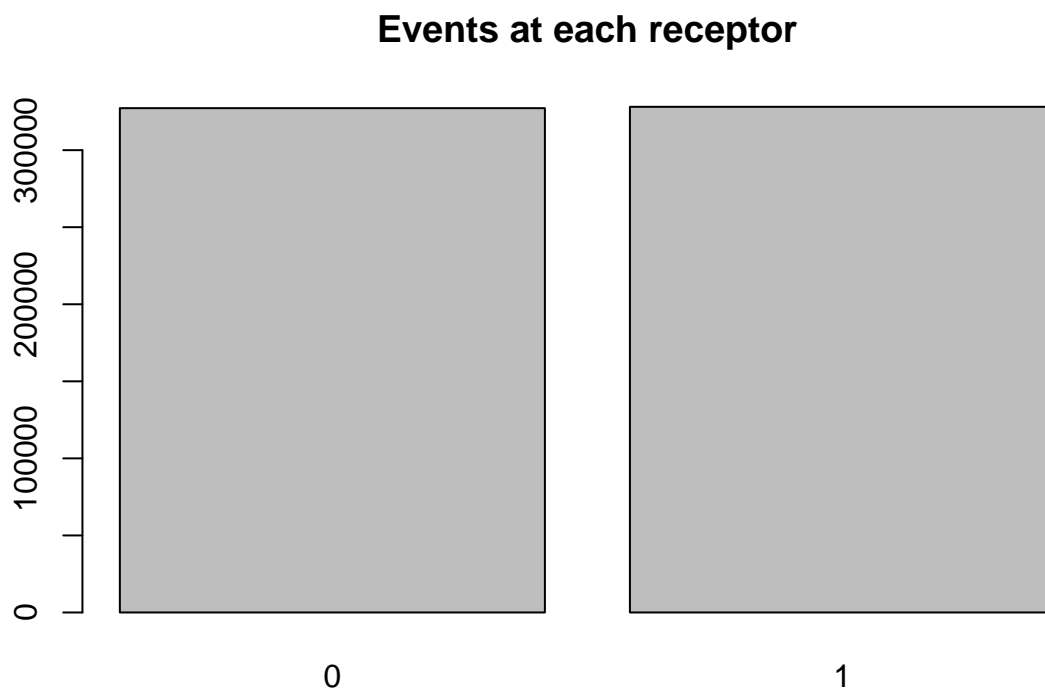


Introductory analysis

Situation: people going inside and outside of a discoteque through two gates

```
data <- read.csv("data.csv")
library(dplyr)

##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
library(tidyrr)
barplot(table(data$receptor), main="Events at each receptor")
```



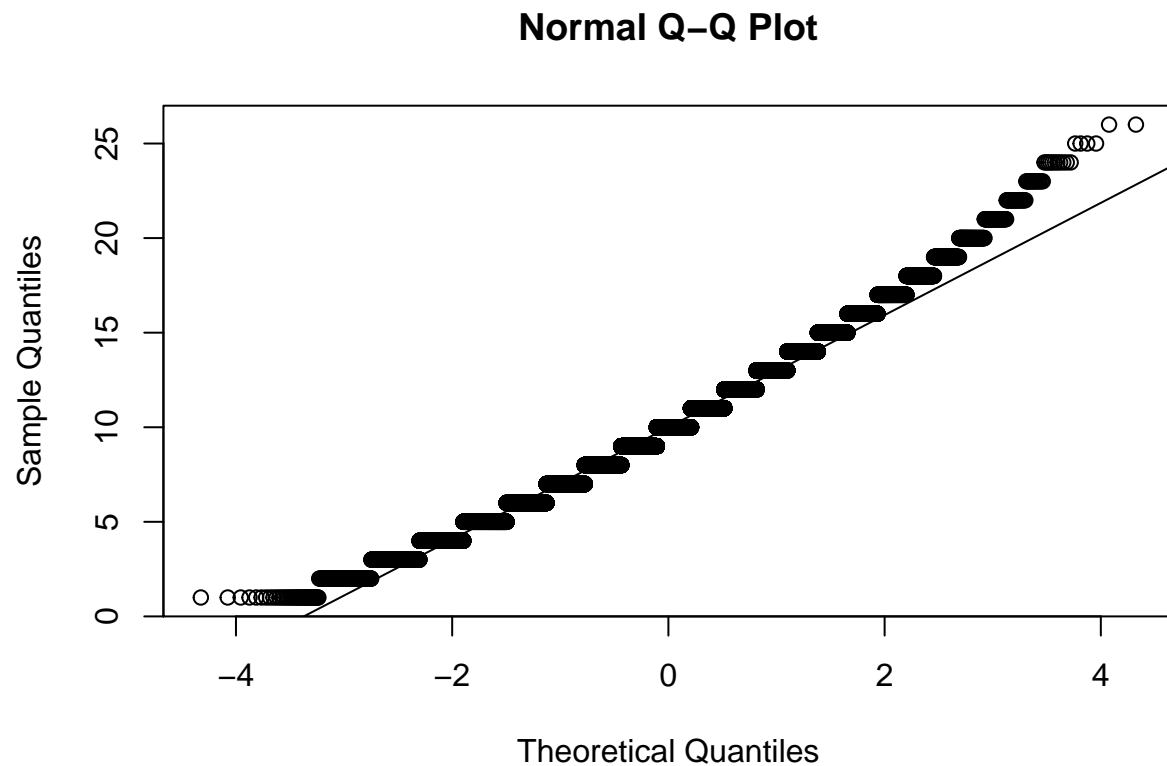
```
num_macs <- count(data, MAC)$n
mean(num_macs)
```

```
## [1] 10.00092
```

```
sd(num_macs)
```

```
## [1] 3.16109
```

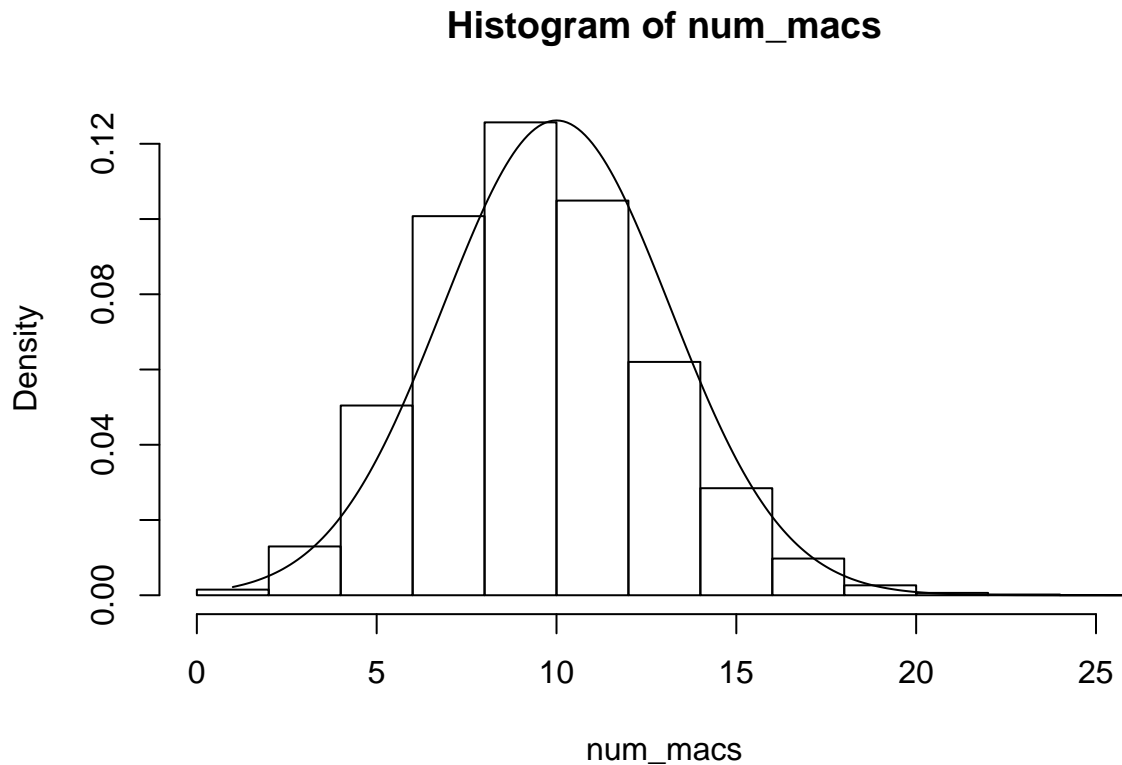
```
qqnorm(num_macs)
qqline(num_macs)
```



Macs distribution

It seems that the distribution the repetition of macs is a bit right-skewed. Seeing it in a histogram, we can confirm this:

```
hist(num_macs, freq = F)
lines(seq(min(num_macs), max(num_macs), 0.1), dnorm(seq(min(num_macs), max(num_macs), by = 0.1), mean =
```



Since this data is random and each timestamp a mac is generated with an uniform probability between all possibilities, and since the times a mac shows up is a sum of uniforms, due to the law of big numbers we can assume normality. TODO: normality test

Devices appearing only once

```
# Only once
sum(num_macs==1)
```

```
## [1] 40
```

```
# More than once
sum(num_macs!=1)
```

```
## [1] 65490
```

Evolution of people inside the system

```
# new column: is entering (T or F)
grouped <- data %>% group_by(MAC) %>% mutate(
  entering=as.logical(rank(timestamp) %% 2))
# Check if column "entering" changes each time a MAC appears
grouped[(grouped$MAC=="90b7"),]
```

```
## Source: local data frame [8 x 4]
```

```
## Groups: MAC [1]
##
##   timestamp    MAC receptor entering
##   <int> <fctr>    <int>    <lgl>
## 1      0    90b7      1     TRUE
## 2    8251    90b7      0    FALSE
## 3    98838    90b7      0     TRUE
## 4   158964    90b7      0    FALSE
## 5   179971    90b7      0     TRUE
## 6   202871    90b7      0    FALSE
## 7   353883    90b7      0     TRUE
## 8   619031    90b7      1    FALSE

# Amount of people in each timestamp: sum of "T" until then
accumulated <- ungroup(grouped) %>% mutate(
  inside=cumsum(entering)-cumsum(!entering))
head(accumulated,40)
```

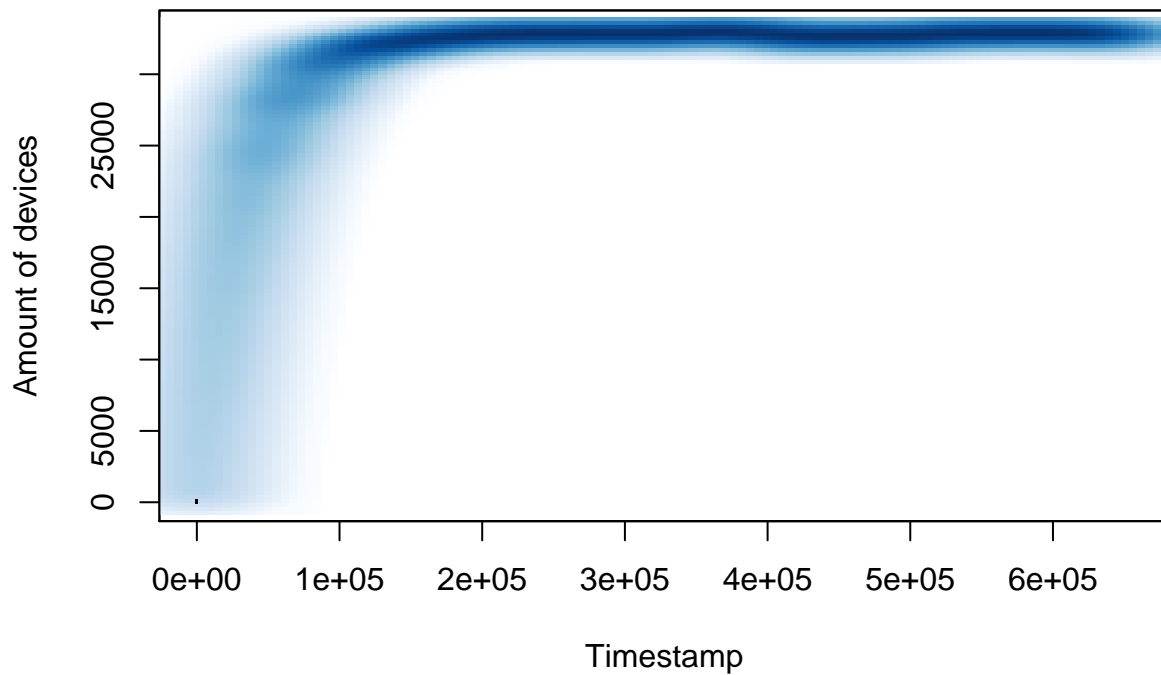
```
## # A tibble: 40 × 5
##   timestamp    MAC receptor entering inside
##   <int> <fctr>    <int>    <lgl> <int>
## 1      0    90b7      1     TRUE     1
## 2      1    21b2      0     TRUE     2
## 3      2    25b3      1     TRUE     3
## 4      3    e9db      1     TRUE     4
## 5      4    aee2      0     TRUE     5
## 6      5    0538      0     TRUE     6
## 7      6    c11f      1     TRUE     7
## 8      7    4883      0     TRUE     8
## 9      8    5d0b      1     TRUE     9
## 10     9    cafe      0     TRUE    10
## # ... with 30 more rows
```

```
# Displays the state of the system each time 90b7 appears
accumulated[(accumulated$MAC=="90b7"),]
```

```
## # A tibble: 8 × 5
##   timestamp    MAC receptor entering inside
##   <int> <fctr>    <int>    <lgl> <int>
## 1      0    90b7      1     TRUE     1
## 2    8251    90b7      0    FALSE    7334
## 3    98838    90b7      0     TRUE   31199
## 4   158964    90b7      0    FALSE  32439
## 5   179971    90b7      0     TRUE  32702
## 6   202871    90b7      0    FALSE  32788
## 7   353883    90b7      0     TRUE  33094
## 8   619031    90b7      1    FALSE  32972
```

```
smoothScatter(accumulated$timestamp, accumulated$inside, xlab = "Timestamp", ylab="Amount of devices",
```

Devices inside the system



Distribution of time inside the system

```
# data %>% filter(timestamp <= ts) %>% group_by(MAC)
time_cum <- function(data, mac, ts){
  mac_info <- data %>% select(timestamp, MAC, entering) %>% filter(timestamp <= ts & MAC == mac)
  time_inside <- NULL
  if(last(mac_info$entering)==FALSE){
    # that is, equal size of vectors, out of the system
    time_inside <- subset(mac_info, entering==F,select=timestamp) - subset(mac_info, entering==T, select=timestamp)
  } else{
    # that is, unequal size of vector, must add last timestamp to the data frame
    # last timestamp: that of the desired time window, or the whole measure
    time_inside <- rbind(subset(mac_info, entering==F,select=timestamp),min(max(data$timestamp),ts))- subset(mac_info, entering==T, select=timestamp)
  }
  time_inside
}
# using grouped causes infite processing time
time_cum(accumulated, "cafe", 1000000)
```

```
## timestamp
## 1 72989
## 2 14827
## 3 25577
## 4 3298
```

```
## 5    76680
## 6    41315
## 7    101142
```

```
# only 1000 macs are used, the full set is too big and takes too long
```

```
mac_list <- sample_n(unique(accumulated), 1000)$MAC
```

```
vec <- mapapply(time_cum, mac_list, MoreArgs = list(data=accumulated, ts=1000000))
```

```
hist(sapply(vec, sum), main = "Time inside the system", xlab="seconds", freq = T)
```

