Relatório Atividade II - Ajuste da Curva Polinomial

William H. Sumida

April 2020

1 Teoria

A Equação 1.1 representa uma equação polinomial *y(x, w)* de grau *M*, sendo w_i um coeficiente que multiplica todo x^i para todo i variando de 0 a *M*.

A equação 1.2 tem como objetivo calcular um valor que representa a diferença entre dois conjuntos de dados. No contexto do livro, este valor representa o erro médio de uma função polinomial que estima os valores da curva de um gráfico. É uma ferramenta importante para saber se os pontos de uma função polinomial se ajustam a uma determinada curva.

Para encontrar um polinômio que represente um conjunto de dados, dividimos o conjunto em dados de treino e teste. Os dados de treino serão utilizados para criar uma função polinomial e os de teste para ver se o modelo se ajusta a dados que ele não conhece.

Na primeira etapa, podemos pré-definir um intervalo de 0 a N e calcular os coeficientes dos polinômios de grau 0 a N com base no conjunto de dados de treino. Agora podemos comparar os dados gerados por cada polinômio com os dados de treino. É importante observar o comportamento da curva em um gráfico e não apenas comparar os datasets, pois o modelo pode prever corretamente os dados de treino e apresentar uma curva distinta do dataset original, fenômeno conhecido como *Over-fitting*.

Para evitar o *Over-fitting*, podemos analisar as diferentes iterações de grau M do polinômio levando em consideração os pontos de inflexão do conjunto de dados de treino, já que o grau (M) do polinômio está correlacionado com os pontos de inflexão da curva dos dados originais.

Um dos problemas ao analisar os pontos de inflexão da curva do polinômio e do conjunto de dados de treino, é a baixa quantidade de amostras para treino. Fazendo com que o modelo não tenha uma referência em certos intervalos, fenômeno chamado *Under-fitting*.

2 Prática

A base de dados das mortes causadas pelo COVID-19 no Brasil apresenta 96 amostras. Podem ser observadas na tabela 1 as colunas e os 5 primeiros e últimos elementos do conjunto de dados (ordenados por data).

	dateRep	day	month	year	cases	deaths	countriesAndTerritories	geoId	countryterritoryCode	popData2018
0	07/04/2020	7	4	2020	926	67	Brazil	BR	BRA	209469333
1	06/04/2020	6	4	2020	852	54	Brazil	BR	BRA	209469333
2	05/04/2020	5	4	2020	1222	73	Brazil	BR	BRA	209469333
3	04/04/2020	4	4	2020	1146	60	Brazil	BR	BRA	209469333
4	03/04/2020	3	4	2020	1074	58	Brazil	BR	BRA	209469333
..	...	...	...	...	...	...	...	...	...	...
91	04/01/2020	4	1	2020	0	0	Brazil	BR	BRA	209469333
92	03/01/2020	3	1	2020	0	0	Brazil	BR	BRA	209469333
93	02/01/2020	2	1	2020	0	0	Brazil	BR	BRA	209469333
94	01/01/2020	1	1	2020	0	0	Brazil	BR	BRA	209469333
95	31/12/2019	31	12	2019	0	0	Brazil	BR	BRA	209469333

Table 1: Dados brutos das mortes causadas pelo COVID-19 no Brasil.

Podemos observar na figura 1 a distribuição dos dados do número de mortes em função do tempo.

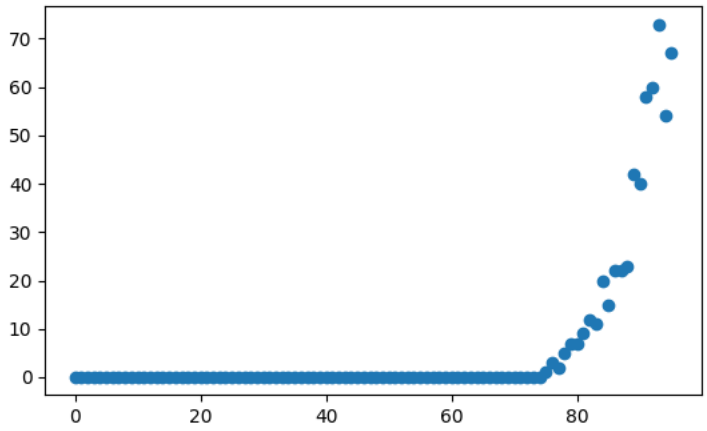

Figure 1: Número de mortes em função do tempo.

Conforme enunciado, os datasets de treino (figura 2) e teste (figura 3) foram separados pelos índices pares e ímpares, respectivamente. Obtendo uma melhor representatividade, pois como podemos observar na figura 1, os dados poderiam desbalancear os datasets.

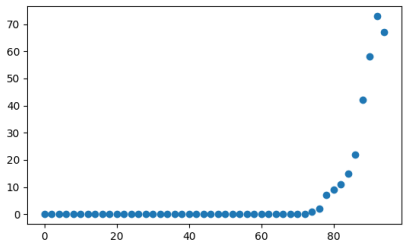

Figure 2: Dados de treino.

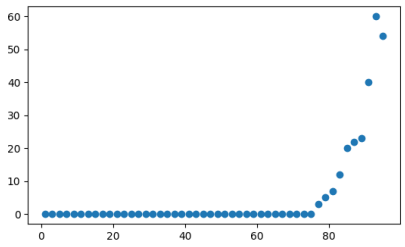

Figure 3: Dados de teste.

Para construir a função de ajuste, foi usado o método *polyfit* da biblioteca *Numpy*. Foram gerados 96 funções de ajuste, sendo elas, polinômios de ordem 0 a 95. Para cada polinômio foi calculado o número de mortes dos mesmos dias do dataset de treino.

Com objetivo de encontrar o polinômio que melhor se ajusta aos dados, foi criado um método que calcula o Erro Médio Quadrático entre o dataset de treino e o dataset gerado pelo modelo. Os dez menores Erros podem ser observados na tabela 2.

Como podemos observar na tabela 2, concluimos que o modelo que teve o melhor ajuste foi o de ordem 19.

Grau	Erro Médio Quadrático
19	2.3209079965942543
18	2.321930424090838
17	2.3403024164518653
9	2.8291890559007893
16	2.8559805457998726
10	2.860680389664886
8	3.141068264179189
11	3.280131482559776
20	3.356653255712421
7	3.5773903114620444

Table 2: 10 menores erros.

Podemos observar a relação entre os dados originais na figura 4, sendo os pontos azuis os dados originais do dataset e a linha laranja os dados preditos pelo modelo.

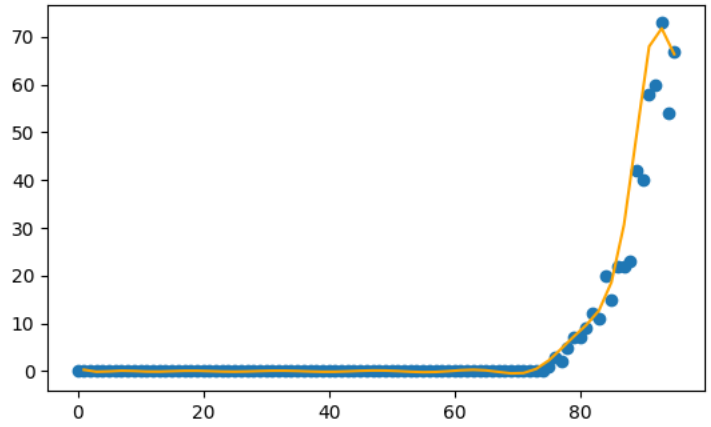

Figure 4: Pontos: Dados originais, linha: previsões.

A tabela 3 ilustra todos os coeficientes do polinômio de décimo nono grau.

Grau	Coeficiente
0	-3.510073980715606 x $10^{-}30$
1	8.691290313044615 x $10^{-}27$
2	-4.91062250202904 x $10^{-}24$
3	1.3187079971232638 x $10^{-}21$
4	-1.9578133843540287 x $10^{-}19$
5	1.4664893524428947 x $10^{-}17$
6	1.0059927021507953 x $10^{-}16$
7	-1.5222069654199412 x $10^{-}13$
8	1.8536422447061876 x $10^{-}11$
9	-1.31142732222998 x $10^{-}09$
10	6.3010256929 x $10^{-}08$
11	-2.147786640878447 x $10^{-}06$
12	5.240584678674485 x $10^{-}05$
13	-0.0009062032752513819
14	0.010816363271510065
15	-0.08508423834243345
16	0.408136097305051
17	-1.0373036649361869
18	1.0112006122199617
19	-0.009925753435447996

Table 3: Coeficientes do polinômio de grau 19.