Programa de Pós-Graduação em Engenharia de Recursos Hídricos e Ambiental

Algoritmos Genéticos

Jorge Centeno

Introdução

Algoritmos genéticos são um conjunto de algoritmos de otimização inspirados no processo evolutivo natural.

Técnicas de otimização são utilizadas para descobrir a solução óptima (ou sub-óptima) dentro de um universo de possíveis soluções.

Um exemplo: Dada uma série de pares de dados (X,Y) encontre os parâmetros da melhor função linear (dois parâmetros, a e b)

$Y=a * x + b$

Dado:

X=[13.11 2.19 18.67 3.74 5.32 15.95 9.75 15.37 7.92 5.45]

Y =[11.55 6.09 14.33 6.874 7.66 12.978 9.87 8.96 7.72]

Se você não sabe fazer a regressão linear, uma solução é arbitrar valores e avaliar o erro. Se o erro for baixo, trata-se de uma boa aproximação. Se o erro for grande, então, devemos buscar uma nova solução.

Podemos calcular TODAS possíveis combinações de a e b para mapear toda a superfície de erro, mas isso pode se tornar impraticável.

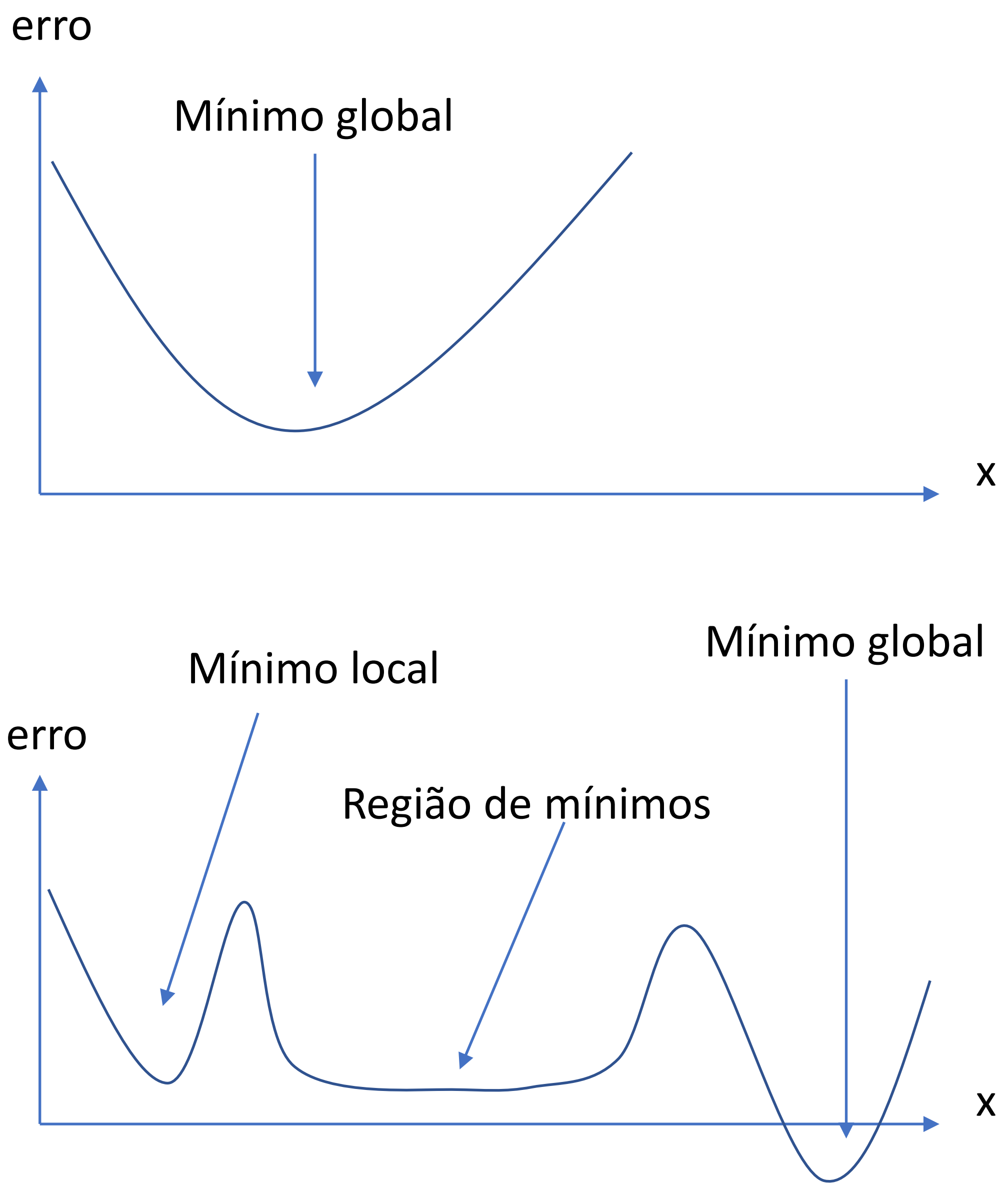
Superfície de erro

No exemplo, o erro é mínimo para apenas uma combinação.

Mínimo global.

Porém em outros problemas podemos ter:

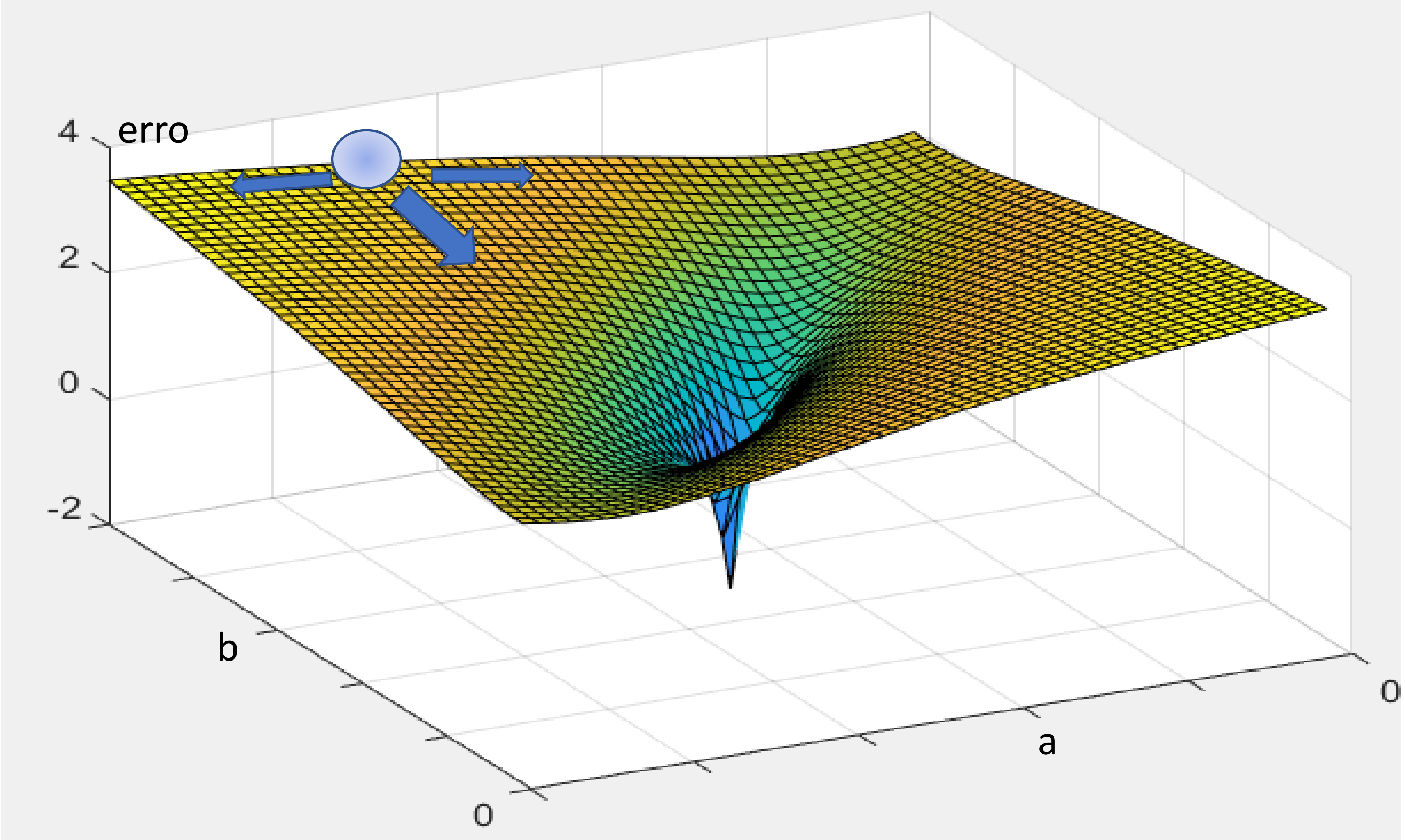
- mínimos locais (vários)
- uma região de mínimos
- Vários mínimos com o mesmo valor



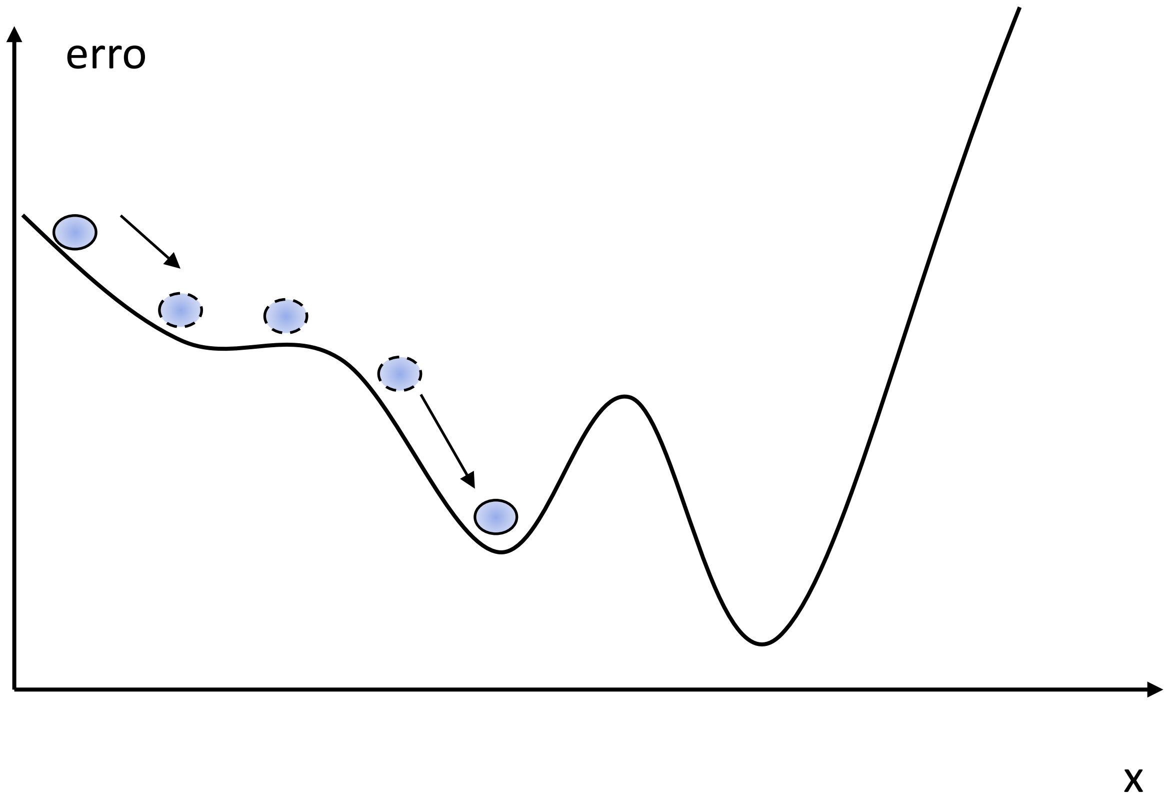
Buscar o mínimo: ex: Gradiente

Busca

Gradiente?



Mínimos locais



A busca pode se estancar em um mínimo local, o que não é desejado.

Otimização

No processo devemos definir:

- Espaço de busca: definido pela faixa de variação das variáveis.
- Função objetivo: uma função que é usada para avaliar a adequação de uma solução para o problema proposto.

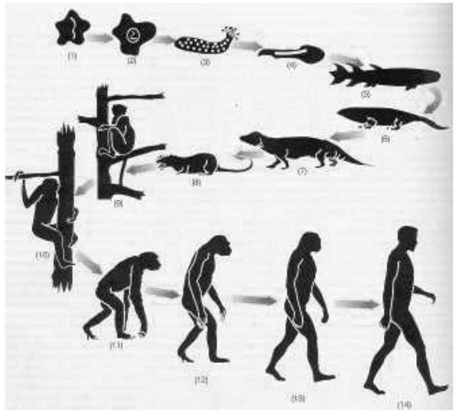
O objetivo é minimizar (ou maximizar) a função objetivo de maneira a se aproximar da solução ideal (o mínimo/máximo global)

Algoritmos genéticos: Ideia

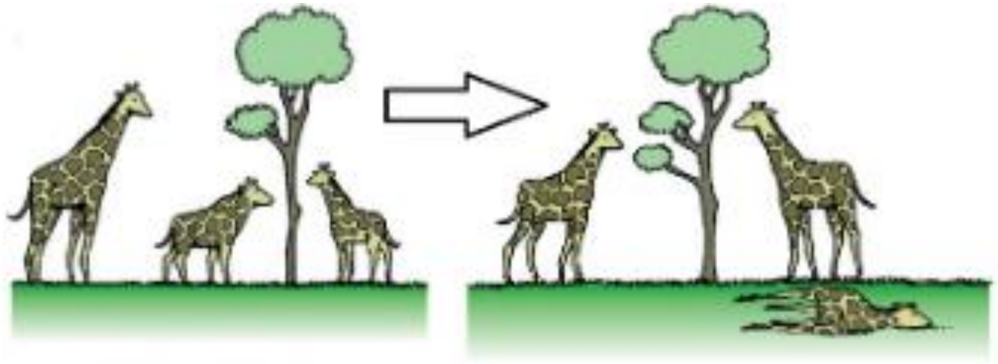
A ideia é aplicar os conceitos de evolução, seleção natural e herança genética, na busca da solução óptima.

Não é pesquisado todo o espaço de busca

Selecionar algumas regiões aleatoriamente e conduzir uma busca guiada do mínimo global.



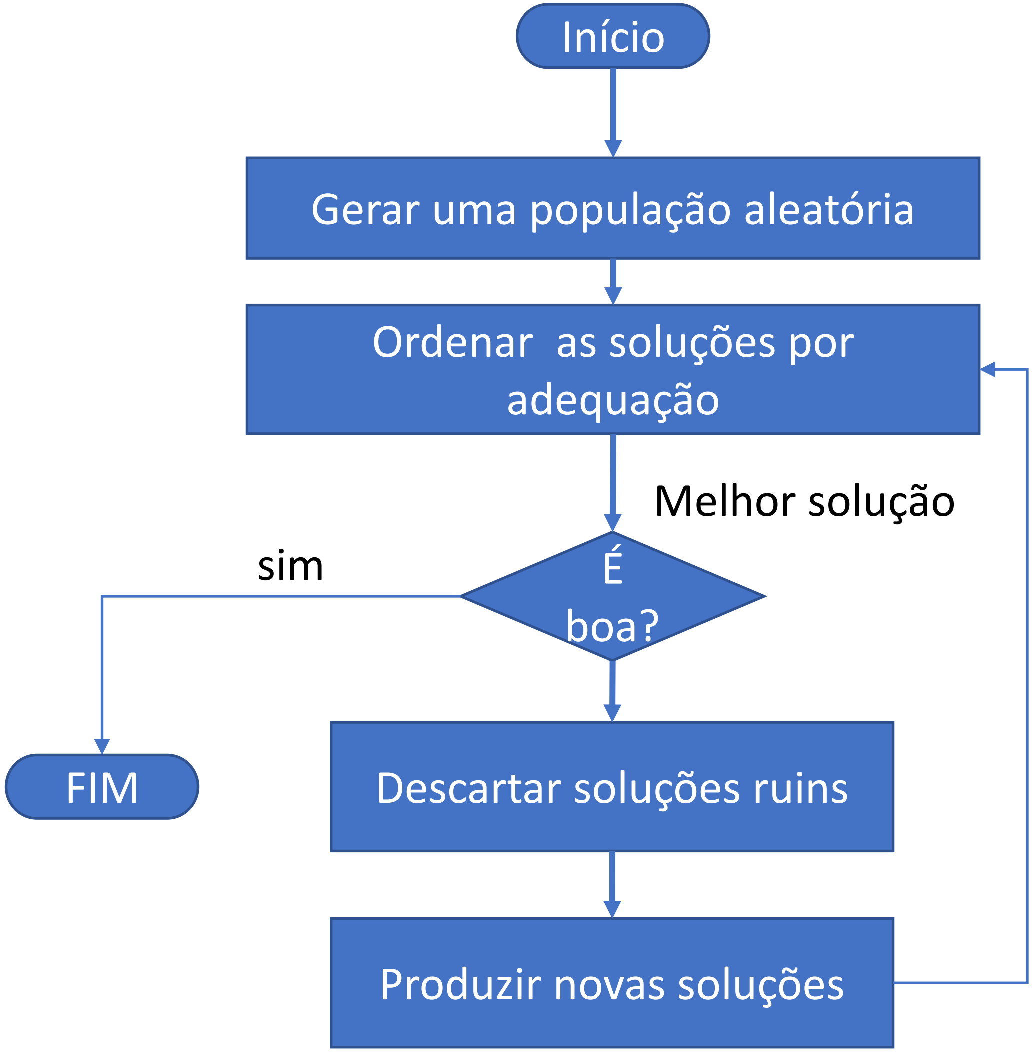
É ideal para lidar com grandes conjuntos de dados e fenómenos difíceis de modelar.



Darwinismo (https://www.todamateria.com.br/selecao-natural/)

O algoritmo

Antes de nada, as variáveis devem ser transformadas a uma “cadeia genética”, para simular uma evolução natural.



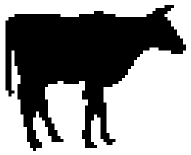
exemplo

EM AG,

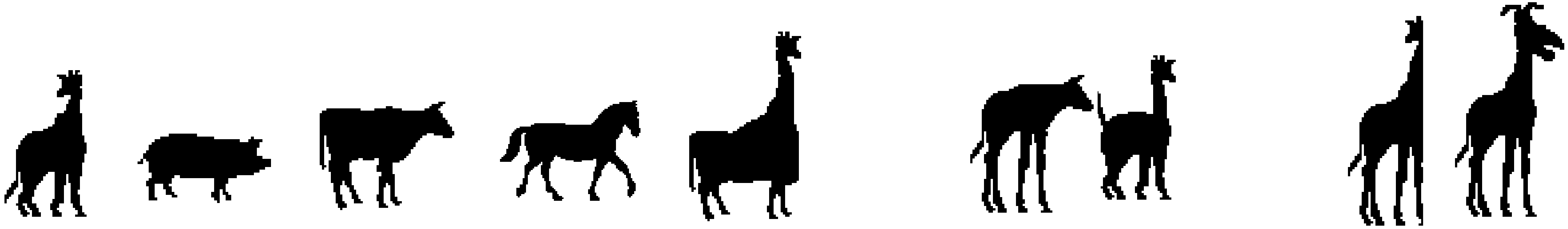
uma solução é considerada um indivíduo.

Ex: (a=0.1 e b=1.0) é um individuo

Ex: (a=0.5 e b=9.0) é outro individuo



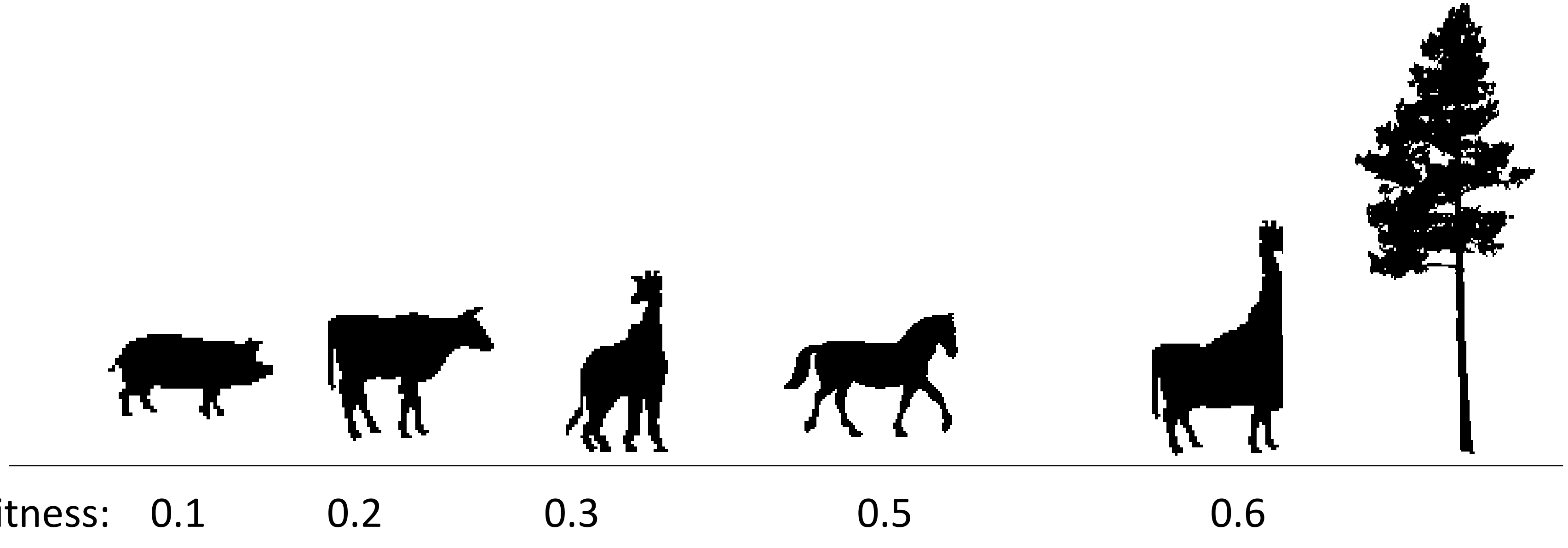
Um conjunto de indivíduos é uma população



avaliação

Cada solução é analisada e um grau de adequação à solução do problema é calculado (função objetivo), o fitness, para cada solução.

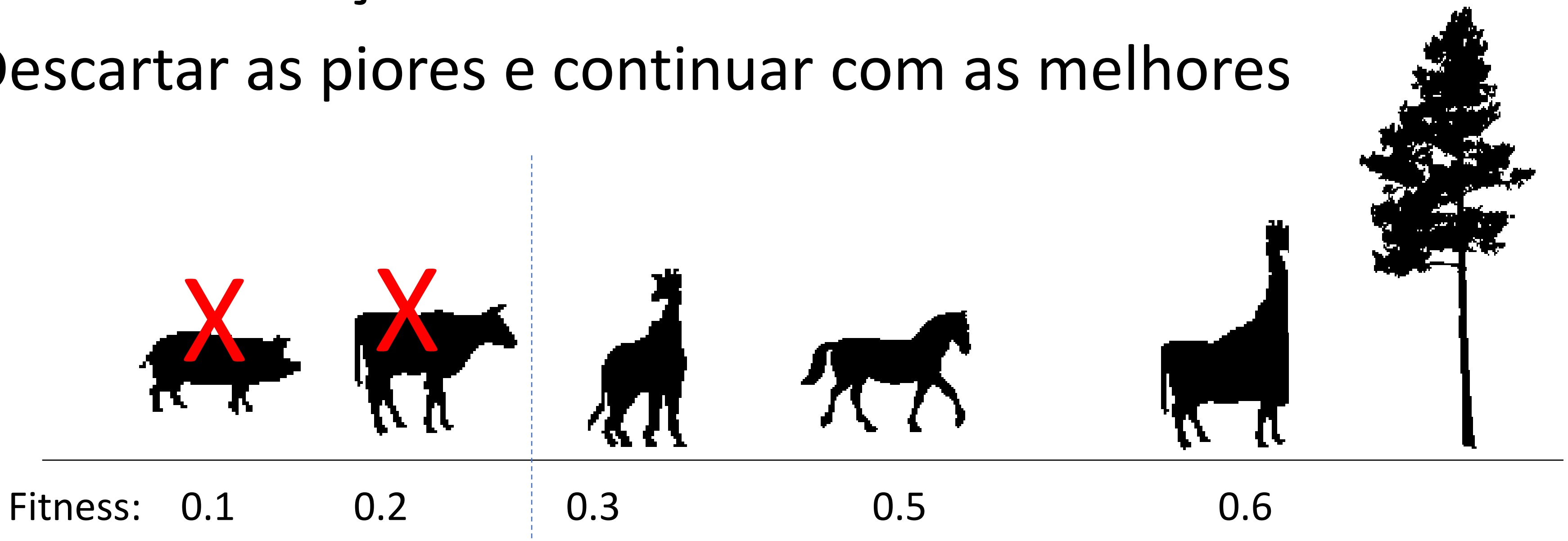
Ex: Quem pode comer as folhas mais altas e correr do predador?



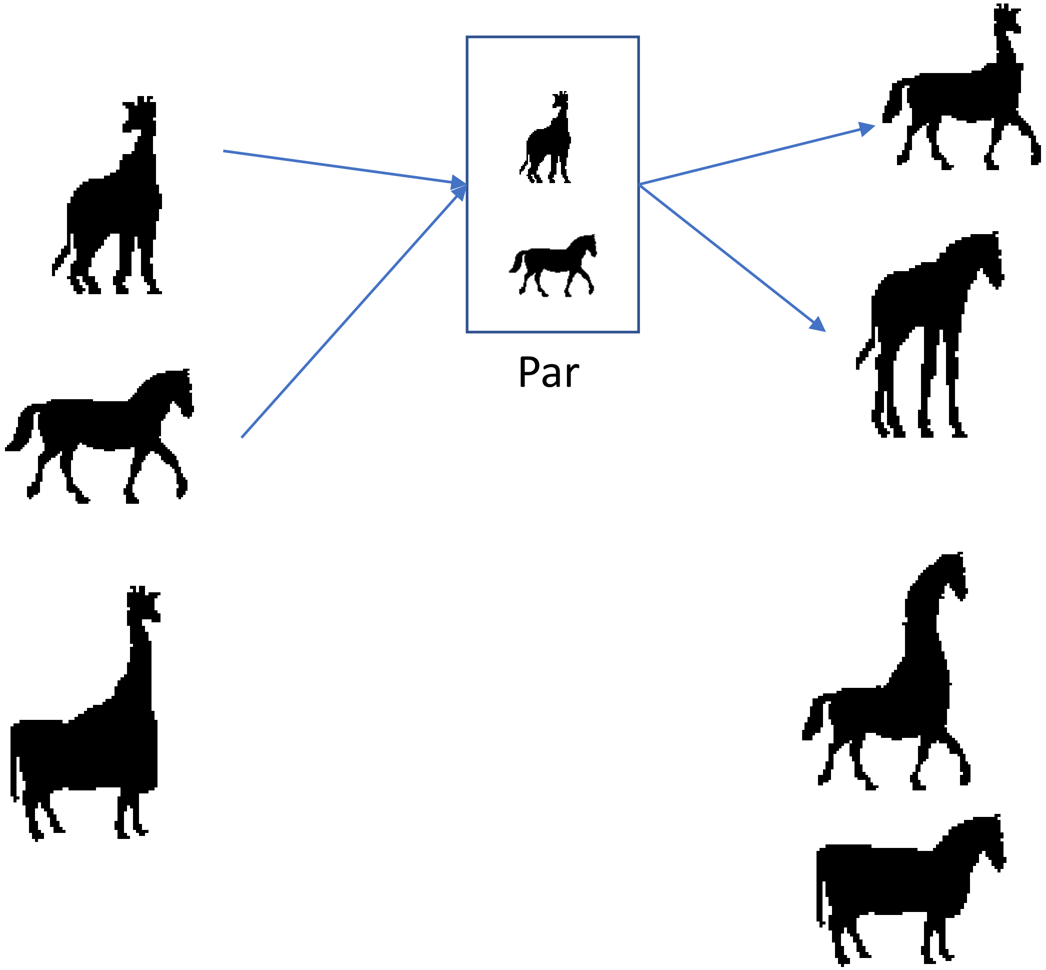
Seleção

A melhor solução ainda não é ótima.

Descartar as piores e continuar com as melhores



Gerar novas soluções (cruzamento)



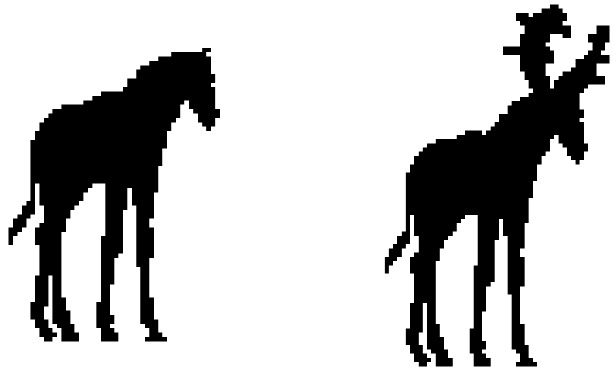
As novas soluções são combinações das anteriores.

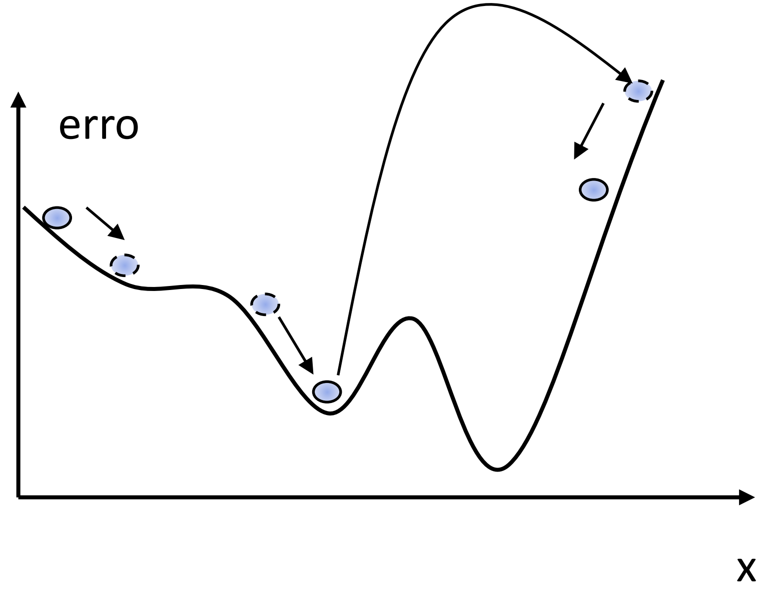
Espera-se que algumas delas herdem (herança) as melhores características dos pais.

Mutação

Para evitar mínimos locais e aumentar o espaço de busca, uma mutação aleatória é introduzida em parte da população nova.

Isto pode ajudar ou não, é aleatória.



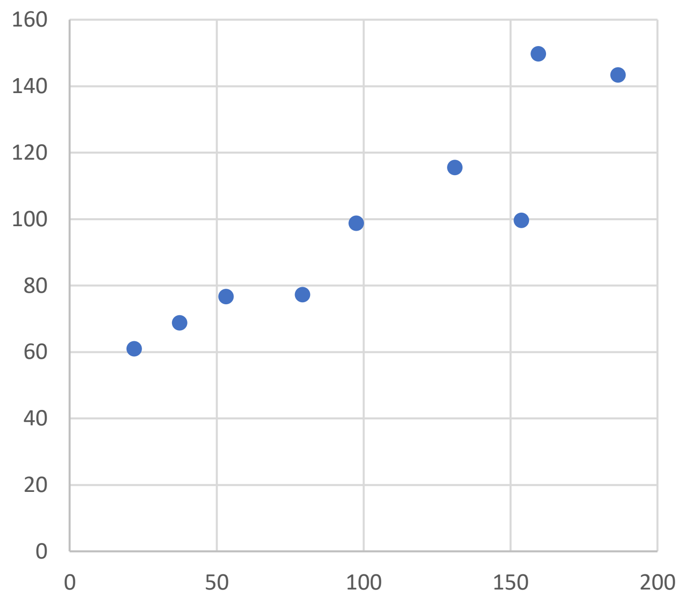


exemplo

Dada uma série de pares de dados (X,Y) encontre os parâmetros da reta

Y=a * x + b

x	Y
131,1	115,5
21,9	60,9
186,7	143,3
37,4	68,74
53,2	76,6
159,5	149,7
97,5	98,7
153,7	99,6
79,2	77,2



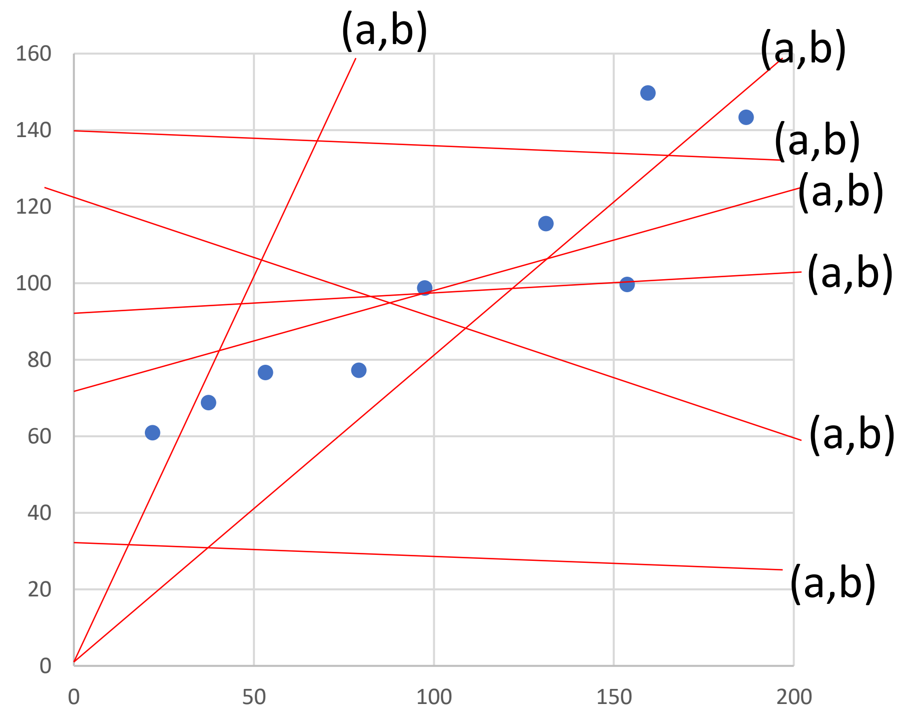
exemplo

Qualquer conjunto de pares (a,b) é uma solução possível

Y=a * x + b

Neste caso, podemos medir a adequação da solução com ajuda da correlação.

Função objetivo: “maximizar a correlação”



Codificação binária

Pode usar...
https://www.exploringbinary.com/binary-converter/

Os AG processas as soluções como “cromossomos”.

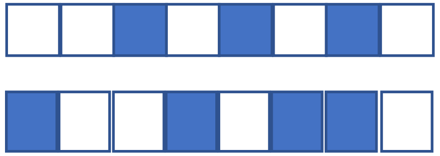
O cromossomo é uma estrutura de dados, tradicionalmente uma cadeia de bits, que armazena os valores das variáveis.

Exemplo: Se consideramos dois valores inteiros: a=10 e b=150, podemos transformar estes valores a uma série de 8 bits (0 ou 1).

A representação binária de

10= [00001010]= $0*2^0+1*2^1+0*2^2+1*2^3+0$ ...= 2 +8 =10

150= [10010110]=2+4+16+128

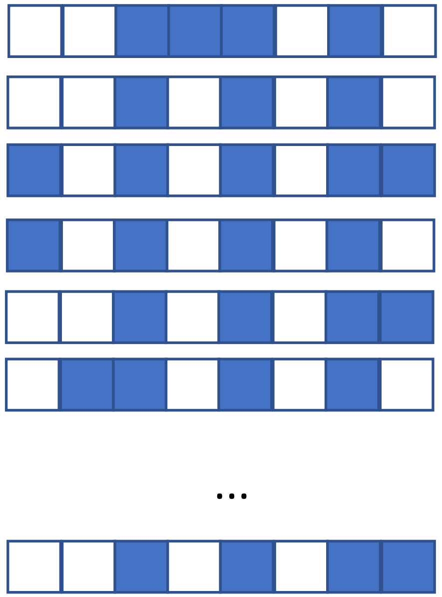


(10,150)

população

Agora, podemos repetir o mesmo processo para produzir uma grande quantidade de cadeias, com bit 0 ou 1, aleatórios. Cada cadeia representa uma possível solução.

Deve ser possível, também, decodificar a cadeia para recuperar os valores armazenados.



Fitness (adequação)

Cada solução pode ser usada para resolver o problema. Em nosso exemplo, podemos usar cadeias contendo (a,b) para aproximar uma reta aos pares de pontos (x,y) dados. Com $y1=a*x + b$.

Podemos então avaliar o erro cometido em cada ponto

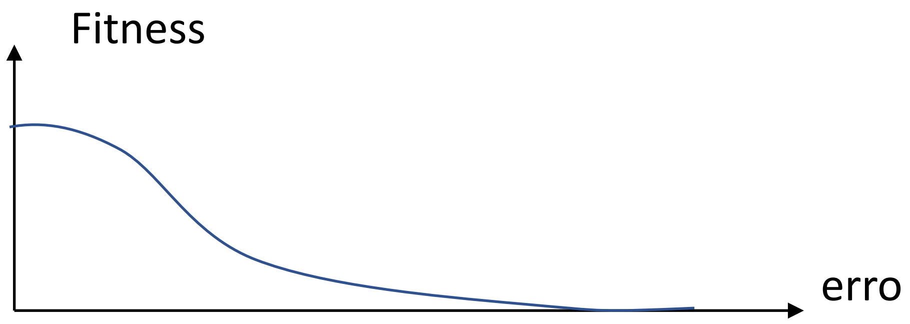
$$\text{erro}=|y1-y|$$

E calcular o erro médio (total) de cada solução.

O valor do erro descreve a adequação da solução ao problema. Quanto menor o erro, melhor a solução. Então, quanto menor o erro, maior o fitness.

fitness

Transformamos o “erro” em um valor de fitness dentro de um intervalo válido.



E agora cada indivíduo pode ser associado a um “fitness”, um grau de adequação.

Fitness=0.681

Organizar segundo o fitness

indivídio	codigo	fitness
1	01010101110101	0,65
2	01010101010101	0,60
3	01010101011111	0,59
4	01010101010101	0,41
5	10101010111010	0,33
6	01010101010100	0,30
...	...	...
N	01011111010101	0,01

E verificar se o melhor indivíduo atingiu o nível ideal (pode ser perto de 1, não necessariamente =1).

Se não, selecionar

indivídio	codigo	fitness
1	01010101110101	0,65
2	01010101010101	0,60
3	01010101011111	0,59
4	01010101010101	0,41
5	10101010111010	0,33
6	01010101010100	0,30
...	...	...
N	01011111010101	0,01

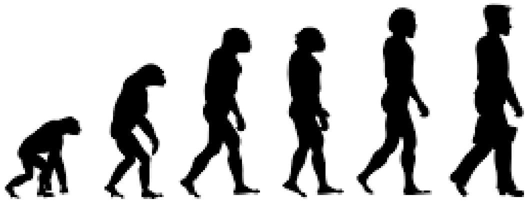
sobrevivem

descartados

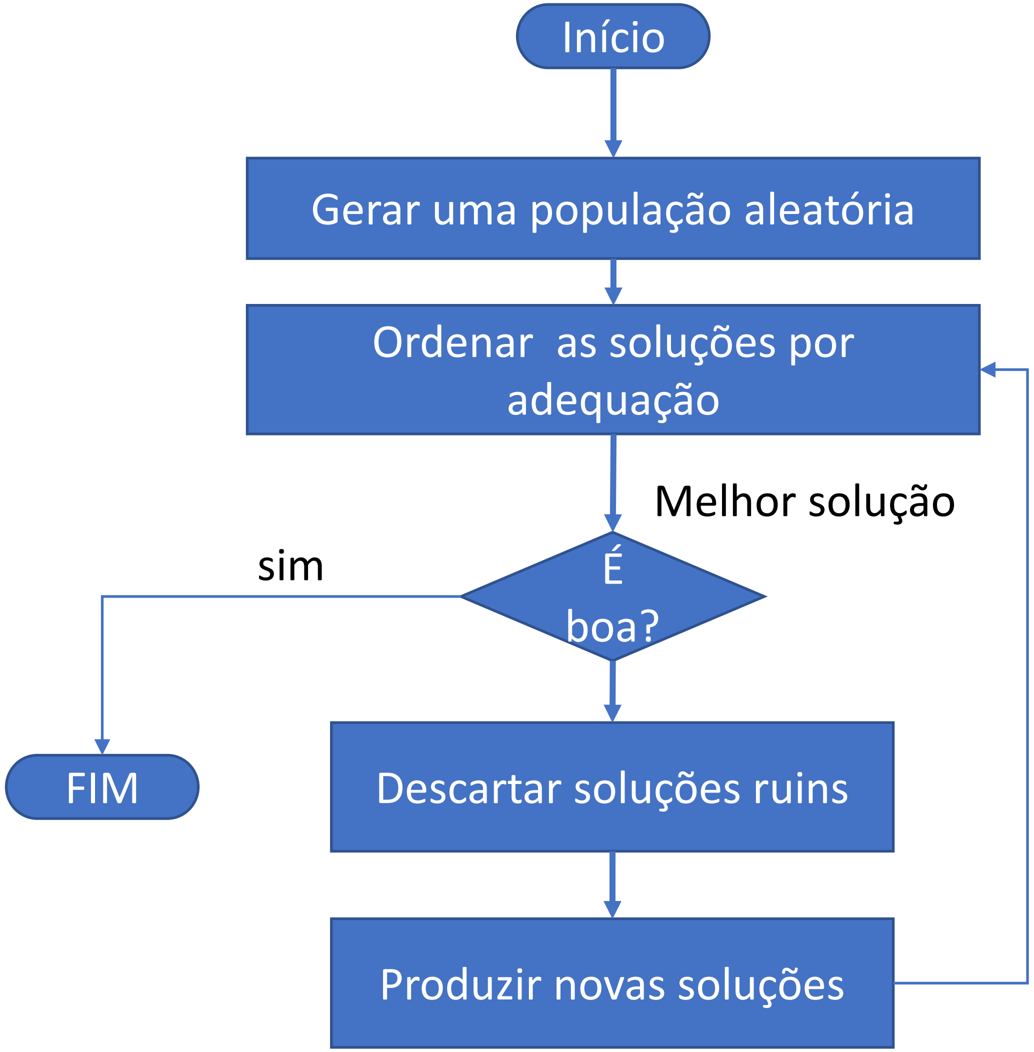
Selecionar as melhores soluções. Quantas? Isso você decide.

O algoritmo

Ok. Até aqui está indo bem, selecionamos os melhores, evolução de Darwin.



... Agora vem a genética



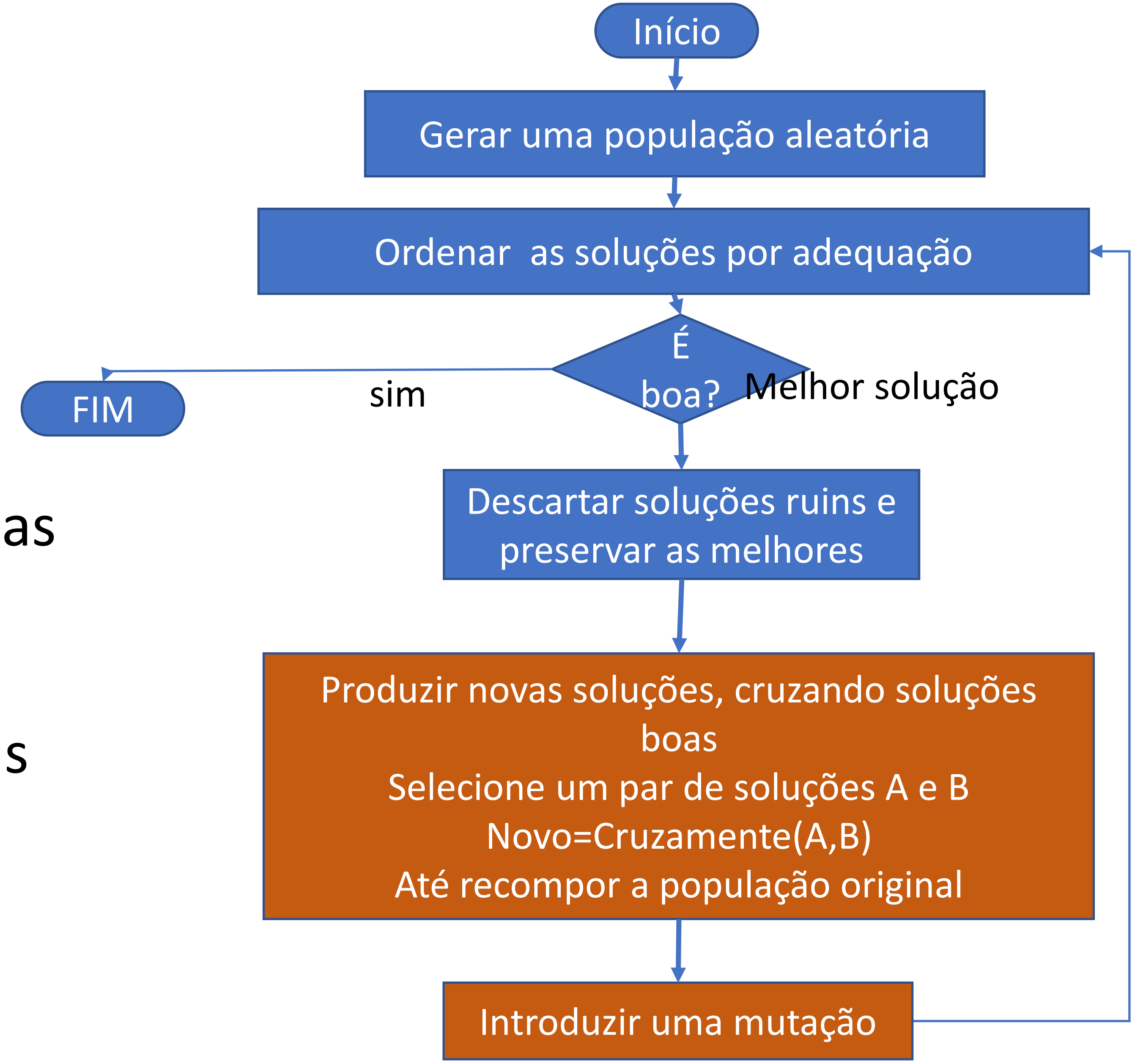
O algoritmo

passos:

Selecionar melhores soluções

Escolher um par de boas soluções

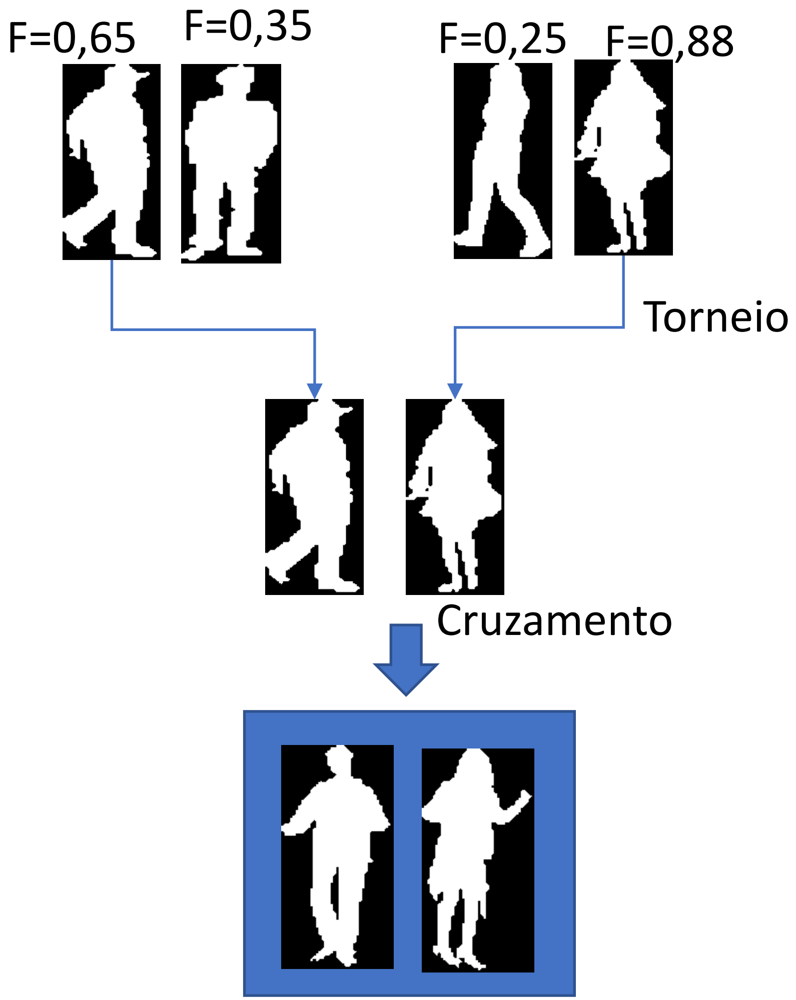
Computar duas novas soluções a partir destas soluções



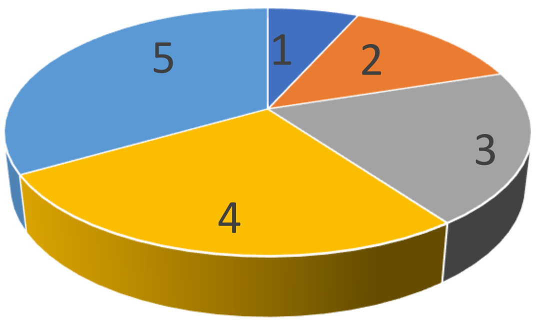
Seleção de par dentre os sobreviventes

Aleatório: escolher dois indivíduos ao azar

Torneio: Selecionar um par e comparar o fitness e selecionar o melhor indivíduo (A). Escolher outro par de indivíduos e novamente selecionar o melhor indivíduo (B).



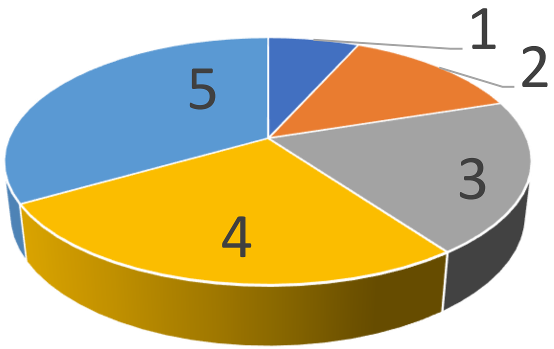
Seleção



Por classificação: Os indivíduos são ordenados em ordem decrescente de fintes. Os primeiros recebem uma probabilidade maior de ser escolhidos. Pode ser usada uma regra linear para calcular a probabilidade. Os indivíduos mais fortes tem maior chance de participar na seleção.

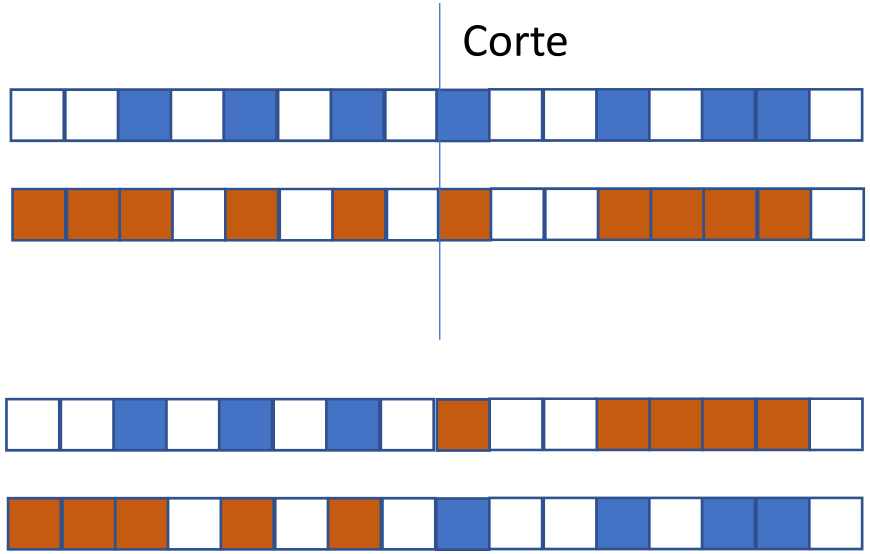
individuo	Fitness	Prob.linear
1	0,65	100
2	0,6	85
3	0,59	70
4	0,41	55
5	0,33	40

Seleção por roleta: É feita uma escolha dando probabilidades diferentes a cada indivíduo. A probabilidade é proporcional ao valor do fitness.



Cruzamento

Consiste em utilizar parte da cadeia genética de um indivíduo e parte do outro. Para isto, um ponto de corte é selecionado (pode ser na metade)

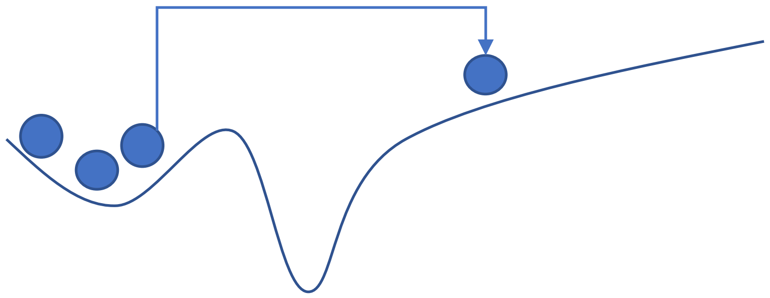


Os novos indivíduos são somados à população sobrevivente para recompor o número de soluções.

herança

A ideia é que, ao combinar partes de boas soluções, eventualmente possam ser geradas soluções ainda melhores, que herdem o melhor de cada pai. Porém , isto não é garantido, por isso são feitas vários cruzamentos e também o processo se repete iterativamente.

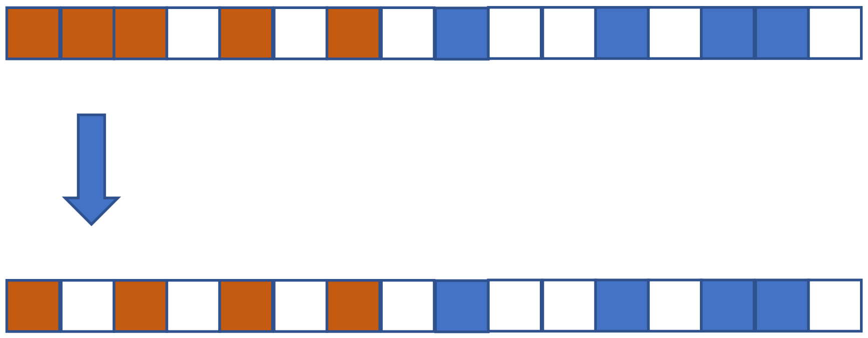
Mutação



Para evitar que o processo evolutivo se estacione, ficando parado em um mínimo local, por exemplo.

E com a finalidade de acelerar o processo, novo material genético é introduzido a cada iteração, através da mutação.

A mutação consiste em inverter um bit aleatoriamente, o que cria um individuo totalmente diferente. Se esta mutação for positiva, o processo pode convergir mais rápido.



A taxa de mutação deve ser mantida baixa, para não prejudicar a convergência do processo.

Elitismo

Uma prática comum é preservar uma percentagem das melhores soluções para não substituir completamente a população por novas soluções. Assim, se garante que o fitness não caia.

indivídio	codigo	fitness
1	01010101110101	0,65
2	01010101010101	0,60
3	01010101011111	0,59
4	01010101010101	0,41
5	10101010111010	0,33
6	01010101010100	0,30
...	...	...
N	01011111010101	0,01

Geração

Assim, os operadores genéticos são:

- Seleção
- Cruzamento
- Mutação
- Elitismo

Após aplicar estes operadores a uma população, uma nova geração é criada e ela deve de novo ser avaliada (fitness) e o processo repetido.

É esperado que, após algumas iterações, novas soluções melhoradas emerjam e o processo caminhe para o mínimo global.

Parâmetros

Tamanho da população: populações maiores podem demandar maior tempo para calcular uma iteração e demorar o processo iterativo. Por outro lado, mais elementos oferece a vantagem de ter mais possibilidades.

Número de iterações: Poderíamos parar o processo quando se chegar ao mínimo global, ou seja, quando o erro for nulo. Como isto não pode ser atingido na prática (erros numéricos, medições erradas), pode ser fixado um limiar mínimo de fitness necessário para o processo parar ou fixar um número máximo de iterações.

Taxa de sobreviventes

Quanto maior for esta taxa, mais rapidamente novos indivíduos serão criados e inseridos na população. Mas... Se substituir a maior parte da população se corre o risco de perder a aptidão atingida.

Poucos sobreviventes incluem maior quantidade de novos indivíduos, correndo-se o risco de se perder bons indivíduos existentes. Mas pode acelerar o processo evolutivo.

Muitos sobreviventes desacelera o processo de criação de novas soluções, tornando a busca mais lenta.

Taxa de cruzamento = N - Nsobreviventes

Recomendações

- **Taxa de Cruzamento**
  deve ser alta, cerca de **80%-95%**. Algumas experiências recomendam uma taxa de 60%.
- **Taxa de Mutação**
  As melhores taxas parecem estar na faixa de **0.5%-1%**.
- **Tamanho da População**
  Um bom tamanho para a população é cerca de **20-30**, entretanto às vezes tamanhos de 50-100 são relatados como os melhores. Alguns autores também mostram que o melhor tamanho da população depende do **tamanho da série codificada** (cromossomas). Isto significa que se você tem cromossomas com 32 bits, a população deve ser maior do que se você tivesse cromossomos com 16 bits.
- **Seleção**
  A Seleção através da **Roleta** pode ser usada, mas às vezes a Seleção por Classificação pode ser melhor.

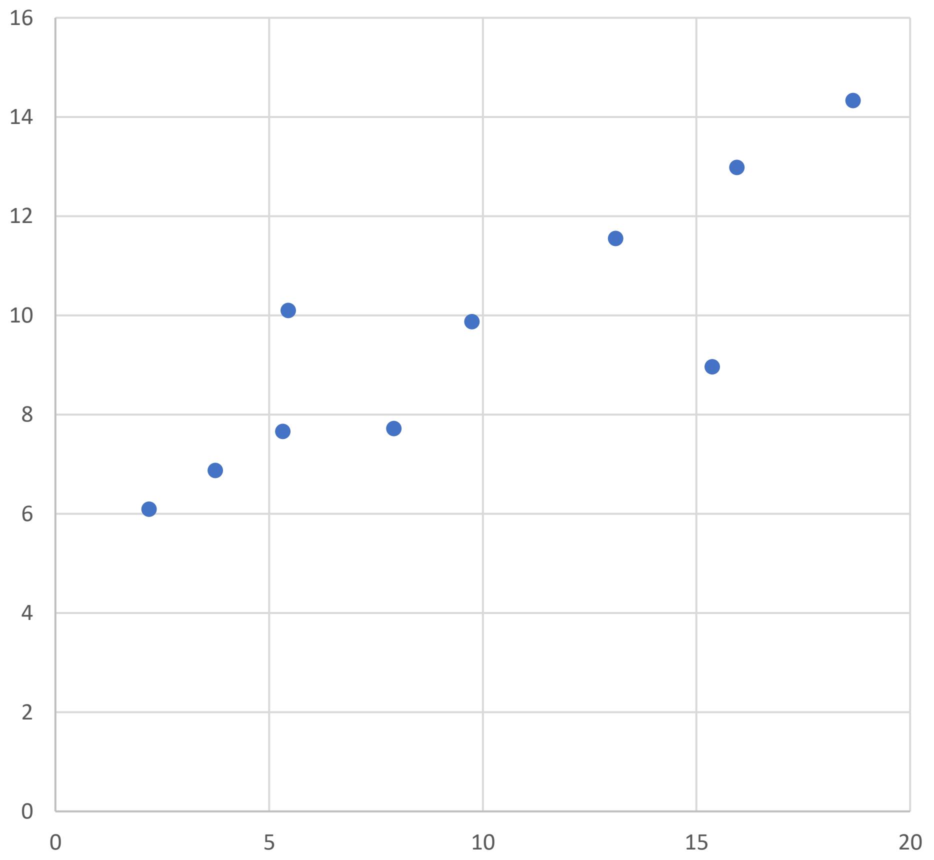
https://www.obitko.com/tutorials/genetic-algorithms/portuguese/recommendations

Um exemplo:

Dada uma série de pares de dados (X,Y) encontre os parâmetros da melhor função linear.

(dois parâmetros, a e b)

x	y
13,11	11,55
2,19	6,09
18,67	14,33
3,74	6,87
5,32	7,66
15,95	12,98
9,75	9,87
15,37	8,96
7,92	7,72
5,45	10,1



Quem pode:

Criar uma função que :

- gere uma população aleatória de 20 indivíduos?
- Calcule o fitness de um individuo?
- Ordene os indivíduos pelo fitness?
- Selecione os K-melhores indivíduos?
- Selecione 2 indivíduos (aleatoriamente) de um conjunto?
- Opere o cruzamento de dois indivíduos?
- Selecione um individuo e introduza uma mutação?

Função objetivo

Minimizar a diferença entre os valores observados e os modelados

Y=observado .. Y=[Y1, Y2,... Yn]

T=modelado ... T=[T1, T2,... Tn]

Dif=abs(T-Y)

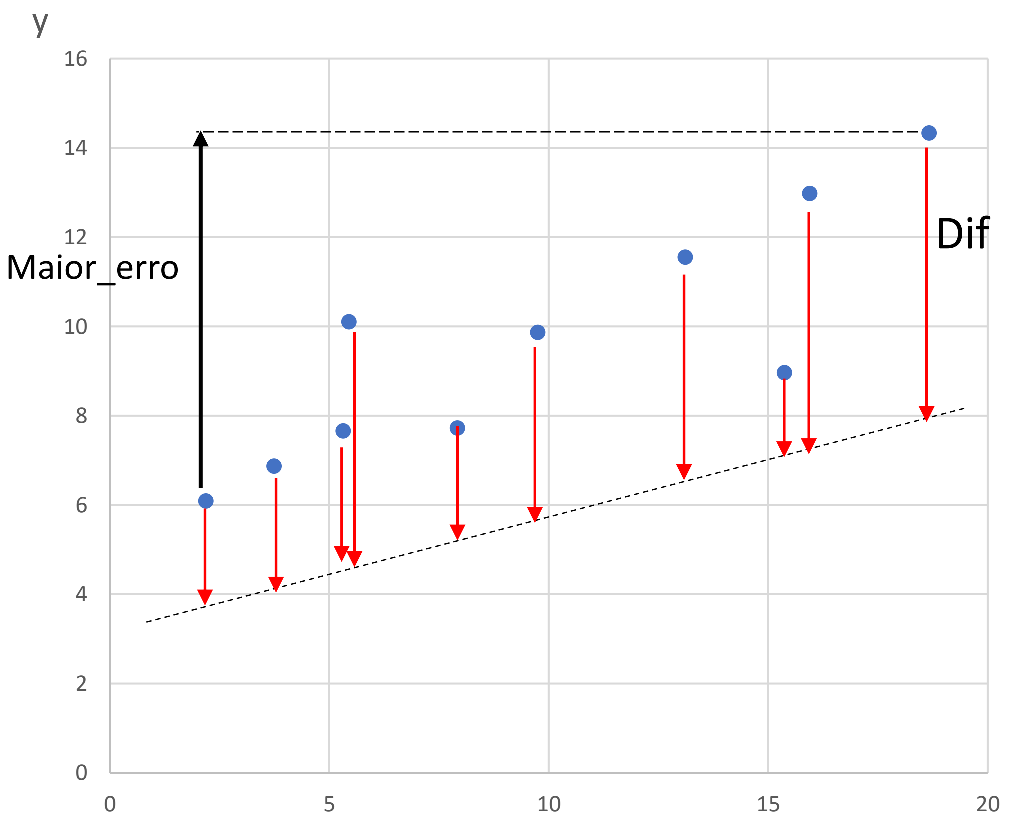
Erro_médio=media(Dif)

Maior_erro=max(Y)-min(Y)

Fitness= 1 – (erro_médio/Maior_erro)

Se Fitness <0

Fitness=0



```
X =[ 13.11, 2.19 , 18.67 , 3.74 , 5.32 , 15.95 , 9.75 , 15.37 , 7.92 , 5.45]
Y =[ 11.55 , 6.09 , 14.33 , 6.874 , 7.66 ,12.978 , 9.87 , 8.96 , 7.72, 10.1]
npop=20                  # total de indivíduos
survive=6                # quantos elementos sobrevivem
nparametros=2            # numero de parâmetros ou elementos da cadeia
pto=1                    # ponto de cruzamento
maxiteracoes=150000      # máximo de iterasses a serem realizadas
np.random.seed(1)        # iniciamos uma serie randômica

Npares=len(X)
X=np.array(X)
Y=np.array(Y)
y1 = np.zeros((Npares),dtype = float) # criação de vetor de saída vazio
```

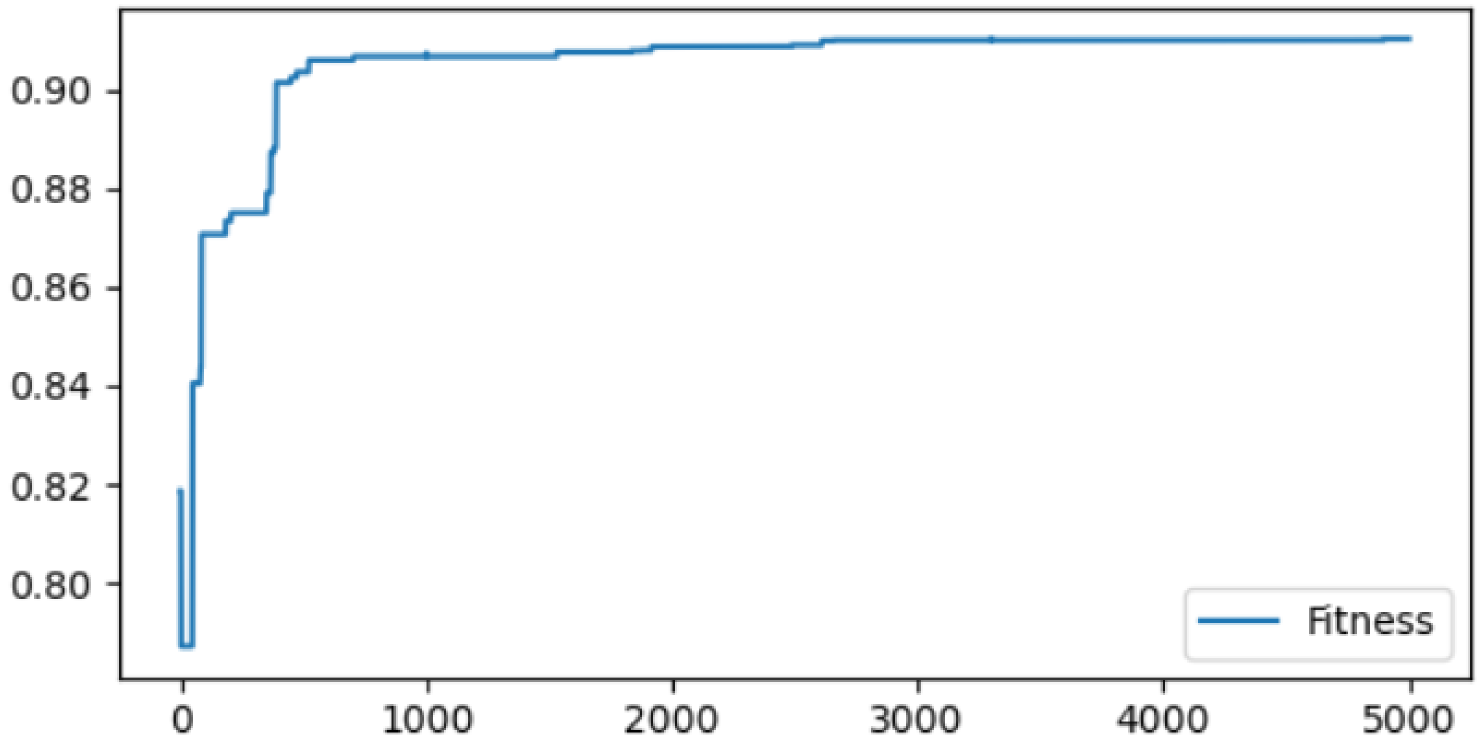
Modela Y a partir de X

```
def modelo(X, param):
    a=param[0]
    b=param[1]
    n=len(X)
    y = np.zeros((n),dtype = float) # criação de vetor de saída
    for i in range(n): # Y estimado
        y[i]= a * X[i]+b
    return y
```

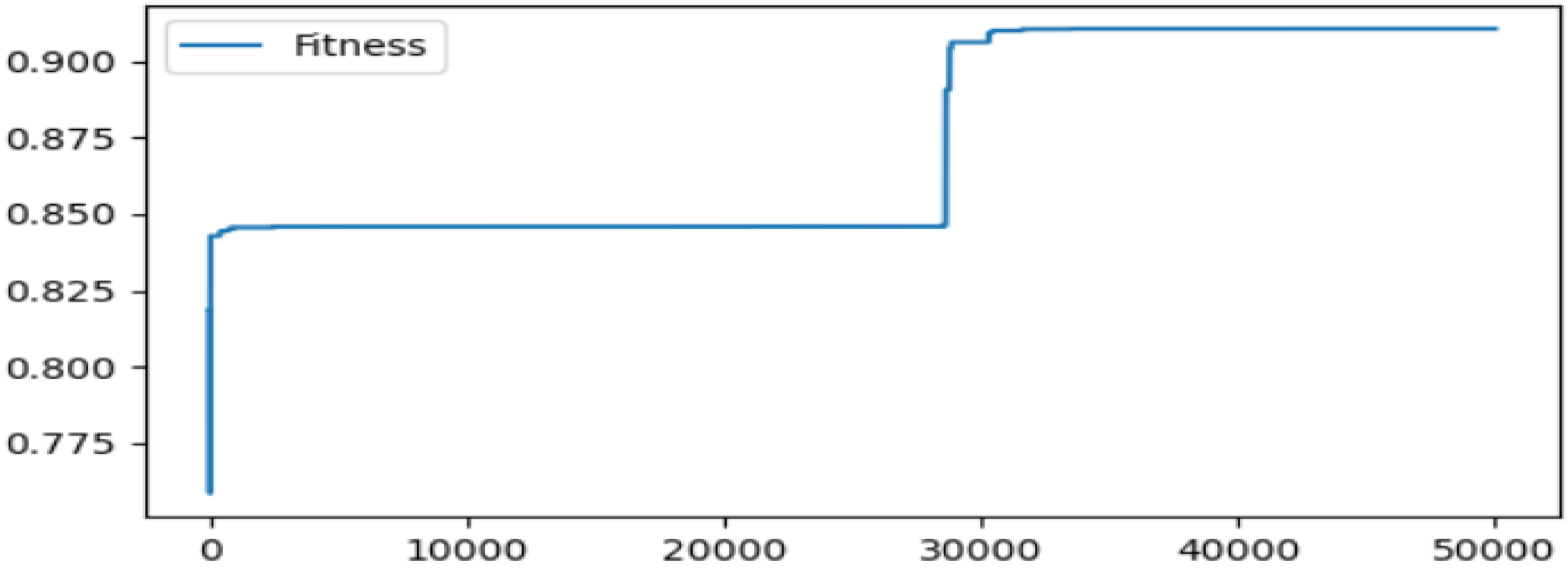
Fitness

```
# calcula fitness: dado um par de vetores X,Y, calcule a distância e F=1-d
def fitn(Y,y1):
    maxi=np.max(Y)
    mini=np.min(Y)
    maior_erro=(maxi-mini)
    dife=np.absolute(Y-y1)
    erro=np.mean(dife)
    erro=erro/maior_erro
    fitne= 1 - erro
    if fitne<0:
        fitne=0
    return fitne
```

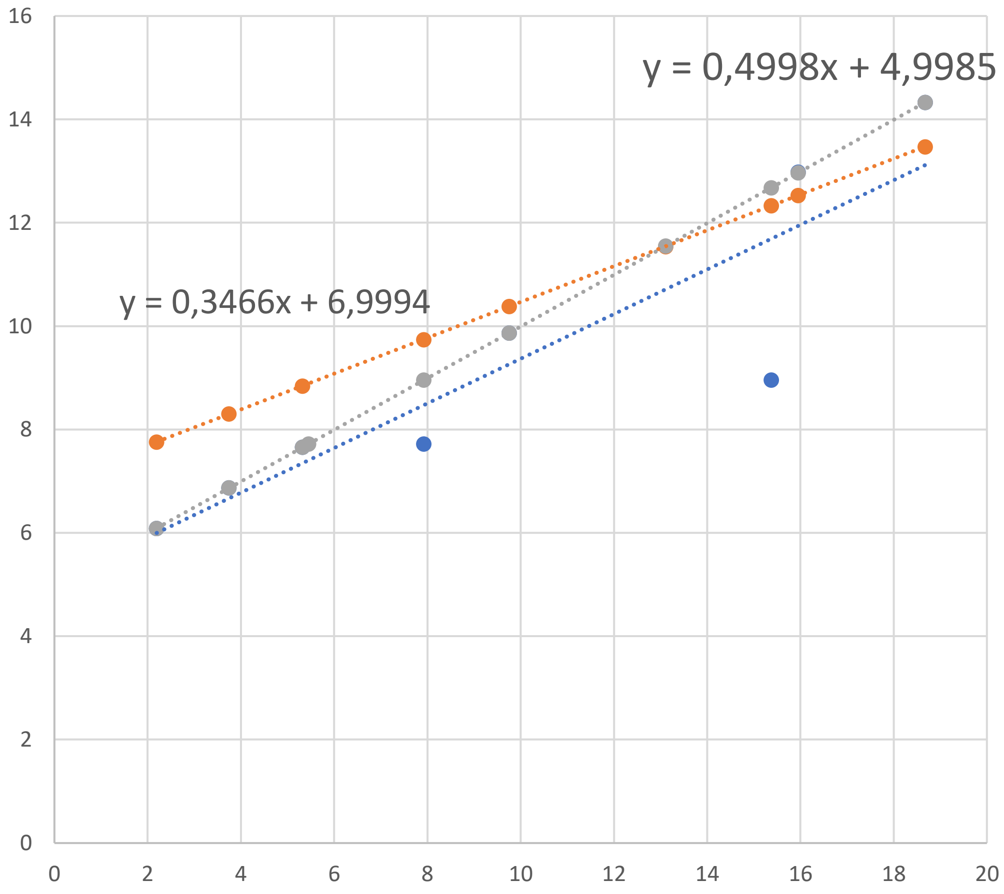
Evolução de fitness com 5000 iterações



Evolução de fitness com 50.000 iterações



Estimativas



Solução obtida pela Regressão linear:
y = 0,432x + 5,0491

A.G.
Com 5000 iterações
y = 0,3466x + 6,9994

Com 50.000 iterações
y = 0,4998x + 4,9985

Trabalho nro 1

Solucione um problema com A.G.

Elabore um relatório contendo:

a) introdução: apresentação do problema e uma hipótese a provar
b) Revisão de literatura: uso de AG no problema
c) Método: descrição dos passos para provar a hipótese
d) Experimento e análise dos resultados
e) Conclusões

Anexo o programa e os dados usados

Exemplo:

https://ainfo.cnptia.embrapa.br/digital/bitstream/item/38739/1/BP200744.pdf

Schwab et al. (1966) e Villela & Mattos (1975) indicam que a equação mais utilizada para expressar a relação intensidade-duração-tempo de retorno é:

$$I = \frac{a.T^b}{(t+c)^d} \tag{1}$$

Tabela 2. Valores de intensidade máxima anual de precipitação (mm h^{-1}) da região de Dourados, MS, com duração de 10 a 1.440 minutos.

Ano	Duração (min)											
	10	20	30	40	50	60	120	180	240	360	720	1440
1979	78,0	67,8	56,4	49,8	40,0	33,4	20,0	16,7	13,3	8,9	4,7	2,3
1980	111,6	85,8	75,2	59,7	49,7	42,0	23,4	19,1	15,8	12,8	6,7	4,2
1981	117,6	86,4	64,0	52,2	45,1	41,0	21,7	15,3	11,6	7,7	4,4	2,2
1982	132,0	102,0	88,4	79,5	75,8	65,8	37,7	25,1	18,9	14,3	9,9	5,2
1983	96,0	76,8	60,0	47,4	38,5	36,1	19,4	14,4	11,9	8,8	4,6	2,3
1984	168,0	102,6	72,4	56,7	45,7	38,2	20,4	13,8	10,4	7,4	3,7	1,9
1985	81,6	78,0	56,8	49,8	42,5	38,6	25,9	23,7	23,1	18,0	9,0	4,7
1986	100,8	87,6	78,0	67,5	63,6	58,2	41,3	27,6	20,8	13,9	7,0	3,5
1987	116,4	97,2	100,8	78,6	72,0	62,4	33,6	22,6	17,2	11,5	5,8	2,9
1988	110,4	88,2	72,8	60,0	51,1	43,6	21,9	21,1	16,8	11,4	5,7	2,8
1989	105,0	83,1	55,8	43,5	35,4	31,6	18,8	12,5	10,6	7,6	3,8	1,9
1990	66,0	53,4	38,0	32,7	27,4	23,4	17,8	14,1	11,7	8,7	5,5	3,4
1992	100,8	65,4	49,6	37,5	30,1	25,1	12,6	8,4	6,3	4,2	2,1	1,1
1993	91,2	67,2	56,4	50,1	43,6	37,4	25,3	16,9	12,7	8,4	4,2	2,1
1994	109,2	69,6	57,2	49,5	43,7	37,0	25,3	16,9	12,7	8,4	4,2	2,1
1995	93,6	67,8	73,2	61,5	49,4	41,6	22,3	16,1	12,5	8,3	4,9	2,9
1996	66,0	60,0	44,0	35,4	30,0	25,8	13,8	9,2	6,9	4,6	2,3	1,2
1997	79,2	75,0	66,8	62,1	51,6	44,6	24,2	16,1	12,1	8,1	4,0	2,0
1998	120,0	78,6	70,0	53,9	43,8	39,4	23,5	17,7	15,3	11,4	5,7	2,9
1999	138,0	114,0	112,0	90,6	97,2	93,2	53,3	36,3	27,2	18,1	9,1	4,5
2000	90,0	84,0	76,0	72,0	69,6	58,4	35,0	23,8	17,9	11,9	6,0	3,0
2001	150,0	120,0	100,0	84,0	73,7	69,6	40,5	27,5	20,6	13,8	6,9	3,4
2002	120,0	90,0	84,0	78,0	84,0	80,0	51,2	35,1	26,3	17,5	8,8	4,4
2003	63,6	53,4	52,8	48,6	39,4	33,2	17,6	11,8	8,9	5,9	3,0	1,5
2004	162,0	115,8	97,6	85,8	72,0	60,0	30,0	20,0	15,0	10,0	5,0	2,8
2005	96,1	82,3	71,6	68,6	57,0	50,6	28,1	19,2	14,4	9,6	5,8	3,4
2006	118,9	114,3	86,4	72,8	67,7	60,7	49,4	37,1	32,1	21,7	11,5	5,8
Média	106,7	83,9	71,0	60,3	53,3	47,1	27,9	19,9	15,6	10,9	5,7	3,0
CV (%)	26	22	26	26	33	36	40	38	40	40	40	41

OUTRO: MOCOCA SP

TABELA 1. Intensidade pluvial máxima em Mococa, SP, de 1970 a 1990, em mm, com diferentes durações de chuva.

Ano	Duração (minutos)									
	5	10	15	20	25	30	45	60	90	120
1970	9,0	13,6	18,8	22,8	28,0	31,4	35,8	39,2	42,0	42,6
1971	10,3	15,0	25,0	27,9	33,1	38,1	40,9	42,4	43,1	44,0
1972	10,7	17,7	22,9	26,3	31,6	32,7	40,9	45,0	48,7	53,0
1973	12,4	19,6	21,7	24,0	24,8	25,4	26,4	29,0	42,7	43,0
1974	13,0	25,6	31,9	36,1	43,4	53,0	63,5	69,3	70,9	71,0
1975	14,4	21,2	29,9	33,3	36,8	43,1	44,3	44,5	44,6	44,6
1976	12,9	18,0	22,9	28,0	30,0	32,0	36,8	37,0	41,0	41,2
1977	12,0	20,0	23,3	26,8	29,8	33,2	42,4	47,0	71,9	75,8
1978	8,2	12,2	17,7	21,7	24,8	26,7	29,6	30,3	31,7	33,3
1979	10,4	15,5	20,8	23,2	25,9	30,1	35,8	37,8	39,8	43,3
1980	13,6	22,6	32,6	35,6	37,0	37,5	38,0	39,0	48,4	53,4
1981	10,9	20,9	23,4	23,5	23,8	23,8	24,2	26,7	30,2	33,2
1982	10,3	13,2	15,6	19,9	19,9	23,3	28,0	36,6	39,8	49,0
1983	7,3	12,5	17,0	19,5	23,0	24,1	31,3	31,8	42,0	47,8
1984	6,5	9,7	13,4	19,5	23,0	25,5	33,2	44,1	47,1	51,0
1985	9,0	14,2	19,4	25,1	29,3	32,4	34,3	34,9	36,0	36,0
1986	9,2	16,2	24,0	29,0	35,0	38,9	42,7	43,5	44,7	46,6
1987	13,1	17,1	25,1	33,8	36,7	42,8	44,0	44,0	44,0	52,6
1988	10,2	16,3	21,5	24,4	27,4	26,8	33,9	36,3	43,0	46,0
1989	9,3	16,6	21,7	26,0	29,5	32,5	42,0	43,3	44,8	45,1
1990	10,3	19,5	28,0	32,8	32,8	33,0	36,8	38,6	41,4	49,1

https://ainfo.cnptia.embrapa.br/digital/bitstream/item/45114/1/RELACAO-ENTRE-INTENSIDADE-DURACAO.pdf

referencias

GALVAO, C. de O. (1999). **Sistemas inteligentes: aplicações a recursos hídricos e ciências ambientais**. Colecao ABRH de Recursos Hidricos, Volume 7. Editora UFRGS: ABRH. ISBN 8570255276, 9788570255273, 246 páginas.

ALGORITMOS GENÉTICOS.

https://www.obitko.com/tutorials/genetic-algorithms/portuguese/index.php