

Insper

Victor Hugo Leal Vieira da Silva
Vinicius Bonelli Falco

STOCK MARKET PREDICION

MACHINE LEARNING

São Paulo

2020

Sumário

1. Contexto.....	3
2. Metodologia	3
3. Dataset.....	4
4. Métricas	5
5. Regressão.....	6
6. Rede Neural Recorrente	8
7. Resultados	9
8. Conclusão	14
9. Referências.....	15

1. Contexto

Nos últimos anos, temos visto uma tendência global de corte da taxa básica de juros. Esses cortes contínuos impactam diretamente o rendimento de títulos de renda fixa associados a essa taxa. No caso do Brasil, a principal aplicação dos brasileiros, a poupança, sofreu um forte impacto devido aos cortes da SELIC. O rendimento da poupança foi de 8,3% a.a.^[1] em 2016, para uma projeção de 1,57% a.a.^[2] em 2020.

O impacto direto no bolso dos brasileiros fez com que muitos migrassem da renda fixa para a renda variável em busca de retornos maiores. O número de CPFs cadastrados na bolsa brasileira subiu vertiginosamente desde 2017^[3], momento em que os cortes começaram.

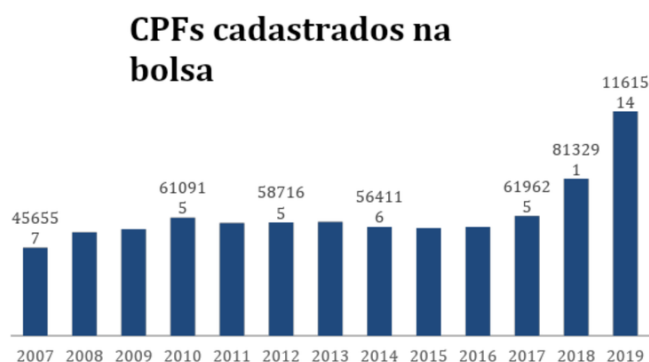


Figura 1 - Evolução dos CPFs cadastrados na bolsa

Com o maior interesse em renda variável, associado a um mercado que veio desde 2015 em uma forte retomada do movimento de alta, surgem diversas pessoas atrás da “fórmula mágica de ganhar dinheiro na bolsa”. Assim, a motivação do trabalho a seguir é averiguar se é possível desenvolver uma estratégia rentável de trading utilizando técnicas clássicas e modernas de Machine Learning.

Apenas um *disclaimer* antes de iniciar: o documento a seguir tem caráter educacional e não deve ser levado como uma recomendação de investimento. As técnicas mostradas aqui devem ser utilizadas com cautela e os riscos devem ser ponderados.

2. Metodologia

O objetivo do projeto foi definido como sendo o seguinte:

“Encontrar um modelo que faça previsões consistentes do preço de fechamento do WIN (Mini-Índice Futuro) baseado em indicadores de eventos contínuos e discretos. O modelo deve acertar o sentido do fechamento mais de 70% das vezes e deve ter expectativa matemática > 1.”

Com o objetivo definido, a primeira coisa a se fazer antes de se iniciar o projeto é pensar em quais as etapas a serem cumpridas e o que se espera atingir em cada uma delas. O fluxo que foi seguido nesse projeto foi o seguinte:

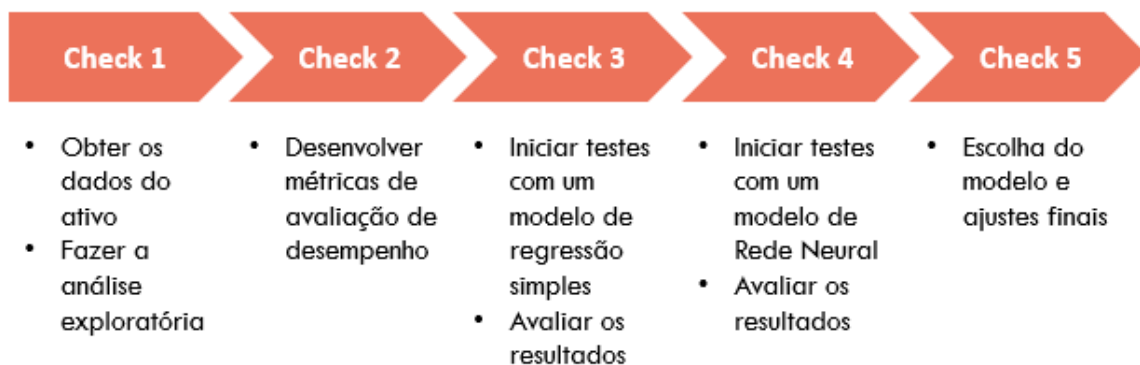


Figura 2 - Fluxo de trabalho

Entendendo o objetivo de cada Check:

- O objetivo do Check 1 era conceber uma base de dados sólida para ser utilizada ao longo do projeto. Isso inclui obter os dados da variável target, o preço de fechamento do ativo, e das features que seriam analisadas. Inclui também tratar possíveis erros na base, como dados faltantes, e analisar as features para identificar possibilidades de desenvolver novas, através de modificação.
- Na etapa seguinte, as métricas de avaliação do modelo de Machine Learning seriam desenvolvidas. Os modelos podem ser analisados sob a ótica das métricas tradicionais de ML, entretanto nesse projeto é interessante também introduzir métricas voltadas para a aplicação do modelo no mercado financeiro, para facilitar o entendimento dos resultados.
- Com os dados e as métricas, a terceira etapa envolve desenvolver um modelo básico de regressão. Apesar de vários estudos e teorias mostrarem que um modelo de regressão não é suficiente para fazer previsões do mercado, devido ao comportamento do mesmo, é interessante para fins educacionais mostrar o porquê o modelo básico não funciona antes de avançar para algo mais complexo.
- Após o modelo de regressão, foram utilizadas técnicas de machine learning modernas para desenvolver um modelo de rede neural. Nesse meio existe um debate sobre quão efetiva pode ser a aplicação dessas técnicas, com alguns trabalhos que mostram resultados excelentes e outros que mostram resultados medianos.
- Por fim, a etapa final envolve os ajustes finais e a discussão do modelo que proporcionou os melhores resultados.

Todo o código desenvolvido ao longo dessas etapas pode ser encontrado em: <https://github.com/Fattson/StockMarketPrediction-ML>

Para criar o dataset a ser manipulado no projeto foi utilizada a ferramenta ProfitChart. Essa ferramenta é uma plataforma de negociação que disponibiliza dados históricos de preço e de indicadores de todos os ativos negociados na bolsa brasileira para backtests.

A primeira decisão a ser tomada na criação do dataset seria a escolha do ativo a ser estudado. Baseado na liquidez, no interesse e no grande volume financeiro que o Mini Índice Futuro (WIN) proporciona diariamente, ele foi o ativo escolhido para o estudo. As informações de preço foram obtidas a partir de um gráfico de candlesticks com intervalo de 5 minutos entre candles. Esse gráfico é interessante pois ao contrário de um gráfico de linha que mostra apenas um preço, os candles mostram 4 preços: abertura, máxima, mínima e fechamento do intervalo. O fechamento é a variável target do modelo, e as outras três informações foram utilizadas como features.

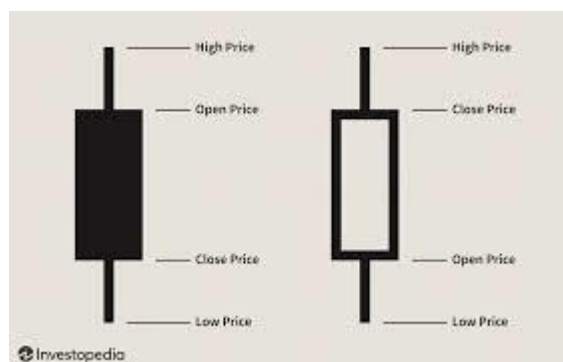


Figura 3 - Informações do Candle. Fonte: [4]

Além dos dados de preço, também foram obtidos alguns indicadores derivados de preço e do volume, como médias móveis, IFR, VWAP e as Bandas de Bollinger. Esses indicadores todos ajudaram a descrever e a prover mais informações sobre os eventos contínuos do preço.

Para lidar com eventos discretos que impactam o mercado, como por exemplo o anúncio de algo importante do governo, um ataque terrorista, ou uma pandemia, existem algumas opções de indicadores. Algo que poderia ser feito seria consolidar notícias e tweets do período analisado e fazer uma análise dos tópicos mais comentados no período e que podem ter alguma influência no mercado. Entretanto essa é uma rota complexa e que exige toda uma análise de tópicos que poderia ser considerada um projeto de ML a parte^[5].

A escolha do indicador então foi baseada em informações já prontas para uso do book de ofertas. O racional por trás da escolha é de que todos os eventos discretos que impactam o mercado afetam diretamente a liquidez dos ativos por causar medo nos participantes. Assim, se observarmos a quantidade de ofertas disponíveis no book podemos desenvolver um termômetro de aptidão a risco dos participantes e identificar possíveis eventos extraordinários.

Todos esses dados consolidados formaram uma base de mais de 12.000 entradas e 14 features. Essa base abrange o período de novembro de 2019 até junho de 2020, que foi considerado um período interessante para a análise pois capta diversos momentos do mercado: desde a euforia de alta no fim de 2019, até uma lateralização no início do ano, a crise da pandemia do COVID-19 e o início da recuperação do mercado a partir de abril. Entretanto, ela é limitada no range de preços que pode atuar, pois não existem na base dados acima da máxima e nem abaixo da mínima para treinarmos o modelo e podermos garantir as métricas de performance fora desse range.

	Abertura	Máxima	Minima	Fechamento	Média Móvel A [20]	Média Móvel E [9]	Média Móvel A [50]	Bandas de Bollinger A [20]	Bandas de Bollinger A [20].1	VWAP D	VWAP Band	Book	IFR (RSI) [9]	Spread	Varição do ativo
0	96510.0	96620.0	96490.0	96570.0	96794.75	96621.73	96824.2	97186.28	96403.22	96533.25	96855.00	9467	39.15	-176.55	NaN
1	96555.0	96570.0	96405.0	96510.0	96807.75	96634.66	96827.3	97185.59	96429.91	96533.13	96854.87	13037	33.18	-236.55	-60.0
2	96610.0	96655.0	96535.0	96560.0	96823.75	96665.83	96832.4	97176.04	96471.46	96533.30	96855.04	8271	35.78	-186.55	50.0
3	96575.0	96695.0	96530.0	96605.0	96837.25	96692.28	96836.3	97168.12	96506.38	96533.14	96854.89	13216	38.18	-141.55	45.0
4	96585.0	96675.0	96555.0	96570.0	96854.75	96714.10	96839.0	97171.35	96538.15	96532.76	96854.51	13019	35.18	-176.55	-35.0

Figura 4 - Cabeçalho do dataset

3. Métricas

Conforme mencionado no tópico de Metodologia, para esse projeto foram criadas algumas métricas de avaliação de performance baseadas nas métricas comuns de avaliação de estratégias de trading. Além das métricas clássicas de modelos de regressão de ML, como RMSE e MSE, foram criadas então: Taxa de Acerto, P/L, Payoff e Expectativa Matemática^[6].

A métrica de Taxa de Acerto serve para nos indicar qual a porcentagem das vezes que o modelo acertou a direção do preço fechamento. Se o dado de validação indica que o fechamento foi maior que o fechamento anterior, e a previsão do modelo indica o mesmo, a direção de ambos foi positiva e o modelo acertou. Se o dado de validação indica que o fechamento foi abaixo do fechamento anterior, e o modelo indica o mesmo, a direção de ambos foi negativa e o modelo acertou. Se houver divergência de sentido entre validação e predição, o modelo errou.

O P/L, também conhecido como Profit/Loss, serve justamente para indicar qual o resultado financeiro da estratégia. Assim, sabendo se o modelo acertou a direção do fechamento com a métrica anterior, é possível computar qual seria o resultado financeiro obtido para o intervalo de tempo até o próximo fechamento.

O Payoff é uma métrica muito utilizada em conjunto com a Taxa de Acerto para indicar a rentabilidade da estratégia. Essa métrica, que é calculada pela razão entre a média dos ganhos e a média das perdas, basicamente informa para 1 de perda, qual o valor de ganho. Por exemplo:

- Para uma estratégia com taxa de acerto de 50% ter $P/L = 0$ o payoff precisa ser de no mínimo 1, pois para cada 1 de prejuízo a estratégia consegue 1 de lucro.
- Para uma estratégia com taxa de acerto de 20% ter $P/L = 0$ o payoff precisa ser de no mínimo 4, pois para cada 1 de prejuízo a estratégia consegue 4 de lucro.
- Para uma estratégia com taxa de acerto de 80% ter $P/L = 0$ o payoff precisa ser de no mínimo 0,25, pois para cada 1 de prejuízo a estratégia consegue 0,25 de lucro.

Finalmente, a Expectativa Matemática é uma forma de abstrair os resultados para o futuro e ter qual a expectativa de retorno do seu investimento se as médias dos resultados observadas se manterem. Assim, espera-se um resultado sempre maior que 1 para que a estratégia seja considerada positiva.

4. Regressão

Como já mencionado anteriormente, o primeiro modelo que seria desenvolvido buscando resolver o desafio da predição de preços do WIN seria um modelo de regressão. O funcionamento desse tipo de modelo é razoavelmente simples de ser entendido:

- Utiliza-se os dados das features para estimar o valor da variável target.
- A estimativa é feita baseada em uma equação, onde cada coeficiente representa o peso que cada feature possui no resultado da variável target.
- Quando o dataset é muito grande, torna-se praticamente impossível encontrar uma combinação de pesos que resulte nos valores corretos da variável target.
- Assim, uma forma de identificar o melhor conjunto de coeficientes é medir o erro entre o valor real da variável target e o valor estimado através da equação.

A primeira coisa a se fazer antes de iniciar um modelo de regressão é analisar os dados em busca de correlações. Afinal, se não houver correlações entre as features e a variável target será difícil estimar os coeficientes. Esse é um dos pontos mais complexos da utilização desse tipo de modelo na aplicação de mercado financeiro, pois as variáveis na sua maioria não possuem grande correlação com a variação dos preços. Para o dataset utilizado, os resultados das correlações foram os seguintes:

```
correlation_matrix = market.corr()
correlation_matrix["Variação do ativo"].sort_values(ascending=False)
```

Out[10]:

Variação do ativo	1.000000
IFR (RSI) [9]	0.363462
Spread	0.064196
Fechamento	-0.004485
Mínima	-0.010103
Máxima	-0.010271
Média Móvel E [9]	-0.014040
index	-0.015521
Abertura	-0.015951
Bandas de Bollinger A [20]	-0.016007
Média Móvel A [20]	-0.016325
Bandas de Bollinger A [20].1	-0.016603
Média Móvel A [50]	-0.016617
VWAP Band	-0.016680
VWAP D	-0.016680
Book	-0.017877

Figura 5 - Resultados do primeiro teste de correlação

Ao observar esse resultado, uma série de tentativas de modificações foram feitas nas features visando melhorar as correlações com o preço de fechamento. Mas os resultados dessas modificações não foram promissores, como pode ser visto a seguir:

```
In [54]: correlation_matrix = market.corr()
correlation_matrix["Variação do ativo"].sort_values(ascending=False)

Out[54]: Variação do ativo      1.000000
Inclinação ME9      0.428750
IFR (RSI) [9]      0.363462
Inclinação M20      0.158585
Inclinação VWAP      0.157589
Inclinação M50      0.102642
Spread      0.064196
Fechamento      -0.004485
Mínima      -0.010103
Máxima      -0.010271
Média Móvel E [9]      -0.014040
index      -0.015521
Abertura      -0.015951
Bandas de Bollinger A [20]      -0.016007
Média Móvel A [20]      -0.016325
Bandas de Bollinger A [20].1      -0.016603
Média Móvel A [50]      -0.016617
VWAP Band      -0.016680
VWAP D      -0.016680
Book      -0.017877
Inclinação Book      -0.017985
```

Figura 6 - Resultados do segundo teste de correlação, com features modificadas

Mesmo com as indições pouco promissoras de que o resultado de um modelo de regressão seria bom, aplicou-se aos dados de treinamento um dos algoritmos comuns de regressão em ML: o RandomForestRegressor. Além das métricas descritas na secção 4, também foi utilizado o RMSE para analisar os resultados. Os resultados desse modelo são discutidos em maior profundidade na secção 7.1.

Após observar os resultados iniciais do modelo, foi feita uma otimização de hiperparâmetros utilizando o GridSearch, com a expectativa de que os resultados melhorassem comparado ao modelo inicial. Os valores obtidos para esse segundo caso também são discutidos mais a fundo na secção 7.1.

5. Rede Neural Recorrente

A dinâmica de um ativo, claramente é contínua. Partindo desse fato, a escolha de implementar um modelo com Rede Neural Recorrente (RNN) é uma escolha pertinente. A RNN tem como característica o reconhecimento a padrões de dados contínuos, como textos, imagens, falas e mercado de ativos.

As RNNs possuem algumas variações com algumas diferenças, para o modelo implementado, foi testado com SimpleRNN, Gated Recurrent Units (GRU) e, por fim, Long Short-Term Memory Units (LSTMs). Durante os testes, a implementação do modelo com LSTMs se mostrou como a de melhor resultado, por tanto, será desse modelo que será tratado nessa secção.

Foram realizadas diversas pesquisas com a finalidade de entender qual a melhor forma de distribuir as camadas (layers) e os neurônios para a ideiação do modelo de RNN. Entretanto, muitos dos artigos encontrados referentes a otimização da escolha desses componentes envolvia uma extensa e dificultosa utilização de álgebra matemática. Assim sendo, foi utilizado o método incremental: iniciar o modelo com poucas camadas e neurônios e ir aumentando ambos até que o modelo esteja satisfatório.

Layer (type)	Output Shape	Param #
lstm_8 (LSTM)	(None, None, 20)	1760
lstm_9 (LSTM)	(None, None, 20)	3280
lstm_10 (LSTM)	(None, None, 40)	9760
lstm_11 (LSTM)	(None, None, 80)	38720
dense_2 (Dense)	(None, None, 1)	81
Total params: 53,601		
Trainable params: 53,601		
Non-trainable params: 0		

Figura 7 - Parâmetros da LSTM utilizada

A melhor combinação encontrada para o modelo é a ilustrada na figura 7.

Para completar a compilação do modelo, deve ser definido a função de erro a ser utilizada e o optimizer adequado. Assim, foi utilizado o “MSE” como função de erro e para optimizer foram testados o Stochastic Gradient Descent (SGD), Adagrad e **Adam**, sendo o último o que atingiu melhor resultado.

6. Resultados

Como discutido anteriormente, os modelos de regressão serão avaliados por cinco (5) diferentes métricas: RMSE, taxa de ganho, P/L, payoff e expectativa matemática. Pode-se observar os resultados obtidos na tabela abaixo:

Tabela 1 - Resultado dos modelos de regressão utilizando o algoritmo RandomForest

MÉTRICAS	SEM GRIDSEARCH	COM GRIDSEARCH
RMSE	145,28	117,47
TAXA DE GANHO	52,84%	59,72%
GANHO FINANCEIRO	8,5 pontos/trade	17,98 pontos/trade
PAYOFF	1,34	1,49
EXPECTATIVA MATEMÁTICA	1,50	2,21

Analisando o resultado obtido do modelo de regressão, se levantou um questionamento: Sabe-se que o sistema só possui dois outcomes, acertou ou errou, e que as correlações são

baixas. Levando isso em consideração, e o fato de que a taxa de acerto foi muito próxima de 50% nos dois casos, será que o modelo não estava predizendo aleatoriamente os resultados?

Nota-se que o resultado financeiro foi positivo, mas com a suspeita da aleatoriedade do modelo seria necessário ir mais afundo no problema e discutir resultados de modelos mais complexos, como as RNNs.

Tabela 2 – Comparativo dos melhores modelos de regressão e de RNN

MÉTRICAS	REGRESSÃO LINEAR	RNN
RMSE	117,47	74,26*
TAXA DE GANHO	59,72%	84,3%
GANHO FINANCEIRO	17,98 pontos/trade	66,9 pontos/trade
PAYOFF	1,49	3,23
EXPECTATIVA MATEMÁTICA	2,21	17,34

* calculado a partir do MSE

Analisando todas as métricas, observa que há uma clara (e grande) diferença entre os dois modelos, sendo o modelo RNN cujo resultados foram muito superiores.

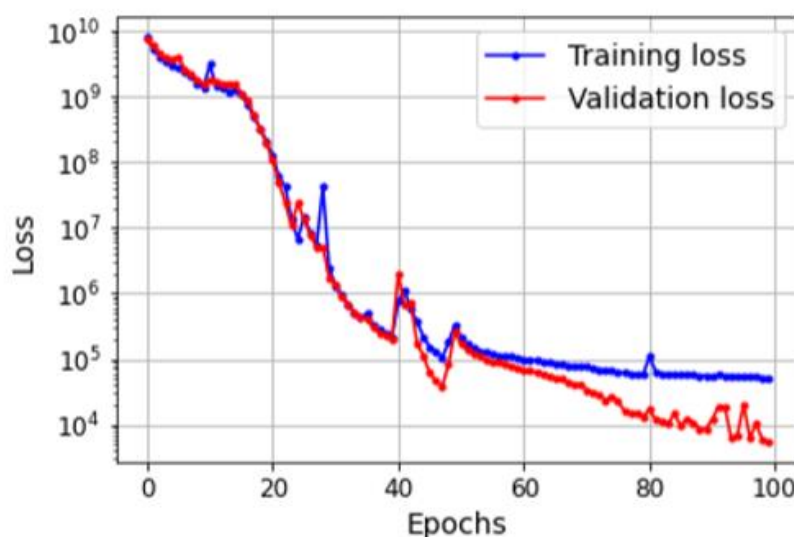


Figura 8 - Variação dos erros de treinamento e validação

A figura 8 acima representa o train_loss e val_loss utilizando o método MSE durante as epochs no treinamento do modelo RNN. Após a epoch 80, observa-se que o train_loss e o val_loss tendem a permanecer constante. Um comportamento desconhecido pode ser observado em diversos momentos, por exemplo epoch 40, do treinamento em que o train_loss e o val_loss têm comportamento de spark (aumento abrupto). Esse fenômeno ocorreu em todos os treinamentos realizados durante o projeto.

O ganho financeiro em pontos/trade foi obtido através da pontuação final obtida dividida pelo número de dados do teste (120). A dinâmica do ganho financeiro pode ser resumida em um crescente constante sem grandes perdas em pontos do ativo. (figura 9).

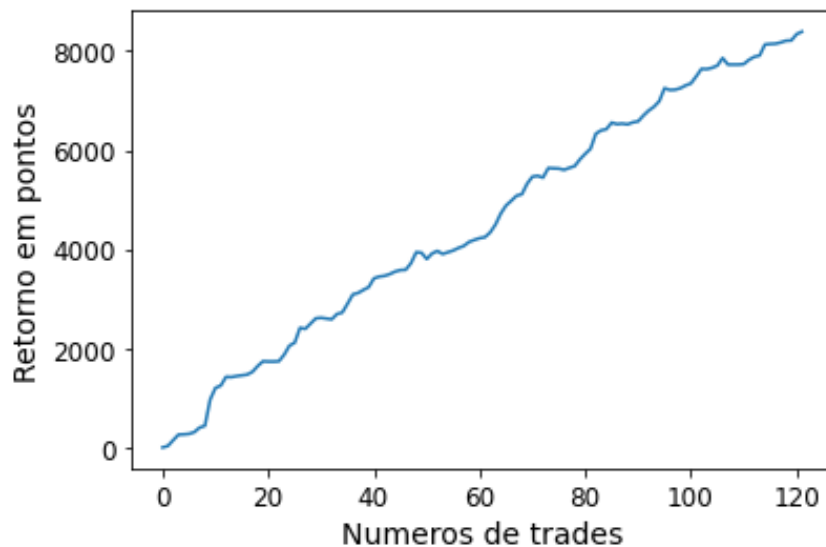


Figura 9 - Curva de capital em pontos ao longo de 120 trades de teste

A próxima análise pode ser feita através da comparação entre os fechamentos do ativo real e os fechamentos do ativo predito pelo modelo (figura 10). O modelo se comportou muito bem com os dados de teste, entretanto, quando há movimentos de grandes amplitudes, o modelo tende a ter um maior erro e com defasagem dos pontos.

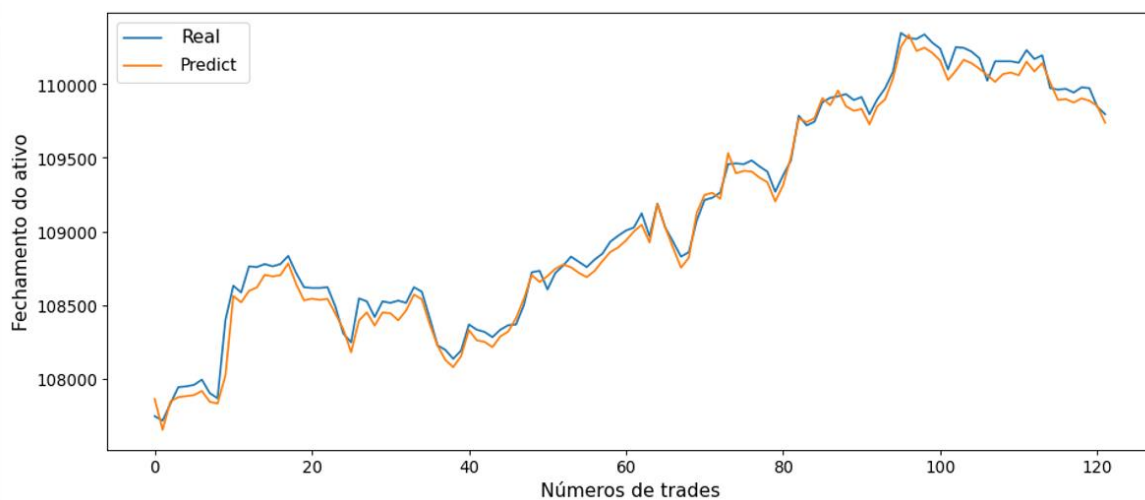


Figura 10 - Comparativo do fechamento predito e do real

Por fim, observar a dinâmica da predição da variação do fechamento do ativo também é importante para visualizar a eficácia do modelo. Define-se variação do fechamento como sendo a diferença entre o fechamento de um instante de tempo comparado com o instante anterior, ressaltando que o instante de tempo equivale a 5 (cinco) minutos. Através da análise anterior do fechamento do ativo, é possível especular que a predição da variação fechamento do ativo também será de boa qualidade, o que se pode confirmar pela figura 11.

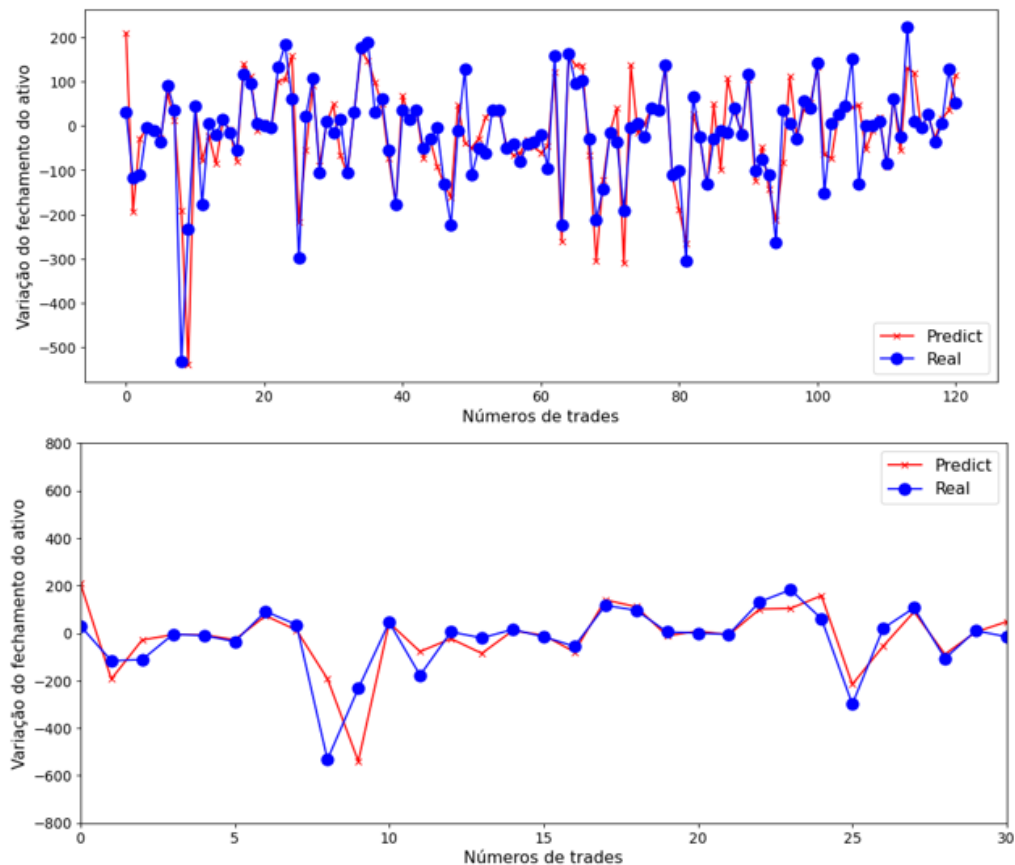


Figura 11 - Comparativo das variações entre candles do modelo predito e do real

O melhor modelo (RNN) projetado nesse trabalho demonstrou resultados além das expectativas iniciais, instigando uma indagação quanto ao resultado do modelo: Com as features atuais seria possível operacionalizar esse modelo?

Após diversas discussões, chegou-se a conclusão de que não seria possível atuar no mercado com o modelo desenvolvido acima. A não operacionalidade do modelo se deve ao fato de que do jeito que o modelo foi concebido, para se prever o fechamento em $t(1)$ são utilizadas features como a máxima e a mínima no instante $t(1)$. Entretanto, o valor dessas features não pode ser confirmado até que o instante $t(1)$ acabe e exista o fechamento $t(1)$.

Com o objetivo de concertar esse equívoco, foi feito um “shift” no dataset inteiro, exceto na abertura do ativo. Ou seja, para cada instante $t(i+1)$, o modelo utilizará a abertura do ativo no instante $t(i+1)$ e todas as demais features do instante $t(i)$. O resultado dessa implementação pode ser observado nas figuras abaixo:

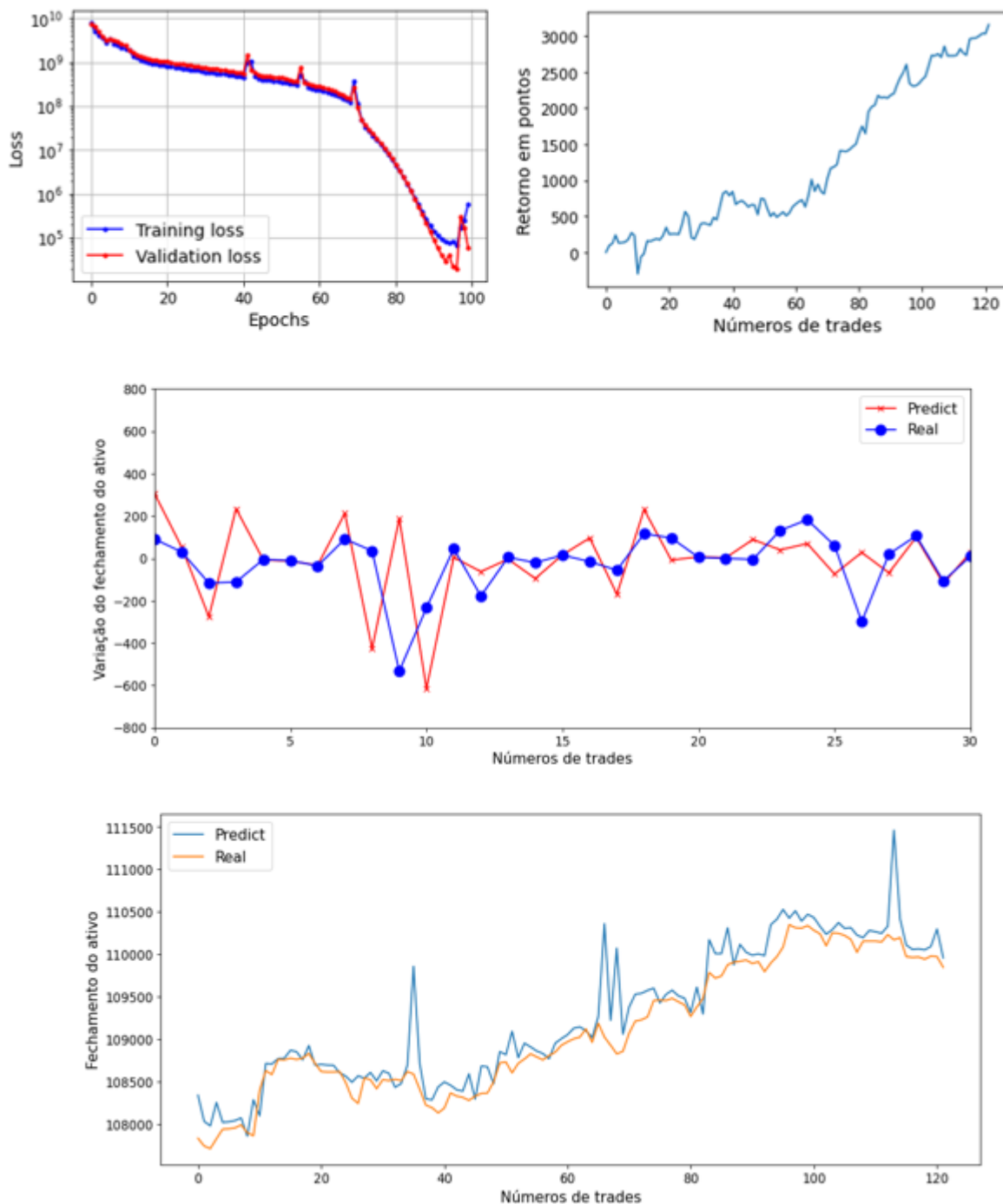


Figura 12 - Curva de capital, comparativo de fechamento e variações para o segundo modelo de RNN testado

Os sparks notados no MSE train_loss do primeiro modelo apresentado também estão presentes nessa nova implementação e o val_loss foi cerca de 10 vezes maior que anteriormente. Embora a dinâmica da predição do modelo siga satisfatoriamente a dinâmica real do ativo, há muitas variações brutas na predição do fechamento do ativo, como pode ser notado no próximo ao trade de número 35,70 e 120.

Resumidamente, os resultados do novo modelo foram todos inferiores ao modelo anteriormente apresentado. Entretanto, esse modelo pode ser utilizado no mercado financeiro. A tabela 3 abaixo representa a comparação dos resultados entre os modelos de RNN.

Tabela 3 - Resultados dos dois modelos implementados de RNN.

MÉTRICAS	PRIMEIRO MODELO	SEGUNDO MODELO
RMSE	74,26*	241,1*
TAXA DE GANHO	84,3%	65,29%
GANHO FINANCEIRO	66,9 pontos/trade	26,3 pontos/trade
PAYOFF	3,23	1,14
EXPECTATIVA MATEMÁTICA	17,34	2,14

* calculado a partir do MSE

7. Conclusão

O último modelo desenvolvido utilizando redes neurais, apesar de não atingir o objetivo inicial de Taxa de Acerto, mostrou-se ser o melhor em termos de resultado no período analisado e de capacidade de se tornar operacional. Mas fica claro que a utilização de métodos de Machine Learning para a predição de um ativo não é trivial. Diversos erros podem ocorrer ao longo do processo e resultar em conclusões precipitadas, como no caso dos resultados aleatórios do modelo de regressão.

Não somente, encontrar um dataset cujos dados são úteis para o modelo também é uma das dificuldades, pois features que não representam bem o objetivo final, no caso o fechamento do ativo, apenas dificultam o processo de treinamento sem retornar resultados satisfatórios.

Com a finalidade de tornar o modelo melhor, algumas modificações são sugeridas nesse relatório como trabalhos futuros:

- **Estender base de dados:** A base utiliza nesse projeto possui aproximadamente 12 mil dados e está contida em um range de preços máximo e mínimo. Aumentar essa base possibilita um melhor treinamento do modelo e uma melhor performance em mais níveis de preço.
- **Incluir outros indicadores de eventos discretos:** O único indicador de eventos discretos utilizado nos modelos foi o Book. Incluir outros indicadores pode melhorar o resultado do modelo em instantes de tempo que possuem fortes variações influenciadas por eventos externos pontuais, como política e saúde.
- **Acrescentar outros ativos ao modelo:** Certos ativos possuem correlação entre eles, incluir esses ativos no modelo pode ser uma adição benéfica.
- **Teste dos modelos em outros ativos:** Embora o modelo seja feito para o WIN, pois foi treinado com os dados desse mesmo ativo, treinar e testar esse modelo em outros mercados, como o a vista (Ações), pode resultar em predições interessantes.
- **Previsão com maiores intervalos:** O intervalo utilizado é de 5 (cinco) minutos, como o modelo se comporta com outros instantes, como 1 (um) dia, 1 (uma) semana?
- **Rever features utilizadas:** Assim como foi identificado o problema com as features, descrito na secção 7, outros detalhes que impactam o resultado final podem não ter sido notados.

Por fim, o projeto por compreendeu diversos aspectos de uma implementação: extração de dados, manipulação de dados, implementação de modelo, comparação de modelos, testes com alterações de parâmetros como layers e optimizers, extração de resultado, análise de resultado e diversas complicações que ajudam a desmistificar um pouco a ideia de que modelos de Machine Learning são a fórmula secreta para ganhar dinheiro no mercado financeiro.

8. Referências

[1] G1. **Descontada a inflação, poupança tem melhor rendimento desde 2009.** Disponível em:<<https://g1.globo.com/economia/seu-dinheiro/noticia/descontada-a-inflacao-poupanca-tem-melhor-rendimento-desde-2009.ghtml>>. Acessado em: 25 de junho de 2020.

[2] Redação Nubank. **Qual o rendimento da poupança 2020?** Disponível em:<<https://blog.nubank.com.br/qual-o-rendimento-da-poupanca-em-2020>>. Acessado em: 25 de junho de 2020.

[3] CHAMMAS, L. **Bull Market – 1 Milhão de investidores na bolsa.** Disponível em:<<https://www.ligafeausp.com/single-post/2019/11/06/Bull-Market---1-milhC3A3o-de-investidores-na-bolsa>>. Acessado em: 25 de junho de 2020.

[4] MITCHELL, C. **Understanding basic candlestick charts.** Disponível em:<<https://www.investopedia.com/trading/candlestick-charting-what-is-it/>>. Acessado em: 25 de junho de 2020.

[5] BANSAL, S. **A comprehensive guide to understand and implement text classification in Python.** Disponível em:<<https://www.analyticsvidhya.com/blog/2018/04/a-comprehensive-guide-to-understand-and-implement-text-classification-in-python/#:~:text=Text%20Classification%20is%20an%20example,1.>>. Acessado em: 25 de junho de 2020.

[6] ANGELIN, O.J.. **Nível de acerto não é referência para setup vencedor.** Disponível em:<<https://profitsideblog.wixsite.com/profitside/post/n%C3%ADvel-de-acerto-n%C3%A3o-%C3%A9-refer%C3%Aancia-para-um-setup-vencedor>>. Acessado em: 25 de junho de 2020.

[7] SILVA, V. FALCO, V. **Stock Market Prediction.** Disponível em:<<https://github.com/Fattson/StockMarketPrediction-ML>>