R Notebook

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

What does Ljung box-statistic do; should they be significant or not? Should residuals look normal? Are the * in eacf mean its significant? What should ACF look like for stationarity?

```
library(TSA)
```

```
## Loading required package: leaps
## Loading required package: locfit
## locfit 1.5-9.1    2013-03-22
## Loading required package: mgcv
## Loading required package: nlme
## This is mgcv 1.8-22. For overview type 'help("mgcv-package")'.
## Loading required package: tseries
##
## Attaching package: 'TSA'
## The following objects are masked from 'package:stats':
##
##     acf, arima
## The following object is masked from 'package:utils':
##
##     tar
```

```
library(tseries)
library(astsa)
library(imputeTS)
```

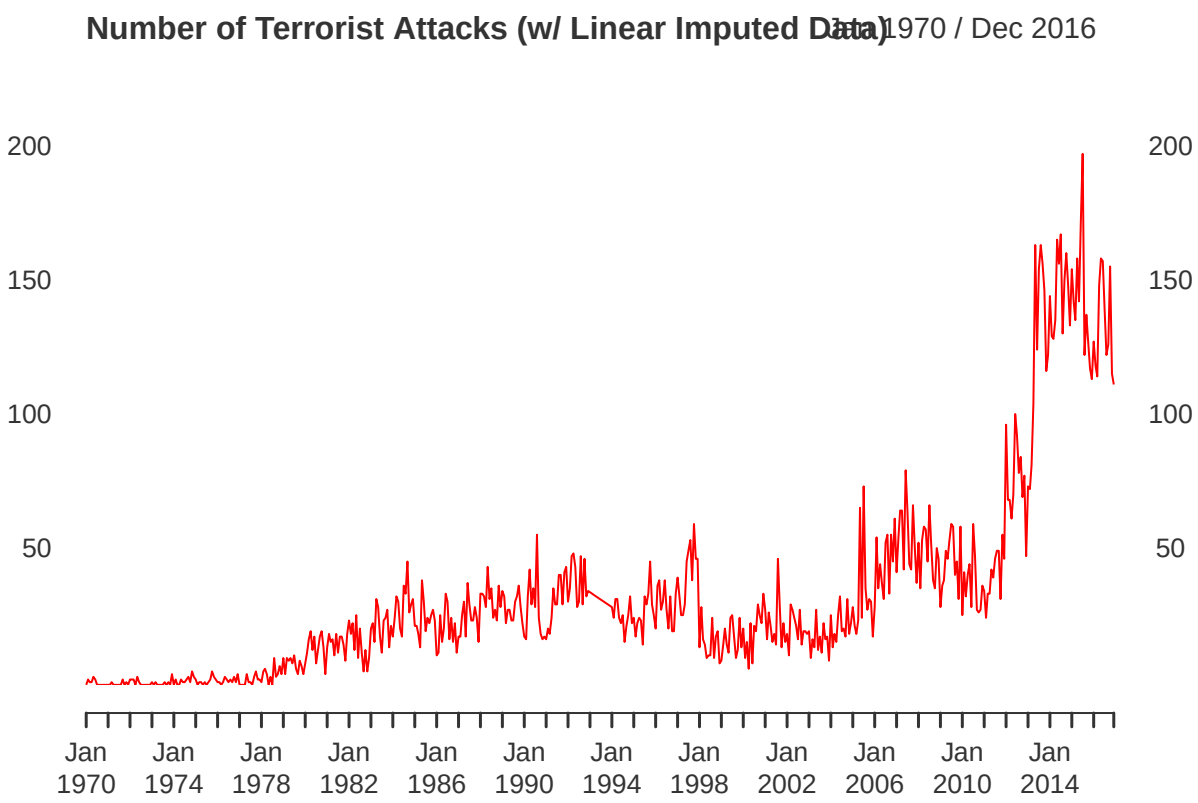
```
##
## Attaching package: 'imputeTS'
## The following object is masked from 'package:tseries':
##
##     na.remove
```

```
library(tsoutliers)
library(xts)
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following object is masked from 'package:imputeTS':
##
##     na.locf
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

```
terror2 <- read.csv("input/og_num_casualities_greater_than_10.csv")
terror3 <- na.interpolation(terror2$num.attacks.with.kill.thresh, option="linear")
plot(as.xts(ts(terror3, frequency = 12, start=1970)), main = "Number of Terrorist Attacks (w/ Linear Imp
```




```
pdf("image/og_ts.pdf")
lines(as.xts(ts(terror2$num.attacks.with.kill.thresh, frequency = 12, start=1970)), col="black")
dev.off()
```

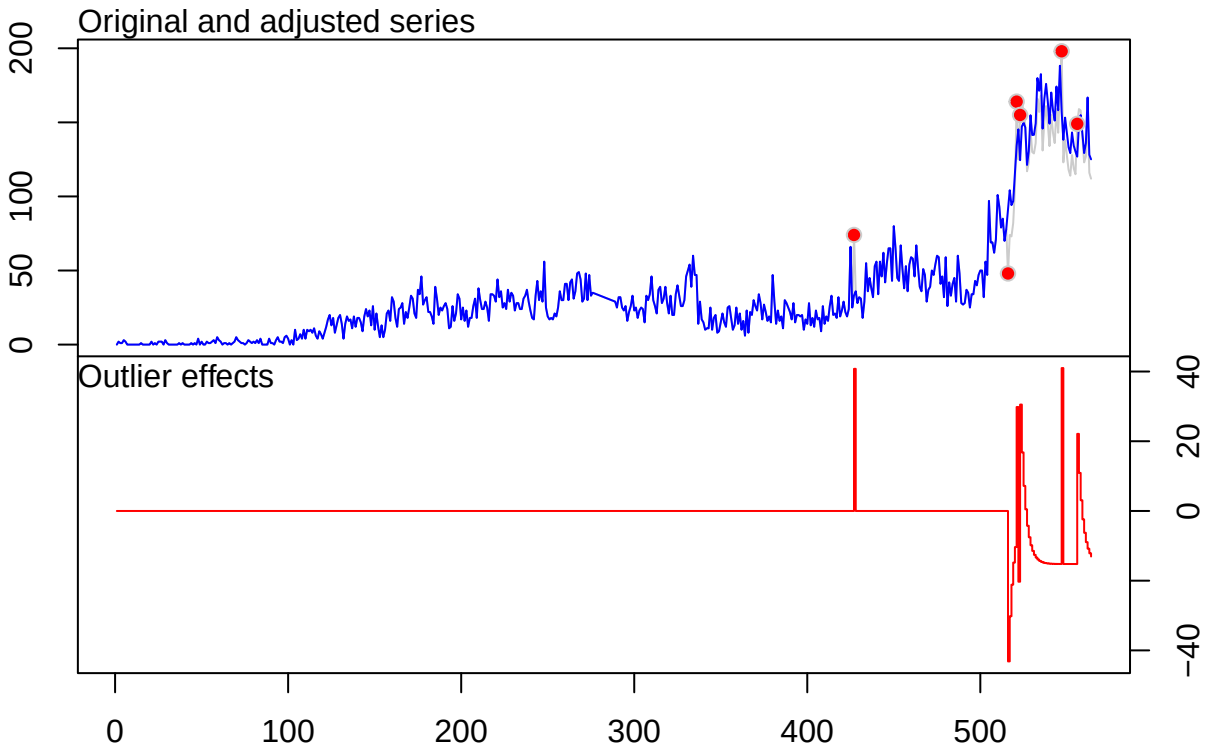
```
## pdf
##   2
```

```
#pdf("image/imputed_ts.pdf")
#plot(as.xts(ts(terror3, frequency = 12, start=1970)), main = "Number of Terrorist Attacks (w/ Linear I
#dev.off()


outlier_terror3 <- tso(ts(terror3), types = c("TC", "AO", "IO"))
```

```
## Warning in locate.outliers.oloop(y = y, fit = fit, types = types, cval =
## cval, : stopped when 'maxit.oloop = 4' was reached
```

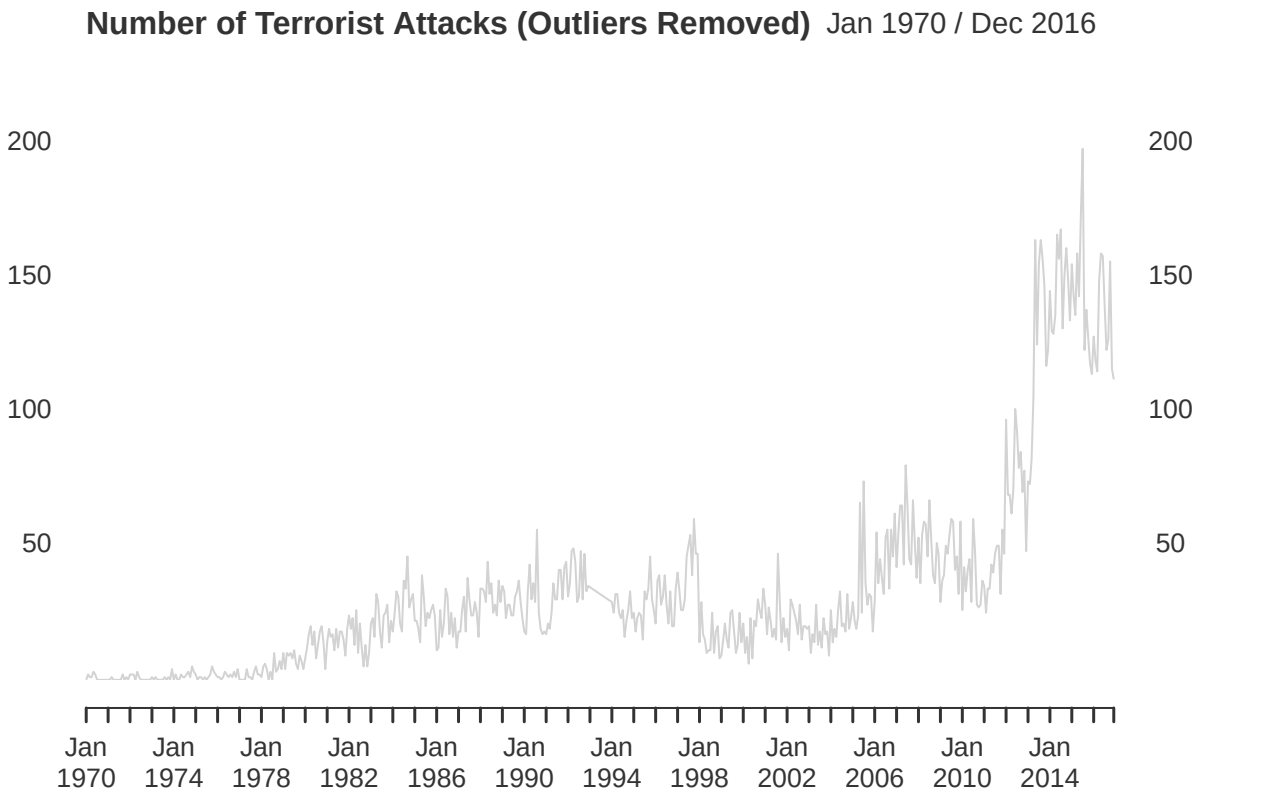
```
plot(outlier_terror3)
```



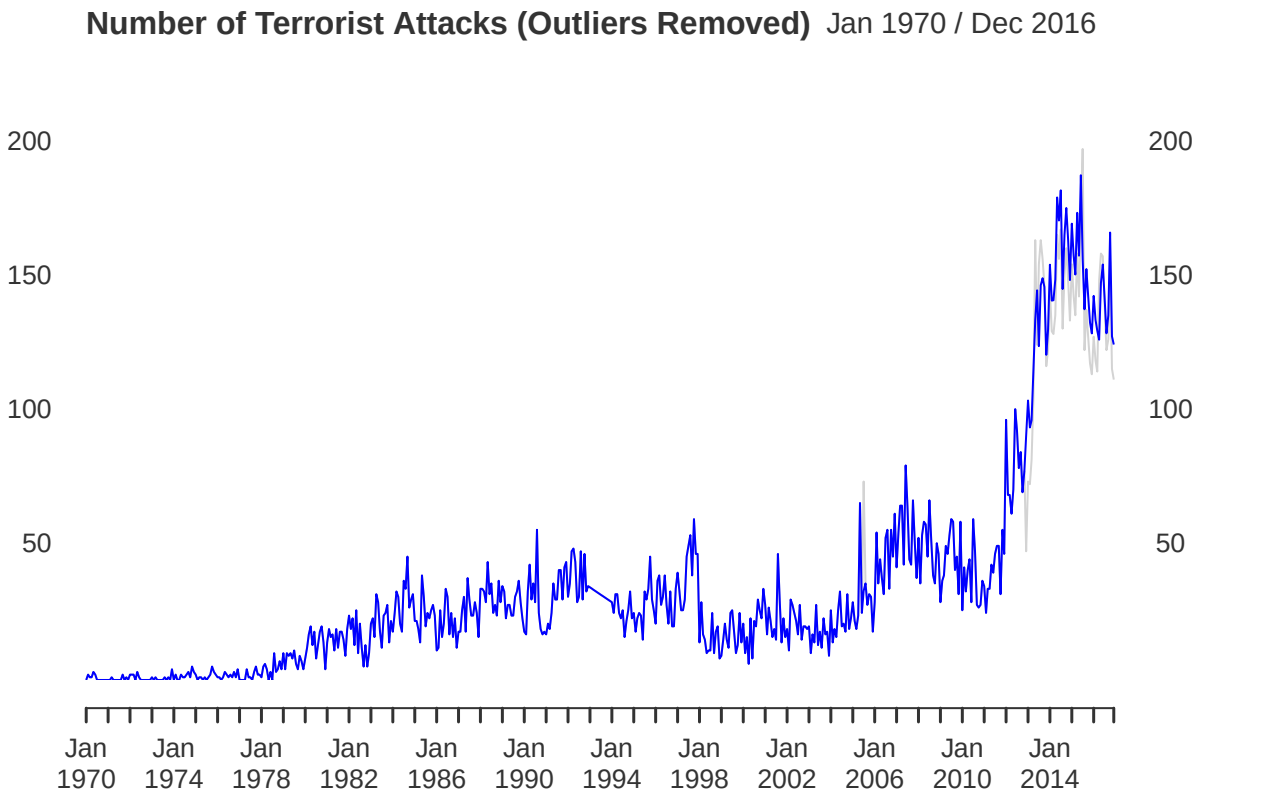

```
#plot outlier effects
pdf("image/outlier_effects.pdf")
plot(as.xts(ts(outlier_terror3$effects, frequency = 12, start=1970)), main = "Outlier Effects", major.f
dev.off()
```

```
## pdf
##   2
```

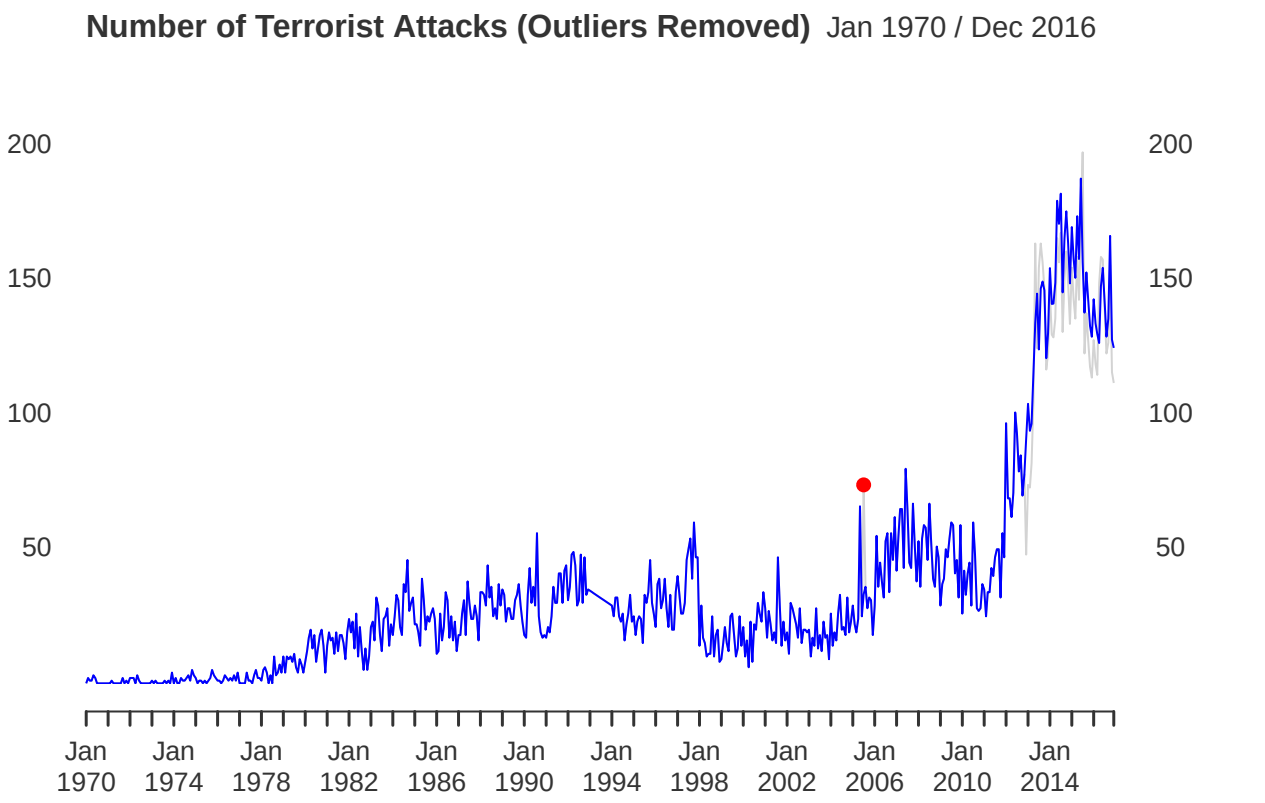
```
#Plot outlier time series
xts.terror3 <- as.xts(ts(terror3, frequency = 12, start=1970))
plot(as.xts(ts(terror3, frequency = 12, start=1970)), main = "Number of Terrorist Attacks (Outliers Rem
```



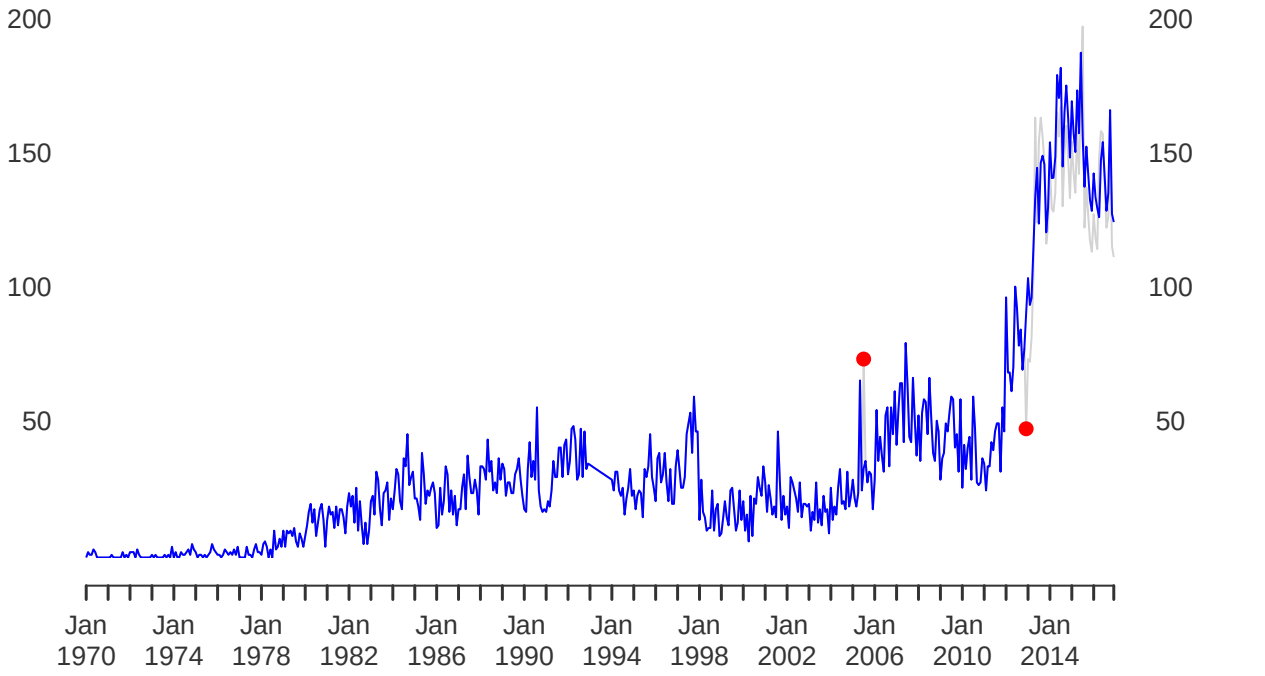

```
lines(as.xts(ts(outlier_terror3$yadj, frequency = 12, start=1970)), col="blue")
```



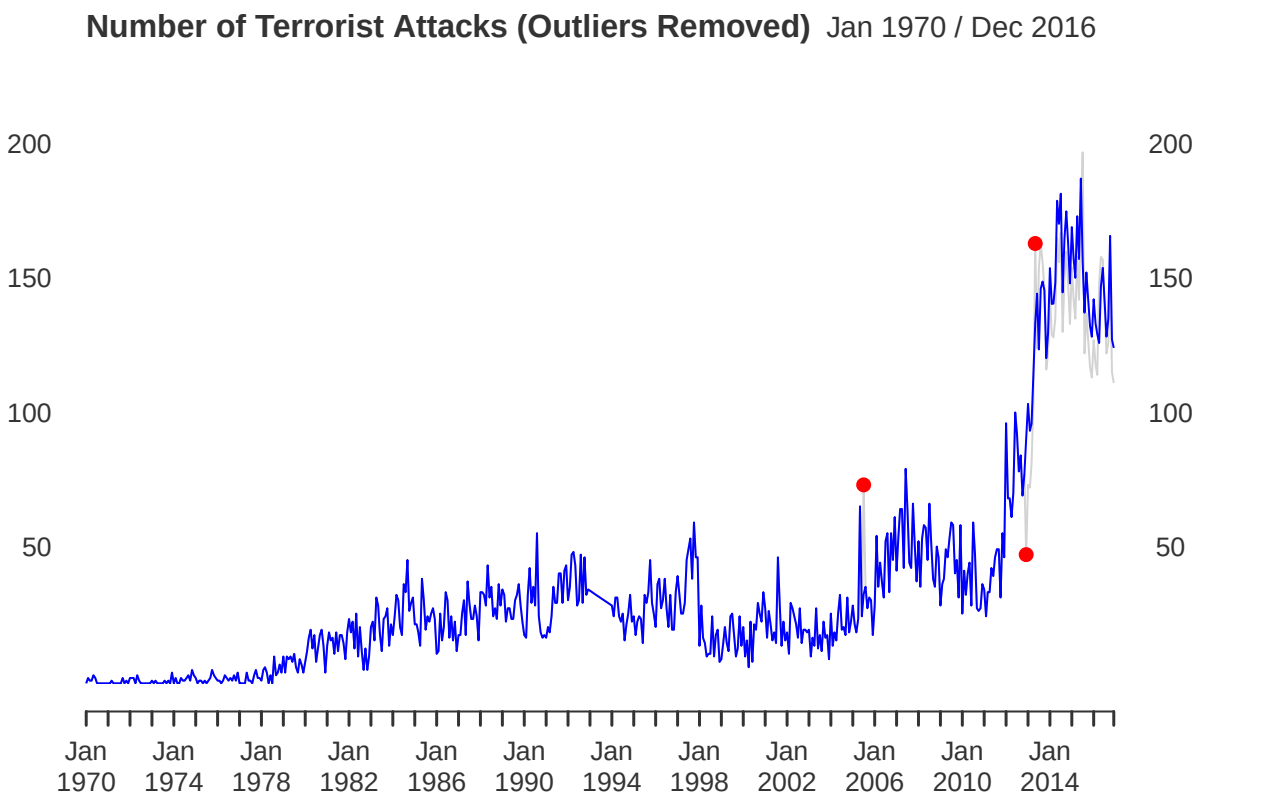
```
points(xts.terror3[427], col="red",pch=19, cex=1)
```



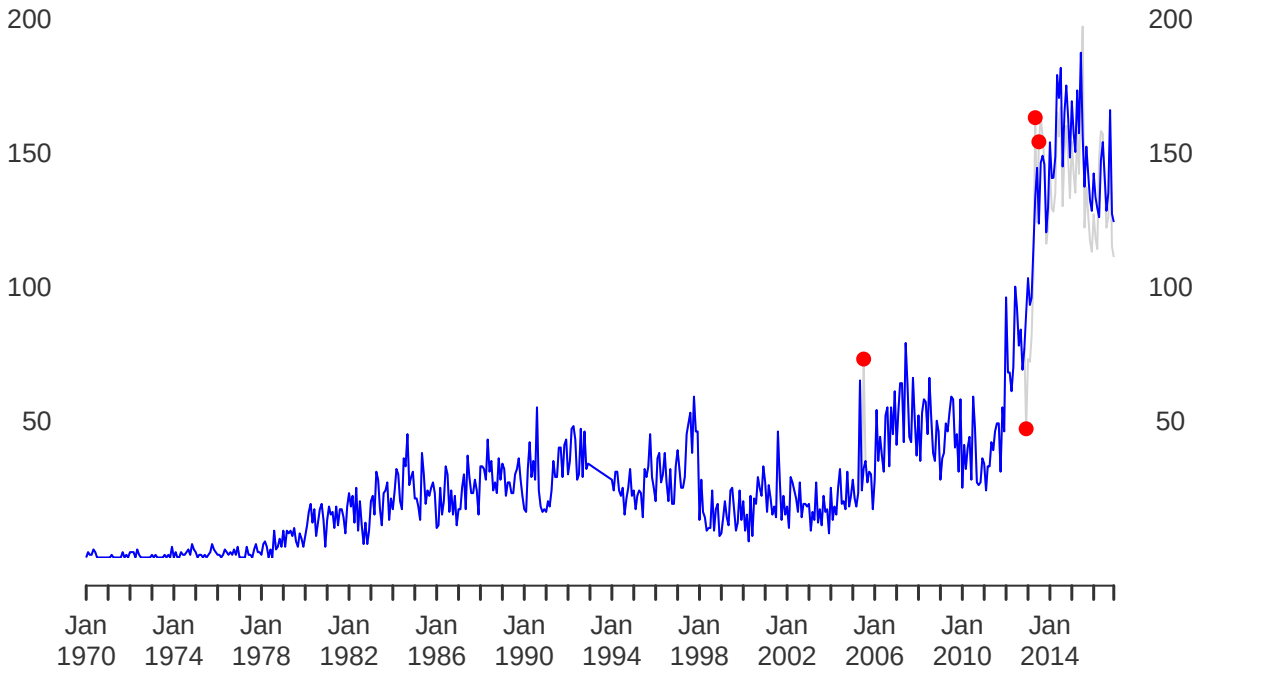

```
points(xts.terror3[516], col="red",pch=19, cex=1)
```



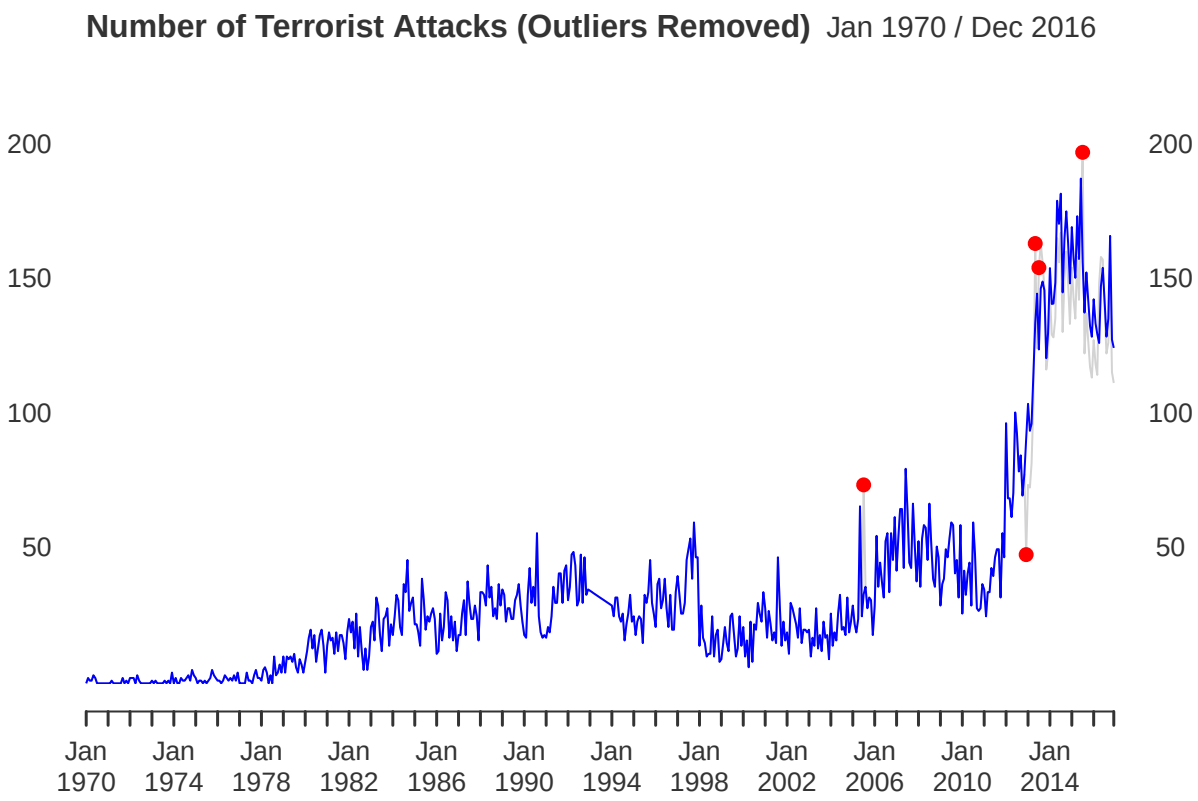
```
points(xts.terror3[521], col="red",pch=19, cex=1)
```




```
points(xts.terror3[523], col="red",pch=19, cex=1)
```



```
points(xts.terror3[547], col="red",pch=19, cex=1)
```




```
pdf("image/outlier_comparison.pdf")
points(xts.terror3[556], col="red",pch=19, cex=1)
dev.off()
```

```
## pdf
##   2
```

```
terror4 <- outlier_terror3$yadj


#terror3 <- na.kalman(terror2$num.attacks, model="auto.arima")
cuttoff.index <- length(terror4) - 48 #floor(0.1 * length(terror3))
cuttoff.index2 <- length(terror4) - 12
terror4.valid <- terror4[(cuttoff.index+1) :cuttoff.index2]
terror4.testing <- terror4[(cuttoff.index2 + 1): length(terror4)]
terror4 <- terror4[1: cuttoff.index]

#plot(as.xts(ts(terror4, frequency = 12, start=1970)), main = "Number of Terrorist Attacks (Training Se

#plot(as.xts(ts(terror4.valid, frequency = 12, start=1970)), main = "Number of Terrorist Attacks (Valid
```

```
#log_terror4 <- log(outlier_terror3$yadj)
adf.test(terror4, k=1)
```

```
## Warning in adf.test(terror4, k = 1): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
```

```
##
## data:  terror4
## Dickey-Fuller = -5.8969, Lag order = 1, p-value = 0.01
## alternative hypothesis: stationary
```

```
adf.test(diff(terror4), k=1)
```

```
## Warning in adf.test(diff(terror4), k = 1): p-value smaller than printed p-
## value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(terror4)
## Dickey-Fuller = -24.495, Lag order = 1, p-value = 0.01
## alternative hypothesis: stationary
```

```
adf.test(diff(diff(terror4)), k=1)
```

```
## Warning in adf.test(diff(diff(terror4)), k = 1): p-value smaller than
## printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(diff(terror4))
## Dickey-Fuller = -32.125, Lag order = 1, p-value = 0.01
## alternative hypothesis: stationary
```

```
pdf("image/first_diff.pdf")
plot(as.xts(ts(diff(terror4), frequency = 12, start=1970)), main = "Number of Terrorist Attacks (First I
dev.off()
```

```
## pdf
##   2
```

```
pdf("image/second_diff.pdf")
plot(as.xts(ts(diff(diff(terror4)), frequency = 12, start=1970)), main = "Number of Terrorist Attacks (S
dev.off()
```

```
## pdf
##   2
```

```
#ts.plot(diff(terror4))
#ts.plot(diff(diff(terror4)))
```

```
m = floor(sqrt(length(diff(terror4))))
pdf("image/raw_periodogram.pdf")
mvspec(diff(terror4), log="no", main="Raw Periodogram")
dev.off()
```

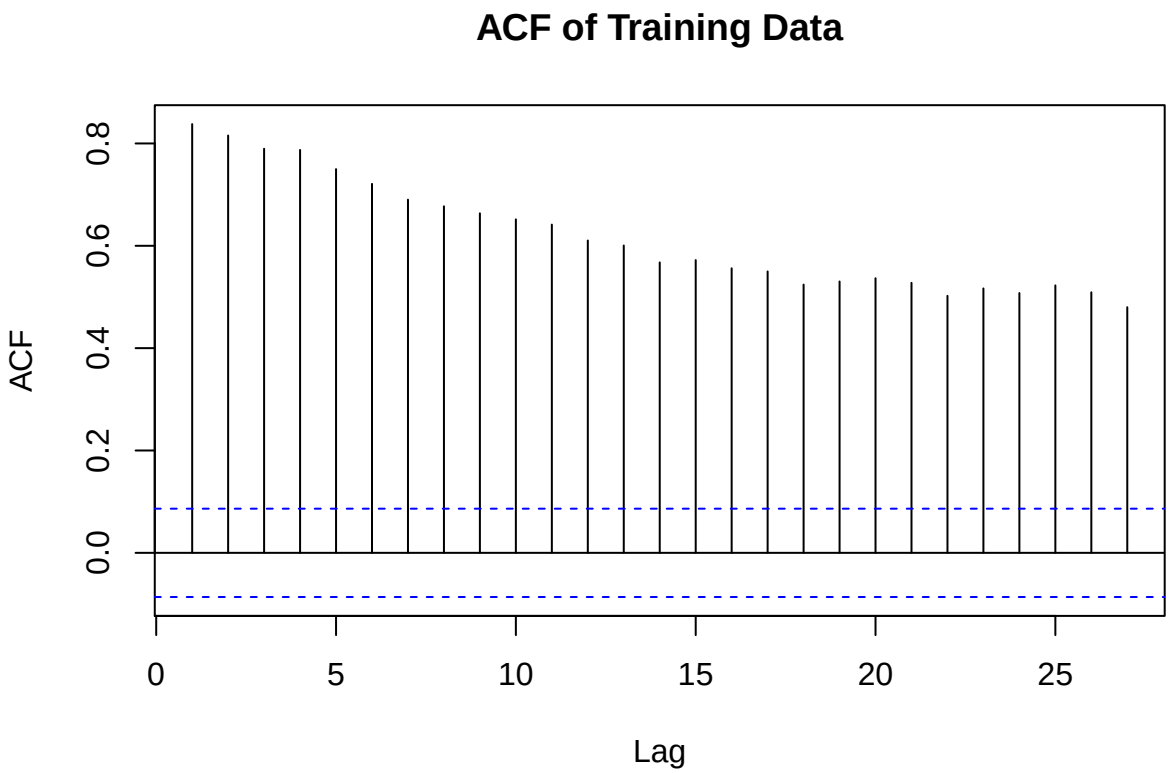
```
## pdf
##   2
```

```
pdf("image/smooth_tapered_periodogram.pdf")
mvspec(diff(terror4), kernel('daniell', m), log="no", taper=0.1, main="Smoothed and Tapered Periodogram
dev.off()
```

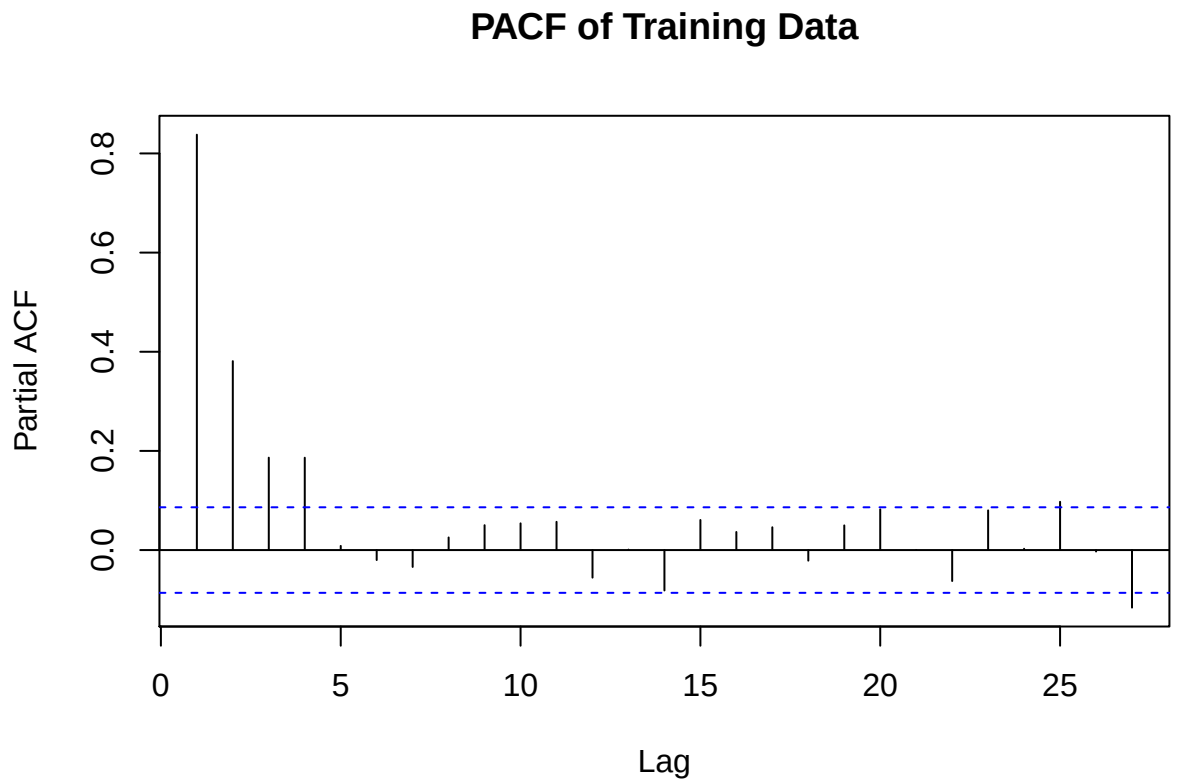
```
## pdf
##   2
```

```r
#muspec(diff(log_terror4),  kernel('daniell', m), log="no")
```

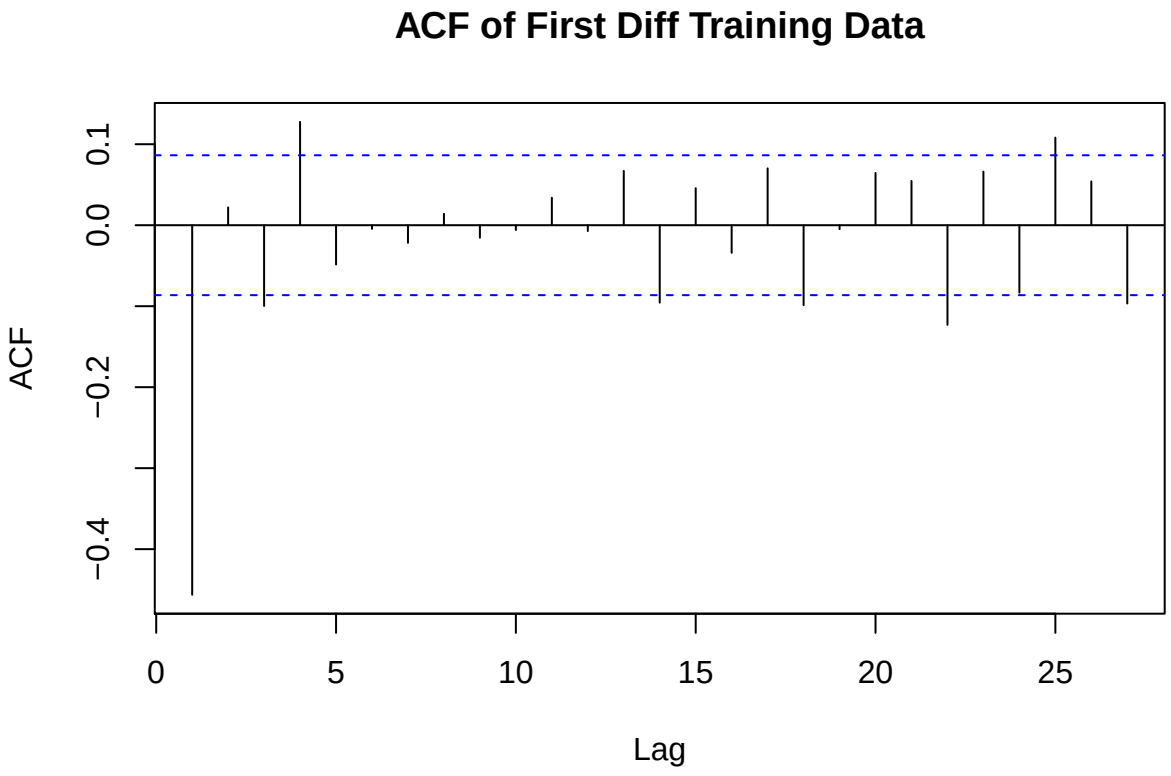
```r
acf(terror4, main="ACF of Training Data")
```



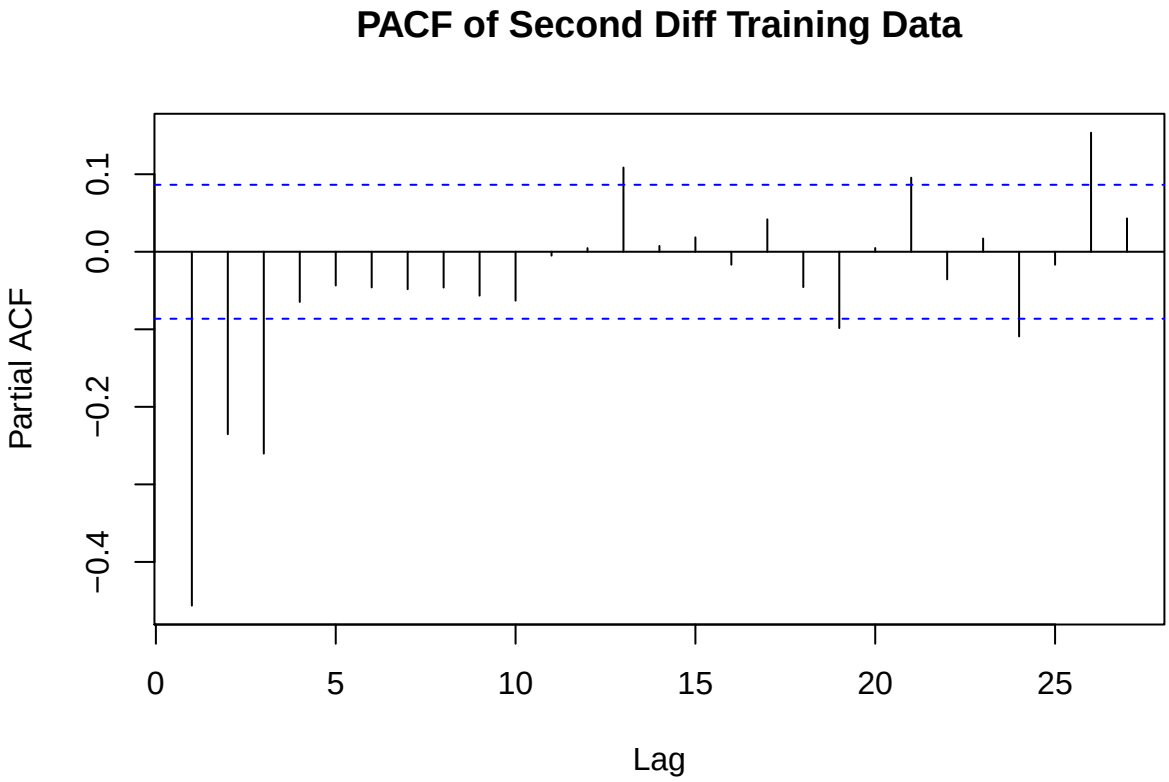

```r
pacf(terror4, main="PACF of Training Data")
```




```
acf(diff(terror4), main="ACF of First Diff Training Data")
```




```
pacf(diff(terror4), main="PACF of Second Diff Training Data")
```




```
eacf(diff(terror4))
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o x x o o o o o o o  o  o  x
## 1 x x o x o o o o o o o  o  o  o
## 2 x x x x o o o o o o o  o  o  o
## 3 x x x x o o o o o o o  o  o  o
## 4 x o o o o o o o o o o  o  o  o
## 5 x x o o o o o o o o o  o  o  o
## 6 x x o o o o o o o o o  o  o  o
## 7 x x o o o o o o o o o  o  o  o
```

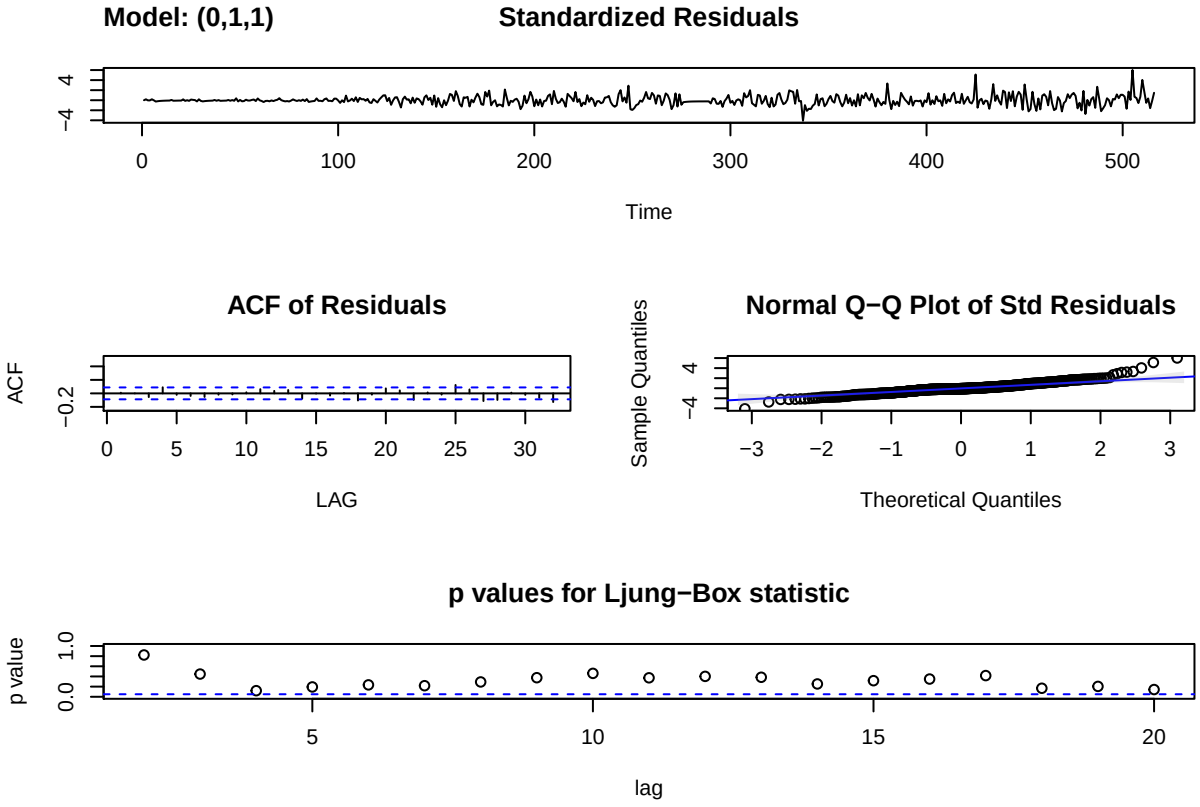
```
eacf(diff(diff(terror4)))
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x o o o o o o o  o  o  x
## 1 x x x x o o o o o o o  o  o  o
## 2 x x x x o o o o o o o  o  o  o
## 3 x o o o o o o o o o o  o  o  o
## 4 x x o o o o o o o o o  o  o  o
## 5 x x o o o o x o o o o  o  o  o
## 6 x x x o o o o o o o o  o  o  o
## 7 x x o o o o o o o o o  o  o  o
```

```
sarima(terror4, 0, 1, 1)
```

```
## initial  value 2.302665
## iter   2 value 2.148680
## iter   3 value 2.132396
```

```
## iter   4 value 2.128997
## iter   5 value 2.121714
## iter   6 value 2.120506
## iter   7 value 2.120430
## iter   8 value 2.120364
## iter   9 value 2.120362
## iter  10 value 2.120362
## iter  10 value 2.120362
## iter  10 value 2.120362
## final  value 2.120362
## converged
## initial  value 2.120944
## iter   2 value 2.120943
## iter   3 value 2.120942
## iter   3 value 2.120942
## iter   3 value 2.120942
## final  value 2.120942
## converged
```



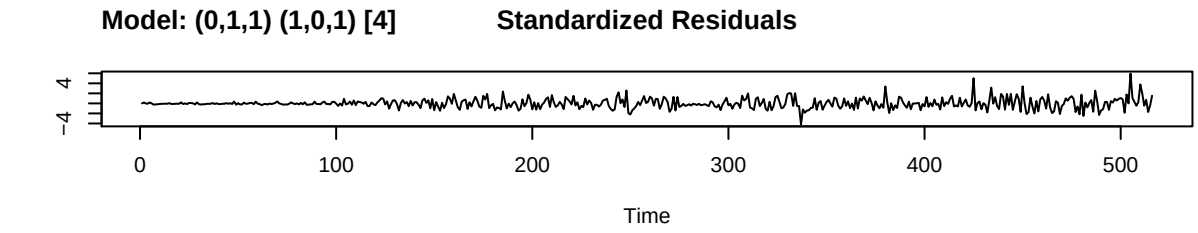

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = constant, optim.control = list(trace = trc, REPORT = 1,
##     reltol = tol))
##
## Coefficients:
##           ma1  constant
##       -0.6746    0.1596
```

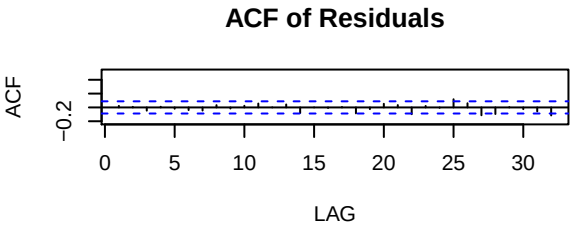
```
## s.e.   0.0336    0.1200
##
## sigma^2 estimated as 69.46:  log likelihood = -1823.04,  aic = 3652.08
##
## $degrees_of_freedom
## [1] 513
##
## $ttable
##          Estimate     SE  t.value p.value
## ma1       -0.6746 0.0336 -20.0780  0.0000
## constant   0.1596 0.1200   1.3306  0.1839
##
## $AIC
## [1] 5.248457
##
## $AICc
## [1] 5.252424
##
## $BIC
## [1] 4.264915
```

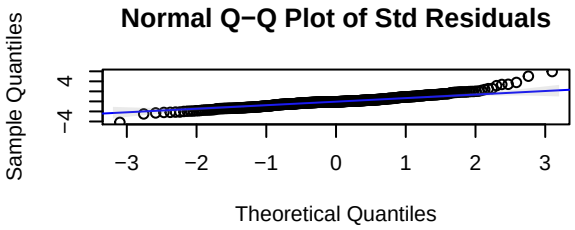
```
sarima(terror4, 0, 1, 1, 1, 0, 1, 4)
```

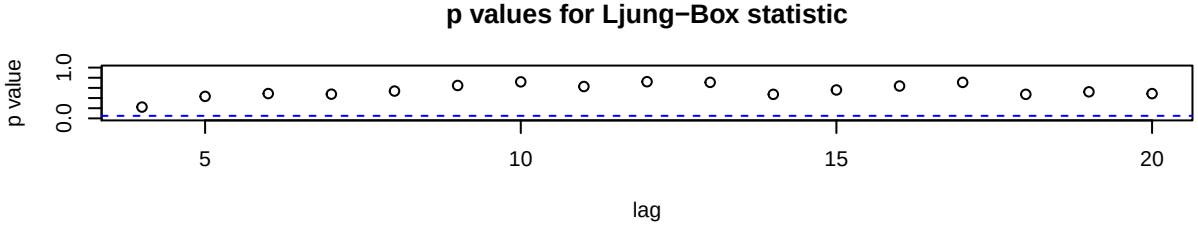
```
## initial  value 2.306485
## iter   2 value 2.160550
## iter   3 value 2.137688
## iter   4 value 2.128502
## iter   5 value 2.123569
## iter   6 value 2.120401
## iter   7 value 2.119951
## iter   8 value 2.119925
## iter   9 value 2.119914
## iter  10 value 2.119906
## iter  11 value 2.119902
## iter  12 value 2.119881
## iter  13 value 2.119531
## iter  14 value 2.119158
## iter  15 value 2.118895
## iter  16 value 2.118789
## iter  17 value 2.118773
## iter  18 value 2.118748
## iter  19 value 2.118211
## iter  20 value 2.116368
## iter  21 value 2.115097
## iter  22 value 2.114878
## iter  23 value 2.114721
## iter  24 value 2.114670
## iter  25 value 2.114631
## iter  26 value 2.114584
## iter  27 value 2.114555
## iter  28 value 2.114550
## iter  29 value 2.114550
## iter  29 value 2.114550
## final  value 2.114550
## converged
```

```
## initial  value 2.111537
## iter   2 value 2.111531
## iter   3 value 2.111529
## iter   4 value 2.111529
## iter   5 value 2.111529
## iter   6 value 2.111529
## iter   7 value 2.111528
## iter   8 value 2.111528
## iter   9 value 2.111528
## iter   9 value 2.111528
## iter   9 value 2.111528
## final  value 2.111528
## converged
```







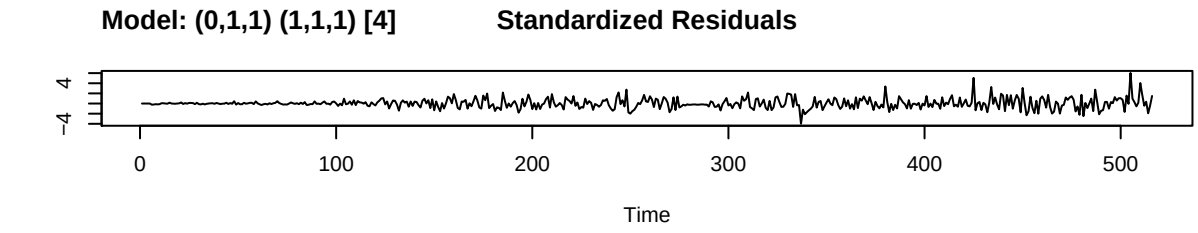


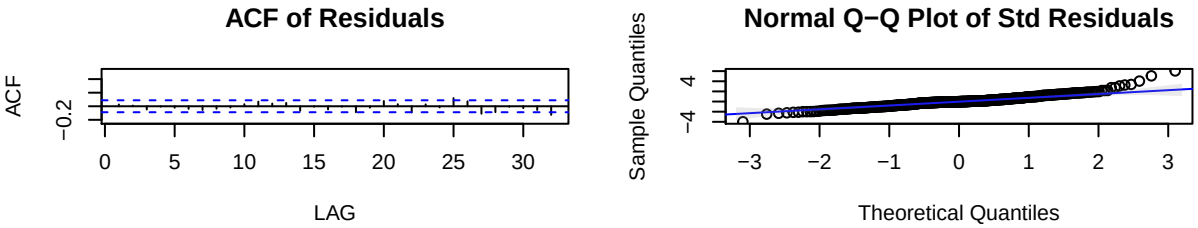

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = constant, optim.control = list(trace = trc, REPORT = 1,
##     reltol = tol))
##
## Coefficients:
##          ma1     sar1    sma1  constant
##       -0.6863  -0.7178  0.8165    0.1595
## s.e.   0.0346   0.0962  0.0774    0.1212
##
## sigma^2 estimated as 68.13:  log likelihood = -1818.19,  aic = 3646.38
##
## $degrees_of_freedom
```

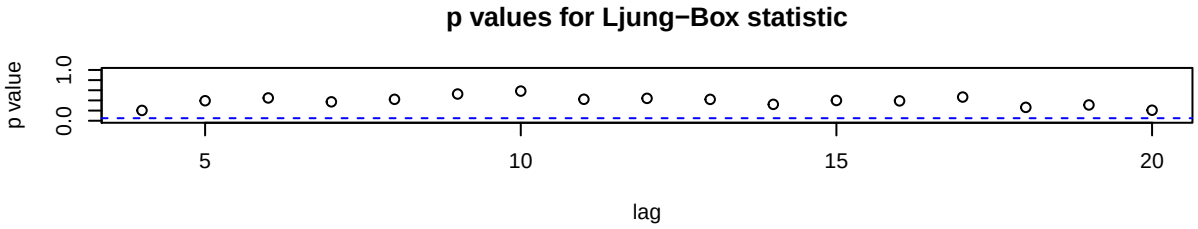
```
## [1] 511
##
## $ttable
##          Estimate     SE  t.value p.value
## ma1       -0.6863 0.0346 -19.8500  0.0000
## sar1      -0.7178 0.0962  -7.4641  0.0000
## sma1       0.8165 0.0774  10.5431  0.0000
## constant   0.1595 0.1212   1.3163  0.1887
##
## $AIC
## [1] 5.236941
##
## $AICc
## [1] 5.241045
##
## $BIC
## [1] 4.269857
```

```
sarima(terror4, 0, 1, 1, 1, 1, 1, 4)
```

```
## initial  value 2.585951
## iter   2 value 2.295422
## iter   3 value 2.231925
## iter   4 value 2.195223
## iter   5 value 2.174548
## iter   6 value 2.150154
## iter   7 value 2.141586
## iter   8 value 2.141306
## iter   9 value 2.141144
## iter  10 value 2.141058
## iter  11 value 2.141057
## iter  11 value 2.141057
## iter  11 value 2.141057
## final  value 2.141057
## converged
## initial  value 2.141567
## iter   2 value 2.139146
## iter   3 value 2.138306
## iter   4 value 2.138235
## iter   5 value 2.138226
## iter   5 value 2.138226
## iter   5 value 2.138226
## final  value 2.138226
## converged
```





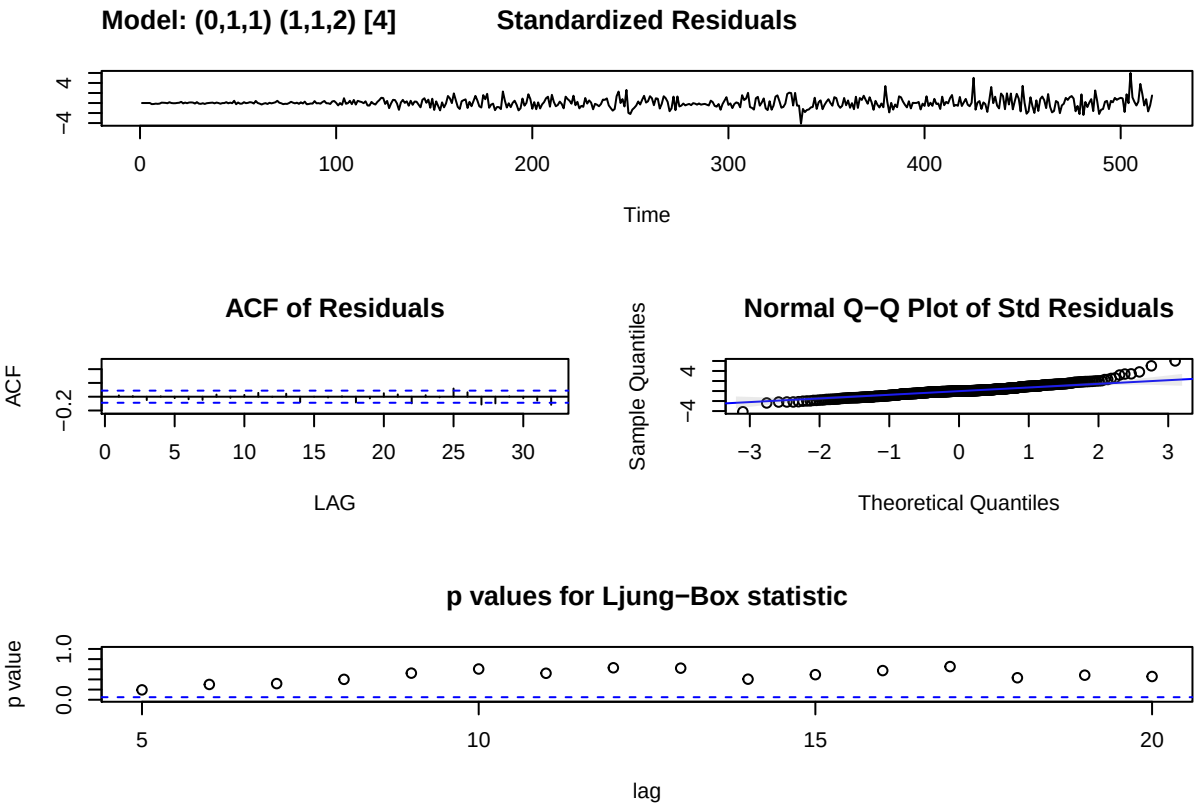



```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##           ma1    sar1     sma1
##       -0.6846  0.0988  -1.0000
## s.e.   0.0362  0.0450   0.0171
##
## sigma^2 estimated as 69.26:  log likelihood = -1817.71,  aic = 3643.42
##
## $degrees_of_freedom
## [1] 508
##
## $ttable
##      Estimate     SE  t.value p.value
## ma1   -0.6846 0.0362 -18.8959  0.0000
## sar1   0.0988 0.0450   2.1935  0.0287
## sma1  -1.0000 0.0171 -58.6006  0.0000
##
## $AIC
## [1] 5.249478
##
## $AICc
## [1] 5.253506
```

```
##
## $BIC
## [1] 4.274165
```

```
sarima(terror4, 0, 1, 1, 1, 1, 2, 4)
```

```
## initial  value 2.585951
## iter   2 value 2.271150
## iter   3 value 2.221817
## iter   4 value 2.190039
## iter   5 value 2.158348
## iter   6 value 2.137558
## iter   7 value 2.136854
## iter   8 value 2.134312
## iter   9 value 2.132861
## iter  10 value 2.132729
## iter  11 value 2.132550
## iter  12 value 2.130744
## iter  13 value 2.127256
## iter  14 value 2.126990
## iter  15 value 2.126819
## iter  16 value 2.126642
## iter  17 value 2.125945
## iter  18 value 2.125863
## iter  19 value 2.125557
## iter  20 value 2.125495
## iter  21 value 2.125487
## iter  22 value 2.125487
## iter  22 value 2.125487
## final  value 2.125487
## converged
## initial  value 2.133766
## iter   2 value 2.133364
## iter   3 value 2.133341
## iter   4 value 2.133331
## iter   5 value 2.133324
## iter   6 value 2.133285
## iter   7 value 2.133273
## iter   8 value 2.133272
## iter   8 value 2.133272
## iter   8 value 2.133272
## final  value 2.133272
## converged
```

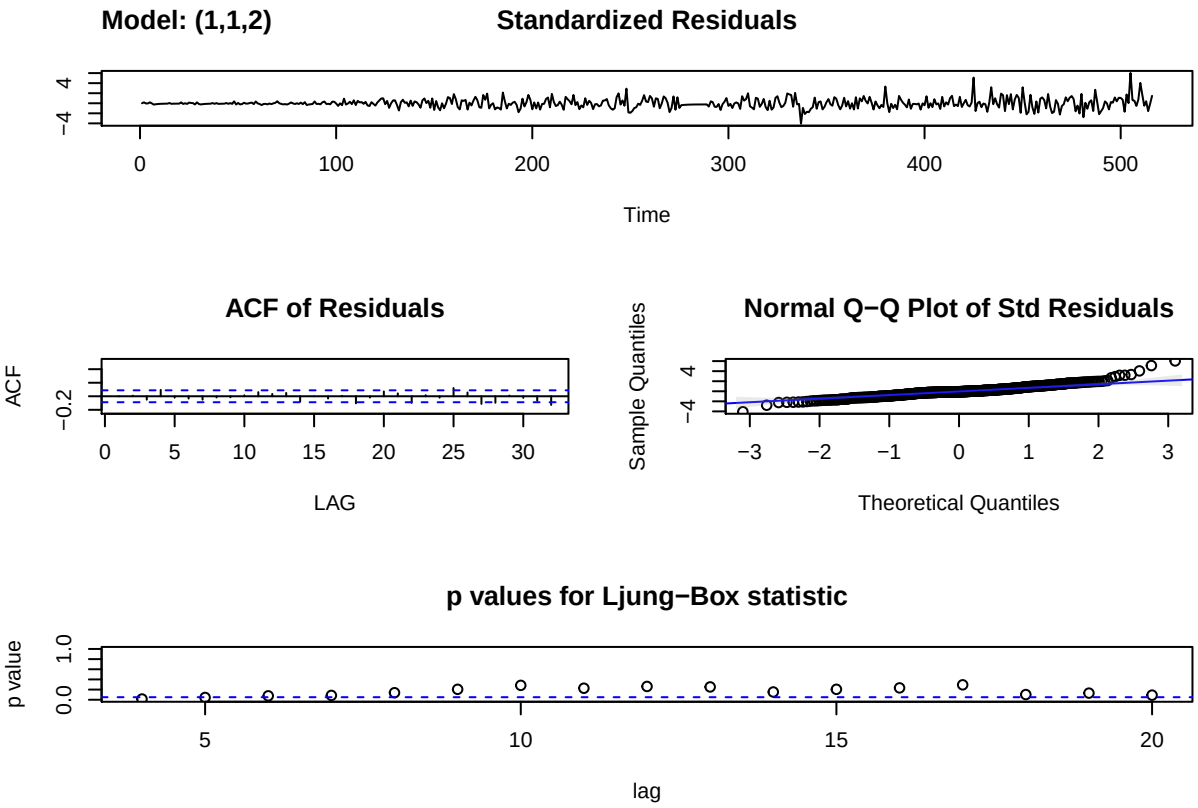



```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##           ma1     sar1     sma1     sma2
##       -0.6840  -0.7141  -0.1856  -0.8144
## s.e.   0.0348   0.0974   0.0807   0.0800
##
## sigma^2 estimated as 68.5:  log likelihood = -1815.18,  aic = 3640.36
##
## $degrees_of_freedom
## [1] 507
##
## $ttable
##      Estimate     SE  t.value p.value
## ma1   -0.6840 0.0348 -19.6474  0.0000
## sar1  -0.7141 0.0974  -7.3348  0.0000
## sma1  -0.1856 0.0807  -2.2985  0.0219
## sma2  -0.8144 0.0800 -10.1848  0.0000
##
## $AIC
## [1] 5.242331
##
## $AICc
```

```
## [1] 5.246435
##
## $BIC
## [1] 4.275247
```

```
sarima(terror4, 1, 1, 2)
```

```
## initial  value 2.303604
## iter   2 value 2.181765
## iter   3 value 2.126083
## iter   4 value 2.122470
## iter   5 value 2.121316
## iter   6 value 2.121288
## iter   7 value 2.121288
## iter   8 value 2.121288
## iter   9 value 2.121288
## iter  10 value 2.121287
## iter  11 value 2.121287
## iter  12 value 2.121287
## iter  13 value 2.121287
## iter  14 value 2.121286
## iter  15 value 2.121285
## iter  16 value 2.121281
## iter  17 value 2.121269
## iter  18 value 2.121254
## iter  19 value 2.121225
## iter  20 value 2.121201
## iter  21 value 2.121196
## iter  22 value 2.121196
## iter  23 value 2.121196
## iter  23 value 2.121196
## iter  23 value 2.121196
## final  value 2.121196
## converged
## initial  value 2.120809
## iter   2 value 2.120807
## iter   3 value 2.120807
## iter   3 value 2.120807
## iter   3 value 2.120807
## final  value 2.120807
## converged
```



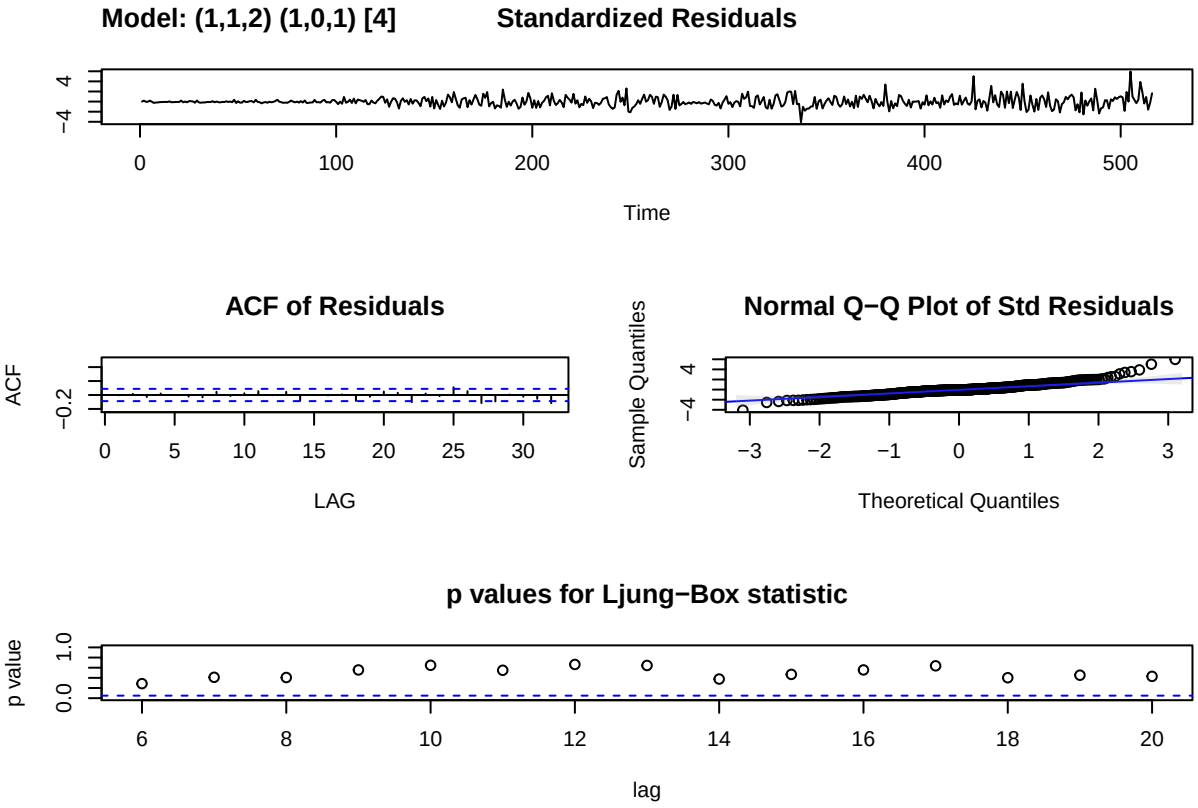

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = constant, optim.control = list(trace = trc, REPORT = 1,
##     reltol = tol))
##
## Coefficients:
##          ar1      ma1      ma2  constant
##       0.0064  -0.6701  -0.0121    0.1591
## s.e.  1.0042   0.9926   0.6599    0.1179
##
## sigma^2 estimated as 69.44:  log likelihood = -1822.97,  aic = 3655.94
##
## $degrees_of_freedom
## [1] 511
##
## $ttable
##          Estimate     SE t.value p.value
## ar1        0.0064 1.0042  0.0064  0.9949
## ma1       -0.6701 0.9926 -0.6751  0.4999
## ma2       -0.0121 0.6599 -0.0184  0.9854
## constant   0.1591 0.1179  1.3492  0.1779
##
## $AIC
## [1] 5.255937
##
## $AICc
```

```
## [1] 5.260041
##
## $BIC
## [1] 4.288853
```

```
sarima(terror4, 1, 1, 2, 1, 0, 1, 4)
```

```
## initial  value 2.307451
## iter   2 value 2.204058
## iter   3 value 2.126118
## iter   4 value 2.123944
## iter   5 value 2.120684
## iter   6 value 2.120475
## iter   7 value 2.120374
## iter   8 value 2.120367
## iter   9 value 2.120339
## iter  10 value 2.120273
## iter  11 value 2.120113
## iter  12 value 2.119505
## iter  13 value 2.118322
## iter  14 value 2.118280
## iter  15 value 2.115656
## iter  16 value 2.115558
## iter  17 value 2.115420
## iter  18 value 2.115254
## iter  19 value 2.114946
## iter  20 value 2.114892
## iter  21 value 2.114761
## iter  22 value 2.114757
## iter  23 value 2.114756
## iter  24 value 2.114754
## iter  25 value 2.114741
## iter  26 value 2.114716
## iter  27 value 2.114657
## iter  28 value 2.114614
## iter  29 value 2.114611
## iter  30 value 2.114610
## iter  31 value 2.114600
## iter  32 value 2.114598
## iter  33 value 2.114597
## iter  34 value 2.114595
## iter  35 value 2.114592
## iter  36 value 2.114591
## iter  37 value 2.114590
## iter  38 value 2.114587
## iter  39 value 2.114580
## iter  40 value 2.114564
## iter  41 value 2.114560
## iter  42 value 2.114558
## iter  43 value 2.114552
## iter  44 value 2.114547
## iter  45 value 2.114540
## iter  46 value 2.114539
## iter  47 value 2.114538
## iter  48 value 2.114537
```

```
## iter  49 value 2.114534
## iter  50 value 2.114527
## iter  51 value 2.114514
## iter  52 value 2.114499
## iter  53 value 2.114490
## iter  54 value 2.114490
## iter  55 value 2.114490
## iter  56 value 2.114490
## iter  57 value 2.114489
## iter  58 value 2.114489
## iter  59 value 2.114489
## iter  59 value 2.114489
## iter  59 value 2.114489
## final  value 2.114489
## converged
## initial  value 2.110481
## iter   2 value 2.110475
## iter   3 value 2.110474
## iter   4 value 2.110473
## iter   4 value 2.110473
## iter   4 value 2.110473
## final  value 2.110473
## converged
```



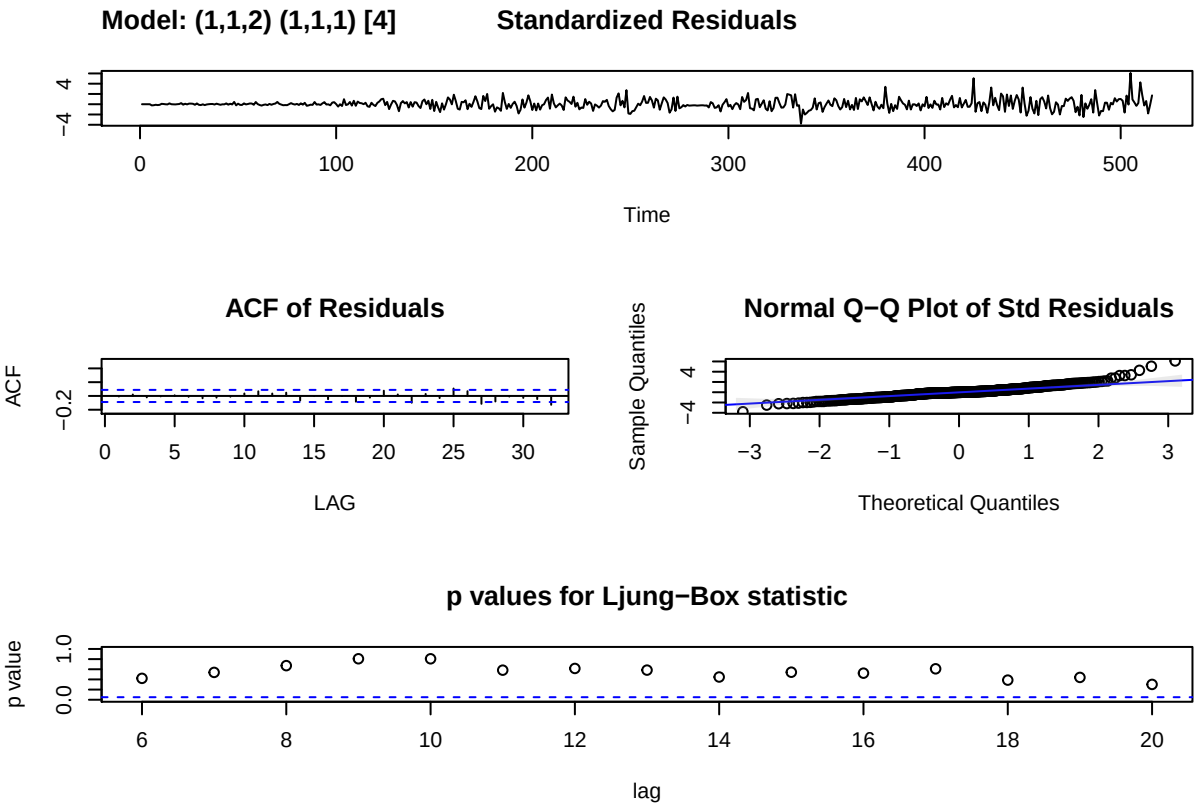

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = constant, optim.control = list(trace = trc, REPORT = 1,
```

```
##     reltol = tol))
##
## Coefficients:
##          ar1      ma1     ma2     sar1    sma1  constant
##       0.5048  -1.1655  0.3073  -0.7013  0.8088    0.1580
## s.e.  0.5414   0.5504  0.3871   0.1014  0.0808    0.1112
##
## sigma^2 estimated as 67.99:  log likelihood = -1817.65,  aic = 3649.29
##
## $degrees_of_freedom
## [1] 509
##
## $ttable
##          Estimate     SE t.value p.value
## ar1        0.5048 0.5414  0.9324  0.3516
## ma1       -1.1655 0.5504 -2.1175  0.0347
## ma2        0.3073 0.3871  0.7938  0.4277
## sar1      -0.7013 0.1014 -6.9171  0.0000
## sma1       0.8088 0.0808 10.0109  0.0000
## constant   0.1580 0.1112  1.4203  0.1561
##
## $AIC
## [1] 5.242543
##
## $AICc
## [1] 5.246847
##
## $BIC
## [1] 4.291917
```

```
sarima(terror4, 1, 1, 2, 1, 1, 1, 4)
```

```
## initial  value 2.586933
## iter   2 value 2.313608
## iter   3 value 2.236307
## iter   4 value 2.190222
## iter   5 value 2.170202
## iter   6 value 2.160653
## iter   7 value 2.153824
## iter   8 value 2.143005
## iter   9 value 2.140794
## iter  10 value 2.139309
## iter  11 value 2.138297
## iter  12 value 2.138283
## iter  13 value 2.138256
## iter  14 value 2.138239
## iter  15 value 2.138099
## iter  16 value 2.137745
## iter  17 value 2.136357
## iter  18 value 2.135551
## iter  19 value 2.132692
## iter  20 value 2.131687
## iter  21 value 2.130851
## iter  22 value 2.130426
## iter  23 value 2.129721
```

```
## iter  24 value 2.129017
## iter  25 value 2.128149
## iter  26 value 2.128113
## iter  27 value 2.127993
## iter  28 value 2.126274
## iter  29 value 2.125847
## iter  30 value 2.125462
## iter  31 value 2.125163
## iter  32 value 2.125044
## iter  33 value 2.125028
## iter  34 value 2.125024
## iter  35 value 2.125024
## iter  35 value 2.125024
## iter  35 value 2.125024
## final  value 2.125024
## converged
## initial  value 2.136116
## iter   2 value 2.136070
## iter   3 value 2.135997
## iter   4 value 2.135991
## iter   5 value 2.135989
## iter   6 value 2.135989
## iter   7 value 2.135988
## iter   8 value 2.135987
## iter   9 value 2.135985
## iter  10 value 2.135983
## iter  11 value 2.135980
## iter  12 value 2.135978
## iter  13 value 2.135978
## iter  14 value 2.135978
## iter  14 value 2.135978
## iter  14 value 2.135978
## final  value 2.135978
## converged
```



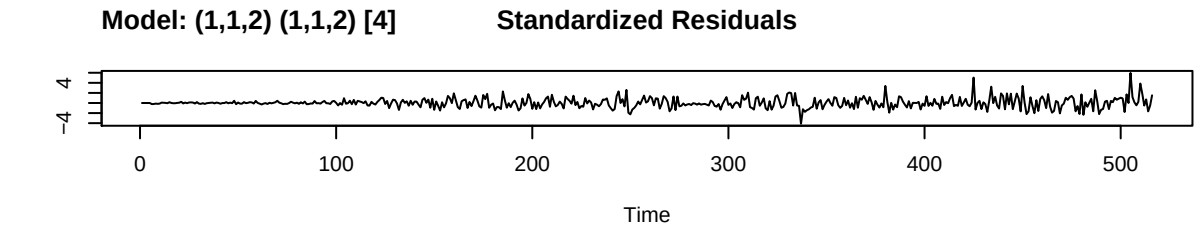

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ma1     ma2    sar1     sma1
##       0.6052  -1.2584  0.3610  0.1332  -1.0000
## s.e.  0.3307   0.3418  0.2472  0.0504   0.0176
##
## sigma^2 estimated as 68.92:  log likelihood = -1816.56,  aic = 3645.12
##
## $degrees_of_freedom
## [1] 506
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1    0.6052 0.3307   1.8302  0.0678
## ma1   -1.2584 0.3418  -3.6817  0.0003
## ma2    0.3610 0.2472   1.4603  0.1448
## sar1   0.1332 0.0504   2.6446  0.0084
## sma1  -1.0000 0.0176 -56.8115  0.0000
##
## $AIC
## [1] 5.252384
##
```

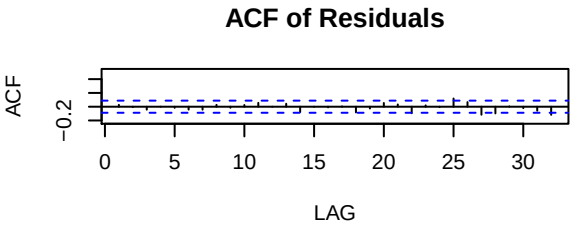
```
## $AICc
## [1] 5.25658
##
## $BIC
## [1] 4.293529
```

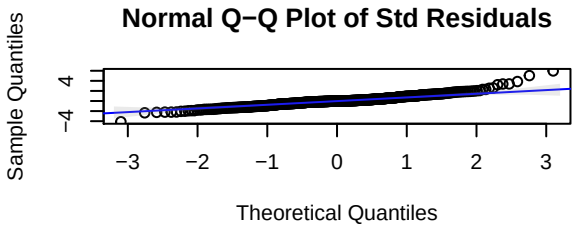
```
sarima(terror4, 1, 1, 2, 1, 1, 2, 4)
```

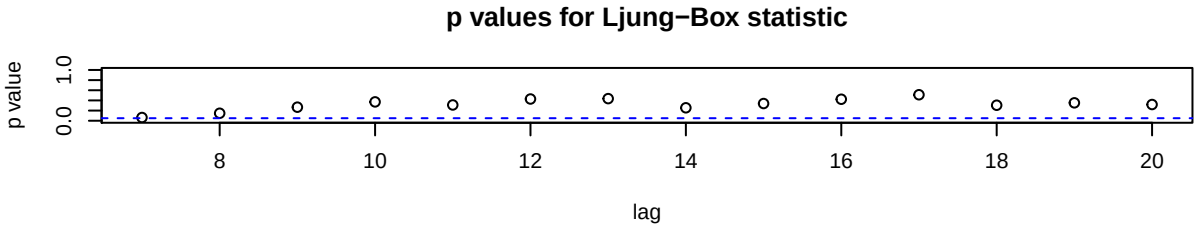
```
## initial  value 2.586933
## iter   2 value 2.289532
## iter   3 value 2.271176
## iter   4 value 2.230852
## iter   5 value 2.225996
## iter   6 value 2.209651
## iter   7 value 2.204810
## iter   8 value 2.190472
## iter   9 value 2.184118
## iter  10 value 2.178785
## iter  11 value 2.176119
## iter  12 value 2.172137
## iter  13 value 2.166308
## iter  14 value 2.164234
## iter  15 value 2.141397
## iter  16 value 2.133893
## iter  17 value 2.129958
## iter  18 value 2.129081
## iter  19 value 2.128562
## iter  20 value 2.128395
## iter  21 value 2.128087
## iter  22 value 2.127913
## iter  23 value 2.127812
## iter  24 value 2.127776
## iter  25 value 2.127771
## iter  26 value 2.127770
## iter  27 value 2.127768
## iter  28 value 2.127764
## iter  29 value 2.127755
## iter  30 value 2.127752
## iter  31 value 2.127750
## iter  32 value 2.127748
## iter  33 value 2.127746
## iter  34 value 2.127741
## iter  35 value 2.127737
## iter  36 value 2.127735
## iter  37 value 2.127733
## iter  38 value 2.127726
## iter  39 value 2.127708
## iter  40 value 2.127662
## iter  41 value 2.127488
## iter  42 value 2.127435
## iter  43 value 2.127375
## iter  44 value 2.127362
## iter  45 value 2.127347
## iter  46 value 2.127333
## iter  47 value 2.127317
```

```
## iter  48 value 2.127314
## iter  49 value 2.127312
## iter  50 value 2.127307
## iter  51 value 2.127296
## iter  52 value 2.127264
## iter  53 value 2.127228
## iter  54 value 2.127190
## iter  55 value 2.127175
## iter  56 value 2.127173
## iter  57 value 2.127169
## iter  58 value 2.127152
## iter  59 value 2.127148
## iter  60 value 2.127139
## iter  61 value 2.127132
## iter  62 value 2.127131
## iter  63 value 2.127131
## iter  64 value 2.127129
## iter  65 value 2.127127
## iter  66 value 2.127123
## iter  67 value 2.127121
## iter  68 value 2.127121
## iter  69 value 2.127121
## iter  70 value 2.127121
## iter  71 value 2.127121
## iter  72 value 2.127121
## iter  73 value 2.127120
## iter  74 value 2.127120
## iter  74 value 2.127120
## iter  74 value 2.127120
## final  value 2.127120
## converged
## initial  value 2.133693
## iter   2 value 2.133179
## iter   3 value 2.133153
## iter   4 value 2.133143
## iter   5 value 2.133139
## iter   6 value 2.133127
## iter   7 value 2.133103
## iter   8 value 2.133100
## iter   9 value 2.133100
## iter  10 value 2.133100
## iter  11 value 2.133099
## iter  12 value 2.133097
## iter  13 value 2.133092
## iter  14 value 2.133082
## iter  15 value 2.133082
## iter  16 value 2.133080
## iter  17 value 2.133080
## iter  18 value 2.133080
## iter  18 value 2.133080
## iter  18 value 2.133080
## final  value 2.133080
## converged
```





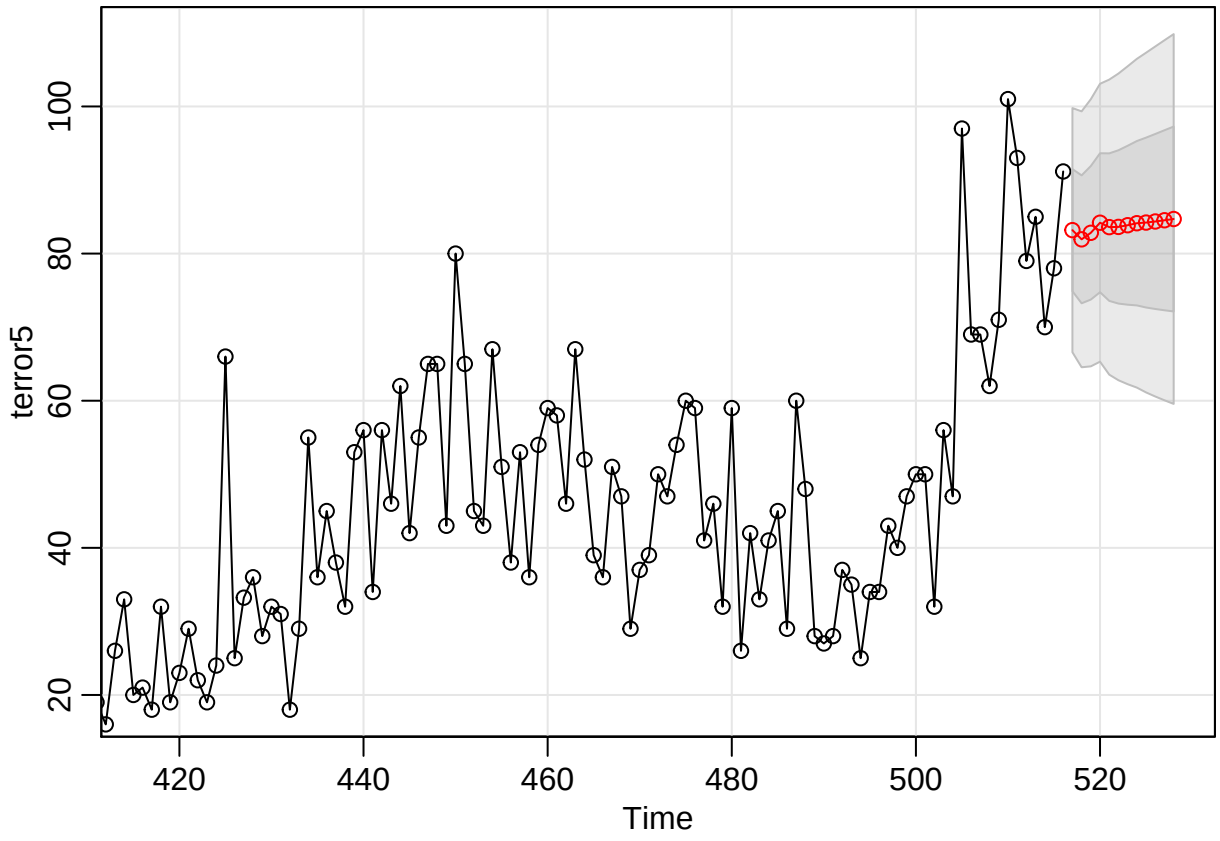


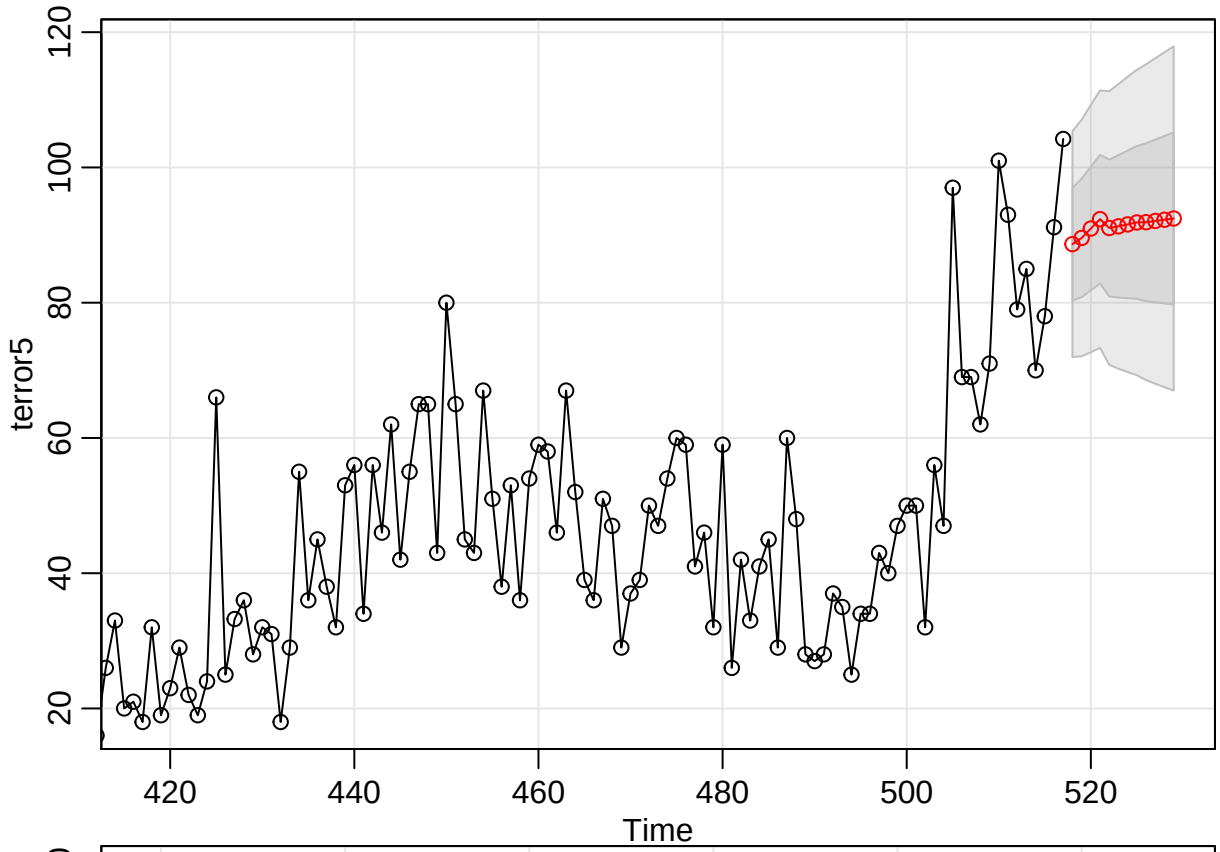


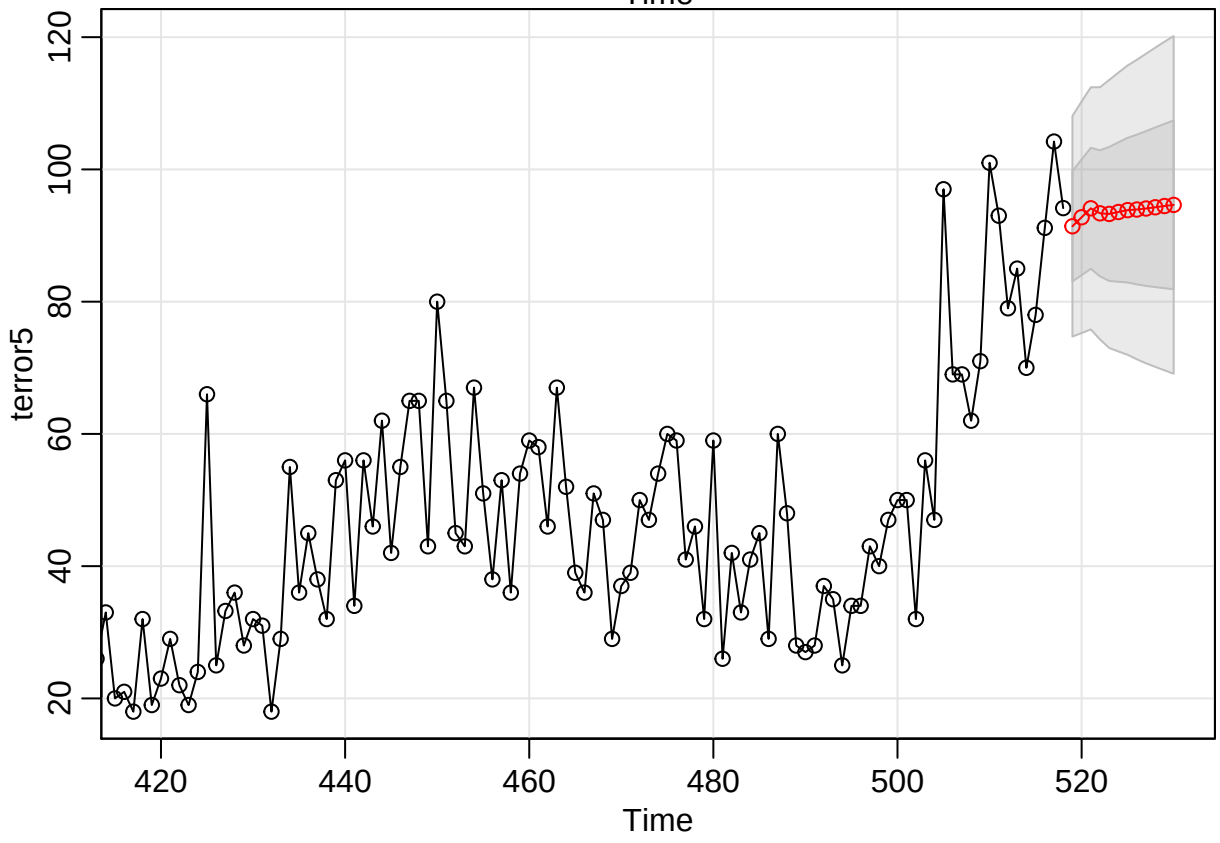

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##           ar1     ma1      ma2     sar1     sma1     sma2
##       -0.8094  0.1185  -0.5406  -0.7206  -0.1827  -0.8172
## s.e.   0.2450  0.2505   0.1769   0.0978   0.0806   0.0798
##
## sigma^2 estimated as 68.49:  log likelihood = -1815.08,  aic = 3644.16
##
## $degrees_of_freedom
## [1] 505
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1   -0.8094 0.2450  -3.3039  0.0010
## ma1    0.1185 0.2505   0.4730  0.6364
## ma2   -0.5406 0.1769  -3.0563  0.0024
## sar1  -0.7206 0.0978  -7.3717  0.0000
## sma1  -0.1827 0.0806  -2.2673  0.0238
## sma2  -0.8172 0.0798 -10.2401  0.0000
##
## $AIC
## [1] 5.24995
```

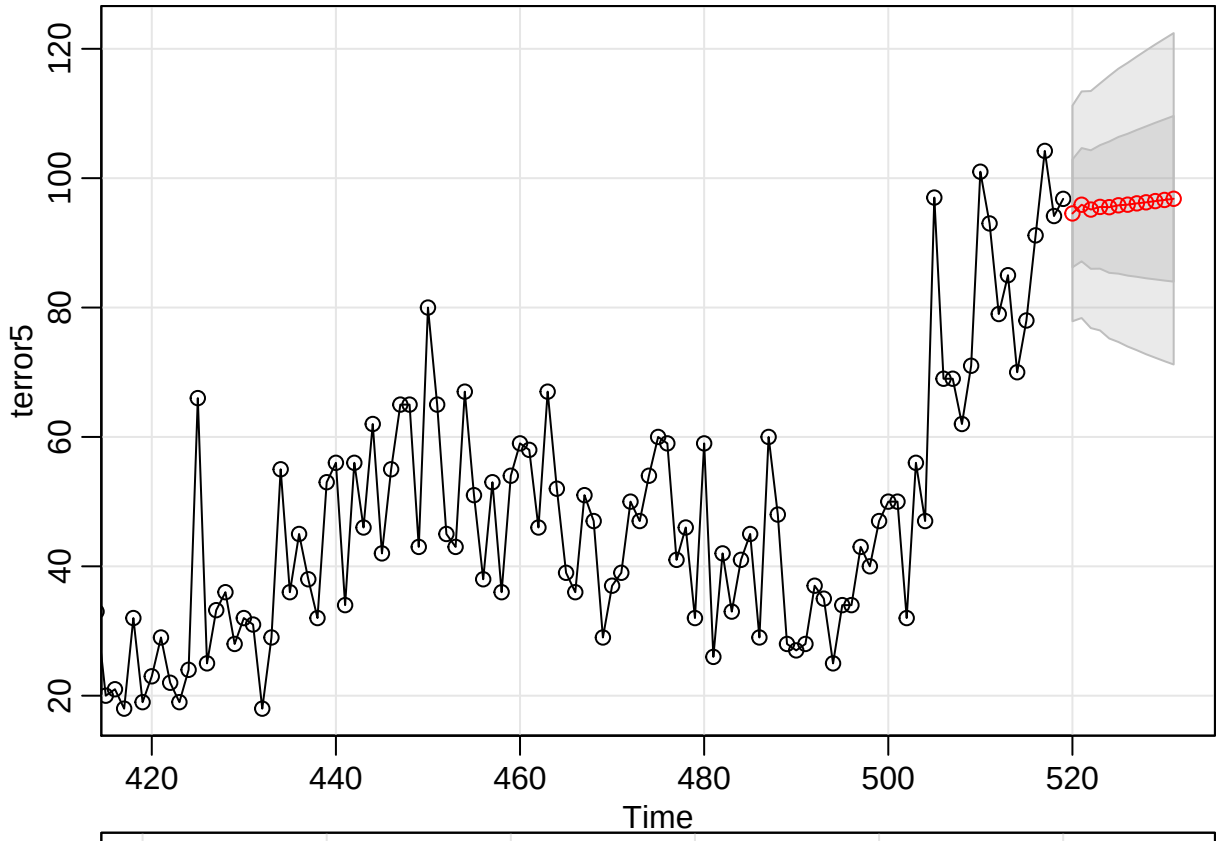
```
##
## $AICc
## [1] 5.254253
##
## $BIC
## [1] 4.299323
```

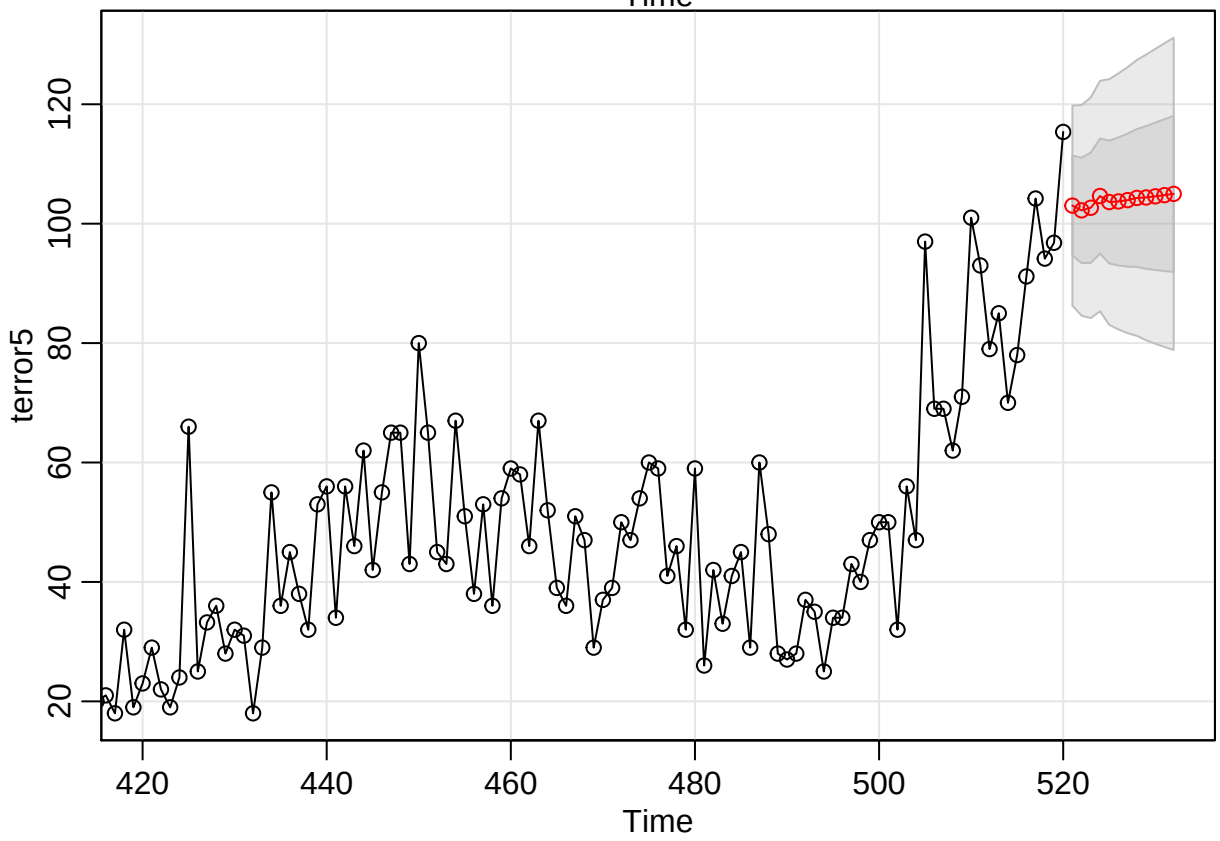
```r
#eacf(diff(diff(log_terror4)))
terror5 <- terror4
total_error <- 0
for (i in 1: (length(terror4.valid) - 11))
{
  actual <- terror4.valid[i : i + 11]
  predicted <- sarima.for(terror5, 12, 0, 1, 1, 1, 0, 0, 4)$pred
  total_error <- total_error + sum((actual - predicted)^2)
  terror5 <- c(terror5, terror4.valid[i])
}
```

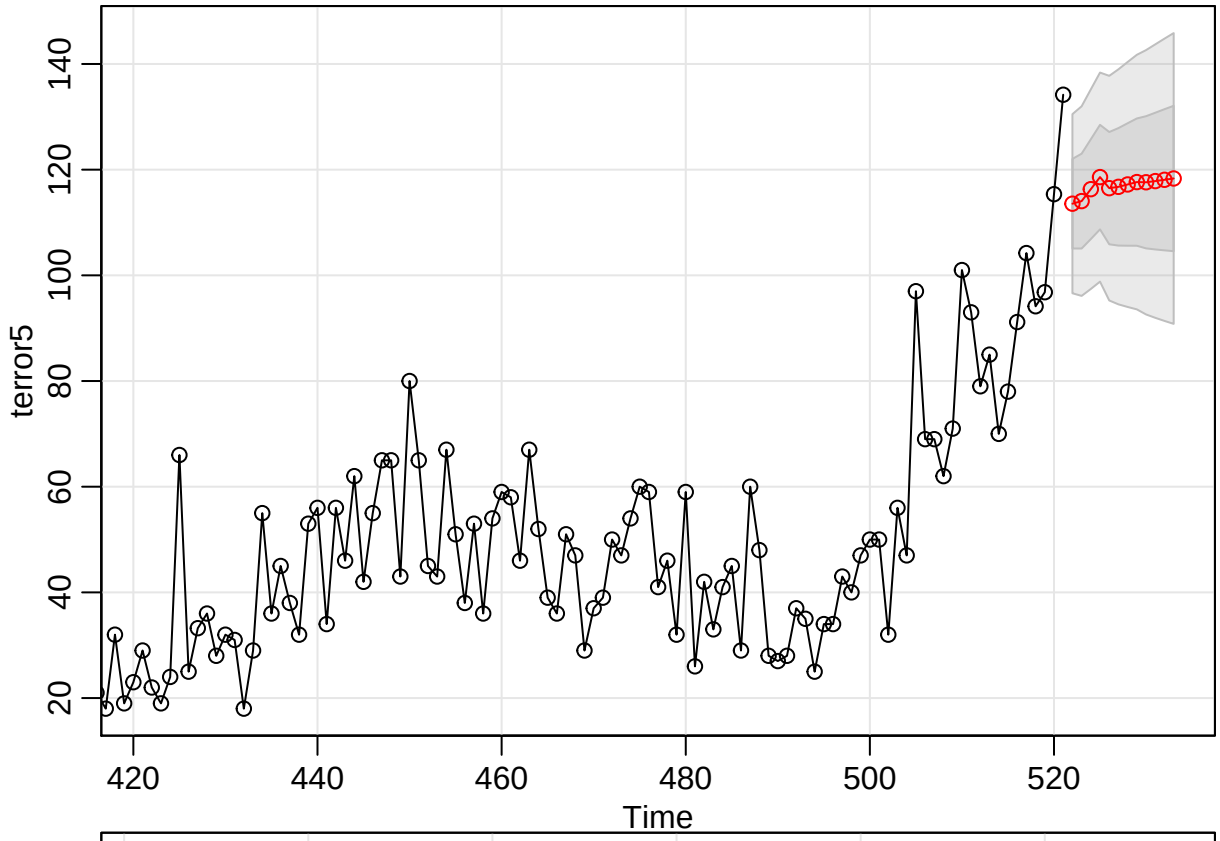


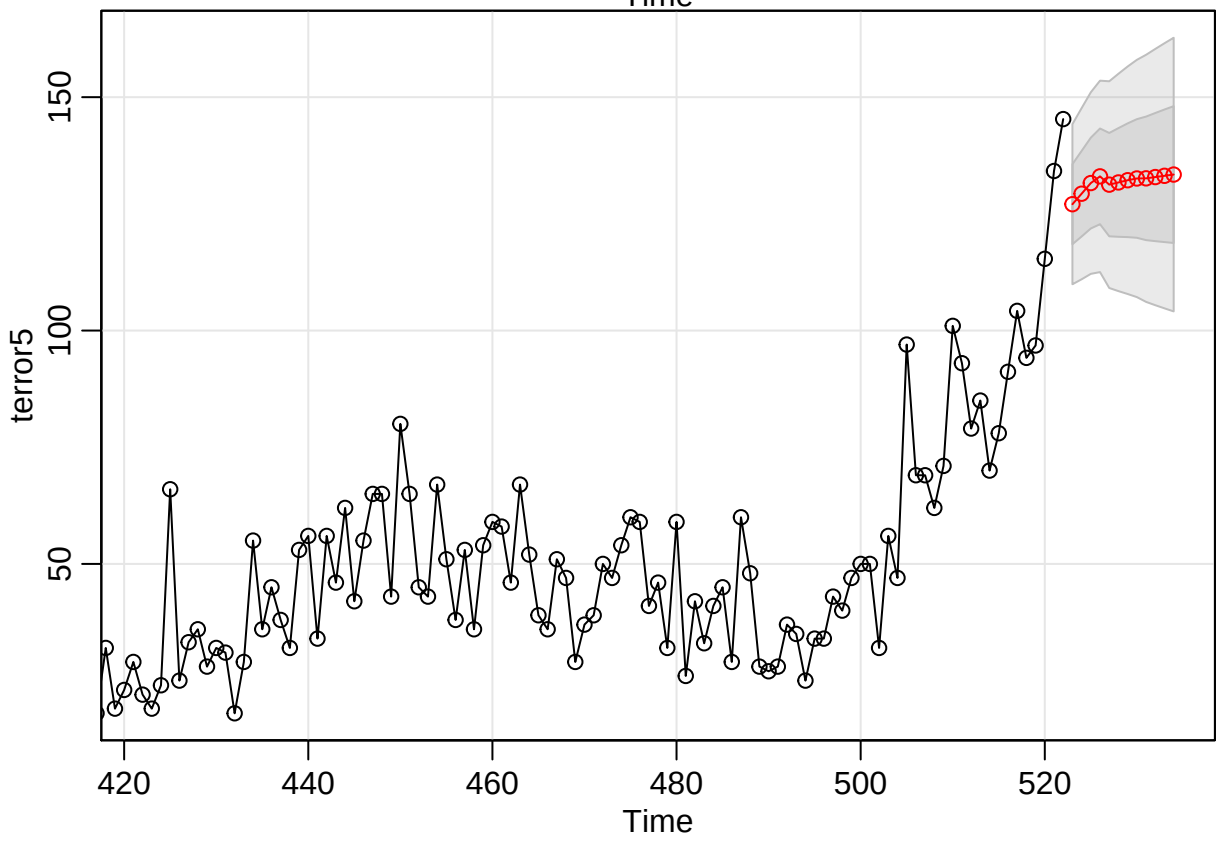

terror5
120
100
80
60
40
20
420
440
460
480
500
520
Time

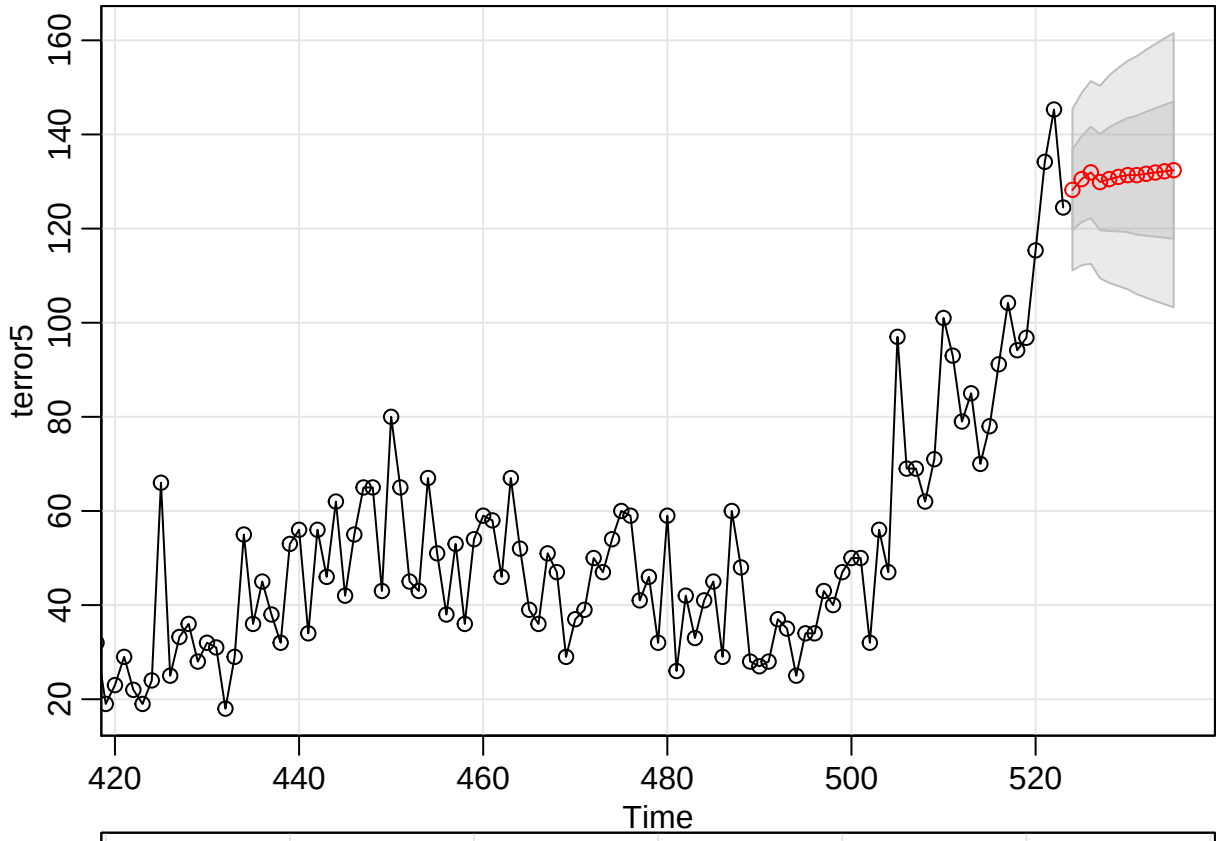

terror5
120
100
80
60
40
20
420
440
460
480
500
520
Time

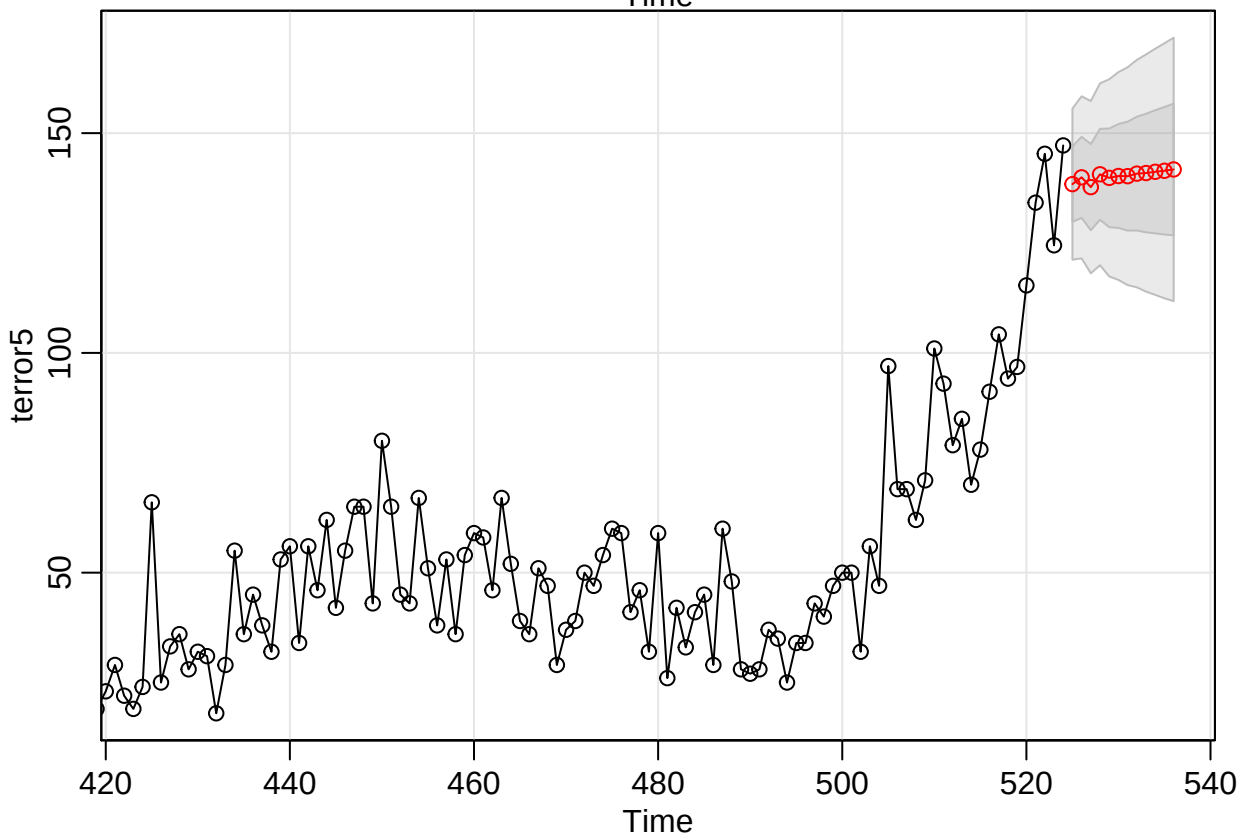

terror5
20
40
60
80
100
120
420
440
460
480
500
520
Time

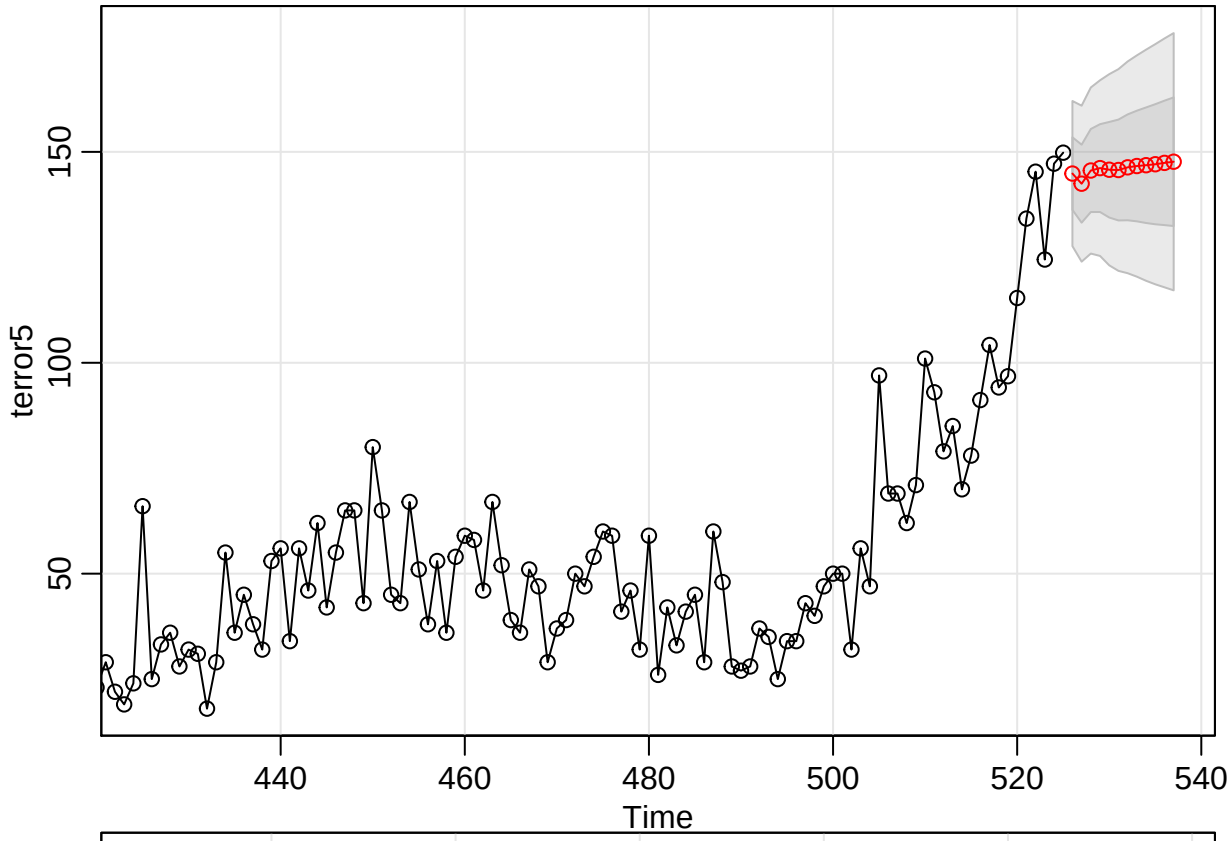

terror5
20
40
60
80
100
120
420
440
460
480
500
520
Time

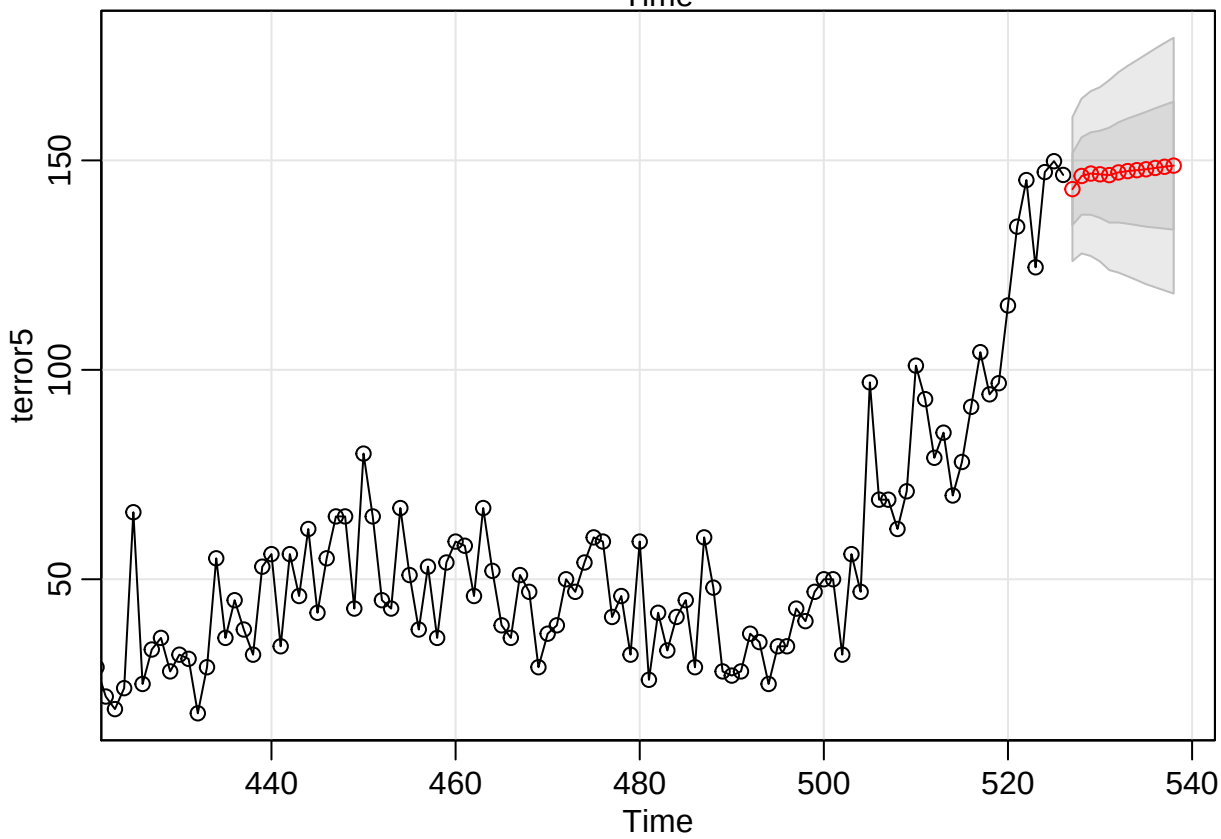

terror5
20
40
60
80
100
120
140
420
440
460
480
500
520
Time

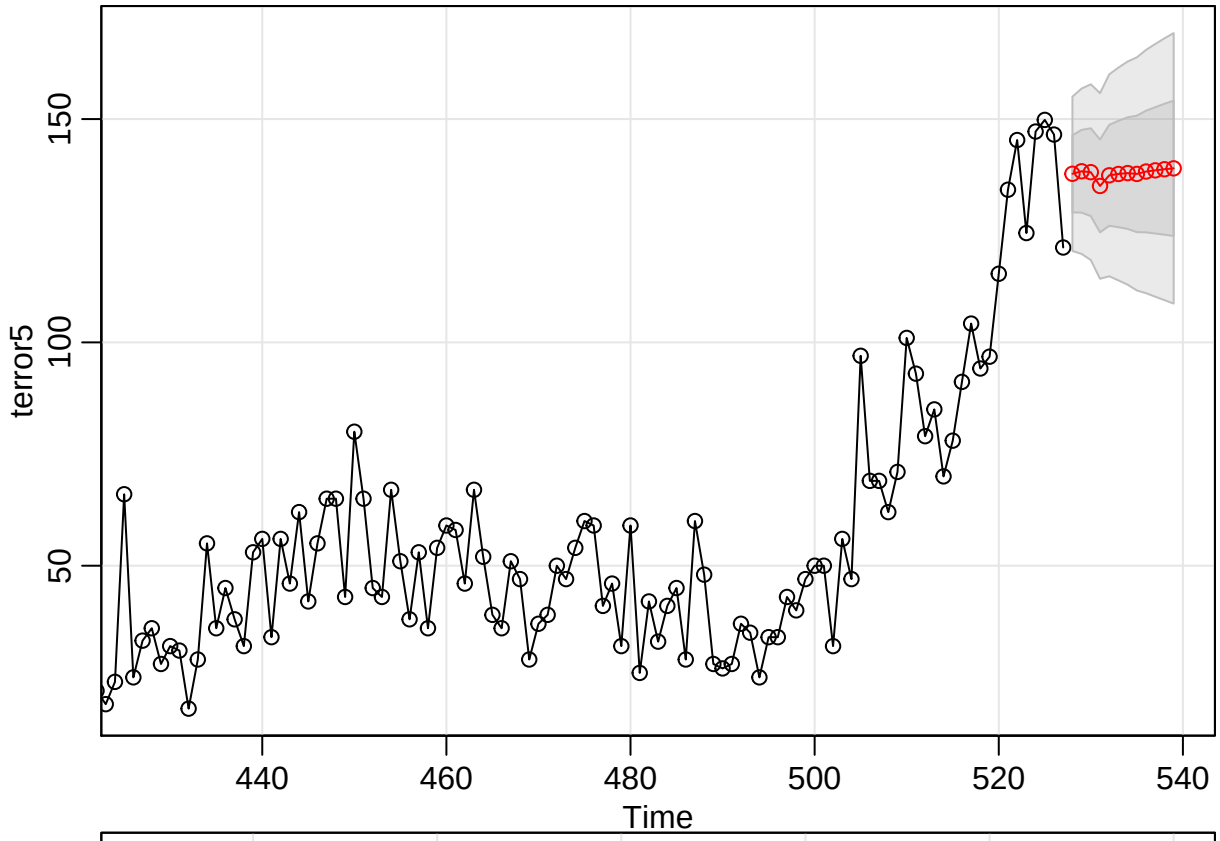

terror5
50
100
150
420
440
460
480
500
520
Time

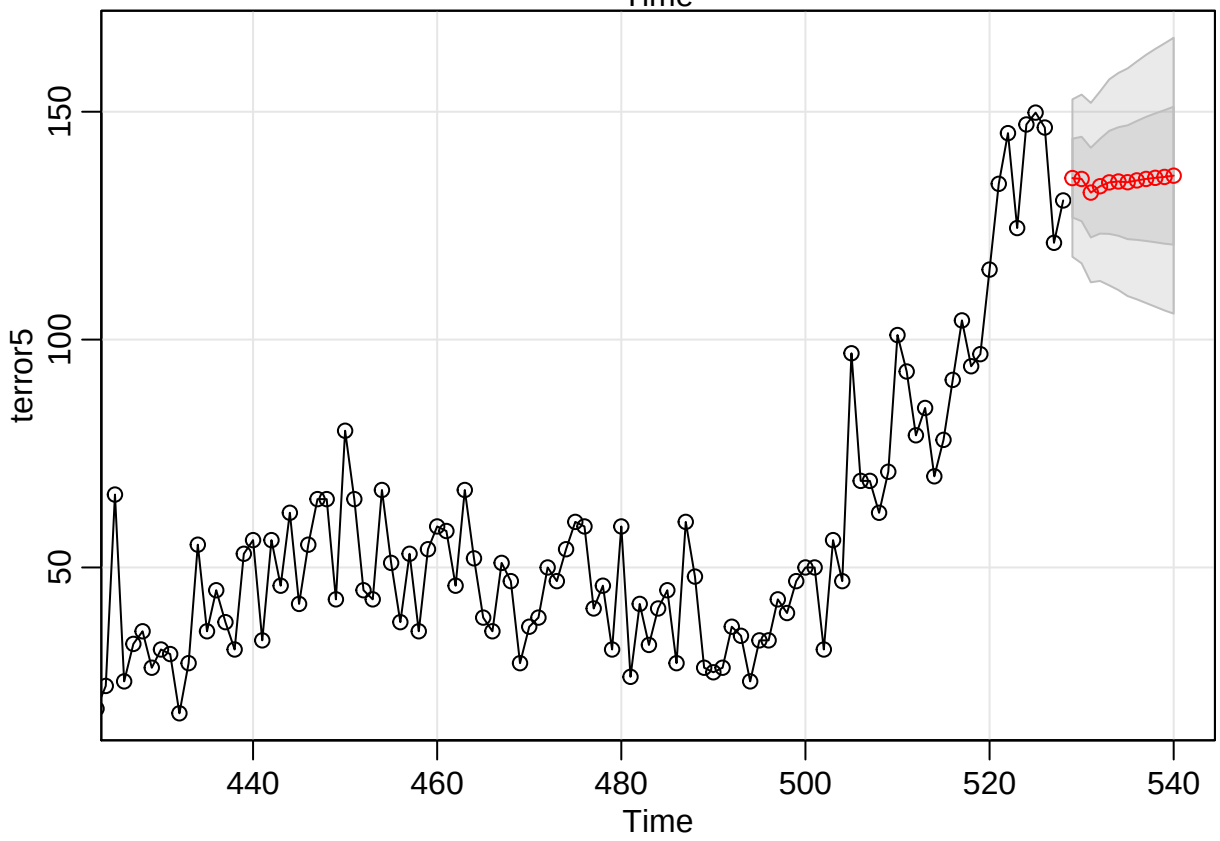

terror5
20
40
60
80
100
120
140
160
420
440
460
480
500
520
Time

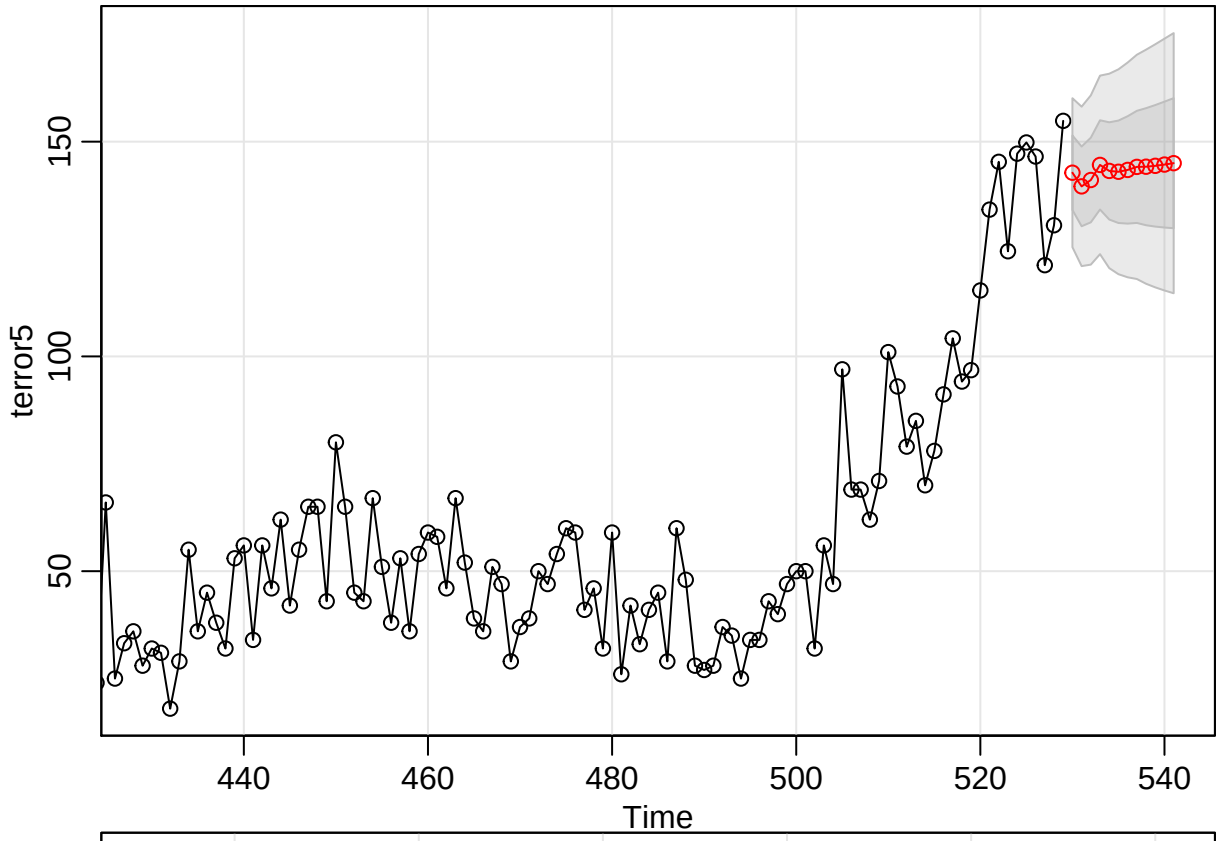

terror5
50
100
150
420
440
460
480
500
520
540
Time

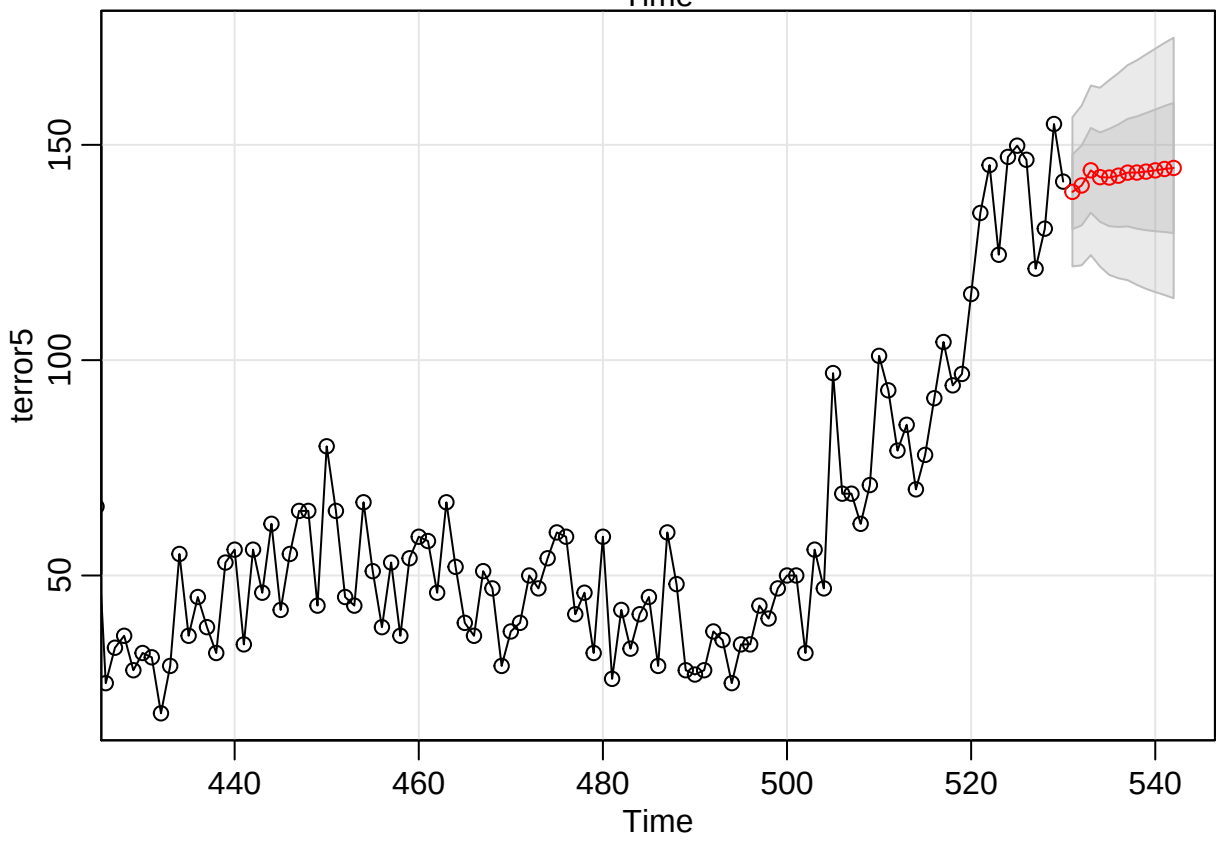

terror5
150
100
50
440
460
480
500
520
540
Time

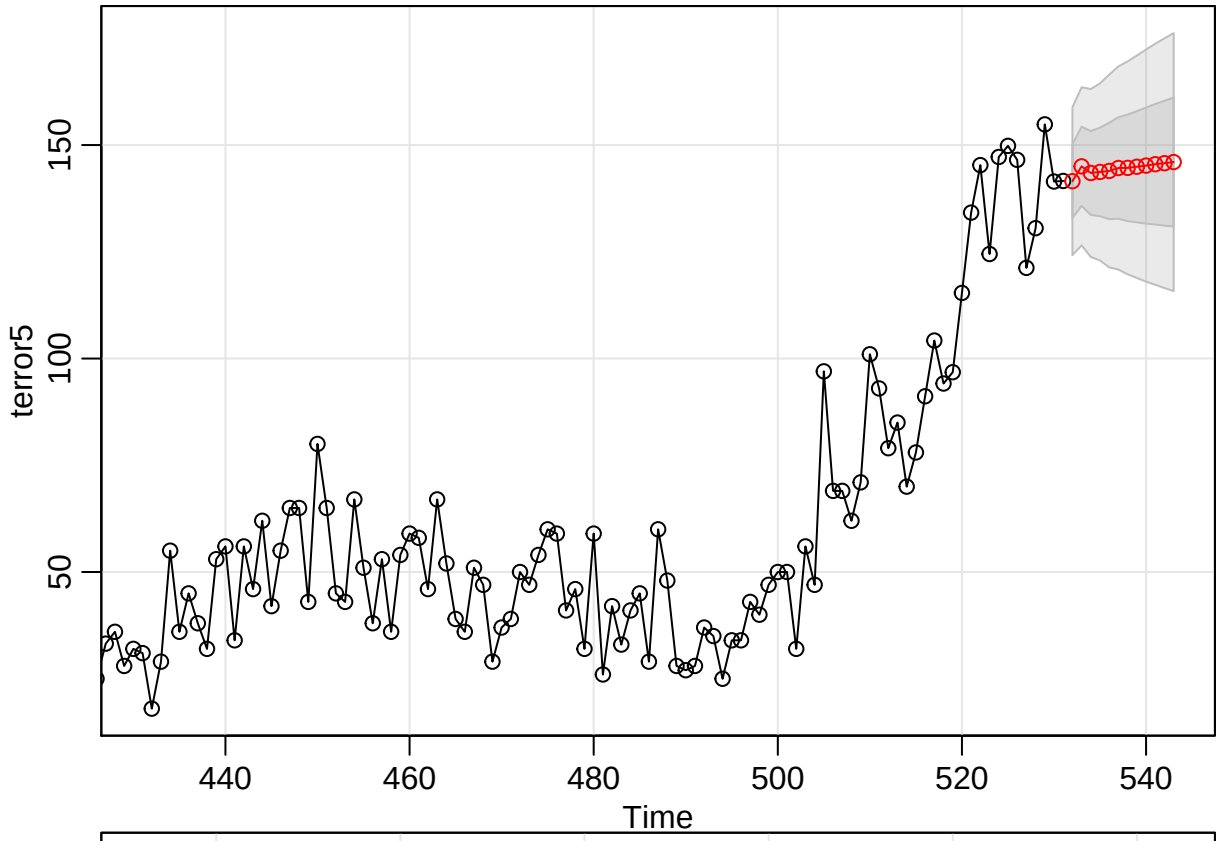

terror5
150
100
50
440
460
480
500
520
540
Time

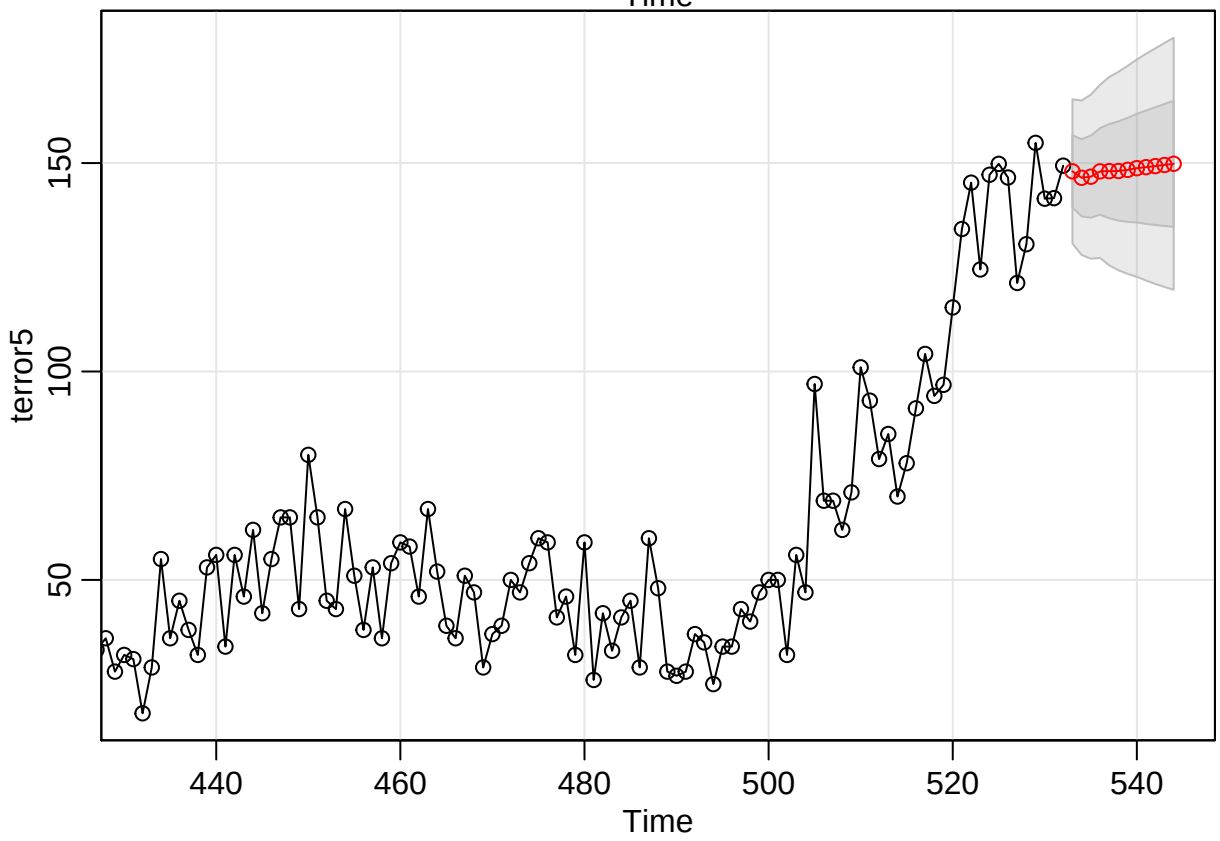

terror5
150
100
50
440
460
480
500
520
540
Time

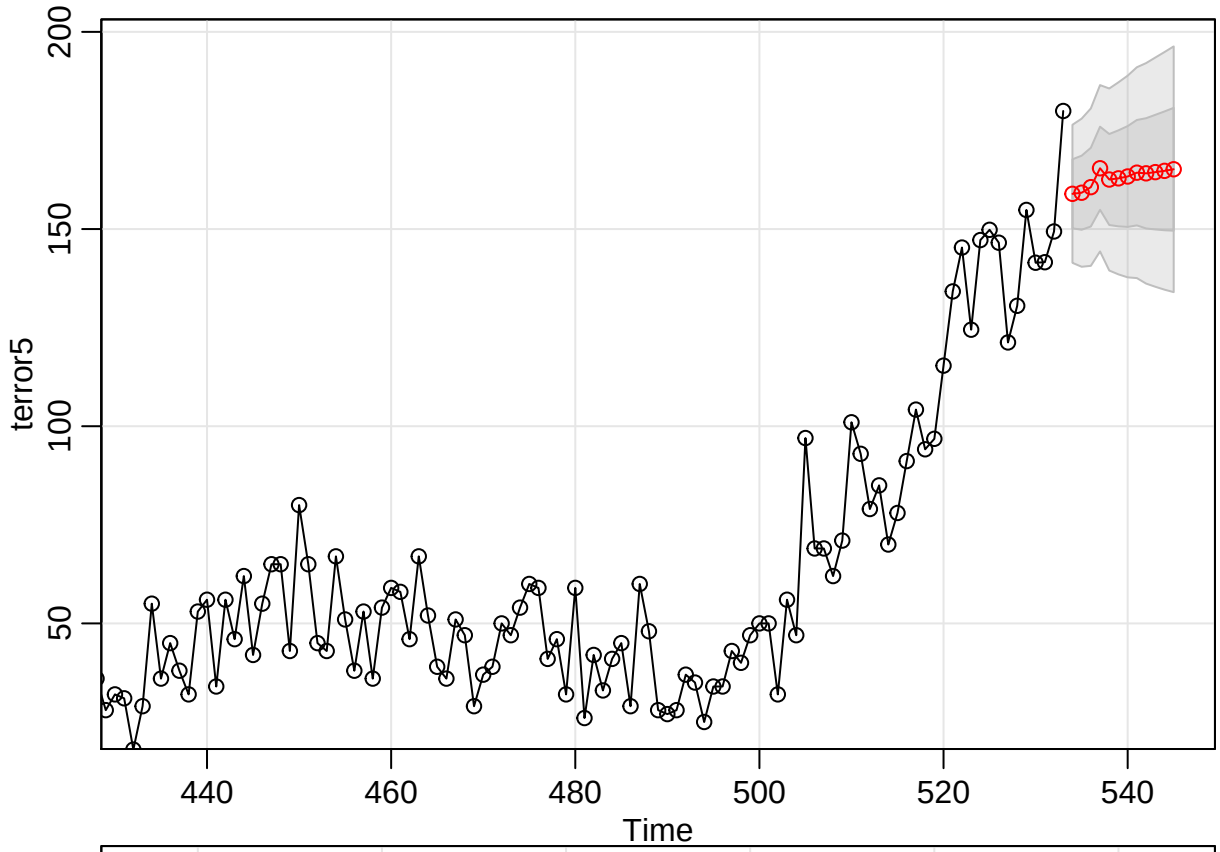

terror5
150
100
50
440
460
480
500
520
540
Time

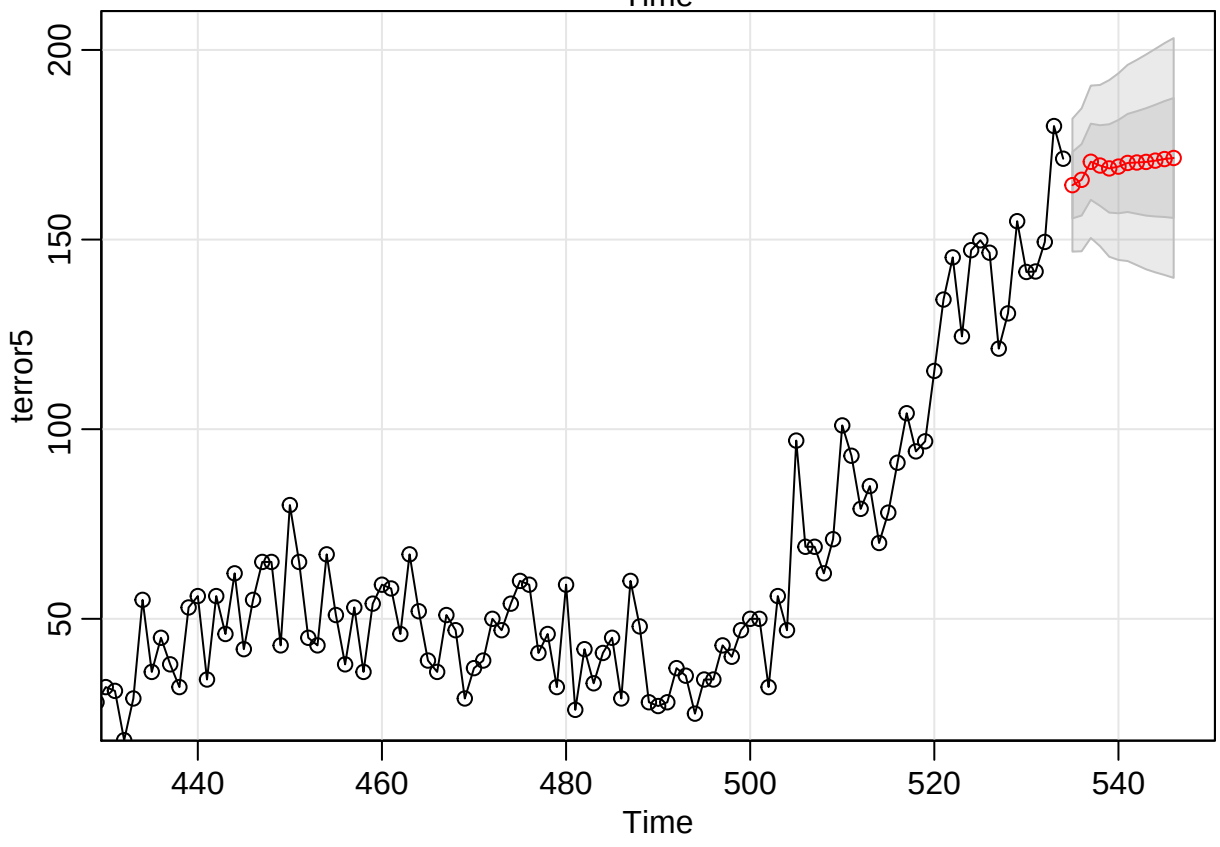

terror5
150
100
50
440
460
480
500
520
540
Time

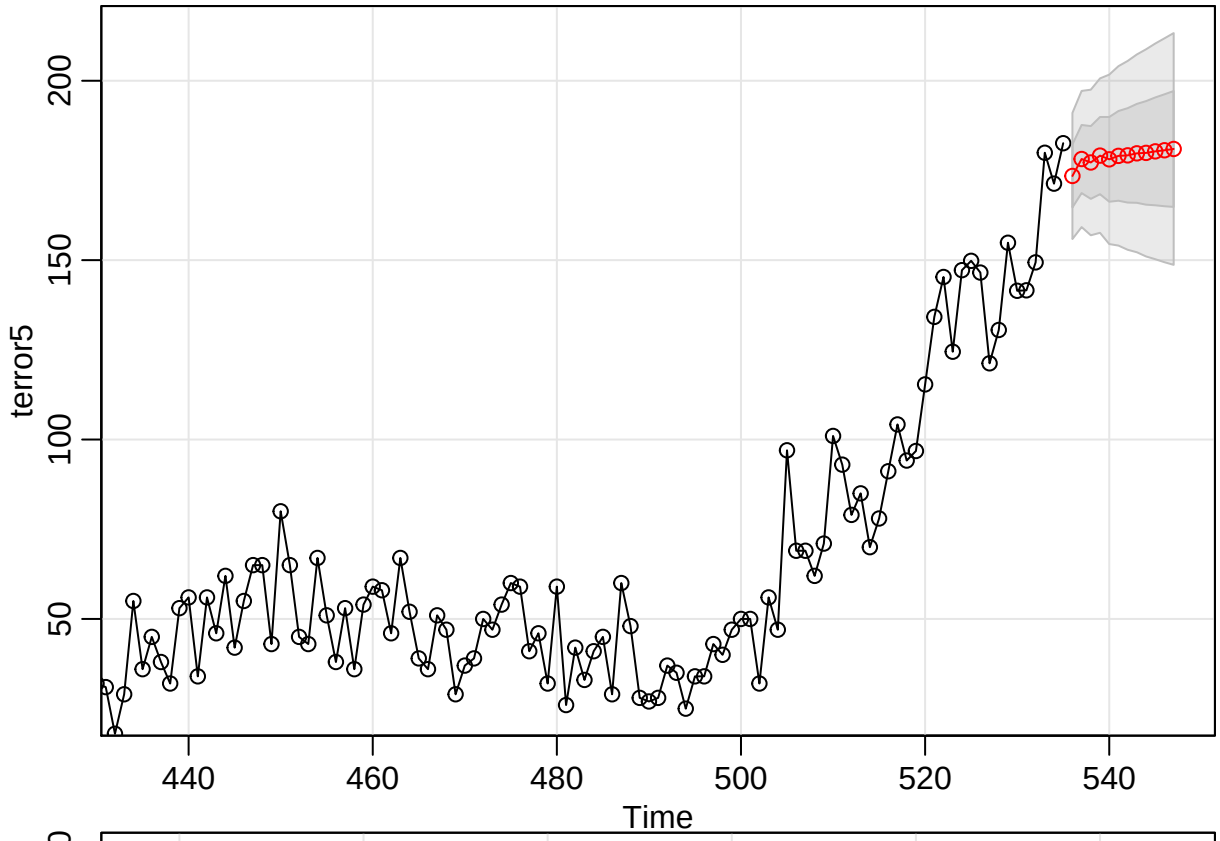

terror5
150
100
50
440
460
480
500
520
540
Time

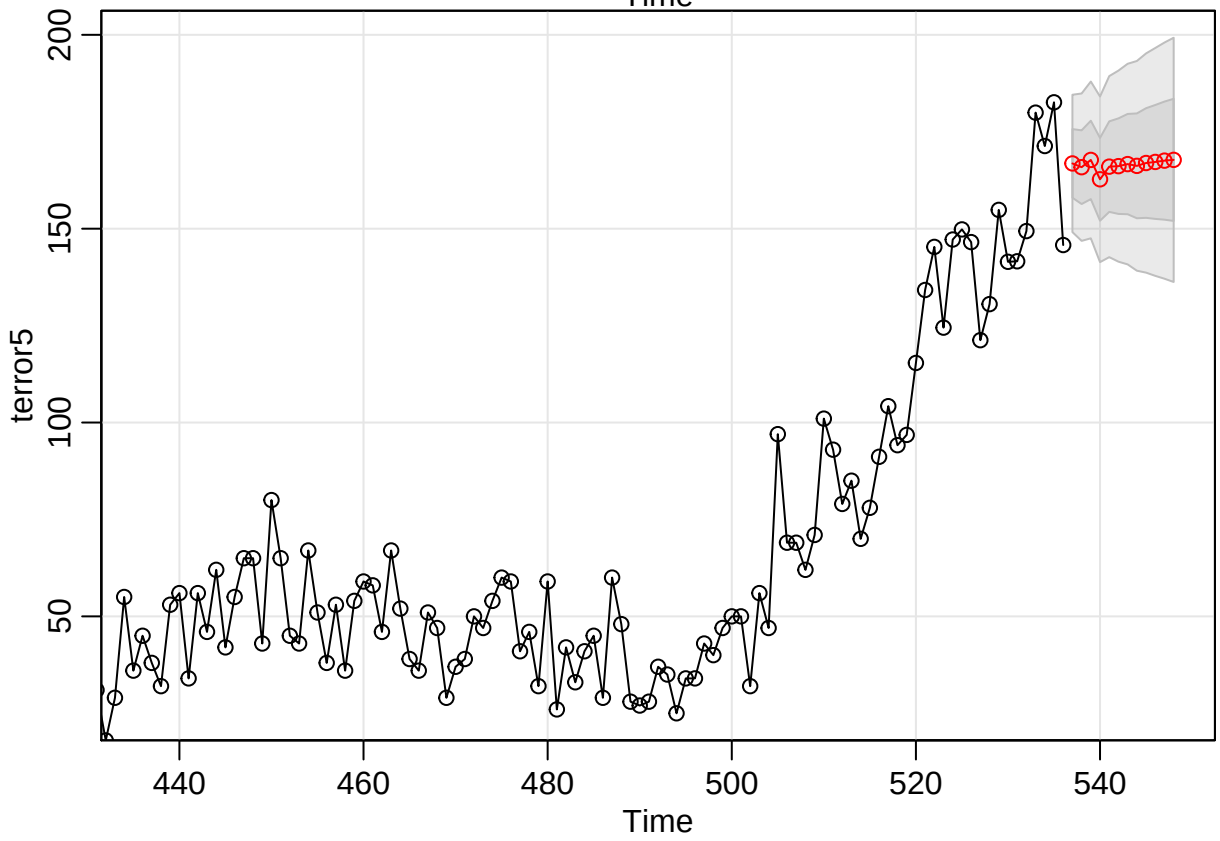

terror5
150
100
50
440
460
480
500
520
540
Time

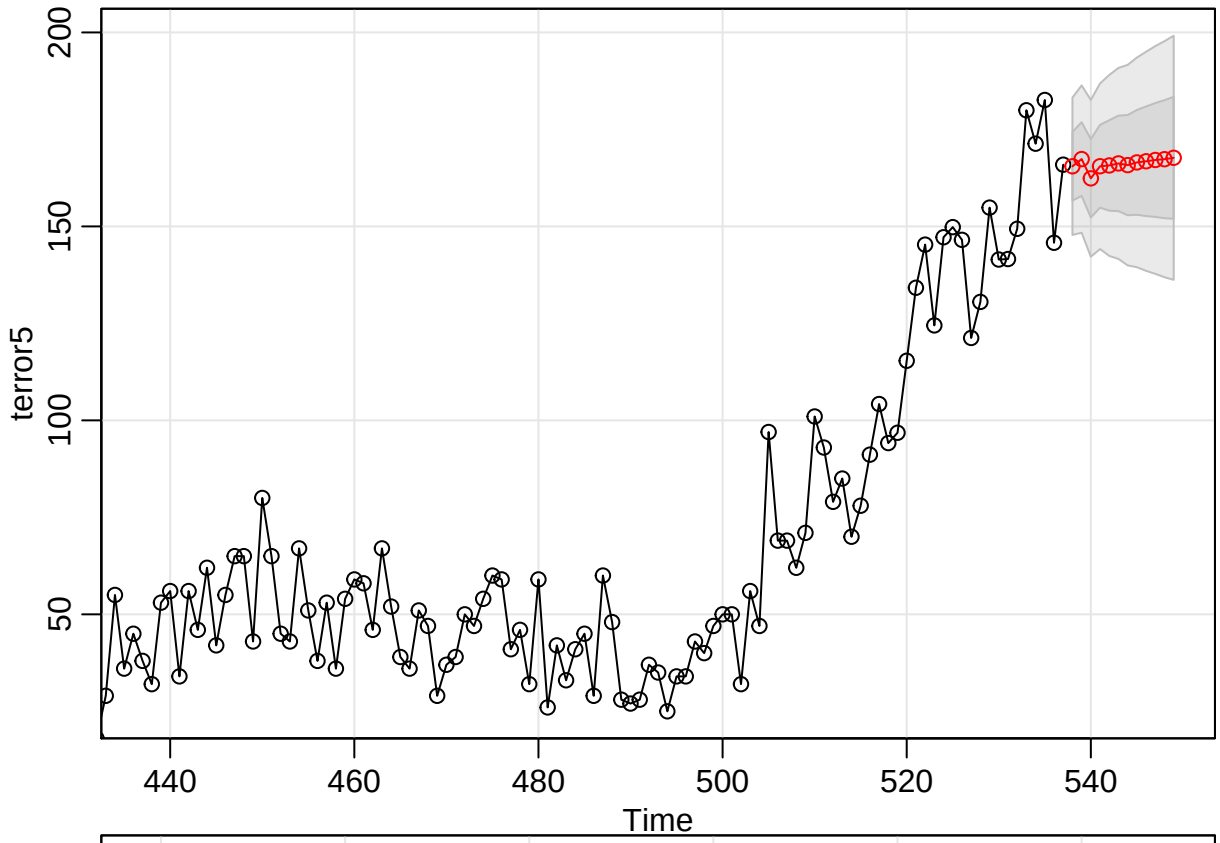

terror5
150
100
50
440
460
480
500
520
540
Time

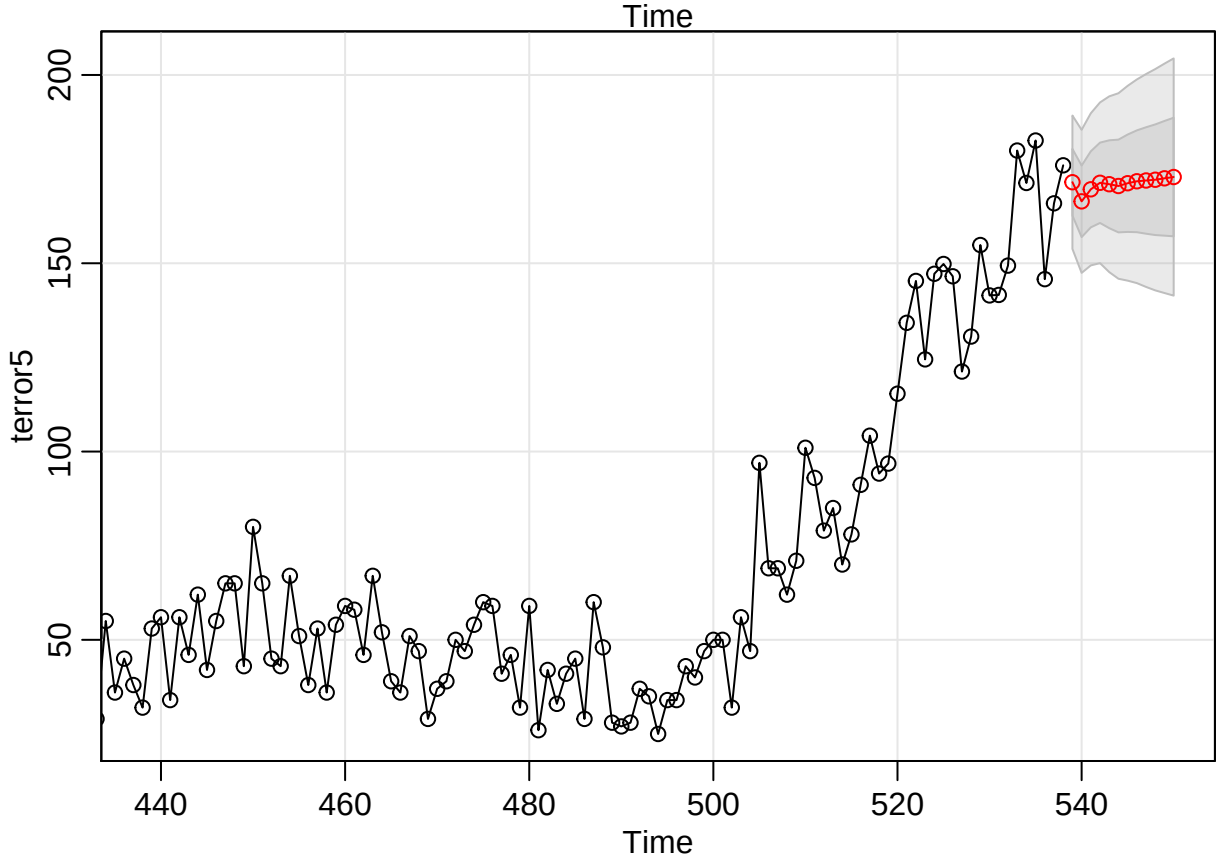

terror5
200
150
100
50
440
460
480
500
520
540
Time

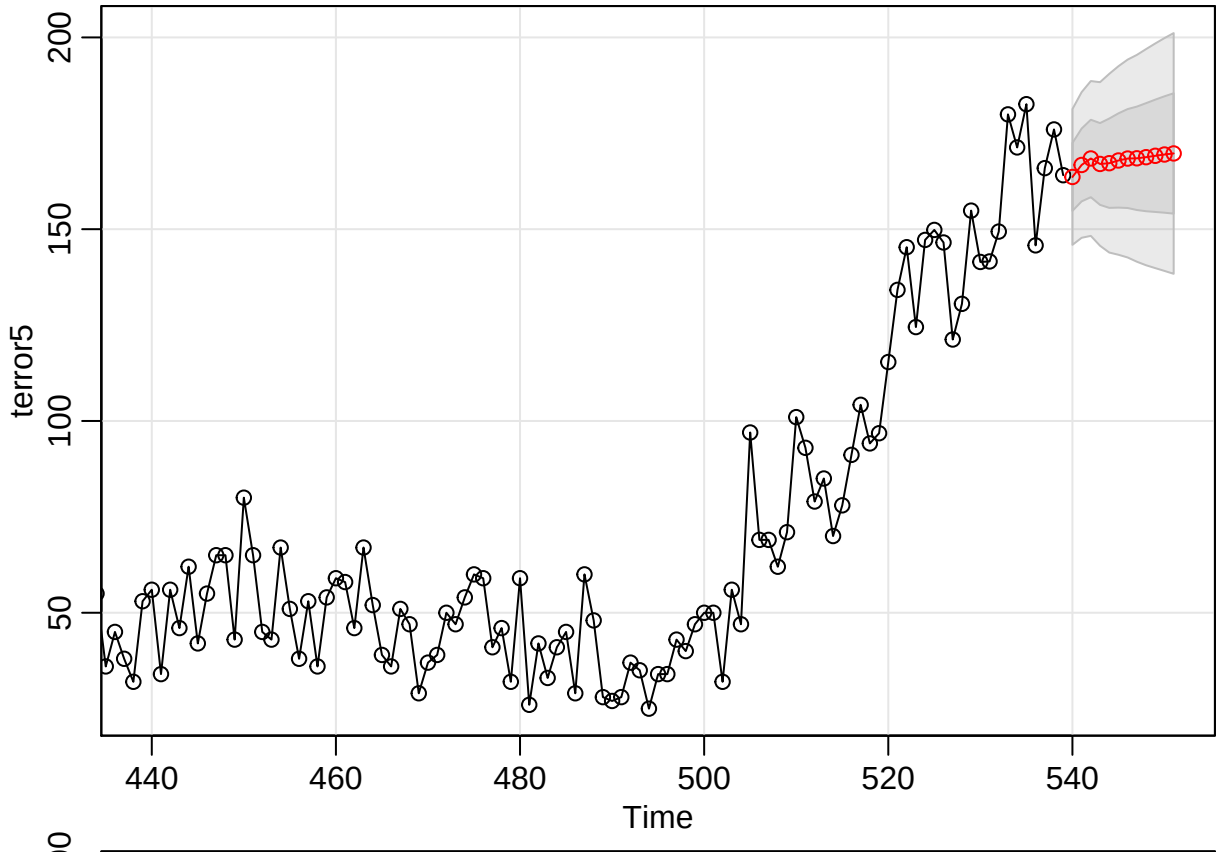

terror5
200
150
100
50
440
460
480
500
520
540
Time

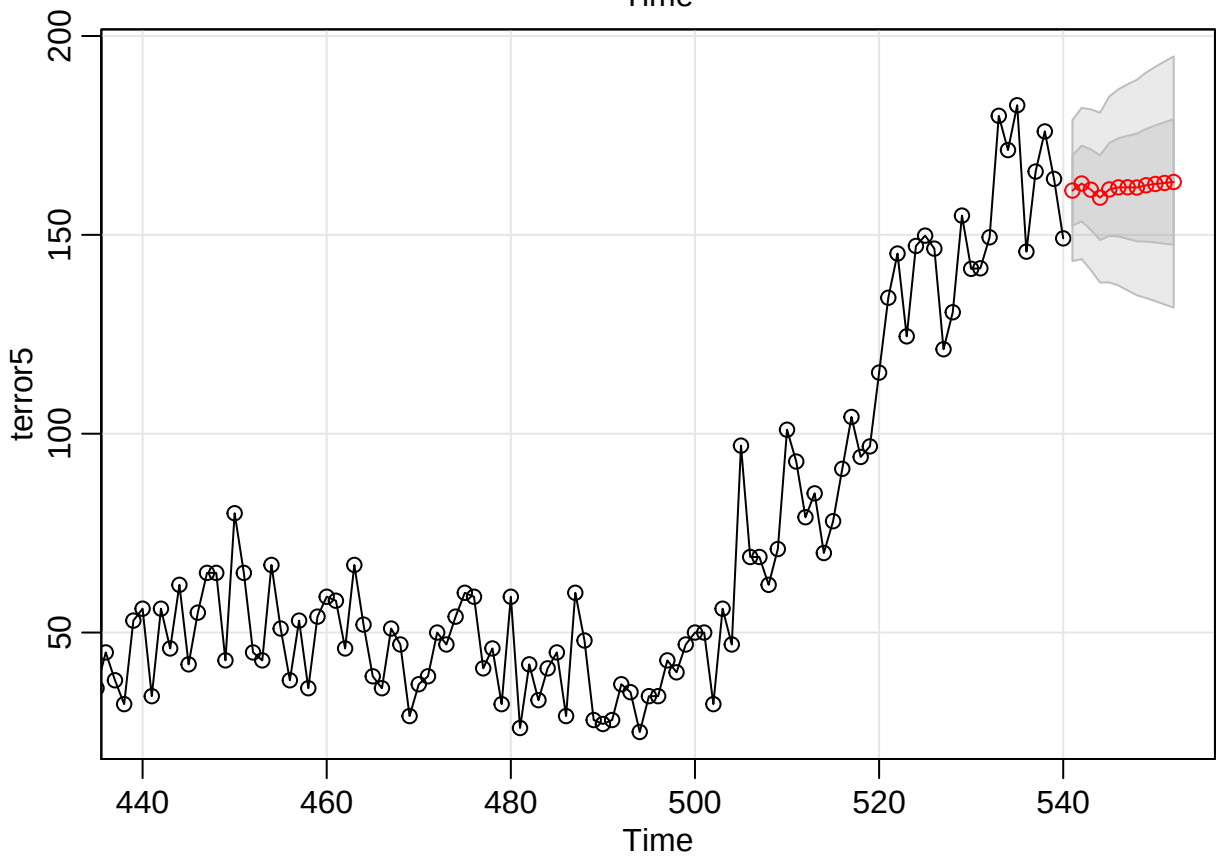

terror5
50
100
150
200
440
460
480
500
520
540
Time


terror5
50
100
150
200
440
460
480
500
520
540
Time


terror5
200
150
100
50
440
460
480
500
520
540
Time

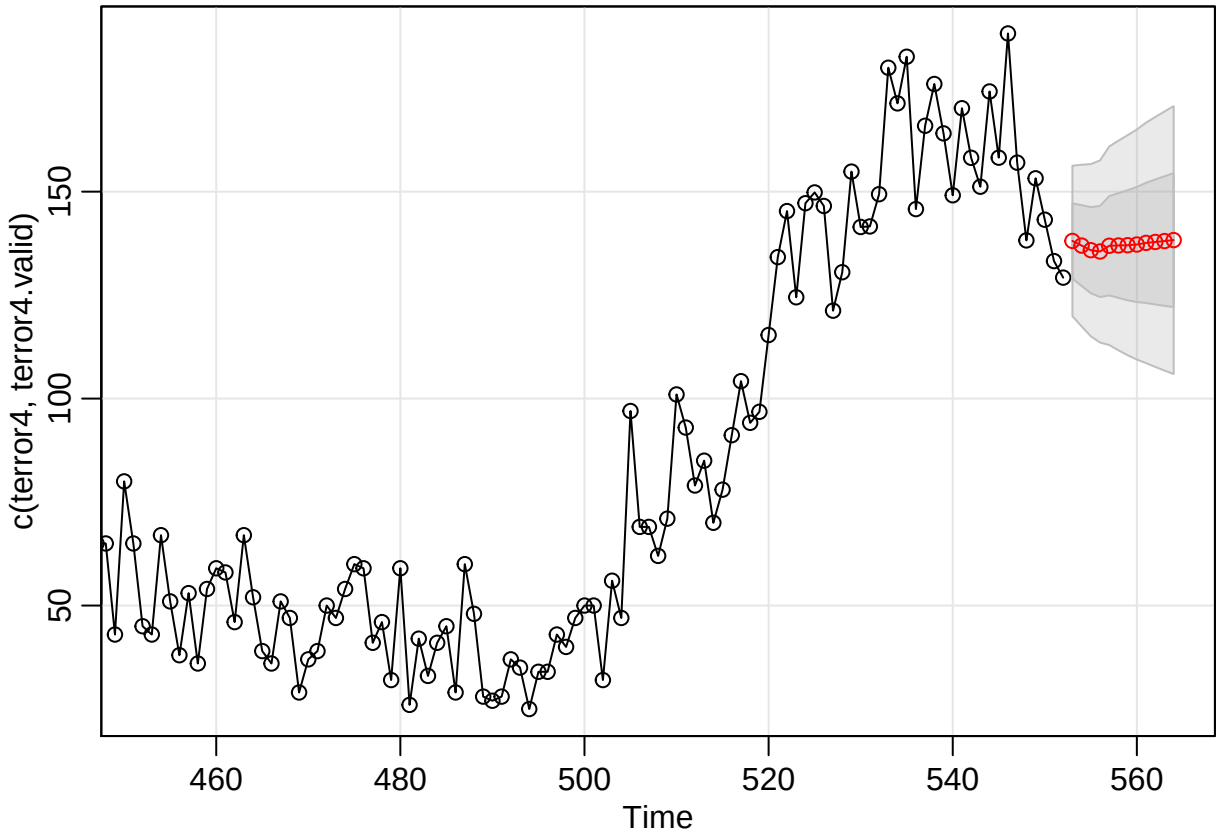

terror5
200
150
100
50
440
460
480
500
520
540
Time


terror5
200
150
100
50
440
460
480
500
520
540
Time

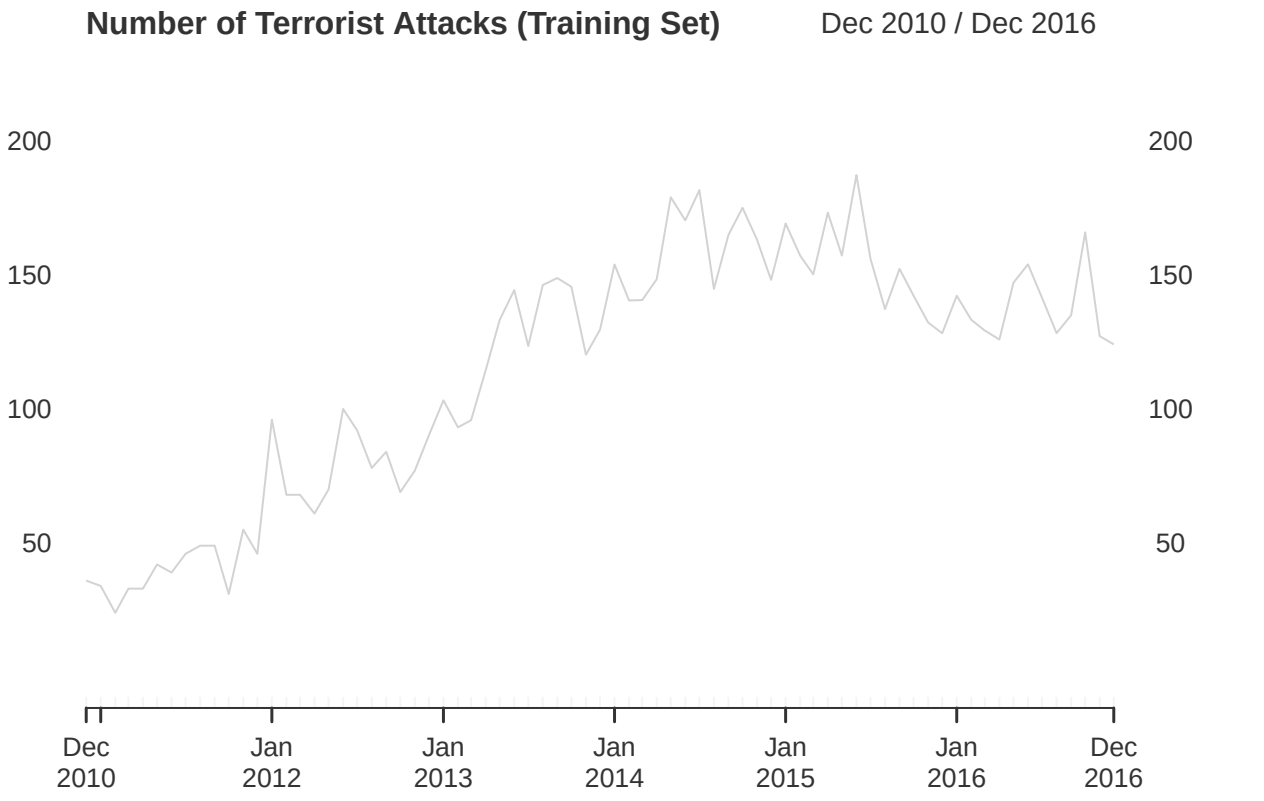

terror5
200
150
100
50
440
460
480
500
520
540
Time

```
mse <- total_error / (length(terror4.valid) - 11)
```

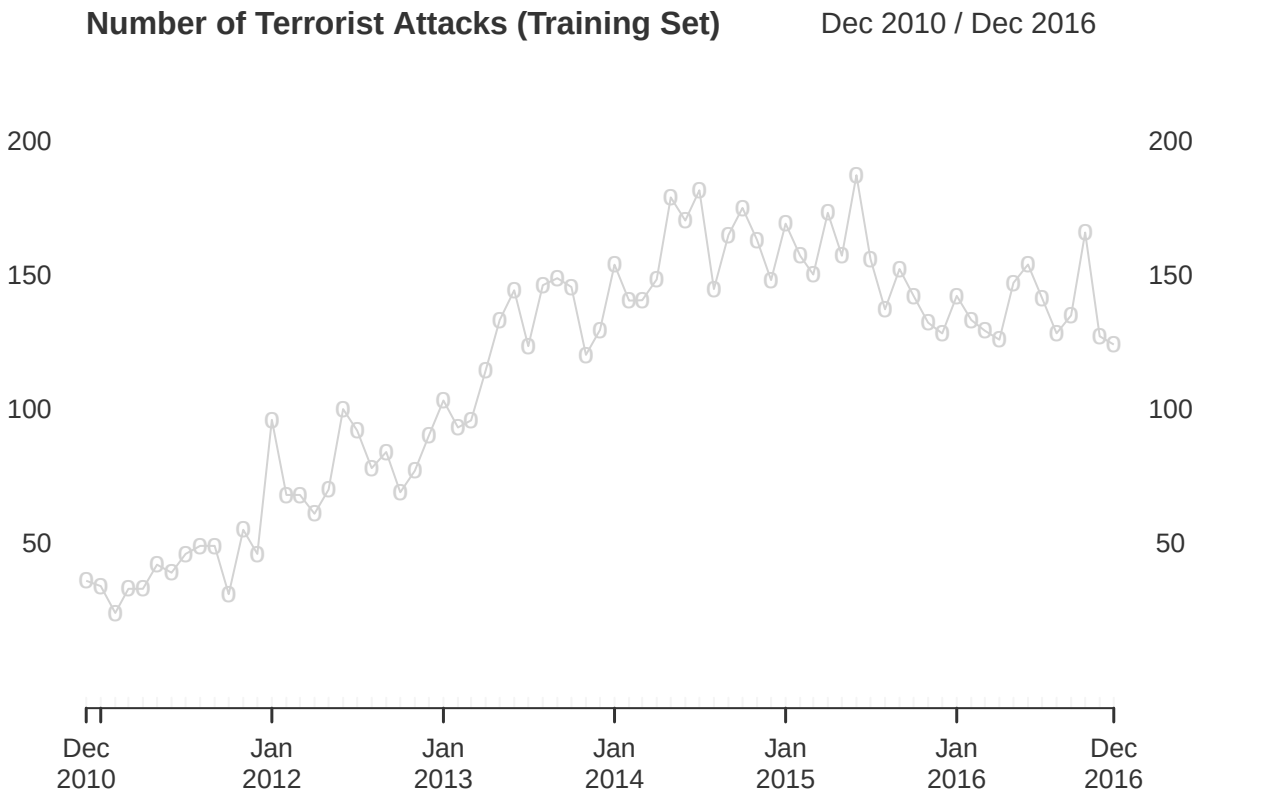
```
val <- sarima.for(c(terror4, terror4.valid), 12, 0, 1, 1, 1, 0, 0, 4)
```



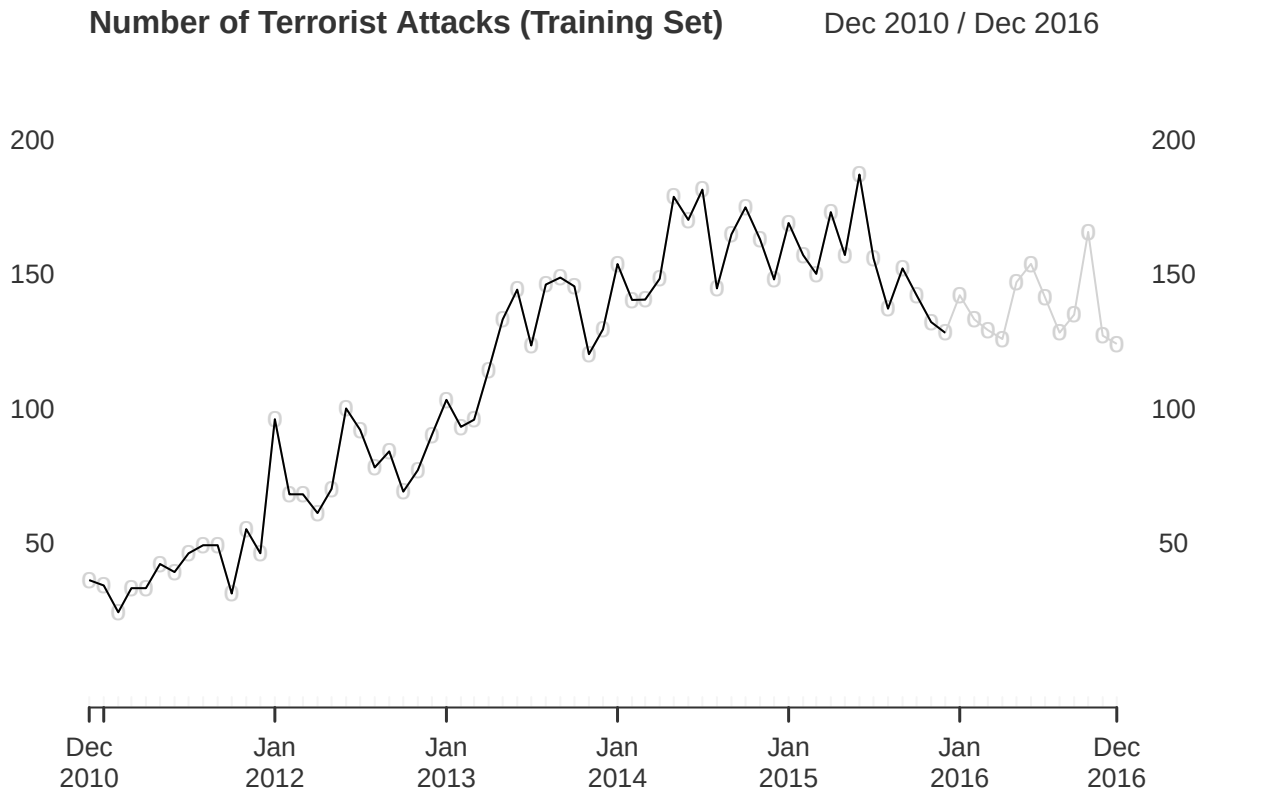

```
pred <-val$pred
err  <-val$se
total <- c(terror4, terror4.valid, terror4.testing)
plot(as.xts(ts(total, frequency = 12, start=1970))[492:length(total)], main = "Number of Terrorist Atta
```



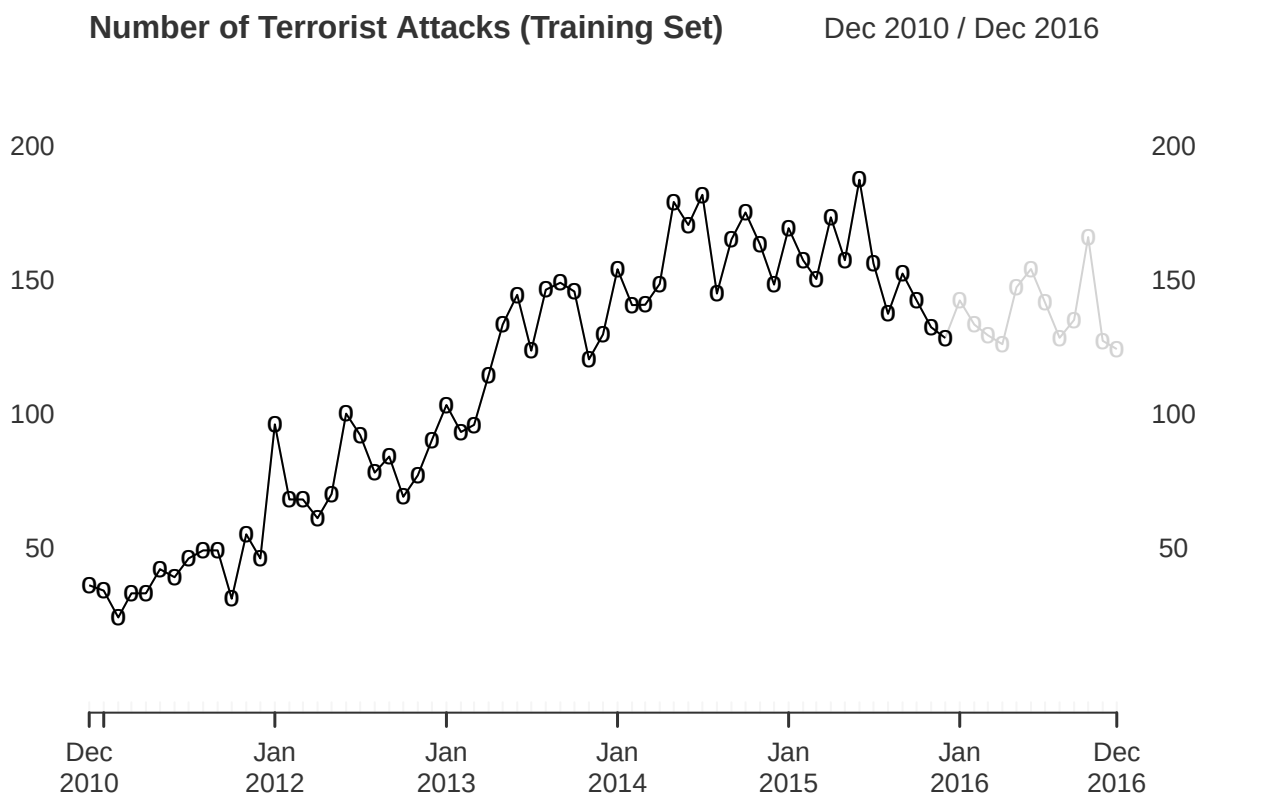

```
points(as.xts(ts(total, frequency = 12, start=1970)),col="lightgray",pch="o")
```



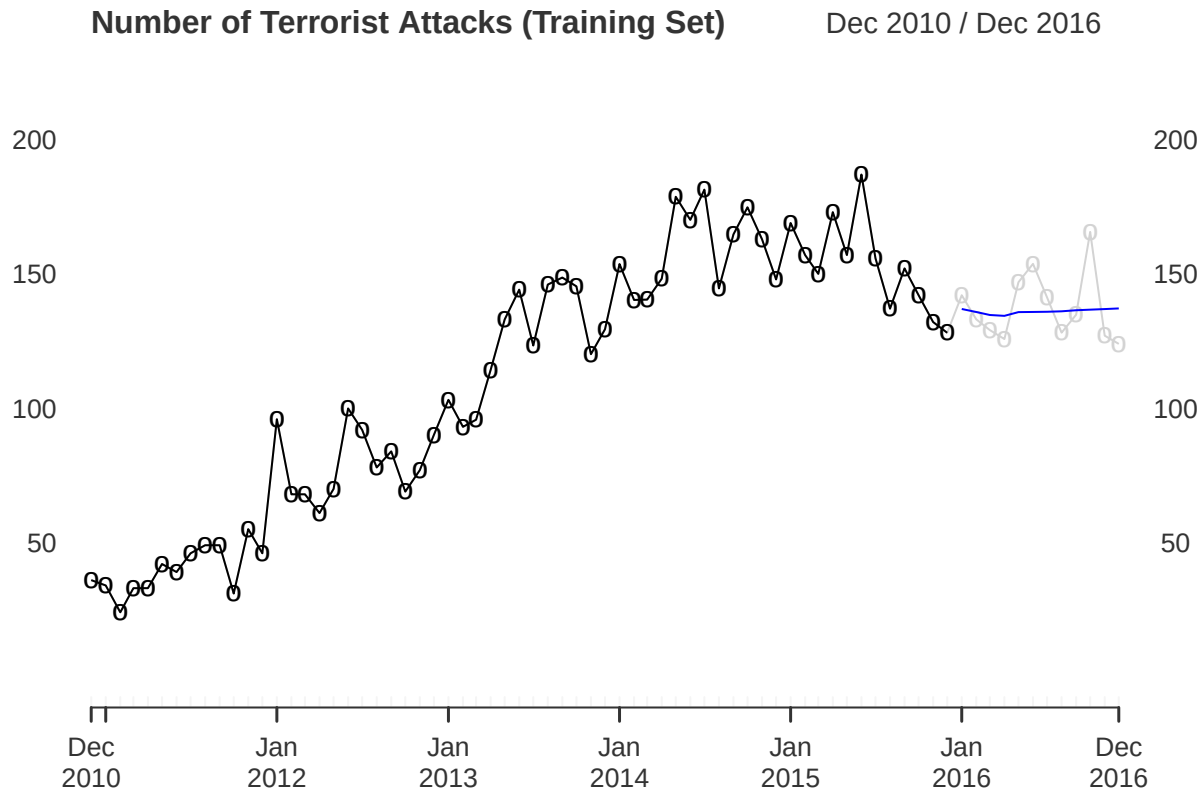
```
lines(as.xts(ts(c(terror4, terror4.valid), frequency = 12, start=1970)),col="black")
```



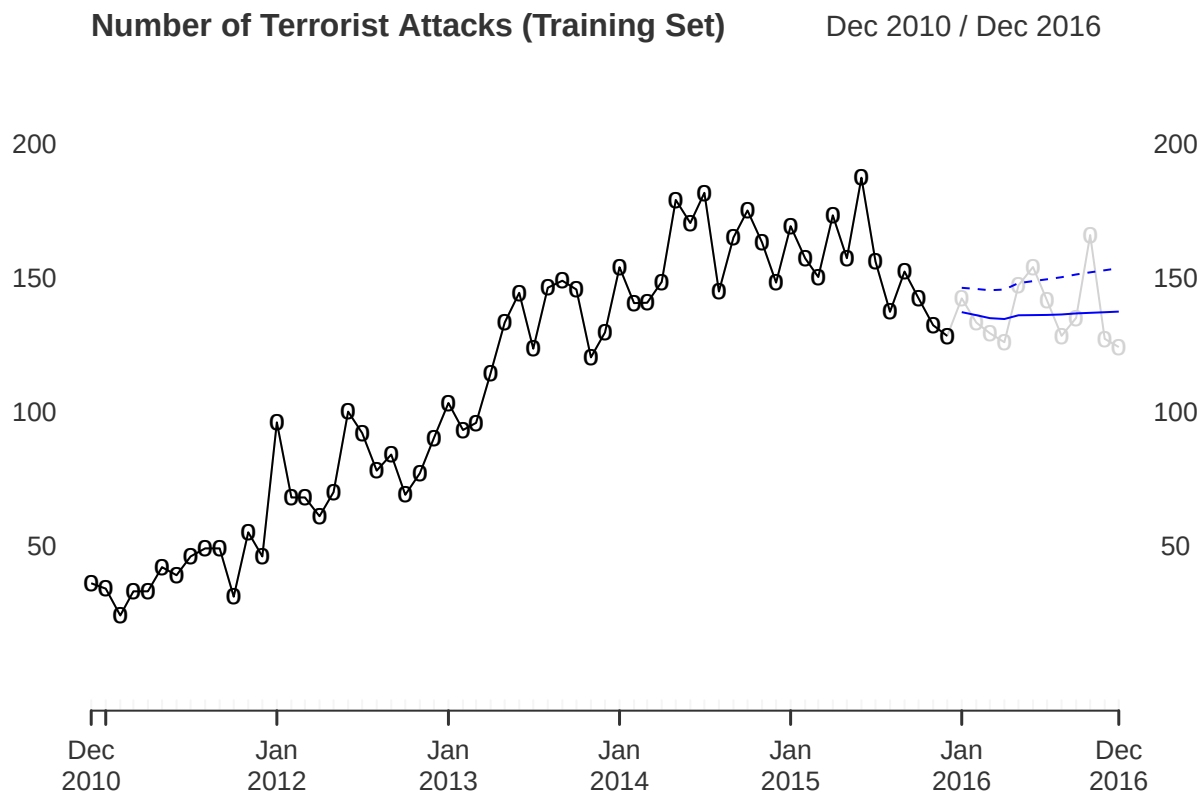

```
points(as.xts(ts(c(terror4, terror4.valid), frequency = 12, start=1970)),col="black",pch="o")
```



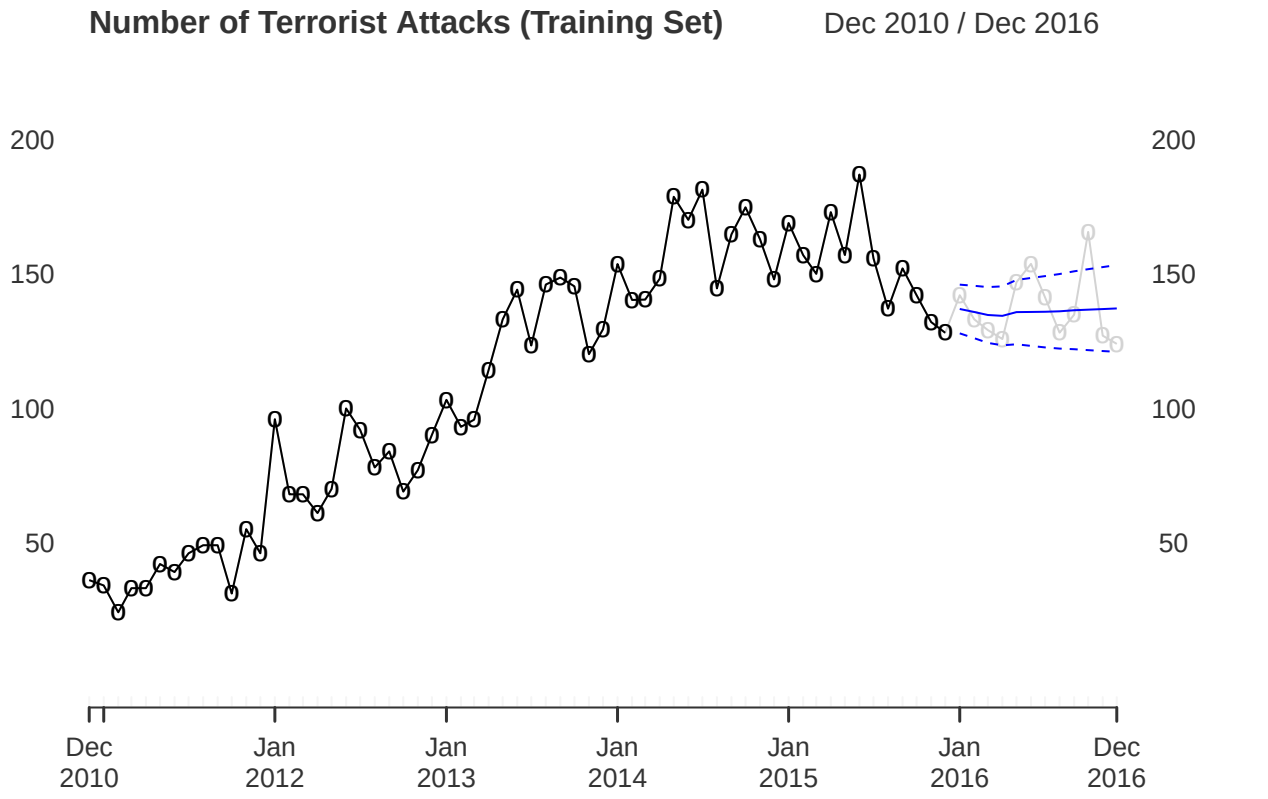
```
lines(as.xts(ts(pred, frequency = 12, start=2016)),col="blue")
```



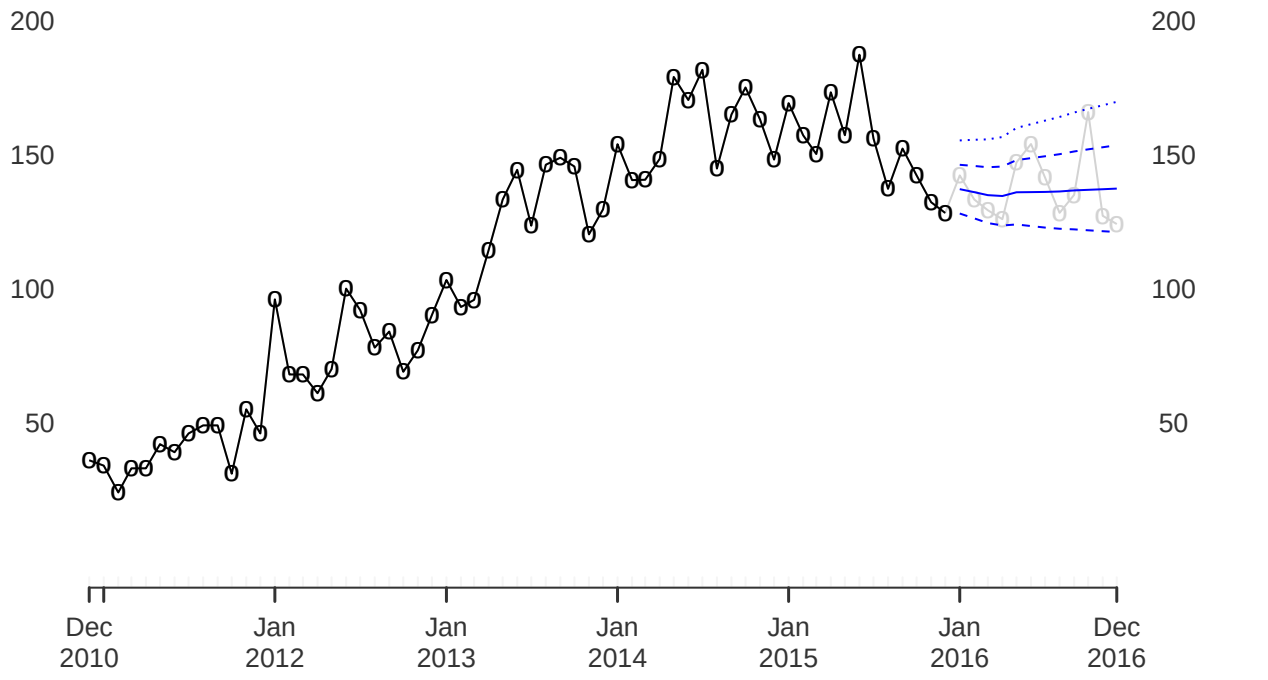

```
lines(as.xts(ts(pred + err, frequency = 12, start=2016)),col="blue", lty="dashed")
```



```
lines(as.xts(ts(pred - err, frequency = 12, start=2016)),col="blue", lty="dashed")
```




```
lines(as.xts(ts(pred + 2*err, frequency = 12, start=2016)),col="blue", lty="dotted")
```



```
lines(as.xts(ts(pred - 2*err, frequency = 12, start=2016)),col="blue", lty="dotted")
```