Exploration de données

Introduction

Dans un premier temps, nous entamerons notre exploration des données par choisir les variables qui joue un rôle très important dans la classification,

- L'année,
- Le mois,
- Jour de la semaine,
- Heure de la journée,
- Localisation,
- District.

Dans le cadre exploratoire, dans la suite, on pourrait ajouter ou enlever certaines variables qui ne pourrait avoir une faible influence dans la classification.

Packages nécessaires

Durant la suite de ce chapitre, nous utiliserons les packages suivants,

```
library(ggmap)
library(ggplot2)
library(dplyr)
library(reshape2)
```

Chargement des données

Le chargement des données d'apprentissage (train) et de test, se fait comme ceci,

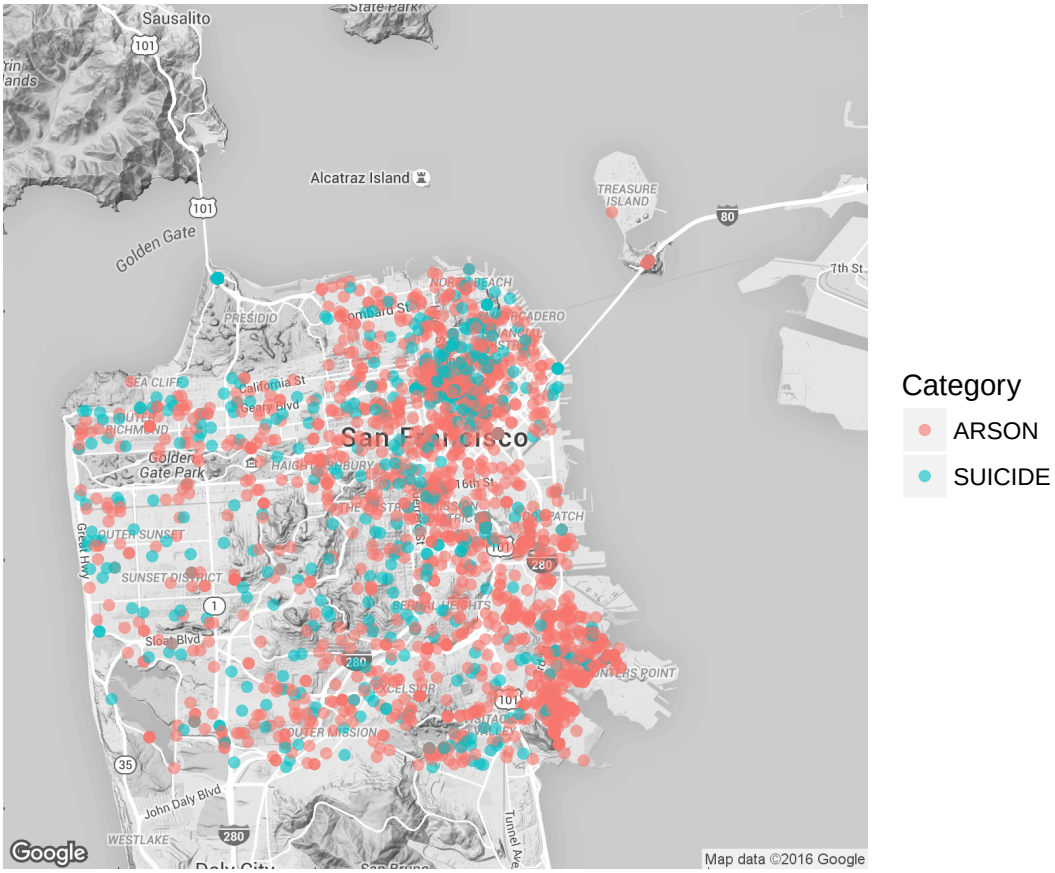
```
train <- read.csv("train.csv")
#test <- read.csv("test.csv")
```

Visualisation

```
# fonction pour filtrer les données selon la catégorie et les projeter sur la map
map_crime <- function(crime_df, crime) {
    filtered <- filter(crime_df, Category %in% crime)
    plot <- ggmap(map, extent = 'device') +
        geom_point(data = filtered, aes(x = X, y = Y, color = Category), alpha = 0.6)
    return(plot)
}

# charger la carte géographique de San Francisco
map <- get_map("San Francisco", zoom = 12, color = "bw")
```

Visualisation des crimes de catégorie *SUICIDE* et *ARSON*,

```
map_crime(train, c('SUICIDE', 'ARSON'))
```



Ajout des nouvelles variables

Nous ajouterons 4 nouvelles variables,

1. Years: l'année de production du crime,
2. Mont: le mois de production du crime, les valeurs sont de 1 à 2 (*de Janvier à Décembre*),
3. DayOfMonth: jour du mois, les valeurs sont de 1 à 31,
4. Hour: heure de production du crime pendant la journée.

```
add_variables <- function(crime_df) {
    crime_df$Years <- strftime(strptime(crime_df$Dates, "%Y-%m-%d %H:%M:%S"), "%Y")
    crime_df$Month <- strftime(strptime(crime_df$Dates, "%Y-%m-%d %H:%M:%S"), "%m")
    crime_df$DayOfMonth <- strftime(strptime(crime_df$Dates, "%Y-%m-%d %H:%M:%S"), "%d")
    crime_df$Hour <- strftime(strptime(crime_df$Dates, "%Y-%m-%d %H:%M:%S"), "%H")

    return(crime_df)
}

data_train <- add_variables(train)
```

Types de crime

La variable *Category* possède 39 variables, chaque crime a une catégorie unique. Dans la figure au dessous, on voit clairement que la répartition des crimes par catégorie n'est pas uniforme, en ordonnant les catégories par ordre de production, on s'aperçoit que les 10 premières catégories représente 83% des crimes, et les 20 premiers, représentent 97%. C'est à dire, en prenant seulement les crimes avec les 20 premières catégories, on perdra seulement 3% de précision.

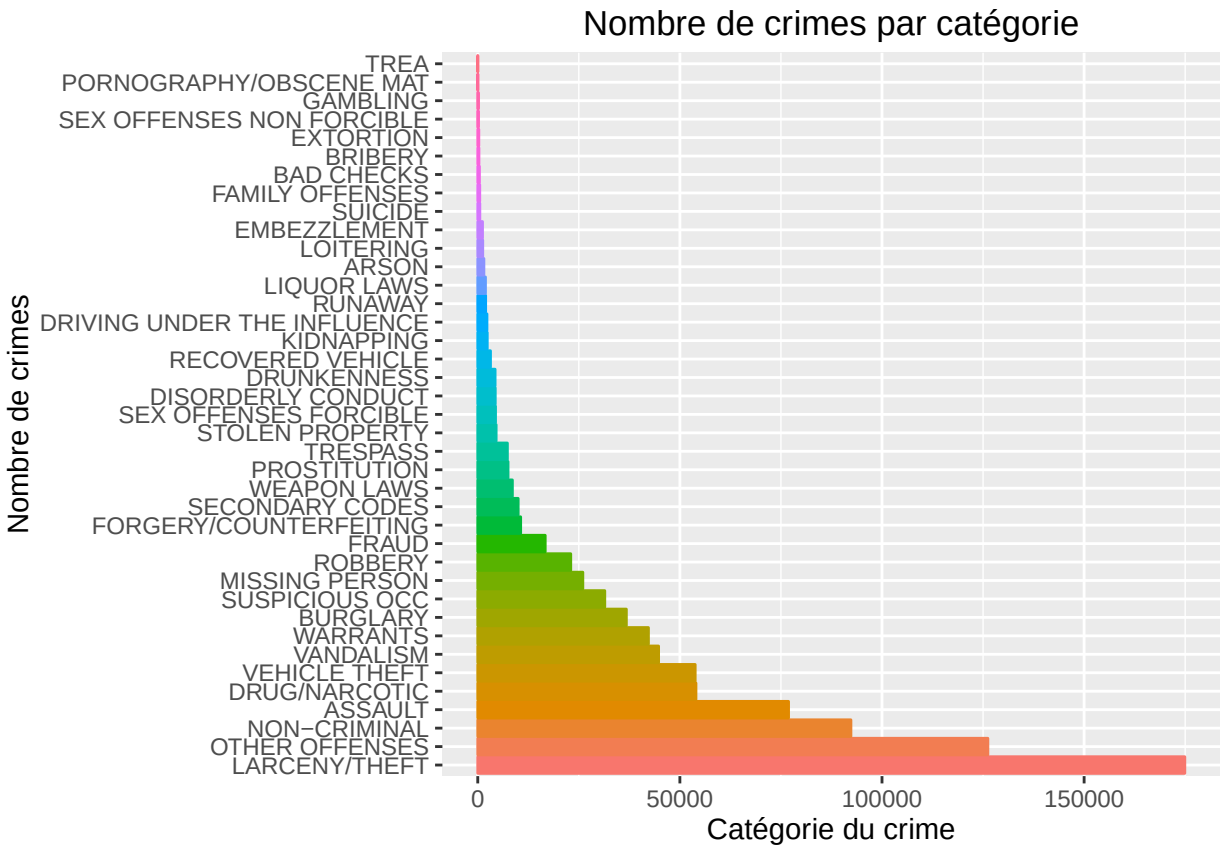
```
type_crime <- data_train %>%
    group_by(Category) %>%
    summarise(count = n()) %>%
    transform(Category = reorder(Category, -count))

top_crimes <- type_crime[with(data = type_crime, order(-count)), ]
top_crimes$percentage <- paste(round(top_crimes$count / sum(top_crimes$count) * 100, 2),
                               "%", sep = " ")
print("Top des crimes")
```

```
## [1] "Top des crimes"
```

```
head(top_crimes, 10)
```

```
##          Category  count percentage
## 17  LARCENY/THEFT 174900    19.92 %
## 22 OTHER OFFENSES 126182    14.37 %
## 21   NON-CRIMINAL  92304    10.51 %
## 2         ASSAULT  76876     8.76 %
## 8   DRUG/NARCOTIC  53971     6.15 %
## 37  VEHICLE THEFT  53781     6.13 %
## 36      VANDALISM  44725     5.09 %
## 38       WARRANTS  42214     4.81 %
## 5        BURGLARY  36755     4.19 %
## 33 SUSPICIOUS OCC  31414     3.58 %
```

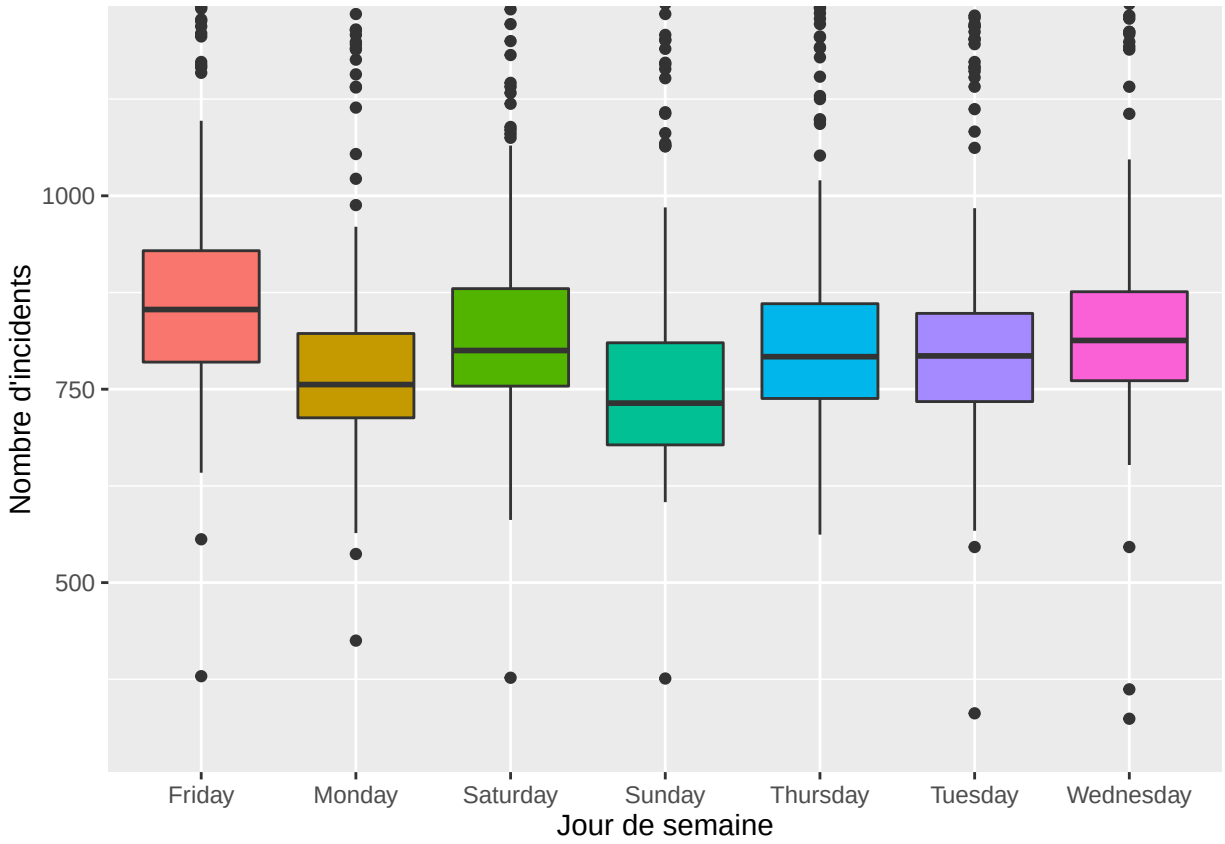
```
ggplot(type_crime) +
    geom_bar(aes(x = Category, y = count,
                 color = Category, fill = Category),
                 stat = "identity") +
    coord_flip() +
    theme(legend.position = "None") +
    ggtitle("Nombre de crimes par catégorie") +
    xlab("Nombre de crimes") +
    ylab("Catégorie du crime")
```



Variations des crimes par jour de semaine

À présent, nous essayerons d'observer la répartition des crimes par les jours de la semaine.

```
data_plot = data_train %>%
        group_by(DayOfWeek, Years, Month) %>%
        summarise(count = n())

ggplot(data = data_plot, aes(x = DayOfWeek, y = count, fill = DayOfWeek)) +
        geom_boxplot() +
        theme(legend.position = "None") +
        xlab("Jour de semaine") +
        ylab("Nombre d'incidents") +
        coord_cartesian(ylim = c(300,1200))
```

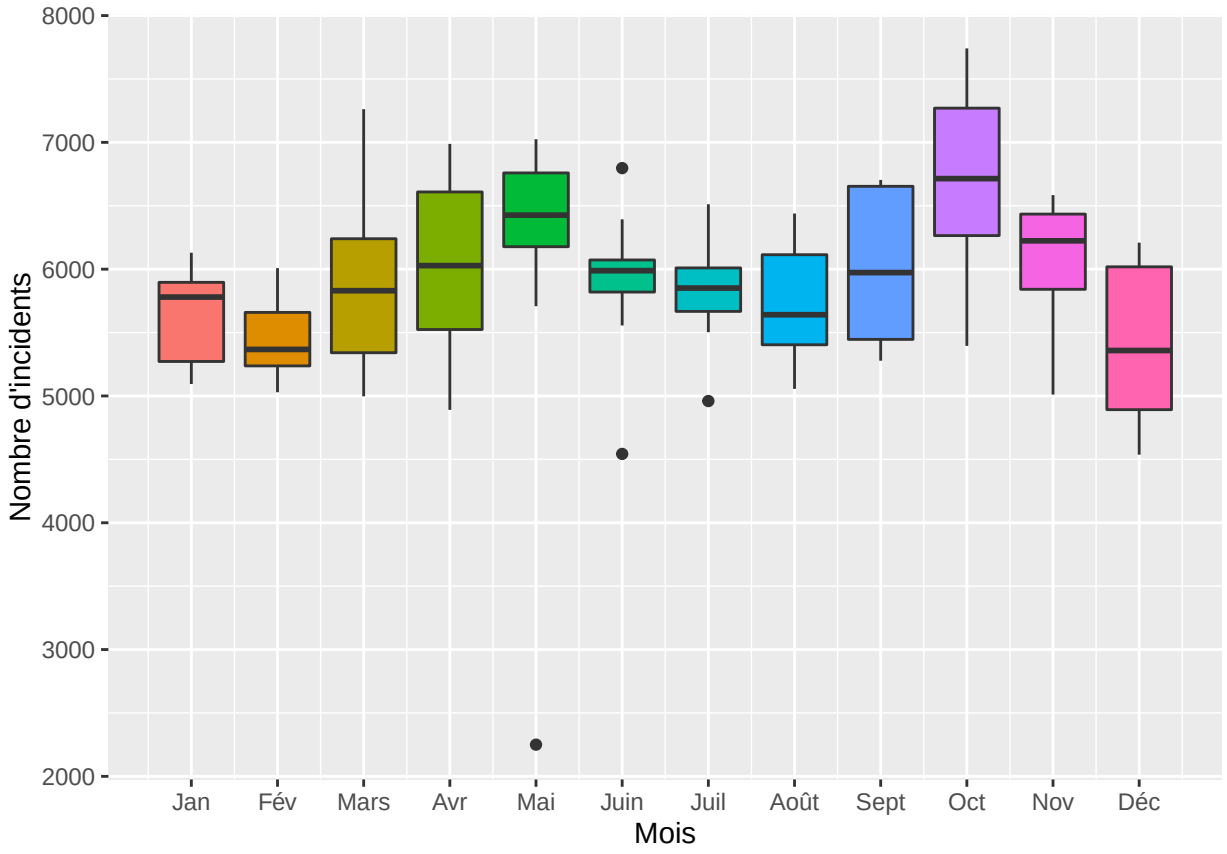


Dans la figure au dessus (boîtes à moustache), on s'aperçoit que le Vendredi, y a une hausse dans le nombre d'incidents des crimes, contrairement au Dimanche, où on remarque une baisse.

Variations des crimes par mois de l'année

On s'aperçoit clairement que le nombre d'incidents de crimes atteint des pics dans les mois d'Octobre et Mai, mais des baisses en Février et Décembre.

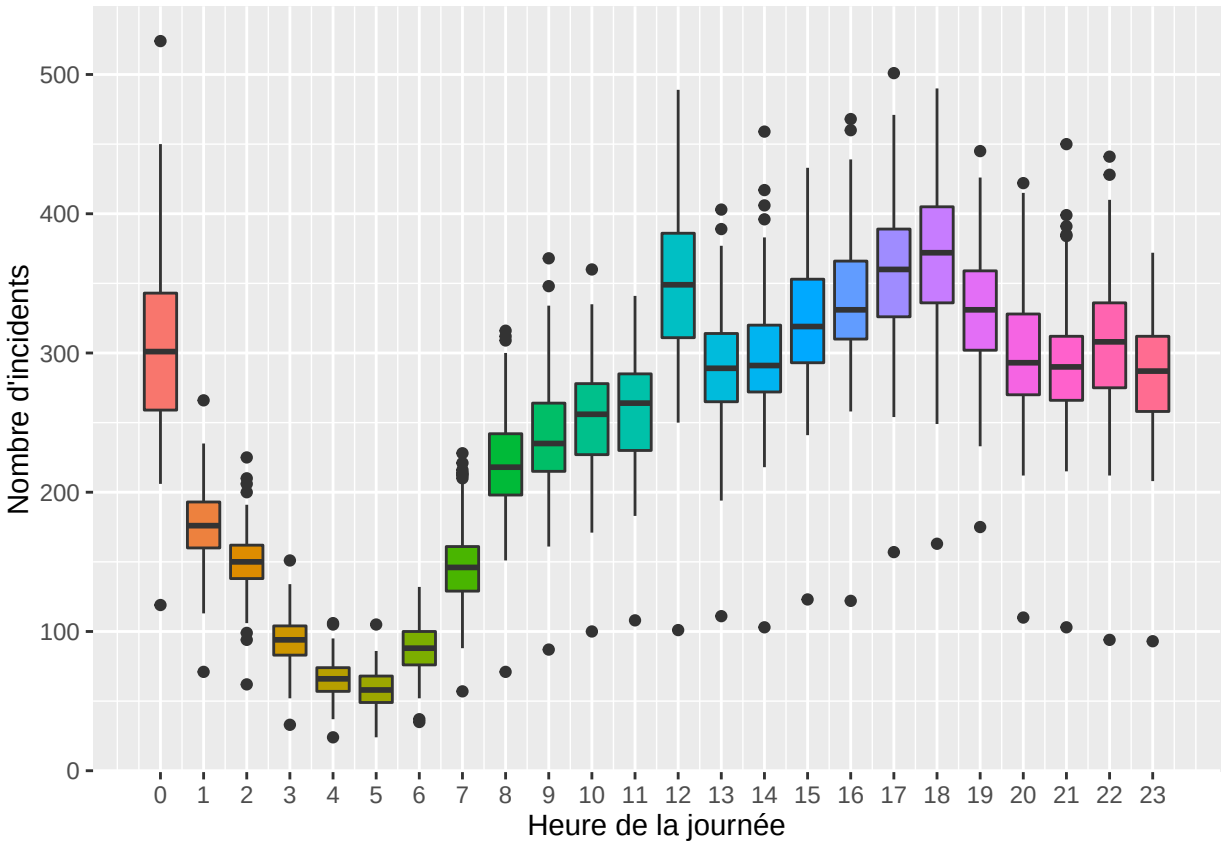
```
data_plot = data_train %>%
        group_by(Month, Years, Month) %>%
        summarise(count = n())

months_name <- c("Jan", "Fév", "Mars", "Avr", "Mai", "Juin",
                 "Juil", "Août", "Sept", "Oct", "Nov", "Déc")

ggplot(data = data_plot, aes(x = as.numeric(Month), y = count, fill = Month)) +
        geom_boxplot() +
        theme(legend.position = "None") +
        xlab("Mois") +
        ylab("Nombre d'incidents") +
        scale_x_continuous(breaks = 1:12, labels = months_name)
```



Variations des crimes par heure de journée

Concernant les incidents de crimes pendant une journée, on remarque qu'entre minuit et 5h du matin, il y a une forte réduction des crimes. Entre 5h du matin et midi, une augmentation du crime, puis une stabilisation du nombre d'incidents jusqu'à minuit.

```
data_plot = data_train %>%
        group_by(Hour, Years, Month) %>%
        summarise(count = n())

ggplot(data = data_plot, aes(x = as.numeric(Hour), y = count, fill = Hour)) +
        geom_boxplot() +
        theme(legend.position = "None") +
        xlab("Heure de la journée") +
        ylab("Nombre d'incidents") +
        scale_x_continuous(breaks = 0:23)
```


Nombre d'incidents
500
400
300
200
100
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Heure de la journée