Trabalho Final Series Temporais

Aluno: Diogo Costa Hutner

Primeiro, carregando os modulos e os dados.

```
library(readxl)
library(zoo)
library(lubridate)
library(ggplot2)
library(tseries)
library(magick)
library(gridExtra)
library(ggfortify)
library(stats)
library(forecast)
library(vars)
library(lmtest)
library(stargazer)
library(urca)


dados_cointegracao <- read_excel("dados sobre cointegracao.xlsx", sheet = 1)
dados_cointegracao$trim <- as.yearqtr(dados_cointegracao$trim, format = "%Yq%q")

dados_sarima <- read_excel("Dados para replicar SARIMA  Vendas de Automoveis USA.xlsx", sheet
= 1)
colnames(dados_sarima) <- c("trim","RCAR6T")
dados_sarima$trim <- as.Date(dados_sarima$trim)

dados_var <- read_excel("Dados exemplo Modelo VAR sobre vendas veiculos e juros.xlsx", sheet=
1)
dados_var$txj <- as.numeric(dados_var$txj)
colnames(dados_var) <- c("trim","vv","txj")
dados_var$trim <- ymd(paste0(dados_var$trim, "01"))
cor_linha <- "#11CC22"
```

Modelo SARIMA

De começo, ver os graficos como eles estão, depois plotar ele e fazer teste de estacionaridade, apos isso ACF e PACF. Por fim postar um summary do modelo

```
dados_sarima
```

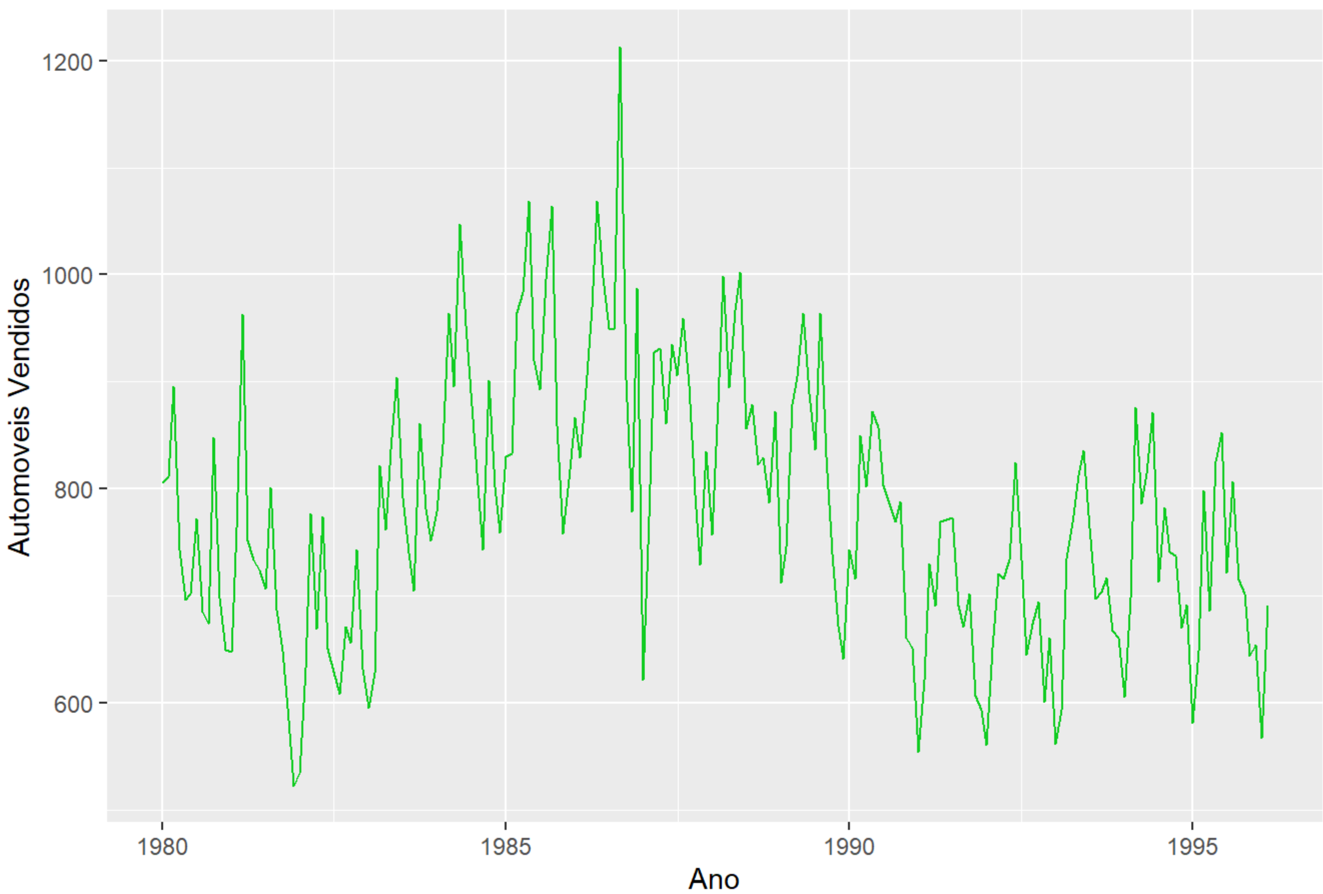
	trim <date>	RCAR6T <dbl>
	1980-01-01	805.8
	1980-02-01	811.6
	1980-03-01	895.2

trim <date>	RCAR6T <dbl>
1980-04-01	743.3
1980-05-01	696.7
1980-06-01	701.9
1980-07-01	772.6
1980-08-01	685.5
1980-09-01	674.1
1980-10-01	847.5

1-10 of 194 rows Previous **1** 2 3 4 5 6 ... 20 Next

```
grafico_sar_inicial <- ggplot(dados_sarima, aes(x = trim, y = RCAR6T)) +
  geom_line(color = cor_linha) +
  labs(x = "Ano", y = "Automoveis Vendidos", title = "Gráfico p/ SARIMA Automoveis")
grafico_sar_inicial
```

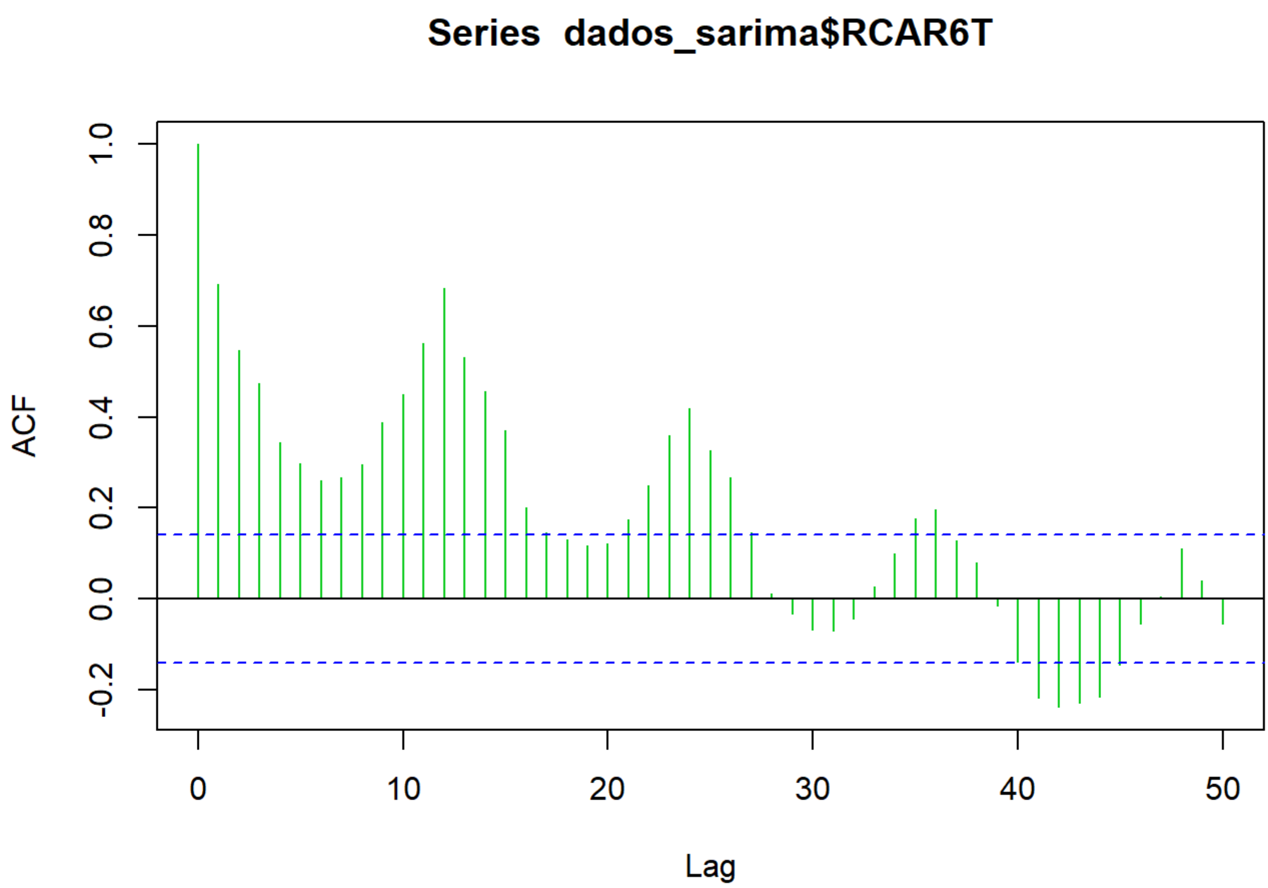
Gráfico p/ SARIMA Automoveis




```
results_adf_sar <- adf.test(dados_sarima$RCAR6T)
results_adf_sar
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  dados_sarima$RCAR6T
## Dickey-Fuller = -3.5165, Lag order = 5, p-value = 0.04263
## alternative hypothesis: stationary
```

```
acf_result <- acf(dados_sarima$RCAR6T, lag=50, col= cor_linha)
```



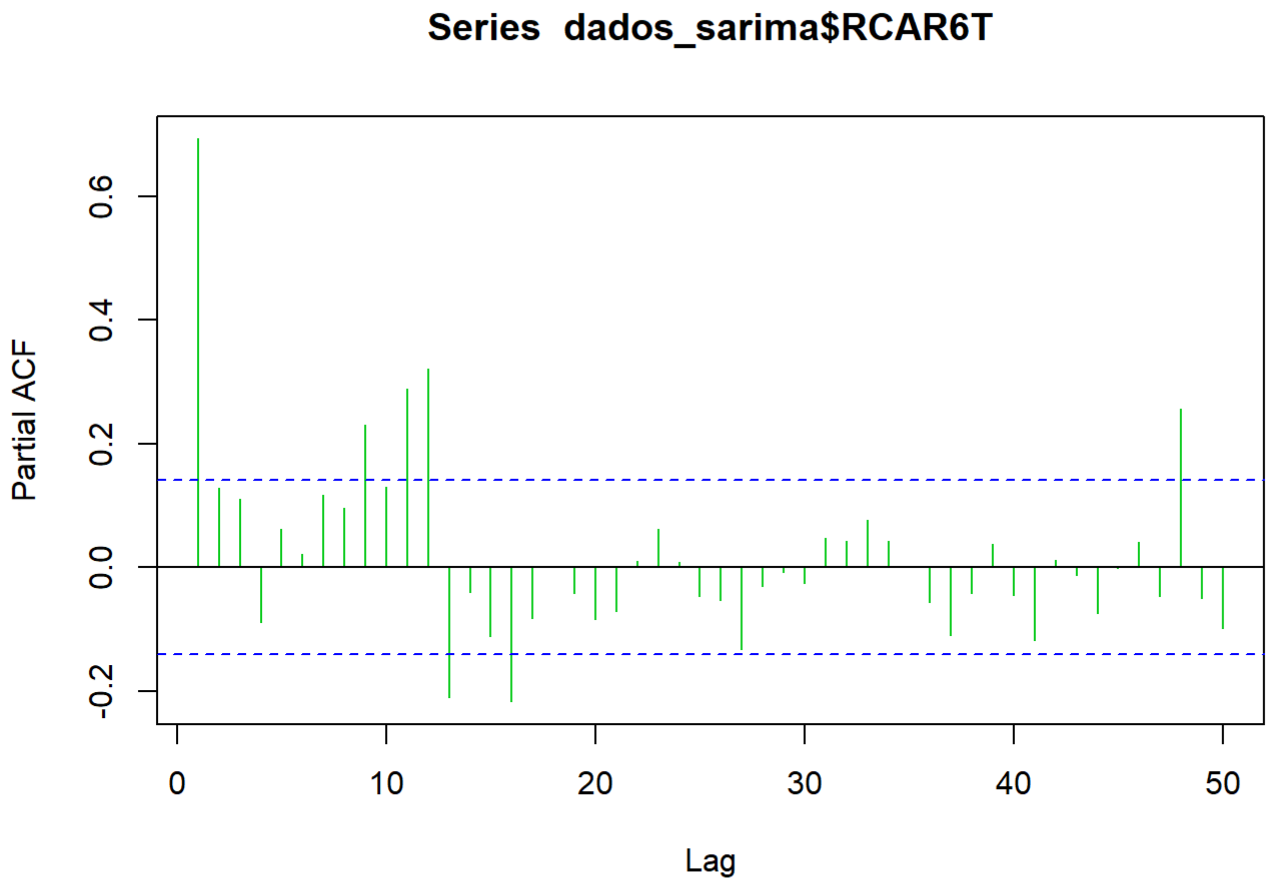

```
as.data.frame(acf_result$acf)
```

V1 <dbl>
1.000000000
0.692407662
0.545943491
0.474553676
0.344853406
0.296686722
0.259254061
0.267136345

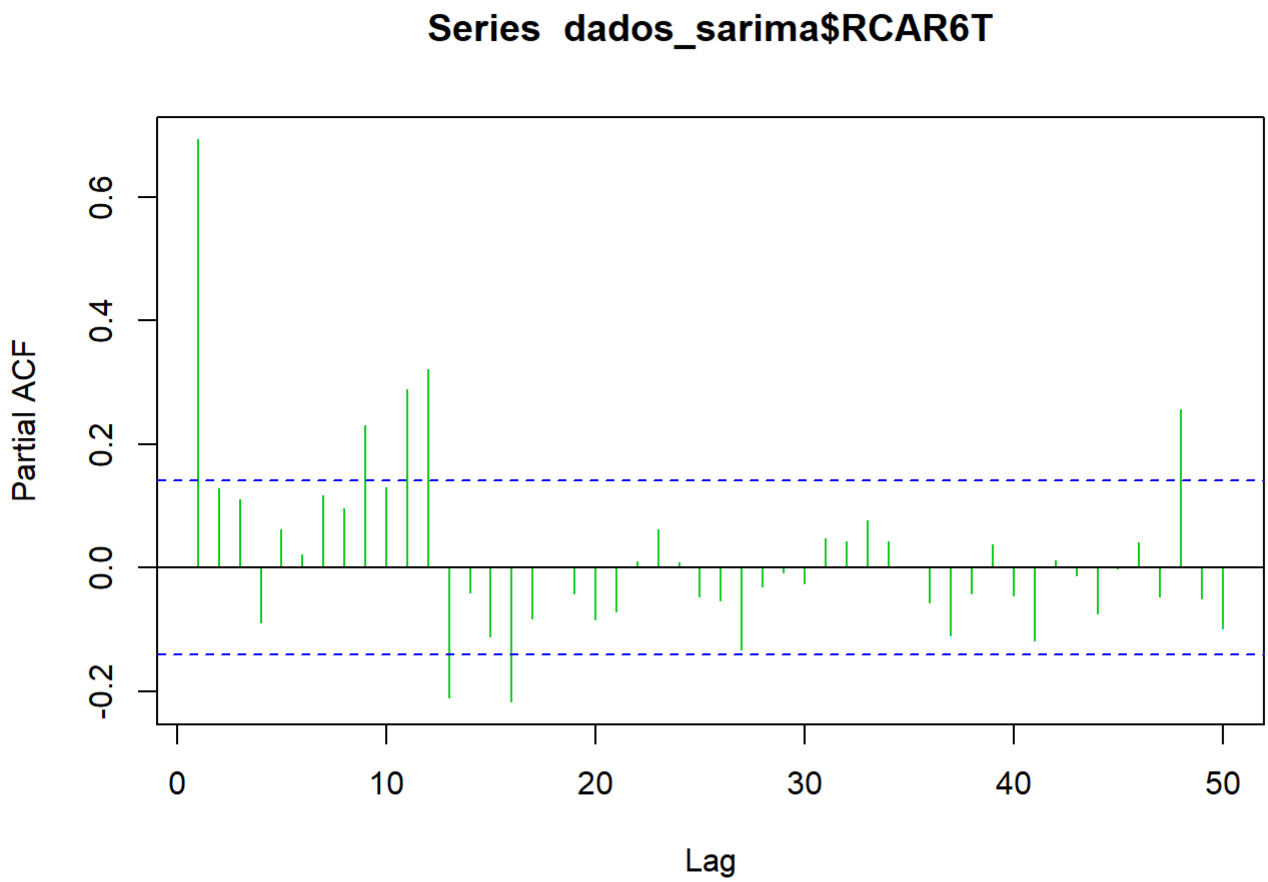
V1
<dbl>
0.295922854
0.388487315

1-10 of 51 rows Previous 1 2 3 4 5 6 Next

```
acf_result <- pacf(dados_sarima$RCAR6T, lag=50, col = cor_linha)
```




```
pacf(dados_sarima$RCAR6T, lag=50, col = cor_linha)
```




```
modelo_sarima <- Arima(dados_sarima$RCAR6T, order = c(1, 0, 0), seasonal = list(order = c(1,
0, 0), period = 12))

summary(modelo_sarima)
```

```
## Series: dados_sarima$RCAR6T
## ARIMA(1,0,0)(1,0,0)[12] with non-zero mean
##
## Coefficients:
##          ar1    sar1      mean
##       0.5438  0.5538  773.7203
## s.e.  0.0638  0.0626   24.2330
##
## sigma^2 = 5580:  log likelihood = -1112.95
## AIC=2233.9   AICc=2234.11   BIC=2246.97
##
## Training set error measures:
##                     ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 0.1506749 74.12011 57.38731 -0.9516228 7.434101 0.7504201
##                     ACF1
## Training set -0.06595613
```

modelo VAR

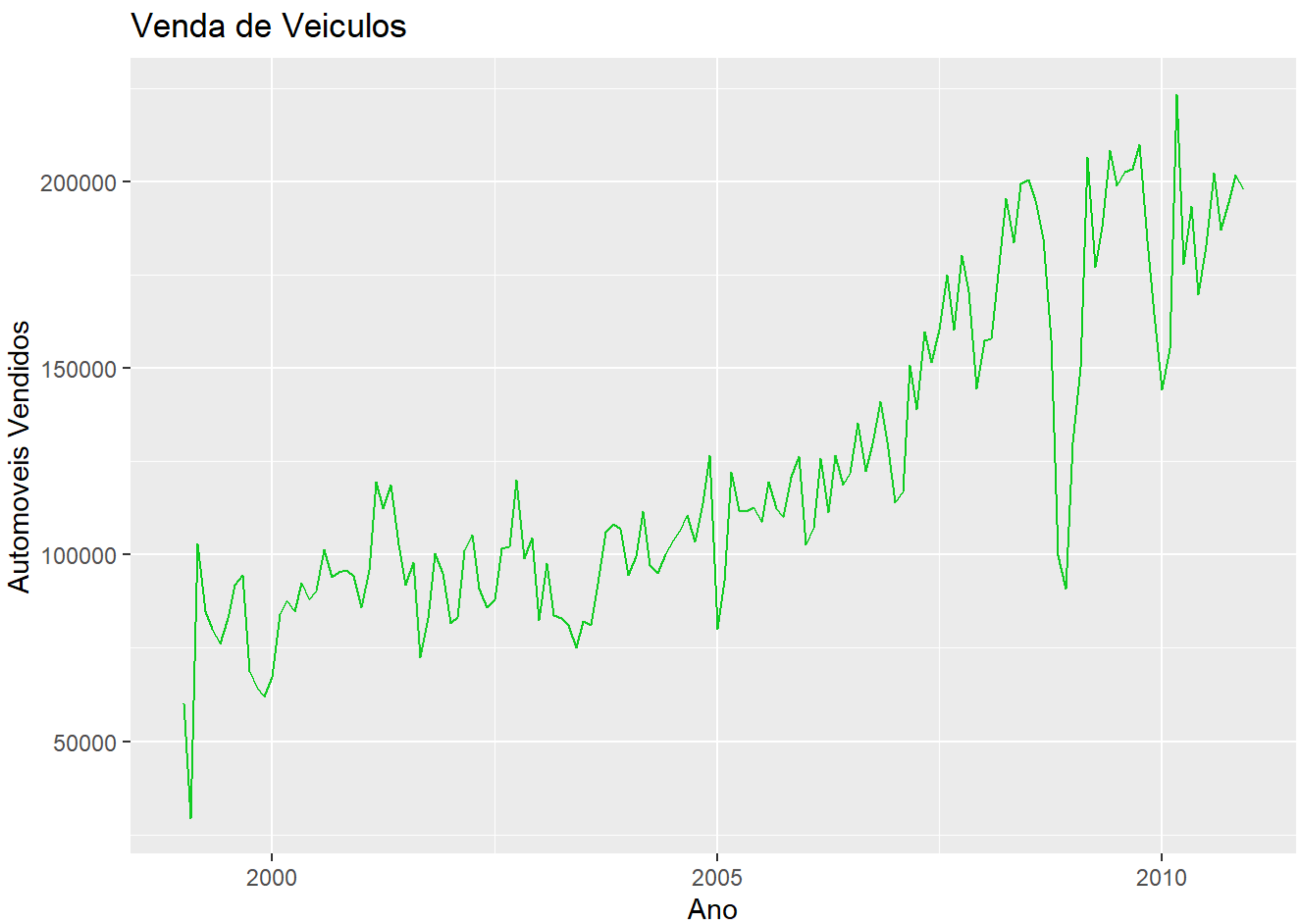
Primeiro, ver os dados, segundo os graficos, depois teste de estacionaridade

```
dados_var
```

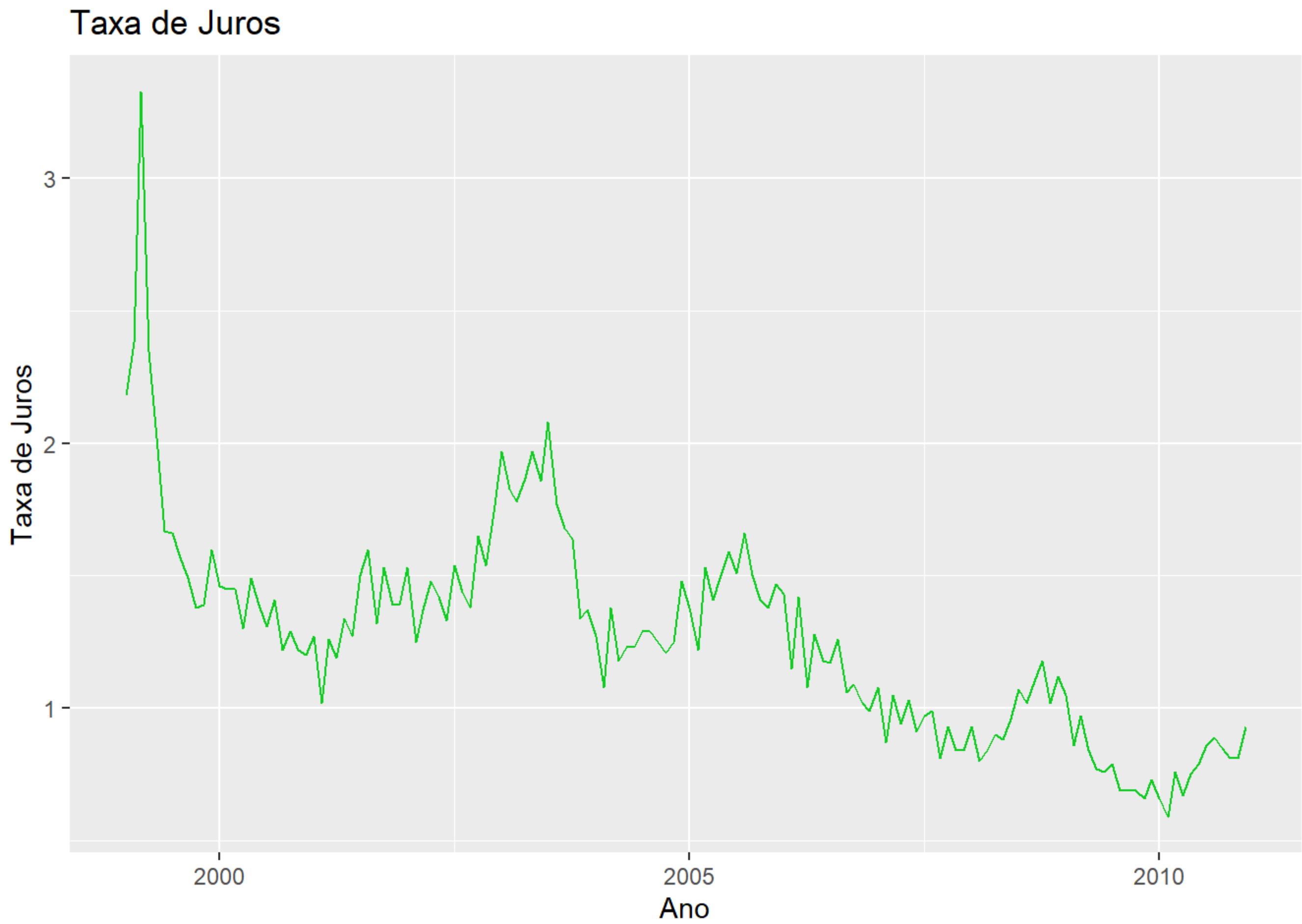
trim	vv	txj
<date>	<dbl>	<dbl>
1999-01-01	60069	2.18
1999-02-01	29474	2.38
1999-03-01	103085	3.33
1999-04-01	84816	2.35
1999-05-01	80241	2.02
1999-06-01	76260	1.67
1999-07-01	82960	1.66
1999-08-01	91896	1.57
1999-09-01	94546	1.49
1999-10-01	68898	1.38

1-10 of 144 rows Previous **1** 2 3 4 5 6 ... 15 Next

```
grafico_var_inicial_vv <- ggplot(dados_var, aes(x = trim, y = vv)) +
  geom_line(color = cor_linha) +
  labs(x = "Ano", y = "Automoveis Vendidos", title = "Venda de Veiculos")
grafico_var_inicial_vv
```



```
grafico_var_inicial_vv <- ggplot(dados_var, aes(x = trim, y = txj)) +
  geom_line(color = cor_linha) +
  labs(x = "Ano", y = "Taxa de Juros", title = "Taxa de Juros")

grafico_var_inicial_vv
```



Teste de Dickey-Fuller sobre vendas de carros e apos sobre taxa de juros

```
adf.test(dados_var$vv)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  dados_var$vv
## Dickey-Fuller = -3.7548, Lag order = 5, p-value = 0.02323
## alternative hypothesis: stationary
```

```
adf.test(dados_var$txj)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  dados_var$txj
## Dickey-Fuller = -3.032, Lag order = 5, p-value = 0.1467
## alternative hypothesis: stationary
```

Montando um modelo VAR:

```
modelo_var <- VAR(dados_var[,c("txj",'vv')], p = 1)
summary(modelo_var)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: txj, vv
## Deterministic variables: const
## Sample size: 143
## Log Likelihood: -1526.109
## Roots of the characteristic polynomial:
## 0.9594 0.499
## Call:
## VAR(y = dados_var[, c("txj", "vv")], p = 1)
##
##
## Estimation results for equation txj:
## ====================================
## txj = txj.l1 + vv.l1 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## txj.l1  7.095e-01  4.964e-02  14.293  < 2e-16 ***
## vv.l1  -2.294e-06  4.792e-07  -4.788 4.23e-06 ***
## const   6.425e-01  1.140e-01   5.635 9.29e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.1634 on 140 degrees of freedom
## Multiple R-Squared: 0.8297,  Adjusted R-squared: 0.8273
## F-statistic: 341.1 on 2 and 140 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation vv:
## ===================================
## vv = txj.l1 + vv.l1 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## txj.l1 -2.293e+04  5.023e+03  -4.565 1.09e-05 ***
## vv.l1   7.489e-01  4.849e-02  15.443  < 2e-16 ***
## const   6.095e+04  1.154e+04   5.283 4.76e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 16540 on 140 degrees of freedom
## Multiple R-Squared: 0.8437,  Adjusted R-squared: 0.8415
## F-statistic: 377.8 on 2 and 140 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##           txj        vv
## txj   0.02671 8.073e+02
## vv  807.28055 2.735e+08
##
## Correlation matrix of residuals:
##        txj     vv
```

```
## txj 1.0000 0.2987
## vv  0.2987 1.0000
```

Causalidade de Granger

```
grangertest(txj ~ vv, data = dados_var)
```

	Res.Df <dbl>	Df <dbl>	F <dbl>	Pr(>F) <dbl>
1	140	*NA*	*NA*	*NA*
2	141	-1	22.92583	4.23232e-06

2 rows

```
print('-------------')
```

```
## [1] "-------------"
```

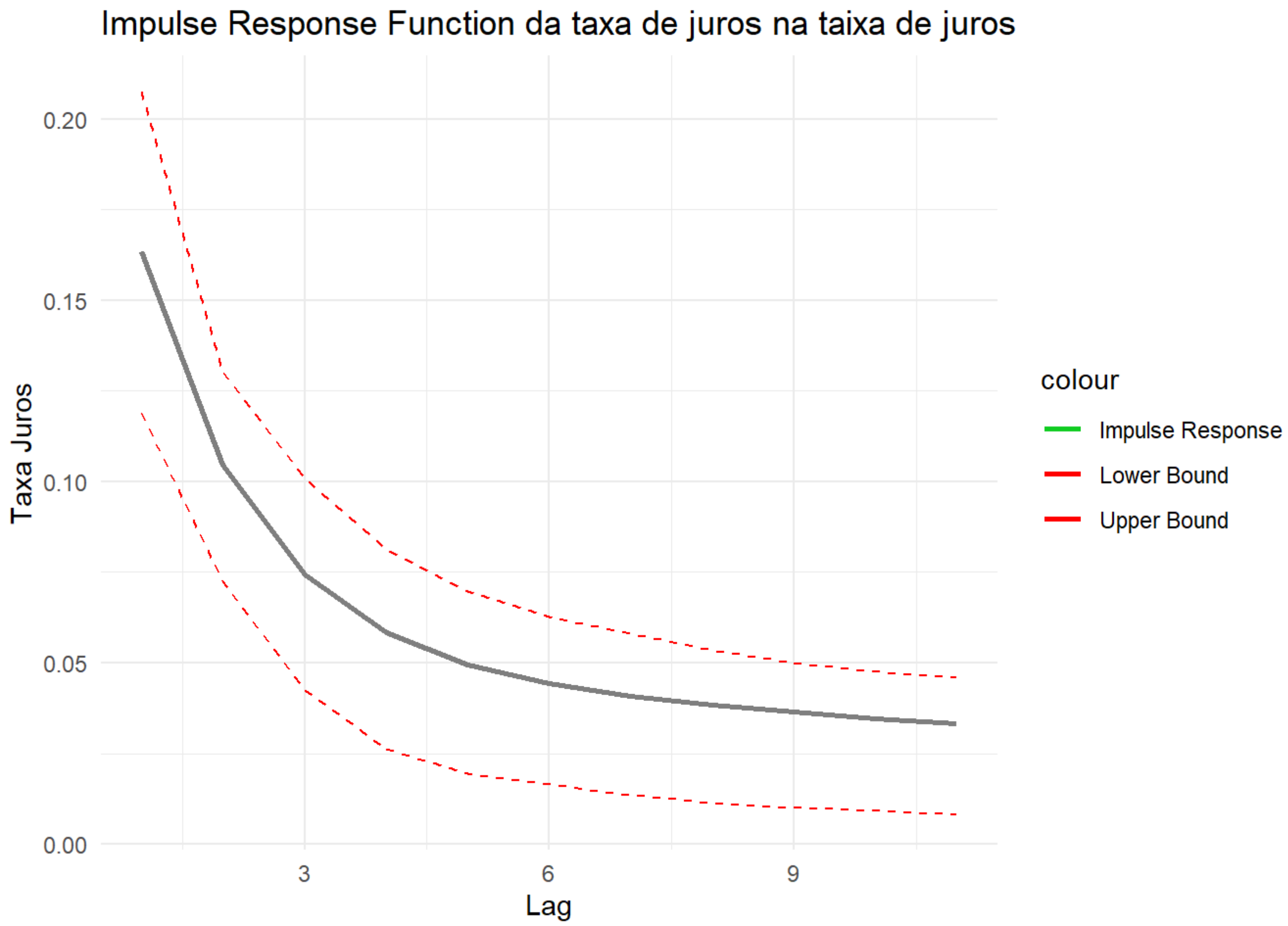
```
grangertest(vv ~ txj, data = dados_var)
```

	Res.Df <dbl>	Df <dbl>	F <dbl>	Pr(>F) <dbl>
1	140	*NA*	*NA*	*NA*
2	141	-1	20.83561	1.085231e-05

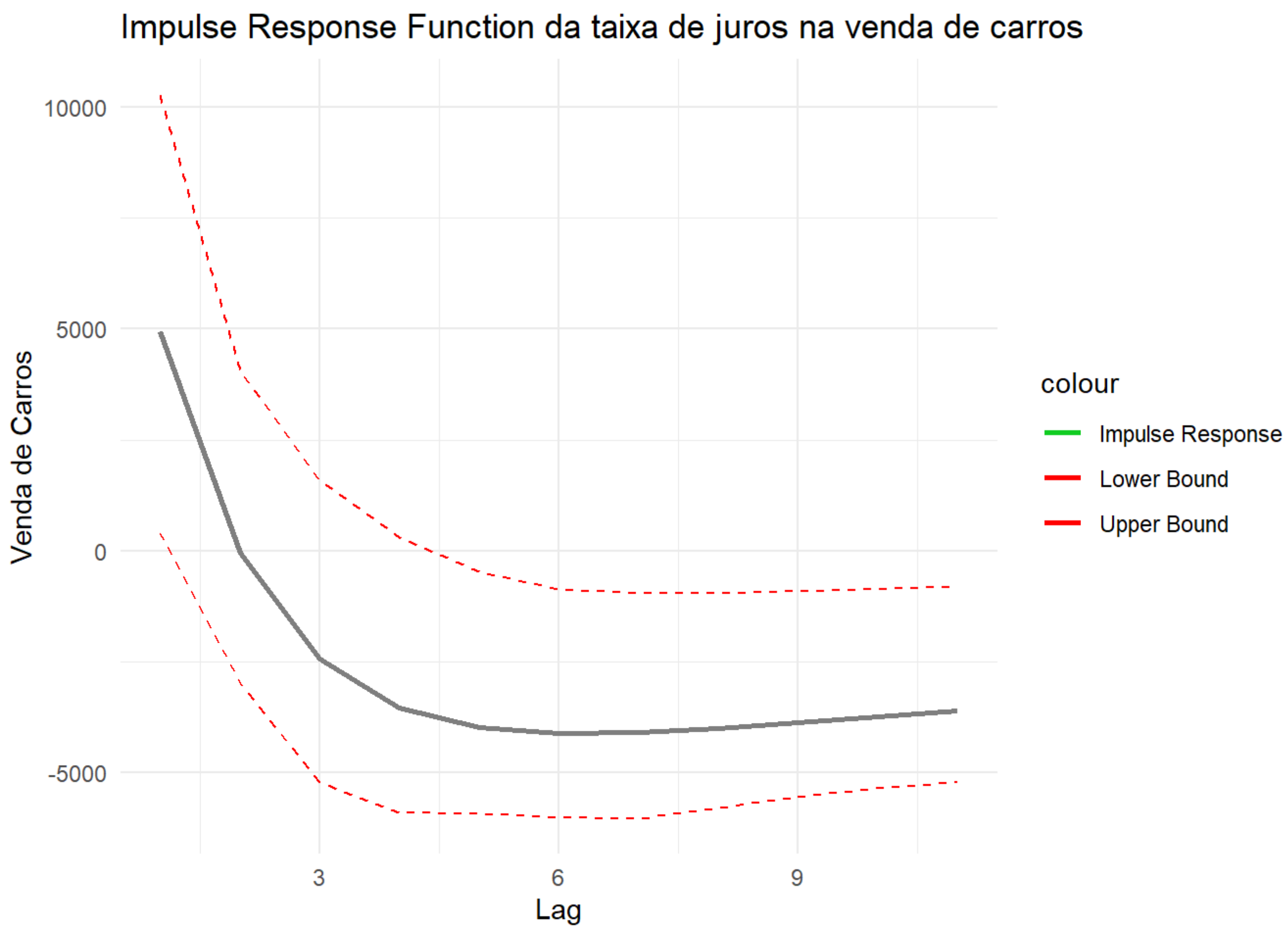
2 rows

Plotando graficos de IRF

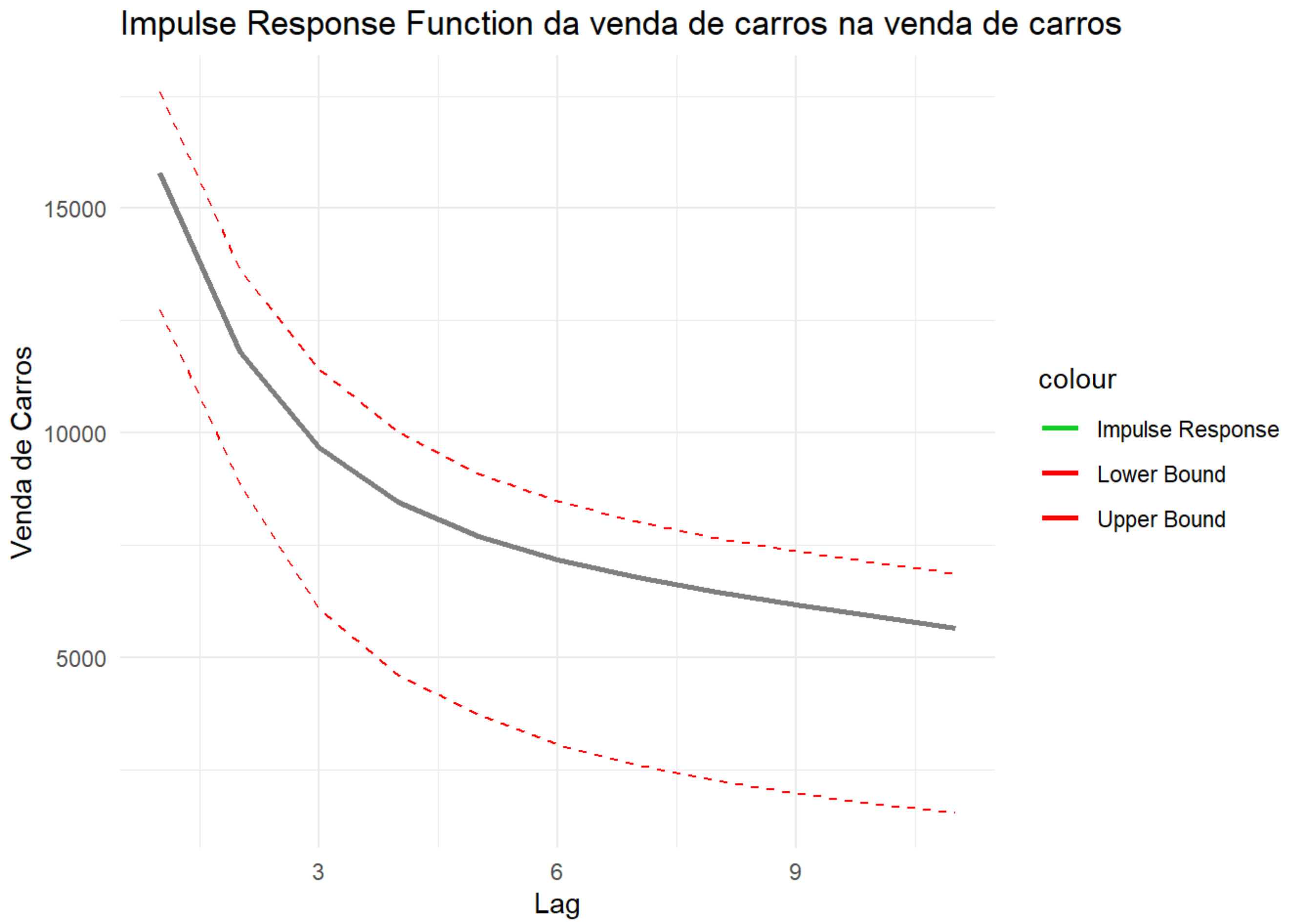
```
irf_txj <- irf(modelo_var,impulse = "txj", response=c("txj","vv"), n.ahead = 10)

dados_irf_txj <- data.frame(
  txj = irf_txj$irf$txj[, 'txj'],
  vv = irf_txj$irf$txj[, 'vv'],
  lower_txj = irf_txj$Lower$txj[, 'txj'],
  upper_txj = irf_txj$Upper$txj[, 'txj'],
  lower_vv = irf_txj$Lower$txj[, 'vv'],
  upper_vv = irf_txj$Upper$txj[, 'vv']
)

grafico_irf_txj_txj <- ggplot(dados_irf_txj, aes(x = seq_along(txj))) +
  geom_line(aes(y = txj, color = "Resposta ao Impulso"), size = 1) +
  geom_line(aes(y = lower_txj, color = "Limite Inferior"), linetype = "dashed", color = "re
d") +
  geom_line(aes(y = upper_txj, color = "Limite Superior"), linetype = "dashed", color = "re
d") +
  scale_color_manual(values = c("Impulse Response" = cor_linha, "Lower Bound" = "red", "Upper
Bound" = "red")) +
  labs(x = "Lag", y = "Taxa Juros", title = "Impulse Response Function da taxa de juros na ta
ixa de juros") +
  theme_minimal()


grafico_irf_txj_vv <- ggplot(dados_irf_txj, aes(x = seq_along(txj))) +
  geom_line(aes(y = vv, color = "Resposta ao Impulso"), size = 1) +
  geom_line(aes(y = lower_vv, color = "Limite Inferior"), linetype = "dashed", color = "red")
+
  geom_line(aes(y = upper_vv, color = "Limite Superior"), linetype = "dashed", color = "red")
+
  scale_color_manual(values = c("Impulse Response" = cor_linha, "Lower Bound" = "red", "Upper
Bound" = "red")) +
  labs(x = "Lag", y = "Venda de Carros", title = "Impulse Response Function da taixa de juros
na venda de carros") +
  theme_minimal()
print(grafico_irf_txj_txj)
```



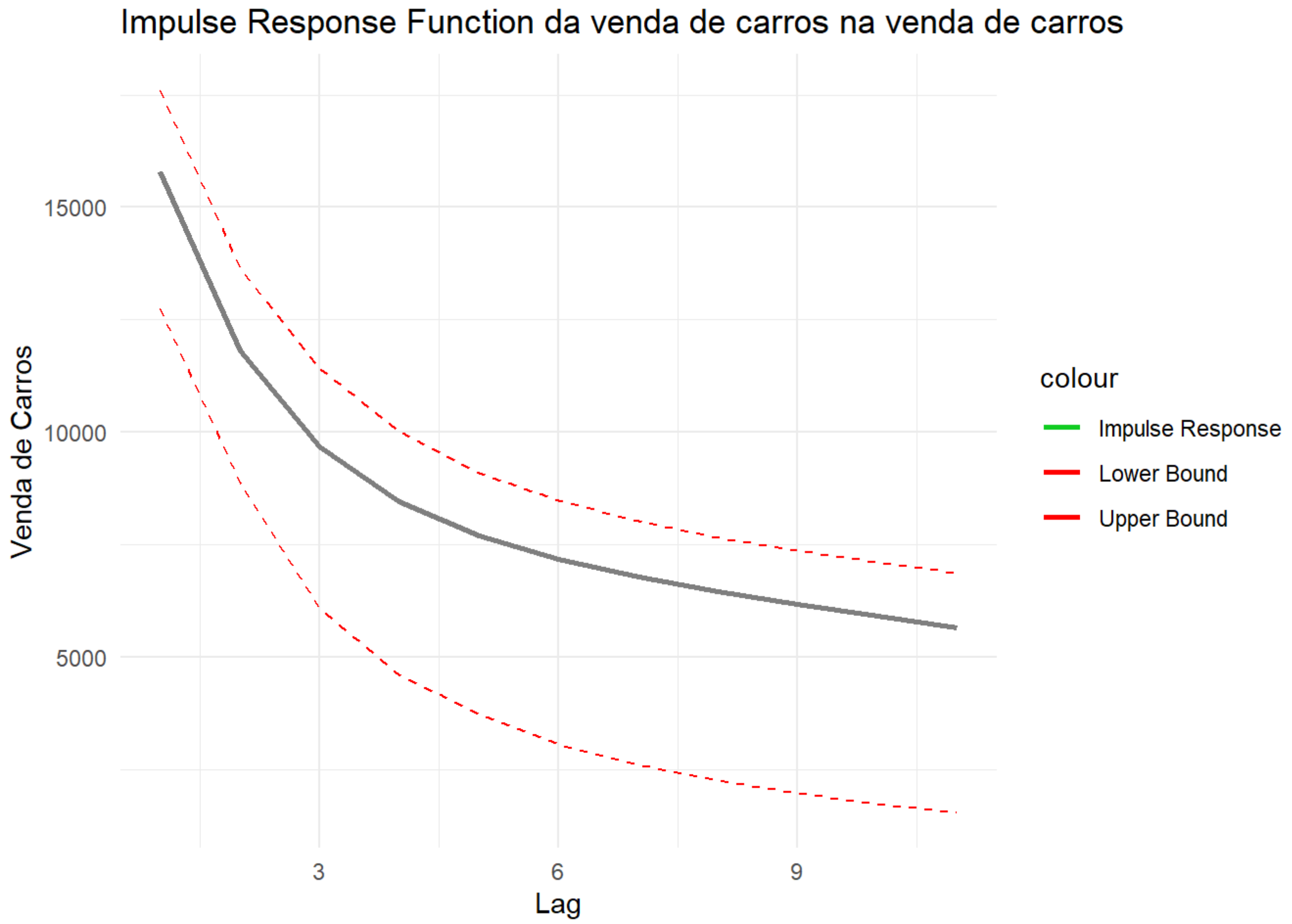

```
print(grafico_irf_txj_vv)
```



```
irf_vv <- irf(modelo_var,impulse = "vv", response=c("txj","vv"), n.ahead = 10)

dados_irf_vv <- data.frame(
  txj = irf_vv$irf$vv[, 'txj'],
  vv = irf_vv$irf$vv[, 'vv'],
  lower_txj = irf_vv$Lower$vv[, 'txj'],
  upper_txj = irf_vv$Upper$vv[, 'txj'],
  lower_vv = irf_vv$Lower$vv[, 'vv'],
  upper_vv = irf_vv$Upper$vv[, 'vv']
)

grafico_irf_vv_txj <- ggplot(dados_irf_vv, aes(x = seq_along(txj))) +
  geom_line(aes(y = txj, color = "Resposta ao Impulso"), size = 1) +
  geom_line(aes(y = lower_txj, color = "Limite Inferior"), linetype = "dashed", color = "re
d") +
  geom_line(aes(y = upper_txj, color = "Limite Superior"), linetype = "dashed", color = "re
d") +
  scale_color_manual(values = c("Impulse Response" = cor_linha, "Lower Bound" = "red", "Upper
Bound" = "red")) +
  labs(x = "Lag", y = "Taxa Juros", title = "Impulse Response Function da venda de carro na t
axa de juros") +
  theme_minimal()


grafico_irf_vv_txj <- ggplot(dados_irf_vv, aes(x = seq_along(txj))) +
  geom_line(aes(y = vv, color = "Resposta ao Impulso"), size = 1) +
  geom_line(aes(y = lower_vv, color = "Limite Inferior"), linetype = "dashed", color = "red")
+
  geom_line(aes(y = upper_vv, color = "Limite Superior"), linetype = "dashed", color = "red")
+
  scale_color_manual(values = c("Impulse Response" = cor_linha, "Lower Bound" = "red", "Upper
Bound" = "red")) +
  labs(x = "Lag", y = "Venda de Carros", title = "Impulse Response Function da venda de carro
s na venda de carros") +
  theme_minimal()
grafico_irf_vv_txj
```




```
grafico_irf_vv_txj
```



#

Cointegração e VEC

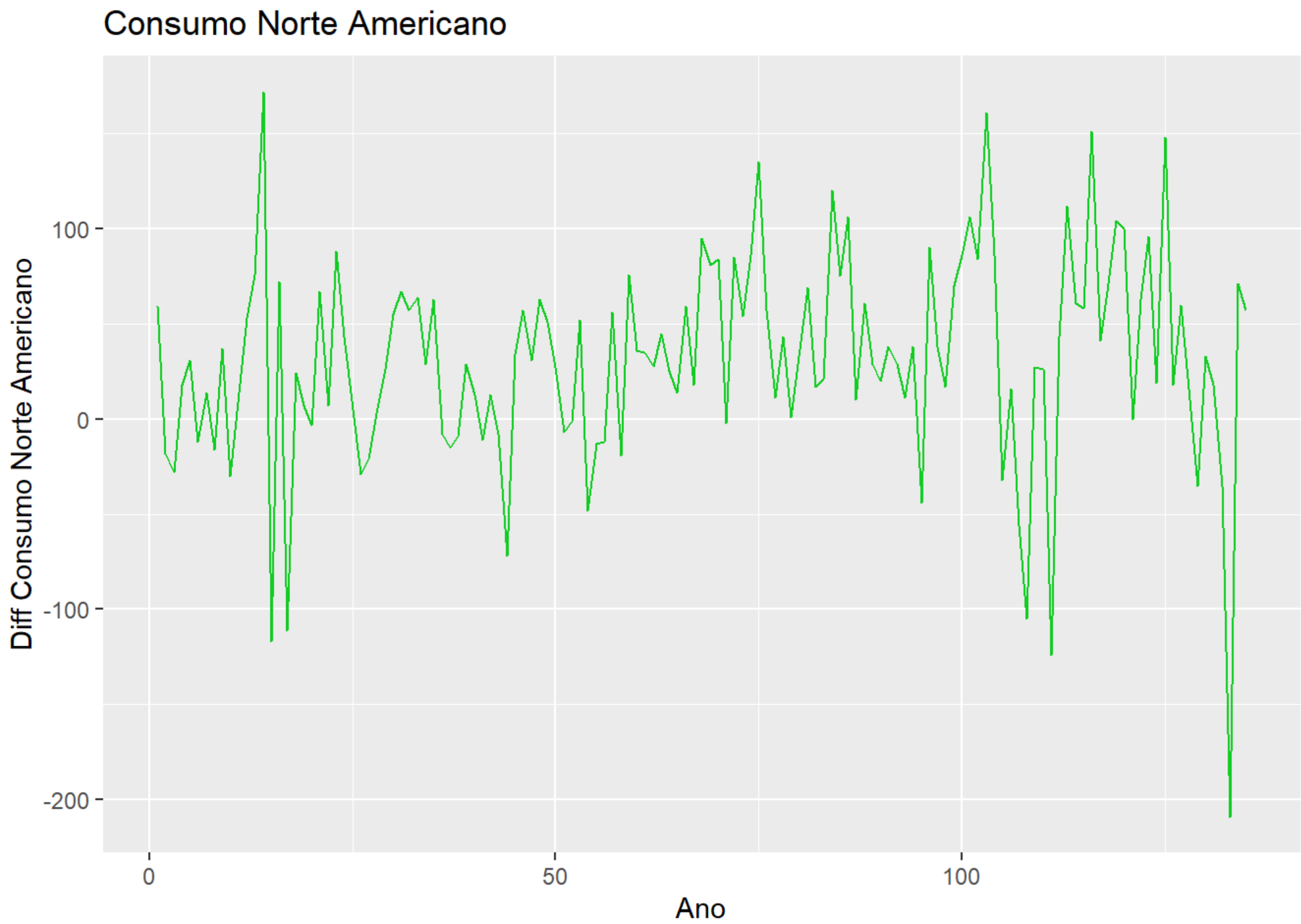
Por questão de espaço, não irei realizar visivelmente aq o Dickey-Fuller, mas mostrou que não é estacionaria, então vamos diferenciar

```
dados_dif_cointegracao <- cbind(diff(dados_cointegracao$c), diff(dados_cointegracao$y))
colnames(dados_dif_cointegracao) <- c("c","y")
rownames(dados_dif_cointegracao) <- dados_cointegracao$trim[-1]
df_dados_dif_cointegracao <- fortify(dados_dif_cointegracao)
df_dados_dif_cointegracao
```

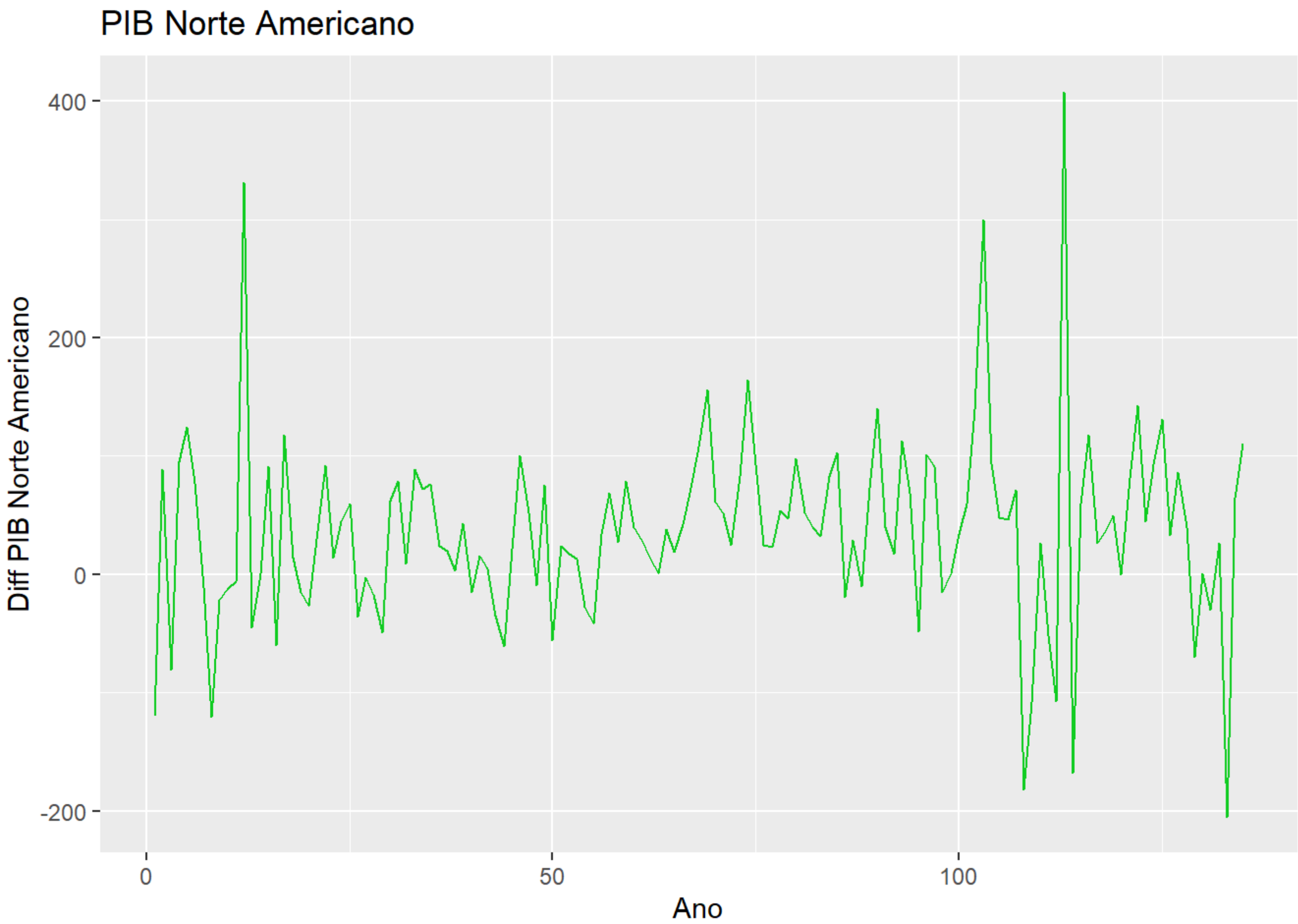
	c <dbl>	y <dbl>
1947.25	59	-120
1947.5	-18	89
1947.75	-28	-81
1948	18	95
1948.25	31	124
1948.5	-12	77
1948.75	14	-11
1949	-16	-121
1949.25	37	-22
1949.5	-30	-13

1-10 of 135 rows Previous **1** 2 3 4 5 6 ... 14 Next

```
grafico_var_inicial_vv <- ggplot(df_dados_dif_cointegracao, aes(x = index(dados_dif_cointegra
cao), y = c)) +
  geom_line(color = cor_linha) +
  labs(x = "Ano", y = "Diff Consumo Norte Americano", title = "Consumo Norte Americano")

grafico_var_inicial_vv
```




```
grafico_var_inicial_vv <- ggplot(df_dados_dif_cointegracao, aes(x = index(dados_dif_cointegra
cao), y = y)) +
  geom_line(color = cor_linha) +
  labs(x = "Ano", y = "Diff PIB Norte Americano", title = "PIB Norte Americano")
grafico_var_inicial_vv
```



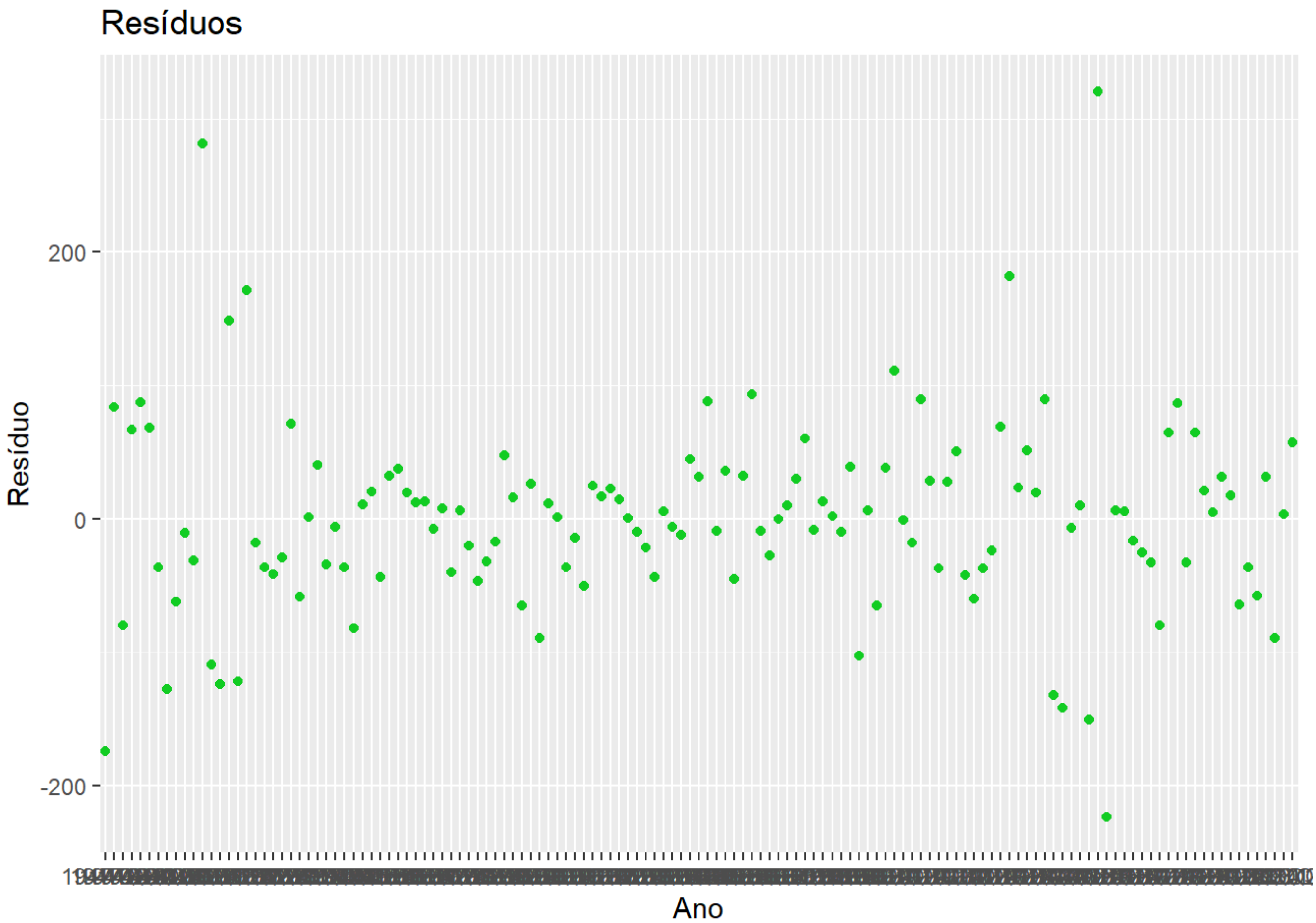
Sumario/Resumo do modelo de cointegração:

```
modelo_cointegracao <- lm(y ~ c, data = as.data.frame(dados_dif_cointegracao))
summary(modelo_cointegracao)
```

```
##
## Call:
## lm(formula = y ~ c, data = as.data.frame(dados_dif_cointegracao))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -223.14  -36.44    1.08   31.42  320.60
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  16.5551     7.1995   2.299    0.023 *
## c             0.6326     0.1134   5.579  1.3e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 72.85 on 133 degrees of freedom
## Multiple R-squared:  0.1897, Adjusted R-squared:  0.1836
## F-statistic: 31.13 on 1 and 133 DF,  p-value: 1.3e-07
```

Analisando os residuos

```
residuos <- fortify(as.data.frame(residuals(modelo_cointegracao)))
colnames(residuos) <- c('res')
rownames(residuos) <- dados_cointegracao$trim[-1]

grafico_resi <- ggplot(residuos, aes(x = rownames(residuos), y = res)) +
  geom_point(color = cor_linha) +
  labs(x = "Ano", y = "Resíduo", title = "Resíduos")
grafico_resi
```



ADF sobre os residuos

```
adf.test(residuos$res)
```

```
## Warning in adf.test(residuos$res): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  residuos$res
## Dickey-Fuller = -5.7417, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

Por fim, um modelo VEC:

```
vec_results <- ca.jo(df_dados_dif_cointegracao, type = "eigen", K=2)
summary(vec_results)
```

```
##
## ######################
## # Johansen-Procedure #
## ######################
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.4249105 0.2331722
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 1 | 35.31  6.50  8.18 11.65
## r = 0  | 73.58 12.91 14.90 19.19
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##           c.l2       y.l2
## c.l2  1.000000 1.00000000
## y.l2 -1.140634 0.01272331
##
## Weights W:
## (This is the loading matrix)
##
##           c.l2       y.l2
## c.d -0.1053298 -0.6953155
## y.d  1.0828513 -0.5656351
```