Wydział Matematyki i Fizyki Stosowanej

Kierunek studiów**:**

Inżynieria i analiza danych, rok II, semestr IV

Marcin Michno

164182

SZEREGI CZASOWE

Projekt końcowy

Rzeszów, 15 maja 2021 r.

SPIS TREŚCI

Rozdział pierwszy – szereg z wyraźną sezonowością

1. Wprowadzenie do analizowanych danych

Dane z szeregu sezonowego zawierają średnie temperatury dzienne stolic wybranych państw na przestrzeni lat: 1995 - 2020. W projekcie tym analizowane będą wyłącznie te dotyczące Polski. Zostały one pobrane ze strony:

https://www.kaggle.com/sudalairajkumar/daily-temperature-of-major-cities

1) Informacje o ramce danych:

```
'data.frame':   2906327 obs. of  8 variables:
 $ Region        : chr  "Africa" "Africa" "Africa" "Africa" ...
 $ Country       : chr  "Algeria" "Algeria" "Algeria" "Algeria" ...
 $ State         : chr  "" "" "" "" ...
 $ City          : chr  "Algiers" "Algiers" "Algiers" "Algiers" ...
 $ Month         : int  1 1 1 1 1 1 1 1 1 1 ...
 $ Day           : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Year          : int  1995 1995 1995 1995 1995 1995 1995 1995 1995 1995 ...
 $ AvgTemperature: num  64.2 49.4 48.8 46.4 47.9 48.7 48.9 49.1 49 51.9 ...
    Region            Country             State               City               Month
 Length:2906327     Length:2906327     Length:2906327     Length:2906327     Min.   : 1.000
 Class :character   Class :character   Class :character   Class :character   1st Qu.: 3.000
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 6.000
                                                                             Mean   : 6.469
                                                                             3rd Qu.: 9.000
                                                                             Max.   :12.000

      Day             Year       AvgTemperature
 Min.   : 0.00   Min.   : 200   Min.   :-99.0
 1st Qu.: 8.00   1st Qu.:2001   1st Qu.: 45.8
 Median :16.00   Median :2007   Median : 62.5
 Mean   :15.72   Mean   :2007   Mean   : 56.0
 3rd Qu.:23.00   3rd Qu.:2013   3rd Qu.: 75.5
 Max.   :31.00   Max.   :2020   Max.   :110.0
```

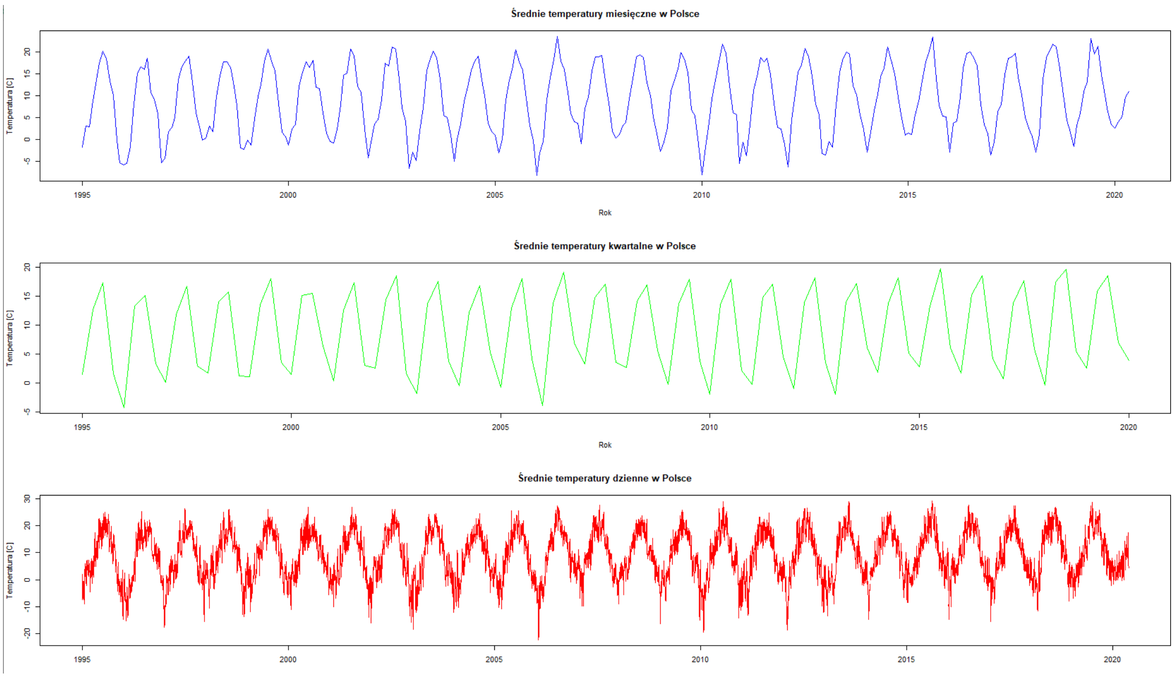
2) Przykładowe rekordy danych:

```
  Region Country State    City Month Day Year AvgTemperature
1 Africa Algeria       Algiers     1   1 1995           64.2
2 Africa Algeria       Algiers     1   2 1995           49.4
3 Africa Algeria       Algiers     1   3 1995           48.8
4 Africa Algeria       Algiers     1   4 1995           46.4
5 Africa Algeria       Algiers     1   5 1995           47.9
6 Africa Algeria       Algiers     1   6 1995           48.7
```

2. Utworzenie szeregów czasowych z średnimi: diennymi, miesięcznymi, kwartałowymi (Po odpowiednim „wycięciu” potrzebnych danych, konwersji skali temperatur)

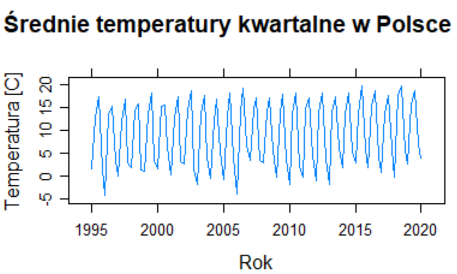
```
> ### Wybór regionu i państwa dla analizy jego danych (np. "Europe", "Poland")
> sort(unique(my_data$Region))
[1] "Africa"                          "Asia"
[3] "Australia/South Pacific"         "Europe"
[5] "Middle East"                     "North America"
[7] "South/Central America & Carribean"

> sort(unique(my_data[ChosenRows,]$Country))
 [1] "Albania"           "Austria"           "Belarus"           "Belgium"
 [5] "Bulgaria"          "Croatia"           "Cyprus"            "Czech Republic"
 [9] "Denmark"           "Finland"           "France"            "Georgia"
[13] "Germany"           "Greece"            "Hungary"           "Iceland"
[17] "Ireland"           "Italy"             "Latvia"            "Macedonia"
[21] "Norway"            "Poland"            "Portugal"          "Romania"
[25] "Russia"            "Serbia-Montenegro" "Slovakia"          "Spain"
[29] "Sweden"            "Switzerland"       "The Netherlands"   "Ukraine"
[33] "United Kingdom"    "Yugoslavia"
> ChosenCountry <- "Poland"
```

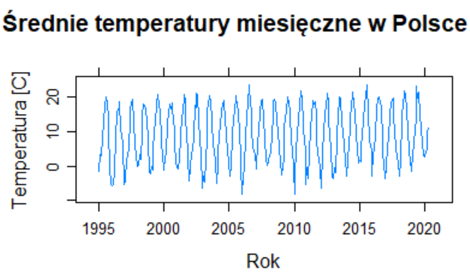
```
### Utworzę średnie dla miesięcy w danych latach dla większej ilości wykresów, ich przejrzystości
#   Można zrobić podobnie dla lat w danym okresie, powstałaby jednak mała, niereprezentatywna próba
MonthTemp <- c()
for (i in unique(NData$Year)){
  for (j in unique(NData$Month)){
    value = mean(NData[((NData[1:dim(NData)[1],2]==i) & (NData[1:dim(NData)[1],3]==j)),5], na.rm=TRUE)
    MonthTemp <- c(MonthTemp, value)
  }
}

### Utworzenie szeregu czasowego dla kwartałów w poszczególnych latach poprzez agregację danych miesięcznych
TimeSerialQ <- aggregate(TimeSerialM, nfrequency = 4, FUN = mean)
```

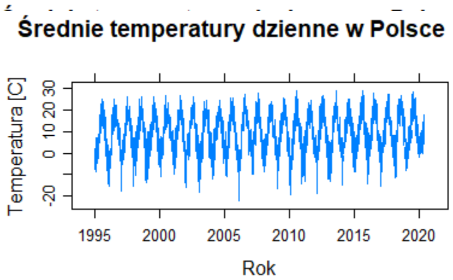


3. Reprezentacja szeregu czasowego na różnego rodzaju wykresach

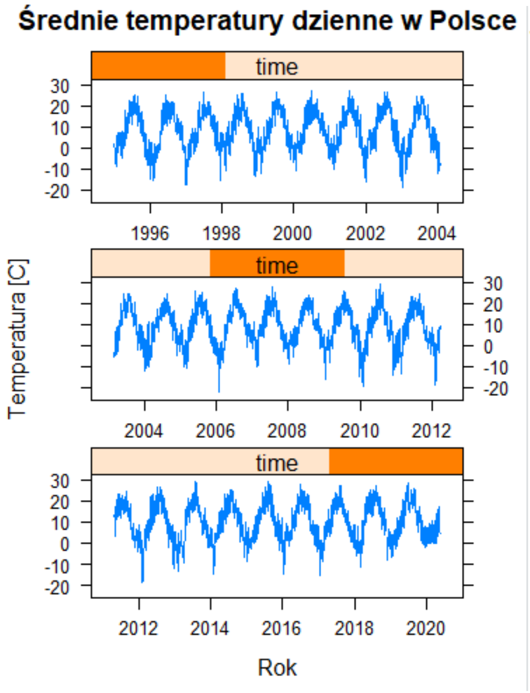
3) Wykresy z użyciem xyplot()

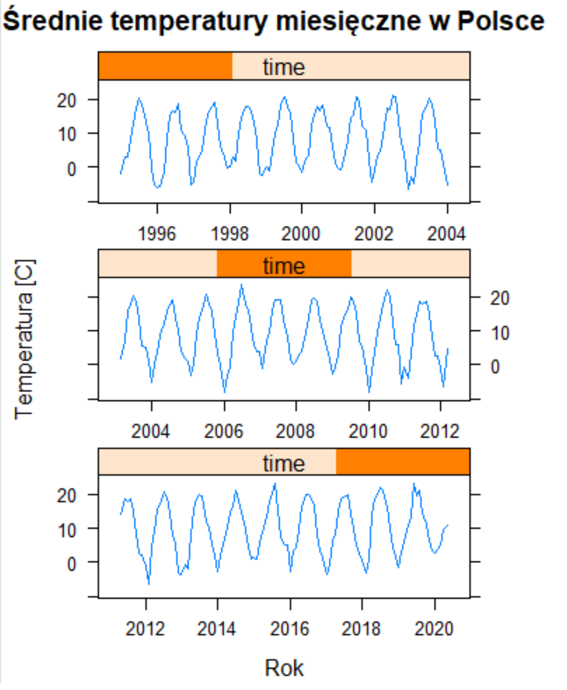


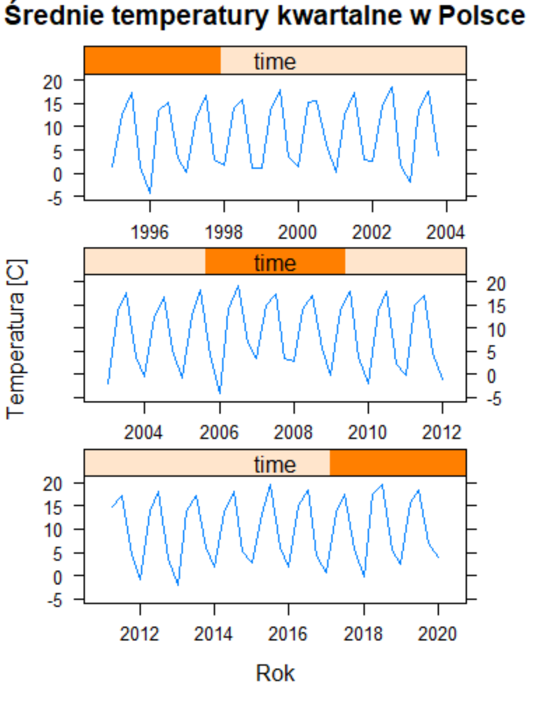




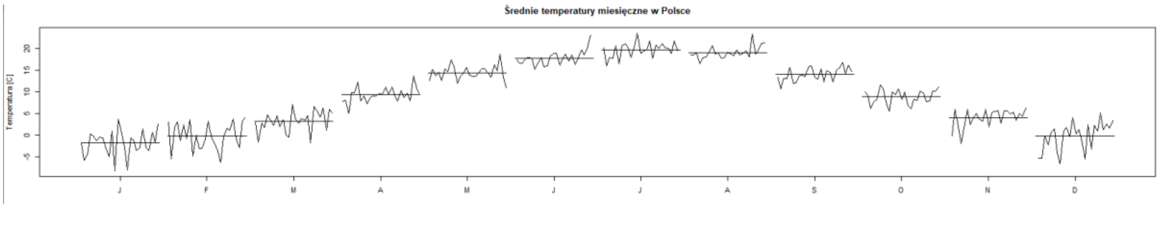
4) Wykresy panelowe

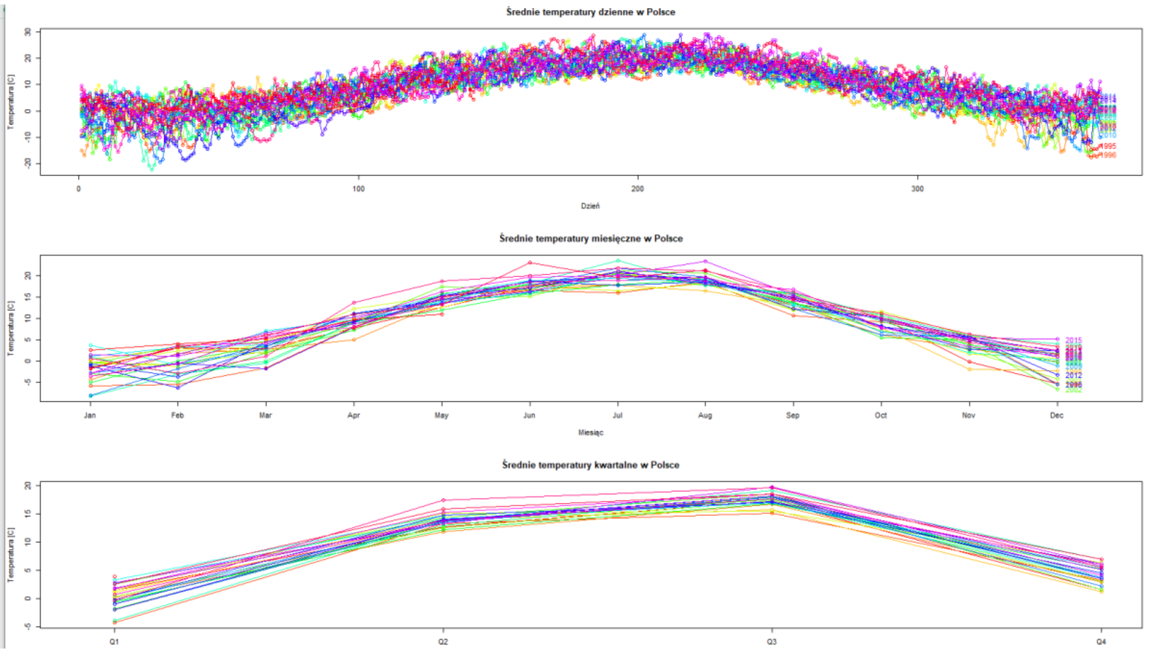






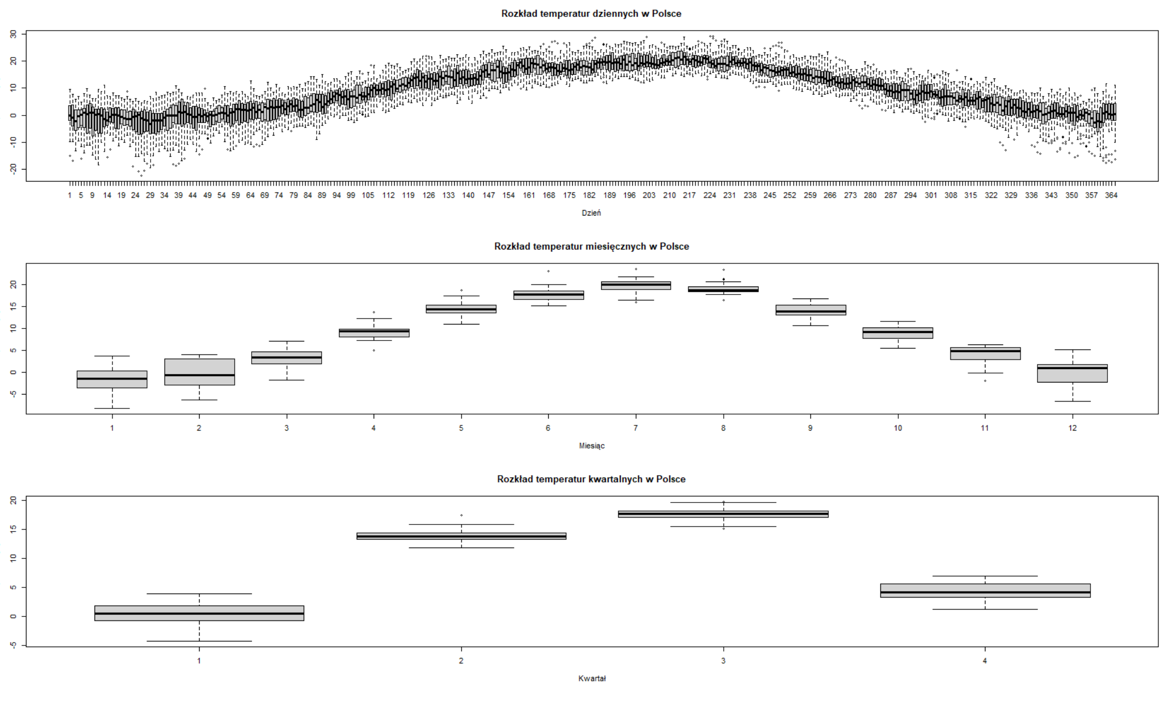
5) Wykresy sezonowe z użyciem monthplot(), seasonplot()



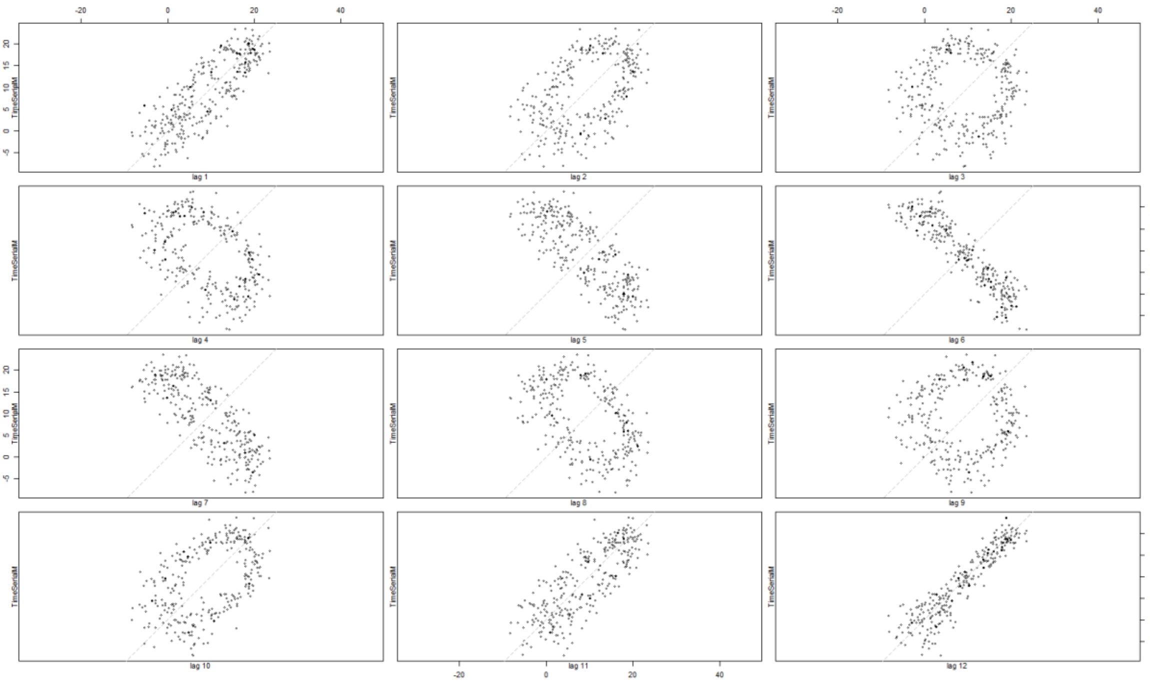


Widoczna jest duża różnorodność temperatur dziennych w poszczególnych latach, jednak zawsze także sezonowość.

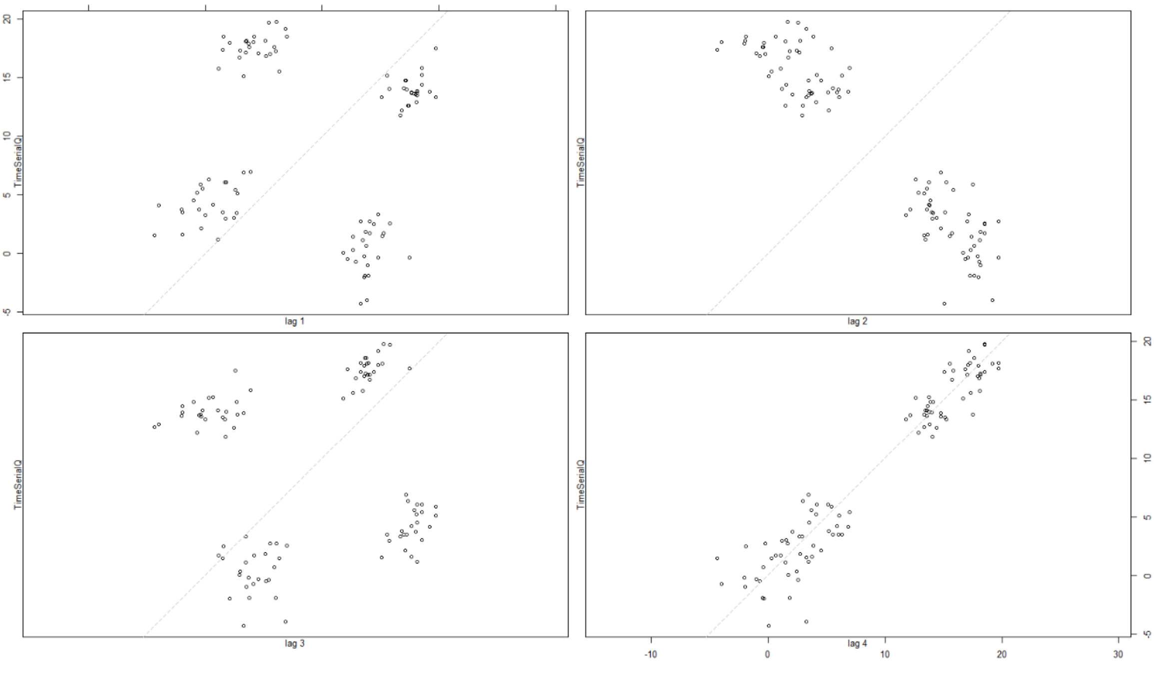
6) Wykresy pudełkowe



7) Wykresy rozrzutu wartości opóźnionych

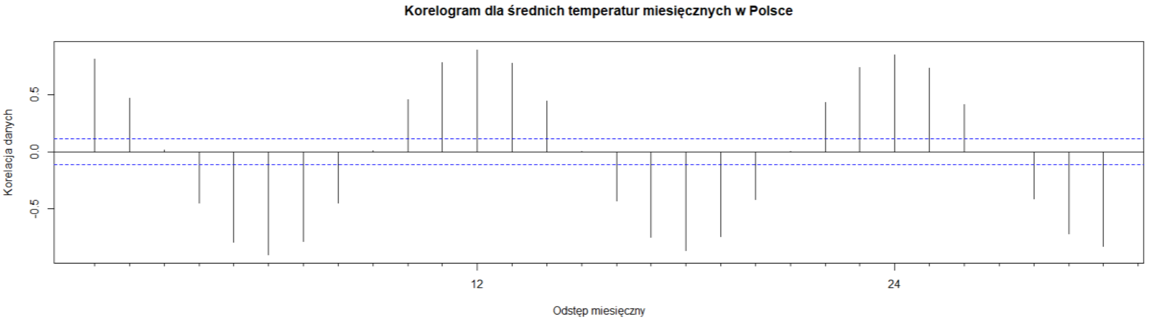


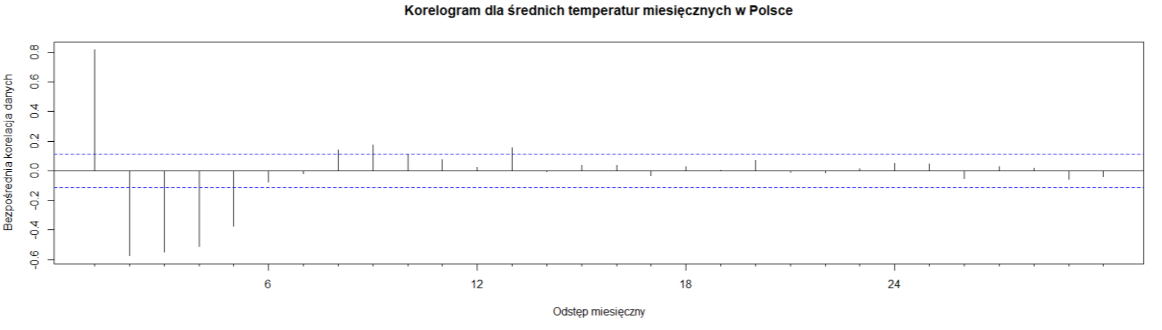
Dla szeregu miesięcznego widzimy największą korelację (zależność) danych dla opóźnienia z krokiem dwunastym, co pokazuje wyraźnie sezonowość roczną. Najsłabsza natomiast dla opóźnienia z krokiem trzecim/dziewiątym, gdzie punkty układają się na kształt koła, co pokazuje działanie poszczególnych pór roku (zimniejszych i cieplejszych). Po przeciwnych stronach widocznej linii na wykresie z opóźnieniem szóstym powinny znaleźć się większości temperatur z przeciwnych pór roku.



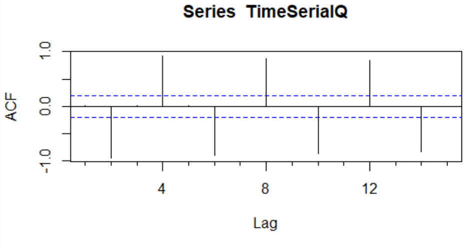
Analogicznie dla szeregu kwartałowego – największa korelacja występuje dla kroku czwartego (sezonowość 4-kwartałowa), a wykresy: pierwszy i trzeci pokazują działanie pór roku, drugi natomiast przeciwne z nich.

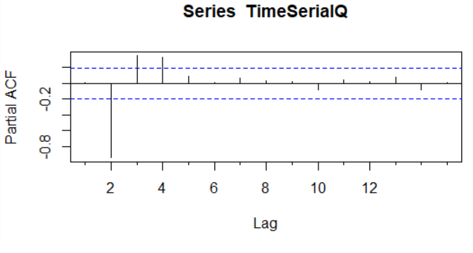
8) Wykresy funkcji autokorelacji – Acf, cząstkowej autokorelacji – Pacf





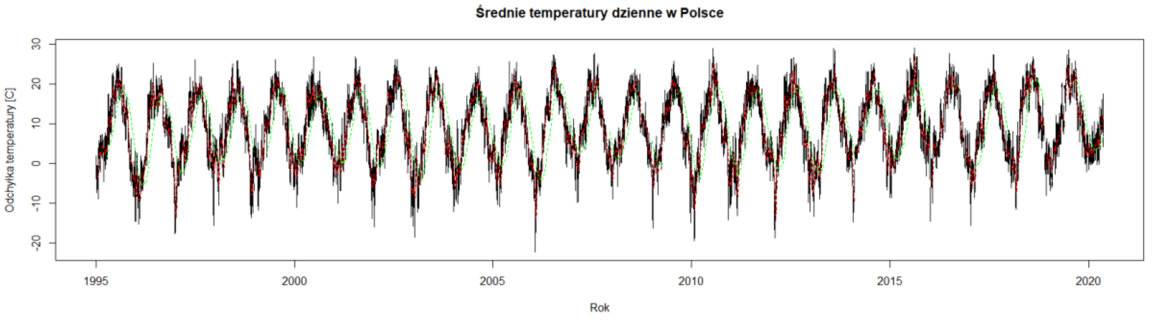
Widoczna jest duża sezonowość poprzez łagodne, regularne zmiany na wykresie funkcji Acf , oraz dzięki dużej „szpilce" przy odstępie sezonowym – 1. Brak widocznego wyraźnie trendu comiesięcznego.

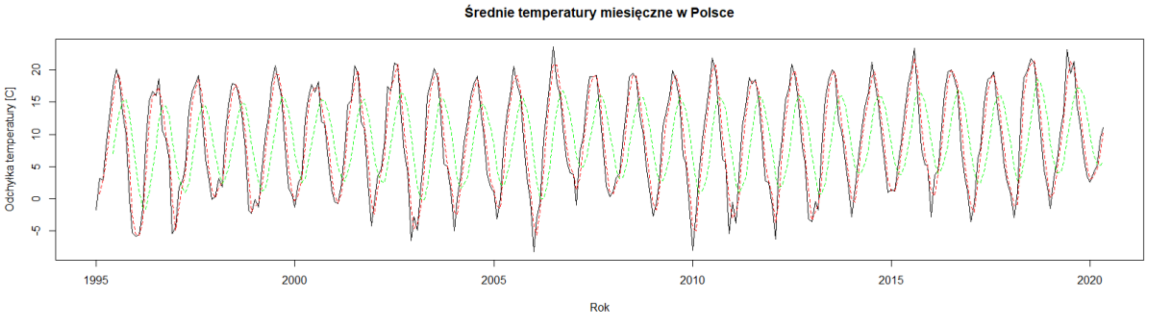




Widoczna sezonowość, jednak dla bardzo krótkich odstępów czasowych (2-kwartałowej)

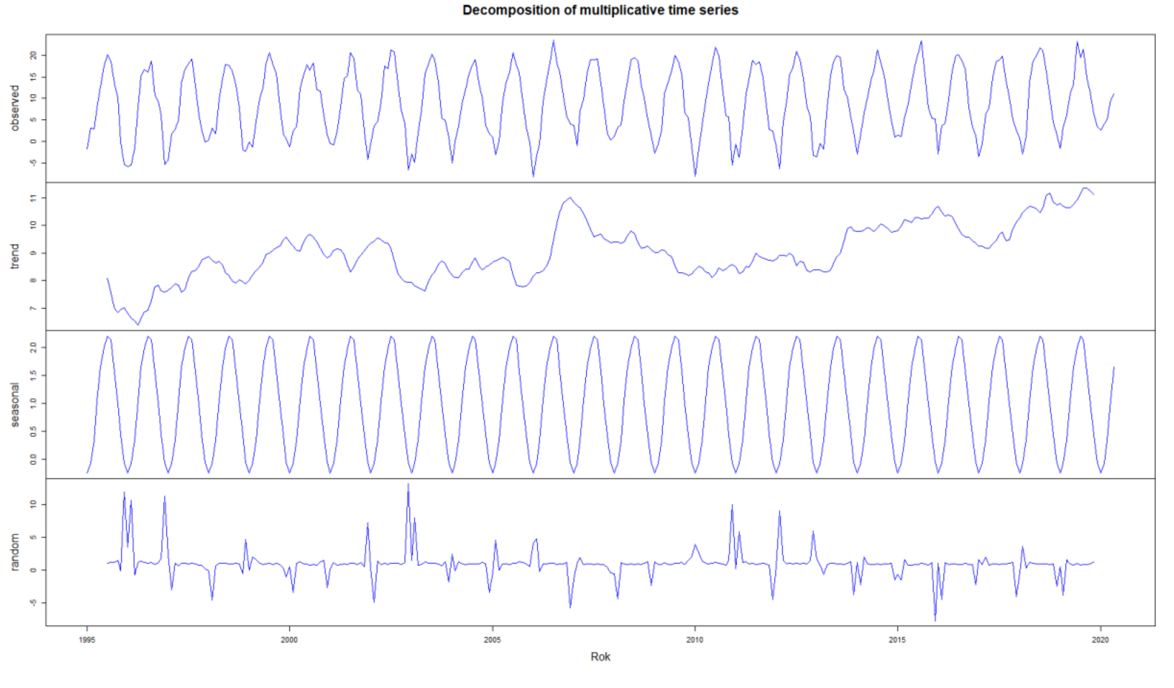
9) Wygładzenie danych poprzez użycie średniej ruchomej

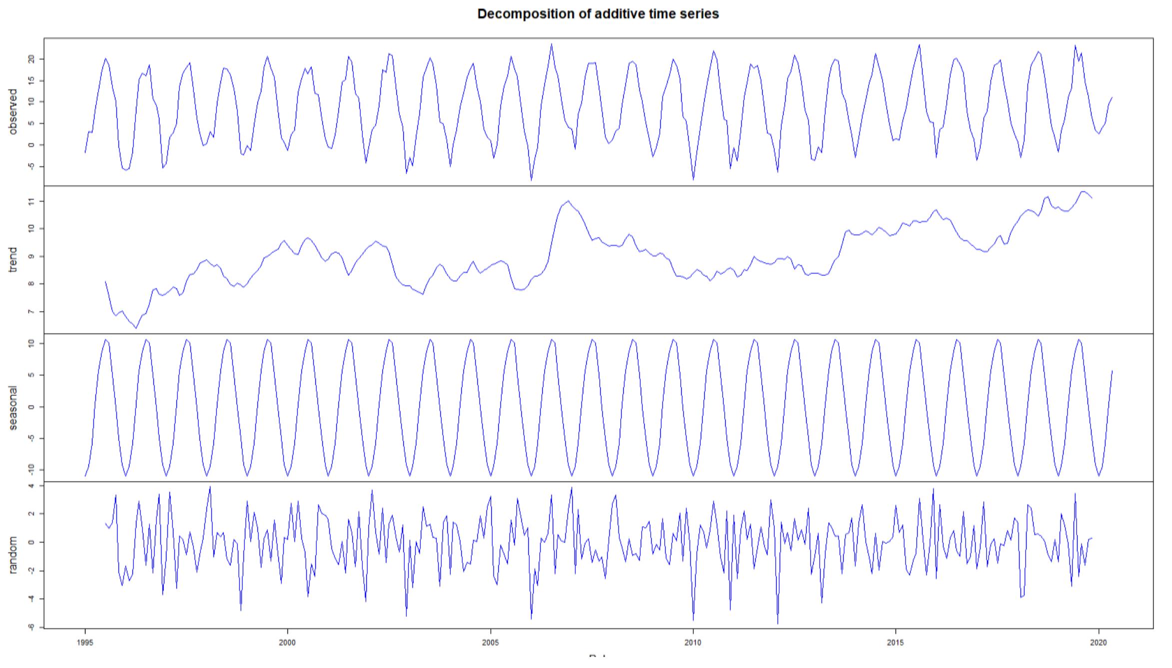




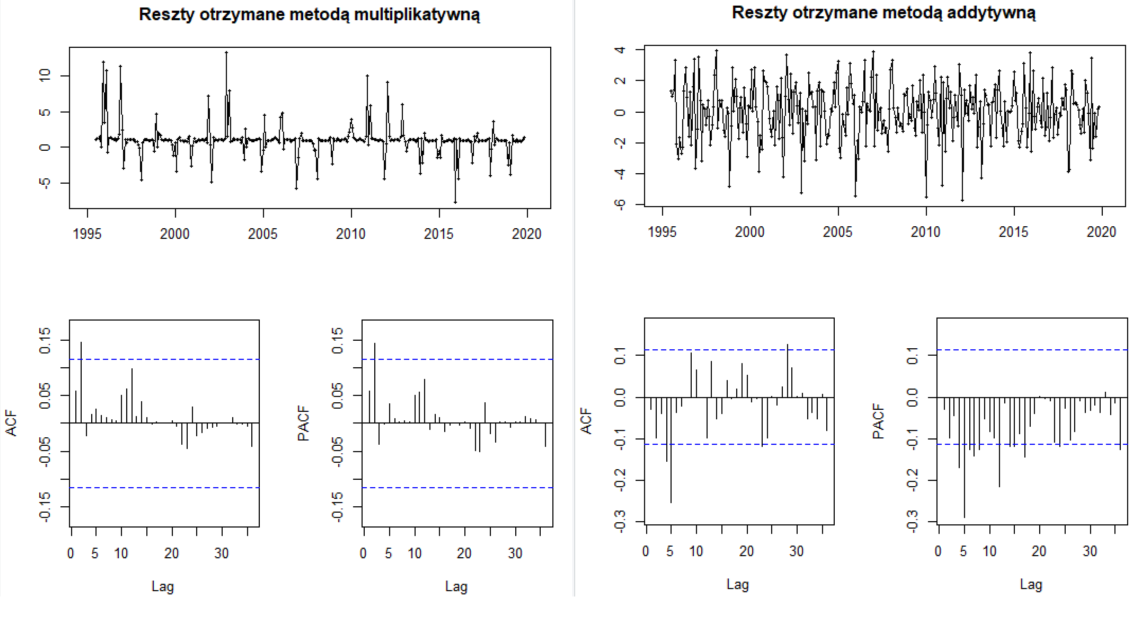
Przykładowe wykresy pokazują wygładzenie szeregu miesięcznego dla odstępu: dwu/sześcio-miesięcznego, oraz szeregu dziennego dla odstępu 14/100 – dniowego oznaczonych kolejno według wielkości spłaszczenia kolorami czerwonym i zielonym.

4. Dekompozycja przykładowych szeregów metodami: addytywną, multiplikatywną



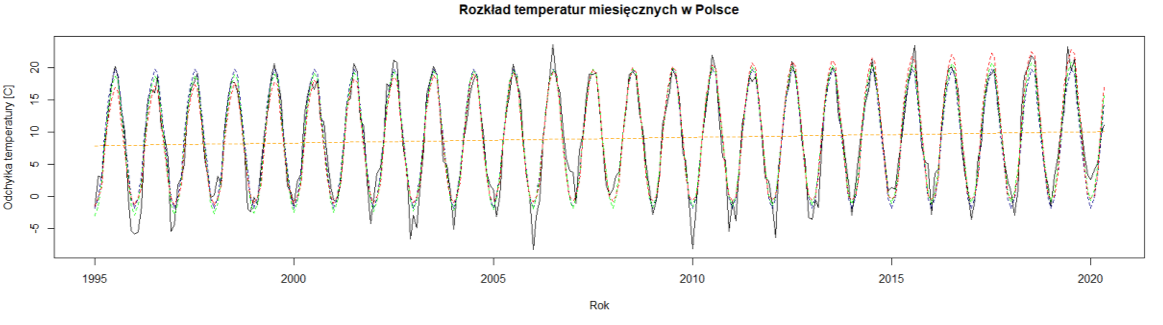


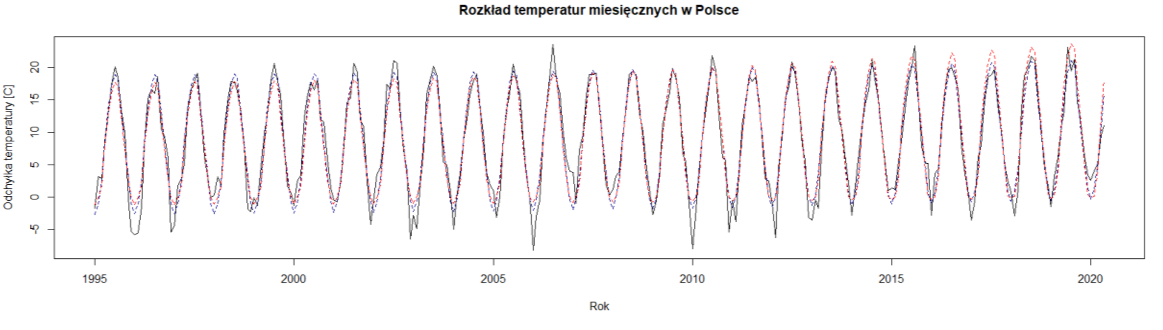
Metoda multiplikatywna w tym przypadku nie poprawiła bardzo rezultatu, ponieważ brak dużego trendu wzrostowego w danym szeregu. Porównajmy ich reszty i wykresy autokorelacji:



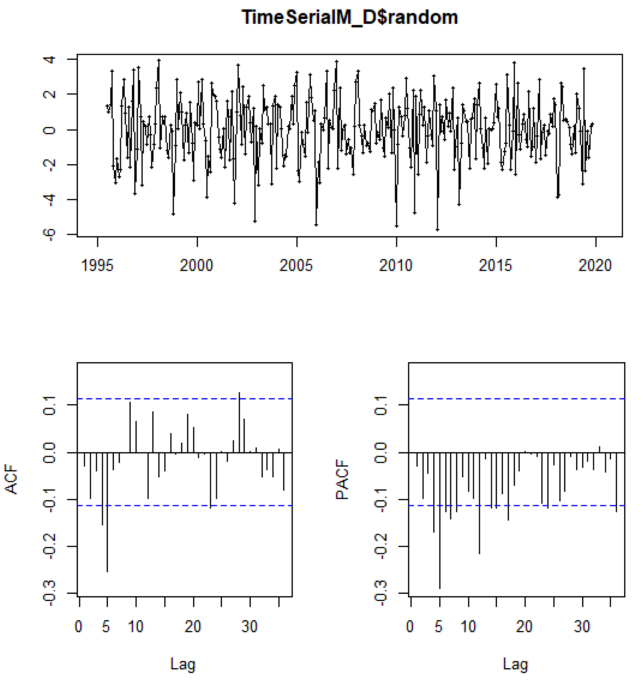
5. Dekompozycje na podstawie modelu regresji: trend liniowy/wielomianowy, sezonowość, transformacja Boxa-Coxa

```
TimeSerialM_T <- tslm(TimeSerialM ~ trend)
TimeSerialM_S <- tslm(TimeSerialM ~ season)
TimeSerialM_TS <- tslm(TimeSerialM ~ trend + season)
TimeSerialM_TSB <- tslm((TimeSerialM) ~ trend + season, lambda = 0.5) # Pierwiastkowa, ew. "auto"
# Trend wielomianowy
TimeSerialM_WS<- tslm(TimeSerialM ~ poly(trend,raw=TRUE,degree=2)+season)
TimeSerialM_WSB<- tslm(TimeSerialM ~ poly(trend,raw=TRUE,degree=2)+season, lambd=0.5)
summary(TimeSerialM_TSB) # Informacje o przykładowym z wybranych modeli

# Porównamy metody
par(mfrow = c(2,1))
plot(TimeSerialM, main="Rozkład temperatur miesięcznych w Polsce",
     xlab = "Rok", ylab="Odchyłka temperatury [C]")
lines(fitted(TimeSerialM_T), col = "orange", lty = 2) # Wizualizacja wykrytego trendu
lines(fitted(TimeSerialM_S), col = "darkblue", lty = 2)  # Wizualizacja wykrytej sezonowości
lines(fitted(TimeSerialM_TS), col = "green", lty = 2) # Wizualizacja wykrytych: trendu, stacjonarności
lines(fitted(TimeSerialM_TSB), col = "red", lty = 2) # Wizualizacja wykrytych: jw., wariancji
plot(TimeSerialM, main="Rozkład temperatur miesięcznych w Polsce",
     xlab = "Rok", ylab="Odchyłka temperatury [C]")
lines(fitted(TimeSerialM_WS), col = "darkblue", lty = 2) # Wizualizacja wykrytych: jw, trend kwadratowy
lines(fitted(TimeSerialM_WSB), col = "red", lty = 2) # Wizualizacja wykrytych: jw, trend kwadratowy
```

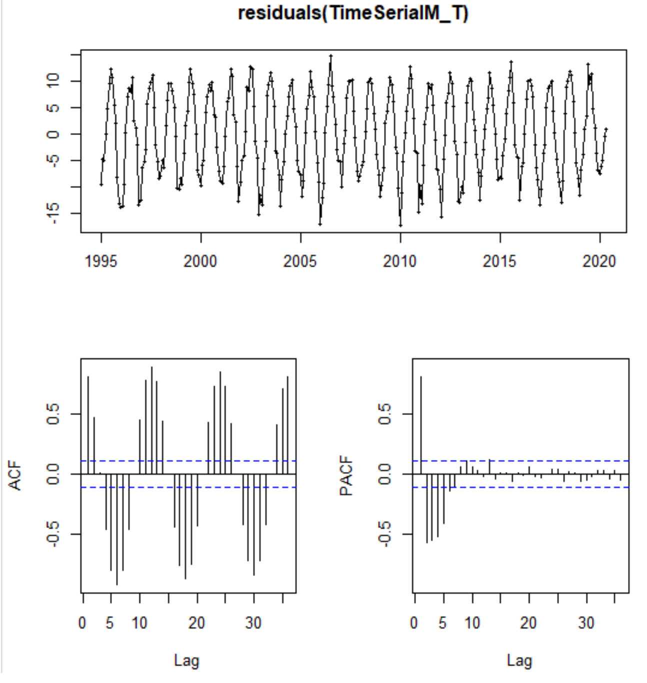




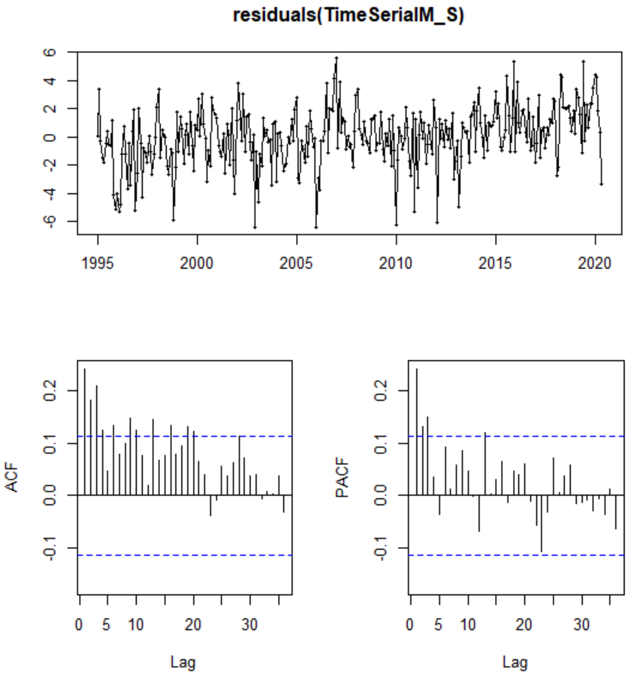
Porównamy reszty z różnych metod dekompozycji na podstawie modelu regresji. Po zastosowaniu dekompozycji metodą addytywną reszty wyglądają, jak na następnej stronie:



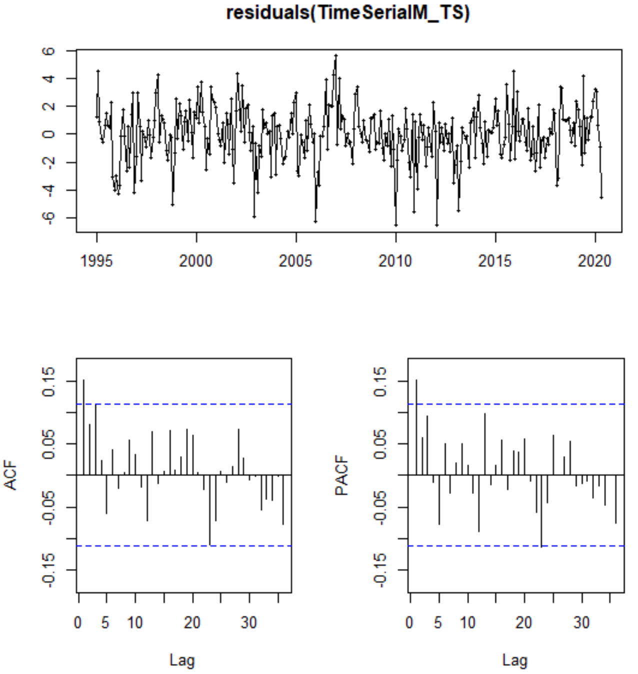
Po wyeliminowaniu trendu rezultat następujący:



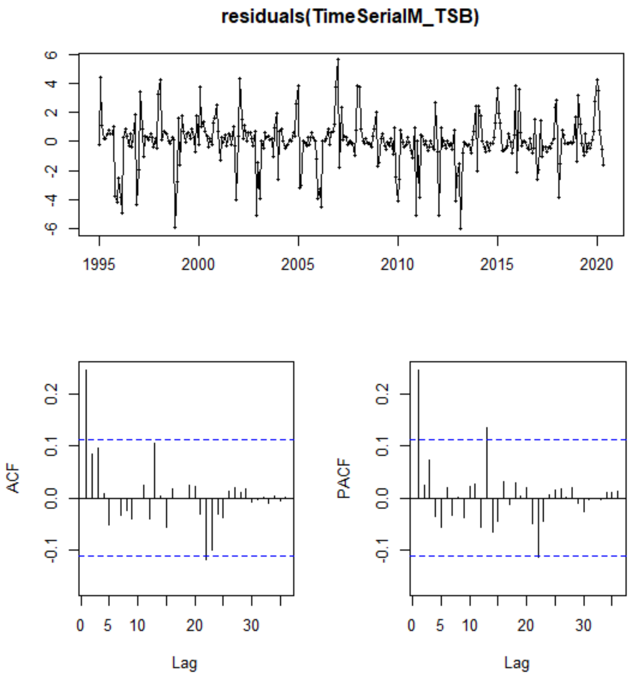
Następnie wyeliminowano sezonowość:



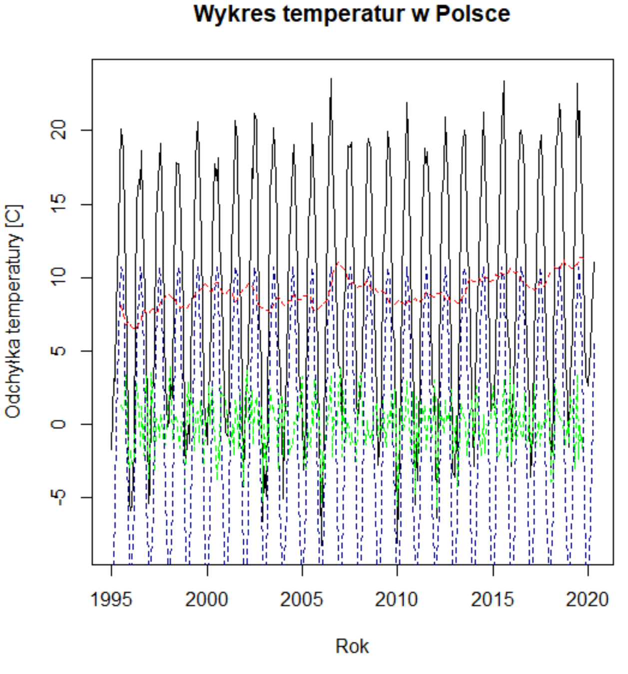
Przy jednoczesnym wyeliminowaniu trendu, sezonowości reszty wyglądają tak:



Przy użyciu dodatkowo transformacji Boxa-Coxa można ustabilizować wariancję, jak na następnej stronie:

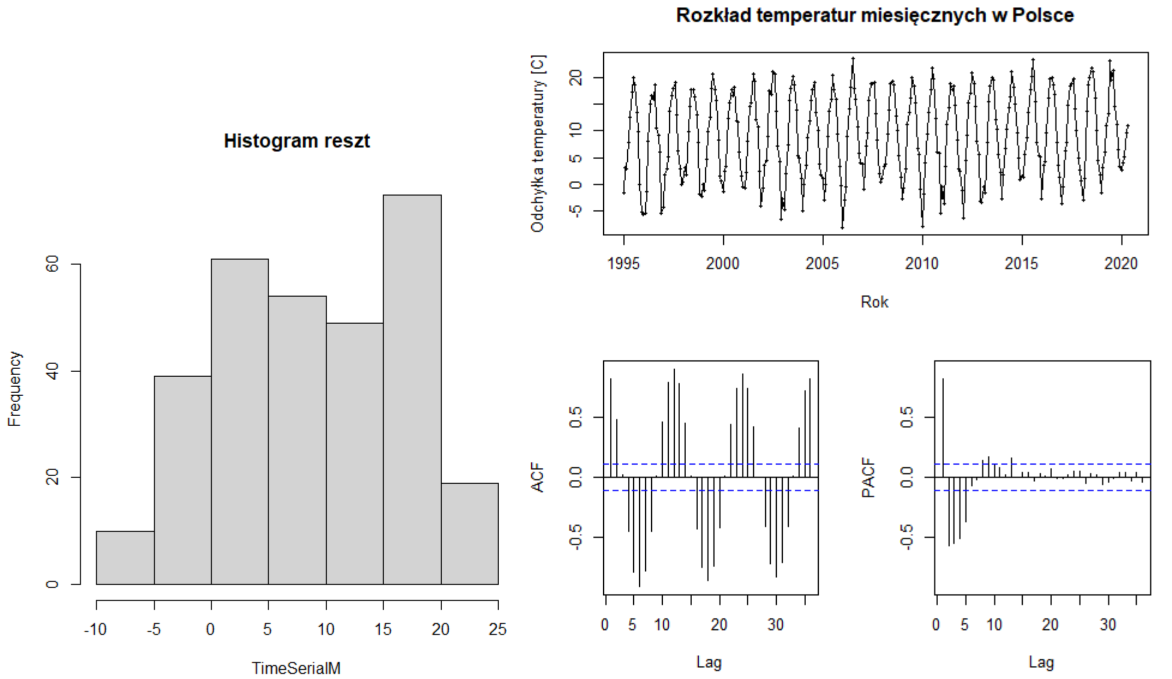


6. Eliminacja trendu i sezonowości z użyciem decompose(), wizualizacja rezultatu



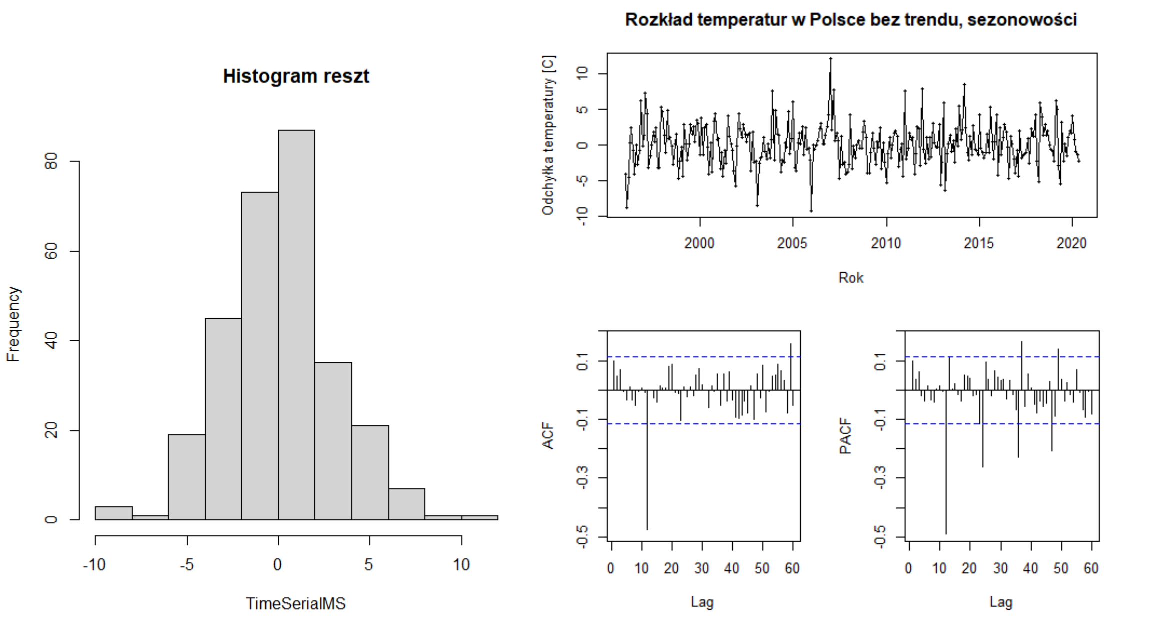
Kolorem ciemnoniebieskim oznaczono wykrytą sezonowość, czerwonym trend, a zielonym pozostałe reszty.

7. Uczynienie szeregu stacjonarnym – eliminacja trendu, sezonowości poprzez różnicowanie i stabilizacja wariancji poprzez transformację Boxa-Coxa




```
> ndiffs(TimeSerialM) # Ilość potrzebnych różnicowań z opóźnieniem wartości jeden
[1] 0
> nsdiffs(TimeSerialM) # Ilość potrzebnych różnicowań z opóźnieniem sezonowym (wartości: dwanaście)
[1] 1
```

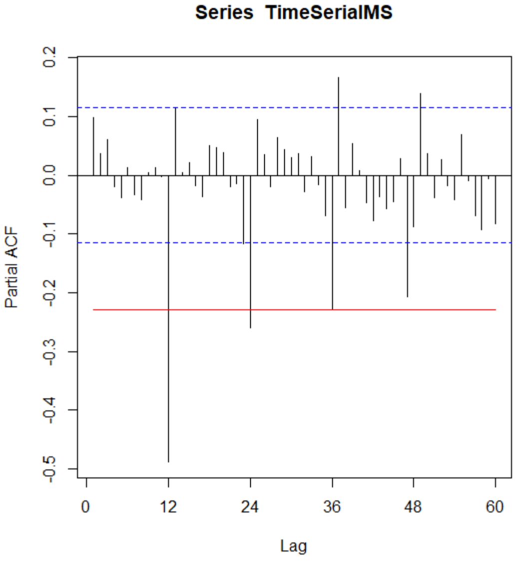
Otrzymany histogram i szereg po zróżnicowaniu z krokiem dwunastym:




```
> ndiffs(TimeSerialMS) # Ilość potrzebnych różnicowań z opóźnieniem wartości jeden
[1] 0
> nsdiffs(TimeSerialMS) # Ilość potrzebnych różnicowań z opóźnieniem sezonowym wartości: dwanaście
```

Po przyglądnięciu się wykresom funkcji autokorelacji bezpośredniej i cząstkowej możemy założyć pewne modele z konkretnymi współczynnikami w celu identyfikacji szeregu. Wartości wykraczające poza przedział ufności na wykresie Acf to: 12, 59 ; dla wykresu Pacf to: 12, 24, 36, 47, 49. Poprawnym byłoby założenie więc modeli np.: MA(12), MA(59), AR(12), AR(49).

8. Sprawdzenie stacjonarności otrzymanego szeregu



Widoczne są wyraźne duże odchyłki od przedziału ufności, nie możemy więc przyjąć danego szereg za stacjonarny. Wartość maksymalnej odchyłki wyznaczyliśmy ze wzoru:

$$x = \frac{1.96}{\sqrt{\text{dł.szeregu}}}$$

9. Wyznaczenie współczynników modelu autoregresji i porównanie dopasowania różnymi metodami estymacji

```
Call:
ar(x = TimeSerialMS2, aic = FALSE, order.max = 12, method = c("yule-walker"))

Coefficients:
      1        2        3        4        5        6        7        8        9       10       11       12  
 0.0909   0.0415   0.0628  -0.0362  -0.0488   0.0261  -0.0462  -0.0504   0.0334   0.0300   0.0420  -0.4881  

Order selected 12  sigma^2 estimated as  6.9
Call:
ar(x = TimeSerialMS2, aic = FALSE, order.max = 12, method = c("burg"))

Coefficients:
      1        2        3        4        5        6        7        8        9       10       11       12  
 0.0903   0.0448   0.0663  -0.0363  -0.0518   0.0236  -0.0474  -0.0509   0.0370   0.0337   0.0436  -0.5140  

Order selected 12  sigma^2 estimated as  6.364
Call:
ar(x = TimeSerialMS2, aic = FALSE, order.max = 12, method = c("ols"))

Coefficients:
      1        2        3        4        5        6        7        8        9       10       11       12  
 0.0657   0.0555   0.0751  -0.0288  -0.0699   0.0036  -0.0604  -0.0428   0.0596   0.0402   0.0423  -0.5009  

Intercept: 0.0474 (0.1482) 

Order selected 12  sigma^2 estimated as  6.147
```

```
Call:
ar(x = TimeSerialMS2, aic = FALSE, order.max = 12, method = c("mle"))

Coefficients:
      1        2        3        4        5        6        7        8        9       10       11       12
 0.0927   0.0511   0.0767  -0.0369  -0.0618   0.0137  -0.0559  -0.0423   0.0457   0.0377   0.0300  -0.5176

Order selected 12  sigma^2 estimated as  6.356
Call:
ar(x = TimeSerialMS2, aic = FALSE, order.max = 12, method = c("yw"))

Coefficients:
      1        2        3        4        5        6        7        8        9       10       11       12
 0.0909   0.0415   0.0628  -0.0362  -0.0488   0.0261  -0.0462  -0.0504   0.0334   0.0300   0.0420  -0.4881

Order selected 12  sigma^2 estimated as  6.9
```

Automatyczny dobór rzędu dzięki kryterium „aic”:

```
Call:
ar(x = TimeSerialMS2, aic = TRUE, order.max = 100, method = c("yule-walker"))

Coefficients:
      1        2        3        4        5        6        7        8        9       10       11       12       13       14
 0.1559   0.0571   0.0539  -0.0206  -0.0349   0.0378  -0.0204  -0.0158  -0.0005   0.0682  -0.0232  -0.7530   0.2404   0.0608
     15       16       17       18       19       20       21       22       23       24       25       26       27       28
 0.0105   0.0244  -0.0157   0.0398   0.0267   0.0122   0.0004   0.0698  -0.1534  -0.5074   0.2659   0.0255  -0.0007   0.0621
     29       30       31       32       33       34       35       36       37       38       39       40       41       42
 0.0343  -0.0121   0.0034  -0.0343   0.0182   0.0584  -0.1703  -0.3350   0.2711  -0.0479   0.0433   0.0100  -0.0186  -0.0625
     43       44       45       46       47       48       49
-0.0392  -0.0309  -0.0326   0.0550  -0.1976  -0.1079   0.1384

Order selected 49  sigma^2 estimated as  5.86
```

10. Wyznaczenie współczynników modelu ruchomej średniej z użyciem funkcji Arima()

```
ARIMA(0,0,12) with non-zero mean

Coefficients:
         ma1     ma2     ma3     ma4     ma5     ma6     ma7     ma8     ma9    ma10     ma11     ma12     mean
      0.0722  0.0040  0.0642  0.0716  0.0005  0.0371  0.0552  0.0267  0.0520  0.0068  -0.0189  -0.9211  -0.0002
s.e.  0.0505  0.0495  0.0528  0.0495  0.0536  0.0522  0.0546  0.0521  0.0537  0.0524   0.0538   0.0503   0.0602

sigma^2 estimated as 4.563:  log likelihood=-645.97
AIC=1319.94   AICc=1321.45   BIC=1371.46

Training set error measures:
                      ME     RMSE      MAE       MPE     MAPE      MASE       ACF1
Training set -0.01116456 2.088196 1.608391 -142.0788 454.1568 0.4036136 0.07980073
ARIMA(0,0,12)(0,1,0)[12]

Coefficients:
        ma1     ma2     ma3      ma4      ma5     ma6     ma7     ma8      ma9     ma10     ma11     ma12
      0.065  0.0162  0.0062  -0.0019  -0.0092  0.0000  0.0091  0.0020  -0.0062  -0.0162  -0.0650  -0.9999
s.e.  0.043  0.0425  0.0442   0.0426   0.0429  0.0323  0.0429  0.0426   0.0442   0.0425   0.0431   0.0321

sigma^2 estimated as 9.504:  log likelihood=-728.11
AIC=1482.23   AICc=1483.59   BIC=1529.53

Training set error measures:
                   ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set 0.127625 2.953832 2.203956 169.0288 217.4336 0.5530662 0.02988073
```

11.Porównanie modelu ruchomej średniej z modelem autoregresji, analiza dobroci dopasowania

```
ARIMA(12,0,0) with non-zero mean

Coefficients:
         ar1     ar2     ar3      ar4      ar5     ar6      ar7      ar8     ar9    ar10    ar11     ar12    mean
      0.0927  0.0511  0.0767  -0.0369  -0.0618  0.0137  -0.0559  -0.0422  0.0457  0.0377  0.0300  -0.5175  0.0136
s.e.  0.0498  0.0508  0.0508   0.0509   0.0509  0.0516   0.0510   0.0509  0.0510  0.0508  0.0511   0.0507  0.1095

sigma^2 estimated as 6.651:  log likelihood=-688.6
AIC=1405.19   AICc=1406.71   BIC=1456.72

Training set error measures:
                     ME     RMSE      MAE      MPE     MAPE     MASE       ACF1
Training set -0.0196012 2.521146 1.956804 -130.573 456.1641 0.491045 0.05388818
ARIMA(12,0,0)(0,1,0)[12]

Coefficients:
         ar1     ar2    ar3      ar4      ar5      ar6      ar7      ar8     ar9    ar10    ar11     ar12
      0.0669  0.0288  0.053  -0.0284  -0.0394  -0.0002  -0.0527  -0.0552  0.0333  0.0373  0.0450  -0.6857
s.e.  0.0431  0.0438  0.044   0.0443   0.0440   0.0454   0.0440   0.0442  0.0443  0.0441  0.0444   0.0435

sigma^2 estimated as 14.59:  log likelihood=-773.01
AIC=1572.02   AICc=1573.38   BIC=1619.32

Training set error measures:
                     ME    RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set 0.09650005 3.65962 2.675325 78.26202 385.4588 0.6713526 0.06324382
ARIMA(24,0,0) with non-zero mean

Coefficients:
         ar1     ar2     ar3      ar4      ar5     ar6      ar7      ar8     ar9    ar10     ar11     ar12    ar13    ar14
      0.1055  0.0568  0.0957  -0.0271  -0.0735  0.0416  -0.0526  -0.0133  0.0493  0.0395  -0.0103  -0.6756  0.1074  0.0331
s.e.  0.0559  0.0565  0.0565   0.0570   0.0569  0.0579   0.0569   0.0570  0.0567  0.0567   0.0569   0.0571  0.0575  0.0582
        ar15     ar16     ar17    ar18    ar19    ar20    ar21    ar22     ar23     ar24    mean
      0.0388  -0.0271  -0.0653  0.0366  0.0141  0.0534  0.0319  0.0135  -0.1035  -0.3038  0.0078
s.e.  0.0584   0.0592   0.0592  0.0596  0.0592  0.0590  0.0589  0.0587   0.0583   0.0585  0.0869

sigma^2 estimated as 6.062:  log likelihood=-670.16
AIC=1392.32   AICc=1397.6   BIC=1488.01

Training set error measures:
                      ME     RMSE      MAE       MPE     MAPE      MASE       ACF1
Training set -0.02159083 2.354773 1.828477 -82.50449 423.5169 0.4588426 0.02828464
```

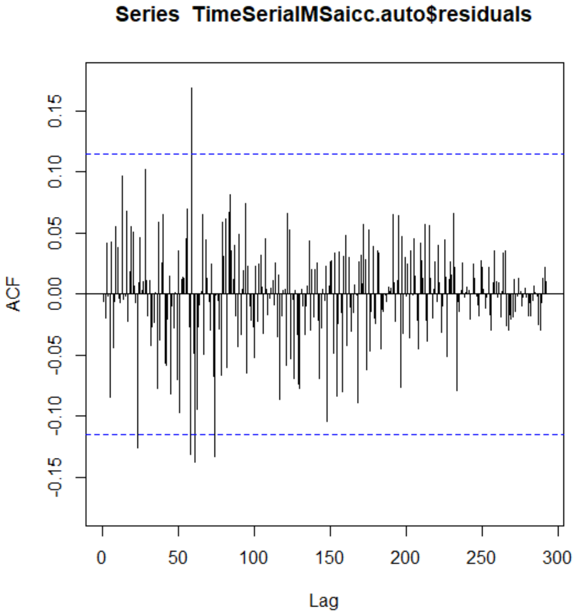
12.Wyznaczenie optymalnych modeli z wykorzystaniem funkcji auto.arima(), oraz ich porównanie na podstawie kryteriów dopasowania, wybór najlepszego

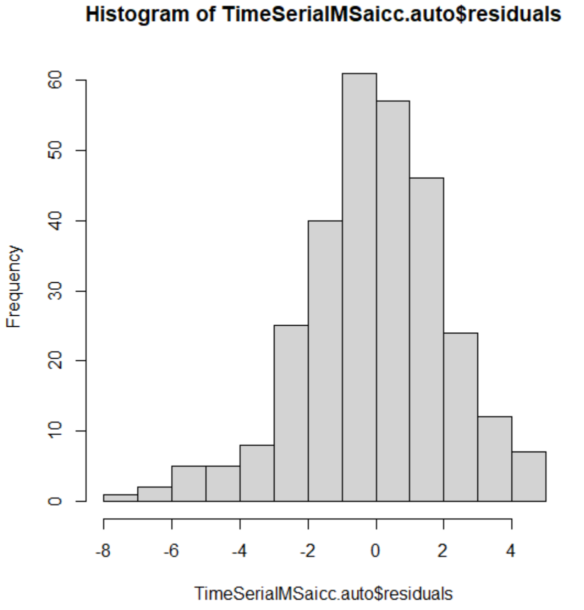
```
ARIMA(1,0,1)(2,0,2)[12] with zero mean

Coefficients:
         ar1      ma1    sar1     sar2    sma1    sma2
      0.6863  -0.5367  0.5299  -0.0758  -1.528  0.6059
s.e.  0.1844   0.2134  0.1899   0.0794   0.180  0.1704

sigma^2 estimated as 4.485:  log likelihood=-643.29
AIC=1300.57   AICc=1300.97   BIC=1326.34
ARIMA(1,0,0)(2,0,2)[12] with zero mean

Coefficients:
         ar1    sar1     sar2     sma1    sma2
      0.1665  0.5372  -0.0623  -1.5269  0.5956
s.e.  0.0616  0.3636   0.0858   0.3560  0.3296

sigma^2 estimated as 4.522:  log likelihood=-645.04
AIC=1302.07   AICc=1302.37   BIC=1324.15
```


```
ARIMA(1,0,0)(2,0,2)[12] with zero mean

Coefficients:
         ar1    sar1     sar2     sma1    sma2
      0.1665  0.5372  -0.0623  -1.5269  0.5956
s.e.  0.0616  0.3636   0.0858   0.3560  0.3296

sigma^2 estimated as 4.522:  log likelihood=-645.04
AIC=1302.07   AICc=1302.37   BIC=1324.15
```


Najlepszym modelem jest na podstawie kryteriów dobroci dopasowania: aic, aicc, bic (głownie dwóch ostatnich) - ARIMA(1,0,0)(2,0,2)[12], ponieważ kryteria tego modelu są najbliższe zeru. Współczynniki każdego z trzech powyższych modeli są bardzo zbliżone, a pierwszych dwóch nawet identyczne. W każdym z przypadków wyznaczono ten sam model.

13.Test stacjonarności automatycznie dobranego modelu



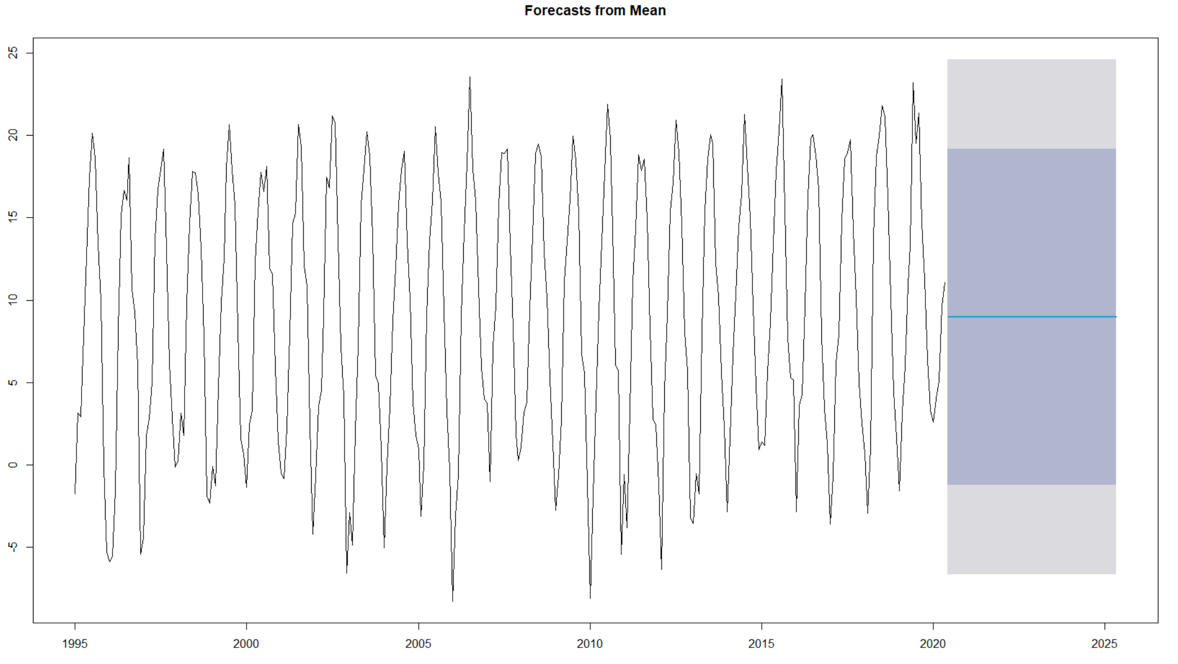


Jest to szereg stacjonarny, ponieważ żadna z wartości nie wystaje znacząco poza przedział ufności (na podstawie wcześniej wspomnianego wzoru). Histogram przedstawia rozkład normalny. Potwierdzone to jest także testem Shapiro-Wilka, gdzie wartość zmiennej p nie przekracza: 0,05.

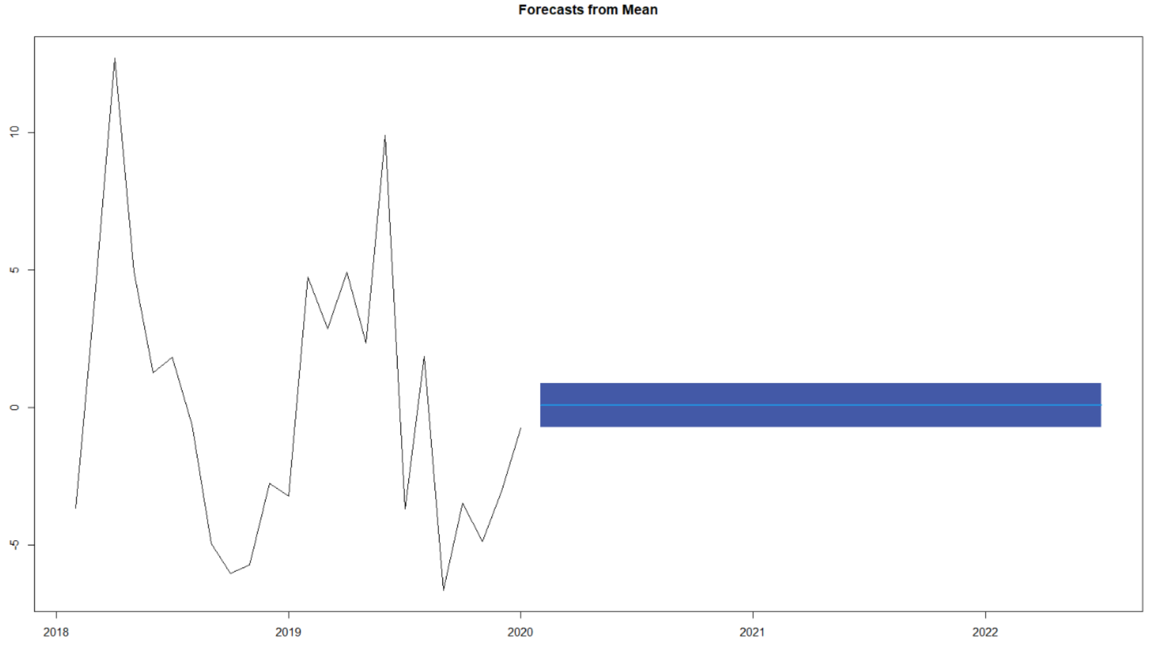
```
        Shapiro-Wilk normality test

data:  TimeSerialMSaicc.auto$residuals
W = 0.98575, p-value = 0.005308
```

14.Prognozowanie z wykorzystaniem metod naiwnych

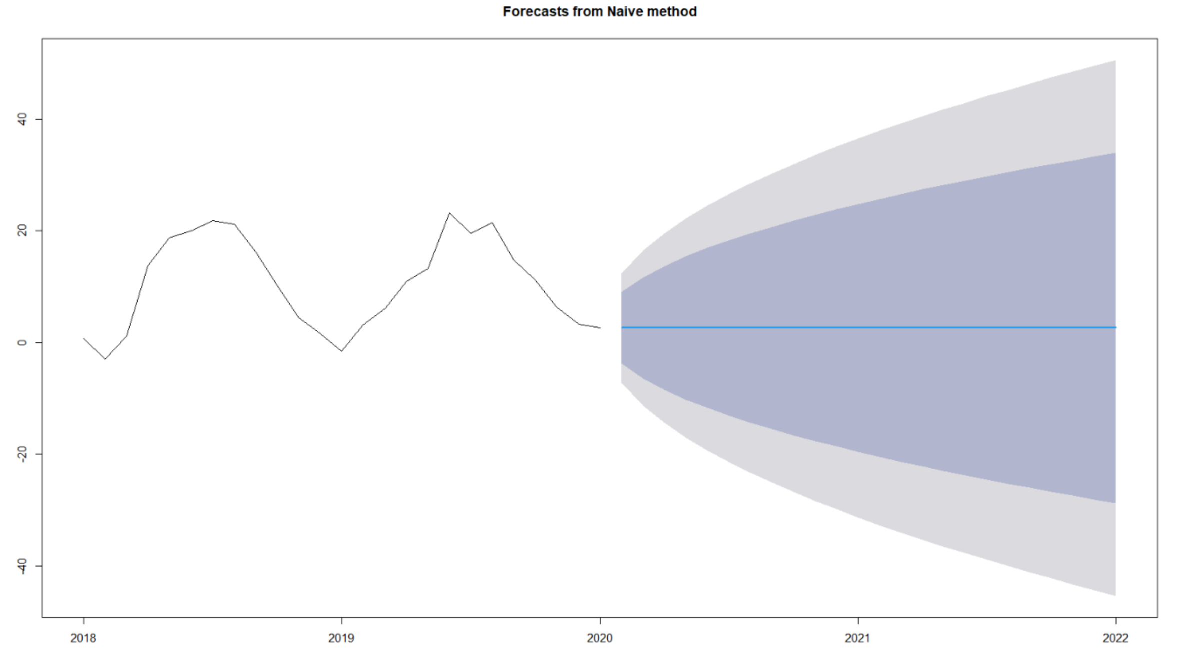
10) Prognoza oparta na średniej



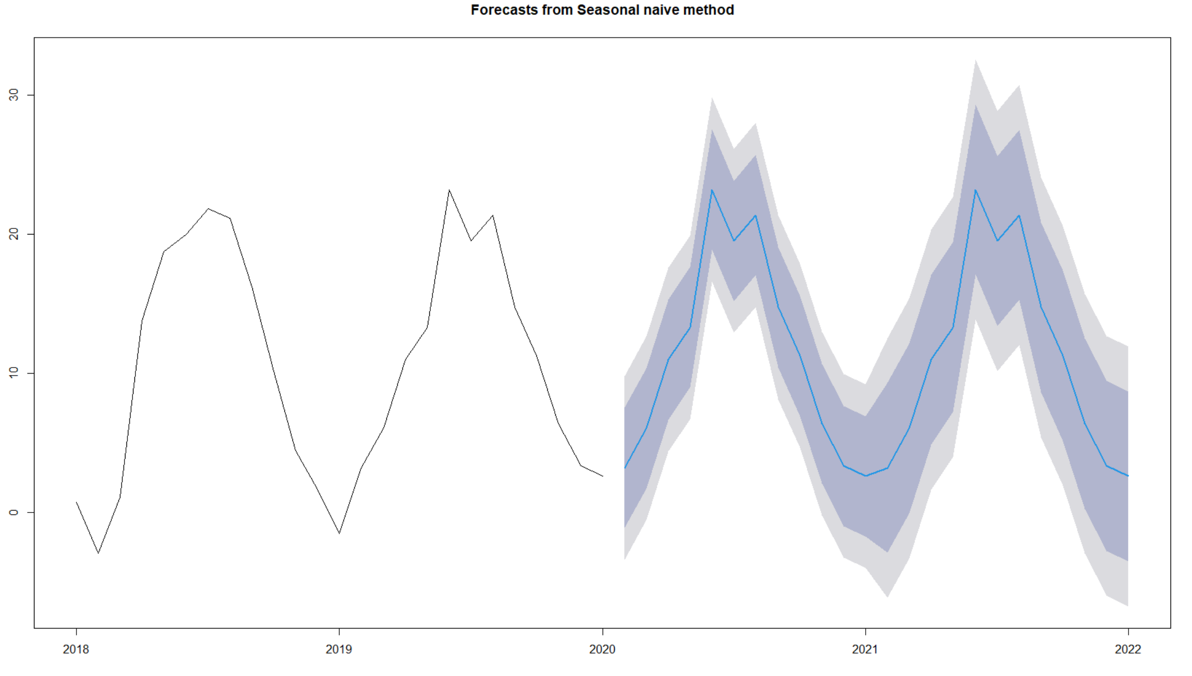
11) Prognoza po wyeliminowaniu sezonowości



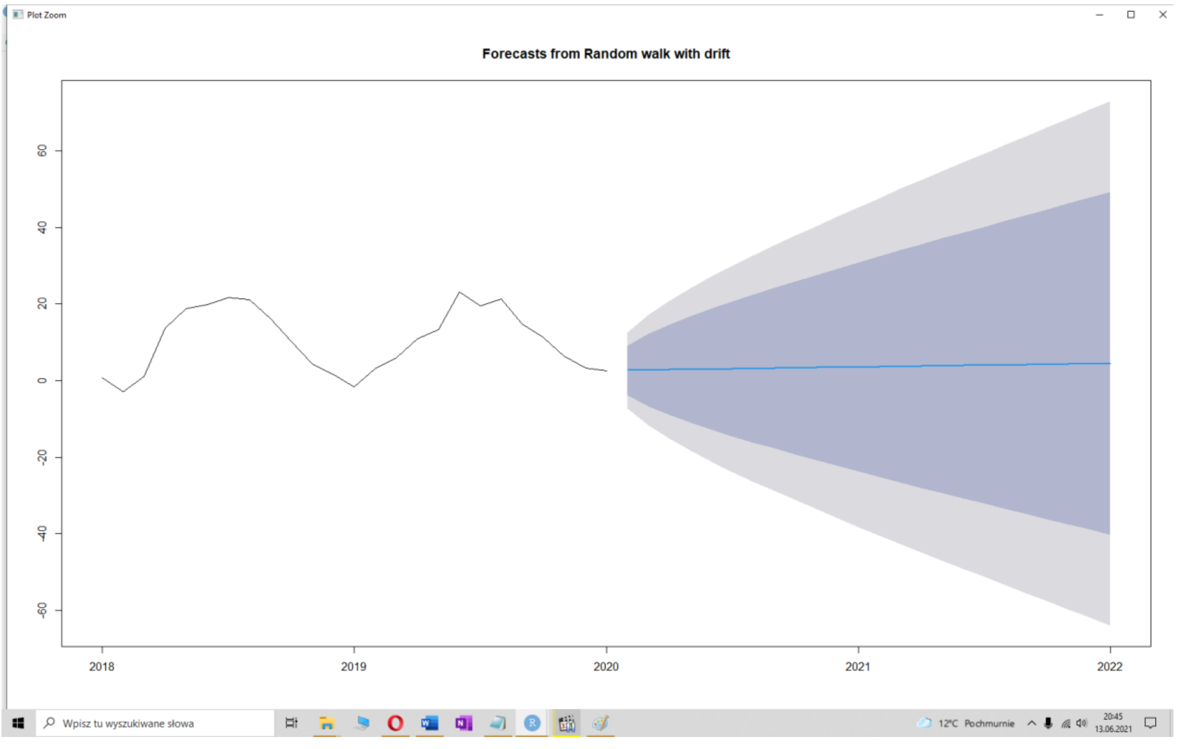
12) Prognozowanie naiwne z użyciem funkcji naive()



13) Prognozowanie z użyciem funkcji snaive()



14) Prognozowanie naiwne z dryfem



Najlepszą metodą w celu prognozy temperatur jest zastosowanie tutaj funkcji snaive() z powodu bardzo wyraźnej sezonowości szeregu.

Rozdział drugi – szereg z wyraźnym trendem

1. Wprowadzenie do analizowanych danych

Dane z szeregu z trendem dotyczą liczby szczepień na COVID-19 w państwach danego roku. Analizowane będą dane wyłącznie dla Polski od marca danego roku. Zostały one pobrane ze strony:

https://www.kaggle.com/gpreda/covid-world-vaccination-progress

1) Informacje o ramce danych:

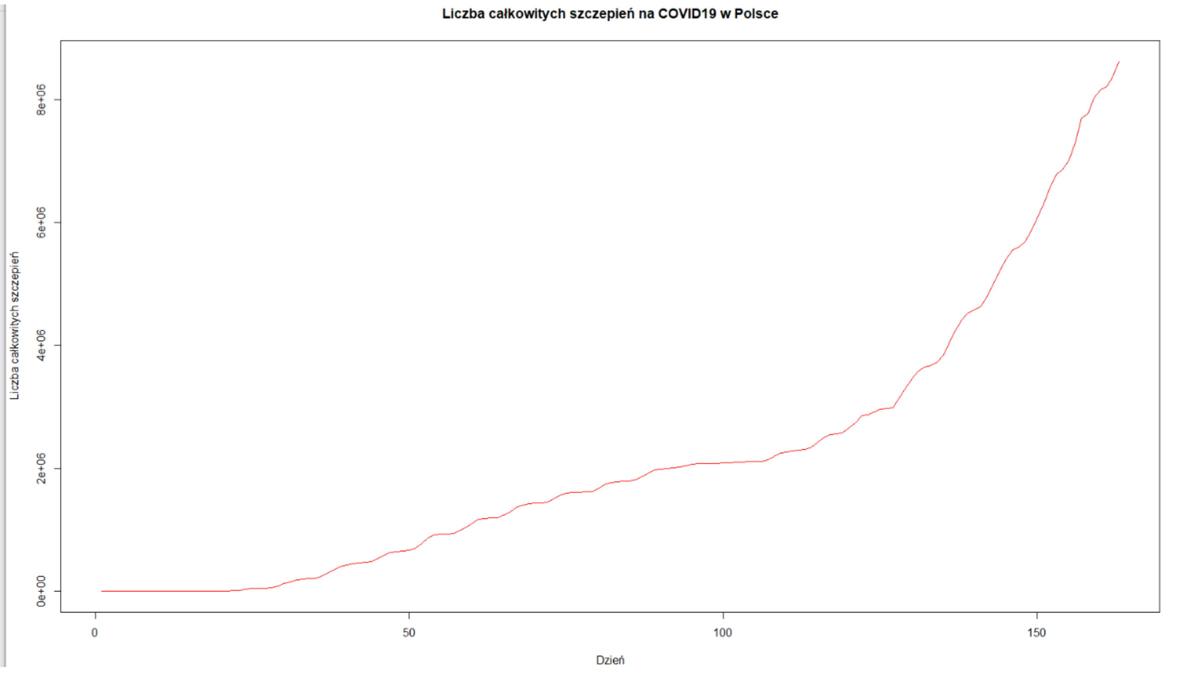
```
'data.frame':   23468 obs. of  15 variables:
 $ country                            : chr  "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
 $ iso_code                           : chr  "AFG" "AFG" "AFG" "AFG" ...
 $ date                               : chr  "2021-02-22" "2021-02-23" "2021-02-24" "2021-02-25" ...
 $ total_vaccinations                 : num  0 NA NA NA NA NA 8200 NA NA NA ...
 $ people_vaccinated                  : num  0 NA NA NA NA NA 8200 NA NA NA ...
 $ people_fully_vaccinated            : num  NA NA NA NA NA NA NA NA NA NA ...
 $ daily_vaccinations_raw             : num  NA NA NA NA NA NA NA NA NA NA ...
 $ daily_vaccinations                 : num  NA 1367 1367 1367 1367 ...
 $ total_vaccinations_per_hundred     : num  0 NA NA NA NA NA 0.02 NA NA NA ...
 $ people_vaccinated_per_hundred      : num  0 NA NA NA NA NA 0.02 NA NA NA ...
 $ people_fully_vaccinated_per_hundred: num  NA NA NA NA NA NA NA NA NA NA ...
 $ daily_vaccinations_per_million     : num  NA 35 35 35 35 35 35 41 46 52 ...
 $ vaccines                           : chr  "Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing" "Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing" "O
xford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing" "Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing" ...
 $ source_name                        : chr  "World Health Organization" "World Health Organization" "World Health Organization" "World Health Organization" ...
 $ source_website                     : chr  "https://covid19.who.int/" "https://covid19.who.int/" "https://covid19.who.int/" "https://covid19.who.int/" ...
```

2) Przykładowe rekordy danych:

```
      country iso_code       date total_vaccinations people_vaccinated people_fully_vaccinated daily_vaccinations_raw daily_vaccinations
1 Afghanistan      AFG 2021-02-22                  0                 0                      NA                     NA                 NA
2 Afghanistan      AFG 2021-02-23                 NA                NA                      NA                     NA               1367
3 Afghanistan      AFG 2021-02-24                 NA                NA                      NA                     NA               1367
4 Afghanistan      AFG 2021-02-25                 NA                NA                      NA                     NA               1367
5 Afghanistan      AFG 2021-02-26                 NA                NA                      NA                     NA               1367
6 Afghanistan      AFG 2021-02-27                 NA                NA                      NA                     NA               1367
  total_vaccinations_per_hundred people_vaccinated_per_hundred people_fully_vaccinated_per_hundred daily_vaccinations_per_million
1                              0                             0                                  NA                             NA
2                             NA                            NA                                  NA                             35
3                             NA                            NA                                  NA                             35
4                             NA                            NA                                  NA                             35
5                             NA                            NA                                  NA                             35
6                             NA                            NA                                  NA                             35
                                                vaccines               source_name           source_website
1 Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing World Health Organization https://covid19.who.int/
2 Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing World Health Organization https://covid19.who.int/
3 Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing World Health Organization https://covid19.who.int/
4 Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing World Health Organization https://covid19.who.int/
5 Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing World Health Organization https://covid19.who.int/
6 Oxford/AstraZeneca, Pfizer/BioNTech, Sinopharm/Beijing World Health Organization https://covid19.who.int/
```

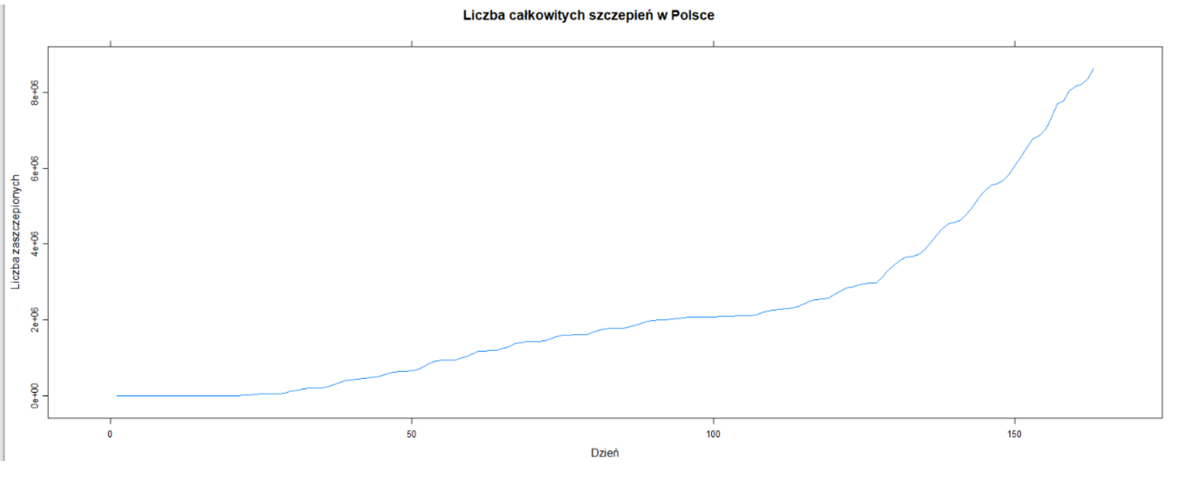
2. Utworzenie szeregów czasowych z średnimi: diennymi, miesięcznymi, kwartałowymi (Po odpowiednim „wycięciu” potrzebnych danych, interpolacji wartości brakujących)

```
Wybór kraju dla analizy danych (np. "Poland", "Palestine")
> sort(unique(my_data$country))
  [1] "Afghanistan"                      "Albania"                          "Algeria"                          "Andorra"
  [5] "Angola"                           "Anguilla"                         "Antigua and Barbuda"              "Argentina"
  [9] "Armenia"                          "Aruba"                            "Australia"                        "Austria"
 [13] "Azerbaijan"                       "Bahamas"                          "Bahrain"                          "Bangladesh"
 [17] "Barbados"                         "Belarus"                          "Belgium"                          "Belize"
 [21] "Benin"                            "Bermuda"                          "Bhutan"                           "Bolivia"
 [25] "Bonaire Sint Eustatius and Saba"  "Bosnia and Herzegovina"           "Botswana"                         "Brazil"
 [29] "British Virgin Islands"           "Brunei"                           "Bulgaria"                         "Burkina Faso"
 [33] "Cambodia"                         "Cameroon"                         "Canada"                           "Cape Verde"
 [37] "Cayman Islands"                   "Central African Republic"         "Chile"                            "China"
 [41] "Colombia"                         "Comoros"                          "Congo"                            "Cook Islands"
 [45] "Costa Rica"                       "Cote d'Ivoire"                    "Croatia"                          "Cuba"
 [49] "Curacao"                          "Cyprus"                           "Czechia"                          "Democratic Republic of Congo"
 [53] "Denmark"                          "Djibouti"                         "Dominica"                         "Dominican Republic"
 [57] "Ecuador"                          "Egypt"                            "El Salvador"                      "England"
 [61] "Equatorial Guinea"                "Estonia"                          "Eswatini"                         "Ethiopia"
```

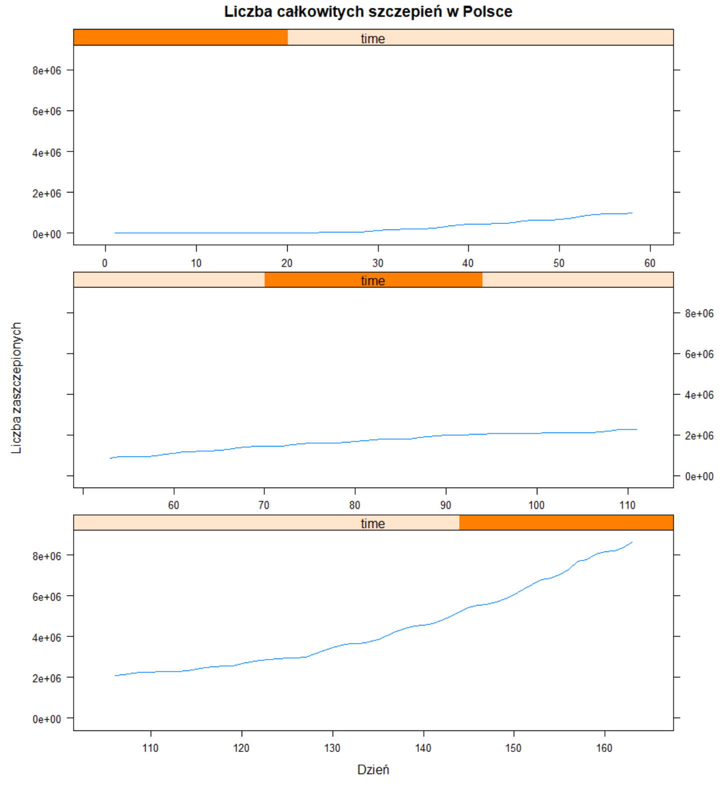


3. Reprezentacja szeregu czasowego na różnego rodzaju wykresach

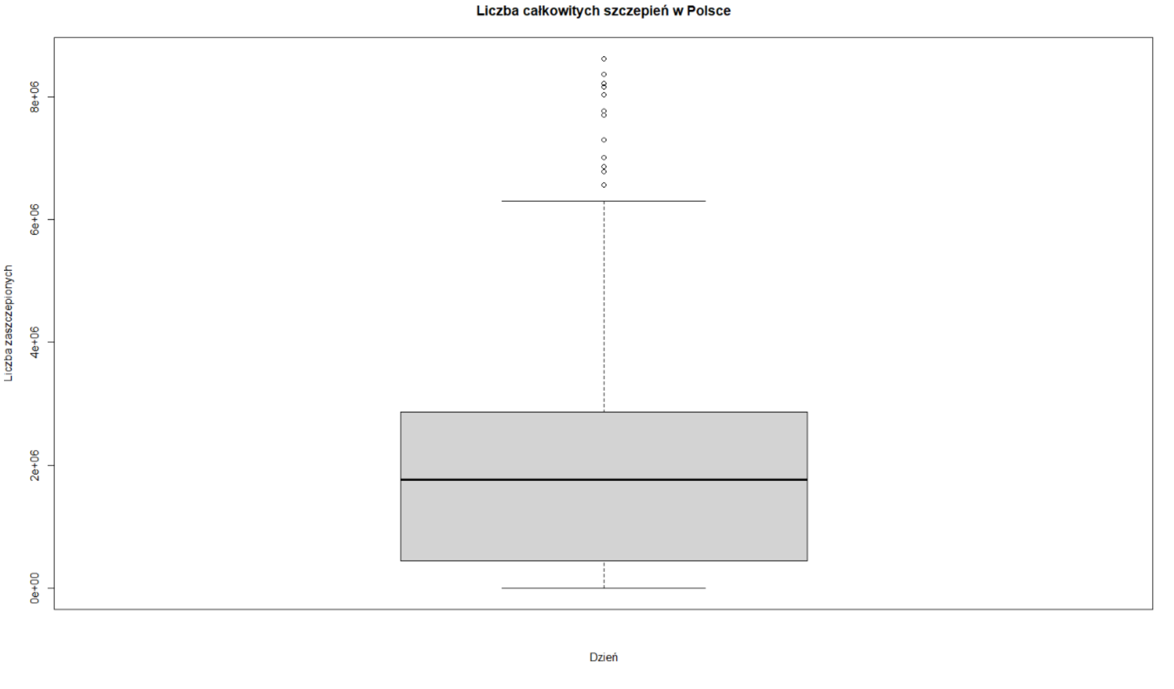
1) Wykres z użyciem xyplot()



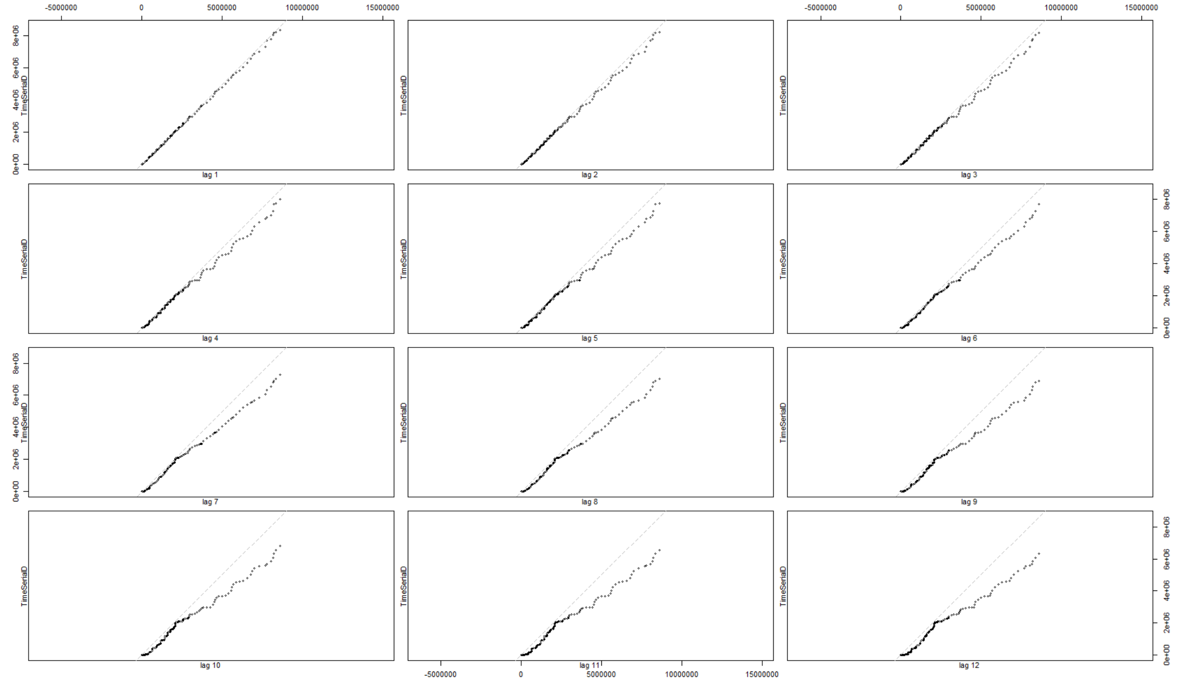
2) Wykres panelowy



3) Wykres pudełkowy

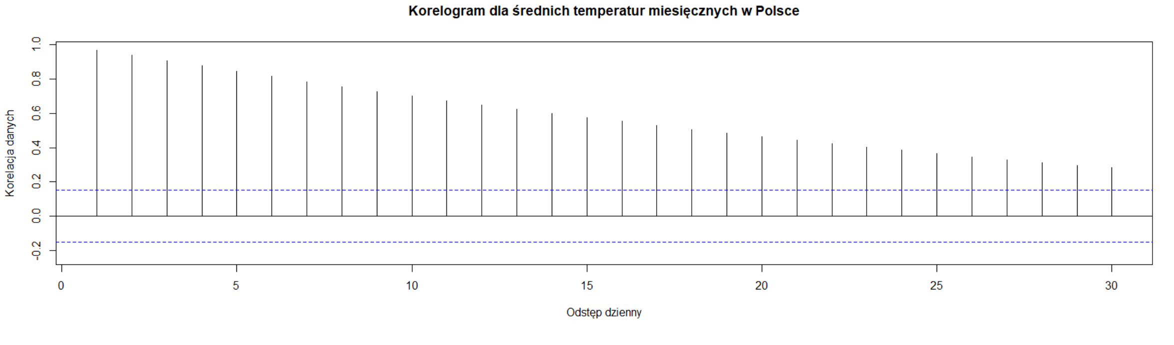


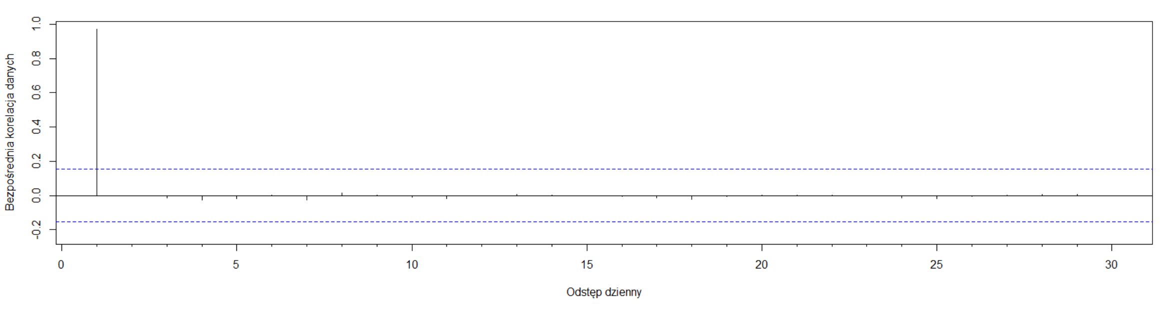
4) Wykres rozrzutu wartości opóźnionych



Dla danego szeregu widzimy największą korelację (zależność) danych dla opóźnienia z krokiem pierwszym, co pokazuje wyraźny trend wzrostowy. Z każdym kolejnym krokiem korelacja danych maleje, co oznacza gorzej dobrany współczynnik opóźnienia.

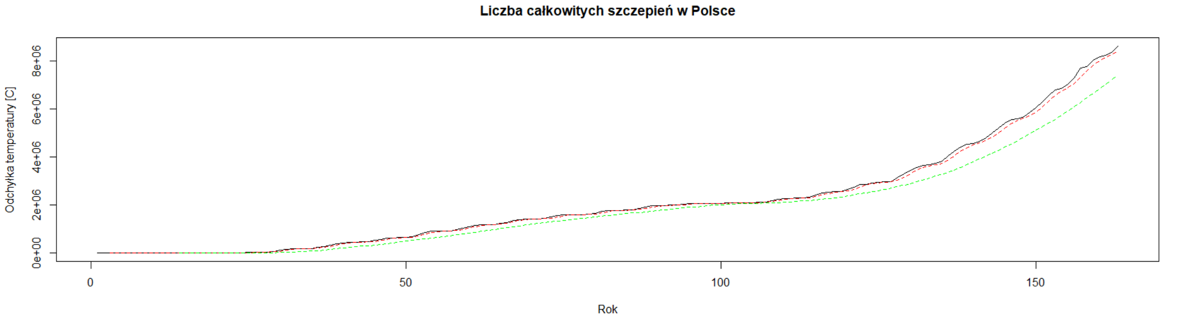
5) Wykresy funkcji autokorelacji – Acf, cząstkowej autokorelacji – Pacf





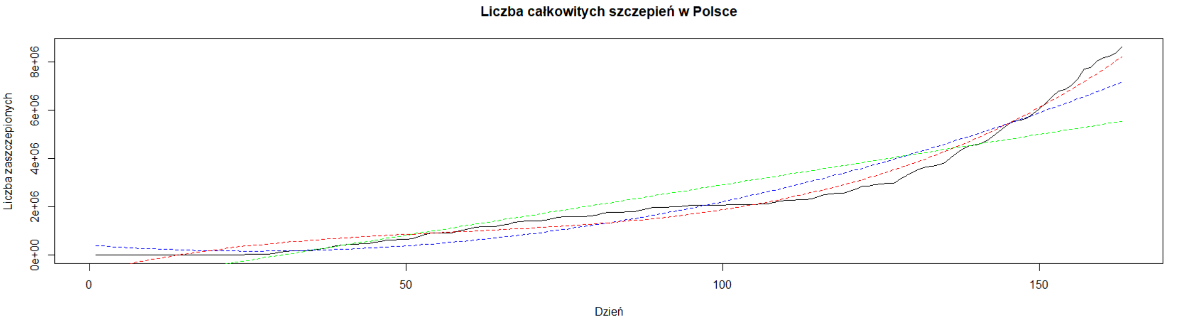
Widoczna jest wyrźny trend wzrostowy z opóźnieniem – jeden poprzez łagodne, regularne zmiany na wykresie funkcji Acf , oraz dzięki dużej „szpilce” przy odstępie sezonowym. Brak trendów z innymi opóźnieniami.

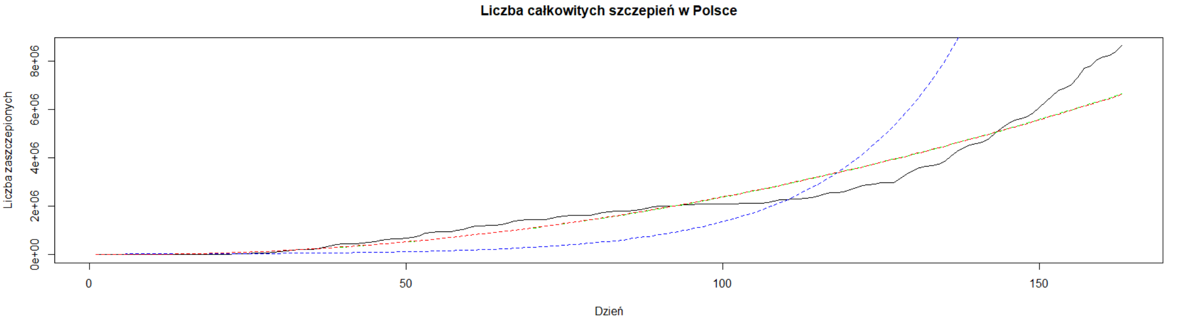
6) Wygładzenie danych poprzez użycie średniej ruchomej



Wykres pokazuje wygładzenie szeregu dziennego dla odstępu: trzy/czternasto-dniowego, oznaczonych kolejno według wielkości spłaszczenia kolorami czerwonym i zielonym.

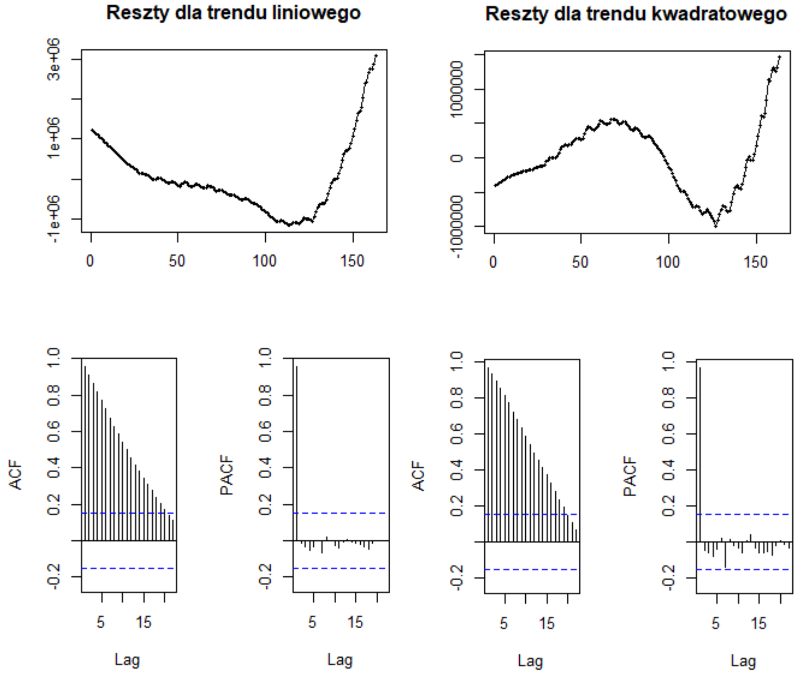
4. Dekompozycje na podstawie modelu regresji: trend liniowy/wielomianowy, transformacje Boxa-Coxa

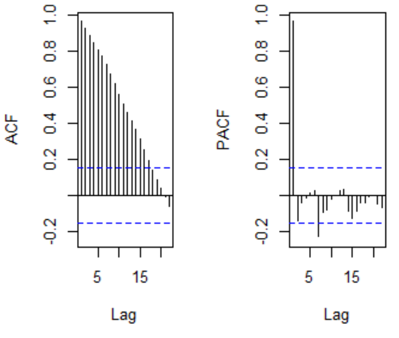




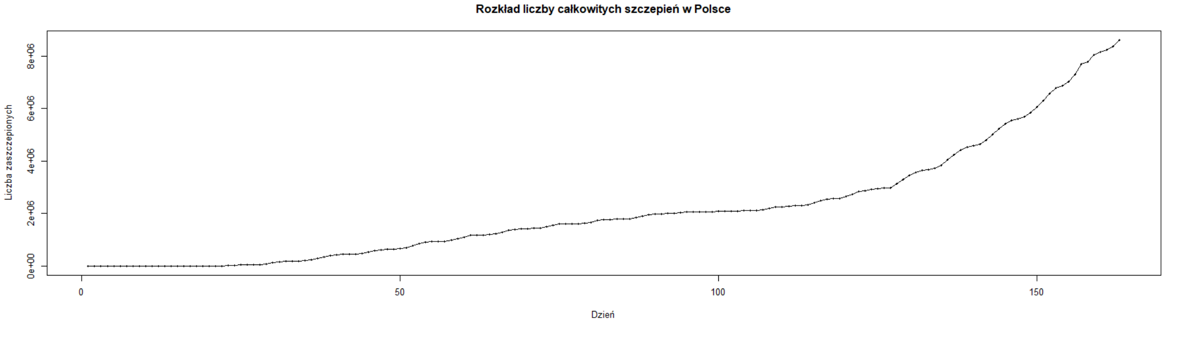
Wykres pierwszy pokazuje wykryte trendy kolejnych stopni: liniowy, kwadratowy, oraz stopnia trzeciego, oznaczonych kolejno kolorami: zielonym, czerwonym, niebieskim. Drugi z wykresów przedstawia transformacje Boxa-Coxa: pierwiastkową, logarytmiczną i z automatycznie dobranym współczynnikiem lambda, oznaczonych kolejno kolorami: zielonym, czerwonym, niebieskim. Na ich podstawie możemy stwierdzić, że koniecznym założeniem jest tutaj trend wielomianowy, najprawdopodobniej stopnia trzeciego.

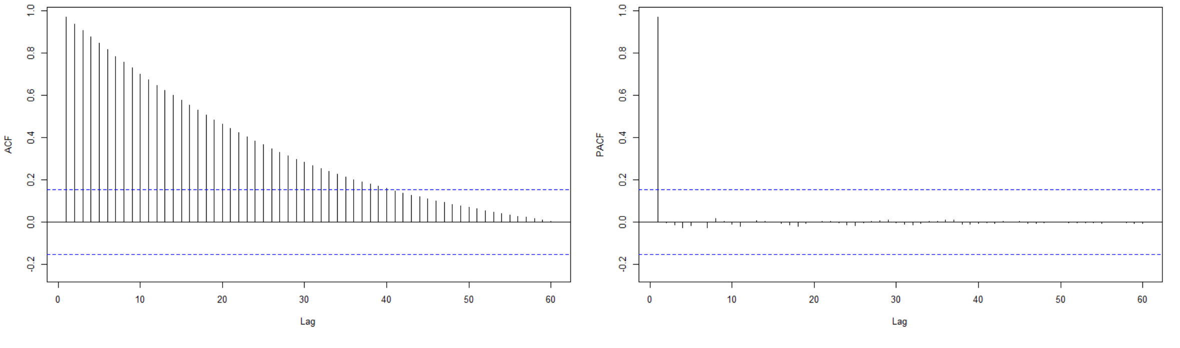
Porównamy reszty z różnych metod dekompozycji na podstawie modelu regresji:

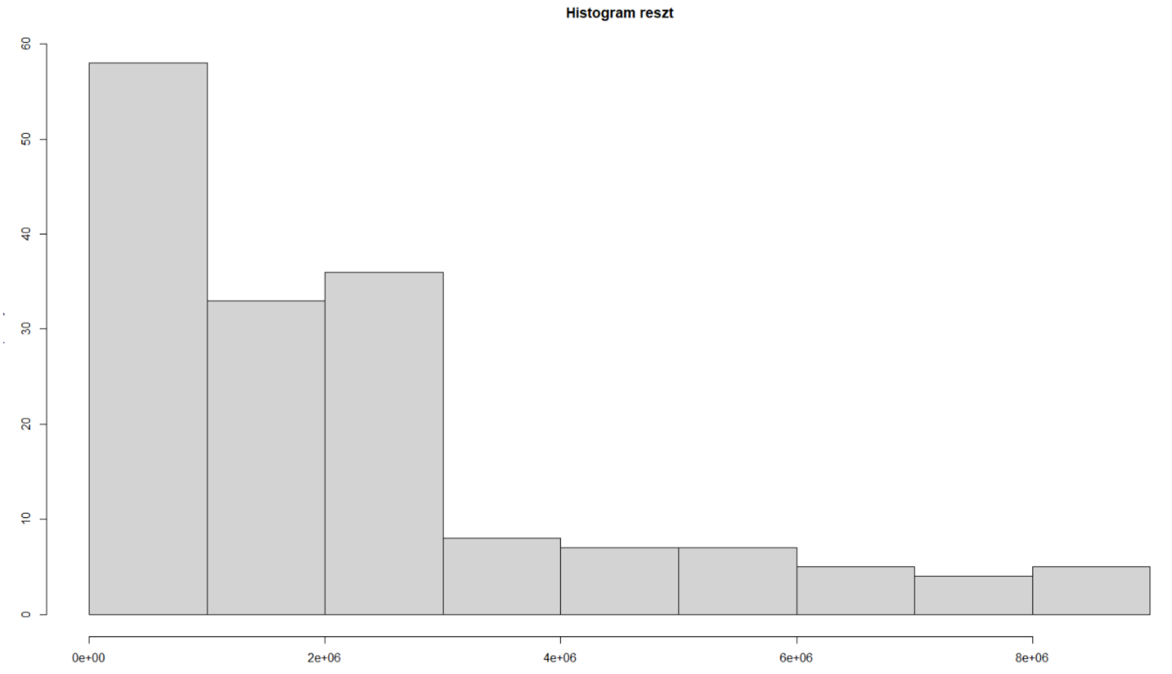




5. Uczynienie szeregu stacjonarnym – eliminacja trendu, sezonowości poprzez różnicowanie i stabilizacja wariancji poprzez transformację Boxa-Coxa

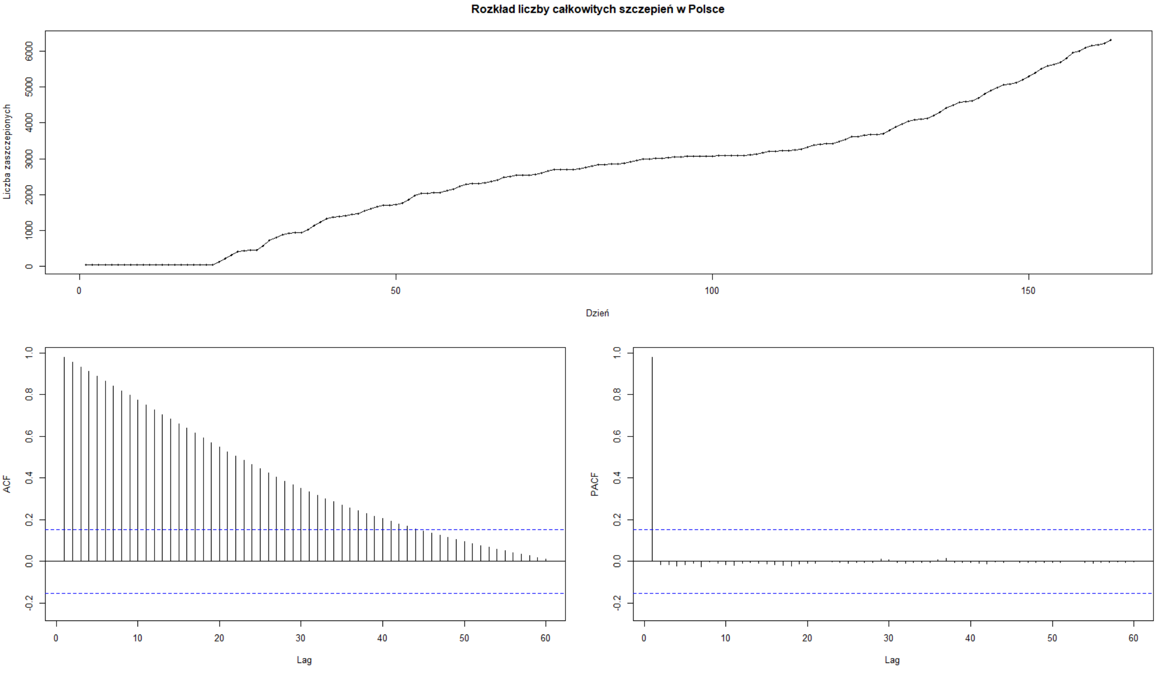




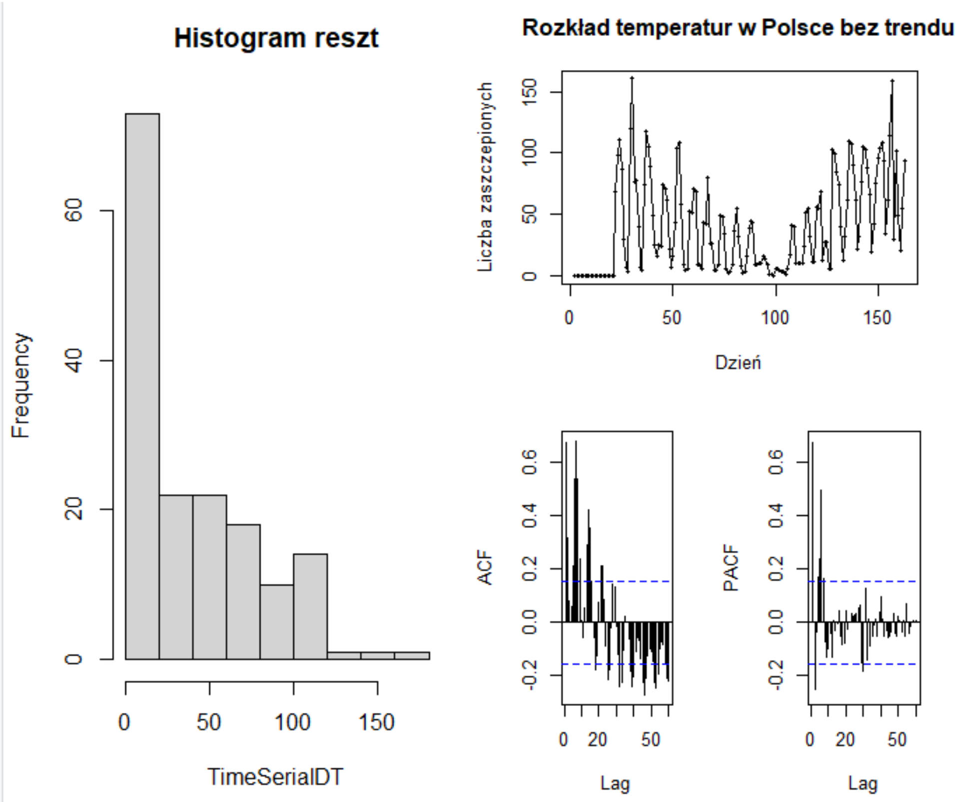



```
> ndiffs(TimeSerialDBox) # Ilość potrzebnych różnicowań rzędu różnicowania z opóźnieniem wartości: jeden
[1] 2
```

Po zastosowaniu transformacji Boxa-Coxa z automatycznie dobranym współczynnikiem lambda szereg i jego histogram, oraz wykresu autokorelacji wyglądają następująco:

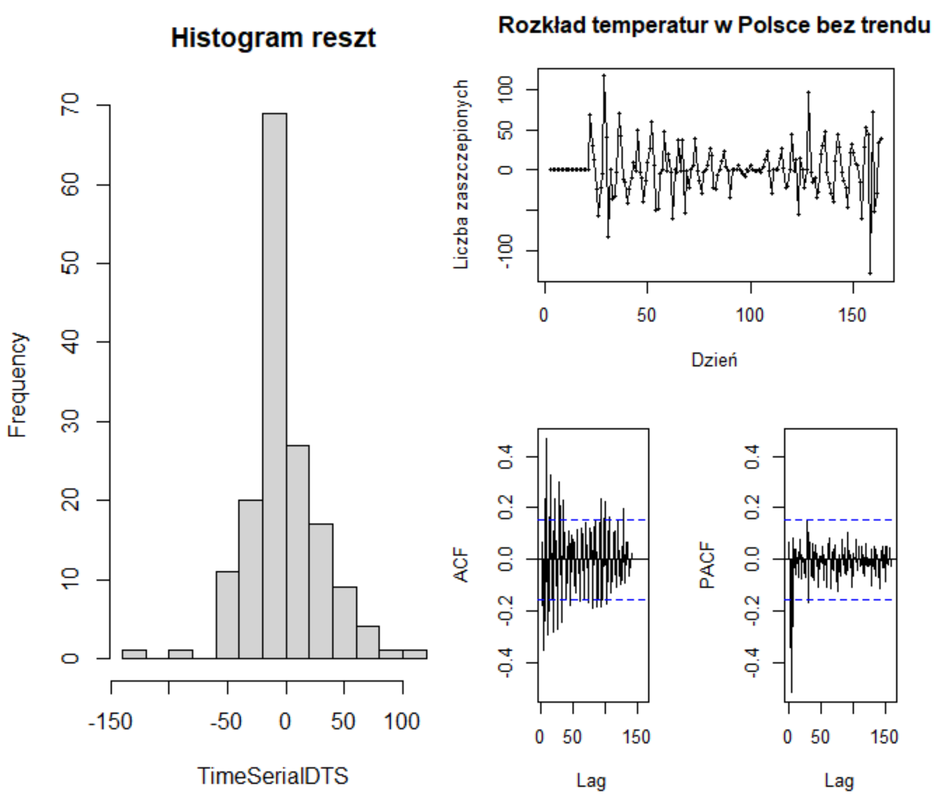


Po pierwszym zróżnicowaniu danych z opóźnieniem pierwszym wygląda to, jak na następnej stronie:

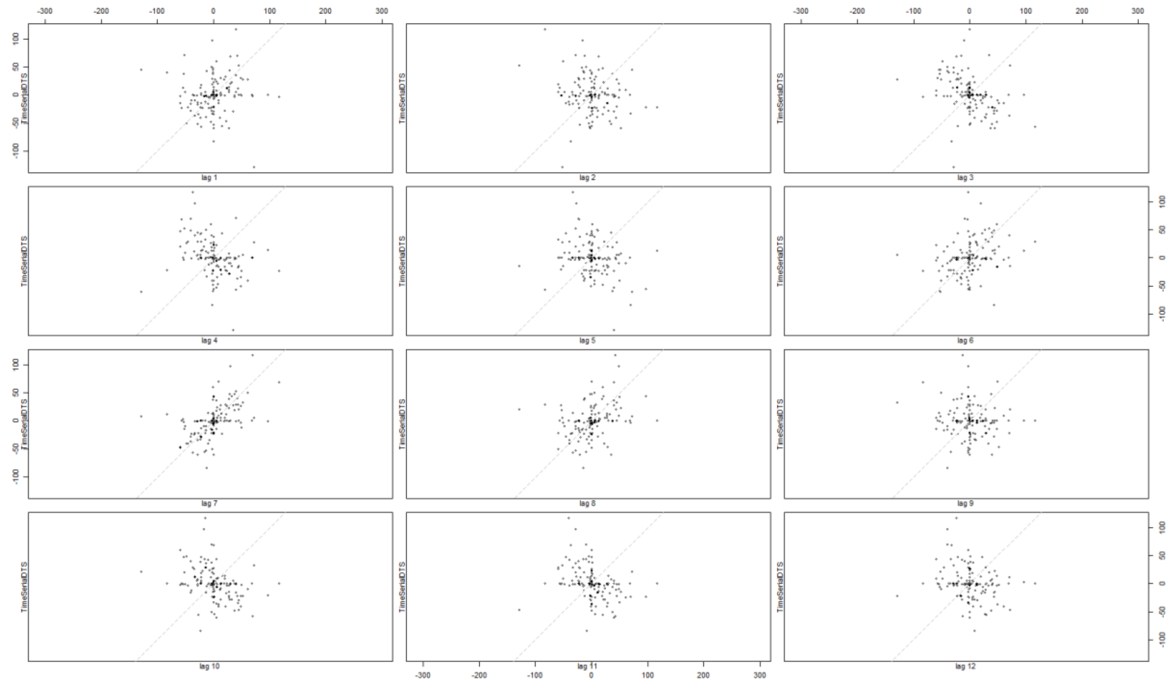



```
> ndiffs(TimeSerialDT) # Ilość potrzebnych różnicowań rzędu różnicowania z opóźnieniem wartości: jeden
[1] 1
```

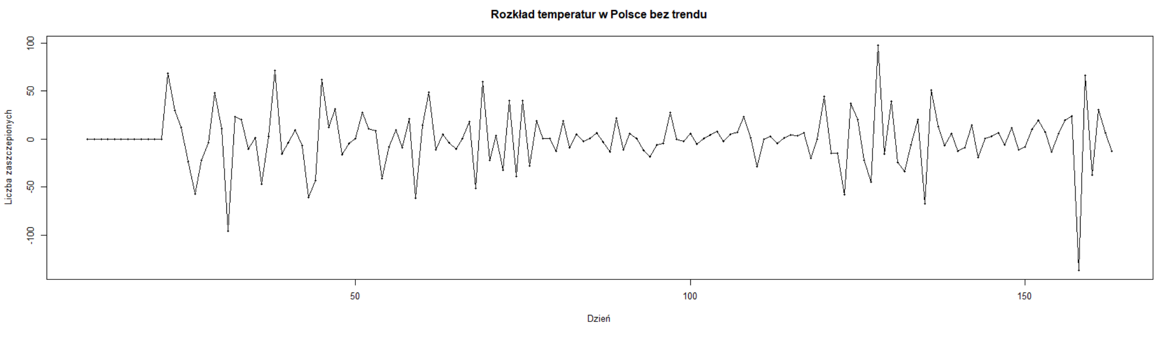
Po drugim zróżnicowaniu danych z opóźnieniem pierwszym wygląda to następująco:

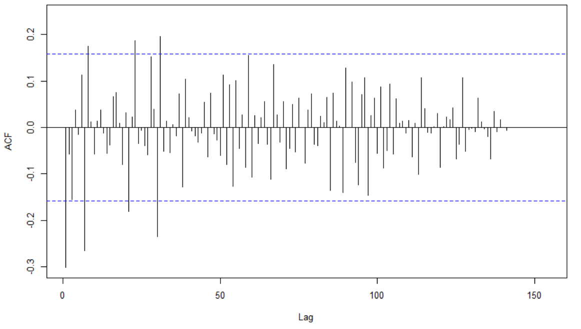


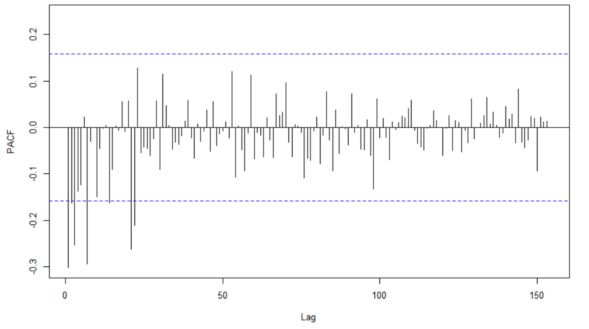
Widoczna jednak dalej jest pewna sezonowość reszt. W celu doboru rzędu różnicowania sprawdzimy wykresy rozrzutu.



Po zróżnicowaniu z opóźnieniem – siedem wygląda to następująco:

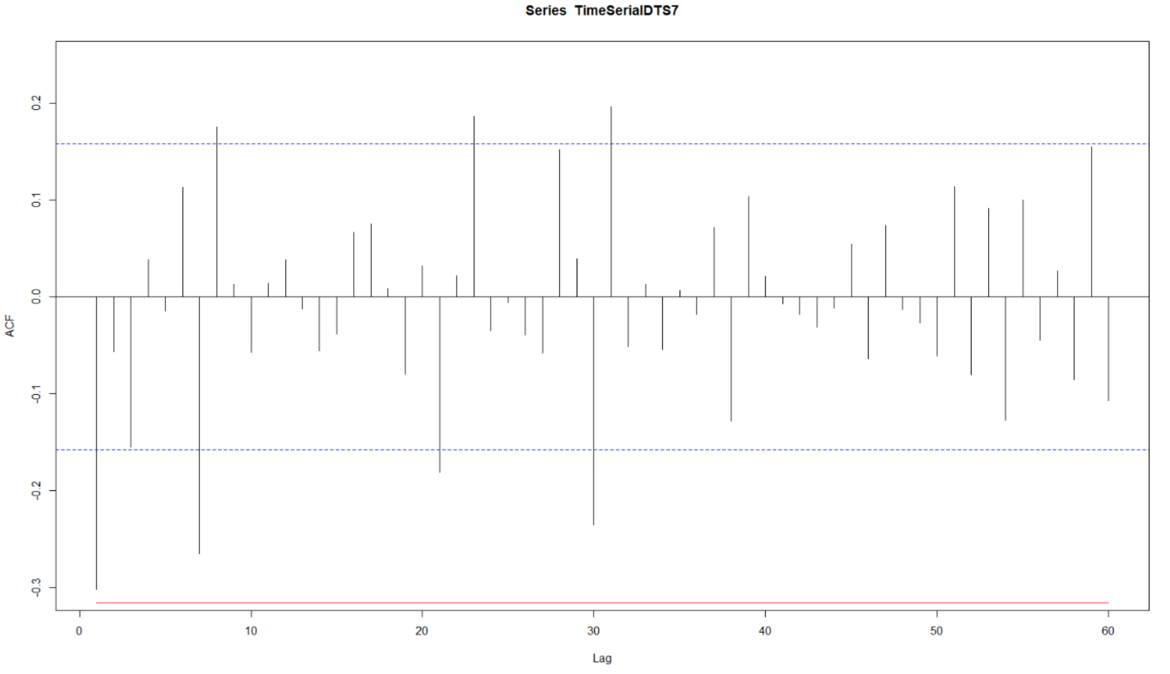






Po przyglądnięciu się wykresom funkcji autokorelacji bezpośredniej i cząstkowej możemy założyć pewne modele z konkretnymi współczynnikami w celu identyfikacji szeregu. Wartości wykraczające poza przedział ufności na wykresie Acf to: 1,7, 8, 21, 23, 30, 31; dla wykresu Pacf to: 1, 3, 7, minimalnie 14, 21, 22. Poprawnym byłoby założenie więc modeli np.: MA(1), MA(31), AR(1), AR(7).

6. Sprawdzenie stacjonarności otrzymanego szeregu



Brak widocznych wyraźnych odchyłek od przedziału ufności, możemy więc przyjąć dany szereg za stacjonarny. Wartość maksymalnej odchyłki wyznaczyliśmy ze wzoru: $x = \frac{1.96}{\sqrt{\text{dł.szeregu}}}$

7. Wyznaczenie współczynników modelu autoregresji i porównanie dopasowania różnymi metodami estymacji

```
Call:
ar(x = TimeSerialDNM, aic = FALSE, order.max = 1, method = c("yule-walker"))

Coefficients:
      1
-0.3017

Order selected 1  sigma^2 estimated as  771.6
Call:
ar(x = TimeSerialDNM, aic = FALSE, order.max = 1, method = c("burg"))

Coefficients:
      1
-0.3019

Order selected 1  sigma^2 estimated as  761.5
Call:
ar(x = TimeSerialDNM, aic = FALSE, order.max = 1, method = c("ols"))

Coefficients:
      1
-0.3021

Intercept: 0.02444 (2.238)

Order selected 1  sigma^2 estimated as  766.5
Call:
ar(x = TimeSerialDNM, aic = FALSE, order.max = 1, method = c("mle"))

Coefficients:
      1
-0.3002

Order selected 1  sigma^2 estimated as  761.5
Call:
ar(x = TimeSerialDNM, aic = FALSE, order.max = 1, method = c("yw"))

Coefficients:
      1
-0.3017

Order selected 1  sigma^2 estimated as  771.6
```

Automatyczny dobór rzędu dzięki kryterium „aic":

```
Call:
ar(x = TimeSerialDNM, aic = TRUE, order.max = 100, method = c("yule-walker"))

Coefficients:
      1        2        3        4        5        6        7
-0.4339  -0.3527  -0.3874  -0.2801  -0.2079  -0.1070  -0.2937

Order selected 7  sigma^2 estimated as  646.1
```

8. Wyznaczenie współczynników modelu ruchomej średniej z użyciem funkcji Arima()

```
ARIMA(0,0,1) with non-zero mean

Coefficients:
          ma1    mean
      -0.6073  0.0219
s.e.   0.1164  0.8493

sigma^2 estimated as 715:  log likelihood=-723.81
AIC=1453.62   AICc=1453.78   BIC=1462.73

Training set error measures:
                      ME     RMSE      MAE      MPE     MAPE      MASE      ACF1
Training set 0.004000951 26.56592 17.68786 24.05806 311.4821 0.5644599 0.1493788
```

```
ARIMA(0,0,7) with non-zero mean

Coefficients:
          ma1      ma2      ma3      ma4     ma5     ma6      ma7    mean
      -0.2768  -0.3701  -0.3430  -0.0049  0.1319  0.4055  -0.5425  0.1010
s.e.   0.0744   0.0808   0.0793   0.0722  0.0832  0.0857   0.0809  0.1115

sigma^2 estimated as 569.6:  log likelihood=-706.35
AIC=1430.69   AICc=1431.94   BIC=1458.02

Training set error measures:
                     ME     RMSE      MAE     MPE     MAPE      MASE       ACF1
Training set -0.2527391 23.23713 15.51878 137.842 370.8928 0.4952398 -0.1249344
```

9. Porównanie modelu ruchomej średniej z modelem autoregresji, analiza dobroci dopasowania

```
ARIMA(1,0,0) with non-zero mean

Coefficients:
          ar1    mean
      -0.3001  0.0190
s.e.   0.0766  1.7129

sigma^2 estimated as 771.5:  log likelihood=-729.48
AIC=1464.97   AICc=1465.13   BIC=1474.08

Training set error measures:
                      ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 0.000248016 27.59571 18.05437 34.48005 240.9512 0.5761562 -0.05071994
```

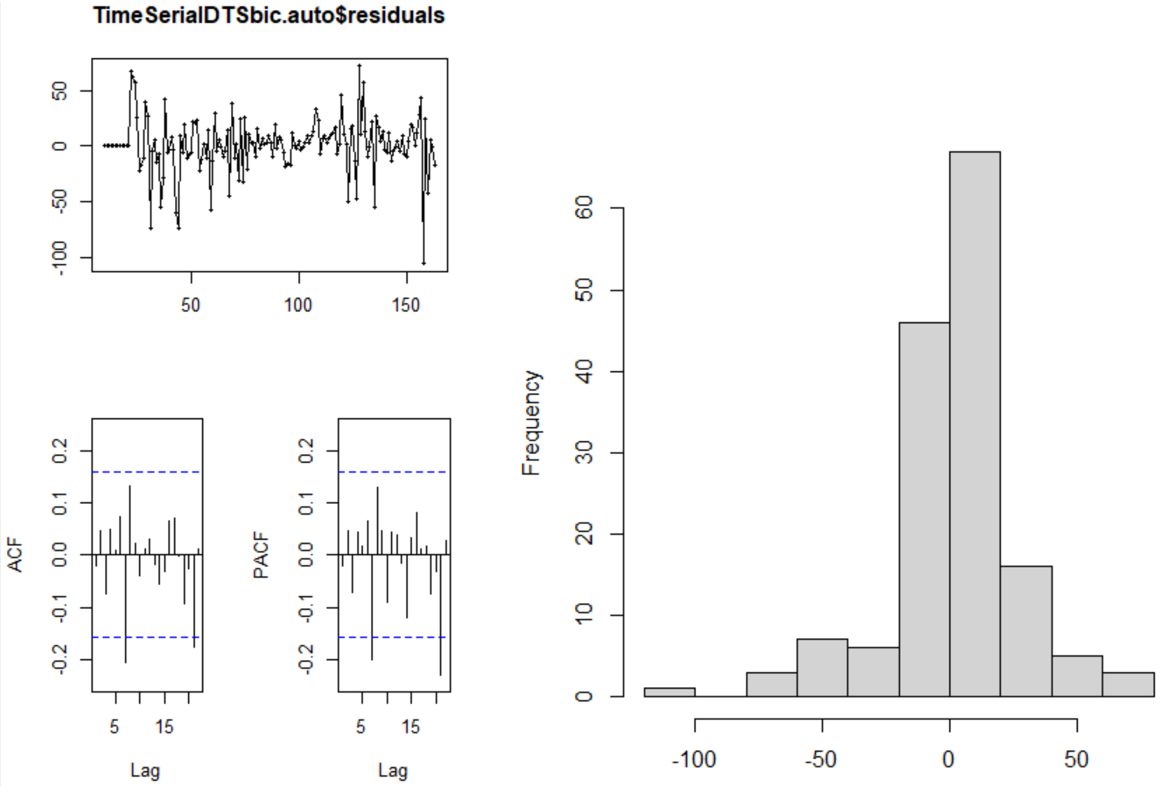
```
ARIMA(7,0,0) with non-zero mean

Coefficients:
          ar1      ar2      ar3      ar4      ar5      ar6      ar7    mean
      -0.4237  -0.3549  -0.3765  -0.2759  -0.2221  -0.0877  -0.3400  0.0341
s.e.   0.0760   0.0829   0.0854   0.0886   0.0857   0.0870   0.0811  0.6502

sigma^2 estimated as 631.1:  log likelihood=-711.56
AIC=1441.12   AICc=1442.37   BIC=1468.46

Training set error measures:
                      ME     RMSE      MAE      MPE     MAPE      MASE         ACF1
Training set 0.003049892 24.46013 15.93834 79.35744 203.1509 0.5086289 -0.003157649
```

10. Wyznaczenie optymalnych modeli z wykorzystaniem funkcji auto.arima(), oraz ich porównanie na podstawie kryteriów dopasowania, wybór najlepszego

```
ARIMA(1,0,1) with zero mean

Coefficients:
         ar1      ma1
      0.4416  -0.9686
s.e.  0.0813   0.0304

sigma^2 estimated as 629.4:  log likelihood=-714.7
AIC=1435.41   AICc=1435.57   BIC=1444.52

Training set error measures:
                  ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 1.21456 24.92532 16.50394 149.1697 213.2714 0.5266785 -0.02032933
ARIMA(1,0,1) with zero mean

Coefficients:
         ar1      ma1
      0.4416  -0.9686
s.e.  0.0813   0.0304

sigma^2 estimated as 629.4:  log likelihood=-714.7
AIC=1435.41   AICc=1435.57   BIC=1444.52

Training set error measures:
                  ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 1.21456 24.92532 16.50394 149.1697 213.2714 0.5266785 -0.02032933
ARIMA(1,0,1) with zero mean

Coefficients:
         ar1      ma1
      0.4416  -0.9686
s.e.  0.0813   0.0304

sigma^2 estimated as 629.4:  log likelihood=-714.7
AIC=1435.41   AICc=1435.57   BIC=1444.52

Training set error measures:
                  ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 1.21456 24.92532 16.50394 149.1697 213.2714 0.5266785 -0.02032933
```

Najlepszym modelem jest więc - ARIMA(1,0,1) wyznaczona identycznie (z identycznymi współczynnikami) przez każde z kryteriów dobroci dopasowania: aic, aicc, bic.

11. Test stacjonarności automatycznie dobranego modelu

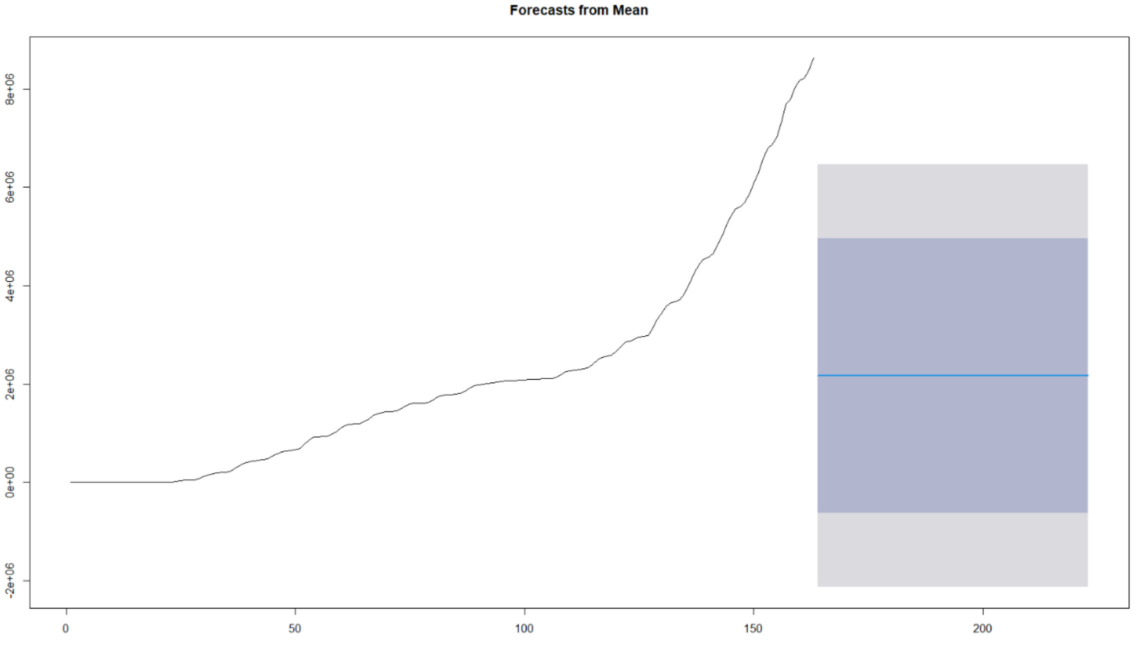


Jest to szereg stacjonarny, ponieważ żadna z wartości nie wystaje znacząco poza przedział ufności (na podstawie wcześniej wspomnianego wzoru). Histogram jest bardzo zbliżony do rozkładu normalnego. Test Shapiro-Wilka potwierdza, że nim jest, ponieważ wartość zmiennej p nie przekracza: 0,05.

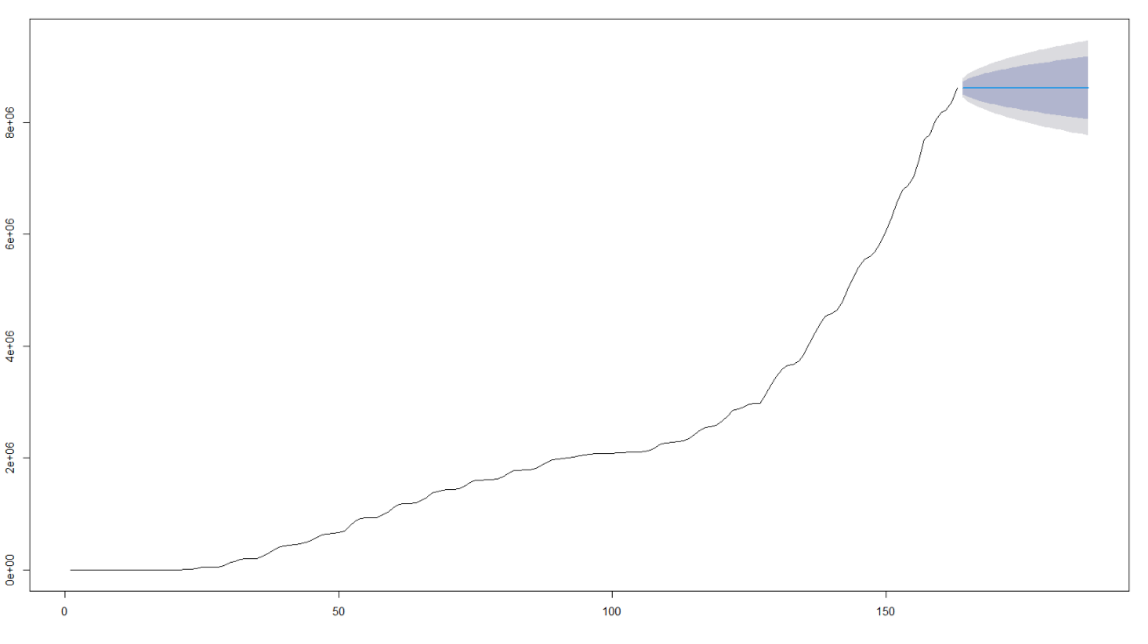
```
        Shapiro-Wilk normality test

data:  TimeSerialMSaicc.auto$residuals
W = 0.98575, p-value = 0.005308
```

12. Prognozowanie z wykorzystaniem metod naiwnych

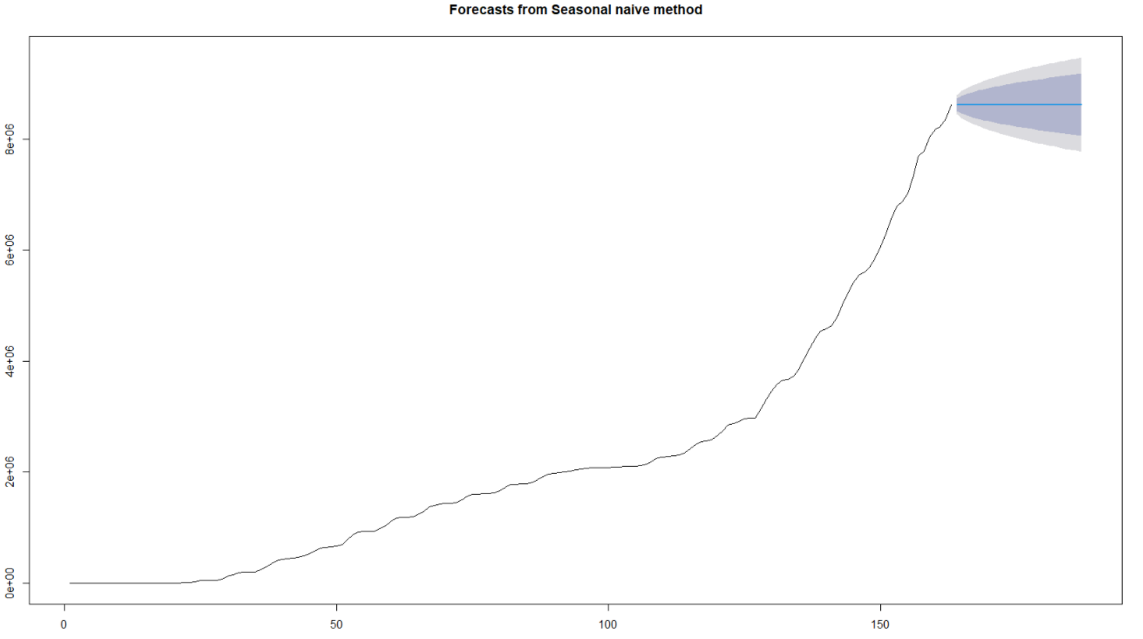
1) Prognoza oparta na średniej



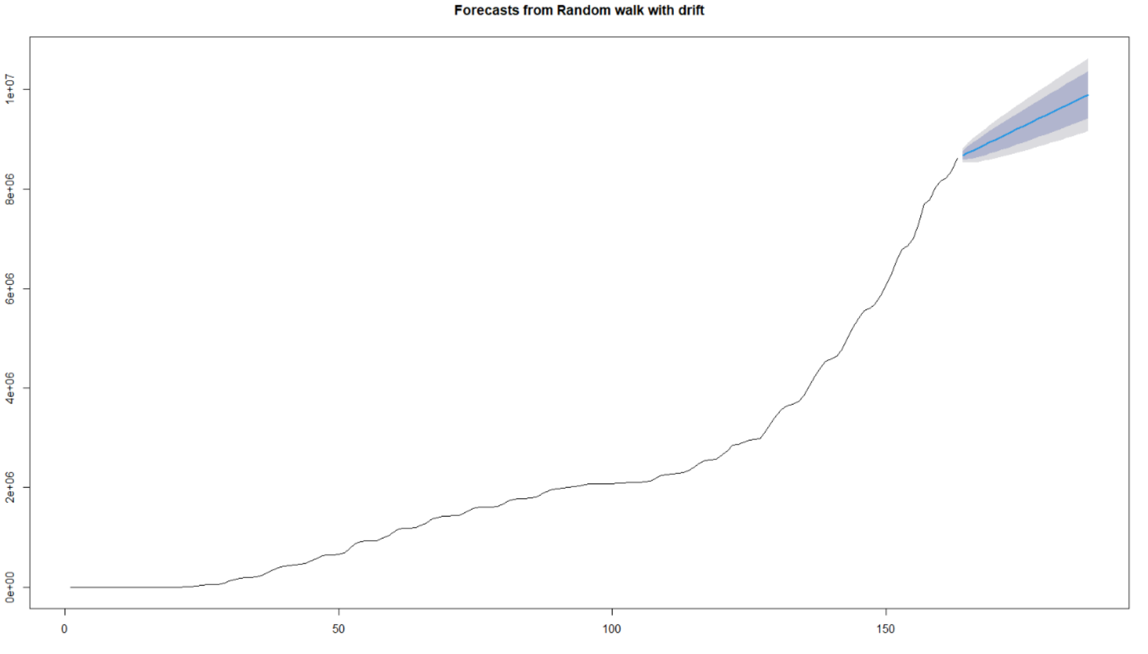
2) Prognozowanie naiwne z użyciem funkcji naive()



3) Prognozowanie z użyciem funkcji snaive()



4) Prognozowanie naiwne z dryfem



Najlepszą metodą w celu prognozy liczby całkowitych szczepień jest zastosowanie tutaj prognozowania naiwnego z dryfem z powodu trendu wielomianowego w szeregu. Nadal nie będzie ona jednak wystarczająco poprawna.