Credit Card Fraud Detection

Cecile Guillot

1. Introduction

This project is a classification project based on data about credit card fraud. The aim is to detect the fraudulent transaction. The dataset is available here: dataset. In order to realise this project, R was used. At the end of the project, LIME library helps us to have an idea of the feature importances.

2. Data Cleaning

Data are already cleaned and processing. PCA was used in order to hide the personal data related to the different transaction made by European Placeholder. Nevertheless, missing values was evaluated and the first column represented time and it was removed of the processing.

```
library(dplyr)
```

```
##
## Attachement du package : 'dplyr'

## Les objets suivants sont masqués depuis 'package:stats':
##
##     filter, lag

## Les objets suivants sont masqués depuis 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(janitor)
```

```
##
## Attachement du package : 'janitor'

## Les objets suivants sont masqués depuis 'package:stats':
##
##     chisq.test, fisher.test
```

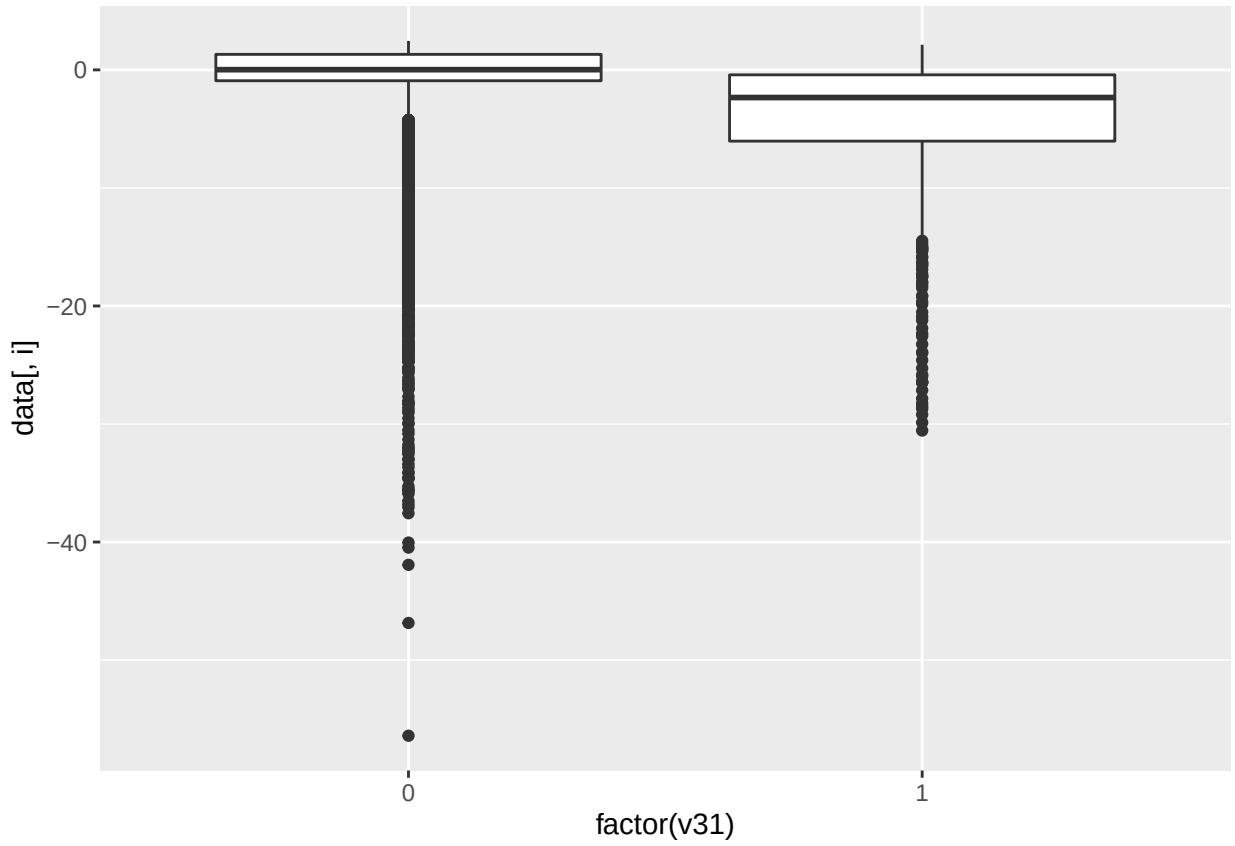
```
library(naniar)
library(stats)

creditcard <- read.csv("~/Credit_Card_Fraud_Detection/00_raw_data/creditcard.csv",
                       header=FALSE)
```

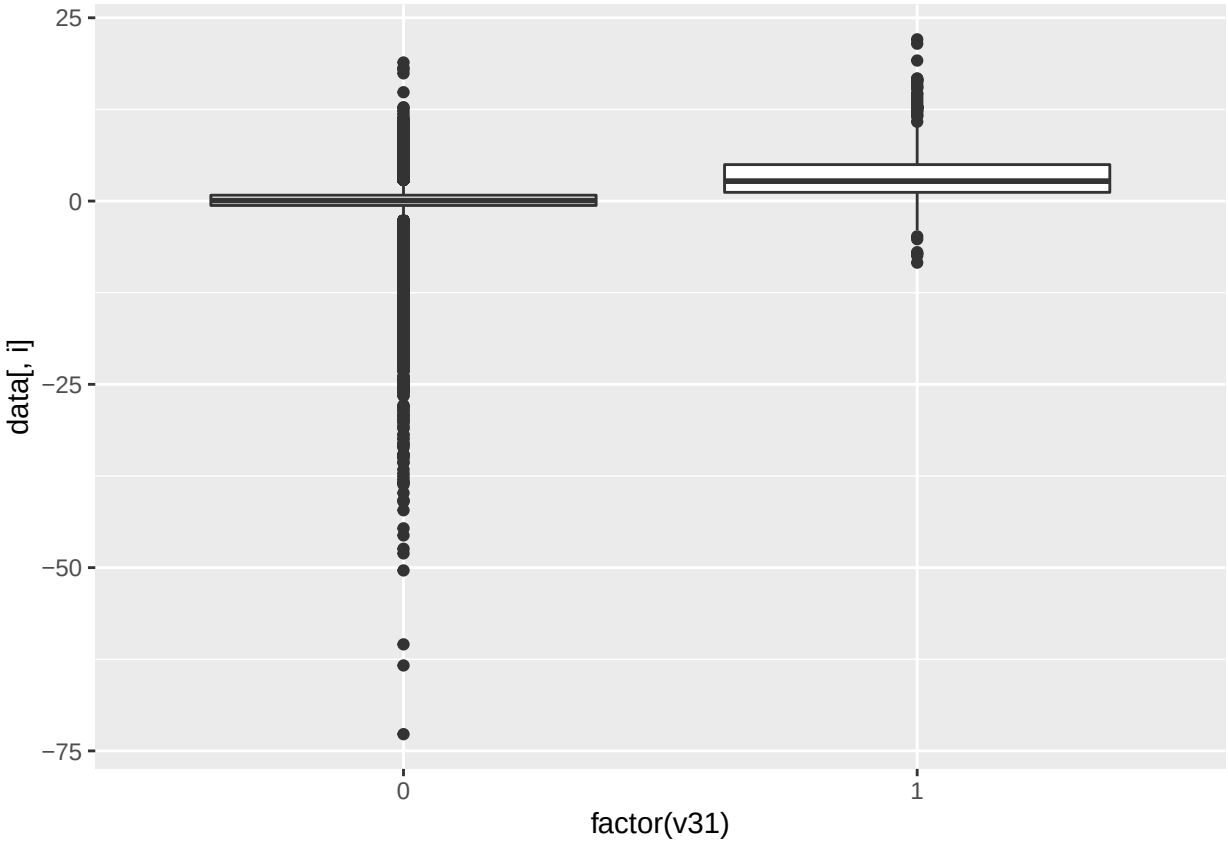
```
creditcard_clean <- clean_names(creditcard)
creditcard_clean <- creditcard_clean[-c(1)]

pct_complete(creditcard_clean)
```

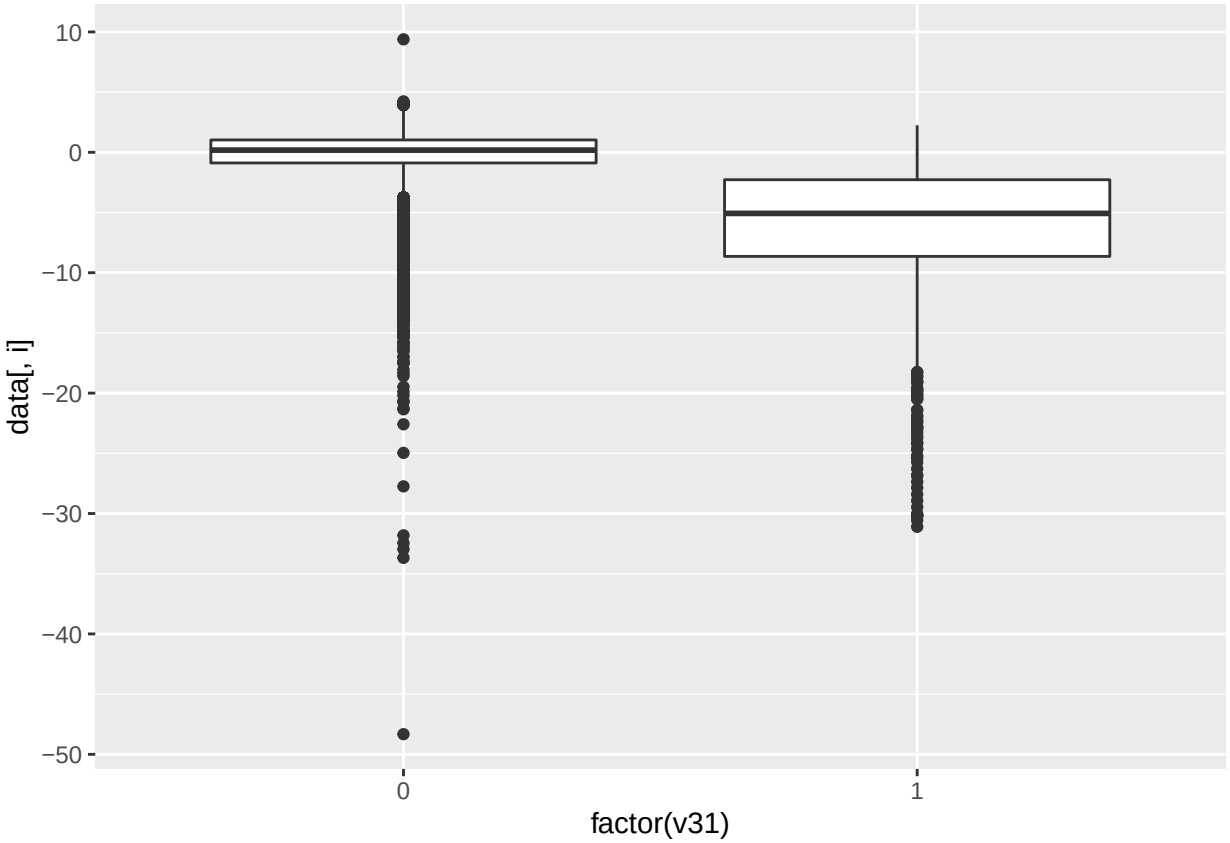
```
## [1] 100
```

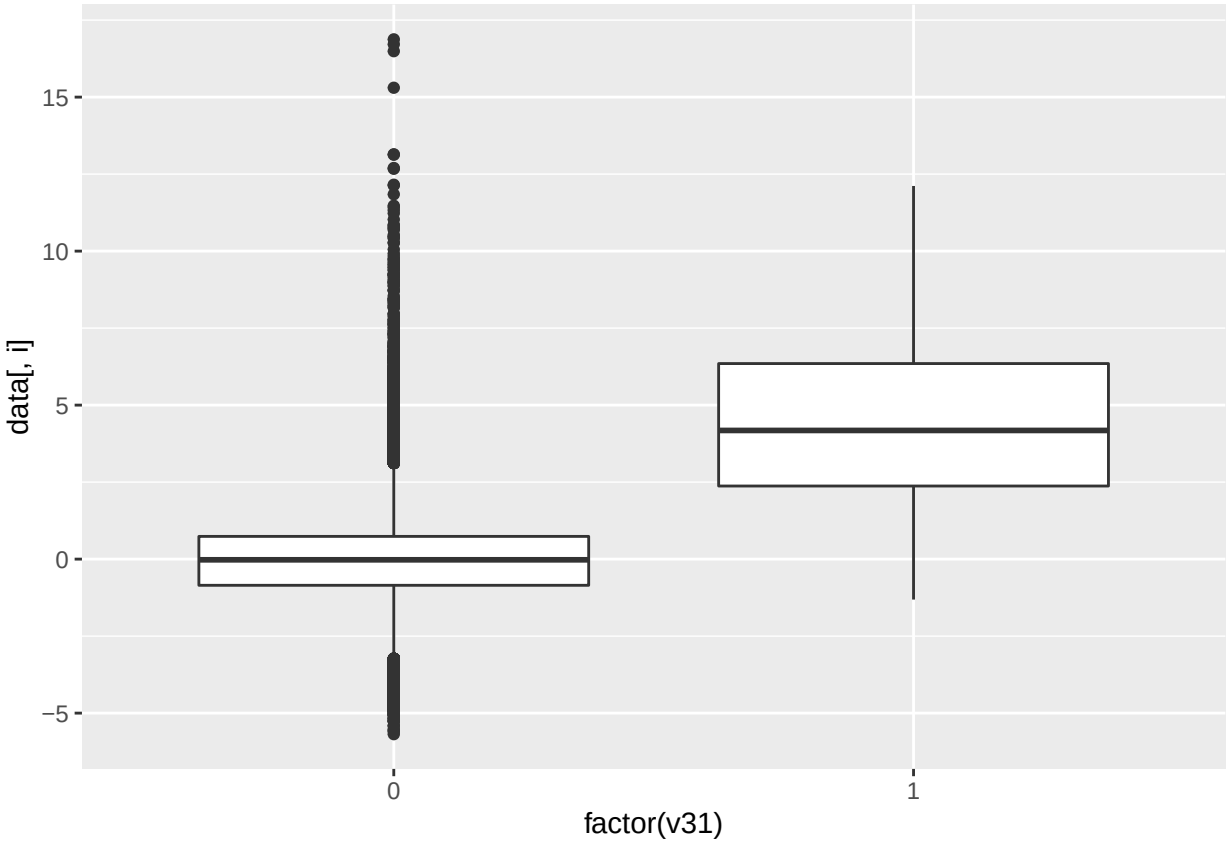
3. Data Analysis

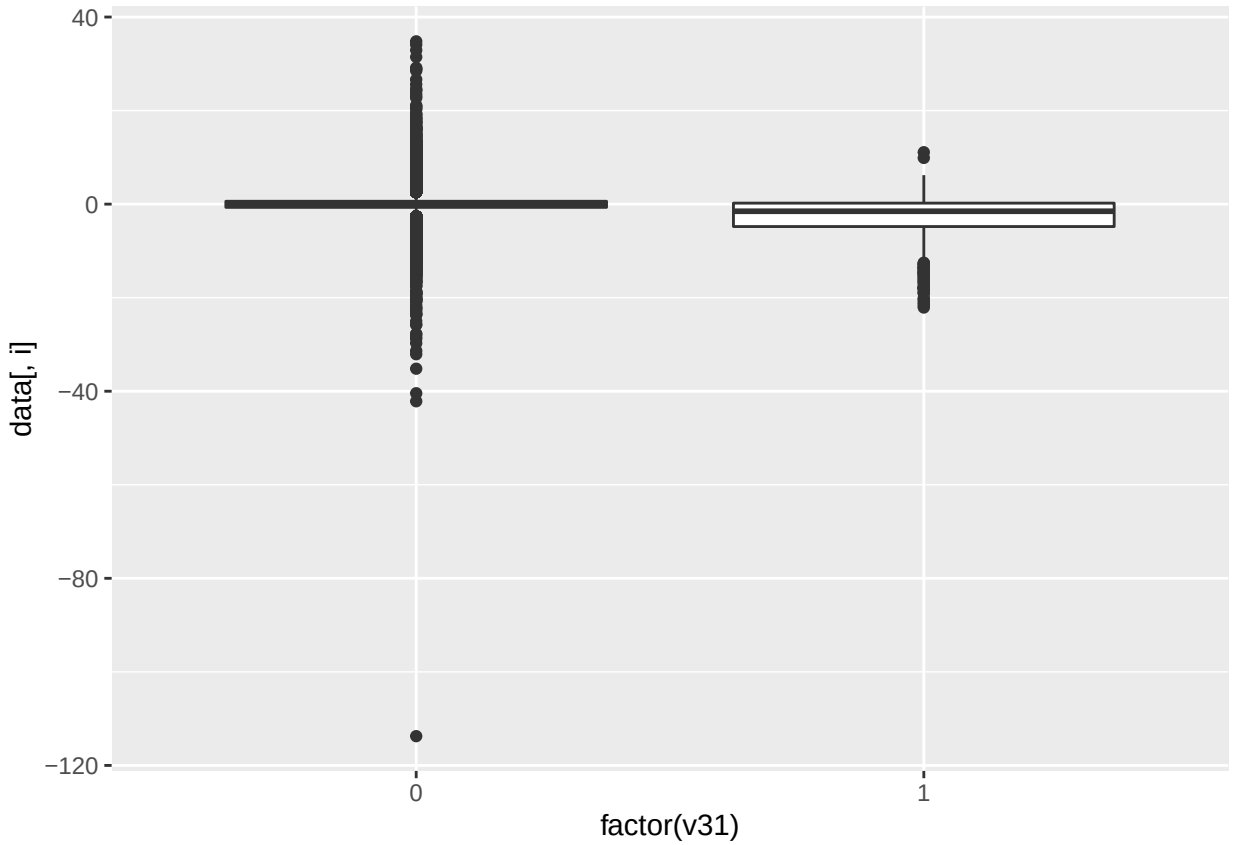
```
library(ggplot2)

data <- read.csv('~/Credit_Card_Fraud_Detection/01_tidy_data/creditcard_clean.csv',
                 header=TRUE)

data <- as.data.frame(sapply(data, as.numeric))

for (i in 1:ncol(data)){
print(ggplot(data, aes(x = factor(v31), y=data[ , i])) +
  geom_boxplot())
  Sys.sleep(2)}
```

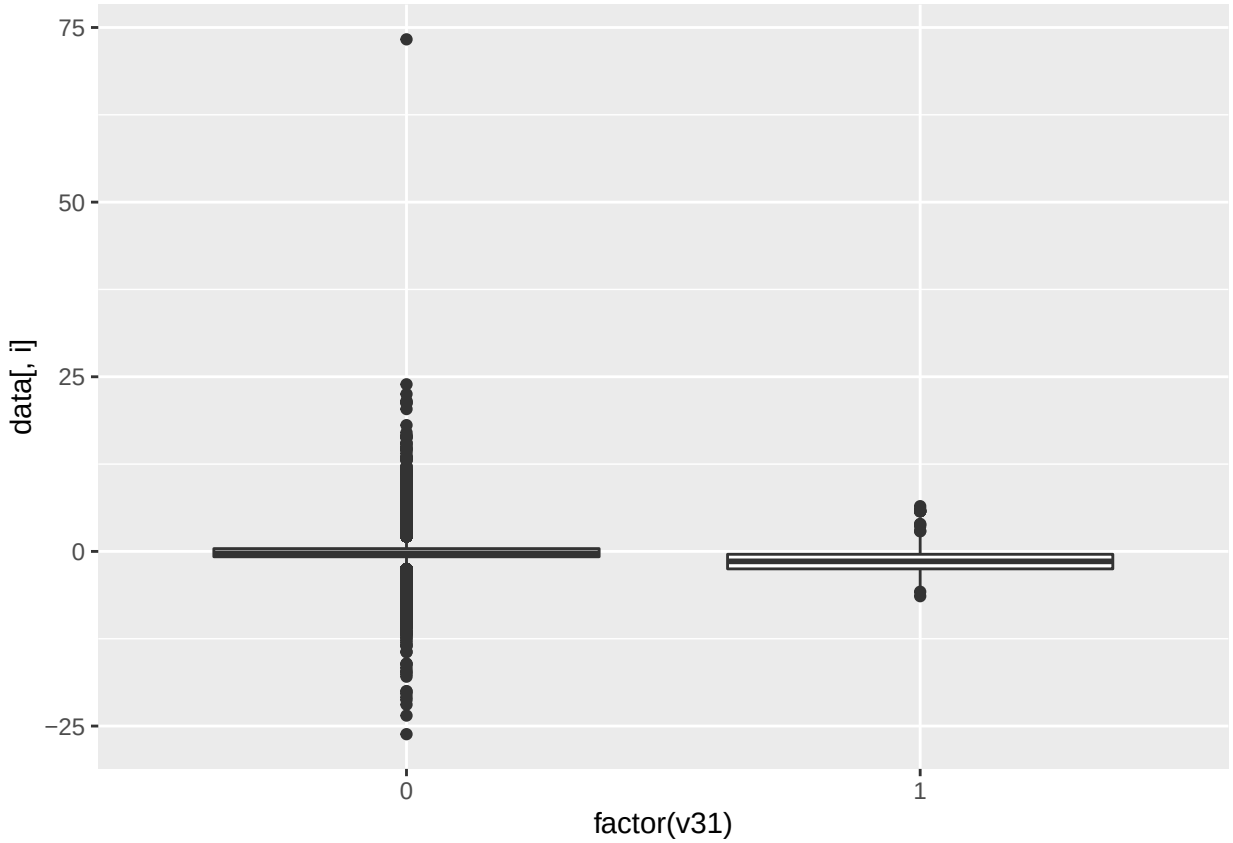


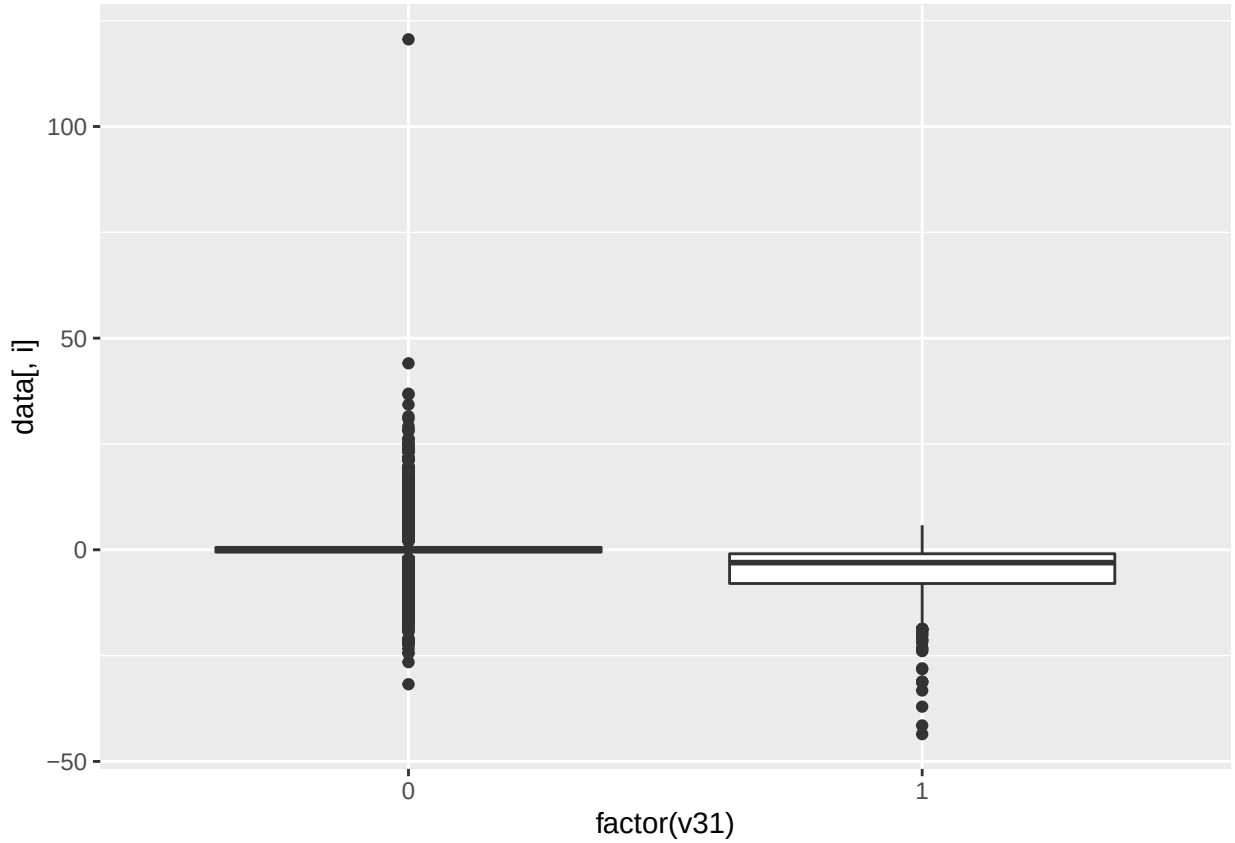

25
0
−25
−50
−75
data[, i]
0
1
factor(v31)

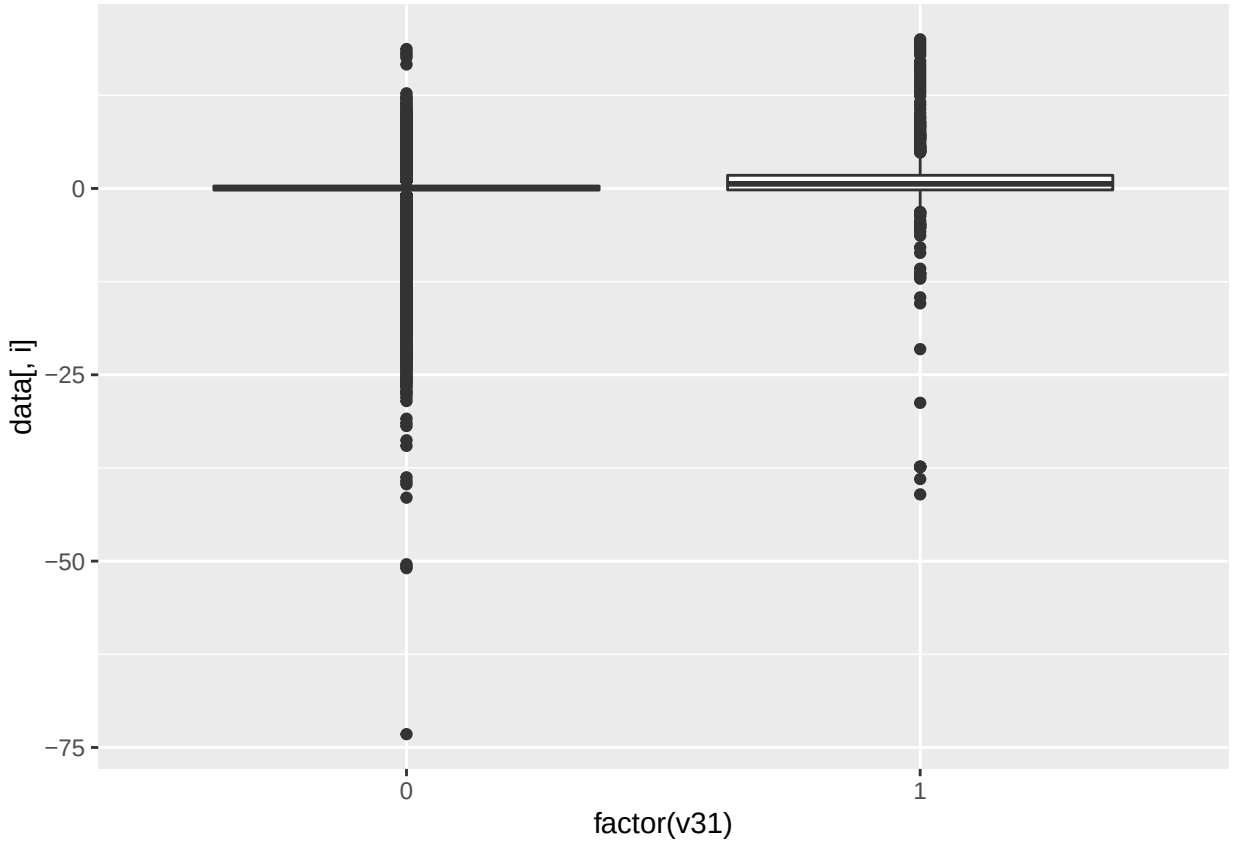

10
0
−10
−20
−30
−40
−50
data[, i]
0
1
factor(v31)

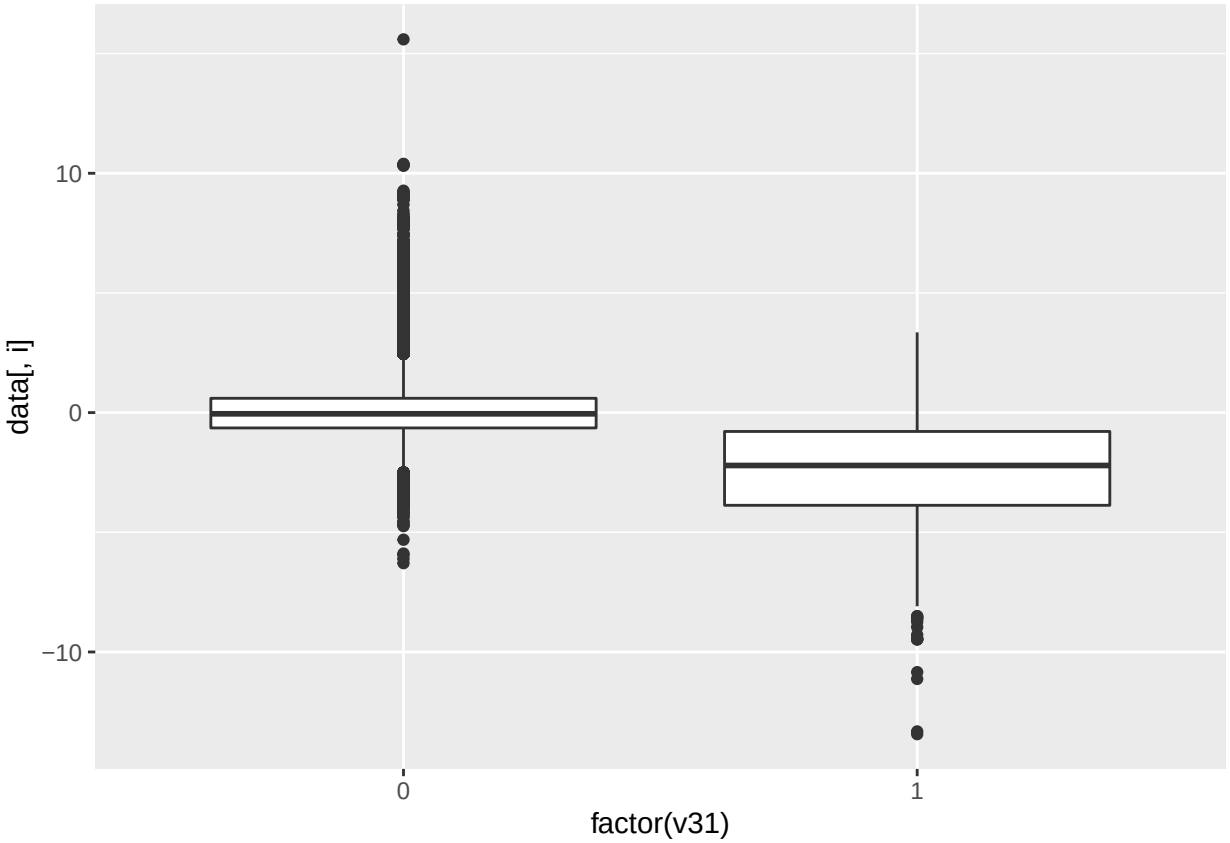

15
10
5
0
−5
data[, i]
0
1
factor(v31)

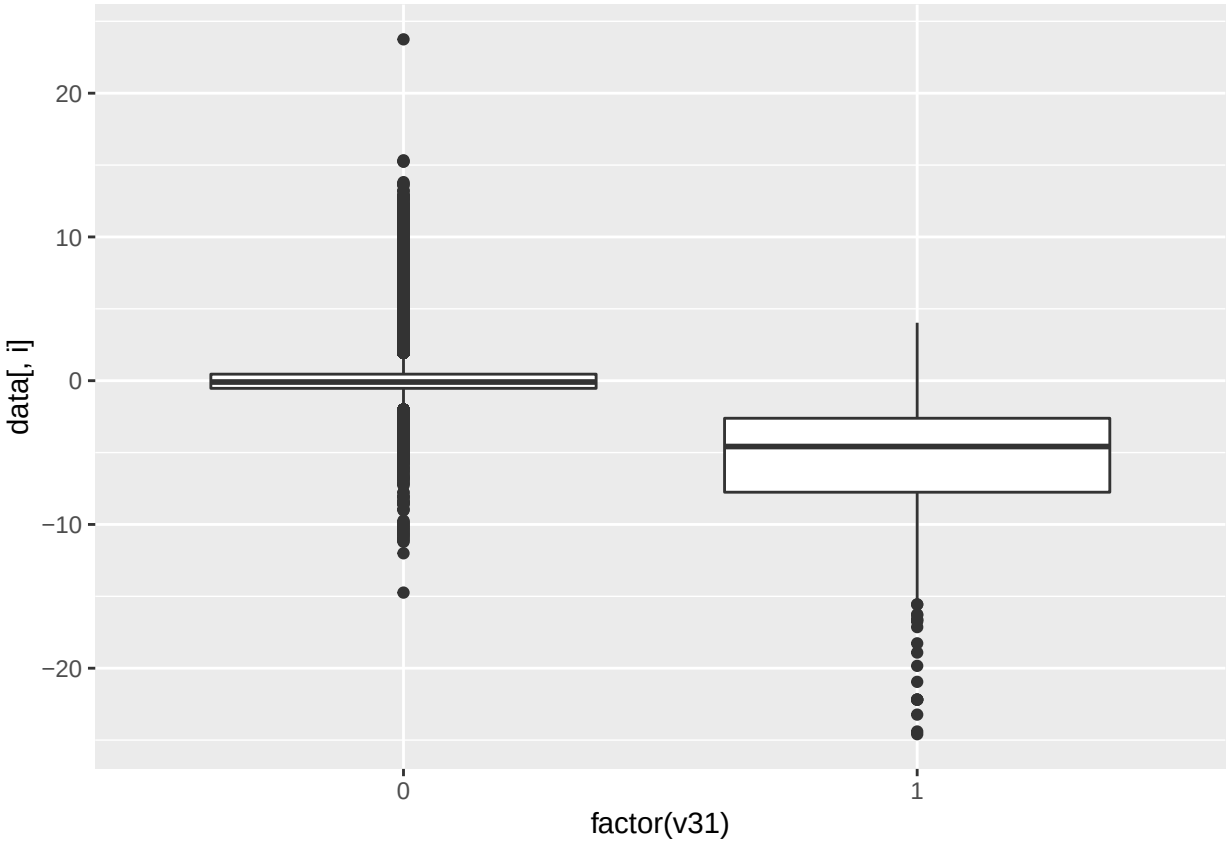

40
0
−40
−80
−120
data[, i]
0
1
factor(v31)

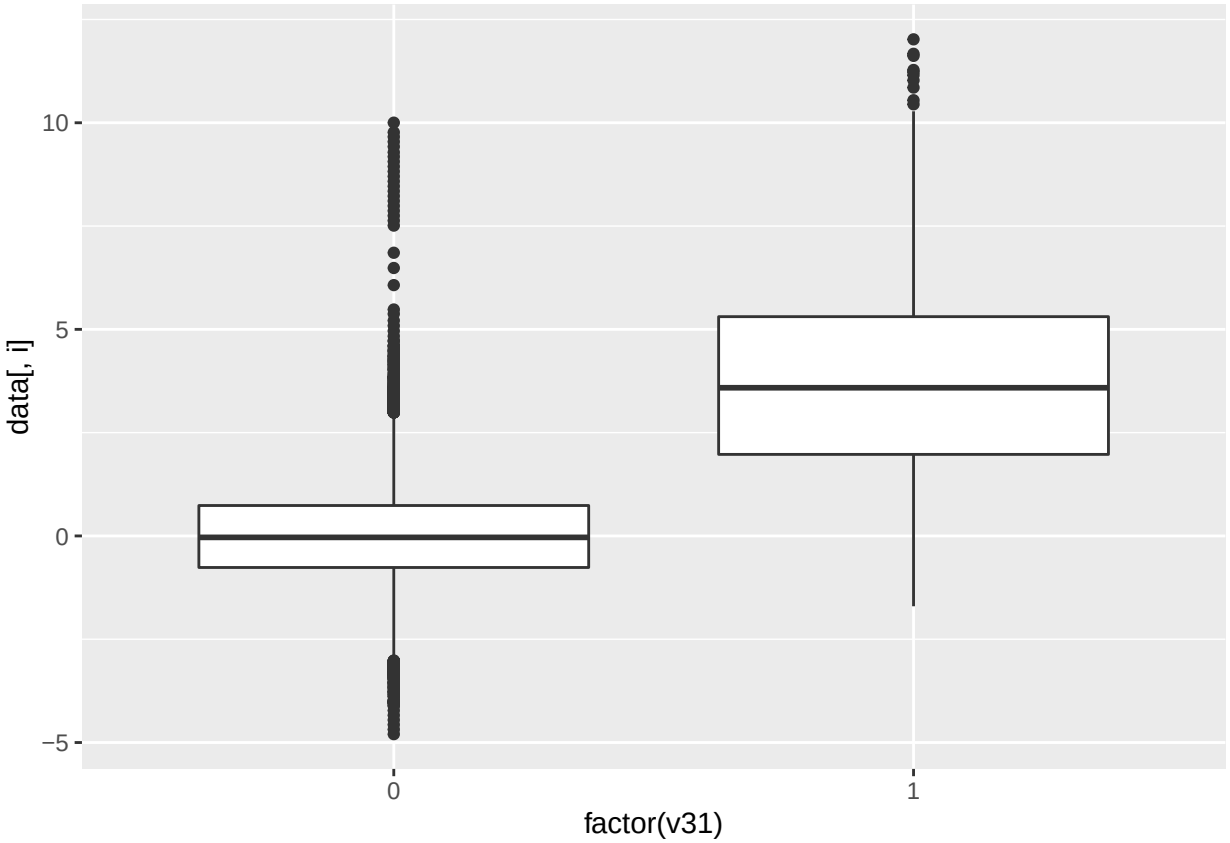

75
50
25
0
−25
data[, i]
0
1
factor(v31)

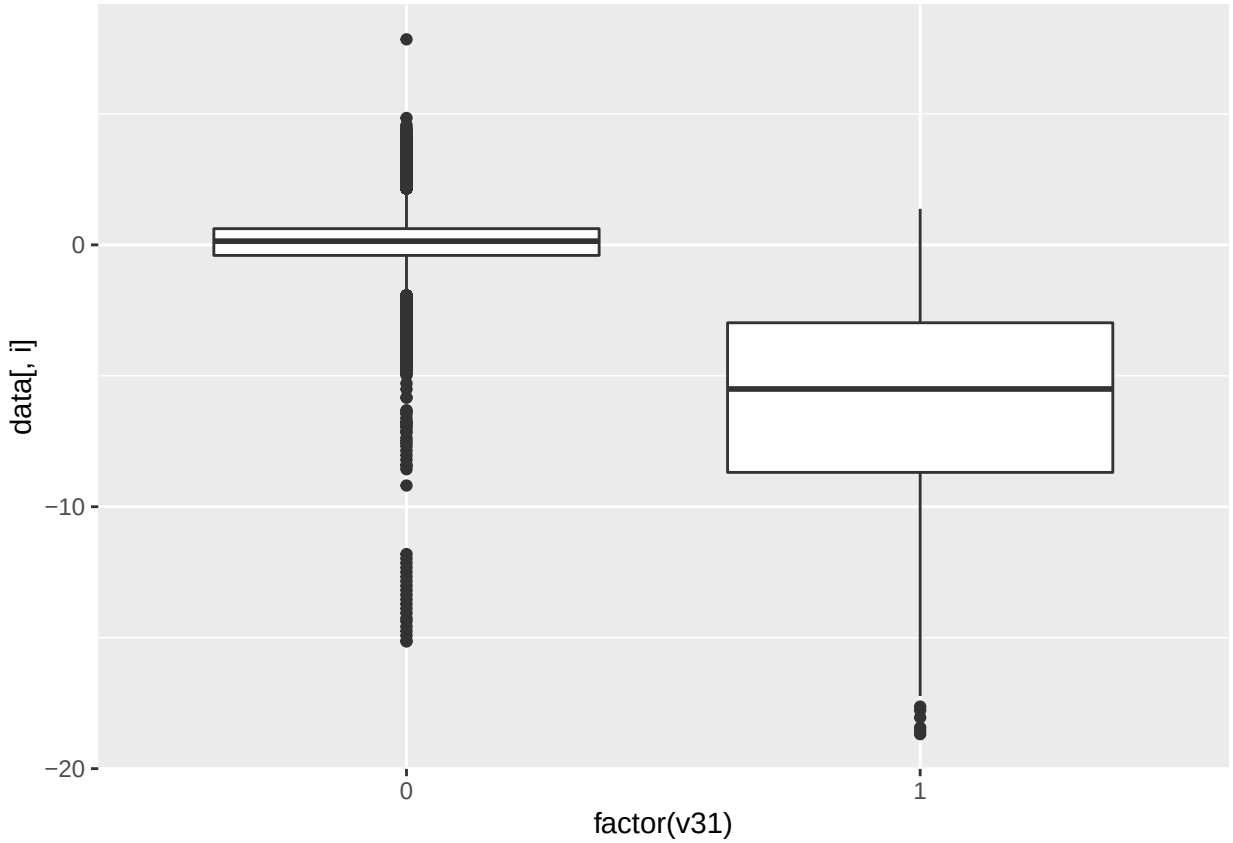

100
50
0
−50
data[, i]
0
1
factor(v31)

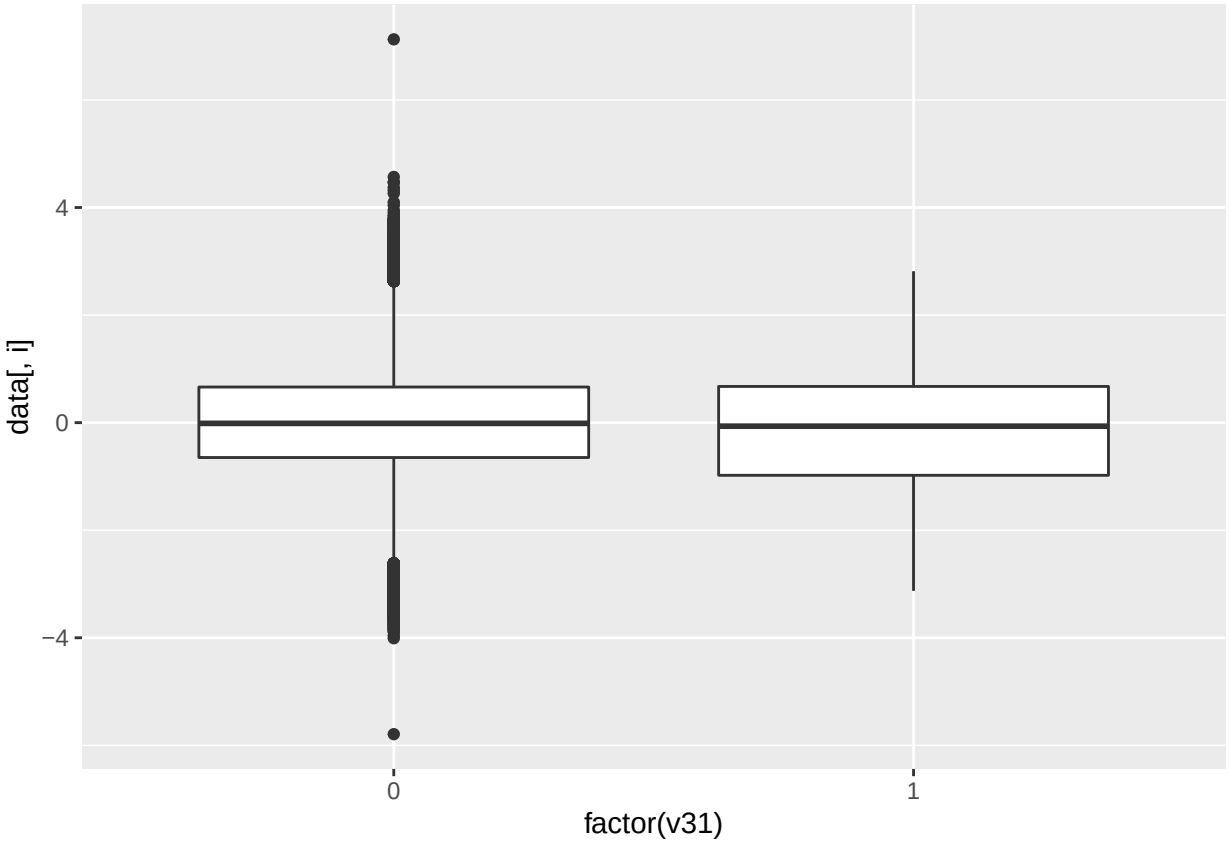

0
−25
−50
−75
data[, i]
0
1
factor(v31)

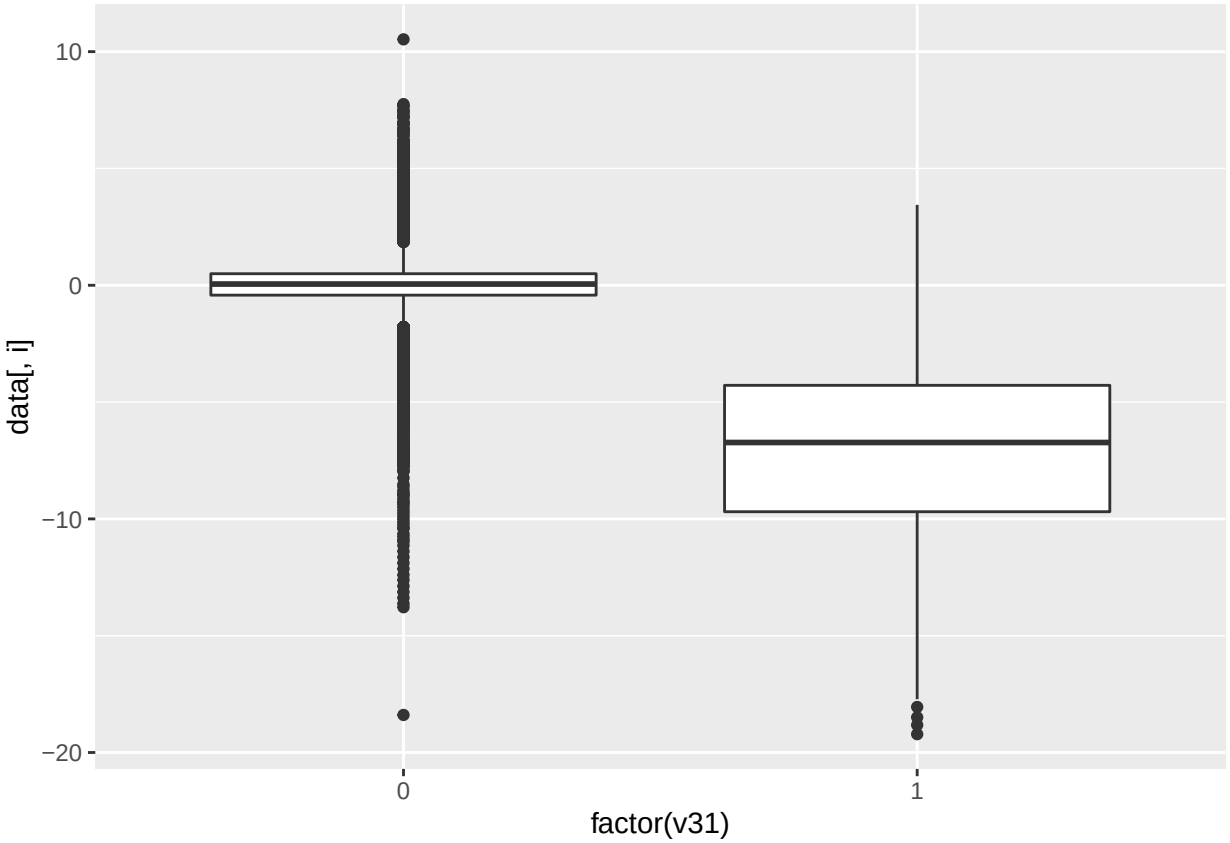

10
0
−10
data[, i]
0
1
factor(v31)

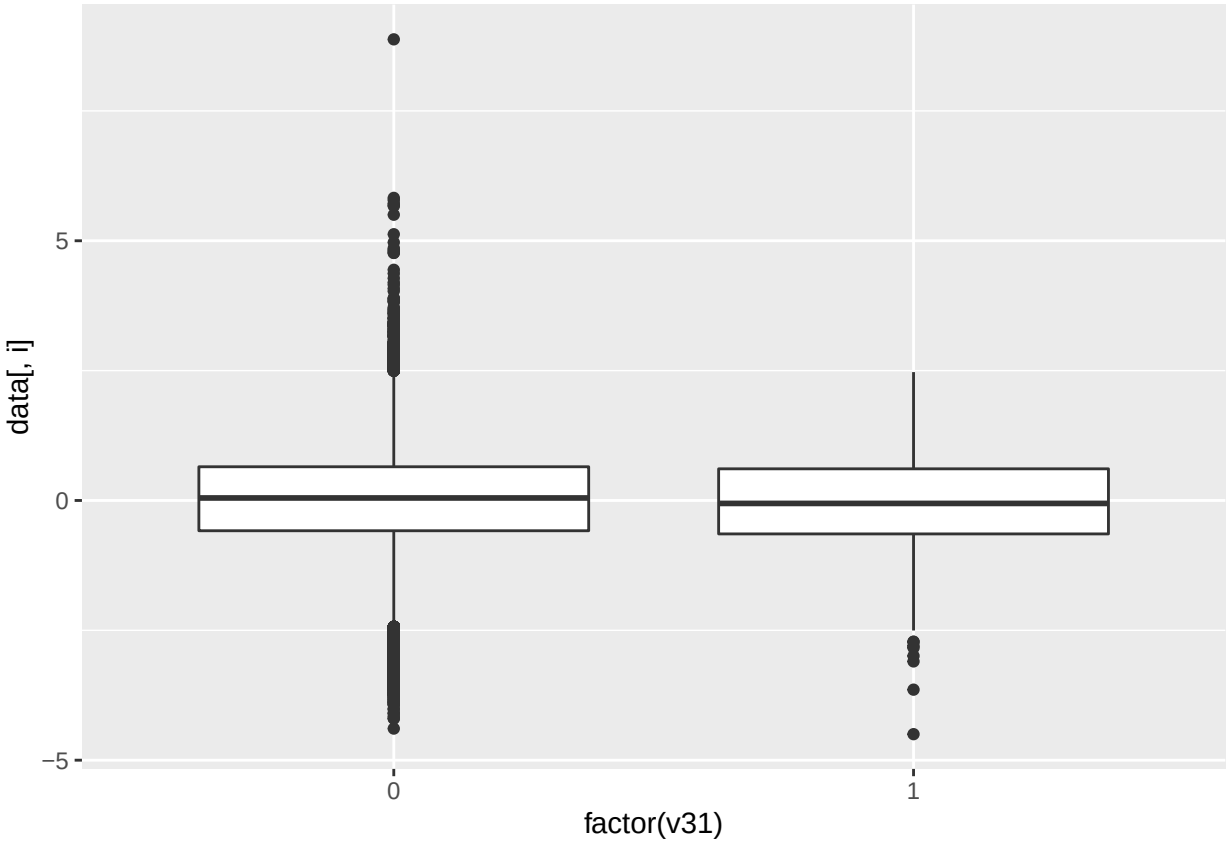

20
10
0
−10
−20
data[, i]
0
1
factor(v31)

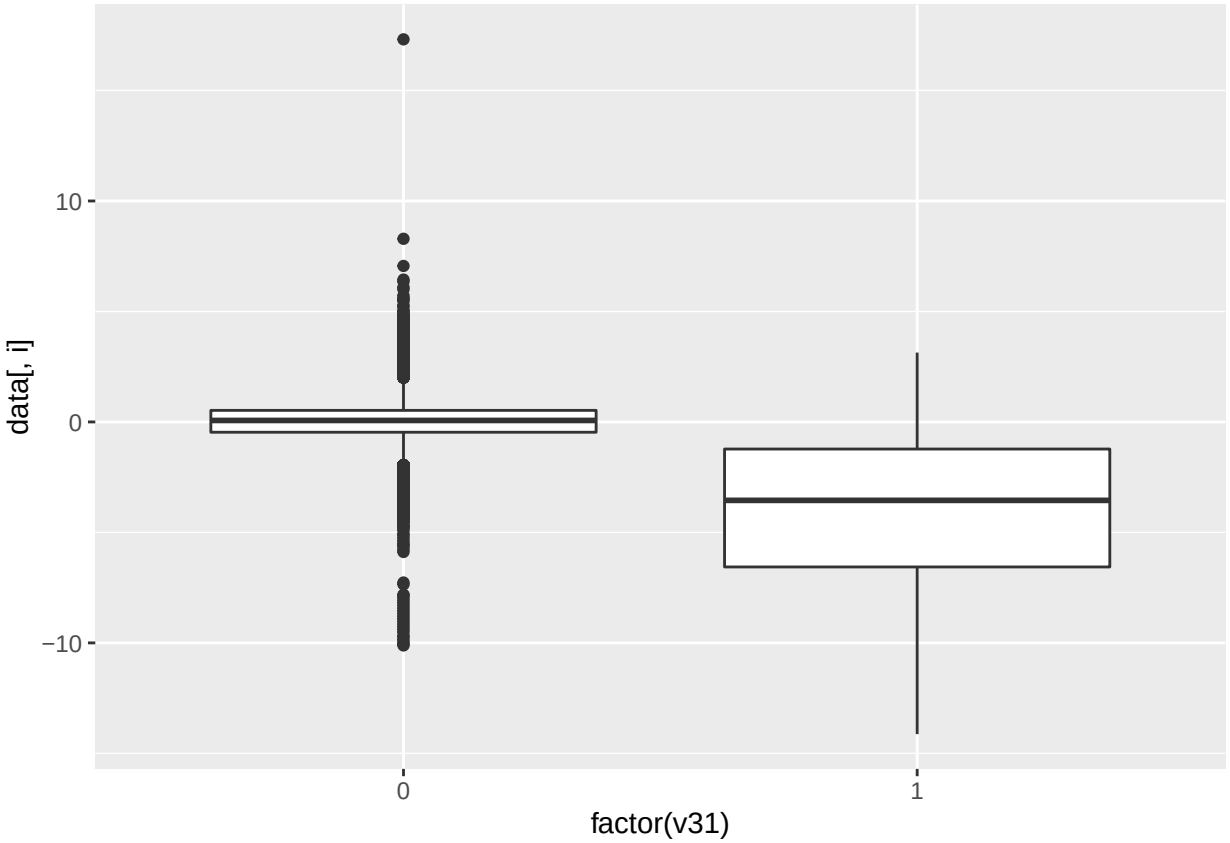

10
5
0
−5
data[, i]
0
1
factor(v31)

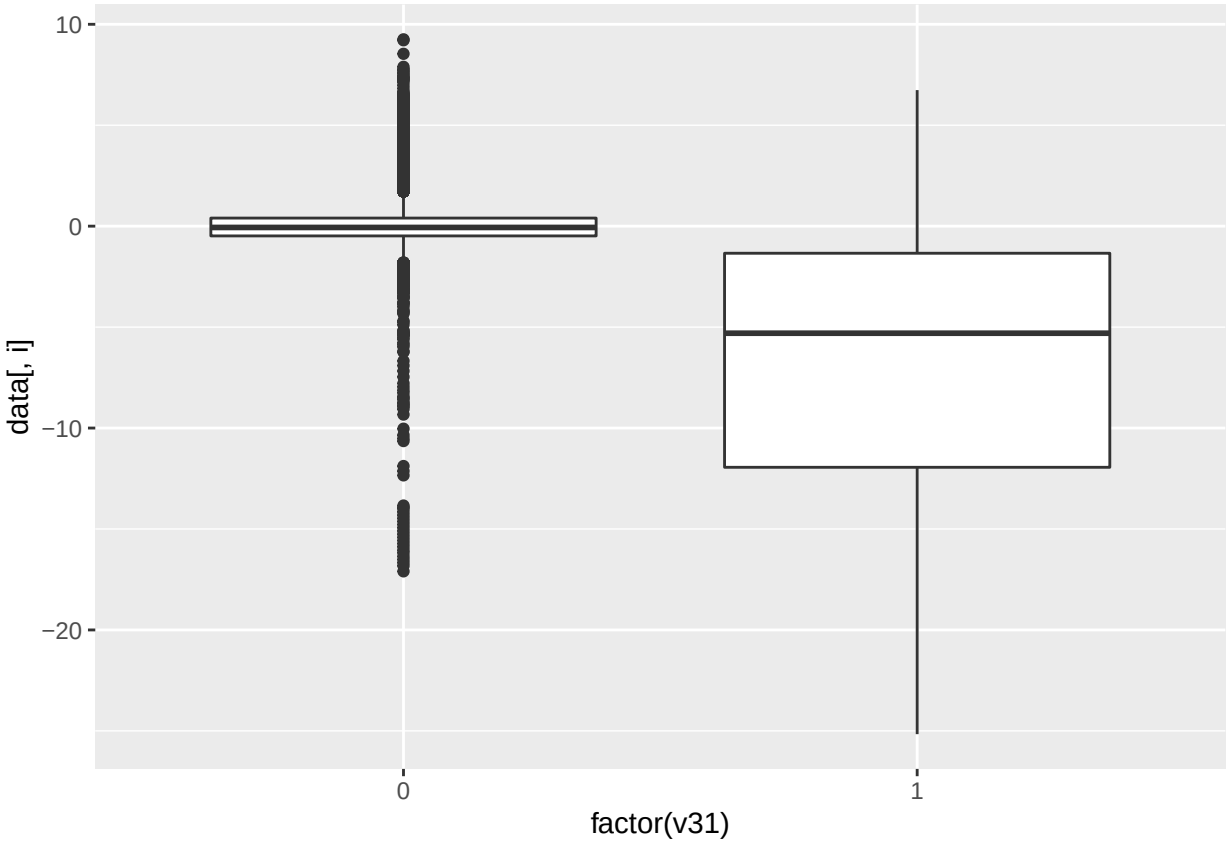

0
−10
−20
data[, i]
0
1
factor(v31)

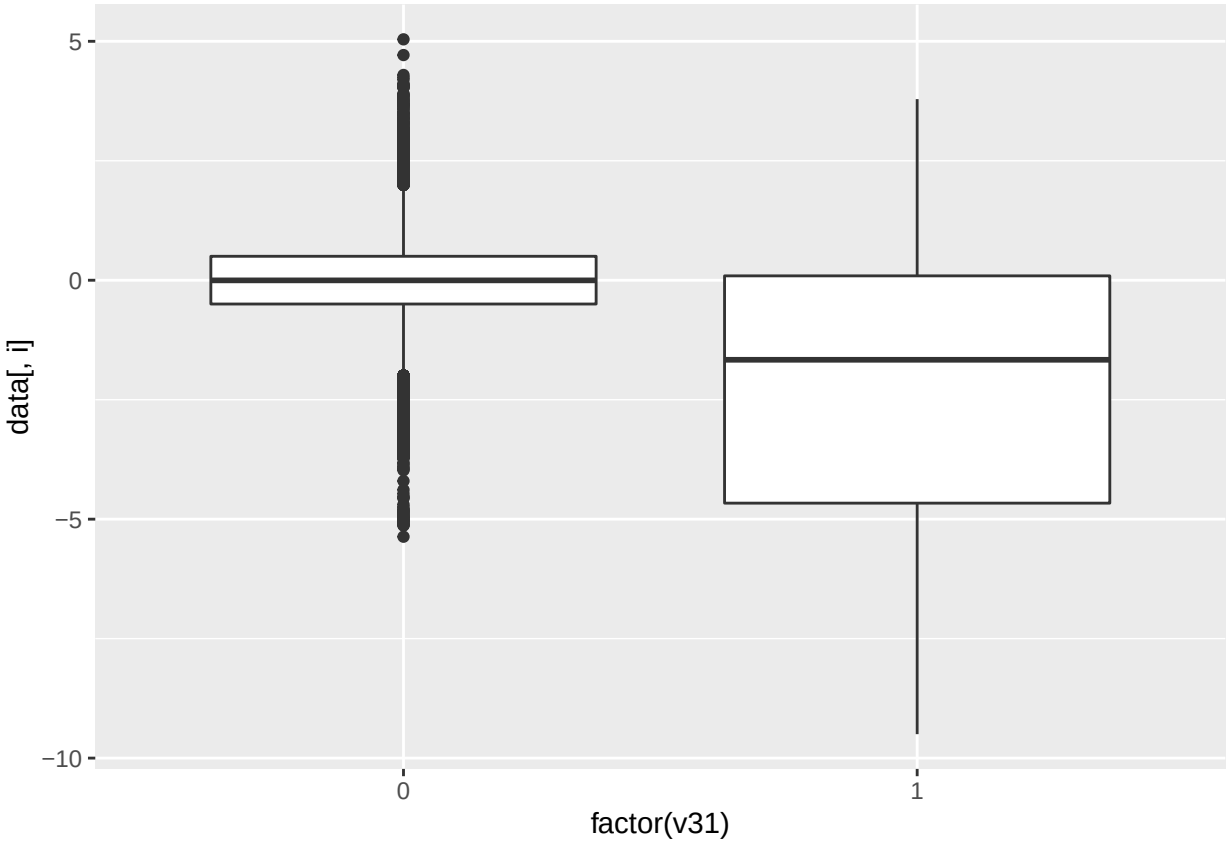

4
0
−4
data[, i]
0
1
factor(v31)

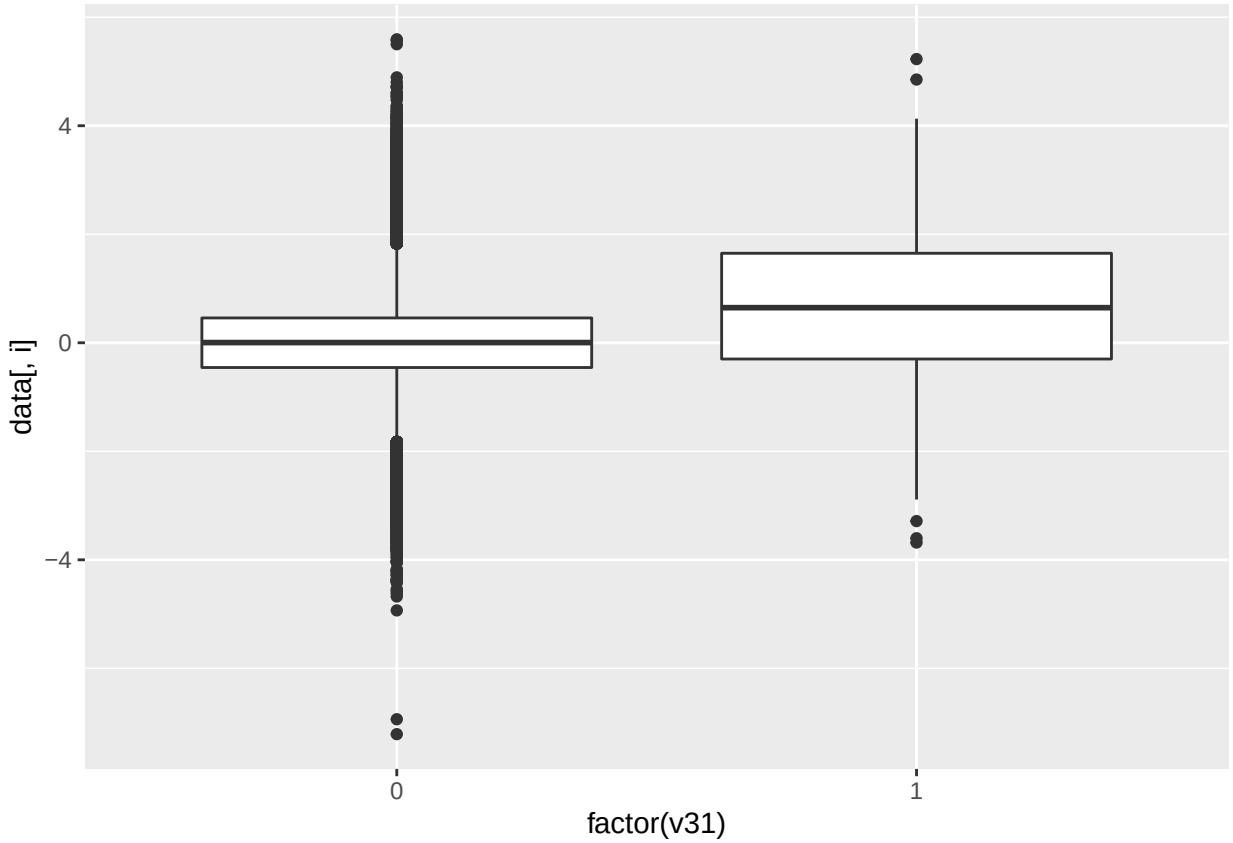

10
0
−10
−20
data[, i]
0
1
factor(v31)

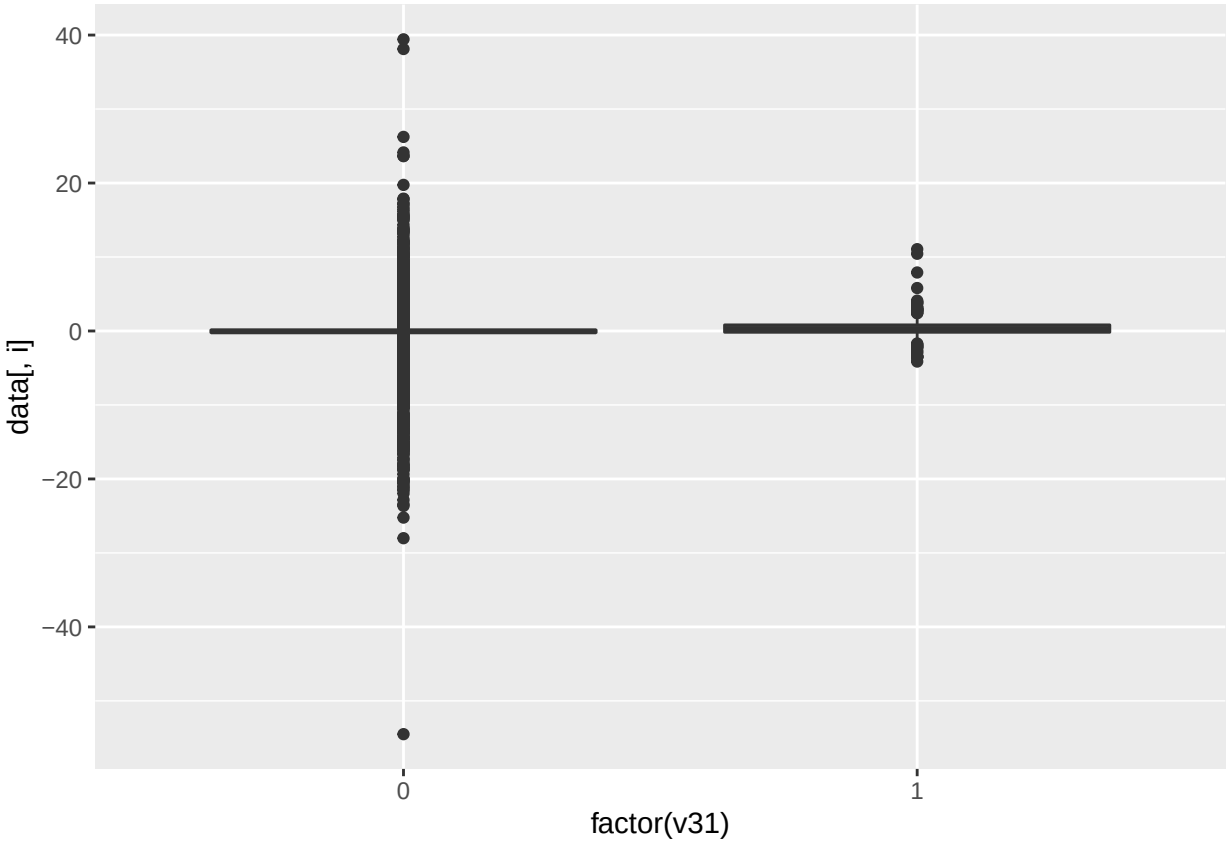

5
0
−5
data[, i]
0
1
factor(v31)

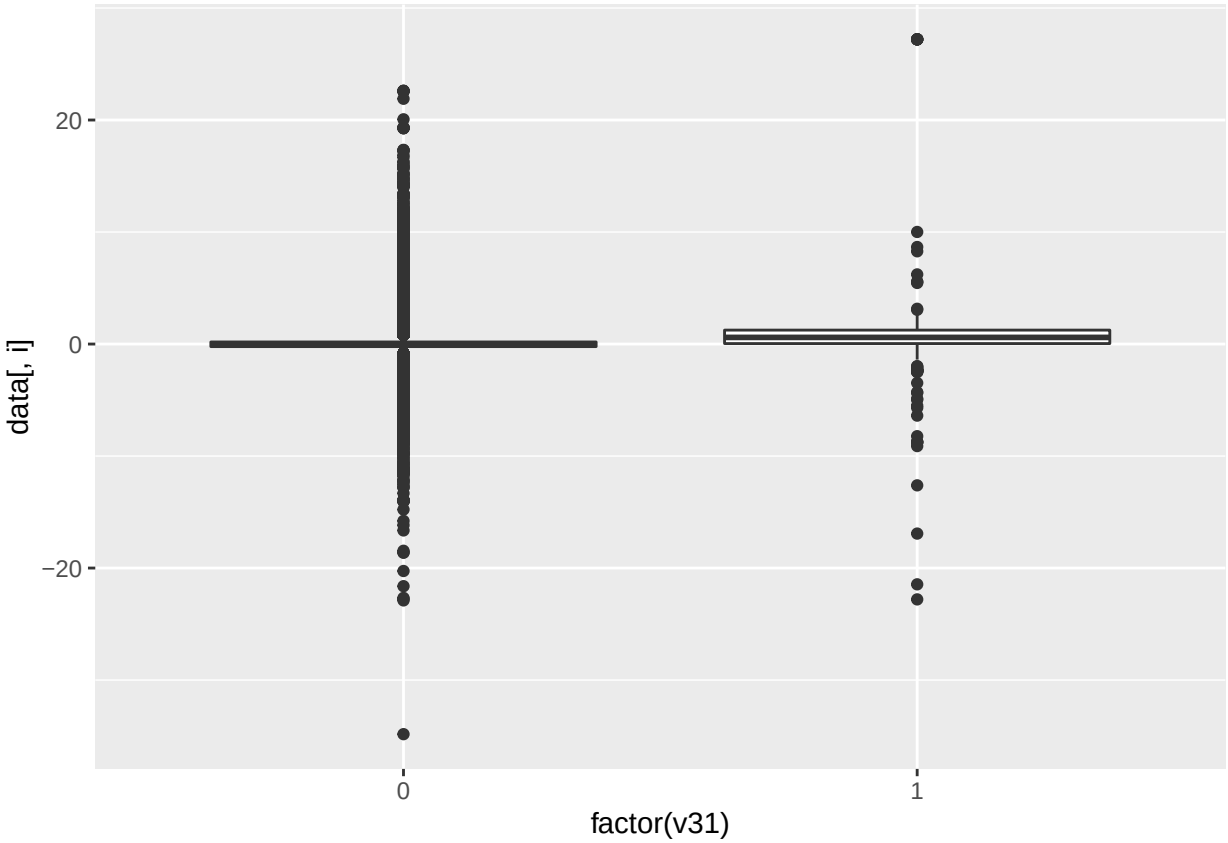

10
0
−10
data[, i]
0
1
factor(v31)

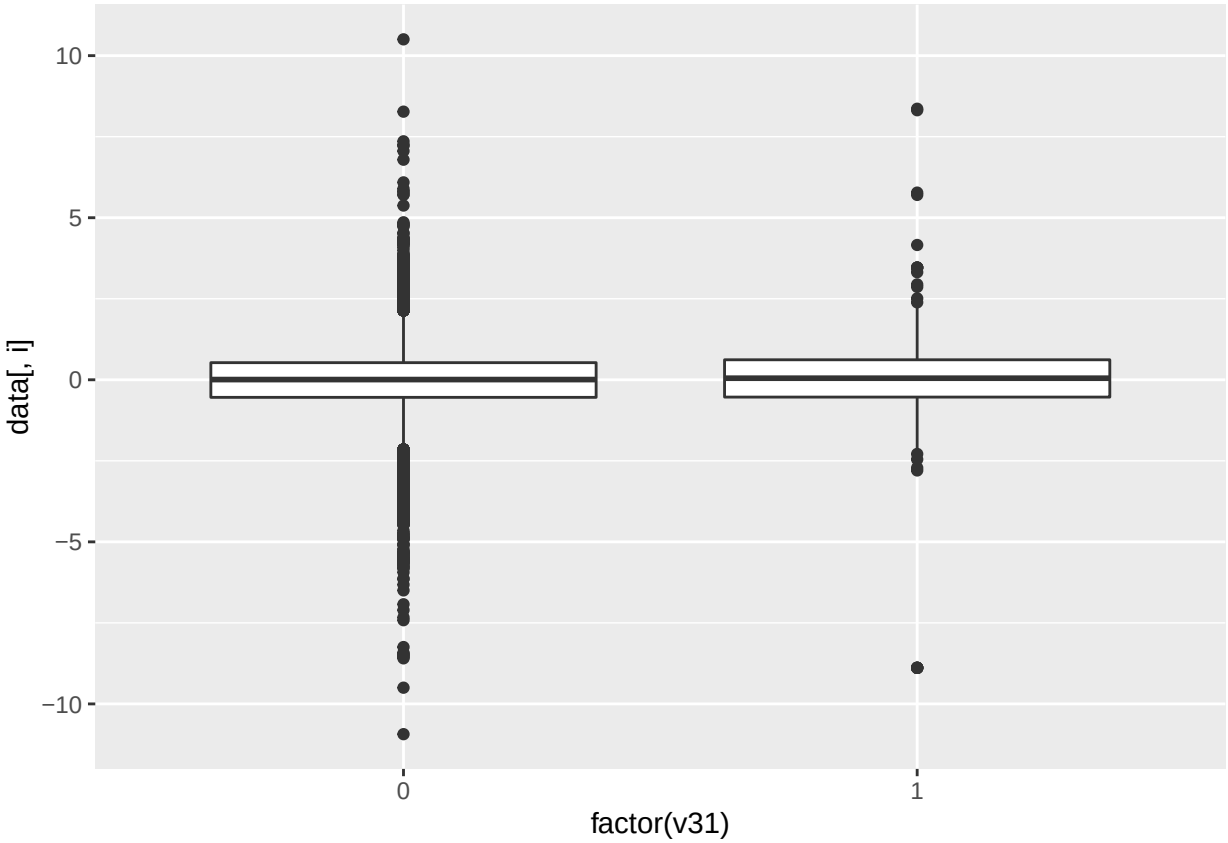

10
0
−10
−20
data[, i]
0
1
factor(v31)

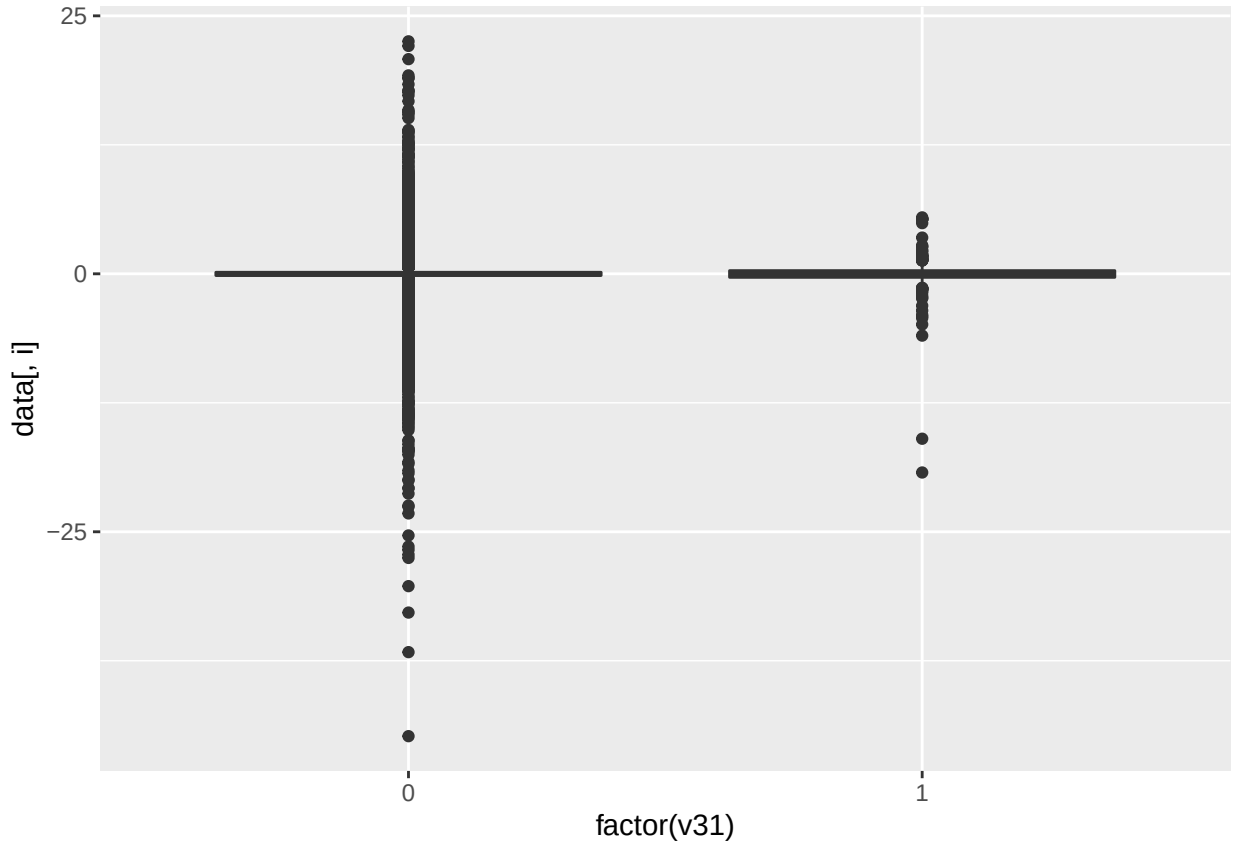

5
0
−5
−10
data[, i]
0
1
factor(v31)

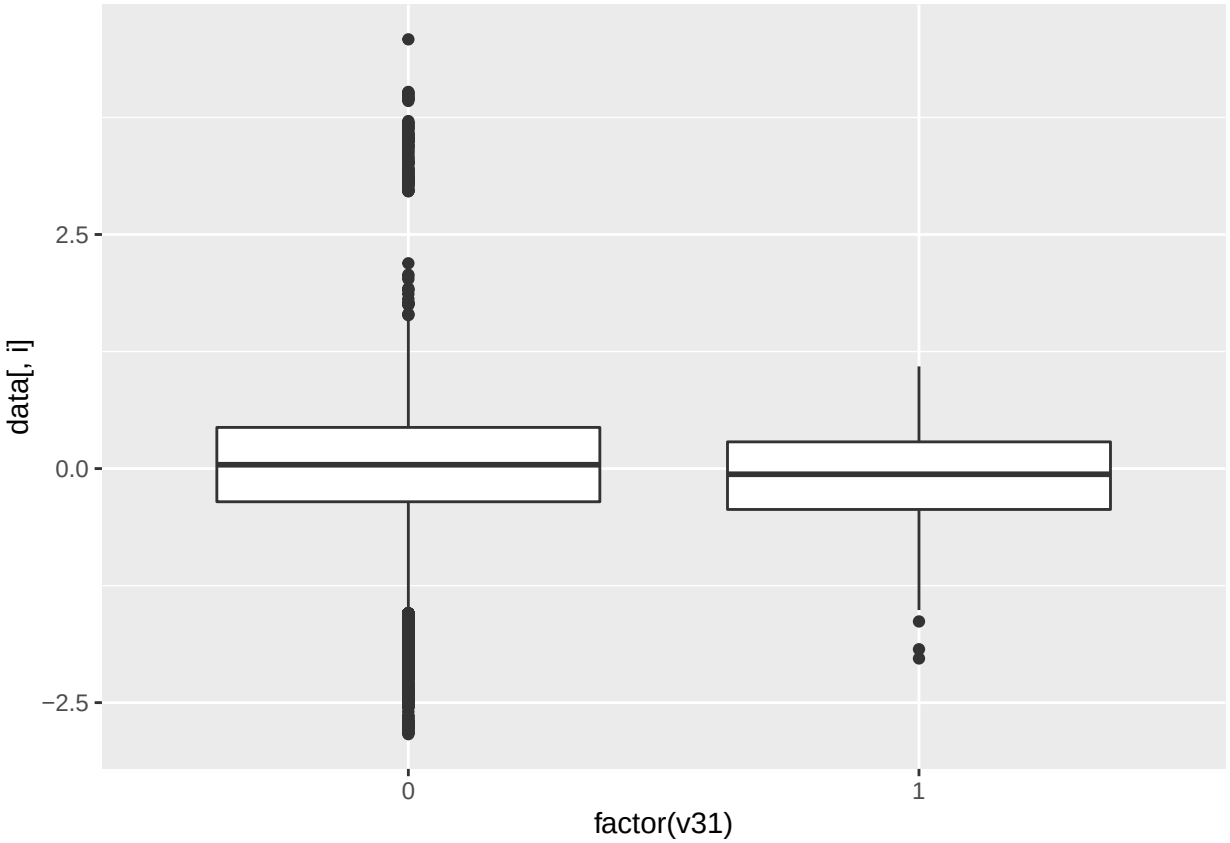

4
0
−4
data[, i]
0
1
factor(v31)

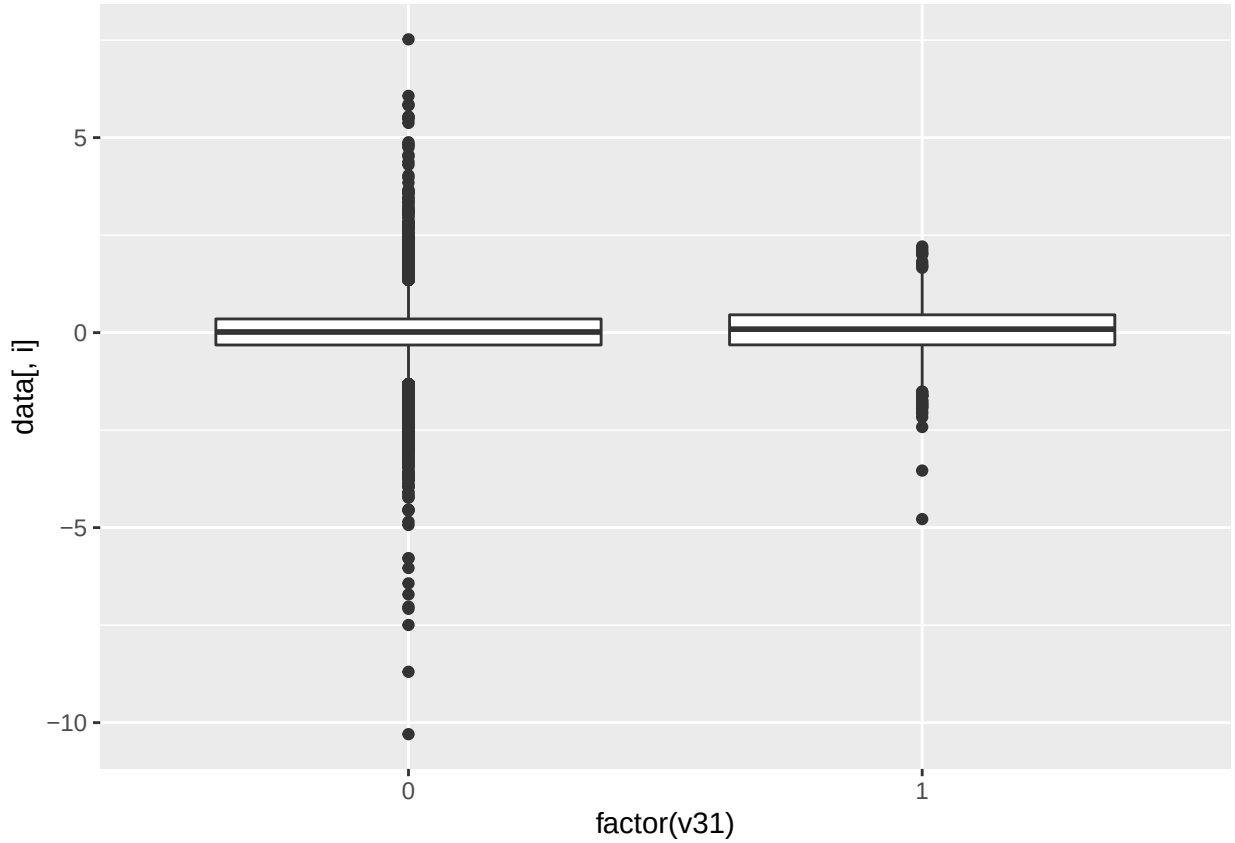

40
20
0
−20
−40
data[, i]
0
1
factor(v31)

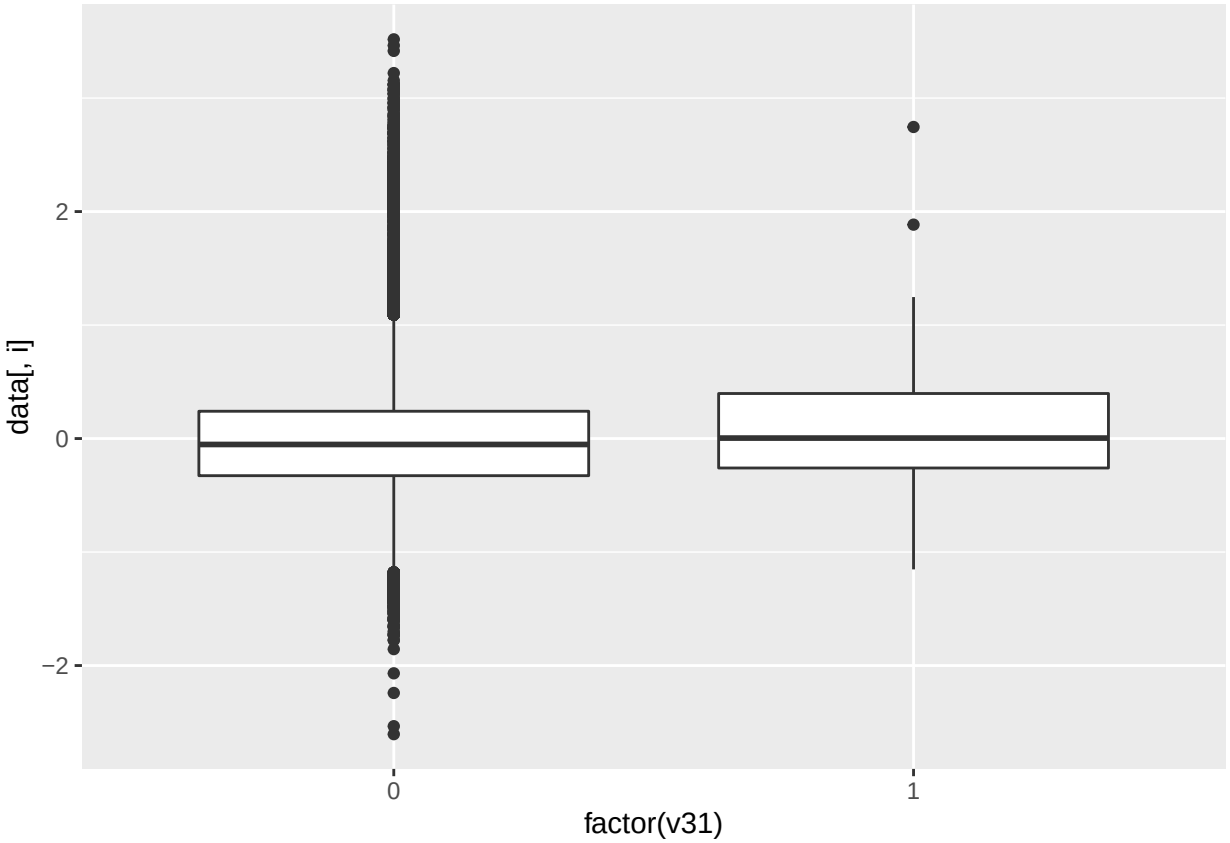

20
0
−20
data[, i]
0
1
factor(v31)

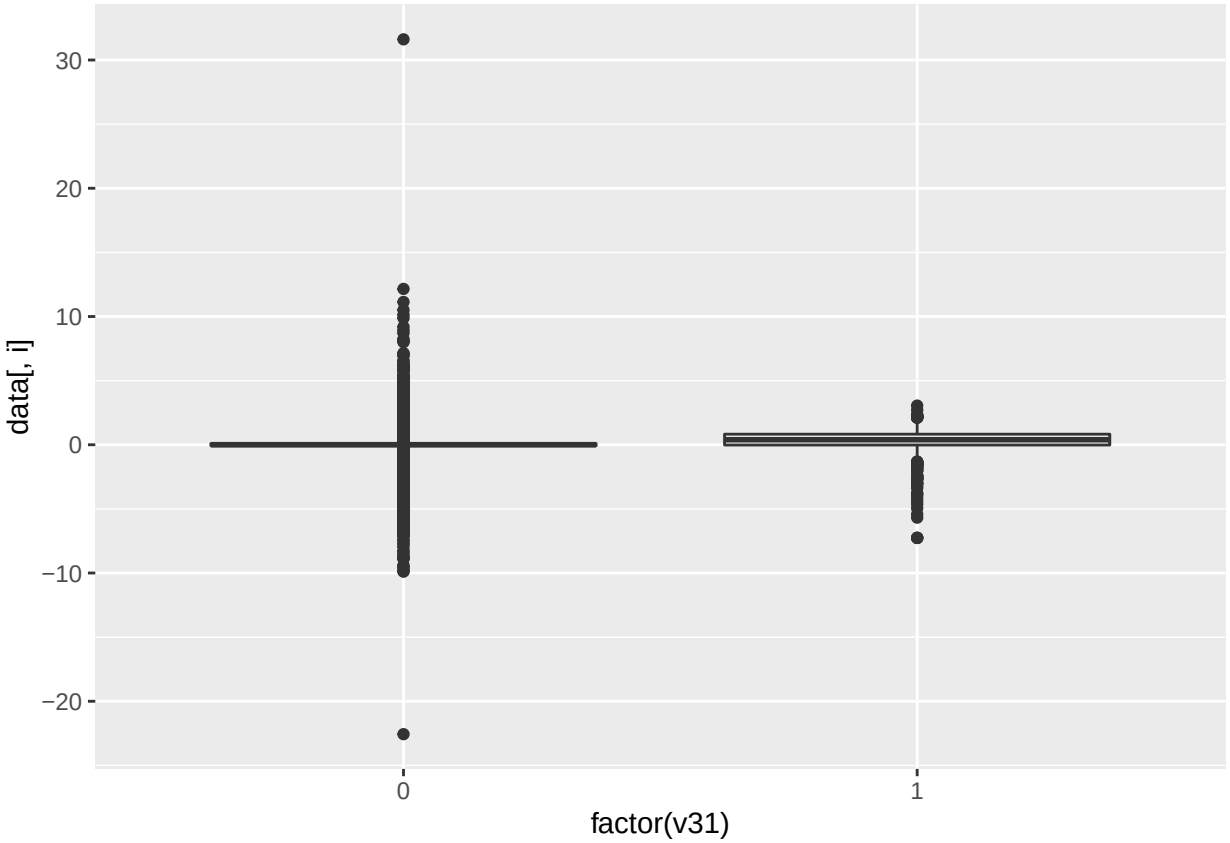

10
5
0
−5
−10
data[, i]
0
1
factor(v31)

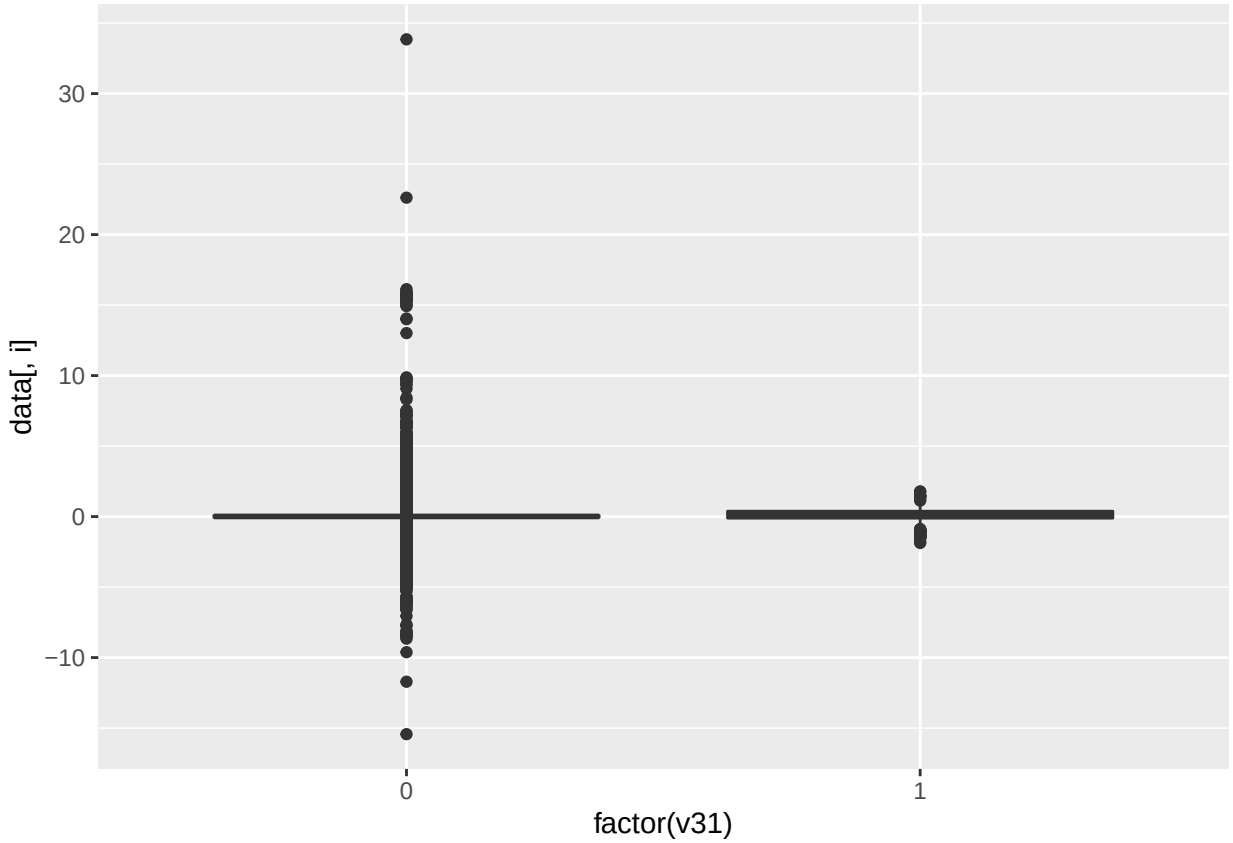

25
0
−25
data[, i]
0
1
factor(v31)

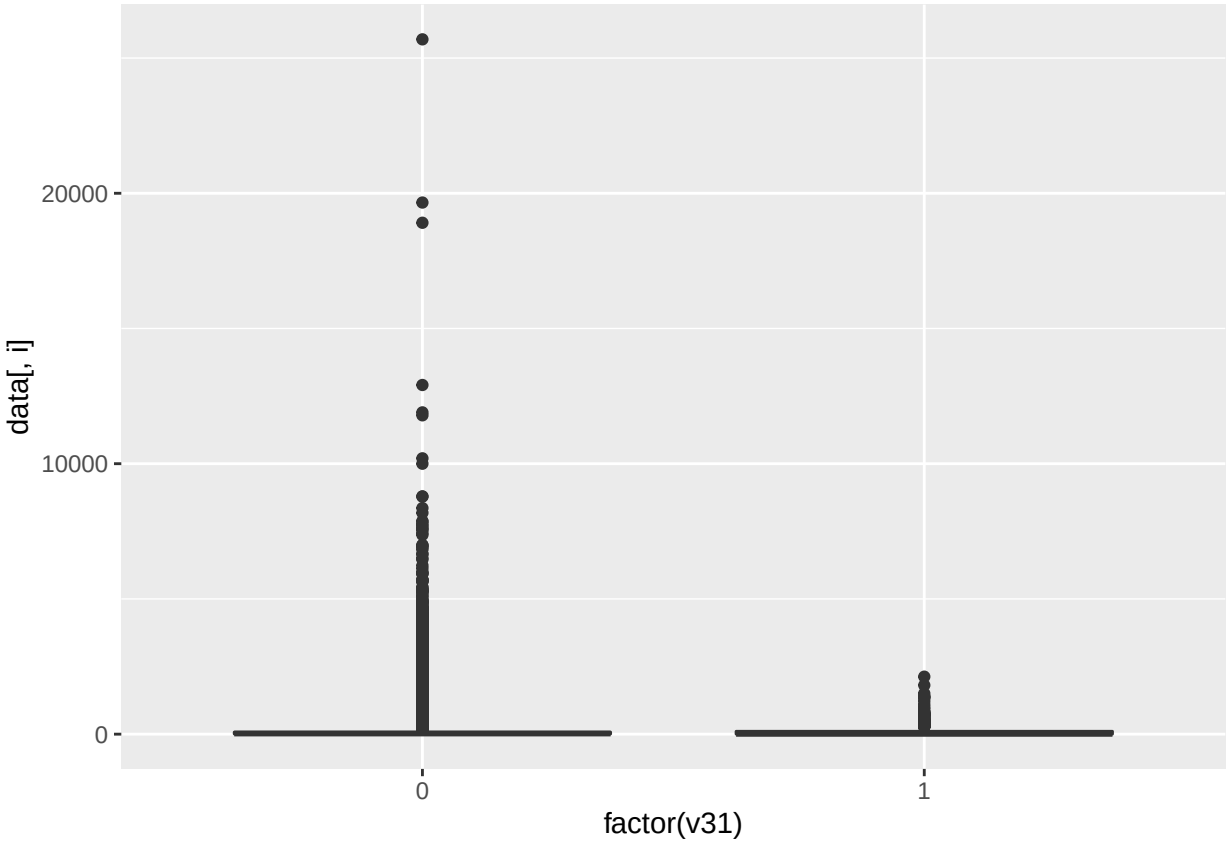

2.5
0.0
−2.5
data[, i]
0
1
factor(v31)

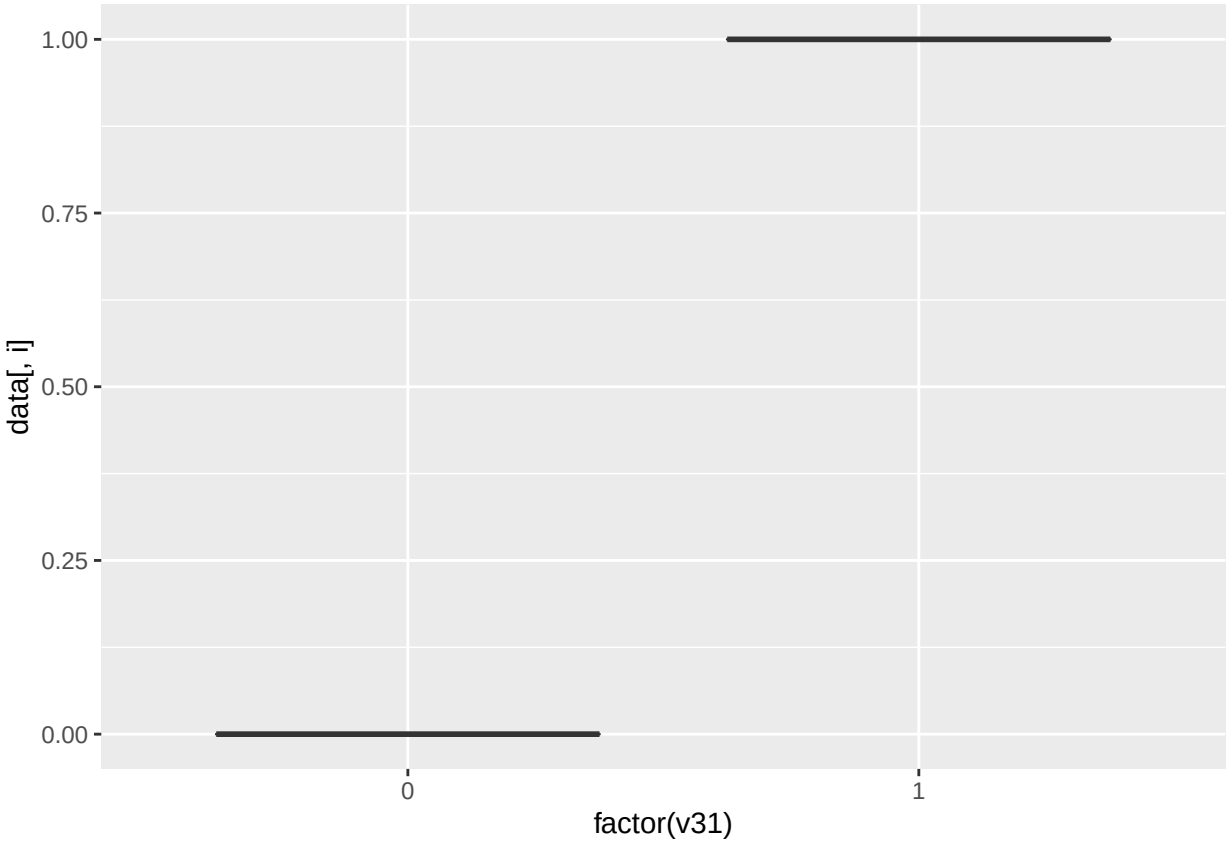

5
0
−5
−10
data[, i]
0
1
factor(v31)


data[, i]
2
0
−2
0
1
factor(v31)


30
20
10
0
−10
−20
data[, i]
0
1
factor(v31)


30
20
10
0
−10
data[, i]
0
1
factor(v31)

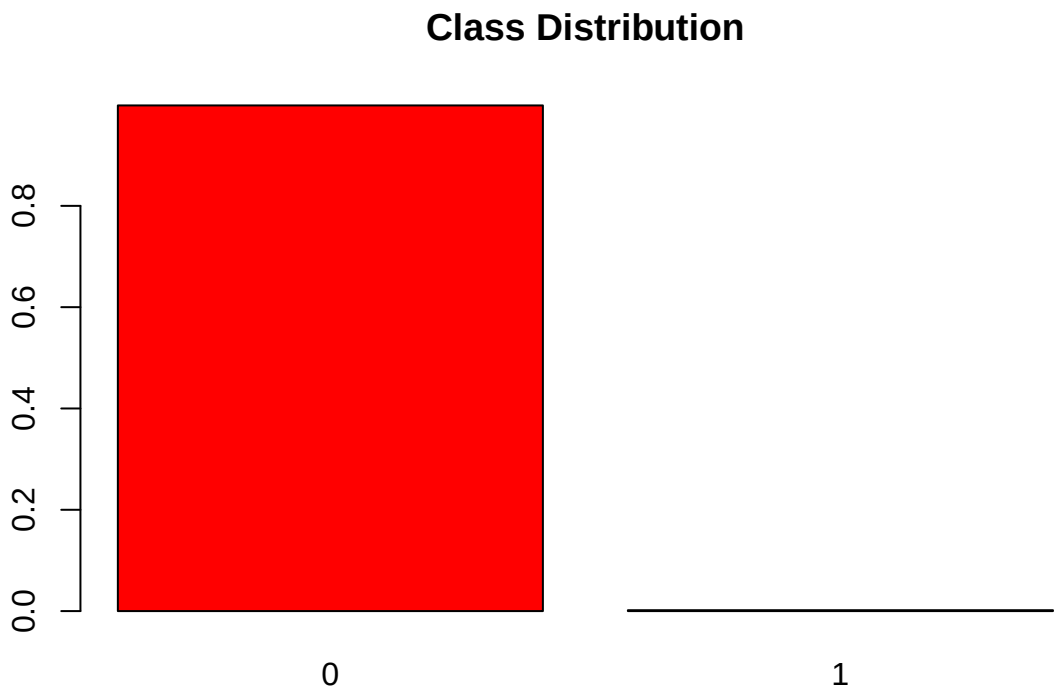

20000
10000
0
data[, i]
0
1
factor(v31)



Data are composed by 30 variables. The last column (v31) contains the label of the data. For each transaction, 29 variables are available and will be used in the modelising step.

An analysis of the class distribution (fraudulent or not) show an imbalanced between them. More than 99% of the transaction are not fraudulent.

```
barplot(prop.table(table(data$v31)),
        col = rainbow(2),
        main = "Class Distribution")
```



Thus, we have to make a specific processing before creating the model.

4. Modelisation