pre-assignment

Jakub Skrajny

27 04 2021

Libraries

```
library(openxlsx)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(forecast)
library(EnvStats)
```

```
##
## Attaching package: 'EnvStats'

## The following objects are masked from 'package:stats':
##
##     predict, predict.lm

## The following object is masked from 'package:base':
##
##     print.default
```

```
library(lmtest)
```

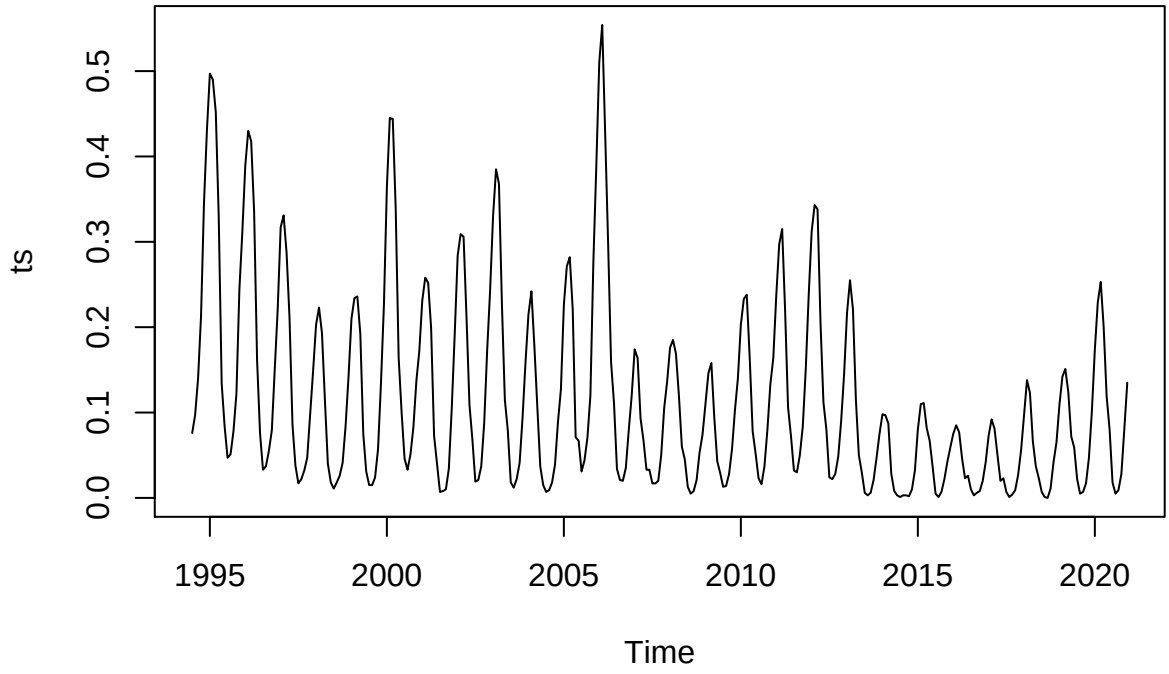
```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

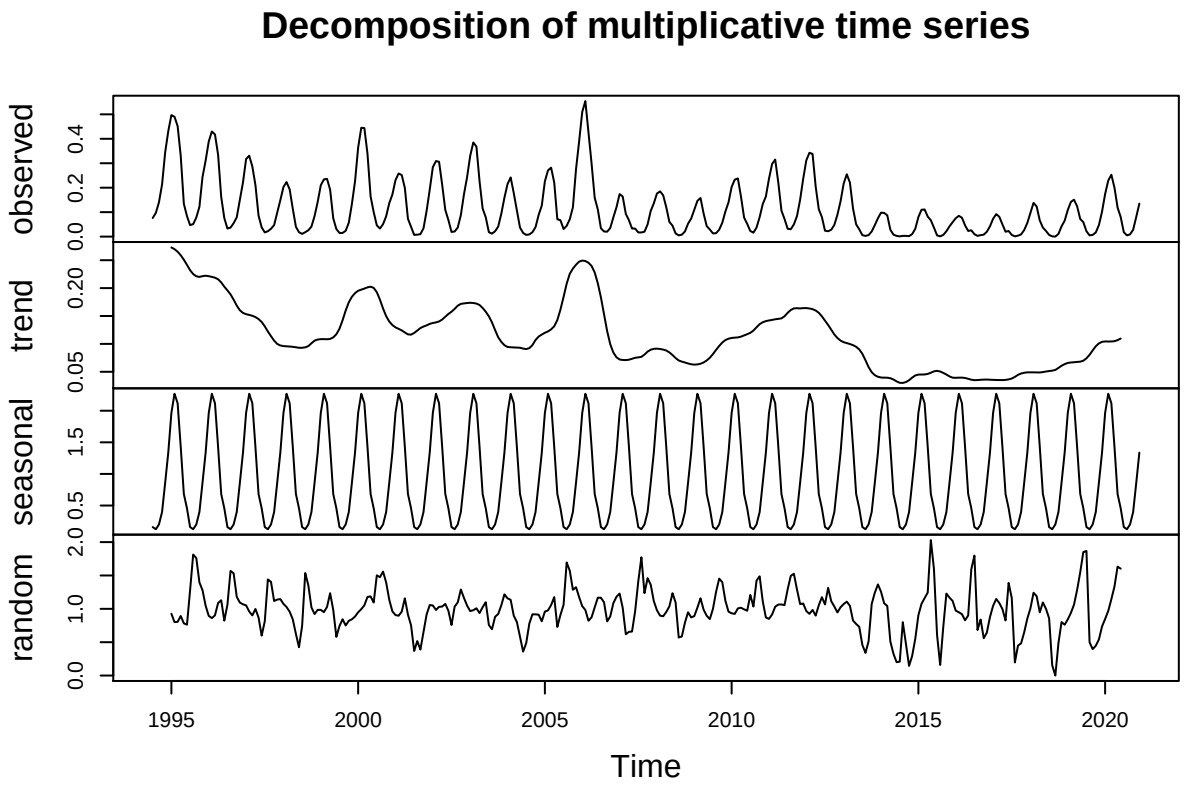
Read data

```
data <- read.xlsx("Lions_Den_data.xlsx")
ts <- ts(unlist(data[2]), start=c(1994, 7), frequency=12)
plot(ts)
```



Decomposition

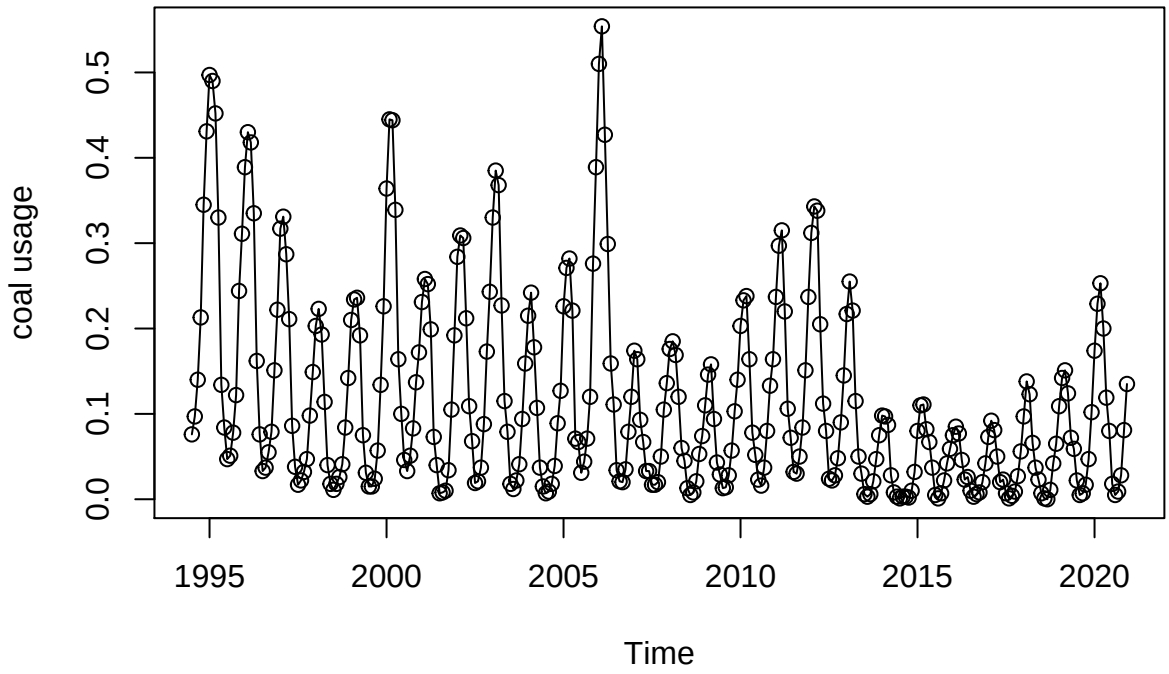
```
decompose <- decompose(ts, "multiplicative")
plot(decompose)
```



Outliers

Rosner's test suggest that there is only one outlier, but on the plot we can see that entire winter 2006 is does not match.

```
plot(ts, type="o", ylab="coal usage")
```



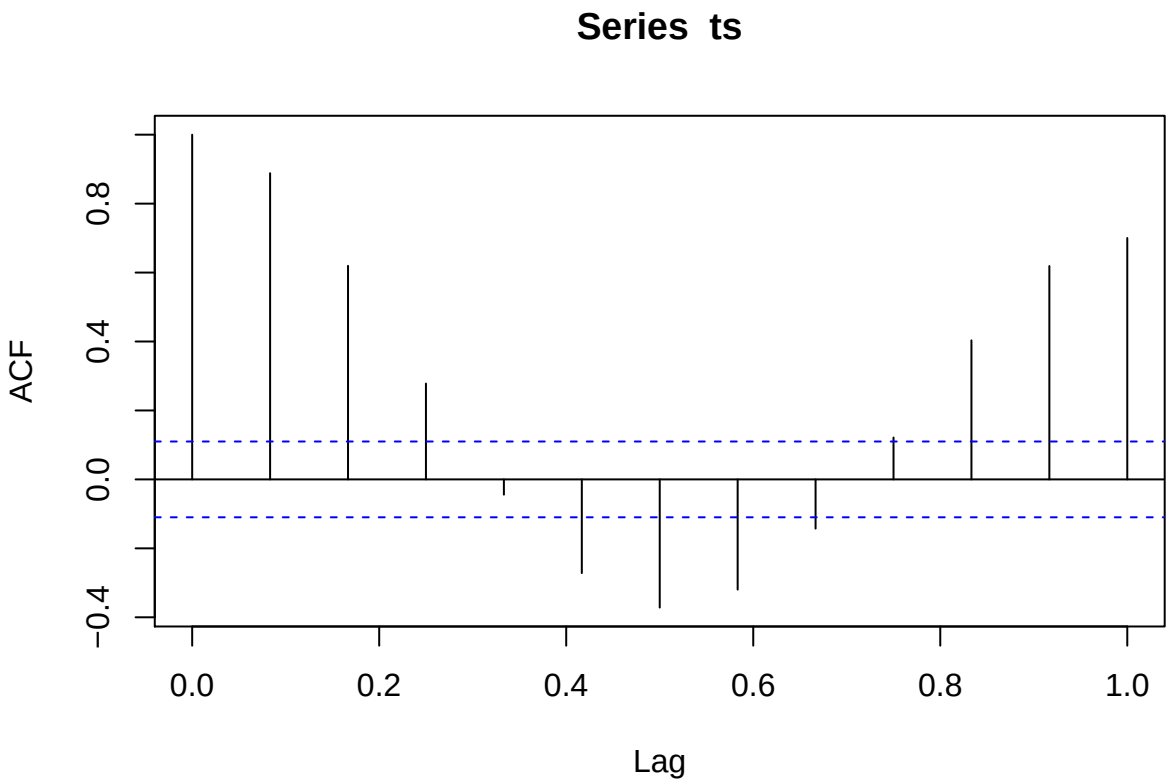
```r
rosnerTest(ts, k = 3)
```

```
##
## Results of Outlier Test
## -------------------------
##
## Test Method:                     Rosner's Test for Outliers
##
## Hypothesized Distribution:       Normal
##
## Data:                            ts
##
## Sample Size:                     318
##
## Test Statistics:                 R.1 = 3.782021
##                                  R.2 = 3.487550
##                                  R.3 = 3.445420
##
## Test Statistic Parameter:        k = 3
##
## Alternative Hypothesis:          Up to 3 observations are not
##                                  from the same Distribution.
##
## Type I Error:                    5%
##
## Number of Outliers Detected:     1
##
##   i    Mean.i      SD.i Value Obs.Num    R.i+1 lambda.i+1 Outlier
## 1 0 0.1170692 0.1155284 0.554     140 3.782021   3.739949    TRUE
## 2 1 0.1156909 0.1130619 0.510     139 3.487550   3.739067   FALSE
## 3 2 0.1144430 0.1110335 0.497       7 3.445420   3.738181   FALSE
```

Autocorrelation and stationarity analysis

We can see that time-series is already stationary and is slightly correlated with itself 12 month earlier.

```r
#autocorrelation
acf(ts, lag.max = 12)
```




```
#stationary test
adf.test(ts)
```

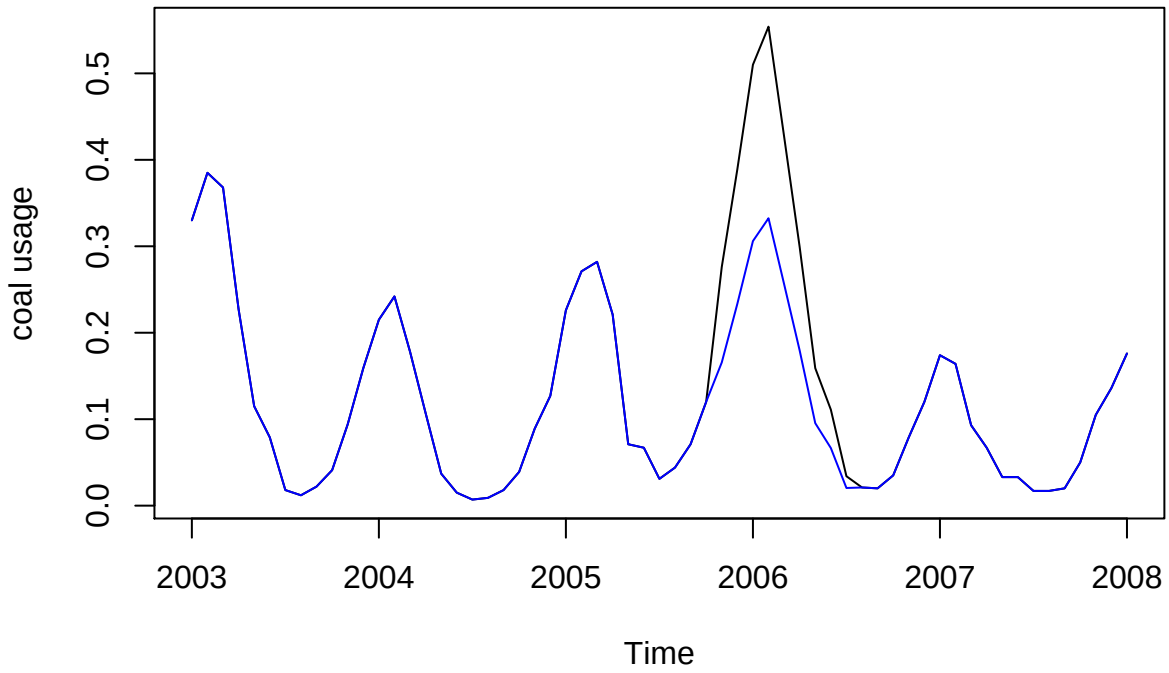
```
## Warning in adf.test(ts): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts
## Dickey-Fuller = -4.9571, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

Prediction

Delete outlier. Lower data from winter 2006.

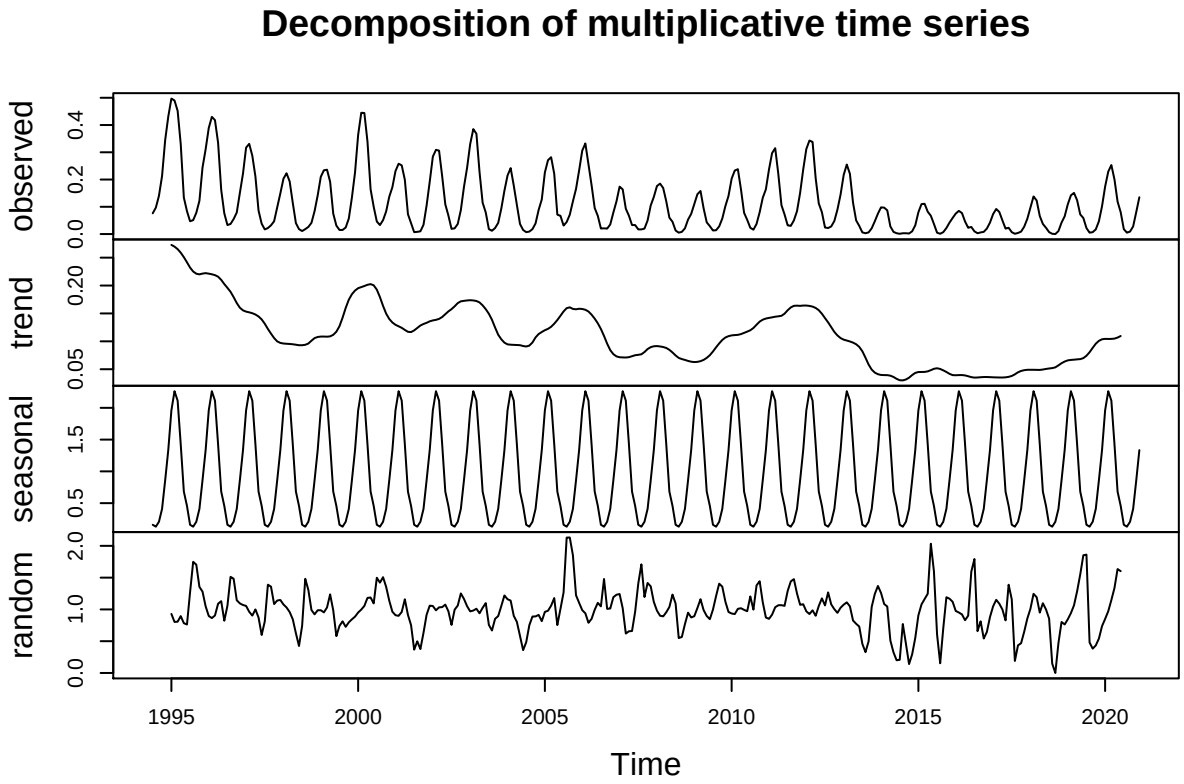
```
ts_ <- ts
plot(window(ts_, 2003, 2008), type="l", ylab="coal usage")
ts_[137:145] = ts_[137:145] * 0.6
lines(window(ts_, 2003, 2008), type="l", col="blue")
```



Per-

form new decomposition.

```
decompose_ <- decompose(ts_, "multiplicative")
plot(decompose_)
```



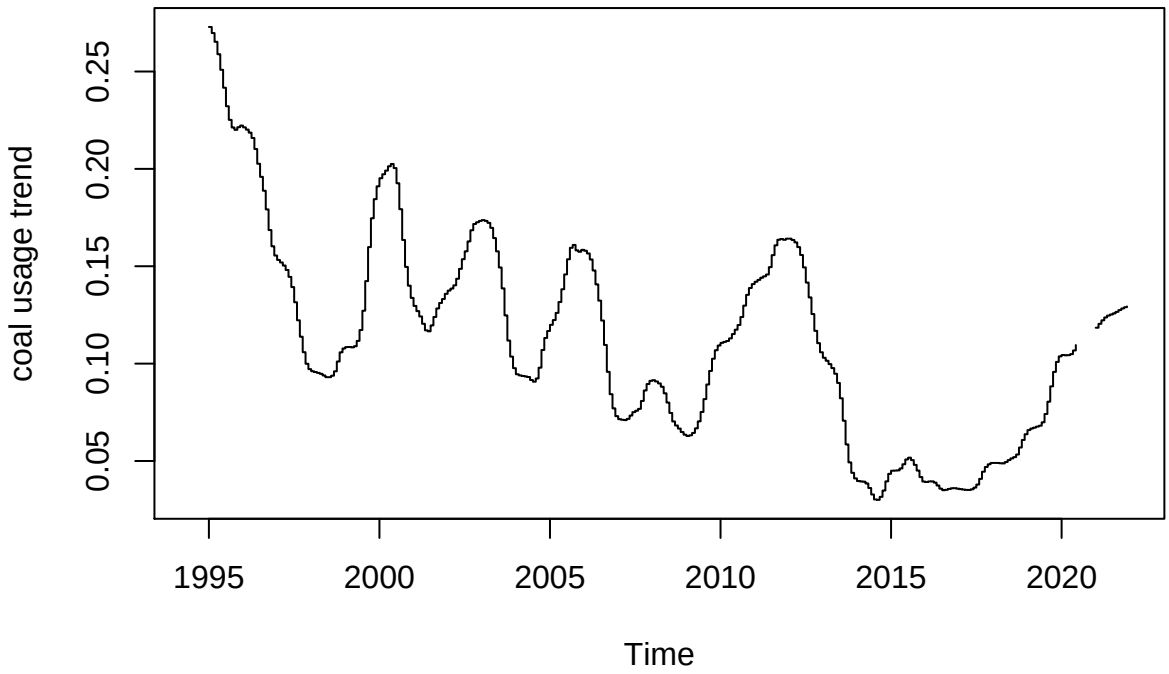
Pre-

dict trend.

```
tsTrend <- ts(decompose_$trend, start=c(1994, 7), frequency=12)
fitARIMA <- arima(tsTrend, order=c(1,1,1),seasonal = list(order = c(1,0,0), period = 12),method="ML")
coeftest(fitARIMA)
```

```
##
## z test of coefficients:
##
##        Estimate Std. Error  z value  Pr(>|z|)
## ar1   0.9188844  0.0221422  41.4992 < 2.2e-16 ***
## ma1   0.9999989  0.0085314 117.2143 < 2.2e-16 ***
## sar1 -0.3092937  0.0551587  -5.6073 2.055e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
yTrend <- predict(fitARIMA,n.ahead = 12)$pred
cTrend <- ts(
  c(tsTrend, yTrend),
  start=start(tsTrend),
  frequency=12
)
plot(cTrend, type="s", ylab="coal usage trend")
```



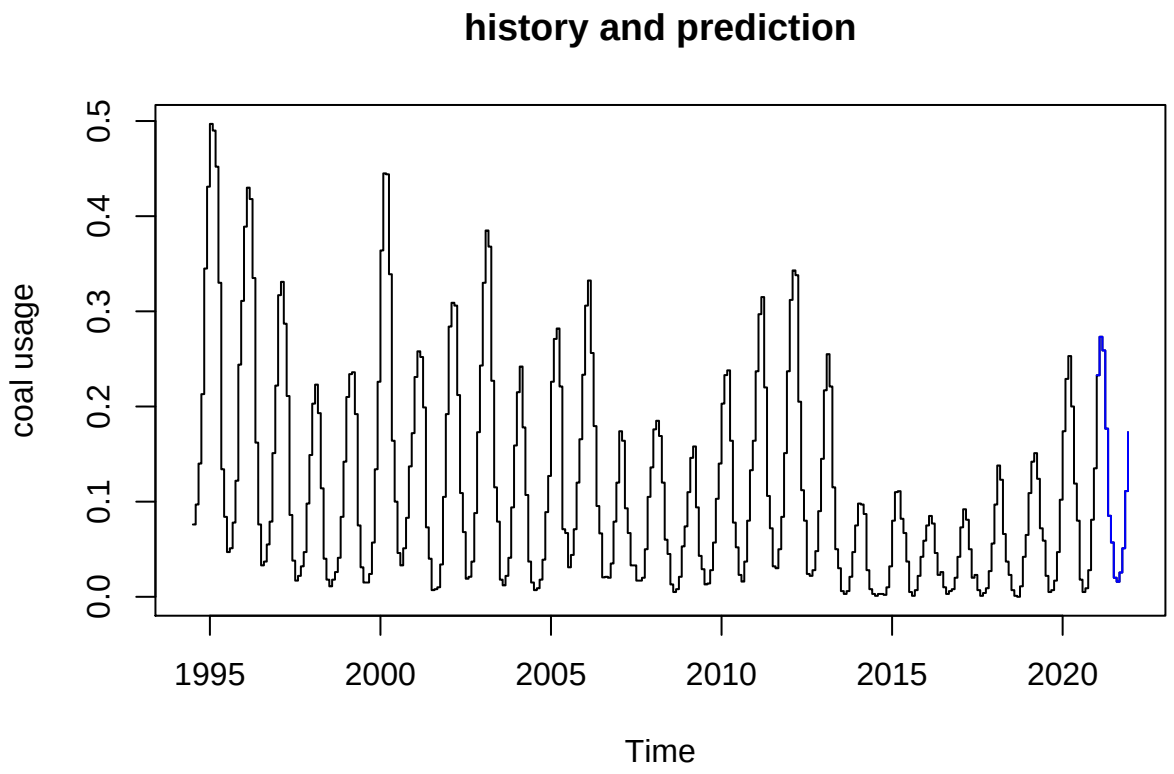
Extract seasonal.

```
ySeasonal <- window(decompose$seasonal, start=2020, end=c(2020,12))
```

Predict coal usage.

```
Y <- ts(as.numeric(yTrend) * as.numeric(ySeasonal),
        star=start(yTrend), frequency=12)
TS <- ts(c(ts_, Y), start=start(ts_), frequency=12)
```

```
plot(TS, type="s", ylab="coal usage",
     main="history and prediction")
lines(Y, type="s", col="blue")
```




```
Y
```

```
##             Jan        Feb        Mar        Apr        May        Jun
## 2021 0.23280717 0.27324848 0.25892658 0.17668279 0.08510665 0.05707731
##             Jul        Aug        Sep        Oct        Nov        Dec
## 2021 0.01999459 0.01580785 0.02552822 0.05100531 0.11110908 0.17305109
```