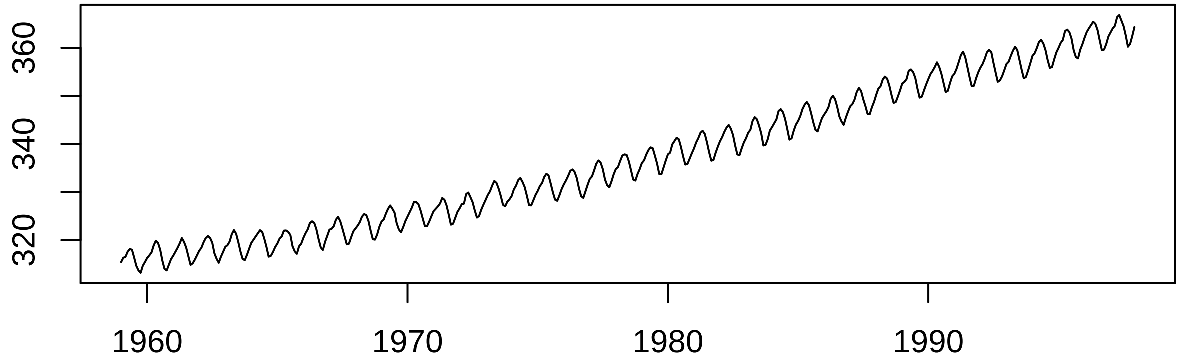
Exam - stochastic processes

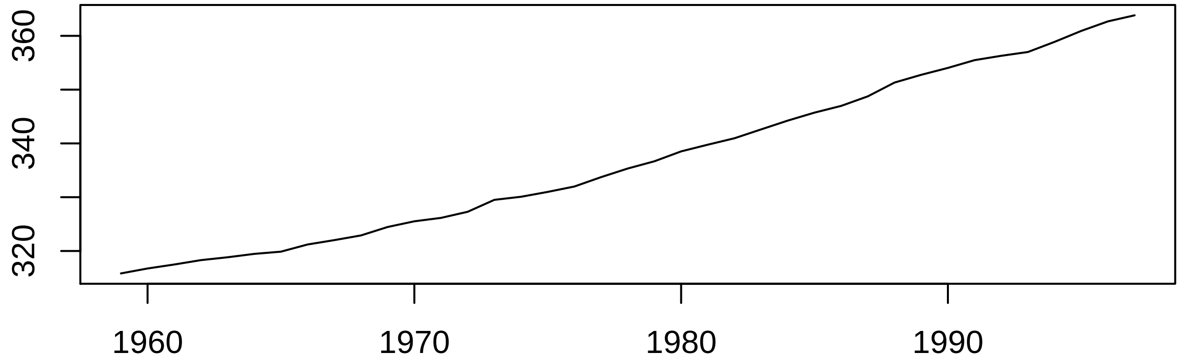
CO2 concentration in atmosphere

R's built-in dataset `co2` is a time series of atmospheric CO2 concentration at the Mauna Loa observatory. We will analyse this dataset below.

Explorative data analysis

- Start as always by plotting the data.

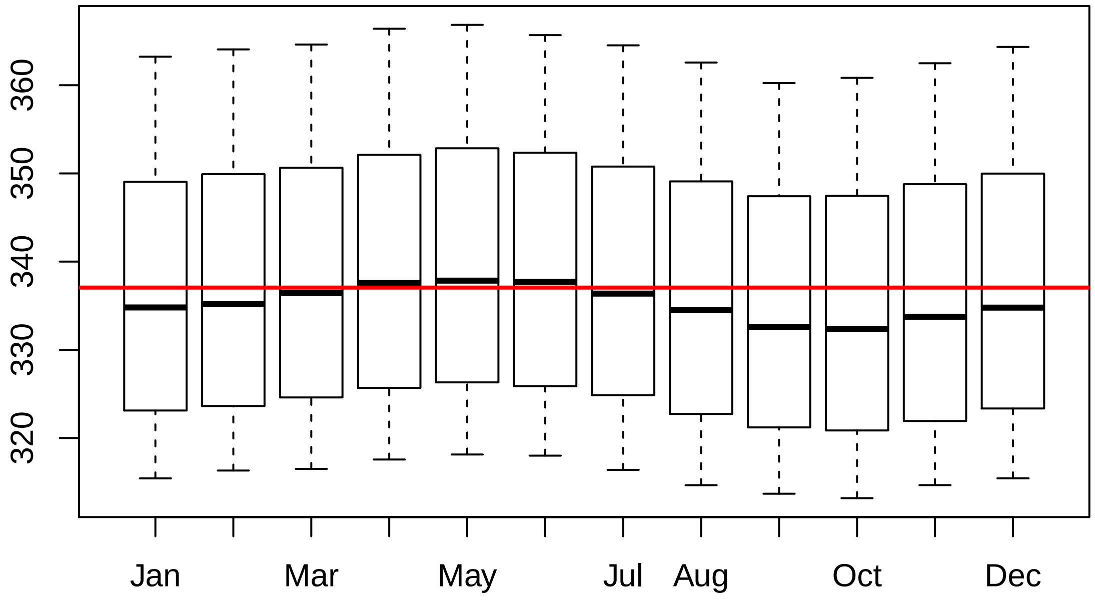




It seems that the CO2 concentration increases linearly with time.

Afterwards a monthly boxplot is shown in the next figure:

```
cyc <- cycle(CO2_month)
cyc <- factor(cyc, labels = month.abb)
boxplot(CO2_month ~ cyc)
abline(h=mean(co2), col = "red", lwd = 2)
```



This boxplot shows the stationality. The CO2 concentration increases a little bit on the first semester of the year, and decreases on the second semester. Although, the variation is very small.

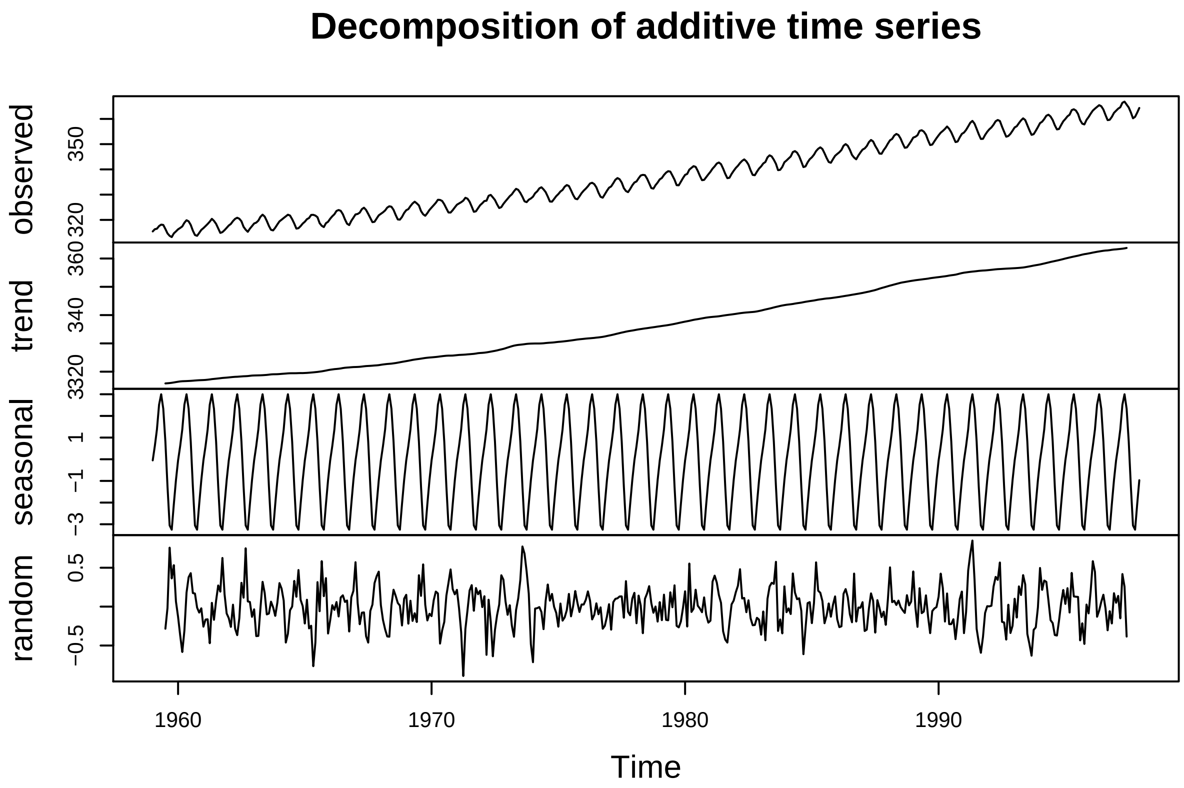
- Is this a second order stationary time series (explain what it means)?

To be a second order stationary time series the mean should be 0 and the variance should be constant in time. In this case we can clearly see that we have a trend (not repetitive sequnece) which means that the mean is not 0 (it is not a time series).

Looking to the different graphs the ramdomness or the error term could be the only parameter close to a second order satationary time series (mean~0).

- Use `decompose` to make a decomposition of the data to remove any trend and seasonal component and explain the method of how this is done.

```r
CO2_decomp <- decompose(CO2_month)
CO2_trend <- CO2_decomp$trend
CO2_no_trend <- CO2_month - CO2_trend
plot(CO2_decomp)
```

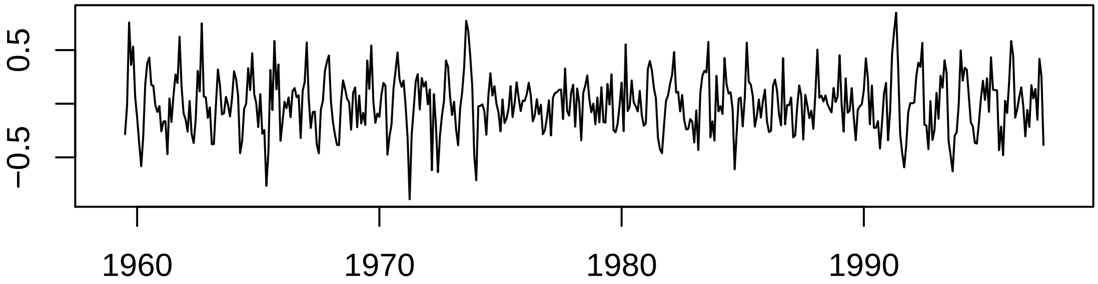


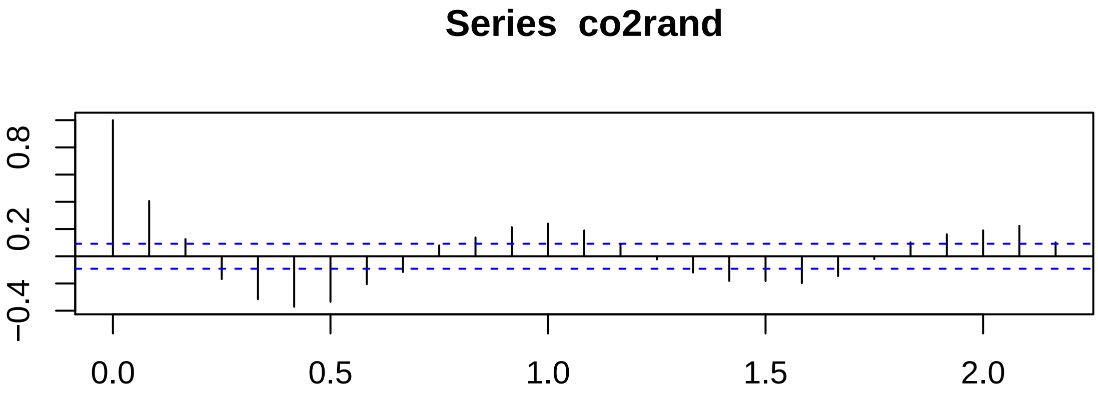
This method is based in the next equations:

$$x_t = m_t + s_t + z_t \quad \text{where } m_t : \text{trend, } s_t : \text{sesasonality and } z_t : \text{randomness}$$

- Save the random component as `co2rand` (omit any `NA` values), plot the correlogram of `co2rand` and explain the correlogram: What is it used for and how is it interpreted? What is assumed about the underlying process?

```r
par(mfrow = c(2,1), mar = c(3,3,3,3))
plot(CO2_decomp$random)
co2rand <- na.omit(CO2_decomp$random)
acf(co2rand)
```





It is a periodic correlogram, there is a sinusoidal seasonality going above the blue lines. In the lower part of the the blue lines there is a strong negative correlation whereas in the upper part of the limits there is a strong positive correlation.

Auto-regressive model of order 1

- Manually **fit a AR(1) model** using `lm` without an intercept:

```
n <- length(co2rand)
y <- co2rand[2:n]
x <- co2rand[1:(n-1)]
fitlm <- lm(y ~ x-1)
summary(fitlm)
```

```
##
## Call:
## lm(formula = y ~ x - 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.75763 -0.16162 -0.00027  0.15264  0.76485
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x  0.40841    0.04289   9.521   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.242 on 454 degrees of freedom
## Multiple R-squared:  0.1664, Adjusted R-squared:  0.1646
## F-statistic: 90.66 on 1 and 454 DF,  p-value: < 2.2e-16
```

- Explain the output of the last command above. Is there significant autocorrelation? What is the estimated lag 1 autocorrelation coefficient?

The summary command shows that there is a strong correlation between the actual value (y) and the previous one(x), as it can be seen in the p-value. Also, the F-test value is very high, what means that the actual model, is good model.

- Write down **the equation expressing the fitted model**.

$$x_t = \alpha_1 x_{t-1} + \epsilon_t$$

$$\text{being: autocorrelation coefficient } = \alpha_1 = 0.408$$

- Based on the data $x_1, \ldots, x_n$ **what is the predicted value** for x_{n+1}?

Based on the previous equation, and assuming that the error $\epsilon_t = 0$ *for predictions, the next expression can be obtained:*

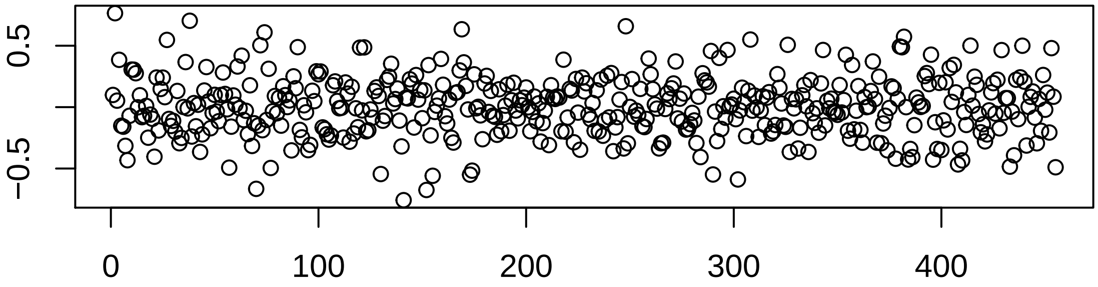
$$x_{n+k} = \alpha^k x_n$$

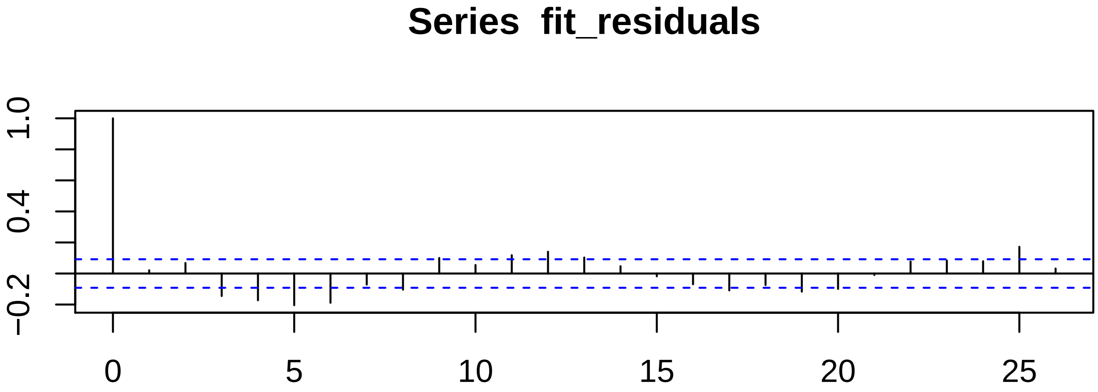
- What is the predicted value for x_{n+10}?

$$x_{n+10} = \alpha^{10} x_n$$

```
##            x
## -4.97182e-05
```

- Save the model residuals (use `residuals(fitlm)`) and **plot the correlogram**. What is the theoretical acf for this model? Is the AR(1) model a good fit?





The ACF we obtain is not white noise, asit is can be seen that there is some seasonality with the lags at 5,11,12,17. This shows us that the model is not fitting properly the randomness component.

Higher order autoregressive moving average (ARMA) models

- How do we **define higher order AR(p) processes**? *An AR(p) process is the one that uses several lag terms for explaining the current value*

  $$x_t = \alpha_1 x_{t-1} + \alpha_2 x_{t-2} + ... + \alpha_p x_{t-p} + w_t$$

  To ensure the model is stationary, all the roots of the characteristic equation must be found, and check that the absolute value of all of them is greater than 1.

  $$1 - \alpha_1 z - \alpha_2 z^2 - ... - \alpha_p z^p = 0$$

- Use `lm` as above to **estimate an AR(2) model** for `co2rand`. Is the lag 2 autocorrelation coefficient significant according to a `summary` of the fitted model?

```
n <- length(co2rand)
x <- co2rand[3:n]
x1 <- co2rand[2:(n-1)]
x2 <- co2rand[1:(n-2)]
fitlm2 <- lm(x ~ x1 + x2 - 1) # -1 is for removing the intercept.
summary(fitlm2)
```

```
##
## Call:
```

```
## lm(formula = x ~ x1 + x2 - 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.75166 -0.15658 -0.00056  0.14880  0.75108
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## x1  0.42982    0.04718   9.110   <2e-16 ***
## x2 -0.04974    0.04717  -1.054    0.292
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2422 on 452 degrees of freedom
## Multiple R-squared:  0.1688, Adjusted R-squared:  0.1651
## F-statistic: 45.89 on 2 and 452 DF,  p-value: < 2.2e-16
```

the predictor, X2 is not significant in this model. So, X2 (first value) doesnt have a strong correlation with the X (third value) due to the fact that the p value exceeds the 5%.

- **What is the partial autocorrelation function (pacf)** and how is it useful in relation to AR(p) processes?

The partial autocorrelation is the function that shows if there is any correlation between values that are not consecutive. The function is :

$$x_t = \beta_0 + \beta_1 * x_{t-1} + \beta_2 * x_{t-2} + \epsilon$$

If a time series has a strong correlation with the values lagged by one time step it most likely also has a trong correlation with the values lagged by two time steps: When today's value is highly influenced by yesterdays, then yesterdays will be highly influenced by the day before, and thus the lag 2 value influences the current value through the lag 1 value.

For a strong deterministic trend the pacf only shows strong correlation in the first lag and after controlling for this the following lags show no extra correlation

- How is a **MA(q) process** defined?

A moving average (MA) process is a finite sum of stationary white noise terms. It is itself stationary, what means that the mean and variance are time-invariant.

$$x_t = w_t + \beta_1 w_{t-1} + \beta w_{t-2} + ... + \beta_q w_{t-q}$$

where w_t is white noise process with mean zero and variance σ_w^2 , and $\beta_1, \beta_2, ..., \beta_q$ * are the parameters to be estimated.*

- Try to fit a collection of **ARMA(p,q) models for p and q at most 2, and find the best fitting model** one based on AIC. (Hint: If you called the models `ar1`, `ar2`, `ma1`, `ma2`, `arma11`, `arma12`, `arma21`, and `arma22` you can compare them all in a single call to `AIC`; `AIC(ar1, ar2, ma1, ma2, arma11, arma12, arma21, arma22)`)

```r
ar1 <- arima(co2rand, order = c(1,0,0))
ar2 <- arima(co2rand, order = c(2,0,0))
ma1 <- arima(co2rand, order = c(0,0,1))
ma2 <- arima(co2rand, order = c(0,0,2))
arma11 <- arima(co2rand, order = c(1,0,1))
arma12 <- arima(co2rand, order = c(1,0,2))
```

```
arma21 <- arima(co2rand, order = c(2,0,1))
arma22 <- arima(co2rand, order = c(2,0,2))
AIC(ar1,ar2,ma1,ma2,arma11,arma12,arma21,arma22)
```

```
##        df         AIC
## ar1     3    5.458649
## ar2     4    6.342985
## ma1     3   18.884945
## ma2     4   -3.220705
## arma11  4    6.947824
## arma12  5   -1.291071
## arma21  5  -89.839092
## arma22  6 -102.569646
```

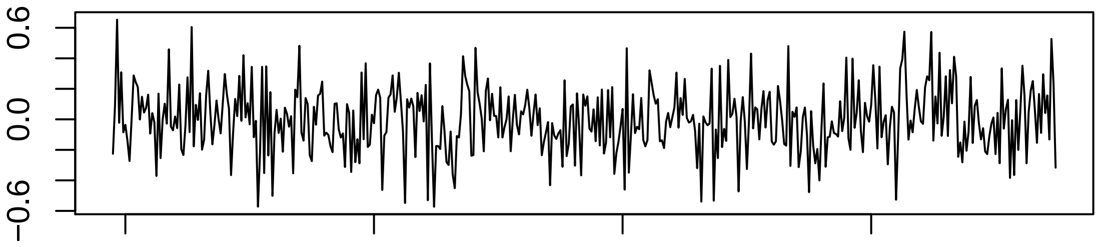
The best fitting one is the last arma22 which has an AIC of -102.569646. In arma22 you take in consideration the last 2 previous values and the last 2 white noise values

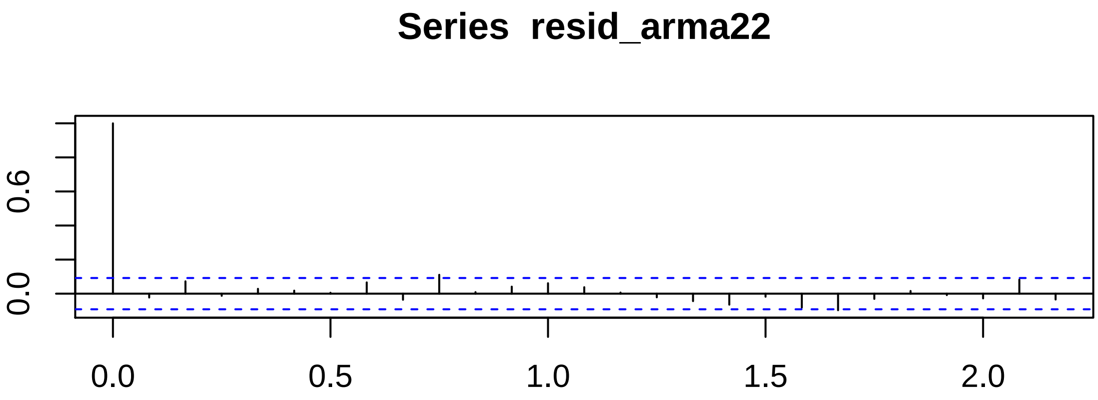
AIC the lower the better. The next function shows how the equation would look like:

$$x_t = \alpha_1 x_{t-1} + \alpha_2 x_{x-2} + w_t + \beta_1 w_{t-1} + \beta_2 w_{t-2}$$

- Write down **the parameter estimates of the final model**, and check whether it is a good fit to the data.

```
##
## Call:
## arima(x = co2rand, order = c(2, 0, 2))
##
## Coefficients:
##          ar1      ar2      ma1     ma2  intercept
##       1.4926  -0.7048  -1.2810  0.2810     -1e-03
## s.e.  0.0518   0.0460   0.0672  0.0665      3e-04
##
## sigma^2 estimated as 0.04498:  log likelihood = 57.28,  aic = -102.57
```





The ACF shows that there is no interaction or seasonality between the lags. the residuals from this model is white noise, what means that the model is a good fit to the data.

- Give **95% confidence intervals for the parameter estimates** of the final model selected by AIC.

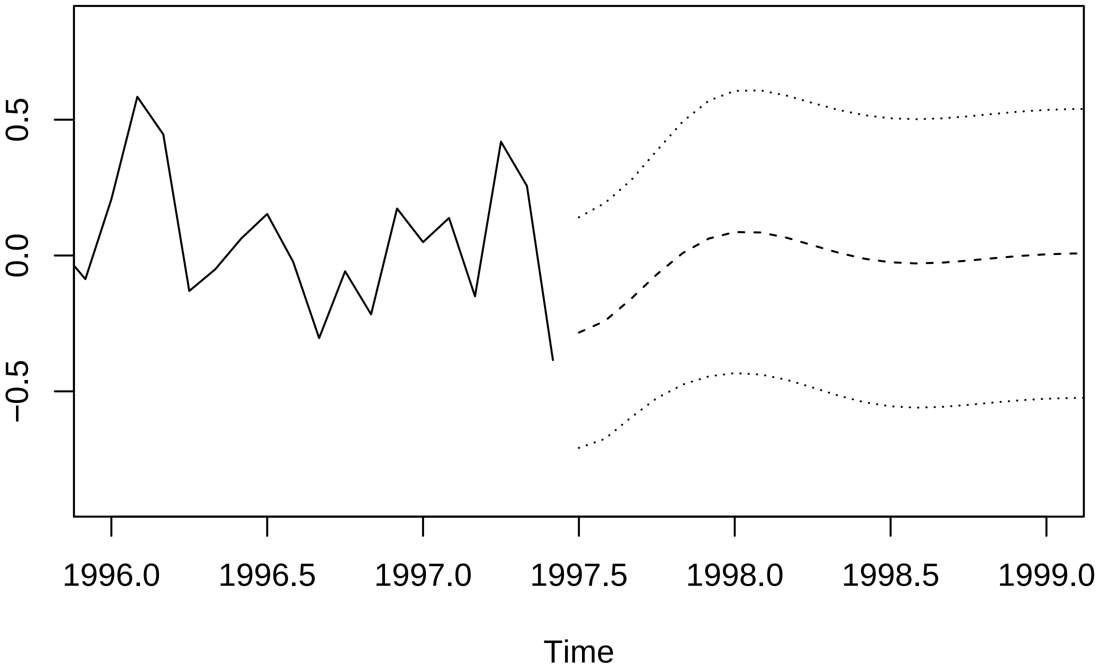
```
confint(arma22)
```

```
##                   2.5 %        97.5 %
## ar1        1.391075658  1.5942024899
## ar2       -0.794926723 -0.6147713101
## ma1       -1.412779479 -1.1492886567
## ma2        0.150722917  0.4113457013
## intercept -0.001467504 -0.0004511736
```

Prediction

- **Make a prediction with an approximate 95% prediction interval** for the next value of the random component x_{n+1} based on this model.

```
pred1 <- predict(arma22, n.ahead = 24)
lower1 <- pred1$pred-2*pred1$se
upper1 <- pred1$pred+2*pred1$se
ts.plot(co2rand, pred1$pred, lower1, upper1, lty = c(1,2,3,3), xlim=c(1996,1999))
```



As we can see, the predicted value goes to zero, when we increase the prediction time.

- **If there was no autocorrelation** in the random component the approximate 95% prediction interval is given by $\bar{x} \pm 2\sqrt{s^2(1 + 1/n)}$. Calculate this prediciton interval and compare it with the one obtained above. What is the difference? Try to explain this.

```
## [1] -0.2636535  0.2671404
```

This confidence interval doesn´t take into account the previous values of the time series, but only a summary of them (by the mean value and the standard error).

This way of getting the confidence interval is easier than the one based in the ARMA model, but it is also less accurate.