Prevendo a propagação de doenças

Leandro Galvão
Luís Felipe
Luiz Fernando Salvalágio
Marcio Colazingari

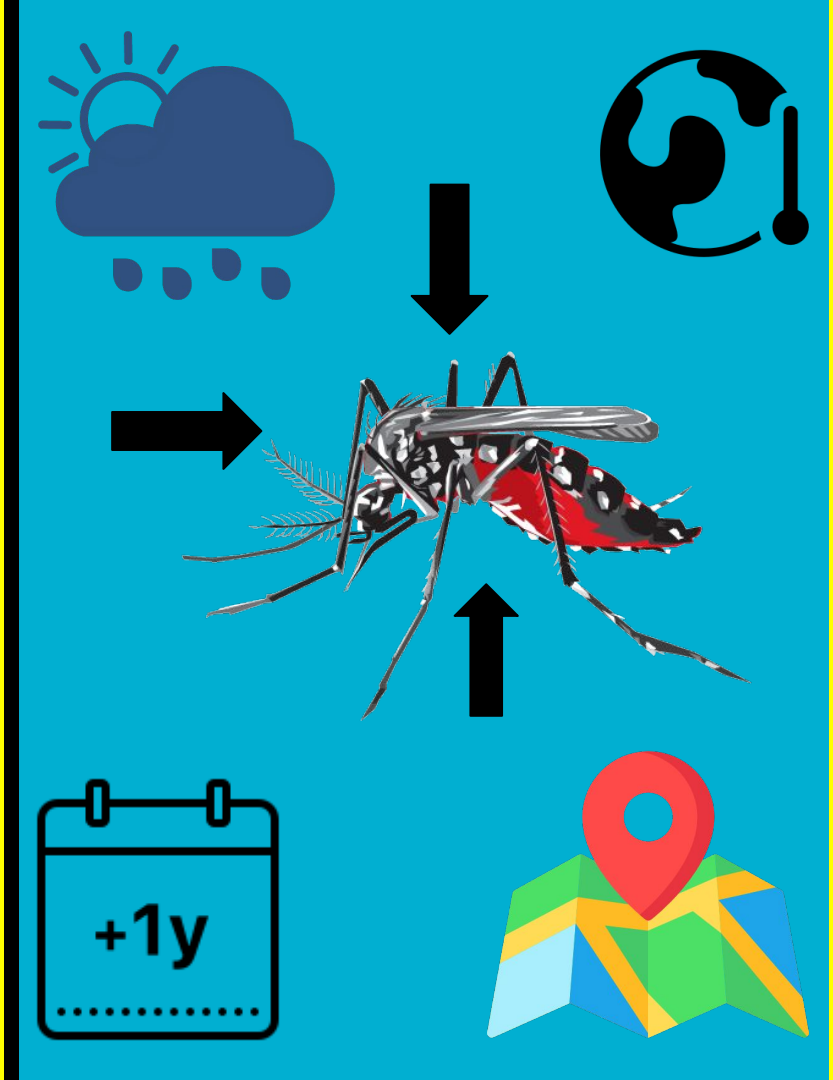
Previsão
de casos semanais de Dengue

?

DRIVENDATA

Download


CSV


+1y

	city	year	weekofyear	week_start_date	ndvi_ne	ndvi_nw	ndvi_se	ndvi_sw	precipitation_amt_mm	reanalysis_air_temp_k	...
0	sj	1990	18	1990-04-30	0.122600	0.103725	0.198483	0.177617	12.42	297.572857	...
1	sj	1990	19	1990-05-07	0.169900	0.142175	0.162357	0.155486	22.82	298.211429	...
2	sj	1990	20	1990-05-14	0.032250	0.172967	0.157200	0.170843	34.54	298.781429	...
3	sj	1990	21	1990-05-21	0.128633	0.245067	0.227557	0.235886	15.36	298.987143	...
4	sj	1990	22	1990-05-28	0.196200	0.262200	0.251200	0.247340	7.52	299.518571	...

reanalysis_relative_humidity_percent	reanalysis_sat_precip_amt_mm	reanalysis_specific_humidity_g_per_kg	reanalysis_tdtr_k	station_avg_temp_c
73.365714	12.42	14.012857	2.628571	25.442857
77.368571	22.82	15.372857	2.371429	26.714286
82.052857	34.54	16.848571	2.300000	26.714286
80.337143	15.36	16.672857	2.428571	27.471429
80.460000	7.52	17.210000	3.014286	28.942857
...	...	...	...	...

station_avg_temp_c	station_diur_temp_rng_c	station_max_temp_c	station_min_temp_c	station_precip_mm	total_cases
25.442857	6.900000	29.4	20.0	16.0	4
26.714286	6.371429	31.7	22.2	8.6	5

Sobre o Dataset

DIMENSÃO
1456 linhas por 25 colunas

24 features
1 label

Sobre o Dataset

{ LABEL : “total_cases” }

FEATURES :

“city”, “year”, “weekofyear”, “ndvi_*”, “preciptation_*”, “reanalysis_avg_temp_...., “station_avg_temp_c”, “station_precip_mm ”, entre outras ...

Sobre o DataSet

is null ?

Features

ndvi_ne -> 194 nulos

ndvi_nw -> 52

ndvi_se -> 22

ndvi_sw -> 22

station_avg_tem_c -> 43

dicionário – algumas colunas

ndvi - Índice de qualidade da vegetação

reanalisys_air_temp_k - Temperatura do Ar

reanalisys_dew_point_temp - Ponto de orvalho

tdr_k - Faixa da temperatura diurna

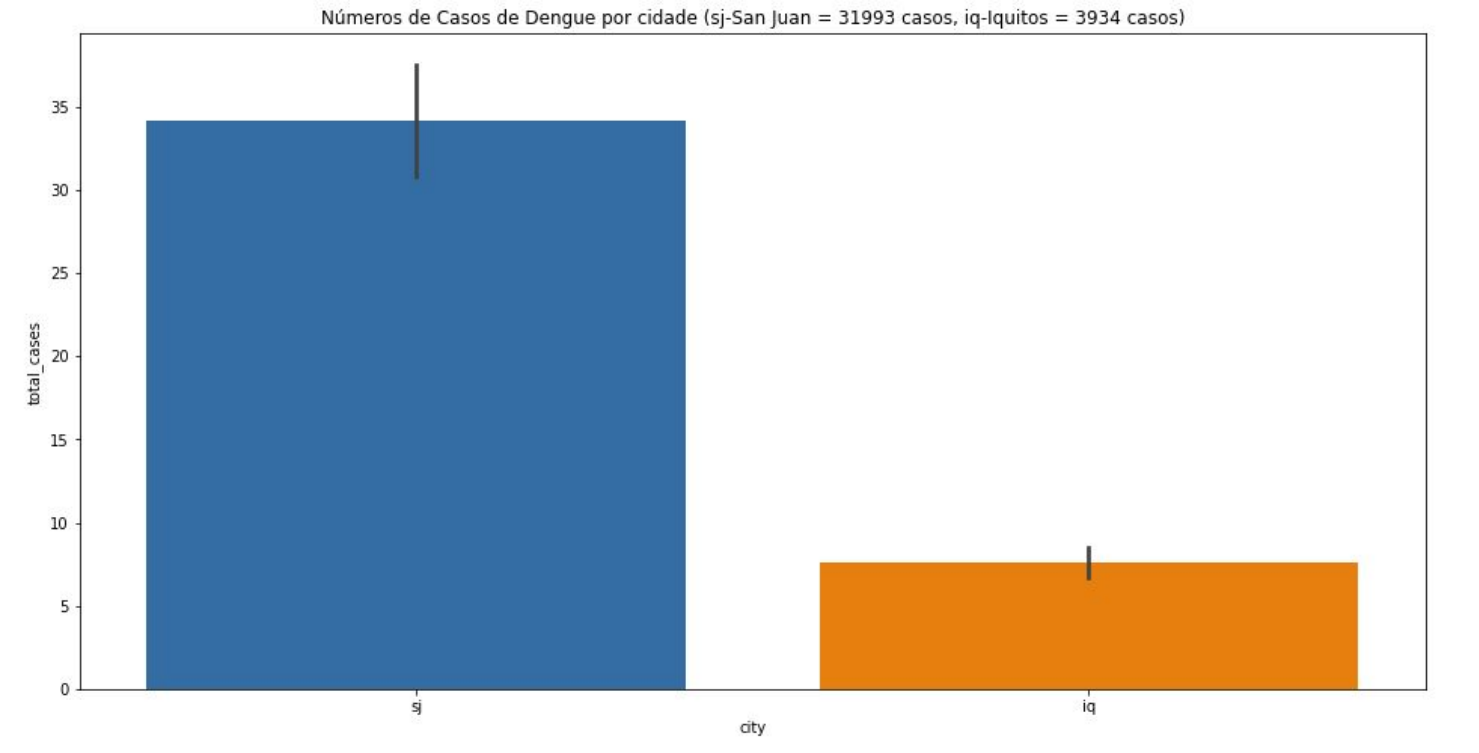
station_diur_temp_rng_c - Temperaturas -Estação climática

total_case - Casos de contaminação semanais (dengue)

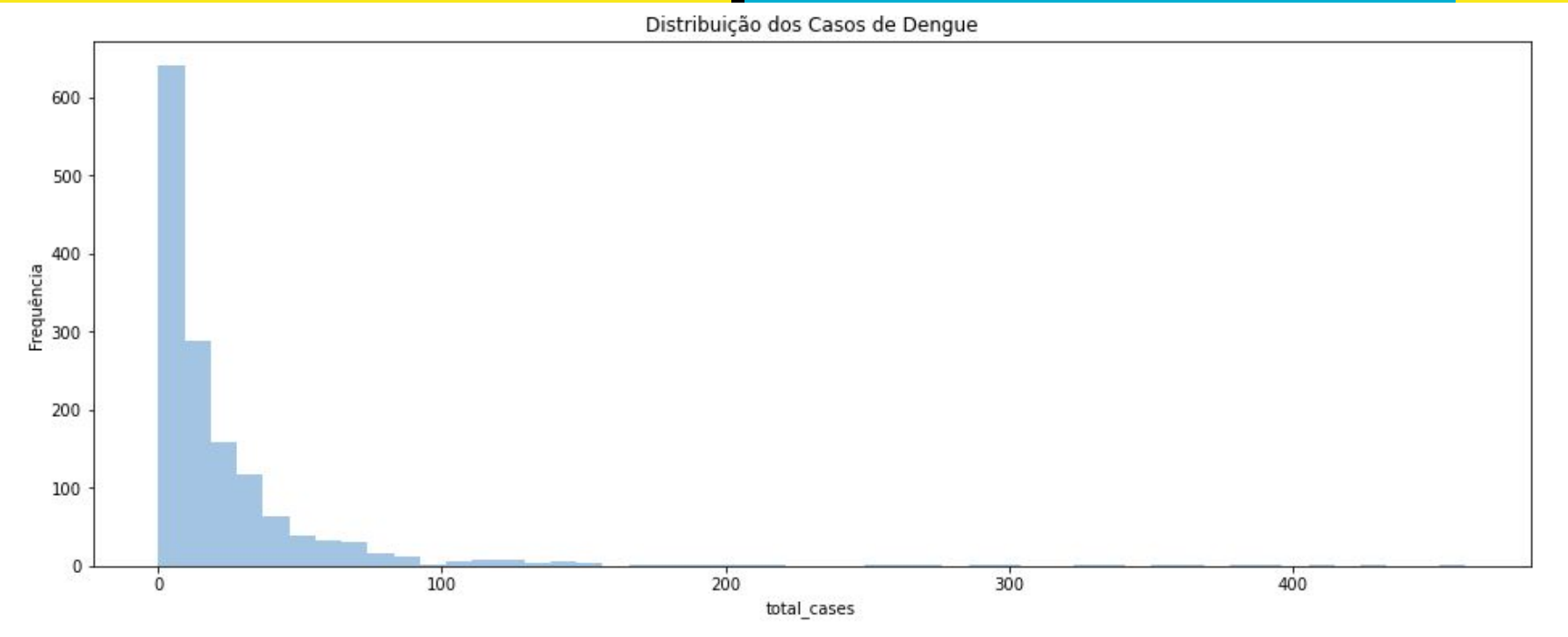
Describe dos dados

	ndvi_ne	precipitation_amt_mm	weekofyear	station_max_temp_c	total_cases
count	1262.000000	1443.000000	1456.000000	1436.000000	1456.000000
mean	0.142294	45.760388	26.503434	32.452437	24.675137
std	0.140531	43.715537	15.019437	1.959318	43.596000
min	-0.406250	0.000000	1.000000	26.700000	0.000000
25%	0.044950	9.800000	13.750000	31.100000	5.000000
50%	0.128817	38.340000	26.500000	32.800000	12.000000
75%	0.248483	70.235000	39.250000	33.900000	28.000000
max	0.508357	390.600000	53.000000	42.200000	461.000000

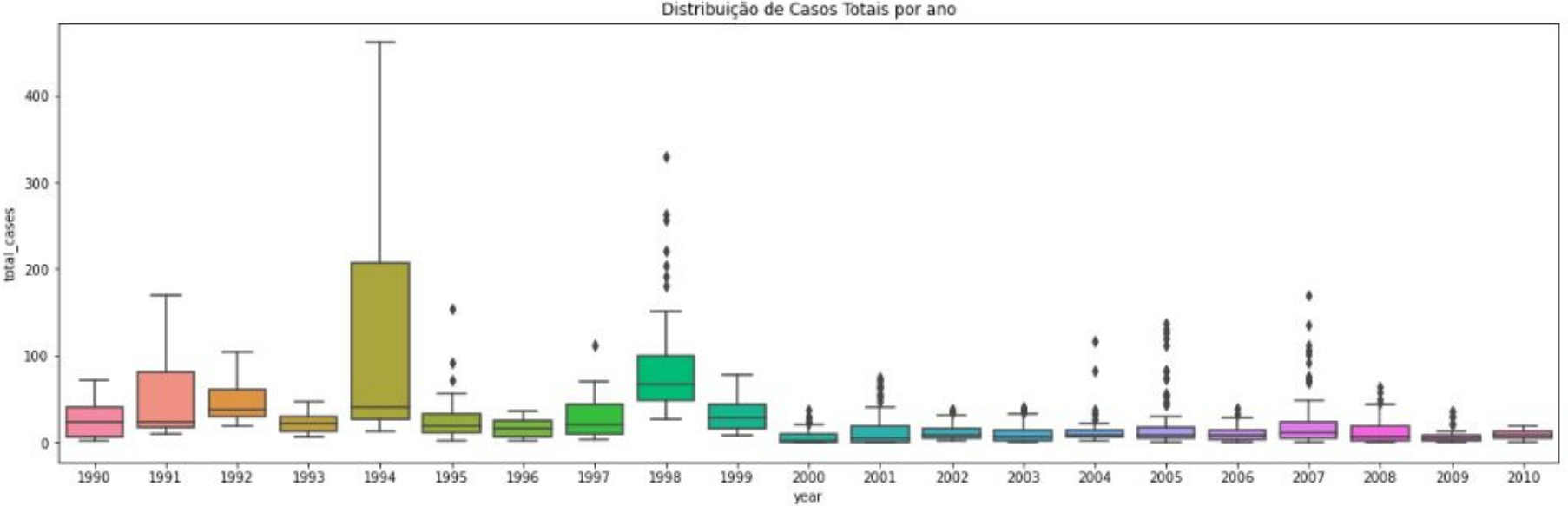
Contagem de casos por cidade



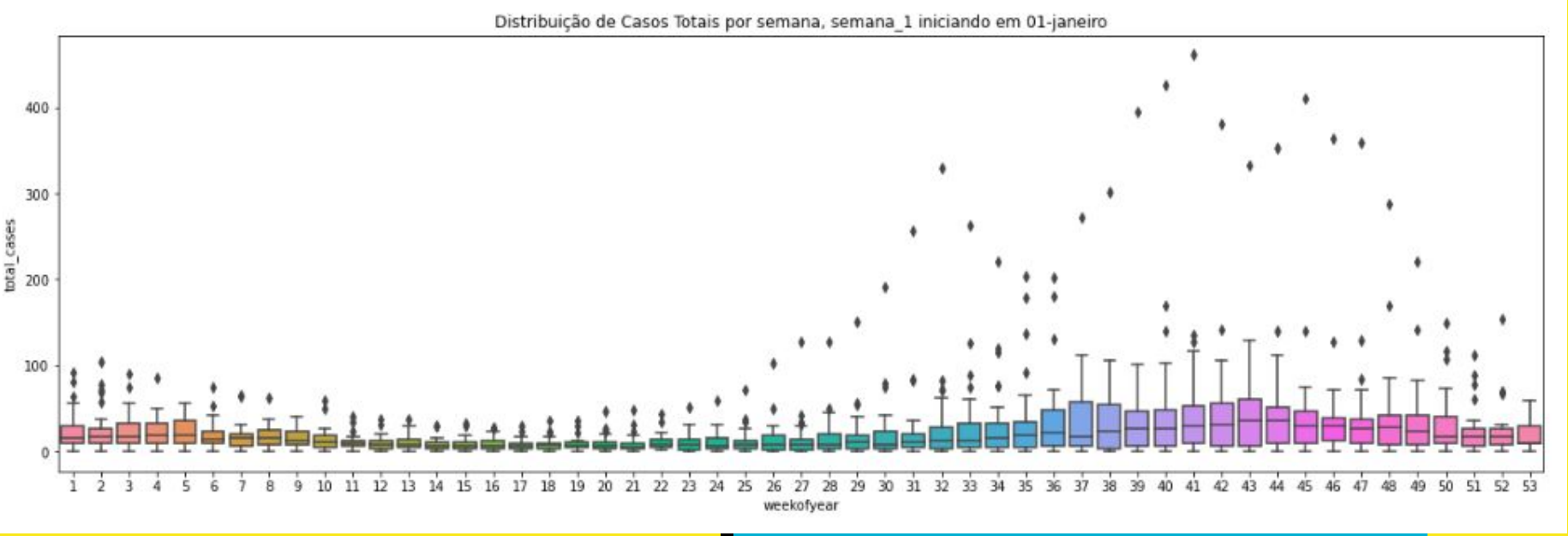
Casos Totais



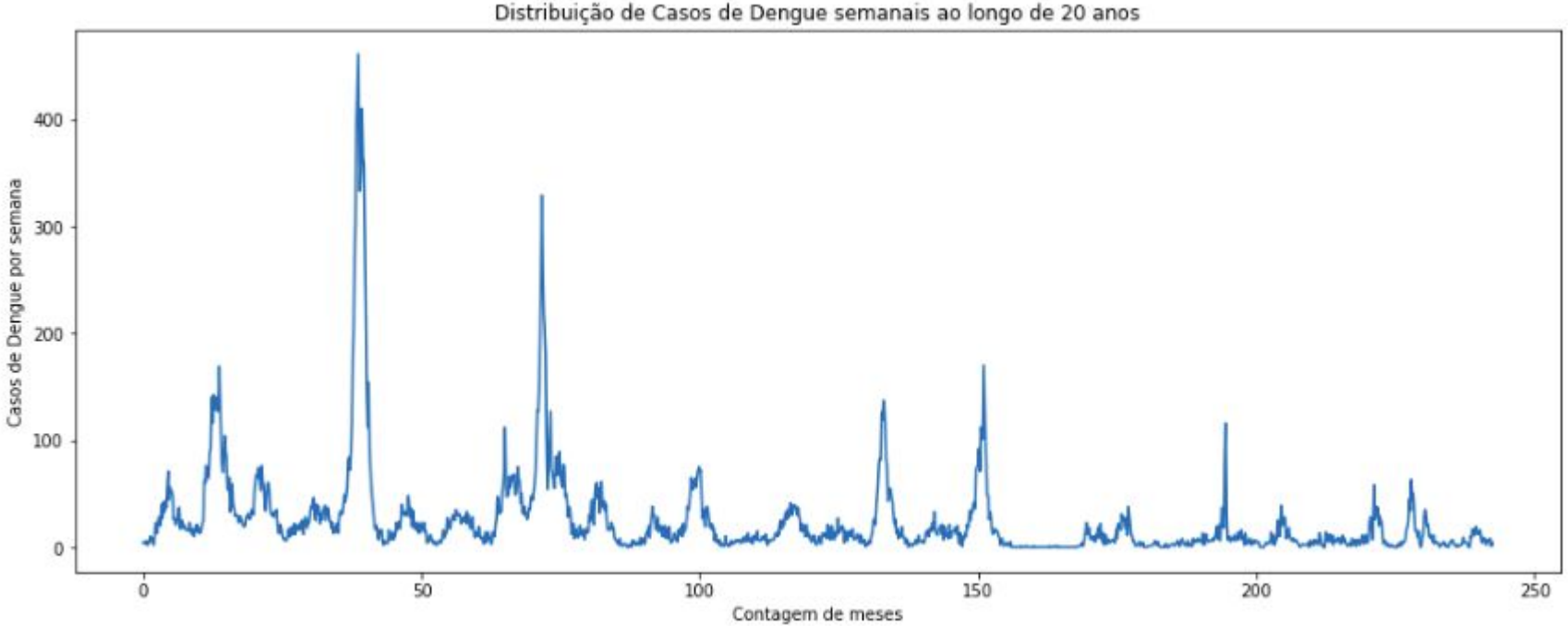
Casos de Dengue por ano



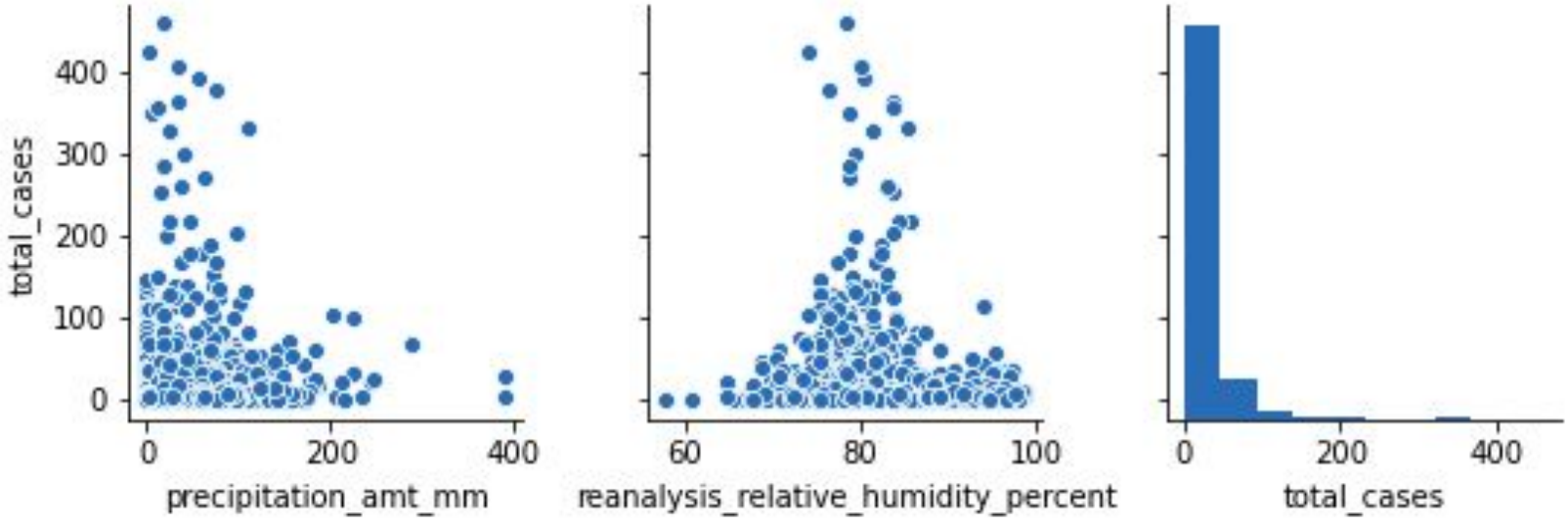
Casos semanais de Dengue



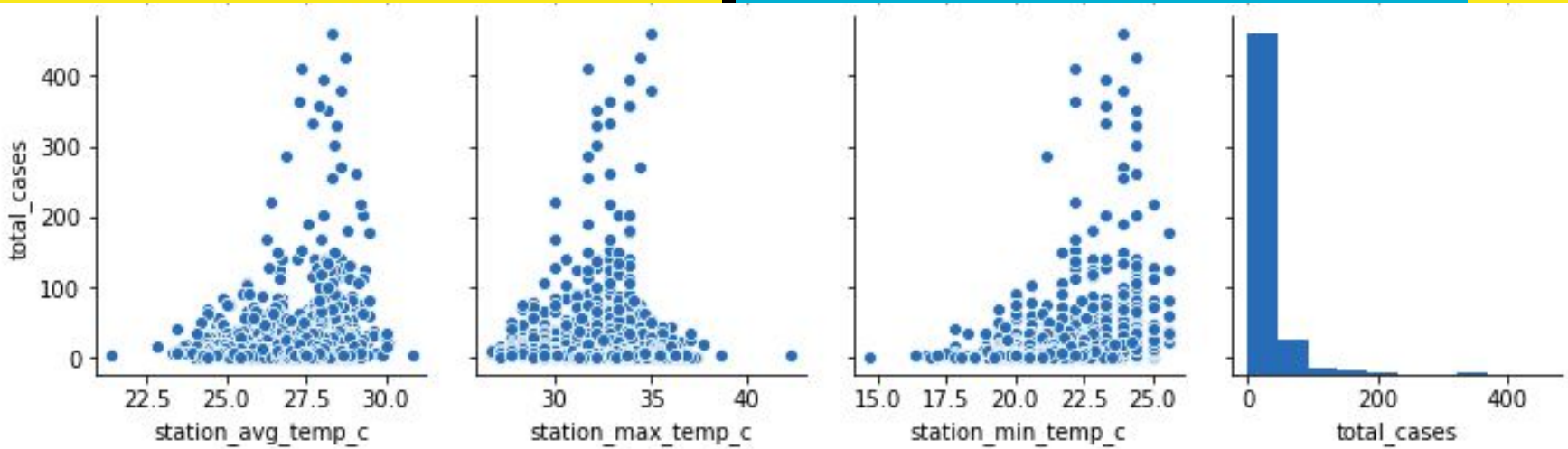
Registros ao longo de 120 meses

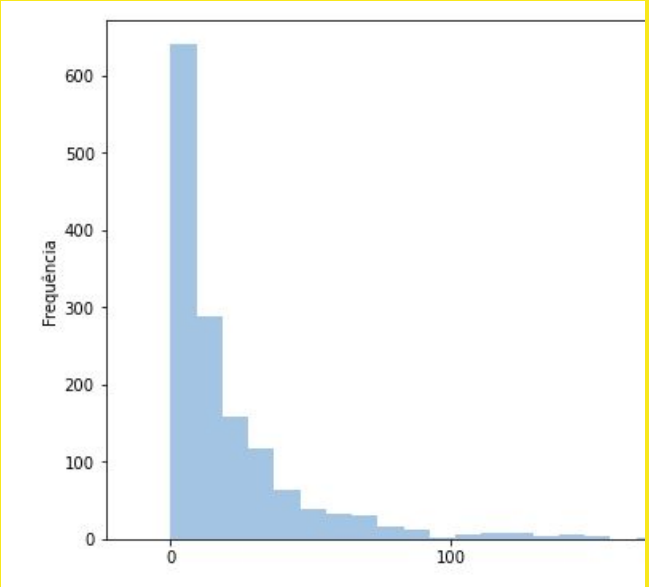


Pairplots - umidade e chuvas

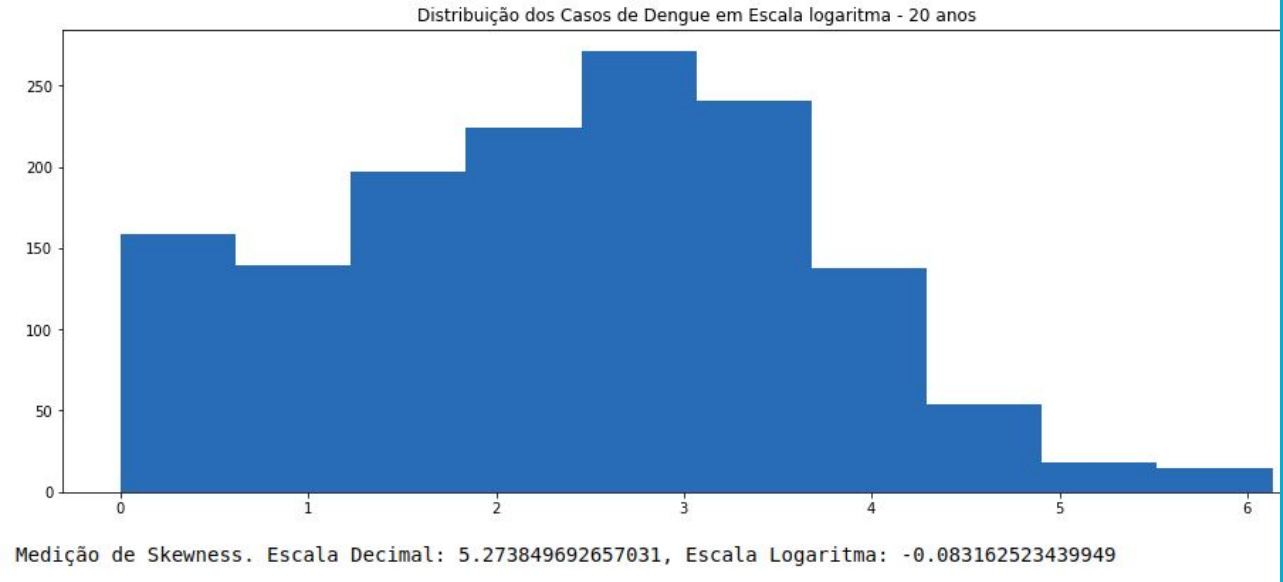


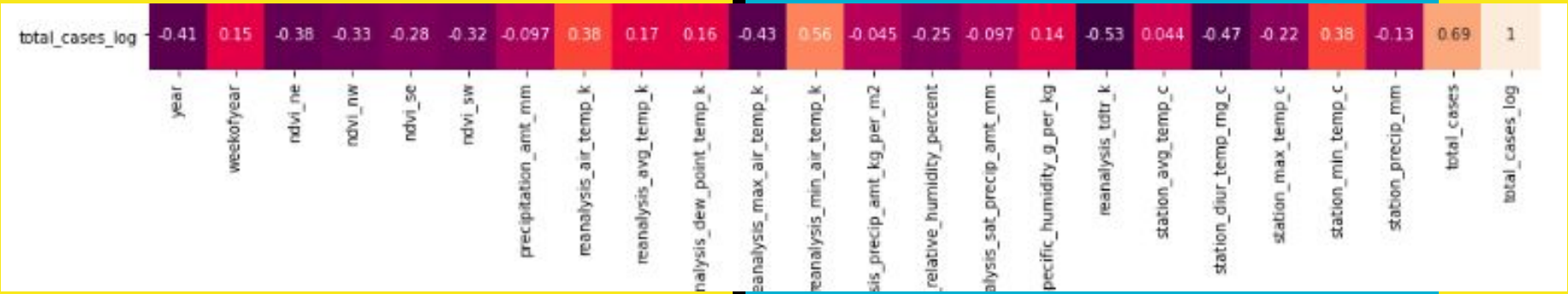
Pairplot - Temperatura





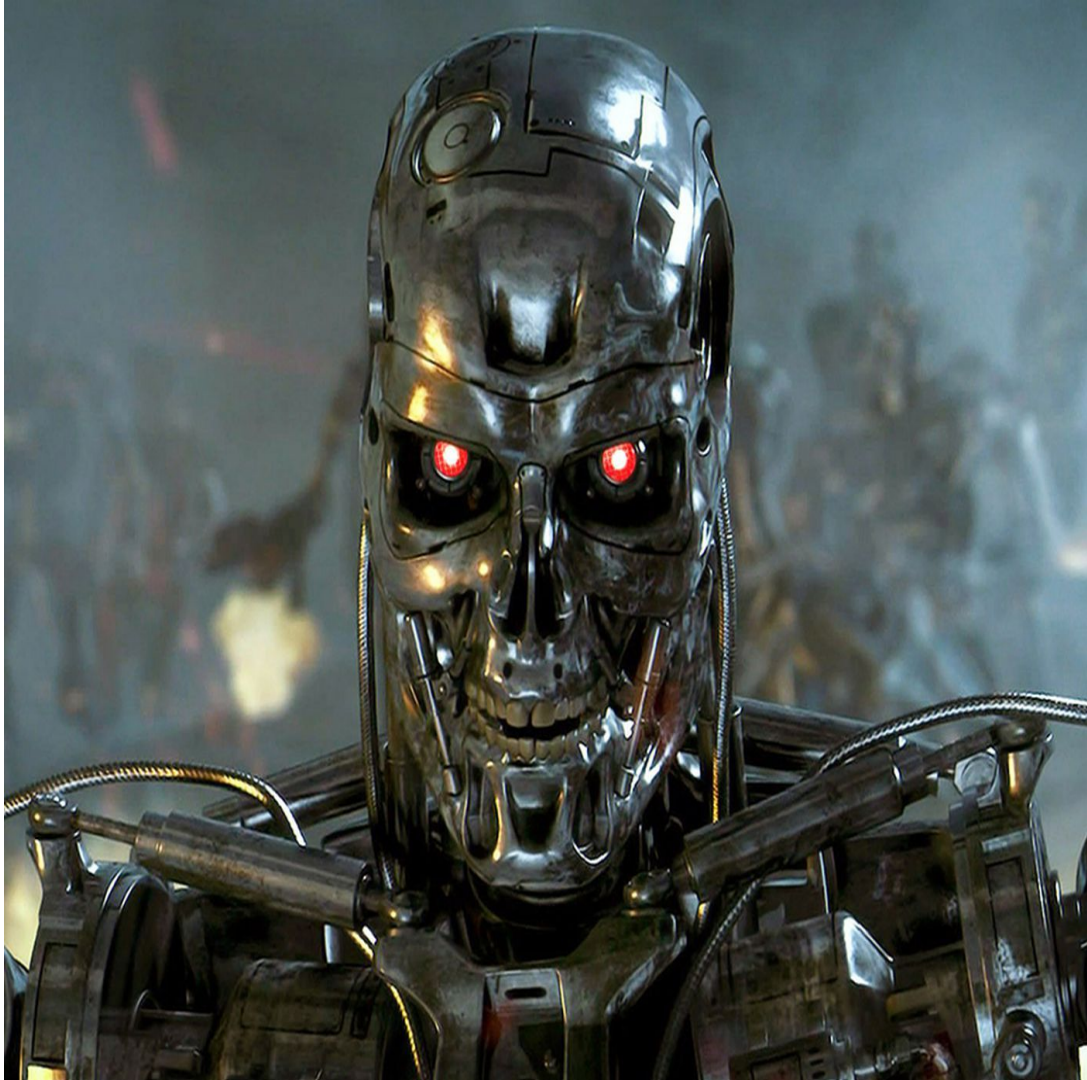
Simetria e correlação linear

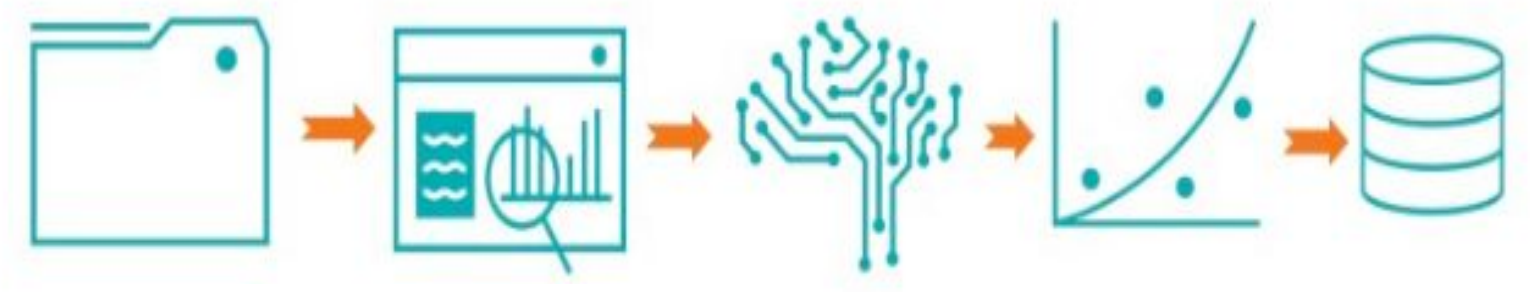

Distribuição dos Casos de Dengue em Escala logaritma - 20 anos
250
200
150
100
50
0
0
1
2
3
4
5
6
Medição de Skewness. Escala Decimal: 5.273849692657031, Escala Logaritma: -0.083162523439949


total_cases_log
-0.41
0.15
-0.38
-0.33
-0.28
-0.32
-0.097
0.38
0.17
0.16
-0.43
0.56
-0.045
-0.25
-0.097
0.14
-0.53
0.044
-0.47
-0.22
0.38
-0.13
0.69
1
year
weekofyear
ndvi_ne
ndvi_nw
ndvi_se
ndvi_sw
precipitation_amt_mm
reanalysis_air_temp_k
reanalysis_avg_temp_k
reanalysis_dew_point_temp_k
reanalysis_max_air_temp_k
reanalysis_min_air_temp_k
reanalysis_precip_amt_kg_per_m2
reanalysis_relative_humidity_percent
reanalysis_sat_precip_amt_mm
reanalysis_specific_humidity_g_per_kg
reanalysis_tdtr_k
station_avg_temp_c
station_diur_temp_rng_c
station_max_temp_c
station_min_temp_c
station_precip_mm
total_cases
total_cases_log

Predição

Machine Learning





Exploração / transformação / pré-processamento dos dados

Treino modelo

Métrica / CV
MAE
RMSE
R score

Avaliação Resultados

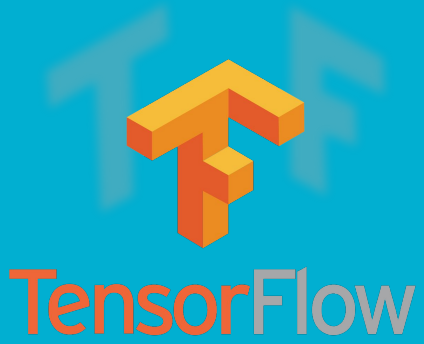
XGBoost

Regressão XGBoost

Regressão Linear

Regressão KNN

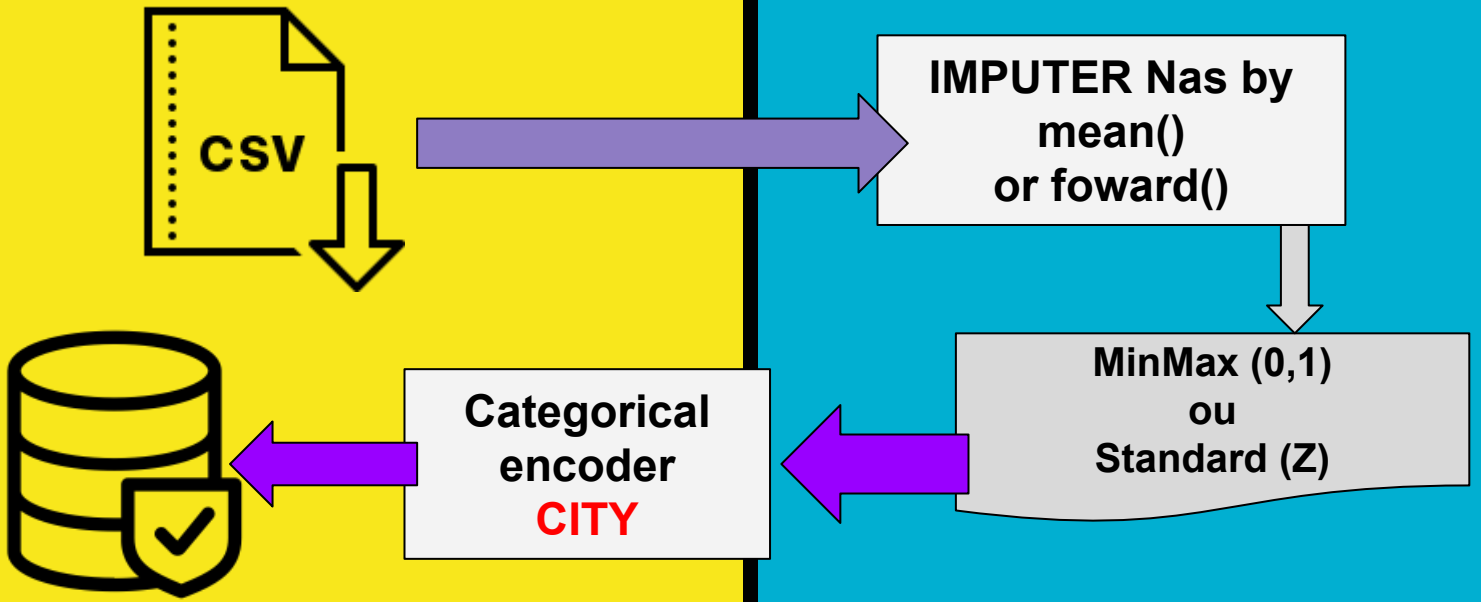
Random Forest



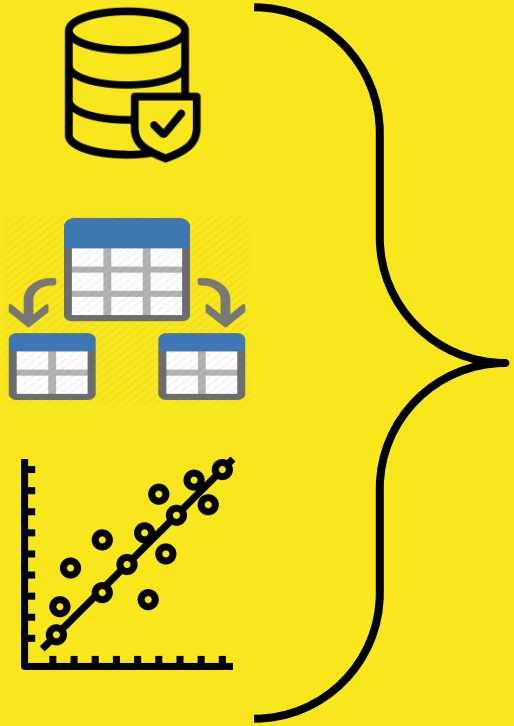
Long Short Term Memory

Pré - Processamento dos dados

sklearn.preprocessing



Regressão Linear - sklearn.LinearRegression



Score R2
0.18

MAE
20

R2
0.40

R2 com validação cruzada e pipeline com Transf. Poli.
0.134

Importância dos Features para a Regressão Linear

R2 Score
0.38

	column	coef	0
17	17.0	2.248113	reanalysis_specific_humidity_g_per_kg
11	11.0	1.804889	reanalysis_dew_point_temp_k
6	6.0	0.848518	ndvi_se
7	7.0	0.611106	ndvi_sw
0	0.0	0.244524	iq
1	1.0	0.244524	sj
19	19.0	0.204515	station_avg_temp_c
9	9.0	0.176241	reanalysis_air_temp_k
4	4.0	0.174318	ndvi_ne
10	10.0	0.119975	reanalysis_avg_temp_k
5	5.0	0.095855	ndvi_nw

Regressão com Random Forest sklearn.RandomForestRegressor

Score R2
0.617

Score R2 - CV(5)
0.435

MAE
12.52

Random Forest
Importância dos Features

2	0.221444	year
3	0.168759	weekofyear
7	0.068568	ndvi_sw
21	0.054900	station_max_temp_c
9	0.054344	reanalysis_air_temp_k
23	0.052915	station_precip_mm
5	0.051883	ndvi_nw
10	0.041701	reanalysis_avg_temp_k
14	0.036071	reanalysis_precip_amt_kg_per_m2
18	0.032097	reanalysis_tdtr_k
13	0.027862	reanalysis_min_air_temp_k
12	0.025179	reanalysis_max_air_temp_k
20	0.022924	station_diur_temp_rng_c
11	0.022350	reanalysis_dew_point_temp_k

XGBoost

Regressão com XGBoost

xgb.XGBRegressor()

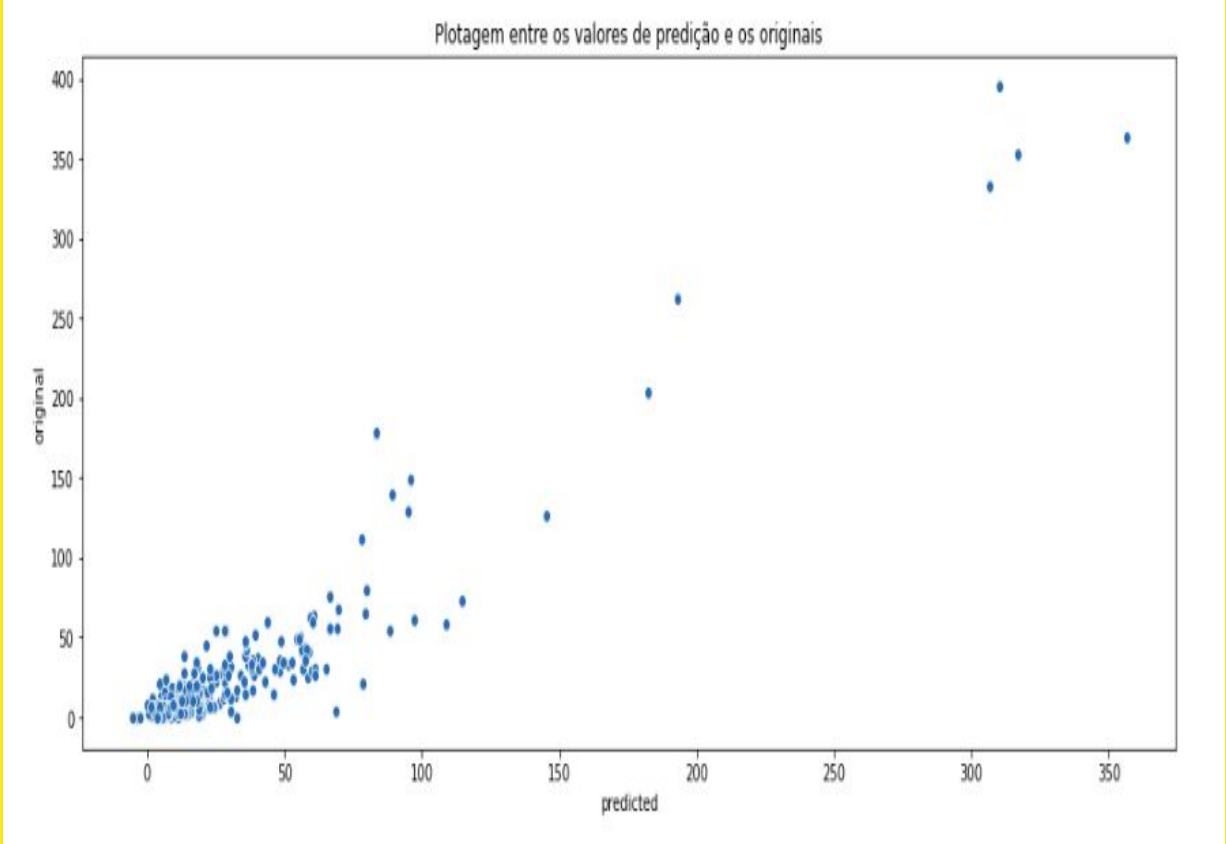
Hiperparâmetros

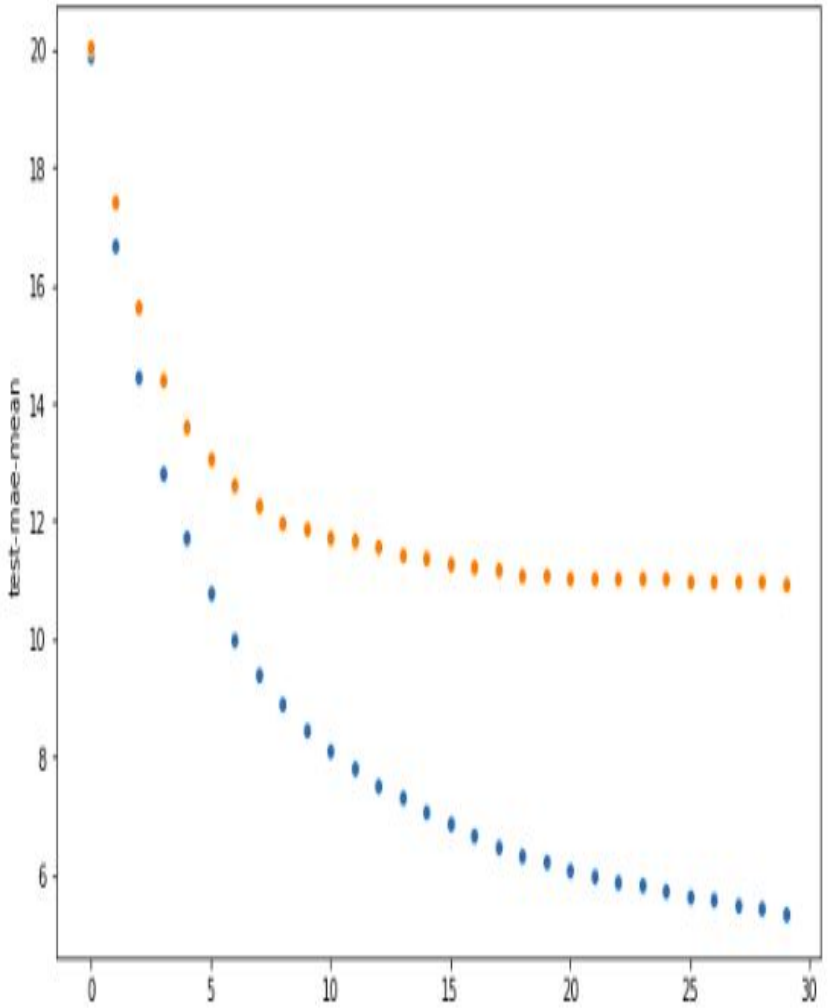
{ colsample_bytree : 0.7,
learning_rate : 0.2,
max_depth : 5,
alpha : 0, n_estimators : 100 }

xgb.XGBRegressor()

Score R2
0.892

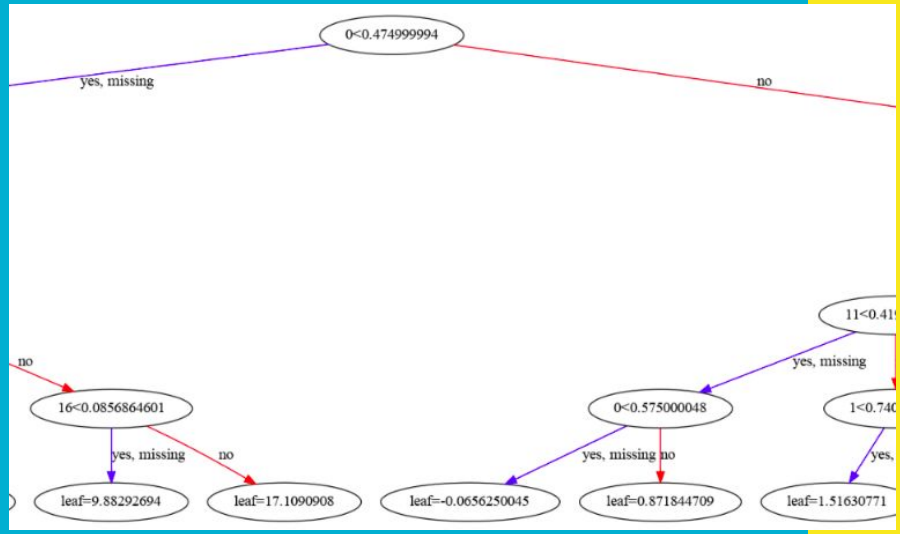
MAE
9.90

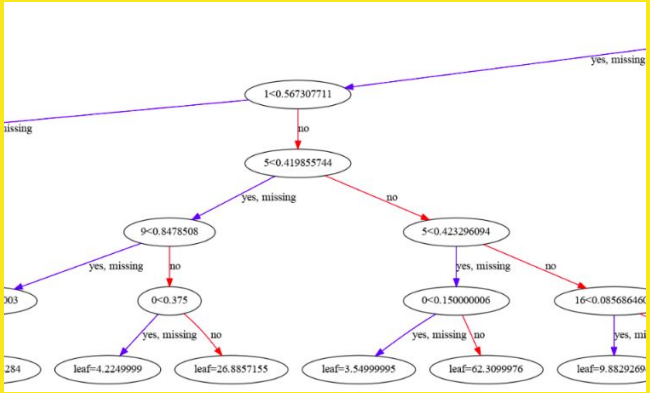


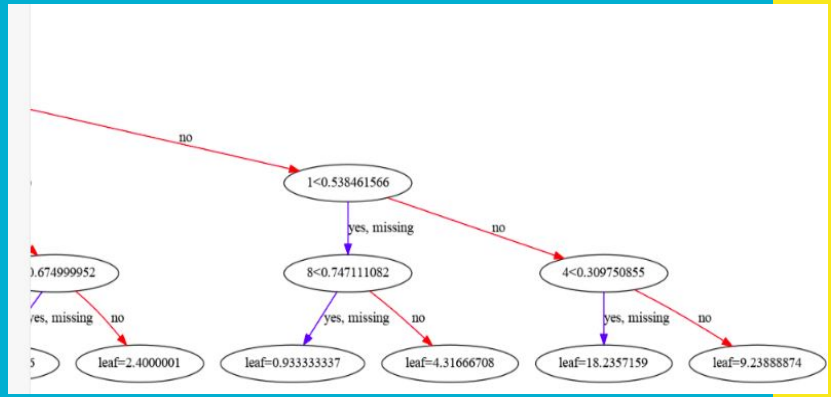


	train-mae-mean	train-mae-std	test-mae-mean	test-mae-std
0	19.557441	0.052126	20.009151	16.222599
1	16.675511	0.042655	17.431552	14.477345
2	14.424593	0.041575	15.627215	12.924915
3	12.515559	0.044414	14.404515	11.945699
4	11.741955	0.103404	13.590594	11.025725
5	10.791653	0.110724	13.035415	10.316955
6	9.955465	0.090935	12.590994	9.515501
7	9.359347	0.095361	12.256203	9.435901
8	8.557577	0.055750	11.954253	9.100195
9	8.452197	0.094323	11.852614	8.577091
10	8.126573	0.097556	11.719472	8.721734
11	7.816190	0.094237	11.664505	8.615342
12	7.542966	0.090115	11.574937	8.510753
13	7.299452	0.090502	11.447210	8.346764
14	7.071457	0.090325	11.362040	8.226231
15	6.863533	0.092317	11.250761	8.146127
16	6.673504	0.091969	11.233053	8.046307
17	6.500236	0.093049	11.154765	8.016410
18	6.344235	0.093441	11.100235	7.990535
19	6.209942	0.096207	11.060649	7.959155
20	6.096951	0.100662	11.043362	7.947526
21	6.000454	0.103737	11.035556	7.911144
22	5.907450	0.106561	11.025140	7.897612
23	5.821220	0.105111	11.025711	7.855045
24	5.736795	0.109951	11.013763	7.863717
25	5.654341	0.112905	11.006257	7.855669
26	5.573527	0.113959	10.996815	7.830360
27	5.497352	0.114030	10.951932	7.827403
28	5.422355	0.114972	10.965360	7.820633
29	5.344505	0.114651	10.954055	7.816565

XGBoost


0<0.474999994
yes, missing
no
11<0.41
no
yes, missing
16<0.0856864601
0<0.575000048
1<0.740
yes, missing
no
yes, missing
no
yes,
leaf=9.88292694
leaf=17.1090908
leaf=-0.0656250045
leaf=0.871844709
leaf=1.51630771

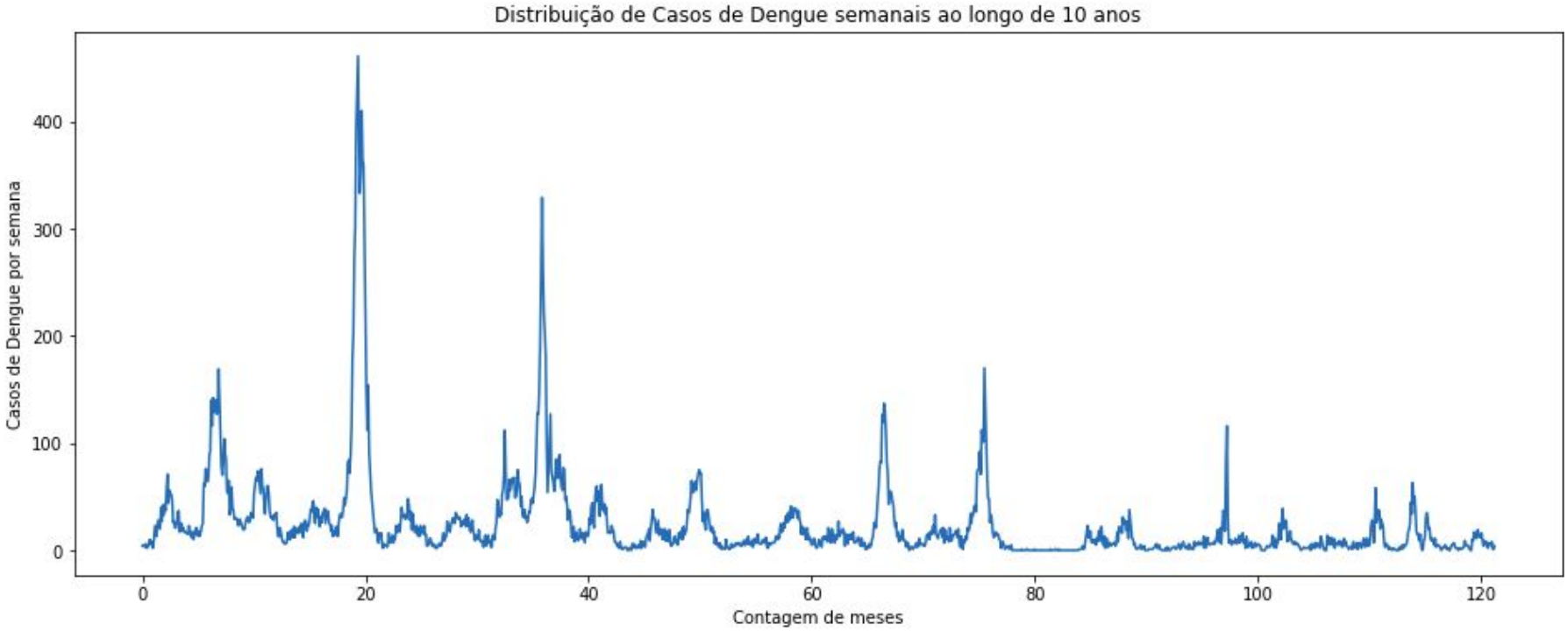

yes, missing
1<0.567307711
no
5<0.419855744
yes, missing
no
9<0.8478508
5<0.423296094
yes, missing
no
yes, missing
no
0<0.375
0<0.150000006
yes, missing
no
yes, missing
no
leaf=4.2249999
leaf=26.8857155
leaf=3.54999995
leaf=62.3099976

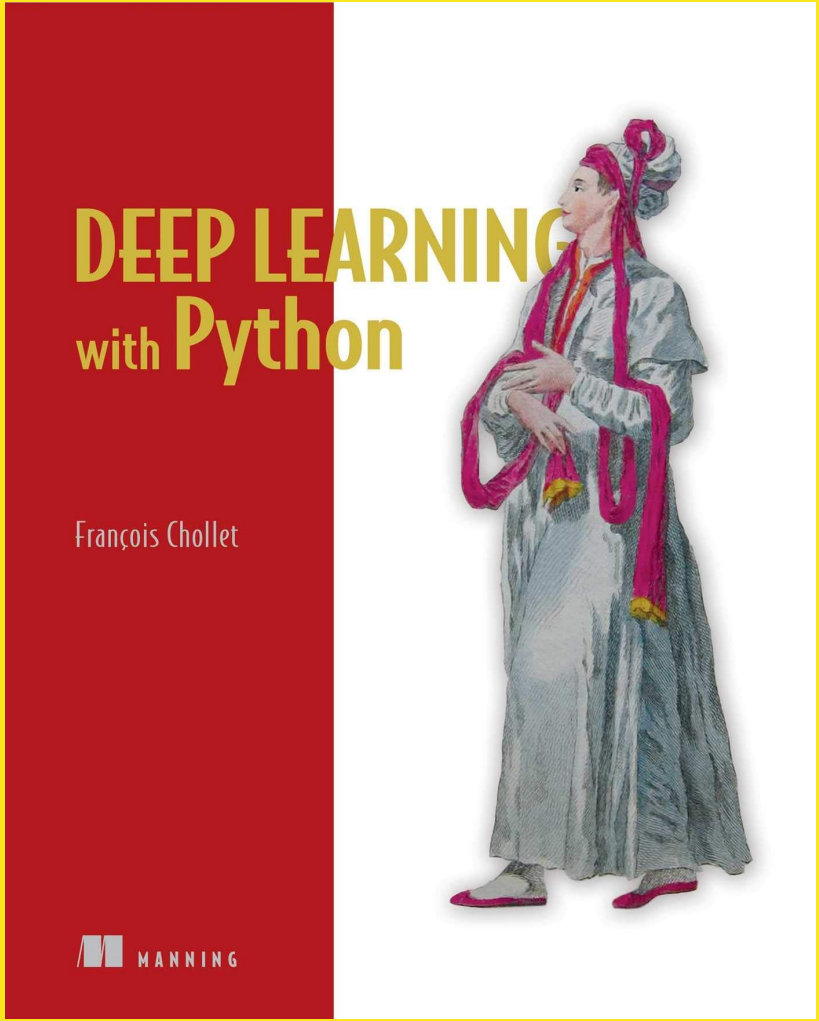

no
1<0.538461566
yes, missing
no
8<0.747111082
4<0.309750855
yes, missing
no
yes, missing
no
yes, missing
no
leaf=2.4000001
leaf=0.933333337
leaf=4.31666708
leaf=18.2357159
leaf=9.23888874

Long Short Term Memory

Dados Sequênciais

Keras





6.3.1 A temperature-forecasting problem

Until now, the only sequence data we've covered has been text data, such as the IMDB dataset and the Reuters dataset. But sequence data is found in many more problems than just language processing. In all the examples in this section, you'll play with a weather timeseries dataset recorded at the Weather Station at the Max Planck Institute for Biogeochemistry in Jena, Germany.[4]

In this dataset, 14 different quantities (such air temperature, atmospheric pressure, humidity, wind direction, and so on) were recorded every 10 minutes, over several years. The original data goes back to 2003, but this example is limited to data from 2009–2016. This dataset is perfect for learning to work with numerical timeseries. You'll use it to build a model that takes as input some data from the recent past (a few days' worth of data points) and predicts the air temperature 24 hours in the future.

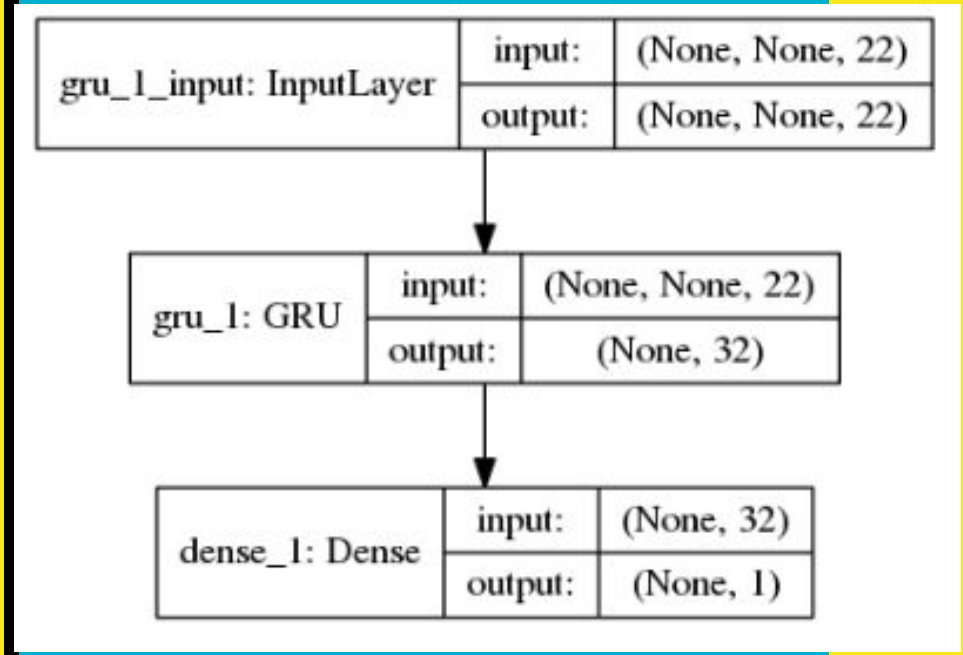
Download and uncompress the data as follows:

```
cd ~/Downloads
mkdir jena_climate
cd jena_climate
wget https://s3.amazonaws.com/keras-datasets/jena_climate_2009_2016.csv.zip
unzip jena_climate_2009_2016.csv.zip
```

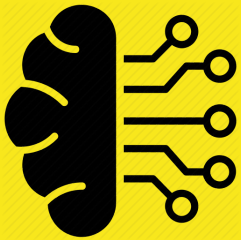
Let's look at the data.

Modelo proposto e adotado

keras. plot_model



Keras

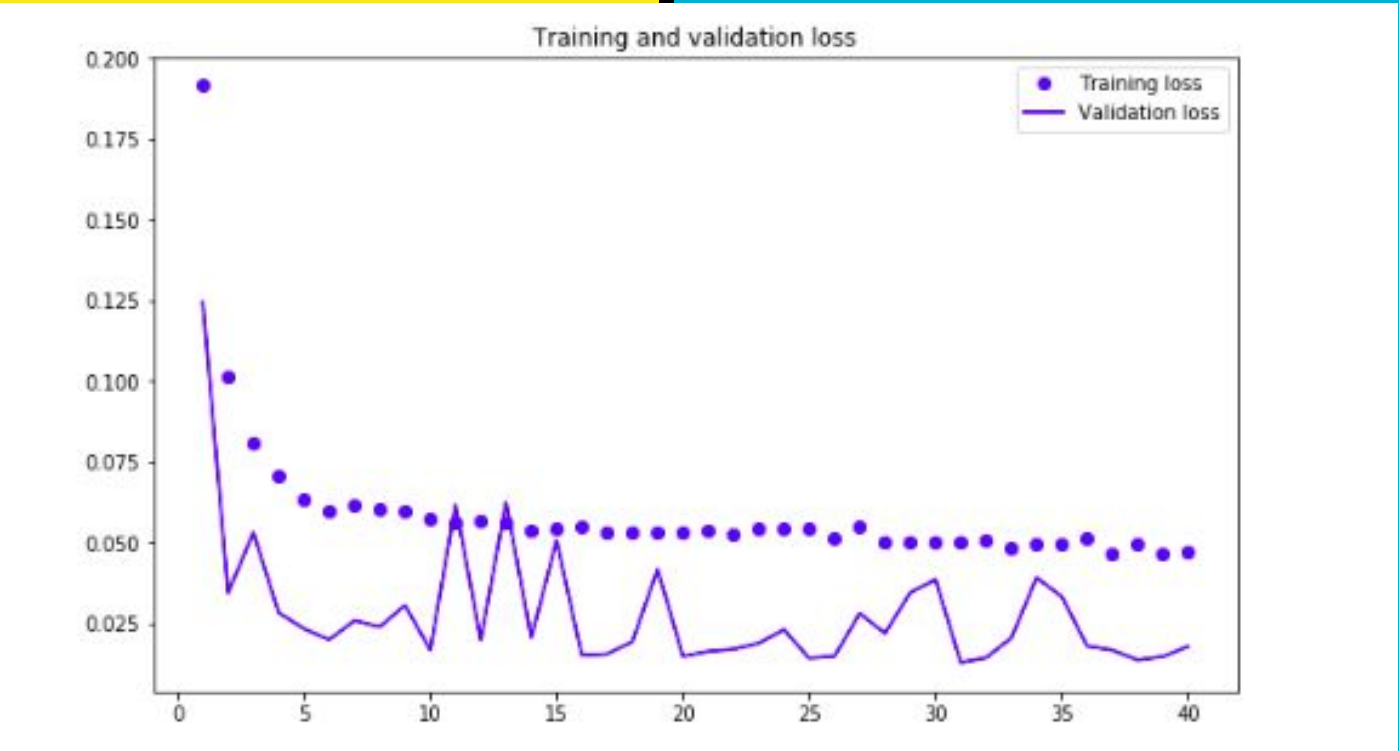


LookBack : 53

Step : 1

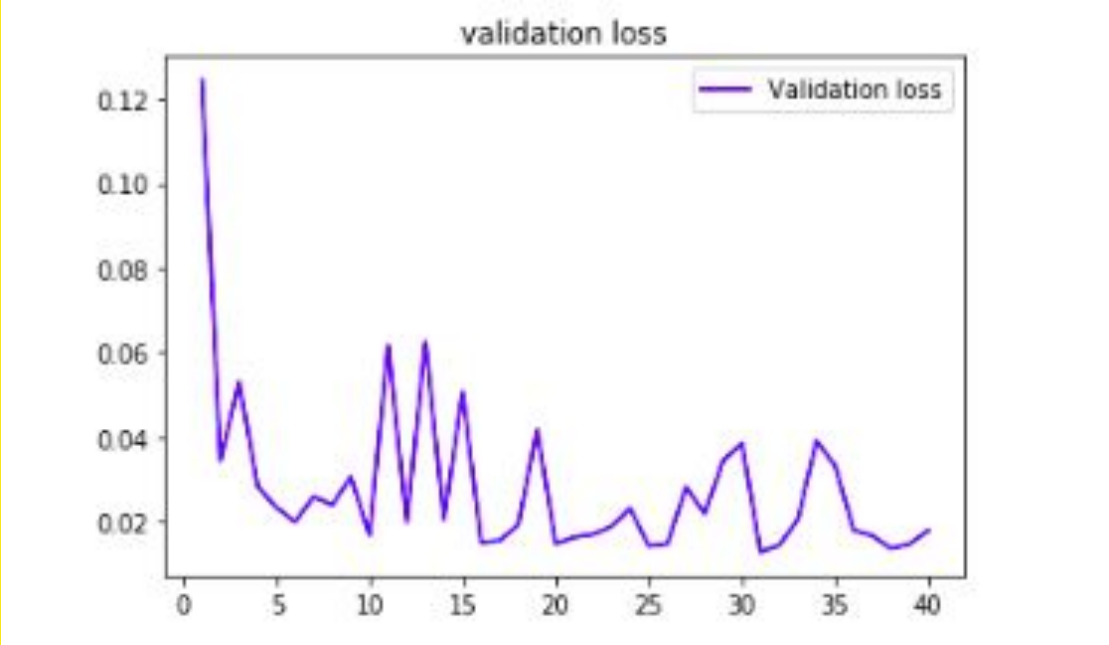
delay : 16

Keras


Training and validation loss
Training loss
Validation loss
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0
5
10
15
20
25
30
35
40

Keras

Long Short Term Memory



```
Epoch 1/40
50/50 [==============================] - 4s 89ms/step - loss: 0.1913 - val_loss: 0.1245
Epoch 2/40
50/50 [==============================] - 4s 79ms/step - loss: 0.1012 - val_loss: 0.0345
Epoch 3/40
50/50 [==============================] - 4s 84ms/step - loss: 0.0807 - val_loss: 0.0531
Epoch 4/40
50/50 [==============================] - 4s 89ms/step - loss: 0.0704 - val_loss: 0.0283
Epoch 5/40
50/50 [==============================] - 4s 87ms/step - loss: 0.0636 - val_loss: 0.0234
Epoch 6/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0599 - val_loss: 0.0200
Epoch 7/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0619 - val_loss: 0.0259
Epoch 8/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0606 - val_loss: 0.0240
Epoch 9/40
50/50 [==============================] - 4s 79ms/step - loss: 0.0597 - val_loss: 0.0306
Epoch 10/40
50/50 [==============================] - 4s 79ms/step - loss: 0.0573 - val_loss: 0.0168
Epoch 11/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0561 - val_loss: 0.0617
Epoch 12/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0570 - val_loss: 0.0199
Epoch 13/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0562 - val_loss: 0.0624
Epoch 14/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0540 - val_loss: 0.0206
Epoch 15/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0542 - val_loss: 0.0506
Epoch 16/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0553 - val_loss: 0.0152
Epoch 17/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0534 - val_loss: 0.0156
Epoch 18/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0533 - val_loss: 0.0194
Epoch 19/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0531 - val_loss: 0.0417
Epoch 21/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0538 - val_loss: 0.0164
Epoch 22/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0524 - val_loss: 0.0170
Epoch 23/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0545 - val_loss: 0.0189
Epoch 24/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0544 - val_loss: 0.0231
Epoch 25/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0544 - val_loss: 0.0143
Epoch 26/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0512 - val_loss: 0.0148
Epoch 27/40
50/50 [==============================] - 4s 80ms/step - loss: 0.0552 - val_loss: 0.0281
Epoch 28/40
50/50 [==============================] - 4s 84ms/step - loss: 0.0503 - val_loss: 0.0220
Epoch 29/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0503 - val_loss: 0.0345
Epoch 30/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0502 - val_loss: 0.0386
Epoch 31/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0504 - val_loss: 0.0129
Epoch 32/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0506 - val_loss: 0.0144
Epoch 33/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0487 - val_loss: 0.0206
Epoch 34/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0498 - val_loss: 0.0392
Epoch 35/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0496 - val_loss: 0.0333
Epoch 36/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0513 - val_loss: 0.0180
Epoch 37/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0467 - val_loss: 0.0168
Epoch 38/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0495 - val_loss: 0.0137
Epoch 39/40
50/50 [==============================] - 4s 77ms/step - loss: 0.0464 - val_loss: 0.0148
Epoch 40/40
50/50 [==============================] - 4s 78ms/step - loss: 0.0473 - val_loss: 0.0179
```

Keras

Long Short Term Memory

(mean)
MAE para 40 epochs
1, 216

Obrigado Chollet

QUADRO RESUMO

Linear : [R2(0.40), MAE(20)]

Random Forest : [R2(0.67), MAE(12.52)]

XGBoost : [R2(0.89), MAE(9.90)]

LSTM : [MAE(1.22)]

Acesse o caderno dos experimentos

This Way

Outro caminho...

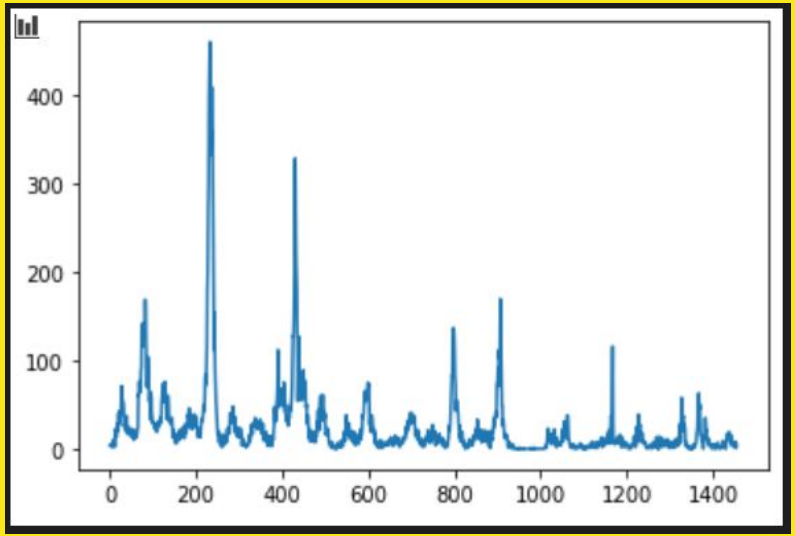
Primeiro tratamento de dados:

- Transformar coluna da cidade em binário
  código: pandas.get_dummies

- Valores nulos: empiricamente provou-se melhor a imputação desses valores
  código: interpolate(method='index', limit_direction='forward')

Teste de regressão com k-Nearest Neighbors (KNN) e Polinômios

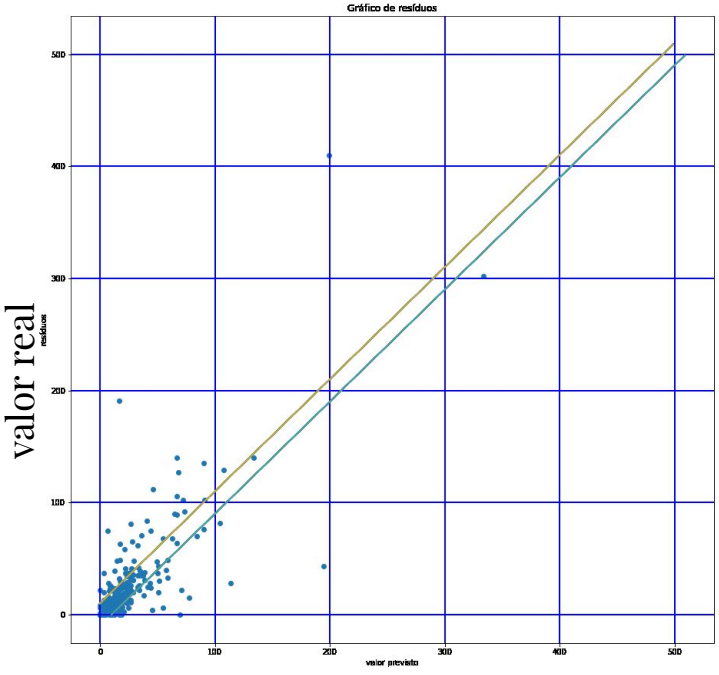
Gráfico dos casos de dengue registrados no dataset



Motivos

- sinuosidade do dado investigado.
- possível relação com variáveis mais próximas
- tentativa de limitar o modelo a apenas algumas variáveis mais relevantes

Resultados

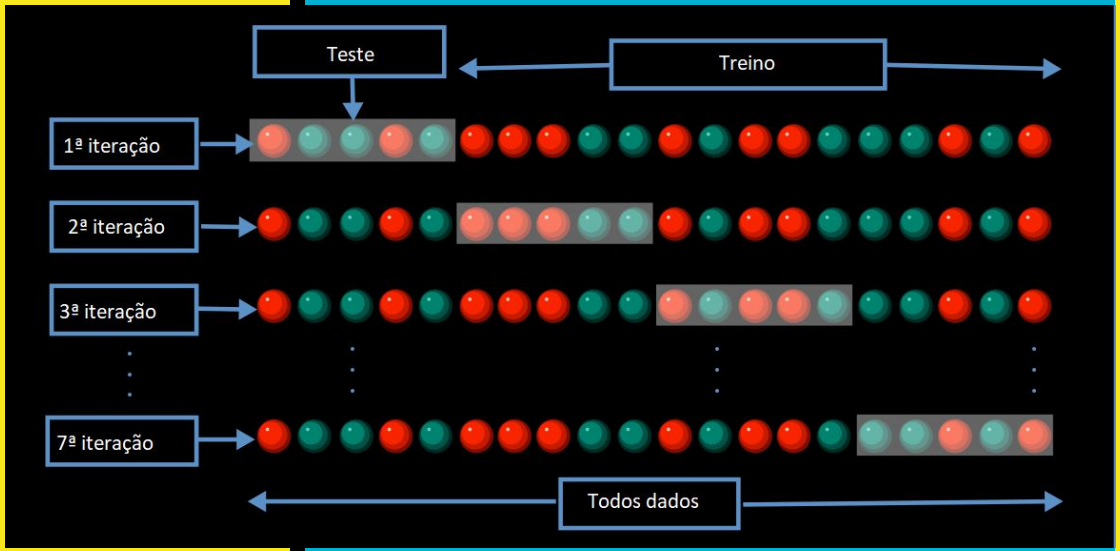


Dados de 7 mil testes
colunas com valores do resultado de testes e duas medidas de modelos estatísticos.
mean_absolute_error é parâmetro de avaliação da competição oficial do dataset

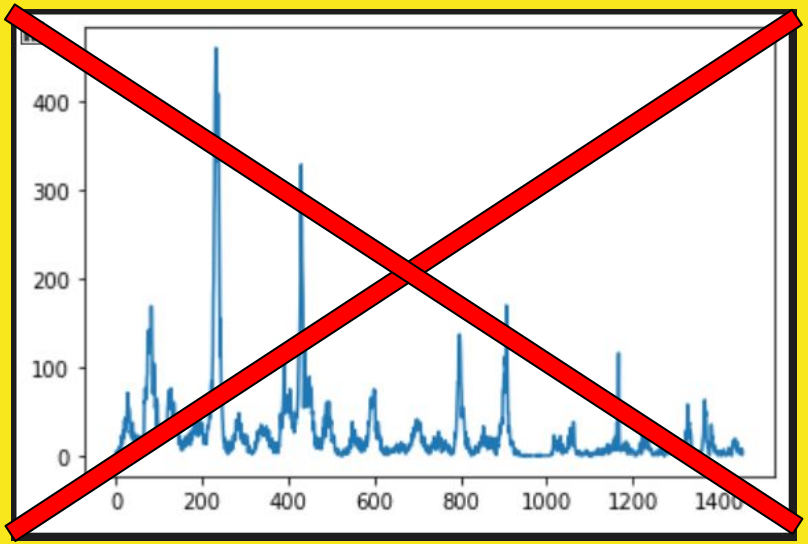
	teste	mean_absolute_error	r2_score
count	7000.000000	7000.000000	7000.000000
mean	0.894963	6.719093	0.894740
std	0.057984	0.887343	0.057995
min	0.783963	5.471154	0.783774
25%	0.837557	5.475962	0.837611
50%	0.909415	6.750000	0.909361
75%	0.949450	7.379808	0.949454
max	0.958374	8.033654	0.958490

Validação cruzada

1. Dataset embaralhado
   código: data.sample(frac=1).reset_index(drop=True)
2. Divisão em 7 partes iguais (6 para treino e 1 para teste)
3. Loops para treinar e testar todo dataset



tratamentos qualificados de dados



Observou-se que a variável mais relevante do primeiro modelo foi a cidade;
Cada cidade possui peculiaridades nas condições ambientais favoráveis aos casos de dengue;
A continuidade dos dados postos, mesmo com o valor binário das cidades pode confundir o modelo em suas multidimensões ou limitá-lo.
é necessário selecionar as variáveis mais relevantes e essas podem não coincidir nas cidades avaliadas.

tratamentos qualificados de dados

A dengue é doença endêmica nessas cidades. os surtos de casos, por sua vez, têm característica de epidemia.

Gobierno y política
16 / oct / 2020

Más de 3,660 casos sospechosos de dengue en Puerto Rico

Los casos confirmados este año (461) son cuatro veces más que todos los registrados en el 2019 y la población pediátrica es la más afectada con el virus.

Puerto Rico
9 / abr / 2020

Epidemiólogo del Estado confirma brote de dengue

David Capó dijo que se han superado los 1,000 casos de dengue.

Gobierno y política
19 / mar / 2020

El dengue amenaza a Puerto Rico

Gobernadora llama a aprovechar el distanciamiento social

Distrito de Belén tiene menos casos de dengue en Iquitos metropolitano

27/10/2020 // No hay comentarios

Gracias al trabajo articulado y permanente de la Municipalidad de Belén y la DIRESA En Belén 7%, en San Juan 29.5%, en Iquitos 27.3% y

Leer más »

Dengue continúa azotando a la ciudad de Iquitos

23/07/2020 // No hay comentarios

Niños son los más afectados El Dr. Bernardo Laulate, responsable del área de Pediatría del Hospital Iquitos, manifestó que el número de niños enfermos con

Leer más »

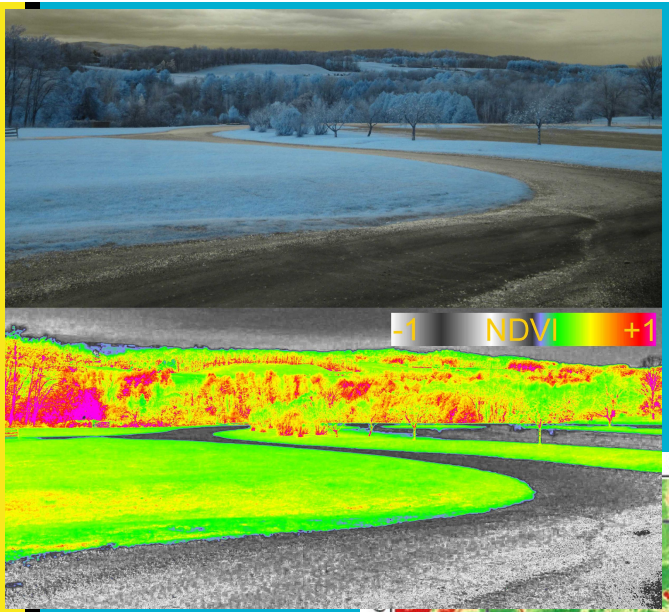
Campaña de fumigación masiva contra el dengue en Iquitos

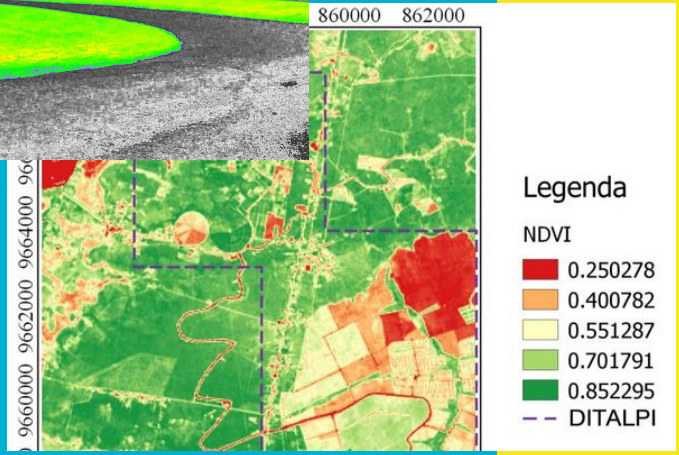
25/03/2020 // No hay comentarios

Del jueves 27 hasta el 12 de marzo Reunión de emergencia

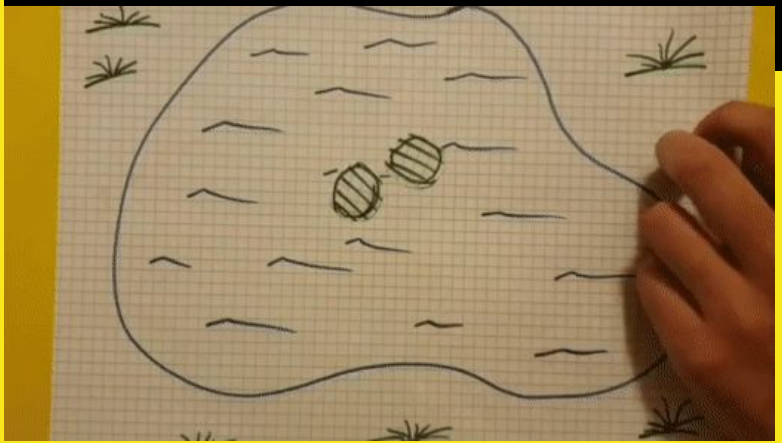
Novas colunas - concentrando dados

Colunas NVDI - Índice de Vegetação por Diferença Normalizada usado para caracterizar a cobertura vegetal, seu estágio de crescimento. Também é capaz de registrar áreas com pouca ou nenhuma presença vegetacional e massas d'água.
O dataset original possui dados de quatro pontos das cidades.
Criação de colunas com nvdi máximo e mínimo.

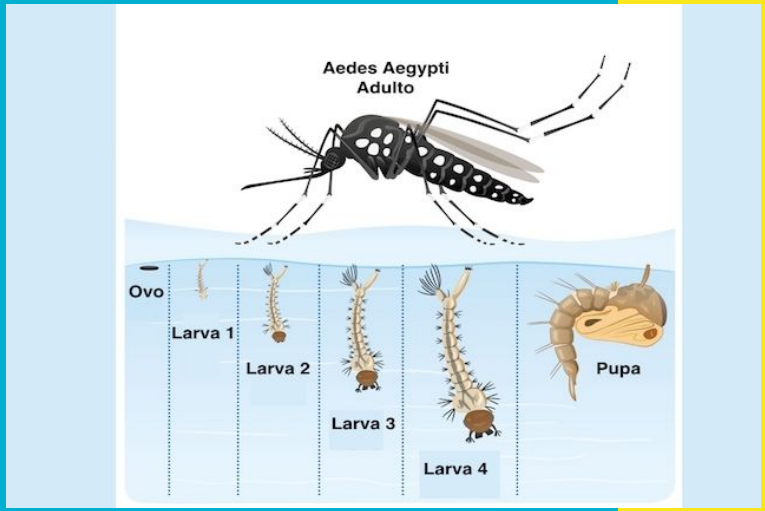




os dados semanais estanques não contemplam a realidade subjacente às contaminações, especialmente de surtos.
O crescimento exponencial, característico dos surtos, decorrem do acúmulo de eventos, como no “exemplo das vitórias-régias”.



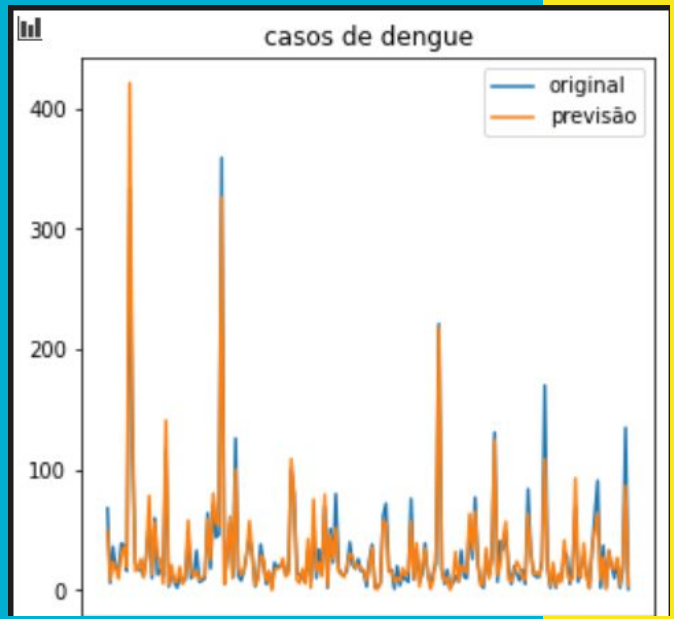
Levando em conta o ciclo de vida do vetor, período de incubação da doença, data de relato e por fim, supondo que a doença prospera exponencialmente quando as condições favoráveis persistem, adotou-se o intervalo entre 15 a 4 semanas como aptos a influenciar nos casos totais registrados.



Resultado cidade de San Juan

Dados de 50 mil testes
aplicado o mesmo procedimento de validação cruzada (número de dados menor, divisão em 5 grupos, teste de 20%)
colunas com valores do resultado de testes e duas medidas de modelos estatísticos.
mean_absolute_error é parâmetro de avaliação da competição oficial do dataset

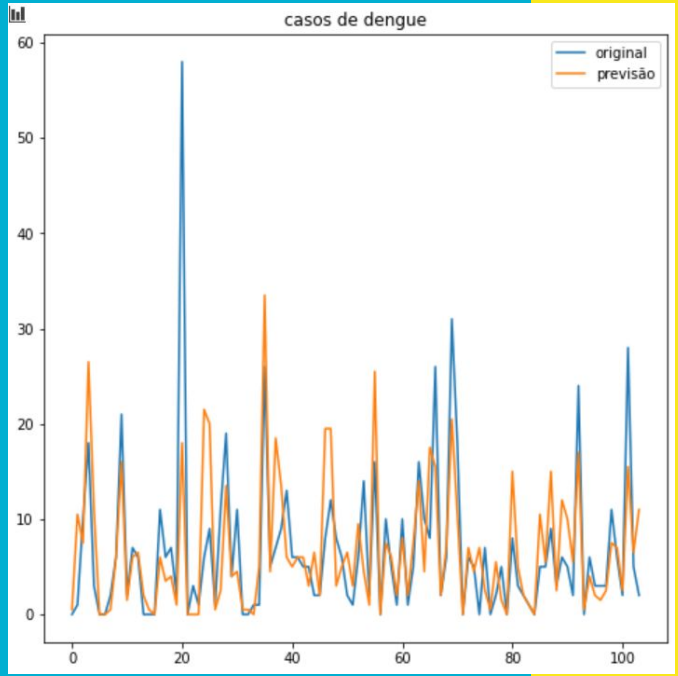
	treino	teste	mean_absolute_error	r2_score
count	50000.000000	50000.000000	50000.000000	50000.000000
mean	0.979997	0.931097	7.516129	0.931014
std	0.001206	0.012580	1.082006	0.012984
min	0.977904	0.908090	6.403226	0.907235
25%	0.979945	0.928043	6.666667	0.928192
50%	0.980162	0.936131	6.897849	0.935625
75%	0.980316	0.939838	8.505376	0.940212
max	0.981656	0.943382	9.107527	0.943806



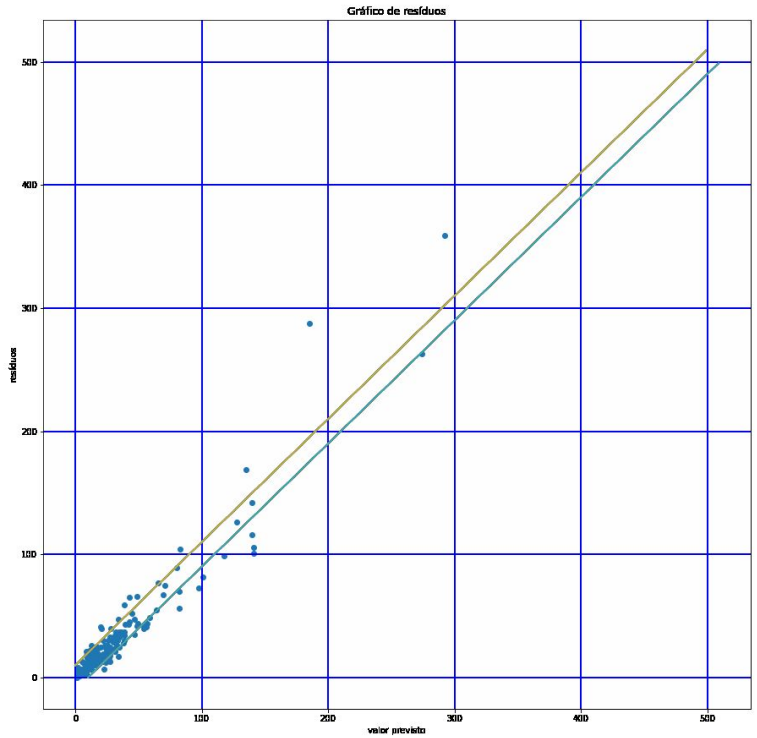
Resultado cidade de Iquitos

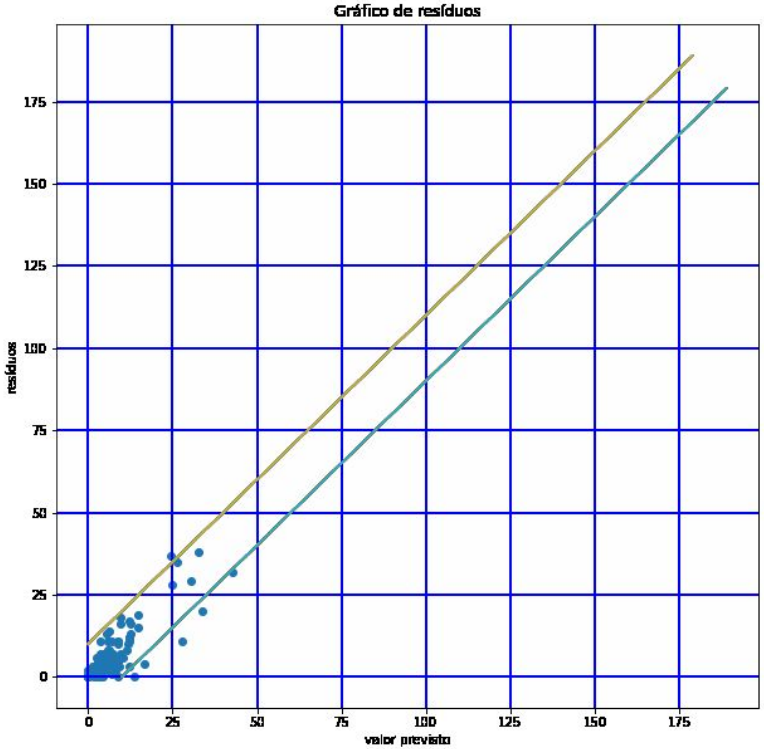
Dados de 50 mil testes
aplicado o mesmo procedimento de validação cruzada (número de dados menor, divisão em 5 grupos, teste de 20%)
colunas com valores do resultado de testes e duas medidas de modelos estatísticos.
mean_absolute_error é parâmetro de avaliação da competição oficial do dataset

	treino	teste	mean_absolute_error	r2_score
count	50000.000000	50000.000000	50000.000000	50000.000000
mean	0.869212	0.601805	3.603128	0.598784
std	0.038734	0.156839	0.217995	0.161191
min	0.805460	0.318857	3.277778	0.308681
25%	0.843761	0.567271	3.504854	0.561218
50%	0.893062	0.659072	3.601942	0.657316
75%	0.895366	0.683946	3.689320	0.684903
max	0.908413	0.779880	3.941748	0.781804



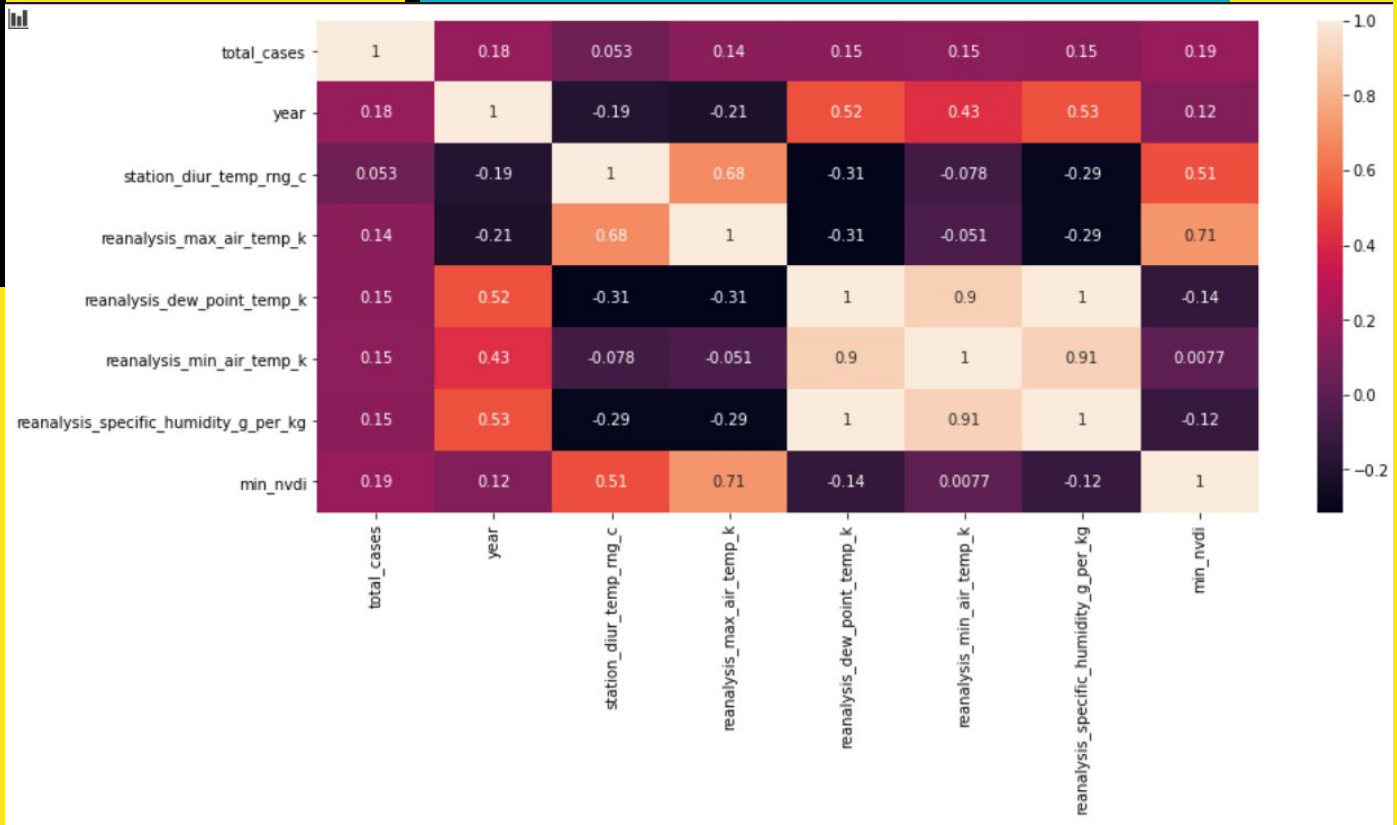
comparação dos resultados





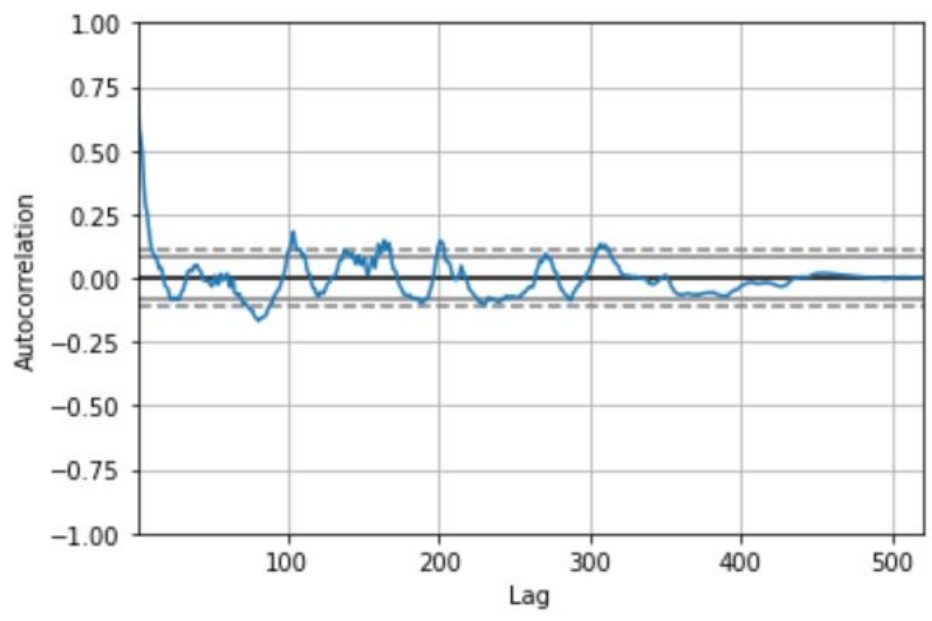
principais dificuldades em treinar o modelo no caso Iquitos

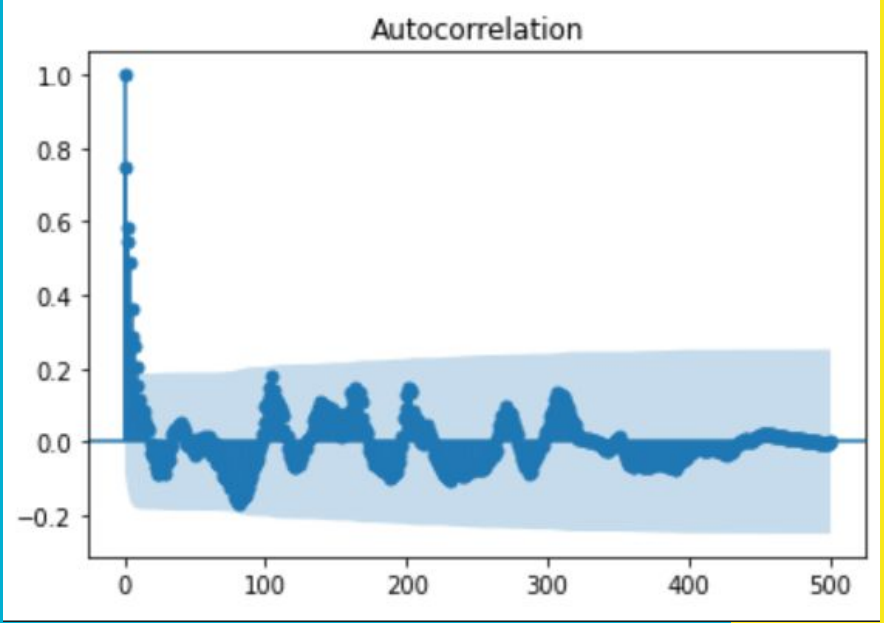
quantidade menor de dados e baixa correlação com as variáveis independentes



tentativa de encontrar autocorrelaçao

Diante das fracas correlações com as variáveis independentes, foram feitos testes de autocorrelação





Teste final

competição aberta há mais de 2 anos;

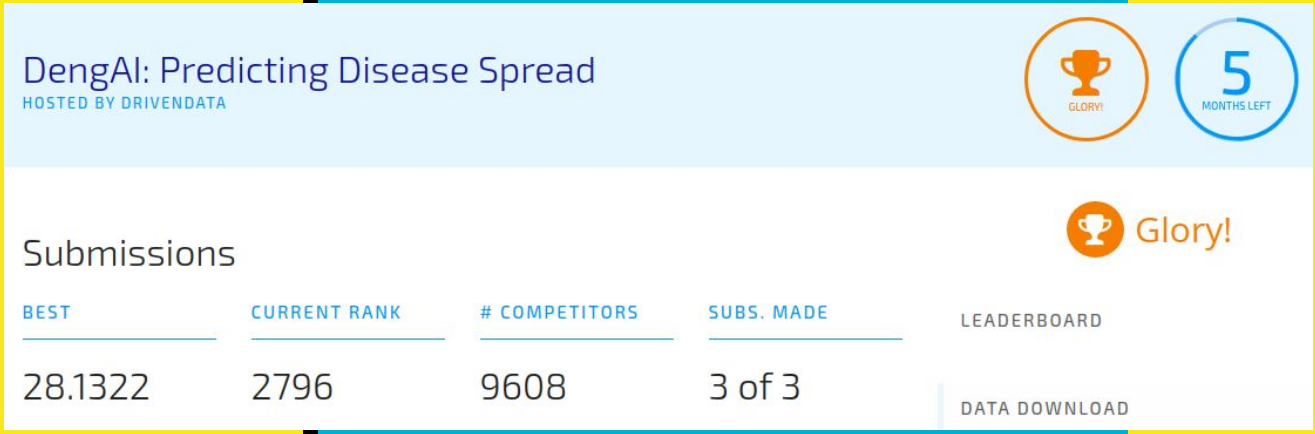
mais de 9.600 participantes;

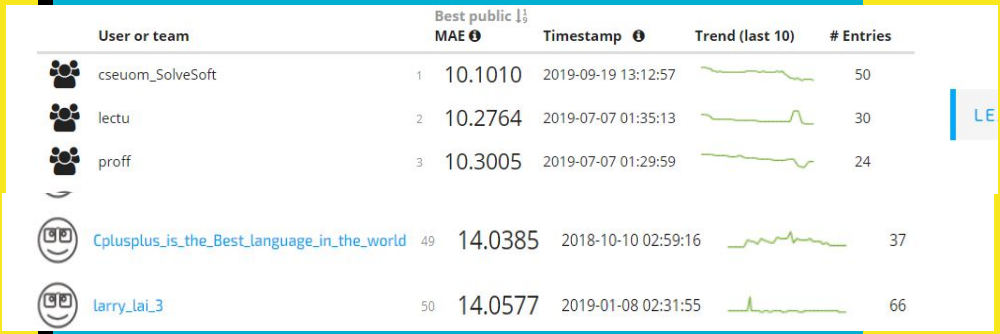
menor erro 10,1010;

descobrimos apenas ontem que podíamos submeter uma resposta imediatamente (limitada a 3 por dia);

nosso erro foi de 30,1514;

ficamos na posição 2.796º (dentro do último terço, 29,10%).





OBRIGADO