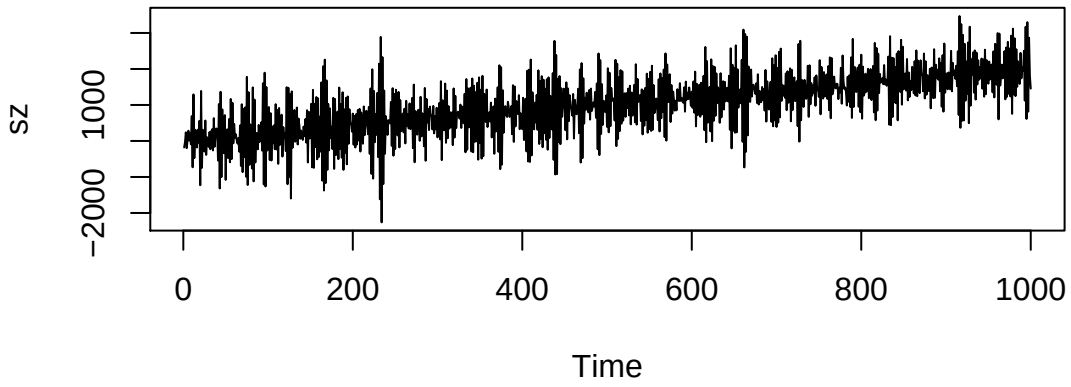
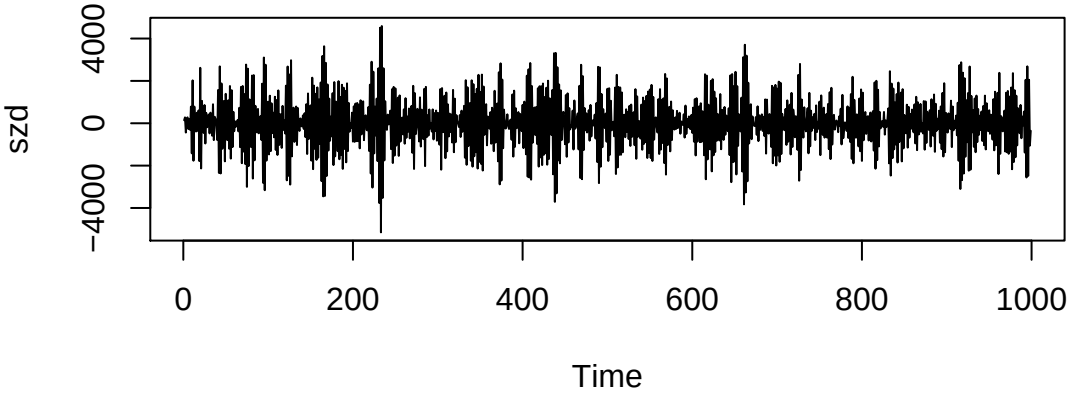
```
# zad.1

sz <- read.table("http://gamma.mini.pw.edu.pl/~szymanowskih/lab5/szereg.txt")
sz <- as.numeric(sz[,1])
head(sz)

## [1] -180.748  -37.494  240.327 -200.448    7.833  285.570

ts.plot(sz)
```




```
n <- length(sz)

# pierwszy sposob - roznicowanie:

szd <- diff(sz)   # zmienia sie rzad szeregu MA!!!
ts.plot(szd)
```




```
mod1 <- arima(szd,c(1,0,3),method="ML",include.mean=TRUE)

phi1 <- mod1$coef[1]
theta1 <- mod1$coef[2]+1
theta2 <- -mod1$coef[4]
a <- mod1$coef[5]
```

```r
# b - tak mozemy je odzyskac:

ut <- sz-(1:n)*a
b <- mean(ut)

par <- c(phi1,theta1,theta2,a,b)
names(par) <- c("phi1","theta1","theta2","a","b")
par
```

```
##    phi1  theta1  theta2       a       b
## -0.7400 -0.2884  0.5300  1.9930  0.8702
```

```r
# drugi sposob - dopasowanie modelu:

l <- lm(sz~c(1:n))
summary(l)
```

```
##
## Call:
## lm(formula = sz ~ c(1:n))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2728.0  -480.9     0.2   486.9  2424.6
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.3527    44.3859    0.03     0.98
## c(1:n)        1.9920     0.0768   25.93   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 701 on 998 degrees of freedom
## Multiple R-squared:  0.403,Adjusted R-squared:  0.402
## F-statistic:  672 on 1 and 998 DF,  p-value: <2e-16
```

```r
ut <- sz - l$coef[1]-l$coef[2]*(1:n)

mod2 <- arima(ut,c(1,0,2),method="ML",include.mean=FALSE)
# dlaczego bez sredniej? Bo byla juz uwzgledniona wczesniej!

par2 <- c(mod2$coef,l$coef[2:1])
names(par2) <- c("phi1","theta1","theta2","a","b")

par
```

```
##    phi1  theta1  theta2       a       b
## -0.7400 -0.2884  0.5300  1.9930  0.8702
```

```r
par2    # oprocz interceptu wyszlo podobnie :D
```

```
##    phi1  theta1  theta2       a       b
## -0.7403 -0.2888  0.5293  1.9920  1.3527
```

```r
# prawdziwa wartosc: c(-3/4,-1/3,1/2,5)  -> tak byly generowane te dane
# wyraz wolny ogolnie sie bardzo slabo estymuje...

# zad.2
```

```r
lac <- read.table("http://gamma.mini.pw.edu.pl/~szymanowskih/lab5/LACounty.txt",header=TRUE)
head(lac,2)
```

```
##   DATE  TEMP  PART CARDIO
## 1    1 72.38 72.72  97.85
## 2    2 67.19 49.60 104.64
```

```r
# a)

plot(lac$TEMP,lac$CARDIO)
# widac, ze w miare ukladaja sie na paraboli,
# wiec wprowadzmy nowa zmienna kwadrawowa:

t <- lac$DATE
y <- lac$CARDIO
t1 <- lac$TEMP - mean(lac$TEMP)
t2 <- (lac$TEMP - mean(lac$TEMP))^2
p <- lac$PART

# b)

mod1 <- lm(y~t+t1+t2+p)
summary(mod1)
```

```
##
## Call:
## lm(formula = y ~ t + t1 + t2 + p)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.076  -4.215  -0.488   3.744  29.245
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 81.59224    1.10215   74.03  < 2e-16 ***
## t           -0.02684    0.00194  -13.82  < 2e-16 ***
## t1          -0.47247    0.03162  -14.94  < 2e-16 ***
## t2           0.02259    0.00283    7.99  9.3e-15 ***
## p            0.25535    0.01886   13.54  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.39 on 503 degrees of freedom
## Multiple R-squared:  0.595,Adjusted R-squared:  0.592
## F-statistic:  185 on 4 and 503 DF,  p-value: <2e-16
```

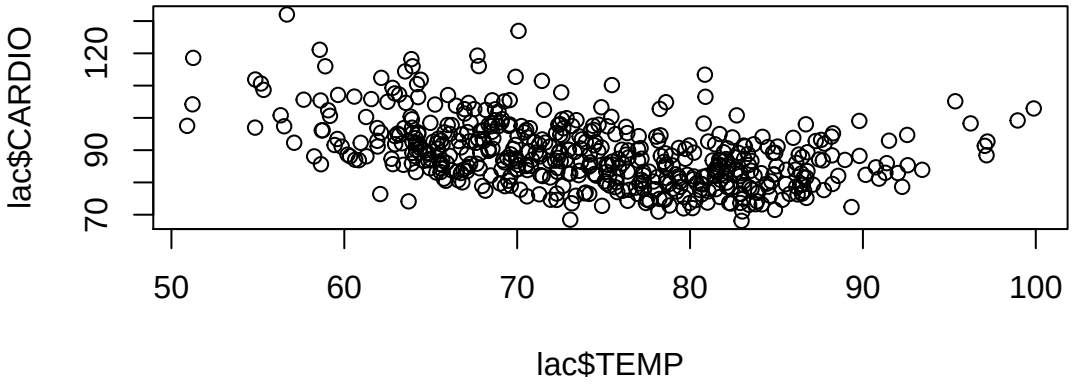
```r
# c)

r1 <- mod1$residuals
Box.test(r1,lag=20,type="Ljung")  # reszty nie sa bialym szumem
```

```
##
##  Box-Ljung test
##
## data:  r1
## X-squared = 270.4, df = 20, p-value < 2.2e-16
```

```r
# d)

modr1 <- ar(r1)
modr1$ar
```

```
## [1] 0.18939 0.34383 0.08343
```

```r
# e)
# model zmodyfikowany:

coeff <- c(1,-modr1$ar)

library("quantmod")
```

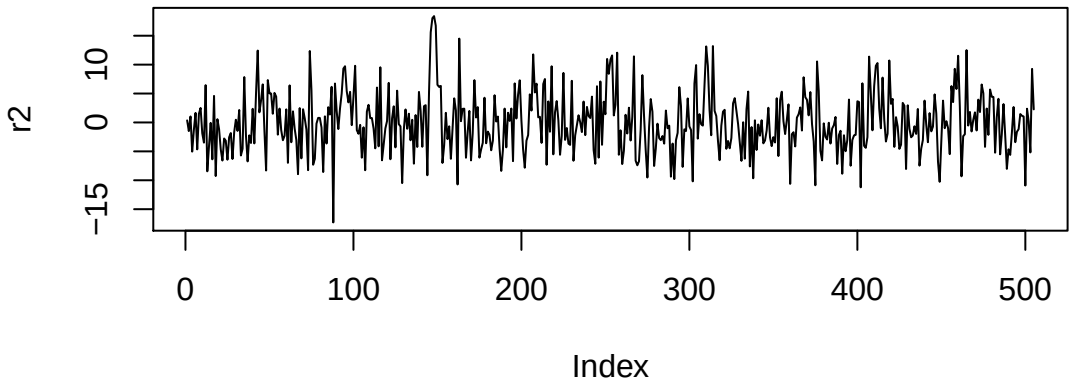



```r
for (name in c("t","y","t1","t2","p")) {
  v <- get(name)    # wartosc zmiennej, ktora ma taka nazwe, jak ten string
  assign(paste("N",name,sep=""),cbind(v,Lag(v,1),Lag(v,2),Lag(v,3))%*%coeff)
  # assingn("x",5) to to samo co x <- 5
}

mod2 <- lm(Ny~Nt+Nt1+Nt2+Np)
summary(mod2)
```

```
##
## Call:
## lm(formula = Ny ~ Nt + Nt1 + Nt2 + Np)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.293  -3.466  -0.466   2.973  18.391
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 32.07638    0.65026   49.33  < 2e-16 ***
## Nt          -0.02816    0.00419   -6.72  4.9e-11 ***
## Nt1         -0.19132    0.03939   -4.86  1.6e-06 ***
## Nt2          0.01720    0.00222    7.76  4.8e-14 ***
## Np           0.22817    0.02305    9.90  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 5.25 on 500 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.297,Adjusted R-squared:  0.292
## F-statistic: 52.9 on 4 and 500 DF,  p-value: <2e-16
```

```
r2 <- mod2$residuals
plot(r2,type="l")
```




```
Box.test(r2,lag=20,type="Ljung")
```

```
##
##  Box-Ljung test
##
## data:  r2
## X-squared = 44.79, df = 20, p-value = 0.001177
```

```
# poprawilo sie, ale dalej jest nie najlepiej

# co proponujemy w takiej sytuacji? jeszcze raz to samo!

# f)

modr2 <- ar(r2)
modr2$ar  # rzad dwa
```

```
## [1] 0.12095 0.08783
```

```
coeff <- c(1,-modr2$ar)

for (name in c("Nt","Ny","Nt1","Nt2","Np")) {
  v <- get(name)
  assign(paste("N",name,sep=""),cbind(v,Lag(v,1),Lag(v,2))%*%coeff)
}

mod3 <- lm(NNy~NNt+NNt1+NNt2+NNp)
summary(mod3)
```

```
##
## Call:
## lm(formula = NNy ~ NNt + NNt1 + NNt2 + NNp)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.715  -3.427  -0.373   3.123  17.008
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 25.95040    0.59561   43.57  < 2e-16 ***
## NNt         -0.02863    0.00524   -5.46  7.4e-08 ***
## NNt1        -0.10139    0.04106   -2.47    0.014 *
## NNt2         0.01604    0.00210    7.65  1.1e-13 ***
## NNp          0.19325    0.02387    8.10  4.4e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.17 on 498 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.248,Adjusted R-squared:  0.242
## F-statistic:   41 on 4 and 498 DF,  p-value: <2e-16
```

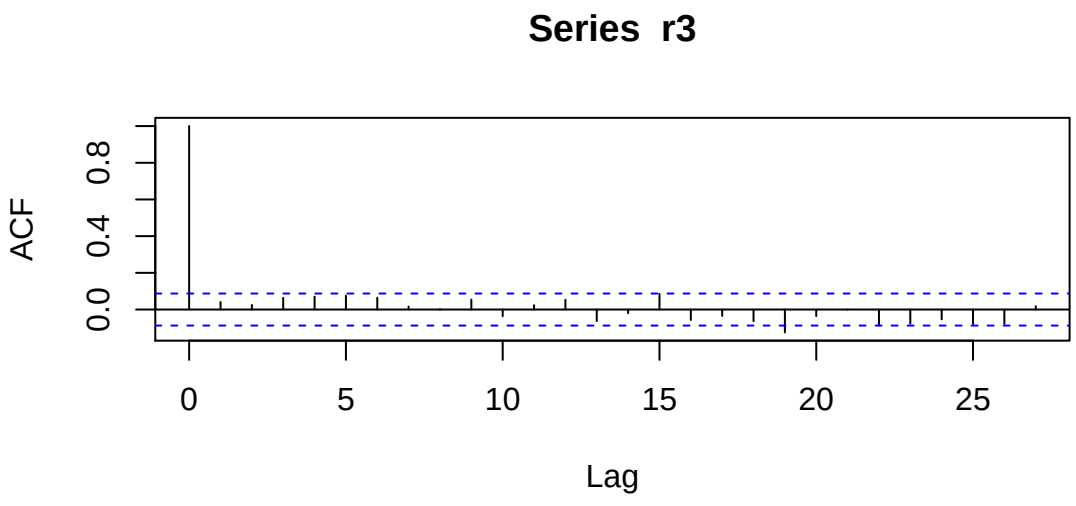
```
r3 <- mod3$residuals
Box.test(r3,lag=20,type="Ljung")  # no prawie :D
```

```
##
##  Box-Ljung test
##
## data:  r3
## X-squared = 34.25, df = 20, p-value = 0.0245
```

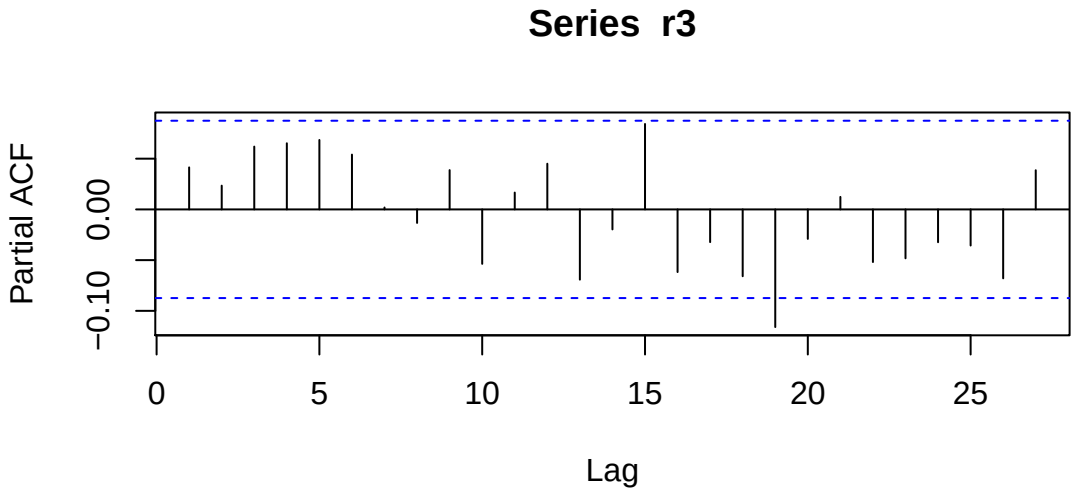
```
# z tych danych juz wiecej nie wycisniemy -> to sa dane rzeczywiste,
# dlatego tak opornie idzie

modr3 <- ar(r3)
modr3$ar  # nic nie zwraca, wiec nie dopasujemy juz dalej
```

```
## numeric(0)
```

```
acf(r3)  # acf nie sa takie zle w sumie, wystaje akurat ten 19,
```



```
        # wiec box test o niego zahacza
pacf(r3) # podobnie
```




```
# z kazda iteracja cos tam uzyskiwalismy, wiec metoda dziala :D
```