A31_Hoermann

Paul Hörmann

12/5/2019

```
library(data.table)
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
set.seed(1)
v3 = rnorm(n = 50, mean = 30, sd = 3)
v4 = runif(n = 50, min = 20, max = 40)
```

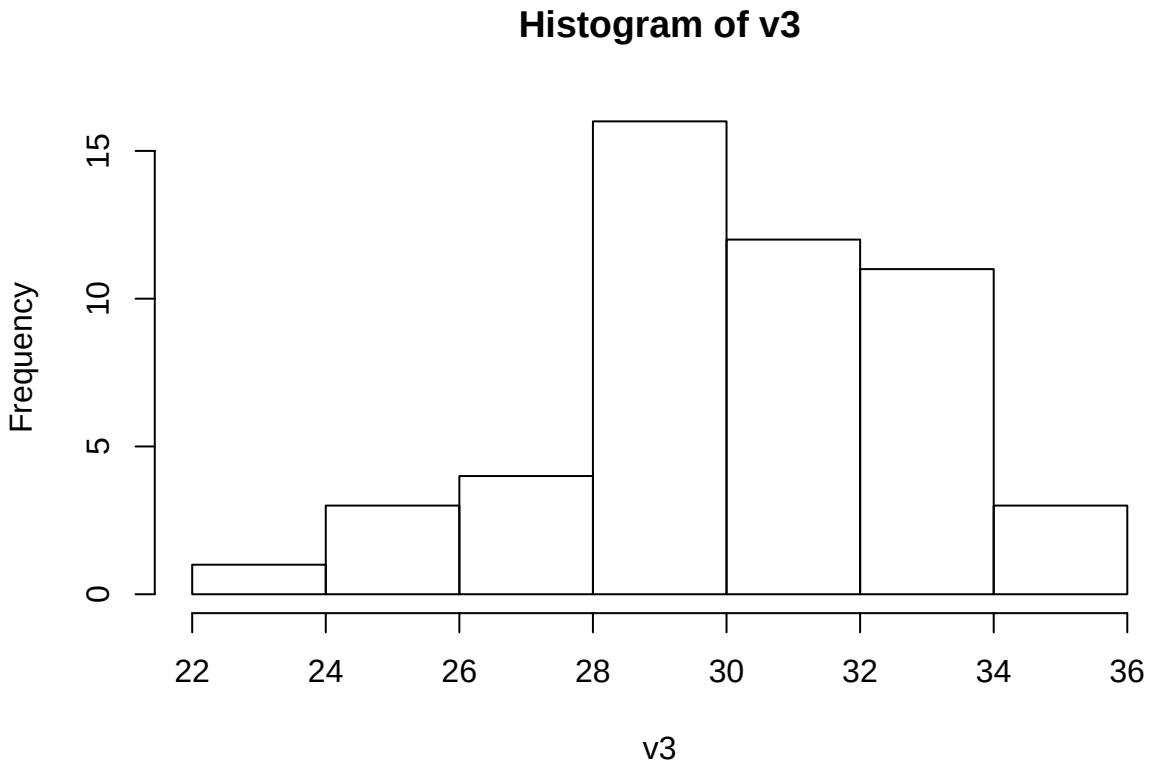
```
head(data.table(v3=v3, v4=v3))
```

```
##          v3       v4
## 1: 28.12064 28.12064
## 2: 30.55093 30.55093
## 3: 27.49311 27.49311
## 4: 34.78584 34.78584
## 5: 30.98852 30.98852
## 6: 27.53859 27.53859
```

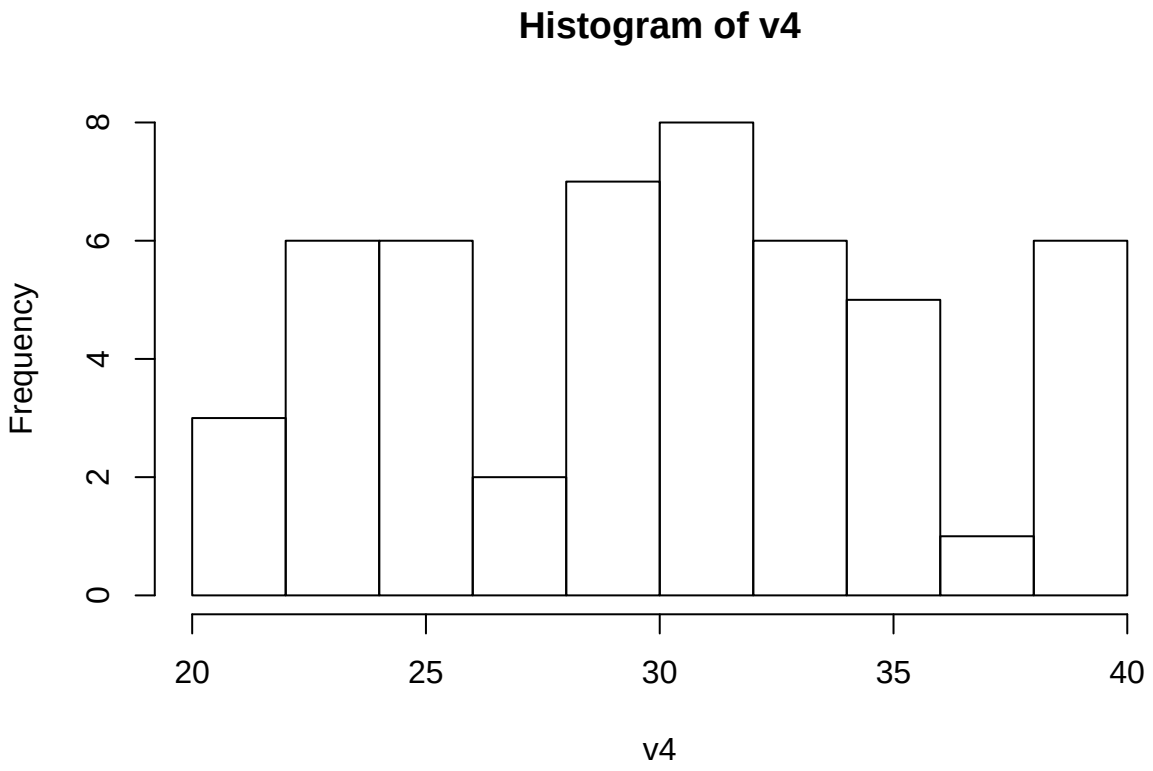
```
tail(data.table(v3=v3, v4=v3))
```

```
##          v3       v4
## 1: 27.93373 27.93373
## 2: 27.87751 27.87751
## 3: 31.09375 31.09375
## 4: 32.30560 32.30560
## 5: 29.66296 29.66296
## 6: 32.64332 32.64332
```

```
h3 = hist(v3)
```


Histogram of v3
Frequency
0
5
10
15
22
24
26
28
30
32
34
36
v3

```
h4 = hist(v4)
```


Histogram of v4
Frequency
0
2
4
6
8
20
25
30
35
40
v4

Test for uniform distribution of h3

```
bincount = length(h3$counts)
chisq.test(h3$counts, p=rep(1/bincount, bincount))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  h3$counts
## X-squared = 27.84, df = 6, p-value = 0.0001007
```

P value of around 0 seems valid, the values are indeed not uniformly distributed,

Function to get p vector for histogram breaks

```
getProbs = function (breaks, mean, sd) {
  sub = 0
  len = length(breaks) - 1
  p = integer(len)
  for (i in 1:(len - 1)) {
    p[i] = pnorm(q = breaks[i + 1], mean = mean, sd = sd) - sum(p)
  }
  p[len] = 1 - sum(p)
  p
}
```

Test for normal distribution of h3

```
chisq.test(h3$counts, p = getProbs(h3$breaks, 30, 3))
```

```
## Warning in chisq.test(h3$counts, p = getProbs(h3$breaks, 30, 3)): Chi-
## squared approximation may be incorrect
```

```
##
##  Chi-squared test for given probabilities
##
## data:  h3$counts
## X-squared = 4.793, df = 6, p-value = 0.5706
```

P value of 0.6 makes sense, as the values are indeed normally distributed but rather heavy on the right side, as visible from the histogram.

Test for uniform distribution of h4

```
bincount = length(h4$counts)
chisq.test(h4$counts, p=rep(1/bincount, bincount))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  h4$counts
## X-squared = 9.2, df = 9, p-value = 0.419
```

P value of around 0.4 makes sense, though randomness got a hit on the equalness of distribution.

Test for normal distribution of h4

```
chisq.test(h4$counts, p = getProbs(h4$breaks, 30, 3))
```

```
## Warning in chisq.test(h4$counts, p = getProbs(h4$breaks, 30, 3)): Chi-
## squared approximation may be incorrect
```

```
##
##  Chi-squared test for given probabilities
##
## data:  h4$counts
## X-squared = 255.99, df = 9, p-value < 2.2e-16
```

P value of basically 0 makes sense, as the values are not normally distributed.