

# Final Project 274

*Katie Schmitzer*

*5/23/2019*

```
#install.packages("devtools")
#devtools::install_github("FinYang/tsdl")
#install.packages("qpcR")
library(qpcR)

## Loading required package: MASS
## Loading required package: minpack.lm
## Loading required package: rgl
## Warning: package 'rgl' was built under R version 3.5.2
## Loading required package: robustbase
## Warning: package 'robustbase' was built under R version 3.5.2
## Loading required package: Matrix
library(tsdl)
#install.packages("tseries")
library(tseries)

## Warning: package 'tseries' was built under R version 3.5.2
library(forecast)

## Warning: package 'forecast' was built under R version 3.5.2
#install.packages("TSA")
require(TSA)

## Loading required package: TSA
##
## Attaching package: 'TSA'
## The following objects are masked from 'package:stats':
##
##     acf, arima
## The following object is masked from 'package:utils':
##
##     tar
#install.packages("GeneCycle")

Get Initial Data

#tsdl
for(i in 200:600) cat("i =", i, attr(tsdl[[i]], "description"), "\n")
chemical <- ts(tsdl[[508]]) #Chemical process readings every two minutes
actualvals<-chemical[91:100]
```

```
chem <- chemical[0:90]
chem[30:40]
```

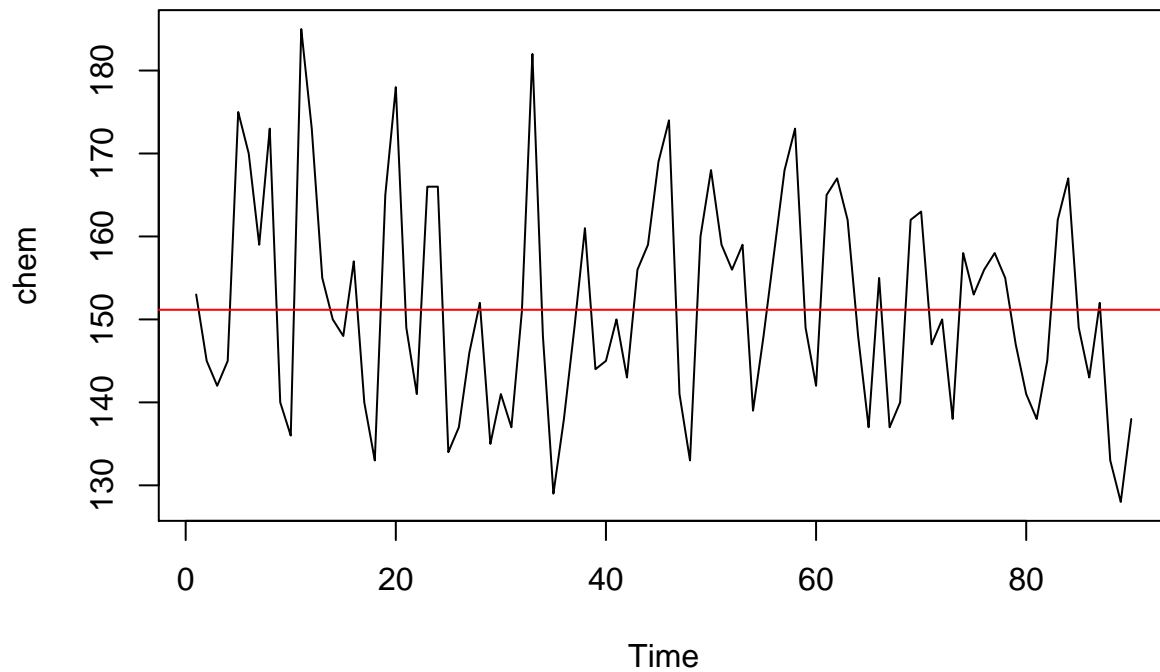
```
## [1] 141 137 151 182 148 129 138 149 161 144 145
```

```
length(chem)
```

```
## [1] 90
```

Examine Raw Data

```
ts.plot(chem) + abline(h=151.16, col = "red", main = "Chemical Process Readings Every Two Minutes in Deg
```



```
## integer(0)
```

```
mean(chem) # mean is 151.9
```

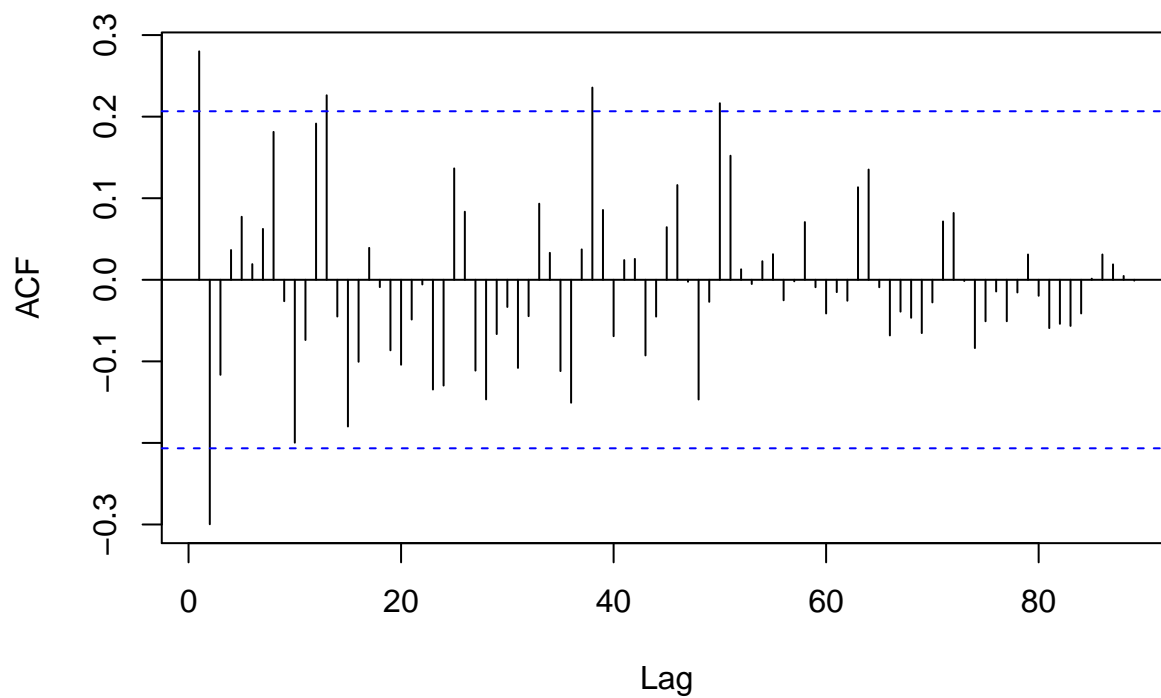
```
## [1] 151.9
```

```
var(chem) # variance is 167.6865
```

```
## [1] 167.6865
```

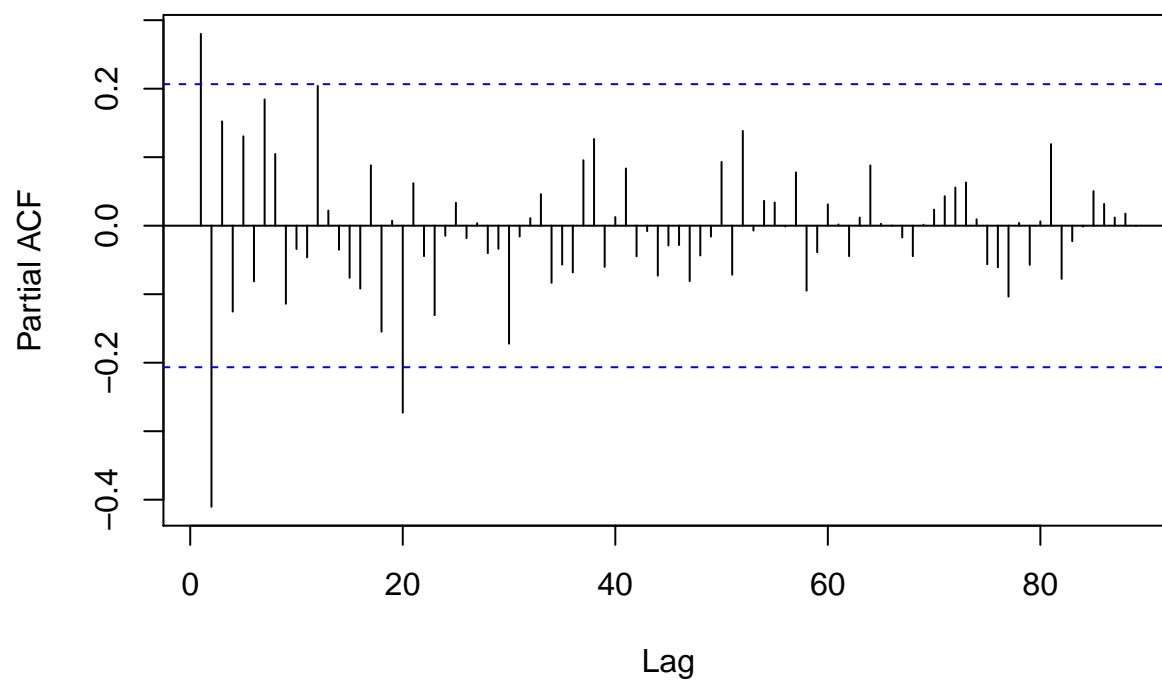
```
acf(chem, main = "ACF for Raw Data", lag.max = 90) # acf has non-zero values at lags 1 and 2 and ones
```

### ACF for Raw Data



```
pacf(chem, main = "PACF for Raw Data", lag.max = 90) # pacf has non-zero values at lags 1 and 2 and 20
```

### PACF for Raw Data



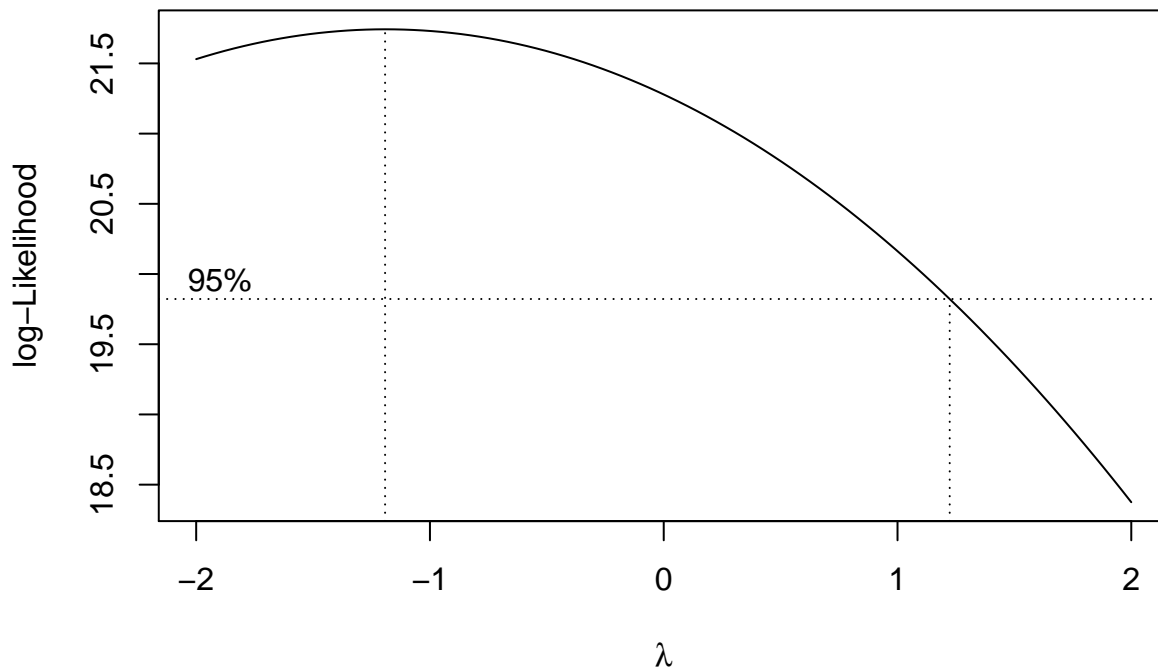
```
adf.test(chem)
```

```
##
```

```
## Augmented Dickey-Fuller Test
##
## data: chem
## Dickey-Fuller = -3.6889, Lag order = 4, p-value = 0.02996
## alternative hypothesis: stationary
```

### Transform the Data

```
bcTransform <- boxcox(chem ~ as.numeric(1:length(chem))) #graph shows that either a log-transformed data
```

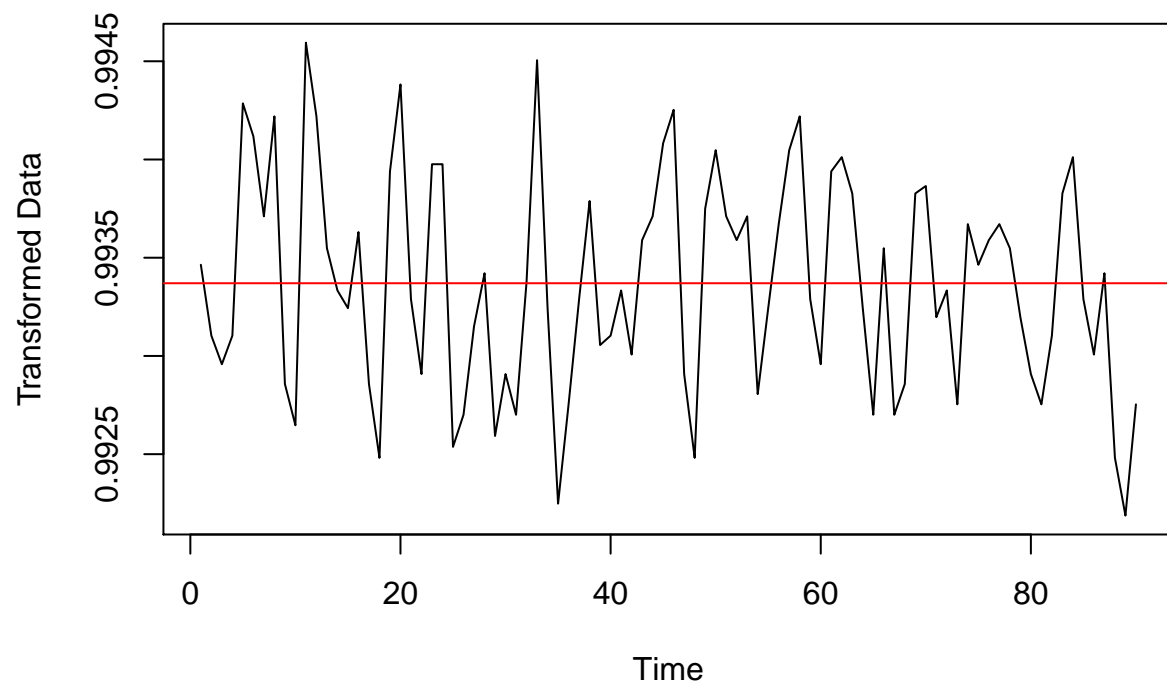


```
lambda <- bcTransform$x[which(bcTransform$y == max(bcTransform$y))]
lambda # ideal lambda = -1.191919
```

```
## [1] -1.191919
```

### Try -1 For Lambda

```
lambda = -1 # chose -1 for simplicity and parsimony
chemneg1 <- (1/lambda)*(chem^lambda - 1)
ts.plot(chemneg1, gpars = list(ylab = "Transformed Data")) + abline(h= 0.9933702, col = "red")
```



```
## integer(0)
```

```
mean(chemneg1) # mean is 0.9933702
```

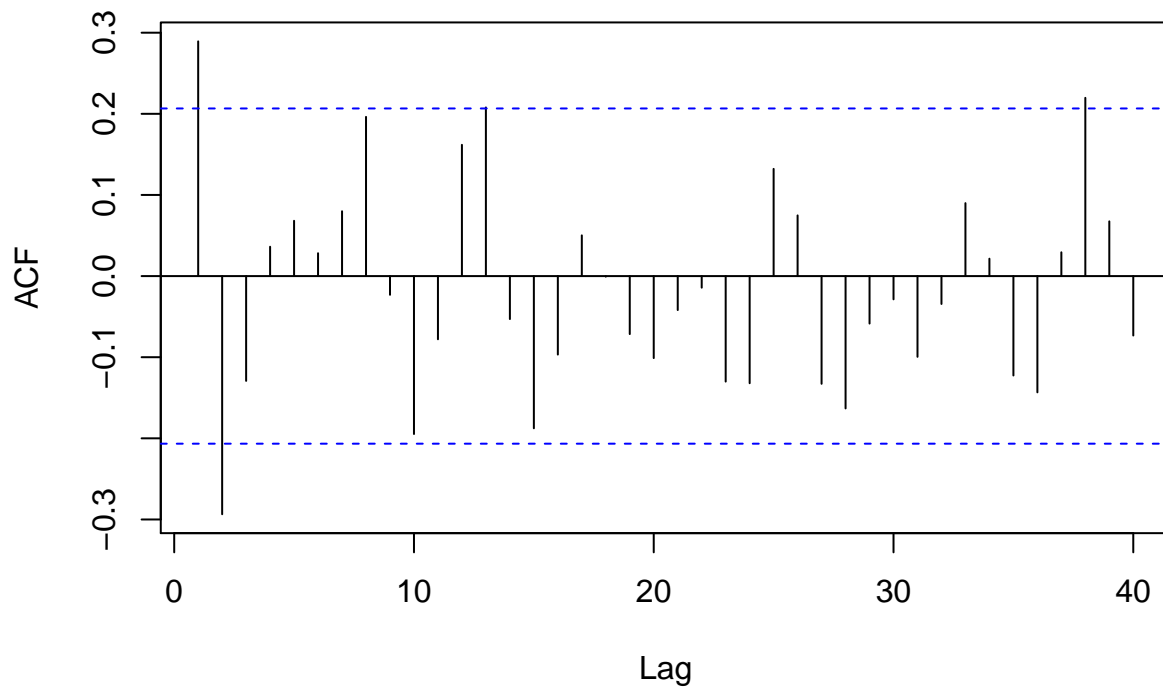
```
## [1] 0.9933702
```

```
var(chemneg1) # variance is 3.073924e-07
```

```
## [1] 3.073924e-07
```

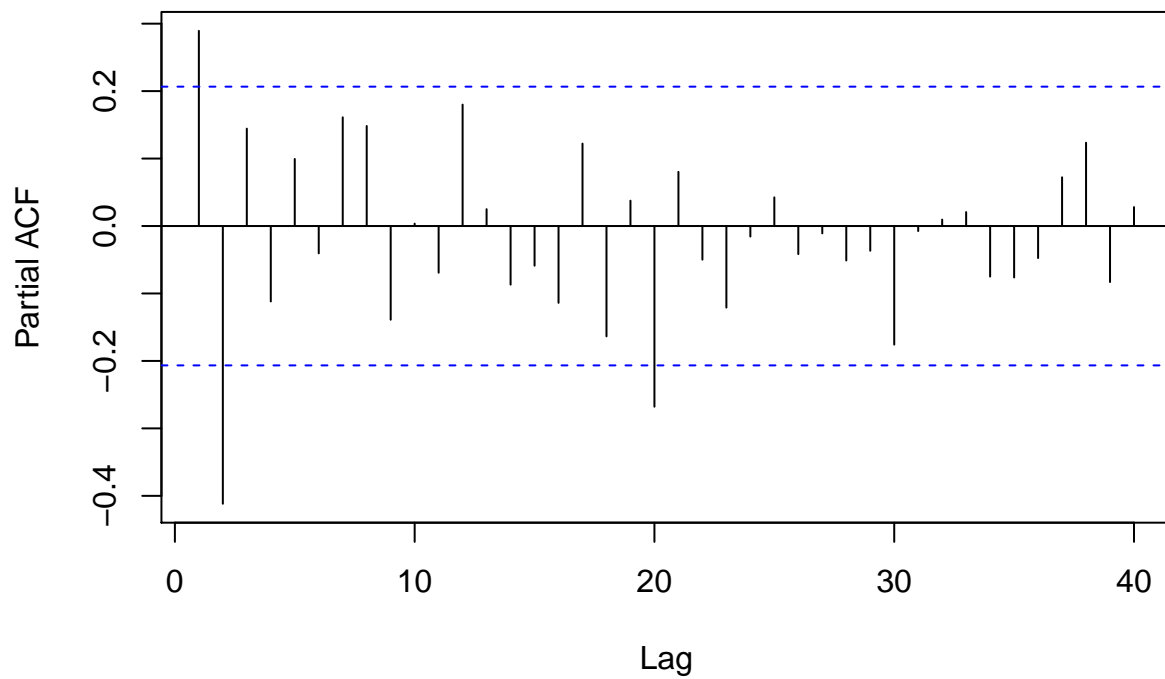
```
acf(chemneg1, main = "ACF for Transformed lambda = -1 Data", lag.max = 40) # acf has non-zero values a
```

### ACF for Transformed $\lambda = -1$ Data



```
pacf(chemneg1, main = "PACF for Transformed  $\lambda = -1$  Data", lag.max = 40) # may be an MA(2) model, ...
```

### PACF for Transformed $\lambda = -1$ Data



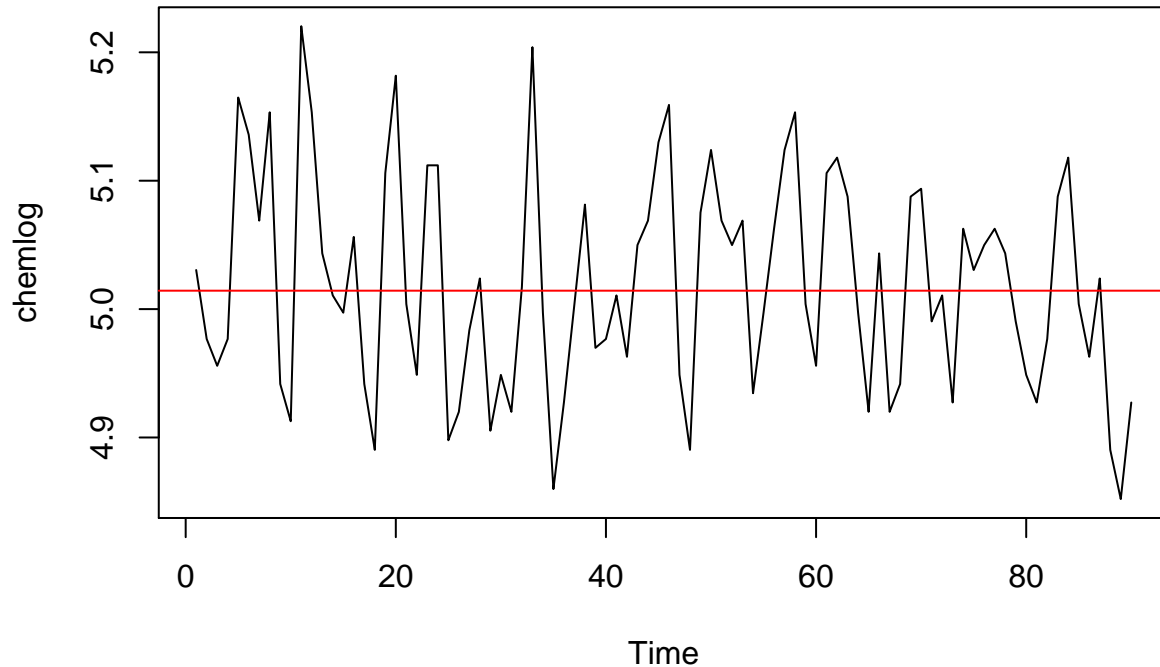
```
adf.test(chemneg1) # data is stationary
```

```
##
```

```
## Augmented Dickey-Fuller Test
##
## data: chemneg1
## Dickey-Fuller = -3.7036, Lag order = 4, p-value = 0.02867
## alternative hypothesis: stationary
```

Try Log-Data

```
chemlog <- log(chem)
ts.plot(chemlog) + abline(h= 5.014343, col = "red")
```

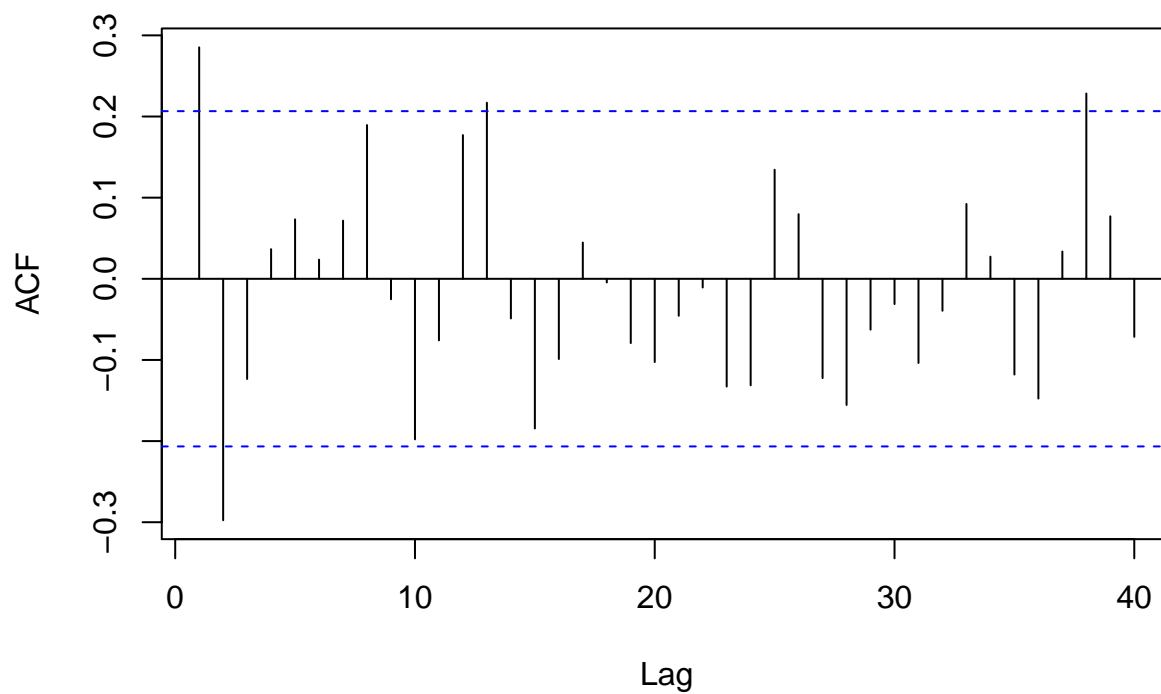


```
## integer(0)
mean(chemlog) # mean is 5.019678

## [1] 5.019678
var(chemlog) # variance is 0.007126066

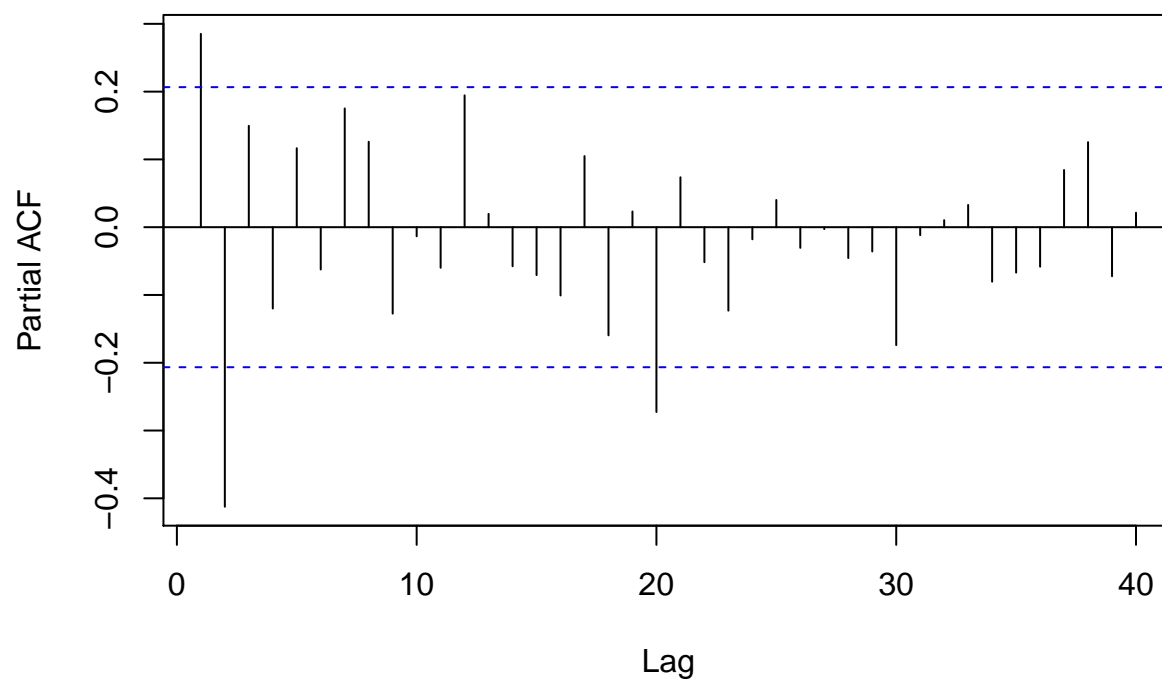
## [1] 0.007126066
acf(chemlog, main = "ACF for Log Data", lag.max = 40) # acf has non-zero values at lags 1 and 2
```

### ACF for Log Data



```
pacf(chemlog, main = "PACF for Log Data", lag.max = 40) # may be an MA(2) model
```

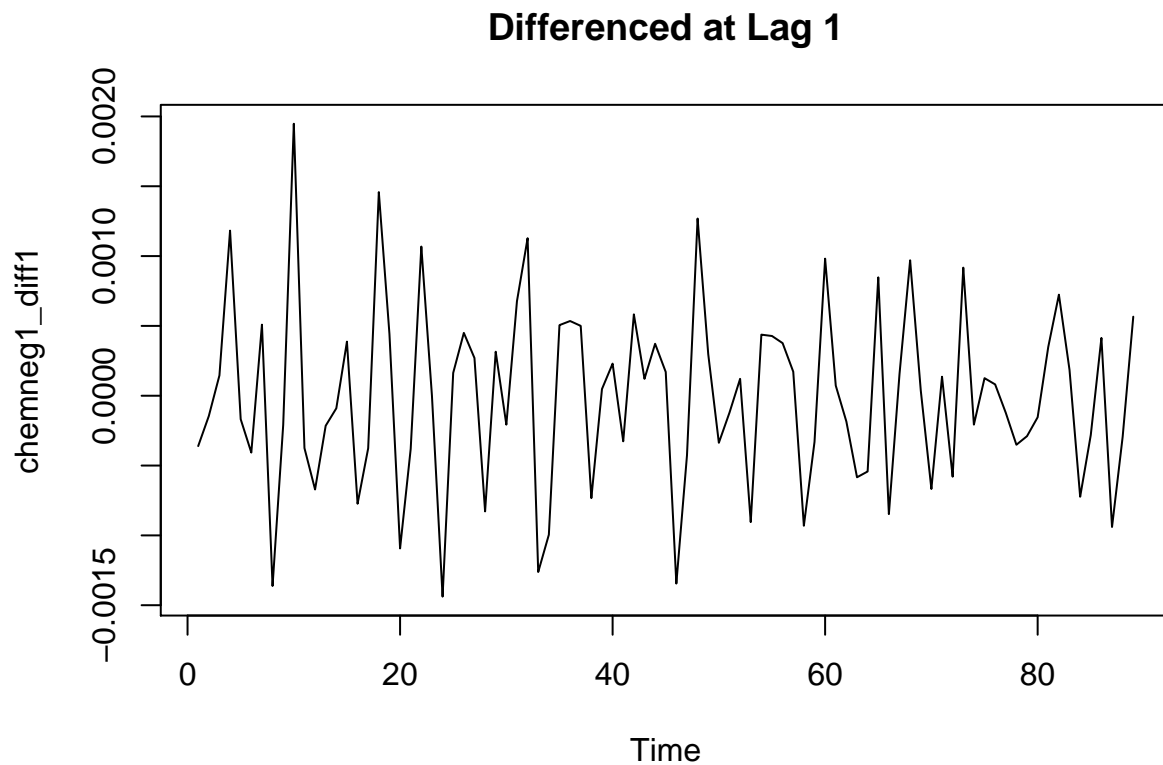
### PACF for Log Data



De-Trend the Data by differencing at lag 1



```
chemneg1_diff1 <-diff(chemneg1,1)
ts.plot(chemneg1_diff1, main = "Differenced at Lag 1")
```



```
mean(chemneg1_diff1) # mean is nearly 0
```

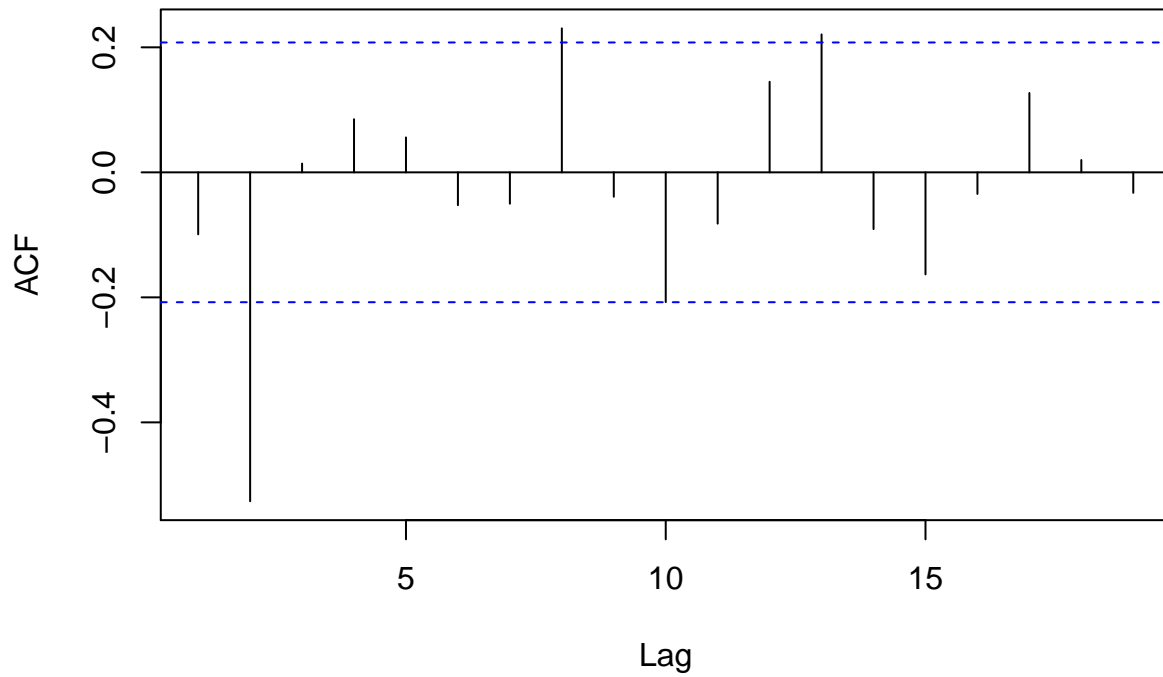
```
## [1] -7.982349e-06
```

```
var(chemneg1_diff1) # - variance increased
```

```
## [1] 4.373868e-07
```

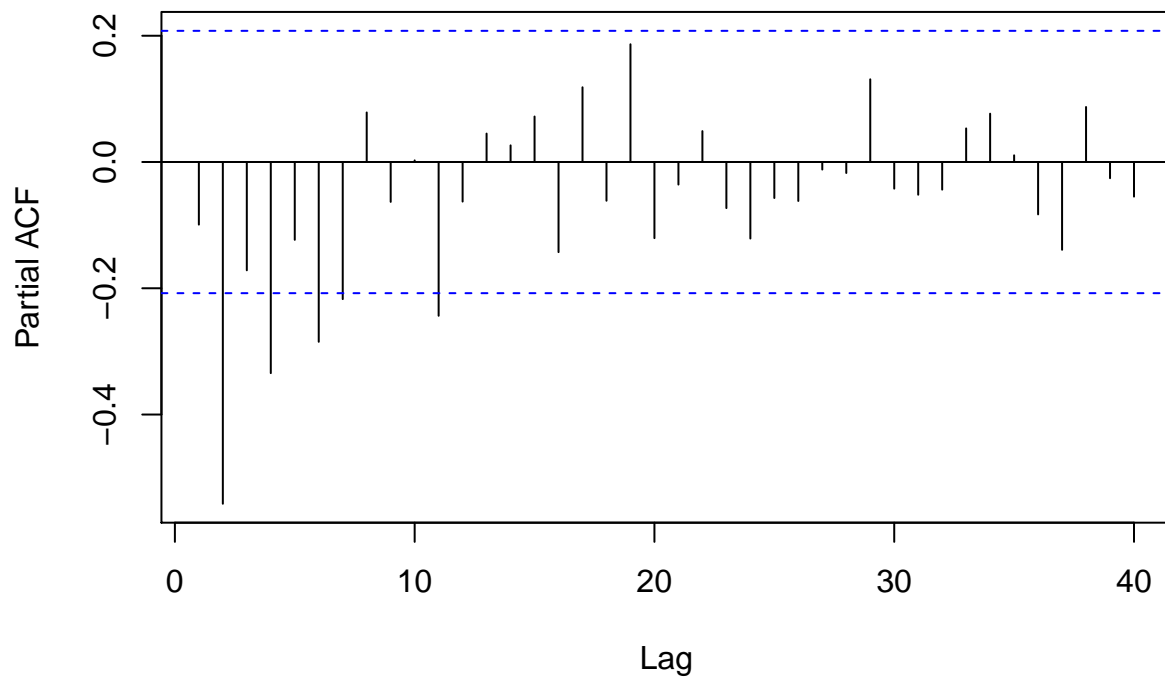
```
acf(chemneg1_diff1, main = "ACF for Differenced Data") # acf has more non zero lags
```

**ACF for Differenced Data**



```
pacf(chemneg1_diff1, main = "PACF for Differenced Data", lag.max = 40) # pacf has many non zero values
```

**PACF for Differenced Data**



Use Innovation Algorithm

```

innovations.algorithm <- function(n.max,acvs)
{
  thetas <- matrix(rep(0,n.max^2),nrow=n.max)
  vs <- rep(0,n.max+1)
  vs[1] <- acvs[1]
  for(n in 1:n.max)
  {
    thetas[n,n] <- acvs[n+1]/vs[1]
    if(n>1)
    {
      for(k in 1:(n-1))
      {
        inner.sum <- acvs[n-k+1]
        for(j in 0:(k-1))
        {
          inner.sum <- inner.sum - thetas[k,k-j]*thetas[n,n-j]*vs[j+1]
        }
        thetas[n,n-k] <- inner.sum/vs[k+1]
      }
    }
    vs[n+1] <- acvs[1]
    for(j in 0:(n-1))
    {
      vs[n+1] <- vs[n+1] - thetas[n,n-j]^2*vs[j+1]
    }
  }
  structure(list(thetas=thetas,vs=vs))
}

get.sigma.2.n.hs <- function(n,h.max,ts.var,ia.stuff)
{
  thetas <- ia.stuff$thetas
  vs <- ia.stuff$vs
  sigma.2.n.hs <- rep(ts.var,h.max)
  for(h in 1:h.max)
  {
    sigma.2.n.hs[h] <- sigma.2.n.hs[h] - sum((thetas[n+h-1,(n+h-1):h])^2*vs[1:n])
  }
  sigma.2.n.hs
}

acvf = acf(chemneg1, plot=FALSE, lag.max = length(chemneg1))$acf[,1,1] * var(chemneg1)
m = length(acvf)

spec.arma <- function(ar=0,ma=0,var.noise=1,n.freq=500, ...)
{
  # check causality
  ar.poly <- c(1, -ar)
  z.ar <- polyroot(ar.poly)
  if(any(abs(z.ar) <= 1)) cat("WARNING: Model Not Causal", "\n")
  # check invertibility

```

```

ma.poly <- c(1, ma)
z.ma <- polyroot(ma.poly)
if(any(abs(z.ma) <= 1)) cat("WARNING: Model Not Invertible", "\n")
if(any(abs(z.ma) <= 1) || any(abs(z.ar) <= 1) ) stop("Try Again")
#
ar.order <- length(ar)
ma.order <- length(ma)
# check (near) parameter redundancy [i.e. are any roots (approximately) equal]
for (i in 1:ar.order) {
  if ( (ar == 0 & ar.order == 1) || (ma == 0 & ma.order == 1) ) break
  if(any(abs(z.ar[i]-z.ma[1:ma.order]) < 1e-03)) {cat("WARNING: Parameter Redundancy", "\n"); break}
}
#
freq <- seq.int(0, 0.5, length.out = n.freq)
cs.ar <- outer(freq, 1:ar.order, function(x, y) cos(2 *
  pi * x * y)) %**% ar
sn.ar <- outer(freq, 1:ar.order, function(x, y) sin(2 *
  pi * x * y)) %**% ar
cs.ma <- outer(freq, 1:ma.order, function(x, y) cos(2 *
  pi * x * y)) %**% -ma
sn.ma <- outer(freq, 1:ma.order, function(x, y) sin(2 *
  pi * x * y)) %**% -ma
spec <- var.noise*((1 - cs.ma)^2 + sn.ma^2)/((1 - cs.ar)^2 + sn.ar^2)
spg.out <- list(freq=freq, spec=spec)
class(spg.out) <- "spec"
plot(spg.out, ci=0, main="", ...)
return(invisible(spg.out))
}
#chemneg1.ia = innovations.algorithm(m+1, acvf)
#chemneg1.ia$thetas[3,1:3]

```

## Fit to an AR Model

```

fit.ar.yw <- ar(chemneg1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
fit.ar.yw # suggests AR(2) with the coefficients : 0.4085 -0.4117

```

```

##
## Call:
## ar(x = chemneg1, aic = TRUE, order.max = NULL, method = c("yule-walker"))
##
## Coefficients:
##          1          2
## 0.4085 -0.4117
##
## Order selected 2  sigma^2 estimated as 2.393e-07
## 0.4085+1.96*2.393e-07
##
## [1] 0.4085005
## 0.4085-1.96*2.393e-07
##
## [1] 0.4084995
## -0.4117+1.96*2.393e-07

```

```
## [1] -0.4116995
-0.4117-1.96*2.393e-07

## [1] -0.4117005

fit.ar.mle <- ar(chemneg1, aic = TRUE, order.max = NULL, method = c("mle"))
fit.ar.mle # suggests AR(2) with the coefficients : 0.4041 -0.4211

##
## Call:
## ar(x = chemneg1, aic = TRUE, order.max = NULL, method = c("mle"))
##
## Coefficients:
##      1      2
## 0.4041 -0.4211
##
## Order selected 2  sigma^2 estimated as 2.296e-07
```

### Fitting to ARMA Model

```
aiccs <- matrix(NA, nr = 9, nc = 9)
dimnames(aiccs) = list(p=0:8, q=0:8)
for(p in 0:8)
{
  for(q in 0:8)
  {
    aiccs[p+1,q+1] = AICc(arima(chemneg1, order = c(p,0,q), method="ML")) }
}
```

```
## Warning in log(s2): NaNs produced

## Warning in stats::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

## Warning in log(s2): NaNs produced

## Warning in stats::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

## Warning in log(s2): NaNs produced

## Warning in log(s2): NaNs produced

## Warning in log(s2): NaNs produced

## Warning in stats::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1

## Warning in stats::arima(x = x, order = order, seasonal = seasonal, xreg =
## xreg, : possible convergence problem: optim gave code = 1
```

```
aiccs # possible models include ARMA(1,3), ARMA(0,2), ARMA(0,3), ARMA(2,1), ARMA(3,5), ARMA(1,3), ARMA(
```

```
##      q
## p      0      1      2      3      4      5      6
## 0 -1091.114 -1111.300 -1112.035 -1112.928 -1111.502 -1110.677 -1114.168
## 1 -1096.920 -1110.635 -1111.463 -1115.872 -1110.604 -1111.473 -1112.403
## 2 -1111.662 -1112.617 -1110.794 -1109.527 -1111.968 -1109.961 -1110.267
```

```

## 3 -1110.938 -1110.673 -1108.501 -1106.158 -1110.900 -1108.679 -1109.558
## 4 -1109.924 -1108.704 -1114.401 -1103.874 -1111.412 -1108.732 -1106.981
## 5 -1108.542 -1106.620 -1104.854 -1110.747 -1108.420 -1106.410 -1104.323
## 6 -1106.385 -1104.324 -1102.407 -1107.595 -1106.322 -1106.273 -1105.464
## 7 -1107.048 -1105.889 -1109.099 -1106.504 -1105.564 -1106.691 -1105.216
## 8 -1108.026 -1107.419 -1104.851 -1104.415 -1108.007 -1105.738 -1103.012
## q
## p 7 8
## 0 -1112.319 -1110.741
## 1 -1109.945 -1109.020
## 2 -1108.156 -1112.044
## 3 -1106.976 -1109.575
## 4 -1107.129 -1106.840
## 5 -1106.207 -1104.808
## 6 -1102.847 -1103.719
## 7 -1102.662 -1100.834
## 8 -1100.247 -1101.570
min(aiccs) # ARMA(1,3)

## [1] -1115.872
auto.arima(chemneg1, stepwise = F, approximation = F, lambda = -1, ic = "aicc")

## Series: chemneg1
## ARIMA(1,0,3) with non-zero mean
## Box Cox transformation: lambda= -1
##
## Coefficients:
##      ar1      ma1      ma2      ma3      mean
##      -0.6924  1.2636  0.0448 -0.4117 -0.0067
## s.e.    0.1286  0.1393  0.1889  0.1039  0.0001
##
## sigma^2 estimated as 2.235e-07: log likelihood=563.1
## AIC=-1114.2 AICc=-1113.19 BIC=-1099.2

Trying ARMA(2,0)

arma20 <- arima(chemneg1, order = c(2, 0, 0), method = c("ML"))
arma20

##
## Call:
## arima(x = chemneg1, order = c(2, 0, 0), method = c("ML"))
##
## Coefficients:
##      ar1      ar2 intercept
##      0.4041 -0.4211  0.9934
## s.e.  0.0956  0.0972  0.0001
##
## sigma^2 estimated as 2.296e-07: log likelihood = 559.97, aic = -1113.94
fit.ar.mle

##
## Call:
## ar(x = chemneg1, aic = TRUE, order.max = NULL, method = c("mle"))

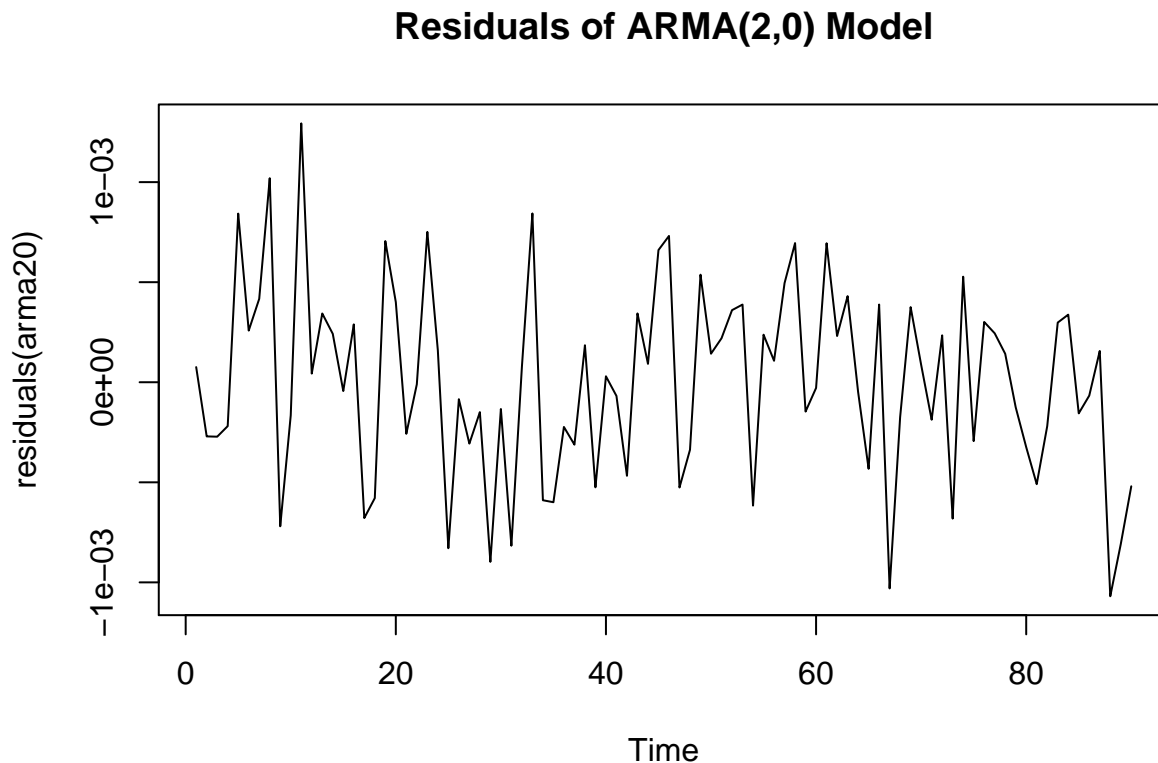
```

```
##
## Coefficients:
##      1      2
## 0.4041 -0.4211
##
## Order selected 2  sigma^2 estimated as 2.296e-07
```

```
mean(residuals(arma20))
```

```
## [1] -9.508091e-07
```

```
plot(residuals(arma20), main = "Residuals of ARMA(2,0) Model")
```



```
Box.test(residuals(arma20), lag = sqrt(90), type = "Ljung-Box", fitdf = 2)
```

```
##
## Box-Ljung test
##
## data: residuals(arma20)
## X-squared = 8.8008, df = 7.4868, p-value = 0.3112
```

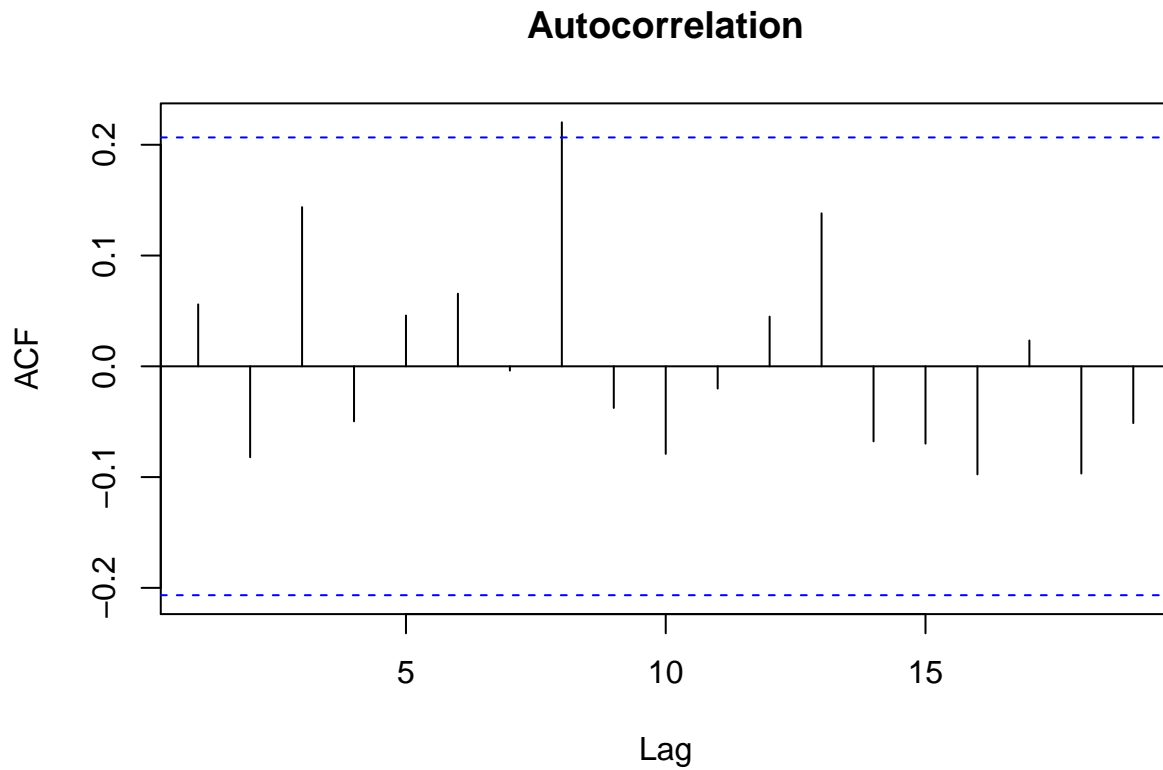
```
Box.test(residuals(arma20), type = "Box-Pierce", lag = sqrt(90), fitdf = 2)
```

```
##
## Box-Pierce test
##
## data: residuals(arma20)
## X-squared = 8.0391, df = 7.4868, p-value = 0.3777
```

```
Box.test(residuals(arma20)^2, type = "Ljung", lag = sqrt(90), fitdf = 0)
```

```
##
## Box-Ljung test
```

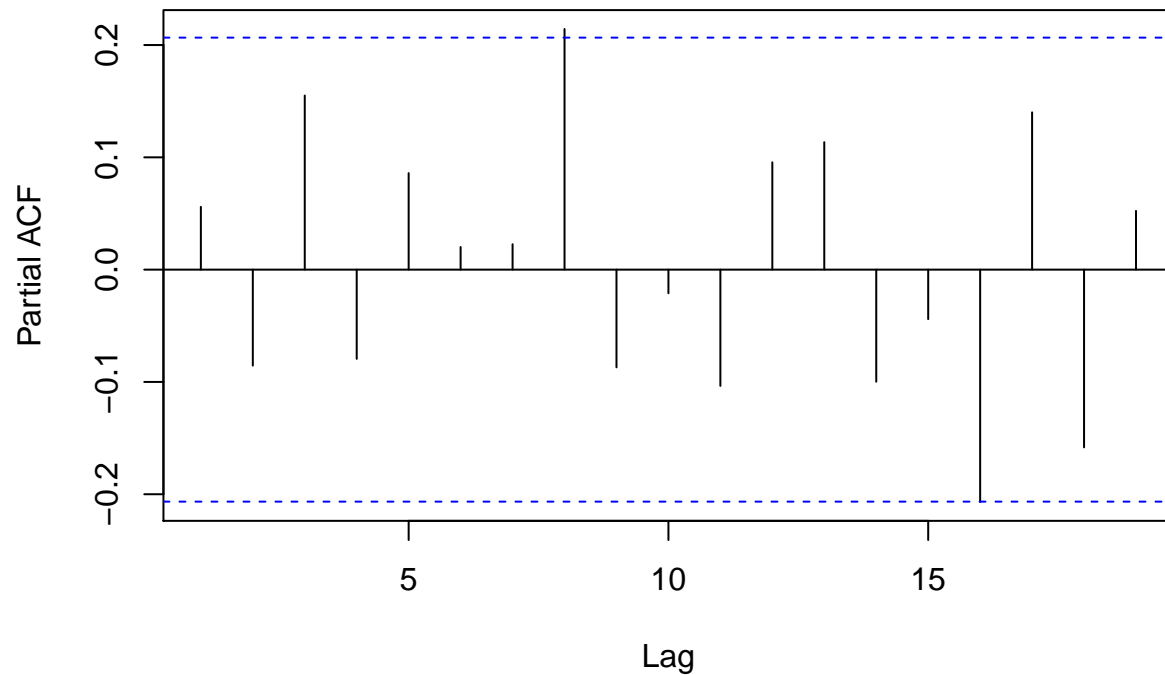
```
##  
## data: residuals(arma20)^2  
## X-squared = 14.47, df = 9.4868, p-value = 0.1278  
acf(residuals(arma20),main = "Autocorrelation")
```



```
pacf(residuals(arma20),main = "Partial Autocorrelation")
```



## Partial Autocorrelation

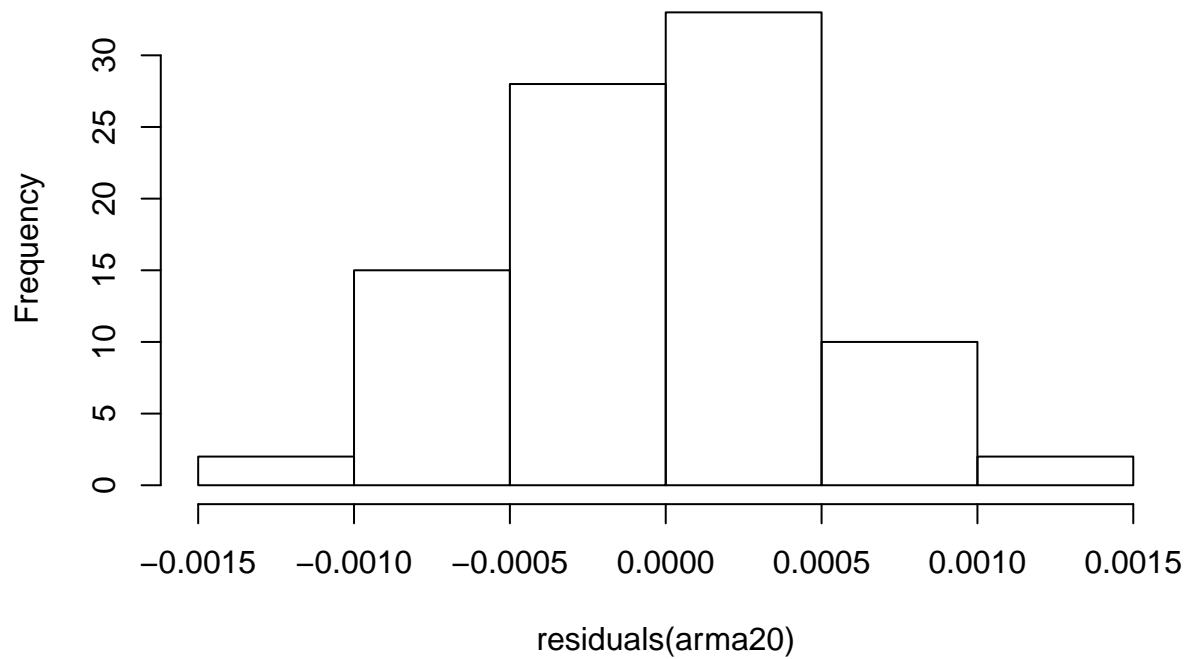


```
residuals(arma20)
```

```
## Time Series:
## Start = 1
## End = 90
## Frequency = 1
## [1] 7.569544e-05 -2.705663e-04 -2.720482e-04 -2.193238e-04 8.427030e-04
## [6] 2.582171e-04 4.170598e-04 1.019703e-03 -7.198666e-04 -1.649985e-04
## [11] 1.293653e-03 4.320287e-05 3.436088e-04 2.419306e-04 -4.393718e-05
## [16] 2.892361e-04 -6.786609e-04 -5.789291e-04 7.054777e-04 4.005060e-04
## [21] -2.577239e-04 -1.023510e-05 7.512829e-04 1.592830e-04 -8.295055e-04
## [26] -8.472649e-05 -3.066338e-04 -1.492834e-04 -8.975182e-04 -1.336545e-04
## [31] -8.169911e-04 7.618551e-05 8.435048e-04 -5.896018e-04 -5.996519e-04
## [36] -2.234807e-04 -3.119135e-04 1.850280e-04 -5.251020e-04 2.977492e-05
## [41] -6.848841e-05 -4.675611e-04 3.438797e-04 9.189784e-05 6.605780e-04
## [46] 7.311529e-04 -5.259131e-04 -3.373366e-04 5.374172e-04 1.426404e-04
## [51] 2.197582e-04 3.603029e-04 3.882411e-04 -6.165068e-04 2.376176e-04
## [56] 1.073740e-04 4.955240e-04 6.954030e-04 -1.465313e-04 -2.866738e-05
## [61] 6.945894e-04 2.311427e-04 4.303890e-04 -4.827475e-05 -4.326456e-04
## [66] 3.883512e-04 -1.030325e-03 -1.743901e-04 3.754506e-04 8.718662e-05
## [71] -1.873731e-04 2.344809e-04 -6.814167e-04 5.274139e-04 -2.942327e-04
## [76] 3.013241e-04 2.445700e-04 1.422111e-04 -1.252219e-04 -3.243967e-04
## [81] -5.094509e-04 -2.192249e-04 2.981883e-04 3.378567e-04 -1.554449e-04
## [86] -6.687833e-05 1.563466e-04 -1.069420e-03 -8.089412e-04 -5.199186e-04
```

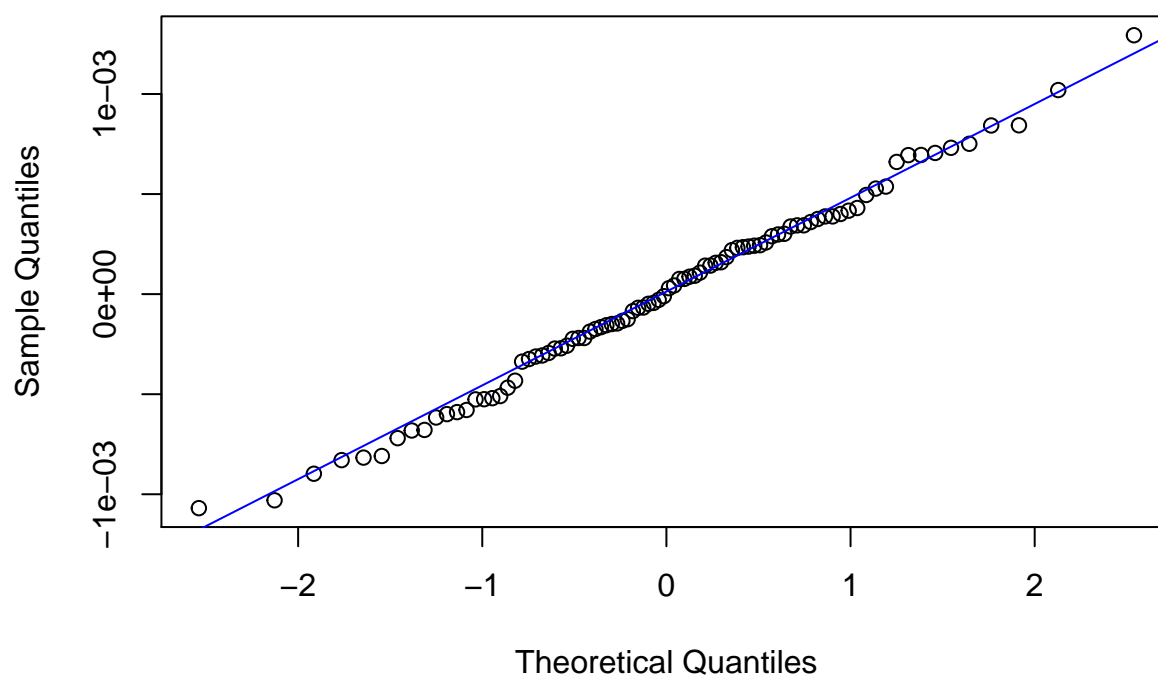
```
hist(residuals(arma20),main = "Histogram")
```

### Histogram



```
qqnorm(residuals(arma20))  
qqline(residuals(arma20),col ="blue")
```

### Normal Q-Q Plot



```
shapiro.test(residuals(arma20))
```

```
##
```

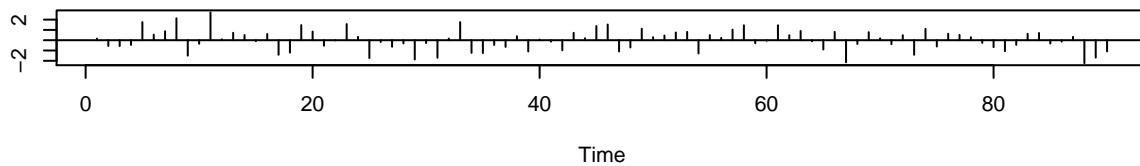
```
## Shapiro-Wilk normality test
##
## data: residuals(arma20)
## W = 0.99335, p-value = 0.9352
```

```
fit.ar <- ar(residuals(arma20),method="yw")
fit.ar
```

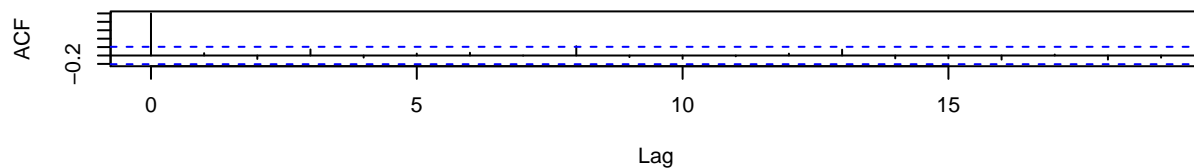
```
##
## Call:
## ar(x = residuals(arma20), method = "yw")
##
##
## Order selected 0 sigma^2 estimated as 2.322e-07
```

```
tsdiag(arima(chemneg1, order=c(2,0,0)))
```

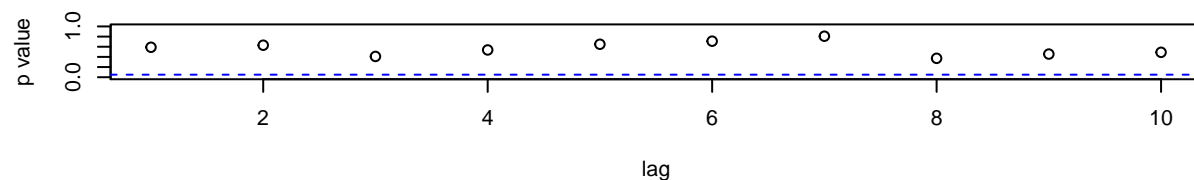
**Standardized Residuals**



**ACF of Residuals**



**p values for Ljung-Box statistic**



### Trying ARMA(1,3)

```
arma13 <- arima(chemneg1, order = c(1, 0, 3), method = c("ML"))
arma13
```

```
##
## Call:
## arima(x = chemneg1, order = c(1, 0, 3), method = c("ML"))
##
## Coefficients:
##      ar1      ma1      ma2      ma3 intercept
##    -0.6922  1.2634  0.0446 -0.4117    0.9934
## s.e.   0.1286  0.1393  0.1889  0.1039    0.0001
```

```
##
## sigma^2 estimated as 2.056e-07: log likelihood = 564.29, aic = -1118.59
-0.6922+1.96*0.1286 #ar1

## [1] -0.440144
-0.6922-1.96*0.1286

## [1] -0.944256
1.2634+1.96*0.1393 #ma1

## [1] 1.536428
1.2634-1.96*0.1393

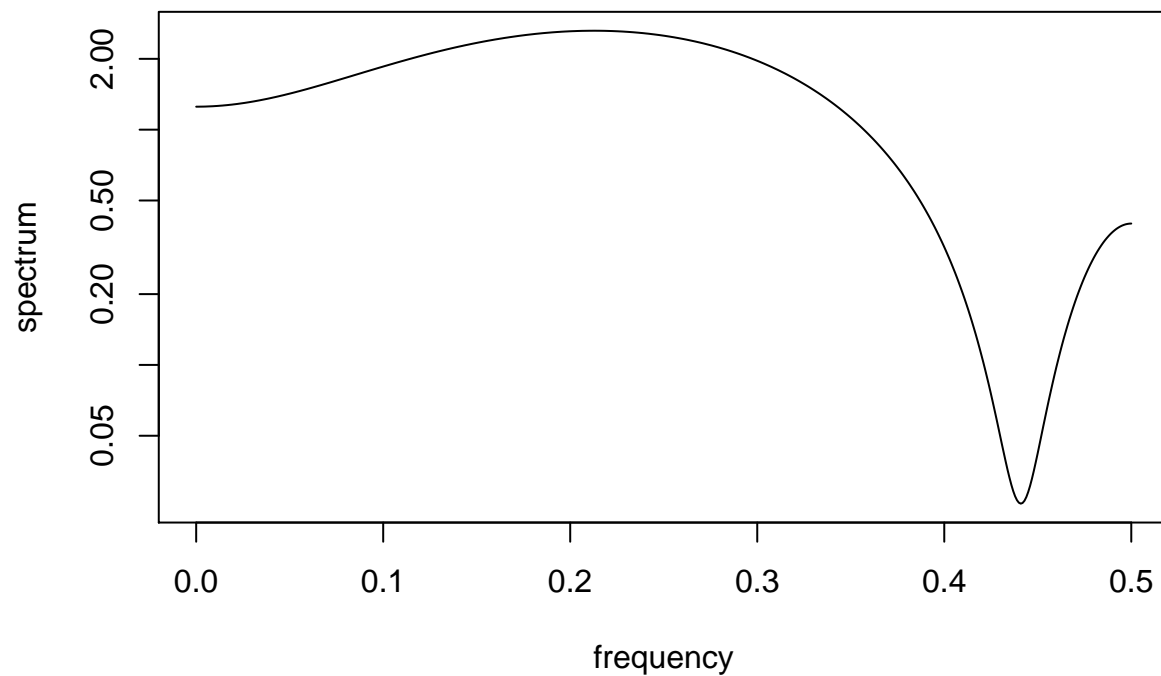
## [1] 0.990372
0.0446+1.96*0.1889 #ma2 contains 0

## [1] 0.414844
0.0446-1.96*0.1889

## [1] -0.325644
-0.4117+1.96*0.1039 #ma3

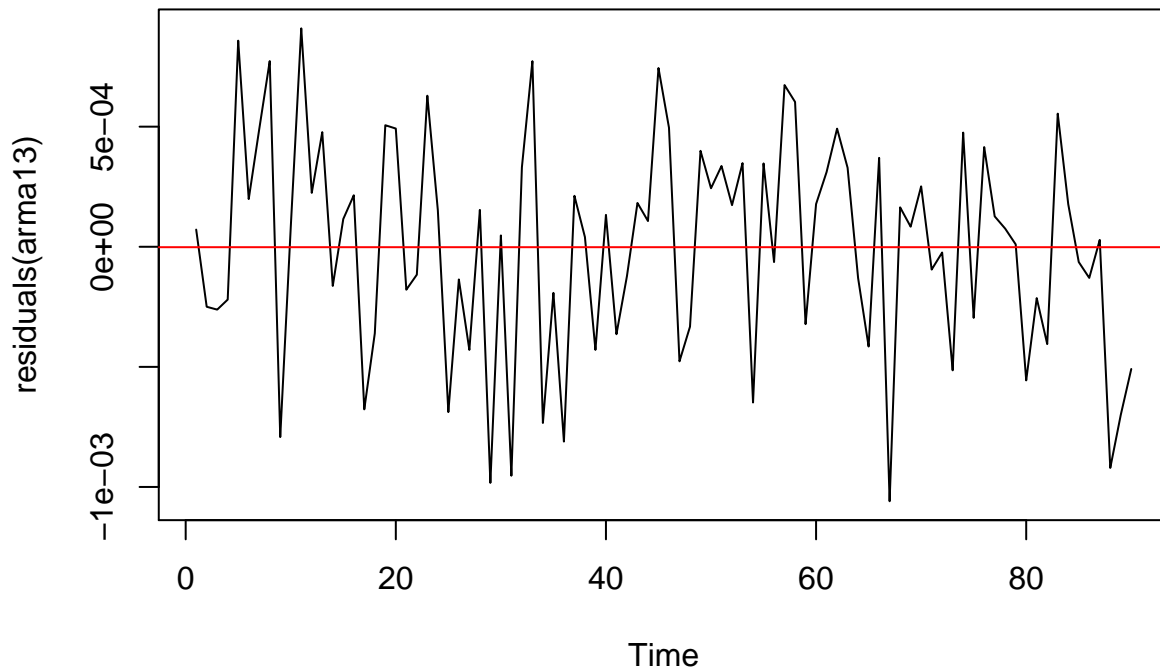
## [1] -0.208056
-0.4117-1.96*0.1039

## [1] -0.615344
spec.arma(ar=c( -0.6946),ma=c(1.2634,0.0446,-0.4117))
```



```
plot(residuals(arma13), main = "Residuals of ARMA(1,3) Model") + abline(h = mean(residuals(arma13)), col = "red")
```

## Residuals of ARMA(1,3) Model



```
## integer(0)
mean(residuals(arma13))

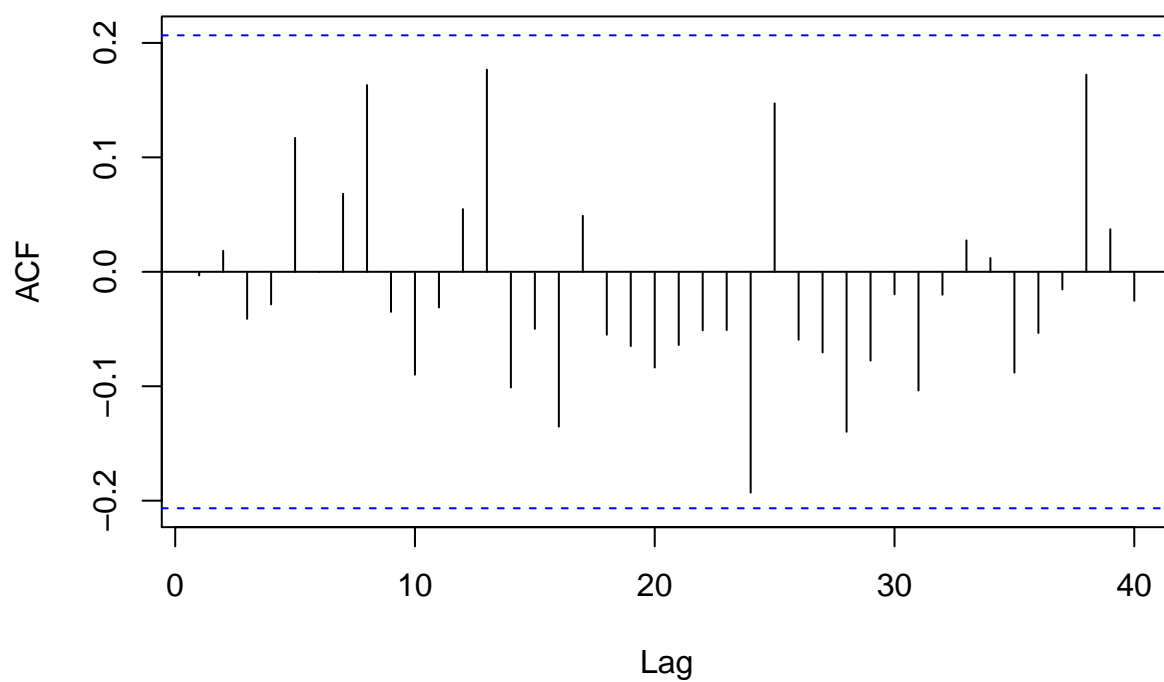
## [1] -1.820197e-06
Box.test(residuals(arma13), type = "Ljung-Box", lag = sqrt(90), fitdf = 4)

##
## Box-Ljung test
##
## data: residuals(arma13)
## X-squared = 4.8822, df = 5.4868, p-value = 0.4941
Box.test(residuals(arma13), type = "Box-Pierce", lag = sqrt(90), fitdf = 4)

##
## Box-Pierce test
##
## data: residuals(arma13)
## X-squared = 4.413, df = 5.4868, p-value = 0.5565
Box.test((residuals(arma13))^2, type = "Ljung-Box", lag = sqrt(90), fitdf = 0)

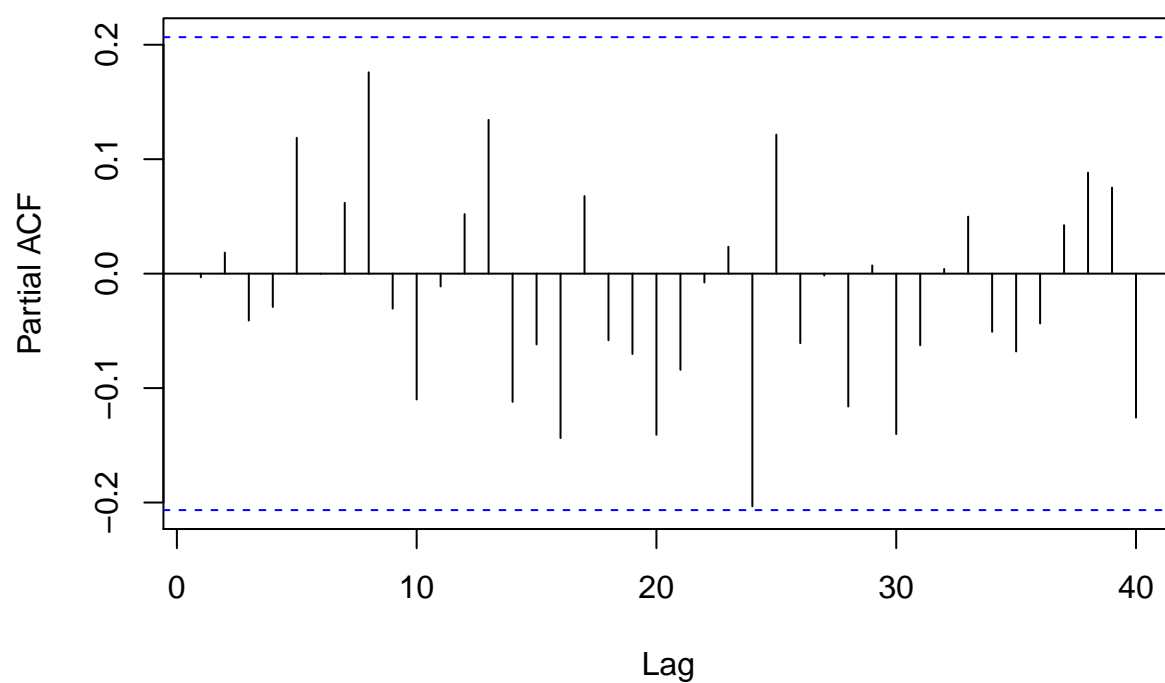
##
## Box-Ljung test
##
## data: (residuals(arma13))^2
## X-squared = 7.8993, df = 9.4868, p-value = 0.5914
acf(residuals(arma13), main = "Autocorrelation", lag.max = 40)
```

## Autocorrelation



```
pacf(residuals(arma13),main = "Partial Autocorrelation", lag.max = 40)
```

## Partial Autocorrelation

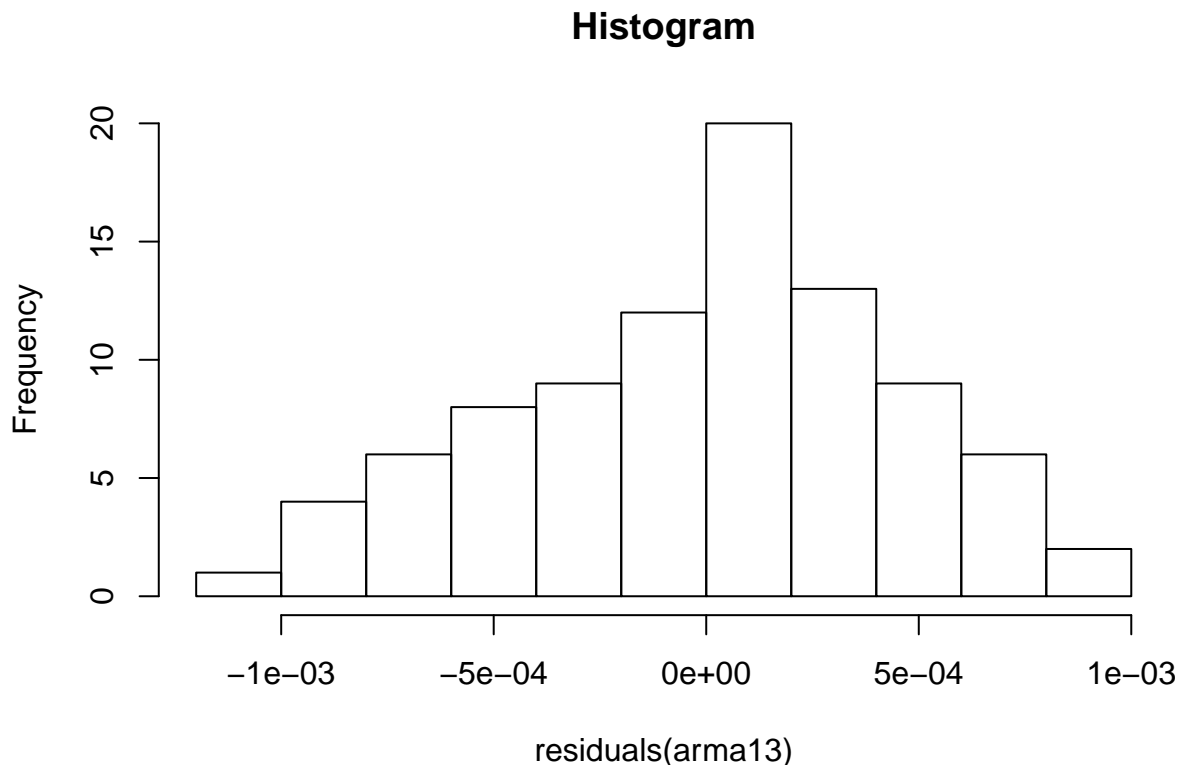


```
residuals(arma13)
```

```
## Time Series:
```

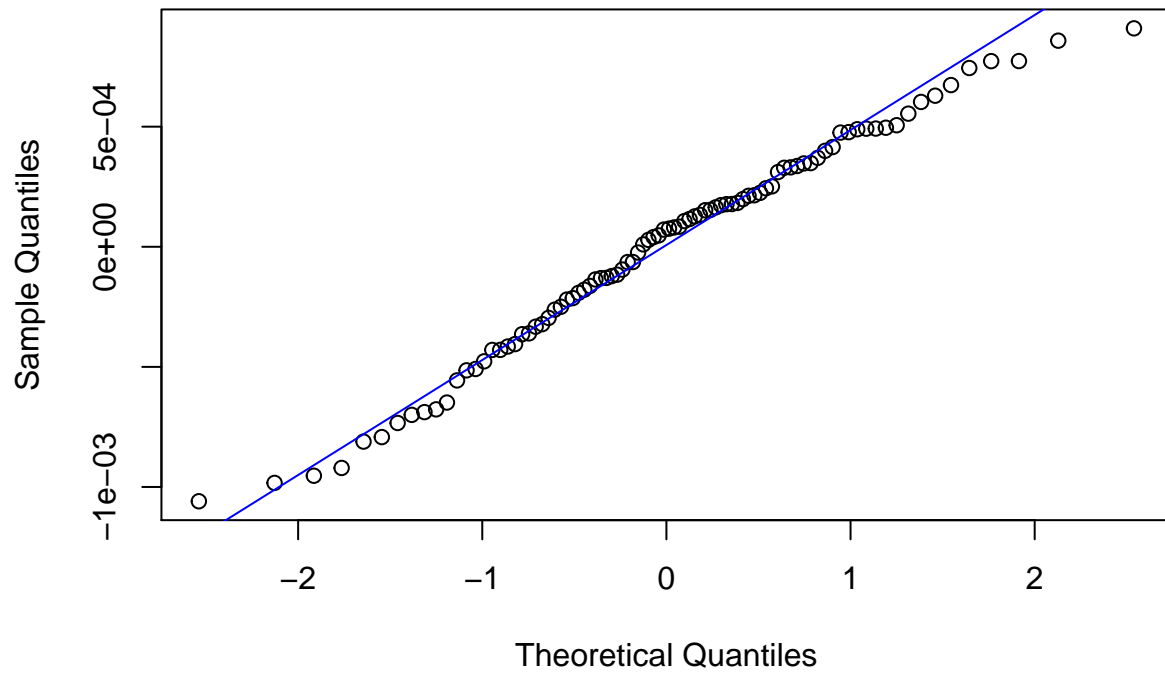
```
## Start = 1
## End = 90
## Frequency = 1
## [1] 7.143938e-05 -2.498605e-04 -2.616619e-04 -2.199328e-04 8.580568e-04
## [6] 1.985452e-04 4.892053e-04 7.730950e-04 -7.923923e-04 8.050398e-05
## [11] 9.093232e-04 2.244046e-04 4.772202e-04 -1.630998e-04 1.156068e-04
## [16] 2.143730e-04 -6.769569e-04 -3.604775e-04 5.059705e-04 4.921805e-04
## [21] -1.787074e-04 -1.162761e-04 6.288940e-04 1.520925e-04 -6.882989e-04
## [26] -1.359725e-04 -4.294672e-04 1.534977e-04 -9.830909e-04 4.731146e-05
## [31] -9.532685e-04 3.306404e-04 7.727759e-04 -7.335883e-04 -1.918131e-04
## [36] -8.110936e-04 2.121699e-04 4.033582e-05 -4.293607e-04 1.327807e-04
## [41] -3.640805e-04 -1.219312e-04 1.823581e-04 1.070695e-04 7.441238e-04
## [46] 4.955700e-04 -4.769873e-04 -3.327187e-04 3.994017e-04 2.436055e-04
## [51] 3.363244e-04 1.733440e-04 3.482052e-04 -6.484645e-04 3.468893e-04
## [56] -6.378870e-05 6.731680e-04 6.029706e-04 -3.221932e-04 1.778275e-04
## [61] 3.110458e-04 4.916758e-04 3.288124e-04 -1.304730e-04 -4.152959e-04
## [66] 3.700831e-04 -1.059451e-03 1.640240e-04 8.359653e-05 2.514944e-04
## [71] -9.488819e-05 -2.402789e-05 -5.145283e-04 4.753931e-04 -2.961457e-04
## [76] 4.150932e-04 1.265800e-04 7.542871e-05 9.818494e-06 -5.563056e-04
## [81] -2.137529e-04 -4.052192e-04 5.542104e-04 1.773944e-04 -6.356626e-05
## [86] -1.296840e-04 2.858973e-05 -9.208639e-04 -6.998654e-04 -5.087884e-04
```

```
hist(residuals(arma13),main = "Histogram")
```



```
qqnorm(residuals(arma13))
qqline(residuals(arma13),col = "blue")
```

## Normal Q-Q Plot

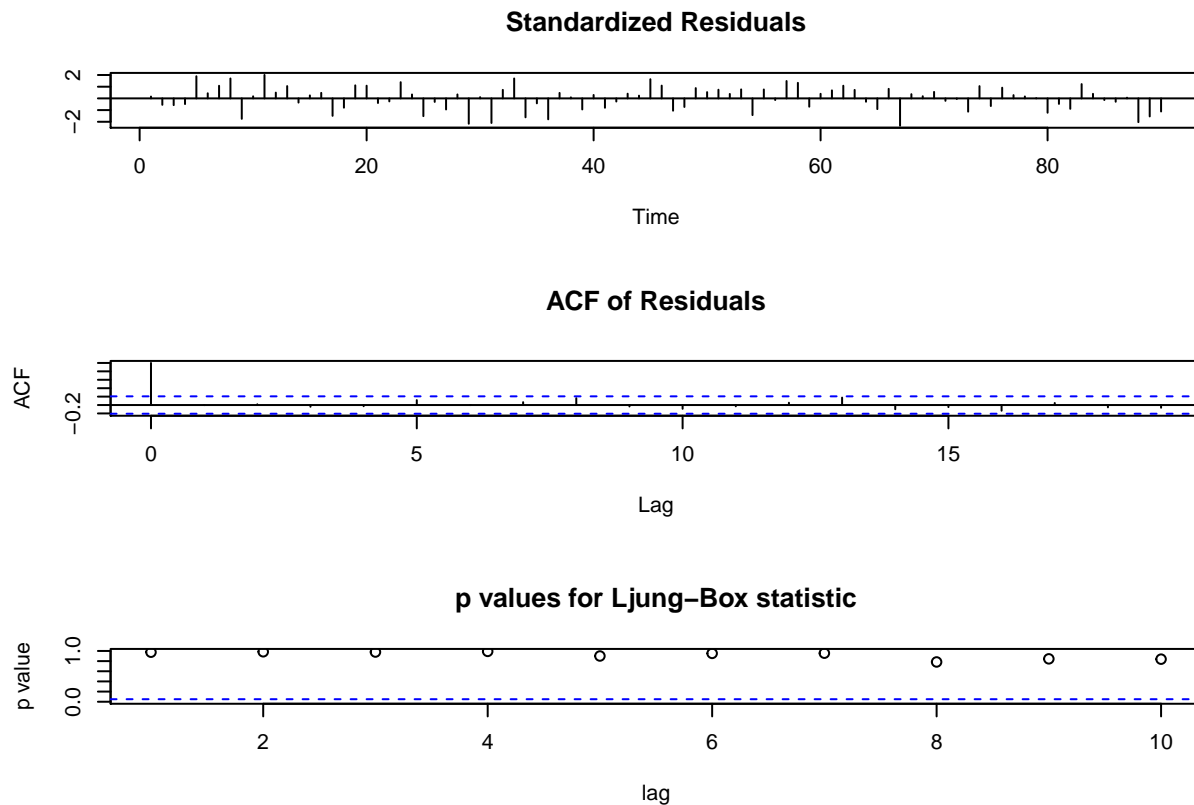


```
shapiro.test(residuals(arma13))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(arma13)  
## W = 0.98285, p-value = 0.2828
```

```
tsdiag(arima(chemneg1, order=c(1,0,3)))
```





Trying ARMA(2,1)

```
arma21 <- arima(chemneg1, order = c(2, 0, 1), method = c("ML"))
arma21

##
## Call:
## arima(x = chemneg1, order = c(2, 0, 1), method = c("ML"))
##
## Coefficients:
##      ar1      ar2      ma1  intercept
##    -0.0077 -0.2791  0.5145     0.9934
## s.e.   0.1879   0.1316  0.1875     0.0001
##
## sigma^2 estimated as 2.214e-07:  log likelihood = 561.54,  aic = -1115.09
-0.0077+1.96*0.1879 #ar1 contains 0

## [1] 0.360584
-0.0077-1.96*0.1879

## [1] -0.375984
-0.2791+1.96*0.1316

## [1] -0.021164
-0.2791-1.96*0.1316

## [1] -0.537036
```

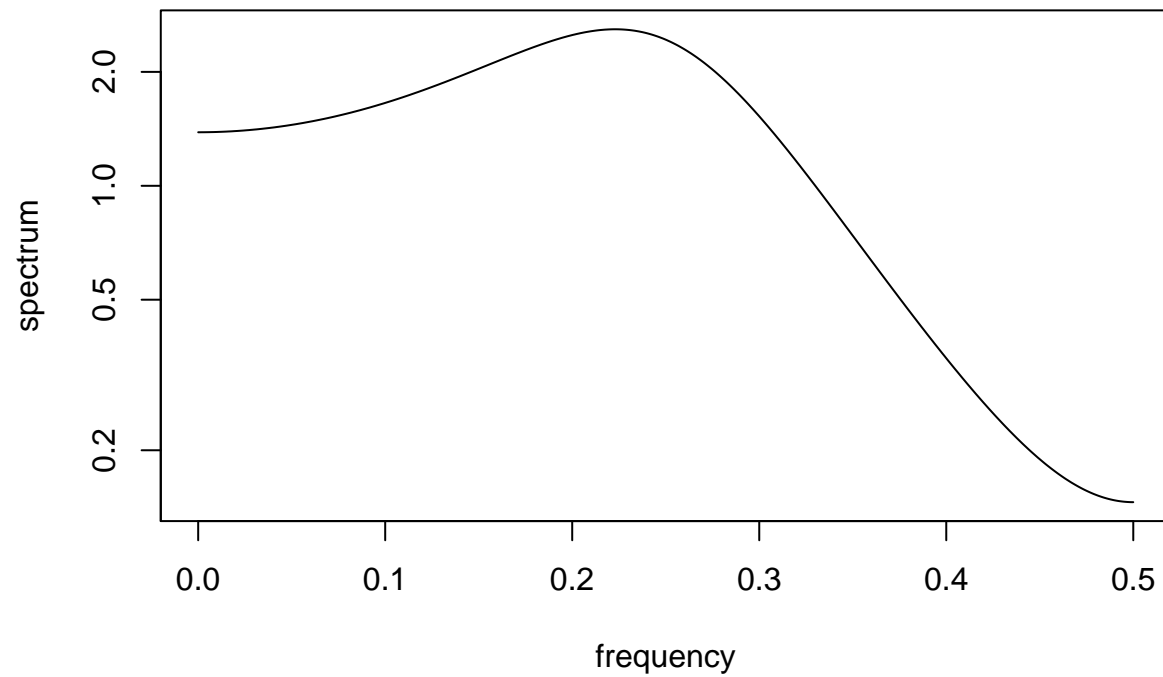
```
0.5145+1.96*0.1875
```

```
## [1] 0.882
```

```
0.5145-1.96*0.1875
```

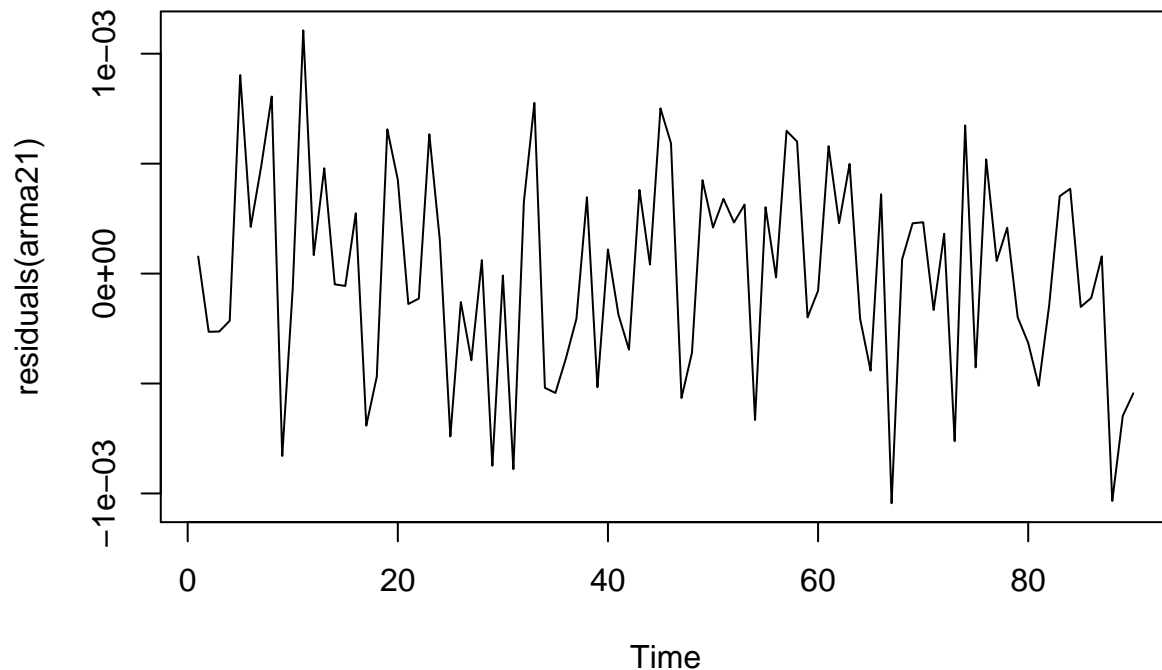
```
## [1] 0.147
```

```
spec.arma(ar=c( -0.0077, -0.2791),ma=c(0.5145))
```



```
plot(residuals(arma21), main = "Residuals of ARMA(2,1) Model")
```

## Residuals of ARMA(2,1) Model



```
mean(residuals(arma21))
```

```
## [1] -7.819497e-07
```

```
Box.test(residuals(arma21), type = "Ljung-Box", lag = sqrt(90), fitdf = 3)
```

```
##
```

```
## Box-Ljung test
```

```
##
```

```
## data: residuals(arma21)
```

```
## X-squared = 7.3699, df = 6.4868, p-value = 0.3375
```

```
Box.test(residuals(arma21), type = "Box-Pierce", lag = sqrt(90), fitdf = 3)
```

```
##
```

```
## Box-Pierce test
```

```
##
```

```
## data: residuals(arma21)
```

```
## X-squared = 6.609, df = 6.4868, p-value = 0.4131
```

```
Box.test(residuals(arma21)^2, type = "Ljung-Box", lag = sqrt(90), fitdf = 0)
```

```
##
```

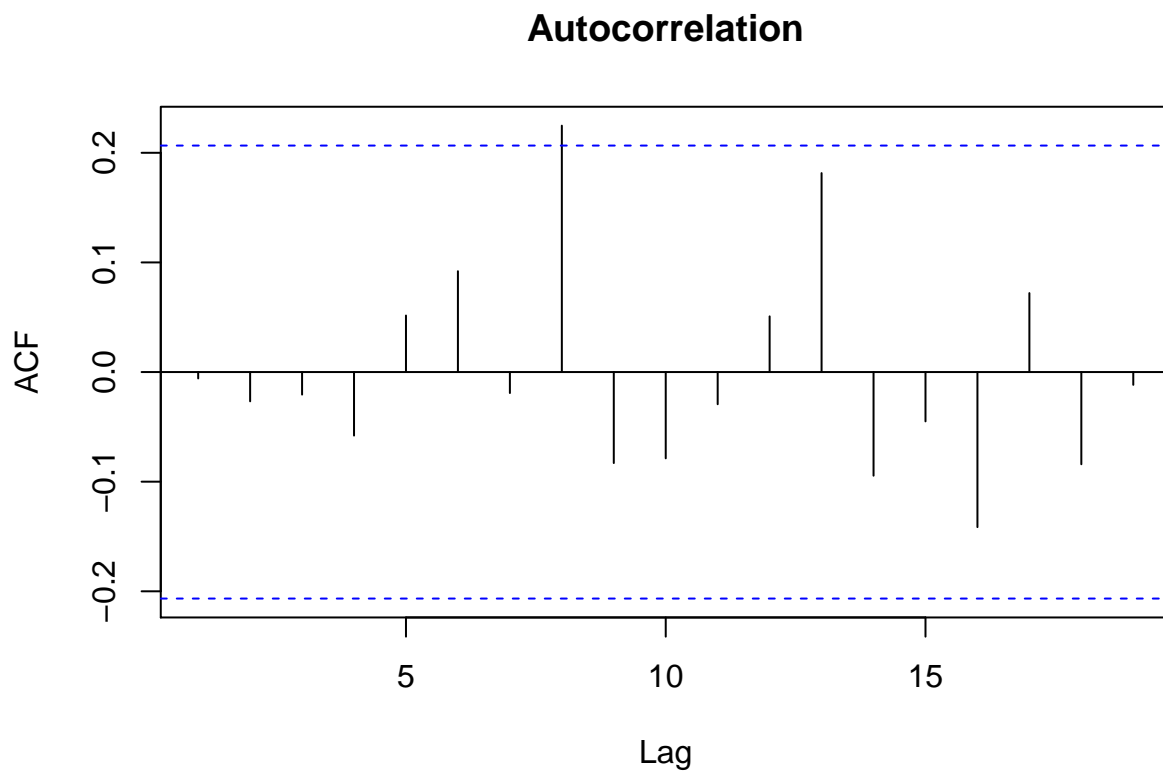
```
## Box-Ljung test
```

```
##
```

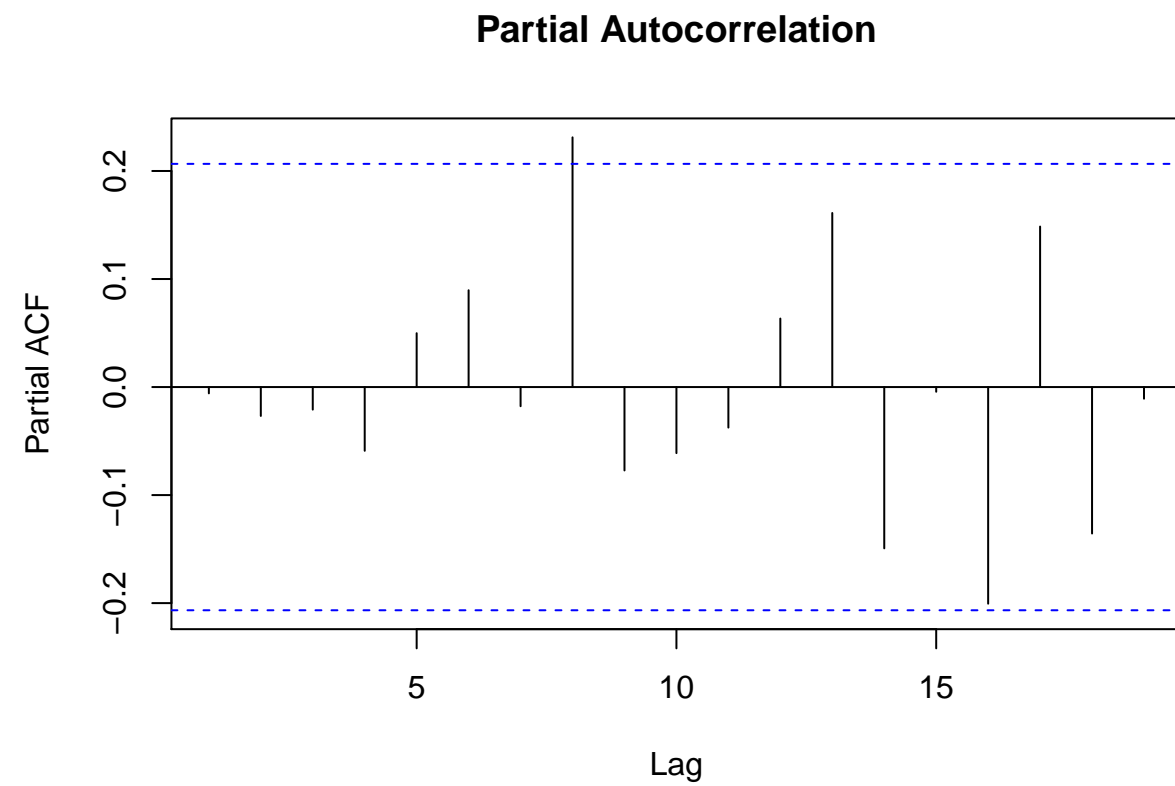
```
## data: residuals(arma21)^2
```

```
## X-squared = 20.946, df = 9.4868, p-value = 0.01664
```

```
acf(residuals(arma21),main = "Autocorrelation")
```



```
pacf(residuals(arma21),main = "Partial Autocorrelation")
```



```
residuals(arma21)
```

```
## Time Series:
```

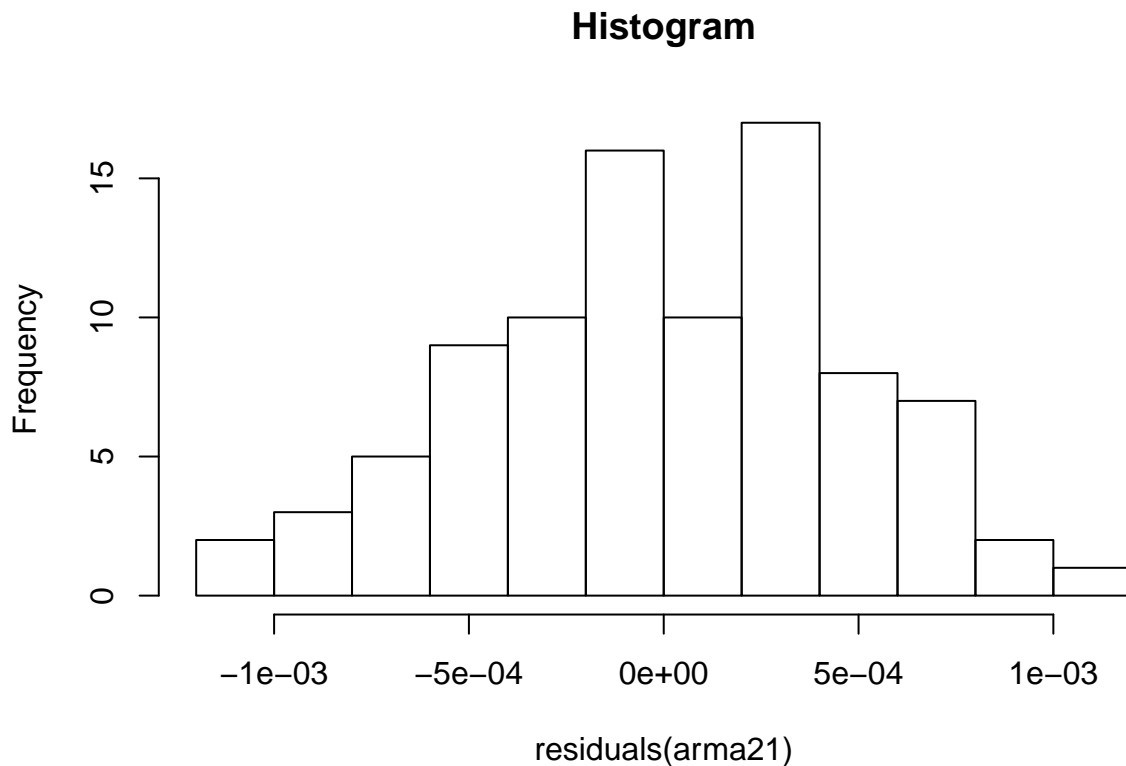
```
## Start = 1
```

```
## End = 90
```

```
## Frequency = 1
```

```
## [1] 7.772356e-05 -2.646526e-04 -2.636113e-04 -2.147209e-04 9.025901e-04
## [6] 2.125071e-04 4.884816e-04 8.054423e-04 -8.297568e-04 -6.710487e-05
## [11] 1.106237e-03 8.405615e-05 4.792520e-04 -4.891885e-05 -5.627629e-05
## [16] 2.741301e-04 -6.914333e-04 -4.685100e-04 6.562616e-04 4.265796e-04
## [21] -1.383484e-04 -1.134167e-04 6.337842e-04 1.513548e-04 -7.409781e-04
## [26] -1.295903e-04 -3.943661e-04 6.128848e-05 -8.739321e-04 -8.534394e-06
## [31] -8.895756e-04 3.267981e-04 7.764751e-04 -5.195581e-04 -5.429351e-04
## [36] -3.852709e-04 -2.052378e-04 3.475823e-04 -5.169333e-04 1.096544e-04
## [41] -1.870691e-04 -3.456223e-04 3.803335e-04 4.123506e-05 7.513912e-04
## [46] 5.927055e-04 -5.655659e-04 -3.592092e-04 4.247749e-04 2.098047e-04
## [51] 3.398415e-04 2.324536e-04 3.136903e-04 -6.658585e-04 3.023355e-04
## [56] -1.727433e-05 6.492669e-04 6.006431e-04 -1.989453e-04 -7.760921e-05
## [61] 5.792384e-04 2.291449e-04 4.989392e-04 -2.049436e-04 -4.414209e-04
## [66] 3.607586e-04 -1.044443e-03 6.489486e-05 2.288661e-04 2.335086e-04
## [71] -1.656406e-04 1.811770e-04 -7.622517e-04 6.738581e-04 -4.264899e-04
## [76] 5.196589e-04 5.730290e-05 2.083651e-04 -1.987574e-04 -3.156824e-04
## [81] -5.099224e-04 -1.421422e-04 3.520361e-04 3.858246e-04 -1.515552e-04
## [86] -1.106982e-04 7.829940e-05 -1.034172e-03 -6.472587e-04 -5.447279e-04
```

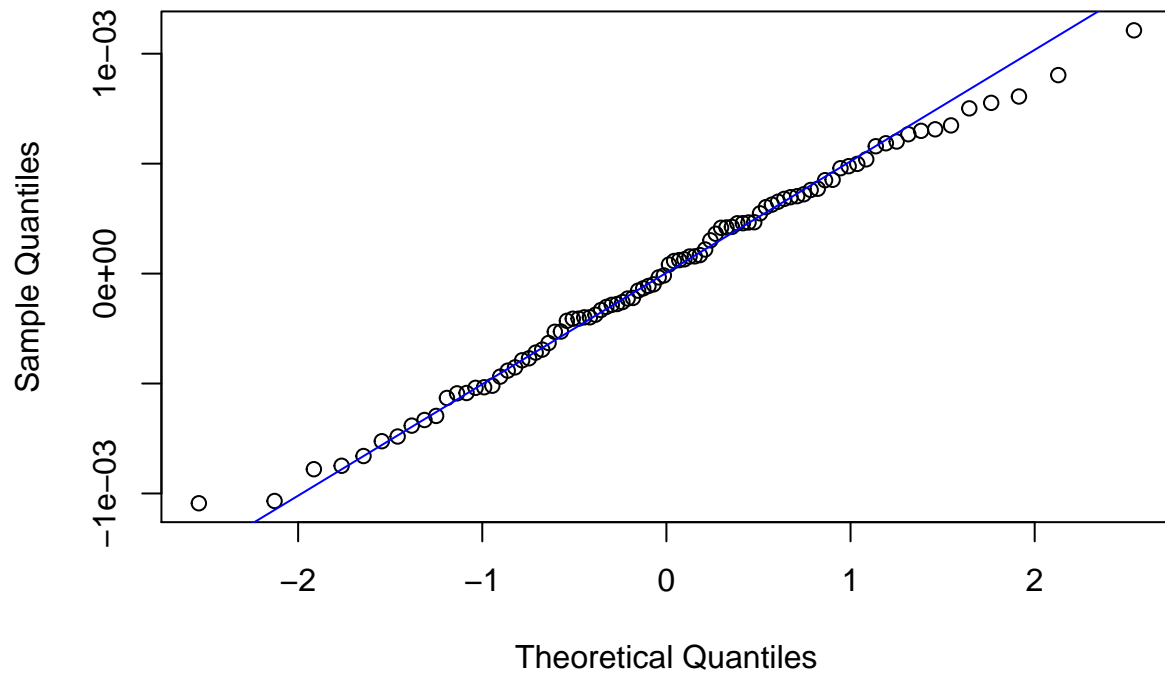
```
hist(residuals(arma21),main = "Histogram")
```



```
qqnorm(residuals(arma21))
```

```
qqline(residuals(arma21),col = "blue")
```

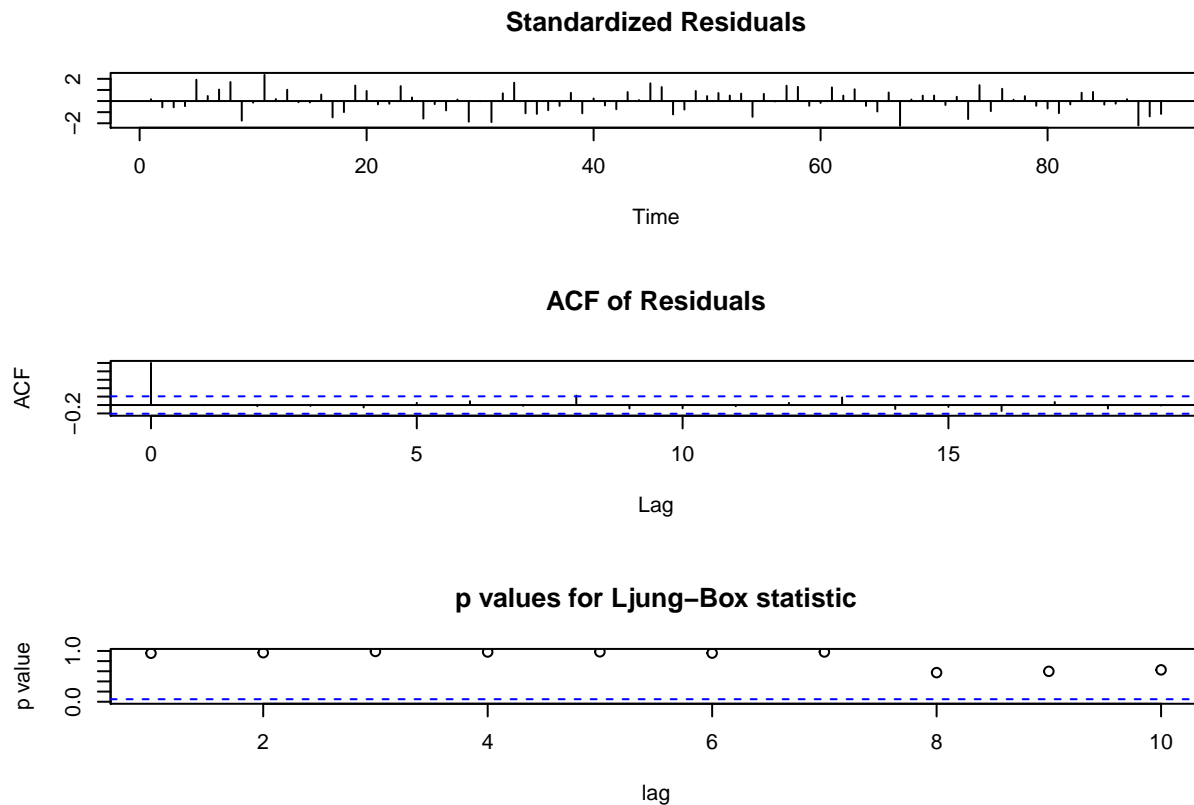
## Normal Q-Q Plot



```
shapiro.test(residuals(arma21))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(arma21)  
## W = 0.99098, p-value = 0.8003
```

```
tsdiag(arima(chemneg1, order=c(2,0,1)))
```



Trying ARMA(0,3)

```
arma03 <- arima(chemneg1, order = c(0, 0, 3), method = c("ML"))
arma03

##
## Call:
## arima(x = chemneg1, order = c(0, 0, 3), method = c("ML"))
##
## Coefficients:
##      ma1      ma2      ma3 intercept
##    0.5137 -0.2883 -0.1797    0.9934
## s.e. 0.1086  0.1048  0.1007    0.0001
##
## sigma^2 estimated as 2.206e-07:  log likelihood = 561.7,  aic = -1115.4
0.5137+1.96*0.1086

## [1] 0.726556
0.5137-1.96*0.1086

## [1] 0.300844
-0.2883+1.96*0.1048

## [1] -0.082892
-0.2883-1.96*0.1048

## [1] -0.493708
```

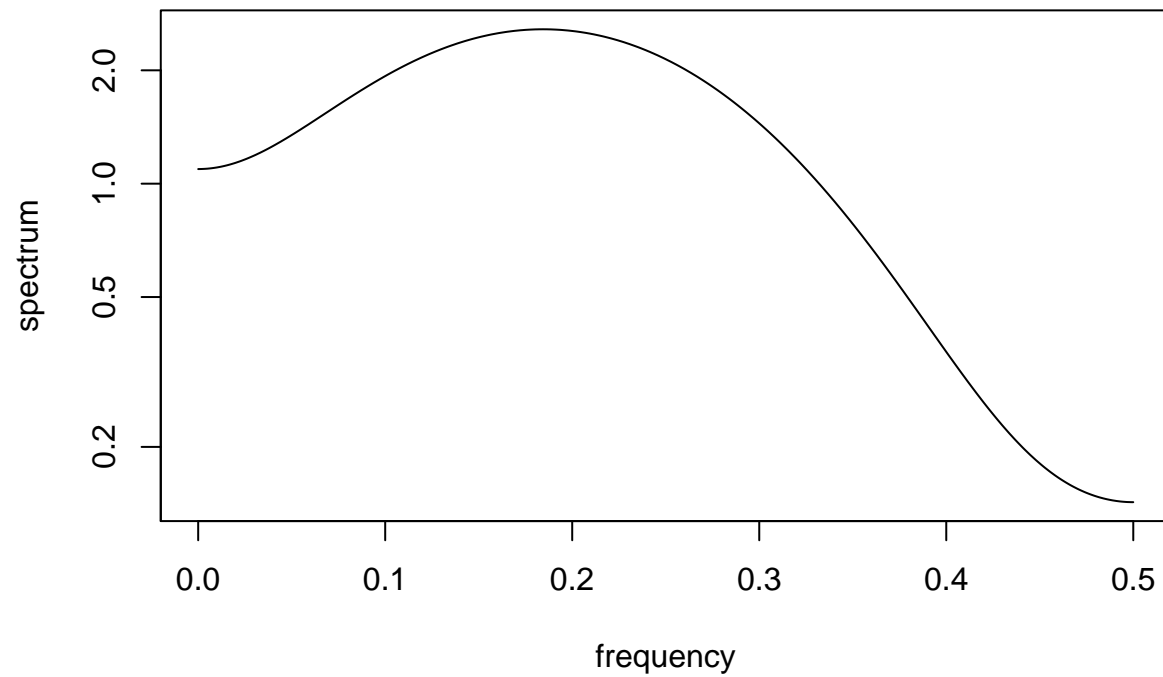
```
-0.1797+1.96*0.1007 # ma3 contains 0
```

```
## [1] 0.017672
```

```
-0.1797-1.96*0.1007
```

```
## [1] -0.377072
```

```
spec.arma(ar=c(0),ma=c(0.5137,-0.2883, -0.1797))
```



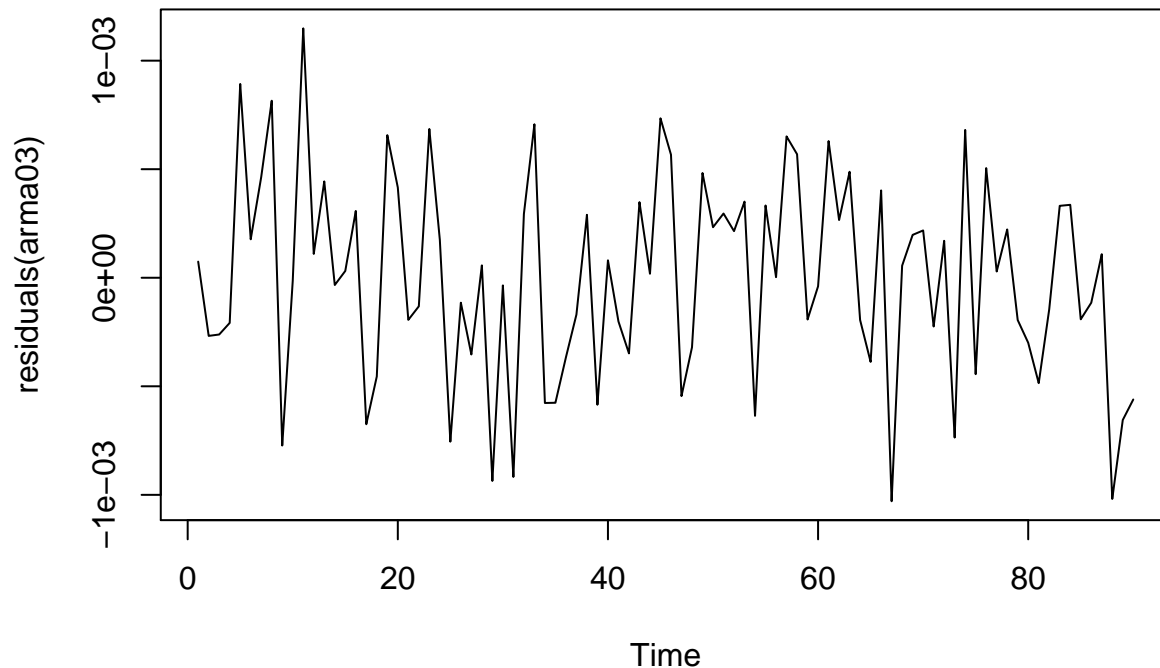
```
mean(residuals(arma03))
```

```
## [1] -1.094635e-06
```

```
plot(residuals(arma03), main = "Residuals of ARMA(0,3) Model")
```



## Residuals of ARMA(0,3) Model



```
Box.test(residuals(arma03), lag = sqrt(90), type = "Ljung-Box", fitdf = 3)
```

```
##  
## Box-Ljung test  
##  
## data: residuals(arma03)  
## X-squared = 7.0006, df = 6.4868, p-value = 0.3729
```

```
Box.test(residuals(arma03), type = "Box-Pierce", lag = sqrt(90), fitdf = 3)
```

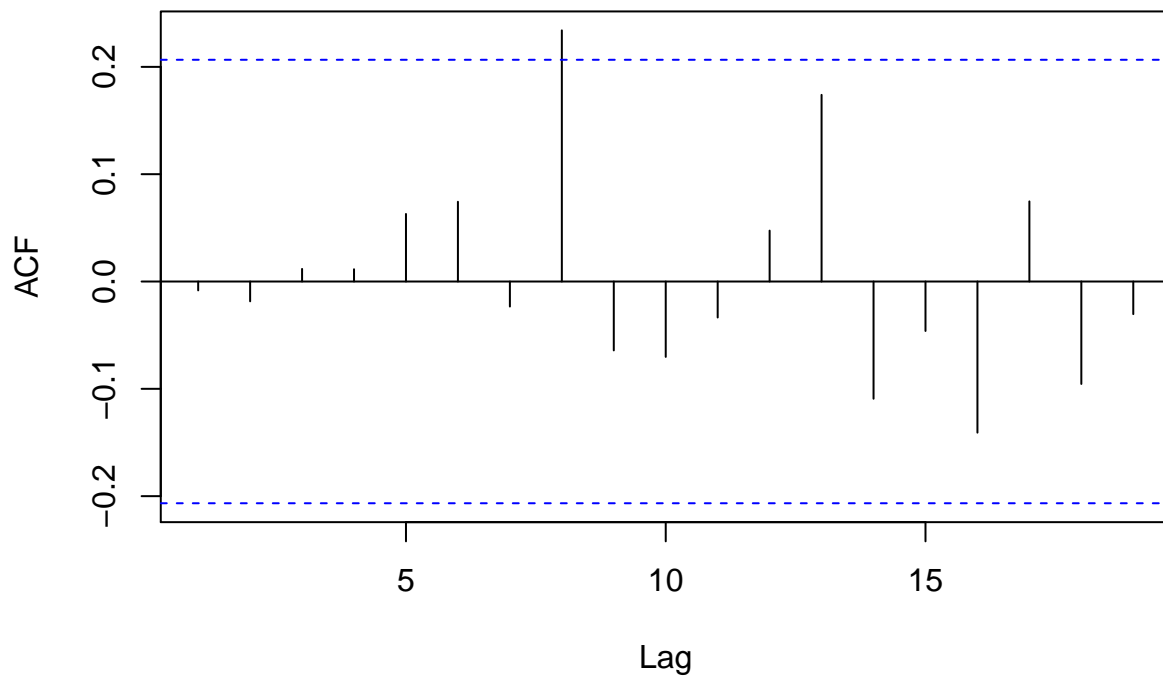
```
##  
## Box-Pierce test  
##  
## data: residuals(arma03)  
## X-squared = 6.2639, df = 6.4868, p-value = 0.4506
```

```
Box.test(residuals(arma03)^2, type = "Ljung", lag = sqrt(90), fitdf = 0)
```

```
##  
## Box-Ljung test  
##  
## data: residuals(arma03)^2  
## X-squared = 21.49, df = 9.4868, p-value = 0.01381
```

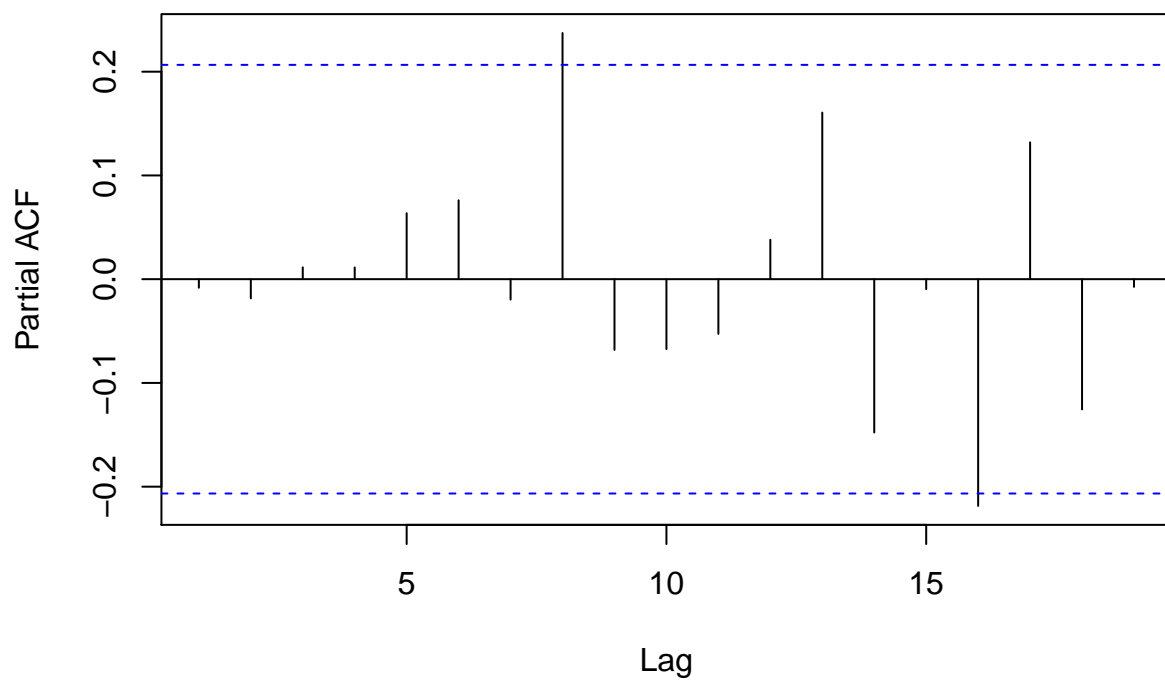
```
acf(residuals(arma03), main = "Autocorrelation")
```

## Autocorrelation



```
pacf(residuals(arma03),main = "Partial Autocorrelation")
```

## Partial Autocorrelation

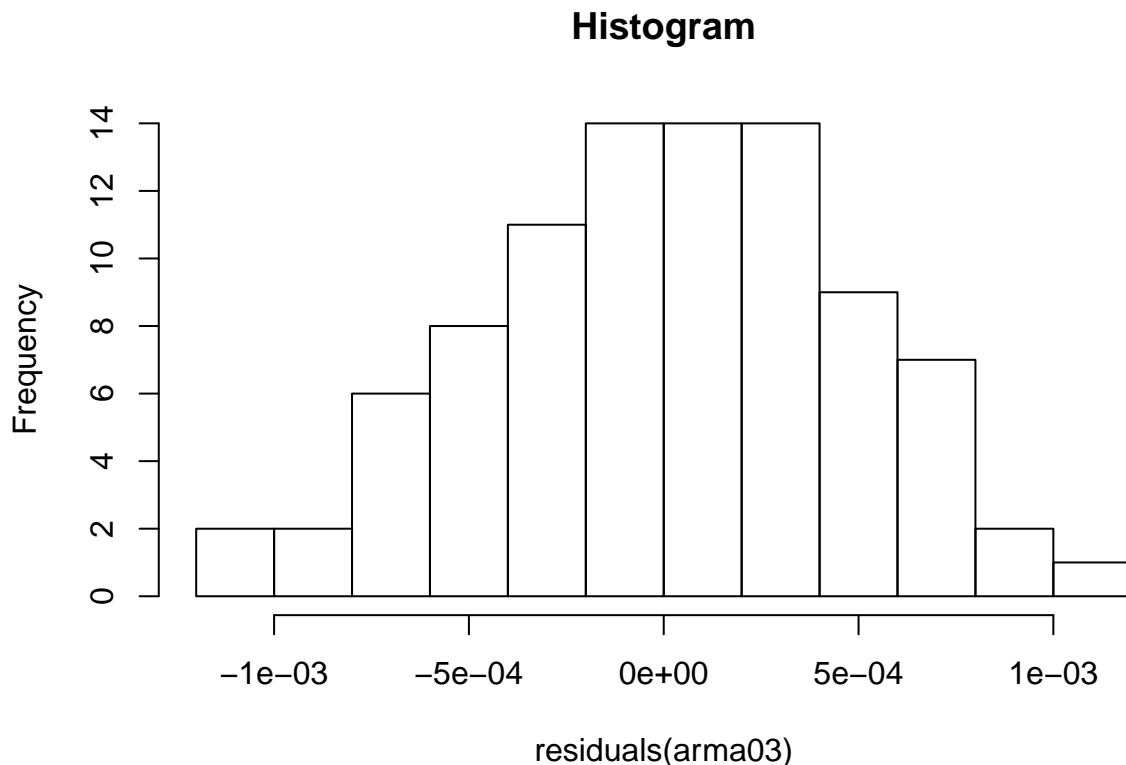


```
residuals(arma03)
```

```
## Time Series:
```

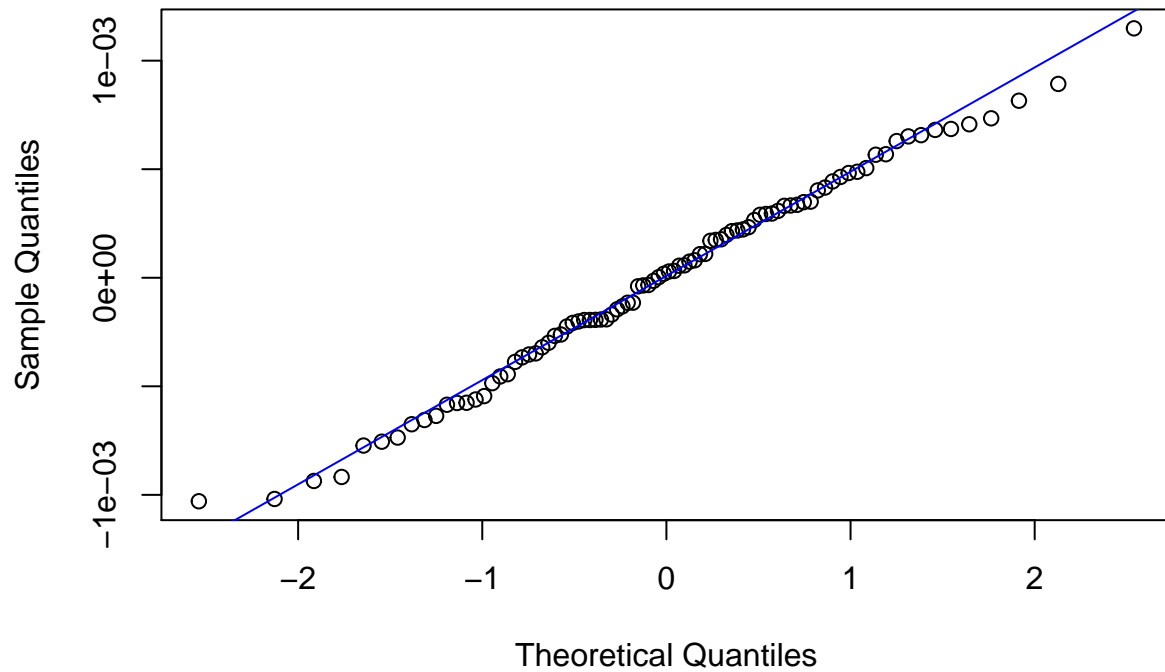
```
## Start = 1
## End = 90
## Frequency = 1
## [1] 7.430028e-05 -2.677907e-04 -2.613705e-04 -2.079899e-04 8.927597e-04
## [6] 1.770593e-04 4.643192e-04 8.154492e-04 -7.729219e-04 -1.419762e-05
## [11] 1.148783e-03 1.097678e-04 4.438071e-04 -3.342454e-05 3.127519e-05
## [16] 3.078106e-04 -6.747649e-04 -4.546048e-04 6.569266e-04 4.154894e-04
## [21] -1.939466e-04 -1.315689e-04 6.854360e-04 1.742247e-04 -7.550306e-04
## [26] -1.148305e-04 -3.534655e-04 5.709527e-05 -9.360511e-04 -3.517213e-05
## [31] -9.175663e-04 2.937536e-04 7.069775e-04 -5.768969e-04 -5.758014e-04
## [36] -3.666607e-04 -1.694672e-04 2.899458e-04 -5.849083e-04 8.026641e-05
## [41] -2.012111e-04 -3.483770e-04 3.483352e-04 1.838282e-05 7.344301e-04
## [46] 5.666738e-04 -5.450862e-04 -3.202778e-04 4.824143e-04 2.327538e-04
## [51] 2.958548e-04 2.147285e-04 3.506899e-04 -6.361411e-04 3.329124e-04
## [56] 2.698475e-06 6.511227e-04 5.689575e-04 -1.923024e-04 -3.927184e-05
## [61] 6.295571e-04 2.659041e-04 4.881931e-04 -1.945878e-04 -3.876013e-04
## [66] 4.023231e-04 -1.029430e-03 5.552350e-05 1.973760e-04 2.179093e-04
## [71] -2.245848e-04 1.701876e-04 -7.361882e-04 6.809946e-04 -4.442172e-04
## [76] 5.051992e-04 2.885743e-05 2.225962e-04 -1.947849e-04 -2.995766e-04
## [81] -4.854322e-04 -1.453221e-04 3.312698e-04 3.359061e-04 -1.913752e-04
## [86] -1.151443e-04 1.085913e-04 -1.018947e-03 -6.552294e-04 -5.607863e-04
```

```
hist(residuals(arma03),main = "Histogram")
```



```
qqnorm(residuals(arma03))
qqline(residuals(arma03),col = "blue")
```

## Normal Q-Q Plot



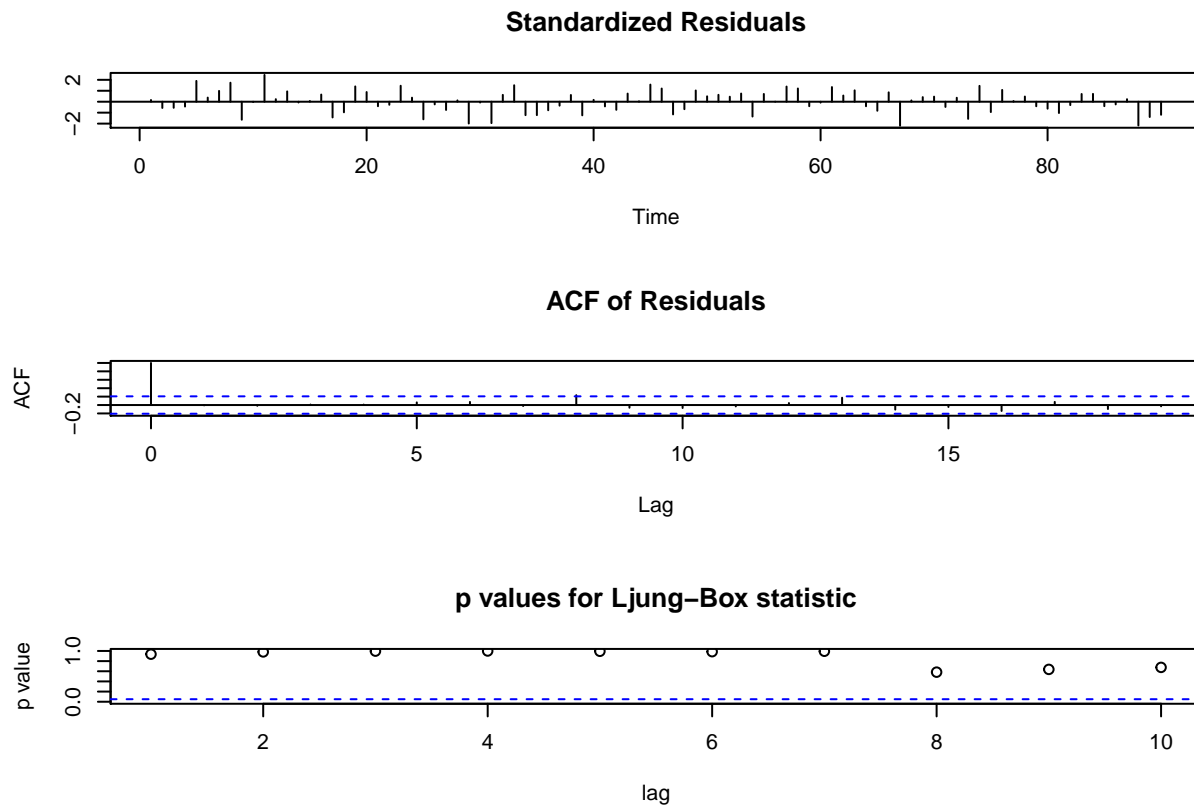
```
shapiro.test(residuals(arma03))
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: residuals(arma03)  
## W = 0.99072, p-value = 0.7824
```

```
fit.ar <- ar(residuals(arma03),method="yw")  
fit.ar
```

```
##  
## Call:  
## ar(x = residuals(arma03), method = "yw")  
##  
##  
## Order selected 0 sigma^2 estimated as 2.231e-07
```

```
tsdiag(arima(chemneg1, order=c(0,0,3)))
```



Trying ARMA(0,2)

```
arma02 <- arima(chemneg1, order = c(0, 0, 2), method = c("ML"))
arma02

##
## Call:
## arima(x = chemneg1, order = c(0, 0, 2), method = c("ML"))
##
## Coefficients:
##          ma1      ma2  intercept
##          0.4548 -0.2093      0.9934
## s.e.  0.1219  0.1196      0.0001
##
## sigma^2 estimated as 2.286e-07:  log likelihood = 560.16,  aic = -1114.31
0.4548+1.96*0.1219

## [1] 0.693724
0.4548-1.96*0.1219

## [1] 0.215876
-0.2093+1.96*0.1196

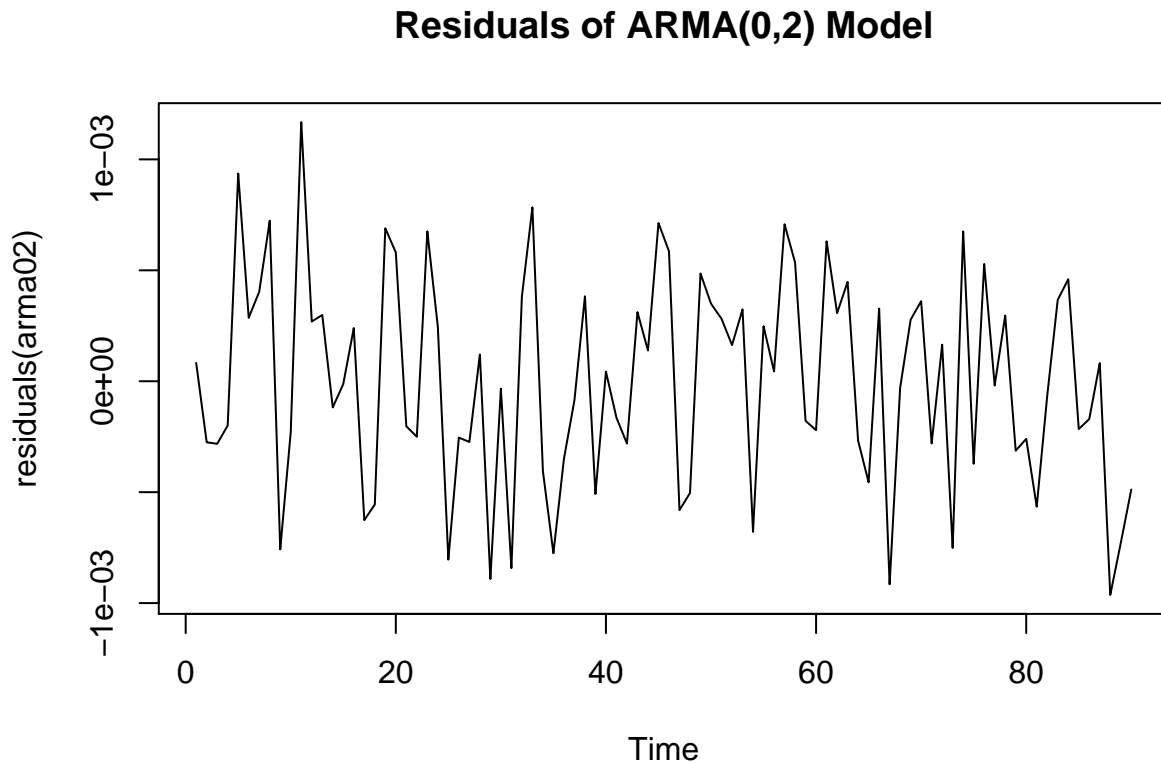
## [1] 0.025116
-0.2093-1.96*0.1196

## [1] -0.443716
```

```
mean(residuals(arma02))
```

```
## [1] -8.771119e-07
```

```
plot(residuals(arma02), main = "Residuals of ARMA(0,2) Model")
```



```
Box.test(residuals(arma02), type = "Ljung-Box", lag = sqrt(90), fitdf = 2)
```

```
##  
## Box-Ljung test  
##  
## data: residuals(arma02)  
## X-squared = 8.8436, df = 7.4868, p-value = 0.3077
```

```
Box.test(residuals(arma02), type = "Box-Pierce", lag = sqrt(90), fitdf = 2)
```

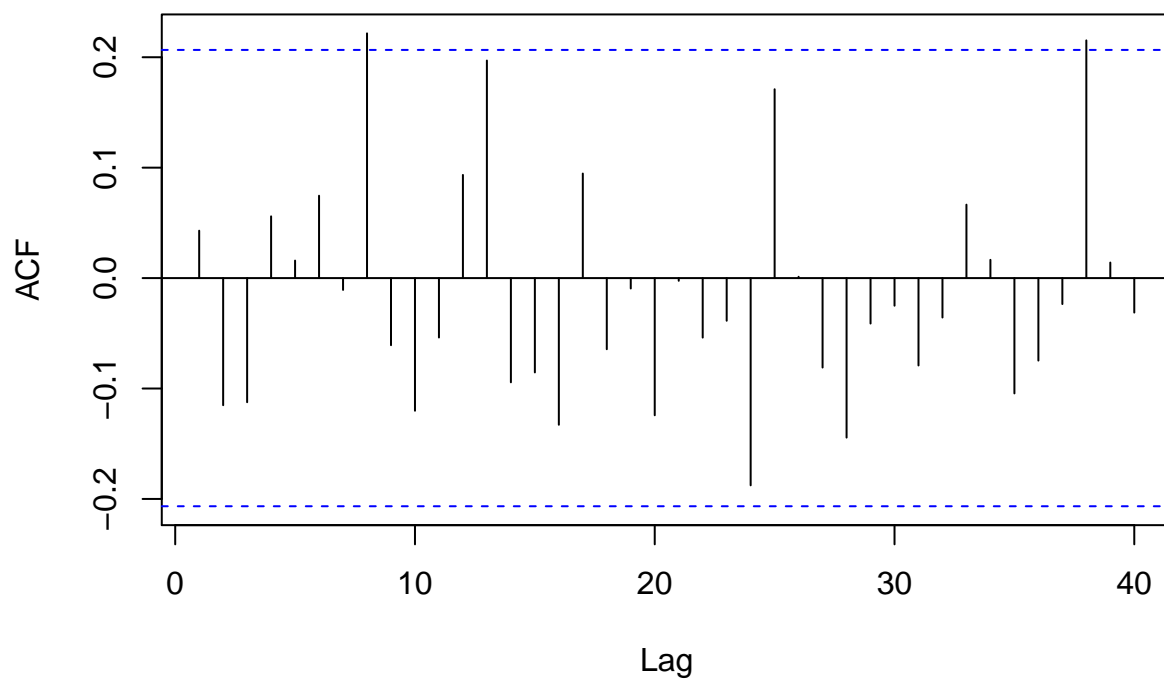
```
##  
## Box-Pierce test  
##  
## data: residuals(arma02)  
## X-squared = 8.0639, df = 7.4868, p-value = 0.3754
```

```
Box.test(residuals(arma02)^2, type = "Ljung", lag = sqrt(90), fitdf = 0)
```

```
##  
## Box-Ljung test  
##  
## data: residuals(arma02)^2  
## X-squared = 18.808, df = 9.4868, p-value = 0.03393
```

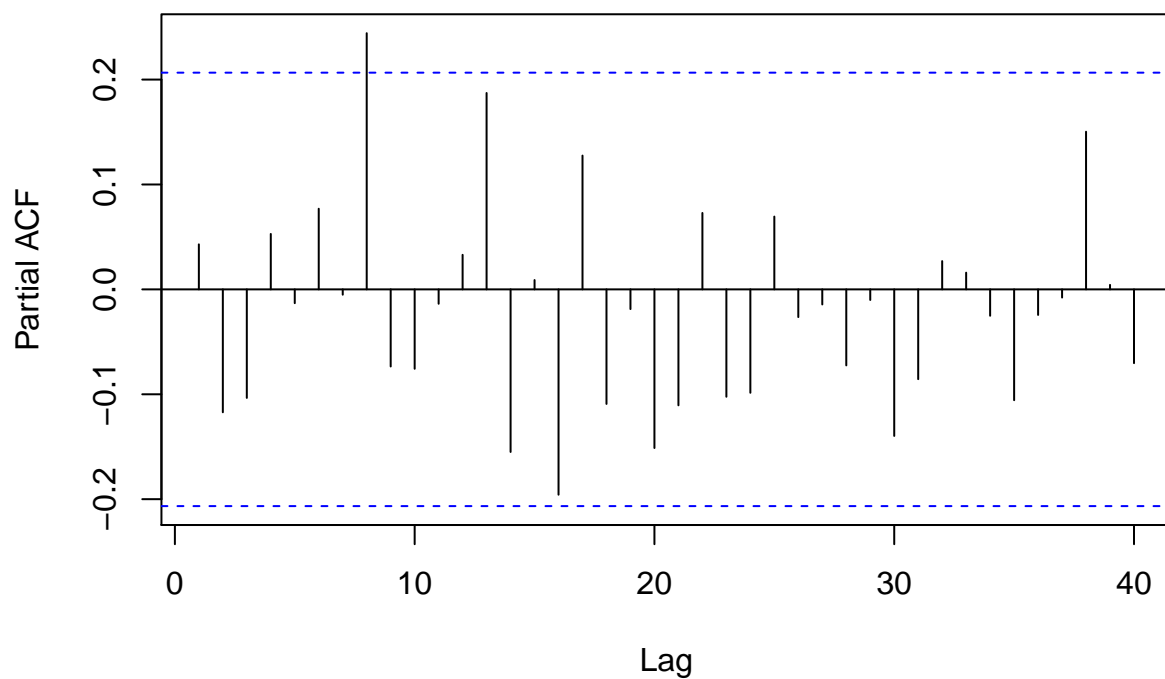
```
acf(residuals(arma02), main = "Autocorrelation", lag.max = 40)
```

## Autocorrelation



```
pacf(residuals(arma02),main = "Partial Autocorrelation", lag.max = 40)
```

## Partial Autocorrelation

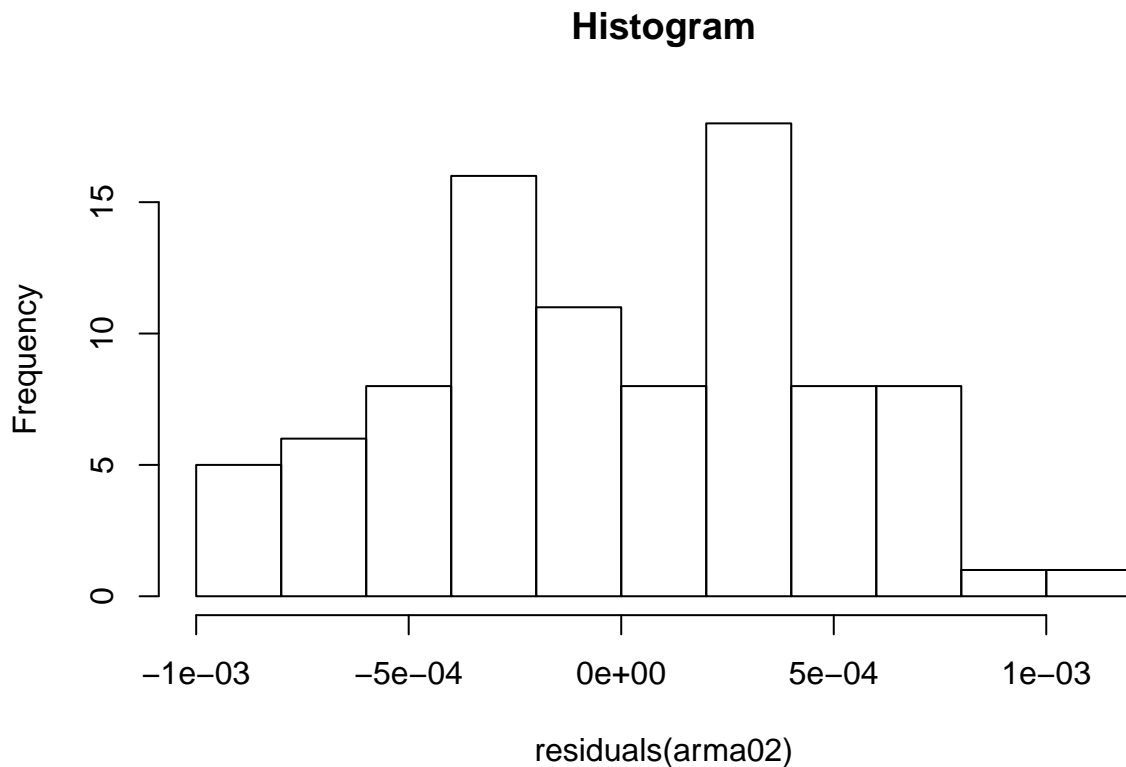


```
residuals(arma02)
```

```
## Time Series:
```

```
## Start = 1
## End = 90
## Frequency = 1
## [1] 8.257859e-05 -2.752412e-04 -2.820208e-04 -1.993827e-04 9.360806e-04
## [6] 2.860659e-04 4.034861e-04 7.240983e-04 -7.580566e-04 -2.290331e-04
## [11] 1.168155e-03 2.690166e-04 2.987971e-04 -1.179324e-04 -1.229757e-05
## [16] 2.397771e-04 -6.261725e-04 -5.555512e-04 6.892797e-04 5.805771e-04
## [21] -2.028768e-04 -2.501244e-04 6.754876e-04 2.446556e-04 -8.042730e-04
## [26] -2.540055e-04 -2.738384e-04 1.207199e-04 -8.913256e-04 -3.328462e-05
## [31] -8.423915e-04 3.819103e-04 7.837957e-04 -4.049754e-04 -7.754209e-04
## [36] -3.502014e-04 -8.614672e-05 3.829963e-04 -5.083549e-04 4.309205e-05
## [41] -1.643662e-04 -2.809421e-04 3.114030e-04 1.385693e-04 7.132959e-04
## [46] 5.857850e-04 -5.810082e-04 -5.036687e-04 4.857461e-04 3.495933e-04
## [51] 2.816696e-04 1.631144e-04 3.237619e-04 -6.790464e-04 2.481151e-04
## [56] 4.422183e-05 7.077352e-04 5.353465e-04 -1.784448e-04 -2.207570e-04
## [61] 6.307362e-04 3.072263e-04 4.477521e-04 -2.677833e-04 -4.554783e-04
## [66] 3.277757e-04 -9.153689e-04 -2.967209e-05 2.773640e-04 3.609791e-04
## [71] -2.805358e-04 1.647630e-04 -7.517263e-04 6.755323e-04 -3.722017e-04
## [76] 5.286965e-04 -1.915627e-05 2.960517e-04 -3.130703e-04 -2.595625e-04
## [81] -5.655641e-04 -6.537852e-05 3.668166e-04 4.597705e-04 -2.154293e-04
## [86] -1.705090e-04 8.180253e-05 -9.633897e-04 -7.289587e-04 -4.882075e-04
```

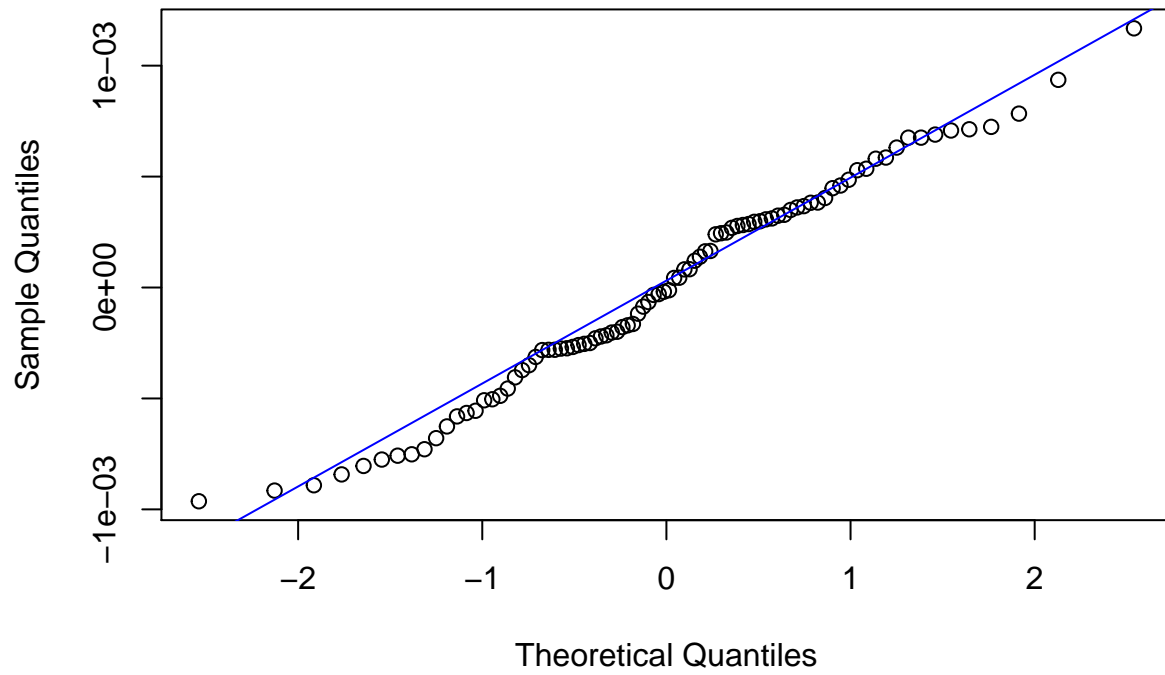
```
hist(residuals(arma02),main = "Histogram")
```



```
qqnorm(residuals(arma02))
qqline(residuals(arma02),col = "blue")
```



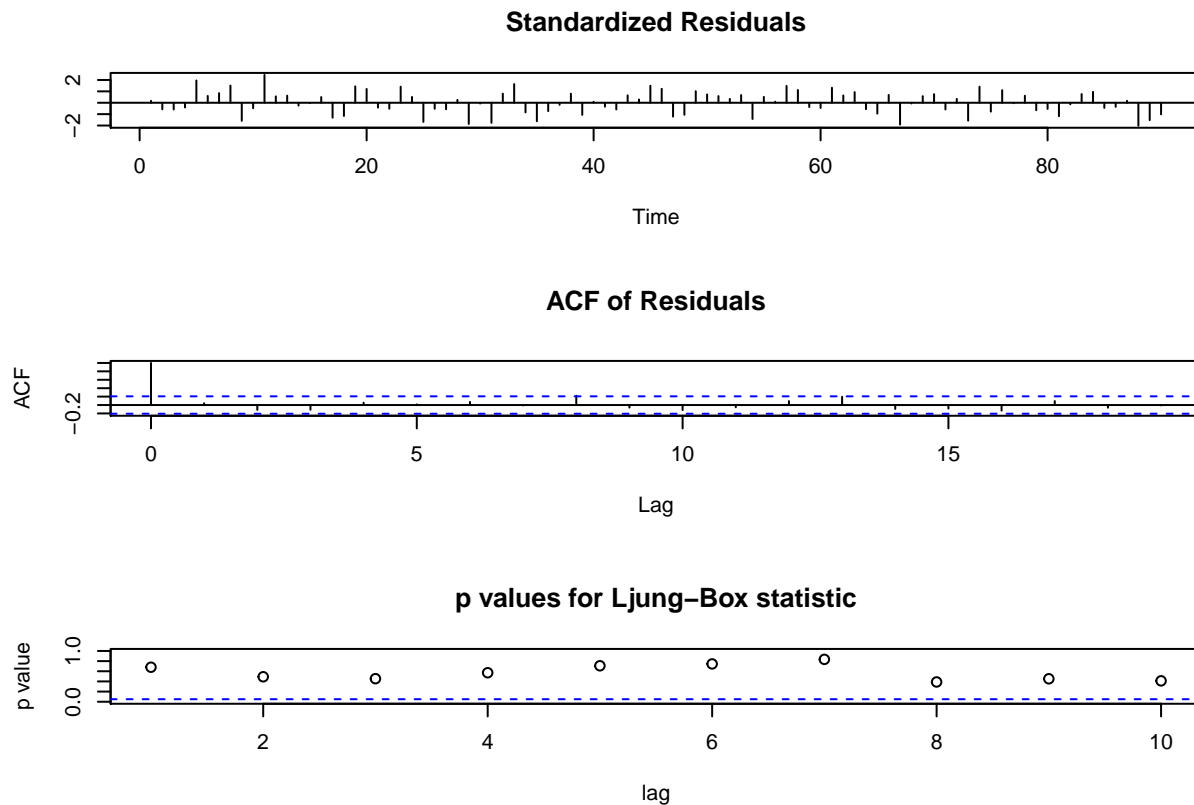
## Normal Q-Q Plot



```
shapiro.test(residuals(arma02))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  residuals(arma02)  
## W = 0.98232, p-value = 0.2607
```

```
tsdiag(arima(chemneg1, order=c(0,0,2)))
```



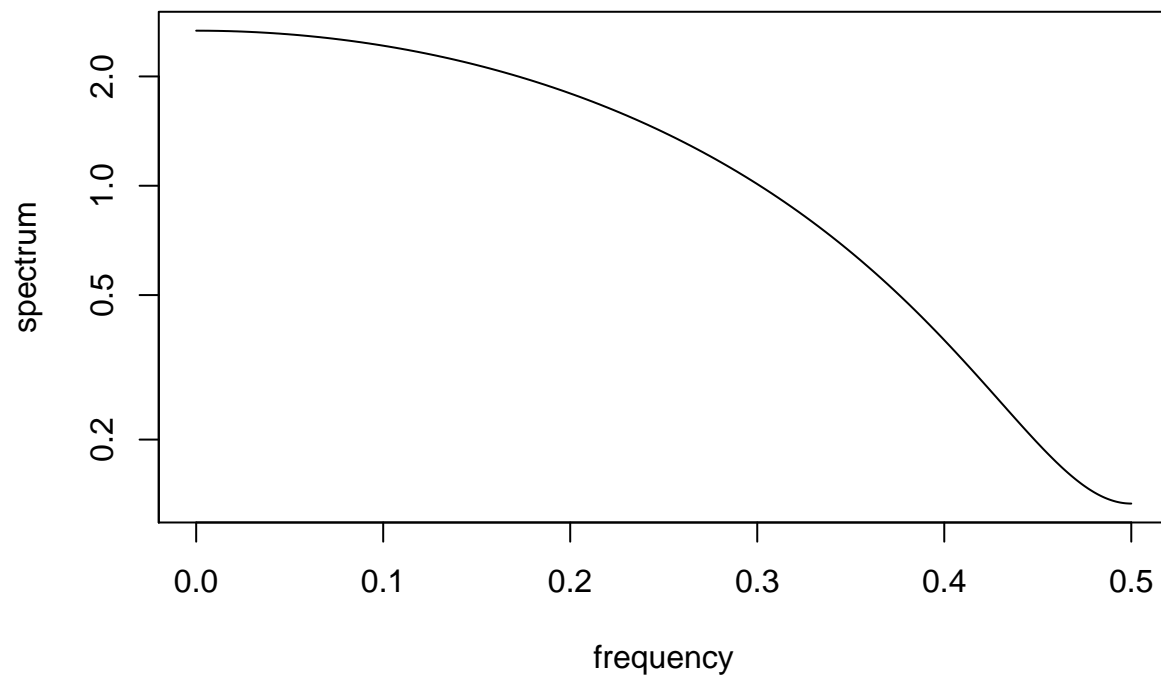
Trying ARMA(0,1)

```
arma01 <- arima(chemneg1, order = c(0, 0, 1), method = c("ML"))
arma01

##
## Call:
## arima(x = chemneg1, order = c(0, 0, 1), method = c("ML"))
##
## Coefficients:
##          ma1  intercept
##          0.6349    0.9934
## s.e.  0.0821    0.0001
##
## sigma^2 estimated as 2.36e-07:  log likelihood = 558.72,  aic = -1113.44
0.6349+1.96*0.0821

## [1] 0.795816
0.6349-1.96*0.0821

## [1] 0.473984
spec.arma(ma=c(0.6349))
```

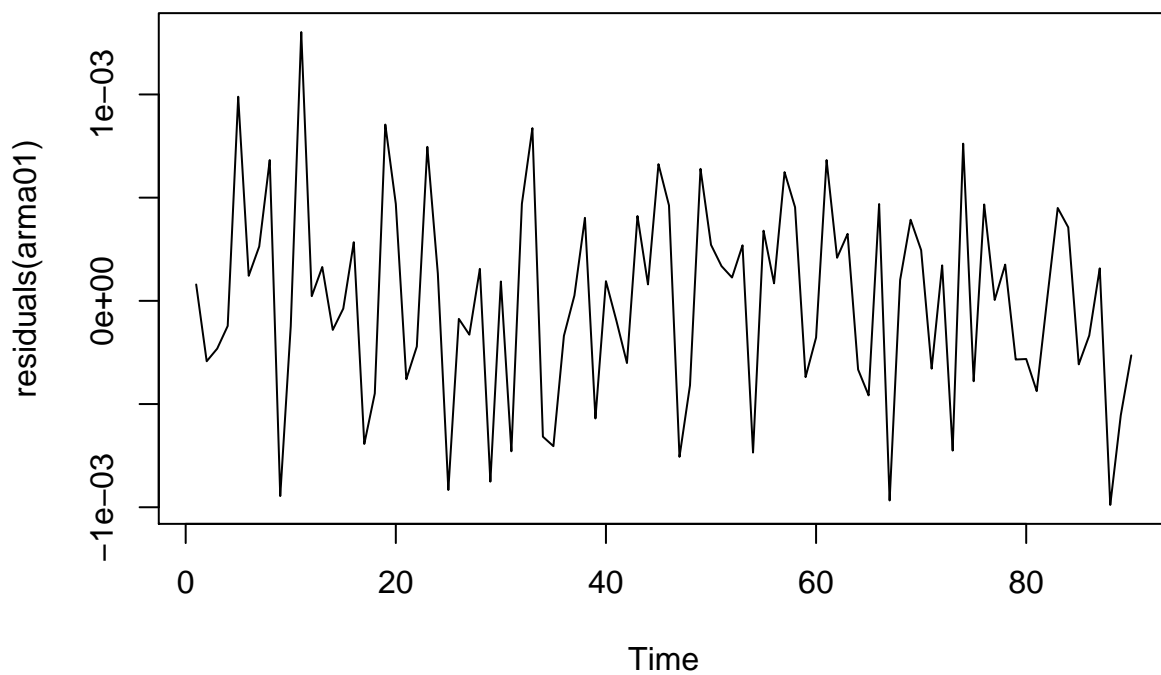


```
mean(residuals(arma01))
```

```
## [1] -9.866494e-07
```

```
plot(residuals(arma01), main = "Residuals of ARMA(0,2) Model")
```

### Residuals of ARMA(0,2) Model



```
Box.test(residuals(arma01), type = "Ljung-Box", lag = sqrt(90), fitdf = 2)
```

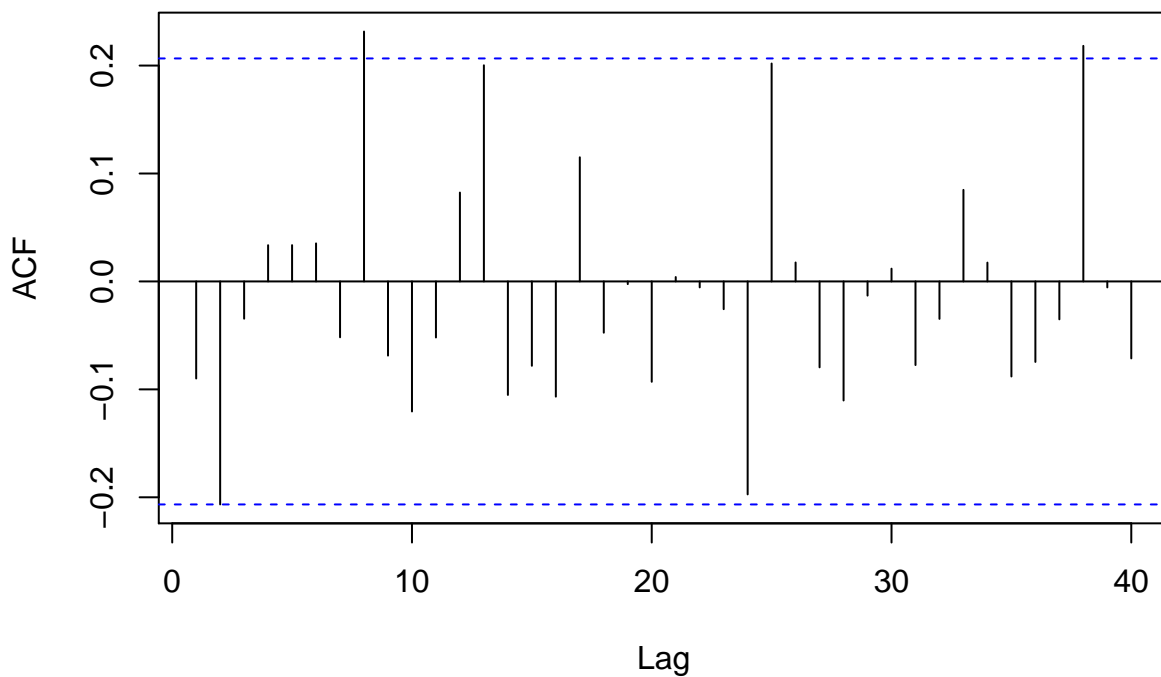
```
##
```

```
## Box-Ljung test
##
## data: residuals(arma01)
## X-squared = 11.386, df = 7.4868, p-value = 0.1494
Box.test(residuals(arma01), type = "Box-Pierce", lag = sqrt(90), fitdf = 2)

##
## Box-Pierce test
##
## data: residuals(arma01)
## X-squared = 10.482, df = 7.4868, p-value = 0.1954
Box.test(residuals(arma01)^2, type = "Ljung", lag = sqrt(90), fitdf = 0)

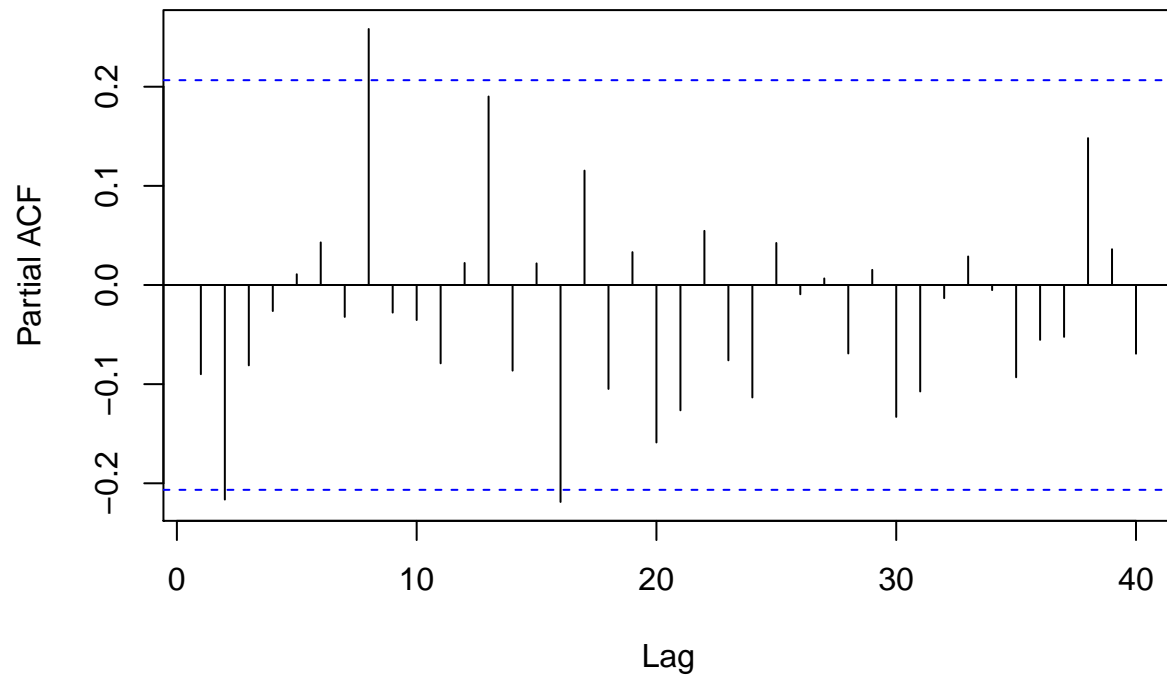
##
## Box-Ljung test
##
## data: residuals(arma01)^2
## X-squared = 22.191, df = 9.4868, p-value = 0.01084
acf(residuals(arma01), main = "Autocorrelation", lag.max = 40)
```

## Autocorrelation



```
pacf(residuals(arma01), main = "Partial Autocorrelation", lag.max = 40)
```

## Partial Autocorrelation

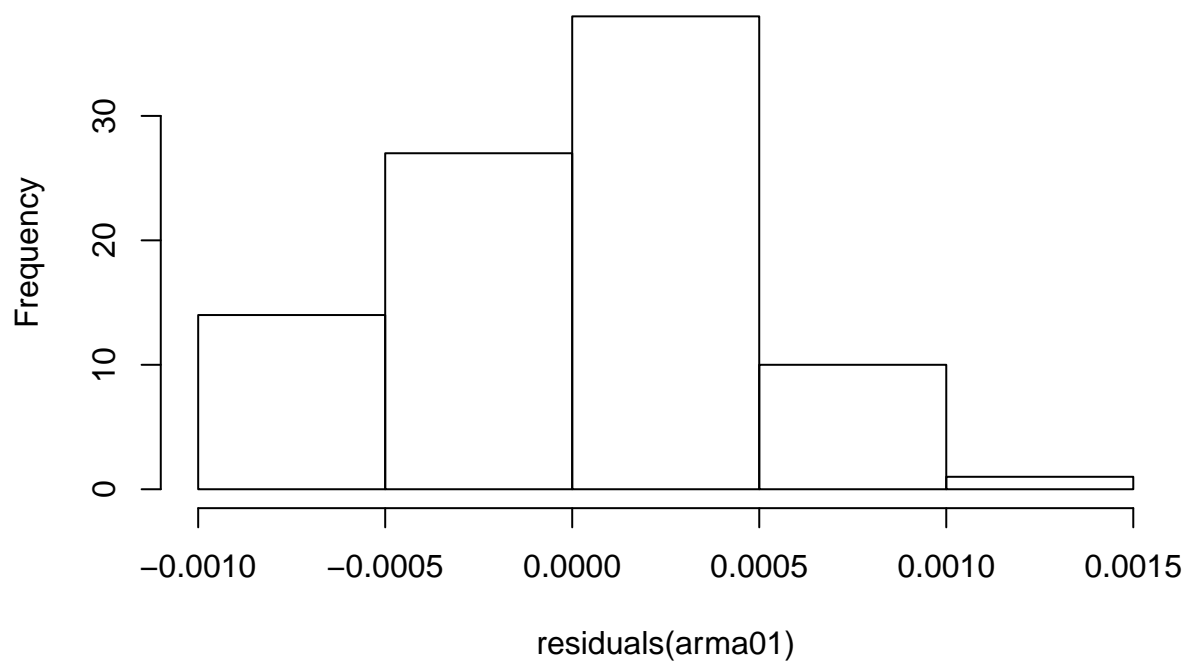


```
residuals(arma01)
```

```
## Time Series:
## Start = 1
## End = 90
## Frequency = 1
## [1] 7.944864e-05 -2.925984e-04 -2.315276e-04 -1.214927e-04 9.891227e-04
## [6] 1.215317e-04 2.635482e-04 6.823205e-04 -9.458559e-04 -1.223766e-04
## [11] 1.302333e-03 2.283114e-05 1.639485e-04 -1.407044e-04 -3.736142e-05
## [16] 2.843530e-04 -6.933436e-04 -4.485139e-04 8.542271e-04 4.697053e-04
## [21] -3.795819e-04 -2.211325e-04 7.463653e-04 1.320712e-04 -9.164848e-04
## [26] -8.730786e-05 -1.638227e-04 1.551266e-04 -8.758442e-04 9.395967e-05
## [31] -7.288701e-04 4.703233e-04 8.369297e-04 -6.580912e-04 -7.040378e-04
## [36] -1.693037e-04 2.614462e-05 4.022777e-04 -5.698051e-04 9.529289e-05
## [41] -9.711317e-05 -3.012890e-04 4.110991e-04 7.973025e-05 6.622749e-04
## [46] 4.624327e-04 -7.557535e-04 -4.088880e-04 6.396734e-04 2.715286e-04
## [51] 1.683478e-04 1.129122e-04 2.690582e-04 -7.350201e-04 3.399879e-04
## [56] 8.507516e-05 6.236601e-04 4.537301e-04 -3.694387e-04 -1.776277e-04
## [61] 6.822331e-04 2.088632e-04 3.246047e-04 -3.328004e-04 -4.579070e-04
## [66] 4.691842e-04 -9.671117e-04 1.012496e-04 3.929319e-04 2.456040e-04
## [71] -3.286048e-04 1.720325e-04 -7.255476e-04 7.616164e-04 -3.894635e-04
## [76] 4.670838e-04 4.378088e-06 1.756652e-04 -2.841984e-04 -2.816944e-04
## [81] -4.374625e-04 1.126454e-05 4.500662e-04 3.562731e-04 -3.075602e-04
## [86] -1.676697e-04 1.575691e-04 -9.887846e-04 -5.546324e-04 -2.641658e-04
```

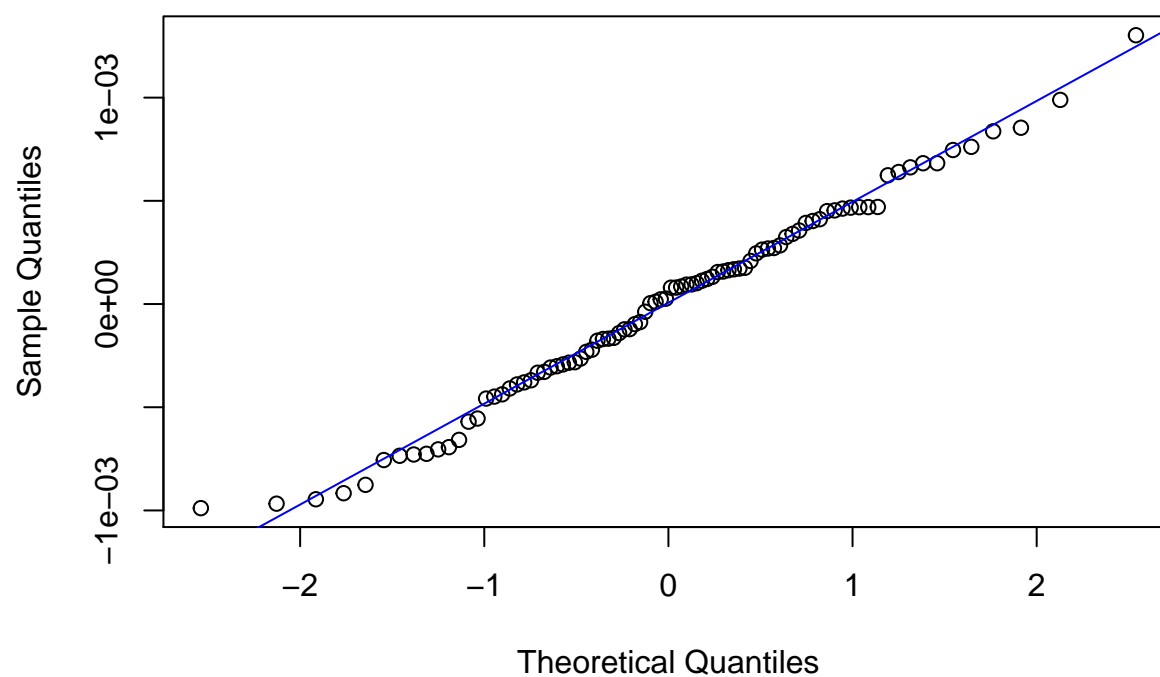
```
hist(residuals(arma01),main = "Histogram")
```

### Histogram



```
qqnorm(residuals(arma01))  
qqline(residuals(arma01),col ="blue")
```

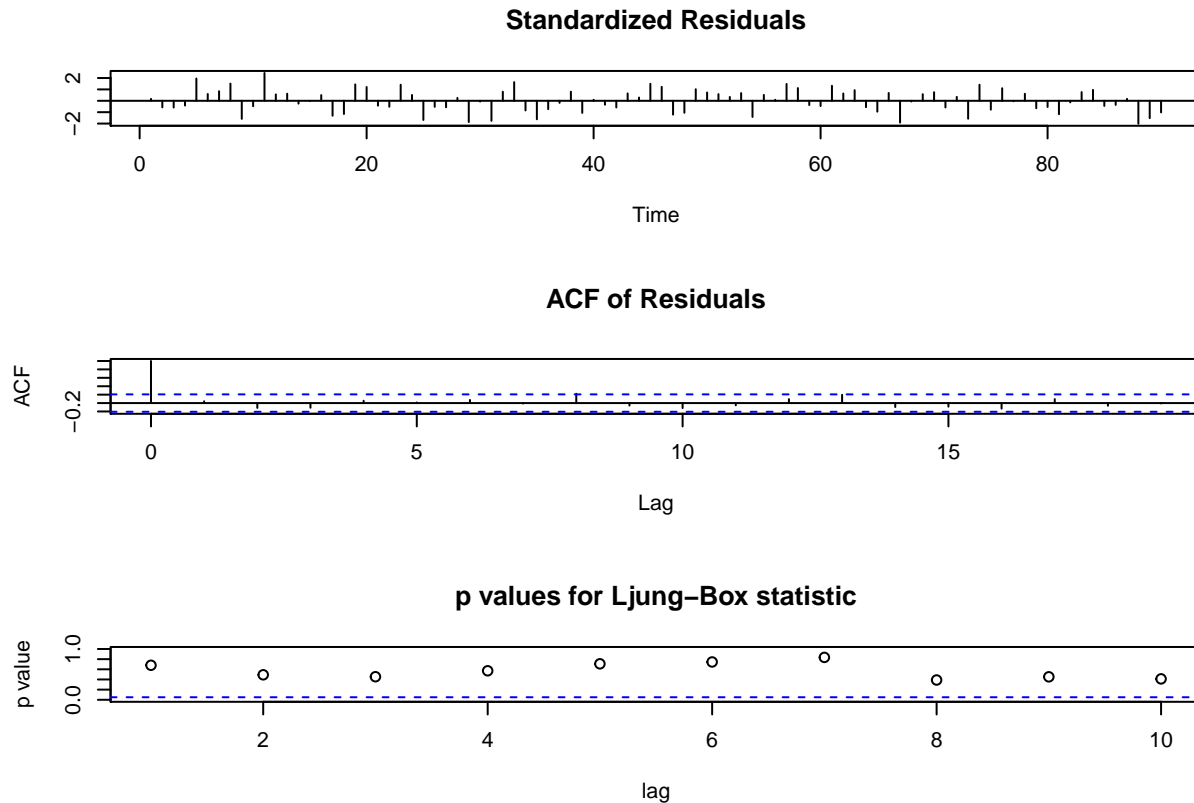
### Normal Q-Q Plot



```
shapiro.test(residuals(arma01))
```

```
##
```

```
## Shapiro-Wilk normality test
##
## data: residuals(arma01)
## W = 0.98925, p-value = 0.676
tsdiag(arima(chemneg1, order=c(0,0,2)))
```

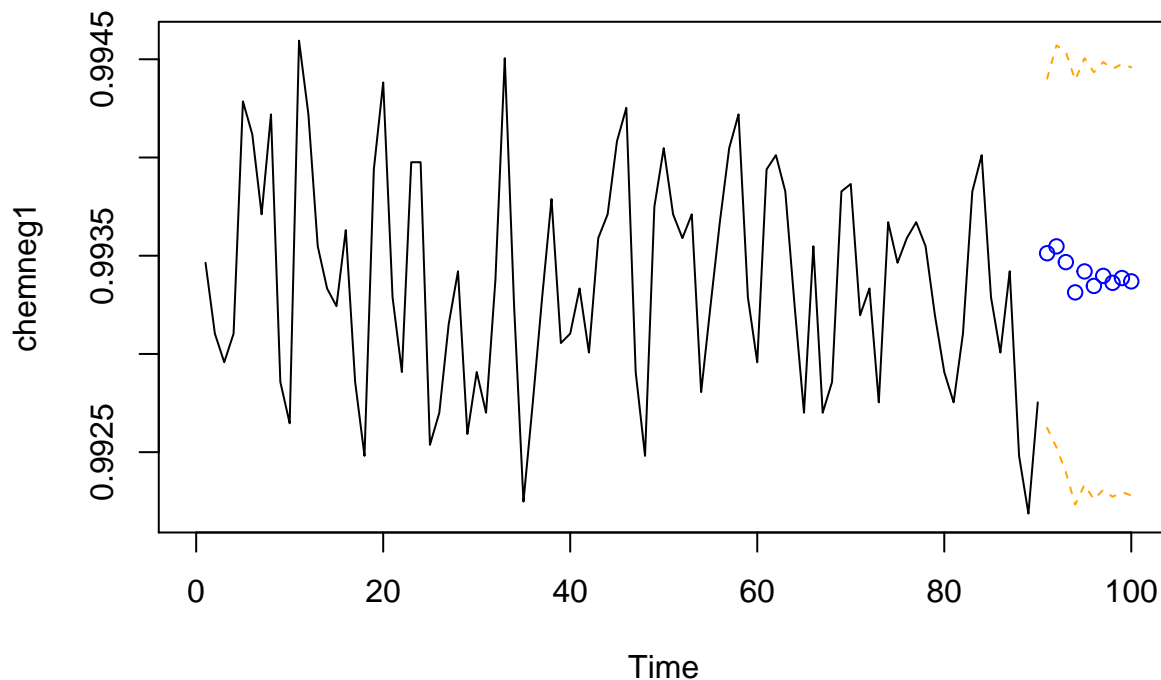


Predicting 10 Steps Ahead using ARMA(1,3) on transformed data

```
pred = predict(arma13, n.ahead=10)
pred$pred
```

```
## Time Series:
## Start = 91
## End = 100
## Frequency = 1
## [1] 0.9935127 0.9935476 0.9934674 0.9933135 0.9934200 0.9933463 0.9933973
## [8] 0.9933620 0.9933864 0.9933695
```

```
ts.plot(chemneg1,xlim=c(0,100))
points(91:100,pred$pred, col = "blue")
lines(91:100,pred$pred-1.96*pred$se,lty=2, col = "orange")
lines(91:100,pred$pred+1.96*pred$se,lty=2, col = "orange")
```



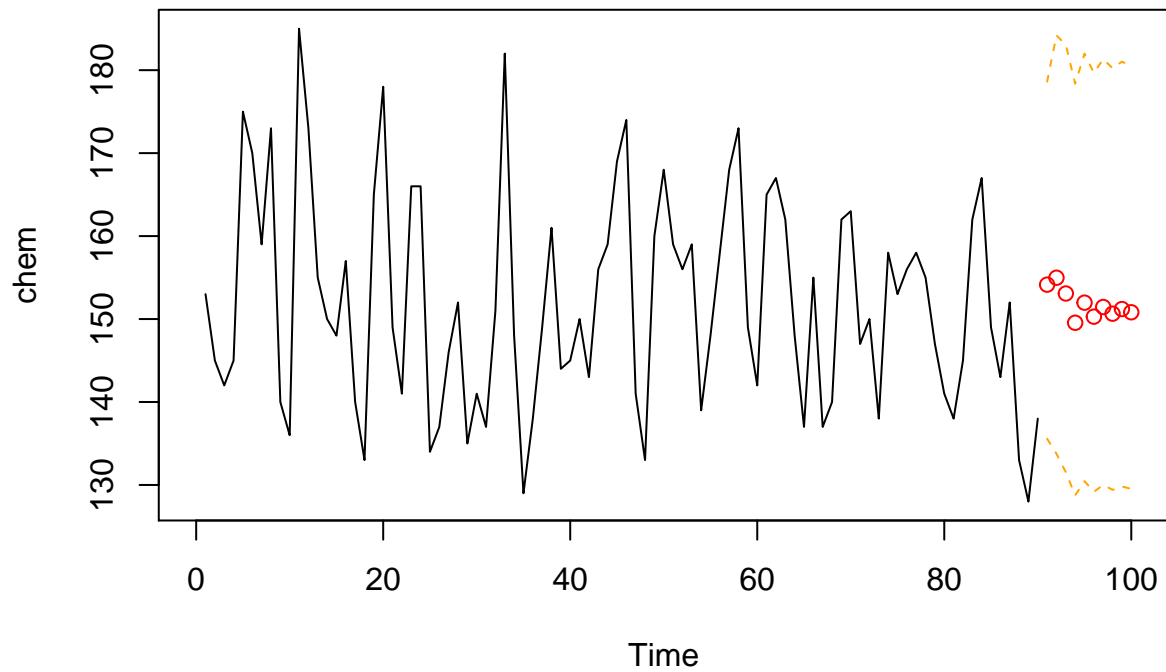
Predicting 10 Steps Ahead using ARMA(1,3) on raw data

```
pred = predict(arma13, n.ahead=10)
pred_raw <- -1/(pred$pred-1)
pred_raw
```

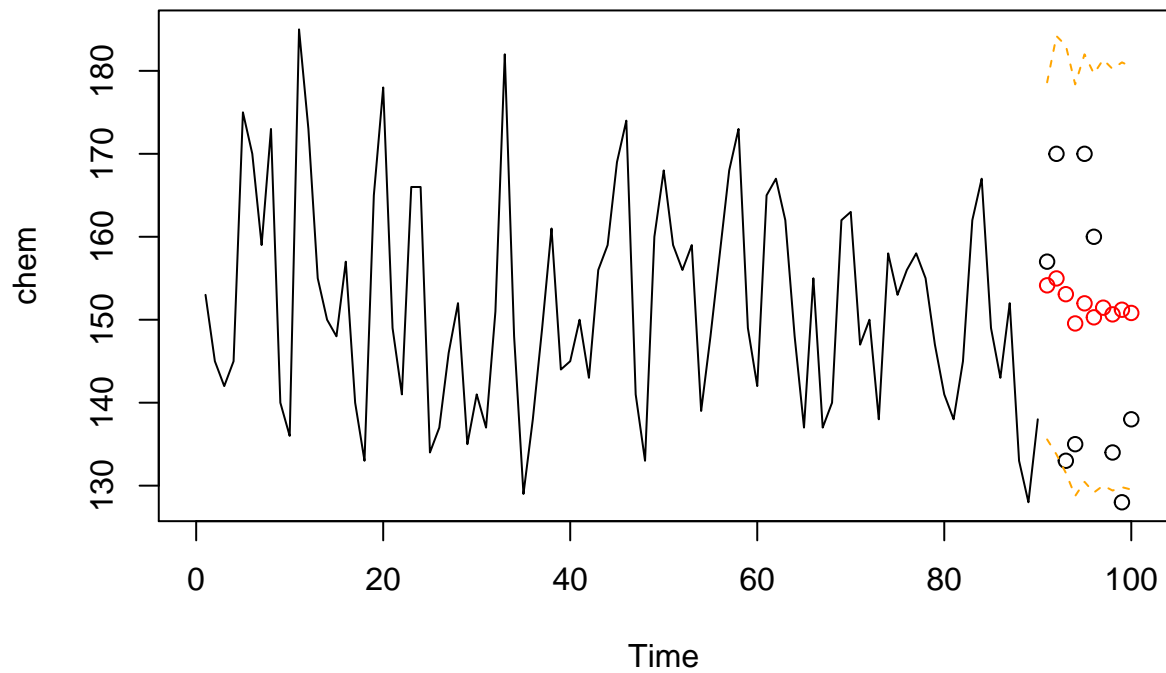
```
## Time Series:
## Start = 91
## End = 100
## Frequency = 1
## [1] 154.1468 154.9809 153.0791 149.5540 151.9765 150.2914 151.4539
## [8] 150.6473 151.2047 150.8184
```

```
utran <- pred$pred+1.96*pred$se
ltran <- pred$pred-1.96*pred$se
u <- -1/(utran-1)
l <- -1/(ltran-1)
ts.plot(chem,xlim=c(0,100))
points(91:100,pred_raw, col = "red")
lines(91:100,u,lty=2, col = "orange")
lines(91:100,l,lty=2, col = "orange")
```

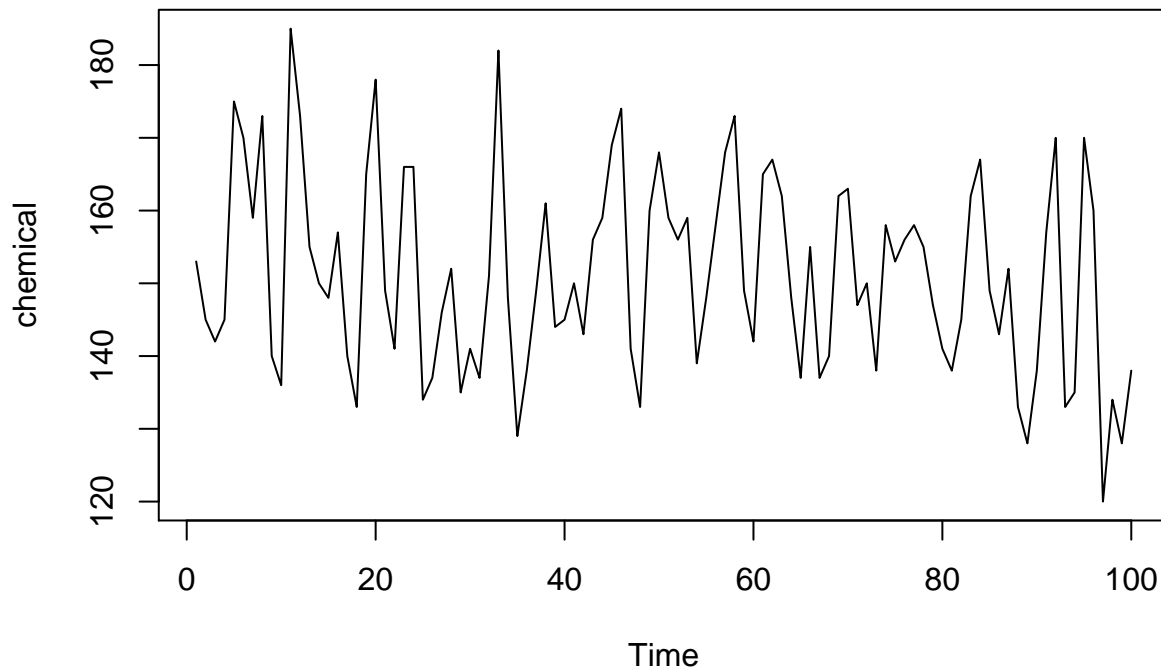




```
ts.plot(chem,xlim=c(0,100))
points(91:100,pred_raw, col = "red")
points(91:100,actualvals, col = "black")
lines(91:100,u,lty=2, col = "orange")
lines(91:100,l,lty=2, col = "orange")
```

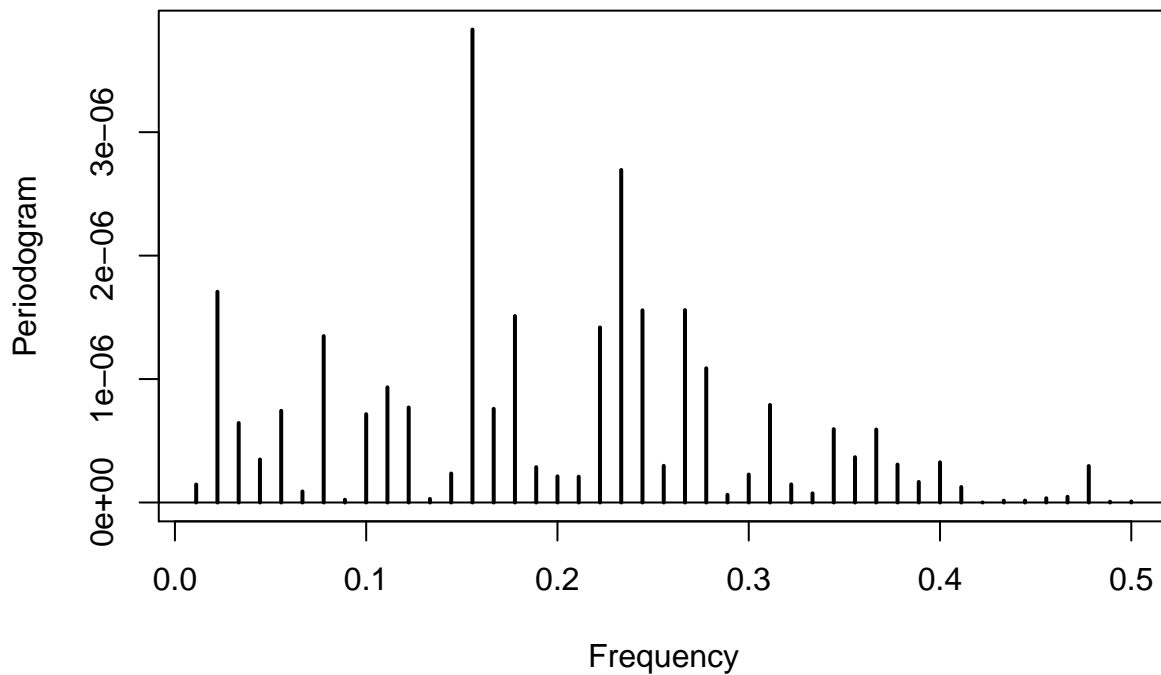


```
ts.plot(chemical)
```



### Spectral Analysis

```
periods = periodogram(chemneg1); abline(h=0)
```



```
periods
```

```
## $freq
## [1] 0.01111111 0.02222222 0.03333333 0.04444444 0.05555556 0.06666667
## [7] 0.07777778 0.08888889 0.10000000 0.11111111 0.12222222 0.13333333
## [13] 0.14444444 0.15555556 0.16666667 0.17777778 0.18888889 0.20000000
## [19] 0.21111111 0.22222222 0.23333333 0.24444444 0.25555556 0.26666667
```

```

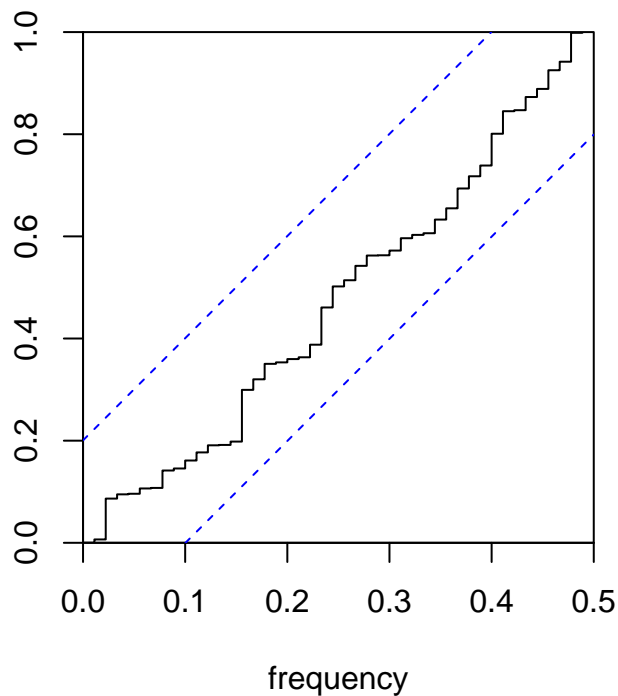
## [25] 0.27777778 0.28888889 0.30000000 0.31111111 0.32222222 0.33333333
## [31] 0.34444444 0.35555556 0.36666667 0.37777778 0.38888889 0.40000000
## [37] 0.41111111 0.42222222 0.43333333 0.44444444 0.45555556 0.46666667
## [43] 0.47777778 0.48888889 0.50000000
##
## $spec
## [1] 1.459251e-07 1.706957e-06 6.441498e-07 3.486891e-07 7.424526e-07
## [6] 8.977693e-08 1.348333e-06 2.278619e-08 7.158397e-07 9.330125e-07
## [11] 7.699174e-07 2.931919e-08 2.350183e-07 3.831406e-06 7.589157e-07
## [16] 1.510466e-06 2.865592e-07 2.121558e-07 2.091720e-07 1.418589e-06
## [21] 2.693945e-06 1.557094e-06 2.971320e-07 1.558660e-06 1.087551e-06
## [26] 6.300177e-08 2.268455e-07 7.901023e-07 1.470304e-07 7.449794e-08
## [31] 5.943117e-07 3.672488e-07 5.911482e-07 3.074308e-07 1.667850e-07
## [36] 3.258188e-07 1.253897e-07 5.920509e-10 1.535186e-08 1.609014e-08
## [41] 3.487233e-08 4.668291e-08 2.956266e-07 6.826844e-09 8.446475e-09
##
## $coh
## NULL
##
## $phase
## NULL
##
## $kernel
## NULL
##
## $df
## [1] 2
##
## $bandwidth
## [1] 0.003207501
##
## $n.used
## [1] 90
##
## $orig.n
## [1] 90
##
## $series
## [1] "x"
##
## $snames
## NULL
##
## $method
## [1] "Raw Periodogram"
##
## $taper
## [1] 0
##
## $pad
## [1] 0
##
## $detrend
## [1] FALSE

```

```
##
## $demean
## [1] TRUE
##
## attr(,"class")
## [1] "spec"
```

```
cpgram(residuals(arma13))
```

### Series: residuals(arma13)



```
FREQ = periods$freq[order(periods$spec, decreasing=T)][1]
FREQ[1]
```

```
## [1] 0.1555556
```

```
library(GeneCycle)
```

```
## Warning: package 'GeneCycle' was built under R version 3.5.2
```

```
## Loading required package: longitudinal
```

```
## Loading required package: corpcor
```

```
## Loading required package: fdrtool
```

```
##
```

```
## Attaching package: 'GeneCycle'
```

```
## The following object is masked from 'package:TSA':
```

```
##
```

```
## periodogram
```

```
## The following object is masked from 'package:forecast':
```

```
##
```

```
## is.constant
```

```
fisher.g.test(residuals(arma13))
```

```
## [1] 0.6914845
```