

Uso de modelos de Redes Neurais para Predição de Consumo de Energia em Sistema de Transmissão

Mateus Raimundo da Cruz, Eduardo Henrique Teixeira,
 Danilo Machado Oliveira e Benedito Donizeti Bonatto
 Instituto de Sistemas Elétricos e Energia - Universidade Federal de Itajubá

Resumo-- As redes elétricas inteligentes, do inglês Smart Grids (SG) são uma evolução do sistema de energia elétrica atual, que acrescenta tecnologias de processamento e troca de dados visando um sistema que consiga prover informações e predições em tempo real. No quesito de monitoramento de consumo por exemplo, a literatura de trabalhos que realizam a predição do consumo de energia é grande, variando desde técnicas mais tradicionais, onde os modelos de predição utilizam cálculos mais simples, até técnicas baseadas em redes neurais, onde a complexidade é maior, mas em contrapartida os modelos muitas das vezes conseguem alcançar resultados mais satisfatórios em termos de métricas relacionadas a precisão da predição. Este artigo propõe o uso de duas técnicas baseadas em Redes Neurais Artificiais para predição de consumo de energia, são elas Long Short Term Memory (LSTM) e Recurrent Neural Network (RNN). Com isso, através de um dataset com dados históricos de 10 anos do consumo de energia em um sistema de transmissão elétrico, foram realizados o treinamento de modelos que conseguem prever dados de consumo futuros e desconhecidos, com base no histórico utilizado para o treinamento. O resultado foi comparado com os dados futuros que a rede desconhecia e a precisão encontrada foi satisfatória.

Palavras-Chave-- LSTM, RNN, Predição de Energia.

I. INTRODUÇÃO

Na maioria das economias desenvolvidas ou em desenvolvimento, as redes elétricas foram criadas há mais de um século e evoluíram de acordo com os avanços da tecnologia. Prever com precisão o consumo de energia de um sistema é essencial para otimizar a geração e distribuição da mesma. Depois de entender o consumo de energia, é necessário formular estratégias apropriadas de economia de energia e otimização dos recursos. Por exemplo, através da previsão de consumo de energia, pode-se obter o comportamento do consumo de eletricidade dos usuários de uma determinada região, dentro de um certo período de tempo, de modo a definir quando é necessário um aumento nos sistemas de geração, ou mesmo da troca de equipamentos em função do aumento da carga [1].

Além disso, com o desenvolvimento de novas tecnologias e recursos, como Inteligência Artificial e grandes Bancos de Dados, o desenvolvimento e a implementação de redes inteligentes e cidades inteligentes tornaram-se a tendência. No entanto, o sistema de consumo de energia cada vez mais complexo trouxe enormes desafios para o equilíbrio entre oferta e demanda. Com isso o uso de preditores de consumo é cada

vez mais necessário para que se tenha um acompanhamento mais próximo desse cenário. Baseando-se nisso, este artigo propõe um método de previsão de consumo de energia baseado em dois diferentes modelos de previsão que usam modelos de redes neurais com recursos de convolução, aplicados a um dataset de consumo histórico de energia.

II. REFERENCIAL TEÓRICO

A. Redes Neurais Artificiais

Uma Redes Neurais Artificiais (RNA) funciona de forma análoga às estruturas neurais biológicas, mas são modelos matemáticos com capacidade computacional adquirida através de centenas ou milhares de unidades de processamento. Os modelos clássicos dessas redes são biologicamente inspirados, sendo as unidades de processamento isoladas, os neurônios, que quando conectados são capazes de uma estrutura de processamento de dados. Eles são organizados em camadas conectadas, que permitem que uma entrada passe pela rede para a camada de saída, fornecendo a resposta a essa entrada.

A Redes Neurais Artificial tem sido aplicada na resolução de várias tarefas no âmbito da visão computacional. Os princípios básicos de funcionamento destas redes são conhecidos há muito tempo, mas, nos últimos anos, o interesse em utilizar estas redes de uma forma prática tem vindo a aumentar consideravelmente [2], [3]. Uma das principais razões para essa crescente atenção dessas redes deve-se à grande precisão que apresentarem nos últimos anos [4], [5]. Outro motivo é que o maior poder de processamento dos dispositivos hoje em dia facilita a viabilidade de aplicação dessa rede. Além disso, a RNA possui uma variedade de aplicações, como classificação de imagens, reconhecimento de padrões, processamento de dados temporalmente e/ou espacialmente correlacionados, entre outras [6].

Como mencionado anteriormente, as RNAs replicam a forma biológica do neurônio humano e, portanto, é comum agrupar esses neurônios artificiais para realizar operações mais complexas [3], [7]. A Figura 1 representa uma RNA, onde podemos observar as entradas em azul, os círculos verdes são os neurônios artificiais ocultos e os vermelhos simbolizam os neurônios de saída.

De forma simplificada, uma rede neural é formada por uma sequência de camadas totalmente conectadas, e cada neurônio recebe sua própria configuração de peso durante o estágio de treinamento. Neste procedimento, os parâmetros de cada

neurônio são aprendidos automaticamente através do aprendizado de máquina [3].

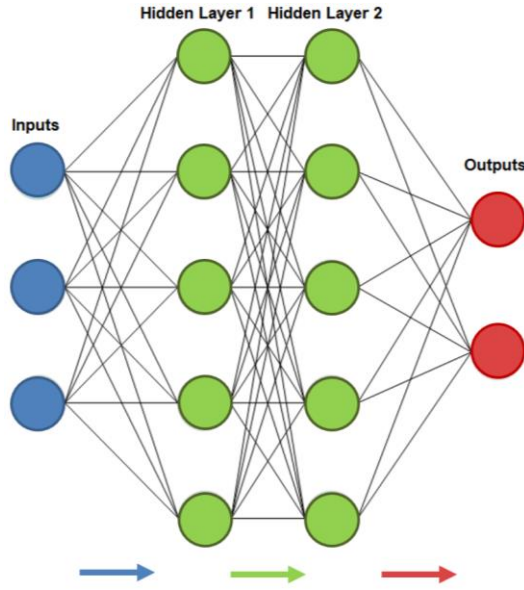


Figura 1 - Exemplo de Rede Neural Artificial.

Considerando o aprendizado supervisionado, o primeiro passo é propagar várias entradas pela rede e comparar as respostas obtidas com a resposta correta. Por exemplo, as imagens rotuladas na entrada são propagadas pela rede para que ocorra um ajuste de peso, capaz de tornar qualquer imagem genérica aplicada à entrada de rede capaz de ser classificada de acordo com seu tipo [7].

Essa técnica de atualização de peso é chamada de retropropagação [7], nela, como o próprio nome diz, ocorre uma retropropagação da derivada de erro, ou seja, os valores dos parâmetros de rede são ajustados por meio de um método de otimização, como o gradiente descendente estocástico. Esse processo é iterado até que o erro seja diminuído para o menor valor possível. Por fim, temos a última camada, na qual os círculos em vermelho podem corresponder às classes e é através dela que a rede expressa o resultado de suas operações.

Outro ponto importante para o treinamento correto é o tamanho do conjunto de dados de entrada rotulado, pois o treinamento é feito quando você sabe a resposta correta para cada possibilidade de entrada e a rede usa essas mesmas informações para ajuste de peso. O treinamento de um RNA com saídas parcialmente rotuladas ou não rotuladas também é possível em algumas aplicações, mas o foco deste trabalho será a aprendizagem supervisionada[6].

Os "neurônios", unidades básicas de uma RNA, são chamados de Perceptron. Neste módulo, cada conexão de entrada tem um peso associado, portanto, quanto mais importante for a entrada, maior tende a ser o peso. Conclusivamente, a saída perceptron é a soma ponderada dos dados de entrada, seguindo uma função de não linearidade para evitar o problema de singularidade que a soma gera, melhorando a capacidade de generalização do modelo [8].

O perceptron contém apenas um neurônio, sendo bastante comum em aplicações de sistemas linearmente separáveis,

como mostrado na Figure 2. O neurônio é capaz de criar um hiperplano de separação entre duas classes, enquanto a saída é responsável por determinar a qual das duas regiões cada dado inserido na entrada pertence, de modo que a função tenha um bom desempenho no retorno das saídas desejadas, conforme especificado na tarefa [8].

A evolução dos recursos computacionais nos últimos anos facilitou a aplicação prática da RNA em problemas reais. O uso favorável dessa rede se deve aos recentes avanços em hardware, como o surgimento da computação de uso geral em GPUs, o que proporcionou uma grande redução no tempo de treinamento de RNAs com arquiteturas complexas.

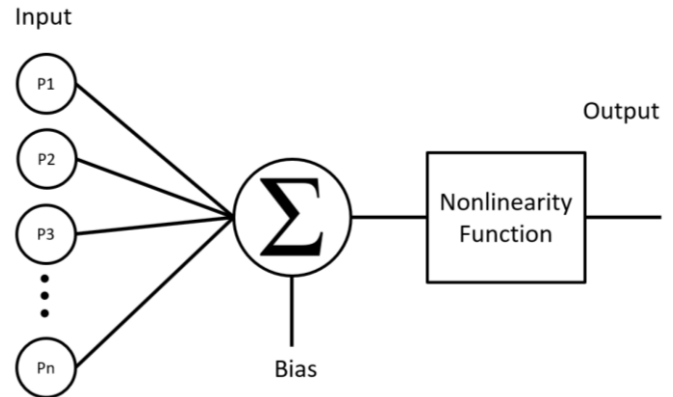
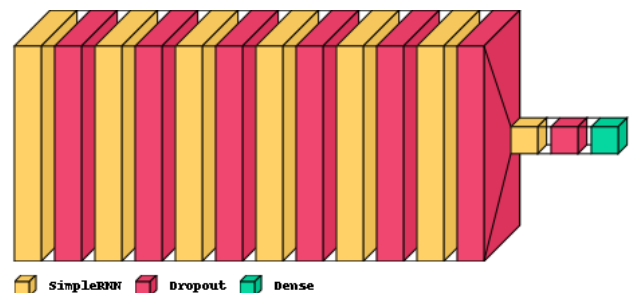


Figure 2 - Perceptron com um neurônio.

III. ARQUITETURAS PROPOSTAS

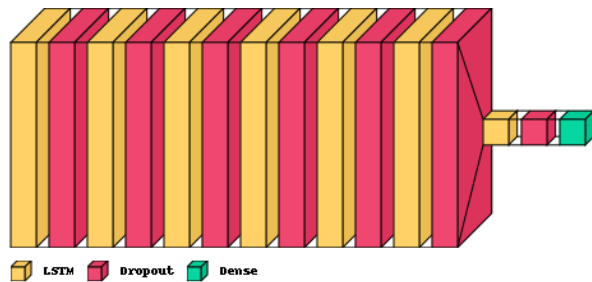
A. Modelo RNN

As redes neurais recorrentes (RNN) são um tipo de rede neural que são especificamente concebidas para aprender e prever sequências de dados. São normalmente utilizadas para aplicações tais como processamento de linguagem natural e tradução automática. Mas quaisquer que sejam os dados da série temporal podem ser utilizados, como a previsão do mercado de ações, previsão do tempo, previsão de vendas, etc. A rede neural é capaz de aprender a sequência porque o RNN tem um loop de feedback, onde a saída de um neurônio é alimentada na entrada do mesmo neurônio, mas com um atraso temporal. Isto permite ao RNN "lembrar" ou "aprender" dados passados, o que é importante para a previsão de dados futuros [9]. A arquitetura desenvolvida para o propósito deste trabalho pode ser visualizada na Figura abaixo:



B. Modelo LSTM

Uma *Long Short-Term Memory* (LSTM) é uma arquitetura de rede neural para a aprendizagem de sequências de eventos, tais como no processamento de linguagem natural ou tradução automática. É uma *Recurrent Neural Network* (RNN), o que significa que os seus neurônios estão ligados de uma forma que lhes permite "lembrar" eventos passados [10]. As LSTM podem também "esquecer" informações que já não são relevantes, o que as torna mais adequadas à aprendizagem de sequências complexas do que as redes neurais tradicionais. As LSTM's têm demonstrado ser muito eficazes numa variedade de tarefas, incluindo reconhecimento de fala, tradução automática e previsão de texto. A arquitetura da LSTM desenvolvida segue a mesma estrutura de camadas:

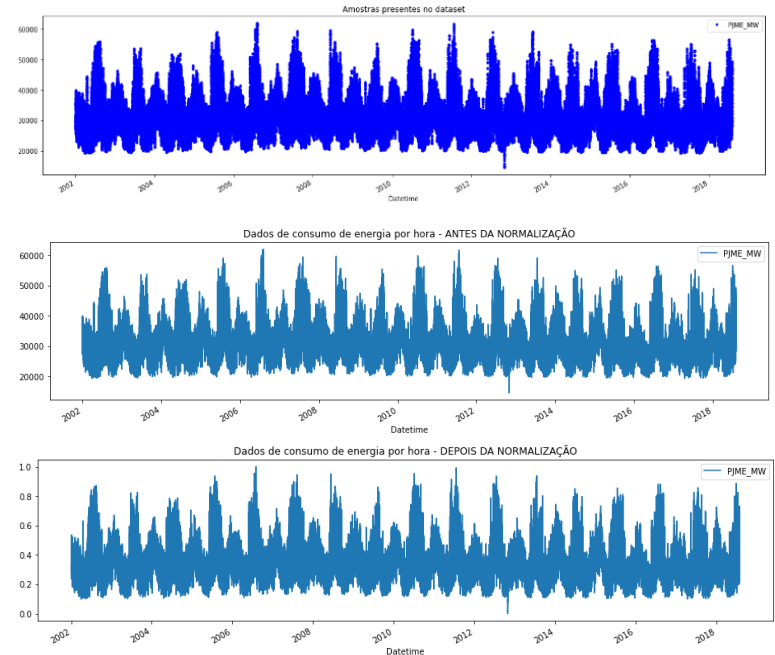


Este tipo de rede neural pode ser facilmente construído utilizando a biblioteca *Tensorflow*. *TensorFlow* é uma biblioteca de software de código aberto para análise de dados e aprendizagem de máquinas e permite aos programadores construir redes neurais complexas para tarefas como o reconhecimento de imagens, processamento de linguagem natural, bem como outras funcionalidades de aprendizagem de máquinas [5]. Para tal, a linguagem *Python* precisa ser instalada e os recursos computacionais disponíveis na máquina local. A Computação na Nuvem também pode ser utilizado para treinar e desenvolver um modelo LSTM para previsão usando o *Tensorflow*. Um recurso gratuito e disponível é o *Google Colab*, um projeto de pesquisa da Google que permite utilizar um ambiente *Jupyter* de forma instantânea e sem a necessidade de nenhuma configuração, executado inteiramente na nuvem. Os modelos apresentados neste trabalho foram todos desenvolvidos usando Computação na Nuvem através da ferramenta Google Colab.

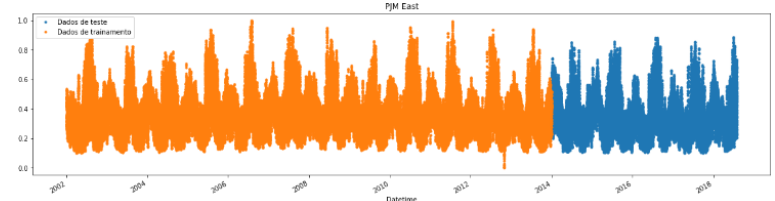
IV. ANÁLISE EXPERIMENTAL

Um bom conjunto de dados é essencial para construção de qualquer modelo no campo da Inteligência Artificial. Isso porque serão os dados com os quais o algoritmo de aprendizagem da máquina aprenderá. Se o conjunto de dados não for bom, o algoritmo de aprendizagem da máquina não será capaz de aprender com ele e não produzirá bons resultados. A base de dados utilizada para treinamento dos modelos pode ser encontrada na comunidade Kaggle, uma comunidade de ciência de dados onde os usuários podem competir para resolver problemas de ciência de dados e compartilhar base de dados. A base de dados contém informações acerca da PJM Interconnection LLC (PJM), uma Organização de Transmissão Regional (OTR) nos Estados Unidos e contém informações

sobre consumo de energia de 12/31/2009 à 06/30/2018, em formato de séries temporais.

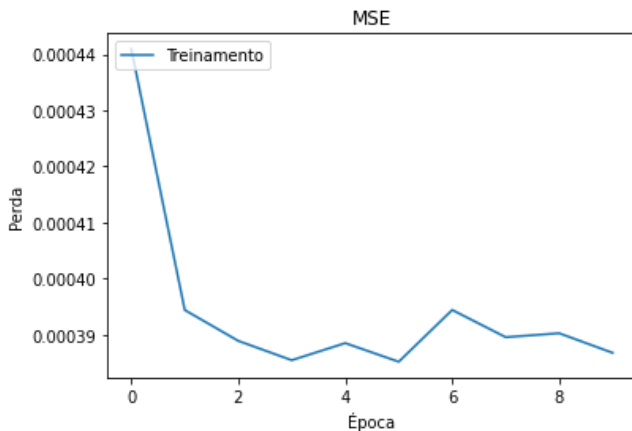


Para treinamento dos modelos, a base de dados foi dividida em conjunto de treinamento e conjunto de teste. Esta divisão é importante, uma vez que o treinamento é o processo de ensino de um algoritmo para reconhecimento de padrões nos dados. O algoritmo é "treinado" num conjunto de dados, que é uma coleção de dados que é utilizada para ensinar o algoritmo a identificar padrões. Uma vez que o algoritmo tenha sido treinado, pode ser testado num conjunto de dados separado para ver o seu desempenho. O conjunto de dados do teste consiste nos dados ainda não utilizados e é utilizado para avaliar a precisão do algoritmo de aprendizagem da máquina.

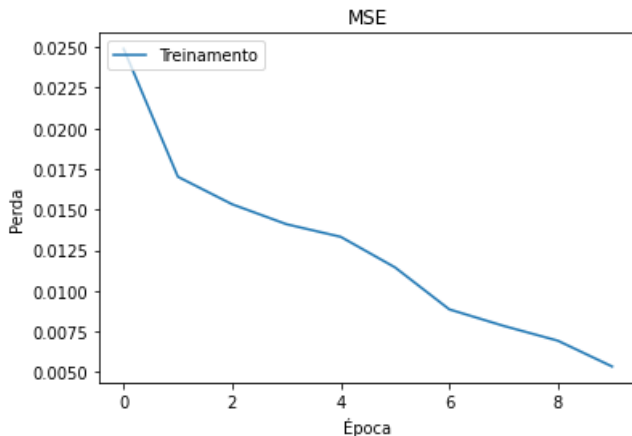


A quantidade de épocas e o tamanho de lote foram as mesmas para as duas técnicas, tornando mais justo a comparação entre as técnicas. O número de épocas numa formação de aprendizagem de máquinas é o número de vezes que os dados da formação são iterados. Em cada iteração, o algoritmo de aprendizagem da máquina ajusta os seus parâmetros para melhor corresponder aos dados de formação. O tamanho do lote, no que lhe concerne, determina quantas observações presentes na base de dados serão utilizadas durante cada iteração. Ambos os modelos desenvolvidos utilizaram 10 épocas para treinamento e um lote igual a 1000, esta combinação de parâmetros foi o suficiente para que os modelos consigam de forma satisfatória prever o consumo de energia dos usuários.

O desempenho dos modelos foi averiguado e comprovado através das métricas Erro Quadrático Médio (EQM) e Função de Perda. Bem como pelas previsões feitas pelos modelos em dados ainda não vistos e posteriormente plotados para melhor entendimento. As Figuras abaixo apresentam a plotagem do Erro Quadrático Médio apresentado pelo modelo durante a etapa de treinamento. O EQM é uma estatística que é utilizada para medir a precisão de uma previsão. É a média das diferenças quadráticas entre os valores previstos e os valores reais. Portanto, o valor EQM representa a quantidade média de erro que é feita nas previsões do modelo. A perda de função é uma métrica utilizada para medir a precisão de um modelo de aprendizagem de máquina. A perda de função é calculada comparando os valores previstos de um modelo de aprendizagem de uma máquina com os valores reais. As Figuras abaixo apresentam as métricas capturadas pelo modelo durante o treinamento.

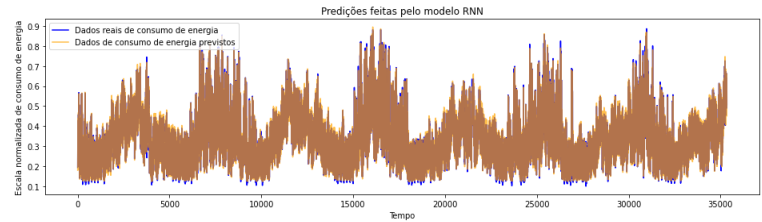


Erro Quadrático Médio apresentado pelo modelo de RNN.

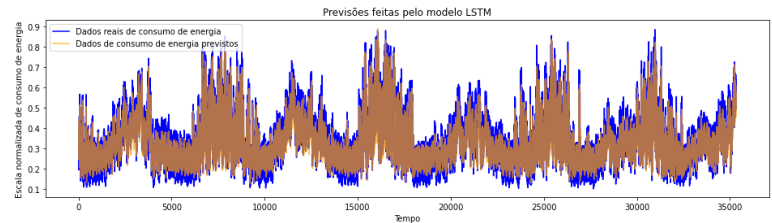


Erro Quadrático Médio apresentado pelo modelo de LSTM.

O conjunto de teste também foi utilizado para avaliação de desempenho dos modelos desenvolvidos. Para tal, foi apresentado ao modelo um conjunto de dados ainda não visto e armazenado os valores preditos pelos modelos para posteriormente comparar as diferenças entre o valor verdadeiro e o valor predito. As Figuras abaixo apresentam as observações presentes na base de dados em comparação as previsões feitas pelos modelos desenvolvidos.



Predições realizadas pelo modelo de RNN.



Predições feitas pelo modelo de RNN.

V. CONCLUSÕES E TRABALHOS FUTUROS

Este artigo foi capaz de demonstrar o uso de duas técnicas baseadas em Redes Neurais Artificiais para predição de consumo de energia, são elas Long Short Term Memory (LSTM) e Recurrent Neural Network (RNN). Com isso, através de um dataset com dados históricos de 10 anos do consumo de energia em um sistema de transmissão elétrico, foram realizados o treinamento de modelos que conseguem prever dados de consumo futuros e desconhecidos, com base no histórico utilizado para o treinamento. O resultado foi comparado com os dados futuros que a rede desconhecia e a precisão encontrada foi satisfatória.

Apesar das LSTM serem uma RNN que utilizam um conjunto diferente de parâmetros e também portões para melhor lembrarem as informações durante períodos de tempo mais longos. O desempenho dos dois modelos foi satisfatório, apesar da pequena diferença nas métricas apresentadas quando comparados. Sendo assim, cabe ao usuário e suas capacidades computacionais determinar qual técnica será melhor aproveitada no cenário de implementação em questão.

Para pesquisas e trabalhos futuros, dada a constante evolução das técnicas de predição de dados, aliada à melhoria do poder de processamento dos dispositivos, propõe-se a análise de diferentes combinações de técnicas, que oferecem desempenhos interessantes para outras aplicações de análise de padrão temporal propostas na literatura.

Referências

- [1] R. Zou, Q. Yang, J. Xing, Q. Zhou, W. Chen, and L. Kong, "Research on public building energy consumption prediction method based on hybrid analysis of dynamic and static data," pp. 7961–7966, 2021.
- [2] M. K. Bhuyan, *Computer Vision and Image Processing Fundamentals and Applications*. Boca Raton: CRC Press/Taylor & Francis Group, 2019.
- [3] S. Haykin, *Redes neurais: princípios e prática*, 2nd ed. Porto Alegre: Bookman, 2007.

- [4] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
- [5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, 2017.
- [6] M. H. Ferreira, “Redes Neurais Artificiais: Princípios Básicos,” *Rev. Eletrônica Científica Inovação e Tecnol.*, vol. 1, no. 13, pp. 47–57, 2016.
- [7] I. Goodfellow, B. Yoshua, and A. Courville, *Deep Learning*, 1st ed. Cambridge, MA, USA: The MIT Press, 2016.
- [8] A. R. Bianchini, “Arquitetura de Redes Neurais para o Reconhecimento Facial Baseado no Neocognitron,” Universidade Federal de São Carlos, 2001.
- [9] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, Dec. 2020.
- [10] E. L. Faria, “Redes Neurais Convolucionais e Máquinas de Aprendizado Extremo Aplicadas ao Mercado Financeiro Brasileiro,” Universidade Federal do Rio de Janeiro, 2018.