l'institut Agro
agriculture • alimentation • environnement

Montpellier SupAgro

Stage de fin d'étude d'ingénieur agronome

Data Science pour l'Agronomie et l'Agroalimentaire

Modélisation et visualisation de la charge à bord en temps réel

Auteur :
Mathilde Vimont

Encadrant :
Rémi Coulaud

Mars - Août 2020

SNCF

Remerciements

Je remercie d'abord Marc et Chantal d'avoir su particulièrement bien gérer une situation aussi inédite que ce début d'année, et d'avoir fait en sorte de garder un contact et un esprit d'équipe fort tout au long du stage. Merci à toute l'équipe du Lab', anciens et nouveaux, avec qui j'ai eu l'occasion de travailler ces 6 derniers mois. Autant de séances de brainstorming pour un nom aussi simple qu'Hector, projet que je suis pourtant ravie de continuer avec vous pendant un peu de temps encore. Rémi, je te suis vraiment reconnaissante pour tout le temps que tu as pu me consacrer pendant ce stage et pour ton encadrement aussi bien mathématique que ferroviaire. Grâce à toi j'ai appris plein de choses dans un environnement toujours bienveillant, et j'ai hâte de voir quelles nouvelles aventures peuvent encore nous arriver ! Merci Léo enfin pour ces deux mois (deux mois !) de confinement partagés et toutes tes relectures.

Résumé

La crise sanitaire actuelle liée à la propagation du coronavirus a renforcé le besoin de rendre les transports en commun plus sûrs et attractifs. Beaucoup de facteurs ont été identifiés comme ayant un impact négatif sur la perception du service par les voyageurs. Fournir de l'information en temps réel sur l'état du trafic et l'affluence en gare et à bord des trains, est considéré comme ayant un effet positif sur le comportement et le confort des voyageurs. Il existe différentes sources de données disponibles donnant accès à une information d'affluence. Les systèmes de comptages automatiques des passagers permettent notamment d'estimer la charge (i.e, le nombre de passagers) à bord de chaque voiture. Ces systèmes ne prennent malheureusement pas en compte le mouvement des passagers entre les voitures, ce qui entraîne une estimation non fiable de la charge. Dans cette étude, nous proposons une méthodologie permettant de modéliser le mouvement des passagers à l'intérieur du train. Le modèle finalement sélectionné permet d'obtenir une estimation de la charge à bord plus réaliste que celle obtenue à partir des données brutes. Cette mise en qualité des données est une étape préliminaire à l'implémentation d'un site web, ayant pour objectif de rendre disponible une information fiable de charge à bord en temps réel.

Abstract

The recent spread of the COVID-19 virus has resulted in a significant need for making public transport safer and more attractive. Many factors have been identified that could have an effect on travellers perception of public transport. Among them, real-time information on traffic and crowding has been found to have a positive impact on travellers' welfare and behaviour. To make crowding information available for travellers, public transport rely on different data sources. Passenger counting systems is one of the tools that allow estimations of the load (i.e, number of passengers) in each car. Unfortunately, those systems do not take into consideration possible movements of passengers between the cars, resulting in unreliable load estimations. In this paper, we present a methodology to model the movement of passengers within the train. The selected model enables a better load estimation compared to the raw data. This data improvement is a preliminary step toward the creation of a website of which the goal is to make available real-time crowding information for passengers.

Table des matières

1 Contexte

1.1 Enjeux

Transilien est la société fille de la SNCF qui exploite les trains de banlieue d'Île-de-France. Ce réseau accueille un important volume de passagers chaque jour, particulièrement en heure de pointe (3,5 millions de voyageurs par jour en service normal). Une des problématiques actuelles de Transilien est l'ouverture à la concurrence et la perte du monopole sur le réseau ferré d'Île-de-France, d'où un besoin fort de rendre davantage attractif son service. « Dans un contexte d'ouverture à la concurrence, un enjeu majeur pour Transilien est de diminuer le stress des voyageurs pendant leurs trajets, voire de réussir à en faire un moment agréable » (Charles 2020). Ce besoin est renforcé par la situation exceptionnelle de crise sanitaire que nous traversons et qui influence négativement l'attractivité des transports en commun (De Vos 2020). Or, même dans un contexte normal, de nombreux facteurs peuvent avoir un impact négatif sur la perception des transports publics. Les périodes de forte affluence [1] sont par exemple associées à de fortes concentrations de passagers dans les trains, qui diminuent le confort des voyageurs, en augmentant leur stress et leur sentiment de fatigue et d'insécurité. Ces périodes de forte affluence sont aussi associées à une élévation des temps d'attente et de la variabilité des temps de voyage, causant en général une perception d'un réseau de transport peu fiable (Tirachini et al. 2013).

Assurer la performance et la qualité des services est donc essentiel à la bonne perception du réseau. Or, la répartition à quai des voyageurs, c'est-à-dire la façon dont ils se disposent le long du quai avant la montée dans le train, est un élément qui peut impacter fortement ce service, notamment au travers du temps de stationnement des trains à quai et de la charge à quai et à bord (Lam et al. 1999). Cette répartition est influencée par plusieurs facteurs dont la disposition des entrées de quai en gare d'origine, la disposition des sorties de quai en gare d'arrivée ou encore, la concentration de passagers à bord du train (Kim et al. 2014). Le premier facteur entraine souvent des concentrations importantes de passagers à proximité des entrées du quai, causant de la même façon des fortes concentrations à bord des voitures desservant ces zones (Peftitsi et al. 2020, Krstanoski 2014). Ces mêmes regroupements de passagers autour de zones spécifiques du quai peuvent se retrouver au travers du deuxième facteur, quand une part importante des passagers voyage vers la même gare d'arrivée et ainsi souhaite descendre à proximité d'une sortie de quai de cette gare d'arrivée (Fang et al. 2019). Cela peut mener à des difficultés d'échange entre descendants et montants (Seriani et al. 2019), et à terme à des temps de stationnement du train en gare rallongés et des retards (Cornet et al. 2019). Par ailleurs, une forte concentration à quai entraine également une forte concentration à bord qui pourrait être évitée si les personnes se répartissaient uniformément le long du quai. L'uniformisation de la répartition des passagers le long du quai a donc une double importance : éviter les fortes concentrations de voyageurs à quai et à bord d'une part, et augmenter la fiabilité des transports en limitant les retards d'autre part.

Heureusement, certains outils existent afin d'influencer à la fois le comportement des voyageurs, ainsi que leur perception du réseau. La communication en temps réel aux voyageurs d'informations sur la charge à bord de chaque voiture est par exemple considérée par Oliveira, Fox, Birrell & Cain (2019) comme l'un des moyens pour jouer sur la répartition à quai. De la même façon, Zhang et al. (2017) ont montré que l'information en

1. Heure de pointe du matin (6-9h) et du soir (17-20h)

temps réel de la charge à bord des trains avait un effet sur le comportement des voyageurs, dont le placement à quai était notamment déplacé vers des voitures moins chargées. Plus généralement, certains auteurs pensent que l'information voyageur fait partie des moyens disponibles pour jouer sur l'attractivité des transports en commun. Blainey et al. (2012) évoquent l'absence d'une information voyageur suffisante comme une des barrières à l'utilisation des transports en commun. Dans leur étude sur un panel d'utilisateurs, Oliveira, Bruen, Birrell & Cain (2019) ont montré que l'accès à une information en temps réel sur le déroulé du voyage et les perturbations du réseau était la deuxième technologie attendue par les voyageurs. De même, en plus de montrer l'influence de l'information en temps réel de la charge à bord des trains sur la répartition à quai, Zhang et al. (2017) ont montré que cette information était perçue positivement par les passagers. Ainsi, fournir de l'information aux voyageurs sur le réseau et encore plus précisément sur leur voyage, semble être une des solutions à envisager pour jouer sur l'attractivité des transports en commun.

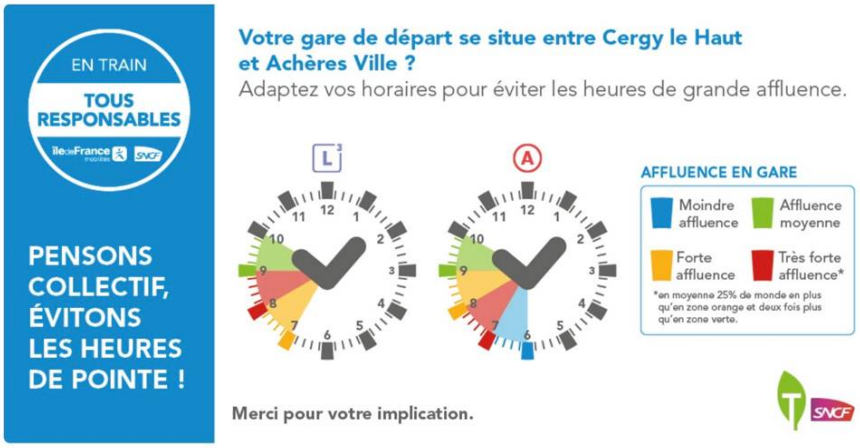
1.2 L'information voyageur à Transilien

Transilien propose différents outils à ses voyageurs, permettant notamment de les garder informés sur l'état du trafic, aux différentes étapes de leur voyage. Le site web Transilien.com [2] et l'application « l'Assistant SNCF » font partie de ces outils permettant au voyageur d'anticiper et de préparer son voyage, en offrant des fonctionnalités de calcul d'itinéraire ou d'accès à des fiches horaires personnalisées par trajet. Ces applications permettent aussi d'en savoir plus sur les travaux ou perturbations qui seront rencontrés lors du trajet. Ces informations sont souvent relayées en gare par des écrans présents à quai et dans les zones d'échange, qui fournissent de l'information sur les prochains trains et les perturbations en cours. Ces annonces visuelles sont couplées à des annonces sonores, et à l'intervention d'agents en cas de situations très perturbées. En plus de cela, chaque ligne de train [3] dispose d'un compte Twitter et d'un blog de ligne, qui permettent entre autres de diffuser de l'information sur le trafic et des conseils pour les voyageurs.

Des travaux passés ou en cours montrent la volonté de Transilien d'aller plus loin dans l'information fournie aux voyageurs, avec un accent mis sur l'accès à des données d'affluence dans les gares et les trains. C'est le cas par exemple du projet collaboratif Tranquilien [4] dont l'objectif était de donner à voir aux voyageurs l'affluence à bord des trains en leur laissant la possibilité de renseigner eux-mêmes la charge à bord de leur train. Malheureusement le faible taux d'utilisation de cette application a mené à son abandon. Le déconfinement a aussi été un moment clé de l'utilisation de ce type d'information, avec l'émergence sur plusieurs des lignes de train, d'indices de fréquentation des trains (voir Figure 1). A plus long terme, Transilien prévoit de rendre disponible sur les écrans à quai, une information en temps réel de charge à bord des trains arrivant en gare. Dans ce contexte, des données de charge à bord fiables et disponibles en temps réel sont nécessaires à la mise en place de ce type d'outils. En l'occurrence, Transilien dispose de certains systèmes de comptage qui permettent d'estimer le nombre de passagers à quai et à bord.

2. https ://www.transilien.com/
3. Voir la *Figure 4* pour une liste exhaustive des lignes de Transilien
4. https ://www.digital.sncf.com/store/applications/tranquilien-0

FIGURE 1 – Indicateurs d'affluence horaire dans certaines gares des lignes A et L entre mars et juin 2020



1.3 Le Comptage Automatique Voyageur Embarqué (CAVE)

Notions

Avant de décrire plus en détails les données disponibles à Transilien, quelques notions sur le matériel doivent être abordées pour la bonne compréhension du rapport. Un train est une association d'une ou plusieurs rames, chaque rame étant elle-même divisée en voitures (voir *Figure 2*). Un train est appelé unité simple (US) s'il est composé d'une seule rame, et unité multiple (UM) s'il est composé de plusieurs rames. Si les voitures d'une rame sont communicantes, c'est-à-dire si les passagers ont la possibilité de se déplacer librement à l'intérieur de la rame et de passer d'une voiture à l'autre, alors ces rames sont dites « BOA ».

FIGURE 2 – Rame Z50000 composée de 8 voitures

Nouveau matériel compteur

Depuis 2008, les trains du réseau Transilien se modernisent avec la mise en service d'un matériel plus récent, les rames Z50000, aussi appelées « Nouvelles Automotrice Transilien »(NAT). Les rames Z50000 sont composées de 8 voitures communicantes, chacune disposant de deux portes placées de chaque côté de la voiture. Ce nouveau matériel a la particularité de disposer au-dessus de chaque porte de capteurs infra-rouges qui comptent le nombre de passagers montant et descendant de la voiture (*Figure 3*). Le comptage commence une fois les portes ouvertes, et se termine dès que le train démarre. Les rames compteuses se généralisent peu à peu à l'ensemble du réseau Transilien et équipent déjà entièrement certaines lignes, dont la H et la R (voir *Figure 4*).

FIGURE 3 – Capteurs infra-rouges au-dessus d'une porte de train de type Z50000

FIGURE 4 – Taux d'équipement en rames compteuses des différentes lignes du réseau Transilien

Taux d'équipement	A	B	C	D	E	H	J	K	L	N	P	R	U	T4	T Express 11
2019	0%	0%	26%	0%	16%	100%	61%	100%	79%	28%	52%	100%	33%	50%	100%
2023	0%	100%	13%	100%	100%	100%	100%	100%	100%	100%	71%	100%	0%	100%	100%

Accessibilité

Les données de comptage des montées et descentes sont accessibles à J+2 sur la plateforme Châtelet, après un ensemble de traitements permettant la mise en qualité de ces données. Parmi les modifications effectuées, on trouve notamment :

- Le rapprochement des données de comptage au plan de transport réalisé de la journée, c'est-à-dire à l'ensemble des trains ayant en effet circulé le jour considéré ;
- L'ajustement des données de façon à ce que la somme des montées soit égale à la somme des descentes à la fin du trajet d'un train i.e, de façon à ce que le train soit vide à son terminus.

Pendant ce processus, certaines données sont écartées notamment si le rapprochement au plan de transport n'aboutit pas ou si les données de comptages sont jugées anormales (e.g, comptages dépassant la capacité d'accueil du train, déséquilibre trop important des données de comptage au terminus). En 2017, le taux de couverture, c'est-à-dire, le pourcentage de trains pour lesquels les données de comptage sont disponibles par rapport aux trains ayant circulé, était de 85.9% sur la ligne H (Coulaud 2019).

Face à la situation de crise sanitaire survenue pendant le premier semestre 2020, la demande d'accès à ces données de comptage en temps réel a été accrue, notamment afin de contrôler la fréquentation des trains. Dans ce contexte, le processus de collecte des données a été internalisé et ces données sont maintenant accessibles en temps réel sur le réseau interne, grâce à une API.

Contenu des données

Les données mises à disposition sur la plateforme Chatelet à J+2 sont regroupées dans le *Tableau 1*. En comptant les personnes qui montent et descendent du train à chaque

gare désservie, les capteurs nous donnent la possibilité d'estimer le nombre de passagers présents à bord du train, appelée charge à bord par la suite.

Plus précisément encore, nous avons accès à cette information de comptage à l'échelle de la voiture. Or, comme évoqué précédemment, les rames équipées de capteurs infra-rouges sont des rames « BOA ». Ainsi, une personne qui est montée dans une voiture, peut tout à fait se déplacer à l'intérieur de la rame et descendre à une autre voiture. Ces mouvements ne sont pas pris en compte par les capteurs infra-rouges. Il en résulte que le nombre de montants dans une voiture comptés par les capteurs, n'est pas forcément représentatif des personnes effectivement restées dans la voiture. C'est une limite de ce type de comptage à laquelle il faudra prêter attention en cas d'estimation des charges à bord des voitures.

TABLE 1 – Description des variables à disposition dans les données CAVE

Nom	**Contenu**
Jour	7 niveaux : lun., ..., dim.
Date de comptage	Explicite
Code CI	Code de la gare
Gare	Nom de la gare de la mesure
N de l'arrêt	Numéro de l'arrêt dans le trajet. Incrémenté de 1 en 1 dans l'ordre des gares sur le trajet
Horaire Réel	%H :%M :%S horaire de passage dans la gare mesuré par le train
Horaire Théorique	%H :%M :%S horaire de passage dans la gare prévu par le plan de transport
Durée Réelle	%H :%M :%S temps de stationnement mesuré par le matériel
Durée Théorique	%H :%M :%S temps de stationnement prévu par le plan de transport
Temps d'échange	Intervalle de temps entre l'ouverture et la fermeture des portes
Mission	Code de la mission (dépend de la desserte)
Lgn	Code de la ligne
Origine/destination	Explicite
Train	Numéro du train (code unique par jour)
Comp	Composition du train : US, UM
Caractéristique des rames	Type de série de la rame
Cap Assis	Capacité en places assises
Cap totale	Capacité totale
Montées	Nombre de montées dans le train
Descentes	Nombre de descentes du train
Charge	Charge à bord du train en sortie de gare
Occ assise	Taux d'occupation assise : (Charge / capacité assise)
Occ tot	Taux d'occupation totale : (Charge / capacité totale)
Pourcent_Cpt	% de comptage. Si 50 % observé pour une UM, alors une seule rame renvoie des données
I	Service irrégulier : train supprimé, train en retard, mission changée...
Num_fiable	Indicateur de la fiabilité de la numérotation des voitures/portes
1_01D1_Montées	Nombre de montées pour la porte droite 1 de la voiture 1 de la rame 1

...	7 autres voitures
2_02D1_Descentes	Nombre de descentes pour la porte droite 1 de la voiture 2 de la rame 2
...	7 autres voitures

1.4 Objectifs

Transilien est donc un acteur du transport de personnes qui a la chance de voir sur son réseau la généralisation d'un nouveau matériel compteur. Celui-ci a la capacité de compter à chaque porte le nombre de passagers montant et descendant du train et ce, à chaque fois que le train dessert une gare. La demande croissante d'information de fréquentation liée au contexte sanitaire nous a permis d'avoir accès à ces données en temps réel, et a encouragé toutes les démarches de mise à disposition des voyageurs d'information sur la charge des trains. Cela nous a permis de mettre en place une expérimentation visant à étudier l'effet de cette information de charge sur la répartition des voyageurs à quai et à bord.

Nous présenterons d'abord la réflexion menée sur la modélisation du mouvement des voyageurs au sein de la rame, dans un but de mise en qualité des données de comptage. Nous décrirons dans un deuxième temps la création de l'application Hector, ayant pour vocation de mettre à disposition des voyageurs en temps réel la charge des trains arrivant à leur gare.

2 Modélisation du déplacement des passagers à l'intérieur du train

Comme évoqué précédemment, le déplacement libre des passagers dans la rame n'est pas pris en compte dans les comptages renvoyés par les capteurs infra-rouges. On se propose donc d'élaborer un modèle de propagation des passagers à bord, permettant de rendre compte au mieux de ces déplacements. Idéalement, cela nous permettra d'accéder à une information de comptage améliorée qui pourra être utilisée à terme dans l'application Hector, dédiée à la visualisation en temps réel par les passagers de la charge à bord des voitures de leurs trains.

2.1 Méthode

Peu de littérature existe sur le déplacement des passagers à l'intérieur d'une rame de train. Au contraire, beaucoup d'auteurs réfléchissent sur des modèles de mouvement de foule, dans les gares par exemple. C'est le cas notamment de Krstanoski (2014) qui a essayé de modéliser le placement à quai des voyageurs, en faisant l'hypothèse que le nombre de montants dans une voiture suit une loi multinomiale. Il arrive à la conclusion qu'un bon estimateur de la probabilité de monter dans une voiture est le rapport entre le nombre de montées dans la voiture et le nombre de montées dans le train. En dehors du transport, des travaux existent sur les mouvements spatiaux et temporels des personnes, par exemple ceux d'employés dans des bureaux pendant les journées de travail (Shelat et al. 2020, Wang et al. 2011). Ces travaux s'appuient sur les chaînes de Markov, et s'intéressent notamment à l'estimation des probabilités de transition d'un état A (e.g, être à son bureau) à un état B (e.g, être dans la cantine). Cette estimation fait echo à notre problème, puisqu'on aimerait estimer le nombre de personnes qui montent dans une voiture v et se déplacent jusqu'à une voiture v', qu'on pourrait écrire comme étant la transition de la voiture v à la voiture v'. Les chaines de Markov ne s'appliquent pas complètement à notre problème, puisque nous ne prenons pas en compte la temporalité du mouvement, mais la notion de matrice de transition est très similaire à ce qu'on présentera par la suite. Par ailleurs, Grimshaw & Alexander (2011) ont essayé d'estimer les matrices de transition de chaines de Markov à l'aide d'une loi multinomiale, ce qui se rapproche de la méthodologie que nous allons présenter dans la suite du rapport.

2.1.1 Attentes

Soient m,d et $c \in \mathbb{N}$, respectivement le nombre de montées, le nombre de descentes et la charge à bord de la rame. On note $\mathbb{G}_k = \{1, \dots, g, \dots, G_k\}$, l'ensemble des gares désservies pendant le trajet $k \in \mathbb{N}$, avec G_k la dernière gare désservie, aussi appelée terminus. Soit une rame composée de V voitures, on définit $\mathbb{V} = \{1, \dots, v, \dots, V\}$, l'ensemble de ces voitures. On peut alors exprimer la charge à bord de la voiture v en sortie de la gare G comme suit :

$$\forall v \in \mathbb{V}, \quad c_{v,G,k} = c_{v,G-1,k} + m_{v,G,k} - d_{v,G,k} = \sum_{g=1}^{G} (m_{v,g,k} - d_{v,g,k})$$

S'il n'y avait aucun déplacement dans la rame, alors on devrait observer trois choses :

Propriété 1 *La charge par voiture est toujours positive ou nulle en cours de trajet, s'il n'y a pas de déplacement dans la rame.*

$$c_{v,G,k} = \sum_{g=1}^{G} (m_{v,g,k} - d_{v,g,k}) \geq 0$$

Propriété 2 *La charge par voiture est toujours nulle au terminus, s'il n'y a pas de déplacement dans la rame.*

$$\sum_{g \in \mathbb{G}_k} (m_{v,g,k} - d_{v,g,k}) = 0$$

Propriété 3 *La charge par voiture ne peut pas excéder 120% de la capacité de la voiture.* [5]

$$c_{v,G,k} = \sum_{g=1}^{G} (m_{v,gk} - d_{v,g,k}) \leq 1.2 * cap$$

Le modèle élaboré dans la prochaine partie vise à ce que les données de comptage respectent au mieux ces trois propriétés.

2.1.2 Formalisation du problème

Le nombre de montants m dans une rame peut s'exprimer $m = \sum_{v=1}^{V} m_v$, où m_v est le nombre de montants observés en voiture v. Le nombre de montants en voiture v peut aussi s'écrire de cette façon : $m_v = \sum_{v'=1}^{V} m_v^{v'}$ avec $m_v^{v'}$, le nombre de personnes s'arrêtant dans la voiture v' sachant qu'elles viennent de la voiture v.

Notons M_v la variable aléatoire telle que $M_v = (M_v^1, ..., M_v^V)$. Alors, $\forall v, v' \in \mathbb{V}$, on peut écrire :

$$M_v \sim \mathcal{M}(m_v; p_v^1, ..., p_v^V) \quad avec \quad \sum_{v'=1}^{V} p_v^{v'} = 1 \quad et \quad \sum_{v'=1}^{V} m_v^{v'} = m_v \tag{1}$$

On appellera par la suite probabilité de transition de v vers v', la probabilité de monter à la voiture v et de s'arrêter à la voiture v', $p_v^{v'}$. L'ensemble des probabilités de transition d'une voiture vers l'autre sont rassemblées au sein d'une matrice de transition notée :

$$P = \begin{pmatrix} p_1^1 & \cdots & p_1^v & \cdots & p_1^V \\ \vdots & \ddots & & & \vdots \\ p_v^1 & & p_v^v & & \vdots \\ \vdots & & & \ddots & \vdots \\ p_V^1 & \cdots & \cdots & \cdots & p_V^V \end{pmatrix}$$

A partir de l'équation (1), on exprime désormais a_v, la quantité de personnes s'étant arrêtées en voiture v comme suit : $a_v = \sum_{v'=1}^{V} m_{v'}^v$.

5. Limite admise à la SNCF, basée sur le nombre de personnes par m^2 pouvant occuper la voiture

Estimation par maximum de vraissemblance

Comme on ne peut pas observer $m_{v'}^{v}$ à cause des contraintes techniques liées aux comptages, on souhaite trouver un estimateur de cette quantité. On se propose d'utiliser la méthode du maximum de vraissemblance. On rappelle que pour toute voiture $v \in \mathbb{V}$, comme M_v suit une loi multinomiale, la probabilité d'observer m_v^1 personnes montant en voiture v et descendant en voiture 1, m_v^2 personnes montant en voiture v et descendant en voiture 2, ..., sachant qu'il y a m_v montants en voiture v s'écrit :

$$P(M_v^1 = m_v^1, \dots, M_v^V = m_v^V | M_v = m_v) = \frac{m_v!}{m_v^1! \dots m_v^V!} (p_v^1)^{m_v^1} \dots (p_v^V)^{m_v^V}$$

Pour tout trajet $k \in \mathbb{N}$, la vraissemblance de M_v peut alors s'écrire :

$$V(M_v) = \prod_{k=1}^{n} P(M_v^1 = m_{v,k}^1, \dots, M_v^V = m_{v,k}^V | M_v = m_{v,k}) = \prod_{k=1}^{n} \frac{m_{v,k}!}{m_{v,k}^1! \dots m_{v,k}^V!} (p_v^1)^{m_{v,k}^1} \dots (p_v^V)^{m_{v,k}^V}$$

Par simplicité, on écrit la log vraissemblance de M_v :

$$\begin{aligned}
\ln(V(M_v)) &= \sum_{k=1}^{n} \ln\left(\frac{m_{v,k}!}{m_{v,k}^1! \dots m_{v,k}^V!} (p_v^1)^{m_{v,k}^1} \dots (p_v^V)^{m_{v,k}^V} \right) \\
&= \sum_{k=1}^{n} \ln(m_{v,k}!) - \ln(m_{v,k}^1! \dots m_{v,k}^V!) + m_{v,k}^1 \ln(p_v^1) + \dots + m_{v,k}^V \ln(p_v^V)
\end{aligned}$$

On intègre la contrainte $\sum_{v'=1}^{V} p_v^{v'} = 1$ dans l'écriture de la log vraissemblance :

$$\ln(V(M_v)) = \sum_{k=1}^{n} \ln(m_{v,k}!) - \ln(m_{v,k}^1! \dots m_{v,k}^V!) + m_{v,k}^1 \ln(p_v^1) + \dots + m_{v,k}^V \ln\left(1 - \sum_{v'=1}^{V-1} p_v^{v'}\right)$$

La vraissemblance est maximale quand sa dérivée première par rapport aux paramètres $p_v^{v'}$ s'annule et que sa dérivée seconde est négative. Le calcul des dérivées d'ordre 1 donne :

$$\begin{aligned}
\frac{\partial \ln(V(M_v))}{\partial p_v^{v'}} = 0 &\Leftrightarrow \sum_{k=1}^{n} \left(\frac{m_{v,k}^V}{p_v^{v'}} - \frac{m_{v,k}^{v'}}{1 - \sum_{v'=1}^{V-1} p_v^{v'}} \right) = 0 \\
&\Leftrightarrow \sum_{k=1}^{n} \frac{m_{v,k}^{v'}}{p_v^{v'}} = \sum_{k=1}^{n} \frac{m_{v,k}^V}{p_v^V} \\
&\Leftrightarrow p_v^{v\prime} = p_v^V \frac{\sum_{k=1}^{n} m_{v,k}^{v'}}{\sum_{k=1}^{n} m_{v,k}^V}
\end{aligned}$$

En utilisant encore une fois la contrainte $\sum_{v'=1}^{V} p_v^{v'} = 1$, on obtient :

$$\sum_{v'=1}^{V} p_v^V \frac{\sum_{k=1}^{n} m_{v,k}^{v'}}{\sum_{k=1}^{n} m_{v,k}^V} = 1 \Leftrightarrow p_v^V = \frac{\sum_{k=1}^{n} m_{v,k}^V}{\sum_{k=1}^{n} \sum_{v'=1}^{V} m_{v,k}^{v'}} \Leftrightarrow p_v^V = \frac{\sum_{k=1}^{n} m_{v,k}^V}{\sum_{k=1}^{n} m_{v,k}}$$

Cette contrainte a été introduite ici pour remplacer p_v^V mais nous aurions pu imaginer l'introduire pour n'importe quelle voiture $v' \in \mathbb{V}$. Ainsi, on trouve :

$$\forall v, v' \in \mathbb{V}, \quad \sum_{k=1}^{n} m_{v,k}^{v'} = p_v^{v'} \sum_{k=1}^{n} m_{v,k} \tag{2}$$

Fonction objective

On note $d_{v,k} = \sum_{g \in \mathbb{G}_k} d_{v,g,k}$, la somme cumulée des descentes en voiture v au cours d'un trajet k et $a_{v,k} = \sum_{g \in \mathbb{G}_k} a_{v,g,k}$, la somme cumulée des personnes s'étant arrêtées en voiture v au cours d'un trajet k. On peut donc définir la charge de la voiture v au terminus du trajet k comme suit :

$$c_{v,k} = a_{v,k} - d_{v,k} = \sum_{g \in \mathbb{G}_k} (a_{v,g,k} - d_{v,g,k})$$

On cherche à estimer les paramètres $p_v^{v'}$ exprimés dans l'équation (1), avec $v \in \mathbb{V}$ la voiture d'origine, et $v' \in \mathbb{V}$ la voiture d'arrêt. Pour rappel, la propriété 2 impose que, pour n'importe quel trajet $k \in \mathbb{N}$ et pour n'importe quelle voiture $v \in \mathbb{V}$, et dans le cas d'une absence de déplacement au sein de la rame, la charge au terminus devrait être nulle. On reformule ainsi cette propriété en autorisant les déplacements à la montée mais pas à la descente. On aimerait donc trouver les paramètres $p_v^{v'}$ qui permettent de minimiser l'écart au carré entre la somme cumulée des personnes arrêtées en voiture v et la somme cumulée des personnes descendues de la voiture v au cours du trajet k, c'est-à-dire qui minimisent : $\sum_{k=1}^{n} (a_{v,k} - d_{v,k})^2$.

Or, d'après l'équation (2), on peut expliciter l'écriture de ce problème puisque :

$$\sum_{k=1}^{n} a_{v,k} = \sum_{k=1}^{n} \sum_{v'=1}^{V} m_{v',k}^{v} = \sum_{k=1}^{n} \sum_{v'=1}^{V} p_{v'}^{v} m_{v',k}$$

On cherche donc à trouver les paramètres qui vérifient :

$$\begin{aligned} \min_{\forall v,v' \in \mathbb{V},\ p_{v'}^{v}} \quad & \sum_{k=1}^{n} \left(\sum_{v'=1}^{V} p_{v'}^{v} m_{v',k} - d_{v,k} \right)^2 \\ \text{s.c.} \quad & \forall v \in \mathbb{V}, \quad \sum_{v'=1}^{V} p_v^{v'} = 1 \\ & \forall v, v' \in \mathbb{V}, \quad 0 \leq p_{v'}^{v} \leq 1 \end{aligned} \tag{3}$$

Pour simplifier l'écriture du problème d'optimisation, on note $p^v = (p_1^v, ..., p_V^v)$, l'ensemble des probabilités de se déplacer à la voiture v, et $m_k = (m_{1,k}, ..., m_{V,k})$, avec m'_k, la transposée de m_k, ce qui donne $\forall v \in \mathbb{V}$:

$$\min_{\forall v,v' \in \mathbb{V},\ p_{v'}^{v}} \sum_{k=1}^{n} (p^v m'_k - d_{v,k})^2 = \min_{\forall v,v' \in \mathbb{V},\ p_{v'}^{v}} \sum_{k=1}^{n} f(p^v, m_k, d_{v,k}) \tag{4}$$

2.1.3 Hypothèses de déplacement

Pour modéliser le comportement des passagers dans la rame, on fait deux hypothèses sur le comportement des passagers.

Déplacement libre

On fait l'hypothèse que les passagers se déplacent sans contraintes dans l'ensemble de la rame. Dans le cas des rames Z50000, composées de 8 voitures, il y aurait donc $8 \times 8 = 64$ paramètres à estimer.

Déplacement avec une contrainte de proximité à la voiture d'origine
On fait l'hypothèse que les passagers choisissent majoritairement leur voiture de montée proche de la voiture à laquelle ils veulent s'arrêter (i.e, rester). C'est une hypothèse cohérente avec tout une littérature sur l'influence des sorties de quais de la gare d'arrivée sur le choix des voitures (Kim et al. 2014, Fang et al. 2019, Peftitsi et al. 2020). Les passagers auraient donc tendance à progresser le long du quai pour se placer à proximité de leur voiture d'intérêt, plutôt que de progresser à l'intérieur même de la rame.

$\forall v \in \mathbb{V}$, on note p_v^v la probabilité de rester dans la voiture dans laquelle on est monté, et $\alpha_v \in [0,1]$, un facteur de progression qui rend improbable un arrêt à une voiture trop éloignée de la voiture de montée. Alors, $\forall v' \in \mathbb{V}$, on décide d'exprimer $p_v^{v'}$ comme suit :

$$p_v^{v'} = p_v^v \times \alpha_v^{|v'-v|}$$

Il n'y a donc plus que 16 paramètres à estimer : les 8 probabilités de rester dans sa voiture de montée, et les 8 facteurs de progression. Le problème d'optimisation devient dans ce cas $\forall v \in \mathbb{V}$:

$$\begin{aligned} \min_{\forall v' \in \mathbb{V},\ (p_{v'}^{v'}, \alpha_{v'})} \quad & \sum_{k=1}^{n} \left(\sum_{v'=1}^{V} p_{v'}^{v'} \alpha_{v'}^{|v-v'|} m_{v',k} - d_{v,k} \right)^2 \\ \text{s.c.} \quad & \forall v' \in \mathbb{V}, \quad \sum_{v=1}^{V} p_{v'}^{v'} \alpha_{v'}^{|v-v'|} = 1 \\ & \forall v' \in \mathbb{V}, \quad 0 \leq p_{v'}^{v'} \leq 1 \\ & \forall v' \in \mathbb{V}, \quad 0 \leq \alpha_{v'} \leq 1 \end{aligned} \tag{5}$$

2.1.4 Solution analytique du problème

On aimerait montrer que notre problème admet une solution unique, c'est-à-dire identifier un minimum global. Pour ce faire, on propose un ensemble de propriétés et de définitions qui permettront de suivre notre raisonnement :

Propriété 4 *Soit f une fonction convexe sur un intervalle ouvert I. Si x_0 est un point critique pour f, alors f présente en x_0 un minimum global sur I.*

Definition 1 *Soit $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, une fonction deux fois dérivable sur $\mathbb{R}^n$. La matrice hessienne de la fonction numérique f est la matrice carrée, notée $H(f)$, de ses dérivées partielles secondes.*
$\forall i, j \in \{1, \ldots, n\}$, $(x_i, x_j) \in \mathbb{R}^2$, on note $H_{ij}(f) = \frac{\partial^2 f}{\partial x_i \partial x_j}$.

Definition 2 *Une matrice réelle symétrique M d'ordre n est semi-définie positive si elle vérifie la propriété : $\forall x = (x_1, \ldots, x_n) \in \mathbb{R}^n, xMx' \geq 0$.*

Propriété 5 *Soit f une fonction deux fois dérivable sur un sous-ensemble ouvert I de $\mathbb{R}^n$. Alors, f est une fonction convexe sur I si et seulement la matrice hessienne de f est semi-définie positive pour tout x de I.*

Au regard de la propriété 4, on cherche donc à montrer que le problème est convexe. Pour cela, on se propose de montrer que la matrice hessienne associée à la fonction objective, est semi-définie positive.

Déplacement libre

On cherche à minimiser la fonction objective explicitée dans l'équation 4. Pour montrer que la hessienne associée à cette fonction est semi-définie positive, on calcule dans un premier temps le gradient, c'est-à-dire les dérivées partielles d'ordre de 1 de $f(p^v, m_k, d_{v,k})$:

$$\begin{cases} \frac{\partial f(p^v, m_k, d_{v,k})}{\partial p_1^v} = 2m_{1,k}(p^v m'_k - d_{v,k}) \\ ... \\ \frac{\partial f(p^v, m_k, d_{v,k})}{\partial p_V^v} = 2m_{V,k}(p^v m'_k - d_{v,k}) \end{cases}$$

De ce gradient, on en déduit ensuite la matrice hessienne :

$$H = \begin{pmatrix} \frac{\partial^2 f(p^v, m_k, d_{v,k})}{\partial p_1^v \partial p_1^v} & \cdots & \frac{\partial^2 f(p^v, m_k, d_{v,k})}{\partial p_1^v \partial p_V^v} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(p^v, m_k, d_{v,k})}{\partial p_V^v \partial p_1^v} & \cdots & \frac{\partial^2 f(p^v, m_k, d_{v,k})}{\partial p_V^v \partial p_V^v} \end{pmatrix} = \begin{pmatrix} m_{1,k}^2 & \cdots & m_{1,k}m_{V,k} \\ \vdots & \ddots & \vdots \\ m_{V,k}m_{1,k} & \cdots & m_{V,k}^2 \end{pmatrix}$$

Pour montrer que la matrice hessienne est semi-définie positive, nous utilisons la définition 2 :

$$\begin{aligned} xHx^t &= \begin{pmatrix} 2(x_1 m_{1,k}^2 + \cdots + x_V m_{V,k}m_{1,k}) & \cdots & 2(x_1 m_{V,k} + \cdots + x_V m_{V,k}^2) \end{pmatrix} x' \\ &= 2\begin{pmatrix} m_{1,k}\sum_{i=1}^V x_i m_{i,k} & \cdots & m_{V,k}\sum_{i=1}^V x_i m_{i,k} \end{pmatrix} x' \\ &= 2(m_{1,k}x_1 \sum_{i=1}^V x_i m_{i,k} + \cdots + m_{V,k}x_V \sum_{i=1}^V x_i m_{i,k}) \\ &= 2(m_{1,k}x_1 + \cdots + m_{V,k}x_V) \sum_{i=1}^V x_i m_{i,k} \\ &= 2(\sum_{i=1}^V x_i m_{i,k})^2 \geq 0 \end{aligned}$$

La matrice hessienne associée à f est bien semi-définie positive et le problème est donc convexe et admet une solution globale. Cela est d'ailleurs cohérent avec l'hypothèse $p_v^{v'} = \frac{m_v^{v'}}{m_v}$ que l'on a faite, et qui ramène notre problème à une régression linéaire avec des contraintes sur $p_v^{v'}$.

Déplacement avec une contrainte de proximité à la voiture d'origine

Nous avons aussi voulu montrer la convexité du problème dans le cas contraint, mais n'avons cette fois pas réussi, après le calcul des dérivées secondes, à montrer que la hessienne était semi-définie positive (voir Annexe A). Nous avons quand même réalisé l'optimisation, en prenant des précautions supplémentaires dans l'étude de la convergence de la fonction objective par la suite, notamment en regardant l'impact des paramètres initiaux sur la solution trouvée.

2.1.5 Solution numérique du problème

Pour réaliser l'optimisation, nous avons utilisé la fonction *solnp* du package **Rsolnp** (Ghalanos & Theussl 2015) qui permet de faire de l'optimisation non-linéaire sous contraintes

d'égalité et d'inégalité. Pour utiliser cette fonction, nous avons eu besoin d'identifier les paramètres à estimer, de renseigner la fonction objective, les données sur lesquelles réaliser l'optimisation mais aussi l'ensemble des contraintes associées au problème. Des valeurs initiales pour les paramètres, qui respectent les contraintes imposées, sont requises pour pouvoir utiliser la fonction.

Dans le cas du modèle libre, nous avons fixé les valeurs initiales des paramètres à $1/8 = 0.125$. Pour fournir l'initialisation des paramètres dans le cas du modèle contraint, nous avons construit une fonction qui tire aléatoirement 8 valeurs p_v^v et 8 valeurs α_v, respectivement entre 1 et 500, et 1 et 750, jusqu'à ce que :

$$\forall v, v' \in \mathbb{V} \left| 1 - \frac{1}{1000} \sum_{v'=1}^{V} p_v^v \alpha_v^{|v'-v|} \right| \leq 10^{-3}$$

En imposant une valeur maximale pour p_v^v et α_v, on ne balaye pas l'ensemble du domaine de définition. Cependant cette contrainte est nécessaire pour pouvoir trouver des paramètres initiaux valides dans un temps raisonnable.

Nous avons étudié la convergence du modèle contraint pour 10 valeurs différentes de paramètres initiaux et remarqué que la convergence ou non de l'algorithme dépendait des paramètres initiaux. Une des initialisations a en effet mené à une non convergence, probablement car un des paramètres s'est mis à tendre vers 0 pendant la procédure. Néanmoins, toutes les initialisations qui mènent à une convergence, aboutissent à des valeurs de fonction objective et de paramètres identiques. Pour gérer la non convergence liée à certaines, nous avons répété l'initialisation jusqu'à convergence de l'algorithme.

2.1.6 Critère de performance des modèles

Nous utilisons le critère d'erreur absolue moyenne (MAE) pour comparer la performance des modèles, notamment pour sa facilité d'interprétation. Ce critère est calculé pour le jeu de données test afin d'éviter le sur-ajustement. Soient $y \in \mathbb{R}^n$, un ensemble de valeurs observées et $\hat{y} \in \mathbb{R}^n$ un ensemble de valeurs prédites, alors :

$$MAE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

Dans notre cas, on s'intéresse au MAE obtenu à l'échelle de la voiture qui, pour tout trajet $k \in \mathbb{N}$ et pour toute voiture $v \in \mathbb{V}$, correspond à :

$$MAE(d, a) = \frac{1}{Vn} \sum_{k=1}^{n} \sum_{v=1}^{V} |d_{v,k} - a_{v,k}|$$

2.2 Cas d'étude

On s'intéresse aux trains circulant sur la ligne H, car c'est une ligne entièrement équipée de rames Z50000, donc de capteurs infra-rouges, depuis 2016. Le tracé géographique de cette ligne est proposé en *Figure 5*. Elle est composée de 50 gares et peut être séparée en cinq axes de circulation, appelés branches par la suite :

- Paris-Nord - Pontoise
- Paris-Nord - Persan-Beaumont via Valmondois
- Paris-Nord - Persan-Beaumont via Montsoult-Maffliers

- Paris-Nord - Luzarches
- Creil - Pontoise, dénommé par la suite Transversale

FIGURE 5 – Tracé de la ligne H

Les trains de la ligne H sont composés d'une ou deux rames, et sont notés respectivement US et UM2. On peut aussi les distinguer grâce à leur numéro commercial pair s'ils se dirigent vers Paris, et impair sinon. On utilise les données CAVE disponibles sur la plateforme Châtelet et présentées en section 1.3. Plus précisément, on utilisera un jeu de données d'entrainement du 1^{er} septembre 2018 au 31 août 2019 i.e, l'année scolaire 2018/2019, et un jeu de données test du 1^{er} septembre 2019 au 30 novembre 2019, constitués de l'ensemble des trains ayant circulé sur la ligne H durant ces périodes. Cela permet d'obtenir un ratio entrainement/test de 80/20 mais aussi d'éviter de prendre en compte le mois de décembre 2019 dans le jeu test, période où le trafic a été très perturbé en raison des grèves liées à la réforme des retraites. La répartition des trains par branche est similaire entre les deux jeux de données, comme le montre le *Tableau 2*.

TABLE 2 – Nombre de trains par branche pour les jeux de données d'entrainement et de test

	Données d'entrainement		Données test	
Pontoise	29656	(23%)	6743	(22%)
Luzarches	36368	(28%)	8823	(29%)
Persan-Beaumont via Valmondois	33010	(26%)	8316	(27%)
Persan-Beaumont via Montsoult-Maffliers	16061	(13%)	3815	(12%)
Transversale	12796	(10%)	3056	(10%)
Total	127891		30753	

Le jeu de données d'entraînement est celui utilisé pour illustrer le déplacement des passagers à l'intérieur de la rame, et pour estimer les paramètres présentés dans la section 2.1.3. Une fois ces paramètres fixés, le jeu test nous permettra de comparer les performances des modèles.

2.2.1 Illustration du problème

On aimerait savoir si les propriétés 1, 2 et 3 sont vérifiées pour les trains circulant sur la ligne H.

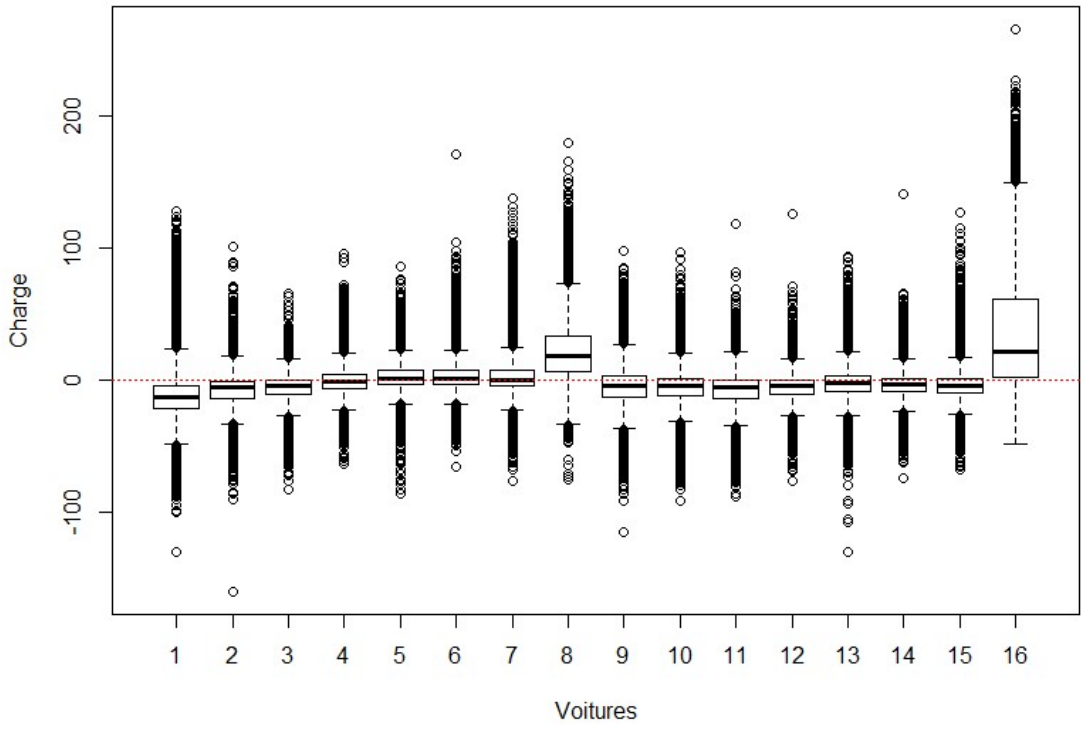
Charges négatives

Si on s'intéresse aux charges à bord des voitures en cours de trajet, c'est-à-dire sans prendre en compte le dernier arrêt d'un trajet, on observe de nombreuses occurences de charges négatives puisque 12% des charges observées sont négatives. Ce taux est également associé au fait que 72% des trains connaissent au moins un arrêt pour lequel la charge d'une voiture est négative.

Charges non nulles

Si on s'intéresse maintenant à la charge à bord des voitures au terminus on remarque que 92% des voitures ont une charge non nulle au terminus, dont 41% sont des charges positives et 59% sont des charges négatives. La *Figure 10* montre la distribution des charges par voiture au terminus, pour les unités multiples. La 8^{e} et la 16^{e} voiture présentent des charges fortement positives en fin de trajet, avec une médiane à 18 et 21. Parallèlement, la première voiture présente des charges plutôt négatives avec une médiane à -13.

FIGURE 6 – Distribution des charges par voiture en fin de trajet pour les unités multiples. La ligne en pointillé rouge matérialise une charge nulle.



Charges extrêmes

On cherche à mettre en évidence l'existence de charges extrêmes non réalistes au cours du trajet. On définit la capacité assise comme le nombre de places assises disponibles, et la capacité totale comme le nombre de places total (assises et debout). Les voitures de la ligne H contiennent chacune 59 places assises et 56 places debout, ce qui fait une capacité totale de 115 personnes. On s'intéresse donc aux trains présentant à au moins un arrêt, une voiture avec une charge supérieure à 120% de cette capacité, à savoir supérieure à 138 personnes. Cette situation concerne 10.5% des trains en cours de trajet. Ces trains ont la particularité d'être à 86.5 % des trains impairs ayant une charge extrême dès leur origine Paris Gare du Nord. La distribution de leur charge moyenne par voiture à cet arrêt est mise en évidence dans la *Figure 7*. Les fortes charges observées en voiture 16 s'expliquent notamment par la localisation de l'entrée du quai qui se trouve à l'extrêmité du train : cela favorise les montées en voiture 16 suivies d'une progression probable à l'intérieur de la rame, non prise en compte par les capteurs. Ce constat est en accord avec la littérature qui existe sur l'influence de la localisation des entrées de quai en gare de départ sur le choix des voitures de montée (Krstanoski 2014).

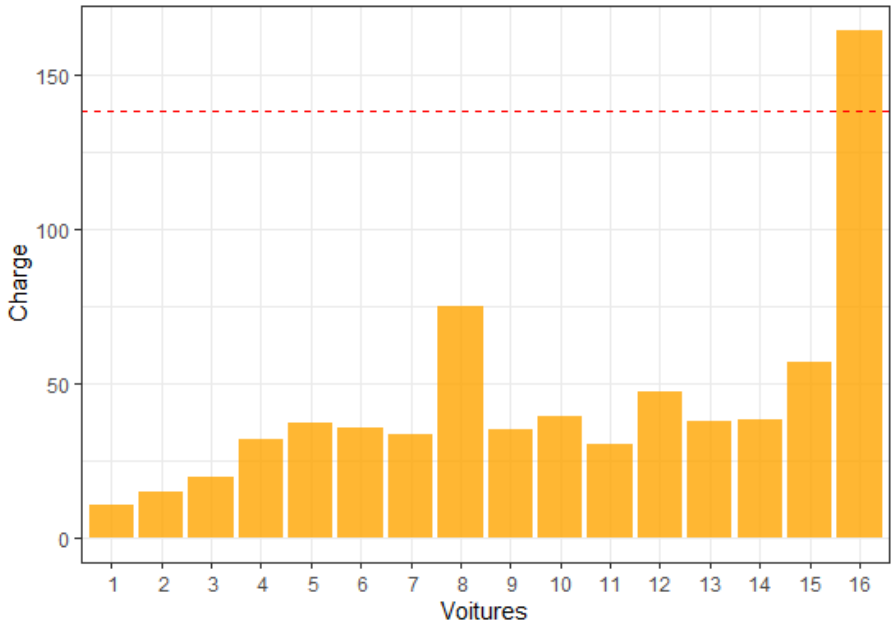


FIGURE 7 – Charge voiture moyenne à Paris Gare du Nord des trains UM2 présentant dès leur origine Paris Nord une charge extrême. La ligne rouge matérialise la limite de 120% de la capacité totale des voitures (138 places)

Conclusion

On a pu mettre en évidence le déplacement des voyageurs dans la rame, en faisant le constat aussi bien de charges extrêmes ou négatives non réalistes en cours de trajet, que de charges non nulles en fin de trajet. On se propose maintenant d'estimer les paramètres présentés dans la section précédente afin de voir si les modèles développés permettent d'améliorer l'information de comptage.

2.2.2 Echelle de calibration

En plus de faire une hypothèse sur le déplacement des passagers à l'intérieur de la rame (i.e, contraint ou libre), on se dit que le réseau pourrait abriter, à des mailles plus ou moins fines, des différences dans le comportement des voyageurs. Il est donc intéressant de calibrer le modèle, non pas globalement sur l'ensemble du jeu de données d'entraînement, mais localement. On pense notamment à deux échelles spatiales qu'il serait intéressant d'étudier :

- la branche sur laquelle circule le train. En effet, les trains de la ligne H se distinguent en fonction des 5 axes de circulation qu'ils peuvent emprunter [6]. Les gares d'origine ou de destination étant différentes d'une branche à l'autre, on pourrait penser que le comportement des voyageurs en termes de déplacement dans la rame est dépendant de la branche empruntée ;

- le sens du train. De la même façon que pour les branches, la disposition des entrées et des sorties de quai diffère entre les voies associées à des trains pairs et celles associées à des trains impairs. On pourrait donc penser que le déplacement des

6. voir section 2.2

voyageurs dans les rames est dépendant du sens de circulation.

Descendre à une échelle encore plus locale, qui croise l'information de branche et de sens (e.g, les trains pairs de la branche Pontoise) a aussi été considéré dans la calibration des modèles.

Par ailleurs, les trains de la ligne H peuvent être composés de deux rames, une rame de tête et une rame de queue, qui ne sont pas situées au même endroit relativement aux entrées et sorties de quai des gares. Le déplacement des voyageurs pourrait donc être dépendant de la rame dans laquelle ils se trouvent, d'où l'intérêt de s'intéresser à différentes unités : l'unité rame, où la calibration dépendrait de la position de la rame, et l'unité train, où les rames seraient prises en compte toutes positions confondues. Cela amène à une décomposition différente de l'ensemble des observations en fonction des 8 échelles de calibration (voir *Tableau 3*).

TABLE 3 – Nombre de sous-jeux de données et nombre minimum d'observations disponibles pour l'optimisation à différentes échelles

Unité	Echelle	Nombre de jeux de données	Plus petit nombre d'observations
Train	global	1	123602
	sens	2	61224
	branche	5	12048
	branche-sens	10	6008
Rame	global	2	75100
	sens	4	37537
	branche	10	7933
	branche-sens	20	3914

2.3 Résultats

2.3.1 Analyse de l'erreur

A l'échelle globale

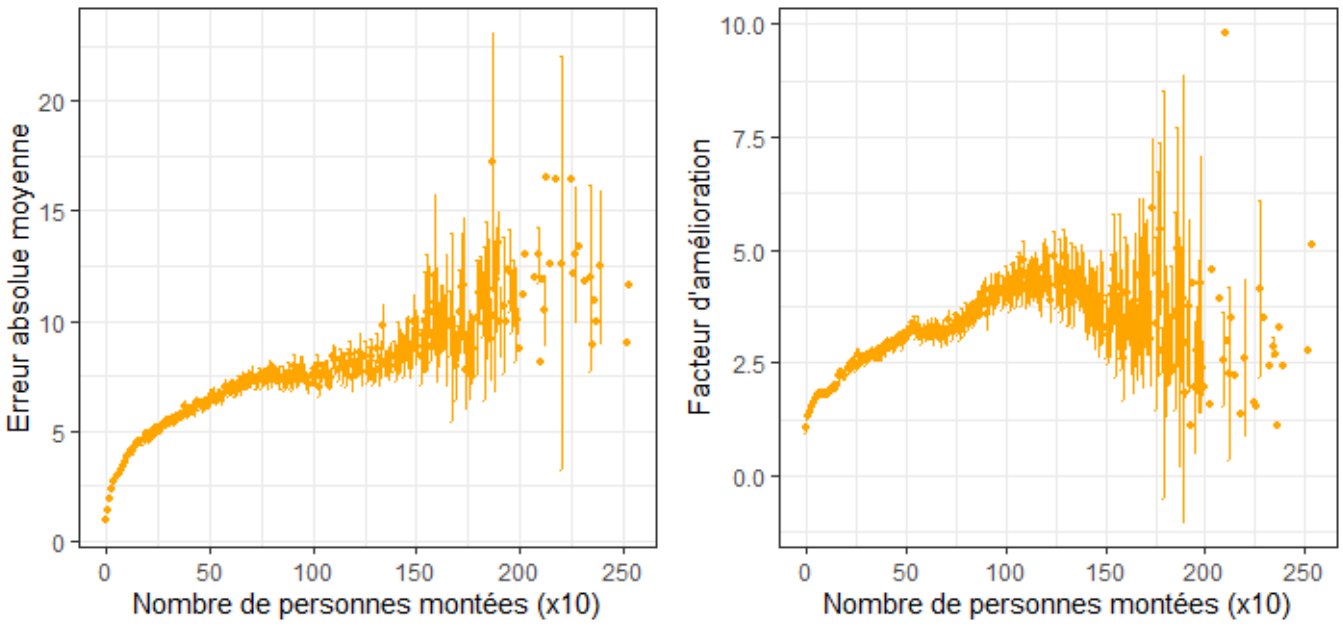
La MAE voiture obtenue sans modèle, et calculée pour l'ensemble du jeu de données test est de 16.8 personnes. On aimerait identifier le modèle qui permet de réduire au maximum cette erreur. Les performances des modèles sont regroupées dans le *Tableau 4*. La première chose à noter est que les modèles libres sont un à un meilleurs que ceux contraints. Cela peut s'expliquer par la contrainte imposée de proximité à la voiture qui est très forte, et ne laisse pas assez de liberté aux paramètres pour capter des spécificités. Par ailleurs les modèles calibrés à l'unité rame, sont également tous un à un meilleurs que ceux calibrés à l'unité train. C'est d'ailleurs le modèle libre calibré séparément sur chacune des rames de tête ou de queue, et à l'échelle branche et sens (ci-après appelé « modèle libre RBS ») qui permet d'obtenir la meilleure MAE de 5.9 personnes, soit une erreur près de 3 fois inférieure à celle observée dans la réalité. Ce modèle fait presque aussi bien que le modèle calibré uniquement par branche, avec une différence de MAE de seulement 0.1. Ainsi, le comportement des voyageurs semble en effet dépendant de la rame dans laquelle ils se trouvent, mais aussi de l'information spatiale, surtout de la branche sur laquelle ils circulent.

TABLE 4 – Performances en termes de MAE des modèles libres ou contraints pour les 8 différentes échelles

Unité	Echelle	MAE - modèle contraint	MAE - modèle libre
Train	Global	16.3	8.4
	Sens	17.1	8.3
	Branche	17.1	7.8
	Sens-Branche	17.4	7.7
Rame	Global	11.6	6.5
	Sens	12.1	6.4
	Branche	11.8	6.0
	Branche-Sens	11.2	**5.9**

On aimerait identifier les facteurs impactant les performances des modèles. On se dit notamment que la quantité de personnes montées pendant un trajet pourrait avoir un effet sur l'erreur commise. On introduit un nouvel indicateur, le facteur d'amélioration, correspondant au rapport entre l'erreur commise avec la méthode naïve et l'erreur commise avec le modèle. On s'intéresse ici au meilleur modèle, à savoir le « modèle libre RBS », et on regarde sur l'ensemble du jeu de données, l'évolution de l'erreur moyenne absolue et du facteur d'amélioration en fonction de la quantité de personnes montées pendant le trajet (voir *Figure 8*). On remarque d'abord que plus la fréquentation du trajet augmente, plus l'erreur augmente, avec un plateau atteint après une somme cumulée des montées de 750 personnes. Cela peut s'expliquer simplement par le fait que plus il y a de personnes à répartir dans le train, et plus on a de chances de se tromper. Parallèlement on remarque que, plus il y a de personnes à répartir, et plus on a de chance d'améliorer l'erreur par rapport à la méthode naïve, mais ce seulement jusqu'à un point critique autour de 1250 personnes.

FIGURE 8 – Evolution de l'erreur moyenne absolue (gauche) et du facteur d'amélioration (droite) du meilleur modèle en fonction de la somme cumulée des montées



A l'échelle locale

On se propose d'étudier ces performances à une échelle plus locale, en comparant les performances obtenues à l'échelle de chaque branche, pour la méthode naïve et pour le modèle libre RBS. Ces performances sont regroupées dans le *Tableau 5*.

TABLE 5 – Comparaison des MAE obtenues par branche pour la méthode naïve et le modèle libre RBS

Branche	Nombre moyen de montées sur le trajet	MAE naïf	MAE meilleur modèle	Facteur d'amélioration
Pontoise	673	13.8	6.6	2.09
Valmondois	650	14.7	6.5	**2.27**
Montsoult	479	12.8	6.0	2.13
Luzarches	378	9.4	5.0	1.87
Transversale	126	7.7	4.2	1.81

On remarque que la branche sur laquelle le modèle libre RBS améliore le plus la MAE est la branche Paris Nord - Persan Beaumont via Valmondois, avec une amélioration de 127%. Au contraire, l'amélioration de la MAE est la moins bonne pour la branche Transversale, d'environ 81%. Cela est cohérent avec l'observation précédente puisque la branche Transversale accueille très peu de passagers comparé à la branche Paris - Persan Beaumont via Valmondois, en moyenne 126 contre 650 personnes par trajet.

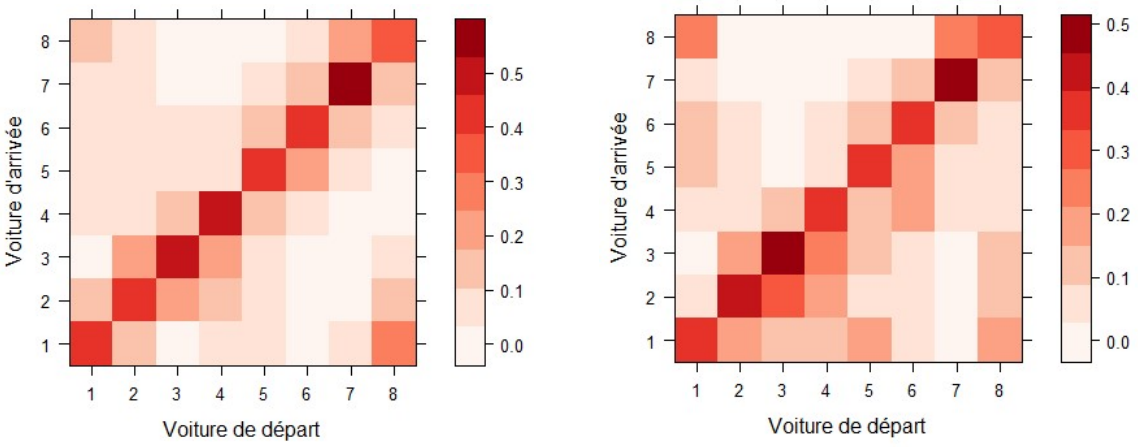
2.3.2 Analyse des matrices de transition

On se propose ici d'étudier les matrices de transition associées au meilleur modèle.

Variabilité par sens et rame

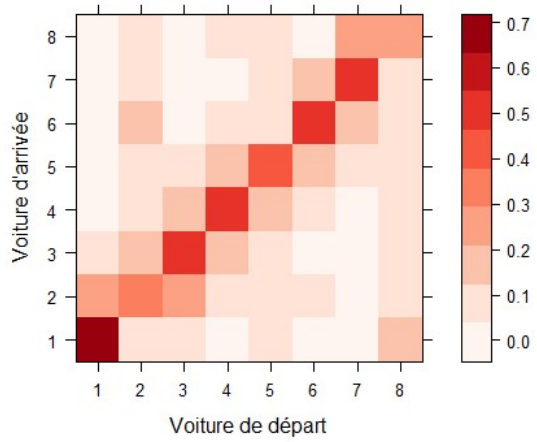
Pour étudier les différences entre les matrices de transition en fonction du sens et de la rame, on se propose d'étudier le cas de la branche Paris Nord - Persan Beaumont via Valmondois, car c'est la branche sur laquelle l'amélioration de l'erreur est la meilleure. Ces quatre matrices sont représentées dans la *Figure 9*. Indépendamment de la rame ou du sens dans lequel circulent les voyageurs, la voiture 8 est celle depuis laquelle les voyageurs se déplacent le plus. Cela paraît cohérent avec la disposition des entrées de quai dans les différentes gares, souvent situées à l'arrière ou au milieu du quai, ce qui encourage la montée dans la dernière voiture de l'une ou l'autre des rames, et la progression à l'intérieur de la rame. Cette disposition particulière des entrées de quai se retrouve notamment dans les gares de Paris Nord en sens impair, ou d'Ermont-Eaubonne et de Sarcelles-Saint-Brice en sens pair, qui sont des gares à forts flux de passagers.

FIGURE 9 – Matrices de transition pour la branche Paris Nord - Persan Beaumont via Valmondois

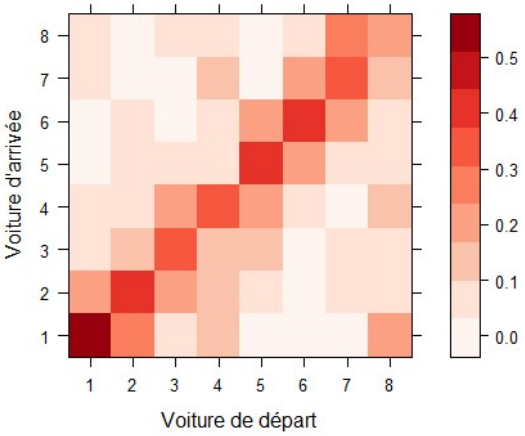


(a) Rame 1 - Pair

(b) Rame 1 - Impair



(c) Rame 2 - Pair



(d) Rame 2 - Impair

On remarque également que les voyageurs en direction de Paris tendent en moyenne à davantage rester dans la voiture dans laquelle ils sont montés, par rapport aux voyageurs circulant en direction de la Province. Par ailleurs, on peut mettre en évidence que chaque rame présente une voiture dans laquelle les voyageurs ont plus de chance de rester s'ils sont montés dedans : la voiture 7 pour la rame de tête, et la voiture 1 pour la rame de queue. Tous ces résultats sont regroupés dans le *Tableau 6*.

TABLE 6 – Probabilités de s'arrêter à la voiture de montée

	Rame 1		Rame 2	
	Pair	Impair	Pair	Impair
Voiture 1	0.41	0.37	**0.67**	**0.54**
Voiture 2	0.41	0.45	0.32	0.41
Voiture 3	0.46	0.46	0.47	0.36
Voiture 4	0.47	0.35	0.51	0.33
Voiture 5	0.44	0.39	0.40	0.42
Voiture 6	0.43	0.35	0.51	0.39
Voiture 7	**0.56**	**0.48**	0.50	0.36
Voiture 8	0.32	0.28	0.25	0.21
Moyenne	0.44	0.39	0.45	0.38

Variabilité par branche

On s'intéresse aux différences entre les matrices de transition des différentes branches. Pour cela, on étudie ces matrices à rame et sens fixé, à savoir la rame 1 en sens pair. En analysant la diagonale de ces matrices de transition, on peut mettre en évidence des comportements spécifiques à certaines branches : La branche Paris Nord - Pontoise, semble accueillir des passagers qui partent beaucoup plus de leur voiture de montée qu'à Paris Nord - Persan Beaumont via Montsoult par exemple, avec une probabilité moyenne de rester dans sa voiture de montée de 0.36 contre 0.47. Valmondois se distingue aussi des autres branches puisque la voiture de montée qui présente le plus haut pourcentage d'arrêt est située à l'arrière de la rame contrairement aux autres branches (voiture 7 vs. 1 ou 3).

TABLE 7 – Probabilités de s'arrêter à la voiture de montée

	Pontoise	Montsoult	Valmondois	Luzarches	Transversale
Voiture 1	**0.44**	**0.68**	0.41	0.39	**0.71**
Voiture 2	0.38	0.50	0.41	0.46	0.50
Voiture 3	0.36	0.44	0.46	**0.49**	0.43
Voiture 4	0.42	0.47	0.47	0.44	0.38
Voiture 5	0.42	0.46	0.44	0.40	0.22
Voiture 6	0.22	0.44	0.43	0.44	0.27
Voiture 7	0.24	0.44	**0.56**	0.40	0.30
Voiture 8	0.38	0.31	0.32	0.40	0.29
Moyenne	0.36	0.47	0.44	0.43	0.39

En sommant les colonnes de chaque matrice de transition, et en ôtant la valeur de la diagonale à cette somme, on peut étudier plus en détails les voitures vers lesquelles les passagers se sont le plus déplacés (voir *Tableau 8*). En l'occurrence, on remarque que pour la majorité des branches, la voiture la plus attractive se trouve en tête de rame (voiture 1 ou 3), sauf pour la branche Paris Nord - Pontoise pour laquelle la voiture 7 est la voiture la plus attractive. Ce premier résultat est cohérent avec le constat d'entrée de quai situées

en majorité à l'arrière du train et qui entrainerait donc une progression des passagers vers la tête de rame.

Pour aller plus loin dans l'analyse de ces résultats, il serait intéressant d'identifier les gares les plus fréqentées de chaque axe et d'étudier plus en détails la disposition des entrées de quai pour ces gares. Cela permettrait peut-être de pouvoir expliquer les différences constatées, notamment avec Pontoise.

TABLE 8 – Somme par colonne de la matrice de transition minorée de la diagonale

	Pontoise	Montsoult	Valmondois	Luzarches	Transversale
Voiture 1	0.66	0.55	**0.68**	**0.85**	**1.00**
Voiture 2	0.69	0.63	0.68	0.67	0.79
Voiture 3	0.66	**0.66**	0.59	0.58	0.63
Voiture 4	0.56	0.54	0.52	0.58	0.46
Voiture 5	0.59	0.55	0.54	0.57	0.49
Voiture 6	0.68	0.48	0.50	0.45	0.43
Voiture 7	**0.77**	0.46	0.46	0.50	0.45
Voiture 8	0.50	0.41	0.52	0.37	0.67

2.3.3 Comparaison d'autres indicateurs de performance

On aimerait que l'application du « modèle libre RBS »permette d'obtenir des valeurs de charge plus réalistes pendant et en fin de trajet. Pour cela, on peut comparer différents critères entre les données test brutes et les données test redressées grâce au modèle. Les *Tableaux 9 et 10* présentent ces critères respectivement en cours de trajet et en fin de trajet.

TABLE 9 – Comparaison entre données brutes et redressées des charges négatives et extrêmes observées en cours de trajet (propriété 2)

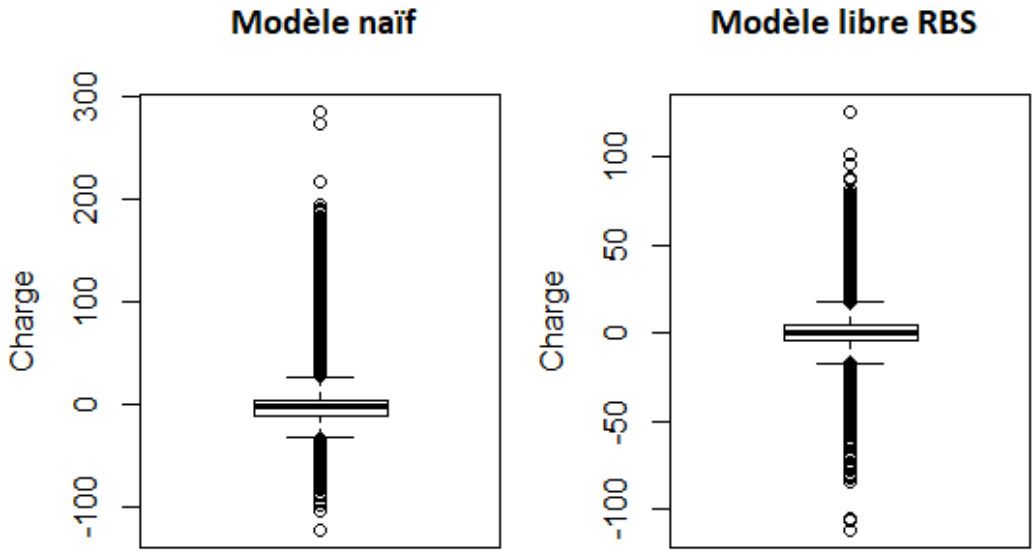
	Données Brutes	Données Redressées
Charges négatives	12%	5%
Trains à charges négatives	74%	61%
Nombre de charges extrêmes	16463	40

TABLE 10 – Comparaison entre données brutes et redressées des charges non nulles observées en fin de trajet (propriétés 1 et 3)

	Données Brutes	Données Redressées
Charges non nulles	95%	93%
dont positives	41%	52%
dont négatives	59%	48%

Un des forts effets du modèle sur ces critères, est la quasi disparition des charges extrêmes en cours de trajet avec un nombre de charges extrêmes qui passe de plusieurs milliers à quelques dizaines avec le redressement des données. On note également une diminution de plus de la moitié du nombre d'occurrences de charges négatives en cours de trajet et de 2% des charges non nulles en fin de trajet. La distribution des charges en fin de trajet mise en évidence dans la *Figure 10* montre leur tendance à se rapprocher de 0 avec la correction apportée par le modèle libre RBS. Ces résultats sont plutôt encourageants au regard des propriétés énoncées en section 2.1.1.

FIGURE 10 – Distribution des charges par voiture en fin de trajet pour la méthode naïve et le modèle libre RBS



2.4 Conclusion

Le modèle de propagation sans contraintes, paramétré sur chacune des rames, en différenciant branches et sens de circulation, est celui qui diminue le plus l'erreur faite sur la charge à bord. Ce modèle a surtout un fort effet sur les charges extrêmes, qui disparaissent quasiment après propagation des voyageurs. D'autres pistes pourraient être envisagées pour être encore plus précis dans la propagation des passagers, notamment considérer des spécificités de comportement par type de jour ou par tranche horaire. Ces hypothèses sont en effet assez classiques dans la littérature (Coulaud 2019, Toqué et al. 2017).

L'analyse des matrices de transition pour ce modèle montre que notre hypothèse d'un déplacement contraint autour des portes de montées est cohérente, même si le modèle contraint ne fait pas mieux que le modèle libre dans ce cas. On pourrait envisager d'ajouter des hypothèses supplémentaires, pour laisser un peu plus de liberté aux paramètres tout en contraignant le problème par rapport au cas libre. On pense notamment à la littérature qui existe sur l'attractivité de certaines voitures (Peftitsi et al. 2020, Krstanoski 2014), et à introduire un élément dans le modèle qui traduit cette attractivité.

Finalement, nous avons pu identifier un modèle qui permet d'améliorer l'information de charge extraite des données de comptage. Son utilisation dans un outil qui vise à rendre disponible une information fiable de charge à bord semble donc tout à fait indiquée.

3 Information voyageur de charge à Bord : la création du site web Hector

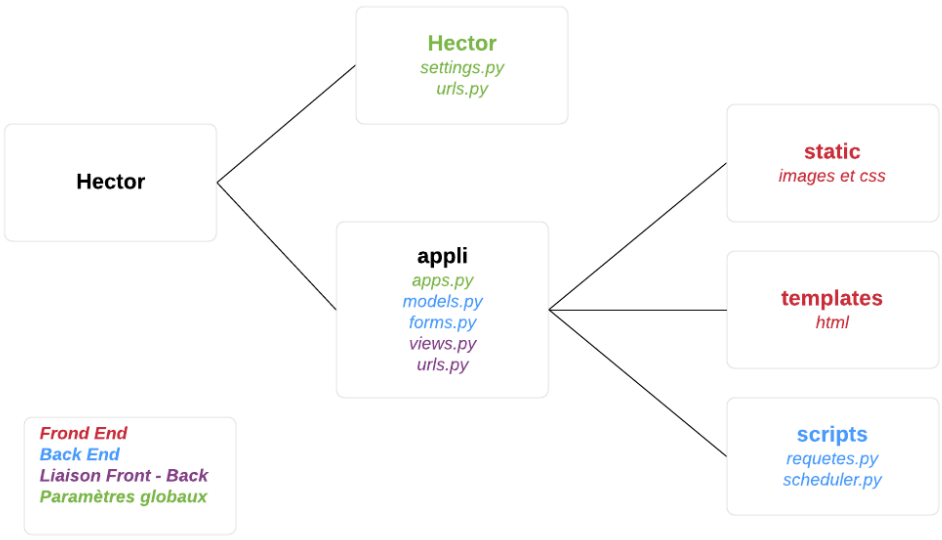
Le projet Hector a pour ambition de donner à voir aux utilisateurs de la ligne H la charge à bord de leurs trains en temps réel afin d'étudier l'impact de l'information voyageur sur la distribution à quai. Il fait suite à la mise en place pendant le déconfinement et dans certaines gares de la ligne H, d'affiches hebdomadaires de charge à bord des trains de la semaine passée (voir Annexe B). L'objectif était d'aider les passagers à mieux se répartir le long du quai pour faciliter le respect de la distanciation sociale. Nous aimerions aller plus loin dans l'information donnée aux voyageurs, et notamment étudier si l'accès à des données de charge à bord en temps réel permet de modifier leur comportement. Cette partie illustre les différentes étapes de la création du site web Hector, qui sera mis à disposition des clients d'ici octobre 2020, de la création de l'architecture à son développement.

3.1 Cadre de développement

Nous avons été trois à développer l'application, et avons pour cela utilisé le logiciel Git pour nous permettre de travailler ensemble et de garder une trace des modifications effectuées. Nous avons hésité entre RShiny et Django pour créer ce site web. Le deuxième a l'avantage à la fois de reposer sur le langage Python, plus reconnu que R dans l'entreprise, et de nous donner plus de liberté quand à l'interface utilisateur. Nous avons donc opté pour cette option, afin de monter en compétences sur le langage Python.

Django permet de faire la distinction entre le développement Front End de l'interface utilisateur, qui peut être réalisé indépendamment dans les dossiers **static** et **templates**, et le développement Back End, surtout effectué dans le dossier **scripts** et les fichiers *models.py* et *forms.py*. Le lien entre les deux se fait surtout via les fichiers *views.py* et *urls.py*. Les autres fichiers sont majoritairement des fichiers de configuration de l'application. L'arborescence de notre cadre de développement est exposé dans la *Figure 11*.

FIGURE 11 – Arborescence des fichiers et dossiers de développement



3.2 Interface graphique

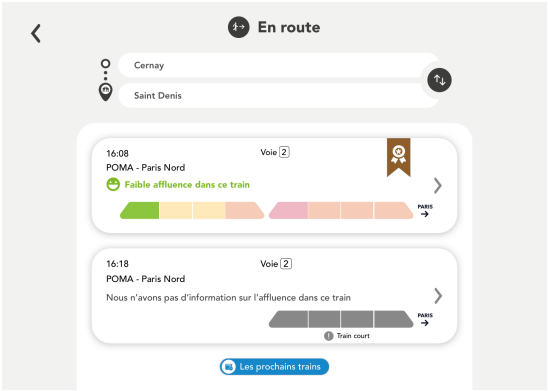
L'interface graphique de l'application Hector a été imaginée et codée par les designers de l'équipe (voir *Figure 12* et Annexe C). L'ensemble des fichiers html sont enregistrés dans le dossier **templates**, et l'ensemble des images et du code css sont stockés dans le dossier **static**. La version simple de l'application, qui est celle présentée dans ce rapport, est composée de 6 pages. La page d'accueil permet d'accéder à une page d'explications du projet ou à une page de recherche d'itinéraire. Sur la page de recherche d'itinéraire, le choix d'une gare de départ et d'une gare d'arrivée redirige vers une page de résultats, contenant des informations sur les prochains trains correspondant à la recherche. Cliquer sur l'un des trains mène vers une page de résultats plus précise, contenant des informations sur le train choisi par l'utilisateur ainsi que des conseils de placement à quai. Enfin sur cette page, l'utilisateur a la possibilité d'accéder à une page de retour d'expérience sur l'utilisation d'Hector, qui nous permettra par la suite d'analyser son impact sur le comportement des passagers.

FIGURE 12 – Différentes pages de l'application Hector

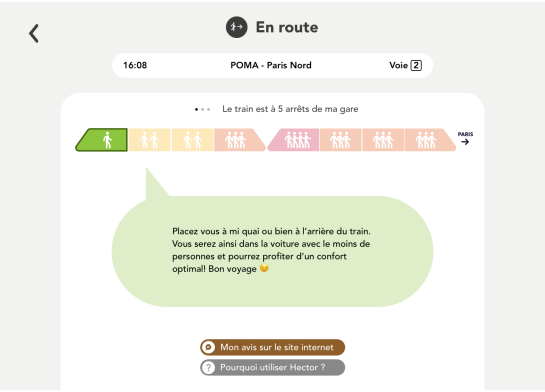
(a) Explications

(b) Accueil

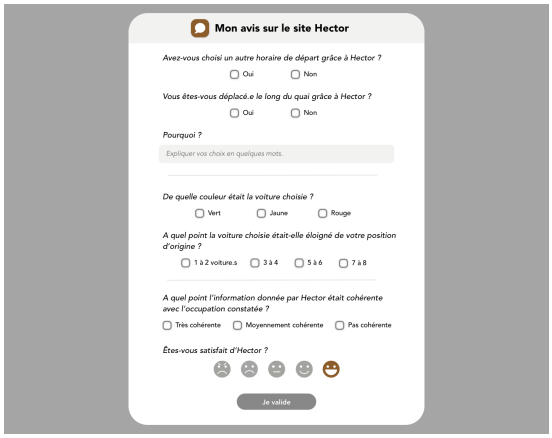
(c) Itinéraire



(d) Résultats et trains



(e) Résultats et conseils



(f) REX

3.3 Gestion des données

3.3.1 API disponibles

Pour mettre en place l'application Hector, nous avons eu besoin d'accéder à deux types d'information. D'abord, une information de charge par voiture, qui est la donnée principale que l'on souhaite délivrer aux voyageurs. Celle-ci est disponible via l'API Cave Keeper. Ensuite, une information sur la desserte du train, qui nous permettra de savoir quelles gares ont été et seront desservies. Cette donnée est disponible via l'API Course, qui rend disponible pour chaque train, la liste des gares desservies ainsi qu'un ensemble d'informations complémentaires comme les heures prévues d'arrivée et de départ, les numéros de voies ainsi que l'état du train (voir Annexe D).

3.3.2 Base de données relationnelle

Pour stocker les données que nous souhaiterions afficher à l'utilisateur, nous avons créé une base de données relationnelle composée de différents objets caractérisés par leurs attributs (voir *Figure 13*). On y trouve notamment deux objets correspondant aux gares (**Station**) et voies (**Platform**) de la ligne H. Ces objets sont finis, et sont intégrés une seule fois à la base de données sans modifications ultérieures. On trouve aussi un objet correspondant aux trains en circulation sur la ligne (**Course**) et aux différents arrêts desservis par ces trains (**Stoppoint**). Ces derniers permettent de suivre la localisation du train tout au long de son trajet. On a enfin pensé aux deux objets rames (**Consist**) et voitures (**Car**) associés à chaque arrêt desservi, et permettant notamment de stocker l'information de charge. Ces six objets sont liés entre eux.

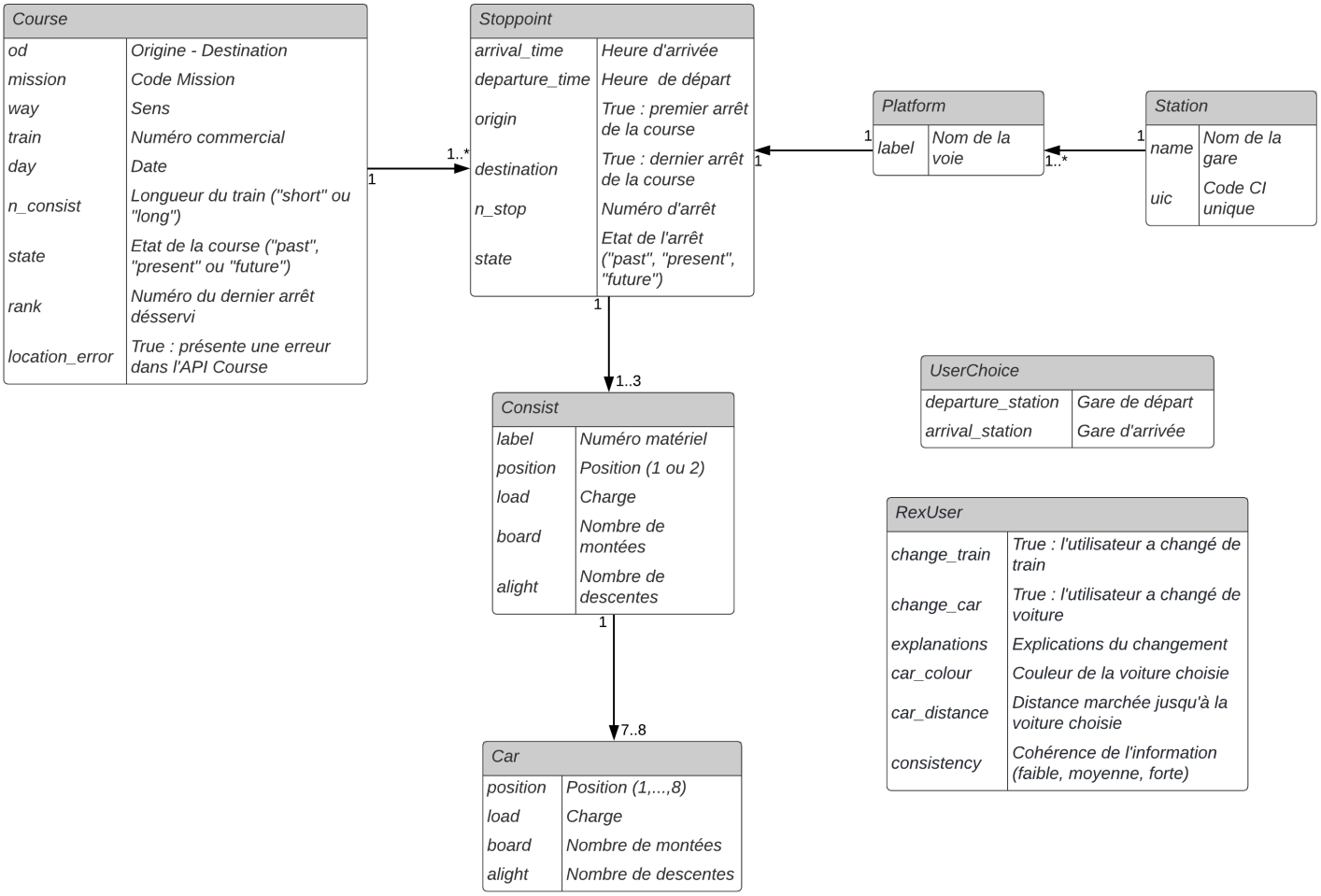


FIGURE 13 – Diagramme UML de la base de données relationnelle

Nous avons eu besoin de créer deux objets supplémentaires : **UserChoice**, correspondant aux choix d'itinéraire de l'utilisateur et **RexUser**, correspondant aux réponses au retour d'expérience utilisateur. Ce retour d'expérience est composé de 6 questions, rassemblées dans le *Tableau 11*. Ces deux objets permettent donc de stocker des informations obtenues par l'interaction de l'utilisateur avec la page, et ne sont pas dépendants des APIs commes les autres objets évoqués. Cette notion sera développée dans la section 3.5.1.

TABLE 11 – Questions associées aux attributs de l'objet RexUser

Attribut	Question	Réponses
change_car	Avez-vous choisi une autre voiture ?	Oui/Non
change_train	Avez-vous choisi un autre train ?	Oui/Non
explanations	Pourquoi ?	Ouverte
car_colour	De quelle couleur était la voiture choisie ?	Vert/Jaune/Rouge
car_distance	De combien de portes vous êtes-vous déplacé.e ?	1 à 8
consistency	L'information fournie était-elle cohérente ?	Faible/Moyenne/Forte

3.4 Mise à jour automatique de la base de données

Pour mettre à jour la base de données, nous avons créé deux scripts python : le script *requetes.py*, qui contient les fonctions d'appel des APIs et d'intégration des données dans la base, et le script *scheduler.py*, qui permet de lancer automatiquement et à intervalle régulier des fonctions. Il repose sur la fonction *BackgroundScheduler* du package **APScheduler** de python, et appelle la fonction de mise à jour de la base de données contenue dans le script *requetes.py* toutes les minutes. Chaque minute, les APIs Course et Cave Keeper sont appelées, et un processus de mise à jour des objets est lancé. Ce processus peut être découpé en deux types d'action :

- la création de nouveaux objets dans la base de données ;
- la mise à jour des attributs associés aux objets déjà existants dans la base de données.

3.4.1 Création des objets

Nous exposons donc ici les règles de création d'un nouvel objet dans la base de données. Deux règles générales se retrouvent d'un objet à l'autre : l'objet à créer ne doit pas déjà exister dans la base et les objets auxquels il est relié doivent déjà exister dans la base. Par exemple, la création de l'objet **Stoppoint** nécessite que la course et la voie auxquelles il est associé existent déjà dans la base de données. En plus de cela, nous avons rédigé quelques règles supplémentaires plus spécifiques. En l'occurrence :

- la création d'un objet **Course** nécessite que le train transporte des passagers ($with\_travellers = True$) ;
- la création d'un objet **Consist** nécessite, si le train est long, que l'information de position des rames (tête ou queue) soit disponible pour au moins une des rames.

3.4.2 Mise à jour des objets

La mise à jour ne concerne que deux objets : **Stoppoint** et **Course**, et plus précisément l'attribut *state* de ces objets, et *rank* de la course. Cette mise à jour est essentielle d'une part, pour savoir si le train est actuellement en circulation sur la ligne, et d'autre part pour pouvoir suivre la localisation du train le long de son trajet. Elle est réalisée si et seulement si les objets existent déjà dans la base de données. Deux processus ont lieu :

- le premier se produit seulement si le train est trouvé dans l'une ou l'autre des APIs ;
- le deuxième se produit à chaque fois que le script est appelé.

Dans le cas du premier processus, l'attribut *location_error* permet de distinguer les courses dont l'information d'avancement est présente dans l'API Course de celle dont l'information est absente. Le processus de mise à jour des attributs dans l'un ou l'autre des cas est mis en évidence dans la *Figure 14*.

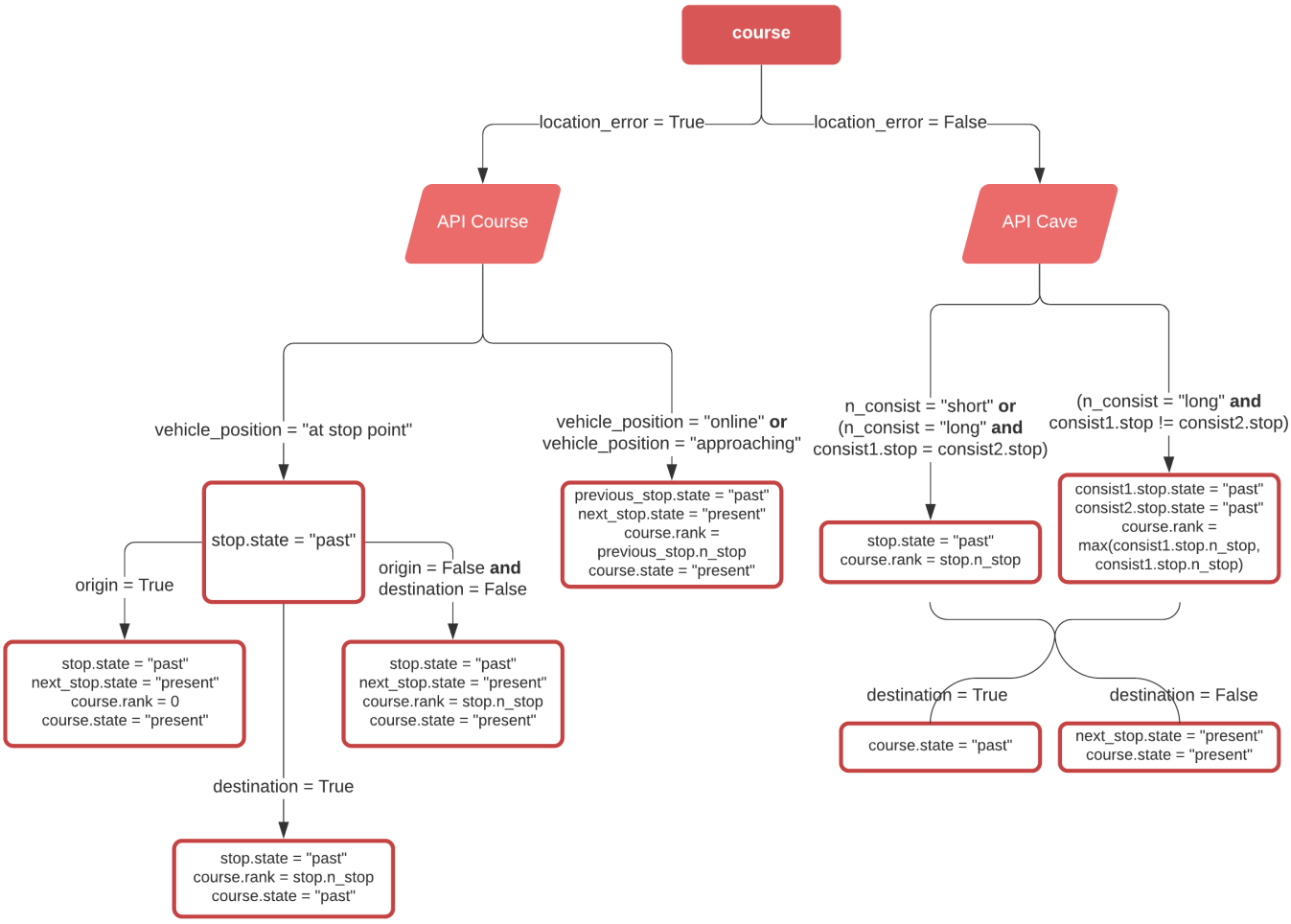


FIGURE 14 – Processus de mise à jour des attributs de la base de données

Le deuxième processus est plus général et concerne tous les trains existant dans la base de données dont l'attribut *state* est « present ». Dans ce cas, on effectue trois tests :

- on regarde si le terminus du trajet a été désservi. Si c'est le cas, le train n'est plus en circulation : le rang de la course devient le numéro d'arrêt du terminus et le statut de la course devient « past » ;
- on regarde si l'heure d'arrivée au terminus est dépassée de plus d'une heure, pour prendre en compte la possibilité d'un retard. Si c'est le cas, le train n'est plus en circulation : le rang de la course devient le numéro d'arrêt du terminus et le statut de la course devient « past » ;
- on regarde si plusieurs arrêts ont un statut « present » c'est-à-dire, prochainement désservi. Si c'est le cas, alors tout arrêt antérieur à l'arrêt « present » le plus proche du terminus doit voir son statut devenir « past ».

3.5 Interactions avec l'utilisateur

3.5.1 Formulaires

Deux pages ont la particularité de laisser l'utilisateur interagir fortement avec l'application : la page de recherche d'itinéraire et la page de retour d'expérience des utilisateurs. Dans le premier cas, l'utilisateur peut choisir les gares de départ et de destination qui l'intéressent, afin d'accéder à une information de charge filtrée pour les trains qui le

concernent. Dans le deuxième cas, les utilisateurs peuvent répondre au questionnaire qui servira à l'étude d'impact sur la répartition à quai. Ces informations doivent pouvoir être stockées, et pour le premier cas, être réutilisées dans les pages suivantes. Pour cela, il existe des objets appelés formulaires (ou forms en anglais), qui peuvent être créés dans le fichier *forms.py*. Nous en construisons deux pour les besoins de l'application :

- **UserChoiceForm**, contenant les gares de départ et d'arrivée sélectionnées par l'utilisateur ;
- **RexUserForm**, contenant les réponses aux différentes questions posées pour l'analyse d'impact.

Ces deux formulaires sont liés aux objets **UserChoice** et **RexUser** présentés dans la section 3.3.2, ce qui permet de stocker toute réponse au formulaire dans la base de données.

3.5.2 Vues et urls

Pour chaque page, un fichier html a été créé et stocké dans le dossier **templates**. Chacune de ses pages est également associée à un lien url qui doit rediriger vers cette page à chaque utilisation. Ces liens urls sont spécifiés dans le fichier *urls.py*. Pour faire la correspondance entre les pages html et les liens url, Django prévoit des vues (ou views, en anglais) qui sont stockées dans le fichier *views.py*. En plus de faire cette correspondance, elles permettent de fournir à chaque page html un contexte, c'est-à-dire un ensemble de variables qu'on aimerait pouvoir intégrer dans le html. Elles permettent aussi de faire des transformations sur ces variables en fonction des interactions avec l'utilisateur.

Typiquement dans la vue associée à la page « Résultats et trains », nous devons extraire les résultats du formulaire **UserChoiceForm** associés à la recherche d'itinéraire et appliquer à ces gares la méthode *select_last_stoppoints* pour obtenir les trains correspondant à la requête. Nous utilisons aussi sur le dernier arrêt désservi par ces trains la méthode *CarTrainColour* pour obtenir le code couleur de la charge à bord. Nous sommes ensuite capables de fournir à la page html un contexte constitué, pour chaque train concerné, des informations sur le code mission, la destination, l'horaire prévu d'arrivée, la voie desservie, la longueur et la charge à bord du train.

3.6 Sécurité et accessibilité des données

Parce que ce site web n'est pas destiné à être utilisé en interne, mais bien à être exposé aux clients de la ligne H, nous avons dû régler certaines contraintes imposées par l'entreprise concernant la sécurité des données. Nous avons en l'occurrence réalisé, avec un Responsable de la Sécurité des Systèmes d'Information (RSSI), une « analyse ZEN » qui a pour but de vérifier que les données sont exposables à l'externe et de spécifier des exigences en termes de sécurité et protection des données.

Des conditions générales d'utilisation (CGU) sont en cours de rédaction et ont vocation à être intégrées au site web. Ces CGU spécifient d'une part ce qu'est Hector, l'usage qui doit en être fait par l'utilisateur, mais surtout la manière dont seront traitées les données personnelles recueillies sur le site. En l'occurrence, aucune donnée personnelle n'est recueillie sur le site, et seules les requêtes faites par l'utilisateur sont enregistrées sous un identifiant différent à chaque requête. L'acceptation des CGU par l'utilisateur est nécessaire afin qu'il puisse accéder au site web.

Afin de rendre disponible le site web à l'externe, et parce que nous avons réalisé nous-même le développement en local sur nos ordinateurs, nous aurons besoin de faire migrer notre environnement de développement vers un serveur, géré par une entité externe à la

SNCF. Les démarches sont actuellement en cours et nécessitent également de faire des demandes d'accès aux APIs en dehors du réseau interne de la SNCF.

3.7 Conclusion

Nous avons réussi à créer un site web permettant aux utilisateurs de la ligne H d'accéder à la charge à bord de leurs trains en temps réel. Cette application est fonctionnelle bien qu'un certain nombre d'améliorations restent à faire. D'une part, le modèle identifié dans la section 2 sera intégré dans le traitement des données afin d'améliorer l'information de charge fournie. D'autre part, la version du site présentée dans ce rapport n'est pas la version cible d'Hector. Des fonctionnalités supplémentaires devraient être intégrées dans un second temps, notamment la possibilité pour le voyageur de préparer son voyage à l'avance. Il reste enfin quelques démarches administratives avant de pouvoir mettre Hector à disposition des clients, idéalement d'ici à fin septembre 2020.

Cette mise à disposition des clients sera précédée d'une campagne de communication autour du projet afin de toucher le plus de voyageurs possible. Elle reposera notamment sur la sollicitation d'un panel d'utilisateurs de la ligne H habitués à ce genre d'expérimentation. Le retour d'expérience inclus dans le site web sera une première façon de juger l'impact de l'information de charge en temps réel sur le comportement des voyageurs. L'analyse de la distribution à quai en comparaison d'une semaine contrôle, où l'application n'est pas mise à disposition, permettra aussi de se faire une idée de l'impact d'Hector sur la répartition à quai. La plus-value apportée par Hector sera déterminante sur la suite du projet, notamment son intégration dans les outils déjà existants, à savoir le site web Transilien.com ou les écrans en gare.

Bibliographie

Blainey, S., Hickford, A. & Preston, J. (2012), 'Barriers to passenger rail use : A review of the evidence', *Transport Reviews* **32**(6), 675–696.

Charles, S. (2020), 'Attractivité du mass transit dans un contexte d'exigence sanitaire élevée'.

Cornet, S., Buisson, C., Ramond, F., Bouvarel, P. & Rodriguez, J. (2019), 'Methods for quantitative assessment of passenger flow influence on train dwell time in dense traffic areas', *Transportation Research Part C : Emerging Technologies* **106**, 345–359.

Coulaud, R. (2019), 'Modélisation et prévision de l'affluence et du temps d'échange à l'interface quai/train : application à la ligne h'.

De Vos, J. (2020), 'The effect of COVID-19 and subsequent social distancing on travel behavior', *Transportation Research Interdisciplinary Perspectives* **5**.

Fang, J., Fujiyama, T. & Wong, H. (2019), 'Modelling passenger distribution on metro platforms based on passengers' choices for boarding cars', *Transportation Planning and Technology* **42**, 1–17.

Ghalanos, A. & Theussl, S. (2015), *Rsolnp : General Non-linear Optimization Using Augmented Lagrange Multiplier Method*. R package version 1.16.

Grimshaw, S. & Alexander, W. (2011), 'Markov chain models for delinquency : Transition matrix estimation and forecasting', *Applied Stochastic Models in Business and Industry* **27**, 267 – 279.

Kim, H., Kwon, S., Wu, S. K. & Sohn, K. (2014), 'Why do passengers choose a specific car of a metro train during the morning peak hours ?', *Transportation Research Part A : Policy and Practice* **61**, 249 – 258.

Krstanoski, N. (2014), 'Modelling passenger distribution on metro station platform', *International Journal for Traffic and Transport Engineering* **4**(4), 456 – 465.

Lam, W. H. K., Cheung, C.-Y. & Lam, C. F. (1999), 'A study of crowding effects at the hong kong light rail transit stations', *Transportation Research Part A : Policy and Practice* **33**(5), 401 – 415.

Oliveira, L., Bruen, C., Birrell, S. & Cain, R. (2019), 'What passengers really want : Assessing the value of rail innovation to improve experiences', *Transportation Research Interdisciplinary Perspectives* **1**.

Oliveira, L. C., Fox, C., Birrell, S. & Cain, R. (2019), 'Analysing passengers' behaviours when boarding trains to improve rail infrastructure and technology', *Robotics and Computer-Integrated Manufacturing* **57**, 282 – 291.

Peftitsi, S., Jenelius, E. & Cats, O. (2020), 'Determinants of passengers' metro car choice revealed through automated data sources : a stockholm case study', *Transportmetrica A : Transport Science* **16**(3), 529–549.

Seriani, S., Fernandez, R., Luangboriboon, N. & Fujiyama, T. (2019), 'Exploring the effect of boarding and alighting ratio on passengers' behaviour at metro stations by laboratory experiments', *Journal of Advanced Transportation* **2019**.

Shelat, S., Daamen, W., Kaag, B., Duives, D. & Hoogendoorn, S. (2020), 'A markov-chain activity-based model for pedestrians in office buildings', *Collective Dynamics* **5**.

Tirachini, A., Hensher, D. A. & Rose, J. M. (2013), 'Crowding in public transport systems : Effects on users, operation and implications for the estimation of demand', *Transportation Research Part A : Policy and Practice* **53**, 36 – 52.

Toqué, F., Khouadjia, M., Côme, E., Trépanier, M. & Oukhellou, L. (2017), Short long term forecasting of multimodal transport passenger flows with machine learning methods, pp. 560–566.

Wang, C., Yan, D. & Jiang, Y. (2011), 'A novel approach for building occupancy simulation', *Building Simulation* **4**(2), 149–167.

Zhang, Y., Jenelius, E. & Kottenhoff, K. (2017), 'Impact of real-time crowding information : a stockholm metro pilot study', *Public Transport* **9**(3), 483–499.

Annexes

Annexe A : gradient et hessienne du modèle contraint

Calcul des dérivées partielles d'ordre 1 :

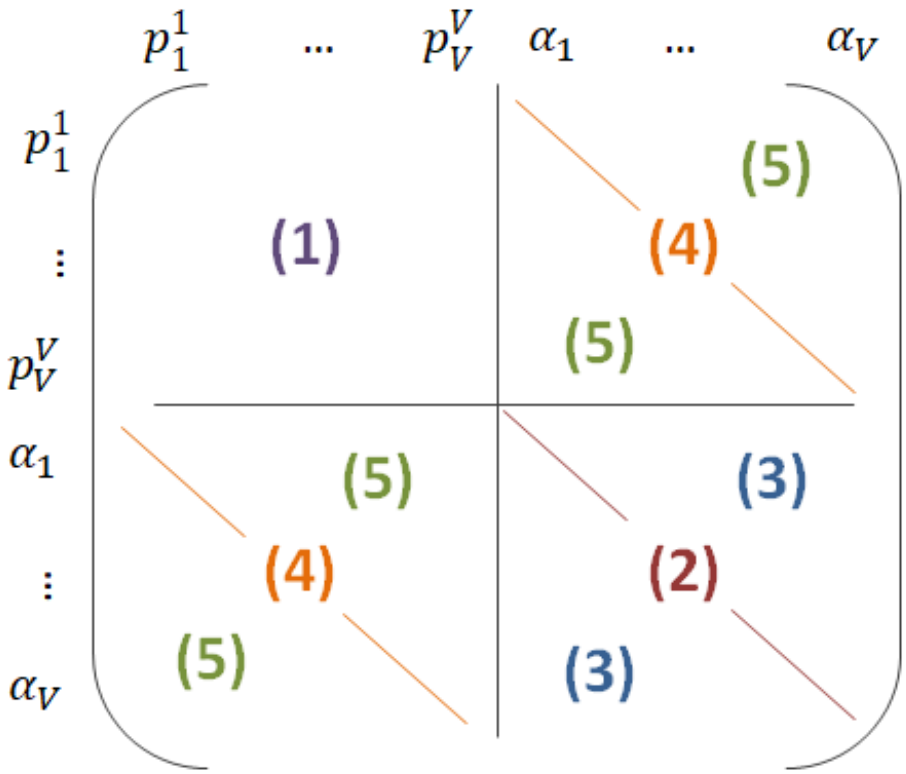
$$\forall v \in \mathbb{V}, \begin{cases} \frac{\partial f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial p_v^v} = 2m_{v,k}\alpha_v^{|v'-v|}(\sum_{v=1}^{V} m_{v,k}p_v^v\alpha_v^{|v'-v|} - d_{v',k}) \\ \frac{\partial f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial \alpha_v} = 2\,|v'-v|\,p_v^v m_{v,k}\alpha_v^{|v'-v|-1}(\sum_{v=1}^{V} m_{v,k}p_v^v\alpha_v^{|v'-v|} - d_{v',k}) \end{cases}$$

Calcul des dérivées partielles secondes : $\forall v, v" \in \mathbb{V}$

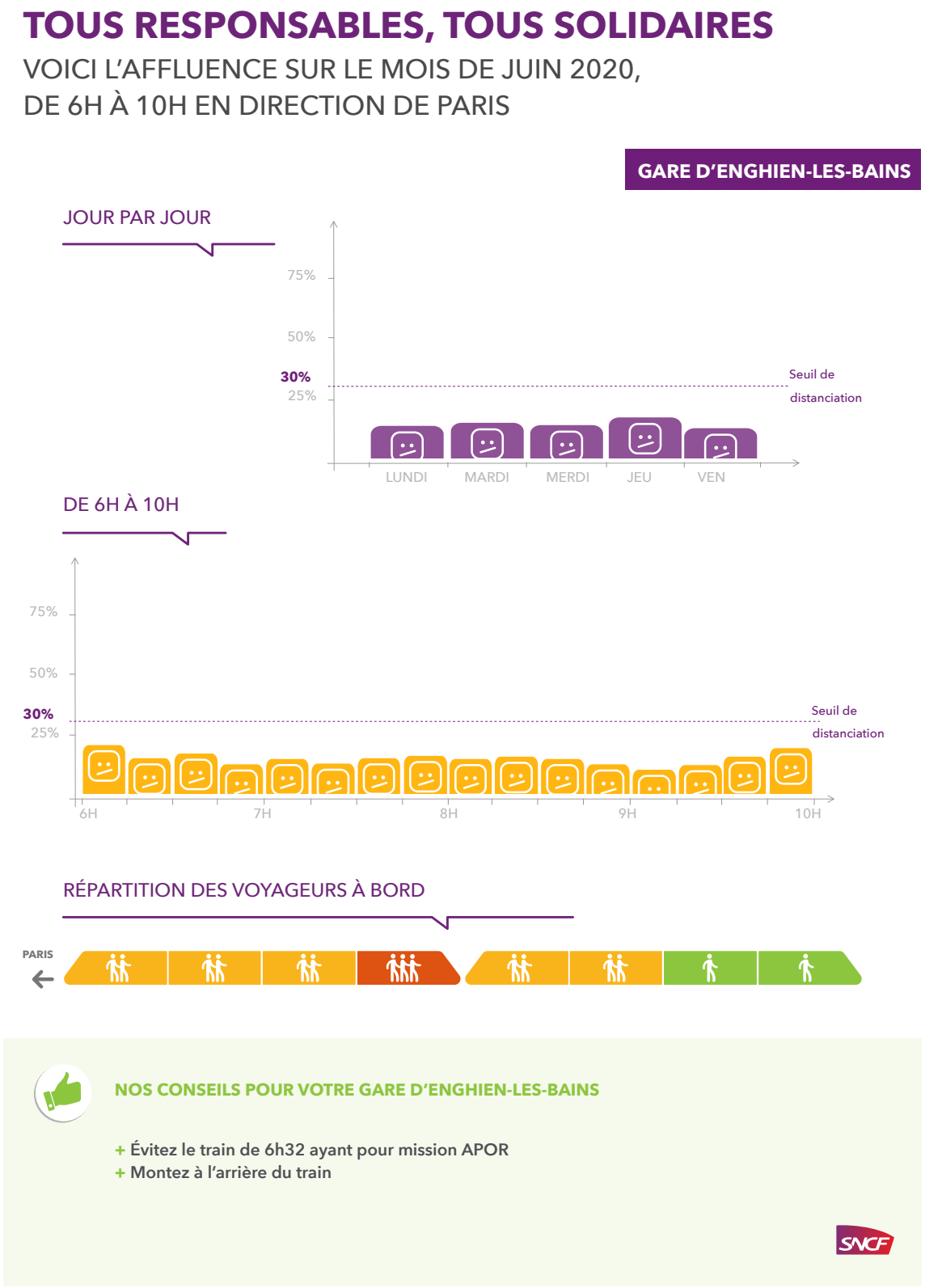
$$\begin{cases} (1) & \frac{\partial^2 f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial p_v^v \partial p_{v"}^{v"}} = 2m_{v,k}\alpha_v^{|v'-v|}m_{v",k}\alpha_{v"}^{|v'-v"|} \\ (2) & \frac{\partial^2 f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial \alpha_v \partial \alpha_v} = 2\,|v'-v|\,p_v^v m_{v,k}\alpha_v^{|v'-v|-1}\,|v'-v|\,p_v^v m_{v,k}\alpha_v^{|v'-v|-1} \\ & \qquad +2\,|v'-v|\,(|v'-v|-1)p_v^v m_{v,k}\alpha_v^{|v'-v|-2}(\sum_{v=1}^{V} m_{v,k}p_v^v\alpha_v^{|v'-v|} - d_{v',k}) \\ (3) & \frac{\partial^2 f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial \alpha_v \partial \alpha_{v"}} = 2\,|v'-v|\,p_v^v m_{v,k}\alpha_v^{|v'-v|-1}\,|v'-v"|\,p_{v"}^{v"} m_{v",k}\alpha_{v"}^{|v'-v"|-1} \\ (4) & \frac{\partial^2 f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial \alpha_v \partial p_v^v} = 2\,|v'-v|\,m_{v,k}\alpha_v^{|v'-v|-1}(m_{v,k}p_v^v\alpha_v^{|v'-v|} + \sum_{v=1}^{V} m_{v,k}p_v^v\alpha_v^{|v'-v|} - d_{v',k}) \\ (5) & \frac{\partial^2 f(p_v^v, \alpha_v, m_{v,k}, d_{v,k})}{\partial \alpha_v \partial p_{v"}^{v"}} = 2\,|v'-v|\,p_v^v m_{v,k}\alpha_v^{|v'-v|-1}m_{v",k}\alpha_{v"}^{|v'-v"|} \end{cases}$$

Un schéma de la matrice hessienne avec les appels à chaque type de dérivée partielle seconde peut être trouvée dans la figure suivante :

FIGURE 1 – Matrice hessienne associée au modèle contraint



Annexe B : Affiches hebdomadaires exposées pendant le déconfinement sur la ligne H



Annexe C : Pages d'Hector

FIGURE 1 – Explications sur Hector

FIGURE 2 – Page d'accueil

FIGURE 3 – Recherche d'itinéraire

FIGURE 4 – Résultats et trains

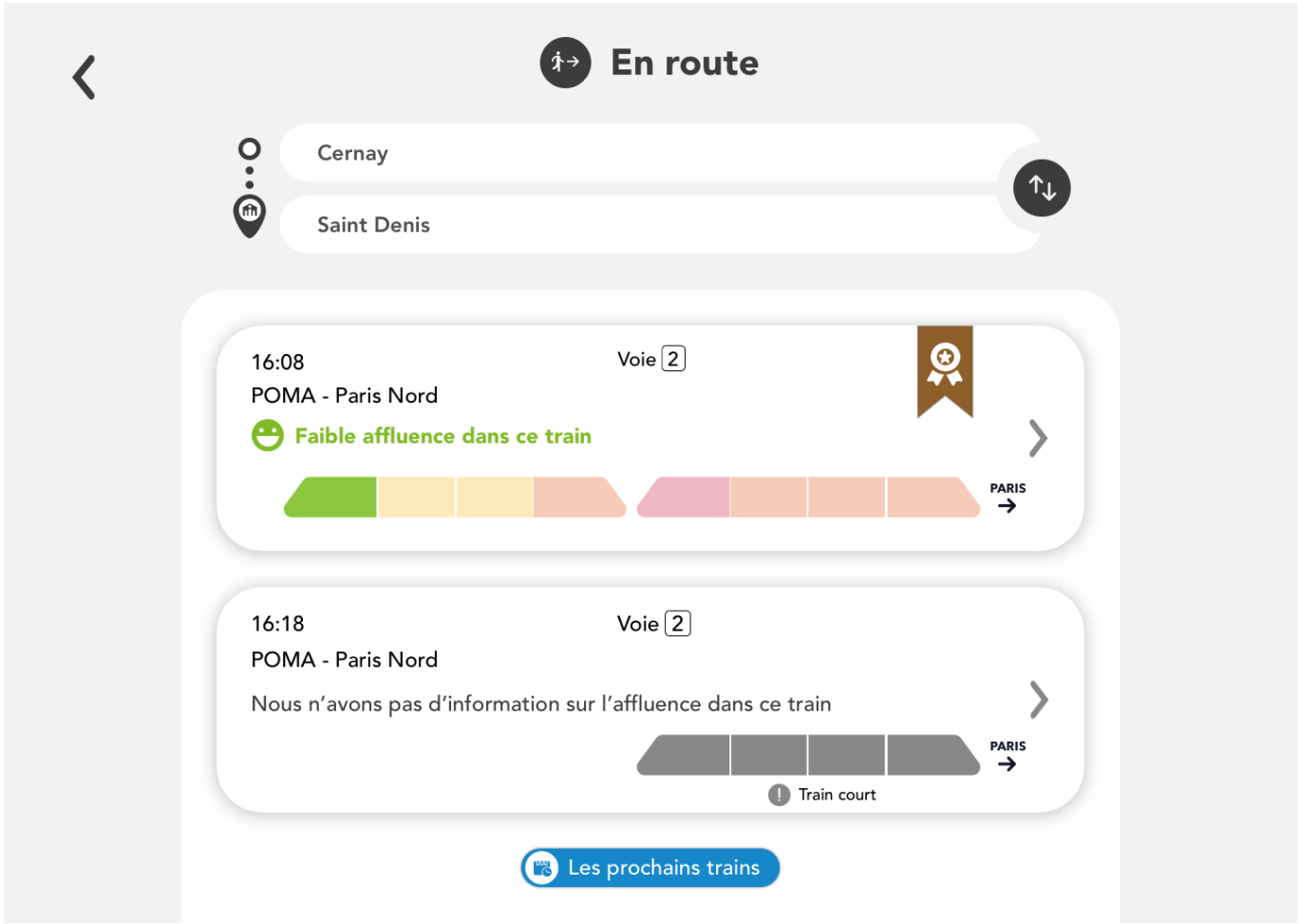


FIGURE 5 – Résultats et conseils

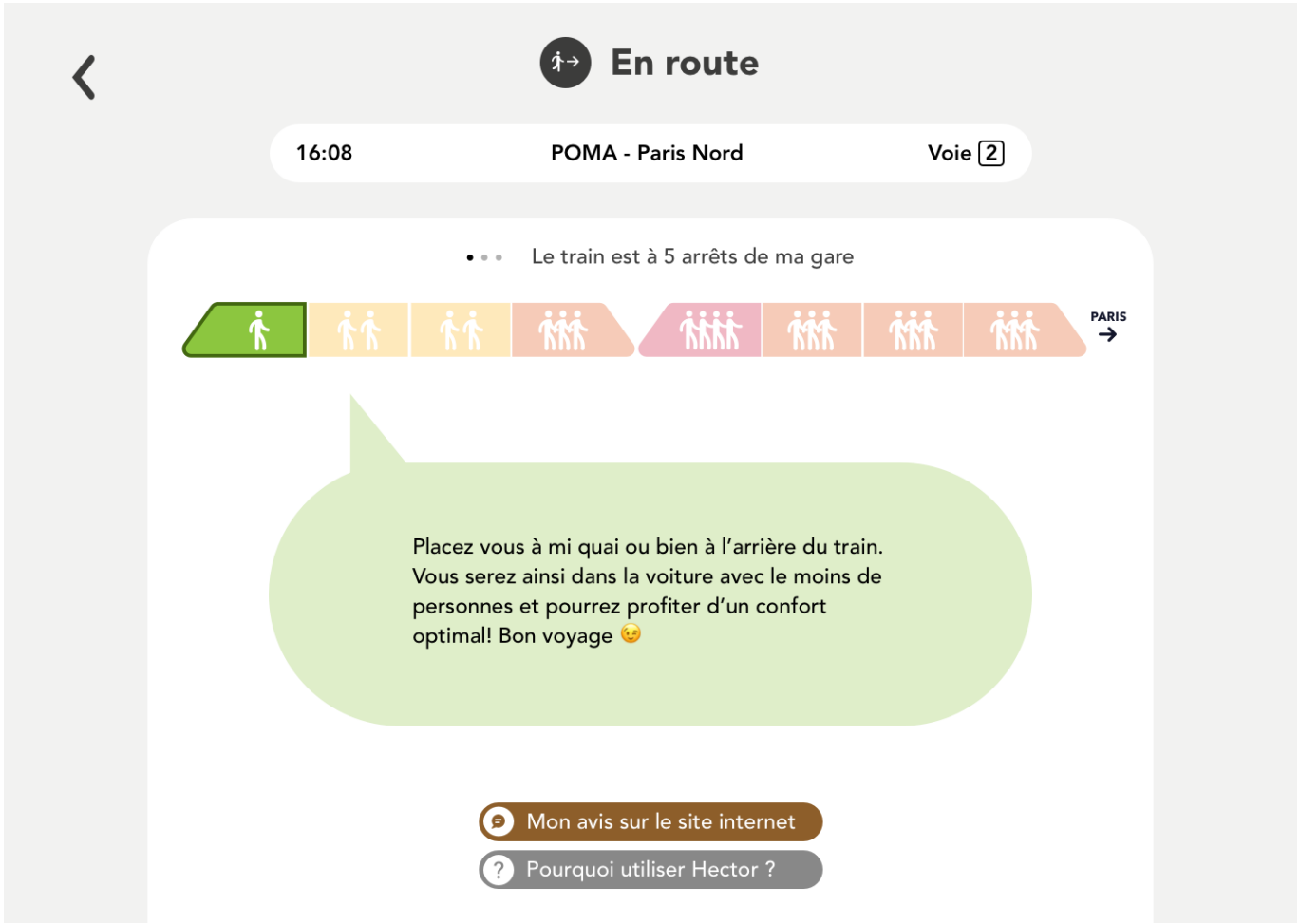
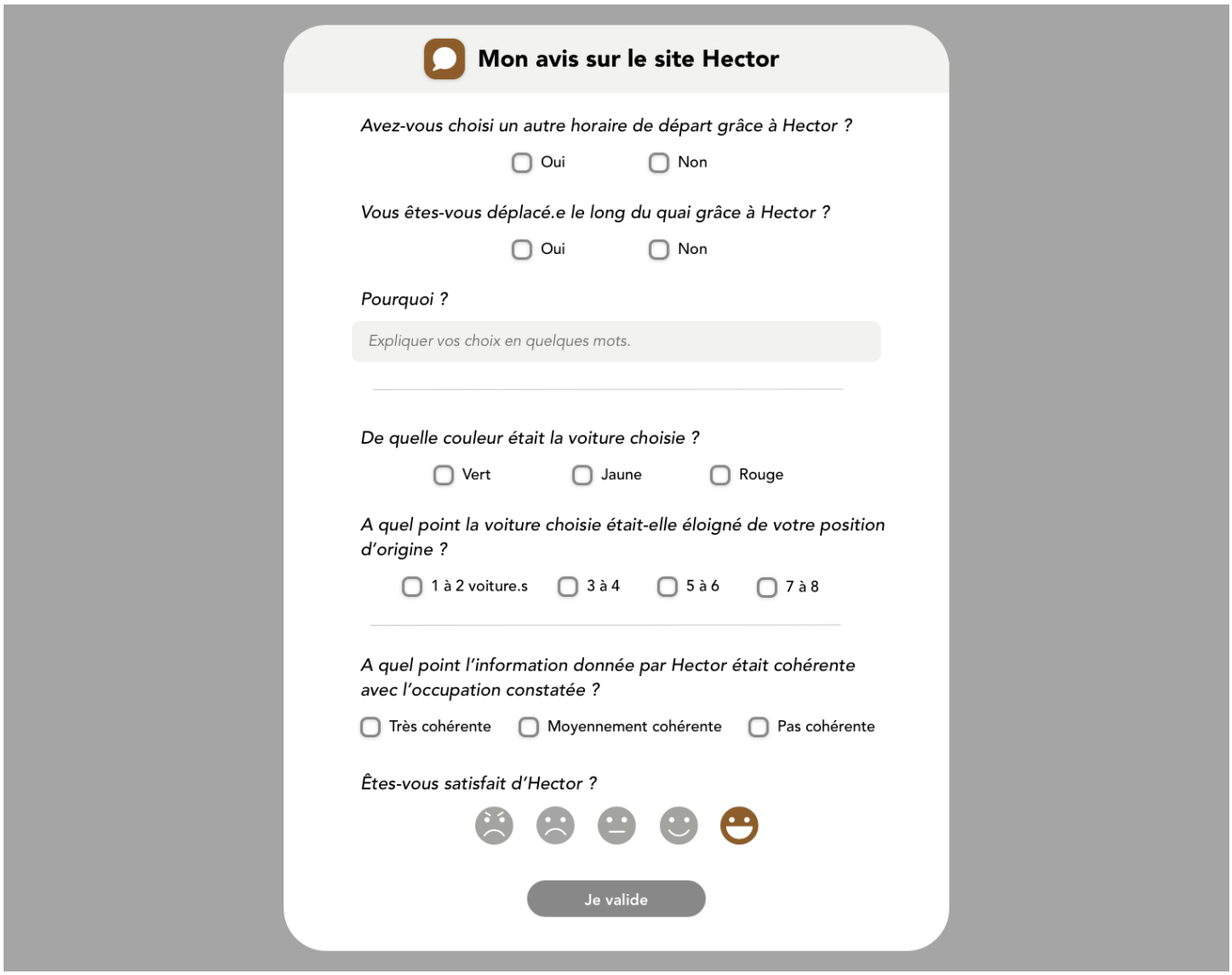


FIGURE 6 – Retour d'expérience



Annexe D : Données disponibles dans l'API Course et l'API Cave Keeper

TABLE 1 – Description des variables à disposition dans l'API Course

Nom des champs	**Description**
number	Numéro commercial du train
mission_code	Code de la mission
with_traveler	True/False : train avec/sans voyageurs
origin_id	Identifiant de la gare de départ
origin_departure_time	Date et heure théorique de départ à l'origine
destination_id	Identifiant de la gare d'arrivée
destination_arrival_time	Date et heure théorique d'arrivée à destination
vehicle_size	Nombre de rames : "long","short"
line.code	Code de la ligne
vehicle_position	Localisation de la course
.state	Position du train : "Online", "Unlocated","At stop point", "Approaching"
.current_stop_point	Identifiant de la gare d'arrêt actuelle
.previous_stop_point	Identifiant de la précédente gare d'arrêt
.next_stop_point	Identifiant de la prochaine gare d'arrêt
stop_points[i]	i^{me} arrêt de la course
.boarding_area.id	Identifiant du quai d'arrêt
.boarding_area.label	Nom du quai d'arrêt
.id	Identifiant de la gare d'arrêt
.label	Nom de la gare d'arrêt
.rank	Numéro d'arrêt
.times.departure_time	Heure de départ théorique de l'arrêt
.times.arrival_time	Heure d'arrivée théorique à l'arrêt

TABLE 2 – Description des variables à disposition dans l'API Cave Keeper

Nom des champs	**Description**
numero	Numéro commercial du train
codeLigneCommerciale	Ligne commerciale
codeMission	Code de la mission
dateHeureComptage	Date et heure d'envoi du dernier fichier de comptage
dateHeureOrigine	Date et heure de départ à l'origine
capacite	Capacité totale du train
total	Charge à bord du train
trainDirection	Indique si l'ordre des rames est inversé
typeMateriel	Type de matériel (NAT/R2N)
materiels[i]	Description de la rame i
.capacite	Capacité totale de la rame
.label	Identifiant de la rame
.ordre	Position de la rame dans le train (1 ou 2)
.comptage.codeGare	Code gare unique
.comptage.in	Nombre montées dans la rame
.comptage.out	Nombre de descentes de la rame
.comptage.total	Charge à bord de la rame
.comptage.estOrigine	True : l'arrêt est une origine
.comptage.estDestination	True : l'arrêt est un terminus
materiels[i].comptage.voitures.pcs.car[j]	Descriptif de la voiture j de la rame i
.lCst.in	Nombre de montées dans la voiture
.lCst.out	Nombre de descentes de la voiture
.lCst.total	Charge à bord de la voiture