

Зад. 9.1 Създайте 15 числа, които са нормално разпределени със средно 10 и стандартно отклонение 5. Направете един z-test за откриване доверителен интервал със ниво на доверие 95%. Верен ли е доверителния интервал?

Реш:

```
mu=10; s=5  
X=rnorm(n=15, mean=mu, sd=s)
```

Тъй като знаем, че разпределението е нормално, може директно да приложим z-test

```
res=simple.z.test(X, sigma=s, conf.level=0.95)  
left=res[1]; right=res[2]; left; right  
# [1] 8.177146  
# [1] 13.23775
```

```
if(left<=mu && mu<=right) print("good CI") else print("bad CI")  
# [1] "good CI"
```

Зад. 9.2 Повторете горното 100 пъти. Изчислете какъв процент от всички доверителни интервали - средното попада в доверителения интервал с ниво на доверие 95%.

z-test брой

I-ви начин:

```
sol=function(N=100, n=15, mu=10, s=5){  
  cnt=0  
  for(i in 1:N){  
    X=rnorm(n=n, mean=mu, sd=s)  
    res=simple.z.test(X, sigma=s, conf.level=0.95)  
    left=res[1]; right=res[2]  
    if(left<=mu && mu<=right) cnt=cnt+1  
  }  
  cnt/N  
}
```

II-ри начин:

```
sol=function(N=100, n=15, mu=10, s=5){  
  f=function() mean(rnorm(n=n, mean=mu, sd=s))  
  xbar=simple.sim(n=N, f)  
  SE=s/sqrt(n)  
  alpha=0.05  
  zstar=qnorm(p=1-alpha/2)  
  sum(abs(xbar-mu)<zstar*SE)/N  
}
```

Зад. 9.3 Направете t-test на същите данни от предходната задача. Верен доверителен интервал ли се получава? Коментирайте връзката между доверителните интервали.

Реш:

```
s=5  
simple.z.test(X, sigma=s, conf.level=0.95)  
# [1] 6.650761 11.711366
```

Тъй като знаем, че разпределението е нормално, може да използваме t-test директно.

```
res=t.test(X, conf.level=0.95)  
left=res$conf.int[1]; right=res$conf.int[2]  
left; right
```

```
# [1] 5.921536
```

```
# [1] 12.44059
```

Получаваме по-широк доверителен интервал от z-test-a, тъй като t-test-a прави оценка на стандартното отклонение, докато z-test-a го взима като даденост (т.е. го взима точно).

```
# t-test брой
```

```
N=100; n=15; mu=10; s=5
```

```
left=0; right=0
```

```
alpha=0.05
```

```
zstar=qnorm(1-alpha/2)
```

```
for(i in 1:N){
```

```
    X=rnorm(n=n, mean=mu, sd=s)
```

```
    SE=sd(x)/sqrt(n)
```

```
    M=mean(X)
```

```
    left[i]=M-zstar*SE
```

```
    right[i]=M+zstar*SE
```

```
}
```

```
sum(left<=10 && 10<=right)/N
```

```
# [1] 0.89
```

Зад. 9.4 Намерете 80% и 90% -тови доверителни интервали за медианата на `exes.pay` данните.

Реш:

```
qqnorm(exes.pay)
```

```
qqline(exes.pay)
```

Очевидно данните `exes.pay` не са нормално разпределени, но нека все пак се обединим в това като направим тест за нормалност.

```
if(sh$p.value>0.05) print("normal") else print("not normal")
```

```
# [1] "not normal"
```

Сега вече сме сигурни, че не са нормално разпределени, което можеше да го видим и чрез `qqplot.das(exes.pay, "norm")` и да проверим дали опашките се побират в доверителните интервали за нормалното разпределение.

```
w80=wilcox.test(exes.pay, conf.int=T, conf.level=0.80)
```

```
left80=w80$conf.int[1]; right80=w80$conf.int[2]
```

```
w95=wilcox.test(exes.pay, conf.int=T, conf.level=0.95)
```

```
left95=w95$conf.int[1]; right95=w95$conf.int[2]
```

```
left80; right80;
```

```
# [1] 27.00005
```

```
# [1] 31.49996
```

```
left95; right95
```

```
# [1] 25.99998
```

```
# [1] 32.99994
```

```
median(exes.pay)
```

```
# [1] 27
```

Зад. 9.5 Базата данни rat съдържа записи за времето в което плъхове са оцелявали. Направете t-test за средното ако данните позволяват правенето на подобен тест. Ако не, аргументирайте се защо не позволяват.

Реш:

```
qqnorm(rat)
qqline(rat)
```

```
sh=shapiro.test(rat)
if(sh$p.value>0.05) print("normal") else print("not normal")
# [1] "normal"
```

Сега вече сме сигурни, че не са нормално разпределени, което можеше да го видим и чрез qqplot.das(exec.pay, "norm") и да проверим дали опашките се побират в доверителните интервали за нормалното разпределение. Може да приложим t-test за данните.

```
res=t.test(rat, conf.level=0.95)
left=res$conf.int[1]; right=res$conf.int[2]
left; right
# [1] 96.69997
# [1] 130.2
mean(rat)
# [1] 113.45
```

Зад. 9.6 Повторете същото от предходната задача, но с данните puerto, който съдържа седмичните приходи на пуерториканци в Маями.

Реш:

```
qnorm(puerto)
qqline(puerto)
sh=shapiro.test(puerto)
if(sh$p.value>0.05) print("normal") else print("not normal")
# [1] "not normal"
qqplot.das(puerto, "norm") # за да видим, че опашките на разпределението на нашите данни
излизат извън доверителните интервали на нормалното разпределение

length(puerto)
# [1] 50
```

Данните обаче имат 50 наблюдения и не са толкова малко. Т.е. ако стандартното отклонение е добре дефинирано (е крайно), по ЦГТ може да получим:

```
mean(puerto)
# [1] 277.5
res=t.test(puerto, conf.level=0.95)
left=res$conf.int[1]; right=res$conf.int[2]
left; right
# [1] 255.9244
# [1] 299.0756
```

Зад. 9.7 Медианата може да бъде не лоша оценка за центъра на извадката. Може да искаме да намерим доверителен интервал и за нея. Намерете 90%-тов доверителен интервал за медианата на данните `malpract`, които съдържат размера на наградите за професионална набрежност. Коментирайте защо това разпределение не се поддава на `z-test` или на `t-test`.

Реш:

```
X=malpract
```

```
qqnorm(X)
qqline(X)
sh=shapiro.test(X)
if(sh$p.value>0.05) print("normal") else print("not normal")
# [1] "not normal"
```

Това, че данните ни не са нормални, освен, чрез теста, може да го видим и като направим `qqplot.das(X, "norm")` и направим наблюдението, че опашките излизат извън доверителните интервали на нормалното разпределение. Освен това:

```
length(X)
# [1] 17
```

Т.е. броя на наблюденията в извадката ни са твърде малко и не може да използваме нормалното приближение по ЦГТ.

```
w=wilcox.test(X, conf.int=T, conf.level=0.90)
left=w$conf.int[1]; right=w$conf.int[2]
left; right
# [1] 385
# [1] 1325
```