

Analiza szeregów czasowych - notowania giełdowe firm McDonald's oraz Starbucks

Alicja Hołowiecka, Matylda Jankowska, Marcin Dziadosz

23 12 2019

Spis treści

Matylda!! (to do, to do, to do to do to do to doooooooo)	1
Wstęp	2
Wczytanie bibliotek	2
McDonald's	2
Opis firmy	2
Wczytanie danych i rysunki	3
Wydzielanie trendu	3
Ruchoma średnia	3
Metoda wykładniczych wag ruchomej średniej	5
Dopasowanie wielomianu	6
Metoda różnicowa	6
Dopasowanie modelu liniowego	7
Dopasowanie wielomianem drugiego stopnia	8
Dopasowanie wielomianem trzeciego stopnia	10
Dopasowanie wielomianem czwartego stopnia	13
Testy na resztach modelu	13
Jednorodność wariancji	13
Normalność	15
Autokorelacja	16
Stacjonarność	19
ARIMA	21
Holt - Winters	22
Podsumowanie analizy notowań McDonald's	22
Starbucks	22
Opis firmy	22
Wczytanie danych	25
Rysunek	26
Dopasowanie wielomianu	26
Model liniowy	26
Drugiego stopnia	28
Trzeciego stopnia	29
Ruchoma średnia	30
Metoda wykładniczych wag ruchomej średniej	33
Testy na resztach modelu sześciennego	37
Metoda różnicowa	42
Stacjonarność	43

Matylda!! (to do, to do, to do to do to do to doooooooo)

- estetycznie można trochę poprawić, np. jak są te testy kpss i inne, to na razie robiłam tak, żeby wyświetlało ten domyślny output, może zamiast tego zrób tak jak Ty robiłaś w testach, że zapisać p-value do zmiennej i opisać? i popatrz jak wyglądają wykresy, szczególnie w pdf, może coś nie wygląda

tak dobrze jak się nam wydawało, np tytuły się nie mieszczą, albo coś jest za bardzo ściśnięte, np wiem że te wykresy dla wielomianu 4 stopnia z nieznanymi dla mnie przyczyn strasznie odwalają w pdf

- jeżeli nadal coś się nie zgadza z tą kolejnością co Kozłowski proponował, to możesz poprzestawiać
- co do opisów firm - ze Starbucksa trzeba wywalić zdjęcia bo dupnie wyglądają w pdf. I nie wiem czy ewentualnie z neta nie skopiować ze dwa zdania więcej do każdego, bo jakoś blado. spadliśmy już do 44 stron także jest coraz lepiej :) a jak wywalimy rysunki to jeszcze kolejne 3-4 strony pójdą
- sprawdź, jak wyglądają tabelki w pdfie - czy na pewno każda się wyświetla dokładnie gdzie ma być. Jeśli nie, to np dla tabelki robionych poprzez `kable_styling` można dodawać parametry: `position = 'center'`, `latex_options = "HOLD_position"` i powinny się trzymać swojego miejsca
- za Chiny ludowe nie wiem jak robić jakieś prognozy czy cokolwiek... bo ten holt winters to chyba tylko dla sezonowych idzie. zobacz czy coś się da, a jak nie to usuwaj to gadownictwo.
- co do ARIMY - opisałam tak, jak na laborkach robiliśmy, zobacz czy z tego modelu da się wyciągnąć reszty i robić na nich testy. jeśli się da, to zrób te testy
- myślałam żeby na koniec każdej z tych firm zrobić takie podsumowanie - że np dla McDonalda z modeli wielomianowych najlepszy był taki i taki, jego reszty spełniają/nie spełniają cośtam, oprócz tego dopasowaliśmy model ARIMA taki śmaki, jego reszty cośtam cośtam... jakieś takie ładne sratatata na zakończenie
- pracuj przede wszystkim nad McDonaldem, potem Marcin spisze do Starbucksa. Chociaż tam też oczywiście możesz coś zacząć żeby mu ułatwić.

Wstęp

W tym raporcie przeanalizujemy dwa szeregi czasowe: notowania firm McDonald's oraz Starbucks z okresu dwóch lat (od początku 2018 do końca 2019). Na potrzeby oceny w raporcie pojawia się nie tylko sama analiza, ale też wszystkie polecenia w języku R, jakich używaliśmy w jej celu.

Wczytanie bibliotek

Na początek wczytamy wszystkie potrzebne biblioteki. Biblioteka `tseries` przyda nam się m. in. do wczytania danych oraz wykonania testów na stacjonarność szeregu. Z pakietu `randtests` skorzystamy przy testach na losowość reszt. W bibliotece `nortest` znajduje się wiele testów na normalność. Dzięki paczce `lmtest` utworzymy modele wielomianowe różnego stopnia i zbadamy ich dopasowanie. W bibliotece `forecast` są m. in. funkcje dotyczące modeli ARIMA. Pakiety `stargazer`, `tables` i `kableExtra` pozwolą nam estetycznie wyświetlać tabele.

```
library(tseries)
library(randtests)
library(nortest)
library(lmtest)
library(forecast)
library(stargazer)
library(kableExtra)
library(tidyverse)
library(tables)
```

McDonald's

Opis firmy

McDonald's to największa na świecie sieć restauracji szybkiej obsługi. Obejmuje ona ponad 30 tys. restauracji, każdego dnia obsługujących ponad 46 mln osób w 119 krajach. Wartość marki McDonald's szacuje się na 24,7 mld dolarów.

Wczytanie danych i rysunki

Dane pobieramy z yahoo finance za pomocą funkcji `get.hist.quote` i zamieniamy na typ numeryczny.

```
mcd<- get.hist.quote(instrument = "MCD", provider = "yahoo",  
                    quote = "Close", start = "2018-01-01", end = "2019-12-31")
```

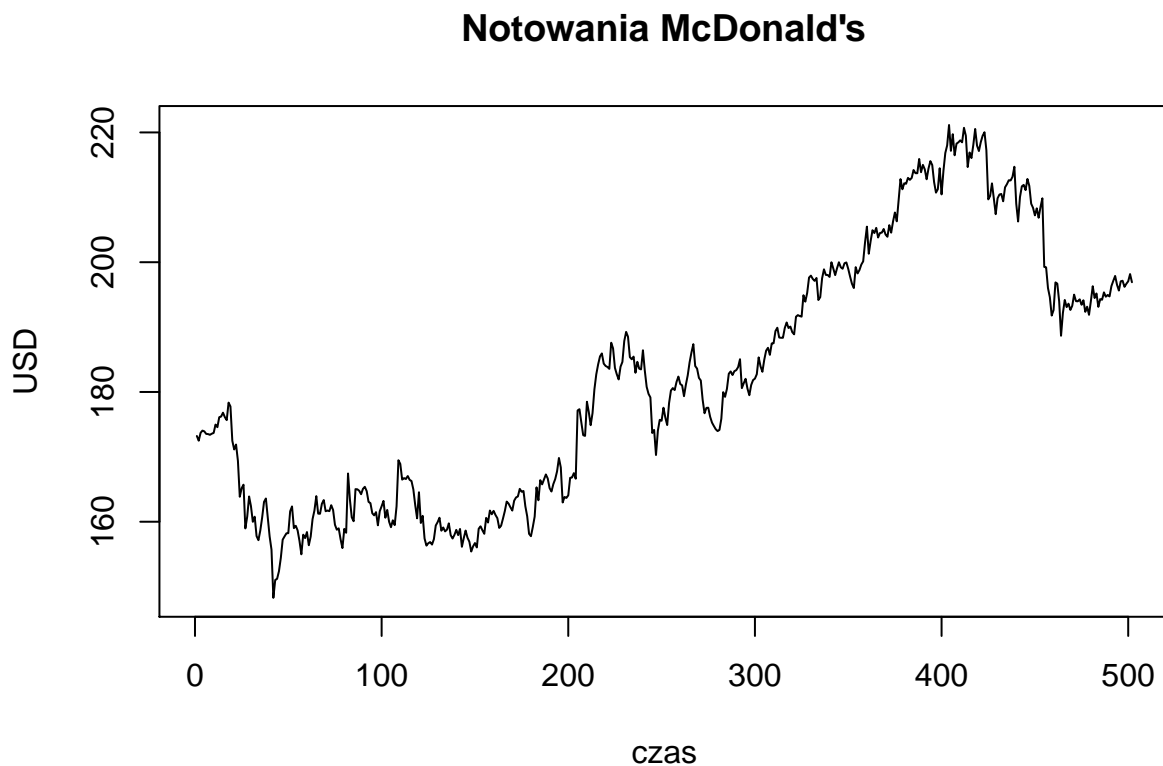
```
## time series starts 2018-01-02
```

```
## time series ends 2019-12-30
```

```
mcd <- as.numeric(mcd)
```

Wykonamy rysunek przedstawiający notowania firmy McDonald's od 01-01-2018 do 31-12-2019

```
plot(mcd, type = "l", xlab = "czas", ylab = "USD", main = "Notowania McDonald's")
```



Na rysunku w ciągu tych dwóch lat wyraźnie widać trend rosnący.

Wydzielanie trendu

Spróbujemy wydzielić część deterministyczną. Do tego celu posłużymy się metodami średnich ruchomych (prostą i wykładniczą).

Ruchoma średnia Wykorzystamy metody ruchomych średnich, aby wygładzić szereg i zaobserwować ogólne trendy. Metoda średniej ruchomej ma na celu zmniejszenie rozrzutu razy $m + 1$.

W metodzie średniej ruchomej estymator części deterministycznej ma postać

$$\hat{f}(t) = \frac{1}{m+1} \sum_{k=0}^m x_{t-k}$$

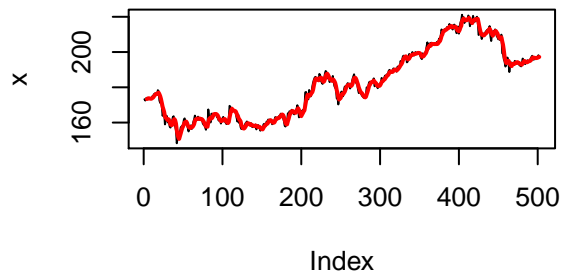
Do wykonania wygładzonych wykresów napisaliśmy funkcję `ruchoma`, której argumentami są `x` - szereg czasowy, `m` - paramter metody średniej ruchomej, `kolor` - kolor, na jaki dorysujemy wygładzoną linię na wykresie.

```
ruchoma <- function(x, m, kolor){
  t <- length(x)
  f <- NULL
  for(i in (m+1):t){
    f[i] <- mean(x[(i-m):i])
  }
  tytul = paste("Średnia ruchoma rzędu ", m)
  plot(x, type = "l", main = tytul)
  lines((m+1):t, f[(m+1):t], lwd = 2, col = kolor)
}
```

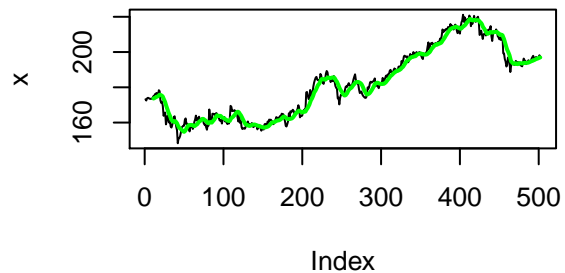
Narysujemy wykresy dla kilku parametrów `m`.

```
par(mfrow = c(2, 2))
ruchoma(mcd, 3, "red")
ruchoma(mcd, 10, "green")
ruchoma(mcd, 30, "blue")
ruchoma(mcd, 50, "pink")
```

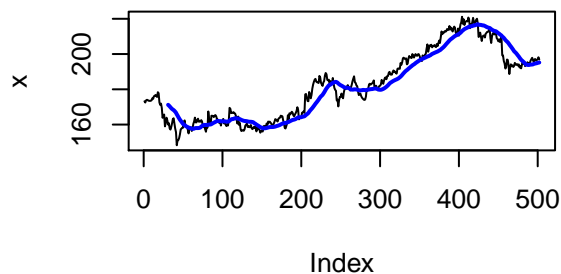
Srednia ruchoma rzędu 3



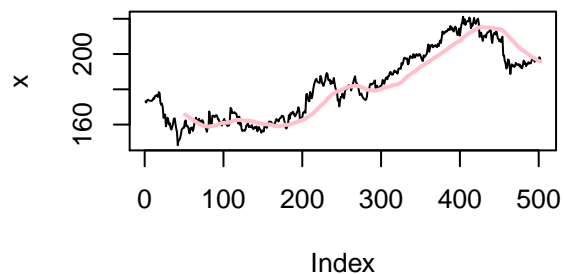
Srednia ruchoma rzędu 10



Srednia ruchoma rzędu 30



Srednia ruchoma rzędu 50



```
par(mfrow = c(1, 1))
```

Jak widać, im większy parametr m przyjmiemy, tym bardziej wygładzony wykres uzyskujemy, ale też mniej dokładny.

Metoda wykładniczych wag ruchomej średniej W metodzie ruchomej średniej obserwacje starsze i nowsze mają taką samą wagę, dlatego ta metoda jest mało dokładna. Skorzystamy teraz z dokładniejszej metody wykładniczych wag ruchomej średniej.

W tej metodzie estymator części deterministycznej ma postać:

$$\hat{f}(t) = \frac{1-\eta}{1-\eta^t} \sum_{k=0}^{t-1} \eta^k x_{t-k}$$

gdzie $\eta \in (0, 1)$

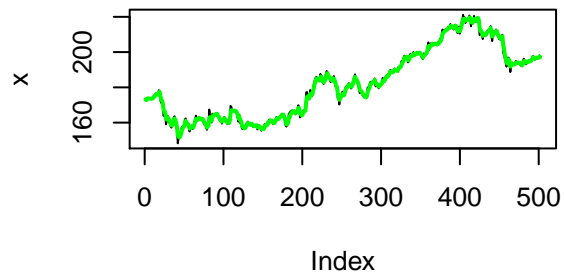
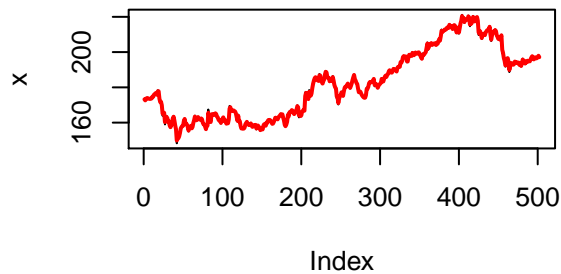
Skorzystamy z postaci rekurencyjnej:

$$\hat{f}(t) = \frac{1-\eta}{1-\eta^t} \left[x_t + \eta \frac{1-\eta^{t-1}}{1-\eta} \hat{f}(t-1) \right]$$

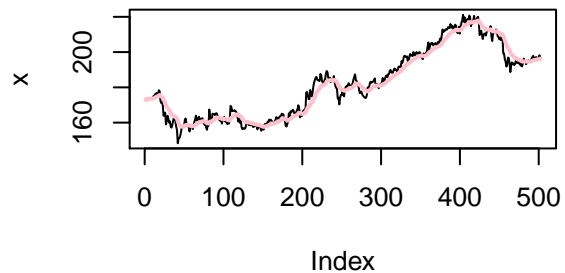
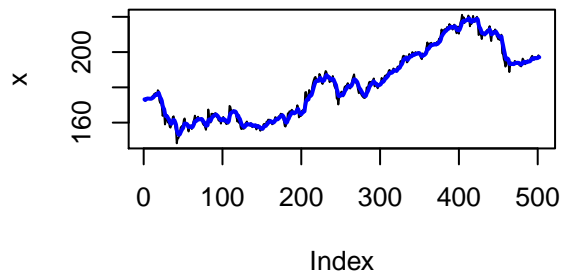
```
wykladnicza <- function(x, mi, kolor){  
  f <- NULL  
  f[1] <- x[1]  
  
  for (i in 2:length(x)){  
    f[i] <- (1-mi)/(1-mi^i)*(x[i]+mi*(1-mi^(i-1)))/(1-mi)*f[i-1])  
  }  
  tytul = paste("Średnia ruchoma z wagami wykładniczymi z parametrem ", mi)  
  plot(x, type = "l", main = tytul)  
  lines(1:length(x), f, lwd = 2, col = kolor)  
}
```

```
par(mfrow = c(2, 2))  
wykladnicza(mcd, 0.2, "red")  
wykladnicza(mcd, 0.5, "green")  
wykladnicza(mcd, 0.7, "blue")  
wykladnicza(mcd, 0.9, "pink")
```

ruchoma z wagami wykładniczymi z para ruchoma z wagami wykładniczymi z para



ruchoma z wagami wykładniczymi z para ruchoma z wagami wykładniczymi z para



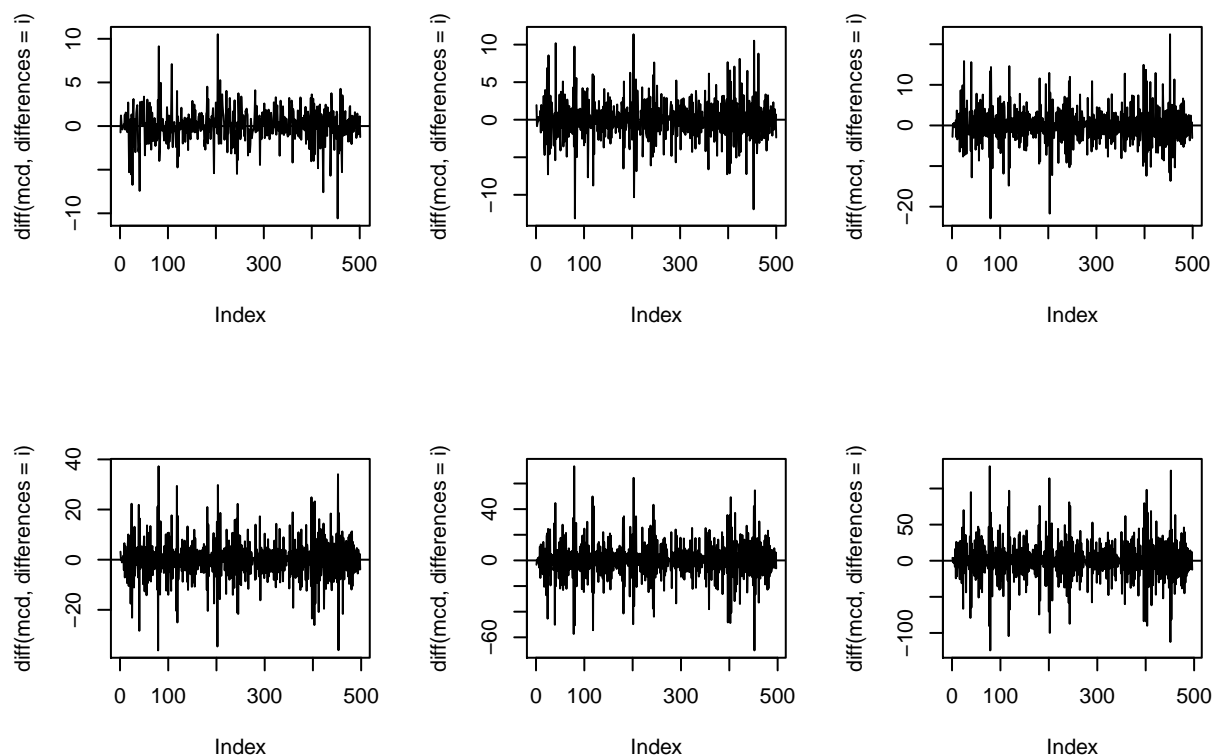
```
par(mfrow = c(1, 1))
```

Podobnie jak w przypadku prostej metody średniej ruchomej - im większy parametr η , tym bardziej wygładzony wykres, ale i mniejsza dokładność. Jednakże, dokładność jest i tak większa niż w przypadku prostej metody ruchomej średniej.

Dopasowanie wielomianu

Metoda różnicowa Za pomocą metody różnicowej sprawdzimy, jaki stopień wielomianu byłby najbardziej odpowiedni.

```
par(mfrow = c(2, 3))
for(i in 1:6){
  plot(diff(mcd, differences = i), type = "l")
  abline(h = 0)}
```



```
par(mfrow=c(1,1))
```

Spróbujemy do danych dopasować wielomian stopnia 1, 2, 3 i 4.

```
t <- 1:length(mcd)
```

```
mod1 <- lm(mcd~t)
```

```
stargazer(mod1, header = F)
```

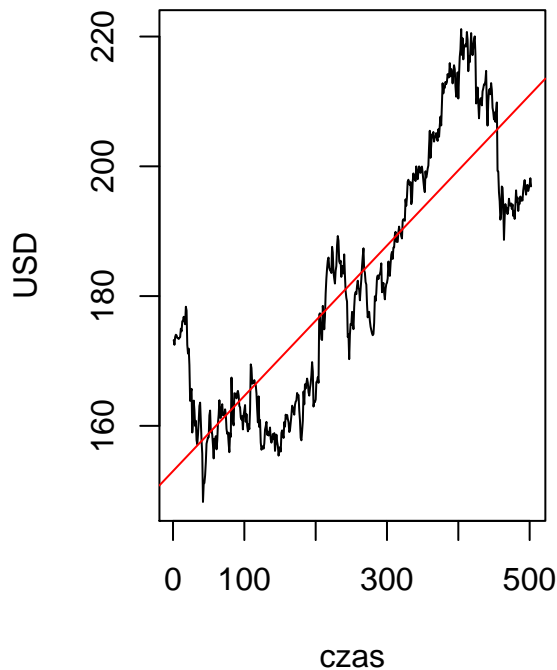
Dopasowanie modelu liniowego Zarówno wyraz wolny, jak i współczynnik kierunkowy są istotne statystycznie. R^2 wynosi około 74%.

```
par(mfrow = c(1, 2))
plot(mcd, type = "l", main = "Model liniowy", xlab = "czas", ylab = "USD")
abline(mod1, col = "red")
plot(mod1$residuals, type = "l",
      main = "Reszty modelu liniowego", xlab = "czas", ylab = "reszty")
abline(h=0)
```

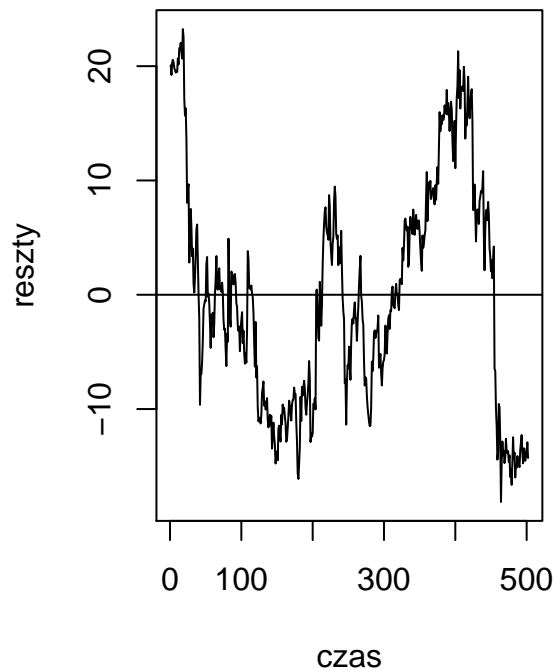
Tablica 1:

<i>Dependent variable:</i>	
mcd	
t	0.116*** (0.003)
Constant	153.029*** (0.885)
Observations	502
R ²	0.743
Adjusted R ²	0.742
Residual Std. Error	9.905 (df = 500)
F Statistic	1,442.546*** (df = 1; 500)
<i>Note:</i> *p<0.1; **p<0.05; ***p<0.01	

Model liniowy



Reszty modelu liniowego



```
par(mfrow = c(1, 1))
```

Dopasowanie wielomianem drugiego stopnia Teraz stworzymy model wielomianowy drugiego stopnia.

```
mod2 <- lm(mcd~t+I(t^2))
```

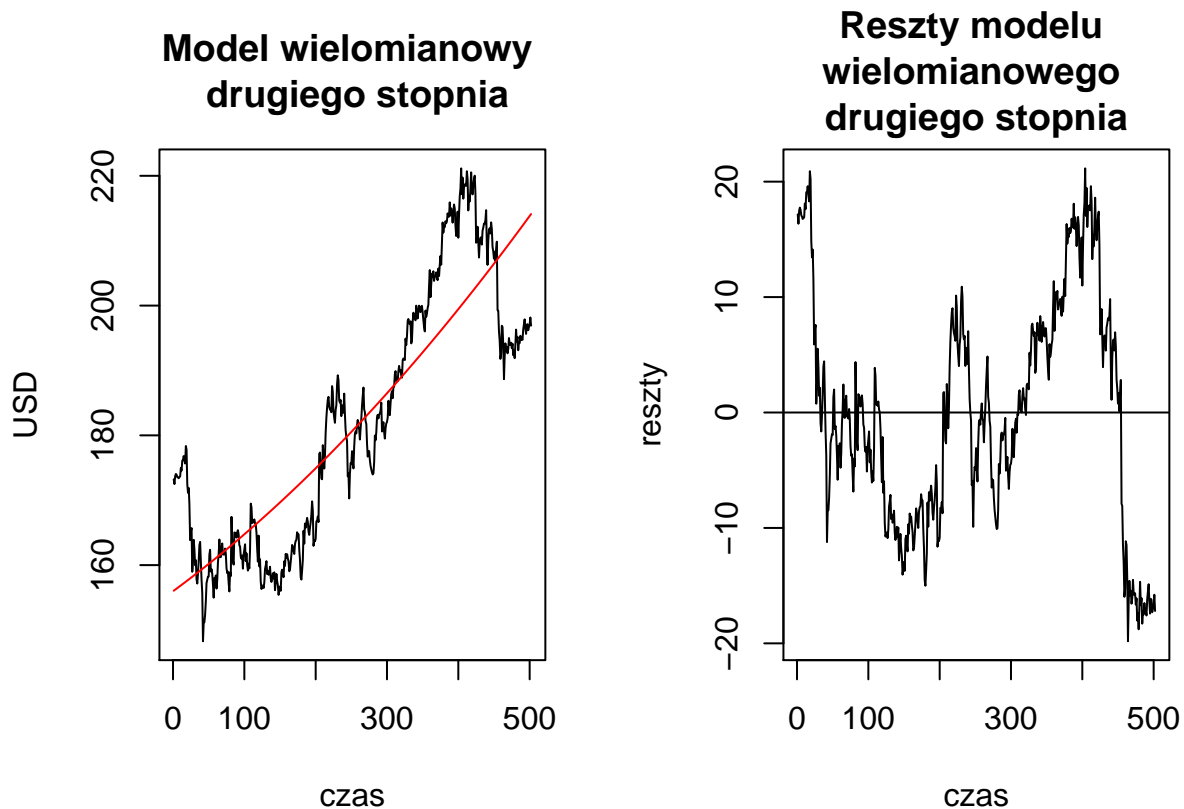


```
stargazer(mod2, header = F)
```

Tablica 2:	
	<i>Dependent variable:</i>
	mcd
t	0.081*** (0.012)
I(t ²)	0.0001*** (0.00002)
Constant	155.975*** (1.321)
Observations	502
R ²	0.747
Adjusted R ²	0.746
Residual Std. Error	9.828 (df = 499)
F Statistic	737.143*** (df = 2; 499)
<i>Note:</i>	*p<0.1; **p<0.05; ***p<0.01

Wszystkie współczynniki są istotne statystycznie. R^2 wynosi około 75%, a więc zmieniło się bardzo nieznacznie.

```
par(mfrow = c(1, 2))
plot(mcd, type = "l", main = "Model wielomianowy \n drugiego stopnia",
     xlab = "czas", ylab = "USD")
lines(t, mod2$fitted.values, col = "red", )
plot(mod2$residuals, type = "l",
     main = "Reszty modelu \nwielomianowego \ndrugiego stopnia",
     xlab = "czas", ylab = "reszty")
abline(h = 0)
```



```
par(mfrow = c(1, 1))
```

Model kwadratowy zachowuje się bardzo podobnie jak model liniowy.

```
mod3 <- lm(mcd~t+I(t^2)+I(t^3))
```

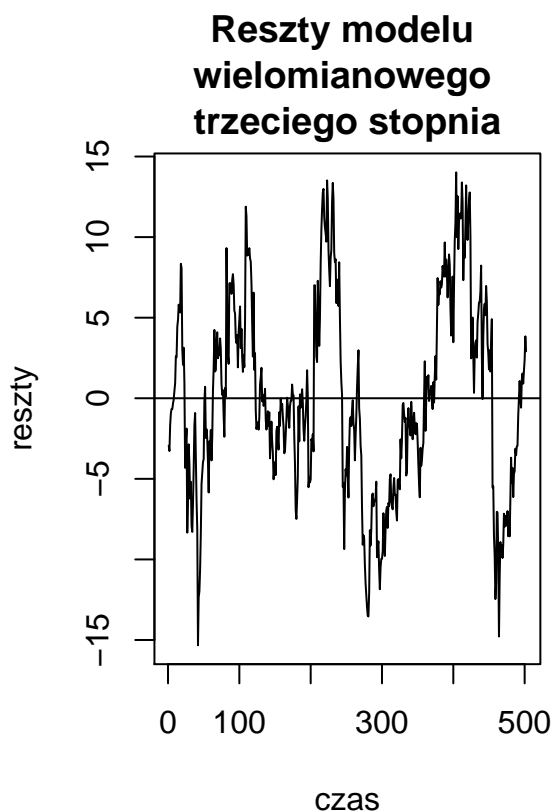
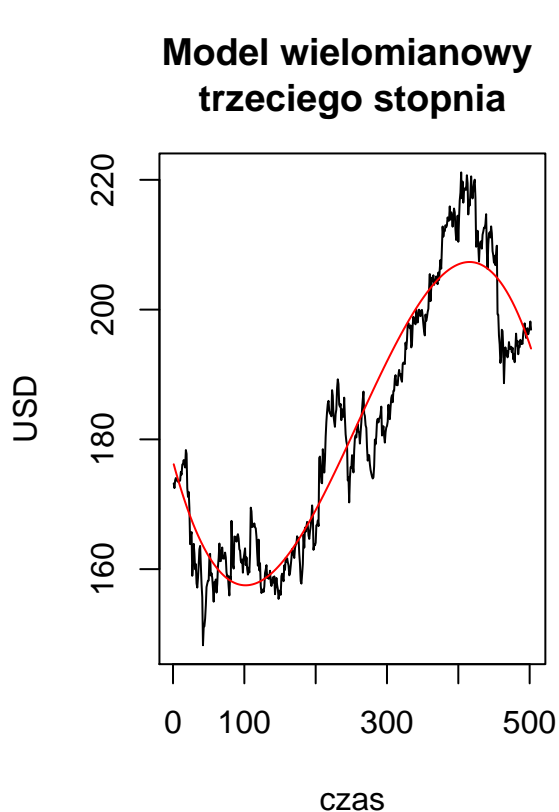
```
stargazer(mod3, header = F)
```

Dopasowanie wielomianem trzeciego stopnia W modelu wielomianowym trzeciego stopnia wszystkie współczynniki są istotne statystycznie. R^2 wynosi 90%, a więc znacząco się poprawił w stosunku do poprzednich dwóch modeli.

```
par(mfrow = c(1, 2))
plot(mcd, type = "l",
     main = "Model wielomianowy \ntrzeciego stopnia",
     xlab = "czas", ylab = "USD")
lines(t, mod3$fitted.values, col = "red")
plot(mod3$residuals, type = "l",
     main = "Reszty modelu \nwielomianowego \ntrzeciego stopnia",
     xlab = "czas", ylab = "reszty")
abline(h= 0)
```

Tablica 3:

<i>Dependent variable:</i>	
mcd	
t	-0.408*** (0.019)
I(t ²)	0.002*** (0.0001)
I(t ³)	-0.00000*** (0.00000)
Constant	176.570*** (1.096)
Observations	502
R ²	0.903
Adjusted R ²	0.902
Residual Std. Error	6.094 (df = 498)
F Statistic	1,544.632*** (df = 3; 498)
<i>Note:</i> *p<0.1; **p<0.05; ***p<0.01	



```
par(mfrow = c(1, 1))
```

Widać, że reszty modelu mają mniejszy rozrzut niż poprzednio - teraz mamy skalę od -15 do 15, a wcześniej było od -20 do 20.

```
mod4 <- lm(mcd~t+I(t^2)+I(t^3)+I(t^4))
```

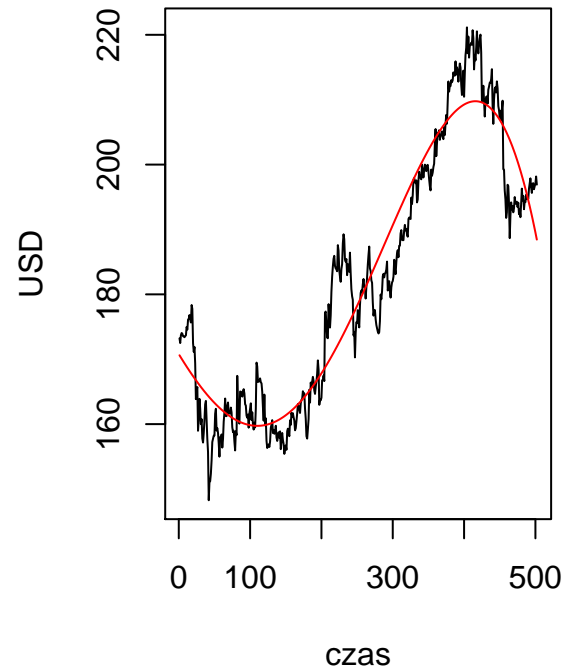
```
stargazer(mod4, header = F)
```

Tablica 4:

	<i>Dependent variable:</i>
	mcd
t	-0.180*** (0.036)
I(t^2)	0.0005 (0.0003)
I(t^3)	0.00000*** (0.00000)
I(t^4)	-0.000*** (0.000)
Constant	170.792*** (1.309)
Observations	502
R ²	0.912
Adjusted R ²	0.912
Residual Std. Error	5.798 (df = 497)
F Statistic	1,293.328*** (df = 4; 497)
<i>Note:</i>	*p<0.1; **p<0.05; ***p<0.01

```
par(mfrow = c(1, 2))
plot(mcd, type = "l", main = "Model wielomianowy \nczwartego stopnia",
     xlab = "czas", ylab = "USD")
lines(t, mod4$fitted.values, col = "red")
plot(mod4$residuals, type = "l",
     main = "Reszty modelu \nwielomianowego \nczwartego stopnia",
     xlab = "czas", ylab = "reszty")
abline(h= 0)
```

Model wielomianowy czwartego stopnia



Dopasowanie wielomianem czwartego stopnia

```
par(mfrow=c(1,1))
```

W modelu wielomianowym czwartego stopnia współczynnik przy t^3 jest nieistotny statystycznie, ale nie możemy go usunąć, ponieważ efekt wyższego rzędu (t^4) jest istotny. R^2 wynosi około 91%, więc niewiele się różni od modelu wielomianowego 3 stopnia. Reszty także znajdują się w podobnym przedziale jak w poprzednim modelu. Model z t^4 niewiele się różni od tego z t^3 , dlatego do dalszych badań wykorzystamy model wielomianowy 3 stopnia.

Testy na resztach modelu

Do danych dobraliśmy wcześniej model wielomianowy trzeciego stopnia. Teraz sprawdzimy, czy reszty tego modelu spełniają założenia:

- losowość
- jednorodność wariancji
- normalność

Jednorodność wariancji Aby zbadać czy jednorodność reszt posłużymy się kilkoma popularnymi testami.

- Test Breuscha-Pagana

H_0 : jednorodność wariancji reszt.

H_1 : wariancja reszt zależy od zmiennych objaśniających w modelu.

```
bptest(mod3)
```

```
##
```

```
## studentized Breusch-Pagan test
##
## data: mod3
## BP = 27.565, df = 3, p-value = 4.483e-06
```

```
pv1 <- bptest(mod3)$p.value
```

P-value wynosi 0.0000044826, zatem Wug testu Breuscha-Pagana należałoby odrzucić hipotezę o jednorodności wariancji reszt.

- Test Goldfelda-Quandt

Weryfikacja hipotezy polega na podziale danych na dwie grupy i sprawdzeniu, czy w obu wariancja ma taką samą wartość.

H_0 : wariancja reszt jest równa w obu grupach.

H_1 : wariancja reszt różni się w obu grupach.

```
gqtest(mod3, order.by = ~fitted(mod3))
```

```
##
## Goldfeld-Quandt test
##
## data: mod3
## GQ = 0.94266, df1 = 247, df2 = 247, p-value = 0.6785
## alternative hypothesis: variance increases from segment 1 to 2
```

```
pv2 <- gqtest(mod3, order.by = ~fitted(mod3))$p.value
```

P-value wynosi 0.678, zatem nie ma podstaw do odrzucenia hipotezy o równości wariancji.

- Test Harrisona-McCabe'a

Sprawdza hipotezę podobną do tej, którą weryfikuje test Goldfelda-Quandt; jednak w tym przypadku porównuje się zależność wariancji reszt dla całości obserwacji i wybranego kwantyla (w tym przypadku rzędu 0.5).

H_0 : wariancja reszt jest równa w porównywanych grupach.

H_1 : wariancja reszt różni się w porównywanych grupach.

```
hmctest(mod3, order.by = ~fitted(mod3))
```

```
##
## Harrison-McCabe test
##
## data: mod3
## HMC = 0.39433, p-value < 2.2e-16
```

```
pv3 <- hmctest(mod3, order.by = ~fitted(mod3))$p.value
```

P-value wynosi jest praktycznie równe 0, należy przyjąć hipotezę alternatywną, czyli wariancja reszt modelu ulega zmianie.

Biorąc pod uwagę uzyskane wyniki, należy przyjąć, że reszty z modelu trzeciego stopnia nie są jednorodne.

```
par(mfrow = c(1, 3))

plot(density(mod3$residuals),
     main = "Wykres gęstości \nrozkładu reszt w \nporównaniu z rozkładem normalnym")
```

```

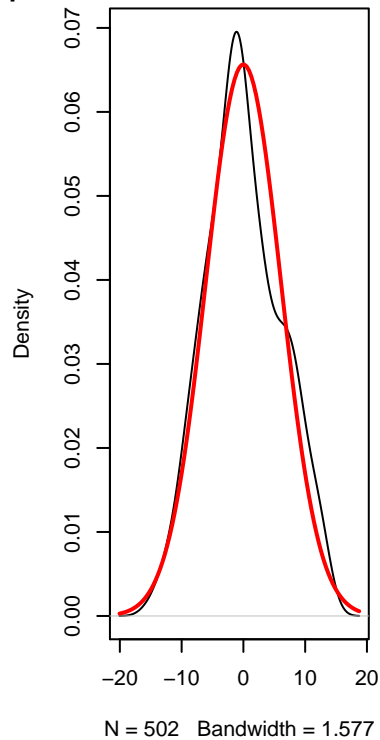
curve(dnorm(x, 0, sd(mod3$residuals)), add = T, col = 2, lwd = 2)

qqnorm(mod3$residuals, main = "Wykres z linią kwantylową")
qqline(mod3$residuals, col=2, lwd = 3)

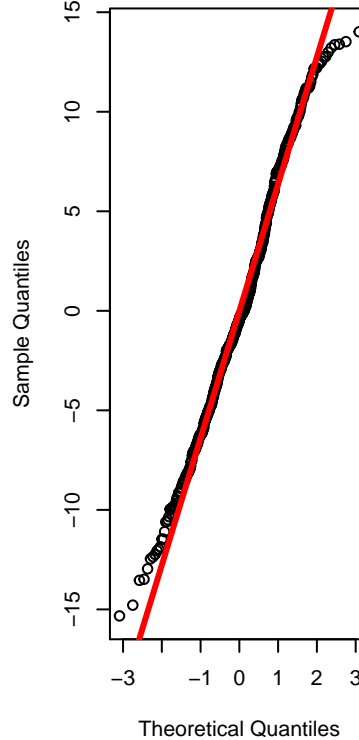
plot(ecdf(mod3$residuals),
     main = "Dystrybuanta empiryczna \nw porównaniu z rozkładem \nnormalnym")
curve(pnorm(x, 0, sd(mod3$residuals)), add = T, col = 2, lwd = 2)

```

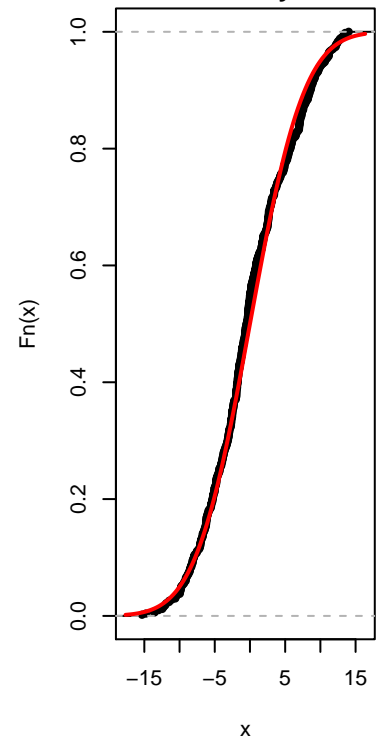
Wykres gestosci
rozkladu reszt w
porównaniu z rozkładem normaln



Wykres z linią kwantylową



Dystrybuanta empiryczna
w porównaniu z rozkładem
normalnym



Normalność

```

par(mfrow=c(1, 1))

ks.test(x = mod3$residuals, y = "pnorm", mean = 0, sd = sd(mod3$residuals))

##
## One-sample Kolmogorov-Smirnov test
##
## data: mod3$residuals
## D = 0.051812, p-value = 0.135
## alternative hypothesis: two-sided

lillie.test(mod3$residuals)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: mod3$residuals
## D = 0.051812, p-value = 0.002624

```

```
shapiro.test(mod3$residuals)
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  mod3$residuals  
## W = 0.98969, p-value = 0.001356
```

```
ad.test(mod3$residuals)
```

```
##  
##  Anderson-Darling normality test  
##  
## data:  mod3$residuals  
## A = 1.3485, p-value = 0.001691
```

Z testów Kołmogorowa-Lillieforsa, Shapiro-Wilka oraz Andersona-Darlinga wynika, że musimy odrzucić hipotezę o normalności rozkładu reszt (dla testu Kołmogorowa-Smirnova nie było podstaw do odrzucenia, p-value około 0.14). Jeżeli chodzi o wykresy, to brak normalności najbardziej widać na wykresie gęstości. Na drugim wykresie (z linią kwantylową) reszty najbardziej odstają od rozkładu normalnego na początku i na końcu. Dystrybucja empiryczna jest zbliżona do dystrybucji rozkładu normalnego.

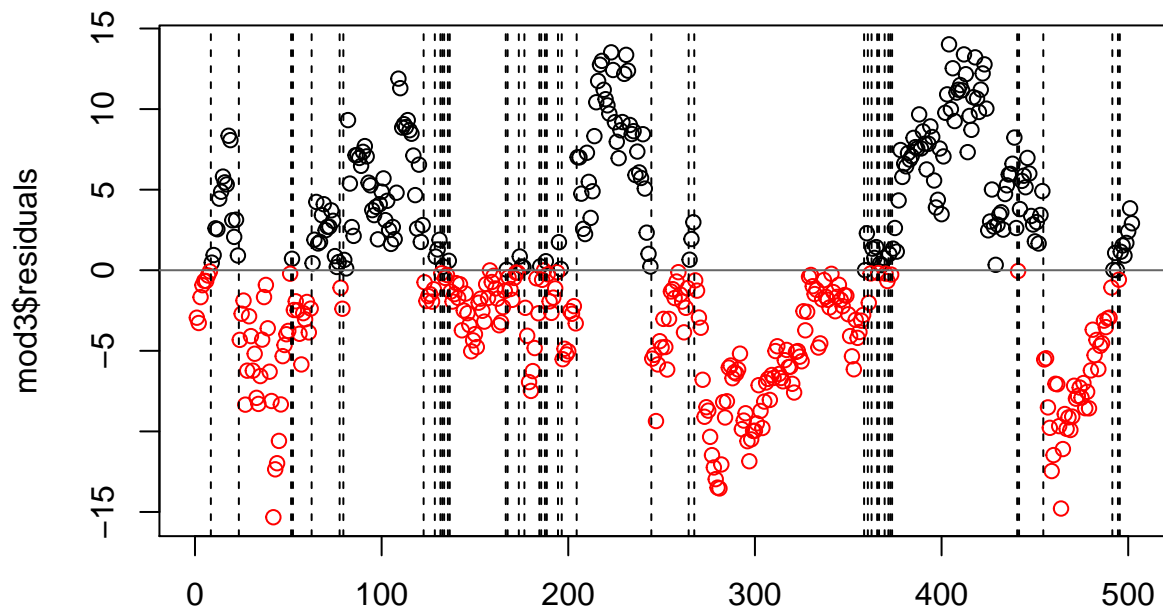
Autokorelacja

- Test serii

H_0 : losowość

H_1 : brak losowości

```
runs.test(mod3$residuals, threshold = 0, plot = T)
```

```
##
## Runs Test
##
## data: mod3$residuals
## statistic = -18.556, runs = 44, n1 = 229, n2 = 273, n = 502, p-value <
## 2.2e-16
## alternative hypothesis: nonrandomness
```

P-value bliskie zero, odrzucamy hipotezę o losowości reszt.

- Test Durbina-Watsona

Weryfikuje hipotezę o niezależności reszt, sprawdzając, czy istotna jest autokorelacja reszt rzędu pierwszego.

```
dwtest(mod3, order.by = ~t)
```

```
##
## Durbin-Watson test
##
## data: mod3
## DW = 0.11243, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

- Test Breuscha-Godfrey

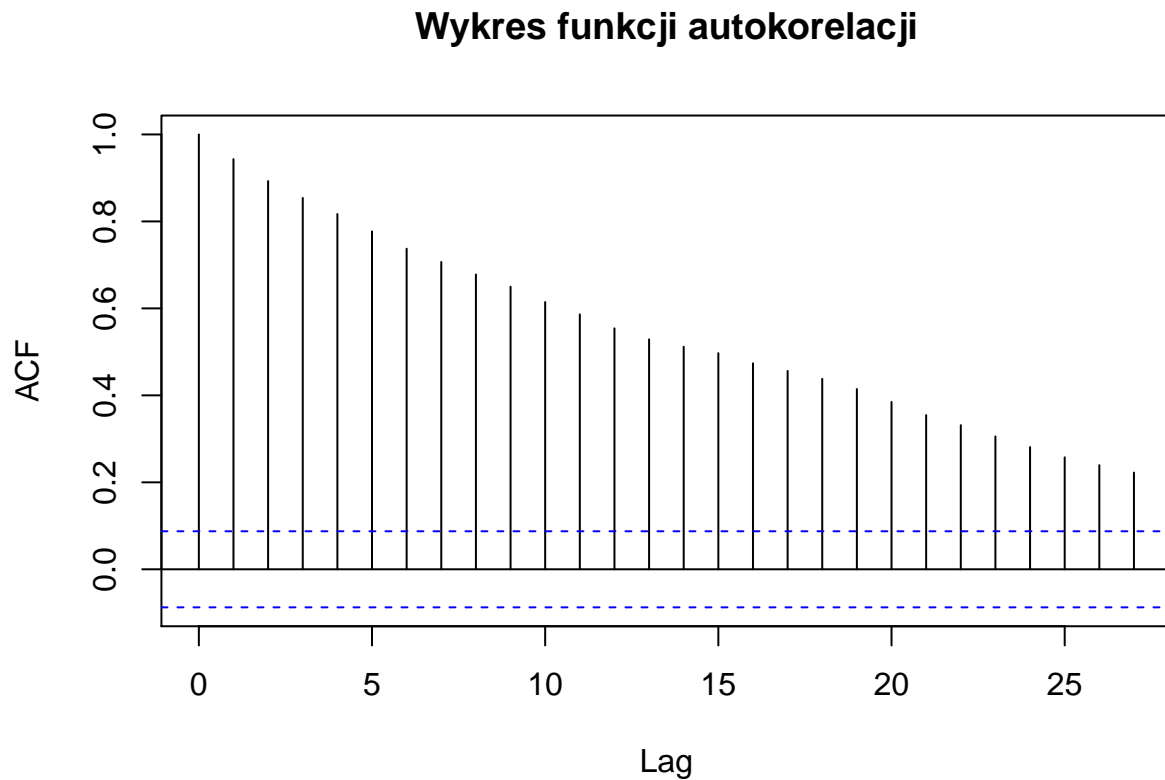
```
bgtest(mod3, order.by = ~t, order = 3)
```

```
##
## Breusch-Godfrey test for serial correlation of order up to 3
##
```

```
## data: mod3
## LM test = 447.34, df = 3, p-value < 2.2e-16
```

- Wykres ACF (funkcji autokorelacji)

```
acf(mod3$residuals, main = "Wykres funkcji autokorelacji")
```

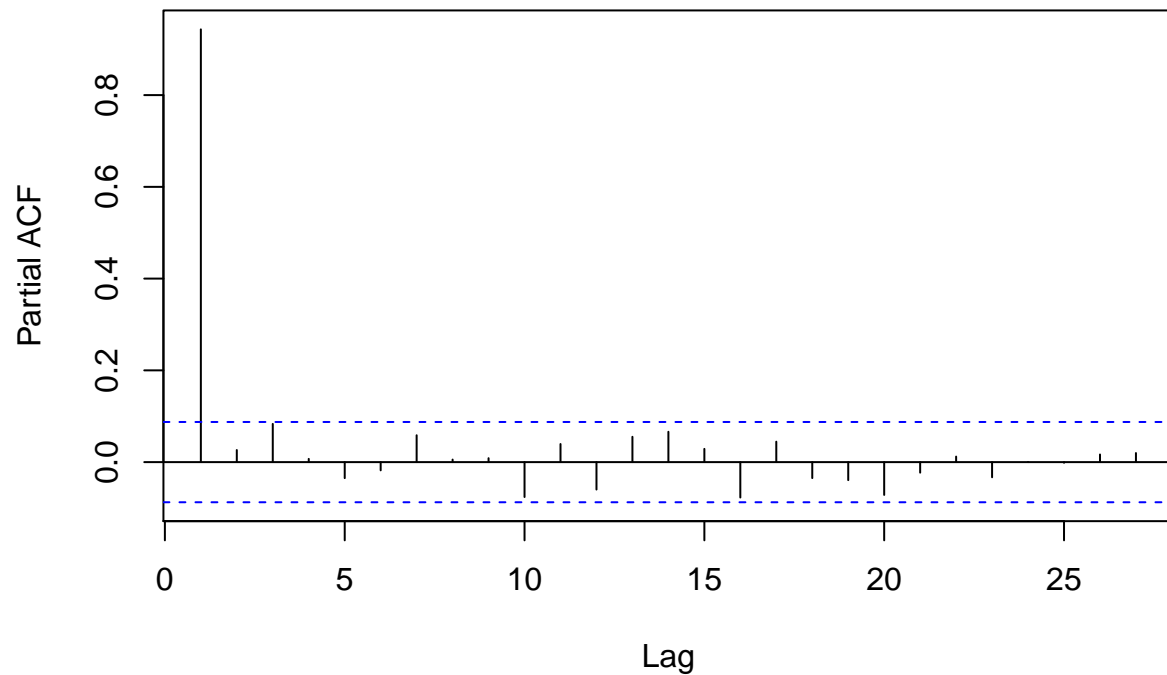


Oba testy wskazują, że istnieje seryjna korelacja błędów. Również z wykresu funkcji `acf` możemy wyciągnąć te same wnioski. Dla opóźnień do rzędu 25 obserwacje nie mieszczą się w niebieskich przerywanych liniach - wnioskujemy, że pojawia się autokorelacja.

- Wykres PACF (częstkowej funkcji autokorelacji)

```
pacf(mod3$residuals, main = "Częstkowa funkcja autokorelacji")
```

Czastkowa funkcja autokorelacji



Wykres PACF jako jedyny nie wykrywa autokorelacji.

- Test Ljunga-Boxa

H_0 : niezależność

H_1 : brak niezależności

```
Box.test(mod3$residuals, type = "Ljung-Box")
```

```
##  
## Box-Ljung test  
##  
## data: mod3$residuals  
## X-squared = 449.38, df = 1, p-value < 2.2e-16
```

P-value jest bardzo bliskie zero, odrzucamy hipotezę zerową, reszty nie są niezależne.

Ostatecznie stwierdzamy, że zachodzi autokorelacja reszt.

Stacjonarność

Zbadamy, czy szereg jest stacjonarny albo TS (trend stationary).

Skorzystamy z dwóch testów :

- adf (Dickey-Fullera)

H_0 : niestacjonarność

H_1 : stacjonarność

- kpss (Kwiatkowskiego-Phillipsa-Schmidta-Shina)

H_0 : stacjonarność

H_1 : niestacjonarność

```
adf.test(mcd)
```

```
##
## Augmented Dickey-Fuller Test
##
## data: mcd
## Dickey-Fuller = -2.2351, Lag order = 7, p-value = 0.4788
## alternative hypothesis: stationary
```

```
kpss.test(mcd)
```

```
##
## KPSS Test for Level Stationarity
##
## data: mcd
## KPSS Level = 7.0755, Truncation lag parameter = 5, p-value = 0.01
```

```
kpss.test(mcd, null = "Trend")
```

```
##
## KPSS Test for Trend Stationarity
##
## data: mcd
## KPSS Trend = 0.66563, Truncation lag parameter = 5, p-value = 0.01
```

Po wykonaniu testów otrzymujemy wniosek, że szereg nie jest stacjonarny, ani stacjonarny wobec trendu.

```
adf.test(diff(mcd, differences = 1)) #stacj
```

```
##
## Augmented Dickey-Fuller Test
##
## data: diff(mcd, differences = 1)
## Dickey-Fuller = -8.574, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

```
kpss.test(diff(mcd, differences = 1)) #stacj
```

```
##
## KPSS Test for Level Stationarity
##
## data: diff(mcd, differences = 1)
## KPSS Level = 0.12182, Truncation lag parameter = 5, p-value = 0.1
```

```
kpss.test(diff(mcd, differences = 1), null = "Trend") #stacj
```

```
##
## KPSS Test for Trend Stationarity
##
## data: diff(mcd, differences = 1)
## KPSS Trend = 0.1159, Truncation lag parameter = 5, p-value = 0.1
```

Po zróżnicowaniu 1 raz, szereg jest zarówno stacjonarny, jak i TS.

```
n <- ndiffs(mcd)
```

Liczba różnicowań uzyskana za pomocą funkcji `ndiffs` wynosi 1, co zgadza się z wcześniejszymi wnioskami.

ARIMA

Nasz szereg jest niestacjonarny, więc spróbujemy do niego dopasować model ARIMA.

```
auto.arima(mcd)
```

```
## Series: mcd
## ARIMA(0,1,0)
##
## sigma^2 estimated as 4.176:  log likelihood=-1068.94
## AIC=2139.87   AICc=2139.88   BIC=2144.09
```

Według funkcji `auto.arima`, najlepszy model dla badanego szeregu czasowego to ARIMA(0, 1, 0), co oznacza, że nie ma składnika ani AR (Auto-Regressive), ani MA (Moving Average), a jedynie należy ten szereg zróżnicować jeden raz.

```
mcd_arima <- arima(mcd, c(0,1,0))
```

Kryterium AKAIKE dla modelu ARIMA(0,1,0) wynosi 2139.8741327, a wariancja 4.1759287

Na wszelki wypadek sprawdzimy dopasowanie kilku innych modeli ARIMA(p, r, q). Przyjmujemy $r = 1$, jako że wcześniej otrzymaliśmy, że szereg należy zróżnicować jeden raz. Parametry p i q będą się zmieniać w pętlach od 0 do 3. Chcemy znaleźć najlepszy model ze względu na kryterium Akaike (tj. szukamy jak najmniejszego `aic`).

```
akaike <- NULL
for (p in 0:3){
  akaike1 <- NULL
  for (q in 0:3){
    akaike1 <- c(akaike1, arima(mcd, c(p,1,q))$aic)
  }
  akaike <- rbind(akaike, akaike1)
}
akaike <- as.data.frame(akaike)
colnames(akaike) <- c("q=0", "q=1", "q=2", "q=3")
rownames(akaike) <- c("p=0", "p=1", "p=2", "p=3")

kable(akaike) %>%
  kable_styling(full_width = T)
```

	q=0	q=1	q=2	q=3
p=0	2139.874	2140.328	2138.043	2140.027
p=1	2140.623	2139.634	2140.032	2141.989
p=2	2137.953	2139.850	2140.285	2141.628
p=3	2139.795	2141.713	2141.873	2141.595

Z powyższej tabeli widać, że faktycznie najlepszym modelem jest ARIMA(0, 1, 0).

W takim razie nasz szereg można by zapisać jako:

$$\varepsilon_t = \epsilon_t$$

gdzie $\epsilon_t \sim N(0, 4.176)$.

Holt - Winters

ZA CHOLERĘ NIE WIEM JAK TYM ZROBIĆ PREDYKCJĘ BEZ SEZONOWOŚCI. Jak ktoś coś wie to niech robi. Pozdro.

Podsumowanie analizy notowań McDonald's

Zrobiliśmy analizę. Nie umiemy jej zinterpretować. Jest fajnie.

Starbucks

Opis firmy

Starbucks Corporation – największa na świecie sieć kawiarni. Została założona 30 marca 1971 w Seattle w stanie Waszyngton.



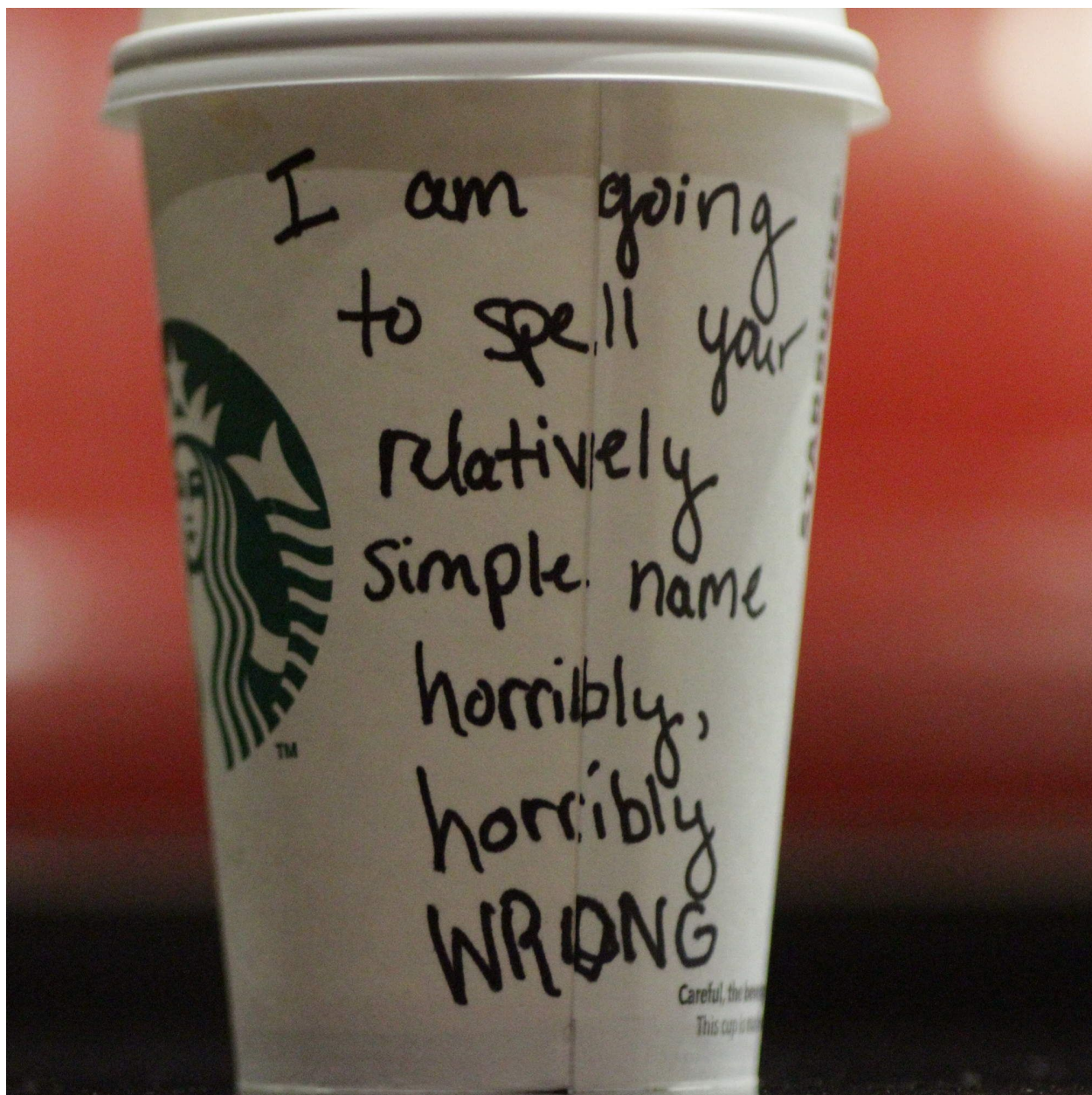
Rysunek 1: Kawa ze Starbucks

Kawa ze Starbucks jest znana z tego, że pojawiła się jako błąd w jednym z odcinków Gry o tron.

Kawiarnia jest znana z tego, że każdy kubek jest podpisany imieniem zamawiającego. Urocze!



Rysunek 2: Starbucks jest tak popularny, że piją go nawet w fantasy, które dzieje się w średniowieczu!



Rysunek 3: hehe



Podobno kawa w Starbucksie jest droższa niż paliwo w Polsce!!!!

Wczytanie danych

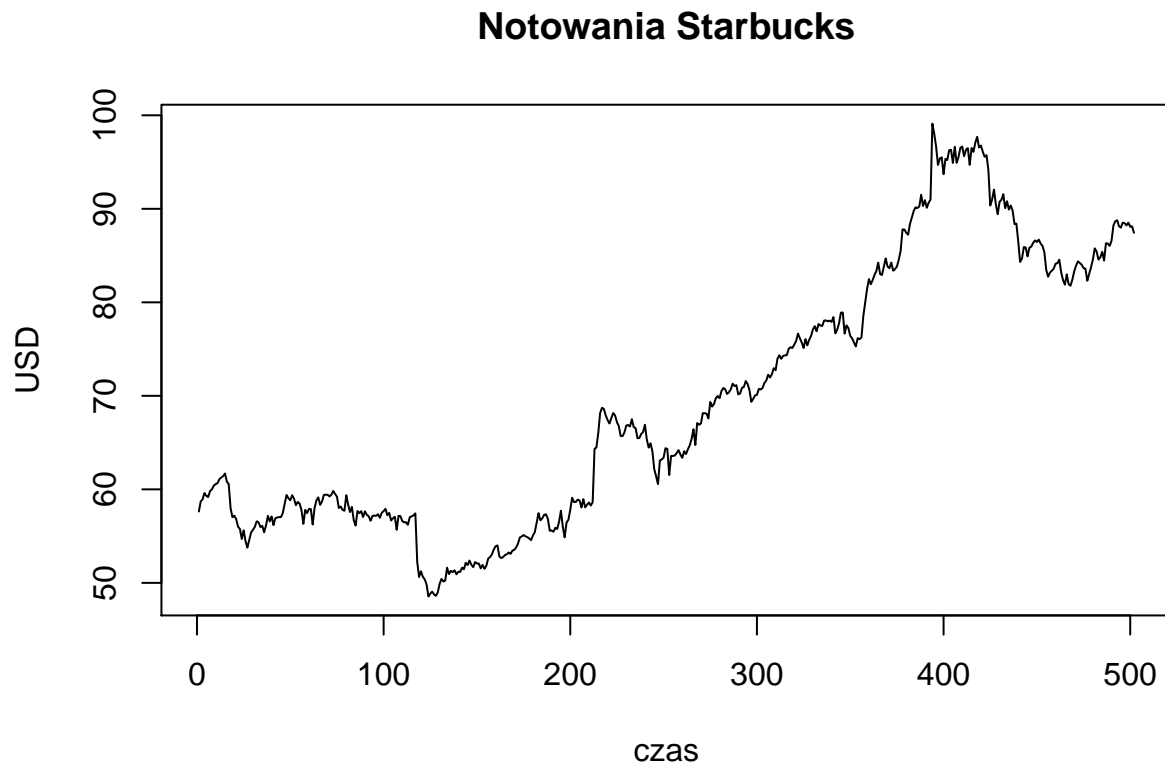
```
sbux<- get.hist.quote(instrument = "SBUX", provider = "yahoo",  
                      quote = "Close", start = "2018-01-01", end = "2019-12-31")
```

```
## time series starts 2018-01-02  
## time series ends   2019-12-30
```

```
sbux<- as.numeric(sbux)
```

Rysunek

```
plot(sbux, type = "l", xlab = "czas", ylab = "USD", main = "Notowania Starbucks")
```



Widać trend rosnący.

Dopasowanie wielomianu

```
t <- 1:length(sbux)
```

```
mod1 <- lm(sbux~t)  
summary(mod1)
```

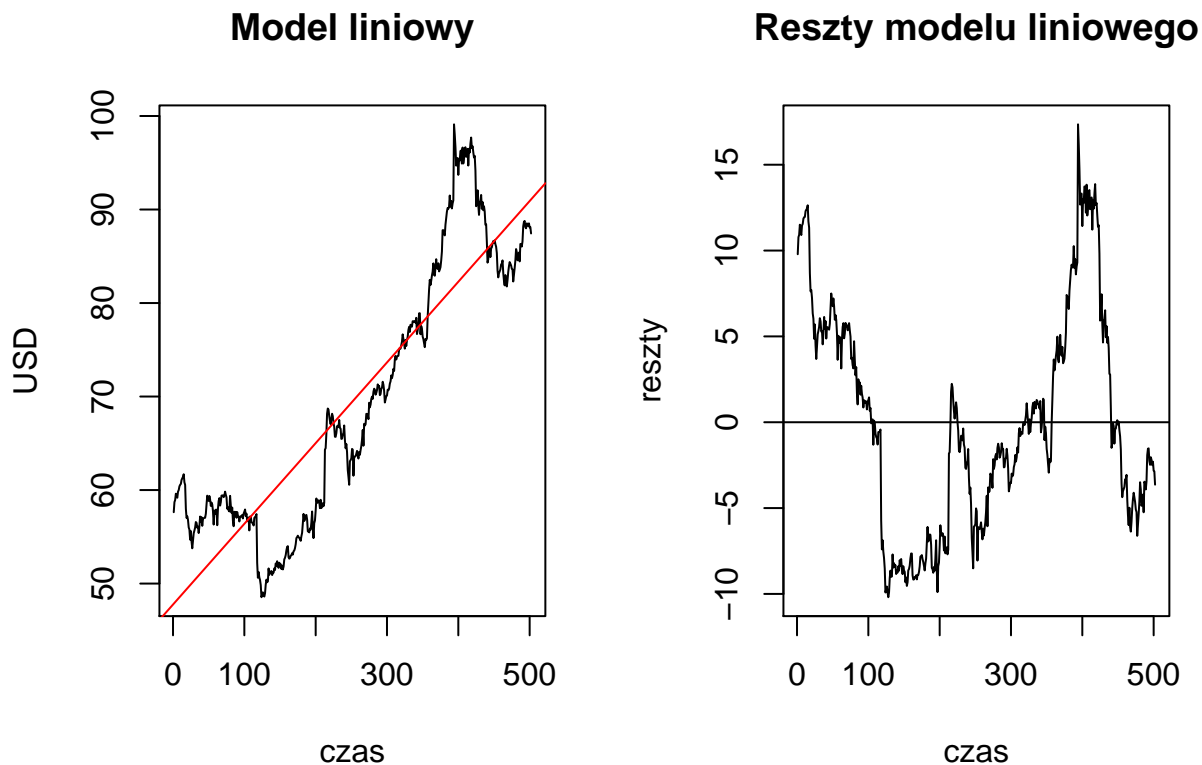
Model liniowy

```
##  
## Call:  
## lm(formula = sbux ~ t)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -10.1924  -5.1586  -0.6162   4.8674  17.3536   
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 47.756892   0.569068  83.92   <2e-16 ***
## t           0.086293   0.001961  44.02   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.366 on 500 degrees of freedom
## Multiple R-squared:  0.7949, Adjusted R-squared:  0.7944
## F-statistic: 1937 on 1 and 500 DF,  p-value: < 2.2e-16
```

Wszystkie współczynniki są istotne statystycznie. R^2 wynosi około 79%.

```
par(mfrow = c(1, 2))
plot(sbox, type = "l", main = "Model liniowy", xlab = "czas", ylab = "USD")
abline(mod1, col = "red")
plot(mod1$residuals, type = "l", main = "Reszty modelu liniowego", xlab = "czas", ylab = "reszty")
abline(h=0)
```



```
par(mfrow = c(1, 1))
```

Na wykresie widać, że reszty mają rozrzut mniej więcej od -10 do 15. Model nie jest zbyt dokładny - na początku przeszacowuje wartości, potem zdecydowanie niedoszacowuje, na koniec znowu przeszacowuje.

```
mod2 <- lm(sbox~t+I(t^2))
summary(mod2)
```

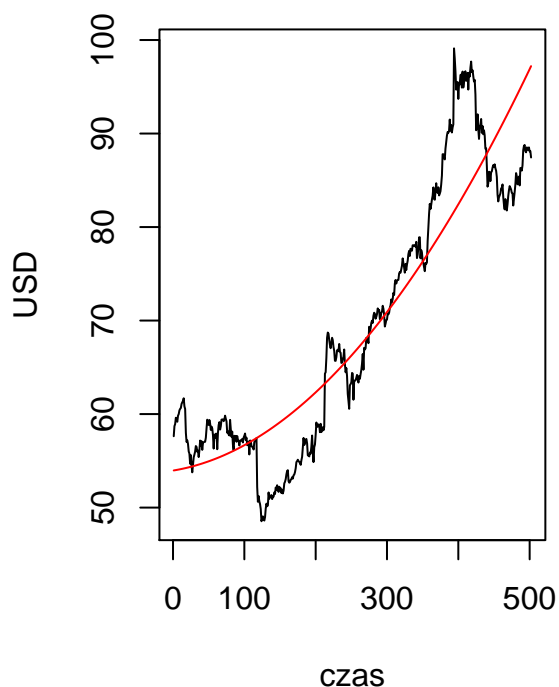
Drugiego stopnia

```
##
## Call:
## lm(formula = sbux ~ t + I(t^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.0010  -4.7965   0.7868   3.2228  17.4555
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.396e+01  7.717e-01  69.924  <2e-16 ***
## t           1.245e-02  7.085e-03   1.757   0.0795 .
## I(t^2)      1.468e-04  1.364e-05  10.763  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.74 on 499 degrees of freedom
## Multiple R-squared:  0.8335, Adjusted R-squared:  0.8328
## F-statistic: 1249 on 2 and 499 DF,  p-value: < 2.2e-16
```

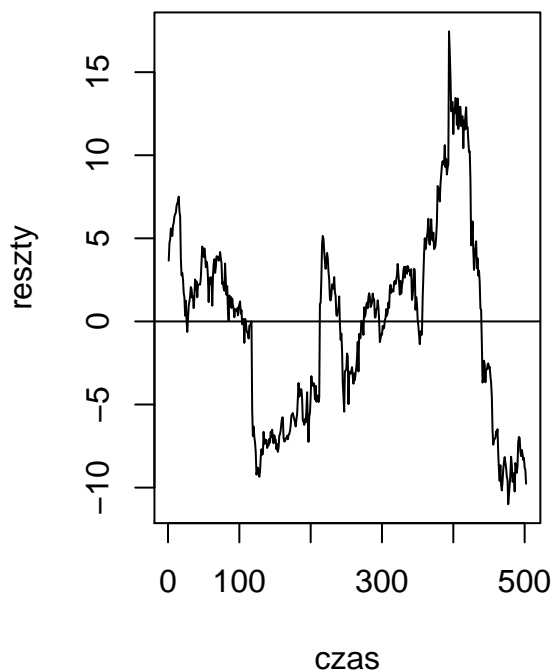
Współczynnik przy t jest nieistotny statystycznie (p-value około 0.08, więc decyzja niejednoznaczna), ale R^2 poprawiło się - wynosi teraz około 83%.

```
par(mfrow = c(1, 2))
plot(sbx, type = "l", main = "Model kwadratowy", xlab = "czas", ylab = "USD")
lines(t, mod2$fitted.values, col = "red")
plot(mod2$residuals, type = "l", main = "Reszty modelu kwadratowego", xlab = "czas", ylab = "reszty")
abline(h = 0)
```

Model kwadratowy



Reszty modelu kwadratowego



```
par(mfrow = c(1, 1))
```

Wykres reszt jest bardzo podobny jak w przypadku modelu liniowego.

```
mod3 <- lm(sbox~t+I(t^2)+I(t^3))
summary(mod3)
```

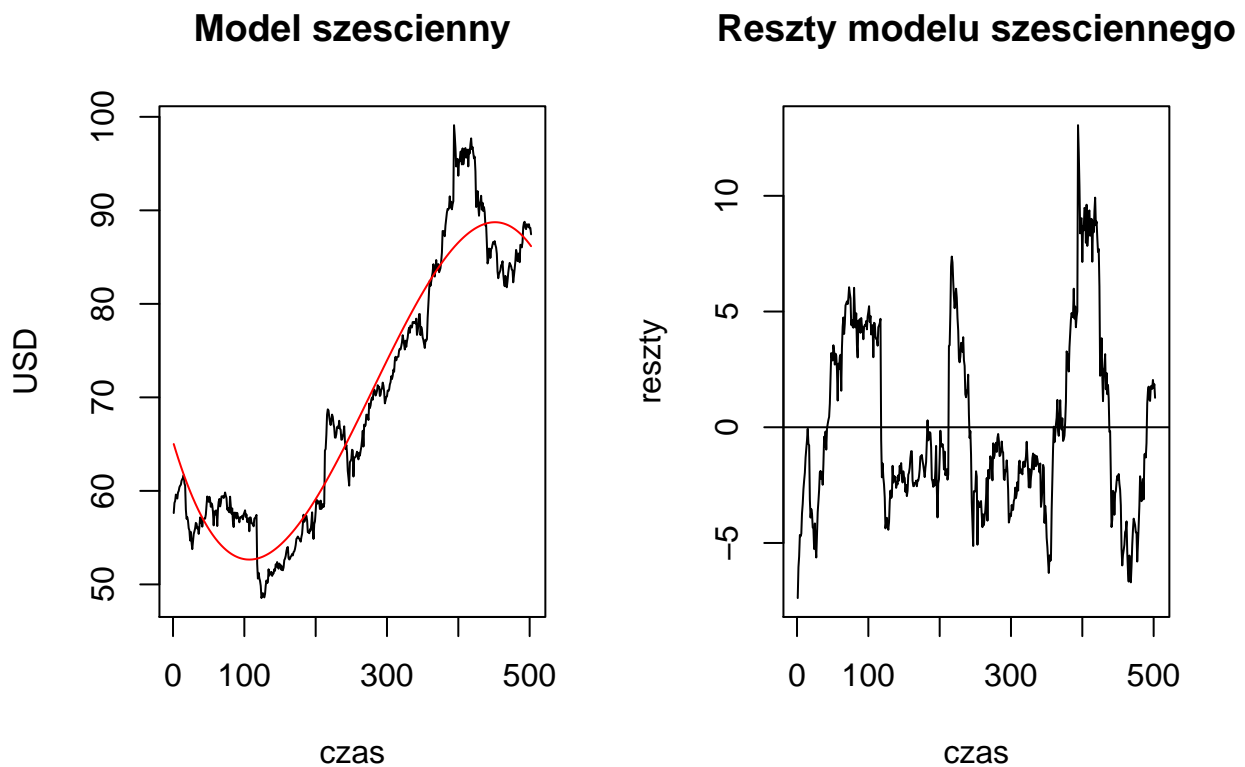
Trzeciego stopnia

```
##
## Call:
## lm(formula = sbox ~ t + I(t^2) + I(t^3))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.380 -2.608 -1.257   3.041 13.053
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.526e+01  6.978e-01   93.53  <2e-16 ***
## t           -2.559e-01  1.200e-02  -21.32  <2e-16 ***
## I(t^2)       1.479e-03  5.542e-05   26.69  <2e-16 ***
## I(t^3)      -1.766e-06  7.243e-08  -24.38  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.879 on 498 degrees of freedom
## Multiple R-squared:  0.9241, Adjusted R-squared:  0.9237
## F-statistic: 2021 on 3 and 498 DF,  p-value: < 2.2e-16
```

Wszystkie współczynniki są istotne statystycznie, a R^2 znów wzrosło - wynosi około 92% (znaczną poprawę).

```
par(mfrow = c(1, 2))
plot(sbox, type = "l", main = "Model sześcienny", xlab = "czas", ylab = "USD")
lines(t, mod3$fitted.values, col = "red")
plot(mod3$residuals, type = "l", main = "Reszty modelu sześciennego", xlab = "czas", ylab = "reszty")
abline(h= 0)
```



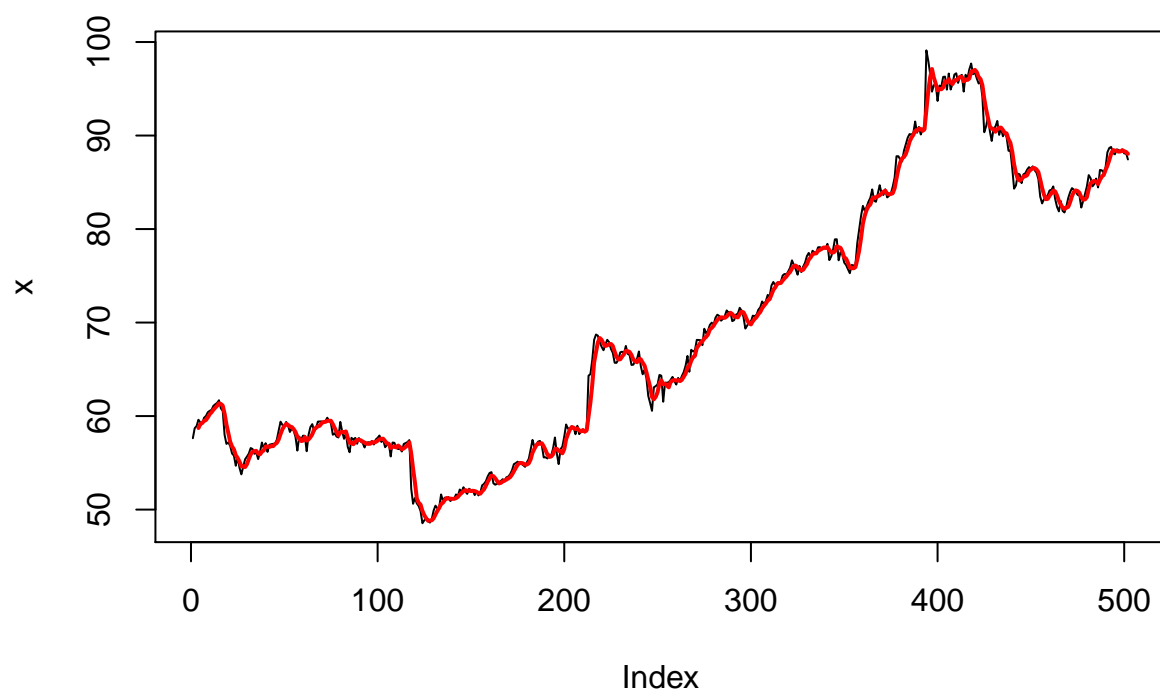
```
par(mfrow=c(1,1))
```

Reszty modelu sześciennego mają mniejszy rozrzut niż w poprzednich przypadkach (od około -5 do 10).

Ruchoma średnia

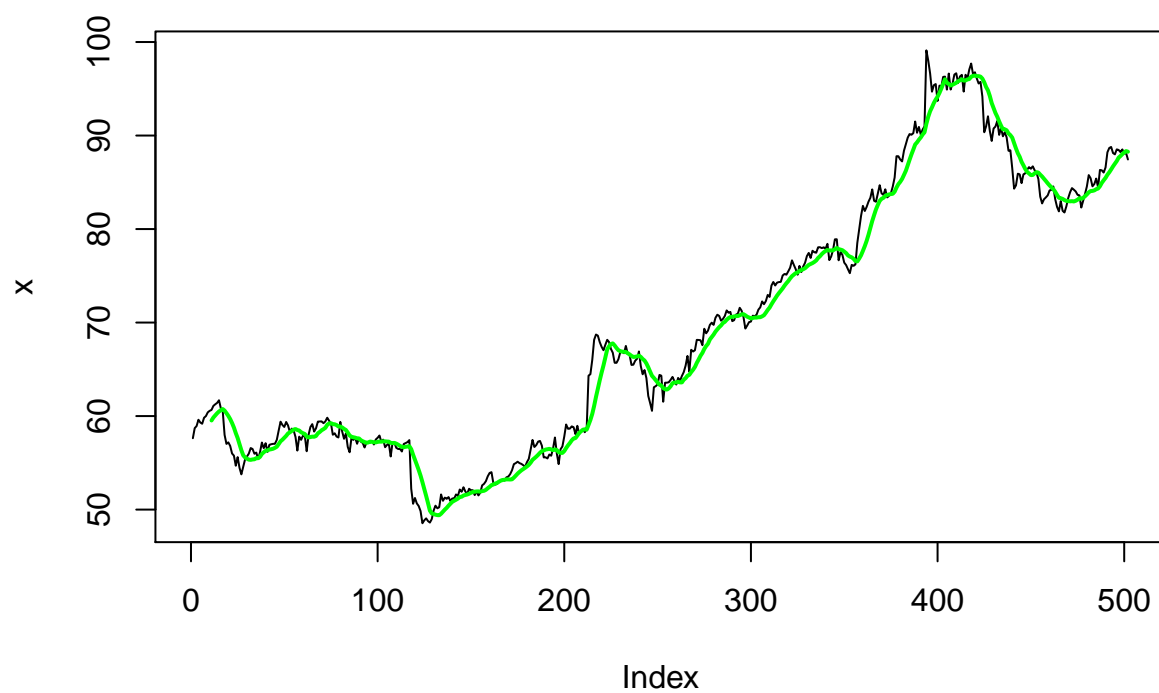
```
ruchoma(sbox, 3, "red")
```

Srednia ruchoma rzędu 3



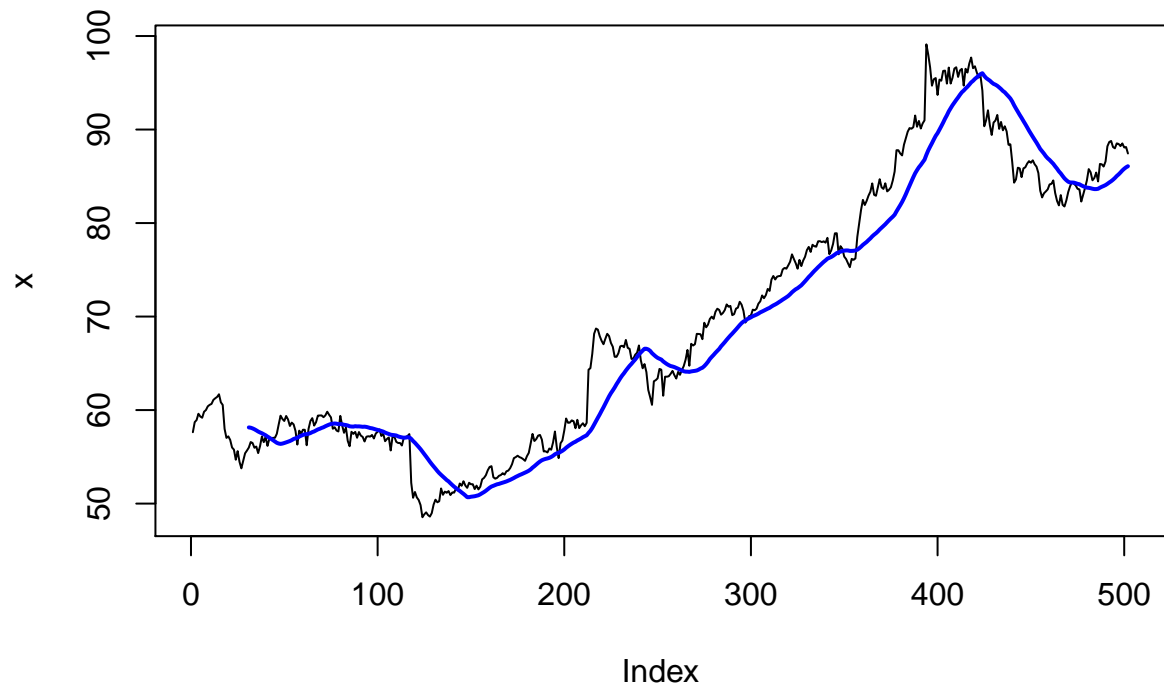
```
ruchoma(sbox, 10, "green")
```

Srednia ruchoma rzędu 10



```
ruchoma(sbox, 30, "blue")
```

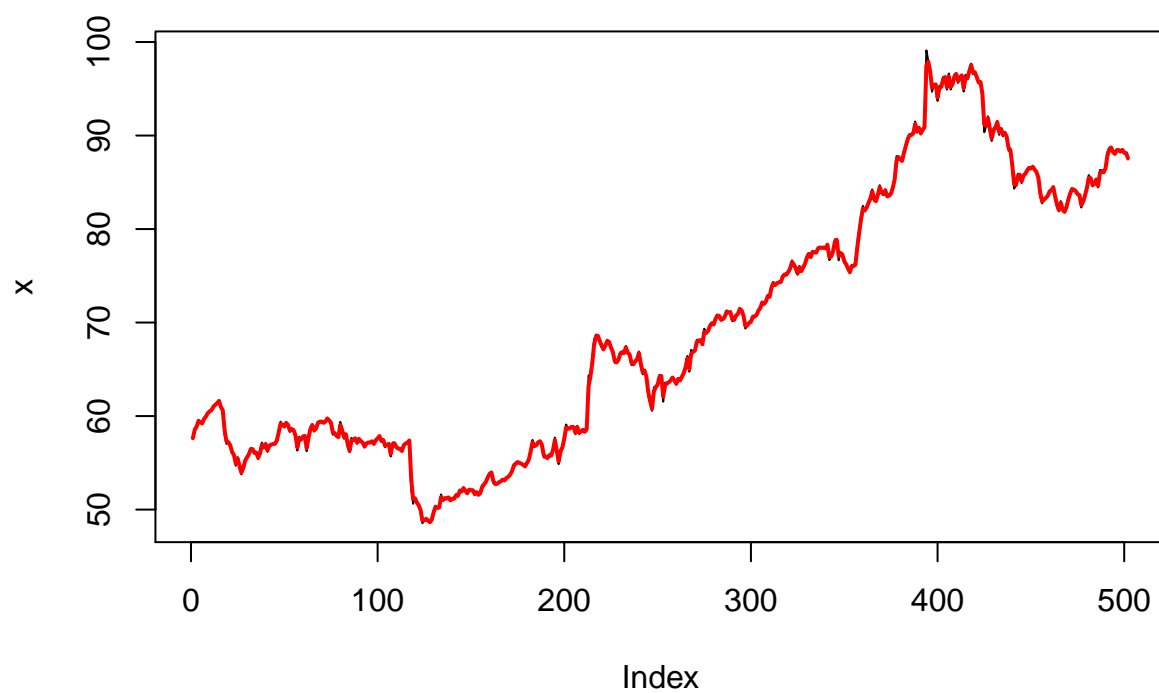

Srednia ruchoma rzędu 30



Metoda wykładniczych wag ruchomej średniej

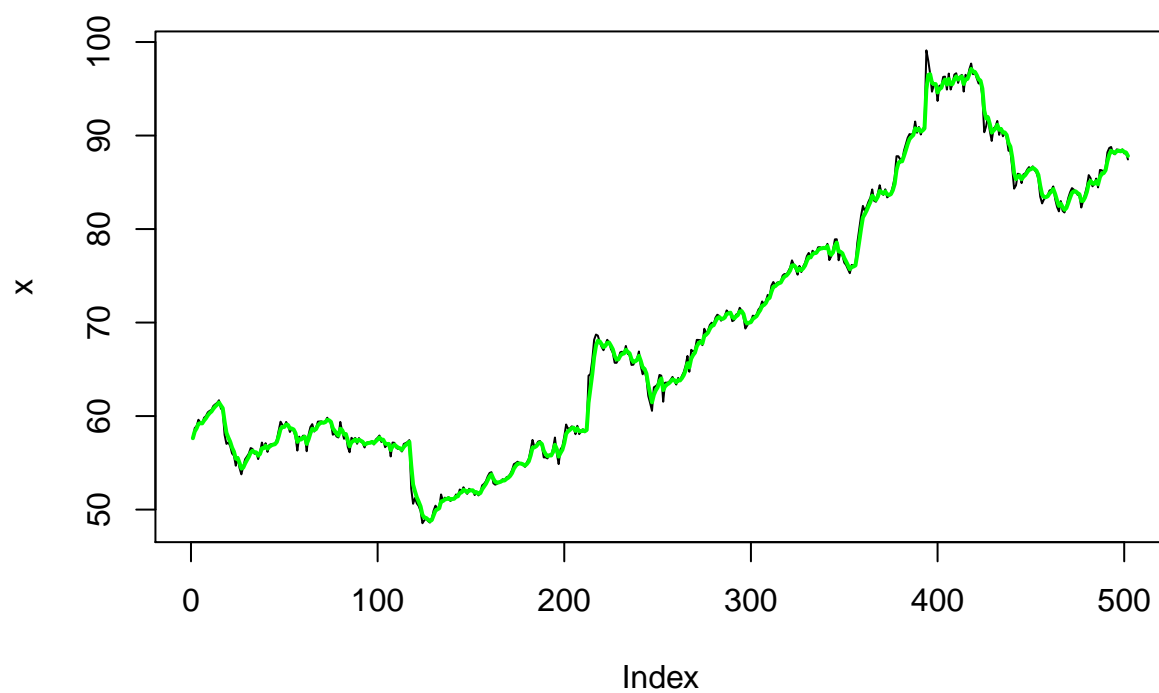
```
wykladnicza(sbox, 0.2, "red")
```

srednia ruchoma z wagami wykladniczymi z parametrem 0.2



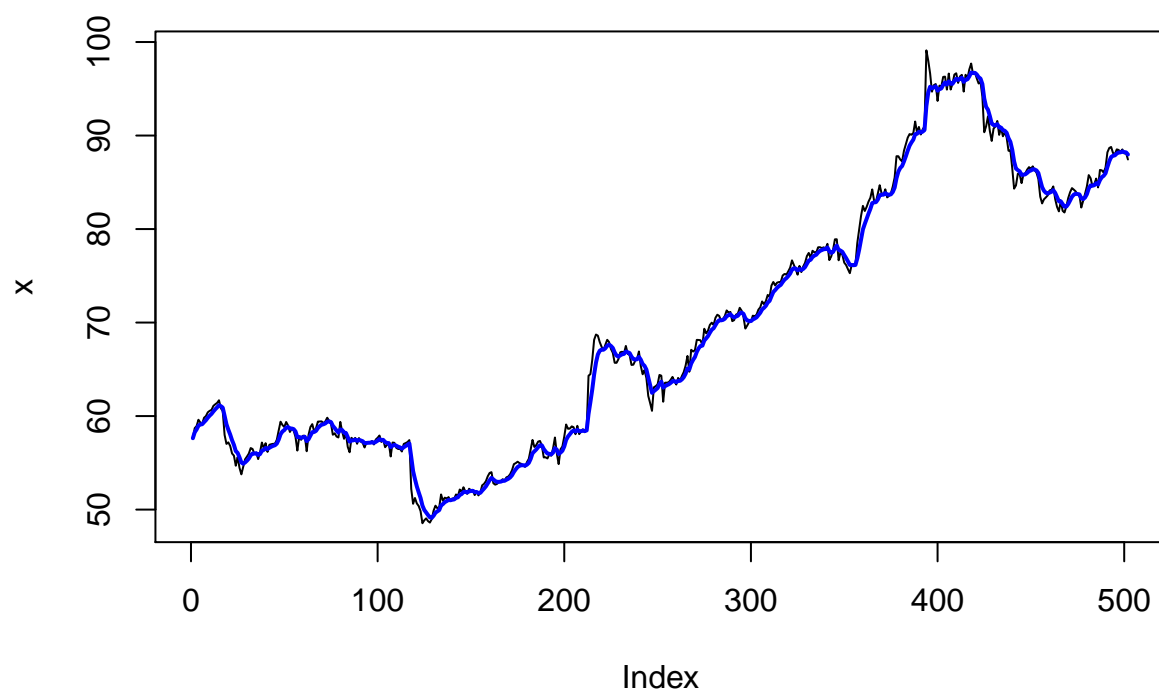
```
wykladnicza(sbox, 0.5, "green")
```

srednia ruchoma z wagami wykladniczymi z parametrem 0.5



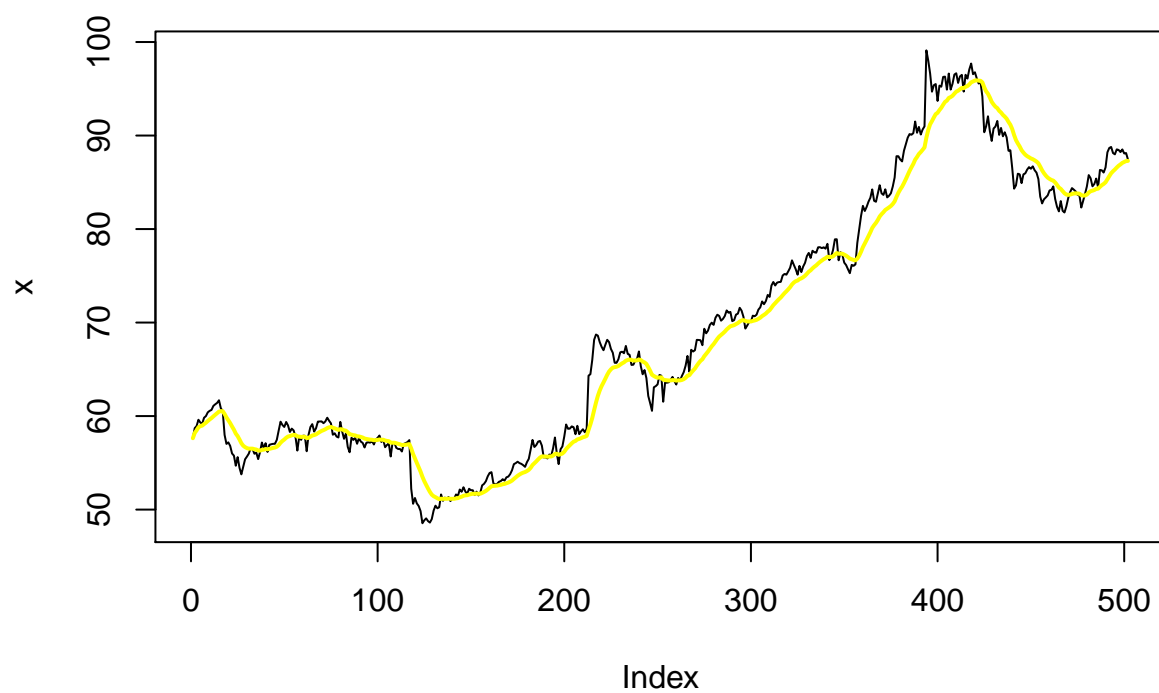
```
wykladnicza(sbox, 0.7, "blue")
```

srednia ruchoma z wagami wykladniczymi z parametrem 0.7



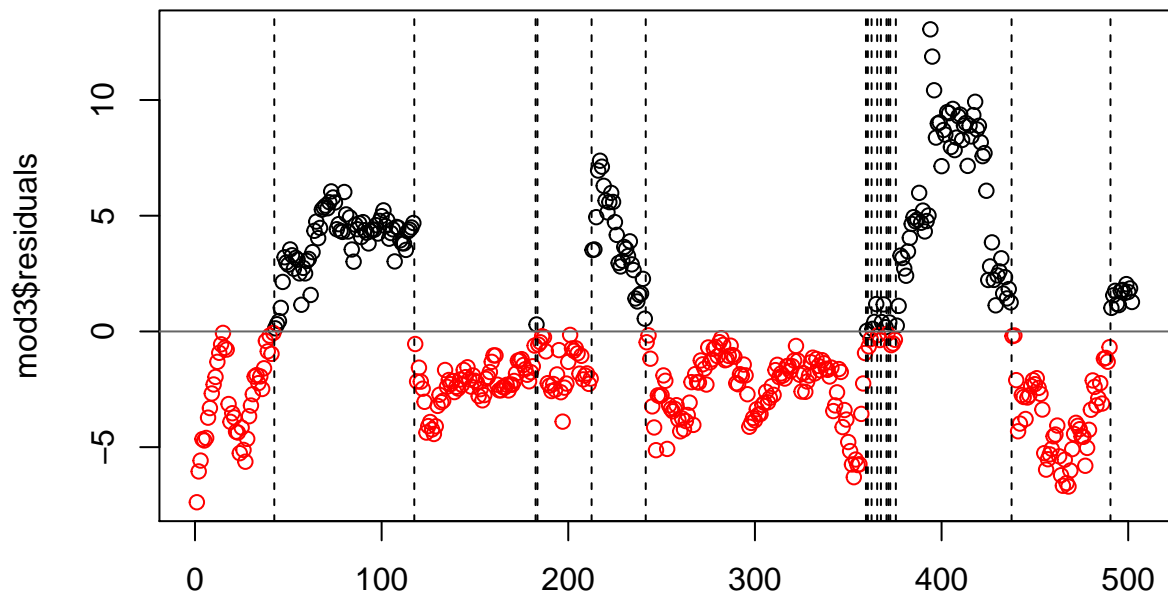
```
wykladnicza(sbox, 0.9, "yellow")
```

srednia ruchoma z wagami wykladniczymi z parametrem 0.9



Testy na resztach modelu sześciennego

```
runs.test(mod3$residuals, threshold = 0, plot = T)
```

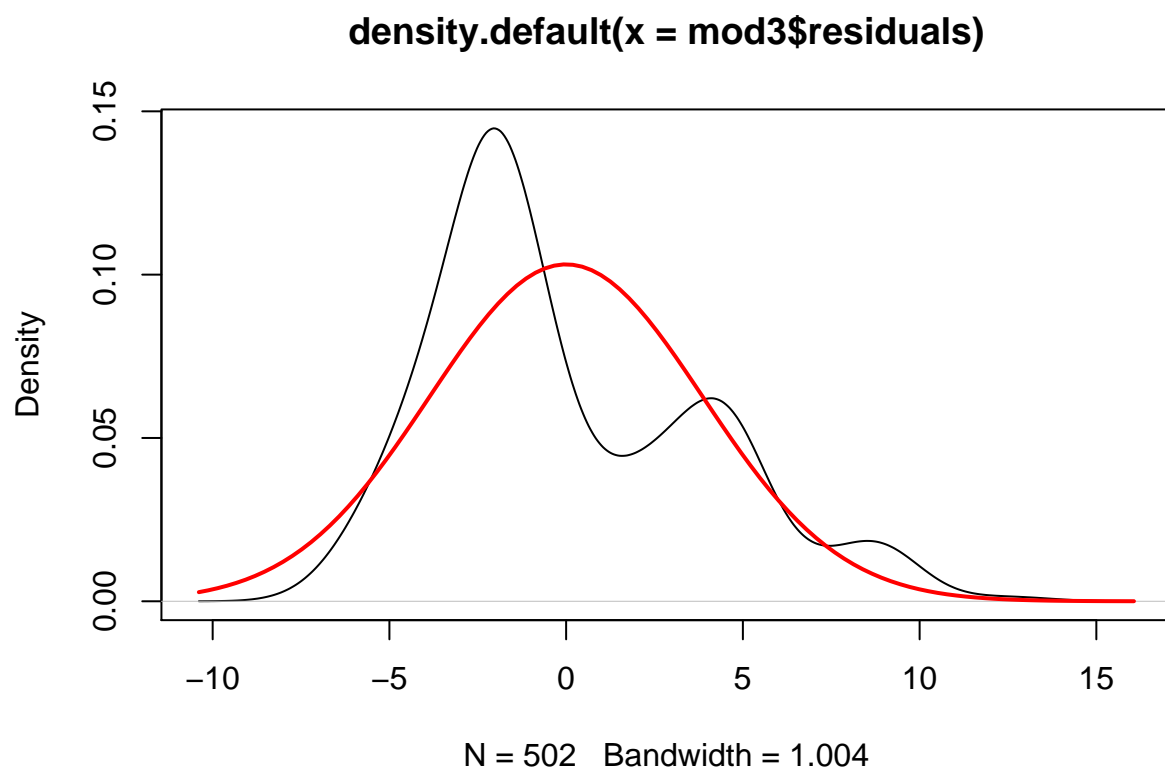


```
##
## Runs Test
##
## data:  mod3$residuals
## statistic = -20.806, runs = 18, n1 = 187, n2 = 315, n = 502, p-value <
## 2.2e-16
## alternative hypothesis: nonrandomness
```

P-value jest bliskie 0, odrzucamy hipotezę zerową o losowości reszt

Wykresy normalności

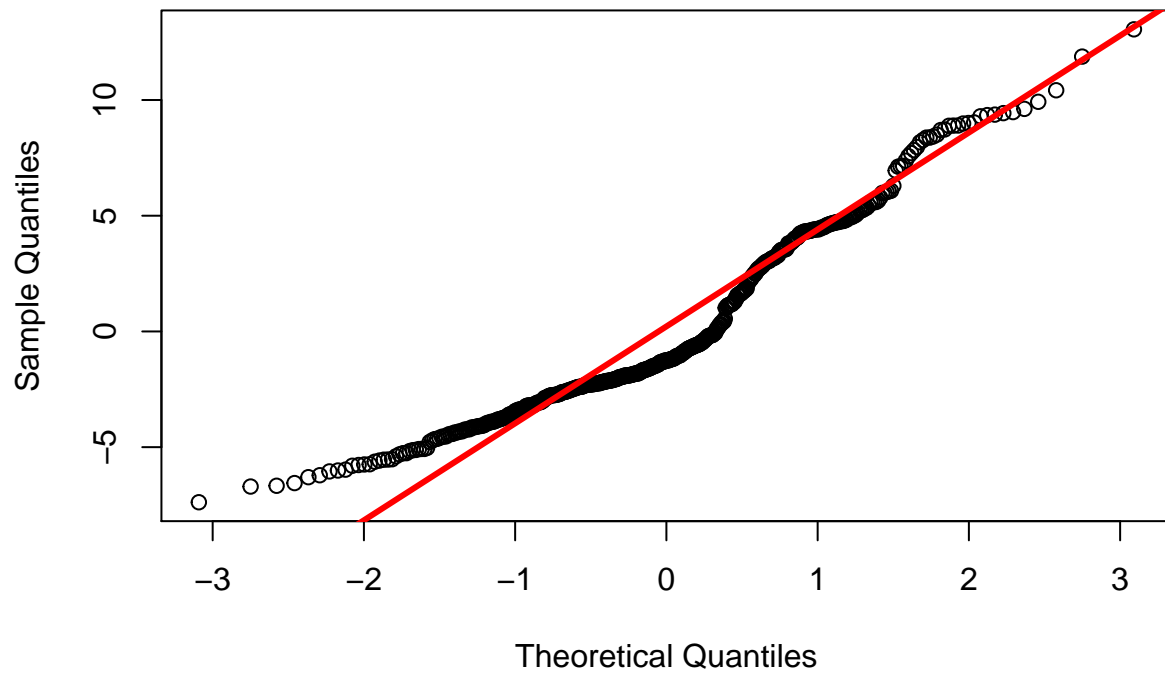
```
plot(density(mod3$residuals))
curve(dnorm(x, 0, sd(mod3$residuals)), add = T, col = 2, lwd = 2)
```



Wykres gęstości empirycznej znacząco różni się od gęstości rozkładu normalnego. Bardzo znacząco.

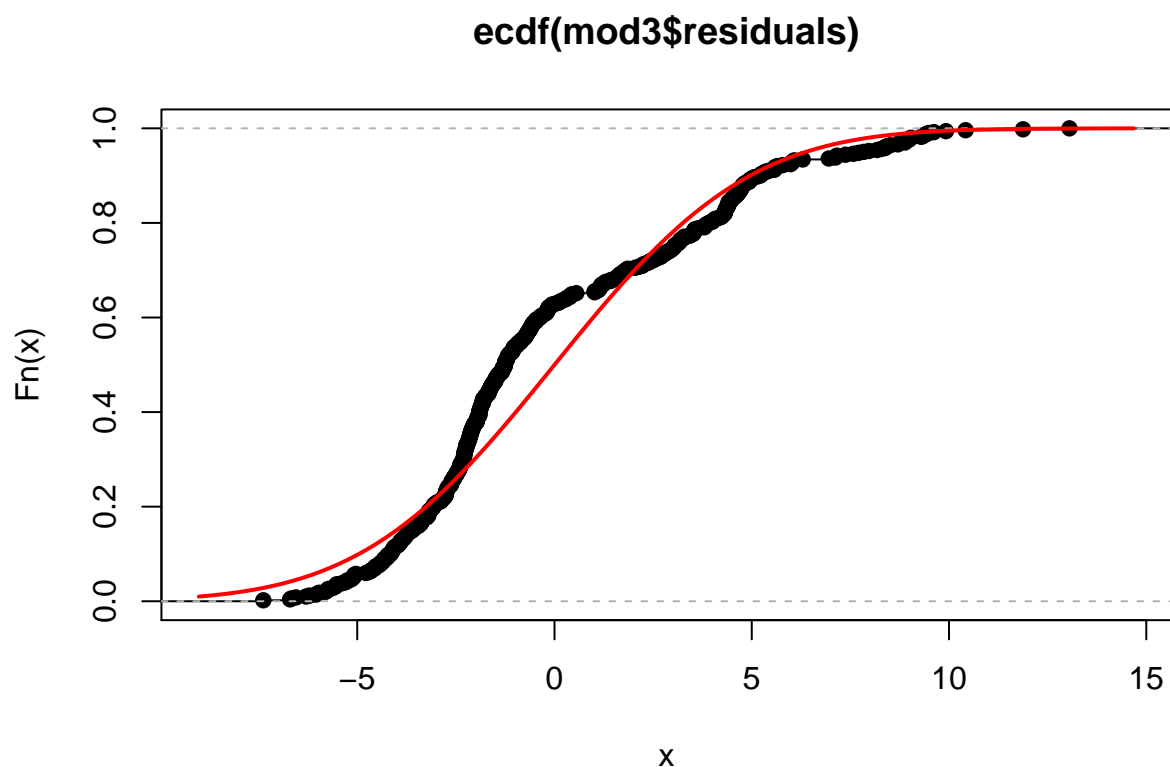
```
qqnorm(mod3$residuals)  
qqline(mod3$residuals, col=2, lwd = 3)
```

Normal Q-Q Plot



Tutaj także wyraźne odchyłki.

```
plot(ecdf(mod3$residuals))  
curve(pnorm(x, 0, sd(mod3$residuals)), add = T, col = 2, lwd = 2)
```

Nawet na dystrybuancie widać, że rozkład normalny wygląda inaczej.

```
ks.test(x = mod3$residuals, y = "pnorm", mean = 0, sd = sd(mod3$residuals))
```

```
##
## One-sample Kolmogorov-Smirnov test
##
## data: mod3$residuals
## D = 0.14346, p-value = 2.127e-09
## alternative hypothesis: two-sided
```

```
lillie.test(mod3$residuals)
```

```
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: mod3$residuals
## D = 0.14346, p-value < 2.2e-16
```

```
shapiro.test(mod3$residuals)
```

```
##
## Shapiro-Wilk normality test
##
## data: mod3$residuals
## W = 0.93719, p-value = 1.043e-13
```

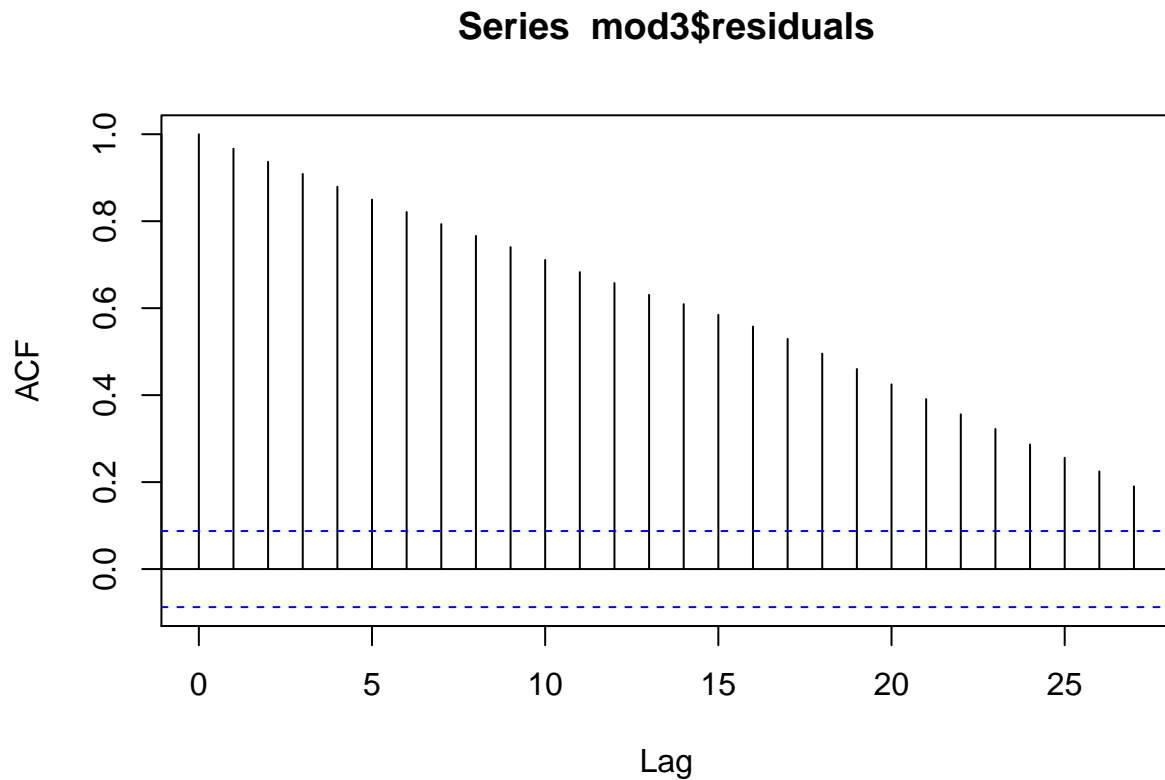
```
ad.test(mod3$residuals)
```

```
##
## Anderson-Darling normality test
##
## data: mod3$residuals
## A = 12.438, p-value < 2.2e-16
```

W każdym z testów p-value jest bardzo bliskie zero, stanowczo odrzucamy hipotezę o rozkładzie normalnym.

Badanie autokorelacji

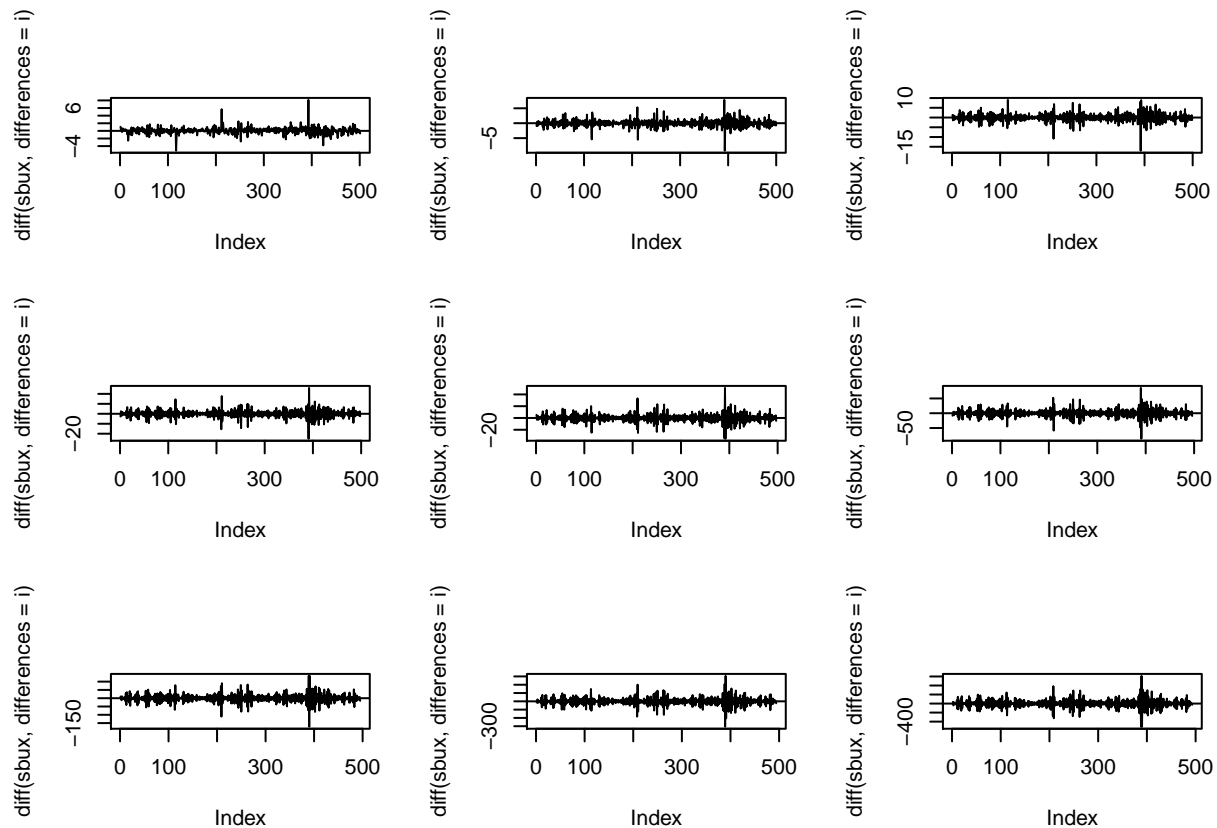
```
acf(mod3$residuals)
```



Słupki nie mieszczą się w niebieskim “pasku”, zatem prawdopodobnie ma miejsce autokorelacja.

Metoda różnicowa

```
par(mfrow = c(3, 3))
for(i in 1:9){
  plot(diff(sbox, differences = i), type = "l")
  abline(h = 0)}
```



```
par(mfrow=c(1,1))
```

Z wykresów widać, że największa stabilizacja jest przy różnicowaniu rzędu 2 lub 3, potem rozrzut zaczyna się znacząco zwiększać.

Stacjonarność

```
adf.test(sbux) #niest
```

```
##
## Augmented Dickey-Fuller Test
##
## data: sbux
## Dickey-Fuller = -2.1662, Lag order = 7, p-value = 0.5079
## alternative hypothesis: stationary
```

```
kpss.test(sbux) #niest
```

```
##
## KPSS Test for Level Stationarity
##
## data: sbux
## KPSS Level = 7.4101, Truncation lag parameter = 5, p-value = 0.01
```

```
kpss.test(sbux, null = "Trend") #niest
```

```
##
## KPSS Test for Trend Stationarity
```

```
##  
## data: sbux  
## KPSS Trend = 1.0249, Truncation lag parameter = 5, p-value = 0.01
```

Szereg nie jest ani stacjonarny ani TS.

```
adf.test(diff(sbox, differences = 1)) #st
```

```
##  
## Augmented Dickey-Fuller Test  
##  
## data: diff(sbox, differences = 1)  
## Dickey-Fuller = -7.8439, Lag order = 7, p-value = 0.01  
## alternative hypothesis: stationary
```

```
kpss.test(diff(sbox, differences = 1)) #st
```

```
##  
## KPSS Test for Level Stationarity  
##  
## data: diff(sbox, differences = 1)  
## KPSS Level = 0.15287, Truncation lag parameter = 5, p-value = 0.1
```

```
kpss.test(diff(sbox, differences = 1), null = "Trend") #st
```

```
##  
## KPSS Test for Trend Stationarity  
##  
## data: diff(sbox, differences = 1)  
## KPSS Trend = 0.11762, Truncation lag parameter = 5, p-value = 0.1
```

Po zróżnicowaniu rzędu 1 szereg jest zarówno stacjonarny jak i TS.

```
auto.arima(sbox)
```

```
## Series: sbux  
## ARIMA(0,1,0)  
##  
## sigma^2 estimated as 0.8884: log likelihood=-681.25  
## AIC=1364.51 AICc=1364.52 BIC=1368.73
```