[II.2313] Analyse de données – TP6

isep
École d'ingénieurs du numérique

Rémi Biolley – Thierry Lincoln

A. Stationarity analysis

1. Install and load the *tseries* library, then load the *USeconomic* data using the command line data(USeconomic).

We want to study the evolution of GNP in the United States between 1954 and 1987. To do so, we'll need to use tools specific to the time series.

We import the needed package, then we import the data *USeconomic* contained in this package thanks to the *data()* function:

```
install.packages("tseries")
library(tseries)
data(USeconomic)
```

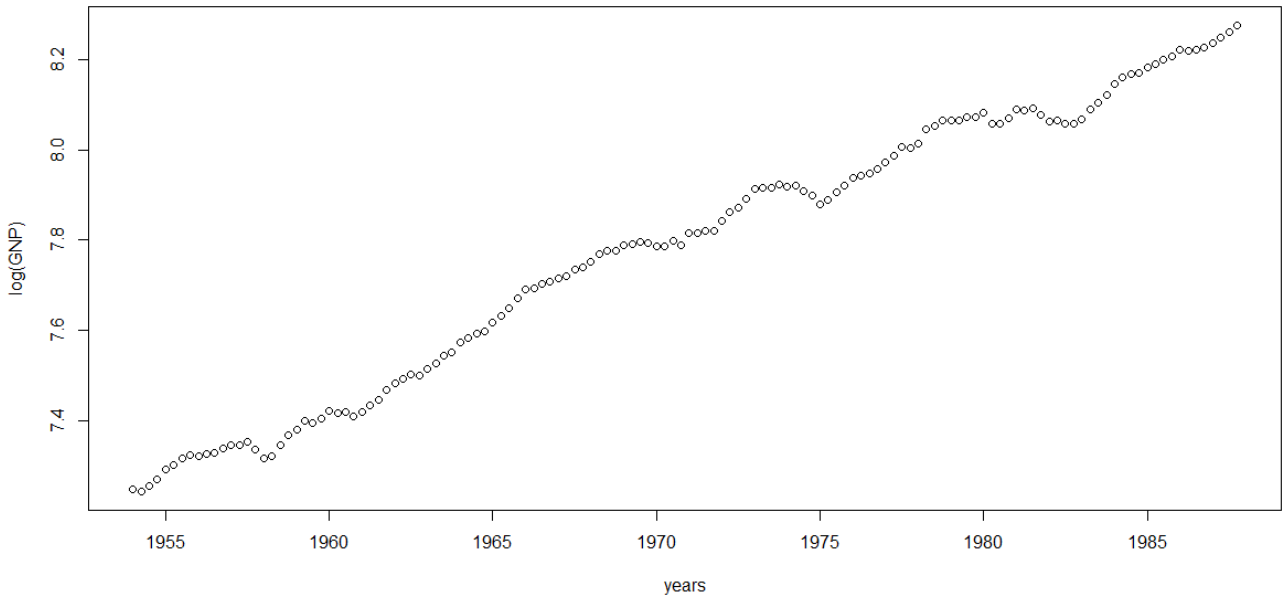
2. We will now create the variables that we are going to study

We store the logarithm of the time series *GNP* in a new variable that we call *clnGNP*. Using the logarithm will ease our later interpretations and helps us to stabilize the variance of the time series.

The data have been acquired on a trimestral basis, therefore we're going to create a variable that will contain every quarter from 1954 to 1987.75 (the 3rd quarter of 1987).

With these variables, we can represent *log(GNP)* depending on the quarters. We'll analyze the resulting graph in the next question.

```
clnGNP=log(GNP)
years=seq(from =  1954 , to = 1987.75,by=0.25)
plot(years,log(GNP))
```



3. Remind what "stationarity" means for a time series. What can you visually say about the stationarity of the log(GNP) time series.

Strict Stationarity: A time series is said to be strictly stationary if all its observations are drawn from the same distribution: the join probability does not change in time.

Weak Stationarity: We do not require that each draw comes from the exact same distribution, only that the distributions have the same mean and variance (all of them not a function of time).

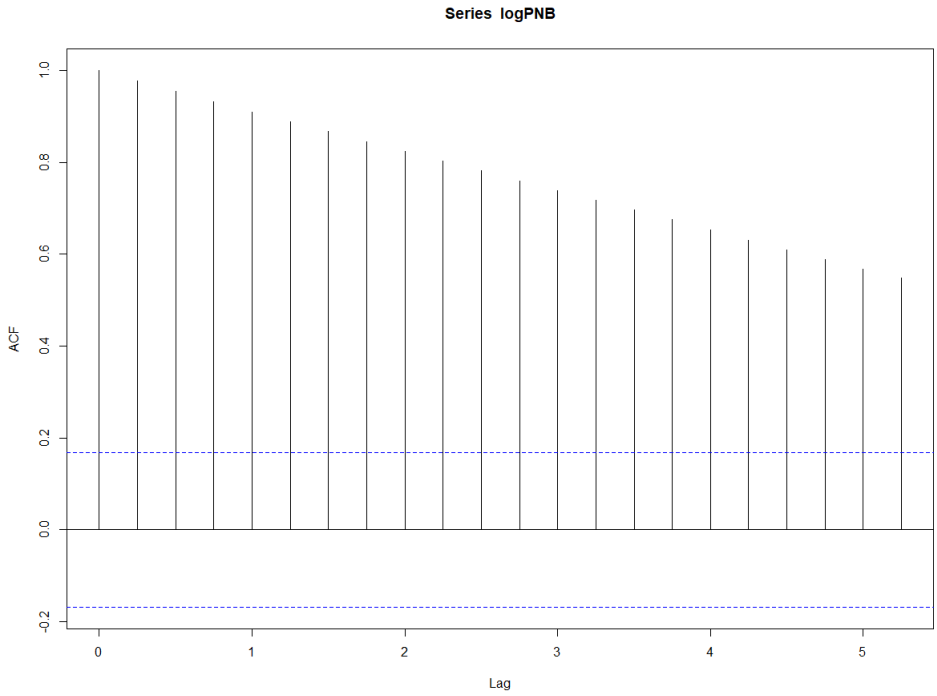
- Constant mean: $E(x_t) = \mu$
- Constant variance: $Var(x_t) = \gamma 0$
- Constant co-variance: $Cov(x_t, x_{t-h}) = \gamma h \, \forall h \in [1..T]$

What can you visually say about the stationarity of the log(GNP) time series?

We can assume that this time series is not Stationary, clearly the mean is not constant over time.

4. Use the acf command to draw this series autocorrelogram. Comment.

The *acf()* function allows us to visualize the autocorrelation of the series.



The obtained graph shows the autocorrelation values depending on the lag (the shift). The values are considered significant only if they are contained in the confidence interval (represented here by the blue dash lines).

We notice that all the autocorrelations are significant. They tend to decrease when we increase the lag (from ~0.98 for a lag of 1, to ~0.6 for a lag of 5.25), which was highly expected because of the variation of *log(GNP)*.

Nevertheless, the autocorrelation values aren't decreasing exponentially. This means that our time series may need to be differentiated so we may get more interesting information.

Furthermore, the autocorrelations beyond lag 1 may be due to the propagation of autocorrelation at lag 1.

5. Use the Box.test function on your series and interpret the result.

We want to evaluate the correlation between the residuals of our time series with a lag of 1. To do so, we use the *Box-Pierce test*:

```
Box.test(clnGNP,lag=1,type="Box-Pierce")
    Box-Pierce test
data:  clnGNP
X-squared = 129.98, df = 1, p-value < 2.2e-16
```

The p-value is <0.05 at 2.2e-16. This is a very low number; therefore, we can conclude that there is a serial correlation in the time series, with a low risk of being mistaken.

Unfortunately, we need independent residuals in order to build a good model. As a result, the study of the values of the GNP doesn't seem to be an acceptable solution.

6. What can you conclude on this time series stationarity?

This time series is not stationary. Therefore, it should not be used to create an acceptable model.

We may try to differentiate our time series to stabilize the mean, by reducing the trend.

B. Study of diffGNP

1. **Create a variable DiffGNP so that: $(Y_t = X_t - X_{t-1})_{2\leq t\leq T}$ where Xt is the logGNP at time t. Explain what this time series represents.**

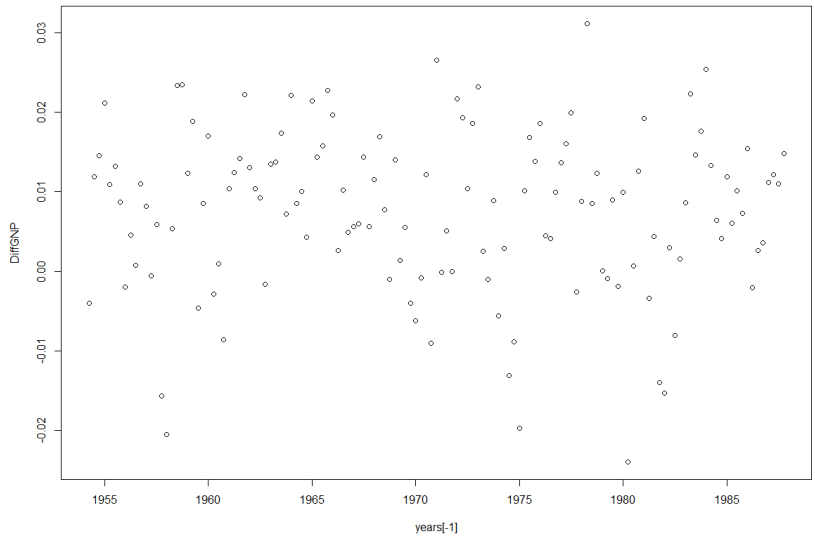
```
DiffGNP = diff(logGNP, lag=1)
```

We differentiate our time series as we proposed earlier. The new time series that we get shows how the GNP evolves from one quarter to another.

2. **Plot the evolution of this series between 1954 and the 3rd semester of 1987.**

Now that we have our differentiated time series, we can visualize it depending on the quarters.

```
plot(years[-1], DiffGNP)
```



The obtained graph is messier than the representation of our first time series. It is hard to conclude on the stationarity of the differentiated time series. We'll have to proceed several tests.

3. Is this series centered? You can use the empirical mean value of the series and a Student test to justify your answer.

We want to see if our series is centered. We compute the mean of it and proceed to a Student Test.

```
> mean(DiffGNP)
    [1] 0.007596586
> t.test(DiffGNP, mu=0)


    One Sample t-test
    data:  DiffGNP
    t = 8.6739, df = 134, p-value = 1.223e-14
    alternative hypothesis: true mean is not equal to 0
    95 percent confidence interval:
    0.005864407 0.009328764
    sample estimates:
    mean of x
    0.007596586
```

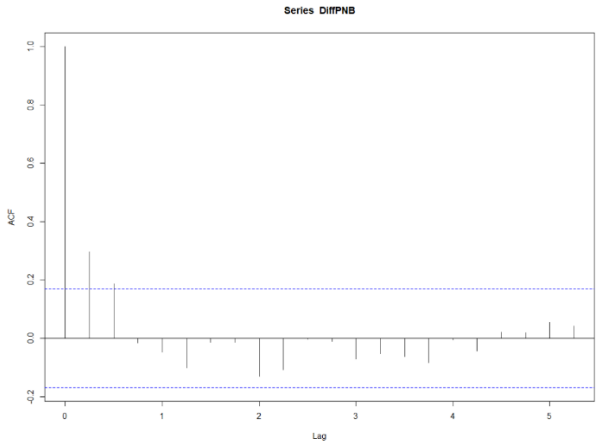
The mean of our series is equal to 0.0076, which means it is not centered. The Student test confirms that the mean is between 0.0059 and 0.0093 with a confidence of 95%. Furthermore, the p-value is below 0.05, so we can say, without too high risks, that the true mean isn't equal to 0.

Therefore, the time series is not centered (almost). We consider that the result is satisfying enough to perform a modelization on it.

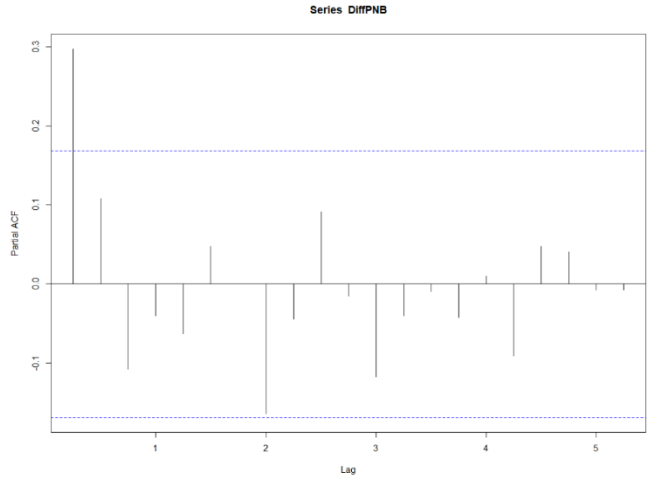
3. Use the acf and pacf functions to draw the autocorrelogram and partial autocorrelogram of this time series. From there, deduce the most likely parameter(s) p and q for an ARMA(p,q) model to modelise DiffGNP.

We need to determine the parameters to use in our ARMA model. To do so, we need to visualize the autocorrelogram and the partial autocorrelogram. The partial autocorrelogram,compared to the simple autocorrelogram, controls the shorter lags.

```
acf(DiffGNP)
```




```
pacf(DiffGNP)
```



We determine the possible parameters by looking at the peaks that are outside the interval (the significant values).

The PACF gives the p parameter, while the ACF gives the q parameter. We take the following parameters on: (0, 1) / (0,2) / (8,2) / (8,1).

5. Test all the couples (p,q) that seemed relevant to you from the previous question. The function arima(DiffGNP,c(p,0,q)) will help you to evaluate each model.

Let's evaluate the parameters we chose by computing their corresponding ARIMA model. The *arima()* function takes a vector with 3 values as a parameter. These values are respectively representing p, the number of differentiations to apply and q.

```
>arima(DiffGNP,c(0,0,1))

Call:
arima(x = DiffGNP, order = c(0, 0, 1))

Coefficients:
         ma1  intercept
      0.2278     0.0076
s.e.  0.0707     0.0010

sigma^2 estimated as 9.578e-05:  log likelihood = 433.03,  aic = -860.05

>arima(DiffGNP,c(0,0,2))

Call:
arima(x = DiffGNP, order = c(0, 0, 2))

Coefficients:
         ma1     ma2  intercept
      0.2681  0.1976     0.0076
s.e.  0.0851  0.0790     0.0012

sigma^2 estimated as 9.178e-05:  log likelihood = 435.86,  aic = -863.73

>arima(DiffGNP,c(8,0,1))

Call:
arima(x = DiffGNP, order = c(8, 0, 1))

Coefficients:
         ar1      ar2     ar3     ar4      ar5     ar6      ar7      ar8      ma1
 intercept
      0.3894   0.1143 -0.1127  -0.0262  -0.0850  0.0782   0.0372  -0.1601   -0.1208
0.0075
s.e.  0.3759   0.1362  0.1180  0.1094   0.1005  0.1047   0.1195   0.0975   0.3312  0.2
354

sigma^2 estimated as 8.646e-05:  log likelihood = 438.49,  aic = -854.98


>arima(DiffGNP,c(8,0,2))

Call:
arima(x = DiffGNP, order = c(8, 0, 2))

Coefficients:
         ar1      ar2     ar3     ar4      ar5     ar6      ar7      ar8      ma1
ma2  intercept
      0.3857  -0.4613  0.0457  0.0523  -0.1347  0.0604  -0.0039  -0.1393  -0.1172  0.6
047     0.0075
s.e.  0.3313   0.2883  0.1180  0.1094   0.1005  0.1047   0.1195   0.0975   0.3312  0.2
354     0.0010

sigma^2 estimated as 8.646e-05:  log likelihood = 439.66,  aic = -855.32
```

Explanation of the Log likelihood and Akaike information criterion:

In our case study here, we want to find the best parameter for the ARIMA model to fit our data. Thus, we need to evaluate different parameters.

- **The log likelihood:**

The purpose of this paragraph is to figure out what's the likelihood, and why we use the logarithmic form

Our Arima model has more than one unknown parameter but for the conciseness of our explanation let's consider a model with only one parameter θ

Definition: The Likelihood is a tool for summarizing the data's evidence about unknown parameters, or simpler-said: the *Likelihood* describes the plausibility of a model parameter value, given specific observed data.

Formalizing may help in understanding the concept:

Let f be the function of a distribution. Because this function depends on the θ parameter we write it as such $f(x, \theta)$.

The likelihood function is defined as:

$$L(\theta, x) = \prod_{i=1}^{n} f(X_i, \theta)$$

In many problems, we will derive our loglikelihood from a sample rather than from a single observation. If we observe an independent sample $x_1, \dots, x_n$ from a distribution $f(x|\theta)$, then the overall likelihood is the product of the individual likelihoods:

$$\mathrm{L}(\theta|\mathrm{x_i}) = \prod_{\mathrm{i=1}}^{\mathrm{n}} f(x_i|\theta) = \prod_{\mathrm{i=1}}^{\mathrm{n}} L(\theta, x_i)$$

For computational and differentiation purposes, we use the **log-likelihood** which converts the product into a sum:

$$\log\left(\prod_{i=1}^{n} f(x_i|\theta)\right) = \sum_{i=1}^{n} \log\big(f(x_i|\theta)\big)$$

as such we can easily differentiate the function L as such: $\frac{\partial L}{\partial \theta}$

This is mainly used to calculate $\hat{\theta}_M$ *which is the maximum likelihood estimator, by setting* $\frac{\partial L}{\partial \theta} = 0$ and this is quite easy to calculate compare to differentiating a product.

- ## Akaike information criterion (AIC)

(Akaike, 1974) is a fined technique based on in-sample fit to estimate the likelihood of a model to predict/estimate the future values. In easy term it's a form of scoring to spot the best model. Here a good model is the one that has minimum AIC among all the other models.

The concept is best-understood with the formula: $AIC = 2k - 2\ln(\hat{L})$

Letting k being the number of estimated parameters in the model and letting $\hat{L}$ being the maximum value of the likelihood function for the model.

- We can see that the AIC both evaluate the best-fitting model with the likelihood and also penalizes too of an important number of estimated parameters (which resolves most time in overfitting)

In our case, we see that we get the best likelihood using the parameters 8 and 2. The likelihood of (8,1) is pretty high as well. Nevertheless, these models are using too many parameters so their aic isn't as good as the ones of (0,1) and (0,2).

In the end, the (0,2) parameter seems to be the best we've got with a better likelihood than (0,1) and with the best aic among the studied parameters.

6. Use the Box-Pierce test and the Shapiro-Wilk test (shapiro.test) on the residuals of all 3 models applied to the logGNP data and display their autocorrelogram. From there, what can you say on the stationarity of the residuals? How do you justify which model is the best?

> ✓ Explanation: Why do we study the residuals?
> **To check whether a regression/autoregression model, simple or multiple, has achieved its goal to explain as much variation as possible in a dependent variable while respecting the underlying assumption. Ideally all residuals should be small and unstructured; this then would mean that the regression analysis has been successful in explaining the essential part of the variation of the dependent variable. If, however, residuals exhibit a structure or present any special aspect that does not seem random, it sheds a "bad light" on the regression.**

Let's proceed to the residual analysis:

First, we need to retrieve the data from the Arima table and convert it as numeric:

```
A01=arima(DiffGNP,c(0,0,1))
A02=arima(DiffGNP,c(0,0,2))
A82=arima(DiffGNP,c(8,0,2))

A01=as.numeric(unlist(A01["residuals"]))
A02=as.numeric(unlist(A02["residuals"]))
A82=as.numeric(unlist(A82["residuals"]))
```

Then we proceed the Box-pierce and Shapiro-Wilk tests on each model residuals.

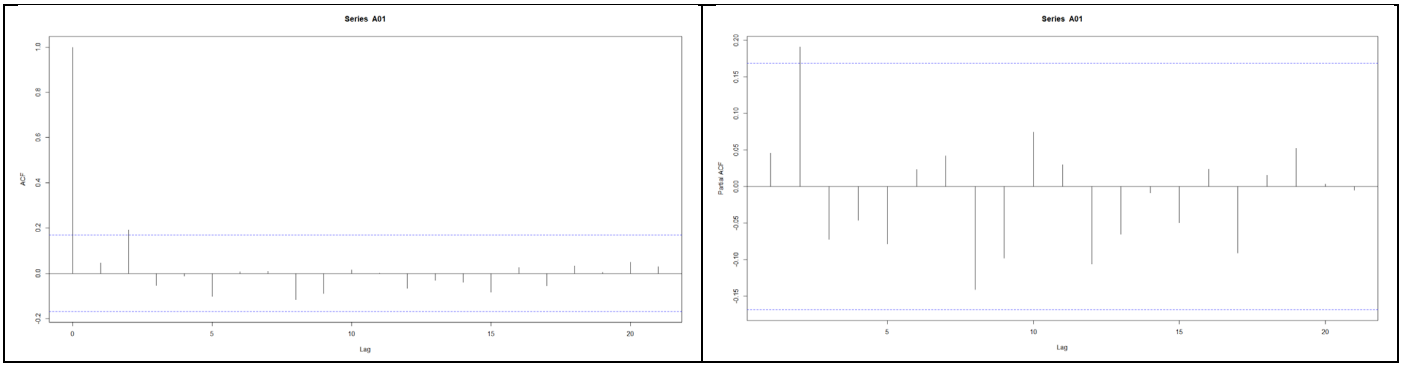
- **A01 residuals**

```
> shapiro.test(A01)
W = 0.97756, p-value = 0.02493

> Box.test(A01,lag=1,type="Box-Pierce")
X-squared = 0.28287, df = 1, p-value = 0.5948
```

For Arima 01 residuals, the shapiro.test returns a *p-value <= 0.05*, we can *reject* the NULL hypothesis that the residuals came from a Normal distribution with a confidence of 95%.

For the box test, the p-value is >0.05 at 0.59. This is an extremely high number; therefore, we can't conclude on the independency of the residuals.

Let's plot both the ACF and the PACF to check these results.



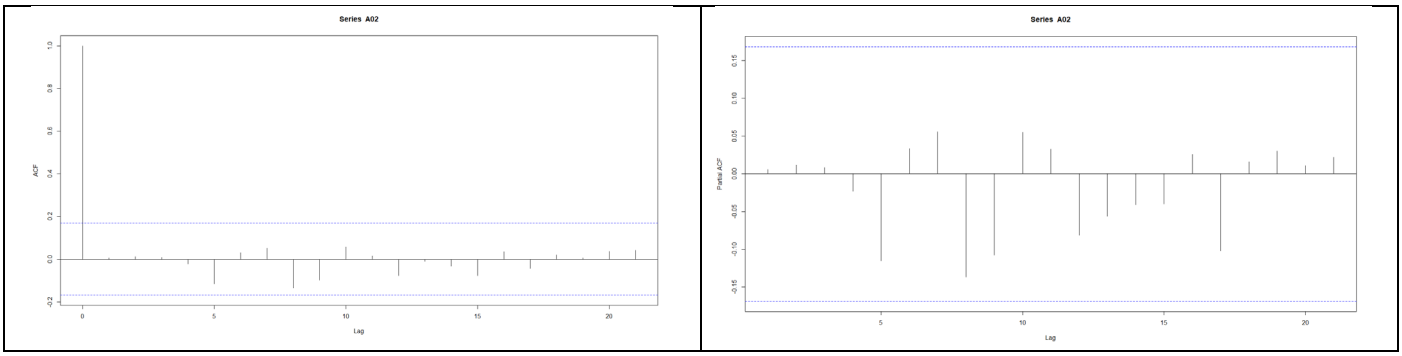
The null hypothesis is rejected; it suggests the time series is stationary.

- **A02 residuals**

```
shapiro.test(A02)
W = 0.98847, p-value = 0.3231

> Box.test(A02,lag=1,type="Box-Pierce")
X-squared = 0.0043454, df = 1, p-value = 0.9474
```

For Arima 02 residuals, the shapiro.test returns a ***p-value > 0.05***, we cannot ***reject*** the NULL hypothesis that the residuals came from a Normal distribution. The null hypothesis is not rejected; it suggests the time series is NOT stationary.

The p-value is >0.05 at 0.94. This is an extremely high number; therefore, we can't conclude on the independency of the residuals neither.

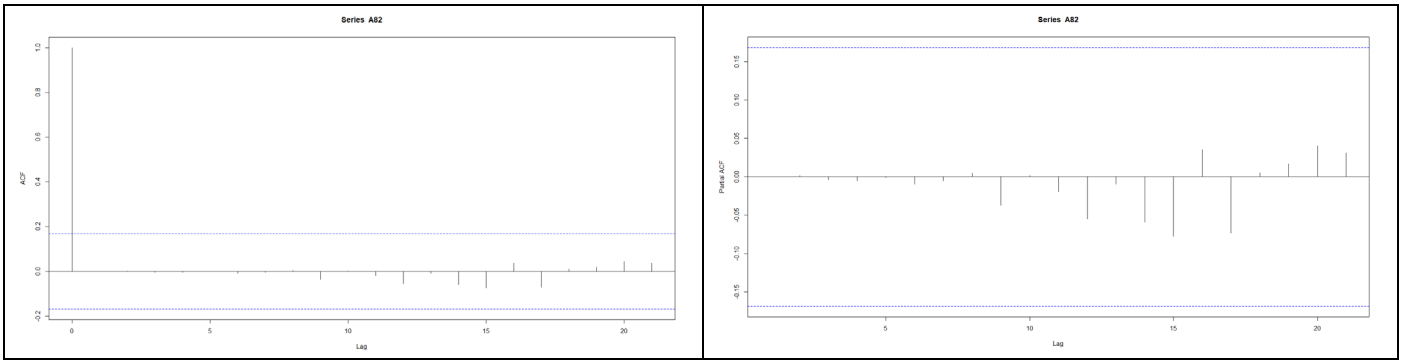


- A82

```
shapiro.test(A82)
W = 0.98606, p-value = 0.1872

> Box.test(A82,lag=1,type="Box-Pierce")
X-squared = 1.3959e-07, df = 1, p-value = 0.9997
```

For Arima 82 residuals, the shapiro.test returns a *p-value > 0.05*, we **cannot** *reject* the NULL hypothesis that the residuals came from a Normal distribution.

The p-value is >0.05 at 0.99. This is an extremely high number; therefore, we can't conclude on the independency of the residuals neither.



The null hypothesis is not rejected, there is no serial correlation in the residuals it suggests the time series is NOT stationary.

To conclude, we see that the model using (0,1) as parameters is the only model whose residuals are nearly assured not to follow a normal distribution. This is one aspect that we are looking for.

Nevertheless, the box test cannot help us to conclude on the independency of the residuals. This is worrying as a dependency between the residuals shows a potential weakness on a model.

In the end, we may wonder which parameter is the best between (0,1) (good shapiro test) and (0,2) (better likelihood and aic).

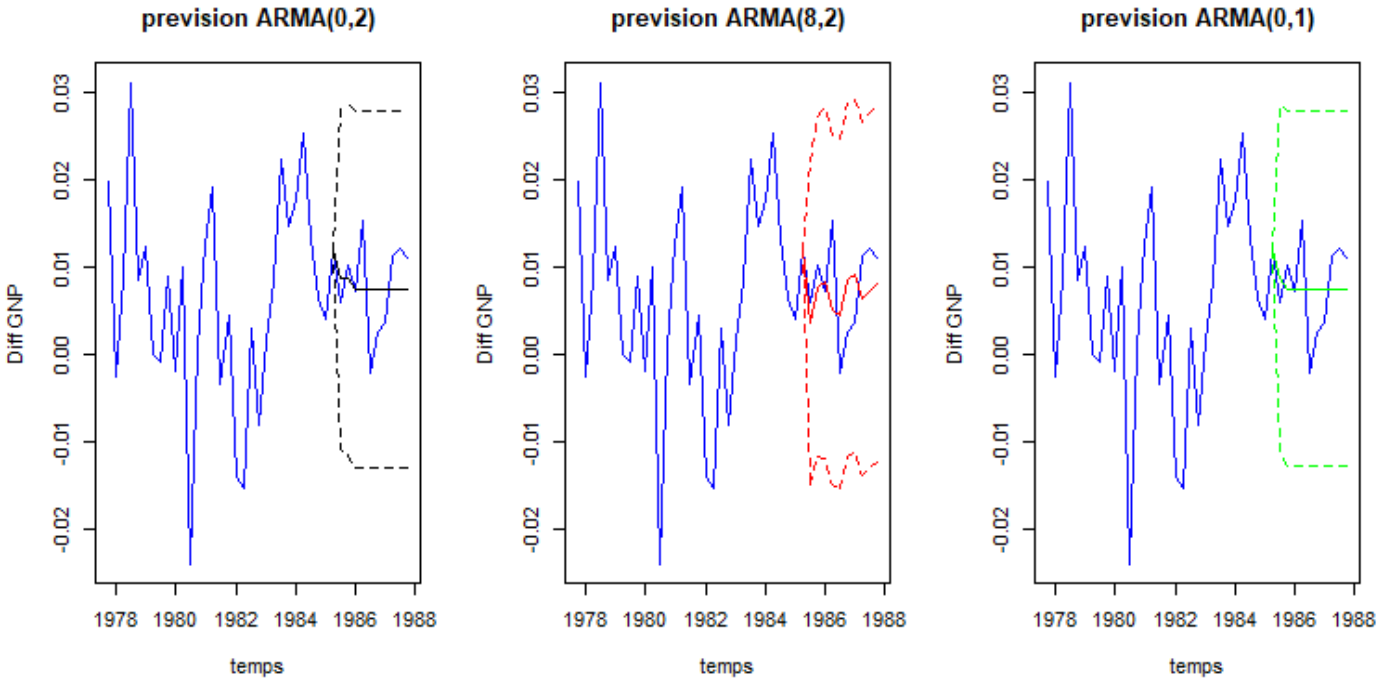
C. Predictions using ARMA

1. **For each model, draw on the same graph the 3 following elements (you may want to zoom on the end of the time series):**

We keep studying the 3 models we built previously. To assess the quality of these models, we are going to try to predict the 10 last values of the series after we removed them and retrained the model.

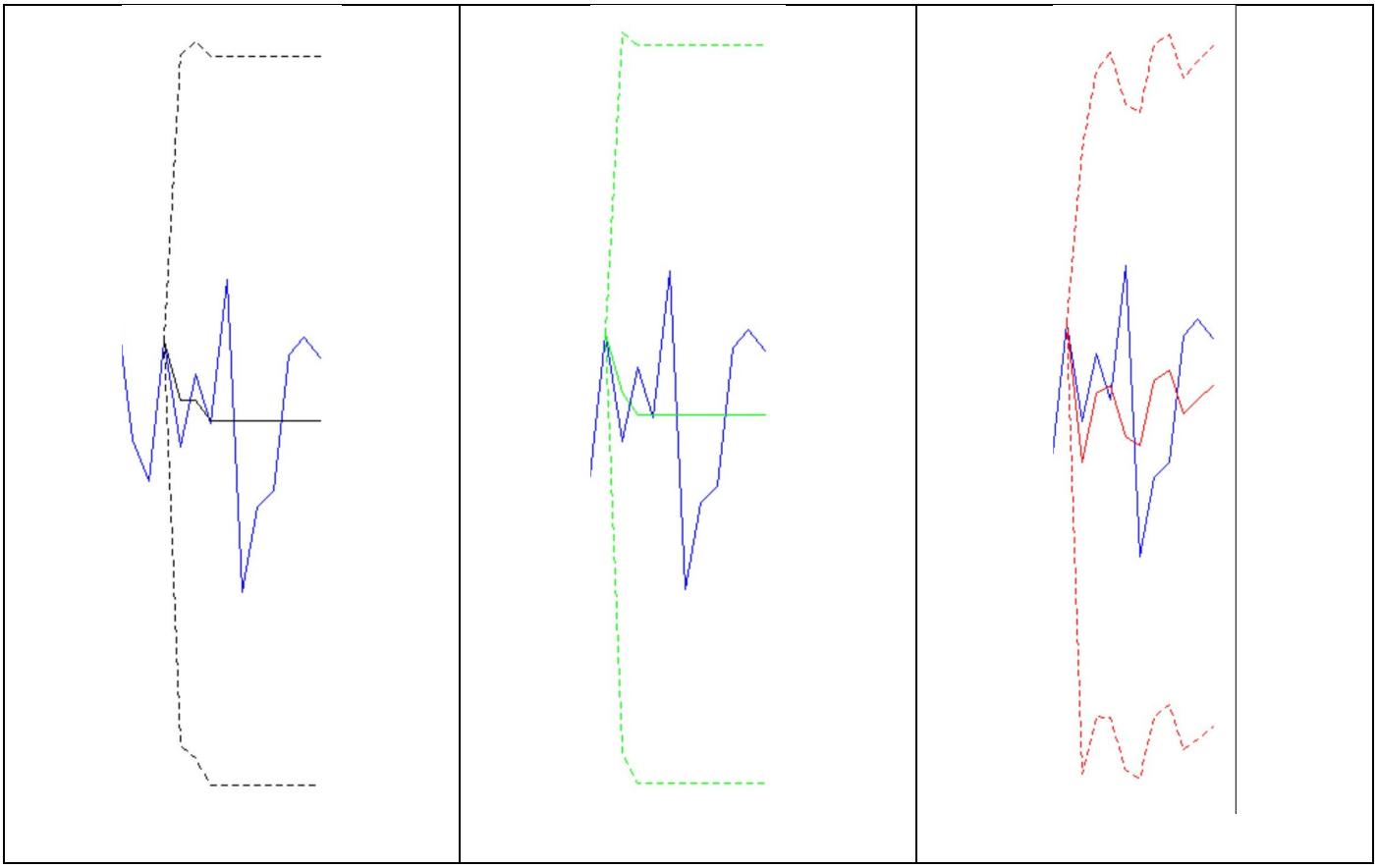
```
n <- 10
T=length(diffGNP)
index <- 1:(T - n - 1)
res01 <- predict(arima(diffGNP[index], c(0, 0, 1)), n)
res02 <- predict(arima(diffGNP[index], c(0, 0, 2)), n)
res82 <- predict(arima(diffGNP[index], c(8, 0, 2)), n)
```

We plot the results of our predictions to evaluate the different models:



2. Using the previous question, which model seem to give the best results?

Let's have a closer look at the predictions and zoom on it:



Prevision ARMA(0,2)	Prevision ARMA(0,1)	Prevision ARMA(8,2)

What's striking at first glance is the clear overfitting of the Arma (8,2) model. The predicted mean does not seem to represent anything. There is not much difference between the A01 and A02, but we can clearly see that the ARMA (0,2) takes longer to converge to the final mean. The ARMA (0,2) prediction is smoother.

In the end, the ARMA (0,1) seems to be the best model for our series. Nevertheless, ARMA (0,1) is a pretty good model too.

3. Translate these models into ARIMA(p,d,q) for the original logGNP time series.

We can easily translate these models as following:

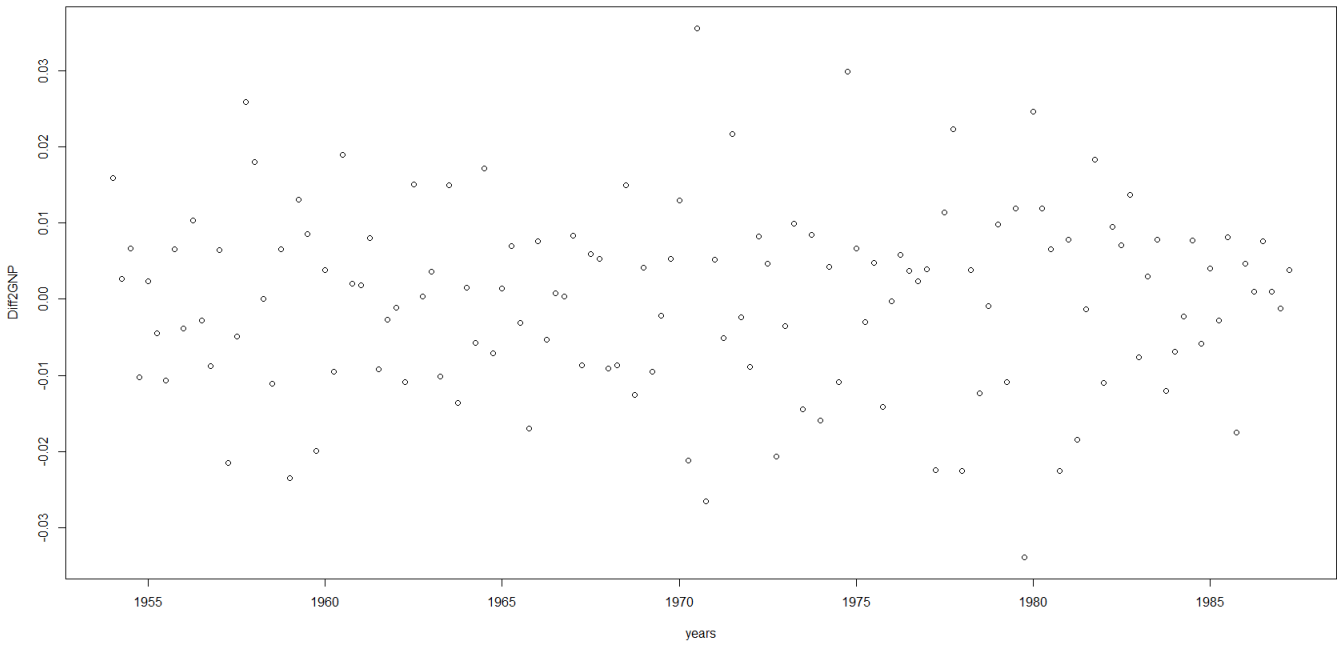
```
A01=arima(clnGNP, c(0,1,1))
A02=arima(clnGNP, c(0,1,2))
A82=arima(clnGNP, c(8,1,2))
```

D. ARIMA Model

We want to repeat the previous steps to build and to evaluate an ARIMA(p, 2, q) model for the GNP series.

Let's differentiate 2 times our original time series and plot the result depending on the quarters:

```
Diff2GNP=diff(diff(clnGNP))
```



We can't say much about the scatter plot. Though, the values seem to be equally distributed along the X-axis (constant variance) and the mean seems to be around 0. This we're going to verify with a Student test.

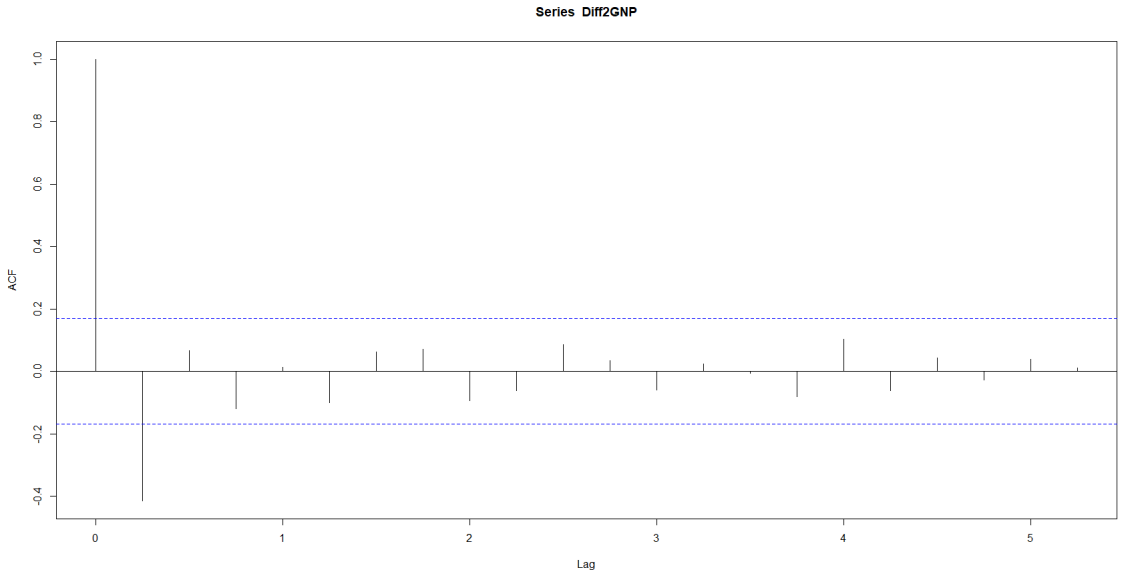
```
mean(Diff2GNP)
[1] 0.0001403431
> t.test(Diff2GNP, mu=0)

    One Sample t-test

data:  Diff2GNP
t = 0.13481, df = 133, p-value = 0.893
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.001918837  0.002199523
sample estimates:
   mean of x
0.0001403431
```

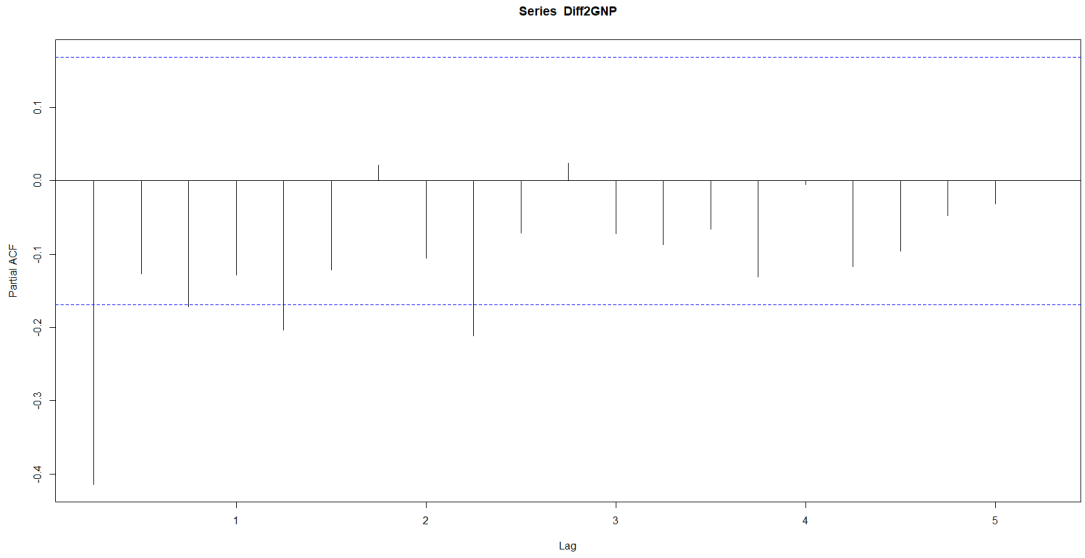
The mean is equal to 0.00014. This is very close to 0. Furthermore, the Student test tells us that the mean has a 95% chance to be between -0.0019 and 0.0022. The p-value doesn't allow us to contest the null hypothesis, which is that the true mean of the series is equal to 0.

Unlike the 1-time differentiated series, the 2-times differentiated can be considered as a centered series. This is good for our future ARIMA models.

We can assume that this series is stationary.

Now that we have verified the quality of our time series, we have to look for the possible parameters for the ARIMA models.





Thanks to the autocorrelogram and to the partial autocorrelogram, we determine that the best parameters appear to be the following:

(2,2,2) / (4,2,2) / (6,2,2) / (10,2,2)

We evaluate each of these parameters by creating ARIMA (p, 2, q) models from the original time series.

```
arima(clnGNP,c(2,2,2))
sigma^2 estimated as 9.246e-05:  log likelihood = 430.14,  aic = -850.28


> arima(clnGNP,c(4,2,2))
sigma^2 estimated as 9.109e-05:  log likelihood = 430.73,  aic = -847.47


> arima(clnGNP,c(6,2,2))
sigma^2 estimated as 9.133e-05:  log likelihood = 430.85,  aic = -843.7
```

```
> arima(clnGNP,c(10,2,2))
sigma^2 estimated as 8.684e-05:  log likelihood = 433.36,  aic = -840.71
```

The log likelihood of the model (10,2,2) is the highest (= 433.36). Nevertheless, its aic is too high compared to the others.

The other likelihoods are very close to each other: 430.14, 430.73 and 430.85. The biggest gap is observed between the value of (2,2,2) and (4,2,2). However, the aic of (2,2,2) is the best (= -850.28) and this value goes up as we had more parameters.

To pick a good compromise between likelihood and aic, we would be tempted to say that the parameter (4,2,2) is the more suitable.

We can deal with our analysis in depth by proceeding to Box-pierce and Shapiro-Wilk tests on each model residuals:

```
A222=arima(clnGNP,c(2,2,2))
A422=arima(clnGNP,c(4,2,2))
A622=arima(clnGNP,c(6,2,2))
A1022=arima(clnGNP,c(10,2,2))

A222=as.numeric(unlist(A222["residuals"]))
A422=as.numeric(unlist(A422["residuals"]))
A622=as.numeric(unlist(A622["residuals"]))
A1022=as.numeric(unlist(A1022["residuals"]))
```

- **A222 residuals**

```
shapiro.test(A222)
W = 0.98863, p-value = 0.3289

> Box.test(A222,lag=1,type="Box-Pierce")
X-squared = 0.0021263, df = 1, p-value = 0.9632
```

- **A422 residuals**

```
shapiro.test(A422)
W = 0.98887, p-value = 0.347

> Box.test(A422,lag=1,type="Box-Pierce")
X-squared = 0.011811, df = 1, p-value = 0.9135
```

- **A622 residuals**

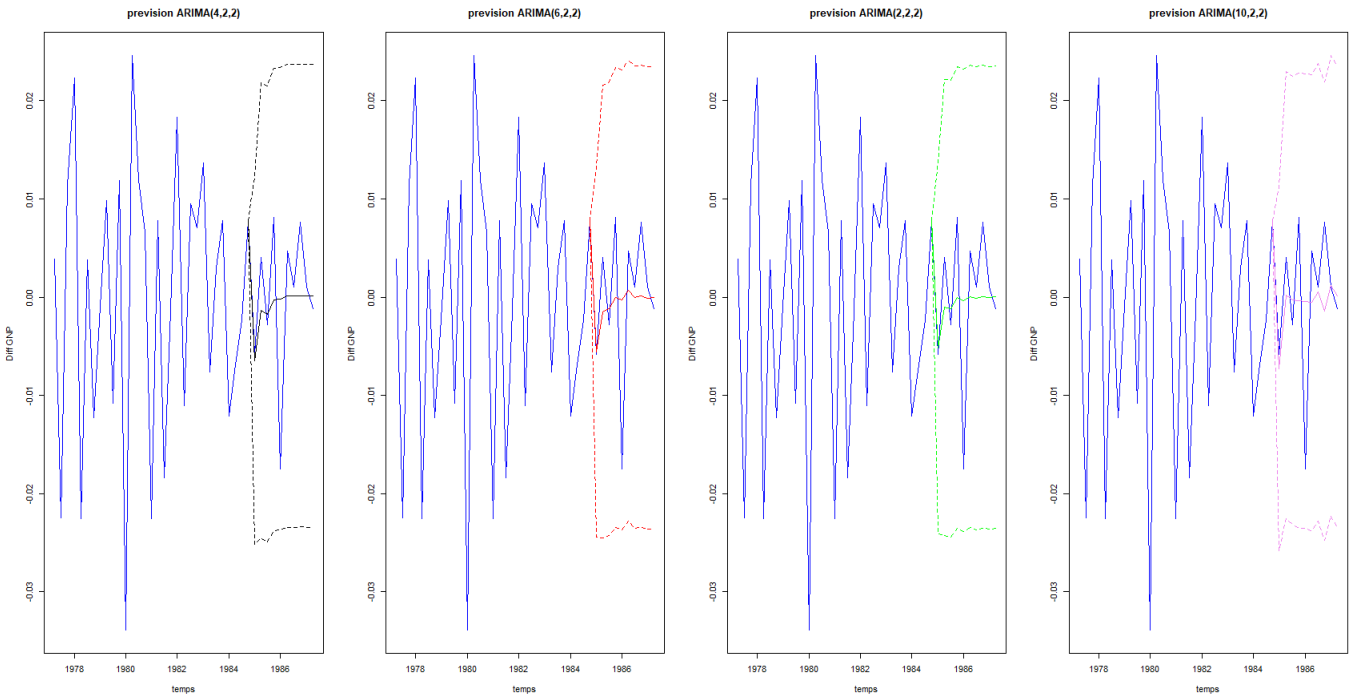
```
shapiro.test(A622)
W = 0.98895, p-value = 0.3527

> Box.test(A622,lag=1,type="Box-Pierce")
X-squared = 0.011653, df = 1, p-value = 0.914
```

- **A1022 residuals**

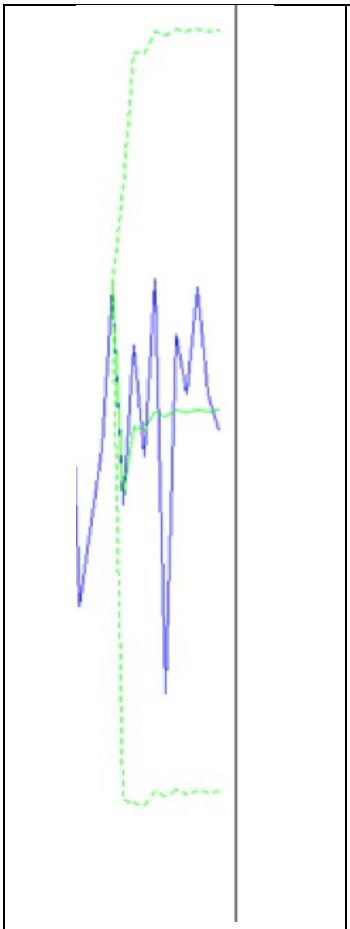
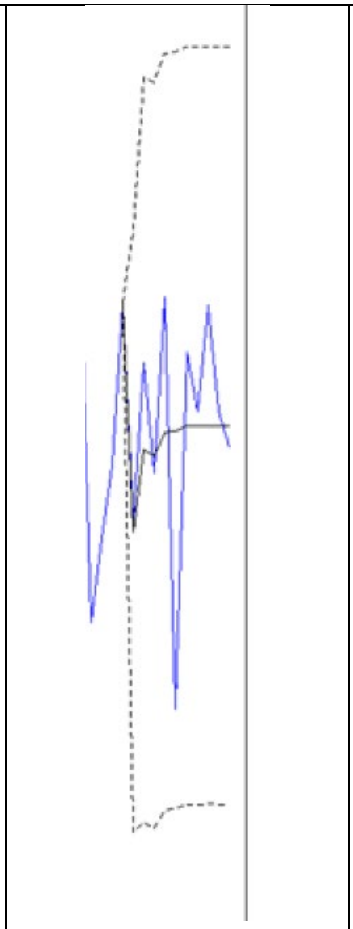
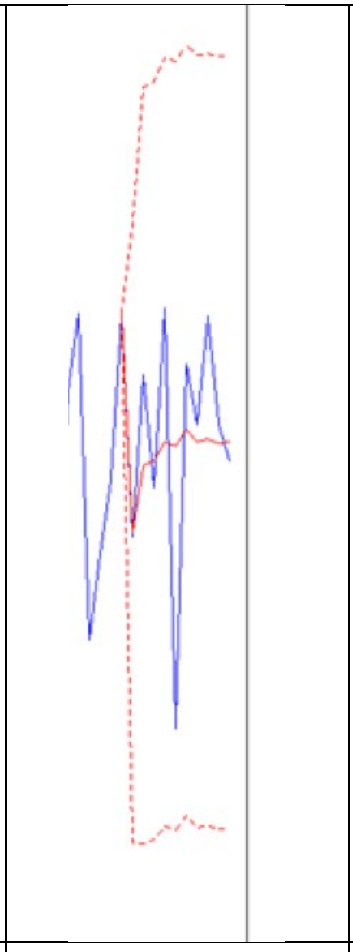
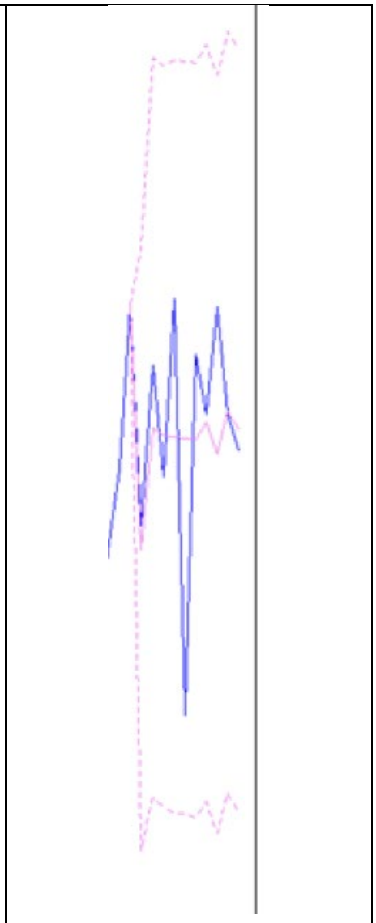
```
shapiro.test(A1022)
W = 0.98713, p-value = 0.2353

> Box.test(A1022,lag=1,type="Box-Pierce")
X-squared = 0.0053055, df = 1, p-value = 0.9419
```

For every test, the p-value is too high, so we can't conclude about the dependency of the residuals nor tell if the residuals come from a Normal distribution. This is a good thing for the Box test as we don't want the null hypothesis to be rejected. However, we would have preferred to be able to reject the null hypothesis of the Shapiro test.

Finally, we evaluate how good our models are to predict the 10 last values of our time series, after we removed them. We get the following result:



We zoom on the interesting spot:

			
Prevision ARIMA(2,2,2)	**Prevision ARIMA(4,2,2)**	**Prevision ARIMA(6,2,2)**	**Prevision ARIMA(10,2,2)**

The 4 predictions converge quite fast to the mean. The prediction using (10,2,2) has some oscillation around the mean, which makes it the worst prediction among the 4. The oscillations have a less important amplitude for (6,2,2). That's even better for (2,2,2). Nevertheless, the best prediction seems to be the one with (4,2,2) as parameters as it reaches the mean value very fast without too many oscillations.

To conclude, we are still able to create a proper model with a second differentiation. However, the obtained model isn't as good as the one we got with a single differentiation. This shows that a differentiation can help to build a model, but you must not abuse it: too many differentiations can deteriorate your model.

References :

https://onlinecourses.science.psu.edu/stat504/node/27/