Klasterovanje tačaka korišćenjem genetskog algoritma

Projekat u okviru kursa Računarska inteligencija

Luka Radenković 59/2018
Ermin Škrijelj 194/2018

Jul 2022

Sadržaj

1 Opis problema

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

2 Implementacija

2.1 Obrada ulaznih podataka

Unos ulaznih podataka je omogućen na dva načina:

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

2.2 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 zadata kao jedsn kroz suma kvadratnih rastojanja(SSE) od tačaka od odgovarajuceg centra klastera kome tačka pripada. Rastojanja se računaju euklidski.

$$\text{SSE} = \sum_{i=1}^{n}(x_i - \bar{x})^2$$

$$Fitness = \frac{1}{SSE}$$

2.3 Implementacija selekcije, ukrštanja i mutacije

- Selekcija je implementirana kao turnirska sa veličinom turnira 5.

- Korišćeno je jednopoziciono ukrštanje , pozicija se bira iz intervala [0,duzina 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

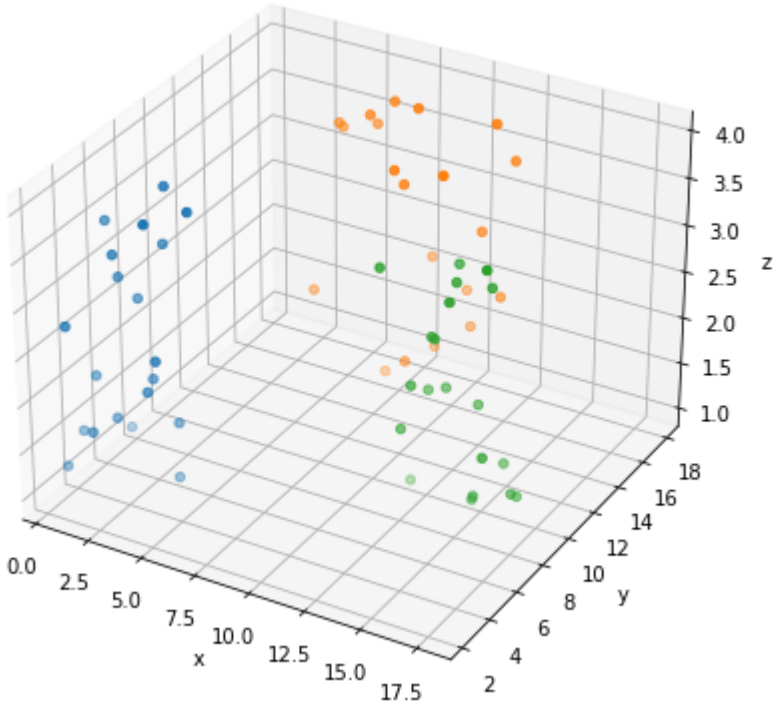
koordinati posmatrane tačke ukoliko je slučajno generisana vrednost manja od MUTATION_RATE koji je 5% (0.05).

2.4 Parametri genetskog algoritma

Vrednosti za parametre genetskog algoritma su uglavnom empirijski odredjene. Veličina populacije je 50, broj generacija 30 i elitizmom čuvamo 20% najboljih jedinki generacije.

3 Rezultati

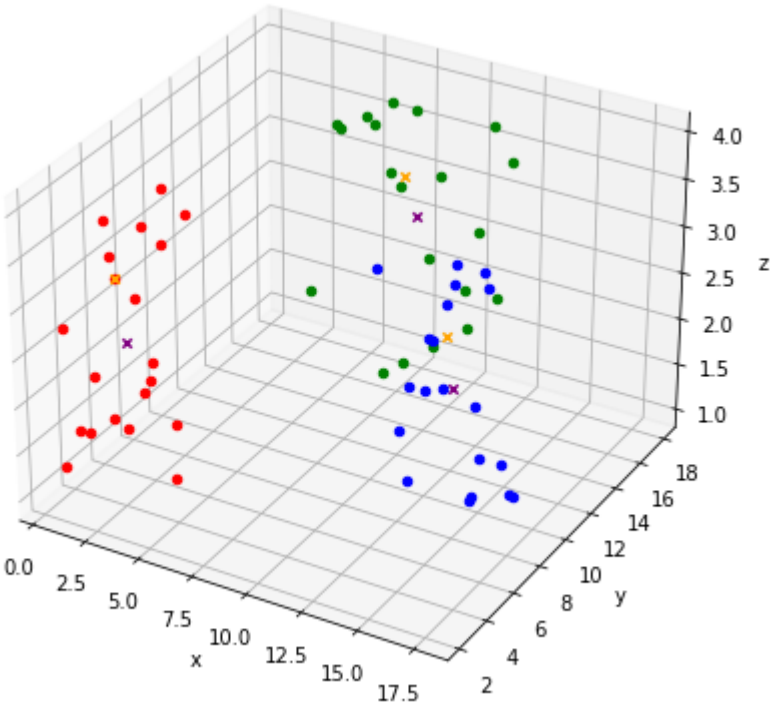
U narednom delu teksta prikazani su rezultati koje smo dobili za primer input_test :



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

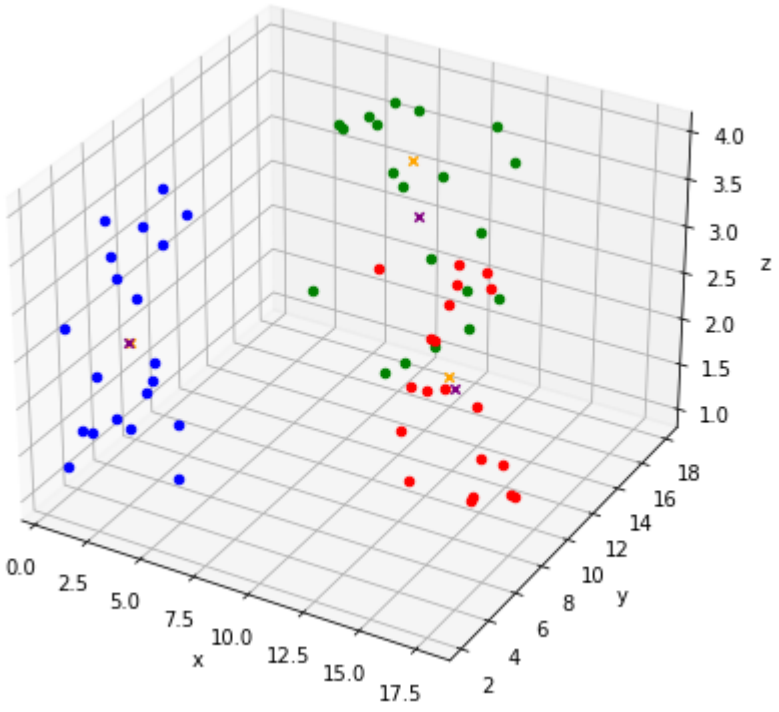
Fitness vrednost dobijena kod poredbenog K means algoritma je 0,00318645 .

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.

4 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.

References

[1] Ujjwal Maulik, Sanghamitra Bandyopadhyay - *Genetic algorithm-based clustering technique*

[2] using-the-genetic-algorithm

[3] Slajdovi profesora Nenada Mitića - Uvod u klaster analizu