs autour de la station, la capacité opérationnelle de la station et le taux d'utilisation du véhicule de la station.

Le tableau comprend également des indicateurs pour la taille de la flotte active, le nombre de véhicules par station, le taux d'acceptation, l'utilisation moyenne de la flotte, le ratio de véhicules par réservation, le temps sans véhicule, le taux d'utilisation d'un véhicule et le temps de réservation. Les indicateurs sont accompagnés de leurs formules mathématiques respectives pour permettre le calcul des mesures en fonction des données recueillies.

Nous pouvons ainsi définir certains indicateurs ici :

- Capacité opérationnelle de la station : c'est un indicateur qui mesure le nombre maximum de véhicules pouvant être garés à une station donnée. Il est utilisé pour évaluer la capacité de la station à répondre aux besoins des utilisateurs et pour identifier les stations qui ont besoin d'une expansion ou d'une révision.

- Taux d'utilisation de véhicule : c'est un indicateur qui mesure la proportion de temps durant lequel un véhicule est utilisé par rapport au temps durant lequel il est disponible pour être utilisé. Il est souvent utilisé pour évaluer l'efficacité des systèmes d'autopartage et pour identifier les véhicules qui sont sous-utilisés ou surutilisés.
- Dispersion spatiale des utilisateurs autour de la station : c'est un indicateur qui mesure la distance géographique entre les utilisateurs d'un système d'autopartage et les stations. Il est utilisé pour évaluer la pertinence des emplacements des stations et identifier les zones où il y a un besoin de stations supplémentaires.
- Taux d'acceptation : c'est un indicateur qui mesure la proportion de demandes de réservation d'un véhicule ou d'une station qui sont acceptées par rapport au nombre total de demandes. En général, dans les études de performance de système d'autopartage, le taux d'acceptation est souvent utilisé pour évaluer la performance des systèmes d'autopartage et identifier les véhicules ou les stations qui sont les plus populaires. Il permet également d'identifier les problèmes de disponibilité ou de déséquilibre des véhicules dans les stations.
- Fréquence d'utilisation du service par l'utilisateur : nombre d'utilisations par unité de temps (par exemple par jour, par mois, par année)
- Fidélité à une voiture : nombre de fois où l'utilisateur a utilisé le même véhicule sur une période donnée
- Fidélité à une station : nombre de fois où l'utilisateur a utilisé la même station sur une période donnée
- Disponibilité des véhicules autour de l'abonné : temps moyen nécessaire pour qu'un véhicule soit disponible à proximité de l'utilisateur
- Capacité opérationnelle du véhicule : nombre de fois où le véhicule a été utilisé sur une période donnée, rapporté à sa durée de disponibilité
- Nombre d'utilisations à la station : nombre de fois où un utilisateur a utilisé une station sur une période donnée
- Nombre d'abonnés différents ayant utilisé la station : nombre d'utilisateurs différents ayant utilisé la station sur une période donnée
- Ratio abonné par véhicule à la station : nombre d'abonnés rapporté au nombre de véhicules disponibles dans la station sur une période donnée

- Fidélité des usagers à la station : nombre de fois où un utilisateur a utilisé la même station sur une période donnée, rapporté au nombre total d'utilisations
- Taille de la flotte active (par jour, mois, année) : C'est un indicateur qui mesure le nombre de véhicules disponibles pour une période donnée (jour, mois, année). Cet indicateur permet de connaître la capacité totale de la flotte et de planifier les besoins en termes d'entretien et d'achat de nouveaux véhicules.
- Véhicules par station (jour, mois, année) : C'est un indicateur qui mesure le nombre de véhicules disponibles dans une station à un moment donné (jour, mois, année). Cet indicateur permet de déterminer si une station a suffisamment de véhicules pour répondre à la demande des utilisateurs.
- Utilisation moyenne de la flotte (jour, mois, année) : C'est un indicateur qui mesure la proportion de véhicules disponibles qui ont été utilisés pour une période donnée (jour, mois, année). Cet indicateur permet de mesurer l'efficacité de la flotte dans son ensemble et de déterminer si la flotte est suffisamment utilisée.
- Ratio véhicules par réservations (par jour) : C'est un indicateur qui mesure le nombre de réservations effectuées par rapport au nombre de véhicules disponibles pour une journée donnée. Cet indicateur permet de déterminer si le nombre de véhicules disponibles est suffisant pour répondre à la demande des utilisateurs.
- Temps avec zéro véhicule : C'est un indicateur qui mesure la durée pendant laquelle aucune voiture n'est disponible dans une station donnée. Cet indicateur permet de déterminer s'il y a des problèmes avec la disponibilité des véhicules dans une station donnée.
- Temps de réservation : C'est un indicateur qui mesure le temps moyen nécessaire pour réserver un véhicule. Cet indicateur permet de mesurer l'efficacité du système de réservation et de déterminer s'il y a des problèmes dans le processus de réservation.

Le fichier des circonstances qui comprend toutes les réservations avec le nombre de véhicules disponibles autour au moment de la réservation, les identifiants des utilisateurs, les horodatages et les informations de station peut grandement aider à déterminer la fréquence d'utilisation d'un véhicule par un client. Les identifiants d'utilisateur permettent de filtrer par utilisation spécifique du client, tandis que les horodatages peuvent déterminer quand un véhicule a été utilisé, permettant

ainsi de calculer la fréquence par jour, mois ou année. Les informations sur la durée et la distance de la réservation qui sont présentes dans le fichier permettent aussi par des sommes et des agrégations de calculer des indicateurs sur la distance et la durée tels que la somme, la moyenne, le maximum et le minimum. Les filtres permettent de rajouter encore plus de possibilités.

Le fichier de l'état du réseau aussi permet, de par ses informations sur l'état des véhicules du réseau de calculer des indicateurs touchant à l'utilisation des véhicules tels que L'utilisation moyenne de la flotte (jour, mois, année), le temps avec zéro véhicule, le temps total des utilisations faites à partir de la station, la fréquence d'utilisation du véhicule.

CHAPITRE 5 VISUALISATION

Dans ce chapitre, nous présentons l'outil de visualisation développé dans le cadre de cette recherche.

5.1 Présentation de Panel Holoviz

Dans ce travail, nous avons utilisé *Panel Holoviz*. Il existe de nombreux outils pour créer des visualisations interactives en Python, mais Panel est l'un des plus complets et des plus flexibles. Il est basé sur les bibliothèques *HoloViews* et *Bokeh*. Il offre une variété de composants pour créer des applications web à partir de données et de visualisations. Nous pouvons trouver la documentation ici : <https://panel.holoviz.org/>

Panel permet de créer des applications web en utilisant un ensemble de composants tels que des boutons, des cases à cocher, des menus déroulants, etc. Ces composants peuvent être liés à des visualisations pour créer des interactions. Cela permet de créer des applications web de visualisation de données interactives qui peuvent être partagées facilement avec d'autres personnes.

Une des caractéristiques les plus intéressantes de Panel est sa capacité à combiner des visualisations provenant de différentes bibliothèques, comme *Matplotlib*, *Bokeh* et *HoloViews*. Cela permet de créer des visualisations complexes à partir de plusieurs types de données, et de les intégrer facilement dans une application web.

Un autre avantage de Panel est qu'il est compatible avec les bibliothèques de calculs en parallèle, comme *Dask* et *CuPy*, pour manipuler de grandes quantités de données.

Enfin, Panel offre une variété de fonctionnalités pour personnaliser les visualisations, comme des options de formatage pour les axes, les étiquettes, les titres et les légendes. Il permet également de créer des visualisations animées et de les intégrer dans des applications web.

En résumé, Panel est un outil puissant pour créer des applications web interactives à partir de données en utilisant une variété de composants et de visualisations. Il offre une flexibilité pour combiner des visualisations provenant de différentes bibliothèques, et il est compatible avec des bibliothèques de calculs en parallèle pour manipuler de grandes quantités de données. Il peut être utilisé pour créer des visualisations animées et des applications de visualisation de données interactives pour les partager facilement avec d'autres personnes.

Panel est en développement actif et la dernière version stable est la version 0.11.0. Il est open source et son code source est disponible sur GitHub. Le projet est soutenu par la communauté *HoloViz*, qui comprend des contributeurs de différents organismes tels que *Anaconda*, *Continuum Analytics*, et *PyViz*.

La communauté *Panel* est en train d'ajouter de nouvelles fonctionnalités pour améliorer l'expérience utilisateur, rendre la création d'applications plus facile et améliorer les performances. Il y a des améliorations en cours pour la mise en page des *widgets*, la prise en charge de la gestion de l'état de l'application, l'intégration de nouveaux types de visualisations. Il y a également une tendance croissante à intégrer les fonctionnalités d'autres bibliothèques telles que *Plotly*, *Altair*, etc.

Il y a également des mises à jour régulières pour résoudre les bogues et améliorer la compatibilité avec les dernières versions de Python et de ses bibliothèques. Il est important de noter qu'il est recommandé de toujours utiliser la dernière version stable de *Panel* pour bénéficier des dernières fonctionnalités et des correctifs de bogues.

5.2 Étapes de réalisation du tableau de bord de visualisation des indicateurs

Les différentes étapes suivantes ont été suivies dans le but de développer chacun des tableaux de bord dont nous allons parler.

1. Récupération des données :

Il faut d'abord récupérer les données relatives aux indicateurs de performance du service d'autopartage, telles que le nombre de réservations, le nombre de véhicules disponibles. Ces données sont stockées dans des fichiers de données (CSV).

2. Prétraitement des données :

Il est nécessaire de nettoyer, transformer et organiser les données pour les rendre utilisables pour les visualisations.

3. Choix de la visualisation des données :

En utilisant la bibliothèque de visualisation de Panel (*hvplot*), il est possible de créer des visualisations pour afficher les différents indicateurs de performance. Les visualisations peuvent être des graphiques à barres, des courbes, des camemberts, etc. Il est possible de combiner des visualisations pour créer des tableaux de bord complets.

4. Ajout des *widgets* de paramétrage :

Pour créer une interaction avec l'utilisateur, il est possible d'ajouter des *widgets* tels que des menus déroulants, des cases à cocher, des boutons, etc. Ces *widgets* peuvent être liés aux visualisations pour permettre à l'utilisateur de filtrer les données ou de changer les options d'affichage.

5. Création de l'application :

En utilisant les étapes précédentes, il est possible de créer une application web interactive qui affiche les indicateurs de performance du service d'autopartage. Il est possible de personnaliser l'apparence de l'application en utilisant des thèmes et des options de formatage. Il est également possible de définir des options de configuration pour les *widgets*.

5.3 Visualisation de quelques indicateurs de performances des objets

Dans cette section, nous expliquons comment a été montée l'interface de visualisation des indicateurs sous forme de graphe.

5.3.1 Besoins de visualisation

Ce tableau de bord permet de visualiser un ensemble d'indicateurs pour l'autopartage. Il utilise plusieurs paramètres pour permettre aux utilisateurs de filtrer et de sélectionner les données qu'ils souhaitent visualiser. Voici ci-après les éléments à considérer.

Le premier paramètre est le "sélecteur", qui permet à l'utilisateur de choisir entre la visualisation des données pour les véhicules ou les utilisateurs. Cela est important, car, en plus de vouloir faire un choix sur l'objet de notre analyse, les données pour les véhicules et les utilisateurs ont des attributs différents. Il peut prendre les valeurs : "Véhicule" ou "Usager".

Le paramètre "temporalité" est utilisé pour sélectionner la granularité temporelle des données, que ce soit par jour, semaine ou mois. Cela est important, car l'utilisateur peut souhaiter voir les données à différents niveaux de détail et la granularité temporelle affecte la visualisation résultante. Il peut prendre les valeurs : "Jour", "Semaine" ou "Mois".

Le paramètre "fenêtre temporelle" est un sélecteur de plage de dates qui permet à l'utilisateur de sélectionner une fenêtre de temps spécifique pour les données. Cela est important, car l'utilisateur peut ne s'intéresser qu'aux données d'une période spécifique et vouloir filtrer les données en dehors de cette fenêtre.

Le paramètre "filtre" permet à l'utilisateur de spécifier le type de filtre qu'il souhaite appliquer aux données. Cela est important, car l'utilisateur peut souhaiter filtrer les données en fonction de différents attributs, tels que le modèle du véhicule ou la langue de l'utilisateur. Les attributs à filtrer vont dépendre de l'objet sélectionné avec le paramètre "sélecteur".

Le paramètre "type d'indicateur" permet à l'utilisateur de sélectionner le type d'indicateur qu'il souhaite visualiser, comme les distances ou les durées d'utilisation. Cela est important, car l'utilisateur peut être intéressé par différentes métriques et souhaiter basculer entre elles. Il peut prendre les valeurs : "Distances d'utilisation" ou "Durées d'utilisation".

Le paramètre "agrégat" permet à l'utilisateur de sélectionner le type d'agrégation qu'il souhaite appliquer aux données. Il peut prendre les valeurs : "Somme", "Moyenne", "Minimum" ou "Maximum".

5.3.2 Données requises en entrée et formalisation

Pour calculer les indicateurs à partir du jeu de données (**circonstances**) selon les paramètres entrés, une fonction est définie. Les types de KPI sont basés sur les distances d'utilisation ou les durées d'utilisation des objets véhicule ou utilisateurs. Les utilisateurs peuvent définir les paramètres présentés dans 5.3.1.

La fonction commence par convertir la fréquence de temps spécifiée en un format utilisable par *Pandas*. Ensuite, le jeu de données est filtré pour ne conserver que les données comprises dans la plage de temps spécifiée. Le jeu de données filtré est ensuite filtré en fonction du type de service, et si des choix sont spécifiés pour un autre filtre (par exemple, une ville), le jeu de données sera filtré en conséquence.

Ensuite, selon le type de KPI spécifié (distance d'utilisation ou durée d'utilisation), les données sont agrégées en utilisant la fonction d'agrégation spécifiée (somme, moyenne, minimum ou maximum).

Si le type d'indicateur est "Distances d'utilisation", on peut définir le KPI comme suit :

- Si agrégat = "Somme", on peut définir le KPI comme la somme des distances parcourues pour chaque période de temps définie par *time_freq*:

$$KPI = \sum_{t \in \text{time_freq}} \text{distance parcourue}_t$$

- Si agrégat = "Moyenne", on peut définir le KPI comme la moyenne des distances parcourues pour chaque période de temps définie par *time_freq*.

$$KPI = \frac{1}{|\text{time_freq}|} \sum_{t \in \text{time_freq}} \text{distance parcourue}_t$$

- Si agrégat = "Minimum", on peut définir le KPI comme la valeur minimale des distances parcourues pour chaque période de temps définie par *time_freq*:

$$KPI = \min_{t \in \text{time_freq}} \text{distance parcourue}_t$$

- Si agrégat = "Maximum", on peut définir le KPI comme la valeur maximale des distances parcourues pour chaque période de temps définie par *time_freq*:

$$KPI = \max_{t \in \text{time_freq}} \text{distance parcourue}_t$$

Si le type d'indicateur est "Durées d'utilisation", on peut définir le KPI de la même manière en utilisant les durées de la réservation au lieu des distances parcourues.

Les données agrégées sont ensuite présentées sous forme de graphique en utilisant la bibliothèque *hvplot*.

Enfin, la fonction renvoie le graphique créé avec un titre détaillé incluant la plage de temps, le type de service, le filtre de sélection et les choix, ainsi que le type de KPI et l'agrégat utilisés.

5.3.3 Réalisation de la plateforme sur *Panel*

Dans cette partie nous montrons la façon dont la plateforme a été réalisée à l'aide de la plateforme *Panel*.

5.3.3.1 Récupération et prétraitement des données

Avant de s'occuper de l'interface de visualisation il est important de préparer la base de données à partir de laquelle notre tableau de bord va créer ses visualisations.

La première étape consiste à lire le fichier CSV des circonstances contenant les réservations du mois de janvier avec les véhicules disponibles par rayon en utilisant la fonction *pd.read_csv()* de la librairie *Pandas* de *Python* et on stocke le résultat dans une variable. La colonne "datetime" du fichier, initialement sous format de chaîne de caractères lorsque ce dernier est importé, est converti en format de type "datetime" en utilisant la fonction *pd.to_datetime()*.

Ensuite le fichier multi des réservations contenant toutes les colonnes de réservation est chargé en utilisant la fonction *pd.read_csv()* et stocké dans une autre variable

Les deux *dataframes* sont fusionnées en utilisant la fonction *pd.merge()* dans le but d'avoir le maximum d'informations avec le fichier des circonstances. La fusion se fait sur la colonne "res_id".

Dans le souci de franciser les données, un dictionnaire est défini qui contient les nouveaux noms pour les colonnes du *dataframe* fusionné, un autre est aussi créé pour les données dans les lignes.

On utilise la fonction *rename()* pour renommer les colonnes et *replace()* pour renommer les données dans les lignes du *dataframe* fusionné.

Le résultat obtenu à la suite nous permettra d'aller chercher les indicateurs dont nous avons besoin pour notre tableau de bord.

5.3.3.2 Ajout des *widgets* de paramétrage

Dans le but de construire le tableau de bord, autant pour le choix des paramètres que pour l'organisation de l'interface, les *widgets* suivants de *Panel* ont été utilisés.

Select : C'est un *widget* de sélection qui permet à l'utilisateur de choisir une option parmi plusieurs options. Il est utilisé pour sélectionner "Objet", "Temporalité", "Filtres véhicules" ou "Filtres usagers", "Type indicateur", "Agrégat".

DatetimeRangePicker : C'est un *widget* de sélection de plage de temps qui permet à l'utilisateur de choisir une plage de temps. Il est utilisé pour sélectionner l'intervalle de temps.

IntInput : C'est un *widget* d'entrée de nombre entier qui permet à l'utilisateur de saisir un nombre entier. Il est utilisé pour entrer le Type d'utilisateur.

MultiChoice : C'est un *widget* de choix multiple qui permet à l'utilisateur de choisir plusieurs options à partir d'une liste. Il est utilisé pour choisir les filtres véhicules ou usagers selon le choix du sélecteur.

CheckBoxGroup : C'est un groupe de cases à cocher qui permet à l'utilisateur de sélectionner une ou plusieurs options à partir d'une liste. Il est utilisé pour choisir le type de service.

WidgetBox : C'est un *widget* qui agit comme une boîte qui peut contenir d'autres *widgets*. Il est utilisé pour regrouper les *widgets* qui sont utilisés pour les indicateurs (véhicules/utilisateurs).

Column : C'est un *widget* qui agit comme une colonne et peut contenir d'autres *widgets*. Il est utilisé pour organiser les *widgets* côte à côte.

Pane : C'est un *widget* qui agit comme un panneau qui peut contenir d'autres *widgets*. Il est utilisé pour afficher le graphique généré.

FastListTemplate : C'est un modèle de liste rapide qui peut inclure un titre, un fond d'accentuation de couleur de base, un fond d'en-tête de couleur, une barre latérale et un contenu principal. Il est utilisé pour organiser et afficher l'interface utilisateur.

Indicateurs
(Véhicule/Usager)

Objet

Vehicule

Temporalité

jour

Intervalle de temps

2021-01-02 00:00:00 to 2021-01-31 23:59:00

☒ flex ☒ basé station

Filtres véhicules

type de véhicule

type de véhicule

Type indicateur

Distances d'utilisation

Aggregat

Somme

Figure 5.1 : *Widget* des paramètres

5.3.3.3 Création de l'application

Dans cette partie nous expliquons la logique derrière l'affichage du tableau de bord.

Après avoir défini les différents *widgets* qui permettent de gérer le choix de l'indicateur et les paramètres, une fonction "update_widget" est créée. Celle-ci détermine l'affichage des *widgets* en fonction du choix de l'objet d'analyse (véhicule ou utilisateur). Si l'utilisateur sélectionne "véhicule", les options de filtre seront différentes de celles pour les utilisateurs.

Le tableau de bord montre les options de filtre selon l'objet choisi. Pour cela, une fonction "update_filters" est définie pour mettre à jour les options de filtre en fonction de la sélection de l'utilisateur dans la boîte de filtre.

La partie définit une fonction dépendante qui utilise les paramètres sélectionnés pour générer un graphique à partir des données en entrée.

Enfin, un modèle "template" est défini pour afficher tous les *widgets* et les graphiques sur la plateforme. Le modèle utilise un modèle "FastListTemplate" pour définir le titre, la barre latérale avec les *widgets*, le contenu principal avec le graphique généré et les couleurs de l'accentuation et de l'arrière-plan de l'en-tête.

Lorsque le code est exécuté, l'interface graphique sera affichée et les utilisateurs pourront interagir avec les *widgets* pour filtrer et afficher les données dans un graphique.

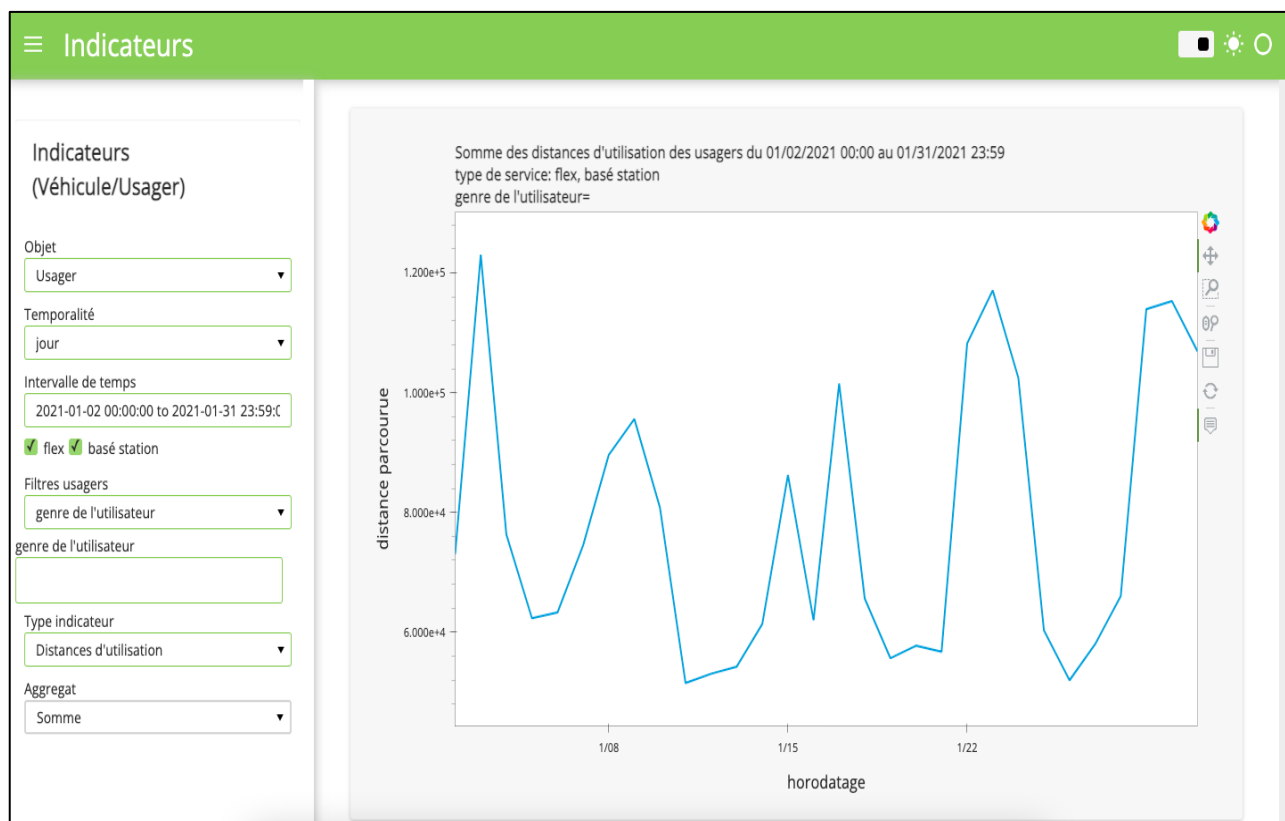


Figure 5.2: Interface de visualisation des indicateurs sous forme de graphe

La figure montre l'interface de visualisation de la première page du tableau de bord, nous trouvons ainsi sur une barre latérale les différents paramètres qui nous permettent de calculer des indicateurs

ainsi que les filtres. Sur le panneau principal, nous avons le graphique en question avec les noms de ses axes qui sont dynamiques tout comme le titre. Le titre sera personnalisé selon les paramètres et les filtres choisis. L'interface permet aussi à l'utilisateur d'effectuer un zoom sur le graphique en sélectionnant une portion du graphique à mettre en valeur. Elle permet aussi de télécharger le graphique sur l'appareil de l'utilisateur. D'autres exemples de visualisations de ce tableau de bord sont les suivantes :

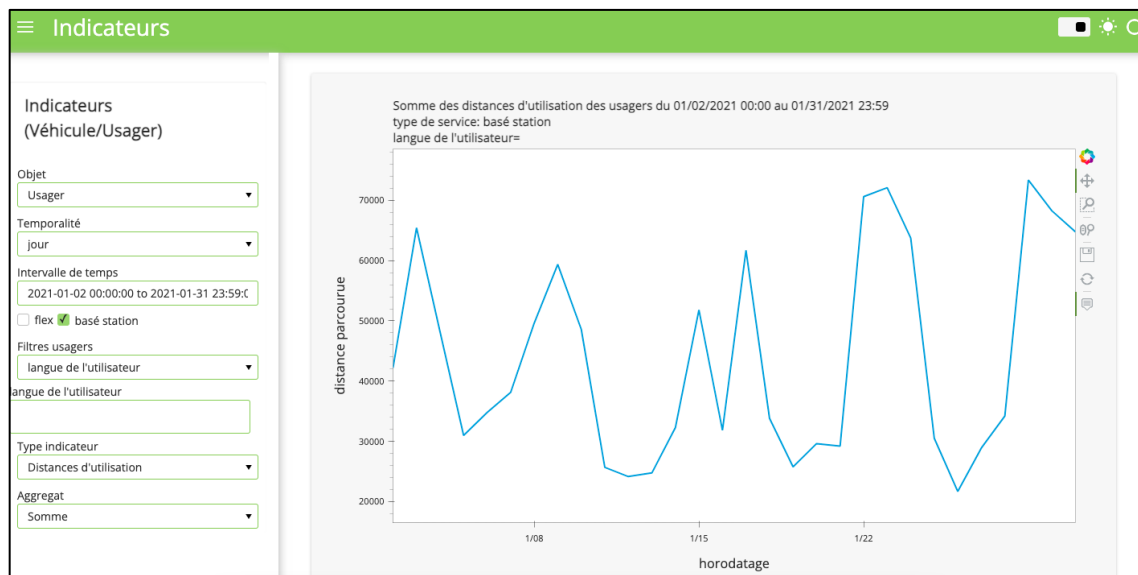


Figure 5.3 : Distances d'utilisation des usagers du service basé station

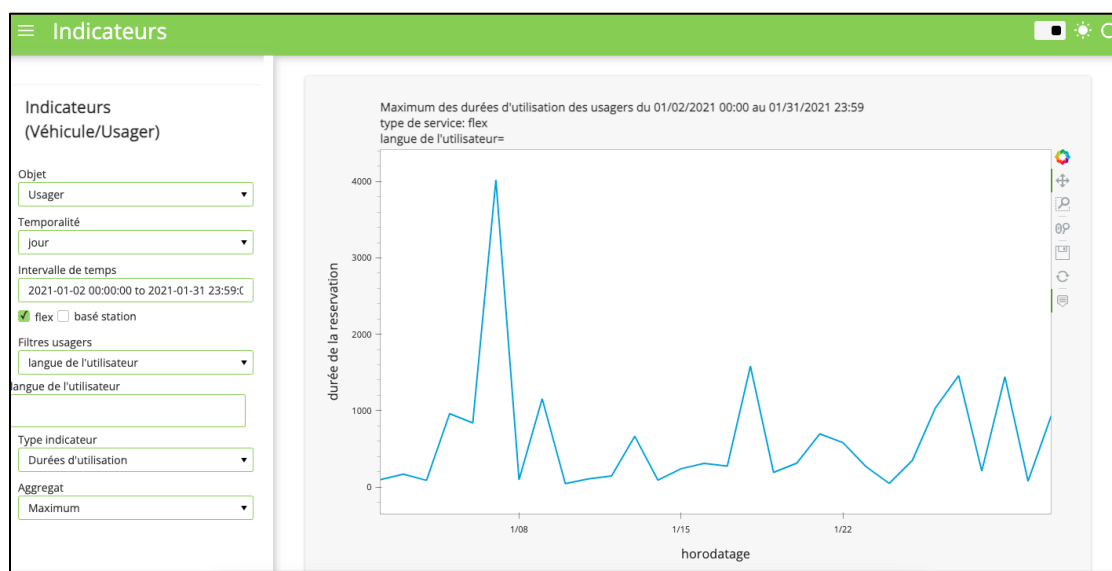


Figure 5.4 : Maximum des durées d'utilisation des usagers flex

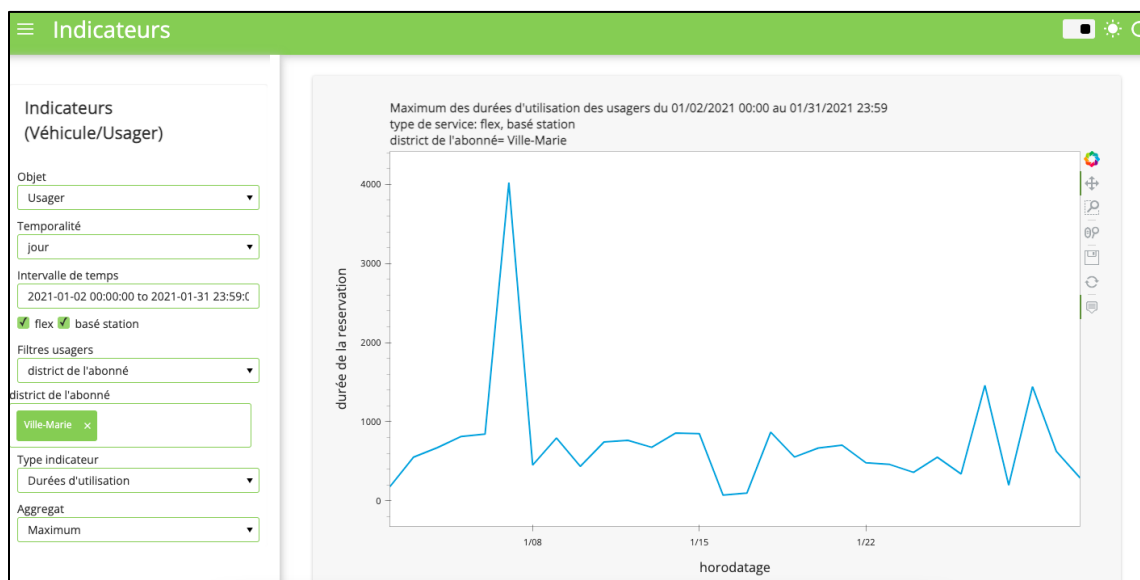


Figure 5.5 : Maximum des durées d'utilisation des usagers dans le district Ville-Marie

Nous pouvons avoir ci-après un arbre représentant les différentes possibilités.

Tableau de bord

+-- Objet d'Intérêt

+-- Véhicule

+-- Temporalité

+-- (Jour – Semaine – Mois)

+-- Distance d'Utilisation

+-- Somme

+-- Moyenne

+-- Minimum

+-- Maximum

+-- Durée d'Utilisation

+-- Somme

+-- Moyenne

+-- Minimum

+-- Maximum

+-- Intervalle de Temps

+-- Date/Heure de Début et de Fin

+-- Filtres

```

| | |
| | +-- Nom du Modèle de Véhicule
| | +-- Type de Véhicule
| | +-- Année du Véhicule
| | +-- Type de service (Flex, Basé station)
+-- Utilisateur
|
| +-- Temporalité
| |
| | +-- (Jour – Semaine – Mois)
| | |
| | | +-- Distance d'Utilisation
| | | |
| | | | +-- Somme
| | | | +-- Moyenne
| | | | +-- Minimum
| | | | +-- Maximum
| | |
| | | +-- Durée d'Utilisation
| | | |
| | | | +-- Somme
| | | | +-- Moyenne
| | | | +-- Minimum
| | | | +-- Maximum
| |
| +-- Intervalle de Temps
| |
| | +-- Date/Heure de Début et de Fin
|
| +-- Filtres
| |
| | +-- Genre de l'Utilisateur
| | +-- Langue de l'Utilisateur
| | +-- Date de Naissance de l'Utilisateur
| | +-- Date d'Abonnement de l'Utilisateur
| | +-- Début de l'Abonnement
| | +-- District de l'Abonné
| | +-- Type de service (Flex, Basé station)

```

5.4 Visualisation de la carte de l'état du réseau

Dans cette section, nous expliquons comment a été montée l'interface de visualisation de l'état du réseau sous forme de cartes.

5.4.1 Besoins de visualisation

Une carte représentant les véhicules libres par heure précise de la journée est importante pour plusieurs raisons. D'abord elle permet aux gestionnaires du système d'autopartage de voir à quelle

heure les véhicules sont le plus souvent disponibles et leur localisation pour ainsi optimiser leur utilisation en les déplaçant éventuellement vers des endroits où ils sont plus nécessaires.

Aussi, la carte aide les gestionnaires à mieux planifier la maintenance des véhicules en fonction des périodes d'utilisation les plus intenses. Cela peut réduire les coûts de maintenance et améliorer la disponibilité des véhicules pour les utilisateurs.

En résumé, la carte représentant les véhicules libres par heure de la journée est un outil précieux pour optimiser l'utilisation des véhicules d'autopartage, améliorer l'expérience des utilisateurs et évaluer la performance du système.

L'objectif de ce tableau de bord est d'avoir une page qui va donner une capture de la carte du réseau avec les positions des différents véhicules libres selon les paramètres choisis. Dans celui-ci nous avons deux paramètres que sont la date et le type de service.

Le paramètre "date" est un sélecteur de date qui permet à l'utilisateur de sélectionner une date spécifique pour les données.

Le paramètre "type de service" est un paramètre d'entrée qui permet à l'utilisateur de spécifier le type de service qu'il souhaite filtrer. Cela est important, car l'utilisateur peut ne souhaiter voir les données que pour un type spécifique de service à savoir le Flex ou le basé station.

5.4.2 Données requises en entrée et formalisation

Pour réaliser ces cartes représentant des captures du réseau à des dates précises, il nous faut avoir le fichier de l'état du réseau et les différents paramètres. Ces derniers permettront juste d'appliquer des filtres lors de sélection de la base de données pour représenter les emplacements des véhicules sur la carte.

5.4.3 Réalisation de la plateforme sur Panel

5.4.3.1 Récupération et prétraitement des données

Tout d'abord le fichier CSV contenant l'état du réseau est chargé et stocké dans un objet pandas appelé DataFrame. La fonction `pd.read_csv` est utilisée pour ce faire.

Ensuite le DataFrame est filtré pour n'inclure que les lignes où la valeur dans la colonne "état" est égale à "libre".

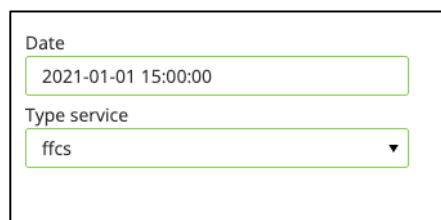
Vu que le champ de l'horodatage est chargé sous format de chaîne de caractères, la troisième étape consiste à convertir les valeurs de la colonne en objets "datetime". La fonction `pd.to_datetime` est utilisée pour le faire. Les valeurs résultantes sont stockées dans la même colonne. Les objets datetime sont un type de données couramment utilisé dans les tâches d'analyse et de manipulation de données pour représenter les dates et les heures.

5.4.3.2 Ajout des *widgets* de paramétrage

Les *widgets* Panel suivants ont été utilisés pour construire le tableau de bord, à la fois pour choisir les paramètres et organiser l'interface.

param.Date : Il s'agit d'un type de *widget* de la bibliothèque **param** en Python. Il permet à l'utilisateur de sélectionner une date à partir d'un widget de calendrier.

param.ObjectSelector : Il s'agit d'un autre type de *widget* de la bibliothèque **param** en Python. Il est utilisé pour fournir une liste déroulante d'options à partir desquelles l'utilisateur peut sélectionner une option unique.



The image shows a rectangular panel with a light gray background. Inside the panel, there are two widgets. The first widget is labeled 'Date' and contains a text input field with the value '2021-01-01 15:00:00'. The second widget is labeled 'Type service' and contains a dropdown menu with the selected option 'ffcs' and a small downward arrow on the right side.

Figure 5.6: Paramètres du tableau de bord de l'état du réseau

5.4.3.3 Création de l'application

Pour créer une interface de visualisation de carte avec la bibliothèque **Folium**, une classe **PanelFoliumMap** est définie comme une classe *Parameterized*; ce qui permet d'utiliser les *widgets* pour contrôler les paramètres d'affichage.

Lors de l'initialisation de la classe, un objet de carte **Folium** est créé à l'aide d'une fonction `get_map()`. Ensuite, un objet **FoliumPane** est créé pour afficher la carte avec des options de taille et de marges spécifiées.

Un objet **WidgetBox** est également créé pour contenir les *widgets* de contrôle pour la date et le type de service, permettant à l'utilisateur de définir les paramètres d'affichage.

La méthode **_update_map()** est définie pour mettre à jour la carte en fonction des paramètres définis par l'utilisateur. Elle dépend des paramètres type de service et date, ce qui signifie que si l'un de ces paramètres change, la méthode sera appelée automatiquement. Dans cette méthode, une nouvelle instance de la carte est créée à l'aide de la fonction **get_map()**. Un sous-ensemble de données est alors sélectionné à partir de la base de données de l'état du réseau en fonction des paramètres de date et de type de service définis par l'utilisateur. La fonction **plotPoint()** est ensuite utilisée pour tracer les points sur la carte en utilisant les données sélectionnées. Finalement, l'objet de la carte est assigné au **FoliumPane** pour afficher la carte mise à jour.

Le **FastListTemplate** utilisé aussi dans le tableau de bord des indicateurs est ensuite utilisé pour assembler les *widgets* et la carte **Folium** dans une seule interface utilisateur.

Enfin, une instance de la classe **PanelFoliumMap** est créée et la méthode **show()** est appelée sur la vue pour afficher l'interface utilisateur.

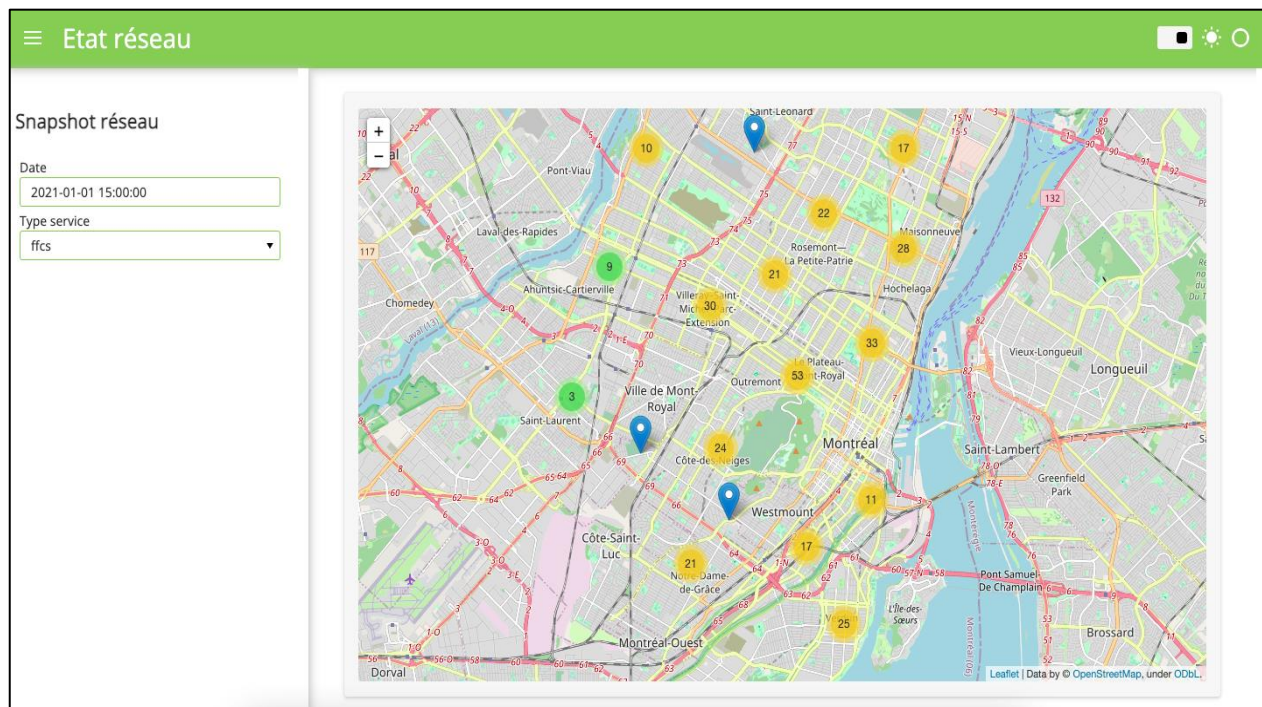


Figure 5.7: Interface de visualisation cartographique de l'état du réseau

Cette page de tableau de bord est dans le même style que la première montrant des indicateurs avec sur une barre latérale montrant les paramètres et la partie principale à droite montrant le graphique. Le fond de carte est fourni par **OpenStreetMap**. Ce dernier est un projet collaboratif mondial de cartographie en ligne. Il s'agit d'une base de données géographique libre et ouverte, créée par une communauté de contributeurs bénévoles du monde entier.

Le but principal **d'OpenStreetMap** est de fournir des cartes détaillées et précises de toutes les régions du monde, à la fois en termes de géographie physique (telles que les montagnes, les rivières et les côtes) et de géographie humaine (telles que les routes, les bâtiments, les lieux publics et les adresses).

L'un de ses avantages les plus importants est sa nature open source, qui signifie que toute personne peut accéder aux données cartographiques brutes et les utiliser pour développer des applications ou des services de cartographie personnalisés, sans avoir à payer de frais de licence ou à se conformer aux restrictions d'utilisation imposées par les propriétaires de données cartographiques propriétaires.

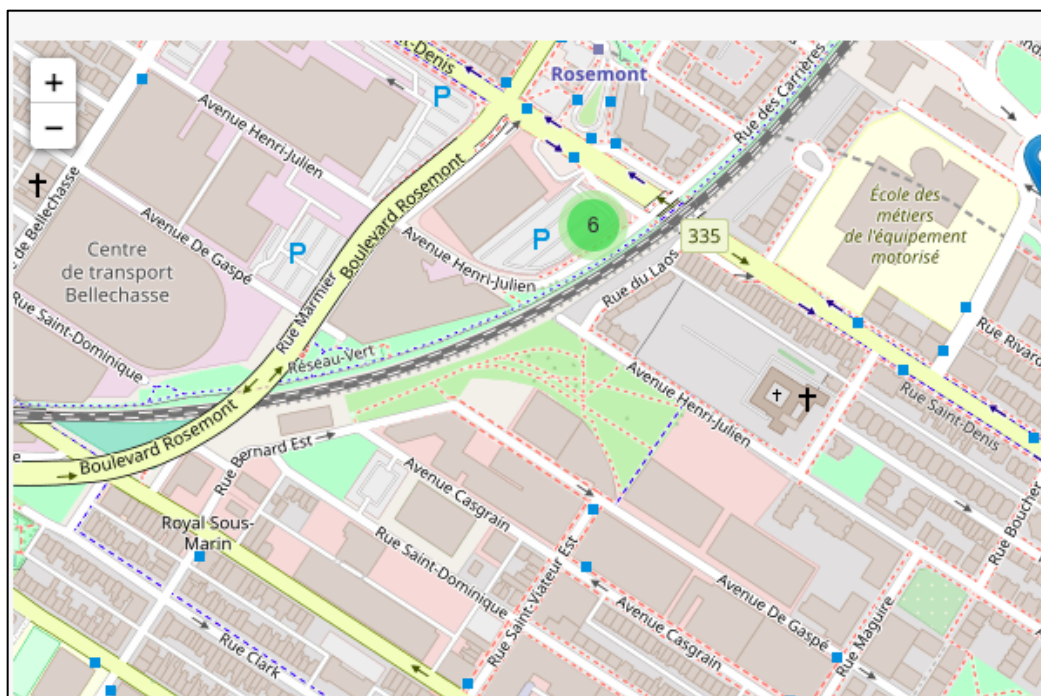


Figure 5.8 : Extrait de la carte de l'état du réseau

L'image présentée est une capture d'écran d'une carte qui affiche le nombre de voitures dans une zone donnée. Cette carte utilise un système qui regroupe les marqueurs de voitures en fonction de leur proximité les uns des autres. Plus précisément, selon le niveau de zoom choisi, on peut regrouper plus ou moins de voitures, ce qui permet d'obtenir une vision plus ou moins globale du nombre de voitures présentes dans la zone.

L'intérêt de cette carte est de fournir une visualisation claire et précise du nombre de voitures par secteur. En effet, les groupes de marqueurs permettent de se faire une idée rapide de la densité de voitures dans chaque secteur de la zone étudiée. En outre, la carte dessine un polygone délimité par les voitures présentes dans la zone, ce qui permet d'obtenir une vue globale et précise de l'étendue de la zone couverte par l'étude.

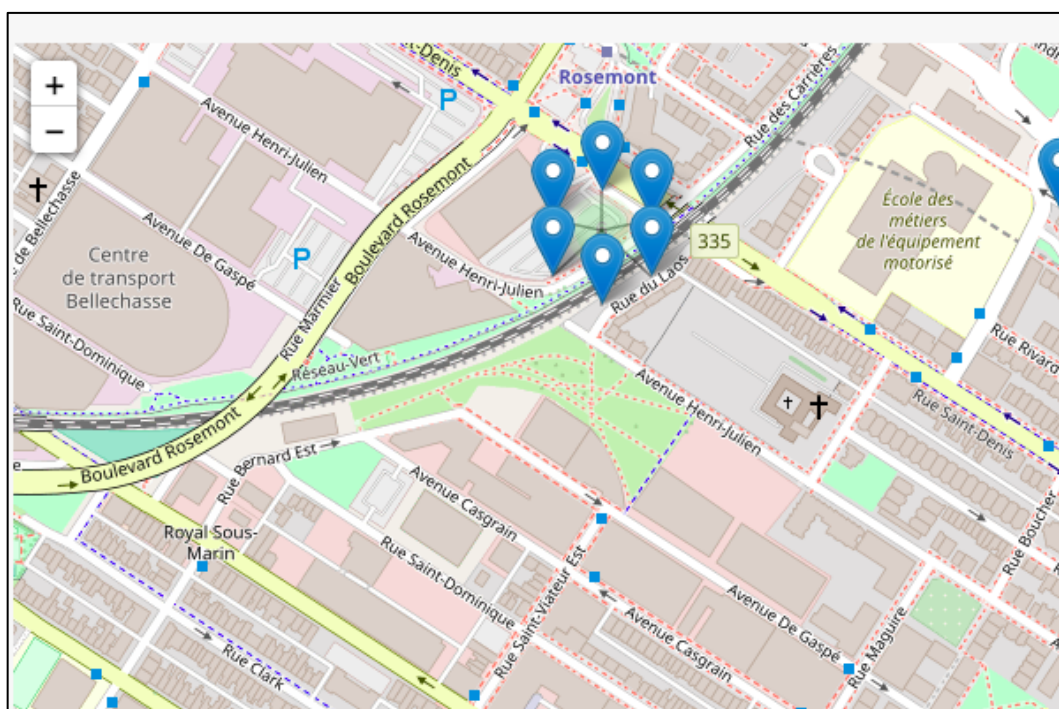


Figure 5.9 : Vue éclatée des véhicules disponibles dans l'extrait de carte

Cette image est une vue éclatée des positions exactes des véhicules présents dans une même station, obtenue en cliquant sur le regroupement présenté sur une image précédente. Cette visualisation permet de voir précisément les positions de tous les véhicules de la station, offrant ainsi une vue détaillée des emplacements de chacun des véhicules.

Cette vue éclatée est idéale pour ceux qui cherchent à obtenir plus de précision sur les emplacements des véhicules dans une station donnée. En effet, la visualisation permet de voir tous les véhicules en détail, avec une vue précise de chaque emplacement. Ce qui peut être particulièrement utile

5.5 Visualisation des circonstances

Dans cette section, nous expliquons comment a été montée l'interface de visualisation des circonstances des réservations sous forme de cartes.

5.5.1 Besoins de visualisation

Une carte représentant le nombre de véhicules disponibles par type dans un rayon donné est un outil crucial pour les gestionnaires du système d'autopartage. Cette carte permet de visualiser en temps réel la disponibilité des véhicules et leur emplacement au moment de réservations, ce qui est essentiel pour optimiser leur utilisation et améliorer l'expérience des utilisateurs.

Un avantage de cette carte est qu'elle permet de filtrer les véhicules disponibles en fonction de leur type. Ainsi, l'utilisateur peut choisir de ne voir que les véhicules de type électrique ou bien de ne voir que les voitures à essence.

En outre, cette carte offre également la possibilité de faire varier le rayon autour du point de réservation, afin de voir le nombre de véhicules disponibles dans une zone précise. Cela permet aux opérateurs du système de mieux avoir une visibilité sur les choix de véhicules selon la position, le nombre de véhicules autour.

En somme, la carte représentant le nombre de véhicules disponibles par type dans un rayon donné est un outil inestimable pour optimiser l'utilisation des véhicules d'autopartage et améliorer l'expérience des utilisateurs. En permettant aux utilisateurs de filtrer les données en fonction de leurs préférences et en offrant la possibilité de sélectionner un rayon spécifique, cette carte permet une utilisation plus efficace des ressources et une meilleure compréhension des besoins des utilisateurs.

5.5.2 Données requises en entrée et formalisation

Afin de produire cette carte représentant les circonstances des réservations à des dates choisies, nous avons besoin du fichier des circonstances ainsi que de différents paramètres. Ces paramètres servent simplement à filtrer les données de la base de données pour afficher les emplacements des véhicules sur la carte.

5.5.3 Réalisation de la plateforme sur Panel

5.5.3.1 Récupération et prétraitement des données

Dans le cadre de ce tableau de bord, nous procédons en plusieurs étapes. Tout d'abord, nous chargeons les données issues du fichier CSV relatif aux circonstances dans un *DataFrame* en utilisant la fonction `read_csv()`. Ensuite, nous appliquons un filtre sur le *DataFrame* pour ne conserver que les lignes présentant un "état" égal à "libre".

La phase suivante consiste à traiter le champ d'horodatage qui, étant sous forme de chaîne de caractères, doit être converti en objets "datetime". Pour ce faire, nous utilisons la fonction `pd.to_datetime()` et conservons les objets résultants dans la colonne d'origine. Les objets datetime sont des éléments clés pour représenter les dates et les heures lors des opérations d'analyse et de manipulation de données.

5.5.3.2 Ajout des widgets de paramétrage

La classe *PanelFoliumMap* définit les paramètres pour la carte web interactive en utilisant la bibliothèque 'param'.

Le paramètre "date" est un sélecteur de date qui permet à l'utilisateur de sélectionner une date spécifique pour prendre les réservations ayant eu lieu à cette date.

Le paramètre "type de service" est un paramètre d'entrée qui permet à l'utilisateur de spécifier le type de service qu'il souhaite filtrer. Il est important de spécifier que c'est le type de service de la réservation. Ainsi les rayons ne seront que pour des réservations du type de véhicule sélectionné à savoir le Flex ou le basé station.

Le paramètre "Rayon" est un paramètre d'entrée important qui permet à l'utilisateur de spécifier le rayon pour lequel il veut compter les véhicules disponibles.

Un paramètre sous forme de case à cocher permet de spécifier si on ne veut compter que les Flex autour de la réservation ou si on veut compter tous types de service confondus.

Circonstances de réservations

Date
2021-01-15 15:00:00

Type service
ffcs ▼

Rayon
250 ▼

☒ Flex autour

Figure 5.10: Paramètres du tableau de bord des circonstances

5.5.3.3 Création de l'application

Le code se compose d'un script Python qui crée une carte web interactive à l'aide de la bibliothèque **Folium**, qui est un *wrapper* Python pour la bibliothèque **JavaScript Leaflet**.

Une fonction **get_map()** crée une carte de base centrée sur Montréal en utilisant la classe **Map** de **Folium**. Les paramètres de latitude, longitude et niveau de zoom pour la carte sont spécifiés en tant que paramètres de fonction avec des valeurs par défaut.

La fonction principale du script est **plotPoint()**, qui ajoute des marqueurs et des cercles à la carte en fonction des données dans le *DataFrame* Pandas. La fonction prend quatre paramètres : l'objet de la carte, le fichier contenant les circonstances, le rayon du cercle à dessiner autour de chaque marqueur, et un indicateur booléen qui détermine s'il faut filtrer les données pour les véhicules électriques seulement.

La classe **MarkerCluster** est utilisée pour regrouper les marqueurs en clusters afin d'améliorer les performances de la carte. Chaque marqueur représente un emplacement où un véhicule est disponible pour la réservation, et il affiche un message contextuel montrant l'utilisateur et le type de véhicule.

Une classe **Circle** est utilisée pour dessiner des cercles autour de chaque marqueur afin de représenter la disponibilité des véhicules électriques. La couleur de chaque cercle indique le nombre de véhicules disponibles à cet endroit, avec une échelle de couleurs allant du rouge au vert utilisée pour représenter la disponibilité de faible à élevée. Une échelle logarithmique a été utilisée.

La classe **PanelFoliumMap** définit les paramètres pour la carte web interactive en utilisant la bibliothèque **param**. La classe inclut un sélecteur de date, un sélecteur pour le type de service (SBCS ou FFCS), un sélecteur pour le rayon des cercles dessinés autour de chaque marqueur, et un filtre pour les véhicules électriques seulement.

L'objet **view** définit la disposition de l'application web, avec les paramètres pour la carte interactive affichés dans une barre latérale et la carte affichée dans la zone principale.

La méthode **_update_map()** est une fonction décorée qui est appelée chaque fois que l'un des paramètres pour la carte interactive est modifié. Cette fonction met à jour la carte en fonction des nouvelles valeurs de paramètres en appelant la fonction **plotPoint()** et en passant les paramètres mis à jour.

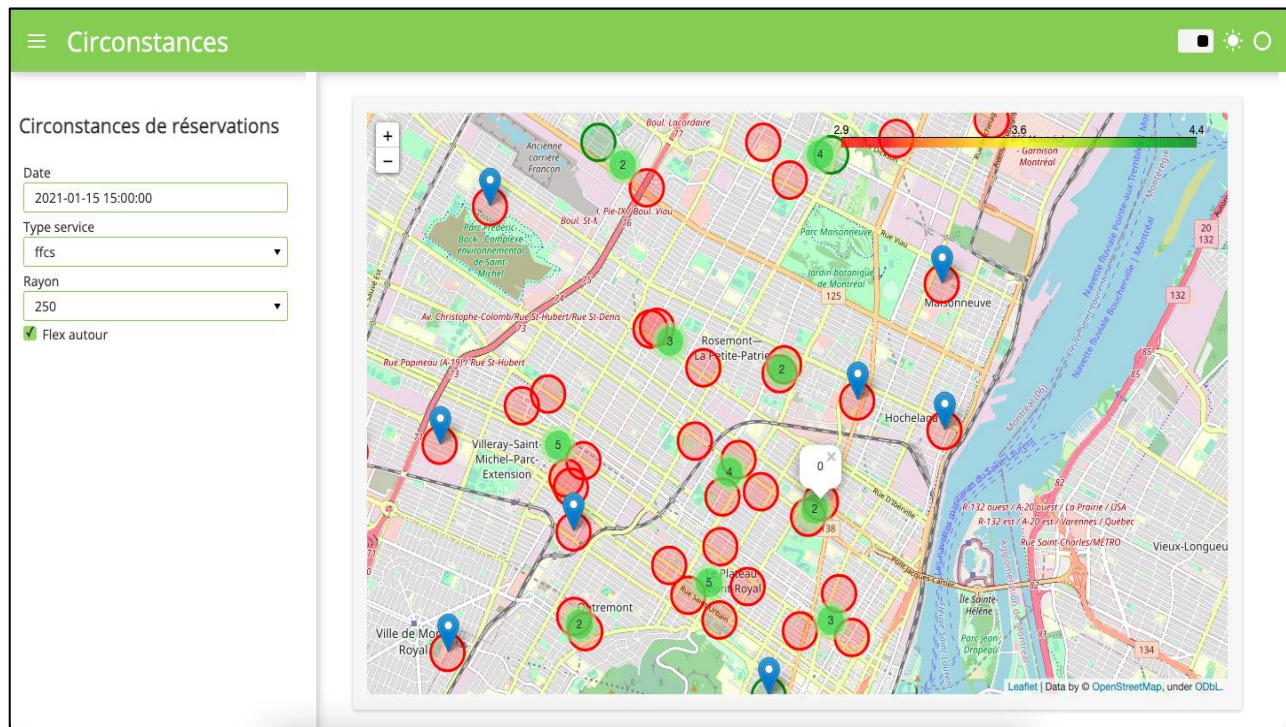


Figure 5.11 : Interface de visualisation cartographique des circonstances liées aux réservations

L'image présentée est une capture d'écran du tableau de bord qui présente les circonstances de réservation d'une flotte de véhicules. Cette image comprend une barre latérale permettant de choisir différents paramètres, ainsi qu'une partie principale composée d'une carte avec un fond *OpenStreetMap*.

Les marqueurs de localisation sur la carte représentent les lieux de réservation à l'heure choisie en paramètre. Le rayon coloré représente le rayon pour lequel on cherche les véhicules aux alentours, et la couleur est un gradient avec une échelle logarithmique qui représente le nombre de véhicules disponibles autour de chaque lieu de réservation.

Il est possible de compter les véhicules selon les paramètres choisis dans la barre latérale. Le système de regroupement des marqueurs de position selon le zoom est également présent sur cette carte, permettant ainsi de voir les zones les plus denses ou les plus clairsemées en fonction du niveau de zoom choisi.

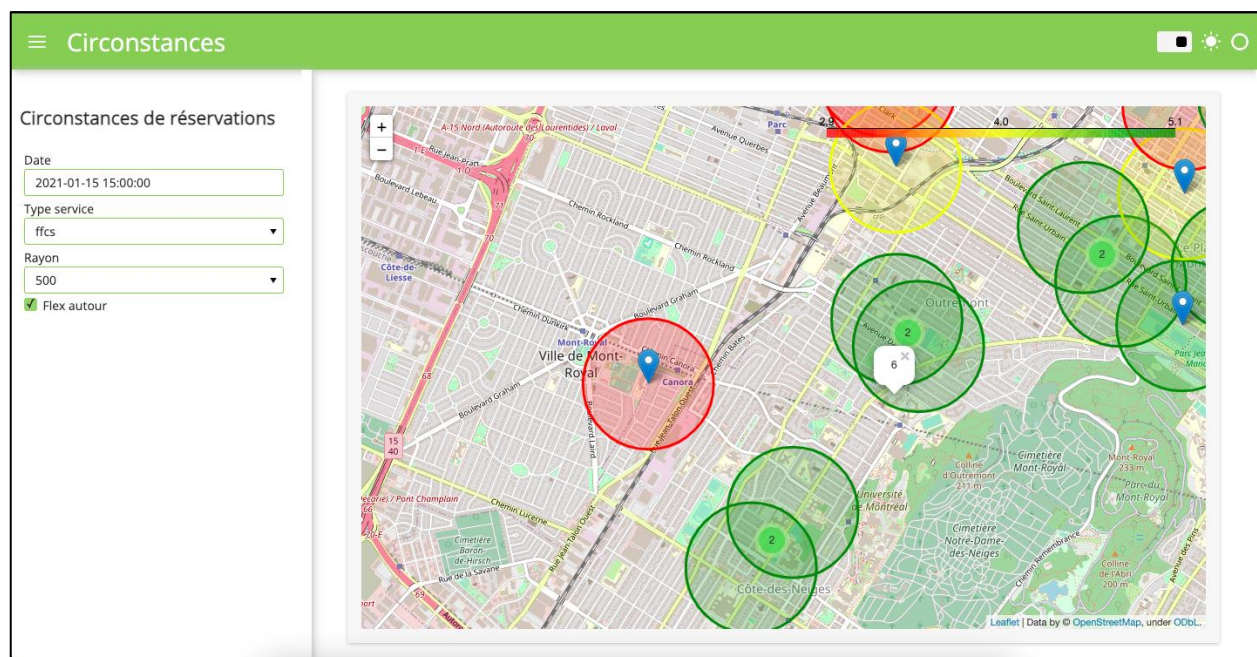


Figure 5.12 : Vue agrandie de la page des circonstances de réservation du tableau de bord

Cette image est une vision agrandie de la carte précédente qui a été décrite. La particularité de cette image est la présence des étiquettes sur les rayons colorés représentant les zones de recherche de véhicules aux alentours des lieux de réservation.

Ces étiquettes indiquent le nombre exact de véhicules dans le rayon correspondant, permettant ainsi une lecture plus précise et rapide de la carte.

5.6 Visualisation d'indicateurs sur les stations

Dans cette section, nous expliquons comment a été montée l'interface de visualisation des indicateurs cumulés sur les stations.

5.6.1 Besoins de visualisation

Le but visé est de créer une page de tableau de bord Web interactif qui visualise les données relatives aux indicateurs plus spécifiques aux stations.

Le tableau de bord a été conçu en gardant à l'esprit les exigences d'affichage suivantes :

Les utilisateurs doivent pouvoir sélectionner le type de graphique relatif à une station qu'ils souhaitent voir (profil cumulé, profil longitudinal ou nombre de stations vides).

- Le profil cumulé représente le nombre de véhicules libres et utilisés par classe de véhicule à une date précise sélectionnée par l'utilisateur. Les classes de véhicules sont définies par l'utilisateur via un menu déroulant. L'utilisateur peut également sélectionner les types de service (*flex* ou basé sur une station).
- Le profil longitudinal représente le nombre de véhicules libres et utilisés par état (libre ou utilisé) sur une plage horaire définie par l'utilisateur. L'utilisateur peut également sélectionner les types de service (*flex* ou basé sur une station) ainsi que les types de véhicules (essence, électrique ou hybride).
- Le nombre de stations vides représente le nombre de stations vides (sans véhicules) sur une plage horaire définie par l'utilisateur. L'utilisateur peut également sélectionner les types de service (*flex* ou basé sur une station) ainsi que les types de véhicules (essence, électrique ou hybride).

Les widgets doivent permettre à l'utilisateur de sélectionner les paramètres de chaque graphique. Ils incluent un menu déroulant pour sélectionner le type de graphique, un choix de plage horaire pour le profil longitudinal et le nombre de stations vides, un choix de date pour le profil cumulé,

une sélection de classe de véhicules, un choix de type de service, et une sélection de types de véhicules pour le profil longitudinal et le nombre de stations vides.

5.6.2 Données requises en entrée et formalisation

Pour créer le tableau de bord, nous devons d'abord rassembler les données d'entrée requises et les formaliser dans un format structuré pouvant être utilisé par la bibliothèque Panel. Les données utilisées dans le tableau de bord proviennent du fichier de l'état du réseau.

5.6.3 Réalisation de la plateforme sur Panel

5.6.3.1 Récupération et prétraitement des données

Pour préparer les données à utiliser dans le tableau de bord, nous avons utilisé la bibliothèque Panda pour lire le fichier CSV afin de le mettre dans un *DataFrame* et de convertir la colonne d'horodatage en objets *datetime*.

5.6.3.2 Ajout des widgets de paramétrage

Le tableau de bord utilise plusieurs widgets qui permettent à l'utilisateur d'interagir avec l'application. Voici une description de chacun de ces widgets :

- ***pn.widgets.Select*** : C'est un *widget* qui permet à l'utilisateur de sélectionner une option à partir d'une liste d'options définies. Dans le code, il est utilisé pour sélectionner le type de graphique à afficher. Il est utilisé pour sélectionner les classes (type de véhicule ou type de service) lorsque l'utilisateur choisit un profil cumulé.
- ***pn.widgets.DatetimeRangePicker*** : C'est un *widget* qui permet à l'utilisateur de sélectionner une plage de temps à partir d'un calendrier. Dans ce code, il est utilisé pour définir l'intervalle de temps sur lequel le graphique sera basé.
- ***pn.widgets.DatetimePicker*** : C'est un *widget* qui permet à l'utilisateur de sélectionner une date et une heure à partir d'un calendrier. Dans ce code, il est utilisé pour sélectionner une date spécifique pour le graphique "Profil cumulé".
- ***pn.widgets.MultiChoice*** : C'est un *widget* qui permet à l'utilisateur de sélectionner plusieurs options à partir d'une liste d'options définies. Dans ce code, il est utilisé pour sélectionner

les types de véhicules à inclure dans le graphique "Profil longitudinal" et "Nombre de stations vides".

- ***pn.widgets.CheckBoxGroup*** : C'est un *widget* qui permet à l'utilisateur de sélectionner plusieurs options à partir d'une liste d'options définies. Dans ce code, il est utilisé pour sélectionner les types de service à inclure dans le graphique.

Tous ces *widgets* sont mis en forme dans des boîtes de *widget* (*pn.WidgetBox*) pour faciliter la présentation.

5.6.3.3 Création de l'application

Les étapes suivantes correspondent à la création de l'interface utilisateur pour afficher les différents graphiques en fonction des choix de l'utilisateur.

La première partie du code définit des *widgets*, c'est-à-dire des éléments interactifs tels que des menus déroulants, des cases à cocher et des sélecteurs de date, qui permettent à l'utilisateur de personnaliser l'affichage des graphiques.

La deuxième partie du code utilise la fonction ***update_widget()*** pour mettre à jour les widgets en fonction du type de graphique choisi par l'utilisateur. La fonction ***update_widget()*** crée un nouveau groupe de *widgets* en fonction du choix de l'utilisateur, et met à jour la variable ***row*** qui contient ces widgets.

La troisième partie du code utilise la fonction ***get_chart()*** pour afficher le graphique choisi par l'utilisateur en fonction des paramètres définis dans les *widgets*. Cette fonction utilise la fonction ***plot_choice()*** pour générer le graphique, en fonction des paramètres sélectionnés par l'utilisateur. Le graphique est ensuite renvoyé sous forme d'objet *Panel*, ce qui permet de l'afficher dans l'interface utilisateur.

Enfin, la dernière partie du code utilise la classe ***FastListTemplate*** de *Panel* pour créer une interface utilisateur complète avec une barre latérale et une zone d'affichage des graphiques. L'interface utilisateur est affichée avec la méthode ***show()*** de la classe ***FastListTemplate***.

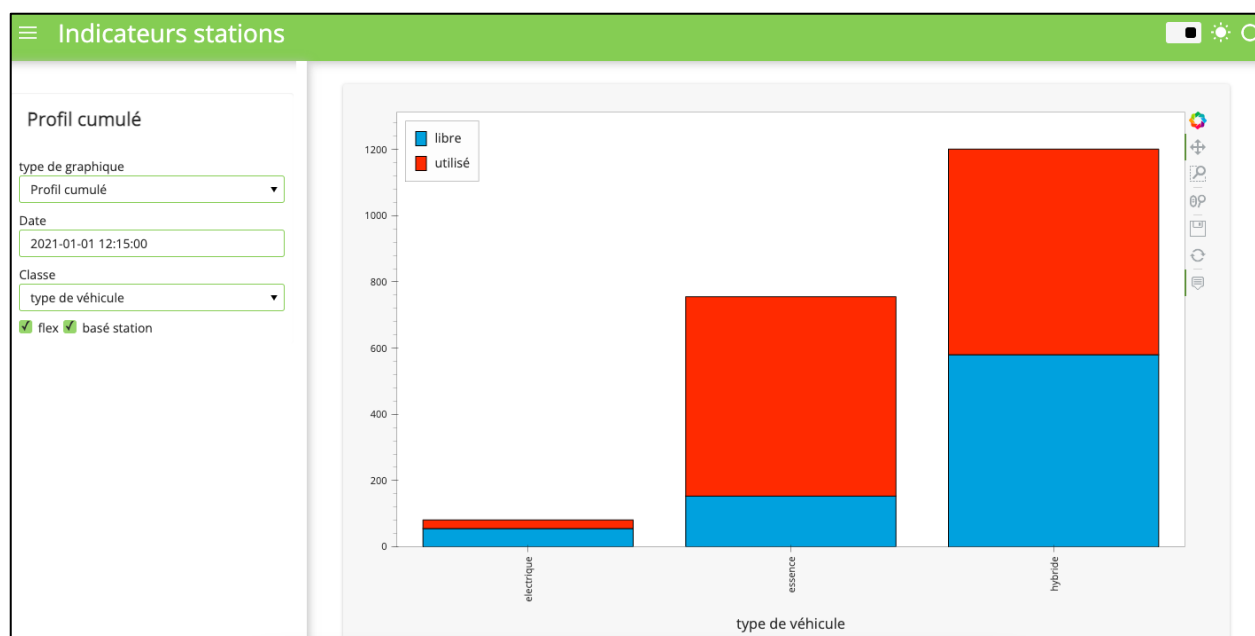


Figure 5.13 : Page des profils cumulés du tableau de bord

La figure 5.13 représente une page du tableau de bord affichant des indicateurs sur les stations. Cette page en particulier est dotée d'une barre latérale qui permet de régler différents paramètres. Parmi ces paramètres, on retrouve le champ « Classe », qui permet de sélectionner avec quelles classes faire les profils cumulés, tels que le type de véhicule ou le type de service.

Sur la page, on peut observer des barres dont les parties rouges représentent la proportion de véhicules utilisés pour chaque classe. En d'autres termes, ces barres rouges reflètent la répartition des différents types de véhicules ou de services utilisés dans les stations. La partie bleue de ces mêmes barres, quant à elle, représente la proportion de véhicules non utilisés dans chaque classe.

En somme, cette image offre une représentation graphique claire et précise des données relatives aux différentes classes de véhicules ou de services utilisés dans les stations. Grâce aux paramètres réglables via la barre latérale, il est possible de personnaliser cette représentation en fonction de ses besoins spécifiques en termes de données.

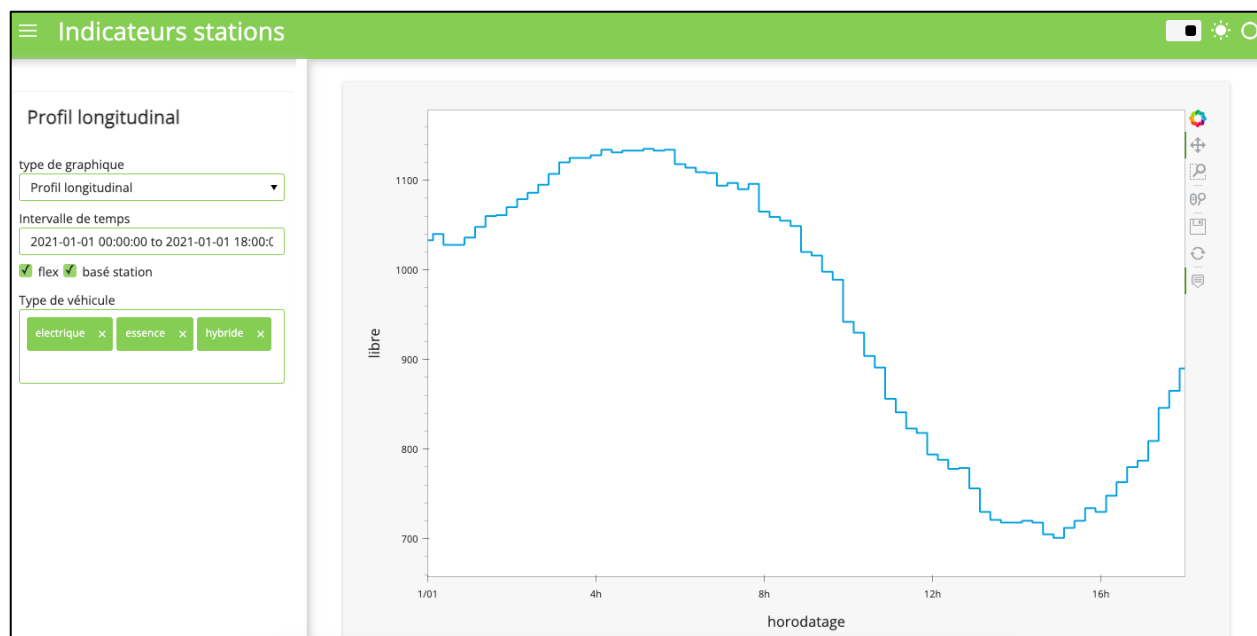


Figure 5.14 : Page du profil longitudinal du nombre de véhicules libres du tableau de bord

La Figure 5.14 présentée est une page du tableau de bord qui affiche quelques indicateurs sur les stations. Elle représente un profil longitudinal du nombre de véhicules disponibles en tout temps. La barre latérale à gauche permet de choisir les filtres et les paramètres pour afficher les informations les plus pertinentes. Par exemple, les utilisateurs peuvent filtrer les données en fonction de la plage de temps souhaitée, sélectionner un type de service spécifique ou filtrer les véhicules en fonction de leur type. Cette image peut s'avérer intéressante pour l'analyse des données, car elle permet de visualiser clairement les variations du nombre de véhicules disponibles en fonction du temps et des différents filtres sélectionnés.

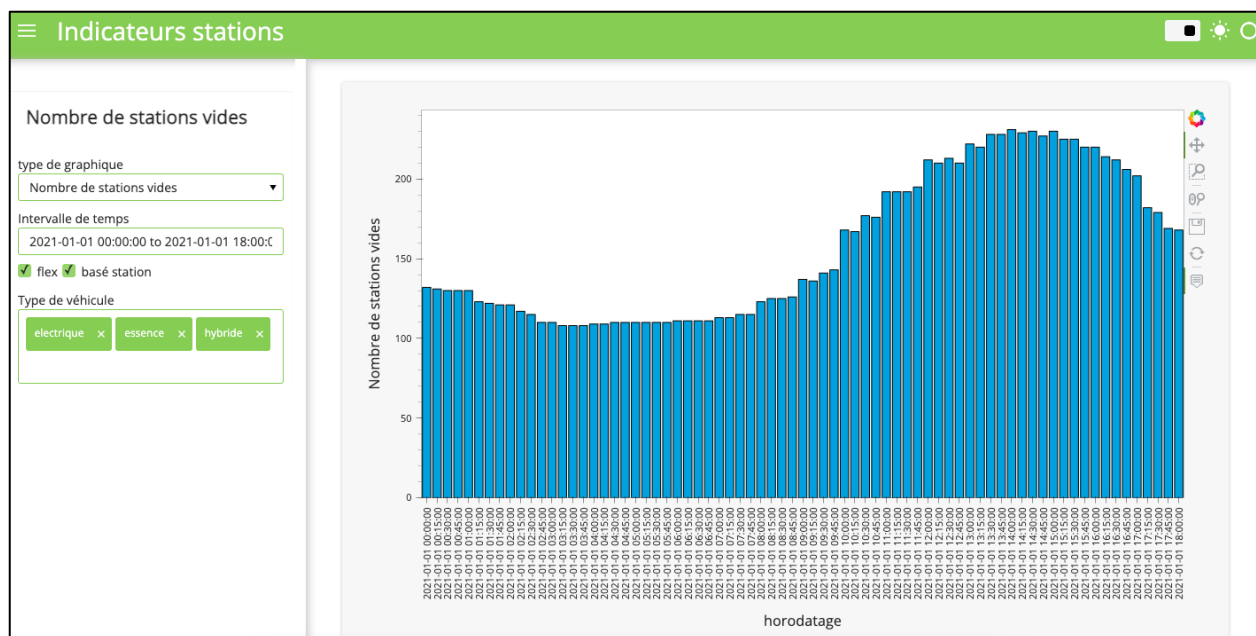


Figure 5.15 : Page du nombre de stations vides du tableau de bord

La Figure 5.15 présentée est une page du tableau de bord affichant quelques indicateurs sur les stations. Elle représente le nombre de stations vides en tout temps. L'image comporte une barre latérale à gauche qui permet de choisir différents paramètres et filtres tels qu'une plage de temps, le type de service avec une boîte à cocher, ou le type de véhicule. Cette image peut s'avérer intéressante pour l'analyse des données, car elle permet de visualiser clairement les variations du nombre de stations vides en fonction du temps et des différents filtres sélectionnés.

CHAPITRE 6 CONCLUSION ET RECOMMANDATIONS

6.1 Contributions

Ce mémoire a proposé des outils permettant de mettre en évidence l'importance croissante de l'autopartage en tant que solution viable et durable dans le domaine de la mobilité urbaine. Il souligne la nécessité d'avoir des indicateurs de performance pertinents et des outils de visualisation adaptés pour optimiser la gestion des opérations d'autopartage et anticiper les besoins futurs. La recherche approfondie effectuée a permis de dresser un tableau exhaustif des indicateurs de performance, de fournir des outils visant à identifier les principales tendances et les enjeux liés à l'autopartage, ainsi que de proposer de bonnes pratiques pour mesurer et évaluer la performance des entreprises d'autopartage.

Dans ce cadre, ce mémoire a proposé la construction de deux fichiers de valorisation des réservations de véhicules Communauto, offrant une représentation détaillée des circonstances entourant les réservations. Ces fichiers, combinés aux tableaux de bord développés, facilitent la compréhension et l'utilisation des indicateurs de performance identifiés, permettant aux entreprises d'autopartage de mieux gérer leur flotte de véhicules et d'améliorer leurs services.

6.2 Limitations

Il est essentiel de reconnaître des limitations et de les prendre en compte lors de l'utilisation des fichiers de valorisation des réservations et des outils développés. Cela permettra d'interpréter les résultats de manière appropriée et d'identifier les domaines où des améliorations peuvent être apportées pour une analyse plus précise et une utilisation optimale des données. Ces limitations comprennent la présence éventuelle de données erronées et des problèmes liés aux événements de réservation annulés. De plus, la résolution temporelle des fichiers, fixée à des intervalles de 15 minutes, peut entraîner une perte d'informations détaillées sur les réservations. Par ailleurs, le manque de modularité de l'interface et la programmation rigide peuvent limiter la flexibilité et l'adaptabilité. Cela peut rendre difficile l'ajout de nouvelles fonctionnalités ou l'adaptation de l'outil aux besoins futurs.

6.3 Perspectives

Les résultats obtenus dans cette étude, notamment les fichiers produits et les tableaux de bord développés, peuvent servir de base pour de futures recherches et applications dans le domaine de l'autopartage. Pour les perspectives, plusieurs actions peuvent être envisagées. Premièrement, il serait pertinent d'intégrer les fichiers et les outils développés chez Communauto, afin d'améliorer la gestion opérationnelle de l'entreprise. Deuxièmement, il serait intéressant de développer des modèles avancés pour mieux expliquer les réservations, en utilisant des techniques d'apprentissage automatique. Cela permettrait d'anticiper les besoins futurs et de prendre des décisions éclairées en matière de planification et de gestion.

Par ailleurs, les données recueillies dans les fichiers pourraient être exploitées pour étudier le comportement des utilisateurs d'autopartage. Cela permettrait d'analyser les habitudes de réservation, les motifs de déplacement, les préférences de véhicule, etc., afin de mieux comprendre les clients et d'adapter les services en conséquence.

RÉFÉRENCES

- Alfian, G., Rhee, J., Ijaz, M. F., Syafrudin, M., & Fitriyani, N. L. (2017). *Performance Analysis of a Forecasting Relocation Model for One-Way Carsharing*. 23.
- Alfian, G., Rhee, J., Kang, Y.-S., & Yoon, B. (2015). Performance Comparison of Reservation Based and Instant Access One-Way Car Sharing Service through Discrete Event Simulation. *Sustainability*, 7(9), 12465-12489. <https://doi.org/10.3390/su70912465>
- Barth, M., & Todd, M. (1999). Simulation model performance analysis of a multiple station shared vehicle system. *Transportation Research Part C: Emerging Technologies*, 7(4), 237-259. [https://doi.org/10.1016/S0968-090X\(99\)00021-2](https://doi.org/10.1016/S0968-090X(99)00021-2)
- Baumgarte, F. (2021). *You'll never share alone : Analyzing carsharing user group behavior*.
- Baumgarte, F. (2022). *Revealing influences on carsharing users' trip distance in small urban areas*.
- Correia, G. H. de A., & Antunes, A. P. (2012). Optimization approach to depot location and trip selection in one-way carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 48(1), 233-247. <https://doi.org/10.1016/j.tre.2011.06.003>
- Costain, C., Ardron, C., & Habib, K. N. (2012). Synopsis of users' behaviour of a carsharing program: A case study in Toronto. *Transportation Research Part A: Policy and Practice*, 46(3), 421-434. doi:<https://doi.org/10.1016/j.tra.2011.11.005>
- de Luca, S., & Pace, R. D. (2015). *Modelling users' behaviour in inter-urban carsharing program : A stated preference approach*.
- Ding, C. (2017). *Influences of built environment characteristics and individual factors on commuting distance : A multilevel mixture hazard modeling approach*.
- Eckhardt, G. M., Houston, M. B., Jiang, B., Lamberton, C., Rindfleisch, A., & Zervas, G. (2019). Marketing in the Sharing Economy. *Journal of Marketing*, 83(5), 5–27. <https://doi.org/10.1177/0022242919861929>
- Ferrero, F., Perboli, G., Rosano, M., & Vesco, A. (2018). Car-sharing services : An annotated review. *Sustainable Cities and Society*, 37, 501-518. <https://doi.org/10.1016/J.SCS.2017.09.020>

- Golalikhani, M., Oliveira, B., Carravilla, M. A., Oliveira, J., & Antunes, A. (2021). Carsharing : A review of academic literature and business practices toward an integrated decision-support framework. *Transportation Research Part E-logistics and Transportation Review*, 149, 102280. <https://doi.org/10.1016/J.TRE.2021.102280>
- Habib, K. M. N., Morency, C., Islam, M. T., & Grasset, V. (2012). *Modelling users' behaviour of a carsharing program : Application of a joint hazard and zero inflated dynamic ordered probability model*.
- Huang, K., An, K., Rich, J., & Ma, W. (2020). Vehicle relocation in one-way station-based electric carsharing systems : A comparative study of operator-based and user-based methods. *Transportation Research Part E-logistics and Transportation Review*, 142, 102081. <https://doi.org/10.1016/J.TRE.2020.102081>
- Kek, A. G. H., Cheu, R. L., Meng, Q., & Fung, C. H. (2009). A decision support system for vehicle relocation operations in carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 45(1), 149-158. <https://doi.org/10.1016/j.tre.2008.02.008>
- Kopp, J., Gerike, R., & Axhausen, K. W. (2015). *Do sharing people behave differently ? An empirical evaluation of the distinctive mobility patterns of free-floating car-sharing members*.
- Liao, F. (s. d.). *Carsharing : The impact of system characteristics on its potential to replace private car trips and reduce car ownership*.
- Lu, R., Correia, G., Zhao, X., Liang, X., & Lv, Y. (2020). Performance of one-way carsharing systems under combined strategy of pricing and relocations. *Transportmetrica B: Transport Dynamics*, 9, 134-152. <https://doi.org/10.1080/21680566.2020.1819912>
- Martin, B. (2007) *Caractérisation du système d'autopartage dans l'agglomération montréalaise et analyse spatio-temporelle de ses différents objets : usagers, stationnements, véhicules*
- M Barth, Han, J., & Todd, M. (s. d.-a). *Performance evaluation of a multi-station shared vehicle system*. 6.
- M Barth, Han, J., & Todd, M. (s. d.-b). *Performance evaluation of a multi-station shared vehicle system*. 6.

- Mishra, G. S. (2015). The effect of carsharing on vehicle holdings and travel behavior: A propensity score and causal mediation analysis of the San Francisco Bay Area. *Research in Transportation Economics*.
- Morency, C., Trepanier, M., Agard, B., Martin, B., & Quashie, J. (s. d.). *Car sharing system : What transaction datasets reveal on users' behaviors*.
- Münzel, K., Boon, W., Frenken, K., & Blomme, J. (s. d.). *Explaining carsharing supply across Western European cities*.
- Nansubuga, B., & Kowalkowski, C. (2021). Carsharing : A Systematic Literature Review and Research Agenda. *Urban Economics & Regional Studies eJournal*. <https://doi.org/10.1108/josm-10-2020-0344>
- Neumann, T. (2021). *The Impact of Carsharing on Transport in the City. Case Study of Tri-City in Poland*.
- Niels, T., & Bogenberger, K. (2017). Booking Behavior of Free-Floating Carsharing Users: Empirical Analysis of Mobile Phone App and Booking Data Focusing on Battery Electric Vehicles. *Transportation Research Record*, 2650(1), 123–132. <https://doi.org/10.3141/2650-15>
- Nourinejad, M., & Roorda, M. J. (2014). A dynamic carsharing decision support system. *Transportation Research Part E: Logistics and Transportation Review*, 66, 36-50. <https://doi.org/10.1016/j.tre.2014.03.003>
- Nourinejad, M., Zhu, S., Bahrami, S., & Roorda, M. J. (2015). Vehicle relocation and staff rebalancing in one-way carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 81, 98-113. <https://doi.org/10.1016/j.tre.2015.06.012>
- Orski, C. K. (2001). Car sharing. *Transportation Quarterly*, 55(4).
- Park, S., & Yu, W. (2021). *Analysis of system parameters for one-way carsharing systems*. <https://doi.org/10.1080/19427867.2021.1906476>
- Prinz, C., Lichtenberg, S., & Willnat, M. (2021). *CASSI: Design of a Simulation Environment for Vehicle Relocation in Carsharing*. <https://doi.org/10.17705/1THCI.00140>
- Repoux, M., Kaspi, M., Boyaci, B., & Geroliminis, N. (2019). Dynamic prediction-based relocation policies in one-way station-based carsharing systems with complete journey

reservations. *Transportation Research Part B-methodological*, 130, 82-104. <https://doi.org/10.1016/j.trb.2019.10.004>

Schiffer, M., Hiermann, G., Rüdel, F., & Walther, G. (2021). A polynomial-time algorithm for user-based relocation in free-floating car sharing systems. *Transportation Research Part B-methodological*, 143, 65-85. <https://doi.org/10.1016/j.trb.2020.11.001>

Shaheen, Susan & Mallery, Mark & Kingsley, Karla. (2012). Personal Vehicle Sharing Services in North America. *Research in Transportation Business & Management*. 3. 71–81. [10.1016/j.rtbm.2012.04.005](https://doi.org/10.1016/j.rtbm.2012.04.005).

Shaheen, Susan & Sperling, Daniel & Wagner, Conrad. (1999). A Short History of Carsharing in the 90's. *J World Transp Policy Pract.* 5.

Steininger, K., Vogl, C., & Zettl, R. (s. d.). *The size of the market segment and revealed change in mobility behavior*.

Tuominen, A., Rehunen, A., Peltomaa, J., & Mäkinen, K. (2019). *Facilitating practices for sustainable car sharing policies—An integrated approach utilizing user data, urban form variables and mobility patterns*.

Wang, L., Ma, W., Wang, M., & Qu, X. (2021). Demand control model with combinatorial incentives and surcharges for one-way carsharing operation. *Transportation Research Part C-emerging Technologies*, 125, 102999. <https://doi.org/10.1016/J.TRC.2021.102999>

Wielinski, G., Trépanier, M., & Morency, C. (s. d.). What about Free-Floating Carsharing? *Transportation Research Record*.

ANNEXE A CODE PYTHON DE LA CRÉATION DES FICHIERS

CRÉATION DU FICHIER DE L'ÉTAT DU RÉSEAU

Ci-après l'algorithme de création du fichier de l'état du réseau

```
header = ['datetime', 'res_id', 'v_id', 'state', 'st_id', 'longitude_start', 'latitude_start', 'res_type', 'u_type', 'v_type_modele', 'res_type_service']

start_date = datetime.datetime(2021, 1, 1, 0, 0)
end_date = datetime.datetime(2021, 1, 31, 23, 59, 0)
delta = datetime.timedelta(minutes=15)

with open('profil_accumulation24.csv', 'w', encoding='UTF8') as f:
    writer = csv.writer(f)

    # write the header
    writer.writerow(header)

    while start_date <= end_date:
        print(start_date)
        for vehicle in stations_autos:
            df1 = jan21_f[jan21_f['v_id'] == vehicle]
            df = df1[(df1['res_trip_start_datetime'] <= start_date) & (df1['res_trip_end_datetime'] >= start_date)]
            used = df.v_id.unique().tolist()
            if df.empty:
                if df1.iloc[0]['res_type_service'] == 'sbcs':
                    writer.writerow([start_date, None, df1.iloc[0]['v_id'], 'Free', df1.iloc[0]['st_id'], df1.iloc[0]['Longitude_1'], df1.iloc[0]['Latitude_1'], None, None, df1.iloc[0]['v_type_modele'], df1.iloc[0]['res_type_service']])
                else:
                    writer.writerow([start_date, None, df1.iloc[0]['v_id'], 'Free', df1.iloc[0]['st_id'], None, None, None, None, df1.iloc[0]['v_type_modele'], df1.iloc[0]['res_type_service']])
            else:
                writer.writerow([start_date, df.iloc[0]['res_id'], vehicle, 'Used', df.iloc[0]['st_id'], df.iloc[0]['Longitude_1'], df.iloc[0]['Latitude_1'], df.iloc[0]['res_type'], df.iloc[0]['u_type'], df.iloc[0]['v_type_modele'], df.iloc[0]['res_type_service']])
            start_date += delta
```

ENRICHISSEMENT DU FICHIER DE L'ÉTAT DU RÉSEAU

```
tripevents = pd.read_csv('/Users/amadoubagnicksarr/Downloads/tripevents_2020-11-01_2022-01-01.csv')
```

```
tripevents['TripEventDate'] = pd.to_datetime(tripevents['TripEventDate'])
```

Un aperçu de tripevents:

```
tripevents
```

AJOUT DERNIERS TRIP EVENTS

On charge le fichier de l'état du réseau dans un dataframe afin de l'enrichir avec les données de tripevents:

```
prof_final = pd.read_csv('profil_accumulation24.csv')
```

```
prof_final['datetime'] = pd.to_datetime(prof_final['datetime'])
```

L'algorithme d'enrichissement avec le fichier tripevents

```
import datetime
import csv
```

```
header = ['datetime', 'res_id', 'v_id', 'state', 'st_id', 'longitude_start', 'latitude_start', 'res_type', 'u_type', 'v_type_modele', 'res_type_service', 'last_longitude', 'last_latitude', 'last_TripEventDate', 'last_event']
```

```
with open('profil_janvier21_vf.csv', 'w', encoding='UTF8') as f:
    writer = csv.writer(f)
```

```
    # write the header
    writer.writerow(header)
    for index, row in prof_final.iterrows():
        if row['state'] == 'Free':
            last_pos = jan21_f[(jan21_f['v_id'] == row['v_id']) & (jan21_f['res_trip_end_datetime'] < row['datetime'])]
            last_pos.sort_values(by=['res_trip_end_datetime'], inplace=True, ascending=False)
            if row['res_type_service'] == 'ffcs':
                if last_pos.empty:
                    writer.writerow([row['datetime'], None, row['v_id'], 'Free', row['st_id'], row['longitude_start'], row['latitude_start'], None, None, row['v_type_modele'], row['res_type_service'], None, None, None, 'Unknown'])
                else:
                    writer.writerow([row['datetime'], None, row['v_id'], 'Free', row['st_id'], row['longitude_start'], row['latitude_start'], None, None, row['v_type_modele'], row['res_type_service'], last_pos.iloc[0]['Longitude_2'], last_pos.iloc[0]['Latitude_2'], last_pos.iloc[0]['res_trip_end_datetime'], 'Parked'])
```

```

        else:
            if last_pos.empty:
                writer.writerow([row['datetime'], None , row['v_id'],'Free', row['st_id'], row['longitude_start'],row['latitude_start'], None , None , row['v_type_modele'], row['res_type_service'] ,row['longitude_start'],row['latitude_start'], 'Unknown','Station'])
            else:
                writer.writerow([row['datetime'], None , row['v_id'],'Free', row['st_id'], row['longitude_start'],row['latitude_start'], None , None , row['v_type_modele'], row['res_type_service'] ,row['longitude_start'],row['latitude_start'], last_pos.iloc[0]['res_trip_end_datetime'],'Station'])

    else:
        last_row = tripevents[(tripevents['CarID'] == row['v_id']) & (tripevents['TripEventDate'] < row['datetime'])].iloc[-1:]
        if last_row.empty:
            writer.writerow([row['datetime'], row['res_id'] , row['v_id'], row['state'], row['st_id'], row['longitude_start'] , row['latitude_start'] , row['res_type'] , row['u_type'] , row['v_type_modele'],row['res_type_service'], None, None, None , 'Unknown'])
        else:
            writer.writerow([row['datetime'], row['res_id'] , row['v_id'], row['state'], row['st_id'], row['longitude_start'] , row['latitude_start'] , row['res_type'] , row['u_type'] , row['v_type_modele'],row['res_type_service'], last_row.iloc[0]['Longitude'],last_row.iloc[0]['Latitude'], last_row.iloc[0]['TripEventDate'],'trip event'])

```

DISPONIBILITÉ PAR RAYON

La fonction haversine définie ici est utilisée dans le code qui crée le fichier des circonstances :

```
import math

def haversine(coord1,row):
    lon1,lat1=coord1
    lon2=row['last_longitude']
    lat2=row['last_latitude']

    R=6371000
    phi_1=math.radians(lat1)
    phi_2=math.radians(lat2)

    delta_phi=math.radians(lat2-lat1)
    delta_lambda=math.radians(lon2-lon1)

    a=math.sin(delta_phi/2.0)**2+\
        math.cos(phi_1)*math.cos(phi_2)*\
        math.sin(delta_lambda/2.0)**2
    c=2*math.atan2(math.sqrt(a),math.sqrt(1-a))

    distance=R*c

    return distance
```

CRÉATION DU FICHIER DES CIRCONSTANCES

```
header = ['res_id', 'datetime', 'res_type', 'u_type', 'v_id', 'v_type_modele', 'res_type_service', 'longitude_res', 'latitude_res', 'nb_veh_avail', 'nb_ffcs_avail_250', 'nb_sbcscs_avail_250', 'nb_ffcs_avail_500', 'nb_sbcscs_avail_500', 'nb_ffcs_avail_750', 'nb_sbcscs_avail_750', 'nb_EV_250', 'nb_EV_500', 'nb_EV_750']
```

```
start_date = datetime.datetime(2021, 1, 1, 0, 0)
end_date = datetime.datetime(2021, 1, 31, 23, 59, 0)
delta = datetime.timedelta(minutes=15)
list_res = list()
```

```
with open('rayon_reserv_jan21_stVF.csv', 'w', encoding='UTF8') as f:
    writer = csv.writer(f)

    writer.writerow(header)

    while start_date <= end_date:
        print(start_date)
        df_date = profil_jan[profil_jan['datetime'] == start_date]
        for index, row in df_date.iterrows():
            if (row['res_id'] in list_res) or (row['state'] == 'Free'):
                continue
            else:
                df_date['distance'] = df_date.apply(lambda coord: haversine([
row['longitude_start'], row['latitude_start']], coord ), axis=1)

                df_veh_avail = df_date[df_date['state'] == 'Free']

                nb_veh_avail = len(df_date[df_date['state'] == 'Free'].v_id.unique())

                nb_ffcs_avail_250 = len(df_veh_avail[(df_veh_avail['distance']
] <= 250) & (df_veh_avail['res_type_service'] == 'ffcs')].v_id.unique())
                nb_sbcscs_avail_250 = len(df_veh_avail[(df_veh_avail['distance']
] <= 250) & (df_veh_avail['res_type_service'] == 'sbcscs')].v_id.unique())
                nb_EV_veh_250 = len (df_veh_avail[(df_veh_avail['distance'] <
= 250) & (df_veh_avail['v_type_modele'] == 'EV')].v_id.unique())

                nb_ffcs_avail_500 = len(df_veh_avail[(df_veh_avail['distance']
] <= 500) & (df_veh_avail['res_type_service'] == 'ffcs')].v_id.unique())
                nb_sbcscs_avail_500 = len(df_veh_avail[(df_veh_avail['distance']
] <= 500) & (df_veh_avail['res_type_service'] == 'sbcscs')].v_id.unique())
                nb_EV_veh_500 = len (df_veh_avail[(df_veh_avail['distance'] <
= 500) & (df_veh_avail['v_type_modele'] == 'EV')].v_id.unique())

                nb_ffcs_avail_750 = len(df_veh_avail[(df_veh_avail['distance']
] <= 750) & (df_veh_avail['res_type_service'] == 'ffcs')].v_id.unique())
                nb_sbcscs_avail_750 = len(df_veh_avail[(df_veh_avail['distance']
```

```

] <= 750) & (df_veh_avail['res_type_service'] == 'sbcs')).v_id.unique())
    nb_EV_veh_750 = len (df_veh_avail[(df_veh_avail['distance'] <
= 750) & (df_veh_avail['v_type_modele'] == 'EV')].v_id.unique())

    writer.writerow([row['res_id'], row['datetime'], row['res_type'],
row['u_type'], row['v_id'],row['v_type_modele'],row['res_type_service'],
row['longitude_start'], row['latitude_start'], nb_veh_avail, nb_ffcs_avail_250,
nb_sbcs_avail_250, nb_ffcs_avail_500,nb_sbcs_avail_500, nb_ffcs_avail_750,nb_sbcs_avail_750,
nb_EV_veh_250, nb_EV_veh_500, nb_EV_veh_750])
    list_res.append(row['res_id'])
    start_date += delta

```

ANNEXE B APERÇU DU FICHIER DE L'ÉTAT DU RÉSEAU

horodatage	id de la reser	id du véhicule	etat	id de la station	longitude_start	latitude_start	type de reserv	type d'utilisateur	type de véhicule	type de service	dernière longitude	dernière latitude	date du dernier événement	dernier événement
2021-01-01	28322897.0	6883	utilisé	175	-73.590658	45.524059	47.0	1.0	essence	basé station	-73.5908231573539	45.5240755494442	2020-12-31 16:06:01	trip event
2021-01-01	28375417.0	6105	utilisé	467	-73.5762731796288	45.4814438498416	47.0	1.0	essence	basé station	-73.5762731796288	45.4814438498416	2020-12-27 16:15:47	trip event
2021-01-01	28168114.0	4480	utilisé	156	-73.5665	45.5224	47.0	1.0	hybride	basé station	-73.5663815282975	45.5224088918579	2020-12-31 15:36:16	trip event
2021-01-01	28311076.0	5944	utilisé	175	-73.590658	45.524059	47.0	1.0	essence	basé station	-73.5907998556636	45.5241421855728	2020-12-31 17:06:01	trip event
2021-01-01	28425570.0	6727	utilisé	501	-73.5862598817397	45.5372305270582	47.0	1.0	hybride	basé station	-73.5862598817397	45.5372305270582	2020-12-20 14:34:49	trip event
2021-01-01	28299348.0	5938	utilisé	244	-73.58154	45.53978	47.0	1.0	essence	basé station	-73.5817031440099	45.5400321781255	2020-12-31 19:23:33	trip event
2021-01-01	28322900.0	6610	utilisé	411	-73.6273315415619	45.5436005218573	47.0	10.0	essence	basé station	-73.6271498219055	45.5434605440777	2020-12-31 17:06:52	trip event
2021-01-01	28301255.0	4573	utilisé	436	-73.540899	45.553426	47.0	1.0	hybride	basé station	-73.540959803327	45.5534655187371	2020-12-31 12:14:37	trip event
2021-01-01	27796116.0	6612	utilisé	436	-73.540899	45.553426	47.0	1.0	essence	basé station	-73.5410381741198	45.5534955259497	2020-12-31 20:00:01	trip event
2021-01-01	28118647.0	6145	utilisé	1001	-73.610986	45.543303	47.0	1.0	essence	basé station	-73.6110781935155	45.5432588754923	2020-12-31 16:15:01	trip event
2021-01-01	28343008.0	6058	utilisé	513	-73.5851681388781	45.5220905471832	47.0	2.0	essence	basé station	-73.5851681388781	45.5220905471832	2020-12-31 13:00:46	trip event
2021-01-01	28247146.0	3370	utilisé	515	-73.565315	45.499713	47.0	8.0	hybride	basé station	-73.5659698092237	45.4985888647257	2020-12-30 17:29:46	trip event
2021-01-01	27887694.0	4937	utilisé	155	-73.578052	45.532228	47.0	8.0	hybride	basé station	-71.5225115701693	46.7472854578594	2020-12-28 15:06:01	trip event
2021-01-01	27887638.0	5942	utilisé	235	-73.52092	45.5261	47.0	8.0	essence	basé station	-73.5211682025805	45.5262088275688	2020-12-23 00:06:01	trip event
2021-01-01	28257712.0	6051	utilisé	894	-73.60003	45.554603	47.0	8.0	essence	basé station	-73.599954821543	45.5545321997089	2020-12-27 19:01:43	trip event
2021-01-01	28273552.0	5993	utilisé	221	-73.568174	45.4569949	47.0	8.0	essence	basé station	-73.5681564800701	45.4569121991473	2020-12-26 17:06:01	trip event
2021-01-01	28260989.0	6768	utilisé	155	-73.576052	45.532228	47.0	8.0	essence	basé station	-73.5762081360618	45.5323038959504	2020-12-25 18:15:01	trip event
2021-01-01	28222234.0	6409	utilisé	427	-73.608657	45.539313	47.0	1.0	essence	basé station	-73.6092548781628	45.5401122252988	2020-12-31 13:47:12	trip event
2021-01-01	28327166.0	5994	utilisé	520	-73.612018	45.523142	47.0	1.0	essence	basé station	-73.6117831953742	45.5233538675986	2020-12-31 22:02:33	trip event
2021-01-01	28329337.0	2805	utilisé	1032	-73.6183498296169	45.5425388700273	47.0	1.0	hybride	flex	-73.6183498296169	45.5425388700273	2020-12-31 16:30:38	trip event
2021-01-01	28282343.0	2857	utilisé	193	-73.570451528922	45.463515546125	47.0	1.0	hybride	basé station	-73.570451528922	45.463515546125	2020-12-26 11:44:17	trip event
2021-01-01	28001372.0	3034	utilisé	377	-73.568799874942	45.5476271880434	47.0	10.0	hybride	basé station	-73.568799874942	45.5476271880434	2020-12-30 18:21:54	trip event
2021-01-01	28355203.0	6744	utilisé	182	-73.6048231984252	45.5380771830771	47.0	1.0	essence	basé station	-73.6048231984252	45.5380771830771	2020-12-23 08:25:01	trip event
2021-01-01	28350755.0	3355	utilisé	1032	-73.5670281921115	45.5901038224411	47.0	10.0	hybride	flex	-73.5670281921115	45.5901038224411	2020-12-31 11:34:01	trip event
2021-01-01	28336304.0	6207	utilisé	996	-73.577327	45.543685	47.0	1.0	essence	basé station	-73.5772965261622	45.5436721871277	2020-12-30 19:36:01	trip event
2021-01-01	28332804.0	3379	utilisé	478	-73.6379248419907	45.4715454929879	47.0	11.0	hybride	basé station	-73.6379248419907	45.4715454929879	2020-12-17 17:16:04	trip event
2021-01-01	28338849.0	4946	utilisé	190	-73.620273141072	45.5073971550983	47.0	1.0	hybride	basé station	-73.620273141072	45.5073971550983	2020-12-26 14:22:43	trip event
2021-01-01	28188686.0	6064	utilisé	198	-73.663833	45.556741	47.0	1.0	essence	basé station	-73.6641398302581	45.5568338702641	2020-12-30 12:06:01	trip event
2021-01-01	28388178.0	4987	utilisé	933	-73.6744748004374	45.5371905453811	47.0	1.0	hybride	basé station	-73.6744748004374	45.5371905453811	2020-12-26 19:13:32	trip event
2021-01-01	28346648.0	6066	utilisé	156	-73.5663614955494	45.5224155135613	47.0	1.0	essence	basé station	-73.5663614955494	45.5224155135613	2020-12-22 13:25:48	trip event
2021-01-01	28348275.0	6177	utilisé	1001	-73.6109431610587	45.5431771519383	47.0	1.0	essence	basé station	-73.6109431610587	45.5431771519383	2020-12-30 11:59:12	trip event

Figure B. 1: Aperçu du fichier de l'état du réseau

ANNEXE C APERÇU DU FICHER DES CIRCONSTANCES

datetime	res_type	u_type	v_id	v_type_moc	res_type	longitude_res	latitude_res	nb_veh_avail	nb_ffcs_avail_250	nb_sbcs_avail_250	nb_ffcs_avail_500	nb_sbcs_avail_500	nb_ffcs_avail_750	nb_sbcs_avail_750	nb_EV_250	nb_EV_500	nb_EV_750	nb_veh_avail_250	nb_veh_avail_500	nb_veh_avail_750
2021-01-01	47.0	1.0	6883	FV	sbcs	-73.590658	45.524059	1033	0	10	0	17	0	32	0	0	0	10	17	32
2021-01-01	47.0	1.0	6105	FV	sbcs	-73.5762731796288	45.4814438498416	1033	0	2	0	5	0	6	0	0	0	2	5	6
2021-01-01	47.0	1.0	4480	HV	sbcs	-73.5665	45.5224	1033	0	7	0	26	0	30	0	0	0	7	26	30
2021-01-01	47.0	1.0	5944	FV	sbcs	-73.590658	45.524059	1033	0	10	0	17	0	32	0	0	0	10	17	32
2021-01-01	47.0	1.0	6727	HV	sbcs	-73.5862598817397	45.5372305270582	1033	0	1	0	2	0	8	0	0	0	1	2	8
2021-01-01	47.0	1.0	5938	FV	sbcs	-73.58154	45.53978	1033	0	1	0	4	0	5	0	0	0	1	4	5
2021-01-01	47.0	10.0	6610	FV	sbcs	-73.6273315415619	45.5436005218573	1033	0	3	0	7	0	8	0	0	0	3	7	8
2021-01-01	47.0	1.0	4573	HV	sbcs	-73.540899	45.553426	1033	0	0	0	3	0	3	0	0	0	0	3	3
2021-01-01	47.0	1.0	6612	FV	sbcs	-73.540899	45.553426	1033	0	0	0	3	0	3	0	0	0	0	3	3
2021-01-01	47.0	1.0	6145	FV	sbcs	-73.610986	45.543303	1033	0	2	0	5	0	10	0	0	0	2	5	10
2021-01-01	47.0	2.0	6058	FV	sbcs	-73.5851681388781	45.5220905471832	1033	0	3	0	18	0	28	0	0	0	3	18	28
2021-01-01	47.0	8.0	3370	HV	sbcs	-73.565315	45.499713	1033	0	1	0	6	0	9	0	0	0	1	6	9
2021-01-01	47.0	8.0	4937	HV	sbcs	-73.576052	45.532228	1033	0	1	0	6	0	9	0	0	0	1	6	9
2021-01-01	47.0	8.0	5942	FV	sbcs	-73.52092	45.5261	1033	0	2	0	4	0	5	0	0	0	2	4	5
2021-01-01	47.0	8.0	6051	FV	sbcs	-73.60003	45.554603	1033	0	1	0	4	0	5	0	0	0	1	4	5
2021-01-01	47.0	8.0	5993	FV	sbcs	-73.568174	45.4569949	1033	0	1	0	1	0	11	0	0	0	1	1	11
2021-01-01	47.0	8.0	6768	FV	sbcs	-73.576052	45.532228	1033	0	1	0	6	0	9	0	0	0	1	6	9
2021-01-01	47.0	1.0	6409	FV	sbcs	-73.608657	45.539313	1033	0	3	0	11	0	11	0	0	0	3	11	11
2021-01-01	47.0	1.0	5994	FV	sbcs	-73.612018	45.523142	1033	0	2	0	6	0	9	0	0	0	2	6	9
2021-01-01	47.0	1.0	2805	HV	ffcs	-73.6183498296169	45.5425388700273	1033	0	0	0	3	0	9	0	0	0	0	3	9
2021-01-01	47.0	1.0	2857	HV	sbcs	-73.570451528922	45.463515546125	1033	0	10	0	10	0	12	0	0	0	10	10	12
2021-01-01	47.0	10.0	3034	HV	sbcs	-73.568799874942	45.5476271880434	1033	0	2	0	2	0	5	0	0	0	2	2	5
2021-01-01	47.0	1.0	6744	FV	sbcs	-73.6048231984252	45.5380771830771	1033	0	2	0	5	0	7	0	0	0	2	5	7
2021-01-01	47.0	10.0	3355	HV	ffcs	-73.5670281921115	45.5901038224411	1033	0	0	0	0	0	1	0	0	0	0	0	1
2021-01-01	47.0	1.0	6207	FV	sbcs	-73.577327	45.543685	1033	0	1	0	1	0	6	0	0	0	1	1	6
2021-01-01	47.0	11.0	3379	HV	sbcs	-73.6379248419907	45.4715454929879	1033	0	0	0	0	0	4	0	0	0	0	0	4
2021-01-01	47.0	1.0	4946	HV	sbcs	-73.620273141072	45.5073971550983	1033	0	0	0	0	0	1	0	0	0	0	0	1
2021-01-01	47.0	1.0	6064	FV	sbcs	-73.663833	45.556741	1033	0	5	0	8	0	10	0	0	0	5	8	10
2021-01-01	47.0	1.0	4987	HV	sbcs	-73.6744748004374	45.5371905453811	1033	0	0	0	0	0	1	0	0	0	0	0	1
2021-01-01	47.0	1.0	6066	FV	sbcs	-73.5663614955494	45.5224155135613	1033	0	6	0	26	0	30	0	0	0	6	26	30
2021-01-01	47.0	1.0	6177	FV	sbcs	-73.6109431610587	45.5431771519383	1033	0	2	0	5	0	10	0	0	0	2	5	10
2021-01-01	47.0	1.0	7093	FV	sbcs	-73.5814	45.5226	1033	0	10	0	13	0	19	0	0	0	10	13	19
2021-01-01	47.0	1.0	6831	FV	sbcs	-73.667037	45.55475	1033	0	2	0	6	0	7	0	0	0	2	6	7
2021-01-01	47.0	1.0	5993	HV	sbcs	-73.6081	45.5454	1033	0	4	0	7	0	9	0	0	0	4	7	9

Figure C. 1: Aperçu du fichier des circonstances