



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

USO DE APRENDIZADO DE MÁQUINA PARA GERAÇÃO DE SÉRIES TEMPORAIS DE IRRADIAÇÃO SOLAR NA SUPERFÍCIE

Rafael Mariano Juvêncio

Relatório de Atividades desenvolvidas durante o período de Iniciação Científica do Programa PIBIC, orientada pelo Dr. André Rodrigues Gonçalves e Dr. Fernando Ramos Martins.

INPE

São José dos Campos

2020

RESUMO DAS ATIVIDADES REALIZADAS

Este projeto de iniciação científica tem como principal objetivo o desenvolvimento de metodologias de aprendizado de máquinas para a previsão de dados de irradiação solar através de séries temporais. Apenas um ano de dados de irradiação solar é pouco para inferirmos se a cidade ou região específica é viável para a instalação de plantas solares para a produção de energia ou outra finalidade desejada, é preciso analisarmos, ao menos, dez anos de dados diversos relacionados a irradiação solar para garantir maior confiabilidade ao estudo para que seja tomada a decisão mais coerente e correta.

O estudante iniciou suas atividades no mês de setembro, cumprindo atividades de revisão bibliográfica e familiarização com o tema de estudo. Realizou pesquisa acadêmica referente ao tema e capacitação no aprendizado das ferramentas que seriam necessárias.

Entretanto, foi absorvido em um dos programas de estágio no Itaú-Unibanco de São Paulo, na área de Estudos e Analytics, o que em função das demais atividades acadêmicas, não permite a continuidade deste vínculo de Iniciação Científica.

1. Motivo/Justificativa

A Empresa de Pesquisa Energética (EPE) em seu Anuário Estatístico de Energia Elétrica publicado em 2020, mostra que no início de 1980 a geração de energia elétrica mundial por fonte (%) para os combustíveis fósseis representava 69,62%, em 2010 foi de 66,5% e em 2017 foi de 64,2%, em contrapartida em termos de Terawatt-hora (*TWh*), considerando todos as fontes, saltou de 8.027 *TWh* para 20.417 *TWh* e 24.344 *TWh*, respectivamente para os anos de 1980, 2010 e 2017. Assim sendo, os padrões de consumo e produção atuais são, ainda, em sua maioria baseados em fontes não renováveis como os combustíveis fósseis, contribuindo e acarretando para problemas como aumento da geração de poluentes, aumento da emissão de gases do efeito estufa e colocam em risco o fornecimento de energia a longos prazos (Goldemberg et al., 2007).

Tendo em vista em vista que é insustentável continuarmos com a utilização de fontes de energia não renováveis desenfreadamente, a preocupação com as mudanças climáticas e meio ambiente têm sido cada vez mais debatidas pelos noticiários, governos, empresas e cientistas em relação aos processos de geração de energia elétrica (Martins et al., 2017). Devido a problemas ambientais, crises como a do petróleo e buscando maior segurança energética, o mundo começou a buscar e a adotar energias alternativas, ou renováveis, em relação àquelas dos combustíveis fósseis, principalmente a partir de 1970 (Simas et al. 2013). Entre as alternativas temos principalmente a biomassa, eólica, hidráulica, geotérmica, nuclear e solar. Quando comparada aos combustíveis fósseis as energias renováveis possuem vantagens como ser abundantes e renováveis em todo o mundo, além de reduzirem emissões de gases como o dióxido de enxofre (*SO₂*) e monóxido de carbono (*CO*) e partículas de poeiras (Wang et al., 2019).

Segundo dados disponibilizados pela Agência Nacional de Energia Elétrica (ANEEL) no ano de 2019, o Brasil atingiu a marca de 7.246,41 megawatts (*MW*) de capacidade instalada sendo que a potência fiscalizada foi de 170.071 *MW*, onde cerca de 127.533,25 *MW* foram a partir de fontes renováveis. A matriz elétrica vigente no Brasil é constituída na maior parte por hidráulica e com complementação térmica, entretanto baixos níveis de reservatórios fazem com que sejam ativadas as

usinas térmicas para complementar o sistema, encarecendo o custo de produção (Nakabayashi, 2014). Além disso existem diversos impactos ambientais como perda de biodiversidade animal e vegetal, perda de área que poderia ser destinada a agricultura, alteração da dinâmica da população local das áreas que serão represadas, assim como impacto nas populações e áreas indígenas e população tradicional, além da interferência no regime hídrico entre outros impactos (Andrade, 2017).

Como alternativa, temos a energia solar fotovoltaica que é obtida pela conversão da luz provinda do Sol em eletricidade, além disso o Brasil possui grandes porções de Terra localizada em áreas intertropicais possuindo grande potencial de energia solar ao longo de todo o ano (Martins et al., 2004). A energia solar relaciona-se com a vida na terra e está presente nos processos químicos, físicos e biológicos, além disso têm efeitos na atmosfera, ionosfera, estratosfera e clima terrestre (Gomez et al., 2018).

A conversão da energia solar é feita através de painéis fotovoltaicos. Para que seja viável a instalação de painéis solares ou para que possa ser feito planos de obtenção, geração e distribuição de energia solar é necessário que seja feita uma detalhada coleta de dados e o passo seguinte é utilizar uma modelagem matemática-computacional eficaz para que se possa realizar estimativas de parte dos diversos cenários possíveis. A modelagem computacional é uma das formas de realizarmos análises de dados obtidos por estações de coleta radiométricos e para tal, é necessário que seja utilizada uma linguagem de programação.

Atualmente existem diversas linguagens como JavaScript, Java, C, C++, Python, Perl, Ruby, Go, Julia entre muitas outras. O Python é uma das principais linguagens de programação, orientada a objetos, com vasta biblioteca disponível, possibilita a utilização de ambientes de desenvolvimento empacotado (IDE), através da adição de módulos pode ser estendido em outras linguagens como C e C++, o Python ainda é capaz de ser utilizado nos principais sistemas operacionais atuais como o Linux, Mac, Windows, Unix, Android e iOS, além disso trata-se de uma linguagem de código aberto.

Atualmente um dos principais usos de programação é para o machine learning, ou em português, aprendizado de máquinas, que tem como objetivo a partir de dados atuais e históricos criar modelos preditivos, em outras

palavras o computador pode aprender através de dados esperados, através de outros dados, imagens, palavras entre outros.

Dentro do machine learning existe o deep learning, ou em português, aprendizado profundo, que é baseado em redes neurais e tentam simular o cérebro humano em relação às formas de compreensão das informações e é realizado uma modelagem computacional para criar modelos de treinamento da máquina como reconhecimento de fala, previsões de dados, reconhecimento de imagens entre outros e essas tecnologias são amplamente utilizadas por grandes empresas, como por exemplo, Google, Amazon, Netflix, IBM, Microsoft e Apple etc.

Algumas das aplicações que podemos fazer uso da metodologia de aprendizado de máquina é em estudo de séries temporais que simplificada se trata de observações realizadas ao decorrer do tempo na qual é preciso realizar uma modelagem computacional para essa estruturação e ela é aplicável nos ramos da meteorologia, energias, economia, finanças, entre outros.

O objetivo do presente trabalho é o desenvolvimento de metodologias de machine learning, para a previsão de dados de irradiação solar através de séries temporais, com base em informações de dados numéricos radiométricos e meteorológicos, para calcular a variância no potencial de energia solar no local de interesse.

FUNDAMENTAÇÃO TEÓRICA

Rotação e Translação

O planeta Terra realiza 2 movimentos importantes, conhecidos como rotação e translação. A rotação é o movimento em que o planeta gira em torno de seu próprio eixo em um período de 24 horas. Esse movimento é responsável pelo ciclo diário com períodos diurnos quando o Sol está percorrendo a abobada celeste em um determinado ponto da superfície e períodos noturnos que ocorrem entre o pôr do Sol e o nascer do Sol no dia seguinte. A translação é o movimento orbital da Terra em torno do Sol que rege o ciclo anual de estações do clima. O solstício e o equinócio são eventos que determinam o início das estações em cada hemisfério. Devido a inclinação do eixo de rotação do planeta praticamente constante, a área que é iluminada pelo Sol em cada um dos hemisférios varia ao longo do ano.

Longitude e Latitude

Os meridianos são linhas imaginárias que cortam a Terra no sentido norte–sul, conectando um polo ao outro. Os paralelos são linhas imaginárias que circundam a Terra no sentido leste–oeste. Paralelos e meridianos são definidos por suas extensões de latitude e longitude, concomitantemente. A longitude é a extensão entre o meridiano de origem e o meridiano local, a distância é dada em graus. Por acordo, toma-se como origem o Meridiano de Greenwich. A latitude de um local é o ângulo, em graus, tomando por base a linha do Equador.

Energia Solar

Dal Pai (2005 apud COLSON, 1973) diz que o Sol é uma esfera gasosa, tendo uma temperatura de 6000 K em sua fotosfera, $1 \times 10^6\text{ K}$ em sua coroa e $1 \times 10^7\text{ K}$ seu interior. A elevada temperatura em seu interior faz com que haja a conversão de hidrogênio em hélio, gerando a emissão de grande quantia de energia eletromagnética. Dos $62 \times 10^3\text{ kW} \cdot \text{m}^{-2}$ emitidos apenas $0,5 \times 10^7$ atinge o topo da atmosfera terrestre, sendo um total de $1,6 \times 10^{24}\text{ kW}$. Segundo Varejão-Silva (2005), a radiação ou energia radiante é a energia eletromagnética que se propaga sem a necessidade da presença de um meio material. A energia radiante se comporta ora por natureza corpuscular conhecida como fóton, ora como ondas eletromagnéticas. Irradiância é o fluxo de energia por unidade de área e por unidade de ângulo sólido a esta área. A Irradiação pode ser entendido como a energia incidente por unidade de área numa superfície.

Leis da Irradiação

Segundo Iqbal (1983), alguns modelos teóricos permitem avaliar os processos físicos que envolvem a radiação emitida por um corpo com temperatura conhecida. A Lei de Planck que equaciona o poder emissivo de um corpo negro em cada comprimento de onda do espectro eletromagnético. A Lei de Stefan Boltzmann é uma integração da lei de Planck e permite determinar o total de energia emitida por

um corpo com temperatura específica. A Lei de Kirchhoff diz que a absorvidade de um corpo negro é igual à sua emissividade. A Lei dos deslocamentos de Wien indica o comprimento de onda em que ocorre máxima emissão de radiação eletromagnética de um corpo negro com a temperatura conhecida. A Lei de Lambert relaciona a absorção de luz com as propriedades do material percorrido pela radiação eletromagnética. A Lei de Beer permite avaliar a atenuação da radiação solar incidente em uma superfície horizontal após atravessar um meio como a atmosfera.

Radiação Difusa, Direta e Global

A radiação difusa é definida com a parcela da radiação solar que incide na superfície após ter sua direção alterada pela interação com os constituintes atmosféricos por processos de espalhamento. A radiação direta é a parcela da radiação que chega à superfície terrestre sem sofrer interações. Segundo Varejão-Silva (2005), a radiação direta e a radiação difusa quando somadas compõe a radiação global.

Séries Temporais

Trata-se de um conjunto de dados observados de forma ordenada ao longo do tempo. Ao analisar um conjunto de séries temporais busca-se entender os mecanismos que geraram a série e predizermos o comportamento futuro dessa série.

Em relação aos mecanismos, podemos: descrever o comportamento dessa série, encontrarmos qual a periodicidade presente na série, tentar obter razões matemáticas para o comportamento da série através da modelagem numérica, assim como a trajetória dessa série.

E ao predizermos o comportamento futuro, possibilita-nos realizar projetos nos mais diversos períodos e tomar as melhores decisões.

Aprendizado de Máquinas

Os elementos básicos para o aprendizado de máquina são: dados, parâmetros e algoritmos. Uma das definições mais famosas é a proposta por Tom Mitchell em seu livro, *Machine Learning*, que diz: “Um programa de computador é dito para aprender com a experiência E com relação a alguma classe de tarefas T e medida de desempenho P , se o seu desempenho em tarefas em T , medida pelo P , melhora com a experiência E ”.

Simplificadamente, utilizando algoritmos alimentados por dados e definindo parâmetros podemos criar formas de fazer às máquinas aprenderem, fazendo com que elas tomem decisões baseadas na “experiência aprendida”.

Aprendizado Profundo

É importante que não haja confusão entre inteligência artificial, machine learning e deep learning. Inteligência artificial é uma grande área que engloba o aprendizado de máquinas e este engloba o aprendizado profundo.

Aprendizado profundo sendo uma subárea do aprendizado de máquina utiliza neurônios matemáticos que tem a função de processar os mais variados tipos de dados que passam para outros neurônios matemáticos e assim sucessivamente até encontrar uma saída, isto é, uma resposta que possua maior probabilidade de ser a correta, simplificadamente ele tenta simular a máquina tenta simular o processamento dos neurônios humanos para aprender.

Modelos de Treinamento

Existem diversos modelos de aprendizado de máquina e alguns deles são: Árvore de Decisões, Classificação Naive Bayes, Regressão Linear de Mínimos Quadrados, Regressão Logística, Support Vector Machine, Ensemble Methods, Clustering, Decomposição em Valores Singulares, Análise de Componentes Principais e Análise de Componentes Independentes.

1. METODOLOGIA E RESULTADOS PARCIAIS

O sistema operacional escolhido para a implementação dos algoritmos e desenvolvimento das atividades trata-se do Linux, com o sistema Ubuntu, por se tratar de um sistema Open Source. O critério de escolha da distro foi baseada na usabilidade do sistema, desempenho geral e possibilidade de uso de GPU (que irá fornecer maior capacidade e velocidade de processamento dos modelos de treinamento) e a distro que atendeu a estes critérios foi o Pop-os que existe desde 2017 e até o momento as duas versões mais recentes são, a 20.04 LTS e a 20.10, a versão escolhida foi a 20.04 LTS por ser a versão baseada no Ubuntu estável.

O IDE para a linguagem de programação Python, foi o Anaconda, devido a fácil instalação dos pacotes, fácil uso de escolha ambientes virtuais, além da interface gráfica de usuário ser bastante intuitiva assim como os recursos disponíveis como o Spyder e Jupyter.

Para a obtenção dos dados necessários, foi utilizado o banco de dados disponíveis pelo INPE, disponibilizados na internet através do site: <http://sonda.ccst.inpe.br/index.html>. Nele obtivemos os arquivos em .csv dos dados ambientais das estações de Petrolina - PE (PTR, ID-11) e São Martinho da Serra - RS (SMS, ID-06). Após a obtenção dos dados, foi realizado o tratamento dos dados, de forma que as colunas que ficassem fossem as de interesse e elas são para cada estação: 'year', 'day', 'min', 'glo_avg', 'dir_avg', 'diff_avg', 'lw_avg', 'par_avg', 'lux_avg', 'tp_sfc', 'humid', 'press', 'rain', 'ws_10m', 'wd_10m'.

O código abaixo representa o código do programa parcialmente, foi utilizado um código complementar para visualizarmos o diagrama solar para a cidade de Petrolina (linhas 17 à 61). Após isso, as linhas seguintes, trata-se do algoritmo para a modelagem dos dados para deixar o arquivo no formato necessário para que fosse pudéssemos aplicar os modelos de treinamento de aprendizado de máquina. Observação: foi também realizado alguns comentários durante o código para que pudesse facilitar a compreensão:

```

1 #dados em relacao a Petrolina:
2 ##latitude: -9,0689
3 ###longitude: -40,3197
4 ####altitude: 387m
5 #####O município de Petrolina localiza-se no semiárido do nordeste brasileiro.
6 ##### Seu clima é tropical semiárido seco e quente na parte norte e semiárido quente estépico na parte sul.
7
8 #bibliotecas
9 import pandas as pd #biblioteca fundamental para os dataframes
10 import numpy as np
11 import pvlib #biblioteca do pvlib
12 import datetime as dt#biblioteca para transformacao de valores em data
13 import matplotlib.pyplot as plt #graficos
14 from pvlib import solarposition #posicao solar
15 from sklearn.linear_model import LinearRegression #da biblioteca sklearn importa a funcao de regressao linear #ainda nao usada
16
17 #pvlib adaptado, para ver o comportamento do diagrama solar
18 tz = 'America/Recife' #como se ttrata do time zone, petrolina e recife tem os mesmos valores
19 lat, lon = -9.0689, -40.3197 #a latitude e a longidade de petrolina
20 datatm = dt.datetime.strptime('21/12', '%d/%m') #para a inicio do verao
21 ndias = datatm.timetuple().tm_yday #numero de dias
22 time = pd.date_range('2018-01-01 12:00:00', '2019-01-01 12:00:00', closed='left', freq='H', tz=tz)
23
24 #equacao do tempo, declinacao, angulo horario
25 et = pvlib.solarposition.equation_of_time_spencer71(ndias)
26 declination = pvlib.solarposition.declination_spencer71(ndias)
27 hour_angle = pvlib.solarposition.hour_angle(time, lon, et)
28 local = pvlib.location.Location(lat, lon, tz, altitude = 0, name ='Petrolina-PE')
29 print(local)
30 print("\n Numero de dias =", ndias, "dias")
31 print("\n Equacao do tempo =", et)
32 print("\n Angulo horario =", hour_angle)
33 print("\n Declinação =", declination)
34
35
36 posicaodosol = solarposition.get_solarposition(time, lat, lon)
37 posicaodosol = posicaodosol.loc[posicaodosol['apparent_elevation']>0, :]
38 axys = plt.subplot(1, 1, 1, projection='polar')
39 points = axys.scatter(np.radians(posicaodosol.azimuth), posicaodosol.apparent_zenith,s=2,
40                      label=None, c=posicaodosol.index.dayofyear)
41 axys.figure.colorbar(points)
42
43 #sequencias
44 for hour in np.unique(posicaodosol.index.hour):
45     subset = posicaodosol.loc[posicaodosol.index.hour == hour, :]
46     r = subset.apparent_zenith
47     position = posicaodosol.loc[r.idxmin(), :]
48     axys.text(np.radians(position['azimuth']), position['apparent_zenith'], str(hour))

```

```

49
50 for dataestacao in pd.to_datetime(['2018-03-21', '2018-06-21', '2018-12-21']):
51     tempo = pd.date_range(dataestacao, dataestacao+pd.Timedelta('24h'), freq='5min', tz=tz)
52     posicaodosol = solarposition.get_solarposition(tempo,lat,lon)
53     posicaodosol = posicaodosol.loc[posicaodosol['apparent_elevation']>0, :]
54     label = dataestacao.strftime('%Y-%m-%d')
55     axys.plot(np.radians(posicaodosol.azimuth), posicaodosol.apparent_zenith, label=label)
56
57 #printa na tela o diagrama
58 axys.figure.legend(loc='upper left')
59 axys.set_theta_zero_location('N')
60 axys.set_theta_direction(-1)
61 axys.set_rmax(90)
62 #A partir de agora são os algoritmos para a modelagem dos dados para o objetivo do trabalho
63 #organizar as colunas de acordo com o necessario
64 columnsTitles
=
    ['year','day','min','glo_avg','dir_avg','diff_avg','lw_avg','par_avg','lux_avg','tp_sfc','humid','press','rain','ws_10m','wd_10m']
65
66 #arquivo com dados de 2008 a 2012
67 df0 = pd.read_csv('2008-2009-2010-2011-2012.csv', delimiter=',')#le o arquivo e separa as colunas por delimitadores
68 df1 = df0.drop(['id', 'datetm', 'Unnamed: 17'], axis=1) #remove as colunas id, datetm e unnamed
69 df2 = df1.reindex(columns=columnsTitles) #reorganiza as colunas
70
71 #arquivos com dados de 2013 a 2019
72 df3 = pd.read_csv('full-2013-2014-2015-2016-2017-2018-2019.csv', delimiter=',') #le o arquivo e separa as colunas por
delimitadores
73 df4 = df3.drop(['id'], axis=1) #remove as colunas id
74 df5 = df4.reindex(columns=columnsTitles) #reorganiza as colunas
75
76 #concatena as planilhas
77 df6 = pd.concat([df2, df5], ignore_index=True) #faz a concatenacao dos dados
78
79 #esse dado exclui os valores nan, ou seja, ausentes
80 ##para salvar o arquivo sem usar essa exclusao e so inverter as # de df6.to_csv para cs7.to_csv
81 df7 = df6.dropna()
82
83 print(df6) #esse print foi apenas para comparmos rapidamente com o df7
84 print(df7)
85
86 #salva o arquivo df6 ou df7
87 ##df6.to_csv('dados-de-2008-ate-2019-semdropna.csv')
88 df7.to_csv('dados-de-2008-ate-2019-comdropna.csv') #salva o arquivo na pasta aonde esta o codigo

```

O diagrama solar obtido para verificarmos seu comportamento:

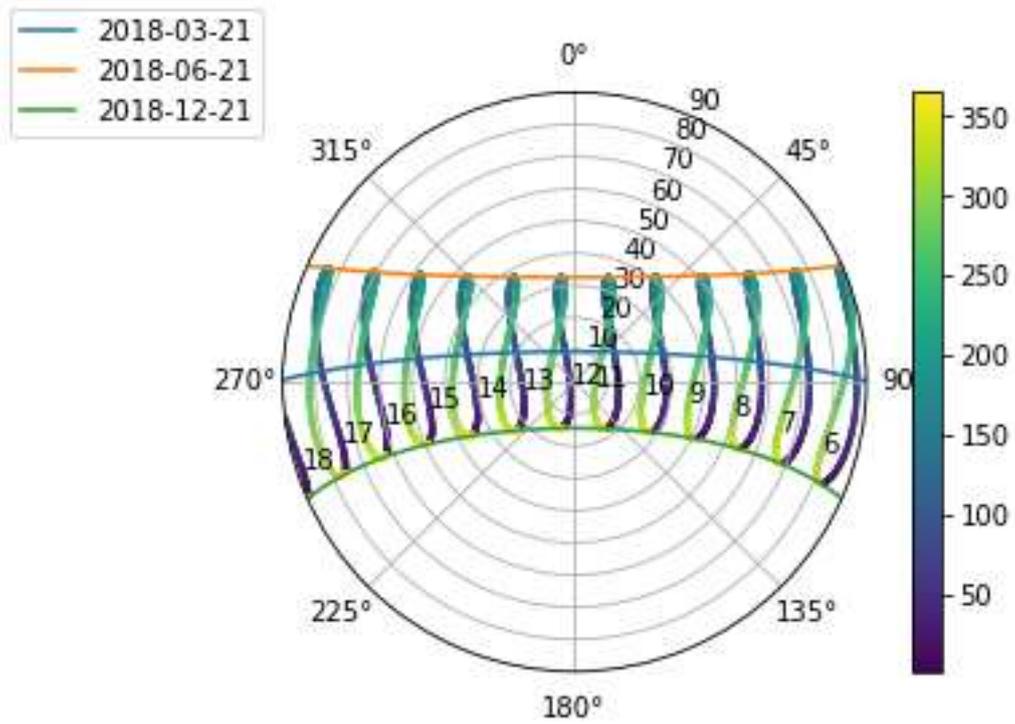


Figura 1. Diagrama Solar de Petrolina - PE. Fonte: Autoral.

Os dados obtidos no console do Spyder (presente no Anaconda) para a modelagem dos dados, mostrando o quando de linhas e colunas temos, isso nos dá uma ideia do tamanho e complexidade dos dados:

```

0      year  day  min  glo_avg  dir_avg  ...  humid  press  rain  ws_10m  wd_10m
1      2008   1    1    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
2      2008   1    2    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
3      2008   1    3    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
4      2008   1    4    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
...
5693755 2019  59  1435    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
5693756 2019  59  1436    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
5693757 2019  59  1437    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
5693758 2019  59  1438    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN
5693759 2019  59  1439    NaN      NaN    ...   NaN   NaN   NaN   NaN   NaN

[5693760 rows x 15 columns]
      year  day  min  glo_avg  dir_avg  ...  humid  press  rain  ws_10m  wd_10m
720     2008   1  720  431.100  458.8  ...  64.01  970.62  0.0  3.537  106.7
721     2008   1  721  680.000  868.0  ...  64.83  970.62  0.0  3.057  116.2
722     2008   1  722  665.800  857.0  ...  64.73  970.68  0.0  3.444  111.3
723     2008   1  723  669.600  865.0  ...  64.90  970.68  0.0  3.332  104.2
724     2008   1  724  683.700  869.0  ...  65.10  970.74  0.0  3.634  92.3
...
5687995 2019  55  1435  -1.066    0.0  ...  77.80  967.47  0.0  1.763  323.6
5687996 2019  55  1436  -1.069    0.0  ...  78.10  967.54  0.0  1.169  331.9
5687997 2019  55  1437  -1.063    0.0  ...  78.10  967.54  0.0  1.322  334.7
5687998 2019  55  1438  -1.069    0.0  ...  78.20  967.54  0.0  0.816  335.0
5687999 2019  55  1439  -1.066    0.0  ...  78.00  967.48  0.0  1.415  329.9

[5113224 rows x 15 columns]

```

Figura 2. Console exibindo os resultados da última modelagem parcial. Fonte: autoral.

E por fim a tabela abaixo, exhibe os valores obtidos na caixa “Variable Explorer”, também do Spyder, nesta caixa podemos visualizar informações gerais como nome, tipo, tamanho e valores que foram gerados pelas variáveis do código:

| Name | Type | Size | Value |
|---------------|-----------------------------------|---------------|--|
| axys | axes._subplots.PolarAxesSubplot | 1 | PolarAxesSubplot object of matplotlib.axes._subplots module |
| columnsTitles | list | 15 | ['year', 'day', 'min', 'glo_avg', 'dir_avg', 'diff_avg', 'lw_avg', 'pa ... |
| dataestacao | _libs.tslibs.timestamps.Timestamp | 1 | 2018-12-21 00:00:00 |
| datatm | datetime | 1 | 1900-12-21 00:00:00 |
| declination | float64 | 1 | -0.40875419804612767 |
| df0 | DataFrame | (2630880, 18) | Column names: id, year, day, datatm, min, glo_avg, dir_avg, diff_avg, ... |
| df1 | DataFrame | (2630880, 15) | Column names: year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, par_ ... |
| df2 | DataFrame | (2630880, 15) | Column names: year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, par_ ... |
| df3 | DataFrame | (3062880, 16) | Column names: id, year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, ... |
| df4 | DataFrame | (3062880, 15) | Column names: year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, par_ ... |
| df5 | DataFrame | (3062880, 15) | Column names: year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, par_ ... |
| df6 | DataFrame | (5693760, 15) | Column names: year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, par_ ... |
| df7 | DataFrame | (5113224, 15) | Column names: year, day, min, glo_avg, dir_avg, diff_avg, lw_avg, par_ ... |
| et | float64 | 1 | 2.155085618955181 |
| hour | int64 | 1 | 18 |
| hour_angle | Array of float64 | (8760,) | [5.2190714 20.2190714 35.2190714 ... -39.7809286 -24.7809286 -9. ... |
| label | str | 1 | 2018-12-21 |
| lat | float | 1 | -9.0689 |
| local | location.Location | 1 | Location object of pylib.location module |
| lon | float | 1 | -40.3197 |
| ndias | int | 1 | 355 |
| points | collections.PathCollection | 1 | PathCollection object of matplotlib.collections module |
| posicaoodosol | DataFrame | (151, 6) | Column names: apparent_zenith, zenith, apparent_elevation, elevation, ... |
| position | Series | (6,) | Series object of pandas.core.series module |
| r | Series | (67,) | Series object of pandas.core.series module |
| subset | DataFrame | (67, 6) | Column names: apparent_zenith, zenith, apparent_elevation, elevation, ... |
| tempo | DatetimeIndex | (289,) | DatetimeIndex: 289 entries, 2018-12-21 00:00:00-03:00 to 2018-12-22 00 ... |
| time | DatetimeIndex | (8760,) | DatetimeIndex: 8760 entries, 2018-01-01 12:00:00-03:00 to 2019-01-01 1 ... |
| tz | str | 1 | America/Recife |

Figura 3. Valores exibidos no variable explorer. Fonte: autoral.

Após esse processo de tratamento seria realizado a inclusão de mais algoritmos para fatiarmos os dados de 10 em 10 minutos e para dados de 30 em 30 minutos, para que fosse feita as implementações dos modelos de séries temporais dos modelos de treinamento. E por fim, seria feita a realização das estimativas e averiguar a eficiência dos resultados.

2. REFERÊNCIAS BIBLIOGRÁFICAS

Agência Nacional de Energia Elétrica (ANEEL). Brasil alcança 170 mil megawatts de capacidade instalada em 2019. 2020. Disponível em: https://www.aneel.gov.br/sala-de-imprensa-exibicao/-/asset_publisher/XGPXSqdMFHrE/content/brasil-alcanca-170-mil-megawatts-de-capacidade-instalada-em-2019/656877?inheritRedirect=false.

Acesso em: 09 oct. 2020.

Andrade, A. L.. A Viabilidade Ambiental de Usinas Hidrelétricas e a Aplicação da Avaliação Ambiental Estratégica no Brasil. Universidade Federal do Rio de Janeiro – UFRJ. 2017. Disponível em: http://www.ppe.ufrj.br/images/publica%C3%A7%C3%B5es/doutorado/Andr%C3%A9_de_Lima_Andrade.pdf. Acesso em: 20 oct. 2020.

Cleveland, C.; Morris, C.. Dictionary of Energy. 2nd Edition. Elsevier. 2014. Disponível em: <https://www.elsevier.com/books/dictionary-of-energy/cleveland/978-0-08-096811-7>. Acesso em: 09 oct. 2020.

Goldemberg, J.; Lucon, O.. Energia e meio ambiente no Brasil. Estud. av., São Paulo, v. 21, n. 59, p. 7-20, Apr. 2007. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-40142007000100003&lng=en&nrm=iso. Acesso em: 10 oct. 2020.

Empresa de Pesquisa Energética (EPE). Anuário Estatístico de Energia Elétrica de 2020 – Ano Base 2019. 2020. Disponível em: <https://www.epe.gov.br/sites-pt/publicacoes-dados-abertos/publicacoes/PublicacoesArquivos/publicacao-160/topico-168/Anu%C3%A1rio%20Estat%C3%ADstico%20de%20Energia%20EI%C3%A9trica%202020.pdf>. Acesso em 12 oct. 2020.

Martins, F. R.; Pereira, E. B.; Pinto, L. I. C.. O mercado brasileiro da energia eólica, impactos sociais e ambientais. Revista Ambiente & Água, 12(6), 1082-1100. 2017. Disponível em: https://www.scielo.br/scielo.php?script=sci_arttext&pid=S1980-993X2017000601082. Acesso em: 15 oct. 2020.

Martins, F. R.; Pereira, E. B.; Echer, M. P. de S.. Levantamento dos recursos de energia solar no Brasil com o emprego de satélite geoestacionário - o Projeto Swera. Revista Brasileira de Ensino de Física, v. 26, n. 2, p. 145 – 159, (2004). Disponível em: <http://www.sbfisica.org.br/rbef/pdf/031109.pdf>. Acesso em: 15 dez. 2020.

Gomez, J.M.R.; Carlesso, F.; Vieira, L. E.; Da Silva, L.. A irradiância solar: conceitos básicos. Rev. Bras. Ensino Fís., São Paulo, v. 40, n. 3, e3312, 2018. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1806-11172018000300412&lng=en&nrm=iso. Acesso em: 15 oct. 2020.

Simas, M.; Pacca, S.. Energia eólica, geração de empregos e desenvolvimento sustentável. Estudos Avançados 27 (77). 2013. Disponível em: <https://www.scielo.br/pdf/ea/v27n77/v27n77a08.pdf>. Acesso em: 16 oct. 2020.

Nakabayashi, R. Microgeração fotovoltaica no Brasil: condições atuais e perspectivas futuras. Universidade de São Paulo – USP. 2015. Disponível em: <http://www.abinee.org.br/informac/arquivos/mifoto.pdf>. Acesso em 20 oct. 2020.

VAREJÃO-SILVA, M. A. Meteorologia e Climatologia. Brasília: INMET, gráfica editora Pax, 2001.

Wang H., Zhenxing, L., Xian, Z.. A review of deep learning for renewable energy forecasting, v. 198, 15 October 2019. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0196890419307812>. Acesso em: 18 dez. 2020.

Mitchell, T. Machine Learning. 1998. Disponível parcialmente em: <http://www.cs.cmu.edu/~tom/mlbook.html>. Acesso em: 05 fev. 2020.