Olivier Lebert

Analyse des profils horaires de fréquentation et modélisation des passages sur les stations de comptage vélo à Bruxelles

Contexte

Dans le cadre du projet *Bruxelles Mobilité*, l'objectif de cette analyse consiste à distinguer les profils horaires des passages vélos enregistrés sur les 18 stations de comptage actuelles à Bruxelles. Ces analyses permettent de mieux comprendre les dynamiques de mobilité douce en ville, d'identifier les variables explicatives majeures, et d'améliorer la prévision des flux à l'aide de modèles statistiques avancés.

En combinant des approches descriptives, des méthodes de réduction de dimensionnalité (ACP), de classification (analyses de clusters), et des techniques de modélisation (régressions et XGBoost), le projet vise à fournir des outils d'aide à la décision pour le pilotage des infrastructures cyclables.

Principales conclusions

- Les **profils horaires** varient significativement selon les **jours** (semaine vs week-end) et la **météo**.
- L'**ACP** et le **clustering** révèlent des groupes de stations aux comportements distincts, en lien avec le **contexte urbain** (axes majeurs, centralité) et la météo.
- Partout, l'usage du vélo diffère nettement **en semaine** (pics pendulaires) et **le week-end** (profil plus étalé).
- La **régression linéaire** explique une part notable de la variabilité, mais le **modèle XGBoost (Poisson)** offre des **performances prédictives nettement supérieures (R2 = 0.85 vs 0.715)**.
- Un **tableau de bord Power BI** permet d'explorer interactivement les résultats et de visualiser les effets croisés (heure, période, station, …).

Dans la suite du rapport sont abordés : l'**exploration** et la **préparation** des données, l'**interprétation** des **dimensions** issues de l'ACP, celle des **clusters**, les **hypothèses et résultats** des régressions linéaires, puis l'**interprétation** du modèle **XGBoost**, et enfin le **tableau de bord Power BI**.

Exploration et préparation

L'analyse commence par l'**import** et le **nettoyage** des données issues des stations de comptage vélo. Les données brutes sont harmonisées :

- Import depuis la base SQL (MariaDB).
- Uniformisation des **formats de date**.
- Calcul de **variables dérivées** : *month*, *jour_semaine*, *is_weekend*, *hourUTC1*, etc.
- Jointure avec la **météo** (température, vent, pression, pluie, nébulosité, ensoleillement).
- Traitement des **valeurs manquantes** (imputations simples).

Des **profils horaires moyens** sont ensuite calculés pour chaque station et type de jour (semaine / week-end), afin d'identifier les **pics** (matin/soir en semaine, profils plus lisses le week-end).

Analyse en composantes principales (ACP)

L'ACP est réalisée sur les profils horaires (matrice centrée-réduite) afin de réduire la dimension et de visualiser les proximités entre stations/heure sans que les heures à forte variance écrasent l'information.

Les deux premiers axes expliquent une part significative de la variance (critère de Kaiser et méthode du coude).

L'axe 1 oppose les stations à forte fréquentation avec pics marqués aux heures de pointe à des profils plus faiblement fréquentés et lisses.

L'axe 2 (part d'info plus faible ~15 %) quant à lui, sépare nettement semaine et week-end. Par ailleurs, le cercle des corrélations permet d'observer que les heures 0–8h sont mieux portées par l'axe 2 (activité nocturne du week-end plus élevée), tandis que le reste de la journée est surtout structuré par l'axe 1 (pics de pointe en semaine).

Clustering (HCPC)

À partir des scores d'ACP, un clustering hiérarchique (Ward) est appliqué puis consolidé par k-means. L'analyse d'inertie retient une partition en 2 groupes.

D'une part, les stations **CJM90** et **CB1101** se distinguent nettement (cluster 2) aussi bien semaine que week-end.

D'autre part, **CB1143** et **CB02411** basculent dans le cluster 2 pour la composante **semaine** (week-end plus proche du cluster 1).

Hormis les 4 stations citées qui se démarquent, le **cluster 1** de son côté regroupe l'ensemble des 14 autres stations (profils plus "courants") pour leur composante de semaine comme celle de week-end.

Les **Eta²** élevés sur les heures de la fin de journée (h_15 à h_23) montrent que ces heures expliquent fortement la séparation des groupes :

Enfin, le **cluster 1** montre des v.test négatifs sur de nombreuses heures → profils moins fréquentés, sans pics marqués.
Le **cluster 2** v.test positifs en fin de journée → profils à forte fréquentation aux heures de pointes et en soirée.

Régression linéaire multiple (Python)

Deux OLS ont été estimées avec erreurs standards clusterisées par station (*FeatureID*) et un riche jeu de variables explicatives : *hourUTC1* (qualitatives), *jour_semaine* (qualitatives), *FeatureID* (qualitatives), ainsi que météo (*Speed, Ta_ucc, wind_speed_ucc, press_ucc, cloud_ucc, rain_ucc, solar_bxl*).

(1) Modèle sur Count

- **$R^2 = 0{,}527$** (Adj. $R^2 = 0{,}527$)
- Effets météo (signes attendus, tailles modestes) :

  - **rain_ucc** négatif (≈ −9,6), **cloud_ucc** négatif, **press_ucc** positif, **Ta_ucc** légèrement positif, **Speed** non significatif.
  - **solar_bxl** marginal (p≈0,062).

- Les variables relatives à **l'heure/le jour de la semaine/la station** portent l'essentiel de l'explication (pics horaires, effets jours, effets station).
- Diagnostics : histogramme de *Count* très **asymétrique** ; **résidus vs prédits** en **entonnoir** (hétéroscédasticité croissante) ; **Q-Q plot** à **queues lourdes**.
  ⇒ Le modèle capte bien la structure jour/heure/station mais ne respecte pas les hypothèses **d'homoscédasticité** et de **normalité**.

(2) Modèle sur log1p(Count)

- **R^2 = 0,715** (Adj. R^2 = 0,715)
- Interprétation multiplicative sur **log(1+Count)** :
  - **rain_ucc ≈ −0,150** ⇒ toutes choses égales par ailleurs, une hausse d'une unité de pluie est associée à **≈ −14 %** de passages en moyenne (approx. exp(−0,150)−1).
  - **cloud_ucc** négatif ; **press_ucc** positif mais faible ; **Ta_ucc** légèrement positif.
- Diagnostics : les résidus s'améliorent par rapport au premier modèle, mais on garde des **queues** et une **DW ≈ 0,40** (corrélation temporelle résiduelle, toutefois plus faible qu'au préalable DW ≈ 0.64).
  ⇒ Le **log-linéaire** améliore nettement l'ajustement mais n'annule pas complètement les déviations (données de comptage très asymétriques).

Conclusion régressions. Les OLS fournissent des repères interprétables (signes des variables météo, structure heure/jour/station), mais les hypothèses gaussiennes sont mises en défaut. Cela motive l'usage d'un modèle de comptage non linéaire et robuste : XGBoost avec objectif Poisson.

Modèle XGBoost (objectif count:poisson)

Le modèle final est un **XGBRegressor** configuré pour le comptage :
n_estimators=300, learning_rate=0,1, max_depth=4, subsample=0,8, colsample_bytree=0,8, reg_lambda=1, enable_categorical=True.
Le jeu de variables inclut *FeatureID, hourUTC1, jour_semaine, month, is_weekend* et la *météo*.

Performances (évaluées sur le test)

- MSE = 1 543,9
- R^2 = 0,85
- MAE ≈ 20,1 (en count)
- Poisson deviance moyenne ≈ 11,1

Diagnostics & importance

- **Réel vs Prédit (densité)** : les points se concentrent autour de la diagonale, avec quelques **écarts** aux volumes élevés.
- **Résidus vs prédits** : hétéroscédasticité attendue (variance croît avec l'intensité), mais structure globalement centrée.
- **Importance (gain)** : *FeatureID* et *hourUTC1* dominent (effets station et rythmes quotidiens), suivis de *solar_bxl, is_weekend, jour_semaine*, puis variables météo (pluie, température, etc.).
  ⇒ XGBoost capture la non-linéarités et les interactions et s'adapte aux écarts de variance propres aux données de comptage, d'où le gain substantiel en R^2 et en erreurs absolues.

Tableau de bord Power BI (suivi du modèle)

Les données finales sont exportées en CSV et importées dans Power BI, avec une table Dates reliée à *DateUTC1* pour piloter le contexte temporel. Les visuels sont synchronisés par filtres (Date, FeatureID, hourUTC1, jour, mois, split train/test, cluster).

Page 1 — Vue synthétique (niveau point/station)

- **KPI** :
  - ***Biais*** = Σ(Prédit) − Σ(Réel), et ***Biais %*** = Biais / Σ(Réel),
  - ***MAE***, ***WAPE %*** = Σ|biais| / Σ(Réel), ***RMSE*** et $\boldsymbol{R^2}$ = 1 − SSE/SST.
- **Nuage de points Réel vs Prédit**, coloré par FeatureID, avec ligne y=x pour juger le calage station par station.
- **Segments (filtres)** : période (DateUTC1), station, heure, jour, mois, split, cluster.

Page 2 — Vue agrégée & diagnostics

- **Courbe "Total Réel vs Total Prédit par Date"** : met en évidence la saisonnalité et les décalages (été, rentrée, dimanche sans voiture, etc.).
- **Heatmap** "WAPE % par jour × heure" avec mise en forme conditionnelle (rouge → vert) pour repérer heures/jours problématiques (2-5h entre 40 et 70%, pic à 100% le dimanche à 6h, meilleure performance entre 10-23h, environ 20-25%)
- **Barres** "WAPE % par station (FeatureID)" (tri décroissant) : priorise les stations à fort écart relatif pour l'amélioration (meilleure performance du modèle sur les stations du cluster 2).

Lecture : fixer la période, filtrer éventuellement la station/heure/jour, passer en **split=test** pour l'évaluation stricte ; lire **Biais/WAPE/MAE/RMSE/R^2**, puis croiser **scatter**, **heatmap** et **barres** pour localiser **où** et **quand** le modèle se trompe et **de quel côté** (sur/sous-estimation). Ce tableau de bord rend l'évaluation **actionnable** et guide des **améliorations ciblées**.

Les prédictions effectuées peuvent ainsi être utiles en vue de décider où placer de nouvelles stations de comptage, prévoir des investissements en infrastructure, réaliser des campagnes d'encouragement à l'emploi du vélo, mais aussi pour envisager de nouveaux types de données à ajouter pour améliorer le modèle, comme par exemple les évènements, les travaux, variables météo

plus fines, la typologie des axes (inclinaison, densité de la circulation, etc …) qui pourrait toutes expliquer une part de la variabilité résiduelle.

Conclusion

Les méthodes descriptives (profils horaires, ACP, clustering) confirment des rythmes différenciés selon le jour et la station, et mettent en avant un groupe de stations à forte activité en fin de journée et aux heures de pointes. Les régressions fournissent une lecture interprétable des effets (notamment météo), mais n'embrassent pas complètement la nature de comptage et l'hétéroscédasticité des données. Le modèle XGBoost (Poisson) s'impose pour la prédiction opérationnelle ($R^2 \approx 0{,}85$, MAE ≈ 20) tout en restant cohérent (aucune valeur négative). Enfin, le tableau de bord Power BI permet un suivi fin de la performance, par station et créneaux, et constitue un support concret pour le pilotage et l'amélioration continue du modèle.