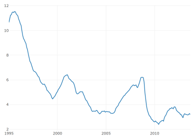
Intelligent Data Analysis - Homework 3

Panagiotis Michalopoulos, Javier de la Rua, Michail Gongolidis,
Ignacio Rodriguez, Daniel Minguez

January 16h, 2019

1 Plot the series and comment

As we are group 10, we have used the file with data related to the monthly interest rate on unsubsidised loans for house purchase (percentage) from January 1995 to September 2013. The first thing we did was plot the series:

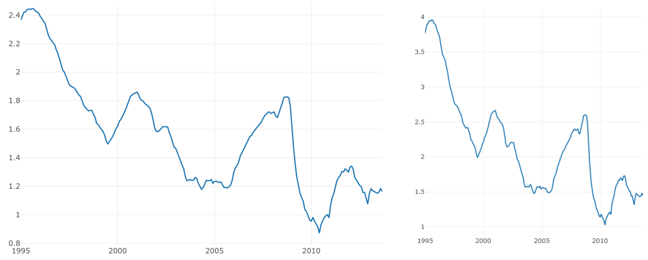


During this report we are going to use the Box-Jenkins methodology that consists basically in:

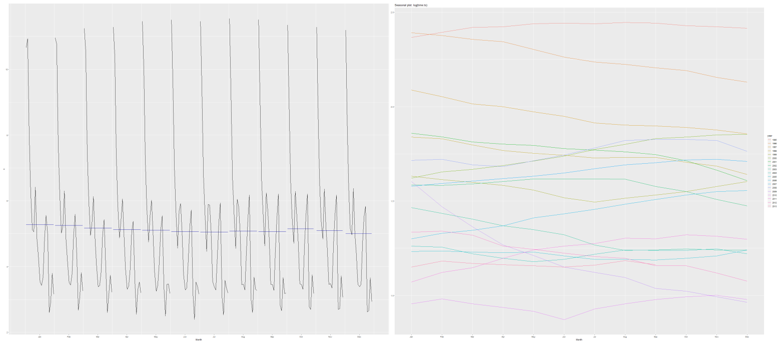
- **Model identification and model selection:** check for stationarity, seasonality and use of plots of autocorrelation and partial autocorrelation.
- **Parameter estimation:** use computation algorithms to arrive at coefficients that best fit the selected ARIMA model.
- **Model checking:** test whether the estimated model conforms to the specifications of a stationary univariate process.

We can appreciate that the time series has an obvious global downward trend without seasonality (so it is not a stationary series). We can appreaciate also upward subtrends in some intervals.

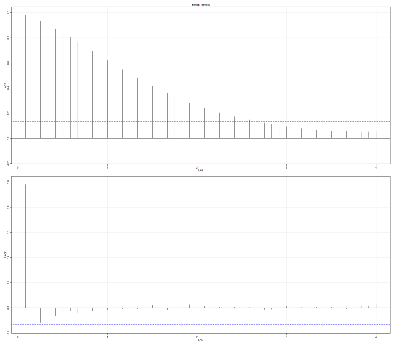
After applying the log transformation (left) and the box-cox transformation (right) to the data we obtain the following result:



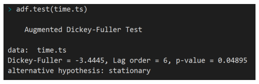
It does not make so much change, neither in trend nor in variability, so we decided to work with the original time series. We can appreciate also that the time series is not stationary and that it does not have a seasonal trend by checking the seasonal differences and the month differences:



Finally we also used the autocorrelation and partial autocorrelation plots:



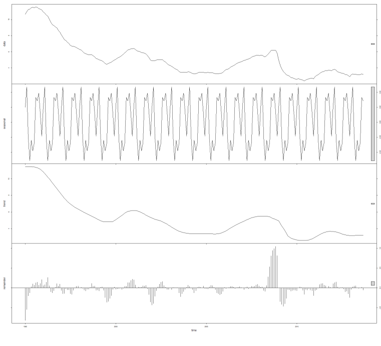
We can see that ACF does not drop quickly to values inside or almost inside the critical values and that r_1 is large and positive. All this are signs of a non-stationary time series. PACF value r_1 is almost 1. This is also a sign of non-stationary process. Therefore the data should be differenced to apply a model. We applied also a stationary test, with a p-vlaue lower than 0.05 we reject the null hypothesis of stationarity:



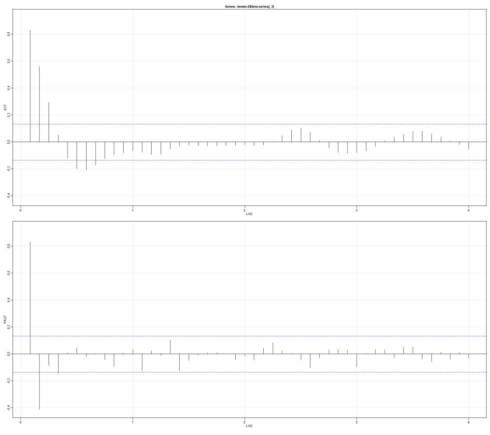
```
> adf.test(time.ts)

	Augmented Dickey-Fuller Test

data:  time.ts
Dickey-Fuller = -3.4445, Lag order = 6, p-value = 0.04895
alternative hypothesis: stationary
```

2 Obtain a plot of the decomposition of the series. Does the remainder look like a white noise to you?

Since the log transformation does not change the variance of our time series and our series does not have seasonality, we use an additive decomposition approach "*Time series = Seasonal + Trend + Random*" with *stl* and the parameters *s.window="periodic", t.window=15, robust=TRUE* to the data. We can appreciate the downtrend:

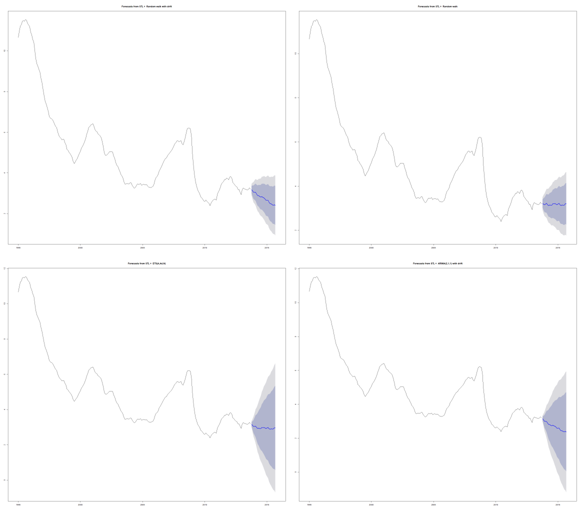


Also the differences of the remainder across time are evident, this could indicate not constant variance through time or outliers in the observations due to specific events. Using the ACF and PCAF graph we can check as well that the r_1 is close to 0.6 and -0.4 respectively, so we can say that the remainder does not seem white noise:



2.1 Use the function forecast() to forecast future values.

We have used the forecast function with Random walk with drift, random walk, ETS and ARIMA:

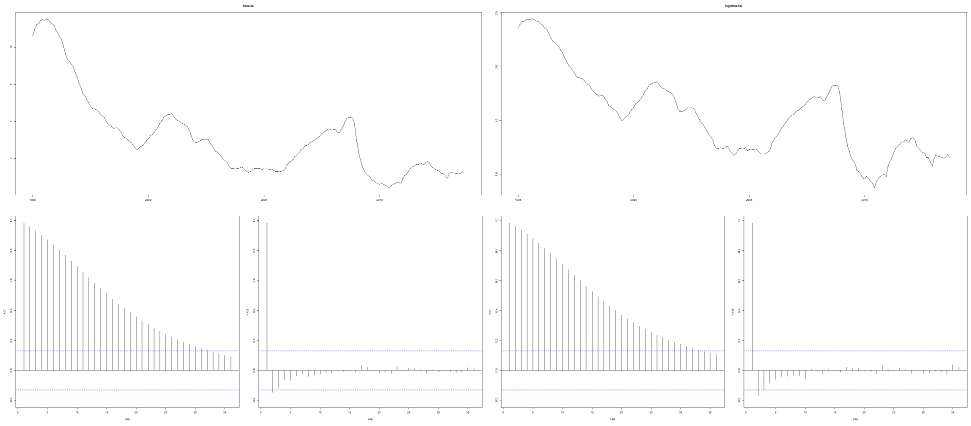


We can appreciate that the confidence intervals of ETS and ARIMA are bigger that the one created with random walk variants because of how these models are constructed.

3 Fit an ARIMA model to your time series

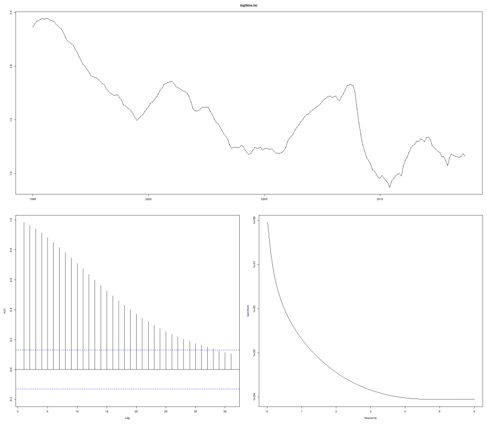
3.1 Decide on whether to work with your original variable or with the log transform one

As we have seen in the previous part, we are going to use the original data. The following is the tsdisplay of the data and log data:



3.2 Are you going to consider a seasonal component?

We are not going to use a seasonal component since, as we checked in the previous part, there is no seasonal component in our series. Using the periodogram we can also intepret this:

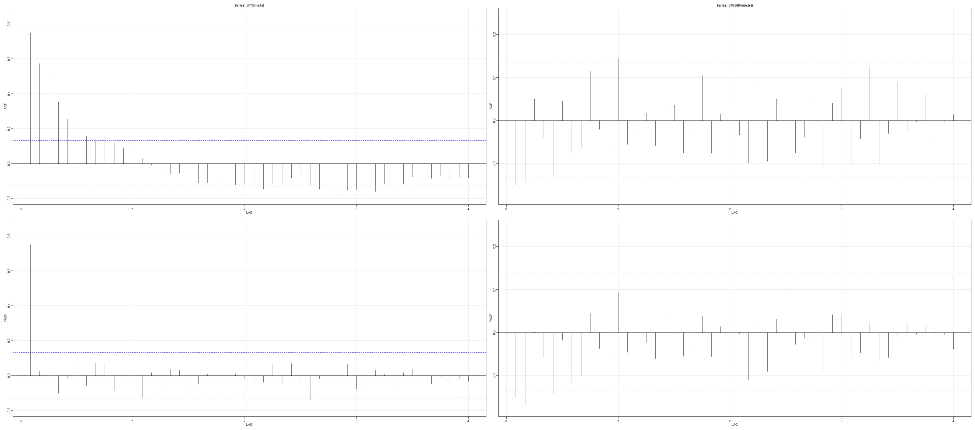


3.3 Decide on the values of d and D to make your series stationary.

In our case, since we do not have a seasonal component, we have $D = 0$ (And so $P = 0$ and $Q = 0$), in order to decide about which d we take, we compare first, second and third differences, obtaining the following standard deviations:

Difference	Standard deviation
Original data	2.239662
1st	0.1486072
2nd	0.1030107
3rd	0.1562024

So we have to decide between 1st and 2nd differences, we apply the adf.test (with p-values lower than 0.05, and Dickey-Fuller of -4.3715 and -7.6347 respectively) and check the ACF-PACF graphs:



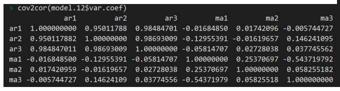
We decided to try with second order differences because of the shape of the PACF, with smaller values.

3.4 Identify values for p and q for the regular part and P and Q for the seasonal part.

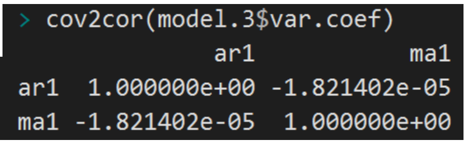
We have fitted different models for different p-q parameters to the train data (data before 2013), we have also calculated the RMSE with the test data (2013 data):

ARIMA Model	AIC	BIC	RMSE
(1,2,0)	-376.4445	-369.7126	0.8832099
(0,2,1)	-377.8681	-371.1361	0.8473354
(1,2,1)	-389.8618	-379.7638	0.51469
(2,2,0)	-378.3003	-368.2024	0.8324347
(0,2,2)	-379.3241	-369.2262	0.8028774
(2,2,1)	-388.1842	-374.7203	0.5174389
(1,2,2)	-388.2293	-374.7654	0.5180392
(2,2,2)	-385.872	-369.0421	0.5150287
(3,2,0)	-376.3377	-362.8738	0.8291334
(3,2,1)	-374.7115	-357.8816	0.8475045
(3,2,2)	-385.057	-364.8612	0.5252064
(3,2,3)	-394.4765	-370.9147	0.4526032
(0,2,3)	-377.8079	-364.344	0.7863561
(1,2,3)	-386.5934	-369.7636	0.5257217
(2,2,3)	-384.8269	-364.631	0.5235576

So, apparently, the best model according with AIC was ARIMA(3,2,3) and according BIC was ARIMA(1,2,1). If we dont focus in RMSE and we check the correlation between the coeficients of ARIMA(3,2,3) we obtain relatively high correlation between ma1 and ma3 and ar coefficients. We include also the result with ARIMA(1,2,1), obtaining the following correlations (It seems ARIMA(1,2,1) behaves better regarding this point):

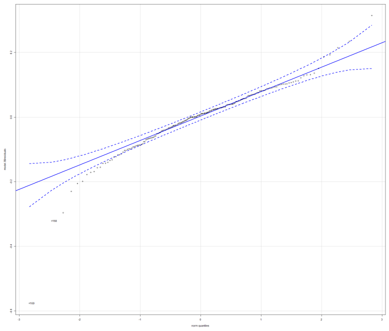


```
> cov2cor(model.12$var.coef)
             ar1         ar2         ar3         ma1         ma2          ma3
ar1  1.000000000  0.95011788  0.98484701 -0.01684850  0.01742096 -0.005744727
ar2  0.950117882  1.00000000  0.98693009 -0.12955391 -0.01619657  0.146241095
ar3  0.984847011  0.98693009  1.00000000 -0.05814707  0.02728038  0.037745562
ma1 -0.016848500 -0.12955391 -0.05814707  1.00000000  0.25370697 -0.543719792
ma2  0.017420959 -0.01619657  0.02728038  0.25370697  1.00000000  0.058255182
ma3 -0.005744727  0.14624109  0.03774556 -0.54371979  0.05825518  1.000000000
```

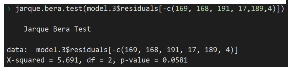


```
> cov2cor(model.3$var.coef)
              ar1           ma1
ar1  1.000000e+00 -1.821402e-05
ma1 -1.821402e-05  1.000000e+00
```

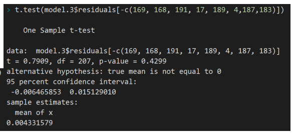
Regarding the residuals we have checked for normality of ARIMA(1,2,1) with jarque bera test, obtaining that the residuals are not normal. We looked for outliers:



After removing some ouliers (we would need to find an explanation for them), we obtained normal residuals (although the mean test say that mean is not equal to 0):

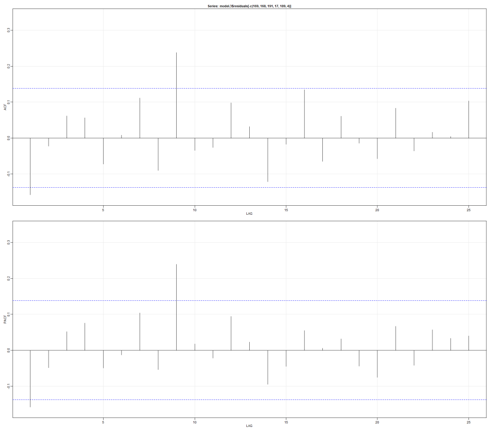


```
> jarque.bera.test(model.3$residuals[-c(169, 168, 191, 17,189,4)])

    Jarque Bera Test

data:  model.3$residuals[-c(169, 168, 191, 17, 189, 4)]
X-squared = 5.691, df = 2, p-value = 0.0581
```



```
> t.test(model.3$residuals[-c(169, 168, 191, 17, 189, 4,187,183)])

    One Sample t-test

data:  model.3$residuals[-c(169, 168, 191, 17, 189, 4, 187, 183)]
t = 0.7909, df = 207, p-value = 0.4299
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.006465853  0.015129010
sample estimates:
  mean of x
0.004331579
```

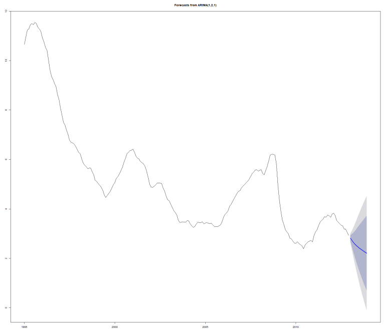
Also we have plot the autocorrelations

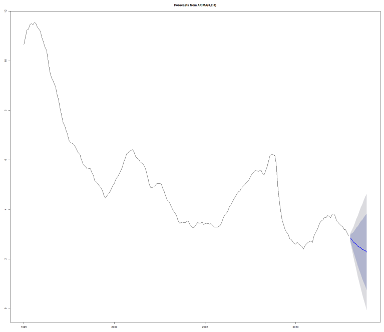


So, forgetting about RMSE, it seems that ARIMA(1,2,1) behaves better that ARIMA(3,2,3).

3.5 Ise the model to make forecasts

We have done forecast with ARIMA(3,2,3) and ARIMA(1,2,1), ARIMA(3,2,3) provides bigger confidence intervals:





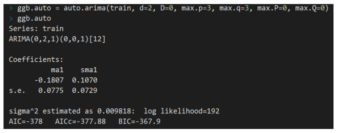
3.6 Use the function getrmse to compute the test set RMSE of some of the models you have already fitted.

As we can see in the previous section, we obtained that the model ARIMA(3,2,3) is the one that get better RMSE, although the correlation of some of its coeficients is close to 1. ARIMA(1,2,1) get the second better results.

So if we take into account only this mesaure the best model would be ARIMA(3,2,3), but we recommend to continue validating this model through future observations since the ARIMA(1,2,1) also behaves well and is more simple.

3.7 You can also use the auto.arima() function with some of its parameters fixed, to see if it suggests a better model that the one you have found.

If we execute the auto.arima function we obtain the following result:



```
> ggb.auto = auto.arima(train, d=2, D=0, max.p=3, max.q=3, max.P=0, max.Q=0)
> ggb.auto
Series: train
ARIMA(0,2,1)(0,0,1)[12]

Coefficients:
          ma1    sma1
      -0.1807  0.1070
s.e.   0.0775  0.0729

sigma^2 estimated as 0.009818:  log likelihood=192
AIC=-378   AICc=-377.88   BIC=-367.9
```

This ARIMA(0,2,1) has an AIC of -378 (as we obtained before), but it is not a better result (as we saw in the previous part) that ARIMA(1,2,1) or ARIMA(3,2,3). Maybe it tooks other measurements into account like the log likelihood or sigma squared showed.