Análise de Séries Temporais - Trabalho 2

Davi Guerra Alves - Henrique Oliveira Dumay

2023-07-02

Apresentação

A série analisada consiste na série número 1686 pertencente ao banco de dados da competição de previsão M3, disponível no pacote *Mcomp* do software R. A série descreve o número de carregamentos com código *TD-AUTOUNITS*, mensalmente, de outubro de 1984 a setembro de 1993.

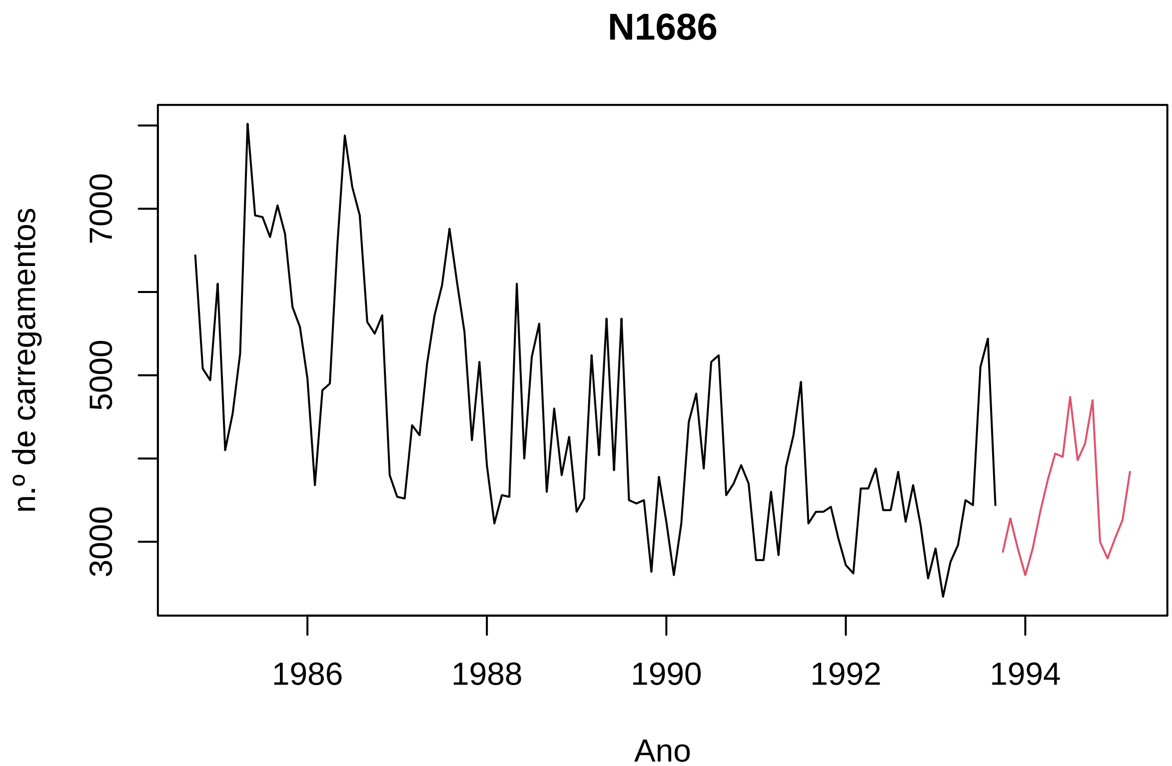


Figure 1: Comportamento da série ao longo do tempo

Decomposição MSTL

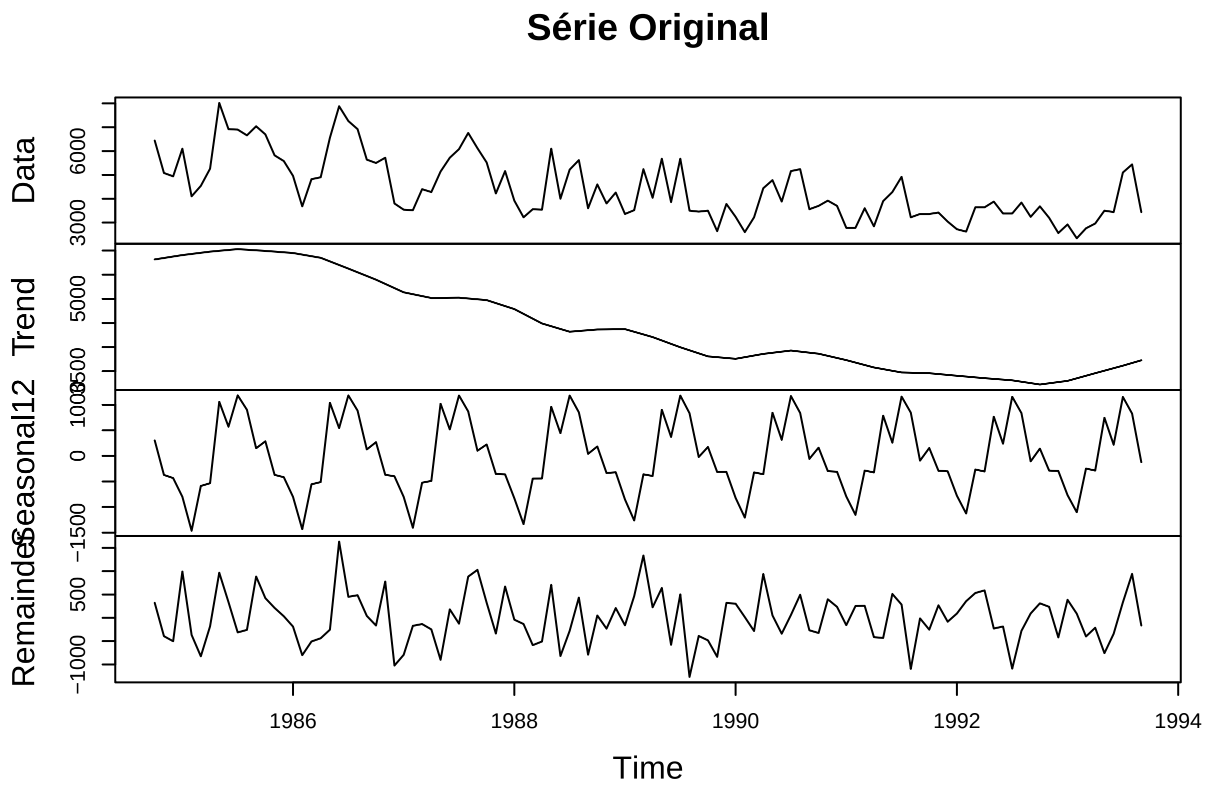


Figure 2: Decomposição MSTL

A decomposição MSTL mostra os componentes de tendência, sazonalidade e erro da série estudada. Percebe-se a presença de uma tendência crescente, com múltiplas sazonalidades que apesentam mudança do comportamento ao longo do tempo. É possível observar, graficamente, um alargamento da sazonalidade ao fim da série quando comparando ao início da série.

Modelos ARIMA

A presença do componente de tendência explicita a não-estacionaridade da série original.A função ndiffs() é utilizada para estimar o número de diferenças exigidas para tornar a série estacionária por meio de um teste de raíz unitária, com a hipótese nula de que a série tem raízes estacionárias contra a hipótese alternativa de que a série tem raíz unitária. O teste retorna o menor número de diferenças exigidas para o teste em um nível de significância de 95%. Já a função nsdiffs() utiliza testes de raíz unitária para determinar o número de diferenças sazonais para tornar a série estacionária.

Com o uso das funções acima, obteve-se o valor para $d = 1$ e $D = 0$. Os modelos candidatos terão a forma:

$$SARIMA(p, 1, q) \times (P, 0, Q)_{12}$$

A estacionariedade da série pode ser testada utilizando o teste Kwiatkowski-Phillips-Schmidt-Shin (KPSS), com a hipótese nula de que a série é estacionária. O teste resulta em um valor de 0.0221285, com p-valor de 0.1, que não nos permite rejeitar a hipótese nula a um nível de significância $\alpha = 0,05$.

Consideramos que a série é, agora, estacionária, observamos os gráficos da função de autocorrelação (ACF) e da função de autocorrelação parcial (PACF) em busca de possíveis autocorrelações entre os diferentes atrasos da série. Os gráficos a seguir ilustram a série diferenciada, assim como os gráficos das funções de ACF e PACF.

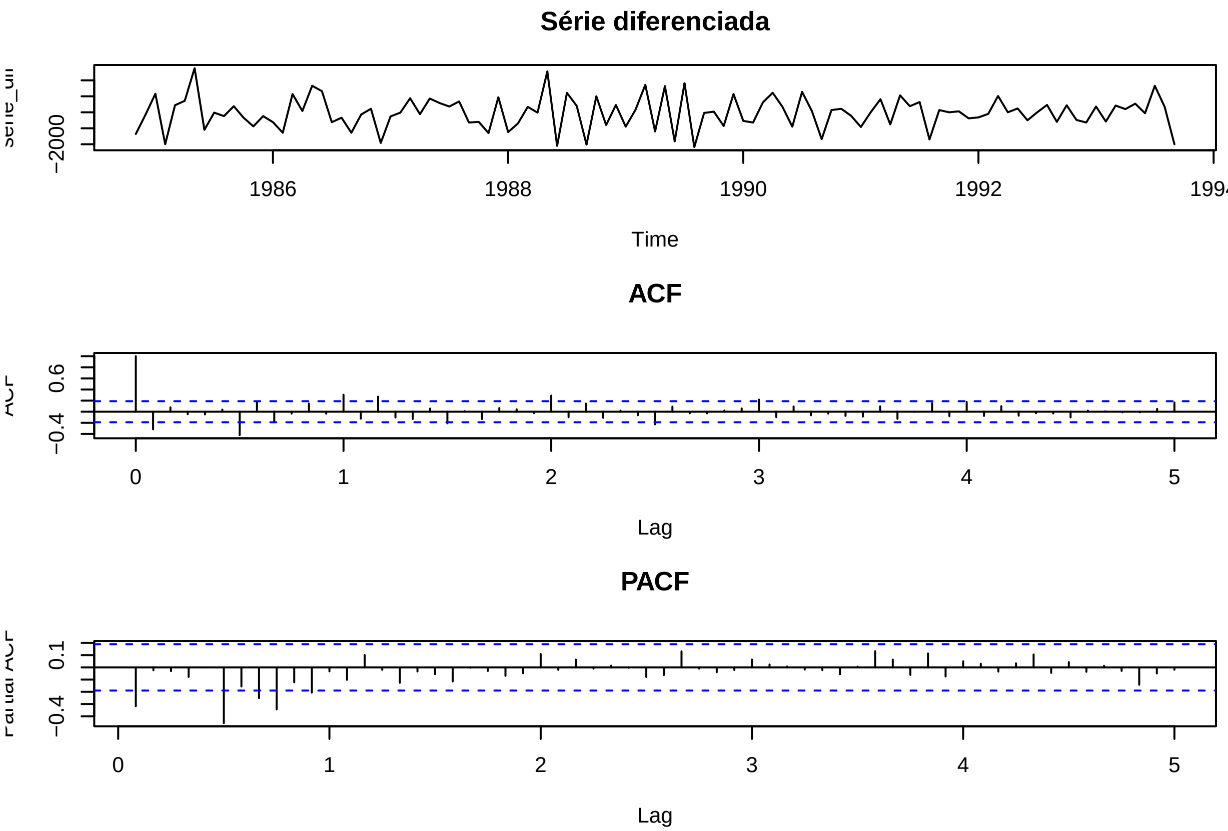


Figure 3: Gráficos ACF e PACF

Dos gráficos apresentados, pode-se afirmar que a série diferenciada não apresenta um padrão claro de autocorrelações simples e sazonais que permita inferir diretamente a modelagem. Neste sentido, serão testados valores diferentes para p, P, q e Q e os diferentes modelos serão comparados por meio do critério AIC.

Para os diferentes valores de (p, q, P, Q) teremos:

```
## p = 0 , q = 0 , P = 0 , Q = 0 , AICc = 1782.086
## p = 0 , q = 1 , P = 0 , Q = 0 , AICc = 1772.446
## p = 0 , q = 3 , P = 0 , Q = 0 , AICc = 1765.736
## p = 2 , q = 3 , P = 0 , Q = 0 , AICc = 1745.548
## p = 0 , q = 2 , P = 1 , Q = 0 , AICc = 1745.396
## p = 1 , q = 1 , P = 1 , Q = 0 , AICc = 1744.079
## p = 2 , q = 2 , P = 1 , Q = 0 , AICc = 1743.72
## p = 3 , q = 3 , P = 1 , Q = 0 , AICc = 1741.292
## p = 0 , q = 1 , P = 1 , Q = 1 , AICc = 1739.715
```

```
## p = 0 , q = 2 , P = 1 , Q = 1 , AICc = 1735.097
## p = 1 , q = 1 , P = 1 , Q = 1 , AICc = 1733.713
```

O modelo com menor AICc foi o $SARIMA(1,1,1) \times (1,0,1)_{12}$.

Os coeficientes do modelo proposto, portanto, serão obtidos do cálculo da função Arima com o modelo acima proposto. Os coeficientes do modelo terão a seguinte forma:

$$\phi_1 = 0,3152; \theta_1 = -0,9218; \varphi = 0,9606; \vartheta = -0,7359$$

Utilizaremos o modelo ARIMA acima definido com a transformação de Box-Cox com o objetivo de estabilizar os diferentes tipos de variação ao longo do tempo. O novo modelo tem os seguintes coeficientes:

$$\phi_1 = 0,2756; \theta_1 = -0,9209; \varphi = 0,9994; \vartheta = -0,9653$$

Análise de Resíduos

Os resíduos do modelo ARIMA sem transformação apresentam o seguinte comportamentos gráficos:

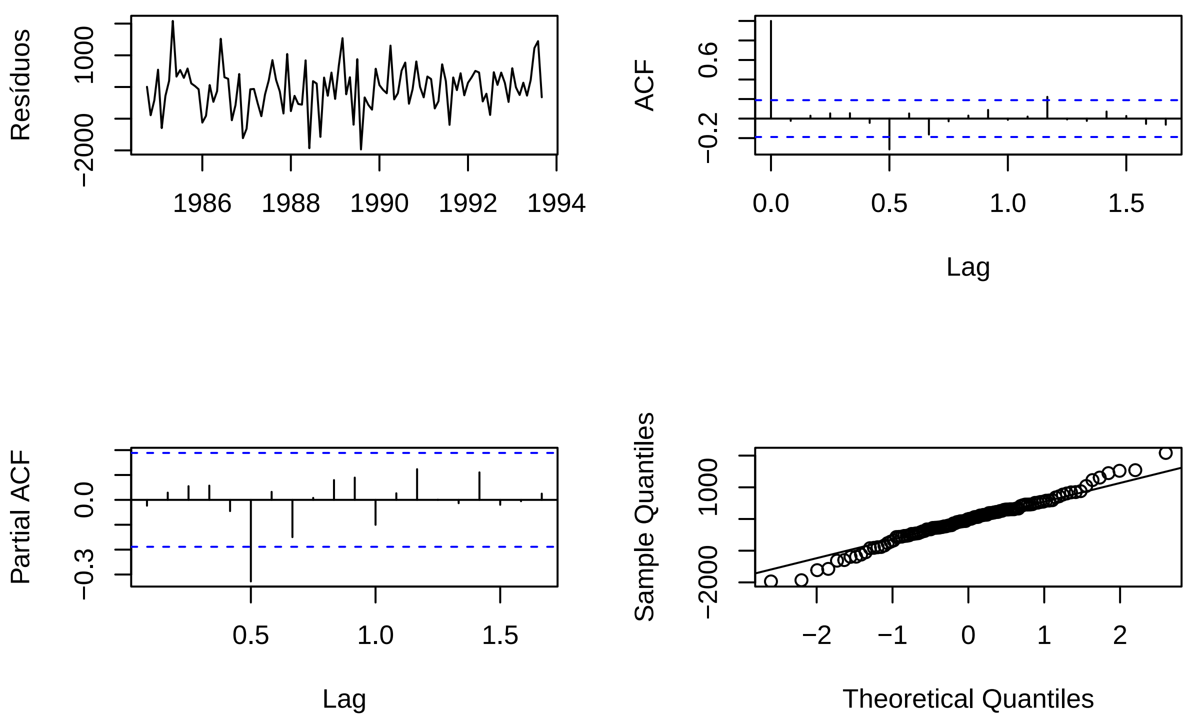


Figure 4: Resíduos ARIMA sem transformação

Já os resíduos do modelo ARIMA com transformação Box-Cox apresentam o seguinte comportamentos gráficos:

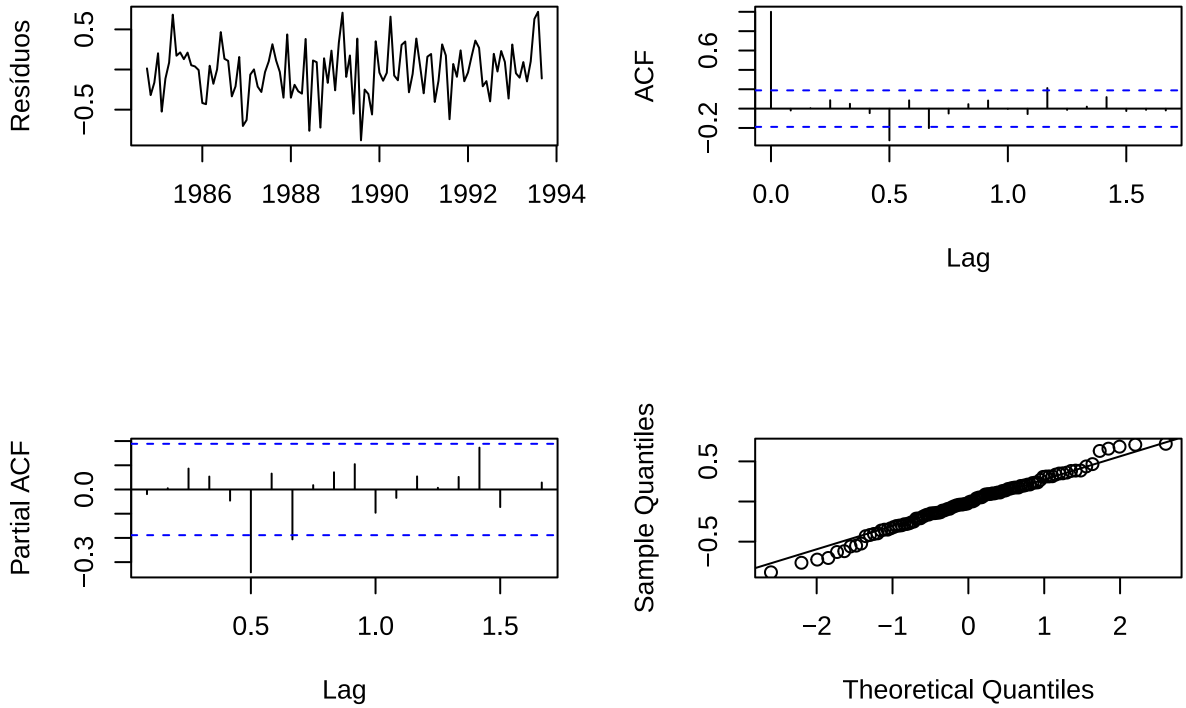


Figure 5: Resíduos ARIMA sem transformação

Graficamente, observa-se que os resíduos de ambos os modelos parecem distribuir-se simetricamente ao retor da origem e não apresentam autocorrelações bem definidas. Precisa-se, entretanto, testá-los para estacionariedade, independência e distribuição normal. Essas hipóteses serão testadas conforme se segue.

A estacionaridade será testada a partir do teste Kwiatkowski-Phillips-Schmidt-Shin (KPSS), com a hipótese nula de que a série é estacionária. O teste para o modelo $SARIMA(1, 1, 1) \times (1, 0, 1)_{12}$ e para o mesmo modelo, utilizando a transformação de Box-Cox:

Modelo	KPSS	P-valor
SARIMA sem Box-Cox	0.2133231	0.1
SARIMA com Box-Cox	0.2075153	0.1

De acordo com o teste KPSS, não se pode rejeitar a hipótese de estacionariedade dos resíduos de ambos os modelos.

O teste de independência dos resíduos é realizado a partir do teste Ljung-Box, com a hipótese H_0 de que os resíduos são idenpendentemente distribuídos. O teste apresenta os seguintes valores para os dois modelos:

Modelo	Chi-Quadrado	Graus de liberdade	P-valor
SARIMA sem Box-Cox	23.72163	15	0.069974
SARIMA com Box-Cox	27.0777088	15	0.0281095

Os resultados acima mostram que a independência dos resíduos pode ser rejeitada ao nível de significância de 5% no modelo que utiliza a transformação de Box-Cox, enquanto não pode ser rejeitada no modelo SARIMA natural.

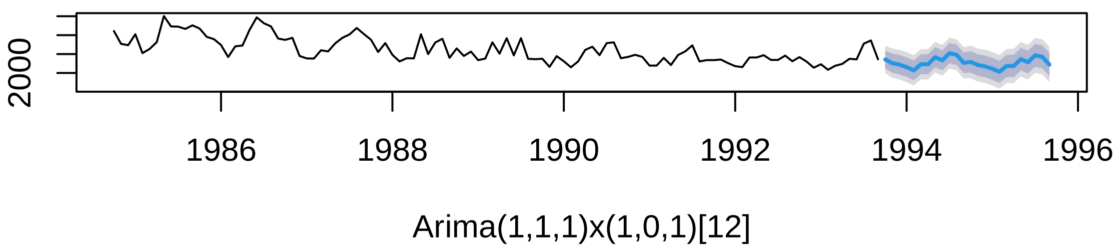
A normalidade dos resíduos é testada com o teste Shapiro-Wilk de Normalidade, com H_0 de que os resíduos apresentam distribuição normal. O valor do teste estatístico para os dois modelos trabalhados é:

Modelo	W	P-valor
SARIMA sem Box-Cox	0.986694	0.3620588
SARIMA com Box-Cox	0.9897878	0.5935039

Do resultado acima, não se pode rejeitar a hipótese de normalidade dos resíduos de ambos os modelos.

Previsões

As previsões dos modelos:



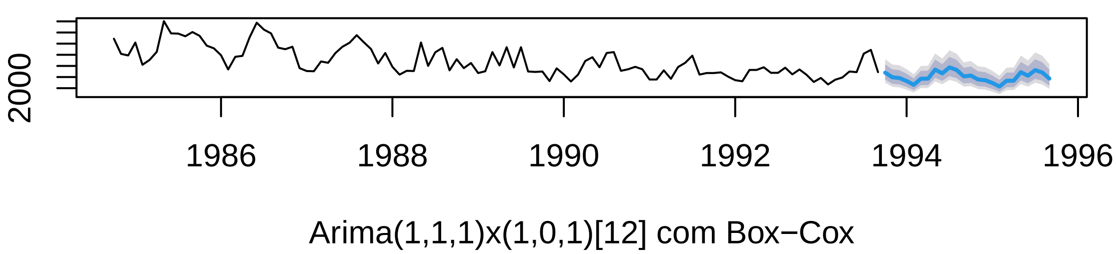


Figure 6: Previsões dos modelos ARIMA

Modelos ETS

O modelo ETS (Error, trend and seasonal) permite descrever os modelos de alisamento exponencial em função dos tipos de suas componentes: tendência, sazonalidade e erro. O modelo utiliza três caracteres como identificação de acordo com a terminologia adotada por Hyndman et al. (2002) e Hyndman et al. (2008). A primeira letra se refere ao componente do erro; a segunda, ao componente da tendência e a terceira, da sazonalidade. A série anteriormente descrita apresenta tendência e sazonalidade claras à decomposição realizada e, portanto, trabalharemos com componentes de modelagem que contenham essas características.

```r
componentes_tendencia = c("A","M")
componentes_sazonais = c("N","A","M")
componentes_erros = c("A","M")

model_ets = function(y,model,damped){
  tryCatch({
    ets(y,model,damped)
  },
  error=function(cond)print('nao pode')
  )
}

damped = F
melhor_AICc = Inf
melhor_modelo = ""
for(comp_erro in componentes_erros){
  for(comp_tend in componentes_tendencia){
    for(comp_saz in componentes_sazonais){
      for(damped in c(T,F)){
        modelo = paste0(comp_erro,comp_tend,comp_saz)
        print("------------")
        print(modelo)
        print("damped=")
        print(damped)
        modelo_ets = model_ets(serie, model=modelo, damped = damped)
        print(modelo_ets['aicc'])
      }
    }
  }
}
```

```
## [1] "------------"
## [1] "AAN"
## [1] "damped="
## [1] TRUE
## $aicc
## [1] 1996.368
##
## [1] "------------"
## [1] "AAN"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1993.913
##
## [1] "------------"
## [1] "AAA"
## [1] "damped="
## [1] TRUE
```

```
## $aicc
## [1] 1953.429
##
## [1] "------------"
## [1] "AAA"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1950.568
##
## [1] "------------"
## [1] "AAM"
## [1] "damped="
## [1] TRUE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AAM"
## [1] "damped="
## [1] FALSE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AMN"
## [1] "damped="
## [1] TRUE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AMN"
## [1] "damped="
## [1] FALSE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AMA"
## [1] "damped="
## [1] TRUE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AMA"
## [1] "damped="
## [1] FALSE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AMM"
## [1] "damped="
## [1] TRUE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "AMM"
```

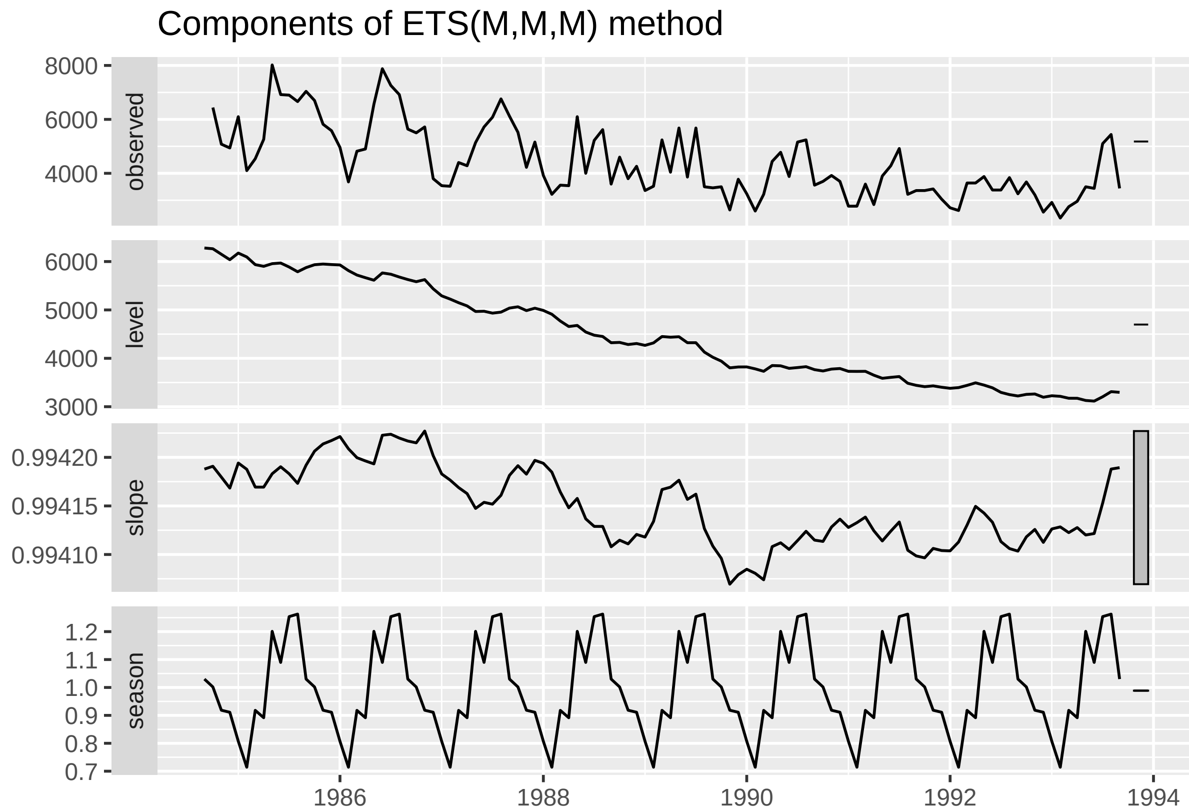
```
## [1] "damped="
## [1] FALSE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "MAN"
## [1] "damped="
## [1] TRUE
## $aicc
## [1] 1990.17
##
## [1] "------------"
## [1] "MAN"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1988.709
##
## [1] "------------"
## [1] "MAA"
## [1] "damped="
## [1] TRUE
## $aicc
## [1] 1942.956
##
## [1] "------------"
## [1] "MAA"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1947.757
##
## [1] "------------"
## [1] "MAM"
## [1] "damped="
## [1] TRUE
## $aicc
## [1] 1928.537
##
## [1] "------------"
## [1] "MAM"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1925.053
##
## [1] "------------"
## [1] "MMN"
## [1] "damped="
## [1] TRUE
## $aicc
## [1] 1991.342
##
## [1] "------------"
```

```
## [1] "MMN"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1988.949
##
## [1] "------------"
## [1] "MMA"
## [1] "damped="
## [1] TRUE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "MMA"
## [1] "damped="
## [1] FALSE
## [1] "nao pode"
## [1] NA
## [1] "------------"
## [1] "MMM"
## [1] "damped="
## [1] TRUE
## $aicc
## [1] 1928.128
##
## [1] "------------"
## [1] "MMM"
## [1] "damped="
## [1] FALSE
## $aicc
## [1] 1923.268
```

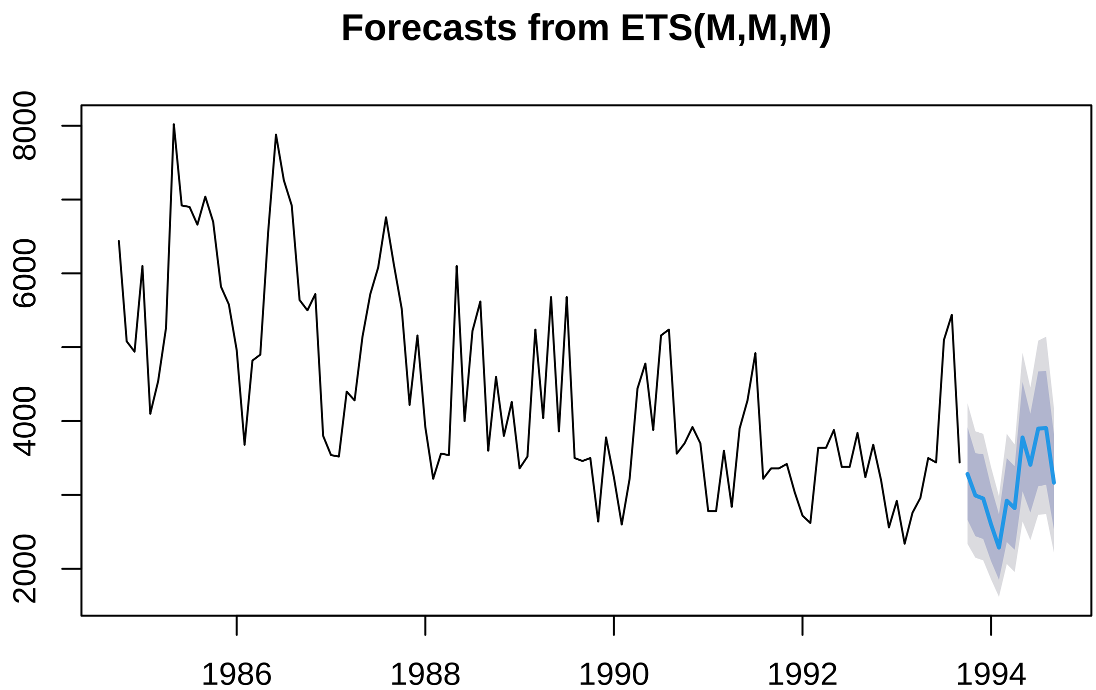
```
fit_ets = ets(serie, model="MMM",damped=F)
fit_ets$aicc
```

```
## [1] 1923.268
```

```
autoplot(fit_ets)
```



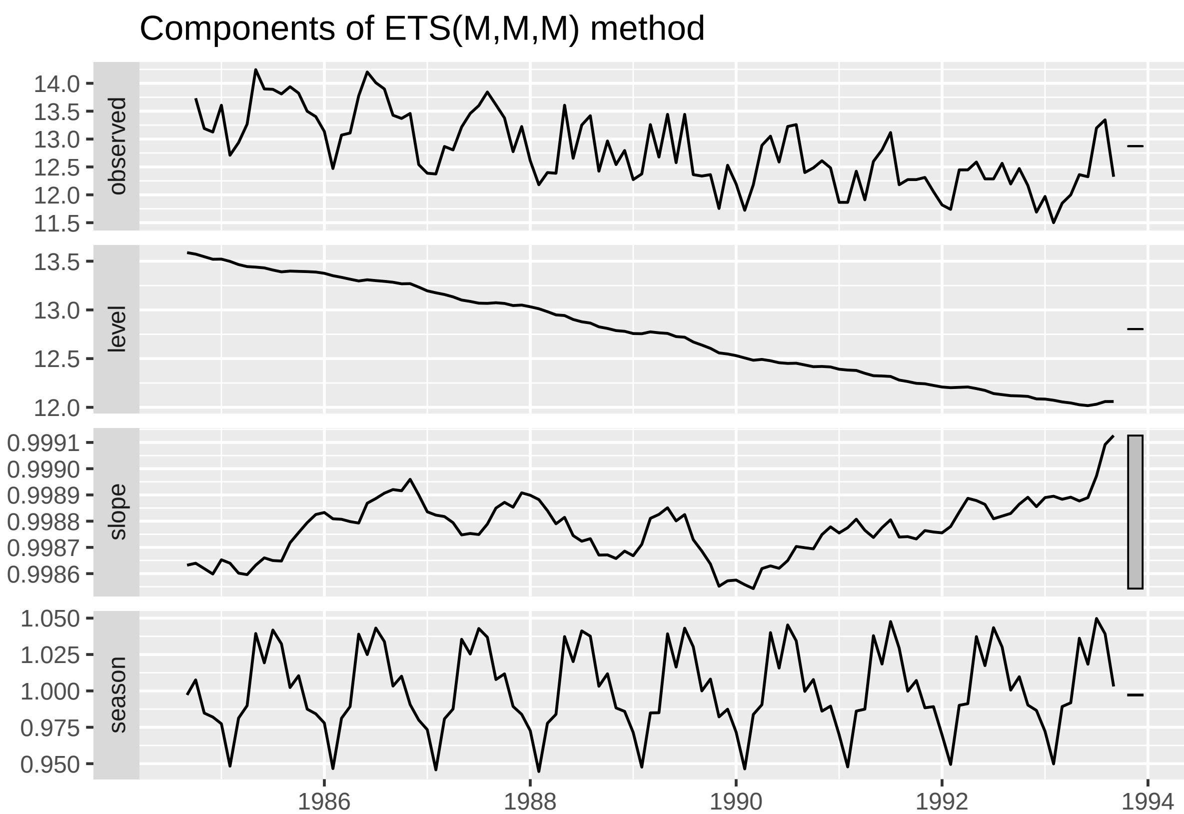

```
forecast(fit_ets,12) %>% plot
```



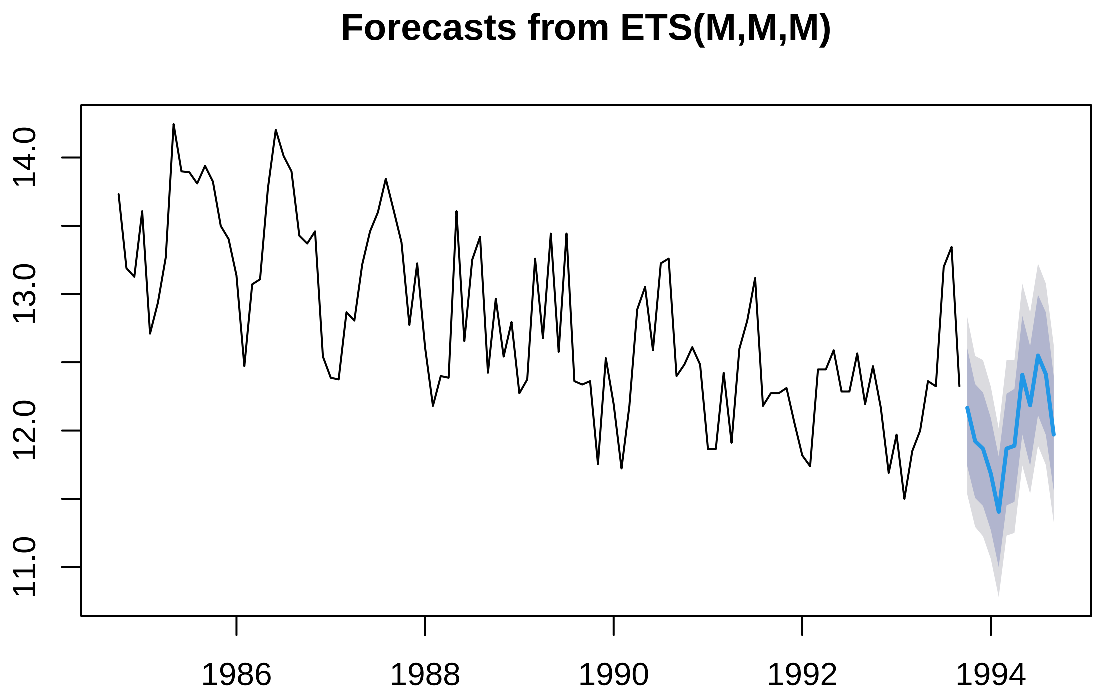

```
fit_ets_boxcox = ets(BoxCox(serie, lambda = 'auto'),model='MMM',damped=F)
fit_ets_boxcox
```

```
## ETS(M,M,M)
##
## Call:
##  ets(y = BoxCox(serie, lambda = "auto"), model = "MMM", damped = F)
##
##   Smoothing parameters:
##     alpha = 0.0427
##     beta  = 0.0015
##     gamma = 0.1126
##
##   Initial states:
##     l = 13.5884
##     b = 0.9986
##     s = 0.9972 1.0324 1.0426 1.0172 1.0367 0.9903
##            0.9841 0.9492 0.9735 0.9835 0.9862 1.007
##
##   sigma:  0.0273
##
##      AIC     AICc      BIC
## 294.0799 300.8799 339.6761
```

```
autoplot(fit_ets_boxcox)
```




```
forecast(fit_ets_boxcox,12) %>% plot
```




```r
ets1 <- ets(serie, model = "AAA", damped = FALSE)
ets2 <- ets(serie, model = "MAM", damped = FALSE)
ets3 <- ets(serie, model = "AAA", damped = TRUE)
ets4 <- ets(serie, model = "MAM", damped = TRUE)
ets1$aicc
```

```
## [1] 1950.568
```

```r
ets2$aicc
```

```
## [1] 1925.053
```

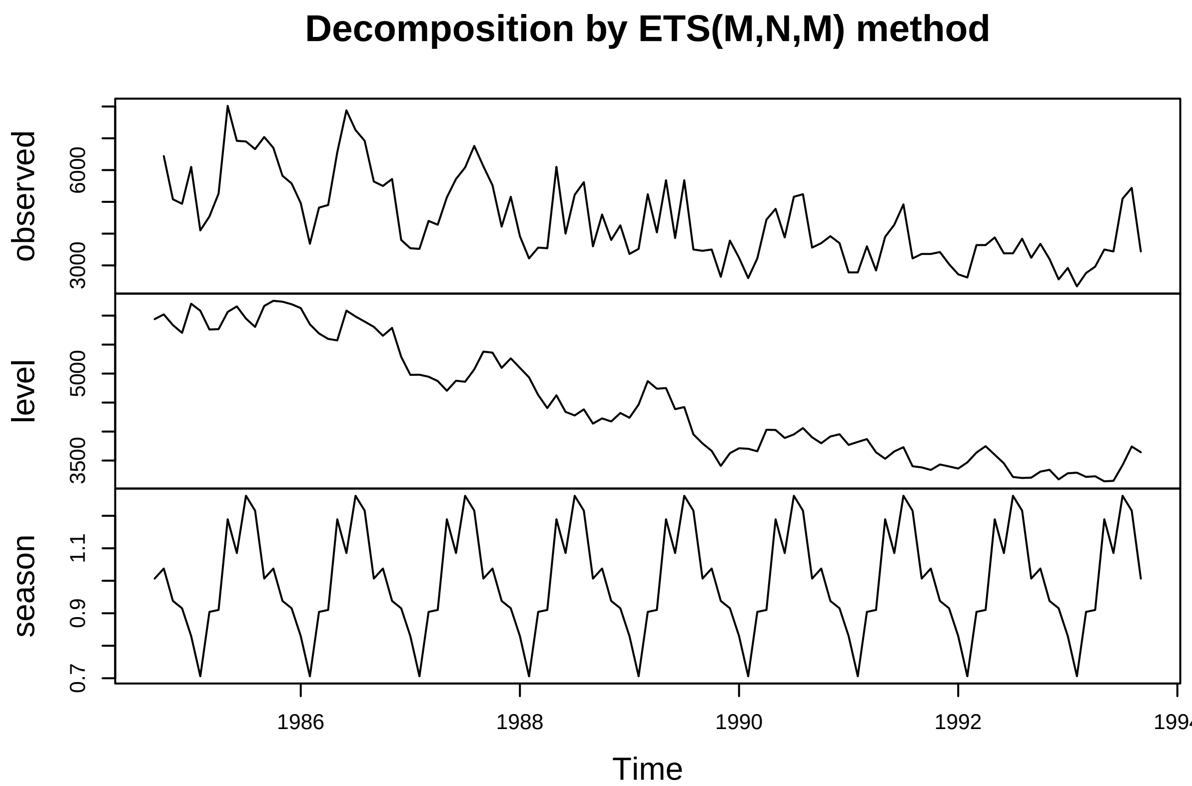
```r
ets3$aicc
```

```
## [1] 1953.429
```

```r
ets4$aicc
```

```
## [1] 1928.537
```

```r
ets(serie) %>% plot()
```




```r
#passo 1) aplicar cada um dos modelos ets na série
fit_ses = ets(serie, model = "ANN", damped=FALSE)
fit_holt = ets(serie, model = "AAN", damped=FALSE)
fit_holt_damped = ets(serie, model = "AAN", damped=TRUE)
fit_holt_aditivo = ets(serie, model = "AAA", damped=FALSE)
fit_holt_multi = ets(serie, model = "MAM", damped=FALSE)
```

```r
#passo 2) selecionar o melhor modelo aicc

fit_ses$aicc
```

```
## [1] 1989.167
```

```r
fit_holt$aicc
```

```
## [1] 1993.913
```

```r
fit_holt_damped$aicc
```

```
## [1] 1996.368
```

```
fit_holt_aditivo$aicc
```

```
## [1] 1950.568
```

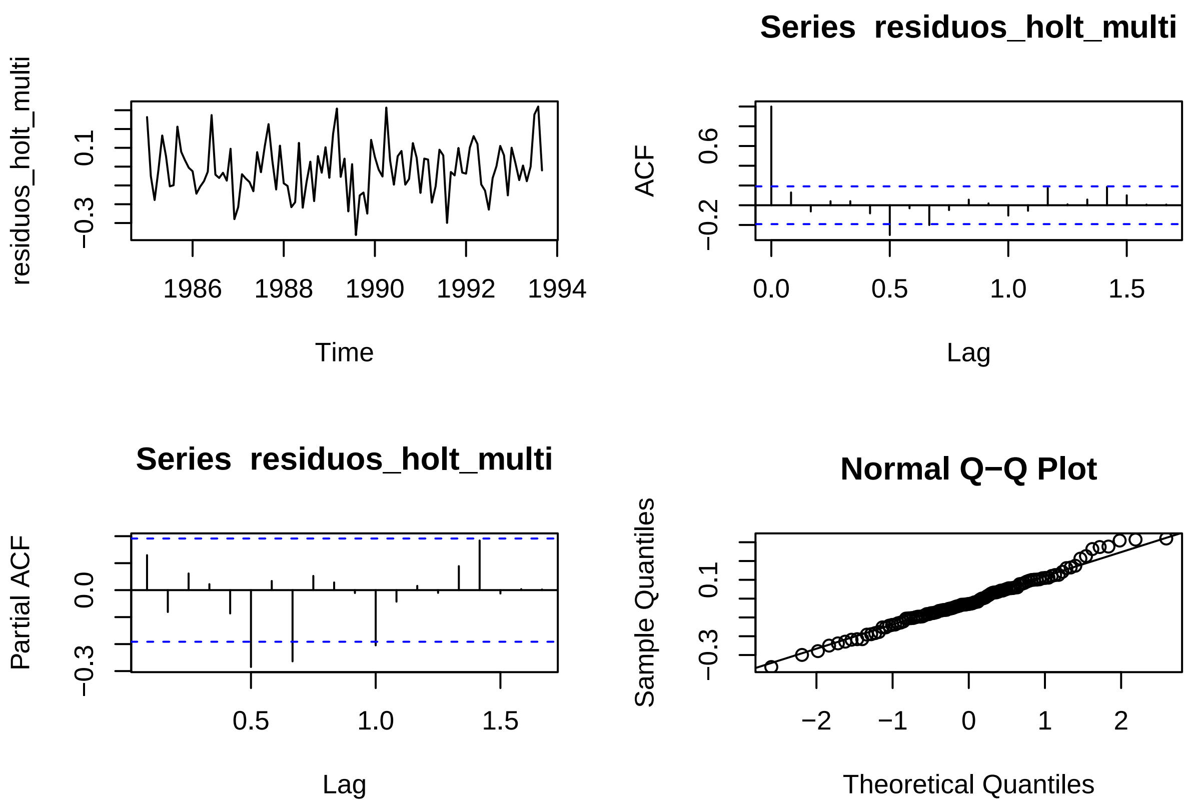
```
fit_holt_multi$aicc
```

```
## [1] 1925.053
```

```
#menor aicc foi do modelo  Holt Winter Multiplicativo, com o valor de 1925.053
```

```
#passo 3) verifique as suposições da série
residuos_holt_multi <- fit_holt_multi$residuals %>% window(start=1985)

par(mfrow=c(2,2))
plot(residuos_holt_multi)
acf(residuos_holt_multi)
pacf(residuos_holt_multi)
qqnorm(residuos_holt_multi)
qqline(residuos_holt_multi)
```




```
#estacionariedade
kpss.test(residuos_holt_multi)
```

```
## Warning in kpss.test(residuos_holt_multi): p-value greater than printed p-value
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  residuos_holt_multi
## KPSS Level = 0.094838, Truncation lag parameter = 4, p-value = 0.1
```

```
#independencia
Box.test(residuos_holt_multi, lag = 15, type ="Ljung-Box")
```

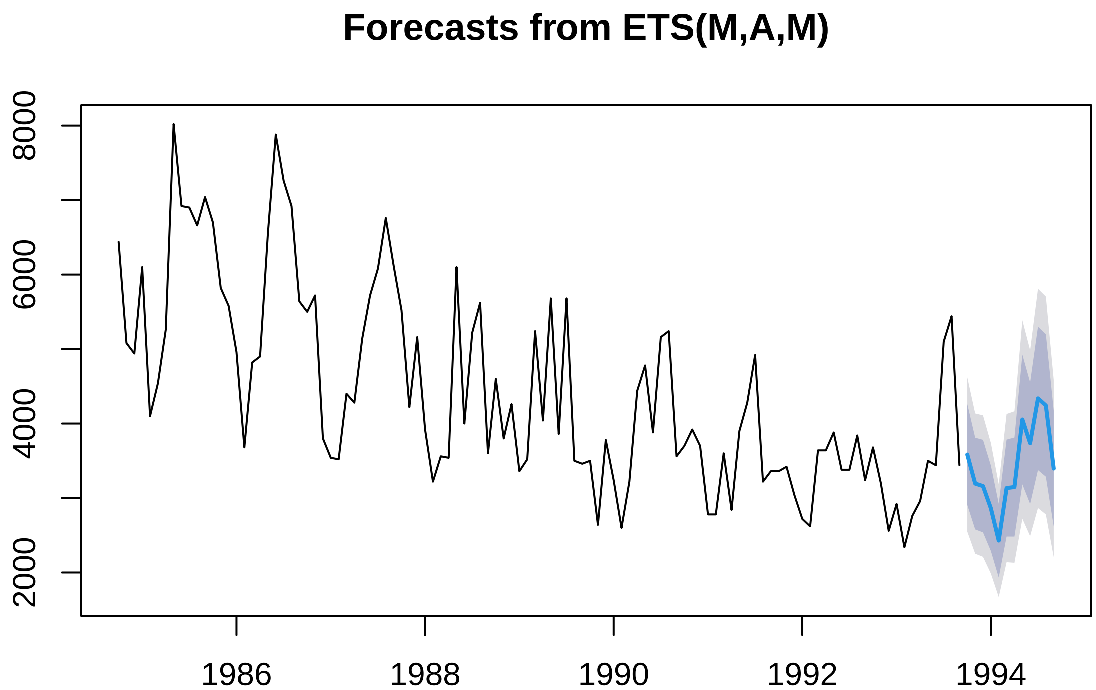
```
##
##  Box-Ljung test
##
## data:  residuos_holt_multi
## X-squared = 24.679, df = 15, p-value = 0.05443
```

```
#normalidade
shapiro.test(residuos_holt_multi)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuos_holt_multi
## W = 0.98801, p-value = 0.4727
```

```
#a série é normal, independente e estacionária.
```

```
#passo 4) Calcule previsões pontuais utilizando o modelo selecionado;
fit_holt_multi %>% forecast(h=12) %>% plot()
```




```
# passo 4) Obtenha previsões intervalares utilizando o modelo de espaço de estado equivalente
```