

Tema1

Horeanga Bogdan gr.332

5 Martie 2021

Algoritmii:

Pentru **Random Search** in problema rucsacului:

S-a folosit o functie pentru generarea unui vector de lungime n(primit ca parametru) cu valorile 0/1, unde 1 reprezinta daca elementul de pe pozitia respectiva este adaugat si 0 daca nu.

O functie de verificare, primind ca parametrii un vector de greutati, unul de pozitii(care a fost generat random cu functia de mai sus) si greutatea ghiozdanului. Aceasta functie calculeaza greutatea obiectelor $suma = suma + pozitii[i] * greutati[i]$. Daca suma este mai mica sau egala decat greutatea ghiozdanului se returneaza suma si in caz contrar se returneaza -1. Si o functie care determina k solutii, primeste ca parametrii k, un vector de greutati, greutatea ghiozdanului, un vector de valori, n(lungimea vect de pozitii generate), numele fisierului in care se vor scrie sol.

1. Cat timp nu s-au generat k solutii care incape in ghiozdan se genereaza un vector de pozitii
2. Daca functia de verificare da o valoare != -1 atunci se determina valoarea ghiozdanului
3. Se adauga in fisier.

Pentru **Random Hill Climbing(RHC)**

Am folosit o functie care genereaza valori random valide, care primeste ca parametrii vectorul de greutati, n(lungimea vect generat) si greutatea ghiozdanului:

```
while True:
```

```
    pozitii=generareRandom(n)
```

```
    if verificareGhiozdan(greutati, greutateGhiozdan, pozitii) != -1 and verificareGhiozdan(greutati, greutateGhiozdan, pozitii) != 0:
```

```
        break
```

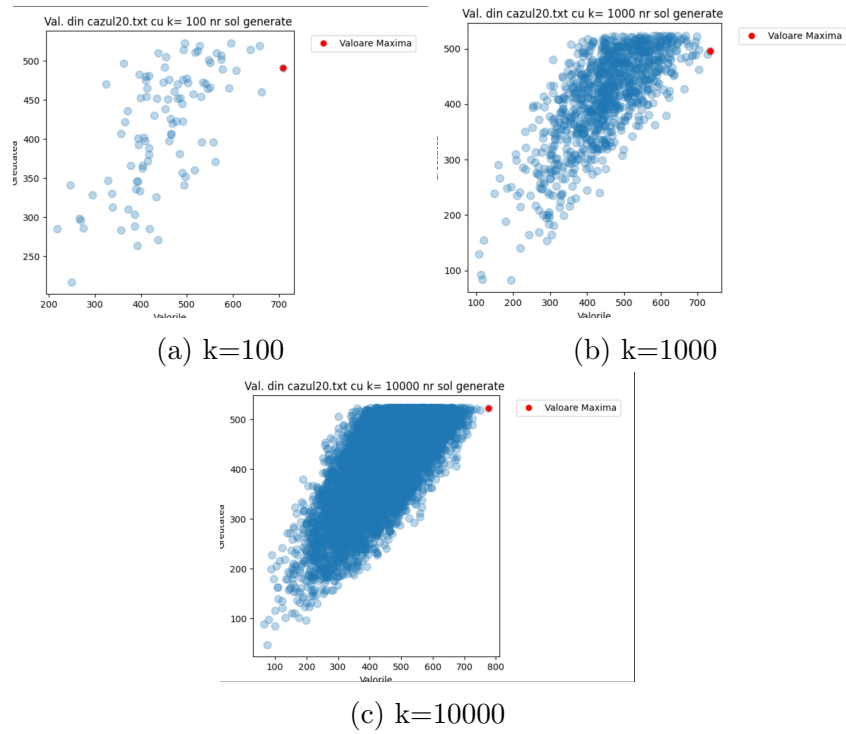
```
    else:
```

```
        continue
```

```
    return pozitii
```

O functie care primeste ca parametrii nr de incercari, pct generat random(punctC) si valid(vector), vectorul de greutate si greutatea ghiozdanului

1. Se initializeaza un inc cu 0 care reprezinta nr de incercari (de cate ori nu s-a ajuns la o val mai buna) si o copie a punctului generat random
2. se genereaza un numar de la 0 la lungimea vectorului punctC
3. Daca elementul vectorului de pe pozitia pct generat la 2 este 1 atunci inc creste cu 1
4. Alt fel se modifica valoarea din zero in in 1 la copia facuta
5. Se verifica daca pozitii are o greutate mai mica sau egala cu greutatea ghiozdanului si atunci punctulC devine copia modificata.
6. alt fel, punctC[vecin]= si inc= inc+1
7. daca inc j= int(incercari) go to 2
8. return punctC



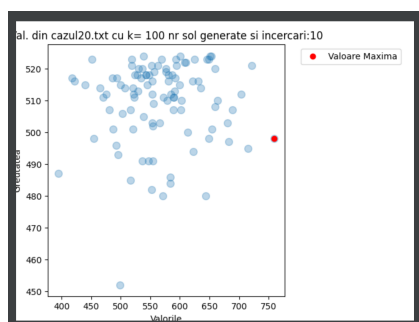
Cele trei figuri de mai sus reprezinta random search pentru problema ghiozdanului in prima istanta primita de la laborator(rucsac-20.txt). Punctul rosu din fiecare imagine reprezinta valoarea maxima pe care o poate atinge in aceste incercari.

(a) valoarea maxima se atinge in **708** si are o greutate de 491. Valoarea medie a testelor efectuate in acest caz este de 452.32 cu o calitate 1.565263530244075

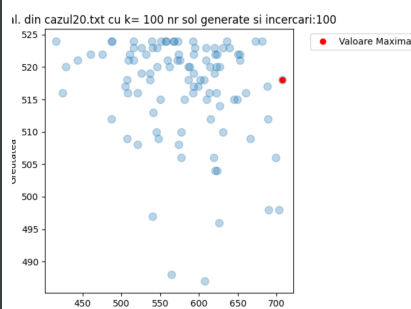
(b) valoarea maxima este **733** cu o greutate de 496. Cu o medie: 454.072 si o calitate:1.6142814355432618.

(c) valoarea maxima este **776** cu o greutate 522. Valoarea medie 452.6428 si o calitate:1.7143761040714665

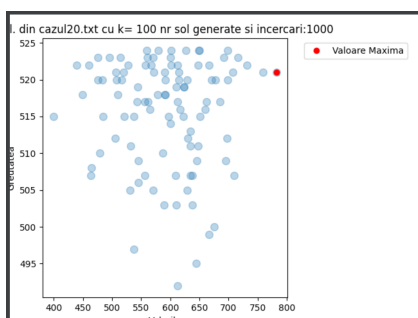
Din datele prezentate mai sus se observa ca atunci cand sunt facute mai multe incercari(teste) se reuseste gasirea unei solutii mai bune. De asemenea se observ ca valoarea medie nu depinde de foarte mult de numarul de incercari. Aceasta(media) nu are schimbari foarte mari atunci cand creste numarul de incercari. Iar diferenta dintre valoarea maxima si media valorilor este mai mare decat la RHC (pe care o sa il vedem mai jos).



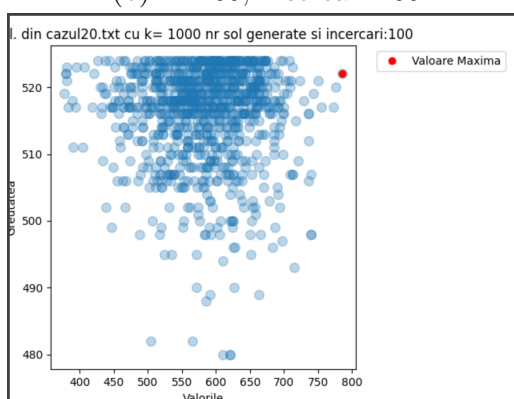
(a) $k=100$, incercari 10



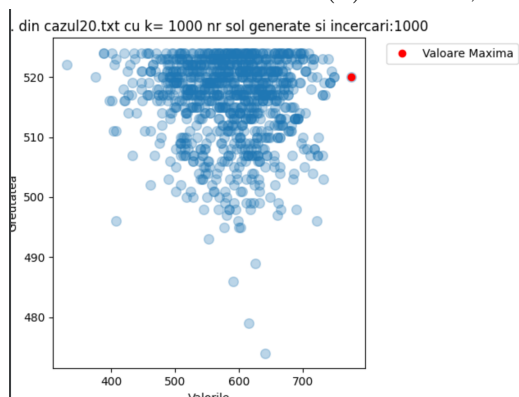
(b) $k=100$, incercari 100



(c) $k=100$, incercari 1000



(d) $k=1000$, incercari 100



(e) $k=1000$, incercari 1000

Cele 5 figuri de mai sus reprezinta **random hill climbing** pentru problema ghiozdanului in prima istanta de la laborator(rucsac-20.txt). Punctu rosu reprezinta valoarea maxima pe care o poate atinge in aceste incercari. (a) valoarea maxima se atinge in **759** si are o greutate de 498. Valoarea medie

a testeleor efectuate in acest caz este de **566.59** cu o calitate **1.339593003759332**

(b) valoarea maxima este **707** cu o greutate de 517. Cu o medie: **579.02** si o calitate:**1.2210286345894787**.

(c) valoarea maxima este **763** cu o greutate 520. Valoarea medie **578.519** si o calitate:**1.3188849458704035**

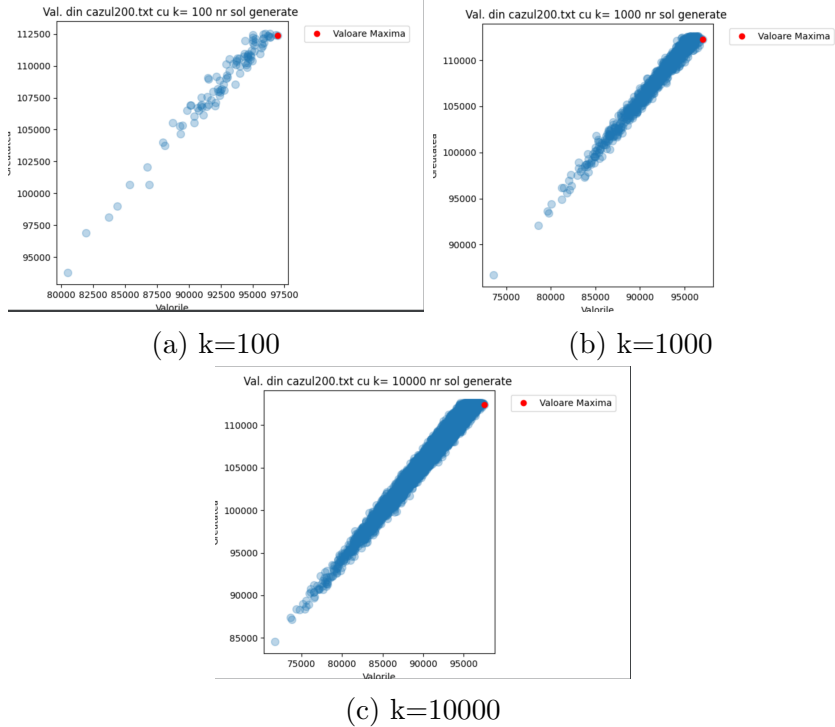
(d) valoarea maxima este **785** cu o greutate 522. Valoarea medie **582.791** si o calitate:**1.3469665797858923**

(e) valoarea maxima este **776** cu o greutate 520. Valoarea medie **586.167** si o calitate:**1.3238548058829651**

Primele 3 imagini sunt reprezentarile cazului in care nr solutilor este 100, unde numarul de incercari este 10,100 si 1000. Comparativ cu **random search** se observa ca media valorilor este mai mare(evident), avand si o calitate a solutilor mai buna. Se observa ca atunci cand numarul de solutii generate este mai mare, creste si media, implicit si calitatea solutilor.

Comparand imaginea (d) cu (e), unde in imaginea (e) numarul de incercari este mai mare, se observa cum media valorilor se muta in spre dreapta(creste media).

Cele trei figuri de mai jos reprezinta random search pentru problema ghiozdanului in prima istanta primita de la laborator(rucsac-200.txt). Punctul rosu din fiecare imagine reprezinta valoarea maxima pe care o poate atinge in aceste incercari.



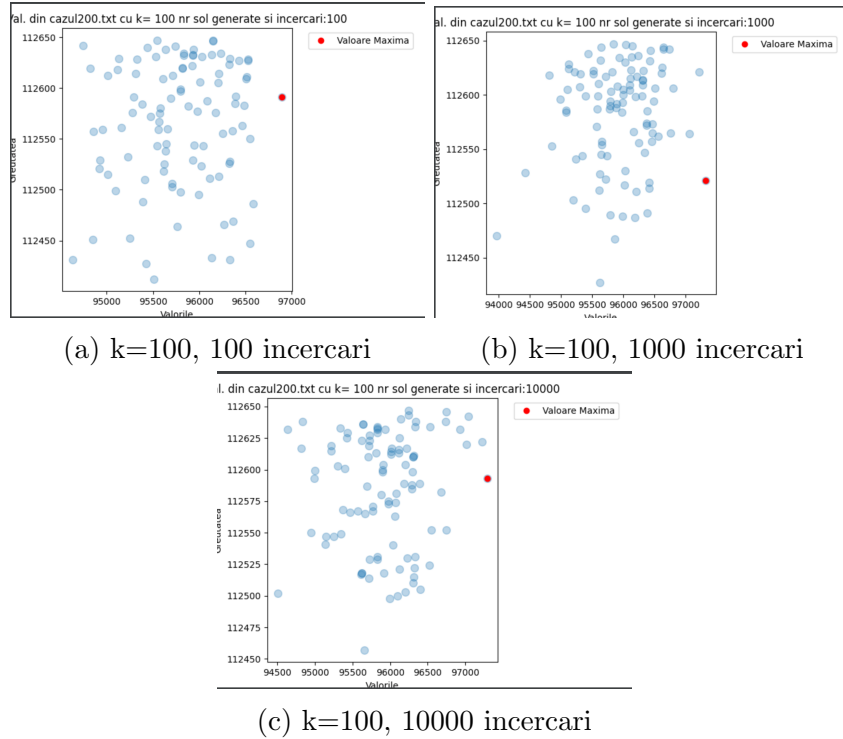
In figura (a) Cea mai mare valoare este: 96960 cu o greutate 112360 Valoarea medie este: 92694.49 Calitatea solutiei este: 1.0460168668062146

In figura (b) Cea mai mare valoare este: 97040 cu o greutate 112240 Valoarea medie este: 92256.772 Calitatea solutiei este: 1.0518469039866256

In figura (c) Cea mai mare valoare este: 97519 cu o greutate 112419 Valoarea medie este: 92163.0805 Calitatea solutiei este: 1.0581135034868978

In aceasta instanta, cand valorile se iau din fisierul rucsac-200.txt, se observa aceeaasi tendinta ca atunci cand creste numarul de solutii generate este mai mare, valoarea maxima creste. Comparativ cu cazul in care valorile se iau din fisierul: **rucsac-20.txt** se observa ca valoarea medie este mult mai apropiata de valoarea maxima. Si calitatea solutiei este mai buna.

In cele trei figuri de mai jos sunt reprezentarile vizuale pentru **Random Hill Climbing** pentru instanta in care valorile se iau din fisierul **rucsac-200.txt**. Ca in toate imaginile de mai sus punctul rosu reprezinta valoarea maxima atinsa.



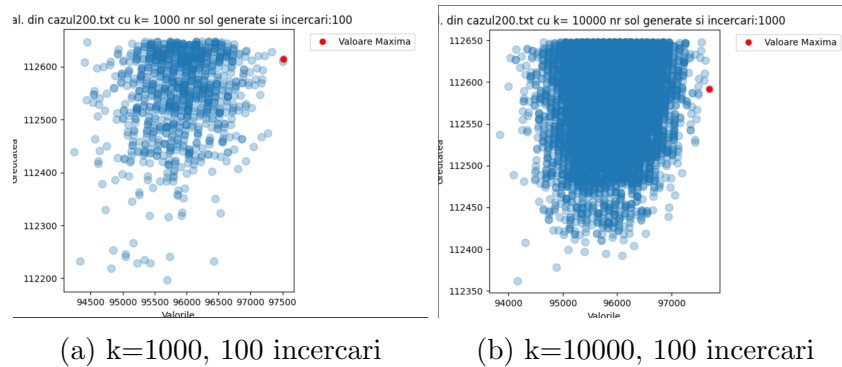
In figura (a) Cea mai mare valoare este: 96891, cu o greutate 112591 Valoarea medie este: 95793.57 Calitatea solutiei este: 1.0114561969034037 Numarul de incercari maxim admise: 100

In figura (b) Cea mai mare valoare este: 97321, cu o greutate 112521 Valoarea medie este: 95931.01 Calitatea solutiei este: 1.0144894753010523 Numarul de incercari maxim admise: 1000

In figura (c) Cea mai mare valoare este: 97293, cu o greutate 112593 Valoarea medie este: 95922.69 Calitatea solutiei este: 1.0142855668455504 Numarul de incercari maxim admise: 10000

Dupa cum se obseva calitatea solutiilor este mai buna decat la **Random Search**, dar daca numarul de incercari creste foarte mult nu se schimba foarte mult calitatea solutiei. In schimb valoarea maxima creste.

In figura (a) Cea mai mare valoare este: 97444 Valoarea medie este:



95888.831 Calitatea solutiei este: 1.0162184582269023 Numarul de incercari maxim admise: 100

In figura (b) Cea mai mare valoare este: 97692 Valoarea medie este: 95889.4571 Calitatea solutiei este: 1.0187981343780077 Numarul de incercari maxim admise: 1000

15	RHC 20 =10 rulari										
16											
17	k=10	incercari 100	best=731	avg=685.7	quality=1.066063876	time=0.125167846679687					
18											
19	k=100	incercari 100	best=766	avg=733.3	quality=1.044592936	time=1.4781169891357422					
20											
21	k=1000	incercari 100	best=787	avg=776.1	quality=1.01405	time=14.659889698028564					
22											
23	RS20 =10 rulari										
24											
25	k=10		best=710	avg=618.3	quality=1.14831	time=9.82221269607544					
26											
27	k=100		best=752	avg=684.4	quality=1.095	time=4.3915181159973145					
28											
29	k=1000		best=771	avg=724.7	quality=1.063	time=4.328670978546143					
30											
31	RHC200 =10 rulari										
32											
33	k=10	incercari 100	best=9732	avg=96711.4	quality=1.0063	time=0.886854887008667					
34											
35	k=100	incercari 100	best=9771	avg=97201.3	quality=1.0053	time=8.555453062057495					
36											
37	k=1000	incercari 100	best=9782	avg=97504.9	quality=1.0032	time=83.61814188957214					
38											
39	RS 200 =10 rulari										
40											
41	k=10		best=9647	avg=95808.1	quality=1.00703	time=3.2866439819335938					
42											
43	k=100		best=9717	avg=96763.5	quality=1.00425	time=6.474422216415405					
44											
45	k=1000		best=9769	avg=97322.7	quality=1.003784	time=26.598511219024658					

(a) Table 10 rularii

In imaginea de mai sus se observa 4 teste in care s-au executat 10 rulari pentru fiecare. In RHC(RANDOM HILL CLIMBING) se observa cum creste valoarea celei mai bune solutii atunci cand creste si numarul de solutii generate.

Comparand RHC20 cu RS20(RANDOM SEARCH) se observa cum RHC obtine valori mai bune decat Random search avand si o calitate a solutiilor mai buna.