Projet_Capstone_Data

Caleb

2025-04-07

Entreprise : Cyclistic

Produit : Vélo en libre-service

Ville : Chicago

Parties prenantes : l'équipe d'analyse marketing, équipe de direction, cyclistes occasionnels, membres annuels.

L'entreprise crée en 2016, possède 5824 vélos géolocalisés et vérouillés dans un réseau de 692 stations de service dans Chicago

Type de vélos : • Vélos inclinables, • Tricycle à main, • Vélos cargo

Utilisateurs de vélos chez Cyclistics • Majorité optent pour des vélos traditionnels • 8% utilisent les options d'assistance • 30% utilisent le vélo pour se rendre au travail chaque jour

Stratégie actuelle pour son succès • Flexibilité des plans tarifaires o Passes pour un trajet o Passes pour une journée o Adhésions annuelles

Type de clients : • Les cyclistes occasionnels sont : les clients qui achètent des passes trajet et pour une journée. • Les membre de Cyclistic : les clients qui achètent des adhésions annuelles

Contexte du projet : Les experts ont déterminé que les adhérents réguliers sont beaucoup plus rentables que les cyclistes occasionnels. La flexibilité des prix encourage un aspect à attirer davantage les clients. Ainsi, pour la directrice Moreno, l'optimisation du nombre d'adhérents annuels sera le facteur déterminant de l'expansion future, plutôt que la mise en place d'une campagne marketing visant à séduire de nouveaux clients. D'après la directrice, une possibilité réside dans la transformation d es cyclistes occasionnels en adhérents.

```
Il est à noter que les cyclistes occasionnels sont familiers avec
le programme Cycllitic et l'ont sélectionné pour répondre à leurs
exigences de mobilité.
```

1.*Énoncé clair de la tâche commerciale* : Déterminer les différences d'utilisation des vélos Cyclistic entre les membres annuels et les cyclistes occasionnels afin d'éclairer les stratégies marketing visant à convertir les cyclistes occasionnels en membres annuels. Plus précisément, il s'agit d'analyser les données de trajets pour identifier les tendances et les schémas d'utilisation distincts entre ces deux groupes.

Pour atteindre l'objectif voici les questions auxquelles nous devrions apporter des éléments de solution 1. En quoi les cyclistes occasionnels et les membres annuels utilisent-ils différemment les vélos Cyclistic ? 2. Pourquoi les cyclistes occasionnels achèteraient un abonnement ? 3. Comment Cyclistic peut-il utiliser les médias numériques pour inciter les cyclistes occasionnels à devenir membres ?

2. **Description des sources de données utilisées**: il s'agit de données historiques de Cyclistic sur les déplacements à vélo, contenant les informations suivantes: • Type d'utilisateur: (Membre annuel /

Cycliste occasionnel) • Date et heure de début et de fin du trajet: (Permettant de calculer la durée du trajet) • Station de départ et d'arrivée: (Permettant d'analyser les itinéraires et les zones populaires) • Type de vélo utilisé: (Vélo standard, vélo inclinable, tricycle à main…)

Installer les librairies

```
library(tidyverse)
library(dplyr)
library(skimr)
library(janitor)
library(ggplot2)
```

Importer les fichiers dans R

```
data_Q1_2019 <- read.csv("C:/Users/Utilisateur/Downloads/données_velo_certif_prof - Copie/Divvy_Trips_2
data_Q2_2019 <- read.csv("C:/Users/Utilisateur/Downloads/données_velo_certif_prof - Copie/Divvy_Trips_2
data_Q3_2019 <- read.csv("C:/Users/Utilisateur/Downloads/données_velo_certif_prof - Copie/Divvy_Trips_2
data_Q4_2019 <- read.csv("C:/Users/Utilisateur/Downloads/données_velo_certif_prof - Copie/Divvy_Trips_2
data_Q1_2020 <-read.csv("C:/Users/Utilisateur/Downloads/données_velo_certif_prof - Copie/Divvy_Trips_202
```

Nettoyage

Vérifier les noms des colonnes :

Résultats

```
##  [1] "trip_id"           "start_time"        "end_time"
##  [4] "bikeid"            "tripduration"      "from_station_id"
##  [7] "from_station_name" "to_station_id"     "to_station_name"
## [10] "usertype"          "gender"            "birthyear"

##  [1] "X01...Rental.Details.Rental.ID"
##  [2] "X01...Rental.Details.Local.Start.Time"
##  [3] "X01...Rental.Details.Local.End.Time"
##  [4] "X01...Rental.Details.Bike.ID"
##  [5] "X01...Rental.Details.Duration.In.Seconds.Uncapped"
##  [6] "X03...Rental.Start.Station.ID"
##  [7] "X03...Rental.Start.Station.Name"
##  [8] "X02...Rental.End.Station.ID"
##  [9] "X02...Rental.End.Station.Name"
## [10] "User.Type"
## [11] "Member.Gender"
## [12] "X05...Member.Details.Member.Birthday.Year"

##  [1] "trip_id"           "start_time"        "end_time"
##  [4] "bikeid"            "tripduration"      "from_station_id"
##  [7] "from_station_name" "to_station_id"     "to_station_name"
## [10] "usertype"          "gender"            "birthyear"
```

```
##  [1] "trip_id"           "start_time"        "end_time"
##  [4] "bikeid"            "tripduration"      "from_station_id"
##  [7] "from_station_name" "to_station_id"     "to_station_name"
## [10] "usertype"          "gender"            "birthyear"
```

```
##  [1] "ride_id"            "rideable_type"      "started_at"
##  [4] "ended_at"           "start_station_name" "start_station_id"
##  [7] "end_station_name"   "end_station_id"     "start_lat"
## [10] "start_lng"          "end_lat"            "end_lng"
## [13] "member_casual"
```

Les noms des variables pour le 2e trimestre 2019 diffèrent de ceux des autres timestres.

Modifiez les noms des colonnes du fichier 2e trimestre 2019 et du 1er trimestre 2020 pour faciliter la fusion avec les autres trimestres

code

```
colnames(data_Q1_2019)
colnames(data_Q2_2019)
colnames(data_Q3_2019)
colnames(data_Q4_2019)
colnames(data_Q1_2020)
```

Changer les noms de colonnes du 2e trimestre 2019 :

Résultats

```
## 'data.frame':    1108163 obs. of  12 variables:
##  $ trip_id          : int  22178529 22178530 22178531 22178532 22178533 22178534 22178535 22178536 22
##  $ start_time       : chr  "2019-04-01 00:02:22" "2019-04-01 00:03:02" "2019-04-01 00:11:07" "2019-04
##  $ end_time         : chr  "2019-04-01 00:09:48" "2019-04-01 00:20:30" "2019-04-01 00:15:19" "2019-04
##  $ bikeid           : int  6251 6226 5649 4151 3270 3123 6418 4513 3280 5534 ...
##  $ tripduration     : chr  "446.0" "1,048.0" "252.0" "357.0" ...
##  $ from_station_id  : int  81 317 283 26 202 420 503 260 211 211 ...
##  $ from_station_name: chr  "Daley Center Plaza" "Wood St & Taylor St" "LaSalle St & Jackson Blvd" "M
##  $ to_station_id    : int  56 59 174 133 129 426 500 499 211 211 ...
##  $ to_station_name  : chr  "Desplaines St & Kinzie St" "Wabash Ave & Roosevelt Rd" "Canal St & Madis
##  $ usertype         : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ gender           : chr  "Male" "Female" "Male" "Male" ...
##  $ birthyear        : int  1975 1984 1990 1993 1992 1999 1969 1991 NA NA ...
```

code

```
data_Q2_2019_v1 <- data_Q2_2019 %>%
  rename(trip_id = X01...Rental.Details.Rental.ID,
         start_time = X01...Rental.Details.Local.Start.Time,
         end_time = X01...Rental.Details.Local.End.Time,
         bikeid = X01...Rental.Details.Bike.ID,
```

```
        tripduration = X01...Rental.Details.Duration.In.Seconds.Uncapped,
        from_station_id = X03...Rental.Start.Station.ID,
        from_station_name = X03...Rental.Start.Station.Name,
        to_station_id = X02...Rental.End.Station.ID,
        to_station_name = X02...Rental.End.Station.Name,
        usertype = User.Type,
        gender = Member.Gender,
        birthyear = X05...Member.Details.Member.Birthday.Year)
```

Changer les noms de colonnes du 1er trimestre 2020 :

Résultats

```
## 'data.frame':    426887 obs. of  13 variables:
##  $ trip_id          : chr  "EACB19130B0CDA4A" "8FED874C809DC021" "789F3C21E472CA96" "C9A388DAC6ABF313
##  $ rideable_type    : chr  "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ start_time       : chr  "2020-01-21 20:06:59" "2020-01-30 14:22:39" "2020-01-09 19:29:26" "2020-0
##  $ end_time         : chr  "2020-01-21 20:14:30" "2020-01-30 14:26:22" "2020-01-09 19:32:17" "2020-0
##  $ from_station_name: chr  "Western Ave & Leland Ave" "Clark St & Montrose Ave" "Broadway & Belmont A
##  $ from_station_id  : int  239 234 296 51 66 212 96 96 212 38 ...
##  $ to_station_name  : chr  "Clark St & Leland Ave" "Southport Ave & Irving Park Rd" "Wilton Ave & Bel
##  $ to_station_id    : int  326 318 117 24 212 96 212 212 96 100 ...
##  $ start_lat        : num  42 42 41.9 41.9 41.9 ...
##  $ start_lng        : num  -87.7 -87.7 -87.6 -87.6 -87.6 ...
##  $ end_lat          : num  42 42 41.9 41.9 41.9 ...
##  $ end_lng          : num  -87.7 -87.7 -87.7 -87.6 -87.6 ...
##  $ usertype         : chr  "member" "member" "member" "member" ...
```

Code

```
data_Q1_2020_v1 <-  data_Q1_2020  %>%
  rename(trip_id = ride_id,
         start_time = started_at,
         end_time = ended_at,
         from_station_id = start_station_id,
         from_station_name = start_station_name,
         to_station_id = end_station_id,
         to_station_name = end_station_name,
         usertype = member_casual)
```

visualiser les variables des 4 trimestres 2019 et ceux du 1er trimestre 2020:

Résultats

```
## 'data.frame':    426887 obs. of  13 variables:
##  $ trip_id          : chr  "EACB19130B0CDA4A" "8FED874C809DC021" "789F3C21E472CA96" "C9A388DAC6ABF313
##  $ rideable_type    : chr  "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ start_time       : chr  "2020-01-21 20:06:59" "2020-01-30 14:22:39" "2020-01-09 19:29:26" "2020-0
##  $ end_time         : chr  "2020-01-21 20:14:30" "2020-01-30 14:26:22" "2020-01-09 19:32:17" "2020-0
##  $ from_station_name: chr  "Western Ave & Leland Ave" "Clark St & Montrose Ave" "Broadway & Belmont A
```

```
##  $ from_station_id  : int  239 234 296 51 66 212 96 96 212 38 ...
##  $ to_station_name  : chr  "Clark St & Leland Ave" "Southport Ave & Irving Park Rd" "Wilton Ave & Bel
##  $ to_station_id    : int  326 318 117 24 212 96 212 212 96 100 ...
##  $ start_lat        : num  42 42 41.9 41.9 41.9 ...
##  $ start_lng        : num  -87.7 -87.7 -87.6 -87.6 -87.6 ...
##  $ end_lat          : num  42 42 41.9 41.9 41.9 ...
##  $ end_lng          : num  -87.7 -87.7 -87.7 -87.6 -87.6 ...
##  $ usertype         : chr  "member" "member" "member" "member" ...

## 'data.frame':    1108163 obs. of  12 variables:
##  $ trip_id          : int  22178529 22178530 22178531 22178532 22178533 22178534 22178535 22178536 22
##  $ start_time       : chr  "2019-04-01 00:02:22" "2019-04-01 00:03:02" "2019-04-01 00:11:07" "2019-04
##  $ end_time         : chr  "2019-04-01 00:09:48" "2019-04-01 00:20:30" "2019-04-01 00:15:19" "2019-04
##  $ bikeid           : int  6251 6226 5649 4151 3270 3123 6418 4513 3280 5534 ...
##  $ tripduration     : chr  "446.0" "1,048.0" "252.0" "357.0" ...
##  $ from_station_id  : int  81 317 283 26 202 420 503 260 211 211 ...
##  $ from_station_name: chr  "Daley Center Plaza" "Wood St & Taylor St" "LaSalle St & Jackson Blvd" "Mc
##  $ to_station_id    : int  56 59 174 133 129 426 500 499 211 211 ...
##  $ to_station_name  : chr  "Desplaines St & Kinzie St" "Wabash Ave & Roosevelt Rd" "Canal St & Madiso
##  $ usertype         : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ gender           : chr  "Male" "Female" "Male" "Male" ...
##  $ birthyear        : int  1975 1984 1990 1993 1992 1999 1969 1991 NA NA ...

## 'data.frame':    365069 obs. of  12 variables:
##  $ trip_id          : int  21742443 21742444 21742445 21742446 21742447 21742448 21742449 21742450 21
##  $ start_time       : chr  "2019-01-01 00:04:37" "2019-01-01 00:08:13" "2019-01-01 00:13:23" "2019-01
##  $ end_time         : chr  "2019-01-01 00:11:07" "2019-01-01 00:15:34" "2019-01-01 00:27:12" "2019-01
##  $ bikeid           : int  2167 4386 1524 252 1170 2437 2708 2796 6205 3939 ...
##  $ tripduration     : chr  "390.0" "441.0" "829.0" "1,783.0" ...
##  $ from_station_id  : int  199 44 15 123 173 98 98 211 150 268 ...
##  $ from_station_name: chr  "Wabash Ave & Grand Ave" "State St & Randolph St" "Racine Ave & 18th St" "
##  $ to_station_id    : int  84 624 644 176 35 49 49 142 148 141 ...
##  $ to_station_name  : chr  "Milwaukee Ave & Grand Ave" "Dearborn St & Van Buren St (*)" "Western Ave
##  $ usertype         : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ gender           : chr  "Male" "Female" "Female" "Male" ...
##  $ birthyear        : int  1989 1990 1994 1993 1994 1983 1984 1990 1995 1996 ...

## 'data.frame':    1640718 obs. of  12 variables:
##  $ trip_id          : int  23479388 23479389 23479390 23479391 23479392 23479393 23479394 23479395 23
##  $ start_time       : chr  "2019-07-01 00:00:27" "2019-07-01 00:01:16" "2019-07-01 00:01:48" "2019-07
##  $ end_time         : chr  "2019-07-01 00:20:41" "2019-07-01 00:18:44" "2019-07-01 00:27:42" "2019-07
##  $ bikeid           : int  3591 5353 6180 5540 6014 4941 3770 5442 2957 6091 ...
##  $ tripduration     : chr  "1,214.0" "1,048.0" "1,554.0" "1,503.0" ...
##  $ from_station_id  : int  117 381 313 313 168 300 168 313 43 43 ...
##  $ from_station_name: chr  "Wilton Ave & Belmont Ave" "Western Ave & Monroe St" "Lakeview Ave & Fullo
##  $ to_station_id    : int  497 203 144 144 62 232 62 144 195 195 ...
##  $ to_station_name  : chr  "Kimball Ave & Belmont Ave" "Western Ave & 21st St" "Larrabee St & Webster
##  $ usertype         : chr  "Subscriber" "Customer" "Customer" "Customer" ...
##  $ gender           : chr  "Male" "" "" "" ...
##  $ birthyear        : int  1992 NA NA NA NA 1990 NA NA NA NA ...

## 'data.frame':    704054 obs. of  12 variables:
##  $ trip_id          : int  25223640 25223641 25223642 25223643 25223644 25223645 25223646 25223647 25
```

```
##  $ start_time       : chr  "2019-10-01 00:01:39" "2019-10-01 00:02:16" "2019-10-01 00:04:32" "2019-10
##  $ end_time         : chr  "2019-10-01 00:17:20" "2019-10-01 00:06:34" "2019-10-01 00:18:43" "2019-10
##  $ bikeid           : int  2215 6328 3003 3275 5294 1891 1061 1274 6011 2957 ...
##  $ tripduration     : chr  "940.0" "258.0" "850.0" "2,350.0" ...
##  $ from_station_id  : int  20 19 84 313 210 156 84 156 156 336 ...
##  $ from_station_name: chr  "Sheffield Ave & Kingsbury St" "Throop (Loomis) St & Taylor St" "Milwaukee
##  $ to_station_id    : int  309 241 199 290 382 226 142 463 463 336 ...
##  $ to_station_name  : chr  "Leavitt St & Armitage Ave" "Morgan St & Polk St" "Wabash Ave & Grand Ave"
##  $ usertype         : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ gender           : chr  "Male" "Male" "Female" "Male" ...
##  $ birthyear        : int  1987 1998 1991 1990 1987 1994 1991 1995 1993 NA ...
```

Ce récapitulatif nous indique que les formats des ID sont de type INT, modifions donc ces formats ID (INT) en caractères.

Code

```
str(data_Q1_2020_v1)
str(data_Q2_2019_v1)
str(data_Q1_2019)
str(data_Q3_2019)
str(data_Q4_2019)
```

Changer les formats ID (INT) en character :

Résultats

Code

```
data_Q1_2019$trip_id <- as.character(data_Q1_2019$trip_id)
data_Q1_2019$bikeid <- as.character(data_Q1_2019$bikeid)
data_Q1_2019$from_station_id <- as.character(data_Q1_2019$from_station_id)
data_Q1_2019$to_station_id <- as.character(data_Q1_2019$to_station_id)

data_Q3_2019$trip_id <- as.character(data_Q3_2019$trip_id)
data_Q3_2019$bikeid <- as.character(data_Q3_2019$bikeid)
data_Q3_2019$from_station_id <- as.character(data_Q3_2019$from_station_id)
data_Q3_2019$to_station_id <- as.character(data_Q3_2019$to_station_id)

data_Q4_2019$trip_id <- as.character(data_Q4_2019$trip_id)
data_Q4_2019$bikeid <- as.character(data_Q4_2019$bikeid)
data_Q4_2019$from_station_id <- as.character(data_Q4_2019$from_station_id)
data_Q4_2019$to_station_id <- as.character(data_Q4_2019$to_station_id)

data_Q2_2019_v1$trip_id <- as.character(data_Q2_2019_v1$trip_id)
data_Q2_2019_v1$bikeid <- as.character(data_Q2_2019_v1$bikeid)
data_Q2_2019_v1$from_station_id <- as.character(data_Q2_2019_v1$from_station_id)
data_Q2_2019_v1$to_station_id <- as.character(data_Q2_2019_v1$to_station_id)

data_Q1_2020_v1$from_station_id <- as.character(data_Q1_2020_v1$from_station_id)
data_Q1_2020_v1$to_station_id <- as.character(data_Q1_2020_v1$to_station_id)
```

Fusionner les trimestres de 2019 et celui de 2020 :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21742443 2019-01-01 00:04:37 2019-01-01 00:11:07   2167        390.0
## 2 21742444 2019-01-01 00:08:13 2019-01-01 00:15:34   4386        441.0
## 3 21742445 2019-01-01 00:13:23 2019-01-01 00:27:12   1524        829.0
## 4 21742446 2019-01-01 00:13:45 2019-01-01 00:43:28    252      1,783.0
## 5 21742447 2019-01-01 00:14:52 2019-01-01 00:20:56   1170        364.0
##   from_station_id                  from_station_name to_station_id
## 1             199              Wabash Ave & Grand Ave            84
## 2              44              State St & Randolph St           624
## 3              15                Racine Ave & 18th St           644
## 4             123      California Ave & Milwaukee Ave           176
## 5             173 Mies van der Rohe Way & Chicago Ave            35
##                   to_station_name   usertype gender birthyear rideable_type
## 1      Milwaukee Ave & Grand Ave Subscriber   Male      1989          <NA>
## 2 Dearborn St & Van Buren St (*) Subscriber Female      1990          <NA>
## 3  Western Ave & Fillmore St (*) Subscriber Female      1994          <NA>
## 4              Clark St & Elm St Subscriber   Male      1993          <NA>
## 5        Streeter Dr & Grand Ave Subscriber   Male      1994          <NA>
##   start_lat start_lng end_lat end_lng
## 1        NA        NA      NA      NA
## 2        NA        NA      NA      NA
## 3        NA        NA      NA      NA
## 4        NA        NA      NA      NA
## 5        NA        NA      NA      NA
```

Code

```
Data <- bind_rows(data_Q1_2019, data_Q2_2019_v1, data_Q3_2019,  data_Q4_2019,
                  data_Q1_2020_v1)
```

Résumé statistique du Dataframe :

Résultats

```
##    trip_id           start_time          end_time            bikeid
##  Length:4244891     Length:4244891     Length:4244891     Length:4244891
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  tripduration       from_station_id    from_station_name  to_station_id
##  Length:4244891     Length:4244891     Length:4244891     Length:4244891
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
```

```
##
##
##
##  to_station_name      usertype            gender            birthyear
##  Length:4244891     Length:4244891     Length:4244891     Min.   :1759
##  Class :character   Class :character   Class :character   1st Qu.:1979
##  Mode  :character   Mode  :character   Mode  :character   Median :1987
##                                                           Mean   :1984
##                                                           3rd Qu.:1992
##                                                           Max.   :2014
##                                                           NA's   :965638
##  rideable_type        start_lat         start_lng          end_lat
##  Length:4244891     Min.   :42        Min.   :-88        Min.   :42
##  Class :character   1st Qu.:42        1st Qu.:-88        1st Qu.:42
##  Mode  :character   Median :42        Median :-88        Median :42
##                     Mean   :42        Mean   :-88        Mean   :42
##                     3rd Qu.:42        3rd Qu.:-88        3rd Qu.:42
##                     Max.   :42        Max.   :-88        Max.   :42
##                     NA's   :3818004   NA's   :3818004    NA's   :3818005
##     end_lng
##  Min.   :-88
##  1st Qu.:-88
##  Median :-88
##  Mean   :-88
##  3rd Qu.:-88
##  Max.   :-88
##  NA's   :3818005
```

Les cellules vides dans les colonnes Birthyear, start_lat, start_lng, end_lat, end_lng sont dues au fait qu'avant 2019, l'entreprise ne collectait pas ces données auprès de ses clients. Ce n'est qu'à partir de 2020 qu'elle a modifié sa politique concernant les informations à conserver dans la base de données.

Code

```
summary(Data)
```

Modifier subscriber en member et customer en casual :

Résultats

En 2019, l'entreprise désignait ses utilisateurs de vélo par les termes « Subscriber » et « Customer ». En 2020, elle a modifié ces appellations en « member » et « Casual ». Afin d'harmoniser les désignations de deux observations dans le dataframe, nous allons remplacer tous les « subscriber » par « member » et les « customer » par « casual ».

Code

```
data_v1 <- Data %>%
  mutate(usertype = case_when(
```

```
    usertype == "Subscriber" ~ "member",
    usertype == "Customer" ~ "casual",
    TRUE ~ usertype
  ))
```

Convertir le format Timestamp en format posixct :

Résultats

Code

```
data_v1$end_time <- as.POSIXct(data_v1$end_time, format = "%Y-%m-%d %H:%M:%S",
                               tz= "UTC")
data_v1$start_time <- as.POSIXct(data_v1$start_time, format = "%Y-%m-%d %H:%M:%S"
                                 , tz= "UTC")
```

Manipulation de données

Calculer la colonne “ride_length” :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21742443 2019-01-01 00:04:37 2019-01-01 00:11:07   2167        390.0
## 2 21742444 2019-01-01 00:08:13 2019-01-01 00:15:34   4386        441.0
## 3 21742445 2019-01-01 00:13:23 2019-01-01 00:27:12   1524        829.0
## 4 21742446 2019-01-01 00:13:45 2019-01-01 00:43:28    252      1,783.0
## 5 21742447 2019-01-01 00:14:52 2019-01-01 00:20:56   1170        364.0
##   from_station_id                  from_station_name to_station_id
## 1             199              Wabash Ave & Grand Ave            84
## 2              44              State St & Randolph St           624
## 3              15                Racine Ave & 18th St           644
## 4             123      California Ave & Milwaukee Ave           176
## 5             173 Mies van der Rohe Way & Chicago Ave            35
##                   to_station_name usertype gender birthyear rideable_type
## 1      Milwaukee Ave & Grand Ave   member   Male      1989          <NA>
## 2 Dearborn St & Van Buren St (*)   member Female      1990          <NA>
## 3  Western Ave & Fillmore St (*)   member Female      1994          <NA>
## 4              Clark St & Elm St   member   Male      1993          <NA>
## 5        Streeter Dr & Grand Ave   member   Male      1994          <NA>
##   start_lat start_lng end_lat end_lng ride_length
## 1        NA        NA      NA      NA    00:06:30
## 2        NA        NA      NA      NA    00:07:21
## 3        NA        NA      NA      NA    00:13:49
## 4        NA        NA      NA      NA    00:29:43
## 5        NA        NA      NA      NA    00:06:04
```

Code

```
data_v2 <- data_v1 %>%
  mutate(ride_length = difftime(end_time, start_time, units = "secs")) %>%
  mutate(ride_length = seconds_to_period(as.numeric(ride_length))) %>%
  mutate(ride_length = sprintf("%02d:%02d:%02d",
                               hour(ride_length),
                               minute(ride_length),
                               second(ride_length)))
```

Calculer la durée du trajet en secondes “durée_sec” :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21742443 2019-01-01 00:04:37 2019-01-01 00:11:07   2167        390.0
## 2 21742444 2019-01-01 00:08:13 2019-01-01 00:15:34   4386        441.0
## 3 21742445 2019-01-01 00:13:23 2019-01-01 00:27:12   1524        829.0
## 4 21742446 2019-01-01 00:13:45 2019-01-01 00:43:28    252      1,783.0
## 5 21742447 2019-01-01 00:14:52 2019-01-01 00:20:56   1170        364.0
##   from_station_id                  from_station_name to_station_id
## 1             199              Wabash Ave & Grand Ave            84
## 2              44              State St & Randolph St           624
## 3              15                Racine Ave & 18th St           644
## 4             123      California Ave & Milwaukee Ave           176
## 5             173 Mies van der Rohe Way & Chicago Ave            35
##                   to_station_name usertype gender birthyear rideable_type
## 1      Milwaukee Ave & Grand Ave   member   Male      1989          <NA>
## 2 Dearborn St & Van Buren St (*)   member Female      1990          <NA>
## 3  Western Ave & Fillmore St (*)   member Female      1994          <NA>
## 4              Clark St & Elm St   member   Male      1993          <NA>
## 5        Streeter Dr & Grand Ave   member   Male      1994          <NA>
##   start_lat start_lng end_lat end_lng ride_length duration_sec
## 1        NA        NA      NA      NA    00:06:30          390
## 2        NA        NA      NA      NA    00:07:21          441
## 3        NA        NA      NA      NA    00:13:49          829
## 4        NA        NA      NA      NA    00:29:43         1783
## 5        NA        NA      NA      NA    00:06:04          364
```

Code

```
data_v2 <- data_v2 %>%
  mutate(duration_sec = as.numeric(difftime(end_time, start_time, units =
                                            "secs")))
```

Filtrer les durées de trajet inférieures à 120 secondes :

Résultats

Puisque les trajets de moins de 120 secondes sont peu probables pour des déplacements en vélo, je les considère comme des anomalies. Je présume que les utilisateurs préfèrent le vélo pour des parcours plus longs.

```
##      trip_id          start_time            end_time bikeid tripduration
## 8   21742450 2019-01-01 00:18:41 2019-01-01 00:20:21   2796        100.0
## 19  21742461 2019-01-01 00:25:28 2019-01-01 00:27:10   3940        102.0
## 103 21742551 2019-01-01 02:23:37 2019-01-01 02:25:30    332        113.0
## 116 21742565 2019-01-01 02:36:43 2019-01-01 02:38:39   6271        116.0
## 214 21742667 2019-01-01 08:48:27 2019-01-01 08:50:20   5327        113.0
##     from_station_id               from_station_name to_station_id
## 8               211          St. Clair St & Erie St           142
## 19               35         Streeter Dr & Grand Ave            35
## 103             217 Racine Ave (May St) & Fulton St           654
## 116             321             Wabash Ave & 9th St            59
## 214             279        Halsted St & 35th St (*)           262
##                      to_station_name usertype gender birthyear rideable_type
## 8               McClurg Ct & Erie St   member   Male      1990          <NA>
## 19           Streeter Dr & Grand Ave   member   Male      1985          <NA>
## 103 Racine Ave & Washington Blvd (*)   member   Male      1989          <NA>
## 116        Wabash Ave & Roosevelt Rd   member   Male      1993          <NA>
## 214             Halsted St & 37th St   member   Male      1989          <NA>
##     start_lat start_lng end_lat end_lng ride_length duration_sec
## 8          NA        NA      NA      NA    00:01:40          100
## 19         NA        NA      NA      NA    00:01:42          102
## 103        NA        NA      NA      NA    00:01:53          113
## 116        NA        NA      NA      NA    00:01:56          116
## 214        NA        NA      NA      NA    00:01:53          113
```

Code

```
data_v2_filtre <- subset(data_v2, duration_sec < 120)
```

Supprimer les durées de trajet inférieures à 120 secondes :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21742443 2019-01-01 00:04:37 2019-01-01 00:11:07   2167        390.0
## 2 21742444 2019-01-01 00:08:13 2019-01-01 00:15:34   4386        441.0
## 3 21742445 2019-01-01 00:13:23 2019-01-01 00:27:12   1524        829.0
## 4 21742446 2019-01-01 00:13:45 2019-01-01 00:43:28    252      1,783.0
## 5 21742447 2019-01-01 00:14:52 2019-01-01 00:20:56   1170        364.0
##   from_station_id                 from_station_name to_station_id
## 1             199             Wabash Ave & Grand Ave            84
## 2              44             State St & Randolph St           624
## 3              15               Racine Ave & 18th St           644
```

```
## 4             123      California Ave & Milwaukee Ave           176
## 5             173 Mies van der Rohe Way & Chicago Ave            35
##                   to_station_name usertype gender birthyear rideable_type
## 1      Milwaukee Ave & Grand Ave   member   Male      1989          <NA>
## 2 Dearborn St & Van Buren St (*)   member Female      1990          <NA>
## 3  Western Ave & Fillmore St (*)   member Female      1994          <NA>
## 4              Clark St & Elm St   member   Male      1993          <NA>
## 5        Streeter Dr & Grand Ave   member   Male      1994          <NA>
##   start_lat start_lng end_lat end_lng ride_length duration_sec
## 1        NA        NA      NA      NA    00:06:30          390
## 2        NA        NA      NA      NA    00:07:21          441
## 3        NA        NA      NA      NA    00:13:49          829
## 4        NA        NA      NA      NA    00:29:43         1783
## 5        NA        NA      NA      NA    00:06:04          364
```

Code

```
data_v3 <- subset(data_v2, duration_sec > 120)
```

Filtrer les durées de trajets trop longues

Filtrer les durées de trajet supérieures à 10800 secondes :

Résultats

Ainsi, je pense que les trajets dont la durée dépasse 3 heures ou 10 800 secondes sont des anomalies. Un utilisateur moyen d'un service de localisation de vélos en libre-service ne ferait probablement pas des trajets de cette longueur.
Je présume que ces durées sont le résultat d'erreurs dans les données, de problèmes techniques ou d'une utilisation non conventionnelle du service.

```
##      trip_id          start_time            end_time bikeid tripduration
## 101 21742549 2019-01-01 02:21:04 2019-01-02 09:35:30   2048    112,466.0
## 146 21742597 2019-01-01 04:07:10 2019-01-02 06:37:40   3500     95,430.0
## 312 21742783 2019-01-01 10:22:26 2019-01-02 10:08:20   1164     85,554.0
## 518 21743130 2019-01-01 12:44:46 2019-01-02 09:57:16   4676     76,350.0
## 521 21743133 2019-01-01 12:45:14 2019-01-02 07:15:36   4750     66,622.0
##     from_station_id            from_station_name to_station_id
## 101              69       Damen Ave & Pierce Ave            67
## 146             506 Spaulding Ave & Armitage Ave           506
## 312              43 Michigan Ave & Washington St            43
## 518             174        Canal St & Madison St            49
## 521             174        Canal St & Madison St            47
##                   to_station_name usertype gender birthyear rideable_type
## 101 Sheffield Ave & Fullerton Ave   casual   Male      1994          <NA>
## 146  Spaulding Ave & Armitage Ave   casual               NA          <NA>
## 312  Michigan Ave & Washington St   member   Male      1977          <NA>
## 518       Dearborn St & Monroe St   casual               NA          <NA>
## 521          State St & Kinzie St   casual               NA          <NA>
##     start_lat start_lng end_lat end_lng ride_length duration_sec
## 101        NA        NA      NA      NA    07:14:26       112466
```

```
## 146        NA        NA      NA      NA    02:30:30        95430
## 312        NA        NA      NA      NA    23:45:54        85554
## 518        NA        NA      NA      NA    21:12:30        76350
## 521        NA        NA      NA      NA    18:30:22        66622
```

Code

```
data_v4_filtre <- subset(data_v3, duration_sec > 10800)
```

Supprimer les durées de trajet supérieures à 10800 secondes :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21742443 2019-01-01 00:04:37 2019-01-01 00:11:07   2167        390.0
## 2 21742444 2019-01-01 00:08:13 2019-01-01 00:15:34   4386        441.0
## 3 21742445 2019-01-01 00:13:23 2019-01-01 00:27:12   1524        829.0
## 4 21742446 2019-01-01 00:13:45 2019-01-01 00:43:28    252      1,783.0
## 5 21742447 2019-01-01 00:14:52 2019-01-01 00:20:56   1170        364.0
##   from_station_id                  from_station_name to_station_id
## 1             199              Wabash Ave & Grand Ave            84
## 2              44              State St & Randolph St           624
## 3              15                Racine Ave & 18th St           644
## 4             123      California Ave & Milwaukee Ave           176
## 5             173 Mies van der Rohe Way & Chicago Ave            35
##                  to_station_name usertype gender birthyear rideable_type
## 1      Milwaukee Ave & Grand Ave   member   Male      1989          <NA>
## 2 Dearborn St & Van Buren St (*)   member Female      1990          <NA>
## 3  Western Ave & Fillmore St (*)   member Female      1994          <NA>
## 4              Clark St & Elm St   member   Male      1993          <NA>
## 5        Streeter Dr & Grand Ave   member   Male      1994          <NA>
##   start_lat start_lng end_lat end_lng ride_length duration_sec
## 1        NA        NA      NA      NA    00:06:30          390
## 2        NA        NA      NA      NA    00:07:21          441
## 3        NA        NA      NA      NA    00:13:49          829
## 4        NA        NA      NA      NA    00:29:43         1783
## 5        NA        NA      NA      NA    00:06:04          364
```

Code

```
data_v4 <- subset(data_v3, duration_sec < 10800)
```

Trie croissant sur la durée de trajet :

Résultats

```
##     trip_id          start_time            end_time bikeid tripduration
## 1  21749035 2019-01-02 21:09:29 2019-01-02 21:11:30    693        121.0
```

```
## 2  21749706 2019-01-03 07:25:31 2019-01-03 07:27:32   3234        121.0
## 3  21754005 2019-01-03 17:31:26 2019-01-03 17:33:27   3939        121.0
## 4  21760855 2019-01-04 16:51:18 2019-01-04 16:53:19   1066        121.0
## 5  21770864 2019-01-06 10:19:44 2019-01-06 10:21:45   3941        121.0
## 6  21771946 2019-01-06 13:11:35 2019-01-06 13:13:36    302        121.0
## 7  21774635 2019-01-07 06:31:28 2019-01-07 06:33:29   6398        121.0
## 8  21776109 2019-01-07 12:45:13 2019-01-07 12:47:14   5821        121.0
## 9  21777249 2019-01-07 16:51:44 2019-01-07 16:53:45    426        121.0
## 10 21779310 2019-01-07 19:59:18 2019-01-07 20:01:19   4564        121.0
##    from_station_id               from_station_name to_station_id
## 1              116      Western Ave & Winnebago Ave           158
## 2              156        Clark St & Wellington Ave           115
## 3               67    Sheffield Ave & Fullerton Ave            87
## 4               73         Jefferson St & Monroe St            73
## 5              159        Claremont Ave & Hirsch St           213
## 6              142             McClurg Ct & Erie St           211
## 7              306          Sheridan Rd & Buena Ave           240
## 8              169           Canal St & Harrison St            68
## 9              654 Racine Ave & Washington Blvd (*)           346
## 10             131        Lincoln Ave & Belmont Ave           153
##                   to_station_name usertype gender birthyear rideable_type
## 1    Milwaukee Ave & Wabansia Ave   member   Male      1984          <NA>
## 2  Sheffield Ave & Wellington Ave   member   Male      1993          <NA>
## 3      Racine Ave & Fullerton Ave   member   Male      1995          <NA>
## 4        Jefferson St & Monroe St   member Female      1991          <NA>
## 5          Leavitt St & North Ave   member   Male      1989          <NA>
## 6          St. Clair St & Erie St   member   Male      1963          <NA>
## 7    Sheridan Rd & Irving Park Rd   member Female      1984          <NA>
## 8          Clinton St & Tilden St   member Female      1968          <NA>
## 9        Ada St & Washington Blvd   member Female      1951          <NA>
## 10 Southport Ave & Wellington Ave   member   Male      1992          <NA>
##    start_lat start_lng end_lat end_lng ride_length duration_sec
## 1         NA        NA      NA      NA    00:02:01          121
## 2         NA        NA      NA      NA    00:02:01          121
## 3         NA        NA      NA      NA    00:02:01          121
## 4         NA        NA      NA      NA    00:02:01          121
## 5         NA        NA      NA      NA    00:02:01          121
## 6         NA        NA      NA      NA    00:02:01          121
## 7         NA        NA      NA      NA    00:02:01          121
## 8         NA        NA      NA      NA    00:02:01          121
## 9         NA        NA      NA      NA    00:02:01          121
## 10        NA        NA      NA      NA    00:02:01          121

##          trip_id          start_time            end_time bikeid tripduration
## 4187299 22660552 2019-05-15 03:00:48 2019-05-15 06:00:45    609     10,797.0
## 4187300 22785888 2019-05-23 16:07:25 2019-05-23 19:07:22   5040     10,797.0
## 4187301 24300605 2019-08-10 13:52:39 2019-08-10 16:52:36    914     10,796.0
## 4187302 22318245 2019-04-15 14:35:12 2019-04-15 17:35:10   1048     10,798.0
## 4187303 23571131 2019-07-05 14:55:37 2019-07-05 17:55:35    497     10,798.0
## 4187304 24341824 2019-08-12 16:17:19 2019-08-12 19:17:17   6127     10,797.0
## 4187305 25940906 2019-12-26 09:53:55 2019-12-26 12:53:53   5647     10,797.0
## 4187306 22386441 2019-04-21 15:23:47 2019-04-21 18:23:46   2159     10,799.0
## 4187307 23056336 2019-06-08 10:42:24 2019-06-08 13:42:23   3489     10,799.0
## 4187308 24011599 2019-07-27 23:17:26 2019-07-28 02:17:25    454     10,798.0
```

```
##         from_station_id          from_station_name to_station_id
## 4187299             110      Dearborn St & Erie St           110
## 4187300             401      Shields Ave & 28th Pl           401
## 4187301              16    Paulina Ave & North Ave            16
## 4187302             174      Canal St & Madison St            85
## 4187303              26   McClurg Ct & Illinois St            26
## 4187304              53        Wells St & Huron St           133
## 4187305             195  Columbus Dr & Randolph St           624
## 4187306              34  Cannon Dr & Fullerton Ave           334
## 4187307             102 Stony Island Ave & 67th St           102
## 4187308             497  Kimball Ave & Belmont Ave            20
##                    to_station_name usertype gender birthyear rideable_type
## 4187299        Dearborn St & Erie St   casual               NA          <NA>
## 4187300        Shields Ave & 28th Pl   casual               NA          <NA>
## 4187301      Paulina Ave & North Ave   casual               NA          <NA>
## 4187302        Michigan Ave & Oak St   casual               NA          <NA>
## 4187303     McClurg Ct & Illinois St   casual               NA          <NA>
## 4187304     Kingsbury St & Kinzie St   casual Female      1994          <NA>
## 4187305   Dearborn St & Van Buren St   casual               NA          <NA>
## 4187306  Lake Shore Dr & Belmont Ave   casual               NA          <NA>
## 4187307   Stony Island Ave & 67th St   casual               NA          <NA>
## 4187308 Sheffield Ave & Kingsbury St   casual   Male      1996          <NA>
##         start_lat start_lng end_lat end_lng ride_length duration_sec
## 4187299        NA        NA      NA      NA    02:59:57        10797
## 4187300        NA        NA      NA      NA    02:59:57        10797
## 4187301        NA        NA      NA      NA    02:59:57        10797
## 4187302        NA        NA      NA      NA    02:59:58        10798
## 4187303        NA        NA      NA      NA    02:59:58        10798
## 4187304        NA        NA      NA      NA    02:59:58        10798
## 4187305        NA        NA      NA      NA    02:59:58        10798
## 4187306        NA        NA      NA      NA    02:59:59        10799
## 4187307        NA        NA      NA      NA    02:59:59        10799
## 4187308        NA        NA      NA      NA    02:59:59        10799
```

Code

```
data_v4 <- data_v4 %>%
  arrange(duration_sec)
```

Compter les cellules vides par variables :

Résultats

Comme indiqué précédemment, en ce qui concerne les variables comportant un grand nombre de cellules vides, cela pourrait être lié à la nature des données recueillies en 2019. De plus, en 2020, d'autres variables ont été ajoutées.

```
##           trip_id        start_time          end_time            bikeid
##                 0                 0                 0            413131
##      tripduration   from_station_id from_station_name     to_station_id
##            413131                 0                 0                 0
```

```
##   to_station_name          usertype            gender         birthyear
##                 0                 0            413131            942770
##     rideable_type         start_lat         start_lng           end_lat
##           3774177           3774177           3774177           3774177
##           end_lng       ride_length      duration_sec
##           3774177                 0                 0
```

Code

```
colSums(is.na(data_v4))
```

Calculer la colonne jour de la semaine à chaque début de trajet :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21749035 2019-01-02 21:09:29 2019-01-02 21:11:30    693        121.0
## 2 21749706 2019-01-03 07:25:31 2019-01-03 07:27:32   3234        121.0
## 3 21754005 2019-01-03 17:31:26 2019-01-03 17:33:27   3939        121.0
## 4 21760855 2019-01-04 16:51:18 2019-01-04 16:53:19   1066        121.0
## 5 21770864 2019-01-06 10:19:44 2019-01-06 10:21:45   3941        121.0
##   from_station_id             from_station_name to_station_id
## 1             116   Western Ave & Winnebago Ave           158
## 2             156     Clark St & Wellington Ave           115
## 3              67 Sheffield Ave & Fullerton Ave            87
## 4              73      Jefferson St & Monroe St            73
## 5             159     Claremont Ave & Hirsch St           213
##                   to_station_name usertype gender birthyear rideable_type
## 1   Milwaukee Ave & Wabansia Ave   member   Male      1984          <NA>
## 2 Sheffield Ave & Wellington Ave   member   Male      1993          <NA>
## 3     Racine Ave & Fullerton Ave   member   Male      1995          <NA>
## 4       Jefferson St & Monroe St   member Female      1991          <NA>
## 5         Leavitt St & North Ave   member   Male      1989          <NA>
##   start_lat start_lng end_lat end_lng ride_length duration_sec day_of_week
## 1        NA        NA      NA      NA    00:02:01          121    mercredi
## 2        NA        NA      NA      NA    00:02:01          121       jeudi
## 3        NA        NA      NA      NA    00:02:01          121       jeudi
## 4        NA        NA      NA      NA    00:02:01          121    vendredi
## 5        NA        NA      NA      NA    00:02:01          121    dimanche
```

Code

```
data_v4<- data_v4 %>%
  mutate(day_of_week = wday(start_time,label=TRUE, abbr = FALSE, week_start=1))
```

Calculer la colonne mois pour chaque début de trajet :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21749035 2019-01-02 21:09:29 2019-01-02 21:11:30    693        121.0
## 2 21749706 2019-01-03 07:25:31 2019-01-03 07:27:32   3234        121.0
## 3 21754005 2019-01-03 17:31:26 2019-01-03 17:33:27   3939        121.0
## 4 21760855 2019-01-04 16:51:18 2019-01-04 16:53:19   1066        121.0
## 5 21770864 2019-01-06 10:19:44 2019-01-06 10:21:45   3941        121.0
##   from_station_id             from_station_name to_station_id
## 1             116   Western Ave & Winnebago Ave           158
## 2             156     Clark St & Wellington Ave           115
## 3              67 Sheffield Ave & Fullerton Ave            87
## 4              73      Jefferson St & Monroe St            73
## 5             159     Claremont Ave & Hirsch St           213
##                   to_station_name usertype gender birthyear rideable_type
## 1   Milwaukee Ave & Wabansia Ave   member   Male      1984          <NA>
## 2 Sheffield Ave & Wellington Ave   member   Male      1993          <NA>
## 3     Racine Ave & Fullerton Ave   member   Male      1995          <NA>
## 4       Jefferson St & Monroe St   member Female      1991          <NA>
## 5         Leavitt St & North Ave   member   Male      1989          <NA>
##   start_lat start_lng end_lat end_lng ride_length duration_sec day_of_week
## 1        NA        NA      NA      NA    00:02:01          121    mercredi
## 2        NA        NA      NA      NA    00:02:01          121       jeudi
## 3        NA        NA      NA      NA    00:02:01          121       jeudi
## 4        NA        NA      NA      NA    00:02:01          121    vendredi
## 5        NA        NA      NA      NA    00:02:01          121    dimanche
##     month
## 1 janvier
## 2 janvier
## 3 janvier
## 4 janvier
## 5 janvier
```

Code

```
data_v4 <- data_v4 %>%
  mutate(month = month(start_time, label = TRUE, abbr = FALSE))
```

Calculer la colonne saison pour chaque début de trajet :

Résultats

```
##    trip_id          start_time            end_time bikeid tripduration
## 1 21749035 2019-01-02 21:09:29 2019-01-02 21:11:30    693        121.0
## 2 21749706 2019-01-03 07:25:31 2019-01-03 07:27:32   3234        121.0
## 3 21754005 2019-01-03 17:31:26 2019-01-03 17:33:27   3939        121.0
## 4 21760855 2019-01-04 16:51:18 2019-01-04 16:53:19   1066        121.0
## 5 21770864 2019-01-06 10:19:44 2019-01-06 10:21:45   3941        121.0
##   from_station_id             from_station_name to_station_id
```

```
## 1             116   Western Ave & Winnebago Ave           158
## 2             156     Clark St & Wellington Ave           115
## 3              67 Sheffield Ave & Fullerton Ave            87
## 4              73      Jefferson St & Monroe St            73
## 5             159     Claremont Ave & Hirsch St           213
##                  to_station_name usertype gender birthyear rideable_type
## 1   Milwaukee Ave & Wabansia Ave   member   Male      1984          <NA>
## 2 Sheffield Ave & Wellington Ave   member   Male      1993          <NA>
## 3     Racine Ave & Fullerton Ave   member   Male      1995          <NA>
## 4       Jefferson St & Monroe St   member Female      1991          <NA>
## 5         Leavitt St & North Ave   member   Male      1989          <NA>
##   start_lat start_lng end_lat end_lng ride_length duration_sec day_of_week
## 1        NA        NA      NA      NA    00:02:01          121    mercredi
## 2        NA        NA      NA      NA    00:02:01          121       jeudi
## 3        NA        NA      NA      NA    00:02:01          121       jeudi
## 4        NA        NA      NA      NA    00:02:01          121    vendredi
## 5        NA        NA      NA      NA    00:02:01          121    dimanche
##     month season
## 1 janvier  Hiver
## 2 janvier  Hiver
## 3 janvier  Hiver
## 4 janvier  Hiver
## 5 janvier  Hiver
```

Code

```r
data_v4 <- data_v4 %>%
  mutate(season = case_when(
    month(start_time) %in% c(12, 1, 2) ~ "Hiver",
    month(start_time) %in% c(3, 4, 5) ~ "Printemps",
    month(start_time) %in% c(6, 7, 8) ~ "Été",
    month(start_time) %in% c(9, 10, 11) ~ "Automne"
  ))
```

Résumé statistique descriptive du dataset :

Résultats

```
##    trip_id             start_time
##  Length:4187308     Min.   :2019-01-01 00:04:37.00
##  Class :character   1st Qu.:2019-06-05 08:52:39.00
##  Mode  :character   Median :2019-08-05 07:07:09.50
##                     Mean   :2019-08-09 12:07:18.71
##                     3rd Qu.:2019-10-05 11:21:56.25
##                     Max.   :2020-03-31 23:51:34.00
##
##     end_time                         bikeid          tripduration
##  Min.   :2019-01-01 00:11:07.00   Length:4187308     Length:4187308
##  1st Qu.:2019-06-05 09:06:48.25   Class :character   Class :character
##  Median :2019-08-05 07:19:18.50   Mode  :character   Mode  :character
##  Mean   :2019-08-09 12:24:45.04
```

```
##  3rd Qu.:2019-10-05 11:44:23.25
##  Max.   :2020-04-01 00:52:37.00
##
##  from_station_id    from_station_name  to_station_id      to_station_name
##  Length:4187308     Length:4187308     Length:4187308     Length:4187308
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    usertype            gender            birthyear      rideable_type
##  Length:4187308     Length:4187308     Min.   :1759     Length:4187308
##  Class :character   Class :character   1st Qu.:1979     Class :character
##  Mode  :character   Mode  :character   Median :1987     Mode  :character
##                                        Mean   :1984
##                                        3rd Qu.:1992
##                                        Max.   :2014
##                                        NA's   :942770
##    start_lat         start_lng          end_lat           end_lng
##  Min.   :42        Min.   :-88       Min.   :42        Min.   :-88
##  1st Qu.:42        1st Qu.:-88       1st Qu.:42        1st Qu.:-88
##  Median :42        Median :-88       Median :42        Median :-88
##  Mean   :42        Mean   :-88       Mean   :42        Mean   :-88
##  3rd Qu.:42        3rd Qu.:-88       3rd Qu.:42        3rd Qu.:-88
##  Max.   :42        Max.   :-88       Max.   :42        Max.   :-88
##  NA's   :3774177   NA's   :3774177   NA's   :3774177   NA's   :3774177
##  ride_length         duration_sec     day_of_week           month
##  Length:4187308     Min.   :  121   lundi   :618410   août     : 583848
##  Class :character   1st Qu.:  408   mardi   :652130   juillet  : 551418
##  Mode  :character   Median :  695   mercredi:645635   septembre: 487990
##                     Mean   : 1046   jeudi   :645592   juin     : 470335
##                     3rd Qu.: 1248   vendredi:630196   octobre  : 367129
##                     Max.   :10799   samedi  :525178   mai      : 363364
##                                     dimanche:470167   (Other)  :1363224
##     season
##  Length:4187308
##  Class :character
##  Mode  :character
##
##
##
##
```

Table 1: Data summary

Name	data_v4
Number of rows	4187308
Number of columns	22
Column type frequency:	
character	12
factor	2

numeric	6
POSIXct	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
trip_id	0	1.0	8	16	0	4187308	0
bikeid	413131	0.9	1	4	0	6013	0
tripduration	413131	0.9	5	8	0	10662	0
from_station_id	0	1.0	1	3	0	618	0
from_station_name	0	1.0	5	43	0	643	0
to_station_id	0	1.0	1	3	0	619	0
to_station_name	0	1.0	5	43	0	644	0
usertype	0	1.0	6	6	0	2	0
gender	413131	0.9	0	6	549877	3	0
rideable_type	3774177	0.1	11	11	0	1	0
ride_length	0	1.0	8	8	0	10673	0
season	0	1.0	3	9	0	4	0

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
day_of_week	0	1	TRUE	7	mar: 652130, mer: 645635, jeu: 645592, ven: 630196
month	0	1	TRUE	12	aoû: 583848, jui: 551418, sep: 487990, jui: 470335

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
birthyear	942770	0.77	1984.06	10.87	1759.00	1979.00	1987.00	1992.00	2014.00	
start_lat	3774177	0.10	41.90	0.04	41.74	41.88	41.89	41.92	42.06	
start_lng	3774177	0.10	-87.64	0.02	-87.77	-87.65	-87.64	-87.63	-87.55	
end_lat	3774177	0.10	41.90	0.04	41.74	41.88	41.89	41.92	42.06	
end_lng	3774177	0.10	-87.64	0.02	-87.77	-87.65	-87.64	-87.63	-87.55	
duration_sec	0	1.00	1046.35	1146.62	121.00	408.00	695.00	1248.00	10799.00	

Variable type: POSIXct

skim_variable	n_missing	complete_rate	min	max	median	n_unique
start_time	0	1	2019-01-01 00:04:37	2020-03-31 23:51:34	2019-08-05 07:07:09	3658707
end_time	0	1	2019-01-01 00:11:07	2020-04-01 00:52:37	2019-08-05 07:19:18	3591543

```
## # A tibble: 2 x 2
## # Groups:   usertype [2]
##   usertype       n
##   <chr>      <int>
## 1 casual    910202
## 2 member   3277106
```

Code

```
summary(data_v4)

skim(data_v4)

resultat_data_v4 <- data_v4 %>%
  group_by(usertype) %>%
  count()
```

Analyse

Comparer la durée moyenne de trajet, la durée max, l'écart-type :

Résultats

Les durées minimales et maximales sont quasiment identiques car j'ai défini une durée minimale de 120 secondes pour qu'un utilisateur puisse emprunter un vélo, et une durée maximale de 10800 secondes.

Le temps moyen de voyage est supérieur chez les clients ponctuels comparés aux membres, atteignant 2111 secondes contre 751 secondes. Cette tendance ne représente pas fidèlement la réalité, car l'écart entre ces deux ensembles est trop significatif.

```
## # A tibble: 2 x 7
##   usertype avg_duration max_duration min_duration sd_duration variance_duration
##   <chr>           <dbl>        <dbl>        <dbl>       <dbl>             <dbl>
## 1 casual          2111.        10799          121       1837.          3375088.
## 2 member           751.        10794          121        583.           339898.
## # i 1 more variable: total_rides <int>
```

Code

```
summary_stats <- data_v4 %>%
  group_by(usertype) %>%
  summarise(
    avg_duration = mean(duration_sec),
    max_duration = max(duration_sec),
    min_duration = min(duration_sec),
    sd_duration = sd(duration_sec),
    variance_duration = var(duration_sec),
    total_rides = n()
  )
```

Comparer la durée moyenne de trajet par jour et par type d'utilisateur :

Résultats

Les trajets des utilisateurs sont généralement plus longs particulièrement le samedi et le dimanche, en comparaison avec les autres jours de la semaine, .

```
## `summarise()` has grouped output by 'usertype'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 14 x 8
## # Groups:   usertype [2]
##    usertype day_of_week avg_duration max_duration min_duration sd_duration
##    <chr>    <ord>              <dbl>        <dbl>        <dbl>       <dbl>
##  1 casual   lundi              2118.        10798          121       1858.
##  2 casual   mardi              1975.        10789          121       1795.
##  3 casual   mercredi           1969.        10797          121       1796.
##  4 casual   jeudi              1964.        10798          121       1769.
##  5 casual   vendredi           2049.        10798          121       1814.
##  6 casual   samedi             2247.        10799          121       1865.
##  7 casual   dimanche           2217.        10799          121       1869.
##  8 member   lundi               733.        10717          121        553.
##  9 member   mardi               732.        10691          121        552.
## 10 member   mercredi            734.        10785          121        551.
## 11 member   jeudi               733.        10786          121        560.
## 12 member   vendredi            727.        10794          121        560.
## 13 member   samedi              833.        10794          121        683.
## 14 member   dimanche            835.        10767          121        692.
## # i 2 more variables: variance_duration <dbl>, total_rides <int>
```

Code

```
weekly_stats <- data_v4 %>%
  group_by(usertype, day_of_week) %>%
  summarise(
    avg_duration = mean(duration_sec),
    max_duration = max(duration_sec),
    min_duration = min(duration_sec),
    sd_duration = sd(duration_sec),
    variance_duration = var(duration_sec),
    total_rides = n()
  ) %>%
  arrange(usertype, day_of_week)
```

Comparer le nombre de trajets par station de départ et par type d'utilisateurs :

Résultats

Les dix stations les moins fréquentées sont principalement utilisées par les usagers occasionnels, tandis que celles ayant le plus de visites sont empruntées par les membres.

```
## `summarise()` has grouped output by 'from_station_name'. You can override using
## the `.groups` argument.
```

```
## # A tibble: 1,281 x 3
## # Groups:   from_station_name [643]
##    from_station_name                    usertype total_rides
##    <chr>                                <chr>          <int>
##  1 Elizabeth St & 59th St               casual             1
##  2 LBS - BBB La Magie                   casual             1
##  3 MTL-ECO5.1-01                        casual             1
##  4 Special Events                       casual             1
##  5 DIVVY CASSETTE REPAIR MOBILE STATION member             2
##  6 DIVVY Map Frame B/C Station          casual             2
##  7 HQ QR                                casual             2
##  8 MTL-ECO5.1-01                        member             2
##  9 South Chicago Ave & Elliot Ave       member             2
## 10 Racine Ave & 61st St                 member             3
## # i 1,271 more rows
```

```
## # A tibble: 10 x 3
## # Groups:   from_station_name [10]
##    from_station_name           usertype total_rides
##    <chr>                       <chr>          <int>
##  1 Canal St & Madison St       member         30326
##  2 Daley Center Plaza          member         33298
##  3 Franklin St & Monroe St     member         34228
##  4 Kingsbury St & Kinzie St    member         34891
##  5 Columbus Dr & Randolph St   member         35215
##  6 Lake Shore Dr & Monroe St   casual         40360
##  7 Clinton St & Washington Blvd member        50754
##  8 Clinton St & Madison St     member         51925
##  9 Streeter Dr & Grand Ave     casual         54040
## 10 Canal St & Adams St         member         57590
```

Code

```r
station_stats <- data_v4 %>%
  group_by(from_station_name, usertype) %>%
  summarise(total_rides = n()) %>%
  arrange(total_rides)
```

Comparer la durée moyenne par saison et par type d'utilisateur :

Résultats

Durant l'été, la durée moyenne des déplacements s'accumule pour les deux catégories d' usagers.

```
## `summarise()` has grouped output by 'season'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 8 x 8
## # Groups:   season [4]
##   season    usertype mean_ride_length max_duration min_duration sd_duration
##   <chr>     <chr>               <dbl>        <dbl>        <dbl>       <dbl>
## 1 Automne   casual              1923.        10795          121       1750.
## 2 Hiver     casual              1683.        10798          121       1615.
## 3 Printemps casual              2208.        10799          121       1830.
## 4 Été       casual              2199.        10799          121       1883.
## 5 Automne   member               730.        10794          121        553.
## 6 Hiver     member               644.        10786          121        504.
## 7 Printemps member               741.        10787          121        585.
## 8 Été       member               828.        10765          121        629.
## # i 2 more variables: variance_duration <dbl>, total_rides <int>
```

Code

```
season_usertype <- data_v4 %>%
  group_by(season, usertype) %>%
  summarize(mean_ride_length = mean(duration_sec),
            max_duration = max(duration_sec),
            min_duration = min(duration_sec),
            sd_duration = sd(duration_sec),
            variance_duration = var(duration_sec),
            total_rides = n()
  ) %>%
  arrange(usertype, season)
```

Comparer la durée moyenne par mois et par type d'utilisateur :

Résultats

```
## `summarise()` has grouped output by 'month'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 24 x 8
## # Groups:   month [12]
##    month     usertype mean_ride_length max_duration min_duration sd_duration
##    <ord>     <chr>               <dbl>        <dbl>        <dbl>       <dbl>
##  1 janvier   casual              1600.        10764          121       1606.
##  2 février   casual              1731.        10788          121       1680.
##  3 mars      casual              2016.        10788          121       1727.
##  4 avril     casual              2262.        10799          121       1834.
##  5 mai       casual              2272.        10797          121       1869.
##  6 juin      casual              2200.        10799          121       1855.
##  7 juillet   casual              2225.        10799          121       1895.
##  8 août      casual              2174.        10798          121       1892.
##  9 septembre casual              2033.        10795          121       1811.
## 10 octobre   casual              1830.        10781          121       1689.
## # i 14 more rows
## # i 2 more variables: variance_duration <dbl>, total_rides <int>
```

Code

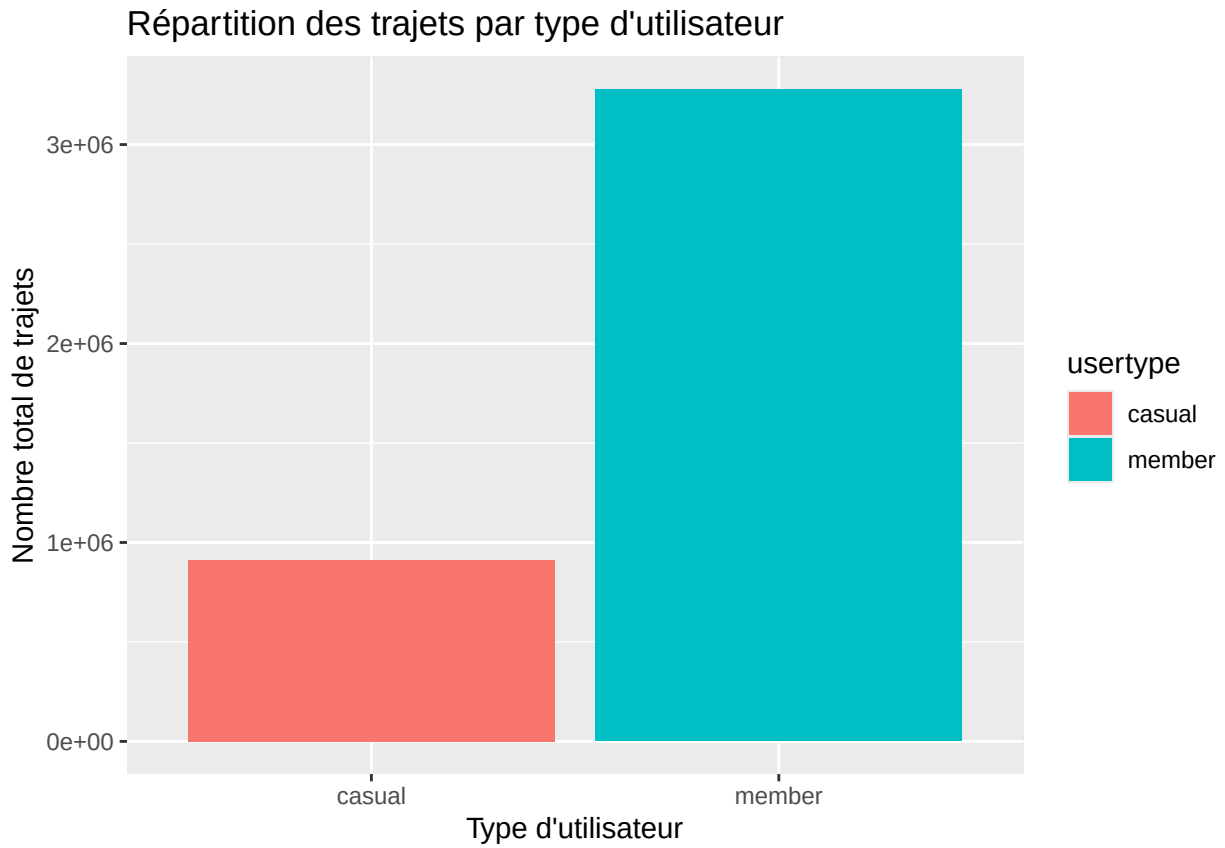
```
month_usertype <- data_v4 %>%
  group_by(month, usertype) %>%
  summarize(mean_ride_length = mean(duration_sec),
            max_duration = max(duration_sec),
            min_duration = min(duration_sec),
            sd_duration = sd(duration_sec),
            variance_duration = var(duration_sec),
            total_rides = n()
  ) %>%
  arrange(usertype, month)
```

Visualiser les tendances

Répartition des trajets par type d'utilisateur :

Résultats

Le nombre de « membres » dépasse largement celui des « clients occasionnels », et ce graphique illustre que les membres effectuent plus de déplacements que les clients occasionnels, ce qui est tout à fait compréhensible.



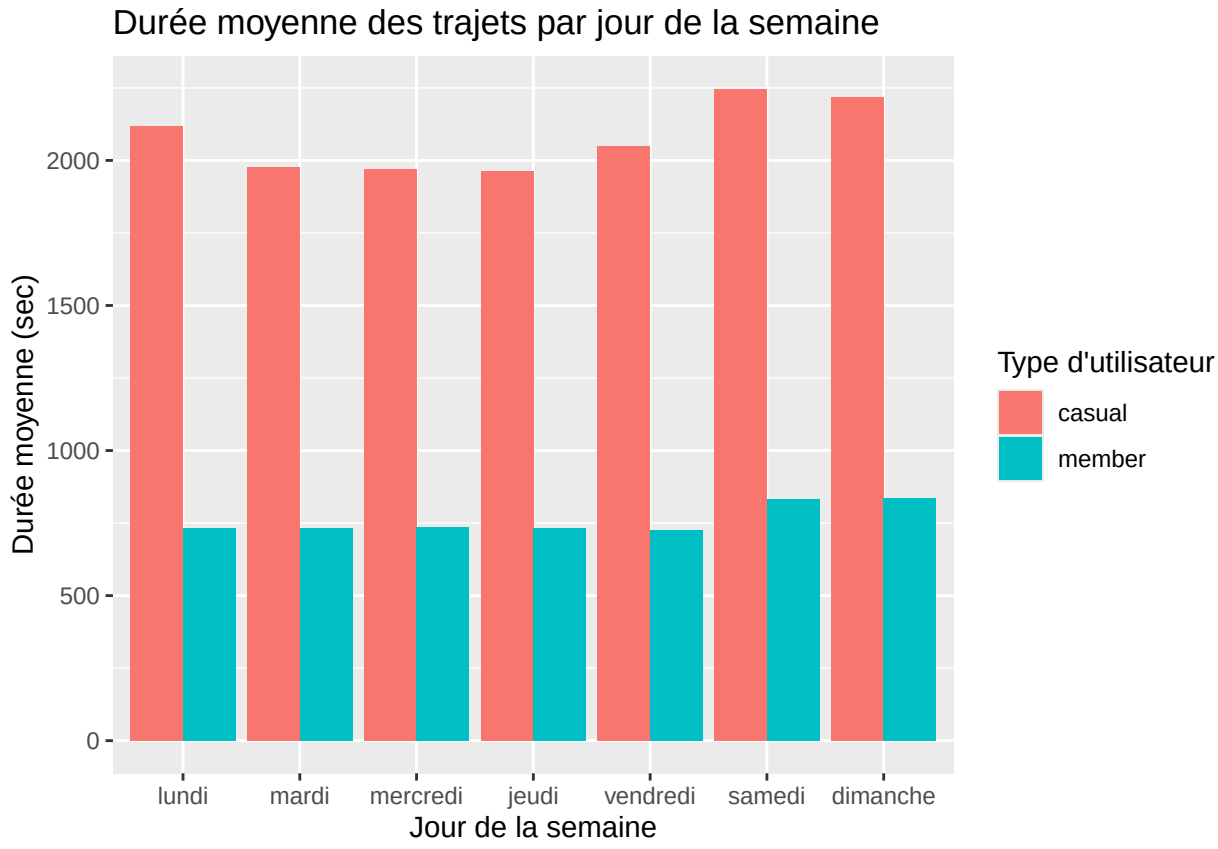
Code

```
ggplot(data_v4, aes(x = usertype, fill = usertype)) +
  geom_bar(position = "dodge", stat = "count") +
  labs(
    title = "Répartition des trajets par type d'utilisateur",
    x = "Type d'utilisateur",
    y = "Nombre total de trajets")
```

visualiser les Durées moyennes des trajets par jour de la semaine :

Résultats

Les durées de voyages atteignent leur maximum les samedis et dimanches. Il est possible que la durée moyenne de déplacement des clients occasionnels soit plus longue que celle des membres. Il faut noter que le groupe « membre » est bien plus grand que celui des clients occasionnels. Dans cette optique, des tests statistiques nous aideront à confirmer ou à infirmer cette supposition.



Code

```
ggplot(weekly_stats, aes(x = day_of_week, y = avg_duration, fill = usertype)) +
  geom_col(position = "dodge") +
```

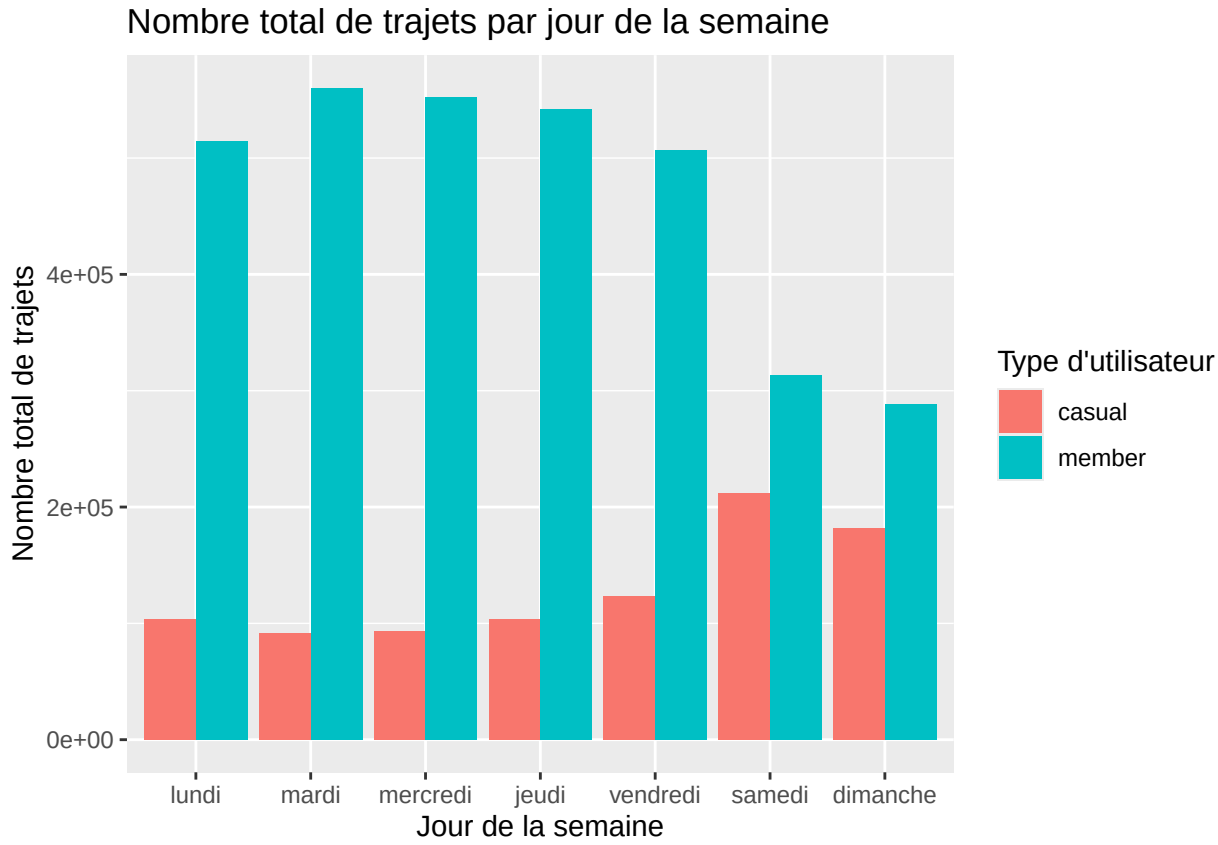
```
  labs(title = "Durée moyenne des trajets par jour de la semaine",
       x = "Jour de la semaine", y = "Durée moyenne (sec)",
       fill = "Type d'utilisateur")
```

visualiser le Nombre total de trajets par jour de la semaine :

Résultats

Selon le comptage du nombre de trajets, les membres utilisent davantage le vélo que les clients occasionnels. Cependant, l'estimation de la durée moyenne des trajets indique que les clients occasionnels roulent plus souvent à vélo que les membres.

Par ailleurs, chez les membres, on constate une utilisation importante du vélo du lundi au vendredi, ce qui est en opposition avec le comportement des clients occasionnels. En revanche, durant le week-end, l'usage du vélo diminue chez les membres et s'accroît chez les clients occasionnels.



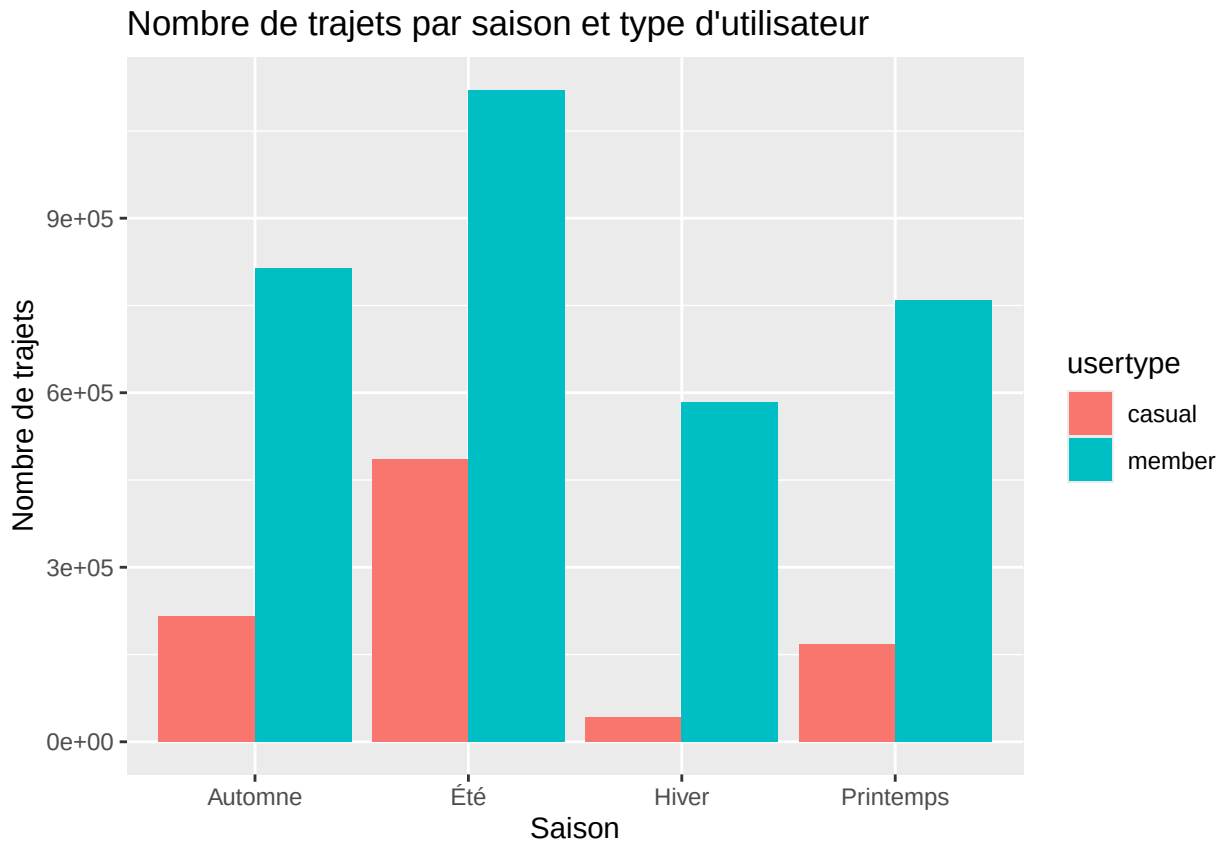
Code

```
ggplot(weekly_stats, aes(x = day_of_week, y = total_rides, fill = usertype)) +
  geom_col(position = "dodge") +
  labs(title = "Nombre total de trajets par jour de la semaine",
       x = "Jour de la semaine", y = "Nombre total de trajets",
       fill = "Type d'utilisateur")
```

visualiser les durées de trajets par saisons et type d'utilisateur :

Résultats

Durant l'été, les utilisateurs ont tendance à opter pour le vélo afin de profiter du soleil. Durant l'hiver, on constate une diminution significative du nombre de trajets pendant cette saison.



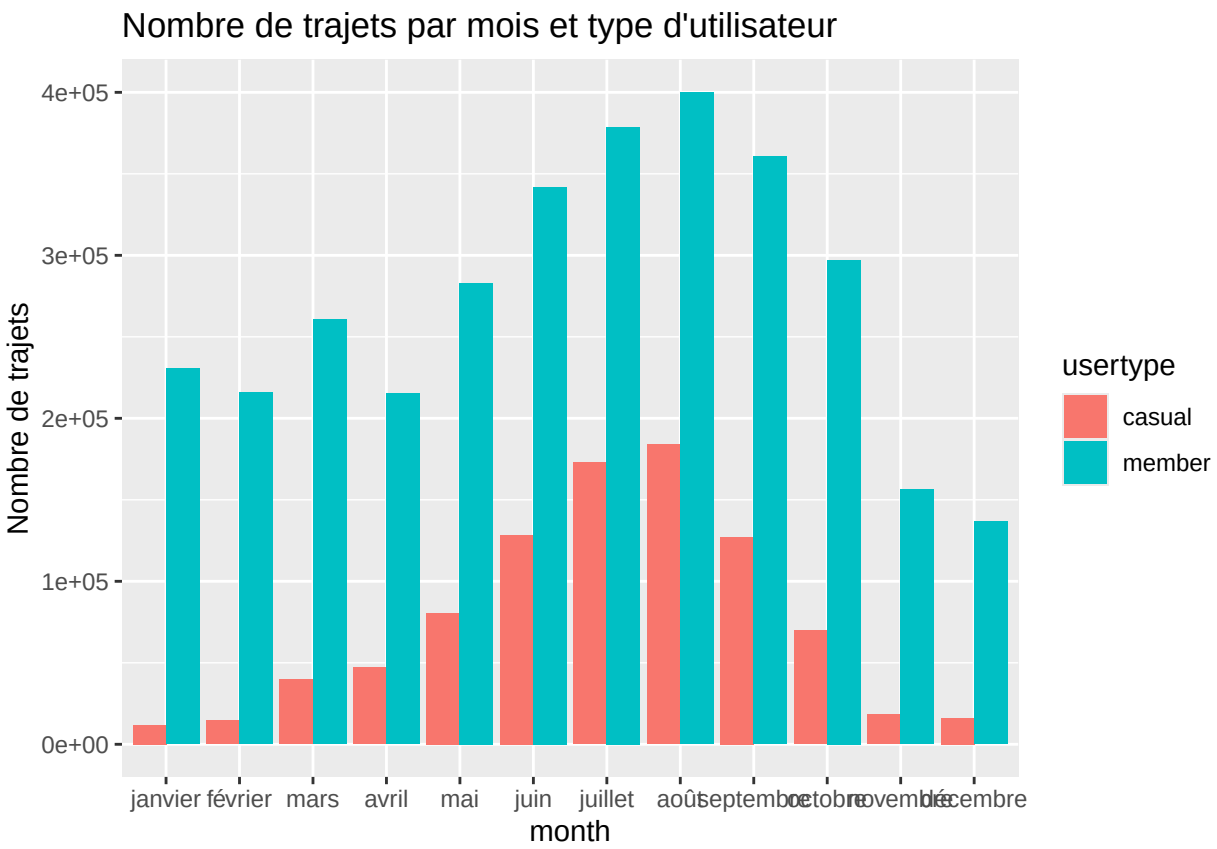
Code

```
ggplot(data_v4, aes(x = season, fill = usertype)) +
  geom_bar(position = "dodge") +
  labs(title = "Nombre de trajets par saison et type d'utilisateur",
       x = "Saison", y = "Nombre de trajets")
```

Visualiser les nombres de trajet par mois :

Résultats

On observe une augmentation de l'utilisation des vélos pendant les mois d'été, suivie d'une diminution pendant les mois d'hiver.



Code

```
ggplot(data_v4, aes(x = month, fill = usertype)) +
  geom_bar(position = "dodge") +
  labs(title = "Nombre de trajets par mois et type d'utilisateur",
       x = "month", y = "Nombre de trajets")
```

Comparer les moyennes

La différence considérable entre le groupe « membre » et celui du « client occasionnel » nous offre la possibilité d'effectuer des tests statistiques pour confirmer ou infirmer les hypothèses.

Vérifier la distribution des durées

Séparer les groupes :

Résultats

```
## [1] 121 121 121 121 121 121
```

```
## [1] 121 121 121 121 121 121
```

Code

```
abonnes <- data_v4 %>% filter(usertype == "member") %>% pull(duration_sec)
non_abonnes <- data_v4 %>% filter(usertype == "casual") %>% pull(duration_sec)
```

Test de Shapiro-Wilk pour la normalité

n Pour rendre l'échantillonnage reproductible

Le test de Shapiro-Wilk est un test statistique puissant utilisé pour évaluer si un échantillon de données provient d'une distribution normale

H0 : Les données suivent une distribution normale. H1 : Les données ne suivent pas une distribution normale

Effectuons un échantillonnage aléatoire pour examiner la normalité des deux groupes, en considérant « member » comme abonné et « casual » comme non abonné, avec n inférieur ou égal à 5000.

Test de Shapiro-Wilk pour la normalité :

Résultats *abonnes* : W = 0.79716 p-value < 2.2e-16 (très faible) La valeur p est extrêmement faible (beaucoup moins que 0,05). Cela signifie qu'il est extrêmement improbable que les données des abonnés proviennent d'une distribution normale.

Nous rejetons l'hypothèse nulle, les données des abonnées ne suivent pas une distribution normale.

non_abonnes : W = 0.79865 p-value < 2.2e-16 (très faible) De même, la valeur p est extrêmement faible. Cela indique que les données des non-abonnés ne sont pas normalement distribuées. Nous rejettons l'hypothèse nulle, les données des non abonnées ne suivent pas une distribution normale.

Conclusions importantes

Non-normalité : Les deux échantillons, abonnés et non-abonnés, montrent une déviation significative de la normalité.

```
##
##  Shapiro-Wilk normality test
##
## data:  sample_abonnes
## W = 0.71769, p-value < 2.2e-16
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample_non_abonnes
## W = 0.7908, p-value < 2.2e-16
```

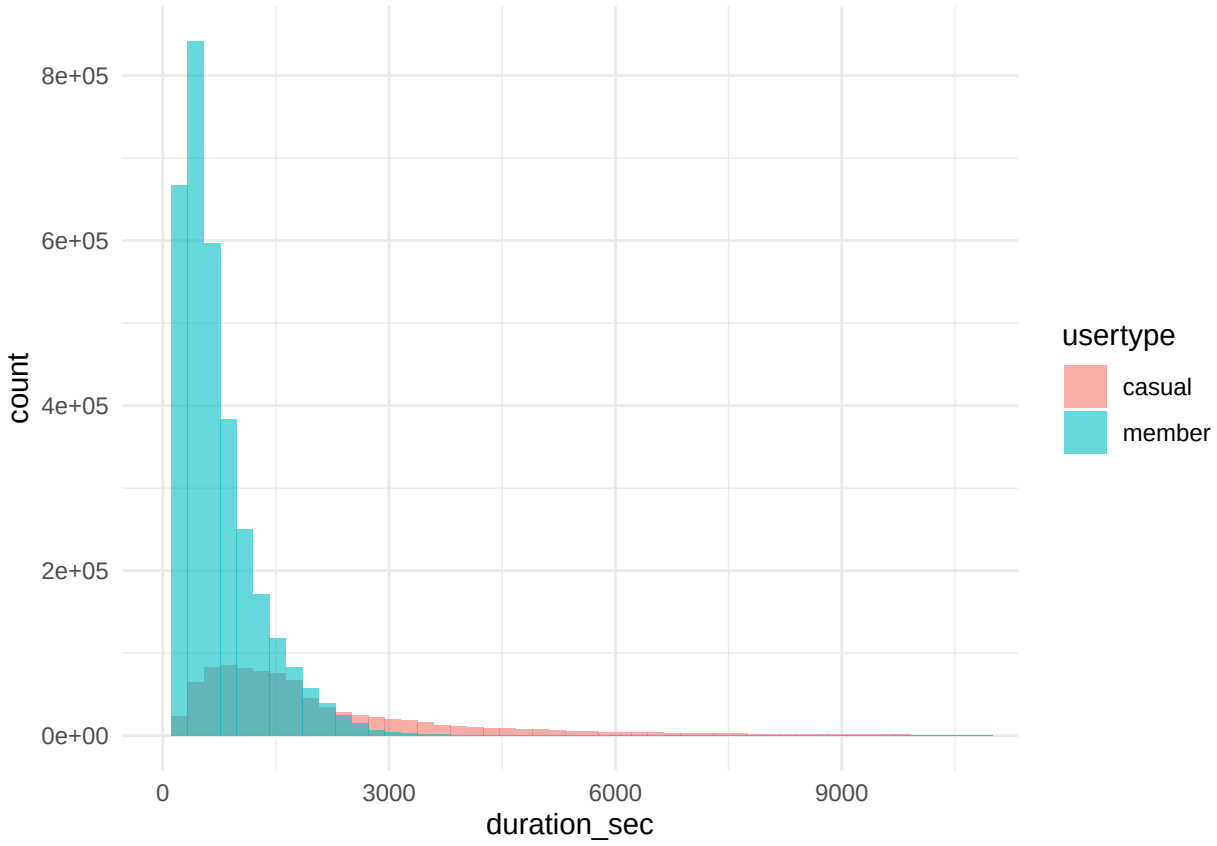
```
set.seed(123)
sample_abonnes <- sample(abonnes, size = 5000)
sample_non_abonnes <- sample(non_abonnes, size = 5000)
```

Code

Visualisation avec un histogramme, ajuster la plage (exemple : 0 à 10800 s :

Résultats

Pour compléter le test de Shapiro-Wilk, effectuons un histogramme.
Ce diagramme en barres illustre bien que la distribution n'est pas normale.



Code

```
ggplot(data_v4, aes(x = duration_sec, fill = usertype)) +
  geom_histogram(alpha = 0.6, bins = 50, position = "identity") +
  coord_cartesian(xlim = c(0, 10800)) +
  theme_minimal()
```

Les données ne sont pas normales → Test de Mann-Whitney (Wilcoxon) :

Résultats

Notre dataset ne suit pas une distribution normale, vérifions leurs différence significative entre les membres et les clients occasionnels.

H0 : Il n'y a pas de différence significative entre les abonnes et non_abonnes. H1 : Il existe une différence significative entre les abonnes et non_abonnes.

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  abonnes and non_abonnes
## W = 5.4745e+11, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Code

```
wilcox_test <- wilcox.test(abonnes, non_abonnes)
```

Analyse

W = 5.4745e+11 : Cette valeur représente la statistique de test de Wilcoxon. C'est une valeur très élevée, ce qui suggère une différence significative entre les groupes.

p-value < 2.2e-16 : La valeur p est extrêmement faible (beaucoup moins que 0,05). Cela signifie qu'il est extrêmement improbable d'obtenir ces résultats si les durées de trajet des abonnés et des non-abonnés étaient similaires. Nous rejetons l'hypothèse nulle.

alternative hypothese : Cela indique que le test a été effectué pour déterminer si les groupes sont différents (test bilatéral). Le résultat confirme qu'il existe une différence significative dans la localisation (médiane) des durées de trajet entre les abonnés et les non-abonnés.

Conclusions importantes

Différence significative : Il existe une différence statistiquement significative dans les durées de trajet entre les abonnés et les non-abonnés.

Non-normalité : Ce test a été utilisé car les données ne sont pas normalement distribuées, comme confirmé par le test de Shapiro-Wilk précédent.

Trois principales recommandations :

• *Recommandation 1*: Cibler les cyclistes occasionnels qui utilisent les vélos pendant les heures de pointe en semaine avec des offres d'abonnement axées sur les déplacements domicile-travail.

• *Recommandation 2*: Promouvoir les abonnements annuels auprès des cyclistes occasionnels qui effectuent des trajets plus longs, en mettant en avant les avantages économiques à long terme.

• *Recommandation 3*: Offrir des essais gratuits ou des réductions sur les abonnements annuels aux cyclistes occasionnels qui utilisent fréquemment les vélos pendant les week-ends, afin de les inciter à adopter un usage plus régulier.