



Klasterovanje tačaka korišćenjem genetskog algoritma

Projekat u okviru kursa
Računarska inteligencija

Opis problema

- Za dati skup tačaka $\text{points}(n\text{-dimenzionog prostora})$ i zadati broj klastera K potrebno je pronaći adekvatne centre klastera kojim će se početni skup tačaka podeliti na odgovarajuće klastere.

Implementacija



Obrada ulaznih podataka


Unos ulaznih podataka je omogućen na dva načina:

- 1) Slučajnim generisanjem tačaka iz odgovarajućih intervala
- 2) Čitanjem fajlova sa zadate putanje. Fajlovi su formatirani na odgovarajući način (svaki red predstavlja jednu tačku u n -dimenzionom prostoru)



Implementacija jedinke


- Svaka jedinka u genetskom algoritmu biće predstavljena kao lista dimenzije K gde svaki od elemenata liste je lista dimenzije n (n je dimenzija prostora u kome vršimo klasterovanje) i predstavlja centar za jedan od K klastera.
- Inicijalna populacija se generiše slučajnim izborom p (p -veličina populacije) tačaka iz skupa points koji je zadat na početku.

- 
- **Fitness** funkcija je zadana kao jedan kroz suma kvadratnih rastojanja(SSE) od tačaka od odgovarajućeg centra klastera kome tačka pripada.
 - Rastojanja se računaju euklidski.
 - $SSE = \sum_{i=1}^n (x_i - x^-)^2$
 - $Fitness = \frac{1}{SSE}$



Implementacija selekcije, ukrštanja i mutacije

- **Selekcija** je implementirana kao turnirska sa veličinom turnira 5
- Korišćeno je jednopoziciono **ukrštanje**, pozicija se bira iz intervala $[0, \text{dužina_jedinke})$ i pritom korišćenom implemetacijom neće postojati problem nepoželjnog ponašanja u kome je jedinka podeljena tako da različite koordinate iz iste tačke(gena) pripadnu različitim potomcima

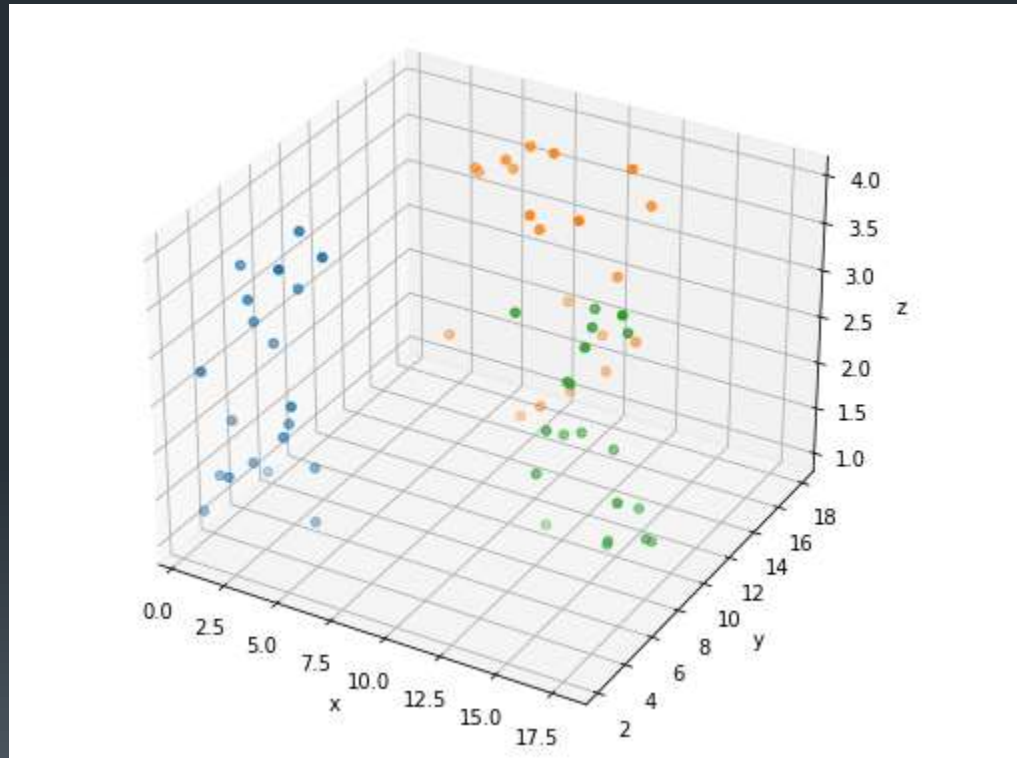
- 
- **Mutacijom** se vrši eksploracija prostora pretrage.
 - Upotrebljena je ideja dodavanja/oduzimanja slučajno generisane vrednosti (iz intervala[0,3]) svakoj koordinati posmatrane tačke ukoliko je slučajno generisana vrednost manja od MUTATION_RATE koji je 5% (0.05).

Parametri genetskog algoritma

- Vrednosti za parametre genetskog algoritma su uglavnom empirijski određene.
- Veličina populacije je 100, broj generacija 30 i elitizmom čuvamo 20% najboljih jedinki generacije.

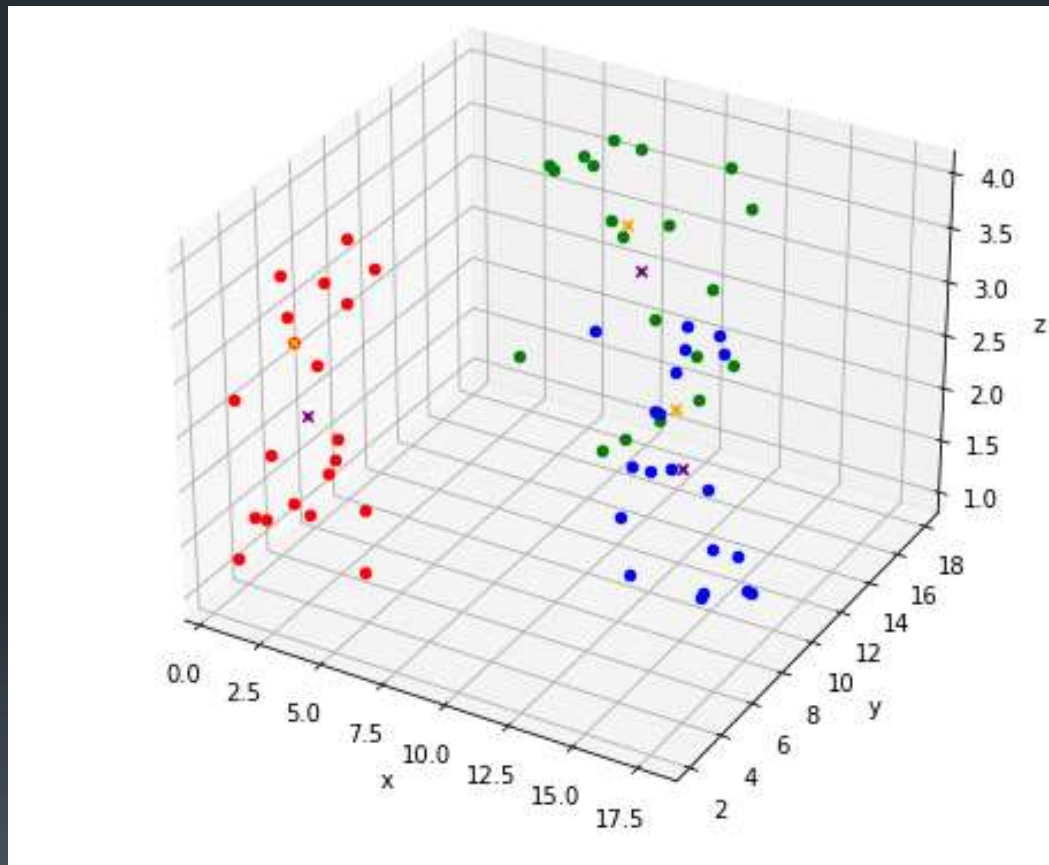
Rezultati

- Kao poredbeni algoritam koristimo K means algoritam jer on uglavnom daje optimalna resenja za posmatrani problem klasterovanja.



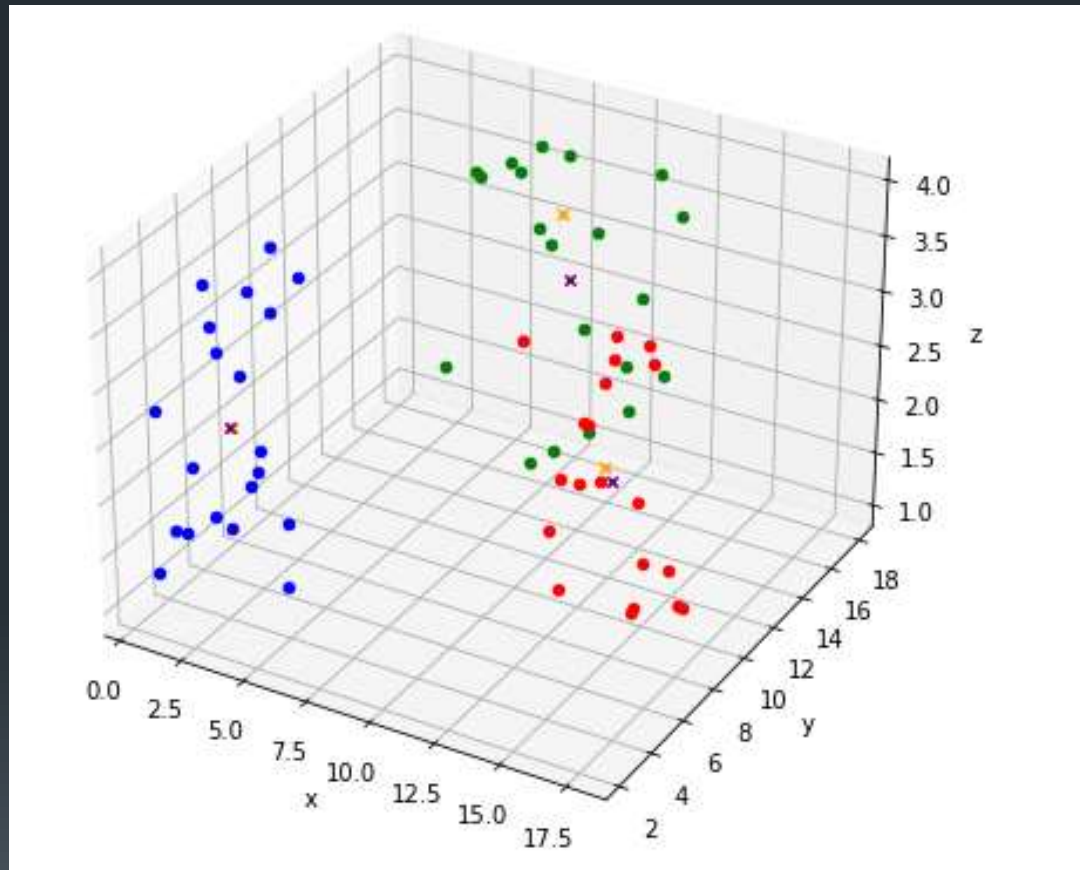
Fitness vrednost dobijena kod poredbenog K means algoritma je 0,00318645.

- Rezultati dobijeni za veličinu populacije 30 i nepromenjene ostale parametre GA:



- Fitness vrednost dobijena za navedene parametre je 0,002268748.

- Rezultati dobijeni za veličinu populacije 100 i nepromenjene ostale parametre GA:



- Fitness vrednost dobijena za navedene parametre je 0,0024967609

Zaključak

- Na osnovu prethodne analize možemo doći do zaključka da povećanjem broja jedinki unutar populacije poboljšava se i kvalitet (fitness vrednost) rešenja samog algoritma i dobijeno rešenje veoma je blizu rešenja uporednog algoritma(K means algoritma) koji u najvećem broju slučajeva daje optimalno rešenje.



Hvala na pažnji

Luka Radenković 59/2018
Ermin Škrijelj 194/2018