Analiza Szeregów Czasowych Raport 1

Paweł Matławski
album 249732

31 marca 2021

Spis treści

1 Symulacyjna analiza własności rozkładów asymptotycznych estymatorów średniej, autokowariancji i autokorelacji

W tej części raportu będziemy się zajmować badaniem własności asymptotycznych rozważanych estymatorów parametrów rozkładu szeregu stacjonarnego drugiego rzędu: estymator próbkowy wartości oczekiwanej, estymator funkcji kowariancji, estymator funkcji autokorelacji. Zobaczymy jak wpłynie na analizę długość szeregu oraz rozkład próby.

1.1 Dane generowane z rozkładu normalnego standardowego

W pierwszej symulacji realizacja szeregu biały szum pochodzi z rozkładu normalnego standardowego $N(0,1)$ i rozpatrzymy długości $n = 50, 100, 500, 2000$.

```
library(kable)

## Error in library(kable): there is no package called 'kable'

set.seed(420)
k <- 100 #realizacje
n <- c(50, 100, 500, 2000) #długość szeregu

realizacje <- vector(mode = "list", length = 0)
srednie <- vector(mode = "list", length = 0)

for (i in n){
  realizacje[match(i, n)] <- list(matrix(rnorm(i*k), i, k))
  srednie[match(i, n)] <- list(apply(realizacje[[match(i, n)]], MARGIN=2, FUN=mean))
}
```

1.1.1 Estymator wartości oczekiwanej

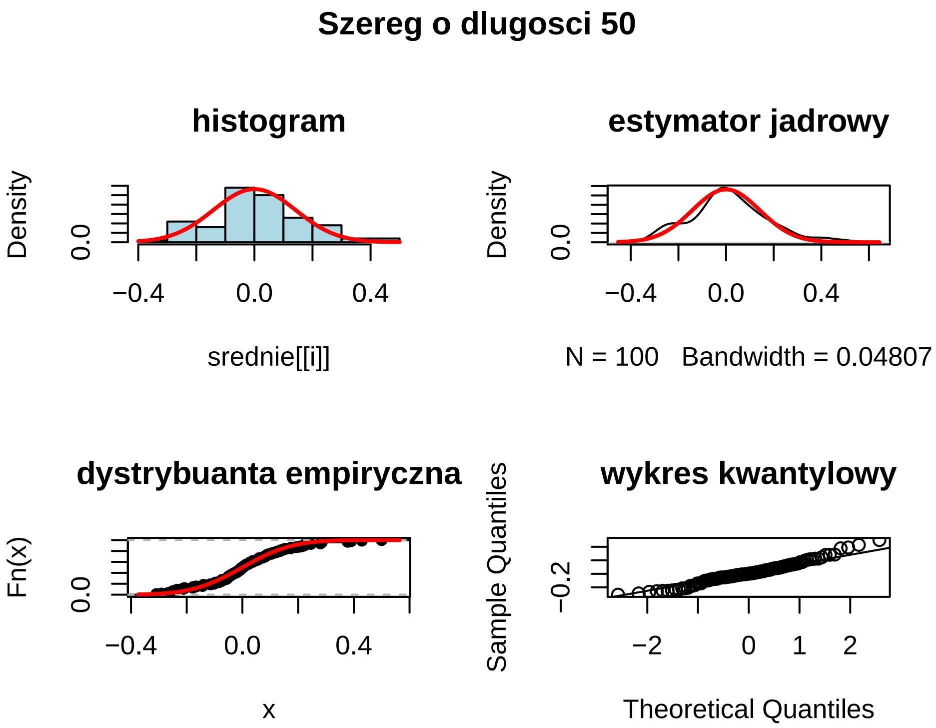
Estymator próbkowy wartości oczekiwanej μ - średnia próbkowa:

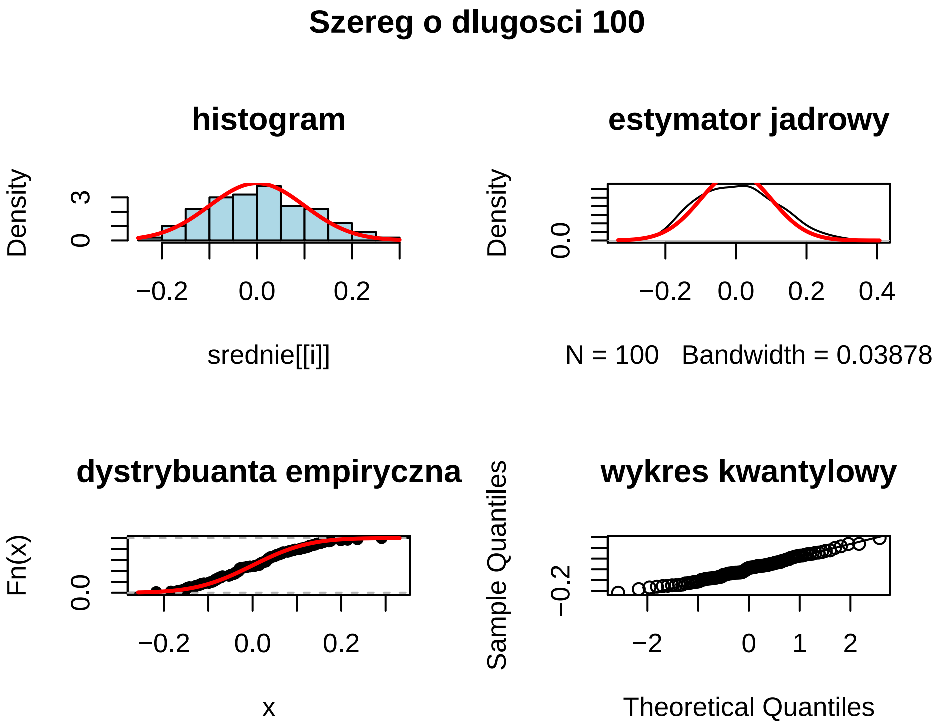
- $\bar{X}_n = \frac{1}{n}\sum_{t=1}^{n} X_t$

Sprawdzimy graficznie, jak zachowuję się średnia próbkowa korzystając z histogramu, estymatora jądrowego, dystrybuanty empirycznej oraz wykresu kwantylowego.

```
par(mfrow=c(2,2))
par(oma=c(0,0,2,0))
for (i in 1:length(n)){
  #par(mfrow=c(2,2))
  hist(srednie[[i]], probability=T, col="lightblue", main = 'histogram')
  curve(dnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  plot(density(srednie[[i]]), main = 'estymator jądrowy')
  curve(dnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)
```

```
  plot(ecdf(srednie[[i]]), main = 'dystrybuanta empiryczna')
  curve(pnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  qqnorm(srednie[[i]], main = 'wykres kwantylowy')
  qqline(srednie[[i]])

  title(paste0('Szereg o długości ',as.character(n[i])), outer = TRUE)
  par(oma=c(0,0,2,0))
}
```




Szereg o dlugosci 100
histogram
Density
0
3
−0.2
0.0
0.2
srednie[[i]]
estymator jadrowy
Density
0.0
−0.2
0.0
0.2
0.4
N = 100 Bandwidth = 0.03878
dystrybuanta empiryczna
Fn(x)
0.0
−0.2
0.0
0.2
x
wykres kwantylowy
Sample Quantiles
−0.2
−2
0
1
2
Theoretical Quantiles

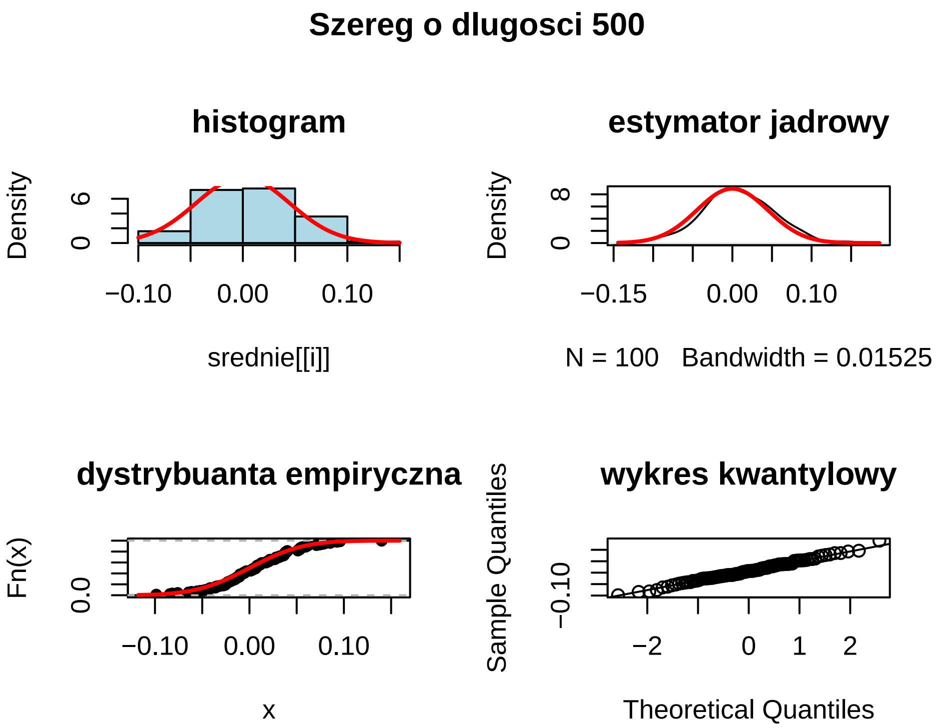
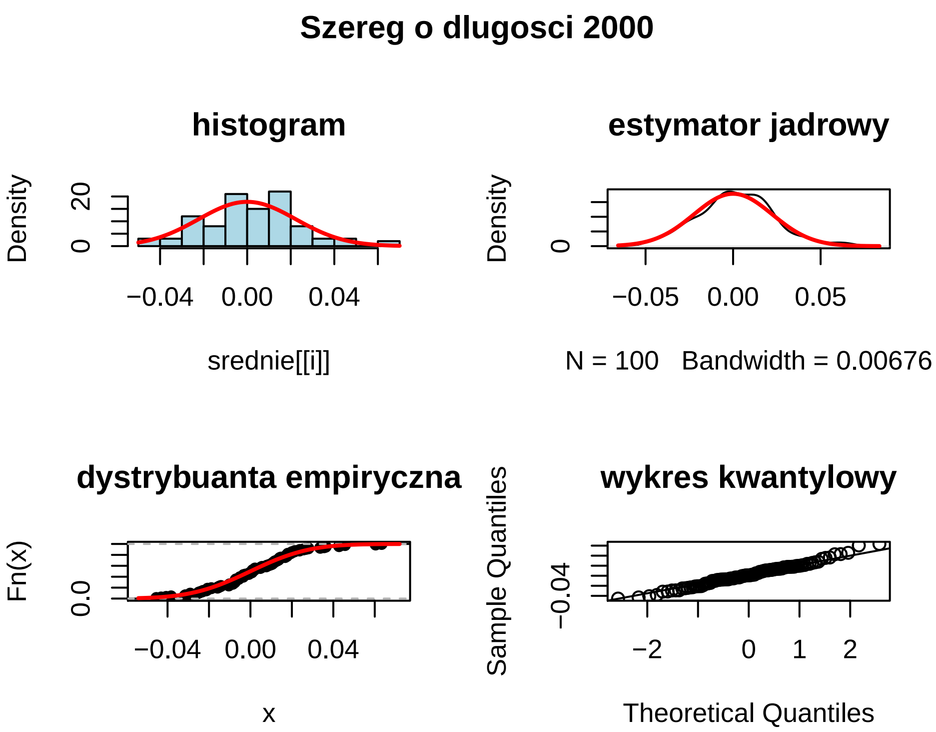

Szereg o dlugosci 500
histogram
Density
0
6
−0.10
0.00
0.10
srednie[[i]]
estymator jadrowy
Density
0
8
−0.15
0.00
0.10
N = 100 Bandwidth = 0.01525
dystrybuanta empiryczna
Fn(x)
0.0
−0.10
0.00
0.10
x
wykres kwantylowy
Sample Quantiles
−0.10
−2
0
1
2
Theoretical Quantiles

Tabela 1: P wartości dla szeregu długości 2000

	50	100	500	2000
Shapiro	0.125122357415997	0.384333385585413	0.96241332958939	0.498424237672453
KS	0.789591217150914	0.765696074095599	0.25232707697401	0.37954430410118



Teraz zastosujemy statystyczne testy zgodności: test Shapiro-Wilka oraz test Kołmogorowa-Smirnova.

```
Shapiro = list()
KStrue = list()
for (i in 1:length(n)){
  Shapiro[i] <- shapiro.test(srednie[[i]])[["p.value"]]
  KStrue[i] <- ks.test(srednie[[i]], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.value"]]
}
results <- matrix(data = c(Shapiro, KStrue), nrow = 2, byrow = TRUE, dimnames = list(c("

kable(results, format = "latex", digits = 2, caption = paste0("P wartości dla szeregu dł
```

Na podstawie wykresów możemy powiedzieć, że graficznie nasz rozkład przypomina bardzo rozkład normalny. Patrząc na wartości p w obydwu testach otrzymujemy potwierdzenie, że nie ma podstaw do odrzucenia hipotezy zerowej, która mówi że wartości pochodzą z rozkładu normalnego.

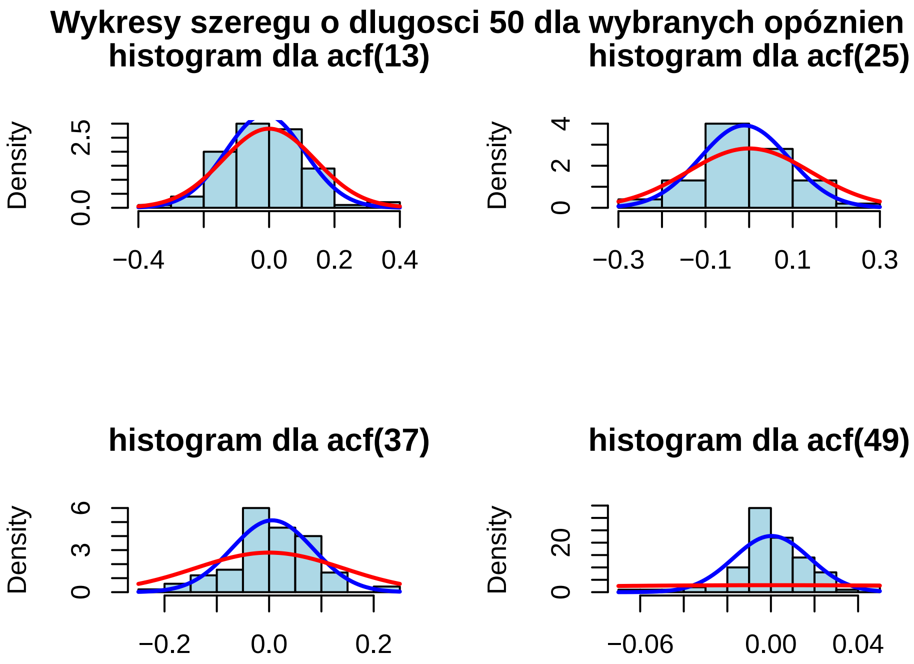
1.1.2 Estymator funkcji autokowariancji

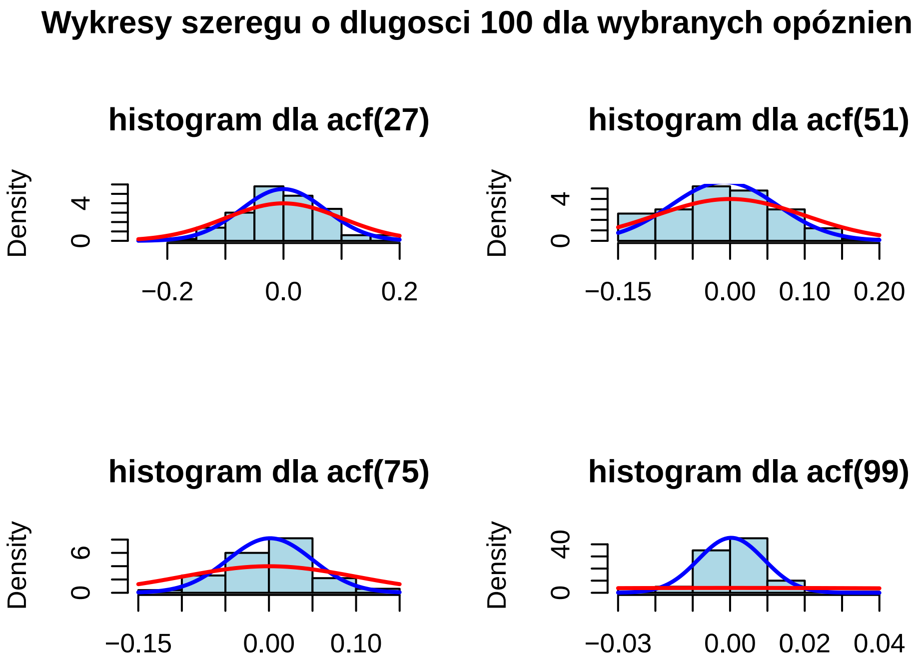
Estymator funkcji autokowariancji $\gamma(h)$:

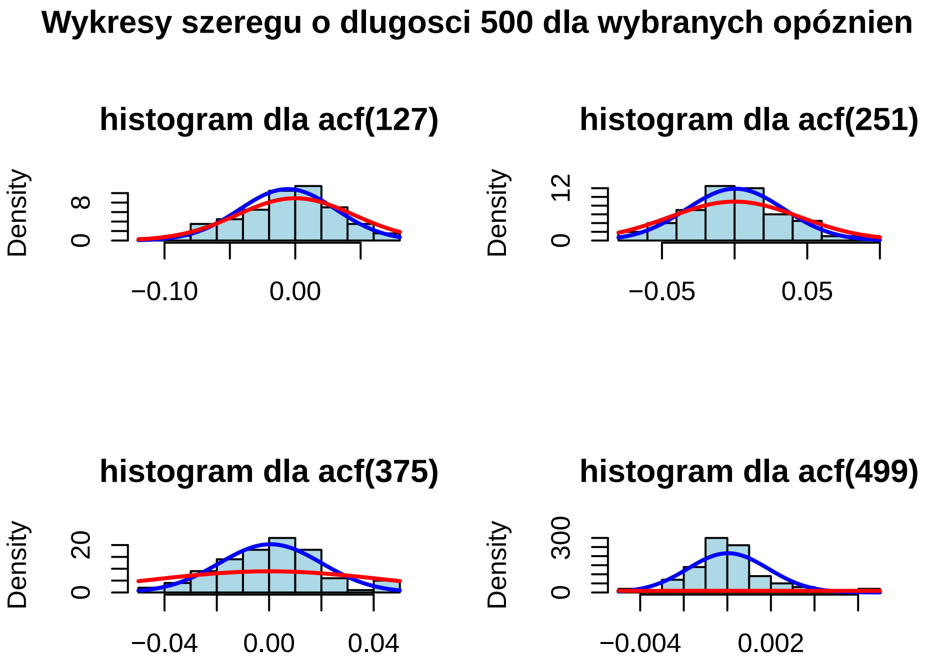
- $\hat{\gamma}(h) = \frac{1}{n} \sum_{t=1}^{n-h} (X_{t+h} - \bar{X}_n)(X_t - \bar{X}_n)$, dla $h = 0, 1, ..., n-1$

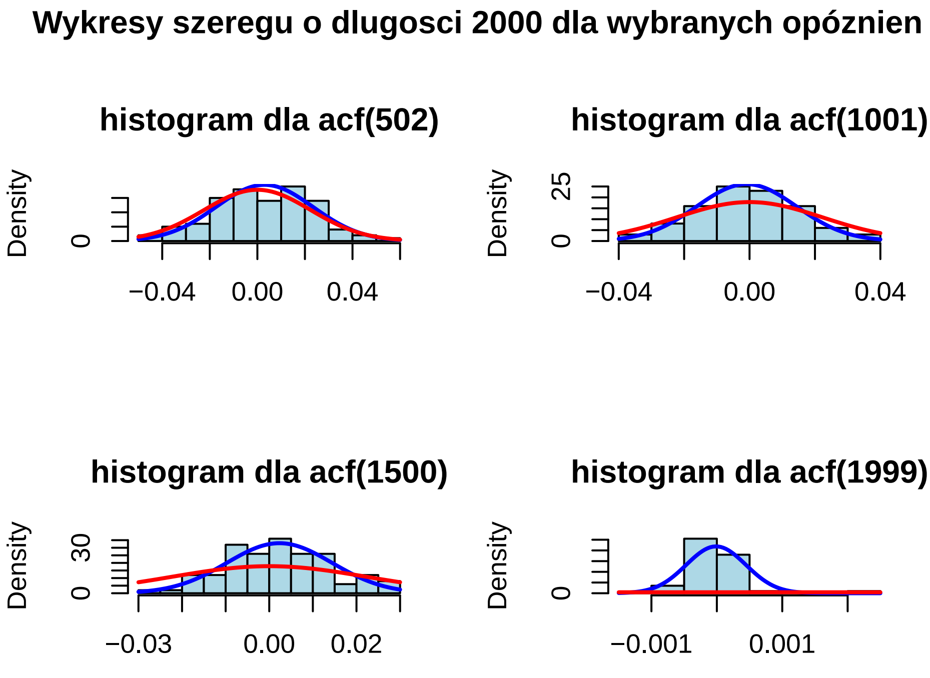
Przedstawmy jak wyglądają histogramy dla funkcji autokowariancji dla kolejnych opóźnień h.

```
acovf.matrix <- vector(mode = "list", length = 0)
h.wybrane <- list()
acovf.h <- list()

h.max <- floor(n-1)
for (i in 1:length(n)){
  acovf.matrix[[i]] <- apply(realizacje[[i]], 2, function(x) acf(x, lag.max=h.max[i], ty
  acovf.matrix[[i]] <- acovf.matrix[[i]][-1,]
  h.wybrane[[i]] <- seq(to = h.max[i], by = floor(h.max[i]/4), length.out = 4)
}


for (i in 1:length(n)){
  par(mfrow=c(2,2))
  for (h in h.wybrane[[i]])
  {
    tytul <- paste0("histogram dla acf(",h,")")
    acovf.h[i] <- list(acovf.matrix[[i]][h,])

    hist( acovf.h[[i]], freq=FALSE, col="lightblue", main=tytul, xlab="")
    curve(dnorm(x,mean=mean(acovf.h[[i]]), sd=sd(acovf.h[[i]])), add=T, col="blue", lwd=
    curve(dnorm(x,mean=0, sd=1/sqrt(n[i])), add=T, col="red", lwd=2)

  }
  par(oma=c(0,0,2,0))
  title(paste0('Wykresy szeregu o długości ',as.character(n[i]),' dla wybranych opóźnień
}
```


Wykresy szeregu o dlugosci 50 dla wybranych opóznien
histogram dla acf(13)
histogram dla acf(25)
histogram dla acf(37)
histogram dla acf(49)
Density
Density
Density
Density


Wykresy szeregu o dlugosci 100 dla wybranych opóznien
histogram dla acf(27)
histogram dla acf(51)
histogram dla acf(75)
histogram dla acf(99)
Density
Density
Density
Density





Obserwując wykresy dla różnych długości n, możemy zauważyć w każdym wypadku wraz ze wzrostem opóźnienia h, estymator radzi sobie coraz słabiej i coraz mniej dokładnie. Zweryfikujmy nasze obserwacje testami poprzednio wspomnianymi.

Tabela 2: P wartości dla szeregu długości 50

	13	25	37	49
Shapiro	0.626353745224954	0.560516779105099	0.236546214169498	0.00038485008705482
KS	0.312611751197762	0.0462173758915828	0.000608648137185308	1.10689235555128e-13

Tabela 3: P wartości dla szeregu długości 100

	27	51	75	99
Shapiro	0.676004502142776	0.343684427542049	0.543677351330346	2.61274139067642e-05
KS	0.110190130420863	0.0690678659322889	0.000314041568723278	3.33066907387547e-15

```
Shapiro2 = list()
KS2 = list()
num = 1
for (i in 1:length(n)){
  num = 1
  for (h in h.wybrane[[i]]){
    Shapiro2[num] <- shapiro.test(acovf.matrix[[i]][h,])[["p.value"]]
    KS2[num] <- ks.test(acovf.matrix[[i]][h,], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.val
    num = num+1
  }
  #results2 <- matrix(data = c(Shapiro2, KS2), nrow = 2, byrow = TRUE, dimnames = list
  results2 <- matrix(data = c(Shapiro2, KS2), nrow = 2, byrow = TRUE, dimnames = list(c(
  print(kable(results2, format = "latex", digits = 2, caption = paste0("P wartości dla s
}
```

Na podstawie informacji zawartych w tabelce możemy stwierdzić, że rzeczywiście wzraz ze wzrostem opóźnienia h, będziemy częściej skłonni do odrzucenia hipotezy zerowej, która mówi nam, że dane pochodzą z rozkładu normalnego. Wartości p zazwyczaj są coraz mniejsze.

1.1.3 Estymator funkcji autokorelacji

Estymator funkcji autokorelacji $\rho(h)$:

- $\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\sigma}^2} = \frac{\hat{\gamma}^2(h)}{\gamma(\hat{0})}$ dla $h = 0, 1, ..., n-1$.

Przedstawmy jak wyglądają histogramy funkcji autokorelacji dla kolejnych opóźnień h.

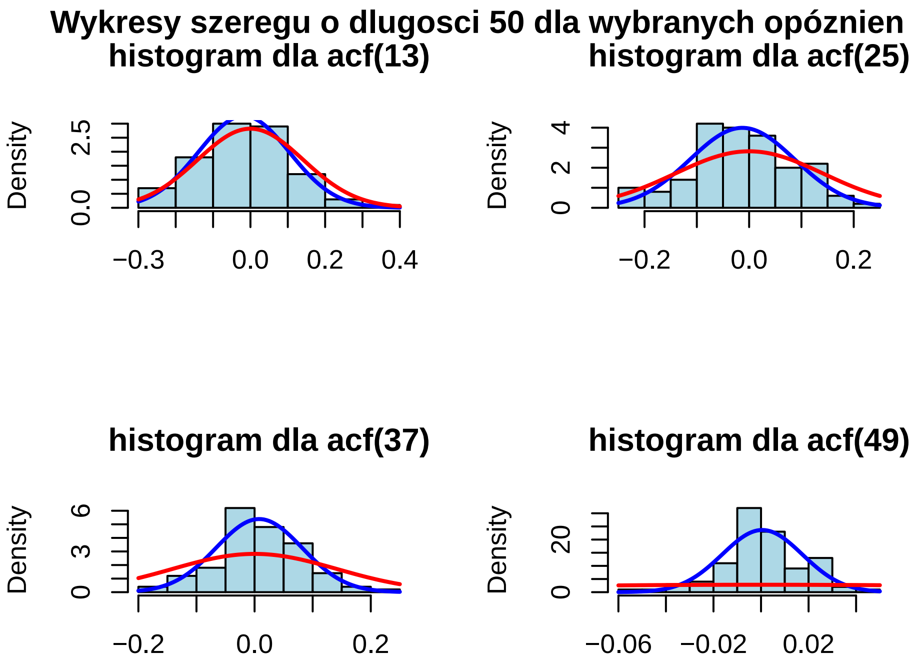
Tabela 4: P wartości dla szeregu długości 500

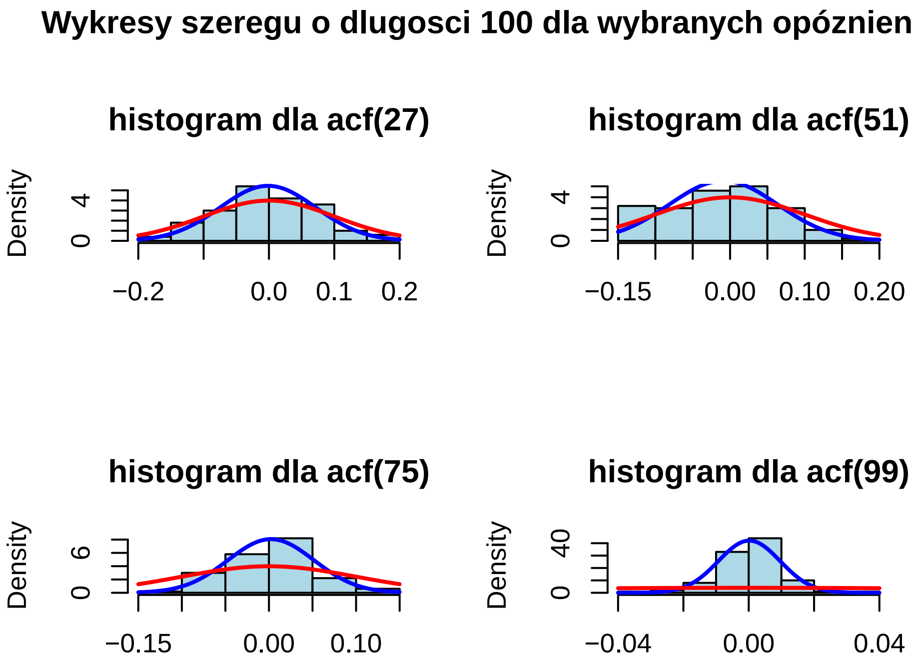
	127	251	375	499
Shapiro	0.601722685348344	0.806976262915576	0.572905525756989	8.09704651238261e-05
KS	0.100537060479501	0.321511845529206	0.000134024302870461	0

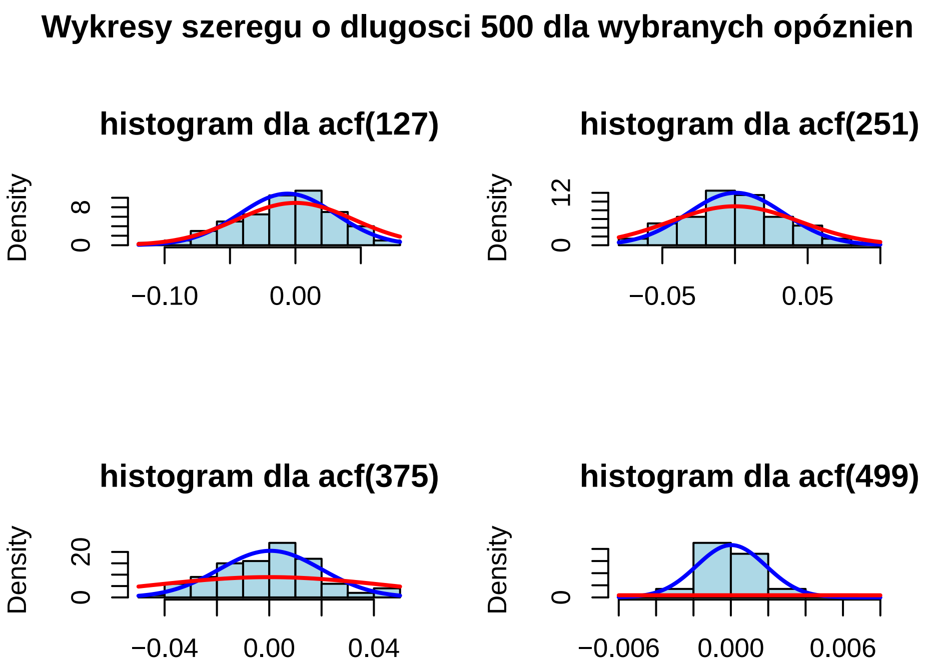
Tabela 5: P wartości dla szeregu długości 2000

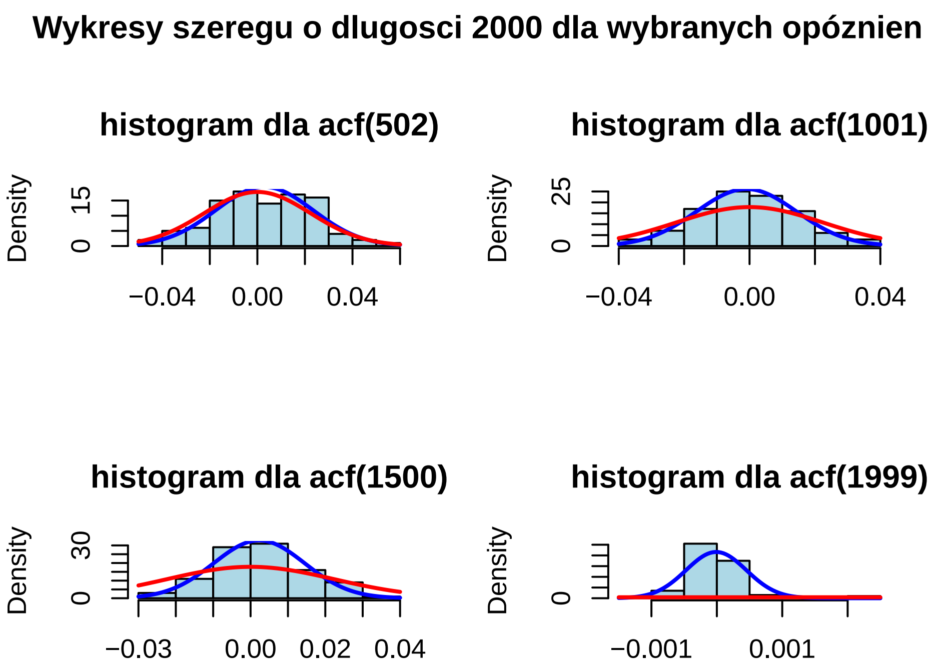
	502	1001	1500	1999
Shapiro	0.856535192539726	0.956729439561629	0.548719554691103	1.15035969851717e-09
KS	0.306795045885751	0.140501764176944	0.000549614655227182	0

```
acf.matrix <- vector(mode = "list", length = 0)
acf.h <- list()

for (i in 1:length(n)){
  acf.matrix[[i]] <- apply(realizacje[[i]], 2, function(x) acf(x, lag.max=h.max[i], type
  acf.matrix[[i]] <- acf.matrix[[i]][-1,]
  h.wybrane[[i]] <- seq(to = h.max[i], by = floor(h.max[i]/4), length.out = 4)
}


for (i in 1:length(n)){
  par(mfrow=c(2,2))
  for (h in h.wybrane[[i]])
  {
    tytul <- paste0("histogram dla acf(",h,")")
    acf.h[i] <- list(acf.matrix[[i]][h,])

    hist( acf.h[[i]], freq=FALSE, col="lightblue", main=tytul, xlab="")
    curve(dnorm(x,mean=mean(acf.h[[i]]), sd=sd(acf.h[[i]])), add=T, col="blue", lwd=2)
    curve(dnorm(x,mean=0, sd=1/sqrt(n[i])), add=T, col="red", lwd=2)

  }
  par(oma=c(0,0,2,0))
  title(paste0('Wykresy szeregu o długości ',as.character(n[i]),' dla wybranych opóźnień
}
```


Wykresy szeregu o dlugosci 50 dla wybranych opóznien
histogram dla acf(13)
histogram dla acf(25)
histogram dla acf(37)
histogram dla acf(49)
Density
0.0
2.5
−0.3
0.0
0.2
0.4
0
2
4
−0.2
0.0
0.2
0
3
6
−0.2
0.0
0.2
0
20
−0.06
−0.02
0.02


Wykresy szeregu o dlugosci 100 dla wybranych opóznien
histogram dla acf(27)
histogram dla acf(51)
histogram dla acf(75)
histogram dla acf(99)
Density
0
4
−0.2
0.0
0.1
0.2
0
4
−0.15
0.00
0.10
0.20
0
6
−0.15
0.00
0.10
0
40
−0.04
0.00
0.04





Podobnie jak dla autokowariancji, widzimy że wraz ze wzrostem opóźnienia estymatora mają coraz gorsze własności i odnoszą gorsze rezulatty.

```
Shapiro3 = list()
KS3 = list()
for (i in 1:length(n)){
  num = 1
  for (h in h.wybrane[[i]]){
```

Tabela 6: P wartości dla szeregu długości 50

	13	25	37	49
Shapiro	0.966945151931991	0.652097100253576	0.798249231908545	0.0209694584697625
KS	0.374636064563366	0.0797232033662924	0.00108708402335966	1.23456800338317e-13

Tabela 7: P wartości dla szeregu długości 100

	27	51	75	99
Shapiro	0.978605206236239	0.191330064619093	0.866715982259453	3.66631054050543e-05
KS	0.259024926874748	0.0556109133908442	0.00248158857377279	9.88098491916389e-15

```
    Shapiro3[num] <- shapiro.test(acf.matrix[[i]][h,])[["p.value"]]
    KS3[num] <- ks.test(acf.matrix[[i]][h,], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.value
    num = num+1
  }
  results3 <- matrix(data = c(Shapiro3, KS3), nrow = 2, byrow = TRUE, dimnames = list(c(
  print(kable(results3, format = "latex", digits = 2, caption = paste0("P wartości dla s
}
```

1.2 Dane generowane z rozkładu wykładniczego

Powtórzymy analizę, tym razem generując dane z rozkładu wykładniczego o parametrze $\lambda = 1$.

1.2.1 Estymator wartości oczekiwanej

```
set.seed(420)
realizacje2 <- vector(mode = "list", length = 0)
srednie2 <- vector(mode = "list", length = 0)

for (i in n){
  realizacje2[match(i, n)] <- list(matrix(rexp(i*k), i, k))
  srednie2[match(i, n)] <- list(apply(realizacje2[[match(i, n)]], MARGIN=2, FUN=mean))
}
```

```
par(mfrow=c(2,2))
par(oma=c(0,0,2,0))
```

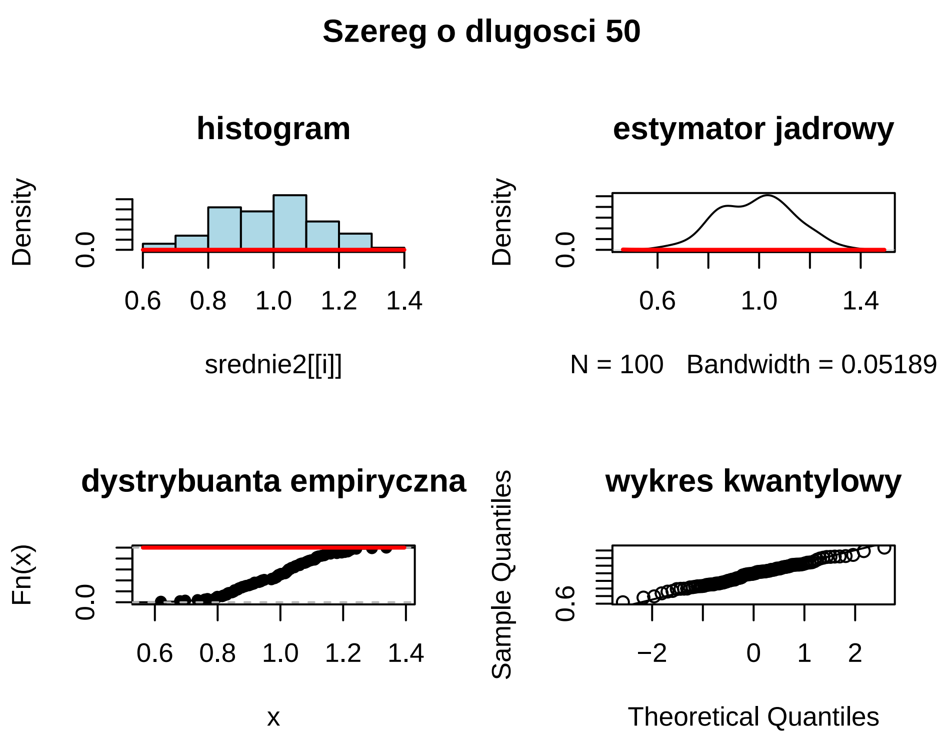
Tabela 8: P wartości dla szeregu długości 500

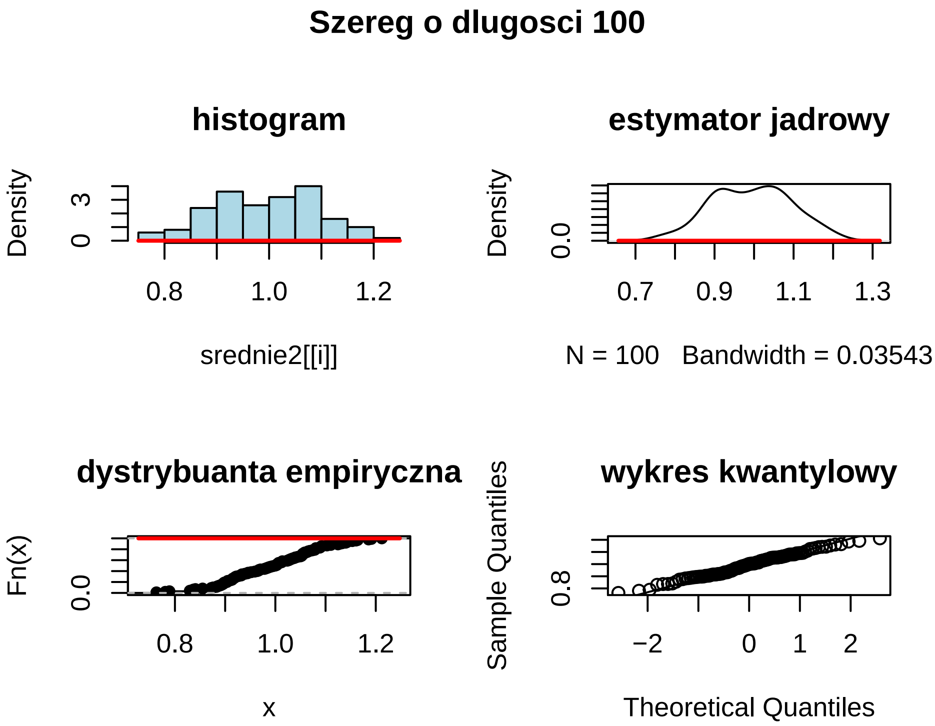
	127	251	375	499
Shapiro	0.501237934195918	0.958441855238468	0.455845082411033	8.88540987718826e-05
KS	0.0488425174593545	0.340436160811764	0.000184390361938402	0

Tabela 9: P wartości dla szeregu długości 2000

	502	1001	1500	1999
Shapiro	0.851606607335579	0.907518459715699	0.796856470757551	5.82448230216116e-10
KS	0.288534469453902	0.143133563339019	0.000761056799644	0

```r
for (i in 1:length(n)){
  hist(srednie2[[i]], probability=T, col="lightblue", main = 'histogram')
  curve(dnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  plot(density(srednie2[[i]]), main = 'estymator jądrowy')
  curve(dnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  plot(ecdf(srednie2[[i]]), main = 'dystrybuanta empiryczna')
  curve(pnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)
  qqnorm(srednie2[[i]], main = 'wykres kwantylowy')
  qqline(srednie2[[i]])

  title(paste0('Szereg o długości ',as.character(n[i])), outer = TRUE)
  par(oma=c(0,0,2,0))
}
```




Szereg o dlugosci 100
histogram
Density
0
3
0.8
1.0
1.2
srednie2[[i]]
estymator jadrowy
Density
0.0
0.7
0.9
1.1
1.3
N = 100 Bandwidth = 0.03543
dystrybuanta empiryczna
Fn(x)
0.0
0.8
1.0
1.2
x
wykres kwantylowy
Sample Quantiles
0.8
−2
0
1
2
Theoretical Quantiles

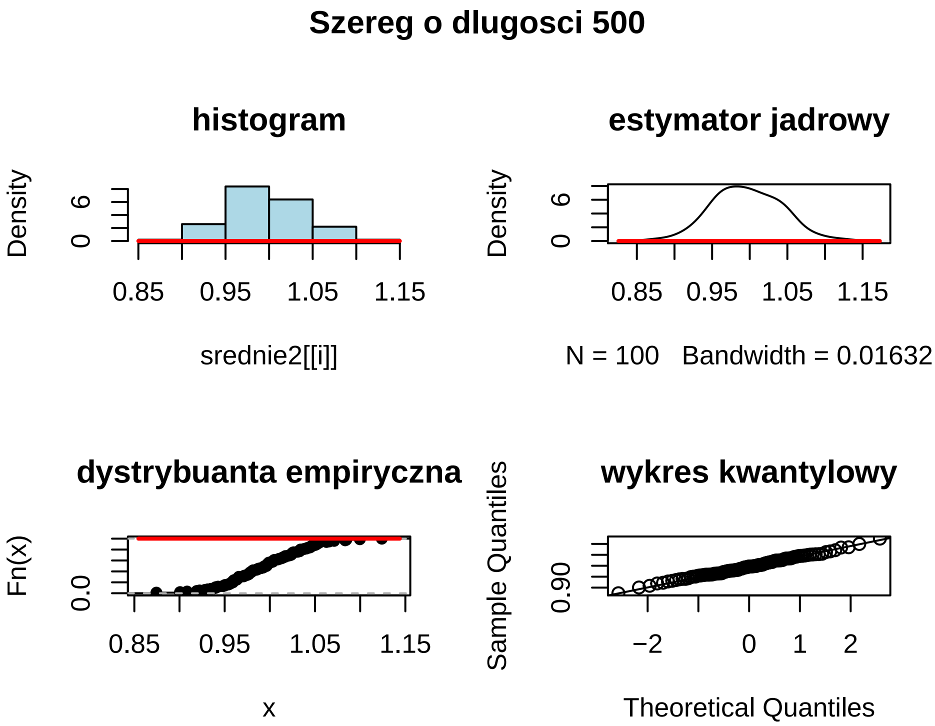
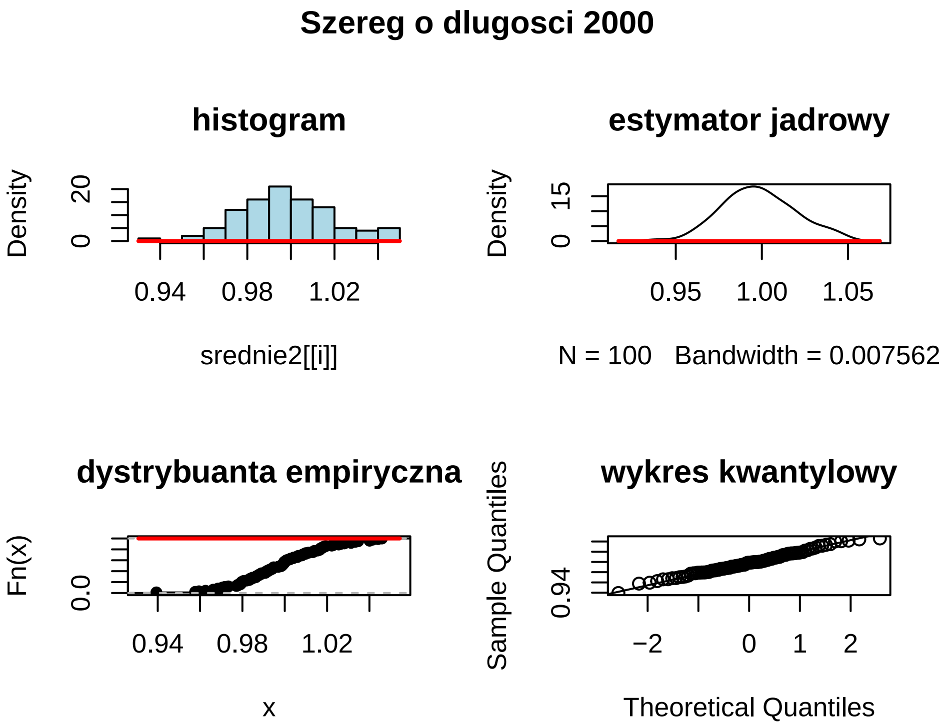

Szereg o dlugosci 500
histogram
Density
0
6
0.85
0.95
1.05
1.15
srednie2[[i]]
estymator jadrowy
Density
0
6
0.85
0.95
1.05
1.15
N = 100 Bandwidth = 0.01632
dystrybuanta empiryczna
Fn(x)
0.0
0.85
0.95
1.05
1.15
x
wykres kwantylowy
Sample Quantiles
0.90
−2
0
1
2
Theoretical Quantiles

Tabela 10: P wartości dla szeregu długości 2000

	50	100	500	2000
Shapiro	0.832413562177183	0.508560277430281	0.994530111829788	0.768786079973345
KS	0	0	0	0



Pomimo, że wykresy wyglądają jakby były generowane na podstawie danych z rozkładu normalnego, to widzimy że nie jest to rozkład $N(0, \frac{1}{\sqrt{n}})$.

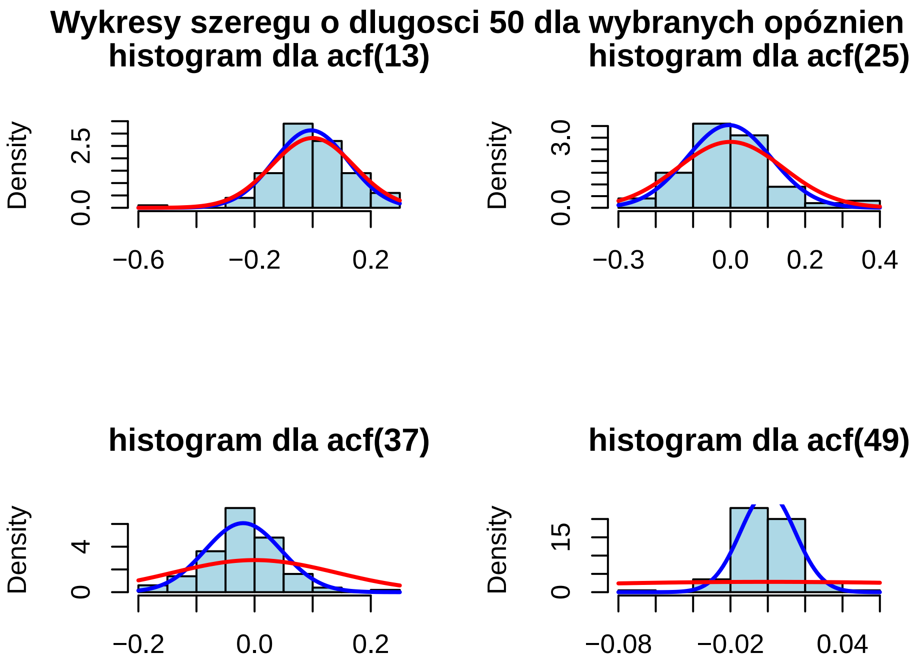
```
Shapiro21 = list()
KS21 = list()
for (i in 1:length(n)){
  Shapiro21[i] <- shapiro.test(srednie2[[i]])[["p.value"]]
  KS21[i] <- ks.test(srednie2[[i]], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.value"]]
}
results21 <- matrix(data = c(Shapiro21, KS21), nrow = 2, byrow = TRUE, dimnames = list(c

kable(results21, format = "latex", digits = 2, caption = paste0("P wartości dla szeregu
```

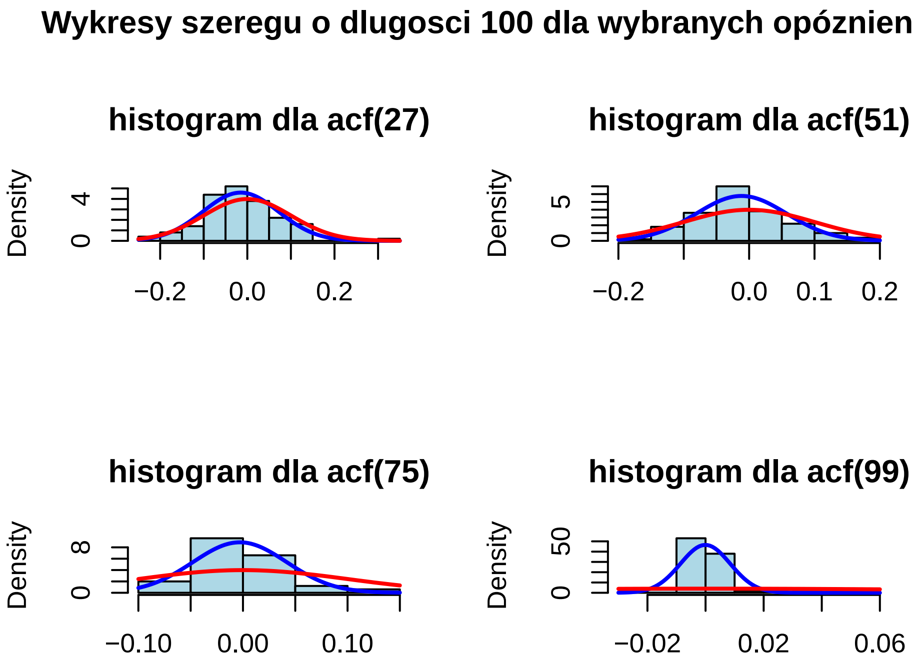
W wynikach mamy rozbieżność z podziałem na typ testu. Test Shapiro-Wilka sugeruje nam, że nie ma podstaw do odrzucenia hipotezy, że rozkład jest normalny, natomiast test Kołmogorowa-Smirnova jednoznacznie (dla każdego n) przyjmuje wartość p zbliżoną do 0.

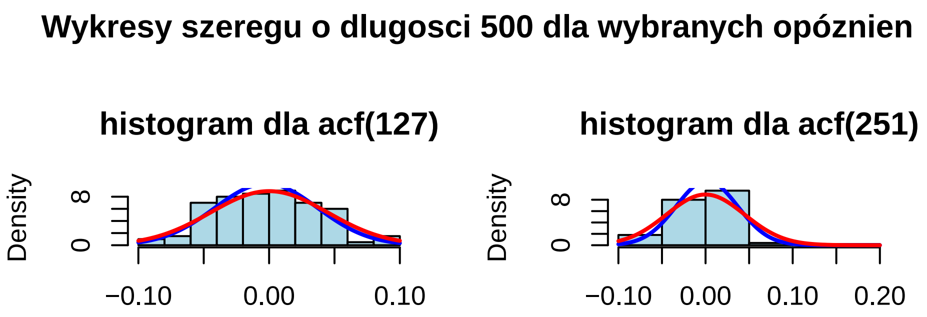
1.2.2 Estymator funkcji autokowariancji

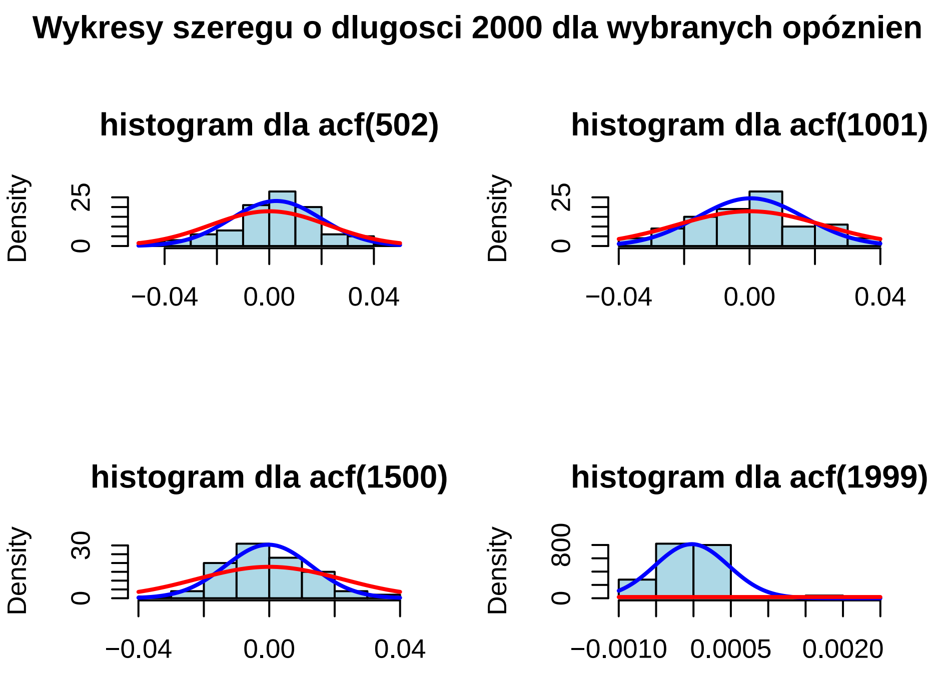
Popatrzmy teraz jak graficznie przedstawia się estymator autokowariancji.

```
acovf.matrix2 <- vector(mode = "list", length = 0)
h.wybrane2 <- list()
acovf.h2 <- list()

h.max2 <- floor(n-1)

for (i in 1:length(n)){
  acovf.matrix2[[i]] <- apply(realizacje2[[i]], 2, function(x) acf(x, lag.max=h.max2[i],
  acovf.matrix2[[i]] <- acovf.matrix2[[i]][-1,]
  h.wybrane2[[i]] <- seq(to = h.max2[i], by = floor(h.max2[i]/4), length.out = 4)
}


for (i in 1:length(n)){
  par(mfrow=c(2,2))
  for (h in h.wybrane2[[i]])
  {
    tytul <- paste0("histogram dla acf(",h,")")
    acovf.h2[i] <- list(acovf.matrix2[[i]][h,])

    hist( acovf.h2[[i]], freq=FALSE, col="lightblue", main=tytul, xlab="")
    curve(dnorm(x,mean=mean(acovf.h2[[i]]), sd=sd(acovf.h2[[i]])), add=T, col="blue", lw
    curve(dnorm(x,mean=0, sd=1/sqrt(n[i])), add=T, col="red", lwd=2)

  }
  par(oma=c(0,0,2,0))
  title(paste0('Wykresy szeregu o długości ',as.character(n[i]),' dla wybranych opóźnień
}
```


Wykresy szeregu o dlugosci 50 dla wybranych opóznien
histogram dla acf(13)
histogram dla acf(25)
histogram dla acf(37)
histogram dla acf(49)
Density
0.0
2.5
3.0
0
4
15
−0.6
−0.2
0.2
−0.3
0.0
0.4
−0.08
−0.02
0.04


Wykresy szeregu o dlugosci 100 dla wybranych opóznien
histogram dla acf(27)
histogram dla acf(51)
histogram dla acf(75)
histogram dla acf(99)
Density
0
4
5
8
50
−0.2
0.0
0.2
0.1
−0.10
0.00
0.10
−0.02
0.02
0.06






```
Shapiro22 = list()
KS22 = list()
for (i in 1:length(n)){
  num = 1
  for (h in h.wybrane2[[i]]){
    Shapiro22[num] <- shapiro.test(acovf.matrix2[[i]][h,])[["p.value"]]
    KS22[num] <- ks.test(acovf.matrix2[[i]][h,], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.v
```

Tabela 11: P wartości dla szeregu długości 50

	13	25	37	49
Shapiro	0.00682153454884103	0.0158625842840976	0.249193338725892	3.03109689995131e-06
KS	0.307499795278359	0.0594402810185622	3.35164156384238e-06	1.92068583260152e-14

Tabela 12: P wartości dla szeregu długości 100

	27	51	75	99
Shapiro	0.0557769962710683	0.335314404699235	0.00357602051299236	1.61897358935028e-13
KS	0.076130495965417	0.0066830454140312	4.79627435622243e-05	2.22044604925031e-16

```
    num = num+1
  }
  results22 <- matrix(data = c(Shapiro22, KS22), nrow = 2, byrow = TRUE, dimnames = list
  print(kable(results22, format = "latex", digits = 2, caption = paste0("P wartości dla
}
```

Pomimo, że wykresy dla wartości opóźnień do h/4, wydają się pokrywać z wykresami dla rozkładu normalnego, to dla większych opóźnień mają o wiele słabsze własności, a wyniki obu testów są zbyt zróżnicowane, aby jednoznacznie ustosunkować się do hipotezy. Jednak dalej zauważamy, że im wraz ze wzrostem opóźnienia testy skłaniają się bardziej ku odrzuceniu hipotezy, że dane pochodzą z rozkładu normalnego.

1.2.3 Estymator funkcji autokorelacji

```
acf.matrix2 <- vector(mode = "list", length = 0)
acf.h2 <- list()

for (i in 1:length(n)){
  acf.matrix2[[i]] <- apply(realizacje2[[i]], 2, function(x) acf(x, lag.max=h.max2[i], t
  acf.matrix2[[i]] <- acf.matrix2[[i]][-1,]
  h.wybrane2[[i]] <- seq(to = h.max2[i], by = floor(h.max2[i]/4), length.out = 4)
}


for (i in 1:length(n)){
  par(mfrow=c(2,2))
  for (h in h.wybrane2[[i]])
  {
```

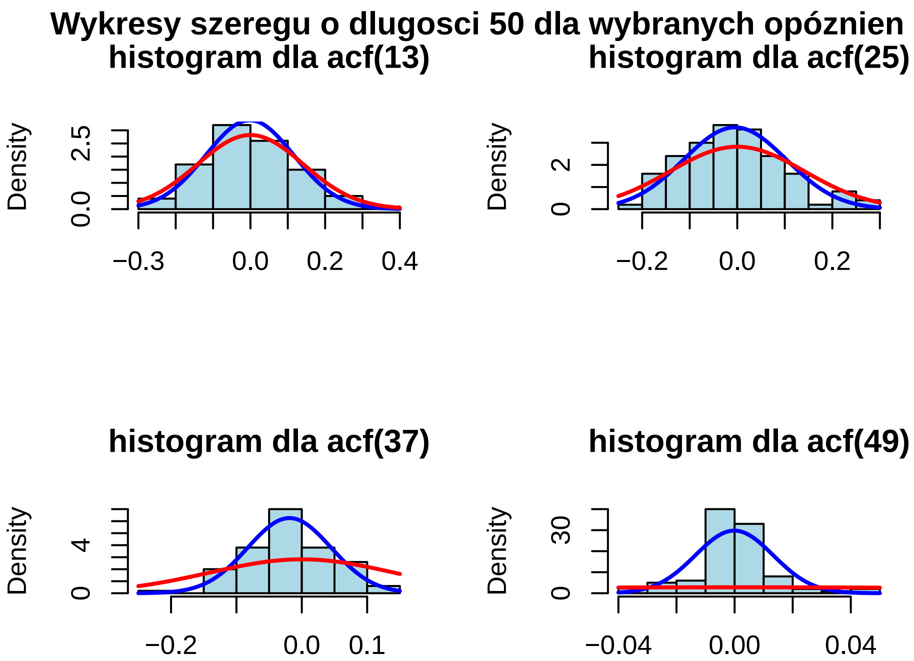
Tabela 13: P wartości dla szeregu długości 500

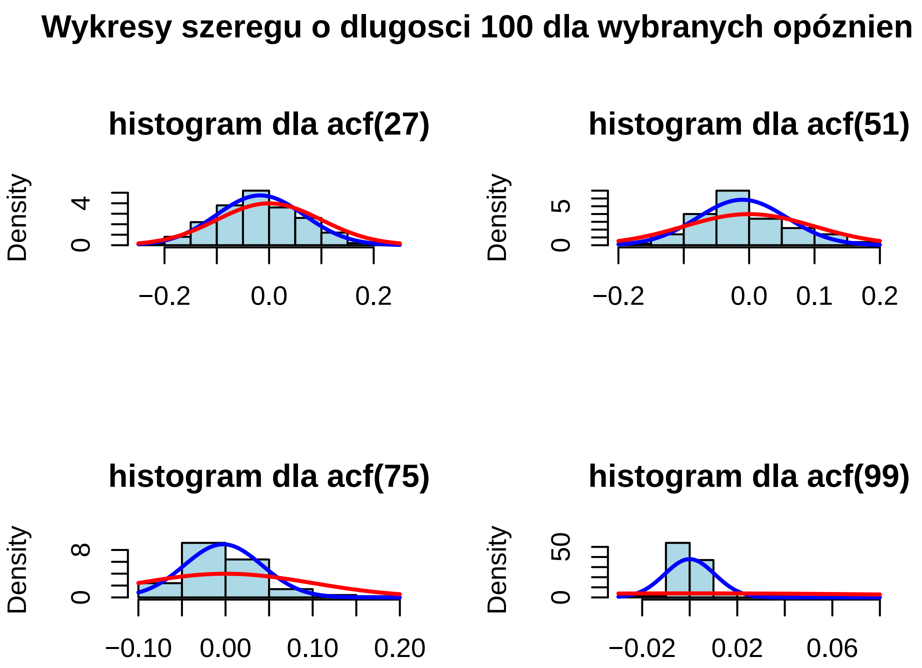
	127	251	375	499
Shapiro	0.777235011873936	0.00176231311744374	0.296022596342716	4.39444185590903e-11
KS	0.647747620002406	0.0475291585723651	3.01672428841826e-05	0

Tabela 14: P wartości dla szeregu długości 2000

	502	1001	1500	1999
Shapiro	0.0485669410503266	0.426599585487006	0.243180415257666	3.69446265025117e-08
KS	0.00247629770700963	0.392850349276899	0.005034464674321	0

```
    tytul <- paste0("histogram dla acf(",h,")")
    acf.h2[i] <- list(acf.matrix2[[i]][h,])

    hist( acf.h2[[i]], freq=FALSE, col="lightblue", main=tytul, xlab="")
    curve(dnorm(x,mean=mean(acf.h2[[i]]), sd=sd(acf.h2[[i]])), add=T, col="blue", lwd=2)
    curve(dnorm(x,mean=0, sd=1/sqrt(n[i])), add=T, col="red", lwd=2)
  }
  par(oma=c(0,0,2,0))
  title(paste0('Wykresy szeregu o długości ',as.character(n[i]),' dla wybranych opóźnień
}
```




Wykresy szeregu o dlugosci 100 dla wybranych opóznien
histogram dla acf(27)
histogram dla acf(51)
histogram dla acf(75)
histogram dla acf(99)
Density
−0.2 0.0 0.2
−0.2 0.0 0.1 0.2
−0.10 0.00 0.10 0.20
−0.02 0.02 0.06

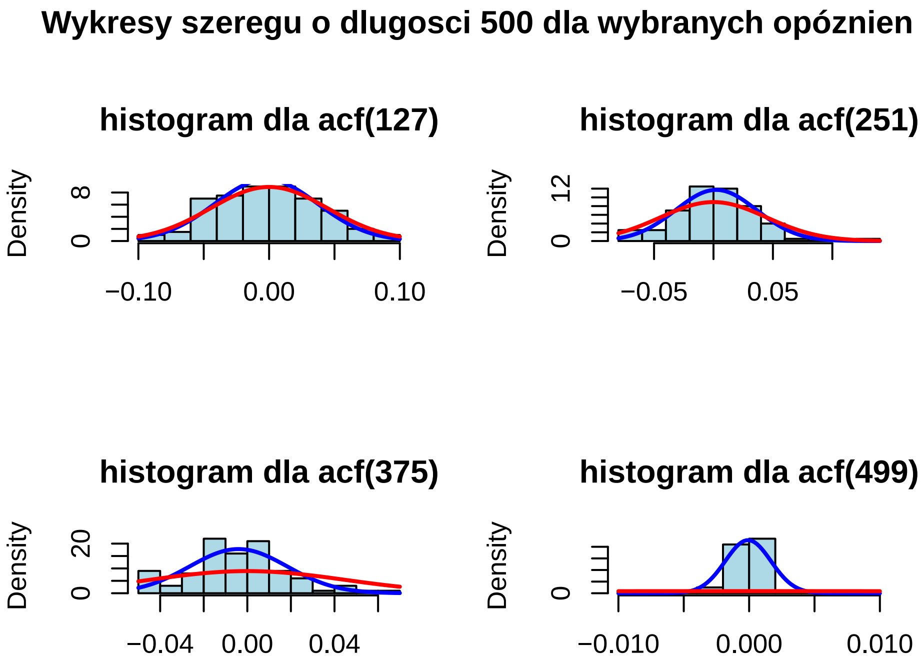
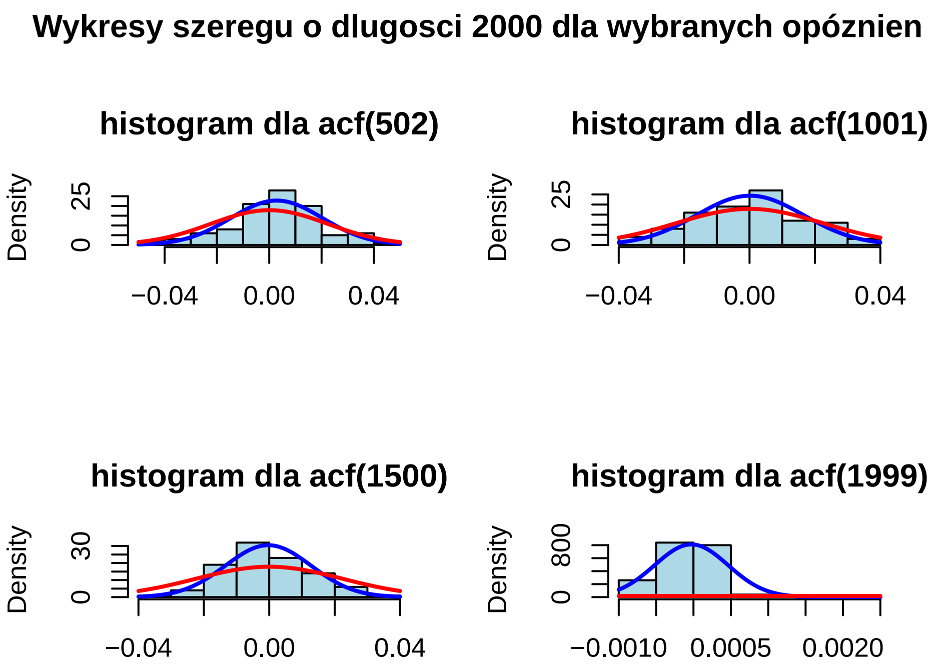

Wykresy szeregu o dlugosci 500 dla wybranych opóznien
histogram dla acf(127)
histogram dla acf(251)
histogram dla acf(375)
histogram dla acf(499)
Density
−0.10 0.00 0.10
−0.05 0.05
−0.04 0.00 0.04
−0.010 0.000 0.010

Tabela 15: P wartości dla szeregu długości 50

	13	25	37	49
Shapiro	0.207978469733893	0.119036090328456	0.787918461216173	4.060580767128e-05
KS	0.252491704225867	0.180988536282511	9.65261208405455e-05	1.49880108324396e-14

Tabela 16: P wartości dla szeregu długości 100

	27	51	75	99
Shapiro	0.912894075178584	0.0837335932603532	0.00183469622245987	9.15277622450034e-15
KS	0.133471482292119	0.0188495378615308	3.88314271732026e-05	3.33066907387547e-16




```
Shapiro23 = list()
KS23 = list()
for (i in 1:length(n)){
  num = 1
  for (h in h.wybrane2[[i]]){
    Shapiro23[num] <- shapiro.test(acf.matrix2[[i]][h,])[["p.value"]]
    KS23[num] <- ks.test(acf.matrix2[[i]][h,], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.val
    num = num+1
  }
  results23 <- matrix(data = c(Shapiro23, KS23), nrow = 2, byrow = TRUE, dimnames = list
  print(kable(results23, format = "latex", digits = 2, caption = paste0("P wartości dla
}
```

Wnioski są bardzo podobne jak w przypadku funkcji autokowariancji.

Tabela 17: P wartości dla szeregu długości 500

	127	251	375	499
Shapiro	0.984573430135666	0.0170243208166904	0.0833432004366341	1.40422047816094e-10
KS	0.78459121933196	0.0427997199866685	0.000147612387038976	0

Tabela 18: P wartości dla szeregu długości 2000

	502	1001	1500	1999
Shapiro	0.0542592497374025	0.465495245631203	0.309727510133097	5.02309795324463e-08
KS	0.00211534565468574	0.409147176929637	0.00314150190828411	0

1.3 Dane generowane z rozkładu normalnego $N(0, 5)$

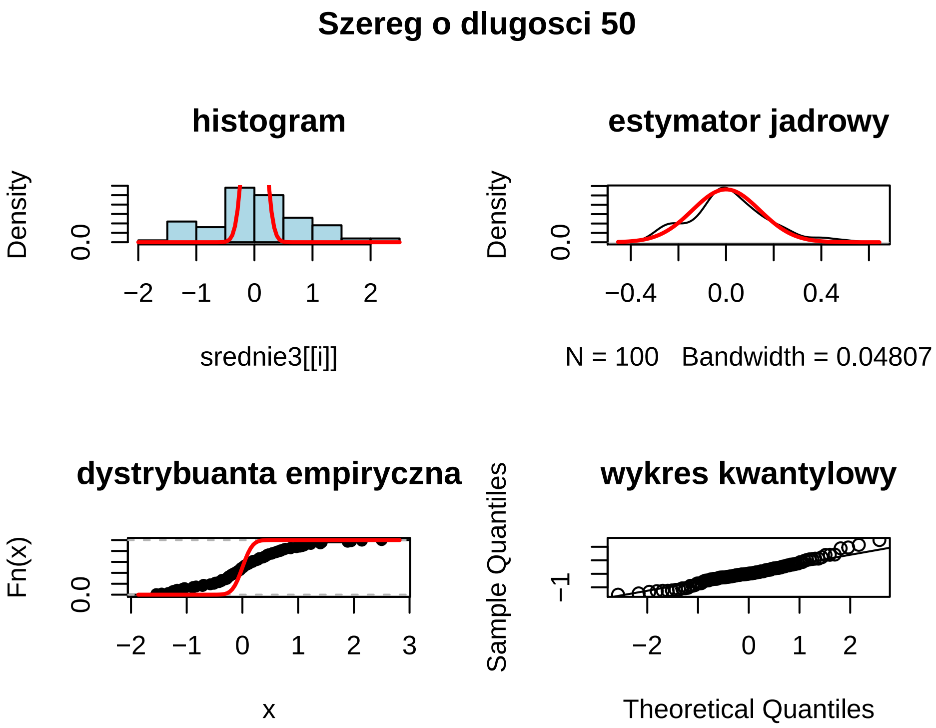
W trzeciej symulacji realizacja szeregu biały szum pochodzi z rozkładu normalnego $N(0, 5)$ i rozpatrzymy długości $n = 50, 100, 500, 2000$. Powtórzymy dokładnie to samo co w pierwszej symulacji i sprawdzimy jaki wpływ na wyniki ma zwiększona wariancja.

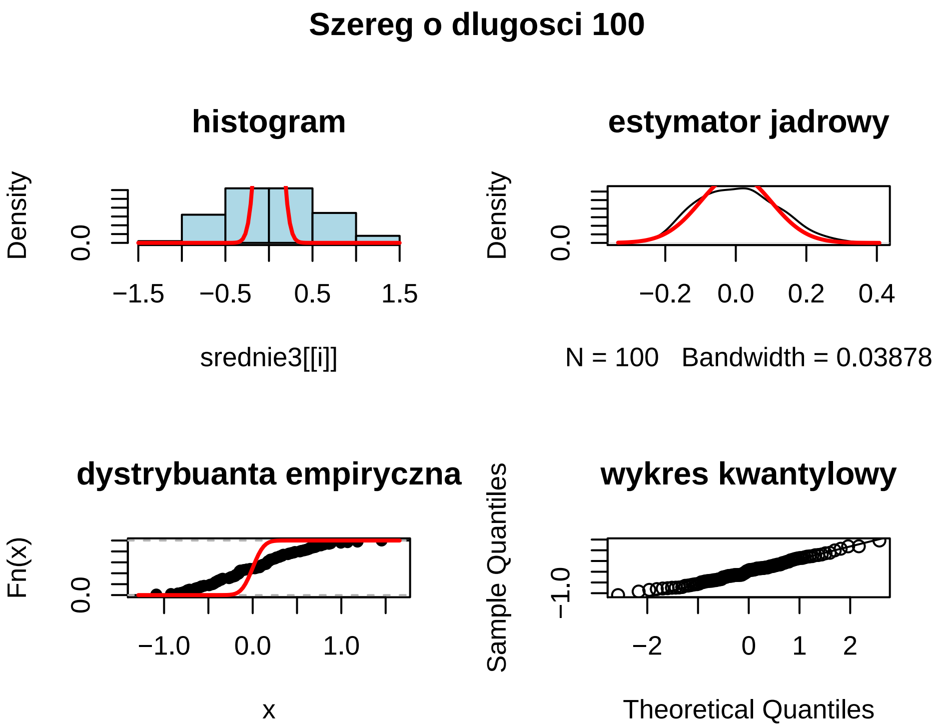
```
library(xtable)
set.seed(420)
k <- 100 #realizacje
n <- c(50, 100, 500, 2000) #długość szeregu

realizacje3 <- vector(mode = "list", length = 0)
srednie3 <- vector(mode = "list", length = 0)

for (i in n){
  realizacje3[match(i, n)] <- list(matrix(rnorm(i*k, mean = 0, sd = 5), i, k))
  srednie3[match(i, n)] <- list(apply(realizacje3[[match(i, n)]], MARGIN=2, FUN=mean))
}
```

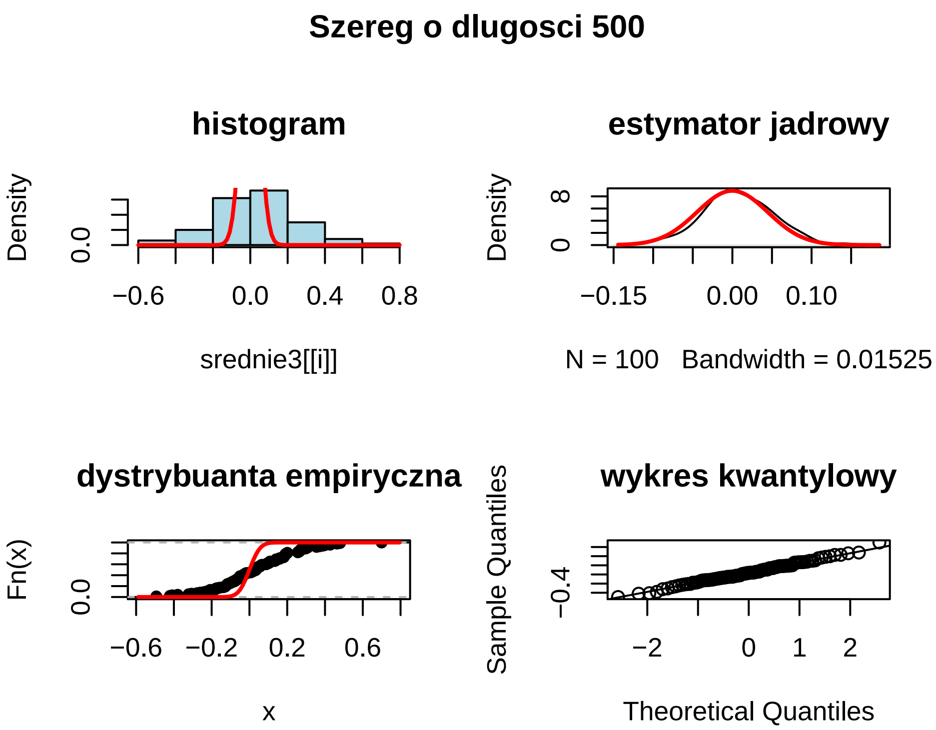
1.3.1 Estymator wartości oczekiwanej

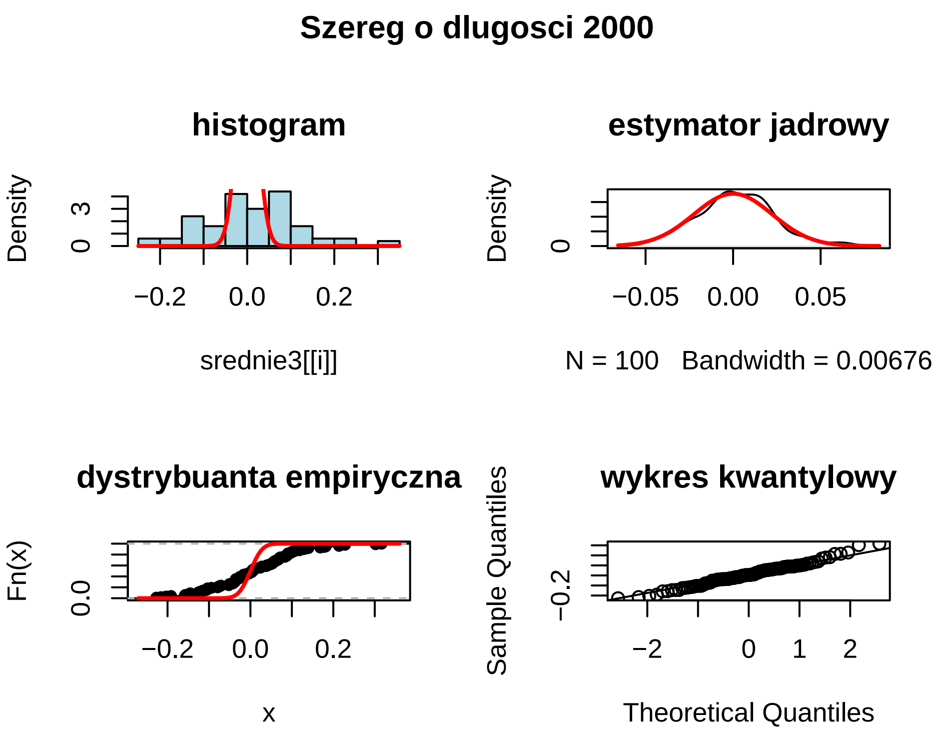
```
par(mfrow=c(2,2))
par(oma=c(0,0,2,0))
for (i in 1:length(n)){
  #par(mfrow=c(2,2))
  hist(srednie3[[i]], probability=T, col="lightblue", main = 'histogram')
  curve(dnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  plot(density(srednie[[i]]), main = 'estymator jądrowy')
  curve(dnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  plot(ecdf(srednie3[[i]]), main = 'dystrybuanta empiryczna')
  curve(pnorm(x,mean=0,sd=1/sqrt(n[i])), col="red", add=T, lwd=2)

  qqnorm(srednie3[[i]], main = 'wykres kwantylowy')
```

```
  qqline(srednie3[[i]])

  title(paste0('Szereg o długości ',as.character(n[i])), outer = TRUE)
  par(oma=c(0,0,2,0))
}
```










```
Shapiro31 = list()
KStrue31 = list()
for (i in 1:length(n)){
  Shapiro31[i] <- shapiro.test(srednie3[[i]])[["p.value"]]
  KStrue31[i] <- ks.test(srednie3[[i]], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.value"]]
}
results31 <- matrix(data = c(Shapiro31, KStrue31), nrow = 2, byrow = TRUE, dimnames = li
```

Tabela 19: P wartości dla szeregu długości 2000

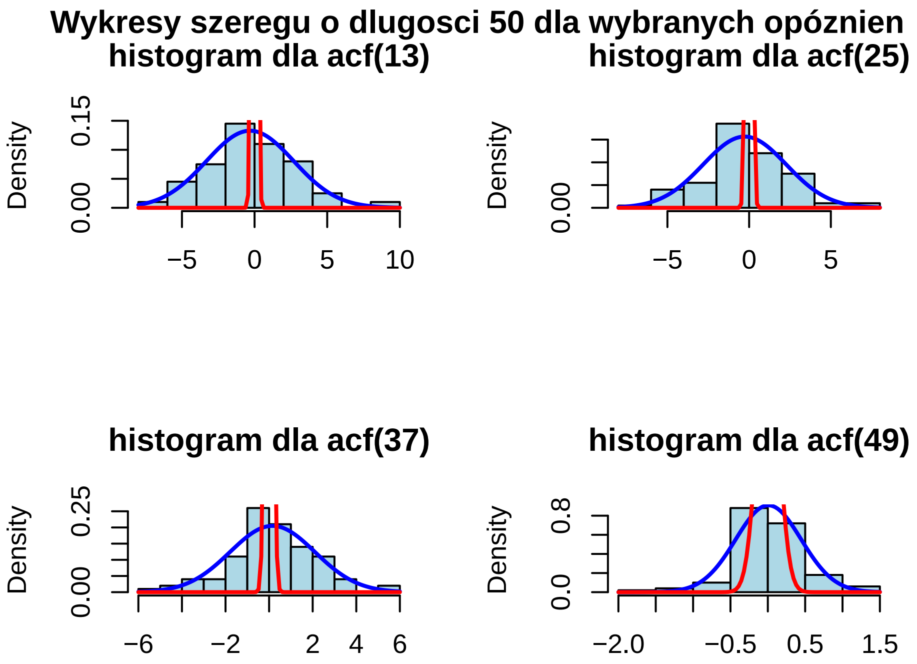
	50	100	500	2000
Shapiro	0.125122357415992	0.384333385585387	0.96241332958939	0.49842423767247
KS	1.09817488436192e-10	1.50623957750895e-12	2.62123656113999e-13	9.47242284610184e-13

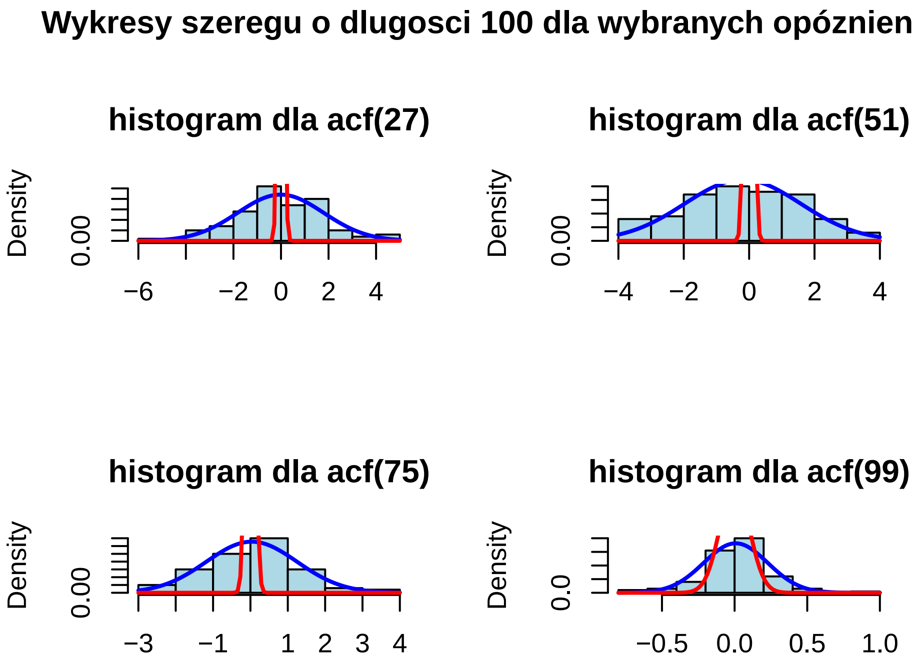
```
kable(results31, format = "latex", digits = 2, caption = paste0("P wartości dla szeregu
```

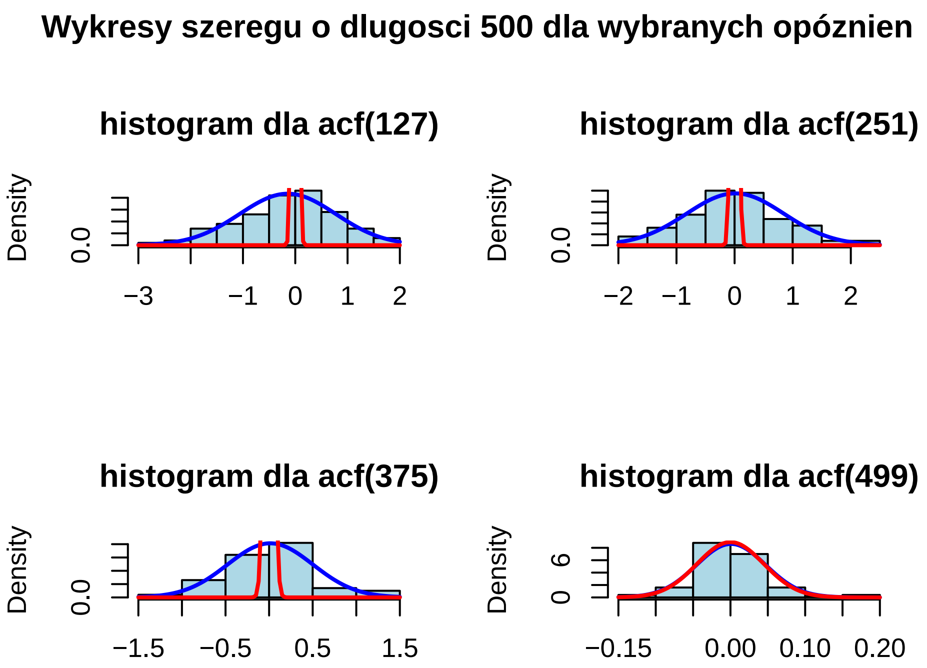
Na podstawie przedstawionej analizy, widzimy że zmiana wariancji wpłynęła bardziej na szeregi o mniejszym n. Dla n=2000 wykresy niemal pokrywają się z wykresami dla rozkładu $N(0, \frac{1}{\sqrt{n}})$. Pomimo, że test Kołmogorowa-Smirnova wygenerował same wartośći bliskie zera, to test Shapiro-Wilka (który jest mocniejszym testem dla prób takiej wielkości) nie daje podstaw do odrzucenia hipotezy zerowej.

1.3.2 Estymator funkcji autokowariancji

```
acovf.matrix3 <- vector(mode = "list", length = 0)
h.wybrane3 <- list()
acovf.h3 <- list()

h.max3 <- floor(n-1)
for (i in 1:length(n)){
  acovf.matrix3[[i]] <- apply(realizacje3[[i]], 2, function(x) acf(x, lag.max=h.max3[i],
  acovf.matrix3[[i]] <- acovf.matrix3[[i]][-1,]
  h.wybrane3[[i]] <- seq(to = h.max3[i], by = floor(h.max3[i]/4), length.out = 4)
}


for (i in 1:length(n)){
  par(mfrow=c(2,2))
  for (h in h.wybrane3[[i]])
  {
    tytul <- paste0("histogram dla acf(",h,")")
    acovf.h3[i] <- list(acovf.matrix3[[i]][h,])

    hist( acovf.h3[[i]], freq=FALSE, col="lightblue", main=tytul, xlab="")
    curve(dnorm(x,mean=mean(acovf.h3[[i]]), sd=sd(acovf.h3[[i]])), add=T, col="blue", lw
    curve(dnorm(x,mean=0, sd=1/sqrt(n[i])), add=T, col="red", lwd=2)

  }
  par(oma=c(0,0,2,0))
  title(paste0('Wykresy szeregu o długości ',as.character(n[i]),' dla wybranych opóźnień
}
```


Wykresy szeregu o dlugosci 50 dla wybranych opóznien
histogram dla acf(13)
histogram dla acf(25)
histogram dla acf(37)
histogram dla acf(49)
Density
0.00
0.15
0.25
0.0
0.8
−5
0
5
10
−6
−2
2
4
6
−2.0
−0.5
0.5
1.5


Wykresy szeregu o dlugosci 100 dla wybranych opóznien
histogram dla acf(27)
histogram dla acf(51)
histogram dla acf(75)
histogram dla acf(99)
Density
0.00
0.0
−6
−2
0
2
4
−4
−3
−1
1
3
−0.5
0.0
0.5
1.0


Wykresy szeregu o dlugosci 500 dla wybranych opóznien
histogram dla acf(127)
histogram dla acf(251)
histogram dla acf(375)
histogram dla acf(499)
Density

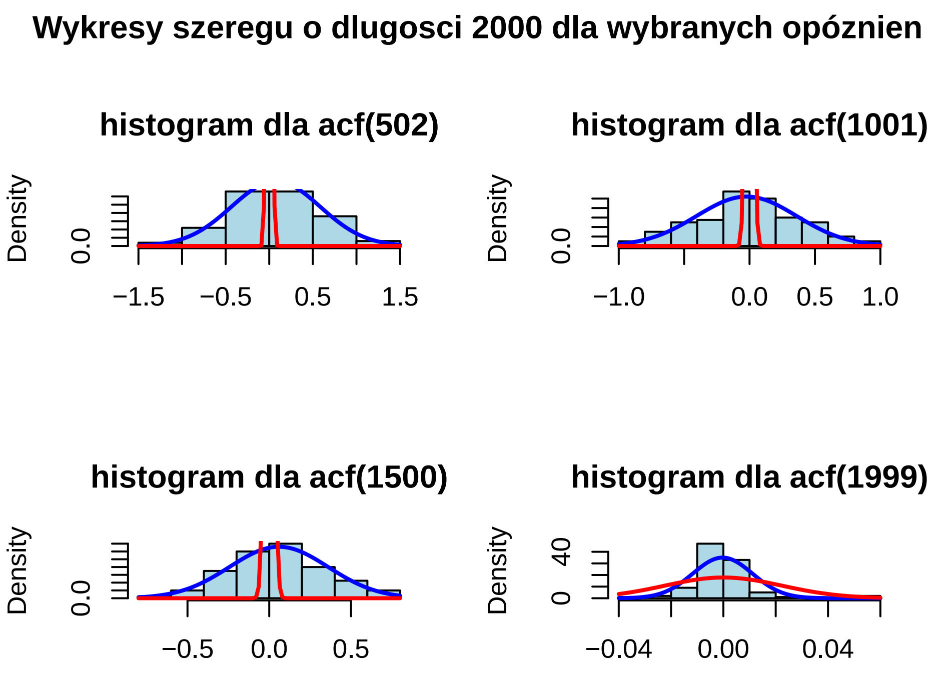

Wykresy szeregu o dlugosci 2000 dla wybranych opóznien
histogram dla acf(502)
histogram dla acf(1001)
histogram dla acf(1500)
histogram dla acf(1999)
Density

Tabela 20: P wartości dla szeregu długości 50

	13	25	37	49
Shapiro	0.626353745224974	0.560516779105041	0.236546214169498	0.000384850087054805
KS	0	0	7.7715611723761e-16	3.01121982000385e-05

Tabela 21: P wartości dla szeregu długości 100

	27	51	75	99
Shapiro	0.676004502142755	0.343684427542027	0.543677351330363	2.61274139067624e-05
KS	0	0	0	0.0333750131152697

```
Shapiro32 = list()
KS32 = list()
num = 1
for (i in 1:length(n)){
  num = 1
  for (h in h.wybrane3[[i]]){
    Shapiro32[num] <- shapiro.test(acovf.matrix3[[i]][h,])[["p.value"]]
    KS32[num] <- ks.test(acovf.matrix3[[i]][h,], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.v
    num = num+1
  }
  #results2 <- matrix(data = c(Shapiro2, KS2), nrow = 2, byrow = TRUE, dimnames = list
  results32 <- matrix(data = c(Shapiro32, KS32), nrow = 2, byrow = TRUE, dimnames = list
    print(kable(results32, format = "latex", digits = 2, caption = paste0("P wartości dl
}
```

Zarówno wykresy, jak i test Shapiro-Wilka nie daje nam podstaw do odrzucenia, że dane pochodzą z rozkładu normalnego. Dalej widać tą samą tendencję, że estymatory funkcji dla h bliskich h.max nie radzą sobie z dokładną estymacją.

1.3.3 Estymator funkcji autokorelacji

```
acf.matrix3 <- vector(mode = "list", length = 0)
acf.h3 <- list()

for (i in 1:length(n)){
  acf.matrix3[[i]] <- apply(realizacje3[[i]], 2, function(x) acf(x, lag.max=h.max3[i], t
  acf.matrix3[[i]] <- acf.matrix3[[i]][-1,]
```

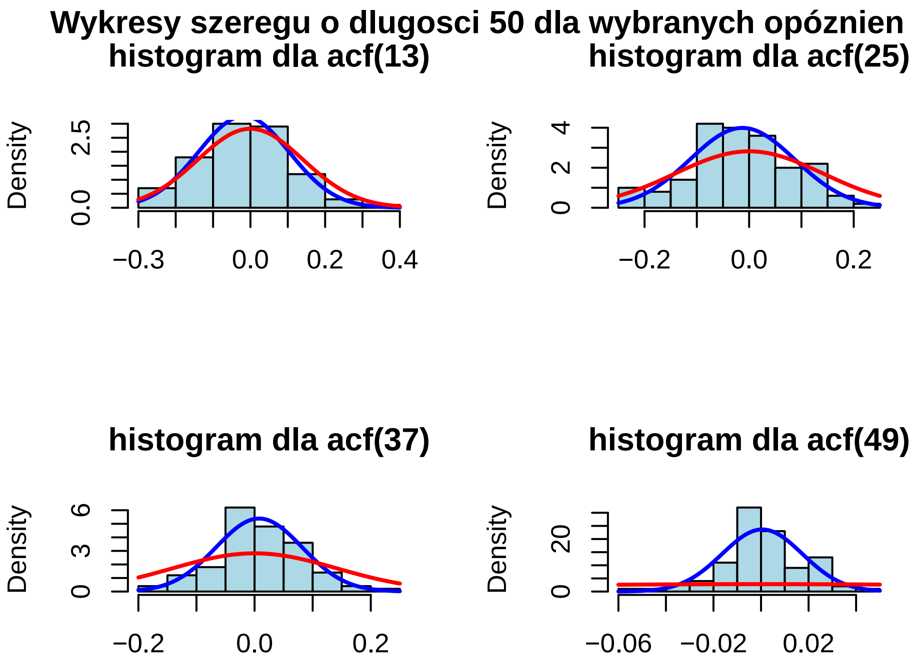
Tabela 22: P wartości dla szeregu długości 500

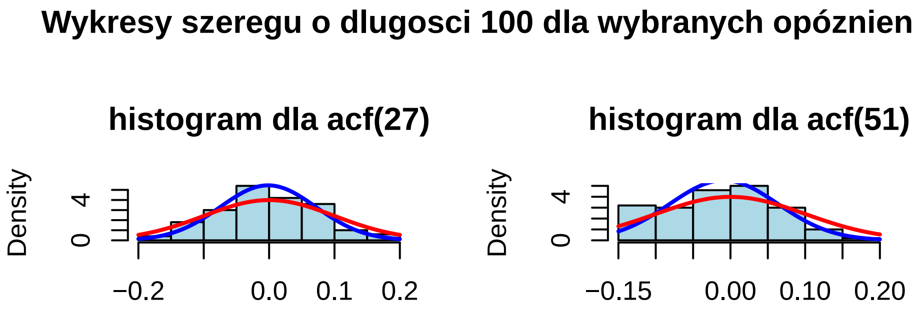
	127	251	375	499
Shapiro	0.60172268534831	0.806976262915606	0.572905525756955	8.09704651238232e-05
KS	0	0	0	0.156294894915424

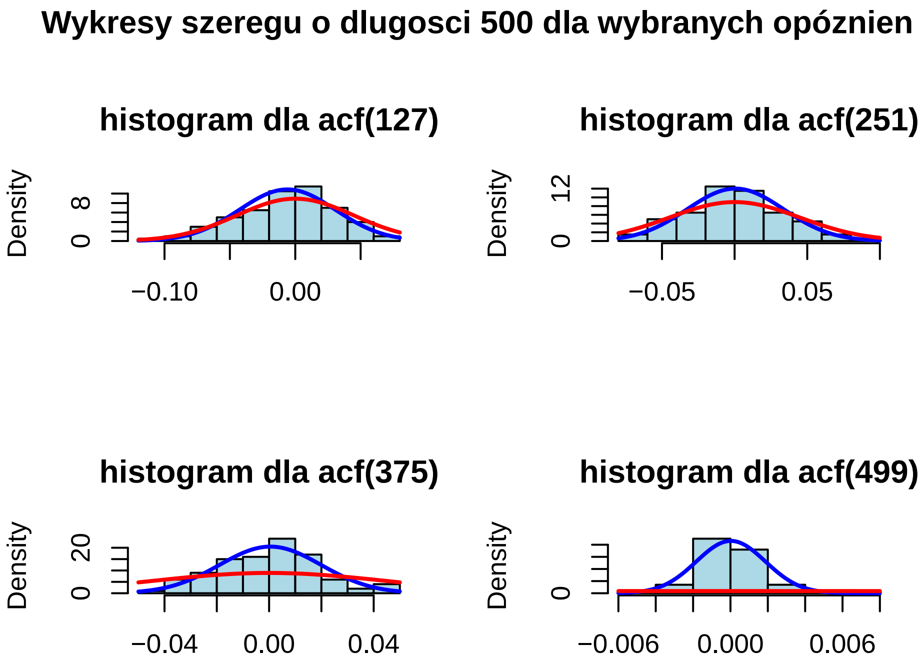
Tabela 23: P wartości dla szeregu długości 2000

	502	1001	1500	1999
Shapiro	0.856535192539786	0.956729439561614	0.548719554691134	1.15035969851717e-09
KS	0	0	0	3.94647574297746e-06

```
  h.wybrane3[[i]] <- seq(to = h.max3[i], by = floor(h.max3[i]/4), length.out = 4)
}


for (i in 1:length(n)){
  par(mfrow=c(2,2))
  for (h in h.wybrane3[[i]])
  {
    tytul <- paste0("histogram dla acf(",h,")")
    acf.h3[i] <- list(acf.matrix3[[i]][h,])

    hist( acf.h3[[i]], freq=FALSE, col="lightblue", main=tytul, xlab="")
    curve(dnorm(x,mean=mean(acf.h3[[i]]), sd=sd(acf.h3[[i]])), add=T, col="blue", lwd=2)
    curve(dnorm(x,mean=0, sd=1/sqrt(n[i])), add=T, col="red", lwd=2)

  }
  par(oma=c(0,0,2,0))
  title(paste0('Wykresy szeregu o długości ',as.character(n[i]),' dla wybranych opóźnień
}
```




Wykresy szeregu o dlugosci 100 dla wybranych opóznien
histogram dla acf(27)
histogram dla acf(51)
histogram dla acf(75)
histogram dla acf(99)
Density


Wykresy szeregu o dlugosci 500 dla wybranych opóznien
histogram dla acf(127)
histogram dla acf(251)
histogram dla acf(375)
histogram dla acf(499)
Density

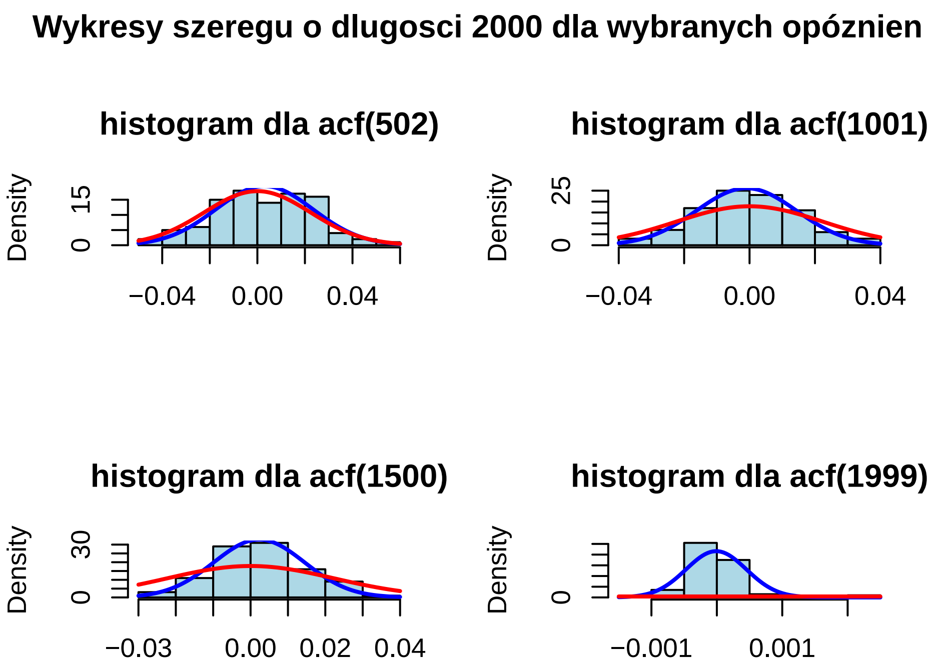

Wykresy szeregu o dlugosci 2000 dla wybranych opóznien
histogram dla acf(502)
Density
0
15
−0.04
0.00
0.04
histogram dla acf(1001)
Density
0
25
−0.04
0.00
0.04
histogram dla acf(1500)
Density
0
30
−0.03
0.00
0.02
0.04
histogram dla acf(1999)
Density
0
−0.001
0.001

Tabela 24: P wartości dla szeregu długości 50

	13	25	37	49
Shapiro	0.966945151932019	0.652097100253609	0.798249231908545	0.0209694584697625
KS	0.374636064563367	0.0797232033662928	0.00108708402335966	1.23456800338317e-13

Tabela 25: P wartości dla szeregu długości 100

	27	51	75	99
Shapiro	0.978605206236239	0.191330064619081	0.866715982259423	3.6663105405057e-05
KS	0.259024926874749	0.0556109133908439	0.00248158857377279	9.88098491916389e-15

```
Shapiro33 = list()
KS33 = list()
par(mfrow=c(2,2))
for (i in 1:length(n)){
  num = 1
  for (h in h.wybrane3[[i]]){
    Shapiro33[num] <- shapiro.test(acf.matrix3[[i]][h,])[["p.value"]]
    KS33[num] <- ks.test(acf.matrix3[[i]][h,], "pnorm", mean=0, sd=1/sqrt(n[i]))[["p.val
    num = num+1
  }
  results33 <- matrix(data = c(Shapiro33, KS33), nrow = 2, byrow = TRUE, dimnames = list
  print(kable(results33, format = "latex", digits = 2, caption = paste0("P wartości dla
}
```

Wnioski analogiczne jak dla funkcji autokowariancji.

1.4 Podsumowanie

Główne wnioski z powyższej analizy:

- Wraz z długością szeregu, dane przypominają coraz dokładniej rozkład normalny
- Warto powołać się na jak najwięcej metod sprawdzających zgodność rozkładu, zarówno grafincznych, jak i testowych. Czasami ciężko nam zinterpretować wykresy, a w innych przypadkach formalne testy nie radzą sobie (np. z powodów wielkości próby).
- Wariancja w rozkładzie normalnym nie wpływa na wyniki testów.
- Estymatory autokowariancji i autokorelacji 'nie radzą' sobie dla dużych wartości opóźnień h.

Tabela 26: P wartości dla szeregu długości 500

	127	251	375	499
Shapiro	0.501237934195918	0.958441855238453	0.455845082411005	8.88540987718852e-05
KS	0.0488425174593551	0.340436160811761	0.000184390361938402	0

Tabela 27: P wartości dla szeregu długości 2000

	502	1001	1500	1999
Shapiro	0.851606607335685	0.907518459715677	0.79685647075752	5.82448230216126e-10
KS	0.288534469453903	0.143133563339012	0.000761056799644	0

- Estymatory autokowariancji i autokorelacji lepiej estymują wartości dla dłuższych szeregów.