

**UNIVERSIDADE DE SÃO PAULO**

Instituto de Ciências Matemáticas e de Computação

**Comparação entre diferentes redes neurais para a predição  
do valor de ações na bolsa de valores**

**Ana Clara Tezoto Figueiroa**

Trabalho de conclusão de curso do MBA em Ciência de Dados (CEMEAI)



SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: \_\_\_\_\_

**Ana Clara Tezoto Figueiroa**

## Comparação entre diferentes redes neurais para a predição do valor de ações na bolsa de valores

Trabalho de conclusão apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestra – MBA em Ciência de Dados. *VERSÃO REVISADA*

Área de Concentração: Matemática, Estatística e Computação

Orientador: Prof. Dr. Afonso Paiva Neto

**USP – São Carlos**  
**Abril de 2022**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados inseridos pelo(a) autor(a)

T253c Tezoto Figueiroa, Ana Clara  
Comparação de diferentes redes neurais na predição  
da bolsa de valores / Ana Clara Tezoto Figueiroa;  
orientador Afonso Paiva Neto. -- São Carlos, 2022.  
42 p.

Trabalho de conclusão de curso (MBA em Ciência  
de Dados) -- Instituto de Ciências Matemáticas e de  
Computação, Universidade de São Paulo, 2022.

1. Bolsa de valores. 2. MLP. 3. LSTM. 4. Sarima.  
5. GRU. I. Paiva Neto, Afonso, orient. II. Título.

**Ana Clara Tezoto Figueiroa**

Comparisson between different neural networks on the  
prediction of stocks close values

Master dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP, in partial fulfillment of the requirements for the degree of the Master – Masters of Business Administration in Data Science. *FINAL VERSION*

Concentration Area: Mathematics, Statistics and Computing

Advisor: Prof. Dr. Afonso Paiva Neto

**USP – São Carlos**  
**April 2022**



# AGRADECIMENTOS

---

---

Agradeço meu orientador Prof. Dr. Afonso Paiva Neto por todo o apoio e disponibilidade na produção desse trabalho.



# RESUMO

FIGUEIROA, A. C. T. . **Comparação entre diferentes redes neurais para a predição do valor de ações na bolsa de valores.** 2022. 44 p. Trabalho de conclusão (curso – MBA em Ciência de Dados) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

Neste trabalho é feita uma análise e comparação de diferentes métodos para a predição da bolsa de valores, considerando a cotação final diária de ações nos últimos 5 anos. Para a modelagem das predições foram utilizados métodos SARIMA (*Seasonal Autoregressive Integrated Moving Average*) e métodos de redes neurais profundas, incluindo *Multilayer perceptron* (MLP), *long short term memory* (LSTM) e *gate recurrent unit* (GRU). Os resultados são analisados não só quanto à acuracidade, mas também o quão tais predições poderiam dar suporte para investidores na bolsa de valores.

**Palavras-chave:** Bolsa de valores, MLP, LSTM, Sarima, GRU.



# ABSTRACT

FIGUEIROA, A. C. T. . **Comparisson between different neural networks on the prediction of stocks close values**. 2022. 44 p. Trabalho de conclusão (curso – MBA em Ciência de Dados) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

This paper is an analysis and comparison of different methods to predict the values on the stock market, considering the daily close prices of stocks of the last 5 years. For the modeling of the predictions it used the methods SARIMA (Seasonal Autoregressive Integrated Moving Average) and deep neural networks as MLP (Multilayer perceptron), long short term memory (LSTM) e gate recurrent unit (GRU). The results are analyzed not only on their accuracy, but also whether they can give support to decision for a stock market investor.

**Keywords:** Stock market, MLP, LSTM, Sarima, GRU.



# LISTA DE ILUSTRAÇÕES

---

---

Figura 1 – Serie Temporal - Apple . . . . .	26
Figura 2 – Boxplot da derivada - Apple . . . . .	27
Figura 3 – Decomposição da serie temporal - Amazon . . . . .	28
Figura 4 – MLP Exemplo . . . . .	31
Figura 5 – RNN Exemplo . . . . .	33
Figura 6 – LSTM Exemplo . . . . .	33
Figura 7 – GRU Exemplo . . . . .	34
Figura 8 – Series normalizada e dividida em treinamento e teste . . . . .	35
Figura 9 – Predição Sarima com separação 70/30 nos dados teste . . . . .	36
Figura 10 – Predição Sarima com separação 80/20 nos dados teste . . . . .	37
Figura 11 – Resultados de predição para ações da Apple . . . . .	38
Figura 12 – Resultados de predição para ações da Amazon . . . . .	39
Figura 13 – Resultados de predição para ações da Google . . . . .	39
Figura 14 – Comparação de resultados da Amazon considerando inclinação das curvas . . . . .	40



# LISTA DE TABELAS

---

---

Tabela 1 – Pesquisas relacionadas . . . . .	22
---	----



# SUMÁRIO

---

---

1	INTRODUÇÃO	17
1.1	Motivação	17
1.2	Objetivo	18
1.3	Estrutura do trabalho	18
2	REVISÃO BIBLIOGRÁFICA	21
3	METODOLOGIA	25
3.1	Introdução a metodologia	25
3.2	Coleta, análise e tratamento de dados	25
3.3	Séries temporais	27
3.4	Modelos auto regressivos e média móvel	29
3.5	Redes neurais	30
3.5.1	<i>Multilayer Perceptron (MLP)</i>	31
3.5.2	<i>Camada recorrente básica (RNN)</i>	32
3.5.3	<i>Long short term memory (LSTM)</i>	33
3.5.4	<i>Gated recurrent unit (GRU)</i>	34
4	RESULTADOS	35
4.1	Arima e Sarima	36
4.2	MLP	36
4.3	LSTM e GRU	37
4.4	Comparação de resultados	38
5	CONCLUSÃO	41
	REFERÊNCIAS	43



---

# INTRODUÇÃO

---

## 1.1 Motivação

A bolsa de valores é extremamente volátil e apresenta grandes riscos, principalmente para investidores amadores. Há diferentes métodos de aplicações, entre eles *day-trade*, *buy and hold* e *swing trade*. No *day-trade*, o acionista faz transações de compra e venda em um mesmo dia, se aproveitando das oscilações de curtíssimo prazo. No *swing trade*, o acionista acompanha as oscilações dos ativos em um período de dias. Na estratégia *buy and hold* o acionista compra as ações com foco em um longo prazo, com a expectativa dessa ativo subir consideravelmente em questão de meses até anos. Aqui também é a parte do foco frequentemente ganhar os dividendos anuais dessas ações.

Independentemente do estilo de aplicação, as transações giram em torno de um ponto central: o valor da ação. A negociação básica na bolsa de valores é a decisão de comprar ou vender ações baseado na cotação atual destas e a expectativa dos valores futuros que esta assumirá. Para *day trade* é analisado pequenas oscilações que ocorrem em questão de minutos e horas, sendo sua predição muito difícil. Nessa estratégia, o principal objetivo é ter uma reação rápida a oscilações e não uma predição correta dos valores. No *swing trade* a predição de valores ganha uma maior importância, como por exemplo qual valor se espera de uma ação nos próximos dias, para a tomada de decisão de compra e venda. Já na estratégia *hold and buy* exige uma maior confiança sobre o valor futuro da ação, devido a estratégia ser uma aposta a longo prazo de que essa ação vai aumentar de valor, e não uma aposta nas oscilações da mesma.

A cotação de uma ação é um indicativo indireto de quanto uma empresa vale. Na maior parte dos casos, conforme uma empresa gera mais lucro, a cotação da ação desta aumenta. Com base nisso os acionistas utilizam diferentes estratégias para predizer se vale a pena investir em uma ação ou não, principalmente para o método *buy and hold*.

A principal estratégia é a análise fundamentalista de [Graham e Dodd \(2008\)](#), na qual o

investidor analisa os fundamentos que sustentam a ação. Aqui são analisados aspectos quantitativos e qualitativos, criando uma perspectiva de como a ação estará a longo prazo. Para valores quantitativos, compara-se por exemplo qual a cotação da ação no mercado e o valor intrínseco. Se o valor intrínseco for maior, pode significar que a cotação do ativo no mercado está muito baixa e tenderá a aumentar. Outros valores quantitativos analisados são por exemplo a receita, o lucro, as dívidas. Para a análise qualitativa, analisa-se por exemplo a notoriedade da marca, quantidade de patentes, política externa e potencial de crescimento. Uma empresa com uma boa notoriedade e um bom plano de desenvolvimento, por exemplo, tende a aumentar de valor.

O problema desse tipo de análise é a exigência de tempo para a coleta e análise de dados, e a experiência própria do analista sobre como valores qualitativos podem afetar o futuro do ativo. Isso dificulta o processo para investidores amadores sem experiência, e encarece o processo para investidores experientes. Por conta disso surgiu a motivação de criar ferramentas que facilitem a predição da ação. Bons sistemas de investimento, por exemplo, já fornecem os valores quantitativos de uma análise fundamental calculados, de maneira que o investidor não precisa procurar os valores da empresa (por exemplo a receita) para calcular. Outros trabalhos procuram utilizar a ciência de dados para procurar prever o valor futuro de uma ação, por exemplo analisando a notoriedade de uma empresa nas redes sociais. Uma conclusão de diferentes trabalhos é a dificuldade de prever o valor correto de ações. Assim, a motivação deste trabalho é analisar métodos que deem suporte à tomada de decisão na escolha de ações para comprar e vender.

## 1.2 Objetivo

O problema da predição da bolsa de valores utilizando a ciência de dados já foi aproximado de diferentes maneiras, seja por análises qualitativas, quantitativas ou misturadas, utilizando tanto métodos de regressão e classificação.

O objetivo deste trabalho é focar na predição do valor de uma ação baseada apenas na série temporal de valores históricos da mesma. Aqui serão analisados diferentes modelos, como SARIMA, redes neurais, LSTM, GRU, e comparadas a sua efetividade na predição. Não será analisada só a acuracidade dos modelos, mas a capacidade de suporte que poderiam dar para um acionista. Por exemplo, se a predição não resulta em um valor extremamente acurado, mas consegue efetivamente prever se a ação vai aumentar de valor ou diminuir, e em qual prazo são mais efetivos (longo, curto, intradia).

## 1.3 Estrutura do trabalho

A seguir no capítulo 2, será feita uma análise bibliográfica sobre os temas deste trabalho. Neste capítulo serão primeiramente analisados os diferentes métodos já utilizados para a predição

da bolsa de valores, incluindo métodos de regressão e classificação. Alguns métodos utilizam séries temporais como neste trabalho, e outros utilizam outras fontes de dados como notícias e redes sociais para a predição, ou a mistura de ambos.

Após a análise de trabalhos focados na predição de bolsa de valores, será feita uma análise de trabalhos em predições especificamente de séries temporais. Para isso há métodos estatísticos, métodos mais simples de regressão, até redes neurais e aprendizado de máquina.

No Capítulo 3 será feito um detalhamento dos métodos utilizados para o trabalho. Inicialmente será detalhado como a base de dados foi escolhida, como essa foi processada e preparada para a modelagem. A seguir os diferentes métodos de predição utilizados estão explicados, incluindo método SARIMA, redes neurais, LSTM, GRU.

No capítulo 4 serão detalhados as aplicações dos métodos em diferentes exemplos reais da bolsa de valores e os seus resultados. Esses resultados serão comparados entre si e entre as diferentes séries temporais, e qual a sua real aplicação e efetividade para um acionista.

No último capítulo 5 serão tomadas as conclusões desse projeto. O quão efetivo são métodos clássicos de predição de séries temporais para as cotações de ações e o suporte que eles podem dar na tomada de decisões. Aqui também será especificado diferentes ideias para trabalhos futuros que possam avançar os estudos de um suporte computacional para tomada de decisão sobre a bolsa de valores.



---

## REVISÃO BIBLIOGRÁFICA

---

O preço de uma ação na bolsa de valores é extremamente volátil. Na maior parte dos casos o preço não é simplesmente o valor da empresa dividido pelo número de ações, mas também depende do que os investidores sentem que a empresa realmente vale. Nesse ponto influenciam situações políticas e diplomáticas que afetem a operação da empresa, declarações da empresa sobre seus planos futuros, problemas na cadeia de fornecimento, entre outros.

Por conta dessa volatilidade, a predição dos preços de ação é extremamente complicada. Uma dos focos de pesquisa é a predição baseada no histórico de preços (série temporal), que procura encontrar a cotação exata de uma ação no futuro. Mas como esse tipo de predição não leva em conta todos os fatores que causam volatilidade, a precisão é limitada. Um segundo foco de pesquisa é utilizar-se de outras fontes de dados para o suporte da predição, como notícias e redes sociais. Aqui a predição numeral pode ser mais limitada, mas permite uma visão mais ampla do problema.

[Seong e Nam \(2021\)](#) por exemplo, procuram fazer uma predição de valores da bolsa coreana baseada em notícias financeiras com segmentação. Para isso é feito uma análise da psicologia dos investidores, o efeito da mídia, a situação de uma empresa sozinha e do setor da empresa combinado. Para o pré-processamento de dados, os autores utilizaram *bag-of-words*, e para o algoritmo de aprendizado de máquinas utilizou *multiple kernel learning*, para uma análise de *cluster*.

Seong também compara tipos de pesquisas na análise de notícias para predição de preços. Como pode ser visto na Tabela 1, a maior parte das pesquisas utiliza-se de notícias financeiras como base de dados, devido a sua maior proximidade ao assunto. Outras pesquisas usaram redes sociais como Twitter, Facebook e declarações das empresas.

Pode-se notar também que uma grande parte das pesquisas procuram classificar se o preço futuro vai aumentar ou diminuir, e poucas exceções tratam de predizer o real preço da ação. Isso é devido ao fato de uma cotação aumentar ou diminuir no futuro depende muito do

Tabela 1 – Pesquisas relacionadas

Reference	Dataset	Feature Extraction	Feature Selection	Feature representation	Machine Learning Algorithm	Forecast Type	Other Effects
Schumaker and Chen (Schumaker & Chen, 2009)	Financial news	Bag-of-words, Noun phrases, Named entities, Proper Nouns	Minimum occurrence per document	Binary	SVR	Price value	Relevant company's financial news
Hagenau et al. (Hagenau et al., 2013)	Corporate announcement and financial news	Bag-of-words, 2-Gram, 2-word combination, Noun phrases	News frequency, Chi-square, Bi-normal-separation	TF-IDF	SVM	Up and Down	None
Shynkevich et al. (Shynkevich et al., 2016)	Financial News	Bag-of-words	Chi-square	TF-IDF	Multiple kernel learning	Up and Down	Relevant company's financial news
De Fortuny et al. (Junqué de Fortuny et al., 2014)	Financial News	Bag-of-words	–	TF-IDF	SVM	Up and Down	Technical Indicators
Groth and Muntermann (Groth & Muntermann, 2011)	Corporate disclosures	Bag-of-words	Chi-square, Information Gain	TF-IDF	Naïve Bayes, k-NN, ANN, SVM	Up and Down	None
Bollen et al. (Bollen, Mao, & Zeng, 2011)	Twitter tweets	Sentiment Analysis	Sentiment Analysis	Sentiment Analysis	SOFNN	Stock Price and Up and Down	None
Deng et al. (Deng, Mitsubuchi, Shioda, Shimada, & Sakurai, 2011)	Social network sentiment, news comment	Sentiment Analysis	Sentiment Analysis	Sentiment Analysis	Multiple kernel learning	Price return	Comment, technical analysis

Fonte: Seong e Nam (2021)

sentimento do mercado, assim tais algoritmos de análise de sentimento do mercado permitem uma boa classificação se a cotação sobe ou desce, mas não uma descrição precisa do valor de uma ação. Para a extração de dados utilizados, temos principalmente *bag-of-words* e análise de sentimento, para algoritmos de aprendizado de máquina temos por exemplo *support vector machine* (SVM), *support vector regression* (SVR), *multiple kernel learning*, *Naive Bayes*, *k-vizinhos mais próximos* (KNN), redes neurais artificiais (ANN).

Geva e Zahavi (2014) procuraram incorporar tanto dados do mercado como notícias de texto, para criar uma predição automática diária de recomendação de compra e venda de ações. O principal objetivo dessa combinação é enriquecer as informações coletadas para ter uma predição mais precisa. Aqui o principal foco foi uma predição a curto prazo (entre dias), juntando predição baseada em uma série temporal com os dados externos textuais e suas influências. No tratamento de dados foi usado por exemplo *piecewise linear representation* (PLR) para as séries temporais, combinado com *bag-of-words* para os conteúdos de texto. Na modelagem foram usadas redes neurais, árvores de decisão e *stepwise-logistic-regression* (SLR).

Assim como Geva e Zahavi (2014), Barak, Arjmand e Ortobelli (2017) procura uma combinação de preditores para ter um melhor resultado. Para isso são usados diversos métodos como *Bagging*, *Boosting* e *Addaboost*. Li et al. (2014) também combina informações de preço de ações com notícias de mercado para a análise, utilizando principalmente *multiple kernel support vector regression* para predizer os valores a curto prazo das ações com maior precisão.

Existem muitas pesquisas no tema de predição de bolsa de valores, testando diferentes métodos de predição e a combinação de diferentes tipos de dados (série temporal e textos), a procura de um modelo de predição mais preciso. As possibilidades de combinações de tipos de

dados e métodos são inúmeras. A maioria das pesquisas atuais usa como base notícias financeiras e/ou dados históricos, apontando que uma análise da influência de outras fontes nos preços pode ser benéfica, como por exemplo declarações de empresas, notícias de setores do mercado, notícias políticas. Também é possível separar as empresas em setores para melhor abordar o comportamento de cada setor do mercado, como falta de fornecimento de matéria prima, queda de popularidade de um tipo de produto, entre outros.

Para predição de séries temporais sobre fechamento de cotação de ações, pesquisas testaram diferentes métodos como VAR (JUNG; BOYD, 1996), BVAR (BLEESSER; LIICOFF, 2005), ARIMA (ADEBIYI; ADEWUMI; AYO, 2014), GARCH (ZHANG; CHENG; WANG, 2005). O problema desses métodos é o alto grau de ruído nas séries temporais de ações, e a relação entre variáveis dos modelos tendem a mudar de maneira volátil com o tempo.



---

## METODOLOGIA

---

### 3.1 Introdução a metodologia

O primeiro passo de um projeto de ciência de dados é a escolha e/ou coleta dos dados a serem analisados. Esses dados geralmente não estão disponíveis de maneira pronta para análise, muitas vezes em formato não compatível ou com dados faltantes e errôneos. Por isso, após a coleta dos dados, esses dados devem ser verificados, corrigidos e preparados. Esse processo é chamado de pré-processamento dos dados.

Os dados prontos são então separados em dois conjuntos, o conjunto treinamento e o conjunto teste. O conjunto de treinamento é usado para o treinamento do modelo, sem influência do conjunto teste, desconhecido pelo modelo. Esse modelo treinado é então testado com o conjunto teste, para garantir uma melhor precisão. Assim podem ser feitas diferentes iterações de treinamento e teste para a escolha do melhor modelo com melhores atributos para os dados escolhidos.

### 3.2 Coleta, análise e tratamento de dados

A primeira atividade é a escolha ou coleta dos dados a serem utilizados e a análise da sua completude e acuracidade. Para esse projeto serão trabalhadas séries temporais do fechamento diário de cotação de ações.

Existem diferentes fontes para esses dados, entre eles fontes pagas e fontes gratuitas. Fontes pagas geralmente têm dados de períodos mais longos, por exemplo 20 anos, e incluem outros tipos de dados já processados como análises de compra e venda do próprio provedor de dados e cálculos para análise fundamentalista. Além disso, garante uma maior estabilidade e precisão do fornecimento de dados. Provedores de dados gratuitos geralmente tem menor alcance de dados, limite de chamadas do API, ou menor estabilidade.

Para o projeto em questão foi escolhido o pacote de Python, YFinance, que é open source e utiliza web scraping para coletar dados do site Yahoo Finance (<<https://finance.yahoo.com/>>). Uma insegurança nesse pacote é que se a estrutura do site Yahoo Finance for alterada, há grandes chances dos dados coletados por web scraping sejam errôneos até uma atualização do pacote para a nova estrutura do site. Os dados históricos deste pacote são limitados em 5 anos, considerados inicialmente suficientes para a modelagem nesse projeto.

Uma alternativa para o pacote YFinance seria fazer o download direto dos dados históricos no site do Yahoo Finance, porém isso adicionaria um passo extra na preparação de dados, de adquirir o documento, ler este no Python e transformar na base necessária.

O Pacote do Yahoo deve ser instalado e importado. Os dados históricos são facilmente chamados para um dataframe pela função:

```
df = yf.download(ticker, start, end)
```

O “ticker” é a abreviação que uma empresa assume na bolsa de valores. Por exemplo, no caso da Apple, o ticket é “AAPL”. A entrada “start” é a definição da data inicial dos dados. Como queremos o maior range possível, a data inicial deve ser 5 anos antes da data atual. A entrada “end” é a data final para os dados.

Os dados são então baixados em um *dataframe* e é feita uma análise inicial se os dados estão completos e se aparentam estar corretos. Os dados são coletados apenas nos dias úteis, não considerando sábados, domingos e feriados. Na figura 1 por exemplo é plotado um gráfico dos dados e analisado que não há nenhuma movimentação extrema não esperada de um gráfico temporal de cotação de ações.

Figura 1 – Serie Temporal - Apple

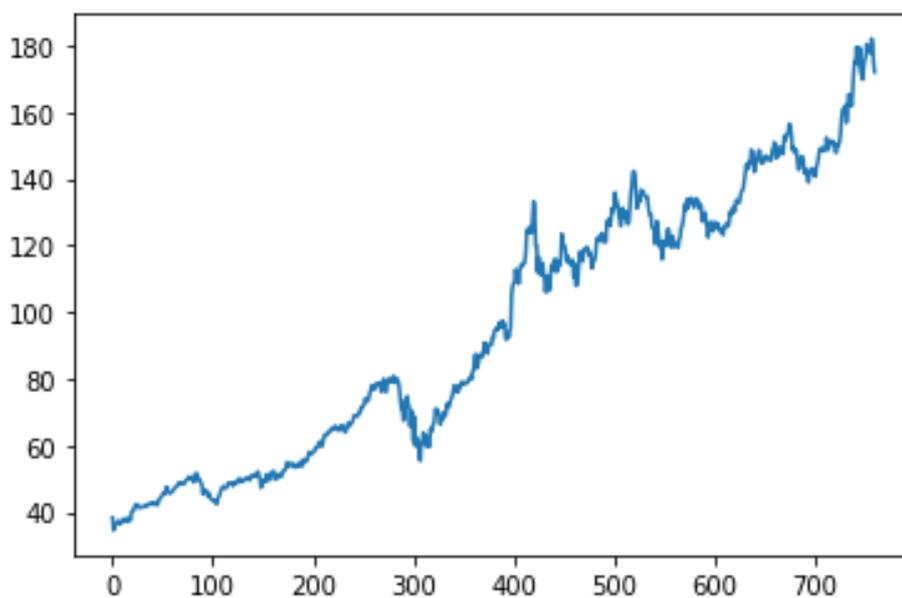


Figura do Autor, dados do [Yahoo](https://finance.yahoo.com/) (2021)

Outra análise que pode ser feita é a diferença entre um ponto e o anterior em porcentagem, e plotar um boxplot (figura 2) dessas diferenças para garantir que nenhum dia esteja com uma diferença extrema. Se houver uma diferença muito extrema pode ser necessária uma análise mais detalhada se o dado está correto ou se algo fora do comum causou a mudança de comportamento extrema. São os chamados “outliers” que podem atrapalhar a análise dos dados, e dependendo de como esses dados serão trabalhados eles devem ser amenizados ou removidos para melhorar a previsão do modelo. Como a série sendo trabalhada possui naturalmente uma grande volatilidade, esses outliers que não são erros dos dados serão mantidos para garantir que sejam considerados na modelagem.

Figura 2 – Boxplot da derivada - Apple

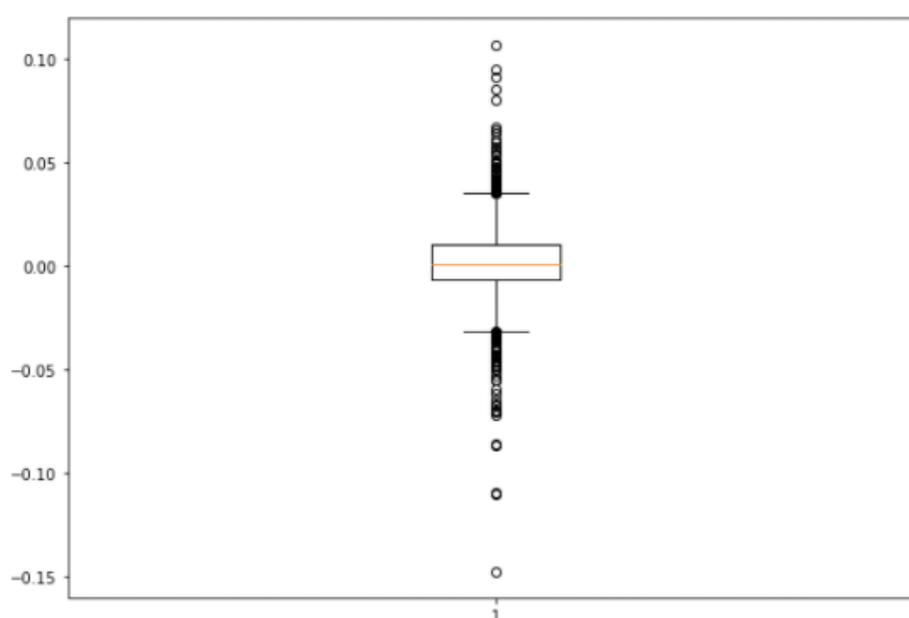


Figura do Autor, dados do [Yahoo](#) (2021)

### 3.3 Séries temporais

Com os dados prontos para a modelagem pode ser feito o próximo processo. Como estamos trabalhando com séries temporais, deve ser feita uma análise da série temporal quando as suas características.

Séries temporais são sequências de dados discretos em ordem cronológica, com o mesmo espaçamento temporal entre cada dado (por exemplo dias), nos quais o dados  $t$  é dependente dos dados anteriores ( $1$  a  $t-1$ ).

As séries temporais apresentam as seguintes propriedades:

- Tendência: A tendência de uma série significa se os valores da série tendem a aumentar,

diminuir ou se manter constante.

- Sazonalidade: Sazonalidade em uma série temporal significa se ela tem uma flutuação periódica. Por exemplo, se as ações de uma empresa que vende mais produtos no verão tendem a subir de cotação, e descer no inverno, mantendo esse ciclo periódico.
- Ciclo: Ciclo é o que define o aumento ou redução da frequência, sendo assim sem intervalos fixos de aumento e redução.
- Erro aleatório ou ruído: São componentes de flutuações inexplicáveis, causados por fatos fora do padrão, como a pandemia e grandes crises.

Para a análise desse atributos pode ser feita uma decomposição sazonal da série temporal, separando a mesma em 3 componentes, a tendência, a sazonalidade e o resíduo, como na figura 3. O modelo pode ser aditivo, onde cada componente é adicionado ou multiplicativo, onde a série é uma multiplicação dos componentes:

- Aditivo: Observação = tendência + sazonalidade + resíduo
- Multiplicativo: Observação = tendência x sazonalidade x resíduo

Figura 3 – Decomposição da serie temporal - Amazon

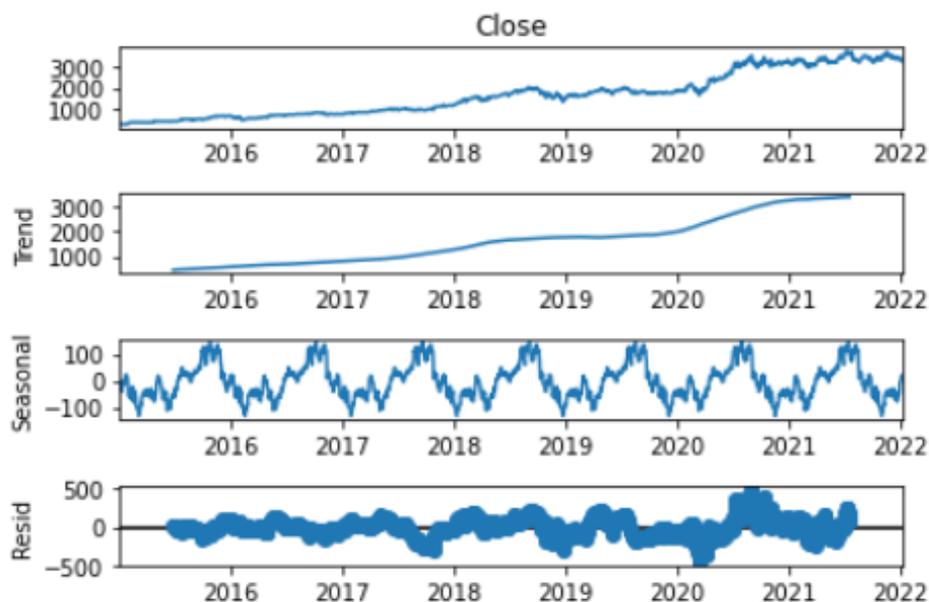


Figura do Autor

Outro conceito de séries temporais é a estacionariedade. Uma série é estacionária se a média, variância e estrutura de autocorrelação se mantêm constante com o tempo. Se a série tem tendência, por exemplo, a série temporal não apresenta estacionariedade. No caso das séries

temporais dos preços de ações, é observado uma não estacionaridade, devido a existência de tendência, com o aumento ou redução da média do preço das ações.

### 3.4 Modelos auto regressivos e média móvel

Uma possibilidade para a predição de séries temporais são os modelos autoregressivos e de média móvel como AR (*Autoregressive model*), MA (*Moving average model*), ARMA (*Autoregressive moving average model*), ARIMA (*Autoregressive Integrated Moving Average Model*) e SARIMA (*Seasonal autoregressive integrated moving average model*).

AR(p) é o modelo autoregressivo simples, que antecipa a dependência do valor futuro nos valores passados e MA(q) é o modelo de media movel, que antecipam a dependência nos erros de predição anteriores. A combinação desses dois modelos se dá no modelo ARMA. O modelo ARMA é expressado pelos componentes p e q, respectivamente da parte autoregressiva e da parte da média móvel.(HYNDMAN; ATHANASOPOULOS, 2018) A primeira parte do modelo ARMA(p,q) é o AR(p):

$$X_t = c + \sum_{i=1}^p \varphi_i X_{t-i} + \varepsilon_t \quad (3.1)$$

A segunda parte do modelo ARMA(p,q) é o modelo MA(q):

$$X_t = \mu + \varepsilon_t + \sum_{i=1}^q \theta_i \varepsilon_{t-i} \quad (3.2)$$

Combinando os dois componentes temos o ARMA(p,q):

$$X_t = c + \varepsilon_t + \sum_{i=1}^p \varphi_i X_{t-i} + \sum_{i=1}^q \theta_i \varepsilon_{t-i} \quad (3.3)$$

Para esse modelo é necessário assumir que a série temporal é estacionária, e a regressão vai falhar no caso de um exemplo não estacionário. Por esse motivo esse modelo não é aplicável para o fechamento de ação, que são séries não estacionárias. Para o caso então é usado o modelo

ARIMA(p,d,q), no qual se usa de diferenciação para transformar a série em estacionária. Aqui entra a componente “d”, definindo quantas vezes as observações originais devem ser derivadas. É também chamado de nível de diferença.

A predição ARIMA(p,d,q), no entanto, não considera componentes de sazonalidade de uma série temporal. No caso por exemplo dos valores de ação de uma empresa que lança produtos a cada 6 meses que tem uma tendência a aumentar a cotação no período do lançamento e descer novamente após um período, tal sazonalidade não seria considerada. Nesse caso deve ser utilizado o modelo SARIMA(p,d,q)x(P,D,Q,s)m, onde m é o fator sazonal do cálculo, e P, D Q são componentes análogos ao p,d,q do ARIMA mas considerando sazonalidade, e s é a período da sazonalidade. (HYNDMAN; ATHANASOPOULOS, 2018)

Para a predição dos valores de cotações de ações, o modelo autorregressivo mais adequado deve ser o SARIMA, e dependendo da existência ou não de sazonalidade o modelo ARIMA pode também ser adequado. Como concluído acima, os modelos AR, MA, ARMA não são adequados devido à hipótese de uma série estacionária, que não condiz com a série desse projeto. Assim, para a comparação foram considerados ambos os modelos ARIMA e SARIMA.

### 3.5 Redes neurais

Para a comparação nesse projeto, além dos modelos auto regressivos foram selecionadas as redes neurais. As redes neurais são modelos de aprendizado de máquina que simulam o funcionamento dos neurônios do cérebro humano, com nós interconectados. Usando algoritmos elas procuram padrões e correlações nos dados de entrada, e procuram aprender e melhorar a precisão continuamente.

Há diferentes modelos de redes neurais, entre eles as convolucionais, as recorrentes, as feedforward, as autoencoders. Para a predição de séries temporais a mais adequada são geralmente as redes neurais recorrentes (RNN).

Para dados não sequenciais as camadas densas das redes neurais convolucionais por exemplo consideram apenas a entrada atual para computar a saída atual. Assim cada iteração é independente da iteração anterior.

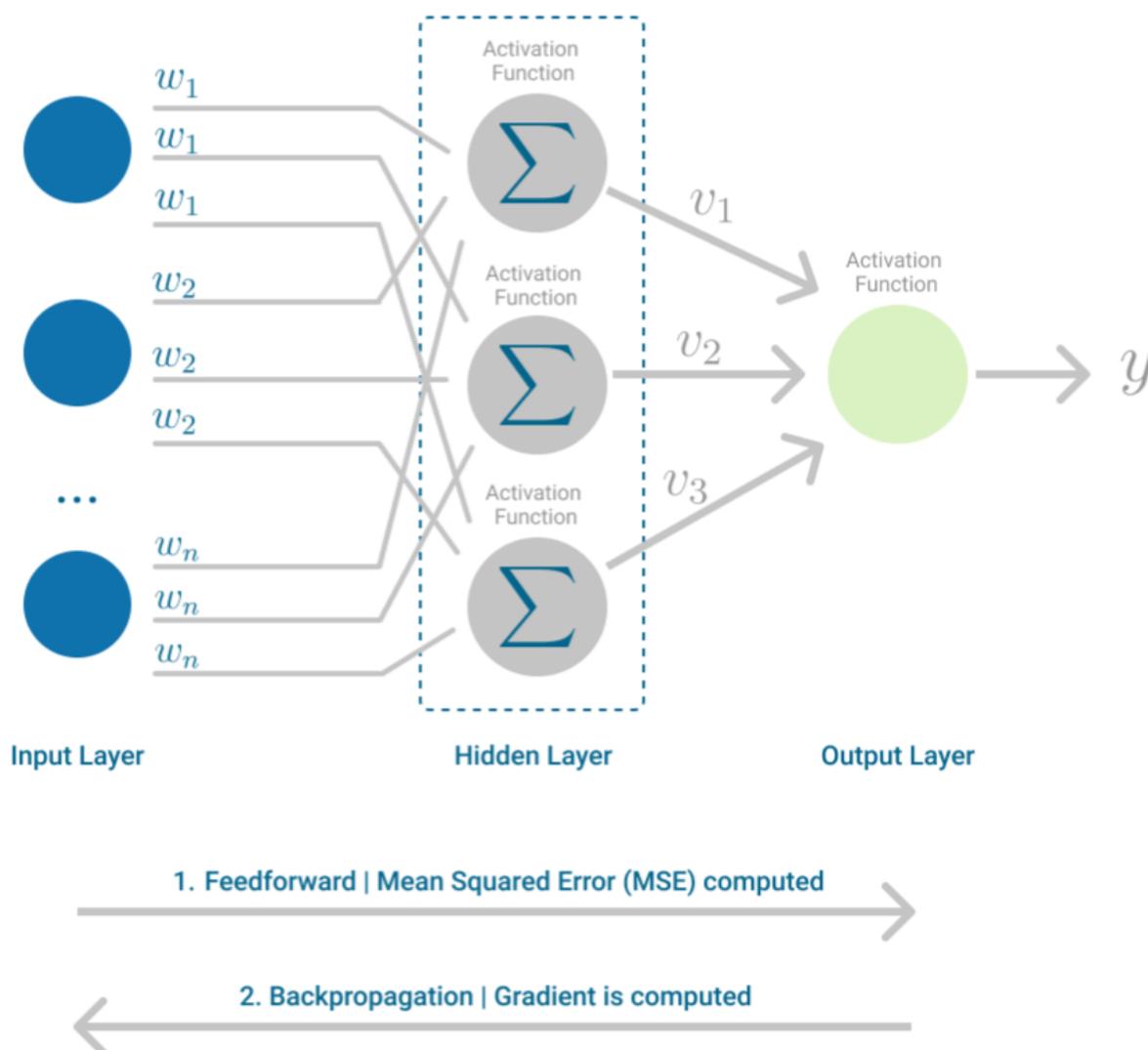
Redes neurais recorrentes por outro lado tem uma “memória”, considerando também dados de iterações anteriores para computar a saída atual. Assim cada iteração é dependente de iterações anteriores. Redes neurais para dados sequenciais podem ter apenas uma entrada para várias saídas, várias entradas para uma saída e várias entradas para várias saídas. Dados sequenciais podem ser por exemplo textos, séries temporais.

Para esse projeto foram testados 3 modelos diferentes de redes neurais: *multilayer perceptron* (MLP), *Long short term memory* (LSTM) e *gated recurrent unit* (GRU).

### 3.5.1 Multilayer Perceptron (MLP)

Multilayer perceptron é um tipo de rede neural profunda feedforward que possui pelo menos 3 camadas: input, output e uma ou mais camadas hidden. Os neurônios têm uma função ativa que define quais dados são passados, como por exemplo Relu (*Rectified Linear Unit*) e *Sigmoid*. Assim, só os dados definidos pela função ativação (*threshold*) conseguem ativar o neurônio. Cada dado então recebe um peso ao entrar no neurônio e são combinados em uma saída  $y$ . Os dados de uma camada são então alimentados para a próxima camada. As redes MLP utilizam *backpropagation* para gerar iterações, de maneira a ajustar iterativamente o peso da rede, de forma a reduzir a função custo. Assim as entrada são multiplicadas por um peso ao entrar nos neurônios, que é ativado ou não e passada para a próxima camada. A figura 4 exemplifica o funcionamento de uma rede MLP. (HAYKIN, 1999)

Figura 4 – MLP Exemplo



Funções ativação podem ser por exemplo *softmax*, *rectifier*, *tanh* e *sigmoid*. A *softmax* é usada para problemas de classificação, garantindo que a soma de todos os resultados seja igual a 1. A *rectifier*, *sigmoid* só deixam valores positivos serem repassados e a função *tanh* limita a saída entre -1 a 1.

A camada mais utilizada é a camada densa, que conecta todos os neurônios de uma camada a outra. Existem também a camada *dropout*, que tem como função transformar parte dos inputs em zero de maneira a reduzir o *overfitting*. Uma terceira camada é a camada *merge*, que combina *input* de múltiplos modelos em um único modelo.

Para a compilação do modelo devem ser definidos alguns atributos, como o otimizador do modelo, a função perda e quais as métricas a serem utilizadas:

1. Otimizador do modelo: Esse atributo é a definição da técnica utilizada para atualizar os pesos do modelo. Otimizadores frequentes são por exemplo o SGD (*Stochastic gradient descent*), o RMSprop (*Root Mean Squared Propagation*) e o Adam (*Adaptive moment estimation*).
2. Função perda: A função perda é a avaliação do modelo que é usada pelo otimizador para encontrar os melhores pesos que reduzem a perda. A mais usada é a MSE (*mean squared error*, ou erro médio quadrado).
3. Métricas do modelo: São as métricas que serão avaliadas pelo modelo durante o treinamento, como por exemplo a acuracidade.

### 3.5.2 Camada recorrente básica (RNN)

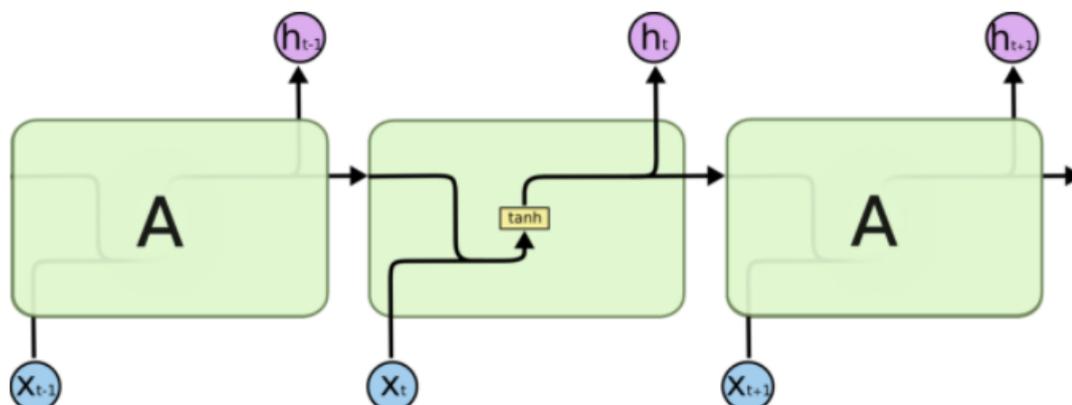
A camada recorrente básica é derivada das redes neurais *feedforward*, como a MLP, mas para dados sequenciais. Nas redes neurais *feedforward* a informação passa por um neurônio uma única vez, de forma que eles não têm memória de estados anteriores. Nas redes neurais recorrentes a informação circula, de forma que o neurônio mantém uma memória do estado anterior. (GERS; SCHRAUDOLPH; SCHMIDHUBER, 2002)

Camada recorrente básica é um tipo de recorrência simples, no qual a saída da iteração anterior é combinado com a nova entrada na nova iteração. Assim se trata de uma memória de curta duração, apenas da entrada anterior. O estado interno é repassado para a própria camada para ser combinado com a entrada para a próxima iteração (“memória”).

A figura 5 é um exemplo de uma camada de RNN básica. A saída  $Y$  é dependente de 2 entradas: a saída  $h(t-1)$  e a entrada  $X(t)$ . O estado interno  $(t)$  alimenta então a próxima iteração nessa camada.

Uma desvantagem desse modelo é que a memória tem curtíssimo prazo. Se a rede neural estivesse lendo uma palavra letra por letra por exemplo, na terceira letra da palavra a rede neural

Figura 5 – RNN Exemplo



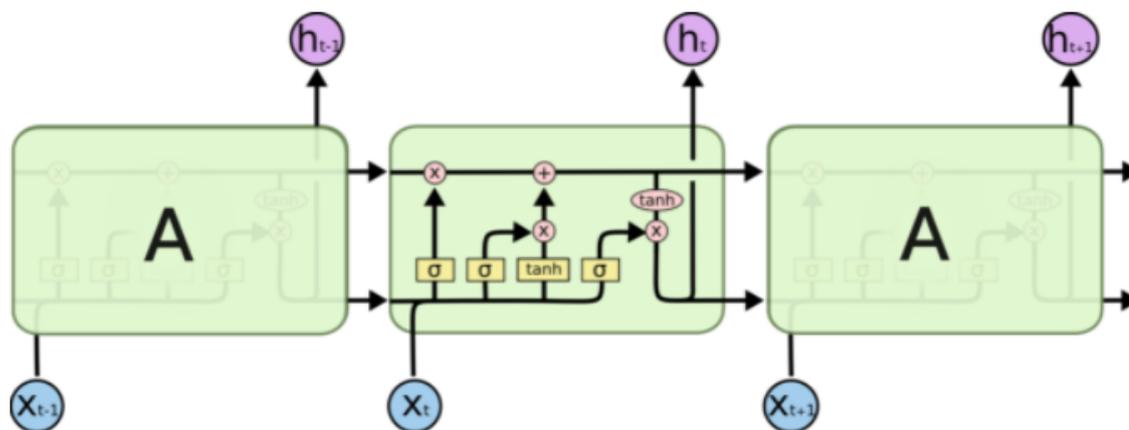
Fonte: Olah (2015)

já teria esquecido a primeira letra, de forma que prever a quarta letra seria muito difícil. Por isso foram criadas versões mais complexas do modelo, como o LSTM.

### 3.5.3 Long short term memory (LSTM)

Long short term memory (LSTM) é um tipo de RNN mais complexa que a camada recorrente básica. Essa rede neural tem uma memória de curto prazo como a RNN básica e uma segunda memória, de longo prazo. A figura 6 ilustra uma camada LSTM.

Figura 6 – LSTM Exemplo



Fonte: Olah (2015)

O *cell state*, ou estado interno é a memória longa da unidade. Essa memória passa por portões (*gates*), que definem quais informações serão repassadas para o próximo passo. LSTMs possuem 3 gates, o primeiro é o *forget gate*, que escolhe o que vai ser esquecido do estado da célula anterior. O valor é multiplicado por um número de 0 a 1, sendo 0 para

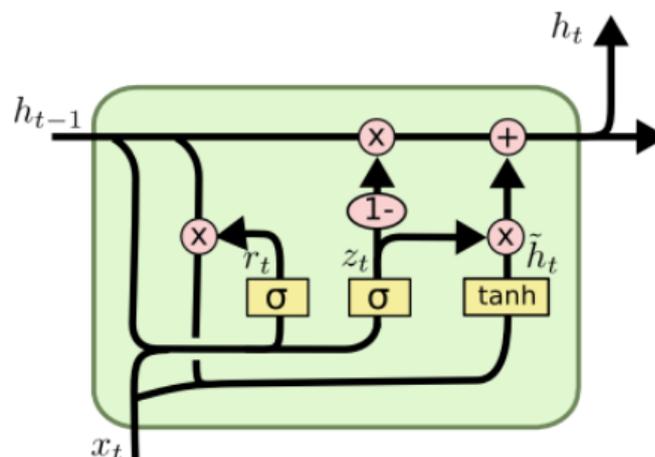
esquecer completamente a memória e 1 para manter completamente. (GERS; SCHRAUDOLPH; SCHMIDHUBER, 2002)

O próximo passo é o *input gate*, que define quais valores serão atualizados na memória, sendo seu resultado somado ao valor mantido pelo *forget gate*. Por fim tem o *output gate*, que define qual é o resultado da célula. Assim, a rede apresenta não só a memória de curto prazo, mas a de longo prazo, possibilitando maiores alcances da correlação. Com o *forget gate* ela pode definir quais memórias devem ser mantidas ou não dependendo da correlação dos dados atuais com dados anteriores.

### 3.5.4 Gated recurrent unit (GRU)

*Gated recurrent Unit* (GRU) é uma versão simplificada do LSTM, que combina o *input gate* e o *forget gate* em um único gate de atualização, *update gate* (figura 7). Assim, enquanto LSTM possui dois estados diferentes passados entre células (estado da célula e estado oculto), o GRU transfere apenas o estado oculto, que mantém as memórias de longo e curto prazo ao mesmo tempo. (CHUNG *et al.*, 2014)

Figura 7 – GRU Exemplo



Fonte: Olah (2015)

---

## RESULTADOS

---

Para a predição dos preços de ações foram testados 4 modelos diferentes: SARIMA, MLP, LSTM e GRU, usando como base as séries temporais dos valores de fechamento diário de cotação de ações. Para as séries temporais foram selecionadas 3 diferentes ações: Amazon, Apple e Google.

Os dados foram separados em 70 por cento dados de treinamento e 30 por cento dados teste como no exemplo na figura 8. Esses dados foram normalizados utilizando uma função min-max entre 0 e 1, e não foi considerado nenhum ruído.

Figura 8 – Series normalizada e dividida em treinamento e teste

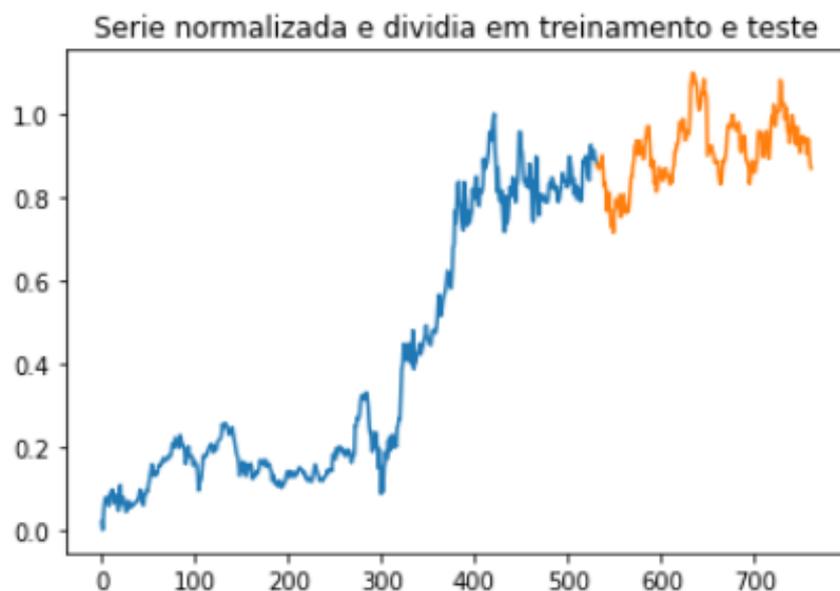


Figura do autor

## 4.1 Arima e Sarima

Foram feitas decomposições aditivas nas séries, de forma a analisar a presença de tendência e sazonalidade. Por isso o modelo arima foi rejeitado por não considerar sazonalidade. Para a compilação Sarima, foram considerados os modelos auto arima, com  $m = 5$  (dias úteis em uma semana). Para ações da Amazon o resultado dos atributos do auto arima foi: SARIMA(1, 1, 1)x(2, 0, 1, 5).

Com a separação inicial de 70/30 dados de treinamento e teste, os resultados da edição Sarima foram péssimos, mantendo constantes em torno de 2000, enquanto os dados teste aumentaram repentinamente para acima de 3000 (figura 9). O modelo SARIMA não foi capaz de prever esse aumento repentino, por não seguir nenhuma sazonalidade ou tendencia vista previamente nos dados treinamento. Para comparação, foi feito um segundo teste com uma separação treinamento/teste de 80/20 por cento (figura 10). A acuracidade do resultado melhorou, mas de maneira artificial ao ajustar a divisão dos dados. Mesmo com essa melhora na acuracidade, os resultados obtidos são praticamente constantes, indicando que a presença de qualquer aumento ou queda do valor a acuracidade irá diminuir significativamente. Concluí-se então que com os atributos escolhidos esse método não é adequado para a predição dos dados de fechamento da bolsa. Uma possibilidade para melhorar a projeção seria encontrar uma melhor frequência sazonal para o modelo. Para essa predição foi utilizado 5 dias (dias úteis em uma semana).

Figura 9 – Predição Sarima com separação 70/30 nos dados teste

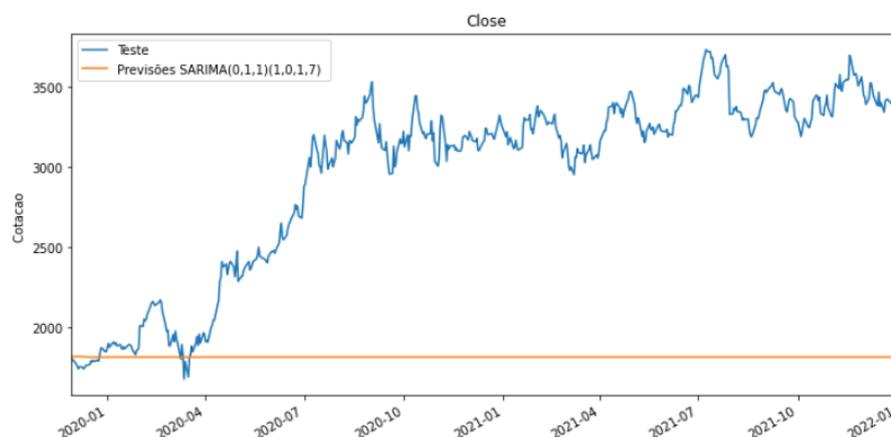


Figura do autor

## 4.2 MLP

Para a compilação do MLP, o atraso entre a entrada e a primeira saída foi considerado de apenas 1. É importante considerar apenas 1 atraso pois ao considerar 2 ou mais, a rede MLP não garante a sequencialidade desses dados ao processá-los, de forma que a única maneira de manter

Figura 10 – Predição Sarima com separação 80/20 nos dados teste

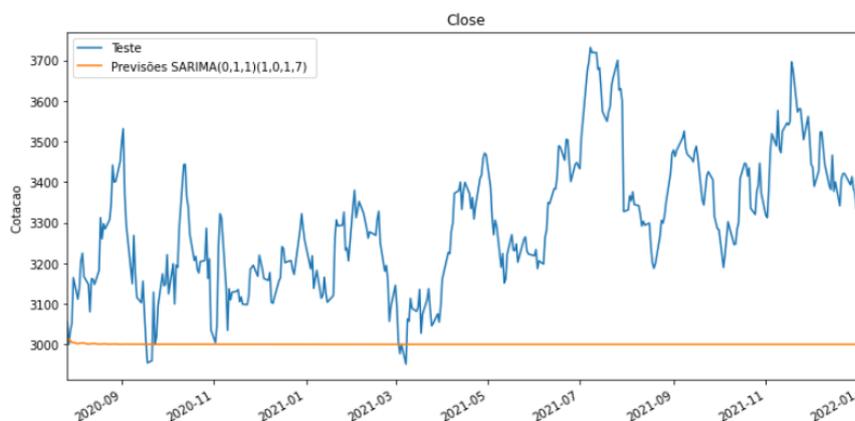


Figura do autor

a consideração de dados sequenciais é ter um atraso de apenas 1 passo. Assim, para a predição da cotação de um dia é usado apenas como input a cotação do dia anterior.

Para as camadas foram utilizadas uma camada entrada padrão, 2 camadas densas com 32 neurônios e ativação *relu*, uma camada *dropout* de 0.2 e uma camada densa final com 1 neurônio e ativação *relu* (combinando os resultados).

Para a compilação a função perda escolhida foi o erro médio quadrático, a função otimizadora Adam com uma taxa de aprendizado de 0.0001 e métricas *mae* (erro médio absoluto). Para o treinamento do modelo foi usado apenas um *batch* devido a ser uma série temporal com dependência sequencial de dados, 80 épocas. Os dados de treinamento convergiram rapidamente em torno de 10 épocas para os 3 bancos de dados.

## 4.3 LSTM e GRU

Para a modelagem de LSTM e GRU foi utilizado um atraso de 2 dias. Para LSTM foram utilizadas 2 camadas LSTM de 32 neurônios, 1 camada *dropout* de 0,2 e uma camada densa de 1 neurônio para a combinação dos dados, e para a GRU extremamente semelhante, mas 2 camadas GRU no lugar de LSTM. A modelagem LSTM convergiu rapidamente nas primeiras 10 épocas, enquanto a modelagem GRU precisou de até 20 épocas para convergir.

Assim como no modelo MLP, para função perda foi considerado o erro quadrático médio, otimizador adam com taxa de aprendizado 0.0001, métricas *mae* (erro absoluto médio) e 80 épocas.

## 4.4 Comparação de resultados

Os três modelos apresentaram uma boa acuracidade para a maioria dos dados (aqui mostrado Apple, Amazon e Google), com erro no teste entre 0.0009 e 0.003, com exceção de LSTM para as ações da Google que tiveram um grande erro. Em primeira comparação, as redes neurais LSTM e GRU apresentam levemente maior acuracidade que MLP na maior parte dos casos. Porém dependendo da base de dados há uma variação de qual método tem o melhor desempenho (por exemplo LSTM falhando na predição da empresa Google).

Erro no teste com predição a cada ponto para ações da Apple (AAPL) (figura 11):

MLP: 0.00158

LSTM: 0.00097

GRU: 0.00141

Figura 11 – Resultados de predição para ações da Apple



Figura do autor

Erro no teste com predição a cada ponto para ações da Amazon (AMZN) (figura 12):

MLP: 0.00143

LSTM: 0.00120

GRU: 0.00117

Erro no teste com predição a cada ponto para ações da Google (GOOG) (figura 13):

MLP: 0.00259

LSTM: 0.14156

Figura 12 – Resultados de predição para ações da Amazon

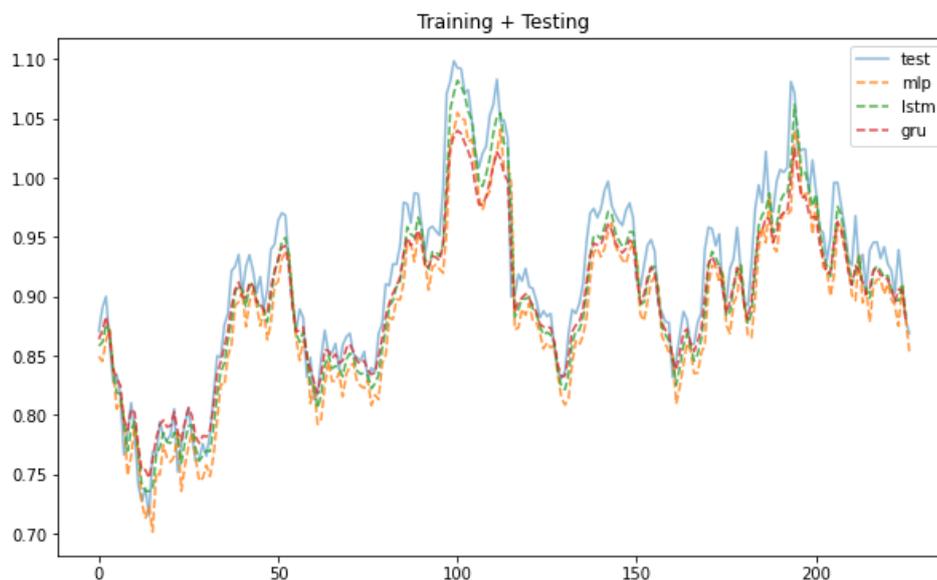


Figura do autor

GRU: 0.00156

Figura 13 – Resultados de predição para ações da Google



Figura do autor

Em uma primeira análise visual e de acuracidade todos os modelos parecem adequados para a predição do valor de ação. Porém, para um investidor a informação importante é se a cotação vai aumentar ou diminuir de tamanho. Ao analisar a diferença entre um dia e o anterior, podemos ver que as predições reagem à mudança de direção do gráfico com pelo menos 1 dia de atraso, ou seja, só quando a ação já começa a descer que as predições também predizem a queda.

Dessa maneira, por mais precisas que essas previsões sejam em erro médio absoluto, elas não atingem o propósito de prever o aumento ou a redução da ação. Uma tentativa para melhorar o modelo foi a adição de mais camadas de *dropout*, porém o efeito desejado não foi atingido. Na imagem 14 pode-se ver a comparação de inclinação dos gráficos para as primeiras 20 previsões. Nessa inclinação o atraso de um ou mais dias na resposta a mudanças de inclinação é visível .

Figura 14 – Comparação de resultados da Amazon considerando inclinação das curvas



Figura do autor

---

## CONCLUSÃO

---

Nesse trabalho foram analisados os métodos SARIMA, MLP, LSTM e GRU para a predição de séries temporais de valores de fechamento diários de ações. As séries temporais foram analisadas contendo tendência e sazonalidade, os dados foram separados em 70/30 para treinamento e teste.

Devido a volatilidade da série temporal, o método SARIMA não foi adequado para as predições, obtendo resultados quase constantes e não condizentes com a realidade. Os métodos de redes neurais MLP, LSTM e GRU tiveram em sua maioria resultados próximos aos dados testes com alta acuracidade, porém atrasados em um ou dois dias. Devido a esse atraso, essa predição não é adequados para o acompanhamento diária de acoes, já que não prediz corretamente se o valor vai subir ou descer no dia seguinte. Ao mesmo tempo as predições escolhidas só predizem um dia no futuro, de forma que não são adequadas para investimentos a longo prazo.

A volatilidade dos dados temporais de ações impossibilita uma boa predição utilizando métodos simples. Apesar de redes neurais terem uma boa predição em acuracidade, ela falha onde realmente é importante: a predição dia a dia se o valor da ação sobe ou desce. Uma possibilidade para melhoria seria trabalhar não com a série temporal original, mas a derivada da mesma. Assim a predição não focaria no valor da ação, mas a curvatura da série temporal original.

A escolha de redes neurais nesse projeto limitou a predição a apenas o valor do próximo dia, e não as próximas semanas ou meses. Foi utilizado 2 dias (e memórias anteriores) para a predição apenas do dia seguinte. Uma mudança a ser analisada no próximo trabalho seria a predição de vários dias baseados em mais dias anteriores. Assim a série talvez dê melhores resultados de subida ou queda de valor em um mais longo prazo, do que uma série de alta granularidade como a feita neste projeto.

A predição Sarima foi limitada pela volatilidade da série temporal, de forma que resultou em baixa acuracidade e nenhum detalhamento real de valores dia-a-dia. Com o método Sarima também não se obteve sucesso nem com a predição se a ação subira de valor subira, descera

ou se mantivera constante. Por outro lado as 3 séries temporais escolhidas para análise não foram adequadas para testar diferentes cenários. Todas as séries temporais escolhidas possuem uma tendência de aumentar de valor, e nenhuma possui um pico e queda a longo prazo, por exemplo, o que poderia mudar a precisão da predição. Assim os resultados Sarima podem não ser adequados, se forem utilizadas outras ações com tendências variadas. Uma possível melhora do modelo sarima é a melhor escolha do componente sazonal.

Assim, a volatilidade das cotações de ações, causadas por fatores externos à série temporal, faz com que falte informação aos modelos baseados apenas na série temporal. Assim, modelos mais misturados com outras fontes de dados, como por exemplo notícias, redes sociais, dados financeiros de cada quartel da empresa tem uma expectativa de maior acuracidade na predição. Motivo disso é a inclusão de dados que afetam a cotação, e não apenas o histórico da mesma. Porém a maior dificuldade de tal análise seria a coleta adequada desses dados, considerando que não há uma única fonte de dados prontos. Aqui poderia ser utilizado por exemplo *web scrapping* em diferentes sites para a coleta de notícias, incluindo leitura de imagens e gráficos, devendo ser também coletada a data de cada notícia ou acontecimento. Por não serem dados prontos haverá também o risco de serem coletadas notícias sobre empresa B como notícia da empresa A, apenas pelo fato do nome da empresa A ter sido brevemente mencionado na notícia. Isso pode criar erros no modelo, considerando notícias atribuídas a empresa errada e assim afetar a acuracidade das predições. Esses dados então teriam que ser combinados no modelo, de forma a unir a predição da série temporal e os acontecimentos noticiados para uma predição com mais entradas.

## REFERÊNCIAS

---

---

ADEBIYI, A. A.; ADEWUMI, A. O.; AYO, C. K. **Stock Price Prediction Using the ARIMA Model. 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, 2014.** 2014. Citado na página 23.

BARAK, S.; ARJMAND, A.; ORTOBELLI, S. Fusion of multiplediverse predictors in stock market. **Information Fusion**, v. 36, 2017. Citado na página 22.

BENTO, C. **Multilayer Perceptron Explained with a Real-Life Example and Python Code: Sentiment Analysis.** 2021. Disponível em: <<https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee931>>. Acesso em: 20/12/2021. Citado na página 31.

BLEESSER, W.; LIICOFF, P. Predicting stock returns with bayesian vector autoregressive. **Data Analysis, Machine Learning and Applications**, v. 1, p. 499–506, 2005. Citado na página 23.

CHUNG, J.; GULCEHRE, C.; CHO, K.; BENGIO, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. 12 2014. Citado na página 34.

GERS, F.; SCHRAUDOLPH, N.; SCHMIDHUBER, J. Learning precise timing with lstm recurrent networks. **Journal of Machine Learning Research**, v. 3, p. 115–143, 01 2002. Citado nas páginas 32 e 34.

GEVA, T.; ZAHAVI, J. Empirical evaluation of an autmaated intraday stock recommendation system incorporating both market data and textual news. **Decision Support Systems**, v. 57, 2014. Citado na página 22.

GRAHAM, B.; DODD, D. **Security Analysis: Sixth Edition, Foreword by Warren Buffett -**. New York: McGraw-Hill, 2008. ISBN 978-0-071-59253-6. Citado na página 17.

HAYKIN, S. **Neural Networks: A Comprehensive Foundation.** Prentice Hall, 1999. (International edition). ISBN 9780139083853. Disponível em: <<https://books.google.de/books?id=M5abQgAACAAJ>>. Citado na página 31.

HYNDMAN, R.; ATHANASOPOULOS, G. **Forecasting: Principles and Practice.** 2nd. ed. Australia: OTexts, 2018. Citado nas páginas 29 e 30.

JUNG, C.; BOYD, R. Forecasting uk stock prices. **Applied Financial Economics**, Routledge, v. 6, n. 3, p. 279–286, 1996. Disponível em: <<https://doi.org/10.1080/096031096334303>>. Citado na página 23.

LI, X.; HUANG, X.; DENG, X.; ZHU, S. Enhanced quantitative intra-day stock return prediction by intergrating both market news and stock prices informtation. **Neurocomputing**, v. 142, 2014. Citado na página 22.

OLAH, C. **Understanding LSTM Networks.** 2015. Disponível em: <<http://colah.github.io/posts/2015-08-Understanding-LSTMs/>>. Acesso em: 20/11/2021. Citado nas páginas 33 e 34.

SEONG, N.; NAM, K. Predicting stock movements based on financial news with segmentation. **Expert Systems With Application**, v. 164, 2021. Citado nas páginas 21 e 22.

YAHOO. **Yahoo Finance**. 2021. Disponível em: <<https://finance.yahoo.com/>>. Acesso em: 20/11/2021. Citado nas páginas 26 e 27.

ZHANG, C.; CHENG, X.-j.; WANG, M. An empirical research in the stock market of shanghai by garch model. **Operations Research and Management Science**, v. 4, p. 144–146, 2005. Citado na página 23.

