

O GRANDE TÍTULO

Alisson Rosa e Vítor Pereira

Resumo

One Piece > Naruto

Sumário

| | | |
|----------|---|-----------|
| 1 | Introdução | 1 |
| 2 | Informações | 1 |
| 3 | Testes | 5 |
| 3.1 | Tendência | 5 |
| 3.2 | Sazonalidade | 5 |
| 3.3 | Raiz unitária | 6 |
| 3.4 | Resultados | 6 |
| 4 | Modelagem | 6 |
| 4.1 | Dados de treino e teste | 6 |
| 4.2 | Recipes | 7 |
| 4.3 | Utilizando o Pacote Modeltime | 7 |
| 4.4 | Utilizando outros Pacotes | 16 |
| | Referências | 19 |

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.

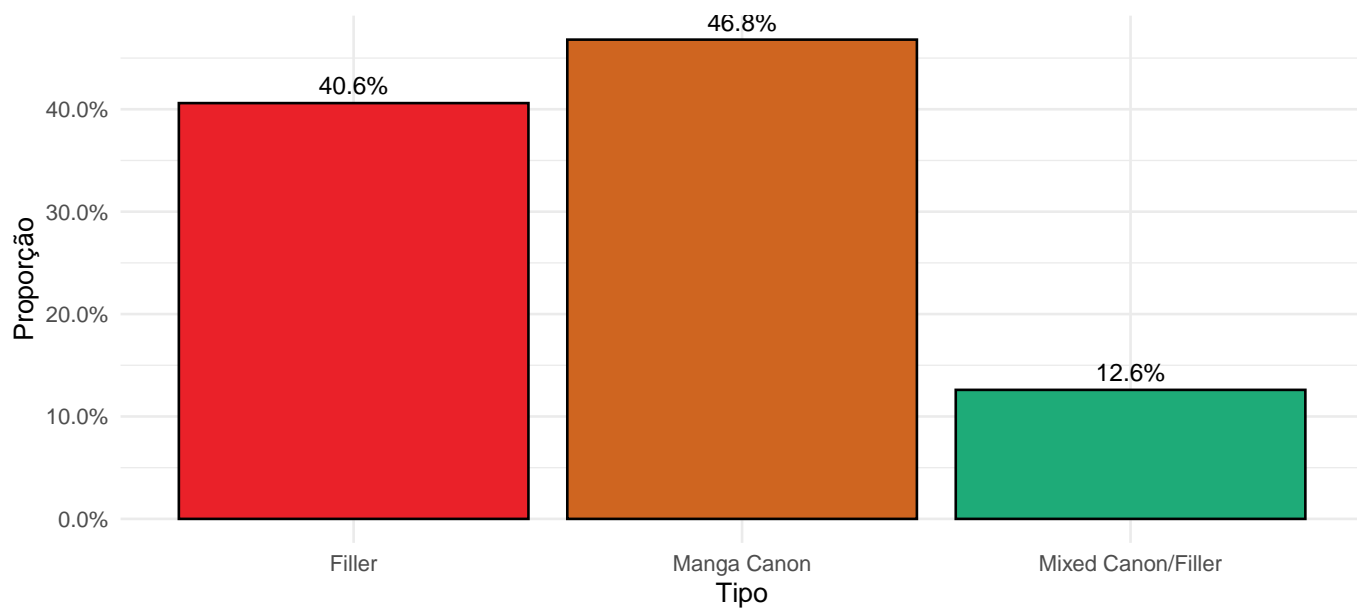
A história é dividida em duas partes, a primeira parte se passa nos anos da pré-adolescência de Naruto (clássico), e a segunda parte se passa em sua adolescência (Shippuden). Nesse trabalho trataremos sobre o Naruto Shippuden, desenvolvendo-o em uma perspectiva de séries temporais, vamos adotar como 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



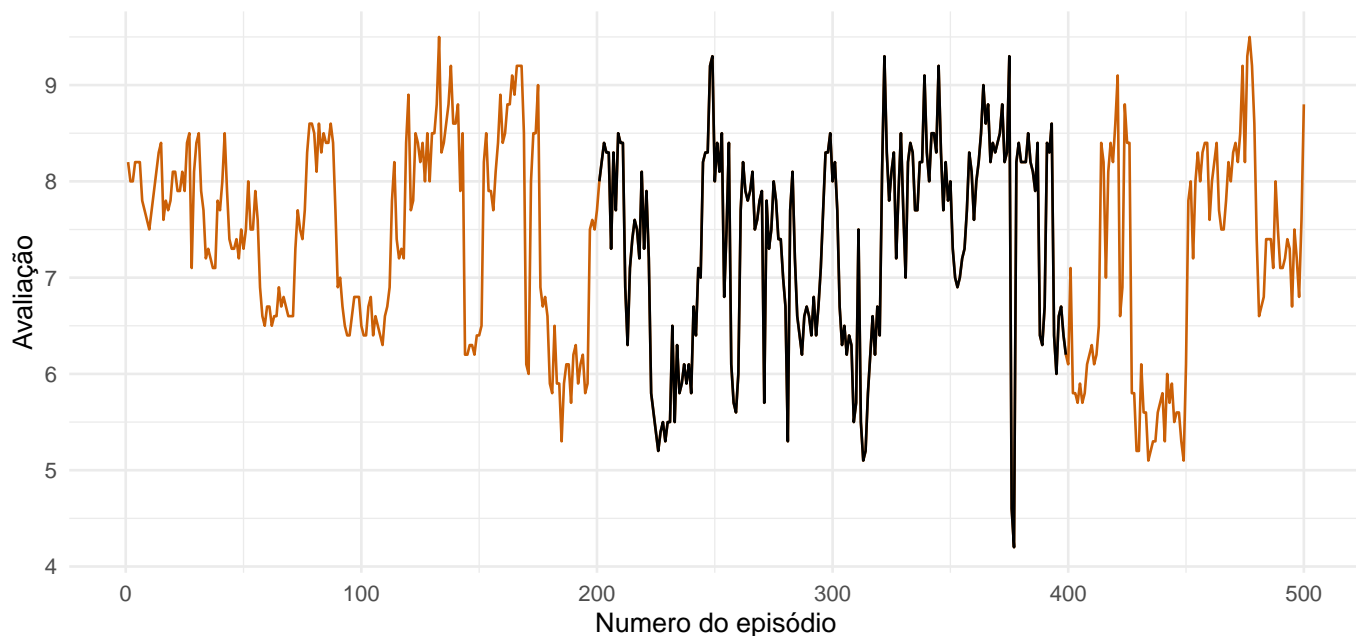
Fonte: Elaborado pelos autores

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

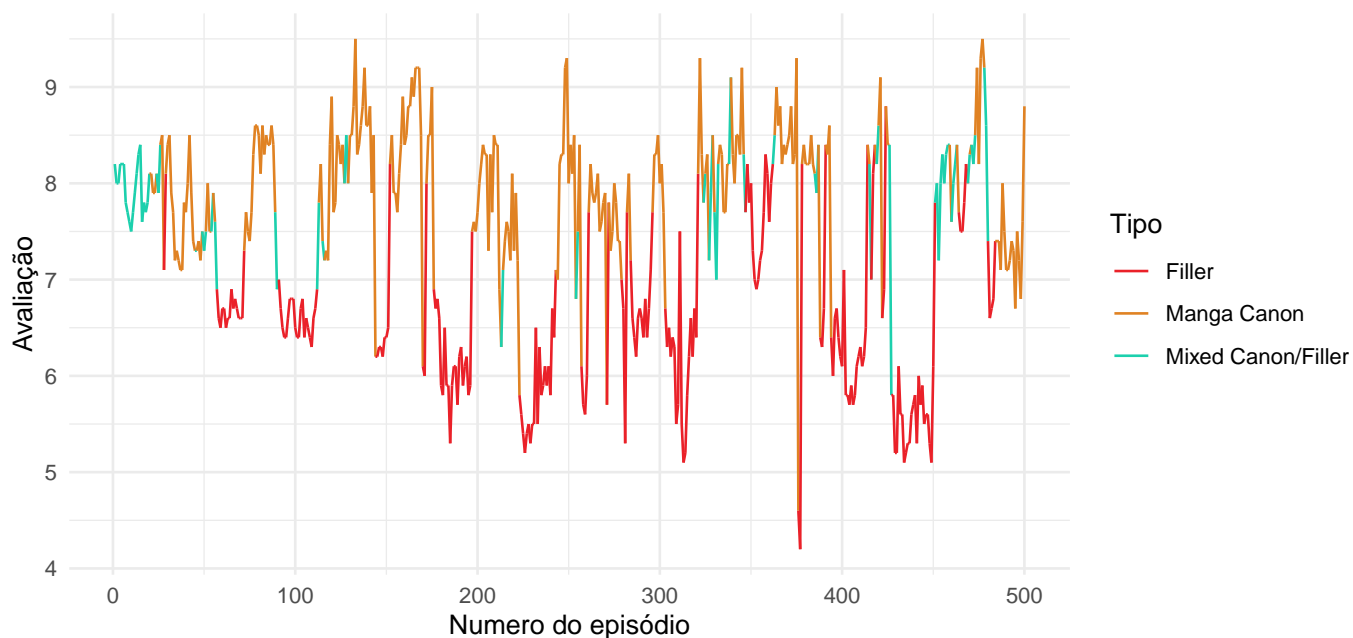
| Tipo | Quantidade |
|--------------------|------------|
| Filler | 203 |
| Manga Canon | 234 |
| Mixed Canon/Filler | 63 |

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.



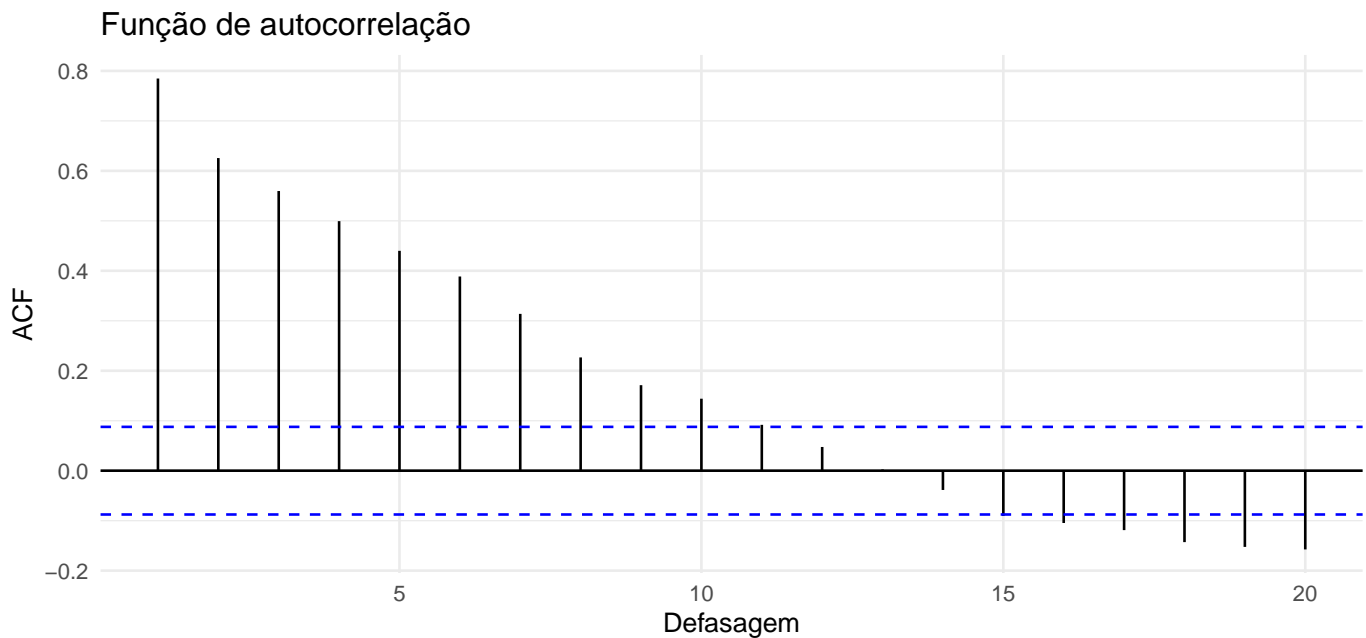
Há bastante coisa a se notar nesse gráfico, primeiramente pelos episódios 200 e 400, existe um indício de alteração na média das variáveis aleatórias, portanto furando o pressuposto de média constante ao longo do tempo, perto dos episódios finais da saga nota-se também inúmeras quedas bruscas na avaliação 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.

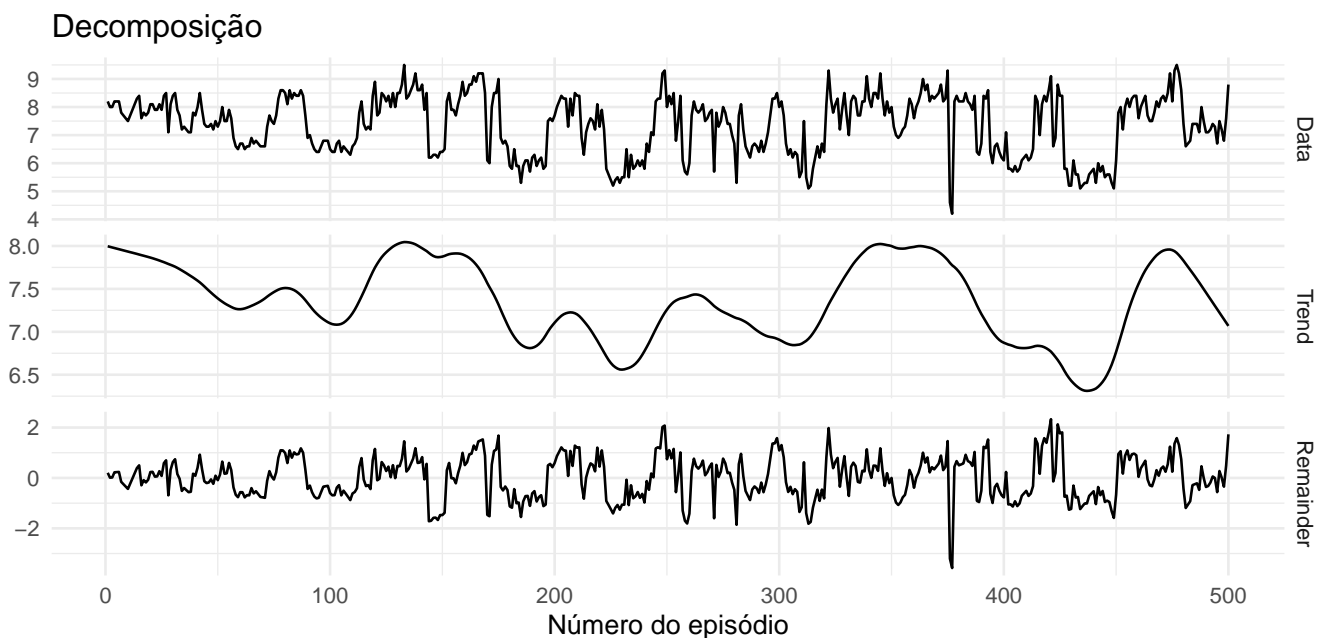
É necessário também avaliar a correlação nas avaliações:



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:

$$H_0 : \text{A série não possui tendência determinística}$$

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) | 0.1 |
| 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.

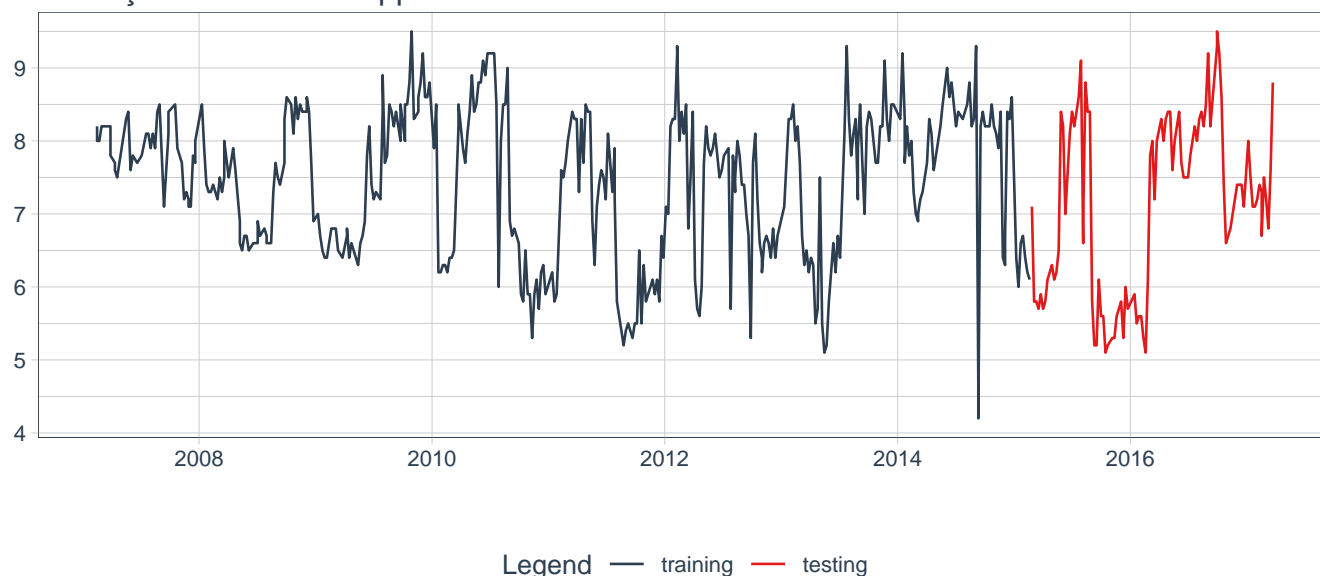
4 Modelagem

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. 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 .

Já no ajuste de modelos podemos perceber que a suavização exponencial não pode utilizar de algumas combinações de erro, tendência 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 tunnar hiperparâmetros considerando duas características, sazonalidade, poderia aditiva ou multiplicativa, crescimento, poderia linear ou logístico, assim como utilizamos duas engines para tunnar 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

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 | .model_desc | 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 | 0.651 | 9.42 | 1.42 | 9.29 | 0.804 | 0.640 |
| 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 | ETX(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 | ETX(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 | ETX(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 | ETX(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 |

Assim temos também que as métricas no dados de treino são:

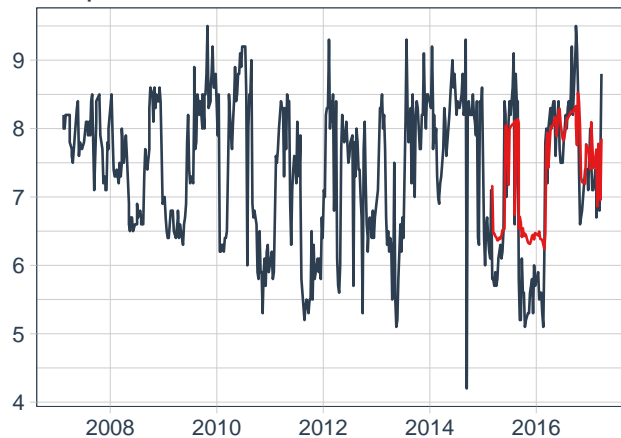
Tabela 7: Métricas nos dados de treino

| Modelo | MAE | MAPE | MASE | SMAPE | RMSE | RSQ |
|-------------------------|-------|-------|-------|-------|-------|-------|
| PROPHET CATBOOST | 0.392 | 5.49 | 0.898 | 5.38 | 0.547 | 0.709 |
| PROPHET XGBOOST TUNNADO | 0.500 | 7.17 | 1.147 | 6.92 | 0.649 | 0.625 |
| PROPHET XGBOOST | 0.193 | 2.64 | 0.442 | 2.62 | 0.270 | 0.932 |
| PROPHET XGBOOST DEFAULT | 0.203 | 2.79 | 0.465 | 2.77 | 0.277 | 0.929 |
| ETSX(MAN) | 0.457 | 6.48 | 1.048 | 6.29 | 0.674 | 0.570 |
| ETS(M,A,N) | 0.467 | 6.62 | 1.071 | 6.42 | 0.688 | 0.552 |
| ETS(M,A,N) | 0.467 | 6.62 | 1.071 | 6.42 | 0.688 | 0.552 |
| ETS(M,A,N) | 0.467 | 6.62 | 1.071 | 6.42 | 0.688 | 0.552 |
| ETS(M,A,N) | 0.467 | 6.62 | 1.071 | 6.42 | 0.688 | 0.552 |
| ETS(M,A,N) | 0.467 | 6.62 | 1.071 | 6.42 | 0.688 | 0.552 |
| ETS(M,A,N) | 0.467 | 6.62 | 1.071 | 6.42 | 0.688 | 0.552 |
| ETSX(MAN) | 0.464 | 6.58 | 1.065 | 6.38 | 0.685 | 0.556 |
| ETS(MAN) | 0.464 | 6.58 | 1.065 | 6.38 | 0.686 | 0.555 |
| ETSX(MAA) | 0.463 | 6.55 | 1.063 | 6.37 | 0.678 | 0.564 |
| ETS(MAA) | 0.463 | 6.55 | 1.063 | 6.36 | 0.679 | 0.563 |
| PROPHET XGBOOST DEFAULT | 0.426 | 6.02 | 0.978 | 5.89 | 0.577 | 0.698 |
| PROPHET XGBOOST DEFAULT | 0.388 | 5.49 | 0.891 | 5.37 | 0.541 | 0.720 |
| PROPHET CATBOOST | 0.734 | 10.44 | 1.685 | 10.11 | 0.889 | 0.285 |
| ETSX(MAA) | 0.457 | 6.47 | 1.048 | 6.28 | 0.668 | 0.579 |
| PROPHET CATBOOST | 0.664 | 9.46 | 1.525 | 9.15 | 0.821 | 0.406 |

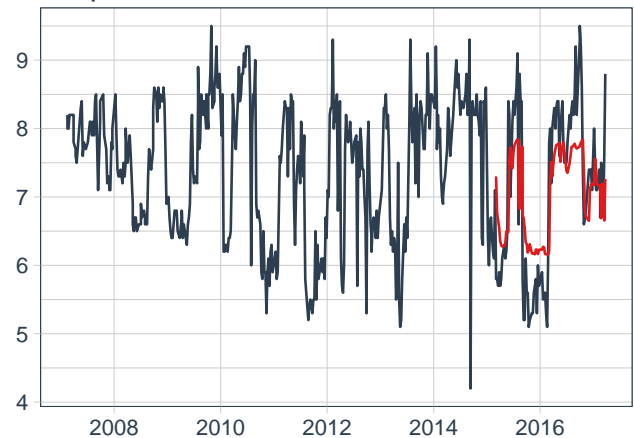
4.3.3.1 Gráficos Nesta seção será mostrado alguns gráficos interessantes dos seguintes modelos: Prophet's, ET SX(MAN), ETS(MAN) do smooth, ETS(MAN) do forecast e ETS(MAA) do smooth.

4.3.3.1.1 Previsões

Prophet com Catboost

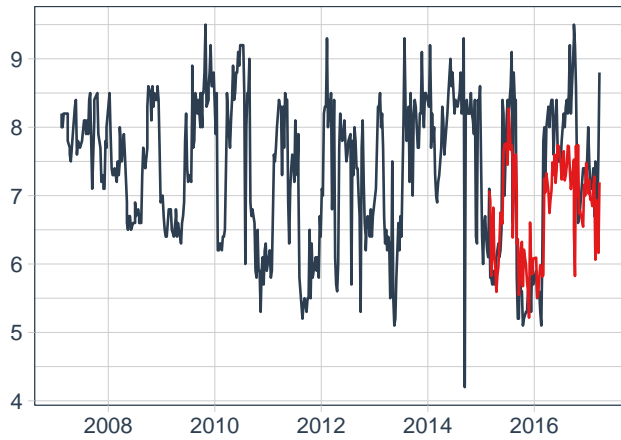


Prophet com XGboost tunnado

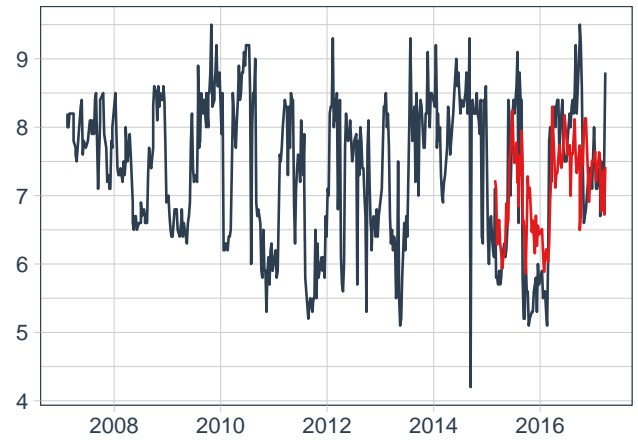


Legend — ACTUAL — 1_PROPHET MODEL W/ CATBOOST ERROR Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERROR

Prophet com XGboost

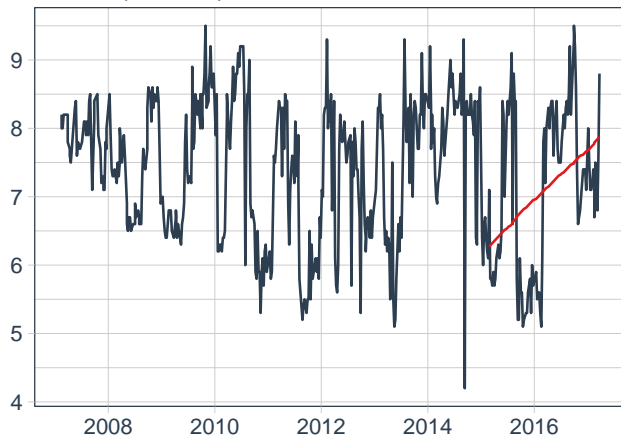


Prophet com XGboost default



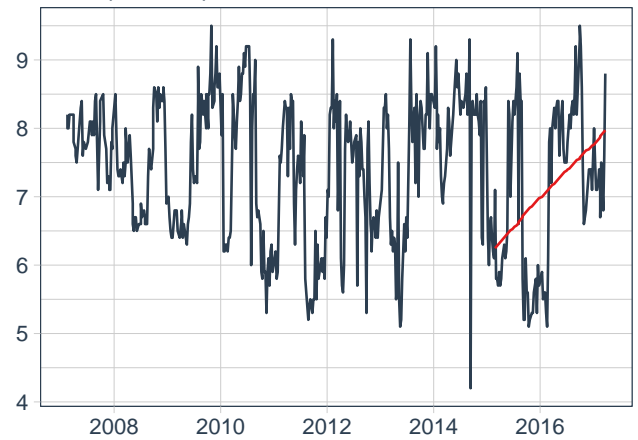
Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERROR

ETSX(M,A,N)



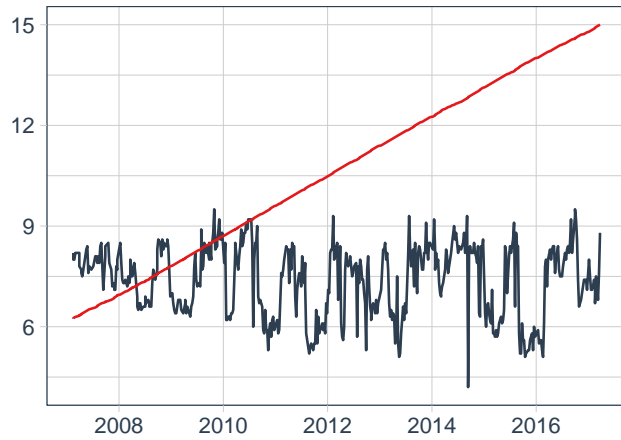
Legend — ACTUAL — 1_ETSX(MAN)

ETS(M,A,N)



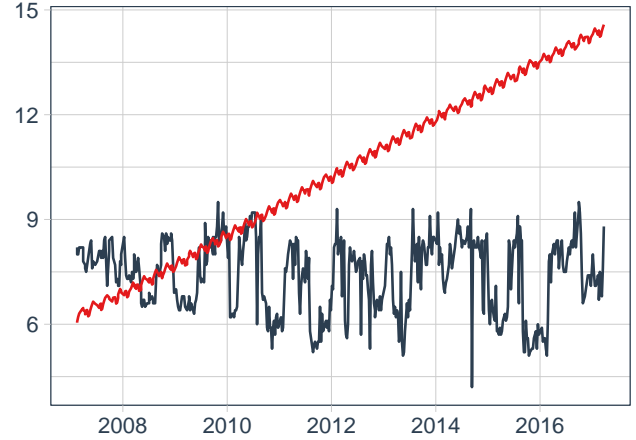
Legend — ACTUAL — 1_ETS(M,A,N)

ETS(M,A,N) do pacote smooth



Legend — ACTUAL — 1_ETS(MAN)

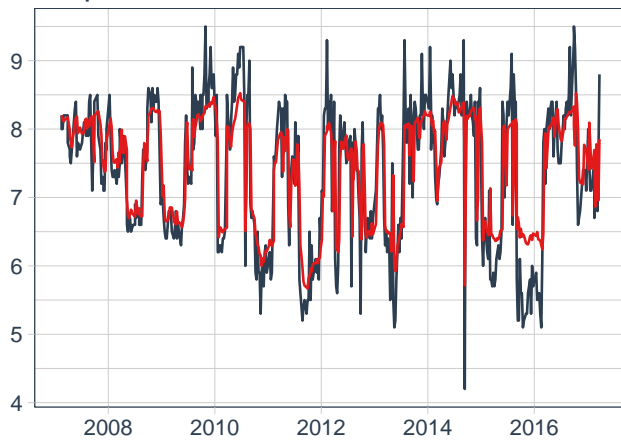
ETSX(M,A,A)



Legend — ACTUAL — 1_ETS(MAA)

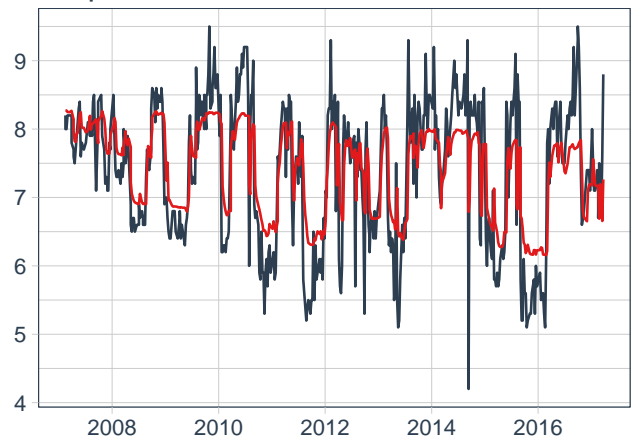
4.3.3.1.2 Prophet Predição e Previsão

Prophet com Catboost



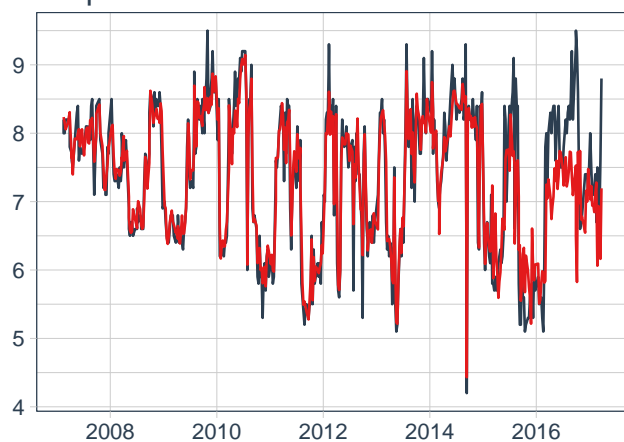
Legend — ACTUAL — 1_PROPHET MODEL W/ CATBOOST ERROR

Prophet com XGboost tunado

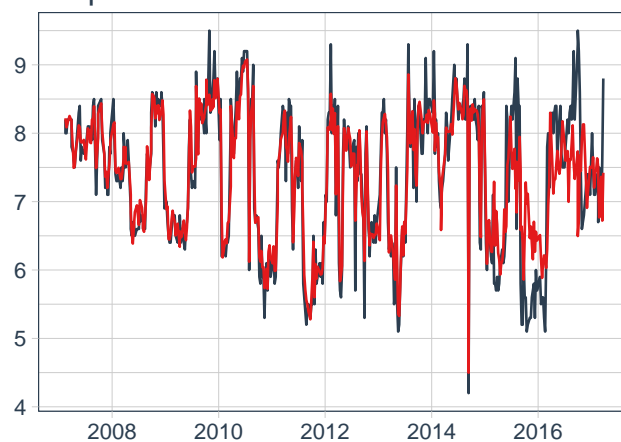


Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERROR

Prophet com XGboost



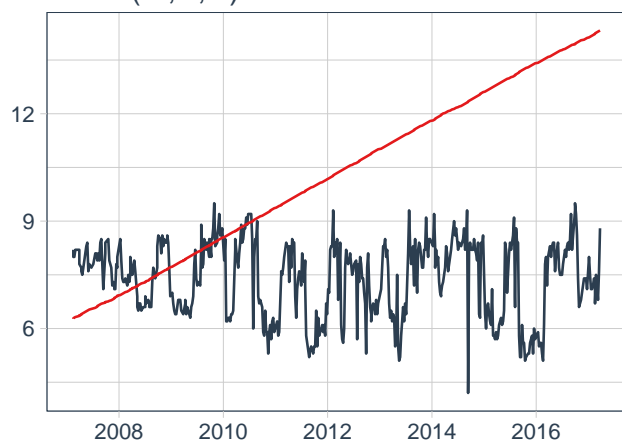
Prophet com XGboost default



Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERRC Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERROF

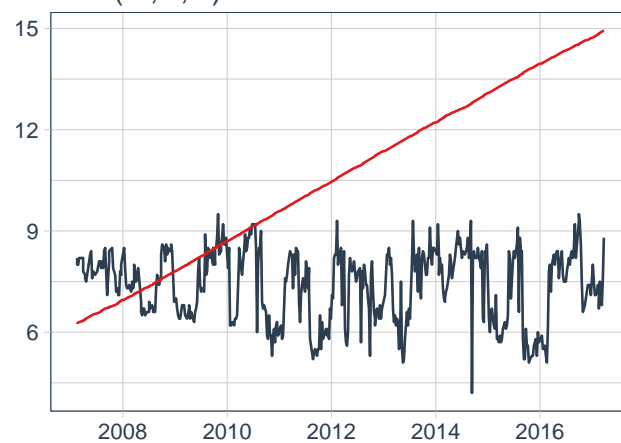
4.3.3.1.3 Suavização Exponencial Predição e Previsão

ETSX(M,A,N)



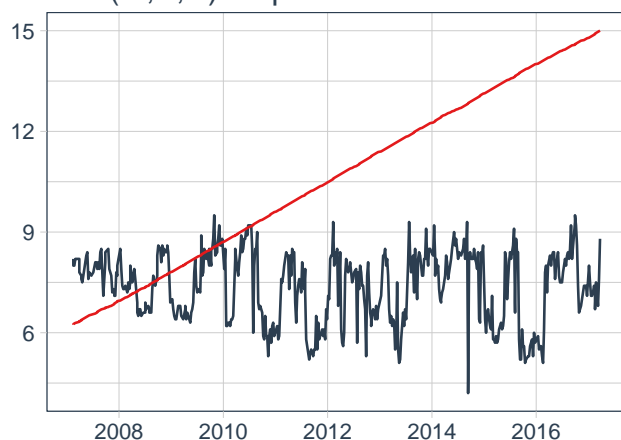
Legend — ACTUAL — 1_ETSX(MAN)

ETS(M,A,N)



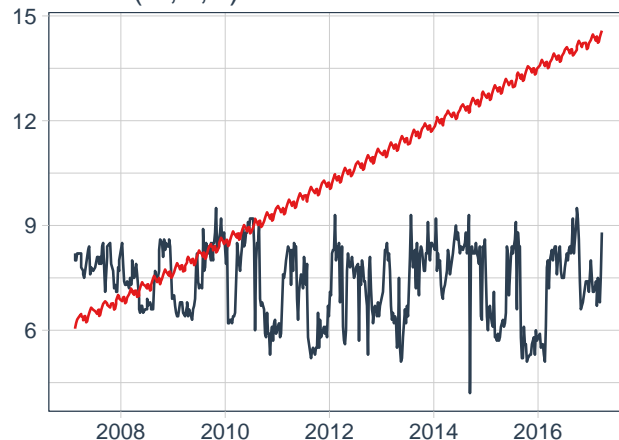
Legend — ACTUAL — 1_ETS(M,A,N)

ETS(M,A,N) do pacote smooth



Legend — ACTUAL — 1_ETS(MAN)

ET SX(M,A,A)



Legend — ACTUAL — 1_ETS(MAA)

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 8: 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

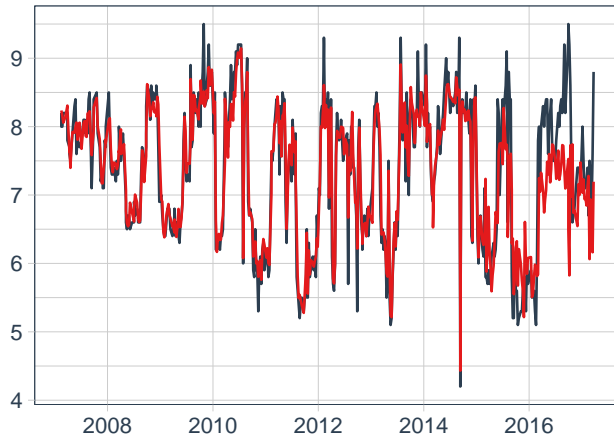
Tabela 9: Modelos que passaram nos teste de resíduos

| .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 nossos 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.

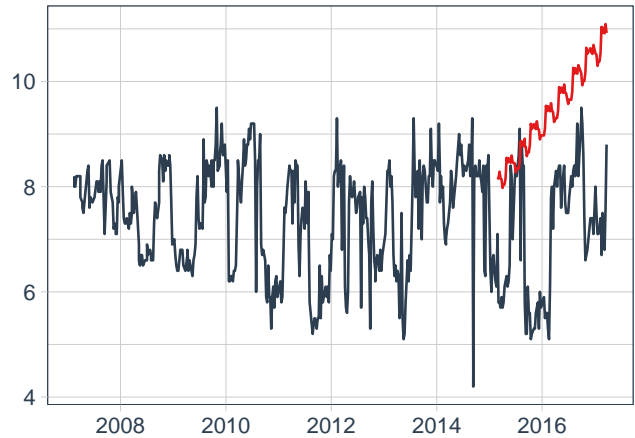
4.3.5.1 Gráficos

Prophet com XGboost reajustada



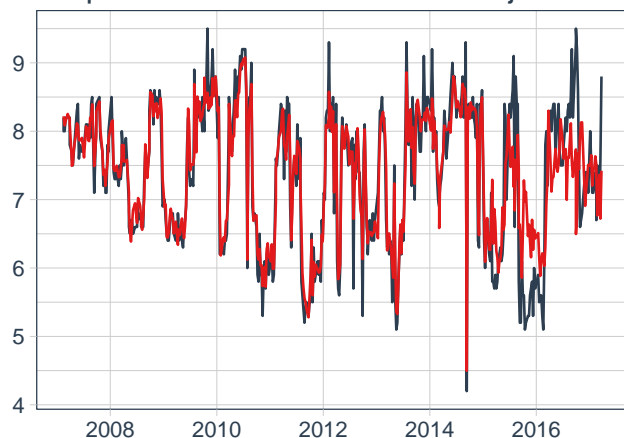
Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERROR

Prophet com XGboost reajustada nos dado



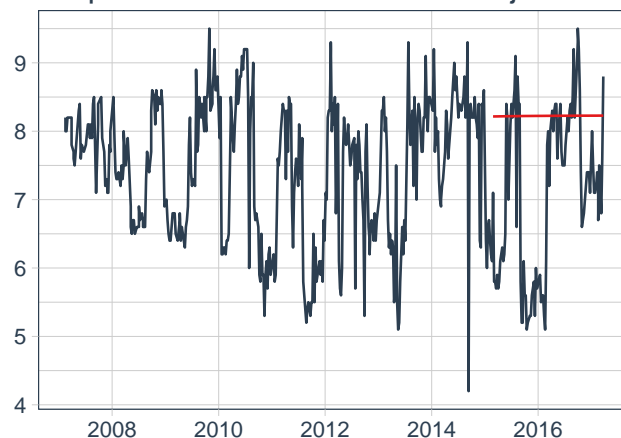
Legend — ACTUAL — 1_ETSM(M,M,M)

Prophet com XGboost default reajustada



Legend — ACTUAL — 1_PROPHET W/ XGBOOST ERRC

Prophet com XGboost default reajustada no



Legend — ACTUAL — 1_ETS(M,MD,N)

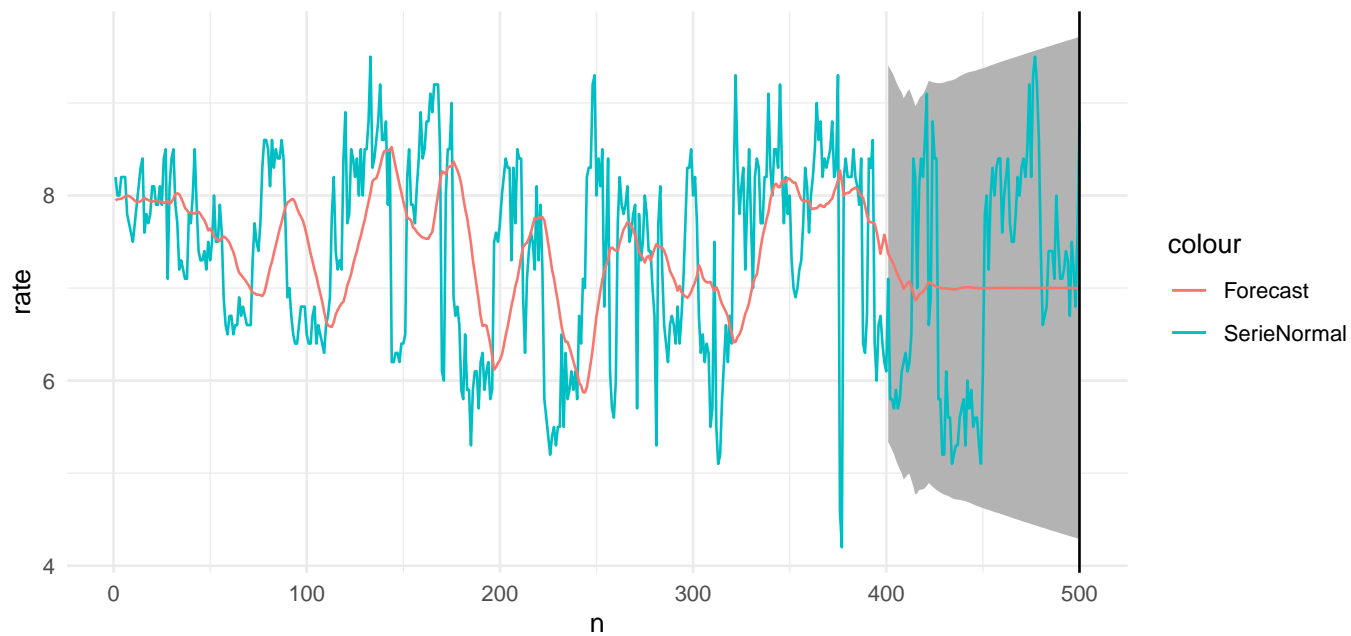
4.4 Utilizando outros Pacotes

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.

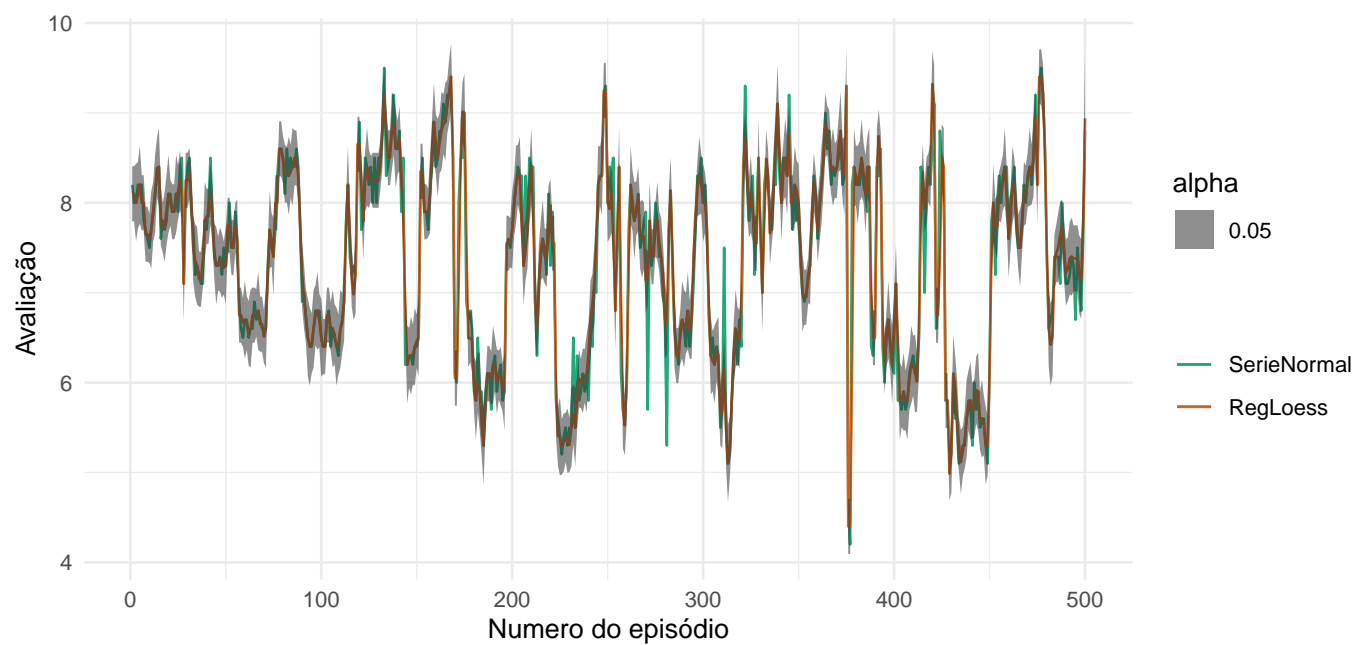
4.4.1 Médias Móveis Simples



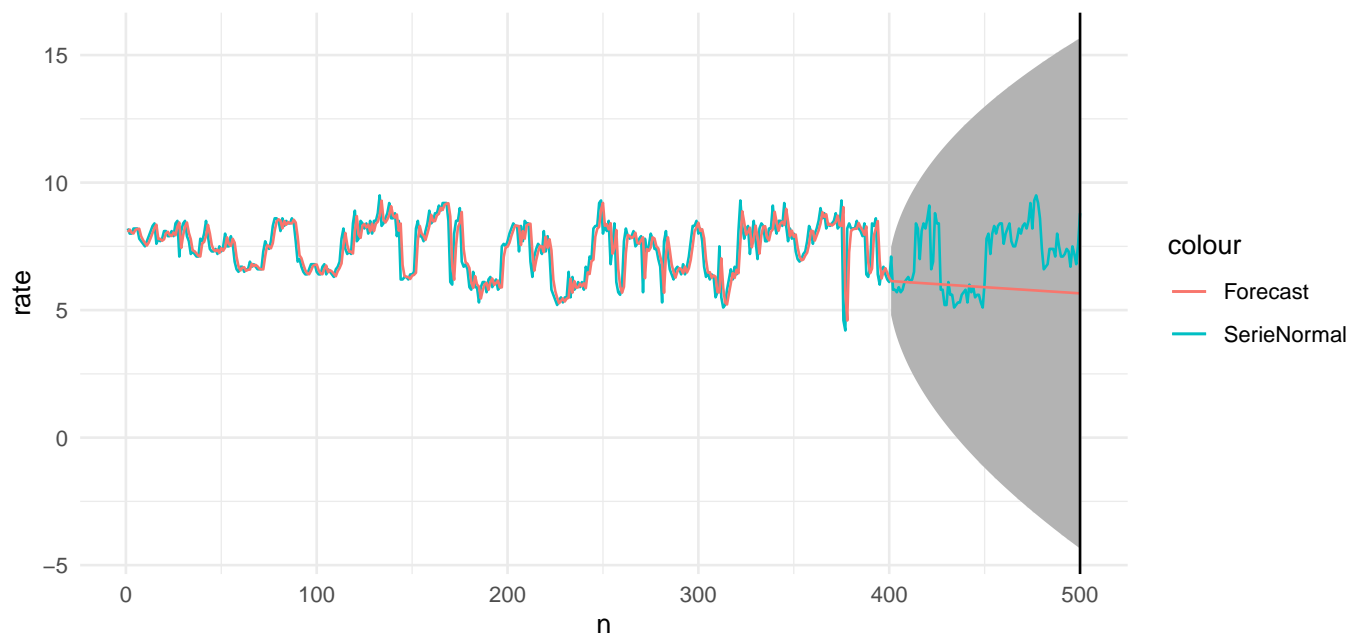
4.4.1.1 Previsão Utilizando o pacote smooth, temos a seguinte previsão



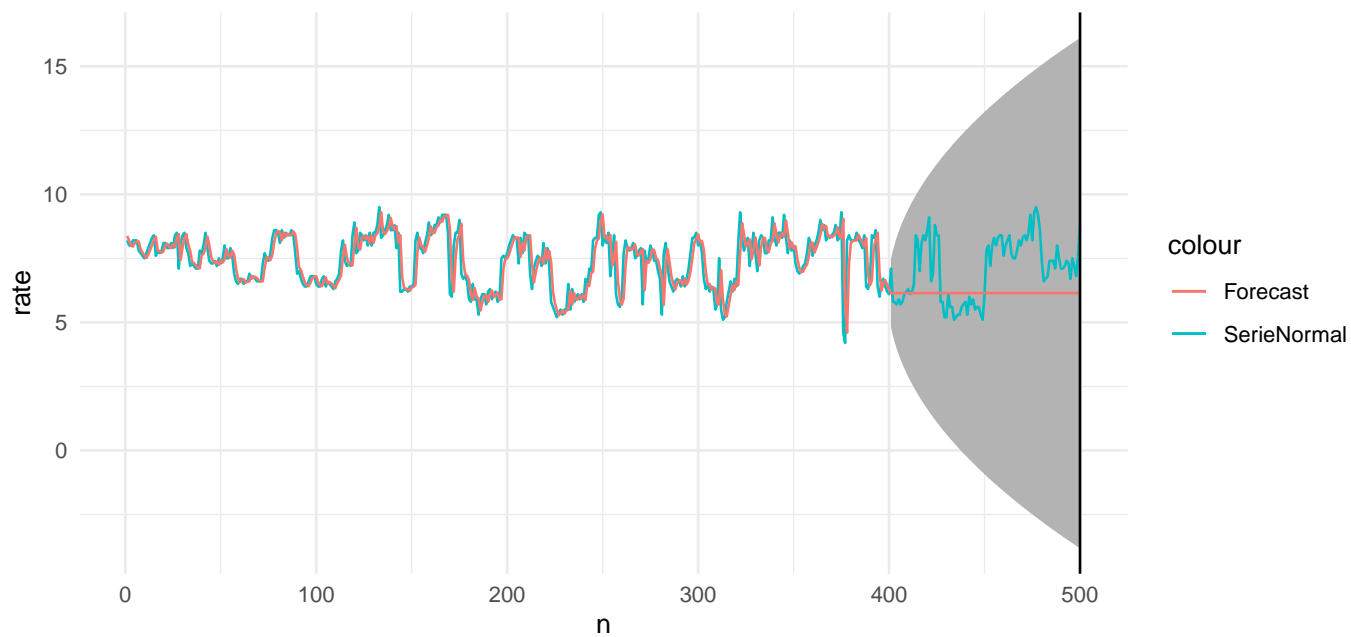
4.4.2 Regressão LOESS



4.4.3 Holt não Amortecido



4.4.4 Holt com Amortecimento



4.4.5 Calculando Métricas de Acurácia dos Modelos não-ModelTime

Tabela 10: 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 |

[1] 23.8

Referências

- [1] Morettin PA, Toloi CM. Análise de séries temporais: Modelos lineares univariados. Editora Blucher; 2018.