UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO – UFERSA
CURSO DE LICENCIATURA EM COMPUTAÇÃO
Trabalho de Conclusão de Curso (2023).

PREDIÇÃO DO ÍNDICE DE CO_2 POR MEIO DE APLICAÇÃO DE IA

David Franklin Pessoa Ferreira[1]
Matheus da Silva Menezes[2]

RESUMO

O presente trabalho tem como objetivo demonstrar a eficiência de modelos de aplicação de IA na predição de dados futuros. Para isso, usa-se a compilação dos índices de CO_2 na atmosfera, disponibilizados pela NASA, e monitorados pelo NOAA, em página de *web* específica. Neste experimento, utiliza-se a ferramenta *Scikit-learn*, para treinar uma IA, e confronta-se os resultados obtidos com outros modelos, como a aplicação de um modelo com base na teoria do aprendizado de máquina, gerado para o este trabalho; além da validação destes modelos, através do modelo matemático do Método dos Mínimos Quadrados (MMQ). Ao final do experimento, pode-se destacar a eficiência da biblioteca *Scikit-learn* e do MMQ; além da pontual previsibilidade do modelo de aprendizagem de máquina com base na teoria.

Palavras-chave: inteligência artificial; co_2; aprendizado de máquina; regressão linear; método dos mínimos quadrados.

1 INTRODUÇÃO

O oxigênio é um dos bens mais preciosos para os seres vivos. Do primeiro respirar até o último, toda uma existência se firma. Todos os seres vivos na face da terra, necessitam desse elemento primordial para vida que permeia nosso ar e nossa atmosfera. Porém, nosso ar e atmosfera não são compostos unicamente por oxigênio respirável (O_2); Outros elementos compõem nossa atmosfera, alguns destes elementos funcionam como uma barreira protetora para nós seres vivos; uma destas barreiras é a Camada de Ozônio (composta de ozônio, O_3) que encontra-se a cerca de 25 a 30 km acima da superfície terrestre, na estratosfera. A Camada de Ozônio nos protege dos raios solares, como um filtro para os raios ultravioletas (raios UV). Um dos fatores que leva ao enfraquecimento da Camada de Ozônio, é a presença excessiva de gases danosos (Gases do Efeito Estufa - GEE) como Óxido Nítrico (NO), Óxido Nitroso (N_2O), Metano (CH_4) e o Dióxido de Carbono (CO_2). Este, (“*representa mais de 70% das*

1 David Franklin Pessoa Ferreira - Graduado em Curso Superior de Tecnologia em Marketing de Vendas - UNP; Graduando em Licenciatura em Computação - UFERSA. Lates: http://lattes.cnpq.br/6483985204821792

2 Matheus da Silva Menezes - Doutor em Sistemas e Computação - UFRN. Lates: http://lattes.cnpq.br/7790866637385232

emissões de GEE") por meio, também, da queima de combustíveis fósseis como carvão e petróleo (gasolina e diesel são exemplos de derivados de petróleo). Para se ter uma ideia, a terra teve um aumento significativo na sua temperatura, cerca de 1,1ºC, se comparado com as temperaturas médias do final do século XIX[1]; algo que pode, potencialmente, causar desastres climáticos, tais como secas intensas, escassez de água, incêndios severos, aumento do nível do mar, inundações, derretimento do gelo polar, tempestades e declínio da biodiversidade[1].

Portanto, o monitoramento dos GEE, em especial do CO_2, pode nos colocar na vanguarda na busca por soluções deste problema. Prever quais serão os próximos níveis de CO_2 na atmosfera, se mantida a tendência atual, pode nos ajudar a compreender o cenário futuro e a planejar ações mitigadoras. Para isso, precisa-se de modelos matemáticos que consigam prever o futuro olhando para o passado. Nesse contexto, as aplicações de IA tem demonstrado grande eficácia. Ao se treinar, de forma correta, uma aplicação com dados já consolidados, e verifica-se sua precisão comparando suas predições com dados reais, pode-se, enfim, ter um modelo matemático de rápida resposta para prever, com grau satisfatório de precisão e confiabilidade, de quais serão as próximas estimativas de CO_2 na atmosfera. Sendo assim, o objetivo deste trabalho é elaborar e treinar uma aplicação de IA, tendo como entrada, dados compilados, entre os anos de 1958 até os dias atuais (janeiro de 2023), pela agência Norte-americana de Administração Nacional da Aeronáutica e Espaço (em inglês: *National Aeronautics and Space Administration - NASA*)[2]; utilizando a classificação por Regressão, considerando ajustes linear e alguns casos não lineares, para que, assim, possa-se prever os próximos indicadores de emissão de CO_2 na atmosfera terrestre, considerando a manutenção da tendência de comportamento temporal desta medida.

2 FUNDAMENTAÇÃO TEÓRICA

Ao longo dos anos o ser humano busca maneiras de evoluir tecnologicamente[11]. O uso de combustíveis fósseis gerou uma série de avanços relevantes para nossa sociedade, todavia, como consequência contribuiu para o aumento da temperatura na superfície terrestre, causado pelo efeito estufa[5][10].

Diametralmente oposto, ainda, o avanço da tecnologia na área da computação, como a exemplo, as aplicações em IA, nos permite direcionar esforços e energia para tarefas menos repetitivas e cognitivas, das quais, tais tarefas ficam a cargo da IA, e a ação humana pode ser direcionada para análise dos dados obtidos, por exemplo.

Nesta etapa do presente trabalho, apresenta-se, brevemente, o Método dos Mínimos Quadrados na seção 2.1; uma breve descrição sobre *Machine Learning* (aprendizagem de máquina) na seção 2.2; e, por fim, discorre-se sobre o CO_2 e a compilação dos dados obtidos para formulação deste.

2.1 Método dos Mínimos Quadrados (MMQ)

Segundo [8], "*dado uma dispersão de pontos em um plano cartesiano XY, onde esses vários pontos não tem uma função que represente seu comportamento. Então o Método dos Mínimos Quadrados (linear) consiste em estabelecer a equação da reta, sob a função G(x) = ax+b, que melhor se ajusta aos pontos.*"

Dada definição, discorre-se sobre os casos estudados e usados neste trabalho, sabendo-se que o objetivo deste trabalho não é descrever de forma minuciosa as equações e funções matemáticas trabalhadas e já amplamente difundidas no meio acadêmico.

Explicado isso, faz-se uma breve descrição das funções de ajuste dos casos envolvidos neste, iniciando-se com Caso Linear, na subseção 2.1.1; na subseção 2.1.2 trata-se do Caso Logarítmico; na subseção 2.1.3 o Caso Exponencial; por fim, na subseção 2.1.4 o Caso Potência. Ademais, esclarece-se que todas as funções não lineares, tem como base a função linear.

2.1.1 Caso Linear

A função de ajuste para o caso linear, nossa função (1), se dá por:

$$G(x)=ax+b \tag{1}$$

Do qual o erro mínimo é dado pela equação:

$$\sum_{i=1}^{n}(y_i-ax_i-b)^2$$

As derivadas parciais do erro mínimo em relação a a e b, em seu ponto mínimo, são nulas, então:

$$\frac{\partial E}{\partial a}=0 \quad \text{e} \quad \frac{\partial E}{\partial b}=0$$

Derivando em função de a, encontra-se a equação (2):

$$\frac{\partial E}{\partial a}=\frac{\partial \sum (y_i-ax_i-b)^2}{\partial a}=0$$

Desenvolvendo esta equação, chega-se a:

$$\frac{\partial E}{\partial a}=a\sum (x_i)^2+b\sum x_i=\sum x_i y_i \tag{2}$$

Derivando em relação de b, tem-se a equação (3):

$$\frac{\partial E}{\partial b}=\frac{\partial \sum (y_i-ax_i-b)^2}{\partial b}$$

Desenvolvendo esta equação, chega-se a:

$$\frac{\partial E}{\partial b}=a\sum x_i+nb=\sum y_i \tag{3}$$

Por meio das equações (2) e (3), pode-se formar o seguinte sistema linear:

$$\begin{cases} a\sum x_i^2 + b\sum x_i = \sum x_i y_i \\ a\sum x_i + nb = \sum y_i \end{cases}$$

Ao isolar o coeficiente *b* na equação (3), encontra-se a equação (4):

$$b = \frac{\sum y_i - a\sum x_i}{n} \tag{4}$$

Caso se queira encontrar o valor do coeficiente *a* da equação utilizando apenas os dados tabelados, basta substituir a equação (4) na (2), assim tem-se a equação (5):

$$a = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - \left(\sum x_i\right)^2} \tag{5}$$

Quando se isola o *a* em (3), obtém-se a equação (6):

$$a = \frac{\sum y_i - nb}{\sum x_i} \tag{6}$$

Mais uma vez, caso se queira encontrar o valor do coeficiente *b* apenas com os dados tabelados, basta substituir a equação (6) na equação (2), encontrando a equação (7):

$$b = \frac{\sum x_i \sum x_i y_i - \sum y_i \sum x_i^2}{\left(\sum x_i\right)^2 - n\sum x_i^2} \tag{7}$$

Com isso, consegue-se encontrar as equações adequadas para se achar os coeficientes *a* e *b*, apenas com os dados tabelados, por meio das equações (5) e (7), respectivamente.

2.1.2 Caso Logarítmico

A função de ajuste para o caso logarítmico, nossa função (8), tem como base a função do caso linear, função (1), e será dada por:

$$G(x) = a.\ln(x) + b \tag{8}$$

Dada a similaridade, com função (1), todo o processo pode ser reescrito alterando apenas *x* por *ln(x)*. De forma prática, reescreve-se apenas as equações (5) e (7), o que gerará nossas equações (9) e (10), respectivamente:

$$a = \frac{n\sum ln(x_i).\, y_i - \sum ln(x_i)\sum y_i}{n\sum ln(x_i)^2 - [\sum ln(x_i)]^2} \tag{9}$$

$$b=\frac{\sum ln(x_i)\sum ln(x_i).y_i-\sum y_i\sum ln(x_i)^2}{[\sum ln(x_i)]^2-n\sum ln(x_i)}$$

(10)

2.1.3 Caso Exponencial

A função de ajuste para o caso exponencial, nossa função (11), será dada por:

$$G(x)=b.e^{ax}$$

(11)

Como informado no caput deste tópico, 2.1, todas as funções não lineares terão como base a função (1), para isso, deve-se aplicar logaritmo natural nos dois lados da função (11), logo, tem-se a seguinte expressão:

$$y=b.\mathrm{e}^{ax}$$

$$\ln(y)=\ln(b.\mathrm{e}^{ax}) \quad \rightarrow \quad \ln(y)=\ln(b)+\ln(\mathrm{e}^{ax}) \quad \rightarrow \quad \ln(y)=\ln(b)+ax.\ln(\mathrm{e})$$

Uma vez que *ln(e) = 1*, chega-se a seguinte função (12):

$$\ln(y)=\ln(b)+ax$$

(12)

Sendo a função (12), similar a função (1), pode-se aplicar o mesmo raciocínio para se obter os coeficientes *a* e *b*, substituindo o *y* por *ln(y)*, assim, para se obter o coeficiente *a*, pode-se adequar a equação (5) e gerar a equação (13), da seguinte forma:

$$a=\frac{n\sum x_i.\ln(y_i)-\sum x_i\sum \ln(y_i)}{n\sum x_i^2-(\sum x_i)^2}$$

(13)

Agora, vai-se determinar o coeficiente *b*, usando o mesmo raciocínio da equação (7), fazendo a devida sobreposição do *y* pelo *ln(y)*, chega-se a equação (14):

$$\ln(b)=\frac{\sum x_i\sum x_i.\ln(y_i)-\sum \ln(y_i)\sum x_i^2}{(\sum x_i)^2-n\sum x_i^2}$$

(14)

Contudo, o valor encontrado é *ln(b)*, logo, é necessário calcular o exponencial do valor obtido, resultando na expressão (15):

$$b=e^{ln(b)}$$

(15)

2.1.4 Caso Potência

A função de ajuste para o caso potência, será dada por:

$$G(x)=b.x^{a} \tag{16}$$

Mais uma vez, precisa-se aproximar a função (16) da função (1), para isso usa-se o mecanismo exposto na expressão (11), o que resultará na função (17), adiante:

$$\ln(y)=\ln(b.x^{a}) \quad \rightarrow \quad \ln(y)=\ln(b)+\ln(x^{a}) \quad \rightarrow \quad \ln(y)=\ln(b)+a.\ln(x) \tag{17}$$

Uma vez, que nossa função, (17), se tornou similar a função (1), pode-se aplicar os mesmos passos para se obter os coeficientes *a* e *b*, agora, substituindo o *x* por *ln(x)* e o *y* por *ln(y)*.

Começa-se com a sobreposição destes na equação (5), para se determinar o coeficiente *a*, isso resultará na equação (18), adiante:

$$a=\frac{n\sum \ln(x_i)\ln(y_i)-\sum \ln(x_i)\sum \ln(y_i)}{n\sum \ln(x_i)^2-[\sum \ln(x_i)]^2} \tag{18}$$

Agora, encontrar-se o coeficiente *b*, por meio da adequação da equação (7), o que resultará na nossa equação (19). Entretanto, deve se tomar o cuidado de lembrar, que estar-se encontrando *ln(b)*, nesse momento:

$$\ln(b)=\frac{\sum \ln(x_i)\sum \ln(x_i).\sum \ln(y_i)-\sum \ln(y_i)\sum \ln(x_i)^2}{[\sum \ln(x_i)]^2-n\sum \ln(x_i)^2} \tag{19}$$

De forma análoga ao ocorrido no caso exponencial, equação de (14), mais uma vez se faz necessário calcular o exponencial do valor obtido, utilizando a equação (15), vista anteriormente:

$$b=e^{ln(b)}$$

2.2 *Machine Learning* (Aprendizado De Máquina – ML)

Antes de adentrar no tema *Machine Learning* (ML), vai-se enveredar no campo da cognição, explicando o processo de aprendizado humano e posteriormente aplicando o conceito à computação.

> *Para [4], O Processo de Aprendizagem Humana é mecanismo pelo qual o conhecimento é adquirido, compreendido e incorporado. Ele pode ser descrito como um conjunto de transformações mentais que ocorrem em resposta a estímulos ou experiências. Em geral, processo de aprendizagem envolve três fase: Aquisição, Retenção e Recuperação*
> *(…) Existem vários modelos e teorias que tentam explicar como o processo de aprendizagem ocorre, incluindo teorias cognitivas, neurobiológicas e*

> *comportamentais. Além disso, existem diferentes tipos de aprendizagem, como aprendizagem declarativa, não-declarativa, por condicionamento por observação, entre outras.*

Feitas as considerações preliminares sobre o que é aprendizagem a luz da cognição humana, retorna-se para a área da computação para se fazer o paralelo de tal conceito.

O ML, ou Aprendizado de máquina, é uma subárea da Inteligência artificial que trabalha com a busca por padronização de dados. Tem-se o seguinte conceito sobre: *"O Aprendizado de Máquina, é um campo da ciência da computação, que se concentra em criar sistemas, que são capazes de aprender, a partir de um conjunto de dados, sem precisar de programação direta[4]"*.

Dito isto, passa-se a tratar melhor do assunto no tópico 2.2.1.

2.2.1 Aprendizado de máquina

Como fora adiantado, o Aprendizado de máquina é uma subárea da IA, cujo objetivo é tentar gerar uma forma sintética do qual um algoritmo possa se automodelar, aprendendo no processo. Na verdade, o algoritmo prevê os padrões e passa ajustar suas variáveis e parâmetros de forma a criar uma função de ajuste para a proposta da aplicação. Encontra-se o seguinte conceito sobre, em [9]:

> *O aprendizado de máquina (incluindo aprendizado profundo) é um estudo de algoritmos de aprendizado. Diz-se que um programa de computador aprende com a experiência E em relação a alguma classe de tarefas T e medida de desempenho P se seu desempenho em tarefas em T, medido por P, melhora com a experiência E[9].*

O processo de ML, ocorre quando a IA é treinada, ao ser exposto a uma boa base de dados. Com advento da *Big Date* é possível treinar aplicações de IA com mais precisão e eficiência. Existem vários tipos de ML, tais como, aprendizado supervisionado, não-supervisionado e por reforço. Esses tipos de aprendizagem, quando se olha melhor, são análogos a forma como a humanidade aprende.

Para se entender melhor, descreve-se as etapas no processo de aprendizagem de uma IA, na qual pode-se encontrar em[4]:

- Coleta e preparação dos dados;
- Escolha do algoritmo/modelo e configuração dos parâmetros;
- Treinamento do modelo com os dados de treinamento;
- Avaliação do modelo com dados de teste;
- Ajuste do modelo e retreinamento, se necessário;
- e Uso do modelo para fazer previsões sobre novos dados.

2.2.2 Regressão

A regressão é uma subcategoria da Classificação. Encontra-se uma definição em [9]:

> *Um programa de computador prevê a saída para a entrada dada. Os algoritmos de aprendizado normalmente produzem uma função f: Rn → R. Um exemplo desse tipo de tarefa é prever o valor do sinistro de uma pessoa segurada (para definir o prêmio do seguro) ou prever o preço do título[9].*

Em *machine learning*, a regressão é um tipo de algoritmo que é usado para prever um valor numérico com base em dados de entrada. É um método estatístico que estuda a relação entre uma variável dependente (a variável a ser prevista) e uma ou mais variáveis

independentes (as variáveis de entrada). A regressão é usada quando a tarefa é de natureza preditiva, ou seja, quando se deseja estimar um valor numérico com base em dados históricos.

Existem vários tipos de regressão, incluindo a regressão linear simples, onde apenas uma variável independente é usada para fazer a previsão, e a regressão linear múltipla, onde várias variáveis independentes são usadas. Além disso, existem também outros tipos de regressão, como regressão polinomial, regressão logística, regressão de séries temporais, entre outros, que podem ser aplicados em situações específicas.

O objetivo da regressão é encontrar a melhor relação matemática entre as variáveis de entrada e a variável de saída (ou variável dependente), de forma a minimizar o erro de previsão. O modelo de regressão é treinado com dados históricos conhecidos, e uma vez treinado, pode ser usado para fazer predições em dados novos e não vistos, com o objetivo de estimar o valor da variável dependente com base nas variáveis independentes fornecidas. A regressão é amplamente utilizada em diversas aplicações, como previsão de preços de ações, previsão de vendas, previsão de demanda, análise de risco, entre outros.

Em nosso estudo, utilizaremos a aproximação por regressão linear simples, e linearizações das funções logarítmica, exponencial e potência aplicadas à predição de CO_2 na atmosfera a partir de dados conhecidos.

2.3 Dióxido de carbono (CO_2)

A ligação de um átomo de Carbono com dois de Oxigênio forma uma estrutura química denominada de Dióxido de Carbono (CO_2)[5]. O CO_2 foi descoberto pelo escocês Joseph Black em 1975[5]. É possível observá-lo na atmosfera sendo gerado naturalmente ou por intervenção do homem. Na natureza, esse gás é importante, dentre outras funções, para plantas e oceanos.

Segundo [3] *"os processo envolvidos na retirada de CO_2 da atmosfera são fotossíntese e a dissolução de CO_2 nos oceanos, e os principais processos envolvidos no retorno do CO_2 para atmosfera são a oxidação da matéria orgânica pela respiração ou queima e a liberação de CO_2 dos oceanos, nas áreas onde a concentração deste gás é superior em relação à atmosfera."*

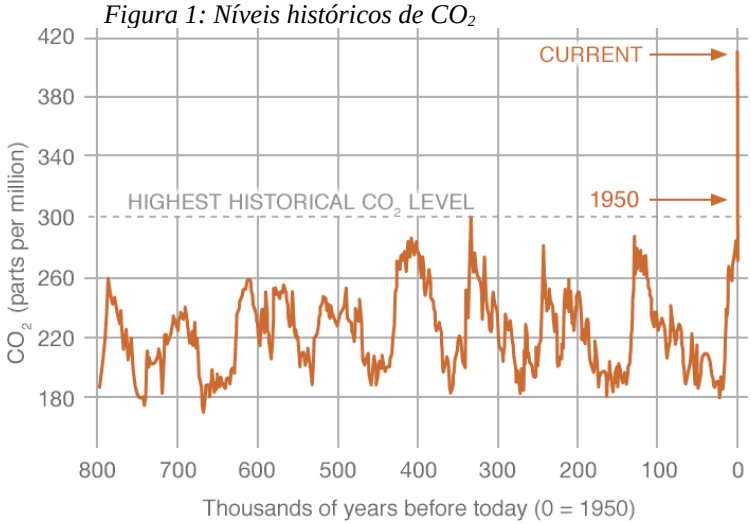
Em contrapartida, como explicado por [3], esses gases são gerados pela decomposição de matéria orgânica, nas erupções vulcânicas[6] e pela liberação das plantas e oceanos. Vale ressaltar, que tal processo é natural, e serve para equilibrar a vida na terra, gerando, assim, o efeito estufa abaixo da Camada de Ozônio.

Diametralmente oposto ao propósito natural da liberação desse gás, a ação antrópica tem sua parcela de contribuição.

Ao observar os dados[2], vê-se que a cada ano aumenta-se a quantidade de CO_2 emitida na atmosfera.

O gráfico adiante mostra os níveis de CO_2 durante os três últimos ciclos glaciais da Terra, que foram obtidos por meio da captura por bolhas de ar presas em camadas de gelo e geleiras.

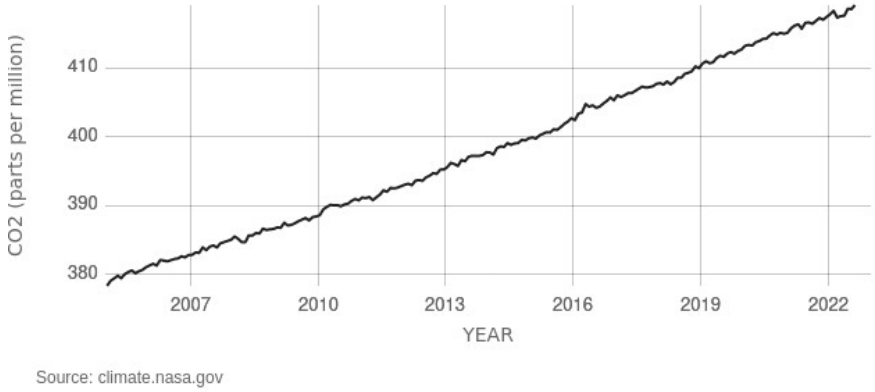
Figura 1: Níveis históricos de CO_2



Fonte: NASA (2022) – Crédito: NOAA

O segundo gráfico mostra os níveis atmosféricos de CO_2 nos últimos anos, com as mudanças naturais e sazonais removidas[6].

Figura 2: Medições diretas de CO_2 (2005-dias atuais)



Fonte: NASA (2022)

Os gráficos apresentados revelam, visualmente, a evolução da concentração de CO_2 na atmosfera.

2.3.1 Dos Dados

Será analisado a concentração média mensal de CO_2 compilados pela NOAA e disponibilizado pela NASA, do qual teve uma aumento do nível de concentração desse gás na atmosfera, saindo de 315.7 partes por milhão por volume (ppmv), em março de 1958, para 418.95ppmv, em dezembro de 2022[2]. Utiliza-se os dados disponíveis em[2], para realização deste trabalho, que compreende a compilação desde março de 1958 até janeiro de 2023.

Para nosso trabalho, considera-se todo o período compilado supramencionado, um total de 779 registros.

3 METODOLOGIA

Os códigos serão desenvolvidos em *Python*, versão 3.8.10, e com o sistema em nuvem da *Google Research* para compilação de códigos, chamada *Colaboratory* ou *Colab*.

Outra informação relevante a ser levantada, é que este trabalho se baseou em um tripé, formado pelas teorias do Método dos Mínimos Quadrados e Aprendizado de máquina (ambos constantes nos capítulos 2.1 e 3.1, respectivamente), mais a experimentação da biblioteca de aprendizagem de máquina de código aberto *Scikit-learn*. Com isso, o experimento pode testar a capacidade e confiabilidade deste recurso ao contrastar com os resultados obtidos via MMQ.

3.1 O Ambiente para Compilação

Dentro do ambiente, foram necessárias as importações de algumas bibliotecas, em suas versões disponíveis no *Colab* atualmente, das quais passo a listar: Importou-se a biblioteca *Pandas*, para trabalhar com os dados; a biblioteca *MatPlotLib*, para criar gráficos e visualizar dados; a biblioteca *Stats* do *software open-source SciPy*, para se lidar com os parâmetros estatísticos; a biblioteca *SeaBorn* para plotar gráficos mais interativos; a biblioteca *NumPy*, para trabalhar com os *arrays* e transformações de números; o pacote *PyPlot* da biblioteca *MatPlotLib*, para gerar figuras, criar uma área de plotagem em figura, plotar algumas linhas em uma área de plotagem, decorar a plotagem com rótulos, dentre outras funções do tipo; a biblioteca *StatsModels*, para se estimar modelos estatísticos e realizar testes estatísticos; a biblioteca *Scikit-learn - SKLearn*, para gerar um modelo de aplicação IA; Diversos pacotes referentes ao *Scikit-learn* fora importados e são detalhados na seção 3.3.2, adiante.

3.2 Método dos Mínimos Quadrados

Trabalhou-se com o MMQ de forma a validar os resultados obtidos por meio da aplicação da teoria do Aprendizado de máquina (ML), em conjunto com uma aplicação gerada a partir da biblioteca *Scikit-learn*. Com isso, trabalhou-se com os ajustes conforme conceito de [8][7], exposto na seção 2.1, por meio dos dados, para estabelecer uma função $G(x)$ cujo o gráfico se ajuste aos pontos observados da melhor forma, sem que ocorra *overfitting* ou *underfitting*.

3.2.1 Caso Linear

Para se trabalhar com os dados no caso linear pelo MMQ, aplicou-se a função $G(x) = ax+b$ depois de descobriu-se a e b, via as equações expostas na seção 2.1.1.

3.2.2 Casos não lineares

Já para se trabalhar com os dados nos casos não lineares pelo MMQ, aplicou-se as funções descritas nas seções 2.1.2, para o caso logarítmico, 2.1.3, para o caso exponencial, e 2.1.4, para o caso potência.

3.3 Aprendizado de Máquina

Para a aplicação do Aprendizado de máquina, utilizou-se o conceito básico (aplicação do Aprendizado de máquina via linhas de códigos escritas manualmente) e o uso da biblioteca *Scikit-learn*, como forma de teste da biblioteca para o experimento em questão e ver o quanto os resultados da nossa aplicação gerada via teoria se aproximava dos resultados obtidos via *Scikit-learn* e dos dados reais.

3.3.1 Conceito básico (a teoria)

A teoria do Aprendizado de máquina prevê o modelo básico de Regressão linear (*wx+b*, que retorna a função $g(x) = a*x+b$).

Tal função recebe os parâmetros *a, x* e *b* e retorna a função *a*x+b*.

Função G(x)

Declare

a	(variável que recebe o valor de *a*)
x	(variável que recebe o valor de *x*)
b	(variável que recebe o valor de *b*)

Leia a, x, b

Função *G_x(a, b, x)*
Retorne *a*x+b*

Em seguida define-se uma função de perda, que recebe os parâmetros *a*, *b*, *x* e *y;* define-se a variável local *n*, que recebe a quantidade de dados analisados como parâmetro *x;* define-se a variável de predição que recebe função *a*x+b*; e retorna, por fim, o somatório da *predição* menos *y*, ao quadrado, multiplicado por *0,5/n*.

Função Perda

Declare

a	(variável que recebe o valor de *a*)
x	(variável que recebe o valor de *x*)
b	(variável que recebe o valor de *b*)
y	(variável que recebe o valor de *y*)
n	(variável com o valor total de *x*)
predição	(variável que receberá a função *G(x)*)
G_x()	(função na variável *a*, *x* e *b*, da equação linear)

Leia a, b, x, y, n, predição, G_x()

Função *Perda(a, b, x, y)*
    n ← total_X
    predição ← G_x()
Retorne *0,5/n * somatório((predição - y)²)*

Dando continuidade, define-se uma função para otimização que também recebe os parâmetros *a, b, x* e *y*; mais uma vez, define-se a variável local *n*, que recebe a quantidade de dados analisados como parâmetro *x*.

Função Otimização

Declare

a	(variável que recebe o valor de *a*)
x	(variável que recebe o valor de *x*)
b	(variável que recebe o valor de *b*)
y	(variável que recebe o valor de *y*)
n	(variável com o valor total de *x*)
predição	(variável que receberá a função *G(x)*)
G_x()	(função na variável *a*, *x* e *b*, da equação linear)
da	(variável da derivada de *a*)
db	(variável da derivada de *b*)
Lr	(variável que recebe o *Learn rate*)

Leia n, predição, G_x(), da, db, a, b, Lr

Função *Otimização(a, b, x, y)*
n ← total_X
predição ← G_x()
*da ← (1.0/n) * somatório((predição - y) * x))*
*db ← (1.0/n) * somatório((predição - y))*
*a ← a – (Lr * da)*
*b ← b – (Lr * db)*
Retorne *a, b*

Por fim, define-se uma função de iteração que retorna *a* e *b* com base no número de *épocas* estipulado.

Função Iteração

Declare

a	(variável que recebe o valor de *a*)
x	(variável que recebe o valor de *x*)
b	(variável que recebe o valor de *b*)
y	(variável que recebe o valor de *y*)
épocas	(variável com o valor de iterações a serem realizadas)
Otimização()	(função que usa derivadas parciais para atualizar os parâmetros *a* e *b*)

Leia a, b, épocas, Otimização()

Função *Iteração(a, b, x, y, épocas)*
Para *i* ***até*** *épocas* ***faça****;*
a, b ← Otimização()
Fim para
Retorne *a, b*

Após esses preparativos, se inicializa os parâmetros utilizando as fórmulas para encontrar *a* e *b*, em seguida se define o valor do *Learn Rate* (Taxa de aprendizado), que passaremos a tratar como *Lr*. Com isso, pode-se iniciar as iterações e por fim plotar os resultados.

Declare

n	(variável com o valor total de *x*)
a	(variável que recebe o valor de *a*)
b	(variável que recebe o valor de *b*)
Lr	(variável que recebe o *Learn rate*)

Iteração()	(função que otimiza os coeficientes *a* e *b* em função da variável épocas)
predição	(variável que receberá a função *G(x)*)
G_x()	(função na variável *a*, *x* e *b*, da equação linear)
perda	(variável que recebe a função *Perda*)
Perda()	(função de perda da diferença de variação média)

Leia n, a, b, Lr, Iteração(), predição, G_X(), perda, Perda()

n ← total_X
a ← (y(n-1) – y0) / (x(n-1) – x0)
*b ← (y0 * x(n-1) – y(n-1) * x0) / (x(n-1) – x0)*
Lr ← 1e-8
a, b ← Iteração()
predição ← G_x()
perda ← Perda()

Escreva *a, b, perda*

Plot *x, y, predição*

Nessa fase, o que mais demonstrou ser o gargalo para aplicação foi a taxa de aprendizado (*Lr*); vez que ao se realizar mudanças de maior, ou menor, escopo que uma *Lr* que variasse entre 10^{-4} e 10^{-8}, geravam resultados não satisfatórios ou mesmo fora do contexto. O que pode caracterizar que a aplicação estava presa em um limite local, ou fora da curva (no caso de um *Lr* alto).

3.3.2 Usando o *Scikit-learn*

O *Scikit-learn* é uma biblioteca de código aberto, disponibilizada e mantida, entre outras empresas, pela *Google*. Utilizou-se no *Colab*, a versão disponível da biblioteca na data atual.

Para nosso experimento, utilizou-se os pacotes *RidgeCV, LassoCV, LinearRegression*, *ElasticNet* da biblioteca *Linear Model* para trabalhar com os modelos de ajuste linear; *StandardScaler* da biblioteca *Preprocessing*, para normalizar os dados; *train_test_split* da biblioteca *model_select*, para separar os dados de Treino e Teste; *GridSearchCV* da biblioteca *Model_Selection*, para se realizar uma busca sobre valores de parâmetro especificados para um estimador; *R2_Score* da biblioteca *Metrics*, para se definir o coeficiente de determinação na regressão linear; *SVR* da biblioteca *SVM*, que é a forma de regressão do *SVM*; *RandomForestRegressor*, para se ter um meta estimador que ajuste uma série de árvores de decisão de classificação em várias subamostras do conjunto de dados e usar a média para melhorar a previsão preditiva e controlar o ajuste excessivo, e *GradientBoostingRegressor* para construir um modelo aditivo de uma maneira progressiva, que vai permitir a otimização de funções de perda diferenciáveis arbitrárias, ambos da biblioteca *Ensemble*.

3.4 Os Dados Analisados

Como relatado na seção 2.3.1, os dados são mantidos e monitorados pelo NOAA periodicamente, desde março de 1958, sendo realizada várias coletas por mês; dessas coletas mensais, é feito uma média mensal, que totalizam 12 médias por ano; A base de dados

compilados continham 779 registros, no momento da realização dos testes (cabe destacar que trabalhou-se diretamente com os dados do site [2] e a cada mês o site recebe um novo índice coletado, logo, no mês seguinte da realização dos testes, o número de registro se alterará, aumentando).

3.5 O Coeficiente de Determinação (R^2)

De forma a mensurar, se o modelo aplicado está aceitável ou não, precisa-se de uma ferramenta que nos dê tal segurança, é nesse contexto que entra o coeficiente de determinação, ou seja, ele vai mensurar a qualidade do ajuste.

Esse coeficiente é dado pela equação:

$$R^2 = \frac{SQreq}{SQtotal} \tag{20}$$

No qual:

$$SQreq = \sum \left(G(x_i) - \bar{y}\right)^2 \tag{21}$$

$$SQtotal = \sum \left(x_i - \bar{y}\right)^2 \tag{22}$$

$$\bar{y} = \frac{1}{n} \sum y \tag{23}$$

Com isso, ($0 \leqslant R^2 \leqslant 1$), sendo, a grosso modo, os valores próximos a 0 ajustes ruins, e os valores próximos a 1 ajustes desejáveis.

3.6 Raiz do Erro Quadrático Médio (*Root Mean Squared Error* - RMSE)

Apesar de ter-se posto o R^2 como forma de mensurar a qualidade dos modelos empregados no presente trabalho, decidiu-se por buscar uma validação mais palpável. Para isso, recorreu-se a uma das métricas mais amplamente usadas para aferição de acurácia de modelos preditivos de IA, o *root mean squared error*, ou RMSE. Tal métrica parte da premissa simples, fazer a média dos erros (do qual subtrai-se o valor real de *y* do valor predito de *y*) elevado ao quadrado, e por fim tirando a raiz da soma.

Pode-se melhor entender tal métrica ao expressá-la:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2} \tag{24}$$

Tem-se o seguinte conceito sobre em [12]: "*RMSE (Root Mean Squared Error) é uma medida de erro absoluto que eleva os desvios ao quadrado para impedir que os desvios*

positivos e negativos se cancelem. Essa medida também tende a superestimar erros grandes, o que pode ajudar a eliminar os métodos com esses erros".

Nessa métrica, quanto menor for seu resultado, melhor a qualidade do modelo.

4 RESULTADOS E DISCUSSÕES

Agora, revela-se os resultados obtidos neste presente trabalho. Cabe salientar, que como forma de melhor apresentar os resultados obtidos, esses serão expostos por meio de quadros, referente a cada um dos casos trabalhados.

4.1 Resultados Obtidos

Os resultados obtidos foram:

Quadro 1 - Resultados obtidos (Linear)

		MMQ	***ML***	***Scikit-learn***
Linear	**a**	1.61647	1.59446	1.61541
	b	-2859.99243	-2806.56941	-2857.88608
	R^2	0.97672	-	0.97671
	RMSE	4.7460	12.6516	3.3811
	Perda	-	11.04976	-

Fonte: Autoria própria (2023)

Agora, o enquadramento dos resultados obtidos no método não linear logarítmico:

Quadro 2 - Resultados obtidos (Logarítmico)

		MMQ	***ML***	***Scikit-learn***
Logarítmico	**a**	3215.72819	3185.55357	3213.54814
	b	-24069.38800	-23830.16038	-24052.83998
	R^2	0.97559	-	0.97558
	RMSE	4.8605	12.9686	3.3882
	Perda	-	61.44547	-

Fonte: Autoria própria (2023)

Em seguida, o enquadramento dos resultados obtidos no método não linear exponencial:

Quadro 3 - Resultados obtidos (Exponencial)

		MMQ	***ML***	***Scikit-learn***
Exponencial	**a**	0.00449	0.00292	0.00449
	b	0.04678	0.05911	0.04688
	R^2	0.98495	-	0.98492
	RMSE	3.7763	9.8087	0.0098
	Perda	-	0.00049	-

Fonte: Autoria própria (2023)

Por fim, o enquadramento dos resultados obtidos no método não linear potência:

Quadro 4 - Resultados obtidos (Potência)

		MMQ	***ML***	***Scikit-learn***
Potência	**a**	8.93430	8.67870	8.93194
	b	1.19707	-0.00602	1.21860
	R^2	0.98407	-	0.98410
	RMSE	3.8833	534949.9047	0.0546
	Perda	-	1802.54102	-

Fonte: Autoria própria (2023)

4.2 Erro no Conjunto de Teste

Como forma de se demonstrar a eficiência do experimento, faz-se comparações entre o valor real mensurado pela NOAA e as estimativas dos modelos por nós empregados.

Ao final, faz-se as predições propostas pelos modelos dos meses de janeiro, fevereiro, março e abril do corrente ano, que estão sublinhado na tabela.

Quadro 5 - Conjunto teste

Período	***Valor real***	***Estimativas em ppmv***		
		MMQ (*Exponencial*)	**ML (*Exponencial*)**	***Scikit-learn* (*Exponencial*)**
03/1958	315.70	308.22	315.70	308.22
03/1959	316.65	309.61	317.08	309.61
03/1960	317.58	311.01	318.48	311.01
03/1965	320.89	318.07	325.53	318.07
03/1970	326.93	325.29	332.75	325.29
03/1975	331.94	332.68	340.13	332.68
03/1980	340.07	340.24	347.66	340.24
03/1985	347.91	347.96	355.37	347.96
03/1990	355.75	355.86	363.24	355.86
03/1995	361.98	363.94	371.29	363.94
03/2000	370.75	372.21	379.52	372.21
03/2005	380.95	380.66	387.93	380.66
03/2010	391.37	389.30	396.52	389.30
03/2015	401.74	398.14	405.31	398.14
03/2020	414.74	407.18	414.29	407.18
03/2022	418.81	410.85	417.94	410.85
12/2022	418.95	412.24	419.31	412.24
PREDIÇÕES ESTIMADAS				
01/2023	*419.47*	*412.39*	*419.46*	*412.39*
02/2023	*420.41*	*412.55*	*419.62*	*412.55*
03/2023	*421.00*	*412.70*	*419.77*	*412.70*
04/2023	*423.28*	*412.86*	*419.92*	*412.86*

Fonte: Autoria própria (2023)

Como visto, apesar de o modelo ML ter seus coeficientes diferentes dos outros dois modelos, ele foi o que melhor se aproximou dos valores reais nos últimos anos, com uma variação em média, nesse período, menor que 1ppmv em seus resultados; muito embora tenha tido altas variações entre os anos de 1965 a 2015, chegando 9.31ppmv em 1995 (março de 1995 o valor real foi de 361.98ppmv, contra a estimativa de 371.29ppmv, do modelo ML).

Já nos modelos MMQ e *SKLearn*, houve uma inversão. Os modelos começaram fazendo estimativas baixas, foi se aproximando do valor real (no período entre 1970 a 1995, quando este voltar a ficar com variações acima de 2ppmv) e voltaram a se afastar dos valores reais a partir de 2010, tendo suas estimativas para o último período tabelado, dezembro de 2022, com uma variação de -6.71ppmv (valor real 418.95ppmv contra a estimativa de 412.24ppmv, de ambos os modelos). As melhores estimativas, no enquadramento, se deu no período de 1985, cuja a variação foi de apenas 0.05ppmv (em março de 1985 o valor real foi de 347.91ppmv, contra a estimativa de 347.96ppmv, em ambos os modelos).

5 CONSIDERAÇÕES FINAIS

Neste trabalho, foi estudado e aplicado o MMQ e a teoria do Aprendizado de máquina, sendo este, em duas vertentes, uma aplicação via teoria e outra por meio de biblioteca *Scikit-learn*.

Os resultados obtidos com os vários métodos foram, em geral, satisfatórios. Nesse contexto, pode-se destacar no MMQ, os casos exponencial e potência, dos quais obtivemos os melhores resultados dessa série. Já via *Scikit-learn* os resultados, também foram todos satisfatórios, sendo, mais uma vez, os casos exponencial e potência os melhores resultados. No método com base na teoria do Aprendizado de máquina, não geramos o coeficiente de determinação, apenas obtivemos os resultados de a e b, próximos aos resultados das outras; ao fazer a aplicação passar por várias iterações com uma taxa de aprendizado fixa, obteve-se uma Função de Perda com o valor de 11.05, quase sem variação, no caso Linear, como demonstrado na seção 4.1 e explicado na seção 3.3.1.

De forma geral o experimento se mostrou satisfatório, pois, os três modelos propostos tiveram bons resultados na predição das estimativas, e apesar de o modelo de Aprendizado de máquina, feito a partir da teoria, ter suas estimativas mais próximas dos índices reais, ela não teve seus coeficientes aproximados dos modelos MMQ e a aplicação feita via biblioteca *Scikit-learn*, que obtiveram um menor erro, sendo esse nosso objeto de estudo.

Ademais, ressalta-se a importância de se manter em mente o controle das estimativas dos Gases do Efeito Estufa, em especial o CO_2 (haja visto esse ter maior parcela nos efeitos desse fenômeno), que tem tido grande responsabilidade nos desastres e catástrofes naturais das últimas décadas, graças ao seu elevado aumento, em função da ação antrópica desordeira. Como reflexão sobre o tema, traz-se a frase atribuída ao pensador Wendell Berry: *"Nós não herdamos o mundo de nossos antepassados, nós a pegamos emprestado dos nossos filhos."*

Com isso, espera-se que esse estudo sirva de base para novas pesquisas e validações dos modelos aqui empregados, assim como outros com a mesma eficiência. Por fim, que este trabalho abra portas para este discente realizar cada vez mais pesquisas, e que cada uma delas seja tão relevante quanto a anterior.

REFERÊNCIAS

[1]**NAÇÕES UNIDAS (Brasil)**. Nações Unidas no Brasil. O que são as mudanças climáticas?. Brasil, mar. 2022. Disponível em: https://brasil.un.org/pt-br/175180-o-que-sao-mudancas-climaticas. Acesso em: 23 set. 2022.

[2]NOAA. **CO2 Measurements: Monthly Measurements from Mauna Loa Observatory**. Havaí: Noaa.gov, 2022. Disponível em: https://gml.noaa.gov/webdata/ccgg/trends/co2/co2_mm_mlo.txt. Acesso em: 21 set. 2022.

[3]RASERA, Maria de Fátima Fernades Lamy. **O papel das emissões de CO2 para a atmosfera, em rios da bacia do Ji- Paraná (RO), no ciclo regional do carbono**. Orientador: Prof. Dr. Alex Vladimir Krusche. 2005. 69 p. Dissertação (Mestra) - Centro de Energia Nuclear na Agricultura, Universidade de São Paulo, Piracicaba-SP, Brasil, 2005. Disponível em: https://www.teses.usp.br/teses/disponiveis/64/64135/tde-31072006-083532/publico/Rasera_MFFL.pdf. Acesso em: 15 out. 2022.

[4]ACADEMY, D. S. **Fundamentos de Inteligência Artificial**. Disponível em: https://www.datascienceacademy.com.br/course/fundamentos-de-inteligencia-artificial. Acesso em: 10 fev. 2023.

[5]SOUZA, Líria Alves de. **Dióxido de Carbono**; Brasil Escola. Disponível em: https://brasilescola.uol.com.br/quimica/dioxido-de-carbono.htm. Acesso em 18 fev. 2023.

[6]NASA. HOLLY SHAFTEL. (ed.). **Vital Signs: carbon dioxide**. Carbon Dioxide. 2022. Editora de Ciências: Susan Callery. Disponível em: https://climate.nasa.gov/vital-signs/carbon-dioxide/. Acesso em: 21 set. 2022.

[7]ALMEIDA, César Guilherme de. **Cálculo Numérico**. Uberlândia-MG: UFA, 2015. 229 p. Disponível em: https://repositorio.ufu.br/bitstream/123456789/25218/1/Calculo%20Numerico.pdf. Acesso em: 23 abr. 2023.

[8]CNUM-019 Método dos Mínimos Quadrados Linear [MMQ]. Apresentação: Matheus da Silva Menezes. Brasil: [S. l.: s. n.], 2018. 1 vídeo (35 min). Publicado pelo canal Matemática Universitária. Disponível em: https://youtu.be/3XPH32srZSQ. Acesso em: 21 set. 2022.

[9]HUAWEI, ICT. **HCIA-AI V3.0 Course**. Disponível em: https://talent.huaweiuniversity.com/portal/courses/HuaweiX+EBG2020CCHW1100087/about. Acesso em: 21 fev. 2023.

[10]PENA, Rodolfo F. Alves. **Combustíveis fósseis**; Brasil Escola. Disponível em: https://brasilescola.uol.com.br/geografia/combustiveis-fosseis.htm. Acesso em 18 fev. 2023.

[11]SOUSA, Rainer Gonçalves. **História dos Combustíveis**; Brasil Escola. Disponível em: https://brasilescola.uol.com.br/historia/historia-dos-combustiveis.htm. Acesso em 18 fev. 2023.

[12]EPM INFORMATION DEVELOPMENT TEAM. Oracle. **Oracle Cloud**: como trabalhar com planejamento preditivo no smart view. Brasil: 2019. 64 p. Disponível em: https://docs.oracle.com/cloud/help/pt_BR/pbcs_common/CSPPU/CSPPU.pdf. Acesso em: 30 jun. 2023.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6023**: informação e documentação: elaboração: referências. Rio de Janeiro, 2018.

ANNUAL ARCTIC SEA Ice Minimum 1979-2021 with Area Graph. [S. l.: s. n.], 2022. 1 vídeo (1 min). Publicado pelo canal NASA Climate Change. Disponível em: https://youtu.be/LBiFcmSOC4w. Acesso em: 23 set. 2022.

CNUM-020 Método dos Mínimos Quadrados - Ajustes não lineares [MMQ]. Apresentação: Matheus da Silva Menezes. Brasil: [S. l.: s. n.], 2018. 1 vídeo (37 min). Publicado pelo canal Matemática Universitária. Disponível em: https://youtu.be/rSAThalXWwQ. Acesso em: 21 set. 2022.

CNUM-021 Método dos Mínimos Quadrados - Calculando o R^2 [MMQ]. Apresentação: Matheus da Silva Menezes. Brasil: [S. l.: s. n.], 2018. 1 vídeo (22 min). Publicado pelo canal Matemática Universitária. Disponível em: https://youtu.be/79CKgf9HvE0. Acesso em: 21 set. 2022.

FRANCO, Neide Bertoldi. Aproximação de Funções: Método dos Mínimos Quadrados. **Cálculo numérico.** São Paulo, 2006.

GLOBAL WARMING from 1880 to 2021. [S. l.: s. n.], 2022. 1 vídeo (1 min). Publicado pelo canal NASA Climate Change. Disponível em: https://youtu.be/haBG2IIbwbA. Acesso em: 23 set. 2022.

OZONE WATCH 2018. [S. l.: s. n.], 2019. 1 vídeo (1 min). Publicado pelo canal NASA Climate Change. Disponível em: https://youtu.be/BL1ZsAlJKXU. Acesso em: 23 set. 2022.

REZENDE, Thyago. **RMSE ou MAE? Como avaliar meu modelo de machine learning?**. Goiânia, 28 nov. 2018. LinkedIn: Thyago Rezende. Disponível em: https://www.linkedin.com/pulse/rmse-ou-mae-como-avaliar-meu-modelo-de-machine-learning-rezende/?originalSubdomain=pt. Acesso em: 4 jun. 2023.