UNIVERSITÉ PARIS-DAUPHINE

Airbnb New User Bookings

Yves Tran

May 28, 2020

Figure 0.1: Carte contenant les probabilités qu'un utilisateur réserve vers les destinations indiquées

CONTENTS

1 Introduction **3**

2 Analyse des données des utilisateurs **4**
2.1 Les destinations 4
2.2 Les variables 4
2.2.1 Le sexe 5
2.2.2 L'âge 6
2.2.3 Le mois d'inscription 8
2.2.4 L'année d'inscription 9
2.2.5 Les autres variables 10

3 L'analyse des historiques d'actions des utilisateurs **10**
3.1 Le temps écoulé sur le site 10
3.2 Les actions discriminatoires 11
3.2.1 Le statut de réservation 11
3.2.2 Les différentes destinations 14
3.3 L'enchaînement des actions 15
3.3.1 Le concept du Word2Vec en skipgram 15
3.3.2 Les résultats de l'encodage des séquences d'actions des utilisateurs 16
3.4 Le nombre d'actions effectués par les utilisateurs 17
3.5 Les autres variables 18

4 Classification des utilisateurs selon leur statut de réservation **18**

5 Suggestion des destinations parmi les utilisateurs ayant réservé **20**

6 Conclusion **23**

1 INTRODUCTION

Depuis 2008, Airbnb propose une plateforme de location et de réservation de logements pour particuliers. Cette compagnie américaine recense plusieurs millions d'annonces vers plus de 34000 destinations réparties sur près de 200 pays autour du monde. Après des chiffres de nouveaux utilisateurs inscrits qui augmentent chaque année, Airbnb lance en 2016 une compétition en science des données consistant à prédire la prochaine destination d'un nouvel utilisateur de la plateforme. Les données sont disponibles via le lien suivant : lien Kaggle

Figure 1.1: Airbnb

Dans le cadre de ce projet, nous allons tenter de construire un modèle répondant à la problématique du défi ; à savoir : réaliser 5 suggestions de destinations triées de la plus pertinente à la moins pertinente pour un utilisateur nouvellement inscrit donné. Les 12 destinations (ou étiquettes) à prédire sont :

- **US** : Etats-unis
- **FR** : France
- **CA** : Canada
- **GB** : Grande-Bretagne
- **ES** : Espagne
- **IT** : Italie
- **PT** : Portugal
- **NL** : Pays-Bas
- **DE** : Allemagne
- **AU** : Australie
- **NDF** : No Destination Found (l'utilisateur n'a pas fait de réservation)
- **other** : tout autre pays que ceux cités ci-dessus

Pour cela, nous sélectionnons 2 fichiers à traiter et analyser :

- **train-user-2.csv :** contenant les informations générales sur les utilisateurs (age, sexe, date d'inscription...),
- **sessions.csv :** contenant l'historique des activités des utilisateurs sur la plateforme. Notons que seuls les historiques des utilisateurs inscrits **après le 1er avril 2014** sont recensés dans ce fichier.

La grande difficulté dans ce problème réside dans les proportions très déséquilibrées des utilisateurs réservant dans les différentes destinations citées. Nous verrons plus en détail ce fait par la suite. C'est pourquoi nous adopterons l'approche suivante :

1. Analyser les informations générales des utilisateurs du fichier *train-users-2.csv* en rééquilibrant les classes pour tenter de séparer les individus qui réservent de ceux qui ne le font pas mais également au niveau des destinations choisies,
2. Analyser les historiques des actions des utilisateurs sur le site toujours en rééquilibrant des classes entre ceux qui réservent de ceux qui s'en abstiennent pour voir les différences de comportement,
3. Construire un modèle pour classifier ses utilisateurs selon qu'ils aient l'intention de faire une réservation ou non à partir des analyses effectuées,
4. Construire un modèle dans le but de prédire la probabilité qu'un utilisateur ayant l'intention de faire une réservation voyage dans ces destinations afin de réaliser une série de suggestions.

Ce rapport reprend les résultats les plus pertinents et intègre les explications théoriques manquantes des notebooks. C'est pourquoi, le lecteur est invité à lire les notebooks où tous les détails et études s'y trouvent.

2 Analyse des données des utilisateurs

2.1 Les destinations

Nous pouvons voir sur le graphique suivant que les classes sont très déséquilibrées. En effet, la classes NDF est présente à 58% suivie de la destination US à 29%. Les autres classes se partagent le reste des utilisateurs. Le Portugal (PT) représente moins de 1% des utilisateurs.

En regroupant les indivudus selon 2 classes : ceux qui réservent (marqués **DF**) et ceux qui ne réservent pas (**NDF**), le jeu de données devient un peu plus équilibrée. Ainsi, prédire si l'utilisateur à l'intention d'effectuer une réservation pourra être utile dans ce problème.

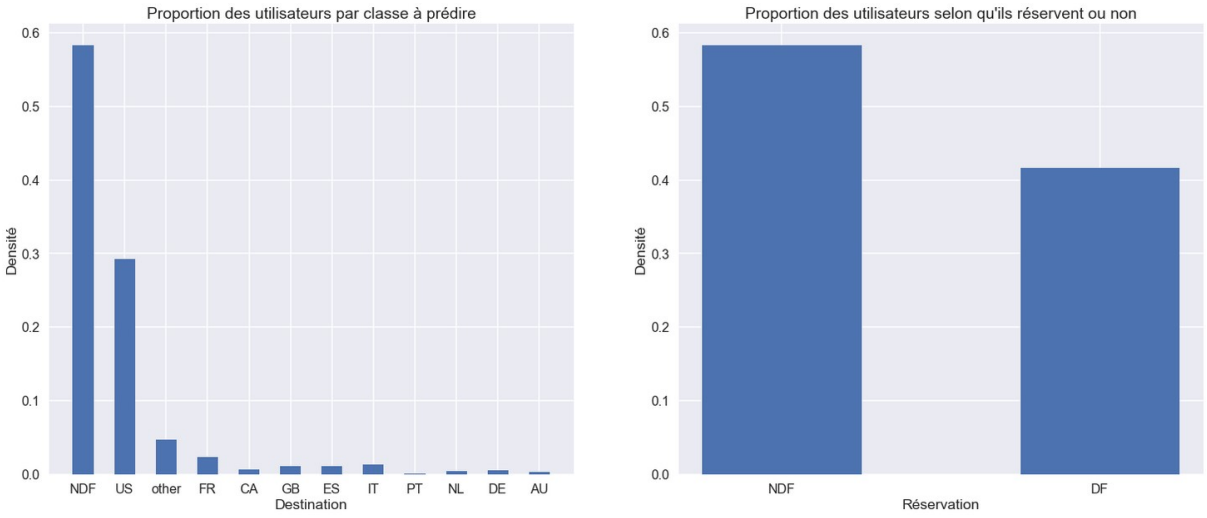

Figure 2.1: Répartition des classes/destinations parmi les utilisateurs

2.2 Les variables

Nous disposons de 15 variables listées ci-dessous. Chacune ont été analysées dans les notebooks, les résultats présentés ici sont uniquement les plus pertinentes.

- **id** : identifiant unique de l'utilisateur
- **date_account_created :** date à laquelle l'utilisateur s'est inscrit sur le site
- **timestamp_first_active :** date et heure à laquelle l'utilisateur s'est rendu sur le site pour la première fois, peut être antérieur à la variable précédente
- **date_first_booking :** date de la réservation s'il y a lieu
- **gender :** sexe de l'utilisateur
- **age :** age de l'utilisateur
- **signup_method :** méthode utilisée par l'utilisateur pour se connecter
- **signup_flow :** numéro de la page de recherche à laquelle l'utilisateur s'est décidé de s'inscrire
- **language :** la langue par défaut du navigateur de l'utilisateur
- **affiliate_channel :** le canal d'affiliation
- **affiliate_provider :** la source d'affiliation
- **first_affiliate_tracked :** la première campagne d'affiliation détectée
- **signup_app :** application utilisée par l'utilisateur pour se connecter
- **first_device_type :** type du premier appareil détecté
- **first_browser :** premier navigateur détecté

2.2.1 Le sexe

On distingue 4 valeurs possibles pour "gender" : "MALE", "FEMALE", "-unknown-" et OTHER.

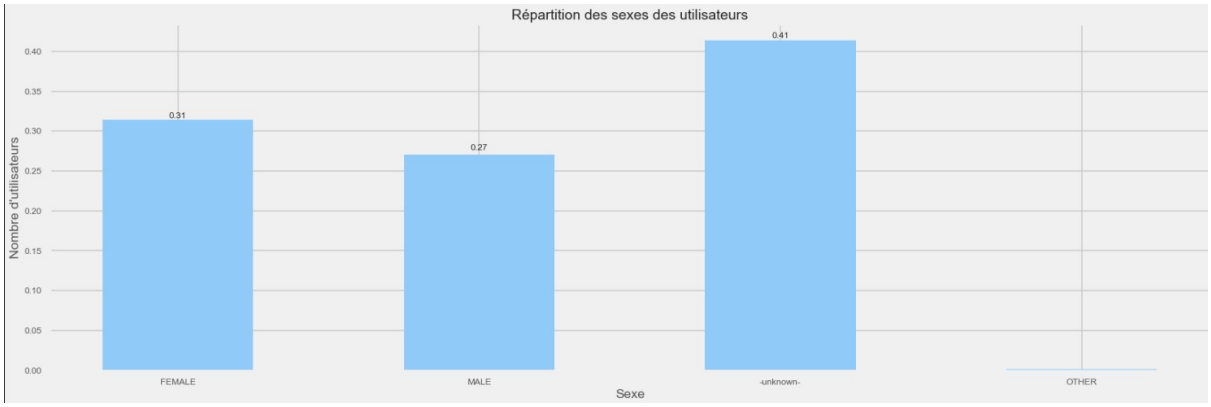

Figure 2.2: Répartition des sexes parmi les utilisateurs

- La valeur "-unknown-" est majoritairement représentée avec 41% d'utilisateurs.
- On voit qu'il y a un peu plus de femmes (31%) que d'homme (27%) dans ces données.
- La valeur "OTHER" est sous représentée très minoritaire (avec seulement 261 représentants en réalité).

Ainsi, la plupart des utilisateurs ne donnent pas d'information concernant leur sexe. Vérifions s'il n'y aurait pas de corrélation avec les variables cibles.

- La valeur "-unknown-" est représenté à 50% parmi les NDF.
- Les utilisateurs qui ont renseigné leur sexe ont tendance à réserver une destination.
- Les utilisateurs ayant réservé en Allemagne (DE) et aux Pays-Bas (NL) sont plus généralement des hommes à 39% et 37%.
- Celles ayant réservé en Italie (IT), France (FR) et Grande-Bretagne (GB) sont plus souvent des femmes à 38% environ pour les 3 pays.

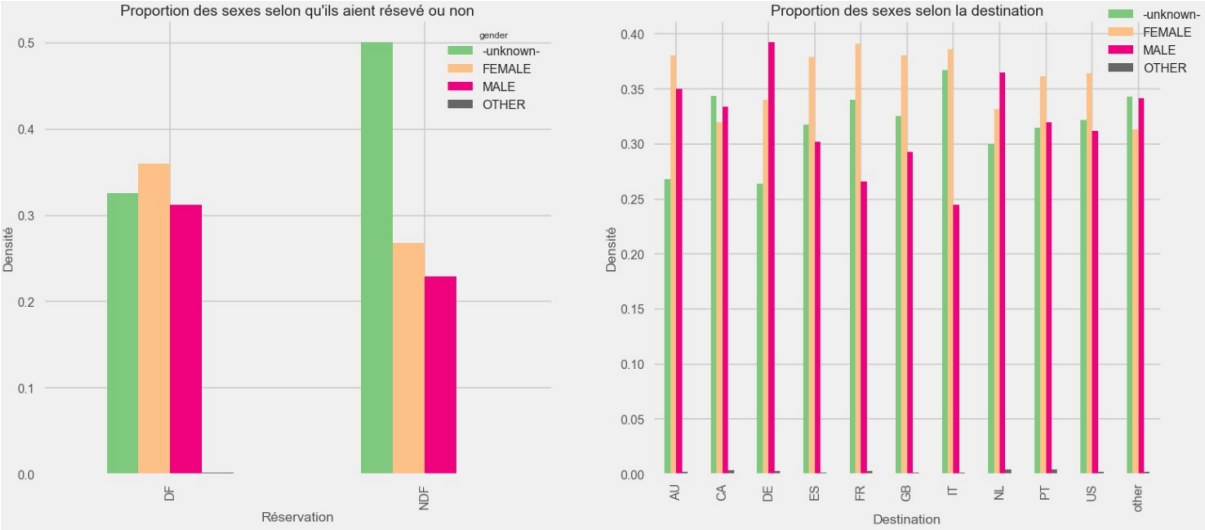

Figure 2.3: Répartition des sexes parmi les utilisateurs et destinations

2.2.2 L'âge

La variable représentant l'âge n'est pas toujours correcte. En effet, certains individus ont inscrits leur année de naissance ou encore une date improbable ce qui donne des valeurs d'âges supérieurs à 100. Nous analysons ici les âges inférieurs à 100. Nous voyons que la médiane est à 34 ans (Fig 2.4) et que la courbe d'âge prend la forme d'une gaussienne asymétrique (Fig 2.5).

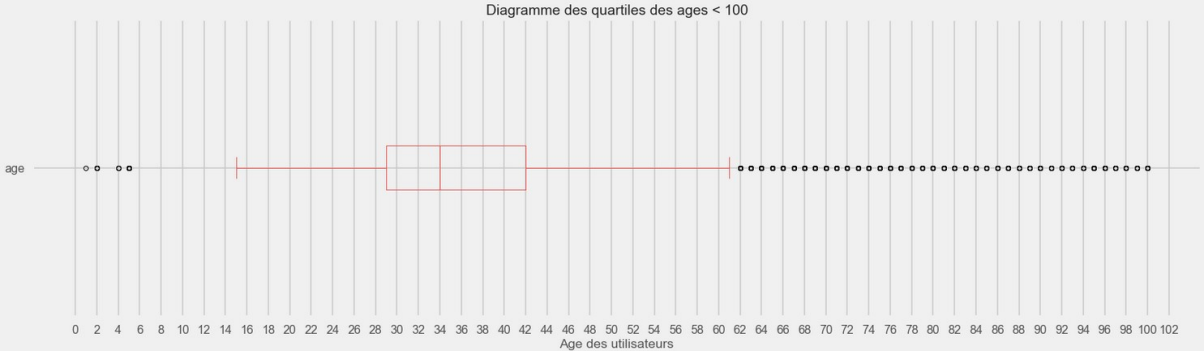

Figure 2.4: L'âge des utilisateurs

- Les individus marqués comme étant DF ont un âge légèrement plus centré vers les 30 ans que ceux marqués comme étant NDF.
- Ces derniers ont une population qui est légèrement plus importante au niveau des 50 ans.

En considérant, sur la Fig 2.6, les utilisateurs comme étant *jeunes* si leur âge est inférieur à la médiane (34 ans) et *âgé* dans le cas contraire, nous pouvons observé que les jeunes utilisateurs préfèrent voyager vers l'Espagne et le Portugal (à 53% parmi ceux réservant dans ces destinations) tandis les utilisateurs plus âgés préfèrent aller en Grande-Bretagne où la proportion de *jeunes* est à 46%.

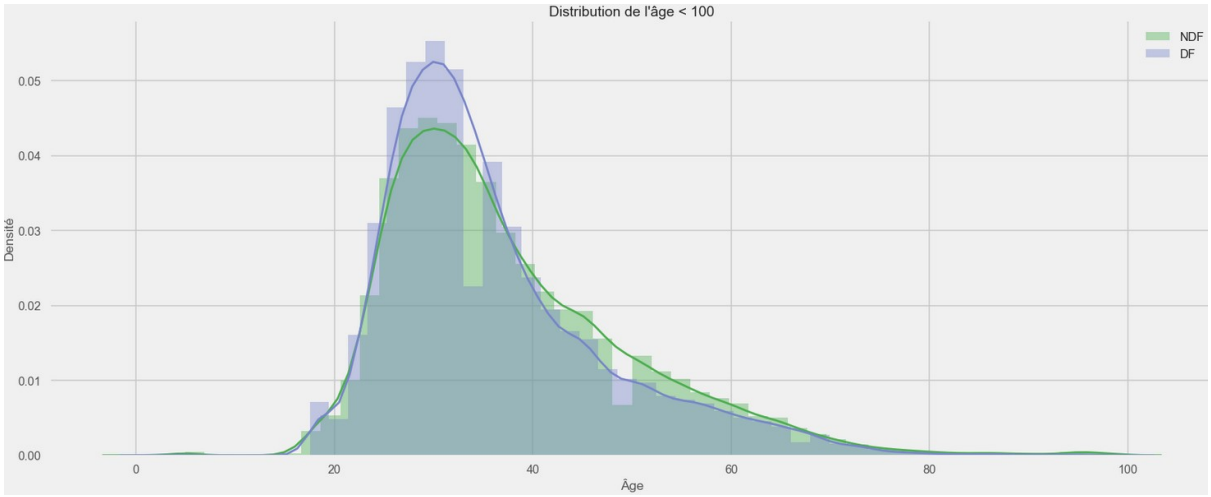

Figure 2.5: L'âge des utilisateurs selon qu'ils réservent ou non

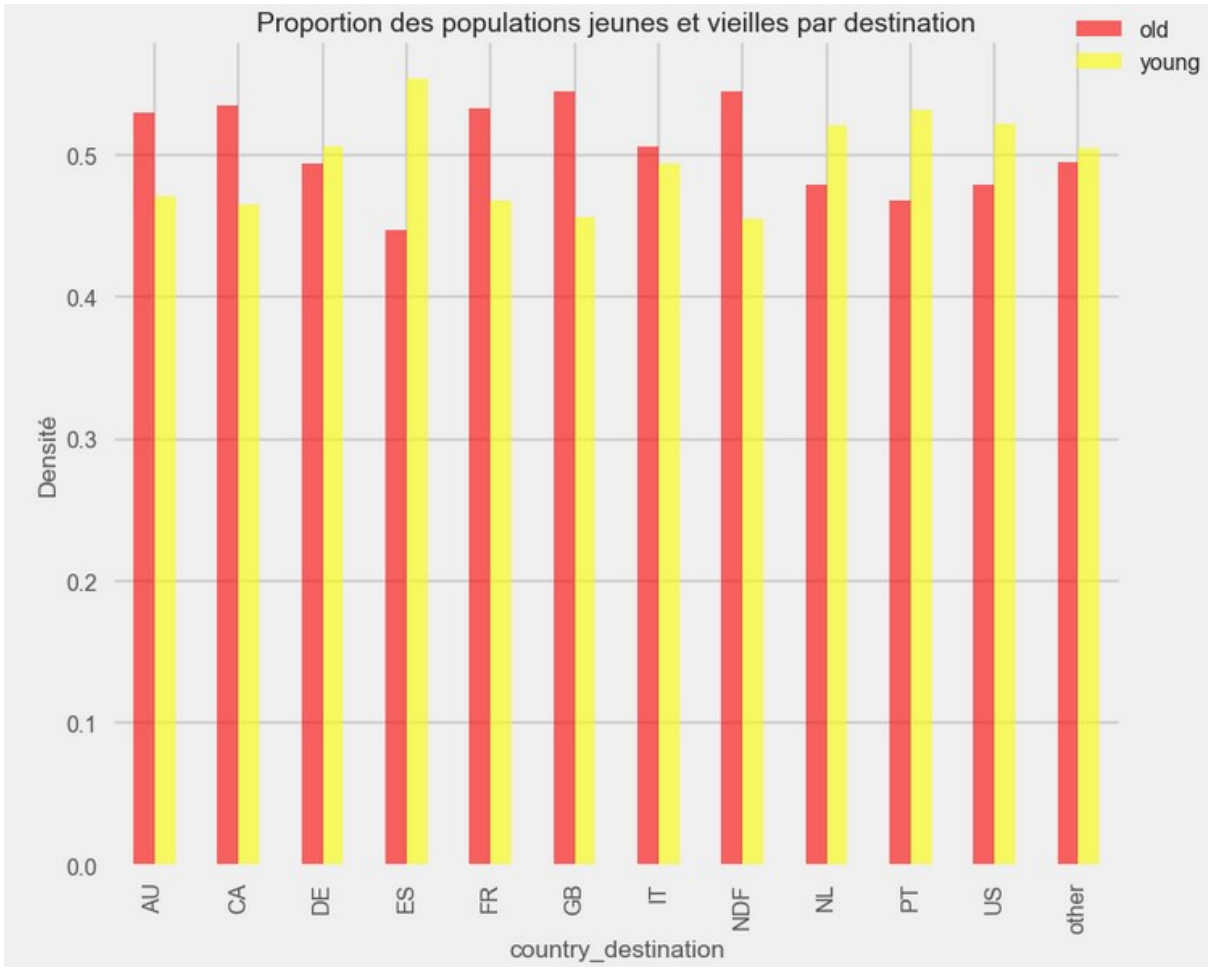

Figure 2.6: Le choix de la destination selon la catégorie d'âge des utilisateurs

2.2.3 *Le mois d'inscription*

Notez que le moment d'inscription coïncide presque tout le temps avec le premier moment passé sur le site à l'exception de quelques utilisateurs que nous négligeons. Ici, nous pouvons nous permettre de confondre les variables *date_account_created* et *timestamp_first_active*. On peut analyser le mois où les utilisateurs se sont inscrits. Les graphiques montrent des pics d'inscriptions en Janvier, Mai et Juin. Il s'agit généralement des mois qui précèdent les périodes de vacances (Summer and Spring breaks).

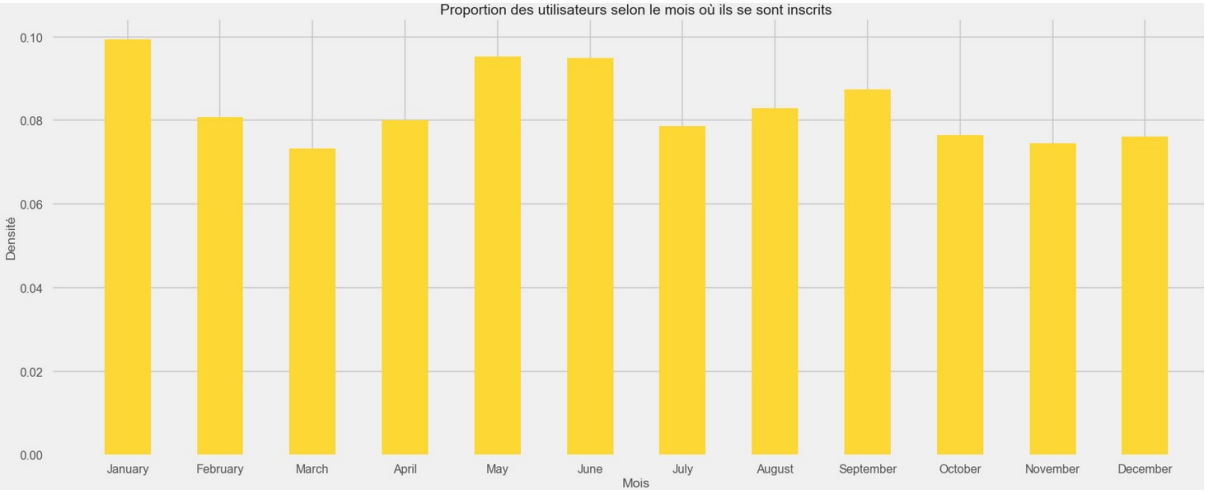

Figure 2.7: Densité d'inscription des utilisateurs selon le mois

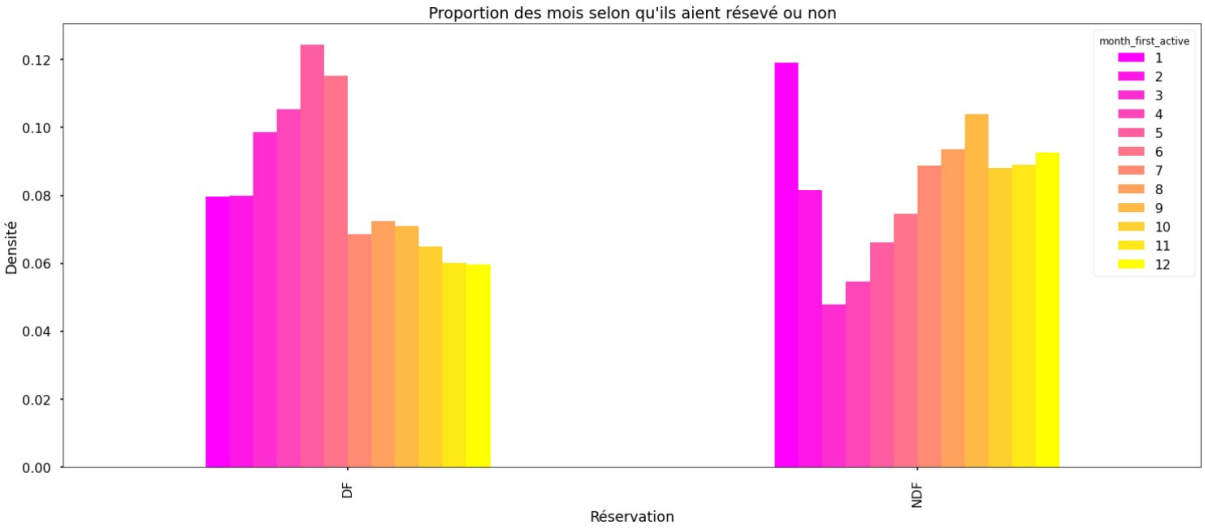

Figure 2.8: Proportion d'utilisateurs selon le mois d'inscription et leur statut de réservation

- De Mars à Juin est la période où les utilisateurs ayant l'intention de réserver s'inscrivent le plus.
- Le reste du temps, les NDF sont plus prépondérants.

Le lieu de destination varie aussi selon le mois. Sur le graphique qui suit, nous pouvons constaté que tous les pays de l'hémisphère nord (représenté ici par l'Allemagne, pour plus de détail,

voir le notebook 2) connaissent un pic vers mai/juin puis le nombre d'inscrit chute. Seul l'Australie (l'unique pays de l'hémisphère sud dans notre étude) a cette tendance inversée. On ressort l'idée que les utilisateurs préfèrent partir lorsque le temps est plus chaud.

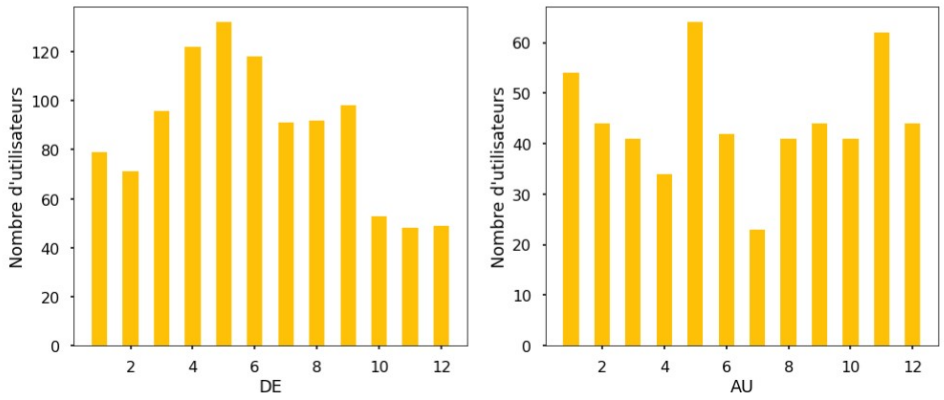

Figure 2.9: Nombre d'utilisateurs selon le mois d'inscription et la destination (Allemagne ou Australie)

2.2.4 L'année d'inscription

En représentant la densité d'inscription pour les différentes années et destinations, nous observons que :

- En 2012, près de 40% des utilisateurs ont réservé en Allemagne.
- En 2013, la destination la plus fréquentés est l'Australie avec 45% des utilisateurs.
- En 2014, peu de NDF présent. On rappelle qu'on ne dispose pas d'utilisateurs inscrit après début avril 2014.

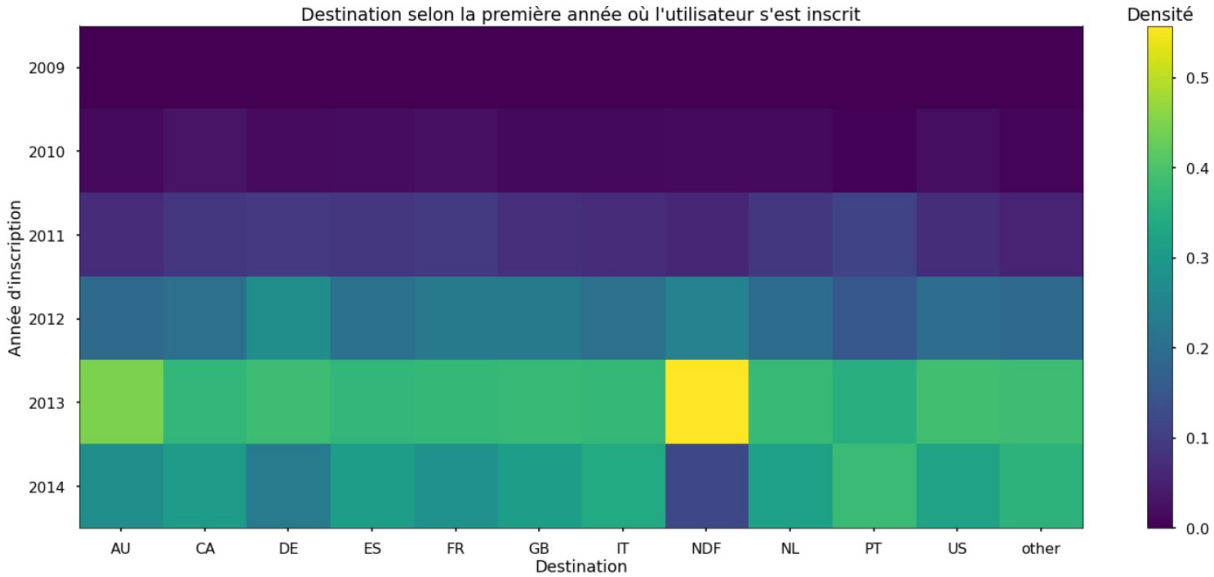

Figure 2.10: Densité d'inscription selon l'année et la destination

2.2.5 Les autres variables

Les autres variables ne sont pas présentées dans ce rapport mais sont détaillées dans les notebooks. Elles contiennent moins d'information pertinente et contribuent peu à la séparation des utilisateurs dans notre problème. La liste des variables étudiée est décrite ci-dessous :

- **signup_method :** méthode utilisée par l'utilisateur pour se connecter
- **signup_flow :** numéro de la page de recherche à laquelle l'utilisateur s'est décidé de s'inscrire
- **language :** la langue par défaut du navigateur de l'utilisateur
- **affiliate_channel :** le canal d'affiliation
- **affiliate_provider :** la source d'affiliation
- **first_affiliate_tracked :** la première campagne d'affiliation détectée
- **signup_app :** application utilisée par l'utilisateur pour se connecter
- **first_device_type :** type du premier appareil détecté
- **first_browser :** premier navigateur détecté
- **le jour d'inscription**
- **l'heure d'inscription**

3 L'ANALYSE DES HISTORIQUES D'ACTIONS DES UTILISATEURS

Dans cette section, les analyses portent sur les données du fichier *sessions.csv* qui est organisé de la manière suivante :

- **user_id** : l'identifiant de l'utilisateur
- **action** : le nom de l'action réalisé par l'utilisateur (*lookup, search_results, ...*)
- **action_detail** : détail supplémentaire sur l'action correspondant (*click, submit, ...*)
- **action_type** : le type d'action (*view_search_results, signup...*)
- **device_type** : l'appareil utilisé par l'utilisateur pour faire l'action
- **secs_elapsed** : le temps passé en secondes pour réaliser l'action.

Les variables *action, action_detail, action_type* sont dépendantes les unes des autres et c'est pourquoi nous les traiterons comme une seule variable *action_name* en les fusionnant, qui définit les actions possibles. Ainsi, nous disposons de 390 actions différentes parmi 73815 utilisateurs du fichier disposant d'un historique. Ici, nous rééquilibrons les classes DF et NDF en prenant 20000 utilisateurs de chaque étiquette.

3.1 Le temps écoulé sur le site

En analysant le temps passé par les utilisateurs sur la plateforme en heures, nous observons que :

- La courbe des NDF suit la courbe de la loi exponentielle.
- Le nombre d'heures passées en relativement élevé (de l'ordre de la centaine) si on considère que toutes ces activités se sont passées avant la première réservation. C'est pourquoi, il est très probable les activités post-reservations sont également enregistrées.
- Les DF passent beaucoup plus de temps sur le site que les NDF, probablement car ils font l'effort de renseigner leur profil, passer du temps sur les différentes annonces, écrire aux hôtes pour poser des questions, etc. Il est possible qu'une partie des NDF ne font que regarder, mais ce ne sont que des suppositions.

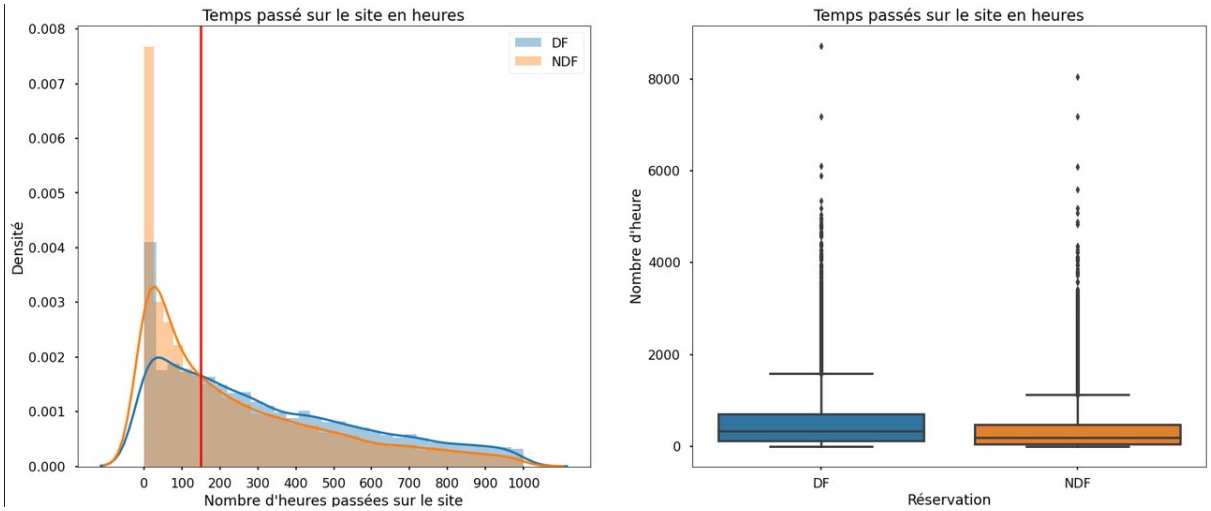

Figure 3.1: Le temps passé en heures par les utilisateurs selon qu'ils réservent ou non

3.2 Les actions discriminatoires

3.2.1 Le statut de réservation

Nous cherchons les actions ayant les plus grandes différences entre le nombre d'utilisateurs DF et d'utilisateurs NDF qui ont réalisé ces actions. Pour chaque action est calculé le taux de différence T_d définit par :

$$T_d = \frac{Nombre\_de\_DF\_ayant\_fait\_l'action - Nombre\_de\_NDF\_ayant\_fait\_l'action}{Nombre\_de\_DF\_ayant\_fait\_l'action + Nombre\_de\_NDF\_ayant\_fait\_l'action} \in [-1, 1]$$

Les actions qui nous intéressent sont celles ayant les plus grands taux de différence en valeur absolue.

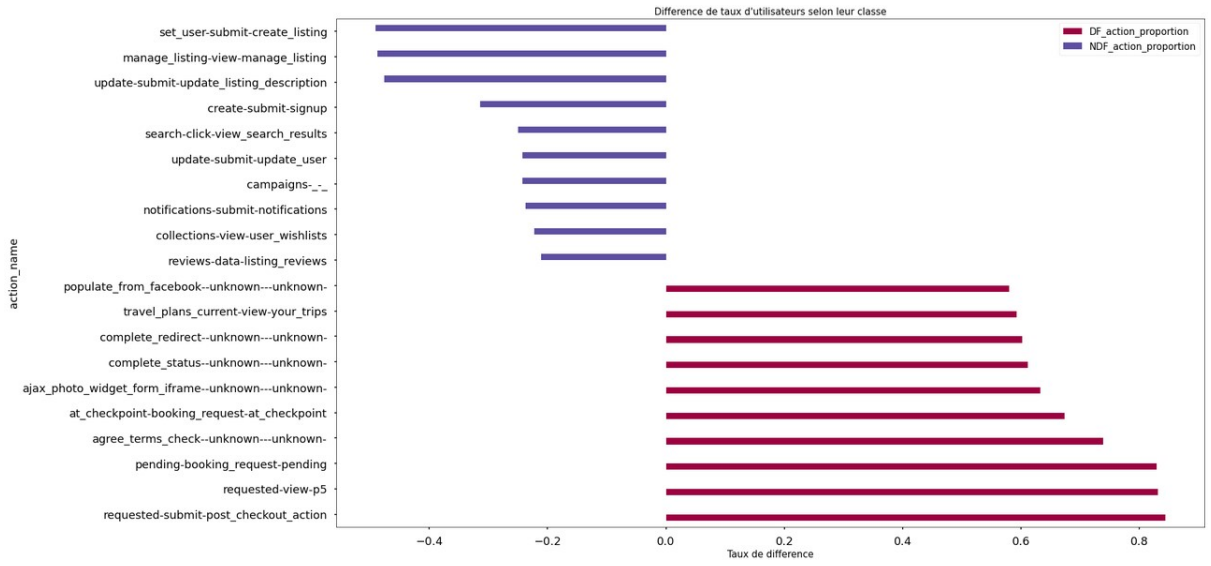

Figure 3.2: Les actions ayant les plus grand taux de différence en valeur absolue

Sur la Fig 3.2 est affiché les 10 actions les plus "fréquentes" chez les NDF et chez les DF en ordonnée. Plus la valeur est proche de +1, plus cette action est propre aux DF. Plus cette valeur est

proche de -1, plus cette action est propre aux NDF. Nous gardons uniquement les actions qui ont été effectuées par au moins 1000 utilisateurs.

- On voit sur le graphique que les DF ont une forte tendance à effectuer les actions liées aux réservations : soumettre la requête de réservation ("requested"), mis en attente ("pending"), "agree_terms_check", communiquer avec l'hôte ("message_to_host_change", 'message_to_host_focus")...
- Les actions relatives au terme "listing" renvoient au processus de création ou mise à jour d'une page d'un logement. Ce sont donc des actions liées aux hôtes qui louent leur logement. C'est ce qu'on voit dans les actions les plus effectuées parmi les NDF : "manage_listing" et "set_user-submit-create_listing". Ainsi, une portion des NDF sont des hôtes.
- On atteint des actions avec un taux de difference > 0.8 du côté des DF.
- Du côté des NDF, le taux de différence des actions ne dépassent pas les 0.5.

La limite dans l'utilisation de ces informations au sein de notre future modèle de prédiction réside dans le fait que beaucoup d'utilisateurs n'ont réalisé aucune de ces actions. C'est ce que montre la figure suivante. Par exemple, seuls 3000 utilisateurs ont fait l'action *requested-submit-post_action_checkout* qui est la plus discriminatoire parmi les actions fréquentes chez les utilisateurs DF.

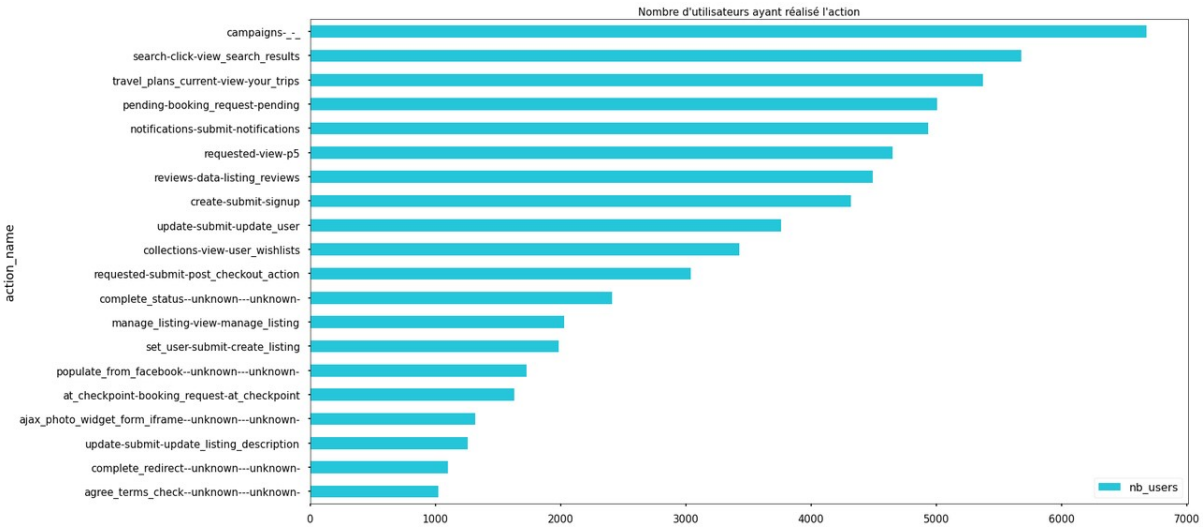

Figure 3.3: Nombre d'utilisateurs ayant réalisé les actions listées

En réponse à cette situation, nous pouvons analyser les actions les plus fréquentes et observer les proportions de DF et NDF. Nous pouvons observer que :

- 'ajax_refresh_subtotal-click-change_trip_characteristics' est réalisée par 57% des DF contre seulement 40% des NDF.
- Les NDF s'abstiennent de faire l'action 'similar_listings-data-similar_listings', seuls 43% le font contre 55% chez les DF.

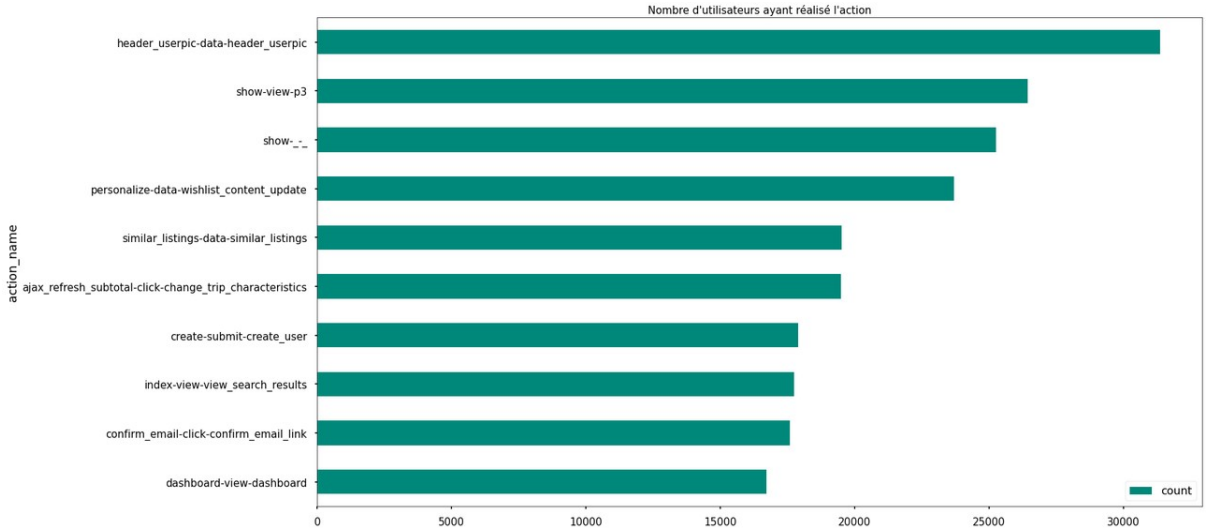

Figure 3.4: Fréquence des actions les plus courante parmi l'ensemble des utilisateurs

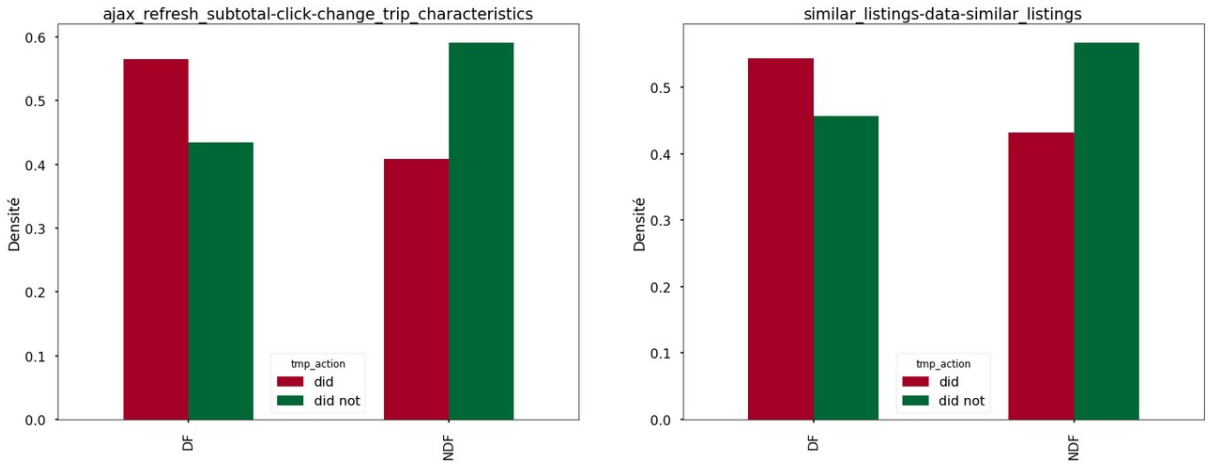

Figure 3.5: Densité des utilisateurs DF et NDF selon qu'il aient réservé ou non sur les deux actions les plus discriminatoires et les plus fréquentes

3.2.2 *Les différentes destinations*

Nous pouvons réaliser un travail similaire sur l'ensemble des destinations ; à savoir :

- compter le nombre d'utilisateur ayant effectué l'action selon les destinations avant de réduire les données (diviser par l'écart-type),
- trier les actions ayant les plus grande disparités, permettant ainsi de séparer les utilisateurs. Pour cela, on va calculer, pour chaque action,
- l'écart-type entre les destinations,
- sélectionner les n actions avec un écart-type maximal.

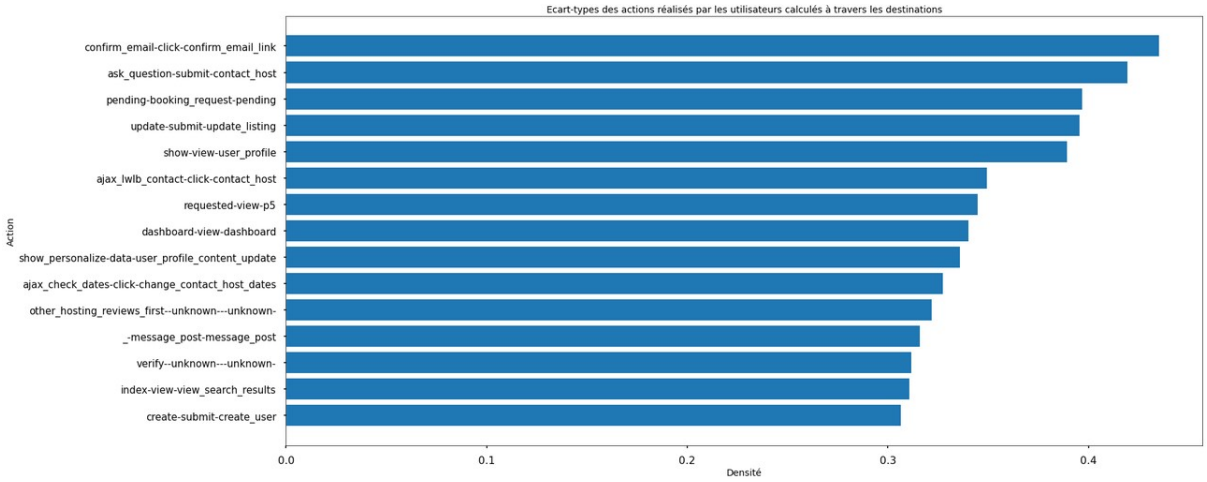

Figure 3.6: Densité d'utilisation des actions les plus discriminatoires et les plus fréquentes

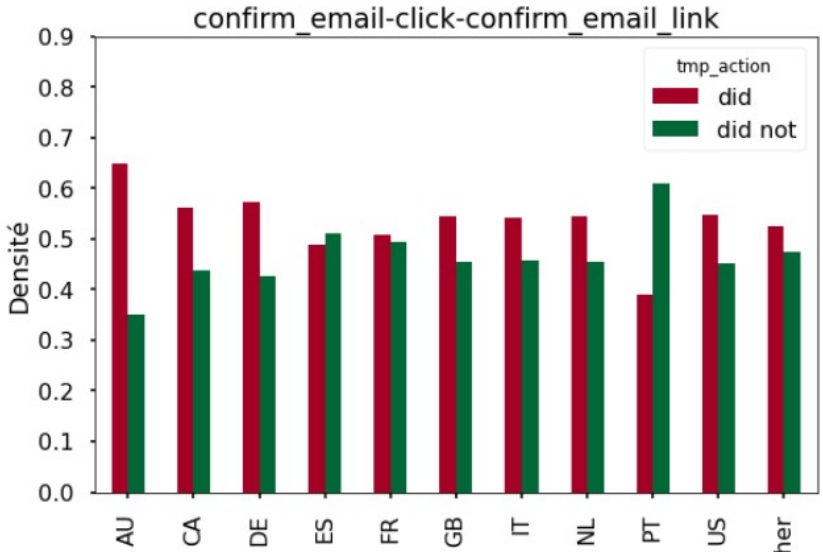

Figure 3.7: Proportions des utilisateurs ayant réalisé l'action concernée selon leur destination choisie

L'action *confirm_email-click-confirm_email_link* est intéressante car on voit que la moitié des populations d'utilisateurs ne fait pas cette action pour le Portugal et l'Espagne. Cette action est plus

fréquentes avec les utilisateurs réservant vers les autres pays (US, Australie...).

3.3 L'enchaînement des actions

On va regarder si les séquences d'actions réalisées par les utilisateurs permettent de discriminer ceux qui réservent de ceux qui ne réservent pas. Pour cela, on peux s'inspirer des méthodes de Natural Language Processing (NLP) puisque la situation est analogue : il s'agit d'étudier une suite item où l'ordre à son importance. Dans notre cas, on va utiliser le Word2Vec[1] qui permet de représenter une action (soit un mot en NLP) sous la forme d'un vecteur de dimensionnalité réduite.

3.3.1 Le concept du Word2Vec en skipgram

Je présente, ici le concept du Word2Vec dans notre contexte avec les actions (et non en NLP). Le but est de pouvoir représenter chacune de nos actions sous la forme d'un vecteur (on parle *d'embeddings*). Pour cela, la méthode du Word2Vec encode les actions de la même manière qu'un *Bag of Words*, c'est-à-dire en ordonnant les actions et chaque action peut être représentée par un vecteur v tel que $\forall i$ la coordonnée v_i est définie par :

$$v_i = \begin{cases} 1 & \text{si i correspond au rang de l'action dans la suite ordonnée} \\ 0 & \text{sinon} \end{cases} \tag{3.1}$$

Pour chaque action, on peut définir son **contexte** par l'ensemble des actions qui avoisinent fréquemment l'action courante lorsqu'on regarde les séquences d'actions des utilisateurs ; par exemple en regardant les 5 actions qui précèdent et les 5 actions qui suivent l'action courante. On peut encoder le contexte d'une action en utilisant un *Bag of Words* tel que pour v' son vecteur, $\forall i$,v'_i est définie par :

$$v'_i = \begin{cases} 1 & \text{si l'action au rang i appartient au contexte de l'action étudiée} \\ 0 & \text{sinon} \end{cases} \tag{3.2}$$

Le Word2Vec utilise un réseau de neurones à 2 couches avec une architecture en Skip-gram dans notre cas ; c'est-à-dire qu'on va entraîner le modèle à prédire un contexte étant donnée une action en entrée du réseau de neurones. Les vecteurs en entrée et en sortie du modèle sont ceux définis précédemment. Une fois que le modèle est capable de prédire le contexte d'une action, toutes les informations sur l'enchaînement des actions est compressé dans les poids de la première couche de neurones. En effet, le résultat en sortie de tout neurone n'est que le produit scalaire entre le vecteur d'entrée et les poids du neurone (à une fonction d'activation près). Cela signifie qu'au sein de tout neurone, le i^e poids interagie uniquement avec la i^e coordonnée du vecteur en entrée, soit, pour la première couche, la i^e action. Ainsi, pour retrouver *l'embedding* de la i^e action, il suffit d'extraire dans l'ordre, les i^e poids de chaque neurone de la première couche du modèle. La taille du vecteur obtenu correspond au nombre de neurone sur la première couche qui est arbitraire. En le prenant inférieur au nombre d'actions total, on réussit à compresser les informations d'une action par rapport à un encodage en *Bag of Words*.

3.3.2 *Les résultats de l'encodage des séquences d'actions des utilisateurs*

En utilisant des vecteurs de taille 20, on peut afficher les actions les plus similaires à *show-view-user_profile* en calculant la similarité cosinus entre les vecteurs. Sur la figure 3.8, les actions similaires à "show-view-user_profile" sont liées aux informations et profile des utilisateurs. On retrouve :

- social_connections-data-user_social_connections : connexion entre utilisateurs
- recommendations-data-user_friend_recommendations : faire une recommandation à un utilisateurs
- reviews-data-listing_reviews : lire les avis sur l'annonce d'un utilisateur hôte.

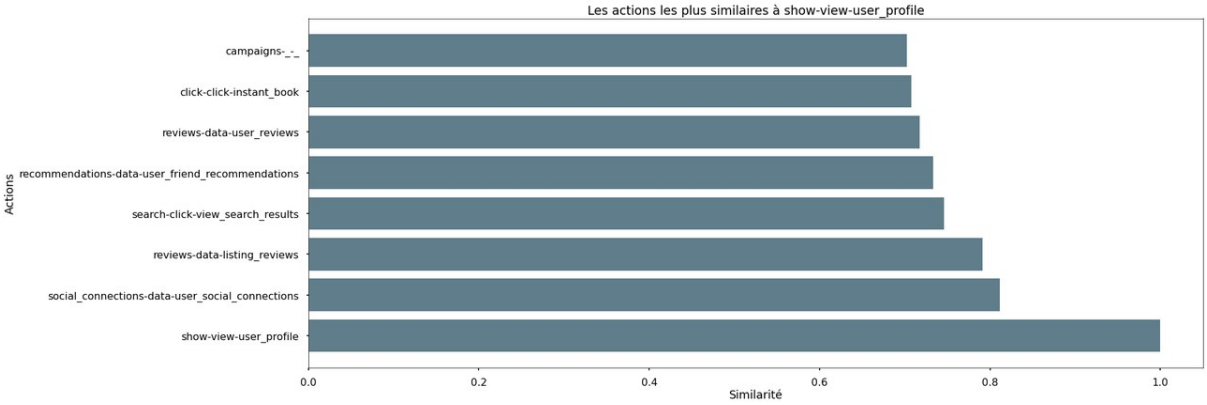

Figure 3.8: Actions similaires à *show-view-user_profile* en utilisant le Word2Vec par similarité cosinus

À chaque utilisateur peut être attribué un vecteur représentant la moyenne des vecteurs des actions qu'il a effectué. On utilise l'ACP pour afficher les utilisateurs sur 2 dimensions.

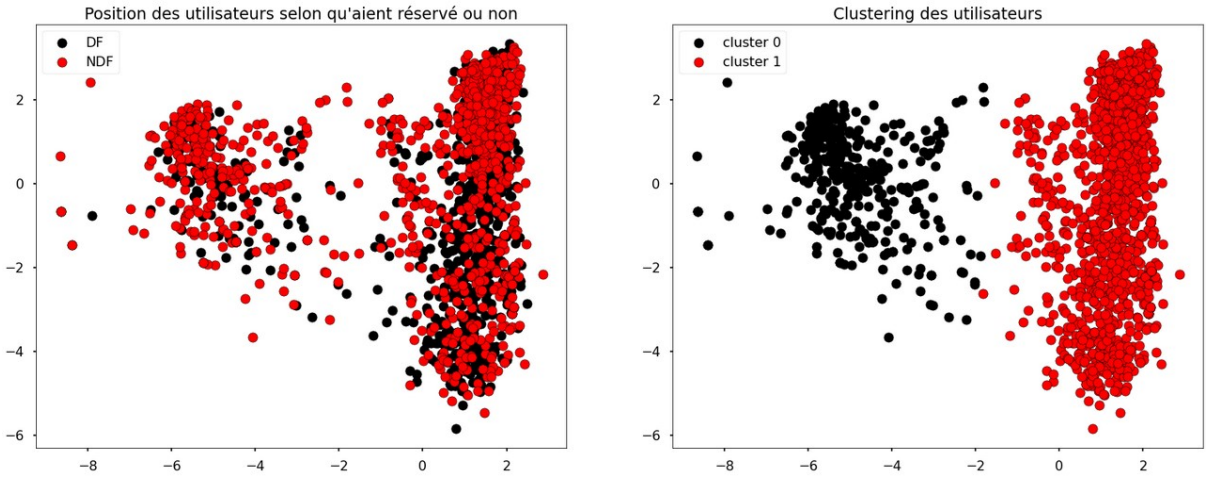

Figure 3.9: ACP des séquences d'action les utilisateurs

On voit sur la figure 3.9 que :

- la méthode du Word2Vec a réparti les utilisateurs en 2 clusters ;

- ces 2 clusters ne répresentent pas ceux qui réservent et ceux qui ne réservent pas ;
- le cluster 1 contient plus de NDF que de DF.

On peut tenter de retrouver les 2 groupes d'utilisateurs formés par le Word2Vec. En testant plusieurs variables, c'est le type d'appareil qui correspond le mieux. On affiche, ici, les utilisateurs PC et TELEPHONE sous 2 couleurs différentes.

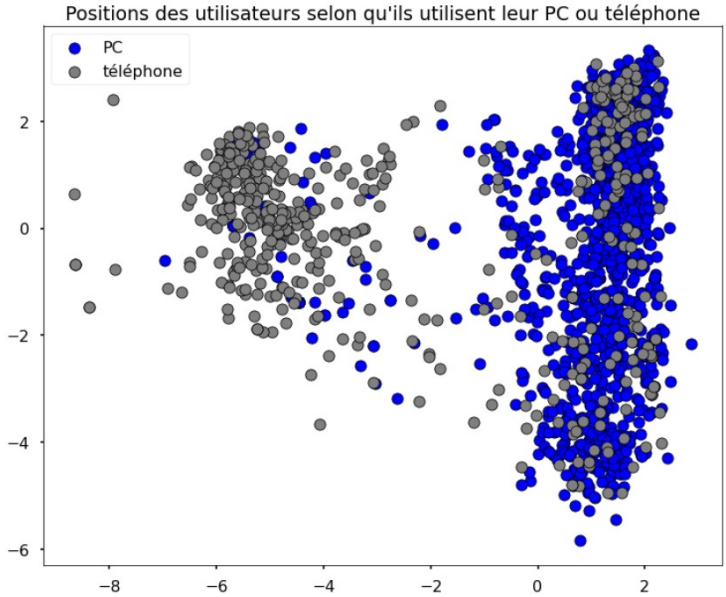

Figure 3.10: ACP des séquences d'actions des utilisateurs selon le type d'appareil utilisé

- l'enchainement des actions ne fait que séparer les utilisateurs selon le type d'appareil utilisé (PC ou TELEPHONE) et peut donc être résumé par cette catégorisation ;
- les utilisateurs sur téléphone sont plus souvent des NDF.

3.4 Le nombre d'actions effectués par les utilisateurs

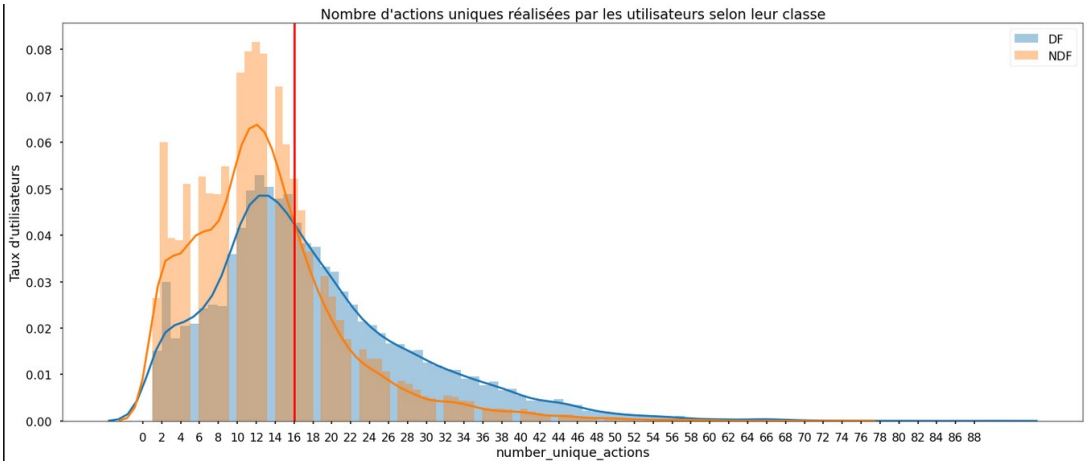

Figure 3.11: La densité d'utilisateurs selon le nombre d'actions et leur statut de réservation

- Le nombre d'actions uniques chez les NDF est généralement moindre que chez les DF.
- On voit une séparation où la densité est plus forte du côté des NDF en-dessous de 16 actions différentes et une densité plus forte du côté des DF après cette valeur.

3.5 Les autres variables

Les autres variables sont moins pertinents et ne sont donc pas présentées ici. Elles sont laissées à l'étude du lecteur sur les notebooks. La liste est présentée ci-dessous :

- le type d'appareil utilisé
- le nombre d'appareil utilisé

4 Classification des utilisateurs selon leur statut de réservation

Les classes à prédire sont :

- **DF :** l'utilisateur fait une réservation,
- **NDF :** l'utilisateur ne fait pas de réservation.

La plupart des données sont qualitatives. Nous choisissons donc d'utiliser des modèles d'arbre pour nos prédictions, soit les arbres de décisions et les forêts aléatoires.

La métrique utilisée est le **f1-score** étant donné que les classes sont déséquilibrées.

Dans le cas des arbres, il est possible, en réponse au déséquilibre des classes, d'accorder un poids équivalent à l'importance d'une donnée. L'idée est de donner plus d'importance à la classe minoritaire et moins d'importance à la classe majoritaire. Cela peut nous être utile dans la manipulation d'arbres étant donné qu'au niveau des feuilles, seul la proportion des classes engendre la prédiction. Ainsi, rééquilibrer les proportions en les multipliant par un poids permet une meilleure prédiction. Les poids p_i choisis pour $i \in \{DF, NDF\}$ sont :

$$p_{DF} = 1 - \frac{|DF|}{|DF| + |NDF|}, p_{NDF} = 1 - \frac{|NDF|}{|DF| + |NDF|} \tag{4.1}$$

On peut utiliser le même principe dans le cadre des forêts aléatoires avec une variante en plus où les poids de chaque arbre est adapté à l'échantillon. Chaque forêt aléatoire est composée de 1000 arbres de décisions, les variables sont tirées aléatoirement pour avoir $\sqrt{m}$ où m est le nombre de variables. Les échantillons sont définis par la méthode du *bootstrap* (tirage avec remise).

Les variables utilisées sont :

- **le mois d'inscription**
- **le sexe**
- **la catégorie d'âge**
- **la méthode de connexion**
- **le numéro de page d'inscription**
- **la source d'affiliation**
- **l'application utilisée**
- **la première catégorie de navigateur détectée**
- **la tranche du nombre d'heures passé sur la plateforme**
- **l'heure d'inscription**
- **la tranche du nombre d'actions uniques réalisées**

- **la catégorie de l'appareil le plus utilisé par l'utilisateur**
- **la fréquence arrondie au dixième d'une action discriminatoire dans l'historique de l'utilisateur (à répéter pour 15 actions relevés)**
- **booléen indiquant si l'utilisateur a réalisé l'action (à répéter pour les 8 plus communes)**

Pour chaque modèle, on calcule la moyenne du f1-score obtenu après une évaluation par **validation croisée** pour un échantillonnage de taille $k = 3$. Le meilleur score est obtenu avec une forêt aléatoire de profondeur 11 avec un rééquilibrage des poids des classes.

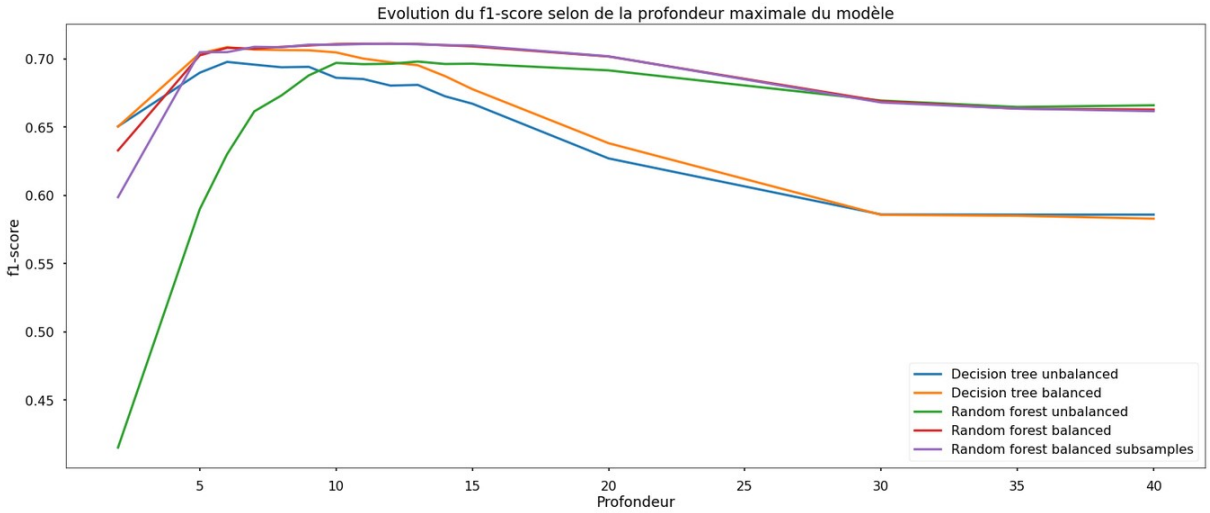

Figure 4.1: Évolution du f1-score selon la profondeur des modèles

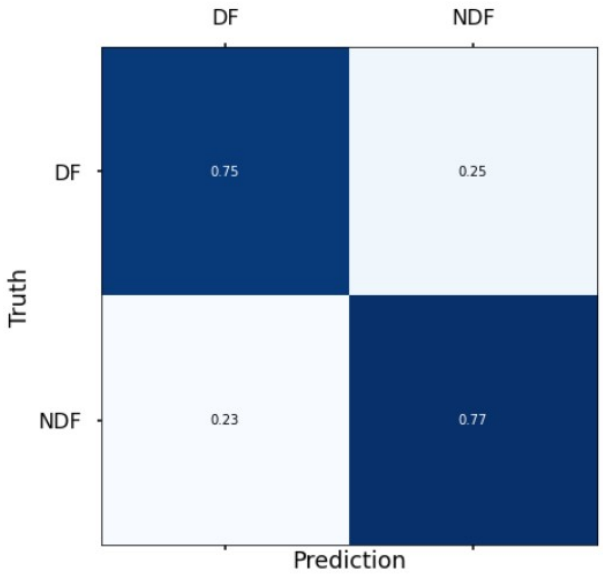

Figure 4.2: Matrice de confusion avec le meilleur modèle - f1-score à 0.7083

5 SUGGESTION DES DESTINATIONS PARMI LES UTILISATEURS AYANT RÉSERVÉ

Nous allons utiliser les mêmes modèles que précédemment (avec des paramètres différents). Les suggestions sont réalisées à partir des proportions des classes restantes dans les feuilles des arbres. Pour les forêts aléatoires, on prend la proportion d'arbres votant pour chacune des classes.

Pour mesurer la qualité des suggestions, la métrique utilisée est le *Normalized Discounted Cumulative Gain (NDCG)* définie, pour une série de 5 suggestions, comme suit :

$NDGC = \frac{DCG}{IDCG}$ où IDCG est le score DCG de la meilleure suggestion

$DCG = \sum_{i=1}^{5} \frac{2^{rel}-1}{log_2(i+1)}$

$$rel = \begin{cases} 1 & \text{l'utilisateur est allé dans cette destination} \\ 0 & \text{sinon} \end{cases} \tag{5.1}$$

Dès que la bonne destination a été trouvé dans la série de suggestions, le reste des destinations est ignoré.

Les classes étant très déséquilibrées, nous réduisons le nombre d'exemples marqués comme étant *US*, classe représentée à 70% dans le jeu de données, pour en garder qu'un tiers des ces exemples.

Les variables utilisées sont :

- **l'année d'inscription**
- **le mois d'inscription**
- **la langue par défaut de l'utilisateur**
- **le sexe**
- **la catégorie d'âge**
- **la méthode de connexion**
- **le numéro de page d'inscription**
- **la source d'affiliation**
- **l'application utilisée**
- **le canal d'affiliation**
- **la première catégorie de navigateur détectée**
- **la première catégorie d'appareil détectée**
- **la première campagne d'affiliation détectée**
- **l'heure d'inscription**
- **la fréquence arrondie au dixième d'une action courante dans l'historique de l'utilisateur (à répéter pour 15 actions relevées)**
- **le nombre d'appareils utilisés**

À titre comparatif, prendre les destinations dans l'ordre de fréquence *US, other, FR, IT, GB* parmi les utilisateurs du jeu de données produit un $NDCG = 0.5707$.

On utilise une forêt aléatoire avec échantillons rééquilibrés et 1000 arbres de décisions à travers différentes profondeurs autorisées et on obtient la figure 5.1 :

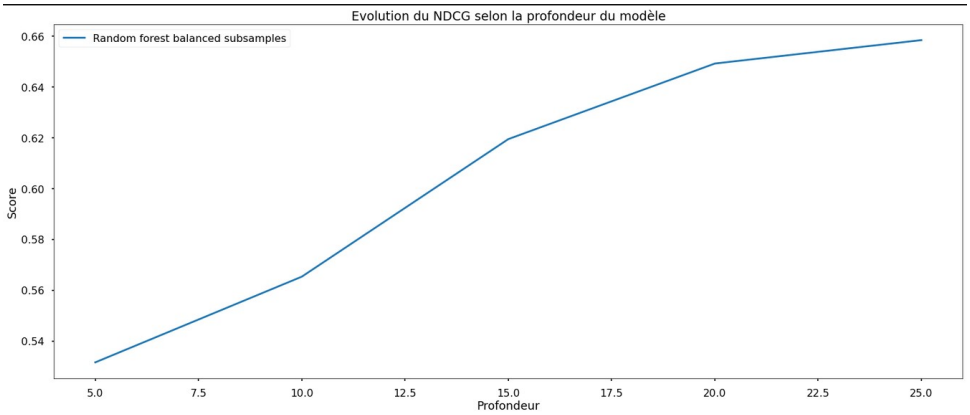

Figure 5.1: Évolution du NDCG selon la profondeur du modèle

On soupçonne un surapprentissage. En effet, il est possible que le NDCG ne nous permet pas de détecter les erreurs commises par le modèle étant donnée qu'on se base sur une série de destinations. Les destinations attendues se retrouvent vers la fin de la série au lieu d'être en premier lieu. Sur la figure 5.2, le modèle avec une profondeur de 25 confondu souvent la destination *other* avec le reste des classes ce qui témoigne bien d'un "mauvais" modèle. On rappelle que *other* est la classe la plus représentée qui suit *US*.

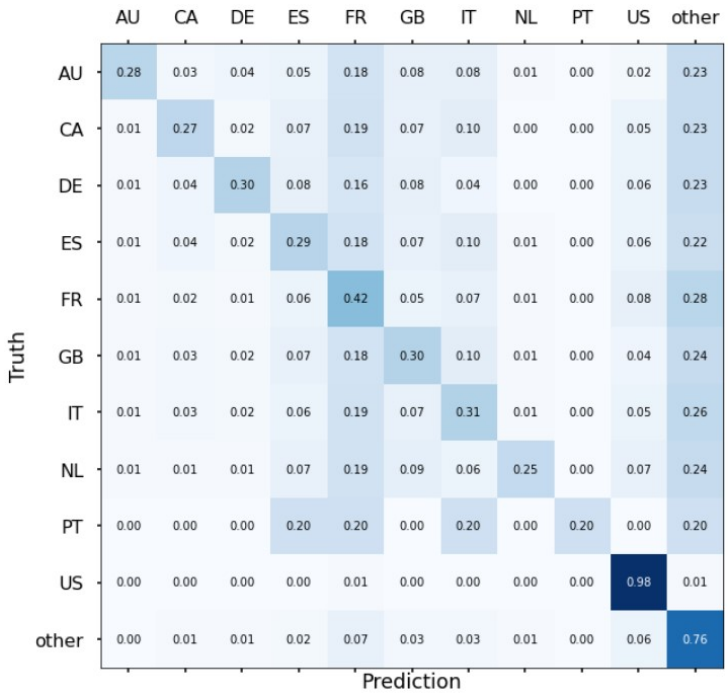

Figure 5.2: Matrice de confusion de la forêt aléatoire de profondeur 25

Notez que la matrice de confusion calculée ci-dessus est basée sur le NDCG. Pour chaque

série de suggestions à un utilisateur du jeu de données test, on ajoute le score NDCG à la position (t, p) où t est la destination attendue et p est la destination prédite pour p allant de la première suggestion à t si trouvée (ou jusqu'à la dernière suggestion dans le cas échéant). Les valeurs sont normalisées en divisant par la somme en ligne.

En utilisant une profondeur à 15, on obtient un $NDCG = 0.6197$ avec des prédictions convenables pour les destinations moins prisées (*ES, DE, CA...*). La matrice de confusion est affichée ci-dessous. Le modèle confond moins la destination *other* avec les autres. On remarque que *PT* est peu correctement prédit puisqu'il y peu d'exemples marqué par cette classe. Par ailleurs, le modèle prédit presque parfaitement la destination *US* (le pourcentage manquant est dû aux arrondis).

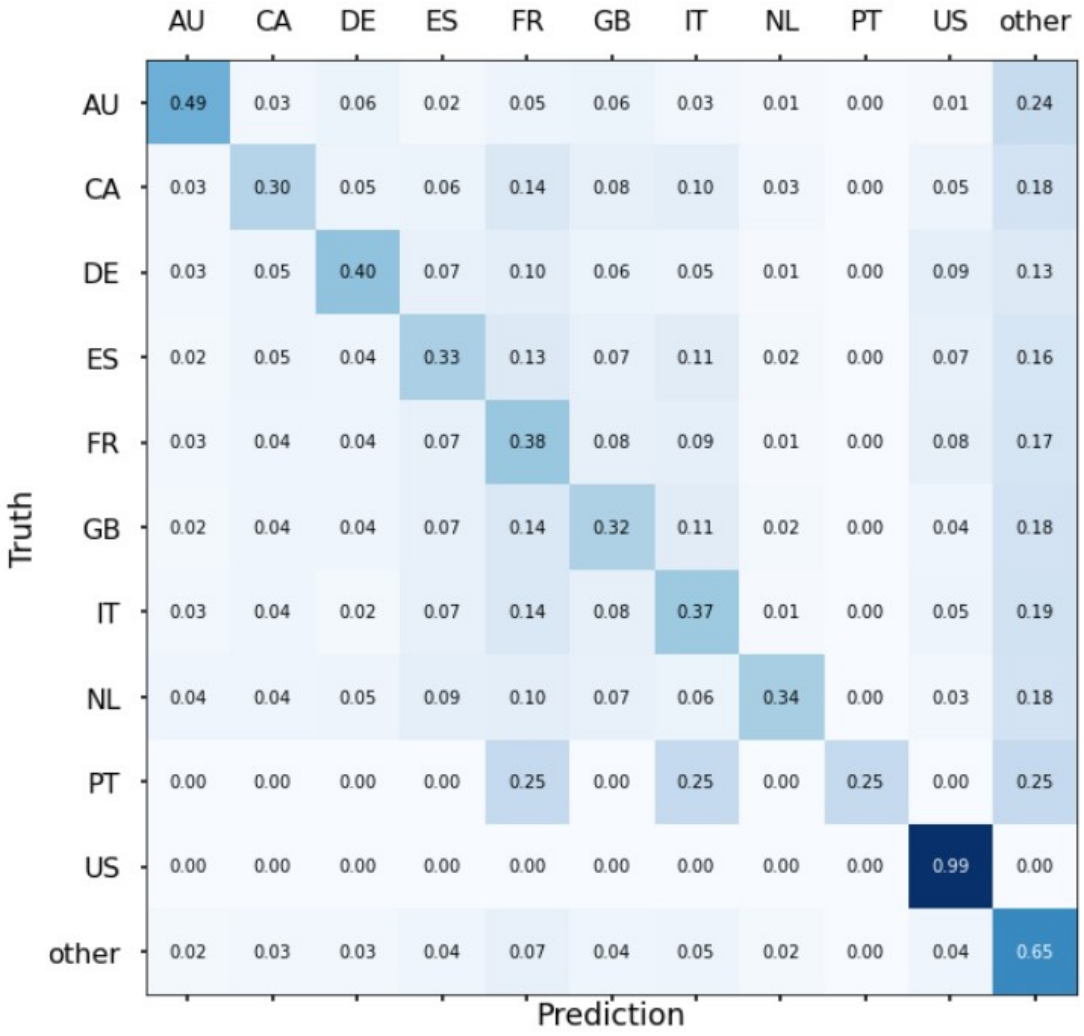

Figure 5.3: Matrice de confusion de la forêt aléatoire de profondeur 15

Il est possible de visualiser les destinations prédites avec les probabilités sur une carte. Le lecteur est invité à regarder la fin du notebook 5 (voir figure 0.1).

6 CONCLUSION

La suggestion des destinations des utilisateurs a pu se faire en prédisant si ces derniers ont l'intention d'effectuer une réservation ou non. La suite des suggestions peut être complétée par un deuxième modèle recommandant les destinations les plus pertinentes à partir des données d'entraînement. Nous avons vu que les méthodes ensemblistes (les forêts aléatoires) ont une meilleure performances que les modèles simples.

La grande difficulté de ce projet réside dans la gestion des classes déséquilibrées. Ici, les données sont très hétérogènes avec des étiquettes occupant près de 60% du jeu de données tandis que d'autres n'occupent à peine 1% du jeu. Grâce à des méthodes de pondération des classes, *undersampling* et par le choix des métriques, ce problème a pu être déjoué et des gains de performances ont été aperçus. De plus, le fait d'implémenter les algorithmes de Machine Learning soi-même a été un grand défi. Cependant, cela m'a permis de bien comprendre le fonctionnement ainsi que les limites de ces derniers. Par ailleurs, il est possible d'améliorer les performances en intégrant des méthodes de boosting aux modèles ensemblistes qui augmente généralement l'efficacité des modèles.

REFERENCES

[1] Greg Corrado Jeffrey Dean Tomas Mikolov, Kai Chen. *Efficient Estimation of Word Representations in Vector Space.* arXiv, 2013.