

Veštačka inteligencija, praktični ispit JUN1

Matematički fakultet

Školska godina 2019/2020

Napomena: Na Desktop-u se nalazi direktorijum `vi.jun1`. Preimenujte ga u oblik `Ime_Prezime_BrojIndeksa_GodinaUpisa`. Na primer za indeks 283/2016 student Pera Perić treba direktorijum `vi.jun1` preimenovati u `Pera_Peric_283_2016`. U ovom direktorijumu ostavite Vaše rešenje. Na Desktop-u možete pronaći dokumentaciju za potrebne Python biblioteke.

1. Data je matrica koja predstavlja mapu po kojoj se kreće Vau-Vau. Na mapi postoje 3 vrste polja: prazno polje (kojim može da se kreće Vau-Vau), polje sa blagom (ako stane na polje sa blagom Vau-Vau osvaja to blago), klopka (ako stane na klopku Vau-Vau umire). Vau-Vau kreće od polja $[0, 0]$, a cilj mu je polje $[4, 4]$. Vau-Vau može da se kreće horizontalno i vertikalno po jedno polje, ali ne može dijagonalno. Dva igrača pomeraju Vau-Vau. Prvi igrač želi da Vau-Vau stigne do cilja sa što više para. Drugi igrač želi da Vau-Vau umre. Da igra ne bi trajala beskonačno, dogovoreno je da Vau-Vau može samo jednom da stane na neko polje. Napisati minimax implementaciju ove igre. Prvi na potezu je čovek (koji želi da osvoji što više blaga), a drugi na potezu je računar (kome je cilj da Vau-Vau umre).

Deo implementacije je već dat u datoteci `zadatak1.ipynb`. Dopuniti implementaciju.

2. (a) U iskaznoj logici korišćenjem DPLL-a proveriti da li je naredna formula valjana:

$$(p \Leftrightarrow q) \Rightarrow (\neg p \wedge r)$$

Za svaki korak DPLL-a navesti pravilo koje se u tom koraku primenjuje. Obrazložiti dobijeno rešenje.

Za simbole u tekstualnoj datoteci koristiti:

- **A** - univerzalni kvantifikator
- **E** - egzistencijalni kvantifikator
- **&** - konjunkcija
- **|** - disjunkcija
- **=>** - implikacija
- **<=>** - ekvivalencija

- (b) Korišćenjem MINISAT rešavača naći sve moguće valuacije koje zadovoljavaju prethodnu formulu.

Ulazne datoteke korišćene prilikom pokretanja minisat rešavača nazvati redom: `minisat1.txt`, `minisat2.txt`, `minisat3.txt`, itd. Obrazložiti dobijene izlaze u datotekama `minisat_izlaz1.txt`, `minisat_izlaz2.txt`, `minisat_izlaz3.txt`, itd.

3. (a) Učitati podatke iz datoteke `car.csv` za problem klasifikacije. Prikazati referentne statistike za attribute koristeći `pandas` metod `describe`.
- (b) Kategoriske attribute enkodirati koristeći `dummy` enkodiranje.
- (c) Učitane podatke razdvojiti u dva podskupa, **X** gde se nalaze svi atributi bez ciljne promenljive `class`, i **y** gde se nalazi ciljna promenljiva `class`. Dozvoljeno je koristiti sve dostupne bibliotečke funkcije.
- (d) Podeliti podatke na skup za obučavanje i validaciju u razmeri 3:1.
- (e) Napraviti potpuno povezanu neuronsku mrežu koja ima jedan skriveni sloj sa 32 neurona. Prikazati njen pregled (eng. *summary*).
- (f) Izvršiti obučavanje prethodno definisane mreže na 10 epoha sa veličinom podskupa 32. Kao grešku koristiti kategoričku unakrsnu entropiju, ostale parametre odabrati proizvoljno.
- (g) Nacrtati na istoj slici kako se menjala tačnost kroz epohe na skupovima za obučavanje i validaciju. Označiti u legendi šta označava koja linija.
- (h) Testirati nekoliko verzija prethodne arhitekture tako što im se varira veličina skrivenog sloja (1, 2, 3, 10, 32, 64). Obučiti svaku mrežu nad podacima za obučavanje i nacrtati kako se menjala tačnost kroz epohe nad skupovima za obučavanje i validaciju. Koristiti iste parametre pri obučavanju kao u delu (e).
- (i) Nacrtati kako se menjala tačnost kroz epohe svih modela na skupu za obučavanje (1. slika) i skupu za validaciju (2. slika). Na slikama označiti koja linija označava koju verziju arhitekture.

Koristiti postavku `zadatak3.ipynb`.