Algoritmi Genetici: 1/0 knapsack problem

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID Obj | A | B | C | D | E | F | G | H | Capacitate  Rucsac |
| Val | 10 | 12 | 5 | 14 | 9 | 7 | 15 | 10 |
| Weight | 7 | 9 | 2 | 10 | 10 | 6 | 12 | 12 | 30 |

Cum arata o solute potential optima pentru aceasta problema?

Un vector binar X de lungime *n* (numarul de obiecte). Unde X[i]=1⇔ obiectul *i* este selectat ca fiind inclus in rucsac.

Ex:

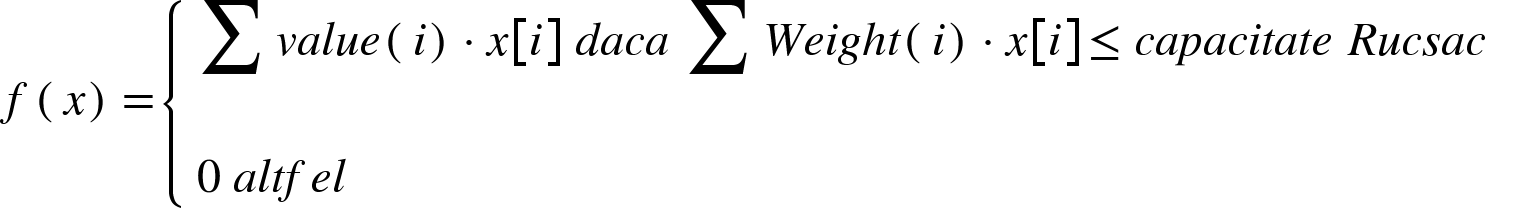
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |

In cazul unui algoritm genetic o astfel de reprezentare poate fi numita cromozom/individ. .

Acum ca avem clarificat cum ar trebui sa arate un cromozom, trecem la definirea functiei de fitness:

f: 2n -> R+

Cum definim functia de fitness in cazul problmei noastre?



Pentru cromozomul de mai sus, avem fitness-ul: val(A)+val(C)+val(F)+val(G)=37

Dar pentru cromozomul:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |

fitness(1,0,0,1,0,1,1,0) = 0, deoarece suma greutatilor obiectelor selectate depaseste capacitatea rucsacului.

Generam poplatia de indivizi (generatia 0)

Geneam cromozomi cu valori aleatoare (uniform distribuite).

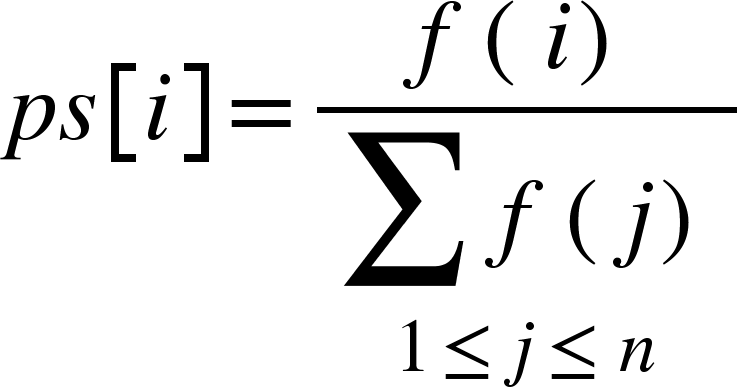
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Populatie: | | | | | | | | Fitness |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 31 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| **0** | **0** | **1** | **1** | **0** | **1** | **1** | **0** | **41** |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 19 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 29 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 32 |

Odata ce avem generatia 0, trebuie sa scontruim urmatoarea generatie pe baza celei curente:

-criteriul elitist: cel mai bun individ trece mai departe (direct,nemodificat) in generatia urmatoare. Facem asta pentru a ne asigura ca elementul elitist din generatia curenta va fi mereu cel putin la fel de bun ca elementul elitist din oricare generatie precedenta.

-criteriu de selectie prin metoda ruletei:

asociem o probabilitate de selectie (numita **ps[i]** - probabilitatea de selectie a lui *i*) fiecarui individ



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| i | 1 | 2 | 3 | 4 | 5 | 6 |
| ps(i) | 0.203 | 0 | 0.267 | 0.125 | 0.184 | 0.221(?) |
| interval de selectie | [0-0.203) | - | [0.203-0.470) | [0.47-0.595) | [0.595-0.779) | [0.779-1) |

Trebuie sa selectam 5 indivizi (corespunzand celor 5 locuri ramase libere pentru generatia urmatoare)

S-au generat aleator, uniform distribuit pe intervalul [0,1) urmatoarele valori:  
0.754 0.384 0.125 0.083 0.839

s-a selectat individul 5  
s-a selectat individul 3

s-a selectat individul 1

s-a selectat individul 1

s-a selectat individul 6

avem multisetul de indivizi selectati: 5,3,1,1,6

dintre acestia trebuie sa selectam cativa pentru procesul de crossing over;

presupunem ca avem o probabilitate de selectie pentru crossing over de 0.35

generam 5 variabile aleatoare corespunzatoare fiecruia dintre indivizii selectati:

0.082 0.326 0.059 0.365 0.512

Indivizii selectati pt crossing over sunt:

5: **[00110001]**

3: **[00110110]**

1: **[11001000]**

Alegem un punct de rupere (ex: 3)

**[001|10001]**

**[001|10110]**

**[110|01000]**

Obtinem:

**[00110110]**

**[00101000]**

**[11010001]**

**Adaugam si elementele care nu au fost selectate pentru crossing over (1,6) si obtinem:**

**[00110110]**

**[00101000]**

**[11010001]**

**[11001000]**

**[10000110]**

**Pe aceasta populatie intermediara trebuie sa aplicam operatorul de mutatie.**

**Acesta presupune sa:**

* **(in cazul mutatiei rare) iteram prin multimea de indivizi. Fiecare individ avand o mica sansa (ex: 0.02) de a fi selectat. Pentru fiecare cromozom selectat, alegem o gena in mod aleator si ii schimbam valoarea cu complementul ei.**
* **(in cazul mutatiei “regular”) iterez prin genele tuturor cromozimlor. Exista o foarte mica probabilitate (<0.01) ca o astfel de gena sa isi schimbe valoarea cu complementul sau**

**Dupa mutatie populatia intermediara va arata:  
[00110110]**

**[00101100]**

**[11010001]**

**[11001000]**

**[11000110]**

**Acesti noi indivizi se vor alatura individului elitist pentru a forma noua generatie  
[00110110]**

**[00110110]**

**[00101100]**

**[11010001]**

**[11001000]**

**[11000110]**

**repet pasii de mai sus pentru noua populatie (caluclez fitnessul fiecarui individ, selectez individul optim - il trec mai departe impplicit, selectez o multime de indivizi folosind criteriul ruletei, aplica operatorii genetici de crossing over si mutatie).**

**Pana cand repet?**

**Pana cand se indeplineste macar unul dintre cele 3 criterii:  
 - fie ajungem la un numar prestabilit de iteratii (ex: fixam limita maxima ca fiind 1000 de generatii)**

* **fie timp de un numar de generatii consecutive elementul elitist nu se schimba (ex: timp de 10 generatii, cea mai buna solutie va fi mereu aceiasi)**
* **fie am ajuns la o solutie suficient de buna (ex am gasit o solutie cu valoare >45 - un prag impus prestabilit)**

**Odata oprit algoritmul, solutia noastra va fi cea codificata de catre elemntul cel mai fit din ultima generatie.**