



UNIVERSIDADE ESTADUAL DE CAMPINAS  
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO CIENTÍFICA  
DEPARTAMENTO DE MATEMÁTICA APLICADA



PEDRO FRANCISCO GODOY BERNARDINELLI

## **Aprendizado profundo na previsão de séries temporais: uma análise comparativa**

Campinas  
18/06/2023

PEDRO FRANCISCO GODOY BERNARDINELLI

**Aprendizado profundo na previsão de séries temporais: uma  
análise comparativa**

Monografia apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos para obtenção de créditos na disciplina Projeto Supervisionado, sob a orientação do(a) Prof. João Batista Florindo.

## Resumo

O projeto analisou o desempenho de redes neurais profundas no problema de previsão de séries temporais. Mais precisamente, foram empregados um modelo denso totalmente conectado e um modelo do estado-da-arte na área (N-BEATS). Para tal, foram utilizadas duas séries temporais com características contrastantes e, para cada uma delas, foram realizados três experimentos com configurações diferentes para os modelos. Os resultados obtidos foram comparados para obter uma variedade de conclusões a respeito das séries e dos modelos: o desempenho dos modelos em uma série temporal mais bem comportada é superior à performance em uma série volátil; o impacto da escolha dos hiperparâmetros nos resultados é significativo; a variação da série temporal e da configuração causa uma grande variabilidade nos resultados, com a alternância de qual modelo teve melhor desempenho em cada caso; o aumento do horizonte de previsão dificulta o problema, com resultados piores nessas circunstâncias e o impacto do aumento da quantidade de dados fornecidos aos modelos varia conforme a série temporal, não sendo possível apresentar uma resposta definitiva sobre isso.

## Abstract

The project analyzed the performance of deep neural networks on the time series forecasting problem. More precisely, a fully connected dense model and a state-of-the-art model in the area, N-BEATS, were employed. For such, two time series with opposing characteristics were used and, for each one, three experiments with different settings were made for each model. The results were compared to reach a variety of conclusions regarding the time series and the models: the performance of the models is superior on a better behaved time series if compared with volatile series; the choice of hyperparameters causes a significant impact on the results; the variation of the time series and of the experimental configuration causes a great variability in the results, changing which model performs better; the increase of the forecasting horizon complicates the problem, with worse results under these circumstances and the impact of increasing the amount of data provided to the models varies according to the time series, not allowing a definitive answer to this question.

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>6</b>
<b>2</b>	<b>Materiais e modelos</b>	<b>6</b>
2.1	Séries temporais . . . . .	7
2.1.1	Série 1: preço do Bitcoin . . . . .	7
2.1.2	Série 2: produção de energia elétrica . . . . .	8
2.2	Modelos . . . . .	11
2.2.1	Modelo 1: Denso . . . . .	11
2.2.2	Modelo 2: N-BEATS . . . . .	12
2.3	Métricas de avaliação . . . . .	13
<b>3</b>	<b>Experimentos</b>	<b>15</b>
<b>4</b>	<b>Resultados</b>	<b>16</b>
<b>5</b>	<b>Conclusão</b>	<b>20</b>

# 1 Introdução

O presente projeto tem como objetivo estudar a previsão de séries temporais utilizando técnicas de aprendizado profundo (*Deep Learning*). A previsão de séries temporais é um problema cada vez mais fundamental na vida contemporânea, estando presente nas mais diversas áreas para as mais diversas finalidades [2, 5].

Um dos principais objetivos de ter uma previsão (acurada) de séries temporais está na melhoria na tomada de decisões. Imagine, por exemplo, um investidor na bolsa de valores: com uma previsão acurada do preço da ação de certa empresa é possível tomar decisões sobre comprar ou vender ações de modo muito mais seguro e com um embasamento melhor. Já para o governo, um modelo capaz de prever o consumo de energia elétrica nos próximos meses permitiria identificar com certa antecipação se haverá problemas de insuficiência energética nos próximos meses e, assim, seria possível tomar decisões que amenizem ou até mesmo evitem essa situação. Por fim, a previsão acurada de séries temporais também é de interesse cotidiano para uma grande porcentagem de pessoas, afinal de contas, seria ótimo saber com antecedência se choverá ou não durante a viagem para a praia daqui a 2 meses. Tal fenômeno deve ser intensificado em um futuro próximo, já que há uma quantidade crescente de dados de todos os tipos e formatos disponíveis pelo mundo. Destarte, há cada vez mais séries temporais para mais aplicações.

Portanto, é fundamental conhecer os modelos disponíveis e como eles funcionam para saber qual escolher dependendo do problema. Por isso, neste projeto, serão analisados dois modelos de *machine learning* populares para este fim, um do estado-da-arte (N-BEATS) e outro já clássico (modelo denso). Será realizada uma comparação entre eles utilizando diversas métricas de avaliação de modelos. Esses modelos foram implementados seguindo os códigos disponíveis no GitHub [1] \*.

## 2 Materiais e modelos

Primeiramente, serão apresentados os materiais (as séries temporais que serão utilizadas para treinar e para testar os modelos) e após a apresentação das séries, serão

---

\*Link do GitHub citado: [https://github.com/mrdbourke/tensorflow-deep-learning/blob/main/10\\_time\\_series\\_forecasting\\_in\\_tensorflow.ipynb](https://github.com/mrdbourke/tensorflow-deep-learning/blob/main/10_time_series_forecasting_in_tensorflow.ipynb)

introduzidos os modelos que serão treinados com tais séries.

## 2.1 Séries temporais

### 2.1.1 Série 1: preço do Bitcoin

A primeira série que será utilizada é uma que consiste nos preços de fechamento do Bitcoin diários (em dólares), indo de outubro de 2013 até maio de 2021. Esse conjunto de dados pode ser encontrado no GitHub citado anteriormente [1]. A melhor maneira de visualizar os dados é por meio de um gráfico com todos os pontos que serão usados:

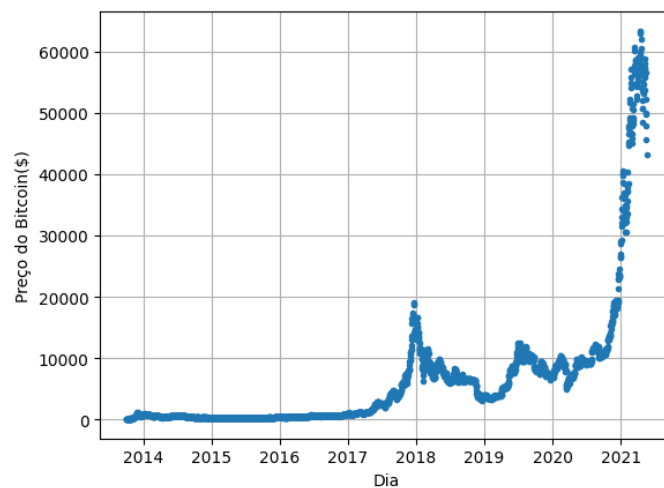


Figura 1: Gráfico com a série temporal do preço dos Bitcoins.

Observando o gráfico, é possível identificar algumas características dessa série temporal. Note, que nos primeiros anos, o preço do Bitcoin quase não varia. Contudo, próximo da metade de 2017, os preços começam a variar mais drasticamente, apresentando um comportamento muito mais volátil do que anteriormente. Além disso, conforme os dados aproximam-se do fim, ocorre um grande pico no preço.

Destarte, é notável que a série temporal apresenta características completamente distintas de 06/2017 para frente e para trás. Não obstante, não é possível identificar qualitativamente nenhuma forma de padrão a respeito dos dados, a exceção do fato do preço do Bitcoin estar, durante a maioria da série temporal, crescendo. Para melhorar a intuição a respeito dos dados, na Tabela 1 temos algumas informações estatísticas básicas (apresentadas em dólares).

Média	6529.845
Desvio padrão	10887.847
Máximo	63346.789
Mínimo	108.585

Tabela 1: Informações básicas sobre a série temporal do preço dos Bitcoins.

Observando a tabela, fica óbvio o quanto o preço do Bitcoin variou na janela analisada. O valor máximo é cerca de 10 vezes o preço médio, enquanto o mínimo é 60 vezes menor. Não obstante, o desvio padrão é 166.75% a média. Tudo isso é consistente com a área em questão, já que as criptomoedas e o próprio mercado financeiro de modo geral tem uma alta volatilidade e imprevisibilidade características.

### 2.1.2 Série 2: produção de energia elétrica

A segunda série temporal que será utilizada para treinar e testar ambos os modelos é uma série temporal da produção de energia elétrica mensal de janeiro de 1985 até janeiro de 2018. Essa série pode ser encontrada em [4]. Não foi possível obter algumas informações a respeito da série temporal: a unidade de medida utilizada para a coleta de dados, nem a origem dessa produção de eletricidade, ou seja, se esses dados são referentes a um país ou a uma usina. Contudo, essas informações ausentes não são impeditivas, pois, considerando que é a mesma unidade de medida durante toda a série temporal, as previsões podem ser feitas do mesmo jeito. Novamente, iniciaremos as discussões preliminares a respeito da série temporal com a exibição de um gráfico de tal:

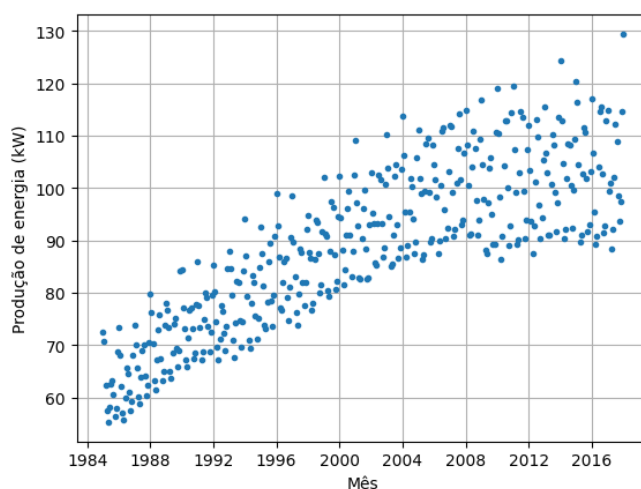


Figura 2: Gráfico com a série temporal da produção de energia elétrica mensal.



As diferenças visuais com a série temporal anterior são imediatas. A primeira diferença notável é a densidade dos pontos. Enquanto a série temporal anterior tinha 2787 entradas entre os anos de 2013 e 2021, nessa série temos apenas 397 espalhados ao longo de mais de 3 décadas (33 anos). Assim, claramente, o gráfico dos pontos é muito mais espalhado, não havendo tantos pontos concatenados.

Há outra diferença notável nos dados: enquanto os dados anteriores tinham um comportamento caótico, sem uma tendência facilmente identificável (com diversos picos que depois reduziam a normalidade, com um comportamento oscilatório), essa nova série temporal evidentemente tende a um crescimento constante da produção de energia elétrica. Além disso, os pontos próximos aparentemente tendem a ficar aproximadamente em um mesmo intervalo de valores, sem variações abruptas extremas como ocorria com a série temporal 1.

Para obter mais conclusões, observe os gráficos com apenas o primeiro ano (1985) (Figura 3) e com a primeira década (Figura 4):

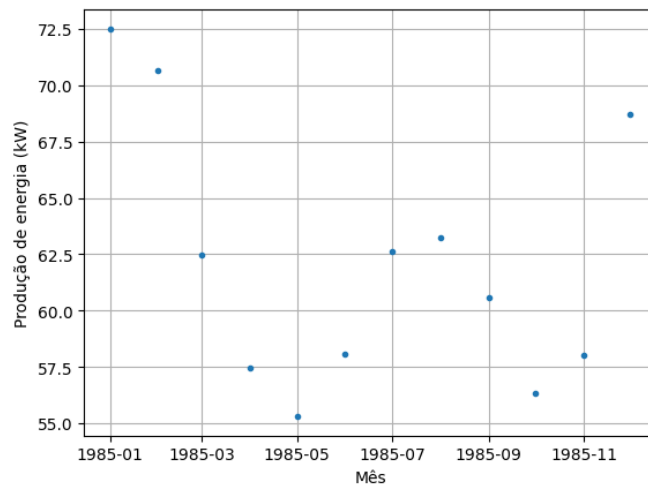


Figura 3: Gráfico da produção de energia elétrica mensal de 1985.

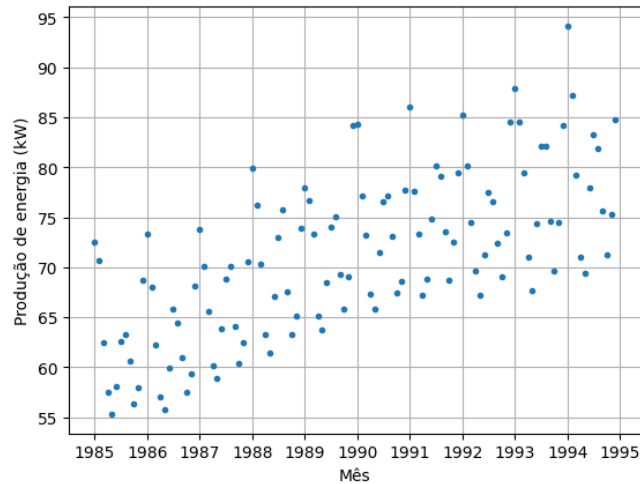


Figura 4: Gráfico da produção de energia elétrica mensal para os primeiros 10 anos.

Assim, analisando a Figura 3 é possível identificar que em 1985, a produção de energia elétrica teve seus maiores valores em janeiro, fevereiro e dezembro e, que nos meses de abril, maio, junho, outubro e novembro teve valores consideravelmente menores. Observando a Figura 4, é possível concluir que esse padrão identificado em 1985 ocorre também nos outros anos. Isso é um forte indício de que essa série temporal possui padrões anuais que se repetem periodicamente ano após ano.

Essa hipótese também faz sentido por conta do contexto desses dados, pois é esperado que a produção de energia seja mais elevada em meses com maior demanda energética, caso essa produção seja advinda de usinas termoelétricas, ou que seja mais elevada quando as condições climáticas são mais favoráveis à produção de energia, como para usinas solares e hidroelétricas, que dependem da disponibilidade de luz do Sol e de um volume mínimo dos rios.

Portanto, após essa análise, é de se esperar que os modelos tenham uma facilidade maior em prever essa série temporal do que a série do preço do Bitcoin, pois é significativamente mais estável e possui padrões de comportamento identificáveis qualitativamente.

Finalmente, serão exibidas as mesmas medidas estatísticas básicas apresentadas para a série temporal anterior (Tabela 2):

Média	88.847
Desvio padrão	15.388
Máximo	129.405
Mínimo	55.315

Tabela 2: Informações básicas sobre a série temporal da produção de eletricidade.

Observe que essas informações também apontam para uma maior estabilidade dos dados, pois o desvio padrão corresponde a apenas 17.3% da média, uma diferença gritante do caso anterior. Assim, é esperado que prever essa nova série temporal seja uma tarefa de menor dificuldade se comparada com o preço do Bitcoin.

## 2.2 Modelos

Antes de apresentar os modelos, é conveniente já apresentar duas notações muito importantes para ambos os modelos:

- $T$ : quantidade de entradas utilizadas pelos modelos para prever  $H$ ;
- $H$ : horizonte, quantas futuras entradas devem ser previstas pelo modelo de cada vez;

Essas não são as únicas notações a serem utilizadas, mas notações específicas ao modelo N-BEATS e às métricas de avaliação serão apresentadas em suas respectivas sessões.

### 2.2.1 Modelo 1: Denso

O primeiro modelo a ser implementado é um modelo denso, isso é, uma rede neural formada por 2 camadas totalmente conectadas. Essas camadas recebem essa nomenclatura porque todas as unidades de ativação (neurônios) de uma camada se conectam com todos os neurônios da camada subsequente. Cada unidade de ativação consiste na combinação linear dos valores dos neurônios da camada anterior multiplicados por pesos (os quais são os parâmetros do modelo que devem ser treinados) e por uma função de ativação aplicada nessa combinação linear. O resultado dessa função é a saída dessa unidade de ativação. A primeira camada possui 128 unidades de ativação, onde a função de

ativação utilizada é a ReLU (*Rectified Linear Unit*) (1):

$$\text{ReLU}(x) = \max(0, x) \quad (1)$$

Já a última camada é simplesmente um único neurônio sem nenhuma função de ativação (ou seja, é simplesmente a combinação linear das saídas da camada anterior).

Esse modelo simples será utilizado como referência para o desempenho do modelo N-BEATS, o qual é consideravelmente mais complexo e, portanto, é esperado que tenha uma performance melhor, para compensar o custo computacional de tal.

### 2.2.2 Modelo 2: N-BEATS

O outro modelo que será implementado é o modelo N-BEATS. Esse modelo foi introduzido no artigo [3]. Basicamente, o modelo N-BEATS é um conjunto de  $L$  blocos residuais: em cada bloco  $l$ , temos um mapeamento não-linear das entradas em duas saídas distintas. Esse mapeamento consiste, basicamente, em quatro camadas totalmente conectadas com alguma função de ativação não-linear (no caso, a função ReLU) e seguidas por uma camada linear que irá gerar as duas saídas distintas. Tais saídas possuem propósitos diferentes: uma das saídas ( $y_l$ ) irá compor a predição final do modelo, enquanto a outra será utilizada ( $\hat{x}_l$ ) para compor a entrada para o próximo bloco. Essa dinâmica pode ser visualizada mais facilmente na Figura 5.

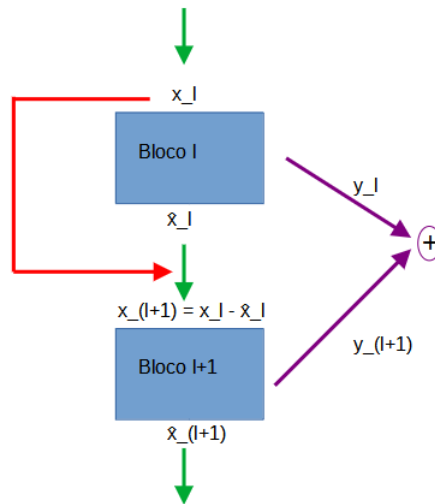


Figura 5: Diagrama de como os dados passam pelo N-BEATS.

Observação: nesse diagrama,  $x_{l-1}$  representa a entrada para o bloco  $l$ ,  $\hat{x}_{l-1}$  representa a saída do bloco  $l$ , que irá fazer parte da entrada do bloco  $l + 1$  e  $y_{l-1}$  é a saída do bloco que irá compor a predição.

Assim, a entrada para o bloco  $l$  (com  $1 < l \leq L$ ) é dada pela seguinte fórmula:

$$x_l = x_{l-1} - \hat{x}_{l-1} \quad (2)$$

Para o caso em que  $l = 1$ , ou seja, para o primeiro bloco do modelo, a entrada dele é simplesmente as  $T$  últimas entradas da série temporal, ou seja, é simplesmente a entrada do modelo. Esse uso da saída de um bloco para calcular a entrada do outro é o que dá o nome a esses blocos de blocos residuais.

Além disso, a predição fornecida pelo modelo é:

$$\hat{y} = \sum_{l=1}^{l=L} y_l \quad (3)$$

Ou seja, a predição do modelo N-BEATS é simplesmente a soma de todas as saídas dos blocos que o compõem.

Assim, esse modelo é significativamente mais complexo do que o Modelo 1. Portanto, é de se esperar que ele apresente uma performance superior a do modelo mais simples, pois além de mais complexo, ele também possui muito mais parâmetros e é muito mais caro (e lento) para ser treinado. Destarte, para o custo-benefício valer a pena, é necessário que ele apresente um desempenho superior.

## 2.3 Métricas de avaliação

Por fim, mas não menos importante, uma das partes mais fundamentais do trabalho é a avaliação e comparação dos resultados obtidos. Para tal, é necessário possuir métricas de avaliação que realmente discriminem a performance dos dois modelos. Portanto, serão utilizadas 4 métricas distintas de avaliação. As novas notações introduzidas nessa sessão são:

- $x$ : as entradas da série temporal que serão utilizadas para o cálculo das predições;

- $\hat{y}$ : predição para as  $H$  próximas entradas da série temporal;
- $y^*$ : as verdadeiras próximas  $H$  entradas da série temporal (a resposta esperada para  $\hat{y}$ );
- $N$ : a quantidade de previsões realizadas pelo modelo durante o teste (ou seja, é o tamanho do conjunto de teste).
- **Mean Absolute Error (MAE)**: mede a magnitude média dos erros da predição:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i^* - \hat{y}_i| \quad (4)$$

Essa é a métrica que o treinamento dos modelos buscará minimizar, assim, essa é a *loss function* (ou função de perda) dos modelos.

- **Root Mean Squared Error (RMSE)**: é a raiz quadrada do erro quadrático médio. Assim, é dado um peso maior para erros de maior magnitude do que no MAE. O resultado está na mesma escala dos dados por conta da raiz quadrada:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i^* - \hat{y}_i)^2} \quad (5)$$

- **Mean Absolute Percentage Error (MAPE)**: é o erro absoluto percentual médio, assim, essa medida não tem escala:

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \frac{|y_i^* - \hat{y}_i|}{|y_i^*|} \quad (6)$$

- **Mean Absolute Scaled Error (MASE)**: outro erro que não depende da escala da série temporal. O MASE consiste, basicamente, em uma comparação do modelo em questão com o preditor *naive*. Esse preditor *naive* simplesmente prevê como saída a entrada anterior da série temporal e, embora, isso pareça extremamente trivial e ineficiente, para horizontes unitários ele é surpreendentemente eficiente e

nem sempre modelos complexos conseguem superá-lo.

$$\text{MASE} = \frac{1}{N} \frac{\sum_{i=1}^N |y_i^* - \hat{y}_i|}{\sum_{i=1}^N |x_i - x_{i-1}|} \quad (7)$$

Assim, pela Equação (7), fica evidente que para valores de MASE maiores do que 1, temos um preditor pior do que o *naive*, enquanto para valores menores temos um preditor melhor.

Além disso, para horizontes maiores do que 1, o MASE perde um pouco do seu significado, pois a ideia por trás dessa métrica é fundamentalmente para horizontes 1. Por conta disso, para horizontes maiores do que 1 essa métrica não será calculada.

Essas são as 4 métricas que serão utilizadas para avaliar os resultados. É importante ressaltar que para todas as medidas apresentadas, valores menores representam desempenho melhor.

### 3 Experimentos

Os experimentos realizados com as duas séries para os dois modelos, consistiram em simplesmente treinar os modelos utilizando os primeiros 80% dos dados e avaliá-los usando os últimos 20%. Em cada experimento, cada modelo foi treinado cinco vezes e foi feita uma média da performance para obter um resultado final. A realização de cinco treinamentos e avaliações é necessária porque os resultados podem variar bastante, já que há uma inicialização aleatória dos parâmetros de cada modelo. Assim, ao fazer isso cinco vezes e calcular uma média, obtém-se um resultado mais confiável a respeito dos modelos em cada configuração.

Para obter uma comparação mais completa dos dois modelos, em problemas diferentes, foi escolhido avaliar os modelos em três situações nas duas séries temporais, variando os valores de  $T$  e de  $H$ . Assim, foram escolhidas as seguintes combinações:  $T = 7$  e  $H = 1$ ;  $T = 30$  e  $H = 1$ ;  $T = 30$  e  $H = 7$ . Assim, será possível avaliar como os modelos melhoram com mais dados disponíveis para realizar as previsões e qual o desempenho

deles quando deseja-se obter uma predição para um horizonte mais longo. Os resultados para esses experimentos serão apresentados e discutidos na seção subsequente.

Além disso, uma das partes mais importantes para qualquer modelo de aprendizado de máquinas, é a escolha dos hiperparâmetros. Para o modelo denso, foi escolhida uma camada de ativação com 128 neurônios, 100 épocas de treinamento e tamanho do *batch* de 128.

Já para o modelo N-BEATS, foram escolhidos um tamanho de *batch* de 1024, 5000 épocas de treinamento, 512 neurônios em cada uma das quatro camadas totalmente conectadas dentro de cada bloco e 30 blocos no total. Em ambos os casos, a função de perda foi a MAE e foi utilizado um otimizador do tipo Adam.

É importante ressaltar, que o modelo N-BEATS possui tantas épocas porque ele demora muito mais tempo para convergir a resultados razoáveis e, caso o treinamento estagnasse por mais de 100 épocas, foi utilizado *early stopping*.

## 4 Resultados

Os resultados dos experimentos realizados para a série dos preços do Bitcoin estão nas Tabelas 3 e 4, respectivamente, para o modelo denso e o modelo N-BEATS:

Métrica	$T = 7, H = 1$	$T = 30, H = 1$	$T = 30, H = 7$
MAE	570.522	604.303	1206.717
RMSE	1077.110	1126.823	1386.485
MAPE	2.588	2.781	5.531
MASE	1.016	1.070	-

Tabela 3: Resultados dos experimentos para o preço do Bitcoin para o modelo denso.

Métrica	$T = 7, H = 1$	$T = 30, H = 1$	$T = 30, H = 7$
MAE	578.968	695.075	837.008
RMSE	1085.529	1317.04	1012.208
MAPE	2.649	3.083	3.911
MASE	1.017	1.215	-

Tabela 4: Resultados dos experimentos para o preço do Bitcoin para o modelo N-BEATS.

Observando os resultados, a primeira conclusão, é que ambos os modelos performaram piores do que o preditor *naive*, pois em todos os 4 experimentos cujo horizonte



era unitário, o MASE foi maior do que 1. Esse fato é consideravelmente surpreendente, já que os modelos utilizados são significativamente mais complexos e, portanto, era de se esperar uma performance melhor do que do preditor *naive*.

Outro resultado notável é que as performances do modelo denso e do N-BEATS são basicamente idênticas quando  $T = 7$ . No entanto, o desempenho do modelo N-BEATS foi consideravelmente pior para um horizonte de 30 entradas, com um erro absoluto médio cerca de 15% maior do que o modelo denso. Além disso, em ambos os casos, a performance do modelo foi pior quando  $T$  foi aumentado. Embora isso pareça inusitado à primeira vista, esse resultado não é completamente inesperado, pois como foi inferido visualmente, a série temporal do Bitcoin é altamente instável, variando drasticamente de um instante para o outro, o que indica que fornecer mais informações para o modelo não necessariamente o auxiliará, já que as informações mais distantes podem não auxiliar na previsão atual. O fato de que o desempenho foi pior para  $T$  maior é um indício de que esse é o caso aqui.

Para a última configuração, quando é demandado do modelo um horizonte de previsões maior do que 1, no caso  $H = 7$ , conclui-se que ambos os modelos apresentam um desempenho pior do que quando  $H = 1$ . Tal piora é esperada, já que é uma tarefa significativamente mais complexa prever horizontes maiores. Contudo, a comparação entre o modelo N-BEATS e o modelo denso é contrastante. Enquanto o modelo N-BEATS teve um erro absoluto médio de apenas \$837.008, o modelo denso teve um erro de \$1206.717, um aumento extremamente significativo de 44.1%. Além disso, o erro do modelo denso quase dobrou de  $H = 1$  para  $H = 7$ . Tal resultado, demonstra que o modelo N-BEATS foi expressivamente superior ao modelo denso com horizontes maiores.

Por último, lembrando que a média dos valores é de \$6529.845, observe que o melhor modelo (modelo denso com  $T = 7, H = 1$ ) teve um erro médio de quase \$570, o que representa aproximadamente 9% do valor médio da série temporal.

Analogamente, as Tabelas 5 e 6 apresentam os resultados para a série temporal da produção de eletricidade, respectivamente para o modelo denso e para o modelo N-BEATS.

Métrica	$T = 7, H = 1$	$T = 30, H = 1$	$T = 30, H = 7$
MAE	3.935	3.061	3.740
RMSE	5.051	4.009	4.556
MAPE	3.777	2.982	3.657
MASE	0.488	0.379	-

Tabela 5: Resultados dos experimentos para a produção de energia para o modelo denso.

Métrica	$T = 7, H = 1$	$T = 30, H = 1$	$T = 30, H = 7$
MAE	2.629	2.645	9.415
RMSE	3.421	3.707	11.665
MAPE	2.527	2.541	9.296
MASE	0.316	0.322	-

Tabela 6: Resultados dos experimentos para a produção de energia para o modelo N-BEATS.

De imediato, é evidente que ambos os modelos foram muito melhores na predição da série 2, como era de se esperar. Contudo, o tamanho da diferença é notável. Enquanto na série 1 nenhum dos modelos conseguiu um MASE menor do que 1, para a segunda série temporal eles conseguiram valores menores do que 0.5 e, de modo especial, o modelo N-BEATS conseguiu um MASE menor do que 0.35. Assim, a performance do modelo é consideravelmente melhor agora. Isso é fortalecido pelo MAE, cujo melhor resultado foi 2.629, o que corresponde a somente 3% do valor médio da série temporal (88.847). Destarte, a avaliação inicial de que a série temporal sobre a produção de eletricidade é consideravelmente mais previsível se comparada com a série temporal do preço dos Bitcoin é atestada por esses resultados.

Entretanto, há um caso em que evidentemente isso não ocorreu: modelo N-BEATS com  $H = 7$ , onde o resultado foi mais de 4 vezes pior do que quando  $H = 1$  e o erro médio foi quase 3 vezes maior se comparado com a configuração equivalente do modelo denso. Evidentemente, esse resultado não é esperado ou adequado para um modelo a ser usado na realidade. Todavia, esse é claramente um caso discrepante e há algumas razões possíveis para isso, nenhuma das quais será provada ou refutada neste projeto. A primeira razão para isso seria algum comportamento atípico do otimizador do modelo. Essa possibilidade é bastante reduzida já que foi feita uma análise minuciosa do modelo e, além disso, é o mesmo modelo utilizado para os outros experimentos, os quais tiveram resultados satisfatórios; assim, essa situação, embora nunca possa ser completamente

descartada, não parece ser a explicação mais provável. A outra possibilidade principal, e a que parece ser a culpada, é que o modelo N-BEATS, com essa configuração específica de hiperparâmetros, de  $H$  e de  $T$ , simplesmente não funcionou como deveria para essa série temporal. Os modelos de *deep learning* em geral costumam ser bastante vulneráveis a variações dos hiperparâmetros e o modelo N-BEATS não é exceção, sendo especialmente suscetível a variações, por exemplo, da quantidade de unidades de ativação em cada camada. Portanto, esse parece ser um caso ilustrativo da importância da escolha dos hiperparâmetros para os modelos de aprendizado de máquinas.

Comparando os resultados do modelo denso com o modelo N-BEATS, temos que para a primeira configuração ( $T = 7$ ), o desempenho do N-BEATS é consideravelmente superior, sendo que o erro do modelo denso é quase 50% maior em relação ao erro médio do N-BEATS.

Com o aumento de  $T$ , os resultados para a série 2 diferiram dos resultados anteriores. Para a série da produção de eletricidade, nenhum dos modelos teve uma piora significativa em relação ao  $T$  menor (o modelo N-BEATS teve um desempenho pior, mas o aumento do erro foi de apenas 0.016, um valor relativamente pequeno se comparado com o valor médio da série), enquanto o modelo denso obteve um resultado melhor em relação à configuração com  $T = 7$ . Em contraste, como discorrido anteriormente, com a série do Bitcoin, os modelos foram piorados com o aumento de  $T$ . Esse resultado também é esperado, já que a série temporal da produção energética tem um comportamento de certo modo periódico, assim, o fornecimento de informações de um passado mais distante deve auxiliar na previsão. No entanto, o modelo N-BEATS permaneceu sendo superior ao modelo denso, embora a diferença de performance entre os dois tenha sido reduzida com  $T = 30$ .

Finalmente, para  $H = 7$ , o modelo denso teve um aumento de 20% do erro médio em relação a  $H = 1$ , contudo, tal resultado ainda foi superior ao modelo denso com  $T = 7$  e  $H = 1$ . Pelos motivos discutidos anteriormente, para o experimento com horizontes maiores, o modelo denso foi superior ao N-BEATS.

## 5 Conclusão

Destarte, os resultados apresentados na seção anterior permitem algumas conclusões interessantes:

- A série temporal da produção de eletricidade é, como previsto, significativamente mais previsível se comparada com a série do preço do Bitcoin;
- A escolha dos hiperparâmetros é uma parte da construção dos modelos extremamente importante e, uma escolha ruim, pode afetar de forma intensa os resultados;
- O aumento de  $H$  dificultou consideravelmente a predição em relação ao caso unitário (as comparações são feitas para  $T = 30$ );
- O aumento de  $T$  é uma escolha cujo impacto varia de série para série. Para a série 1, muito mais volátil, essa não é uma boa escolha, já que impacta negativamente os resultados. Por outro lado, para a série 2, esse aumento melhora o desempenho dos modelos (ou piora minimamente). Assim, não há uma resposta definitiva para essa questão;
- Para séries temporais mais complicadas (série 1), ambos os modelos apresentaram uma dificuldade incrível em superar o preditor *naive*. Por outro lado, para a série mais bem comportada, tal preditor foi facilmente superado pelos dois modelos. Assim, a dificuldade de vencer esse modelo simples varia de acordo com a série temporal e suas características;
- Finalmente, nenhum dos modelos conseguiu demonstrar que é irrefutavelmente superior ao outro. Ambos os modelos apresentaram resultados inferiores ao ideal com a série 1, onde o modelo denso foi superior para  $H = 1$ , enquanto o N-BEATS apresentou um desempenho melhor para  $H = 7$ . Todavia, a situação oposta ocorreu para a série 2, onde os resultados foram satisfatórios, o N-BEATS foi superior para  $H = 1$  e inferior para  $H = 7$ . Portanto, não é possível apontar um vencedor evidente dessa comparação. Esse resultado inconclusivo não é uma surpresa, já que o desempenho de modelos de aprendizado de máquinas costuma variar conforme o problema.

## Referências

- [1] Daniel Bourke. Milestone project 3: Time series forecasting in tensorflow.
- [2] Shi Dong, Ping Wang, and Khushnood Abbas. A survey on deep learning and its applications. *Computer Science Review*, 40:100379, 2021.
- [3] B. N. Oreshkin, D. Capov, N. Chapados, and Y. Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *ICLR*, 2020.
- [4] ShenbagaKumarS. Time series datasets.
- [5] José F. Torres, Dalil Hadjout, Abderrazak Sebaa, Francisco Martínez-Álvarez, and Alicia Troncoso. Deep learning for time series forecasting: A survey. *Big Data*, 9, 2021.