

Analiza i prognozowanie szeregów czasowych- projekt zaliczeniowy MSAD 2018/19

Adam Matuszczyk

2 07 2019

Celem niniejszej pracy jest ocena skutków wprowadzenia reformy, która zaczęła obowiązywać w styczniu 2012, w efekcie działań reformy otrzymaliśmy niższe poziomy refundacji, ale niestety pojawił się problem wywozu dużej grupy leków za granicę na terenie UE, ale również poza strefę unii celnej. W pracy wykorzystałem dane sprzedażowe pewnego produktu, który w efekcie wprowadzenia w życie reformy został wykreślony z listy i powrócił na nią w marcu 2012

W pracy wykorzystałem materiały z zajęć prowadzonych przez dra Tomasza Wójtowicza w ramach MSAD 2018/2019, a także publikację Adam Zagdański, Artur Suchwałko "Analiza i prognozowanie szeregów czasowych" oraz Jared P. Lander "R dla każdego"

```
library("tidyverse")
library("lubridate")
library("skimr")
library("data.table")
library("forecast")
library("lattice")
```

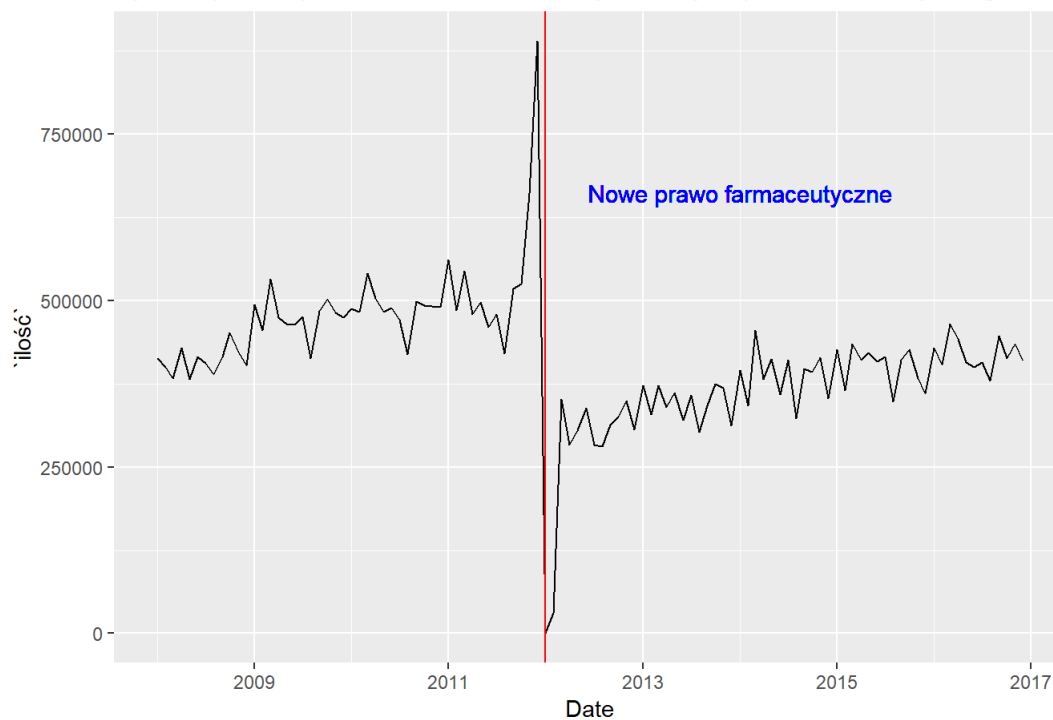
```
dane <- read.table("d:/AGH/Analiza i Prognozowanie Szeregów Czasowych/testy.csv", sep= ";", header = TRUE)
#dane  %>% skim()
```

```
dane %>% mutate(Date = ymd(Data, truncated =1)) %>% #ustawienie klasy "date"
select(-Data) -> dane
```

Wykres sprzedaży za lata 2018-2017 analizowanego produktu

```
dane %>%
  ggplot(aes(Date, ilość)) +
  geom_line() +
  scale_x_date(date_breaks = "2 years", date_labels = "%Y") +
  geom_vline(xintercept = ymd("2012-01-01"), col = "red") +
  geom_text(aes(x = dane$Date[73], y = dane$ilość[47], label = "Nowe prawo farmaceutyczne"), color = "blue", size
= 4) +
  labs(title = "wykres sprzedaży wraz z zaznaczeniem wejścia w życie prawa farmaceutycznego") +
  theme(plot.title=element_text(size=12))
```

wykres sprzedaży wraz z zaznaczeniem wejścia w życie prawa farmaceutycznego



```
dane %>% mutate(Month = as.factor(month(Date)), Year = year(Date)) -> dane
head(dane) %>% kable()
```

ilość	Date	Month	Year
413194	2008-01-01	1	2008
400442	2008-02-01	2	2008
383625	2008-03-01	3	2008
429479	2008-04-01	4	2008
382434	2008-05-01	5	2008
415443	2008-06-01	6	2008

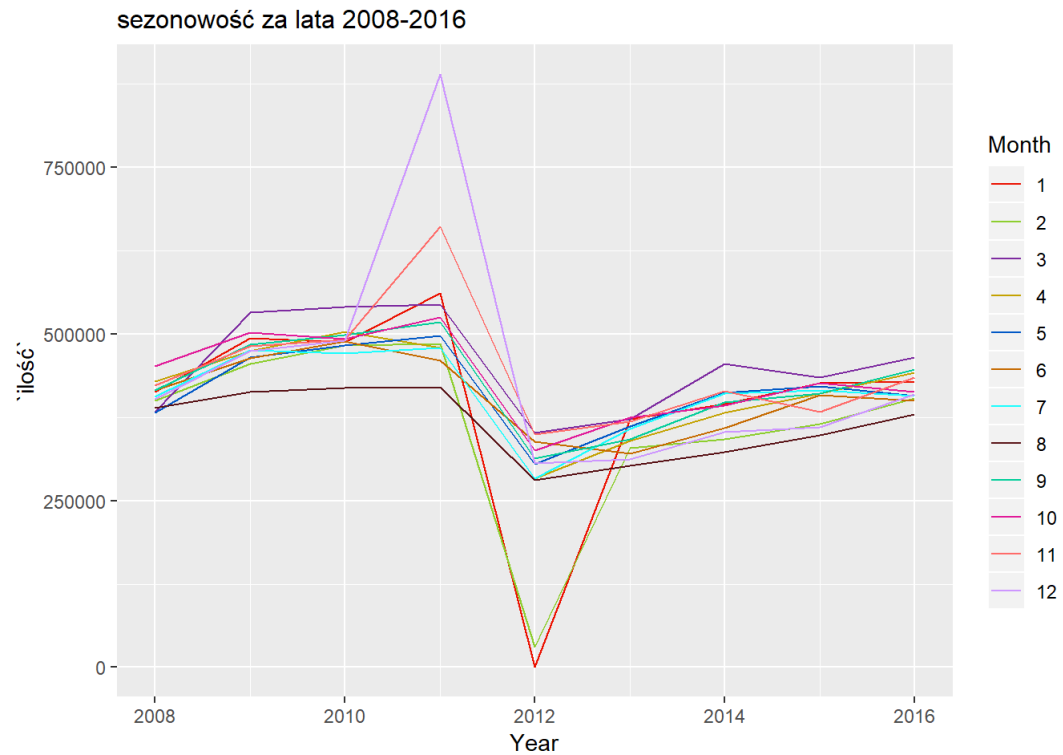
Składniki szeregu, dekompozycja

zastosowane "kody" kolorów pochodzą ze strony:

<http://tools.medialab.sciences-po.fr/iwanthue/>

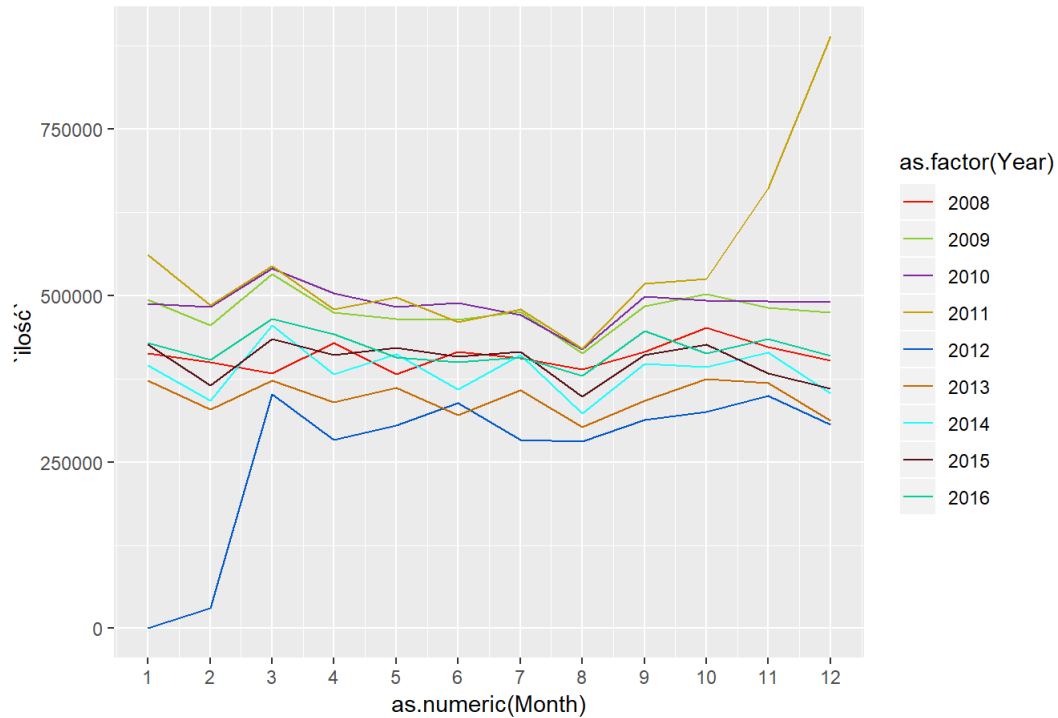
```
cols <- c("#eb1100", "#8dce2e", "#7d2aa0", "#c4a200", "#0158c6", "#c66a00", "#1bfeff", "#581216", "#02ce9a", "#e11a99",
          "#ff6b69", "#cd96ff")
```

```
dane %>%
  ggplot(aes(Year, ilość, col = Month)) +
  geom_line() +
  scale_color_manual(values = cols) +
  labs(title = "sezonowość za lata 2008-2016") +
  theme(plot.title=element_text(size=12))
```



```
dane %>%
  ggplot(aes(as.numeric(Month), ilość, col = as.factor(Year))) +
  geom_line(show.legend = TRUE) +
  scale_color_manual(values = cols) +
  scale_x_continuous(breaks = 1:12) +
  labs(title = "miesięczne zmiany w kolejnych latach 2008-2016") +
  theme(plot.title=element_text(size=12))
```

miesięczne zmiany w kolejnych latach 2008-2016



Jak widać wykresy są nieprzejrzyste, ze względu na wejście reformy w życie, rok 2011 jest zaburzony przez napompowanie rynku przez koncern co uwidatniło się w ostatnich trzech miesiącach. Początek 2012 również ze względu na usunięcie z listy i powrót w kolejnej publikacji odstaje od reszty. Postanowiłem wprowadzić filtry do analizy aby w dalszej pracy uzyskać czytelną formę.

```
#wprowadzenie filtrów "Before", "Before2", "After"

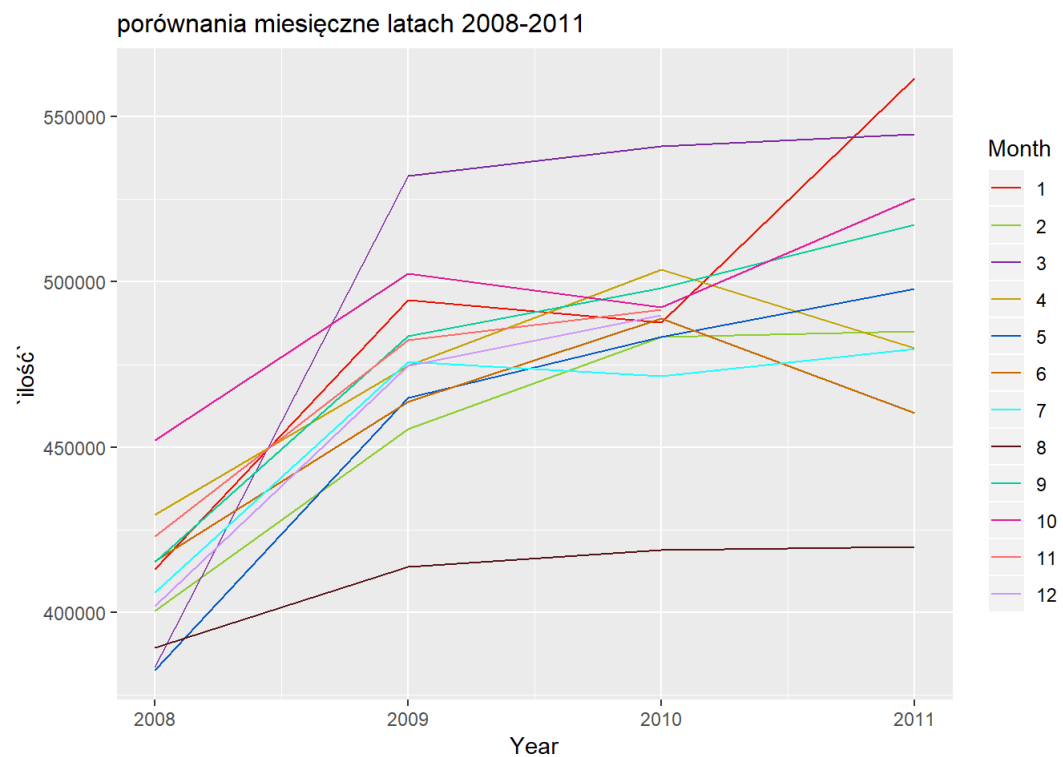
dane %>% mutate(Before = Date < "2011-11-01",
                Before2 = Date < "2011-11-01" & Date >= "2009-01-01", After = Date > "2012-03-01") -> dane
tail(dane) %>% kable()
```

	ilość	Date	Month	Year	Before	Before2	After
103	407763	2016-07-01	7	2016	FALSE	FALSE	TRUE
104	379531	2016-08-01	8	2016	FALSE	FALSE	TRUE
105	446926	2016-09-01	9	2016	FALSE	FALSE	TRUE
106	412795	2016-10-01	10	2016	FALSE	FALSE	TRUE
107	435520	2016-11-01	11	2016	FALSE	FALSE	TRUE

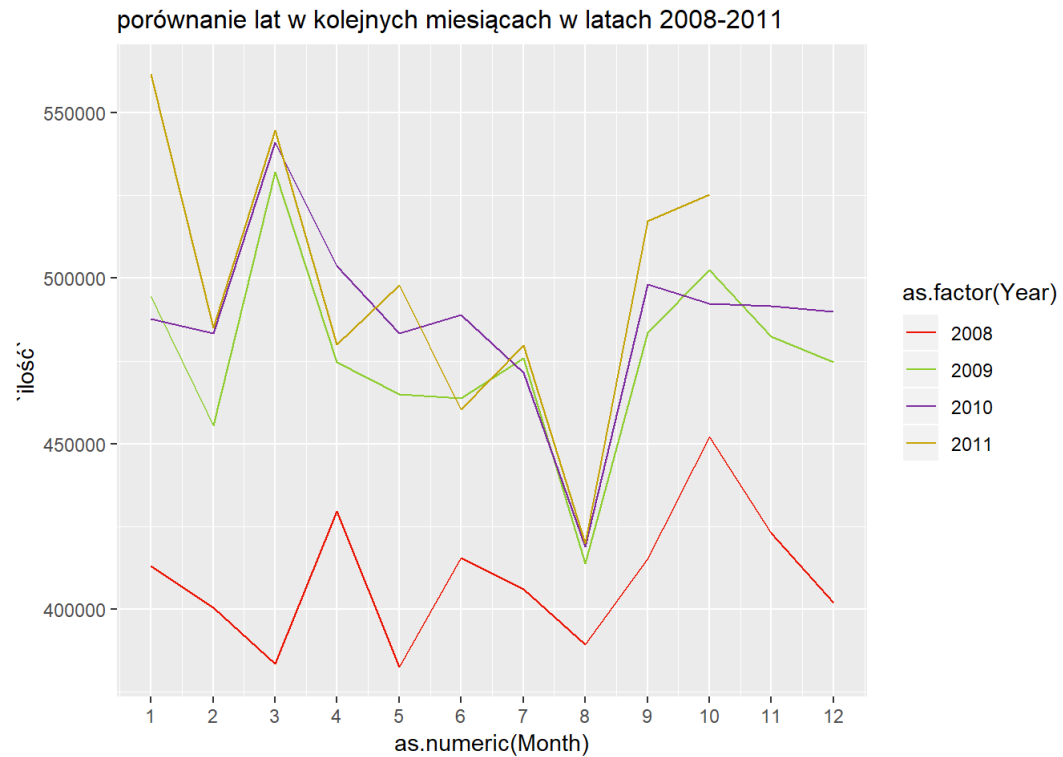
	ilość	Date	Month	Year	Before	Before2	After
108	409834	2016-12-01	12	2016	FALSE	FALSE	TRUE

W pierwszej części pracy będę korzystał z filtrów "Before" oraz "Before2"

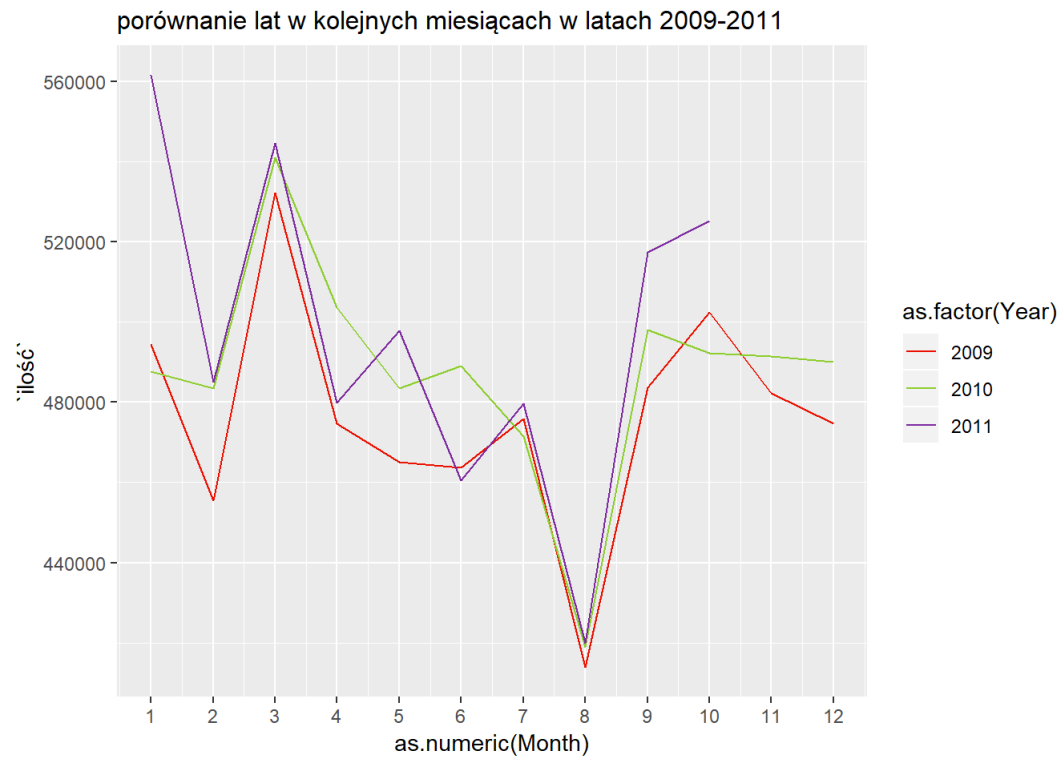
```
dane %>%
  filter(Before) %>%
  ggplot(aes(Year, ilość, col = Month)) +
  geom_line() +
  scale_color_manual(values = cols) +
  labs(title = "porównania miesięczne latach 2008-2011") +
  theme(plot.title=element_text(size=12))
```



```
dane %>%
  filter(Before) %>%
  ggplot(aes(as.numeric(Month), ilość, col = as.factor(Year))) +
  geom_line(show.legend = TRUE) +
  scale_color_manual(values = cols) +
  scale_x_continuous(breaks = 1:12) +
  labs(title = "porównanie lat w kolejnych miesiącach w latach 2008-2011") +
  theme(plot.title=element_text(size=12))
```



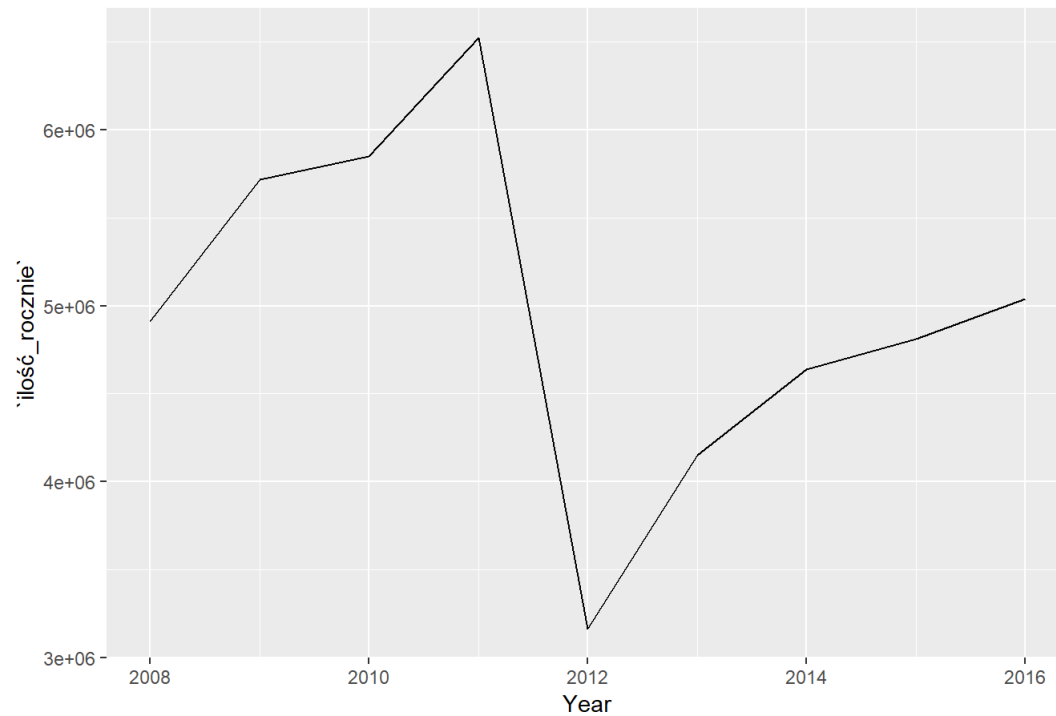
```
dane %>%
  filter(Before2) %>%
  ggplot(aes(as.numeric(Month), ilość, col = as.factor(Year))) +
  geom_line(show.legend = TRUE) +
  scale_color_manual(values = cols) +
  scale_x_continuous(breaks = 1:12) +
  labs(title = "porównanie lat w kolejnych miesiącach w latach 2009-2011") +
  theme(plot.title=element_text(size=12))
```



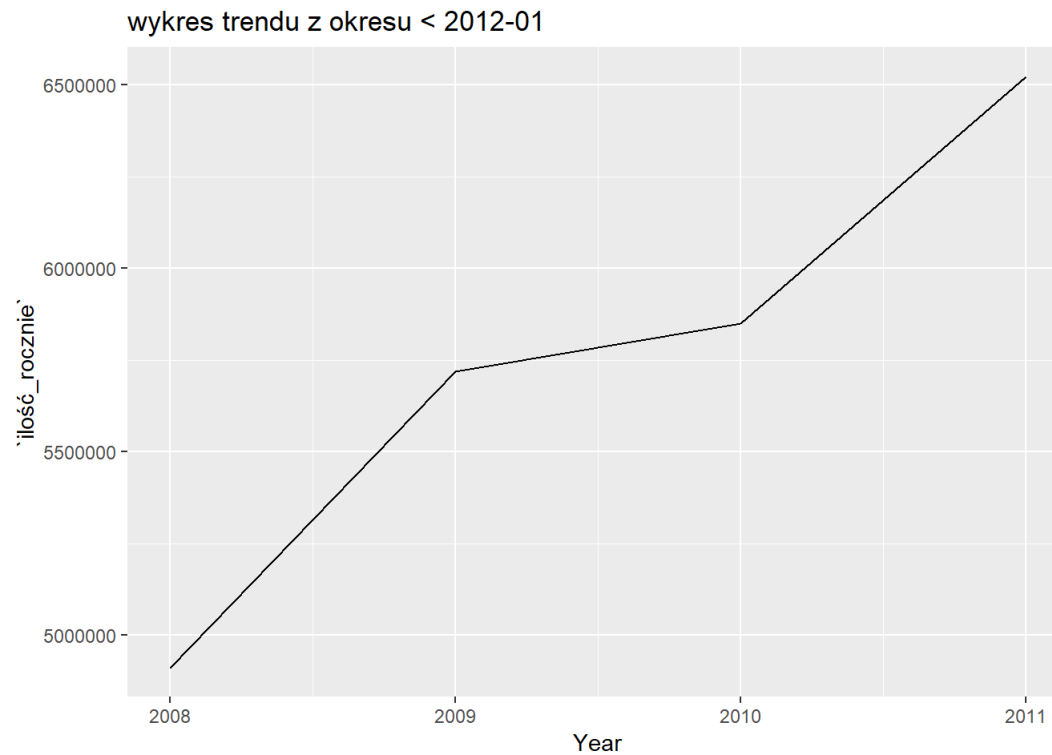
Wykresy stały się bardziej czytelne, widać na nich sezonowość oraz trend wzrostowy w kolejnych latach. Niestety ze względu na zakłócenie sezonowości przez rok 2008, postanowiłem do dalszej analizy wykorzystać obydwa filtry.

```
dane%>%
  group_by(Year) %>%
  summarise(ilość_rocznie = sum(ilość)) %>%
  ggplot(aes(Year, ilość_rocznie))+
  geom_line() +
  labs(title = "wykres trendu z okresu 2008-01 do 2016-12")
```

wykres trendu z okresu 2008-01 do 2016-12



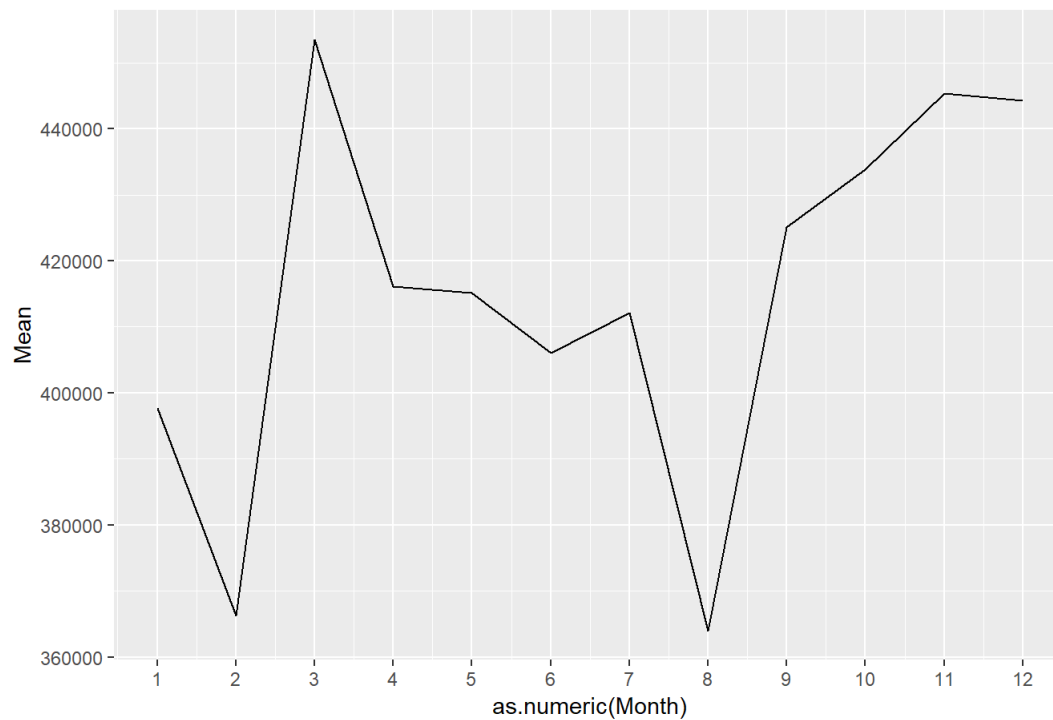
```
dane %>%  
  filter(Year < 2012) %>%  
  group_by(Year) %>%  
  summarise(ilość_rocznie = sum(ilość)) %>%  
  ggplot(aes(Year, ilość_rocznie)) +  
  geom_line() +  
  labs(title = "wykres trendu z okresu < 2012-01")
```

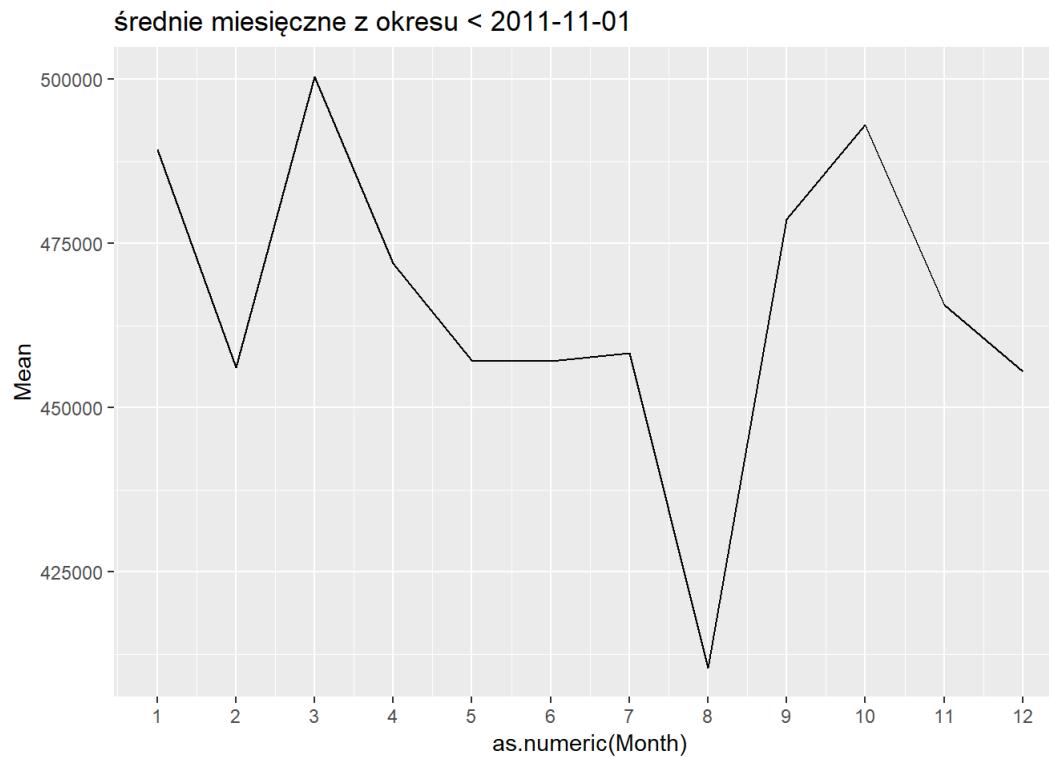
Na dwóch wykresach widać trend wzrostowy na przeszczepi lat. Jak również widać, że należy traktować wykresy jako dwa nie do końca zależne od siebie

```
dane %>%
  group_by(Month) %>%
  summarise(Mean = mean(ilość)) %>%
  ggplot(aes(as.numeric(Month), Mean)) +
  geom_line() +
  scale_x_continuous(breaks = 1:12) +
  labs(title = "średnie miesięczne z okresu 2008-01 do 2016-12")
```

średnie miesięczne z okresu 2008-01 do 2016-12



```
dane %>%  
  filter(Before) %>%  
  group_by(Month) %>%  
  summarise(Mean = mean(ilość)) %>%  
  ggplot(aes(as.numeric(Month), Mean)) +  
  geom_line() +  
  scale_x_continuous(breaks = 1:12) +  
  labs(title = "średnie miesięczne z okresu < 2011-11-01")
```



Potwierdzenie sezonowości łącznej oraz z filtrem “Before”. Widać, że marzec i sierpień są charakterystyczne. Sprzedaż w pierwszym kwartale roku wzrasta później spada i w okresie wakacyjnym osiąga minimum. W kolejnych miesiącach wzrasta z zatrzymaniem grudniowym co pewnie jest związane ze stosunkowo krótkim okresem handlowym w Polsce

#####Regresja#####

```
dane %>% mutate(Time = row_number()) -> dane # dodanie kolumny czas(lp)
model <- lm(ilość ~ Time + Month, filter(dane, Before))
summary(model)
```

```
##
## Call:
## lm(formula = ilość ~ Time + Month, data = filter(dane, Before))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -74097 -14008   1315   17661  46016
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 444212.1    14726.7   30.164 < 2e-16 ***
## Time         2372.5      306.7    7.735 6.56e-09 ***
## Month2       -35529.8    19128.9   -1.857  0.0722 .
## Month3        6392.4    19136.3    0.334  0.7405
## Month4       -24430.8    19148.6   -1.276  0.2109
## Month5       -41580.4    19165.7   -2.170  0.0373 *
## Month6       -44014.2    19187.8   -2.294  0.0283 *
## Month7       -45249.7    19214.8   -2.355  0.0246 *
## Month8       -95434.0    19246.6   -4.958 2.08e-05 ***
## Month9       -29614.8    19283.2   -1.536  0.1341
## Month10      -17571.5    19324.6   -0.909  0.3698
## Month11      -33133.7    20695.3   -1.601  0.1189
## Month12      -45607.6    20715.8   -2.202  0.0348 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27050 on 33 degrees of freedom
## Multiple R-squared:  0.7388, Adjusted R-squared:  0.6439
## F-statistic: 7.78 on 12 and 33 DF, p-value: 1.355e-06
```

```
dane$Sierpien <- as.numeric(dane$Month == 8)
model2 <- lm(ilość ~ Time + Sierpien, filter(dane, Before))
summary(model2)
```

```
##
## Call:
## lm(formula = ilość ~ Time + Sierpien, data = filter(dane, Before))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -54101 -19698   1480   13431   79351
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 418611.3      8793.6  47.604 < 2e-16 ***
## Time         2283.1        324.7   7.032 1.16e-08 ***
## Sierpien     -67507.7     15298.0  -4.413 6.74e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29190 on 43 degrees of freedom
## Multiple R-squared:  0.6038, Adjusted R-squared:  0.5854
## F-statistic: 32.77 on 2 and 43 DF,  p-value: 2.265e-09
```

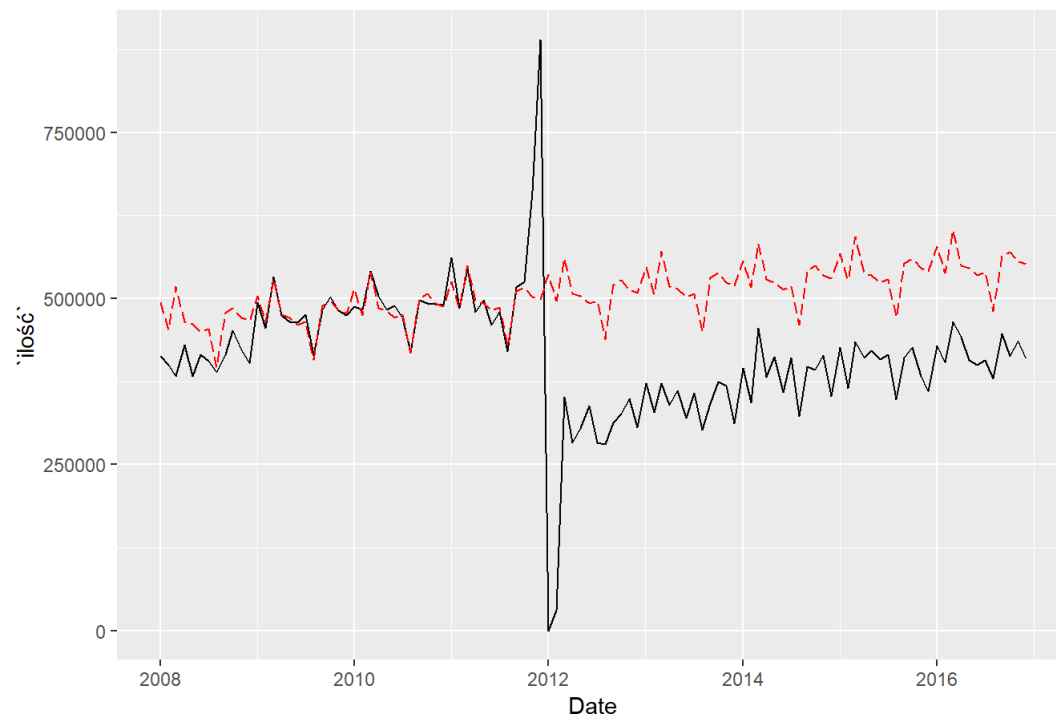
```
model3 <- lm(ilość ~ Time + Month, filter(dane, Before2)) #główny
summary(model3)
```

```
##
## Call:
## lm(formula = ilość ~ Time + Month, data = filter(dane, Before2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26872  -6210    245    6130   36479
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  492540.1    10748.7  45.823  < 2e-16 ***
## Time           884.6       266.0   3.326  0.003210 **
## Month2       -40843.6    11945.8  -3.419  0.002580 **
## Month3        22937.1    11954.7   1.919  0.068722 .
## Month4       -31166.5    11969.5  -2.604  0.016576 *
## Month5       -36072.2    11990.2  -3.008  0.006691 **
## Month6       -48041.4    12016.7  -3.998  0.000653 ***
## Month7       -44306.7    12049.0  -3.677  0.001402 **
## Month8      -103334.3    12087.1  -8.549  2.8e-08 ***
## Month9       -21981.6    12131.0  -1.812  0.084307 .
## Month10      -15884.3    12180.4  -1.304  0.206320
## Month11      -31266.2    13394.9  -2.334  0.029599 *
## Month12      -36730.3    13418.6  -2.737  0.012344 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14630 on 21 degrees of freedom
## Multiple R-squared:  0.8725, Adjusted R-squared:  0.7996
## F-statistic: 11.97 on 12 and 21 DF,  p-value: 7.957e-07
```

W “model1” zastosowałem najprostsze rozwiązanie i usunąłem trend wraz z sezonowością. Uzyskałem 74% dopasowanie modelu. “model2” jest pozbawiony sierpnia ze względu na wysoką zależność z poprzednich obliczeń, niestety spadło dopasowanie. W ostatnich obliczeniach wyeliminowałem rok 2008 co poprawiło dopasowanie (87%), ale niestety zmniejszyłem ilość danych. Pomimo tego postanowiłem do predykcji zastosować “model3”

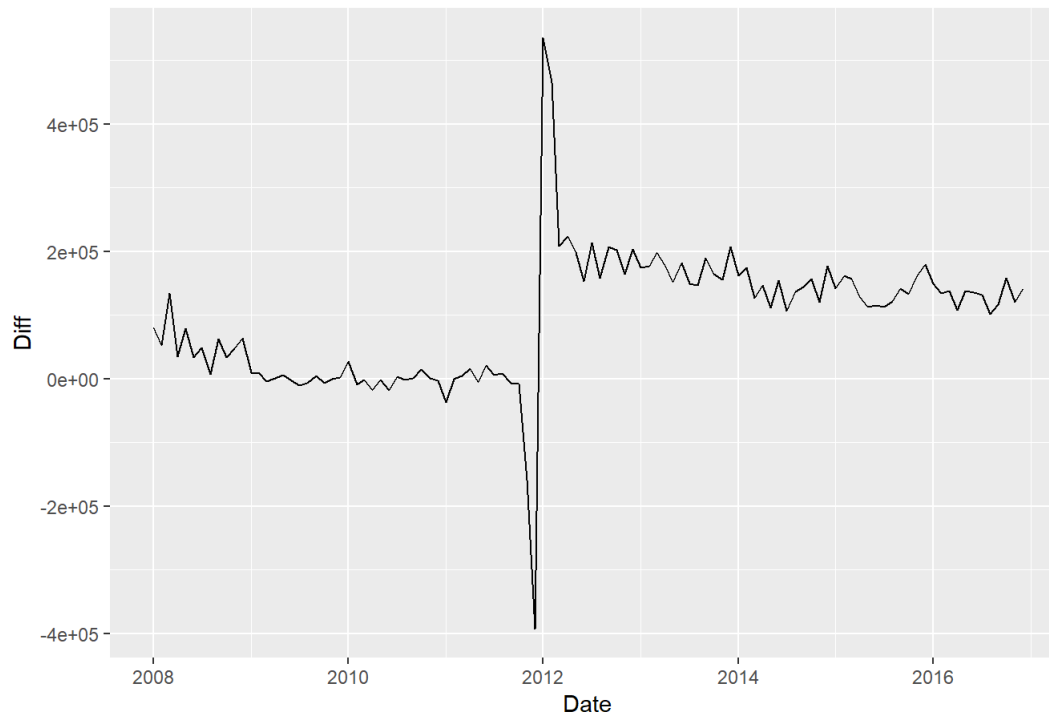
```
dane$Prediction <- predict(model3, dane) #dodanie kolumny Predykcja(Prediction)
ggplot(dane) +
  geom_line(aes(Date, ilość)) +
  geom_line(aes(Date, Prediction), col = "red", linetype = "longdash") +
  labs(title = "sprzedaż 2008-2016 z naniesioną predykcją")
```

sprzedaż 2008-2016 z naniesioną predykcją



```
dane %>% mutate(Diff = Prediction - ilość) -> dane # dodanie kolumny Diff(różnica= Predykcja-ilość)
dane %>%
  ggplot(aes(Date, Diff)) +
  geom_line() +
  labs(title = "podsumowanie modelu: Diff = Prediction - ilość ")
```

podsumowanie modelu: Diff = Prediction - ilość



```
dane %>% filter(!Before) %>% summarise(1 - sum(ilość) / sum(Prediction)) %>% kable()
```

1 - sum(ilość)/sum(Prediction)

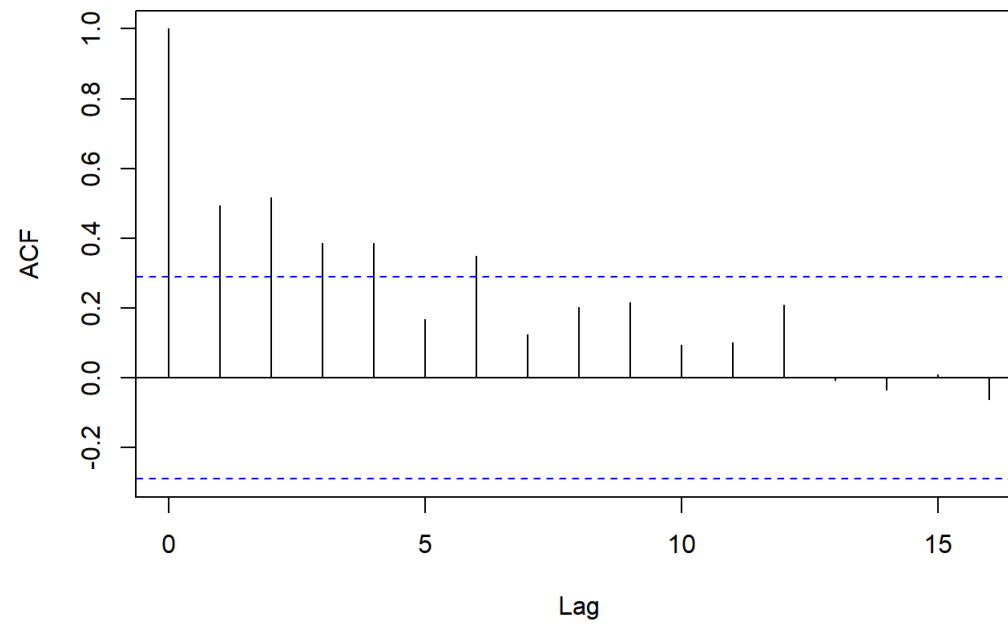
0.2874484

Z zastosowania "model3" wynika, że całkowita sprzedaż spadła o ok. 29% lub można po przeliczeniu refundacji chwalić się oszczędnościami dla budżetu państwa

#####Autokorelacja#####

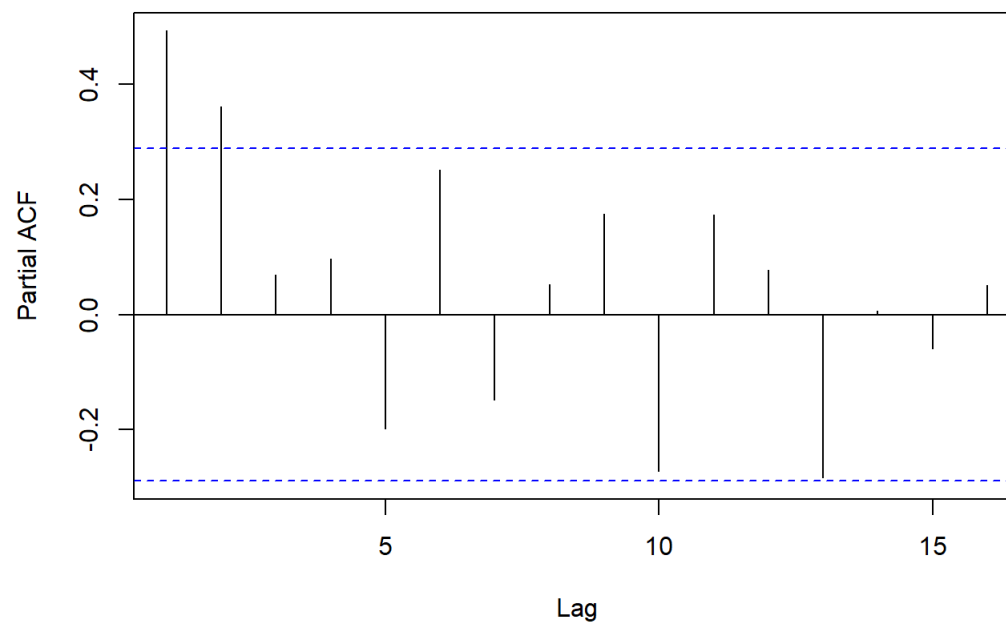
```
dane_before <- filter(dane, Before) # wyeliminować I rzędu i II jest znacznie mniej istotna  
acf(dane_before$ilość)
```


Series dane_before\$ilość



```
pacf(dane_before$ilość)
```

Series dane_before\$ilość

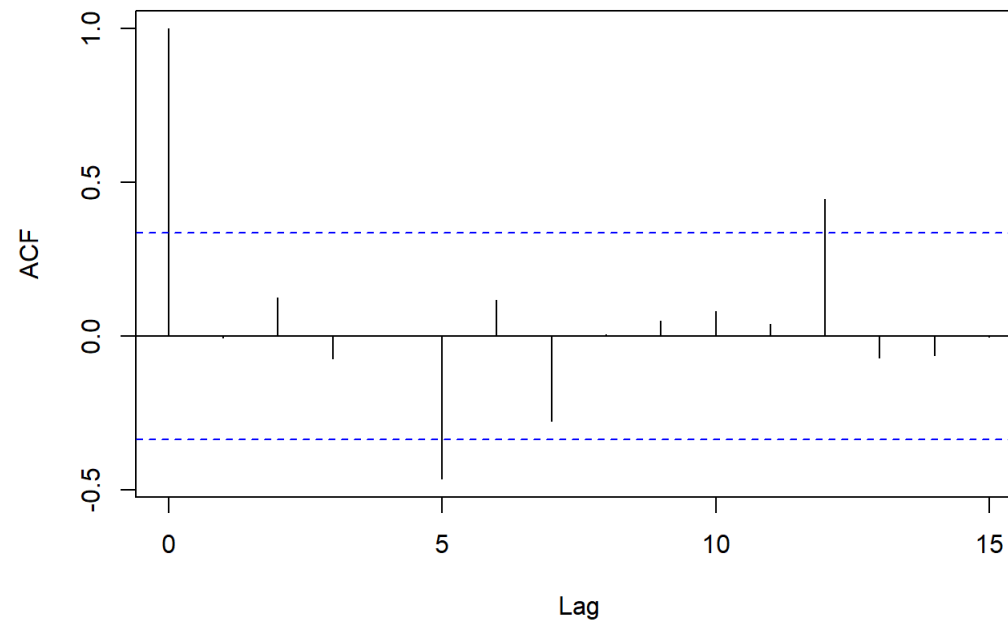


```
pacf(dane_before$ilość, plot = FALSE)
```

```
##
## Partial autocorrelations of series 'dane_before$ilość', by lag
##
##      1      2      3      4      5      6      7      8      9     10
## 0.493  0.362  0.069  0.097 -0.198  0.252 -0.148  0.052  0.175 -0.272
##      11     12     13     14     15     16
## 0.174  0.078 -0.283  0.006 -0.059  0.051
```

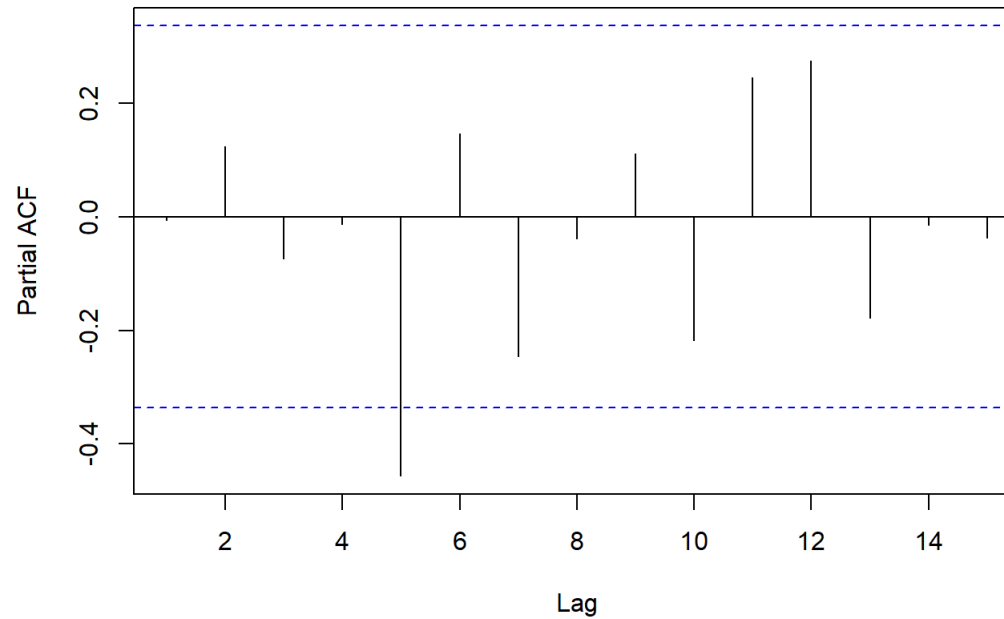
```
dane_before2 <- filter(dane, Before2) # wyeliminować I rzędu i II jest znacznie mniej istotna
acf(dane_before2$ilość)
```

Series dane_before2\$ilość



```
pacf(dane_before2$ilość)
```

Series dane_before2\$ilość

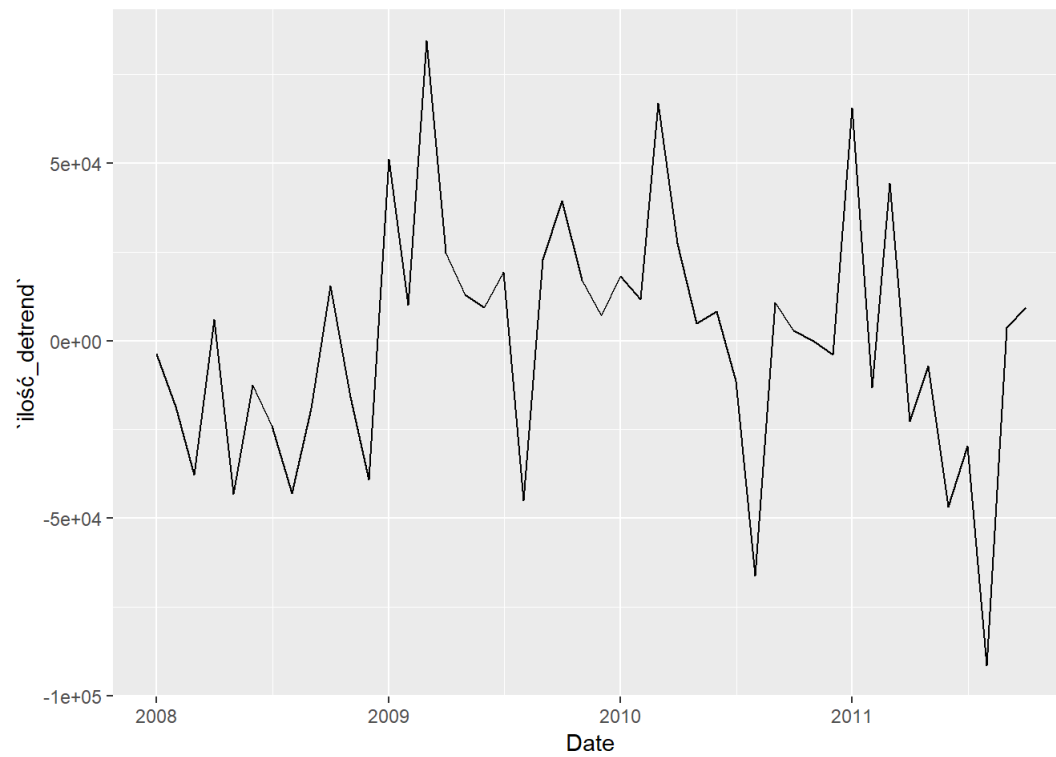


```
pacf(dane_before2$ilość, plot = FALSE)
```

```
##  
## Partial autocorrelations of series 'dane_before2$ilość', by lag  
##  
##      1      2      3      4      5      6      7      8      9     10  
## -0.005  0.124 -0.073 -0.013 -0.456  0.146 -0.245 -0.038  0.112 -0.217  
##      11     12     13     14     15  
##  0.245  0.275 -0.177 -0.014 -0.038
```

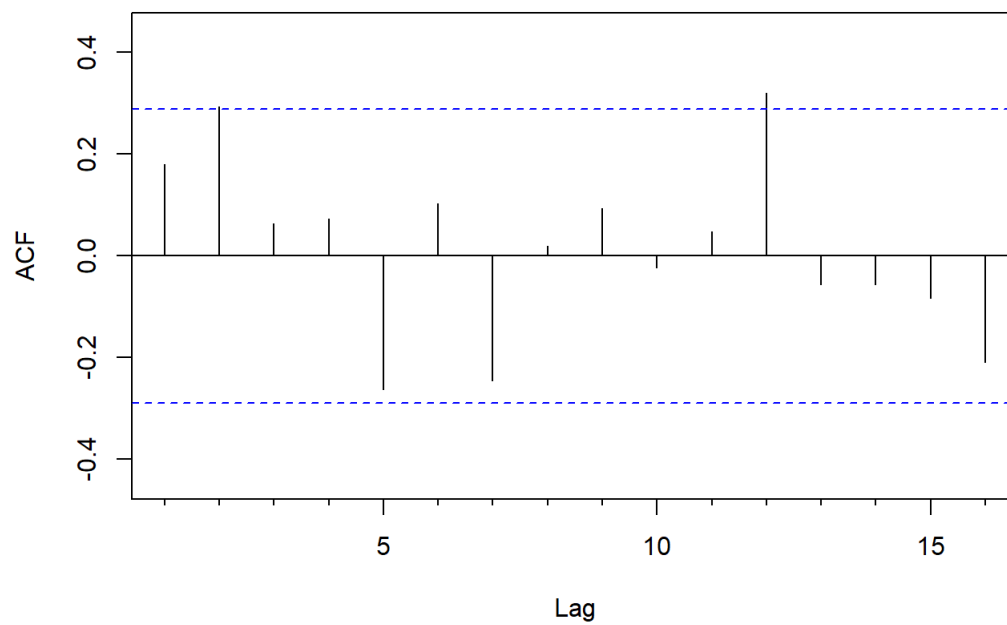
Usunięcie trendu

```
dane_before$ilość_detrend <- lm(ilość ~ Time, dane_before) %>% resid()  
ggplot(dane_before, aes(Date, ilość_detrend)) + geom_line()
```



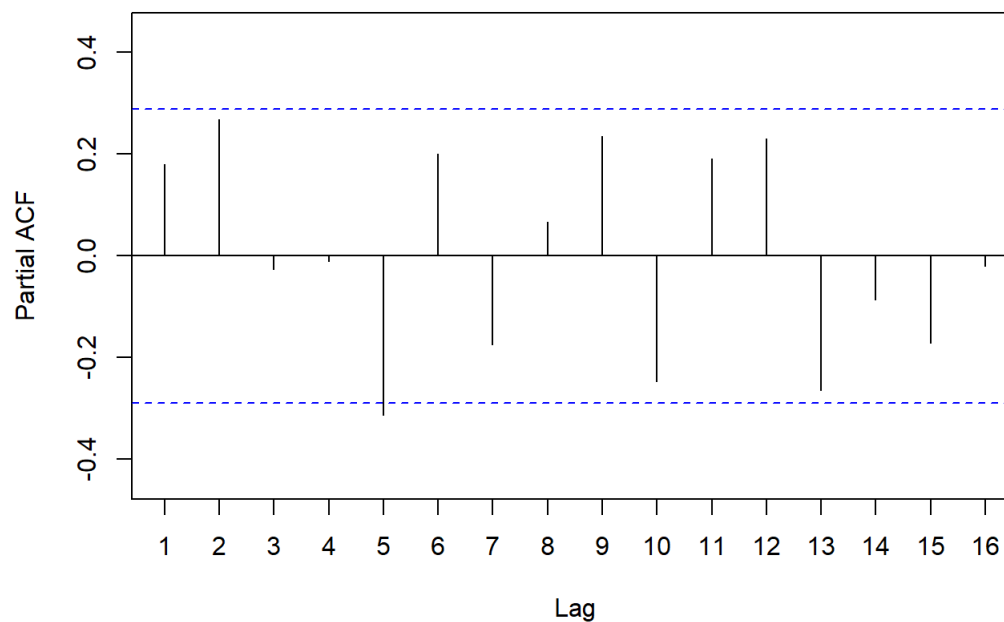
```
Acf(dane_before$ilość_detrend)
```

Series dane_before\$ilość_detrend



`Pacf(dane_before$ilość_detrend)` # wykresy sugerują dominujący wpływ trendu, nawet do pominięcia okresowości

Series dane_before\$ilość_detrend



```
Pacf(dane_before$ilość_detrend, plot = FALSE)
```

```
##
## Partial autocorrelations of series 'dane_before$ilość_detrend', by lag
##
##      1      2      3      4      5      6      7      8      9     10
## 0.181  0.269 -0.027 -0.011 -0.313  0.200 -0.175  0.066  0.235 -0.246
##      11     12     13     14     15     16
## 0.191  0.230 -0.264 -0.087 -0.171 -0.020
```

Niestety po usunięciu trendu okazało się, że znikły najważniejsze zależności. Życie weryfikuje zapędy udoskonalania wykresów stąd należałoby zamknąć dalszą analizę

usunięcie sezonowosci

```
#dane_before$ilość_detrend_deseason <- lm(ilość ~ Time + Month, dane_before) %>% resid()
#ggplot(dane_before, aes(Date, ilość_detrend_deseason)) + geom_line()
#acf(dane_before$ilość_detrend_deseason)
#pacf(dane_before$ilość_detrend_deseason)
#pacf(dane_before$ilość_detrend_deseason, plot= FALSE)
```

Autoregresja

usunięcie trendu rzędu 1

```
#@dane_before$ilość_lag1 <- lag(dane_before$ilość, n = 1)
#model4 <- lm(ilość ~ Time + Month + ilość_lag1 , dane_before)
#summary(model4)

#dane_before$ilość_detrend_deseason_decorr <- c(NA, resid(model4))
#ggplot(dane_before, aes(Date, ilość_detrend_deseason_decorr)) + geom_line()
#acf(dane_before$ilość_detrend_deseason_decorr[-1])
#pacf(dane_before$ilość_detrend_deseason_decorr[-1])
```

Predykacja dla AR(model4)

```
#set.seed(3)
#pred <- predict(model4, dane_before)
#dane %>%
#   mutate(Prediction4 = ifelse(Before, pred, NA),
#          ilość_lag1 = lag(ilość, n = 1)) -> dane

#start <- which(dane$Date == "2011-01-01")
#end <- nrow(dane)
#for (i in start:end) {
#   pred <- predict(model4, newdata = slice(dane, i))
#   dane$Prediction4[i] <- pred
#   if (i < end) dane$ilość_lag1[i + 1] <- pred
#}

#ggplot(dane) +
#   geom_line(aes(Date, ilość)) +
#   geom_line(aes(Date, Prediction4), col = "red", linetype = "longdash")
#dane %>% mutate(Diff4 = Prediction4 - ilość) -> dane
#dane %>% filter(!Before) %>% summarise(1 - sum(ilość) / sum(Prediction4)) %>% kable()
```

```
#model5 <- lm(ilość ~ Time + Month, filter(dane, After))
#summary(model5)

#dane$Prediction5 <- predict(model5, dane)#dodanie kolumny Predykcja(Prediction)
#ggplot(dane) +
#   geom_line(aes(Date, ilość)) +
#   geom_line(aes(Date, Prediction), col = "red", linetype = "longdash")+
#   geom_line(aes(Date, Prediction5), col = "blue", linetype = "longdash")
```

Analiza wykresu po reformie

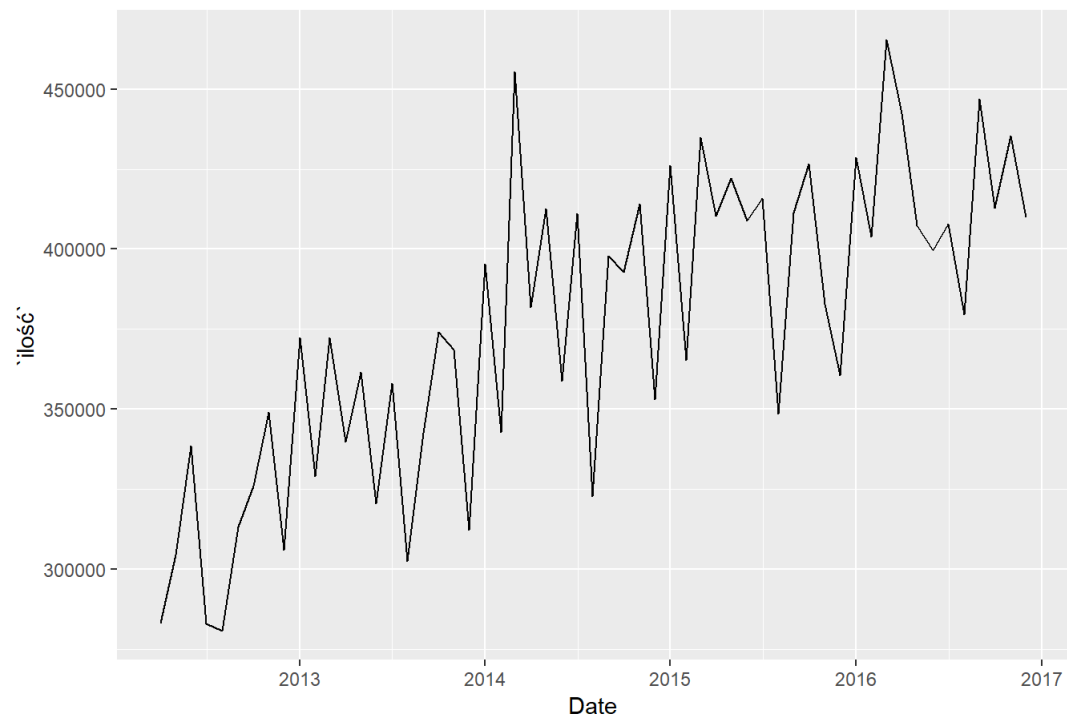
Postanowiłem sprawdzić przy pomocy dostępnych danych po wejściu w życie reformy czy jest możliwe i kiedy sprzedaż miałaby szansę zrównać się z danymi uzyskanymi przy pomocy prdykcji “model3”


```
dane2012 <- dane[which(dane$Year >= 2012),] %>% filter(After)
head(dane2012)
```

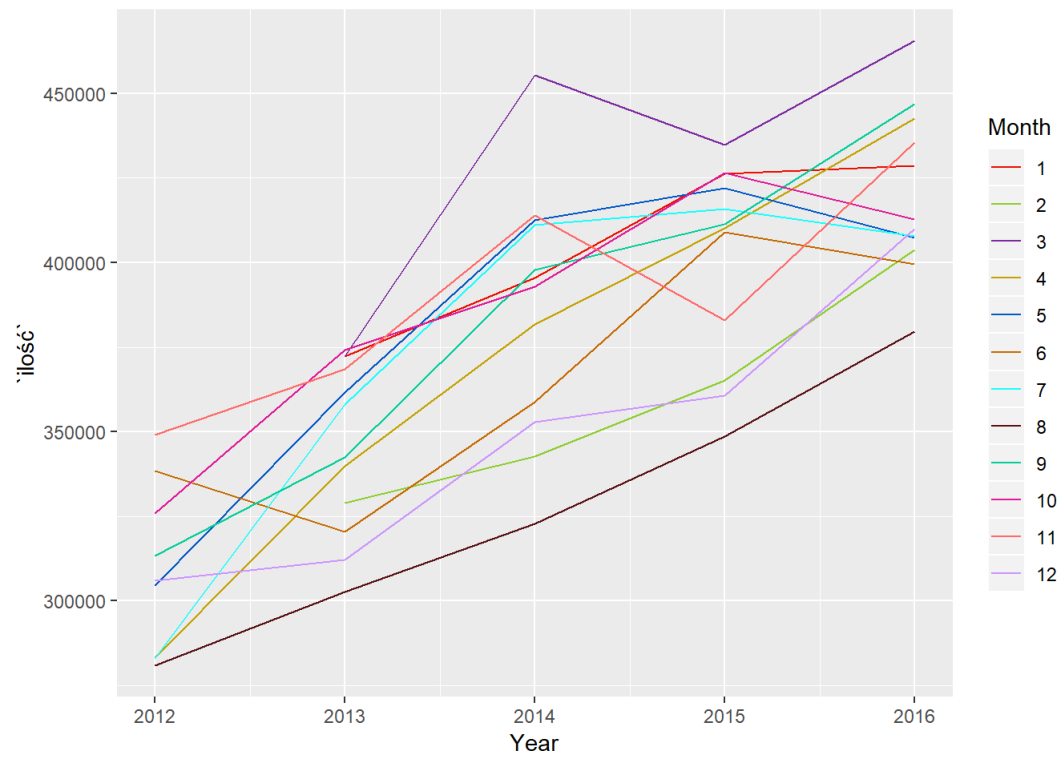
```
##      ilość      Date Month Year Before Before2 After Time Sierpien
## 1 283191 2012-04-01      4 2012  FALSE  FALSE  TRUE   52         0
## 2 304471 2012-05-01      5 2012  FALSE  FALSE  TRUE   53         0
## 3 338475 2012-06-01      6 2012  FALSE  FALSE  TRUE   54         0
## 4 283010 2012-07-01      7 2012  FALSE  FALSE  TRUE   55         0
## 5 280781 2012-08-01      8 2012  FALSE  FALSE  TRUE   56         1
## 6 313246 2012-09-01      9 2012  FALSE  FALSE  TRUE   57         0
##      Prediction      Diff
## 1    507373.9 224182.9
## 2    503352.9 198881.9
## 3    492268.2 153793.2
## 4    496887.6 213877.6
## 5    438744.6 157963.6
## 6    520981.9 207735.9
```

```
dane2012 %>%
  ggplot(aes(Date, ilość)) +
  geom_line() +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y") +
  labs(title = "Sprzedaż w latach 2012-2016 ")
```

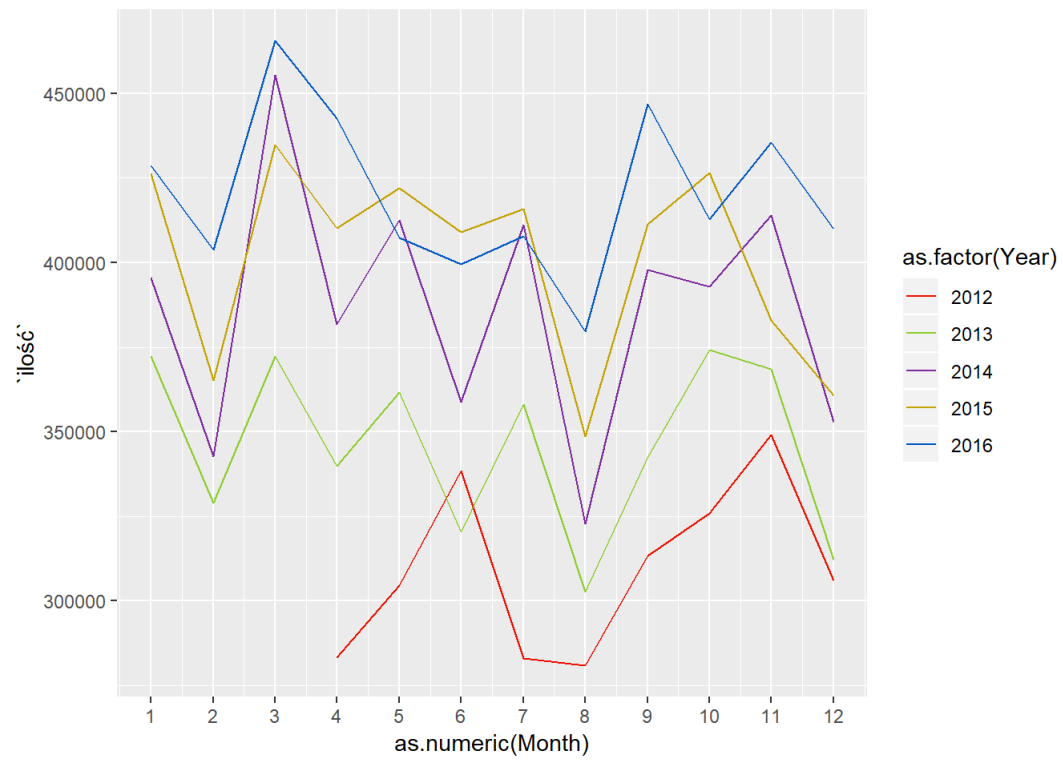
Sprzedaż w latach 2012-2016



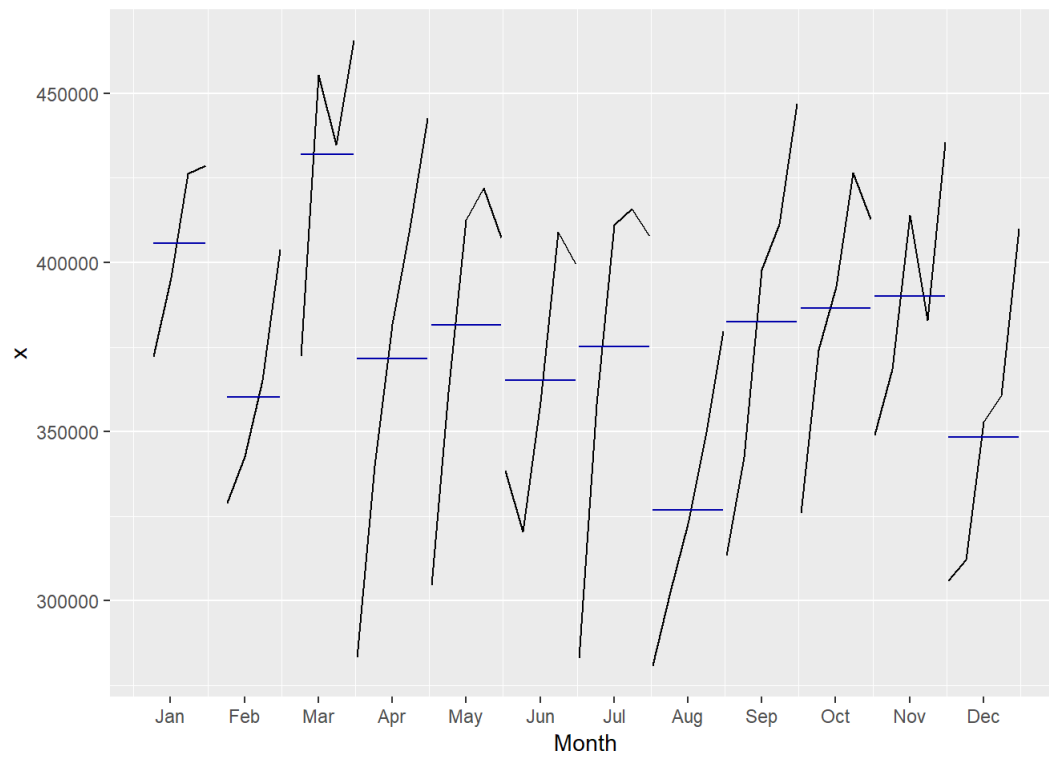
```
dane2012 %>%  
  ggplot(aes(Year, ilość, col = Month)) +  
  geom_line() +  
  scale_color_manual(values = cols)
```



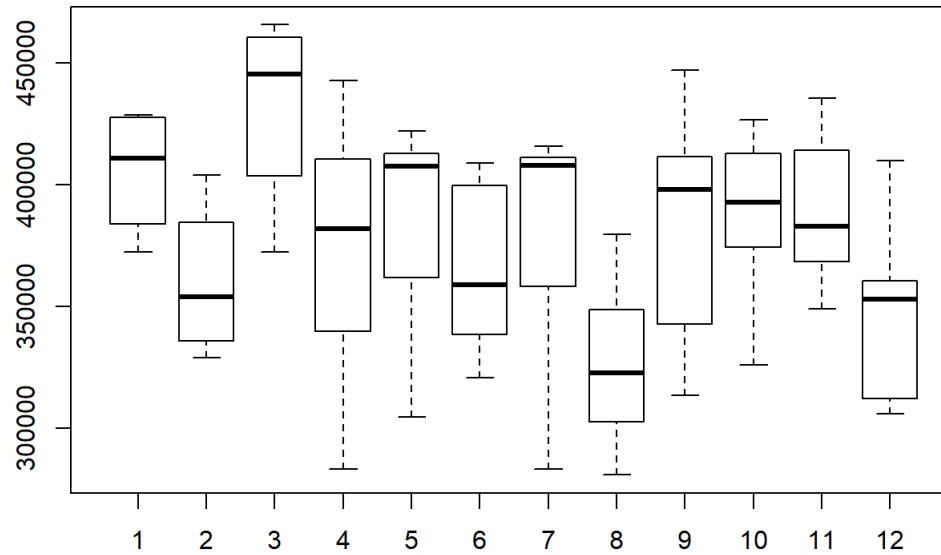
```
dane2012 %>%
  ggplot(aes(as.numeric(Month), ilość, col = as.factor(Year))) +
  geom_line(show.legend = TRUE) +
  scale_color_manual(values = cols) +
  scale_x_continuous(breaks = 1:12)
```



```
volume.ts2 <- ts(dane2012$ilość, start= c(2012,4), frequency= 12)
ggmonthplot(volume.ts2)
```



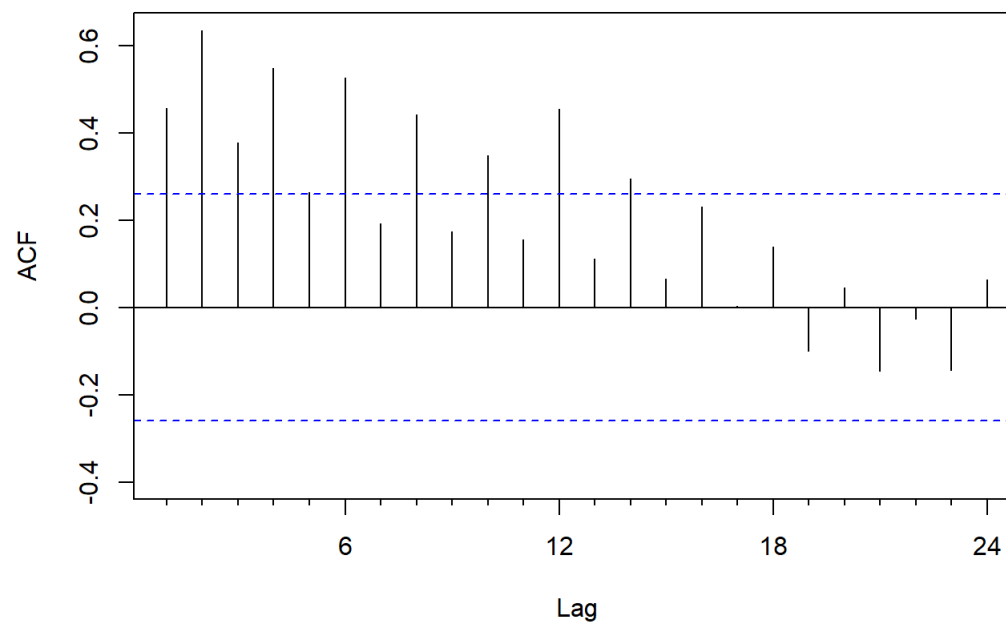
```
boxplot(volume.ts2 ~ cycle(volume.ts2))
```



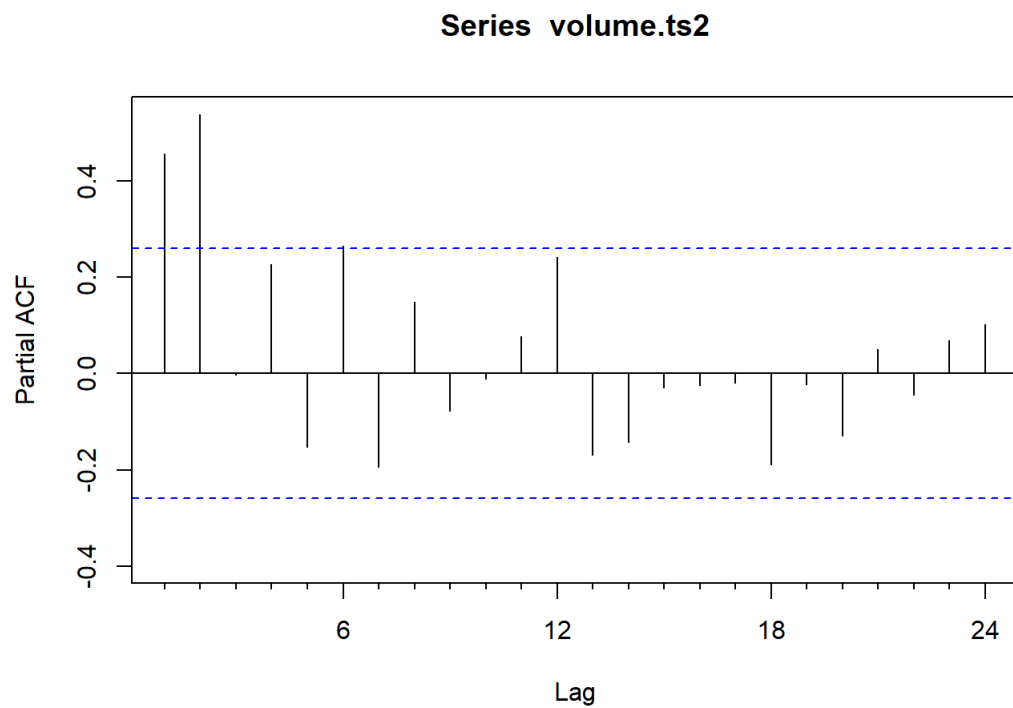
Dzięki funkcji “monthplot” potwierdziłem informację o wzroście sprzedaży w kolejnych miesiącach na przestrzeni kolejnych lat, a także widać sezonowość związaną ze wzrostem sprzedaży w marcu oraz spadkiem w miesiącu wakacyjnym sierpniu i ponownym wzroście w miesiącach jesienno_zimowych

```
Acf(volume.ts2)
```

Series volume.ts2



```
Pacf(volume.ts2)
```



Wykresy “Acf” oraz “Pacf” wyraźnie pokazują występowanie zależności pierwszego i drugiego rzędu oraz korelacji trendowej. Szczególnie widać to w “Acf” gdzie rząd 12 jest skorelowany dodatnio

#####Dekompozycja#####

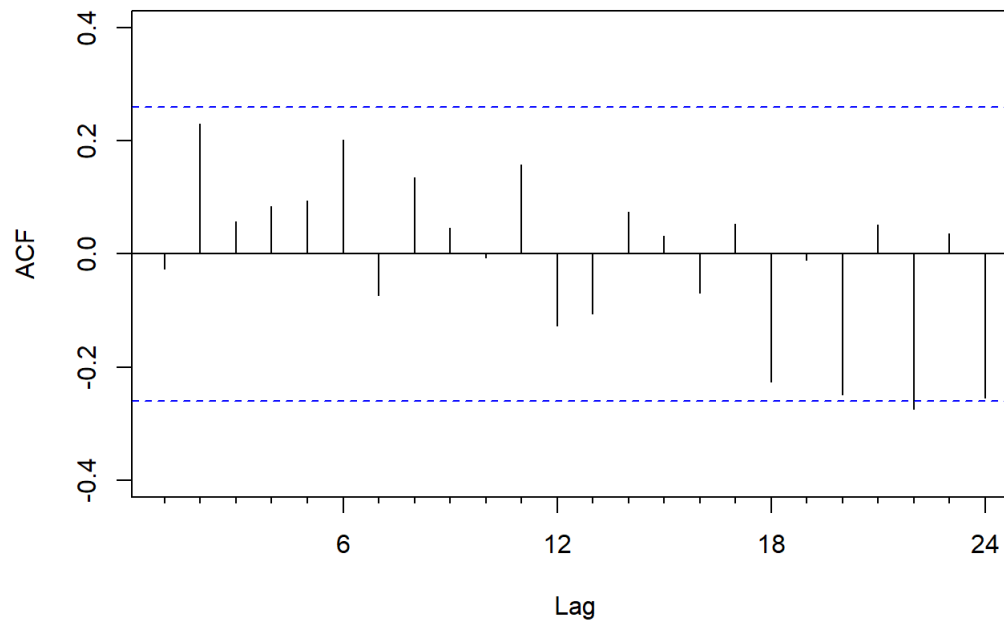
```
#dekompozycja <- decompose(ts2)
model.tslm2 <- tslm(volume.ts2 ~ trend + season)
summary(model.tslm2)
```



```
##
## Call:
## tslm(formula = volume.ts2 ~ trend + season)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -39244 -10327   3011  11695  36604
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 343928.1    10669.7   32.234 < 2e-16 ***
## trend        2203.1      157.9   13.949 < 2e-16 ***
## season2     -47673.6    13733.1   -3.471  0.00117 **
## season3      21984.8    13735.9    1.601  0.11664
## season4     -27507.7    13036.1   -2.110  0.04057 *
## season5     -19602.5    13031.4   -1.504  0.13966
## season6     -38170.0    13028.5   -2.930  0.00536 **
## season7     -30486.7    13027.5   -2.340  0.02387 *
## season8     -80980.4    13028.5   -6.216 1.63e-07 ***
## season9     -27665.5    13031.4   -2.123  0.03942 *
## season10    -25784.7    13036.1   -1.978  0.05422 .
## season11    -24467.0    13042.8   -1.876  0.06731 .
## season12    -68351.5    13051.4   -5.237 4.39e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19420 on 44 degrees of freedom
## Multiple R-squared:  0.8678, Adjusted R-squared:  0.8318
## F-statistic: 24.07 on 12 and 44 DF,  p-value: 1.906e-15
```

```
Acf(residuals(model.tslm2))
```

Series residuals(model.tslm2)



```
Box.test(residuals(model.tslm2), type = "Ljung-Box", lag = 20)
```

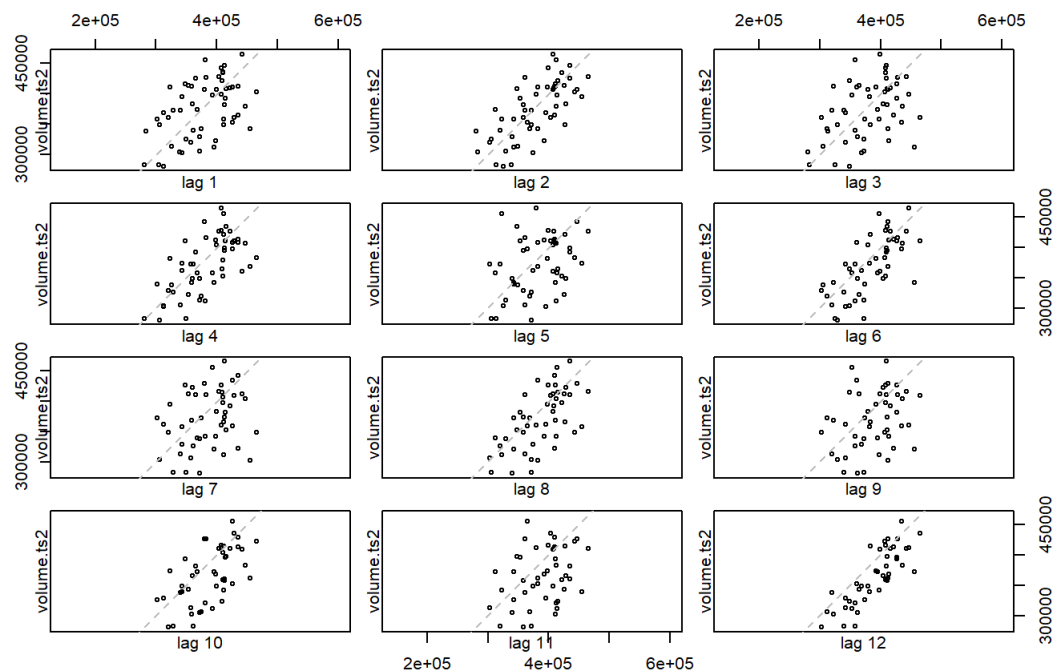
```
##  
## Box-Ljung test  
##  
## data: residuals(model.tslm2)  
## X-squared = 23.959, df = 20, p-value = 0.2442
```

Po dekompozycji z użyciem regresji znikły zależności, a test Ljung-Boxa pozwala przyjąć hipotezę o losowości resz

Przeanalizuję jeszcze same reszty pod kątem korelacji związanej z okresowością

```
lag.plot(volume.ts2, lags = 12, do.lines = FALSE, main = "szereg po marzec.2012-wykres lag.plot" )
```

szereg po marzec.2012-wykres lag.plot

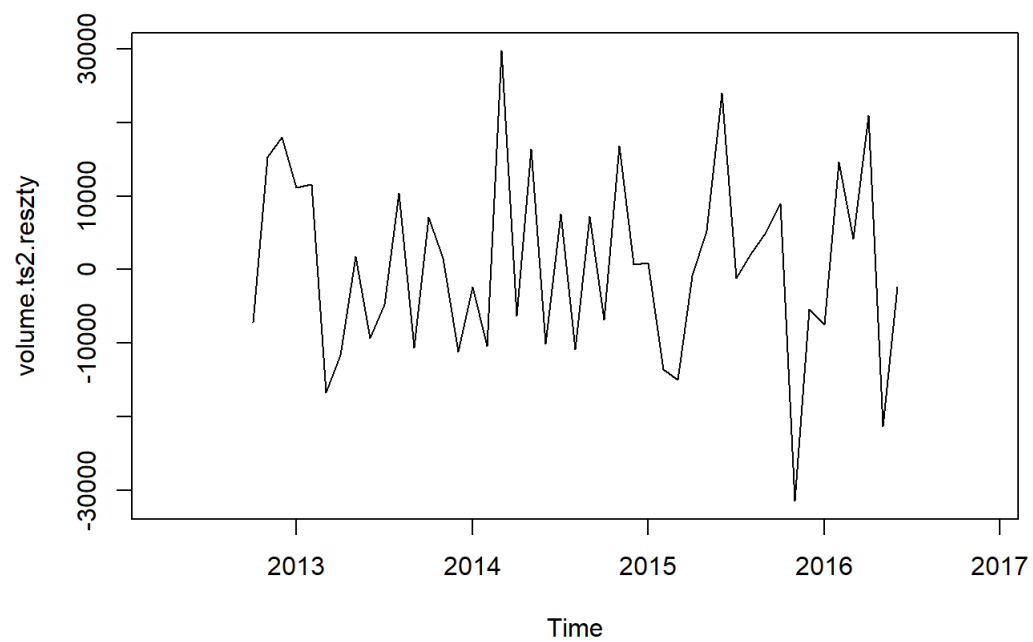


Po wykresach widać, że największa zależność występuje w 12 rzędzie co jest związane z roczną powtarzającą się krotnością. W pewnym zakresie widać zależność również w pierwszym i drugim rzędzie.

Usunę z modelu trend i sezonowość. Ponownie przeanalizuję wykresy reszt

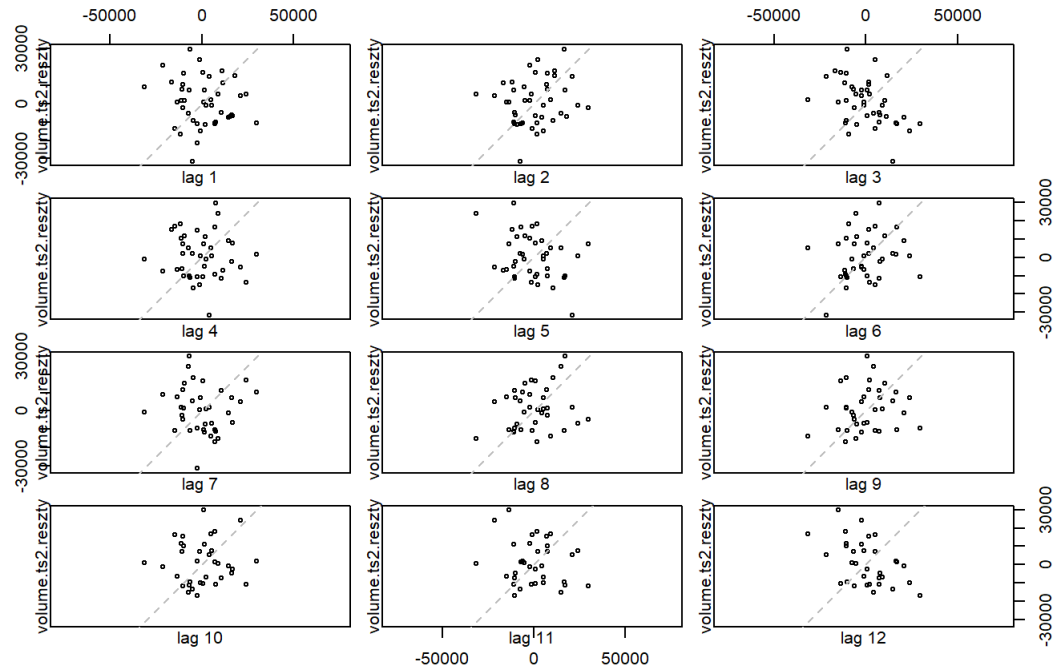
```
volume.ts2.reszty <- decompose(volume.ts2)$random  
plot(volume.ts2.reszty, main= "Szereg volume.ts2 po usunięciu trendu i sezonowości")
```

Szereg volume.ts2 po usunięciu trendu i sezonowości



```
volume.ts2.reszty <- na.omit(volume.ts2.reszty)
lag.plot(volume.ts2.reszty, lags = 12, do.lines = FALSE, main = "szereg po marzec.2012-po usnięciu trendu i sezonowości" )
```

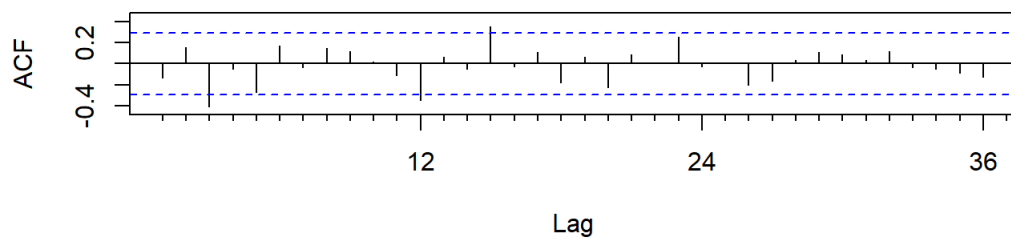
szereg po marzec.2012-po usnięciu trendu i sezonowości



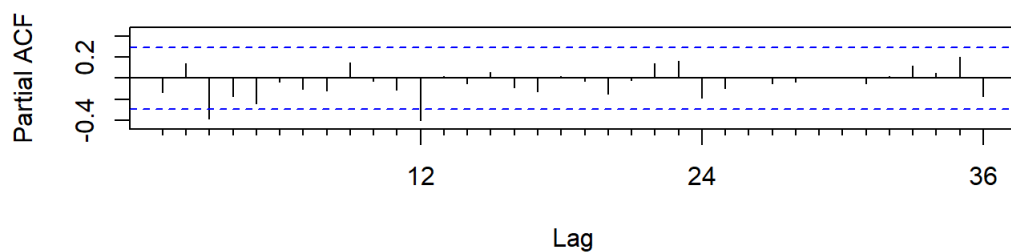
Nie widać już tak silnych zależności czasowych jak w przypadku danych wyjściowych

```
par(mfrow = c(2,1))  
Acf(volume.ts2.reszty, lag.max = 36)  
Pacf(volume.ts2.reszty, lag.max = 36)
```

Series volume.ts2.reszty



Series volume.ts2.reszty



Wykresy autokorelacji potwierdzają, że zależności zostały w dużym zakresie wyeliminowane

#####Wyglądanie przy pomocy średniej ruchomej#####

```
ma4 <- ma(volume.ts2, order = 3)
ma6 <- ma(volume.ts2, order = 6)
head(ma4)
```

```
##          Apr      May      Jun      Jul      Aug      Sep
## 2012      NA 308712.3 308652.0 300755.3 292345.7 306640.3
```

```
head(ma6)
```

```
##          Apr      May      Jun      Jul      Aug      Sep
## 2012      NA      NA      NA 304087.6 311354.2 312350.5
```

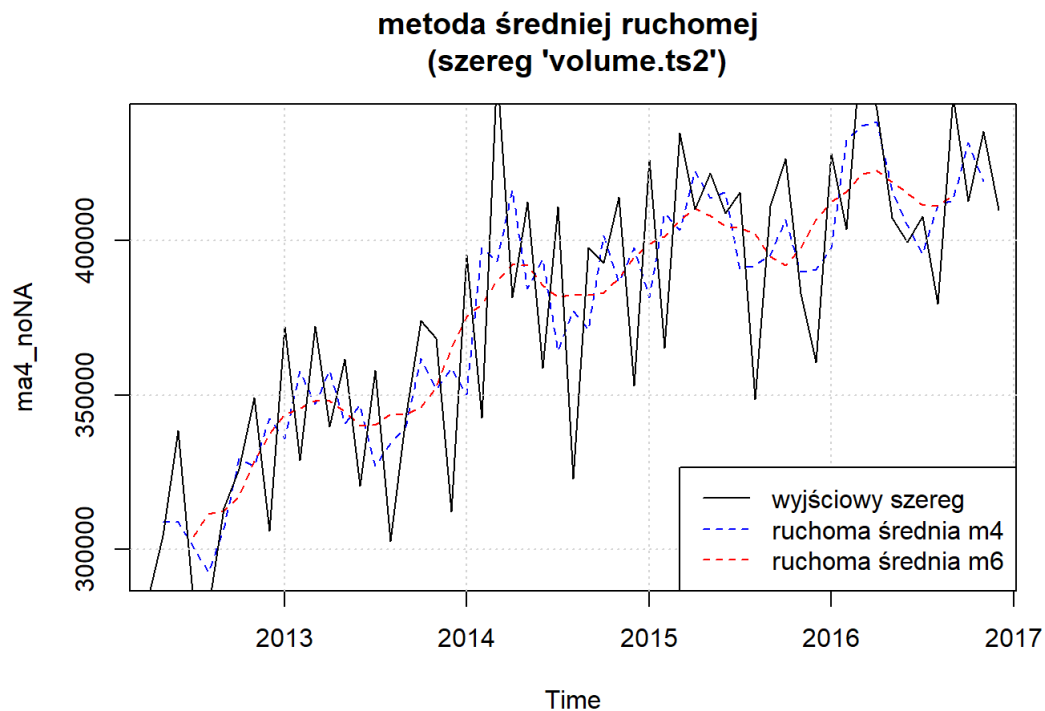
usuwam NA z szeregów

```

ma4_noNA <- na.omit(ma4)
ma6_noNA <- na.omit(ma6)

plot(ma4_noNA, main = paste("metoda średniej ruchomej \n", "(szereg 'volume.ts2')"), col = "blue", lty = 2)
lines(ma6_noNA, col = "red", lty = 2)
lines(volume.ts2, col = "black", lty = 1)
grid()
legend("bottomright",
      legend = c("wyjściowy szereg",
                  "ruchoma średnia m4",
                  "ruchoma średnia m6"),
      col = c("black", "blue", "red"),
      lty = c(1, 2, 2))

```



Najbardziej dopasowane wygładzenie do analizowanych danych jest wygładzenie rzędu $q=3$. Wyższe rzędy wydają się być niedokładne

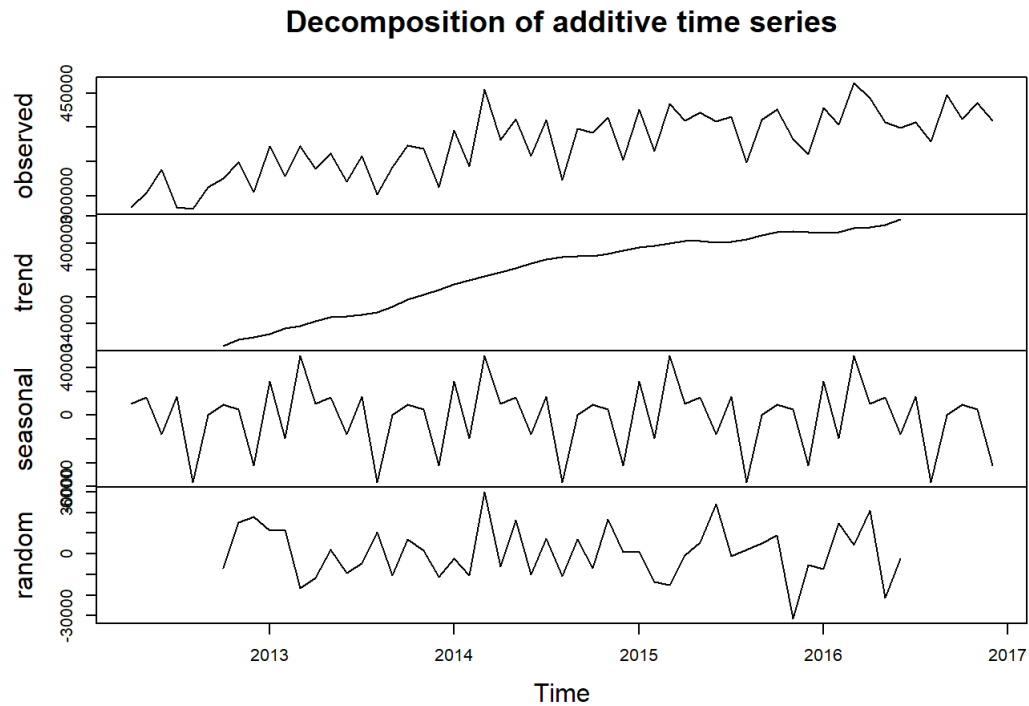
#####Dekompozycja addytywna#####

Wykres wykazuje cechy addytywne, dlatego sprawdzę również tę dekompozycję przy pomocy funkcji "decompose" z pakietu "stats"

```

volume.ts2.dekomp.add <- decompose(volume.ts2, type = "additive")
plot(volume.ts2.dekomp.add)

```



Również został potwierdzony trend długoterminowy oraz sezonowość

```

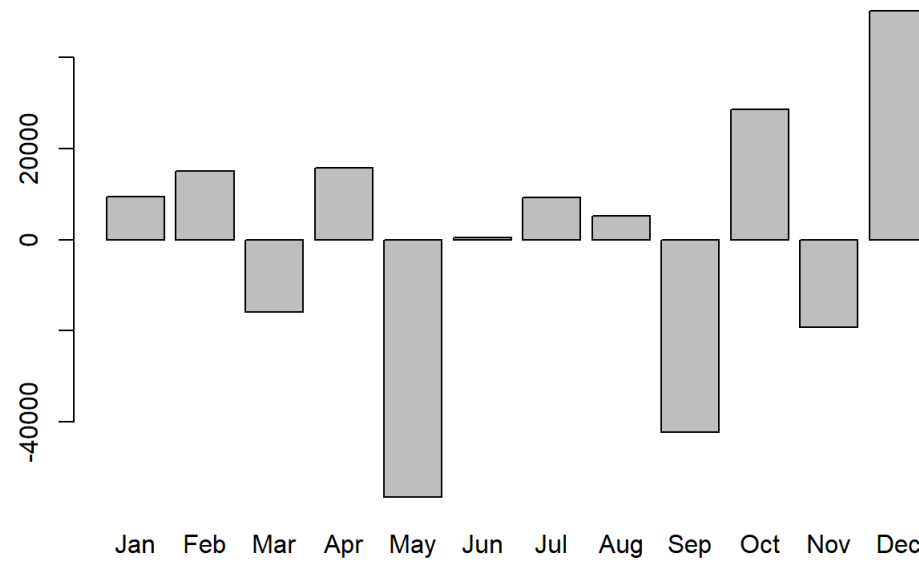
volume.ts2.dekomp.add.trend <- volume.ts2.dekomp.add$trend
volume.ts2.dekomp.add.sezonowość <- volume.ts2.dekomp.add$seasonal
volume.ts2.dekomp.add.ind.sezon <- volume.ts2.dekomp.add$figure
volume.ts2.dekomp.add.reszty <- volume.ts2.dekomp.add$random

#wykres współczynników sezonowych

barplot(volume.ts2.dekomp.add.ind.sezon, names.arg = month.abb, main = "indeksy sezonowe")

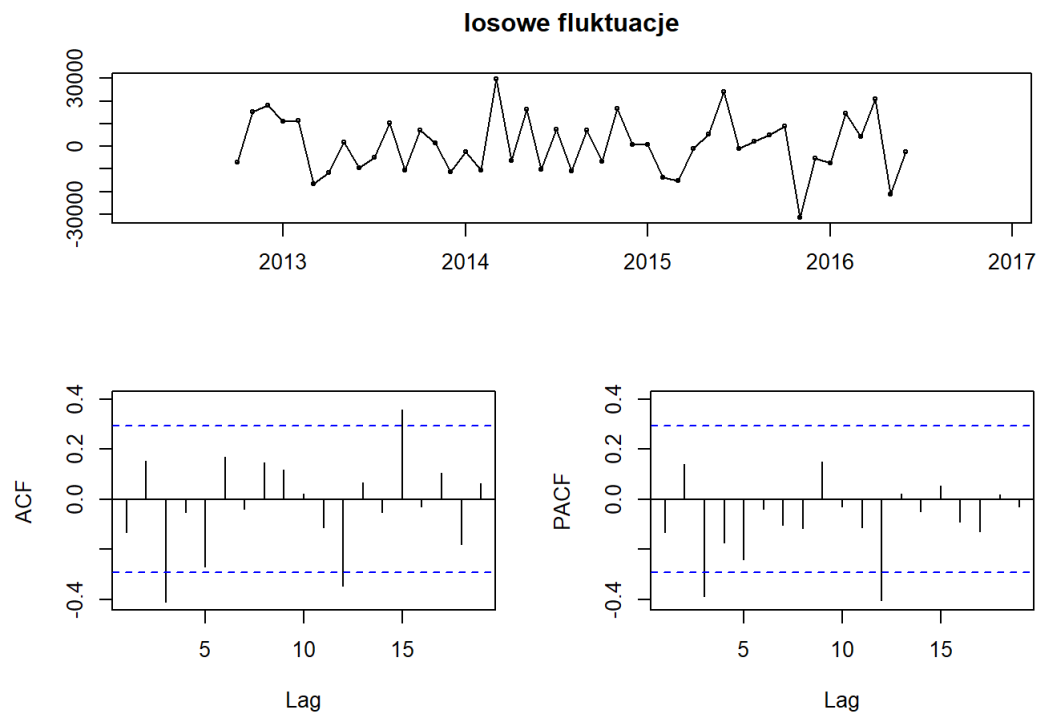
```


indeksy sezonowe



```
#wykresy reszt
```

```
tsdisplay(volume.ts2.dekomp.add.reszty, main = "losowe fluktuacje")
```



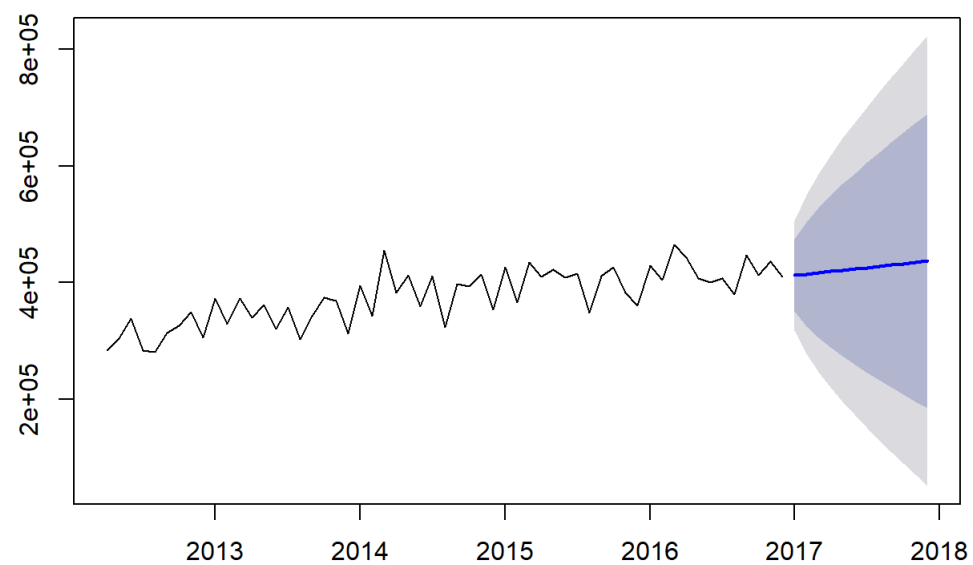
Potwierdzone zostały dzięki “indeksom sezonowym” wzrosty w miesiącach jesienno zimowych oraz wyraźne zanikanie sprzedaż w okresach letnich. Jak wcześniej widzimy występującą korelację rzędu 12 wynikającą z trendu

#####Proгноzy#####

Na koniec przedstawię dwie prognozy z wykorzystaniem metody “błędzenia losowego z dryfem” oraz “wygładzonej_średniej_expotencjalnej”

```
volume.ts2.forecast.rwf <- rwf(x = volume.ts2, drift = TRUE, h=12)
plot(volume.ts2.forecast.rwf, main = "volume.ts2:błędzenie losowe z dryfem")
```

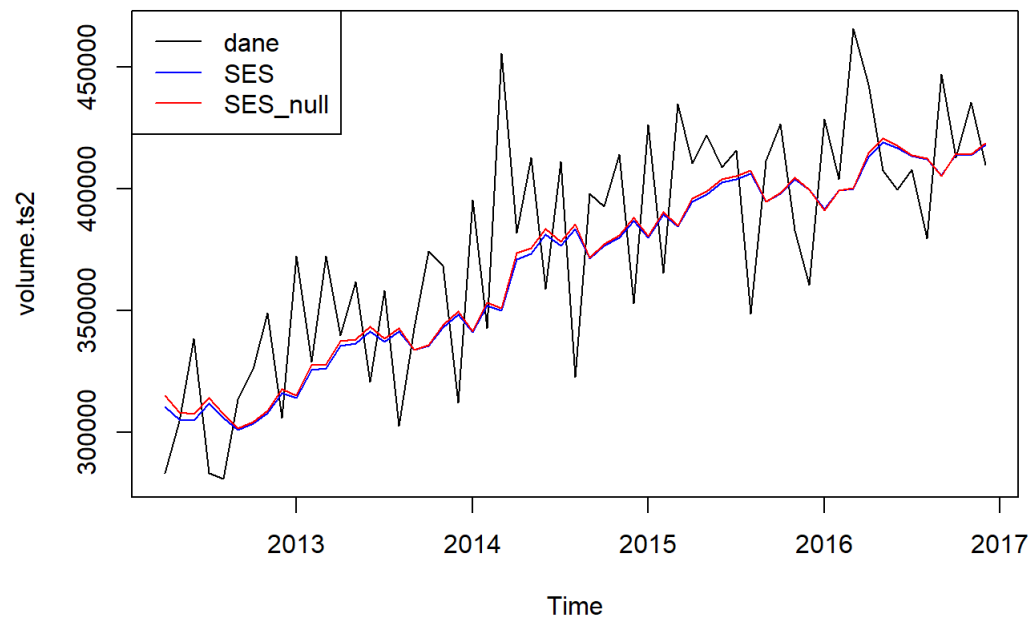
volume.ts2: błędzenie losowe z dryfem



```
volume.ts2_ses <- ses(volume.ts2, h=12, level=c(80,95), fan=FALSE, alpha=0.2)
#summary(volume.ts2_ses)
```

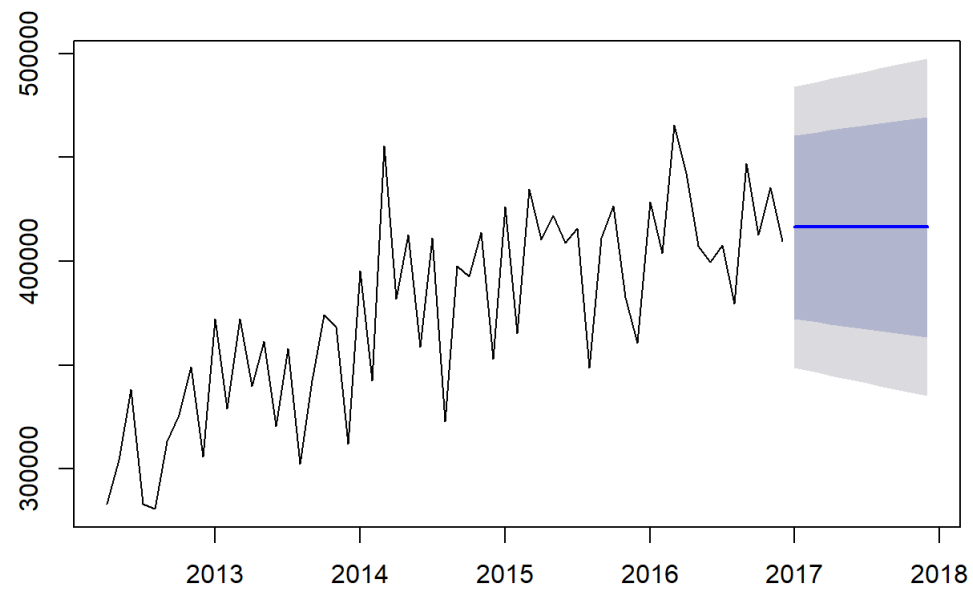
```
volume.ts2_ses_null <- ses(volume.ts2, h=10, level=c(80,95), fan=FALSE, alpha=NULL)
plot(volume.ts2)
lines(volume.ts2_ses$fitted, col="blue")
lines(volume.ts2_ses_null$fitted, col="red")

legend("topleft", legend=c("dane", "SES", "SES_null"), col=c("black", "blue", "red"), lty=c(1,1,1))
```



```
plot(volume.ts2_ses)
```

Forecasts from Simple exponential smoothing



__Niestety dość