

Комп'ютерна статистика

Шкляр Ірина, ксад 4 курс

Робота 1, варіант 4

1.

(a) `n<-500 # кількість чисел`
`a<-7^5 # RANDU параметри`
`c0<-0`
`m<-2^31 - 1`
`l<-numeric(n) # цілочислова послідовність`
`l[1]<-2^15+2`
`for(i in 2:n){`
 `l[i]<-(a*l[i-1]+c0)%% m`
`}`
`x1<-l/m # псевдовипадкові числа`
`x1[1:20]`

або:

```
park_miller <- function() {  
  m <- 2^31  
  a <- 7^5  
  x <- 2^15  
  x <- (a * x) %% m  
  return(x / m)  
}
```

`# Генерація послідовностей`
`set.seed(123) # Задаємо початковий стан генератора`
`seq1 <- replicate(500, park_miller())`

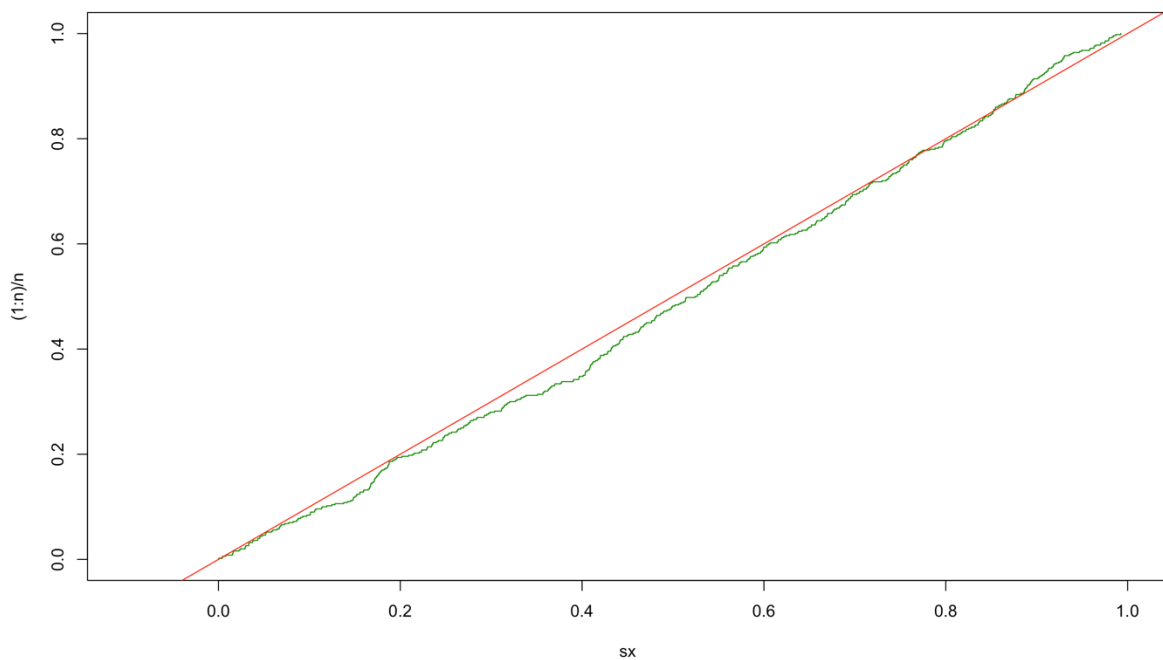
(b) `n<-500 # кількість чисел`
`a<-75831 # RANDU параметри`
`c0<-0`
`m<-2^31`
`l<-numeric(n) # цілочислова послідовність`
`l[1]<-2^15+2`
`for(i in 2:n){`
 `l[i]<-(a*l[i-1]+c0)%% m`
`}`
`x2<-l/m # псевдовипадкові числа`
`x2[1:20]`

або ось така функція:

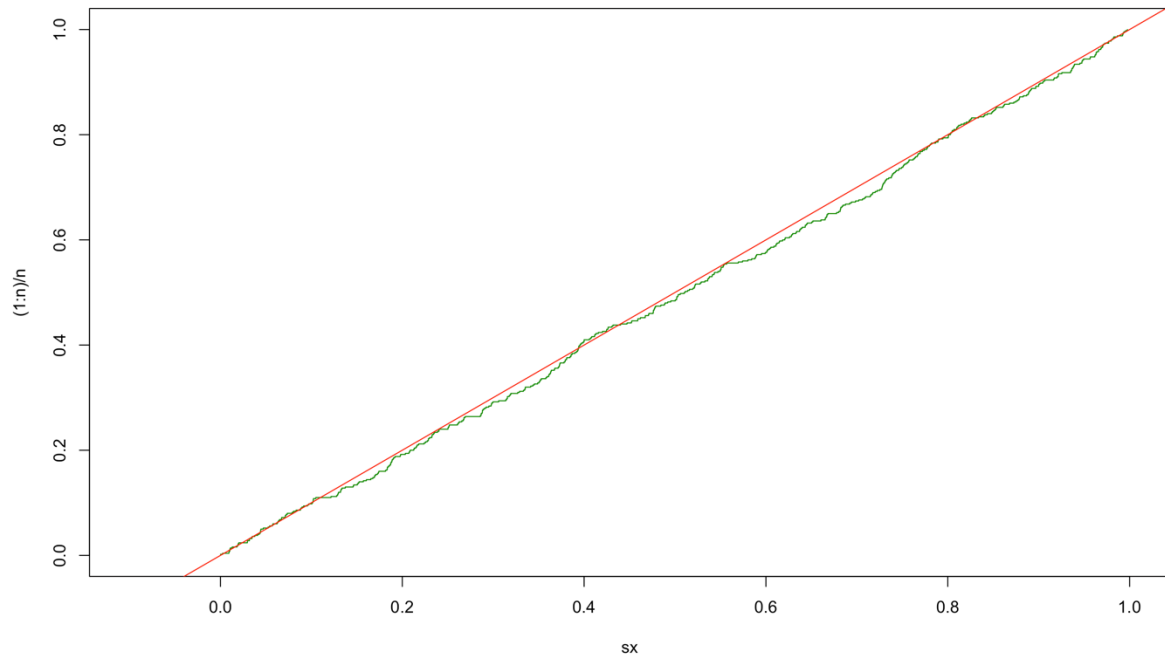
```
lincon_generator <- function(n) {  
  a <- 75831  
  c0 <- 0  
  m <- 2^31  
  seed <- 2^15  
  l <- numeric(n) # Цілочислова послідовність  
  l[1] <- seed  
  
  generate <- function() {  
    l <- (a * l + c0) %% m  
    return(l / m)}  
  
  return(replicate(n, generate()))  
}
```

2.

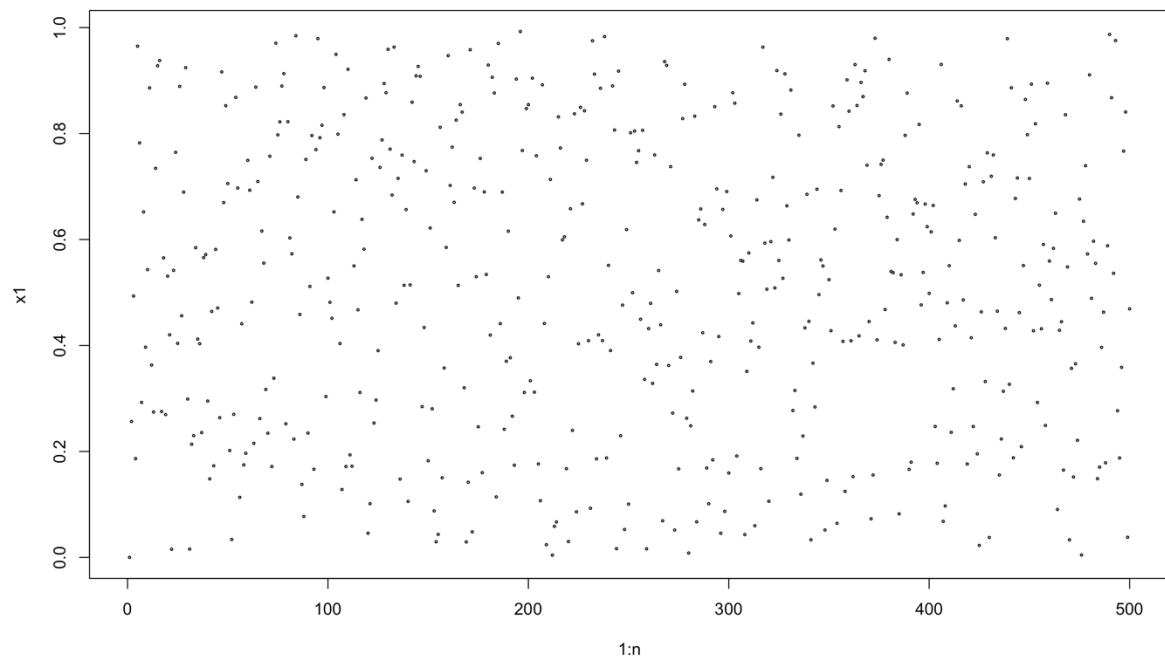
```
(a.1) sx <- sort(x1)  
# рисуємо емпіричну функцію розподілу:  
plot(sx, (1:n)/n, type="s", xlim=c(-0.1,1), ylim=c(0,1), col="green4")  
# графік теоретичної функції розподілу:  
abline(0,1,col="red")
```



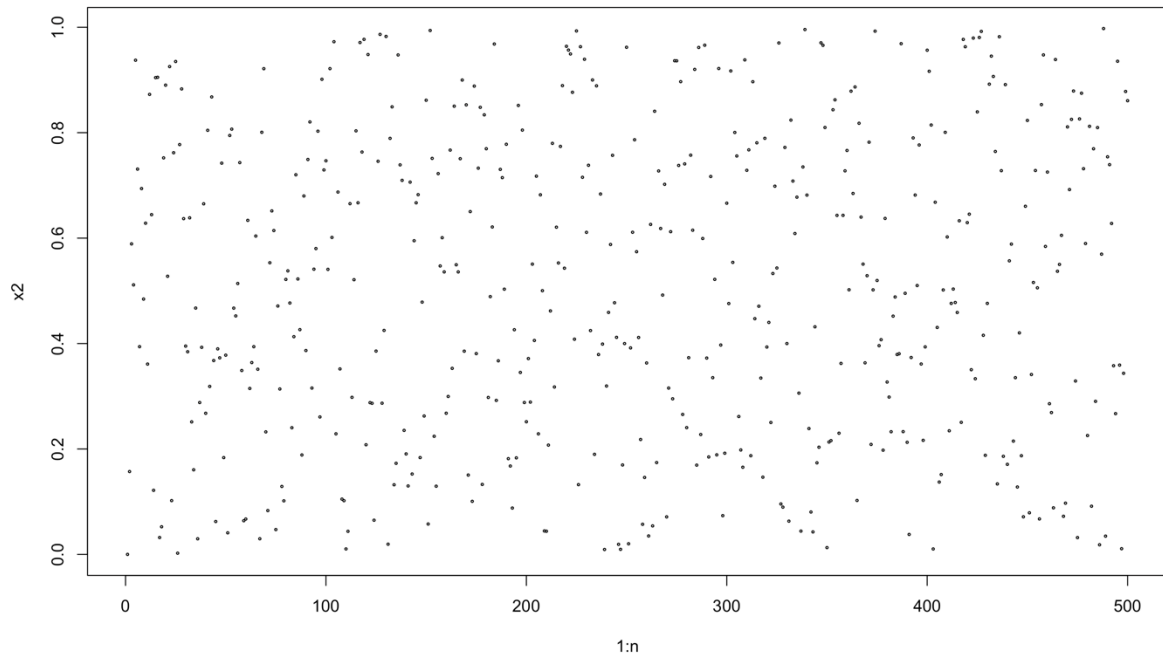
```
(a.2) # функція розподілу
sx<-sort(x2)
# рисуємо емпіричну функцію розподілу:
plot(sx,(1:n)/n, type="s", xlim=c(-0.1,1), ylim=c(0,1), col="green4")
# графік теоретичної функції розподілу:
abline(0,1,col="red")
```



```
(b.1) plot(1:n, x1, sech=0.3) # рисуємо діаграму чисел
```



(b.2) `plot(1:n, x2, cex=0.3)`



(c.1) `#перевіряємо незалежність d=2`

```
x11 <- x1[1:(n-1)]
```

```
x12 <- x1[2:n]
```

```
plot(x11,x12,cex=0.5)
```

```
#d=3
```

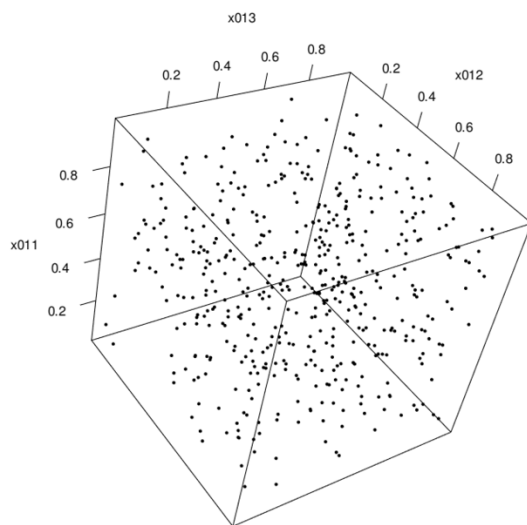
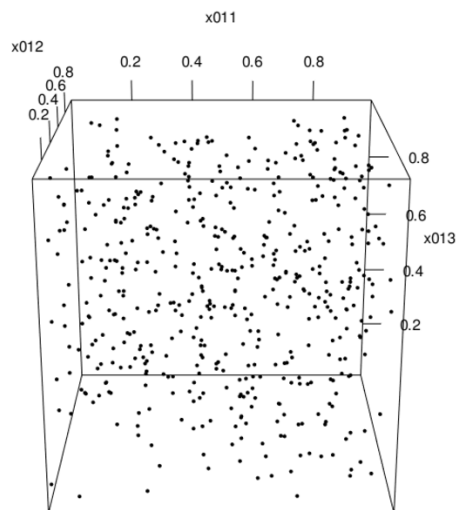
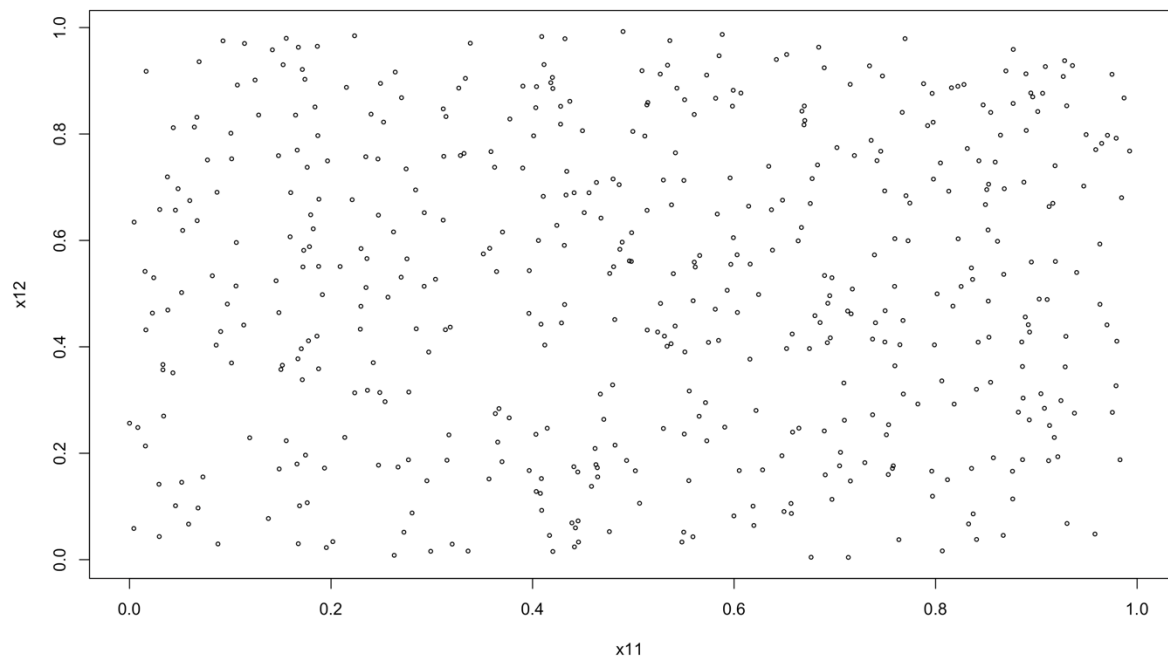
```
x011 <- x1[1:(n-2)]
```

```
x012 <- x1[2:(n-1)]
```

```
x013 <- x1[3:n]
```

```
library(rgl)
```

```
plot3d(x011,x012,x013)
```



(с.2) #перевіряємо незалежність d=2

```
x21 <- x2[1:(n-1)]
```

```
x22 <- x2[2:n]
```

```
plot(x21,x22,cex=0.5)
```

```
#d=3
```

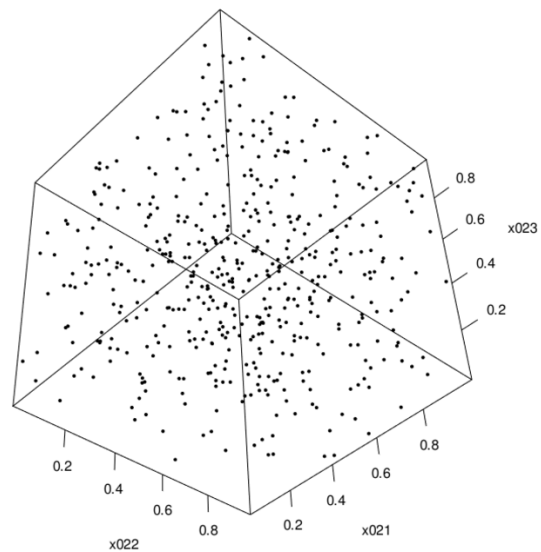
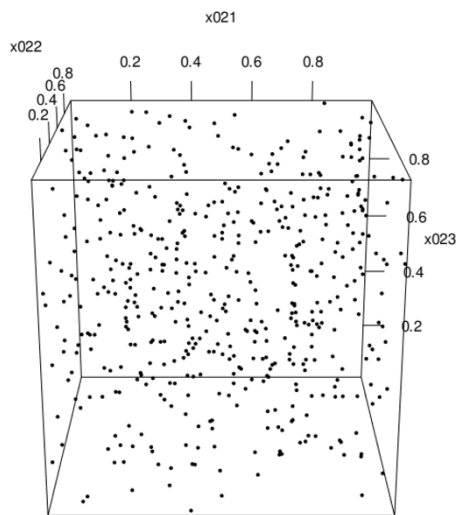
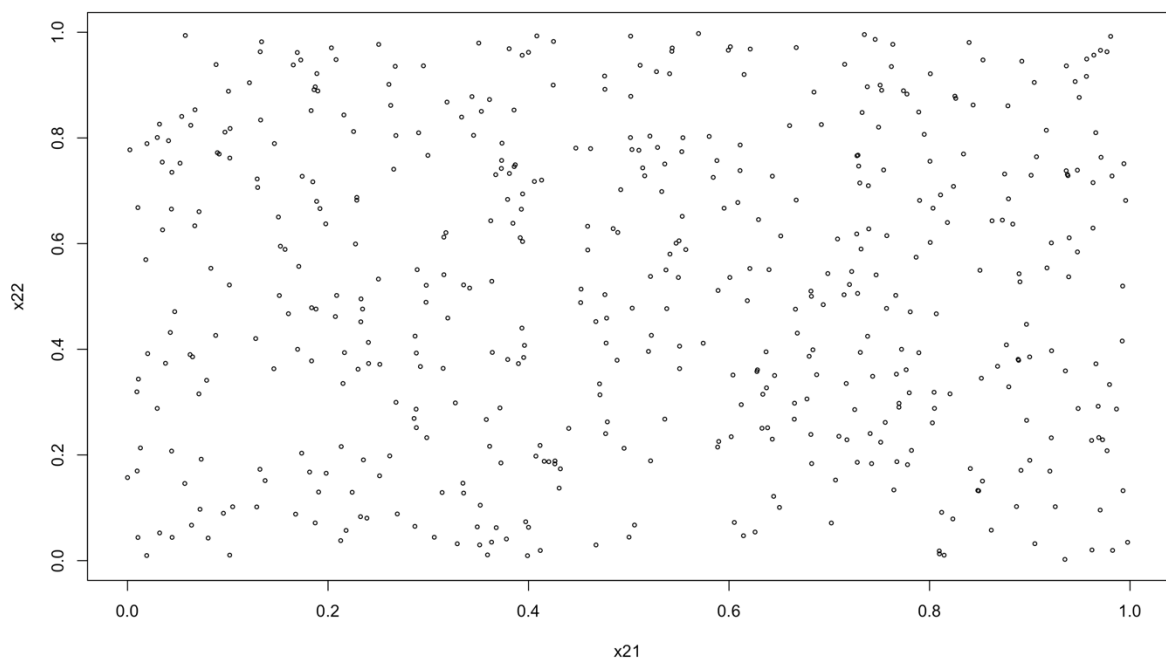
```
x021 <- x2[1:(n-2)]
```

```
x022 <- x2[2:(n-1)]
```

```
x023 <- x2[3:n]
```

```
library(rgl)
```

```
plot3d(x021,x022,x023)
```

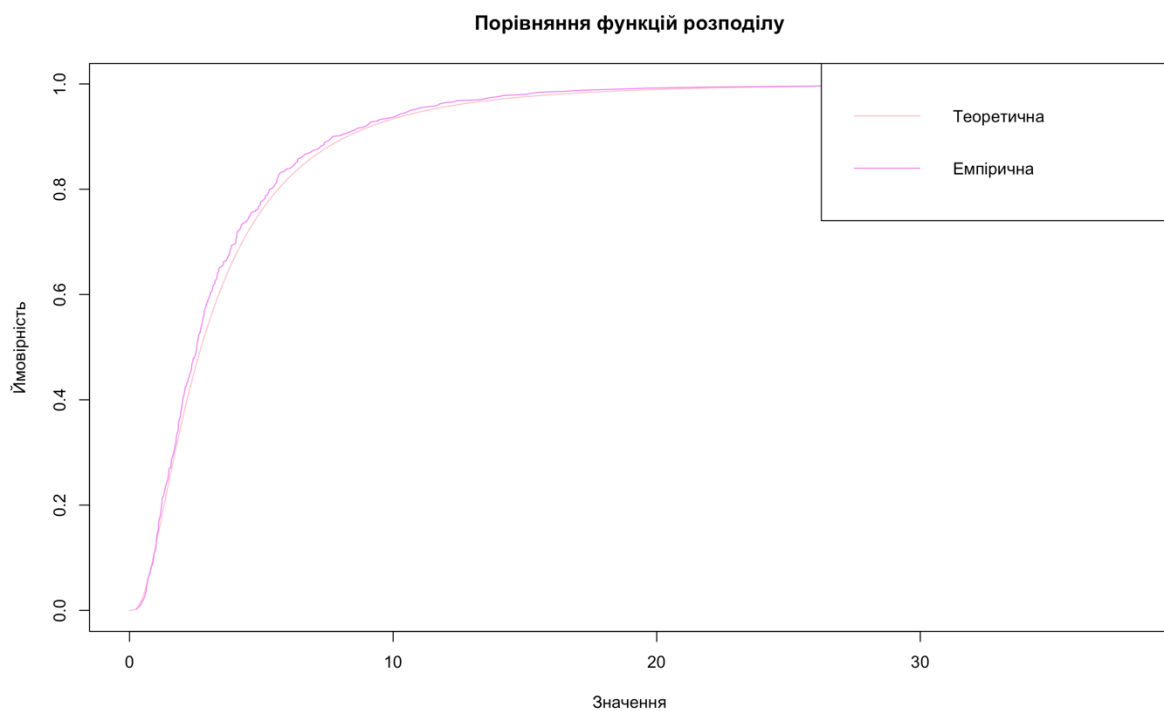


Бачимо, що і при використанні генератора Парка-Міллера, і при використанні генератора з параметрами з таблиці, дані дійсно є випадковими, тому що при перевірці двійок та трійок залежності не видно. Емпіричні та теоретичні функції розподілу збігаються в обох випадках. Отже, два генератори пройшли перевірку, та більш менш якісні.

3.

```
lognormal_generator <- function(n, meanlog, sdlog) {  
  rlnorm(n, meanlog = meanlog, sdlog = sdlog)  
}  
  
n <- 500  
meanlog <- 1  
sdlog <- sqrt(0.75)  
  
lognormal_seq_generator <- lincon_generator(n)  
lognormal_seq <- lognormal_generator(n, meanlog, sdlog)
```
4.

```
# Теоретична функція розподілу для логнормального розподілу  
theoretical_cdf <- plnorm(seq(0, max(lognormal_seq), length.out = n), meanlog =  
meanlog, sdlog = sdlog)  
  
# Емпірична функція розподілу для згенерованої послідовності  
empirical_cdf <- ecdf(lognormal_seq)  
  
# Порівняння функцій розподілу  
plot(seq(0, max(lognormal_seq), length.out = n), theoretical_cdf, type = "l", col =  
"pink", xlab = "Значення", ylab = "Ймовірність", main = "Порівняння функцій  
розподілу")  
lines(sort(lognormal_seq), empirical_cdf(sort(lognormal_seq)), col = "violet")  
legend("topright", legend = c("Теоретична", "Емпірична"), col = c("pink", "violet"),  
lty = 1)
```



```

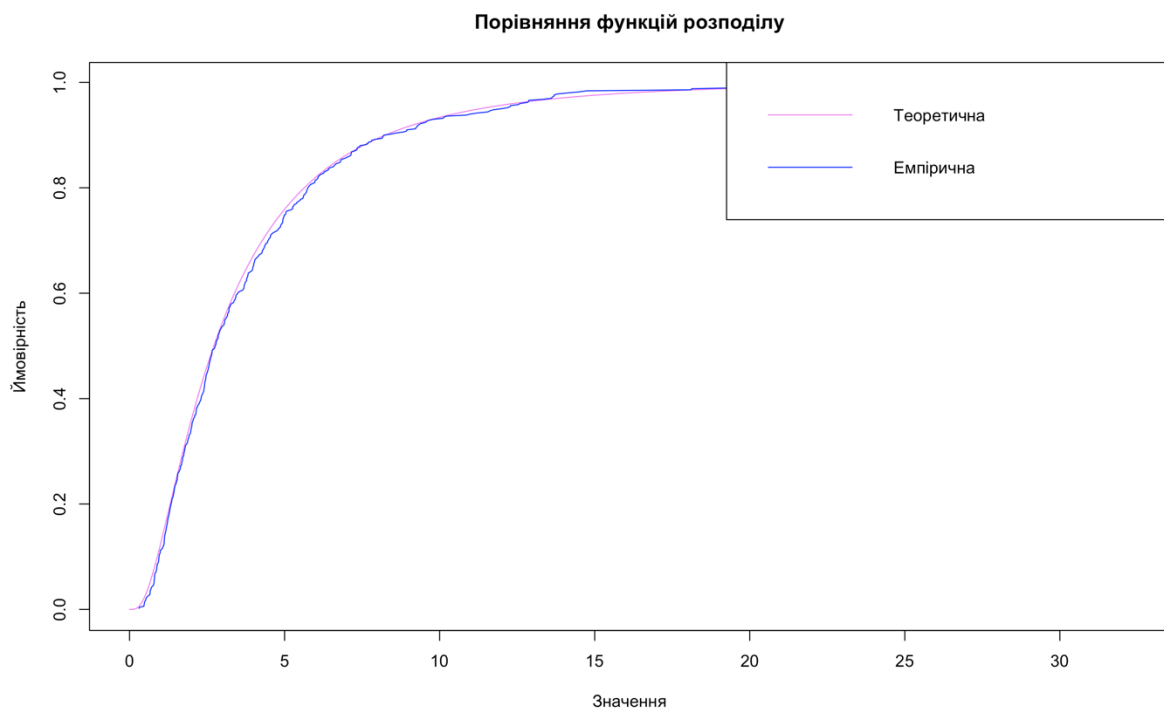
5. # Генерація стандартної послідовності з логнормальним розподілом
stand_lognormal_seq <- rlnorm(n, meanlog = meanlog, sdlog = sdlog)

# Теоретична функція розподілу для логнормального розподілу
theoretical0_cdf <- plnorm(seq(0, max(stand_lognormal_seq), length.out = n),
meanlog = meanlog, sdlog = sdlog)

# Емпірична функція розподілу для стандартної послідовності
empirical0_cdf <- ecdf(stand_lognormal_seq)

# Порівняння функцій розподілу
plot(seq(0, max(stand_lognormal_seq), length.out = n), theoretical0_cdf, type = "l",
col = "violet", xlab = "Значення", ylab = "Ймовірність", main = "Порівняння функцій
розподілу")
lines(sort(stand_lognormal_seq), empirical0_cdf(sort(stand_lognormal_seq)), col =
"blue")
legend("topright", legend = c("Теоретична", "Емпірична"), col = c("violet", "blue"),
lty = 1)

```



З рисунків порівняння функцій розподілу бачимо для генераторів з п.4 та п.5, що останній генератор має трішки кращу якість ніж в пункті 4. Але два генератори є якісними.