

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

Alicja Hołowiecka, Matylda Jankowska, Marcin Dziadosz

23 12 2019

Spis treści

Wstęp	2
Wczytanie bibliotek	2
McDonald's	2
Opis firmy	2
Wczytanie danych i rysunki	2
Wydzielanie trendu	3
Dopasowanie wielomianu	6
Testy na resztach modelu	15
Stacjonarność	20
ARIMA	20
Prognoza	24
Podsumowanie analizy notowań McDonald's	26
Starbucks	26
Opis firmy	26
Wczytanie danych i rysunki	27
Wydzielanie trendu	27
Dopasowanie wielomianu	29
Testy na resztach modelu	35
Stacjonarność	40
ARIMA	41
Prognoza	43
Podsumowanie analizy notowań Starbucks	46
Podsumowanie i wnioski	46

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.

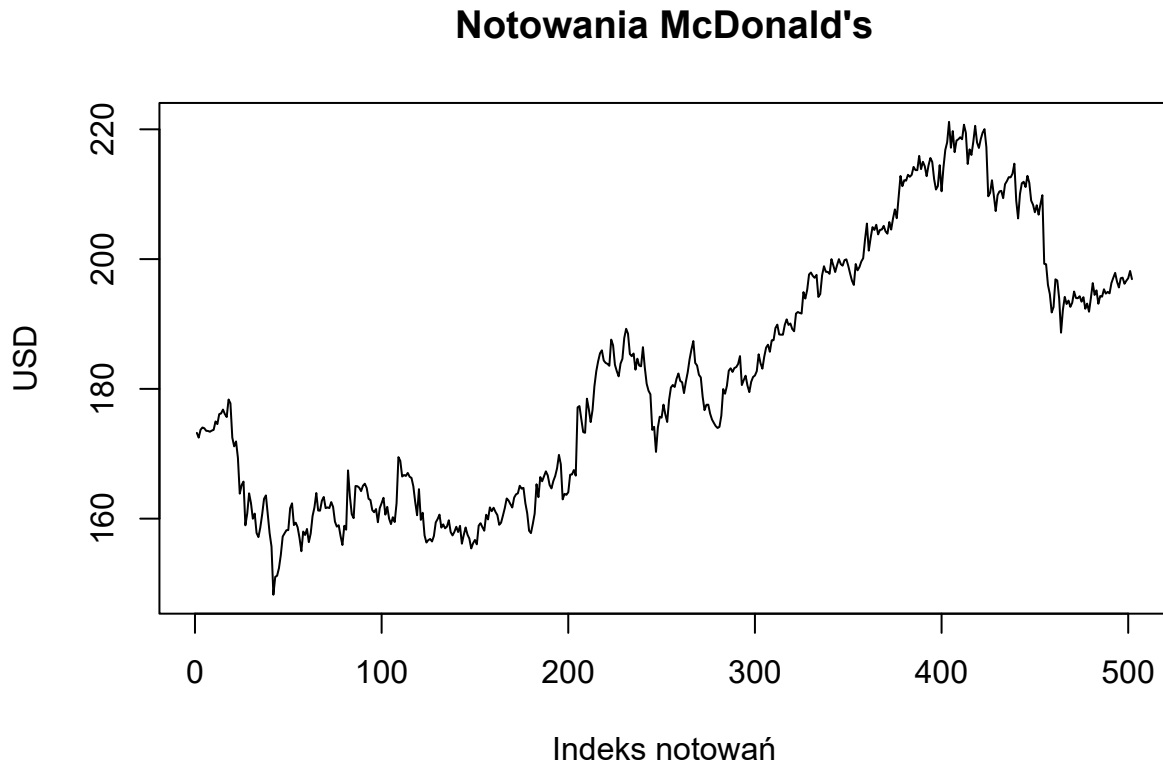
```
mcd<- get.hist.quote(instrument = "MCD", provider = "yahoo",
                    quote = "Close", start = "2018-01-01", end = "2019-12-31")
```

```
## 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 = "Indeks notowań", 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, xlab = "Indeks notowań", ylab = "USD")
  lines((m+1):t, f[(m+1):t], lwd = 2, col = kolor)
}

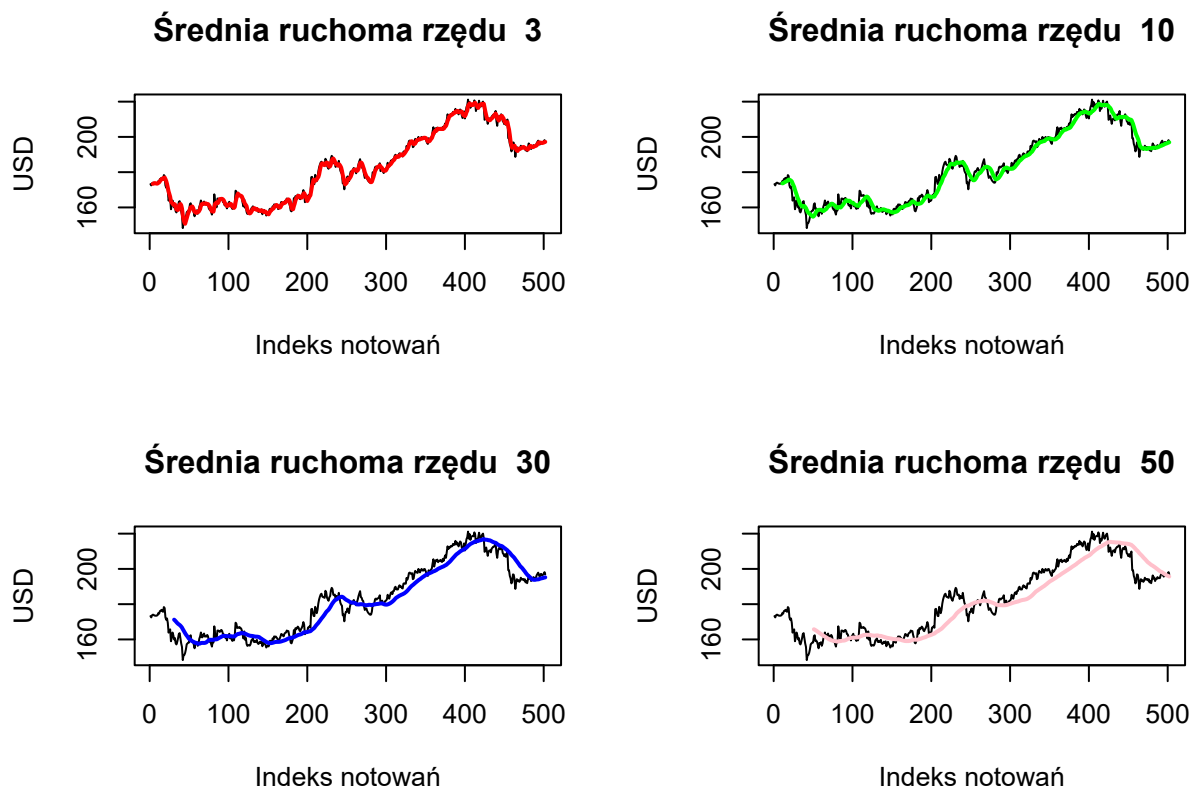
```

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")

```



```

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 $\eta \in (0, 1)$

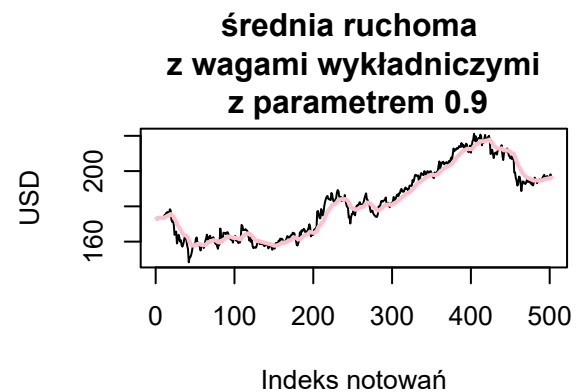
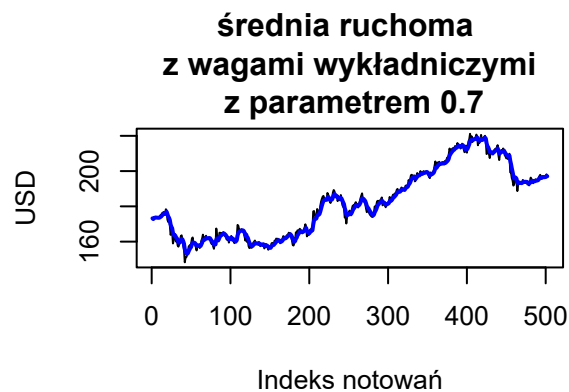
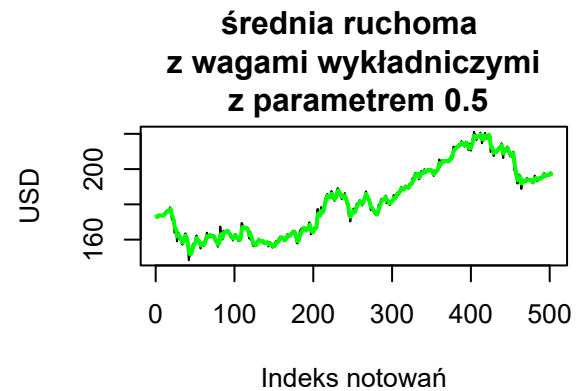
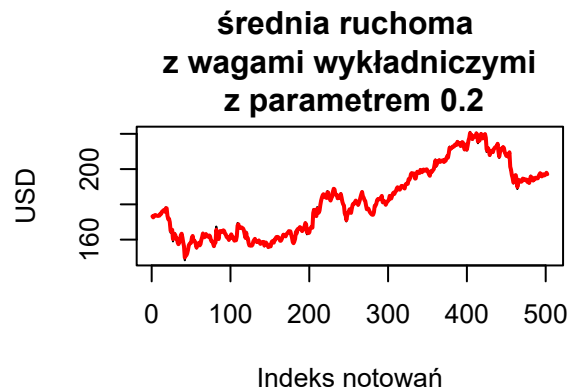
Skorzystamy z postaci rekurencyjnej:

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

```
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 \n z wagami wykładniczymi \n z parametrem", mi)
  plot(x, type = "l", main = tytul, xlab = "Indeks notowań", ylab = "USD")
  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")
```



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

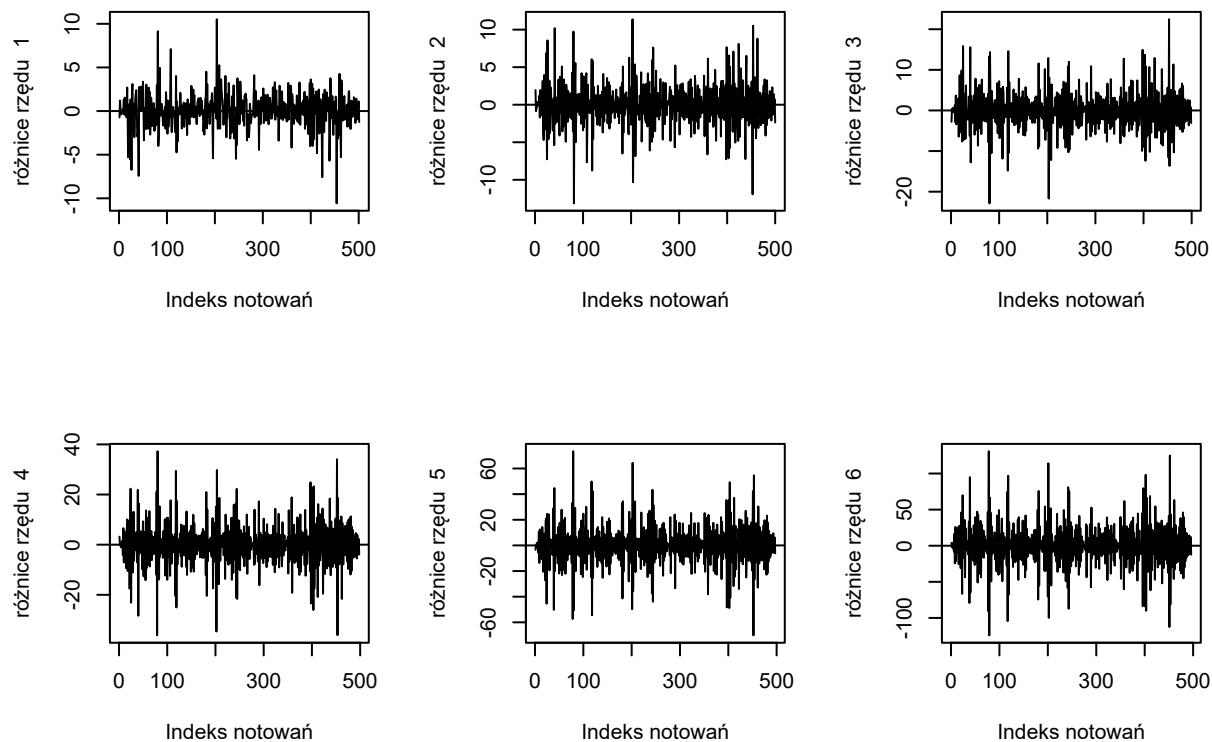
Podobnie jak w przypadku prostej metody średniej ruchomej - im większy parametr η , tym bardziej wygładzony wykres, ale i mniejsza dokładność. Jednakże, dokładność jest i tak większa niż w przypadku prostej metody ruchomej średniej.

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){
  ylab <- paste("różnice rzędu ", i)
  plot(diff(mcd, differences = i), type = "l", xlab = "Indeks notowań", ylab = ylab)
  abline(h = 0)}
```



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

Z wykresów można wysnuć wniosek, że największą stabilizację osiągamy przy różnicowaniu rzędu drugiego. Wraz ze wzrostem rzędu różnicowania, wahania ulegają znacznemu rozszerzeniu.

Gdyby przyjąć, że największą stabilizację osiągamy przy różnicowaniu rzędu 2, to należałoby dopasować wielomian pierwszego stopnia. Spróbujemy do danych dopasować wielomian stopnia 1, 2, 3 i 4.

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

Dopasowanie modelu liniowego

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

```
stargazer(mod1, header = F, table.placement = "H")
```

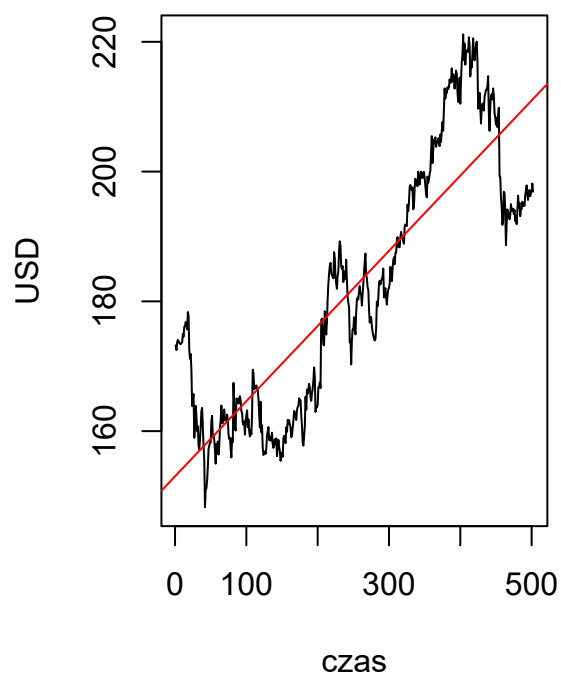
Tablica 1:

<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

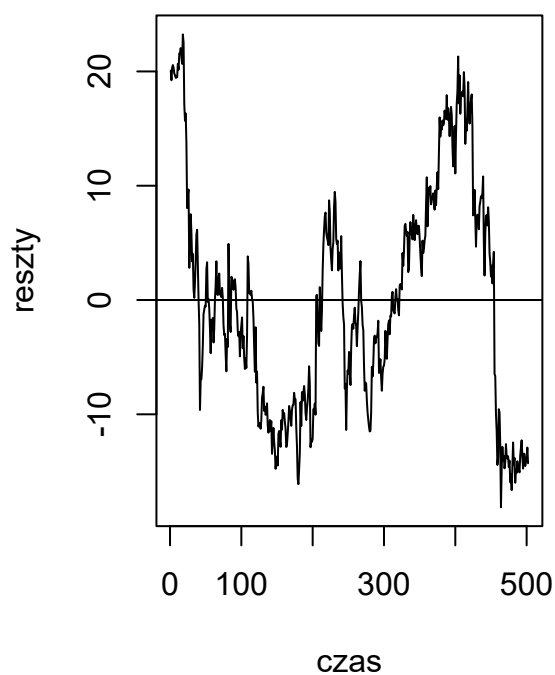
Zarówno wyraz wolny, jak i współczynnik kierunkowy są istotne statystycznie. R^2 wynosi około 74%.

```
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)
```


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))
```

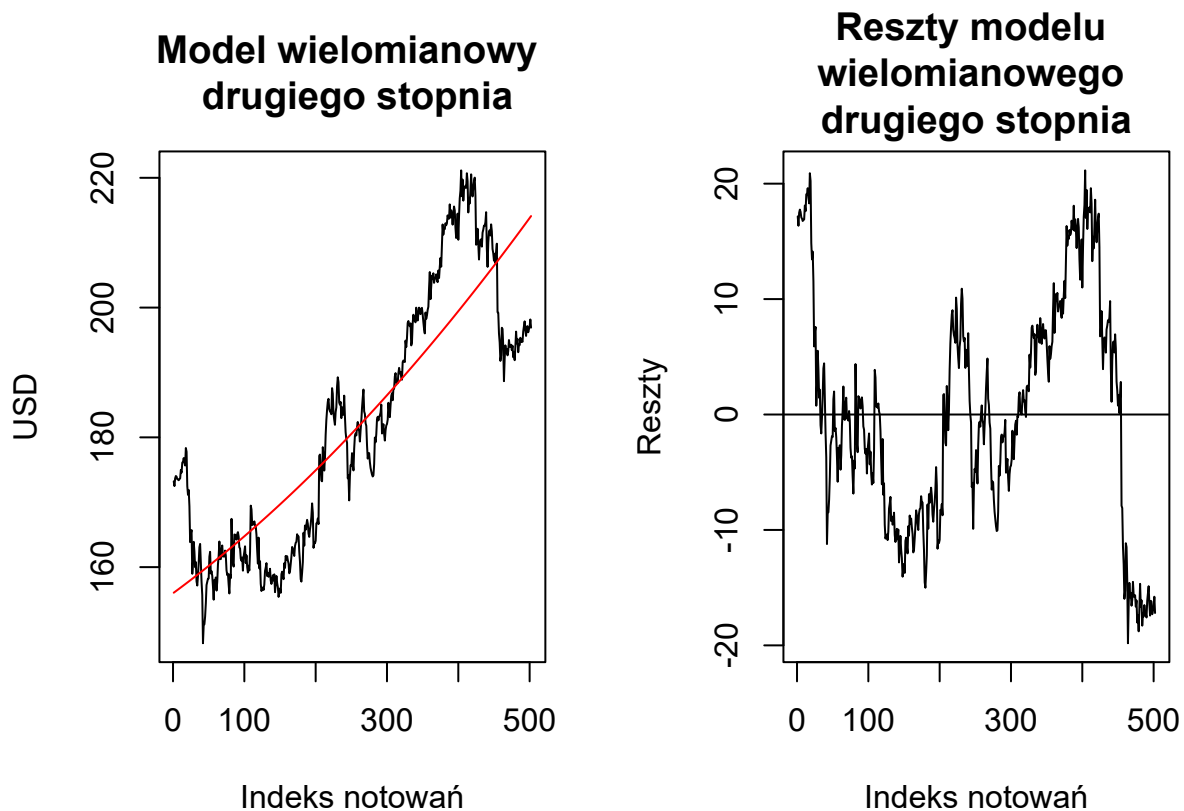
```
stargazer(mod2, header = F, table.placement = "H")
```

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 = "Indeks notowań", ylab = "USD")
lines(t, mod2$fitted.values, col = "red")
plot(mod2$residuals, type = "l",
     main = "Reszty modelu \nwielomianowego \ndrugiego stopnia",
     xlab = "Indeks notowań", ylab = "Reszty")
abline(h = 0)
```



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

Model kwadratowy zachowuje się bardzo podobnie jak model liniowy.

Dopasowanie wielomianem trzeciego stopnia

Jako kolejny zbudowany zostanie model wielomianowy trzeciego stopnia.

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

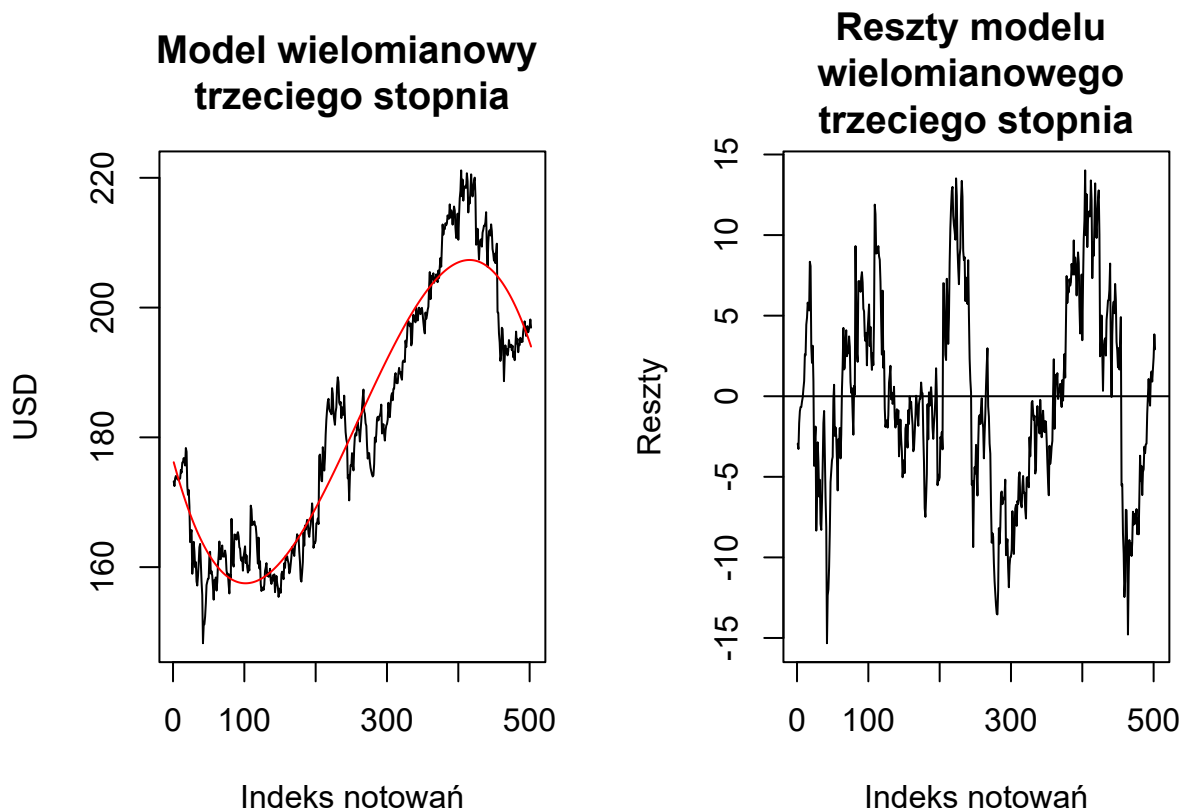
```
stargazer(mod3, header = F, table.placement = "H")
```

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

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 = "Indeks notowań", ylab = "USD")
lines(t, mod3$fitted.values, col = "red")
plot(mod3$residuals, type = "l",
     main = "Reszty modelu \nwielomianowego \ntrzeciego stopnia",
     xlab = "Indeks notowań", ylab = "Reszty")
abline(h= 0)
```



```
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.

Dopasowanie wielomianem czwartego stopnia

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

```
stargazer(mod4, header = F, table.placement = "H")
```

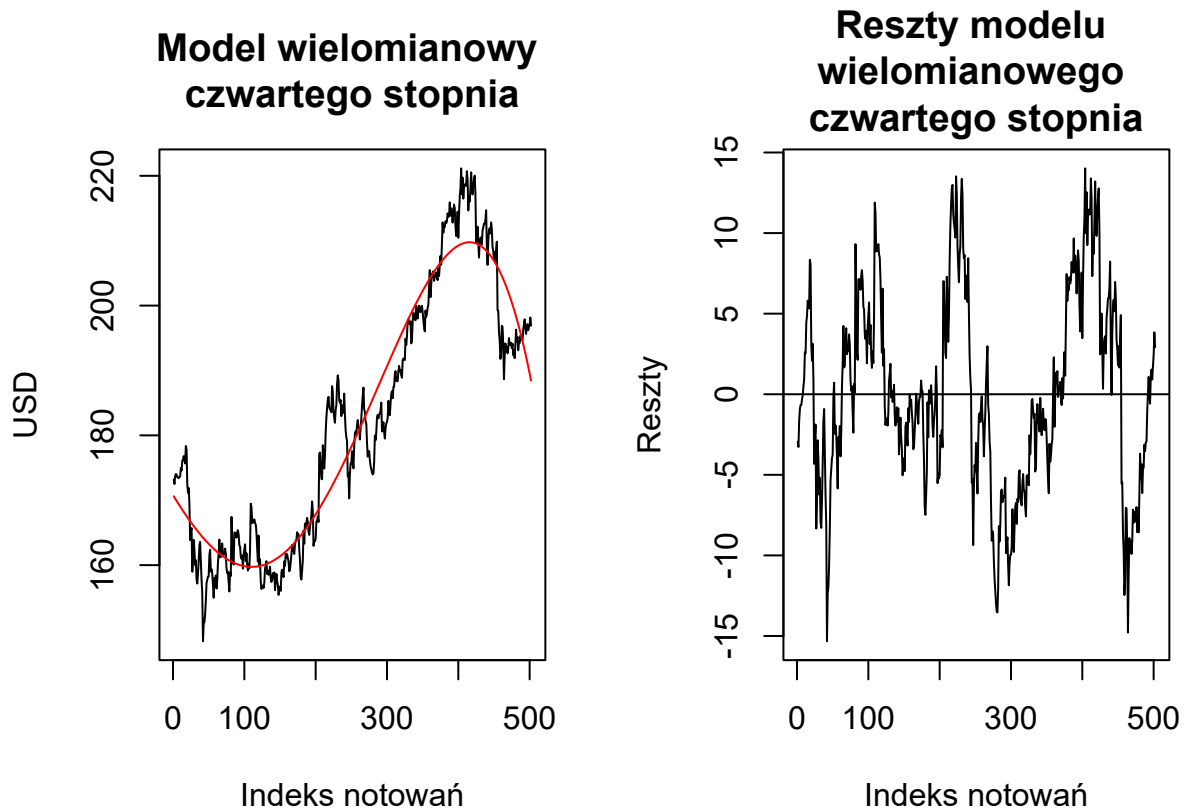
Tablica 4:

	<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 = "Indeks notowań", ylab = "USD")
lines(t, mod4$fitted.values, col = "red")
plot(mod3$residuals, type = "l",
     main = "Reszty modelu \nwielomianowego \nczwartego stopnia",
     xlab = "Indeks notowań", ylab = "Reszty")
abline(h= 0)

```



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

W modelu wielomianowym czwartego stopnia współczynnik przy t^2 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:

- jednorodność wariancji
- normalność
- losowość

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.

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

- Test Goldfelda-Quandta

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.

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

- Test Harrisona-McCabe'a

Sprawdza hipotezę podobną do tej, którą weryfikuje test Goldfelda-Quandta; 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.

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

```
pvals <- cbind(c("Test", "Breuscha-Pagana", "Goldfelda-Quandta", "Harrisona-McCabe'a"),
              c("P-value", pv1, pv2, pv3))
```

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

	Test	P-value
BP	Breuscha-Pagana	4.4825648537984e-06
	Goldfelda-Quandta	0.678495577817587
	Harrisona-McCabe'a	0

Z powyższej tabeli widać, że testy Breuscha-Pagana oraz Harrisona-McCabe'a odrzucają hipotezę o jednorodności wariancji. Dla testu Goldfelda-Quandta nie mieliśmy podstaw do odrzucenia hipotezy o jednorodności.

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

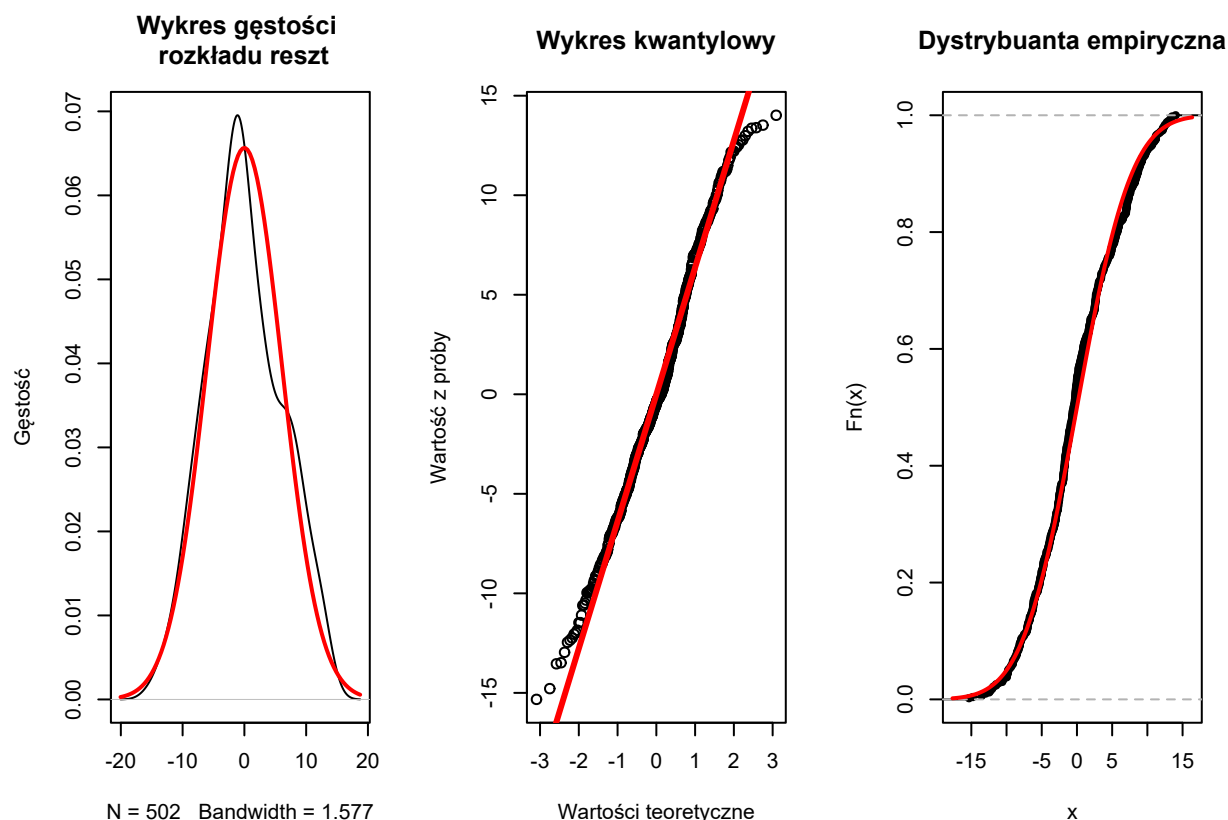
Normalność

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

plot(density(mod3$residuals),
     main = "Wykres gęstości \n rozkładu reszt", ylab = "Gęstość")
curve(dnorm(x, 0, sd(mod3$residuals)), add = T, col = 2, lwd = 2)

qqnorm(mod3$residuals, main = "Wykres kwantylowy", xlab = "Wartości teoretyczne",
       ylab = "Wartość z próby")
qqline(mod3$residuals, col=2, lwd = 3)

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

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

```
kstest <- ks.test(x = mod3$residuals, y = "pnorm",
                  mean = 0, sd = sd(mod3$residuals))$p.value
lillie <- lillie.test(mod3$residuals)$p.value
shapiro <- shapiro.test(mod3$residuals)$p.value
adtest <- ad.test(mod3$residuals)$p.value
pvals <- cbind(c("Test", "Kołmogorowa-Smirnova", "Kołmogorowa-Lillieforsa",
                  "Shapiro-Wilka", "Andersona-Darlinga"),
               c("P-value", kstest, lillie, shapiro, adtest))
```

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

Test	P-value
Kołmogorowa-Smirnova	0.135008394373501
Kołmogorowa-Lillieforsa	0.00262391169251782
Shapiro-Wilka	0.00135637781682188
Andersona-Darlinga	0.00169110383024725

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). 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. Dystrybuanta empiryczna jest zbliżona do dystrybuanty rozkładu normalnego.

Autokorelacja

- Test serii

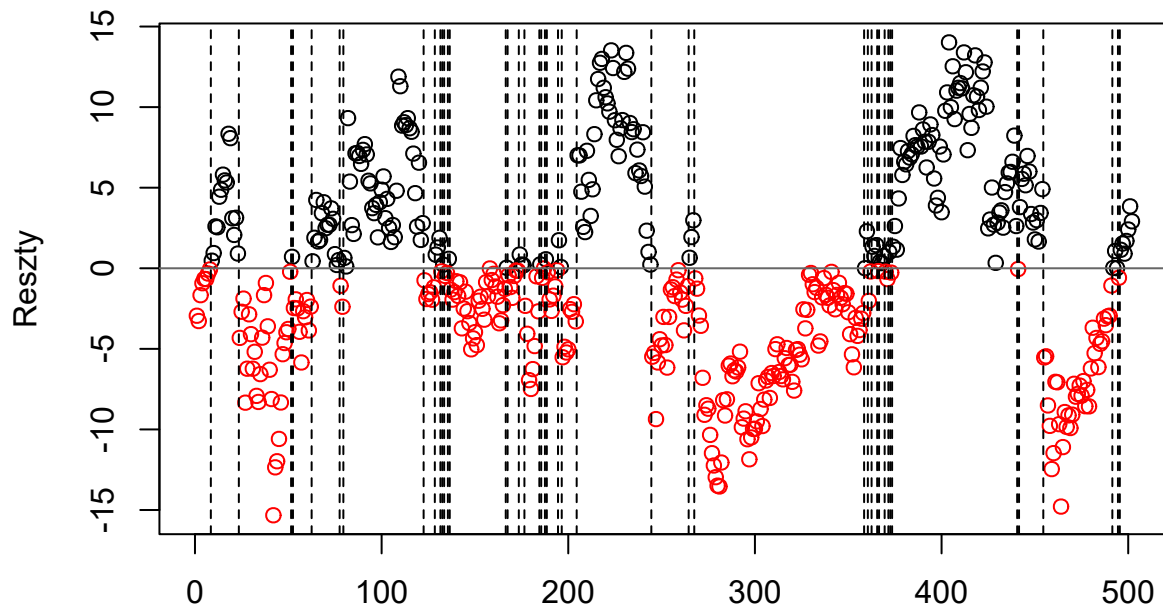
H_0 : losowość

H_1 : brak losowości

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

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

runstest <- runs.test(Reszty, threshold = 0, plot = T)$p.value
```



P-value bliskie zero, odrzucamy hipotezę o losowości reszt. Także na wykresie widać, że prawdopodobnie istnieje zależność między kolejnymi resztami, prawdopodobnie występuje autokorelacja.

- Test Durbina-Watsona

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

```
dwtest <- dwtest(mod3, order.by = ~t)$p.value
```

- Test Breuscha-Godfrey

```
bgtest <- bgtest(mod3, order.by = ~t, order = 3)$p.value
```

- Test Ljunga-Boxa

H_0 : niezależność

H_1 : brak niezależności

```
lbox <- Box.test(mod3$residuals, type = "Ljung-Box")$p.value
```

```
pvals <- cbind(c("Test", "Serii", "Durbina-Watsona", "Breuscha-Godfrey", "Ljunga-Boxa"), c("P-value", r
```

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

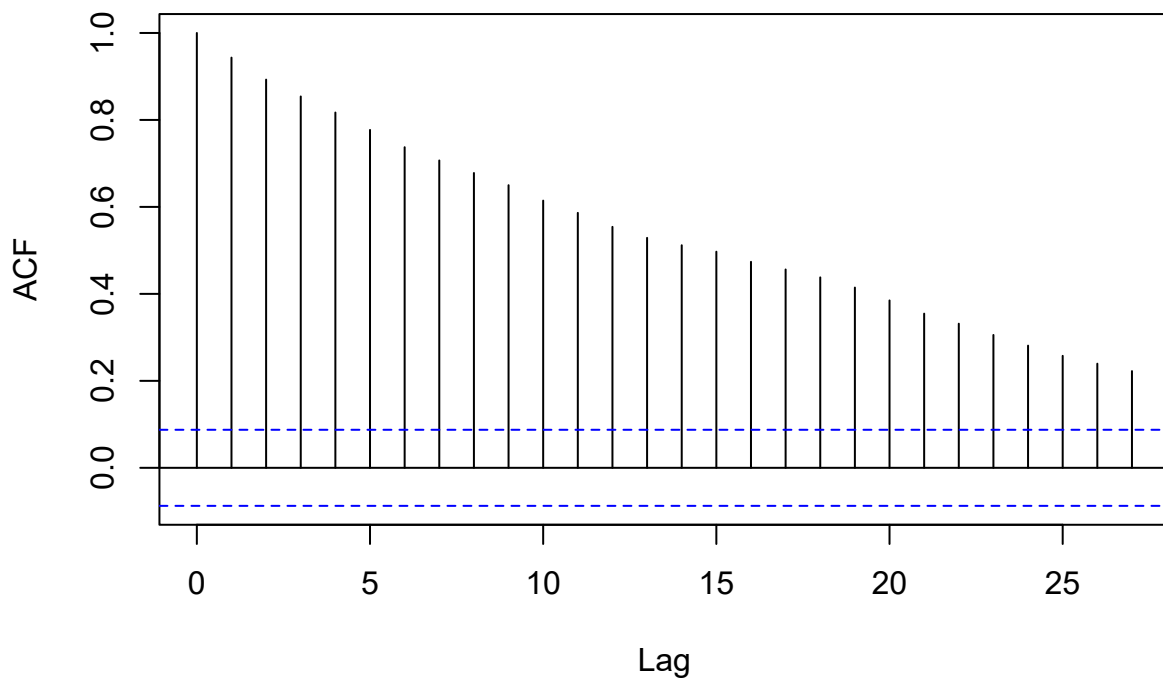
Test	P-value
Serii	7.28183695286577e-77
Durbina-Watsona	3.33292618539736e-101
Breuscha-Godfrey	1.23174387492681e-96
Ljunga-Boxa	0

Jak widać z powyższej tabelki, w każdym z tych czterech testów p-value było praktycznie równe zero, zatem odrzucamy hipotezy o losowości, braku autokorelacji i niezależności reszt.

- Wykres ACF (funkcji autokorelacji)

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

Wykres funkcji autokorelacji



Również z wykresu funkcji ACF możemy wyciągnąć podobne wnioski. Dla opóźnień do rzędu 25 obserwacje nie mieszczą się w niebieskich przerywanych liniach - wnioskujemy, że pojawia się autokorelacja.

Ostatecznie na podstawie testów oraz wykresu ACF 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 <- adf.test(mcd)$p.value
kpss <- kpss.test(mcd)$p.value
kpsst <- kpss.test(mcd, null = "Trend")$p.value
```

```
pvals <- cbind(c("Test", "ADF", "KPSS", "KPSS, null = Trend"),
              c("P-value", adf, "<0.01", "<0.01") )
```

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

Test	P-value
ADF	0.478786706719168
KPSS	<0.01
KPSS, null = Trend	<0.01

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

```
adf2 <- adf.test(diff(mcd, differences = 1))$p.value
kpss2 <- kpss.test(diff(mcd, differences = 1))$p.value
kpsst2 <- kpss.test(diff(mcd, differences = 1), null = "Trend")$p.value
```

```
pvals <- cbind(c("Test", "ADF", "KPSS", "KPSS, null = Trend"),
              c('P-value', "<0.01", ">0.1", ">0.1"))
```

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

Test	P-value
ADF	<0.01
KPSS	>0.1
KPSS, null = Trend	>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

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:

$$\Delta \varepsilon_t = \varepsilon_t$$

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

Testy na resztach modelu ARIMA

Dla reszt otrzymanego modelu ARIMA można badać skorelowanie oraz normalność rozkładu. Tak jak poprzednio, posłużymy się testami Kołmogorowa-Smirnova, Lillieforsa-Smirnova, Shapiro-Wilka i Andersona-Darlinga dla zbadania normalności.

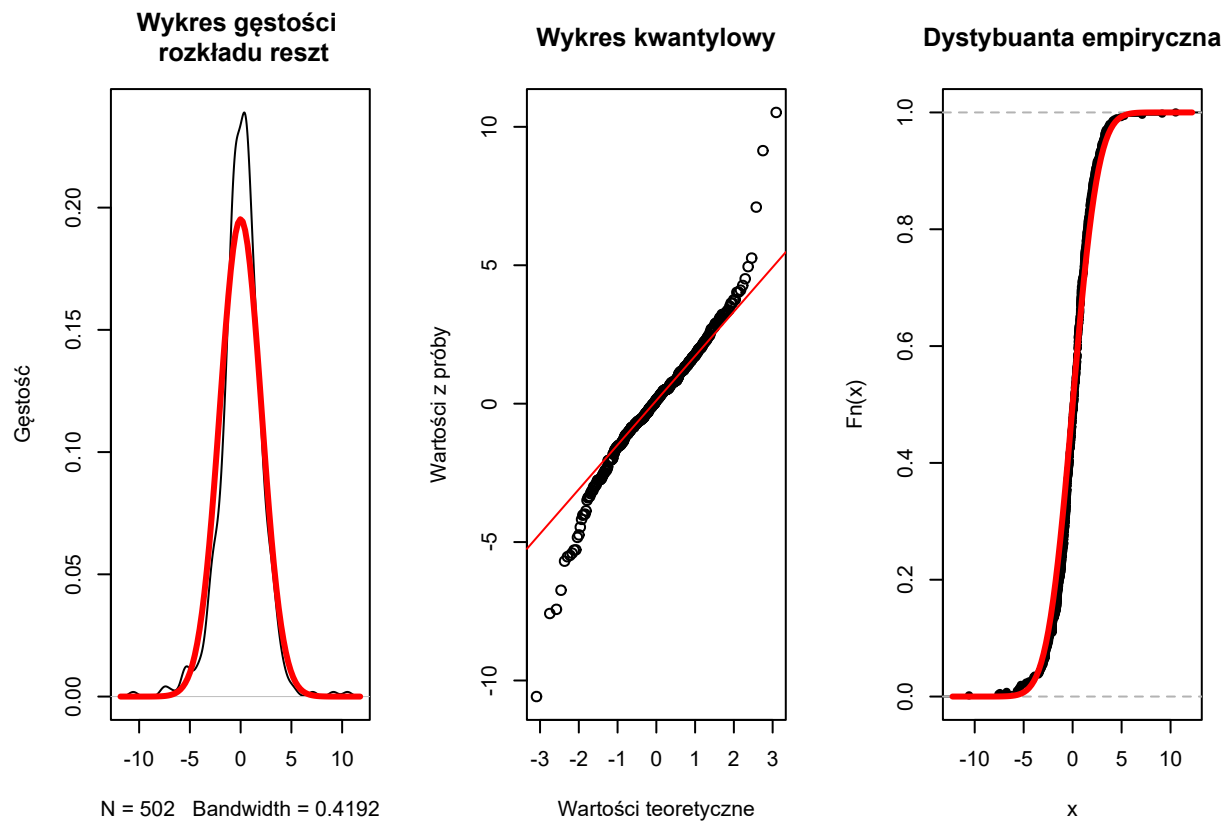
```

par(mfrow = c(1,3))
plot(density(mcd_arma$residuals), main = "Wykres gęstości \n rozkładu reszt",
     ylab = "Gęstość")
curve(dnorm(x,0, sd = sd(mcd_arma$residuals)), col = 2, add = T, lwd = 3)

qqnorm(mcd_arma$residuals, main = "Wykres kwantylowy", xlab = "Wartości teoretyczne",
       ylab = "Wartości z próby")
qqline(mcd_arma$residuals, col = 2)

plot(ecdf(mcd_arma$residuals), main = "Dystybuanta empiryczna")
curve(pnorm(x,0, sd = sd(mcd_arma$residuals)), add = T, lwd = 3, col = 2)

```



```

par(mfrow = c(1,1))

```

Z wykresu gęstości rozkładu reszt można wnioskować, że prawdopodobnie ten rozkład charakteryzuje się dodatnią kurtozą, co powoduje jego “wyciągnięcie” w górę, natomiast z wykresu kwantylowego jasno wynika, że rozkład nie może być normalny.

```

kstest <- ks.test(x = mcd_arma$residuals, y = "pnorm",
                  mean = 0, sd = sd(mcd_arma$residuals))$p.value
lillie <- lillie.test(mcd_arma$residuals)$p.value
shapiro <- shapiro.test(mcd_arma$residuals)$p.value
adtest <- ad.test(mcd_arma$residuals)$p.value
pvals <- cbind(c("Test", "Kołmogorowa-Smirnova", "Kołmogorowa-Lillieforsa",
                  "Shapiro-Wilka", "Andersona-Darlinga"),
               c("P-value", kstest, lillie, shapiro, adtest))

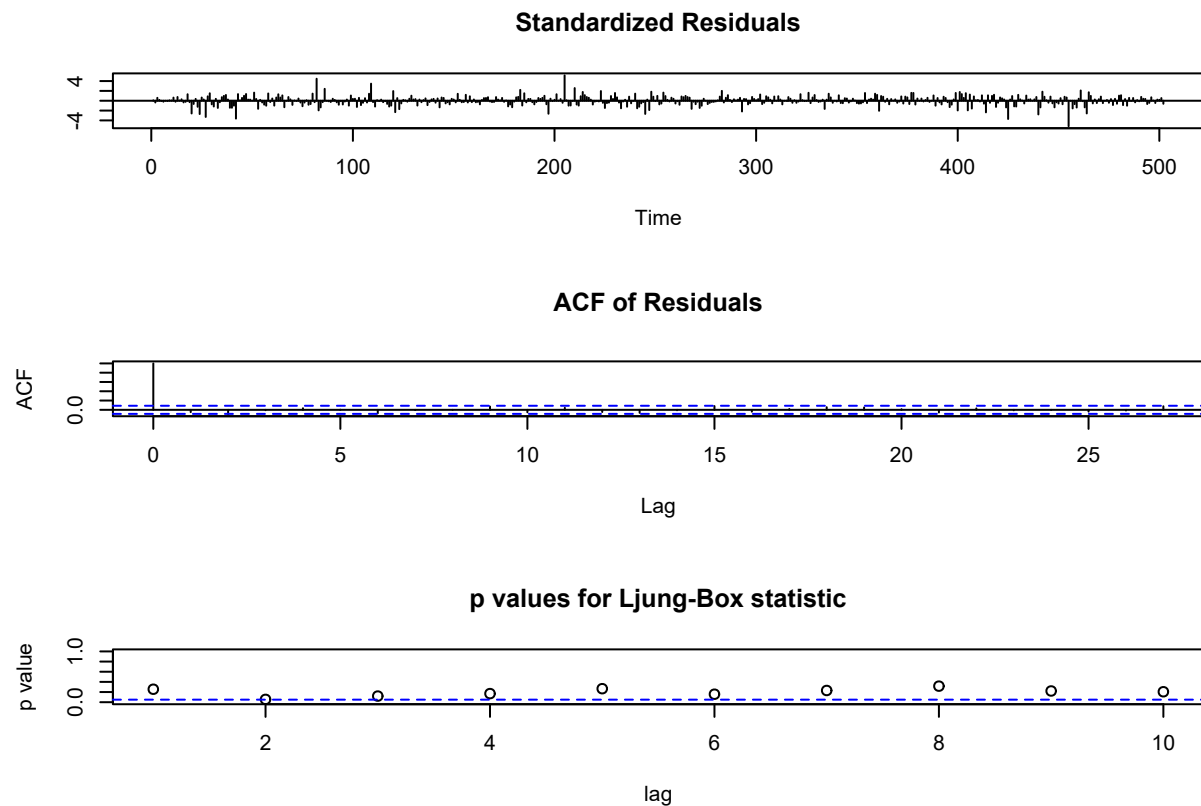
```

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

Test	P-value
Kołmogorowa-Smirnova	0.00592771595058472
Kołmogorowa-Lillieforsa	8.43073447122985e-06
Shapiro-Wilka	1.7002038615427e-11
Andersona-Darlinga	1.18854921878444e-09

Wszystkie testy jednoznacznie każą odrzucić hipotezę o normalności rozkładu reszt modelu ARIMA, co potwierdza wnioski, które nasuwają się przy analizie wykresów.

```
tsdiag(mcd_arima)
```



Na pierwszym wykresie widać, że rozrzut reszt to około od -4 do 4.

Na drugim wykresie można zauważyć brak autokorelacji - kolejne opóźnienia mieszczą się w przerywanych niebieskich liniach.

Na ostatnim wykresie mamy zaznaczone wartości p-value z testu Ljunga-Boxa dla reszt. Są one powyżej poziomu istotności (zaznaczonego także niebieską przerywaną linią), zatem zależności między resztami nie są istotne statystycznie.

Na podstawie powyższych wykresów możemy przypuszczać, że nie zachodzi skorelowanie reszt.

Prognoza

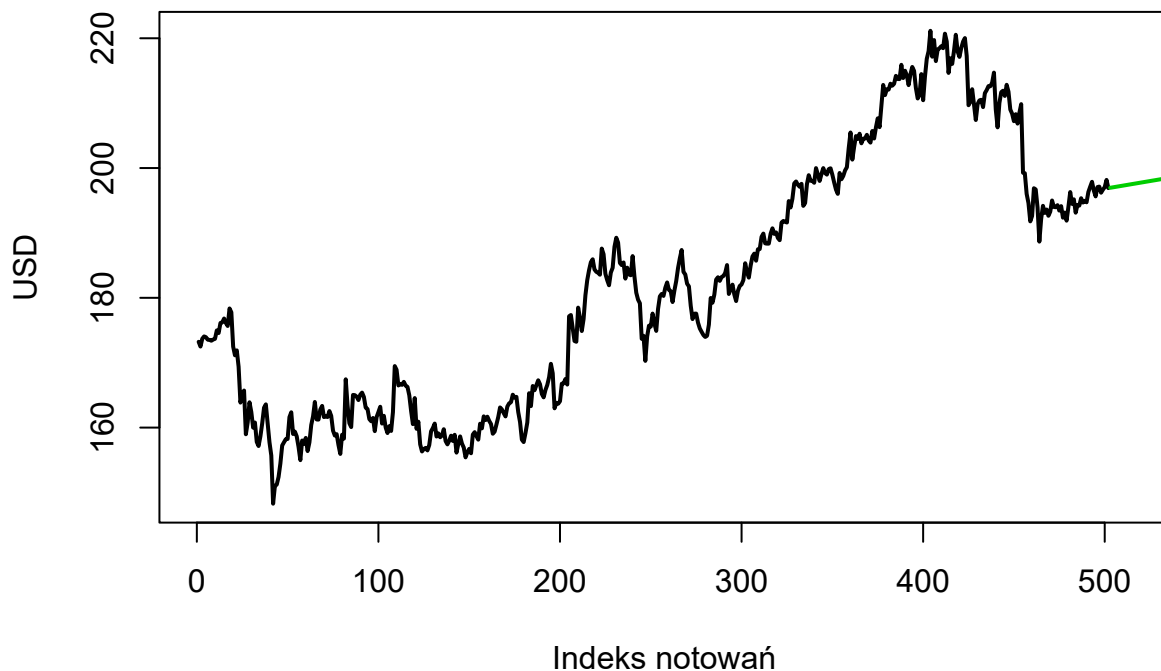
Metoda dryftu

Ponieważ badany szereg czasowy nie wykazuje sezonowości, nie możemy zatem przeprowadzić predykcji metodą Holta-Wintersa. Zamiast tego użyjemy metody dryftu, będącej odmianą metody naiwnej.

Metoda dryftu polega na połączeniu linią prostą pierwszej i ostatniej obserwacji. Przedłużenie tej prostej jest prognozą. Zatem ta metoda może być niedokładna, pozwala jedynie zaobserwować przybliżony trend.

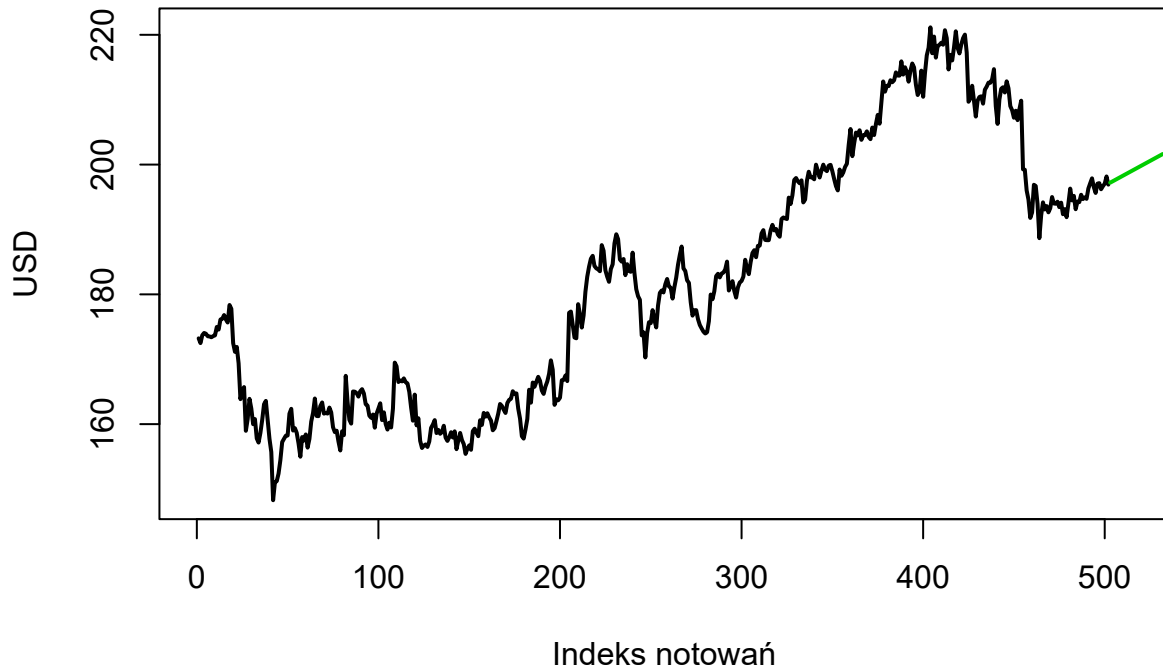
```
prognoza <- rwf(mcd, drift = T, h = 30)
prognoza <- as.data.frame(prognoza)[,1]
plot(mcd, type = "l", main = "Prognoza z całości danych na najbliższe 30 notowań",
     xlim = c(0,515), xlab = "Indeks notowań", ylab = "USD", lwd = 2)
t1 <- 503:(503+length(prognoza)-1)
lines(t1, prognoza, col = 3, lwd = 2)
```

Prognoza z całości danych na najbliższe 30 notowań



```
prognoza <- rwf(mcd[480:500], drift = T, h = 30)
prognoza <- as.data.frame(prognoza)[,1]
plot(mcd, type = "l", main = "Prognoza z ostatnich 20 notowań na najbliższe 30",
     xlim = c(0,515), xlab = "Indeks notowań", ylab = "USD", lwd = 2)
t1 <- 503:(503+length(prognoza)-1)
lines(t1, prognoza, col = 3, lwd = 2)
```


Proгноза z ostatnich 20 notowań na najbliższe 30



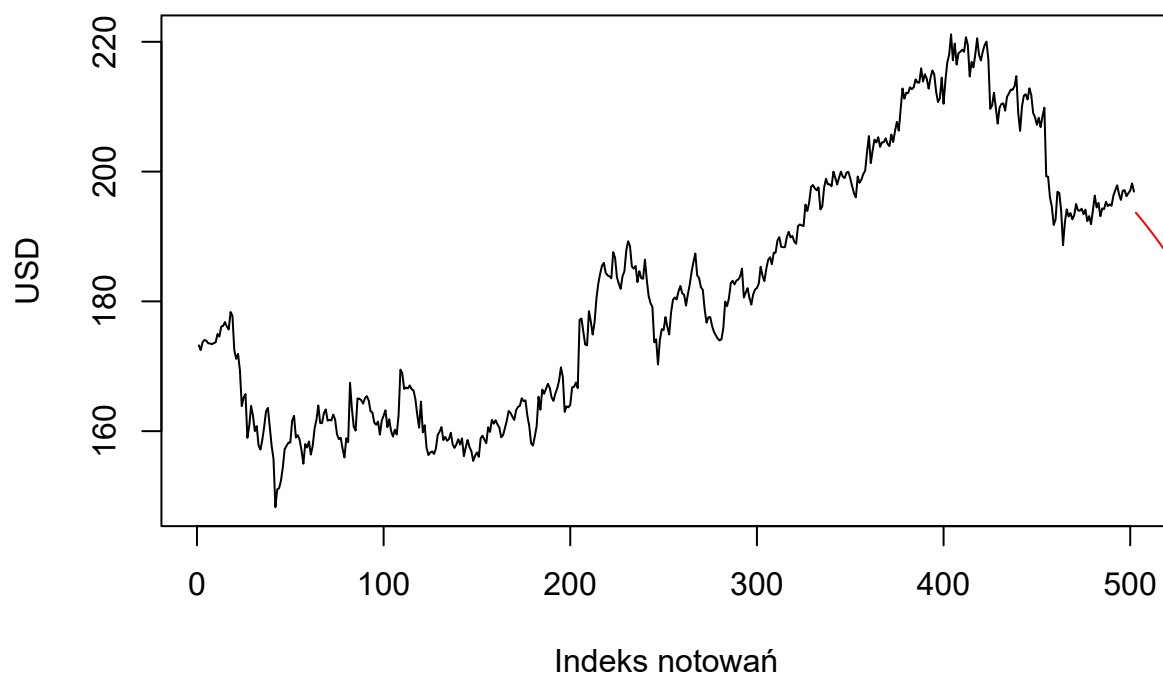
Z racji braku sezonowości, prognoza nie uwzględnia wahań, a jedynie ogólnie występujący trend. Z tego względu jest dość niedokładna i bardziej wiarygodne wyniki można uzyskać prognozując na krótsze odcinki czasu. Widać to z porównania powyższych wykresów - nachylenie linii prognozy, biorącej pod uwagę wszystkie wartości notowań jest mniejsze i nie uwzględnia istotnych zmian, które zaszły pomiędzy pierwszym a ostatnim notowaniem.

Przewidywania z modelu

Próbujemy przewidywać z modelu wielomianowego trzeciego stopnia.

```
wsp <- mod3$coefficients
funkcja <- function(t){
  wsp[1]+wsp[2]*t+wsp[3]*t^2+wsp[4]*t^3
}
t2 <- 503:520
plot(mcd, type = "l",
     main = "Przewidywane notowania McDonald's \nna podstawie modelu wielomianowego",
     xlab = "Indeks notowań", ylab = "USD")
lines(t2, funkcja(t2), col = 2)
```

Przewidywane notowania McDonald's na podstawie modelu wielomianowego



Model trzeciego stopnia okazuje się niewystarczający do prognoz, ponieważ funkcja trzeciego stopnia była w tym miejscu malejąca, a notowania firmy zaczęły wzrastać.

Podsumowanie analizy notowań McDonald's

Do danych o notowaniach firmy McDonald's spróbowaliśmy dobrać dwa rodzaje modeli: wielomianowy oraz ARIMA. Mamy następujące wnioski:

- z modeli wielomianowych za najlepszy uznaliśmy model trzeciego stopnia. W tym modelu R^2 wynosiło około 90%, wariancja reszt nie była jednorodna, rozkład różnił się od normalnego, reszty były skorelowane.
- z modeli typu ARIMA najlepiej dopasowany okazał się model ARIMA(0, 1, 0). Reszty tego modelu nie mają rozkładu normalnego. Nie stwierdziliśmy autokorelacji reszt
- najlepsze prognozy otrzymaliśmy metodą dryftu, ale tylko dla kilkunastu ostatnich obserwacji. Przewidywania z modelu wielomianowego nie były poprawne

Starbucks

Opis firmy

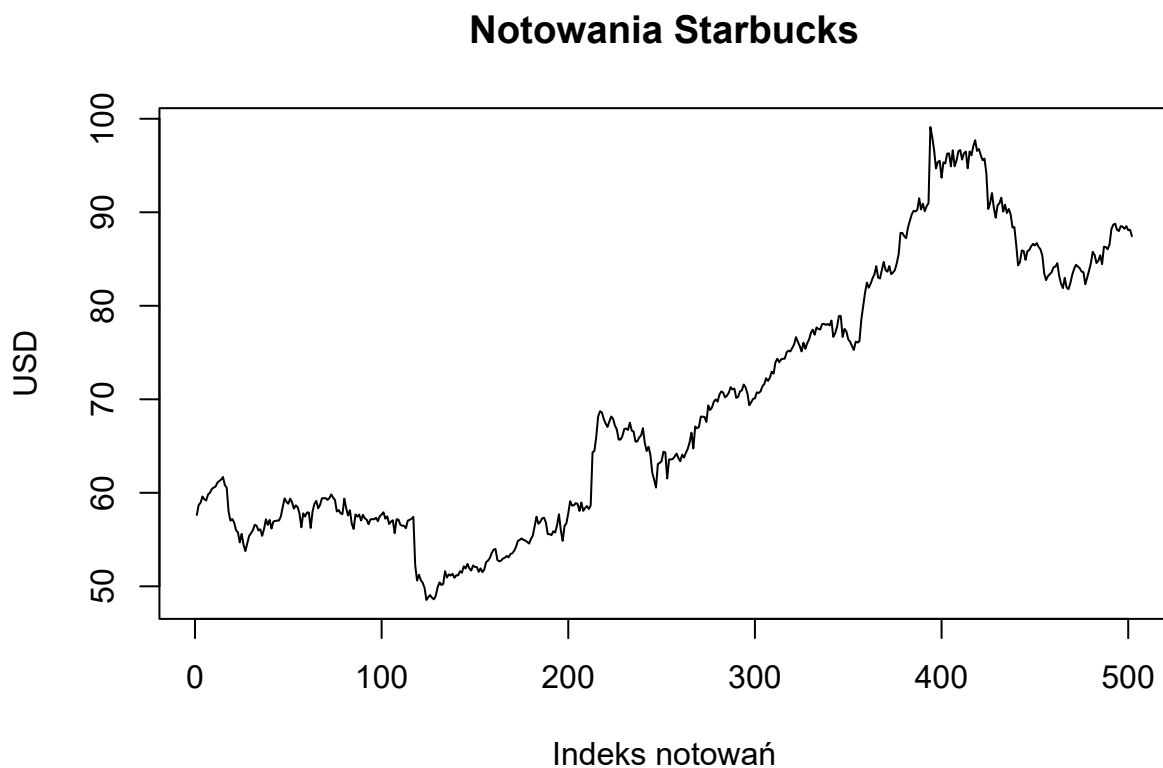
Starbucks Corporation – największa na świecie sieć kawiarni. Została założona 30 marca 1971 w Seattle w stanie Waszyngton.

Wczytanie danych i rysunki

```
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)
```

Wykonamy rysunek przedstawiający notowania firmy Starbucks z ostatnich dwóch lat.

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



Na rysunku możemy zaobserwować trend rosnący.

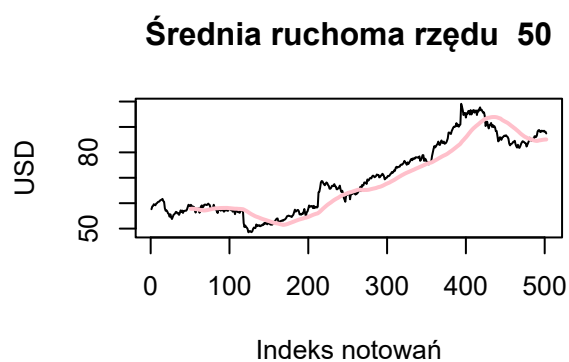
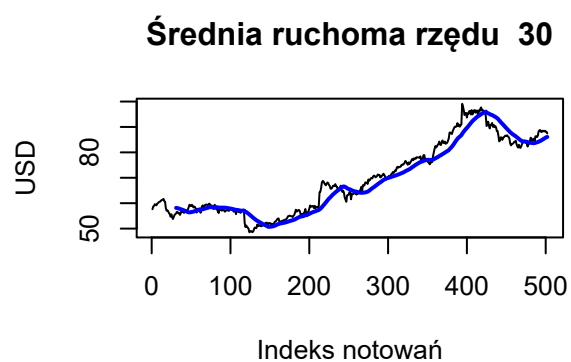
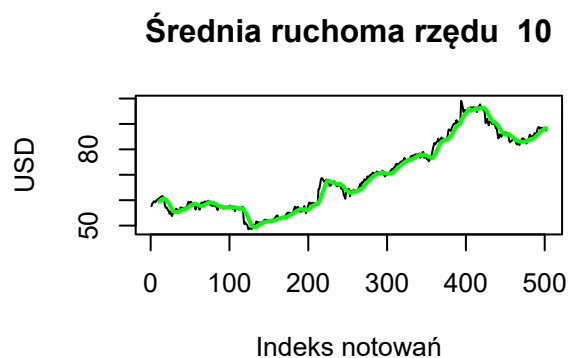
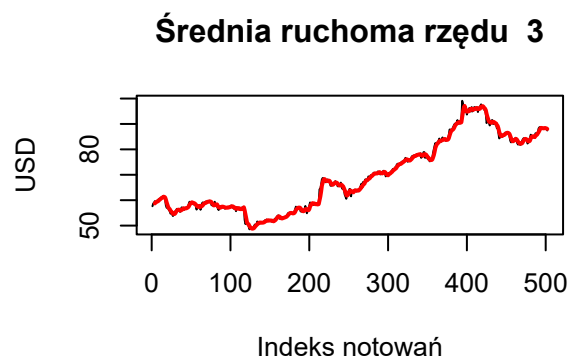
Wydzielanie trendu

Wydzielamy część deterministyczną. Zastosujemy metody średnich ruchomych (prostą i wykładniczą).

Ruchoma średnia

Skorzystamy z napisanej wcześniej funkcji `ruchoma`. Narysujmy wykresy dla kilku parametrów `m`.

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

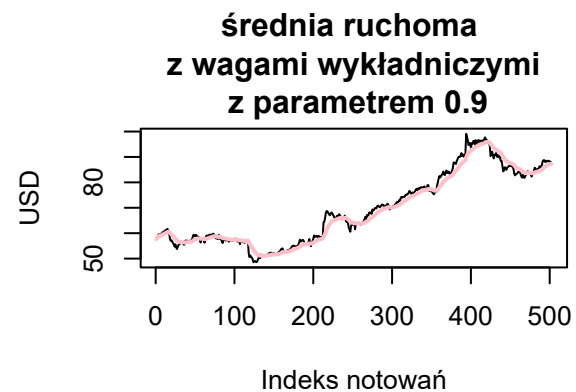
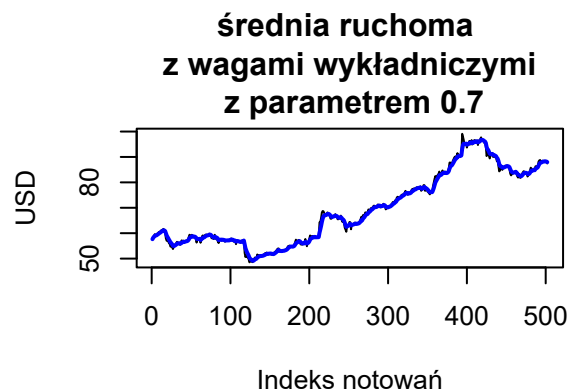
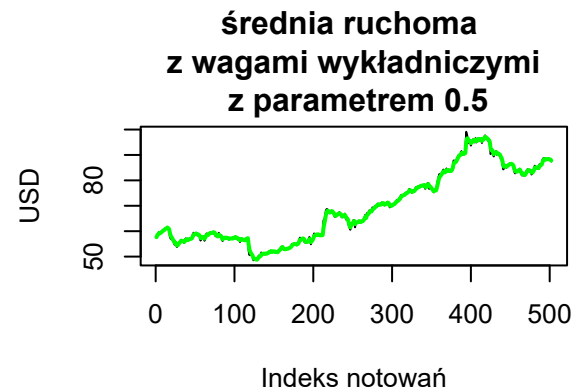
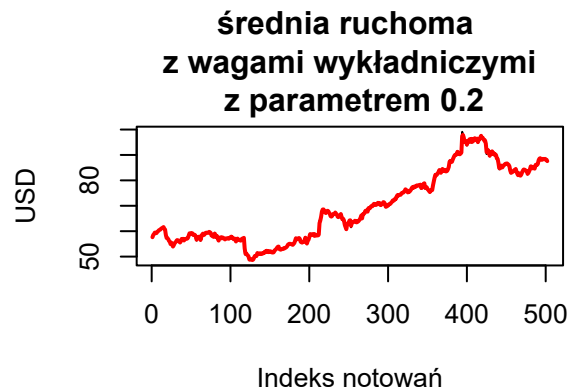


```
par(mfrow=c(1,1))
```

Wraz ze wzrostem parametru m otrzymujemy bardziej wygładzony wykres, ale niestety mniej dokładny.

Metoda wykładniczych wag ruchomej średniej

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



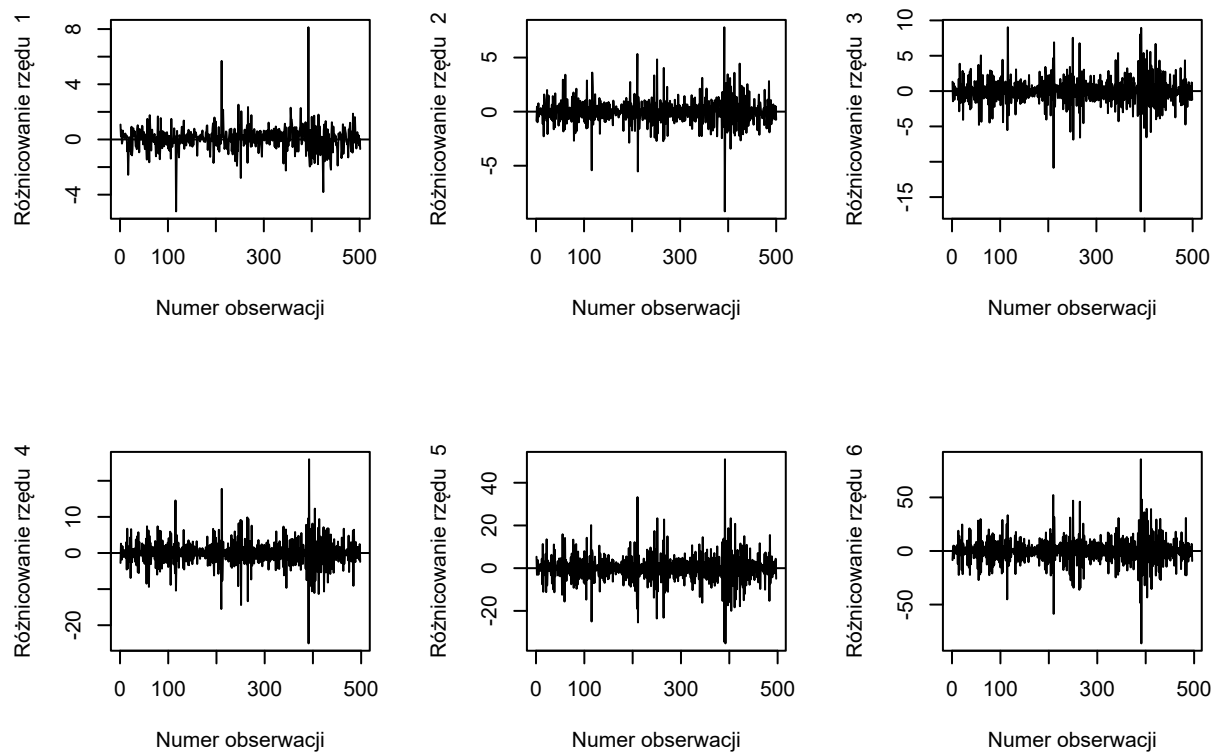
```
par(mfrow=c(1,1))
```

Wraz ze wzrostem parametru η uzyskujemy bardziej wygładzony wykres, ale mniejszą dokładność. Jest ona jednak większa niż w przypadku prostej metody ruchomej średniej.

Dopasowanie wielomianu

Metoda różnicowa

```
par(mfrow = c(2, 3))
for(i in 1:6){
  plot(diff(sbox, differences = i), type = "l",
       xlab = "Numer obserwacji", ylab = paste("Różnicowanie rzędu ", i))
  abline(h = 0)}
```



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

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

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

Dopasowanie modelu liniowego

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

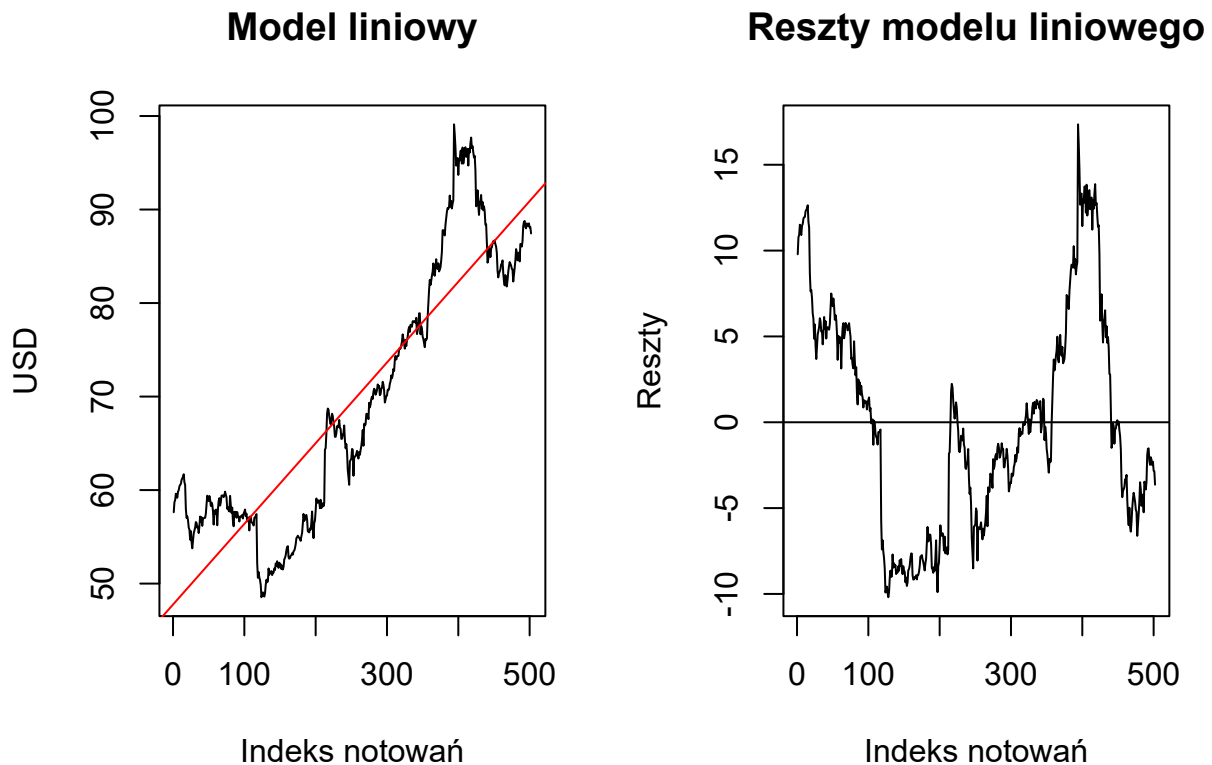
```
stargazer(mod1, header = F, table.placement = "H")
```

Tablica 5:

	<i>Dependent variable:</i>
	sbux
t	0.086*** (0.002)
Constant	47.757*** (0.569)
Observations	502
R ²	0.795
Adjusted R ²	0.794
Residual Std. Error	6.366 (df = 500)
F Statistic	1,937.353*** (df = 1; 500)
<i>Note:</i>	*p<0.1; **p<0.05; ***p<0.01

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

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



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

Dopasowanie wielomianu drugiego stopnia

```
mod2 <- lm(sbx~t+I(t^2), table.placement = "H")
```

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

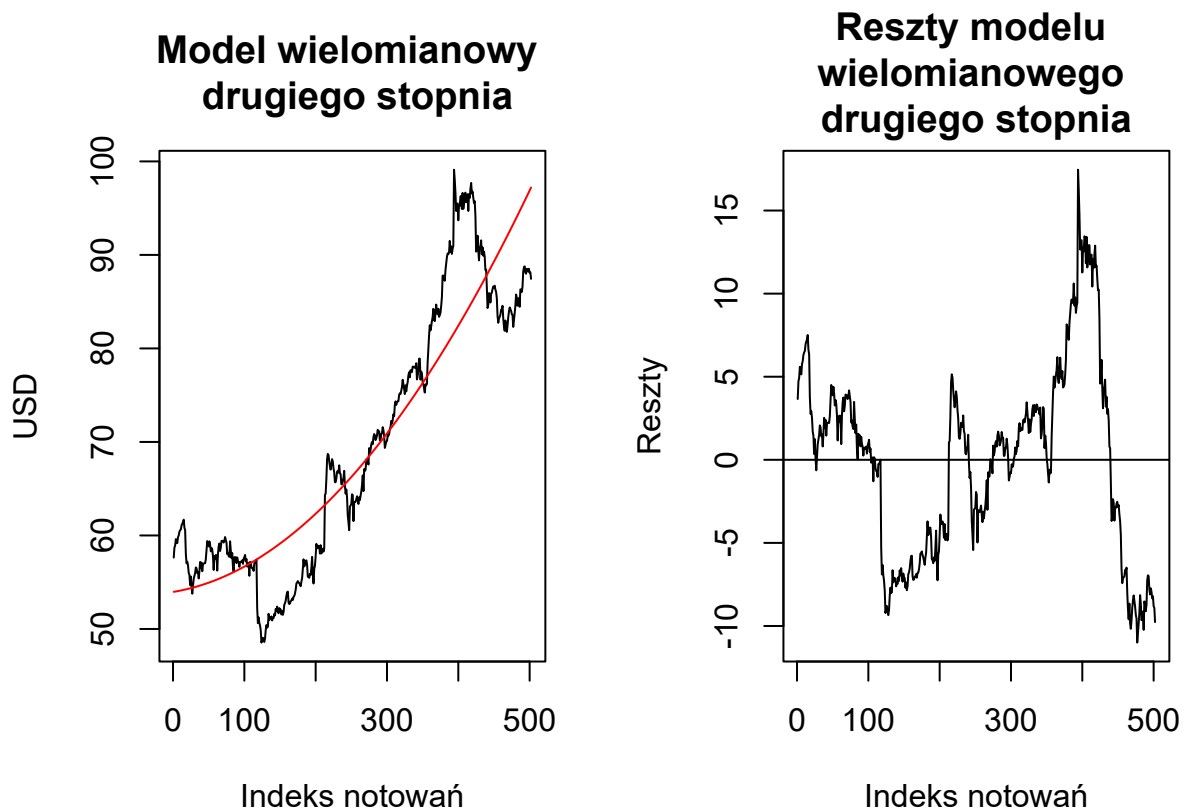
Współczynnik przy t jest nieistotny statystycznie (p-value około 0.08, więc decyzja niejednoznaczna), ale R^2 poprawiło się - wynosi teraz około 83%.

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


Tablica 6:

<i>Dependent variable:</i>	
sbux	
t	0.012* (0.007)
I(t ²)	0.0001*** (0.00001)
Constant	53.960*** (0.772)
Observations	502
R ²	0.834
Adjusted R ²	0.833
Residual Std. Error	5.740 (df = 499)
F Statistic	1,249.086*** (df = 2; 499)

Note: *p<0.1; **p<0.05; ***p<0.01



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

Wykres reszt jest bardzo podobny jak w przypadku modelu liniowego.

Dopasowanie wielomianu trzeciego stopnia

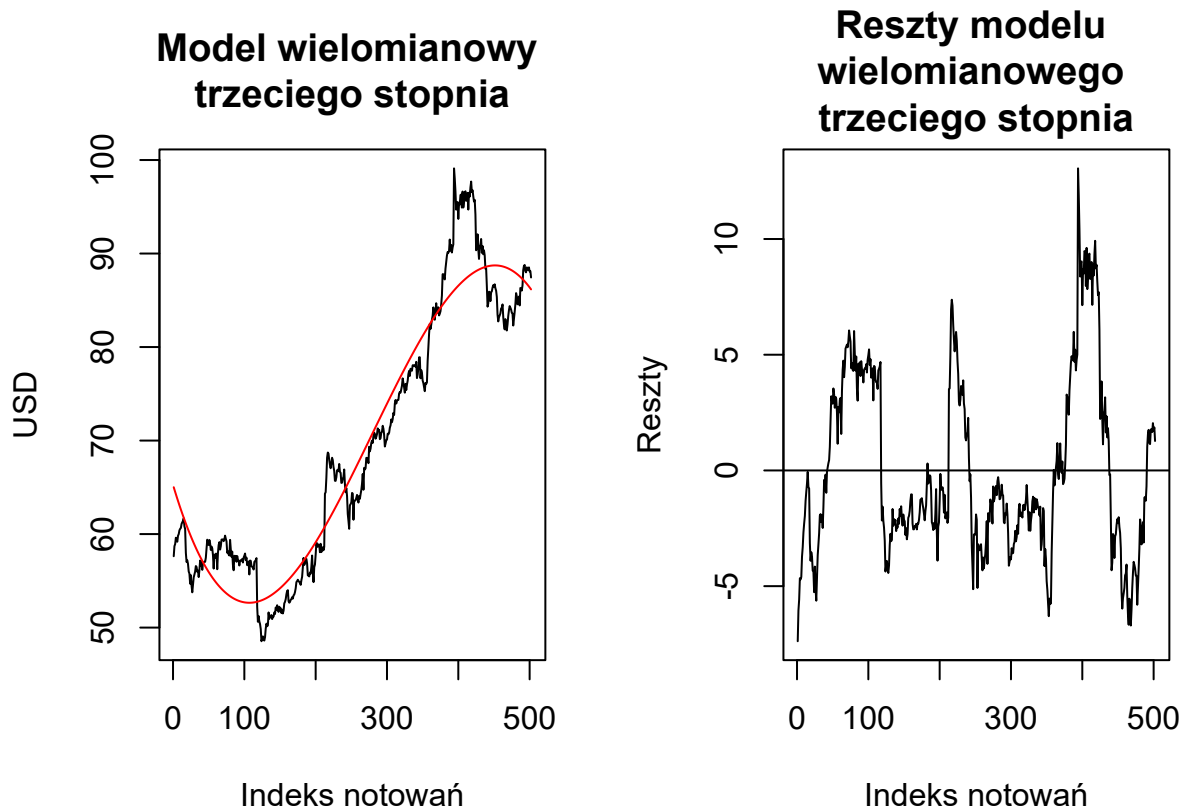
```
mod3 <- lm(sbox~t+I(t^2)+I(t^3))
stargazer(mod3, header = F, table.placement = "H")
```

Tablica 7:

	<i>Dependent variable:</i>
	sbox
t	-0.256*** (0.012)
I(t^2)	0.001*** (0.0001)
I(t^3)	-0.00000*** (0.00000)
Constant	65.265*** (0.698)
Observations	502
R ²	0.924
Adjusted R ²	0.924
Residual Std. Error	3.879 (df = 498)
F Statistic	2,021.491*** (df = 3; 498)
<i>Note:</i>	*p<0.1; **p<0.05; ***p<0.01

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

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



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

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

Testy na resztach modelu

Jednorodność wariancji

- Test Breuscha-Pagana

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

- Test Goldfelda-Quandta

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

- Test Harrisona-McCabe'a

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

```
pvals <- cbind(c("Test", "Breusha-Pagana", "Goldfelda-Quandta", "Harrisona-McCabe'a"),
               c('P-value', pv1, pv2, pv3))
```

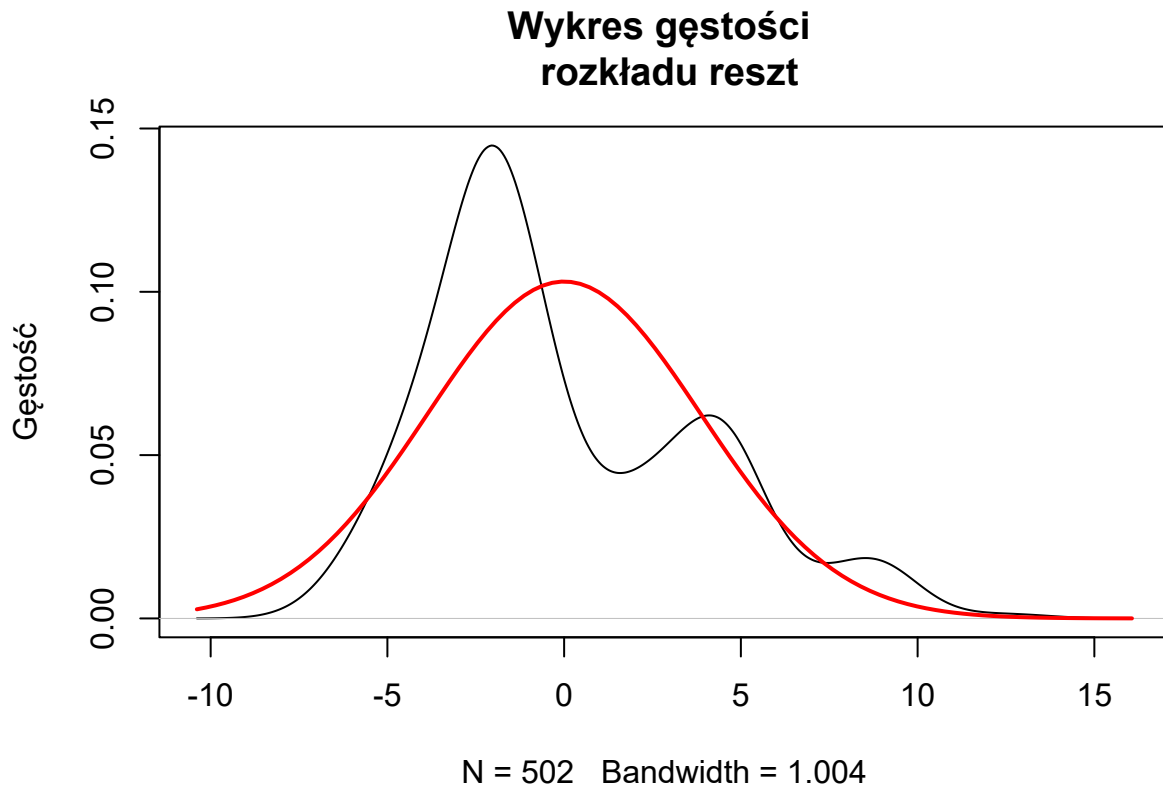
```
kable(pvals)%>%
  kable_styling(full_width = T)
```

	Test	P-value
BP	Breusha-Pagana	1.89655250267285e-09
	Goldfelda-Quandta	1.210963133708e-06
	Harrisona-McCabe'a	0

Testy jednoznacznie wskazują, że należy odrzucić hipotezę o jednorodności wariancji.

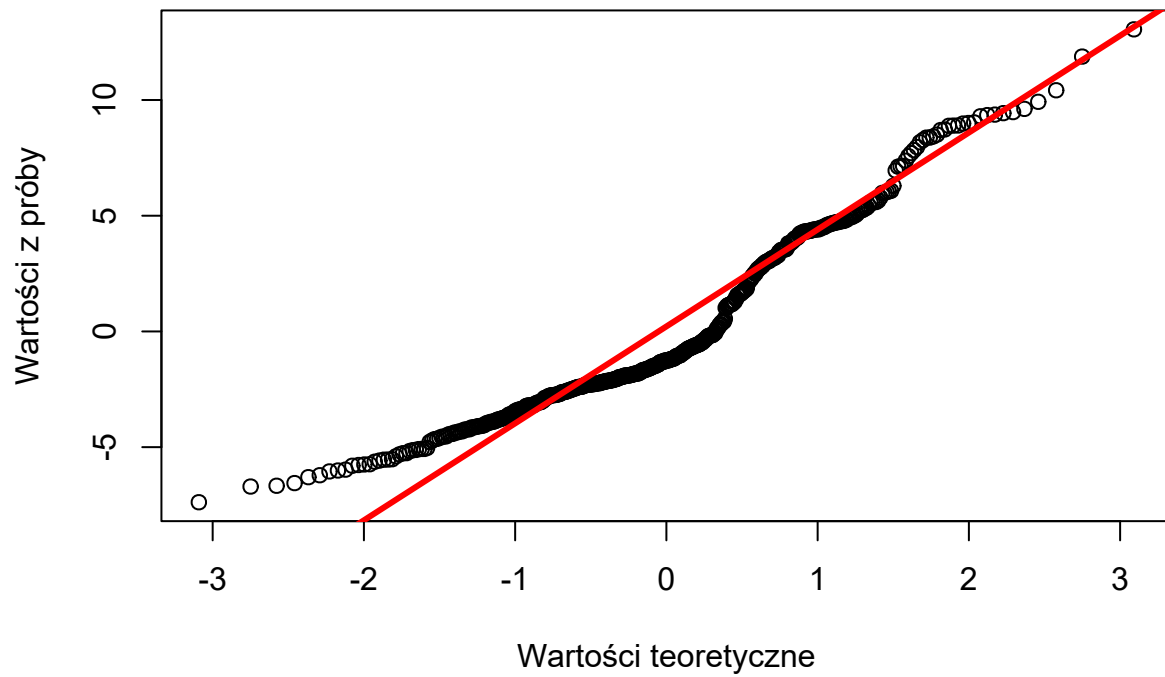
Normalność

```
plot(density(mod3$residuals), main = "Wykres gęstości \n rozkładu reszt",  
     ylab = "Gęstość")  
curve(dnorm(x, 0, sd(mod3$residuals)), add = T, col = 2, lwd = 2)
```



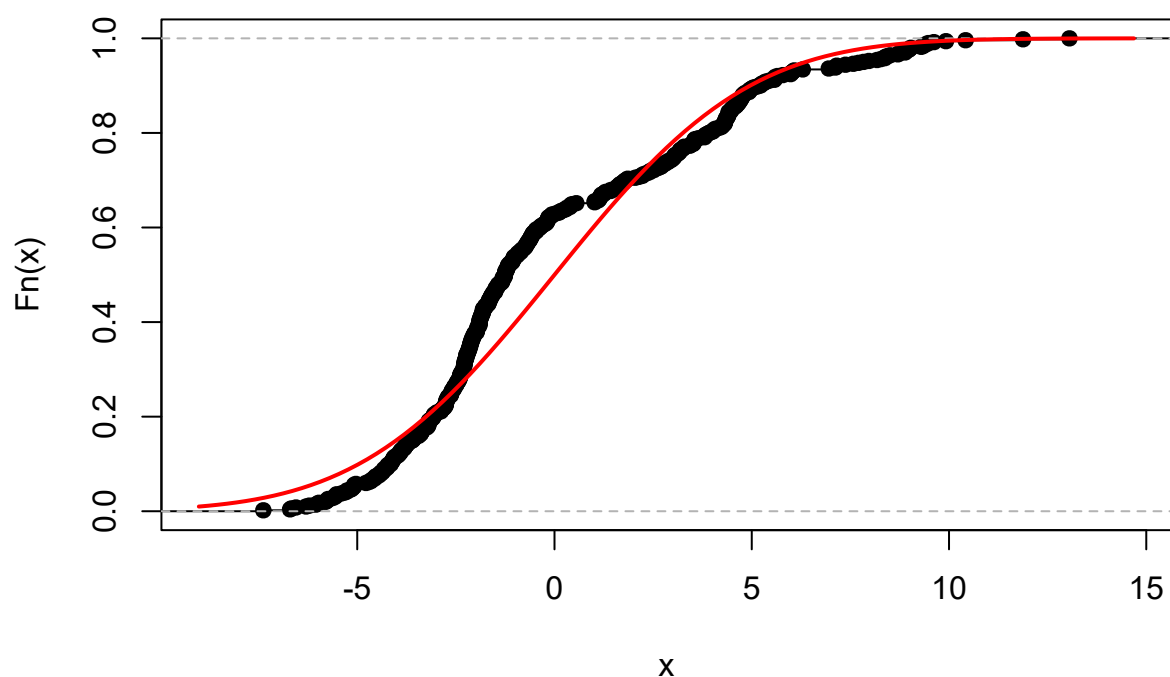
```
qqnorm(mod3$residuals, main = "Wykres kwantylowy",  
       xlab = "Wartości teoretyczne", ylab = "Wartości z próby")  
qqline(mod3$residuals, col=2, lwd = 3)
```

Wykres kwantylowy



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

Dystrybuanta empiryczna



Wykres gęstości rozkładu reszt bardzo znacząco różni się od gęstości rozkładu normalnego. Widać, że rozkład jest co najmniej dwumodalny i ma zbyt silną asymetrię prawostronną. Na wykresie kwantylowym widać znaczące odchyłki od linii kwantylowej. Podobne zjawisko można zaobserwować na wykresie dystrybuanty empirycznej.

```
kstest <- ks.test(x = mod3$residuals, y = "pnorm",
                 mean = 0, sd = sd(mod3$residuals))$p.value
lillie <- lillie.test(mod3$residuals)$p.value
shapiro <- shapiro.test(mod3$residuals)$p.value
adtest <- ad.test(mod3$residuals)$p.value
```

```
pvals <- cbind(c("Test", "Kołmogorowa-Smirnova", "Kołmogorowa-Lillieforsa",
                 "Shapiro-Wilka", "Andersona-Darlinga"),
               c("P-value", kstest, lillie, shapiro, adtest))
```

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

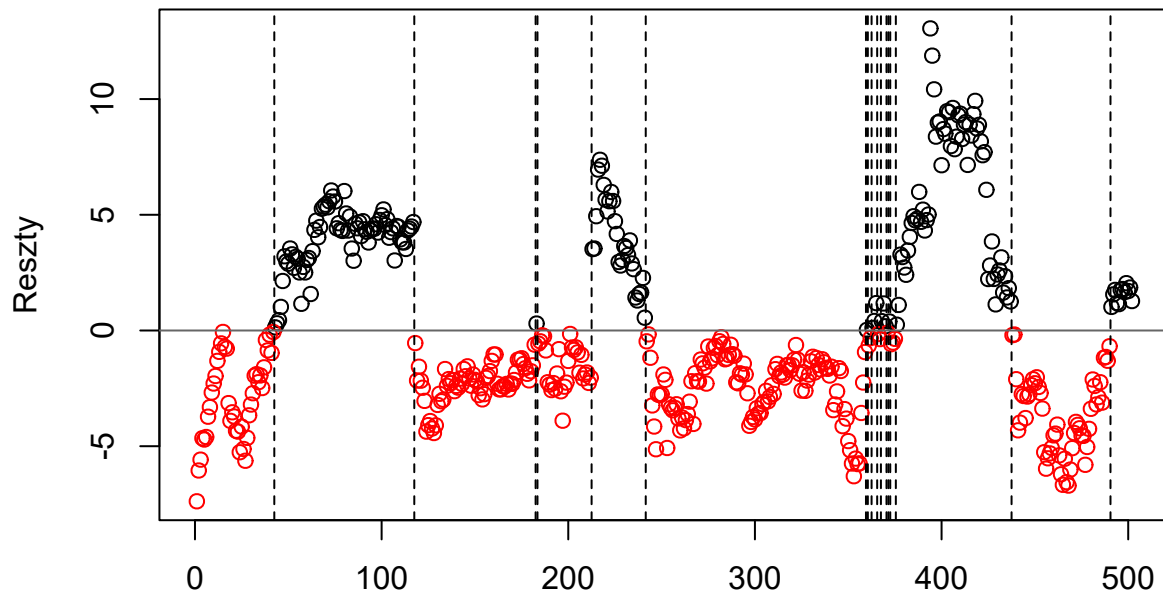
Test	P-value
Kołmogorowa-Smirnova	2.12664008625296e-09
Kołmogorowa-Lillieforsa	7.20238344111382e-28
Shapiro-Wilka	1.04291352020794e-13
Andersona-Darlinga	3.7e-24

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

Autokorelacja

- Test serii

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



```
##  
## Runs Test  
##  
## data: Reszty  
## statistic = -20.806, runs = 18, n1 = 187, n2 = 315, n = 502, p-value <  
## 2.2e-16  
## alternative hypothesis: nonrandomness  
runstest <- runs.test(Reszty, threshold = 0, plot = F)$p.value
```

P-value jest bliskie 0, odrzucamy hipotezę zerową o losowości reszt. Wykres także wskazuje na skorelowanie reszt.

- Test Durbina-Watsona

```
dwtest <- dwtest(mod3, order.by = ~t)$p.value
```

- Test Breuscha-Godfrey

```
bgtest <- bgtest(mod3, order.by = ~t, order = 3)$p.value
```

- Test Ljung-Boxa

```
lbox <- Box.test(mod3$residuals, type = "Ljung-Box")$p.value

pvals <- cbind(c("Test", "Serii", "Durbina-Watsona", "Breuscha-Godfrey", "Ljunga-Boxa"),
              c("P-value", runstest, dwtest, bgtest, lbox))

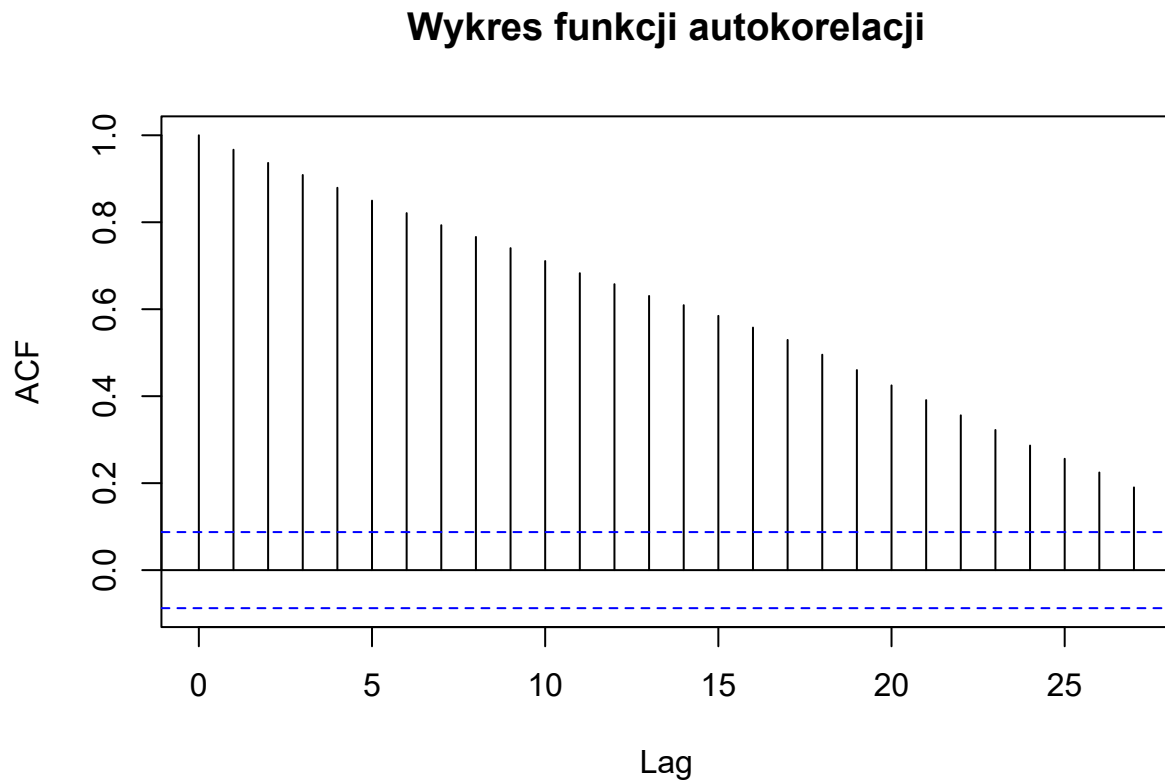
kable(pvals)%>%
  kable_styling(full_width = T)
```

Test	P-value
Serii	3.81849533915512e-96
Durbina-Watsona	7.60673257752725e-107
Breuscha-Godfrey	2.12336808627621e-101
Ljunga-Boxa	0

W każdym z powyższych testów p-value było praktycznie równe 0.

- Wykres ACF (funkcji autokorelacji)

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



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

Na podstawie testów oraz wykresu ACF wnioskujemy, że reszty z modelu `mod3` są skorelowane.

Stacjonarność

```
adf <- adf.test(sbx)$p.value
kpss <- kpss.test(sbx)$p.value
```



```
kpsst <- kpss.test(sbx, null = "Trend")$p.value
```

```
pvals <- cbind(c("Test", "ADF", "KPSS", "KPSS, null = Trend"),
              c("P-value", adf, "<0.01", "<0.01"))
```

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

Test	P-value
ADF	0.507943256908374
KPSS	<0.01
KPSS, null = Trend	<0.01

Szereg nie jest ani stacjonarny ani TS.

```
adf2 <- adf.test(diff(sbx, differences = 1))$p.value
kpss2 <- kpss.test(diff(sbx, differences = 1))$p.value
kpsst2 <- kpss.test(diff(sbx, differences = 1), null = "Trend")$p.value
```

```
pvals <- cbind(c("Test", "ADF", "KPSS", "KPSS, null = Trend"),
              c("P-value", "<0.01", ">0.1", ">0.1"))
```

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

Test	P-value
ADF	<0.01
KPSS	>0.1
KPSS, null = Trend	>0.1

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

```
n <- ndiffs(sbx)
```

Liczba różnicowań uzyskana za pomocą funkcji `ndiffs` wynosi 1.

ARIMA

```
auto.arima(sbx)
```

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

Najlepszy model dla badanego szeregu czasowego to ARIMA(0, 1, 0).

```
sbx_arima <- arima(sbx, c(0,1,0))
```

Kryterium Akaike dla modelu ARIMA(0, 1, 0) wynosi około 1364.51.

Model możemy zapisać w postaci:

$$\Delta \epsilon_t = \epsilon_t$$

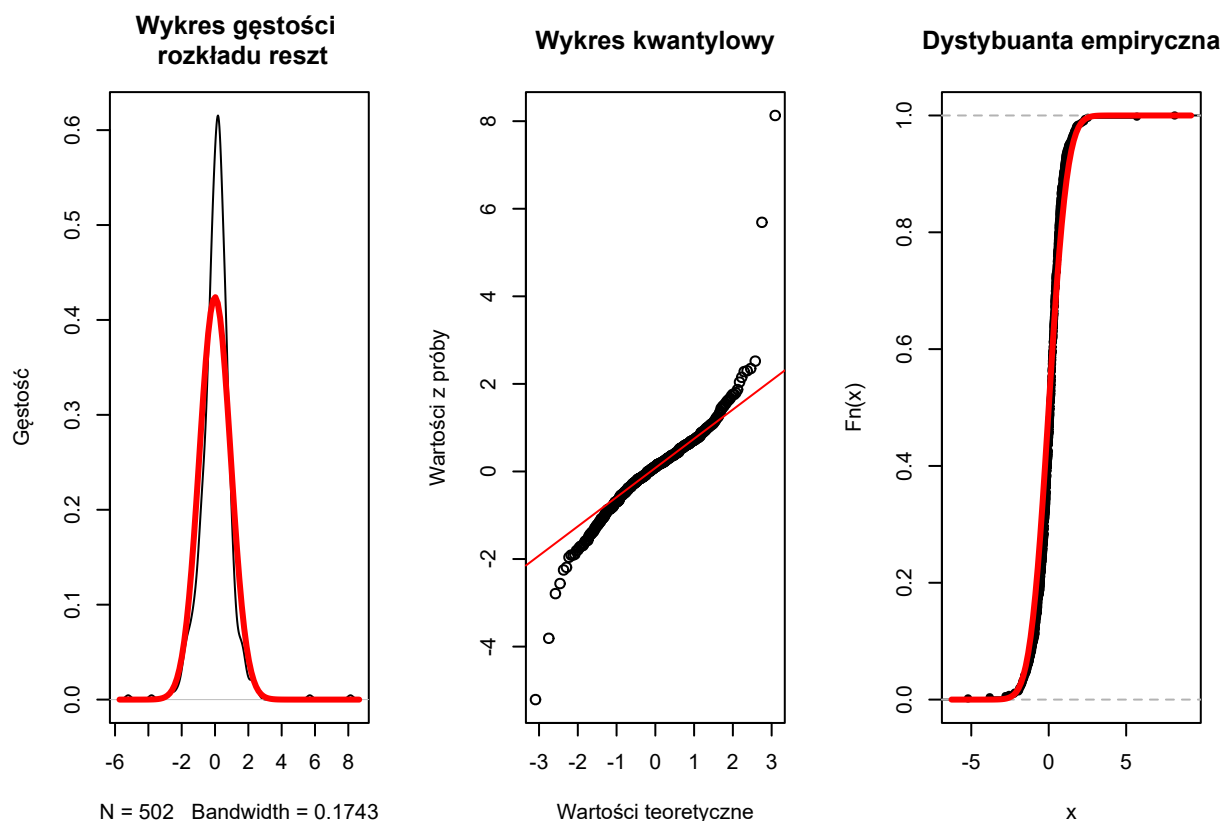
gdzie $\epsilon_t \sim N(0, 0.884)$

Testy na resztach modelu ARIMA

```
par(mfrow = c(1,3))
plot(density(sbox_arima$residuals), main = "Wykres gęstości \n rozkładu reszt",
     ylab = "Gęstość")
curve(dnorm(x,0, sd = sd(sbox_arima$residuals)), col = 2, add = T, lwd = 3)

qqnorm(sbox_arima$residuals, main = "Wykres kwantylowy", xlab = "Wartości teoretyczne",
       ylab = "Wartości z próby")
qqline(sbox_arima$residuals, col = 2)

plot(ecdf(sbox_arima$residuals), main = "Dystybuanta empiryczna")
curve(pnorm(x,0, sd = sd(sbox_arima$residuals)), add = T, lwd = 3, col = 2)
```



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

Z wykresu gęstości rozkładu reszt można wnioskować, że prawdopodobnie ten rozkład charakteryzuje się dodatnią kurtozą, co powoduje jego “wyciągnięcie” w górę, natomiast z wykresu kwantylowego jasno wynika, że rozkład nie może być normalny.

```
kstest <- ks.test(x = sbox_arima$residuals, y = "pnorm",
                  mean = 0, sd = sd(sbox_arima$residuals))$p.value
lillie <- lillie.test(sbox_arima$residuals)$p.value
shapiro <- shapiro.test(sbox_arima$residuals)$p.value
adtest <- ad.test(sbox_arima$residuals)$p.value
pvals <- cbind(c("Test", "Kołmogorowa-Smirnova", "Kołmogorowa-Lillieforsa",
                 "Shapiro-Wilka", "Andersona-Darlinga"),
```

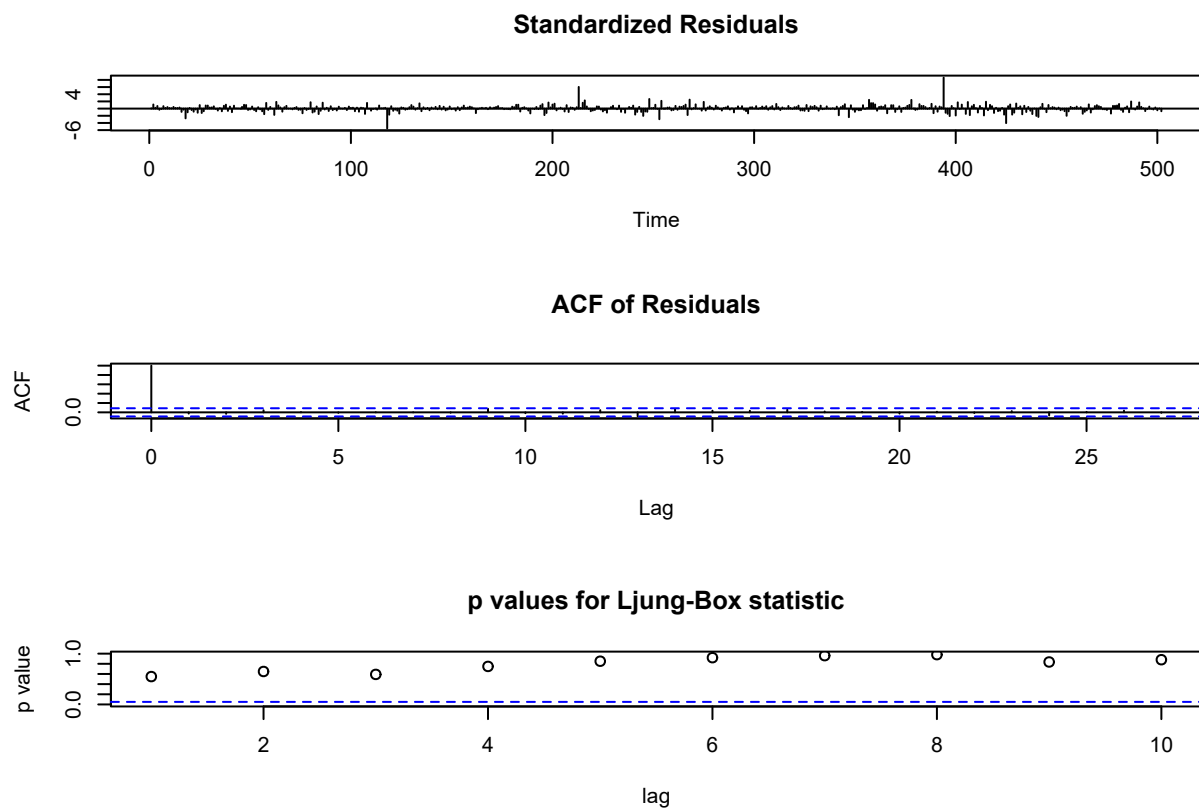
```
c("P-value", kstest, lillie, shapiro, adtest))
```

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

Test	P-value
Kołmogorowa-Smirnova	5.6907304115783e-05
Kołmogorowa-Lillieforsa	1.57181163023406e-10
Shapiro-Wilka	1.37676919077512e-19
Andersona-Darlinga	1.04879735533434e-19

Wszystkie testy jednoznacznie każą odrzucić hipotezę o normalności rozkładu reszt modelu ARIMA, co potwierdza wnioski, które nasuwają się przy analizie wykresów.

```
tsdiag(sbox_arima)
```



Reszty z modelu ARIMA mieszczą się w przedziale od -6 do 8.

Na wykresie ACF dla reszt widać stanowczo, że nie występuje autokorelacja.

P-values dla testu Ljunga-Boxa są wysokie, czasem nawet bliskie 1, co nie daje podstaw do odrzucenia hipotezy o niezależności reszt.

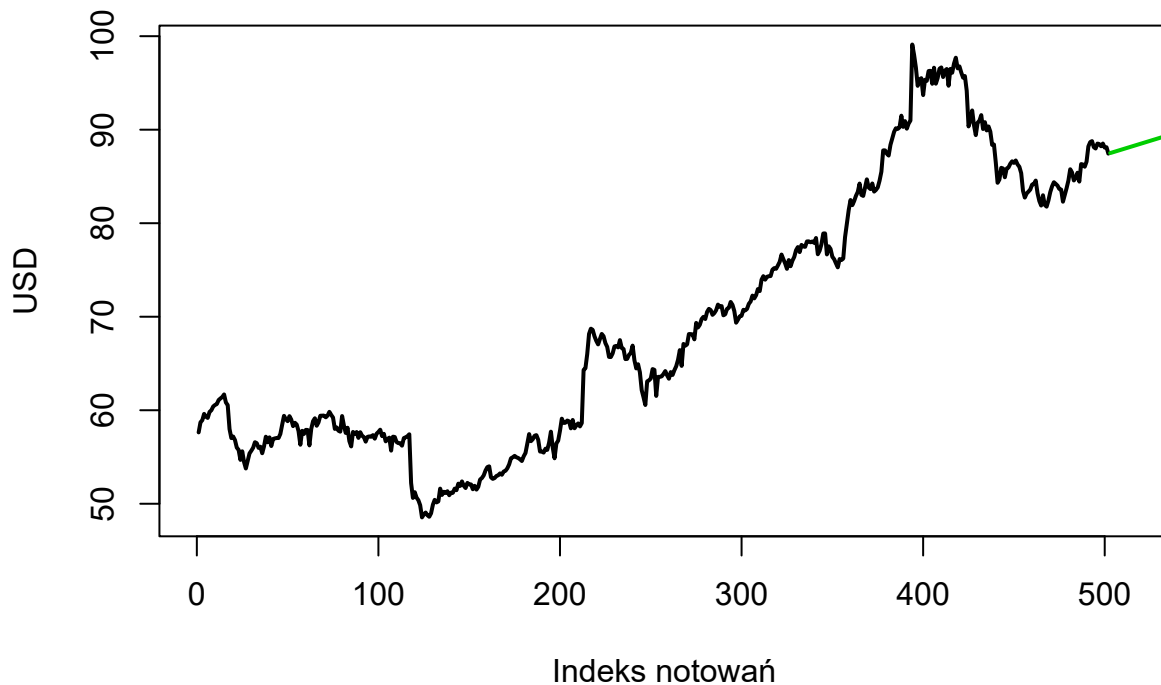
Proгноza

Metoda dryftu

```
progniza <- rwf(sbox, drift = T, h = 30)  
progniza <- as.data.frame(progniza)[,1]
```

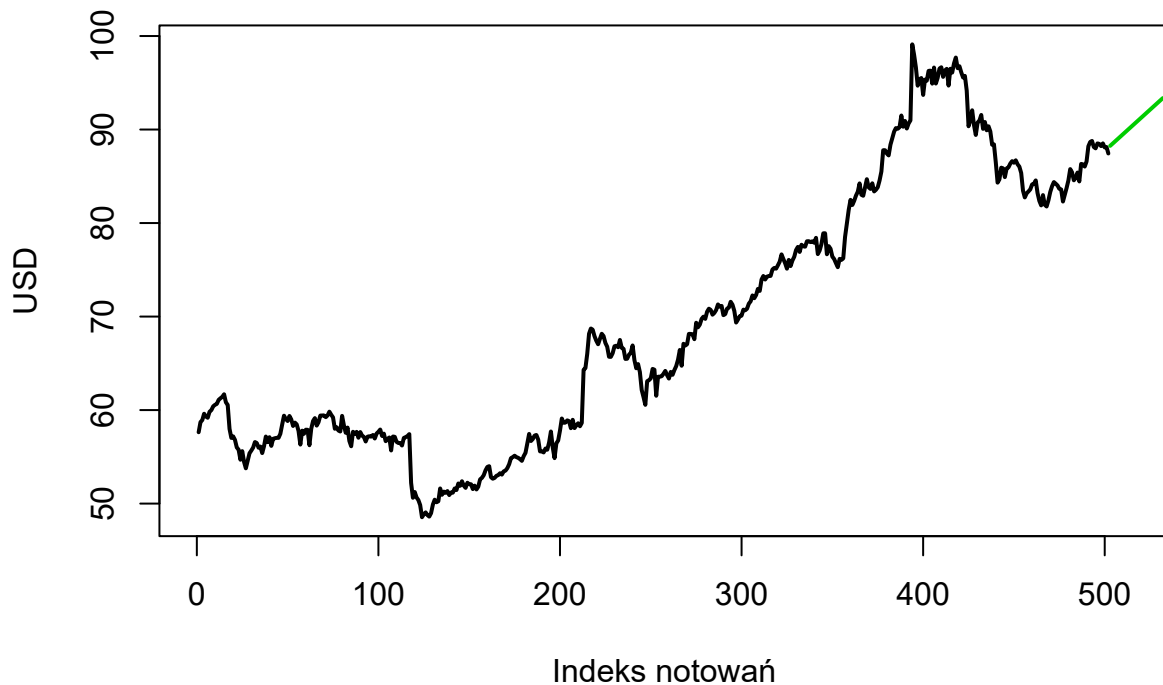
```
plot(sbox, type = "l", main = "Proгноза з całości danych na najbliższe 30 notowań",
xlim = c(0,515), xlab = "Indeks notowań", ylab = "USD", lwd = 2)
t1 <- 503:(503+length(proгноза)-1)
lines(t1, прогноза, col = 3, lwd = 2)
```

Proгноза z całości danych na najbliższe 30 notowań



```
proгноза <- rwf(sbox[480:500], drift = T, h = 30)
proгноза <- as.data.frame(proгноза)[,1]
plot(sbox, type = "l", main = "Proгноза z ostatnich 20 notowań na najbliższe 30",
xlim = c(0,515), xlab = "Indeks notowań", ylab = "USD", lwd = 2)
t1 <- 503:(503+length(proгноза)-1)
lines(t1, прогноза, col = 3, lwd = 2)
```

Prognoza z ostatnich 20 notowań na najbliższe 30

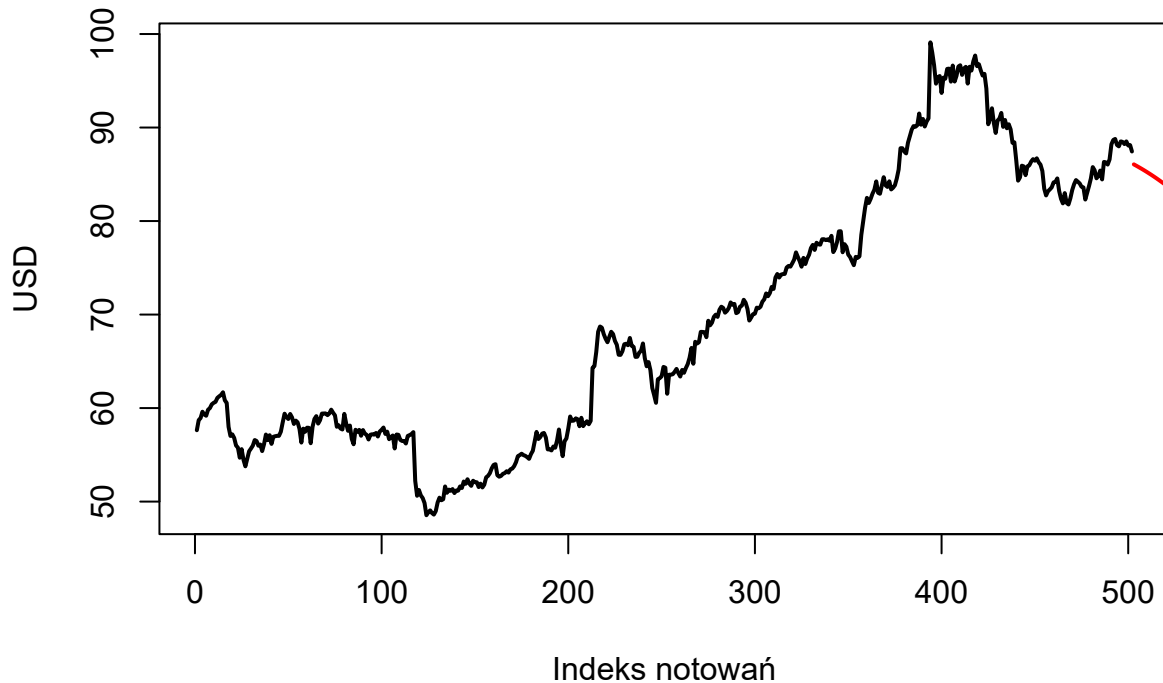


W przypadku danych z firmy Starbucks możemy mieć podejrzenie, że metoda dryftu nie sprawdzi się - pod sam koniec wykresu widać, że notowania zaczynają spadać. Trend wyznaczony przez metodę dryftu nie uwzględnia tego.

Przewidywania z modelu wielomianowego

```
wsp <- mod3$coefficients
funkcja <- function(t){
  wsp[1]+wsp[2]*t+wsp[3]*t^2+wsp[4]*t^3
}
t2 <- 503:520
plot(sbox, type = "l",
     main = "Przewidywane notowania Starbucks \nna podstawie modelu wielomianowego",
     xlab = "Indeks notowań", ylab = "USD", lwd = 2)
lines(t2, funkcja(t2), col = 2, lwd = 2)
```

Przewidywane notowania Starbucks na podstawie modelu wielomianowego



Dla Starbucks zdecydowanie lepiej wygląda prognoza z modelu wielomianowego trzeciego stopnia - wyraźnie widać tu tendencję spadkową.

Podsumowanie analizy notowań Starbucks

Do danych dla firmy Starbucks dobraliśmy model wielomianowy trzeciego stopnia, oraz model ARIMA(0, 1, 0).

Jeżeli chodzi o metody prognozowania, to najlepszą prognozę otrzymaliśmy z modelu wielomianowego.

Podsumowanie i wnioski

Dla firmy Starbucks otrzymaliśmy bardzo podobne wyniki jak dla McDonald's. W obu przypadkach zdecydowaliśmy się na dopasowanie wielomianem trzeciego stopnia, a funkcja `auto.arima` podpowiedziała nam model ARIMA(0, 1, 0).

Podobieństwa mogą być spowodowane faktem, że obie te firmy pochodzą z segmentu gastronomicznego. Być może ich notowania są poddane podobnym wpływom i wykazują podobne zależności.

Główną różnicą był dobór metody prognozowania - w przypadku McDonald's lepiej się sprawdziła metoda dryftu, a w przypadku Starbucks - dopasowanie wielomianem. Jednakże nie jest to kwestia różnic między metodami, a raczej konkretnych danych w tym momencie czasu.