

Veštačka inteligencija, praktični ispit JUN1, Grupa 1

Matematički fakultet

Školska godina 2018/2019

Napomena: Na Desktop-u se nalazi direktorijum `vi.jun1`. Preimenujte ga u oblik Vašeg indeksa `miGGBBB`. Na primer za indeks 283/2016 treba direktorijum `vi.jun1` preimenovati u `mi15283`. U ovom direktorijumu ostavite Vaše rešenje. Na Desktop-u možete pronaći dokumentaciju za potrebne Python biblioteke.
Vreme za rad: 2 sata

1. Zadat je niz celih brojeva i ciljni broj. Potrebno doći do ciljnog broj sabiranjem elemenata iz zadatog niza brojeva, koriscenjem minimalnog broja sabiraka. Sabirci se mogu ponavljati. Problem rešiti korišćenjem A* algoritma, pri čemu za heuristiku uzeti apsolutnu razliku ciljnog broja od tekućeg zbira.

```
Ulaz:  
elementi: [1, 2, 3, 6]  
cilj: 5  
Izlaz:  
1. sabirak: 3  
2. sabirak: 2
```

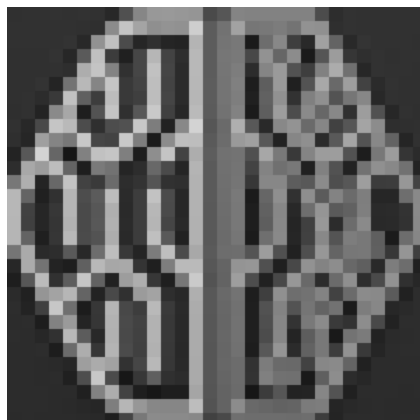
Koristiti postavku `zadatak1.ipynb`

2. Mladi Nikola Tesla je odlučio da ode na odmor od 9 dana. Tokom putovanja ka svojoj destinaciji zaspao je i imao vrlo čudan san o algoritmu koji podseća na evolutivne procese u prirodi. Daljim razmišljanjem o algoritmu, mladi Tesla je smislio da takav algoritam može da iskoristi da od slike koja predstavlja proizvoljni šum koristeći samo pogodnu funkciju prilagođenoisti generiše bilo koju sliku, a usled konstrukcije samog algoritma može i da prati kako se evolucijom originalna slika šuma polako pretvara u ciljnu sliku!

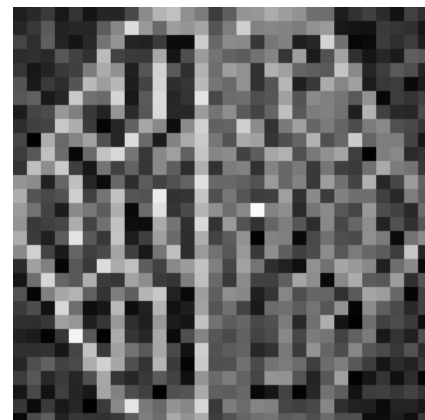
Nakon što je pročitao prethodnu priču, vredni student Matematičkog fakulteta je zamislio da pokuša da implementira Teslin san koristeći jezik Python i genetski algoritam. Uzeo je polaznu sliku 1, i iz nje odabrao jedan deo koji je isekao i čiju je rezoluciju smanjio kako bi smanjio računsku zathevnost izračunavanja - slika 2. Vredni student se nada da će dobiti prepoznatljivu aproksimaciju ciljne slike kao što je na slici 3. Pomozite mu!



Slika 1: Originalna slika



Slika 2: Ciljna slika



Slika 3: Genetski algoritam

Koristiti postavku `zadatak2.ipynb`

3. U priloženoj datoteci za rad se nalazi implemetacija klase za zadavanje formula, implementacija KNF algoritma i deo implementacije DPLL algoritma.
 - (a) Dopuniti implementaciju DPLL algoritma implementacijom pravila *pure literal*.
 - (b) Testirati rad DPLL nad formulom: $(p \Leftrightarrow q) \Rightarrow (\neg p \wedge r)$
 - (c) Napisati funkciju koja korišćenjem funkcije DPLL ispituje da li je formula *valjana*.
 - (d) Testirati rad napisane funkcije nad formulom:
 $(p \wedge (\neg p \vee q) \wedge (p \vee \neg q)) \Rightarrow (p \wedge q)$

Koristiti postavku `zadatak3.ipynb`

4. Fransis, potomak čuvenog Fransisa Galtona (eng. Francis Galton) pokušava da reši računski intenzivan optimizacioni problem. Kako bi ga rešio, potrebno je da izračunava funkciju $f(x)$ u tački x , ali na žalost izračunavanje funkcije f je jako skup proces. Fransis je zamislio da funkciju $f(x)$ aproksimira na osnovu nekog atributa x tekućeg čvora nekom linearnom funkcijom čije je izračunavanje izuzetno brzo, te je došao na ideju da koristi algoritam linearne regresije, i potrebna mu je vaša pomoć tako što ćete implementirati sledeće korake:

- (a) Vizualizujte podatke. Na x osi označite atribut x , a na y osi $f(x)$.
- (b) Izvršiti podjelu podataka na podatke za obučavanje i testiranje u razmeri 3:1.
- (c) Koristeći modul `linear_model` napraviti model linearne regresije i obučiti ga na skupu podataka za obučavanje.
- (d) Na standardni izlaz ispisati dobijene koeficijente modela (prikazati i slobodni član - eng. intercept).
- (e) Ponovo nacrtati podatke i na istoj slici nacrtati pravu dobijenu linearnom regresijom. Ose označiti isto kao u prethodnoj slici i dodati legendu koja prikazuje šta označava koja crtež na slici.
- (f) Vizualizujte podatke. Na x osi označite atribut x , a na y osi $f(x)$.

Koristiti postavku `zadatak4.ipynb`