Box & Jenkins with transfer function in R: a case study to Natural Inflow Energy series from the Brazilian southwest subsystem

Pedro Guilherme Costa Ferreira[1], Reinaldo Castro Souza[2], and Daiane Marcolino de Mattos[1]

[1] *Instituto Brasileiro de Economia (IBRE)*

[2] *Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio)*

Abstract: The use of auxiliary variables in univariate models of Box & Jenkins through transfer function can improve the modeling and time series forecasting. This article aims to present this methodology using the R software, a fact not yet addressed in international literature. Furthermore, a study using the Sunspot and Southern Oscillation Index (SOI) series for modeling the Natural Inflow Energy (NIE) time series from the Brazilian southwest subsystem showed superior results when compared with ARIMA model.

Key-words: Box & Jenkins with transfer function, Natural Inflow Energy, Southern Oscillation Index (SOI), Sunspot.

1 Introduction

The Box & Jenkins time series models can incorporate other auxiliary variables. The way these auxiliary variables x_t influence the response variable Y_t, that is, how the oscilations of these variables affect the course of the response variable is given by a transfer function $f(x_t)$:

$$Y_t = f(X_t) + \varepsilon_t \tag{1}$$

where ε_t can be a white noise or an ARIMA model.

The $f(X_t)$ can collate past and/or present values from one or more time series (TS), which can be quantitative or binary (*dummy*) and this distinction implies how the identification of the $f(x_t)$ will be. In the case of a quantitative TS, the generic form of the $f(X_t)$ can be denoted as:

$$f(X_t) = \frac{(w_0 + w_1 L + w_2 L^2 + ... + w_s L^s)}{(1 - \delta_1 L - \delta_2 L^2 - ... - \delta_r L^r)} X_{t-b} \tag{2}$$

[1]Instituto Brasileiro de Economia (IBRE|FGV), R. Barão de Itambi, Botafogo, Rio de Janeiro, Brasil; tel: +55 21 3799-6751; emails: pedro.guilherme@fgv.br; daiane.mattos@fgv.br.

[2]Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio). R. Marquês de São Vicente -225, Gávea, Rio de Janeiro, Brasil; email: reinaldo@ele.puc-rio.br.

Note that to identify $f(x_t)$, it is necessary to find the values of the r, s and b and estimate the parameters $w_i, i = 0, ..., s$ and $delta_j, j = 1, ..., r$.

In literature you can find articles on the topic: Hanssens (1980), for example, when modeling sales of vegetable compounds from advertising expenses, it showed that the method outperforms some univariate models. Nogales and Conejo (2006) used the method to predict the electricity price and was shown to be a good alternative to the ARIMA and neural network models. Kannan and Farook (2013), focusing on global warming, modeled the atmospheric temperature by CO_2 emissions and concluded that the results are of good quality. McDowell (2002) teaches how it can be easy to adjust a transfer function model using the software STATA.

Given the usefulness of the methodology, this article aims to present such a method using the software R, a fact not yet addressed in international literature. To achieve the objective, the article was divided into four more sections. In the section 2, the proposed algorithm and methodology are addressed simultaneously. In other words, using the TS available from Box & Jenkins (BJ) explains the methodology proposed by the authors (eg identify and estimate transfer functions) using the R software. In the section 3, we applied the proposed algorithm in a real problem experienced by the Brazilian Electric Sector (SEB) which is the forecasting/simulation of the Natural Energy Affluent[1]. At the end of this section we compare the results measured by BJ model with and without a transfer function. Finally, in the section 4 we present the conclusion and future works.

2 Proposed Algorithm and Methodology

This section presents the proposed algorithm and methodology. The R code are available at `http://git.io/vC9qY` and has been developed using R software (Keeble (2012); Ripley (2002); Meyer (2002)).

For modeling the following packages must be installed: `forecast` (used in the estimation and forecasting of ARIMA models); `TSA` (used in the estimation of ARIMA models with transfer function); `tseries` (used to perform normality tests in residuals); and `FinTS` (used to perform conditional heterocedasticity tests in time series).

To model and estimate the transfer function $f(X_t)$, we evaluated the classic case extract from Box and Jenkins (1970). This example investigates the adaptive optimization made by gas heater, this is, the authors used an air combination and metano to get a gas mixture with CO_2 (carbon dioxide). The air input has been kept stable while the metano input could vary as researchers wish. After the combination, the CO_2 concentration has been measured. The goal was to estimate the relationship between the metano volume (X_t) and CO_2 (Y_t). Each variable represents a time series with 296 observations and the database can be downloaded directly to R running the command lines below:

[1] In general terms, NIE is the quantity of electricity that can be generated by hydroelectric facilities with the water retrieved by the hydroelectric plants. This energy is estimated by assuming that the level of the reservoirs comprises an average level of 65% of their total capacity and assuming an operational policy. Note that this value may vary according to the operational policy (Terry et al., 1986).

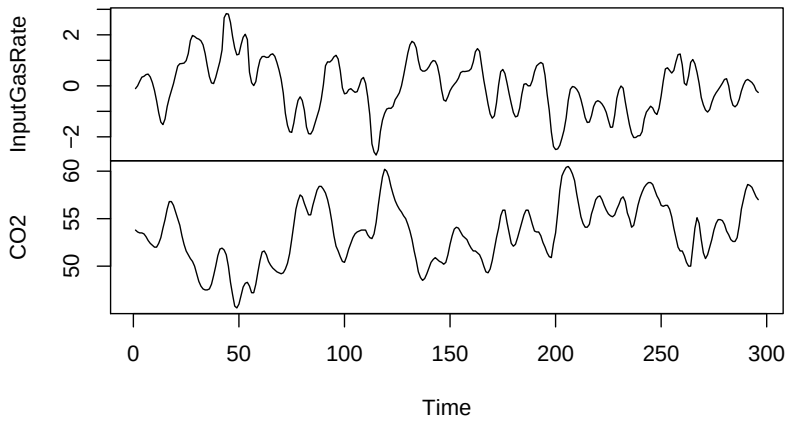

Figura 1: Input Gas Rate (X) and CO_2 (Y)

```
> devtools::source_url("http://git.io/vCXJC")
> gas
> plot(gas, main = "Input Gas Rate (X) and CO2 (Y)")
```

The columns `gas` object refer to the independent variable x_t and the dependent variable Y_t respectively. Note that the series does not have a steady behavior for the whole time span, with non-constant variations and a slight tendency (Figure 1). To corroborate this statement the correct way would be to do a unit root test(Dickey and Fuller (1979); Phillips and Perron (1988)), however as the goal of the article is just to explore the methodology Box & Jenkins, we didn't explore this topic here.

With the data available, we can follow the identification steps proposed by $f(X_t)$, which according to BJ methodology are:

1. Calculate the Cross Correlation function (CCF) between Y_t and X_t;
2. Identify r, s and b;
3. Estimate the Box & Jenkins model with Transfer function;
4. Verify if the model is suitable.

2.1 Estimating the Cross Correlation function between Y and X

As seen in section 1 to identify transfer functions, initially, simply stipulate values for r, s and b. The identification of these values is made by calculating the cross-correlation function (CCF) between Y_t and auxiliary variables (only one in this example). The CCF between the time series Y_t

and x_t shows the correlations between them for different time lags. It is defined mathematically as:

$$ccf(k) = \frac{c_{xy}(k)}{s_x s_y}, \quad k = 0, \pm 1, \pm 2, ... \tag{3}$$

where:

s_x e s_y represents, respectively, the X_t and Y_t standard deviations.

$c_{xy}(k)$ represents the covarianece between the two variables in lag k:

$$c_{xy}(k) = \begin{cases} \frac{1}{n}\sum_{t=1}^{n-k}(x_t - \bar{x})(y_{t+k} - \bar{y}), & k = 0, 1, 2, ... \\ \frac{1}{n}\sum_{t=1}^{n+k}(y_t - \bar{y})(x_{t-k} - \bar{x}), & k = 0, -1, -2, ... \end{cases}$$

Assuming $k > 0$, the CCF shows the relationship between X in t time and Y in future time $t+k$ (leading). Assuming negative values for k, the CCF shows the relationship between X in t time and Y in past time $t-k$ (lagging).

It is important to know that the CCF is affected by the autocorrelation x_t and Y_t, and if the TS are not stationary, the result of CCF does not truly reflect the degree of association between them (Hamilton, 1994; Phillips and Perron, 1988). The pre-whitening method is suggested by BJ to fix this problem.

Pre-whitening method consists to extract the trend (deterministic and stochastic) and the autocorrelation present in the time series. The method consists the follow steps:

a. Adjust an ARIMA model to X_t;
b. "Filter"Y_t using the estimated model from (a), this is, Y_t is the same X_t model (with the same estimated parameters);
c. Save the residuals from both models;
d. Estimate the CCF using the residuls obtained from (c).

Now, we'll follow these steps using the TS available by Box and Jenkins (1970).

a. Adjusting an ARIMA model to X_t:

To identify the orders of the ARIMA model for x_t, we use the autocorrelation function (ACF) and partial autocorrelation function (PACF)[2], that the R function are respectively `acf()` e `pacf()`.

```
> acf(gas[,"InputGasRate"], lag.max = 36)
> pacf(gas[,"InputGasRate"], lag.max = 36)
```

The exponential decay of the ACF and the abrupt cut of the PACF in lag 3 (Figure**??** suggest an ARIMA model (3,0,0). When adjusting the ARIMA model (function/codeArima of the package **forecast**), the constant was not significant, which is excluded from the model. Following are the routines for the estimation of the model.

[2]A more robust method that assists in identifying ARIMA models are the criteria information, such as (Akaike, 1973) e Bayesian Information Criterium (Schwarz, 1978). They will be used in the section 3, which presents a more current example

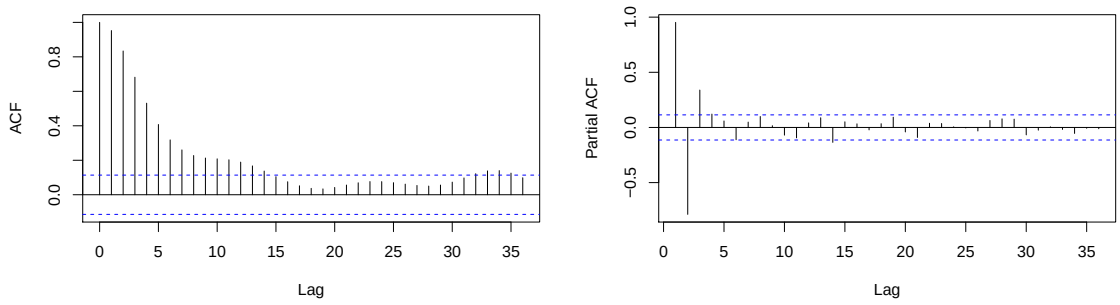

Figura 2: ACF and PACF: Input Gas Rate (X)

```
> library("forecast")
> model_x <- Arima(gas[,"InputGasRate"], order = c(3,0,0), include.mean = F)


# InputGasRate ARIMA Model
Coefficients:
         ar1      ar2     ar3
      1.9696  -1.3659  0.3399
s.e.  0.0544   0.0985  0.0543

sigma^2 estimated as 0.03531:  log likelihood=72.52
AIC=-137.04   AICc=-136.9   BIC=-122.27
```

b. "Filter"Y_t using the model estimated from (a)

To filter Y_t through the X_t model, we also used the function `Arima`. However, you must add the argument `model` indicating the previously estimated model.

```
> model_y <- Arima(gas[,"CO2"], model = model_x)


# CO2 ARIMA Model
Coefficients:
         ar1      ar2     ar3
      1.9696  -1.3659  0.3399
s.e.  0.0000   0.0000  0.0000

sigma^2 estimated as 9.56:  log likelihood=-756.47
AIC=1514.94   AICc=1514.96   BIC=1518.63
```

c. Save the residuals from both models

The residuals can be obtained by the function `resid()` and will be saved with the following names: `alpha` and `beta` for the X and Y models respectively.

```
> alpha <- resid(model_x)
> beta <- resid(model_y)
```

d. Estimate the CCF using the residuls obtained from (c)

After the pre-whitening, we can calculate the CCF between residues using the `ccf()` function from the package **stats**. The CCF shows the correlation between Y_t and the lags of X_t. Note that there is no significant correlation between Y_t and X_t in the present time (t = 0) and the first significant correlation is given for t = 3, i.e. between Y in the present time and X lagged in 3 lags (figura 3).

```
> ccf(beta, alpha, xlim = c(0,20))
```

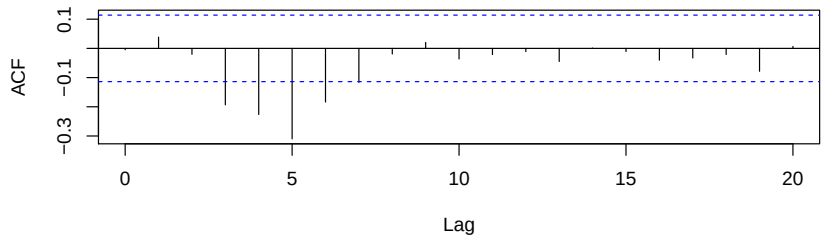

Figura 3: CCF: Input Gas Rate (X) and CO_2 (Y)

2.2 Identify r, s and b

Using estimated CCF, we can get the r, s and b orders:

- b: it refers the first significant lag. It represents the first X lag in the model. In this case, $b = 3$.
- s: number of increasing *lags* after b. It represents the next X lags that will enter the model. Thus, $s = 2$.
- r: how we have exponential decrease after increase lags, $r = 1$.

So, the model has X_{t-3}, X_{t-4} e X_{t-5} and the $f(X_t)$ is defined as

$$f(X_t) = \frac{(w_0 + w_1 L + w_2 L^2)}{(1 - \delta_1 L)} X_{t-3}$$

2.3 Estimating the Box & Jenkins model with Transfer function

Once $f(X_t)$ is identified, we can estimate the model with transfer function. Initially, it is necessary to identify the order of the ARIMA model for the seriesY_t, as was done for x_t.

```
> acf(gas[,"CO2"], lag.max = 36, main = "ACF: CO2 (Y)")
> pacf(gas[,"CO2"], lag.max = 36, main = "PACF: CO2 (Y)")
```

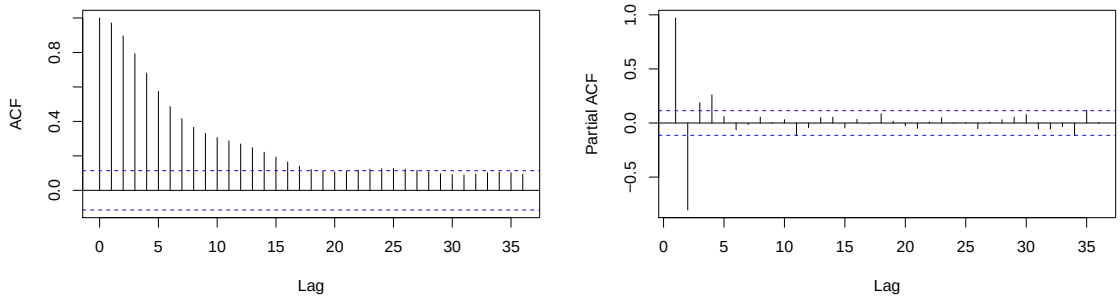

Figura 4: ACF and PACF: CO_2 (Y)

The exponential decrease of the ACF and the abrupt cut of the PACF in the lag 2 (figura 4) suggest an ARIMA (2,0,0) model. We estimate the model with transfer function using the function `arimax()` of the **TSA** package. The arguments used in the function are:

- `x`: dependent Time Series (Y_t);
- `order`: Y_t ARIMA model order;
- `xtransf`: independent time series (X_t) lagged in b;
- `transfer`: r and s values (`list(c(r,s))`).

Lagged variable X_t (3 lags) can be made through the function `lag()` of the **stats** package .

```
> new_x <- lag(gas[,"InputGasRate"], k = -3)
```

Since three values of X_t have been "lost"to estimate the model, it is necessary to cut the first three values Y_t for the two data sets to have the same size.

```
> new_gas <- na.omit(cbind(new_x, gas[,"CO2"]))
> colnames(new_gas) <- c("InputGasRate", "CO2")
> head(new_gas)
```

```
 InputGasRate  CO2
       -0.109 53.5
        0.000 53.4
        0.178 53.1
        0.339 52.7
        0.373 52.4
        0.441 52.2
```

With the data in the correct form, we estimate a model for the dependent variable CO_2.

```
> model_tf <- arimax(x = new_gas[,"CO2"], order = c(2,0,0),
+                      xtransf = new_gas[,"InputGasRate"],
+                      transfer = list(c(1,2)))
```

```
# BJTF Model
Coefficients:
         ar1      ar2  intercept  T1-AR1   T1-MA0   T1-MA1   T1-MA2
      1.5272  -0.6288    53.3618  0.5490  -0.5310  -0.3801  -0.5180
s.e.  0.0467   0.0495     0.1375  0.0392   0.0738   0.1017   0.1086

sigma^2 estimated as 0.0571:  log likelihood = 2.08,  aic = 9.83
```

The model can be represented by the following equation:

$$Y_t = 53.4 + \frac{(-0.5310 - 0.3801L - 0.5180L^2)}{(1 - 0.5490L)} X_{t-3} + \frac{1}{1 - 1.5272L + 0.6288L^2} e_t$$

2.4 Verify if the model is suitable

To assess whether the adjusted model is appropriate, the following analyses are proposed [3]:

- Residuals ACF;
- CCF between the residuals and the Pre-whitening auxiliary variable X_t

If there are correlation patterns, it is suggested that the model does not fit well, and therefore must be modified.

```
> residuals <- resid(model_tf)
> acf(residuals, na.action = na.omit, lag.max = 36)
> ccf(residuals, alpha, na.action = na.omit, ylab = "CCF",
+     main = "CCF: Residuals vs. alpha")
```

From the analysis of correlation functions (figura 5), we can conclude that this transfer function model is appropriate. The graph of the observed values versus those adjusted for the models with and without transfer function is displayed in Figure 6.

```
> model_y <- Arima(gas[,"CO2"], order = c(2,0,0), include.mean = T)
> fitted <- fitted(model_y)
> fitted_tf <- fitted(model_tf)
```

[3] for better diagnosis of residuals we should use, for example, the Ljung and Box (1978); Engle (1982); Jarque and Bera (1980) tests. In the case study presented in section three, the same are used

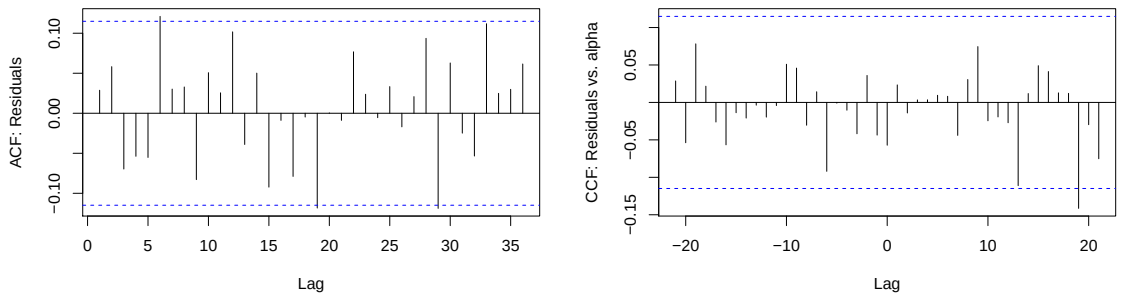

Figura 5: ACF: Resíduos

```
> ts.plot(gas[,"CO2"], fitted, fitted_tf, lty = c(1,3,2),
+         lwd = c(1,3,2), col = c(1,2,4))
> legend("bottomright", col = c(1,2,4), text.width = 35,
+        legend = c("Observed", "no TF", "TF"),
+        lty = c(1,3,2), lwd = c(1,2,3), cex = 0.7)
```

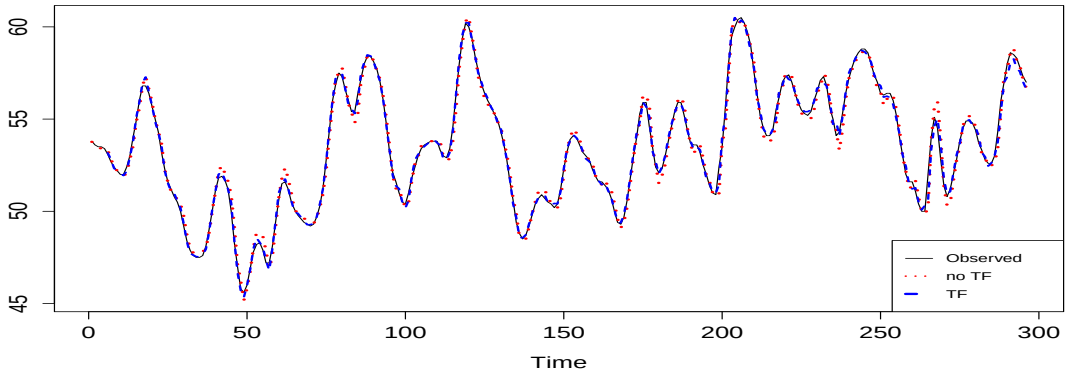

Figura 6: Observed values vs. fitted values

From the analysis of models with and without FT, we note that both are suitable for the modeling of CO_2. The results of the model with transfer function are less erratic and the MAPE between the predicted and observed values is 60% lower compared to the MAPE of the model without the FT. However, the MAPE of the two models are significantly small (0.3% and 0.5%). Another way to check which of the two models is more efficient to analyze their respective forecasts out of the sample. This analysis will be done in 3 section with a more current example.

3 Case Study: Predicting NIE using BJTF (South Subsysthem)

The objective of the hydroelectric planning of an electric power system is to determine the optimal strategy to electricity generation through the dispatch of the thermal and hydraulic generators,

aiming at minimizing the total cost of operation during the planning horizon. This planning is crucial to assure that the energy demand is met in a reliable and efficient way.

The Interconnected National Energy System (INES) is a large system that produces and transmits electric energy, where currently 93% of the generation come from water resources. The INES is composed of, and segmented in, four subsystems (or submarkets) that correspond to the regions of the interconnected systems: South, Southeast/Middle-East, Northeast and part of the North.

Taking into account the predominance in Brazil of hydraulic energy generation, the strong dependence of the hydrological regimes makes evident the uncertainty of the problem; this suggests the stochastic modeling of the affluences (e.g. Natural Inflow Energy), aiming at the optimization of the performance of system's operations, with the consequent reliability improvement and costs reduction (Ferreira et al., 2015a,b).

Given the uncertainties associated with natural phenomena and still under the influence of long-term phenomena such as El Niño, La Niña and Sunspots, the planning and operation of the Brazilian hydroelectric system becomes a complex problem, stochastic and of non-trivial solution. In this context, the aim of this study is to obtain the best forecasts of the NIE temporal series making use of the Box & Jenkins model with transfer function using as independent variables the SOI and Sunspots.

We have a monthly database with 77 years time span (1932-jan a 2008-dec). Our covariates are: (i) SOI: Southern Oscillation Index, is a measure of the high scale fluctuations in air pressure that occur between the West and the East of the tropical Pacific (i.e., South Oscillation state) during the episodes of the climate phenomena El Niño and La Niña. Traditionally, this index has been calculated based on the differences in air pressure among Tahiti and Darwin, Australia; and (ii) Sunspots, in general terms, sunspots are temporary phenomena on the photosphere of the Sun that appear visibly as dark spots compared to surrounding regions. The number of sunspots varies according to the approximately 11-year solar cycle. The databese can be downloaded direct from R running the code lines bellow.

```
> devtools::source_url("http://git.io/vCjzX")
> south
```

As can be seen in figure 7, the series appear to be stationary, although there are a few outliers. An analysis of the unit root tests Augmented Dickey-Fuller (ADF) shows that there is no evidence of a unit root in any of the three time series (Table 1). Thus, it is not necessary to tell them apart.

TS	**Equation**	**Lag**	**Statistic τ**	**Critical value**
NIE	Trend	15	-7.3913	-3.41
SOI	Trend	13	-7.6717	-3.41
SS	Drift	20	-6.0147	-2.86

Tabela 1: Augmented Dickey-Fuller Test

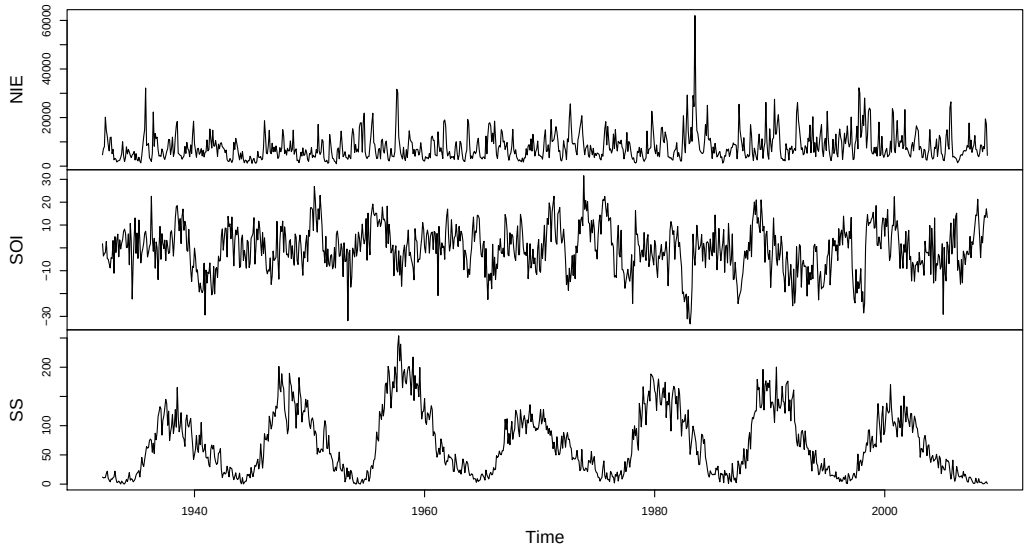

Figura 7: NIE, SOI and SS

To adjust the BJ model with TF follows the steps defined in section 2. As there is more than one covariate, the pre-whitening procedure has to be done twice. Note that in order to stabilize the variance, a log transformation was applied in NIE.

3.1 Pre-whitening

The ARIMA models identified by the independent SOI and SS series are, respectively, an ARIMA (3,0,0) without drift and ARIMA (3,0,0) with drift. The coefficients of the models can be seen below.

```
# SOI ARIMA Model
Coefficients:
         ar1     ar2     ar3
      0.4189  0.1831  0.1714
s.e.  0.0324  0.0347  0.0324

sigma^2 estimated as 54.81:  log likelihood=-3161.26
AIC=6330.52   AICc=6330.56   BIC=6349.84

# SS ARIMA Model
Coefficients:
         ar1     ar2     ar3  intercept
      0.6469  0.0921  0.2247    66.0899
s.e.  0.0320  0.0383  0.0320    15.5028

sigma^2 estimated as 319.8:  log likelihood=-3977.02
AIC=7964.05   AICc=7964.11   BIC=7988.19
```

The coefficients of the models are significant. With respect to the diagnosis, while according to figure 8 (FAC residuals) there are significant autocorrelations in some lags, it is believed that this occurs due to the number of observations, making the confidence interval narrower . Thus, it was considered both models are suitable. Further, the NIE series was filtered by the two models presented previously and the residuals were stored.

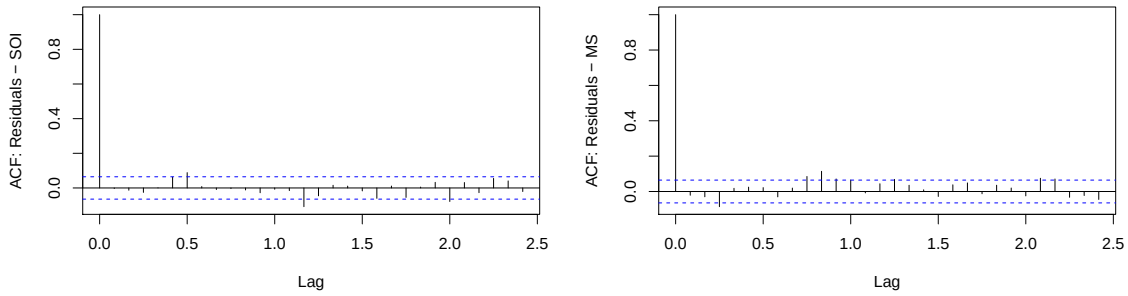

Figura 8: Residuals ACF: SOI and Sunspot

3.2 Identify r, s and b

The CCF between pre-whitening variables and NIE did not result in significant lags indicating that past or present values of covariates would be related to the present NIE values (Figure **??**). To solve this problem, we set up a model with FT that minimizes the criterion of Schwarz (BIC) information from various combinations between the two lagged covariates from lag 1 to lag 12, and that all were significant in the model.

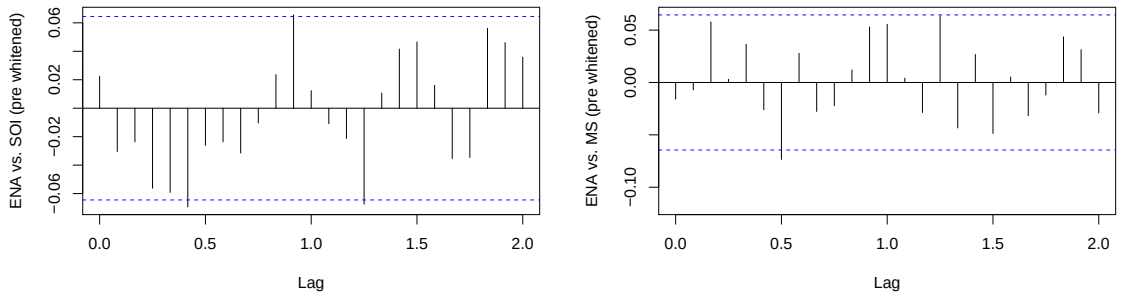

Figura 9: CCF: NIE vs. SOI and Sunspot (pre-whitening)

To create the lagged variables with lag i, $i = 1, ..., 12$, the following codes have been used:

```
> SOI_i <- lag(SOI, k = -i)
> SS_i <- lag(SS, k = -i)
```

3.3 Estimating the Box & Jenkins model with transfer function

The estimated ARIMA model for NIE (log transformed) using the ACF and PACF (figura 10) was am ARIMA (1,0,0)(0,0,1).

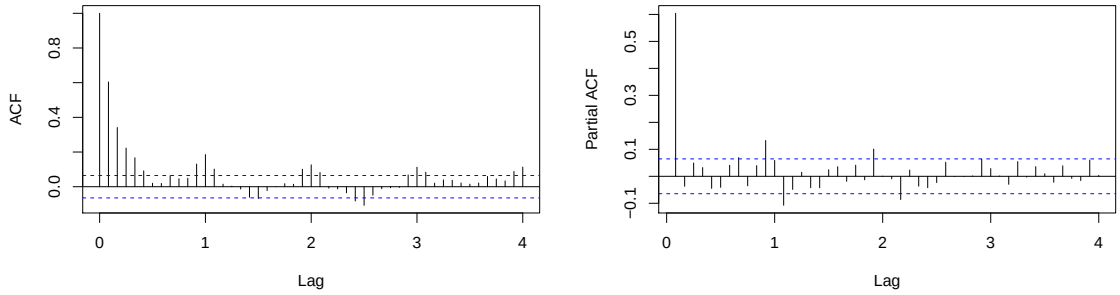

Figura 10: ACF and PACF: log(NIE)

Table 2 shows three adjusted models with FT and the BJ traditional model. The first model was obtained considering a minimizing of the BIC information criterion and the significance of the parameters. The second and third models were obtained by just separating covariates of the first model. Note that the model 4 model has the smallest information criterion, although there is no significant change when compared to the other three models.

Modelo	Covariáveis	BIC
1	SOI(-7) SS(-12)	1389.66
2	SOI(-7)	1387.11
3	Sunspot(-12)	1387.58
4	without covariates	1384.82

Tabela 2: Estimated models and Information Criterium

It is important to inform that the function `arimax` does not allow predictions through functions already implemented in R. In this case, the function `Arima` was used, with the argument `xreg` to specify covariates for the adjustment of the models 1, 2 and 3. This form of adjustment considers the r term null[4] from the $f(X_t)$ - equation 2. Therefore, the transfer function can now be defined as follows (in the case of only one variable):

$$f(X_t) = (w_0 + w_1 L + ... + w_s L^s) X_t \tag{4}$$

Model 1, for example, was adjusted in the following manner:

```
> model_tf1 <- Arima(new_south[,"NIE"], order = c(1,0,0), seasonal = c(0,0,1),
+                    lambda = 0, xreg = new_south[,c("SOI_7", "SS_12")])
```

[4]remember that the term r had been identified as zero when analyzing the CCF.

Thus, the thee models can be defined as:

Modelo 1: $Y_t = 8.6773 - 0.0046SOI(-7) + 0.0014SS(-12) + \frac{(1-0.1714L^{12})e_t}{1-0.5972L}$

Modelo 2: $Y_t = 8.7762 - 0.0045SOI(-7) + \frac{(1-0.1667L^{12})e_t}{1-0.5991L}$

Modelo 3: $Y_t = 8.6810 - 0.0014SS(-12) + \frac{(1-0.1698L^{12})e_t}{1-0.6017L}$

Modelo 4: $(1-0.6037L)Y_t = (1-0.1651L^{12})e_t$

where Y_t is a NIE time series.

3.4 Verify if the model is suitable

Besides to verify if there exists some linear residuals correlation through residuals ACF, others statistical test were done to confirm if the model is suitable. Below we applied Ljung-Box, Jarque-Bera and ARCH tests to verify autocorrelation, normality and heteroskedastic from residuals, respectively.

```
> # Ljung-Box test for autocorrelation
> Box.test(residuals, lag = i, type = "Ljung-Box")
> # Jarque-Bera test for normality
> tseries::jarque.bera.test(residuals)
> # Arch test for heteroskedastic
> FinTS::ArchTest(residuals)
```

The Ljung-Box for 24 first lags showed there is no residuals correlation evidence. Jarque-Bera test showed, even with outliers corrections, that the residuals are not normal. This kind of caracteristic were already expected once NIE has assymetric distribution. Arch test, showed there is no heteroskedastic evidence.

Thus, the models are similar with respect the diagnosis since the tests results are the same. To evaluate the better model an out-of-sample forecast will be done.

3.5 Out-of-sample model performance

It was done forecasts 6-steps-ahead (2009-jan to 2009-jun) for all models using **forecast** package (Hyndman and Khandakar (2008)). How the covariates were lagged in at least 7 lags it was not necessary to predict them. Covariates future values were specified through `xreg` argument.

```
> SOI_7_fcst <- window(south[,"SOI"], start = c(2008,6), end = c(2008,11), freq = 12)
> SS_12_fcst <- window(south[,"SS"], start = c(2008,1), end = c(2008,6), freq = 12)
> south_fcst <- ts(cbind(c(SOI_7_fcst), c(SS_12_fcst)), start = c(2009,1), freq = 12)
```

```
> colnames(south_fcst) <- c("SOI_7", "SS_12")
> fcst1 <- forecast(model_tf1, h = 6, level = 0.95, xreg = south_fcst)
> fcst2 <- forecast(model_tf2, h = 6, level = 0.95, xreg = south_fcst[,"SOI_7"])
> fcst3 <- forecast(model_tf3, h = 6, level = 0.95, xreg = south_fcst[,"SS_12"])
> fcst4 <- forecast(model_tf4, h = 6, level = 0.95)
```

	Observed	**Model TF1**	**Model TF2**	**Model TF3**	**Model TF4**
jan/09	5702	4816.438	5013.87	4960.932	5155.946
feb/09	5018	4902.927	5204.699	5002.253	5299.537
mar/09	4492	4898.023	5198.68	5117.163	5419.337
abr/09	1865	5126.876	5516.195	5486.839	5885.795
may/09	2635	5575.077	6026.881	5915.264	6377.187
jun/09	3877	5695.744	6165.191	6135.452	6620.883

Tabela 3: Forecasts 6-steps-ahead for the NIE from Southwest Subsystem

Although the model performance were similar in-sample, when the out-of-sample forecasts are compared, we can observe that the BJ model with the covariates SOI and Sunspot had better performance than traditional SARIMA. As we can observe in table 4, there is n important decrease in Mean Absolute Percentage Error (MAPE).

Models	**MAPE (%)**
1	60.04
2	69.17
3	67.36
4	77.37

Tabela 4: Mean Absolute Percentage Error (MAPE)

4 Final remarks

This article achieved its goal showing the Box & Jenkins with transfer function steps using the R software. As observed in the whole article, this is not a trivial task and a large number of packages (eg. **TSA, forecast**) were used to achieve this goal.

Two more points should be highlighted: first, the BJTF application to the real problem, which is the NIE forecast in the Brazilian South, very hard solution task. Second, there is no similar publication who covers BJ with TF in R software. This allows new users replicate the presented studies and estimate the proposed model with new variables.

Future efforts will be direct to improve the NIE time series forecast. This is, even the BJTF has increased the model performance, the forecast error is still big. In this context, estimate the model with new variables and/or choose a new model are the ways to be follow.

Referências

H. Akaike. Information theory and an extension of the maximum likelihood principle. *2nd International Symposium on Information Theory*, 1973.

G. E. P. Box and G. M. Jenkins. *Time Series Analysis forecasting and control*. Holden Day, San Francisco, 1970.

Kung-Sik Chan and Brian Ripley. **TSA***: Time Series Analysis*, 2012. URL `http://CRAN.R-project.org/package=TSA`.

D. A. Dickey and W. A. Fuller. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 1979.

R. F. Engle. Autoregressive conditional heteroskedasticity with estimates of the variance of united kingdom inflation. *Econometrica*, 50, 1982.

P. G. C. Ferreira, F. L. C. Oliveira, and R. C. Souza. The stochastic effects on the brazilian electrical sector. *Energy Economics*, 49, 2015a.

P. G. C. Ferreira, R. C. Souza, and A. L. M. Marcato. The par(p) interconfigurations model used by the brazilian electric sector. *International Journal of Electrical Power & Energy Systems*, 73, 2015b.

Spencer Graves. **FinTS***: Companion to Tsay (2005) Analysis of Financial Time Series*, 2014. URL `http://cran.r-project.org/package=FinTS`.

J. D. Hamilton. *Time Series Analysis*. Princeton University Press, 1994.

D. M. Hanssens. Bivariate time-series analysis of the relationship between advertising and sales. *Applied Economics*, 12, 1980.

R. J. Hyndman and Y. Khandakar. Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, 27, 2008. URL `http://www.jstatsoft.org/v27/i03/`.

Rob J. Hyndman. **forecast***: Forecasting Functions for Time Series and Linear Models*, 2015. URL `http://cran.r-project.org/package=forecast`.

C. M. Jarque and A. K. Bera. Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economic Letters*, 1980.

K.S. Kannan and A. J. Farook. Transfer function modeling for global warming. *International Journal of Scientific Research*, 2, 2013.

C. Keeble. *The R primer*. Journal of Applied Statistics, 2012.

G. M. Ljung and G. E. P. Box. On a measure of a lack of fit in time series models. *Biometrika*, 1978.

A. McDowell. From the help desk: Transfer functions. *The Stata Journal*, 2002.

D. Meyer. Naive time series forecasting methods. *R News*, 2(2):7–10, June 2002.

F. J. Nogales and A. J. Conejo. Electricity price forecasting through transfer function models. *Journal of the Operational Research Society*, 57, 2006.

P. C. Phillips and P. Perron. Testing for a unit root in time series regression. *Biometrika*, 1988.

R Core Team. **graphics***: The R Graphics Package*, 2015a.

R Core Team. **stats***: The R Stats Package*, 2015b.

B. D. Ripley. Time series in R 1.5.0. *R News*, 2(2):2–7, June 2002.

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6, 1978.

L. A. Terry, M. V. Pereira, T. A. Neto, L. F. Silva, and P. R. Sales. Coordinating the energy generation of the brazilian national hydrotherma electrical generating system. *Interfaces*, 16, 1986.

A. Traplett, K. Hornik, and B. LeBaron. **tseries***: Time Series Analysis and Computational Finance*, 2015. URL `http://cran.r-project.org/package=tseries`.