

Um estudo das avaliações de Naruto Shippuden

Alisson Rosa e Vítor Pereira

Sumário

1	Introdução	1
2	Informações	1
3	Testes	5
3.1	Tendência	5
3.2	Sazonalidade	6
3.3	Raiz unitária	6
3.4	Resultados	6
4	Modelagem	7
4.1	Dados de treino e teste	7
4.2	Recipes	7
4.3	Utilizando o Pacote Modeltime	7
4.4	Utilizando outros Pacotes	15
5	Conclusão	21
6	Referências	21
	Apêndice - Código R	21

1 Introdução

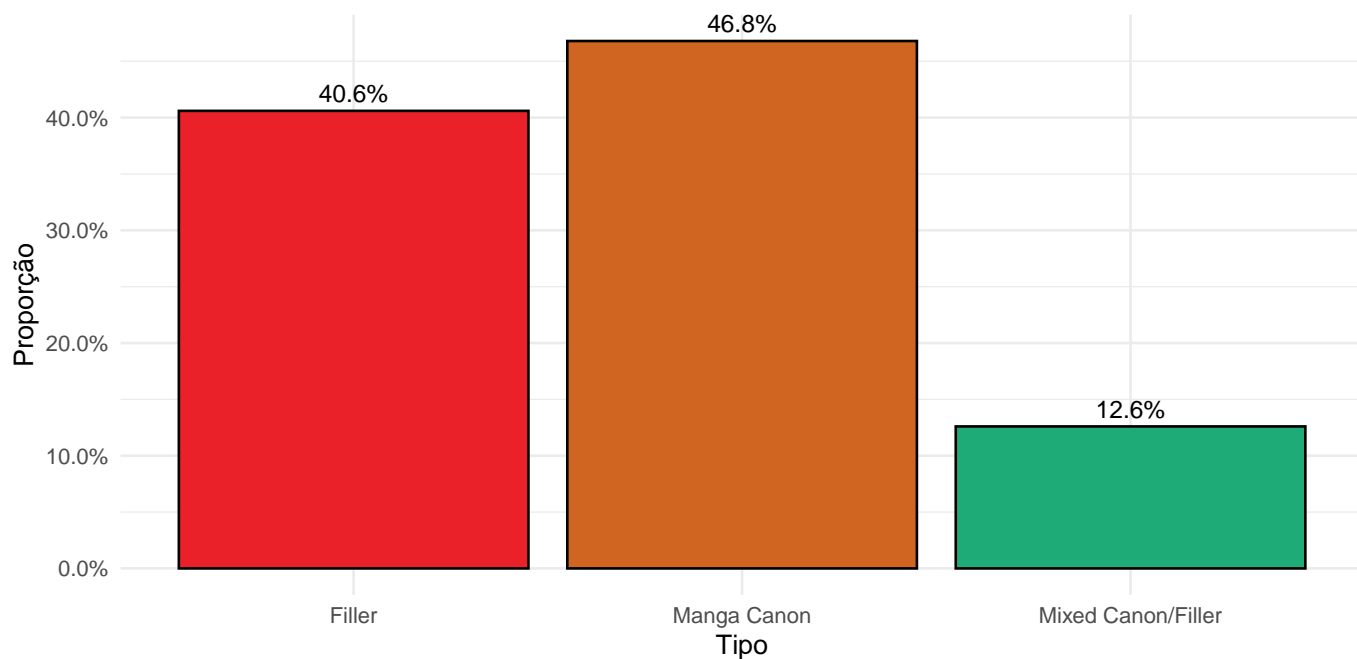
Naruto é uma animação japonesa (anime) que adapta a série de mangá escrita e ilustrada por Masashi Kishimoto, que conta a história de Naruto Uzumaki, um jovem ninja que constantemente procura por reconhecimento e sonha em se tornar Hokage, o ninja líder de sua vila.

A história é dividida em duas partes, a primeira parte se passa nos anos da pré-adolescência de Naruto (clássico), e a segunda parte se passa em sua adolescência (Shippuden). Nesse trabalho trataremos sobre o Naruto Shippuden, desenvolvendo-o em uma perspectiva de séries temporais, vamos adotar como variável de interesse a avaliação dos episódios, assim veremos pontos importantes da saga ao longo do tempo.

O Banco de dados utilizado é totalmente original, foi criado fazendo *Web scraping* de dois sites diferentes, para aqueles interessados o banco foi disponibilizado aqui.

2 Informações

Como o anime é uma adaptação, existem episódios fiéis ao mangá (Manga Canon) e episódios originais do próprio anime, em outras palavras não seguem o material original, esses episódios são chamados de Fillers, existem também episódios que seguem a trama do mangá mas além disso possuem elementos novos, esses chamados de Mixed Canon. Vejamos pelo gráfico a seguir a porcentagem do tipo dos episódios



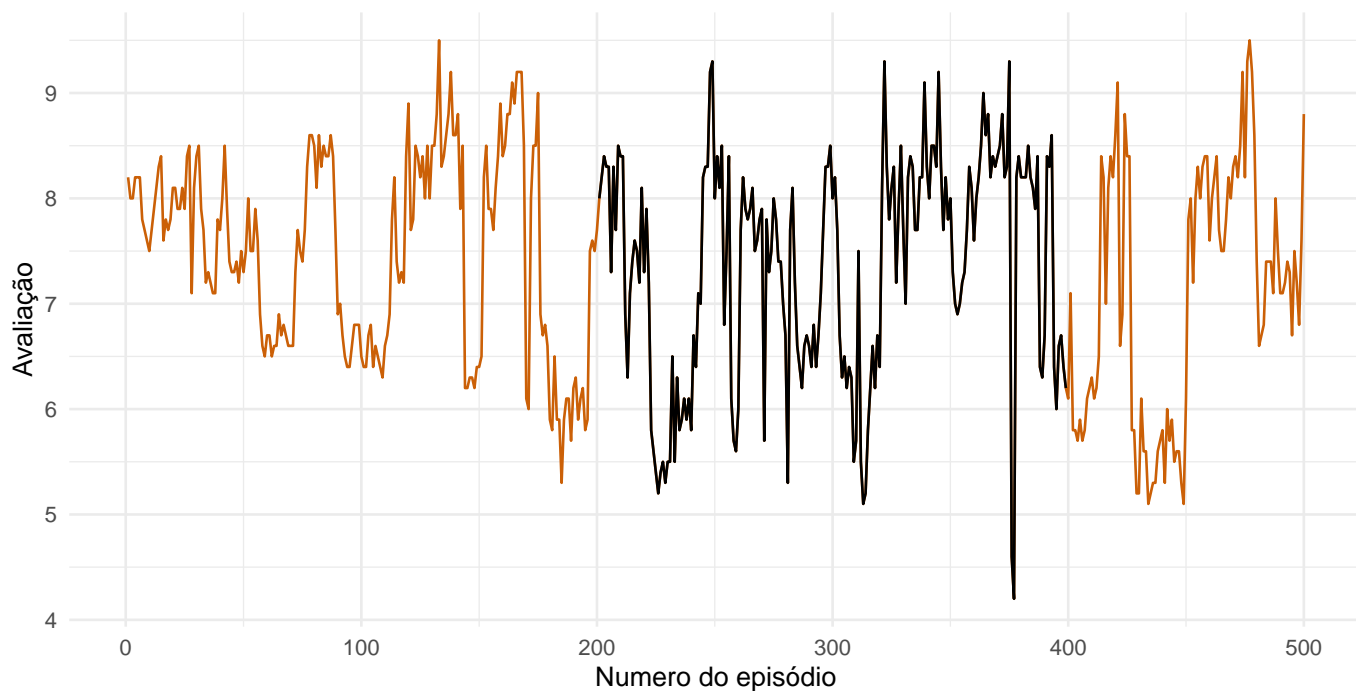
Fonte: Elaborado pelos autores

É um fato bastante curioso, Naruto Shippuden possui uma quantidade altíssima de episódios fillers, aproximadamente 41%, note pela tabela que são somente 31 episódios a mais canônicos.

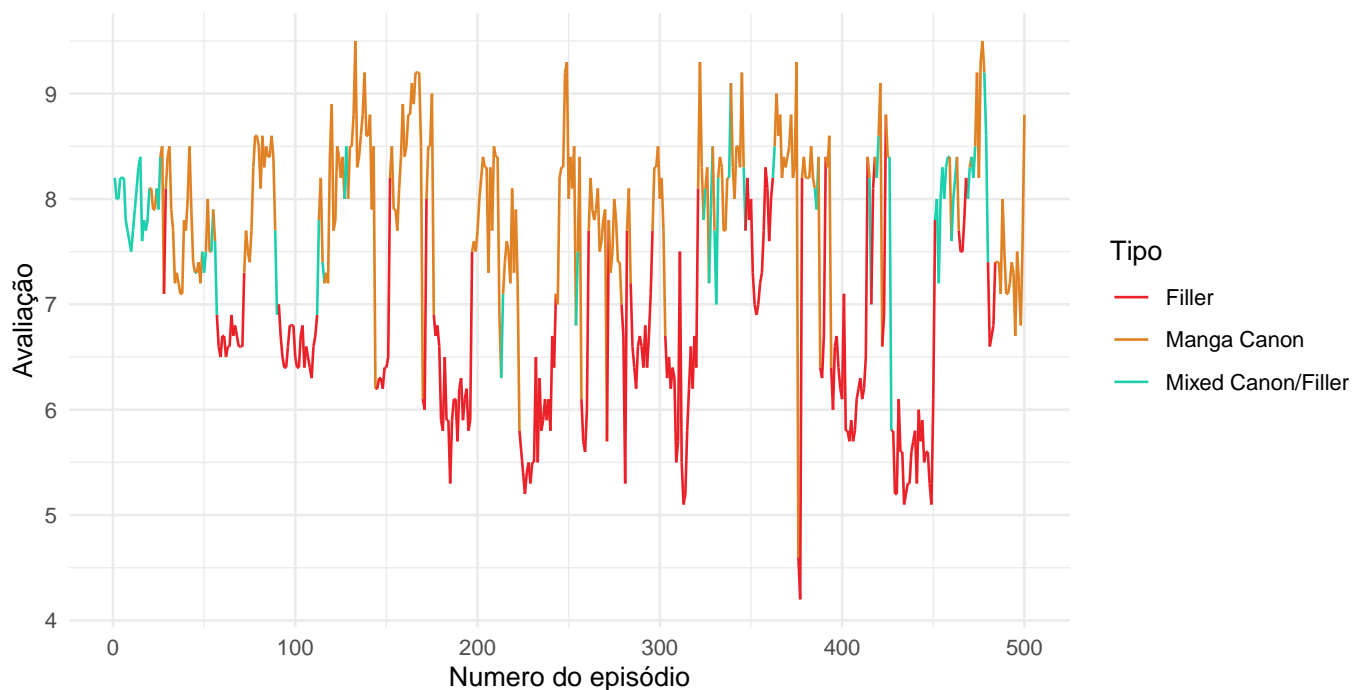
Tabela 1: Valores Absolutos do Tipo de Episódio

Tipo	Quantidade
Filler	203
Manga Canon	234
Mixed Canon/Filler	63

Em termos de avaliação ao longo do tempo, podemos utilizar como índice ordinal o próprio número do episódio já que que é uma função injetora no tempo.

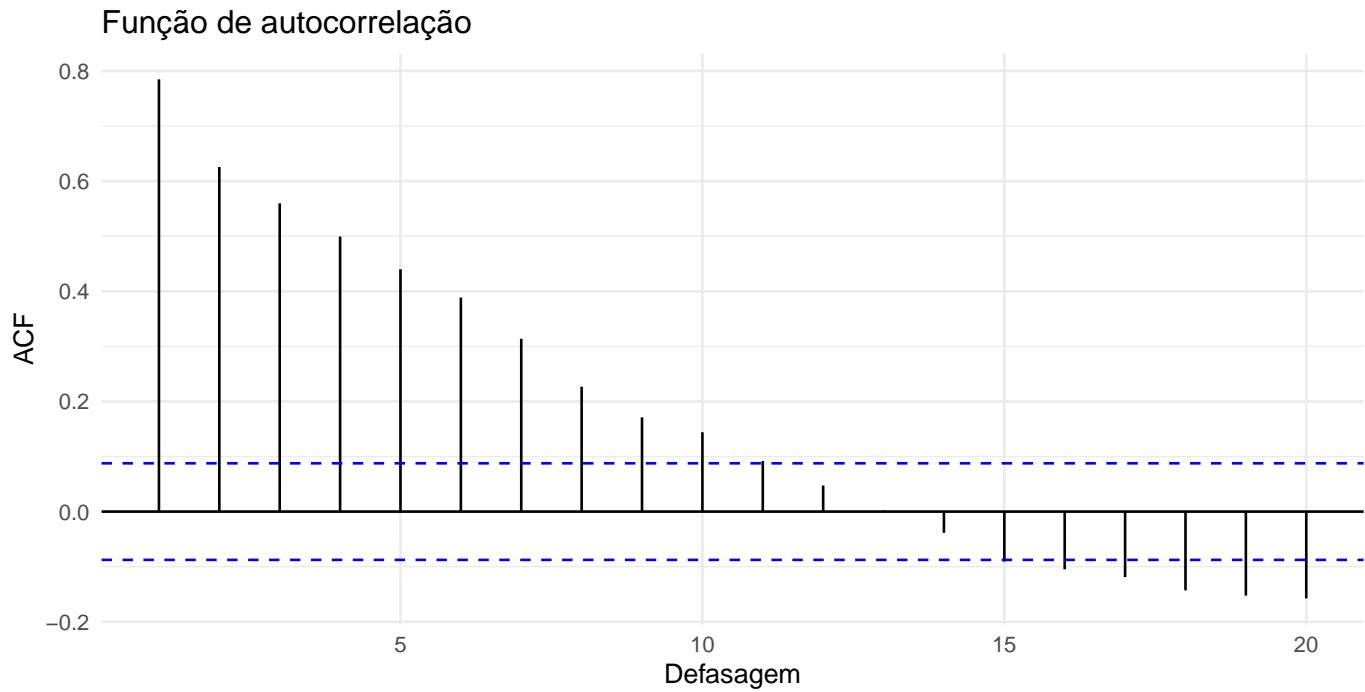


Há bastante coisa a se notar nesse gráfico, primeiramente pelos episódios 200 e 400, existe um indício de alteração na média das variáveis aleatórias, portanto furando o pressuposto de média constante ao longo do tempo, perto dos episódios finais da saga nota-se também inúmeras quedas bruscas na avaliação sem um contra-peso de avaliações com notas altas, outro indicio de não estacionariedade. É importante tentarmos entender um pouco sobre esses episódios com notas baixas, assim vamos ver esse gráfico pelo tipo de episódio.



Assim fica fácil ver que os episódios com menor avaliação são em sua maior parte fillers, toda queda brusca de avaliação tem um episódio filler envolvido.

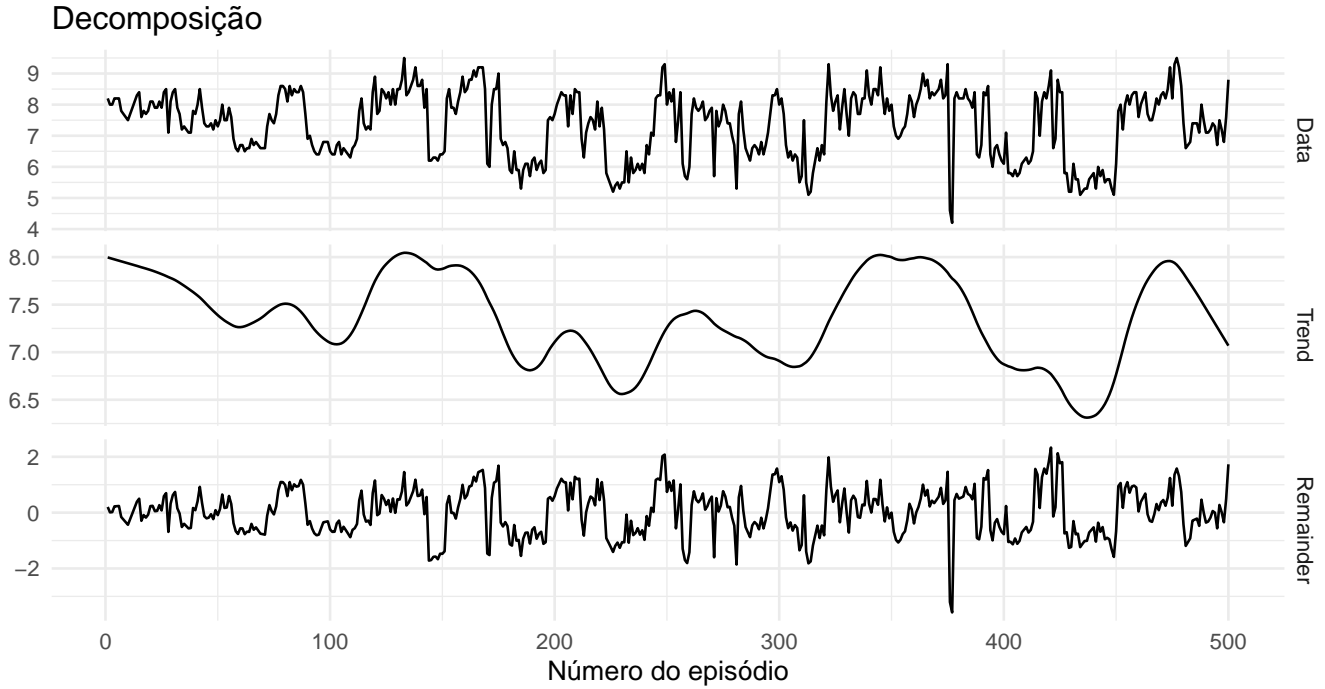
É necessário também avaliar a correlação nas avaliações:



Note que o gráfico anterior as autocorrelações tendem a ficar significativas até 10 defasagens, pois em média a duração dos arcos é 23.8, assim 10 defasagens significa em geral que:

- o episódio está tendo influência do final do arco anterior;
- o episódio está gozando consequências dos episódios iniciais do arco que está situado;
- o episódio está apanhando desfechos dos episódios pós metade do arco que está situado

Uma abordagem importante também é decompor a série temporal, apriori supõe-se a existência de três elementos, a saber: Tendência, sazonalidade e resíduo. Todos esses serão explorados cuidadosamente nas seções posteriores, assim vamos decompor a série para ter um vislumbre de tais elementos:



A questão de curiosidade vamos ver se os resíduos seguem uma distribuição normal, para isso utilizamos o teste de Shapiro Wilk que possui como hipótese:

$$H_0 : \text{Os dados seguem uma distribuição Normal}$$

Efetuada o teste obtemos um p-valor de 0.006, assim portanto rejeitando a hipótese H_0 , porém os resíduos não possuírem distribuição normal não afetará as análises daqui pra frente.

3 Testes

Os gráficos de seção anterior evidenciaram a possibilidade da série não ser estacionária, assim faz-se necessário verificar se vale de fato para o processo estocástico, com isso vamos precisar aplicar testes de hipóteses para averiguar algumas propriedades, como existência de tendência e sazonalidade.

3.1 Tendência

Tendência refere-se a um algum comportamento não - estocástico da série em algum momento do tempo, se tal comportamento só acontece em alguns momentos específicos do tempo, chamamos de tendência estocástica, do contrário é dita determinística. Vamos começar pelos testes de tendência determinística, que em termos sumarizados possuem como hipótese:

Tabela 2: Testes de Tendência Determinística

Testes	P-valor
Cox-Stuart	0.050
Wald-Wolfowitz	0.000
Mann-Kendall	0.039

Conforme mostrado pela Tabela anterior podemos ver que pelos testes de Cox-Stuart, Wald-Wolfowitz, Mann-Kendall, temos que todos os p-valores são menores pequenos, então concluímos que existe tendência determinística na série temporal.

3.2 Sazonalidade

Sazonalidade acontece quando a série possui um comportamento que se repete frequencialmente, vamos nessa subseção testar se existe sazonalidade nas avaliação dos episódios, assim temos como hipótese:

$$H_0 : \text{A série não possui sazonalidade}$$

Os testes aplicados foram Kruskal-Wallis, Friedman e Autocorrelação em lags Sazonais, assim gerando a seguinte tabela:

Tabela 3: Testes de Sazonalidade

Testes	P-valor
Kruskal-Wallis	0.745
Friedman	0.878
Autocorrelação em lags Sazonais	0.701

Assim, como os testes obtiveram p-valor > 0.5, concluímos portanto que não existem indícios de sazonalidade.

3.3 Raiz unitária

Para começar tal seção, primeiros vamos definir um passeio aleatório em sua forma simplificada:

$$Y_t = Y_{t-1} + \epsilon_t$$

Onde t refere-se aos índices de ordenação, e ϵ um termo aleatório, para facilidade vamos assumir que $E(\epsilon_t) = \mu$ e $var(\epsilon_t) = \sigma^2 \forall t$, assim subtraindo-se Y_{t-1} em ambos os lados tem-se

$$Y_t - Y_{t-1} = \epsilon_t$$

que é um processo estacionário, o processo anterior é dito processo estacionário em diferença, em outros casos também chamado de processo integrado. O exemplo anterior trata-se de um caso mais geral

$$Y_t = \phi Y_{t-1} + \epsilon_t$$

onde evidentemente $\phi = 1$, é fácil mostrar que se $|\phi| < 1$ tem-se um processo estocástico, aqui portanto, estamos interessados nesse caso, para isso utilizaremos testes de hipótese.

Fala-se também que o caso anterior na forma simplificada possui raiz unitária, por causa do operador Lag, aqui não definido, porém pode-se ler sobre em [1]

Tabela 4: Testes de Tendência Estocástica

Testes	P-valor
Kwiatkowski-Phillips-Schmidt-Shin (KPSS)	0.10
Augmented Dickey-Fuller (ADF)	0.01
Phillips-Perron (PP)	0.01

Conforme mostrado pela Tabela anterior podemos ver que pelos testes de Dickey-Fuller Aumentado, Phillips-Perron, KPSS, com índice de significância de 5%, temos que os p-valores dos testes ADF e PP são menores que 0.05 e do teste KPSS é maior que 0.05, então concluímos que não existe tendência estocástica na série temporal. Pois, nos testes ADF e PP, a hipótese nula é a existência de raiz unitária e no teste de KPSS, a hipótese nula é não existência de raiz unitária.

3.4 Resultados

Vimos visualmente que a série a partir de um certo começa ter uma descrescimento, assim portanto dando um vislumbre de tendência determinística, aplicando os testes obtemos mais uma evidência de existência de tendência na série. Em relação a sazonalidade os testes também confirmaram o que os gráficos mostraram: não existe evidência de sazonalidade na série.

4 Modelagem

Nesta seção trabalharemos com os modelos para a série temporal, realizando a análise de métricas (MAE, MAPE, MASE, SMAPE, RMSE e RSQ), análise de resíduos e realizando previsões.

4.1 Dados de treino e teste

Utilizaremos para treinar os nossos modelos uma proporção de 80%, como temos uma amostra grande, os 20% são suficientes para os dados de teste, assim a série ficará da seguinte maneira:

Avaliação de Naruto Shippuden Dados de Treino e Teste



4.2 Recipes

Utilizaremos 6 recipes (receitas): normal, mês e ano como covariáveis, dia do ano como covariável, trimestre e semestre, utilizando lags e utilizando séries de Fourier.

4.3 Utilizando o Pacote Modeltime

Aqui iremos testar diversos modelos como: suavização exponencial (ets), suavização exponencial no modelo de espaço de estado SSOE, suavização exponencial theta (equivalente a suavização exponencial simples com tendência constante (drift) e o procedimento Prophet do Facebook, é um modelo aditivo utilizando tendências não lineares, para entender mais o modelo Prophet [clique aqui](#).

4.3.1 Suavização exponencial

Começaremos utilizando todos os modelos ets possíveis, smooth_es, theta e croston. Assim, para escolhermos os melhores, de acordo principalmente com três métricas: MAE, RMSE e R^2 . Sendo ao total 51 modelos testados.

4.3.1.1 Ajustando os modelos Para os modelos de suavização exponencial iremos os escolher pelo menos um modelo que seja o melhor em cada uma dessas categorias: Erro Aditivo, Erro Multiplicativo, Sem Tendência, Tendência Aditiva, Tendência Multiplicativa, Tendência Amortecida, Sem Sazonalidade, Sazonalidade Aditiva, Sazonalidade Multiplicativa e bom R^2 , para o ajuste do modelo foi utilizado o pacote Sknifedatar em conjunto com o ModelTime, sendo essa uma maneira mais simples e mais próxima do tidyverse de acordo com [2].

Já no ajuste de modelos podemos perceber que a suavização exponencial não pode utilizar de algumas combinações de erro, tendencia e sazonalidade, esses são modelos são: ets(A,N,M), ets(A,A,M), ets(A,AD,M), ets(A,M,N), ets(A,MD,N), ets(A,M,A), ets(A,MD,A), ets(A,M,M), ets(A,MD,M), ets(M,M,A) e ets(M,MD,A). Assim, temos que entre os modelos válidos, escolhendo a melhor combinação de receita e modelos, temos as seguintes métricas para cada modelo: (ordenados pela métrica MAE)

Tabela 5: Métricas dos Modelos de Suavização Exponencia

rank	.model_desc	mae	mape	mase	smape	rmse	rsq
1	ETSX(MAN)	0.9482563	13.76290	2.063239	13.52065	1.107536	0.1541240
3	ETS(MAN)	0.9503808	13.83595	2.067861	13.53914	1.107940	0.1541242
4	ETS(M,A,N)	0.9506912	13.84659	2.068537	13.54501	1.107840	0.1541242
10	ETS(MAA)	0.9588516	13.84171	2.086293	13.66376	1.123567	0.1329984
11	ETSX(MAA)	0.9598439	13.87724	2.088452	13.67609	1.123165	0.1331725
13	CROSTON METHOD	1.0566146	15.54127	2.299008	15.09949	1.204980	NA
19	ETSX(MMM)	1.0910167	14.79220	2.373861	15.57832	1.300304	0.0827720
21	ETS(MMM)	1.1242293	15.11030	2.446125	16.07203	1.344843	0.0703107
22	ETS(M,M,M)	1.1572583	15.51976	2.517991	16.57406	1.377681	0.0310785
28	ETSX(ANM)	1.1764872	16.02217	2.559829	16.87385	1.364653	0.0076139
29	ETSX(MNN)	1.2462616	16.49726	2.711646	17.94631	1.480544	0.1541241
31	ETS(M,N,N)	1.2529108	16.54194	2.726114	18.05000	1.493865	NA
37	ETS(MNN)	1.2529333	16.54211	2.726163	18.05036	1.493907	NA
39	ETS(M,MD,N)	1.2538124	16.55009	2.728075	18.06417	1.495339	0.0849640
45	ETS(M,AD,N)	1.2566748	16.58070	2.734303	18.10941	1.499276	0.1351984
51	ETSX(ANA)	1.2572438	16.66570	2.735541	18.12533	1.493752	0.0080862
52	ETSX(ANN)	1.2597240	16.59663	2.740938	18.15692	1.506055	0.1541242
53	ETS(M,N,A)	1.2616011	16.66740	2.745022	18.19174	1.503203	0.0047931
59	ETS(M,MD,M)	1.2622415	16.68977	2.746415	18.20149	1.501747	0.0119256
65	ETS(ANM)	1.2647154	16.70804	2.751798	18.24044	1.507543	0.0092757
66	ETS(ANA)	1.2664086	16.73464	2.755482	18.26878	1.509865	0.0090474
68	ETS(M,AD,M)	1.2692679	16.74742	2.761704	18.31245	1.513117	0.0100718
74	ETS(M,A,M)	1.2695095	16.76412	2.762229	18.31714	1.513554	0.0140288
80	ETS(A,AD,A)	1.2720614	16.77874	2.767782	18.35787	1.519923	0.0101815
87	ETS(A,N,N)	1.2764738	16.74424	2.777383	18.42147	1.534184	NA
93	ETS(ANN)	1.2766417	16.74570	2.777748	18.42413	1.534473	NA
94	ETS(A,AD,N)	1.2767148	16.74648	2.777907	18.42529	1.534574	0.0624370
100	ETS(A,N,A)	1.2785790	16.84264	2.781963	18.46063	1.530631	0.0125995
106	ETS(M,N,M)	1.2830405	16.90115	2.791671	18.53364	1.535727	0.0112603
112	THETA METHOD	1.2846034	16.83626	2.795071	18.55142	1.544815	0.1541242
118	ETS(M,A,A)	1.2913573	17.00737	2.809766	18.66618	1.541874	0.0222350
124	ETS(M,AD,A)	1.3483684	17.57995	2.933813	19.58484	1.626831	0.0582191
130	ETS(A,A,N)	1.4355075	18.59210	3.123412	21.03802	1.742123	0.1541242
136	ETSX(AAN)	1.4418638	18.66746	3.137242	21.14579	1.750517	0.1541242
137	ETS(AAN)	1.4423830	18.67268	3.138372	21.15451	1.751335	0.1541242
138	ETS(A,A,A)	1.4551112	18.85483	3.166066	21.37075	1.766832	0.1171396
145	ETS(AMA)	1.7832282	22.88661	3.879991	27.25304	2.187881	0.1628969
146	ETSX(AMA)	1.8083857	23.20880	3.934729	27.73209	2.219819	0.1624729

Assim os modelos escolhidos foram:

- ETSX(M,A,N) - Melhor modelo;
- ETS(M,A,N) - Melhor modelo da suavização exponencial normal;
- CROSTON - Bom método num geral e é de engine diferente;

- THETA - Bom R^2 e é de engine diferente;
- ETSX(M,A,A) - Melhor modelo com sazonalidade aditiva;
- ETSX(M,M,M) - Melhor modelo com sazonalidade multiplicativa;
- ETSX(A,N,M) - Melhor modelo com erro aditivo e sem tendencia;
- ETS(M,M,M) - Segundo melhor modelo da suavização exponencial normal;
- ETS(M, AD, N) - Melhor modelo com tendencia amortecida;
- ETSX(M,N,N) - Modelo equilibrado em todas as medidas e com bom R^2 entre os ETS();
- ETSX(A,M,A) - Modelo com melhor R^2 .

4.3.2 Modelos Prophet

Para a seleção dos modelos prophet realizamos o procedimento de tunar hiperparâmetros considerando duas características, sazonalidade, poderia aditiva ou multiplicativa, crescimento, poderia linear ou logístico, assim como utilizamos duas engines para tunar os hiperparâmetros: XGBoost e Catboost.

Assim utilizaremos 4 modelos Prophets para o ajuste de modelos e comparação com os modelos de suavização exponencial:

- Prophet com todos hiperparâmetros padrões e engine xgboost;
- Prophet com crescimento logístico, changepoint_range 0.8, restante de hiperparâmetros padrões e engine xgboost;
- Prophet com crescimento logístico, changepoint_range 0.8628, changepoint_num 33, trees 2000, tree_depth 2, learn_rate 0.000975, mtry 7 e engine xgboost;
- Prophet com crescimento linear, changepoint_range 0.7323, changepoint_num 29, trees 2000, tree_depth 4, learn_rate 0.00335856, mtry 9 e engine catboost.

4.3.3 Ajustando os modelos

As métricas utilizadas para escolher os modelos são:

MAE - Erro médio de previsão - calcula o “erro absoluto médio” dos erros entre valores observados (reais) e predições (hipóteses);

$$MAE = \frac{1}{n} \sum_{t=1}^n |e(t)|$$

MAPE - Percentual médio absoluto de erro - A média dos erros percentuais absolutos (MAPE) é uma das medidas de acurácia mais utilizada, devido as vantagens de interpretação e independência na escala;

$$MAPE = \left(\frac{1}{n} \sum_{t=1}^n \left| \frac{e(t)}{X_t} \right| \right)$$

MASE - Erro Escalonado Absoluto Médio - Erros escalonados foram propostos como uma alternativa ao uso de erros percentuais ao comparar a precisão da previsão em séries com unidades diferentes. Eles propuseram escalonar os erros com base no treinamento MAE a partir de um método simples de previsão;

$$MASE = \frac{1}{n} \sum_{t=1}^n |q_t|$$

RMSE - Raiz quadrada da média dos erros quadráticos - Se a acurácia de técnicas de previsão é avaliada em séries temporais que apresentam a mesma escala, uma das medidas sugeridas é a raiz quadrada da média dos erros quadráticos (RMSE);

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n [e(t)]^2}$$

SMAPE - Erro percentual absoluto médio simétrico - É um medida baseada no percentual relativo de erros;

$$SMAPE = (\frac{1}{n} \sum_{t=1}^n |\frac{F_t - A_t}{\frac{F_t + A_t}{2}}|)$$

R² - Coeficiente de determinação - É uma medida de ajuste do modelo aos dados reais.

$$\hat{R}^2 = 1 - \frac{n-1}{n-(k_1)}(1 - R^2)$$

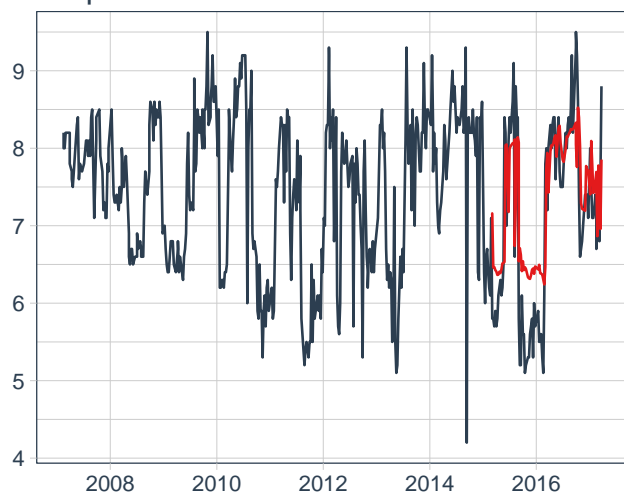
Com os modelos ajustados, agora podemos ver quais são os 20 modelos possuem as melhores métricas, ordenados de acordo com o RMSE:

Tabela 6: Métricas nos dados de teste

rank	Modelo	mae	mape	mase	smape	rmse	rsq
1	PROPHET CATBOOST	0.632	9.59	1.38	9.16	0.785	0.648
2	PROPHET XGBOOST TUNNADO	0.653	9.46	1.42	9.33	0.806	0.639
3	PROPHET XGBOOST	0.701	9.55	1.53	9.88	0.922	0.495
4	PROPHET XGBOOST DEFAULT	0.764	11.07	1.66	10.87	0.950	0.388
5	ETSX(MAN)	0.948	13.76	2.06	13.52	1.108	0.154
6	ETS(M,A,N)	0.951	13.85	2.07	13.54	1.108	0.154
7	ETS(M,A,N)	0.951	13.85	2.07	13.54	1.108	0.154
8	ETS(M,A,N)	0.951	13.85	2.07	13.54	1.108	0.154
9	ETS(M,A,N)	0.951	13.85	2.07	13.54	1.108	0.154
10	ETS(M,A,N)	0.951	13.85	2.07	13.54	1.108	0.154
11	ETS(M,A,N)	0.951	13.85	2.07	13.54	1.108	0.154
12	ETSX(MAN)	0.950	13.83	2.07	13.54	1.108	0.154
13	ETS(MAN)	0.950	13.84	2.07	13.54	1.108	0.154
14	ETSX(MAA)	0.960	13.88	2.09	13.68	1.123	0.133
15	ETS(MAA)	0.959	13.84	2.09	13.66	1.124	0.133
16	PROPHET XGBOOST DEFAULT	0.987	13.83	2.15	14.08	1.140	0.317
17	PROPHET XGBOOST DEFAULT	0.995	13.97	2.16	14.19	1.150	0.259
18	PROPHET CATBOOST	0.998	14.33	2.17	14.25	1.153	0.198
19	ETSX(MAA)	0.997	14.89	2.17	14.13	1.162	0.141
20	PROPHET CATBOOST	1.008	15.06	2.19	14.41	1.173	0.053

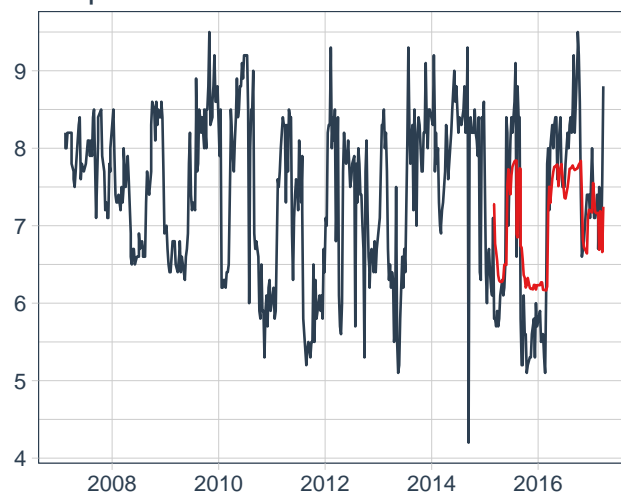
4.3.3.1 Gráficos Nesta seção será mostrado alguns gráficos interessantes dos seguintes modelos: Prophet's, ET SX(MAN), ETS(MAN) do smooth, ETS(MAN) do forecast e ETS(MAA) do smooth.

Prophet com Catboost



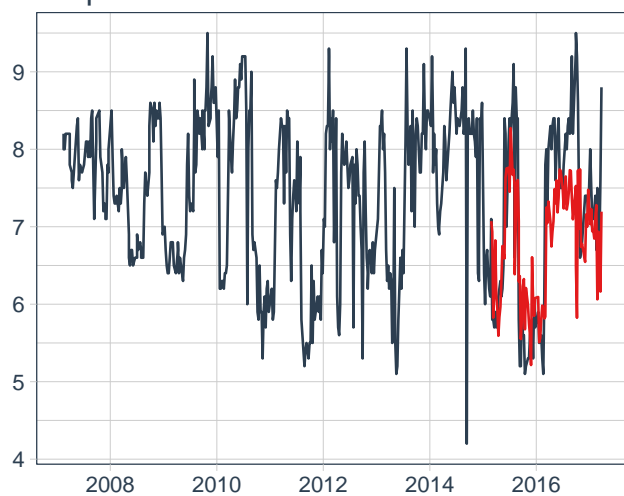
Legend — ACTUAL — 1_PROPHET CATBOOST

Prophet com XGboost tunnado



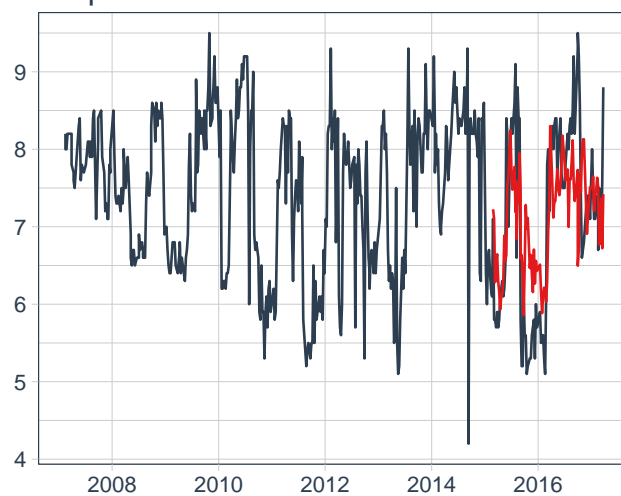
Legend — ACTUAL — 1_PROPHET XGBOOST TUNNAD

Prophet com XGboost



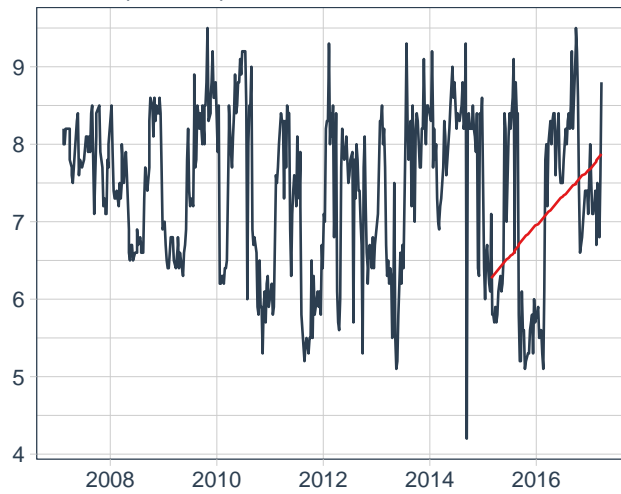
Legend — ACTUAL — 1_PROPHET XGBOOST

Prophet com XGboost default



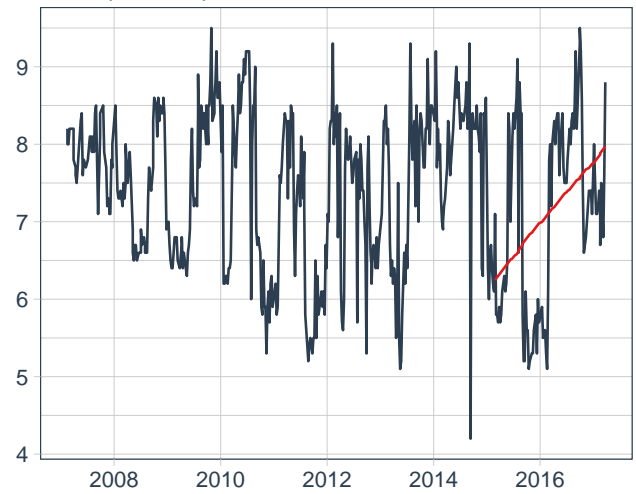
Legend — ACTUAL — 1_PROPHET XGBOOST DEFAULT

ETSX(M,A,N)



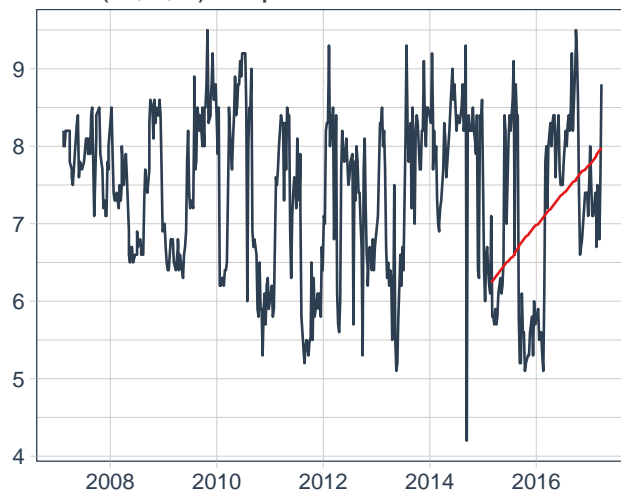
Legend — ACTUAL — 1_ETSX(MAN)

ETS(M,A,N)



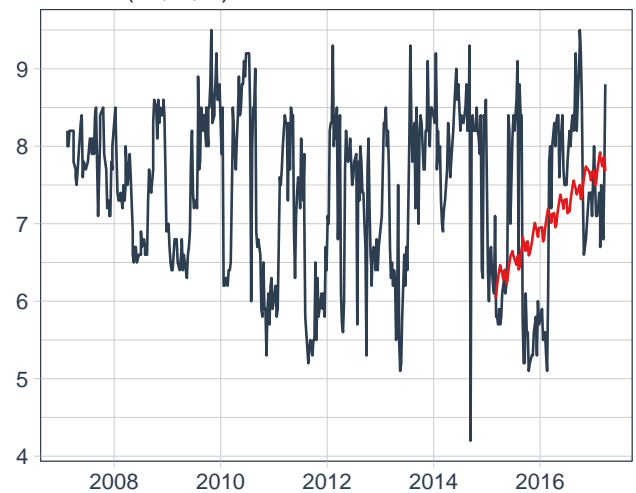
Legend — ACTUAL — 1_ETS(M,A,N)

ETS(M,A,N) do pacote smooth



Legend — ACTUAL — 1_ETS(MAN)

ETSX(M,A,A)



Legend — ACTUAL — 1_ETS(MAA)

Assim conseguimos compreender a diferença entre os modelos ETS e os modelos PROPHET nos dados de teste, podemos ver que os modelos PROPHET se adaptam muito melhor.

4.3.4 Reajuste do modelo

Agora iremos reajustar o modelo utilizando não apenas os dados de treino, mas toda a base de dados.

Para começar a análise dos modelos reajustados temos, as seguintes métricas para os dados completos:

Tabela 7: Métricas na série inteira

Modelo	MAE	MAPE	MASE	SMAPE	RMSE	RSQ
CROSTON METHOD	0.718	10.27	1.625	9.97	0.911	0.262
ETS(M,A,N)	0.463	6.57	1.049	6.39	0.685	0.600
ETS(M,M,M)	0.470	6.66	1.064	6.48	0.682	0.603
ETS(M,MD,N)	0.466	6.57	1.054	6.43	0.685	0.598
PROPHET XGBOOST DEFAULT	0.848	12.18	1.921	11.78	1.010	0.085
PROPHET XGBOOST	0.858	12.34	1.942	11.91	1.019	0.069
PROPHET XGBOOST TUNNADO	0.907	13.04	2.054	12.56	1.059	0.014
ETS(AMA)	0.452	6.34	1.022	6.24	0.670	0.623
ETS(ANM)	0.451	6.34	1.020	6.23	0.670	0.623
ETS(MAA)	0.462	6.54	1.046	6.36	0.677	0.612
ETS(MAN)	0.464	6.58	1.049	6.40	0.685	0.599
ETS(MMM)	0.467	6.60	1.057	6.44	0.679	0.607
ETS(MNN)	0.466	6.57	1.054	6.43	0.685	0.598
THETA METHOD	0.450	6.32	1.018	6.21	0.679	0.614
CROSTON METHOD	0.718	10.27	1.625	9.97	0.911	0.262
ETS(M,A,N)	0.463	6.57	1.049	6.39	0.685	0.600
ETS(M,M,M)	0.470	6.66	1.064	6.48	0.682	0.603
ETS(M,MD,N)	0.466	6.57	1.054	6.43	0.685	0.598
PROPHET XGBOOST DEFAULT	0.288	4.07	0.653	4.02	0.380	0.896
PROPHET XGBOOST	0.276	3.89	0.625	3.85	0.367	0.902

4.3.5 Análise de resíduo

Apesar dos modelos acima apresentarem bons valores nas métricas acima, não significa que eles são inferencialmente adequadas, então nessa seção testaremos quais dos modelos ajustados são adequados de acordo com os seguintes critérios: Todos os parâmetros estimados são estatisticamente significantes, os resíduos não apresentam autocorrelação serial, correlação cruzada ou heterocedasticidade condicional, para os resíduos se comportarem como ruídos brancos e os resíduos não violam a hipótese de distribuição normal.

Assim usaremos os seguintes testes:

Shapiro-Wilk para testar a normalidade dos resíduos, com a seguinte hipótese:

$$H_0 : \text{Os dados seguem uma distribuição Normal}$$

Ljung-Box e Box-Pierce para testar a independência dos resíduos, com a seguinte hipótese:

$$H_0 : \text{Os dados são independentes e identicamente distribuídos.}$$

Durbin-Watson para analisar correlação serial nos resíduos:

$$H_0 : \text{Não há correlação serial e } \rho = 0$$

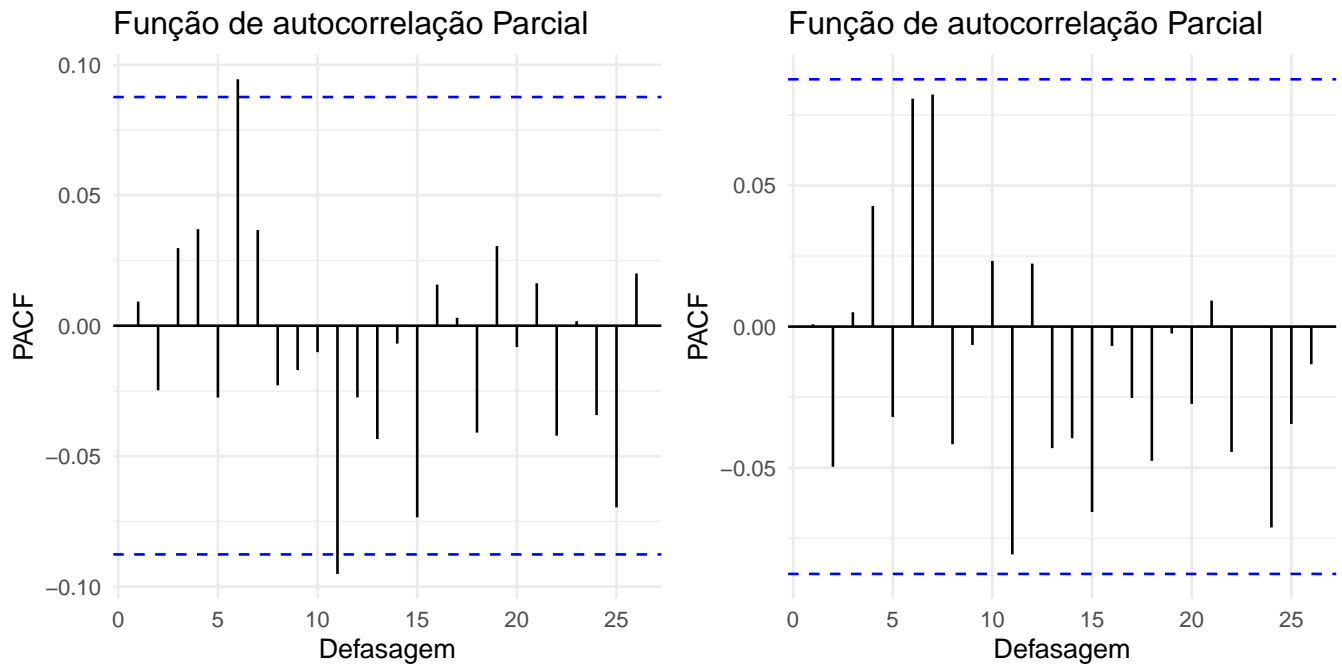
Sendo ρ o coeficiente de correlação. Nesse teste não é usado p-valor e sim a Estatística de Durbin-Watson, assim para não ter correlação serial a estatística deve ser o mais próximo de 2.

Tabela 8: Modelos que passaram nos teste de resíduos

.model_id	.model_desc	shapiro_wilk	box_pierce	ljung_box	durbin_watson
1	ETS(AMA)	0	0.235	0.234	1.88
1	ETS(ANM)	0	0.206	0.205	1.88
1	THETA METHOD	0	0.222	0.221	1.88
1	THETA METHOD	0	0.222	0.221	1.88
1	PROPHET XGBOOST DEFAULT	0	0.836	0.836	1.98
1	PROPHET XGBOOST	0	0.985	0.985	1.99
1	THETA METHOD	0	0.222	0.221	1.88
1	THETA METHOD	0	0.222	0.221	1.88
1	ETXS(AMA)	0	0.226	0.225	1.88
1	ETXS(ANM)	0	0.202	0.201	1.88
1	THETA METHOD	0	0.222	0.221	1.88
1	ETXS(AMA)	0	0.162	0.161	1.86
1	ETXS(ANM)	0	0.190	0.189	1.88
1	THETA METHOD	0	0.222	0.221	1.88

Então apenas 7 dos nosso modelos passaram nos testes de Shapiro-Wilk, Box-Pierce e Ljung-Box: ETS(A,M,A), ETS(A,N,M), THETA, PROPHET XGBOOST DEFAULT, PROPHET XGBOOST, ETXS(A,M,A) e ETXS(A,N,M). Desses nossos 7 modelos apenas 2 apresentaram boas métricas: PROPHET XGBOOST DEFAULT e PROPHET XGBOOST. Então esses seriam os modelos indicados para a predição e previsão da Série Temporal.

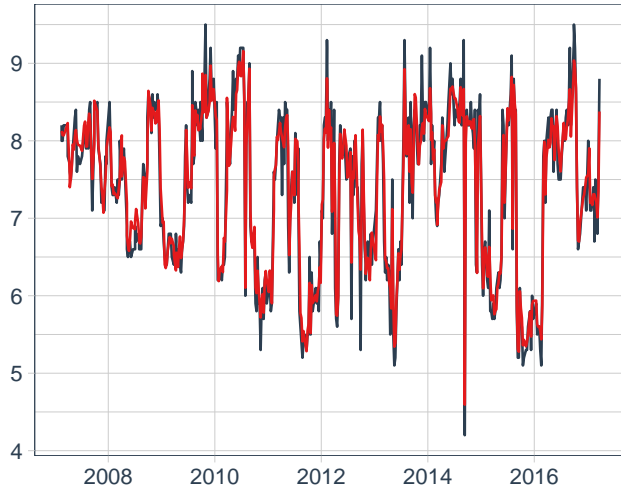
4.3.5.1 Gráfico da AutoCorrelação Parcial Além dos testes acima também podemos buscar uma confirmação visual de que os modelos são adequados, assim iremos usar a autocorrelação parcial para corroborar com o que foi mostrado pelos testes:



Na FacP do modelo PROPHET XGBOOST DEFAULT, podemos notar que em alguns pequenos pontos existe uma correlação significativa, mas estando no limite da correlação e com valores muito pequenos, assim podendo ser desconsiderada. No entanto, no modelo PROPHET XGBOOST não há nenhuma correlação que saia dos limites de não autocorrelação.

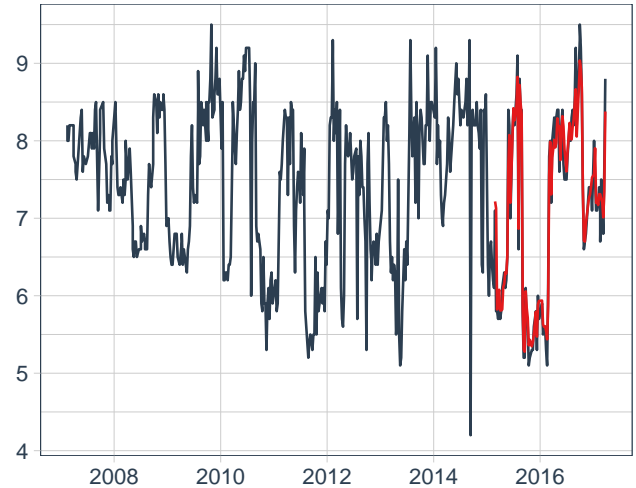
4.3.5.2 Gráficos dos modelos reajustados Nesta seção veremos os dois modelos escolhidos, que passaram na análise de resíduo e tem boas métricas:

Prophet XGboost



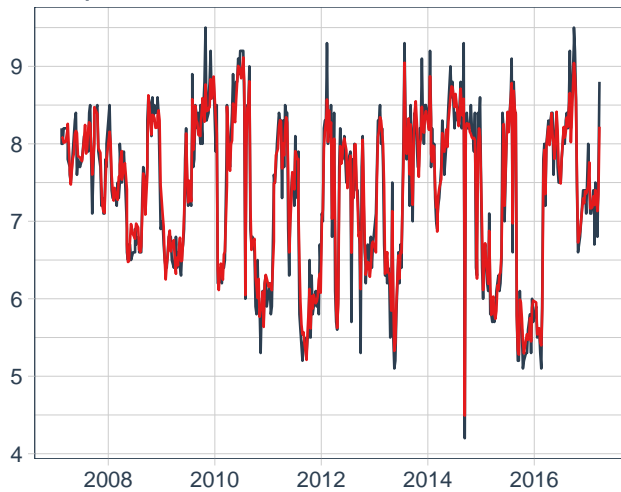
Legend — ACTUAL — 1_PROPHET XGBOOST DEFAU

Prophet XGboost dados de teste



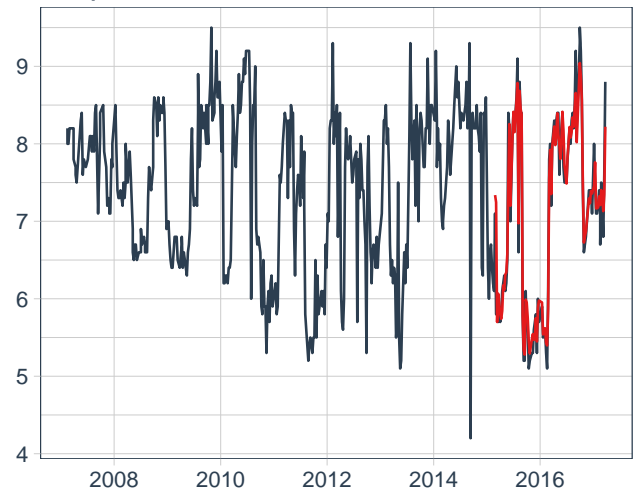
Legend — ACTUAL — 1_PROPHET XGBOOST DEFAUL

Prophet XGboost default



Legend — ACTUAL — 1_PROPHET XGBOOST

Prophet XGboost default dados de teste



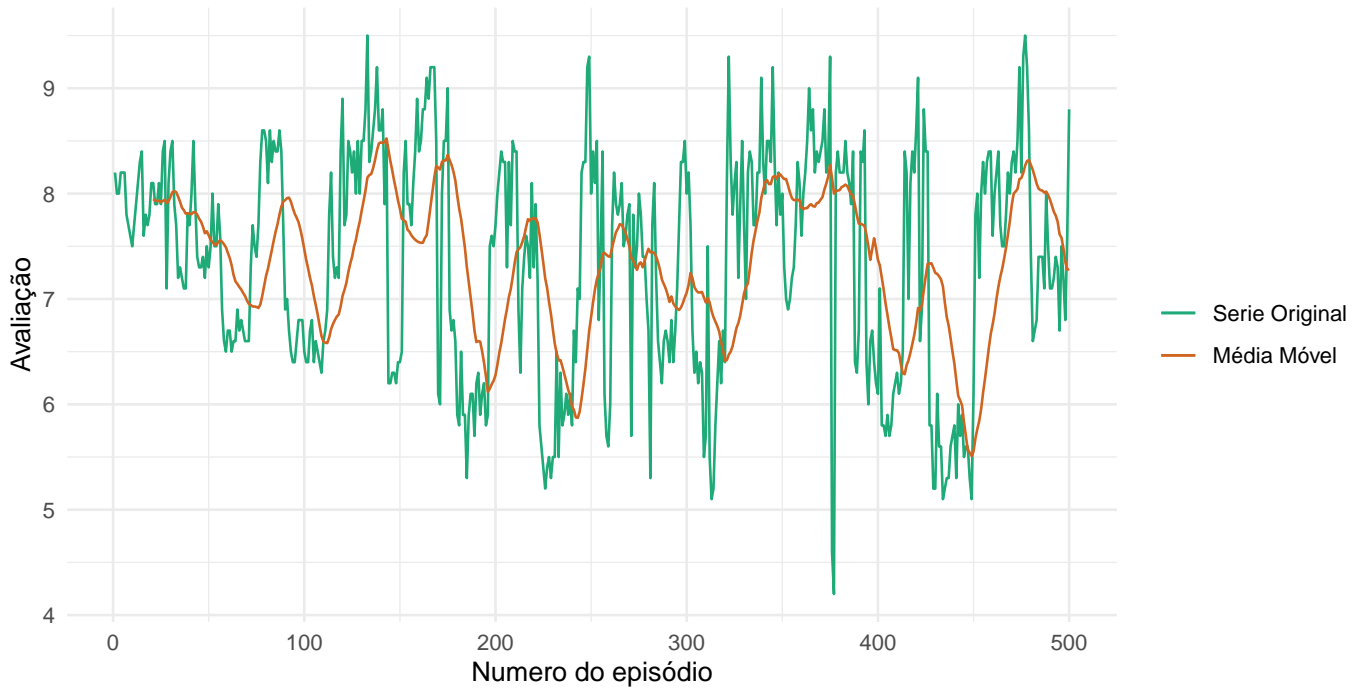
Legend — ACTUAL — 1_PROPHET XGBOOST

4.4 Utilizando outros Pacotes

Nesta seção iremos construir a modelagem de séries temporais utilizando outros pacotes além do modeltime, como o pacote forecast, smooth e SMA principalmente para construção de modelos holt, médias móveis e regressão LOESS.

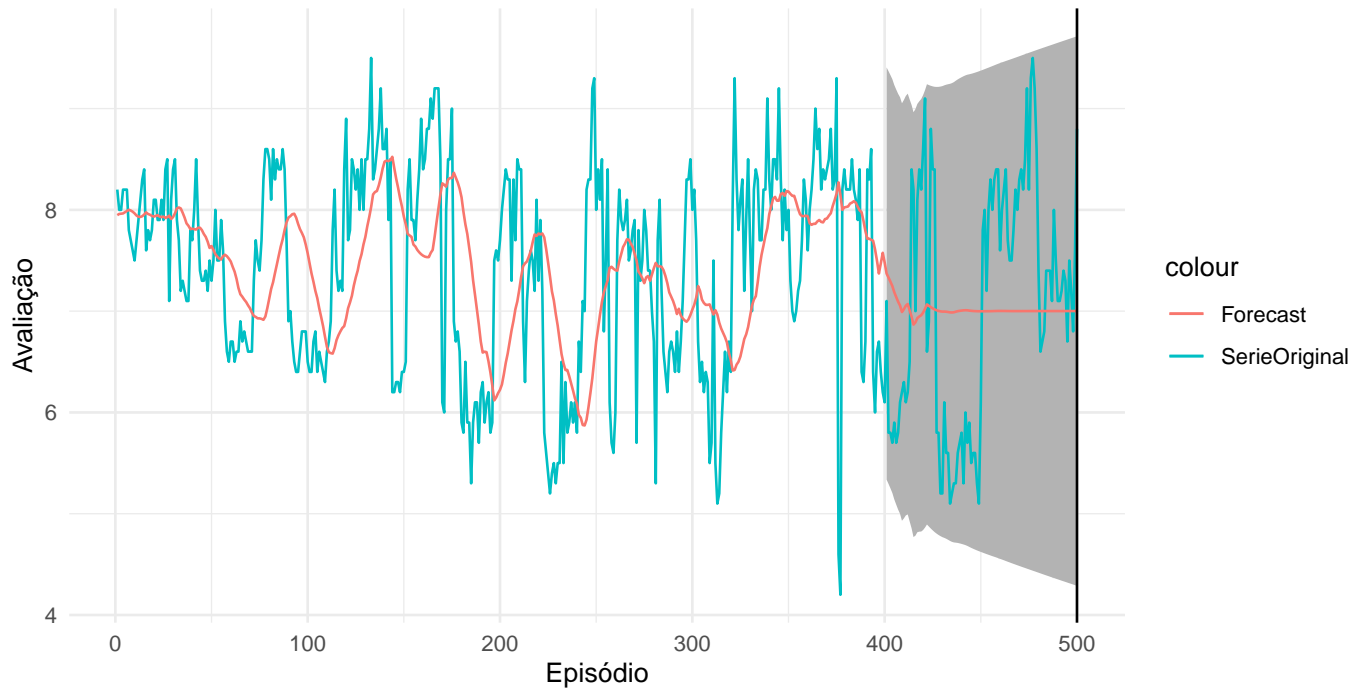
4.4.1 Médias Móveis Simples

Começaremos os modelos nessa seção com um dos elementos estatísticos mais citados nos últimos tempos: as médias móveis, que foram muito utilizadas para analisar o avanço da COVID-19 sendo constantemente noticiada, assim ganhando destaque. Aqui veremos se a média móvel simples também é um bom modelo para a avaliação de episódios de Naruto. Sabendo que as médias móveis é o cálculo de média aritmética das k observações mais recentes, nesse caso utilizaremos as 21 observações mais recentes pois é o número médio da temporada de Naruto Shippuden.



Assim nota-se que existe uma suavização dos picos de crescimento e de queda da avaliação dos episódios, não compreendendo toda a variação dos dados.

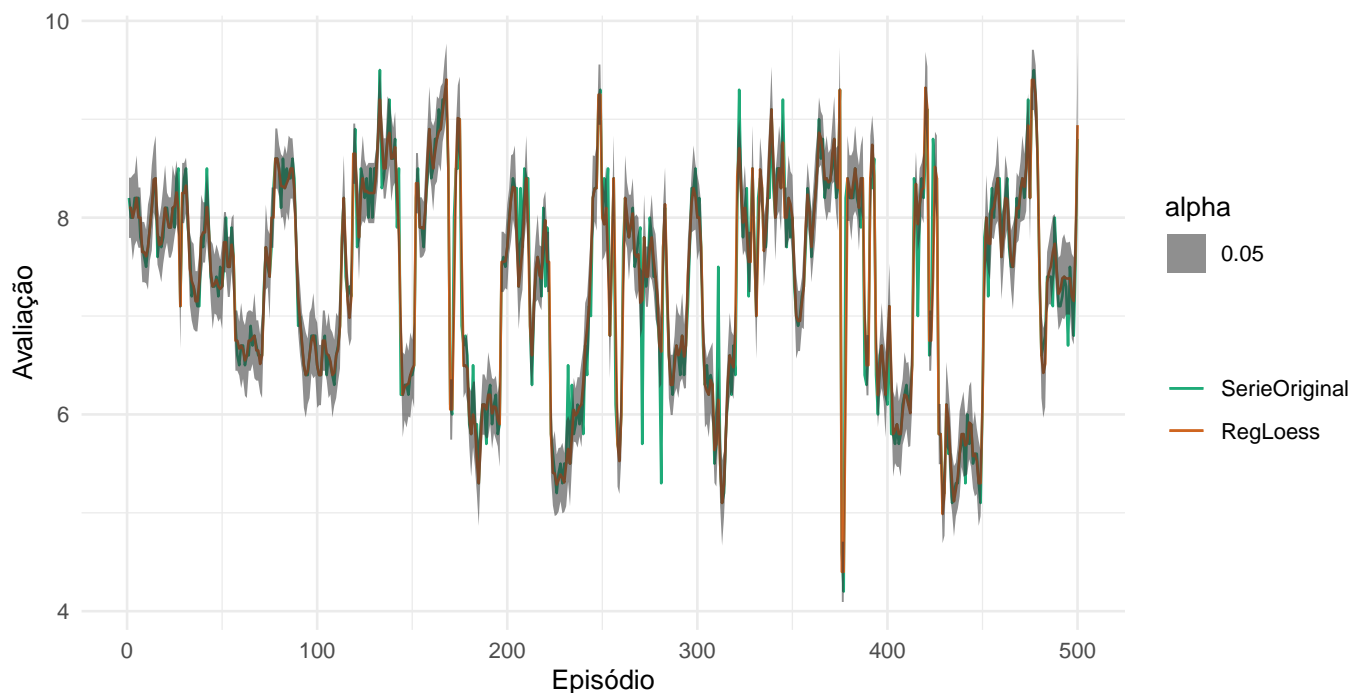
4.4.1.1 Previsão Não foi possível realizar a previsão com o pacote TTR, assim buscamos uma alternativa e resolvemos utilizar o pacote smooth para não ficarmos com a predição, então temos o gráfico abaixo que dá a previsão e os índices de confiança:



Percebe-se que o modelo de médias, acaba sobreajustando os dados, ou seja, é muito bom nos dados de treino, mas tende a uma reta e constante nos dados de teste, assim temos indícios de que não será um bom modelo.

4.4.2 Regressão LOESS

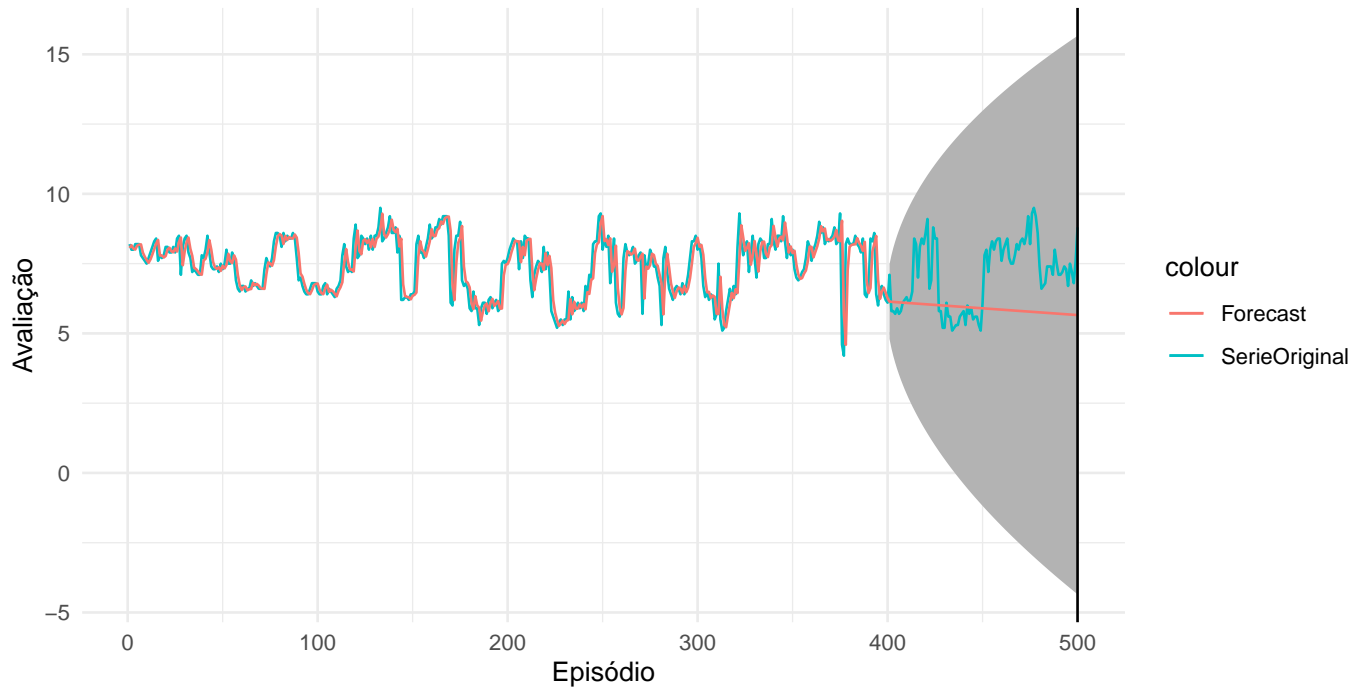
A Regressão LOESS, é um método não paramétricos para estimar curvas e superfícies com suavização nessa seção buscamos mostrar, uma Regressão LOESS um pouco diferente, utilizando o parâmetro de suavização buscando o menor SSE entre a predição da LOESS e a série verdadeira.



4.4.3 Modelos Holt

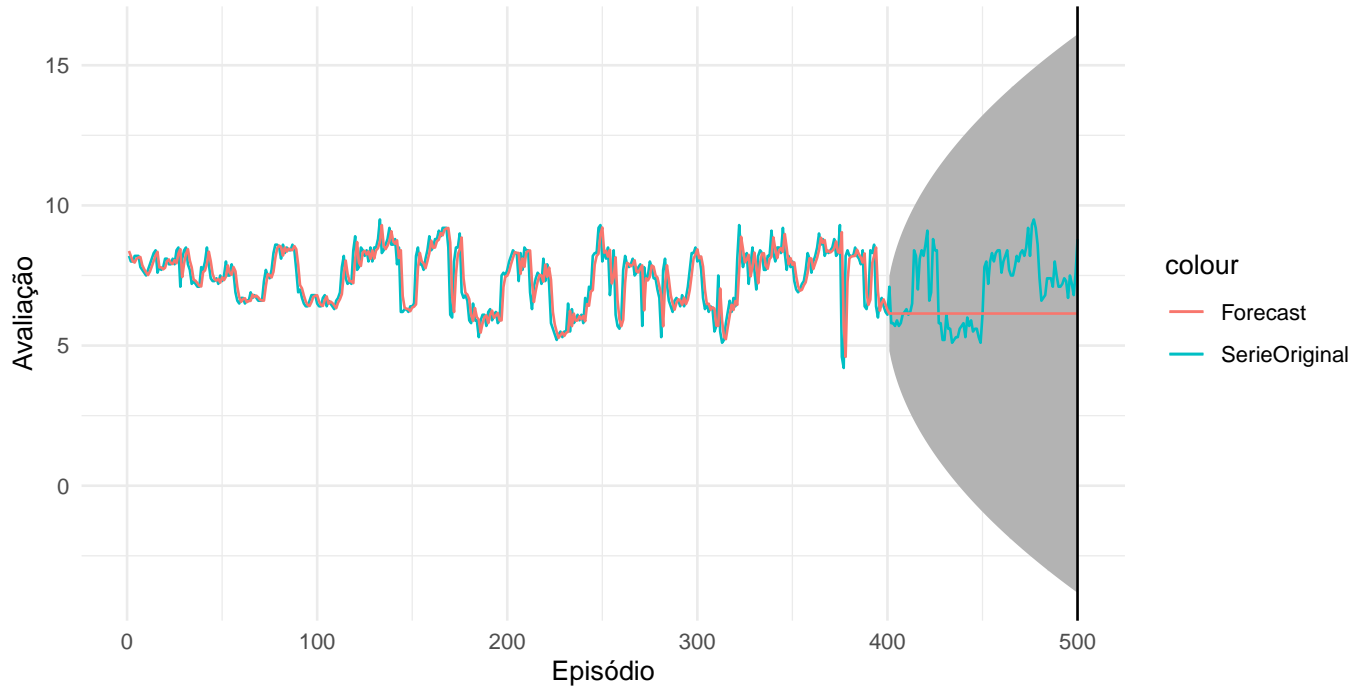
Buscaremos estudar e modelar mais uma técnica de Suavização Exponencial, os modelos de Holt, acabamos não conseguindo usar os modelos de Holt-Winters, pois tivemos problemas com a frequência da Série. Assim utilizaremos dois modelos de Holt Clássico com amortecimento na tendencia e sem amortecimento.

4.4.3.1 Holt não Amortecido Assim então o gráfico do modelo do Holt clássico não amortecido é:



Logo, podemos ver que ele se comporta bem em todos os dados de teste, no entanto nos dados de treino ele tende a uma reta.

4.4.3.2 Holt com Amortecimento Assim então o gráfico do modelo do Holt clássico com amortecimento é:



Observa-se que todos os modelos nessa seção, estão sobreajustados, todos tendendo a uma reta dos dados de treino.

4.4.4 Métricas

Utilizaremos as mesmas métricas utilizadas no dados de teste dos Modelos do Pacote ModelTime, nos dados de teste dos modelos propostos nessa seção:

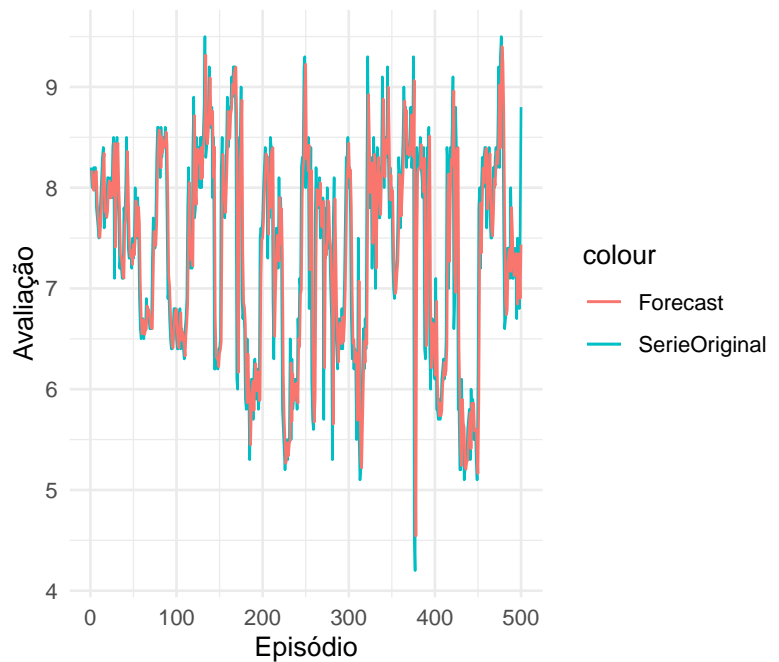
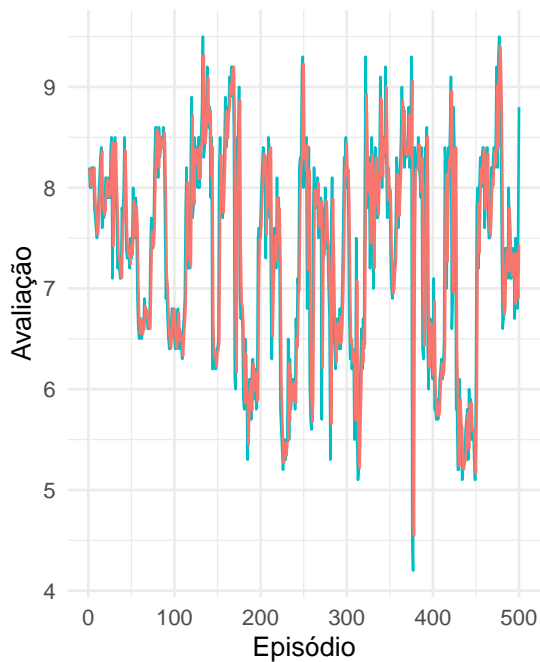
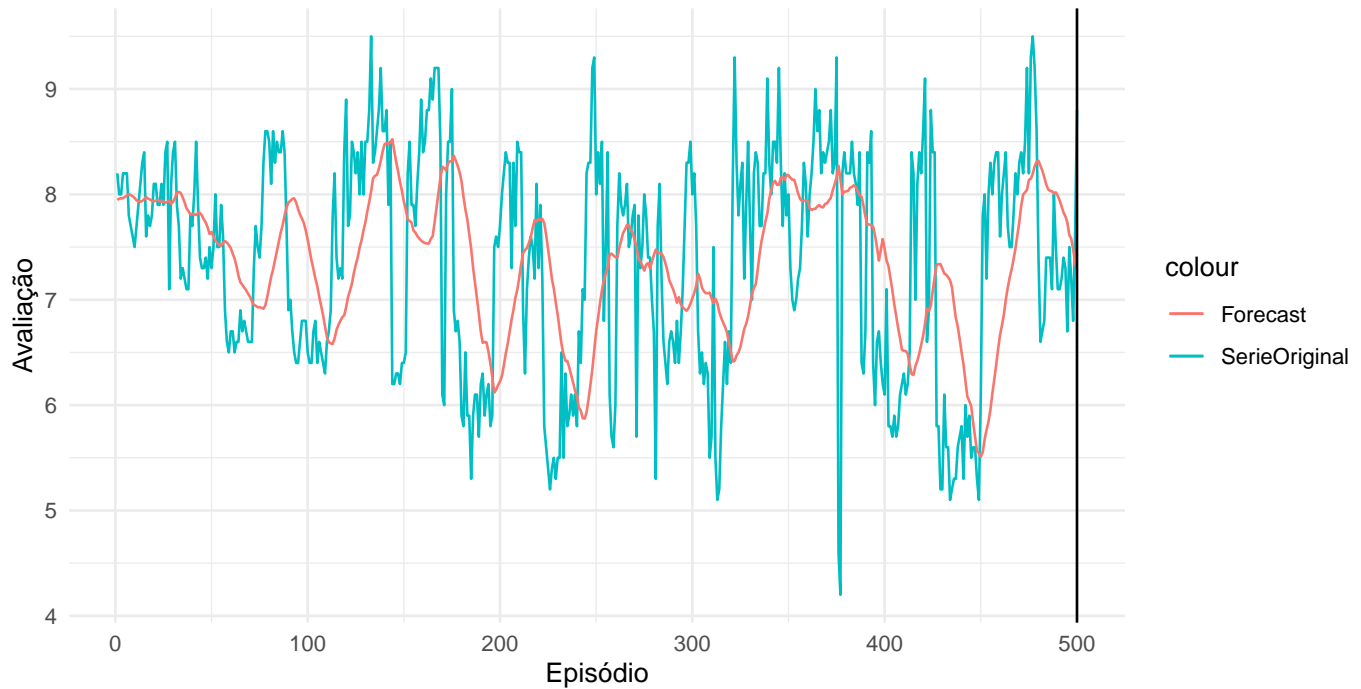
Tabela 9: Métricas nos dados de teste

	MAE	MAPE	MASE	SMAPE	RMSE	RSQ
SMA	1.08	15.9	2.36	15.5	1.23	0.060
Holt	1.44	18.6	3.12	21.0	1.74	0.154
Holt amortecido	1.28	16.7	2.78	18.4	1.53	0.062

Observa-se claramente que as métricas dos modelos desta seção são menos satisfatórios, que os modelos propostos anteriormente, com a métrica MAE não superando nem os piores modelos ETS's, assim como o RMSE e as outras métricas. Então confirmamos o que foi notado com os gráficos que modelos que tendem a reta nos dados de treino, não serão melhores que os modelos que captam melhor as variabilidades.

4.4.5 Reajustando os dados

Mesmo com as métricas dos outros modelos serem superiores iremos verificar se os modelos propostos, reajustados à todos os dados, passam nos testes da análise de resíduos.



Nota-se que os modelos completos se ajustam muito bem aos dados, os problemas desses modelos são para fazer previsão.

4.4.6 Análise de Resíduos

Igualmente aos modelos do pacote ModelTime, não significa que os modelos acima são inferencialmente adequadas, então nessa seção usaremos os mesmos testes definidos na seção de análise de resíduos anterior para verificação do modelo:

Tabela 10: Testes aplicados nos resíduos

Modelos	Ljung_box	Box_Pierce	Durbin_Watson	Shapiro_wilk
SMA	0.000	0.000	0.436	0.148
Holt	0.224	0.225	1.883	0.000
Holt amortecido	0.226	0.228	1.884	0.000

Assim, podemos ver que os modelos que não são significativos pelos testes de Shapiro-Wilk, pois os resíduos não possuem distribuição normal. Porém, nos outros testes tem um bom comportamento tendo os resíduos independentes de acordo com Ljung-Box e Box-Pierce, assim como não possuem correlação serial pelo teste de Durbin-Watson, o único modelo que é rejeitado em todos os testes é o Modelo de Médias Móveis.

5 Conclusão

Desse modo, percebemos que os dois modelos PROPHET XGBOOST que não foram selecionados são os melhores modelos inferenciais dentre os testados para essa série, pois estavam no top 4 considerando as métricas de avaliação do modelo e passaram por todos os testes da análise de resíduo. Outro ponto interessante, apesar de não ser um bom modelo de acordo com a análise de resíduo o modelo ETS(M,A,N) e ETSX(M,A,N) que foram os melhores modelos em métrica dos modelos de Suavização Exponencial, confirmam o que os testes de sazonalidade afirmaram, o ‘N’ indica que é um modelo de suavização exponencial sem sazonalidade, mas mesmo assim em métricas não chegaram próximos de nenhum dos modelos PROPHET. Outro ponto interessante da análise de resíduo, são os modelos THETA que passaram com todas as receitas e assim como os modelos ETS(A,M,A) e ETS(A,N,M) são modelos que tem métricas ruins comparadas aos outros.

Para os modelos não-Modeltime, o modelo SMA, se destacou entre os outros nas métricas, mas não passou por nenhum teste da análise de resíduos, assim nenhum dos modelos mais básicos se candidatou a melhor modelo.

6 Referências

- [1] Morettin PA, Toloi CM. Análise de séries temporais: Modelos lineares univariados. Editora Blucher; 2018.
- [2] Bartolomé K, Zambrano R. Karina bartolome: Workflowsets in time series [Internet]. 2021. Available from: <https://karbartolome-blog.netlify.app/posts/workflowsets-timeseries/>.
- [3] R Core Team. R: A language and environment for statistical computing [Internet]. Vienna, Austria: R Foundation for Statistical Computing; 2021. Available from: <https://www.R-project.org/>.
- [4] Wickham H, Averick M, Bryan J, et al. Welcome to the tidyverse. Journal of Open Source Software [Internet]. 2019;4:1686. Available from: <https://doi.org/10.21105/joss.01686>.
- [5] Taylor SJ, Letham B. Forecasting at scale. The American Statistician. 2018;72:37–45.
- [6] Durbin J, Watson GS. Testing for serial correlation in least squares regression. III. Biometrika [Internet]. 1971;58:1–19. Available from: <http://www.jstor.org/stable/2334313>.
- [7] Racine J, Hyndman R. Using r to teach econometrics. Journal of Applied Econometrics [Internet]. 2002;17:175–189. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/jae.657>.

Apêndice - Código R

Para acessar o código desse pdf, [clique aqui](#). Para acessar o código da modelagem, [clique aqui](#).