Trabalho 1 de Análise de séries temporais

O Objetivo deste trabalho é de responder as seguintes arguições:

1. Carregar o arquivo: DepartmentStoreSales\_V2.xls
2. Plotar o gráfico da série temporal
3. Calcular o resumo estatístico da base
4. Separar em amostra de desenvolvimento e teste da seguinte maneira 4.1 Desenvolvimento: 2005/1 até 2009/4 4.2 Teste: 2010/1 até 2010/4
5. Estimar um modelo de tendência linear
6. Estimar um modelo de tendência quadrática (polinômio de grau 2)
7. Estimar um modelo de tendência linear com sazonalidade
8. Estimar um modelo de tendência quadrática (polinômio de grau 2) com sazonalidade
9. Calcular os erros de projeção para cada modelo
10. Escolher o melhor modelo de projeção justificando ———————————————————————————–
11. Carregar o arquivo: DepartmentStoreSalesV2.xls

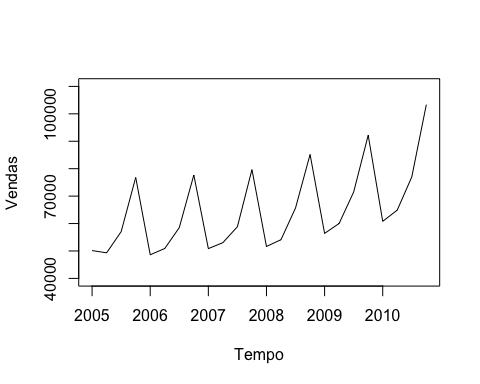
sales<- read\_excel("DepartmentStoreSales\_V2.xls")

1. Plotar o gráfico da série temporal: Converter em série temporal

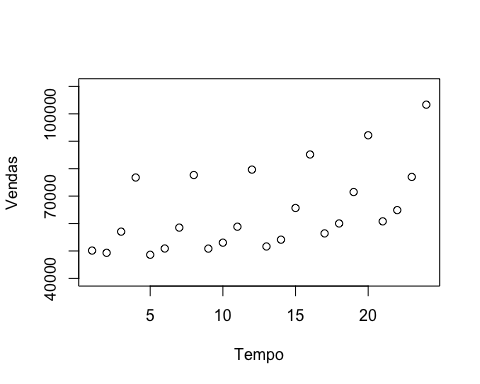
sales\_ts <- ts(sales$Sales, start=c(2005,1), end=c(2010,04), frequency = 4)

Plotar os gráficos:

plot(sales\_ts, xlab="Tempo", ylab="Vendas",ylim=c(40000, 110000), type = "l")



plot(sales$Sales, xlab="Tempo", ylab="Vendas", ylim=c(40000, 110000), type="p")

 3. Calcular Resumo estatístico da base:

summary(sales)

## Year Quarter Sales   
## Min. :2005 Min. : 1.00 Min. : 48617   
## 1st Qu.:2006 1st Qu.: 6.75 1st Qu.: 52681   
## Median :2008 Median :12.50 Median : 59440   
## Mean :2008 Mean :12.50 Mean : 64757   
## 3rd Qu.:2009 3rd Qu.:18.25 3rd Qu.: 76835   
## Max. :2010 Max. :24.00 Max. :103337

summary(sales\_ts)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 48617 52681 59440 64757 76835 103337

1. Separar em amostra de desenvolvimento e teste da seguinte maneira 4.1 Desenvolvimento: 2005/1 até 2009/4

tam\_amostra\_teste <- 4  
tam\_amostra\_treinamento <- length(sales\_ts) - tam\_amostra\_teste  
treinamento\_ts <- window(sales\_ts, start=c(2005, 1), end=c(2005,tam\_amostra\_treinamento))  
treinamento\_ts

## Qtr1 Qtr2 Qtr3 Qtr4  
## 2005 50147 49325 57048 76781  
## 2006 48617 50898 58517 77691  
## 2007 50862 53028 58849 79660  
## 2008 51640 54119 65681 85175  
## 2009 56405 60031 71486 92183

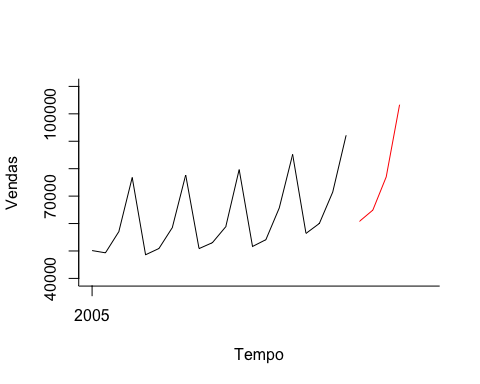
4.2 Teste: 2010/1 até 2010/4

validacao\_ts <- window(sales\_ts, start=c(2005, tam\_amostra\_treinamento + 1), end=c(2005,tam\_amostra\_treinamento+tam\_amostra\_teste))  
validacao\_ts

## Qtr1 Qtr2 Qtr3 Qtr4  
## 2010 60800 64900 76997 103337

Avaliação gráfica do treinamento e validação

plot(treinamento\_ts, xlab="Tempo", ylab="Vendas", xaxt="n" , ylim=c(40000, 110000), xlim=c(2005, 2011.25), bty="l")  
  
axis(1, at=seq(2005, 2010, 24), labels=format(seq(2005, 2010,24)))  
  
lines(validacao\_ts, bty="l", col="red")



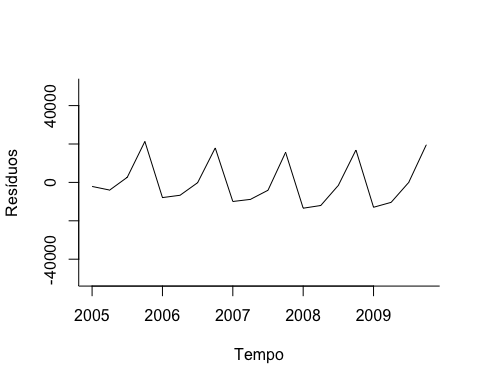
1. Estimar um modelo de tendência linear

modelo\_tendencia\_linear <- tslm(treinamento\_ts ~ trend)  
summary(modelo\_tendencia\_linear)

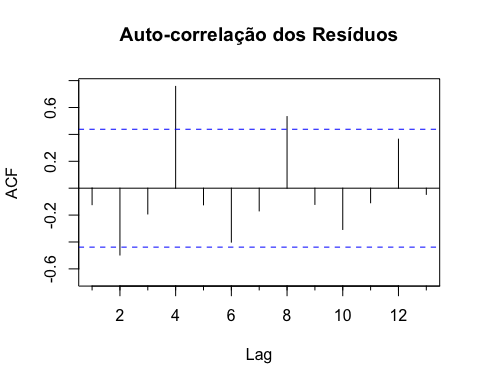
##   
## Call:  
## tslm(formula = treinamento\_ts ~ trend)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13439 -9119 -3051 5906 21322   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 51183.7 5609.0 9.125 3.58e-08 \*\*\*  
## trend 1068.9 468.2 2.283 0.0348 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 12070 on 18 degrees of freedom  
## Multiple R-squared: 0.2245, Adjusted R-squared: 0.1814   
## F-statistic: 5.211 on 1 and 18 DF, p-value: 0.03482

Plotar resíduos

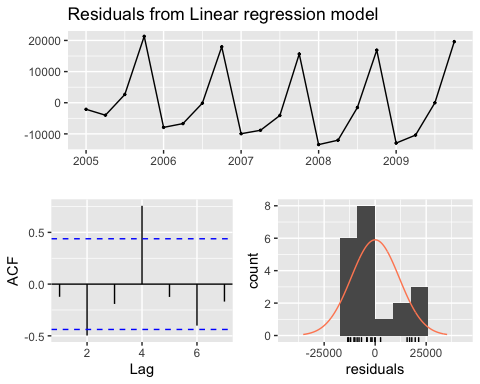
plot(modelo\_tendencia\_linear$residuals, xlab="Tempo", ylab="Resíduos", ylim=c(-50000, 50000), bty="l")

 Calcular a auto-correlação dos resíduos

Acf(modelo\_tendencia\_linear$residuals, main="Auto-correlação dos Resíduos")

 Verificar resíduos com teste de “Ljung-Box”

checkresiduals(modelo\_tendencia\_linear, test="LB")



##   
## Ljung-Box test  
##   
## data: Residuals from Linear regression model  
## Q\* = 23.539, df = 3, p-value = 3.117e-05  
##   
## Model df: 2. Total lags used: 5

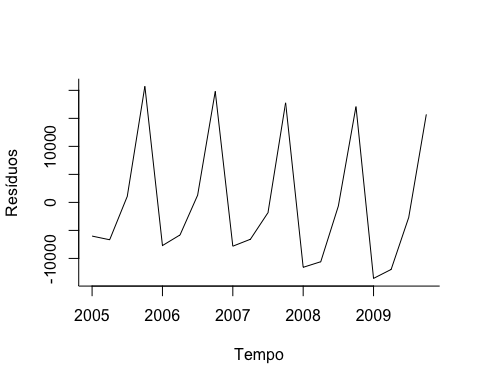
1. Estimar um modelo de tendência quadrática (polinômio de grau 2)

modelo\_tendencia\_poli <- tslm(treinamento\_ts ~ trend + I(trend^2))  
modelo\_tendencia\_poli

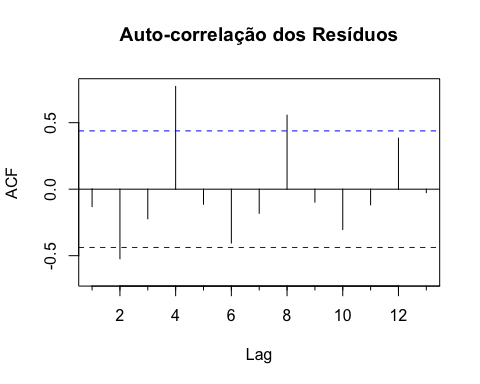
##   
## Call:  
## tslm(formula = treinamento\_ts ~ trend + I(trend^2))  
##   
## Coefficients:  
## (Intercept) trend I(trend^2)   
## 56462.70 -370.83 68.56

Plotar resíduos

plot(modelo\_tendencia\_poli$residuals, xlab="Tempo", ylab="Resíduos", bty="l")

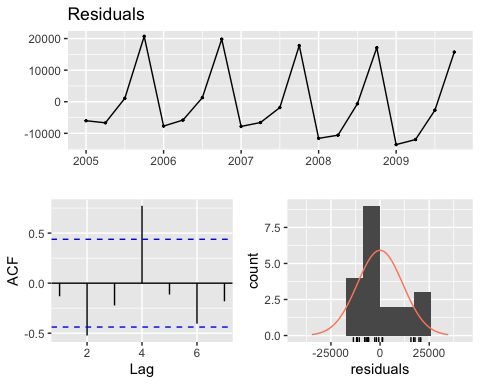
 Calcular a auto-correlação dos resíduos

Acf(modelo\_tendencia\_poli$residuals, main="Auto-correlação dos Resíduos")

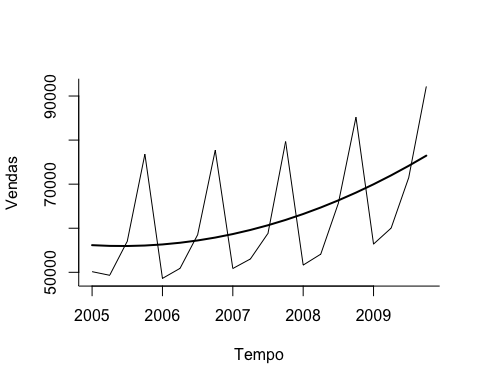
 Verificar resíduos com teste de “Ljung-Box”

checkresiduals(modelo\_tendencia\_poli$residuals, test="LB")

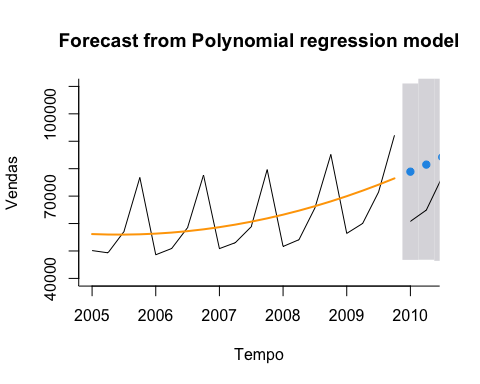
## Warning in modeldf.default(object): Could not find appropriate degrees of  
## freedom for this model.

 plot do modelo com tendência

plot(treinamento\_ts, xlab="Tempo", ylab="Vendas", bty="l")  
lines(modelo\_tendencia\_poli$fitted.values, lwd=2)

 Projeta o modelo no período de Validação

modelo\_tendencia\_poli\_proj <- forecast(modelo\_tendencia\_poli, h = tam\_amostra\_teste, level=0.95)  
plot(modelo\_tendencia\_poli\_proj, xlab="Tempo", ylab="Vendas", xaxt="n" , ylim=c(40000, 110000), xlim=c(2005, 2010.25), bty="l", flty=2,main="Forecast from Polynomial regression model")  
  
axis(1, at=seq(2005, 2010, 1), labels=format(seq(2005, 2010,1)))  
  
lines(validacao\_ts)  
lines(modelo\_tendencia\_poli\_proj$fitted, lwd=2, col="orange")

 Verificar a acurácia do modelo:

accuracy(modelo\_tendencia\_poli\_proj, validacao\_ts)

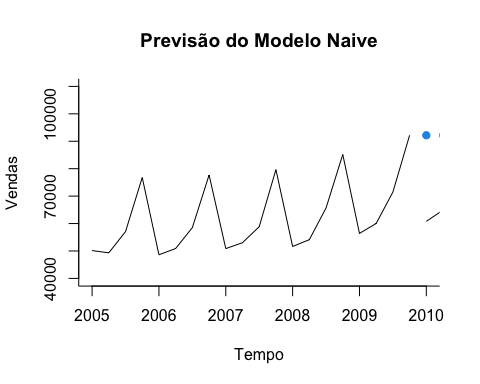
## ME RMSE MAE MPE MAPE MASE  
## Training set 1.455458e-12 11273.48 9346.619 -2.920362 14.51938 2.999076  
## Test set -6.403989e+03 15167.09 14546.239 -12.235160 20.11448 4.667492  
## ACF1 Theil's U  
## Training set -0.1307409 NA  
## Test set 0.1444087 0.9171101

Calular o modelo ingenuo e verifica a sua acurácia:

modelo\_ingenuo <- naive(treinamento\_ts, level=0, h=tam\_amostra\_teste)  
accuracy(modelo\_ingenuo, validacao\_ts)

## ME RMSE MAE MPE MAPE MASE  
## Training set 2212.421 17122.92 14065.58 -0.3423223 22.56419 4.513261  
## Test set -15674.500 22826.96 21251.50 -25.6460832 31.04299 6.819028  
## ACF1 Theil's U  
## Training set -0.2899524 NA  
## Test set 0.1722954 1.331418

plot(modelo\_ingenuo, xlab="Tempo", ylab="Vendas", xaxt="n" ,main="Previsão do Modelo Naive", ylim=c(40000, 110000), xlim=c(2005, 2010), bty="l", flty=2)  
  
axis(1, at=seq(2005, 2010, 1), labels=format(seq(2005, 2010,1)))  
  
lines(validacao\_ts)

 Projetar o futuro

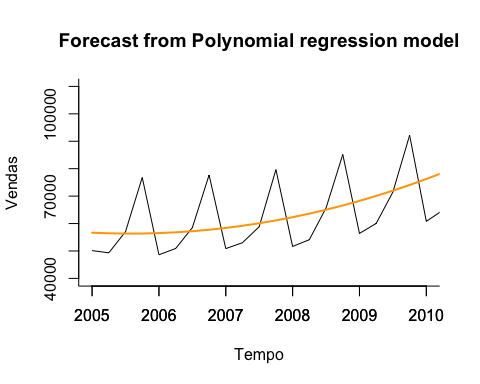
modelo\_tendencia\_poli\_final <- tslm(sales\_ts ~ trend + I(trend^2))  
summary(modelo\_tendencia\_poli\_final)

##   
## Call:  
## tslm(formula = sales\_ts ~ trend + I(trend^2))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15444 -8297 -4950 5517 20416   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 57028.00 8488.12 6.719 1.2e-06 \*\*\*  
## trend -420.03 1564.42 -0.268 0.791   
## I(trend^2) 63.57 60.75 1.046 0.307   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 12720 on 21 degrees of freedom  
## Multiple R-squared: 0.3398, Adjusted R-squared: 0.277   
## F-statistic: 5.405 on 2 and 21 DF, p-value: 0.01277

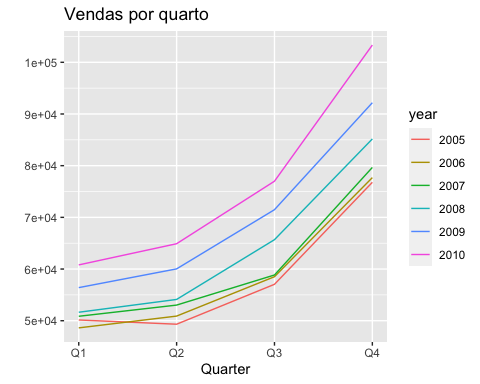
modelo\_tendencia\_poli\_final\_proj <- forecast(modelo\_tendencia\_poli\_final, h=36, level=0.95)

PLotagem do futuro

plot(modelo\_tendencia\_poli\_final\_proj, xlab="Tempo", ylab="Vendas", ylim=c(40000, 110000), xlim=c(2005, 2010), bty="l", flty=2, main="Forecast from Polynomial regression model")  
axis(1, at=seq(2005, 2010, 1), labels=format(seq(2005, 2010,1)))  
lines(modelo\_tendencia\_poli\_final\_proj$fitted, lwd=2, col="orange")

 7. Estimar um modelo de tendência linear com sazonalidade

ggseasonplot(sales\_ts, main="Vendas por quarto")

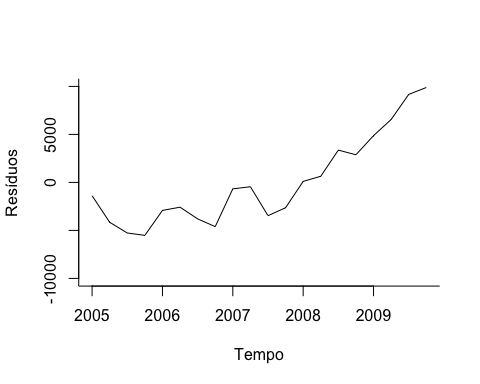


dummies\_mensais <- seasonaldummy(sales\_ts)  
modelo\_sazonalidade\_linear <- tslm(treinamento\_ts ~ season)  
summary(modelo\_sazonalidade\_linear)

##   
## Call:  
## tslm(formula = treinamento\_ts ~ season)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5517 -3550 -1030 2999 9885   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 51534 2281 22.591 1.45e-13 \*\*\*  
## season2 1946 3226 0.603 0.55483   
## season3 10782 3226 3.342 0.00414 \*\*   
## season4 30764 3226 9.536 5.30e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5101 on 16 degrees of freedom  
## Multiple R-squared: 0.877, Adjusted R-squared: 0.8539   
## F-statistic: 38.02 on 3 and 16 DF, p-value: 1.652e-07

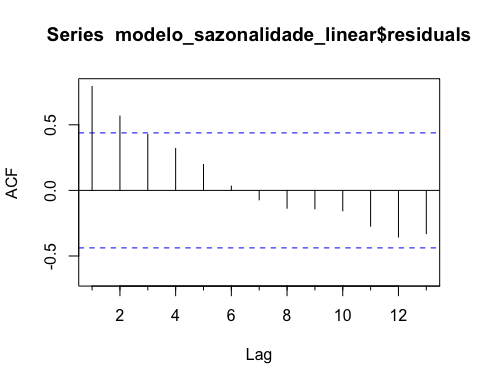
PLotando

plot(modelo\_sazonalidade\_linear$residuals, xlab="Tempo", ylab="Resíduos", ylim=c(-10000, 10000), bty="l")

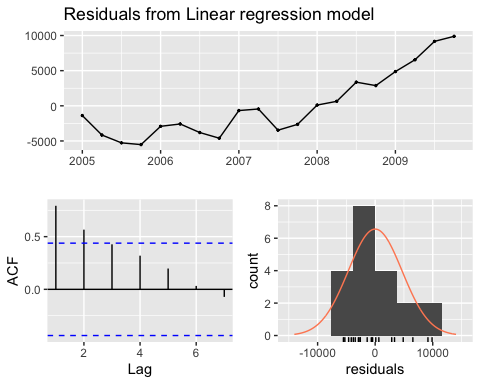


Calcula a Autocorrelação dos resíduos

Acf(modelo\_sazonalidade\_linear$residuals)

 Checagem dos resíduos com teste de Ljung - Box

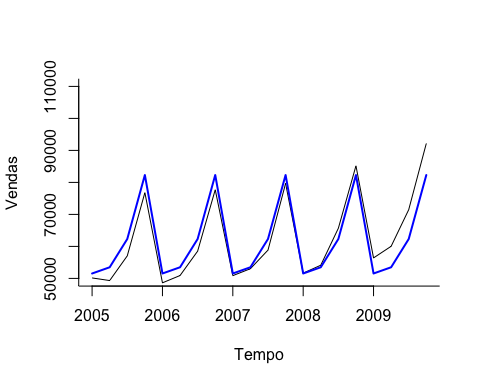
checkresiduals(modelo\_sazonalidade\_linear, test="LB")



##   
## Ljung-Box test  
##   
## data: Residuals from Linear regression model  
## Q\* = 31.276, df = 3, p-value = 7.437e-07  
##   
## Model df: 4. Total lags used: 7

PLota o modelo com sazonalidade

plot(treinamento\_ts, xlab="Tempo", ylab="Vendas", ylim=c(50000, 110000), bty="l")  
lines(modelo\_sazonalidade\_linear$fitted.values, lwd=2, col="blue")

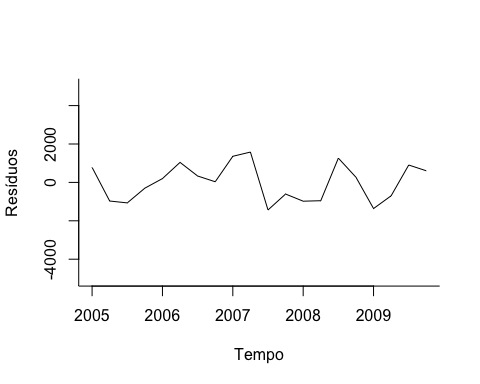
 8. Estimar um modelo de tendência quadrática (polinômio de grau 2) com sazonalidade Criar o modelo

modelo\_sazonal\_tend\_linear <- tslm(treinamento\_ts ~ season + trend + I(trend^2))  
summary(modelo\_sazonal\_tend\_linear)

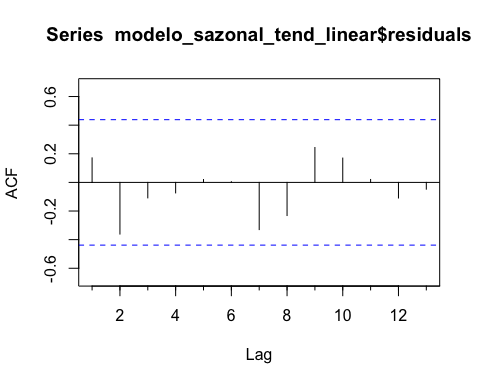
##   
## Call:  
## tslm(formula = treinamento\_ts ~ season + trend + I(trend^2))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1435.4 -957.9 116.3 812.9 1579.6   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 49919.124 914.665 54.576 < 2e-16 \*\*\*  
## season2 1357.363 715.862 1.896 0.07877 .   
## season3 9477.738 720.025 13.163 2.83e-09 \*\*\*  
## season4 28616.925 726.712 39.379 9.66e-16 \*\*\*  
## trend -617.750 184.575 -3.347 0.00479 \*\*   
## I(trend^2) 63.494 8.528 7.445 3.13e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1129 on 14 degrees of freedom  
## Multiple R-squared: 0.9947, Adjusted R-squared: 0.9928   
## F-statistic: 527.9 on 5 and 14 DF, p-value: 2.013e-15

Plotando Resíduos

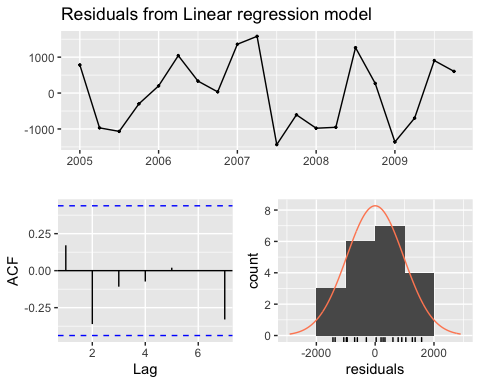
plot(modelo\_sazonal\_tend\_linear$residuals, xlab="Tempo", ylab="Resíduos", ylim=c(-5000, 5000), bty="l")

 Calcular a autocorrelação dos resíduos

Acf(modelo\_sazonal\_tend\_linear$residuals)

 Aplicando o teste de Ljung-Box

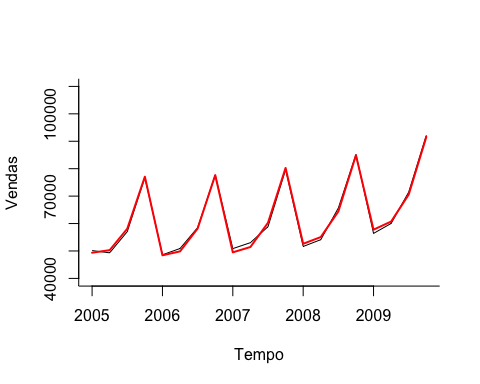
checkresiduals(modelo\_sazonal\_tend\_linear, test="LB")



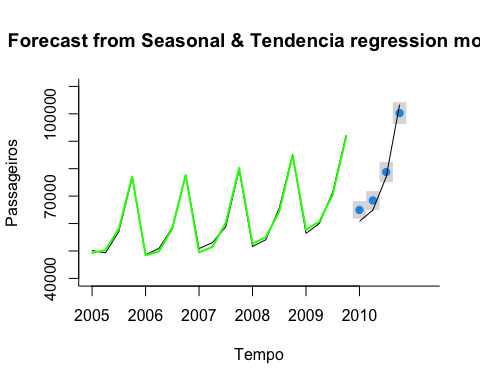
##   
## Ljung-Box test  
##   
## data: Residuals from Linear regression model  
## Q\* = 12.353, df = 3, p-value = 0.006266  
##   
## Model df: 6. Total lags used: 9

Plotagem do modelo sazonal com tendência

plot(treinamento\_ts, xlab="Tempo", ylab="Vendas", ylim=c(40000, 110000), bty="l")  
lines(modelo\_sazonal\_tend\_linear$fitted.values, lwd=2, col="red")

 Modelo durante a validação:

modelo\_sazonal\_tend\_linear\_proj <- forecast(modelo\_sazonal\_tend\_linear, h = tam\_amostra\_teste, level=0.95)  
plot(modelo\_sazonal\_tend\_linear\_proj, xlab="Tempo", ylab="Passageiros", xaxt="n" , ylim=c(40000, 110000), xlim=c(2005, 2011.25), bty="l", flty=2, main="Forecast from Seasonal & Tendencia regression model")  
  
axis(1, at=seq(2005, 2010, 1), labels=format(seq(2005, 2010,1)))  
  
lines(validacao\_ts)  
lines(modelo\_sazonal\_tend\_linear\_proj$fitted, lwd=2, col="green")

 9. Calcular os erros de projeção para cada modelo: modelo de tendência linear

summary(modelo\_tendencia\_linear)

##   
## Call:  
## tslm(formula = treinamento\_ts ~ trend)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13439 -9119 -3051 5906 21322   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 51183.7 5609.0 9.125 3.58e-08 \*\*\*  
## trend 1068.9 468.2 2.283 0.0348 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 12070 on 18 degrees of freedom  
## Multiple R-squared: 0.2245, Adjusted R-squared: 0.1814   
## F-statistic: 5.211 on 1 and 18 DF, p-value: 0.03482

modelo Naive

accuracy(modelo\_ingenuo, validacao\_ts)

## ME RMSE MAE MPE MAPE MASE  
## Training set 2212.421 17122.92 14065.58 -0.3423223 22.56419 4.513261  
## Test set -15674.500 22826.96 21251.50 -25.6460832 31.04299 6.819028  
## ACF1 Theil's U  
## Training set -0.2899524 NA  
## Test set 0.1722954 1.331418

modelo de tendência quadrática (polinômio de grau 2)

accuracy(modelo\_tendencia\_poli\_proj, validacao\_ts)

## ME RMSE MAE MPE MAPE MASE  
## Training set 1.455458e-12 11273.48 9346.619 -2.920362 14.51938 2.999076  
## Test set -6.403989e+03 15167.09 14546.239 -12.235160 20.11448 4.667492  
## ACF1 Theil's U  
## Training set -0.1307409 NA  
## Test set 0.1444087 0.9171101

modelo de tendência quadrática (polinômio de grau 2) com sazonalidade

accuracy(modelo\_sazonal\_tend\_linear\_proj, validacao\_ts)

## ME RMSE MAE MPE MAPE MASE  
## Training set 0.000 944.8833 837.1447 -0.01101701 1.455328 0.268617  
## Test set -1597.486 3242.9622 3124.6746 -2.89909815 4.376970 1.002623  
## ACF1 Theil's U  
## Training set 0.1713358 NA  
## Test set 0.1379815 0.1904888

1. Escolher o melhor modelo de projeção justificando O melhor modelo de projeção foi o modelo de tendência quadrática (polinômio de grau 2) com sazonalidade, pois no teste de resíduos não houve auto correlação dos resíduos, o MAPE é menor do que o modelo Naive.