

hw01.R

Zhenyok Nazedwox

Mon Dec 12 09:51:29 2016

```
# 01 data
data.df <- read.table("rnf6080.dat")
nrow(data.df)
```

```
## [1] 5070
```

```
ncol(data.df)
```

```
## [1] 27
```

```
colnames(data.df)
```

```
## [1] "V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8" "V9" "V10" "V11"
## [12] "V12" "V13" "V14" "V15" "V16" "V17" "V18" "V19" "V20" "V21" "V22"
## [23] "V23" "V24" "V25" "V26" "V27"
```

```
data.df[5,7]
```

```
## [1] 0
```

```
data.df[2,]
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20
## 2 60  4  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   V21 V22 V23 V24 V25 V26 V27
## 2   0   0   0   0   0   0   0
```

```
names(data.df) <- c("year", "month", "day", seq(0,23)) # присвоение имен столбцам таблицы
head(data.df) # первые 6 строк
```

```
##   year month day 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
## 1   60     4   1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 2   60     4   2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 3   60     4   3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 4   60     4   4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5   60     4   5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 6   60     4   6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##   22 23
## 1   0  0
## 2   0  0
## 3   0  0
## 4   0  0
## 5   0  0
## 6   0  0
```

```
tail(data.df) # последние 6 строк
```

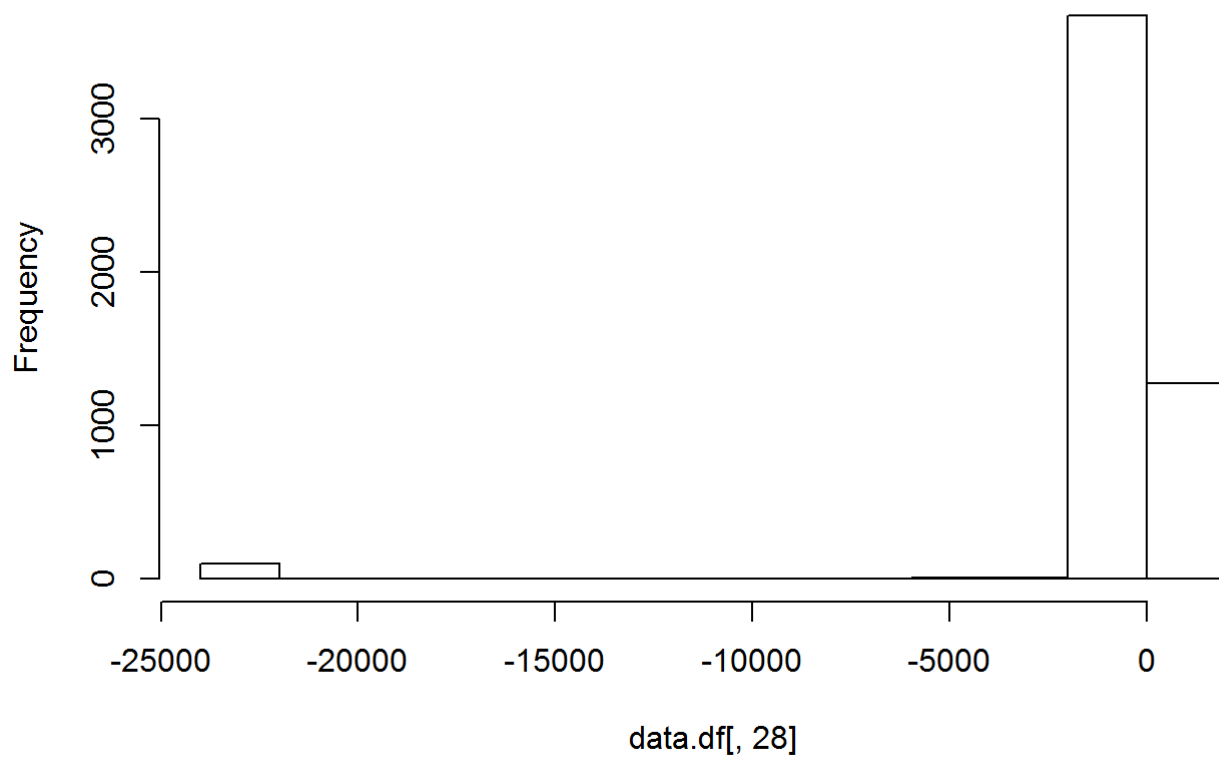
```
##   year month day 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
## 5065   80    11 25 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5066   80    11 26 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5067   80    11 27 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5068   80    11 28 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5069   80    11 29 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5070   80    11 30 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##   21 22 23
## 5065   0  0  0
## 5066   0  0  0
## 5067   0  0  0
## 5068   0  0  0
## 5069   0  0  0
## 5070   0  0  0
```

```
head(data.df)[,4:27] # последние 24 столбца показывают осадки в соответствующий час с уток
```

```
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
## 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

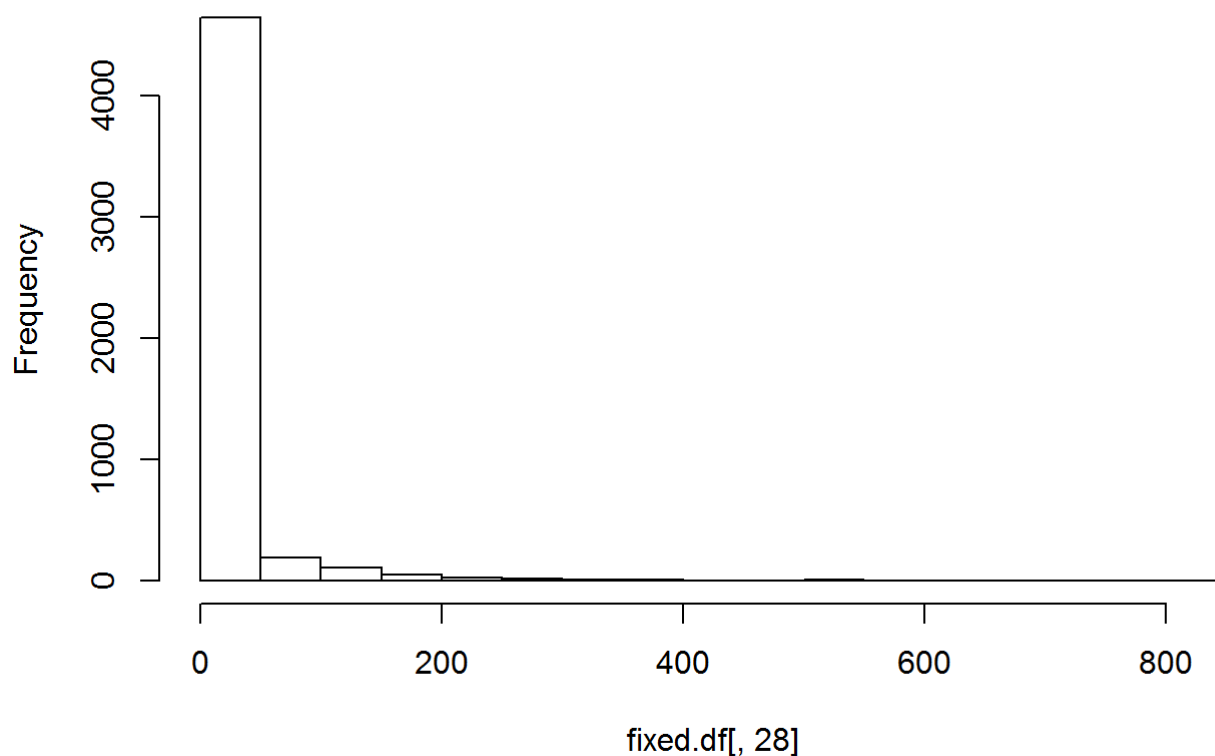
```
for(i in 1:nrow(data.df)) {
  data.df$daily[i] <- sum(data.df[i,4:27])
}
hist(data.df[,28]) # досадная подстава из-за 6/20/1980
```

Histogram of data.df[, 28]



```
fixed.df <- data.df
fixed.df$daily[fixed.df$daily < 0] <- 0 # нет смысла в отрицательных значениях выпадений осадков
hist(fixed.df[,28])
```

Histogram of fixed.df[, 28]



```
# 02 syntax
v <- c("4", "8", "15", "16", "23", "42")
max(v)
```

```
## [1] "8"
```

```
sort(v)
```

```
## [1] "15" "16" "23" "4"  "42" "8"
```

```
# sum(v) # выдает ошибку
# все функции ошибочны, так как в массиве представлены в типе chr
# с каждым элементом массива функции работают посимвольно

v2 <- c("5",7,12)
# v2[2] + v2[3] # опять же вместо чисел символы

df3 <- data.frame(z1="5",z2=7,z3=12)
df3[1,2] + df3[1,3] # сложили 2 и 3 элементы строки 1 в дата фрейме, то есть 7 и 12,
# все хорошо, получили 19
```

```
## [1] 19
```

```
l4 <- list(z1="6", z2=42, z3="49", z4=126)
l4[[2]] + l4[[4]]
```

```
## [1] 168
```

```
# 14[2] + 14[4] # выдает ошибку  
# с одними квадратными скобками мы получаем имя переменной списка и ее значение  
# с двумя квадратными скобками мы получаем только значение переменной списка  
# в первом случае мы взяли 42 и 126 и сложили их, получили 168  
# во втором функция выдала ошибку, так как она приняла неверные для сложения аргументы  
14[[2]]
```

```
## [1] 42
```

```
14[2]
```

```
## $z2  
## [1] 42
```

```
# 03 functions  
seq(1,10000,372)
```

```
## [1] 1 373 745 1117 1489 1861 2233 2605 2977 3349 3721 4093 4465 4837  
## [15] 5209 5581 5953 6325 6697 7069 7441 7813 8185 8557 8929 9301 9673
```

```
seq(1,10000, length.out = 50)
```

```
## [1] 1.0000 205.0612 409.1224 613.1837 817.2449 1021.3061  
## [7] 1225.3673 1429.4286 1633.4898 1837.5510 2041.6122 2245.6735  
## [13] 2449.7347 2653.7959 2857.8571 3061.9184 3265.9796 3470.0408  
## [19] 3674.1020 3878.1633 4082.2245 4286.2857 4490.3469 4694.4082  
## [25] 4898.4694 5102.5306 5306.5918 5510.6531 5714.7143 5918.7755  
## [31] 6122.8367 6326.8980 6530.9592 6735.0204 6939.0816 7143.1429  
## [37] 7347.2041 7551.2653 7755.3265 7959.3878 8163.4490 8367.5102  
## [43] 8571.5714 8775.6327 8979.6939 9183.7551 9387.8163 9591.8776  
## [49] 9795.9388 10000.0000
```

```
rep(1:5,times=3) # повторяет всю последовательность целиком 3 раза
```

```
## [1] 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5
```

```
rep(1:5, each=3) # повторяет каждый элемент последовательности по 3 раза
```

```
## [1] 1 1 1 2 2 2 3 3 3 4 4 4 5 5 5
```