Data Velib' Company

Contenu

I. Présentation de l'entreprise

À propos de Data Vélib :

Data Vélib' est un projet d'analyse de données issues du réseau Vélib' Métropole, l'un des principaux services de vélos en libre-service d'Europe. Ce système, qui couvre Paris et ses environs, met à disposition des utilisateurs des vélos mécaniques et électriques. Avec plus de 1 393 stations réparties entre Paris intra-muros et la périphérie, Vélib' joue un rôle clé dans la mobilité urbaine durable, en proposant une alternative pratique et écologique aux modes de transport traditionnels.

II. Cadre et contexte de l'analyse

Thèmes : Mobilité et Espace Public

Entreprise : Vélib' Métropole

Jeu de données :

L'analyse porte sur la disponibilité des bornettes, des vélos mécaniques et électriques dans les 1 393 stations de Vélib'. Ces stations sont réparties comme suit :

Paris intra-muros : 973 stations (70 %)

Périphérie parisienne : 420 stations (30 %)

Données analysées :

Période : Mars, Avril, Mai 2022 (31 jours pour le mois de mai)

Taille des données : 1 fichier csv de 913 Mo

Type de données : Fichier plat

Objectifs

- Améliorer la rentabilité des ressources de l'entreprise
-
- L'objectif principal de cette analyse est d'optimiser la gestion des stations Vélib' en se concentrant sur la disponibilité des vélos et des bornettes, afin de maximiser l'utilisation des ressources et réduire les inefficacités opérationnelles.
-

Sous-objectifs

- Analyse des disponibilités :
- Étudier la répartition et la disponibilité des vélos mécaniques et électriques dans les 1 393 stations.
- Identifier les stations sous-utilisées ou en surcharge régulière.
- Évaluer les écarts entre l'offre et la demande selon les périodes horaires et les zones géographiques (Paris intra-muros vs périphérie).

Optimisation des flux

- Proposer des stratégies pour redistribuer les vélos entre les stations afin de répondre aux pics de demande.
- Réduire le temps de rotation des vélos (durée pendant laquelle un vélo reste inutilisé).

Amélioration de l'expérience utilisateur

- Garantir une meilleure disponibilité des vélos pour les utilisateurs lors des heures de pointe.
- Minimiser les problèmes de station pleine (impossible de rendre un vélo) ou vide (impossible d'en louer un).

Rentabilité

- Maximiser l'utilisation des ressources (vélos et bornes) pour accroître les revenus liés aux abonnements et locations ponctuelles.
- Réduire les coûts opérationnels liés à la gestion des vélos inutilisés ou à leur redéploiement.

Outils utilisés :

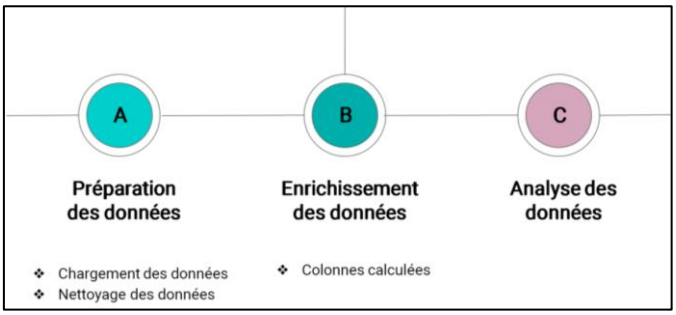
Python : Pour le nettoyage et la préparation des données.

Power BI et Qlik Sense : Pour l'analyse visuelle et les tableaux de bord interactifs.

Jupyter Notebook : Pour documenter et exécuter les analyses.

Temps total de l'analyse : 5 jours

III. Méthodologie



Étapes de l'analyse *: Préparation des données, enrichissement, analyse*

Préparation des données :

Consolidation et nettoyage des fichiers plats, identification des éventuelles incohérences, et transformation des données pour une analyse plus fluide.

Choix des KPI :

Traitement de qualité de données :

Qualité des données :

Qualité de données		
Modifications	Données manquantes	Doublons
Taux	_	**47%**
Standardisation des formats	coordonnées_geo: Float stationcode : string et int	
Analyse sur 1420 stations	• 23 stations : incohérence de capacité • 4 stations : inactives • Augmentation de 31 bornes sur une station • 2 stations : manque installation de bornes • Ouverture de 3 stations	

Analyse des KPIs :

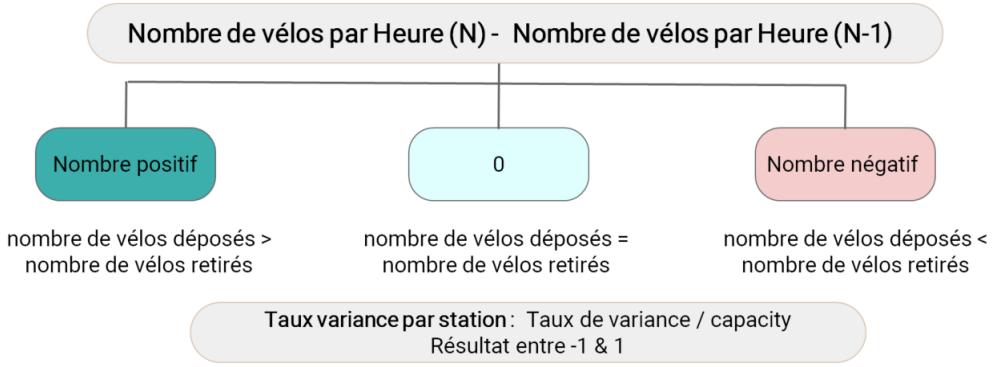
Calcule du taux moyen de vélos et bornes disponibles :

- $taux\_velo = \frac{\text{Nombre de vélos disponibles}}{\text{Capacité}} x100$
- $taux\_velo = \frac{\text{Nombre de bornes disponibles}}{\text{Capacité}} x100$

Calcule de la variance :

Objectif: Obtenir les activités de chaque station (retraits et dépôts des vélos)

Calcul de la différence sur une journée du nombre de vélos disponible en fonction des heures et par station.



IV. Explication Technique et Outils Utilisés

Jupyter notebook (python) : Pandas,

Fonctions : pd.read_csv(), df.head(), df.info(), pd.to_datetime(), df.to_csv(), os.listdir(), pd.concat(), reset_index(), drop(), rename(), to_excel(), isnull().sum(), duplicated().sum(), sort_values(), pd.read_excel(), astype, info, select_dtypes, pd.Index, plt.figure, sns.displot, plt.xlabel, and plt.show.

Power BI : visualisation

Qlik Sense : visualisation

Description Script python :

L'objectif du **scrpit « 06-15-prep-together »** est de préparer un jeu de données pour l'analyse de la disponibilité des stations de vélos. Le jeu de données est traité pour :

Fusionner les informations : Combiner les données provenant de différentes sources (par exemple, les informations sur les stations de vélos et les statistiques de disponibilité).

Nettoyage et transformation des données : Convertir les types de données, extraire les informations nécessaires et dériver de nouvelles fonctionnalités pour soutenir les analyses ultérieures.

Ingénierie des fonctionnalités : Créer de nouvelles variables qui aideront à comprendre la disponibilité des vélos et les modèles opérationnels, comme l'identification des jours ouvrables.

Étapes et Méthodologie :

Fusion des jeux de données :

Le script commence par fusionner df_s avec moy_taux_borne sur la colonne stationcode. Cela crée un jeu de données combiné qui inclut les détails de la station et les métriques de disponibilité des vélos (par exemple, taux_borne_dispo, qui représente le pourcentage de vélos disponibles).

```
python

df_s = df_s.merge(moy_taux_borne, how="left", on="stationcode")
```

Conversion des types de données :

La colonne capacity est convertie en type entier (int64) pour garantir la cohérence des opérations numériques.

```
python

df['capacity'] = df['capacity'].astype('int64')

Extraction des informations sur la date et l'heure :
```

La colonne duedate est séparée en une date et une heure. La date est convertie en un objet datetime pour faciliter la manipulation, et l'heure est extraite de la chaîne de caractères pour une analyse ultérieure.

```
python

datetimestr = df['duedate'].str.split(pat='T', expand=True)

date = pd.to_datetime(datetimestr[0], format="%Y-%m-%d")

hour = datetimestr[1].str[0:2].astype('int64')
```

Création de nouvelles fonctionnalités :

Jour de la semaine : Le jour de la semaine est dérivé de la colonne date et converti en un format lisible (par exemple, "Lundi", "Mardi").

```
python

df['dayofweek'] = df['date'].dt.dayofweek

df['dayofweek'] = df['dayofweek'].replace({

   0:"Lundi", 1:"Mardi", 2:"Mercredi", 3:"Jeudi",

   4:"Vendredi", 5:"Samedi", 6:"Dimanche"

})
```

Indicateur de jour ouvrable : Une nouvelle colonne workday est créée, avec la valeur par défaut de 1, indiquant un jour ouvrable. Le script ajuste ensuite cette valeur à 0 pour les week-ends (samedi, dimanche) et les jours fériés spécifiés (par exemple, 2022-05-26).

```
python

df['workday'] = 1

holidays = ["2022-05-26"]

df['date'] = pd.to_datetime(df['date'])

c = ((df['dayofweek'] == "Samedi") |

    (df['dayofweek'] == "Dimanche") |

    (df['date'].dt.strftime('%Y-%m-%d').isin(holidays)))

df.loc[c, 'workday'] = 0
```

Ajustements finaux des données et exportation :

Les colonnes inutiles (par exemple, name, coordonnees_geo, nom_arrondissement_communes) sont supprimées pour se concentrer sur les variables pertinentes pour l'analyse des stations de vélos.

Le jeu de données final est ensuite exporté dans un fichier CSV (stationstatus.csv), prêt pour une analyse ou un rapport ultérieur.

```
python

df.to_csv('output1/stationstatus.csv', index=False)
```

Caractéristiques résultantes dans le jeu de données :

Après l'exécution de ce script, le jeu de données contient les colonnes suivantes :

stationcode : L'identifiant unique de chaque station.

ebike, mechanical : Le nombre de vélos électriques et mécaniques disponibles.

duedate : L'horodatage pour les données de disponibilité des vélos.

numbikesavailable : Le nombre de vélos disponibles à la station.

numdocksavailable : Le nombre de docks disponibles à la station.

capacity : La capacité totale de la station (nombre de docks).

is_renting, is_installed, is_returning : Indicateurs de statut de la station.

taux_borne_dispo : Le pourcentage de disponibilité des vélos à chaque station.

date : La partie date de duedate.

hour : L'heure extraite de duedate.

dayofweek : Le jour de la semaine correspondant à chaque entrée.

workday : Un indicateur binaire pour savoir si l'entrée correspond à un jour ouvrable (1) ou non (0).

Cas d'utilisation :

→ Identifier les tendances de la disponibilité des vélos selon les jours de la semaine (week-ends vs jours ouvrables).
→ Analyser l'impact des jours fériés sur l'occupation des stations de vélos.
→ Comparer la disponibilité des vélos par rapport à la capacité des stations pour optimiser la répartition des vélos et leur utilisation.

scrpit « 06-12-analyse-stationflux »

Chargement des données : Le jeu de données est supposé être déjà chargé dans un DataFrame Pandas df, contenant 3 267 940 lignes et 14 colonnes. Les colonnes représentent divers attributs tels que la disponibilité des vélos, les détails des stations et les statuts opérationnels.

Conversion des types de données : Le script convertit certaines colonnes de type int64 en type category. Les colonnes hour, day, dayofweek et workday sont transformées en types catégoriels pour optimiser l'utilisation de la mémoire et faciliter l'analyse. La conversion de ces colonnes peut améliorer les performances lors du traitement de grands jeux de données et permet une interprétation plus facile dans certains types d'analyses.

Colonnes transformées :

- hour
- day
- dayofweek
- workday

Exploration des types de données : Le script vérifie les types de données du DataFrame avec la méthode df.info() pour s'assurer que les transformations ont été appliquées correctement. Après la conversion, le DataFrame contient quatre colonnes de type catégoriel et six colonnes de type entier, avec une utilisation mémoire de 261,8 Mo.

Sélection des caractéristiques numériques : Une liste des colonnes numériques est générée en utilisant select_dtypes() pour identifier les caractéristiques qui seront impliquées dans l'analyse des distributions. Ces caractéristiques comprennent :

- stationcode
- ebike
- mechanical
- numbikesavailable
- numdocksavailable
- capacity

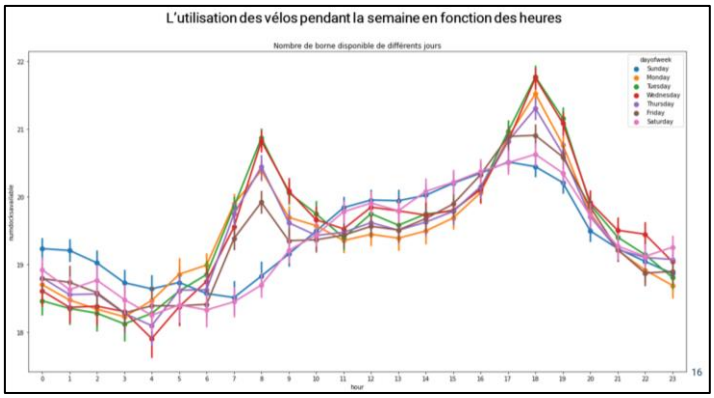
Analyse des distributions : Le script vise à visualiser les distributions des caractéristiques numériques en utilisant displot() de Seaborn. Cependant, la sortie attendue des graphiques de distribution n'est pas rendue, comme l'indiquent les avertissements d'absence de figure (<Figure size 720x432 with 0 Axes>). Cela suggère qu'il pourrait y avoir un problème avec la manière dont les figures sont affichées ou les données passées à displot().

Données finales : Le DataFrame df est affiché à la fin du script, montrant les premières lignes, avec les colonnes catégorielles transformées (hour, dayofweek, workday, et day) apparaissant aux côtés des autres colonnes comme stationcode, ebike, mechanical et les données de disponibilité.

V. Visualisation des Résultats

Après la préparation des données avec Python, plusieurs outils de visualisation (Python Plot, Power BI et Qlik Sense) ont été utilisés pour analyser et représenter les données sous différents angles.

1. L'utilisation des vélos pendant la semaine en fonction des heures



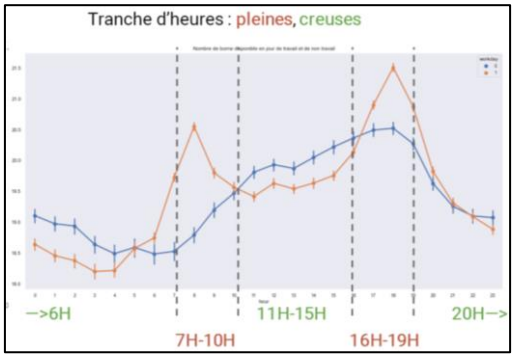
Graphique généré avec python : pic d'utilisation vélos pendant la semaine en fonction des heures

Objectif : Étudier les périodes de forte et faible utilisation des vélos sur l'ensemble de la semaine, heure par heure.

Résultats attendus : Identification des pics d'utilisation durant les heures de pointe (matin et soir) et des creux en heures creuses.

- bornes disponibles = nombre de vélos en utilisation
- dans la semaine, les gens prends des vélos le plus entre 6H et 10H et entre 16H et 20H pour aller travailler et rentrer, plus à 18H
- dans le weekend, les gens prends des vélos plutôt entre 9H et 21H, plus à 18H

2. calcul par tranches : heures capacité et jours de la semaine



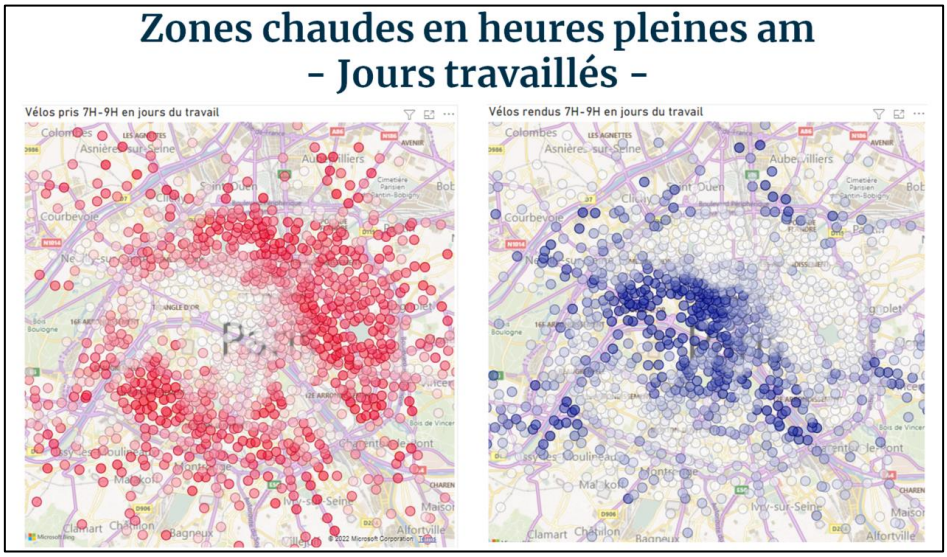
Graphique généré avec python : les heures pleines et creuse entre 6h et 20h

- **Courbe orange** : du lundi au vendredi => jours travaillés
- **Courbe bleue** : les weekends et jour fériés

Objectif : Analyser l'utilisation des vélos en tenant compte des créneaux horaires, des capacités des stations et des jours de la semaine (travail vs week-end).
Résultats attendus : Mise en évidence des variations de la demande selon la capacité des stations et les jours spécifiques

3. Les zones chaudes en heures pleines am

Objectif : Identifier les stations les plus utilisées durant les heures de pointe matinales les jours ouvrés.



Graphique généré avec powerbi : Utilisation des vélos 7h-9h en jours ouvrés

Les deux graphiques représentent l'utilisation de vélos dans une ville pendant les heures de pointe (7h-9h) les jours de travail. Le graphique de gauche montre où les vélos sont pris, et le graphique de droite montre où ils sont déposés.

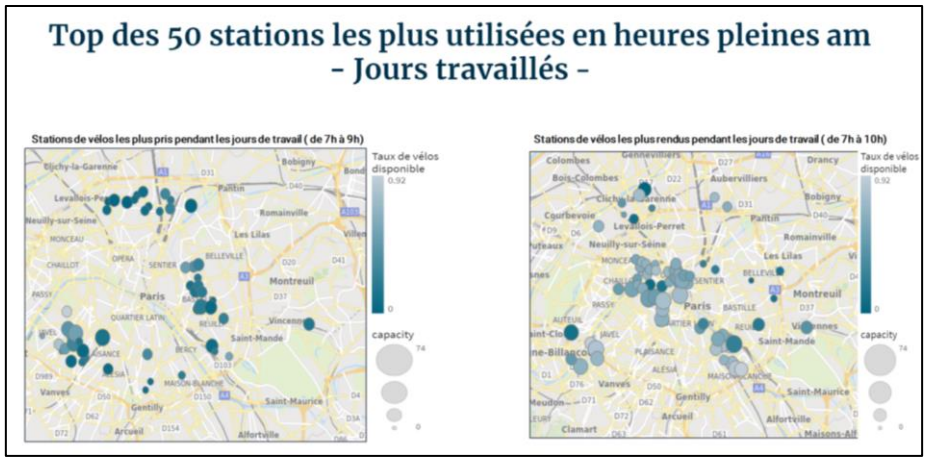
Observations clés

Concentration des prises et des dépôts: Les deux graphiques montrent une concentration claire de l'activité cycliste dans la partie centrale de la ville, en particulier autour de la zone étiquetée "Paris". Cela suggère que la zone centrale est une destination populaire pour les déplacements à vélo pendant les heures de pointe.

Direction de déplacement: En comparant les deux graphiques, on peut voir qu'il y a un flux général de vélos des zones extérieures vers le centre de la ville. Ceci est plus évident dans le graphique de droite, qui montre les lieux de dépôt.

Densité de l'activité: La taille et la couleur des points indiquent la densité de l'activité cycliste. Plus les points sont grands et sombres, plus la concentration de vélos est élevée. Cette visualisation nous aide à comprendre les zones les plus fréquentées pour l'utilisation des vélos pendant les heures de pointe.

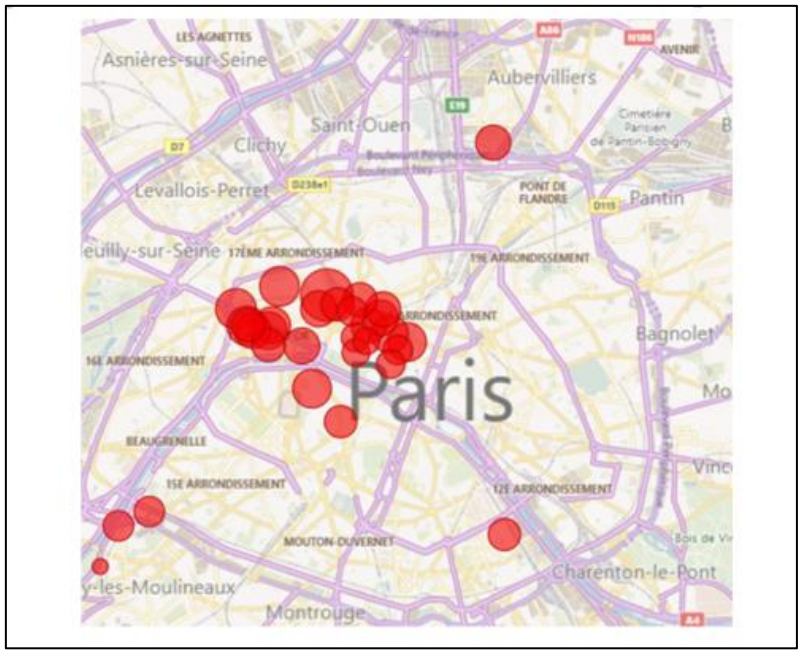
4. Top des 50 stations les plus utilisées en heures pleines am



Graphique généré avec Qlick sense : Utilisation des vélos 7h-9h en jours ouvrés

Objectif : Classer les stations ayant la plus forte demande durant les heures de pointe matinales

5. Les Stations hyper chaudes de manière générale



Graphique généré avec powerbi : 32 stations hyper chaudes

Beaucoup de gens prennent les vélos mais il y a très peu de vélos disponibles

Objectif : Identifier les stations où l'offre est insuffisante par rapport à la demande.

Critères d'identification :

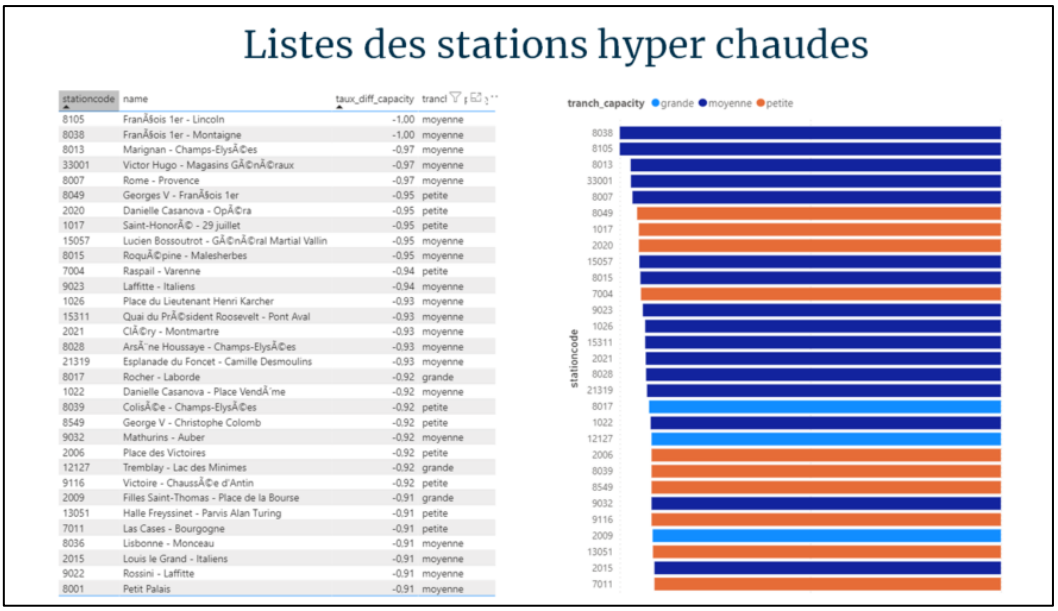
- Taux de variance < -0,9.
- Taux de vélos disponibles < 0,1.

Compositions :

- 3 grandes stations.

- 19 moyennes stations.
- 10 petites stations.

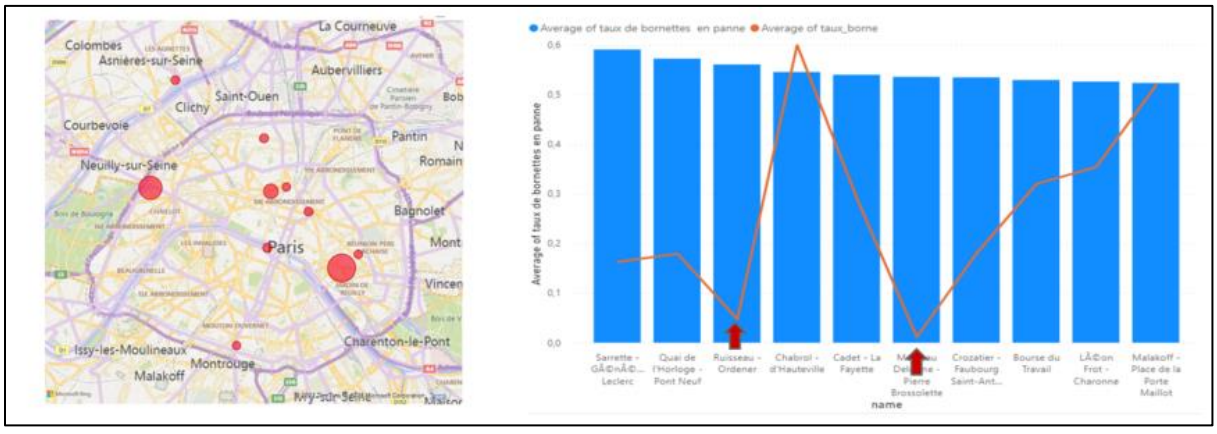
Localisation : 30 à Paris, 1 à Issy-les-Moulineaux et 1 à Aubervilliers.



stationcode	name	taux_diff_capacity	trancl
8105	FranÃ§ois 1er - Lincoln	-1.00	moyenne
8038	FranÃ§ois 1er - Montaigne	-1.00	moyenne
8013	Marignan - Champs-ElysÃ©es	-0.97	moyenne
33001	Victor Hugo - Magasins GÃ©nÃ©raux	-0.97	moyenne
8007	Rome - Provence	-0.97	moyenne
8049	Georges V - FranÃ§ois 1er	-0.95	petite
2020	Danielle Casanova - OpÃ©ra	-0.95	petite
1017	Saint-HonorÃ© - 29 juillet	-0.95	petite
15057	Lucien Bossoutrot - GÃ©nÃ©ral Martial Vallin	-0.95	moyenne
8015	RoquÃ©pine - Malesherbes	-0.95	moyenne
7004	Raspail - Varenne	-0.94	petite
9023	Laffitte - Italiens	-0.94	moyenne
1026	Place du Lieutenant Henri Karcher	-0.93	moyenne
15311	Quai du PrÃ©sident Roosevelt - Pont Aval	-0.93	moyenne
2021	ClÃ©ry - Montmartre	-0.93	moyenne
8028	ArsÃ¨ne Houssaye - Champs-ElysÃ©es	-0.93	moyenne
21319	Esplanade du Foncet - Camille Desmoulins	-0.93	moyenne
8017	Rocher - Laborde	-0.92	grande
1022	Danielle Casanova - Place VendÃ´me	-0.92	moyenne
8039	ColisÃ©e - Champs-ElysÃ©es	-0.92	petite
8549	George V - Christophe Colomb	-0.92	petite
9032	Mathurins - Auber	-0.92	moyenne
2006	Place des Victoires	-0.92	petite
12127	Tremblay - Lac des Minimes	-0.92	grande
9116	Victoire - ChaussÃ©e d'Antin	-0.92	petite
2009	Filles Saint-Thomas - Place de la Bourse	-0.91	grande
13051	Halle Freyssinet - Parvis Alan Turing	-0.91	petite
7011	Las Cases - Bourgogne	-0.91	petite
8036	Lisbonne - Monceau	-0.91	moyenne
2015	Louis le Grand - Italiens	-0.91	moyenne
9022	Rossini - Laffitte	-0.91	moyenne
8001	Petit Palais	-0.91	moyenne

Liste des 32 stations hyper chaudes

- 63% du temps sur 1390 stations
- en moyen 2 bornes par station
- tout le temps sur 76 stations
- taux des bornes en panne > 50% - 10 stations (noms voir annexe)
- critères : taux de bornes disponible, taux de bornes en panne / totale par station



Les bornes en panne

VI. Conclusion

Axes d'amélioration :

Pour offrir une meilleure expérience utilisateur, plusieurs problématiques doivent être adressées, notamment la gestion des pannes, l'équilibre entre l'offre et la demande dans les stations, et l'optimisation des ressources. Voici les solutions proposées :

1. Résolution des problèmes de pannes

- Analyse des bornettes en panne :

Si le taux de bornettes disponibles est faible, une maintenance prioritaire doit être planifiée pour améliorer rapidement la disponibilité des bornettes.

Impact attendu : Réduction des stations où les utilisateurs ne peuvent pas restituer les vélos.

2. Assurer un équilibre dans les stations

- Camion dédié aux 50 stations les plus utilisées :

Déploiement d'un camion logistique focalisé sur les stations à forte demande entre 7h et 10h les jours travaillés.

Objectif : Résoudre et anticiper les problèmes de vélos non disponibles en assurant un réapprovisionnement en temps réel.

- Analyse de variance :

Pour les zones chaudes en heures pleines, organiser le déplacement des vélos vers les stations les plus fréquentées.

Pour les stations hyper chaudes, augmenter leur capacité en ajoutant davantage de bornes et vélos.

3. Gestion des cas "0 vélos" et "0 bornettes disponibles"

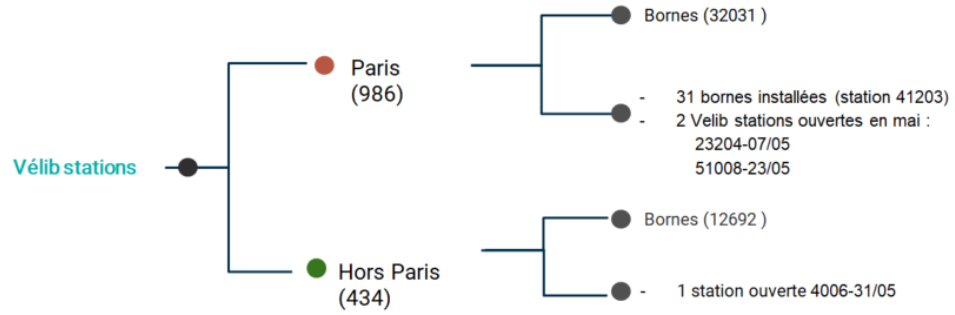
- Analyse des heures pleines :

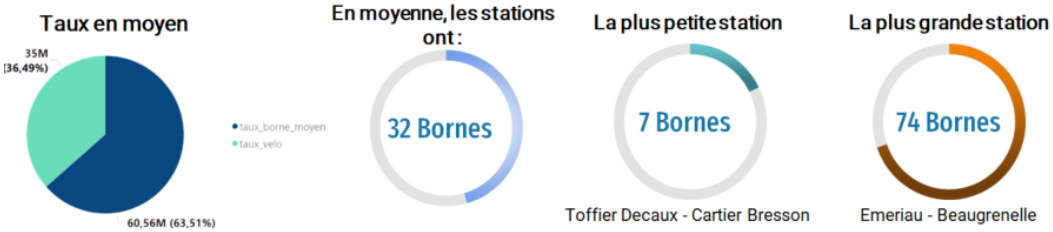
Redistribuer les vélos pendant les périodes de forte utilisation (jours ouvrés, heures de pointe) afin de limiter la surexploitation des vélos.

- Proposition tarifaire :

Introduire des tarifs d'abonnement légèrement plus élevés pour limiter la surutilisation excessive tout en générant des revenus supplémentaires pour le maintien des infrastructures.

Données détaillées





10 stations - taux de bornettes en panne >50%

- Sarette - Général Leclerc
- Quai de l'Horloge - Pont Neuf
- Ruisseau - Ordener
- Chabrol - d'Hauteville
- Cadet - La Fayette
- Marceau Delorme - Pierre
- Crozatier - Faubourg
- Bourse du Travail
- Léon Frot - Charonne
- Malakoff - Place de la Porte Maillot