

UNIVERSIDAD DE GUANAJUATO

DIVSIÓN DE CIENCIAS E INGENIERÍAS

GRUPO A

CLAVE IILI0505

CATEDRÁTICO: JOSÉ DE JESÚS IBARRA SÁNCHEZ

TAREA:

“EVOLUCIÓN DIFERENCIAL”

ALUMNA:

ANA ISABEL ESQUIVEL CASTRO

FECHA DE ENTREGA: 07/11/2025



Reporte No. 6

“EVOLUCIÓN DIFERENCIAL”

1. Objetivo

- ❖ Implementar y aplicar el algoritmo de Evolución Diferencial para resolver problemas de optimización no lineal multivariada.
- ❖ Analizar el desempeño del método estocástico en la minimización de funciones objetivo difíciles y comparar su eficacia con el enfoque deterministas.

2. Código

```

1. close all
2. clear
3. clc
4. disp( '                UNIVERSIDAD DE
GUANAJUATO                ')
5. disp( '                DIVISION DE CIENCIAS E
INGENIERIAS                ')
6. disp( '                DIQEB
')
7. disp( ' ')
8. disp( ' DESCRIPCION: Este programa
optimiza un modelo multivariado no
')
9. disp( '                lineal mediante
el método de EVOLUCION DIFERENCIAL
')
10. disp( ' ')
11. disp('*****
*****')
12. disp( ' ')
13. input 'Presione ENTER para ejecutar el
programa'
14. disp( ' ')
15. disp('*****
*****')
16. disp( ' ')
17. disp('
Constantes encontradas
')
18. disp('*****
*****')
19. disp('                X1                X2                X3
X4                f_obj                ')
20. disp('*****
*****')
21.%
*****
***** %
22.% ***** MÉTODO DE
EVOLUCION DIFERENCIAL
***** %
23.%
*****
***** %
24.
25.tic
26.h = 0; % inicialización SR
27.for replicas = 1:25
28.
29.%
*****
***** %
30.% ***** DATOS CONOCIDOS DE
LA FUNCION PRUEBA ***** %
31.%
*****
***** %
32.
33.fopt = 0; % valor conocido de la
función objetivo
34.TOL = 1e-6; % tolerancia
35.
36.%
*****
***** %
37.% ***** LIMITES
DEL PROBLEMA ***** %
38.%
*****
***** %
39.
40.D = 5;
41.L = -0000;
42.H = +1e9;
43.

```

```

44.%
*****
***** %
45.% ***** PARAMETROS
DE CONTROL ***** %
46.%
*****
***** %
47.
48.SCmax = 100*D;
49.%
*****
***** %
50.% ***** VARIABLES
DEL ALGORITMO *****
%
51.%
*****
***** %
52.
53.NP = 1000;
54.GEN = 1000;
55.F = 0.5;
56.CR = 0.9;
57.%
*****
***** %
58.% ***** CREACIÓN DE
LA POBLACION ***** %
59.%
*****
***** %
60.
61.X = zeros(D,1);
62.Pop = zeros(D,NP);
63.Fit = zeros (1,NP);
64.iBest = 1;
65.r = zeros(3,1);
66.
67.% inicialización aleatoria de un
generador de números
68.
69.rand('state',sum(100*clock));
70.for j = 1:NP
71.    Pop(:,j) = L + (H-L)*rand(D,1);
72.    Fit(1,j) = fnc(Pop(:,j));
73.end
74.
75.%
*****
***** %
76.% *****
OPTIMIZACION
***** %
77.%
*****
***** %
78.
79.for g = 1:GEN
80.    for j = 1:NP
81.
82.
83.        r(1) = floor(rand()*NP) + 1;
84.        while r(1)==j
85.            r(1) = floor(rand()*NP) +
1;
86.        end
87.        r(2) = floor(rand()*NP) + 1;
88.        while (r(2)==r(1))||(r(2)==j)
89.            r(2) = floor(rand()*NP) +
1;
90.        end
91.        r(3) = floor(rand()*NP) + 1;
92.        while
(r(3)==r(2))||(r(3)==r(1))||(r(3)==j)
93.            r(3) = floor(rand()*NP) +
1;
94.        end
95.
96.    % crear una nueva población en la
cual el último parámetro es cambiado
97.
98.        Rnd = floor(rand()*D) + 1;
99.        for i = 1:D
100.            if ( rand()<CR ) ||
(Rnd==i )
101.                X(i) =
Pop(i,r(3)) + F * (Pop(i,r(1)) -
Pop(i,r(2)));
102.            else
103.                X(i) =
Pop(i,j);
104.            end
105.        end
106.
107.        % verificar que las variables
se encuentren dentro de los límites
108.
109.        for i = 1:D
110.            if
(X(i)<L)||X(i)>H)
111.                X(i) = L + (H-
L)*rand();
112.            end
113.        end
114.
115.        % seleccionar el mejor de los
individuos y calcular fobj
116.
117.        f = fnc(X);    % si el
valor prueba es mejor o igual que el
actual
118.
119.            if f <= Fit(j)

```

```

120.      Pop(:,j) = X; %
      reemplazar actual por el valor prueba
121.      Fit(j) = f;
122.      % si prueba es
      mejor que el actual
123.
124.      if f <= Fit(iBest)
125.          iBest = j;
126.      end
127.      end
128.      end
129.      end
130.
131.      f = Fit(iBest);
132.      X = Pop(:,iBest);
133.
134.      %
      *****
      ***** %
135.      % *****
      RESULTADOS
      ***** %
136.      %
      *****
      ***** %
137.
138.      fprintf (' %f %f %f %f
      %f %f\n',X(1),X(2),X(3),X(4),X(5),f);
139.
140.      %
      *****
      ***** %
141.      % *****
      CALCULO DEL SR
      ***** %
142.      %
      *****
      ***** %
143.
144.      if abs(f - fopt)< TOL
145.          h = h+1;
146.      end
147.
148.      fun_obj(replicas) = f;
149.      [f_best,UBIB] = min(fun_obj);
150.
151.      X_opt(1,replicas) = X(1);
152.      X_opt(2,replicas) = X(2);
153.      X_opt(3,replicas) = X(3);
154.      X_opt(4,replicas) = X(4);
155.      X_opt(5,replicas) = X(5);
156.      end
157.
158.      X_best = [X_opt(1,UBIB)
      X_opt(2,UBIB)];
159.
160.      SR = (h/replicas)*100;

```

```

161.      Fobj = (sum(fun_obj))/replicas;
162.
163.      disp('*****
      *****')
164.      disp(' ')
165.
166.      disp([' El porcentaje de SR es
      =', num2str(SR)]);
167.      disp(' ')
168.      disp([' El valor promedio de la
      Fobj es =', num2str(Fobj)]);
169.      disp(' ')
170.      disp([' El mejor valor de la
      Fobj encontrado es =',
      num2str(f_best)]);
171.      disp(' ')
172.      disp([' El ID del mejor valor
      de la Fobj es =', num2str(UBIB)]);
173.      disp(' ')
174.      disp(' El vector solucion del
      mejor valor de la Fobj es =')
175.      disp(' ')
176.      fprintf (' %f',X_best);
177.      disp(' ')
178.      disp('*****
      *****');
179.      disp(' ')
180.
181.      toc
182.      disp(' ')
183.
184.      %
      *****
      ***** %

```

function f=fnc(X)

```

%
*****
***** %

% time = [0.5 1.0 1.5];
% y_obs = [0.263 0.455 0.548];
%
% n = length(time);
%
% suma = 0;
%
% for j = 1:n
%     func = (y_obs(j) - ((X(1)/(X(1)-X(2))))
% * (exp(-X(2)*time(j)) - exp(-X(1)*time(j)))
% )^2;
%     suma = func + suma;
% end
%
% f = suma;

```

```
%
*****
***** %

k1    = [1 2 1 2 0.1];
k2    = [1 1 2 2 0];
y_obs = [0.126 0.219 0.076 0.126 0.186];

n = length(k1);

suma = 0;

for j = 1:n
    func = (y_obs(j) - (X(1)*k1(j)/
(1+X(2)*k1(j) + X(3)*k2(j)) ) )^2;

    suma = func + suma;
end

f = suma;

%
*****
***** %

% k1    = [2.0 6.0 9.0 2.5 4.5 9.5 8.0 4.0
3.0 7.0 6.5];
% k2    = [36.0 8.0 3.0 6.25 7.84 1.44 4.0
7.0 9.0 2.0 5.0];
% y_obs = [46.5 591 1285 36.8 241 1075 1024
151 80 485 632];
%
% n = length(k1);
%
% suma = 0;
%
% for j = 1:n
%     func = (y_obs(j) - (X(1)*
(k1(j))^X(2) )*(k2(j))^X(3) ) )^2;
%     suma = func + suma;
% end
%
% f = suma;

%
*****
***** %

% a = 0.20;
% cc = 12.50;
% ci = 0.50;
% cx = 0.90;
% ir = 0.10;
% n = 2;
% p = 7000;
%
```

```
% c1 = (2.09e4)*(X(2)^-0.3017)/360;
% c2 =
((1.064e6)*a*(X(2)^0.4925))/(52.47*X(1)*360)
;
% c3 = (((4.242e4)*a*X(2)^0.7952) +
1.813*ir*p*(n*X(2) +
1.2*X(1))^0.861)/(52.47*X(1)*360);
% c4 =
((4.25e3)*a*(n*X(2)+1.2*X(1)))/(52.47*X(1)*3
60);
% c5 = ((5.042e3)*X(1)^-0.1899)/360;
% c6 = (0.1049/360)*X(1)^0.671;
%
% f = cc + ci + cx + c1 + c2 + c3 + c4 + c5
+ c6;

%
*****
***** %
```

3. Resultados

a) Función 1

Estimate the values of the parameters k_1 and k_2 by minimizing the sum of the squares of the deviations

$$\phi = \sum_{i=1}^n (y_{\text{observed}} - y_{\text{predicted}})_i^2$$

where

$$y_{\text{predicted}} = \frac{k_1}{k_1 - k_2} (e^{-k_2 t} - e^{-k_1 t})$$

for the following data:

t	y_{observed}
0.5	0.263
1.0	0.455
1.5	0.548

El porcentaje de SR es =0

El valor promedio de la Fobj es =0.5765

El mejor valor de la Fobj encontrado es =0.5765

El ID del mejor valor de la Fobj es =1

El vector solucion del mejor valor de la Fobj es =

428454954.533228 477885232.670503

b) Función 2

Repeat Problem 6.41 for the following model and d

$$y = \frac{k_1 x_1}{1 + k_2 x_1 + k_3 x_2}$$

y_{observed}	x_1	x_2
0.126	1	1
0.219	2	1
0.076	1	2
0.126	2	2
0.186	0.1	0

```
El porcentaje de SR es =0
El valor promedio de la Fobj es =4.3553e-05
El mejor valor de la Fobj encontrado es =4.3553e-05
El ID del mejor valor de la Fobj es =8
El vector solucion del mejor valor de la Fobj es =
2.442770 3.131505
```

c) Función 3

Estimate the coefficients in the correlation

$$y = ax_1^b x_2^c$$

from the following experimental data by minimizing the sum of the square of the deviations between the experimental and predicted values of y .

y_{expt}	x_1	x_2
46.5	2.0	36.0
591	6.0	8.0
1285	9.0	3.0
36.8	2.5	6.25
241	4.5	7.84
1075	9.5	1.44
1024	8.0	4.0
151	4.0	7.0
80	3.0	9.0
485	7.0	2.0
632	6.5	5.0

d) Función 4

$$c = c_r + c_i + c_s + \frac{2.09 \times 10^4 t^{-0.3017}}{360} + \frac{1.064 \times 10^6 a r^{0.4925}}{52.47 q(360)} + \frac{4.242 \times 10^4 a r^{0.7952} + 1.813 p (a r + 1.2 q)^{0.861}}{52.47 q(360)} + \frac{4.25 \times 10^3 a (a r + 1.2 q)}{52.47 q(360)} + \frac{5.042 \times 10^3 q^{-0.1899}}{360} + \frac{0.1049 q^{0.671}}{360}$$

where a = annual fixed charges, fraction (0.20)
 c_r = crude oil price, \$/kL (12.50)
 c_i = insurance cost, \$/kL (0.50)
 c_s = customs cost, \$/kL (0.90)
 i = interest rate (0.10)
 n = number of ports (2)
 p = land price, \$/m² (7000)
 q = refinery capacity, bbl/day
 t = tanker size, kL

Given the values indicated in parentheses, use a computer code to compute the minimum cost of oil and the optimum tanker size t and refinery size q by Newton's method and the quasi-Newton method (note that 1 kL = 6.29 bbl).
 (The answers in the reference were

$$t = 427,000 \text{ dwt} \approx 485,000 \text{ kL} \\ q = 185,000 \text{ bbl/day}$$

```
El porcentaje de SR es =0
El valor promedio de la Fobj es =17.8849
El mejor valor de la Fobj encontrado es =17.8849
El ID del mejor valor de la Fobj es =1
El vector solucion del mejor valor de la Fobj es =
179845.379426 446967.519025
```

En conclusión, el método determinista puede asumir que un conjunto específico de condiciones conducirá a un resultado exacto sin variación, mientras que el modelo estocástico integra la aleatoriedad y por lo tanto, tiene la probabilidad de que ocurran diferentes resultados, además, a menudo se recurren a modelos estocásticos cuando no se pueden formular leyes matemáticas precisas. Por estas razones, se considera que el método determinista es una mejor opción para la búsqueda de soluciones para funciones multivariantes, y cabe destacar que, para la mayoría de los problemas de optimización en esta carrera, pueden tener características realistas como la no linealidad, que sean multimodales o con restricciones complejas,

```
El porcentaje de SR es =0
El valor promedio de la Fobj es =Inf
El mejor valor de la Fobj encontrado es =Inf
El ID del mejor valor de la Fobj es =1
El vector solucion del mejor valor de la Fobj es =
578989347.121385 390496740.672912
```

entre otros, así que el enfoque de la evolución

4. Conclusión

diferencial determinista es más acercada o aproximada
a las soluciones reales.
