

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Previsão de consumo nos estados do Brasil

Julia Leite da Silva

MONOGRAFIA FINAL

MAC 499 — TRABALHO DE
FORMATURA SUPERVISIONADO

Supervisor: Prof. Dr. Marcelo Finger

São Paulo
2022

*O conteúdo deste trabalho é publicado sob a licença CC BY 4.0
(Creative Commons Attribution 4.0 International License)*

Dedico este trabalho a minha família, meu pai e minha mãe que têm iluminado meu dia e me dado força, alegria e paz. Acima de tudo, dedico a Deus, quem me presentou com capacidade, com uma família maravilhosa e que foi adicionando pessoas no meu caminho para me apoiar e me ajudar em cada passo da jornada.

Agradecimentos

Fight with determination, embrace life and live it with passion. Lose your battles with class and dare to win because the world belongs to those who dare to live. Life is worth too much to be insignificant.

— Charles Chaplin

Queria agradecer, antes de tudo, a Deus por tudo que Ele tem me dado. Uma vez minha mãe me disse que Deus abençoa o trabalho das nossas mãos, que várias vezes onde ela pensou que não daria certo ou não daria conta, ela pedia ajuda e, no final, tudo se endireitava. Ela me disse também que Deus colocava pessoas na vida dela que a ajudavam e orientavam a encontrar um caminho. Fico muito grata de enxergar isso acontecendo na minha vida também.

Agradeço aos meus pais, que têm sido minha força, alegria e lar. Agradeços aos meus amigos e amigas, por colorir meus dias e encher meu rosto de sorrisos.

Quero também agradecer ao Marcelo Finger, meu orientador obrigada para o Felipe, que me surpreendeu com a parceria e a nessa jornada. Muito ajuda. Agradecer também ao Gabriel, pelo apoio para escrever esse trabalho.

Resumo

Julia Leite da Silva. **Previsão de consumo nos estados do Brasil**. Monografia (Bacharelado). Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2022.

[illegible]

Palavras-chave: Palavra-chave1. Palavra-chave2. Palavra-chave3.

Abstract

Julia Leite da Silva. . Capstone Project Report (Bachelor). Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2022.

[illegible]

Keywords: Keyword1. Keyword2. Keyword3.

Lista de Abreviaturas

IA	Inteligência Artificial
ML	<i>Machine Learning</i>
IBGE	Instituto Brasileiro de Geografia e Estatística
FGV	Fundação Getúlio Vargas
PIB	Produto Interno Bruto
BACEN	Banco Central do Brasil
IPEA	Instituto de Pesquisa Econômica Aplicada
RMSE	<i>root-mean-square error</i>
MAPE	<i>mean absolute percentual error</i>
MAE	<i>mean absolute error</i>

Lista de Símbolos

ω	Frequência angular
ψ	Função de análise <i>wavelet</i>
Ψ	Transformada de Fourier de ψ

Lista de figuras

1.1	Relação entre inteligência artificial, aprendizado de máquina e aprendizado profundo PATTERSON e GIBSON, s.d.	4
1.2	Exemplo de um modelo simples de regressão linear HYNDMAN e ATHANASOPOULOS, 2021	6
1.3	Ilustração de um neurônio biológico. PATTERSON e GIBSON, s.d. Destaca-se a informação chegando a um dendrito do neurônio por meio de uma sinapse, além de uma sinapse que se inicia no axônio do neurônio e propaga informação a diante.	7
1.5	Ilustração de uma rede neural simples.ACADEMY, 2022	8
1.6	Rectified linear unit (ReLU) PATTERSON e GIBSON, s.d.	9
1.7	Função <i>swish</i> RAMACHANDRAN <i>et al.</i> , 2017	10
1.9	Comparação entre a entrada de uma rede neural tradicional e um recorrente PATTERSON e GIBSON, s.d.	11
1.12	Célula de memória de rede LSTM ACADEMY, 2022	13
1.13	Unidade da rede GRU ACADEMY, 2022	13
2.1	Evolução do consumo de cimento no Brasil SNIC, 2021	16
2.2	Evolução do consumo mensal de cimento em São Paulo	16
2.3	Evolução do consumo de cimento nos estados do Sudeste	17
2.4	Gráfico <i>boxplot</i> SIQUEIRA, 2021	19
2.6	Matriz de correlação	20

Lista de tabelas

2.1	Indicadores utilizados no trabalho	18
3.1	Experimentos realizados na regressão linear	23
3.2	resultado dos experimentos para regressão linear	24

Lista de programas

Sumário

Introdução	1
1 Fundamentação teórica	3
1.1 Aprendizado de máquina	3
1.2 Problema	4
1.3 Regressão linear	5
1.4 Redes neurais	6
1.4.1 Neurônios artificiais	7
1.4.2 Função de ativação	9
1.4.3 Redes neurais multi-layer perceptrons	10
1.4.4 Redes Neurais Recorrentes	10
1.4.5 Normalização	14
2 Metodologia	15
2.1 Dados	15
2.2 Avaliação de performance	21
2.2.1 Mean absolute error (MAE)	21
2.2.2 Root mean squared error (RMSE)	21
2.2.3 Mean absolute percentage error (MAPE)	21
2.2.4 Delta percentual	21
3 Experimentos	23
3.1 Regressao linear	23
3.2 Redes neurais <i>Feed Forward</i>	24
3.3 Redes recorrentes	24
3.4 regressao linear	24
4 Resultados	25

5 Conclusão	27
Referências	29

Introdução

O que prédios, pontes, hidrelétricas e aeroportos têm em comum? Todos são frutos da indústria da construção civil, um importante componente do investimento brasileiro e, consequentemente, uma das grandes engrenagens responsáveis por movimentar a atividade econômica no Brasil. Em 2021, por exemplo, segundo [VASCONCELOS, 2022](#), o Produto Interno Bruto (PIB) desse setor registrou alta de 9,7%, enquanto o PIB do Brasil cresceu 4,6% e o PIB do Agronegócio registrou queda de 0,2%. Dessa forma, o setor da construção figura como importante impulsionador da economia do país, capaz de gerar renda e empregos.

Nesse contexto, conforme relembra [LIMA PERESSIM, 2021](#), o cimento, como é conhecido mundialmente o Cimento de Portland, ao ser misturado com materiais como areia, pedra britada e cal resulta na argamassa e no concreto, alguns dos principais insumos da indústria da construção civil. ([CIMENTO PORTLAND, 2002](#))

Motivação

Contudo, a falta de um modo bem fundamentado para prever o consumo de cimento nos estados do Brasil é uma demanda entre as empresas cimenteiras, uma vez que a construção de uma fábrica é custosa e demorada, a exemplo da fábrica construída pela Votorantim Cimentos no Complexo Industrial e Portuário do Pecém, no Ceará, cuja construção demorou 3 anos e custou cerca de R\$ 200 milhões, conforme noticiado no portal do Instituto Brasileiro de Mineração (IBRAM) ¹. Além disso, o cimento é um produto que não suporta longas estadias em estoque, uma vez que a norma brasileira recomenda o uso em até 90 dias após a fabricação. ([NORMAS TÉCNICAS, 2018](#))

Assim, um modelo que permitisse prever a demanda a nível de estados do Brasil poderia auxiliar gestores a tomar melhores decisões e a estruturar a estratégia de forma mais embasada, de modo a reduzir os riscos do setor. Além disso, poder-se-ia apoiar órgãos governamentais a direcionar ações para mitigar o impacto ambiental da fabricação desse produto, pontuado como um grande emissor de gases de efeito-estufa em escala mundial. ([CARVALHO, 2008](#))

¹ [BRASIL, 2021](#)

Objetivos

Este trabalho, então, propõe-se a aplicar modelos de aprendizado de máquina para determinar qual é mais eficiente para prever a demanda por cimento nos estados do Brasil. Os algoritmos avaliados em ordem crescente de complexidade e robustez são: regressão linear, redes neurais *multi-layer perceptron* (MLP) e redes neurais recorrentes.

Capítulo 1

Fundamentação teórica

1.1 Aprendizado de máquina

Atualmente, a inteligência artificial (IA) permeia diversos momentos do cotidiano. Um exemplo é a empresa norte-americana de *streaming* Netflix, que utiliza um conjunto de técnicas de IA para recomendar conteúdo personalizado aos usuários da plataforma de acordo com os interesses particulares de cada um. Dessa forma, proporciona uma experiência única a cada indivíduo que acessa a plataforma com o objetivo de aumentar a satisfação a longo prazo e garantir a retenção dos membros, uma vez que a plataforma é monetizada com assinaturas mensais.

Em particular para a Netflix, não há um modelo ou algoritmo único utilizado para todas as recomendações de conteúdo. Essa tarefa é dividida em subtarefas realizadas por diferentes modelos de acordo com a atividade a ser realizada e os dados disponíveis. Por exemplo, a escolha de qual vídeo será exibido para o usuário ao logar no perfil da plataforma é executada por um modelo de IA diferente do que o que elenca os vídeos já assistidos que o membro pode continuar a ver. (STECK *et al.*, 2021)

Mas a final, o que é inteligência artificial ? O termo "inteligência artificial", de *artificial intelligence* em inglês, foi elaborado por John McCarthy e utilizado oficialmente pela primeira vez em 1956 no seminário de Dartmouth, um *workshop* sobre a área que reuniu os maiores estudiosos do ramo durante dois meses, segundo (RUSSELL e NORVIG, s.d.). Esse termo apresenta várias definições, de acordo com o pioneiro Arthur Samuel, pode ser definida como o campo de estudo que dá aos computadores a habilidade de aprender sem serem explicitamente programados. (PATTERSON e GIBSON, s.d.)

"The field of study that gives computers the ability to learn without being explicitly programmed." Arthur Samuel

Aprendizado de máquina, por sua vez, vem do inglês *machine learning* e compreende sistemas de IA capazes de adquirir seu próprio conhecimento por meio da extração de padrões dos dados brutos. Configura-se, portanto, como uma sub-área de inteligência artificial de acordo com GOODFELLOW *et al.*, 2016. O aprendizado profundo, ou *deep learning*, é uma categoria específica de *machine learning* que compreende modelos de redes neurais

com várias camadas de neurônios segundo [ZHANG *et al.*, 2021](#). Então, aprendizado profundo configura-se como um tipo particular de algoritmo de aprendizado de máquina que, por sua vez, trata-se de um sub-campo da área de inteligência artificial, essa relação pode ser vista na imagem 1.1.

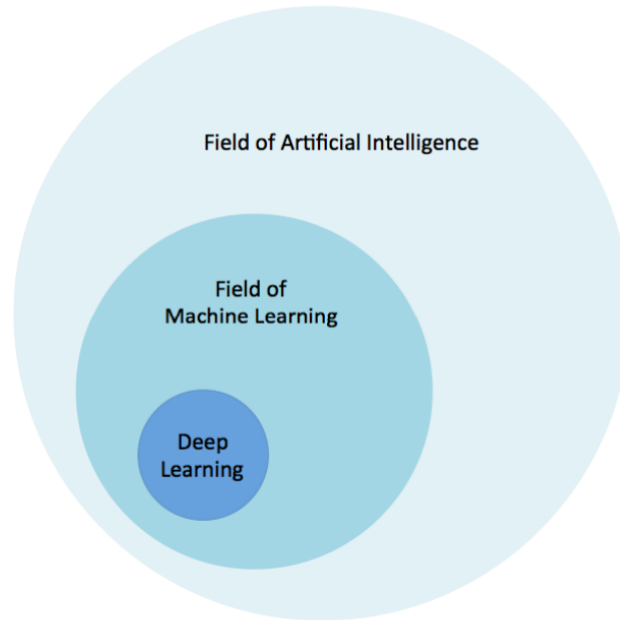


Figura 1.1: *Relação entre inteligência artificial, aprendizado de máquina e aprendizado profundo* [PATTERSON e GIBSON, s.d.](#)

1.2 Problema

As tarefas de *machine learning* são descritas de acordo com o processamento que o modelo deve realizar a partir de um exemplo de entrada (*input*), em geral, descrito com um vetor $x \in \mathbb{R}^n$, $x = \{x_1, x_2, \dots, x_n\}$. Conforme explica [GOODFELLOW *et al.*, 2016](#), nas tarefas de classificação, o modelo deve prever a qual das k categorias disponíveis um *input* pertence, o algoritmo, então, cria uma função $f : \mathbb{R}^n \rightarrow \{1, \dots, k\}$, quando $f(x) = y$, o vetor de entrada x foi classificado na categoria y . Um exemplo de tarefa de classificação, no contexto deste trabalho, seria determinar se o consumo de cimento em um estado em um mês específico representa um aumento, queda ou estabilidade em relação ao mês anterior, essa análise, contudo, foge do escopo do estudo.

Outra categoria de tarefas são as de regressão, cujo objetivo é prever um valor numérico a partir da entrada, a função criada pelo modelo, então, pode ser dada por $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Logo, o problema abordado neste trabalho pertence à categoria de regressão, uma vez que o objetivo é prever o valor do consumo de cimento em um estado e mês específicos a partir dos dados de entrada.

Os problemas de aprendizado de máquina também podem ser divididos entre aprendizado não-supervisionado e supervisionado. No primeiro, o modelo recebe um conjunto

de dados (*dataset*) não rotulado e então aprende propriedades da estrutura do *dataset* e então performar tarefas como a clusterização, que consiste em dividir o conjunto de dados em *clusters* com exemplos similares. No último, por sua vez, os dados de entrada estão associados a rótulos, resultados conhecidos, chamados de *labels* ou *target* em inglês. Neste trabalho, a utiliza-se aprendizado supervisionado, uma vez que o real consumo de cimento no mês é conhecido. [GOODFELLOW et al., 2016](#)

1.3 Regressão linear

A regressão linear é um modelo de aprendizado de máquina que assume um relacionamento linear entre a variável que será prevista (*target*) e os dados de entrada. Dessa forma, o objetivo é obter uma função linear que receba um vetor $x \in \mathbb{R}^n$, $x = \{x_1, x_2, \dots, x_n\}$ como entrada e devolva a previsão de um escalar $y \in \mathbb{R}$. [GOODFELLOW et al., 2016](#) Seja \hat{y} o valor previsto pelo modelo para \hat{y} , então:

$$\hat{y} = w^T x + b \quad (1.1)$$

aonde $w \in \mathbb{R}^n$, $w = \{w_1, w_2, \dots, w_n\}$ é um vetor de parâmetros. Em particular, w corresponde a um conjunto de pesos que determina como cada variável afeta a previsão. Então, se x_i for associado a um peso w_i positivo, aumentar o valor de x_i resulta em um aumento na previsão \hat{y} , se w_i for negativo, por outro lado, um aumento de x_i resulta em diminuição de \hat{y} . Se w_i for igual a 0, por sua vez, a variável x_i não influencia no valor previsto.

A constante b , do inglês *bias*, é um parâmetro para mensurar viés, já que o *output* da função é b na ausência de uma entrada. Dessa forma, a equação 1.1 pode ser escrita da seguinte forma:

$$\hat{y} = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b \quad (1.2)$$

Na ilustração abaixo, um exemplo de um modelo de regressão linear com apenas uma variável:

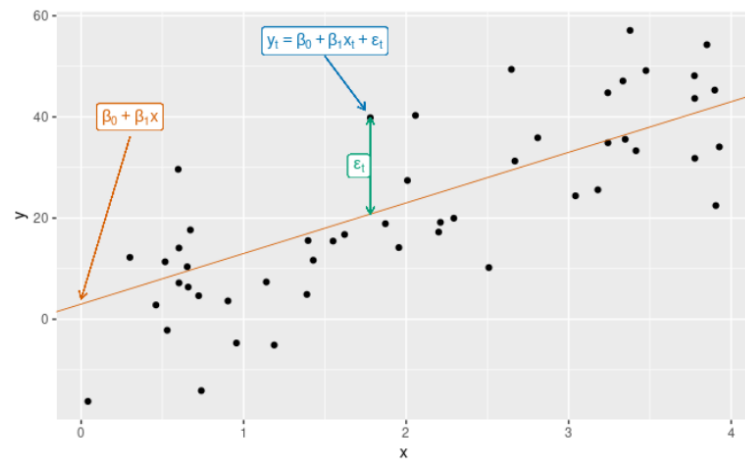


Figura 1.2: Exemplo de um modelo simples de regressão linear *HYNDMAN e ATHANASOPOULOS, 2021*

Nessa imagem, as observações estão representadas nos pontos pretos, enquanto a linha em laranja corresponde à previsão realizada pelo modelo. Observa-se que o modelo não prevê com total exatidão os dados observados, há um erro associado a cada previsão, como o destacado em verde na ilustração.

Dessa forma, cada observação y_i possui um erro ε_i associado e pode ser descrita por $y_i = w^T x_i + b + \varepsilon_i$. O vetor de pesos w , então, é escolhido de modo a minimizar os erros em cada previsão.

Por se tratar de um modelo mais simples, é utilizada neste trabalho como base para comparar o desempenho de outros modelos mais robustos.

1.4 Redes neurais

Redes neurais são modelos computacionais inspirados no funcionamento do cérebro animal, onde neurônios trabalham em paralelo sem uma unidade central de controle. Um neurônio biológico é uma célula nervosa que se comunica com outros neurônios e passa impulsos eletro-químicos de uma célula para outra por meio das sinapses. A comunicação entre neurônios, sinapse, ocorre apenas se o impulso for forte o bastante para ativar a liberação de químicos na fenda sináptica.

Um neurônio é composto de vários dendritos, um axônio e corpo celular, como na imagem 1.3.

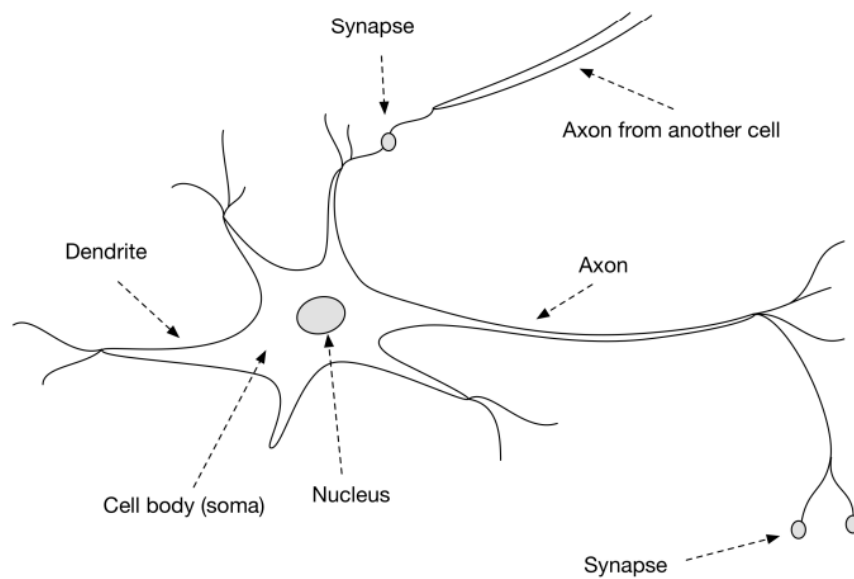


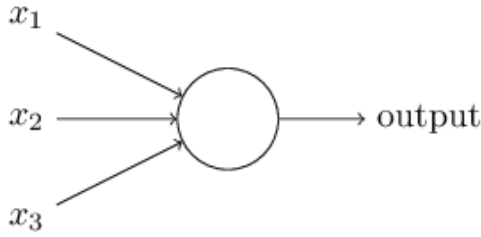
Figura 1.3: Ilustração de um neurônio biológico. *PATTERSON e GIBSON, s.d.* Destaca-se a informação chegando a um dendrito do neurônio por meio de uma sinapse, além de uma sinapse que se inicia no axônio do neurônio e propaga informação a diante.

Os dendritos recebem informações de outros neurônios vizinhos, na forma de impulsos elétricos, e conduzi-las até o corpo celular. Ao chegar no corpo celular, a informação é processada e novos impulsos são gerados. Essa informação é, então, repassada para outro neurônio através do axônio por meio de sinapse. Dessa forma, sinapse é o ponto de contato entre a terminação axônica de um neurônio e o dendrito de outro. *ACADEMY, 2022.*

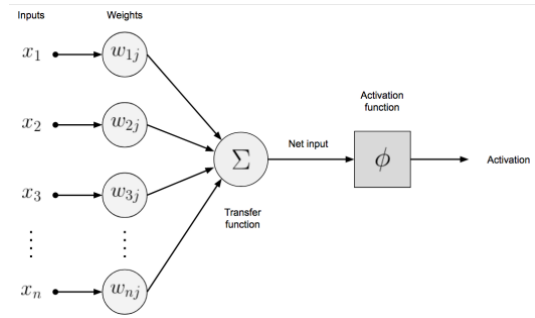
A estrutura e funcionamento dos neurônios biológicos foram base para os cientistas criarem neurônios artificiais, como os *perceptrons*.

1.4.1 Neurônios artificiais

O *perceptron*, foi desenvolvido em 1957 por Frank Rosenblatt, inspirado nos trabalhos de Warren McCulloch e Walter Pitts. Trata-se de um modelo linear de classificação binária que recebe n entradas e produz uma saída binária, como mostrado na ilustração simplificada 1.4a. *ACADEMY, 2022* Esse modelo inicial apresentava limitações e foi evoluído com o passar do tempo, as redes neurais atualmente, em geral, utilizam outro modelo de neurônio como ilustrado em 1.4b.



(a) Visão simplificada de um neurônio *ACADEMY, 2022*



(b) Ilustração da arquitetura de um neurônio artificial *PATTERSON e GIBSON, s.d.*

Como é possível observar na figura 1.4b, cada uma das entradas x_i está associada a um peso w_i . Os pesos w_1, w_2, \dots, w_n expressam a importância das respectivas entradas para o valor de saída, de modo semelhante à regressão linear na seção 1.3. O produto escalar entre os pesos e as respectivas entradas, chamada de *net input* na imagem 1.4b, passa por uma função de ativação ϕ que determina a saída do neurônio. Além disso, um valor de *bias* ou polarização é adicionado ao produto escalar para aumentar a liberdade da função. O *bias* possibilita que um neurônio que possua todas as entradas nulas apresente saída não nula, dessa forma, aumenta a capacidade de aproximação da rede. *ACADEMY, 2022*

Seja x o vetor das n entradas do neurônio, então $x \in \mathbb{R}^n, x = \{x_1, x_2, \dots, x_n\}$, seja também w o vetor com os pesos associados a cada entrada, $w \in \mathbb{R}^n, w = \{w_1, w_2, \dots, w_n\}$. Além disso, seja b o valor de *bias* e Φ a função de ativação. Dessa forma, o *output* de um neurônio é dado por:

$$h_{w,b} = \Phi(w \cdot x + b) \quad (1.3)$$

Essa saída é utilizada como uma das entradas dos neurônios na camada seguinte, de modo a formar a estrutura das redes neurais semelhante ao funcionamento do cérebro. Os neurônios, então, formam a unidade que compõe as redes neurais artificiais, como ilustrado na figura abaixo:

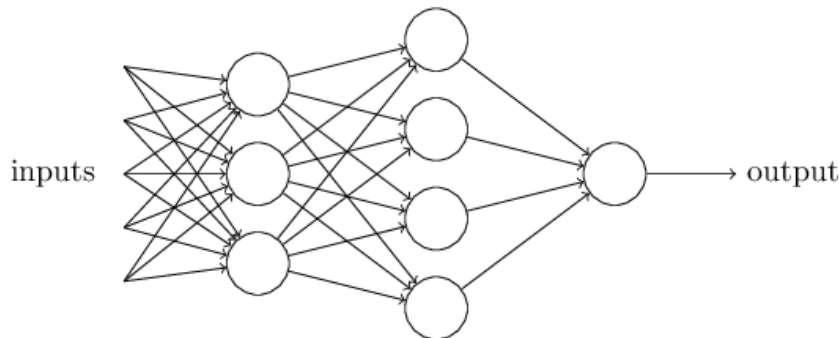


Figura 1.5: Ilustração de uma rede neural simples. *ACADEMY, 2022*

1.4.2 Função de ativação

A função de ativação propaga a saída de um neurônio na rede neural para a camada seguinte. Enquanto os pesos e o *bias* realizam uma transformação linear nos dados de entrada, a função de ativação aplica uma transformação não linear e dessa forma torna possível ue a rede neural resolva problemas não lineares e complexos, como reconhecer padrões de escrita. [ACADEMY, 2022](#)

A função de ativação é um atributo de cada uma das camadas da rede e é escolhida de acordo com a tarefa que será executada, por exemplo, a função sigmóide é recomendada para problemas de classificação. Neste trabalho, foram utilizada: *rectified linear unit* (ReLU) e *swish*.

Rectified linear unit (ReLU)

A função ReLU, do inglês *rectified linear unit*, é o estado da arte atualmente, uma vez que apresenta bom desempenho em diferentes tarefas. [PATTERSON e GIBSON, s.d.](#) A função ReLu é dada por:

$$f(x) = \max(0, x) \quad (1.4)$$

Ao utilizar essa função, a derivada, utilizada para atualizar os pesos e *bias* no treinamento da rede, é zero, quando a entrada é nula ou uma constante, como na imagem abaixo. Dessa forma, a ReLU não sofre do problema da dissipação do gradiente como a sigmóide.

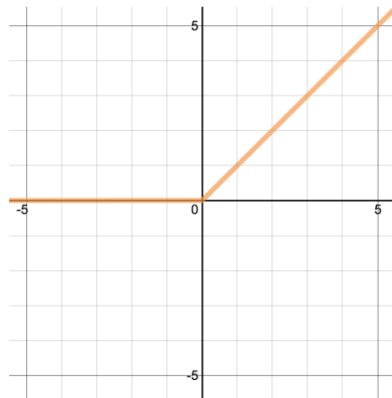


Figura 1.6: *Rectified linear unit (ReLU)* [PATTERSON e GIBSON, s.d.](#)

Swish

A função *swish* foi proposta por pesquisadores da Google com a prerrogativa de apresentar melhor desempenho que a ReLU em redes neurais profundas. [RAMACHANDRAN et al., 2017](#)

A *swish* é uma função não monotônica e suave dada por:

$$f(x) = x \cdot \text{sigmoid}(x) = \frac{x}{1 + e^{-x}} \quad (1.5)$$

O gráfico da função é similar ao da ReLU:

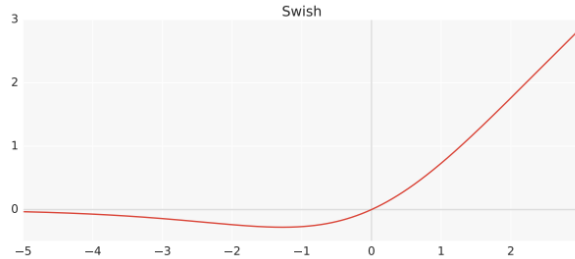


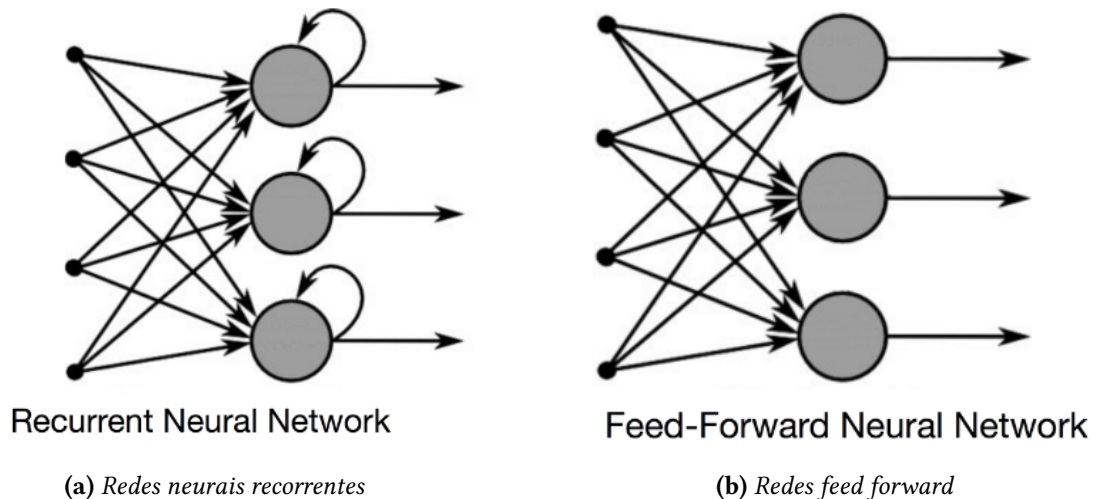
Figura 1.7: Função swish *RAMACHANDRAN et al., 2017*

1.4.3 Redes neurais multi-layer perceptrons

Redes neurais são modelos de *machine learning* inspiradas no cérebro humano aonde o aprendizado ocorre ao se agregarem neurônios matemáticos que estabelecem conexões de acordo com o treinamento fornecido. Neste trabalho, aplicaram-se redes *multilayer perceptrons* (MLPs), ou seja, que apresentam múltiplas camadas de neurônios e *feedforward*, onde a saída de uma camada de neurônios é utilizada como entrada para a camada seguinte, sem utilizar retropropagação.

1.4.4 Redes Neurais Recorrentes

Se por um lado as redes neurais tradicionais são projetadas para realizar previsões sem modelar dimensão temporal, as redes neurais recorrentes se diferem por capturar informação de contexto, ou seja, histórico, para realizar previsões. Essa dimensão temporal, então, se dá por meio de ciclos nas conexões, ou *feedback loops*, como mostrado na figura abaixo.



Dessa forma, os neurônios nas redes recorrentes recebem entradas de ativação de seu próprio estado atual e dos nós anteriores na rede, não apenas dos anteriores na rede, como nas redes tradicionais.

As redes recorrentes, então, não recebem como entrada apenas um vetor de tamanho fixo como as redes *feed forward*, seu *input* é formado por vários vetores, um para cada *time-step*. Na figura abaixo, o número total de amostras corresponde ao campo "examples", o vetor com as variáveis de entrada ao campo "inputs" ou "values per time step" (na figura de Recurrent Network Data) e o tamanho das sequências consideradas pelo modelo ao "time steps".

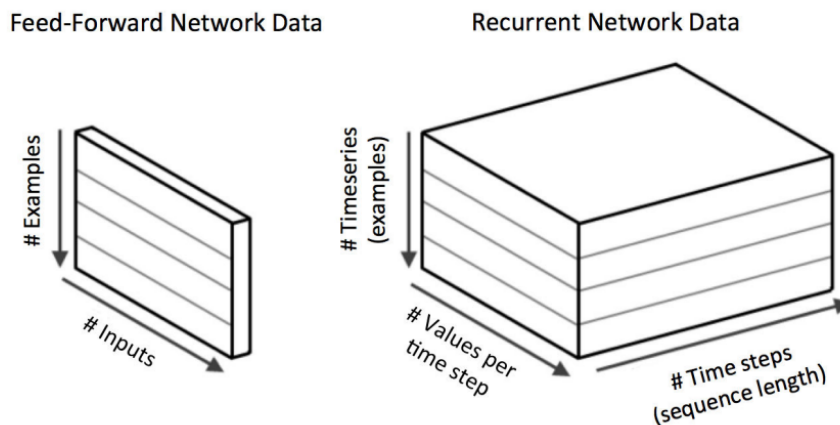


Figura 1.9: Comparação entre a entrada de uma rede neural tradicional e um recorrente *PATTERSON e GIBSON, s.d.*

Redes recorrentes são uma subclasse das redes neurais, contudo, possuem mais de uma categoria, neste trabalho testaram-se as redes LSTM e GRU.

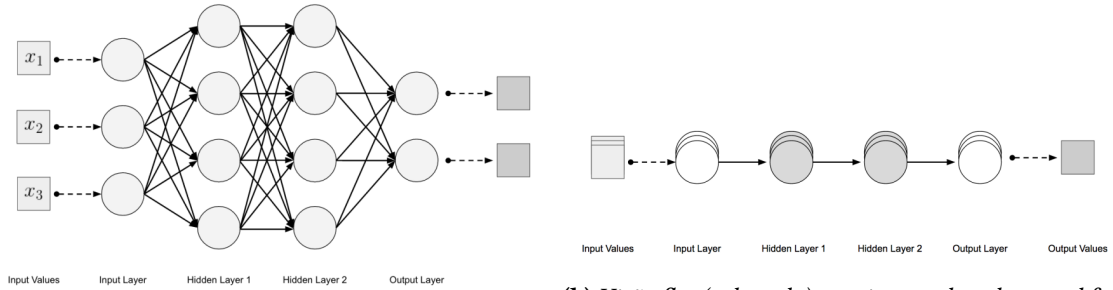
LSTM

As redes Long Short Term Memory (LSTM) foram introduzidas em 1997 por Hochreiter and Schmidhuber¹ e são a variação mais utilizada das redes neurais recorrentes atualmente, por exemplo para classificar séries temporais, reconhecimento de fala e de escrita a mão. Redes LSTM apresentam células de memória conectadas, o conteúdo dessas células é modulado pelo *input gate* e o *forget gate*, por exemplo, se ambos estiverem fechados em um tempo o conteúdo da célula permanecerá o mesmo no tempo em questão e no próximo. Essa estrutura permite que a informação seja retida ao longo do tempo e evita o problema do *vanishing gradient* que ocorre com a maioria das redes recorrentes.

As redes neurais LSTM apresentam uma arquitetura diferente das redes tradicionais, uma vez que há ciclos de *feedback* nas conexões entre as células. Para melhor ilustrar essa arquitetura, *PATTERSON e GIBSON, s.d.* utiliza a visualização *flat* ou achatada das redes neurais, na qual as células de uma mesma camada da rede estão representadas como um único nó. Pode-se observar essa representação em , na imagem ?? está a representação

¹ HOCHREITER e SCHMIDHUBER, 1997

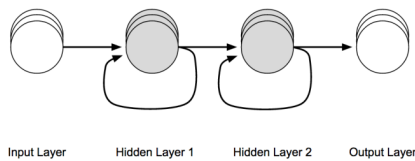
tradicional de uma rede neural *feed forward*, já em em 1.10b está a mesma rede com a representação achatada.



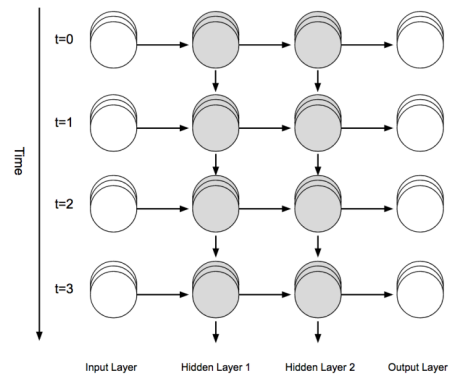
(a) Arquitetura de rede neural feed forward

(b) Visão flat (achatada) arquitetura de rede neural feed forward

As redes LSTM introduzem o conceito de conexão entre a saída de uma das camadas ocultas da rede neural com a entrada da mesma camada. A partir desse ciclo, obtém-se entradas de tempos anteriores como parte da informação que chega à rede neural no tempo atual. Na figura 1.11a, essas conexões recorrentes são representadas como as setas que saem de uma célula e atingem a mesma célula, uma vez que utiliza-se a representação achatada. A imagem 1.11b, por sua vez, representa a rede LSTM desenrolada através do eixo do tempo.



(a) Visão flat (achatada) arquitetura de rede neural LSTM



(b) Visão da rede neural LSTM ao longo do tempo

As unidades que formam cada uma das camadas de uma LSTM são uma variação dos neurônios artificiais clássicos. Essas unidades, representadas em 1.12, permitem que a rede mantenha o estado ao longo do tempo e apresentam dois tipos de conexões: conexões vindas da camada anteriores e vindas *outputs* dessa mesma unidade em tempos anteriores.

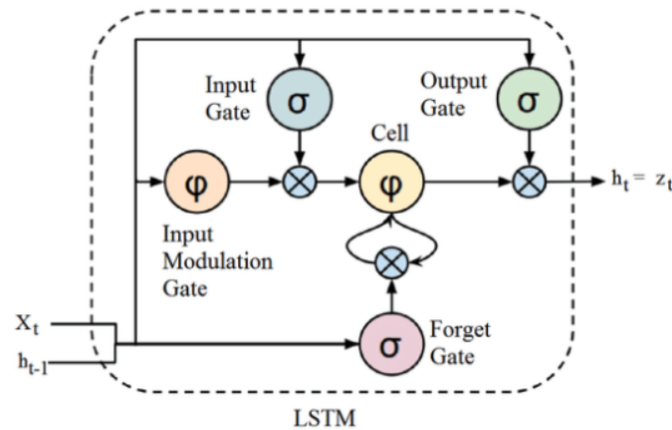


Figura 1.12: Célula de memória de rede LSTM *ACADEMY, 2022*

A unidade LSTM recebe duas entradas, a observação no tempo atual e a saída do último estado oculto. Nas unidades LSTM, a informação é retida nas células e as manipulações de memória são realizadas nos *gates* (portões). O *forget gate* é responsável por remover informações que não são mais úteis para a célula, ou seja, pelo esquecimento, essa operação é realizada por meio de multiplicação de matrizes de pesos, adição de parâmetro de viés e função de ativação que fornece uma saída binária. O *input gate*, por sua vez, adiciona informações úteis ao estado da célula a partir da aplicação de funções sigmóides e tangente hiperbólica, além da multiplicação de vetores. Já o *output gate* extrai informações úteis ao estado da célula atual para apresentar como saída da célula e entrada para a próxima com funções tangente hiperbólica e sigmoide, além da multiplicação de vetores. *ACADEMY, 2022*

GRU

As GRU, sigla para *gated recurrent units* no inglês, são unidades recorrentes similares às LSTM. Enquanto as unidades das redes LSTM apresentam dois estados passados entre as células, estado da célula e estado oculto, que carregam memória, respectivamente, de longo e curto prazo, as unidades GRU apresentam apenas um estado oculto carregado ao longo do tempo e capaz de manter as dependências de curto e longo prazo ao mesmo tempo. A estrutura das unidades GRU pode ser vista em 1.13.

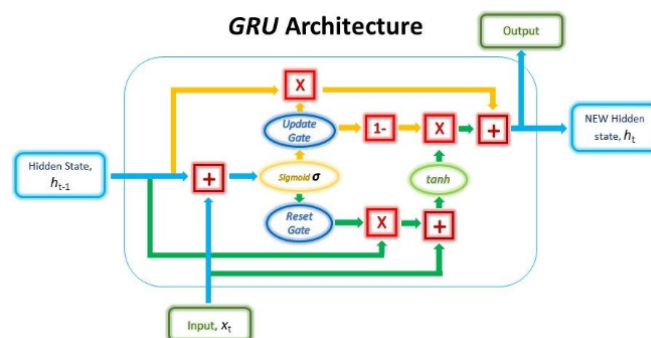


Figura 1.13: Unidade da rede GRU *ACADEMY, 2022*

As unidades GRU apresentam portões, assim como as unidades LSTM, para controlar o fluxo de informações. O *reset gate*, ou portão de redefinição, é responsável por controlar quais informações das etapas anteriores serão mantidas com as da etapa atual por meio da função sigmoide e multiplicação de vetores. O *update gate*, por sua vez, tem como objetivo determinarmos quanto das informações armazenadas no estado oculto anterior devem ser retidas para o futuro.

1.4.5 Normalização

O objetivo da normalização dos dados, é garantir que as variáveis de entrada estejam na mesma escala. Esse processo é necessário não apenas para realizar melhor comparação entre variáveis com diferentes unidades, mas também para o melhor funcionamento dos algoritmos de aprendizado, uma vez que se as *features* estiverem em escalas diferentes os pesos podem ser atualizados mais rápidos que outros.

Standard scaler

O *standard scaler* garante que as variáveis de entrada estejam em uma escala com as propriedades de uma distribuição normal, ou seja, média (μ) igual a zero e variância (σ) igual a um. Dessa forma, a operação realizada para aplicar o processo em uma entrada x é

$$z = \frac{x - \mu}{\sigma} \quad (1.6)$$

Min-Max scaler

O *min-max scaler* transforma os dados de modo que a escala dos dados tenha um intervalo definido, em geral, entre 0 e 1.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1.7)$$

Capítulo 2

Metodologia

Neste trabalho, utilizou-se dados públicos de fontes como o Instituto Brasileiro de Geografia e Estatística (IBGE), a Fundação Getúlio Vargas, FGV, e o Banco Central do Brasil (BACEN) para prever o consumo mensal de cimento nos estados da União. Para melhorar o desempenho dos modelos de aprendizado de máquina, realizaram-se técnicas de preparação de dados, como o preenchimento de valores nulos, e para mensurar o acerto das previsões são aplicadas as métricas estatísticas: *root mean squared error* (RMSE), *mean absolute error* (MAE) e *mean absolute percentage error* (MAPE), além do delta percentual.

2.1 Dados

O consumo mensal de cimento nos estados do Brasil, alvo da previsão dos modelos aplicados neste trabalho, é um indicador fornecido pelo SNIC, Sindicato Nacional da Indústria de Cimento. Os dados obtidos apresentavam granularidade mensal para cada um dos estados da União e estavam disponíveis de 1990 até 2021, a exceção dos anos de 2001 e 2002. Dessa forma, optou-se por utilizar apenas dados posteriores a 2003 no trabalho, uma vez que os modelos recorrentes se valem do contexto para obter a próxima previsão [1.4.4](#), então esse lapso nos dados poderia trazer inconsistências no modelo.

Além disso, o descarte de dados mais antigos favorece a assertividade do modelo, pois houve alteração no total de cimento consumida no Brasil ao longo do tempo, então dados mais antigos poderiam privilegiar tendências mais antigas. Retirar dados mais antigos, então evita, por exemplo, que a estagnação econômica da época de hiperinflação ou o otimismo do Plano real impacte a previsão atual. Essa alteração na quantidade de cimento consumida pela União pode ser observada na imagem abaixo.

Consumo aparente no Brasil (em milhões de toneladas)

Apparent consumption in Brazil (millions of tonnes)

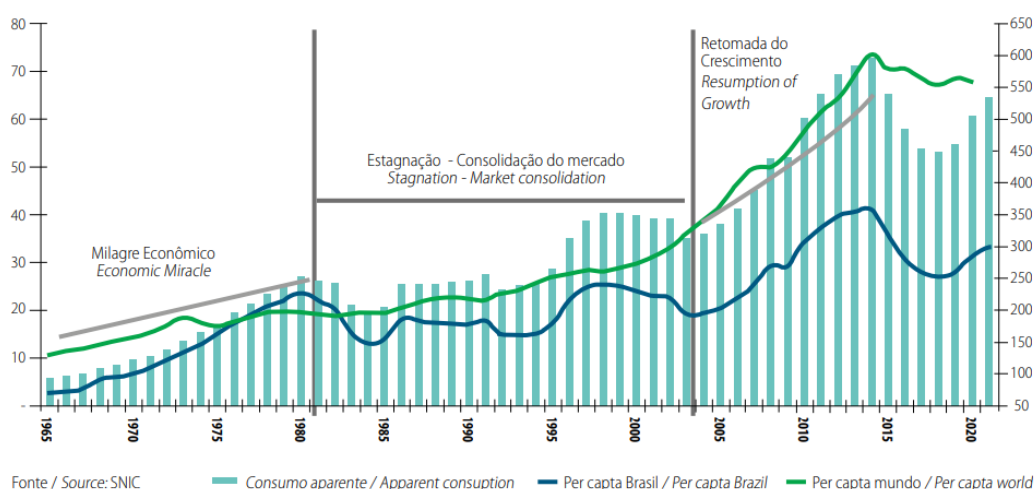


Figura 2.1: Evolução do consumo de cimento no Brasil SNIC, 2021

Observa-se, por exemplo, uma queda no consumo de cimento entre 2015 e 2018, fruto das crises econômicas nesse período. Ressalta-se, também, que o consumo não é uniforme ao longo do ano como pode ser visto na figura 2.2, aonde observa-se a grande variação na demanda de cimento no estado de São Paulo, não apenas ao longo dos anos, mas também ao longo dos meses do ano

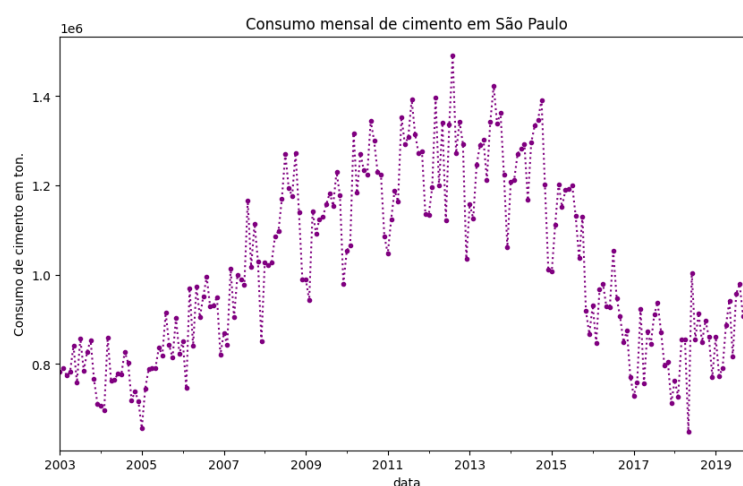


Figura 2.2: Evolução do consumo mensal de cimento em São Paulo

Além disso, destaca-se que existe uma grande diferença no consumo entre os estados do Brasil, por exemplo, o consumo médio em São Paulo de 2003 a 2019 foi de aproximadamente 1014,7 mil toneladas enquanto o de Alagoas foi de 41,5 mil toneladas. Mesmo ao comparar apenas estados de uma mesma região, é evidente que há singularidades de cada unidade da federação, como pode ser visto na imagem 2.3.

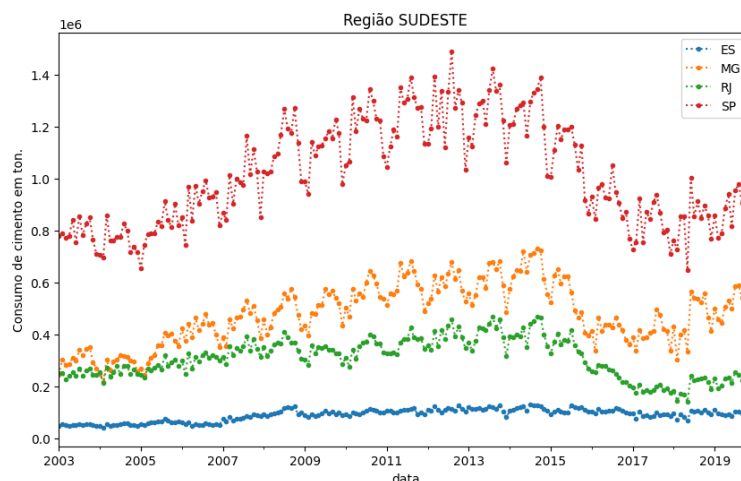


Figura 2.3: *Evolução do consumo de cimento nos estados do Sudeste*

No gráfico 2.3, os pontos representam cada uma das medições mensais do consumo de cimento em cada um dos estados da região.

A região Sudeste foi escolhida neste trabalho para demonstrar as previsões do modelo. Essa escolha se deu pois possui apenas quatro estados, então o gráfico com as previsões é mais simples de ler e interpretar que o de outras regiões com maior número de estados. Além disso, os estados dessa região apresentam certa distância na quantidade de cimento consumida, dessa forma, há pouca sobreposição entre as curvas, isso torna o gráfico mais legível também. Por exemplo, a média de consumo de cimento do período analisado no estado de São Paulo foi de 1014,7 mil toneladas, no Rio de Janeiro foi de 310,2 mil toneladas, já em Minas Geral foi de 480,3 mil tonelada e no Espírito Santo, por sua vez, foi de 91,9 mil toneladas.

O modelo utiliza dados econômicos, sociais e da construção civil para treinar estimar a demanda por cimento. Na tabela 2.1, estão apresentadas a fonte, granularidade e o período em que os dados estavam disponíveis para todos os indicadores utilizados neste trabalho. Na tabela são utilizadas siglas para melhor visualização, como IBGE para Instituto Brasileiro de Geografia e Estatística, FGV para Fundação Getúlio Vargas, BACEN para Banco Central do Brasil, IPEA para Instituto de Pesquisa Econômica Aplicada e SNIC para Sindicato Nacional da Indústria do Cimento todas fontes utilizadas neste trabalho. Além disso, utilizou-se abreviação para Produto Interno Bruto (PIB), Índice Nacional de Preços ao Consumidor Amplo (IPCA), Índice Nacional de Custo da Construção (INCC), Índice Geral de Preços (IGP), Necessidade de Financiamento do Setor Público (NFSP) e Índice de Desenvolvimento Humano (IDH).

Indicador	Fonte	Período disponível	Granularidade
PIB a preços constantes	IBGE ^a	1983 até 2019	anual por estado
PIB a preços de mercado	IBGE ^a	1985 até 2019	anual por estado
PIB per capita	IBGE ^a	1985 até 2019	anual por estado
PIB da construção civil	IBGE ^a	1985 até 2019	anual por estado
Desemprego	IBGE ^a	1991 até 2022	irregular ^b
IPCA	IBGE ^c	1981 até 2021	mensal para o Brasil
INCC	FGV ^d	1980 até 2021	mensal para o Brasil
IGP	FGV ^a	1944 até 2021	mensal para o Brasil
Taxa Selic	IBGE ^e	1986 até 2022	mensal para o Brasil
NFSP	BACEN ^a	1991 até 2022	mensal para o Brasil
Estoque líquido de capital fixo	IPEA ^a	1947 até 2019	anual para o Brasil
População	IBGE ^f	1991 até 2021	anual por estado
IDH	IBGE ^a	1991 até 2017	irregular ^g
Produção mensal de cimento	SNIC ^h	2003 até 2022	mensal por estado
Valor médio do cimento ⁱ	IPEA ^h	1947 até 2019	anual para o Brasil

Tabela 2.1: Indicadores utilizados no trabalho

^a Dado retirado do portal do Ipeadata em <http://www.ipeadata.gov.br/Default.aspx>

^b Havia dados de 1992 até 2014 com granularidade anual e por estado, contudo, a partir de 2012 foram disponibilizados dados mensais a nível de Brasil por conta da Pesquisa Nacional por Amostra de Domicílios (PNAD) Contínua mensal realizada pelo IBGE. Neste trabalho, utilizou-se os dados anuais até 2012 e, após 2012, os dados provenientes da PNAD Contínua

^c Dado retirado do IBGE em <https://sidra.ibge.gov.br/tabela/1737>

^d Dado obtido a partir do portal da FGV em <https://www.debit.com.br/tabelas/tabela-completa-pdf.php?indice=incc>

^e Dado obtido em <https://www.debit.com.br/tabelas/tabela-completa.php?indice=selic>

^f Dado obtido do portal Base dos Dados em <https://basedosdados.org/dataset/br-ibge-populacao>

^g Os indicadores de IDH (Renda, Longevidade e Educação) estão disponíveis em anos de censo do IBGE (1990, 2000, 2010), há dados, também, de 2014 a 2017 por conta da PNAD Contínua

^h Dados retirados do portal <http://www.cbicdados.com.br/menu/materiais-de-construcao/cimento>

ⁱ Evolução do valor médio/mediano do cimento Portland 32 em US\$/Tonelada

Com o objetivo de direcionar a estratégia de preparação de dados foi realizada uma análise exploratória de cada um dos dados de entrada e da variável resposta. A partir dessa análise, optou-se por utilizar os dados de 2003 até 2019 para o estudo. Destaca-se nessa análise, a alta taxa de variação dos dados de entrada, em alguns indicadores, por exemplo para o PIB da construção civil, o desvio padrão é maior que o valor médio do indicador. Dessa forma, está presente nos dados um grande número de *outliers*, em especial ao analisar o Brasil como um todo, por conta da forte diferença entre as regiões do Brasil.

An outlier is an observation strikingly far from some central value. It is an unusual value relative to the bulk of the data... TUKEY, 1977

Outliers são observações discrepantes do restante dos dados, que parecem inconsistentes e podem interferir no processo de previsão. HOAGLIN, 2013 Para ilustrar melhor o alto

volume de *outliers* presentes nos dados, utilizou-se gráficos *boxplot*, como mostrado na figura abaixo.

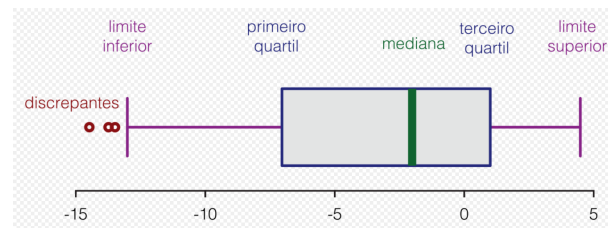
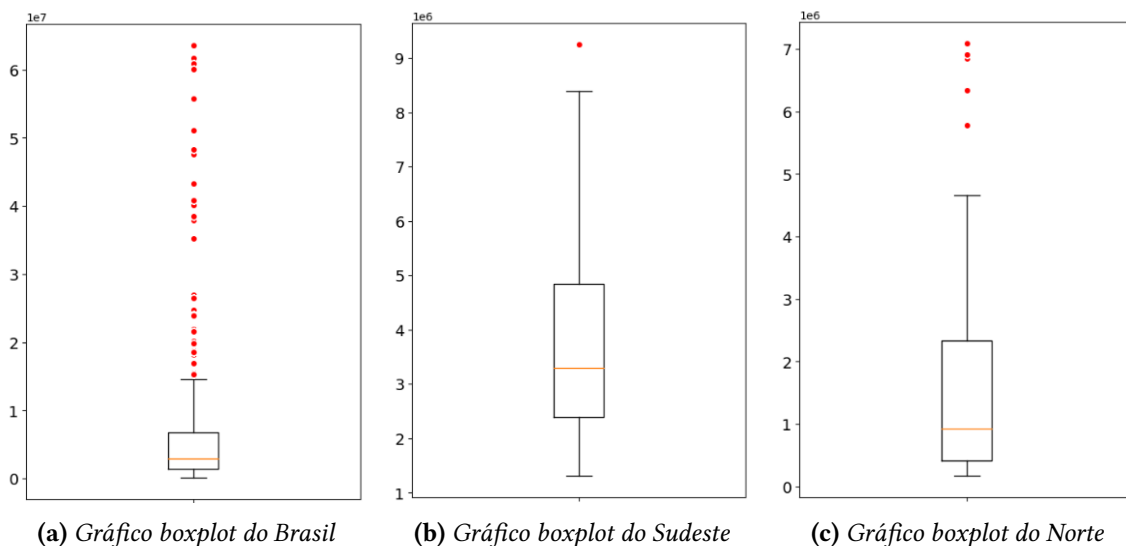


Figura 2.4: Gráfico *boxplot* SIQUEIRA, 2021

O *boxplot* WILLIAMSON *et al.*, 1989 é uma técnica estatística utilizada na análise exploratória os dados para identificar visualmente padrões, como a distribuição dos dados. Na figura, acima linha em verde corresponde à mediana¹ dos dados, os limite inferior e o superior ao retângulo representam o primeiro e terceiro quartil², respectivamente.



Conforme explicado na figura 2.4, a linha em laranja nos gráficos acima representa a mediana dos dados, o retângulo que envolve a mediana é delimitado pelo primeiro e terceiro quartil e retém metade central das amostras. Os valores reoesentados por pontos vermelhos acima ou abaixo dos limites superiores ou inferiores são *outliers*. Observa-se, então, que pode-se obter uma significativa redução no número de amostras com *outliers* ao separar a análise por região. WILLIAMSON *et al.*, 1989

Analizou-se, também, a correlação entre as variáveis de entrada e observou-se alta correlação entre os indicadores de PIB do estado, PIB da construção civil e população.

¹ valor que fica no meio quando os dados estão ordenados ou a média dos dois valores centrais se o número de pontos de dados for par ESSELMAN, 2022

² O primeiro e terceiro quartis também representam a mediana dos valores superiores à mediana dos dados e a mediana dos valores inferiores à mediana. Então metade dos dados está contida dos quadrados nas imagens

Além disso, há alta correlação entre os três indicadores de IDH, além do preço do saco de cimento e o preço do quilograma, como pode-se validar na figura 2.6.

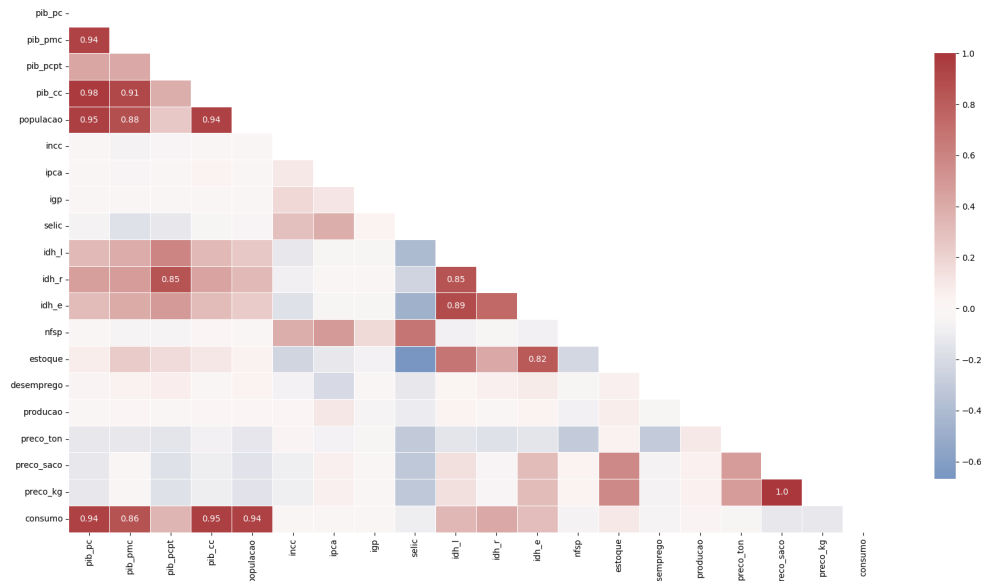


Figura 2.6: Matriz de correlação

Foram adotadas estratégias para garantir dados na granularidade mensal e por estado. Caso os indicadores apresentassem granularidade anual, o valor foi dividido por 12 de modo a obter a média mensal, já caso a granularidade fosse a nível de Brasil, o valor apresentado foi repetido para todos os estados.

A estratégia utilizada para lidar com dados faltantes foi, sempre que possível, repetir o valor anterior que estava disponível nos dados de entrada para preencher a ocorrência. Contudo, alguns indicadores não apresentavam valores mais antigos, então foi usado um valor não presente no intervalo de dados de entrada (-1) para marcar como nulo. Os indicadores com dados faltantes são, em ordem decrescente de acordo com o percentual de dados faltantes: produção de cimento, valor médio do quilo, saco e da tonelada e cimento e desemprego.

Além disso, tomou-se um cuidado para evitar que a previsão fosse realizada com os dados do mês anterior ou do ano anterior no caso dos indicadores anuais. Deslocou-se, portanto, os dados de entrada para frente em um ocorrência de modo a associar os dados de um mês com o consumo no mês seguinte.³

Por fim, o estado correspondente à medição foi usado como dado de entrada. Como os modelos de inteligência artificial aceitam apenas caracteres numéricos, utilizou-se o método de codificação *one hot* para criar 27 colunas, uma para cada estado, nas quais o valor é 1 quando a linha possui dados daquele estado e é 0 caso contrário.

³ os dados correspondentes a, por exemplo, fevereiro de 2004 estão relacionados ao consumo de cimento em março de 2004, com o objetivo de propor um cenário mais pertinente, uma vez que o objetivo do projeto é prever a demanda por cimento no mês seguinte em um estado a partir dos dados do mês atual e, eventualmente, dos anteriores.

2.2 Avaliação de performance

Para comparar a eficiência dos modelos mede-se os erros de cada previsão, ou seja, a distância entre o valor previsto pelo algoritmo e o valor do dado real. Neste trabalho, utilizou-se as seguintes métricas estatísticas para mensurar o desempenho: *mean absolute error* (MAE), *root mean square error* (RMSE) e *mean absolute percentage error* (MAPE). Além disso, foi utilizado o delta percentual (Δ) para avaliar se o modelo tende a subestimar ou superestimar o valor previsto, se é otimista ou pessimista.

2.2.1 Mean absolute error (MAE)

O MAE, sigla do inglês para *mean absolute error* ou média do erro absoluto mede o erro absoluto de cada previsão e é dado por: [HEWAMALAGE et al., 2022](#)

$$MAE = \frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n} \quad (2.1)$$

2.2.2 Root mean squared error (RMSE)

A RMSE, sigla para *root mean squared error* é semelhante à MAE, contudo eleva os erros ao quadrado antes de somá-los e tira a raiz logo depois. A RMSE é, por tanto, mais sensível a *outliers*. [HEWAMALAGE et al., 2022](#)

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}} \quad (2.2)$$

2.2.3 Mean absolute percentage error (MAPE)

Foi utilizada também a MAPE, *Mean absolute percentage error*, para mensurar a escala do erro em relação ao tamanho das medições.

$$MAPE = \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right| \quad (2.3)$$

2.2.4 Delta percentual

O delta percentual, Δ , é utilizado para mensurar se o modelo apresenta tendência de subestimar ou superestimar a variável, se é otimista ou pessimista.

$$\Delta = \frac{\hat{y}_i - y_i}{y_i} \quad (2.4)$$

Capítulo 3

Experimentos

Para cada um dos modelos selecionados foram realizados experimentos alterando parâmetros e os dados visando melhoria de performance. O treino dos modelos foram realizados com dados de 2003 até junho de 2017, os testes, por sua vez, foram realizados com dados de julho de 2017 até dezembro de 2019, dessa forma 85% do conjunto de dados foi usado para treino e 15% para teste. Para cada modelo foram alterados diferentes parâmetros.

3.1 Regressao linear

A regressão linear é um modelo simples de aprendizado de máquina que assume uma relação linear entre as variáveis de entrada e o alvo da previsão. Foram testados métodos de normalização e transformação dos dados, além da remoção de variáveis de alta correlação.

Experimento	Transformação nos dados	Período de dados	Remoção de variáveis
A	nenhum	de 2003 até 2019	não
B	<i>Standard scaler</i>	de 2003 até 2019	não
C	<i>MinMax scaler</i>	de 2003 até 2019	não
D	<i>Power Transformer</i>	de 2003 até 2019	não
E	nenhum	de 2003 até 2019	sim ^a

Tabela 3.1: Experimentos realizados na regressão linear

^a As variáveis removidas foram: PIB *per capita*, INCC, IGP, taxa Selic, IDH Educação e IDH Longevidade, NFSP, preço do saco de cimento e preço da tonelada de cimento

O resultado dos experimentos se encontra na tabela abaixo:

Experimento	MAE	RMSE	MAPE
A	35148.6	55391.1	0.6
B	35148.6	55391.1	0.6
C	35148.6	55391.1	0.6
D	49073.6	67446.7	1.05
E	26435.6	38970.6	0.32

Tabela 3.2: resultado dos experimentos para regressão linear

Observa-se que não houve melhoria de performance ao normalizar os dados com *standard scaler* ou *minmax scaler*, ao utilizar *power transformer*, contudo, houve piora no desempenho. Ressalta-se a grande melhoria nas métricas ao remover variáveis de alta correlação tanto nas métricas de erro quantitativo, como o MAE e RMSE, quanto proporcionalmente ao atingir 32% de erro percentual médio.

Na sessão 4, estão apresentados os resultados e previsões do modelo de regressão linear com melhor desempenho dentre os testados no modelo, ou seja, sem normalização de dados e deletando *features* com alta correlação.

3.2 Redes neurais *Feed Forward*

As redes neurais *feed forward* podem ser construídas com várias arquiteturas e hiperparâmetros. Neste estudo, testou-se alterar a quantidade de camadas da rede, o número de neurônios em cada camada, a função de ativação utilizada, além da quantidade de *epoch*¹ utilizada no treinamento.

3.3 Redes recorrentes

3.4 regressao linear

¹ Uma *epoch* é uma passada pelos dados de entrada PATTERSON e GIBSON, s.d.

Capítulo 4

Resultados

blafgdf .

Capítulo 5

Conclusão

ahahahahahh

Referências

- [ACADEMY 2022] Data Science ACADEMY. *Deep Learning Book*. disponível em: <https://www.deeplearningbook.com.br>. Acesso em: 17 Dezembro. 2022. 2022 (citado nas pgs. 7–9, 13).
- [BRASIL 2021] IBRAM Mineração do BRASIL. “Votorantim cimentos inicia a operação de nova fábrica em pecém com lançamento de produto no mercado cearense”. 2021. URL: <https://ibram.org.br/noticia/votorantim-cimentos-inicia-a-operacao-de-nova-fabrica-em-pecem-com-lancamento-de-produto-no-mercado-cearense/> (citado na pg. 1).
- [CARVALHO 2008] Maria Beatriz Maury de CARVALHO. “Impactos e Conflitos da Produção de Cimento no Distrito Federal”. Diss. de maestr. Universidade de Brasília, 2008 (citado na pg. 1).
- [CIMENTO PORTLAND 2002] Associação Brasileira de CIMENTO PORTLAND. “Guia básico de utilização do cimento portland”. 2002 (citado na pg. 1).
- [ESSELMAN 2022] Amy ESSELMAN. “What is a boxplot?” 2022. URL: <https://www.storytellingwithdata.com/blog/what-is-a-boxplot> (citado na pg. 19).
- [GOODFELLOW *et al.* 2016] Ian GOODFELLOW, Yoshua BENGIO e Aaron COURVILLE. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016 (citado nas pgs. 3–5).
- [HEWAMALAGE *et al.* 2022] Hansika HEWAMALAGE, Klaus ACKERMANN e Christoph BERGMEIR. *Forecast Evaluation for Data Scientists: Common Pitfalls and Best Practices*. 2022. DOI: [10.48550/ARXIV.2203.10716](https://doi.org/10.48550/ARXIV.2203.10716). URL: <https://arxiv.org/abs/2203.10716> (citado na pg. 21).
- [HOAGLIN 2013] David C. HOAGLIN. “Volume 16: how to detect and handle outliers”. Em: 2013 (citado na pg. 18).
- [HOCHREITER e SCHMIDHUBER 1997] Sepp HOCHREITER e Jürgen SCHMIDHUBER. “Long short-term memory”. Em: *Neural computation* 9 (dez. de 1997), pgs. 1735–80. DOI: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735) (citado na pg. 11).

- [HYNDMAN e ATHANASOPOULOS 2021] Rob J HYNDMAN e George ATHANASOPOULOS. *Forecasting: principles and practice*. 3ª ed. OTexts.com/fpp3. Accessed on 17 December 2022. OTexts, 2021 (citado na pg. 6).
- [JULIO DA MOTTA SINGER 2022] Pedro Alberto Morettin e JULIO DA MOTTA SINGER. *Estatística e ciência de dados*. 2022.
- [LIMA PERESSIM 2021] Felipe de LIMA PERESSIM. “Previsão da resistência à compressão do cimento”. Diss. de maestr. Instituto de Matemática e Estatística da Universidade de São Paulo, 2021 (citado na pg. 1).
- [NIELSEN 2015] Michael A. NIELSEN. *Neural Networks and Deep Learning*. Determination Press, 2015.
- [NORMAS TÉCNICAS 2018] Associação Brasileira de NORMAS TÉCNICAS. “Cimento Portland — requisitos”. Em: 2018 (citado na pg. 1).
- [PATTERSON e GIBSON s.d.] Josh PATTERSON e Adam GIBSON. *Deep Learning: A practitioner’s approach* (citado nas pgs. 3, 4, 7–9, 11, 24).
- [POOLE *et al.* 1998] David POOLE, Alan MACKWORTH e Randy GOEBEL. *Computational Intelligence: A Logical Approach*. Oxford University Press, 1998.
- [RAMACHANDRAN *et al.* 2017] Prajit RAMACHANDRAN, Barret ZOPH e Quoc V. LE. *Searching for Activation Functions*. 2017. DOI: [10.48550/ARXIV.1710.05941](https://arxiv.org/abs/1710.05941). URL: <https://arxiv.org/abs/1710.05941> (citado nas pgs. 9, 10).
- [RASCHKA 2014] Sebastian RASCHKA. “About feature scaling and normalization - and the effect of standardization for machine learning algorithms”. 2014. URL: https://sebastianraschka.com/Articles/2014_about_feature_scaling.html.
- [RUSSELL e NORVIG s.d.] Stuart J. RUSSELL e Peter NORVIG. *Artificial Intelligence: A Modern Approach, third edition* (citado na pg. 3).
- [SILVA BARBOZA 2016] Lucas da SILVA BARBOZA. “Estudo sobre o impacto da redução do consumo de cimento no comportamento mecânico do concreto autoadensável”. Diss. de maestr. Universidade Federal de São Carlos, 2016.
- [SIQUEIRA 2021] Daniel SIQUEIRA. “Melhorando a análise com o boxplot”. 2021. URL: <https://www.alura.com.br/artigos/melhorando-a-analise-com-o-boxplot> (citado na pg. 19).
- [SNIC 2021] SNIC. “Relatório nacional 2021 sindicato nacional da indústria do cimento”. 2021 (citado na pg. 16).
- [STECK *et al.* 2021] Harald STECK *et al.* “Deep learning for recommender systems: a netflix case study”. Em: *AI Magazine* 42.3 (nov. de 2021), pgs. 7–18. DOI: [10.1609/aimag.v42i3.18140](https://ojs.aaai.org/index.php/aimagazine/article/view/18140). URL: <https://ojs.aaai.org/index.php/aimagazine/article/view/18140> (citado na pg. 3).

REFERÊNCIAS

- [TUKEY 1977] John W. TUKEY. *Exploratory Data Analysis*. Addison-Wesley, 1977 (citado na pg. 18).
- [TURING 1950] A. M. TURING. “Computing machinery and intelligence”. Em: *Mind* 59.236 (1950).
- [VASCONCELOS 2022] Ieda VASCONCELOS. *Informativo Econômico PIB*. 2022. URL: <https://cbic.org.br/wp-content/uploads/2022/03/informativo-economico-pib-04-marco-2022.pdf> (citado na pg. 1).
- [WILLIAMSON *et al.* 1989] D WILLIAMSON, RA PARKER e Juliette KENDRICK. “The box plot: a simple visual method to interpret data”. Em: *Annals of internal medicine* 110 (jul. de 1989), pgs. 916–21. DOI: [10.1059/0003-4819-110-11-916](https://doi.org/10.1059/0003-4819-110-11-916) (citado na pg. 19).
- [ZHANG *et al.* 2021] Aston ZHANG, Zachary C. LIPTON, Mu LI e Alexander J. SMOLA. “Dive into deep learning”. Em: *arXiv preprint arXiv:2106.11342* (2021) (citado na pg. 4).