A36_Hoermann

Paul Hörmann

1/6/2020

```
library(data.table)
library(DescTools)
```

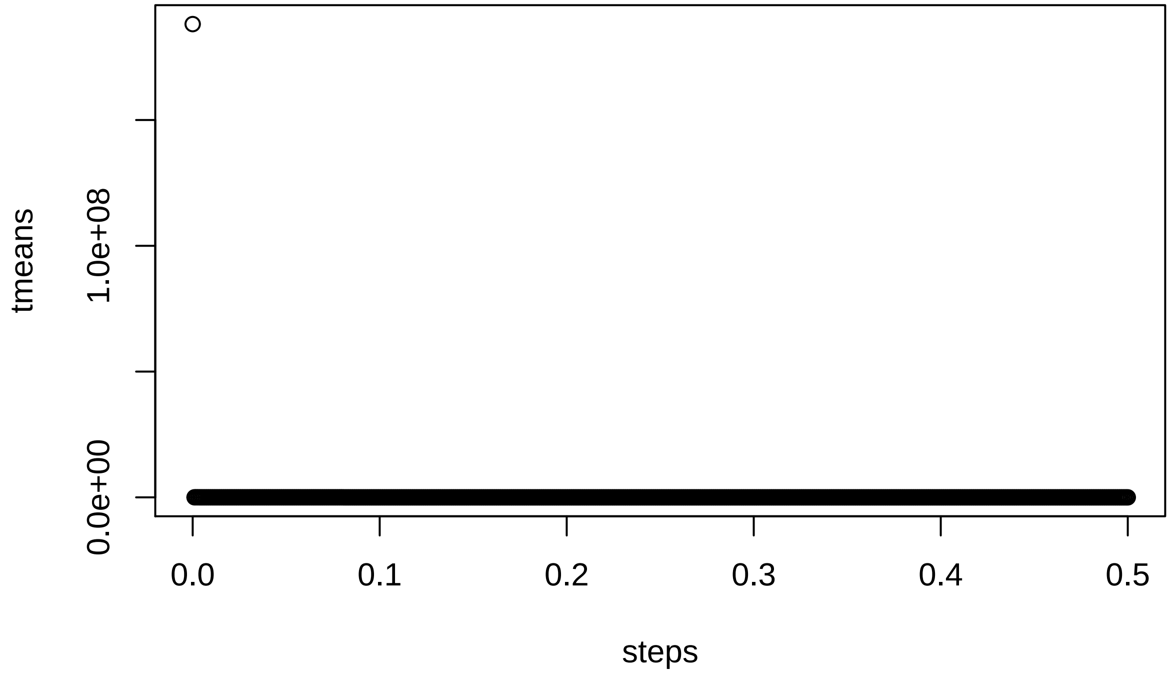
```
##
## Attaching package: 'DescTools'
```

```
## The following object is masked from 'package:data.table':
##
##     %like%
```

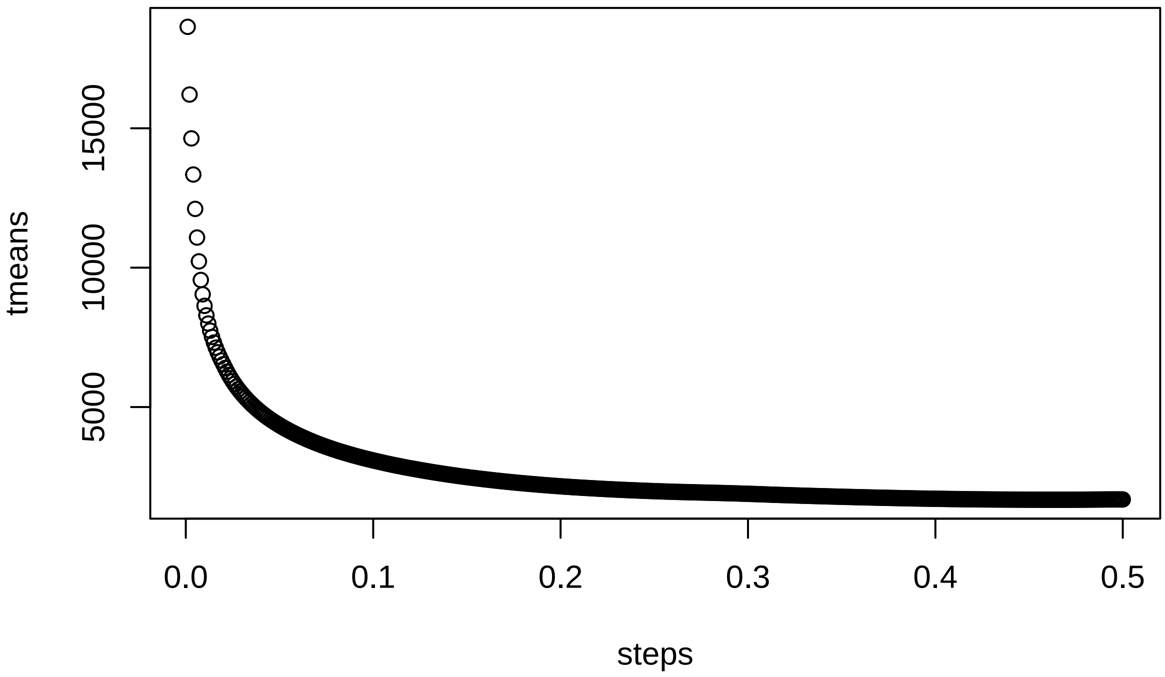
```
data = read.csv('sizes.csv')
data = as.numeric(data[,1])
data.table(Mean=mean(data), Median=median(data), IQR=IQR(data), SD=sd(data), Max=max(data), Min=min(data
```

```
##         Mean Median  IQR           SD          Max Min
## 1: 188127746   1685 4516 162684133224 1.407375e+14   0
```

```
trim_between = function(trim_factor) {
  mean(data, trim=trim_factor)
}
steps = seq(from = 0, to = 0.5, by = 0.001)
tmeans = lapply(steps, trim_between)
plot(steps, tmeans)
```



```
# start with 0.001 because with 0 the mean is extremely high as visible above
steps = seq(from = 0.001, to = 0.5, by = 0.001)
tmeans = lapply(steps, trim_between)
plot(steps, tmeans)
```



Looks like upside down logarithm.

What does trim do?

It cuts elements from left and right. A trim value of 0.1 means to cut of the right 10% of values and also the left 10% of values (Meaning 80% of values remain). This cuts off outsiders, as visible from the two graphs above.

Why non-symmetric?

Because then the mean would more or less stay the same.