

Análise de sazonalidade de afluência em reservatórios de água

Cristiane Rodrigues Maragno, Jenifer Sofia Ovejero e Vinicius Guazzelli Dias

November 21, 2023

1 Introdução e Objetivos

No presente trabalho, exploraremos métodos estatísticos e ferramentas de visualização de dados para desvelar padrões sazonais, analisando como fatores como estações do ano, variações climáticas e eventos específicos podem impactar a afluência dos reservatórios. A compreensão desses padrões sazonais não apenas contribuirá para uma gestão mais eficaz dos recursos hídricos, mas também servirá como base sólida para a implementação de estratégias de adaptação às mudanças climáticas e eventos extremos, garantindo a sustentabilidade a longo prazo do abastecimento de água.

2 Fonte de Dados

A base de dados utilizada neste estudo provém das medições diárias de 713 reservatórios monitorados pela Agência Nacional de Águas e Saneamento Básico (ANA). Como entidade responsável pela implementação da Política Nacional de Recursos Hídricos no âmbito federal, a ANA desempenha um papel crucial na coleta, gestão e disponibilização de informações relacionadas aos recursos hídricos no território brasileiro.

2.1 Exploração da base e seleção de variáveis

A base de dados da ANA compreende medições diárias da afluência, cota, defluência e vazão de reservatórios de água, no período de 01/01/2000 até 01/01/2022. Essas medições são fundamentais para a compreensão das variações sazonais na entrada de água nos reservatórios, permitindo uma análise detalhada das flutuações ao longo do tempo.

Destaca-se, entre as variáveis da base, a afluência, pois essa variável desempenha um papel crucial ao possibilitar a correlação do comportamento do reservatório de água com variáveis climáticas correlatas, tais como temperatura e precipitação. Essa abordagem visa fornecer uma análise mais abrangente dos fatores que podem influenciar significativamente o nível e desempenho dos reservatórios.

Cabe ressaltar que, dos 713 reservatórios monitorados, optamos por selecionar dois exemplares para este estudo: o reservatório de Itaipu em Foz do Iguaçu, Paraná, e o Reservatório de Sobradinho na Bahia. Essa seleção estratégica visa representar diferentes

regiões geográficas, climas e contextos, proporcionando uma análise mais abrangente e representativa.



Figure 1: Reservatório de Itaipu - Paraná



Figure 2: Reservatório de Sobradinho - Bahia

2.2 Análise Exploratória

A análise aprofundada do gráfico revela claramente a existência de padrões sazonais nos dados de afluência de reservatórios de água. A presença de picos e vales ao longo do tempo indica uma recorrência consistente, sugerindo que certos comportamentos se repetem em intervalos específicos.

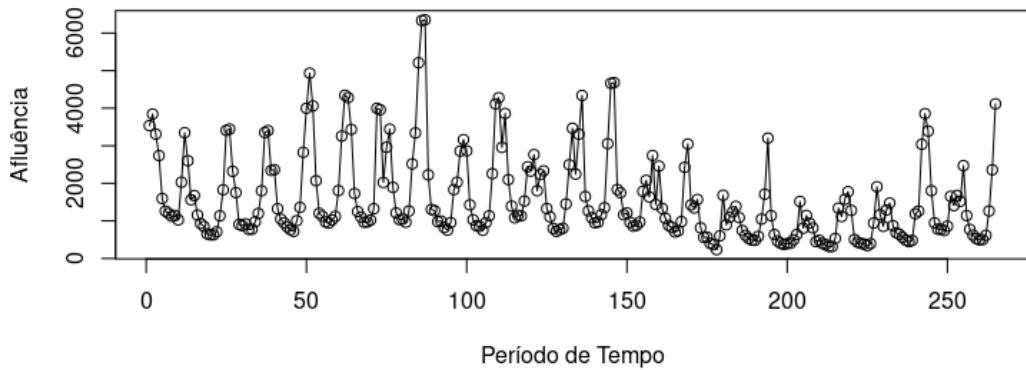


Figure 3: Plot dos dados do reservatório de Sobradinho

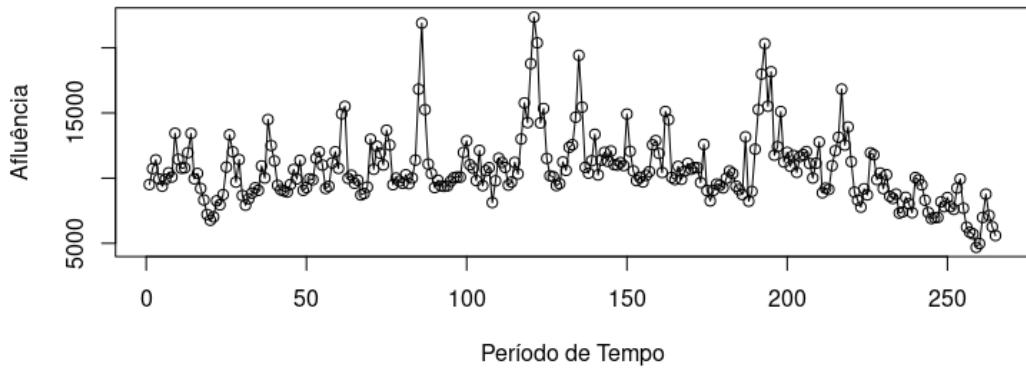


Figure 4: Plot dos dados do reservatório de Itaipu

Ao comparar os dois gráficos que representam reservatórios em regiões distintas do país, Sul e Nordeste, torna-se evidente que as disparidades observadas estão intimamente ligadas às características climáticas específicas de cada localidade. As variações sazonais nos dados de afluência revelam padrões distintos, refletindo as condições climáticas singulares de cada região. Enquanto a sazonalidade nos reservatórios do Sul pode estar associada a padrões climáticos típicos da região, como estações bem definidas, o gráfico do Nordeste pode mostrar variações mais acentuadas devido à presença de eventos climáticos extremos, como secas prolongadas ou chuvas intensas. Essas diferenças sublinham a importância de considerar os contextos regionais ao interpretar dados hidrológicos e ressaltam a necessidade de abordagens específicas para o gerenciamento sustentável dos recursos hídricos, adaptadas às particularidades climáticas de cada região.

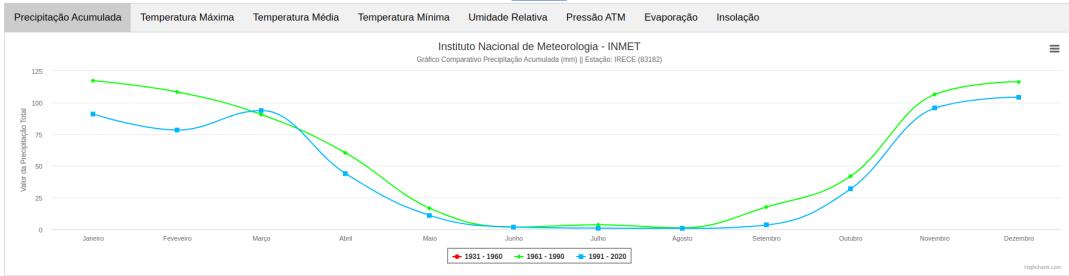


Figure 5: Precipitação aculuda no ano na estação de Irece, próximo ao reservatório de Sobradinho

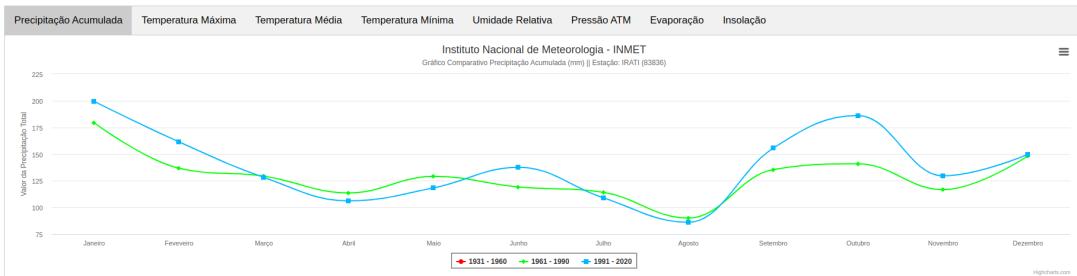


Figure 6: Precipitação aculuda no ano na estação de Irati, próximo ao reservatório de Itaipu

2.3 Sobradinho

A etapa subsequente envolve a exploração meticulosa de diversos modelos, visando identificar aquele que demonstrará maior eficácia na previsão dos dados futuros. Este processo de experimentação e comparação entre modelos é essencial para aprimorar a precisão das projeções, permitindo-nos escolher a abordagem mais robusta e adaptada às características específicas do conjunto de dados em questão.

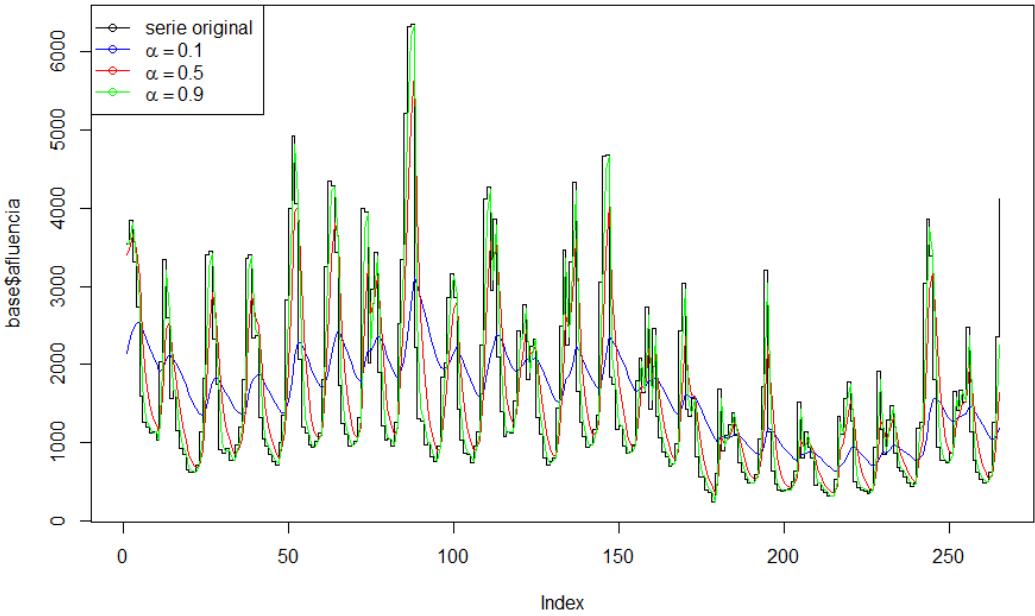


Figure 7: Sobradinho Simples

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
alpha1	Modelo 1	-25.137270	1079.0969	853.1927	-40.876033	67.82331	1.645554	0.6905917
alpha2	Modelo 2	-3.966157	946.3784	704.8701	-20.517244	47.96247	1.359484	0.5448314
alpha3	Modelo 3	1.537144	808.8068	541.9291	-9.474562	33.43731	1.045220	0.2666656

Figure 8: Tabela de erro de Sobradinho para o modelo Simples

Ao analisar as métricas de erro, como o Erro Médio Quadrático (RMSE) e o Erro Absoluto Médio (MAE), torna-se possível identificar de maneira precisa o desempenho dos diferentes modelos em consideração. O RMSE proporciona uma medida da dispersão dos erros, penalizando desvios mais significativos, enquanto o MAE fornece uma visão direta da magnitude média dos erros. Um RMSE e MAE menores indicam uma correspondência mais próxima entre as previsões do modelo e os valores reais, sinalizando a escolha do modelo mais eficaz. Ao comparar essas métricas entre os modelos avaliados, conseguimos discernir que o alpha3 apresenta um ajuste mais preciso aos dados observados.

Este mesmo processo será executado minuciosamente ao avaliar o modelo de previsão Holt, com sua capacidade de lidar com tendências lineares, e o modelo de Holt-Winter, que incorpora componentes sazonais, para identificar qual deles oferece a melhor adaptação aos padrões presentes nos dados.

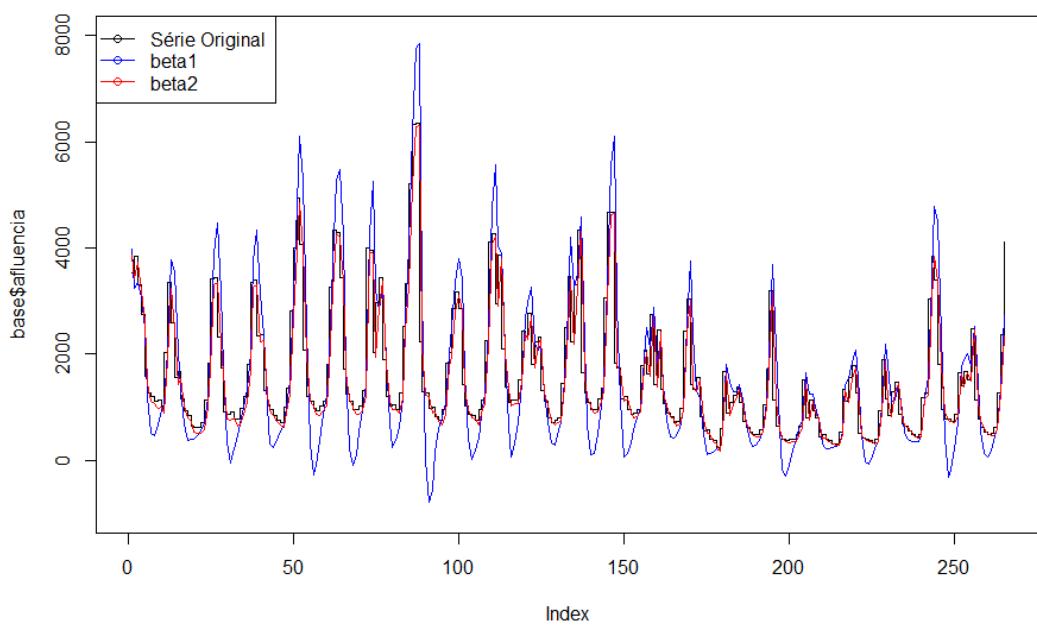


Figure 9: Sobradinho holt

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
beta1	Modelo 1	15.27376	1025.4066	734.8264	8.257265	54.43748	1.417261	0.3940861
beta2	Modelo 2	63.86008	796.1264	522.2182	-2.328062	31.18327	1.007203	0.1831915

Figure 10: Tabela Sobradinho holt

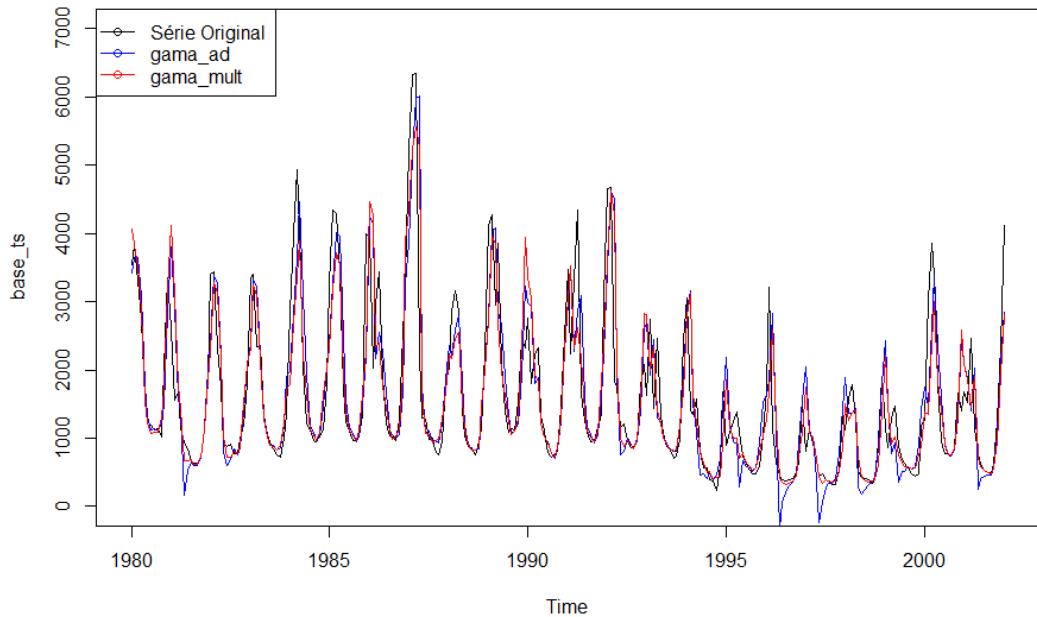


Figure 11: Sobradinho holt winter

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
gama_ad	Modelo 1	-4.889953	634.6691	402.6758	-3.731965	26.74572	0.6400908	0.04473488
gama_mult	Modelo 2	14.671536	601.7826	364.7703	-4.881387	21.13726	0.5798364	0.07674112

Figure 12: Tabela Sobradinho holt winter

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
alpha_otimo	Modelo 1	6.912394	794.7543	521.2758	-7.912437	31.65711	1.0053855	0.18560425
beta2	Modelo 2	63.860082	796.1264	522.2182	-2.328062	31.18327	1.0072030	0.18319150
gama_mult	Modelo 3	14.671536	601.7826	364.7703	-4.881387	21.13726	0.5798364	0.07674112

Figure 13: Tabela Sobradinho análise geral

Ao empregar as mesmas métricas de erro, como o Erro Médio Quadrático (RMSE) e o Erro Absoluto Médio (MAE), para avaliar os modelos de previsão anteriormente mencionados, buscamos identificar de forma consistente o método que apresenta o desempenho mais eficaz. Este procedimento sistemático permite-nos discernir qual abordagem de modelagem melhor se ajusta às características específicas do conjunto de dados, oferecendo uma previsão mais precisa e confiável. Nesse caso, notamos que a abordagem mais eficaz para a previsão dos dados é a utilização de uma série temporal com modelo multiplicativo.

2.4 Itaipu

Este mesmo procedimento será aplicado com igual rigor ao reservatório de água de Itaipu, reconhecendo a necessidade de desenvolver um novo modelo de previsão devido à natureza única dos dados associados a este reservatório específico.

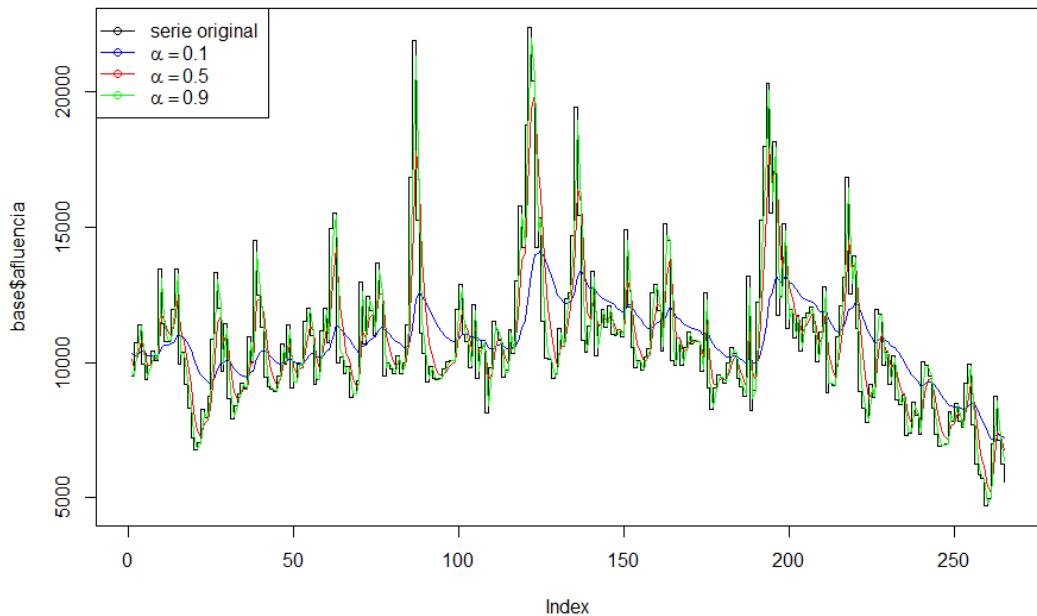


Figure 14: Itaipu Simples

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
alpha1	Modelo 1	-123.96137	2300.103	1687.219	-5.014944	15.55721	1.2054268	0.60684067
alpha2	Modelo 2	-29.51784	2005.584	1456.348	-2.462886	13.17349	1.0404813	0.31014008
alpha3	Modelo 3	-16.64261	1927.152	1376.412	-1.625389	12.28469	0.9833717	-0.03568906

Figure 15: Tabela Itaipu Simples

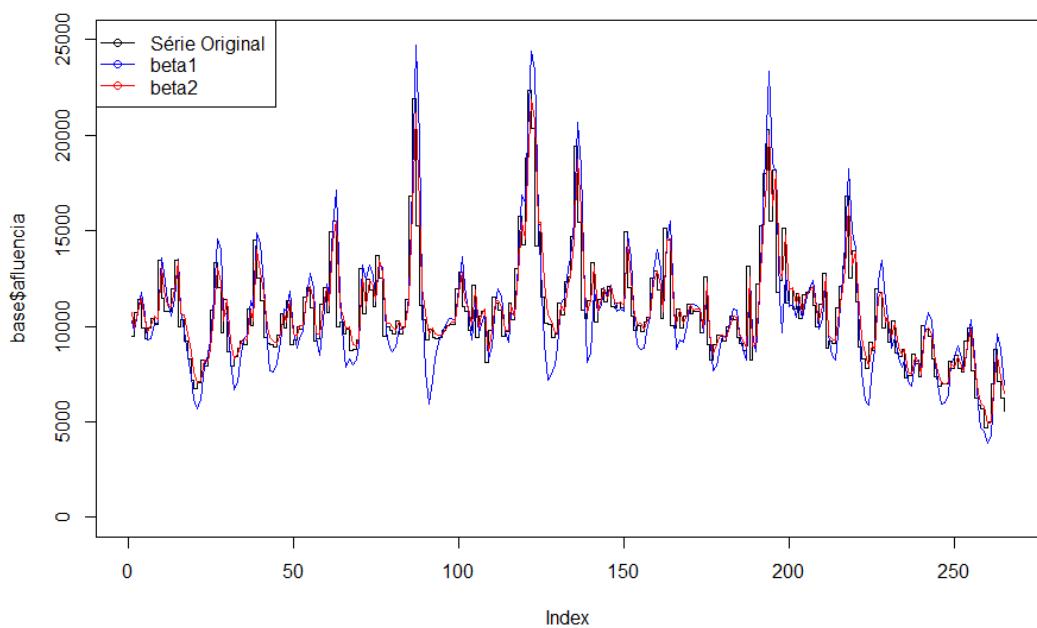


Figure 16: Itaipu holt

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
beta1	Modelo 1	-10.97036	2336.129	1701.424	-0.7181872	15.66191	1.2155750	0.15327400
beta2	Modelo 2	-147.23253	1934.701	1393.986	-2.9463939	12.56387	0.9959275	0.01476812

Figure 17: Tabela Itaipu holt

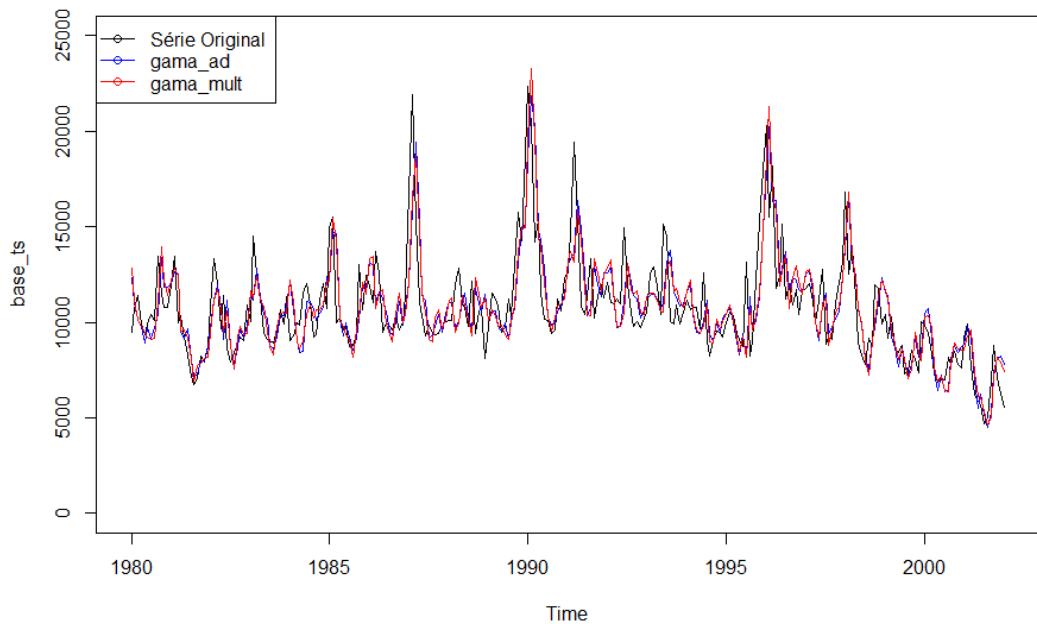


Figure 18: Itaipu holt winter

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
gama_ad	Modelo 1	-47.7965	1747.998	1268.969	-1.875266	11.41591	0.5795156	0.04885277
gama_mult	Modelo 2	-57.3732	1748.803	1298.150	-1.921301	11.71245	0.5928421	0.04811449

Figure 19: Tabela Itaipu holt winter

	Modelo	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
alpha_otimo	Modelo 1	-18.0542	1924.024	1375.915	-1.723872	12.30088	0.9830167	0.02194365
beta2	Modelo 2	-147.2325	1934.701	1393.986	-2.946394	12.56387	0.9959275	0.01476812
gama_ad	Modelo 3	-47.7965	1747.998	1268.969	-1.875266	11.41591	0.5795156	0.04885277

Figure 20: Tabela Itaipu análise geral

Nesse segundo caso, torna-se evidente que a estratégia mais eficiente para a previsão dos dados reside na aplicação de uma série temporal com modelo aditivo.

3 Resultados e Discussões

Ao examinar os gráficos de Sazonalidade Exponencial Simples, Sazonalidade Exponencial Holt e Sazonalidade Exponencial Holt-Winters, juntamente com suas tabelas correspondentes de cálculo de erro de ajuste, para os reservatórios de Sobradinho e Itaipu, é evidente que, em ambos os casos, as opções "alpha3" (Sazonalidade Exponencial Simples)

e "beta2" (Sazonalidade Exponencial Holt) se destacam. No entanto, é importante notar uma distinção no modelo Sazonalidade Exponencial Holt-Winters, onde a abordagem multiplicativa foi mais eficaz no primeiro caso, enquanto a abordagem aditiva mostrou-se superior no segundo. Essa análise aprofundada das diferentes configurações revela nuances cruciais no desempenho dos modelos em questão, proporcionando insights valiosos para aprimorar a precisão das previsões nos reservatórios estudados, Sobradinho e Itaipu.

4 Considerações finais

Fornecendo uma análise mais refinada das tendências de afluência, os modelos que adotam a abordagem Holt-Winters com sazonalidade multiplicativa para Sobradinho e aditiva para Itaipu oferecem uma resposta proativa diante das flutuações climáticas. Ao antecipar as variações na disponibilidade de água nos respectivos reservatórios, torna-se possível implementar estratégias de gerenciamento mais eficazes, otimizando assim o uso dos recursos hídricos. Esta habilidade preditiva revela-se de importância crucial diante dos desafios climáticos emergentes, incluindo eventos extremos e mudanças climáticas. Portanto, a seleção criteriosa de modelos de previsão não apenas facilita uma gestão sustentável, mas também fortalece a resiliência dos sistemas hídricos, assegurando o abastecimento de água a longo prazo e mitigando impactos adversos.

Referências

- [1] Conjunto de dados "Reservatórios brasileiros" no basedosdados: <https://basedosdados.org/dataset/fc7e9d13-714d-42c1-8986-bd2a3108e208?table=31415e13-ba3d-4fe0-ba27-0242ee076f7f>
- [2] Site de gráficos em relação ao clima: <https://clima.inmet.gov.br/>

5 Anexo: Sintaxe R Utilizada

```
library(tidyverse)
library(forecast)
library(lubridate)
library(data.table)

# ITAIPU

base <- fread(input = paste0("itaipu_mensal_oficial.csv"), header = T,
  na.strings = "NA", data.table = FALSE, dec=",")

base$tempo <- seq(1:nrow(base))
base$intervalo <- as.factor(base$mes)
base$afluencia <- as.numeric(base$afluencia)
```

```

# Visualizao grfica
plot(base$tempo,base$afluencia,xlab="Perodo de Tempo", ylab="Afluncia")
lines(base$tempo,base$afluencia, col = "black")

# SUAVIZACAO EXPONENCIAL SIMPLES (SES)
#####
alpha1 <- ses(base$afluencia, alpha = 0.1)
alpha2 <- ses(base$afluencia, alpha = 0.5)
alpha3 <- ses(base$afluencia, alpha = 0.9)

#calculo dos erros de cada ajuste
itaipu_resultados_SES <- list(alpha1, alpha2, alpha3) %>% map(accuracy)

itaipu_tabela_SES <- do.call(rbind, lapply(seq_along(itaipu_resultados_SES),
  function(i) {
  data.frame(Modelo = paste("Modelo", i), itaipu_resultados_SES[[i]])
})) 
rownames(itaipu_tabela_SES) <- c("alpha1", "alpha2", "alpha3")

# grfico do ajuste
plot(base$afluencia,type="s")
lines(fitted(alpha1), col="blue")
lines(fitted(alpha2), col="red")
lines(fitted(alpha3), col="green")
legend("topleft",lty=1, col=c(1,"blue","red","green"),
  c("serie original",
    expression(alpha == 0.1),
    expression(alpha == 0.5),
    expression(alpha == 0.9)),
  pch=1)

alpha_otimo <- ses(base$afluencia)
summary(alpha_otimo)
# alpha = 0.8332

# SUAVIZACAO EXPONENCIAL DE HOLT
#####
beta1 <- holt(base$afluencia, alpha = 0.6, beta = 0.4)
summary(beta1)
beta2 <- holt(base$afluencia)
summary(beta2)
# beta1 = 0.4
# beta = 0.0043

#calculo dos erros de cada ajuste
itaipu_resultados_holt <- list(beta1, beta2) %>% map(accuracy)

itaipu_tabela_holt <- do.call(rbind,
  lapply(seq_along(itaipu_resultados_holt), function(i) {
  data.frame(Modelo = paste("Modelo", i), itaipu_resultados_holt[[i]])
}))

```

```

rownames(itaipu_tabela_holt) <- c("beta1", "beta2")

# gráfico do ajuste
plot(base$afluencia, type="s", ylim = c(0, 25000))
lines(fitted(beta1), col = "blue")
lines(fitted(beta2), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Srie Original", "beta1", "beta2"),
       pch = 1)

# SUAVIZACAO EXPONENCIAL DE HOLT-WINTER
#####
base_ts <- ts(base$afluencia, frequency=12, start=c(1980,1))
base_ts

gama_ad <- hw(base_ts, seasonal = "additive")
gama_mult <- hw(base_ts, seasonal = "multiplicative")

# calculando o erro de cada ajuste
itaipu_tabela_holtwinter <- list(gama_ad, gama_mult) %>% map(accuracy)

itaipu_tabela_holtwinter <- do.call(rbind,
  lapply(seq_along(itaipu_tabela_holtwinter), function(i) {
    data.frame(Modelo = paste("Modelo", i), itaipu_tabela_holtwinter[[i]])))
rownames(itaipu_tabela_holtwinter) <- c("gama_ad", "gama_mult")

# analise grafica do ajuste
plot(base_ts, , ylim = c(0, 25000))
lines(fitted(gama_ad), col = "blue")
lines(fitted(gama_mult), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Srie Original", "gama_ad", "gama_mult"),
       pch = 1)

# COMPARACAO GERAL ITAIPU
#####

itaipu_tabela_geral <- list(alpha_otimo, beta2, gama_ad) %>% map(accuracy)

itaipu_tabela_geral <- do.call(rbind, lapply(seq_along(itaipu_tabela_geral),
  function(i) {
    data.frame(Modelo = paste("Modelo", i), itaipu_tabela_geral[[i]])))
rownames(itaipu_tabela_geral) <- c("alpha_otimo", "beta2", "gama_ad")

# SOBRADINHO

base <- fread(input = paste0("sobradinho_mensal_oficial.csv"), header = T,
  na.strings = "NA", data.table = FALSE, dec=",")

```

```

base$tempo <- seq(1:nrow(base))
base$intervalo <- as.factor(base$mes)
base$afluencia <- as.numeric(base$afluencia)

# Visualizao grfica
plot(base$tempo,base$afluencia,xlab="Perodo de Tempo", ylab="Afluncia")
lines(base$tempo,base$afluencia, col = "black")

# SUAVIZACAO EXPONENCIAL SIMPLES (SES)
#####
alpha1 <- ses(base$afluencia, alpha = 0.1)
alpha2 <- ses(base$afluencia, alpha = 0.5)
alpha3 <- ses(base$afluencia, alpha = 0.9)

#calculo dos erros de cada ajuste
sobradinho_resultados_SES <- list(alpha1, alpha2, alpha3) %>% map(accuracy)

sobradinho_tabela_SES <- do.call(rbind,
  lapply(seq_along(sobradinho_resultados_SES), function(i) {
    data.frame(Modelo = paste("Modelo", i), sobradinho_resultados_SES[[i]])
}))
rownames(sobradinho_tabela_SES) <- c("alpha1", "alpha2", "alpha3")

# grfico do ajuste
plot(base$afluencia,type="s")
lines(fitted(alpha1), col="blue")
lines(fitted(alpha2), col="red")
lines(fitted(alpha3), col="green")
legend("topleft",lty=1, col=c(1,"blue","red","green"),
      c("serie original",
        expression(alpha == 0.1),
        expression(alpha == 0.5),
        expression(alpha == 0.9)),
      pch=1)

alpha_otimo <- ses(base$afluencia)
summary(alpha_otimo)
# alpha = 0.9999

# SUAVIZACAO EXPONENCIAL DE HOLT
#####
beta1 <- holt(base$afluencia, alpha = 0.6, beta = 0.4)
summary(beta1)
beta2 <- holt(base$afluencia)
summary(beta2)
# beta1 = 0.4
# beta = 0.007

#calculo dos erros de cada ajuste
sobradinho_resultados_holt <- list(beta1, beta2) %>% map(accuracy)

```

```

sobradinho_tabela_holt <- do.call(rbind,
  lapply(seq_along(sobradinho_resultados_holt), function(i) {
    data.frame(Modelo = paste("Modelo", i), sobradinho_resultados_holt[[i]]))
}))
rownames(sobradinho_tabela_holt) <- c("beta1", "beta2")

# gráfico do ajuste
plot(base$afluencia, type="s", ylim = c(-1000, 8000))
lines(fitted(beta1), col = "blue")
lines(fitted(beta2), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Série Original", "beta1", "beta2"),
       pch = 1)

# SUAVIZAÇÃO EXPONENCIAL DE HOLT-WINTER
#####
base_ts <- ts(base$afluencia, frequency=12, start=c(1980,1))
base_ts

gama_ad <- hw(base_ts, seasonal = "additive")
gama_mult <- hw(base_ts, seasonal = "multiplicative")

# calculando o erro de cada ajuste
sobradinho_tabela_holtwinter <- list(gama_ad, gama_mult) %>% map(accuracy)

sobradinho_tabela_holtwinter <- do.call(rbind,
  lapply(seq_along(sobradinho_tabela_holtwinter), function(i) {
    data.frame(Modelo = paste("Modelo", i), sobradinho_tabela_holtwinter[[i]]))
}))
rownames(sobradinho_tabela_holtwinter) <- c("gama_ad", "gama_mult")

# análise gráfica do ajuste
plot(base_ts, , ylim = c(0, 7000))
lines(fitted(gama_ad), col = "blue")
lines(fitted(gama_mult), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Série Original", "gama_ad", "gama_mult"),
       pch = 1)

# COMPARAÇÃO GERAL SOBRADINHO
#####

sobradinho_tabela_geral <- list(alpha_otimo, beta2, gama_ad) %>% map(accuracy)

sobradinho_tabela_geral <- do.call(rbind,
  lapply(seq_along(sobradinho_tabela_geral), function(i) {
    data.frame(Modelo = paste("Modelo", i), sobradinho_tabela_geral[[i]]))
}))
rownames(sobradinho_tabela_geral) <- c("alpha_otimo", "beta2", "gama_ad")

```

analise.R

```
library(tidyverse)

## — Attaching core tidyverse packages ————— tidyverse
2.0.0 —
## ✓ dplyr     1.1.3     ✓ readr      2.1.4
## ✓forcats    1.0.0     ✓ stringr    1.5.0
## ✓ ggplot2    3.4.1     ✓ tibble     3.2.1
## ✓ lubridate  1.9.3     ✓ tidyverse   1.3.0
## ✓ purrr     1.0.1
## — Conflicts ——————
tidyverse_conflicts() —
## X dplyr::filter() masks stats::filter()
## X dplyr::lag()   masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force
all conflicts to become errors

library(forecast)

## Registered S3 method overwritten by 'quantmod':
##   method           from
##   as.zoo.data.frame zoo

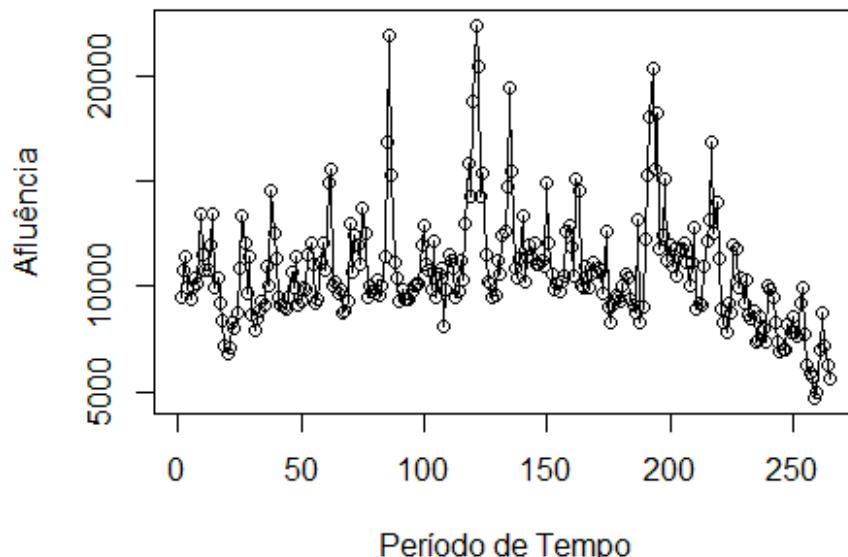
library(lubridate)
library(data.table)

##
## Attaching package: 'data.table'
##
## The following objects are masked from 'package:lubridate':
##
##   hour, isoweek, mday, minute, month, quarter, second, wday, week,
##   yday, year
##
## The following objects are masked from 'package:dplyr':
##
##   between, first, last
##
## The following object is masked from 'package:purrr':
##
##   transpose

# ITAIPU

base <- fread(input = paste0("itaipu_mensal_oficial.csv"), header = T,
na.strings = "NA", data.table = FALSE, dec=",")  
base$tempo <- seq(1:nrow(base))
base$intervalo <- as.factor(base$mes)
base$afluencia <- as.numeric(base$afluencia)
```

```
# Visualização gráfica
plot(base$tempo,base$afluencia,xlab="Período de Tempo", ylab="Afluência")
lines(base$tempo,base$afluencia, col = "black")
```

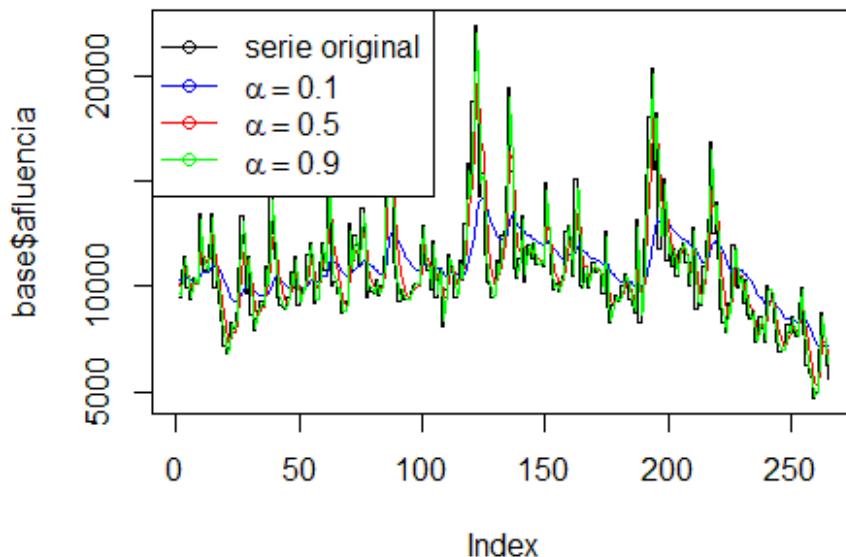


```
# SUAVIZACAO EXPONENCIAL SIMPLES (SES)
#####
alpha1 <- ses(base$afluencia, alpha = 0.1)
alpha2 <- ses(base$afluencia, alpha = 0.5)
alpha3 <- ses(base$afluencia, alpha = 0.9)

#calculo dos erros de cada ajuste
itaipu_resultados_SES <- list(alpha1, alpha2, alpha3) %>% map(accuracy)

itaipu_tabela_SES <- do.call(rbind,
lapply(seq_along(itaipu_resultados_SES), function(i) {
  data.frame(Modelo = paste("Modelo", i), itaipu_resultados_SES[[i]])))
rownames(itaipu_tabela_SES) <- c("alpha1", "alpha2", "alpha3")

#gráfico do ajuste
plot(base$afluencia,type="s")
lines(fitted(alpha1), col="blue")
lines(fitted(alpha2), col="red")
lines(fitted(alpha3), col="green")
legend("topleft",lty=1, col=c(1,"blue","red","green"),
      c("serie original",
        expression(alpha == 0.1),
        expression(alpha == 0.5),
        expression(alpha == 0.9)),
      pch=1)
```



```

alpha_otimo <- ses(base$afluencia)
summary(alpha_otimo)

##
## Forecast method: Simple exponential smoothing
##
## Model Information:
## Simple exponential smoothing
##
## Call:
##   ses(y = base$afluencia)
##
## Smoothing parameters:
##   alpha = 0.8332
##
## Initial states:
##   l = 9718.8769
##
## sigma: 1931.325
##
##      AIC     AICc      BIC
## 5492.581 5492.673 5503.320
##
## Error measures:
##               ME      RMSE      MAE      MPE      MAPE      MASE
ACF1
## Training set -18.0542 1924.024 1375.915 -1.723872 12.30088 0.9830167
0.02194365
##
## Forecasts:
##       Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## 266      5732.532 3257.4385 8207.625 1947.2034 9517.86

```

```

## 267      5732.532 2510.8891  8954.174   805.4546 10659.61
## 268      5732.532 1907.3682  9557.695  -117.5510 11582.61
## 269      5732.532 1386.8706 10078.193  -913.5834 12378.65
## 270      5732.532  922.3690 10542.694  -1623.9772 13089.04
## 271      5732.532  498.9326 10966.131  -2271.5673 13736.63
## 272      5732.532  107.2802 11357.783  -2870.5480 14335.61
## 273      5732.532 -258.8245 11723.888  -3430.4568 14895.52
## 274      5732.532 -603.8114 12068.875  -3958.0687 15423.13
## 275      5732.532 -930.9612 12396.024  -4458.4012 15923.46

# alpha = 0.8332

# SUAVIZACAO EXPONENCIAL DE HOLT
#####
beta1 <- holt(base$afluencia, alpha = 0.6, beta = 0.4)
summary(beta1)

##
## Forecast method: Holt's method
##
## Model Information:
## Holt's method
##
## Call:
## holt(y = base$afluencia, alpha = 0.6, beta = 0.4)
##
## Smoothing parameters:
## alpha = 0.6
## beta  = 0.4
##
## Initial states:
## l = 9989.6635
## b = 286.0696
##
## sigma: 2353.962
##
##      AIC     AICc      BIC
## 5595.441 5595.533 5606.180
##
## Error measures:
##               ME      RMSE      MAE      MPE      MAPE      MASE
ACF1
## Training set -10.97036 2336.129 1701.424 -0.7181872 15.66191 1.215575
0.153274
##
## Forecasts:
##      Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## 266    5246.76680    2230.0432    8263.490    633.0861   9860.447
## 267    4369.97777    103.6863    8636.269   -2154.7520  10894.708
## 268    3493.18875   -2510.0155    9496.393   -5687.9199  12674.297
## 269    2616.39972   -5478.3192  10711.119   -9763.4046  14996.204
## 270    1739.61069   -8728.0291  12207.250  -14269.2629  17748.484
## 271    862.82166   -12217.3804 13943.024  -19141.6210  20867.264
## 272   -13.96737  -15919.8555 15891.921  -24339.9241  24311.989

```

```

## 273      -890.75640 -19816.9410 18035.428 -29835.8579 28054.345
## 274      -1767.54543 -23894.7558 20359.665 -35608.1934 32073.103
## 275      -2644.33445 -28142.3423 22853.673 -41640.1722 36351.503

beta2 <- holt(base$afluencia)
summary(beta2)

##
## Forecast method: Holt's method
##
## Model Information:
## Holt's method
##
## Call:
##   holt(y = base$afluencia)
##
## Smoothing parameters:
##   alpha = 0.8407
##   beta  = 0.0043
##
## Initial states:
##   l = 10515.9894
##   b = 193.3977
##
## sigma: 1949.47
##
##          AIC      AICc      BIC
## 5499.514 5499.745 5517.412
##
## Error measures:
##               ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -147.2325 1934.701 1393.986 -2.946394 12.56387 0.9959275
##                  ACF1
## Training set 0.01476812
##
## Forecasts:
##    Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## 266 5754.676 3256.32997 8253.023 1933.7852 9575.568
## 267 5779.206 2508.25650 9050.155 776.7206 10781.691
## 268 5803.735 1904.72766 9702.742 -159.2820 11766.752
## 269 5828.264 1384.00763 10272.520 -968.6395 12625.167
## 270 5852.793 918.54393 10787.042 -1693.4898 13399.076
## 271 5877.322 493.20743 11261.437 -2356.9708 14111.615
## 272 5901.851 98.63843 11705.064 -2973.3969 14777.100
## 273 5926.380 -271.42662 12124.188 -3552.3475 15405.108
## 274 5950.910 -621.42057 12523.240 -4100.6020 16002.421
## 275 5975.439 -954.61519 12905.493 -4623.1642 16574.042

# beta1 = 0.4
# beta  = 0.0043

#calculo dos erros de cada ajuste
itaipu_resultados_holt <- list(beta1, beta2) %>% map(accuracy)

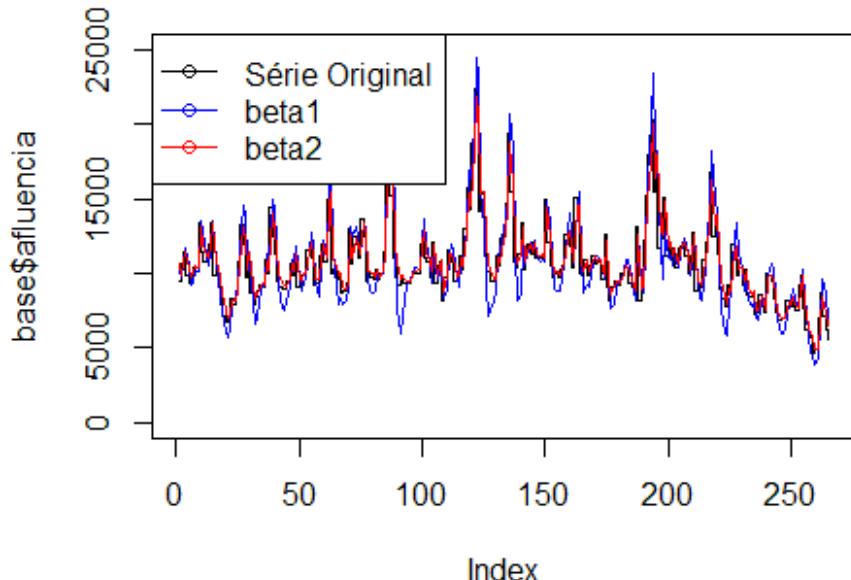
```

```

itaipu_tabela_holt <- do.call(rbind,
lapply(seq_along(itaipu_resultados_holt), function(i) {
  data.frame(Modelo = paste("Modelo", i), itaipu_resultados_holt[[i]])))
rownames(itaipu_tabela_holt) <- c("beta1", "beta2")

#gráfico do ajuste
plot(base$afluencia, type="s", ylim = c(0, 25000))
lines(fitted(beta1), col = "blue")
lines(fitted(beta2), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Série Original", "beta1", "beta2"),
       pch = 1)

```



```

# SUAVIZACAO EXPONENCIAL DE HOLT-WINTER
#####
base_ts <- ts(base$afluencia, frequency=12, start=c(1980,1))
base_ts

##                Jan      Feb      Mar      Apr      May      Jun
Jul
## 1980  9505.871 10722.276 11403.097  9936.600  9379.290  9908.633
10409.613
## 1981 11947.387 13432.857  9947.452 10361.333  9204.355  8324.033
7207.452
## 1982 10846.419 13322.393 12020.290  9694.233 11422.516  8662.400
7923.774
## 1983 10008.806 14495.357 12500.871 11315.867  9444.548  9103.033
9020.387
## 1984  9065.677  9274.345  9947.613  9876.200 11526.290 12018.933
10997.387
## 1985 14932.387 15507.607  9985.710 10223.067  9604.806  9844.100

```

```

8709.097
## 1986 11914.839 11008.071 13678.548 12535.967 9496.387 10003.200
9769.065
## 1987 16825.834 21894.548 15253.702 11077.402 10352.178 9275.790
9841.009
## 1988 10082.396 10076.140 11967.382 12864.504 11022.488 10759.533
9814.690
## 1989 9791.832 11531.235 11202.948 10821.955 9461.495 9699.214
11210.389
## 1990 22356.948 20382.784 14233.089 15341.942 11509.636 10176.839
10105.635
## 1991 12560.671 14675.388 19424.982 15448.052 10827.973 10382.394
11335.058
## 1992 12081.456 11033.137 10994.075 11200.670 10961.088 14918.787
12056.935
## 1993 10497.979 12554.527 12882.475 11895.324 10430.807 15115.308
14486.794
## 1994 11140.852 10674.360 10810.633 10791.888 9657.476 12587.324
9046.249
## 1995 9984.706 10545.801 10351.057 9397.109 9117.649 8740.439
13171.730
## 1996 20319.835 15516.441 18162.568 11759.566 12420.527 15110.524
11236.243
## 1997 11806.452 12037.111 11120.533 10004.505 11141.226 12777.621
8875.401
## 1998 16830.785 12533.727 13928.755 11262.582 8927.947 8311.235
7773.581
## 1999 10391.561 9186.739 10263.199 8614.319 8440.668 8775.478
7320.028
## 2000 9875.186 9479.321 8302.685 7365.502 6899.244 6967.628
6986.034
## 2001 9232.826 9935.652 7688.881 6238.021 5845.759 5713.713
4688.531
## 2002 5589.512
##          Aug      Sep      Oct      Nov      Dec
## 1980 10086.323 13444.000 11447.613 10755.367 10787.742
## 1981 6767.710 7027.967 8255.774 7950.667 8732.613
## 1982 8419.032 8851.533 9256.516 9064.000 10934.968
## 1983 8931.710 9510.400 10665.935 9938.900 11369.903
## 1984 9205.290 9374.933 11163.871 12022.833 10745.839
## 1985 8821.645 9315.533 12991.161 10678.667 12446.903
## 1986 9603.484 10258.479 9575.806 10003.431 11384.177
## 1987 9401.214 9367.673 9401.236 9764.156 10027.740
## 1988 12128.665 9421.470 10566.604 10802.473 8125.121
## 1989 10320.347 13000.994 15771.712 14260.631 18783.704
## 1990 9427.588 9578.053 11239.367 10595.046 12367.187
## 1991 13363.768 10242.812 11389.413 11977.760 11290.709
## 1992 10550.506 9809.226 10064.204 9703.310 10212.908
## 1993 10055.182 9888.325 10904.961 9909.742 10534.878
## 1994 8252.758 9064.045 9541.569 9448.044 9248.261
## 1995 8225.973 8980.790 12217.819 15262.546 17985.169
## 1996 11977.205 10901.427 11739.602 10439.929 11647.377
## 1997 9227.611 9132.782 10953.261 12097.701 13138.624
## 1998 9183.271 8708.860 11939.883 11791.102 9884.999

```

```

## 1999 7408.461 8529.145 8033.097 7346.908 10047.218
## 2000 8184.143 7808.492 8509.971 7813.352 7599.960
## 2001 4973.714 6982.994 8773.112 7132.700 6266.761
## 2002

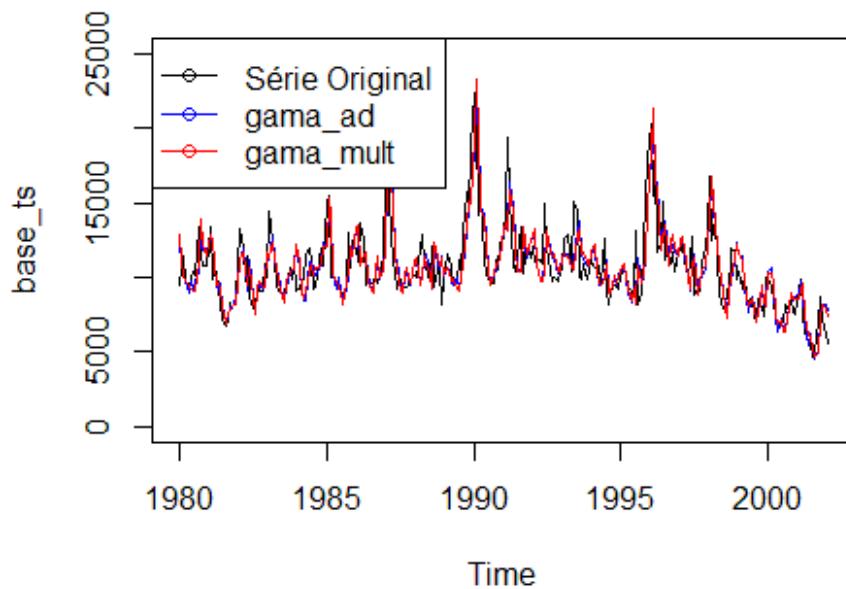
gama_ad <- hw(base_ts, seasonal = "additive")
gama_mult <- hw(base_ts, seasonal = "multiplicative")

# calculando o erro de cada ajuste
itaipu_tabela_holtwinter <- list(gama_ad, gama_mult) %>% map(accuracy)

itaipu_tabela_holtwinter <- do.call(rbind,
lapply(seq_along(itaipu_tabela_holtwinter), function(i) {
  data.frame(Modelo = paste("Modelo", i), itaipu_tabela_holtwinter[[i]])
}))
rownames(itaipu_tabela_holtwinter) <- c("gama_ad", "gama_mult")

# analise grafica do ajuste
plot(base_ts, , ylim = c(0, 25000))
lines(fitted(gama_ad), col = "blue")
lines(fitted(gama_mult), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Série Original", "gama_ad", "gama_mult"),
       pch = 1)

```



```

# COMPARACAO GERAL ITAIPU
#####
itaipu_tabela_geral <- list(alpha_otimo, beta2, gama_ad) %>% map(accuracy)

itaipu_tabela_geral <- do.call(rbind,
lapply(seq_along(itaipu_tabela_geral), function(i) {

```

```

data.frame(Modelo = paste("Modelo", i), itaipu_tabela_geral[[i]])
})
rownames(itaipu_tabela_geral) <- c("alpha_otimo", "beta2", "gama_ad")

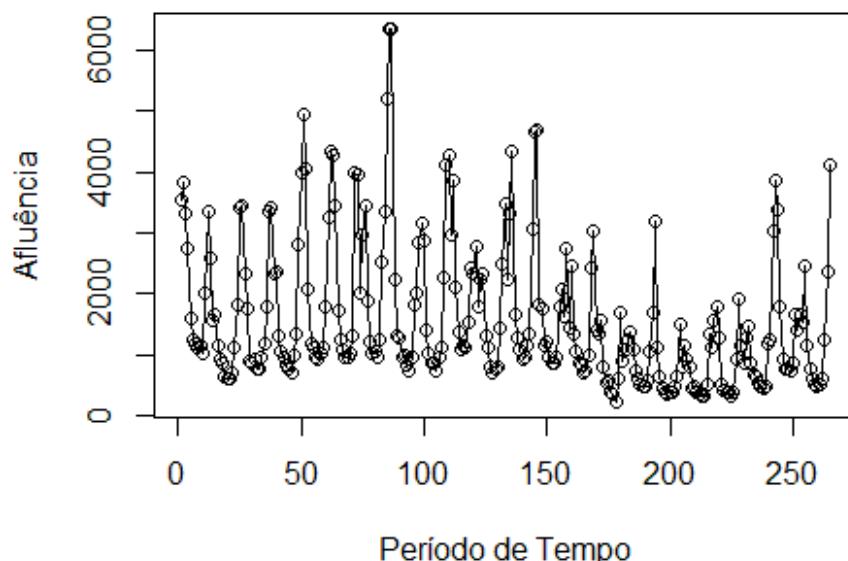
# SOBRADINHO

base <- fread(input = paste0("sobradinho_mensal_oficial.csv"), header = T,
na.strings = "NA", data.table = FALSE, dec=",")

base$tempo <- seq(1:nrow(base))
base$intervalo <- as.factor(base$mes)
base$afluencia <- as.numeric(base$afluencia)

# Visualização gráfica
plot(base$tempo, base$afluencia, xlab="Período de Tempo", ylab="Afluência")
lines(base$tempo, base$afluencia, col = "black")

```



```

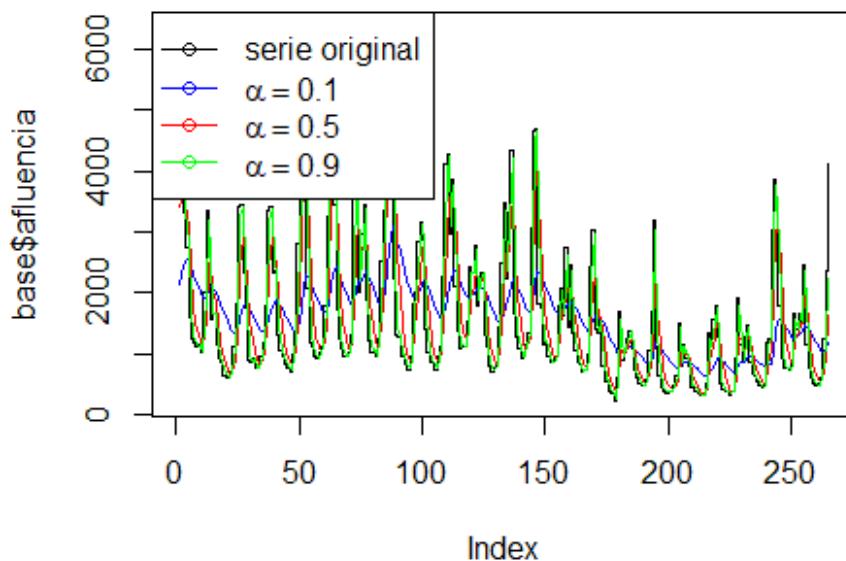
# SUAVIZACAO EXPONENCIAL SIMPLES (SES)
#####
alpha1 <- ses(base$afluencia, alpha = 0.1)
alpha2 <- ses(base$afluencia, alpha = 0.5)
alpha3 <- ses(base$afluencia, alpha = 0.9)

#calculo dos erros de cada ajuste
sobradinho_resultados_SES <- list(alpha1, alpha2, alpha3) %>%
map(accuracy)

sobradinho_tabela_SES <- do.call(rbind,
lapply(seq_along(sobradinho_resultados_SES), function(i) {
  data.frame(Modelo = paste("Modelo", i), sobradinho_resultados_SES[[i]])))
)
rownames(sobradinho_tabela_SES) <- c("alpha1", "alpha2", "alpha3")

```

```
#gráfico do ajuste
plot(base$afluencia, type="s")
lines(fitted(alpha1), col="blue")
lines(fitted(alpha2), col="red")
lines(fitted(alpha3), col="green")
legend("topleft", lty=1, col=c(1,"blue","red","green"),
       c("serie original",
         expression(alpha == 0.1),
         expression(alpha == 0.5),
         expression(alpha == 0.9))),
       pch=1)
```



```
alpha_otimo <- ses(base$afluencia)
summary(alpha_otimo)

##
## Forecast method: Simple exponential smoothing
##
## Model Information:
## Simple exponential smoothing
##
## Call:
##   ses(y = base$afluencia)
##
##   Smoothing parameters:
##     alpha = 0.9999
##
##   Initial states:
##     l = 2283.6777
##
##   sigma: 797.7704
```

```

##          AIC      AICc      BIC
## 5023.986 5024.078 5034.725
##
## Error measures:
##               ME      RMSE      MAE      MPE      MAPE      MASE
ACF1
## Training set 6.912394 794.7543 521.2758 -7.912437 31.65711 1.005385
0.1856043
##
## Forecasts:
##    Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
## 266      4115.279 3092.8951 5137.663 2551.67772 5678.880
## 267      4115.279 2669.4821 5561.076 1904.12342 6326.435
## 268      4115.279 2344.5762 5885.982 1407.22268 6823.335
## 269      4115.279 2070.6645 6159.894 988.31097 7242.247
## 270      4115.279 1829.3419 6401.216 619.23994 7611.318
## 271      4115.279 1611.1688 6619.389 285.57286 7944.985
## 272      4115.279 1410.5372 6820.021 -21.26656 8251.825
## 273      4115.279 1223.7936 7006.764 -306.86632 8537.424
## 274      4115.279 1048.3999 7182.158 -575.10790 8805.666
## 275      4115.279 882.5081 7348.050 -828.81741 9059.375

# alpha = 0.9999

# SUAVIZACAO EXPONENCIAL DE HOLT
#####
beta1 <- holt(base$afluencia, alpha = 0.6, beta = 0.4)
summary(beta1)

##
## Forecast method: Holt's method
##
## Model Information:
## Holt's method
##
## Call:
## holt(y = base$afluencia, alpha = 0.6, beta = 0.4)
##
## Smoothing parameters:
##   alpha = 0.6
##   beta  = 0.4
##
## Initial states:
##   l = 4284.693
##   b = -302.6462
##
## sigma: 1033.234
##
##          AIC      AICc      BIC
## 5159.036 5159.128 5169.775
##
## Error measures:
##               ME      RMSE      MAE      MPE      MAPE      MASE

```

```

ACF1
## Training set 15.27376 1025.407 734.8264 8.257265 54.43748 1.417261
0.3940861
##
## Forecasts:
##   Point Forecast    Lo 80     Hi 80     Lo 95     Hi 95
## 266      4871.620 3547.477 6195.763 2846.5182 6896.722
## 267      6187.993 4315.372 8060.614 3324.0661 9051.919
## 268      7504.365 4869.354 10139.377 3474.4632 11534.267
## 269      8820.738 5267.689 12373.786 3386.8190 14254.657
## 270      10137.110 5542.506 14731.715 3110.2692 17163.952
## 271      11453.483 5712.136 17194.830 2672.8500 20234.116
## 272      12769.855 5788.218 19751.493 2092.3625 23447.348
## 273      14086.228 5778.879 22393.577 1381.2347 26791.221
## 274      15402.601 5690.212 25114.989 548.7848 30256.416
## 275      16718.973 5527.027 27910.919 -397.6315 33835.578

beta2 <- holt(base$afluencia)
summary(beta2)

##
## Forecast method: Holt's method
##
## Model Information:
## Holt's method
##
## Call:
## holt(y = base$afluencia)
##
## Smoothing parameters:
##   alpha = 0.9999
##   beta  = 0.007
##
## Initial states:
##   l = 3990.4784
##   b = -121.2353
##
## sigma: 802.2038
##
##      AIC      AICc      BIC
## 5028.900 5029.132 5046.799
##
## Error measures:
##               ME     RMSE     MAE     MPE     MAPE     MASE
ACF1
## Training set 63.86008 796.1264 522.2182 -2.328062 31.18327 1.007203
0.1831915
##
## Forecasts:
##   Point Forecast    Lo 80     Hi 80     Lo 95     Hi 95
## 266      4112.794 3084.7282 5140.859 2540.5032 5685.084
## 267      4110.310 2651.3686 5569.251 1879.0517 6341.568
## 268      4107.826 2314.7594 5900.893 1365.5671 6850.085
## 269      4105.342 2027.6611 6183.024  927.8031 7282.882

```

```

## 270      4102.859 1771.8446 6433.873   537.8803 7667.837
## 271      4100.375 1537.9945 6662.755   181.5522 8019.198
## 272      4097.891 1320.5967 6875.186  -149.6143 8345.397
## 273      4095.407 1116.0712 7074.744  -461.0942 8651.909
## 274      4092.924  921.9401 7263.907  -756.6771 8942.525
## 275      4090.440  736.4058 7444.474 -1039.1127 9219.993

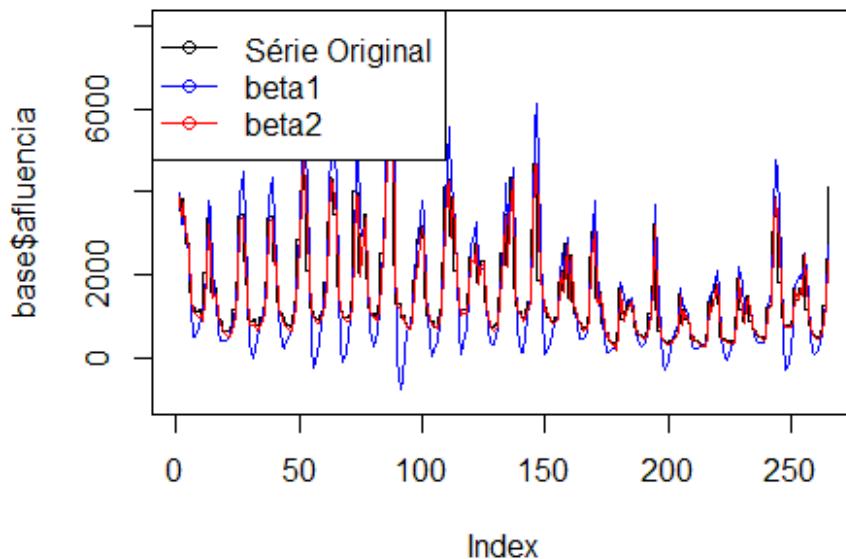
# beta1 = 0.4
# beta  = 0.007

#calculo dos erros de cada ajuste
sobradinho_resultados_holt <- list(beta1, beta2) %>% map(accuracy)

sobradinho_tabela_holt <- do.call(rbind,
lapply(seq_along(sobradinho_resultados_holt), function(i) {
  data.frame(Modelo = paste("Modelo", i), sobradinho_resultados_holt[[i]])))
rownames(sobradinho_tabela_holt) <- c("beta1", "beta2")

#gráfico do ajuste
plot(base$afluencia, type="s", ylim = c(-1000, 8000))
lines(fitted(beta1), col = "blue")
lines(fitted(beta2), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Série Original", "beta1", "beta2"),
       pch = 1)

```



```

# SUAVIZACAO EXPONENCIAL DE HOLT-WINTER
#####
base_ts <- ts(base$afluencia, frequency=12, start=c(1980,1))
base_ts

```

##	Jan	Feb	Mar	Apr	May	Jun
Jul						
## 1980	3536.4516	3843.7931	3309.5806	2737.6667	1594.8387	1260.3333
1197.0968						
## 1981	2600.9677	1561.0714	1673.5484	1154.8333	929.5161	852.5000
654.8387						
## 1982	3409.3548	3444.3214	2318.6774	1752.0333	904.9355	871.0333
921.2258						
## 1983	3353.0968	3404.6429	2343.2258	2364.3333	1322.9032	1046.3333
950.0000						
## 1984	2824.5161	3997.9310	4933.2258	4061.3333	2069.3548	1201.6667
1120.6452						
## 1985	3257.3548	4344.5000	4281.6774	3435.6667	1728.5161	1247.2667
1103.8387						
## 1986	3959.4839	2021.9286	2960.9032	3443.6000	1892.6129	1210.3333
1031.9677						
## 1987	5209.7419	6333.5000	6352.2903	2225.6333	1303.6129	1270.7667
977.0645						
## 1988	2028.5806	2859.3448	3163.5484	2865.0000	1425.4194	1028.4000
870.9677						
## 1989	4113.5484	4280.0000	2956.1290	3859.0000	2098.0645	1393.3333
1082.2581						
## 1990	2765.4839	1802.1429	2234.5161	2331.3333	1327.0968	1107.6667
799.0323						
## 1991	3465.8065	2244.2857	3307.4194	4339.0000	1649.3548	1267.0000
1085.1613						
## 1992	4662.2581	4681.0345	1835.4839	1748.6667	1162.9032	1210.9333
958.4839						
## 1993	1635.9032	2740.3571	1435.3226	2458.0000	1331.6129	1067.3333
880.8710						
## 1994	3046.7742	1425.0000	1342.5806	1569.6667	815.1613	560.0000
568.0645						
## 1995	895.4839	1094.2857	1237.0968	1386.0000	1086.7742	742.6667
616.1290						
## 1996	1711.2903	3202.7586	1139.0323	636.3333	465.8065	400.0000
371.6129						
## 1997	805.1613	1142.5000	938.0645	811.6667	447.7419	484.3333
399.0323						
## 1998	1114.8629	1572.3321	1778.1094	1279.3907	504.9145	430.5653
400.2294						
## 1999	1165.6510	854.7500	1291.2903	1474.6667	872.5806	684.6667
643.2258						
## 2000	1261.2903	3039.3103	3854.8387	3391.3333	1803.2258	946.8333
780.6452						
## 2001	1676.1290	1522.8571	2470.9677	1140.6667	773.2258	620.6667
528.3871						
## 2002	4115.4545					
	Aug	Sep	Oct	Nov	Dec	
## 1980	1120.3226	1143.6667	1026.1290	2030.0000	3347.4194	
## 1981	628.7097	628.6667	709.0323	1134.0000	1827.4194	
## 1982	771.7097	777.5000	977.0323	1197.5667	1801.8065	
## 1983	842.9032	764.0000	720.3226	1005.6667	1359.0323	
## 1984	970.3226	940.0000	1023.5484	1120.3333	1806.7742	
## 1985	962.5161	968.0000	1021.9355	1323.0667	3997.1935	

```

## 1986 1046.5806 963.2667 1263.5484 2517.0667 3340.8065
## 1987 996.2581 823.6667 751.7742 954.1667 1831.6129
## 1988 853.0968 749.1333 953.0968 1134.7000 2256.1290
## 1989 1146.1290 1130.6667 1527.4194 2438.6667 2325.8065
## 1990 720.3226 766.3333 802.2581 1450.3333 2494.8387
## 1991 947.7419 968.3333 1163.5484 1352.0000 3055.1613
## 1992 859.2903 881.3000 967.7097 1786.3333 2079.7742
## 1993 824.3226 702.7333 734.1613 980.4667 2428.0000
## 1994 399.6774 366.3333 234.1935 601.0000 1682.2581
## 1995 529.3548 486.0000 482.9032 587.3333 1046.1290
## 1996 397.7419 402.0000 497.4194 638.0000 1516.7742
## 1997 368.3871 317.6667 314.8387 531.6667 1339.0323
## 1998 385.0590 341.4333 393.0268 943.1093 1908.6697
## 1999 561.9355 485.0000 446.1290 480.3333 1187.4516
## 2000 773.2258 741.3333 858.0645 1653.6667 1406.7742
## 2001 492.2581 502.0000 615.8065 1264.0000 2360.3226
## 2002

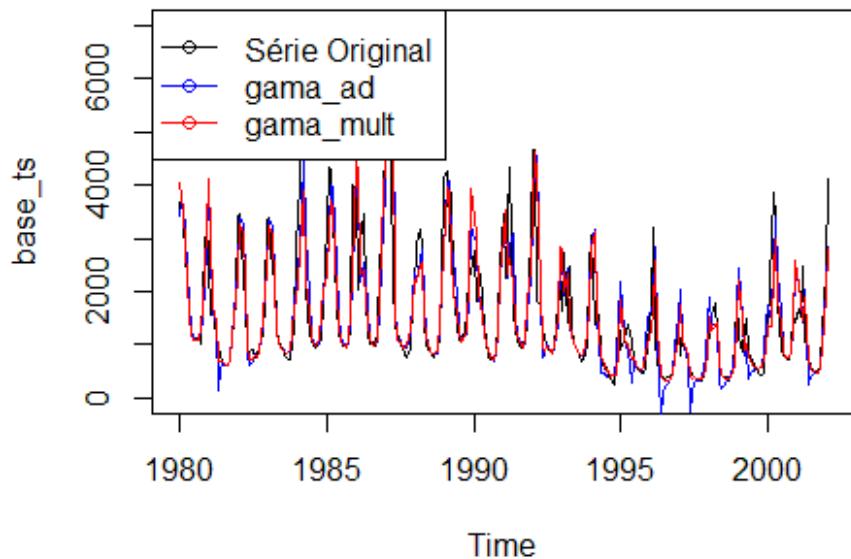
gama_ad <- hw(base_ts, seasonal = "additive")
gama_mult <- hw(base_ts, seasonal = "multiplicative")

# calculando o erro de cada ajuste
sobradinho_tabela_holtwinter <- list(gama_ad, gama_mult) %>% map(accuracy)

sobradinho_tabela_holtwinter <- do.call(rbind,
lapply(seq_along(sobradinho_tabela_holtwinter), function(i) {
  data.frame(Modelo = paste("Modelo", i),
  sobradinho_tabela_holtwinter[[i]])
}))
rownames(sobradinho_tabela_holtwinter) <- c("gama_ad", "gama_mult")

# analise grafica do ajuste
plot(base_ts, , ylim = c(0, 7000))
lines(fitted(gama_ad), col = "blue")
lines(fitted(gama_mult), col = "red")
legend("topleft", lty = 1, col = c(1, "blue", "red"),
       legend = c("Série Original", "gama_ad", "gama_mult"),
       pch = 1)

```



```

# COMPARACAO GERAL SOBRADINHO
#####
sobradinho_tabela_geral <- list(alpha_otimo, beta2, gama_mult) %>%
map(accuracy)

sobradinho_tabela_geral <- do.call(rbind,
lapply(seq_along(sobradinho_tabela_geral), function(i) {
  data.frame(Modelo = paste("Modelo", i), sobradinho_tabela_geral[[i]])
}))
rownames(sobradinho_tabela_geral) <- c("alpha_otimo", "beta2",
"gama_mult")

```