Big data analysis in the field of transportation

Léna Carel

université PARIS-SACLAY

BNNT: 2019SACLG001

Thèse de doctorat

Analyse de données volumineuses dans le domaine du transport

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'École Nationale de la Statistique et de l'Administration Économique

École doctorale n° 574 EDMH
Spécialité de doctorat: Mathématiques Appliquées

Thèse présentée et soutenue à l'ENSAE, le 8 février 2019, par

Léna Carel

Après avis des rapporteurs:

Charles BOUVEYRON	Université Côte d'Azur
Latifa OUKHELLOU	IFSTTAR

Composition du jury:

Pierre ALQUIER ENSAE	Directeur de thèse
Denis COUTROT Transdev Group	Invité
Yohann DE CASTRO École des Ponts ParisTech	Examinateur
André DE PALMA ENS Paris-Saclay	Invité
Mathilde MOUGEOT ENSIIE	Présidente du jury
Latifa OUKHELLOU IFSTTAR	Rapporteur

Au début, la montagne est une montagne.
À la fin, la montagne est une montagne.
Mais au milieu, la montagne est une pente.

Anonyme

Remerciements

Même s'il est universellement reconnu qu'une thèse est le fruit d'un travail solitaire, le manuscrit que vous tenez actuellement entre vos mains n'aurait probablement jamais vu le jour sans la rencontre et le soutien de nombreuses personnes. Je tiens donc à leur adresser ici mes remerciements les plus sincères.

Dans un premier temps, je tiens à remercier infiniment Pierre, mon directeur de thèse, qui a été d'un soutien sans faille et un encadrant hors pair. Merci pour sa patience, sa gentillesse et son humour. Cette thèse aurait indéniablement été différente sans lui. Je lui en suis entièrement reconnaissante.

Je voudrais également remercier Denis et Nadir, tout d'abord pour leur accueil chez Transdev et leur intégration avant le début de la thèse, puis pour leur soutien tout au long des trois années passées à travailler ensemble. Je les remercie de m'avoir permis de travailler sur des données réelles et ainsi avoir pu me confronter aux enjeux des transports et de la mobilité urbaine.

Je tiens ensuite à remercier l'ensemble des chercheurs du laboratoire de statistiques du CREST pour les échanges enrichissants que j'ai pu avoir avec eux.

Merci également et chaleureusement aux doctorants anciens et actuels. Le laboratoire serait bien vide sans leur présence, leurs pauses café, leurs parties de ping-pong, leurs discussions (rarement mathématiques il faut l'avouer), leurs fous-rires et j'en passe. Merci donc à: The Tien, Vincent, Mehdi, Edwin, Alexander, Lionel, Jérémy, Badr, Arthur, Jérôme, Mohammed, Gautier, Boris, Geoffrey, Solenne, Lucie, Yannick, Alexis, Avetik, Amir, Aria, François-Pierre.

Enfin, je remercie les rapporteurs et membres du jury de m'avoir fait l'honneur d'accepter de lire et d'examiner tout le travail des ces trois

dernières années.

D'un point de vue personnel, je tiens bien entendu à remercier en premier lieu mes parents. Ils m'ont éduquée de telle manière à éveiller ma curiosité scientifique, ils ont su me motiver dans les moments de doutes sans jamais me juger, ils ont soutenu tous mes choix et m'ont encouragé à devenir la mathématicienne que je suis aujourd'hui. Je ne peux clôturer ce paragraphe sans me rappeler avec nostalgie d'une discussion que j'ai eu avec mon père. Au début de mes études de mathématiques, je lui ai assuré que JAMAIS DE LA VIE je ne ferais de thèse, huit ans d'études c'est trop long. Il m'a rappelé qu'on ne sait jamais ce qui peut arriver, et je crois que ce souvenir m'a aidé à me lancer dans cette grande aventure doctorale, alors un grand merci pour ces paroles.

Depuis maintenant 26 ans, nous nous sommes souvent challengés avec mon frère quant à nos connaissances et nos notes en mathématiques. Je le remercie pour cette stimulation intellectuelle constante. Il a pourtant toujours cru en moi et je suis fière d'avoir pu être un modèle pour lui.

Je tiens aussi à remercier l'intégralité de ma famille et de ma belle-famille pour leur soutien durant ces trois années.

Merci aussi à mes amis, pour leur compréhension dans les moments où le travail a été trop important ou que j'étais trop fatiguée pour avoir le temps de les voir. Merci pour les soirées, les week-ends, les vacances, les verres, les séances de cinéma, les brunchs, les déjeuners, les dîners et tous les autres bons moments passés ensemble qui m'ont été bénéfiques à chaque fois. Merci Hélène, Anne, Claire, Mathilde, Baptiste, Ronan, Salma, Nina, Hamid, Quentin, Ismaïl, Alice,...

Enfin, ces remerciements ne seraient pas complets si je ne remerciais pas mon soutien indéfectible quotidien. Merci pour tout, pour nos fous-rires, pour nos projets, pour nos discussions statistiques, pour ton humour, pour ton sérieux, mais surtout pour ton soutien et tes encouragements. Merci Thomas !

Contents

Chapter 1

Introduction (French)

1.1 Contexte et motivations

Après la seconde guerre mondiale, les prix fonciers dans les centres-villes des États-Unis ont soit baissé soit connu une très légère augmentation, tandis que ceux des zones périphériques des villes et métropoles ont explosé. Deux principales théories ont expliqué ce phénomène. L'une d'un point de vue économique et l'autre d'un point de vue écologique. Les économistes soutenaient qu'une personne cherchant à acheter un terrain fera face à un compromis entre la taille du terrain et la distance du centre-ville. Alors que les écologistes affirment que cet individu "maximiserait sa satisfaction en évitant les biens qu'il n'aime pas et en possédant et consommant ceux qu'il aime". Dans ce contexte, les auteurs de [Alonso, 1964] affirmèrent qu'une telle prise de décision est bien plus complexe. En effet, leur théorie est qu'ayant un certain revenu à dépenser comme il le souhaite, un individu prendrait en compte différents critères comme le prix foncier, le coût des trajets domicile-travail et autres dépenses. C'est la première fois que des chercheurs utilisèrent le transport pour expliquer la croissance urbaine et de nombreux autres travaux ont suivi.

Depuis, les auteurs de [Duranton and Turner, 2012] ont étudié en 2012 l'impact du développement des autoroutes sur les villes américaines et ont découvert que celui-ci a un fort impact positif sur le taux d'emploi. Ce résultat a donc montré que le développement des infrastructures de transport influe sur la croissance urbaine, puisque celles-ci permettent aux personnes habitant plus loin d'accéder plus facilement au centre-ville pour travailler.

Bien évidemment, il y a beaucoup d'autres facteurs à prendre en compte. Par exemple, [Black and Henderson, 1999] révéla que les tailles des villes sont fortement impactées par le niveau d'éducation de leurs citadins. En fait, les citoyens ayant fait des études supérieures sont plus susceptibles d'aider la ville à croître en innovant et en créant des emplois. C'est ce qu'a prouvé [Van Oort, 2017] aux Pays-Bas, qui indique que les échanges d'informations et de connaissances ont entraîné des taux de croissance économique plus élevés dans les zones urbaines et une plus forte intensité d'innovation dans les régions à forte activité économique.

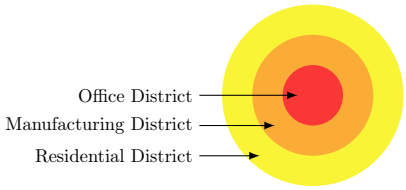
Dans l'Union européenne (UE), mais aussi ailleurs, l'une des pierres angulaires des stratégies de développement économique a été les infrastructures de transport. Cependant, [Crescenzi and Rodriguez-Pose, 2012] a montré que la croissance régionale dans l'UE est principalement menée par une combinaison de politiques sociales adéquates, d'une bonne capacité d'innovation dans la région et de l'attrait pour les nouveaux venus.

L'auteur de [Meyronin, 2015] attire l'attention du lecteur sur les villes qui utilisent des stratégies d'entreprise telles que le marketing pour renforcer l'image de marque de leur territoire. Par exemple, en 2006 la ville de Lyon a profité de la Fête des Lumières pour faire tester aux spectateurs un jeu vidéo simulant une balade en vélo à travers la ville. L'objectif de cette opération étant de renforcer l'image de ville digitale de Lyon. Dans le même style, Dubaï a développé son image de marque à l'international en communiquant sur ses principales attractions touristiques, comme les tours du Burj-Al-Arab et du Burj Khalifa, le complexe immobilier Palm Islands ou encore la toute première piste de ski située au milieu d'un désert.

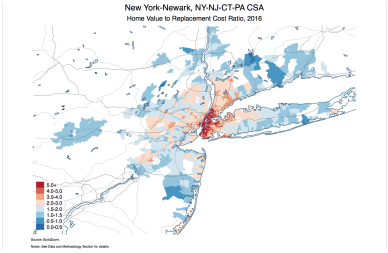
Un bon résumé du rôle du transport dans la croissance des villes est réalisé par [Duranton and Puga, 2014]. Non seulement les auteurs rappellent que dans les modèles de villes monocentriques – voir Figure 2.1 pour plus de détails sur le modèle de ville monocentrique – la croissance démographique, une plus grande suburbanisation et une consommation accrue de terres sont impliquées par des coûts de transport inférieurs. Mais ils rappellent aussi que les infrastructures sont souvent attribuées en fonction de la croissance attendue et que toute amélioration met du temps avant d'avoir un quelconque impact sur la croissance.

1.1.1 Considérations sociologiques

Il est assez évident qu'un sytème de transports public de grande qualité permet aux citadins de minimiser leur utilisation de véhicules motorisés



(a) Schéma théorique du prix du terrain dans le modèle de ville monocentrique. La valeur du terrain décroît lorsque la distance au centre-ville croît.



(b) Rapport valeur à neuf à New York et dans ses environs. Voici un exemple des valeurs foncières qui décrivent cette métropole comme étant monocentrique. Image de Issi Romem sur le blog Buildzoom [Romem, 2017].

Figure 1.1: Modèle de ville monocentrique – schéma théorique et exemple

personnels pour des trajets à travers la ville.

Ainsi, [Litman, 2016] a montré qu'un bon réseau de transport a un impact positif sur la santé de ses utilisateurs. En effet, ceci réduit les accidents de voiture et la pollution tout en augmentant la qualité de la santé mentale et physique des passagers, puisqu'ils marchent davantage pour accéder aux arrêts et stations de transport que les personnes utilisant leur véhicule. De plus, un réseau efficace permet aux personnes défavorisées sans voiture d'accéder à un plus grand nombre de quartiers et donc à davantage de services (tels que des magasins ou des services de santé) et d'améliorer leur mode de vie.

Les auteurs de [Gendron-Carrier et al., 2018] ont montré que lorsqu'une nouvelle ligne de métro est inaugurée, on mesure généralement une baisse des particules polluantes. Enfin dans [Zion and Lerner, 2017], les schémas de mobilité sont utilisés dans le but de comprendre les usages des différents quartiers de la ville et d'ainsi analyser la sociologie de celle-ci.

1.1.2 Considérations économiques

Dans le modèle de ville monocentrique que nous avons évoqué précédemment, les trajets domicile-travail quotidiens ont souvent été étudiés. En effet, dans les articles [Arnott et al., 1993; Anderson and De Palma, 2007] par exemple, les auteurs ont étudié la congestion aux entrées de la ville

ainsi que dans ses parkings durant les heures de pointe. Bien que les villes américaines – sur lesquelles la plupart des études sont basées – ont généralement des infrastructures de transport public peu développées, ce n'est pas le cas des villes européennes. Malheureusement, nous n'avons trouvé que quelques références traitant de l'impact d'un réseau de transport en commun développé sur la santé économique de la ville.

Certains papiers [De Palma et al., 2015, 2017] se sont concentrés sur les problèmes de congestion dans les transports en commun pour les voyageurs quotidiens. Ces deux papiers mettent en évidence le besoin d'horaires et de taille de trains optimales pour offrir à ces passagers des trajets dans des rames moins encombrées. En effet, les trains de plus grande capacité permettent un plus grand confort en heure de pointe, tandis que des horaires optimaux permettent aux passagers voulant éviter la foule d'arriver à destination plus tôt ou plus tard dans des trains moins remplis. Cette dernière proposition permet aussi l'étalement de l'heure de pointe.

Cependant, [Litman, 2015] a étudié les différences entre les villes avec de grosses de petites ou pas du tout d'infrastructure ferrée. Les auteurs ont tiré comme conclusion qu'une plus grosse infrastructure – donc des villes avec un réseau plus efficace – implique un plus fort taux d'achalandage, moins d'accidents de la route mortels et une plus faible part du budget des foyers allouée aux transports. Ainsi, le budget alloué aux autres biens et services peut être plus élevé. Tous ces résultats montrent qu'un réseau de transport efficace permet une économie locale plus saine et plus riche. Un dernier exemple est donné par [Pang, 2018], qui montre qu'un réseau dense facilite l'employabilité des travailleurs peu qualifiés.

Les études économiques mentionnées ci-dessus montrent qu'un réseau de transport public optimal n'est pas seulement d'une attractivité plus élevées pour les citoyens, mais semble aussi être un axe de développement économique pour les villes. Tout ce qui a été décrit dans cette thèse jusqu'alors motive l'étude détaillée des données de transport.

1.1.3 Villes intelligentes et données urbaines

Dans les études économiques traditionnelles, des données annuelles sont généralement utilisées. Mais depuis le développement de la miniaturisation numérique, chaque type d'objet imaginable contient un mini-ordinateur qui génère d'énormes quantités de données. Et comme [Batty, 2013] le présente, nous sommes à présent capables de mieux comprendre le fonctionnement des grandes villes à bien plus court terme que précédemment. Comme nous avons pu le voir plus tôt, les études traditionnelles se con-

centrent sur la localisation des différentes utilisations foncières et le fonctionnement à long terme des villes. Cependant, avec l'omniprésence de ces nouveaux capteurs, cela devient de plus en plus facile d'étudier le mouvement et la mobilité des citadins. De la même façon que nous nommons nos téléphones "intelligents" puisque ce sont de petits ordinateurs, on peut à présent se targuer de vivre dans des villes intelligentes.

Ce terme est apparu en 1994, mais est de plus en plus utilisé dans des articles scientifiques. Une définition complète des villes intelligentes peut être obtenue en combinant les travaux de [Dameri and Cocchia, 2013; Nam and Pardo, 2011; Harrison et al., 2010], pour n'en citer que quelques uns. Le but d'une ville intelligente est d'offrir les meilleures conditions de vie possibles à ses citoyens et visiteurs.

Pour atteindre ce but, ces villes doivent d'abord être instrumentalisées, c'est-à-dire être capable de collecter une large quantité de données à travers l'utilisation de capteurs, compteurs, kiosques, appareils personnels, appareils photo, appareils électroménagers, téléphones intelligents, dispositifs médicaux implantés et même des réseaux sociaux. Les autorités doivent ensuite être interconnectées en mettant en place une plate-forme où ces données peuvent être stockées et transférées entre les différents services de la ville. Enfin, elles devront devenir intelligentes en utilisant des méthodes d'analyse, de modélisation, informatiques et de visualisation. Tout ceci pourra alors servir à résoudre des problématiques urbaines de plus en plus récurrentes, comme la congestion des routes, la pollution sonore et de l'air, la consommation d'énergies et d'eau ainsi que le traitement des déchets.

Une ville intelligente est aussi une ville durable, puisque ces deux types de villes essaye de résoudre des problèmes similaires, principalement liés à l'environnement. En 2014 [Lee et al., 2014] cite, entre autres, l'exemple de la ville de San Francisco. À ce moment là, la ville était encore en train de définir sa stratégie "intelligente", mais avait déjà lancé sa propre plateforme de données ouvertes 'DataSF', avait plusieurs outils d'analyse intelligents basés sur des données en temps réel issues de services de transport intégrés pour de la prédiction en temps réel ou une tarification adaptée à la demande pour le stationnement.

Comme l'UE et les Nations Unies (ONU) ont fixé des objectifs ambitieux en matière de climat et d'énergie pour les années à venir, il a été souligné par [Ahvenniemi et al., 2017] qu'il était urgent de trouver des moyens plus intelligents de réduire la pollution et d'améliorer l'efficacité énergétique.

(a) Pollution à Paris (France). Photo de Alberto Hernandez sur flickr.

(b) Embouteillages à Gurgaon (Inde). Photo de Taresh Bhardwaj sur flickr.

Figure 1.2: Des problèmes rencontrés par les grandes villes

À vrai dire, [Zheng et al., 2014a] souligne qu'une partie de la solution est d'utiliser des méthodes statistiques et d'informatique urbaine. Aussi, les auteurs rappellent que l'on trouve différents types de données urbaines:

- Données géographiques: À Pékin, plusieurs études ont utilisé les trajectoires GPS de taxis. Ces données ont servi à évaluer l'efficacité de l'urbanisme, détecter des anomalies de circulation et détecter l'utilisation faite de chaque zone de la ville [Zheng et al., 2011; Pan et al., 2013; Yuan et al., 2012]. En Australie, des détecteurs Bluetooth ont été installés dans la ville de Brisbane. Ces données ont été regroupées spatio-temporellement pour pouvoir décrire la dynamique des véhicules dans la ville par [Laharotte et al., 2015].

- Données de trafic: Les auteurs de [Zhang et al., 2017] prévoient les affluences dans chaque région des villes de New-York et Pékin. À Pise, les données GPS sont utilisées pour détecter les embouteillages et incidents de circulation, puis informer les autres conducteurs présents dans la zone [D'Andrea and Marcelloni, 2017]. Enfin, [Castro et al., 2013] indique que ces données sont utilisées pour analyser trois dynamiques différentes : la dynamique sociale, la dynamique de circulation et la dynamique opérationnelle.

- Signaux de téléphone portable: Les téléphones mobiles génèrent une variété de données utiles à la planification urbaine. Par exemple, les capteurs de géolocalisation permettent de recommander de nouveaux lieux ou événements aux utilisateurs [Bothorel et al., 2018].

À Singapour, les auteurs de [Jiang et al., 2017] ont rapporté que l'enregistrement des détails des appels téléphoniques est utilisé pour mieux comprendre les schémas de mobilité humaine à travers la ville.

- Données de suivi environnemental: En France, [Abadi et al., 2017] explique que les compteurs intelligents sont tout à fait récents pour l'eau, mais qu'ils ont été capables d'aider à comprendre et prévoir la consommation d'eau. Concernant la qualité de l'air, le papier de [Zheng et al., 2014b] utilise des données historiques de neuf villes chinoises pour prédire en temps réel la pollution atmosphérique et en déduire la qualité de l'air dans les zones urbaines sans station de surveillance.

- Données des réseaux sociaux: Les réseaux sociaux offrent une grande quantité de données détaillées sur leurs utilisateurs. Par exemple, [Zheng, 2011] utilise l'historique de localisation des utilisateurs pour leur recommander de nouveaux contacts et créer des communautés ayant les mêmes intérêts. Au Japon, [Lee and Sumiya, 2010] détecte les événements inhabituels tels que des festivals avec des tweets géolocalisés, alors qu'à Pékin [Pan et al., 2013] décrit les anomalies de circulation avec la plateforme de micro-blogging WeiBo. Plus récemment, [Atefeh and Khreich, 2015] propose une étude sur les techniques de détection d'événements sur les tweets.

- Données économiques: Les auteurs de [Di Clemente et al., 2018; Louail et al., 2017] ont utilisé les achats par carte de crédit pour regrouper la population afin de révéler son mode de vie urbain et d'analyser les pratiques de mobilité commerciale pour compenser les inégalités socio-économiques entre quartiers.

- Données énergétiques: En Irlande, les ménages ont été regroupés selon leur comportement de consommation, grâce à des compteurs électriques intelligents [Melzi et al., 2017]. De plus, des données socio-économiques ont été utilisées pour analyser les regroupements.

- Données de santé: Dans [Dzhambov et al., 2018], les auteurs utilisent des données sur la santé mentale et sur la pollution sonore en ville pour établir un lien entre ces deux phénomènes, tandis que les auteurs de [Guarnieri and Balmes, 2014] établissent un lien entre l'asthme et la pollution de l'air en milieu urbain.

- Données sur les trajets quotidiens: Enfin, les données qui nous intéressent le plus sont les données sur les trajets en transports en com-

mun. Il existe beaucoup de documentation sur les études relatives à ce type de données. Sur le territoire français du Val d'Amboise, les déplacements intermodaux ont été étudiés et ces données ont permis de faire une comparaison économique entre un service de combinaison vélo/train et un service de combinaison parking/train [Papon et al., 2017]. Plusieurs articles traitent des données de systèmes de vélo en libre-service [Côme and Oukhellou, 2014; Bouveyron et al., 2015] afin de comprendre les relations entre le type de quartier et le modèle de mobilité y étant le plus courant afin d'attribuer une fonction à chaque zone. Dans [Briand et al., 2017; El Mahrsi et al., 2017], les auteurs regroupent les données des cartes à puce afin de créer des groupes de passagers ayant un comportement temporel similaire et des groupes de stations ayant le même type d'utilisation. Afin d'être capable d'identifier les pickpockets, [Du et al., 2018] détecte les enregistrements de transit quotidiens inhabituels. Dans [Toqué et al., 2017], les auteurs utilisent les données des cartes à puce pour prévoir la demande de mobilité à court (15 à 30 minutes) et long (1 an) terme dans le quartier de La Défense à Paris.

Il existe une grande diversité de données urbaines, et les différents auteurs de [De Palma and Dantan, 2017] abordent les défis que pose le traitement d'une telle variété de données. En effet, même si ce n'est pas le sujet de cette thèse, l'accumulation de telles données personnelles pose plusieurs problèmes. Ces problèmes éthiques concernent principalement la sécurité des données [Hardt et al., 2016], leur anonymisation [Gadouche and Picard, 2017] et leur non-utilisation à des fins de discrimination [Abadi et al., 2016; Ji et al., 2014].

Dans cette thèse, menée au CREST grâce à un financement de Transdev, nous avions pour objectif de proposer des avancées sur l'analyse de certains types de données décrites précédemment et notamment sur les données de transport.

1.1.4 Transdev

Transdev est un opérateur de transport français opérant à l'international. La société a été créée en 2011, d'abord sous le nom de Veolia Transdev, en fusionnant Veolia Transport (de Veolia) et Transdev (de la Caisse des Dépôts et Consignations). Au moment de la rédaction du présent rapport, ces sociétés restent les principaux actionnaires de Transdev, mais Veolia a annoncé son intention de céder ses parts au groupe Rethmann avant fin 2018 [Trompiz and Mazzilli, 2018]. En exploitant des bus, tramways,

ferries, taxis, lignes d'autocar, trains, navettes, services médicaux, services scolaires et véhicules autonomes dans 20 pays, Transdev transporte 11 millions de passagers par jour.

Table 1.1: Transdev en quelques chiffres

Création	2011
Employés	82000
Passagers quotidiens	11 millions
Pays avec des réseaux exploités par Transdev	États-unis, Canada, Chili, Colombie, Maroc, France, Allemagne, République tchèque, Finlande, Irlande, Espagne, Royaume-Uni, Suède, Pays-Bas, Portugal, Inde, Corée du Sud, Chine, Australie, Nouvelle-Zélande

La quantité de données générées par ces opérations est énorme. En effet, les données billettiques, de ventes, de ressources humaines et de maintenance sont enregistrées quotidiennement. Ces données permettent à l'entreprise de suivre des phénomènes qui n'étaient pas suivis jusqu'à récemment. Par exemple, l'apparition des cartes à puce a offert la possibilité de suivre les achats de produits des clients et les habitudes des passagers, ce qui pourrait révolutionner les stratégies marketing. Si les ressources humaines enregistrent l'absentéisme et les services des conducteurs, il pourrait être facile de déterminer les facteurs à l'origine de longs congés maladie. Enfin, en anticipant l'usure des pièces des véhicules, les pannes pourraient être évitées.

Et pourtant, toutes ces nouvelles possibilités ne peuvent être atteintes que si l'on dispose des connaissances théoriques adéquates et si l'on utilise des méthodologies et des technologies appropriées. L'objectif de cette thèse est donc d'établir des méthodes utiles pour l'analyse et la valorisation des données billettiques.

Au cours de ce travail de thèse, nous avons reçu des données billettiques de plusieurs réseaux. Dans [Pelletier et al., 2009], Les données billettiques sont décrites comme les données "stockées à chaque validation: date et heure de la validation, état de la transaction, numéro de carte, type de

tarif, numéro d'itinéraire, itinéraire, direction, numéro d'arrêt, numéro de bus, numéro de conducteur, numéro de course, et numéro de base de données interne". La structure des bases de données peut différer selon le réseau, mais ces bases ne contiennent toujours que des données de cartes à puce. Les validations effectuées par billets magnétiques ne sont pas consignées dans des bases de données.

1.2 Résumé substantiel des chapitres

L'apprentissage statistique est, selon la définition de [Friedman et al., 2001], le principe de l'apprentissage à partir de données grâce à des méthodes et modèles statistiques. Ces données sont composées de variables observées qui sont soit quantitatives (le nombre de passagers d'un autobus par exemple), soit catégoriques (comme le jour de la semaine). Pour les problèmes d'apprentissage supervisé (la différence entre les problèmes supervisés et non supervisés est détaillée dans la sous-sous-section 1.2.1.1), la base de données contient une variable cible (celle que nous voulons prédire) et un ensemble de variables explicatives. Généralement, les données sont séparées en deux échantillons. L'échantillon d'apprentissage aide à construire un modèle de prédiction, qui est appliqué à l'échantillon de test pour mesurer la capacité de généralisation du modèle sur des observations nouvelles et jusqu'alors non connues du modèle.

La plupart des questions d'apprentissage statistique supervisé sont classées en deux grandes catégories : les problèmes de classification et les problèmes de régression. Le but de ces problèmes est d'expliquer la valeur de la variable cible grâce aux caractéristiques. Si la cible est une variable quantitative, on parle de régression (prévision du nombre de passagers d'un autobus par exemple), alors que si elle est catégorique, on parle de classification (comme détecter si un achat est frauduleux). La classification tend à regrouper les observations en fonction des valeurs cibles grâce aux variables explicatives.

Dans l'apprentissage non supervisé, on analyse des données sans variable cible. Il ne s'agit donc pas de prédire une valeur grâce aux variables explicatives, mais d'y détecter des tendances (détecter des groupes de passagers avec les mêmes heures de voyage par exemple). La technique non supervisée la plus utilisée s'appelle le clustering. Cette technique regroupe des observations présentant des caractéristiques similaires.

1.2.1 Segmentation

1.2.1.1 Définition

Comme expliqué brièvement ci-dessus, il existe deux types de classifications. Nous parlons de classification supervisée lorsque les données sont étiquetées et l'algorithme doit discriminer les données sur les variables explicatives pour expliquer la différence entre les étiquettes. Dans la classification non supervisée – ou clustering, comme décrit par [Sathya and Abraham, 2013], l'objectif est d'"identifier des régularités cachées des données sans étiquette". L'autre différence entre les deux techniques est que dans la classification, le nombre de classes est fixe – c'est le nombre de catégories de la variable cible, tandis que le nombre optimal de groupes dans la segmentation est inconnu *a priori*.

Au cours de ce travail de doctorat, nous nous sommes concentrés uniquement sur un problème de segmentation et non pas de classification. Ce type de problème est principalement résolu par deux approches différentes. Certains algorithmes – appelés algorithmes sans modèle – n'ont besoin d'aucune hypothèse sur les données, alors que les algorithmes basés sur des modèles ont toujours besoin d'une hypothèse sur la structure des données à traiter. Pour illustrer nos propos, nous utiliserons un exemple-jouet de données simulées tout au long de cette sous-section. Ces données, représentées sur la Figure 1.3, sont une simulation d'un mélange de deux distributions gaussiennes bivariées. La première distribution – représentée par des points orange – compte 50 observations d'usagers non abonnés, tandis que la seconde – avec des points bleus – compte 30 observations d'usagers abonnés. L'abscisse représente l'âge de l'utilisateur, et l'ordonnée le nombre de validations effectuées au cours d'un mois.

1.2.1.2 Algorithmes ne reposant pas sur un modèle

Il existe plusieurs algorithmes ne reposant pas sur un modèle qui nous permettent de procéder à un clustering. Nous en présenterons quelques-uns, mais les paragraphes suivants ne donnent pas une liste exhaustive de ces algorithmes.

L'algorithme de clustering le plus intuitif est la classification ascendante hiérarchique (CAH), introduite dans [Johnson, 1967]. Cette méthode est basée sur $D \in \mathbb{R}^{n \times n}$ la matrice de dissimilarité, telle que $\forall i \neq j \; D_{i,j} = D_{j,i} = d(C_i, C_j)$, où $d(C_i, C_j)$ est une fonction calculant la distance entre les $i^{\text{ème}}$ et $j^{\text{ème}}$ groupes. Son principe est de trouver à chaque itération la plus petite distance contenue dans la matrice D et de regrouper les clusters

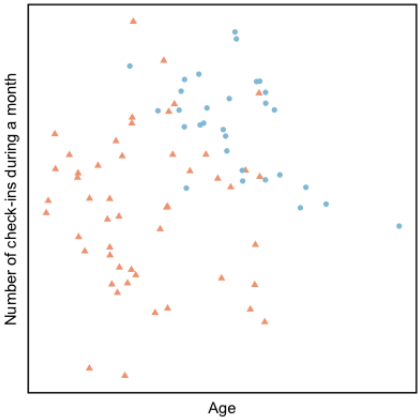
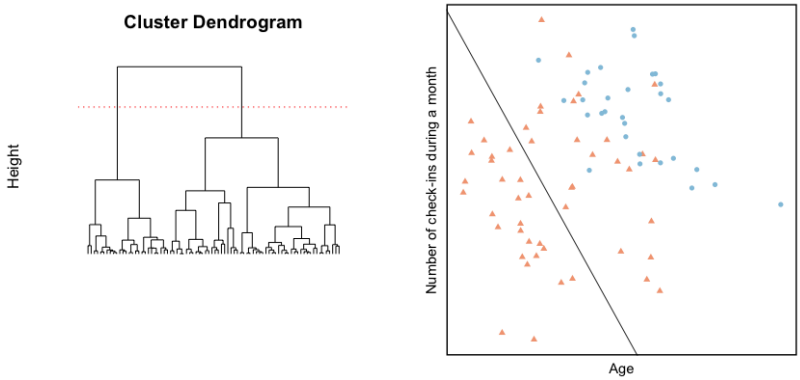


Figure 1.3: Données simulées comme un mélange de deux distributions gaussiennes bivariées. Il y a 50 observations oranges et 30 bleues.

y correspondant en un seul nouveau cluster, puis de calculer la nouvelle matrice de dissimilarité entre l'ensemble de ces clusters. L'algorithme s'arrête une fois que toutes les observations sont contenues dans un seul groupe. L'information de segmentation est alors contenue dans un dendrogramme, tel que celui de la Figure 1.4a. Pour trouver le nombre optimal de groupes, on regarde la plus grande hauteur entre deux groupes consécutifs sur le dendrogramme. La Figure 1.4b montre la partition de l'espace de nos données jouets. Bien que cette méthode soit facile à mettre en œuvre, elle peut facilement être coûteuse pour les données de grande dimension. En effet, à chaque itération, la matrice de dissimilarité est recalculée.

Le second algorithme le plus courant est la méthode k-means. Il s'agit d'une méthode itérative en deux étapes, dont le concept a été proposé en 1956 par [Steinhaus, 1956], développé par [MacQueen, 1967] et dont [Selim and Ismail, 1984] a prouvé la convergence. Étant donné K centres de groupes, l'algorithme calcule la distance entre chaque observation et chaque centre puis attribue l'observation au centre le plus proche. Le nouveau centre est alors calculé et l'algorithme s'arrête lorsque la convergence est atteinte. L'algorithme détaillé est décrit dans l'Algorithme 1. Pour sélectionner le nombre optimal de groupes K, plusieurs méthodes tendent à minimiser l'inertie intra-groupe tout en maximisant l'inertie inter-groupe. La méthode des écarts est la plus courante et a été introduite par [Tibshi-



(a) Dendrogramme de la CAH sur les données jouets

(b) Partition de l'espace par CAH

Figure 1.4: Classification ascendante hiérarchique

rani et al., 2001]. Mathématiquement, cela se traduit par:

$$K^* = \underset{k}{\operatorname{argmax}}\, \mathbb{E}_n^* \left[\log \left(\sum_{r=1}^{k} \frac{1}{2n_r} \sum_{i,i' \in C_r} d_{i,i'} \right) \right] - \log \left(\sum_{r=1}^{k} \frac{1}{2n_r} \sum_{i,i' \in C_r} d_{i,i'} \right),$$

où $\mathbb{E}_n^*[\cdot]$ est l'espérance sous l'hypothèse qu'il n'y a aucun cluster dans les données, C_r est le $r^{\text{ème}}$ cluster, n_r son nombre d'observations et $d_{i,}$ la distance entre les $i^{\text{ème}}$ et $i'^{\text{ème}}$ observations. Nous avons appliqué cet algorithme à nos données, et le résultat est sur la Figure 1.5.

La segmentation spectrale a été introduite plus récemment, et de nombreux chercheurs se sont penchés théoriquement sur le sujet [Ng et al., 2002; Von Luxburg, 2007; Zelnik-Manor and Perona, 2005; Dhillon et al., 2004; Filippone et al., 2008; Stella and Shi, 2003]. Le principe est d'effectuer une segmentation sur les principaux vecteurs propres K de la transformation de la matrice de similarité des données. Cette méthode permet de détecter des structures de données plus complexes, comme les groupes non convexes tels que ceux qui apparaissent sur la Figure 1.6a. La Figure 1.6b montre l'application de la segmentation spectrale sur nos données-jouets avec une fonction de noyau polynomiale.

Plusieurs méthodes de segmentation sont basées sur la factorisation de matrices. La factorisation de matrices est le principe d'estimer les valeurs

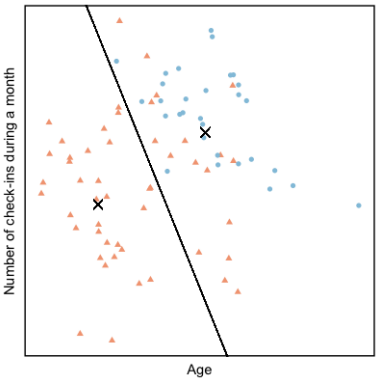
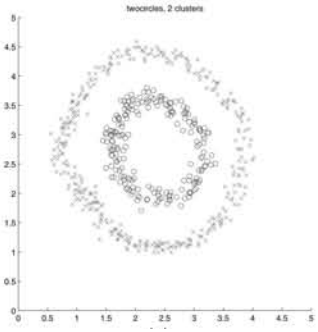
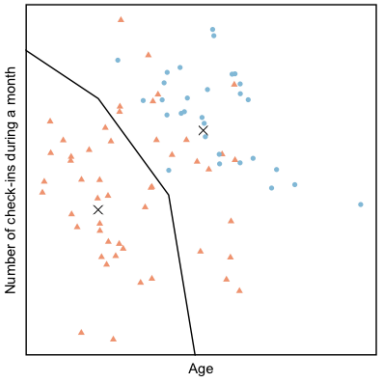


Figure 1.5: Cloisonnement de l'espace par k-means. Les centres sont représentés par des croix noires.



(a) Segmentation spectrale sur une structure de données complexe. Graphique de [Ng et al., 2002]



(b) Partition de l'espace par segmentation spectrale

Figure 1.6: Exemples de segmentation spectrale.

Algorithm 1 k-means

Choisir arbitrairement $m_1^{(0)}, \ldots, m_K^{(0)}$, K centres parmi les observations.
$t = 0$
while Le critère d'arrêt n'a pas été atteint **do**
  for $i \in \{1, \ldots, n\}$ **do**
    for $k \in \{1, \ldots, K\}$ **do**
      $d_{i,k} = ||x_i - m_k^{(t)}||$
    end for
    $c_i^{(t)} \leftarrow \text{argmax}_k \, d_{i,k}$
  end for
  $N_k^{(t)} = \sum_{i=1}^{n} \mathbb{1}_{[c_i^{(t)}=k]}$
  $m_k^{(t+1)} = \frac{1}{N_k^{(t)}} \sum_{i \in \mathcal{C}_k^{(t)}} x_i$
  $t = t + 1$
end while

d'une matrice $X \in \mathbb{R}^{n \times m}$ par au moins deux matrices:

$$X \approx UV$$

avec $U \in \mathbb{R}^{n \times p}$ et $V \in \mathbb{R}^{p \times m}$ et $p \ll n, m$. La factorisation de matrices est principalement utilisée pour réduire la dimension des données afin de les stocker (en analyse numérique) ou d'estimer les valeurs d'une matrice (en statistiques). L'analyse en composantes principales (ou ACP) est l'une des méthodes statistiques les plus connues et repose sur la factorisation matricielle. En effet, elle a été créée au début du XX$^{\text{ème}}$ siècle par [Pearson, 1901] pour représenter de grands ensembles de données en deux ou trois dimensions en créant des composantes et axes principaux sur lesquels les données seraient projetées. L'objectif est de trouver une nouvelle base permettant de mettre en évidence la variabilité des données. Comme [Wold et al., 1987; Mackiewicz and Ratajczak, 1993] le décrit, une composante principale est une combinaison linéaire des variables des données. En effet, mathématiquement, nous avons:

$$X = WU,$$

où $X \in \mathbb{R}^{n \times m}$ est la matrice de données contenant n observations de m variables, $W \in \mathbb{R}^{n \times p}$ est la projection des n observations sur les p

composantes principales et $U \in \mathbb{R}^{p \times m}$ la matrice contenant l'expression des p composantes principales en fonction des m variables. Les colonnes de U sont orthogonales tandis que celles de W sont orthonormées.

Nous pouvons également interpréter la méthode k-means – décrite plus haut – comme une méthode de factorisation de matrices. En effet, étant donné:

$$X \approx CM,$$

où $X \in \mathbb{R}^{n \times m}$ est la matrice de données des n observations en m dimensions, $C \in \{0,1\}^{n \times K}$ donne l'allocation des observations dans les groupes, tel que $C_{i,k} = \mathbb{1}_{[x_i \in \mathcal{C}_k]}$ et $M \in \mathbb{R}^{K \times m}$ contient les centres des groupes. De cette façon, les observations sont estimées par le centre du groupe auquel elles appartiennent.

La factorisation de matrices non négatives (ou NMF pour Nonnegative Matrix Factorization en anglais), comme son nom l'indique, est une méthode de factorisation de matrices proposée par [Lee and Seung, 1999]. Dans cet article, la méthode est présentée comme un outil de réduction de la dimension pour les matrices avec des entrées non négatives. En effet, si $\theta \approx \Phi\Lambda$ avec $\theta \mathbb{R}_+^{n \times M}$, $\Phi \in \mathbb{R}_+^{n \times H}$, $\Lambda \in \mathbb{R}_+^{H \times M}$ et $H \ll n, M$, alors le nombre d'entrées dans θ est beaucoup plus grand que ceux dans Φ et Λ : $Hn + HM \ll nM$. Plusieurs algorithmes existent pour résoudre ce problème d'optimisation, mais l'algorithme multiplicatif – décrit dans Algorithm 5 – est le plus courant.

Algorithm 2 Algorithme multiplicatif pour la NMF

1: Fixer $\varepsilon > 0$, choisir arbitrairement $\Lambda^{(0)}$ et $\Phi^{(0)}$ non-négatives.
2: $t = 0$
3: **while** $||\theta - \Phi^{(t)}\Lambda^{(t)}|| > \varepsilon$ **do**
4: $\quad \forall h,k \quad \Lambda_{h,k}^{(t+1)} \leftarrow \Lambda_{h,k}^{(t)} \frac{(\Phi^{T(t)}\theta)_{h,k}}{(\Phi^{T(t)}\Phi^{(t)}\Lambda^{(t)})_{h,k}}$
5: $\quad \forall h,k \quad \Lambda_{h,k}^{(t+1)} \leftarrow \frac{\Lambda_{h,k}^{(t+1)}}{\sum_{k'} \Lambda_{h,k'}^{(t+1)}}$
6: $\quad \forall j,h \quad \Phi_{j,h}^{(t+1)} \leftarrow \Phi_{j,h}^{(t)} \frac{(\theta\Lambda^{T(t+1)})_{j,h}}{(\Phi_{j,h}^{(t)}\Lambda^{(t+1)}\Lambda^{T(t+1)})_{j,h}}$
7: $\quad \forall j,h \quad \Phi_{j,h}^{(t+1)} \leftarrow \frac{\Phi_{j,h}^{(t+1)}}{\sum_{h'} \Phi_{j,h'}^{(t+1)}}$
8: $\quad t = t + 1$
9: **end while**

1.2.1.3 Contribution principale du chapitre 3

La problématique de ce chapitre est de trouver un moyen de regrouper les voyageurs par leurs habitudes temporelles, et d'identifier les passagers ayant des habitudes similaires. Dans [El Mahrsi et al., 2014b], les auteurs proposent un mélange de k distributions multinomiales comme modèle pour regrouper les profils temporels des voyageurs. Ils estiment ensuite les paramètres des différentes distributions et assignent les voyageurs à ces groupes en utilisant l'algorithme d'Espérance-Maximisation (EM).

Bien que les résultats obtenus permettent d'identifier des profils d'utilisateurs pertinents, certains groupes ne sont pas faciles à interpréter. Pour surmonter ce problème, nous proposons de réduire la dimension des profils par NMF. La NMF conduit dans de nombreuses applications en haute dimension à la définition d'un dictionnaire parcimonieux et facilement interprétable: [Lee and Seung, 1999] fournit des exemples en analyse d'images, [Xu et al., 2003; Shahnaz et al., 2006] en segmentation de textes, entre autres. Ici, la NMF fournit un dictionnaire des profils typiques, et une projection de chaque profil dans l'étendue de ce dictionnaire. Comme les profils typiques sont contenus dans un dictionnaire, nous les appelons des mots.

N'importe quelle méthode de segmentation peut alors être utilisée dans cet espace réduit. Nous choisissons d'appliquer une méthode k-means à cet espace plus petit pour obtenir nos groupes. Cela a conduit à des groupes facilement interprétables. Deux des mots et un des clusters obtenus avec cette méthodologie sont représentés sur la Figure 1.7. Nous voyons clairement que le groupe représenté est une combinaison linéaire des deux mots montrés ici.

1.2.1.4 Algorithmes basés sur des modèles - Modèles de mélange

Parmi les algorithmes de segmentation, le modèle de mélange est l'un des outils basés sur des modèles les plus courants. Son principe est basé sur l'hypothèse que dans une population, "les individus peuvent souvent être divisés en sous-groupes" [Benaglia et al., 2009]. Par conséquent, la population n'est plus décrite par une seule distribution, mais par un mélange de K distributions. Étant donné une famille paramétrique de distributions $(f_\vartheta)_{\vartheta \in \mathbb{R}^M}$, supposons que les observations $Y_1, \ldots, Y_n$ sont indépendantes et identiquement distribuées, alors:

$$f(\cdot) = \sum_{k=1}^{K} p_k f_{\theta_{\cdot,k}}(\cdot), \tag{1.1}$$

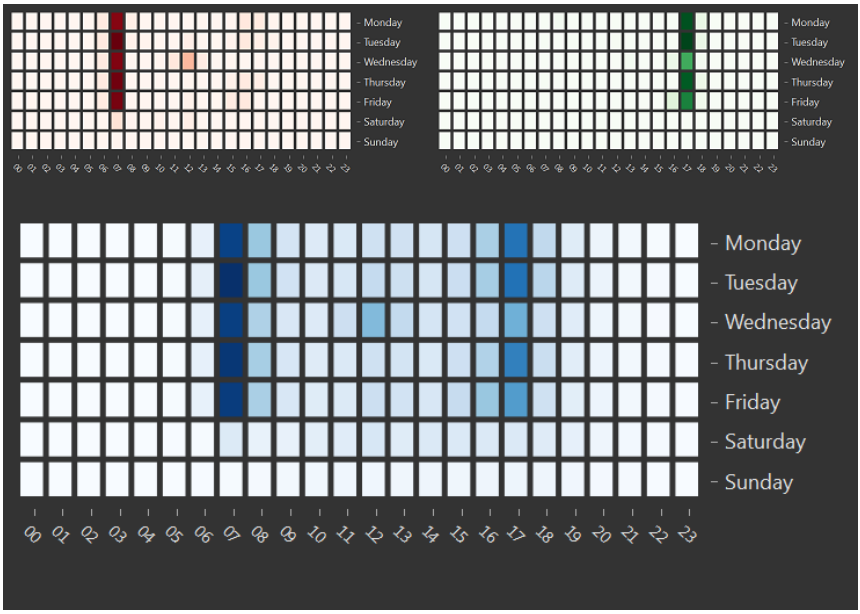


Figure 1.7: Deux mots et un groupe obtenus avec la méthodologie du chapitre 3

où chaque $\theta_{\cdot,k} \in \mathbb{R}^M$ est une colonne d'une matrice θ de taille $K \times M$ contenant les paramètres de la $k^{\text{ème}}$ distribution, et $p = (p_1, \ldots, p_K)$ est le vecteur contenant la probabilité pour une observation choisie aléatoirement d'être présente dans chaque distribution. L'auteur de [Bishop, 2007] explique plus précisément le cas des mélanges gaussiens. Il existe de nombreuses façons d'estimer la matrice θ, et [Scrucca et al., 2016] passe en revue les différents packages `R` implémentés pour estimer cette dernière, et nous en citerons quelques unes ci-dessous.

En biostatistique, plusieurs articles traitent du modèle bayésien de mélange [Lartillot and Philippe, 2004; Medvedovic et al., 2004; Do et al., 2005]. Le principe de cette méthode est d'avoir une hypothèse sur les paramètres θ des différentes distributions. Ces hypothèses sont contenues dans le prior:

$$\pi(\theta, p) = \pi_p(p) \times \prod_{k=1}^{K} \pi_k(\theta_{\cdot,k}),$$

où $\pi_p(p)$ est l'hypothèse sur la valeur du vecteur p et $\pi_k(\theta_{\cdot,k})$ est l'hypothèse sur les paramètres de la $k^{\text{ème}}$ distribution. Par conséquent, l'estimation des paramètres est donnée par le postérieur :

$$\pi_n(\theta, p|X) = \frac{\pi(\theta, p)L(\theta, p|X)}{\displaystyle\int \pi(\phi)L(\phi|X)},$$

où $L(\theta, p|X)$ est la vraisemblance de θ et p lorsque $X = (X_1, \ldots, X_n)$ est connu. Des études théoriques de l'approche bayésienne peuvent être trouvées dans [Ghosal et al., 2000; Chérief-Abdellatif and Alquier, 2018] par exemple. Nous avons appliqué un algorithme EM bayésien pour modèle de mélange à nos données et le résultat est illustré à la Figure 1.8.

Généralement, pour estimer les paramètres d'une distribution à partir d'observations, la façon la plus simple et la plus connue est de trouver les valeurs maximisant la vraisemblance, appelés estimateurs du Maximum de Vraisemblance. Voici la vraisemblance d'un modèle de mélange:

$$L(\theta, p|X) = \prod_{i=1}^{n} \sum_{k=1}^{K} p_k f_{\theta_{\cdot,k}}(x_i),$$

et sa log-vraisemblance:

$$\ell(\theta, p|X) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} p_k f_{\theta_{\cdot,k}}(x_i) \right).$$

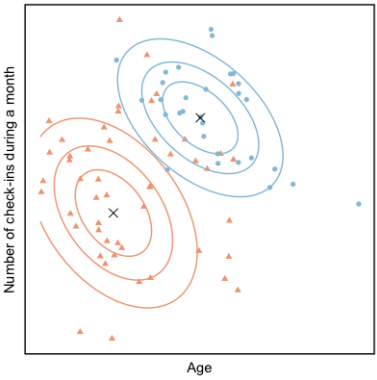


Figure 1.8: Densité des distributions trouvées par un algorithme bayésien EM pour un modèle de mélange de gaussiennes. Les moyennes postérieures des distributions sont représentées par des croix noires. Nous avons utilisé les paramètres de la simulation comme priors.

Il est assez évident qu'en raison de la forme complexe de la fonction, cette log-vraisemblance est impossible à maximiser. Comme nous ne savons pas quelles observations ont été générées par quelles distributions, l'algorithme EM propose d'introduire une variable latente $Z_{i,k}$ contenant cette information, telle que $Z_{i,k} = \mathbb{1}_{[X_i \text{ générée par la } k^{\text{ème}} \text{ distribution}]}$. Ainsi, la log-vraisemblance complétée devient:

$$\ell(\theta, p|X, Z) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k} \log\left(p_k f_{\theta_{\cdot,k}}(x_i)\right).$$

EM est un algorithme récursif, et étant donné les paramètres actuels $(\theta^{(c)}, p^{(c)})$ une itération est:

$$\begin{aligned}
\textbf{E-step}\text{: } Q^{(c)}(\theta, p) &= \mathbb{E}_{\theta^{(c)},p^{(c)}}[\ell(\theta, p|X, Z)|X] \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} \mathbb{E}_{\theta^{(c)},p^{(c)}}[Z_{i,k}|X] \log\left(p_k f_{(\theta)_{\cdot,k}}(X_i)\right) \\
\text{and } t_{i,k}^{(c)} &:= \mathbb{E}_{\theta^{(c)},p^{(c)}}[Z_{i,k}|X] \\
&= \frac{p_k^{(c)} f_{(\theta^{(c)})_{\cdot,k}}(X_i)}{\sum_{k'=1}^{K} p_{k'}^{(c)} f_{(\theta^{(c)})_{\cdot,k'}}(X_i)}.
\end{aligned} \tag{1.2}$$

$$\textbf{M-step}: \ (\theta^{(c+1)}, p^{(c+1)}) := \underset{\theta_{j,k} \geq 0}{\arg\max} Q^{(c)}(\theta, p), \tag{1.3}$$

avec évidemment pour $k \in \{1, \ldots, K\}$:

$$p_k^{(c+1)} = \frac{\sum_{i=1}^{n} t_{i,k}^{(c)}}{\sum_{i=1}^{n} \sum_{k'=1}^{K} t_{i,k'}^{(c)}}. \tag{1.4}$$

Bien sûr cet algorithme dépend de la famille de distributions $(f_\vartheta)_{\vartheta \in \mathbb{R}^M}$. Un exemple est donné ci-dessous dans le cas d'un mélange de distributions multinomiales. La Figure 1.9 montre l'application d'un algorithme EM pour un modèle de mélange de gaussiennes à nos données jouets.

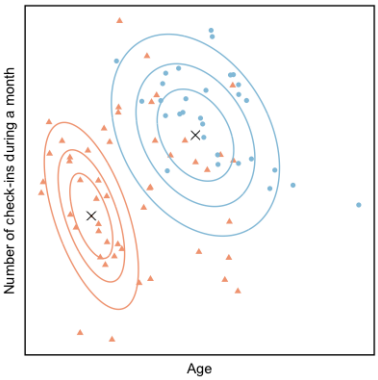


Figure 1.9: Densité des distributions trouvées par l'algorithme EM pour le modèle de mélange de gaussiennes. Les moyennes des distributions sont représentées par des croix noires.

Dans le domaine des transports, certaines études ont été menées avec des segmentations pour les modèles de mélange. En effet, les auteurs de [Côme and Oukhellou, 2014; Randriamanamihaga et al., 2013] utilisent des modèles de mélange de Poisson pour regrouper les stations de systèmes de vélos en libre-service, alors que dans le même ordre d'idées [Bouveyron et al., 2015] utilise un modèle fonctionnel pour regrouper la suite temporelle des occupations de la station. L'article qui nous intéresse le plus est [El Mahrsi et al., 2014a]. En effet, dans cet article, les auteurs utilisent un modèle de mélange de multinomiales pour regrouper les passagers sur

leurs profils temporels dans le réseau STAR de Rennes, en France. Un profil temporel compte le nombre de validations qui ont eu lieu à chaque jour de la semaine et à chaque heure pendant la période étudiée, comme le montre la Figure 1.10. Il est évident que le profil u_i du passager i suit

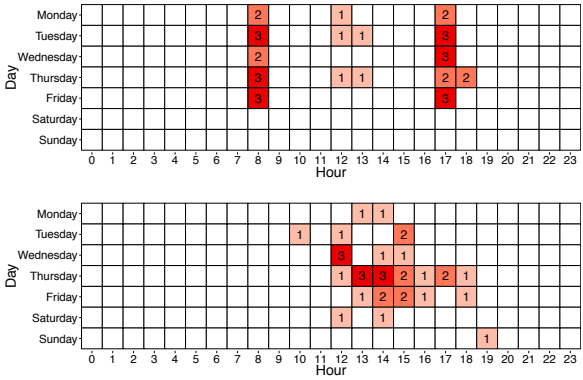


Figure 1.10: Profils temporels de deux passagers – Figure issue de [El Mahrsi et al., 2014a].

une distribution multinomiale:

$$u_i \sim \mathcal{M}(D_i, \pi_i),$$

où D_i est le nombre total de validations effectuées par le passager i pendant la période d'étude et $\pi_i = (\pi_{i,1}, \ldots, \pi_{i,M})$ est le simplex contenant la probabilité des M événements. C'est simplement la probabilité qu'une validation choisie aléatoirement du passager i se produise à chaque instant $m \in \{1, \ldots, M\}$. Dans l'hypothèse où l'ensemble des profils est la réalisation d'un mélange de K distributions, chaque groupe β_k suit une distribution multinomiale telle que:

$$\beta_k \sim \mathcal{M}(1, \pi_k).$$

Et le profil temporel du passager i suit :

$$u_i | z_i \sim \mathcal{M}(D_i, \pi_k),$$

avec $z_i \in \{\beta_1, \ldots, \beta_K\}$. Ainsi, l'équation de mélange 1.1 devient:

$$f(X_i) = \sum_{k=1}^{K} p_k D_i! \prod_{j=1}^{M} \frac{\pi_{k,j}^{X_{i,j}}}{X_{i,j!}}. \tag{1.5}$$

L'algorithme détaillé est ainsi décrit dans l'Algorithme 3. Par souci de clarté, supposons que $\Pi = (\pi_{k,j})_{1 \leq k \leq K, 1 \leq j \leq M}$.

Algorithm 3 Algorithme EM pour un modèle de mélange multinomial

1: Fixer $\varepsilon > 0$, choisir arbitrairement $\Pi^{(0)}$ et $p^{(0)}$; CRIT $\leftarrow \infty$
2: $c \leftarrow 0$
3: **while** $|\ell(\Pi^{(c)}, p^{(c)}) - \text{CRIT}| > \varepsilon$ **do**
4: CRIT $\leftarrow \ell(\Pi^{(c)}, p^{(c)})$
5: $\forall i, k \quad t_{i,k}^{(c)} \leftarrow \frac{\pi_k^{(c)} \prod_{j=1}^{M} \pi_{j,k}^{X_{i,j}}}{\sum_{k'=1}^{K} \pi_{k'}^{(c)} \prod_{j=1}^{M} \pi_{j,k'}^{X_{i,j}}}$
6: $\forall k \quad p_k^{(c+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} t_{i,k}^{(c)}$
7: $\forall k, j \quad \pi_{j,k}^{(c+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} t_{i,k}^{(c)} X_{i,j}$
8: $c \leftarrow c + 1$
9: **end while**

1.2.1.5 Contribution principale du chapitre 4

Dans le chapitre 3, la NMF était un bon outil de prétraitement pour segmenter les profils temporels des passagers. Cependant, nous avons noté que nous avons utilisé une procédure en deux étapes et que nous voulions définir une procédure en une seule étape qui permettrait d'estimer directement un dictionnaire optimisant un critère lié à l'objectif de segmentation.

De plus, l'article [El Mahrsi et al., 2014a] propose une méthode basée sur un modèle pour une question similaire. Dans cet article, nous avons remarqué que la matrice des paramètres à estimer peut facilement être lourde. En effet, ils ont besoin d'estimer les paramètres de vingt-quatre heures, pour sept jours et seize groupes, soit 2688 paramètres.

C'est pourquoi nous proposons dans le chapitre 4 un nouvel algorithme appelé NMF-EM qui permet la réduction de dimension et la segmentation simultanément. Pour ce faire, nous suggérons d'appliquer une NMF à la matrice des paramètres à estimer par EM. Le nombre de paramètres à estimer sera alors largement inférieur. En effet, si $\theta \in \mathbb{R}_+^{M,K}$ est la matrice contenant en colonne les paramètres non négatifs des distributions K de la même famille de distribution $(f_\vartheta)_{\vartheta \in \mathbb{R}M}$, nous avons:

$$\theta = \Phi\Lambda,$$

avec $H \leq K, M$, $\Phi \in \mathbb{R}_+^{M,H}$ et $\Lambda \in \mathbb{R}_+^{H,K}$. Dans notre cas, nous appliquons l'algorithme NMF-EM pour un mélange de multinomiales. Ainsi, le modèle 1.5 devient:

$$f(X_i|Z_{i,k}=1) = \sum_{k=1}^{K} p_k D_i! \prod_{j=1}^{M} \frac{(\Phi\Lambda)_{j,k}^{X_{i,j}}}{X_{i,j!}},$$

et la log-vraisemblance complétée:

$$\ell(\Phi, \Lambda, p|X) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k} \log\left(p_k \left[N_i! \prod_{j=1}^{M} \frac{(\Phi\Lambda)_{j,k}^{X_{i,j}}}{X_{i,j}!}\right]\right).$$

Ainsi, à partir de l'Algorithme 3, la mise à jour des paramètres devient :

$$t_{i,k}^{(c)} := \frac{p_k^{(c)} f_{(\Phi^{(c)}\Lambda^{(c)})_{\cdot,k}}(Y_i)}{\sum_{k'=1}^{K} p_{k'}^{(c)} f_{(\Phi^{(c)}\Lambda^{(c)})_{\cdot,k'}}(Y_i)},$$

$$(\Phi^{(c+1)}, \Lambda^{(c+1)}, p^{(c+1)}) := \underset{\Phi_{j,h}, \Lambda_{h,k} \geq 0}{\arg\max}\, Q^{(c)}(\Phi, \Lambda, p).$$

Cette mise à jour des paramètres est obtenue en injectant l'algorithme multiplicatif de NMF (Algorithme 2) dans l'algorithme EM. L'application de cet algorithme à des données réelles a donné de bons résultats empiriques. En effet, nous avons prouvé que pour des clusters colinéaires, notre algorithme surpasse à la fois les algorithmes EM et k-means.

De plus, grâce à NMF-EM, nous avons obtenu cinq mots et dix clusters avec nos données et la Figure 1.11 montre deux des mots et un des clusters. Dans une deuxième phase, nous avons analysé socio-économiquement ces clusters et obtenu des informations intéressantes sur les passagers du réseau étudié.

1.2.2 Régression et Prévision

1.2.2.1 Définition

La prédiction et la régression peuvent facilement être confondues. Comme décrit dans [oneclick.ai, 2018], la principale différence est que "la régression prédit la valeur d'un nombre dans un événement futur hypothétique" tandis que "la prévision utilise une chronologie compilée des moments de données pour ensuite dire comment elle se poursuivra dans le futur selon

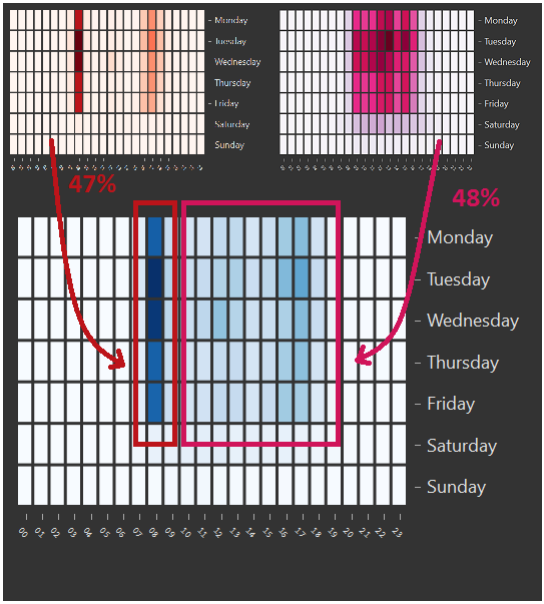


Figure 1.11: Décomposition d'un cluster par deux mots. This cluster est composé à 47% du mot en orange et 48% du mot en rose. Les 5% restants sont un mélange des trois autres mots.

ces tendances". Nous pouvons résumer cela en disant que toutes les régressions ne sont pas des prévisions, mais que toutes les prévisions sont des régressions. Dans les parties suivantes de cette section, nous ne traiterons que des méthodes de régression, mais celles-ci sont également pertinentes pour la prévision.

La régression peut être décrite comme un processus permettant de prédire une variable cible quantitative Y à partir d'un ensemble de m variables explicatives $X = (X_1, \ldots, X_m)$. Les méthodes de régression permettent d'estimer une fonction $f(\cdot)$ telle que $Y = f(X) + \varepsilon$, où ε est la différence entre la valeur observée Y et la valeur prédite $\hat{Y} = f(X)$. Plus ε est petit, plus la fonction de régression $f(\cdot)$ est précise.

1.2.2.2 Méthodes linéaires

Les méthodes linéaires sont parmi les modèles de régression les plus simples. La plus célèbre est la régression linéaire. En effet, ce modèle est basé sur l'hypothèse qu'il existe des liens linéaires entre les variables explicatives $(X_1, \ldots, X_m)$ et la variable cible Y comme expliqué dans [Seber, 2009; Weisberg, 2005; Bibby et al., 1979]. Ainsi, pour une population de n individus, et $\forall i \in \{1, \ldots, n\}$:

$$Y_i = \beta_0 + \sum_{j=1}^{m} \beta_j X_{i,j} + \varepsilon_i,$$

où $\beta = (\beta_0, \beta_1, \ldots, \beta_m) \in \mathbb{R}^{m+1}$ est le vecteur contenant les coefficients de régression, et $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ sont indépendants et distribués de manière identique. La façon la plus populaire d'effectuer une régression linéaire et d'estimer le vecteur β est d'utiliser la méthode des moindres carrés ordinaires (MCO), qui donne $\hat{\beta} = (X^t X)^{-1} X^t Y$.

Dans le cas d'un problème mal conditionné (i.e. matrice X avec colonnes colinéaires), mais en voulant conserver chaque variable pour des raisons d'interprétation, on peut utiliser un estimateur biaisé pour améliorer les propriétés numériques et la variance des estimations par un processus de normalisation. Des preuves théoriques de son efficacité peuvent être trouvées dans [Le Cessie and Van Houwelingen, 1992; Marquardt and Snee, 1975]. Pour cela, nous devons formaliser le modèle sous sa forme matricielle. Ainsi, le modèle ridge proposé par [Hoerl and Kennard, 1970]

est:

$$\underbrace{\begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}}_{Y} = \underbrace{\begin{pmatrix} 1 & X_{1,1} & \dots & X_{1,m} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n,1} & \dots & X_{n,m} \end{pmatrix}}_{X} \underbrace{\begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_m \end{pmatrix}}_{\beta} + \underbrace{\begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}}_{\varepsilon}.$$

Dans la régression ridge, l'estimateur $\hat{\beta}$ est donné par:

$$\hat{\beta}_{\text{ridge}} = \underset{\beta \in \mathbb{R}^{m+1}}{\operatorname{argmin}} \left(\sum_{i=1}^{n} (Y_i - \sum_{j=0}^{m} X_{i,j}\beta_j)^2 + \lambda \sum_{j=1}^{m} \beta_j^2 \right),$$

où λ est un terme positif de pénalisation.

Le modèle linéaire, tel que décrit ci-dessus, est basé sur l'hypothèse d'un lien linéaire entre les caractéristiques et la variable cible. En effet, dans certains cas, les variables peuvent avoir un lien parabolique ou plus complexe. Le Modèle Additif Généralisé (GAM en anglais), introduit par [Hastie and Tibshirani, 1986] et théoriquement étudié par [Abramovich and Lahav, 2015; Guedj and Alquier, 2013], propose d'estimer des fonctions pour chaque variable, telles que:

$$Y_i = \beta_0 + \sum_{j=1}^{m} s_j(X_j) + \varepsilon_i.$$

Généralement, les fonctions $s_j(\cdot)$ sont estimées par splines cubiques.

1.2.2.3 Méthodes de Machine Learning

Les méthodes de régression linéaires sont faciles à calculer et à expliquer, mais elles peuvent parfois manquer de précision. En effet, les modèles linéaires reposent sur de grandes hypothèses au détriment de l'exactitude. Les algorithmes d'apprentissage automatique n'ont en principe aucune hypothèse sur la structure des données. Dans la partie suivante de cette sous-sous-section, nous allons énumérer quelques algorithmes d'apprentissage machine parmi les plus connus. Cette liste n'est évidemment pas exhaustive. De plus, tous ces algorithmes sont également utilisés dans la classification mais nous allons les introduire pour des problèmes de régression. Ces méthodes sont souvent appelées algorithmes "boîtes noires" car il n'existe aucun moyen de les décrire de manière fonctionnelle explicite.

Une des méthodes les plus faciles à comprendre est l'arbre de décision. Comme nous l'enseigne [Quinlan, 1986], [Hunt et al., 1966] a introduit les arbres de décision pour la première fois en 1966 avec l'algorithme Classification And Regression Trees (CART). Étant donné une variable cible quantitative $Y \in \{u_1, \ldots, u_K\}$ et une matrice de variables explicatives X contenue dans un espace $\mathcal{D}$, le principe est de partager l'espace $\mathcal{D}$ en régions binaires R_m de façon récursive. Chaque nouveau fractionnement est effectué sur une seule région, afin de discriminer les données autant que possible. L'arbre cesse de se développer lorsque la création d'un nouveau nœud n'améliorerait plus la prévision. Nous appelons les noeuds finaux des feuilles. Mathématiquement, la fonction de régression est:

$$\hat{a}(X) = \sum_{m=1}^{M} c_m \mathbb{1}_{[X \in R_m]},$$

où M est le nombre final de régions, R_m la $m^{\text{ème}}$ région et c_m la réponse de celle-ci. Les coefficients c_m sont estimés par:

$$\hat{c}_m = \frac{1}{n_m} \sum_{i|X_i \in R_m} Y_i,$$

où n_m est le nombre d'observations de l'échantillon d'apprentissage qui sont contenues dans la région R_m. Un exemple d'arbre de régression est représenté sur Figure 1.12. Le principe de l'algorithme Random Forest – introduit dans [Breiman, 2001] – est basé sur l'algorithme CART. Une forêt aléatoire est composée de T arbres de décision, où chaque arbre est partiellement indépendant des autres arbres. En effet, chaque arbre n'est formé qu'à partir d'un échantillon d'observations et d'un échantillon de variables. La valeur de régression finale d'une observation est alors une valeur agrégée de toutes les valeurs obtenues par chaque arbre:

$$\hat{Y}_i = \frac{1}{T} \sum_{t=1}^{T} a_t(X_{i,1}, \ldots, X_{i,m}),$$

avec a_t étant l'arbre t^{th} dans la forêt. Les méthodes des forêts aléatoires sont parmi les meilleures en termes de prévision et sont largement étudiées [Biau and Scornet, 2016; Genuer et al., 2008], mais ont été théoriquement mal comprises pendant longtemps. Les premières études ont été dirigées par [Genuer, 2012; Arlot and Genuer, 2014], et une percée importante s'est produite plus récemment par [Scornet et al., 2015].

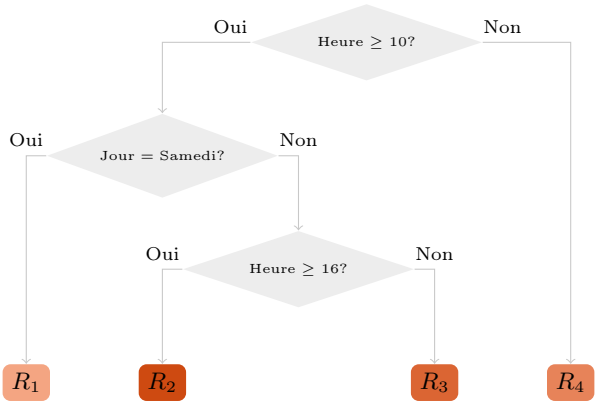


Figure 1.12: Exemple d'arbre de régression. Plus les feuilles sont oranges, plus le nombre de validations prévisionnelles est élevé.

Une façon intuitive de prédire la valeur de la variable cible est de regarder la valeur cible des voisins les plus proches de l'observation et de les moyenner. C'est en effet ce que [Benedetti, 1977; Stone, 1977; Tukey, 1977] ont introduit avec l'algorithme des plus proches voisins (NN en anglais). Mathématiquement, cela se traduit par:

$$\hat{Y}_i = \frac{1}{k} \sum_{l=1}^{k} Y_{(l)},$$

avec $X_{(1)}, \ldots, \ldots, X_{(k)}$ étant les k plus proches voisins de X_i et $Y_{(1)}, \ldots, Y_{(k)}$ leur valeur cible correspondante. L'enjeu est de trouver un entier optimal k. En effet, k trop petit conduit à un sur-ajustement, alors qu'une valeur trop grande conduit à un manque de précision et à un écart énorme. L'efficacité de l'algorithme a été étudiée dans plusieurs articles tels que [Altman, 1992; Stute, 1984; Devroye, 1978].

1.2.2.4 Contribution principale du chapitre 5

L'objectif du chapitre 5 était de trouver un modèle permettant de prévoir le nombre de validations d'un réseau de transport public et de pouvoir détecter si une observation est normale ou non. Nous avons commencé par comparer la performance des modèles linéaires, des modèles additifs et des forêts aléatoires pour prévoir le nombre de validations dans nos données,

par $Y_t = f(X_t) + \varepsilon$. Le résultat de cette comparaison est contenu dans la Table 1.2. Nous avons donc choisi de prévoir l'affluence par GAM. Afin

Table 1.2: Root Mean Squared Error (RMSE) of the three models on the test set by the two link functions used.

	Algorithm		
Link function	OLS	GAM	RF
Y_t	95	**46**	81
$\log(Y_t + 1)$	85	52	125

de créer un intervalle de confiance à 95%, nous avons ensuite appliqué ces mêmes algorithmes aux carrés des erreurs de la première prévision. Nous l'avons appliqué à des modèles simples:

$$\mathbb{E}[\varepsilon_t^2] = h(X_t)$$

et auto-regressifs:

$$\mathbb{E}[\varepsilon_t^2] = l(X_t, \varepsilon_{t-1}^2).$$

Au final, nous avons construit notre intervalle de confiance par forêts aléatoires auto-régressives, par:

$$Y_t \in \left[\hat{Y}_t - 1.96 \times \hat{\varepsilon}, \hat{Y}_t + 1.96 \times \hat{\varepsilon}\right].$$

Nous pouvons observer cette prévision ainsi que son intervalle de confiance sur la Figure 1.13.

Comme cette méthodologie a donné de bons résultats, nous l'avons appliquée aux données d'un autre réseau de transport pour la détection des anomalies. En effet, nous voulions pouvoir quantifier l'impact de la grève sociale de la SNCF au printemps 2018 sur les validations d'un réseau de bus en Île-de-France. Nos travaux ont mis en évidence que l'impact sur les validations, et donc sur le plan financier, est très modeste. Cependant, nous avons montré que pendant la grève, les passagers ont changé de trajectoire en utilisant les lignes différemment. Les validations de deux lignes pendant la période de grève sont représentées sur la Figure 1.14, et nous observons clairement que la grève a eu un impact négatif sur la ligne 601 pendant les jours de grève, mais aussi pendant les jours d'entre-grèves, les observations sont plus élevées que prévu pour la ligne 609 durant ces deux types de journée.

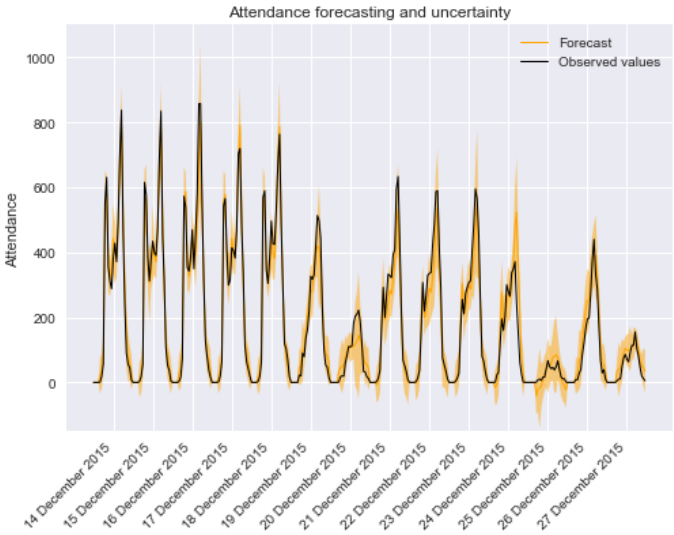
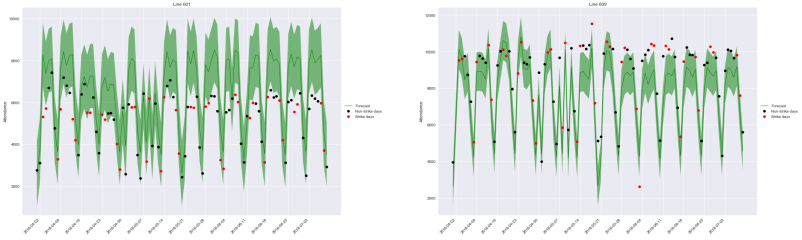


Figure 1.13: Prévision et incertitude de l'affluence dans une station de tramway.



(a) Ligne 601 (b) Ligne 609

Figure 1.14: Observations, prévisions et intervalle de confiance de deux lignes au printemps 2018. Les observations sont représentées par des points et les jours de grève sont en rouge, tandis que la prévision est la ligne verte et l'intervalle de confiance est le ruban vert autour.

Chapter 2

Introduction (English)

2.1 Context and motivations

After World War II in the United States of America (USA), land prices in the center of cities have declined or increased little. In the mean time, land prices in peripheral areas of cities and metropolises largely boomed. Two mains theories were supported by economists and ecologists. Firsts claimed that an individual looking for a place to settle would face a trade-off between the size of the land purchased and the distance to the city center. Whereas ecologists asserted that this individual would "maximize his satisfaction by avoiding the goods he dislikes and owning and consuming those he likes". In this context, in 1964, the authors of [Alonso, 1964] affirmed that the decision of where to settle of an individual is more complex. Indeed, they theorized that having a certain income to spend as wished to find a place to settle, an individual would take into account several criteria such as land cost, commuting costs and other expenditures. It was the first time a researcher used transportation to explain urban growth, and many other works followed.

Since that, in 2012 the authors of [Duranton and Turner, 2012] studied the impact of development of highways on cities in the USA and found out that it has a large positive influence on employment rate. This result showed that developing the transport infrastructure affect certainly urban growth. Indeed, these infrastructures allow people to commute more easily from farther.

There are obviously other factors included in a city growth. For example, [Black and Henderson, 1999] revealed that city sizes are strongly

affected to local educational attainment. In fact, citizens with higher education are more likely to help the city grow by innovate and create employment. This has although been proved in the Netherlands by [Van Oort, 2017], which indicates that exchanges of information and knowledge have implicated higher economic growth rates across urban areas and higher intensity of innovation in dense economic activity regions.

In the European Union (EU), but elsewhere as well, one of the cornerstones of the development strategies have been transport infrastructures. However, [Crescenzi and Rodriguez-Pose, 2012] showed that regional growth in the EU is mostly driven by a combination of adequate social politics, good innovation capacity across the region and the attractiveness for migrants.

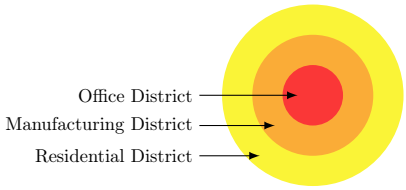
Also, [Meyronin, 2015] draws attention on cities that use enterprises strategies as marketing to enhance their territories. For example, in 2006 the French city of Lyon allowed the spectators of the "Fête des Lumières" to experiment a video game feigning a bike ride through the city. The purpose of this operation was to develop the city's digital image. In the same style, Dubai developed its brand image internationally by publicizing its principal touristic projects as the Burj-Al-Arab and the Burj Khalifa buildings, the Palm Islands real estate complex or even the first ski slope in the middle of a desert.

A good summary of the role of transportation in cities' growth is conducted by [Duranton and Puga, 2014]. Not only are the authors recalling that in monocentric models of city – see Figure 2.1 for details on the monocentric city model – population growth, greater suburbanization and increased land consumption are implied by lower transportation costs. But they also remind that infrastructures are often assigned on the basis of expected growth and that any improvement takes time to have influence on the growth.

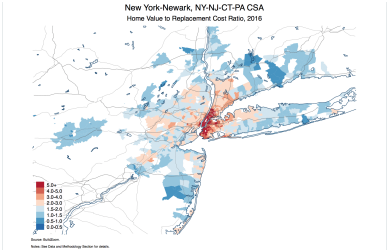
2.1.1 Sociological considerations

It is pretty obvious that a high quality public transportation system allows citizens to minimize their use of personal motorized vehicles to travel across the city.

Thus, [Litman, 2016] showed that it also have a positive impact on user's health. Indeed, he proved that it reduces vehicles accidents and pollution emissions while increasing passenger's mental health and fitness, as they walk more to access stations and stops than people using their



(a) Theoretical diagram of the land value in a monocentric city model. The land value decreases when the distance to the city center increases.



(b) Home value to replacement cost ratio in New York City and its surroundings. This is an example of the land values that describes this metropolis as monocentric. Image by Issi Romem on Buildzoom [Romem, 2017].

Figure 2.1: Monocentric city model – theoretical diagram and example

vehicle. Moreover, an efficient network permits disadvantaged people with no car to access more neighborhood and thus more services (such as shops or health services) and to improve their lifestyle.

The authors of [Gendron-Carrier et al., 2018] also showed that when a new subway line is opened, a decrease in pollution particles is measured. In [Zion and Lerner, 2017], mobility pattern are used to understand the different neighborhoods of the city, and then understand the sociology of the city.

2.1.2 Economical considerations

In the monocentric city model we mentioned earlier, commute trips by private mode have often been studied. Indeed, in [Arnott et al., 1993; Anderson and De Palma, 2007] for example, the authors studied congestion at one city gates and in its parking during peak periods. Although American cities – on which most of these studies are based – have generally little developed infrastructures of public transportation, it is not the case of European cities. Unfortunately, we found only a few references dealing with the impact of a developed public transportation network on the economic health of the city.

Some papers [De Palma et al., 2015, 2017] focused on congestion issues in public transportation for commuters. Both these papers highlight the

need of optimal timetables and train capacities to offer less congestion to commuters. Indeed, trains with larger capacities will allow more comfort to peak-period passengers, but an optimal timetable will also allow travelers willing to avoid crowd to arrive earlier or later to their destination in less crowded trains. This last proposition permits a congestion spread.

However, [Litman, 2015] studied the differences between cities with a large rail infrastructure, cities with a small rail infrastructure and cities with no rail infrastructure. The author stated that a bigger rail infrastructure – thus cities with a more efficient network – implies bigger ridership by capita, less traffic fatalities and less households budget allocated to transportation. Then, budgets are higher for other goods or services. All these results show that an efficient transportation network implies a healthier and wealthier local economy. Another example is given by [Pang, 2018], which shows that a dense network facilitates the employability of low-skilled workers.

The economic studies mentioned here show that an optimal public transportation network not only have a better attractiveness for citizens, but also appears to be an axis of economic development for cities. Everything that has been described in this thesis so far motivates the detailed study of transport data.

2.1.3 Smart Cities and Urban Data

In traditional studies, yearly data are used. But since the development of digital miniaturization, every conceivable type of object contains a computer that generates huge quantities of data. And as [Batty, 2013] present it, we are now able to better understand how cities function on much shorter term than before. As seen previously, traditional studies focus on location of land use and long-term functioning in cities. However, with these new ubiquitous sensors, it is easier to study movement and mobility than before. In the same way we call our phones "smart" since they are small computers, we can now boast about smart cities.

This term first appeared in 1994, but is more and more used in many papers. A full definition of smart cities is obtained by combining the works of [Dameri and Cocchia, 2013; Nam and Pardo, 2011; Harrison et al., 2010] to name but a few. The goal of a smart city is to offer the best life conditions to its citizens and visitors.

To achieve it, the city needs to be instrumented, that is being able to collect a large amount of data through the use of senses, meters, kiosks, personal devices, cameras, appliances, smart phones, implanted medical

devices and even social networks. The authorities then need to be interconnected by implementing a platform where these data can be stocked and communicate about it among the several city services. Finally, the city needs to become intelligent, by including analytics, modeling, computing and visualization. All these can then serve to solve more and more recurring urban issues, such as road congestion, noise and air pollution, energy and water consumption and waste treatment.

A smart city is also a sustainable city, since they try to solve similar problems, that are mostly environment related. In 2014, [Lee et al., 2014] gave the example of the city of San Francisco, among other cities. At that time, the city was still defining its smart strategy, but had already launched its own open data platform called 'DataSF', had several intelligent analytical tools based on real-time and integrated transportation services for real-time prediction or demand responsive pricing for parking.

As both the EU and the United Nations (UN) set ambitious climate and energy targets for the years to come, [Ahvenniemi et al., 2017] pointed that we urgently have to find smarter ways to decrease pollution and improve energy efficiency.

(a) Pollution over Paris (France). Photo by Alberto Hernandez on flickr.

(b) Congestion in Gurgaon (India). Photo by Taresh Bhardwaj on flickr.

Figure 2.2: Some issues encountered by large cities

In fact, a part of the solution is to use urban computing and statistics as pointed in [Zheng et al., 2014a]. The authors also recall that we can find several type of urban data:

- Geographical data: In Beijing, several studies have been using the GPS trajectories of taxicabs. These data served to evaluate the effectiveness of urban planning, detect traffic anomalies and detect the function of each area of the city [Zheng et al., 2011; Pan et al., 2013; Yuan et al., 2012]. In Australia, Bluetooth detectors have been placed across the city of Brisbane. These data have been spatiotemporally clustered to be able to describe vehicles dynamics in the city in [Laharotte et al., 2015].

- Traffic data: The authors of [Zhang et al., 2017] forecast the crowds traffic in each region of the cities of New York City and Beijing. In Pisa, GPS data are used to detect traffic congestion and incident and inform other drivers in the area [D'Andrea and Marcelloni, 2017]. Finally, [Castro et al., 2013] indicates that these data are used to analyze three different dynamics: social dynamics, traffic dynamics and operational dynamics.

- Mobile Phone Signals: Mobile Phones generates a variety of data, useful to urban planning. For example, geolocation sensors help recommend new places or event to users [Bothorel et al., 2018]. In Singapore, the authors of [Jiang et al., 2017] related that mobile phone call details record are used to better understand spatial human mobility pattern through the city.

- Environmental Monitoring Data: In France, [Abadi et al., 2017] explains that smart meters are quite new for water, but they have been able to understand and forecast water consumption. For air quality, [Zheng et al., 2014b] showed that in nine Chinese cities, they used historical and real-time air pollution data to infer air quality in cities area without monitor station.

- Social Network Data: Social Networks offer a large amount of detailed data about their users. For example, [Zheng, 2011] use user's location history to recommend him new friends to meet and to create communities with the same interests. In Japan, [Lee and Sumiya, 2010] detect unusual events such as festivals with geotagged tweets, whereas in Beijing [Pan et al., 2013] describe traffic anomalies with the WeiBo microblogging platform. More recently, [Atefeh and Khreich, 2015] propose a survey of techniques for event detection on tweets.

- Economy: The authors of [Di Clemente et al., 2018; Louail et al.,

2017] used credit card purchases to cluster the population to reveal their urban lifestyle and to analyze the shopping mobility practices to counterbalance the socioeconomic inequalities between neighborhoods.

- Energy: In Ireland, households have been clustered on their consumption behavior, thanks to smart electric meters [Melzi et al., 2017]. Moreover, socio-economic data have been used to analyze the clusters.

- Health Care: In [Dzhambov et al., 2018], the authors use mental health data and urban noise data to establish a link between these two phenomenon, whereas [Guarnieri and Balmes, 2014] establish a link between asthma and urban air pollution.

- Commuting Data: Finally, the data we have the most interest in are commuting data. There is a lot of literature about studies on that type of data. In the French territory of Val d'Amboise, intermodal travels are studied and these data served to make an economic comparison between a bike-and-ride service and a park-and-ride service [Papon et al., 2017]. Several papers deal with bike sharing systems data [Côme and Oukhellou, 2014; Bouveyron et al., 2015] in order to understand the relationships between neighborhood's type and the most common mobility pattern and to assign a function to each region. In [Briand et al., 2017; El Mahrsi et al., 2017], the authors cluster smart card data in order to create groups of passengers having similar temporal behavior and group of stations having the same type of usage. To be able to identify pickpocket suspects, [Du et al., 2018] detect unusual daily transit records. In [Toqué et al., 2017], the authors use smart card data to forecast travel demand on a short (15 to 30 minutes) and a long (1 year) term in the area of La Défense in Paris.

There is a large diversity of urban data and [De Palma and Dantan, 2017] addresses the challenges of dealing with such a variety of data. Indeed, even if this is not the topic of this thesis, the accumulation of such personal data raises several problems. These ethical problems deal mainly with data security [Hardt et al., 2016], data anonymization [Gadouche and Picard, 2017] and non-use for purposes of discrimination [Abadi et al., 2016; Ji et al., 2014].

In this thesis, conducted at CREST thanks to Transdev funding, we aimed to propose progress on the analysis of certain types of data described

above and in particular on transport data

2.1.4 Transdev

Transdev is a French transportation operator operating internationally. The company as been created in 2011 – firstly under the name of Veolia Transdev – by merging Veolia Transport (from Veolia) and Transdev (from Caisses des Dépôts et Consignations). As this is being written, these companies are still the main shareholders of Transdev, but Veolia announced its intention to sell its shareholding to the Rethmann Group before the end of 2018 [Trompiz and Mazzilli, 2018]. By operating buses, tramways, ferries, taxis, coach lines, trains, shuttles, medical services, school services and autonomous vehicles across 20 countries, Transdev transports 11 millions passengers everyday.

Table 2.1: Transdev in a few numbers

Creation	2011
Employees	82000
Daily passengers	11 millions
Countries with operated networks	USA, Canada, Chile, Colombia, Morocco, France, Germany, Czech Republic, Finland, Ireland, Spain, United Kingdom, Sweden, Netherlands, Portugal, India, South Korea, China, Australia, New Zealand

The amount of data generated by these operations is huge. Indeed, ticketing, sales, human resources and maintenance data are registered daily. These data allow the company to monitor phenomenon that were not followed until recently. For example, smart cards appearance provided the possibility to follow customers product purchases and passengers patterns, which could revolutionize marketing strategies. If human resources register absenteeism and shifts data about drivers, it could be easy to determine which factors drive long sick leaves. Finally, by anticipating the wear of vehicles parts, break down could be avoided.

And still, all these potential new possibilities are only reachable by having the right theoretical knowledge and using appropriate methodologies

and technologies. Thus, this thesis' goal is to establish methods useful to analyze and valorize ticketing data.

During this thesis work, we were provided with ticketing data from several networks. In [Pelletier et al., 2009], ticketing data is described as the data "stored at each onboard validation: date and time of the validation, status of the transaction (boarding acceptance, boarding refusal, transfer), card ID, fare type, route ID, route direction, stop ID, bus ID, driver ID, run ID, and internal database ID". The databases structure may differ according to the network, but these bases always contain only smart card data. Validations made by magnetic tickets are not consigned in databases.

2.2 Summary of the chapters

Statistical learning is, according to the definition of [Friedman et al., 2001], the principle of learning from data thanks to statistical methods and models. These data are composed of observed variables that are either quantitative (the number of passengers in a bus for example) or categorical (such as the day of the week). For supervised learning problems (the difference between supervised and unsupervised problems is detailed in Subsubsection 2.2.1.1), the database contains a target variable (the one we want to predict) and a set of features (the other variables). Generally, the data are separated in two sets. The training set helps and builds a prediction model, which is applied to the test set to measure the prediction performance of the model on new and unseen observations.

Most of the supervised statistical learning issues are classified in two big categories: classification and regression problems. The aim of these problems is to explain the value of the target variable thanks to the features. If the target is a quantitative variable, it is called regression (forecasting the number of a passenger in a bus for example), while if it is categorical, we talk about classification (such as detecting if a purchase is fraudulent). Classification tends to group observations according to the target values thanks to the features.

In unsupervised learning, one analyzes data with no target variable. Thus, the aim is not to predict a value thanks to the features, but to detect patterns in it (detect groups of passengers with the same travel hours for example). The most used unsupervised technique is called clustering. This technique groups the observation with similar features. As this technique allocates observations into groups, we speak about unsupervised

classification.

2.2.1 Clustering

2.2.1.1 Definition

As explained briefly above, there are two types of classification. We talk about supervised classification when the data are labelled and the algorithm needs to discriminate the data on the features to explain the difference between the labels. In unsupervised classification – or clustering, as written by [Sathya and Abraham, 2013], the goal is to "identify hidden patterns in unlabelled data". The other difference between the two techniques is that in classification the number of classes is fixed – it is the number of categories of the target variable, while the optimal number of group in clustering is not known *a priori*.

During this doctoral work, we only focused on a clustering problem. This type of problem is mainly solved by two different approaches. Some algorithms – called model free algorithms – do not need any assumption on the data, whereas model based algorithms always need an assumption on the data structure to be processed. To illustrate our words, we will use a toy example of simulated data all along this Subsection. These data, represented in Figure 2.3, are a simulation of a mixture of two bivariate gaussian distributions. The first distribution – represented by orange dots – have 50 observations of non-subscriber passengers, whereas the second – with blue dots – counts 30 observations of subscriber passengers. The x-axis represents the user's age, and the y-axis represents the number of check-ins made by the user during a month.

2.2.1.2 Model free algorithms

There are several model free algorithms that allows us to proceed a clustering. We shall present some of them, but the following paragraphs don't give an exhaustive list of these algorithms.

The most intuitive clustering algorithm is the hierarchical cluster analysis (HCA), introduced in [Johnson, 1967]. This method is based on $D \in \mathbb{R}^{n \times n}$ the dissimilarity matrix, such that $\forall i \neq j \ D_{i,j} = D_{j,i} = d(C_i, C_j)$, where $d(C_i, C_j)$ is a function computing the distance between the i^{th} and the j^{th} clusters. Its principle is to find at each iteration the smallest distance contained in matrix D and group the corresponding clusters into a new one, and then compute the new dissimilarity matrix between the collection of clusters. The algorithm stops once all the observations are

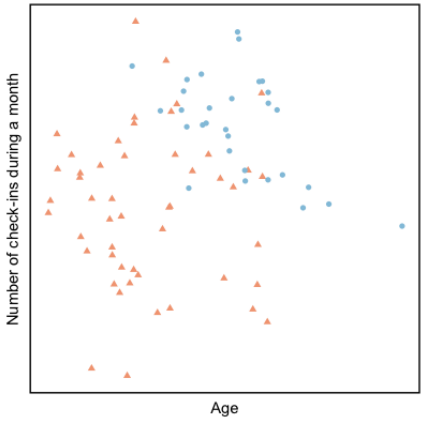
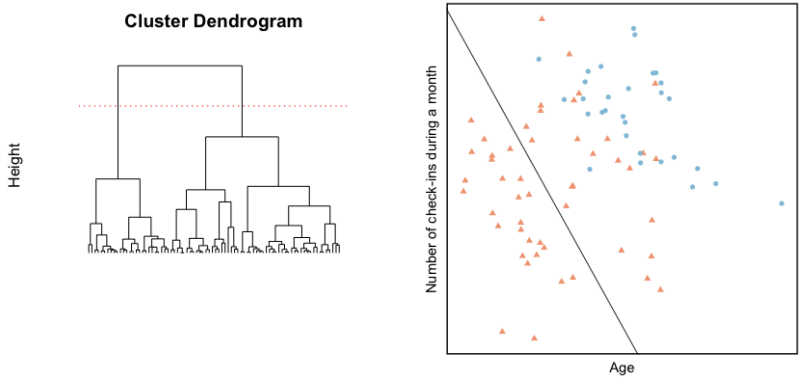


Figure 2.3: Toy data simulated as a mixture of two bivariate gaussian distributions. There are 50 orange observations and 30 blue.

contained in a single cluster. The segmentation information is then contained in a dendrogram, such as the one in Figure 2.4a. To find the optimal number of cluster, one look at the biggest height between two consecutive number of groups on the dendrogram. Figure 2.4b shows the partition of the space of our toy data. While this method is easy to implement, it can easily be costly for large dimension data. Indeed, at each iteration the dissimilarity matrix is computed.

The second most common algorithm is the k-means method. It is a two-steps iterative method, which concept was proposed in 1956 by [Steinhaus, 1956], developed by [MacQueen, 1967] and that [Selim and Ismail, 1984] proved convergence. Given K centers of clusters, the algorithm calculates the distance between each observation and every center and allocates the observation to the nearest center. The new center is then calculated and the algorithm stops once convergence is reached. The detailed algorithm is the Algorithm 4. To select K the optimal number of clusters, several methods tends to minimize the within-cluster inertia while maximizing the between-cluster inertia. The gap method is the most common one and was introduced by [Tibshirani et al., 2001]. Mathematically, it translates into:

$$K^* = \underset{k}{\operatorname{argmax}}\, \mathbb{E}_n^* \left[\log \left(\sum_{r=1}^{k} \frac{1}{2n_r} \sum_{i,i' \in C_r} d_{i,i'} \right) \right] - \log \left(\sum_{r=1}^{k} \frac{1}{2n_r} \sum_{i,i' \in C_r} d_{i,i'} \right),$$



(a) Dendrogram of the HCA on the toy data

(b) Partition of the space by HCA

Figure 2.4: Hierarchical cluster analysis

where $\mathbb{E}_n^*[\cdot]$ is the expectation under the assumption that there is no cluster in the data, C_r is the r^{th} cluster and n_r its number of observations and $d_{i,i'}$ the distance between the i^{th} and i'^{th} observations. We performed this algorithm on our data, and the result is on Figure 2.5.

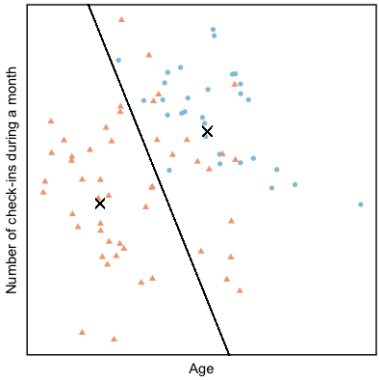


Figure 2.5: Partition of the space by k-means. The centers are represented by black crosses.

Algorithm 4 k-means

Choose arbitrary $m_1^{(0)}, \ldots, m_K^{(0)}$, K centers among the observations.
$t = 0$
while Stopping criterion has not been met **do**
  for $i \in \{1, \ldots, n\}$ **do**
    for $k \in \{1, \ldots, K\}$ **do**
      $d_{i,k} = ||x_i - m_k^{(t)}||$
    end for
    $c_i^{(t)} \leftarrow \text{argmax}_k \, d_{i,k}$
  end for
  $N_k^{(t)} = \sum_{i=1}^{n} \mathbb{1}_{[c_i^{(t)}=k]}$
  $m_k^{(t+1)} = \frac{1}{N_k^{(t)}} \sum_{i \in \mathcal{C}_k^{(t)}} x_i$
  $t = t + 1$
end while

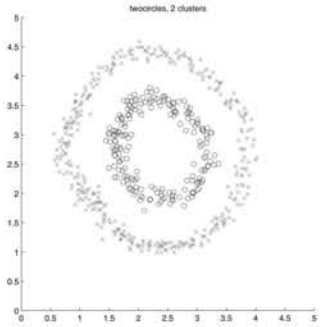
Spectral clustering have been introduced more recently, and a lot of researchers have worked on it [Ng et al., 2002; Von Luxburg, 2007; Zelnik-Manor and Perona, 2005; Dhillon et al., 2004; Filippone et al., 2008; Stella and Shi, 2003]. The principle is to perform a clustering on the K main eigenvectors of the transformation of the data's similarity matrix. This method allows to detect more complicated structures of data, like non-convex clusters such as the ones on Figure 2.6a. Figure 2.6b shows the application of spectral clustering with a polynomial kernel function on our toy data.

Several clustering methods are based on matrix factorization. Matrix factorization is the principle of approximate a matrix $X \in \mathbb{R}^{n \times m}$ by at least two matrices such as:

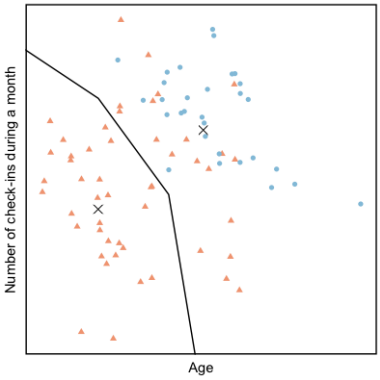
$$X \approx UV$$

with $U \in \mathbb{R}^{n \times p}$ and $V \in \mathbb{R}^{p \times m}$ and $p \ll n, m$. Matrix factorization is mainly used in order to reduce data dimension for sake of stocking them (in numerical analysis) or to estimate the matrix (in statistics).

Principal Components Analysis (or PCA) is one of the most known statistical method and is based on matrix factorization. Indeed, it has been created at the beginning of the XX^{th} century by [Pearson, 1901] to represent large datasets in two or three dimensions by creating principal components and axis on which the data would be projected. The aim is to



(a) Spectral clustering on complex data structure. Chart from [Ng et al., 2002]



(b) Partition of the space by spectral clustering

Figure 2.6: Examples of spectral clustering.

find a new basis allowing to emphasize the data variability. As [Wold et al., 1987; Mackiewicz and Ratajczak, 1993] describe it, a principal component is a linear combination of the data's variables. Indeed, mathematically we have:

$$X = WU,$$

where $X \in \mathbb{R}^{n \times m}$ is the data matrix of n observations of m variables, $W \in \mathbb{R}^{n \times p}$ is the projection of the n data observations on the p principal components and $U \in \mathbb{R}^{p \times m}$ the matrix containing the expression of the p principal components in function of the m variables. The columns of U are orthogonal while W's are orthonormal.

We can also interpret the k-means method – described above – as a matrix factorization method. Indeed, given:

$$X \approx CM,$$

where $X \in \mathbb{R}^{n \times m}$ is the data matrix of n observations in m dimensions, $C \in \{0,1\}^{n \times K}$ gives the allocation of the observations into the clusters such as $C_{i,k} = \mathbb{1}_{[x_i \in \mathcal{C}_k]}$ and $M \in \mathbb{R}^{K \times m}$ gives the clusters center. In this way, the observations are approximated by the center of the cluster they belong to.

Nonnegative Matrix Factorization (or NMF), as the name indicates, is a matrix factorization method proposed by [Lee and Seung, 1999]. In this

celebrated paper, the method is presented as a dimension reduction tool for matrices with nonnegative entries. Indeed, if $\theta \approx \Phi\Lambda$ with $\theta \in \mathbb{R}_+^{n\times M}$, $\Phi \in \mathbb{R}_+^{n\times H}$, $\Lambda \in \mathbb{R}_+^{H\times M}$ and $H \ll n, M$, then the number of entries in θ is much bigger than the ones in Φ and Λ: $Hn + HM \ll nM$. Several algorithm exist to solve this optimization problem, but the multiplicative algorithm – described in Algorithm 5 – is the most common.

Algorithm 5 Multiplicative algorithm for NMF

1: Fix $\varepsilon > 0$, choose arbitrary $\Lambda^{(0)}$ and $\Phi^{(0)}$ non-negative.
2: $t = 0$
3: **while** $||\theta - \Phi^{(t)}\Lambda^{(t)}|| > \varepsilon$ **do**
4: $\quad \forall h,k \quad \Lambda_{h,k}^{(t+1)} \leftarrow \Lambda_{h,k}^{(t)} \frac{(\Phi^{T(t)}\theta)_{h,k}}{(\Phi^{T(t)}\Phi^{(t)}\Lambda^{(t)})_{h,k}}$
5: $\quad \forall h,k \quad \Lambda_{h,k}^{(t+1)} \leftarrow \frac{\Lambda_{h,k}^{(t+1)}}{\sum_{k'} \Lambda_{h,k'}^{(t+1)}}$
6: $\quad \forall j,h \quad \Phi_{j,h}^{(t+1)} \leftarrow \Phi_{j,h}^{(t)} \frac{(\theta\Lambda^{T(t+1)})_{j,h}}{(\Phi_{j,h}^{(t)}\Lambda^{(t+1)}\Lambda^{T(t+1)})_{j,h}}$
7: $\quad \forall j,h \quad \Phi_{j,h}^{(t+1)} \leftarrow \frac{\Phi_{j,h}^{(t+1)}}{\sum_{h'} \Phi_{j,h'}^{(t+1)}}$
8: $\quad t = t + 1$
9: **end while**

2.2.1.3 Main contribution of Chapter 3

The problematic of this chapter is to find a way to cluster travelers by their temporal habits, and to identify passengers with similar habits. In [El Mahrsi et al., 2014b], the authors propose a mixture of k multinomial distributions as a model to cluster travelers temporal profiles. They then estimate the parameters of the models, and assign the travelers to clusters, using the Expectation-Maximization (EM) algorithm.

Although the results they obtained allow to identify relevant users profiles, some clusters are not easily interpretable. To overcome this issue, we propose to reduce the dimension of the profiles by NMF. NMF leads in many high-dimensional applications to the definition of a sparse and easily interpretable dictionary: [Lee and Seung, 1999] provided examples in image analysis, [Xu et al., 2003; Shahnaz et al., 2006] in text document clustering, among others. Here, NMF provides a dictionary of typical profiles, and a projection of each profile in the span of this dictionary. As the typical profiles are contained in a dictionary, we call them words.

Any clustering method can then be used in this smaller space. We choose to apply a k-means method to this smaller space to obtain our clusters. This led to easily interpretable clusters. Two words and one cluster obtained with this methodology are represented on the Figure 2.7. We clearly see that the represented cluster is a linear combination of the two words showed here.

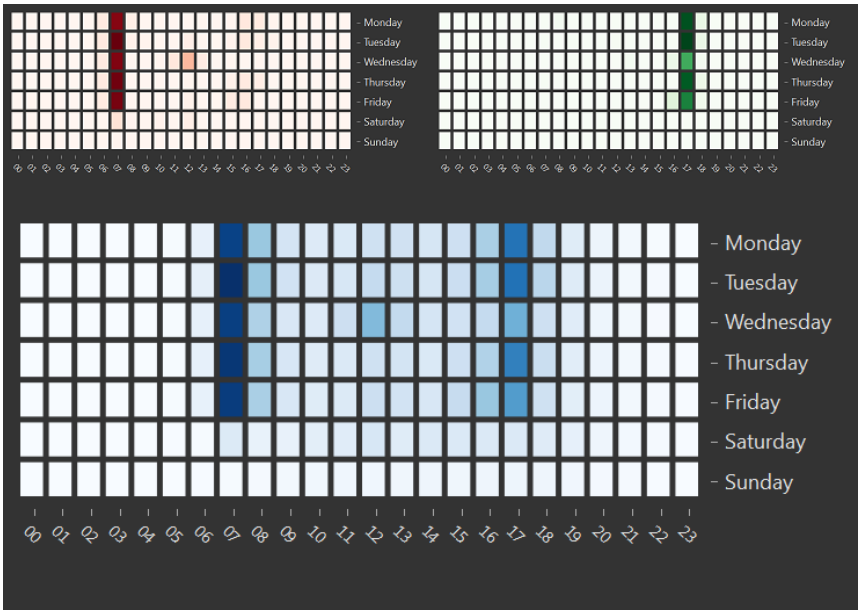


Figure 2.7: Two words and one cluster obtained with the methodology from Chapter 3

2.2.1.4 Model based algorithms - Mixture models

Among clustering algorithms, mixture model is one of the most common model based tools. Its principle is based on the assumption that in a population, "individuals may often be divided into subgroups" [Benaglia et al., 2009]. Therefore, the population isn't described by one distribution anymore, but by a mixture of K distributions. Given a parametric family

of distributions $(f_\vartheta)_{\vartheta\in\mathbb{R}^M}$, assume the observations $Y_1,\dots,Y_n$ are i.i.d from

$$f(\cdot)=\sum_{k=1}^{K} p_k f_{\theta_{\cdot,k}}(\cdot), \tag{2.1}$$

where each $\theta_{\cdot,k}\in\mathbb{R}^M$ is a column of a $K\times M$ matrix θ containing the parameters of the k^{th} distribution, and $p=(p_1,\dots,p_K)$ is the vector containing the probability for a random observation to belong to each distribution. The author of [Bishop, 2007] explains more precisely the case of Gaussian mixtures. There are numerous manners to estimate the θ matrix, and [Scrucca et al., 2016] reviews the different `R` packages implemented to estimate it.

In the field of biostatistics, several papers deal with the Bayesian Mixture Model [Lartillot and Philippe, 2004; Medvedovic et al., 2004; Do et al., 2005]. The principle of the method is to have an assumption on the parameters θ of the different distributions. These assumptions are contained in the prior:

$$\pi(\theta,p)=\pi_p(p)\times\prod_{k=1}^{K}\pi_k(\theta_{\cdot,k}),$$

where $\pi_p(p)$ is the assumption on the value of the vector p and $\pi_k(\theta_{\cdot,k})$ is the assumption on the parameters of the k^{th} distribution. Therefore, the estimation of the parameters is given by the posterior:

$$\pi_n(\theta,p|X)=\frac{\pi(\theta,p)L(\theta,p|X)}{\displaystyle\int\pi(\phi)L(\phi|X)},$$

where $L(\theta,p|X)$ is the likelihood of θ and p when $X=(X_1,\dots,X_n)$ is known. Theoretical studies of the Bayesian approach can be found in [Ghosal et al., 2000; Chérief-Abdellatif and Alquier, 2018] for example. We applied a Bayesian EM algorithm for mixture model to our data and the result is shown on Figure 2.8.

Generally, to estimate a distribution's parameters based on observations, the easiest and most famous way is to find the values maximizing the likelihood, called Maximum Likelihood parameters. Here is the likelihood of a mixture model:

$$L(\theta,p|X)=\prod_{i=1}^{n}\sum_{k=1}^{K}p_k f_{\theta_{\cdot,k}}(x_i),$$

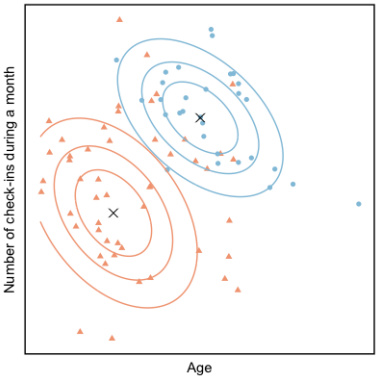


Figure 2.8: Density of the distributions found by a Bayesian EM algorithm for Gaussian mixture model. The posterior means of the distributions are represented by black crosses. We used the parameters of simulation as priors.

and its log-likelihood:

$$\ell(\theta, p|X) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} p_k f_{\theta_{\cdot,k}}(x_i) \right).$$

It is pretty obvious that because of the complex form of the function, this log-likelihood is impossible to maximize. As we don't know which observations were generated by which distributions, the EM algorithm proposes to introduce a latent variable $Z_{i,k}$ containing this information, such that $Z_{i,k} = \mathbb{1}_{[X_i \text{ generated by the } k^{\text{th}} \text{ distribution}]}$. Thus, the completed log-likelihood becomes:

$$\ell(\theta, p|X, Z) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{i,k} \log \left(p_k f_{\theta_{\cdot,k}}(x_i) \right).$$

The EM is a recursive algorithm, and given current parameters $(\theta^{(c)}, p^{(c)})$

one loop is:

$$\begin{aligned}
\textbf{E-step: } Q^{(c)}(\theta,p) &= \mathbb{E}_{\theta^{(c)},p^{(c)}}[\ell(\theta,p|X,Z)|X] \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K}\mathbb{E}_{\theta^{(c)},p^{(c)}}[Z_{i,k}|X]\log\left(p_k f_{(\theta)_{\cdot,k}}(X_i)\right) \\
\text{and } t_{i,k}^{(c)} &:= \mathbb{E}_{\theta^{(c)},p^{(c)}}[Z_{i,k}|X] \\
&= \frac{p_k^{(c)} f_{(\theta^{(c)})_{\cdot,k}}(X_i)}{\sum_{k'=1}^{K} p_{k'}^{(c)} f_{(\theta^{(c)})_{\cdot,k'}}(X_i)}. \qquad (2.2)
\end{aligned}$$

$$\textbf{M-step: } (\theta^{(c+1)},p^{(c+1)}) := \underset{\theta_{j,k}\geq 0}{\arg\max} Q^{(c)}(\theta,p), \qquad (2.3)$$

with obviously for $k \in \{1,\dots,K\}$:

$$p_k^{(c+1)} = \frac{\sum_{i=1}^{n} t_{i,k}^{(c)}}{\sum_{i=1}^{n}\sum_{k'=1}^{K} t_{i,k'}^{(c)}}. \qquad (2.4)$$

Of course this algorithm depends on the family of distributions $(f_\vartheta)_{\vartheta\in\mathbb{R}^M}$. An example is given below in the case of a mixture of multinomial distributions. Figure 2.9 shows the application of an EM algorithm for Gaussian mixture model to our toy data.

In the field of transportation, some studies have been leaded involving mixture model clustering. Indeed, the authors of [Côme and Oukhellou, 2014; Randriamanamihaga et al., 2013] use Poisson mixture models to cluster bike-sharing systems stations, whereas for the same purpose [Bouveyron et al., 2015] use functional mixture model to cluster temporal series of station occupancy. The article that we have the most interest in is [El Mahrsi et al., 2014a]. Indeed, in this article the authors use a multinomial mixture model to cluster passengers on their temporal profiles in the network STAR from Rennes, France. A temporal profile count the number of validations that took place at each day of the week and hour during the studied period, as showed in Figure 2.10. It is obvious that the profile u_i of the passenger i follows a multinomial distribution:

$$u_i \sim \mathcal{M}(D_i,\pi_i),$$

where D_i is the total number of validations made by passenger i during the study period and $\pi_i = (\pi_{i,1},\dots,\pi_{i,M})$ is the simplex containing the

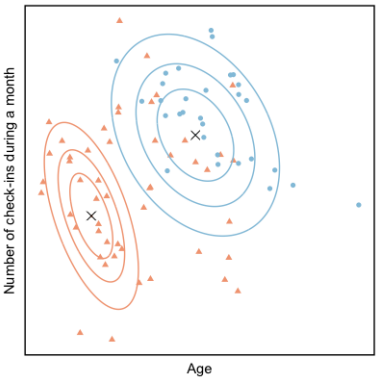


Figure 2.9: Density of the distributions found by EM algorithm for Gaussian mixture model. The mean point of the distributions are represented by black crosses.

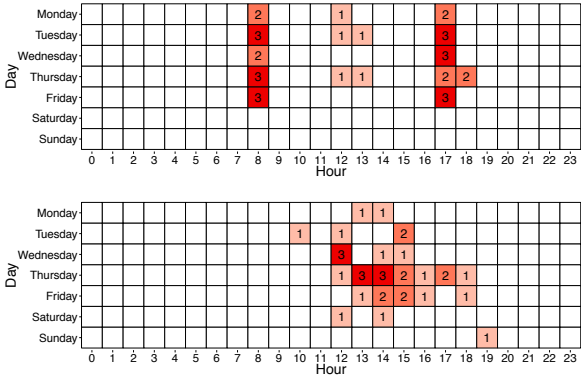


Figure 2.10: Temporal profiles of two passengers – Figure from [El Mahrsi et al., 2014a].

probabilities of the M events. It is simply the probability for a random validation from passenger i to happen at each moment $m \in \{1, \ldots, M\}$. Under the assumption that the collection of all the profiles is the realization of a mixture of K distributions, each cluster β_k follows a multinomial distribution such as:

$$\beta_k \sim \mathcal{M}(1, \pi_k).$$

And the temporal profile of passenger i follows:

$$u_i | z_i \sim \mathcal{M}(D_i, \pi_k),$$

with $z_i \in \{\beta_1, \ldots, \beta_K\}$. Thus, the mixture equation 2.1 becomes:

$$f(X_i) = \sum_{k=1}^{K} p_k D_i! \prod_{j=1}^{M} \frac{\pi_{k,j}^{X_{i,j}}}{X_{i,j!}}. \tag{2.5}$$

The detailed algorithm is thus described in Algorithm 6. For the sake of clarity, let $\Pi = (\pi_{k,j})_{1 \leq k \leq K, 1 \leq j \leq M}$.

Algorithm 6 EM algorithm for multinomial mixture model

Fix $\varepsilon > 0$, choose arbitrary $\Pi^{(0)}$ and $p^{(0)}$; CRIT $\leftarrow \infty$
$c \leftarrow 0$
while $|\ell(\Pi^{(c)}, p^{(c)}) - \text{CRIT}| > \varepsilon$ **do**
  CRIT $\leftarrow \ell(\Pi^{(c)}, p^{(c)})$
  $\forall i, k \quad t_{i,k}^{(c)} \leftarrow \frac{\pi_k^{(c)} \prod_{j=1}^{M} \pi_{j,k}^{X_{i,j}}}{\sum_{k'=1}^{K} \pi_{k'}^{(c)} \prod_{j=1}^{M} \pi_{j,k'}^{X_{i,j}}}$
  $\forall k \quad p_k^{(c+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} t_{i,k}^{(c)}$
  $\forall k, j \quad \pi_{j,k}^{(c+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} t_{i,k}^{(c)} X_{i,j}$
  $c \leftarrow c + 1$
end while

2.2.1.5 Main contribution of Chapter 4

In Chapter 3, NMF was a nice pre-processing tool for clustering passenger temporal profiles. However, we noted that we used a two-step procedure and wanted to define a one-step procedure that would directly estimate a dictionary optimizing a criterion related to the clustering objective.

Moreover, the article [El Mahrsi et al., 2014a] proposes a model-based method for a similar issue. In this article, we noticed that the parameter matrix to estimate can easily be heavy. Indeed, they need to estimate the parameters of twenty-four hours, for seven days and sixteen clusters, i.e. 2688 parameters.

Therefore, we propose in Chapter 4 a new algorithm called NMF-EM and allowing simultaneous dimension reduction and clustering. To that end, we suggest to apply a NMF to the parameter matrix to estimate by EM. The number of parameters to estimate will then be largely smaller. Indeed, if $\theta \in \mathbb{R}_+^{M,K}$ is the matrix containing in column the non-negative parameters of K distributions from the same distribution family $(f_\vartheta)_{\vartheta \in \mathbb{R}^M}$, we have:

$$\theta = \Phi\Lambda,$$

with $H \leq K, M$, $\Phi \in \mathbb{R}_+^{M,H}$ and $\Lambda \in \mathbb{R}_+^{H,K}$. In our case, we apply the NMF-EM algorithm for a mixture of multinomials. Thus, the model 2.5 becomes:

$$f(X_i|Z_{i,k}=1) = \sum_{k=1}^{K} p_k D_i! \prod_{j=1}^{M} \frac{(\Phi\Lambda)_{j,k}^{X_{i,j}}}{X_{i,j!}},$$

and the completed log-likelihood:

$$\ell(\Phi, \Lambda, p|X) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k} \log\left(p_k \left[N_i! \prod_{j=1}^{M} \frac{(\Phi\Lambda)_{j,k}^{X_{i,j}}}{X_{i,j}!}\right]\right).$$

Thus, from Algorithm 6, the update of the parameters becomes:

$$t_{i,k}^{(c)} := \frac{p_k^{(c)} f_{(\Phi^{(c)}\Lambda^{(c)})_{\cdot,k}}(Y_i)}{\sum_{k'=1}^{K} p_{k'}^{(c)} f_{(\Phi^{(c)}\Lambda^{(c)})_{\cdot,k'}}(Y_i)},$$

$$(\Phi^{(c+1)}, \Lambda^{(c+1)}, p^{(c+1)}) := \underset{\Phi_{j,h}, \Lambda_{h,k} \geq 0}{\arg\max}\ Q^{(c)}(\Phi, \Lambda, p).$$

This last update is obtained by injecting the multiplicative algorithm for NMF (Algorithm 5) into the EM algorithm. The application of this algorithm to real life data showed nice empirical results. Indeed, we proved that for collinear clusters, our algorithm outperformed both EM and k-means algorithms.

Moreover, thanks to NMF-EM we obtained five words and ten clusters with our data and Figure 2.11 shows two of the words and one of the clusters. In a second phase, we analyzed socio-economically these clusters and obtained interesting insights on the passengers of the studied network.

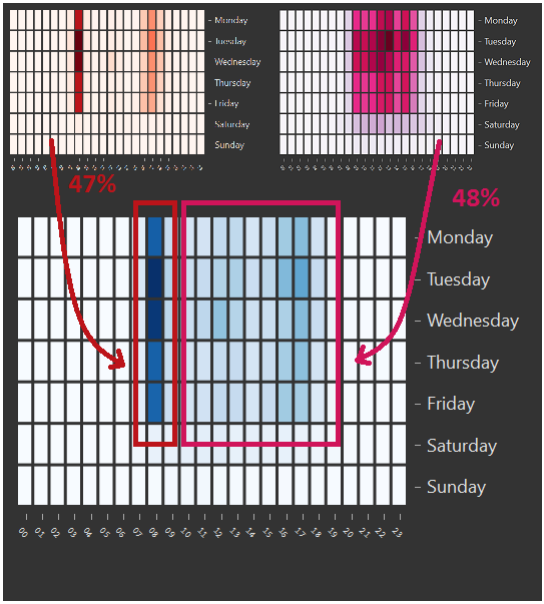


Figure 2.11: Decomposition of one cluster from two words. This cluster is 47% composed of the orange word and 48% of the pink one. The remaining 5% are a mixture of the three other words.

2.2.2 Regression and Forecasting

2.2.2.1 Definition

Forecasting and regression can easily be confused. As described on the blogpost [oneclick.ai, 2018], the main difference is that "regression predicts the value of a number in a hypothetical future event" while "forecasting uses a compiled timeline of data moments to then tell how it will continue into the future along those trends". We can summarize this by saying that not all regressions are forecast but all forecast are regressions. In the following parts of this section, we will only discuss regressions methods, but these are also relevant for forecasting.

Regression can be described as a process to predict a quantitative target variable Y from a set of m features $X = (X_1, \ldots, X_m)$. Regression methods allow to estimate a function $f(\cdot)$ such that $Y = f(X) + \varepsilon$, where ε is the difference between the observed value Y and the predicted value $\hat{Y} = f(X)$. The smallest ε, the more accurate is the regression function $f(\cdot)$.

2.2.2.2 Linear methods

Linear methods are among the simplest models for regression. The most famous one is the linear regression. Indeed, this model is based on the assumption that there are some linear links between the explanatory variables $(X_1, \ldots, X_m)$ and the target variable Y as explained in [Seber, 2009; Weisberg, 2005; Bibby et al., 1979]. Thus, for a population of n individuals, and $\forall i \in \{1, \ldots, n\}$:

$$Y_i = \beta_0 + \sum_{j=1}^{m} \beta_j X_{i,j} + \varepsilon_i,$$

where $\beta = (\beta_0, \beta_1, \ldots, \beta_m) \in \mathbb{R}^{m+1}$ is the vector containing the regression coefficients, and $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ are independent and identically distributed. The most popular way to perform a linear regression and estimate the β vector is to use the Ordinary Least Squares (OLS) method, that gives $\hat{\beta} = (X^t X)^{-1} X^t Y$.

In the case of badly conditioned problem (i.e. matrix X with collinear columns), but willing to keep every variable for sake of interpretation, one can use a biased estimator to enhance numerical properties and estimations variance by a regularization process. Theoretical proofs of its efficiency can be found in [Le Cessie and Van Houwelingen, 1992; Marquardt and Snee,

1975]. For that, we need to formalize the model in its matrix form. Thus, the ridge model proposed by [Hoerl and Kennard, 1970] is:

$$\underbrace{\begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}}_{Y} = \underbrace{\begin{pmatrix} 1 & X_{1,1} & \dots & X_{1,m} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n,1} & \dots & X_{n,m} \end{pmatrix}}_{X} \underbrace{\begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_m \end{pmatrix}}_{\beta} + \underbrace{\begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}}_{\varepsilon}.$$

In ridge regression, the estimator $\hat{\beta}$ is given by:

$$\hat{\beta}_{\text{ridge}} = \underset{\beta \in \mathbb{R}^{m+1}}{\operatorname{argmin}} \left(\sum_{i=1}^{n} (Y_i - \sum_{j=0}^{m} X_{i,j}\beta_j)^2 + \lambda \sum_{j=1}^{m} \beta_j^2 \right),$$

where λ is a positive term of penalization.

The linear model, as described above, is based on the assumption of a linear link between the features and the target variable. Indeed, in some cases the variables can have a parabolic or more complex link. The Generalized Additive Model (GAM), introduced by [Hastie and Tibshirani, 1986] and theoretically studied by [Abramovich and Lahav, 2015; Guedj and Alquier, 2013], proposes to estimate functions for every feature, such as:

$$Y_i = \beta_0 + \sum_{j=1}^{m} s_j(X_j) + \varepsilon_i.$$

Generally, $s_j(\cdot)$ functions are estimated by cubic splines.

2.2.2.3 Machine Learning methods

Linear regression methods are easy to compute and explain, but can sometimes lack accuracy. Indeed, linear models lie on huge assumptions at the cost of accuracy. Machine Learnings algorithms have for principle to have no assumption on the data structure. In the following part of this Subsubsection, we will list a few algorithm of Machine Learning among the most famous. This list is clearly not exhaustive. Moreover, all these algorithm are also used in classification but we will introduce them for regression problems. These methods are often called "Black Box" algorithms as there is no way to describe them in an explicit functional manner.

One of the easiest method to understand is the decision tree. As [Quinlan, 1986] teaches us, [Hunt et al., 1966] first introduced decision trees

in 1966 with the Classification And Regression Trees (CART) algorithm. Given a quantitative target variable $Y \in \{u_1, \ldots, u_K\}$ and a feature matrix X contained in a space $\mathcal{D}$, the principle is to split the feature space $\mathcal{D}$ into binary regions R_m recursively. Each new split is performed on one region only, in order to discriminate the data as much as possible. The tree stops expanding when creating a new binary node would not improve the forecast anymore. We call the final nodes leaves. Mathematically, the regression function is:

$$\hat{a}(X) = \sum_{m=1}^{M} c_m \mathbb{1}_{[X \in R_m]},$$

where M is the final number of regions, R_m the m^{th} region and c_m the response of the region. Coefficients c_m are estimated by:

$$\hat{c}_m = \frac{1}{n_m} \sum_{i|X_i \in R_m} Y_i,$$

where n_m is the number of observations from the train set that are contained in the region R_m. An example regression tree is represented on Figure 2.12. The principle of the Random Forest algorithm – introduced

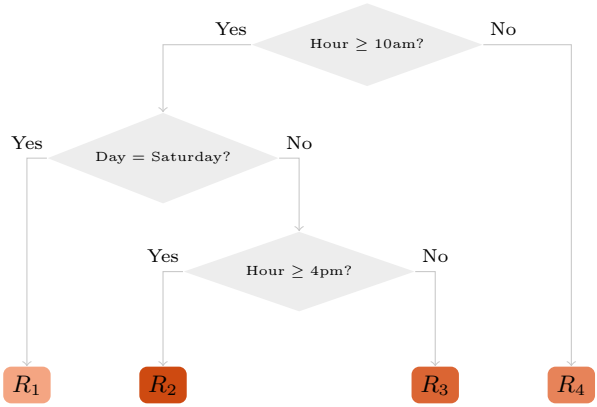


Figure 2.12: Example of regression tree. The more orange are the leaves, higher is number of check-ins forecasted.

in [Breiman, 2001] – is based on the CART algorithm. A random forest

is composed of T decision trees, where each tree is partially independent from every other tree. Indeed, every tree is trained only from a sample of observations and a sample of features. The final regression value of an observation is then an aggregated value of all the values obtained by each tree:

$$\hat{Y}_i = \frac{1}{T}\sum_{t=1}^{T} a_t(X_{i,1}, \ldots, X_{i,m}),$$

with a_t being the t^{th} tree in the forest. Random Forest methods are among the very best in term of forecasting and are widely studied [Biau and Scornet, 2016; Genuer et al., 2008], but were theoretically misunderstood for a long time. First theoretical studies were leaded by [Genuer, 2012; Arlot and Genuer, 2014] and an important theoretical breakthrough happened more recently [Scornet et al., 2015].

An intuitive way to predict the value of the target variable is to look at the target value of the nearest neighbors of the observation and to average them. This is indeed what [Benedetti, 1977; Stone, 1977; Tukey, 1977] introduced with the Nearest Neighbors (NN) algorithm. Mathematically, it translates into:

$$\hat{Y}_i = \frac{1}{k}\sum_{l=1}^{k} Y_{(l)},$$

with $X_{(1)}, \ldots, X_{(k)}$ being the k nearest neighbors of X_i and $Y_{(1)}, \ldots, Y_{(k)}$ their corresponding target value. What is at stake is to find an optimal integer k. Indeed, k too small leads to overfitting, while a too big value drives to a lack of accuracy and a huge variance. The efficiency of the algorithm has been studied in several papers such as [Altman, 1992; Stute, 1984; Devroye, 1978].

2.2.2.4 Main contribution of Chapter 5

The aim of Chapter 5 was to find a model to forecast the number of check-ins of a public transportation network and to be able to detect if a realization is normal or not. We began to compare the performance of Linear Models, Additive Models and Random Forests for forecasting the number of check-ins in our data, by $Y_t = f(X_t) + \varepsilon$. The results of this comparison is contained in Table 2.2. Thus, we chose to forecast the check-ins by GAM. In order to create a 95% confidence interval, we then processed the same algorithms on the squared errors of this first forecast. We applied them to simple:

$$\mathbb{E}[\varepsilon_t^2] = h(X_t)$$

Table 2.2: Root Mean Squared Error (RMSE) of the three models on the test set by the two link functions used.

	Algorithm		
Link function	OLS	GAM	RF
Y_t	95	**46**	81
$\log(Y_t + 1)$	85	52	125

and auto-regressive models:

$$\mathbb{E}[\varepsilon_t^2] = l(X_t, \varepsilon_{t-1}^2).$$

Finally, we constructed our confidence interval by an auto-regressive Random Forest, by:

$$Y_t \in \left[\hat{Y}_t - 1.96 \times \hat{\varepsilon}, \hat{Y}_t + 1.96 \times \hat{\varepsilon}\right].$$

And we can observe this forecast and confidence interval on Figure 2.13.

As this methodology produced nice results, we applied it to data from another transportation network for anomaly detection. Indeed, we wanted to be able to quantify the impact of the SNCF social strike on the validations of a bus network in Île-de-France during Spring 2018. Our work highlighted that the impact on the validations, and thus financially, is very modest. However, we showed that during the strike, passengers changed their paths by using the lines differently. The check-ins of two lines during the social strike period are represented on Figure 2.14, and we clearly observe that the strike had a negative impact on line 601 during strike days but also during inter-strike days, while the observations are higher than expected for line 609 on both type of days.

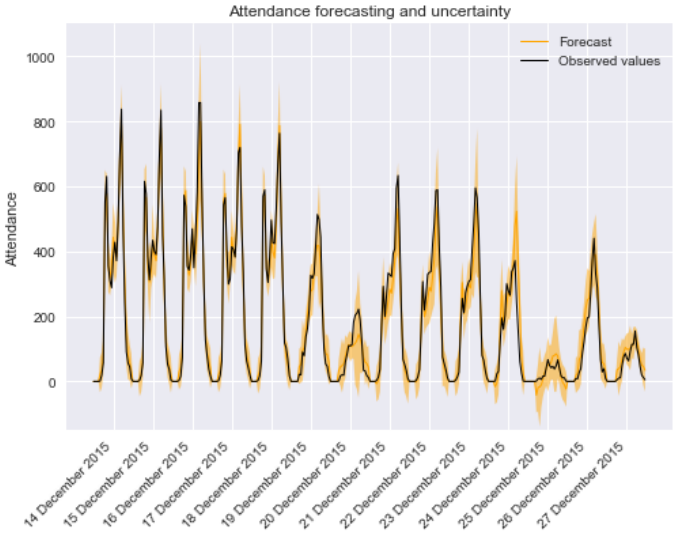
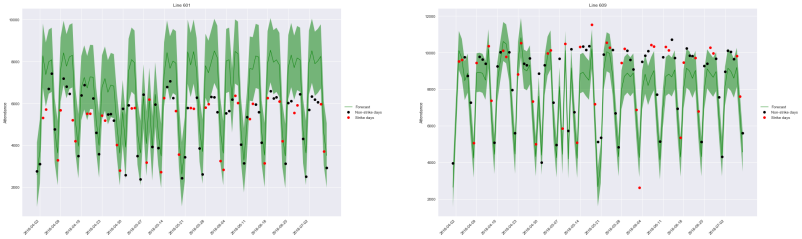


Figure 2.13: Attendance forecasting and uncertainty.



(a) Line 601 (b) Line 609

Figure 2.14: Observations, forecast and confidence interval during Spring 2018 of two lines. Observations are represented by dots and strike days are red, while the forecast is the green line and the confidence interval is the green ribbon around it.

Chapter 3

Non-negative Matrix Factorization as a pre-processing tool for travelers temporal profiles clustering

Abstract

We propose to use non-negative matrix factorization (NMF) to build a dictionary of travelers temporal profiles. Clustering based on decomposition in this dictionary rather than on the full profiles (as in previous works) lead to more interpretable clusters.

3.1 Introduction

In recent years, more and more travel networks use smart card automated fare collection systems. The main purpose of these systems is to collect the fare revenues. However, they also allow to collect a large amount of information on onboard transactions that can be used for various objectives: to analyze nowadays cities through global urban problematics [Zheng et al.,

2014a], to study the variability of the travels from a spatial and temporal perspective [Morency et al., 2007], to help transit planners [Pelletier et al., 2009] or to analyze the travel habits of smart card holders as in [El Mahrsi et al., 2014b].

More precisely, in [El Mahrsi et al., 2014b], the authors propose a mixture of k multinomial distributions as a model for the travelers temporal profiles. They then estimate the parameters of the models, and assign the travelers to clusters, using the Expectation-Maximization (EM) algorithm. Although the results they obtained allow to identify relevant users profiles, some clusters are not easily interpretable. To overcome this issue, we propose to reduce the dimension of the profiles by non-negative matrix factorization (NMF). NMF was introduced by [Lee and Seung, 1999] and leads in many high-dimensional applications to the definition of a sparse and easily interpretable dictionary: [Lee and Seung, 1999] provided examples in image analysis, [Xu et al., 2003; Shahnaz et al., 2006] in text document clustering, among others. Here, NMF provides a dictionary of temporal profiles, and a projection of each profile in the span of this dictionary. Any clustering method can then be used in this smaller space (we use k-means in this chapter). This leads to easily interpretable clusters.

3.2 The data

We study here validations made during the month of September 2014 on the network of Rouen metropolis. Ticketing data are the information obtained at each transaction made by a smart card on a validator system. For privacy reasons it is not possible to connect each validation to the user that made it. The feature that permits us to realize our study and create temporal profiles is a card number which is encrypted, and re-initialized every three months. It is thus impossible to follow the long-term behavior of a user. This is the reason why we focus on a one month period in a first time. This period (September) have been chosen because it has no vacation or bank holidays. We use the same method as [El Mahrsi et al., 2014b] in order to keep only the regular smart card holders: to be a regular card holder, the traveler must have used his card for at least ten days during the studied period and must have made his first boarding after 4am each day at the same station 50% of the time. The data are then aggregated so that for each traveler, for each day of the week (Monday to Sunday) and each hour (00 to 23) we have the mean of the number of validation during the studied period.

3.3 Results obtained by EM

In [El Mahrsi et al., 2014b] the authors assume that there is a given number of clusters of users, and in each cluster, the profiles are independently generated from a common multinomial distribution. Thus, the distribution on all profiles is a mixture of multinomial distributions. They used an EM-algorithm to estimate the parameters of each multinomial and the probabilities for any users to belong to each cluster (we refer the reader to [El Mahrsi et al., 2014b] for more details on this model, and to Chapter 9 in [Bishop, 2007] for an introduction to the EM algorithm). We use the same methodology on our dataset. The results for 10 clusters are shown in Figure 3.1 (we tested other numbers of cluster but do not show the results here for the sake of shortness). Two comments are in order: first, while some profiles are easily interpretable, it is not so easy to give an interpretation to Cluster 1 when compared to Cluster 6 and Cluster 9. Moreover, the clusters are really unbalanced: almost 35% of the travelers are in Cluster 6 while Cluster 4 contains only 3.7% of the travelers.

3.4 Results obtained by NMF

Consider the matrix θ that contains the temporal profiles θ_i of all users as rows. The principle of Nonnegative Matrix Factorization is to factorize the matrix $\theta \in \mathbb{R}^{n\times M}$ into two matrices $\Phi \in \mathbb{R}^{n\times H}$ and $\Lambda \in \mathbb{R}^{H\times M}$ such that $\theta \approx \Phi\Lambda$. When $H \ll n, M$ the number of entries in θ is much bigger than the ones in Φ and Λ: $Hn+HM \ll nM$. So each profile θ_i is approximated by $\Phi_{i,1}\Lambda_1 + \ldots \Phi_{i,H}\Lambda_H$ and so the Λ_j's can be interpreted as a dictionary of profiles. Moreover, the non-negativity constraint sets some $\Phi_{i,j} = 0$ and so each profile is approximated as a small number of elements in the dictionary.

It is important to find a H small enough to ensure that $Hn + HM \ll nM$, but large enough so that $\Phi\Lambda$ remains an acceptable approximation of θ. Let $D(\theta|\Phi\Lambda)$ denote a generic function measuring the distance between θ and its approximation $\Phi\Lambda$. To summarize, the aim of the NMF is to solve the following problem :

$$\min D(\theta|\Phi\Lambda), \text{ subject to } \Phi \geq 0, \Lambda \geq 0 \tag{3.1}$$

where the inequalities are interpreted element-wise. Several algorithms are known to compute Φ and Λ. These methods are discussed and compared in [Kim and Park, 2008].

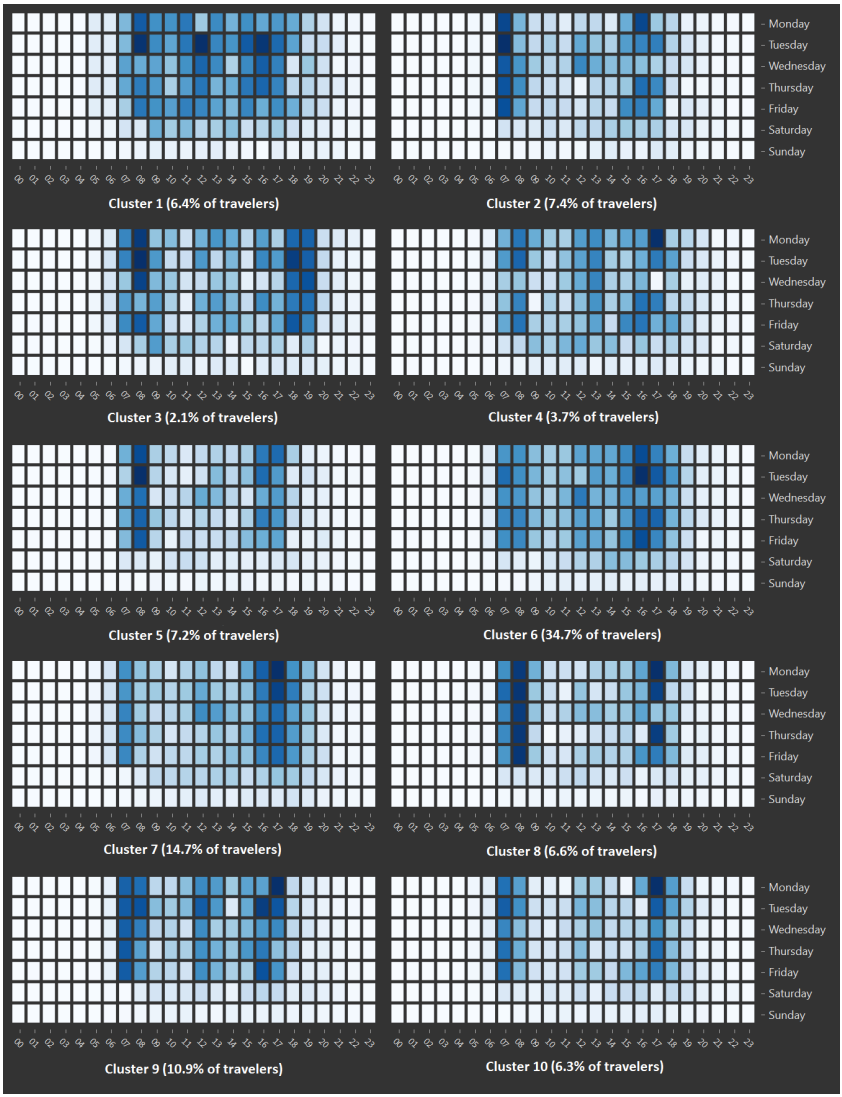


Figure 3.1: Clusters obtained by EM-algorithm

We chose $D(\theta|\Phi\Lambda) = \|\theta - \Phi\Lambda\|_F^2$ in (3.1). We tested different algorithms recommended in [Kim and Park, 2008]: the multiplicative algorithm and the projected gradient method. The results being similar on our dataset we only present the results obtained by the multiplicative algorithm. For the sake of completeness, we remind that the algorithm is an iteration of the following updates:

$$\Phi_{i,a} \leftarrow \Phi_{i,a} \frac{(\theta\Lambda^T)_{i,a}}{(\Phi\Lambda\Lambda^T)_{i,a}} \text{ and } \Lambda_{a,\mu} \leftarrow \Lambda_{a,\mu} \frac{(\Phi^T\theta)_{a,\mu}}{(\Phi^T\Phi\Lambda)_{a,\mu}} \ \forall i, \mu, a.$$

Here again we tested several dimensions H, and then used the k-means algorithm on the matrix Φ to get our clusters. The dictionary and the clusters centers are shown in Figure 3.2 for $H = 7$, which were particularly easy to analyze. Indeed, as it can be seen on the Figure 3.2 the first word corresponds to the first hour of the morning peak (7 a.m.). The second word corresponds to the second hour of morning peak (8 a.m.), the third to the last two hours of afternoon peak (6-7 p.m.) and Saturdays afternoons, the fourth to the off-peak periods, mostly in the morning (9-11 a.m. and 2-3 p.m.), the fifth to the midday hours (12 and 1 p.m.), the sixth to the first hour of afternoon peak (5 p.m.) and the seventh to the off-peak period in the afternoon (3 to 4 p.m.).

In the middle of the Figure 3.2, the first cluster obtained with a k-means method applied to our reduced space is mostly a combination of the first and the sixth words that respectively explain 33.2% and 20.5% of the cluster. The rest of the cluster is explained by all the others words in negligible proportions. Each cluster is similarly a linear combination of the seven "words".

We can note that the clusters of travelers temporal profiles are more easily interpretable than the ones obtained by EM-algorithm. Clusters 1, 2, 5, 8 and 10 represent groups of people traveling during the peaks during the week and sometimes in Wednesdays noons. Cluster 4 represents travelers who use the public transportation during the peaks but also during the midday hours. Cluster 6 represents people traveling almost only during the afternoon (it may be people who use an other modal transport in the morning). Cluster 7 gathers users who only travel in off-peak. Finally, Clusters 3 and 9 contain travelers who have diffuse habits of travel during the day. Also note that the groups are more balanced as largest cluster contains only 15.5% of the users.

By observing the Table 3.1, we note that the clusters obtained with our method of NMF as a pre-processing tool do not correspond to clusters obtained by EM-algorithm. Indeed, the clusters obtained by EM are distributed between all the clusters obtained by NMF with k-means.

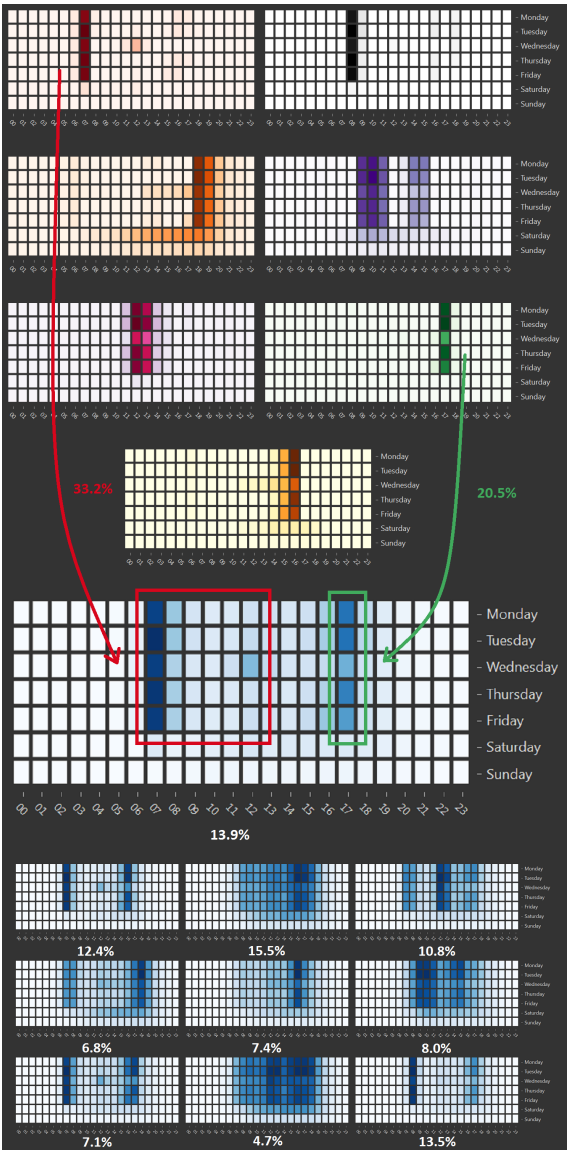


Figure 3.2: Up : "words" of the dictionary obtained by NMF; Middle : Decomposition in "words" of one of the clusters obtained by k-means; Down : The other clusters obtained by k-means on the reduced space

Table 3.1: Repartition of individuals between the clusters obtained by EM-algorithm and the clusters obtained by k-means on the reducted space.

		NMF + k-means									
		1	2	3	4	5	6	7	8	9	10
EM-algorithm	1	4%	6%	24%	13%	3%	7%	22%	3%	5%	12%
	2	31%	22%	11%	5%	4%	5%	8%	8%	2%	5%
	3	4%	12%	36%	5%	17%	2%	6%	1%	1%	14%
	4	6%	11%	28%	14%	6%	5%	8%	3%	4%	16%
	5	14%	9%	12%	6%	2%	4%	6%	6%	2%	40%
	6	15%	8%	13%	12%	6%	11%	9%	8%	7%	11%
	7	9%	14%	20%	6%	15%	10%	5%	10%	7%	4%
	8	4%	26%	11%	5%	4%	2%	5%	7%	1%	35%
	9	14%	16%	12%	22%	2%	5%	7%	8%	2%	11%
	10	3%	33%	15%	13%	12%	2%	2%	7%	1%	12%

3.5 Conclusion

This short empirical study seemed to confirm our idea that NMF could be a powerful tool to get efficient dimension reduction and clustering in transports data analysis. And indeed it was re-used since then, for example by [Tonnelier et al., 2018]. On the other hand, the analysis of [El Mahrsi et al., 2014a] has the elegance and interpretability of model-based clustering. This lead us to try to include this NMF-based approach in a model bases framework.

In the next chapter, we propose a mixture model including dimension reduction via factorization. Thus, the dimension reduction and clustering will no longer be seen as two independent steps, but will both be included in the model. Also, the likelihood of the model leads to the construction of criterion for the choice of H and K.

Chapter 4

Simultaneous Dimension Reduction and Clustering via the NMF-EM Algorithm

Abstract

Mixture models are among the most popular tools for clustering. However, when the dimension and the number of clusters is large, the estimation of the clusters become challenging, as well as their interpretation. Restriction on the parameters can be used to reduce the dimension. An example is given by mixture of factor analyzers for Gaussian mixtures. The extension of MFA to non-Gaussian mixtures is not straightforward. We propose a new constraint for parameters in non-Gaussian mixture model: the K components parameters are combinations of elements from a small dictionary, say H elements, with $H \ll K$. Including a nonnegative matrix factorization (NMF) in the EM algorithm allows us to simultaneously estimate the dictionary and the parameters of the mixture. We propose the acronym NMF-EM for this algorithm, implemented in the R package `nmfem`. This original approach is motivated by passengers clustering from ticketing data: we apply NMF-EM to data from two Transdev public transport networks. In this case, the words are easily interpreted as typical

slots in a timetable.

4.1 Introduction

With the growing ability to collect and store data in transports system, electricity consumption and more, urban computing is becoming a major tool in urban policy and planning [Zheng et al., 2014a]. For example, for transports system, there is a growing literature on ticketing and smart-card data processing in trains and buses [Morency et al., 2007; Pelletier et al., 2009; El Mahrsi et al., 2014a; Poussevin et al., 2014; Carel and Alquier, 2017; Tonnelier et al., 2018], bike-sharing systems [Randriamanamihaga et al., 2013; Côme and Oukhellou, 2014; Bouveyron et al., 2015; Hamon et al., 2015] or taxis [Peng et al., 2012]. Our objective in this chapter is to propose a clustering method for users, and for stations, that would be adapted to ticketing data collected by Transdev, a public network company. This method could be suitable for clustering structured high-dimensional data in other applications.

The range of machine learning and statistical tools used in urban computing is large. This goes from descriptive data-mining techniques as in [Morency et al., 2007] to statistical models as in [El Mahrsi et al., 2014a]. The model-based clustering approach in [El Mahrsi et al., 2014a] is actually close to our objective: journeys of a user are seen as realizations of multinomials random variables. The parameters of these distributions depends of the user only through the cluster the user belongs to. The complete model for journeys is thus a mixture of multinomials. The authors estimate the parameters and the clusters by the EM algorithm (see Chapter 9 in [Bishop, 2007] for an introduction; many R packages are available, `mcclust` [Scrucca et al., 2016] is extremely complete for clustering with Gaussian mixtures, `mixtools` [Benaglia et al., 2009] is a more generalist package covering other distributions, including multinomials). Model-based clustering was also used for transport data in [Côme and Oukhellou, 2014; Bouveyron et al., 2015] with nice results. However, there are some issues with this approach. When the dimension is large, the estimates are likely to have a large variance (curse of dimensionality). It might also be difficult to interpret clusters described by a huge number of parameters: it is indeed argued in [Carel and Alquier, 2017] that some profiles in [El Mahrsi et al., 2014a] are not easily interpretable. It seems then necessary to reduce the dimension, that is, to impose some restrictions on the parameters that will reduce the variance and increase the

interpretability.

Since the seminal work on model-based clustering [Wolfe, 1963], various examples of such restrictions have been proposed. We refer the reader to [Fraley and Raftery, 2002; Bouveyron and Brunet-Saumard, 2014; McNicholas, 2016a,b; Grün, 2018] for recent surveys on existing approaches (see also [McLachlan and Peel, 2004; Celeux et al., 2018a] for a more general overview on mixtures). A first approach is variable selection [Raftery and Dean, 2006]. This method is now well understood from an empirical perspective [Steinley and Brusco, 2008] as well as from a theoretical point of view [Maugis et al., 2009a,b]. See [Celeux et al., 2018b] for more recent advances and [Fop and Murphy, 2017] for a nice survey. The underlying assumption is that clusters differ only through a few variables. This assumption is satisfied in many examples presented in the aforementioned papers. However, it does not seem to be adapted to our case. The difference between two users, say a student and a retired person, is that the student has a regular travel schedule, while the retired person usually doesn't. This is a typical example of a strong structure that is not summarized by a small number of variables. Another approach for dimension reduction in mixtures is the mixture of factor analyzer (MFA) introduced in [Ghahramani and Hinton, 1996; McLachlan et al., 2003], see [Montanari and Viroli, 2010; McNicholas and Murphy, 2008; Murphy et al., 2017] for recent extensions. In MFA, the means and variances depends on the cluster, and the variance might be concentrated in some directions. This is more related to our objective, but this model was developed for mixture of Gaussians. The extension to non Gaussian mixtures is not direct, see however [Murray et al., 2014] for mixtures of skew-t factors analyzers. Travels patterns are modeled by mixture of multinomials in [El Mahrsi et al., 2014a].

In this chapter, we propose a new model that can be seen as an adaptation of MFA to mixture of distributions with nonnegative parameters (including multinomial distributions). The decomposition in Gaussian factors in MFA is replaced by a nonnegative matrix factorization (NMF). Introduced by [Lee and Seung, 1999], NMF rewrites columns of a given matrix with nonnegative entries as combinations of elements in a small dictionary. These elements are often referred to as "words". These words play a somewhat similar role to factors in MFA, even though the formalism is different. For example these words are not modeled as random variables. We provide an adaptation of the celebrated EM algorithm to this setting. We refer to this algorithm as NMF-EM. It is available as an R package, `nmfem`.

The chapter is organized as follows. In Section 4.2 we describe our model and the general form of NMF-EM. Motivated by the ticketing data, we provide the detailed form of the algorithm in the case of mixture of multinomials (Subsection 4.2.3). The clustering abilities of NMF-EM are compared to the ones of EM (without reduction of dimension) and of k-means in a short simulation study in Section 4.3. We finally present results on ticketing data provided by the Transdev Group in Section 4.4 (more details on this real data study can be found in the supplementary material).

4.2 Factorization of mixture parameters and the NMF-EM algorithm

4.2.1 Factorization of mixture parameters

Given a parametric family of distributions $(f_\vartheta)_{\vartheta\in\mathbb{R}^M}$, assume the observations $Y_1,\dots,Y_n$ are i.i.d from

$$\sum_{k=1}^{K} p_k f_{\theta_{\cdot,k}}(\cdot), \tag{4.1}$$

where each $\theta_{\cdot,k}\in\mathbb{R}^M$ is a column of a $K\times M$ matrix θ. For the sake of brevity, let $p=(p_1,\dots,p_K)$, which belongs to the simplex $\mathcal{S}_K=\{\rho\in\mathbb{R}_+^K:\rho_1+\cdots+\rho_K=1\}$. A way to rephrase this mixture model, which is useful for clustering purposes, is to introduce i.i.d hidden class variables: $Z_i=(Z_{i,1},\dots,Z_{i,K})\sim\mathcal{M}ult(p,1)$. Here, $\mathcal{M}ult(p,1)$ denotes the multinomial distribution, that is, the probability that Z_i is the k-th basis vector $(0,\dots,1,\dots,0)$ is given by p_k. Taking $Y_i\big|(Z_{i,k}=1)\sim f_{\theta_{\cdot,k}}(\cdot)$ implies that the Y_i's are actually i.i.d from (4.1).

In model-based clustering, estimation of the Z_i's allow us to assign each Y_i to a cluster k while the estimation of $\theta_{\cdot,k}$ provides a summary of the information on location, scale and shape of cluster k. Still, as argued in the introduction, when the dimension M is too large, this information can be unreliable and difficult to interpret. Many dimension reduction methods were proposed, among them MFA for mixtures of Gaussians. A standard mixture of Gaussian in $\mathbb{R}^d$ is $Y_i\big|(Z_{i,k}=1)\sim\mathcal{N}(\mu_k,\Sigma_k)$, the simplest form of MFA is given by $Y_i\big|(Z_{i,k}=1)\sim\mathcal{N}(\mu_k,\Lambda_k\Lambda_k^T+\Psi)$, where Λ_k is a $d\times H$ matrix with $H\ll d$ and Ψ is some diagonal matrix with positive diagonal entries. Thus, the estimation of the $d\times d$ matrix Σ_k is essentially reduced to the estimation of the much smaller $H\times d$ matrix Λ_k. An interpretation

of this model is that Y_i depends not only on the hidden variable Z_i but also on hidden factors $X_i \sim \mathcal{N}(0, I_H)$: $\mathbb{E}(Y_i|X_i = x, Z_{i,k} = 1) = \Lambda_k x + \mu_k$. See the references given in the introduction, e.g Section 5 in [Bouveyron and Brunet-Saumard, 2014]. This model provides reduction of dimension and has a nice interpretation, but it is not obvious how to extend it beyond Gaussian variables.

In the case where of multinomial distributions, and more generally in the case where the parameters ϑ of $(f_\vartheta)_{\vartheta \in \mathbb{R}^M}$ are actually nonnegative, one could think of restrictions on the mixture parameters matrix θ that would similarly involve a small number H of hidden factors. But one has to be careful: Gaussian hidden factors would in general not generate nonnegative parameters. In a celebrated paper [Lee and Seung, 1999], Lee and Seung proposed a dimension reduction tool for matrices with nonnegative entries: NMF (nonnegative matrix factorization). The idea is to factorize a $K \times M$ matrix θ as

$$\underbrace{\begin{pmatrix} \theta_{1,1} & \dots & \theta_{1,K} \\ \vdots & \ddots & \vdots \\ \theta_{M,1} & \dots & \theta_{M,K} \end{pmatrix}}_{\theta} = \underbrace{\begin{pmatrix} \Phi_{1,1} & \dots & \Phi_{1,H} \\ \vdots & \ddots & \vdots \\ \Phi_{M,1} & \dots & \Phi_{M,H} \end{pmatrix}}_{\Phi} \underbrace{\begin{pmatrix} \Lambda_{1,1} & \dots & \Lambda_{1,K} \\ \vdots & \ddots & \vdots \\ \Lambda_{H,1} & \dots & \Lambda_{H,K} \end{pmatrix}}_{\Lambda} \tag{4.2}$$

with $H \leq K, M$, under the assumption that all the entries in Φ and Λ are nonnegative. When $H \ll K, M$, the dimension reduction is substantial. NMF rewrites columns of a given matrix as positive combinations of elements, or words, in a small dictionary Λ. It turns out that this dictionary is often easily interpretable. NMF was successfully used as a data mining tool in document clustering [Xu et al., 2003; Shahnaz et al., 2006], collaborative filtering and recommender systems on the Web [Koren et al., 2009; Luo et al., 2014], dictionary learning for images [Lee and Seung, 1999], topic extraction in texts [Paisley et al., 2014] or time series recovering [Mei et al., 2017], among others. It was also used as a data mining tool for transports data by [Hamon et al., 2015; Peng et al., 2012; Poussevin et al., 2014; Tonnelier et al., 2018] and our previous work [Carel and Alquier, 2017]: we "compressed" the data $Y_1, \dots, Y_n$ using an NMF and then used a (model-free) clustering algorithm on the compressed observations. The improvement in terms of interpretability with respect to [El Mahrsi et al., 2014a] was substantial. However, this approach was completely *ad hoc*: there are many possible criterion to approximate NMF: the Poisson-likekihood [Lee and Seung, 1999, 2001], the quadratic criterion or Gaussian-likelihood [Boyd et al., 2011; Lee and Seung, 2001], the

Ikuro-Saito divergence [Févotte et al., 2009]... In a model-free approach, the choice of the criterion is difficult. The mixture model (4.1) leads to a natural criterion: the likelihood.

We are finally in position to define our model: we use NMF as a restriction on nonnegative parameters in mixture models. That is, $Y_1, \dots, Y_n$ are i.i.d from

$$g_{p,\Phi,\Lambda}(\cdot) = \sum_{k=1}^{K} p_k f_{(\Phi\Lambda)_{\cdot,k}}(\cdot) \tag{4.3}$$

or equivalently, $Y_i|(Z_{i,k}=1)$ is drawn from $f_{(\Phi\Lambda)_{\cdot,k}}$ and $Z_i \sim \mathcal{M}ult(p,1)$. The model is parametrized by $p \in \mathcal{S}_K$, $\Lambda \in \mathbb{R}_+^{M\times H}$ and $\Phi \in \mathbb{R}_+^{H\times K}$. For short, put $Y=(Y_1,\dots,Y_n)$ and $Z=(Z_1,\dots,Z_n)$. The log-likelihood is given by

$$\ell(\Phi,\Lambda,p|Y) = \sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} p_k f_{(\Phi\Lambda)_{\cdot,k}}(Y_i)\right).$$

This model can be seen as offering a connection between "model-free clustering" relying on NMF or spectral clustering as in [Ding et al., 2005; Yang et al., 2016] and model-based clustering. Unrestricted mixture models can of course be seen as a special case by taking $H=K$ and $\Lambda = I_K$.

Remark 4.2.1 *The first example we have in mind is the mixture of multinomials that was used in [El Mahrsi et al., 2014a] to model travel patterns. As our main application, this example is detailed in Subsection 4.2.3. Note a similarity with the Latent Dirichlet Allocation (LDA) model in [Blei et al., 2003]: LDA involves two layers of multinomials. First, a topic is a multinomial on words, then a text is described by a multinomial on topics. However, LDA does not involve clusters of similar texts. It was not designed as a clustering tool.*

Beyond multinomials, any distribution with nonnegative parameters can be used. Consider sales analysis. Assume that the owner of a supermarket observes, for each good $m \in \{1,\dots,M\}$ and each customer $i \in \{1,\dots,n\}$, the number of items of m bought by i during one year: $Y_{i,m}$. Put $Y_i = (Y_{i,1},\dots,Y_{i,M})$. We propose the model $Y_{i,m}|(Z_{i,k}=1) \sim \mathcal{P}(\theta_{m,k})$, a Poisson distribution. The column $\theta_{\cdot,k}$ is the "standard basket" of any customer i in cluster k. But the number of goods is so huge that the estimation of standard baskets is subject to a large variance, and prevents their interpretation. In (4.2), the columns of Φ are representations of columns of θ in a smaller subspace. It is likely that substitutable goods are gathered. This example is simply a model-based version of the NMF

analysis used in [Koren et al., 2009; Luo et al., 2014], see also [Wu, 2007] for an early application on the Netflix prize data. Sales analysis, customer clustering and recommender systems are indeed applications of NMF that generated a huge number of publications. More examples could include exponential or gamma mixtures in survival analysis, or Pareto and Weibull mixtures in extreme analysis.

We now discuss the adaptation of the EM algorithm to this parameter restriction.

4.2.2 The NMF-EM algorithm

We recall the expression of the completed likelihood

$$\ell(\Phi, \Lambda, p|Y, Z) = \sum_{i=1}^{n}\sum_{k=1}^{K} Z_{i,k} \log\left(p_k f_{(\Phi\Lambda)_{\cdot,k}}(Y_i)\right).$$

A step of the EM algorithm, given current parameters $(\Phi^{(c)}, \Lambda^{(c)}, p^{(c)})$ is as follows:

$$\begin{aligned}
\textbf{E-step: } Q^{(c)}(\Phi, \Lambda, p) &= \mathbb{E}_{\Phi^{(c)},\Lambda^{(c)},p^{(c)}}[\ell(\Phi, \Lambda, p|Y, Z)|Y] \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} \mathbb{E}_{\Phi^{(c)},\Lambda^{(c)},p^{(c)}}[Z_{i,k}|Y] \log\left(p_k f_{(\Phi\Lambda)_{\cdot,k}}(Y_i)\right) \\
\text{and } t_{i,k}^{(c)} &:= \mathbb{E}_{\Phi^{(c)},\Lambda^{(c)},p^{(c)}}[Z_{i,k}|Y] \\
&= \frac{p_k^{(c)} f_{(\Phi^{(c)}\Lambda^{(c)})_{\cdot,k}}(Y_i)}{\sum_{k'=1}^{K} p_{k'}^{(c)} f_{(\Phi^{(c)}\Lambda^{(c)})_{\cdot,k'}}(Y_i)}. \qquad (4.4)
\end{aligned}$$

$$\textbf{M-step: } (\Phi^{(c+1)}, \Lambda^{(c+1)}, p^{(c+1)}) := \underset{\Phi_{j,h},\Lambda_{h,k}\geq 0}{\arg\max}\, Q^{(c)}(\Phi, \Lambda, p). \qquad (4.5)$$

Obviously, the challenging step is the M-step. While we obviously have, for $k \in \{1, \ldots, K\}$,

$$p_k^{(c+1)} = \frac{\sum_{i=1}^{n} t_{i,k}^{(c)}}{\sum_{i=1}^{n}\sum_{k'=1}^{K} t_{i,k'}^{(c)}}, \qquad (4.6)$$

the non-negativity constraint on Φ and Λ makes the optimization with respect to these two matrices much harder. This is where one has to use

ideas from the NMF literature. Many options might be possible, depending on the form of $f_\vartheta(\cdot)$. The most commonly used algorithm is the so-called multiplicative update, an alternating optimization method with respect to Φ and Λ, that was proposed in the seminal papers [Lee and Seung, 1999, 2001]. Other algorithms include ADMM [Boyd et al., 2011; Sun and Fevotte, 2014], alternating projected gradient [Lin, 2007], and for Bayesian approaches, Monte-Carlo methods [Paisley et al., 2014] and variational approximations [Alquier and Guedj, 2017]. A numerical comparison of many algorithms can be found in [Lin, 2007]. In practice, the multiplicative update is efficient in many settings and is very simple to use: it does not depend on any tuning parameter such as the step size in gradient based method. So this is the method we will use from now. This method iterates a step in Φ, and a step in Λ. Each step is shown to improve the fit criterion in [Lee and Seung, 2001]. Note that the author claims that it also leads to convergence, but as argued in [Gonzalez and Zhang, 2005] the proof of this fact is actually incomplete. We explicit the multiplicative update in the case of mixture of multinomials below.

4.2.3 The NMF-EM algorithm for mixture of multinomials

In [El Mahrsi et al., 2014a] the authors modeled a passenger temporal profile by a mixture of multinomial distribution. The time and days of smart card validations of a passenger i are recorded over a period of time (e.g. 1 month). The numbers of journeys, N_i, is not our variable of interest, and will be considered as deterministic. We obtain as a result a vector $Y_i = (Y_{i,1}, \dots, Y_{i,M})^T \in \mathbb{R}^M$ where each coordinate represents the number of travels at a given pair time-day during the considered period. Note that of course $\sum_{k=1}^M Y_{i,k} = N_i$, let $N = \sum_{i=1}^n N_i$ be the total number of journeys. We consider a hourly grid, that is, Mon-12am, Mon-1am, etc... to Sun-11pm, with means that $M = 7 \times 24 = 168$. An example of a traveler profile is given in Figure 4.1.

It is natural to assume that there are clusters of passengers with rather similar profiles: for examples, employees with similar work hours or students in the same University are likely to commute at similar times. We follow the previous construction: we define the hidden cluster variables, $Z_i \sim \mathcal{M}ult(p, 1)$ for some $p \in \mathcal{S}_K$. We then set

$$Y_i \big| (Z_{i,k} = 1) \sim \mathcal{M}ult(\theta_{\cdot,k}, N_i)$$

where $\theta_{\cdot,k} \in \mathcal{S}_M$ is the k-th column of an $M \times K$ matrix θ that satisfies

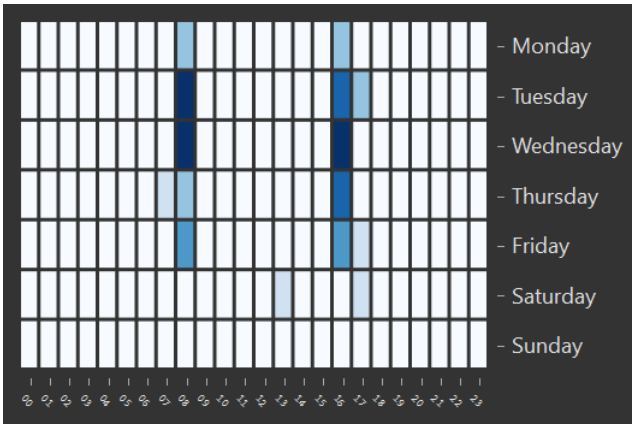


Figure 4.1: Temporal profile of a network user, taken from the data described in Section 4.4. Opacity is proportional to the number of smart-card validations. This user travels generally at 8 a.m. and 4 p.m. on weekdays.

$\theta = \Phi\Lambda$ where Φ is $M \times H$ and Λ is $H \times K$ for some $H \leq M, K$. The log-likelihood is given by

$$\ell(\Phi, \Lambda, p|Y) = \sum_{i=1}^{n} \log \left\{ \sum_{k=1}^{K} p_k \left[N_i! \prod_{j=1}^{M} \frac{(\Phi\Lambda)_{j,k}^{Y_{i,j}}}{Y_{i,j}!} \right] \right\}.$$

Note that a simple way to ensure $\theta_{\cdot,k} \in \mathcal{S}_M$ is to impose similar constraints on the columns of Φ and Λ. So we define $\mathcal{M}_{M,H,K}$ as the set of all pairs (Φ, Λ) of matrices $M \times H$ and $H \times K$ respectively, with $\Phi_{\cdot,k}, \Lambda_{\cdot,j} \in \mathcal{S}_M$ for any k and j. Note that we actually have $H(M-1) + K(H-1) + K - 1$ degrees of freedom for the parameters of our model $(\Phi, \Lambda) \in \mathcal{M}_{M,H,K}$ and $p \in \mathcal{S}_K$. This knowledge is required for computing model selection criterion such as AIC (see the discussion on model selection in Subsection 4.2.4 below).

Let $(\hat{\Phi}, \hat{\Lambda}, \hat{p})$ denote the MLE, that is, a maximizer of $\ell(\Phi, \Lambda, p|Y)$ with respect to $(\Phi, \Lambda) \in \mathcal{M}_{M,H,K}$ and $p \in \mathcal{S}_K$. We make explicit the NMF-EM algorithm to approximate $(\hat{\Phi}, \hat{\Lambda}, \hat{p})$.

From (4.4), values $t_{i,k}^{(c)}$ are given by

$$t_{i,k}^{(c)} = \frac{p_k^{(c)} N_i! \prod_{j=1}^{M} \frac{\left(\sum_{h=1}^{H} \Phi_{j,h}^{(c)} \Lambda_{h,k}^{(c)}\right)^{Y_{i,j}}}{Y_{i,j}!}}{\sum_{k'=1}^{K} p_{k'}^{(c)} N_i! \prod_{j=1}^{M} \frac{\left(\sum_{h=1}^{H} \Phi_{j,h}^{(c)} \Lambda_{h,k'}^{(c)}\right)^{Y_{i,j}}}{Y_{i,j}!}}$$

$$= \frac{p_k^{(c)} \prod_{j=1}^{M} \left(\sum_{h=1}^{H} \Phi_{j,h}^{(c)} \Lambda_{h,k}^{(c)}\right)^{Y_{i,j}}}{\sum_{k'=1}^{K} p_{k'}^{(c)} \prod_{j=1}^{M} \left(\sum_{h=1}^{H} \Phi_{j,h}^{(c)} \Lambda_{h,k'}^{(c)}\right)^{Y_{i,j}}}.$$

We have

$$Q^{(c)}(\Phi, \Lambda, p) = \sum_{i=1}^{n} \sum_{k=1}^{K} t_{i,k}^{(c)} \log \left(p_k N_i! \prod_{j=1}^{M} \frac{\left(\sum_{h=1}^{H} \Phi_{j,h} \Lambda_{h,k}\right)^{Y_{i,j}}}{Y_{i,j}!} \right)$$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} t_{i,k}^{(c)} \Bigg[\log(p_k) + \log(N_i!) + \sum_{j=1}^{M} \left(Y_{i,j} \log \left(\sum_{h=1}^{H} \Phi_{j,h} \Lambda_{h,k} \right) - \log(Y_{i,j}!) \right) \Bigg].$$

As stated in (4.6), $p_k^{(c+1)} \propto \sum_{i=1}^{n} t_{i,k}^{(c)}$, and

$$(\Phi^{(c+1)}, \Lambda^{(c+1)}) = \underset{(\Phi,\lambda) \in \mathcal{M}_{M,H,K}}{\arg\max} \sum_{k=1}^{K} \sum_{j=1}^{M} \left(\sum_{i=1}^{n} Y_{i,j} t_{i,k}^{(c)} \right) \log \left(\sum_{h=1}^{H} \Phi_{j,h} \Lambda_{h,k} \right).$$

Put $M_{j,k}^{(c)} = \sum_{i=1}^{n} Y_{i,j} t_{i,k}^{(c)}$ for short. The previous equation becomes

$$(\Phi^{(c+1)}, \Lambda^{(c+1)}) = \underset{(\Phi,\Lambda) \in \mathcal{M}_{M,H,K}}{\arg\max} \sum_{k=1}^{K} \sum_{j=1}^{M} M_{j,k}^{(c)} \log \left(\sum_{h=1}^{H} \Phi_{j,h} \Lambda_{h,k} \right). \quad (4.7)$$

The maximization in (4.7) is equivalent to the minimization of

$$D(M^{(c)}||\Phi\Lambda) := -\sum_{k=1}^{K}\sum_{j=1}^{M}\left\{M_{j,k}^{(c)}\log\left(\sum_{h=1}^{H}\Phi_{j,h}\Lambda_{h,k}\right)-\sum_{h=1}^{H}\Phi_{j,h}\Lambda_{h,k}\right\}. \tag{4.8}$$

Indeed, for $(\Phi,\Lambda)\in\mathcal{M}_{M,H,K}$ we have

$$\sum_{k=1}^{K}\sum_{j=1}^{M}\sum_{h=1}^{H}\Phi_{j,h}\Lambda_{h,k}=\sum_{j=1}^{M}\sum_{k=1}^{K}(\Phi\Lambda)_{j,k}=\sum_{j=1}^{M}1=M$$

that does not depend on (Φ,Λ). The multiplicative algorithm in [Lee and Seung, 2001] was actually introduced to minimize $D(M^{(c)}||\Phi\Lambda)$. So we just use the update steps of [Lee and Seung, 2001] (steps 9 and 10 in Algorithm 7 below) followed by a renormalization of the matrices Φ and Λ in order to ensure that the columns remain in the parameter space (steps 10 and 12). This completes the derivation of the NMF-EM algorithm for mixture of multinomials, see: Algorithm 7 page 90. We implemented this algorithm for the R software [Ihaka and Gentleman, 1996], the package `nmfem` can be found on the CRAN repository.

4.2.4 Discussion on the choice of H and K

The choice of K is not a straightforward issue in mixture models. *A fortiori* the choice of the pair (H,K) is not easier.

From the likelihood and the degrees of freedom above we can derive the AIC and BIC criteria

$$\text{AIC}=\ell(\hat{\Phi},\hat{\Lambda},\hat{p}|Y)-\frac{H(M-1)+K(H-1)+K-1}{2}$$
$$\text{BIC}=\ell(\hat{\Phi},\hat{\Lambda},\hat{p}|Y)-\frac{[H(M-1)+K(H-1)+K-1]\log(N)}{2}$$

that are widely used in practice. Among the papers mentioned above, BIC is used for choosing the number of clusters of users in [Bouveyron et al., 2015]. However, the consistency of AIC and BIC depend on conditions that might not be satisfied in mixture models. Other criteria more suitable for mixtures were investigated, like NEC and variants [Biernacki et al., 1999]. The slope heuristic [Baudry et al., 2012] is known to give nice results in practice, and can also be show to be consistent in some settings [Arlot and Massart, 2009]. It is actually used in [El Mahrsi et al., 2014a] for mixtures of multinomials.

Algorithm 7 NMF-EM

Fix $\epsilon > 0$. Choose arbitrary $\Phi^{(0)}$, $\Lambda^{(0)}$ and $p^{(0)}$; $c := 0$, CRIT $:= \infty$.
while $|\ell(\Phi^{(c)}, \Lambda^{(c)}, p^{(c)}) - \text{CRIT}| > \epsilon$ **do**
  CRIT $:= \ell(\Phi^{(c)}, \Lambda^{(c)}, p^{(c)})$.
  For all $i \in \{1, \dots, n\}$ and $k \in \{1, \dots, K\}$,

$$t_{i,k}^{(c)} := \frac{p_k^{(c)} \prod_{j=1}^{M} \left(\sum_{h=1}^{H} \Phi_{j,h}^{(c)} \Lambda_{h,k}^{(c)} \right)^{Y_{i,j}}}{\sum_{k'=1}^{K} p_{k'}^{(c)} \prod_{j=1}^{M} \left(\sum_{h=1}^{H} \Phi_{j,h}^{(c)} \Lambda_{h,k'}^{(c)} \right)^{Y_{i,j}}} \text{ and } p_k^{(c+1)} =: \frac{\sum_{i=1}^{n} t_{i,k}^{(c)}}{\sum_{i=1}^{n} \sum_{k'=1}^{K} t_{i,k'}^{(c)}}.$$

  $\forall j, k \quad M_{j,k}^{(c)} = \sum_{i=1}^{n} Y_{i,j} t_{i,k}^{(c)}$.
  Initialization of Φ and Λ (arbitrarily), $q := \infty$.
  while $|Q^{(c)}(\Phi, \Lambda, p^{(c+1)}) - q| > \epsilon$ **do**
    $q := Q^{(c)}(\Phi, \Lambda, p^{(c+1)})$.
    $\forall h, k \quad \Lambda_{h,k} \leftarrow \Lambda_{h,k} \frac{\sum_j \Phi_{j,h} M_{j,k}^{(c)} / (\Phi\Lambda)_{j,k}}{\sum_j \Phi_{j,h}}$
    $\forall h, k \quad \Lambda_{h,k} \leftarrow \frac{\Lambda_{h,k}}{\sum_{k'} \Lambda_{h,k'}}$
    $\forall j, h \quad \Phi_{j,h} \leftarrow \Phi_{j,h} \frac{\sum_k \Lambda_{h,k} M_{j,k}^{(c)} / (\Phi\Lambda)_{j,k}}{\sum_k \Lambda_{h,k}}$
    $\forall j, h \quad \Phi_{j,h} \leftarrow \frac{\Phi_{j,h}}{\sum_{h'} \Phi_{j,h'}}$
  end while
  $(\Phi^{(c+1)}, \Lambda^{(c+1)}) := (\Phi, \Lambda)$.
  $c := c + 1$.
end while

An important point is that our criterion should actually depend on the objective we have in mind. In regular models, AIC finds the optimal balance between bias and variance, while BIC identifies the true model, when there is one. These two objectives are usually not compatible [Yang, 2005]. In our collaboration with Transdev, interpretability of the results was actually one of the main objectives. We will use the slope heuristic in what follows.

4.3 Simulation study

In this section, we illustrate the dimension-reduction effect of NMF-EM on synthetic data. As our main interest is here clustering, we will compare the "pairwise misclassification rate" of NMF-EM with both of the EM and k-means algorithms – that is, the proportion of pairs (i, j) of individuals that are either assigned to the same component by the algorithm while they were actually generated from different components, or assigned to different components while they were simulated from the same.

The experimental setting is as follow: the dimension is $m = 100$, for each experiment we generate H_0 words in $\mathbb{R}^m$ from a uniform distribution and then $K = 10$ parameters $\theta_{\cdot,1}, \ldots, \theta_{\cdot,K}$ as linear combinations of these H_0 words – the coefficients of each parameters are independently drawn from a Dirichlet distribution $\mathcal{D}(\alpha, \ldots, \alpha)$. We finally draw $n = 1500$ individuals from the corresponding mixture of multinomials with uniform weights.

We compare NMF-EM with $H = 4$, EM (without reduction of dimension) and k-means in various settings: in the case $H_0 = 4$, where the dimension reduction in NMF-EM is actually correct, and $H_0 = 8$ - this case is less favorable to NMF-EM with $H = 4$ as it reduces the dimension too much. We also use different values for α, leading to different shapes for the set of parameters $\{\theta_{\cdot,1}, \ldots, \theta_{\cdot,K}\}$. The results are in Tables 4.1 and 4.2. We note that the misclassification rate of our algorithm NMF-EM is smaller than the ones of EM and k-means algorithms for $\alpha \in [0.2, 0.9]$. It means that our algorithm outperform the others in the case of distinct clusters with some being linear combinations of others. Indeed, when α is too small or too close or bigger than 1 the clusters are identical or too similar, and then hard to detect.

Table 4.1: Pairwise misclassification rate of the algorithms on simulated data when $H_0 = 4$ ($m = 100$, $n = 1500$, $N = 150$, $K = 10$).

	α = .01	α = .1	α = .2	α = .3	α = .4	α = .5	α = .6
NMF-EM	9.5%	6.3%	**4.7%**	**5.0%**	**4.9%**	**5.3%**	**5.9%**
EM	8.4%	6.8%	5.0%	5.7%	5.6%	5.9%	6.7%
k-means	**6.4%**	**5.9%**	5.3%	5.5%	5.6%	6.0%	6.2%
	α = .7	α = .8	α = .9	α = 1.0	α = 1.1	α = 1.2	α = 1.3
NMF-EM	**6.5%**	**6.7%**	**6.7%**	7.6%	7.3%	7.5%	8.8%
EM	7.2%	7.3%	7.0%	7.7%	8.1%	8.0%	8.9%
k-means	6.6%	6.7%	6.8%	**7.0%**	**7.2%**	**7.1%**	**7.5%**

Table 4.2: Pairwise misclassification rate of the algorithms on simulated data when $H_0 = 8$ ($m = 100$, $n = 1500$, $N = 150$, $K = 12$).

	α = .01	α = .1	α = .2	α = .3	α = .4	α = .5	α = .6
NMF-EM	5.2%	4.5%	5.8%	5.8%	6.5%	6.9%	8.1%
EM	4.3%	3.1%	**3.1%**	**3.8%**	5.0%	6.1%	6.1%
k-means	**3.8%**	**3.1%**	3.4%	4.0%	**4.8%**	**5.5%**	**5.6%**
	α = .7	α = .8	α = .9	α = 1.0	α = 1.1	α = 1.2	α = 1.3
NMF-EM	8.2%	9.1%	10.0%	10.5%	10.3%	11.3%	11.5%
EM	6.4%	7.2%	7.5%	7.5%	8.3%	8.6%	8.5%
k-means	**5.8%**	**6.3%**	**6.3%**	**6.5%**	**6.8%**	**7.0%**	**6.9%**

So, when the intrinsic dimension is small enough, NMF-EM really improves the clustering ability of EM. In any case, our main claim is that it leads to easily interpretable clusters, a fact that will be illustrated in the next section.

4.4 Application to ticketing data

4.4.1 Description of the data

The data used in our study are the validations made during the month of September 2015 on one Transdev network in a medium size city. Ticketing data are the information obtained at each transaction made by a smart card on a validator system. For privacy reasons it is not possible to connect each validation to the user who made it. The feature that allows us to realize our study and create temporal profiles is a card number which is encrypted, and re-initialized every three months. It is thus impossible to follow the long-term behavior of a user. This is the reason why we focus on a one month period. This period (September) have been chosen because it has no vacation nor bank holiday. During September 2015, more than $4,000,000$ check-ins have been made on the network by $232,430$ passengers.

The data are aggregated so that for each traveler, for each day of the week (Monday to Sunday) and each hour (00 to 23), we have the number of validation during the studied period. A passenger profile is thus defined by $24 * 7 = 168$ features. Figure 4.1 page 87 already provided an example of a temporal profile of one of the users. This traveler uses mainly the network at 8 a.m and 4 p.m.

Remark 4.4.1 *We used the same strategy to create stations profiles: for each station, for each day of the week and each hour of the day, we know*

the number of validations that occurred at this station during the study period. In Figure 4.2, we show the temporal profile of the station "Palais de Justice" (courthouse), a tramway station in the city center. This station has travelers all day long, but knows an attendance peak every day from 4 *to* 6 *p.m. The results of this analysis are provided in the supplementary*

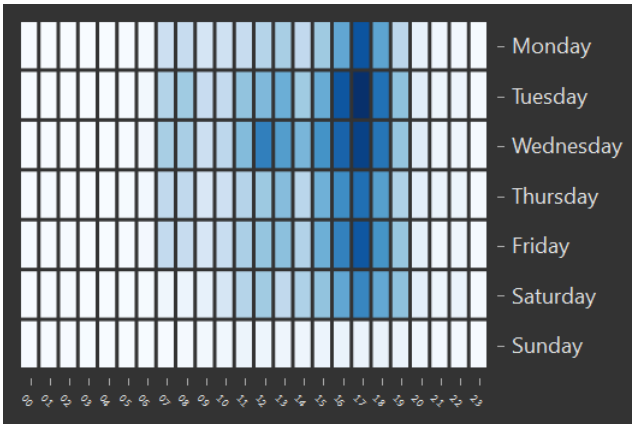


Figure 4.2: Temporal profile of station "Palais de Justice". Opacity is proportional to the number of smart-card validations. This station has travelers all day long, but knows an attendance peak every afternoon.

material.

In order to avoid users who would not use their smart card enough to exhibit a clear pattern, data have been cleaned. We define a "regular card holder" as a card holder who

- travelled on at least four days during September 2015 (so in particular we have $N_i \geq 4$);
- made their first boarding after 4 a.m each day at the same station 50% of the time.

We only kept regular card holders for our analysis. After this cleaning step, we end up with $72,359$ profiles of passengers, which represent a bit more than $3,000,000$ check-ins – that means 31% of passengers represent 75% of check-ins. We also have 475 stations profiles. These data are provided in the `nmfem` package.

4.4.2 Passenger profile clustering

We first focus on passenger profile clustering. This allows us to create groups of people that have similar temporal habits. The method used to create these clusters is the NMF-EM algorithm from Subsection 4.2.3.

To choose the parameters H and K, we begin with the analysis of the log-likelihood of our model when $H = K$ for $K = 2 \dots 30$. Note that the estimation of the model in this case can be made by the usual EM algorithm for multinomial mixture model. Figure 4.3 shows the evolution of the log-likelihood as a function of K. This function clearly exhibits a linear

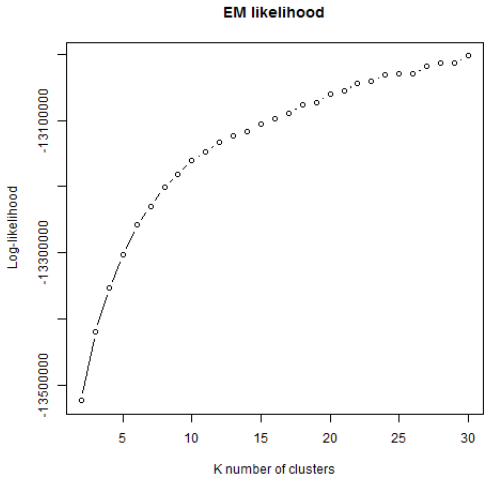


Figure 4.3: The log-likelihood as a function of K under $H = K$. By slope heuristic, we chose $K = 10$ clusters.

behavior when $K \geq 10$. Thus, the slope heuristic suggests considering $K = 10$.

Now keeping $K = 10$ fixed, we chose the value of H in the same way. First, we plot the log-likelihood as a function H in Figure 4.4. By using again the slope heuristic method, we choose $H = 5$.

The $H = 5$ words and the $K = 10$ clusters are represented in Figure 4.5 and in Figure 4.7 respectively. Remember that each cluster can be decomposed as a convex combination of words, some of them might have a null weight. For example, Figure 4.6 shows how the parameter of Cluster 5 can be written as a convex combination of words 4 and 2.

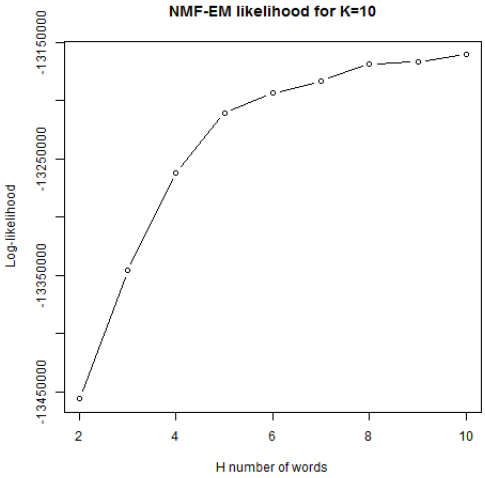


Figure 4.4: The log-likelihood as a function of $H \in \{2, \ldots, K\}$ under $K = 10$. By slope heuristic, we chose $H = 5$ words.

The interpretation of the words is direct:

1. Word 1: travels between 6 a.m and 7 a.m.
2. Word 2: diffuse component during off-peak periods (i.e. from 9 a.m to 4 p.m).
3. Word 3: travels at school hours. Indeed it is composed of travel between 7 and 8 a.m and between 4 and 5 p.m, except on Wednesdays, when the afternoon travel is replaced by one at noon.
4. Word 4: travels between 8 and 9 a.m.
5. Word 5: late afternoon peak, from 5 to 7 p.m, and Wednesdays and Saturdays afternoon.

We now attempt an interpretation of the clusters:

1. Clusters 1, 3, 4 and 6 present high travel probabilities in the morning and in the afternoon except Wednesdays where the afternoon travel is replaced by a higher probability of travel around noon. These four clusters are typical of French schools and high-schools hours. The main differences are:

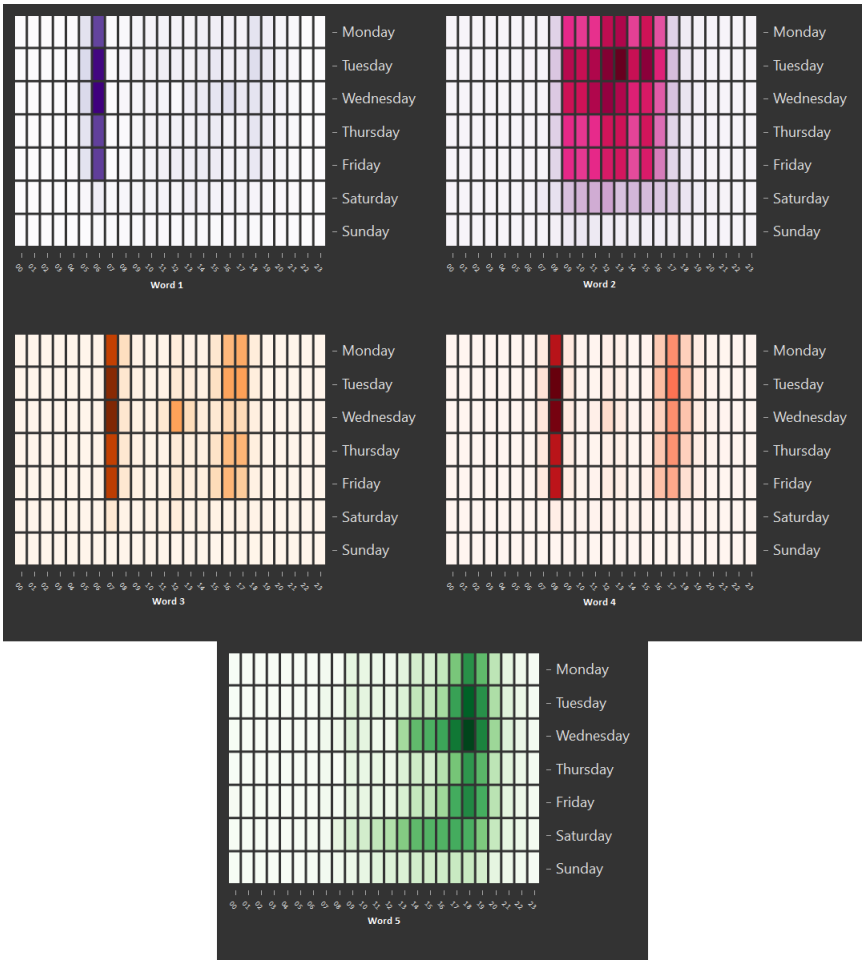


Figure 4.5: Words obtained by NMF-EM on users data with $K = 10$ and $H = 5$. The first word contains the travel pattern of 6 a.m. on weekdays. The second contains mainly the travel pattern of off-peak period on weekdays. The third and fourth words contains mostly travel patterns at respectively 7 and 8 a.m. on weekdays. The fifth word contains travel patterns of the afternoon peak during weekdays and Wednesdays and Saturdays afternoons.

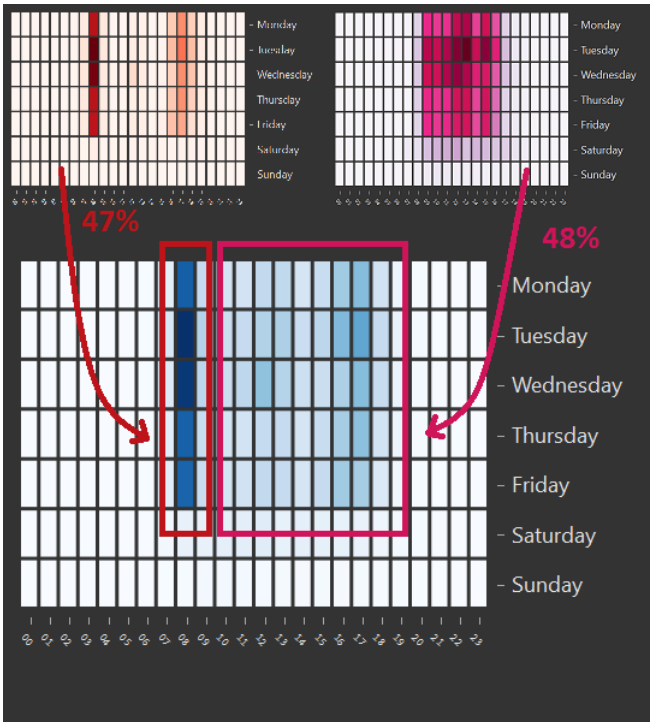


Figure 4.6: Decomposition of cluster 5 from words 4 and 2. This cluster is 47% composed of the fourth word and 48% of the second. The remaining 5% are a mixture of the other words.

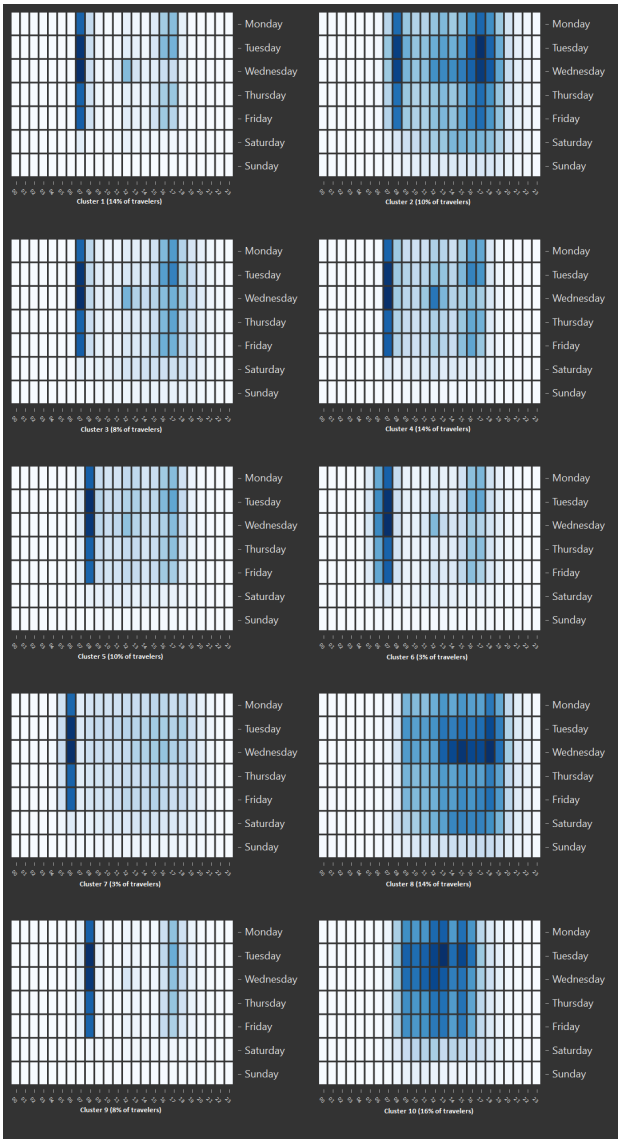


Figure 4.7: Clusters obtained by NMF-EM on users data with $K = 10$ and $H = 5$. Clusters 1, 3, 4 and 6 have scholar travel schedule. Clusters 2, 8 and 10 have diffuse travel habits. Clusters 5, 7 and 9 have strong morning habits with slighter off-peak or afternoon peak patterns. We refer the reader to the main paragraph for more details.

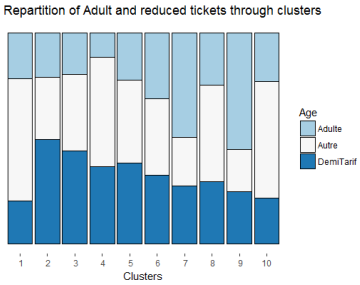
   (a) Cluster 1: travels at 7 a.m and around 4 or 5 p.m.
   (b) Cluster 3: travel a bit more at 8 a.m.
   (c) Cluster 4: travelers are less susceptible to travel after 5 p.m.
   (d) Cluster 6: travels at 6 and 7 a.m.

2. Cluster 5: travels at 8 a.m and at 4 or 5 p.m.
3. Cluster 7: travels mainly at 6 a.m.
4. Cluster 9: travels at 8 a.m and at 5 p.m.
5. Clusters 2, 8 and 10: diffuse travel habits.
   (a) Cluster 2: travels Mondays to Saturdays from 7 a.m to 7 p.m with highest probabilities at 8 a.m and 5 p.m Mondays to Fridays.
   (b) Cluster 8: diffuse travels Mondays to Saturdays from 9 a.m to 7 p.m.
   (c) Cluster 10: travels Mondays to Fridays from 9 a.m to 4 p.m.

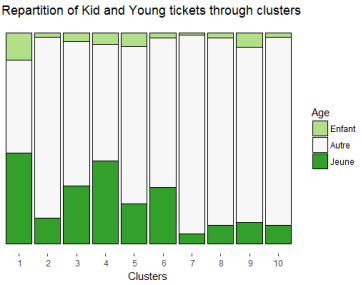
As written above, we have no personal information in our data. Therefore, we are not able to describe individually the users in each cluster. However, for each transaction made, we have the encrypted card number and the transport ticket used. So we can recover for each card the most used transport ticket during the period. This provides interesting information as some schemes are associated to age ranges (Young, Senior...) and to time periods (Unit, Annual or Monthly Subscription). Let us now provide the description of each cluster in terms of age ranges (Figures 4.3a to 4.3c in Table 4.3).

Adults are more present in clusters 7 and 9, that are clusters with check-ins mostly in the morning. People benefiting from half-price are present in every cluster but with highest rates in clusters 2, 3, 4 and 5. Children (4 to 6) are not very present on the network, but they are more represented in clusters 1, 5 and 9. Young travelers (6 to 25) are more present in clusters 1 and 4. These clusters correspond to scholar time slot. In clusters 8 and 10 there are large rate of seniors and free travelers. As these clusters have profiles of diffuse travels during the week and as free travelers are unemployed or low salaries people, these regroupments make sense.

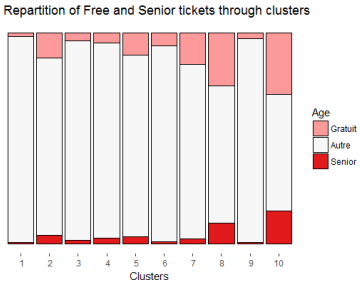
Figure 4.8 shows the repartition of transport ticket type through clusters. Unit products are more used in clusters 8 and 10 that are clusters



(a) Adults (“Adult”) and Reduced tickets (“DemiTarif”)



(b) Children (“Enfant”) and Youngs (“Jeune”)



(c) Free travelers (“Gratuit”) and Elderly (“Senior”)

Table 4.3: Age range analysis of the clusters

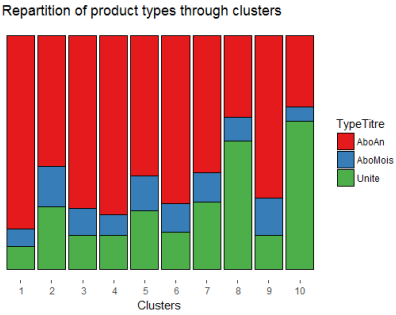


Figure 4.8: Transportation ticket type analysis of the clusters.

with a lots of seniors and free travelers. As they don’t have obligations,

they likely use unit products for occasional trips. Clusters 1, 3, 4 and 9, that have mostly scholar profiles also have a large majority of annual subscribers. A possible interpretation is that schoolchildren and students are public transportation captives, and have to use the network in order to go to class every day. Thus, buying an annual pass is more advantageous than buying any other product type.

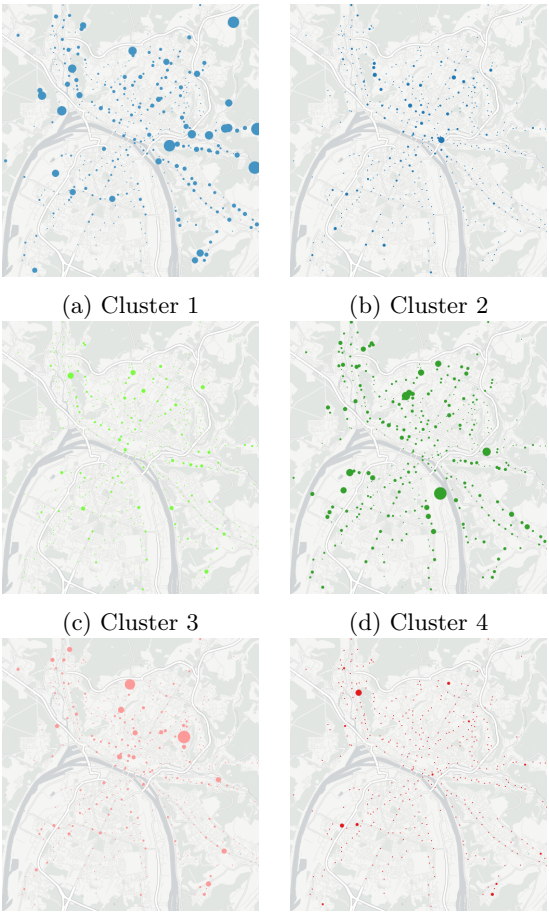
As described in Subsection 4.4.1, we kept only users whose first trip of the day is made at the same station at least 50% of the study time. That main "morning station" is thus called the "home station" as it gives us an estimation of the residence place of users. In Tables 4.4 and 4.5, we can observe the shares of clusters by home stations. It shows the share of travelers identified as belonging to every cluster leaving near each station.

We note that:

1. Cluster 1: travelers are over represented at peripheral stations.
2. Cluster 2: no particular pattern observed.
3. Cluster 3: no particular pattern observed.
4. Cluster 4: few stations show over representation of cluster 4.
5. Cluster 5: over representation of the cluster at two stations in the north.
6. Cluster 6: no particular pattern observed.
7. Cluster 7: One station is 100% represented by cluster 7. As only one user is assigned to this station, no particular pattern is observed.
8. Cluster 8: the cluster is over represented at one station in the city center and at another further.
9. Cluster 9: cluster 9 is over represented in few stations in the center.
10. Cluster 10: cluster is over represented in poorest neighborhoods of the city.

4.4.3 Stations profile clustering

Clustering the different stations of the network would allow us to better know the different type of stations, and to group them by temporal similarity. As we have very few number of stations (475), it is not safe to



(a) Cluster 1 (b) Cluster 2

(c) Cluster 3 (d) Cluster 4

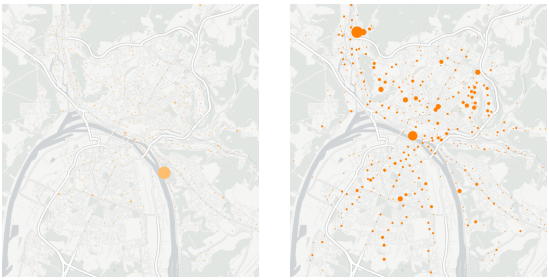
(e) Cluster 5 (f) Cluster 6

Table 4.4: Share of clusters per home station — Clusters 1 to 6

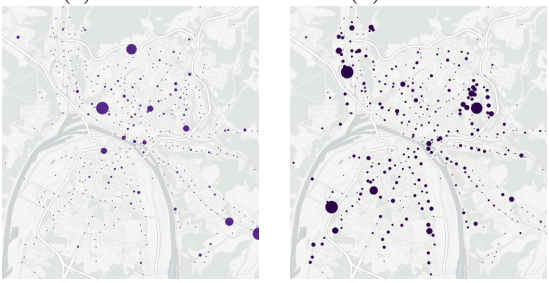
process as described above for the users clustering. Indeed, a K larger than 6 or 7 leads to very small clusters. In place we fixed H and K *a priori* to 3 and 5 respectively.

The 3 words obtained are the ones in Figure 4.9. The first time component is described by check-ins at 7 and 8 a.m. We will call it the "morning component". The second time component shows check-ins at 4 and 5 p.m

Figure 4.9: Words obtained by NMF-EM on stations data with $K = 5$ and $H = 3$.



(a) Cluster 7 (b) Cluster 8



(c) Cluster 9 (d) Cluster 10

Table 4.5: Share of clusters per home station — Clusters 7 to 10

on Mondays, Tuesdays, Thursdays and Fridays and check-ins at 12 p.m on Wednesdays. We will name it the "end of school component". The third component shows check-ins at 6 p.m, during Wednesdays afternoons, during Saturdays and off-peaks periods. This component will be called the "off-peak component".

Figure 4.10 shows the 5 clusters. Stations in cluster 1 are stations where there are check-ins only in the morning at 7 or 8 a.m. These stations are likely in residential areas. In cluster 2, the stations have check-ins all day long, but with highest probabilities during peaks. Stations in cluster 3 have check-ins in the morning and at the end of school. They are likely to be near schools in residential areas. Stations in cluster 4 have check-ins only at end of school times. Thus, these stations are probably near schools. Finally, stations in cluster 5 are pretty similar than the ones in cluster 1: a large majority of check-ins are made in the morning (7 or 8 p.m). The only difference is that it is more likely to have check-ins during the rest of the day in cluster 5 than in cluster 1.

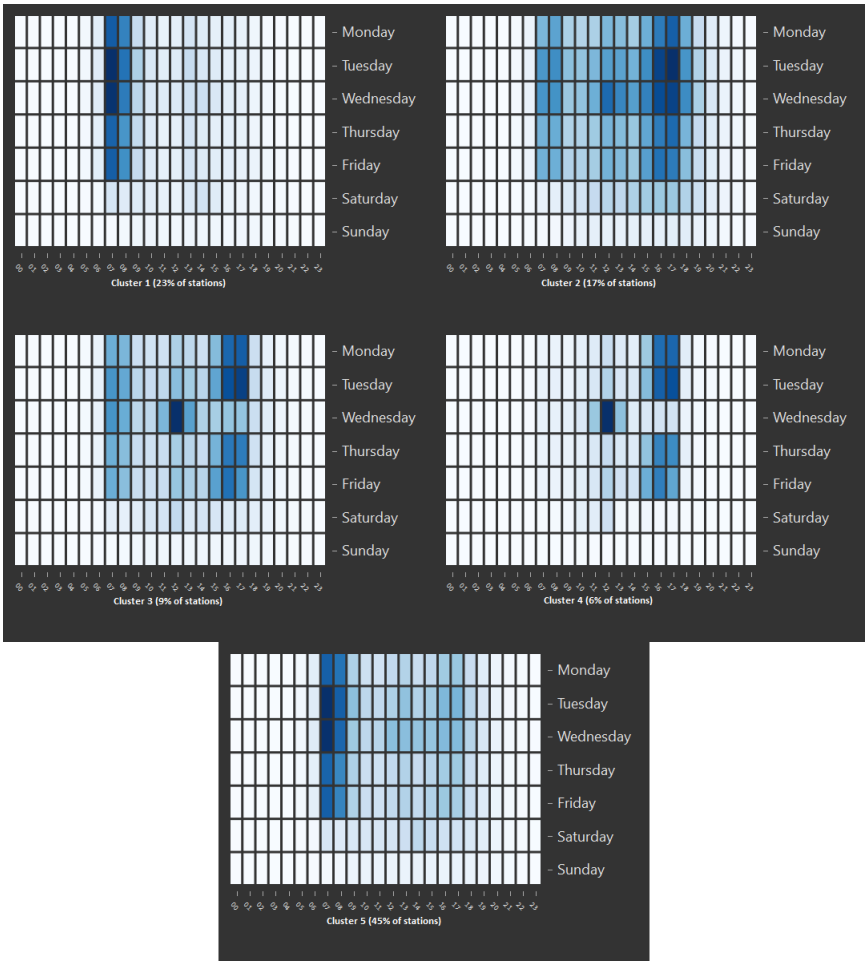


Figure 4.10: Clusters obtained by NMF-EM on stations data with $K = 5$ and $H = 3$.

Thanks to the French National Institute of Statistics and Economics Studies (INSEE), there are open data permitting us to introduce contextual information. Firstly, a database containing socioeconomic data on a grid of $200m \times 200m$ is available. We used two indicator of it: the number of inhabitants and the percentage of households living in collective housing per tiles. Secondly, we used a database referencing and geolocating every french company or administration. In this way, we were able to know the number of employees per tile. By clustering the tiles in the study area, we obtained different group of areas that will allow us to lead the study on stations more finely. Table 4.6 contains the description of the mean tile by cluster.

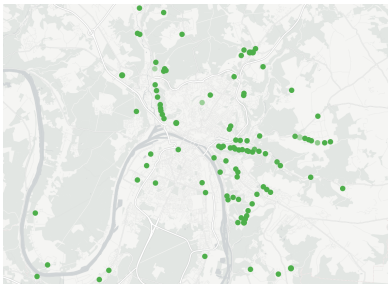
Table 4.6: Description of tiles clusters

Tiles cluster	Inhabitants	Percentage of collective housing	Employees
1	223.75	57.73	824.43
2	162.03	30.08	40.32
3	268.74	58.66	2758.97
4	114.50	98.98	11576.50

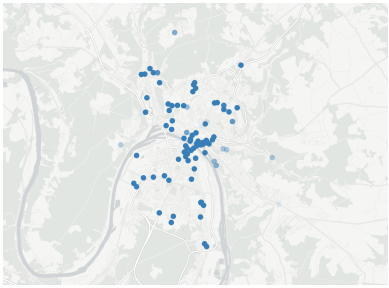
As tiles contained in cluster 1 and 2, are those with the least number of employees, they can be described as residential areas. Moreover, the percentage of collective housing allows to distinguish them. Indeed, cluster 1 have more households living in collective housing than cluster 3. That is why we will refer as tiles from cluster 1 as residential areas in collective housing and as residential areas in individual housing for tiles from cluster 2. Since the number of inhabitants and of employees are high, tiles from cluster 3 will be referred as mixed areas. Finally, as the number of employees in cluster 4 is very large, we will refer these tiles as business areas.

The figures in Table 4.7 show the geographical repartition of the five clusters. In Figure 4.7a, we observe the stations contained in cluster 1. This cluster groups stations that have check-ins only in the morning. On the figure, we observe that these stations are distant from the city center and are mainly located in residential areas. Figure 4.7b shows stations of cluster 2, that have check-ins all day long with stronger attendance during peak-periods. These stations are mainly located in the city center. Figures 4.7c and 4.7d look alike. Indeed, clusters 3 and 4 have the "end of school" component and the points on the map are close to educational

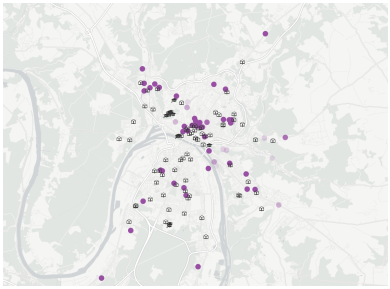
establishment. Figure 4.7e shows stations from cluster 5. These stations have check-ins all day long but most are made in the morning. By looking at the map, we cannot notice any significant pattern.



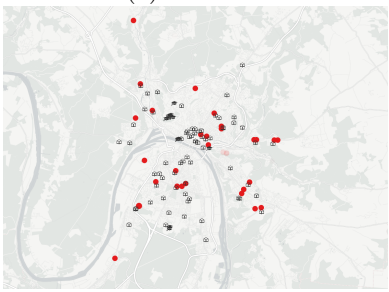
(a) Cluster 1



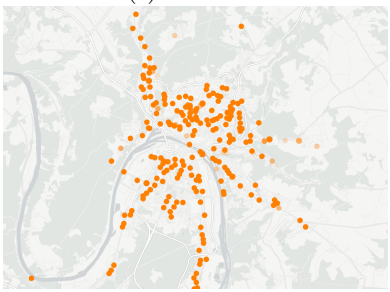
(b) Cluster 2



(c) Cluster 3



(d) Cluster 4



(e) Cluster 5

Table 4.7: Map of the stations — opacity of the points are proportional to the adequacy between the stations and the clusters.

4.4.4 Passengers profile clustering on another network

To ensure efficiency of the algorithm, we applied it on another network located in the Netherlands. By applying the same model selection method as in Section 4.4.2, we obtained the optimal values of $K = 10$ and $H = 7$. Figures 4.11 and 4.12 contain respectively the profiles of the words and clusters obtained.

The interpretation of the words is:

1. Word 1: travels at 6 or 7 a.m and slightly around 4 p.m during the week.
2. Word 2: travels during the week-end.
3. Word 3: diffuse travel habits from 8 a.m to 4 p.m Mondays to Fridays.
4. Word 4: travels at 7a.m on weekdays.
5. Word 5: diffuse habits with highest probabilities from 5 p.m to 12 a.m during the week.
6. Word 6: diffuse habits from 9 a.m to 5 p.m with highest probability at 1 p.m Mondays to Saturdays.
7. Word 7: travels at 8 a.m and 5 p.m.

We can interpret the cluster as follows:

1. Cluster 1: diffuse habits from 9 a.m to 5 p.m with highest probability at 1 p.m Mondays to Saturdays.
2. Cluster 2: travels at 6 or 7 a.m and at 4 or 5 p.m during the week.
3. Cluster 3: diffuse habits from 7 a.m to 6 p.m on weekdays.
4. Cluster 4: diffuse travel habits from 9 a.m to 11 p.m.
5. Cluster 5: travels at 7 or 8 a.m diffuse habits during the afternoon.
6. Cluster 6: travels at 8 a.m and 5 p.m.
7. Cluster 7: diffuse travel habits from 7 a.m to 5 p.m Mondays to Fridays.
8. Cluster 8: diffuse habits from 8 a.m to 4 p.m during the week.
9. Cluster 9: travels during the week-end.
10. Cluster 10: travels at 7 or 8 a.m and around 4 p.m.

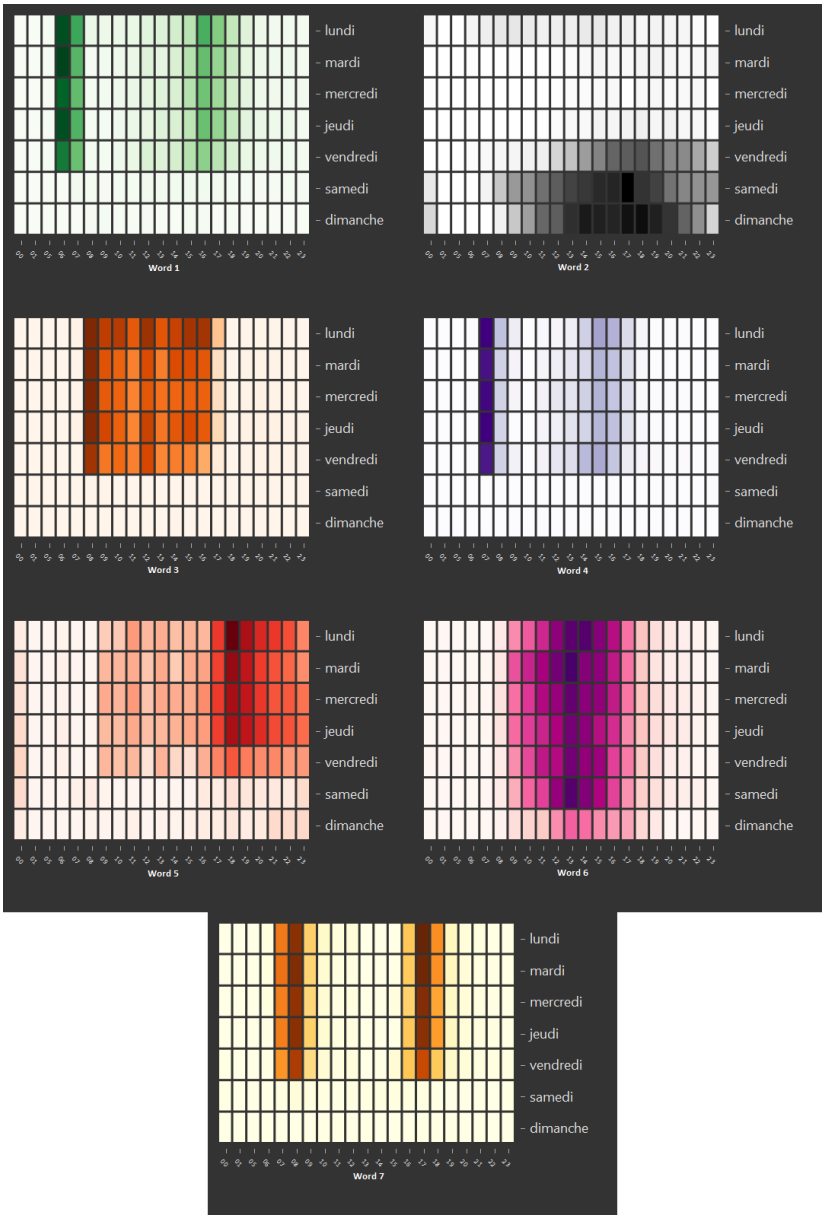


Figure 4.11: Words obtained by NMF-EM on users data with $K = 10$ and $H = 7$.

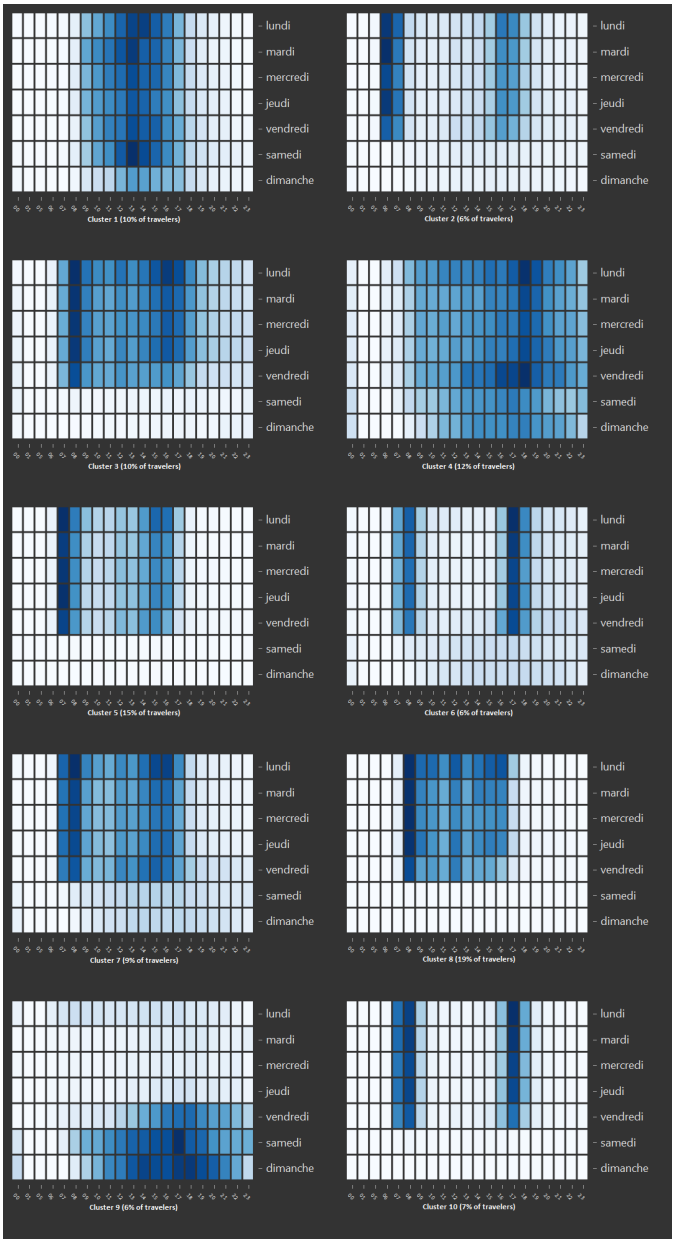


Figure 4.12: Clusters obtained by NMF-EM on users data with $K = 10$ and $H = 7$.

4.5 Conclusion

We provided a new approach for dimension reduction that can be compared to MFA in non-Gaussian mixture models. This approach is based on NMF, an extremely popular data mining algorithm. We adapted the EM algorithm to this setting. This new algorithm, NMF-EM, is implemented in a package for R in the case of mixtures of multinomials. Results on simulated and real data are promising. In addition to a theoretical study of algorithm, future work should include an application to mixture of other distribution with nonnegative parameters like the Poisson distribution.

Chapter 5

Forecasting the public transportation attendance and its application to anomaly detection

5.1 Introduction

In order to improve the quality of an urban transportation network, it is very important to be able to anticipate the users' demand. As population travels are complex phenomena dependent on a large number of variables, it is important to know what are the extreme events. Giving a range of normal level of check-ins also allows to detect occurrences of too low or too high levels of attendance, in order for the operating teams of the network to better understand why there are these anomalies. In this chapter, based on and improving our previous work [Carel and Alquier, 2018], we propose to compare several Machine Learning algorithms to forecast the number of passengers at a tramway stop hour by hour and then to process to another comparison of algorithms to create a confidence interval around the forecast.

5.2 Data presentation

We have at our disposal two years of ticketing data from the Astuce network of the Rouen-Normandie metropolis (January 1st, 2014 to December 31st, 2015). To be able to forecast the number of check-ins by station, we aggregated the data by day, hour, station and line. A commercial station is divided in several physical stations, depending of the direction of the line(s) stopping by. On the network, there are 2308 commercial stations for 615 physical stations. For the sake of clarity, we inform the reader that we refer to physical stations when talking about stations.

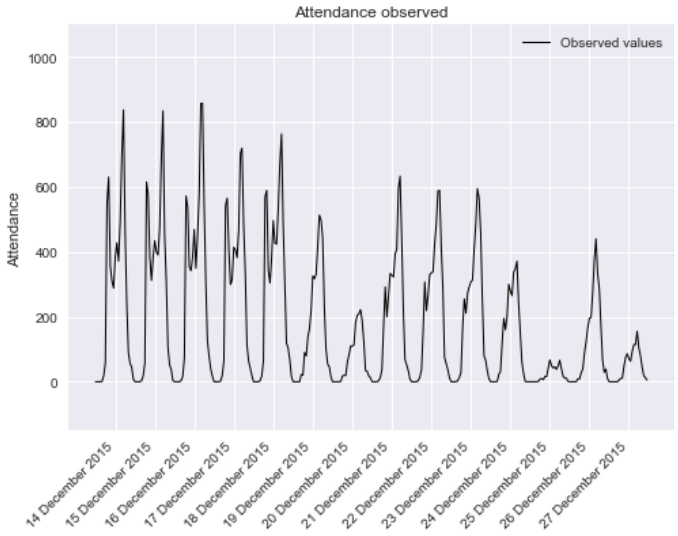


Figure 5.1: Observed values of attendance at the "Théâtre des Arts" tramway station in the north to south direction. The week of 14 to 20 December is a scholar one, whereas the one of 21 to 27 is a vacation week. Attendance is lower during vacations and weak on the 25th December, which is a holiday. The observations of the 21^{st} and 24th of December are contained in the test dataset.

As we have a dataset of 8 198 346 observations, we will modelize the check-ins of only one station. This station is the one generating the most traffic: the "Théâtre des Arts" tramway station in the north to south direction. On Figure 5.1, two weeks of observation of this station are rep-

resented. To take account of the several seasonalities in the data, we create the hour, weekday, month, period (scholar, summer vacations, other vacations), holidays and common days-off features as calendar features. We also introduced meteorological[1] features of average temperature, average wind speed, average humidity and rainfall of the day.

We separated our data in two datasets: the train dataset – containing 80% of the dates – will allow us to train the models, whereas the test dataset – containing 20% of the dates randomly selected – will permit us to compare the performance of each trained model on a a new dataset.

5.3 Modelization

Let Y be the attendance at a certain station during the study period. Then we call Y_t the attendance at the station at the time t. The different features at the same moment are contained in the vector X_t. Given X_t, the aim of forecasting is to find an estimated value $\hat{Y}_t$ of Y_t that is as accurate as possible, thanks to a function $f(\cdot)$. Mathematically, we have:

$$Y_t = f(X_t) + \varepsilon_t,$$

ε_t being the difference between the observed value Y_t and the predicted value $\hat{Y}_t = f(X_t)$. The smaller ε_t, the better is the forecast at the time t. As we want to find a model forecasting Y as precisely as possible, we tested several models. We choose to estimate $f(\cdot)$ with Linear Models (LM), Generalized Additive Model (GAM) and Random Forest (RF). These three algorithms are presented respectively in Subsections 5.3.1, 5.3.2 and 5.3.3.

5.3.1 Linear model

In statistics, the simplest model for forecasting is called the linear model. It is based on the assumption that there are some linear links between the explanatory variables $(X_1, \ldots, X_p) \in \mathbb{R}^{n \times p}$ and the target variable $Y \in \mathbb{R}^n$, such that for an observation $i \in \{1, \ldots, n\}$ we have:

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j X_{i,j} + \varepsilon_i \tag{5.1}$$

where $\beta = (\beta_0, \beta_1, \ldots, \beta_p) \in \mathbb{R}^{p+1}$ is the vector containing the regression coefficients, and ε_i is the regression error term. The most popular way

[1] SYNOP database from Météo France

to perform a linear regression and estimate the β vector is to use the Ordinary Least Squares (OLS) method, that gives $\hat{\beta} = (X^t X)^{-1} X^t Y$. We used the OLS implemented in the Python library `sklearn` with the `LinearRegression` function.

5.3.2 Generalized Additive Model

The linear model, as described above, is based on the assumption of a linear link between the forecast features and the target variable. The link between some variables, as temperature and the number of passengers at a tramway station, is unlikely to be linear. The Generalized Additive Model (GAM), introduced by [Hastie and Tibshirani, 1986], proposes:

$$Y_i = \beta_0 + \sum_{j=1}^{p} s_j(X_{i,j}) + \varepsilon_i \tag{5.2}$$

in order to estimate functions $s_j(\cdot)$ instead of regression coefficients β_j in (5.1). Theoretical analyses of GAM models has been produced by [Guedj and Alquier, 2013; Abramovich and Lahav, 2015] among others. There are different methods to estimate these s_j functions, we chose to use the cubic splines implemented in the function `gam` from the `R` package `mgcv`.

5.3.3 Random Forest

To better understand the Random Forest (RF) algorithm, it is essential to know the principle of decision trees. At the root of the tree, every observation is represented. Then, at the first node, the variable allowing to discriminate the most the observations on their target variable is selected and the observations are splitted into two nodes. The process is reiterated on the new nodes until every observation at the node have the same target value or splitting does not improve the forecast anymore. Final nodes are called leaves. Figure 5.2 gives an example decision tree.

The principle of the RF algorithm, introduced in [Breiman, 2001], is based on decisions trees. A random forest is composed of several decision trees, where each tree is partially independent from every other tree. Indeed, every tree is trained only from a sample of observations and a sample of variables. The forecast value of an observation is thus an aggregated value of all the values obtained by each tree:

$$Y_i = \frac{1}{M} \sum_{m=1}^{M} a_m(X_{i,1}, \ldots, X_{i,p}) + \varepsilon_i, \tag{5.3}$$

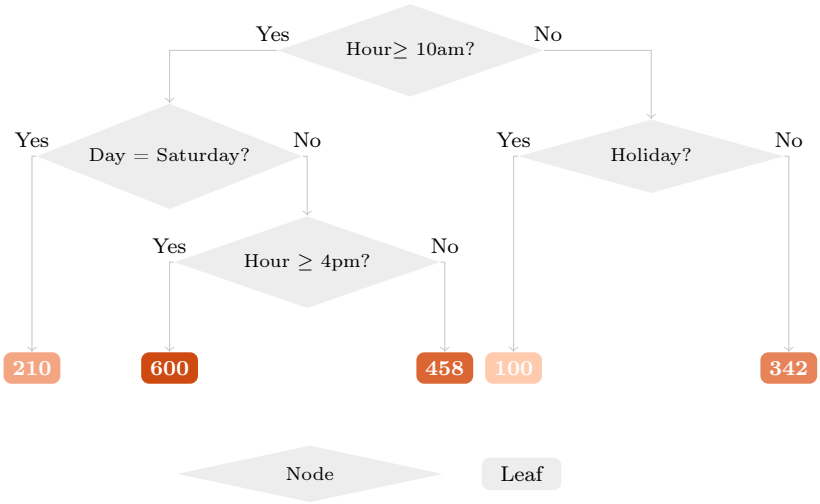


Figure 5.2: Decision tree representing the number of hourly check-ins on a transportation network. The leaves give the average value of each population.

with $a_m(\cdot)$ being the m^{th} tree in the forest. Random Forest methods are among the very best in term of forecasting and are widely studied [Biau and Scornet, 2016; Genuer et al., 2008], but were theoretically misunderstood for a long time. First theoretical studies were leaded by [Genuer, 2012; Arlot and Genuer, 2014] and an important theoretical breakthrough happened more recently [Scornet et al., 2015]. We used the RF algorithm implemented in the Python library `sklearn` with the `RandomForestRegressor` function.

In the OLS algorithm, forecasting Y_t from the variables contained in X_t assumes that each feature has an additive effect, whereas it can also have a multiplicative one. This is why, we wanted to process a regression on $\log(Y_t + 1)$ too:

$$\begin{aligned} \log(Y_t + 1) &= f(X_t) + \varepsilon_t \\ Y_t + 1 &= e^{f(X_t)} e^{\varepsilon_t} \end{aligned}$$

where $f(\cdot)$ is estimated either by (5.1), (5.2) or (5.3).

Finally, in order to be able to compare the performances of the different models, we used the Root Mean Squared Error, which is:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\varepsilon_t^2}.$$

Table 5.1: Root Mean Squared Error (RMSE) of the three models on the test set by the two link functions used.

	Algorithm		
Link function	OLS	GAM	RF
Y_t	95	**46**	81
$\log(Y_t + 1)$	85	52	125

In Table 5.1, we note that the criterion is minimized when we use the Generalized Additive Model algorithm with an identity link function. Thus, the final model is:

$$Y_t = f(X_t) + \varepsilon_t, \tag{5.4}$$

where $f(\cdot)$ is a function estimated by GAM. This is the model we will use in the following parts of this chapter. We observe on Figure 5.3 the forecasting on the two same weeks as on Figure 5.1.

5.4 Confidence intervals

In order to adapt the transportation offer to the demand, organizing authorities are too often often relying on punctual forecasts. It is more useful to be able to guard against the worst of congestion at every moments. Statistically, it is thus essential to enrich the forecast with a confidence interval. However, it seems natural that the forecast uncertainty depends on the several features. Indeed, during peak periods, the uncertainty should depend on the traffic conditions and will be bigger than during off-peak periods. There exist several methods to add confidence interval to the forecast. In a non-parametric way, quantile losses are often minimized in economics [Cornec, 2014] and machine learning [Alquier and Li, 2012] and new methods are proposed [Deswarte, 2018]. In this work we are focusing

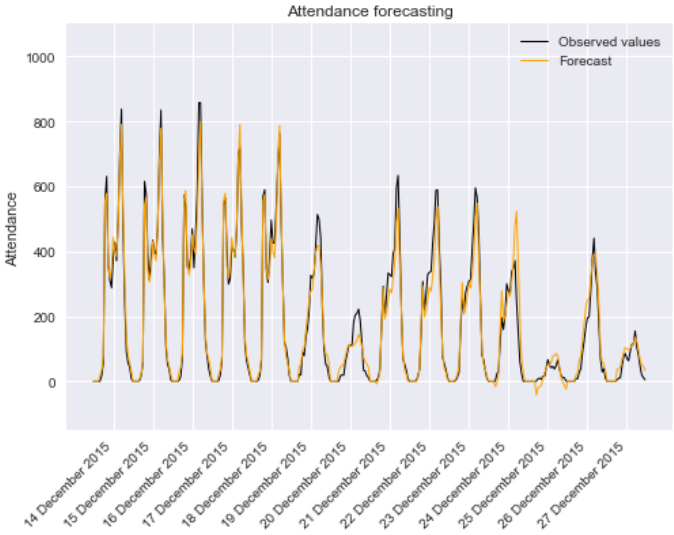


Figure 5.3: Forecast attendance at the "Théâtre des Arts" tramway station from 14 to 27 December 2015.

on parametric confidence interval under the assumption of gaussian noise by estimating the variance in an heteroskedastic model.

From (5.4) and for a fixed X_t, we can tell that the variance of the estimation is the variance σ_t^2 of the residual ε_t. Moreover, as $\varepsilon_t \sim \mathcal{N}(0, \sigma_t^2)$, we have $\mathbb{E}[\sigma_t^2] = \mathbb{E}[\varepsilon_t^2]$. Thus, to estimate the variance for each t, we introduce the simple model:

$$\mathbb{E}[\varepsilon_t^2] = h(X_t).$$

We can also make the assumption that the error variance depends on the previous error. Thus we introduce the auto-regressive model:

$$\mathbb{E}[\varepsilon_t^2] = l(X_t, \varepsilon_{t-1}^2).$$

As for the check-ins forecast, we choose to estimate the functions $h(\cdot)$ and $l(\cdot)$ by OLS, GAM and RF, and we tested each algorithm with an identity and a logarithmic link function. We will compare the performances of the different modelization combinations thanks to the RMSE criterion, as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(\varepsilon_t^2 - \hat{\varepsilon}_t^2)^2}.$$

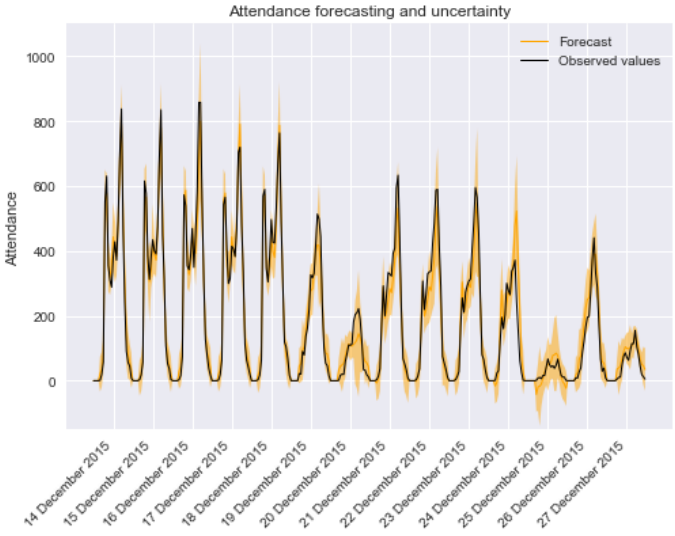
On Table 5.2, we observe that the modelization combination allowing to minimize the RMSE criterion is an auto-regressive Random Forest algorithm using an identity link function. We note for each algorithm, the auto-regressive/identity combination always offer a more precise forecast. We will use the RF/auto-regressive/identity in the following parts of this chapter. But let note that both the simple and the auto-regressive methods in this combination have great performance in comparison to the other combinations. The main difference is that with the simple method, we can forecast the number of check-ins at twenty-four hours, whereas with the auto-regressive method, we can only forecast at one hour as we need the observation and forecast of the last hour in the auto-regressive method. However, it is possible to use an estimation of the previous observation at $t-1$ to forecast the number of check-ins at time t. As we estimated the variance of the check-ins forecast at each moment t, we can now construct the 95% confidence interval as follows:

$$Y_t \in \left[\hat{Y}_t - 1.96 \times \hat{\varepsilon}_t, \hat{Y}_t + 1.96 \times \hat{\varepsilon}_t\right]. \tag{5.5}$$

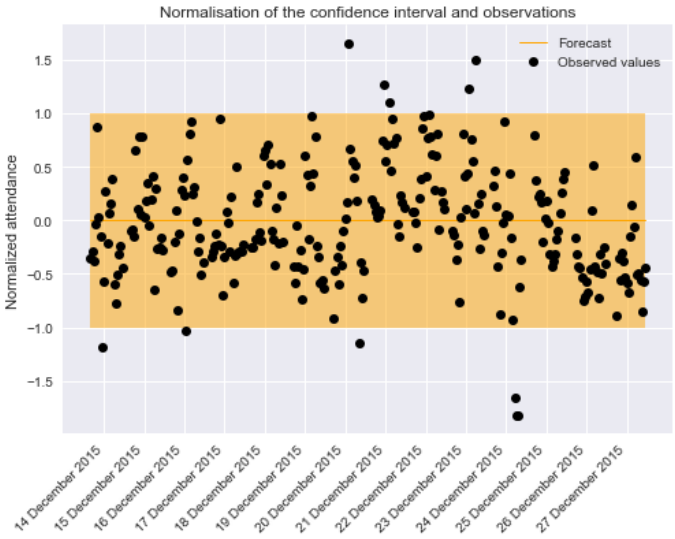
Table 5.2: Root Mean Square Error and coverage rate of the corresponding confidence interval for each variance model on the test set

Algorithm	Method	Target	RMSE	Coverage rate
Constant		ε^2	5165	95.3%
Ordinary Least Squares	simple	ε^2	5008	75.5%
		$\texttt{log}(\varepsilon^2 + 1)$	5185	66.2%
	auto-regressive	ε^2	**4977**	77.6%
		$\texttt{log}(\varepsilon^2 + 1)$	5239	64.5%
Generalized Additive Model	simple	ε^2	5199	69.3%
		$\texttt{log}(\varepsilon^2 + 1)$	4931	66.4%
	auto-regressive	ε^2	**4686**	72.6%
		$\texttt{log}(\varepsilon^2 + 1)$	5132	59.4%
Random Forest	simple	ε^2	4894	96.8%
		$\texttt{log}(\varepsilon^2 + 1)$	5300	67.2%
	auto-regressive	ε^2	**4618**	95.7%
		$\texttt{log}(\varepsilon^2 + 1)$	5182	51.5%

On the test dataset, our confidence interval has a 95.7% coverage rate. As this rate is close to the one expected (95%), we keep this interval. We observe on Figure 5.4 the interval from (5.5) on our two weeks period. As the confidence interval is pretty tight on most of the observed days, we also drawed a normalized confidence interval, in order to better detect the observations outside the interval. We notice that several observations are over the interval from 20 to 23 December. A possible explanation is that, being the last days before Christmas, the commercial activity increased in the area of the station. Similarly, we notice that three observations are clearly under the interval in the afternoon of the 24^{th} of December. As it is on Christmas Eve, citizens travel habits are likely different than on every other day.



(a) Attendance forecasting and uncertainty.



(b) Normalized confidence interval and attendance.

Figure 5.4: Forecast attendance at the "Théâtre des Arts" tramway station from 14 to 27 December 2015. Most of the observations are contained in the confidence interval.

5.5 Application: impact of the 2018 SNCF social strike on one network

5.5.1 Introduction

From April to the beginning of July 2018, the french national railway company (SNCF) have known a big social strike. Two days every five days were impacted by the movement. The region around Paris (Île-de-France) have been impacted, as most express regional rail lines (RER and Transilien lines) are operated by SNCF. Some local public transportation networks are connected to the regional railway network. Our intuition is that these local networks have been impacted by the strike in term of check-ins. Indeed, in the region, transport operators are paid in function of the number of check-ins made on the networks. In other words, a decrease number of passengers on the rail network could lead to a decrease in check-ins on the connected local networks lines and thus a decrease of revenue. On the contrary, some other lines connected to subway or tramway lines (and not impacted by the SNCF strike) could have known a positive impact in term of check-ins and revenue.

In this section, our study focus on another Transdev network: TRA, which operates through Seine-Saint-Denis department. The network is composed of 25 bus lines, covering territories of 24 municipalities. Our data are decomposed in two datasets giving the number of check-ins made by day and line. In one hand, we have a historic dataset of one year of data before the strike from April 1st, 2017 to March 31st, 2018. In the other hand we have an application dataset, covering the strike period from April 1st, 2018 to July 8th, 2018. We used as a feature the number of commercial kilometers driven by line and day. To enrich these datasets, we created calendar features, such as weekdays, periods of the year (scholar, summer vacations or other vacations), indicators on holidays, on common extra days off, on strike days and on inter-strike days – i.e. non-strike days between two strike days. We also used meteorological data[2] as mean temperature, mean wind speed, mean humidity and rainfalls of the day. As the historic is shorter than on the first part of this chapter and we have daily and not hourly data, it is important to note that this part is processed on fewer observations.

Our aim is to be able to quantify the impact of the strike on the network by line in term of check-ins. Our methodology took place in two stages: firstly, we computed on the historic data one model of forecast and variance

[2]SYNOP database from Météo France

by line as explained above. We then applied these models to the application dataset. We bring to the attention of the reader that these models don't use the strike and inter-strike indicators as features. To be able to prove the impact of the strike on the different lines we compared by line the coverage rates of the computed confidence interval on the historic period, the strike days and the non-strike days. Secondly, to corroborate these first results we incorporated the strike and inter-strike indicators into econometrics models, which regression coefficients allowed us to quantify the impact of the strike by line.

5.5.2 Model selection

We applied the same methodology as above on our new data. The performance of the different forecast algorithms on the test set by line are contained in Table 5.3. An aggregate version is contained in Table 5.4. On both Tables, we observe that the combination offering the best performance by line and aggregatively is a Least Squares algorithm with a logarithmic link function. Thus, this is the combination we choose for the further stages of the study. On Table 5.5, we note that the model with the best RMSE criterion to forecast the variability is the constant one. This combination offers also one of the best coverage rate. The corresponding confidence interval is represented in Figure 5.5 for three months before the strike period.

For both the forecast and the uncertainty, we note that non-parametric models are disadvantaged. Indeed, in this application there are far fewer observations than on the first part of this chapter and this type of model is more efficient for big datasets.

5.5.3 Results

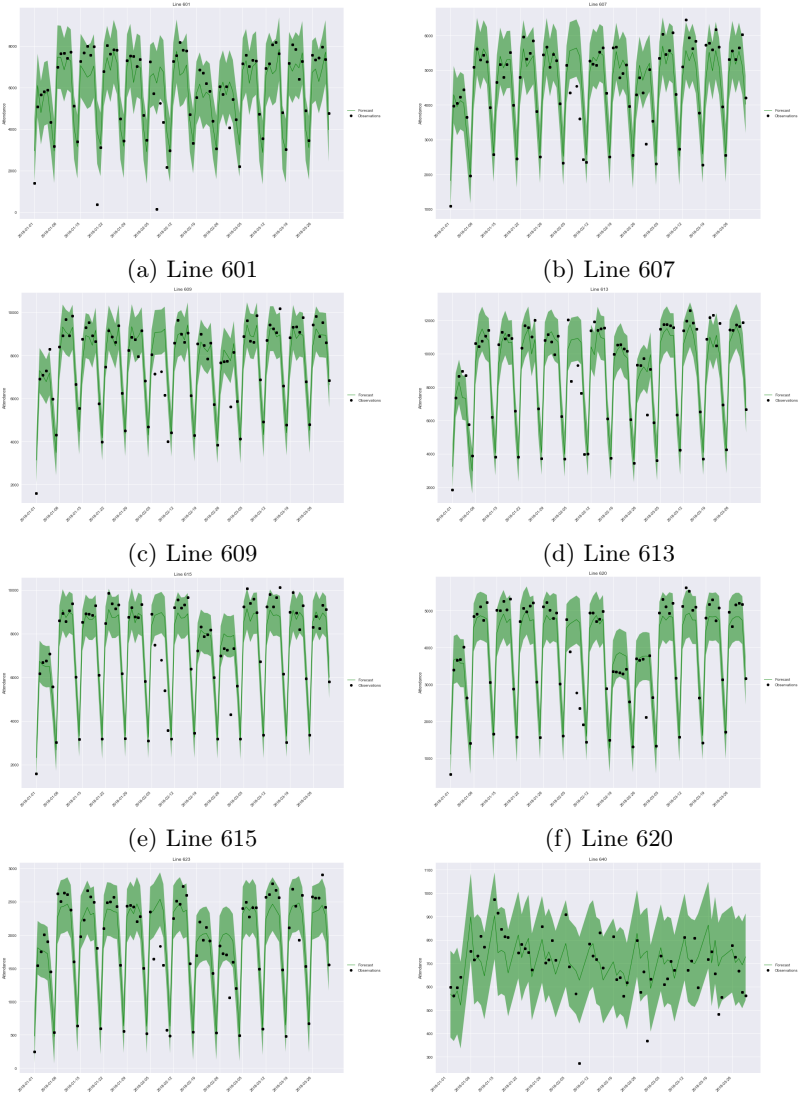
On Table 5.6, the observation repartition by line and period are recorded. We notice that while numerous lines have mixed effects from strike or inter-strike days, one line (609) have known a rise of validations during strike days and five a decrease (601, 615, 620, 623 and 640). During inter-strike days, lines 607, 609 and 613 have clear positive impact, whereas lines 601 and 623 have clear negative ones. The observations, forecasts and confidence intervals of these eight lines are represented in Figure 5.6.

Table 5.3: Root Mean Squared Error (RMSE) by line of the three models on the test set by the two link functions used.

	Algorithm and link function					
Line	OLS	logOLS	GAM	logGAM	RF	logRF
601	589	637	805	1002	**516**	838
602	**409**	463	683	725	434	459
603	203	**200**	279	279	203	228
604	225	186	248	258	**185**	205
605	218	212	209	244	**190**	210
607	534	532	638	643	**477**	492
609	**570**	571	902	915	885	1373
610	**194**	230	201	205	535	595
613	1000	1045	1356	1345	887	**822**
615	663	632	1050	1067	623	**604**
616	387	**371**	572	576	380	401
617	269	248	318	321	258	**246**
618	291	289	404	407	**285**	301
619	167	151	240	240	154	**134**
620	392	352	570	577	400	**330**
623	263	**224**	329	337	225	244
627	93	**89**	108	110	94	97
637	89	**88**	106	110	100	102
640	266	**114**	184	181	126	133
642	391	**317**	495	495	366	268

Table 5.4: Root Mean Squared Error (RMSE) of the three models on the test set by the two link functions used.

	Algorithm		
Link function	OLS	GAM	RF
Y_t	395	582	406
$\log(Y_t + 1)$	**394**	602	479



(a) Line 601

(b) Line 607

(c) Line 609

(d) Line 613

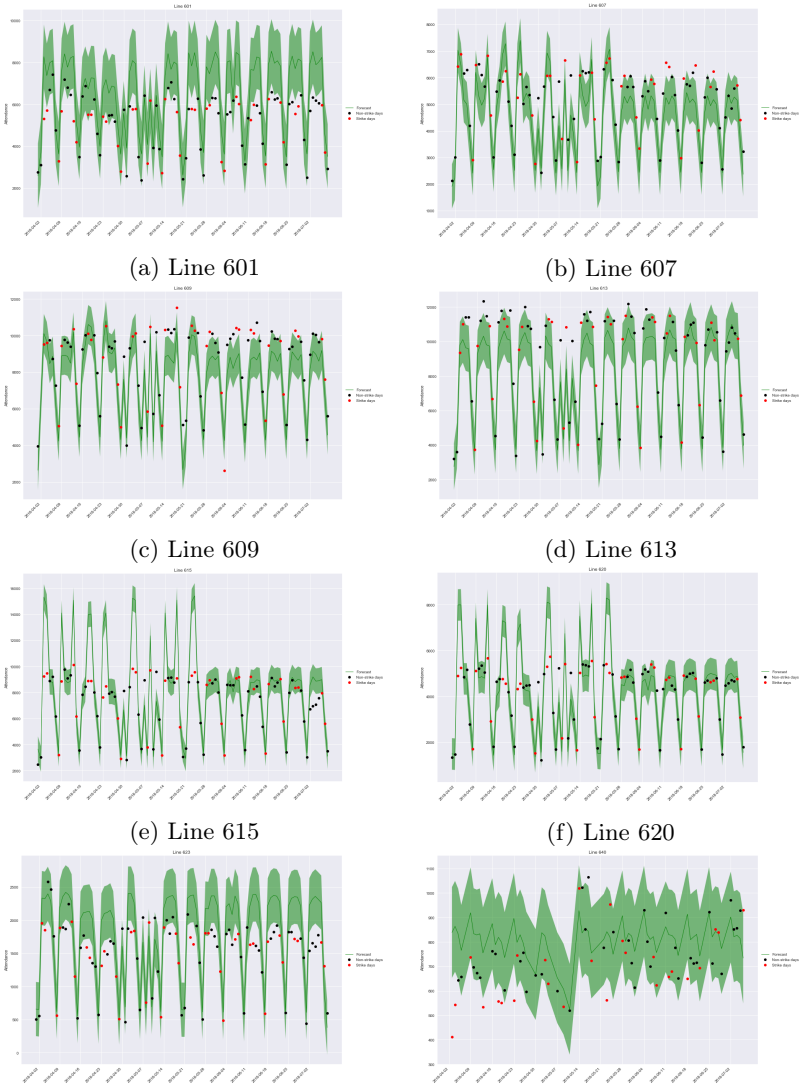
(e) Line 615

(f) Line 620

(g) Line 623

(h) Line 640

Figure 5.5: Observations, forecast and confidence of eight lines before the strike period of 2018.



(a) Line 601

(b) Line 607

(c) Line 609

(d) Line 613

(e) Line 615

(f) Line 620

(g) Line 623

(h) Line 640

Figure 5.6: Observations, forecast and confidence interval during Spring 2018 of eight lines. Observations during strike days are represented in red.

Table 5.5: Root Mean Square Error and coverage rate of the corresponding confidence interval for each variance model on the test set

Algorithm	Method	Target	RMSE	Coverage rate
Constant		ε^2	**5.7e+05**	**92.8%**
Ordinary Least Squares	simple	ε^2	5.9e+05	80.2%
		$\texttt{log}(\varepsilon^2 + 1)$	6.0e+05	65.4%
	auto-regressive	ε^2	5.9e+05	78.5%
		$\texttt{log}(\varepsilon^2 + 1)$	6.0e+05	64.6%
Generalized Additive Model	simple	ε^2	7.1e+05	70.8%
		$\texttt{log}(\varepsilon^2 + 1)$	4.0e+07	66.3%
	auto-regressive	ε^2	7.9e+05	75.3%
		$\texttt{log}(\varepsilon^2 + 1)$	1.8e+07	66.1%
Random Forest	simple	ε^2	1.0e+06	92.5%
		$\texttt{log}(\varepsilon^2 + 1)$	6.0e+05	64.4%
	auto-regressive	ε^2	9.2e+05	93.4%
		$\texttt{log}(\varepsilon^2 + 1)$	5.9e+05	63.9%

Lines 607, 609, 615 and 620 have really high forecast in weekdays for strike days during April to end of May. Indeed, bus services have been strengthened in order to allow a better mobility for passengers during the strike. This explains the high forecast for these days. However, we observe that for lines 615 and 620, the expected growth of validations have not been achieved. Table 5.7 contains the regression coefficients of the econometric model and their significance. Globally, the biggest impact we noticed on the previous Table are confirmed by the regression model. The only exception is for line 620. Indeed, with the intervals we noted a clear negative impact, while the regression coefficient is negative but non-significant. For the other lines, the coefficients correspond mostly to what have been found with the intervals. These Table also contains the global impact of the strike period on the validations of every line. We note that the total impact is a loss about 88000 validations (or 917 validations/day), which is small for a network as big as TRA. Thus, through this study, we highlighted that during the SNCF social strike, the impact on the validations is very modest. Nevertheless, we showed that during that period, passengers flows were different from before the strike period. Indeed, passengers

Table 5.6: Repartition of the observations by period and line. Clear positive impacts are represented in green while clear negative impacts are represented in red.

	Training period			Strike period			Inter-strike period		
Line	Higher	In	Lower	Higher	In	Lower	Higher	In	Lower
601	1%	96%	4%	0%	45%	**55%**	0%	62%	**38%**
602	1%	95%	4%	3%	97%	0%	5%	92%	3%
603	2%	95%	3%	0%	89%	11%	7%	91%	2%
604	2%	94%	3%	0%	97%	3%	7%	93%	0%
605	2%	96%	2%	3%	94%	3%	9%	85%	6%
607	2%	94%	4%	21%	66%	13%	**28%**	72%	0%
609	2%	96%	2%	**34%**	63%	3%	**40%**	60%	0%
610	2%	95%	3%	3%	97%	0%	5%	95%	0%
613	5%	91%	4%	16%	84%	0%	**30%**	70%	0%
615	1%	97%	2%	0%	58%	**42%**	5%	85%	10%
616	1%	97%	2%	3%	81%	16%	3%	89%	8%
617	1%	95%	4%	0%	92%	8%	0%	92%	8%
618	3%	95%	2%	5%	87%	8%	10%	88%	2%
619	1%	94%	5%	3%	87%	10%	8%	84%	8%
620	1%	97%	2%	0%	61%	**39%**	10%	90%	0%
623	1%	95%	4%	0%	50%	**50%**	0%	61%	**39%**
627	2%	96%	2%	0%	97%	3%	4%	96%	0%
637	1%	95%	4%	0%	86%	14%	3%	92%	5%
640	3%	95%	2%	4%	65%	**31%**	5%	92%	3%
642	1%	95%	4%	0%	84%	16%	0%	92%	8%

preferred bus lines connecting to subway or tramway lines whether than with train lines.

5.6 Conclusion

We illustrated with a short example how modern Machine Learning methods can be used to predict the number of users in a public transport system. In addition to direct application to forecasting, this also makes it possible to assess the impact of exogenous variables on the network, such as failures in another network.

Table 5.7: Regression coefficients of the econometric model and their significance. "***" : p-value $\leq$ 0.001, "**" : 0.001 $\leq$ p-value $\leq$ 0.01, "*" : 0.01 $\leq$ p-value $\leq$ 0.05, "." : 0.05 $\leq$ p-value $\leq$ 0.1, " " : 0.1 $\leq$ p-value.Column $\Delta_{\text{val.}}$ gives the difference between normal and observations during April to beginning of July 2018.

Line	β_S	Sign.	β_{IS}	Sign.	$\Delta_{\text{val.}}$
601	**−1288**	***	**−908**	***	−101608
602	−93		108		
603	−25		**82**	*	4756
604	**−137**	**	**−105**	**	−11296
605	**−143**	***	**−93**	*	−10828
607	**483**	***	**448**	***	44338
609	**408**	**	**708**	***	56568
610	**164**	***	**160**	***	15512
613	87		**300**	*	17400
615	**−599**	***	−170		−22762
616	**−341**	***	**−210**	***	−25138
617	**−179**	***	**−126**	***	−14110
618	−34		−16		
619	**−63**	*	16		−2394
620	−117		**169**	*	9802
623	**−404**	***	**−336**	***	−34840
627	**−72**	***	**−32**	.	−4592
637	**−65**	***	**−38**	**	−4674
640	**−66**	**	3		−2508
642	**−121**	.	−41		−1558
				Total	−87932

Bibliography

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318. ACM, 2016.

Milad Leyli Abadi, Allou Samé, Latifa Oukhellou, Nicolas Cheifetz, Pierre Mandel, Cédric Féliers, and Olivier Chesneau. Predictive classification of water consumption time series using non-homogeneous markov models. In *Data Science and Advanced Analytics (DSAA), 2017 IEEE International Conference on*, pages 323–331. IEEE, 2017.

Felix Abramovich and Tal Lahav. Sparse additive regression on a regular lattice. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(2):443–459, 2015.

Hannele Ahvenniemi, Aapo Huovila, Isabel Pinto-Seppä, and Miimu Airaksinen. What are the differences between sustainable and smart cities? *Cities*, 60:234–245, 2017.

William Alonso. *Location and land use. Toward a general theory of land rent.* Cambridge, Mass.: Harvard Univ. Pr., 1964.

Pierre Alquier and Benjamin Guedj. An oracle inequality for quasi-Bayesian non-negative matrix factorization. *Mathematical Methods of Statistics*, 26(1):55–67, 2017.

Pierre Alquier and Xiaoyin Li. Prediction of quantiles by statistical learning and application to gdp forecasting. In *International Conference on Discovery Science*, pages 22–36. Springer, 2012.

Naomi S Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185, 1992.

Simon P Anderson and André De Palma. Parking in the city. *Papers in Regional Science*, 86(4):621–632, 2007.

Sylvain Arlot and Robin Genuer. Analysis of purely random forests bias. *arXiv preprint arXiv:1407.3939*, 2014.

Sylvain Arlot and Pascal Massart. Data-driven calibration of penalties for least-squares regression. *Journal of Machine Learning Research*, 10 (Feb):245–279, 2009.

Richard Arnott, Andre De Palma, and Robin Lindsey. A structural model of peak-period congestion: A traffic bottleneck with elastic demand. *The American Economic Review*, pages 161–179, 1993.

Farzindar Atefeh and Wael Khreich. A survey of techniques for event detection in twitter. *Computational Intelligence*, 31(1):132–164, 2015.

Michael Batty. Big data, smart cities and city planning. *Dialogues in Human Geography*, 3(3):274–279, 2013.

Jean-Patrick Baudry, Cathy Maugis, and Bertrand Michel. Slope heuristics: overview and implementation. *Statistics and Computing*, 22(2): 455–470, 2012.

Tatiana Benaglia, Didier Chauveau, David Hunter, and Derek Young. mixtools: An R package for analyzing finite mixture models. *Journal of Statistical Software*, 32(6):1–29, 2009.

Jacqueline K Benedetti. On the nonparametric estimation of regression functions. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 248–253, 1977.

Gérard Biau and Erwan Scornet. A random forest guided tour. *Test*, 25 (2):197–227, 2016.

John Bibby, John Kent, and Kanti Mardia. Multivariate analysis. Academic Press, London, 1979.

Christophe Biernacki, Gilles Celeux, and Gérard Govaert. An improvement of the NEC criterion for assessing the number of clusters in a mixture model. *Pattern Recognition Letters*, 20(3):267–272, 1999.

Christopher M Bishop. Pattern recognition and machine learning (information science and statistics), 1st edn. 2006. corr. 2nd printing edn, 2007.

Duncan Black and Vernon Henderson. A theory of urban growth. *Journal of Political Economy*, 107(2), 1999.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

Cécile Bothorel, Neal Lathia, Romain Picot-Clemente, and Anastasios Noulas. Location recommendation with social media data. In *Social Information Access*, pages 624–653. Springer, 2018.

Charles Bouveyron and Camille Brunet-Saumard. Model-based clustering of high-dimensional data: a review. *Computational Statistics and Data Analysis*, 71:52–78, 2014.

Charles Bouveyron, Etienne Côme, and Julien Jacques. The discriminative functional mixture model for a comparative analysis of bike sharing systems. *The Annals of Applied Statistics*, 9(4):1726–1760, 2015.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Anne Sarah Briand, Etienne Come, Martin Trepanier, and Latifa Oukhellou. Smart card clustering to extract typical temporal passenger habits in transit network. two case studies: Rennes in france and gatineau in canada. In *3rd International Workshop and Symposium:" Research and applications on the use of passive data from public transport"*, 2017.

Léna Carel and Pierre Alquier. Non-negative matrix factorization as a pre-processing tool for travelers temporal profiles clustering. In M. Verleysen, editor, *Proceedings of the 25th European Symposium on Artificial Neural Networks*, pages 417–422. i6doc.com, 2017.

Léna Carel and Pierre Alquier. Prévision de la fréquentation d'un réseau de transport à l'aide de modèles additifs généralisés. In *Proceedings of the 50th "Journées des Statistiques"*, 2018.

Pablo Samuel Castro, Daqing Zhang, Chao Chen, Shijian Li, and Gang Pan. From taxi gps traces to social and community dynamics: A survey. *ACM Computing Surveys (CSUR)*, 46(2):17, 2013.

Gilles Celeux, Sylvia Frühwirth-Schnatter, and Christian P. Robert, editors. *Handbook of Mixture Analysis*. CRC Press, 2018a.

Gilles Celeux, Cathy Maugis-Rabusseau, and Mohammed Sedki. Variable selection in model-based clustering and discriminant analysis with a regularization approach. To appear in Advances in Data Analysis and Classification, 2018b.

Badr-Eddine Chérief-Abdellatif and Pierre Alquier. Consistency of variational bayes inference for estimation and model selection in mixtures. *Electronic Journal of Statistics*, 12(2):2995–3035, 2018.

Etienne Côme and Latifa Oukhellou. Model-based count series clustering for bike sharing system usage mining: a case study with the vélib' system of paris. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):39, 2014.

Matthieu Cornec. Constructing a conditional gdp fan chart with an application to french business survey data. *OECD Journal: Journal of Business Cycle Measurement and Analysis*, 2013(2):109–127, 2014.

Riccardo Crescenzi and Andrés Rodriguez-Pose. Infrastructure and regional growth in the european union. *Papers in Regional Science*, 3(91): 487–513, 2012.

Renata Paola Dameri and Annalisa Cocchia. Smart city and digital city: twenty years of terminology evolution. In *X Conference of the Italian Chapter of AIS, ITAIS*, pages 1–8, 2013.

Eleonora D'Andrea and Francesco Marcelloni. Detection of traffic congestion and incidents from gps trace analysis. *Expert Systems with Applications*, 73:43–56, 2017.

André De Palma and Sophie Dantan. Big data et politiques publiques dans les transports. Technical report, Economica, 2017.

André De Palma, Moez Kilani, and Stef Proost. Discomfort in mass transit and its implication for scheduling and pricing. *Transportation Research Part B: Methodological*, 71:1–18, 2015.

André De Palma, Robin Lindsey, and Guillaume Monchambert. The economics of crowding in public transport. *Journal of Urban Economics*, 101:106–122, 2017.

Raphaël Deswarte. *Régression linéaire et apprentissage: contributions aux méthodes de régularisation et d'agrégation (in English)*. PhD thesis, Université Paris Saclay, 2018.

Luc Devroye. The uniform convergence of nearest neighbor regression function estimators and their application in optimization. *IEEE Transactions on Information Theory*, 24(2):142–151, 1978.

Inderjit S Dhillon, Yuqiang Guan, and Brian Kulis. Kernel k-means: spectral clustering and normalized cuts. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 551–556. ACM, 2004.

Riccardo Di Clemente, Miguel Luengo-Oroz, Matias Travizano, Sharon Xu, Bapu Vaitla, and Marta C Gonzalez. Sequences of purchases in credit card data reveal lifestyles in urban populations. *Nature communications*, 9, 2018.

Chris Ding, Xiaofeng He, and Horst D Simon. On the equivalence of nonnegative matrix factorization and spectral clustering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 606–610. SIAM, 2005.

Kim-Anh Do, Peter Müller, and Feng Tang. A bayesian mixture model for differential gene expression. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54(3):627–644, 2005.

Bowen Du, Chuanren Liu, Wenjun Zhou, Zhenshan Hou, and Hui Xiong. Detecting pickpocket suspects from large-scale public transit records. *IEEE Transactions on Knowledge and Data Engineering*, 2018.

Gilles Duranton and Diego Puga. The growth of cities. In *Handbook of economic growth*, volume 2, pages 781–853. Elsevier, 2014.

Gilles Duranton and Matthew A Turner. Urban growth and transportation. *Review of Economic Studies*, 79(4):1407–1440, 2012.

Angel M Dzhambov, Iana Markevych, Boris G Tilov, and Donka D Dimitrova. Residential greenspace might modify the effect of road traffic noise exposure on general mental health in students. *Urban Forestry & Urban Greening*, 34:233–239, 2018.

Mohamed K El Mahrsi, Etienne Côme, Johanna Baro, and Latifa Oukhellou. Understanding passenger patterns in public transit through smart

card and socioeconomic data: a case study in Rennes, France. In *ACM SIGKDD Workshop on Urban Computing*, 2014a.

Mohamed K El Mahrsi, Etienne Côme, Johanna Baro, and Latifa Oukhellou. Understanding passenger patterns in public transit through smart card and socioeconomic data. 2014b.

Mohamed K El Mahrsi, Etienne Côme, Latifa Oukhellou, and Michel Verleysen. Clustering smart card data for urban mobility analysis. *IEEE Transactions on Intelligent Transportation Systems*, 18(3):712–728, 2017.

Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence: with application to music analysis. *Neural computation*, 21(3):793–830, 2009.

Maurizio Filippone, Francesco Camastra, Francesco Masulli, and Stefano Rovetta. A survey of kernel and spectral methods for clustering. *Pattern recognition*, 41(1):176–190, 2008.

Michael Fop and Thomas Brendan Murphy. Variable selection methods for model based clustering. *arXiv preprint arXiv:1707.00306*, 2017.

Chris Fraley and Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, 97(458):611–631, 2002.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, NY, USA, 2001.

Kamel Gadouche and Nathalie Picard. *L'accès aux données très détaillées pour la recherche scientifique*, chapter 5. Economica, 2017.

Nicolas Gendron-Carrier, Marco Gonzalez-Navarro, Stefano Polloni, and Matthew A Turner. Subways and urban air pollution. Technical report, National Bureau of Economic Research, 2018.

Robin Genuer. Variance reduction in purely random forests. *Journal of Nonparametric Statistics*, 24(3):543–562, 2012.

Robin Genuer, Jean-Michel Poggi, and Christine Tuleau. Random forests: some methodological insights. *arXiv preprint arXiv:0811.3619*, 2008.

Zoubin Ghahramani and Geoffrey E Hinton. The EM algorithm for mixtures of factor analyzers. Technical report, CRG-TR-96-1, University of Toronto, 1996.

Subhashis Ghosal, Jayanta K Ghosh, and Aad W Van Der Vaart. Convergence rates of posterior distributions. *Annals of Statistics*, 28(2): 500–531, 2000.

Edward F Gonzalez and Yin Zhang. Accelerating the Lee-Seung algorithm for non-negative matrix factorization. *Dept. Comput. & Appl. Math., Rice Univ., Houston, TX, Tech. Rep. TR-05-02*, 2005.

Bettina Grün. Model-based clustering. In Gilles Celeux, Sylvia Frühwirth-Schnatter, and Christian P. Robert, editors, *Handbook of Mixture Analysis*, pages 155–188. CRC Press, 2018.

Michael Guarnieri and John R Balmes. Outdoor air pollution and asthma. *The Lancet*, 383(9928):1581–1592, 2014.

Benjamin Guedj and Pierre Alquier. Pac-bayesian estimation and prediction in sparse additive models. *Electronic Journal of Statistics*, 7: 264–291, 2013.

Ronan Hamon, Pierre Borgnat, Cédric Févotte, Patrick Flandrin, and Céline Robardet. Factorisation de réseaux temporels: étude des rythmes hebdomadaires du système Vélo'v. In *Colloque GRETSI 2015*, 2015.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

Colin Harrison, Barbara Eckman, Rick Hamilton, Perry Hartswick, Jayant Kalagnanam, Jurij Paraszczak, and Peter Williams. Foundations for smarter cities. *IBM Journal of Research and Development*, 54(4):1–16, 2010.

Trevor Hastie and Robert Tibshirani. Generalized additive models. *Statistical Science*, 1(3):297–318, 1986.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Earl B Hunt, Janet Marin, and Philip J Stone. Experiments in induction. *Oxford, England: Academic Press*, 1966.

Ross Ihaka and Robert Gentleman. R: a language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5:299–314, 1996.

Zhanglong Ji, Zachary C Lipton, and Charles Elkan. Differential privacy and machine learning: a survey and review. *arXiv preprint arXiv:1412.7584*, 2014.

Shan Jiang, Joseph Ferreira, and Marta C Gonzalez. Activity-based human mobility patterns inferred from mobile phone data: A case study of singapore. *IEEE Transactions on Big Data*, 3(2):208–219, 2017.

Stephen C Johnson. Hierarchical clustering schemes. *Psychometrika*, 32 (3):241–254, 1967.

Jingu Kim and Haesun Park. Sparse nonnegative matrix factorization for clustering. 2008.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Pierre-Antoine Laharotte, Romain Billot, Etienne Come, Latifa Oukhellou, Alfredo Nantes, and Nour-Eddin El Faouzi. Spatiotemporal analysis of bluetooth data: Application to a large urban network. *IEEE Transactions on Intelligent Transportation Systems*, 16(3):1439–1448, 2015.

Nicolas Lartillot and Hervé Philippe. A bayesian mixture model for across-site heterogeneities in the amino-acid replacement process. *Molecular biology and evolution*, 21(6):1095–1109, 2004.

Saskia Le Cessie and Johannes C Van Houwelingen. Ridge estimators in logistic regression. *Applied statistics*, pages 191–201, 1992.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

Jung Hoon Lee, Marguerite Gong Hancock, and Mei-Chih Hu. Towards an effective framework for building smart cities: Lessons from seoul and san francisco. *Technological Forecasting & Social Change*, 89:80–99, 2014.

Ryong Lee and Kazutoshi Sumiya. Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proceedings of the 2nd ACM SIGSPATIAL international workshop on location based social networks*, pages 1–10. ACM, 2010.

Chih-Jen Lin. Projected Gradient Methods for Non-negative Matrix Factorization. *Neural computation*, 19(10):2756–2779, 2007.

Todd Litman. Rail transit in america: a comprehensive evaluation of benefits. 2015.

Todd Litman. *Evaluating public transportation health benefits.* Victoria Transport Policy Institute, 2016.

Thomas Louail, Maxime Lenormand, Juan Murillo Arias, and José J Ramasco. Crowdsourcing the robin hood effect in cities. *Applied Network Science*, 2(1):11, 2017.

Xin Luo, Mengchu Zhou, Yunni Xia, and Qingsheng Zhu. An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Transactions on Industrial Informatics*, 10(2):1273–1284, 2014.

Andrzej Mackiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers and Geosciences*, 19:303–342, 1993.

James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

Donald W Marquardt and Ronald D Snee. Ridge regression in practice. *The American Statistician*, 29(1):3–20, 1975.

Cathy Maugis, Gilles Celeux, and Marie-Laure Martin-Magniette. Variable selection for clustering with gaussian mixture models. *Biometrics*, 65(3): 701–709, 2009a.

Cathy Maugis, Gilles Celeux, and Marie-Laure Martin-Magniette. Variable selection in model-based clustering: a general variable role modeling. *Computational Statistics & Data Analysis*, 53(11):3872–3882, 2009b.

Geoffrey J McLachlan and David Peel. *Finite mixture models.* John Wiley & Sons, 2004.

Geoffrey J McLachlan, David Peel, and RW Bean. Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics & Data Analysis*, 41(3-4):379–388, 2003.

Paul D McNicholas. Model-based clustering. *Journal of Classification*, 33 (3):331–373, 2016a.

Paul D McNicholas. *Mixture model-based classification*. CRC Press, 2016b.

Paul D McNicholas and Thomas Brendan Murphy. Parsimonious gaussian mixture models. *Statistics and Computing*, 18(3):285–296, 2008.

Mario Medvedovic, Ka Yee Yeung, and Roger Eugene Bumgarner. Bayesian mixture model based clustering of replicated microarray data. *Bioinformatics*, 20(8):1222–1232, 2004.

Jiali Mei, Yohann De Castro, Yannig Goude, and Georges Hébrail. Recovering multiple nonnegative time series from a few temporal aggregates. In *34th International Conference on Machine Learning (ICML)*, 2017.

Fateh Nassim Melzi, Allou Same, Mohamed Haykel Zayani, and Latifa Oukhellou. A dedicated mixture model for clustering smart meter data: identification and analysis of electricity consumption behaviors. *Energies*, 10(10):1446, 2017.

Benoit Meyronin. *Marketing territorial: enjeux et pratiques*. Vuibert, 2015.

Angela Montanari and Cinzia Viroli. Heteroscedastic factor mixture analysis. *Statistical Modelling*, 10(4):441–460, 2010.

Catherine Morency, Martin Trepanier, and Bruno Agard. Measuring transit use variability with smart-card data. *Transport Policy*, 14(3):193–203, 2007.

Keefe Murphy, Isobel Claire Gormley, and Cinzia Viroli. Infinite mixtures of infinite factor analysers: nonparametric model-based clustering via latent gaussian models. *arXiv preprint arXiv:1701.07010*, 2017.

Paula M Murray, Paul D McNicholas, and Ryan P Browne. A mixture of common skew-t factor analysers. *Stat*, 3(1):68–82, 2014.

Taewoo Nam and Theresa A Pardo. Conceptualizing smart city with dimensions of technology, people, and institutions. In *Proceedings of the*

12th annual international digital government research conference: digital government innovation in challenging times, pages 282–291. ACM, 2011.

Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

oneclick.ai. Forecasting vs regression. `https://www.oneclick.ai/guide/index.php/2018/02/07/forecasting-vs-regression/`, 02 2018.

John William Paisley, David M Blei, and Michael I Jordan. Bayesian nonnegative matrix factorization with stochastic variational inference., 2014.

Bei Pan, Yu Zheng, David Wilkie, and Cyrus Shahabi. Crowd sensing of traffic anomalies based on human mobility and social media. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 344–353. ACM, 2013.

Jindong Pang. *The Effect of Urban Transportation Systems on Employment Outcomes and Traffic Congestion.* PhD thesis, Syracuse University, 2018.

Francis Papon, Jean-Marie Beauvais, Sophie Midenet, Etienne Come, Nadine Polombo, Sylvie Abours, Leslie Belton Chevallier, and Claude Soulas. Evaluation of the bicycle as a feeder mode to regional train stations. *Transportation research procedia*, 25:2721–2740, 2017.

Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

Marie-Pier Pelletier, Martin Trépanier, and Catherine Morency. *Smart card data in public transit planning: a review.* CIRRELT, 2009.

Chengbin Peng, Xiaogang Jin, Ka-Chun Wong, Meixia Shi, and Pietro Liò. Collective human mobility pattern from taxi trips in urban area. *PloS one*, 7(4):e34487, 2012.

Mickaël Poussevin, Emeric Tonnelier, Nicolas Baskiotis, Vincent Guigue, and Patrick Gallinari. Mining ticketing logs for usage characterization with nonnegative matrix factorization. In *International Workshop on Modeling Social Media*, pages 147–164. Springer, 2014.

J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1): 81–106, 1986.

Adrian E Raftery and Nema Dean. Variable selection for model-based clustering. *Journal of the American Statistical Association*, 101(473): 168–178, 2006.

Andry Njato Randriamanamihaga, Etienne Côme, Latifa Oukhellou, and Gérard Govaert. Clustering the Vélib' origin-destinations flows by means of poisson mixture models. In *ESANN*, 2013.

Issi Romem. Paying for dirt: Where have home values detaches from construction costs? `https://www.buildzoom.com/blog/paying-for-dirt-where-have-home-values-detached-from-construction-costs`, 10 2017.

R Sathya and Annamma Abraham. Comparison of supervised and unsupervised learning algorithms for pattern classification. *International Journal of Advanced Research in Artificial Intelligence*, 2(2):34–38, 2013.

Erwan Scornet, Gérard Biau, and Jean-Philippe Vert. Consistency of random forests. *The Annals of Statistics*, 43(4):1716–1741, 2015.

Luca Scrucca, Michael Fop, Thomas Brendan Murphy, and Adrian E Raftery. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R journal*, 8(1):289, 2016.

George AF Seber. *Multivariate observations*, volume 252. John Wiley & Sons, 2009.

Shokri Z Selim and Mohamed A Ismail. K-means-type algorithms: A generalized convergence theorem and characterization of local optimality. *IEEE Transactions on pattern analysis and machine intelligence*, (1): 81–87, 1984.

Farial Shahnaz, Michael W Berry, V Paul Pauca, and Robert J Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, 2006.

Hugo Steinhaus. Sur la division des corp materiels en partie. *Bull. Acad. Polon. Sci*, 1(804):801, 1956.

Douglas Steinley and Michael J Brusco. Selection of variables in cluster analysis: an empirical comparison of eight procedures. *Psychometrika*, 73(1):125–144, 2008.

X Yu Stella and Jianbo Shi. Multiclass spectral clustering. In *null*, page 313. IEEE, 2003.

Charles J Stone. Consistent nonparametric regression. *The annals of statistics*, pages 595–620, 1977.

Winfried Stute. Asymptotic normality of nearest neighbor regression function estimates. *The Annals of Statistics*, 12(3):917–926, 1984.

Dennis L Sun and Cedric Fevotte. Alternating direction method of multipliers for non-negative matrix factorization with the beta-divergence. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 6201–6205. IEEE, 2014.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

Emeric Tonnelier, Nicolas Baskiotis, Vincent Guigue, and Patrick Gallinari. Anomaly detection in smart card logs and distant evaluation with twitter: a robust framework. *To appear in Neurocomputing*, 2018.

Florian Toqué, Mostepha Khouadjia, Etienne Come, Martin Trepanier, and Latifa Oukhellou. Short & long term forecasting of multimodal transport passenger flows with machine learning methods. In *Intelligent Transportation Systems (ITSC), 2017 IEEE 20th International Conference on*, pages 560–566. IEEE, 2017.

Gus Trompiz and Meredith Mazzilli. Veolia to sell stake in transport firm to germany's rethmann. *Reuters*, 9 2018.

John W Tukey. *Exploratory data analysis*, volume 2. Reading, Mass., 1977.

Frank G Van Oort. *Urban growth and innovation: Spatially bounded externalities in the Netherlands*. Routledge, 2017.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

Sanford Weisberg. *Applied linear regression*, volume 528. John Wiley & Sons, 2005.

Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

J. H. Wolfe. Object cluster analysis of social areas. MSc thesis, Univ. of California, 1963.

Mingrui Wu. Collaborative filtering via ensembles of matrix factorizations. In *Proceedings of KDD Cup and Workshop*, volume 2007, 2007.

Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–273. ACM, 2003.

Yuhong Yang. Can the strengths of AIC and BIC be shared? A conflict between model indentification and regression estimation. *Biometrika*, 92(4):937–950, 2005.

Zhirong Yang, Jukka Corander, and Erkki Oja. Low-rank doubly stochastic matrix decomposition for cluster analysis. *Journal of Machine Learning Research*, 17(187):1–25, 2016.

Jing Yuan, Yu Zheng, and Xing Xie. Discovering regions of different functions in a city using human mobility and pois. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 186–194. ACM, 2012.

Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Advances in neural information processing systems*, pages 1601–1608, 2005.

Junbo Zhang, Yu Zheng, and Dekang Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *AAAI*, pages 1655–1661, 2017.

Yu Zheng. Location-based social networks: Users. In *Computing with spatial trajectories*, pages 243–276. Springer, 2011.

Yu Zheng, Yanchi Liu, Jing Yuan, and Xing Xie. Urban computing with taxicabs. In *Proceedings of the 13th international conference on Ubiquitous computing*, pages 89–98. ACM, 2011.

Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):38, 2014a.

Yu Zheng, Xuxu Chen, Qiwei Jin, Yubiao Chen, Xiangyun Qu, Xin Liu, Eric Chang, Wei-Ying Ma, Yong Rui, and Weiwei Sun. A cloud-based knowledge discovery system for monitoring fine-grained air quality. Technical report, Microsoft, 2014b.

Eyal Ben Zion and Boaz Lerner. Learning human behaviors and lifestyle by capturing temporal relations in mobility patterns. In *Proceedings of the 25th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2017.

université PARIS-SACLAY | École doctorale de mathématiques Hadamard (EDMH)

Titre: Analyse de données volumineuses dans le domaine du transport

Mots clés: Statistiques, Apprentissage machine, Transport, Villes intelligentes, Fouille de données

Résumé: L'objectif de cette thèse est de proposer de nouvelles méthodologies à appliquer aux données du transport public. En effet, nous sommes entourés de plus en plus de capteurs et d'ordinateurs générant d'énormes quantités de données. Dans le domaine des transports publics, les cartes sans contact génèrent des données à chaque fois que nous les utilisons, que ce soit pour les chargements ou nos trajets. Dans cette thèse, nous utilisons ces données dans deux buts distincts. Premièrement, nous voulions être capable de détecter des groupes de passagers ayant des habitudes temporelles similaires. Pour ce faire, nous avons commencé par utiliser la factorisation de matrices non-négatives comme un outil de pré-traitement pour la classification. Puis nous avons introduit l'algorithme NMF-EM permettant une réduction de la dimension et une classification de manière simultanée pour un modèle de mélange de distributions multinomiales. Dans un second temps, nous avons appliqué des méthodes de régression à ces données afin d'être capable de fournir une fourchette de ces validations probables. De même, nous avons appliqué cette méthodologie à la détection d'anomalies sur le réseau.

Title: Big Data analysis in the field of transportation

Keywords: Statistics, Machine Learning, Transportation, Smart Cities, Data Mining

Abstract: The aim of this thesis is to apply new methodologies to public transportation data. Indeed, we are more and more surrounded by sensors and computers generating huge amount of data. In the field of public transportation, smart cards generate data about our purchases and our travels every time we use them. In this thesis, we used this data for two purposes. First of all, we wanted to be able to detect passenger's groups with similar temporal habits. To that end, we began to use the Non-negative Matrix Factorization as a pre-processing tool for clustering. Then, we introduced the NMF-EM algorithm allowing simultaneous dimension reduction and clustering on a multinomial mixture model. The second purpose of this thesis is to apply regression methods on these data to be able to forecast the number of check-ins on a network and give a range of likely check-ins. We also used this methodology to be able to detect anomalies on the network.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin,
France