Trabalho de Econometria

Gustavo Alovisi

28 de novembro de 2017

Modelagem ARIMA

Introdução:

O presente trabalho busca realizar uma modelagem ARIMA dado a metodologia de Box and Jenkins. Primeiramente, selecionamos uma série temporal não-estacionária (número de vendas de casas nos EUA - mensalmente: 1982-2004). Após isto, são realizados testes de estacionariedade para a série temporal, assim como transformações (diferenças) para tornar a série escolhida estacionária. A partir da série estacionária, realiza-se então a seleção do modelo de maior ordem, os testes de raiz unitária da série, a comparação dos Critérios de Informação e a análise de resíduos dos modelos candidatos a melhor escolha. Após a seleção dos modelos finais, analisamos a relação entre o modelo gerado pela função auto.arima() e o modelo final que escolhemos, bem como o teste arch para heterocedasticidade condicional e a adição de um regressor (US Treasure Bond Yields, 10 years) no modelo SARIMA. Também é feita a previsão 15 meses a frente dado o modelo SARIMAX ajustado e uma comparação entre erros de previsão do modelo SARIMAX e os erros de previsão do modelo de Suavização Exponencial gerado pela função ets(). Por fim, estimamos a estatística U de Theil para a qualidade de previsão do modelo.

As séries foram coletadas pelo site do Banco Central de St. Louis (FRED)[1]. As libraries utilizadas estão no final do trabalho.

[1] *HSN1FNSA https://fred.stlouisfed.org/series/HSN1FNSA?utm_source=series_page&utm_medium=related_content&utm_term=other_formats&utm_campaign=other_format

**Long-Term Government Bond Yields: 10-year https://fred.stlouisfed.org/series/IRLTLT01USM156N

Definição e importação da série temporal

Como início do trabalho, vamos primeiro importar para uma estrutura de dados xts nossa série temporal não estacionária do banco de dados do FRED, através da função getSymbols(). A série representa no eixo y o número de casas vendidas nos EUA (em milhares), possui periodicidade mensal e compreende o período de 1982-10-01 a 2004-11-01. Importaremos também 15 observações a frente, a fim de servir como o nosso banco de testes para a futura previsão.

```
library(xts)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

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

```
library(quantmod)
```

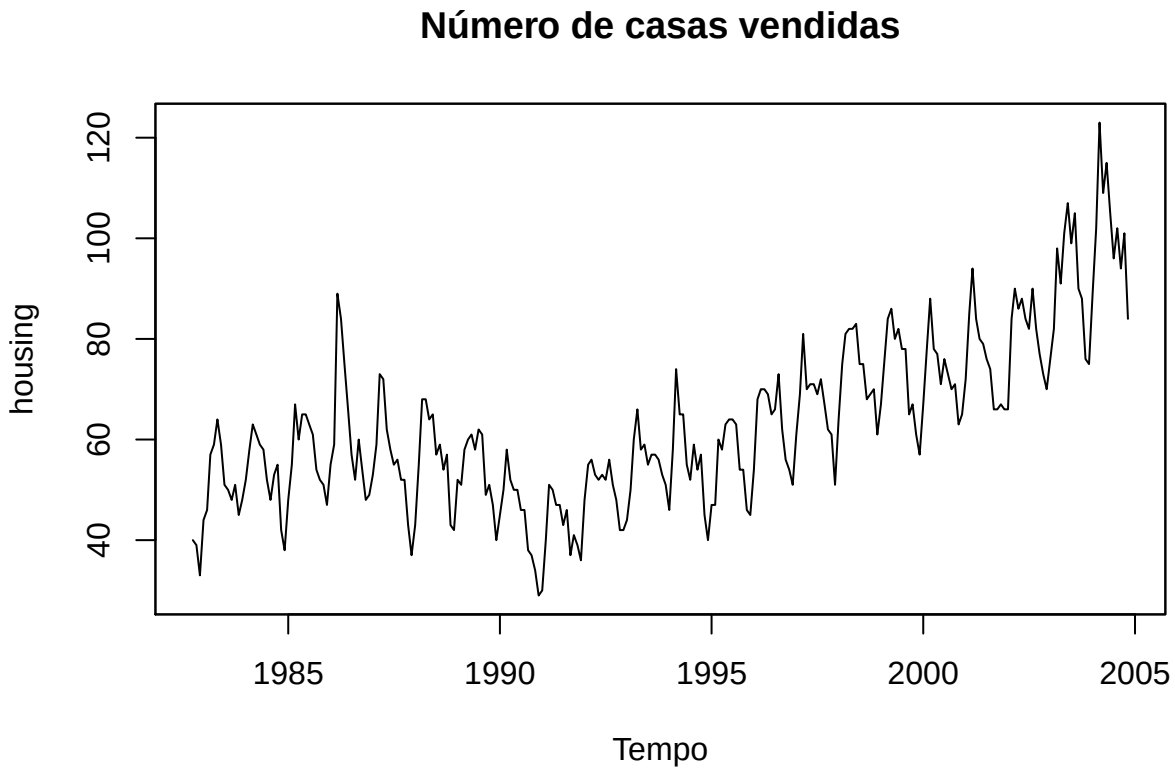
```
## Loading required package: TTR
```

```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
housing=getSymbols('HSN1FNSA', src="FRED", auto.assign = F)
```

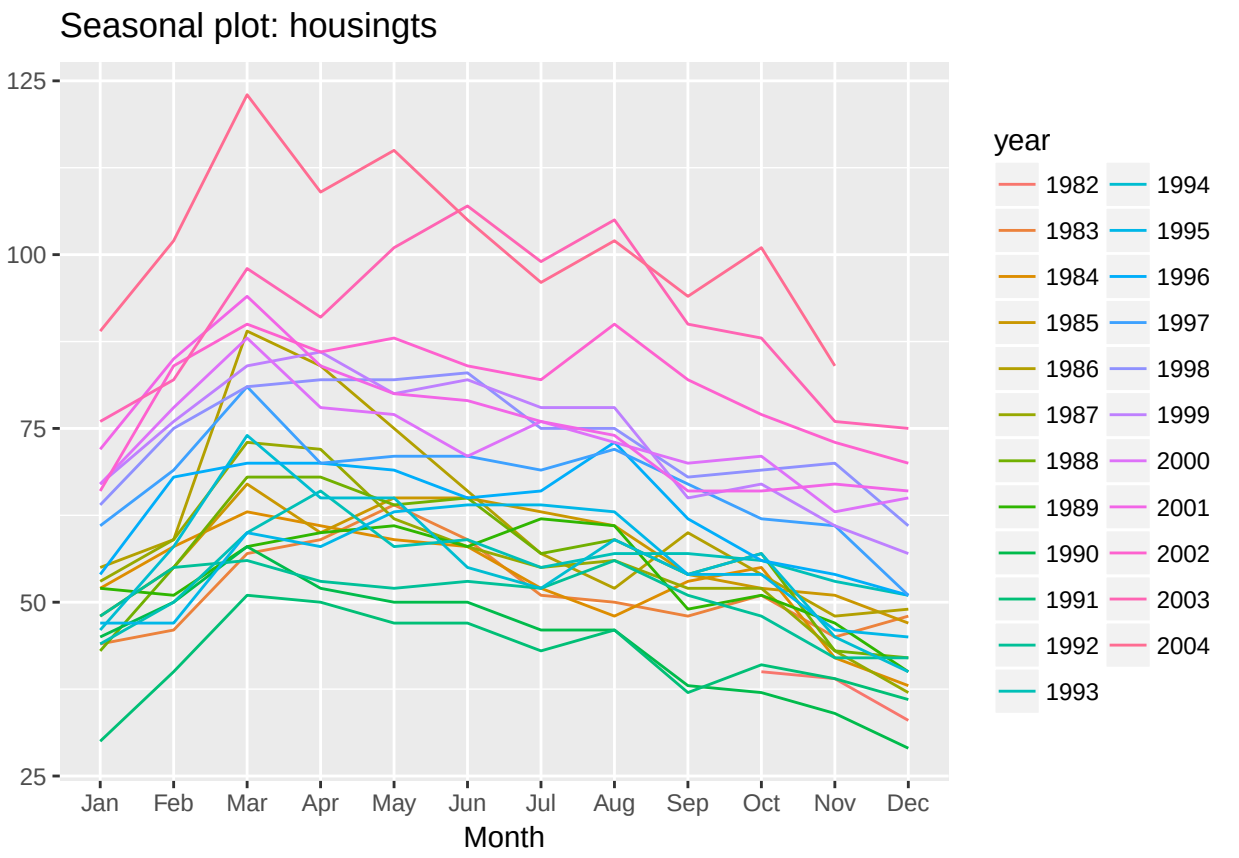
```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```
housingtest <- housing["2004-12-01/2006-02-01"]
housing <- housing["1982-10-01/2004-11-01"] # utilizaremos 15 obsevações a menos no final de nossa amos
 # carregaremos estas 15 observações para fora da amostra, a fim de testar a previsão com o modelo sari

plot.zoo(housing, main = "Número de casas vendidas", xlab= "Tempo")
```

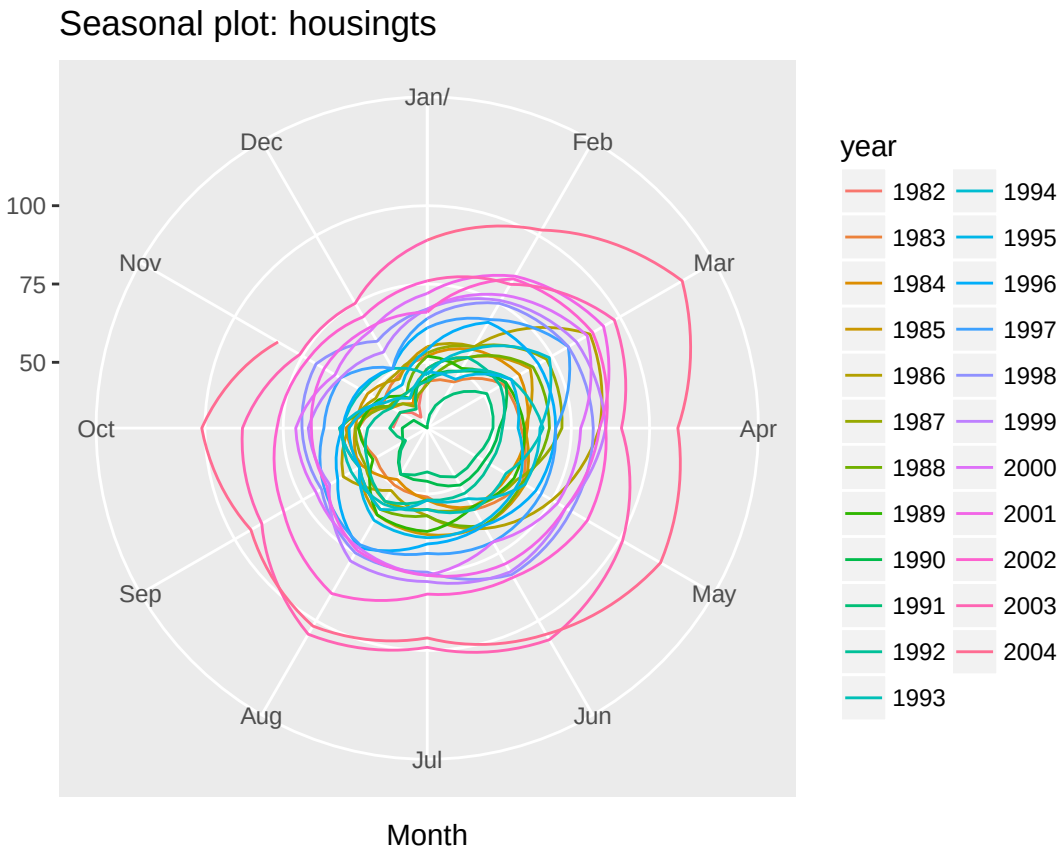


Utilizaremos a library ggplot para analisarmos a sazonalidade de nossa série:

```
library(ggplot2)
library(forecast)
housingts <- ts(housing[,1], start = c(1982,10), frequency = 12)
#tail(housingts)
ggseasonplot(housingts, polar = F)
```




```
ggseasonplot(housingts, polar = T)
```



A partir da representação em linhas, do plot de linhas sazonal e do plot sazonal de coordenadas polares, podemos claramente ver uma sazonalidade na série temporal, com o mês de Junho atingindo o pico de vendas de casas, em contraste com Fevereiro e Março que aprensetam o menor numero de vendas. O gráfico em coordenadas polares plota o eixo X (tempo) como um círculo de ângulo phi e raio 'r' para cada periodo sazonal presente na série temporal. Como o valor y da série cresce em quase todo o período, o raio também cresce. A conversão para coordenadas polares é dada por phi=arctg(y,x) e r = raiz(x^2+y^2).

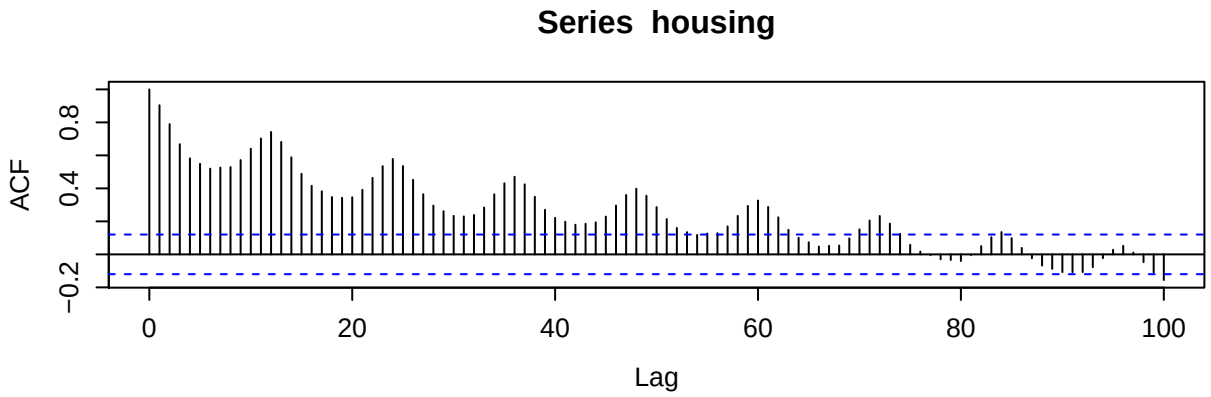
Para estudarmos a hipótese de sazonalidade e não estacionariedade, plotaremos a Função de Autocorrelação e Autocorrelação Parcial da série de vendas.

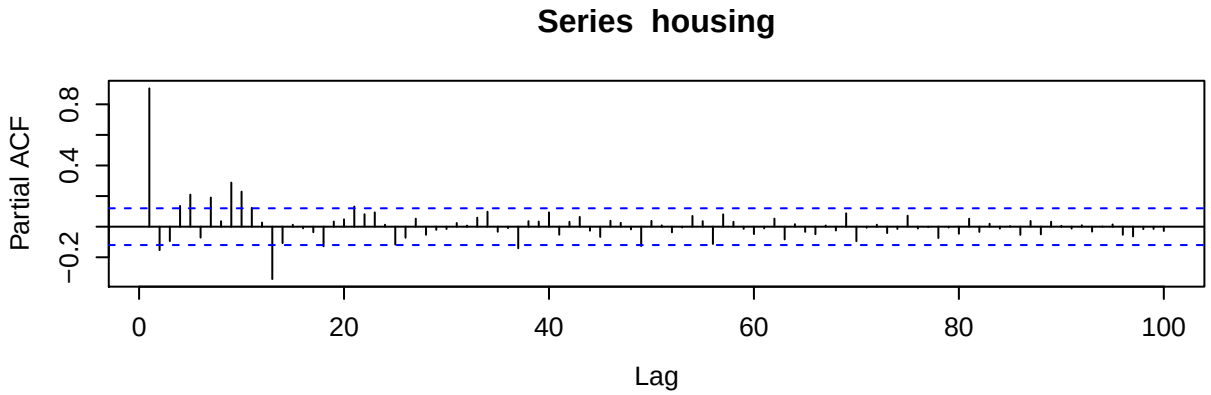
```
library(astsa)
```

```
##
## Attaching package: 'astsa'
```

```
## The following object is masked from 'package:forecast':
##
##     gas
```

```
par(mfrow=c(2,1), mex = 0.8, cex = 0.8)
acf(housing, lag.max = 100)
pacf(housing, lag.max = 100)
```



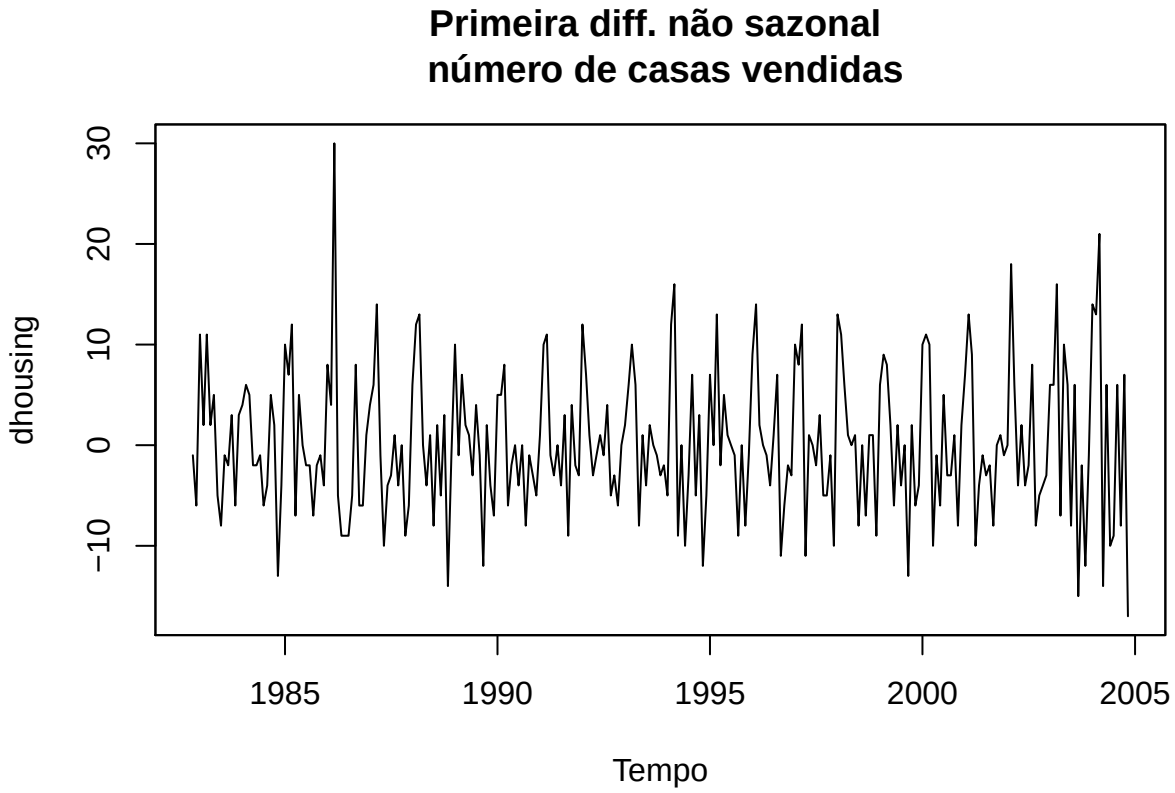


Notamos que a função de autocorrelação (ACF) possui um decaímento lento e não exponencial, com períodos de maior e menor autocorrelação da série.

Como primeiro passo para torná-la estacionária, tiraremos a primeira diferença não sazonal da série.

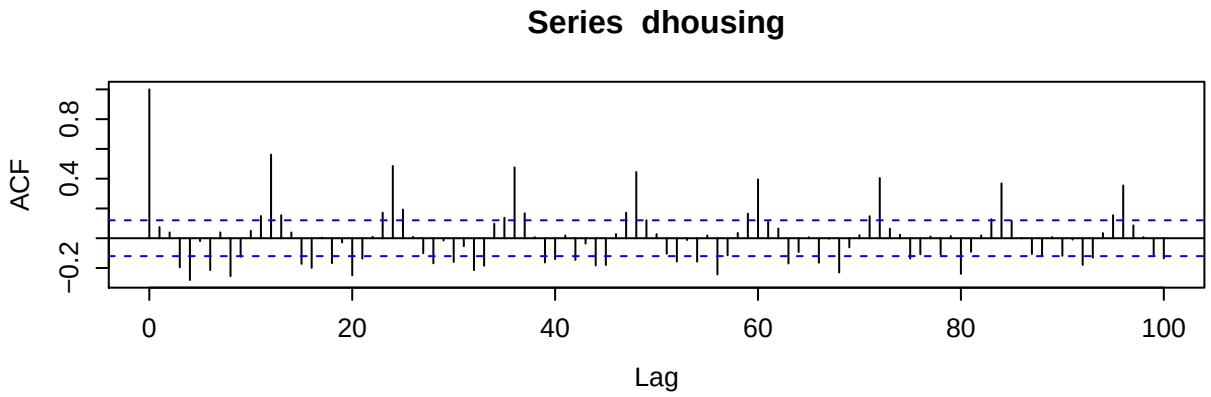
Após tirarmos a primeira diferença da série, notamos que ela exibe um comportamento parecido a de um White-Noise (média 0, variancia sigma^2), porém com uma sazonalidade de tempos em tempos representada pelo constante pico->queda nos valores diferenciados.

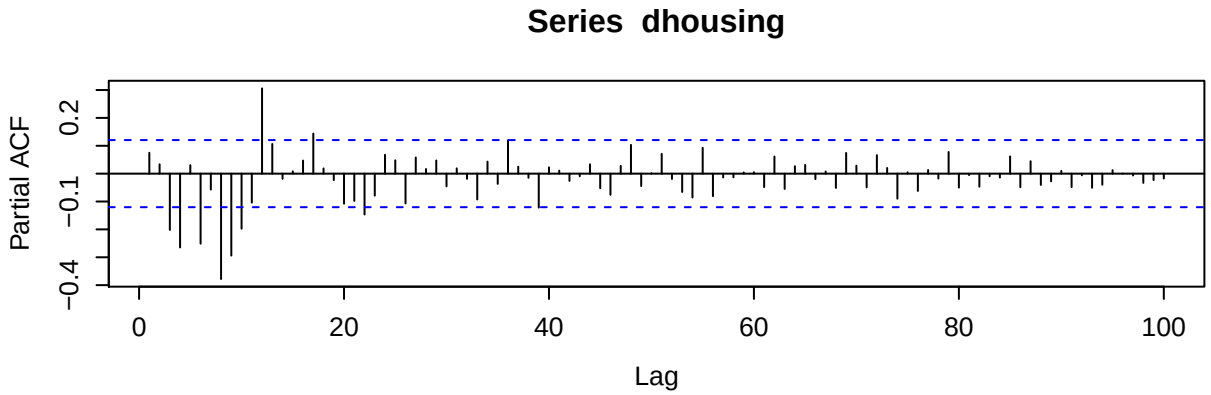
```
dhousing <- diff.xts(housing, na.pad = F)
plot.zoo(dhousing, main = "Primeira diff. não sazonal \n número de casas vendidas", xlab= "Tempo")
```



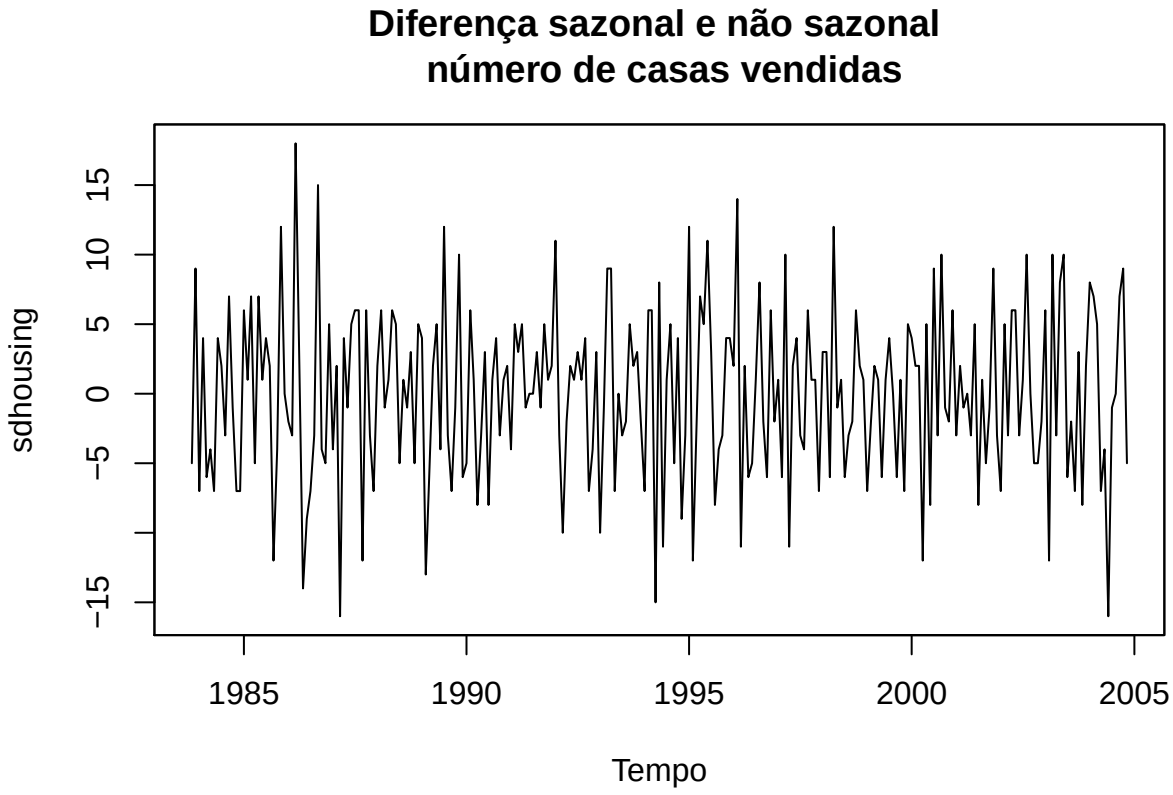
Em seguidas exibiremos a ACF e PACF da série diferenciada. A partir das funções ACF e PACF da série diferenciada(1), notamos que de fato ainda existe uma sazonalidade de 12 em 12 meses, apesar da parte não-sazonal parecer estacionária. Por isso, tiraremos a diferença sazonal de lag = 12.

```
par(mfrow=c(2,1), mex = 0.8, cex = 0.8)
acf(dhousing, lag.max = 100)
pacf(dhousing, lag.max = 100)
```






```
sdhousing <- diff.xts(dhousing, lag= 12, na.pad = F)
plot.zoo(sdhousing, main = "Diferença sazonal e não sazonal \n número de casas vendidas", xlab= "Tempo"
```



Testes de Raíz Unitária

Após a remoção da sazonalidade, podemos ver que a série exibe um comportamento parecido com o de um White Noise, com média 0 e var sigma², sem sazonalidade aparente. Vamos então testar a hipótese de Raiz Unitária da série diferenciada sazonalmente e não sazonalmente - sdhousing - para nos certificarmos que ela pode ser dita estacionária.

Os testes utilizados serão o ADF (Augmented DFuller) e PP (Philips-Pherron). Vale a pena notar que o teste ADF pode ser realizado de três maneiras (i) sem drift e com tendência linear em t (ii) com drift e sem tendência linear em t (iii) com drift e tendência determinística em t. Procurei testar as três versões do teste DF conforme a página 86 da apostila do Guilherme. "Teste para raíz unitária", "Teste para raíz unitária com drift" e "Teste de raiz unitária com drift e tendência temporal deterministica". Para isso, procurei as funções que testavam cada caso e inclui no trabalho.

O primeiro CADFtest testa a hipótese [H0: possui raíz unitária] sem drift e sem tendência, o segundo com drift e sem tendência e o terceiro com drift e com tendência.

```
library(tseries)
library(CADFtest)
```

```
## Loading required package: dynlm
```

```
## Loading required package: sandwich
```

```
## Loading required package: urca
```

```
##tseries::adf.test(sdhousing, alternative = c("s"))   # Com drift e tendência  H0: tem raz unitária
CADFtest(sdhousing, type=c("none"), max.lag.y=5) # Sem drift e sem tendência H0: tem raiz unitária
```

```
##
##  ADF test
##
## data:  sdhousing
## ADF(5) = -10.638, p-value < 2.2e-16
## alternative hypothesis: true delta is less than 0
## sample estimates:
##     delta
## -2.521741
```

```
CADFtest(sdhousing, type = c("drift"), max.lag.y=5) # Com drift e sem tendência : H0: tem raiz unitaria
```

```
##
##  ADF test
##
## data:  sdhousing
## ADF(5) = -10.615, p-value < 2.2e-16
## alternative hypothesis: true delta is less than 0
## sample estimates:
##     delta
## -2.521873
```

```
CADFtest(sdhousing, type = c("trend"), max.lag.y = 5) #Com drift e com tendência: H0: tem raíz unitária
```

```
##
##  ADF test
##
## data:  sdhousing
## ADF(5) = -10.602, p-value < 2.2e-16
## alternative hypothesis: true delta is less than 0
## sample estimates:
##     delta
## -2.525377
```

```
pp.test(sdhousing) #philips-perron
```

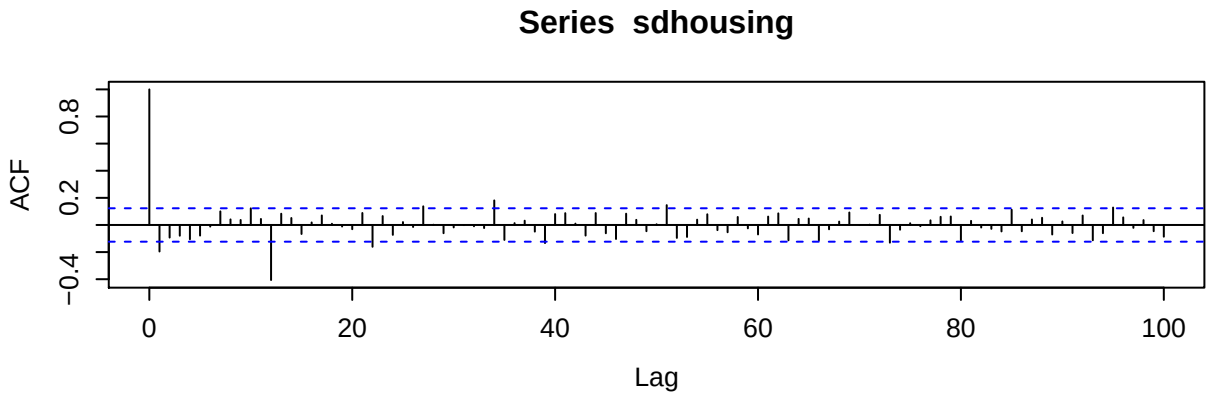
```
## Warning in pp.test(sdhousing): p-value smaller than printed p-value
```

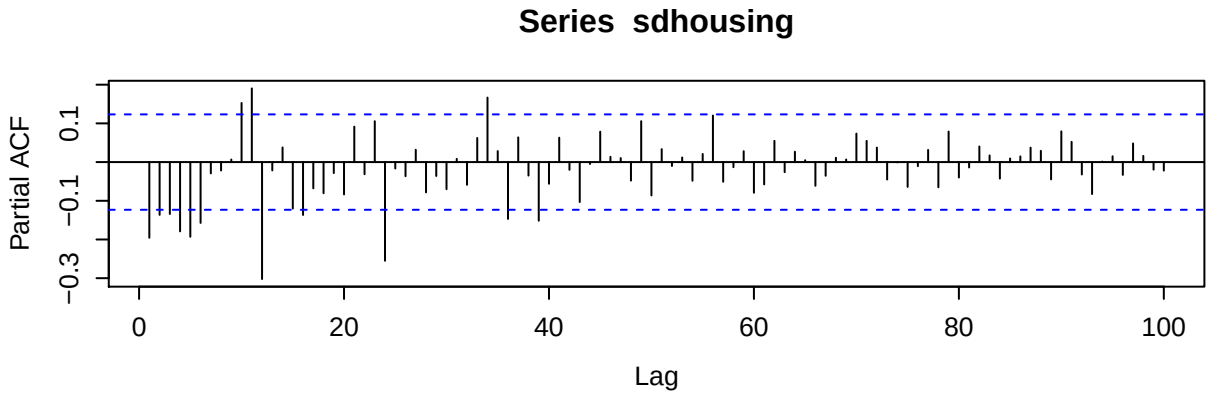
```
##
##  Phillips-Perron Unit Root Test
##
## data:  sdhousing
## Dickey-Fuller Z(alpha) = -239.68, Truncation lag parameter = 5,
## p-value = 0.01
## alternative hypothesis: stationary
```

A partir dos testes de raíz unitária ADF, rejeitamos H0 com 5% de chance de erro para os casos: com drift e tendência e com drift e sem tendência e sem drift e sem tendência, indicando que não há raíz unitária e a série sdhousing é de fato estacionáia. O PP-test também rejeita a hipótese H0 5% de raíz unitária com p-valor = 0.01 para todos os tipos de teste.

Agora que a série está estacionária, prosseguiremos a análise da ACF e PACF para a seleção e ajustamento de um modelo:

```
par(mfrow=c(2,1), mex = 0.8, cex = 0.8)
acf(sdhousing, lag.max = 100)
pacf(sdhousing, lag.max = 100)
```





Seleção do Modelo

A partir da análise das funções ACF e PACF, definimos o modelo de mais alta ordem: p=3 (PACF cai e volta a subir a partir de 4)

d=1 (diferenciado 1 vez)

q=1 (ACF não sazonal mostra 1 valor significativo)

P=3 (PACF tails-off a partir do lag sazonal 3)

D=1 (diferenciado sazonalmente 1 vez)

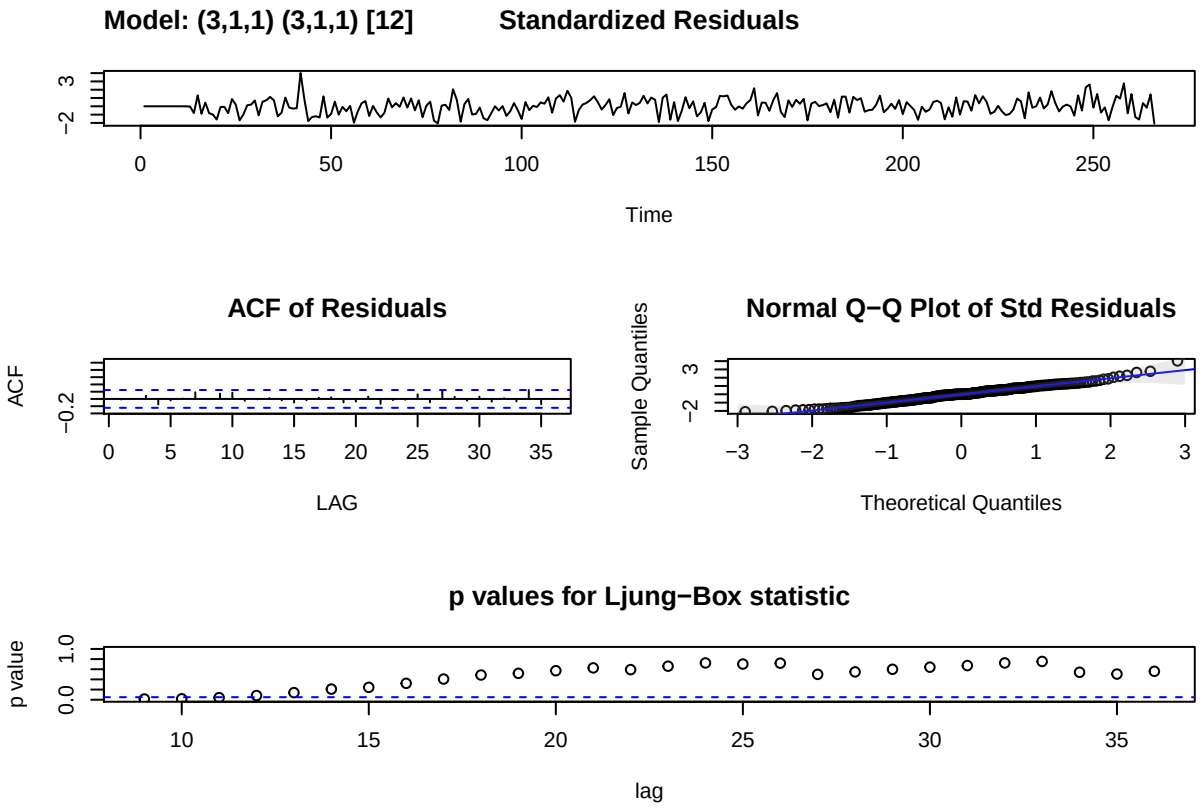
Q=1 (ACF sazonal mostra 1 valor significativo)

S=12 (lag sazonal = 12)

logo, **sarima(housing, p=3,d=1,q=1,P=3,D=1,Q=1,S=12)**

Vamos então ajustar o modelo de mais alta ordem utilizando a função sarima() da library astsa do professor Stoffer.

```
sarima(housing, p=3,d=1,q=1,P=3,D=1,Q=1,S=12, details = F)
```

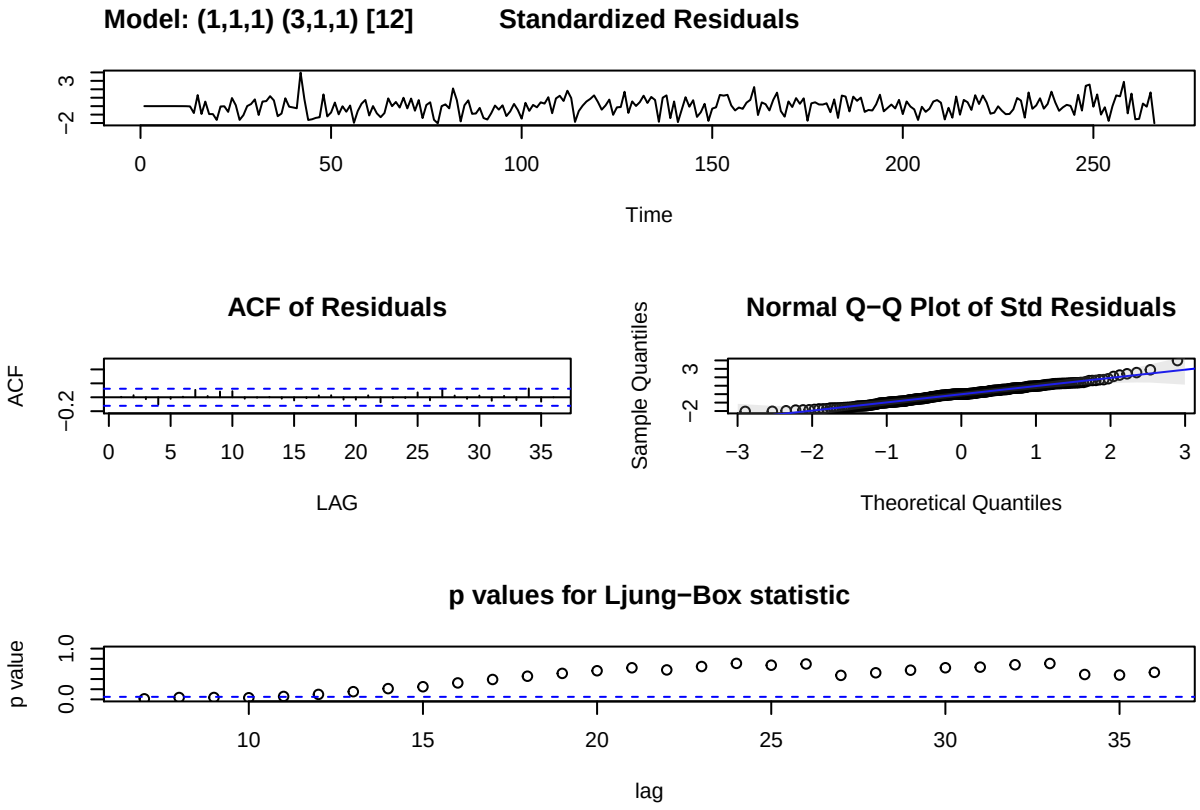



```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1     ar2      ar3      ma1    sar1     sar2     sar3     sma1
##       0.4289  0.0130  -0.1055  -0.7601  0.0128  -0.0925  -0.0514  -0.8087
## s.e.  0.1003  0.0741   0.0720   0.0816  0.1055   0.0902   0.0895   0.0925
##
## sigma^2 estimated as 19.83:  log likelihood = -744.55,  aic = 1507.1
##
## $degrees_of_freedom
## [1] 258
##
## $ttable
##      Estimate     SE t.value p.value
## ar1    0.4289 0.1003  4.2767  0.0000
## ar2    0.0130 0.0741  0.1757  0.8606
## ar3   -0.1055 0.0720 -1.4648  0.1442
## ma1   -0.7601 0.0816 -9.3171  0.0000
## sar1   0.0128 0.1055  0.1214  0.9035
## sar2  -0.0925 0.0902 -1.0255  0.3061
## sar3  -0.0514 0.0895 -0.5745  0.5661
```

```
## sma1  -0.8087 0.0925 -8.7450  0.0000
##
## $AIC
## [1] 4.047536
##
## $AICc
## [1] 4.057698
##
## $BIC
## [1] 3.15531
```

O modelo inicial apresenta Critérios de Informação AIC 4.047536, AICc 4.057698 e BIC 3.15531. Apesar do gráfico da função de autocorrelação dos resíduos parecer de um white-noise, o plot do teste de ljung-box indica incapacidade de rejeitar a hipótese nula de ausência de correlação dos resíduos até a ordem X apenas a partir do lag 12. Para melhorar nosso ajuste em relação aos Critérios de Informação e Ljung-box, vamos retirar um termo AR e continuar a análise.

```
sarima(housing, 1,1,1,3,1,1,12, details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
```

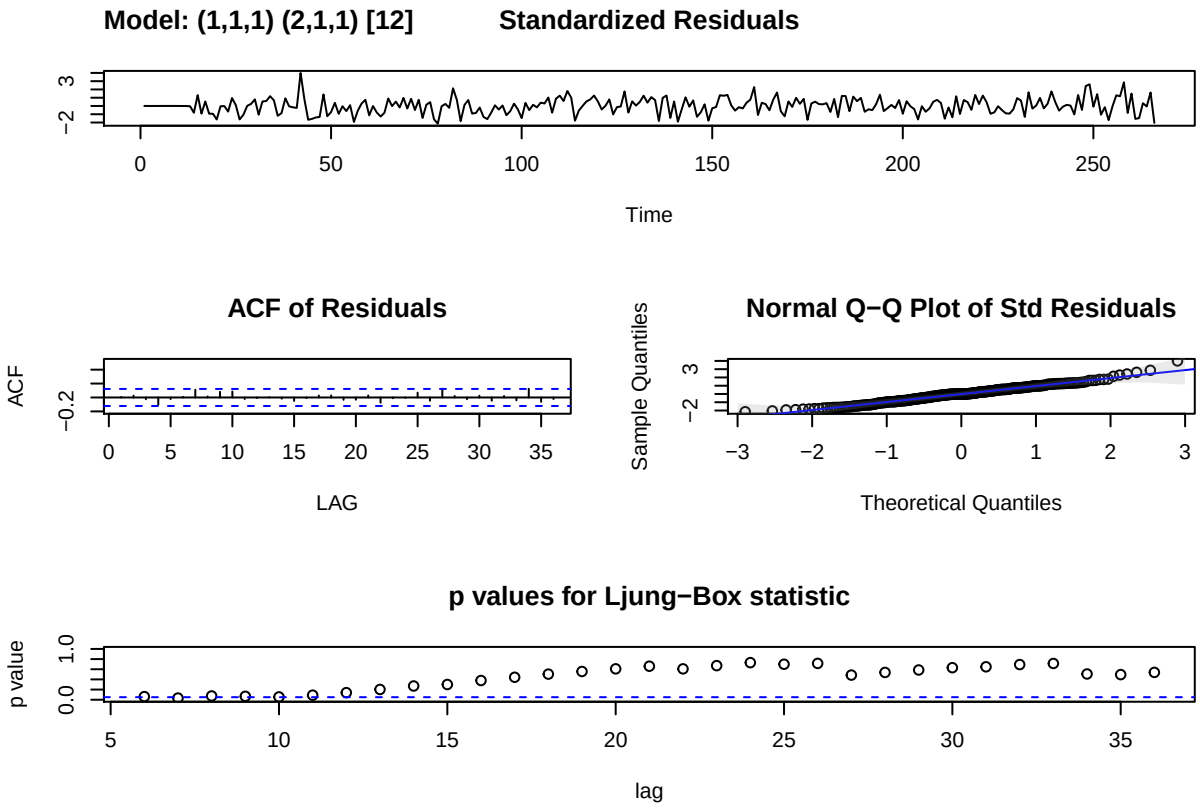
```
##          ar1      ma1    sar1     sar2     sar3     sma1
##       0.4876  -0.8204  0.0175  -0.1072  -0.0425  -0.8128
## s.e.  0.0892   0.0564  0.1048   0.0889   0.0895   0.0914
##
## sigma^2 estimated as 19.97:  log likelihood = -745.58,  aic = 1505.16
##
## $degrees_of_freedom
## [1] 260
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1    0.4876 0.0892   5.4645  0.0000
## ma1   -0.8204 0.0564 -14.5461  0.0000
## sar1   0.0175 0.1048   0.1670  0.8675
## sar2  -0.1072 0.0889  -1.2048  0.2294
## sar3  -0.0425 0.0895  -0.4753  0.6350
## sma1  -0.8128 0.0914  -8.8921  0.0000
##
## $AIC
## [1] 4.039192
##
## $AICc
## [1] 4.048342
##
## $BIC
## [1] 3.120022
```

Como podemos ver na análise de resíduos, a função de autocorrelação dos resíduos parece com a de um White noise e os p-valores para a estatística de Ljung-Box para os resíduos até a ordem determinada são razoáveis a partir do lag 12. Vamos continuar a procura de um melhor modelo que minimize os critérios de informação e exiba valores mais adequados para o teste de ljung box dos resíduos, principalmente dos lags iniciais.

Vamos agora retirar um componente SAR:

O novo modelo ficou então:

```
sarima(housing, p=1,d=1,q=1,P=2,D=1,Q=1,S=12,details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ma1    sar1     sar2     sma1
##       0.4883  -0.8212  0.0444  -0.0896  -0.8383
## s.e.  0.0891   0.0562  0.0863   0.0803   0.0702
##
## sigma^2 estimated as 19.97:  log likelihood = -745.69,  aic = 1503.39
##
## $degrees_of_freedom
## [1] 261
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1    0.4883 0.0891   5.4837  0.0000
## ma1   -0.8212 0.0562 -14.6239  0.0000
## sar1   0.0444 0.0863   0.5150  0.6070
## sar2  -0.0896 0.0803  -1.1165  0.2652
## sma1  -0.8383 0.0702 -11.9391  0.0000
##
## $AIC
```

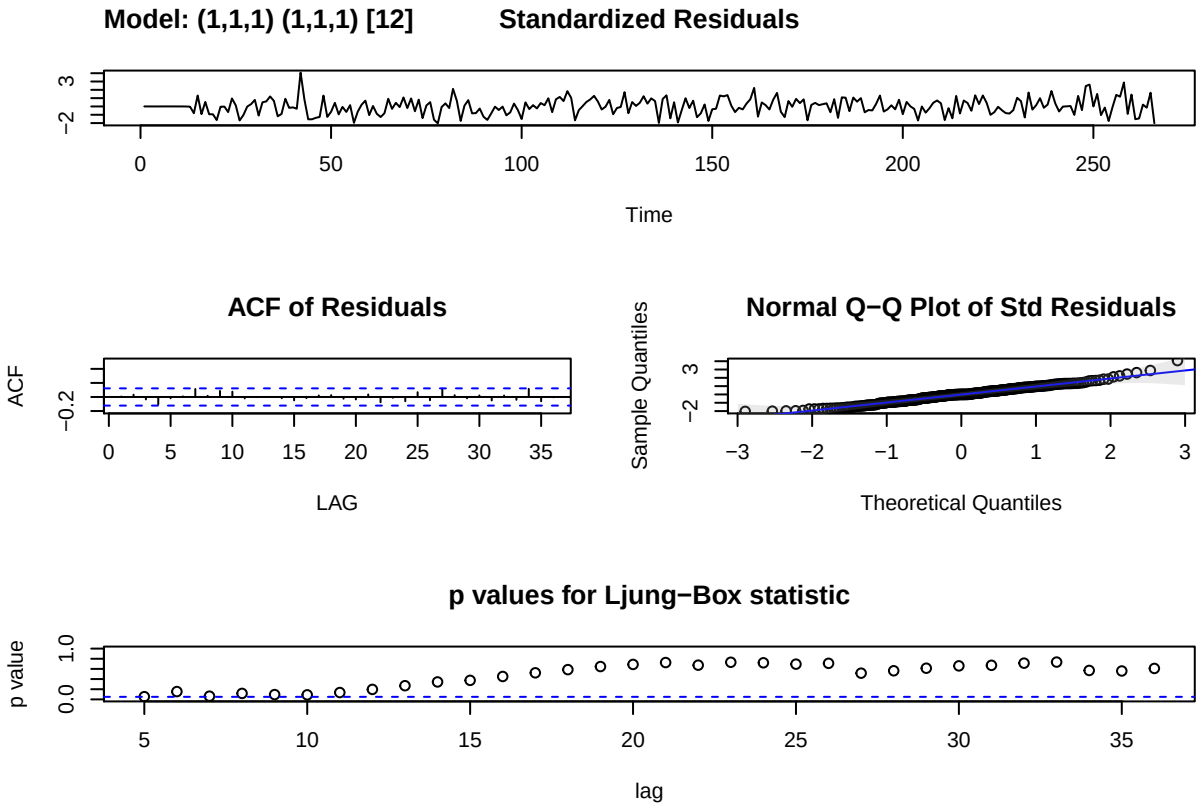
```
## [1] 4.031672
##
## $AICc
## [1] 4.04041
##
## $BIC
## [1] 3.099031
```

Notamos que esta modificação minimizou todos os 3 os critérios de informação AIC, AICc e BIC. O teste de Ljung-box para os resíduos também melhorou, não podendendo rejeitar a hipótese de ausência de autocorrelação dos resíduos até a ordem H a partir do lag 10.

Por fim, ao realizarmos a análise da ttable para os termos SARMA, notamos que ambos SAR1 e SAR2 ainda continuam exibindo p-valores altos (0.607 e 0.265).

Para continuar a adequação do modelo, removeremos mais um dos termos SAR e analisaremos novamente o resultado:

```
sarima(housing, p=1,d=1,q=1,P=1,D=1,Q=1,S=12,details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
```

```
##          ar1      ma1    sar1     sma1
##       0.4862  -0.8216  0.0772  -0.8765
## s.e.  0.0892   0.0562  0.0804   0.0597
##
## sigma^2 estimated as 20.02:  log likelihood = -746.3,  aic = 1502.61
##
## $degrees_of_freedom
## [1] 262
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1    0.4862 0.0892   5.4510  0.0000
## ma1   -0.8216 0.0562 -14.6217  0.0000
## sar1   0.0772 0.0804   0.9600  0.3379
## sma1  -0.8765 0.0597 -14.6929  0.0000
##
## $AIC
## [1] 4.026867
##
## $AICc
## [1] 4.035253
##
## $BIC
## [1] 3.080754
```

```
x<-arima(housing, order =  c(1,1,1), seasonal = list(order=c(1,1,1), period =12))
Box.test(resid(x), lag = 24, type=c("Ljung-Box"))
```

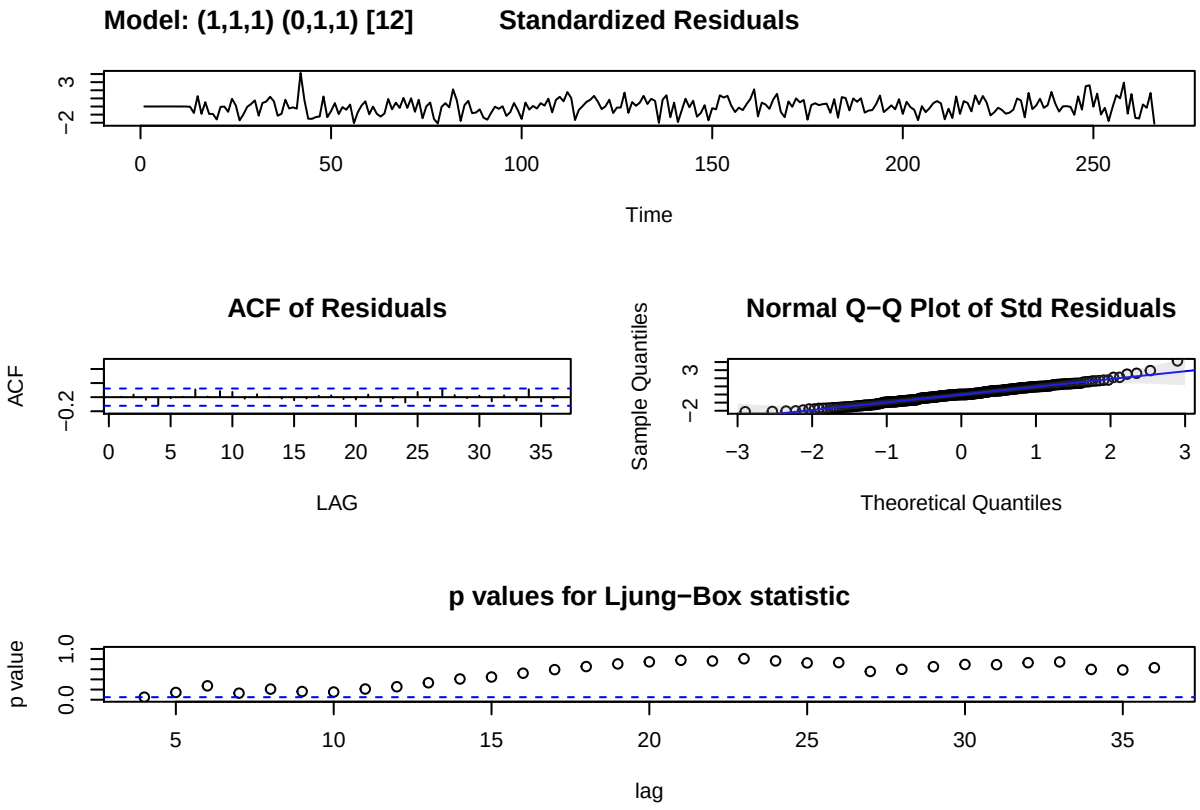
```
##
##  Box-Ljung test
##
## data:  resid(x)
## X-squared = 15.939, df = 24, p-value = 0.8903
```

Novamente notamos uma melhora na análise dos resíduos, com p-valores para a estatística de Ljung-box satisfatórios a partir do lag 5 e uma ACF dos resíduos muito semelhante a uma ACF de um White Noise.

Todos os 3 critérios de informação também foram reduzidos.

Vamos remover o último termo SAR e refazer a análise:

```
sarima(housing, p=1,d=1,q=1,P=0,D=1,Q=1,S=12,details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ma1     sma1
##       0.4806  -0.8160  -0.8465
## s.e.  0.0889   0.0559   0.0518
##
## sigma^2 estimated as 20.16:  log likelihood = -746.77,  aic = 1501.54
##
## $degrees_of_freedom
## [1] 263
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1    0.4806 0.0889   5.4035       0
## ma1   -0.8160 0.0559 -14.5960       0
## sma1  -0.8465 0.0518 -16.3322       0
##
## $AIC
## [1] 4.026198
##
```

```
## $AICc
## [1] 4.034293
##
## $BIC
## [1] 3.066613
```

```
x<-arima(housing, order =  c(1,1,1), seasonal = list(order=c(0,1,1), period =12))
Box.test(resid(x), lag = 24, type=c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  resid(x)
## X-squared = 16.103, df = 24, p-value = 0.8843
```

```
Box.test(resid(x), lag = 1, type=c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  resid(x)
## X-squared = 0.0072419, df = 1, p-value = 0.9322
```

```
Box.test(resid(x), lag = 100, type=c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  resid(x)
## X-squared = 79.228, df = 100, p-value = 0.9378
```

Ao reajustar o modelo sem o termo SAR, podemos ver que os todos os critérios de informação apresentam um descréscimo:

Critérios SARIMA(p=1,d=1,q=1,P=1,D=1,Q=1,S=12)

$AIC [1] 4.026867

$AICc [1] 4.035253

$BIC [1] 3.080754

Critérios SARIMA(p=1,d=1,q=1,P=0,D=1,Q=1,S=12)

$AIC [1] 4.026198

$AICc [1] 4.034293

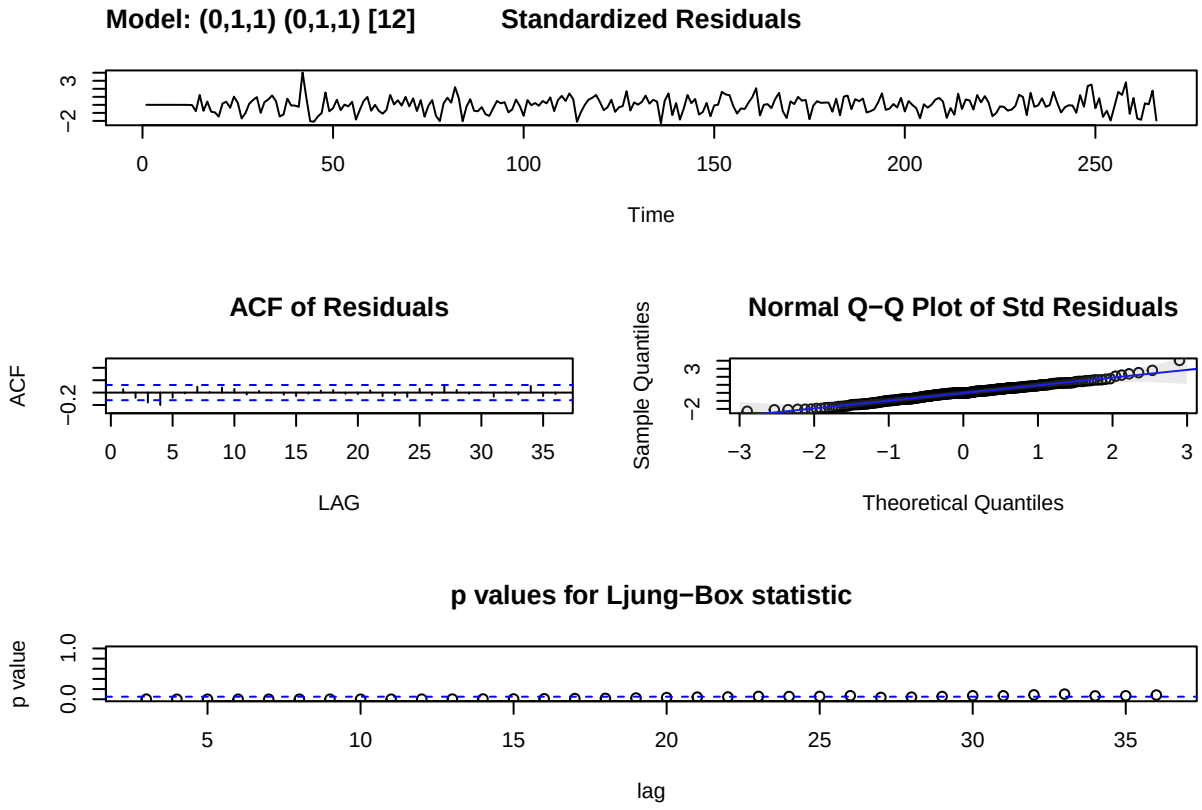
$BIC [1] 3.066613

Além disso, o teste de ljung-box continua tendo evidências insuficientes para rejeitar a hipótese de não haver autocorrelação para os resíduos até a ordem = 24 (p=0.88). A ordem de 24 foi escolhida como rule of thumb do lag a ser testado corresponder ao dobro do período de tempo sazonal da série(12). Se aumentarmos ou diminuirmos a ordem, o p-valor continua exibindo valores maiores que 0.05 até o determinado lag. O p-valor para lag = 1 é 0.9322. O p-valor para lag = 100 é 0.9378. Este parece ser o melhor modelo até então.

Para continuar a análise, removeremos outros termos para checar se existe diminuição dos critérios de informação.

Primeiro, vamos remover o primeiro termo AR:

```
sarima(housing, p=0,d=1,q=1,P=0,D=1,Q=1,S=12,details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##           ma1     sma1
##       -0.3345  -0.8541
## s.e.   0.0825   0.0501
##
## sigma^2 estimated as 21.45:  log likelihood = -754.73,  aic = 1515.45
##
## $degrees_of_freedom
## [1] 264
##
## $ttable
##      Estimate     SE  t.value p.value
## ma1   -0.3345 0.0825  -4.0546   1e-04
## sma1  -0.8541 0.0501 -17.0312   0e+00
##
## $AIC
## [1] 4.080953
```

```
##
## $AICc
## [1] 4.088816
##
## $BIC
## [1] 3.107896
```

$AIC [1] 4.080953

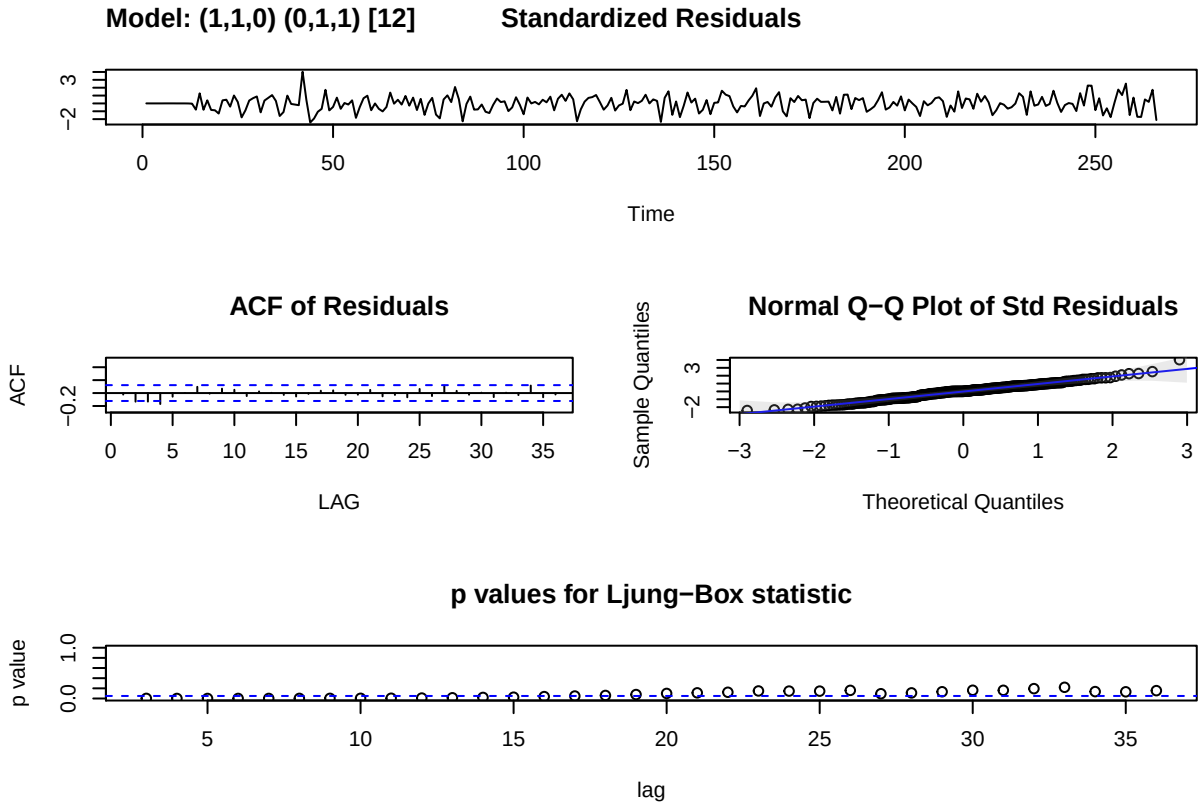
$AICc [1] 4.088816

$BIC [1] 3.107896

O ajustamento teve consideralvemente maiores critérios de informação e piores p-valores para a estatística de ljung-box, sendo assim inferior.

Vamos então adicionar de volta o termo AR e remover o termo q=MA1:

```
sarima(housing, p=1,d=1,q=0,P=0,D=1,Q=1, S=12,details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##           ar1     sma1
```

```
##       -0.2283  -0.8502
## s.e.   0.0620   0.0494
##
## sigma^2 estimated as 21.89:  log likelihood = -757.12,  aic = 1520.25
##
## $degrees_of_freedom
## [1] 264
##
## $ttable
##      Estimate     SE  t.value p.value
## ar1   -0.2283 0.0620  -3.6811   3e-04
## sma1  -0.8502 0.0494 -17.2015   0e+00
##
## $AIC
## [1] 4.101287
##
## $AICc
## [1] 4.10915
##
## $BIC
## [1] 3.128231
```

$AIC [1] 4.101287

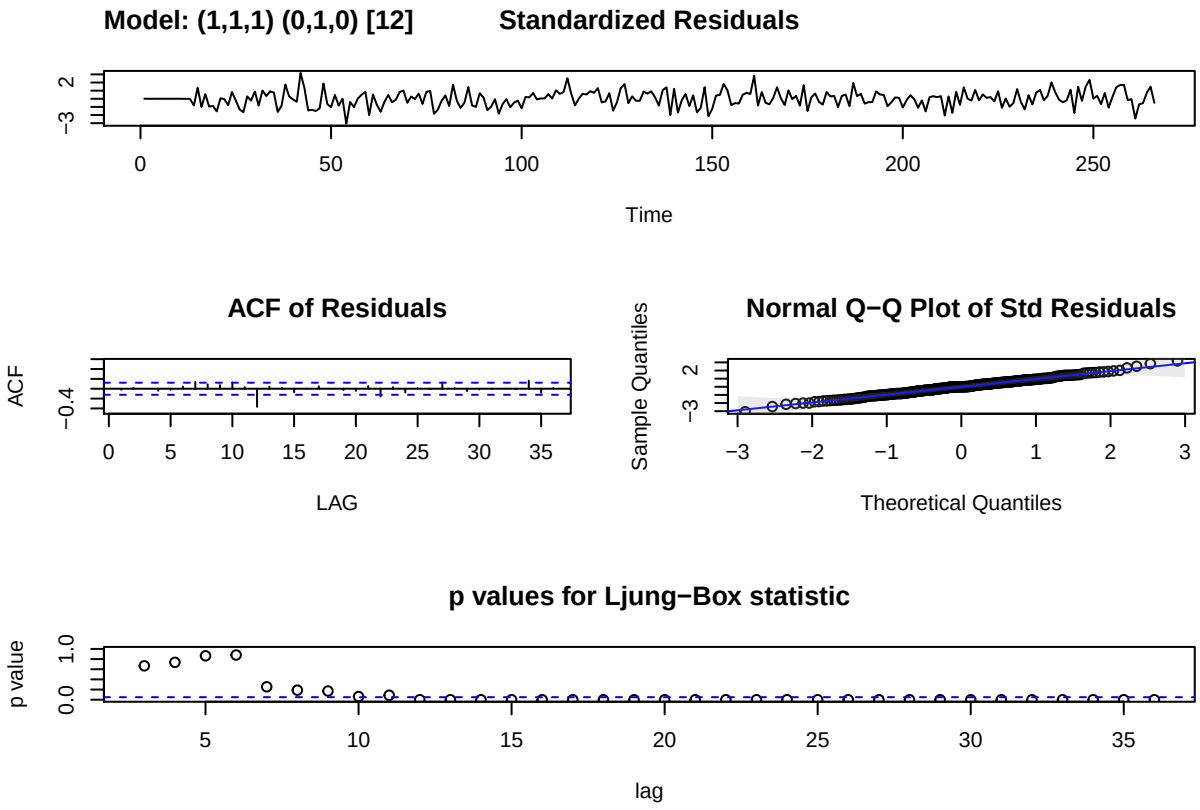
$AICc [1] 4.10915

$BIC [1] 3.128231

O novo modelo também se mostrou inferior, com maiores critérios de informação e piores valores para a estatística de Ljung-box para os resíduos.

Por último, realizaremos a última transformação e removeremos o termo Q=SMA(1):

```
sarima(housing, p=1,d=1,q=1,P=0,D=1,Q=0, S=12,details = F)
```



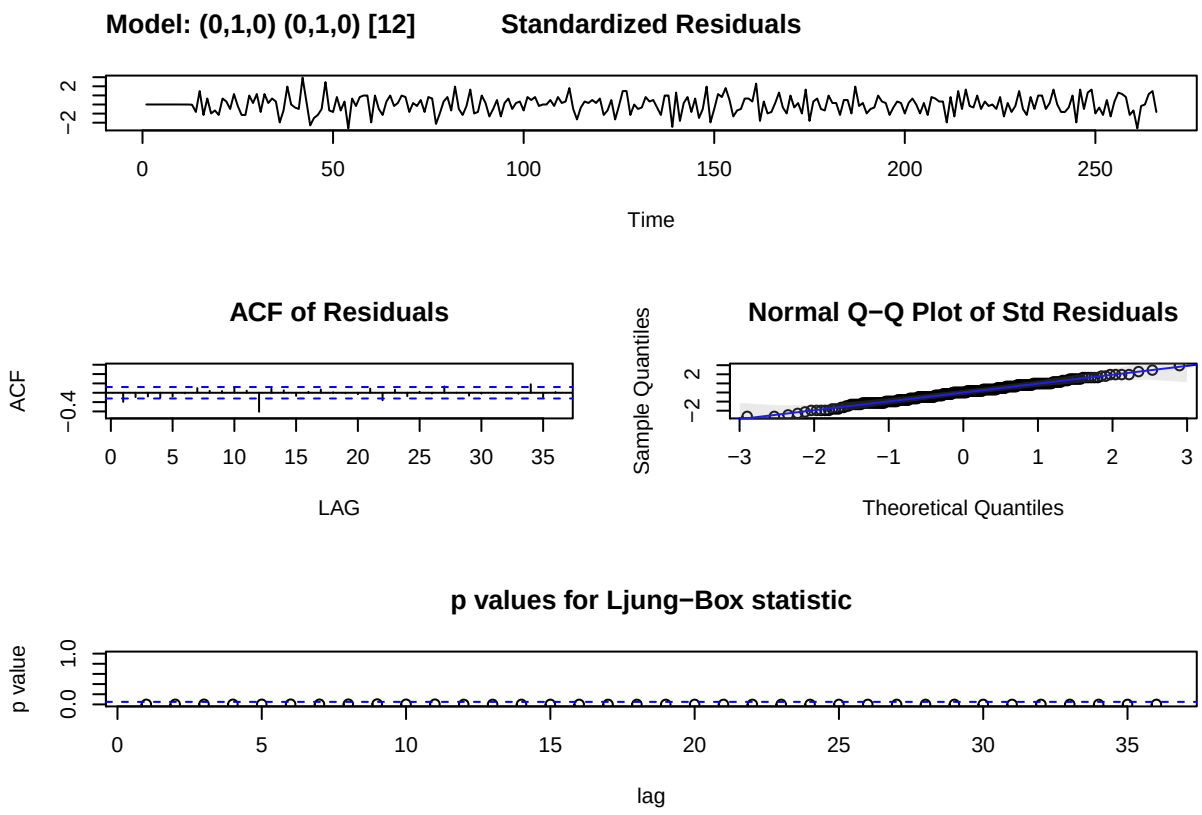

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ma1
##       0.6537  -0.9705
## s.e.  0.0681   0.0352
##
## sigma^2 estimated as 31.59:  log likelihood = -796.47,  aic = 1598.94
##
## $degrees_of_freedom
## [1] 264
##
## $ttable
##     Estimate     SE  t.value p.value
## ar1   0.6537 0.0681   9.5994       0
## ma1  -0.9705 0.0352 -27.5669       0
##
## $AIC
## [1] 4.46785
##
## $AICc
```

```
## [1] 4.475713
##
## $BIC
## [1] 3.494793
```

O modelo sem SMA1 também é inferior, com critérios de informação muito mais elevados.

Por último, vamos remover ambos os termos AR1 e MA1:

```
sarima(housing, p=0,d=1,q=0,P=0,D=1,Q=0, S=12,details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace = trc,
##     REPORT = 1, reltol = tol))
##
##
## sigma^2 estimated as 37.14:  log likelihood = -816.26,  aic = 1634.52
##
## $degrees_of_freedom
## [1] 266
##
## $ttable
##      Estimate p.value
##
## $AIC
```

```
## [1] 4.614756
##
## $AICc
## [1] 4.622332
##
## $BIC
## [1] 3.614756
```

Os critérios de informação aumentaram consideravelmente após a remoção de ambos os termos.

Logo, o modelo final para a previsão de *housing* ficou **sarima(housing, p=1, d=1, q=1, P=0, D=1, Q=1, S=12)**.

O modelo final possui critérios de informação AIC, AICc e BIC menores que todos os outros modelos, assim como maior p-valor para o teste de Ljung-Box. Além disso, a análise de resíduos mostra um QQ Plot bem comportado, um plot de resíduos semelhante a de um White-noise estacionário e uma ACF dos resíduos mostrando ausência de autocorrelação.

auto.arima()

Agora que já possuímos o melhor modelo ajustado, podemos realizar outras análises antes da previsão.

Primeiramente, iremos comparar o melhor modelo identificado com o melhor modelo identificado pela função auto.arima() da library forecast.

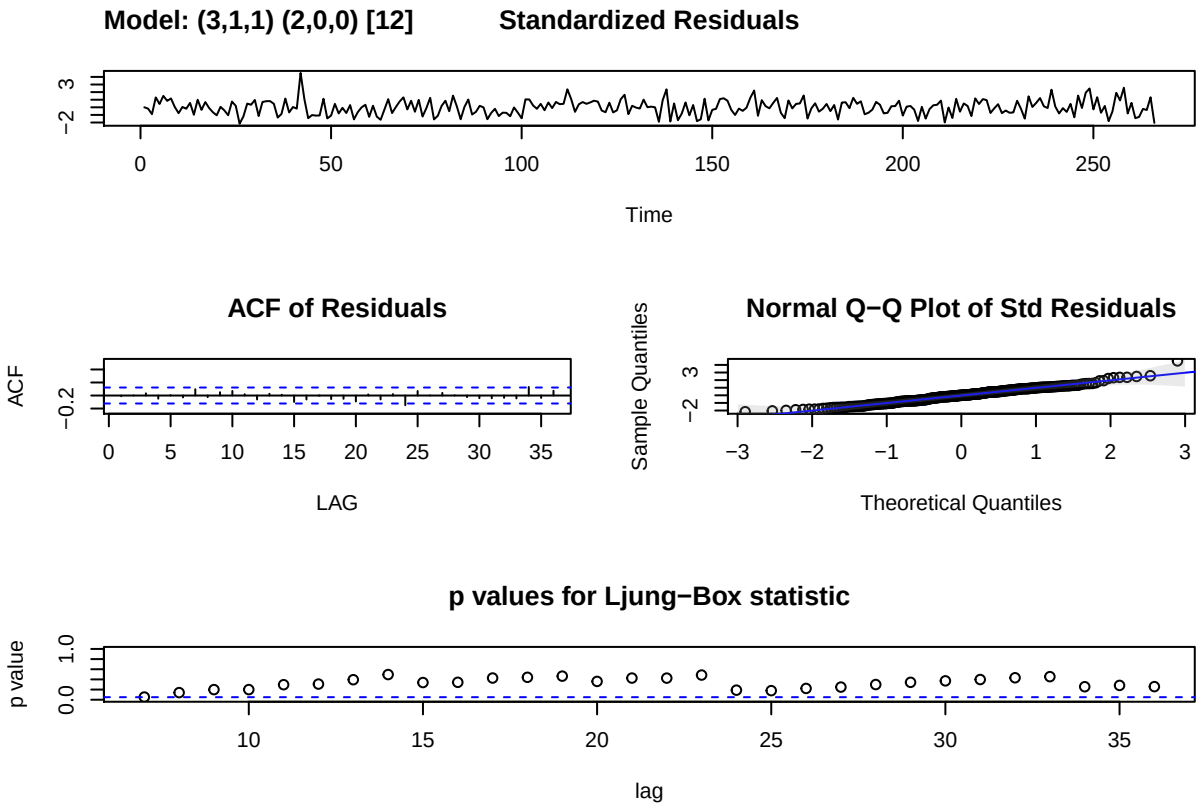
A função auto.arima retornou o modelo sarima(housing, p=3, d=1, q= 1, P=2, D=0,Q=0). Antes a função não encontrava um modelo sazonal (arima 4,1,2) pois ela requer que a sazonalidade ja esteja implicita na série temporal quando define a série (pensei que ele encontrava automaticamente). Aí setei a frequência para 12 (1 ano, mensal) e ele encontrou o modelo sazonal ao invés do modelo não sazonal.

A análise de resíduos da função sarima nos mostrou p-valores do teste de ljung-box razoáveis com p>0.05 para os testes até o lag=35 e incapacidade de rejeitar a hipótese de ausência de autocorrelação. Porém, houve um aumento significativo nos Critérios de Informação do modelo em relação a nosso melhor modelo ajustado. Portanto, podemos afirmar que o modelo sarima(housing, p=1,d=1,q=1,P=0,D=1,Q=1,S=12) possui melhor ajustamento do que o modelo gerado pela função auto.arima().

```
housing2 = ts(housing, frequency = 12)
auto.arima(housing2, seasonal = T, trace = F, test = c("adf"))
```

```
## Series: housing2
## ARIMA(3,1,1)(2,0,0)[12]
##
## Coefficients:
##          ar1     ar2      ar3      ma1    sar1    sar2
##       0.5264  0.0539  -0.1678  -0.8178  0.4448  0.3229
## s.e.  0.0961  0.0722   0.0699   0.0760  0.0598  0.0635
##
## sigma^2 estimated as 24.77:  log likelihood=-803.2
## AIC=1620.4   AICc=1620.84   BIC=1645.46
```

```
sarima(housing, 3,1,1,2,0,0,12, details = F)
```



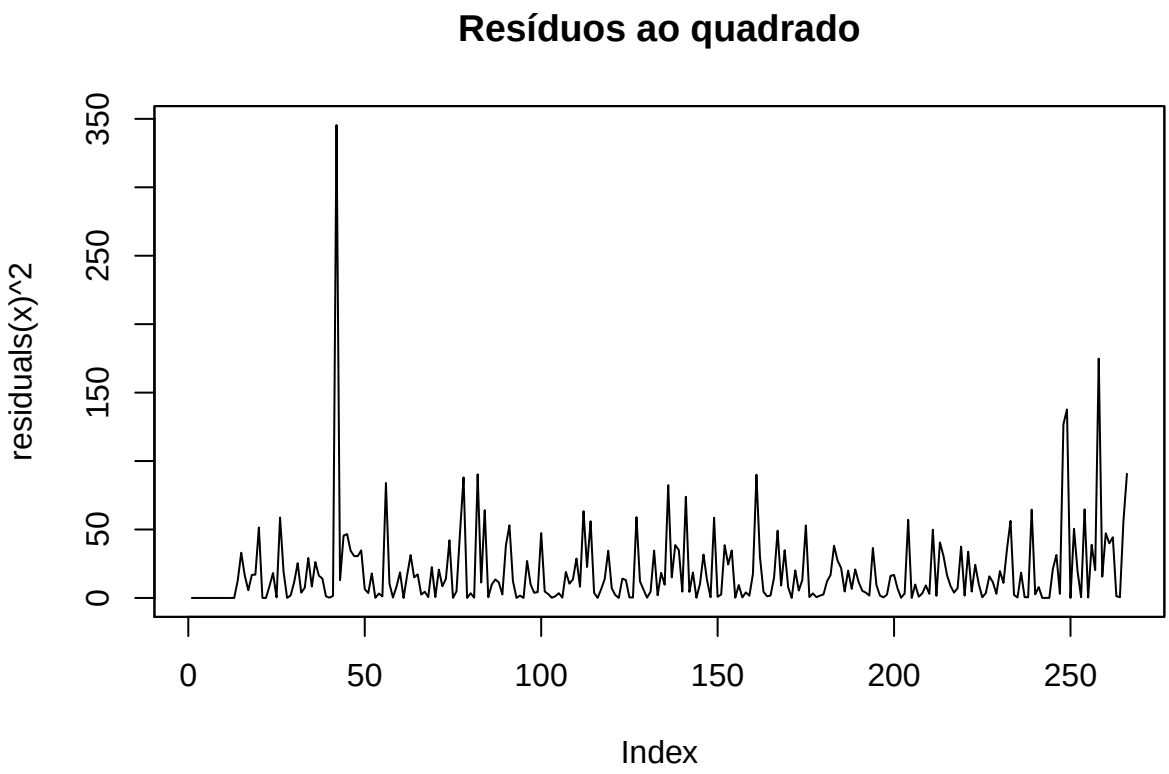

```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = constant, optim.control = list(trace = trc, REPORT = 1,
##     reltol = tol))
##
## Coefficients:
##          ar1     ar2      ar3      ma1    sar1    sar2  constant
##       0.5388  0.0564  -0.1654  -0.8326  0.4429  0.3185    0.2992
## s.e.  0.0981  0.0727   0.0711   0.0785  0.0600  0.0641    0.3179
##
## sigma^2 estimated as 24.15:  log likelihood = -802.78,  aic = 1621.57
##
## $degrees_of_freedom
## [1] 259
##
## $ttable
##          Estimate     SE  t.value p.value
## ar1        0.5388 0.0981   5.4939  0.0000
## ar2        0.0564 0.0727   0.7757  0.4386
## ar3       -0.1654 0.0711  -2.3259  0.0208
## ma1       -0.8326 0.0785 -10.6119  0.0000
## sar1       0.4429 0.0600   7.3829  0.0000
## sar2       0.3185 0.0641   4.9693  0.0000
## constant   0.2992 0.3179   0.9412  0.3475
```

```
##
## $AIC
## [1] 4.237104
##
## $AICc
## [1] 4.24673
##
## $BIC
## [1] 3.331407
```

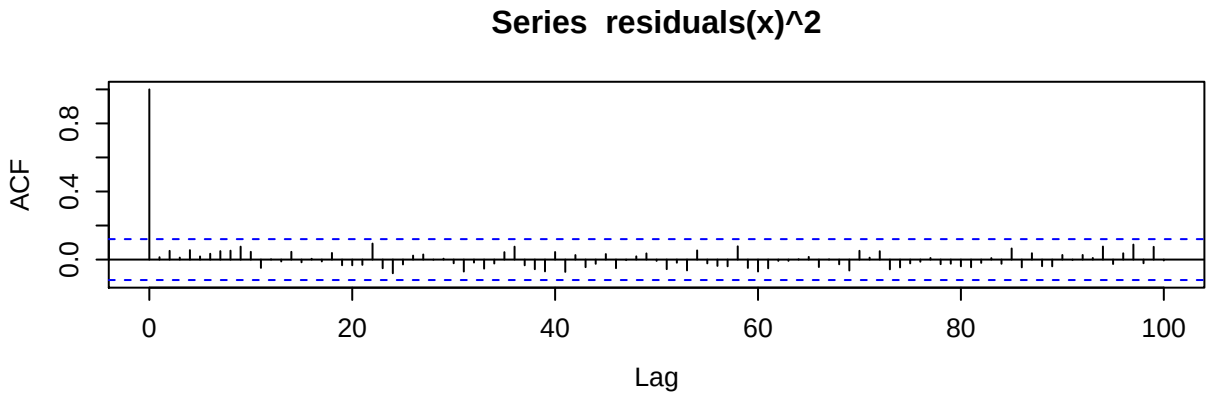
ARCH Test

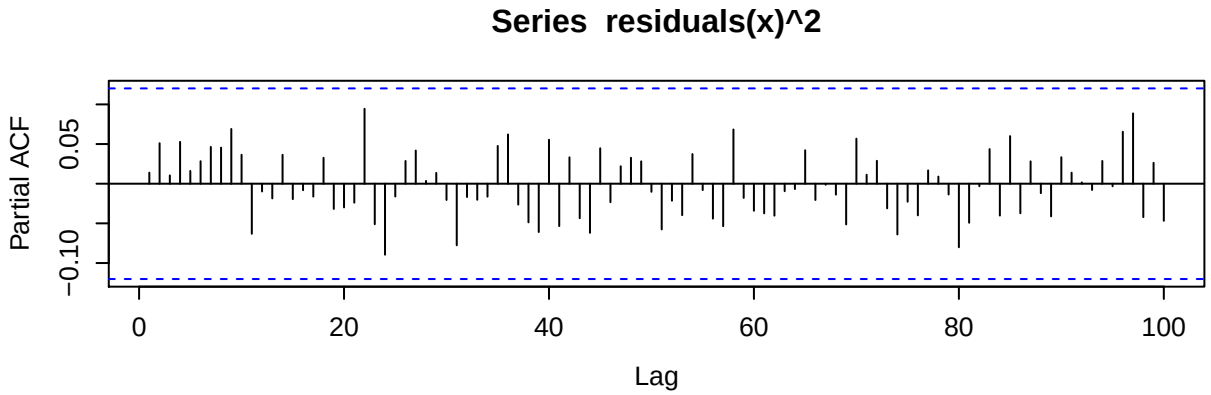
Iremos testar a necessidade de modelar a variância condicional da série pelo modelo GARCH. Primeiramente, realizaremos o plot dos resíduos ao quadrado da série e de sua função de autocorrelação e autocorrelação parcial. Podemos ver a partir das funções de autocorrelação que o modelo GARCH teria p=0 e q=0, uma vez que nenhum valor do ACF/PACF é significativo. Portanto, há indícios que a modelagem da heterocedasticidade condicional pode ser descartada. Para nos certificarmos, realizaremos o Teste de Engle para a variância condicional.

```
x<-arima(housing, order =  c(1,1,1), seasonal = list(order=c(0,1,1), period =12))
plot.zoo(residuals(x)^2, main = 'Resíduos ao quadrado')
```




```
par(mfrow=c(2,1), mex = 0.8, cex = 0.8)
acf(residuals(x)^2, lag.max = 100)
pacf(residuals(x)^2, lag.max = 100)
```





O teste de Engle testa a hipótese nula da sequência dos quadrados dos resíduos do modelo ser um white-noise. Para a realização do teste foi aproveitado uma função de arch test de uma library que não está mais disponível para a última versão do R (Financial Timeseries - FinTs), uma vez que o teste que foi utilizado antes possuía valores não desejados para os lags.

O teste de engle para a variância condicional retorna um p-value de 0.818 para o lag = 1, levando-nos a incapacidade de rejeitar a hipótese nula dos resíduos ao quadrado serem um white-noise. O resultado do teste está de acordo com o que observamos nas funções de autocorrelação e autocorrelação parcial dos resíduos ao quadrado.

```
library(aTSA)
```

```
##
## Attaching package: 'aTSA'
```

```
## The following objects are masked from 'package:tseries':
##
##     adf.test, kpss.test, pp.test
```

```
## The following object is masked from 'package:forecast':
##
##     forecast
```

```
## The following object is masked from 'package:graphics':
##
##     identify
```

```
ArchTest <- function (x, lags=1, demean = FALSE)
{
```

```
# Capture name of x for documentation in the output
  xName <- deparse(substitute(x))
#
  x <- residuals(x)
  if(demean) x <- scale(x, center = TRUE, scale = FALSE)
#
  lags <- lags + 1
  mat <- embed(x^2, lags)
  arch.lm <- summary(lm(mat[, 1] ~ mat[, -1]))
  STATISTIC <- arch.lm$r.squared * length(resid(arch.lm))
  names(STATISTIC) <- "Chi-squared"
  PARAMETER <- lags - 1
  names(PARAMETER) <- "df"
  PVAL <- 1 - pchisq(STATISTIC, df = PARAMETER)
  METHOD <- "ARCH LM-test;  Null hypothesis:  no ARCH effects"
  result <- list(statistic = STATISTIC, parameter = PARAMETER,
                 p.value = PVAL, method = METHOD, data.name =
                 xName)
  class(result) <- "htest"
  return(result)
}

ArchTest(x)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  x
## Chi-squared = 0.052982, df = 1, p-value = 0.818
```

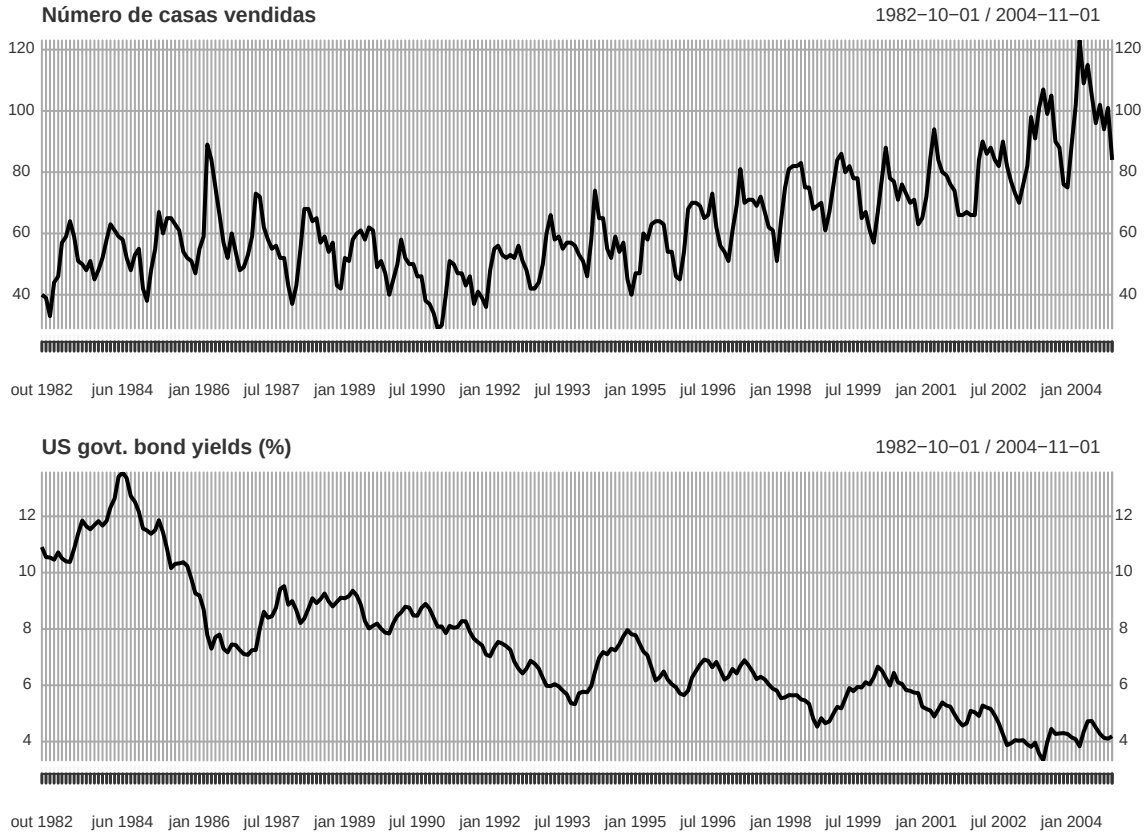
Adição de um Regressor

Por fim, vamos adicionar um regressor no modelo SARIMA ajustado, afim de testar a causalidade entre casas vendidas nos EUA e a Taxa de juros de longo prazo (10 anos) de títulos de dívida do Governo dos Estados Unidos. Esta série foi escolhida pois históricamente a taxa de hipotecagem é definiada em relação a Taxa de Juros de longo prazo dos títulos do governo.

A especificação do modelo fica então SARIMAX(housing, p=1,d=1,q=1, P=0,D=1,Q=1,S=12, xreg= irl2). Ou seja, a venda de casas agora passa a ser explicada pela relação entre beta*A taxa de juros de longo prazo dos títulos mais um componente estocástico de erros da forma SARIMA.

Abaixo, podemos ver uma comparação de ambas as séries temporais:

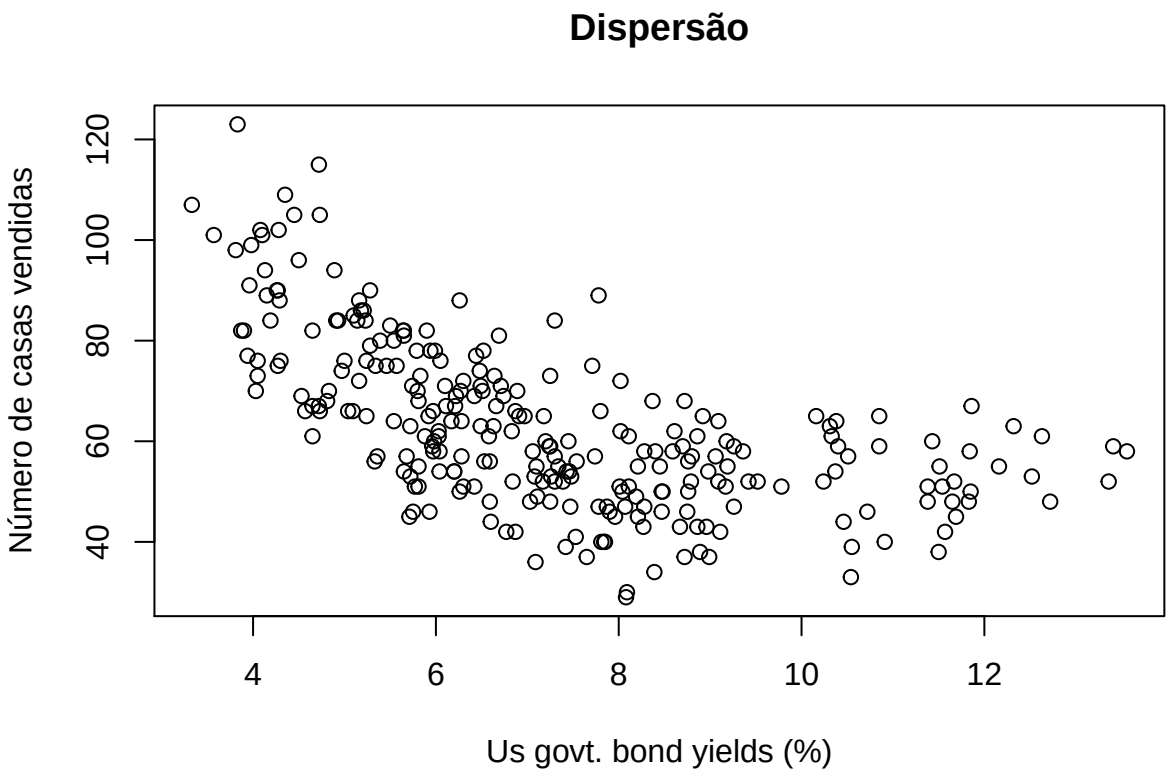
```
irl2<- getSymbols("IRLTLT01USM156N", src = 'FRED', auto.assign= F)
irl2 <- irl2["1982-10-01/2004-11-01"]

par(mfrow = c(2,1))
plot.xts(housing, main = "Número de casas vendidas")
plot.xts(irl2, main = "US govt. bond yields (%)")
```




```
#cor(housing, irl2)
```

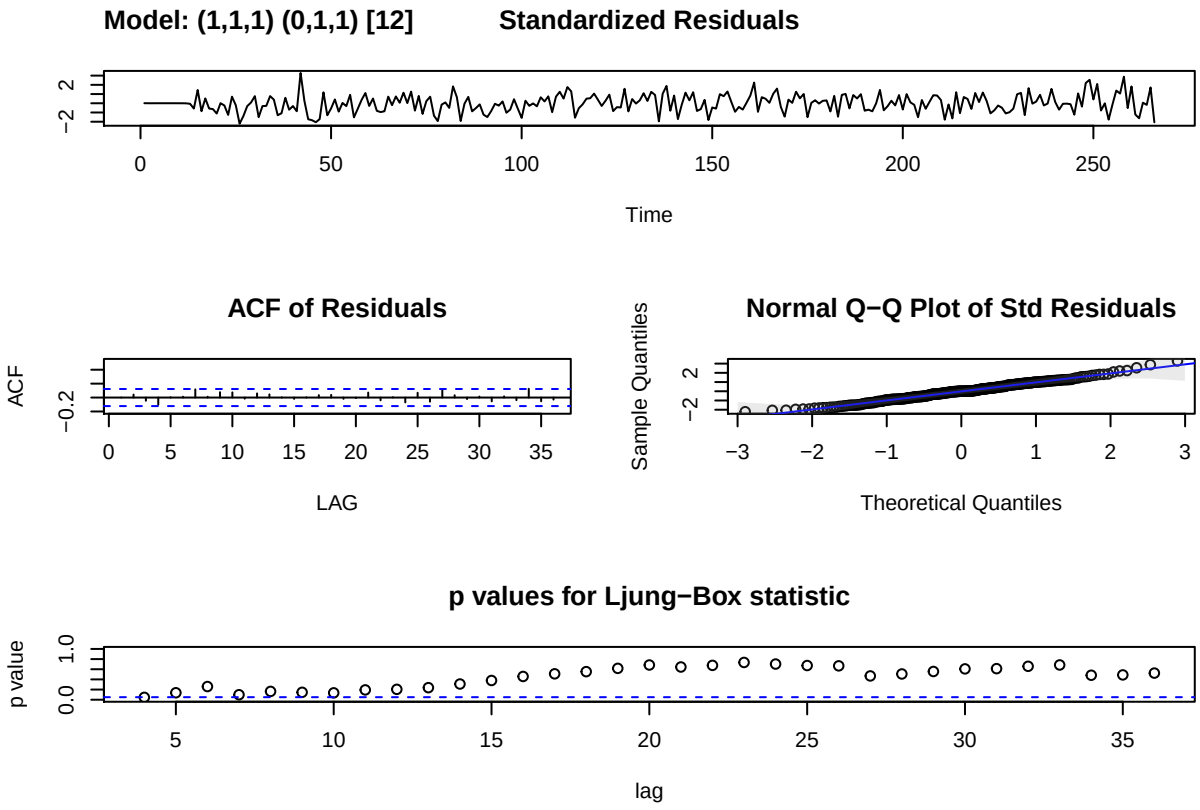
E também a dispersão entre as séries:

```
plot.default(x= irl2, y= housing, main =
"Dispersão", xlab = "Us govt. bond yields (%)", ylab= "Número de casas vendidas")
```



Estimando o modelo:

```
sarima(housing, 1,1,1,0,1,1,12, xreg= irl2, details = F)
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = xreg, optim.control = list(trace = trc, REPORT = 1,
##     reltol = tol))
##
## Coefficients:
##         ar1      ma1     sma1  IRLTLT01USM156N
##       0.425  -0.7951  -0.8350          -2.8860
## s.e.  0.090   0.0580   0.0518           0.7178
##
## sigma^2 estimated as 18.98:  log likelihood = -738.76,  aic = 1487.53
##
## $degrees_of_freedom
## [1] 262
##
## $ttable
##                 Estimate     SE  t.value p.value
## ar1               0.4250 0.0900   4.7199   0e+00
## ma1              -0.7951 0.0580 -13.7030   0e+00
## sma1             -0.8350 0.0518 -16.1275   0e+00
## IRLTLT01USM156N  -2.8860 0.7178  -4.0205   1e-04
##
## $AIC
## [1] 3.973664
```

```
##
## $AICc
## [1] 3.98205
##
## $BIC
## [1] 3.027551
```

Após incluirmos o regressor na análise, podemos concluir que a adição do regressor minizou todos os três critérios de informação.

Para continuar a análise, vamos realizar o teste de Ljung-Box para os resíduos do novo modelo SARIMAX.

```
final<-arima(housing, order =  c(1,1,1), seasonal = list(order=c(0,1,1), period =12), xreg = irl2)
Box.test(resid(final), lag = 24, type = c("Lj"))
```

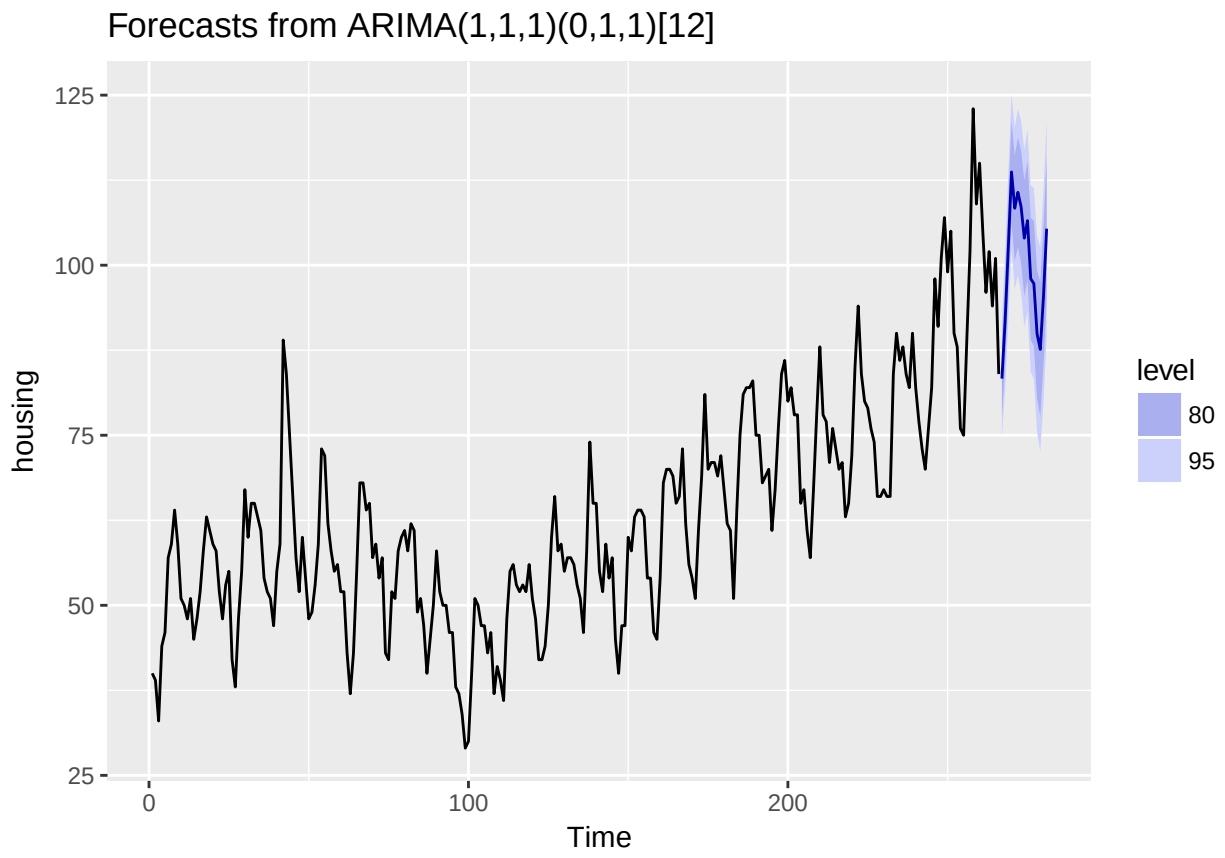
```
##
##  Box-Ljung test
##
## data:  resid(final)
## X-squared = 17.143, df = 24, p-value = 0.8425
```

Com p-valor de 0.84 para o teste de ljung-box, continuamos não tendo evidências para rejeitar a hipótese nula de ausência de autocorrelação dos resíduos até a ordem 24. Houve uma diminuição em todos os Critérios de Informação e podemos afirmar que o modelo com regressor aumentou a qualidade de nosso ajuste.

Previsão

Realizaremos a previsão para 15 meses a frente. Para isso, carregaremos os 15 meses seguintes de nosso regressor (taxa de retornos do tesouro americano), e plotaremos a previsão com 80% e 95% de confiança.

```
irlahead<- getSymbols("IRLTLT01USM156N", src = 'FRED', auto.assign= F)
irlahead <- irlahead['2004-12-01/2006-02-01']


fitted<-arima(housing, order =  c(1,1,1), seasonal = list(order=c(0,1,1), period =12), xreg = irl2)

predito <- forecast::forecast(fitted,xreg=irlahead, level = c(80, 95))

autoplot(predito, showgap = F)
```



Veremos agora a precisão da previsão comparando o que foi predito para 2004-12-01/2006-02-01 com os valores observados no período.

```
accuracy(predito, housingtest)
```

```
##                       ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -0.03030181 4.249270 3.345952 -0.5200454 5.507364 0.6131931
## Test set      2.64406684 8.175285 6.461890  1.7583969 6.103723 1.1842329
##                    ACF1
## Training set 0.001311006
## Test set              NA
```

A partir dos erros de previsão, podemos ver que a Raíz do Erro Quadrado Médio (RMSE) é aproximadamente 8.1752, com a previsão plotada graficamente.

Por fim, compararemos de uma forma visual os valores do test set (os valores reais de HSN1FNSA de 2004-12-01 a 2005-09-01) e os valores médios previstos com o nosso modelo:

```
preditomedio<-predito$mean
compar<-data.frame(housingtest, preditomedio)
compar
```

```
##            HSN1FNSA preditomedio
## 2004-12-01       83     83.32788
## 2005-01-01       92     91.80615
## 2005-02-01      109    102.69472
## 2005-03-01      127    113.70872
## 2005-04-01      116    108.36963
## 2005-05-01      120    110.71607
```

```
## 2005-06-01      115    108.58731
## 2005-07-01      117    103.97610
## 2005-08-01      110    106.54673
## 2005-09-01       99     98.00368
## 2005-10-01      105     97.29620
## 2005-11-01       86     89.89295
## 2005-12-01       87     87.62771
## 2006-01-01       89     95.40943
## 2006-02-01       88    105.37570
```

Através da análise entre o valor observado e o valor predito, podemos ver que nosso modelo prevê com consistência diversas observações.

U de Theil

Por fim, vamos calcular para nosso modelo SARIMAX o indicator de U de Theil para as 15 observações fora da amostra. Como primeiro argumento, utilizaremos os valores observados das 15 observações. Como segundo argumento, utilizaremos o valor predito pelo modelo, para estas 15 observações.

```
library(DescTools)
```

```
## Warning: package 'DescTools' was built under R version 3.4.3
```

```
##
## Attaching package: 'DescTools'
```

```
## The following object is masked from 'package:forecast':
##
##     BoxCox
```

```
TheilU(as.ts(housingtest, start = 267, end= 281), preditomedio, type = 2, na.rm = F)
```

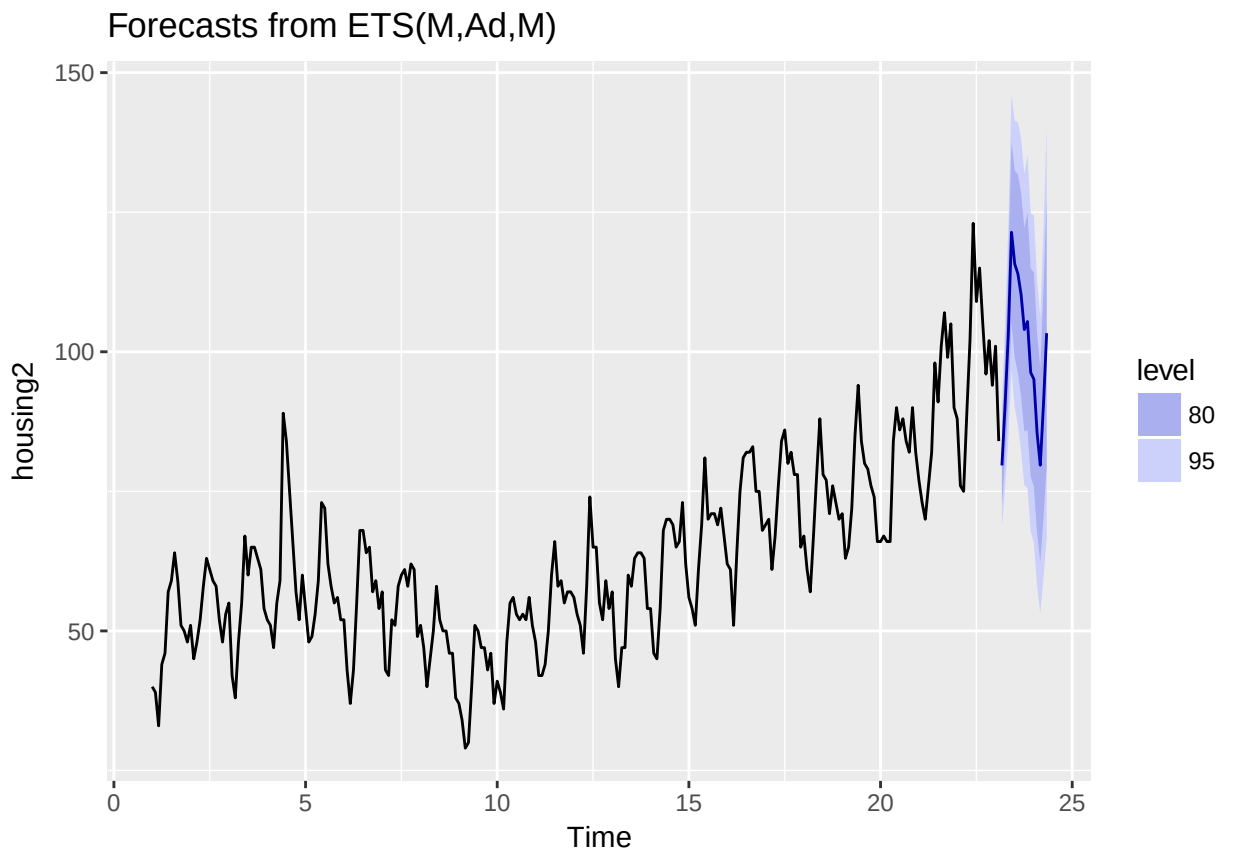
```
## [1] 0.07874233
```

A função U de Theil nos retornou o valor de 0.0787, o que indica que utilizar nosso modelo ajustado é significativamente melhor que apenas assumir que o valor de hoje será o valor passado.

ARIMA e Suavização Exponencial

Iremos, por fim, comparar a previsão do modelo SARIMAX com outro famoso modelo de previsão, o modelo Suavização Exponencial (ETS).

```
modelo <- ets(housing2, model = "ZZZ")
prevets <- forecast::forecast.ets(modelo, h=15)
autoplot(prevets)
```



O modelo automaticamente detecta os parâmetros Error, Trend e Seasonality como Multiplicativo, Aditivo e Multiplicativo.

```
accuracy(prevets, housingtest)
```

```
##                       ME    RMSE      MAE        MPE     MAPE      MASE
## Training set 0.1786368 4.26425 3.426766 -0.1163345 5.609504 0.6280034
## Test set     3.2271824 6.95083 5.483817  2.8022491 5.363898 1.0049873
##                    ACF1
## Training set 0.07706078
## Test set             NA
```

O modelo de suavização exponencial possui RMSE de 6.95 e apresenta uma previsão aparentemente melhor do que o modelo selecionado a partir de Box and Jenkins (RMSE = 8.1752) e consequentemente do modelo selecionado pela função auto.arima (que possuí todos critérios de informação maiores que o modelo por box and jenkins). A previsão agora ficou muito melhor pois ocorreu o mesmo erro da função auto.arima em que a sazonalidade não estava explicitada na série. Depois que explicitei uma sazonalidade de 12, a função selecionou um modelo adequado, ao invés de uma previsão constante.

Conclusão

A partir do apresentado no trabalho, podemos afirmar que a o processo de Box and Jenkings para modelar séries temporais através do modelo ARIMA pode ser aplicado razoavelmente bem para a série de Venda mensal de casas nos Estados Unidos. O modelo final apresenta um termo AR, um termo MA, um termo MA sazonal, uma diferença não sazonal e uma diferença sazonal de lag 12, além do regressor US Treasure Bond Yields. Os testes de raiz unitária rejeitam a presença da raiz unitária e atestam que a série final é causal. A modelagem da heterocedasticidade condicional foi descartada com base no teste LM. Ademais, o modelo final

conta com Critérios de Informação de menor magnitude que os Critérios da função auto.arima. Nosso modelo apresenta uma melhor qualidade de previsão em relação ao modelo de Suavização Exponencial e um valor satisfatório para a estatística U de Theil.

Bibliografia:

Libraries utilizadas: astsa CADFtest forecast ggplot2 quantmod tseries TTR urca xts DescTools