FACULTÉ DES SCIENCES ECONOMIQUES
UNIVERSITÉ DE RENNES 1

UNIVERSITÉ RENNES 2

Rapport de projet AS

20/01/2023

Le trafic de taxis à New York

Observations et lissages sous R

BEGO Liam - EL FARKH Youssef - LEBRETON Louis – LEHMANN Matthieu

Université de Rennes I & II

Licence 3 MIASHS spécialité Economie

Année 2022-2023

Sommaire

Figures

Introduction

Les taxis jaunes new yorkais, icones de la ville de New York représentent un secteur économique important pour la ville, fournissant un service de transport régulier aux habitants et aux touristes.

Par le biais de la Taxi and Limousine Commission, la ville de New York collecte l'ensemble des trajets réalisés par ces taxis depuis 2009. Ces données brutes sont disponibles à partir du lien suivant : https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page

Un utilisateur de Kaggle nommé Julien JATYKA, Data Scientist chez Canal +, a proposé un set de ces données qu'il a lui-même agrégées par tranche sur plus d'une année. Ces données agrégées disponibles sur le lien : https://www.kaggle.com/datasets/julienjta/nyc-taxi-traffic seront celles que nous traiterons ici. Il s'agit donc du nombre de passagers de taxi à New York par tranche de 30 minutes sur la période allant de juillet 2014 à décembre 2015. Le fichier de base comprend 10 320 observations et 2 variables : Date & Horaire et Nombre de passagers.

Afin d'obtenir une série temporelle intéressante à traiter dans le cadre du projet, nous avons décidé de nous focaliser sur le mois de janvier 2015. Nous analyserons donc par la suite 1488 observations (car 48 tranches de 30 minutes * 31 jours).

Notre objectif sera d'analyser ces données pour déterminer les tendances et les prédictions nécessaires à réaliser afin de mieux comprendre l'utilisation des taxis à New York. Pour cela, nous allons utiliser le logiciel R qui nous permettra d'effectuer une analyse descriptive des données, une moyenne mobile, un lissage ainsi que des prédictions.

I / Analyse descriptive

Dans cette analyse descriptive, nous cherchons à mieux comprendre l'évolution du nombre de passagers de taxis à New York. Les résultats de cette analyse nous aideront lors de la réalisation de la moyenne mobile et du lissage.

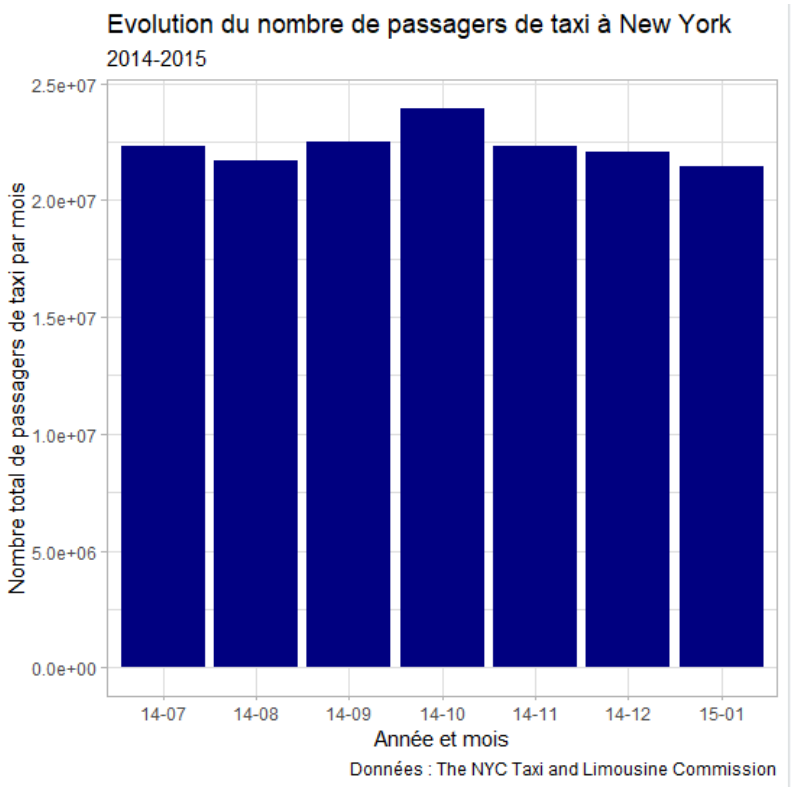

Figure 1 : Évolution du nombre de passagers de taxi à New York de juillet 2014 à février 2015

Ce premier graphique nous montre le nombre total de trajets effectués par mois à New York. On constate qu'environ 20 000 000 de passagers prennent un taxi chaque mois. Les valeurs observées sont plutôt proches d'un mois à l'autre avec un pic pour octobre 2014.

Concentrons maintenant sur notre période d'étude définie préalablement : le mois de janvier 2015.

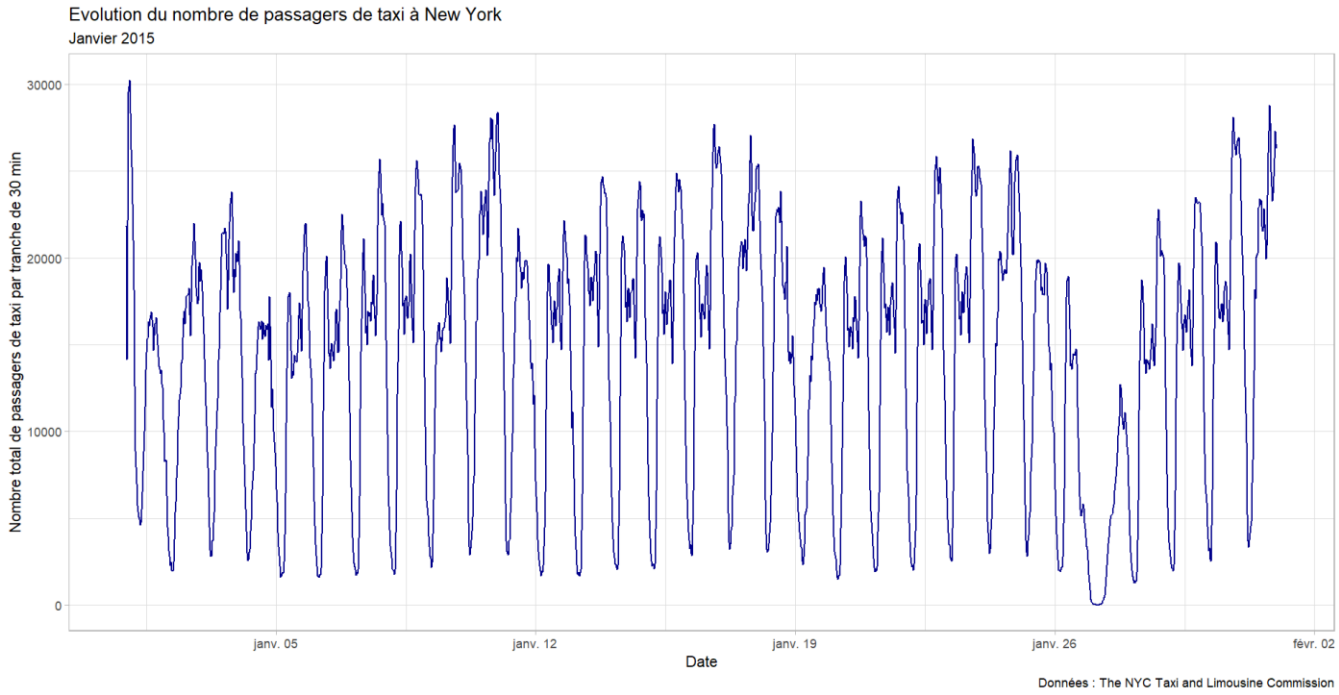

Figure 2 : Évolution du nombre de passagers de taxi à New York au mois de janvier 2015

La figure 2 représente le nombre de passagers par tranche de 30 minutes durant le mois de janvier 2015. Nous pouvons voir une saisonnalité journalière avec des pics et des creux qui semblent apparaître chaque jour au même horaire. On peut voir graphiquement qu'il n'y pas de tendance à la hausse ou la baisse sur l'ensemble du mois ce que confirme le modèle de régression linéaire réalisé (p-value = 0.75 et R2=0).

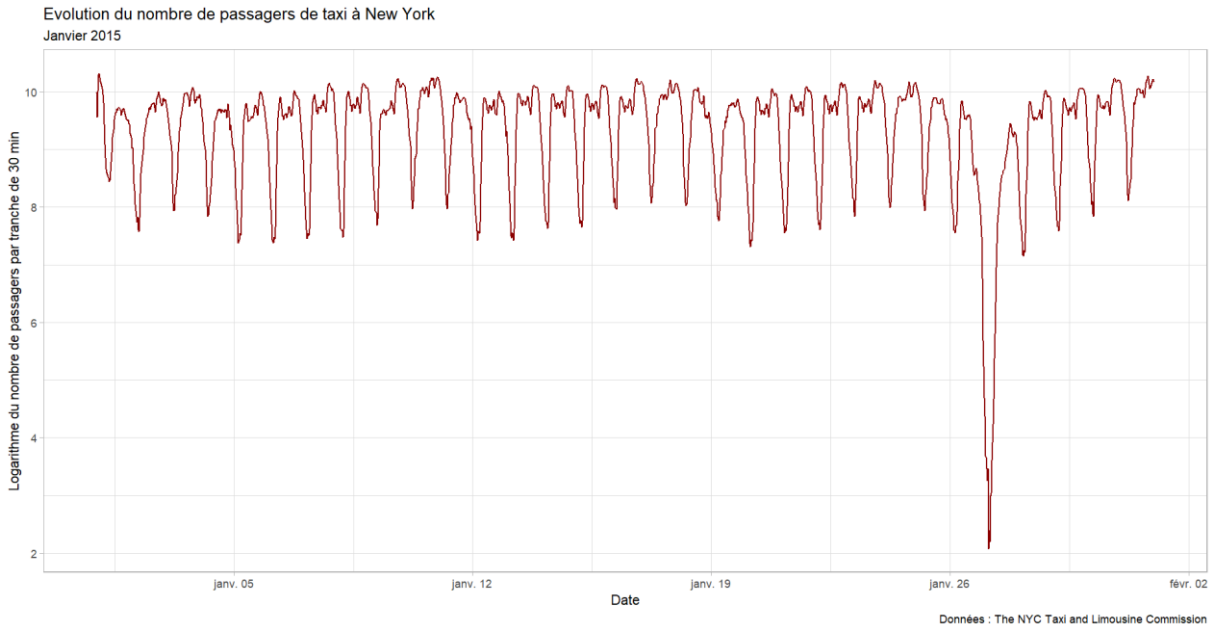

Figure 3 : Évolution logarithmique du nombre de passagers de taxi à New York au mois de janvier 2015

La figure 3 reprend la même forme que le graphique précédent mais en valeur logarithmique ce qui nous permet de mieux observer les variabilités en mettant en valeur les valeurs extrêmes. Ce graphique montre une saisonnalité mais pas de tendance. De plus, on observe nettement une valeur très faible autour du 27 janvier (Expliqué par la suite).

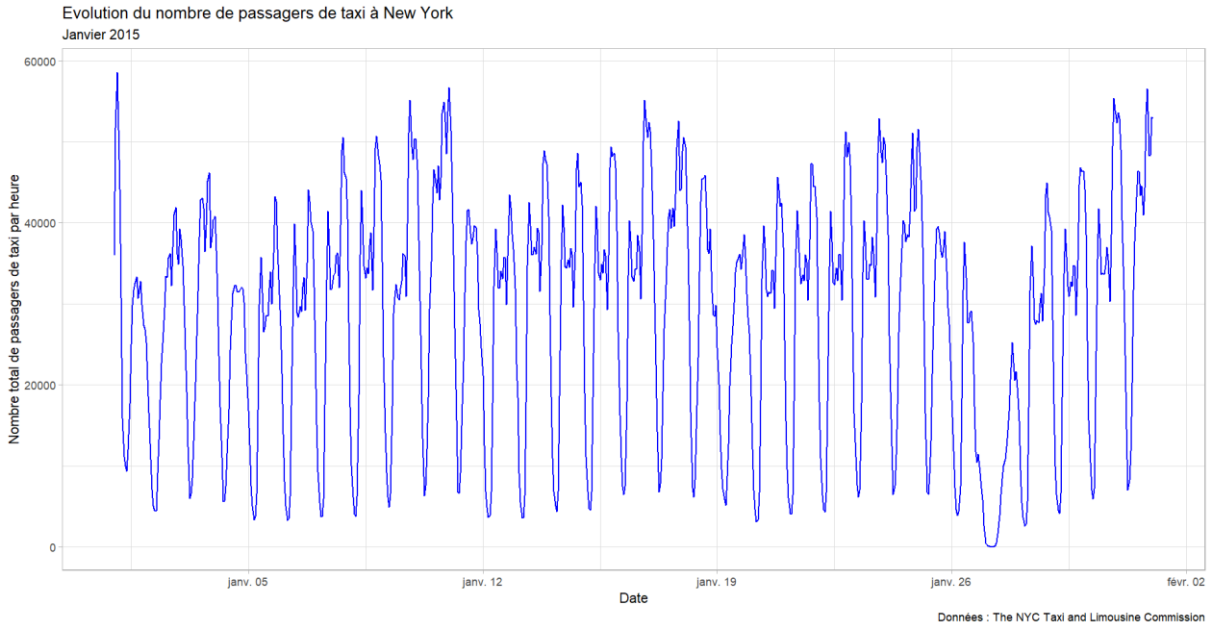

Figure 4 : Evolution du nombre de passagers de taxi à New York par tranche de 30 minutes

La figure 4 est, cette fois-ci, issue de l'agrégation des données par heure. On y observe le nombre de passagers sur le mois de janvier 2015. Une heure creuse typique se situe aux alentours des 5 000 passagers contre près de 55 000 passagers pour les heures de grande affluence.

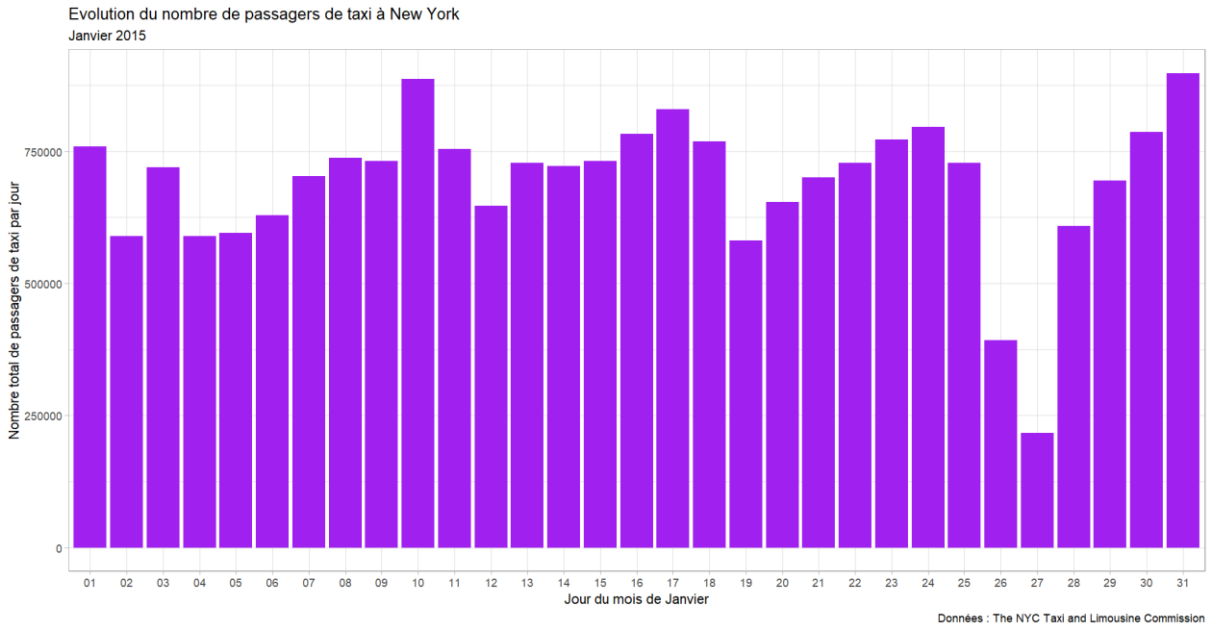

Figure 5 : Evolution du nombre de passagers de taxi à New York par jour

La figure 5 nous montre l'évolution journalière du nombre de passagers. On constate une tendance à la hausse sur 6 jours puis une légère baisse sur le 7ème jour.

Sur les 4 derniers graphiques, nous avons pu observer une valeur qui semble être aberrante autour du 26, 27 et 28 janvier, il s'agit en réalité d'une chute du nombre de passagers de taxi à cause du blizzard qui a eu lieu à cette date :

https://fr.wikipedia.org/wiki/Blizzard_de_janvier_2015_dans_le_Nord-Est_de_l'Am%C3%A9rique_du_Nord

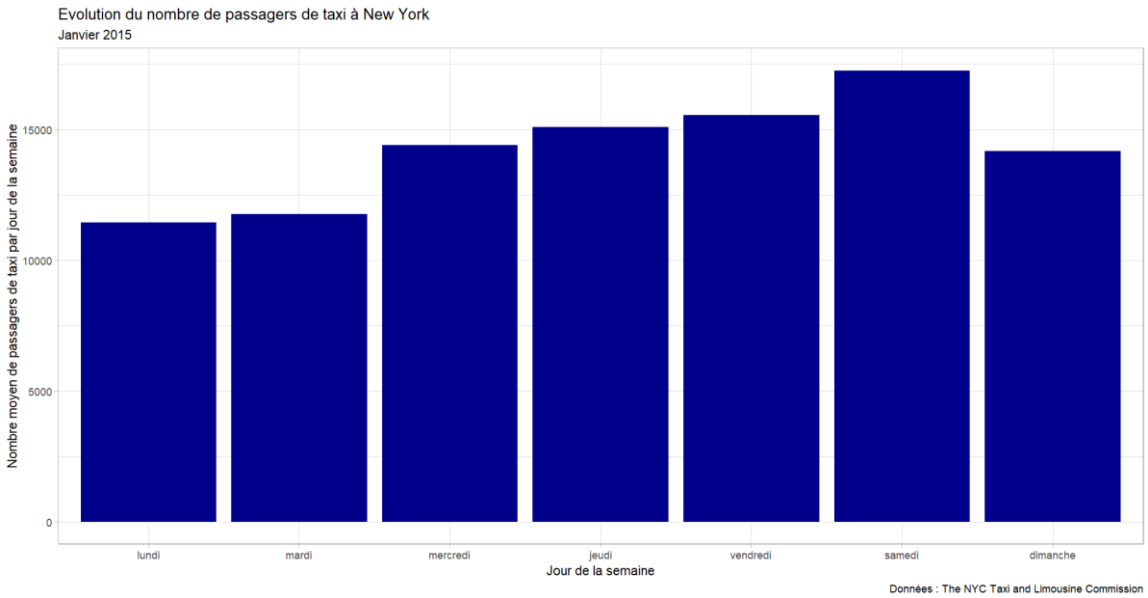

Figure 6 : Evolution du nombre de passagers de taxi à New York par jour de la semaine

Le graphique ci-dessus nous montre une demande de taxi croissante tout au long de la semaine. Le dimanche vient rééquilibrer cette demande avec une valeur plus faible.

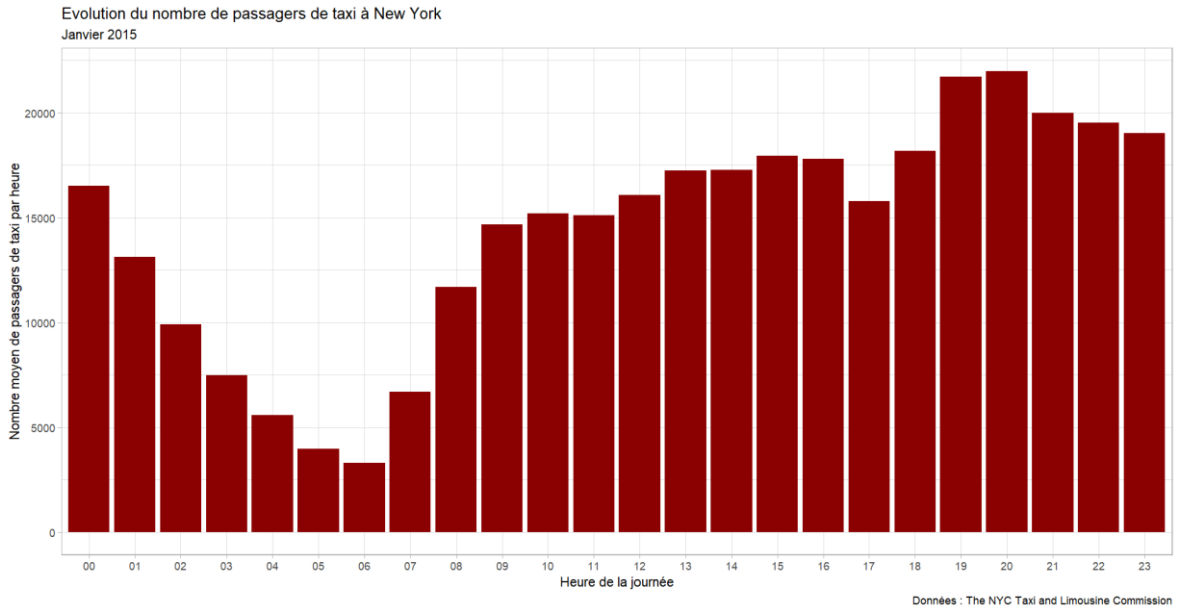

Figure 7 : Evolution du nombre de passagers de taxi à New York par heure de la journée

La figure 7 nous montre l'évolution du nombre de passagers de taxi à New York en fonction des heures de la journée. On peut observer qu'il y a une forte chute de nombre des passagers à partir de minuit jusqu'à 6h, et puis une augmentation des utilisateurs de taxi entre 7h et 16h.

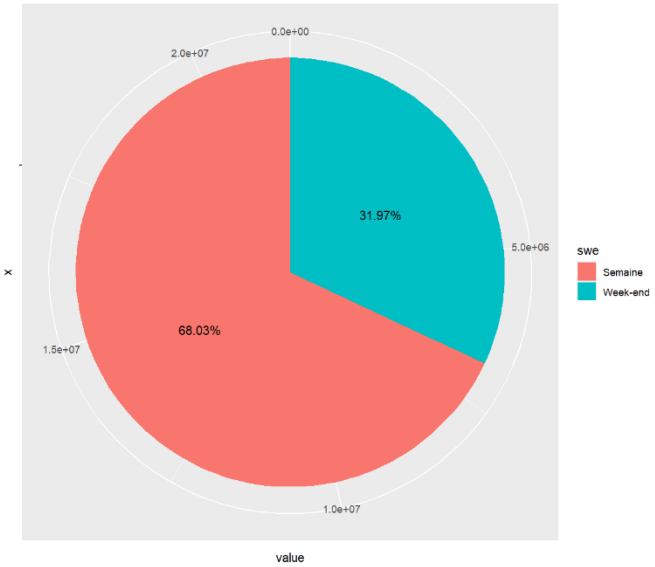

Figure 8 : Répartition du nombre de passagers entre la semaine et le week-end au cours du mois de janvier 2015

Le week-end ne représente que 28.5% (2/7) de la durée de la semaine mais ce diagramme circulaire nous montre que 32% du nombre de passagers observé ont pris le taxi le week-end.

Cela montre qu'en dépit du fait qu'il y ait moins de travailleurs le week-end, il y a tout de même plus de clients qui prennent le taxi.

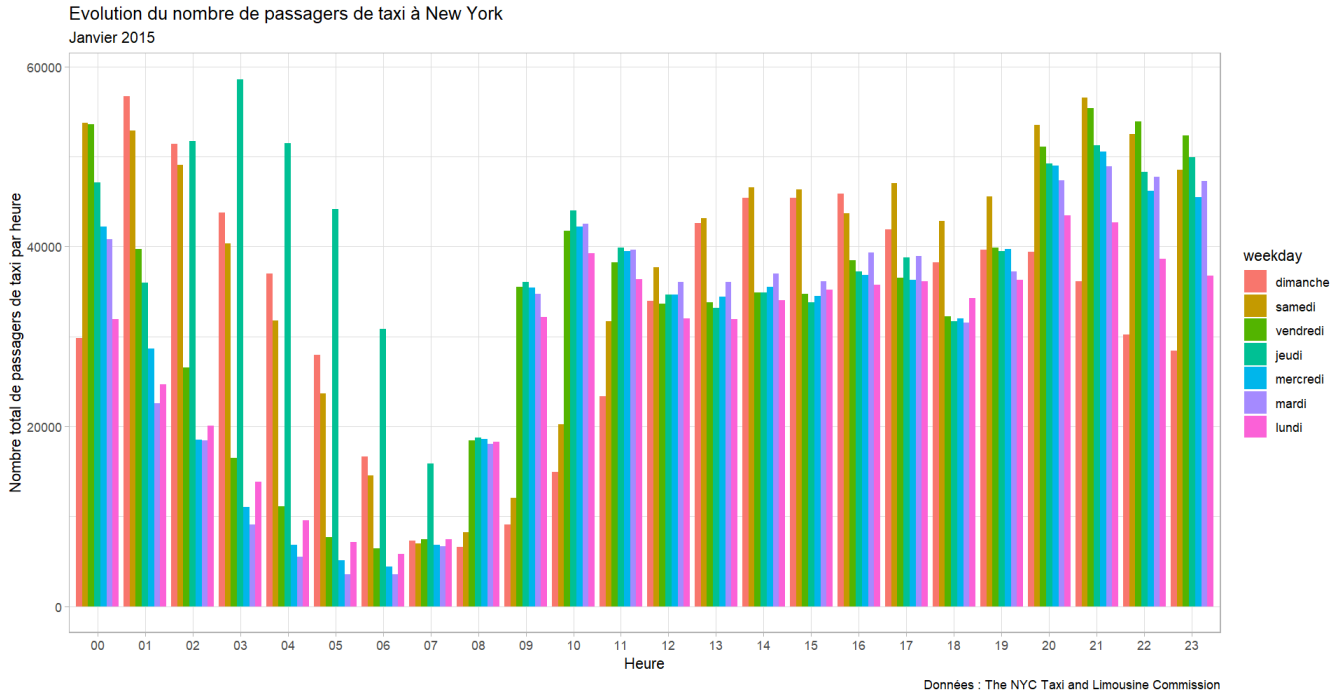

Figure 9 : Evolution du nombre de passagers de taxi à New York par heure de la journée (en fonction des jours de la semaine)

La figure 9 nous permet de constater qu'il y a des anomalies le jeudi entre 2h et 6h. En effet sur ce jour et cette tranche horaire, le nombre de passagers est anormalement haut.

Ces valeurs élevées sont dues au nouvel an 2015 qui a eu lieu un jeudi et qui a fait augmenter de façon exceptionnel le nombre de passagers de taxi après minuit.

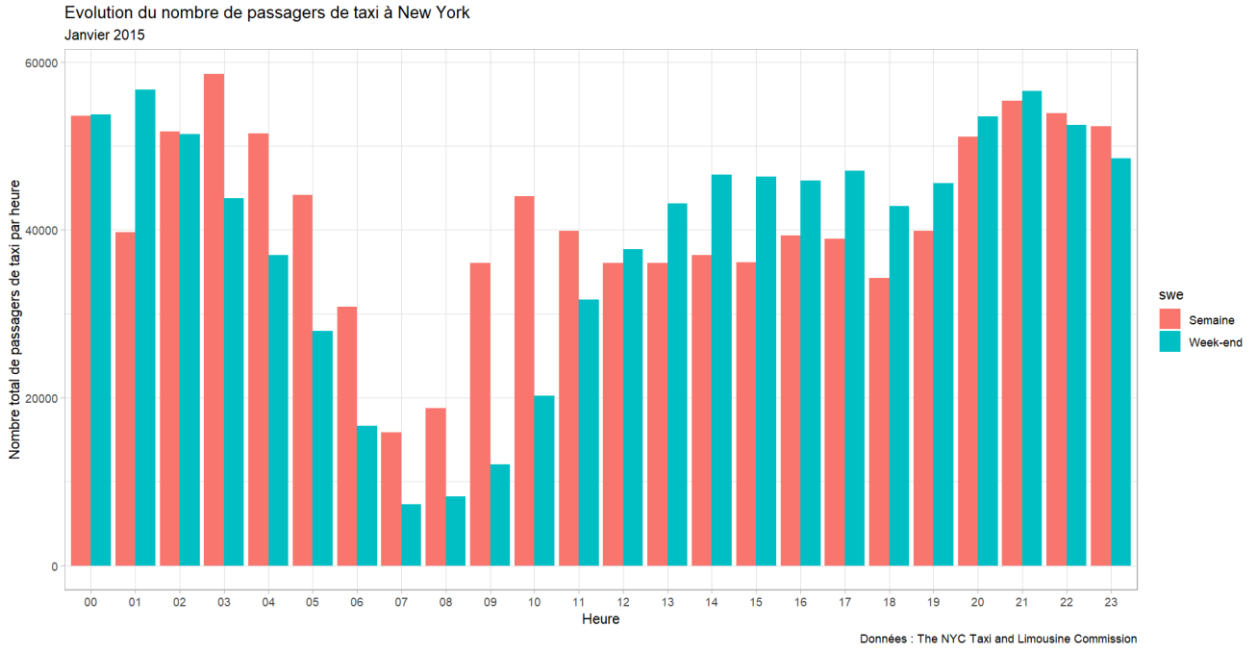

Figure 10 : Evolution du nombre de passagers de taxi à New York par heure de la journée (la semaine et le week-end)

A partir de la figure 10, nous constatons que le nombre de passagers de taxis est moins important le matin en période de week-end (car moins de personnes qui travaillent le week-end).

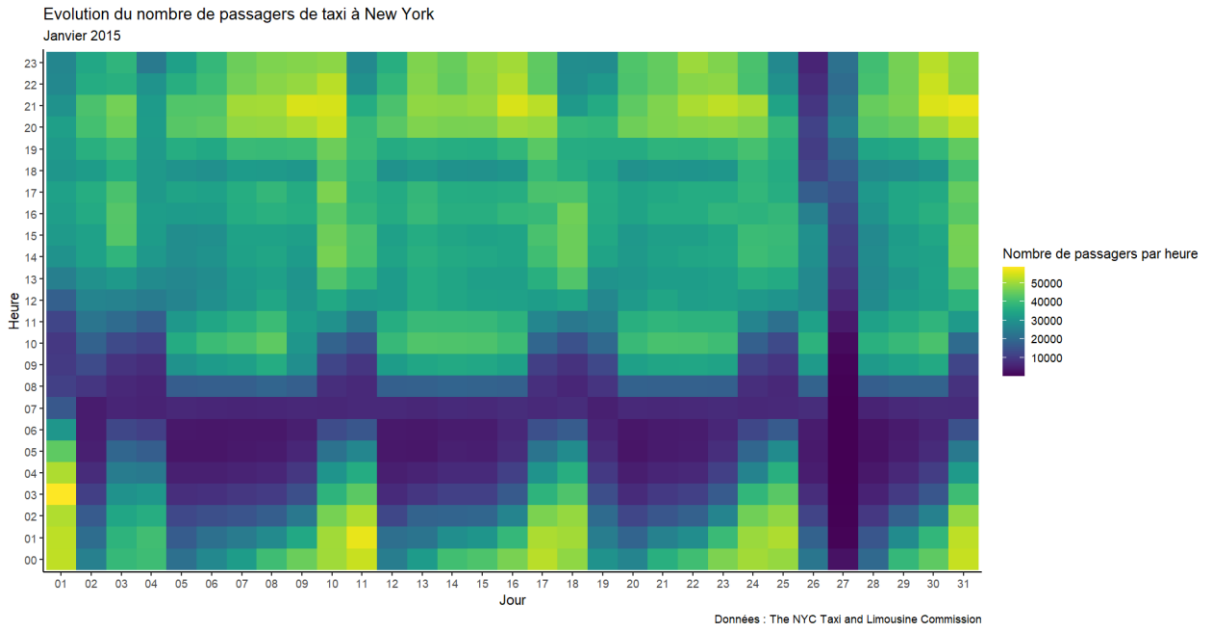

Figure 11 : Carte de chaleur du nombre de passagers par heure en fonction du jour du mois et de l'heure de la journée

A partir de la figure 11, on observe que la même tendance se répète durant les 3 premières semaines pleines (entre le 5 et le 25 du mois). Nous remarquons donc que les habitudes des utilisateurs de taxi sont presque les mêmes chaque semaine.

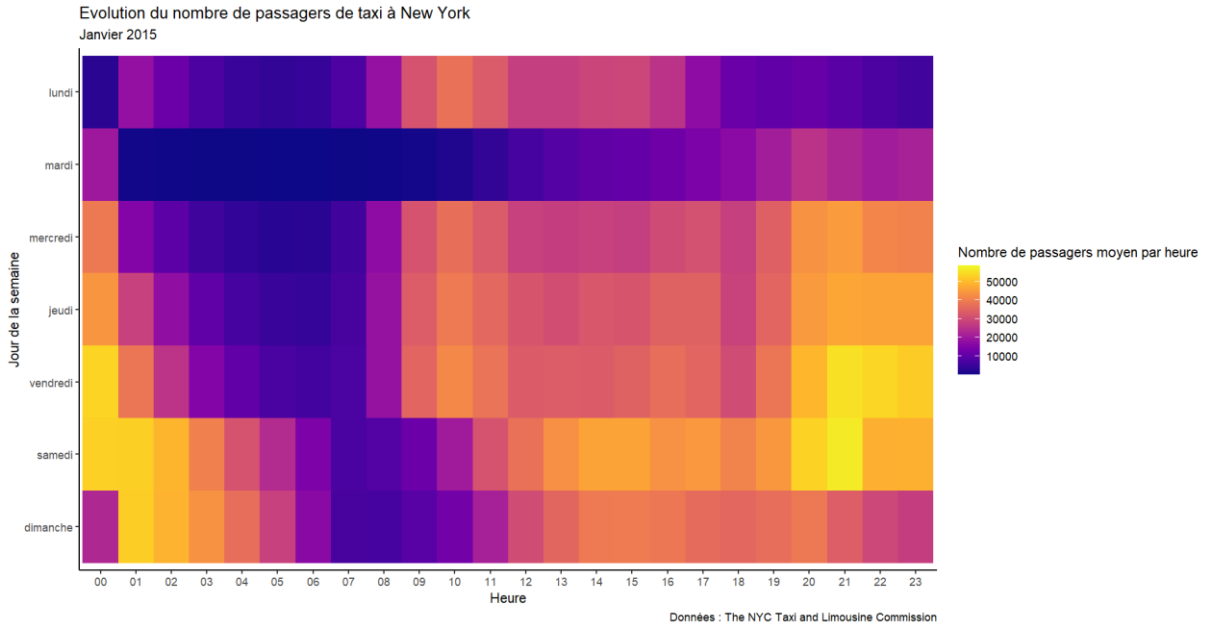

Figure 12 : Carte de chaleur du nombre moyen de passager par heure en fonction de l'heure de la journée et du jour de la semaine

La figure 12 nous montre le décalage dans les horaires de trajet entre le week-end et la semaine. Le nombre d'utilisateurs de taxis à New York a donc tendance à être plus tardif la nuit.

II / Moyenne mobile

Suppression / Remplacement des anomalies

Comme précisé précédemment, nous avons détecté deux anomalies :

- La période du nouvel an avec une forte augmentation des taxis entre minuit et 10h
- La période du blizzard qui a eu lieu du 26 au 28 janvier.

Nous avons donc décidé de remplacer ces périodes par des périodes correspondant aux mêmes jours de la semaine et mêmes horaires mais qui ont eu lieu des semaines différentes.

On remplace donc les valeurs des 10 premières heures du Jeudi 01 janvier 2015 par les valeurs des 10 premières heures du Jeudi 08 janvier 2015. De la même façon, nous remplaçons les

valeurs de la période allant du 26 au 28 janvier par les valeurs de la période allant du 19 au 21 janvier (d'une semaine plus tôt).

Choix du modèle

Dans cette partie, nous déterminerons si la série est additive ou multiplicative en utilisant trois méthodes différentes : la méthode de la bande, la méthode du profil et le test de Buys-Ballot. Nous expliquerons chacune de ces méthodes et discuterons des résultats obtenus. Nous examinerons également la fiabilité des résultats obtenus.

1ère méthode : Méthode de la bande

Méthodologie :

Cette méthode consiste à tracer les points de données sur un graphique et à déterminer si la courbe obtenue est une droite ou une courbe. Si la série suit une droite, cela signifie qu'elle est additive, et si elle est en forme d'entonnoir, cela signifie qu'elle est multiplicative.

Nous obtenons donc le graphique suivant :

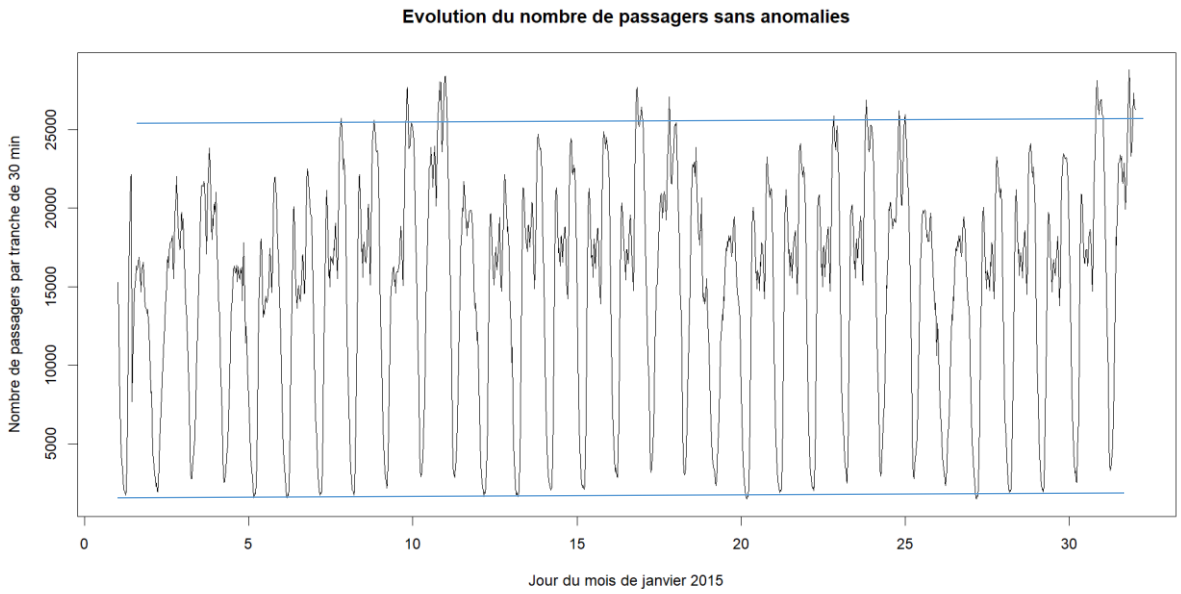

Figure 13 : Evolution du nombre de passager par tranche de 30 minutes

Conclusion :

On voit que les bandes sont assez droites même si la valeur des pics n'est pas uniforme. On peut conclure de part cette méthode à un modèle additif.

2ème méthode : Méthode du profil

Méthodologie :

Ici, nous allons superposer les différentes périodes qui sont au nombre de 31 (période journalière) pour déterminer si les profils se superposent.

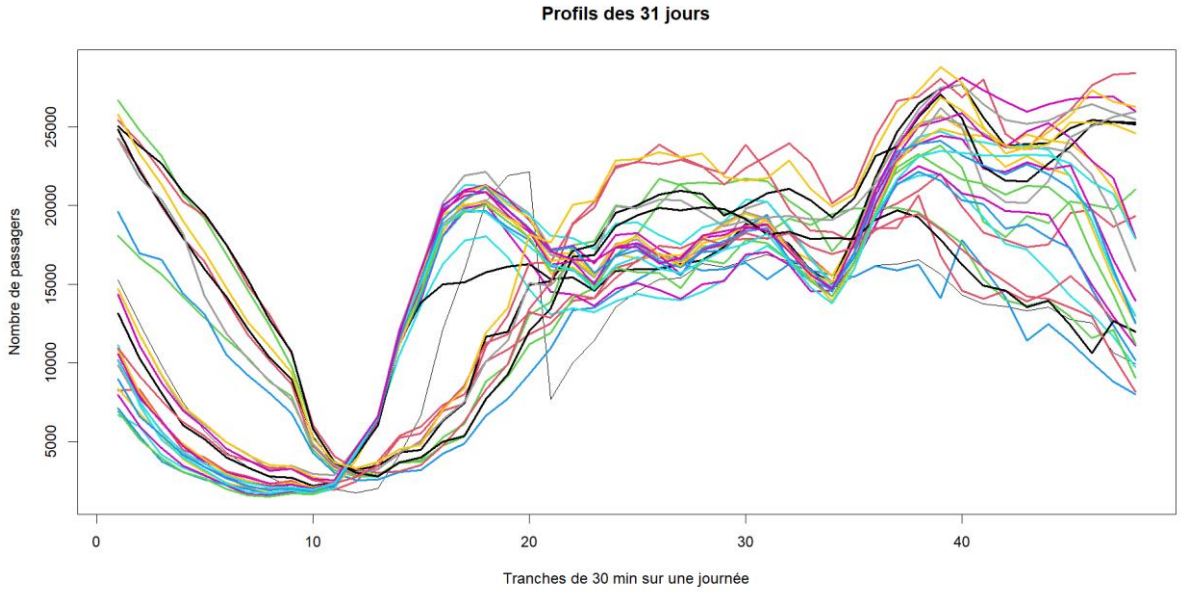

Figure 14 : Profils des 31 jours du mois

Ci-dessus, les profils s'entrecroisent et donc on peut conclure à un modèle multiplicatif.

3ème méthode : Test de Buys-Ballot

Méthodologie :

Cette méthode repose sur le principe selon lequel la série est multiplicative si les écarts-types et les moyennes des valeurs des 31 jours observés suivent une tendance linéaire. Si ce n'est pas le cas, cela signifie que la série est additive.

Test de Buys-Ballot:

P-value: 3.977e-06 < 0.05

Par le biais du test de Buys-Ballot, nous constatons qu'il y a bien une tendance linéaire. Ainsi on peut conclure que cette série est un modèle multiplicatif.

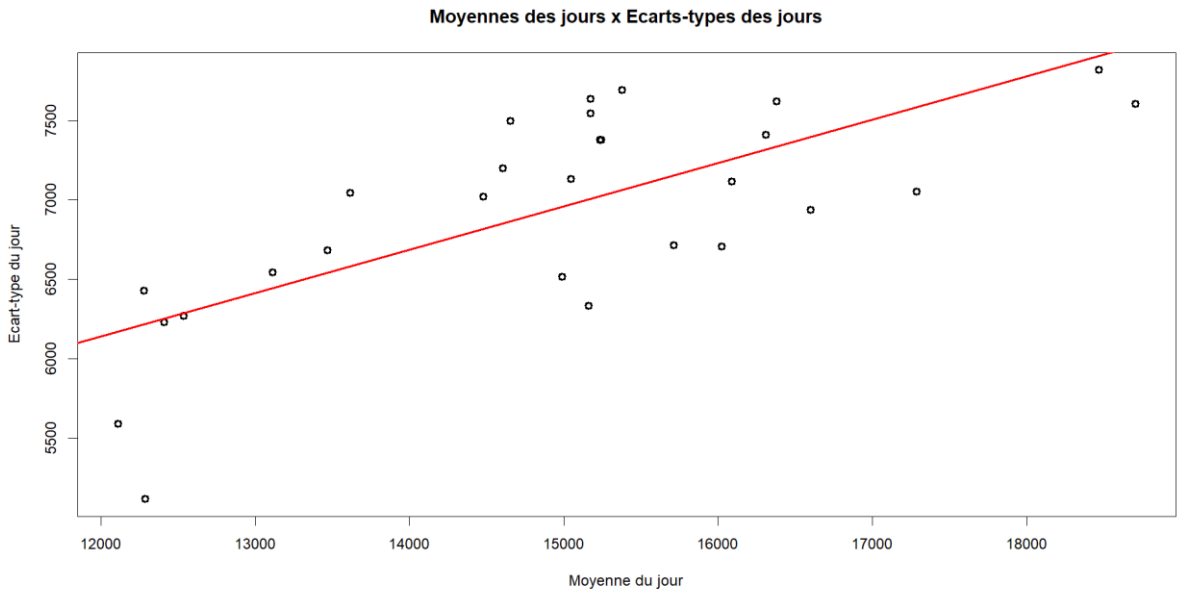

Figure 15 : Écart-type du nombre de passagers en fonction de la moyenne du jour.

Transformation de BOX-COX

Étant donné que l'on a un modèle multiplicatif, nous allons transformer les données en passant par la transformation de BOX-COX afin d'obtenir un modèle additif. Les données transformées sont représentées ci-dessous (Figure 16).

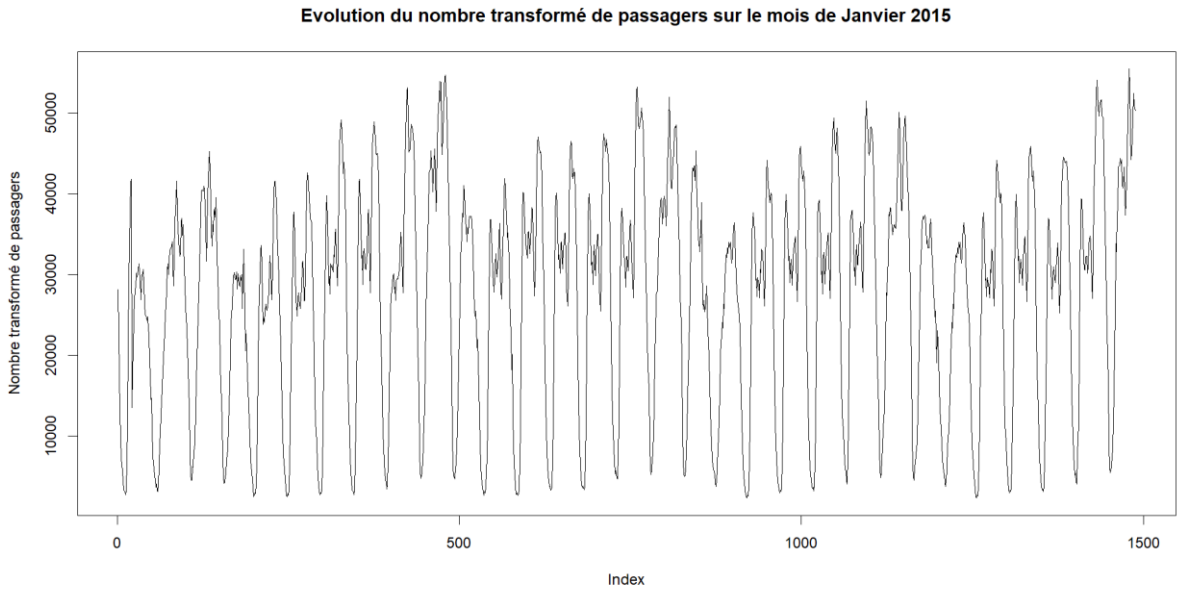

Figure 16 : Evolution du nombre transformé de passagers au cours du mois

Moyenne mobile appliquée

On applique maintenant une moyenne mobile d'ordre 49 pour les 48 tranches de 30 minutes des 31 jours :

filter48=c(1/96,rep(1/48,47),1/96)

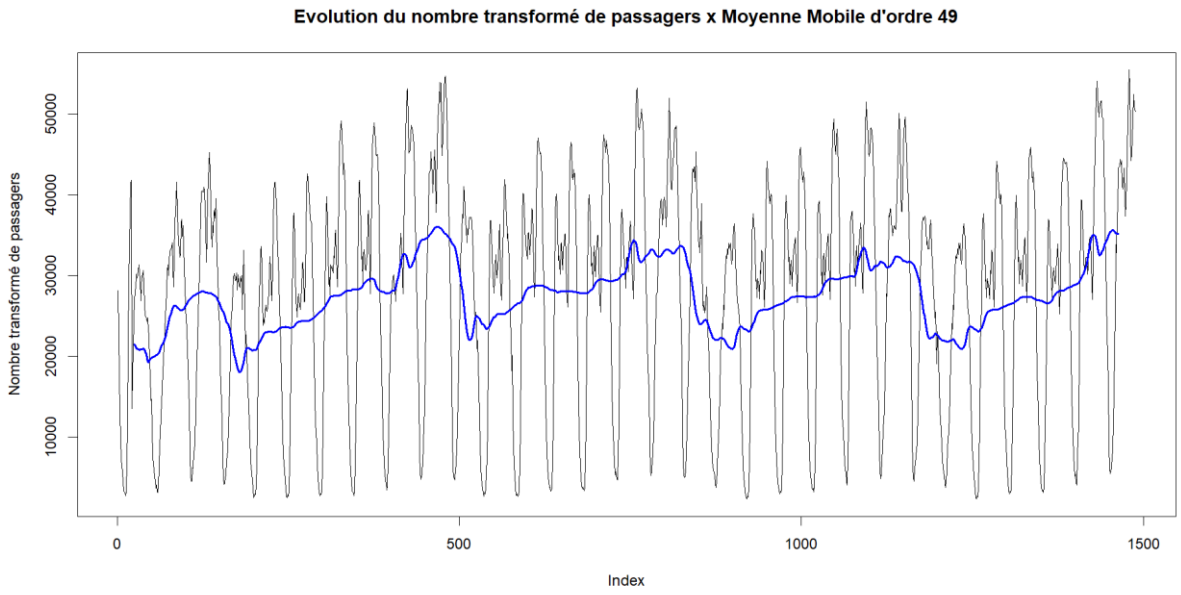

Figure 17 : Evolution du nombre transformé de passagers au cours du mois x Moyenne mobile d'ordre 49

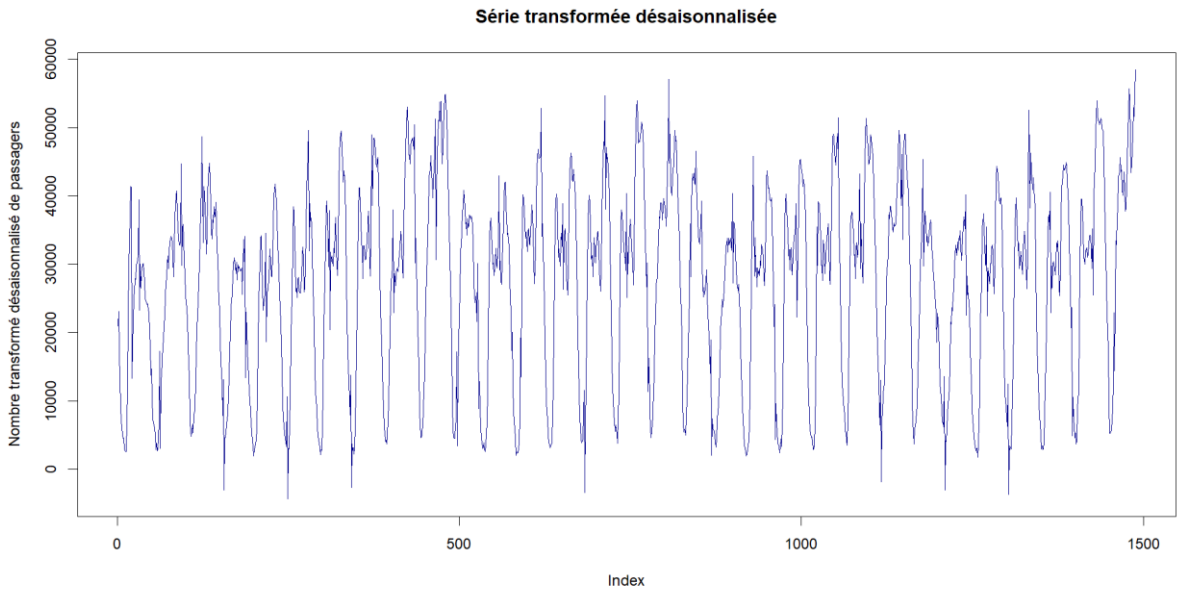

Figure 18 : Evolution du nombre désaisonnalisé de passagers au cours du mois

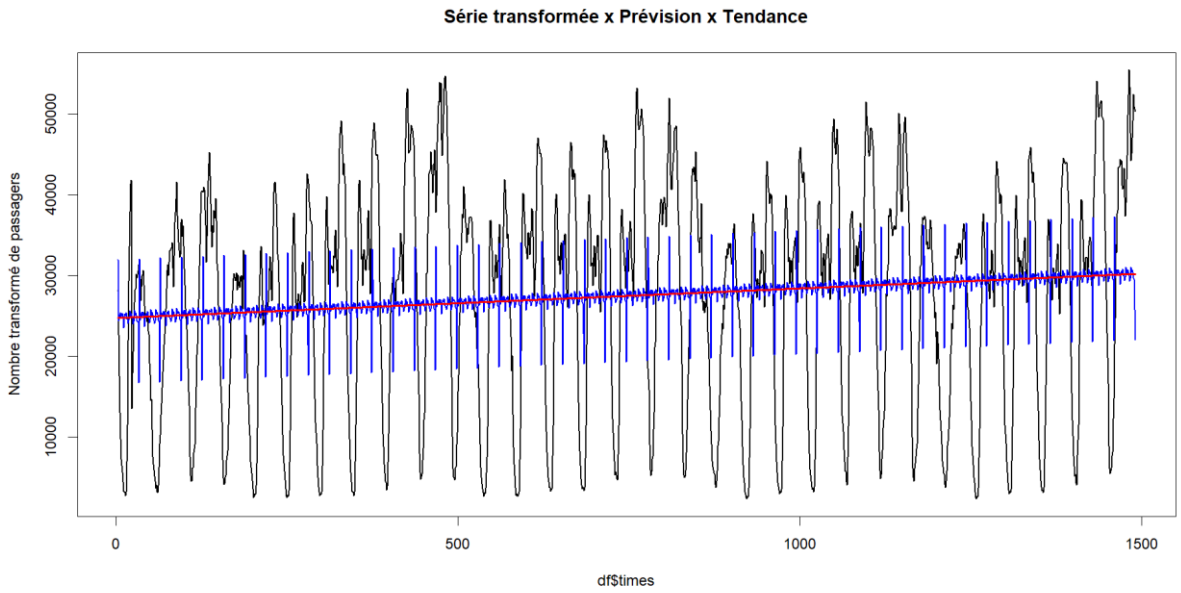

Figure 19 : Série transformée, sa prévision et sa tendance

Observons maintenant notre série estimée réalisée à partir de la moyenne mobile sur les données non-transformées (Figure 20).

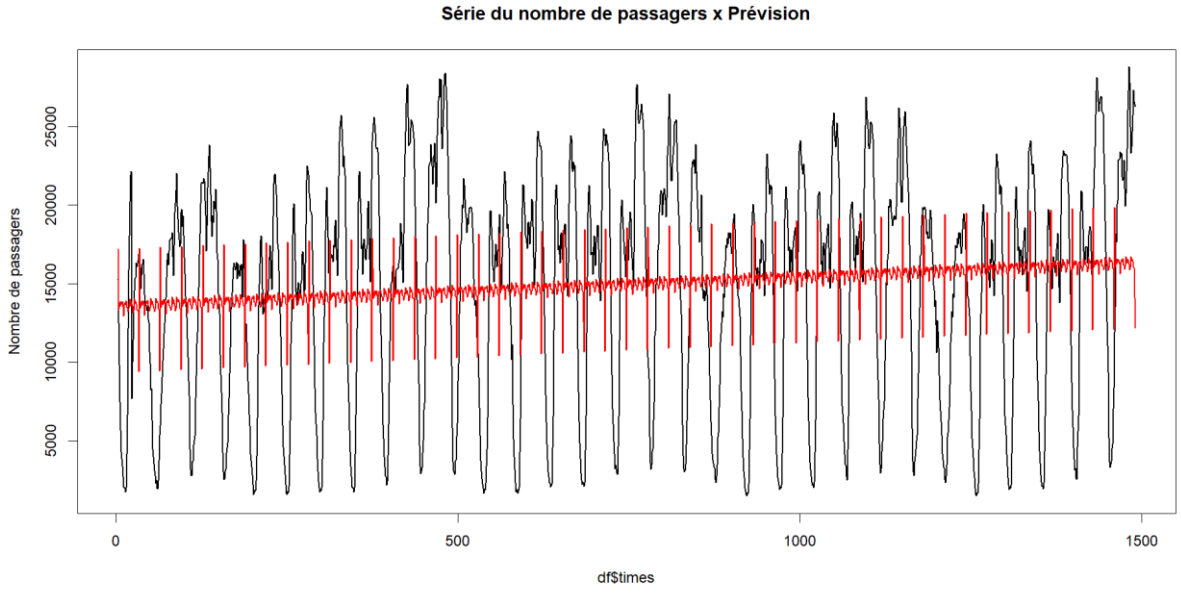

Figure 20 : Evolution du nombre de passagers et sa prévision

III / Lissages

Notre série a une tendance et une saisonnalité. Ainsi, il nous faut procéder à un lissage de Holt-Winters dans les deux modèles possibles (additif et multiplicatif). Lorsque l'on observe la différence de la somme des carrés des résidus entre le modèle additif et multiplicatif, on voit que la différence est négative. Donc le lissage du modèle additif est le meilleur. Voici le graphique obtenu pour le lissage additif :

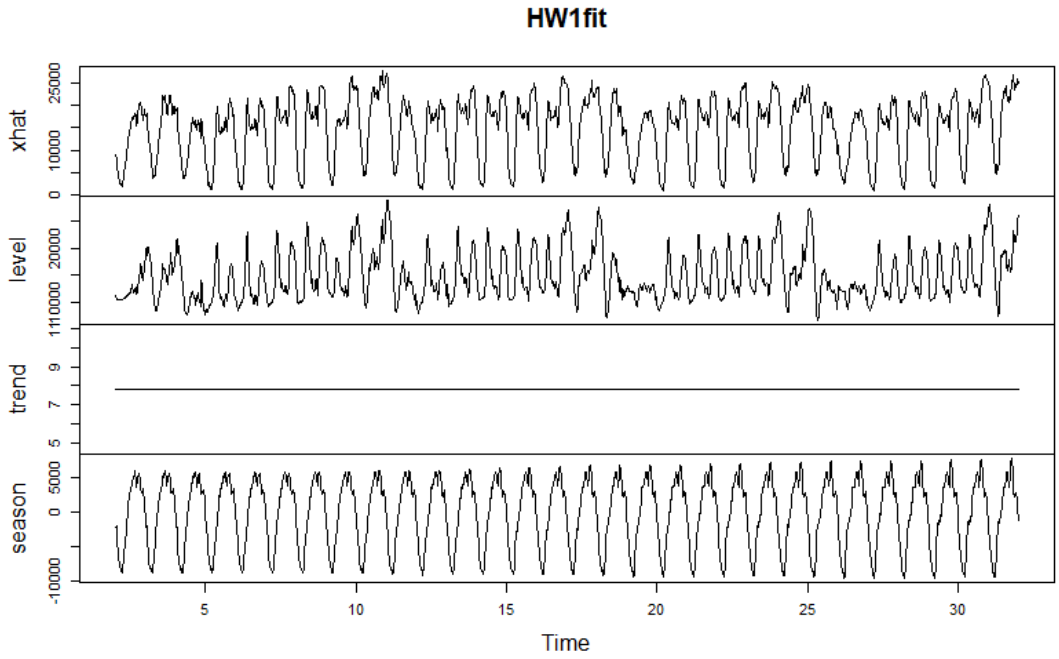

Figure 21 : Tendance, saisons et prévision du lissage Holt-Winters additif

Ce graphique ci-dessus possède quelques défauts visibles. En effet, les motifs de la partie 'season', indiquant les saisonnalités, ne sont pas tous identiques. Ceux de la fin sont différents de ceux du début. Or, voici le graphique obtenu avec le lissage multiplicatif :

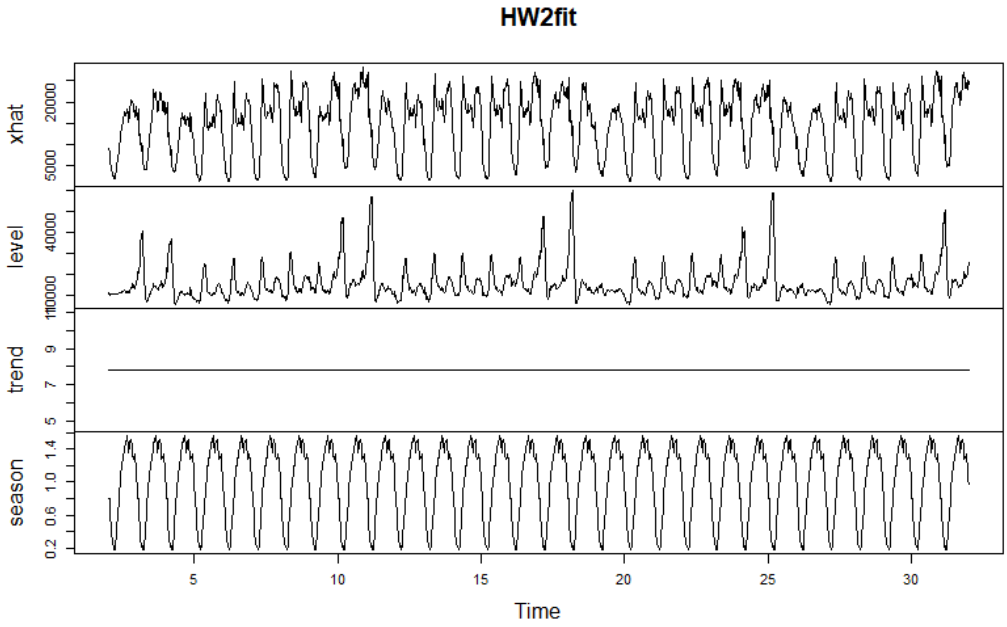

Figure 22 : Tendance, saisons et prévision du lissage Holt-Winters multiplicatif

On remarque clairement que ci-dessus les saisonnalités sont parfaitement identiques et donc que les schémas du nombre de passagers sont donc clairement identifiés.

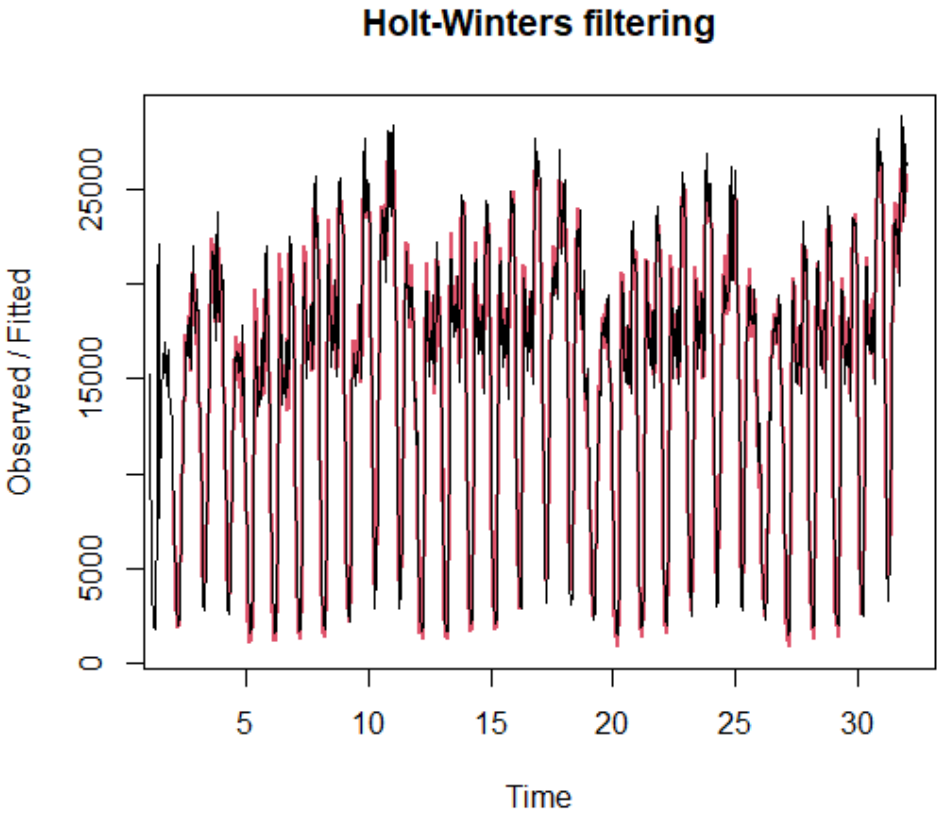

Figure 23 : Filtre Holt-Winters additif

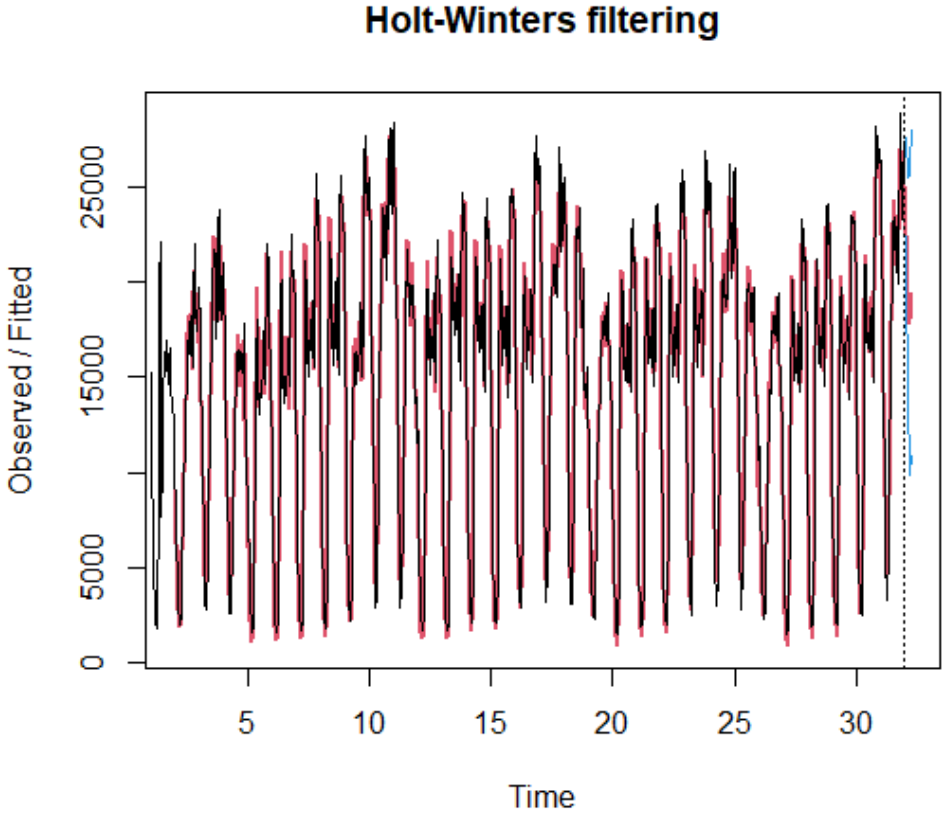

Figure 24 : Filtre Holt-Winters additif et sa prévision

Conclusion

Au final, notre analyse des données sur les taxis de New York a montré que les heures de pointe pour les passagers se produisent principalement le soir, avec un pic en fin de semaine.

Nous avons utilisé une moyenne mobile pour annuler l'effet de saisonnalité et une technique de lissage pour prédire les tendances futures. Les résultats ont montré que ces méthodes étaient efficaces pour annuler l'effet de saisonnalité et pour prévoir les tendances à venir.

Nos résultats ont permis de mieux comprendre les schémas d'utilisation des taxis new-yorkais et permettraient de prédire la demande future. Grâce à ces méthodes, il serait donc possible de prédire et de prendre des décisions qui aiderait par exemple à l'amélioration de la circulation pendant ces horaires.

Il est important de souligner que cette étude comporte des limites, notamment en ce qui concerne les biais potentiels dans les données. Le fait qu'il ait fallu exclure des anomalies telles que les épisodes de froid et la nuit du nouvel an a eu une incidence.

Annexe : Programme R

```r
## L3 MIASHS - Année 2022-2023
## Université de Rennes I & II
## Projet AS - Le trafic de taxis à New York au mois de janvier 2015

#########################################################################################

## Répertoire de travail
setwd("C:/Users/lebre/OneDrive/Bureau/AS/Projet AS")
getwd()

## Packages
library(lubridate)
library(ggplot2)
library(viridis)
library(timeDate)
library(car)


#########################################################################################
# I-Analyse descriptive ################################################################
#########################################################################################

# Importation des données
df=read.csv(file = "new_york_taxis_2014-2015.csv",sep = ",",row.names=1)
nrow(df)
df_complet = df

# conversion de timestamp en date + heure
df$timestamp <- ymd_hms(as.character(df$timestamp))

df <- df[(df$timestamp>"2015-01-01 00:30:00"),]
df$times=seq(1,nrow(df))
nrow(df)
summary(df)

df_complet$moisannee <- strftime(df_complet$timestamp, "%y-%m")

### Graphiques

# graphique des données complètes par mois
ggplot(df_complet, aes(x=moisannee,y=value)) +
  geom_bar(stat = "identity",fill="navy") +
  theme_light()+
# légende
  labs(title = "Evolution du nombre de passagers de taxi à New York",
       subtitle = "2014-2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Année et mois", y = "Nombre total de passagers de taxi par mois")

```

```r
# graphique des données brutes
ggplot(df, aes(x=timestamp, y=value)) +
  geom_line(size=0.5,color="darkblue") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Date", y = "Nombre total de passagers de taxi par tranche de 30 min")

# Régression linéaire : Nombre de passagers x Temps
reg <- lm(df$value~df$timestamp)
summary(reg) # p-value = 0.75 et R2=0 donc il n'y a pas de tendance linéaire notable au cours du mois

# graphique des données brutes en log
ggplot(df, aes(x=timestamp, y=log(value))) +
  geom_line(size=0.5,color="darkred") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Date", y = "Logarithme du nombre de passagers par tranche de 30 min")

# création d'une colonne jour
df$jour <- strftime(df$timestamp, "%d")

# création d'une colonne jour de la semaine (weekday)
df$weekday <- weekdays(df$timestamp)
df$weekday <- factor(df$weekday, levels = c('lundi', 'mardi', 'mercredi', 'jeudi','vendredi','samedi','dimanche'))

# création d'une colonne Semaine/week-end
df$swe <-isweekend(df$timestamp)
df$swe <- factor(df$swe)
levels(df$swe)[levels(df$swe)==TRUE] <- "Week-end"
levels(df$swe)[levels(df$swe)==FALSE] <- "Semaine"

# création d'une colonne heure
df$heure <- strftime(df$timestamp, "%H")
# création d'une colonne jour + heure
df$jour_heure <- strftime(df$timestamp, "%Y-%m-%d %H")
```

```r
# agrégation des données par jour de la semaine (weekday) (moyenne)
df_weekday <- aggregate(value ~ weekday,df,FUN = mean)
# agrégation des données par semaine/week-end (sum)
df_swe <- aggregate(value ~ swe,df,FUN = sum)
# agrégation des données par heure (moyenne)
df_heure <- aggregate(value ~ heure,df,FUN = mean)
# agrégation des données par jour + heure (somme) et ajout de colonnes intéressantes pour la suite
df_jour_heure <- aggregate(value ~ jour_heure,df,FUN = sum)
df_jour_heure$jour_heure <- ymd_h(as.character(df_jour_heure$jour_heure))
df_jour_heure$heure <- strftime(df_jour_heure$jour_heure, "%H")
df_jour_heure$jour <- strftime(df_jour_heure$jour_heure, "%d")
df_jour_heure$weekday <- weekdays(df_jour_heure$jour_heure)
df_jour_heure$weekday <- factor(df_jour_heure$weekday, levels =c('dimanche','samedi','vendredi',
                                                                 'jeudi','mercredi','mardi','lundi'))
df_jour_heure$swe <-isweekend(df_jour_heure$jour_heure)
df_jour_heure$swe <- factor(df_jour_heure$swe)
levels(df_jour_heure$swe)[levels(df_jour_heure$swe)==TRUE] <- "Week-end"
levels(df_jour_heure$swe)[levels(df_jour_heure$swe)==FALSE] <- "Semaine"
summary(df_jour_heure)

# graphique des données par jour + heure
ggplot(df_jour_heure, aes(x=jour_heure, y=value)) +
  geom_line(size=0.7,color="blue") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Date", y = "Nombre total de passagers de taxi par heure")

# graphique des données par jour
ggplot(df, aes(x=jour,y=value,group=jour)) +
  geom_bar(stat = "identity",fill="purple") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Jour du mois de Janvier", y = "Nombre total de passagers de taxi par jour")

# graphique des données par heure
ggplot(df_weekday, aes(x=weekday, y=value)) +
  geom_bar(stat = "identity",fill="darkblue") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Jour de la semaine", y = "Nombre moyen de passagers de taxi par jour de la semaine")
```

```r
# graphique des données par heure
ggplot(df_heure, aes(x=heure, y=value)) +
  geom_bar(stat = "identity",fill="darkred") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Heure de la journée", y = "Nombre moyen de passagers de taxi par heure")


# diagramme circulaire week-end/Semaine
df_swe$pourcentage=round(df_swe$value/sum(df_swe$value)*100,2)
ggplot(df_swe,aes(x="", y=value, fill=swe)) +
  geom_bar(stat="identity", width=1)+
  geom_text(aes(label = paste(pourcentage,"%",sep ="")),
            position = position_stack(vjust = 0.5)) +
  coord_polar("y", start=0)

# graphique des données par heure + jour de la semaine
ggplot(df_jour_heure, aes(x=heure,y=value,group=weekday,fill=weekday)) +
  geom_bar(stat = "identity",position="dodge") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York ",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Heure", y = "Nombre total de passagers de taxi par heure")

# graphique des données par heure + semaine ou week-end
ggplot(df_jour_heure, aes(x=heure,y=value,group=swe,fill=swe)) +
  geom_bar(stat = "identity",position="dodge") +
  theme_light()+
  # légende
  labs(title = "Evolution du nombre de passagers de taxi à New York",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Heure", y = "Nombre total de passagers de taxi par heure")

# carte de chaleur : Jour x Heure
ggplot(df_jour_heure,aes(jour,heure,fill=value)) +
  geom_tile()+
  theme_classic() +
  scale_fill_viridis(name="Nombre de passagers par heure",option ="viridis") +
  labs(title = "Evolution du nombre de passagers de taxi à New York",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Jour", y = "Heure")
```

```r
# carte de chaleur : Weekday x Jour
ggplot(df_jour_heure,aes(heure,weekday,fill=value)) +
  geom_tile()+
  theme_classic() +
  scale_fill_viridis(name="Nombre de passagers moyen par heure",option ="plasma") +
  labs(title = "Evolution du nombre de passagers de taxi à New York",
       subtitle = "Janvier 2015",
       caption = "Données : The NYC Taxi and Limousine Commission",
       x = "Heure", y = "Jour de la semaine")


##############################################################################################
# II-Moyennes mobiles ########################################################################
##############################################################################################

## Suppression des anomalies

# Le nouvel an : on le remplace par le jeudi suivant (les 10 premières heures)
prim8 <- which(df$jour=="08")[1]
df[1:20,]$value <- df[prim8:(prim8+19),]$value

# Le blizzard du 26, 27 et 28 janvier : on les remplace par les 19,20 et 21 du mois
prim26 <- which(df$jour=="26")[1]
prim19 <- which(df$jour=="19")[1]
df[prim26:(prim26+48*3),]$value <- df[prim19:(prim19+48*3),]$value

# création de la série temporelle
df_periode<- ts(data = df$value,start = 01+(1/48),end = 31+(48/48),frequency = 48)
length(df_periode)

## Méthode de la bande

# graphique de la série temporelle sans anomalies
plot(df_periode,type="l",main="Evolution du nombre de passagers sans anomalies",
     xlab="Jour du mois de janvier 2015",ylab="Nombre de passagers par tranche de 30 min")
# les droites ne sont pas paralleles donc il semble s'agir d'un modèle multiplicatif

## Méthode du profil

mat=matrix(data=df_periode,nrow=31,ncol=48,byrow=TRUE)
ymin=min(mat[1:31,])
ymax=max(mat[1:31,])
plot(mat[1,],type="l",col=1,ylim=c(ymin,ymax),main="Profils des 31 jours",
     xlab="Tranches de 30 min sur une journée",ylab="Nombre de passagers")
for(i in 2:31) {lines(mat[i,],type="l",col=i,lwd=2)}
# les courbes ne se superposent pas vraiment donc il semble s'agir d'un modèle multiplicatif
```

```r
## Test de Buys-Ballot pour déterminer le modèle

# calcul des moyenne par jour pour chaque observation
aggmean<- aggregate(df$value,list(jour=df$jour),mean)
# calcul des ecarts types par jour pour chaque observation
aggsd<-aggregate(df$value,list(jour=df$jour),FUN="sd")
# on effectue une regression lineaire
buys_ballot<- lm(aggsd$x~aggmean$x)
reg_bb <- summary(buys_ballot)
reg_bb
plot(aggsd$x~aggmean$x,main="Moyennes des jours x Ecarts-types des jours",xlab="Moyenne du jour",ylab="Ecart-type du jour",lwd=2)
abline(reg_bb$coefficients[1],reg_bb$coefficients[2],col='red',lwd=2)

# p-value < 0.05 donc relation linéaire effective
# On rejette l'hypothese nulle. Donc le modele est multiplicatif

## Transformation de la série mutliplicative : transformation de BOX-COX

lambda <- powerTransform(df$value)$lambda
df$mod <- ((df$value^lambda)-1)/lambda
# graphique de la série transformée
plot(df$mod,type="l",main="Evolution du nombre transformé de passagers sur le mois de Janvier 2015",
     ylab="Nombre transformé de passagers")

## MM  d'ordre 49

filter48=c(1/96,rep(1/48,47),1/96)
df$modf49=filter(df$mod, filter48, method = "convolution",sides = 2, circular = FALSE)

# graphique de la moyenne mobile d'odre 49
plot(df$mod,type='l',col='black',lwd=1,main="Evolution du nombre transformé de passagers x Moyenne Mobile d'ordre 49",
     ylab="Nombre transformé de passagers")
lines(df$modf49,type='l',col='blue',lwd=2)

# calcul des coefficients saisonniers
df$modsais0=df$mod-df$modf49
modsais0=tapply(df$modsais0,df$jour,mean,na.rm=TRUE)
sais0moy=mean(modsais0)
sais0bis=modsais0-sais0moy # coefficients saisonniers finals
df$sais0bis=rep(sais0bis,48) # ajout des coefficients saisonniers au dataframe

# extrapolation de la tendance
df$desais=df$mod-df$sais0bis
# graphique de la série désaisonnalisée
plot(df$desais,type='l',main="Série transformée désaisonnalisée",
     ylab="Nombre transformé désaisonnalisé de passagers",col='darkblue')

reg1=lm(df$desais~df$times)
summary(reg1) # p-value < 0.05 donc relation linéaire effective
# estimations de la tendance
df$tchap0=reg1$coefficients[1]+reg1$coefficients[2]*df$times
# previsions de la serie mod
df$prev0=df$tchap0+df$sais0bis
```

```r
# graphique de la série transformée x prévision x tendance
plot(df$mod~df$times,type='l',col='black',lwd=1.5,main="Série transformée x Prévision x Tendance",
     ylab="Nombre transformé de passagers")
lines(df$prev0~df$times,type='l',col='blue',lwd=1.5)
lines(df$tchap0~df$times,type='l',col='red',lwd=2.5)


# estimation et previsions de la serie df
df$prev1=(lambda*df$prev0+1)^(1/lambda)

# graphique de la série x prévision
plot(df$value~df$times,type='l',col='black',lwd=1.5,main="Série du nombre de passagers x Prévision",
     ylab="Nombre de passagers")
lines(df$prev1~df$times,type='l',col='red',lwd=1.5)


################################################################################################
# III-Lissages ################################################################################
################################################################################################

# Il y a une tendance et une saisonnalité donc on effectue un lissage de Holt-Winters

# Lissage de Holt-Winters sans declaration des coefficients de lissage, Modele additif
HW1=HoltWinters(df_periode)
HW1
plot(HW1,lwd=2,col="black")
HW1fit=fitted(HW1)
HW1fit
p1=predict(HW1,12,prediction.interval=TRUE)
plot(HW1,p1,lwd=2,col="black")


# Lissage de Holt-Winters sans declaration des coefficients de lissage, Modele multiplicatif
HW2=HoltWinters(df_periode,seasonal="multiplicative")
HW2
plot(HW2,lwd=2,col="black")
HW2fit=fitted(HW2)
HW2fit
p2=predict(HW2,12,prediction.interval=TRUE)
plot(HW2,p2,lwd=2,col="black")

plot(HW1fit)
plot(HW2fit)

dif= HW1$SSE-HW2$SSE
dif # La somme des carrés des résidus du 1er modèle est plus faible
# Donc le premier lissage (additif) est le meilleur
```