Centro de Estatística Aplicada

Gustavo Kanno[1]
Rodrigo Marcel Araujo[2]
Victor Ribeiro Baião Decanini[3]

Julho de 2021

Sumário

[1] Número USP: 9795810
[2] Número USP: 9299208
[3] Número USP: 9790502

```r
library(randtests)
```

```
##
## Attaching package: 'randtests'

## The following object is masked from 'package:tseries':
##
##     runs.test
```

```r
library(zoo)
library(TSA)
```

```
## Registered S3 methods overwritten by 'TSA':
##   method       from
##   fitted.Arima forecast
##   plot.Arima   forecast

##
## Attaching package: 'TSA'

## The following object is masked from 'package:GeneCycle':
##
##     periodogram

## The following object is masked from 'package:readr':
##
##     spec

## The following objects are masked from 'package:stats':
##
##     acf, arima

## The following object is masked from 'package:utils':
##
##     tar
```

```r
library(gridExtra)
library(FitAR)
```

```
## Loading required package: lattice

##
## Attaching package: 'lattice'

## The following object is masked from 'package:faraway':
##
##     melanoma
```

```
## Loading required package: leaps

## Loading required package: ltsa

## Loading required package: bestglm

##
## Attaching package: 'FitAR'

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

## The following object is masked from 'package:car':
##
##     Boot
```

```r
library(glmnet)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loaded glmnet 4.1-1
```

```r
library(astsa)
```

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

## The following objects are masked from 'package:fma':
##
##     chicken, sales

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

## The following object is masked from 'package:fpp2':
##
##     oil
```

```
## The following object is masked from 'package:faraway':
##
##     star
```

```r
library(lmtest)
```

Análise das séries temporais mensais

Análise Descritiva

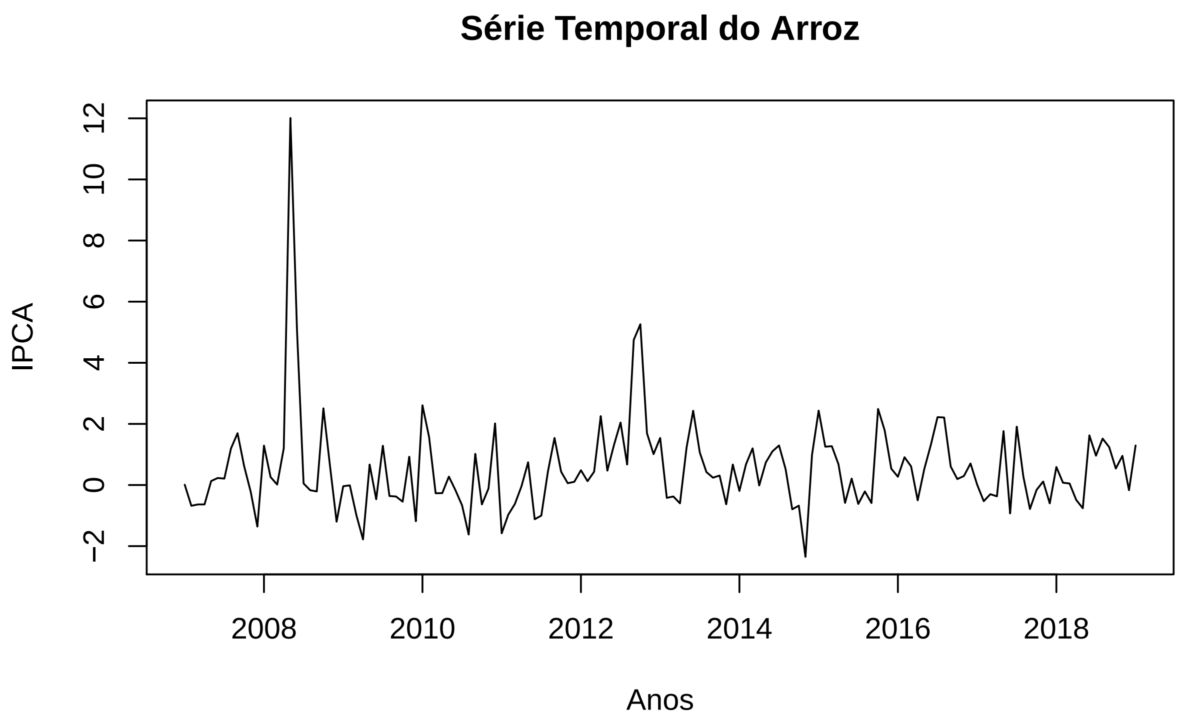
```r
data$Data <- as.Date(data$Data)
head(data)
```

```
## # A tibble: 6 x 24
##   Data        Arroz `Avicultura de ~ `Avicultura de ~ Banana Batata
##   <date>      <dbl>            <dbl>            <dbl>  <dbl>  <dbl>
## 1 2007-01-01  0.01             0.295             3.43  -2.86   0.75
## 2 2007-02-01 -0.68             1.71              2.82  -1.62  -3.83
## 3 2007-03-01 -0.635            2.26             10.1    1.05   7.61
## 4 2007-04-01 -0.635           -0.56              1.31  -2.65  36.4
## 5 2007-05-01  0.13            -0.13             -1.11  -1.46  11.6
## 6 2007-06-01  0.230            0.27              4.93  -1.07  -5.17
## # ... with 18 more variables: Bovinocultura <dbl>, `Cacau e produtos` <dbl>,
## #   Café <dbl>, Cebola <dbl>, `Complexo soja` <dbl>, `Complexo
## #   sucroalc.` <dbl>, Feijão <dbl>, Frutas <dbl>, Hortícolas <dbl>,
## #   Indefinido <dbl>, `Laranja e citros` <dbl>, Lácteos <dbl>, Mandioca <dbl>,
## #   Milho <dbl>, Pescado <dbl>, Suinocultura <dbl>, Tomate <dbl>, Trigo <dbl>
```

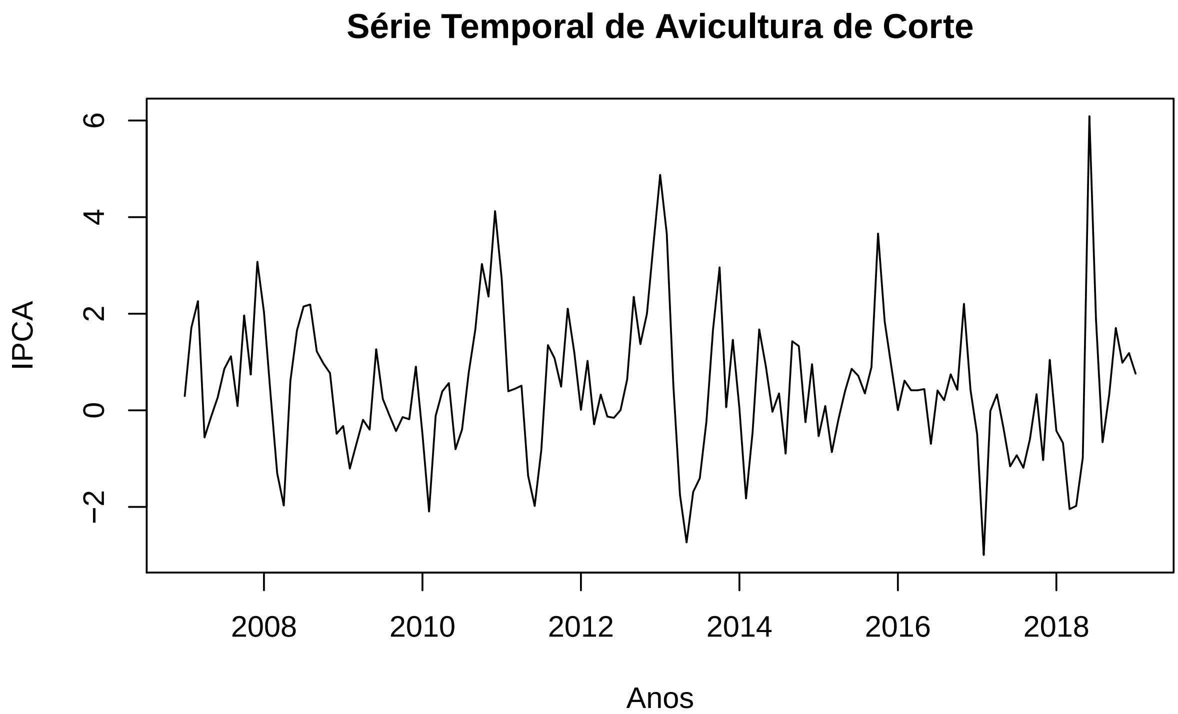
```r
zt2 <- ts(data[,2], frequency = 12, start = 2007, end = 2019)
zt3 <- ts(data[,3], frequency = 12, start = 2007, end = 2019)
zt4 <- ts(data[,4], frequency = 12, start = 2007, end = 2019)
zt5 <- ts(data[,5], frequency = 12, start = 2007, end = 2019)
zt6 <- ts(data[,6], frequency = 12, start = 2007, end = 2019)
zt7 <- ts(data[,7], frequency = 12, start = 2007, end = 2019)
zt8 <- ts(data[,8], frequency = 12, start = 2007, end = 2019)
zt9 <- ts(data[,9], frequency = 12, start = 2007, end = 2019)
zt10 <- ts(data[,10], frequency = 12, start = 2007, end = 2019)
zt11 <- ts(data[,11], frequency = 12, start = 2007, end = 2019)

zt12 <- ts(data[,12], frequency = 12, start = 2007, end = 2019)
zt13 <- ts(data[,13], frequency = 12, start = 2007, end = 2019)
zt14 <- ts(data[,14], frequency = 12, start = 2007, end = 2019)
zt15 <- ts(data[,15], frequency = 12, start = 2007, end = 2019)
zt16 <- ts(data[,16], frequency = 12, start = 2007, end = 2019)
zt17 <- ts(data[,17], frequency = 12, start = 2007, end = 2019)
zt18 <- ts(data[,18], frequency = 12, start = 2007, end = 2019)
zt19 <- ts(data[,19], frequency = 12, start = 2007, end = 2019)
zt20 <- ts(data[,20], frequency = 12, start = 2007, end = 2019)
zt21 <- ts(data[,21], frequency = 12, start = 2007, end = 2019)
```

```r
zt22 <- ts(data[,22], frequency = 12, start = 2007, end = 2019)
zt23 <- ts(data[,23], frequency = 12, start = 2007, end = 2019)
zt24 <- ts(data[,24], frequency = 12, start = 2007, end = 2019)
```

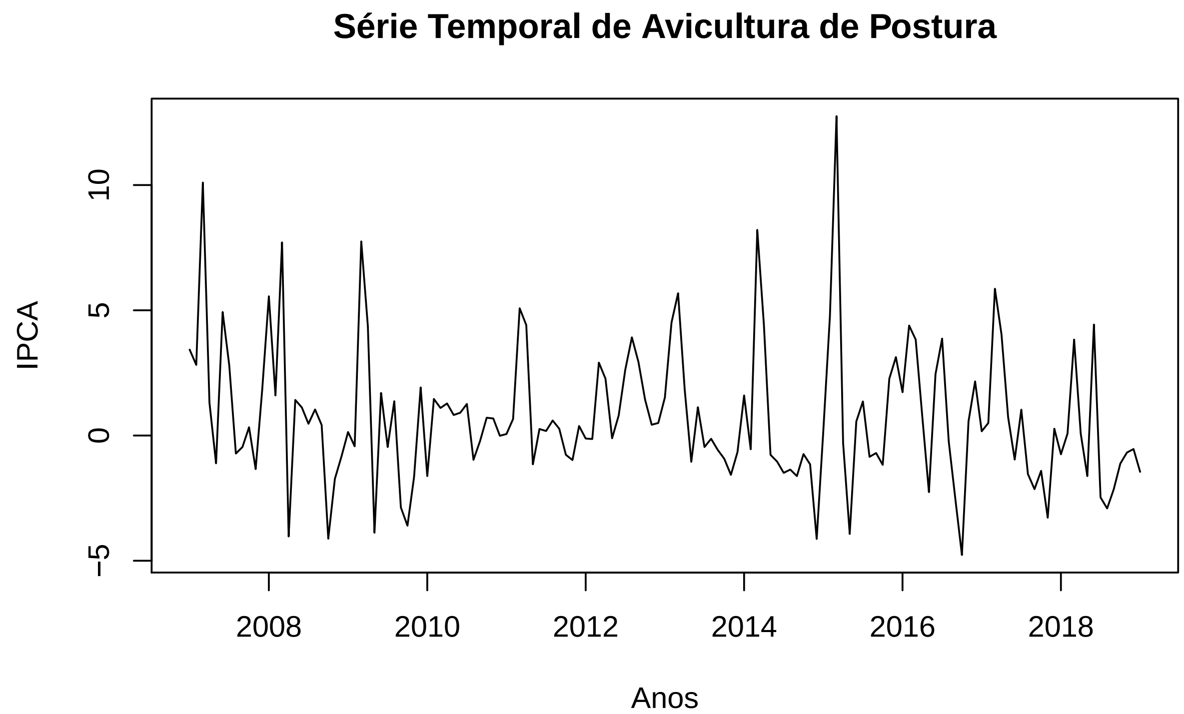
```r
plot(zt2,main="Série Temporal do Arroz", xlab= "Anos", ylab="IPCA")
```



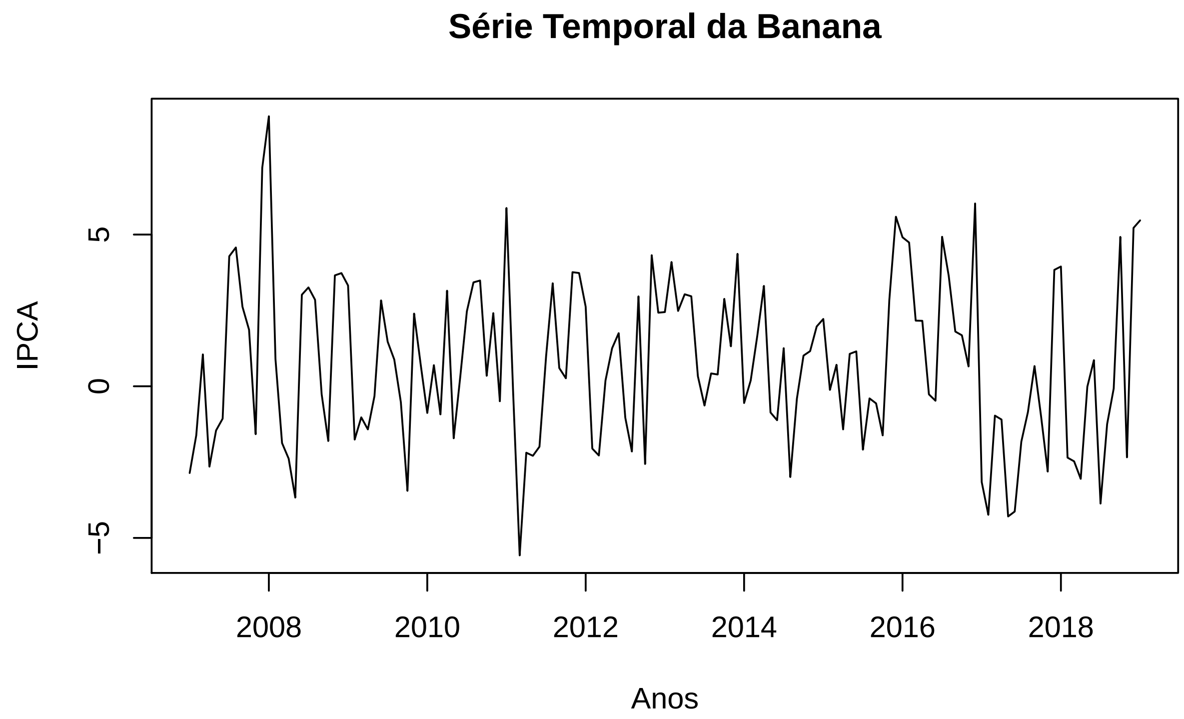

```r
#par(mfrow = c(2, 2))
plot(zt3,main="Série Temporal de Avicultura de Corte", xlab= "Anos", ylab="IPCA")
```



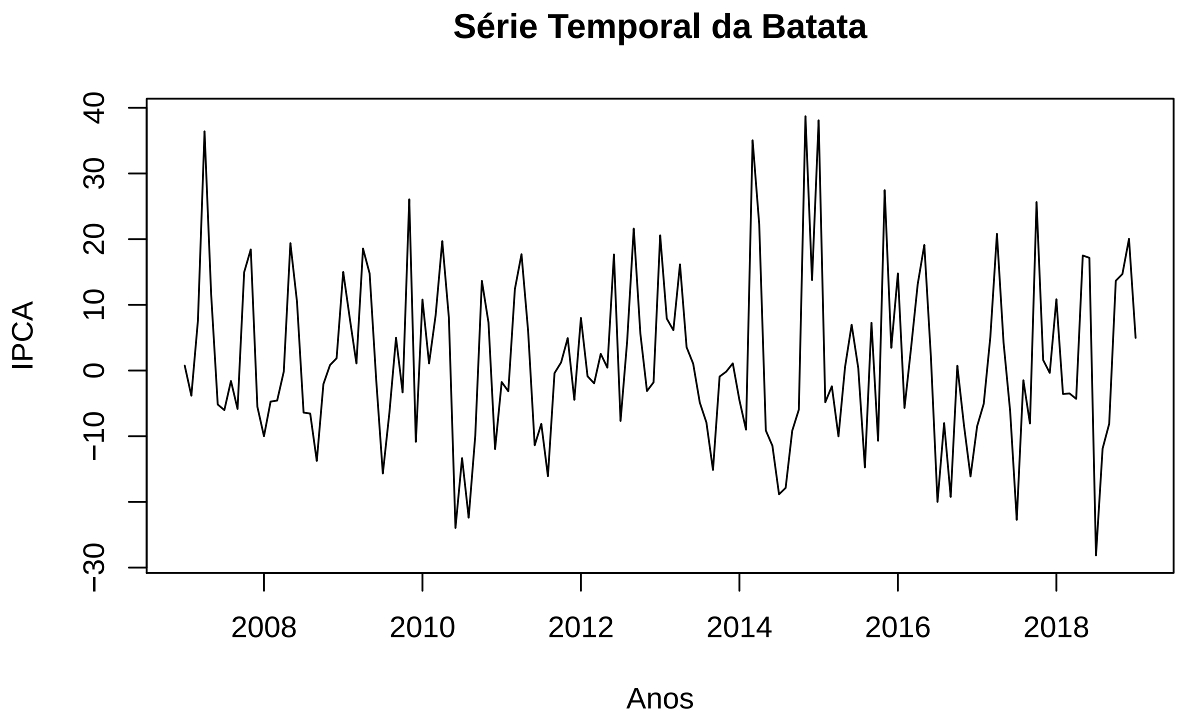

```
plot(zt4,main="Série Temporal de Avicultura de Postura", xlab= "Anos", ylab="IPCA")
```



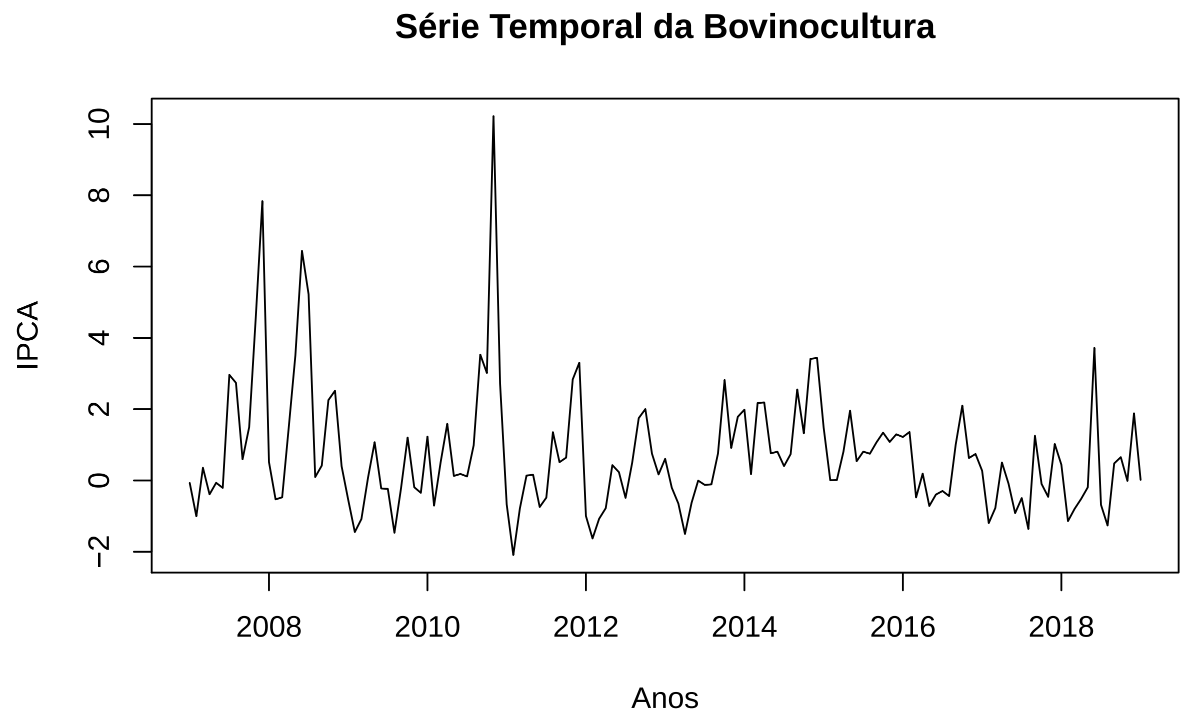

```
plot(zt5,main="Série Temporal da Banana", xlab= "Anos", ylab="IPCA")
```



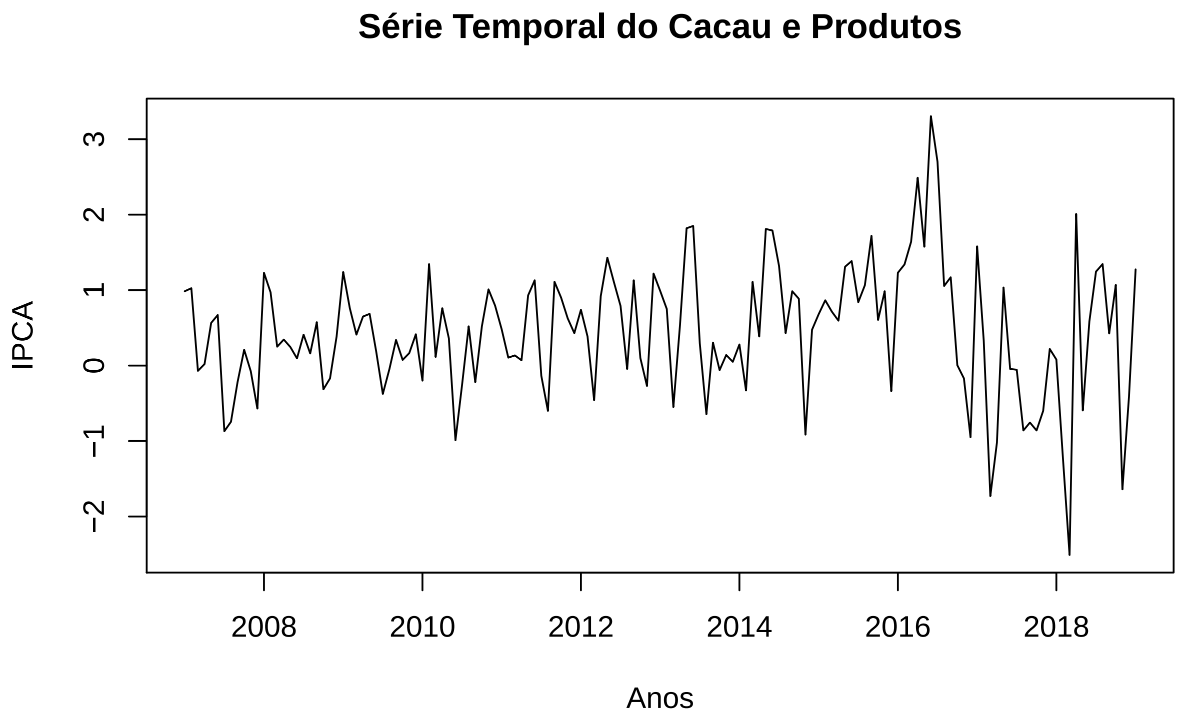

```r
plot(zt6,main="Série Temporal da Batata", xlab= "Anos", ylab="IPCA")
```



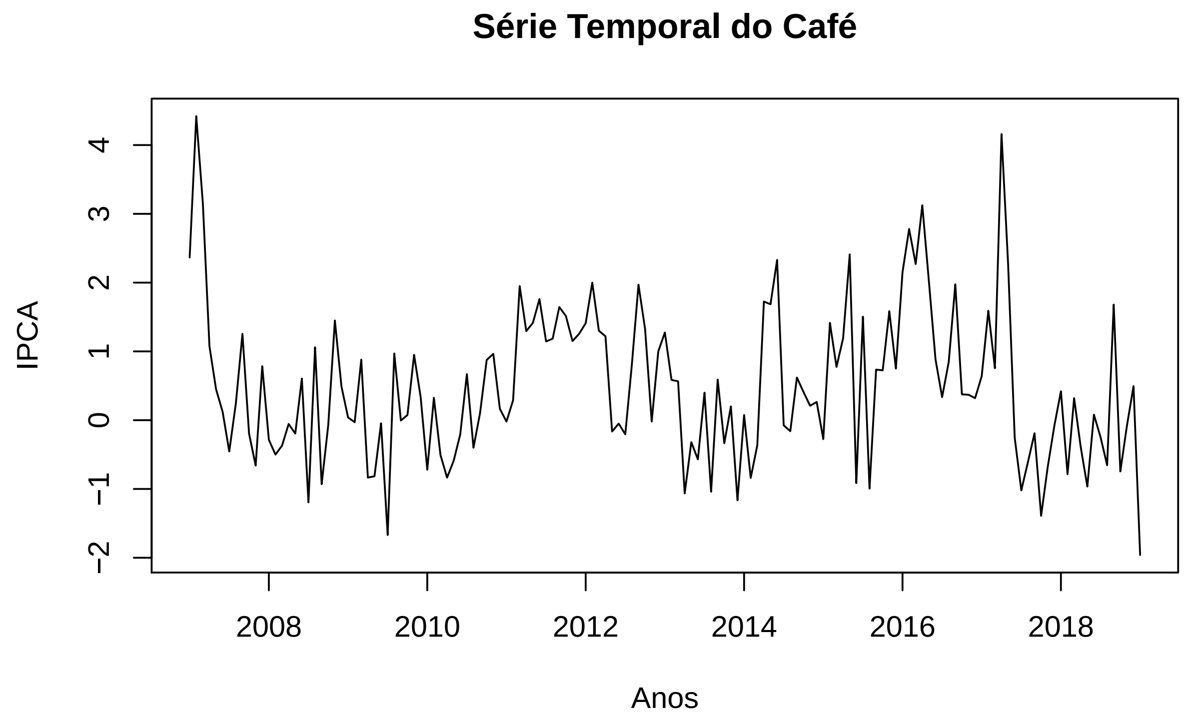

```r
#par(mfrow = c(3, 2))


plot(zt7,main="Série Temporal da Bovinocultura", xlab= "Anos", ylab="IPCA")
```



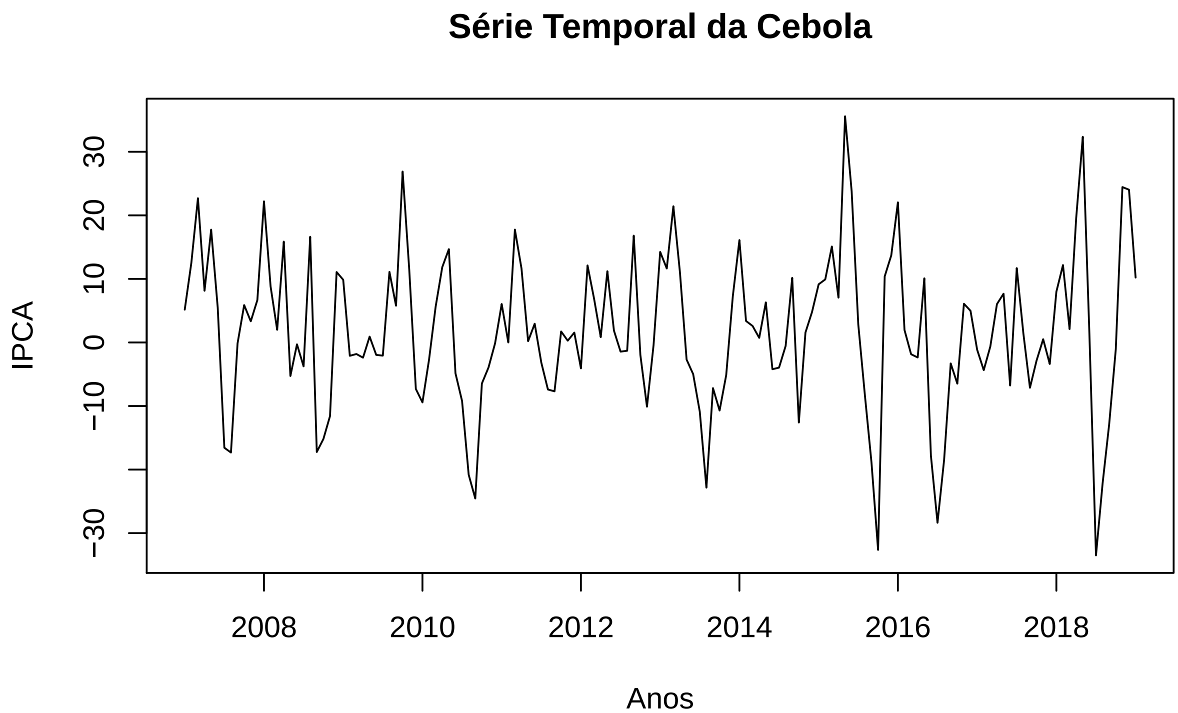

```
plot(zt8,main="Série Temporal do Cacau e Produtos", xlab= "Anos", ylab="IPCA")
```



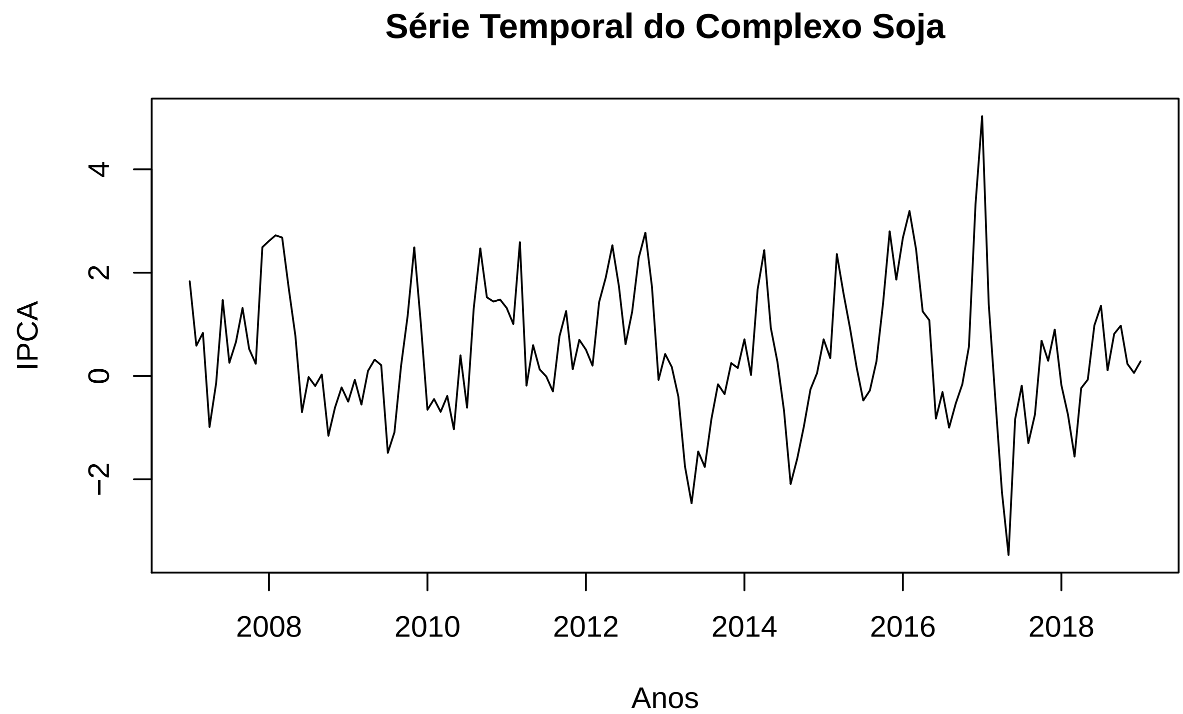

```
plot(zt9,main="Série Temporal do Café", xlab= "Anos", ylab="IPCA")
```



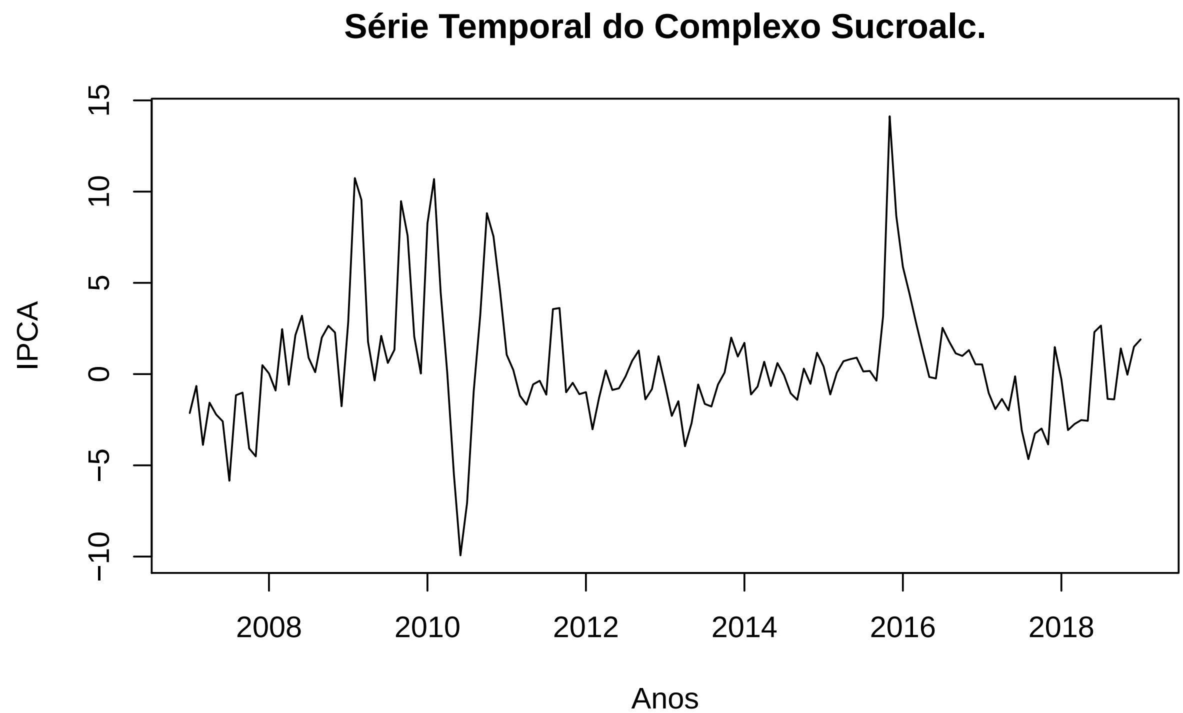

```r
plot(zt10,main="Série Temporal da Cebola", xlab= "Anos", ylab="IPCA")
```



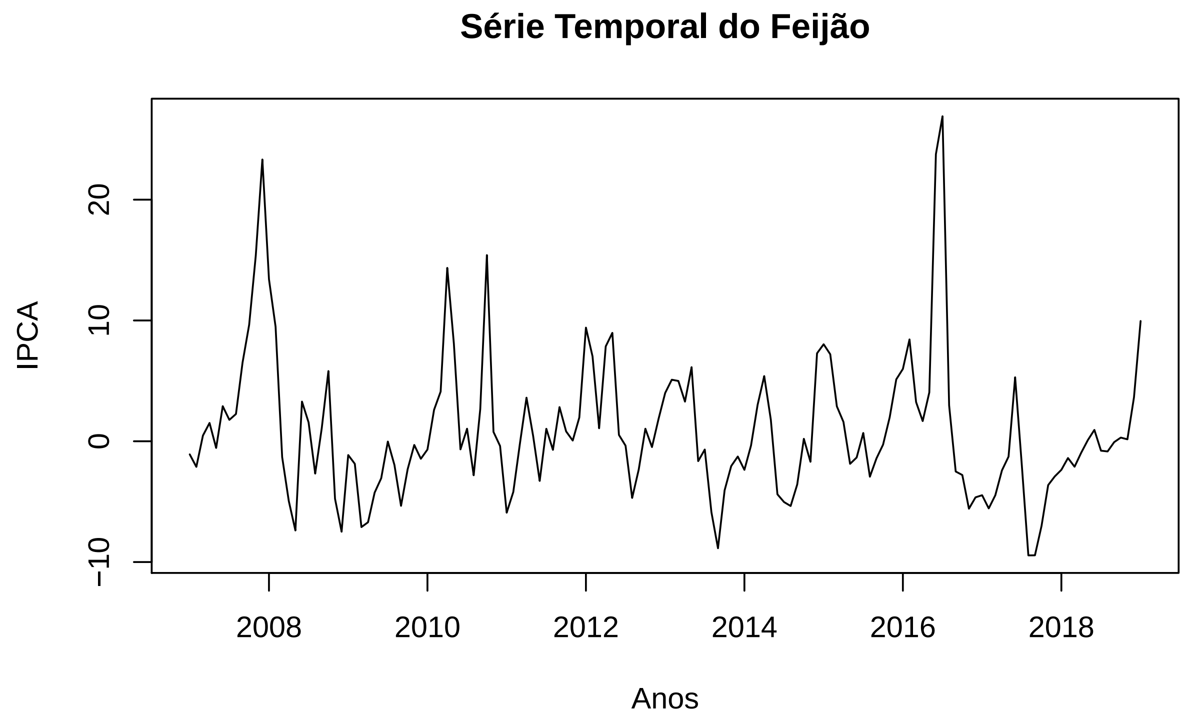

```
plot(zt11,main="Série Temporal do Complexo Soja", xlab= "Anos", ylab="IPCA")
```



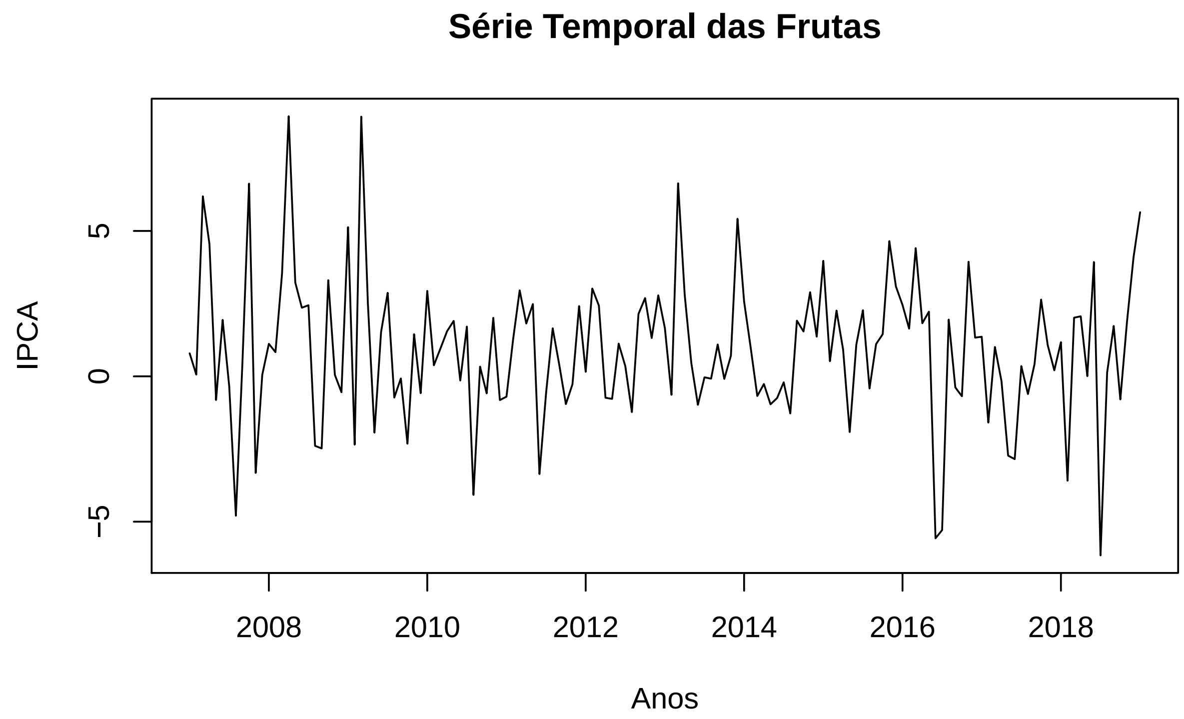

```
plot(zt12,main="Série Temporal do Complexo Sucroalc.", xlab= "Anos", ylab="IPCA")
```



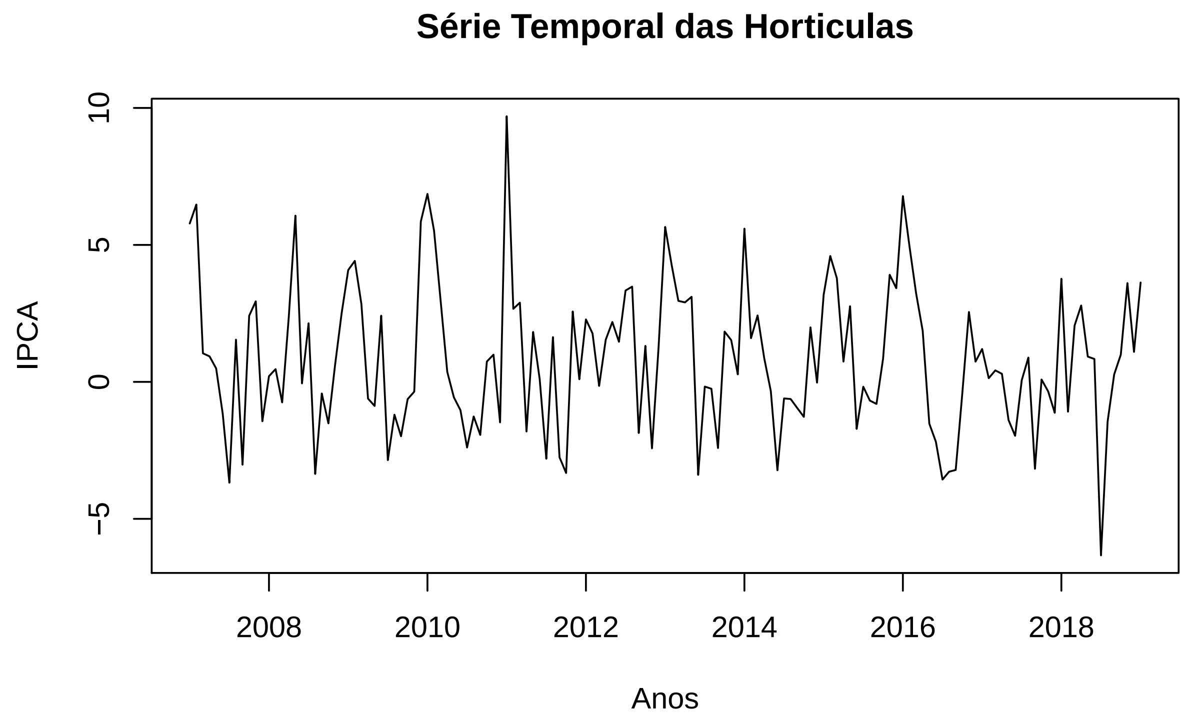

```r
#par(mfrow = c(3, 2))


plot(zt13,main="Série Temporal do Feijão", xlab= "Anos", ylab="IPCA")
```



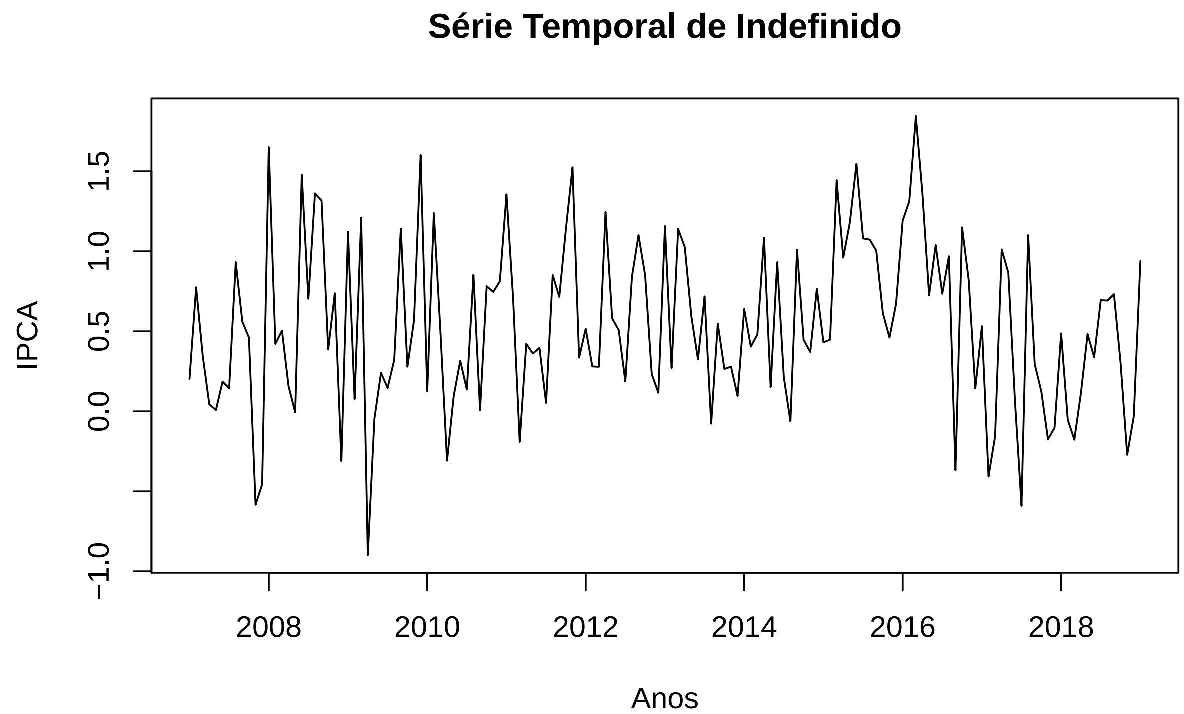

```r
plot(zt14,main="Série Temporal das Frutas", xlab= "Anos", ylab="IPCA")
```



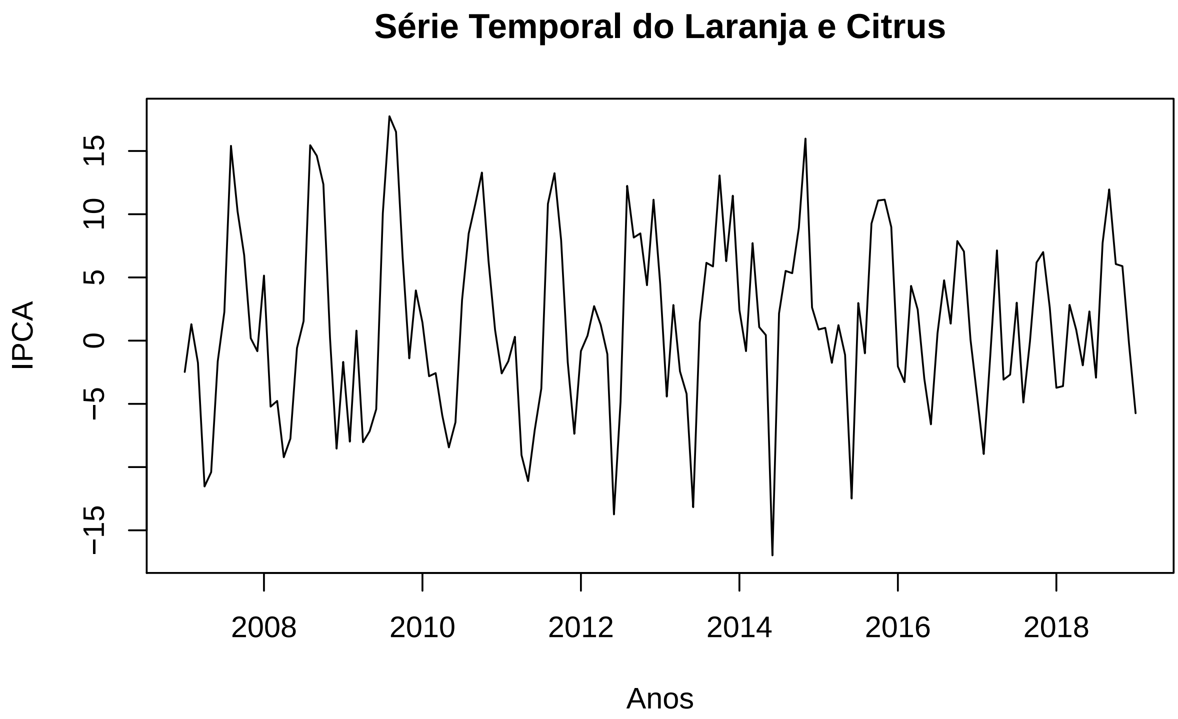

```
plot(zt15,main="Série Temporal das Horticulas", xlab= "Anos", ylab="IPCA")
```



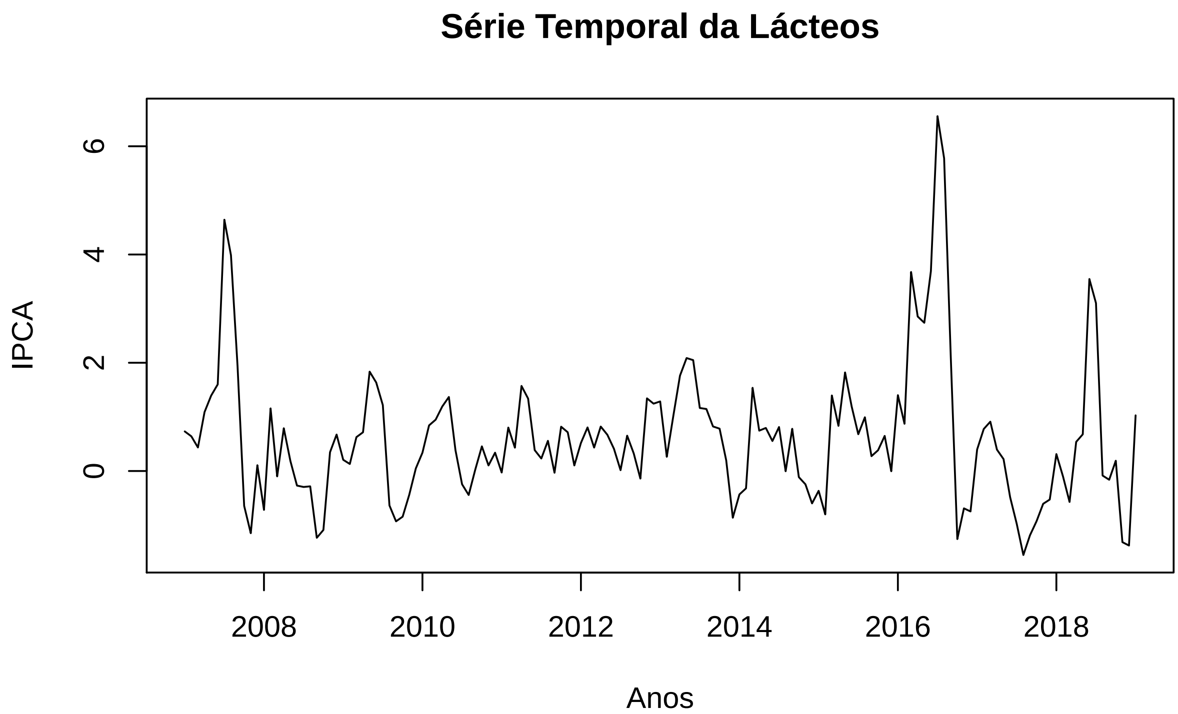

```r
plot(zt16,main="Série Temporal de Indefinido", xlab= "Anos", ylab="IPCA")
```



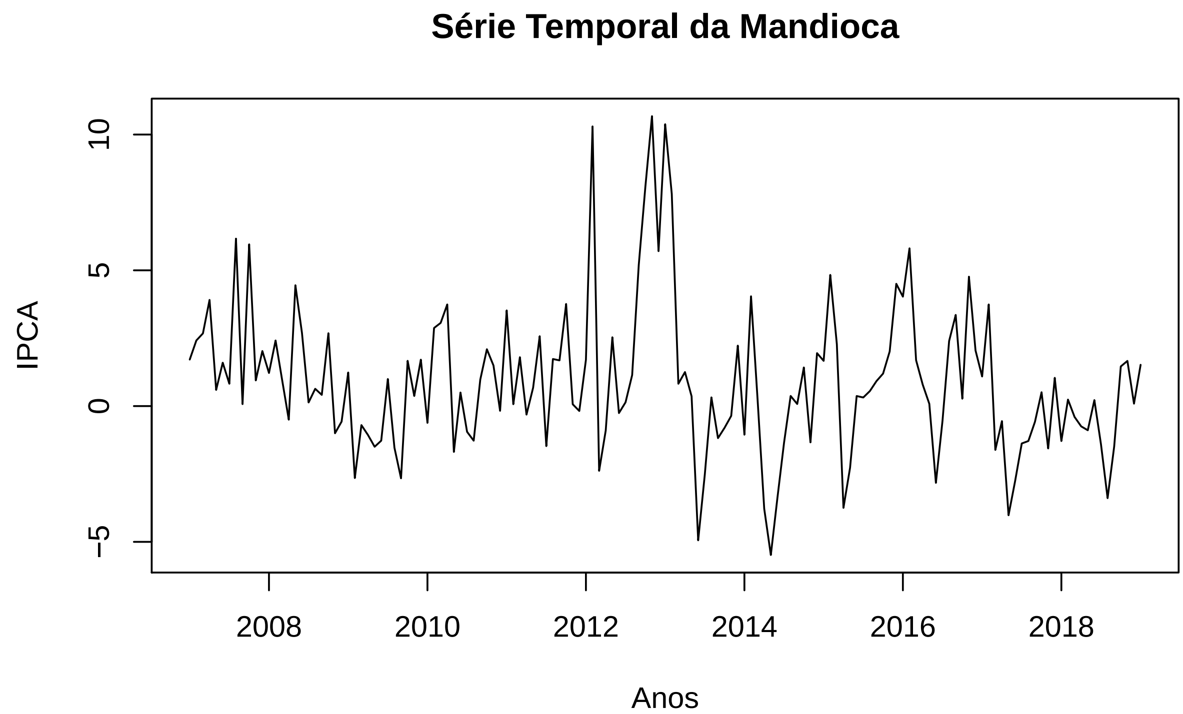

```r
plot(zt17,main="Série Temporal do Laranja e Citrus", xlab= "Anos", ylab="IPCA")
```



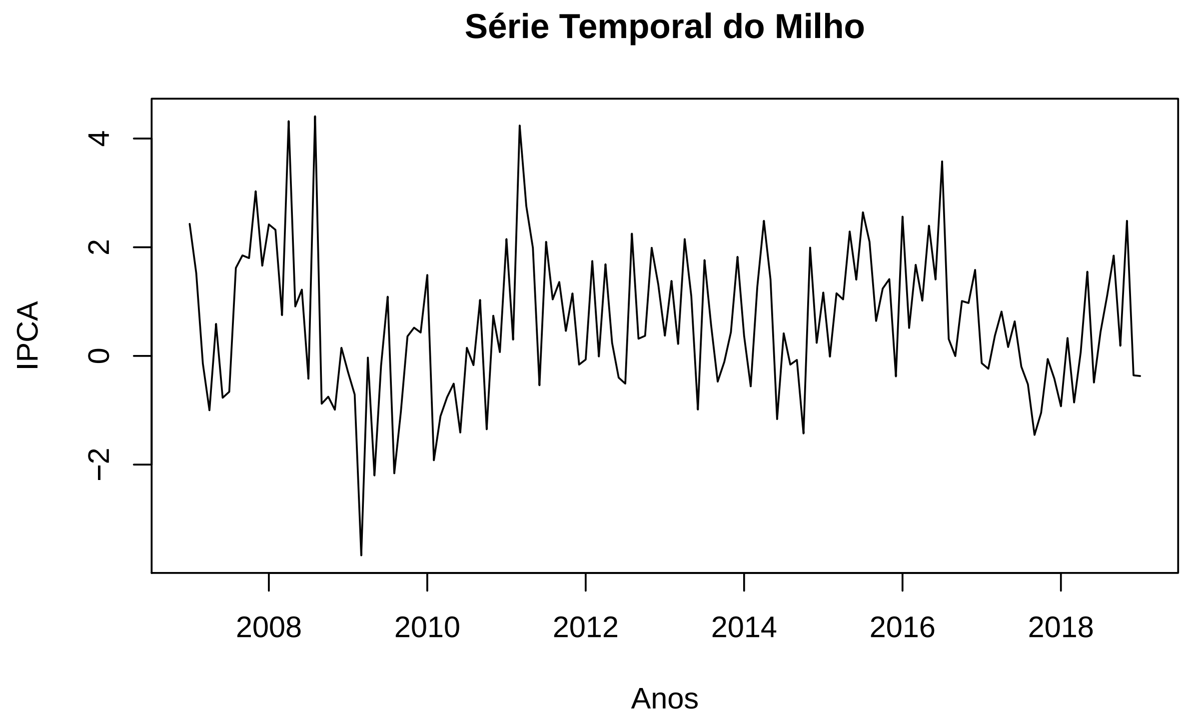

```
plot(zt18,main="Série Temporal da Lácteos", xlab= "Anos", ylab="IPCA")
```



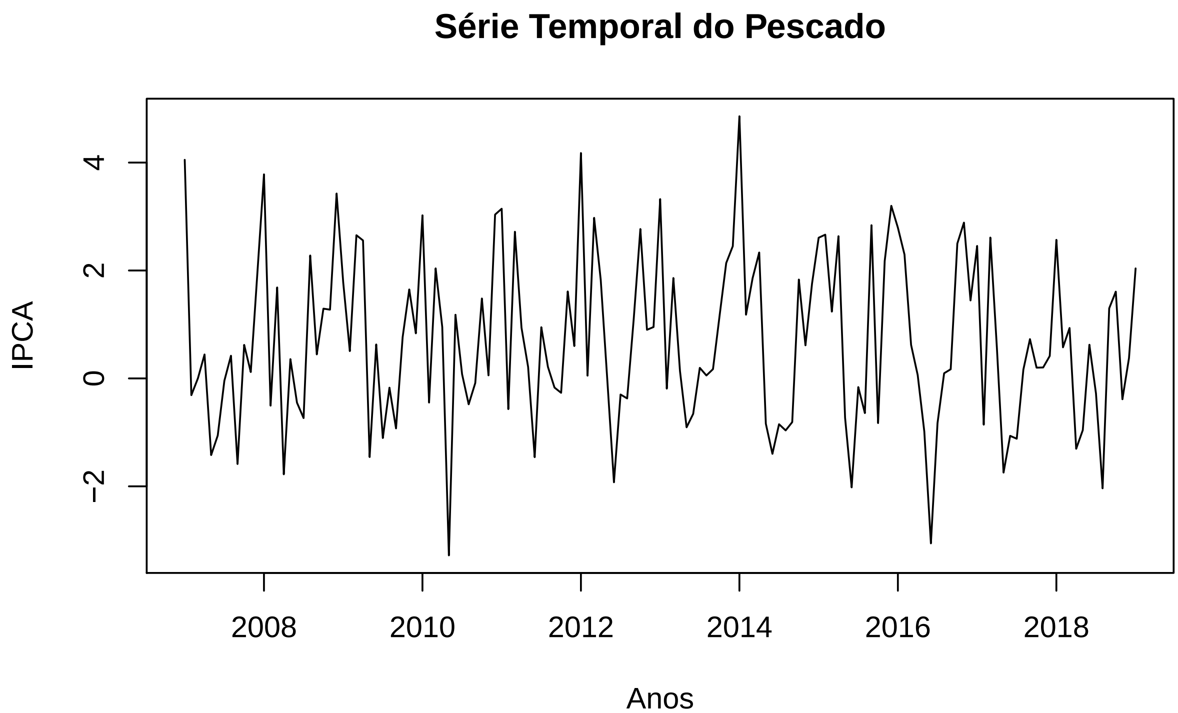

```r
#par(mfrow = c(3, 2))


plot(zt19,main="Série Temporal da Mandioca", xlab= "Anos", ylab="IPCA")
```



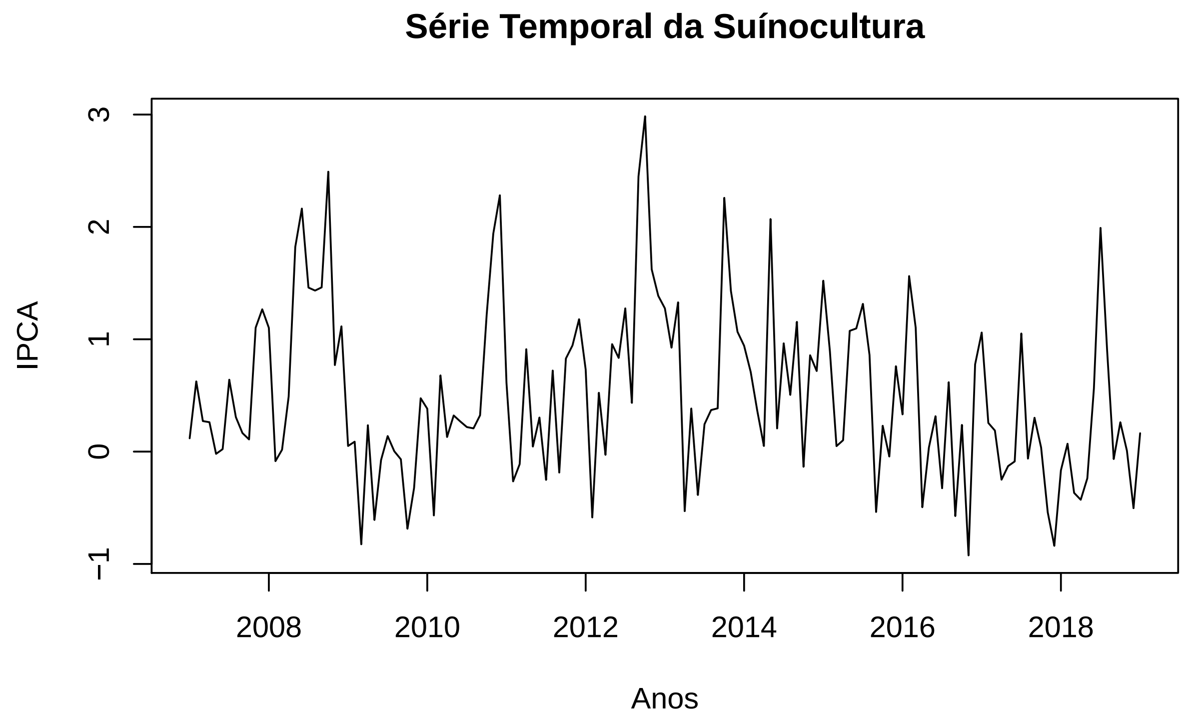

```r
plot(zt20,main="Série Temporal do Milho", xlab= "Anos", ylab="IPCA")
```



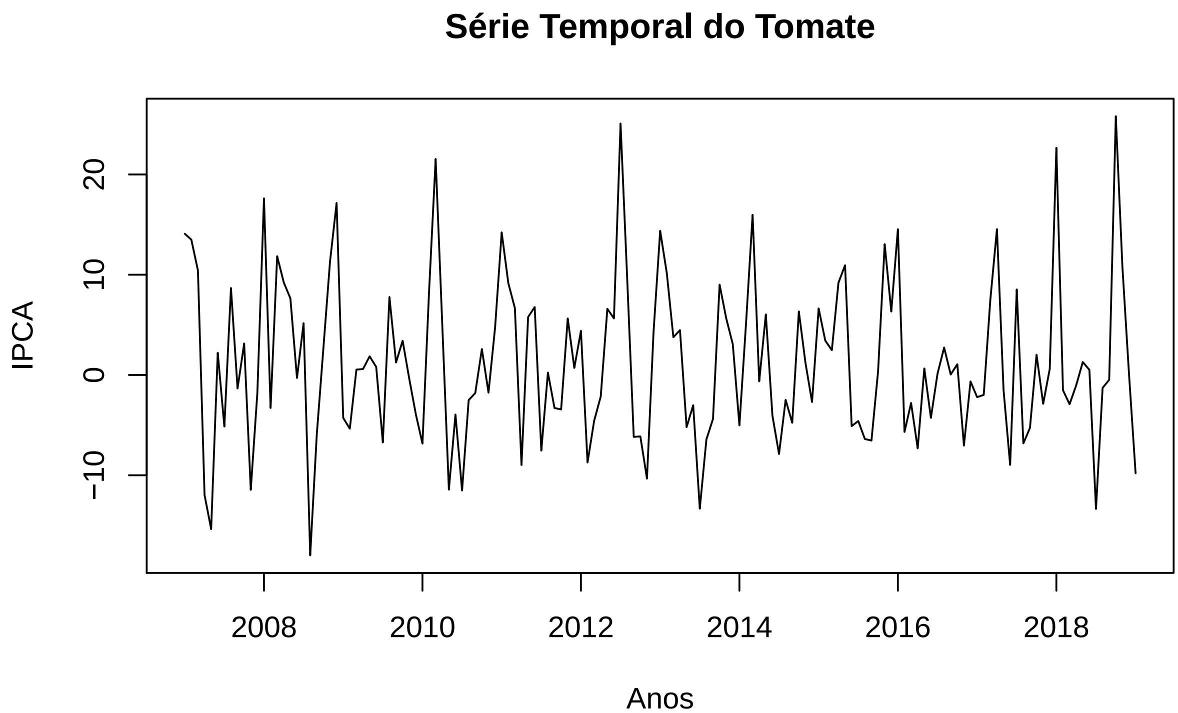

```r
plot(zt21,main="Série Temporal do Pescado", xlab= "Anos", ylab="IPCA")
```



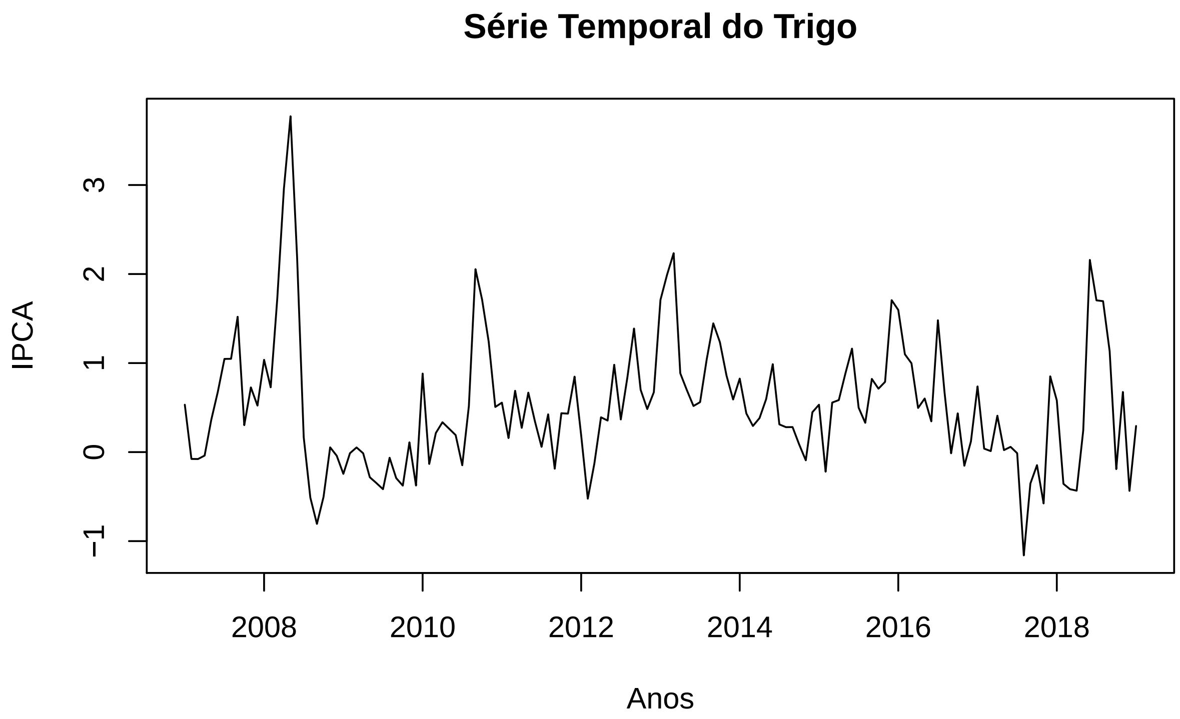

```r
plot(zt22,main="Série Temporal da Suínocultura", xlab= "Anos", ylab="IPCA")
```



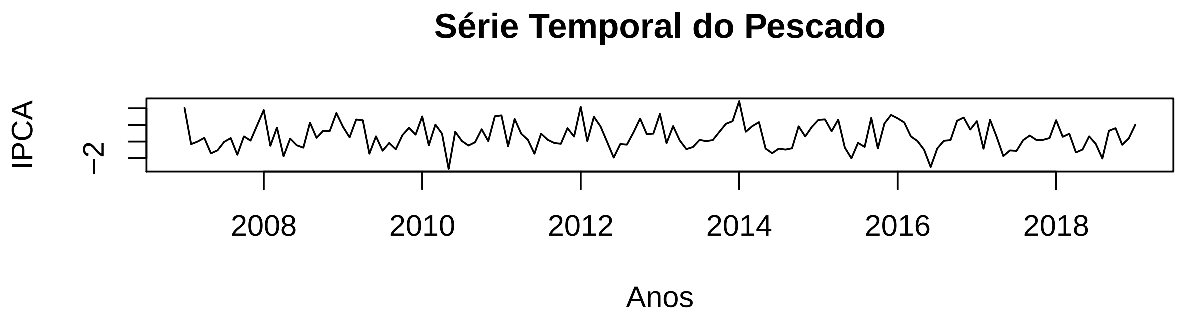

```r
plot(zt23,main="Série Temporal do Tomate", xlab= "Anos", ylab="IPCA")
```

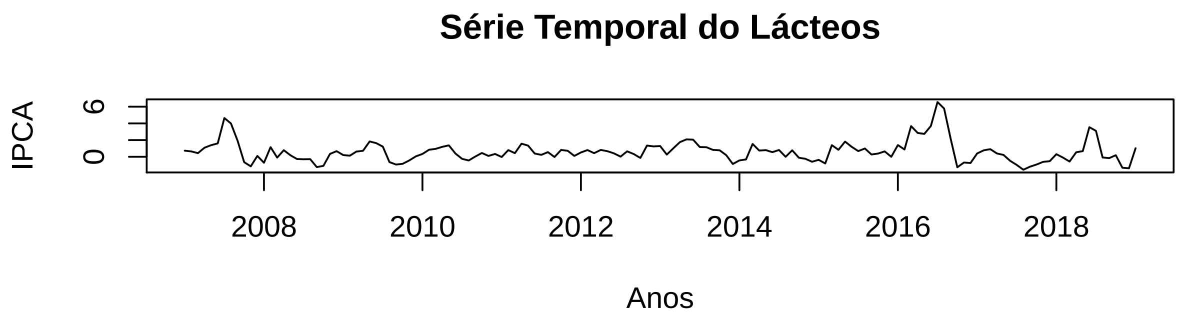



```r
plot(zt24,main="Série Temporal do Trigo", xlab= "Anos", ylab="IPCA")
```



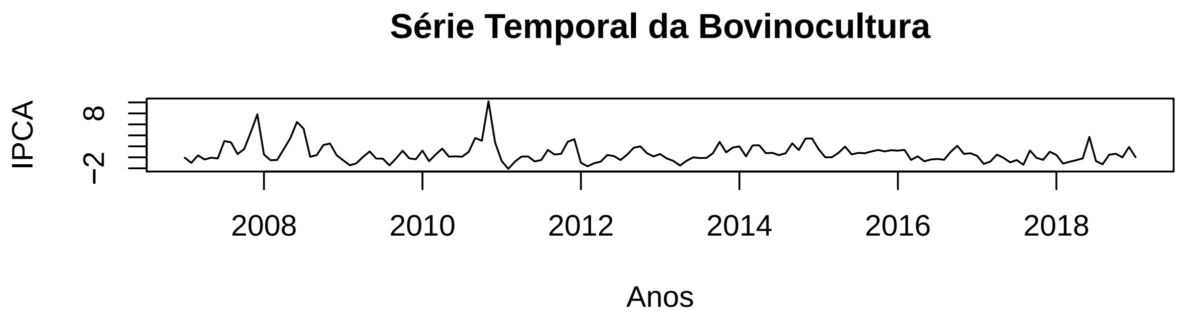

```r
par(mfrow = c(2, 1))
plot(zt21,main="Série Temporal do Pescado", xlab= "Anos", ylab="IPCA")
plot(zt18,main="Série Temporal do Lácteos", xlab= "Anos", ylab="IPCA")
```

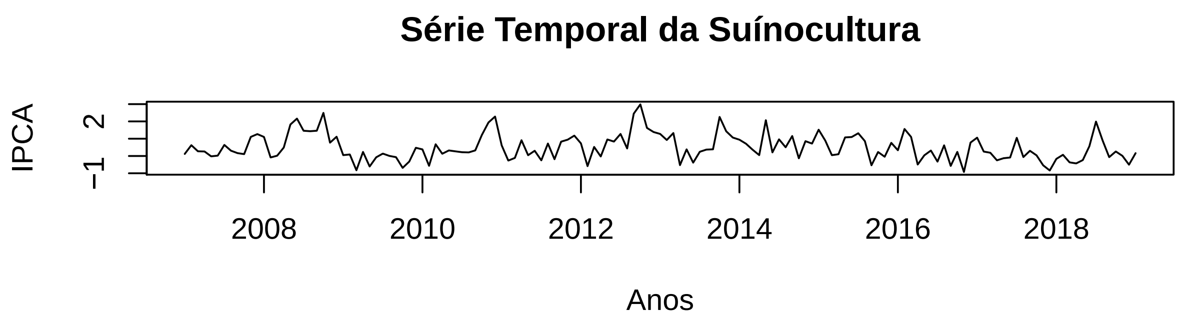




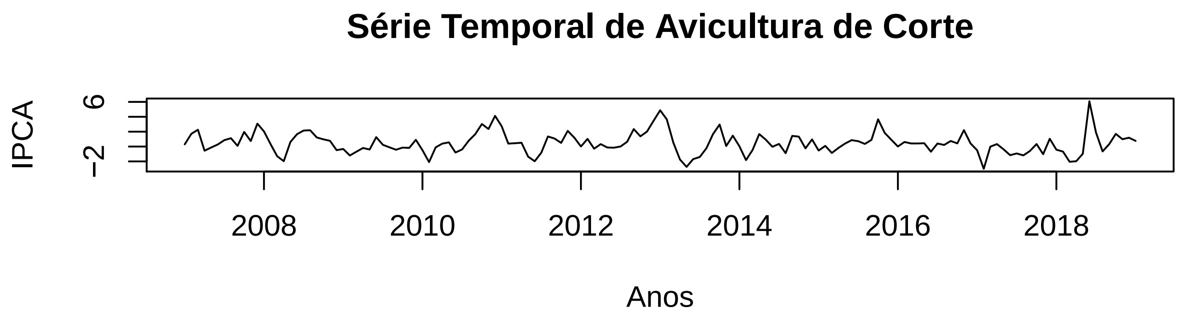

```
#900#650
```

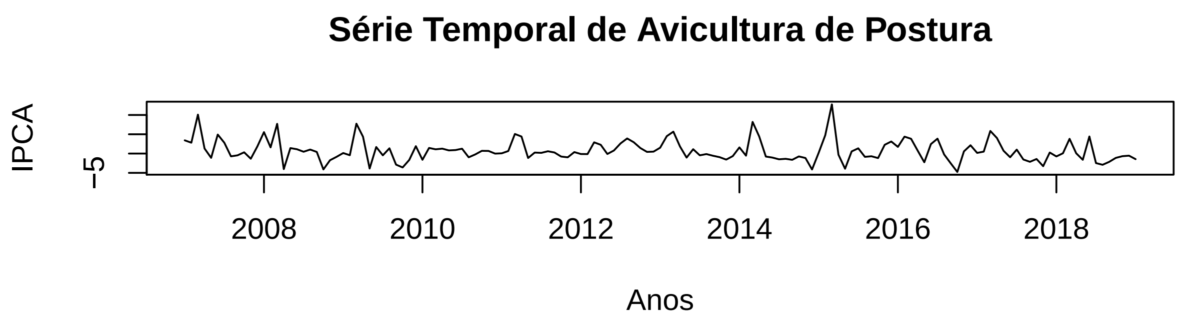
```
par(mfrow = c(2, 1))
plot(zt7,main="Série Temporal da Bovinocultura", xlab= "Anos", ylab="IPCA")
plot(zt22,main="Série Temporal da Suínocultura", xlab= "Anos", ylab="IPCA")
```






```
par(mfrow = c(2, 1))
plot(zt3,main="Série Temporal de Avicultura de Corte", xlab= "Anos", ylab="IPCA")
plot(zt4,main="Série Temporal de Avicultura de Postura", xlab= "Anos", ylab="IPCA")
```

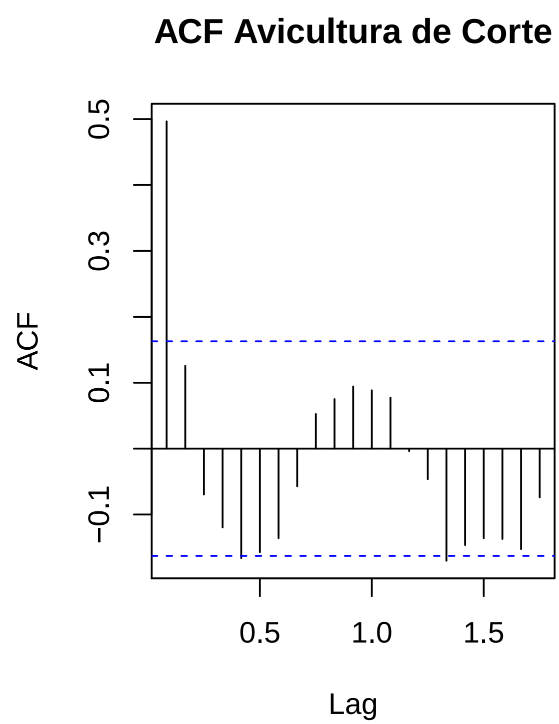


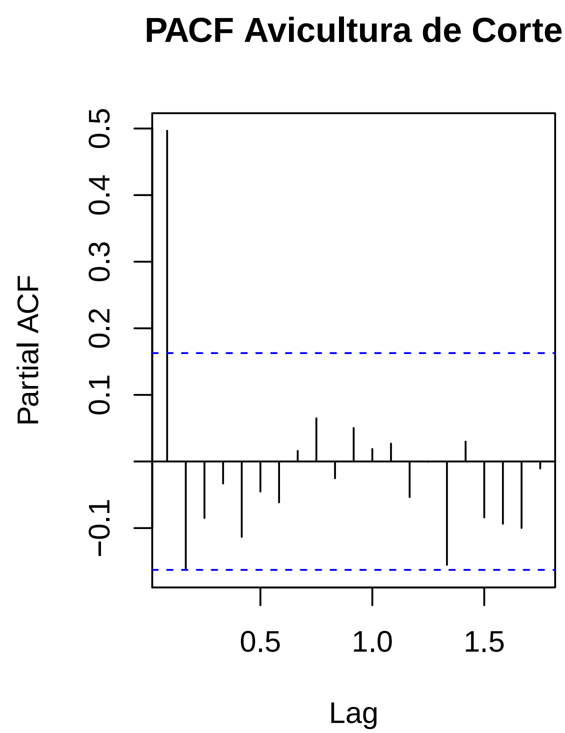


Funções de Autocorrelações

Funções de Autocorrelações para Avicultura de Corte

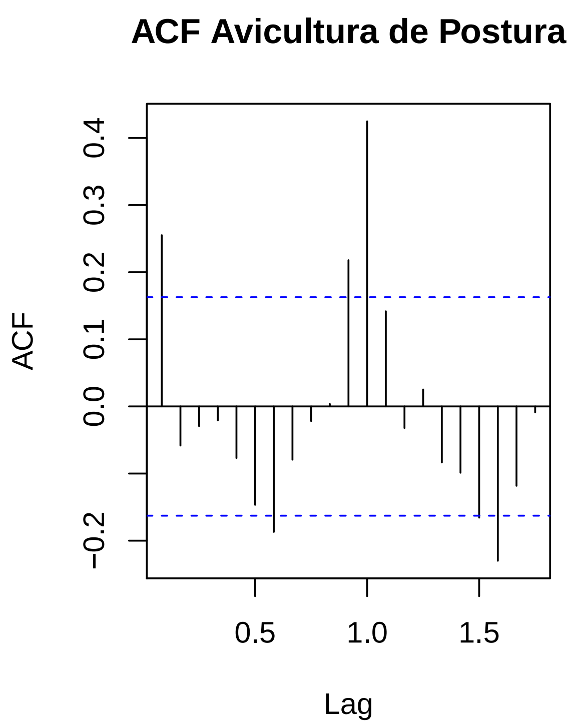
```
#Funções de Autocorrelações para Avicultura de Corte
par(mfrow = c(1, 2))
acf(zt3, main="ACF Avicultura de Corte")
pacf(zt3, main="PACF Avicultura de Corte")
```

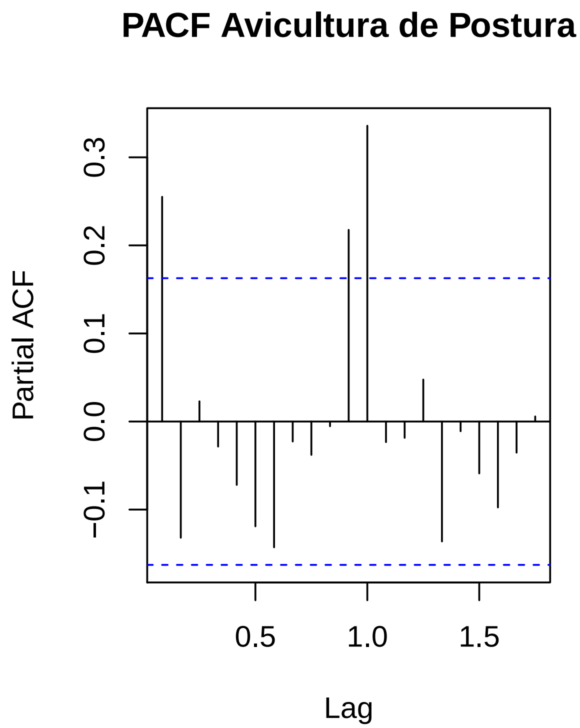




Funções de Autocorrelações para Avicultura de Postura

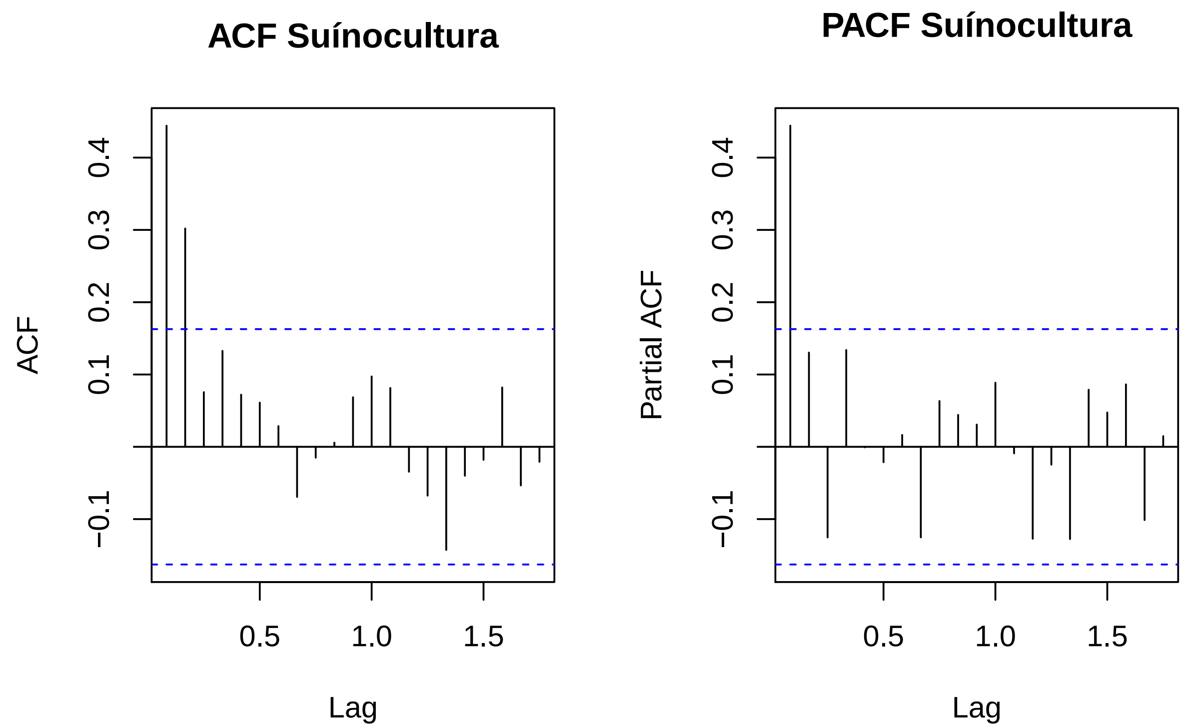
```r
#Funções de Autocorrelações para Avicultura de Postura
par(mfrow = c(1, 2))
acf(zt4, main="ACF Avicultura de Postura")
pacf(zt4, main="PACF Avicultura de Postura")
```





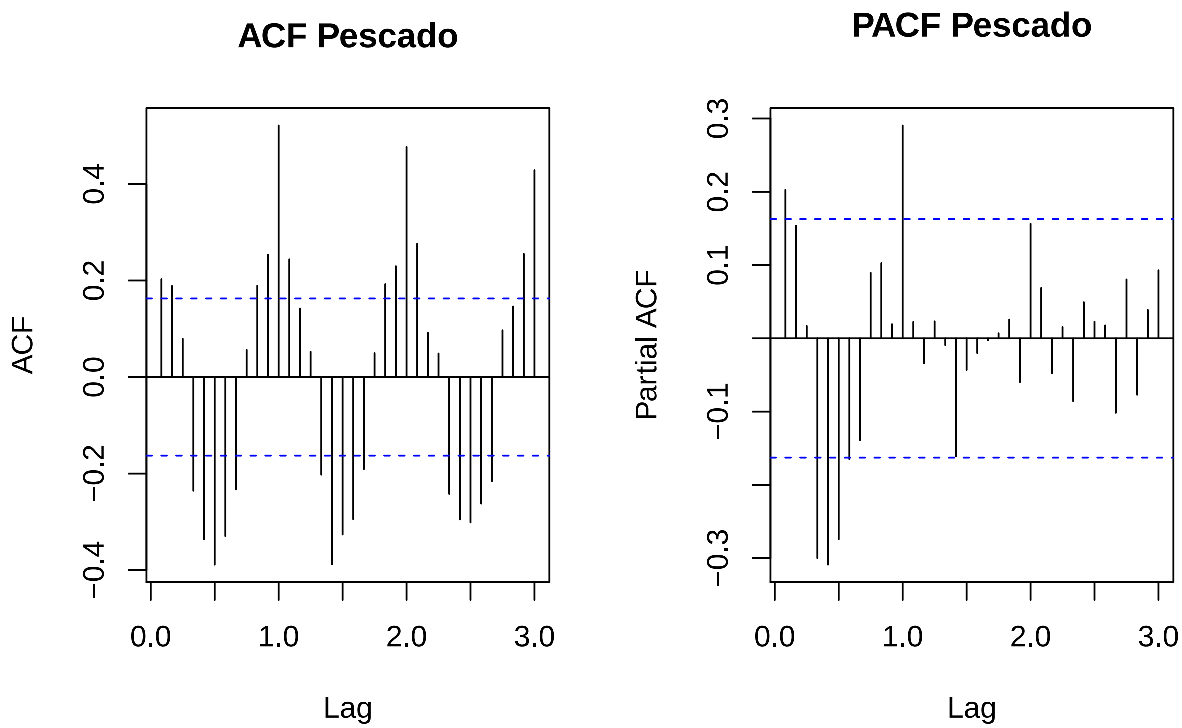
Funções de Autocorrelações para Suinocultura

```
#Funções de Autocorrelações para Suinocultura
par(mfrow = c(1, 2))
acf(zt22, main="ACF Suínocultura")
pacf(zt22, main="PACF Suínocultura")
```



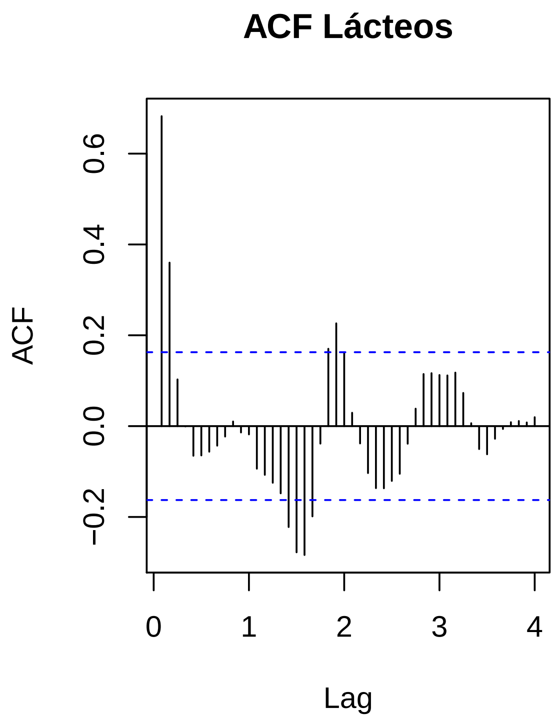
Funções de Autocorrelações para Pescado

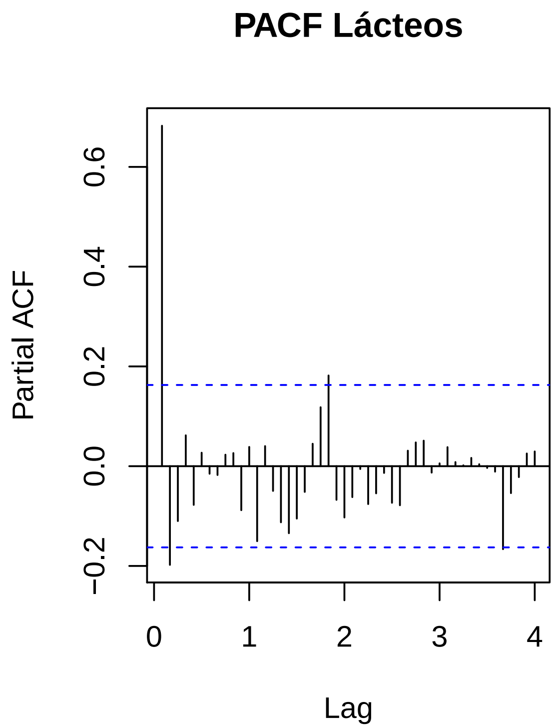
```
#Funções de Autocorrelações para Pescado
par(mfrow = c(1, 2))
acf(zt21, main="ACF Pescado", lag.max = 36)
pacf(zt21, main="PACF Pescado", lag.max = 36)
```



Funções de Autocorrelações para Lácteos

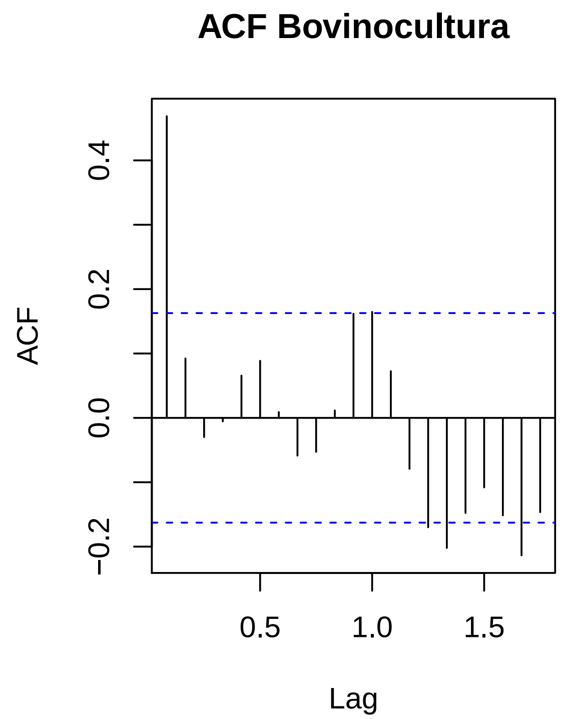
```
#Funções de Autocorrelações para Lácteos
par(mfrow = c(1, 2))
acf(zt18, main="ACF Lácteos", lag.max = 48)
pacf(zt18, main="PACF Lácteos", lag.max = 48)
```

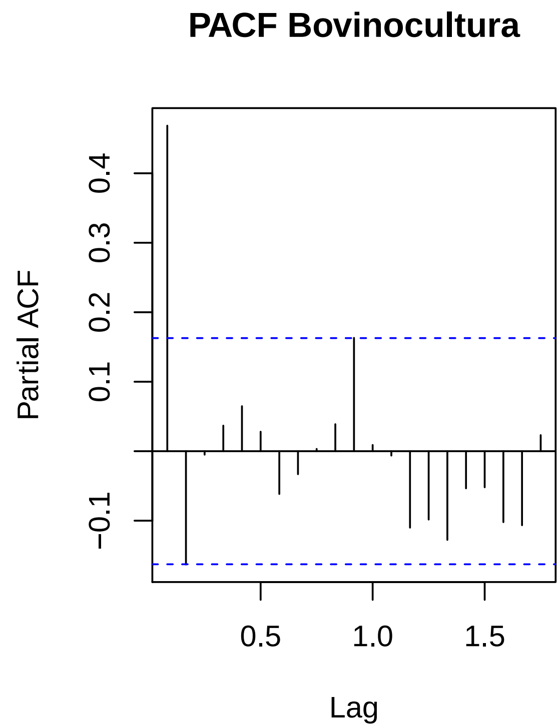




Funções de Autocorrelações para Bovinocultura

```
#Funções de Autocorrelações para Bovinocultura
par(mfrow = c(1, 2))
acf(zt7, main="ACF Bovinocultura")
pacf(zt7, main="PACF Bovinocultura")
```





Testes de Dickey-Fuller e Phillips-Perron

Teste de Dickey-Fuller

```r
# Teste de Dickey-Fuller
adf.test(zt7)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  zt7
## Dickey-Fuller = -4.4888, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```r
adf.test(zt3)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  zt3
## Dickey-Fuller = -5.4727, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```r
adf.test(zt4)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  zt4
## Dickey-Fuller = -6.117, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```r
adf.test(zt18)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  zt18
## Dickey-Fuller = -4.3253, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```r
adf.test(zt21)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  zt21
## Dickey-Fuller = -8.7741, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```r
adf.test(zt22)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  zt22
## Dickey-Fuller = -4.0878, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

Teste de Phillips-Perron

```r
# Teste de Phillips-Perron
pp.test(zt7)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  zt7
## Dickey-Fuller Z(alpha) = -70.675, Truncation lag parameter = 4, p-value
## = 0.01
## alternative hypothesis: stationary
```

```r
pp.test(zt3)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  zt3
## Dickey-Fuller Z(alpha) = -69.133, Truncation lag parameter = 4, p-value
## = 0.01
## alternative hypothesis: stationary
```

```r
pp.test(zt4)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  zt4
## Dickey-Fuller Z(alpha) = -99.344, Truncation lag parameter = 4, p-value
## = 0.01
## alternative hypothesis: stationary
```

```r
pp.test(zt18)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  zt18
## Dickey-Fuller Z(alpha) = -47.067, Truncation lag parameter = 4, p-value
## = 0.01
## alternative hypothesis: stationary
```

```r
pp.test(zt21)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  zt21
## Dickey-Fuller Z(alpha) = -125.86, Truncation lag parameter = 4, p-value
## = 0.01
## alternative hypothesis: stationary
```

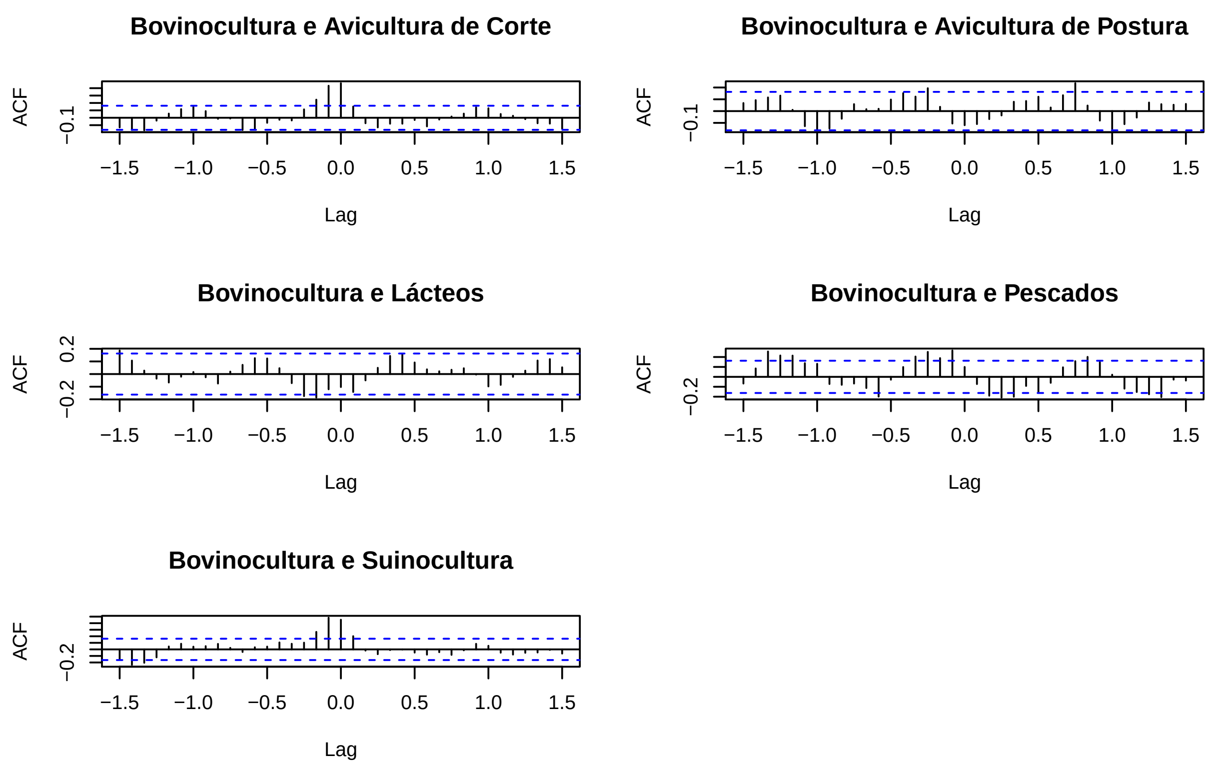
```r
pp.test(zt22)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  zt22
## Dickey-Fuller Z(alpha) = -84.151, Truncation lag parameter = 4, p-value
## = 0.01
## alternative hypothesis: stationary
```

Análise Correlação Cruzada

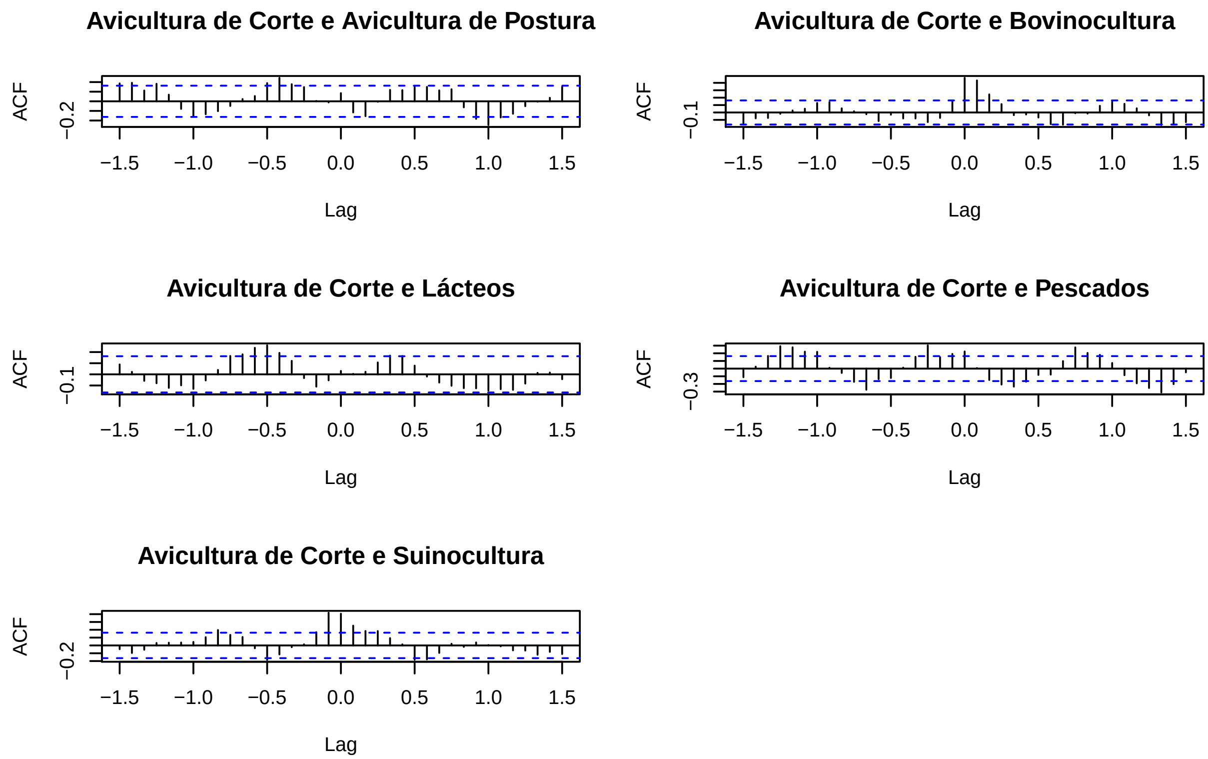
Correlaões cruzadas da Bovincultura

```
#Correlaões cruzadas da Bovincultura
par(mfrow = c(3,2))
ccf(zt7,zt3,main="Bovinocultura e Avicultura de Corte")
ccf(zt7,zt4,main="Bovinocultura e Avicultura de Postura")
ccf(zt7,zt18,main="Bovinocultura e Lácteos")
ccf(zt7,zt21,main="Bovinocultura e Pescados")
ccf(zt7,zt22,main="Bovinocultura e Suinocultura")
```



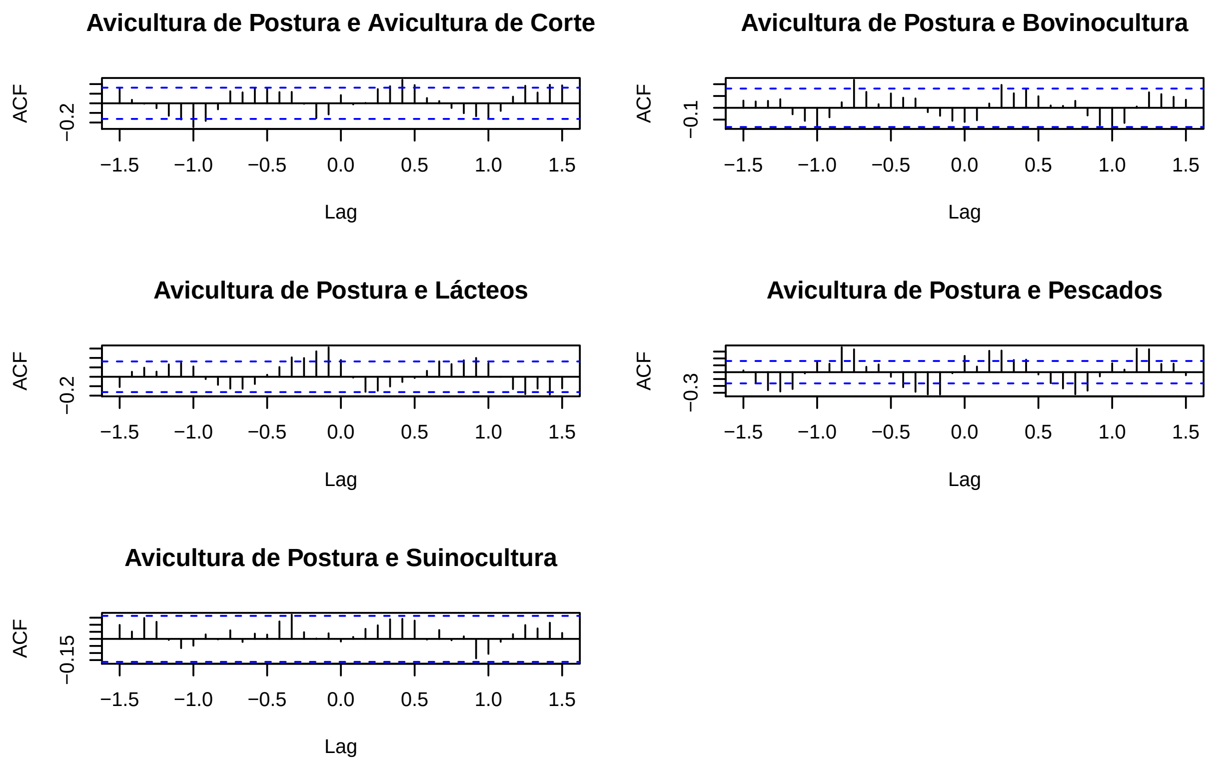
Correlações cruzadas da Avicultura de Corte

```
#Correlações cruzadas da Avicultura de Corte
par(mfrow = c(3,2))
ccf(zt3,zt4,main="Avicultura de Corte e Avicultura de Postura")
ccf(zt3,zt7,main="Avicultura de Corte e Bovinocultura")
ccf(zt3,zt18,main="Avicultura de Corte e Lácteos")
ccf(zt3,zt21,main="Avicultura de Corte e Pescados")
ccf(zt3,zt22,main="Avicultura de Corte e Suinocultura")
```



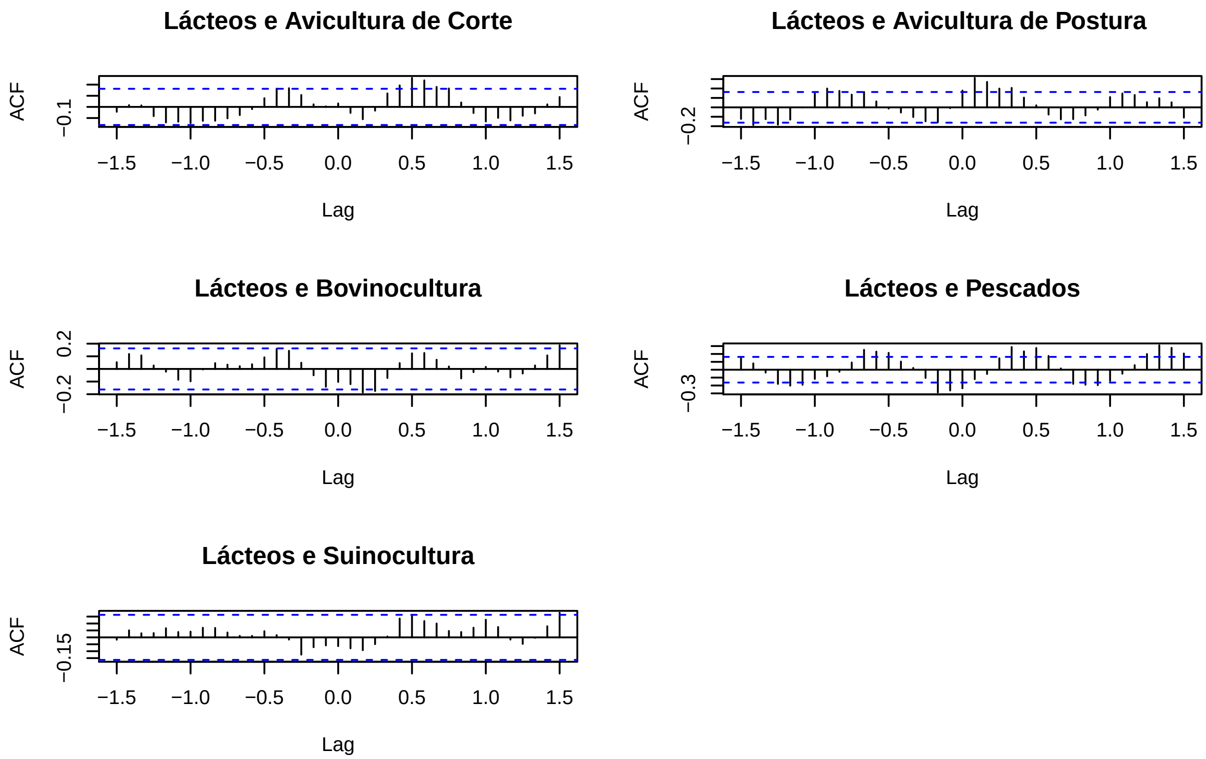
Correlações cruzadas da Avicultura de Postura

```
#Correlações cruzadas da Avicultura de Postura
par(mfrow = c(3,2))
ccf(zt4,zt3,main="Avicultura de Postura e Avicultura de Corte")
ccf(zt4,zt7,main="Avicultura de Postura e Bovinocultura")
ccf(zt4,zt18,main="Avicultura de Postura e Lácteos")
ccf(zt4,zt21,main="Avicultura de Postura e Pescados")
ccf(zt4,zt22,main="Avicultura de Postura e Suinocultura")
```



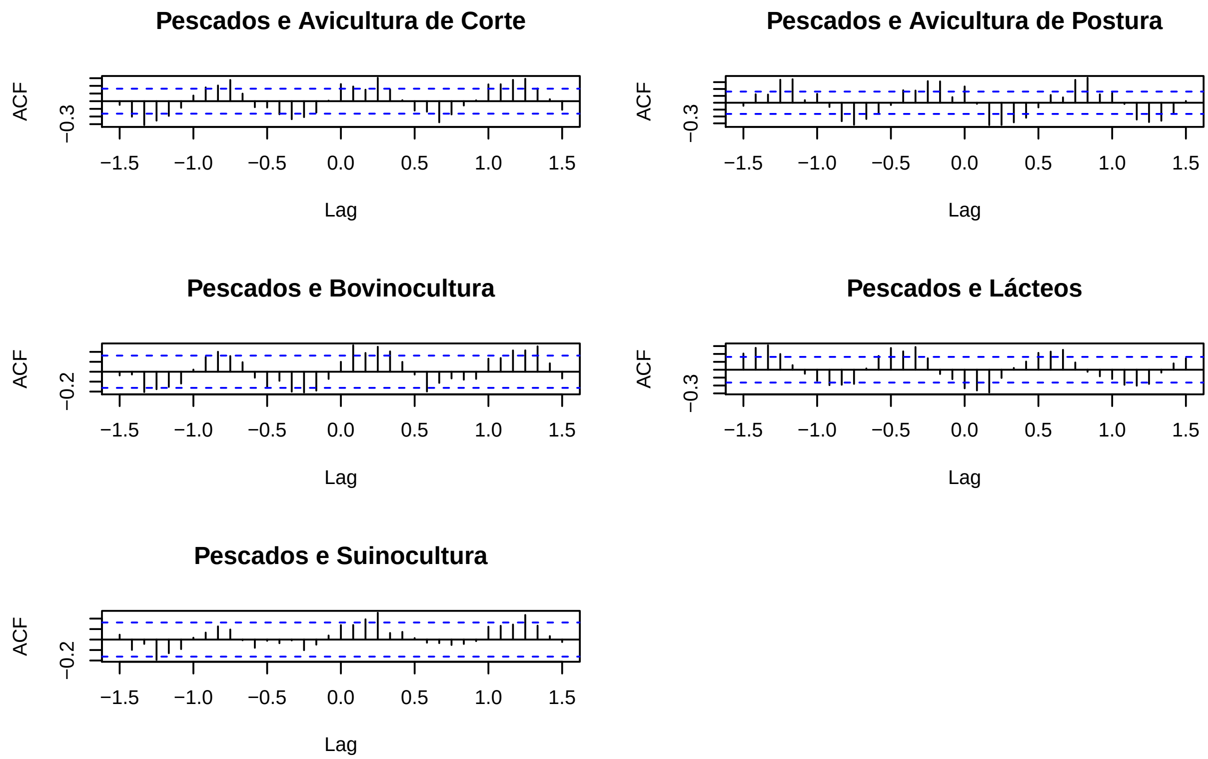
Correlações cruzadas dos Lácteos

```
#Correlações cruzadas dos Lácteos
par(mfrow = c(3,2))
ccf(zt18,zt3,main="Lácteos e Avicultura de Corte")
ccf(zt18,zt4,main="Lácteos e Avicultura de Postura ")
ccf(zt18,zt7,main="Lácteos e Bovinocultura")
ccf(zt18,zt21,main="Lácteos e Pescados")
ccf(zt18,zt22,main="Lácteos e Suinocultura")
```



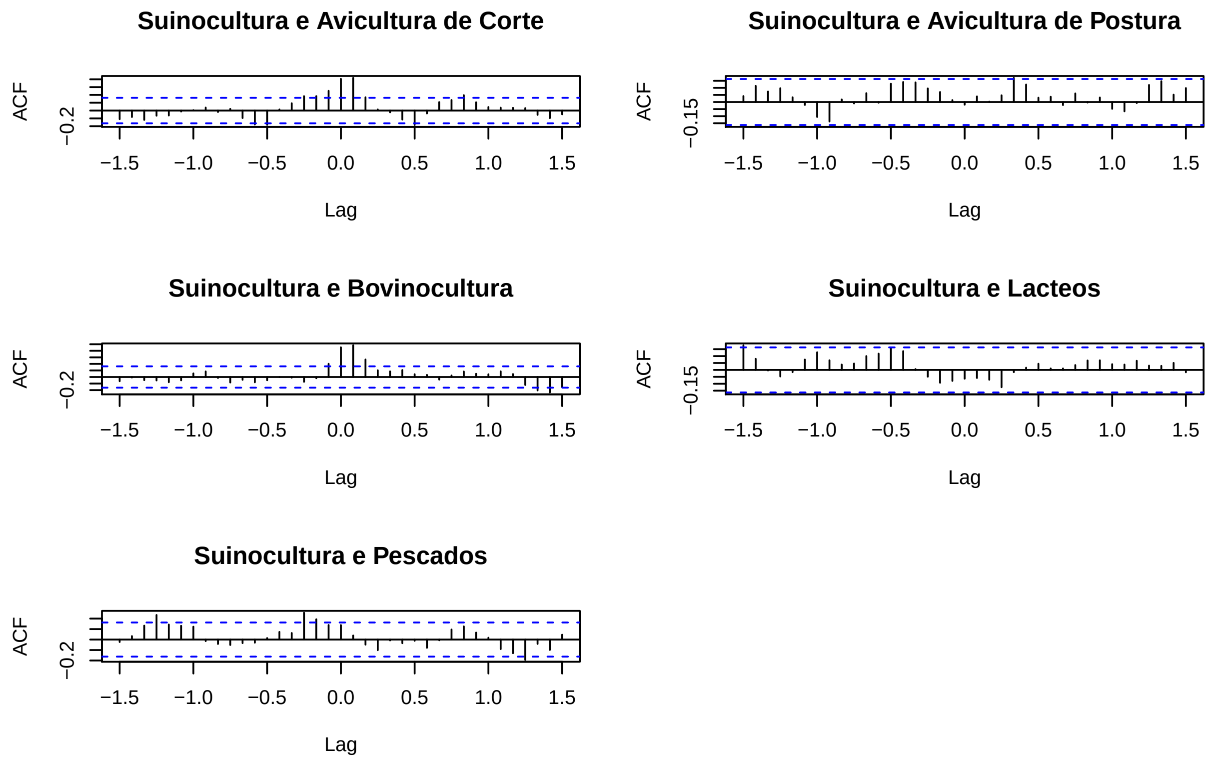
Correlaões cruzadas dos Pescados

```
# Correlações cruzadas dos Pescados
par(mfrow = c(3,2))
ccf(zt21,zt3,main="Pescados e Avicultura de Corte")
ccf(zt21,zt4,main="Pescados e Avicultura de Postura")
ccf(zt21,zt7,main="Pescados e Bovinocultura")
ccf(zt21,zt18,main="Pescados e Lácteos")
ccf(zt21,zt22,main="Pescados e Suinocultura")
```



Correlações cruzadas da Suinocultura

```
#Correlações cruzadas da Suinocultura
par(mfrow = c(3,2))
ccf(zt22,zt3,main="Suinocultura e Avicultura de Corte")
ccf(zt22,zt4,main="Suinocultura e Avicultura de Postura")
ccf(zt22,zt7,main="Suinocultura e Bovinocultura")
ccf(zt22,zt18,main="Suinocultura e Lacteos")
ccf(zt22,zt21,main="Suinocultura e Pescados")
```



Selecionado as variáveis de interesse do estudo

Essa função retorna a coluna com a lag a ser considerada na análise

```
#Essa função retorna a coluna com a lag a ser considerada na análise

funcao_lags = function(df,coluna,nome,lag){
  n = nrow(df)
  pre = rep(NA,lag)
  newcol = c(pre,coluna)
  for (k in 1:lag){
    df = rbind(df,rep(NA,ncol(df)))
  }
  df[nome] = newcol
  return (df)
}
```

A função a baixo retira as variáveis do modelo em função do p-valor

```
#A função a baixo retira as variáveis do modelo em função do p-valor
tirar_variaveis = function(p,d,q,x,y){
  v = p + q + 1
  max = 0.06
  while (max > 0.05){
    model = Arima(y,order=c(p,d,q),xreg = x)
    ct = coeftest(model)
```

```
    pvalues = ct[(v+1):nrow(ct),4]
    maxi = which.max(pvalues)
    max = ct[v + maxi,4]
    if (max > 0.05) {
      x = x[,-maxi]
      }
  }
  lista = list(ct, x)
  return (lista)
}
```

A seguir vamos selecionar apenas as variáveis de interesse para análise

```
#A seguir vamos selecionar apenas as variáveis de interesse para análise
data_cut = data[,c("Bovinocultura","Avicultura de Corte","Avicultura de Postura","Pescado","Lácteos","Su
```

Modelo da Bovinocultura

Estruturando a base

```
#Estruturando a base
df1<- funcao_lags(data_cut, data_cut$`Avicultura de Postura`, 'avp9', 9)
df1 <- funcao_lags(df1, df1$Pescado, 'p3', 3)
df1 <- funcao_lags(df1, df1$Pescado, 'p10', 10)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'b1', 1)


df2 <- na.omit(df1)
```

Separando variável preditora e as covariáveis

```
#Separando variável preditora e as covariáveis
x = model.matrix(Bovinocultura~.,df2)[,-1]
y = df2$Bovinocultura
```

Regressão classifica no contexto de Séries Temporais

Criando o modelo de Regressão Simples

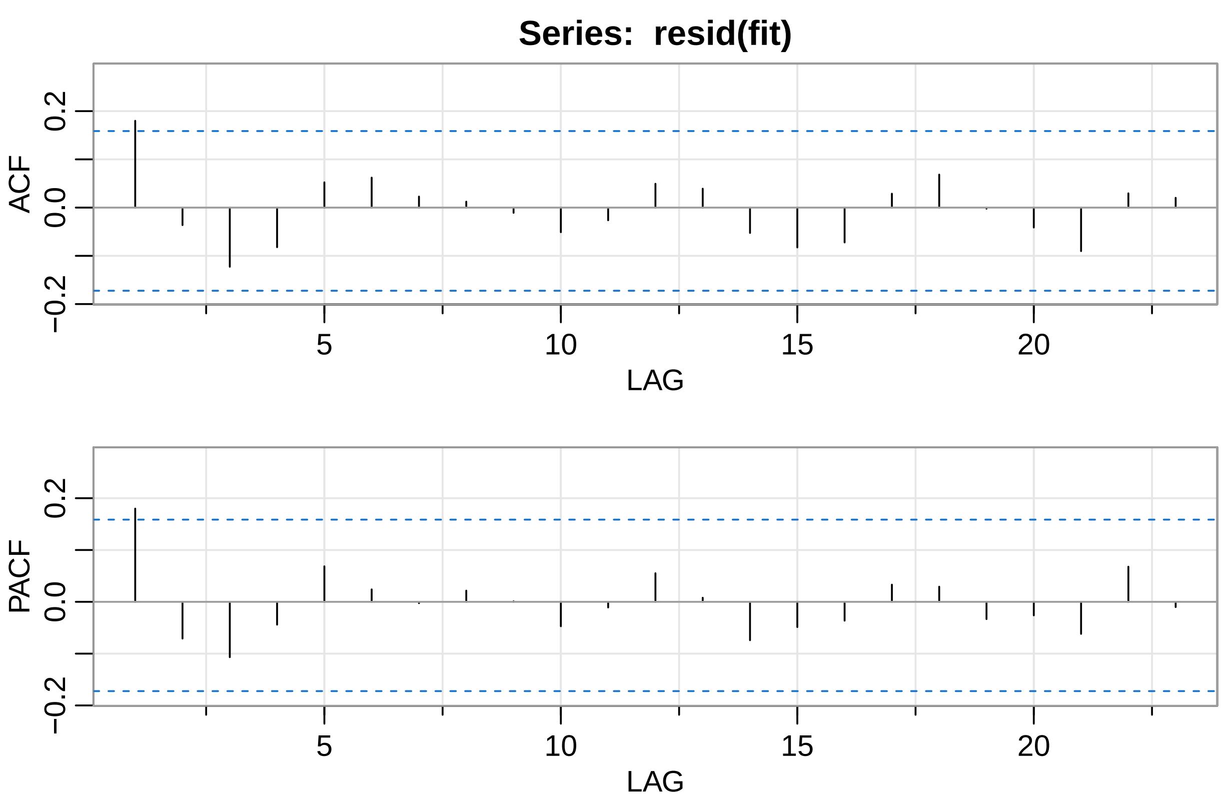
```
#Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
```

```
##      Min       1Q  Median       3Q      Max
## -3.5314 -0.9189 -0.0157  0.5586  8.5757
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 0.28536    0.20364   1.401 0.163405
## x`Avicultura de Corte`      0.41328    0.11349   3.642 0.000384 ***
## x`Avicultura de Postura`    0.04542    0.06035   0.753 0.452982
## xPescado                   -0.26037    0.11194  -2.326 0.021498 *
## xLácteos                   -0.20785    0.12322  -1.687 0.093939 .
## xSuinocultura               0.28048    0.21162   1.325 0.187266
## xavp9                       0.17980    0.05358   3.356 0.001026 **
## xp3                        -0.02202    0.10186  -0.216 0.829147
## xp10                        0.07166    0.10163   0.705 0.481954
## xb1                         0.37950    0.09758   3.889 0.000157 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.636 on 136 degrees of freedom
## Multiple R-squared:  0.4224, Adjusted R-squared:  0.3841
## F-statistic: 11.05 on 9 and 136 DF,  p-value: 8.134e-13
```

Análise dos Resíduos

```
#Análise dos Resíduos
acf2(resid(fit))
```



```
##      [,1]  [,2]  [,3]  [,4] [,5] [,6] [,7] [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.18 -0.04 -0.12 -0.08 0.05 0.06 0.02 0.01 -0.01 -0.05 -0.03  0.05  0.04
## PACF 0.18 -0.07 -0.11 -0.04 0.07 0.02 0.00 0.02  0.00 -0.05 -0.01  0.06  0.01
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.05 -0.08 -0.07  0.03  0.07  0.00 -0.04 -0.09  0.03  0.02
## PACF -0.07 -0.05 -0.04  0.03  0.03 -0.03 -0.03 -0.06  0.07 -0.01
```

Regressão com erros autocorrelacionais

Análise dos resíduos e seleção de variáveis de acordo com p-valor

```r
#Análise dos resíduos e seleção de variáveis de acordo com p-valor
fit2 <- tirar_variaveis(0, 0, 0, x, y)
fit2[[1]]
```

```
##
## z test of coefficients:
##
##                        Estimate Std. Error z value  Pr(>|z|)
## intercept              0.288306   0.153111  1.8830 0.0597017 .
## `Avicultura de Corte`  0.442471   0.105529  4.1929 2.754e-05 ***
## Pescado               -0.200620   0.097816 -2.0510 0.0402669 *
## avp9                   0.179147   0.052350  3.4221 0.0006214 ***
## b1                     0.442780   0.088166  5.0221 5.110e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx)
fit3
```

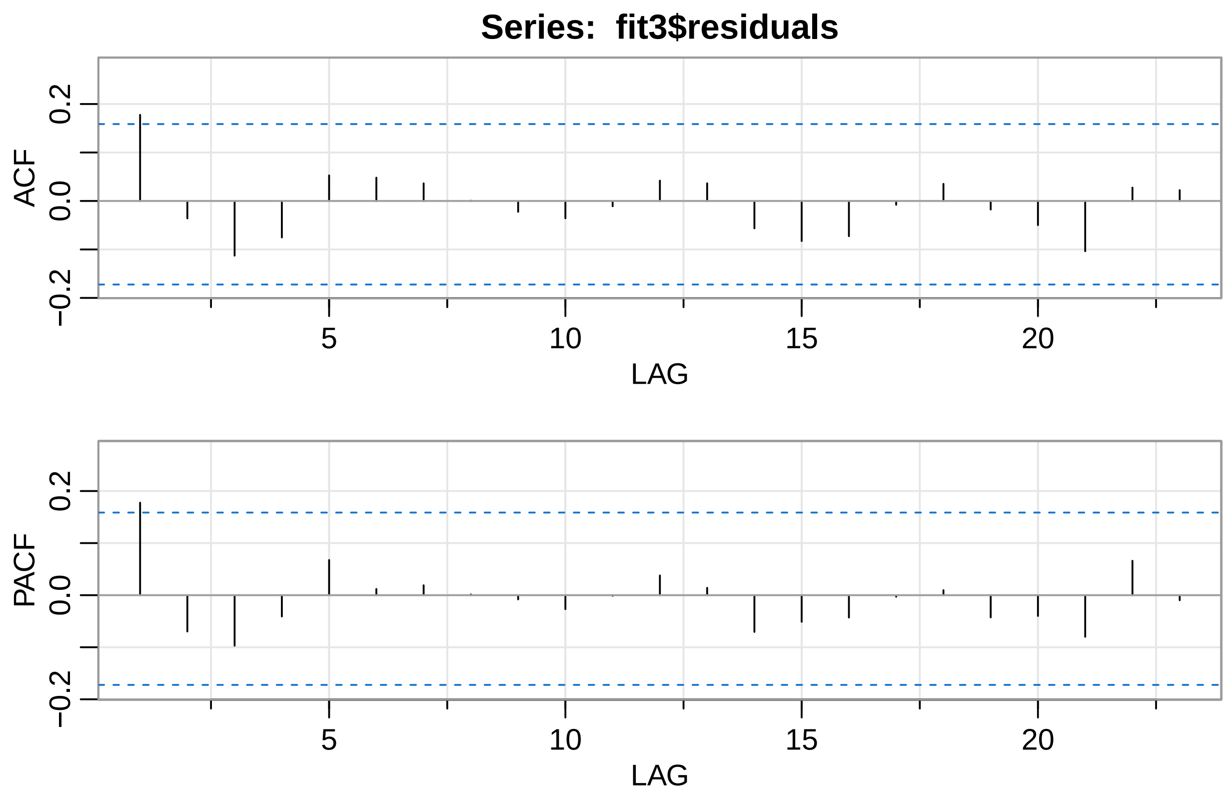
```
## Series: y
## Regression with ARIMA(0,0,0) errors
##
## Coefficients:
##       intercept  `Avicultura de Corte`  Pescado    avp9      b1
##          0.2883                 0.4425  -0.2006  0.1791  0.4428
## s.e.     0.1531                 0.1055   0.0978  0.0523  0.0882
##
## sigma^2 estimated as 2.693:  log likelihood=-276.95
## AIC=565.9   AICc=566.51   BIC=583.81
```

```r
coeftest(fit3)
```

```
##
## z test of coefficients:
##
```

```
##                          Estimate Std. Error z value  Pr(>|z|)
## intercept                0.288306   0.153111  1.8830 0.0597017 .
## `Avicultura de Corte`    0.442471   0.105529  4.1929 2.754e-05 ***
## Pescado                 -0.200620   0.097816 -2.0510 0.0402669 *
## avp9                     0.179147   0.052350  3.4221 0.0006214 ***
## b1                       0.442780   0.088166  5.0221 5.110e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
acf2(fit3$residuals)
```




```
##      [,1]  [,2]  [,3]  [,4] [,5] [,6] [,7] [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.18 -0.04 -0.11 -0.08 0.05 0.05 0.04    0 -0.02 -0.04 -0.01  0.04  0.04
## PACF 0.18 -0.07 -0.10 -0.04 0.07 0.01 0.02    0 -0.01 -0.03  0.00  0.04  0.01
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.06 -0.08 -0.07 -0.01  0.04 -0.02 -0.05 -0.10  0.03  0.02
## PACF -0.07 -0.05 -0.04  0.00  0.01 -0.04 -0.04 -0.08  0.07 -0.01
```

```
fit4 = Arima(y,order=c(1,0,0),xreg=xx)
fit4
```

```
## Series: y
## Regression with ARIMA(1,0,0) errors
##
```

```
## Coefficients:
##          ar1  intercept  ‘Avicultura de Corte‘  Pescado     avp9      b1
##       0.4823     0.5436                 0.5648  -0.1257  0.1491  0.1027
## s.e.  0.1250     0.2797                 0.1101   0.0892  0.0492  0.1226
##
## sigma^2 estimated as 2.485:  log likelihood=-270.68
## AIC=555.36   AICc=556.17   BIC=576.24
```

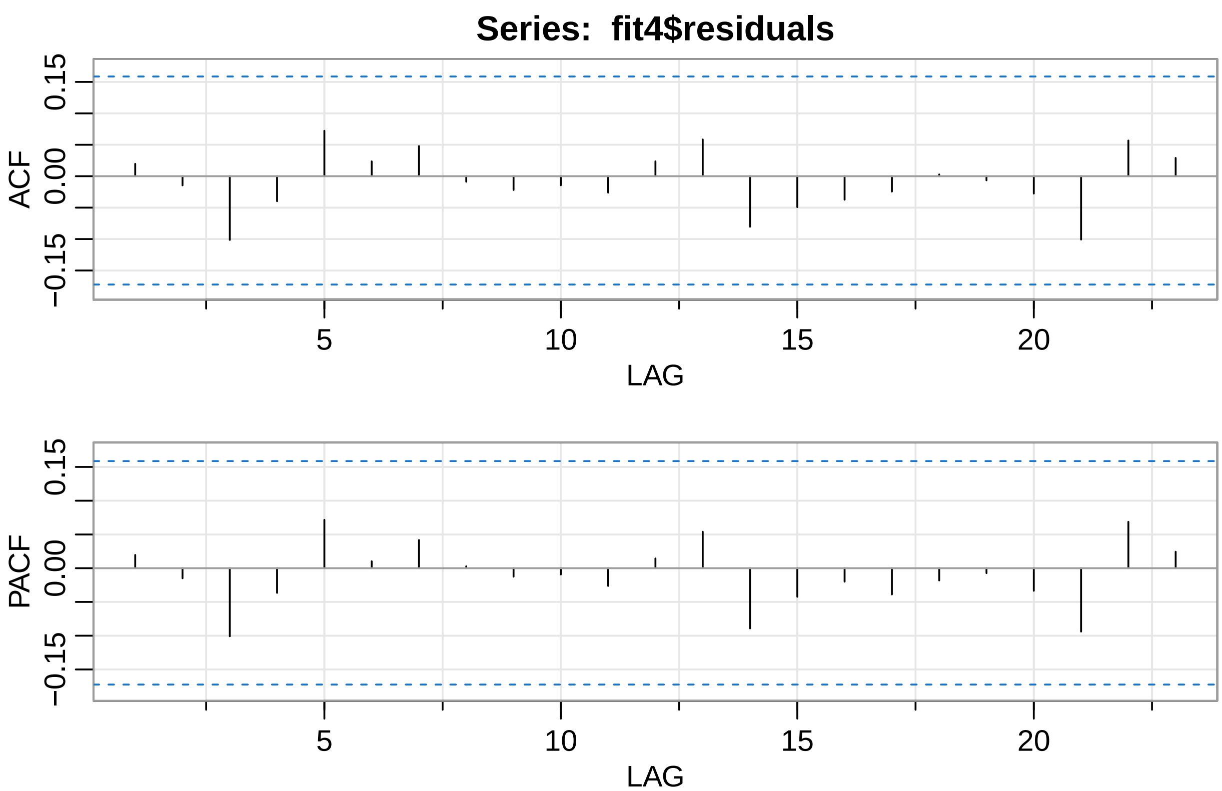
```r
coeftest(fit4)
```

```
##
## z test of coefficients:
##
##                         Estimate Std. Error z value  Pr(>|z|)
## ar1                     0.482252   0.124956  3.8594 0.0001137 ***
## intercept               0.543622   0.279737  1.9433 0.0519760 .
## ‘Avicultura de Corte‘  0.564759   0.110114  5.1289 2.915e-07 ***
## Pescado                -0.125731   0.089156 -1.4102 0.1584680
## avp9                    0.149073   0.049215  3.0290 0.0024534 **
## b1                      0.102674   0.122567  0.8377 0.4022023
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
checkresiduals(fit4)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(1,0,0) errors
## Q* = 3.2481, df = 4, p-value = 0.5172
##
## Model df: 6.   Total lags used: 10
```

```
acf2(fit4$residuals)
```




```
##      [,1]  [,2] [,3]  [,4] [,5] [,6] [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.02 -0.01 -0.1 -0.04 0.07 0.02 0.05 -0.01 -0.02 -0.01 -0.03  0.02  0.06
## PACF 0.02 -0.01 -0.1 -0.04 0.07 0.01 0.04  0.00 -0.01 -0.01 -0.03  0.01  0.05
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.08 -0.05 -0.04 -0.02  0.00 -0.01 -0.03 -0.10  0.06  0.03
## PACF -0.09 -0.04 -0.02 -0.04 -0.02 -0.01 -0.03 -0.09  0.07  0.02
```

```
fit5 <- tirar_variaveis(1, 0, 0, xx, y)
fit5[[1]]
```

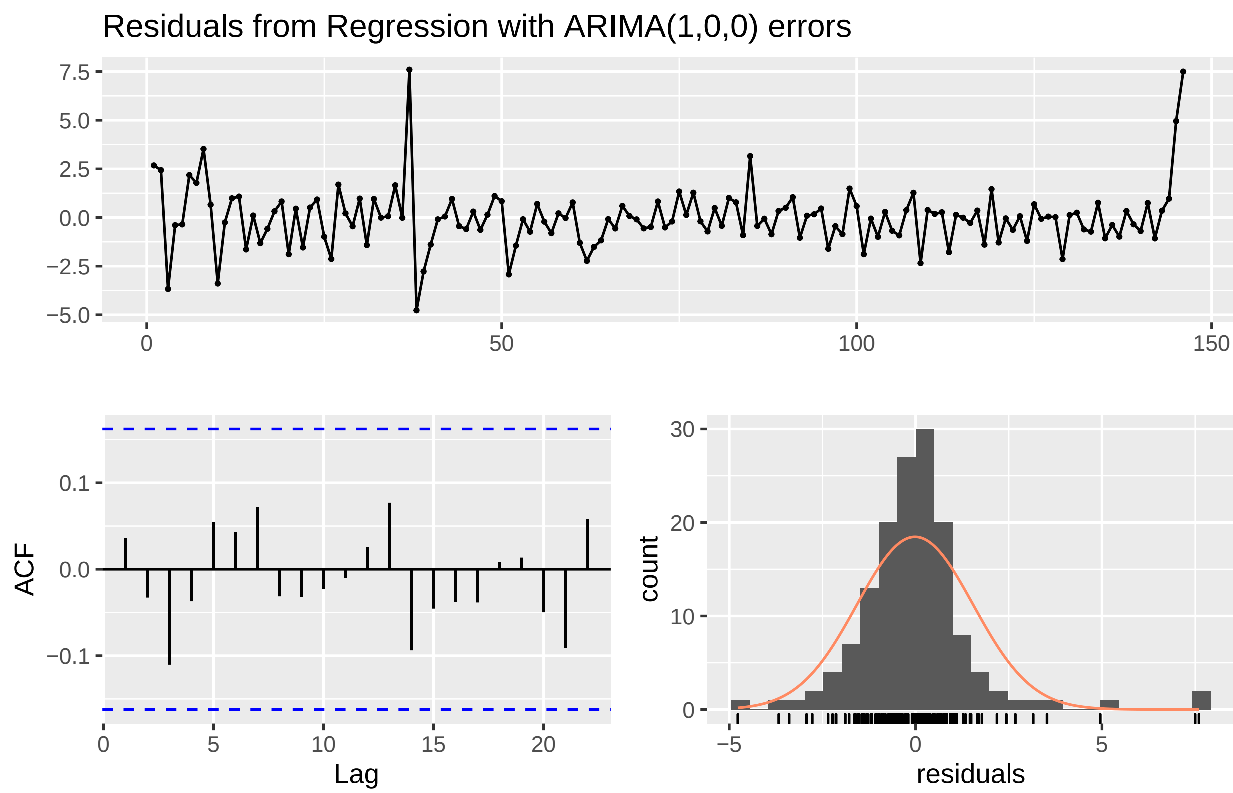
```
##
## z test of coefficients:
##
##                         Estimate Std. Error z value  Pr(>|z|)
```

```
## ar1                    0.547259   0.087657  6.2432 4.288e-10 ***
## intercept              0.584360   0.293969  1.9878   0.04683 *
## `Avicultura de Corte` 0.560426   0.106041  5.2850 1.257e-07 ***
## avp9                   0.132716   0.047193  2.8122   0.00492 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
xx <- fit5[2]
xx<- xx[[1]]



fit6 = Arima(y,order=c(1,0,0),xreg=xx,fixed=c(NA,NA, NA, NA))
cof.fit6 = coeftest(fit6)
cof.fit6
```

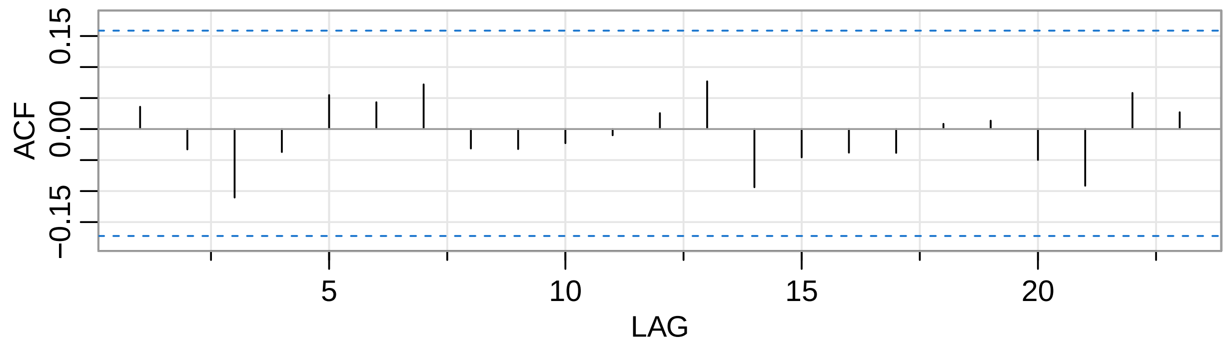
```
##
## z test of coefficients:
##
##                        Estimate Std. Error z value  Pr(>|z|)
## ar1                    0.547259   0.087657  6.2432 4.288e-10 ***
## intercept              0.584360   0.293969  1.9878   0.04683 *
## `Avicultura de Corte` 0.560426   0.106041  5.2850 1.257e-07 ***
## avp9                   0.132716   0.047193  2.8122   0.00492 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

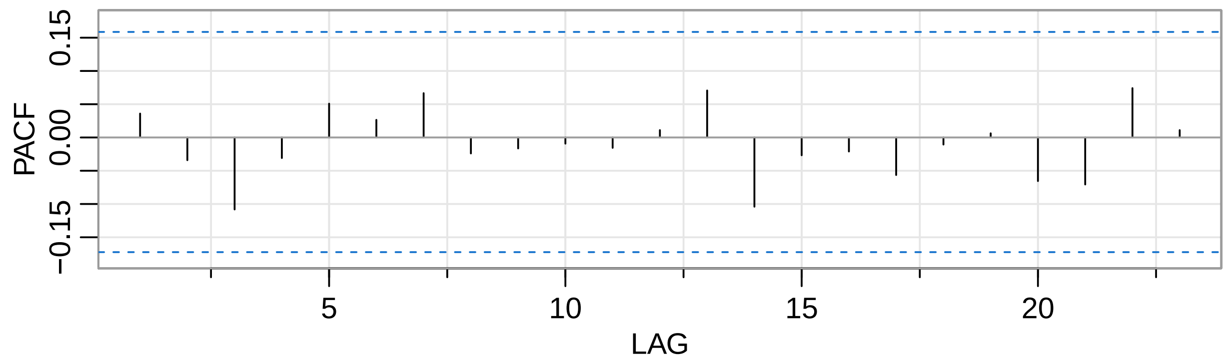
```
checkresiduals(fit6)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(1,0,0) errors
## Q* = 4.357, df = 6, p-value = 0.6285
##
## Model df: 4.   Total lags used: 10
```

```
acf2(fit6$residuals, main = "")
```






```
##      [,1]  [,2]  [,3]  [,4] [,5] [,6] [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.04 -0.03 -0.11 -0.04 0.05 0.04 0.07 -0.03 -0.03 -0.02 -0.01  0.03  0.08
## PACF 0.04 -0.03 -0.11 -0.03 0.05 0.03 0.07 -0.02 -0.02 -0.01 -0.02  0.01  0.07
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.09 -0.05 -0.04 -0.04  0.01  0.01 -0.05 -0.09  0.06  0.03
## PACF -0.10 -0.03 -0.02 -0.06 -0.01  0.01 -0.07 -0.07  0.07  0.01
```

Modelo da Avicultura de Corte

Estruturando a base

```
#Estruturando a base
df1<- funcao_lags(data_cut, data_cut$`Avicultura de Corte`, 'cort1', 1)
df1 <- funcao_lags(df1, df1$`Avicultura de Postura`, 'pos12', 12)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov1', 1)
df1 <- funcao_lags(df1, df1$Pescado, 'pes4', 4)
df1 <- funcao_lags(df1, df1$Pescado, 'pes9', 9)
df1 <- funcao_lags(df1, df1$Suinocultura, 'sui1', 1)
df1 <- funcao_lags(df1, df1$Suinocultura, 'sui6', 6)

df2 <- na.omit(df1)
```

Separando variável preditora e as covariáveis

```r
#Separando variável preditora e as covariáveis
x = model.matrix(`Avicultura de Corte`~.,df2)[,-1]
y = df2$`Avicultura de Corte`
```

Regressão classica no contexto de Séries Temporais

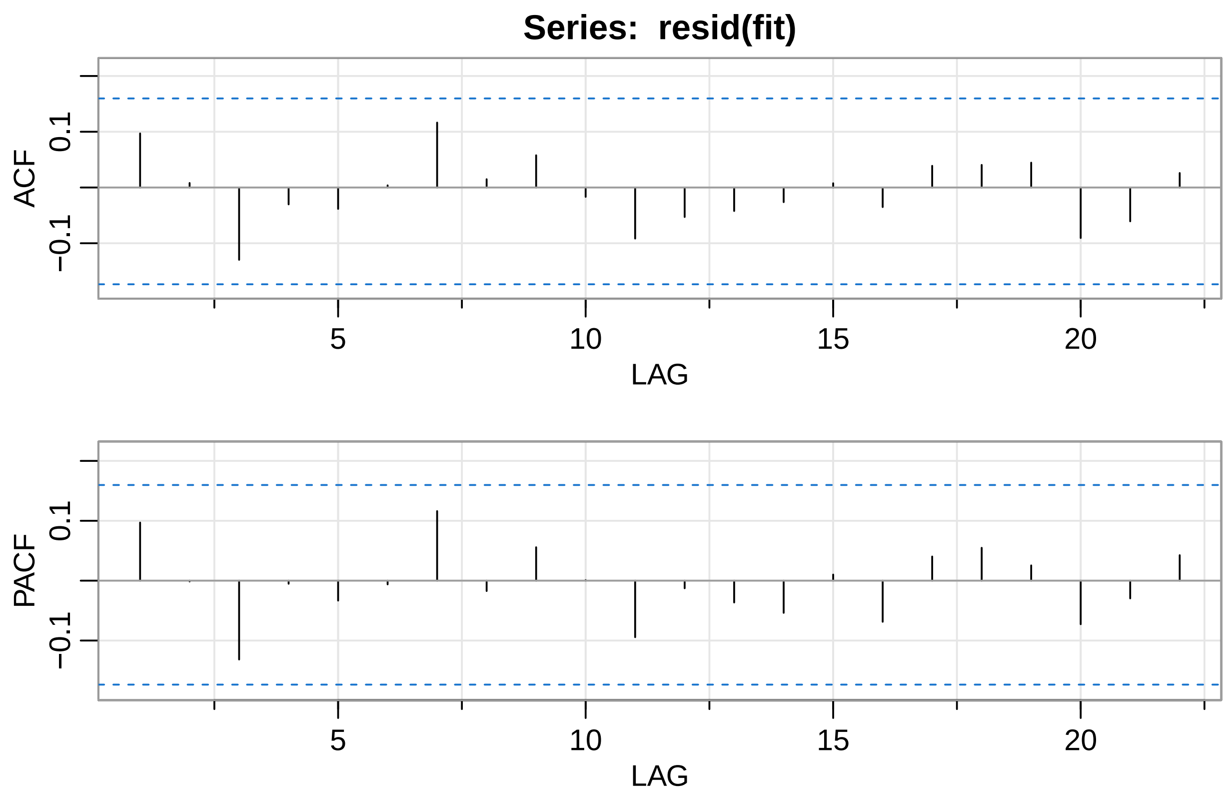
Criando o modelo de Regressão Simples

```r
#Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8583 -0.5435 -0.0324  0.5123  3.4823
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -0.01212    0.13955  -0.087 0.930894
## xBovinocultura            0.22080    0.05196   4.249 4.05e-05 ***
## x`Avicultura de Postura`  0.12991    0.03793   3.425 0.000822 ***
## xPescado                  0.07105    0.06226   1.141 0.255841
## xLácteos                  0.24253    0.07549   3.213 0.001655 **
## xSuinocultura             0.19667    0.13949   1.410 0.160939
## xcort1                    0.33941    0.07225   4.698 6.56e-06 ***
## xpos12                   -0.10059    0.03611  -2.785 0.006139 **
## xbov1                     0.07239    0.06584   1.099 0.273628
## xpes4                    -0.06147    0.06412  -0.959 0.339505
## xpes9                     0.15784    0.06198   2.547 0.012035 *
## xsui1                    -0.01789    0.13031  -0.137 0.891035
## xsui6                    -0.44583    0.11305  -3.944 0.000130 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9766 on 131 degrees of freedom
## Multiple R-squared:  0.5835, Adjusted R-squared:  0.5454
## F-statistic:  15.3 on 12 and 131 DF,  p-value: < 2.2e-16
```

Análise dos Resíduos

```r
#Análise dos Resíduos
acf2(resid(fit))
```




```
##       [,1] [,2]  [,3]  [,4]  [,5]  [,6] [,7]  [,8] [,9] [,10] [,11] [,12] [,13]
## ACF   0.1 0.01 -0.13 -0.03 -0.04  0.00 0.12  0.01 0.06 -0.02 -0.09 -0.05 -0.04
## PACF  0.1 0.00 -0.13 -0.01 -0.03 -0.01 0.12 -0.02 0.06  0.00 -0.09 -0.01 -0.04
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## ACF  -0.03  0.01 -0.04  0.04  0.04  0.04 -0.09 -0.06  0.03
## PACF -0.05  0.01 -0.07  0.04  0.05  0.03 -0.07 -0.03  0.04
```

Seleção de variáveis

```
#Seleção de variáveis
fit2 <- tirar_variaveis(0, 0, 0, x, y)
xx <- fit2[2]
xx <- xx[[1]]

fit3 = Arima(y,order=c(0,0,0), include.mean = FALSE, xreg=xx)
fit3
```

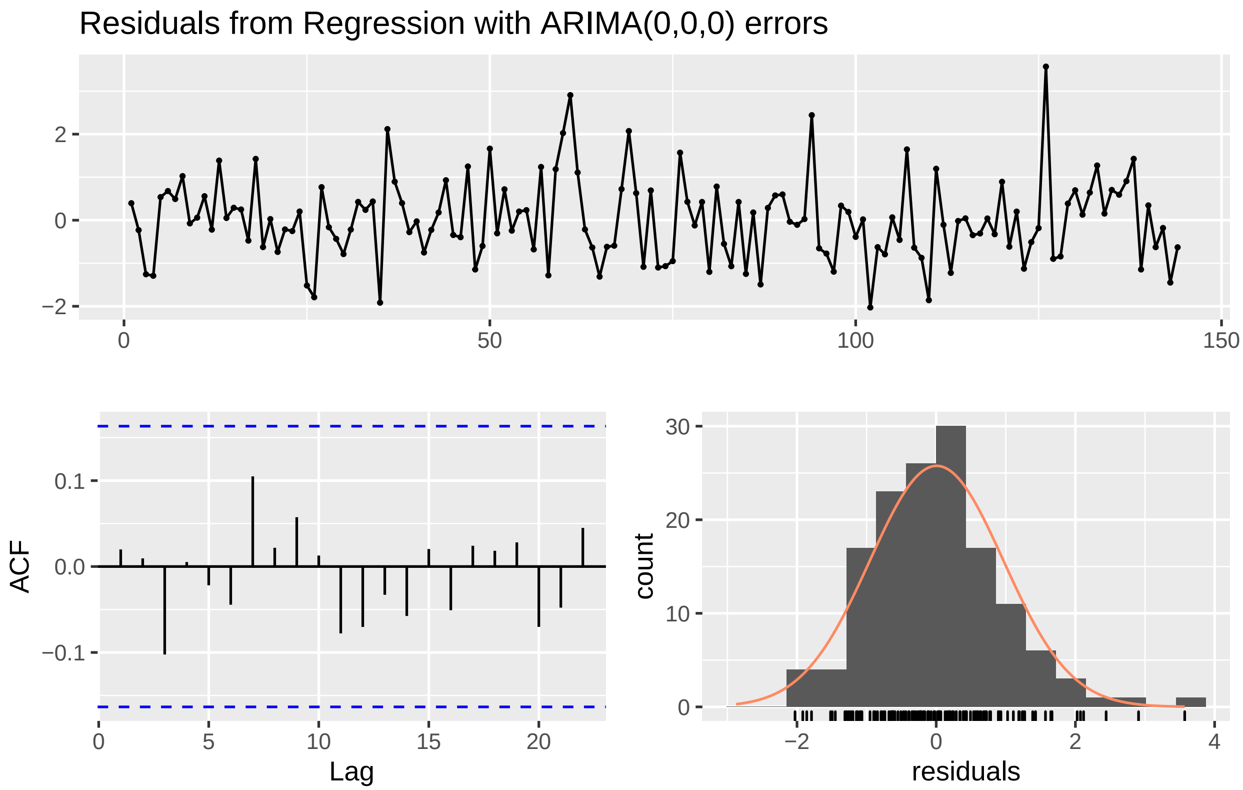
```
## Series: y
## Regression with ARIMA(0,0,0) errors
##
## Coefficients:
##       Bovinocultura  `Avicultura de Postura`  Lácteos   cort1    pos12   pes9
##              0.2870                   0.1343   0.2003  0.4368  -0.0781  0.195
## s.e.         0.0401                   0.0350   0.0681  0.0561   0.0329  0.053
##          sui6
##       -0.4269
```

```
## s.e.   0.0992
##
## sigma^2 estimated as 0.9627:  log likelihood=-198
## AIC=412   AICc=413.07   BIC=435.76
```

```r
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## Bovinocultura            0.286983   0.040122  7.1527 8.509e-13 ***
## `Avicultura de Postura`  0.134314   0.035007  3.8368 0.0001247 ***
## Lácteos                  0.200323   0.068070  2.9429 0.0032516 **
## cort1                    0.436756   0.056084  7.7875 6.833e-15 ***
## pos12                   -0.078065   0.032944 -2.3696 0.0178072 *
## pes9                     0.195018   0.053005  3.6793 0.0002339 ***
## sui6                    -0.426896   0.099200 -4.3034 1.682e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

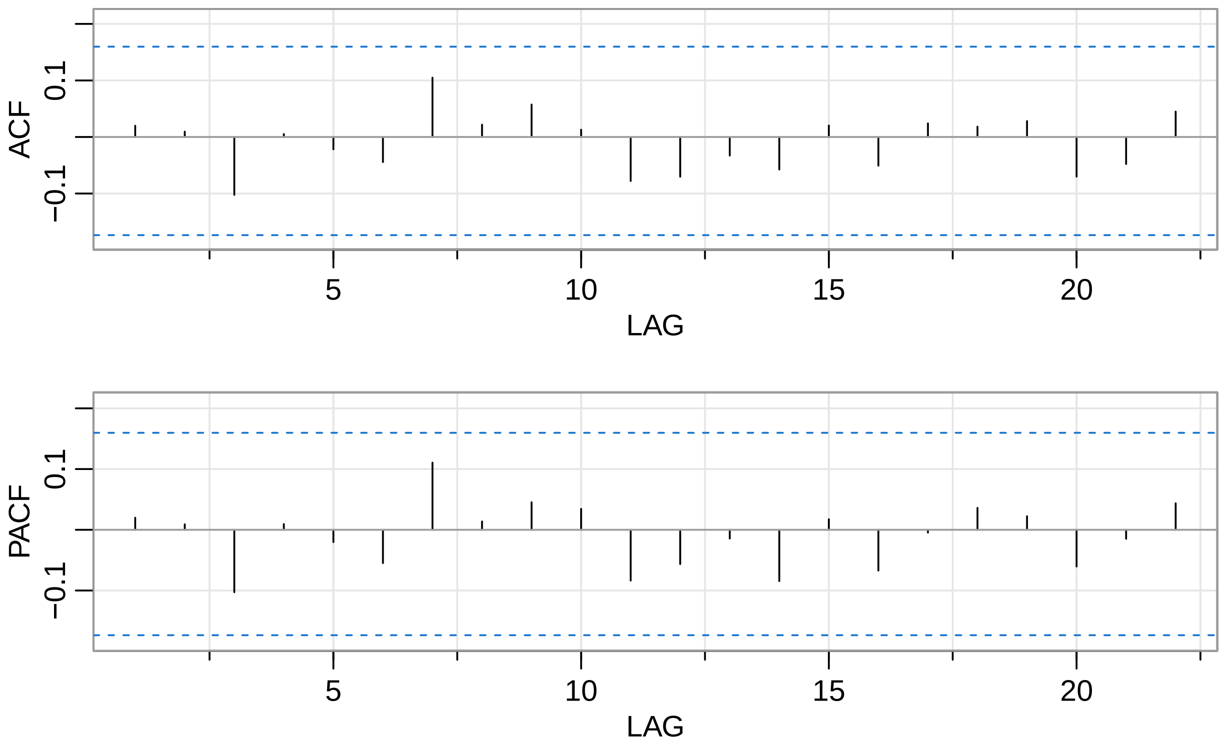
```r
checkresiduals(fit3)
```




```
##
```

```
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 4.3146, df = 3, p-value = 0.2294
##
## Model df: 7.   Total lags used: 10
```

```
acf2(fit3$residuals, main = "")
```




```
##      [,1] [,2] [,3] [,4]  [,5]  [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.02 0.01 -0.1 0.01 -0.02 -0.04 0.10 0.02 0.06  0.01 -0.08 -0.07 -0.03
## PACF 0.02 0.01 -0.1 0.01 -0.02 -0.05 0.11 0.01 0.05  0.03 -0.08 -0.06 -0.01
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## ACF  -0.06  0.02 -0.05  0.02  0.02  0.03 -0.07 -0.05  0.04
## PACF -0.08  0.02 -0.07  0.00  0.04  0.02 -0.06 -0.01  0.04
```

Modelo da Pescados

Estruturando a base

```
# Estruturando a base
df1<- funcao_lags(data_cut, data_cut$Pescado, 'pes1', 1)
df1 <- funcao_lags(df1, df1$Pescado, 'pes5', 5)
df1 <- funcao_lags(df1, df1$Pescado, 'pes12', 12)
```

```r
df1 <- funcao_lags(df1, df1$`Avicultura de Corte`, 'cort3', 3)
df1 <- funcao_lags(df1, df1$`Avicultura de Corte`, 'cort8', 8)


df1 <- funcao_lags(df1, df1$`Avicultura de Postura`, 'pos2', 2)
df1 <- funcao_lags(df1, df1$`Avicultura de Postura`, 'pos9', 9)


df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov1', 1)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov3', 3)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov7', 7)


df1 <- funcao_lags(df1, df1$Lácteos, 'lact2', 2)
df1 <- funcao_lags(df1, df1$Lácteos, 'lact8', 8)


df1 <- funcao_lags(df1, df1$Suinocultura, 'sui3', 3)



df2 <- na.omit(df1)
```

Separando variável preditora e as covariáveis

```r
#Separando variável preditora e as covariáveis
x = model.matrix(Pescado~.,df2)[,-1]
y = df2$Pescado
```

Regressão classifica no contexto de Séries Temporais

Criando o modelo de Regressão Simples

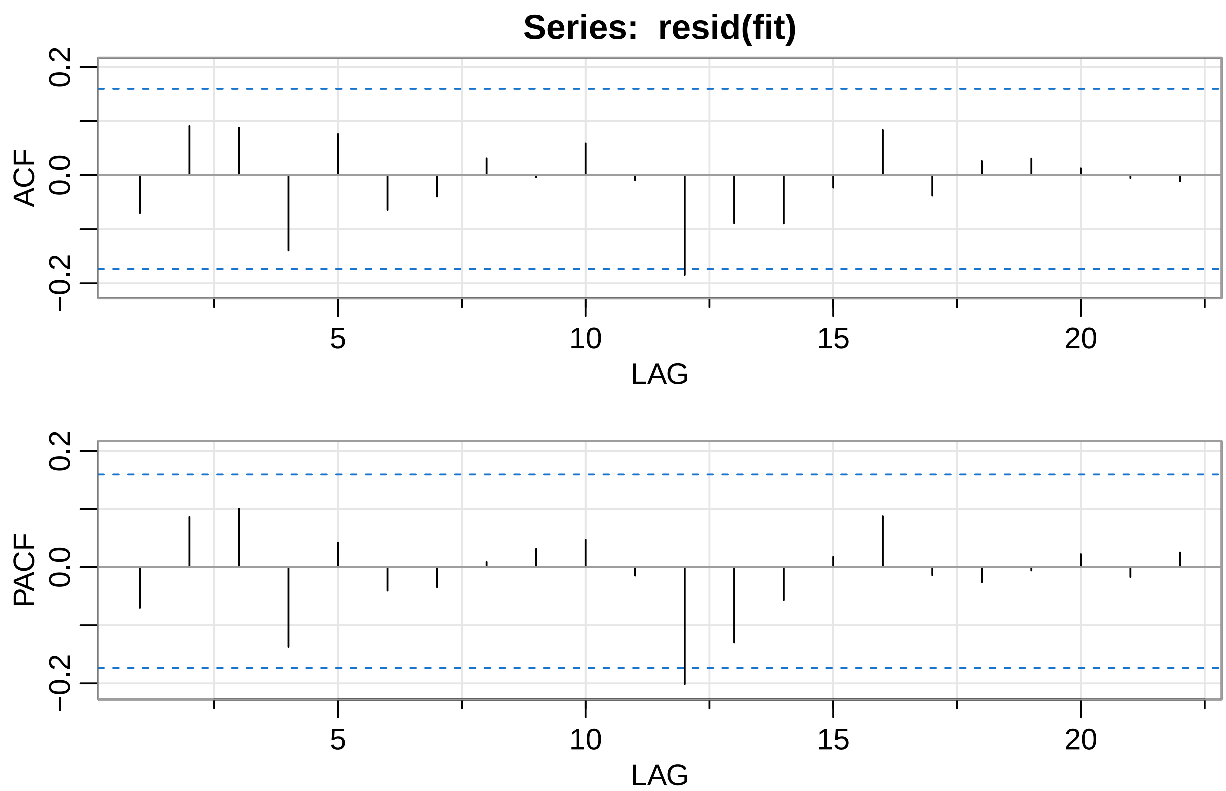
```r
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8416 -0.7307 -0.0757  0.6792  3.1091
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.19703    0.16044   1.228 0.221743
## xBovinocultura           -0.03365    0.06031  -0.558 0.577890
## x`Avicultura de Corte`    0.01262    0.08765   0.144 0.885759
```

```
## x`Avicultura de Postura`  0.14380    0.04089   3.517 0.000609 ***
## xLácteos               -0.11286    0.09287  -1.215 0.226550
## xSuinocultura           0.11153    0.14706   0.758 0.449621
## xpes1                  -0.03033    0.06781  -0.447 0.655479
## xpes5                  -0.08437    0.07483  -1.128 0.261683
## xpes12                  0.30991    0.07164   4.326 3.08e-05 ***
## xcort3                  0.07747    0.07984   0.970 0.333785
## xcort8                 -0.14097    0.07260  -1.942 0.054438 .
## xpos2                  -0.03105    0.04007  -0.775 0.439934
## xpos9                   0.09715    0.03980   2.441 0.016048 *
## xbov1                   0.16976    0.07257   2.339 0.020910 *
## xbov3                  -0.05010    0.06816  -0.735 0.463668
## xbov7                  -0.12174    0.05980  -2.036 0.043905 *
## xlact2                  0.02538    0.09274   0.274 0.784790
## xlact8                  0.05811    0.08427   0.690 0.491745
## xsui3                   0.37341    0.15003   2.489 0.014128 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.098 on 125 degrees of freedom
## Multiple R-squared:  0.5513, Adjusted R-squared:  0.4867
## F-statistic: 8.534 on 18 and 125 DF,  p-value: 1.673e-14
```

Análise dos Resíduos

```
# Análise dos Resíduos
acf2(resid(fit))
```



```
##       [,1] [,2] [,3]  [,4] [,5]  [,6]  [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.07 0.09 0.09 -0.14 0.08 -0.06 -0.04 0.03 0.00  0.06 -0.01 -0.18 -0.09
## PACF -0.07 0.09 0.10 -0.14 0.04 -0.04 -0.03 0.01 0.03  0.05 -0.01 -0.20 -0.13
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## ACF  -0.09 -0.02  0.08 -0.04  0.03  0.03  0.01 -0.01 -0.01
## PACF -0.06  0.02  0.09 -0.01 -0.03 -0.01  0.02 -0.02  0.03
```

Regressão com erros autocorrelacionais

Análise dos resíduos e seleção de variáveis de acordo com p-valor

```
# Análise dos resíduos e seleção de variáveis de acordo com p-valor
y = ts(y, frequency=12)

x = x[,-1]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                    -0.386944   0.091562 -4.2261 2.378e-05 ***
## intercept                0.080072   0.119976  0.6674 0.5045201
## `Avicultura de Corte`    0.021593   0.070335  0.3070 0.7588441
```

```
## `Avicultura de Postura`  0.116637   0.033488  3.4830 0.0004959 ***
## Lácteos                -0.062973   0.080753 -0.7798 0.4354975
## Suinocultura            0.155301   0.120650  1.2872 0.1980220
## pes1                   -0.037597   0.052277 -0.7192 0.4720305
## pes5                   -0.095328   0.059900 -1.5915 0.1115067
## pes12                   0.507059   0.069068  7.3415 2.112e-13 ***
## cort3                   0.077964   0.068351  1.1406 0.2540252
## cort8                  -0.099856   0.067674 -1.4755 0.1400699
## pos2                   -0.012863   0.032304 -0.3982 0.6905015
## pos9                    0.065823   0.031207  2.1092 0.0349240 *
## bov1                    0.095679   0.063358  1.5101 0.1310108
## bov3                   -0.025267   0.056244 -0.4492 0.6532543
## bov7                   -0.103807   0.050848 -2.0415 0.0411989 *
## lact2                   0.003111   0.079860  0.0390 0.9689257
## lact8                   0.033576   0.077945  0.4308 0.6666430
## sui3                    0.345969   0.120310  2.8757 0.0040319 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-15]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                   -0.387179   0.091333 -4.2392 2.243e-05 ***
## intercept               0.081189   0.116486  0.6970 0.4858110
## `Avicultura de Corte`    0.021848   0.070023  0.3120 0.7550344
## `Avicultura de Postura`  0.116206   0.031601  3.6772 0.0002358 ***
## Lácteos                -0.061895   0.075859 -0.8159 0.4145439
## Suinocultura            0.155191   0.120610  1.2867 0.1981900
## pes1                   -0.038027   0.051090 -0.7443 0.4566819
## pes5                   -0.095203   0.059810 -1.5917 0.1114423
## pes12                   0.507119   0.069034  7.3459 2.044e-13 ***
## cort3                   0.077986   0.068345  1.1411 0.2538471
## cort8                  -0.099416   0.066727 -1.4899 0.1362542
## pos2                   -0.012825   0.032287 -0.3972 0.6912055
## pos9                    0.065767   0.031172  2.1098 0.0348757 *
## bov1                    0.095336   0.062735  1.5197 0.1285958
## bov3                   -0.025241   0.056236 -0.4488 0.6535506
## bov7                   -0.103724   0.050801 -2.0418 0.0411723 *
## lact8                   0.033821   0.077687  0.4354 0.6633040
## sui3                    0.346308   0.119977  2.8865 0.0038960 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-1]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
```

```
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                    -0.384320   0.091107 -4.2184 2.461e-05 ***
## intercept                0.087499   0.114897  0.7615 0.4463318
## `Avicultura de Postura`  0.117141   0.031538  3.7142 0.0002038 ***
## Lácteos                 -0.058925   0.075323 -0.7823 0.4340391
## Suinocultura             0.160818   0.119384  1.3471 0.1779589
## pes1                    -0.040538   0.050535 -0.8022 0.4224562
## pes5                    -0.096844   0.059644 -1.6237 0.1044420
## pes12                    0.505835   0.069119  7.3183 2.511e-13 ***
## cort3                    0.074506   0.067497  1.1038 0.2696650
## cort8                   -0.100503   0.066625 -1.5085 0.1314272
## pos2                    -0.013420   0.032278 -0.4158 0.6775852
## pos9                     0.067293   0.030822  2.1833 0.0290167 *
## bov1                     0.103928   0.056423  1.8419 0.0654843 .
## bov3                    -0.025070   0.056307 -0.4452 0.6561505
## bov7                    -0.109781   0.046989 -2.3363 0.0194757 *
## lact8                    0.029144   0.076293  0.3820 0.7024574
## sui3                     0.356336   0.115786  3.0775 0.0020872 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-14]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                    -0.388147   0.090164 -4.3049 1.671e-05 ***
## intercept                0.095826   0.112606  0.8510  0.394777
## `Avicultura de Postura`  0.118704   0.031256  3.7978  0.000146 ***
## Lácteos                 -0.058338   0.075318 -0.7746  0.438605
## Suinocultura             0.157438   0.118989  1.3231  0.185792
## pes1                    -0.036520   0.049362 -0.7398  0.459399
## pes5                    -0.097461   0.059586 -1.6356  0.101919
## pes12                    0.513113   0.066348  7.7336 1.045e-14 ***
## cort3                    0.073818   0.067452  1.0944  0.273796
## cort8                   -0.096605   0.065857 -1.4669  0.142403
## pos2                    -0.011675   0.031917 -0.3658  0.714510
## pos9                     0.070328   0.029759  2.3633  0.018114 *
## bov1                     0.103383   0.056372  1.8340  0.066660 .
## bov3                    -0.022418   0.055839 -0.4015  0.688067
## bov7                    -0.112367   0.046468 -2.4182  0.015598 *
## sui3                     0.351878   0.115074  3.0578  0.002229 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-9]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                    -0.390935   0.089362 -4.3747 1.216e-05 ***
## intercept                0.096283   0.112406  0.8566  0.391686
## `Avicultura de Postura`  0.120786   0.030699  3.9345 8.338e-05 ***
## Lácteos                 -0.063597   0.073916 -0.8604  0.389575
## Suinocultura             0.154805   0.118734  1.3038  0.192303
## pes1                    -0.040696   0.047980 -0.8482  0.396326
## pes5                    -0.104471   0.056397 -1.8524  0.063964 .
## pes12                    0.517458   0.064975  7.9639 1.667e-15 ***
## cort3                    0.072869   0.067380  1.0815  0.279492
## cort8                   -0.097380   0.065850 -1.4788  0.139192
## pos9                     0.070694   0.029732  2.3777  0.017421 *
## bov1                     0.104262   0.056282  1.8525  0.063952 .
## bov3                    -0.021411   0.055754 -0.3840  0.700960
## bov7                    -0.112548   0.046456 -2.4227  0.015406 *
## sui3                     0.349353   0.114804  3.0430  0.002342 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
x = x[,-11]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                    -0.392673   0.088904 -4.4168 1.002e-05 ***
## intercept                0.092348   0.111873  0.8255  0.409104
## `Avicultura de Postura`  0.117943   0.029764  3.9625 7.415e-05 ***
## Lácteos                 -0.060075   0.073328 -0.8193  0.412638
## Suinocultura             0.156577   0.118658  1.3196  0.186979
## pes1                    -0.042151   0.047810 -0.8816  0.377972
## pes5                    -0.102738   0.056211 -1.8277  0.067592 .
## pes12                    0.519187   0.064688  8.0260 1.007e-15 ***
## cort3                    0.063206   0.062513  1.0111  0.311977
## cort8                   -0.098137   0.065817 -1.4911  0.135945
## pos9                     0.070944   0.029689  2.3895  0.016869 *
## bov1                     0.101801   0.055946  1.8196  0.068815 .
## bov7                    -0.112213   0.046453 -2.4156  0.015708 *
## sui3                     0.335663   0.109074  3.0774  0.002088 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
x = x[,-2]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
```

```
## z test of coefficients:
##
##                            Estimate Std. Error z value  Pr(>|z|)
## sar1                      -0.400868   0.087551 -4.5787 4.679e-06 ***
## intercept                  0.071346   0.108529  0.6574 0.5109328
## `Avicultura de Postura`    0.112130   0.028898  3.8802 0.0001044 ***
## Suinocultura               0.147507   0.118103  1.2490 0.2116763
## pes1                      -0.041784   0.047712 -0.8758 0.3811637
## pes5                      -0.110446   0.055348 -1.9955 0.0459902 *
## pes12                      0.526696   0.063928  8.2389 < 2.2e-16 ***
## cort3                      0.062889   0.062508  1.0061 0.3143661
## cort8                     -0.098307   0.065984 -1.4899 0.1362606
## pos9                       0.071948   0.029595  2.4311 0.0150537 *
## bov1                       0.106356   0.055664  1.9107 0.0560455 .
## bov7                      -0.119520   0.045542 -2.6244 0.0086805 **
## sui3                       0.334402   0.108969  3.0688 0.0021494 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-3]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                            Estimate Std. Error z value  Pr(>|z|)
## sar1                      -0.398561   0.088489 -4.5041 6.667e-06 ***
## intercept                  0.053521   0.106943  0.5005 0.6167476
## `Avicultura de Postura`    0.108768   0.028746  3.7837 0.0001545 ***
## Suinocultura               0.151192   0.118435  1.2766 0.2017491
## pes5                      -0.102299   0.054775 -1.8676 0.0618157 .
## pes12                      0.522676   0.064592  8.0920 5.869e-16 ***
## cort3                      0.059915   0.062636  0.9566 0.3387898
## cort8                     -0.095019   0.066096 -1.4376 0.1505515
## pos9                       0.071221   0.029727  2.3958 0.0165836 *
## bov1                       0.102061   0.055734  1.8312 0.0670717 .
## bov7                      -0.120174   0.045699 -2.6297 0.0085467 **
## sui3                       0.317169   0.107541  2.9493 0.0031851 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-5]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

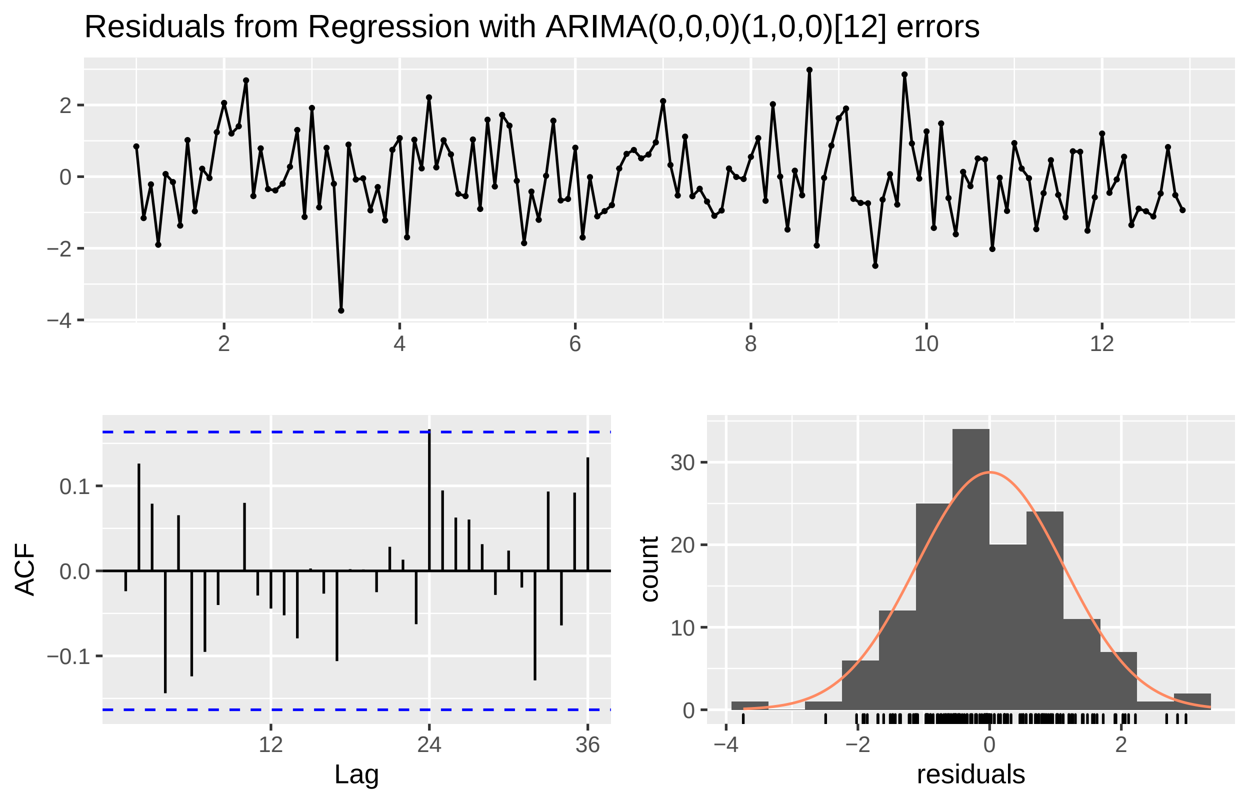
```
##
## z test of coefficients:
##
##                            Estimate Std. Error z value  Pr(>|z|)
## sar1                      -0.398172   0.088518 -4.4982 6.854e-06 ***
## intercept                  0.060722   0.107058  0.5672 0.5705875
```

```
## `Avicultura de Postura`  0.112567   0.028586  3.9378 8.223e-05 ***
## Suinocultura             0.163490   0.118145  1.3838 0.1664180
## pes5                    -0.106665   0.054751 -1.9482 0.0513943 .
## pes12                    0.535268   0.063461  8.4345 < 2.2e-16 ***
## cort8                   -0.102429   0.065732 -1.5583 0.1191654
## pos9                     0.072864   0.029770  2.4475 0.0143834 *
## bov1                     0.091020   0.054774  1.6617 0.0965686 .
## bov7                    -0.119608   0.045864 -2.6079 0.0091094 **
## sui3                     0.348003   0.103094  3.3756 0.0007366 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
x = x[,-4]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

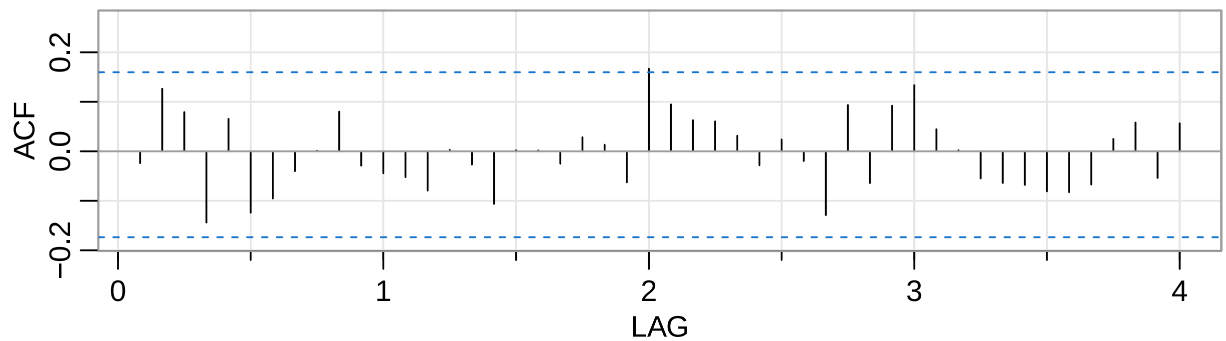
```
##
## z test of coefficients:
##
##                           Estimate Std. Error z value  Pr(>|z|)
## sar1                     0.2469874  0.1061663  2.3264 0.0199962 *
## intercept                0.4025207  0.1625312  2.4766 0.0132650 *
## `Avicultura de Postura`  0.1477883  0.0400297  3.6920 0.0002225 ***
## Suinocultura            -0.0079504  0.1475616 -0.0539 0.9570319
## pes5                    -0.1227962  0.0783012 -1.5683 0.1168216
## cort8                   -0.1584009  0.0703558 -2.2514 0.0243585 *
## pos9                     0.1308314  0.0400853  3.2638 0.0010992 **
## bov1                     0.1896427  0.0636054  2.9816 0.0028679 **
## bov7                    -0.1488028  0.0572052 -2.6012 0.0092895 **
## sui3                     0.3490096  0.1297387  2.6901 0.0071431 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

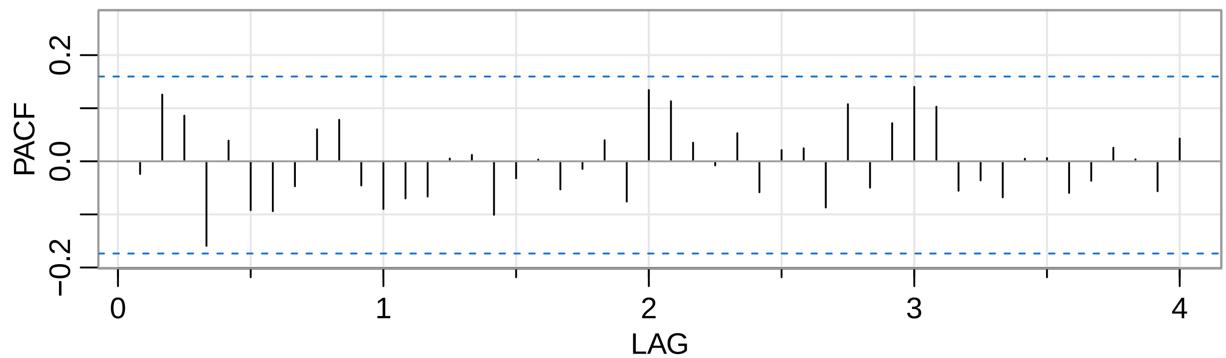
```r
checkresiduals(fit3)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0)(1,0,0)[12] errors
## Q* = 21.828, df = 14, p-value = 0.08222
##
## Model df: 10.   Total lags used: 24
```

```
acf2(fit3$residuals, main = "")
```






```
##       [,1] [,2] [,3]  [,4] [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  -0.02 0.13 0.08 -0.14 0.07 -0.12 -0.10 -0.04 0.00  0.08 -0.03 -0.04 -0.05
## PACF -0.02 0.13 0.09 -0.16 0.04 -0.09 -0.09 -0.05 0.06  0.08 -0.05 -0.09 -0.07
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.08  0.00 -0.03 -0.11  0.00     0 -0.03  0.03  0.01 -0.06  0.17  0.09
## PACF -0.07  0.01  0.01 -0.10 -0.03     0 -0.05 -0.01  0.04 -0.08  0.13  0.11
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF   0.06  0.06  0.03 -0.03  0.02 -0.02 -0.13  0.09 -0.06  0.09  0.13  0.04
## PACF  0.04 -0.01  0.05 -0.06  0.02  0.02 -0.09  0.11 -0.05  0.07  0.14  0.10
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.00 -0.05 -0.06 -0.07 -0.08 -0.08 -0.07  0.02  0.06 -0.05  0.06
## PACF -0.06 -0.04 -0.07  0.01  0.01 -0.06 -0.04  0.03  0.00 -0.06  0.04
```

```
x = x[,-2]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## sar1                     0.245630   0.103130  2.3818 0.0172305 *
## intercept                0.400724   0.158954  2.5210 0.0117018 *
## `Avicultura de Postura`  0.147907   0.039969  3.7005 0.0002151 ***
## pes5                    -0.123470   0.077226 -1.5988 0.1098647
## cort8                   -0.158979   0.069562 -2.2854 0.0222871 *
```

```
## pos9                        0.130901   0.040054  3.2681 0.0010827 **
## bov1                        0.187958   0.055461  3.3890 0.0007014 ***
## bov7                       -0.148780   0.057202 -2.6010 0.0092965 **
## sui3                        0.348588   0.129523  2.6913 0.0071169 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

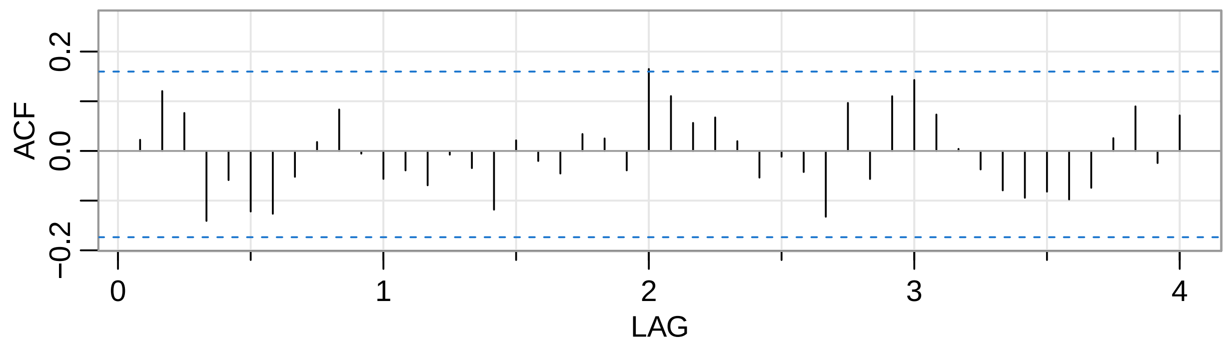
```
x = x[,-2]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

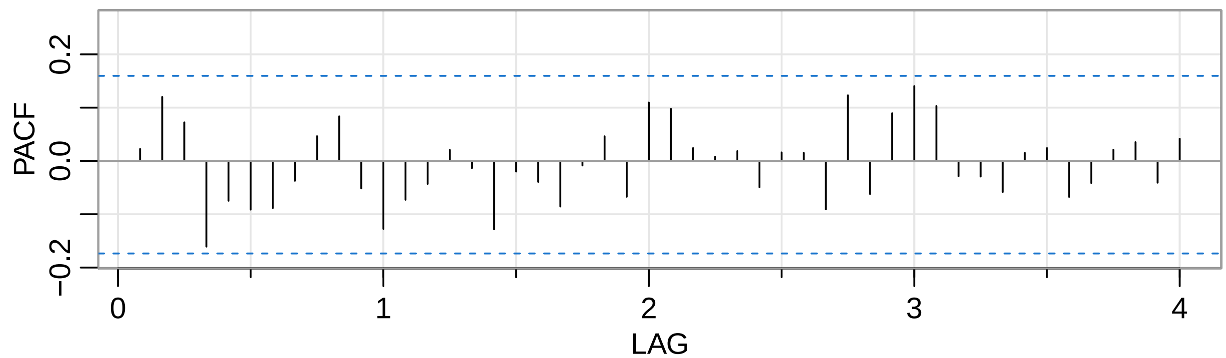
```
##
## z test of coefficients:
##
##                             Estimate Std. Error z value  Pr(>|z|)
## sar1                        0.309151   0.098927  3.1250 0.0017778 **
## intercept                   0.331766   0.161882  2.0494 0.0404197 *
## `Avicultura de Postura`  0.133601   0.039667  3.3681 0.0007570 ***
## cort8                      -0.161858   0.070924 -2.2821 0.0224811 *
## pos9                        0.135222   0.041512  3.2574 0.0011243 **
## bov1                        0.198338   0.055331  3.5845 0.0003377 ***
## bov7                       -0.152063   0.057899 -2.6264 0.0086302 **
## sui3                        0.327597   0.130070  2.5186 0.0117818 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
checkresiduals(fit3)
```


```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0)(1,0,0)[12] errors
## Q* = 22.744, df = 16, p-value = 0.1208
##
## Model df: 8.   Total lags used: 24
```

```
acf2(fit3$residuals, main = "")
```






```
##      [,1] [,2] [,3]  [,4]  [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.02 0.12 0.08 -0.14 -0.06 -0.12 -0.13 -0.05 0.02  0.08 -0.01 -0.06 -0.04
## PACF 0.02 0.12 0.07 -0.16 -0.07 -0.09 -0.09 -0.04 0.05  0.08 -0.05 -0.13 -0.07
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.07 -0.01 -0.03 -0.12  0.02 -0.02 -0.05  0.03  0.03 -0.04  0.16  0.11
## PACF -0.04  0.02 -0.01 -0.13 -0.02 -0.04 -0.09 -0.01  0.05 -0.07  0.11  0.10
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF   0.06  0.07  0.02 -0.05 -0.01 -0.04 -0.13  0.10 -0.06  0.11  0.14  0.07
## PACF  0.02  0.01  0.02 -0.05  0.02  0.02 -0.09  0.12 -0.06  0.09  0.14  0.10
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.00 -0.04 -0.08 -0.09 -0.08 -0.10 -0.07  0.03  0.09 -0.02  0.07
## PACF -0.03 -0.03 -0.06  0.02  0.02 -0.07 -0.04  0.02  0.04 -0.04  0.04
```

```
S
```

```
## function (object, brief, ...)
## {
##     UseMethod("S")
## }
## <bytecode: 0x00000000217c8498>
## <environment: namespace:car>
```

Modelo da Avicultura de postura

Estruturando a base

```r
# Estruturando a base
df1<- funcao_lags(data_cut, data_cut$`Avicultura de Postura`, 'avp1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Postura`, 'avp12', 12)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc5', 5)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov3', 3)
df1 <- funcao_lags(df1, df1$Lácteos, 'lact11', 11)
df1 <- funcao_lags(df1, df1$Pescado, 'pes2', 2)
df1 <- funcao_lags(df1, df1$Pescado, 'pes9', 9)


df2 <- na.omit(df1)
```

Separando variável preditora e as covariáveis

```r
#Separando variável preditora e as covariáveis
x = model.matrix(`Avicultura de Postura`~.,df2)[,-1]
y = df2$`Avicultura de Postura`
```

Regressão classifica no contexto de Séries Temporais

Criando o modelo de Regressão Simples

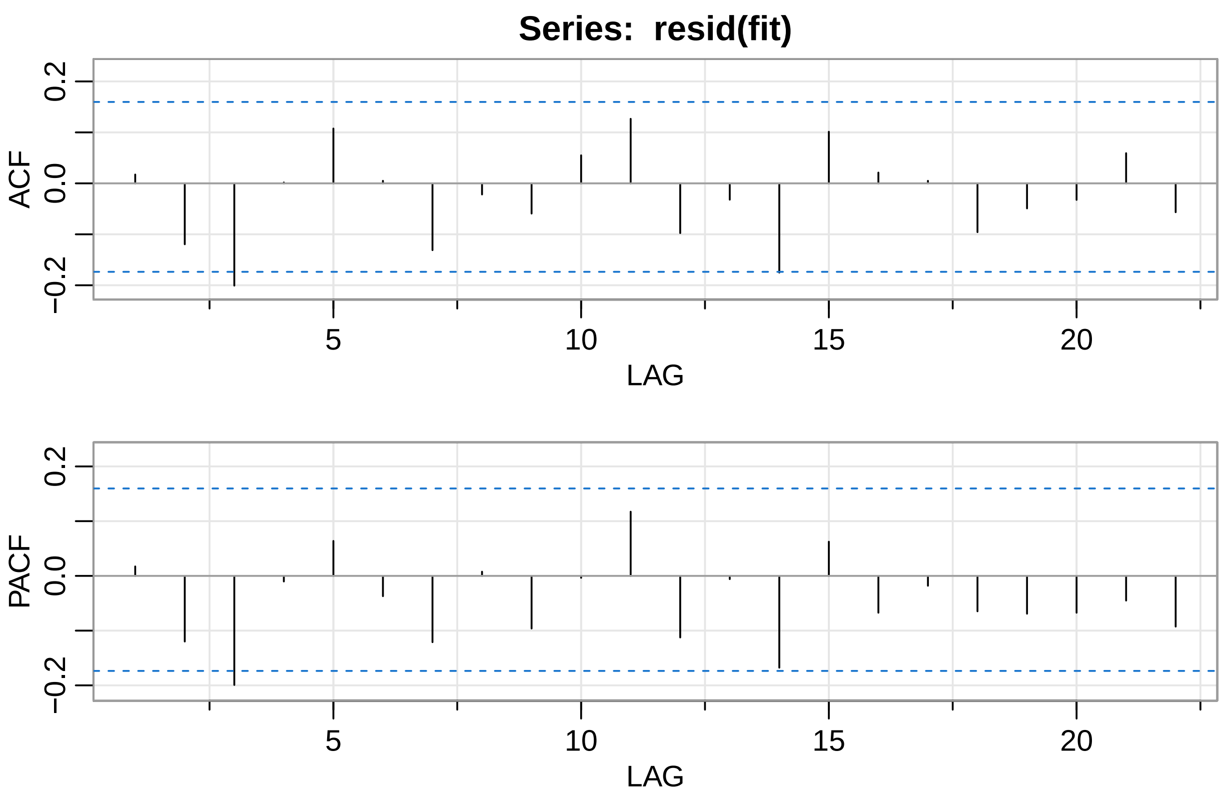
```r
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2151 -1.3755 -0.1872  1.4374  8.2788
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -0.01295    0.31733  -0.041 0.967509
## xBovinocultura         -0.01770    0.10989  -0.161 0.872253
## x`Avicultura de Corte`  0.45460    0.16365   2.778 0.006274 **
## xPescado                0.35251    0.13562   2.599 0.010415 *
## xLácteos                0.16496    0.17453   0.945 0.346327
## xSuinocultura          -0.28456    0.27621  -1.030 0.304802
## xavp1                   0.09421    0.07633   1.234 0.219312
## xavp12                  0.31398    0.08246   3.808 0.000215 ***
## xavc5                   0.31548    0.13972   2.258 0.025600 *
## xbov3                   0.07400    0.12114   0.611 0.542366
## xlact11                 0.11042    0.16012   0.690 0.491666
## xpes2                  -0.02661    0.14349  -0.185 0.853182
## xpes9                  -0.36608    0.14250  -2.569 0.011319 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2.178 on 131 degrees of freedom
## Multiple R-squared:  0.3902, Adjusted R-squared:  0.3343
## F-statistic: 6.984 on 12 and 131 DF,  p-value: 1.096e-09
```

Análise dos Resíduos

```
# Análise dos Resíduos
acf2(resid(fit))
```




```
##      [,1]  [,2] [,3]  [,4] [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.02 -0.12 -0.2  0.00 0.11  0.01 -0.13 -0.02 -0.06  0.05  0.13 -0.10 -0.03
## PACF 0.02 -0.12 -0.2 -0.01 0.06 -0.04 -0.12  0.01 -0.10  0.00  0.12 -0.11 -0.01
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## ACF  -0.18  0.10  0.02  0.00 -0.10 -0.05 -0.03  0.06 -0.06
## PACF -0.17  0.06 -0.07 -0.02 -0.06 -0.07 -0.07 -0.05 -0.09
```

Regressão com erros autocorrelacionais

Análise dos resíduos e seleção de variáveis de acordo com p-valor

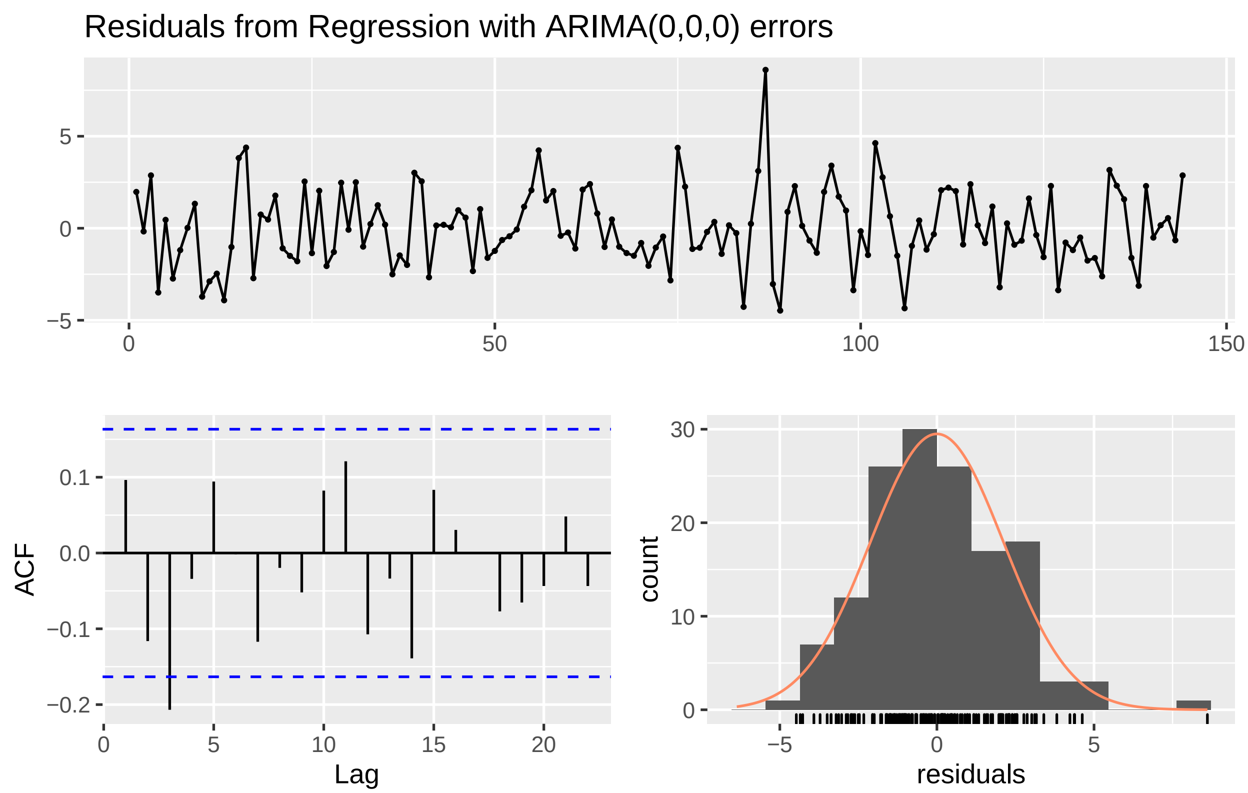
```
# Análise dos resíduos e seleção de variáveis de acordo com p-valor
fit2<- tirar_variaveis(0, 0, 0, x, y)
fit2[1]
```

```
## [[1]]
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## intercept                0.122436   0.232096  0.5275 0.5978295
## ‘Avicultura de Corte‘  0.430956   0.133305  3.2328 0.0012256 **
## Pescado                  0.309973   0.122283  2.5349 0.0112488 *
## avp12                    0.357888   0.071081  5.0350 4.779e-07 ***
## avc5                     0.341335   0.128150  2.6636 0.0077320 **
## pes9                    -0.441287   0.128116 -3.4444 0.0005722 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx)
coeftest(fit3)
```

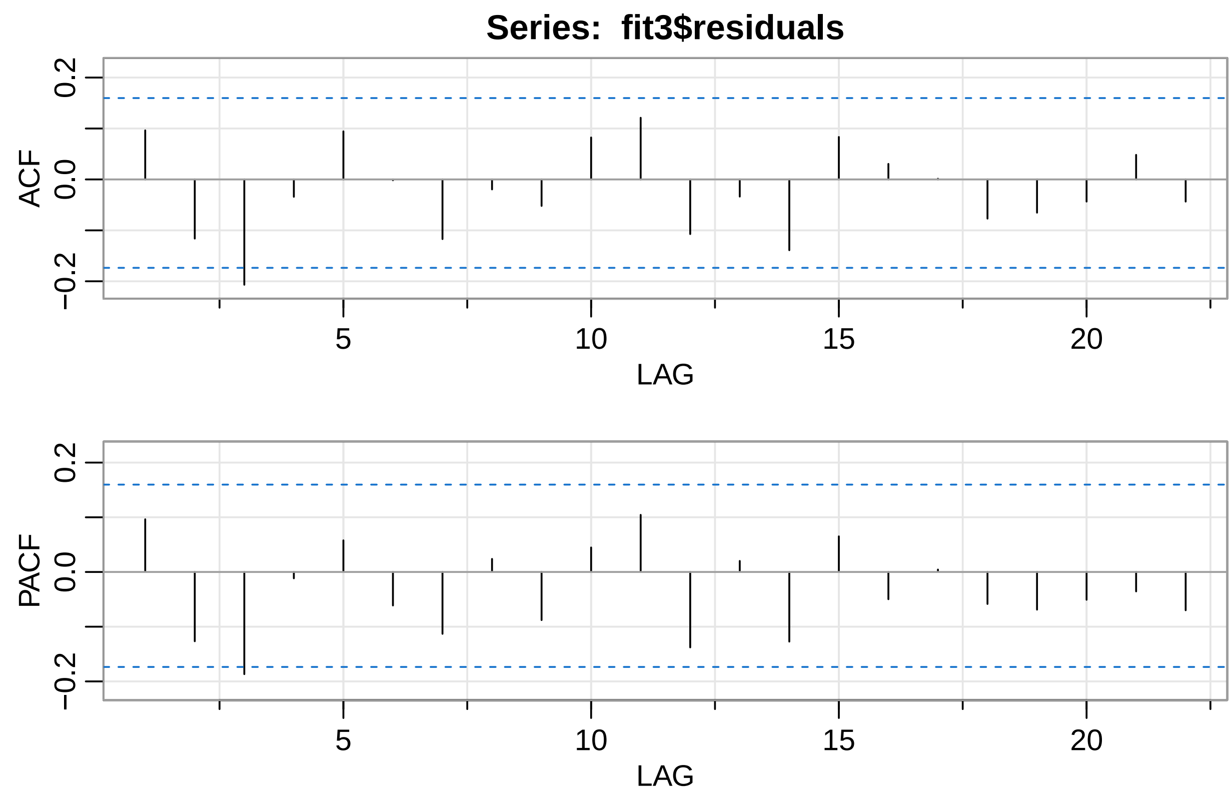
```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## intercept                0.122436   0.232096  0.5275 0.5978295
## ‘Avicultura de Corte‘  0.430956   0.133305  3.2328 0.0012256 **
## Pescado                  0.309973   0.122283  2.5349 0.0112488 *
## avp12                    0.357888   0.071081  5.0350 4.779e-07 ***
## avc5                     0.341335   0.128150  2.6636 0.0077320 **
## pes9                    -0.441287   0.128116 -3.4444 0.0005722 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
checkresiduals(fit3)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 14.912, df = 4, p-value = 0.004888
##
## Model df: 6.   Total lags used: 10
```

```
acf2(fit3$residuals)
```




```
##       [,1]  [,2]  [,3]  [,4] [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF   0.1 -0.12 -0.21 -0.03 0.09  0.00 -0.12 -0.02 -0.05  0.08  0.12 -0.11
## PACF  0.1 -0.13 -0.19 -0.01 0.06 -0.06 -0.11  0.02 -0.09  0.04  0.10 -0.14
##       [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## ACF   -0.03 -0.14  0.08  0.03     0 -0.08 -0.07 -0.04  0.05 -0.04
## PACF   0.02 -0.13  0.07 -0.05     0 -0.06 -0.07 -0.05 -0.04 -0.07
```

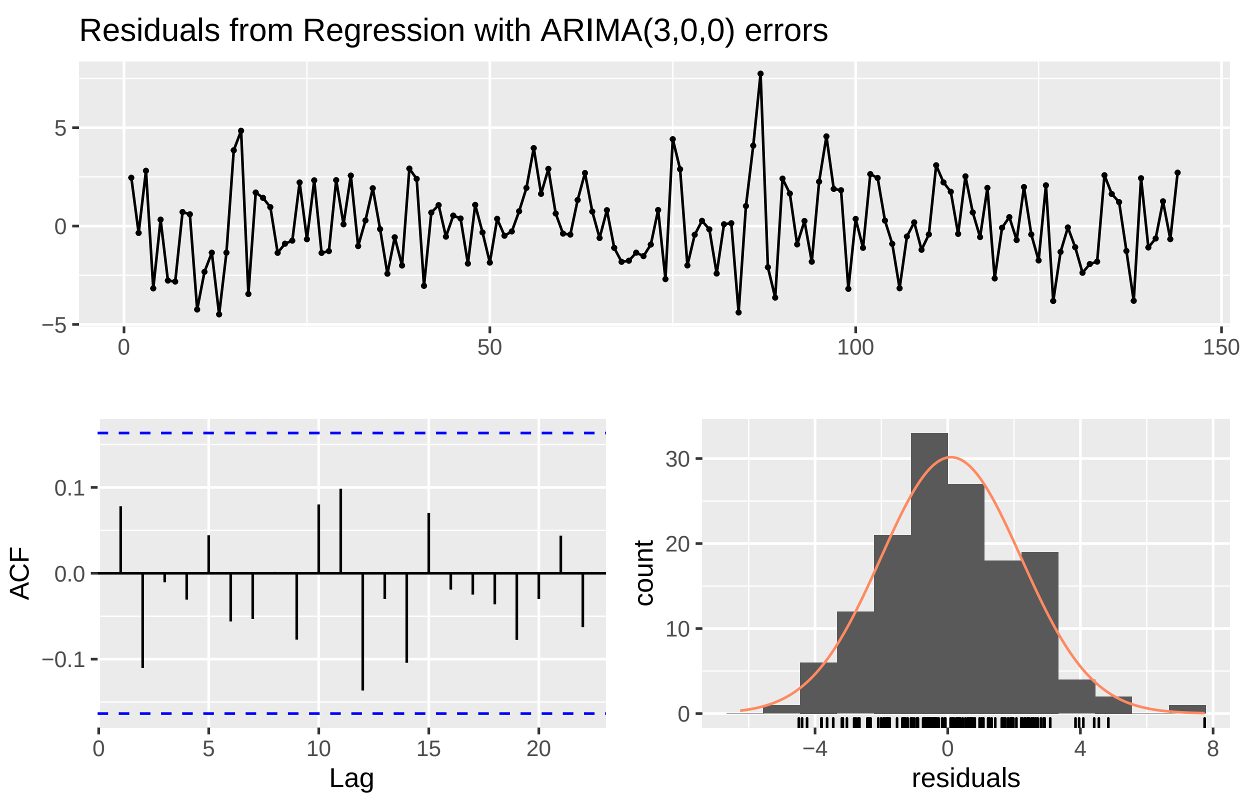
```
fit4 = Arima(y,order=c(3,0,0),xreg=xx,include.mean = FALSE,fixed=c(0,0,NA,NA,0,NA,NA,NA))
fit4
```

```
## Series: y
## Regression with ARIMA(3,0,0) errors
##
## Coefficients:
##        ar1  ar2      ar3  `Avicultura de Corte`  Pescado   avp12    avc5
##          0    0  -0.2280                 0.6010        0  0.4380  0.4151
## s.e.     0    0   0.0826                 0.1232        0  0.0624  0.1163
##           pes9
##        -0.3460
## s.e.    0.1074
##
## sigma^2 estimated as 4.628:  log likelihood=-312.18
## AIC=636.35   AICc=636.97   BIC=654.17
```

```r
coeftest(fit4)
```

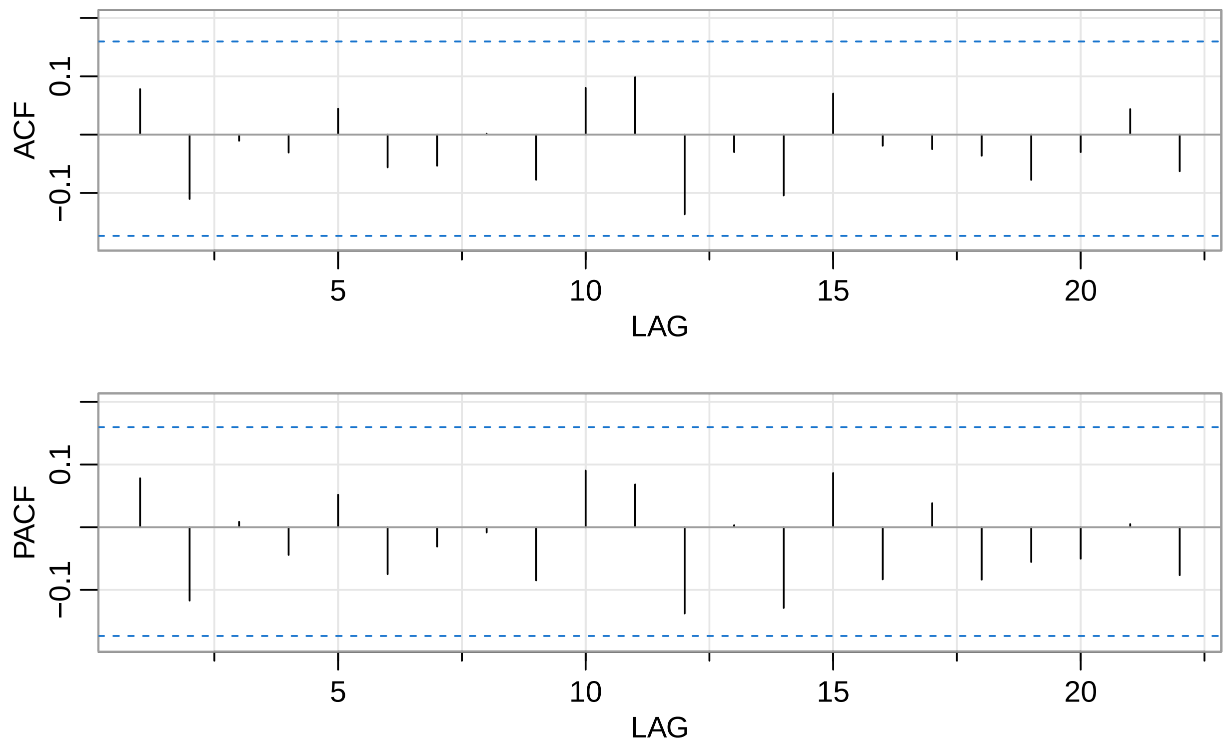
```
##
## z test of coefficients:
##
##                          Estimate Std. Error z value  Pr(>|z|)
## ar3                     -0.227984   0.082567 -2.7612 0.0057588 **
## `Avicultura de Corte`  0.601047   0.123219  4.8779 1.072e-06 ***
## avp12                    0.438035   0.062446  7.0147 2.305e-12 ***
## avc5                     0.415126   0.116253  3.5709 0.0003558 ***
## pes9                    -0.346032   0.107404 -3.2218 0.0012739 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
checkresiduals(fit4)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(3,0,0) errors
## Q* = 7.5364, df = 3, p-value = 0.05663
##
## Model df: 8.   Total lags used: 11
```

```r
acf2(fit4$residuals, main = "")
```




```
##      [,1]  [,2]  [,3]  [,4] [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF  0.08 -0.11 -0.01 -0.03 0.04 -0.06 -0.05  0.00 -0.08  0.08  0.10 -0.14
## PACF 0.08 -0.12  0.01 -0.04 0.05 -0.07 -0.03 -0.01 -0.08  0.09  0.07 -0.14
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
## ACF  -0.03 -0.10  0.07 -0.02 -0.02 -0.04 -0.08 -0.03  0.04 -0.06
## PACF  0.00 -0.13  0.09 -0.08  0.04 -0.08 -0.06 -0.05  0.00 -0.08
```

Modelo do Lácteos

Estruturando a base

```r
# Estruturando a base

df1<- funcao_lags(data_cut, data_cut$Lácteos, 'lact1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Postura`, 'avp1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc6', 6)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov2', 2)
df1 <- funcao_lags(df1, df1$Pescado, 'pes4', 4)
df1 <- funcao_lags(df1, df1$Pescado, 'pes9', 9)



df2 <- na.omit(df1)
```

Separando variável preditora e as covariáveis

```r
#Separando variável preditora e as covariáveis
x = model.matrix(Lácteos~.,df2)[,-1]
y = df2$Lácteos
```

Regressão classifica no contexto de Séries Temporais

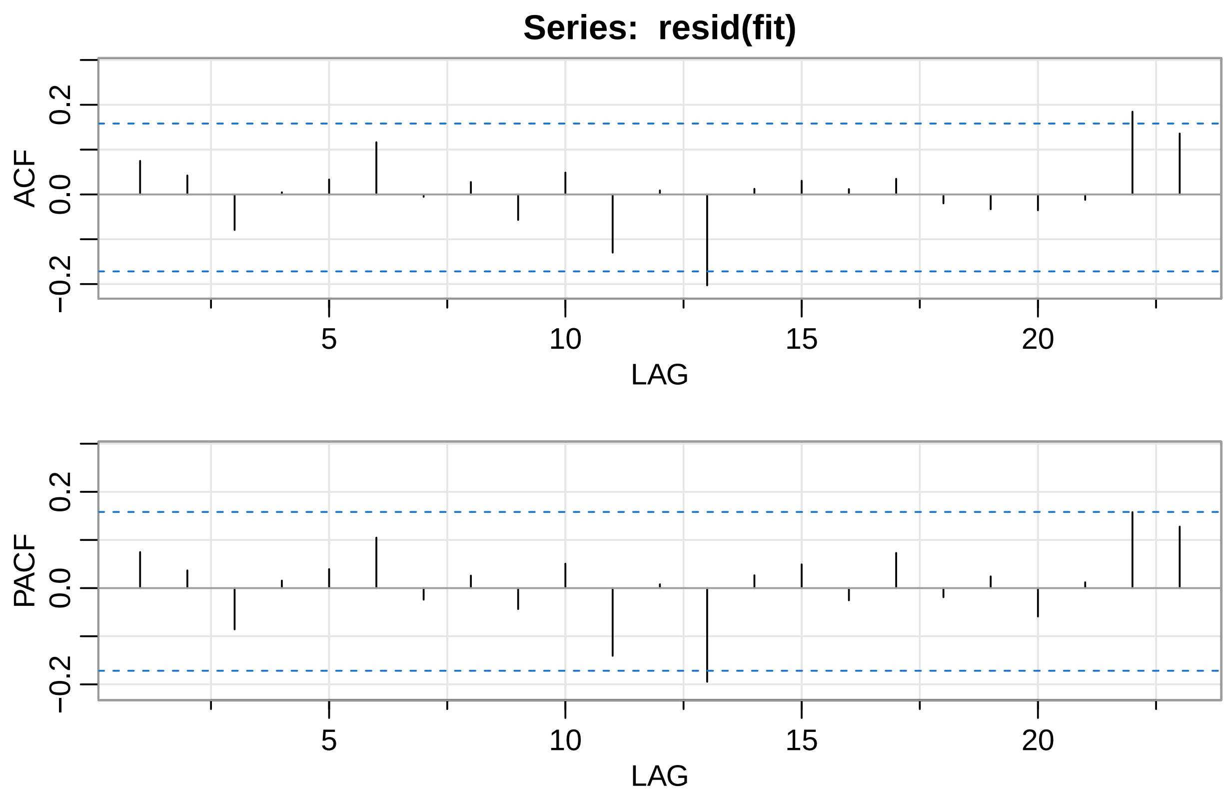
Criando o modelo de Regressão Simples

```r
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9889 -0.5093 -0.0365  0.3740  3.7350
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              0.15734    0.11576   1.359   0.1763
## xBovinocultura          -0.03768    0.04196  -0.898   0.3707
## x`Avicultura de Corte`   0.13320    0.06205   2.147   0.0336 *
## x`Avicultura de Postura` 0.03863    0.03119   1.238   0.2177
## xPescado                -0.03187    0.05250  -0.607   0.5449
## xSuinocultura           -0.01598    0.11157  -0.143   0.8863
## xlact1                   0.58502    0.06341   9.225 5.09e-16 ***
## xavp1                    0.03060    0.03009   1.017   0.3110
## xavc6                    0.07371    0.05424   1.359   0.1765
## xbov2                   -0.07026    0.04512  -1.557   0.1217
## xpes4                    0.08919    0.05472   1.630   0.1055
## xpes9                   -0.09059    0.05370  -1.687   0.0939 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8391 on 135 degrees of freedom
## Multiple R-squared:  0.5426, Adjusted R-squared:  0.5053
## F-statistic: 14.56 on 11 and 135 DF,  p-value: < 2.2e-16
```

Análise dos Resíduos

```r
# Análise dos Resíduos
acf2(resid(fit))
```




```
##      [,1] [,2]  [,3] [,4] [,5] [,6]  [,7] [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.07 0.04 -0.08 0.00 0.03 0.12 -0.01 0.03 -0.06  0.05 -0.13  0.01 -0.20
## PACF 0.07 0.04 -0.09 0.02 0.04 0.10 -0.02 0.03 -0.04  0.05 -0.14  0.01 -0.19
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF   0.01  0.03  0.01  0.03 -0.02 -0.03 -0.04 -0.01  0.18  0.14
## PACF  0.03  0.05 -0.03  0.07 -0.02  0.02 -0.06  0.01  0.16  0.13
```

Análise dos resíduos e seleção de variáveis de acordo com p-valor

```
# Análise dos resíduos e seleção de variáveis de acordo com p-valor
fit2 <- tirar_variaveis(0, 0, 0, x, y)

fit2[1]
```

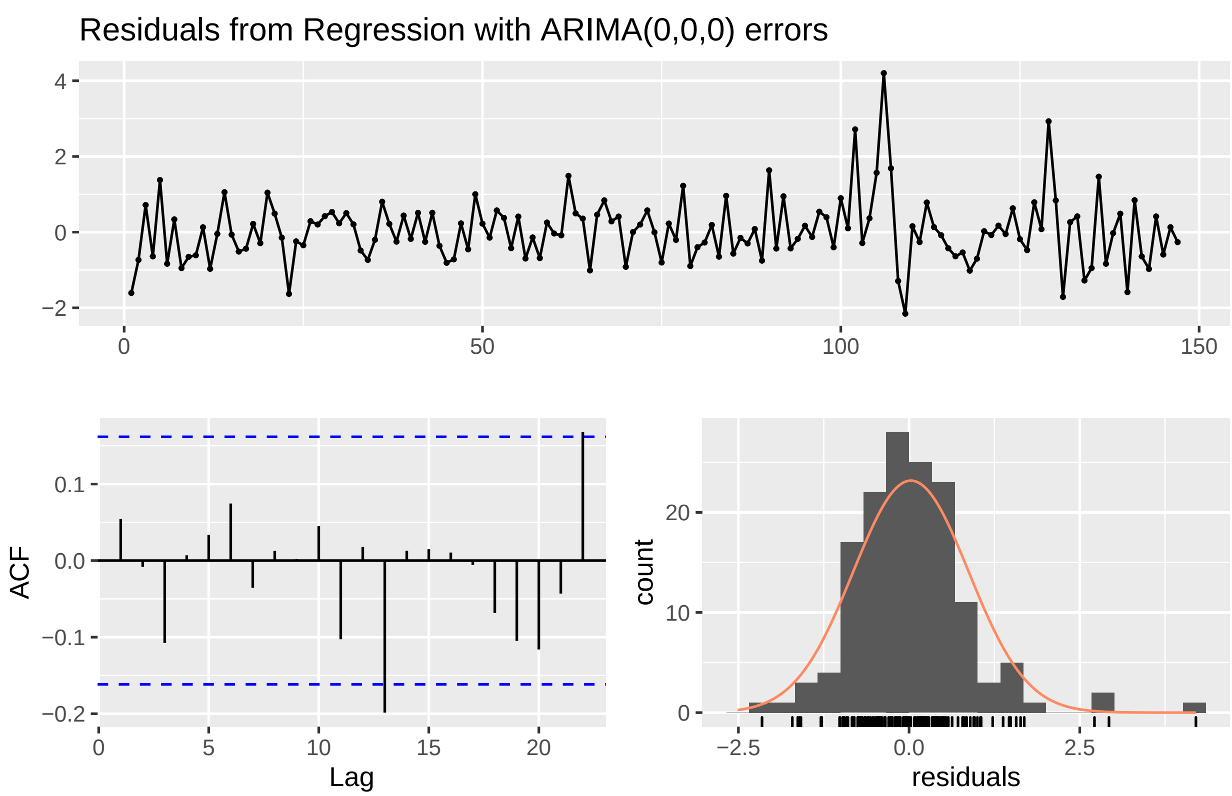
```
## [[1]]
##
## z test of coefficients:
##
##           Estimate Std. Error z value  Pr(>|z|)
## intercept 0.036024   0.080495  0.4475  0.654489
## lact1     0.604999   0.060150 10.0582 < 2.2e-16 ***
## avc6      0.107164   0.049809  2.1515  0.031436 *
## pes4      0.145117   0.046861  3.0968  0.001956 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx,include.mean = FALSE)
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##       Estimate Std. Error z value  Pr(>|z|)
## lact1 0.611972   0.058137 10.5265 < 2.2e-16 ***
## avc6  0.112091   0.048610  2.3059 0.0211138 *
## pes4  0.150462   0.045344  3.3182 0.0009059 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

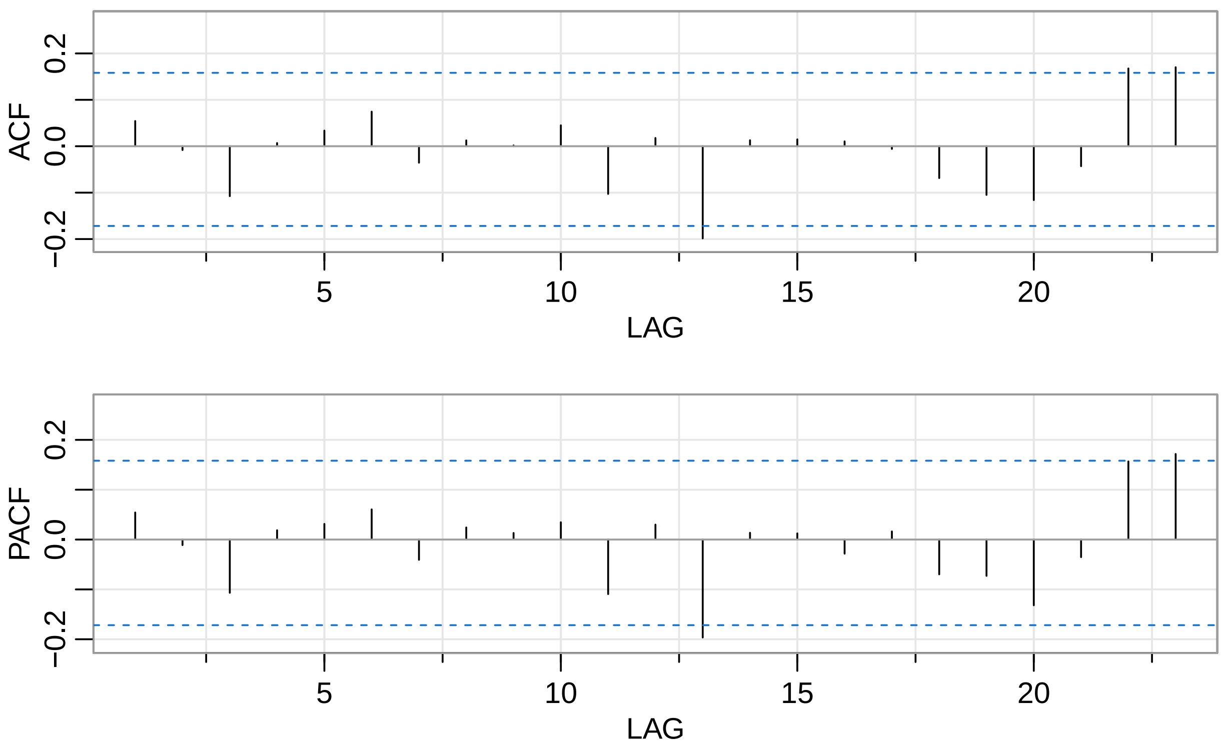
```
checkresiduals(fit3)
```




```
##
##  Ljung-Box test
```

```
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 3.8077, df = 7, p-value = 0.8016
##
## Model df: 3.   Total lags used: 10
```

```
acf2(fit3$residuals, main = "")
```



```
##      [,1]  [,2]  [,3] [,4] [,5] [,6]  [,7] [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.05 -0.01 -0.11 0.01 0.03 0.07 -0.04 0.01 0.00  0.05 -0.10  0.02  -0.2
## PACF 0.05 -0.01 -0.11 0.02 0.03 0.06 -0.04 0.02 0.01  0.03 -0.11  0.03  -0.2
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF   0.01  0.01  0.01 -0.01 -0.07 -0.10 -0.12 -0.04  0.17  0.17
## PACF  0.01  0.01 -0.03  0.02 -0.07 -0.07 -0.13 -0.04  0.16  0.17
```

Modelo do Suinocultura

Estruturando a base

```
# Estruturando a base
df1<- funcao_lags(data_cut, data_cut$Suinocultura, 'su1', 1)
```

```r
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc6', 6)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc10', 10)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov1', 1)
df2 <- na.omit(df1)
```

Separando variável preditora e as covariáveis

```r
# Separando variável preditora e as covariáveis
x = model.matrix(Suinocultura~.,df2)[,-1]
y = df2$Suinocultura
```

Regressão classifica no contexto de Séries Temporais

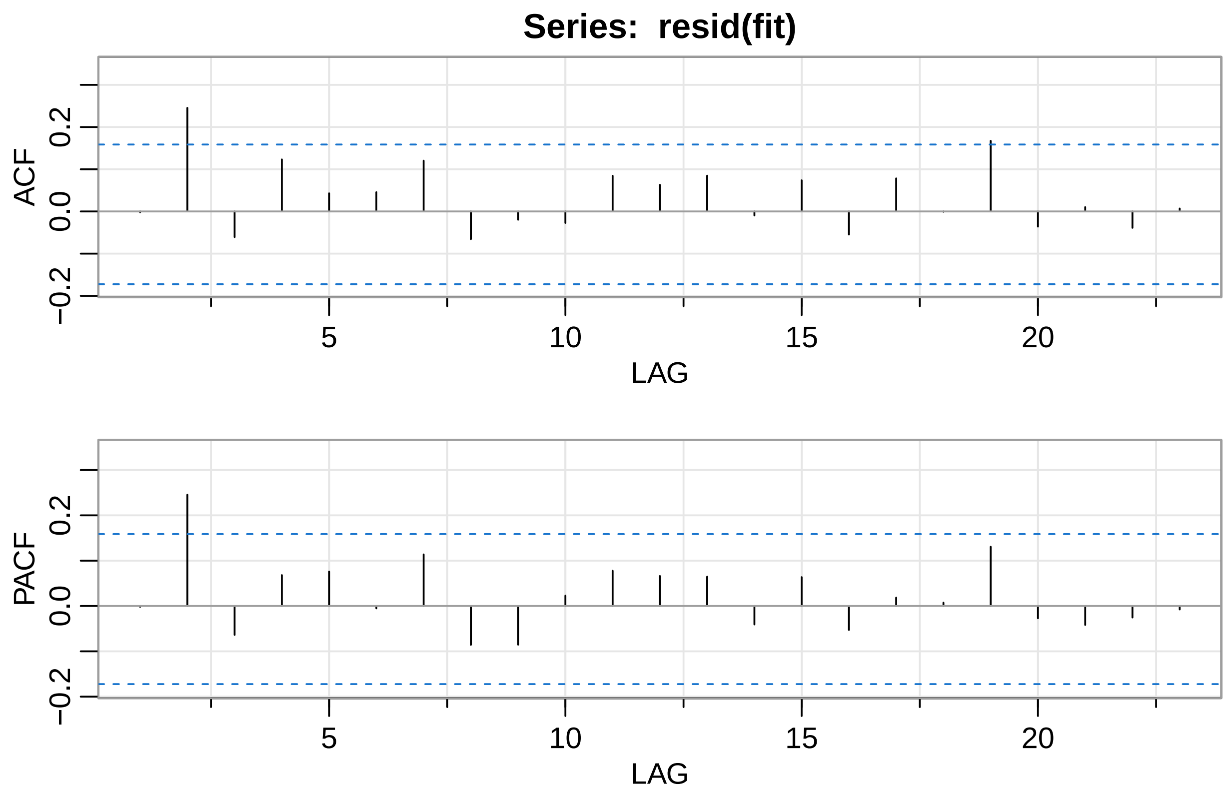
Criando o modelo de Regressão Simples

```r
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.41583 -0.39128 -0.06148  0.35167  1.67796
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.1920512  0.0735974   2.609  0.01009 *
## xBovinocultura            0.0804015  0.0323240   2.487  0.01409 *
## x`Avicultura de Corte`    0.0242656  0.0481760   0.504  0.61530
## x`Avicultura de Postura`  0.0098311  0.0211227   0.465  0.64237
## xPescado                 -0.0008281  0.0373895  -0.022  0.98236
## xLácteos                  0.0338683  0.0463772   0.730  0.46649
## xsu1                      0.2273228  0.0776042   2.929  0.00399 **
## xavc1                     0.0865616  0.0462072   1.873  0.06318 .
## xavc6                    -0.0686232  0.0384825  -1.783  0.07680 .
## xavc10                    0.0625848  0.0364282   1.718  0.08808 .
## xbov1                     0.0738746  0.0405940   1.820  0.07100 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6066 on 135 degrees of freedom
## Multiple R-squared:  0.3972, Adjusted R-squared:  0.3526
## F-statistic: 8.896 on 10 and 135 DF,  p-value: 3.949e-11
```

Análise dos Resíduos

```r
# Análise dos Resíduos
acf2(resid(fit))
```




```
##      [,1] [,2]  [,3] [,4] [,5]  [,6] [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF     0 0.25 -0.06 0.12 0.04  0.05 0.12 -0.07 -0.02 -0.03  0.08  0.06  0.08
## PACF    0 0.25 -0.06 0.07 0.08 -0.01 0.11 -0.09 -0.09  0.02  0.08  0.07  0.06
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.01  0.07 -0.05  0.08  0.00  0.17 -0.04  0.01 -0.04  0.01
## PACF -0.04  0.06 -0.05  0.02  0.01  0.13 -0.03 -0.04 -0.03 -0.01
```

Análise dos resíduos e seleção de variáveis de acordo com p-valor

```r
# Análise dos resíduos e seleção de variáveis de acordo com p-valor

fit2 <- tirar_variaveis(0, 0, 0, x, y)

fit2[1]
```

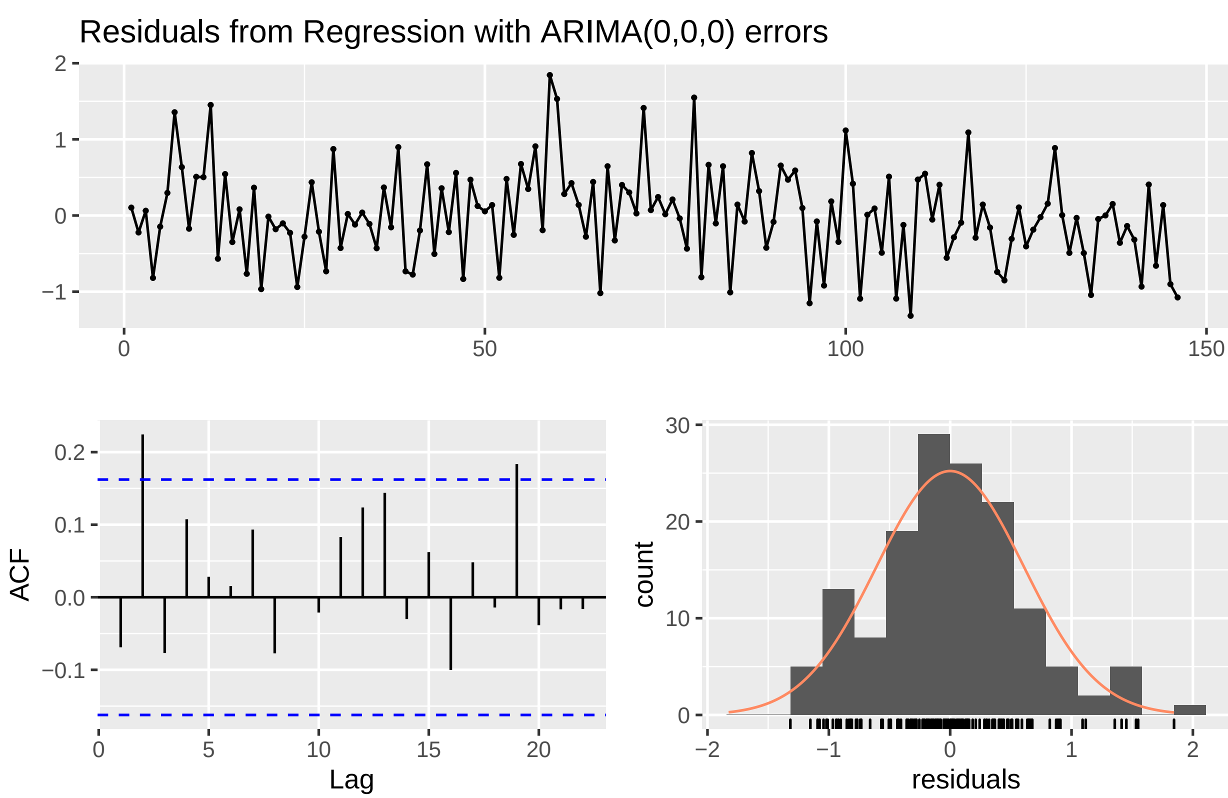
```
## [[1]]
##
## z test of coefficients:
##
##                Estimate Std. Error z value  Pr(>|z|)
## intercept      0.196715   0.062298  3.1576 0.0015905 **
```

```
## Bovinocultura 0.110931   0.024454  4.5363 5.725e-06 ***
## su1          0.293553   0.073307  4.0044 6.217e-05 ***
## avc1         0.134372   0.038667  3.4751 0.0005106 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx)
coeftest(fit3)
```

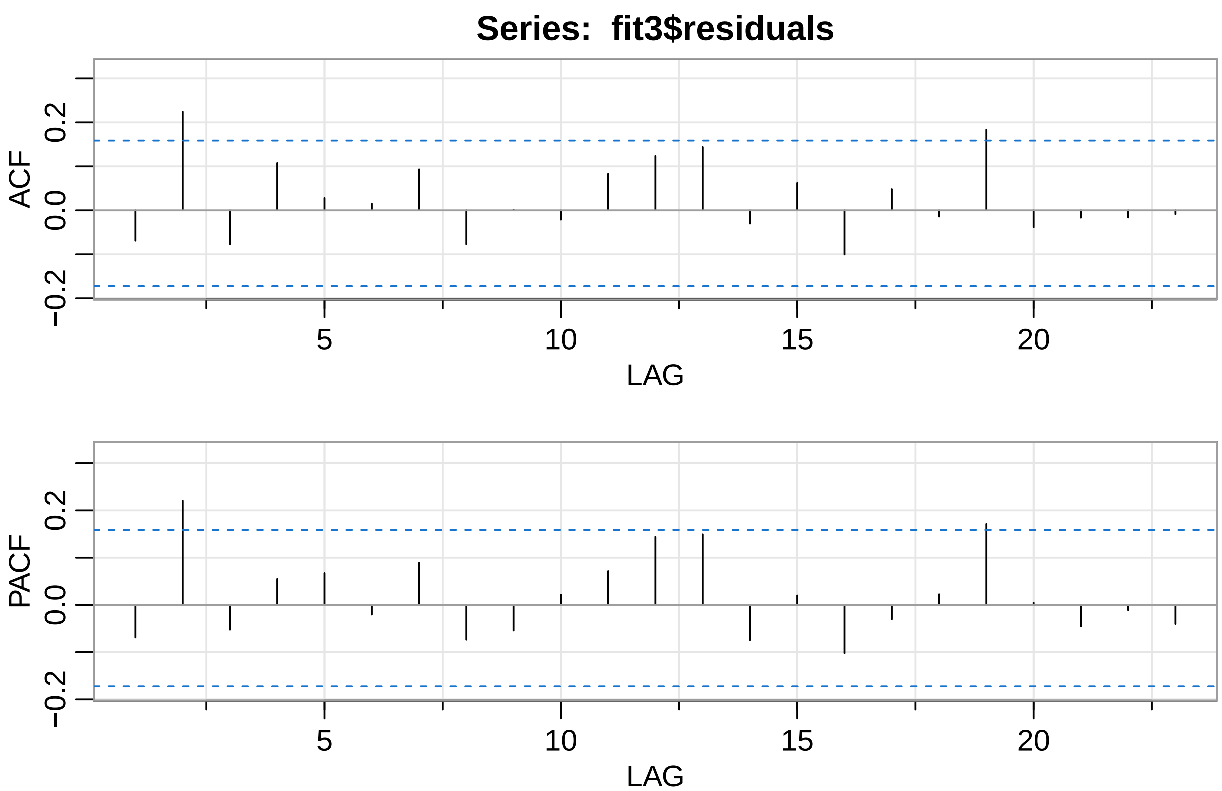
```
##
## z test of coefficients:
##
##              Estimate Std. Error z value  Pr(>|z|)
## intercept    0.196715   0.062298  3.1576 0.0015905 **
## Bovinocultura 0.110931   0.024454  4.5363 5.725e-06 ***
## su1          0.293553   0.073307  4.0044 6.217e-05 ***
## avc1         0.134372   0.038667  3.4751 0.0005106 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
checkresiduals(fit3)
```



```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,0,0) errors
## Q* = 13.432, df = 6, p-value = 0.03667
##
## Model df: 4.   Total lags used: 10
```

```
acf2(fit3$residuals)
```




```
##       [,1] [,2]  [,3] [,4] [,5]  [,6] [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  -0.07 0.22 -0.08 0.11 0.03  0.02 0.09 -0.08  0.00 -0.02  0.08  0.12  0.14
## PACF -0.07 0.22 -0.05 0.05 0.07 -0.02 0.09 -0.07 -0.05  0.02  0.07  0.14  0.15
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.03  0.06  -0.1  0.05 -0.01  0.18 -0.04 -0.02 -0.02 -0.01
## PACF -0.07  0.02  -0.1 -0.03  0.02  0.17  0.00 -0.05 -0.01 -0.04
```

```
fit4 = Arima(y,order=c(2,0,0),xreg=xx,fixed =c(0,NA,NA,NA,NA,NA))
fit4
```

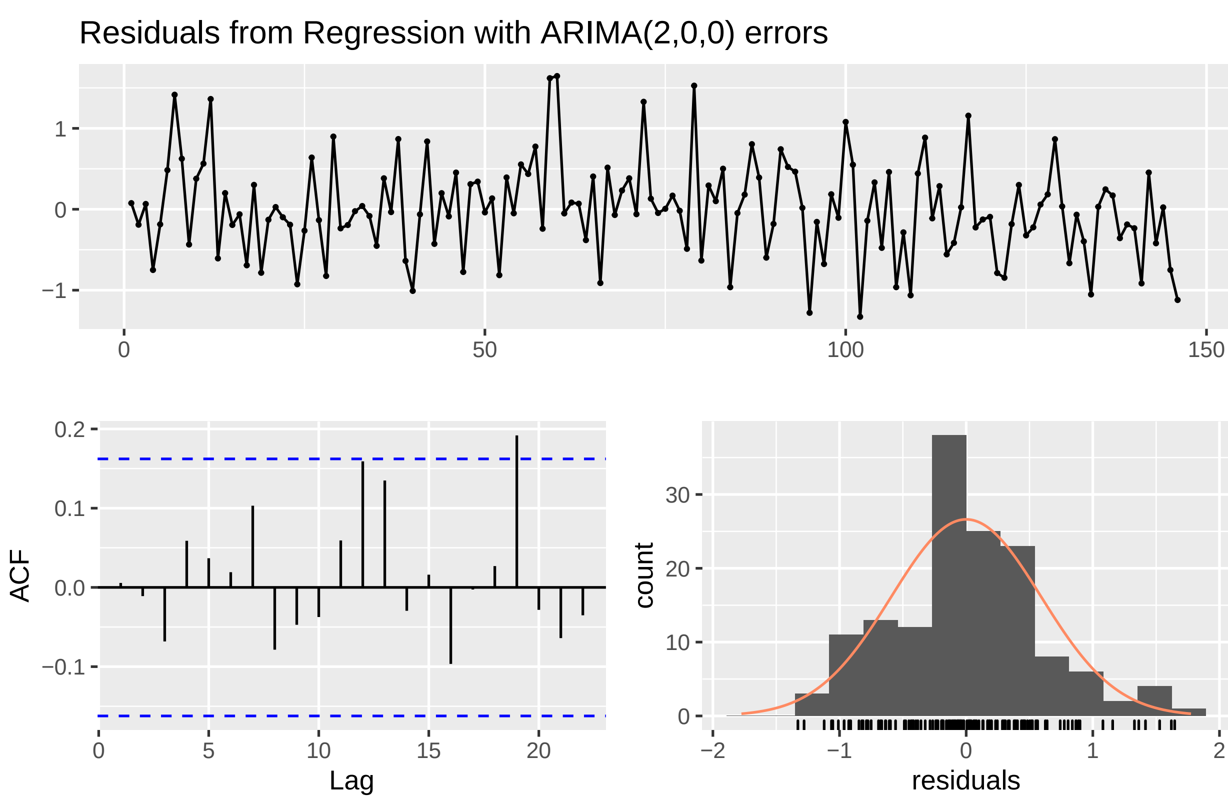
```
## Series: y
## Regression with ARIMA(2,0,0) errors
##
## Coefficients:
```

```
##       ar1     ar2  intercept  Bovinocultura     su1    avc1
##         0  0.2407     0.2107         0.1116  0.2485  0.1412
## s.e.    0  0.0832     0.0750         0.0235  0.0747  0.0370
##
## sigma^2 estimated as 0.36:  log likelihood=-130.1
## AIC=272.2   AICc=272.81   BIC=290.11
```

```
coeftest(fit4)
```

```
##
## z test of coefficients:
##
##                Estimate Std. Error z value  Pr(>|z|)
## ar2            0.240708   0.083246  2.8915 0.0038338 **
## intercept      0.210659   0.075006  2.8086 0.0049764 **
## Bovinocultura 0.111554   0.023545  4.7379 2.159e-06 ***
## su1            0.248548   0.074714  3.3267 0.0008789 ***
## avc1           0.141187   0.037044  3.8113 0.0001382 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

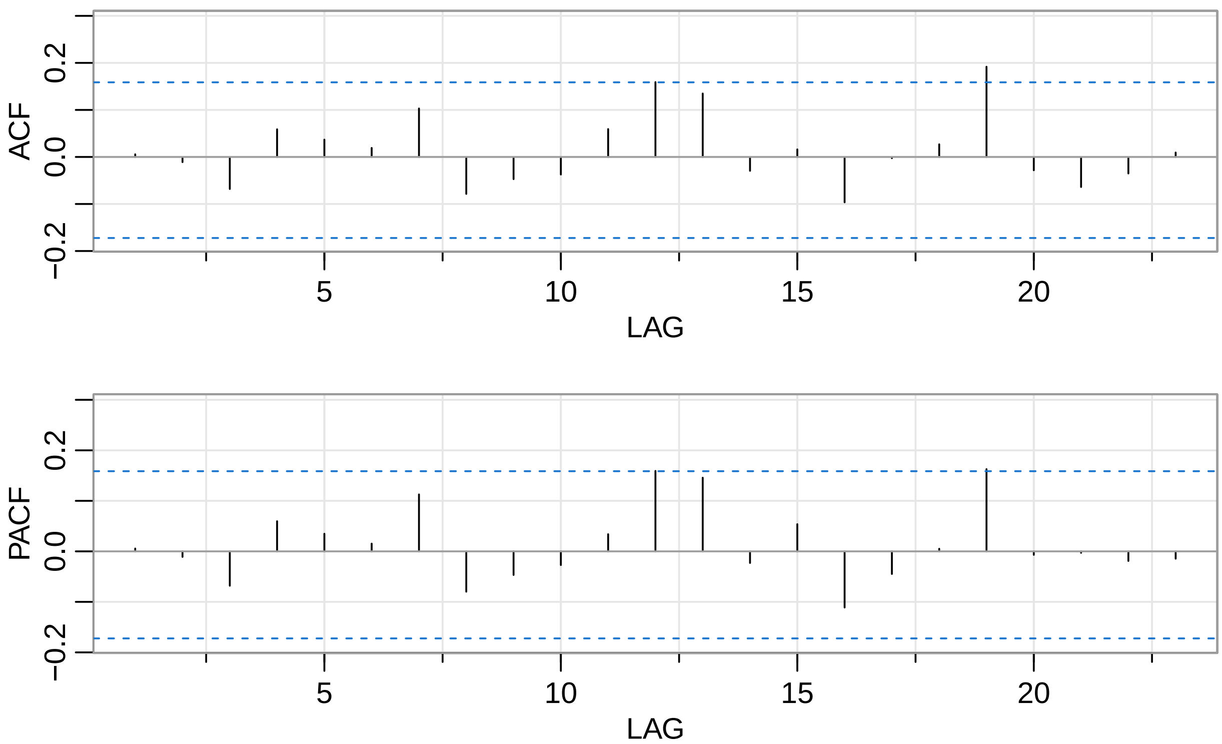
```
checkresiduals(fit4)
```




```
##
```

```
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(2,0,0) errors
## Q* = 4.7092, df = 4, p-value = 0.3185
##
## Model df: 6.   Total lags used: 10
```

```
acf2(fit4$residuals, main = "")
```



```
##      [,1]  [,2]  [,3] [,4] [,5] [,6] [,7]  [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF  0.01 -0.01 -0.07 0.06 0.04 0.02 0.10 -0.08 -0.05 -0.04  0.06  0.16  0.13
## PACF 0.01 -0.01 -0.07 0.06 0.03 0.02 0.11 -0.08 -0.05 -0.03  0.03  0.16  0.15
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23]
## ACF  -0.03  0.02 -0.10  0.00  0.03  0.19 -0.03 -0.06 -0.04  0.01
## PACF -0.02  0.05 -0.11 -0.04  0.01  0.16 -0.01  0.00 -0.02 -0.01
```

Análise das séries temporais anuais

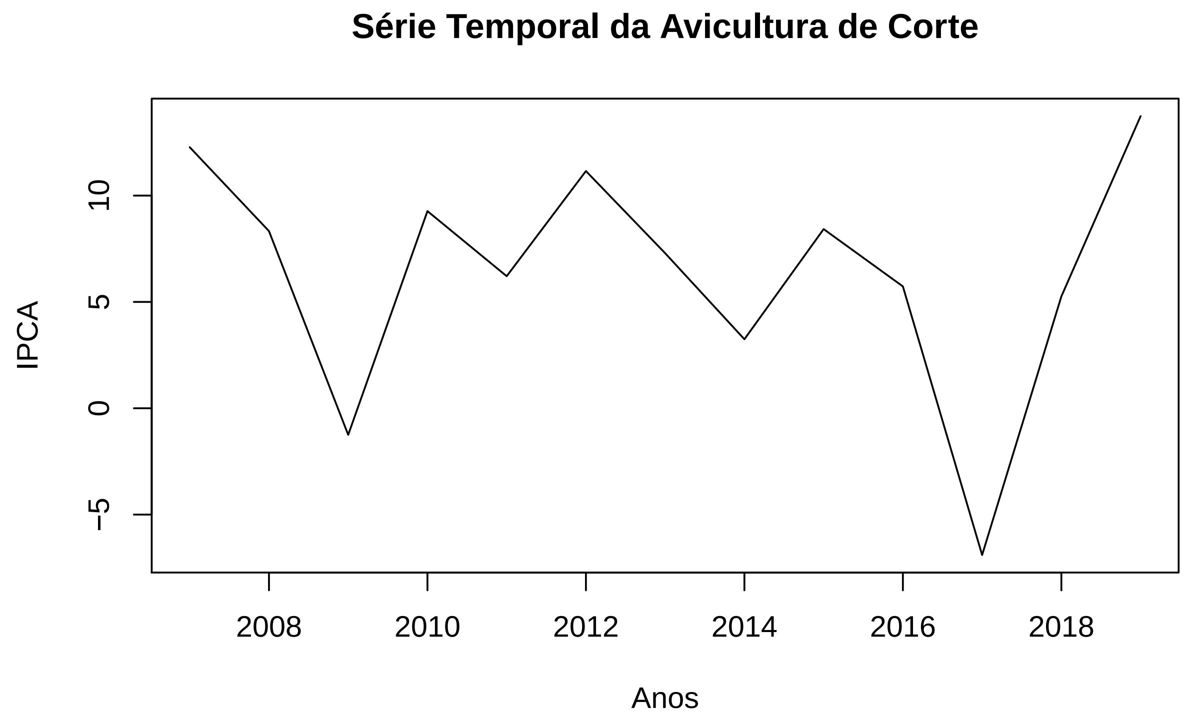
Análise Descritiva

```
# Análise das séries temporais anuais
head(data_anual)
```

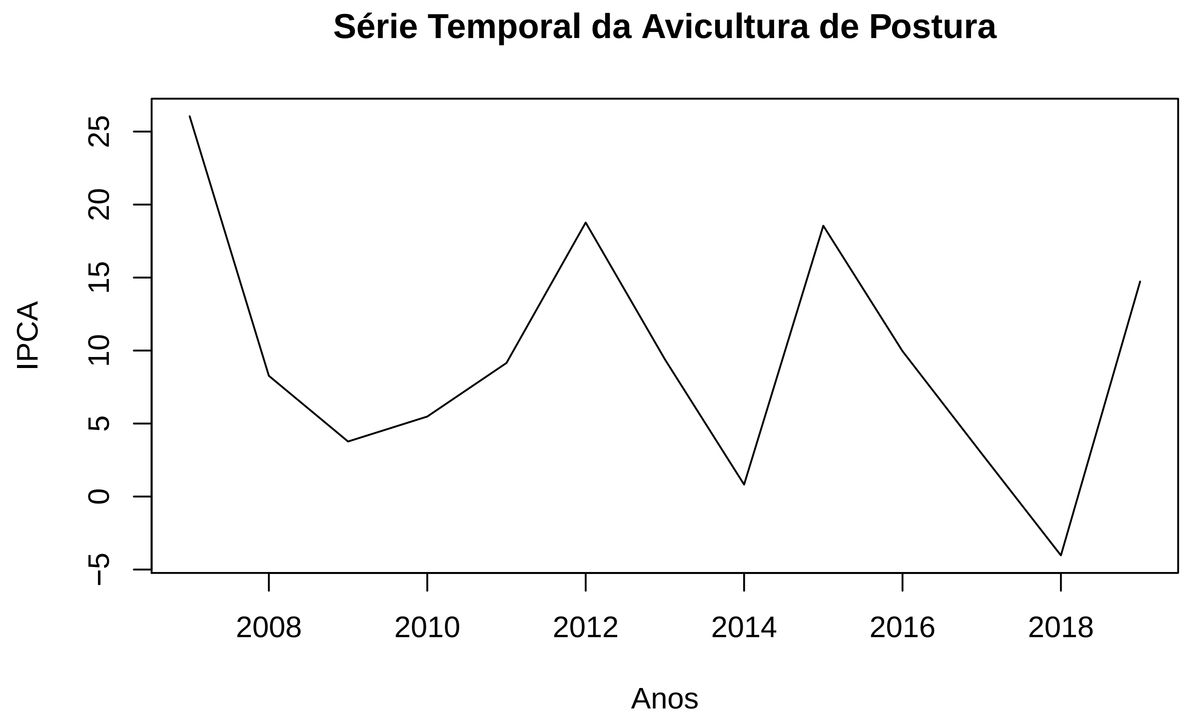
```
## # A tibble: 6 x 7
##    Anos ‘Avicultura de ~ ‘Avicultura Pos~ ‘Bovinocultura ~ Lácteos Pescado
##   <dbl>            <dbl>            <dbl>            <dbl>   <dbl>   <dbl>
## 1  2007            12.3             26.0            20.5     21.7     1.40
## 2  2008             8.33             8.27           23.7     -2.41    9.89
## 3  2009            -1.25             3.77           -3.75     4.55    7.12
## 4  2010             9.27             5.48           25.9      4.36    8.02
## 5  2011             6.21             9.15            3.67     7.51    6.61
## 6  2012            11.2             18.8             0.792    7.76   14.2
## # ... with 1 more variable: Suinocultura <dbl>
```

```r
# Análise Descritiva
z_avc = data_anual$`Avicultura de Corte`
z_avc = ts(z_avc, frequency = 1, start = 2007, end = 2019)

z_avp = data_anual$`Avicultura Postura`
z_avp = ts(z_avp, frequency = 1, start = 2007, end = 2019)

z_bov = data_anual$`Bovinocultura de corte`
z_bov = ts(z_bov, frequency = 1, start = 2007, end = 2019)

z_lac = data_anual$`Lácteos`
z_lac = ts(z_lac, frequency = 1, start = 2007, end = 2019)

z_pesc = data_anual$Pescado
z_pesc = ts(z_pesc, frequency = 1, start = 2007, end = 2019)

z_suino = data_anual$Suinocultura
z_suino = ts(z_suino, frequency = 1, start = 2007, end = 2019)
```

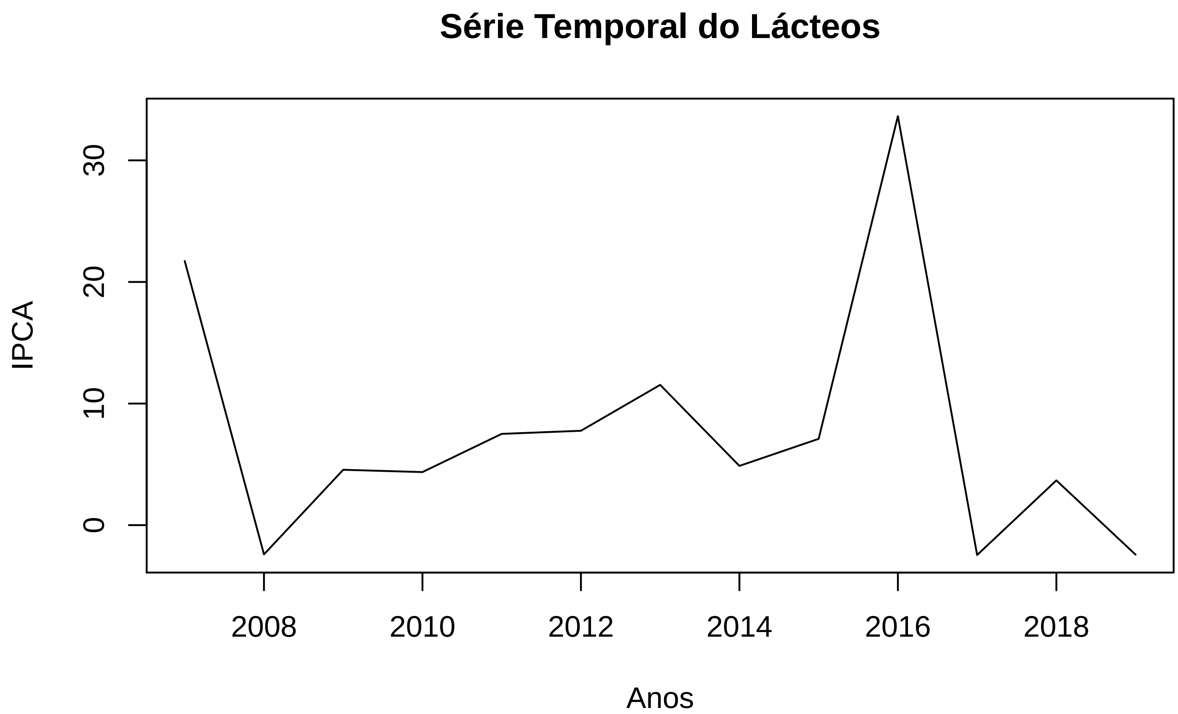
```r
# Análise Descritiva
plot(z_avc,main="Série Temporal da Avicultura de Corte", xlab= "Anos", ylab="IPCA")
```



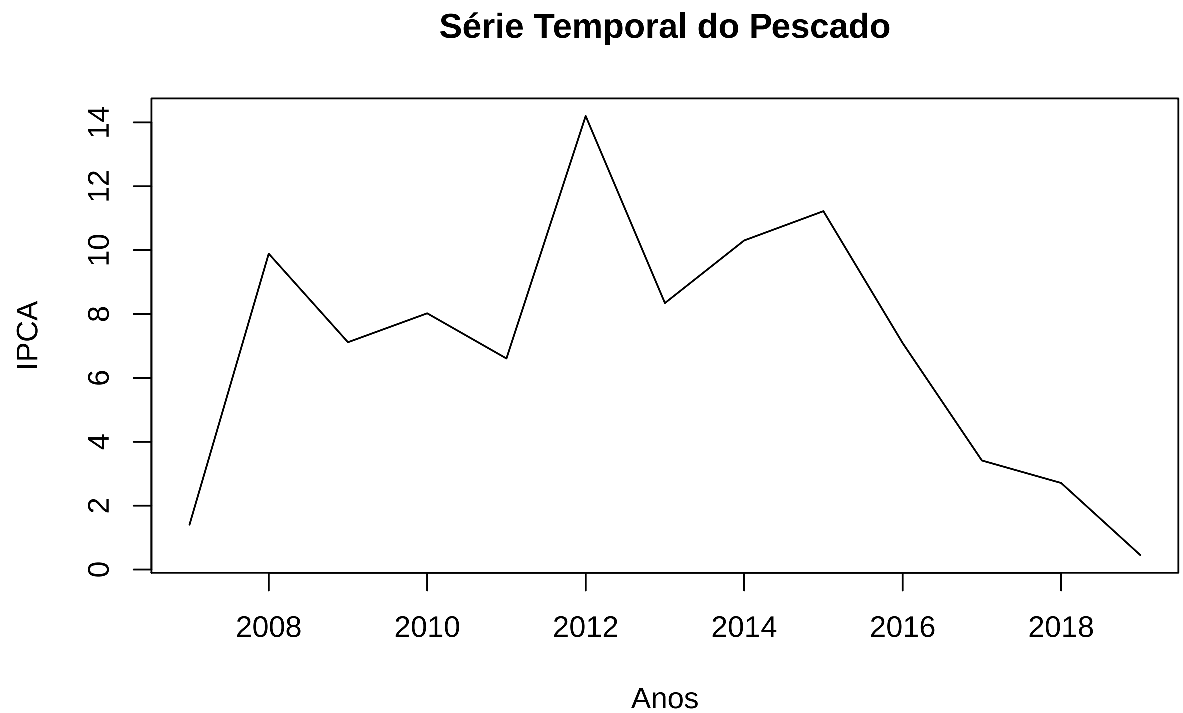

```
plot(z_avp,main="Série Temporal da Avicultura de Postura", xlab= "Anos", ylab="IPCA")
```



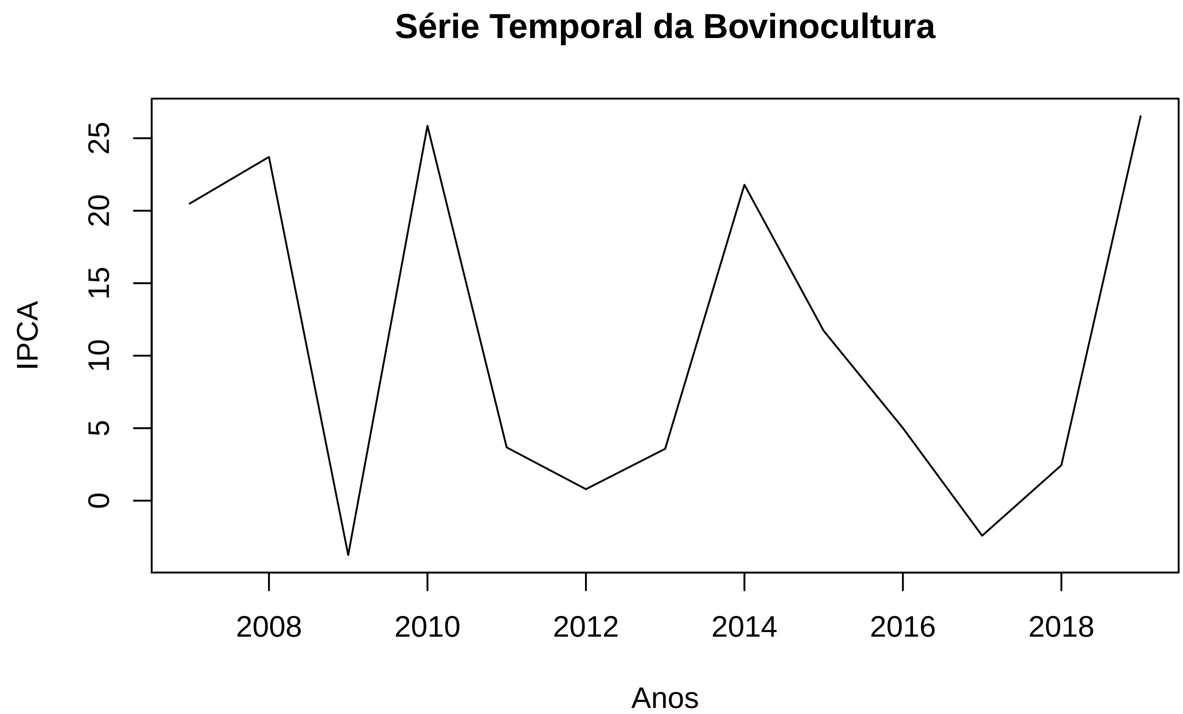

```r
plot(z_lac,main="Série Temporal do Lácteos", xlab= "Anos", ylab="IPCA")
```



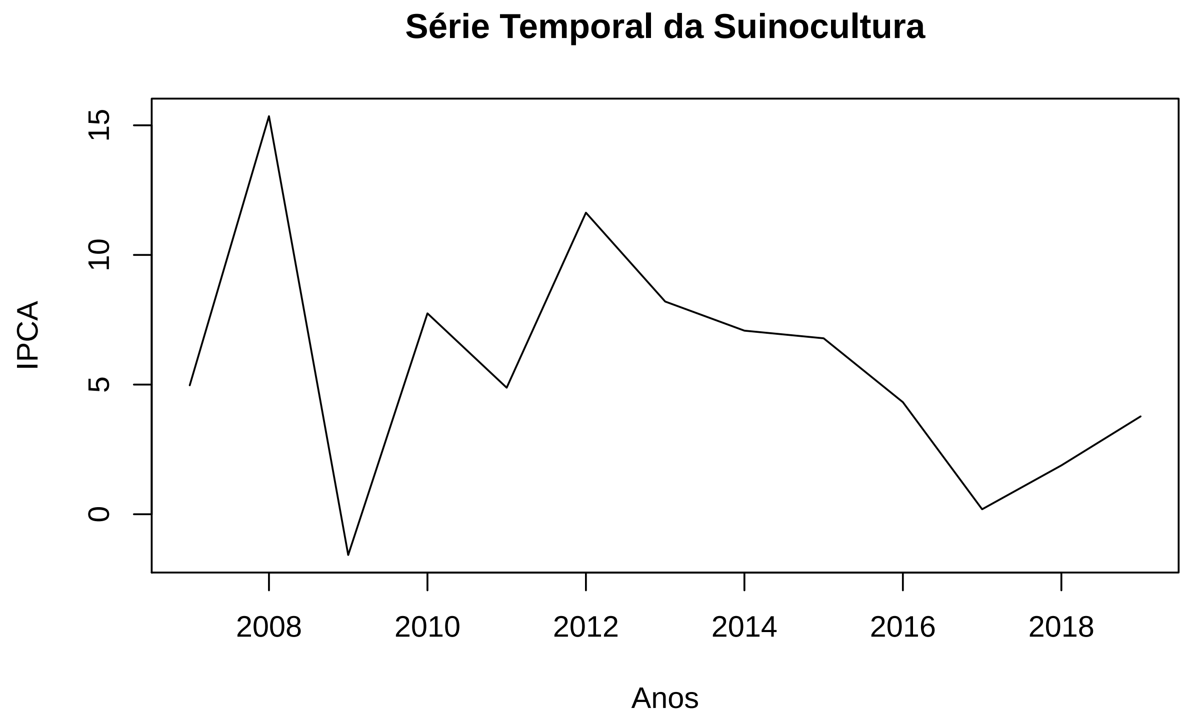

```
plot(z_pesc,main="Série Temporal do Pescado", xlab= "Anos", ylab="IPCA")
```




```
plot(z_bov,main="Série Temporal da Bovinocultura", xlab= "Anos", ylab="IPCA")
```

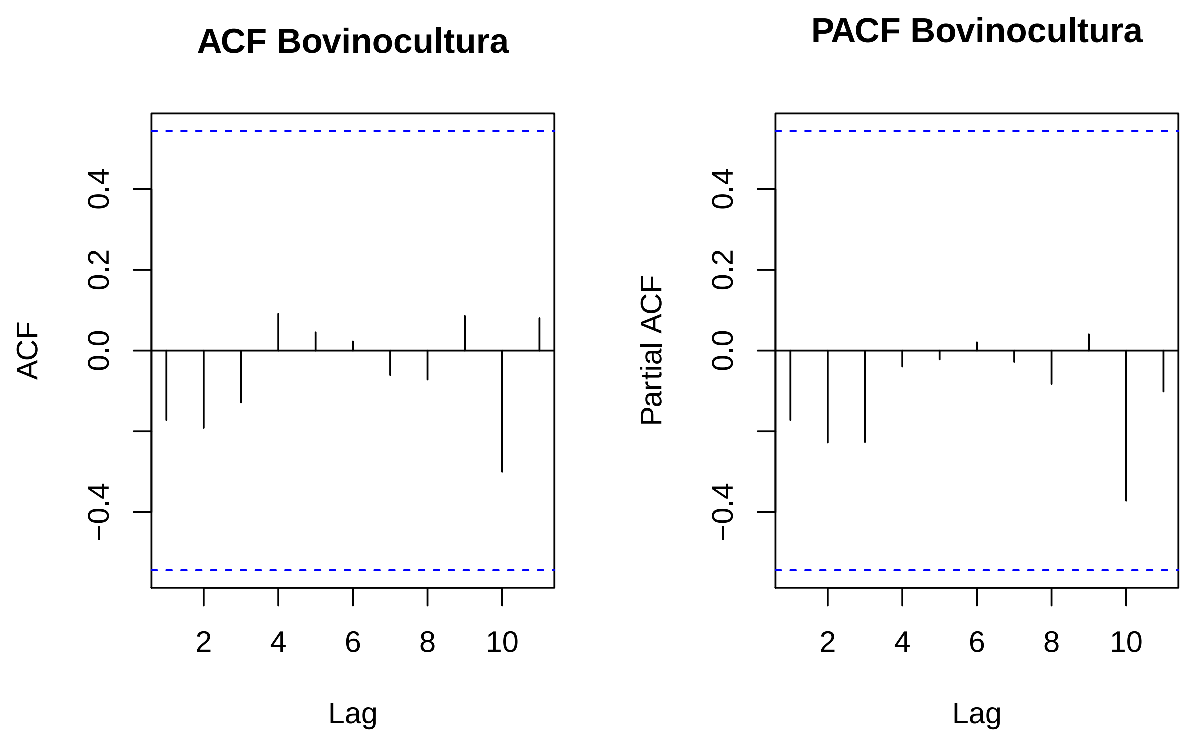



```
plot(z_suino,main="Série Temporal da Suinocultura", xlab= "Anos", ylab="IPCA")
```



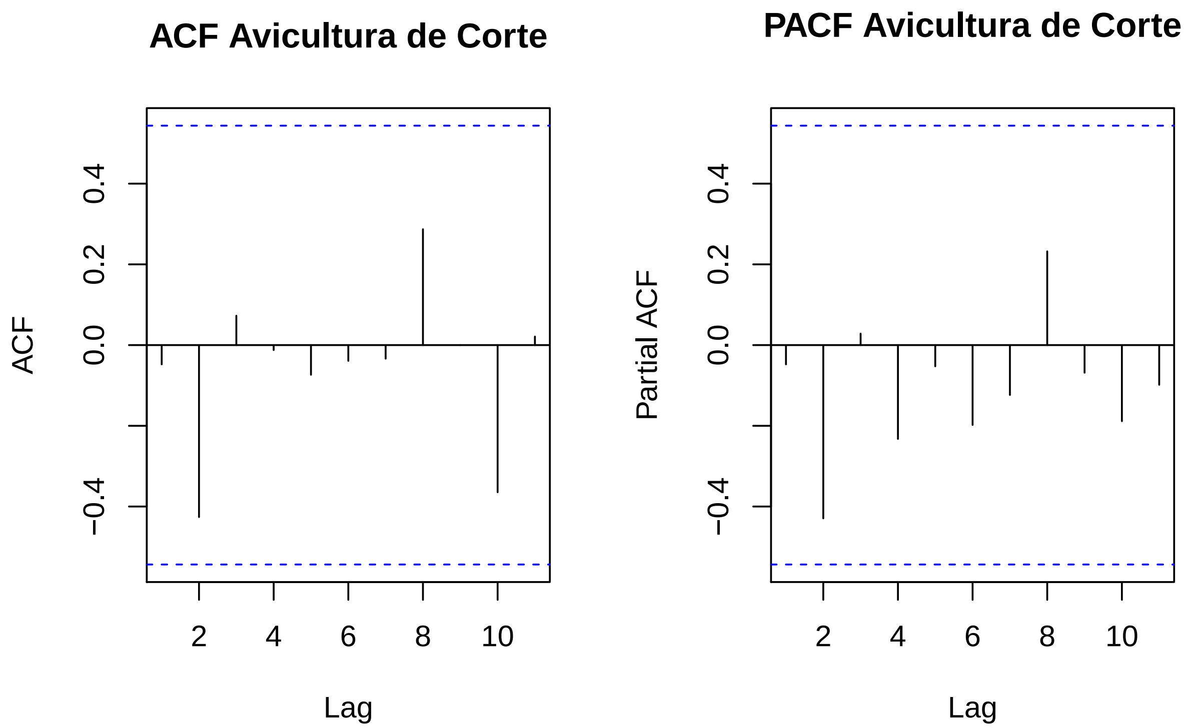
Funções de Autocorrelações para Bovinocultura

```
#Funções de Autocorrelações para Bovinocultura
par(mfrow = c(1, 2))
acf(z_bov, main="ACF Bovinocultura")
pacf(z_bov, main="PACF Bovinocultura")
```



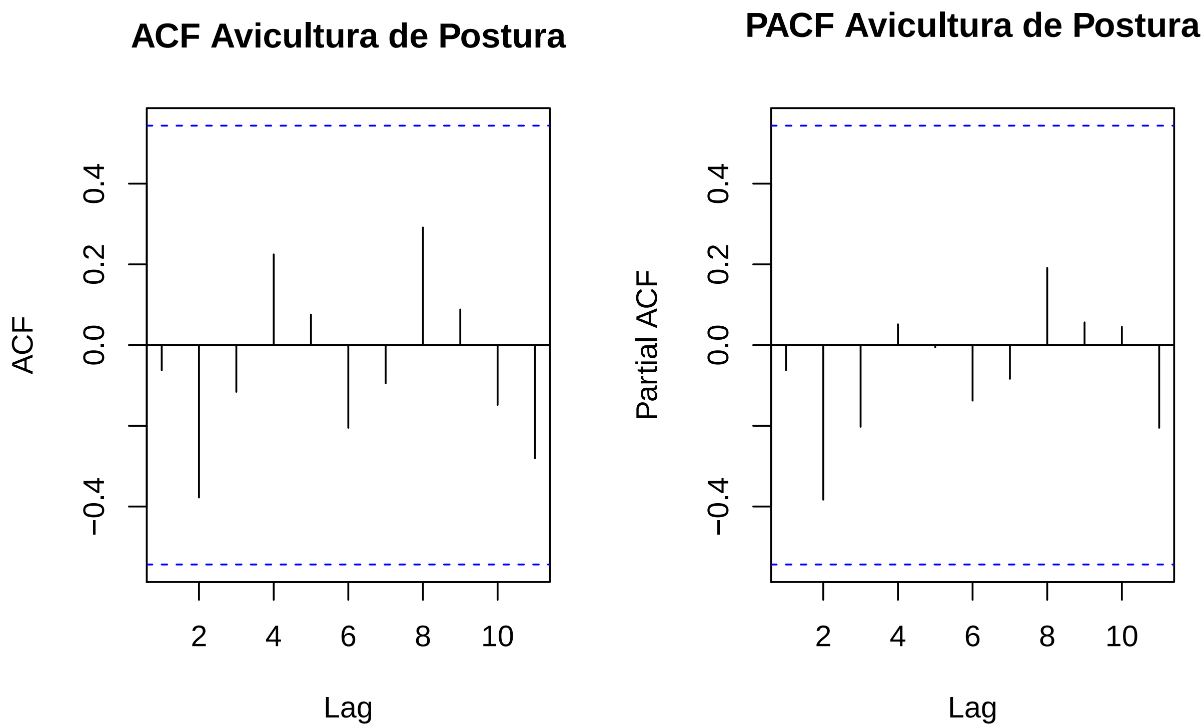
Funções de Autocorrelações para Avicultura de Corte

```
#Funções de Autocorrelações para Avicultura de Corte
par(mfrow = c(1, 2))
acf(z_avc, main="ACF Avicultura de Corte")
pacf(z_avc, main="PACF Avicultura de Corte")
```



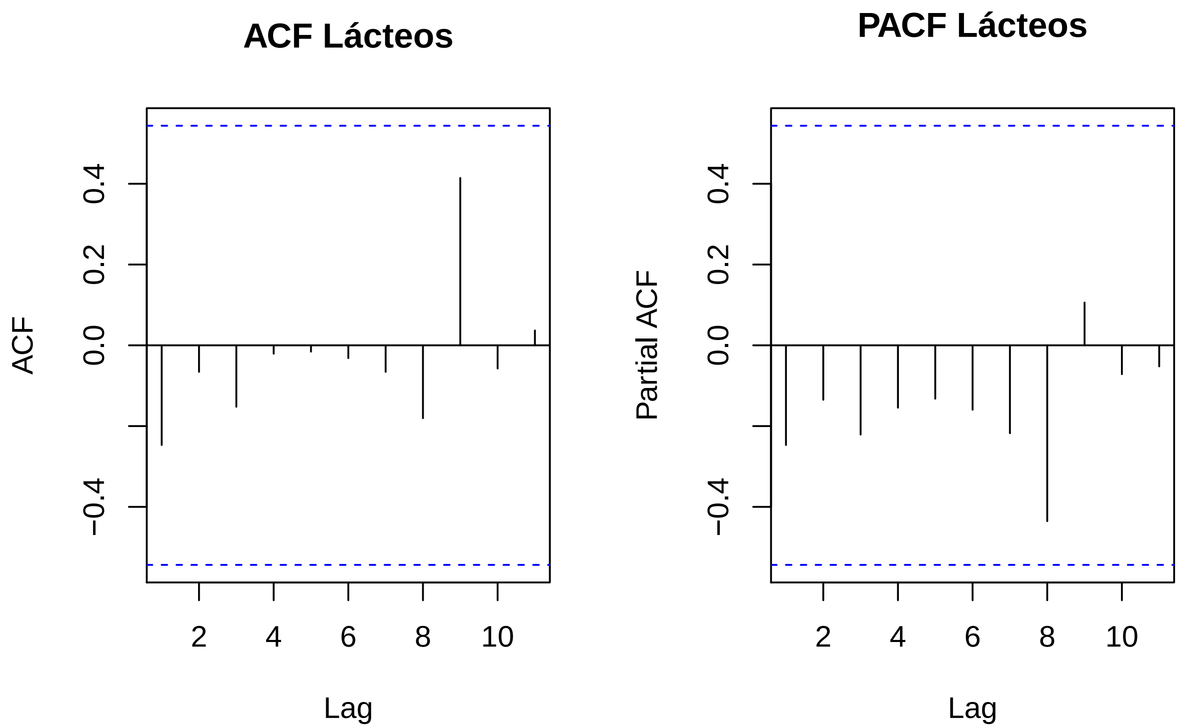
Funções de Autocorrelações para Avicultura de Postura

```r
#Funções de Autocorrelações para Avicultura de Postura
par(mfrow = c(1, 2))
acf(z_avp, main="ACF Avicultura de Postura")
pacf(z_avp, main="PACF Avicultura de Postura")
```



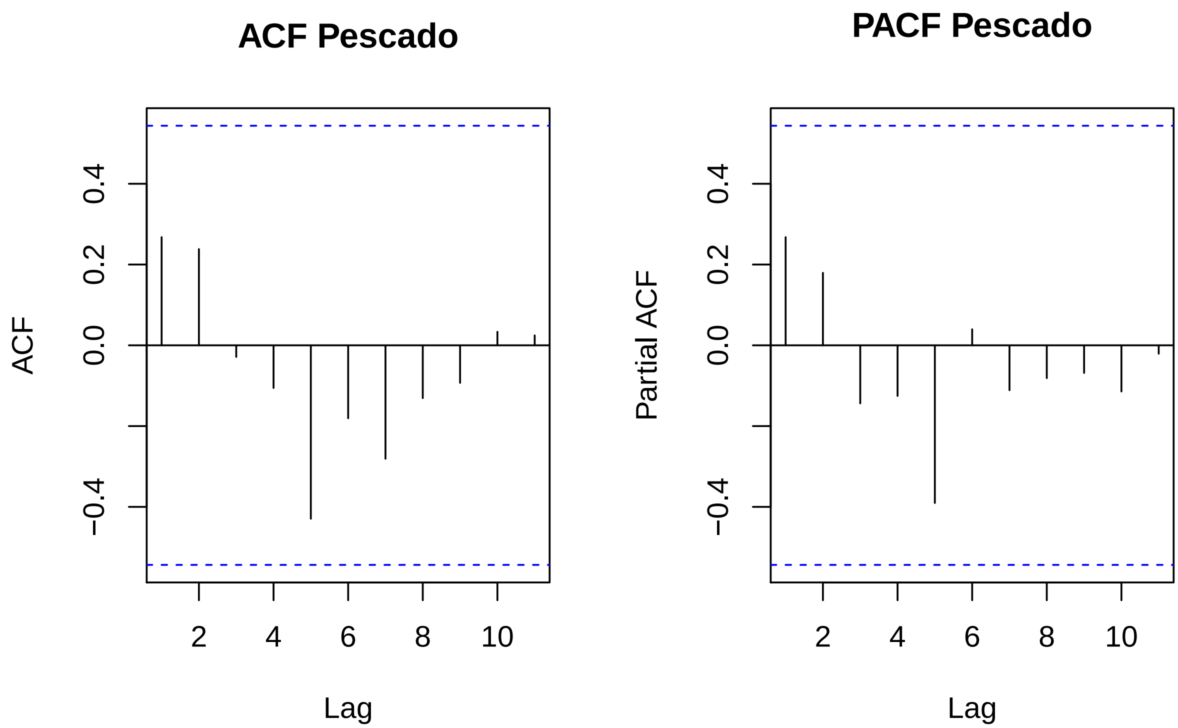
Funções de Autocorrelações para Lácteos

```
#Funções de Autocorrelações para Lácteos
par(mfrow = c(1, 2))
acf(z_lac, main="ACF Lácteos")
pacf(z_lac, main="PACF Lácteos")
```



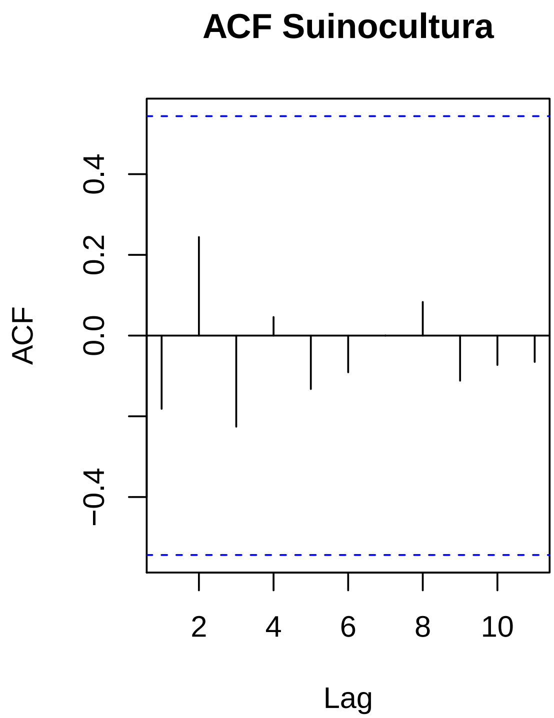
Funções de Autocorrelações para Pescado

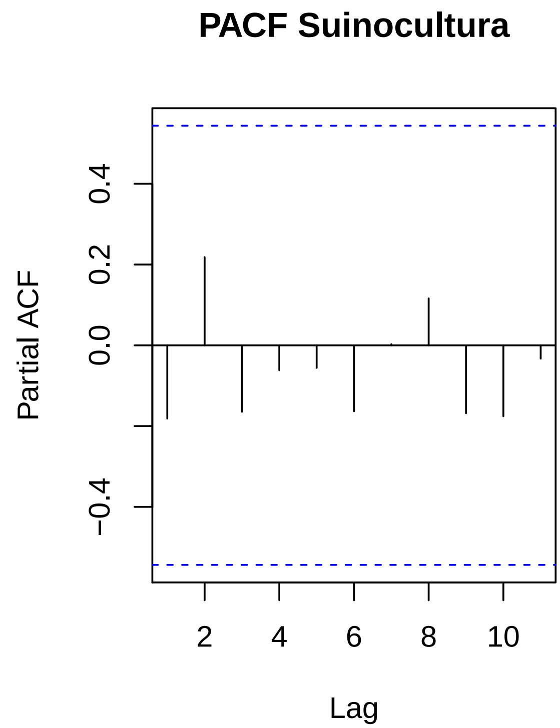
```r
#Funções de Autocorrelações para Pescado
par(mfrow = c(1, 2))
acf(z_pesc, main="ACF Pescado")
pacf(z_pesc, main="PACF Pescado")
```



Funções de Autocorrelações para Suinocultura

```
#Funções de Autocorrelações para Suinocultura
par(mfrow = c(1, 2))
acf(z_suino, main="ACF Suinocultura")
pacf(z_suino, main="PACF Suinocultura")
```





Testes de Dickey-Fuller e Phillips-Perron

Teste de Dickey-Fuller

```
# Teste de Dickey-Fuller
adf.test(z_bov)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  z_bov
## Dickey-Fuller = -2.4786, Lag order = 2, p-value = 0.3901
## alternative hypothesis: stationary
```

```
adf.test(z_avc)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  z_avc
## Dickey-Fuller = -1.9839, Lag order = 2, p-value = 0.5785
## alternative hypothesis: stationary
```

```r
adf.test(z_avp)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  z_avp
## Dickey-Fuller = -3.0526, Lag order = 2, p-value = 0.1714
## alternative hypothesis: stationary
```

```r
adf.test(z_lac)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  z_lac
## Dickey-Fuller = -1.8165, Lag order = 2, p-value = 0.6423
## alternative hypothesis: stationary
```

```r
adf.test(z_pesc)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  z_pesc
## Dickey-Fuller = -0.28347, Lag order = 2, p-value = 0.9843
## alternative hypothesis: stationary
```

```r
adf.test(z_suino)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  z_suino
## Dickey-Fuller = -3.3194, Lag order = 2, p-value = 0.08898
## alternative hypothesis: stationary
```

Teste de Phillips-Perron

```r
# Teste de Phillips-Perron
pp.test(z_bov)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  z_bov
## Dickey-Fuller Z(alpha) = -12.303, Truncation lag parameter = 2, p-value
## = 0.3209
## alternative hypothesis: stationary
```

```r
pp.test(z_avc)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  z_avc
## Dickey-Fuller Z(alpha) = -10.175, Truncation lag parameter = 2, p-value
## = 0.4635
## alternative hypothesis: stationary
```

```r
pp.test(z_avp)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  z_avp
## Dickey-Fuller Z(alpha) = -11.209, Truncation lag parameter = 2, p-value
## = 0.3942
## alternative hypothesis: stationary
```

```r
pp.test(z_lac)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  z_lac
## Dickey-Fuller Z(alpha) = -14.738, Truncation lag parameter = 2, p-value
## = 0.1577
## alternative hypothesis: stationary
```

```r
pp.test(z_pesc)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  z_pesc
## Dickey-Fuller Z(alpha) = -8.6126, Truncation lag parameter = 2, p-value
## = 0.5682
## alternative hypothesis: stationary
```

```r
pp.test(z_suino)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  z_suino
## Dickey-Fuller Z(alpha) = -17.616, Truncation lag parameter = 2, p-value
## = 0.05617
## alternative hypothesis: stationary
```

Definindo variáveis do modelo

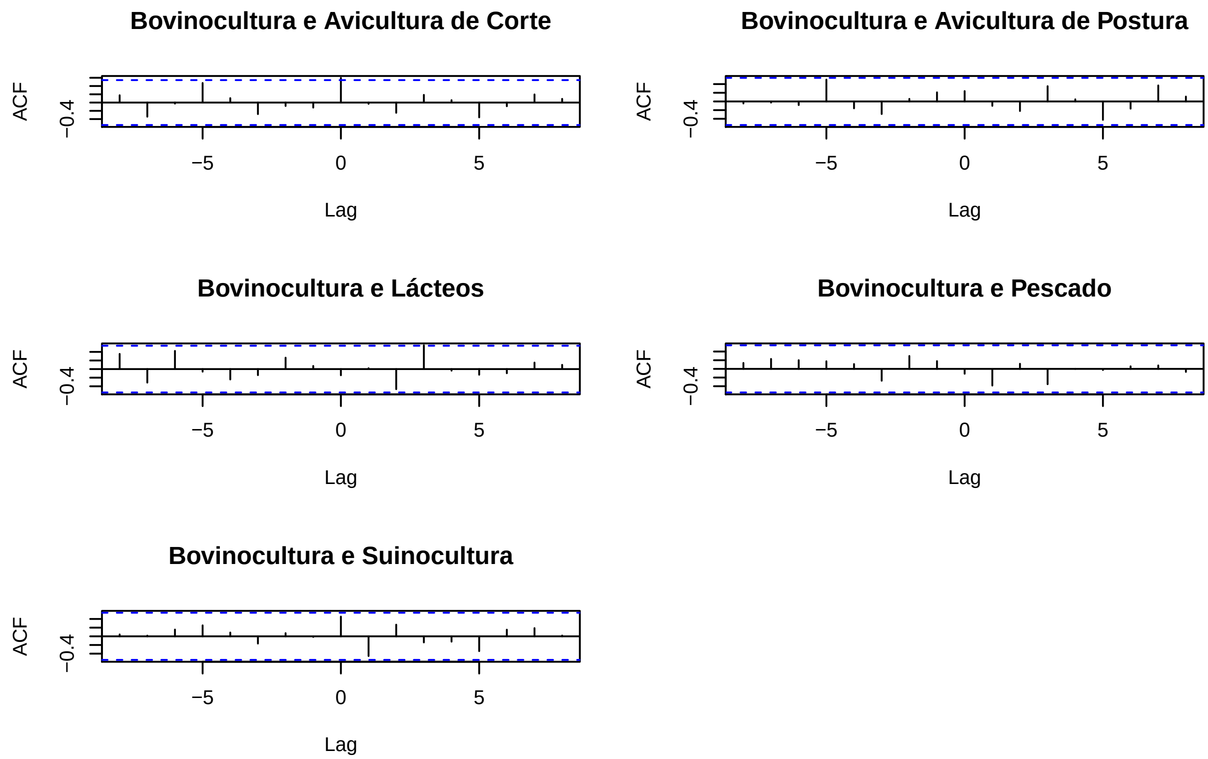
```r
# Variáveis do modelo
library(glmnet)

colnames(data_anual) = c("ANO", "AVC","AVP","BOV","LAC","PESC","SUIN")
data_anual = data_anual[,-1]
```

Regressão Lasso para Bovinocultura

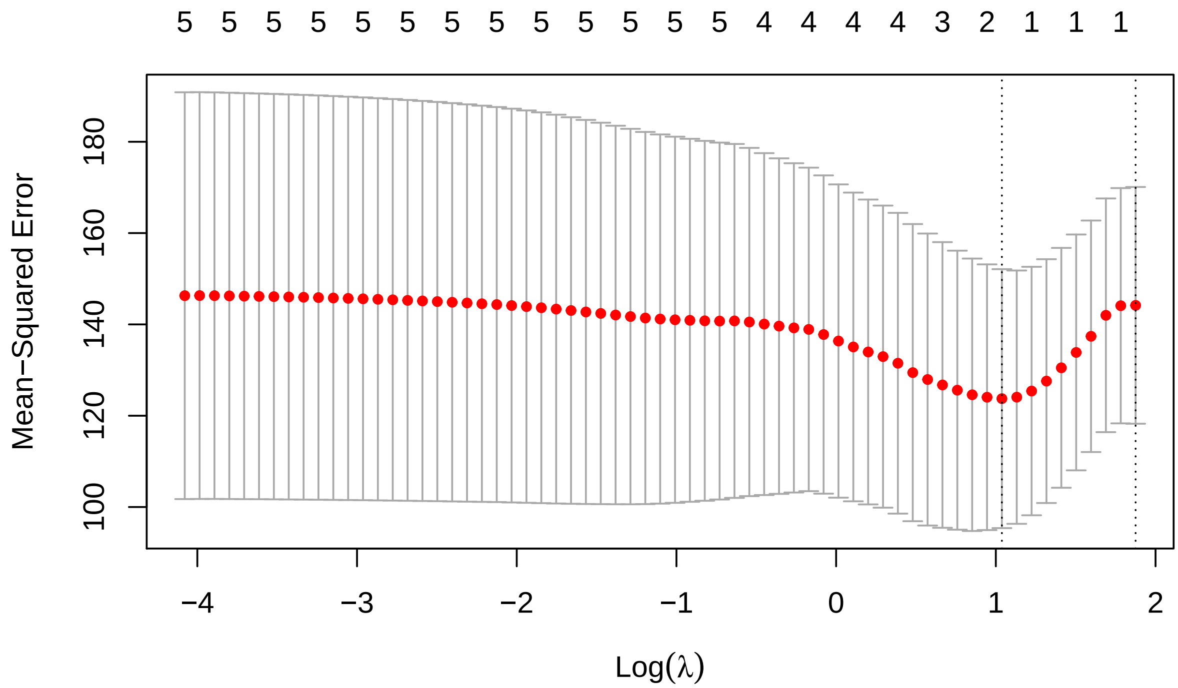
```r
#Correlaões cruzadas da Bovincultura
par(mfrow = c(3,2))
ccf(z_bov,z_avc,main="Bovinocultura e Avicultura de Corte")
ccf(z_bov,z_avp,main="Bovinocultura e Avicultura de Postura")
ccf(z_bov,z_lac,main="Bovinocultura e Lácteos")
ccf(z_bov,z_pesc,main="Bovinocultura e Pescado")
ccf(z_bov,z_suino,main="Bovinocultura e Suinocultura")


# Regressão LASSO
set.seed(1)
x = model.matrix(BOV~ .,data=data_anual)[,-1]
y = data_anual$BOV

cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
```

```
##
## Call:  cv.glmnet(x = x, y = y, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure    SE Nonzero
## min  2.823    10   123.7 28.36       2
## 1se  6.522     1   144.2 25.92       0
```

```r
par(mfrow=c(1,1))
```




```r
plot(cv.model)
```




```
coef(cv.model,cv.model$lambda.min)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                      1
## (Intercept) 6.24941948
## AVC         0.67510339
## AVP         .
## LAC         .
## PESC        .
## SUIN        0.03047208
```

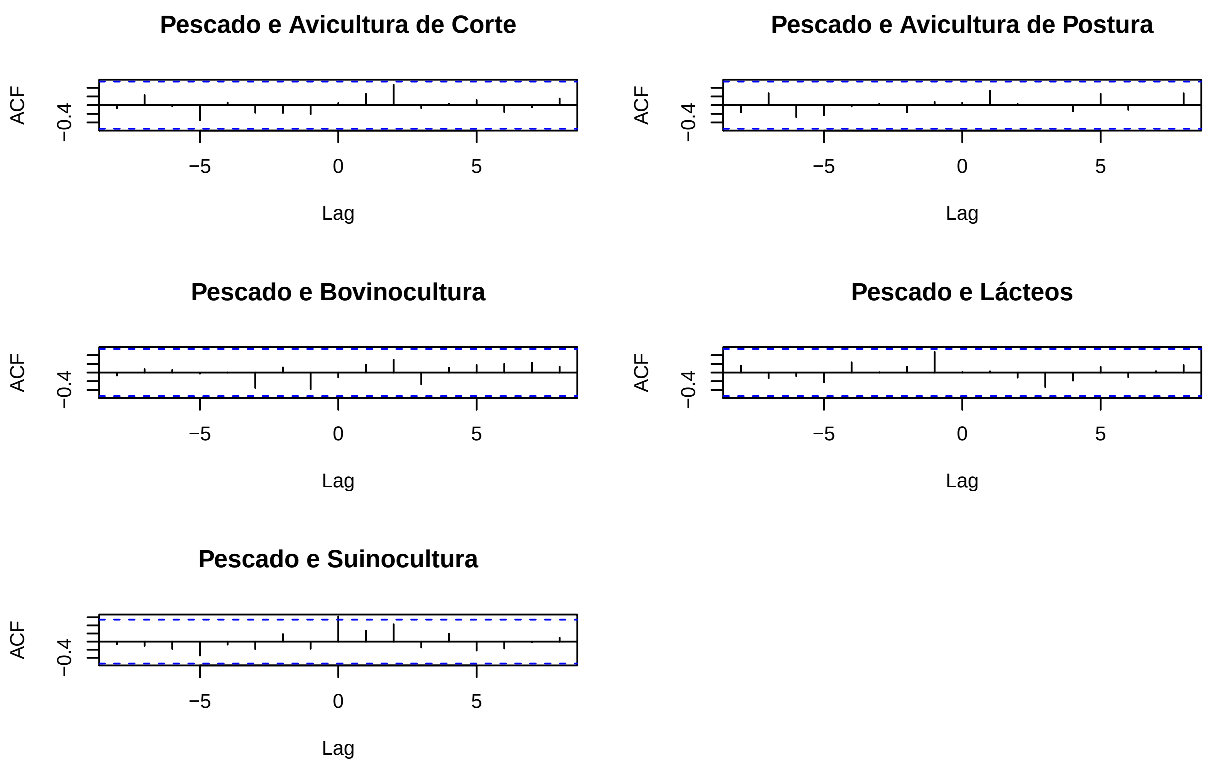
Regressão Lasso para o Pescado

```
# Pescados

par(mfrow = c(3,2))
ccf(z_pesc,z_avc,main="Pescado e Avicultura de Corte")
ccf(z_pesc,z_avp,main="Pescado e Avicultura de Postura")
ccf(z_pesc,z_bov,main="Pescado e Bovinocultura")
ccf(z_pesc,z_lac,main="Pescado e Lácteos")
ccf(z_pesc,z_suino,main="Pescado e Suinocultura")



# Regressão LASSO
```

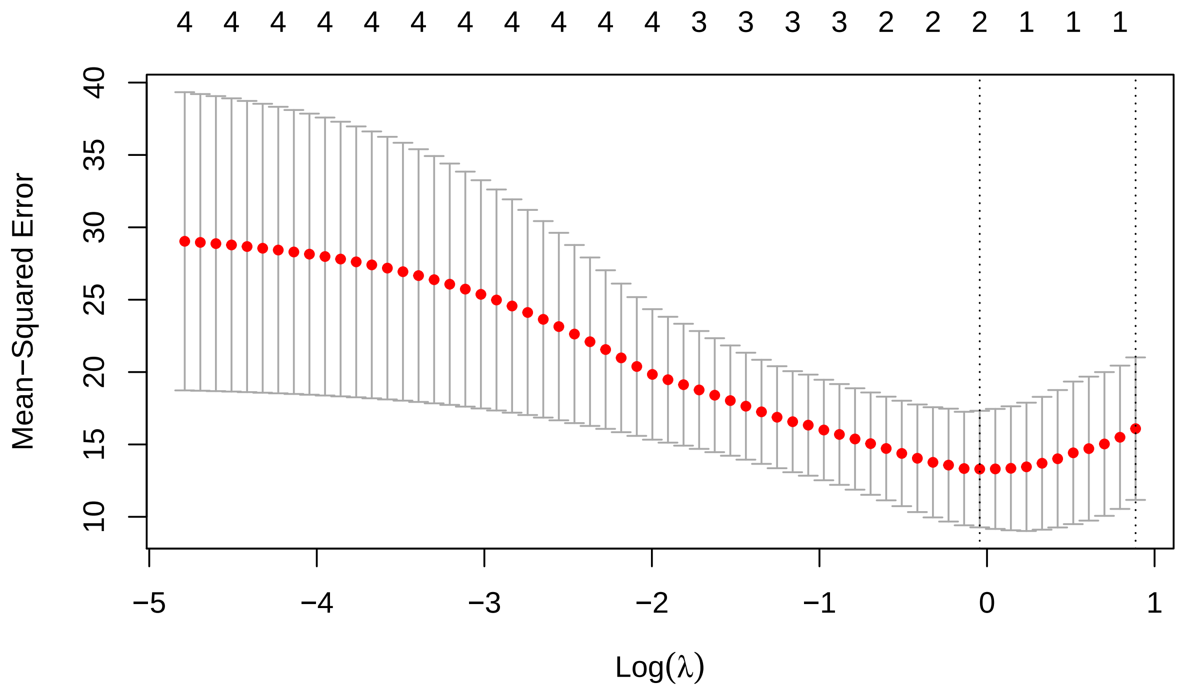
```
set.seed(2)
x = model.matrix(PESC~ .,data=data_anual)[,-1]
y = data_anual$PESC

cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
```

```
##
## Call:  cv.glmnet(x = x, y = y, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure    SE Nonzero
## min 0.9574    11   13.30 4.025       2
## 1se 2.4274     1   16.09 4.922       0
```

```
par(mfrow=c(1,1))
```




```
plot(cv.model)
```




```r
coef(cv.model,cv.model$lambda.min)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept)  5.10009457
## AVC          .
## AVP          .
## BOV         -0.01742377
## LAC          .
## SUIN         0.35756988
```

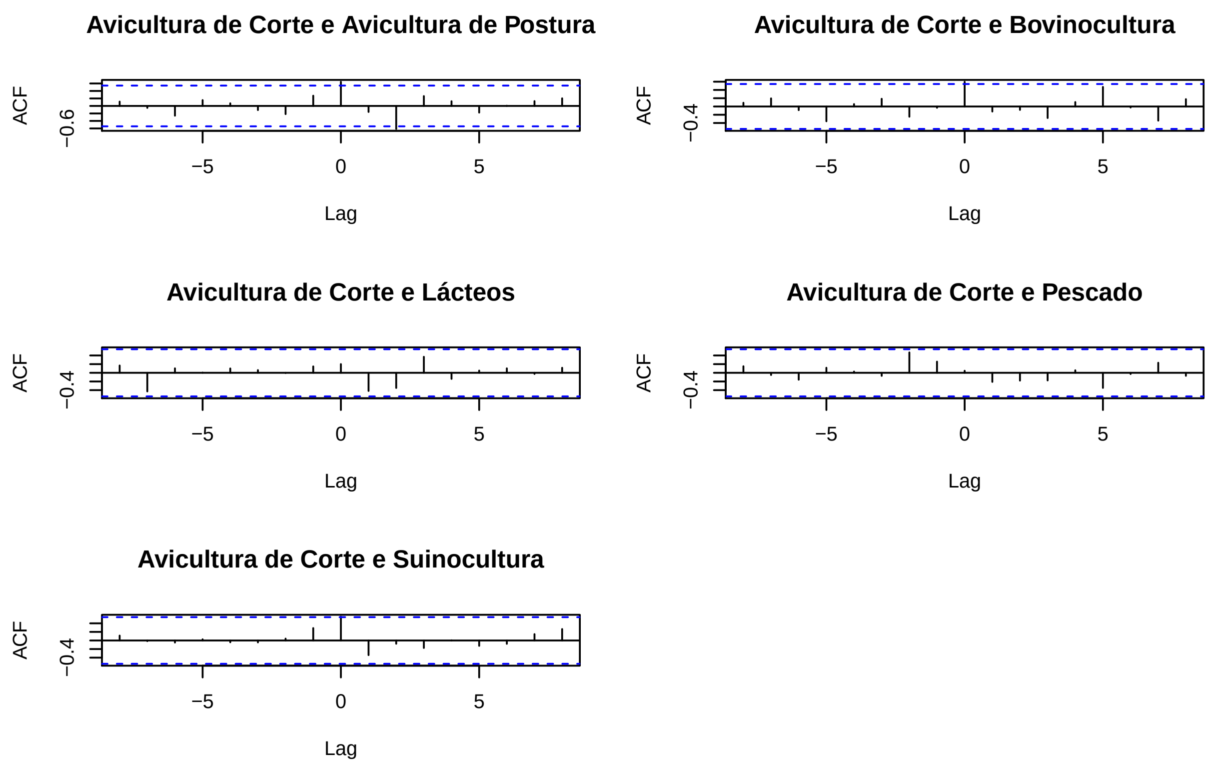
Regressão Lasso para a Avicultura de Corte

```r
# Avicultura de Corte

par(mfrow = c(3,2))
ccf(z_avc,z_avp,main="Avicultura de Corte e Avicultura de Postura")
ccf(z_avc,z_bov,main="Avicultura de Corte e Bovinocultura")
ccf(z_avc,z_lac,main="Avicultura de Corte e Lácteos")
ccf(z_avc,z_pesc,main="Avicultura de Corte e Pescado")
ccf(z_avc,z_suino,main="Avicultura de Corte e Suinocultura")


# Regressão LASSO
```

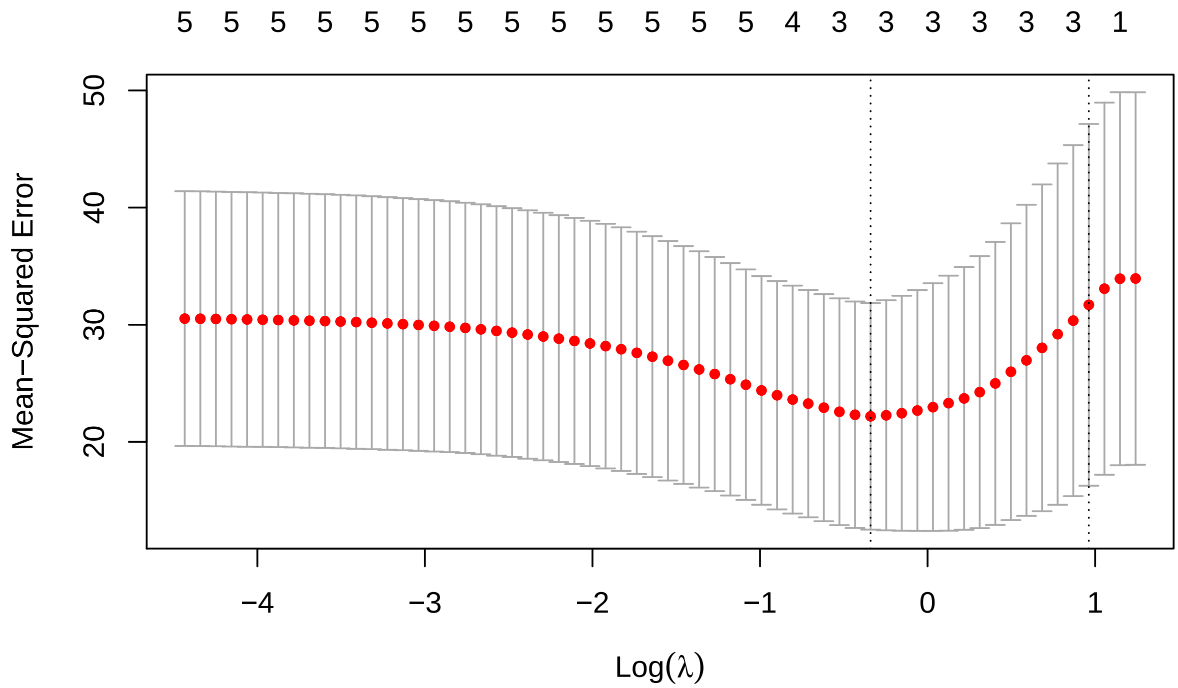
```r
set.seed(3)
x = model.matrix(AVC~ .,data=data_anual)[,-1]
y = data_anual$AVC

cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
```

```
##
## Call:  cv.glmnet(x = x, y = y, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure     SE Nonzero
## min 0.7118    18   22.18  9.671       3
## 1se 2.6183     4   31.70 15.445       2
```

```r
par(mfrow=c(1,1))
```




```r
plot(cv.model)
```



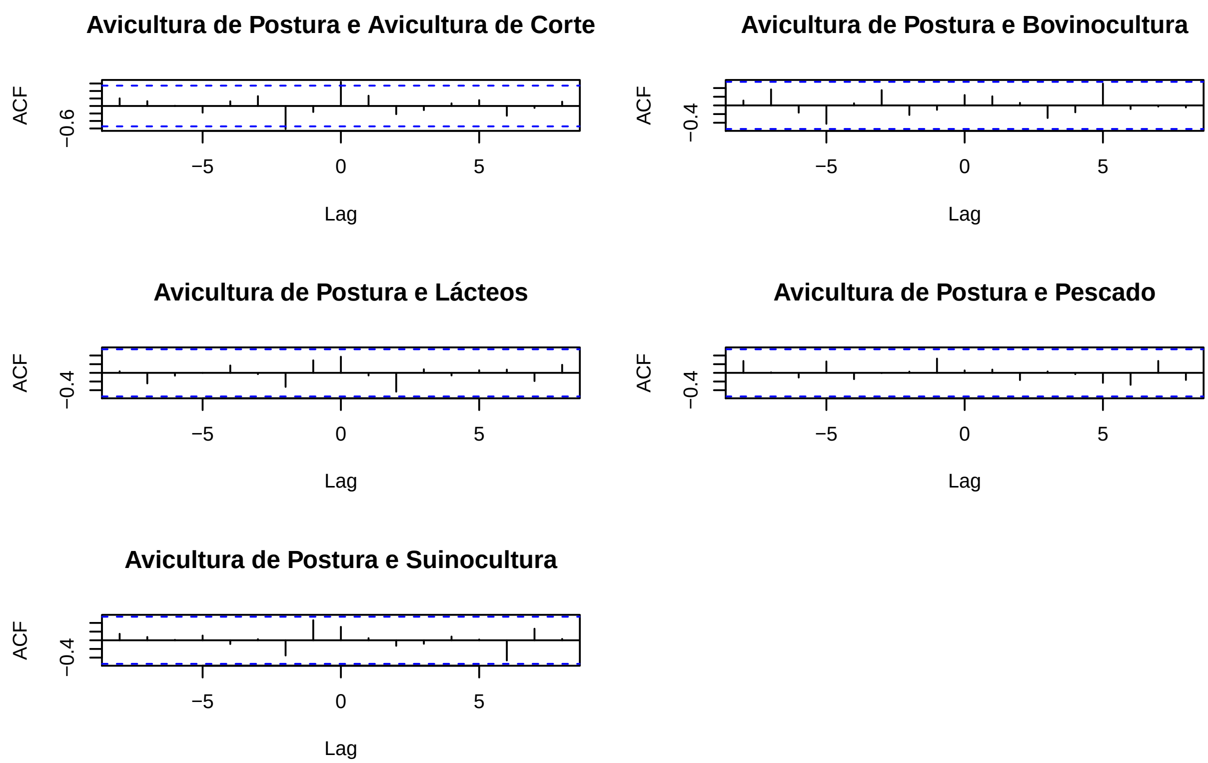

```
coef(cv.model,cv.model$lambda.min)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                     1
## (Intercept) 1.1062106
## AVP         0.2630295
## BOV         0.1454421
## LAC         .
## PESC        .
## SUIN        0.2064983
```

Regressão Lasso para Avicultura de Postura

```
# Avicultura de Postura

par(mfrow = c(3,2))
ccf(z_avp,z_avc,main="Avicultura de Postura e Avicultura de Corte")
ccf(z_avp,z_bov,main="Avicultura de Postura e Bovinocultura")
ccf(z_avp,z_lac,main="Avicultura de Postura e Lácteos")
ccf(z_avp,z_pesc,main="Avicultura de Postura e Pescado")
ccf(z_avp,z_suino,main="Avicultura de Postura e Suinocultura")


# Regressão LASSO
```

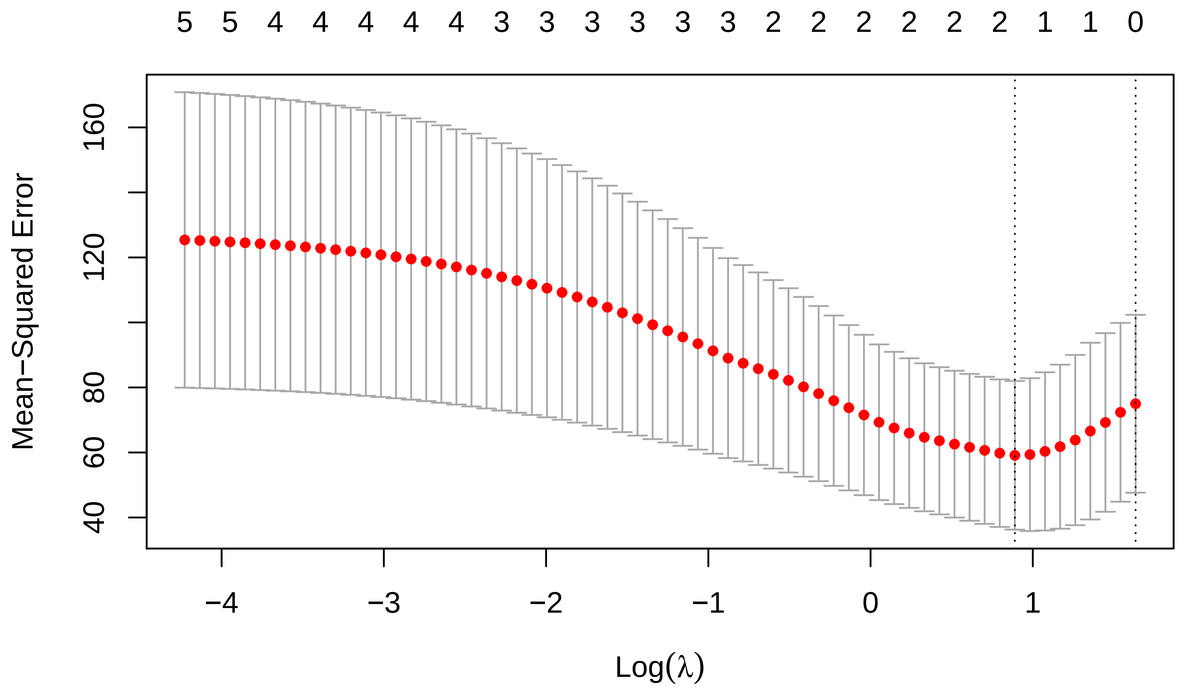
```r
set.seed(4)
x = model.matrix(AVP~ .,data=data_anual)[,-1]
y = data_anual$AVP

par(mfrow=c(1,1))
```




```r
cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
```

```
##
## Call:  cv.glmnet(x = x, y = y, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure    SE Nonzero
## min  2.434     9   59.12 22.85       1
## 1se  5.123     1   74.99 27.37       0
```

```r
plot(cv.model)
```



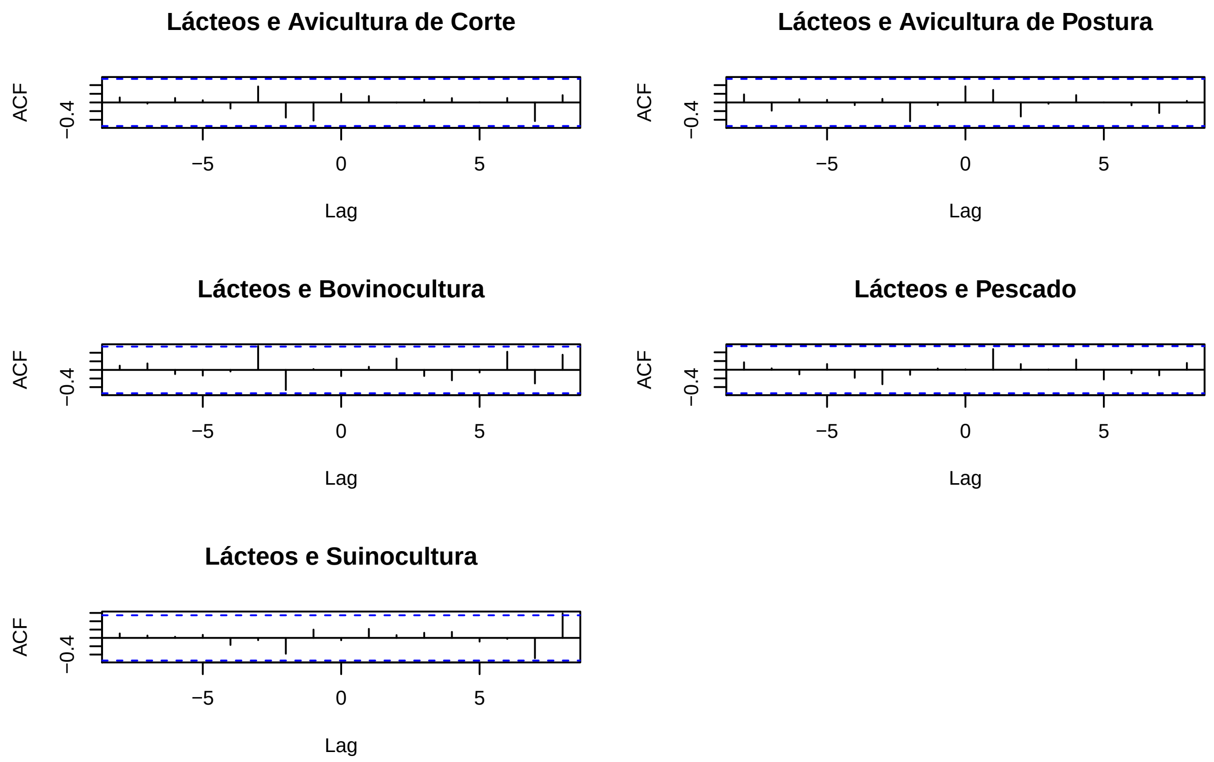

```
coef(cv.model,cv.model$lambda.min)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                     1
## (Intercept) 6.3392348
## AVC         0.5007544
## BOV         .
## LAC         .
## PESC        .
## SUIN        .
```

Regressão Lasso para o Lácteos

```
# Lacteos
par(mfrow = c(3,2))
ccf(z_lac,z_avc,main="Lácteos e Avicultura de Corte")
ccf(z_lac,z_avp,main="Lácteos e Avicultura de Postura")
ccf(z_lac,z_bov,main="Lácteos e Bovinocultura")
ccf(z_lac,z_pesc,main="Lácteos e Pescado")
ccf(z_lac,z_suino,main="Lácteos e Suinocultura")


# Regressão LASSO
set.seed(5)
```

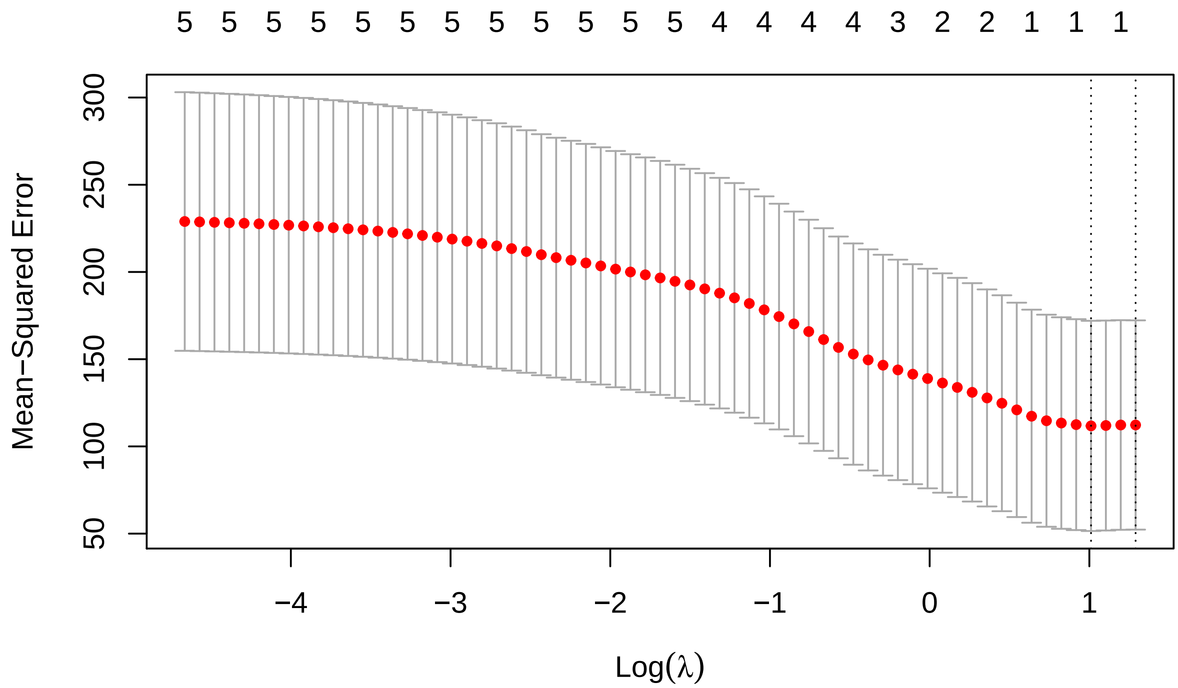
```r
x = model.matrix(LAC~ .,data=data_anual)[,-1]
y = data_anual$LAC

par(mfrow=c(1,1))
```




```r
cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
```

```
##
## Call:  cv.glmnet(x = x, y = y, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure    SE Nonzero
## min  2.747     4   111.8 60.25       1
## 1se  3.631     1   112.3 59.97       0
```

```r
plot(cv.model)
```



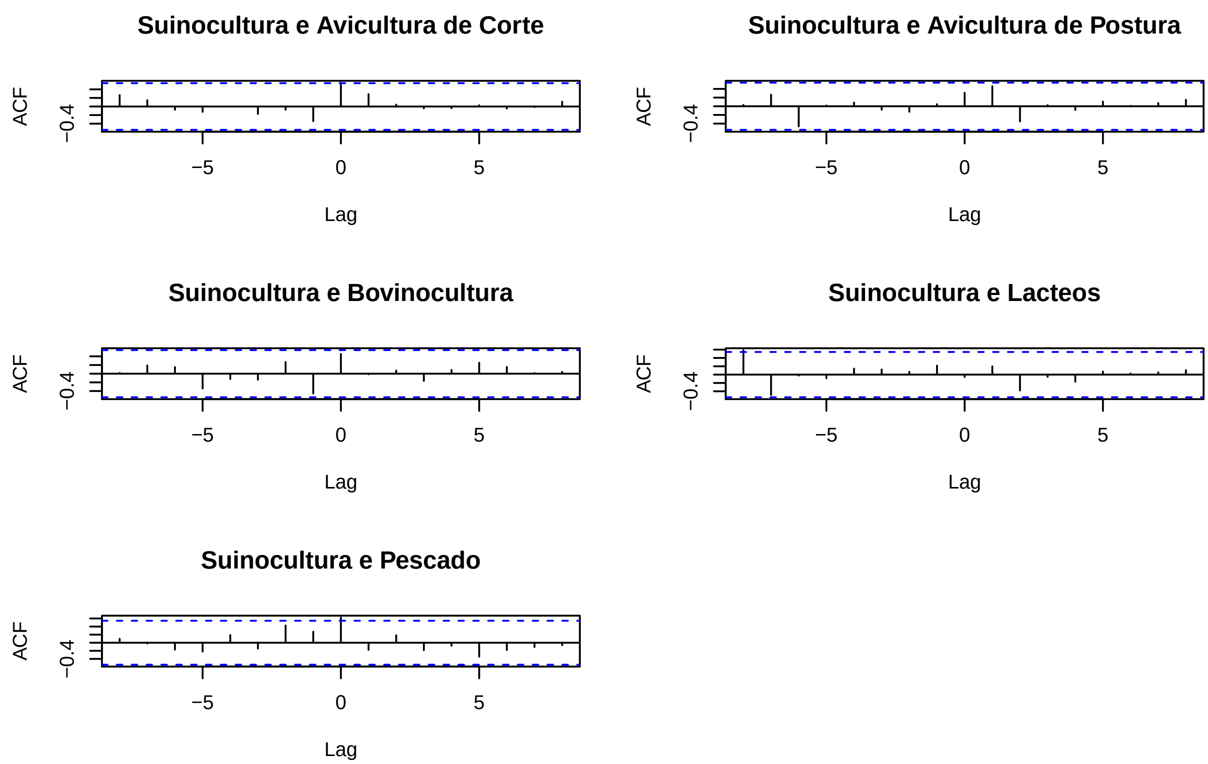

```
coef(cv.model,cv.model$lambda.min)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                     1
## (Intercept) 6.5890827
## AVC         .
## AVP         0.1112567
## BOV         .
## PESC        .
## SUIN        .
```

Regressão Lasso para Suinocultura

```
# Suinocultura


par(mfrow = c(3,2))
ccf(z_suino,z_avc,main="Suinocultura e Avicultura de Corte")
ccf(z_suino,z_avp,main="Suinocultura e Avicultura de Postura")
ccf(z_suino,z_bov,main="Suinocultura e Bovinocultura")
ccf(z_suino,z_lac,main="Suinocultura e Lacteos")
ccf(z_suino,z_pesc,main="Suinocultura e Pescado")
```

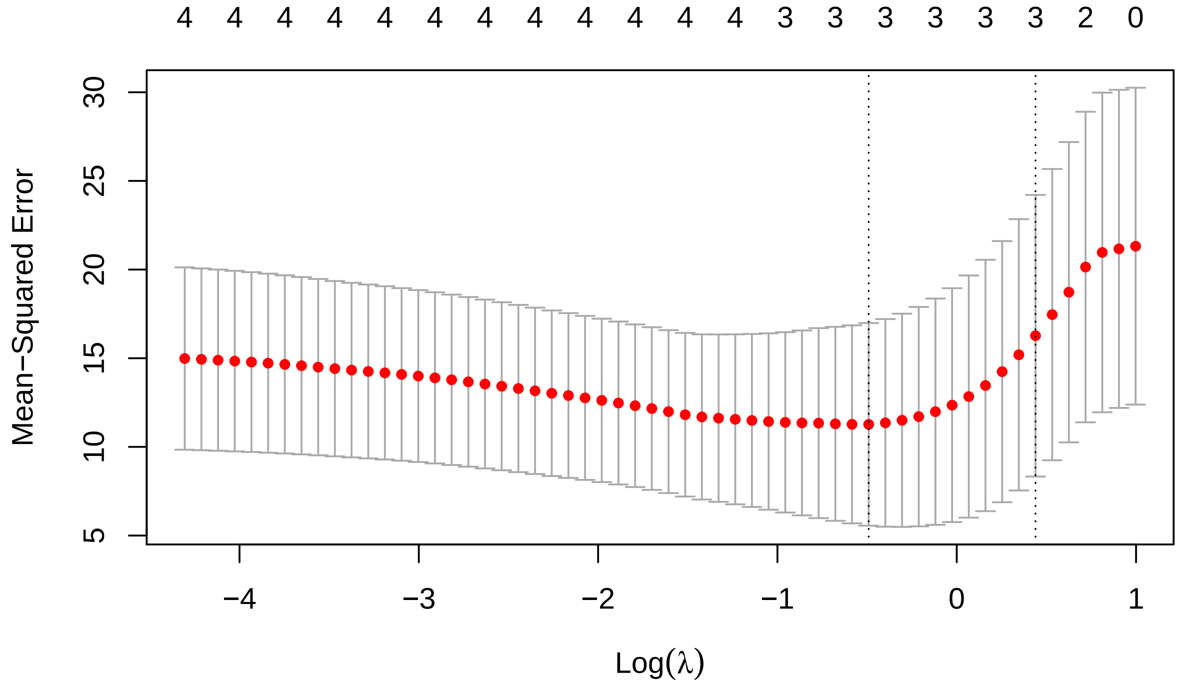
```r
# Regressão LASSO
set.seed(6)
x = model.matrix(SUIN~ .,data=data_anual)[,-1]
y = data_anual$SUIN

par(mfrow=c(1,1))
```




```r
cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
```

```
##
## Call:  cv.glmnet(x = x, y = y, alpha = 1)
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure    SE Nonzero
## min  0.612    17   11.27 5.713       3
## 1se  1.552     7   16.27 7.939       3
```

```r
plot(cv.model)
```




```r
coef(cv.model,cv.model$lambda.min)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept) -0.33554640
## AVC          0.21438530
## AVP          .
## BOV          0.08463707
## LAC          .
## PESC         0.55149018
```

```r
# paramentros padrao para os plots
knitr::opts_chunk$set(fig.width = '\\textwidth',
                      fig.align = 'center',
                      out.width = "\\textwidth",
                      warning = FALSE, echo = TRUE)


library(robustbase)
library(knitr)
library(stargazer)
library(tidyverse)
library(car)
library(readxl)
library(MASS)
library(gridExtra)
library(ggplot2)
```

```
library(faraway)
require(BatchGetSymbols)
require(Amelia)
require(reshape2)
require(ggthemes)
require(plyr)
library(ggplot2)

suppressMessages(library(foreign))
suppressMessages(library(dynlm))
suppressMessages(library(car))
suppressMessages(library(lmtest))
suppressMessages(library(sandwich))
suppressMessages(library(fpp2))
suppressMessages(library(tseries))
suppressMessages(library(zoo))
suppressMessages(library(forecast))
library(BETS)
library(quantmod)
library(fpp2)
library(GeneCycle)
library(randtests)
library(zoo)
library(TSA)
library(gridExtra)
library(FitAR)
library(glmnet)
library(astsa)
library(lmtest)
data = read_xlsx("IPCA_DADOS_AGRUPADOS.xlsx", sheet = 1)
data$Data <- as.Date(data$Data)
head(data)
zt2 <- ts(data[,2], frequency = 12, start = 2007, end = 2019)
zt3 <- ts(data[,3], frequency = 12, start = 2007, end = 2019)
zt4 <- ts(data[,4], frequency = 12, start = 2007, end = 2019)
zt5 <- ts(data[,5], frequency = 12, start = 2007, end = 2019)
zt6 <- ts(data[,6], frequency = 12, start = 2007, end = 2019)
zt7 <- ts(data[,7], frequency = 12, start = 2007, end = 2019)
zt8 <- ts(data[,8], frequency = 12, start = 2007, end = 2019)
zt9 <- ts(data[,9], frequency = 12, start = 2007, end = 2019)
zt10 <- ts(data[,10], frequency = 12, start = 2007, end = 2019)
zt11 <- ts(data[,11], frequency = 12, start = 2007, end = 2019)

zt12 <- ts(data[,12], frequency = 12, start = 2007, end = 2019)
zt13 <- ts(data[,13], frequency = 12, start = 2007, end = 2019)
zt14 <- ts(data[,14], frequency = 12, start = 2007, end = 2019)
zt15 <- ts(data[,15], frequency = 12, start = 2007, end = 2019)
zt16 <- ts(data[,16], frequency = 12, start = 2007, end = 2019)
zt17 <- ts(data[,17], frequency = 12, start = 2007, end = 2019)
zt18 <- ts(data[,18], frequency = 12, start = 2007, end = 2019)
zt19 <- ts(data[,19], frequency = 12, start = 2007, end = 2019)
zt20 <- ts(data[,20], frequency = 12, start = 2007, end = 2019)
zt21 <- ts(data[,21], frequency = 12, start = 2007, end = 2019)
```

```
zt22 <- ts(data[,22], frequency = 12, start = 2007, end = 2019)
zt23 <- ts(data[,23], frequency = 12, start = 2007, end = 2019)
zt24 <- ts(data[,24], frequency = 12, start = 2007, end = 2019)

plot(zt2,main="Série Temporal do Arroz", xlab= "Anos", ylab="IPCA")
#par(mfrow = c(2, 2))
plot(zt3,main="Série Temporal de Avicultura de Corte", xlab= "Anos", ylab="IPCA")
plot(zt4,main="Série Temporal de Avicultura de Postura", xlab= "Anos", ylab="IPCA")
plot(zt5,main="Série Temporal da Banana", xlab= "Anos", ylab="IPCA")
plot(zt6,main="Série Temporal da Batata", xlab= "Anos", ylab="IPCA")

#par(mfrow = c(3, 2))


plot(zt7,main="Série Temporal da Bovinocultura", xlab= "Anos", ylab="IPCA")
plot(zt8,main="Série Temporal do Cacau e Produtos", xlab= "Anos", ylab="IPCA")
plot(zt9,main="Série Temporal do Café", xlab= "Anos", ylab="IPCA")
plot(zt10,main="Série Temporal da Cebola", xlab= "Anos", ylab="IPCA")
plot(zt11,main="Série Temporal do Complexo Soja", xlab= "Anos", ylab="IPCA")
plot(zt12,main="Série Temporal do Complexo Sucroalc.", xlab= "Anos", ylab="IPCA")

#par(mfrow = c(3, 2))


plot(zt13,main="Série Temporal do Feijão", xlab= "Anos", ylab="IPCA")
plot(zt14,main="Série Temporal das Frutas", xlab= "Anos", ylab="IPCA")
plot(zt15,main="Série Temporal das Horticulas", xlab= "Anos", ylab="IPCA")
plot(zt16,main="Série Temporal de Indefinido", xlab= "Anos", ylab="IPCA")
plot(zt17,main="Série Temporal do Laranja e Citrus", xlab= "Anos", ylab="IPCA")
plot(zt18,main="Série Temporal da Lácteos", xlab= "Anos", ylab="IPCA")

#par(mfrow = c(3, 2))


plot(zt19,main="Série Temporal da Mandioca", xlab= "Anos", ylab="IPCA")
plot(zt20,main="Série Temporal do Milho", xlab= "Anos", ylab="IPCA")
plot(zt21,main="Série Temporal do Pescado", xlab= "Anos", ylab="IPCA")
plot(zt22,main="Série Temporal da Suínocultura", xlab= "Anos", ylab="IPCA")
plot(zt23,main="Série Temporal do Tomate", xlab= "Anos", ylab="IPCA")
plot(zt24,main="Série Temporal do Trigo", xlab= "Anos", ylab="IPCA")
par(mfrow = c(2, 1))
plot(zt21,main="Série Temporal do Pescado", xlab= "Anos", ylab="IPCA")
plot(zt18,main="Série Temporal do Lácteos", xlab= "Anos", ylab="IPCA")
#900#650
par(mfrow = c(2, 1))
plot(zt7,main="Série Temporal da Bovinocultura", xlab= "Anos", ylab="IPCA")
plot(zt22,main="Série Temporal da Suínocultura", xlab= "Anos", ylab="IPCA")

par(mfrow = c(2, 1))
plot(zt3,main="Série Temporal de Avicultura de Corte", xlab= "Anos", ylab="IPCA")
plot(zt4,main="Série Temporal de Avicultura de Postura", xlab= "Anos", ylab="IPCA")

#Funções de Autocorrelações para Avicultura de Corte
```

```r
par(mfrow = c(1, 2))
acf(zt3, main="ACF Avicultura de Corte")
pacf(zt3, main="PACF Avicultura de Corte")
#Funções de Autocorrelações para Avicultura de Postura
par(mfrow = c(1, 2))
acf(zt4, main="ACF Avicultura de Postura")
pacf(zt4, main="PACF Avicultura de Postura")
#Funções de Autocorrelações para Suinocultura
par(mfrow = c(1, 2))
acf(zt22, main="ACF Suínocultura")
pacf(zt22, main="PACF Suínocultura")
#Funções de Autocorrelações para Pescado
par(mfrow = c(1, 2))
acf(zt21, main="ACF Pescado", lag.max = 36)
pacf(zt21, main="PACF Pescado", lag.max = 36)
#Funções de Autocorrelações para Lácteos
par(mfrow = c(1, 2))
acf(zt18, main="ACF Lácteos", lag.max = 48)
pacf(zt18, main="PACF Lácteos", lag.max = 48)
#Funções de Autocorrelações para Bovinocultura
par(mfrow = c(1, 2))
acf(zt7, main="ACF Bovinocultura")
pacf(zt7, main="PACF Bovinocultura")
# Teste de Dickey-Fuller
adf.test(zt7)
adf.test(zt3)
adf.test(zt4)
adf.test(zt18)
adf.test(zt21)
adf.test(zt22)

# Teste de Phillips-Perron
pp.test(zt7)
pp.test(zt3)
pp.test(zt4)
pp.test(zt18)
pp.test(zt21)
pp.test(zt22)

#Correlaões cruzadas da Bovincultura
par(mfrow = c(3,2))
ccf(zt7,zt3,main="Bovinocultura e Avicultura de Corte")
ccf(zt7,zt4,main="Bovinocultura e Avicultura de Postura")
ccf(zt7,zt18,main="Bovinocultura e Lácteos")
ccf(zt7,zt21,main="Bovinocultura e Pescados")
ccf(zt7,zt22,main="Bovinocultura e Suinocultura")
#Correlações cruzadas da Avicultura de Corte
par(mfrow = c(3,2))
ccf(zt3,zt4,main="Avicultura de Corte e Avicultura de Postura")
ccf(zt3,zt7,main="Avicultura de Corte e Bovinocultura")
ccf(zt3,zt18,main="Avicultura de Corte e Lácteos")
ccf(zt3,zt21,main="Avicultura de Corte e Pescados")
ccf(zt3,zt22,main="Avicultura de Corte e Suinocultura")
```

```r
#Correlações cruzadas da Avicultura de Postura
par(mfrow = c(3,2))
ccf(zt4,zt3,main="Avicultura de Postura e Avicultura de Corte")
ccf(zt4,zt7,main="Avicultura de Postura e Bovinocultura")
ccf(zt4,zt18,main="Avicultura de Postura e Lácteos")
ccf(zt4,zt21,main="Avicultura de Postura e Pescados")
ccf(zt4,zt22,main="Avicultura de Postura e Suinocultura")
#Correlações cruzadas dos Lácteos
par(mfrow = c(3,2))
ccf(zt18,zt3,main="Lácteos e Avicultura de Corte")
ccf(zt18,zt4,main="Lácteos e Avicultura de Postura ")
ccf(zt18,zt7,main="Lácteos e Bovinocultura")
ccf(zt18,zt21,main="Lácteos e Pescados")
ccf(zt18,zt22,main="Lácteos e Suinocultura")
# Correlaões cruzadas dos Pescados
par(mfrow = c(3,2))
ccf(zt21,zt3,main="Pescados e Avicultura de Corte")
ccf(zt21,zt4,main="Pescados e Avicultura de Postura")
ccf(zt21,zt7,main="Pescados e Bovinocultura")
ccf(zt21,zt18,main="Pescados e Lácteos")
ccf(zt21,zt22,main="Pescados e Suinocultura")

#Correlações cruzadas da Suinocultura
par(mfrow = c(3,2))
ccf(zt22,zt3,main="Suinocultura e Avicultura de Corte")
ccf(zt22,zt4,main="Suinocultura e Avicultura de Postura")
ccf(zt22,zt7,main="Suinocultura e Bovinocultura")
ccf(zt22,zt18,main="Suinocultura e Lacteos")
ccf(zt22,zt21,main="Suinocultura e Pescados")
#Essa função retorna a coluna com a lag a ser considerada na análise

funcao_lags = function(df,coluna,nome,lag){
  n = nrow(df)
  pre = rep(NA,lag)
  newcol = c(pre,coluna)
  for (k in 1:lag){
    df = rbind(df,rep(NA,ncol(df)))
  }
  df[nome] = newcol
  return (df)
}
#A função a baixo retira as variáveis do modelo em função do p-valor
tirar_variaveis = function(p,d,q,x,y){
  v = p + q + 1
  max = 0.06
  while (max > 0.05){
    model = Arima(y,order=c(p,d,q),xreg = x)
    ct = coeftest(model)
    pvalues = ct[(v+1):nrow(ct),4]
    maxi = which.max(pvalues)
    max = ct[v + maxi,4]
    if (max > 0.05) {
```

```r
      x = x[,-maxi]
      }
  }
  lista = list(ct, x)
  return (lista)
}
#A seguir vamos selecionar apenas as variáveis de interesse para análise
data_cut = data[,c("Bovinocultura","Avicultura de Corte","Avicultura de Postura","Pescado","Lácteos","Su
#Estruturando a base
df1<- funcao_lags(data_cut, data_cut$`Avicultura de Postura`, 'avp9', 9)
df1 <- funcao_lags(df1, df1$Pescado, 'p3', 3)
df1 <- funcao_lags(df1, df1$Pescado, 'p10', 10)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'b1', 1)


df2 <- na.omit(df1)
#Separando variável preditora e as covariáveis
x = model.matrix(Bovinocultura~.,df2)[,-1]
y = df2$Bovinocultura
#Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1

#Análise dos Resíduos
acf2(resid(fit))
#Análise dos resíduos e seleção de variáveis de acordo com p-valor
fit2 <- tirar_variaveis(0, 0, 0, x, y)
fit2[[1]]
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx)
fit3
coeftest(fit3)
acf2(fit3$residuals)


fit4 = Arima(y,order=c(1,0,0),xreg=xx)
fit4
coeftest(fit4)
checkresiduals(fit4)
acf2(fit4$residuals)



fit5 <- tirar_variaveis(1, 0, 0, xx, y)
fit5[[1]]
xx <- fit5[2]
xx<- xx[[1]]
```

```r
fit6 = Arima(y,order=c(1,0,0),xreg=xx,fixed=c(NA,NA, NA, NA))
cof.fit6 = coeftest(fit6)
cof.fit6
checkresiduals(fit6)
acf2(fit6$residuals, main = "")



#Estruturando a base
df1<- funcao_lags(data_cut, data_cut$`Avicultura de Corte`, 'cort1', 1)
df1 <- funcao_lags(df1, df1$`Avicultura de Postura`, 'pos12', 12)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov1', 1)
df1 <- funcao_lags(df1, df1$Pescado, 'pes4', 4)
df1 <- funcao_lags(df1, df1$Pescado, 'pes9', 9)
df1 <- funcao_lags(df1, df1$Suinocultura, 'sui1', 1)
df1 <- funcao_lags(df1, df1$Suinocultura, 'sui6', 6)

df2 <- na.omit(df1)
#Separando variável preditora e as covariáveis
x = model.matrix(`Avicultura de Corte`~.,df2)[,-1]
y = df2$`Avicultura de Corte`
#Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1

#Análise dos Resíduos
acf2(resid(fit))
#Seleção de variáveis
fit2 <- tirar_variaveis(0, 0, 0, x, y)
xx <- fit2[2]
xx <- xx[[1]]

fit3 = Arima(y,order=c(0,0,0), include.mean = FALSE, xreg=xx)
fit3
coeftest(fit3)
checkresiduals(fit3)
acf2(fit3$residuals, main = "")

# Estruturando a base
df1<- funcao_lags(data_cut, data_cut$Pescado, 'pes1', 1)
df1 <- funcao_lags(df1, df1$Pescado, 'pes5', 5)
df1 <- funcao_lags(df1, df1$Pescado, 'pes12', 12)

df1 <- funcao_lags(df1, df1$`Avicultura de Corte`, 'cort3', 3)
df1 <- funcao_lags(df1, df1$`Avicultura de Corte`, 'cort8', 8)


df1 <- funcao_lags(df1, df1$`Avicultura de Postura`, 'pos2', 2)
df1 <- funcao_lags(df1, df1$`Avicultura de Postura`, 'pos9', 9)


df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov1', 1)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov3', 3)
```

```
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov7', 7)


df1 <- funcao_lags(df1, df1$Lácteos, 'lact2', 2)
df1 <- funcao_lags(df1, df1$Lácteos, 'lact8', 8)


df1 <- funcao_lags(df1, df1$Suinocultura, 'sui3', 3)



df2 <- na.omit(df1)
#Separando variável preditora e as covariáveis
x = model.matrix(Pescado~.,df2)[,-1]
y = df2$Pescado
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1
# Análise dos Resíduos
acf2(resid(fit))
# Análise dos resíduos e seleção de variáveis de acordo com p-valor
y = ts(y, frequency=12)

x = x[,-1]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)


x = x[,-15]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)



x = x[,-1]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)



x = x[,-14]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)


x = x[,-9]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)


x = x[,-11]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)
```

```
x = x[,-2]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)



x = x[,-3]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)



x = x[,-5]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)


x = x[,-4]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)

checkresiduals(fit3)
acf2(fit3$residuals, main = "")

x = x[,-2]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)

x = x[,-2]
fit3 = Arima(y,order=c(0,0,0), seasonal = c(1, 0, 0),xreg=x)
coeftest(fit3)

checkresiduals(fit3)
acf2(fit3$residuals, main = "")
S


# Estruturando a base
df1<- funcao_lags(data_cut, data_cut$`Avicultura de Postura`, 'avp1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Postura`, 'avp12', 12)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc5', 5)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov3', 3)
df1 <- funcao_lags(df1, df1$Lácteos, 'lact11', 11)
df1 <- funcao_lags(df1, df1$Pescado, 'pes2', 2)
df1 <- funcao_lags(df1, df1$Pescado, 'pes9', 9)


df2 <- na.omit(df1)
#Separando variável preditora e as covariáveis
x = model.matrix(`Avicultura de Postura`~.,df2)[,-1]
y = df2$`Avicultura de Postura`
# Criando o modelo de Regressão Simples
```

```r
fit1 <- summary(fit <- lm(y~x))
fit1

# Análise dos Resíduos
acf2(resid(fit))
# Análise dos resíduos e seleção de variáveis de acordo com p-valor
fit2<- tirar_variaveis(0, 0, 0, x, y)
fit2[1]
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx)
coeftest(fit3)
checkresiduals(fit3)
acf2(fit3$residuals)


fit4 = Arima(y,order=c(3,0,0),xreg=xx,include.mean = FALSE,fixed=c(0,0,NA,NA,0,NA,NA,NA))
fit4
coeftest(fit4)
checkresiduals(fit4)
acf2(fit4$residuals, main = "")

# Estruturando a base

df1<- funcao_lags(data_cut, data_cut$Lácteos, 'lact1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Postura`, 'avp1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc6', 6)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov2', 2)
df1 <- funcao_lags(df1, df1$Pescado, 'pes4', 4)
df1 <- funcao_lags(df1, df1$Pescado, 'pes9', 9)


df2 <- na.omit(df1)
#Separando variável preditora e as covariáveis
x = model.matrix(Lácteos~.,df2)[,-1]
y = df2$Lácteos
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1

# Análise dos Resíduos
acf2(resid(fit))
# Análise dos resíduos e seleção de variáveis de acordo com p-valor
fit2 <- tirar_variaveis(0, 0, 0, x, y)

fit2[1]
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx,include.mean = FALSE)
```

```r
coeftest(fit3)
checkresiduals(fit3)
acf2(fit3$residuals, main = "")


# Estruturando a base
df1<- funcao_lags(data_cut, data_cut$Suinocultura, 'su1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc1', 1)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc6', 6)
df1<- funcao_lags(df1, df1$`Avicultura de Corte`, 'avc10', 10)
df1 <- funcao_lags(df1, df1$Bovinocultura, 'bov1', 1)
df2 <- na.omit(df1)
# Separando variável preditora e as covariáveis
x = model.matrix(Suinocultura~.,df2)[,-1]
y = df2$Suinocultura
# Criando o modelo de Regressão Simples
fit1 <- summary(fit <- lm(y~x))
fit1

# Análise dos Resíduos
acf2(resid(fit))
# Análise dos resíduos e seleção de variáveis de acordo com p-valor

fit2 <- tirar_variaveis(0, 0, 0, x, y)

fit2[1]
xx <- fit2[2]
xx<- xx[[1]]


fit3 = Arima(y,order=c(0,0,0),xreg=xx)
coeftest(fit3)
checkresiduals(fit3)
acf2(fit3$residuals)


fit4 = Arima(y,order=c(2,0,0),xreg=xx,fixed =c(0,NA,NA,NA,NA,NA))
fit4
coeftest(fit4)
checkresiduals(fit4)
acf2(fit4$residuals, main = "")

library(readxl)
data_anual = read_xlsx("Cadeia-Ano.xlsx")
# Análise das séries temporais anuais
head(data_anual)
# Análise Descritiva
z_avc = data_anual$`Avicultura de Corte`
z_avc = ts(z_avc, frequency = 1, start = 2007, end = 2019)

z_avp = data_anual$`Avicultura Postura`
z_avp = ts(z_avp, frequency = 1, start = 2007, end = 2019)
```

```r
z_bov = data_anual$`Bovinocultura de corte`
z_bov = ts(z_bov, frequency = 1, start = 2007, end = 2019)

z_lac = data_anual$`Lácteos`
z_lac = ts(z_lac, frequency = 1, start = 2007, end = 2019)

z_pesc = data_anual$Pescado
z_pesc = ts(z_pesc, frequency = 1, start = 2007, end = 2019)

z_suino = data_anual$Suinocultura
z_suino = ts(z_suino, frequency = 1, start = 2007, end = 2019)
# Análise Descritiva
plot(z_avc,main="Série Temporal da Avicultura de Corte", xlab= "Anos", ylab="IPCA")
plot(z_avp,main="Série Temporal da Avicultura de Postura", xlab= "Anos", ylab="IPCA")
plot(z_lac,main="Série Temporal do Lácteos", xlab= "Anos", ylab="IPCA")
plot(z_pesc,main="Série Temporal do Pescado", xlab= "Anos", ylab="IPCA")
plot(z_bov,main="Série Temporal da Bovinocultura", xlab= "Anos", ylab="IPCA")
plot(z_suino,main="Série Temporal da Suinocultura", xlab= "Anos", ylab="IPCA")
#Funções de Autocorrelações para Bovinocultura
par(mfrow = c(1, 2))
acf(z_bov, main="ACF Bovinocultura")
pacf(z_bov, main="PACF Bovinocultura")
#Funções de Autocorrelações para Avicultura de Corte
par(mfrow = c(1, 2))
acf(z_avc, main="ACF Avicultura de Corte")
pacf(z_avc, main="PACF Avicultura de Corte")
#Funções de Autocorrelações para Avicultura de Postura
par(mfrow = c(1, 2))
acf(z_avp, main="ACF Avicultura de Postura")
pacf(z_avp, main="PACF Avicultura de Postura")
#Funções de Autocorrelações para Lácteos
par(mfrow = c(1, 2))
acf(z_lac, main="ACF Lácteos")
pacf(z_lac, main="PACF Lácteos")
#Funções de Autocorrelações para Pescado
par(mfrow = c(1, 2))
acf(z_pesc, main="ACF Pescado")
pacf(z_pesc, main="PACF Pescado")
#Funções de Autocorrelações para Suinocultura
par(mfrow = c(1, 2))
acf(z_suino, main="ACF Suinocultura")
pacf(z_suino, main="PACF Suinocultura")
# Teste de Dickey-Fuller
adf.test(z_bov)
adf.test(z_avc)
adf.test(z_avp)
adf.test(z_lac)
adf.test(z_pesc)
adf.test(z_suino)

# Teste de Phillips-Perron
pp.test(z_bov)
pp.test(z_avc)
```

```r
pp.test(z_avp)
pp.test(z_lac)
pp.test(z_pesc)
pp.test(z_suino)

# Variáveis do modelo
library(glmnet)

colnames(data_anual) = c("ANO", "AVC","AVP","BOV","LAC","PESC","SUIN")
data_anual = data_anual[,-1]

#Correlaões cruzadas da Bovincultura
par(mfrow = c(3,2))
ccf(z_bov,z_avc,main="Bovinocultura e Avicultura de Corte")
ccf(z_bov,z_avp,main="Bovinocultura e Avicultura de Postura")
ccf(z_bov,z_lac,main="Bovinocultura e Lácteos")
ccf(z_bov,z_pesc,main="Bovinocultura e Pescado")
ccf(z_bov,z_suino,main="Bovinocultura e Suinocultura")


# Regressão LASSO
set.seed(1)
x = model.matrix(BOV~ .,data=data_anual)[,-1]
y = data_anual$BOV

cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
par(mfrow=c(1,1))
plot(cv.model)

coef(cv.model,cv.model$lambda.min)


# Pescados

par(mfrow = c(3,2))
ccf(z_pesc,z_avc,main="Pescado e Avicultura de Corte")
ccf(z_pesc,z_avp,main="Pescado e Avicultura de Postura")
ccf(z_pesc,z_bov,main="Pescado e Bovinocultura")
ccf(z_pesc,z_lac,main="Pescado e Lácteos")
ccf(z_pesc,z_suino,main="Pescado e Suinocultura")


# Regressão LASSO
set.seed(2)
x = model.matrix(PESC~ .,data=data_anual)[,-1]
y = data_anual$PESC

cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
par(mfrow=c(1,1))
plot(cv.model)
```

```r
coef(cv.model,cv.model$lambda.min)

# Avicultura de Corte

par(mfrow = c(3,2))
ccf(z_avc,z_avp,main="Avicultura de Corte e Avicultura de Postura")
ccf(z_avc,z_bov,main="Avicultura de Corte e Bovinocultura")
ccf(z_avc,z_lac,main="Avicultura de Corte e Lácteos")
ccf(z_avc,z_pesc,main="Avicultura de Corte e Pescado")
ccf(z_avc,z_suino,main="Avicultura de Corte e Suinocultura")


# Regressão LASSO
set.seed(3)
x = model.matrix(AVC~ .,data=data_anual)[,-1]
y = data_anual$AVC

cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
par(mfrow=c(1,1))
plot(cv.model)

coef(cv.model,cv.model$lambda.min)
# Avicultura de Postura

par(mfrow = c(3,2))
ccf(z_avp,z_avc,main="Avicultura de Postura e Avicultura de Corte")
ccf(z_avp,z_bov,main="Avicultura de Postura e Bovinocultura")
ccf(z_avp,z_lac,main="Avicultura de Postura e Lácteos")
ccf(z_avp,z_pesc,main="Avicultura de Postura e Pescado")
ccf(z_avp,z_suino,main="Avicultura de Postura e Suinocultura")


# Regressão LASSO
set.seed(4)
x = model.matrix(AVP~ .,data=data_anual)[,-1]
y = data_anual$AVP

par(mfrow=c(1,1))
cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
plot(cv.model)

coef(cv.model,cv.model$lambda.min)

# Lacteos
par(mfrow = c(3,2))
ccf(z_lac,z_avc,main="Lácteos e Avicultura de Corte")
ccf(z_lac,z_avp,main="Lácteos e Avicultura de Postura")
ccf(z_lac,z_bov,main="Lácteos e Bovinocultura")
ccf(z_lac,z_pesc,main="Lácteos e Pescado")
ccf(z_lac,z_suino,main="Lácteos e Suinocultura")
```

```r
# Regressão LASSO
set.seed(5)
x = model.matrix(LAC~ .,data=data_anual)[,-1]
y = data_anual$LAC

par(mfrow=c(1,1))
cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
plot(cv.model)

coef(cv.model,cv.model$lambda.min)

# Suinocultura


par(mfrow = c(3,2))
ccf(z_suino,z_avc,main="Suinocultura e Avicultura de Corte")
ccf(z_suino,z_avp,main="Suinocultura e Avicultura de Postura")
ccf(z_suino,z_bov,main="Suinocultura e Bovinocultura")
ccf(z_suino,z_lac,main="Suinocultura e Lacteos")
ccf(z_suino,z_pesc,main="Suinocultura e Pescado")


# Regressão LASSO
set.seed(6)
x = model.matrix(SUIN~ .,data=data_anual)[,-1]
y = data_anual$SUIN

par(mfrow=c(1,1))
cv.model = cv.glmnet(x,y,alpha = 1)
cv.model
plot(cv.model)

coef(cv.model,cv.model$lambda.min)
```