Projet

Jean Guirguis (11290393) et David Gascon (xxxxxxx)

xx/12/2020

Contents

Exploration des données

```
# Importation des données

train=read.csv("data/churn-bigml-80.csv")
test=read.csv("data/churn-bigml-20.csv")
```

```
summary(train)
```

```
##     State           Account.length    Area.code     International.plan
##  Length:2666        Min.   :  1.0   Min.   :408.0   Length:2666       
##  Class  :character  1st Qu.: 73.0   1st Qu.:408.0   Class  :character 
##  Mode   :character  Median :100.0   Median :415.0   Mode   :character 
##                     Mean   :100.6   Mean   :437.4                     
##                     3rd Qu.:127.0   3rd Qu.:510.0                     
##                     Max.   :243.0   Max.   :510.0                     
##  Voice.mail.plan    Number.vmail.messages Total.day.minutes Total.day.calls
##  Length:2666        Min.   : 0.000        Min.   :  0.0     Min.   :  0.0  
##  Class  :character  1st Qu.: 0.000        1st Qu.:143.4     1st Qu.: 87.0  
##  Mode   :character  Median : 0.000        Median :179.9     Median :101.0  
##                     Mean   : 8.022        Mean   :179.5     Mean   :100.3  
##                     3rd Qu.:19.000        3rd Qu.:215.9     3rd Qu.:114.0  
##                     Max.   :50.000        Max.   :350.8     Max.   :160.0  
##  Total.day.charge Total.eve.minutes Total.eve.calls Total.eve.charge
##  Min.   : 0.00    Min.   :  0.0     Min.   :  0     Min.   : 0.00   
##  1st Qu.:24.38    1st Qu.:165.3     1st Qu.: 87     1st Qu.:14.05   
##  Median :30.59    Median :200.9     Median :100     Median :17.08   
##  Mean   :30.51    Mean   :200.4     Mean   :100     Mean   :17.03
##  3rd Qu.:36.70    3rd Qu.:235.1     3rd Qu.:114     3rd Qu.:19.98
##  Max.   :59.64    Max.   :363.7     Max.   :170     Max.   :30.91
##  Total.night.minutes Total.night.calls Total.night.charge Total.intl.minutes
##  Min.   : 43.7       Min.   : 33.0     Min.   : 1.970     Min.   : 0.00
##  1st Qu.:166.9       1st Qu.: 87.0     1st Qu.: 7.513     1st Qu.: 8.50
##  Median :201.2       Median :100.0     Median : 9.050     Median :10.20
##  Mean   :201.2       Mean   :100.1     Mean   : 9.053     Mean   :10.24
##  3rd Qu.:236.5       3rd Qu.:113.0     3rd Qu.:10.640     3rd Qu.:12.10
##  Max.   :395.0       Max.   :166.0     Max.   :17.770     Max.   :20.00
##  Total.intl.calls Total.intl.charge Customer.service.calls    Churn
##  Min.   : 0.000   Min.   :0.000     Min.   :0.000          Length:2666
##  1st Qu.: 3.000   1st Qu.:2.300     1st Qu.:1.000          Class :character
##  Median : 4.000   Median :2.750     Median :1.000          Mode  :character
##  Mean   : 4.467   Mean   :2.764     Mean   :1.563
##  3rd Qu.: 6.000   3rd Qu.:3.270     3rd Qu.:2.000
##  Max.   :20.000   Max.   :5.400     Max.   :9.000
```

Réencodage des variables "State", "International.plan" et "Voice.mail.plan" en facteur. La variable d'intérêt est réencodée en variable logique, à savoir que les valeurs "vrai" sont codées "1" et les valeurs "faux" sont codées en 0

```
### TRAIN ####
#transformation des variables "character" en "facteur"

train$State=as.factor(train$State)
train$International.plan=as.factor(train$International.plan)
train$Voice.mail.plan=as.factor(train$Voice.mail.plan)

#transformation variable d'intérêt en variable logique
train$Churn=as.logical(train$Churn)

### TEST ####
#transformation des variables "character" en "facteur"

test$State=as.factor(test$State)
test$International.plan=as.factor(test$International.plan)
test$Voice.mail.plan=as.factor(test$Voice.mail.plan)

#transformation variable d'intérêt en variable logique
test$Churn=as.integer(test$Churn)
```

```
## Warning: NAs introduits lors de la conversion automatique
```

Étude descriptive des données

```
# fonction
analyse_table = function (nom,variable,nb_donne)
{
  table_temporaire= table(variable)
  table_temporaire = as.data.frame(table_temporaire)
  table_temporaire = data.frame(table_temporaire,pourcentage=round(table_temporaire[2]/nb_donne*100, dig
  names(table_temporaire)[3] = "% Freq"
  names(table_temporaire)[1] = nom
  #table_temporaire = head(table_temporaire[order(-table_temporaire[3]),],3)
  table_temporaire = table_temporaire[order(-table_temporaire[3]),]
  return(table_temporaire)
}

analyse_table('State',train$State,nrow(train))
```

```
##    State Freq % Freq
## 50    WV   88   3.30
## 24    MN   70   2.63
## 35    NY   68   2.55
## 46    VA   67   2.51
## 2     AL   66   2.48
## 36    OH   66   2.48
## 51    WY   66   2.48
## 38    OR   62   2.33
## 34    NV   61   2.29
## 49    WI   61   2.29
## 21    MD   60   2.25
## 45    UT   60   2.25
## 6     CO   59   2.21
## 7     CT   59   2.21
## 23    MI   58   2.18
## 47    VT   57   2.14
## 14    ID   56   2.10
## 28    NC   56   2.10
## 44    TX   55   2.06
## 10    FL   54   2.03
## 16    IN   54   2.03
## 27    MT   53   1.99
## 17    KS   52   1.95
## 20    MA   52   1.95
## 37    OK   52   1.95
## 9     DE   51   1.91
## 25    MO   51   1.91
## 32    NJ   50   1.88
## 11    GA   49   1.84
## 22    ME   49   1.84
## 41    SC   49   1.84
## 42    SD   49   1.84
## 26    MS   48   1.80
## 40    RI   48   1.80
## 48    WA   48   1.80
## 3     AR   47   1.76
## 4     AZ   45   1.69
## 8     DC   45   1.69
## 15    IL   45   1.69
## 30    NE   45   1.69
## 12    HI   44   1.65
## 29    ND   44   1.65
## 33    NM   44   1.65
## 1     AK   43   1.61
## 18    KY   43   1.61
## 31    NH   43   1.61
## 43    TN   41   1.54
## 13    IA   38   1.43
## 39    PA   36   1.35
## 19    LA   35   1.31
## 5     CA   24   0.90
```

```
analyse_table('International.plan',train$International.plan,nrow(train))
```

```
##   International.plan Freq % Freq
## 1                 No 2396  89.87
## 2                Yes  270  10.13
```

```
analyse_table('Voice.mail.plan',train$Voice.mail.plan,nrow(train))
```

```
##   Voice.mail.plan Freq % Freq
## 1              No 1933  72.51
## 2             Yes  733  27.49
```

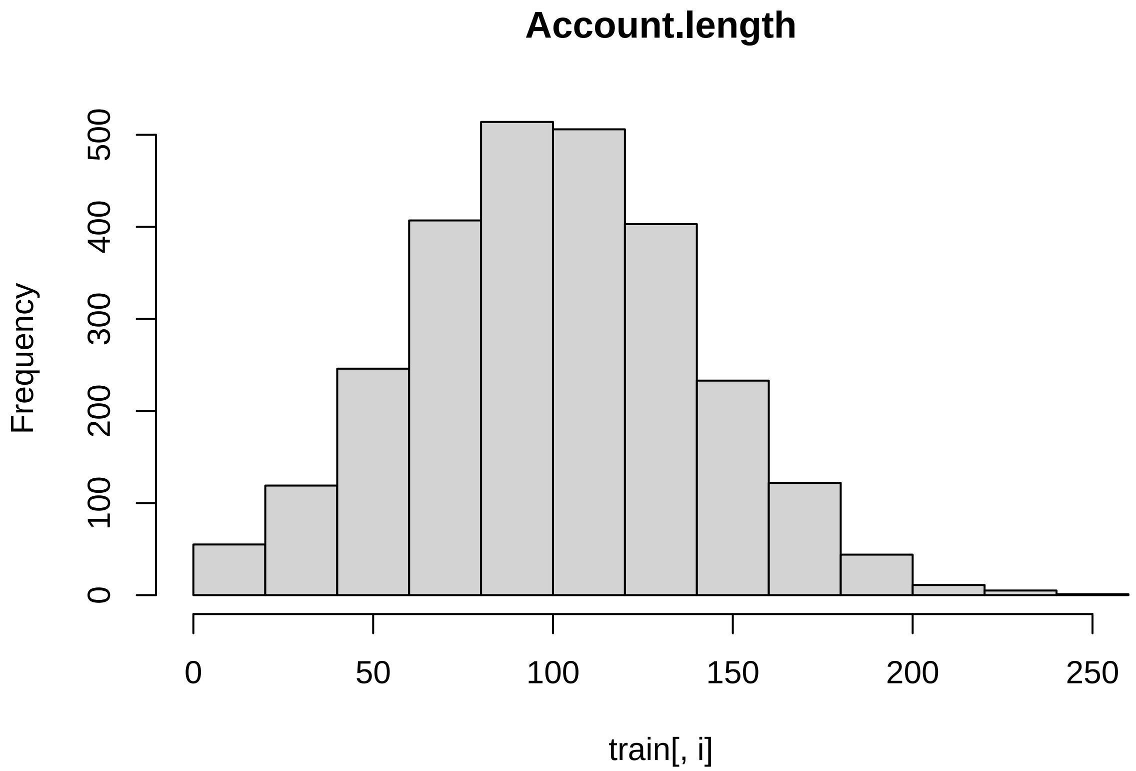
```
analyse_table('Churn',train$Churn,nrow(train))
```

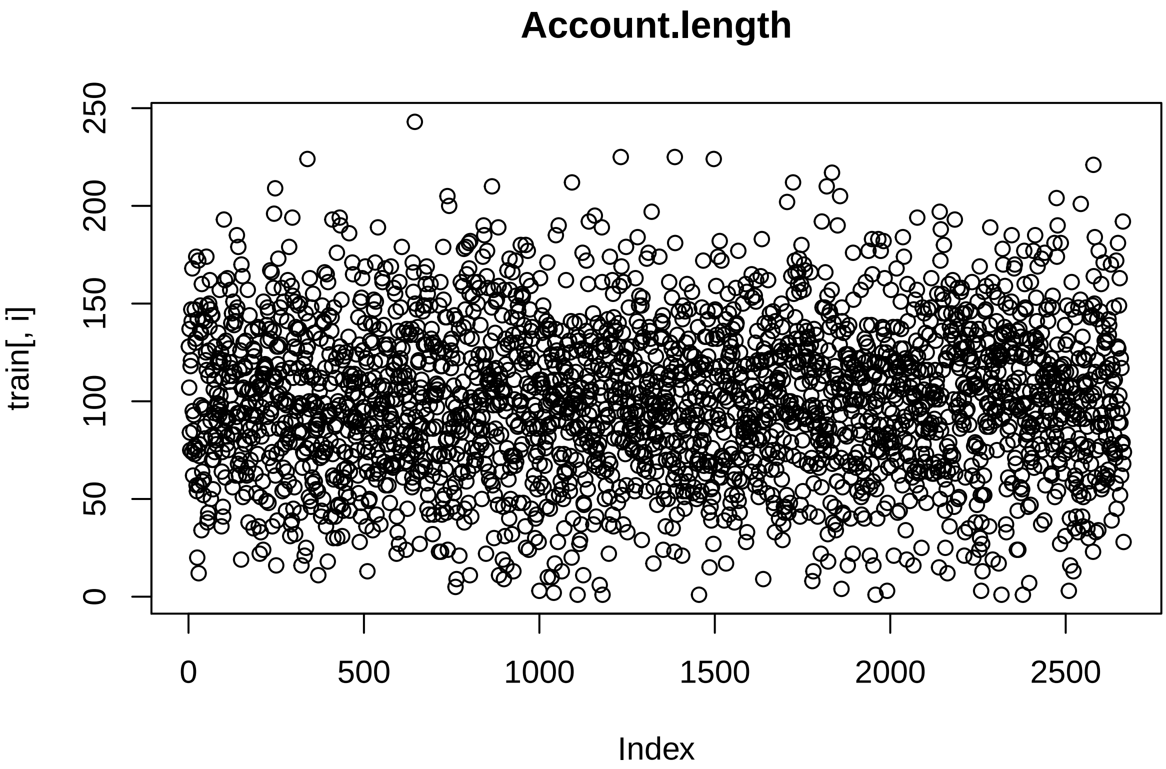
```
##   Churn Freq % Freq
## 1 FALSE 2278  85.45
## 2  TRUE  388  14.55
```

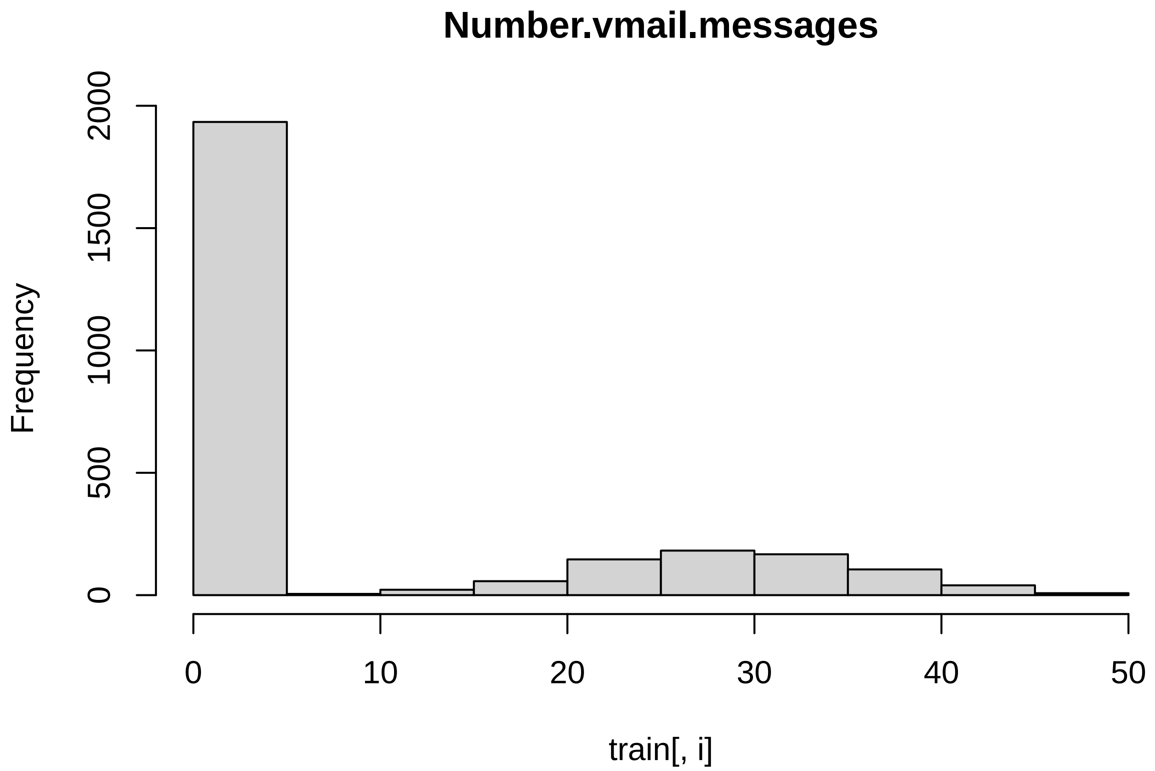
```
analyse_table('Area.code',train$Area.code,nrow(train))
```

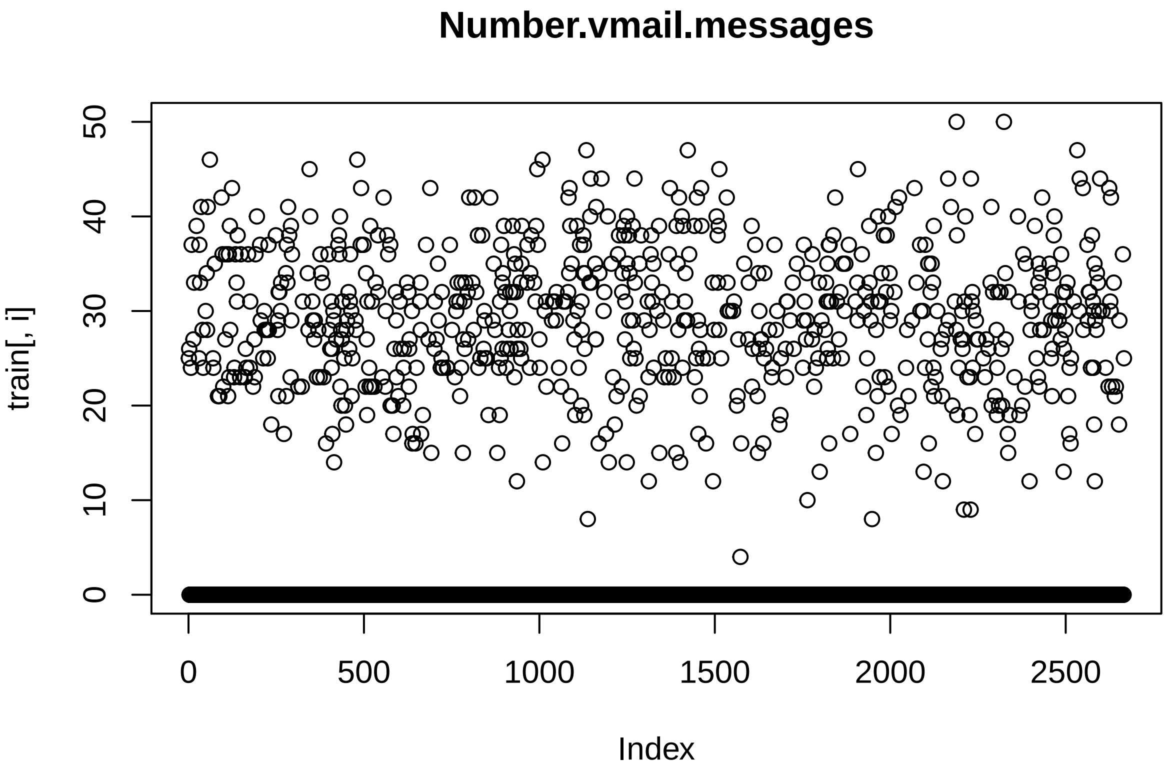
```
##   Area.code Freq % Freq
## 2       415 1318  49.44
## 3       510  679  25.47
## 1       408  669  25.09
```

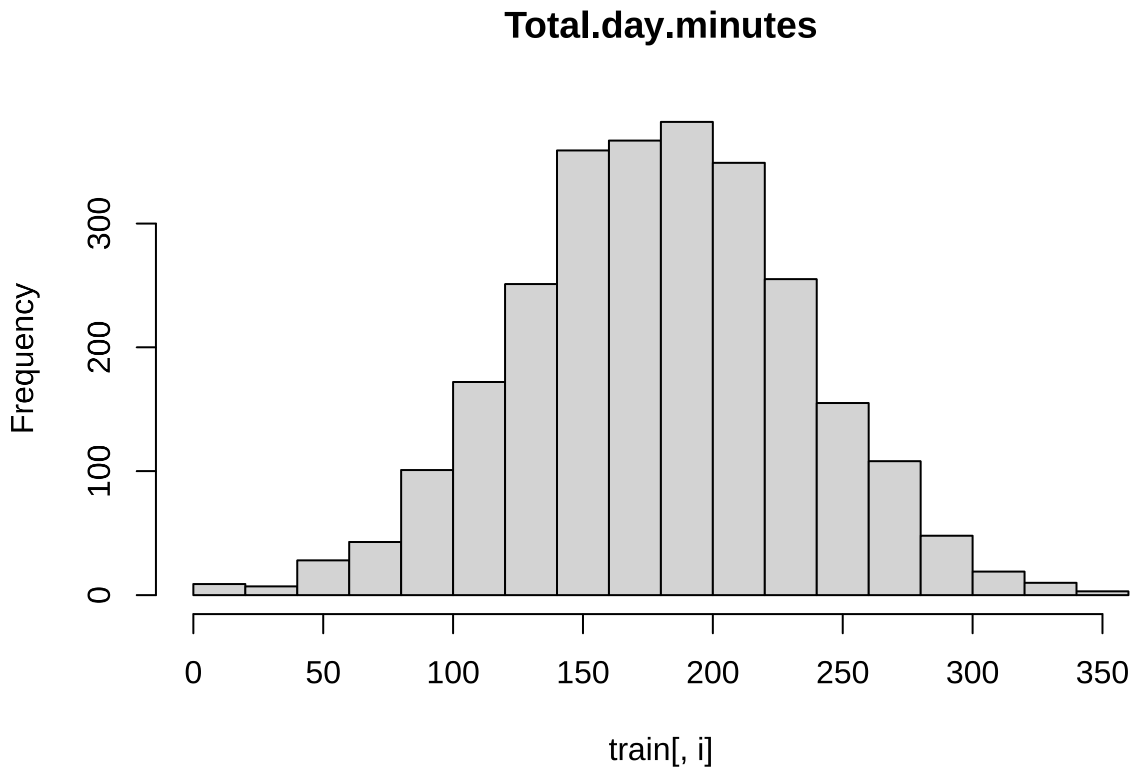
```
for (i in 1:length(colnames(train)))
{
  if (i != 1 & i !=3  & i != 4 & i != 5 & i !=length(colnames(train)))
  {
    hist(train[,i], main = (colnames(train)[i] ))
    plot(train[,i], main = (colnames(train)[i] ))
  }
}
```

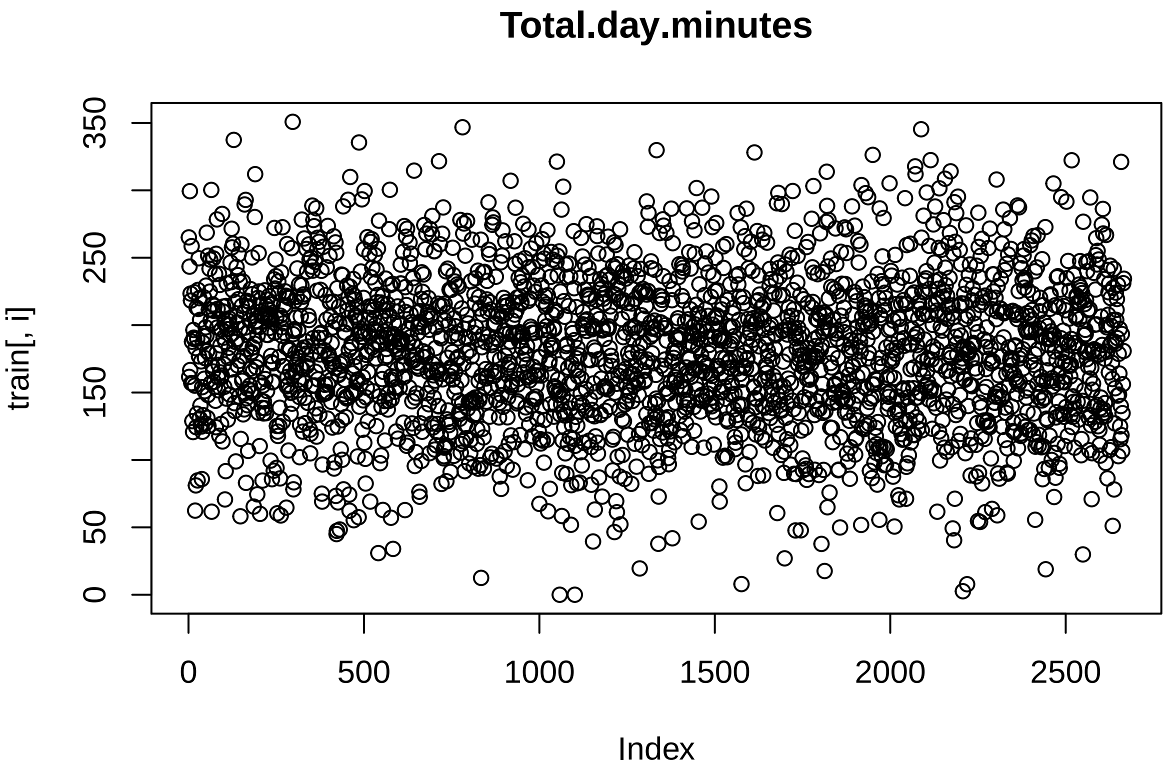

Account.length
Frequency
0
100
200
300
400
500
0
50
100
150
200
250
train[, i]

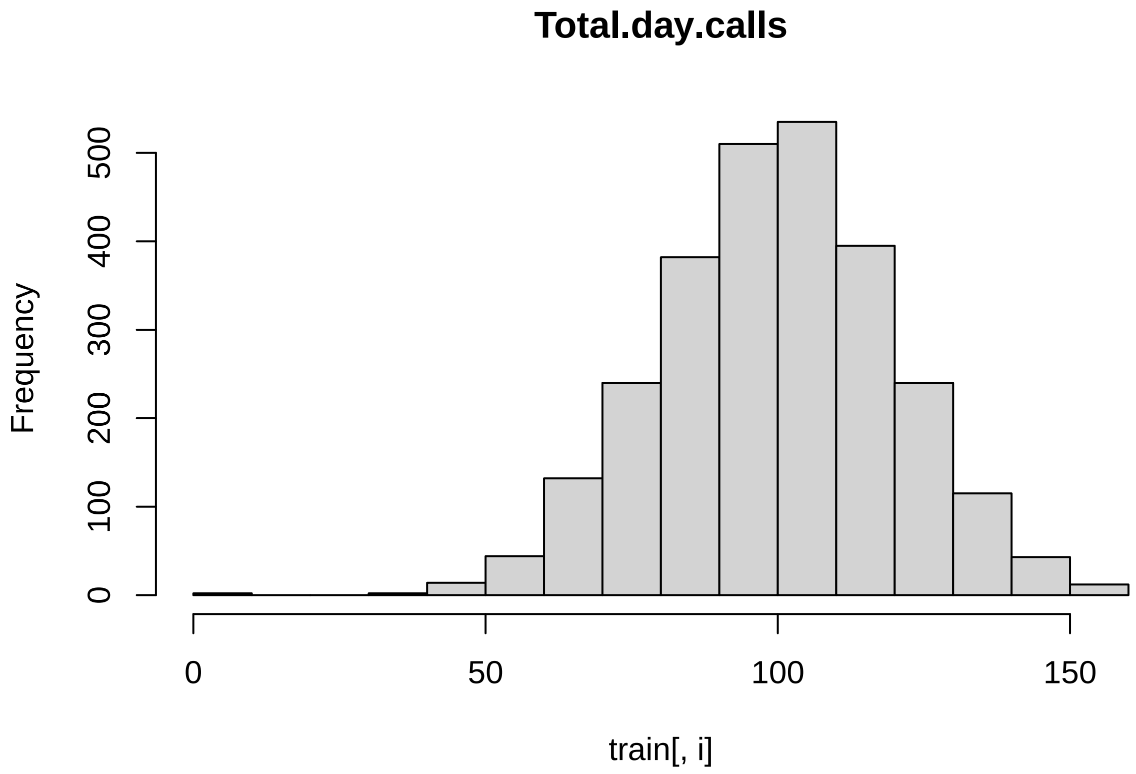

Account.length
train[, i]
0
50
100
150
200
250
Index
0
500
1000
1500
2000
2500

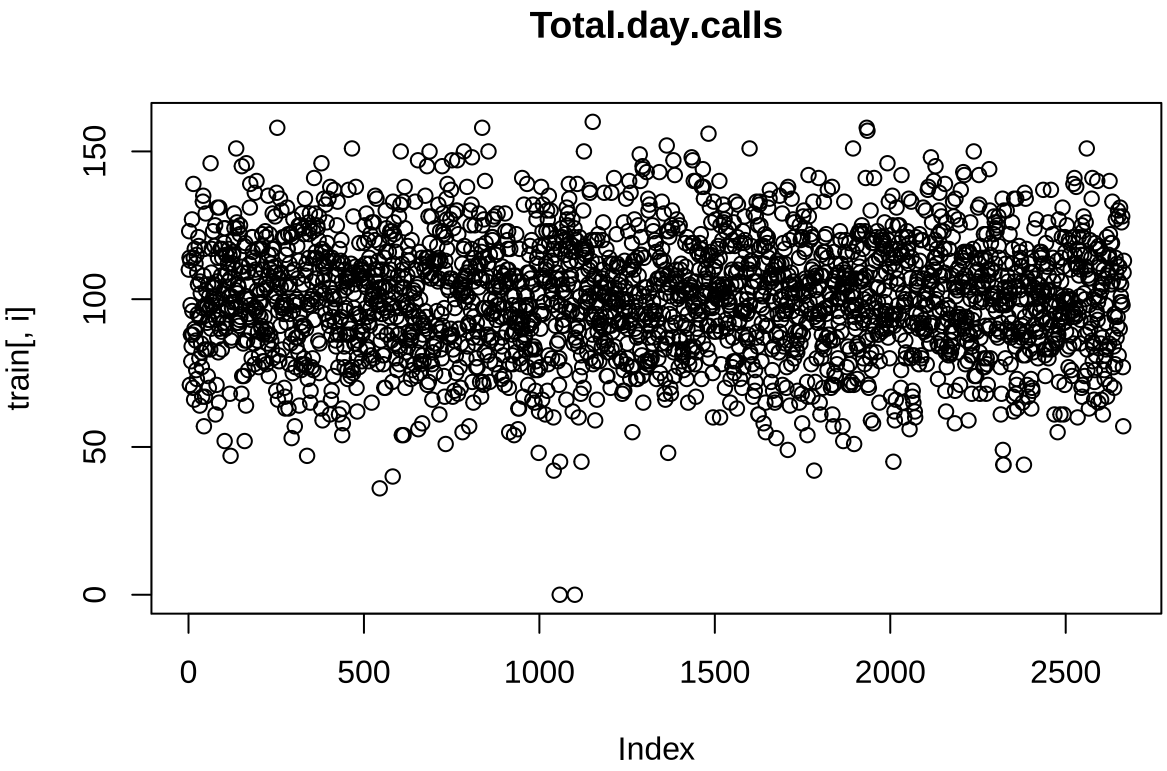

Number.vmail.messages
Frequency
0
500
1000
1500
2000
0
10
20
30
40
50
train[, i]

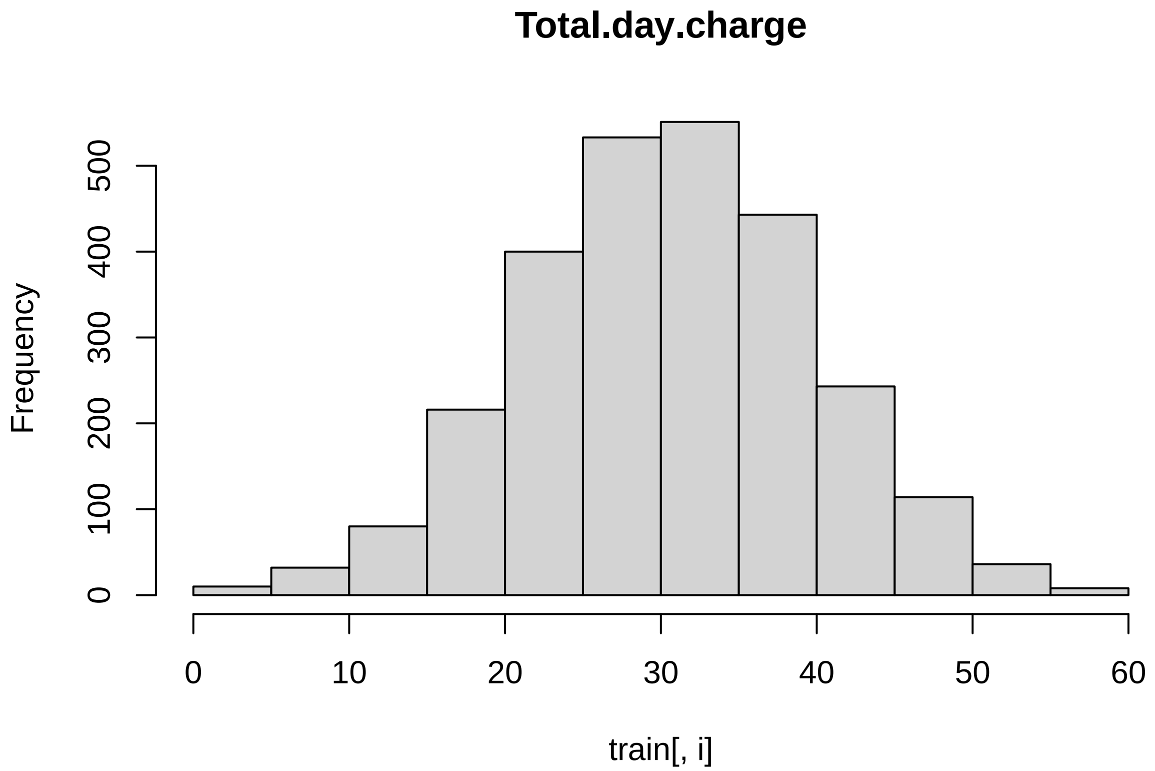

Number.vmail.messages
train[, i]
Index
0
10
20
30
40
50
0
500
1000
1500
2000
2500

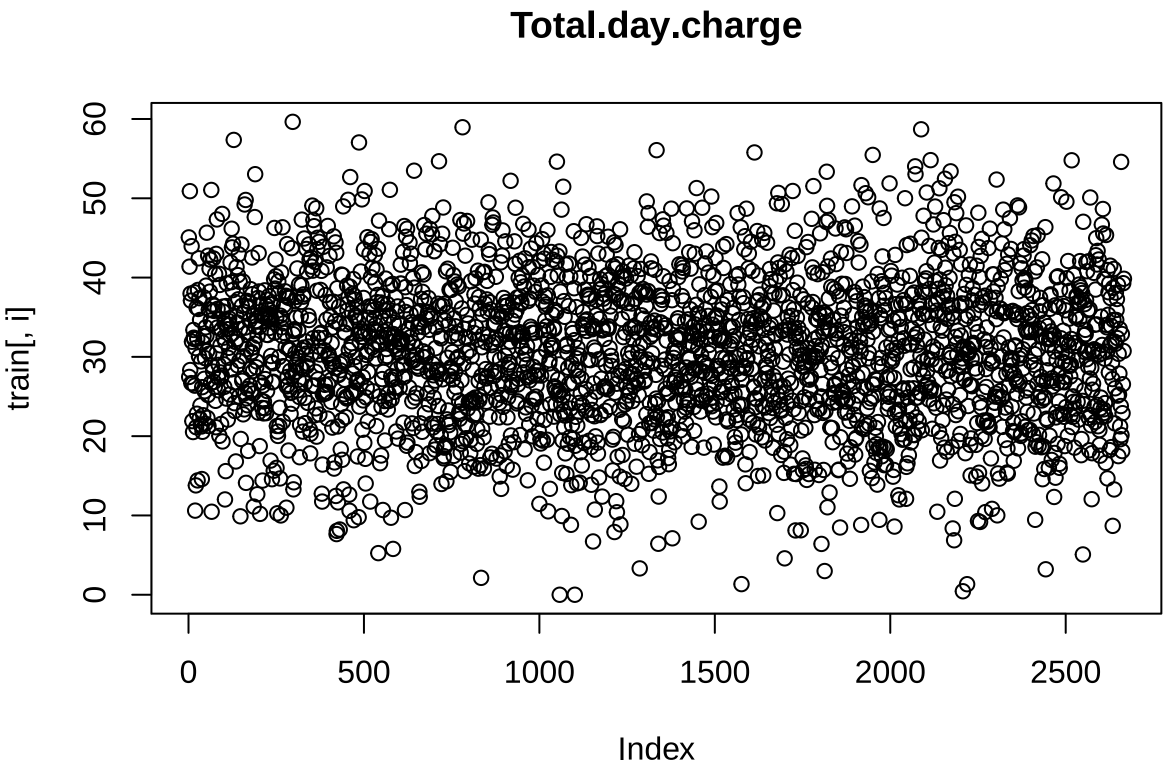

Total.day.minutes
Frequency
0
100
200
300
0
50
100
150
200
250
300
350
train[, i]

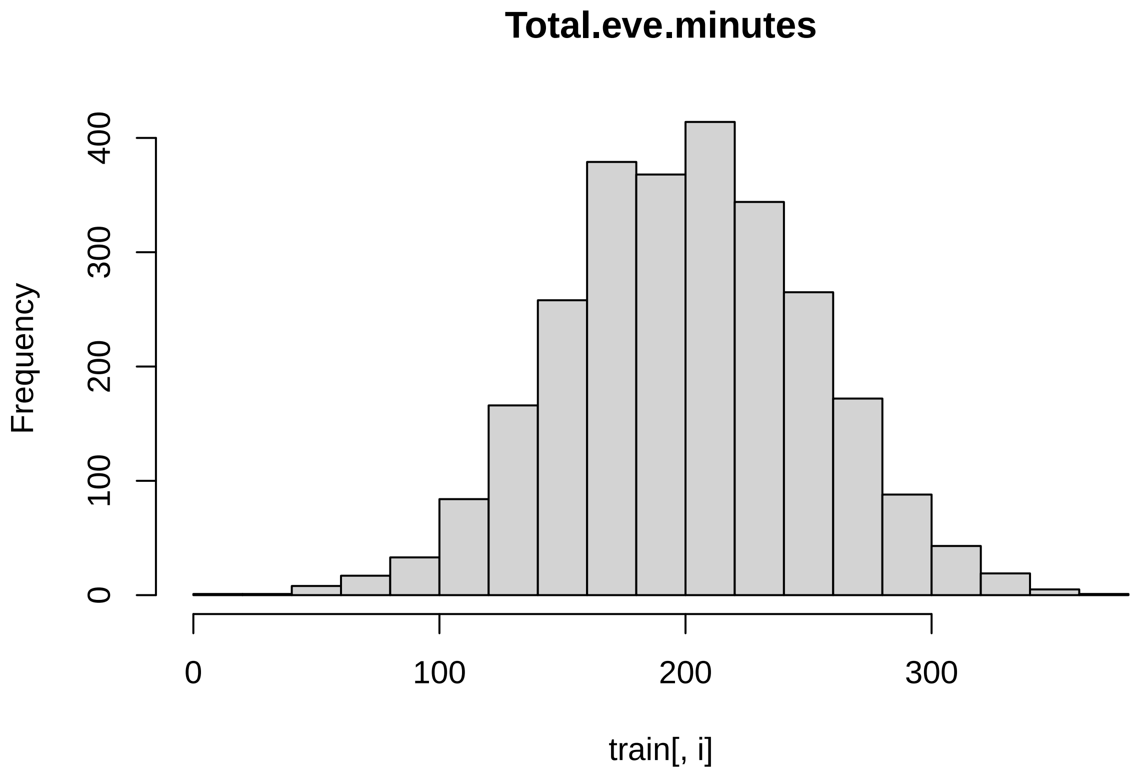

Total.day.minutes
train[, i]
0
50
150
250
350
Index
0
500
1000
1500
2000
2500

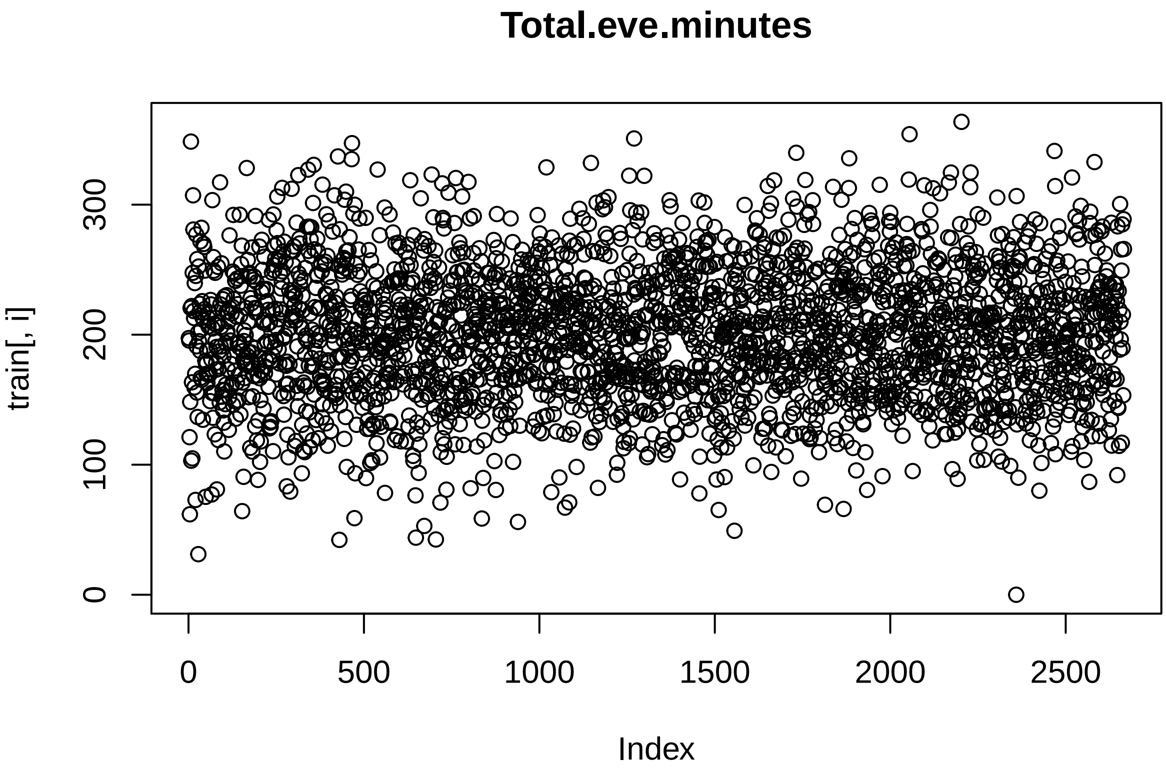

Total.day.calls
Frequency
0
100
200
300
400
500
0
50
100
150
train[, i]

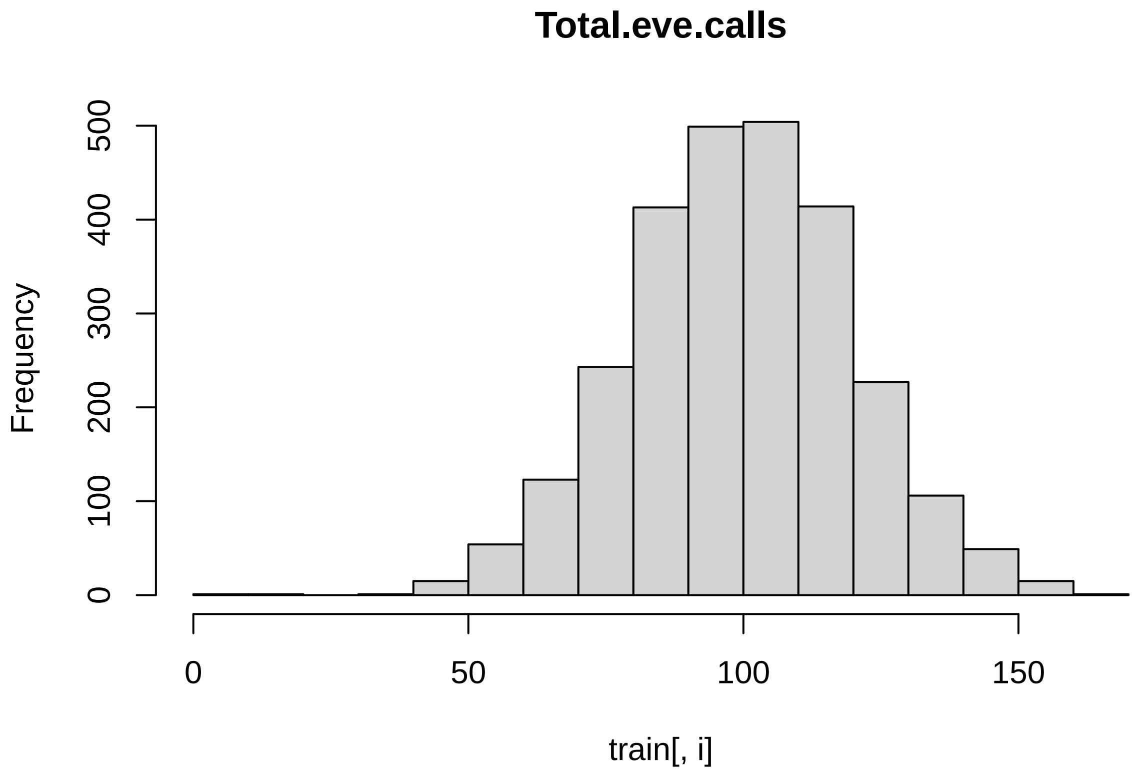

Total.day.calls
train[, i]
0
50
100
150
0
500
1000
1500
2000
2500
Index

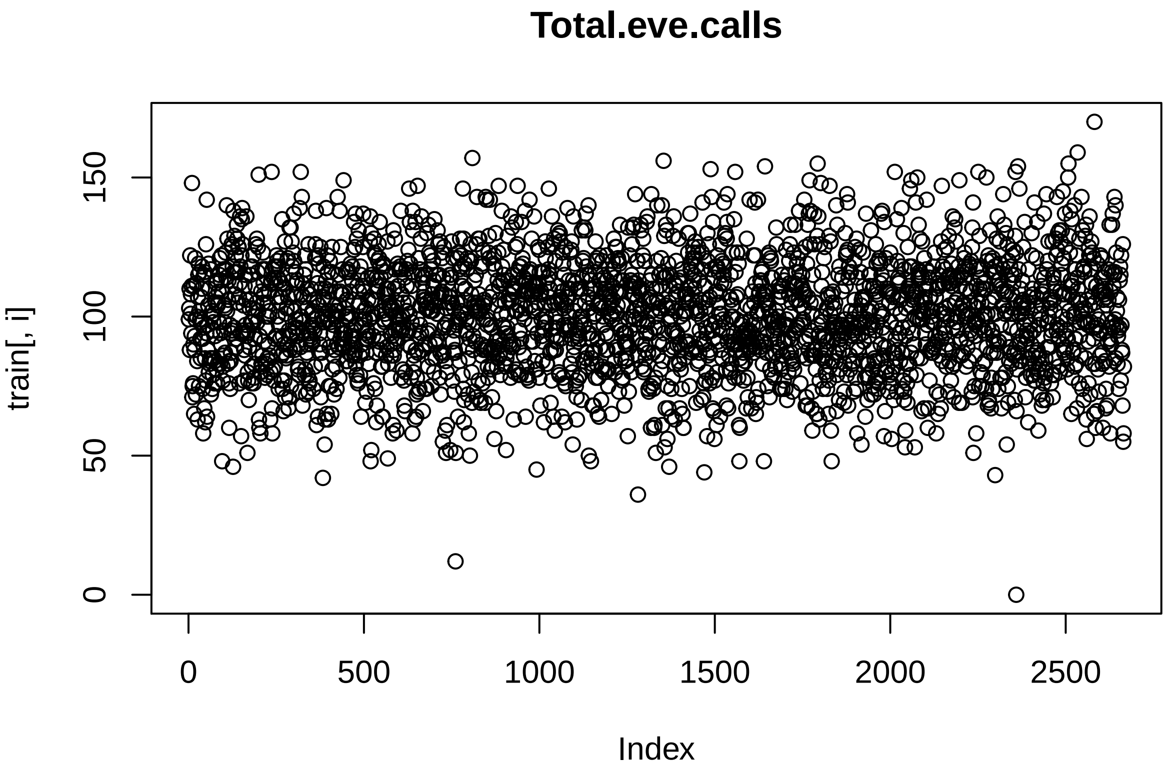

Total.day.charge
Frequency
0
100
200
300
400
500
0
10
20
30
40
50
60
train[, i]

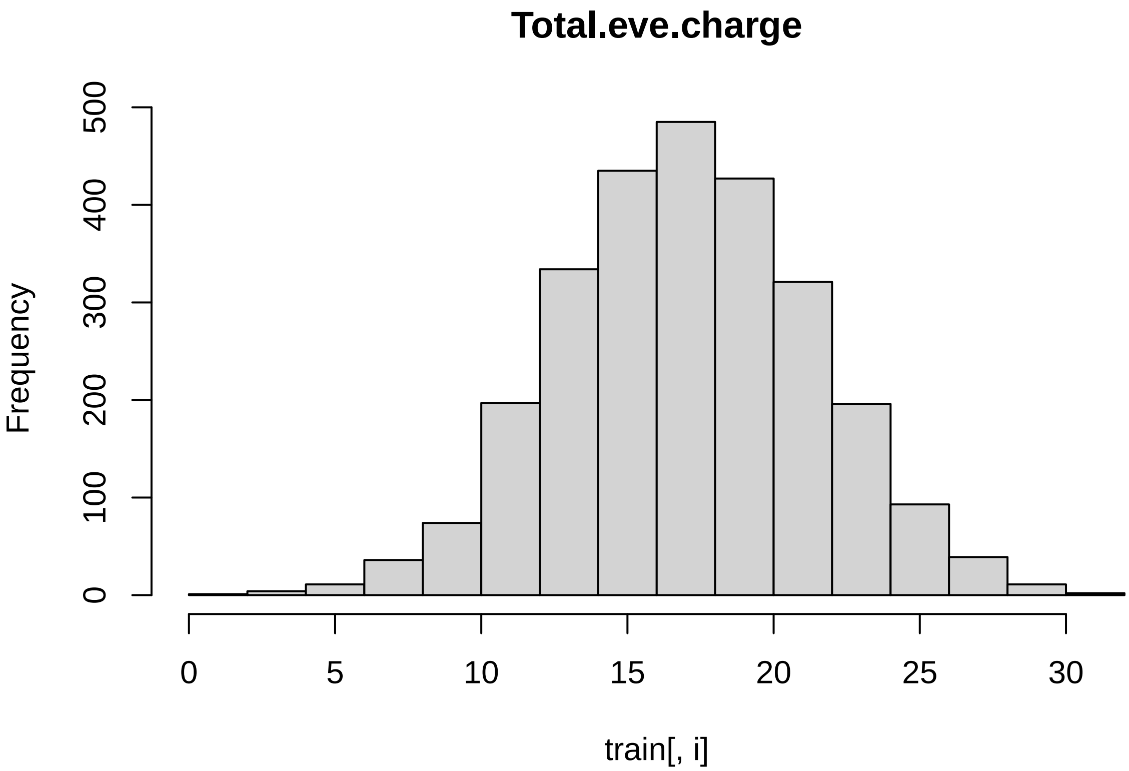

Total.day.charge
train[, i]
Index
0
10
20
30
40
50
60
0
500
1000
1500
2000
2500

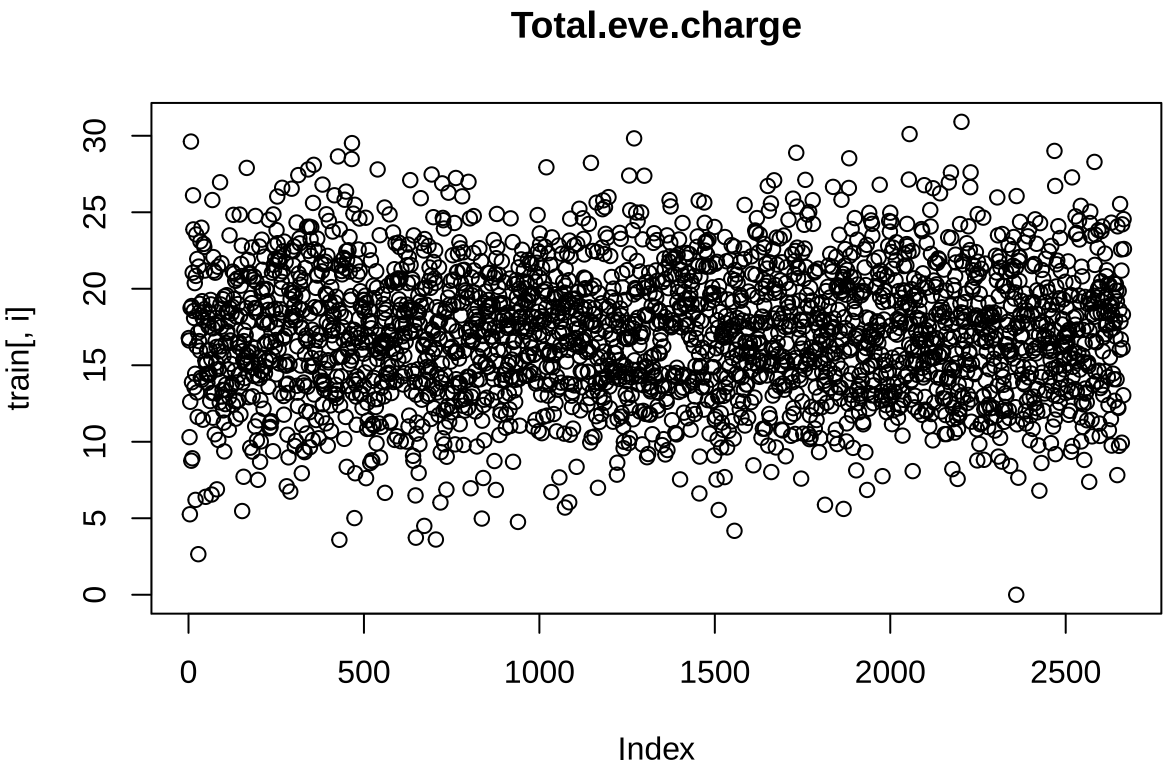

Total.eve.minutes
Frequency
0
100
200
300
400
0
100
200
300
train[, i]

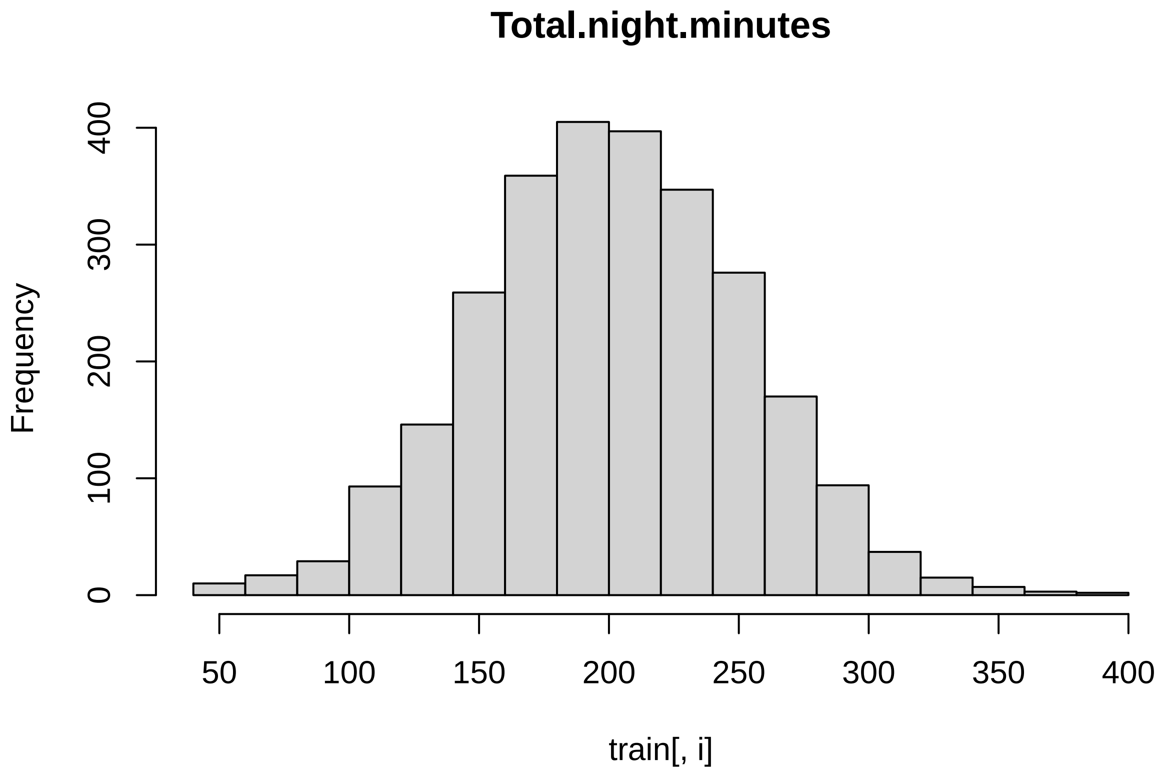

Total.eve.minutes
train[, i]
0
100
200
300
0
500
1000
1500
2000
2500
Index

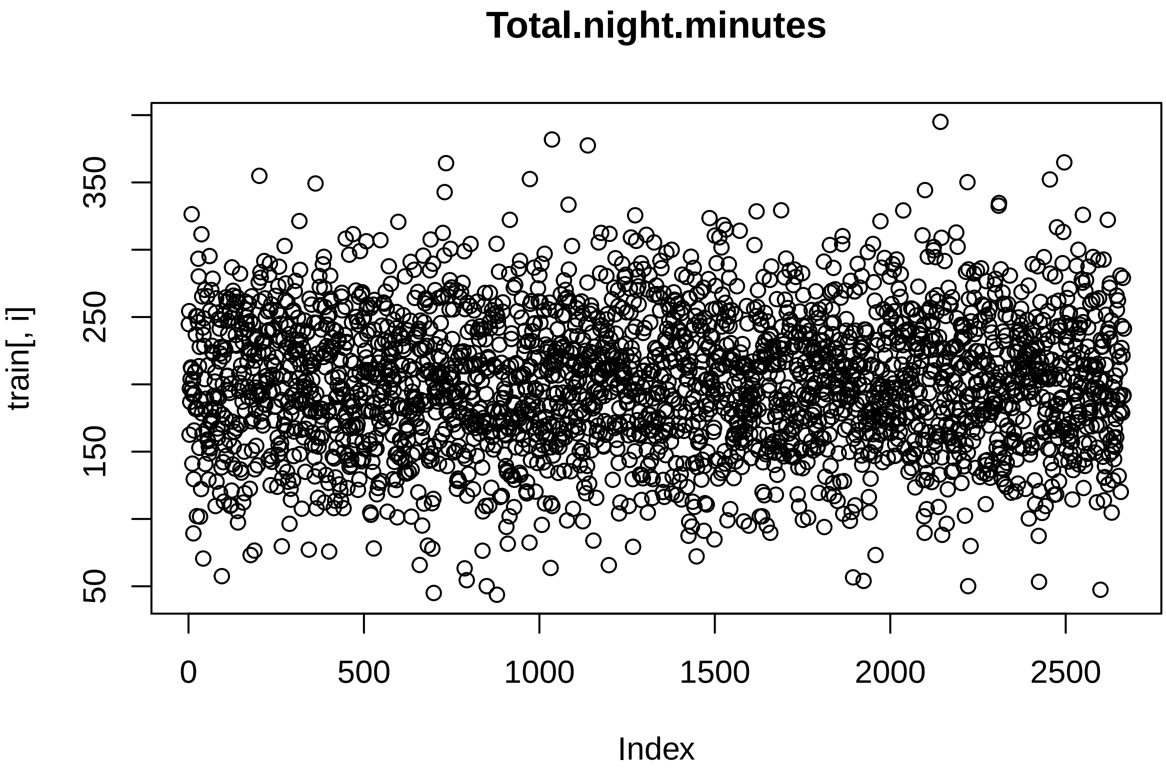

Total.eve.calls
Frequency
0
100
200
300
400
500
0
50
100
150
train[, i]

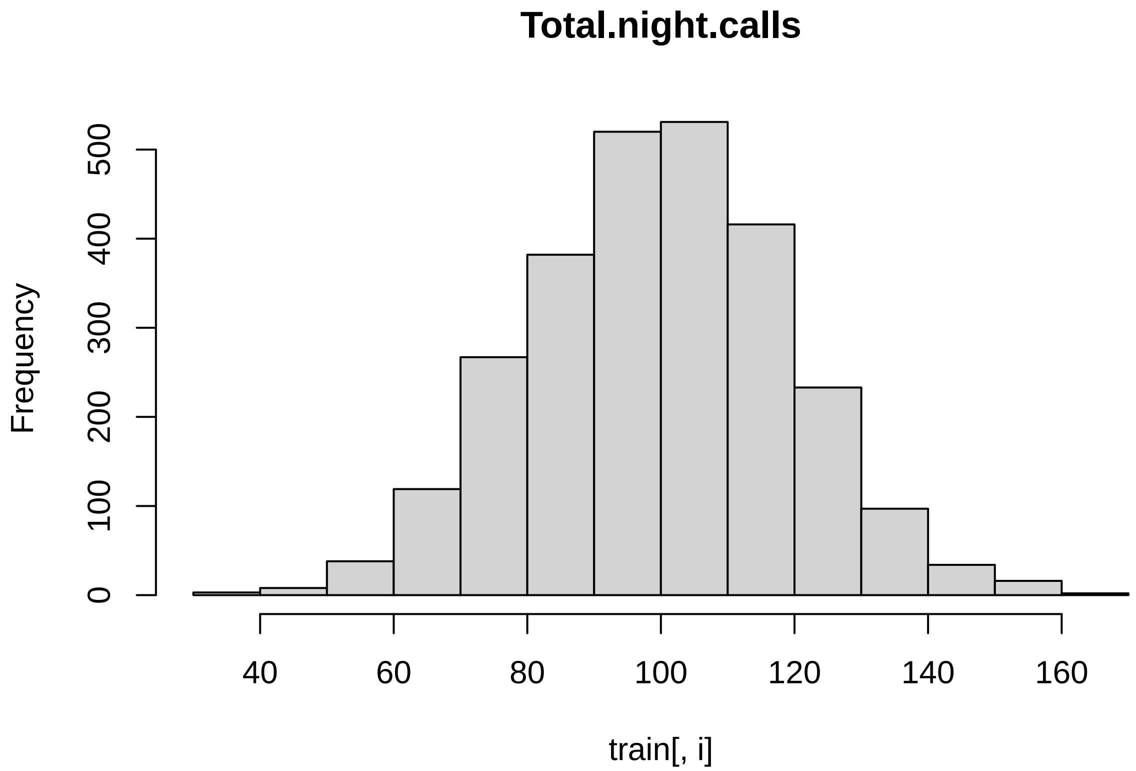

Total.eve.calls
train[, i]
0
50
100
150
Index
0
500
1000
1500
2000
2500

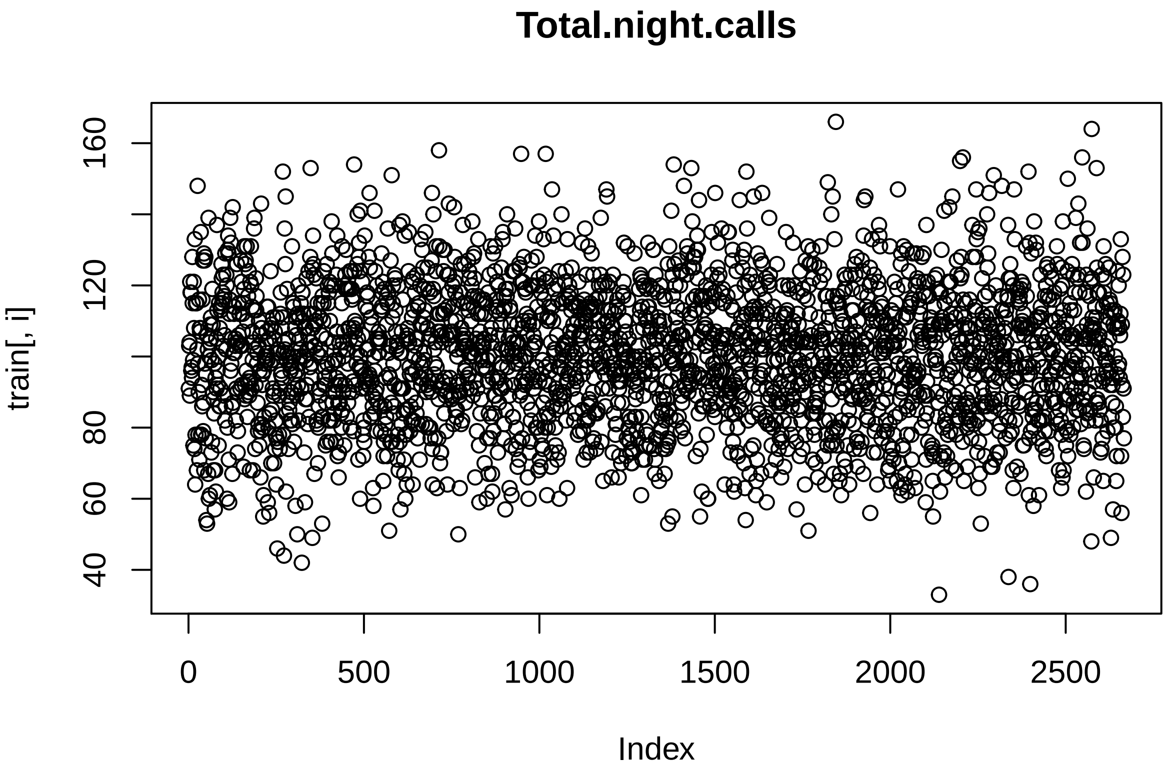

Total.eve.charge
Frequency
0
100
200
300
400
500
0
5
10
15
20
25
30
train[, i]

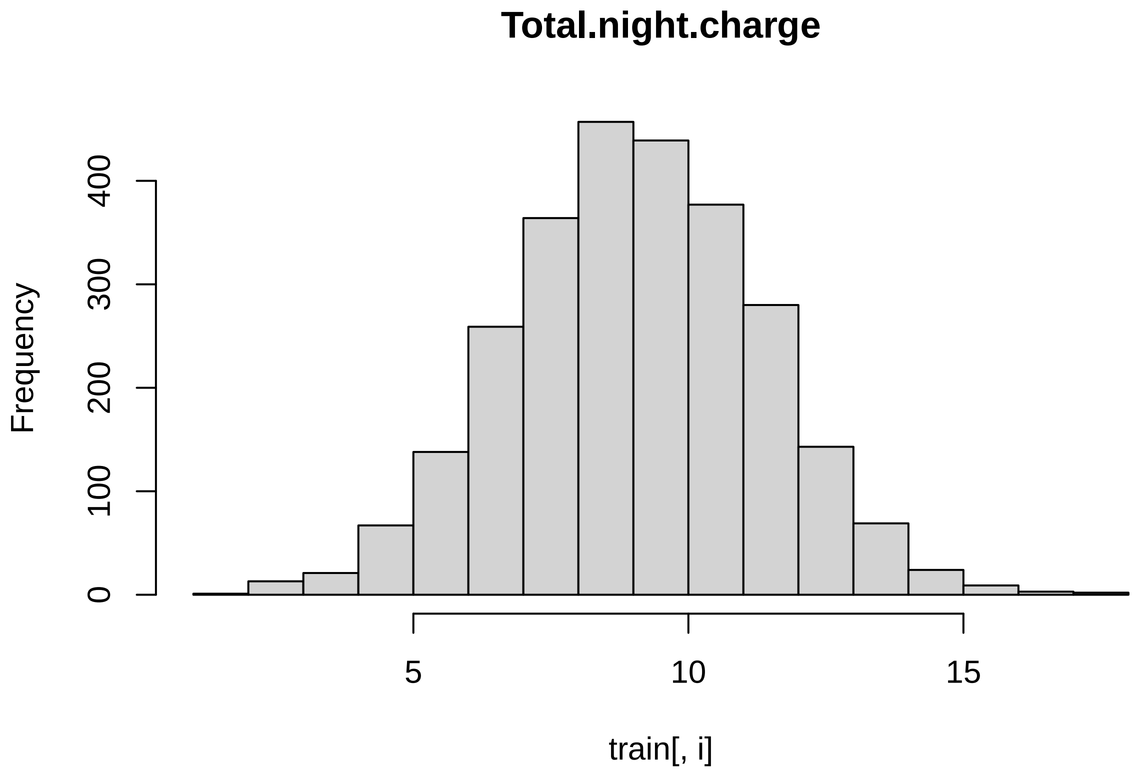

Total.eve.charge
train[, i]
0
5
10
15
20
25
30
0
500
1000
1500
2000
2500
Index

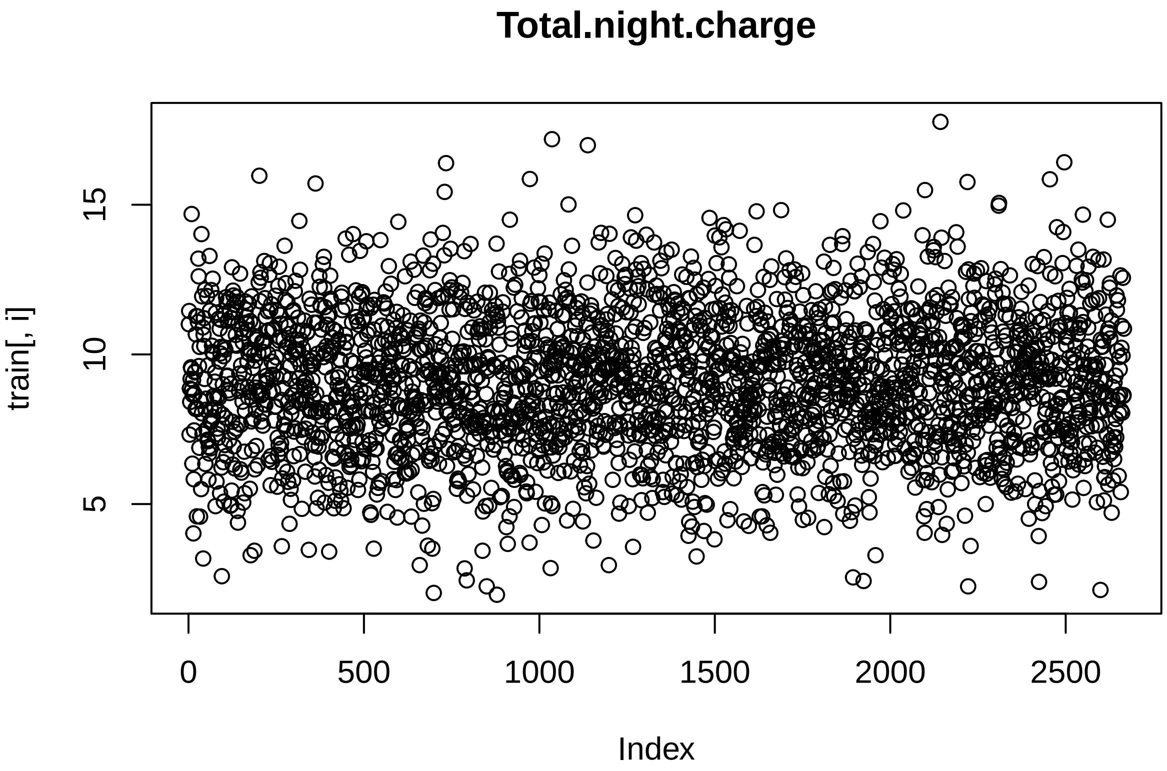

Total.night.minutes
Frequency
0
100
200
300
400
50
100
150
200
250
300
350
400
train[, i]

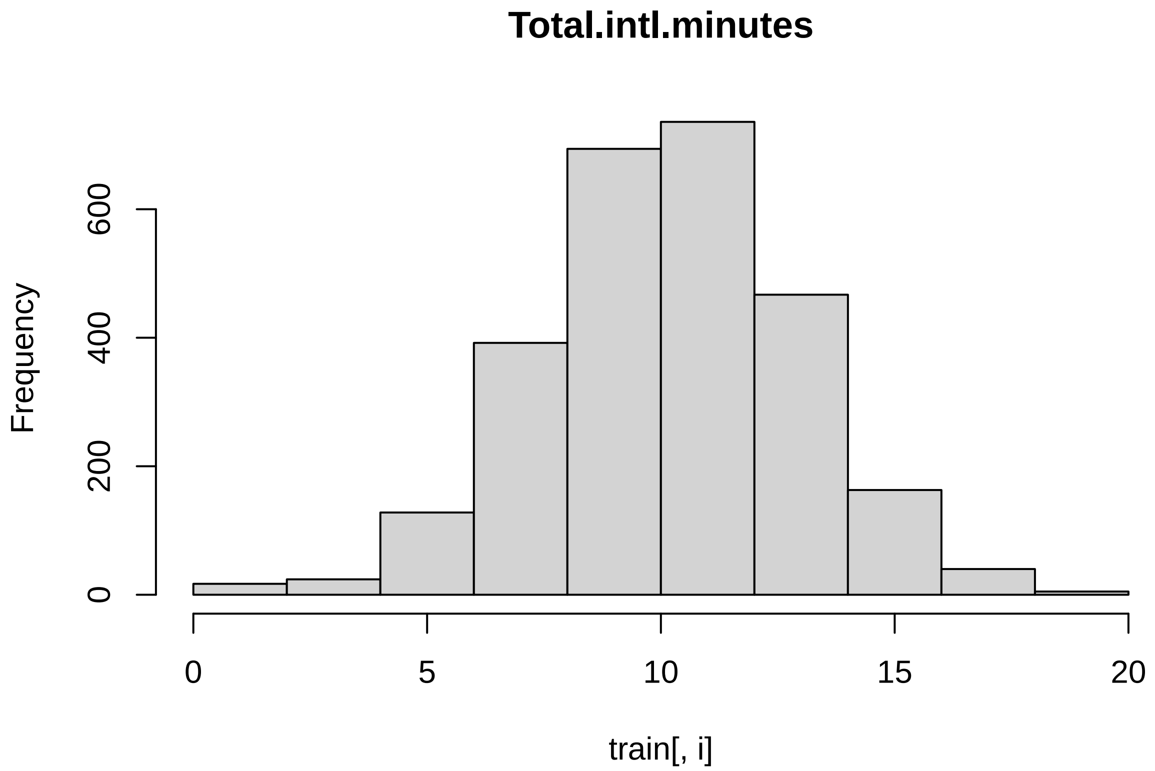

Total.night.minutes
train[, i]
50
150
250
350
Index
0
500
1000
1500
2000
2500

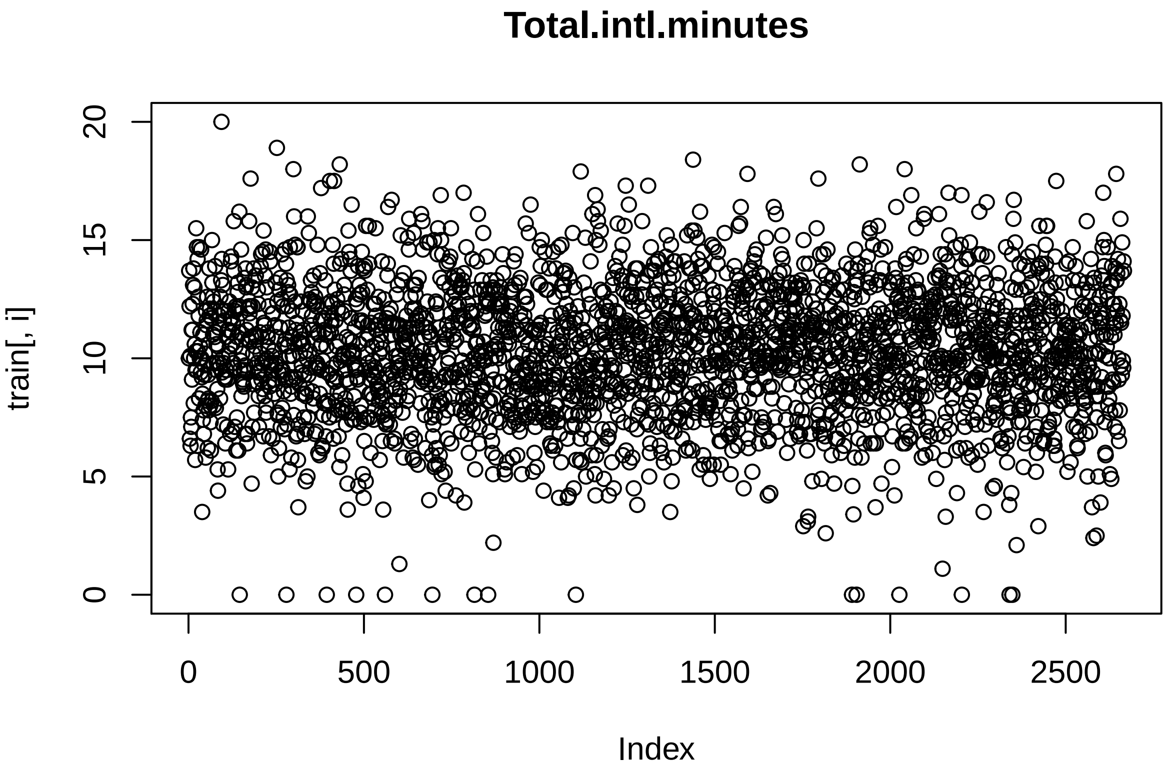

Total.night.calls
Frequency
0
100
200
300
400
500
40
60
80
100
120
140
160
train[, i]

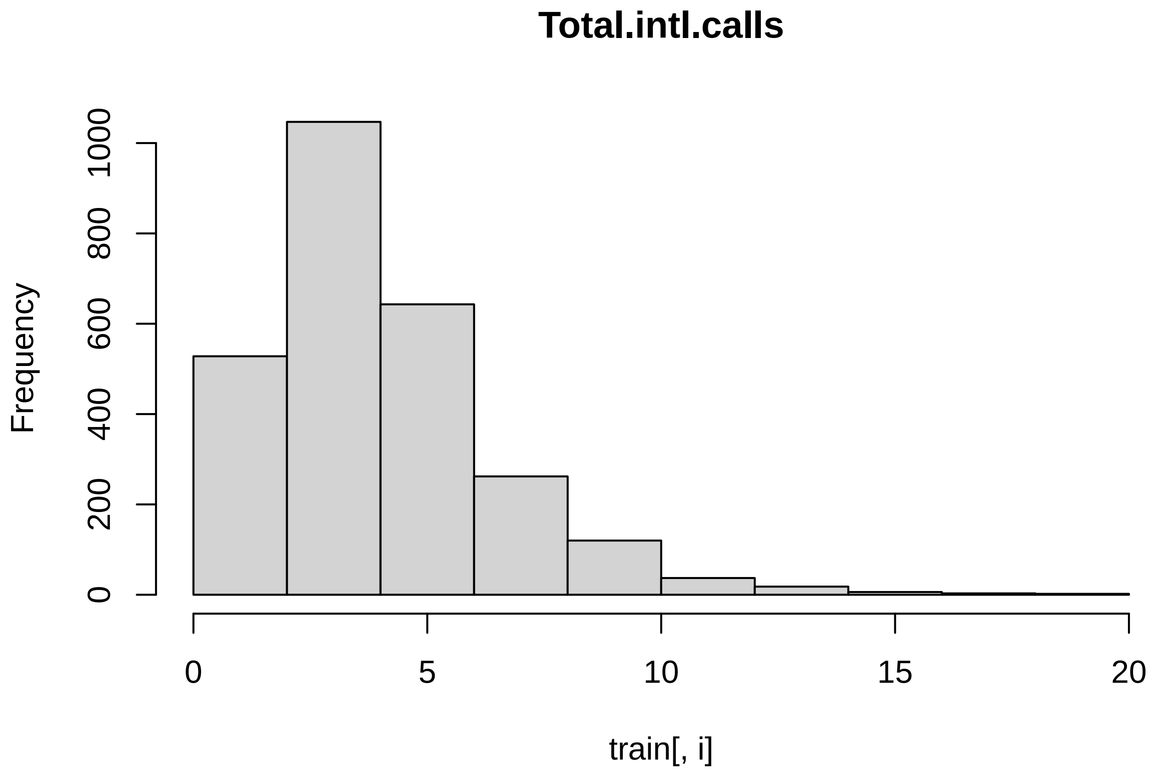

Total.night.calls
train[, i]
40
60
80
120
160
0
500
1000
1500
2000
2500
Index

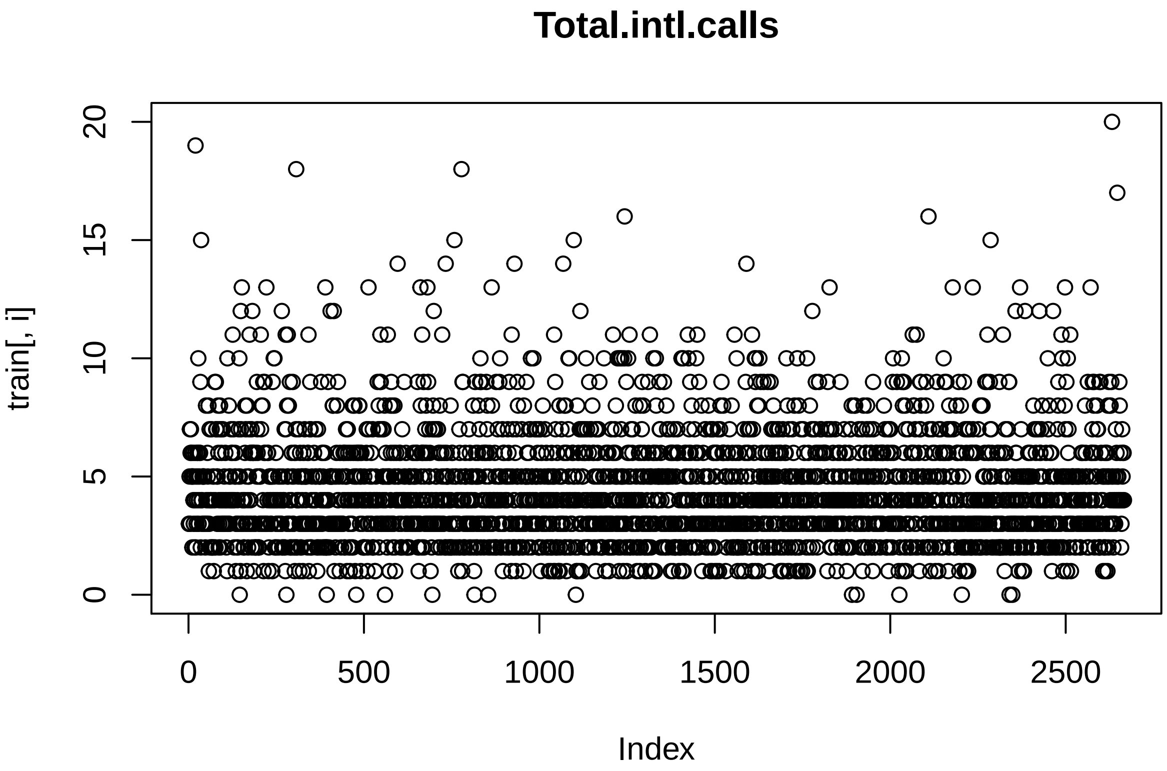

Total.night.charge
Frequency
0
100
200
300
400
5
10
15
train[, i]

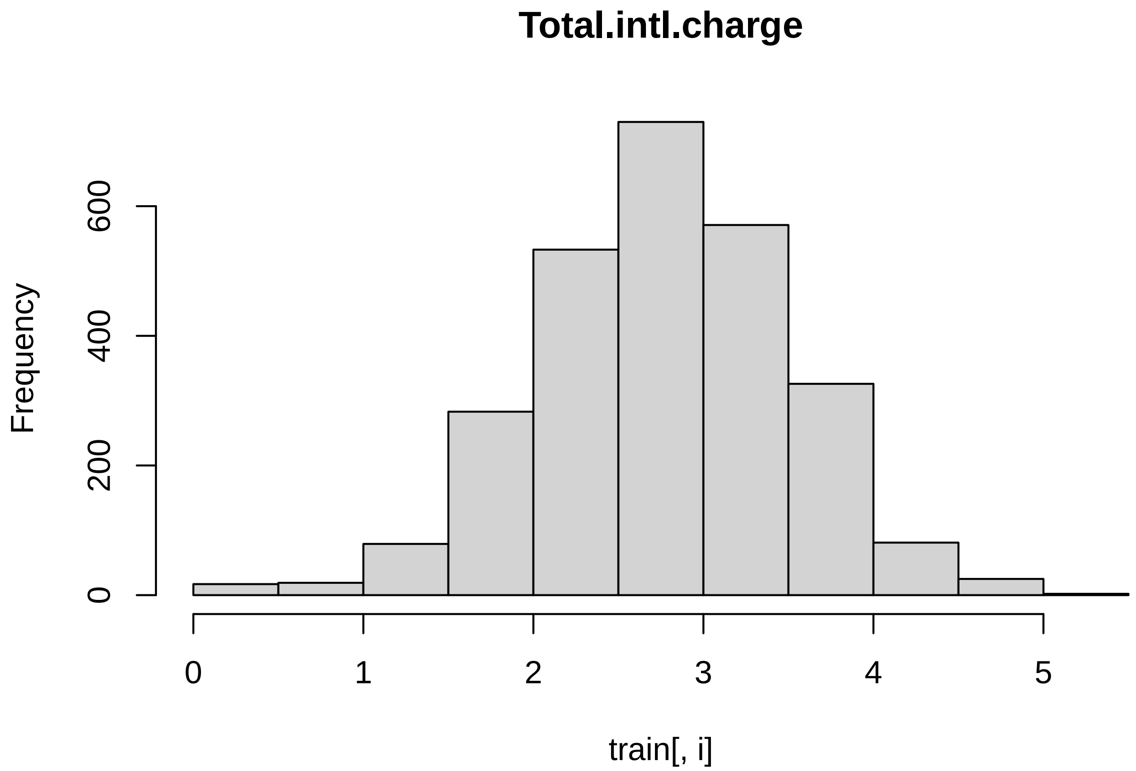

Total.night.charge
train[, i]
Index
5
10
15
0
500
1000
1500
2000
2500

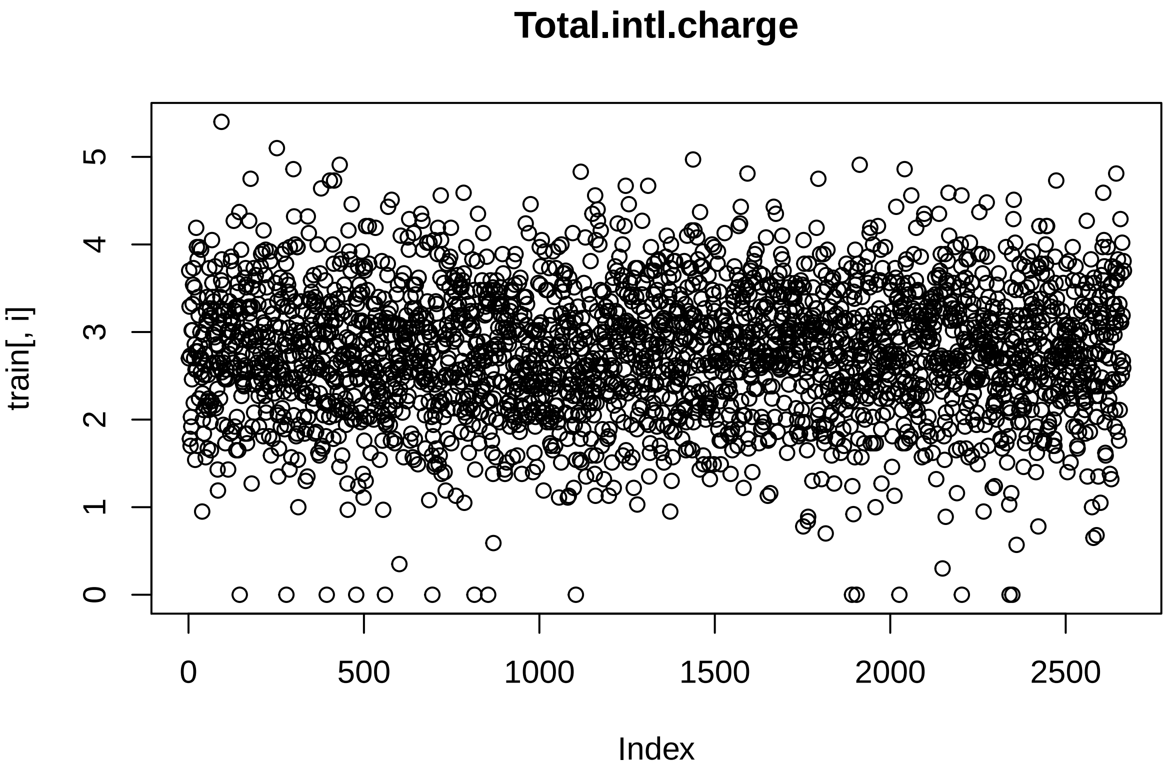

Total.intl.minutes
Frequency
0
200
400
600
0
5
10
15
20
train[, i]

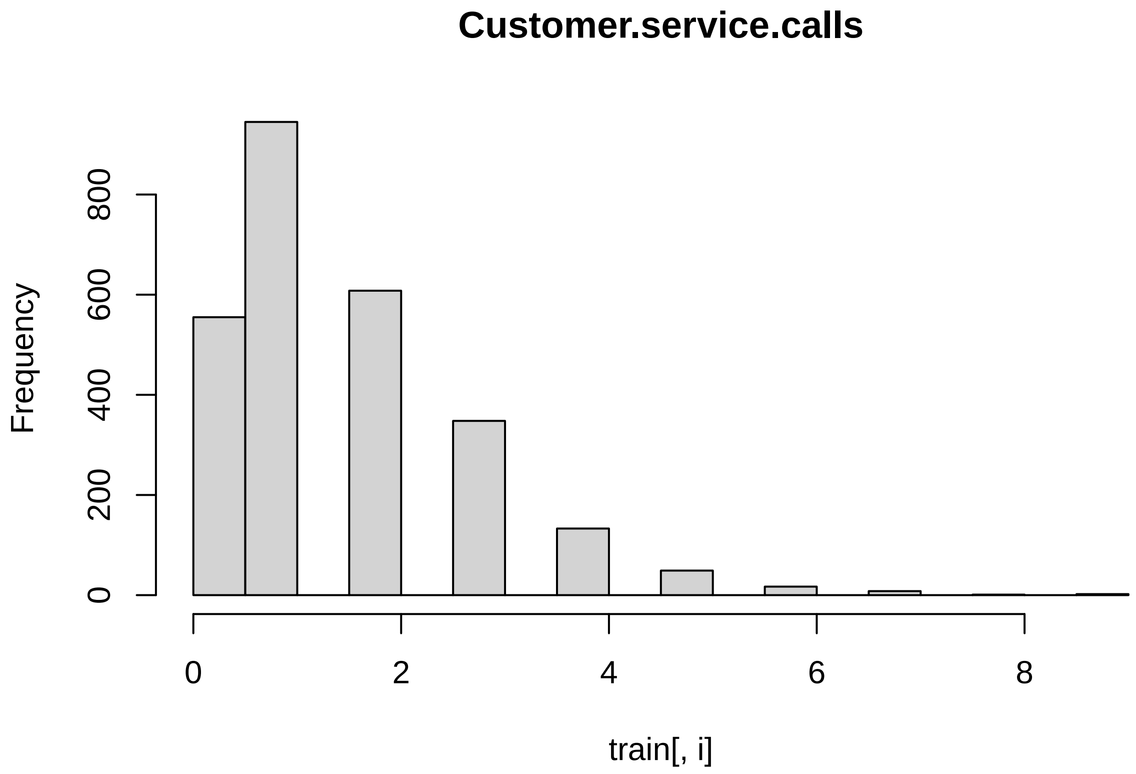

Total.intl.minutes
train[, i]
0
5
10
15
20
0
500
1000
1500
2000
2500
Index

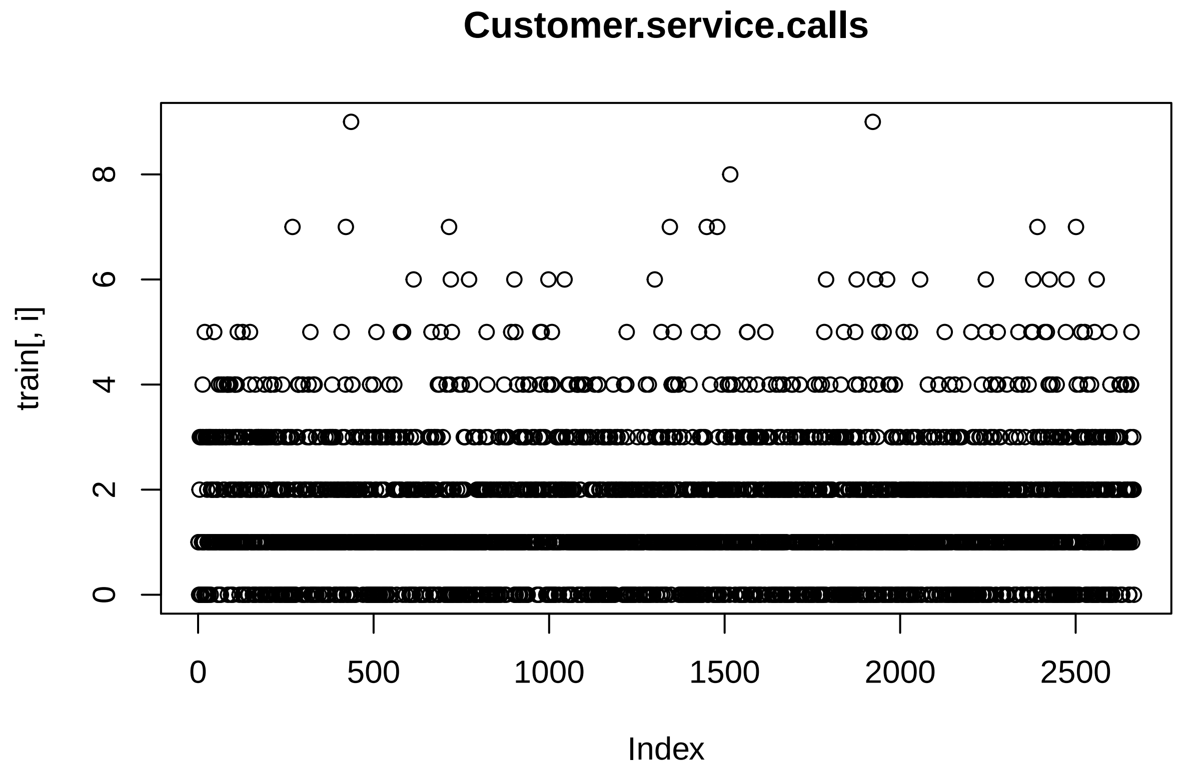

Total.intl.calls
Frequency
0
200
400
600
800
1000
0
5
10
15
20
train[, i]


Total.intl.calls
train[, i]
Index
0
5
10
15
20
0
500
1000
1500
2000
2500


Total.intl.charge
Frequency
0
200
400
600
0
1
2
3
4
5
train[, i]


Total.intl.charge
train[, i]
0
1
2
3
4
5
Index
0
500
1000
1500
2000
2500


Customer.service.calls
Frequency
0
200
400
600
800
0
2
4
6
8
train[, i]



REMARQUE : Si on conserve les varialbes Customer.service.calls et total.intl,calls en variables numériques, il faudra surement prendre le log de ces variables car elles sont asymetriques à droite.!!

Analyse descriptive des données

```
summary(train)
```

```
##      State      Account.length    Area.code     International.plan
##  WV     :  88   Min.   :  1.0   Min.   :408.0   No :2396
##  MN     :  70   1st Qu.: 73.0   1st Qu.:408.0   Yes: 270
##  NY     :  68   Median :100.0   Median :415.0
##  VA     :  67   Mean   :100.6   Mean   :437.4
##  AL     :  66   3rd Qu.:127.0   3rd Qu.:510.0
##  OH     :  66   Max.   :243.0   Max.   :510.0
##  (Other):2241
##  Voice.mail.plan Number.vmail.messages Total.day.minutes Total.day.calls
##  No :1933        Min.   : 0.000        Min.   :  0.0     Min.   :  0.0
##  Yes: 733        1st Qu.: 0.000        1st Qu.:143.4     1st Qu.: 87.0
##                  Median : 0.000        Median :179.9     Median :101.0
##                  Mean   : 8.022        Mean   :179.5     Mean   :100.3
##                  3rd Qu.:19.000        3rd Qu.:215.9     3rd Qu.:114.0
##                  Max.   :50.000        Max.   :350.8     Max.   :160.0
##
##  Total.day.charge Total.eve.minutes Total.eve.calls Total.eve.charge
##  Min.   : 0.00    Min.   :  0.0     Min.   :  0     Min.   : 0.00
##  1st Qu.:24.38    1st Qu.:165.3     1st Qu.: 87     1st Qu.:14.05
##  Median :30.59    Median :200.9     Median :100     Median :17.08
##  Mean   :30.51    Mean   :200.4     Mean   :100     Mean   :17.03
##  3rd Qu.:36.70    3rd Qu.:235.1     3rd Qu.:114     3rd Qu.:19.98
##  Max.   :59.64    Max.   :363.7     Max.   :170     Max.   :30.91
##
##  Total.night.minutes Total.night.calls Total.night.charge Total.intl.minutes
##  Min.   : 43.7       Min.   : 33.0     Min.   : 1.970     Min.   : 0.00
##  1st Qu.:166.9       1st Qu.: 87.0     1st Qu.: 7.513     1st Qu.: 8.50
##  Median :201.2       Median :100.0     Median : 9.050     Median :10.20
##  Mean   :201.2       Mean   :100.1     Mean   : 9.053     Mean   :10.24
##  3rd Qu.:236.5       3rd Qu.:113.0     3rd Qu.:10.640     3rd Qu.:12.10
##  Max.   :395.0       Max.   :166.0     Max.   :17.770     Max.   :20.00
##
##  Total.intl.calls Total.intl.charge Customer.service.calls   Churn
##  Min.   : 0.000   Min.   :0.000     Min.   :0.000          Mode :logical
##  1st Qu.: 3.000   1st Qu.:2.300     1st Qu.:1.000          FALSE:2278
##  Median : 4.000   Median :2.750     Median :1.000          TRUE :388
##  Mean   : 4.467   Mean   :2.764     Mean   :1.563
##  3rd Qu.: 6.000   3rd Qu.:3.270     3rd Qu.:2.000
##  Max.   :20.000   Max.   :5.400     Max.   :9.000
##
```

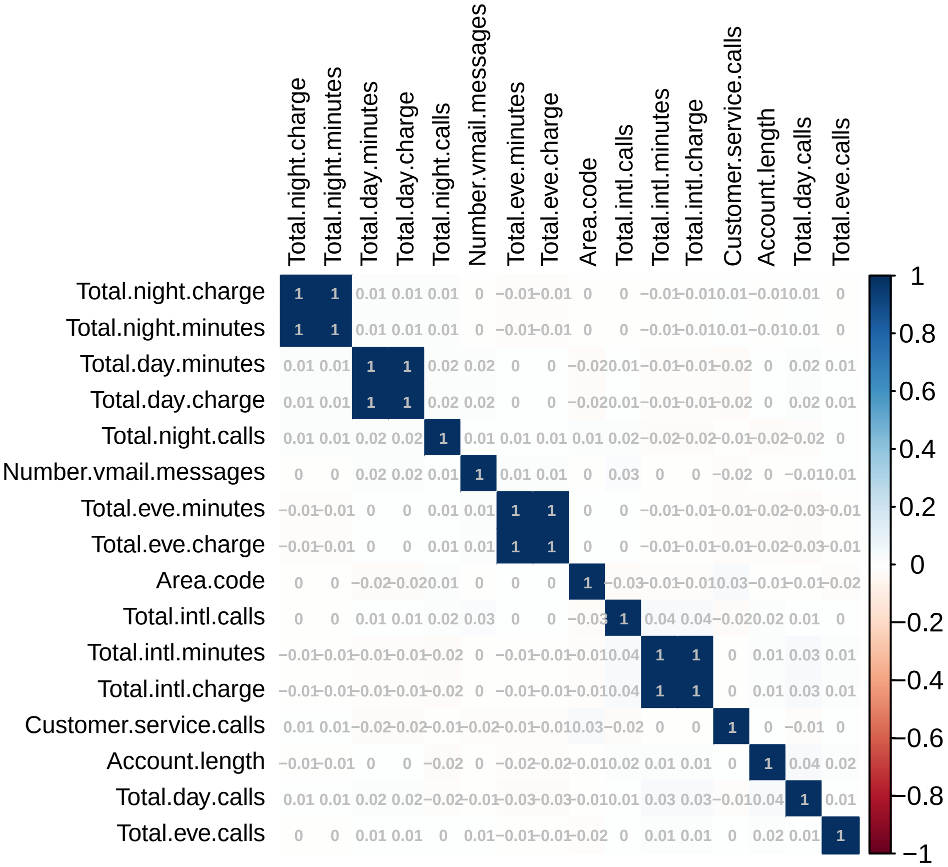
Le dataset d'entraînement ne contient aucune valeur manquante. il y a 15 variables continues, 3 variables catégorielles, et une variables binaires.

```
round(mean(train$Churn),digits=2)
```

```
## [1] 0.15
```

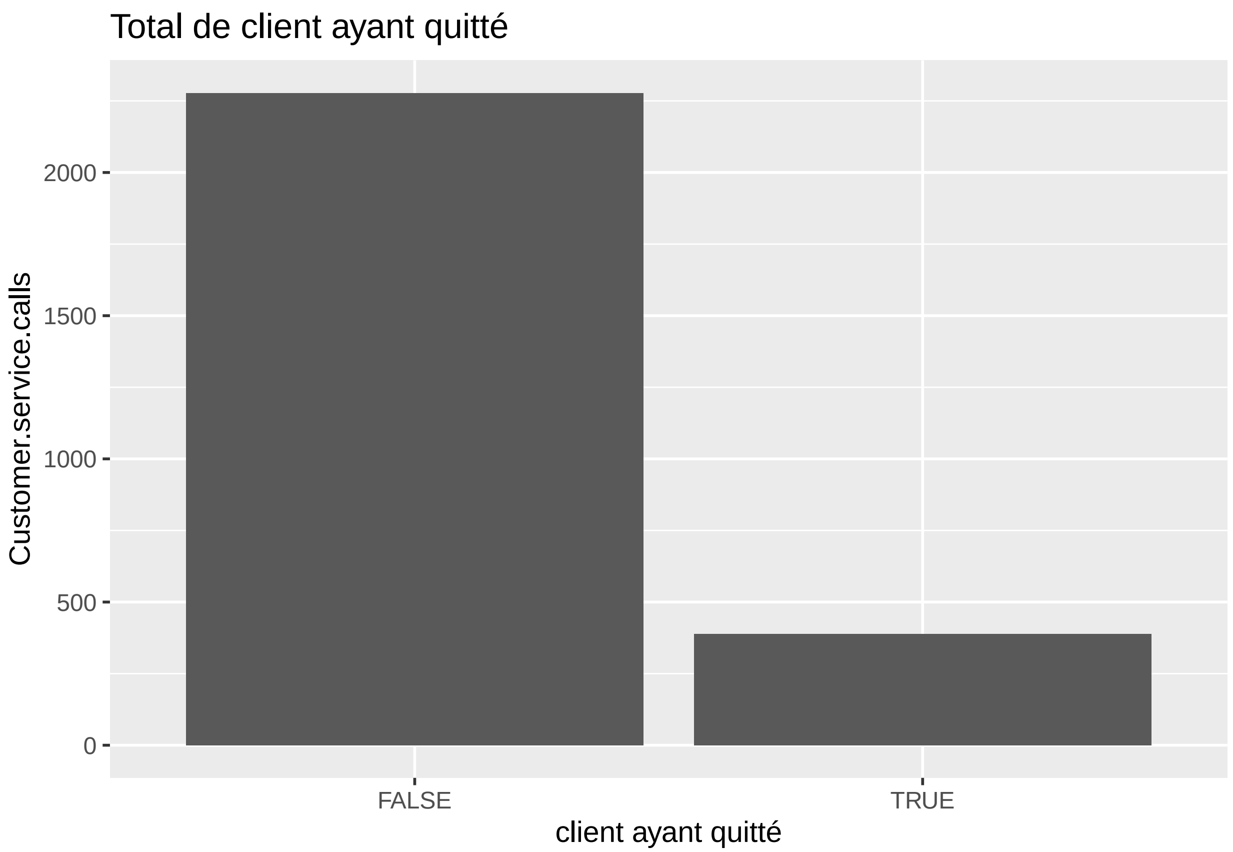
Le pourcentage de clients ayant quittés la compagnie est de 15%.

```
corrplot(cor(train[,c(2,3,6:19)]), method = "color", addCoef.col="grey", order = "AOE",tl.cex=0.75,tl.c
```



Les varaibles du dataset sont faiblement corrélées à l'exception des variables indiquant le nombre de muinutes consommés et les frais chargés associés, comme les varaibles : "Total.night charge" et "Total.night.minutes". Comme ces variables ont une corrélation parfaites, nous décidons de supprimer du dataset les variables "charge". Ce qui revient à supprimer 4 variables du dataset.

```r
library(ggplot2)
ggplot(train, aes(x = Churn, fill=Customer.service.calls)) +
  geom_bar( ) +
  xlab("client ayant quitté") + ylab("Customer.service.calls") +
  ggtitle("Total de client ayant quitté")
```



```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.3
```

```r
library(usmap)
```

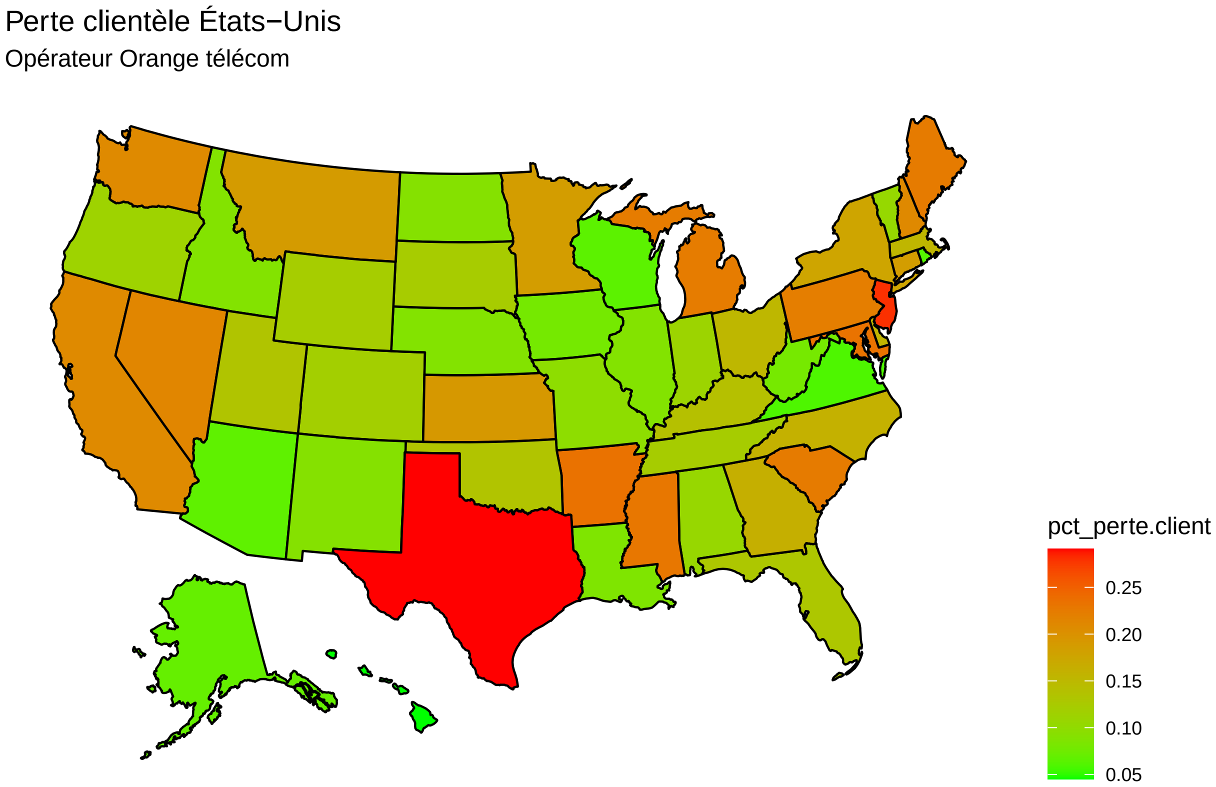
```
## Warning: package 'usmap' was built under R version 4.0.3
```

```r
us.map=train
names(us.map)[names(us.map)=="State"]="state"
us.map = data.frame(us.map)

state.churn=us.map %>%
  group_by(state) %>%
  summarise(pct_perte.client = mean(Churn))
```

```r
us.map = data.frame(us.map)

plot_usmap(regions="state", data=state.churn, values = "pct_perte.client", color="black")+
  scale_fill_continuous(low = "green", high = "red", name = "pct_perte.client")+
  labs(title = "Perte clientèle États-Unis", subtitle = "Opérateur Orange télécom")+
  theme(legend.position = "right")
```



On constate que les états du Texas et New Jersey sont les états ayant perdus le plus de clientèle, avec un pourcentage de perte supérieur à 25%.