

Objektno programiranje, Ispit SEPT1, Grupa 1

Matematički fakultet

Školska godina 2017/2018

Napomena: Napraviti direktorijum /home/ispit/IdeaProjects. Pokrenuti *Intellij Idea* i za Project location postaviti /home/ispit/IdeaProjects direktorijum, a za Project name oop_Ime_Prezime_Indeks_Asistent (npr. oop_Pera_Peric_mi12082_NM). U napravljenom projektu, paket takođe nazvati isto tako.

Kod **ne sme** imati sintaksnih grešaka niti izbacivanje NullPointerException-a.

Vreme za rad: **2.5 sata**

Inicijalni asistenti: Biljana - BS, Anja - AB, Božidar - BA, Nemanja - NM

U tekstu je dat opis klasa, njihovih atributa i metoda. **Dozvoljeno** je (i ohrabrujemo Vas) dodati nove atribute, klase, metode, enume, interfejsa u slučaju da Vam olakšavaju implementaciju, i/ili smatrate da Vam poboljšavaju kvalitet koda i slično. Nekada će zahtevi u zadatku i zahtevati od Vas da dodate novi atribut ili slično.

Da bi se uspešno položio ispit potrebno je osvojiti barem 50% poena.

Klasifikacija je problem predviđanja kategoričke ciljne promenljive. Kategoričkim se smatraju promenljive koje uzimaju konačan broj vrednosti među kojima nema uređenja. Na primer, prepoznavanje jedne iz skupa poznatih osoba čije se lice nalazi na slici je problem klasifikacije. Prepoznavanje da li se novinski članak tiče ekonomije, sporta ili politike je takođe problem klasifikacije. [M. Nikolić, A. Zečević - Mašinsko Učenje]

Kompanija *MLA Corp.* je postavila problem određivanja pola ljudi na osnovu njihove visine i težine. Potrebno je razviti klasifikator koji će sa visokom preciznošću na osnovu podataka koje je kompanija priložila biti u stanju da reši postavljen problem.

1. Napraviti enum `Pol` koji može imati vrednosti `ZENSKI` (ženski pol) i `MUSKI` (muški pol). Napisati statički metod `static Pol createFromString(String s)` koji od stringa `s` konstruiše enum `ZENSKI` ako je prosleđen string "z", a inače konstruiše enum `MUSKI`.
2. Napraviti klasu `Podatak` koji opisuje visinu i težinu ljudi. Klasa poseduje atribute `visina (int)`, `tezina (int)` i `pol (Pol)`. Napraviti konstruktor koji prihvata sva tri atributa, napraviti `get` metode i `String toString()` metod koji vraća podatak u obliku `visina tezina pol` (videti primer ispod).

170 64 z

3. Napraviti klasu `Rastojanje`. Implementirati statički metod:

- `static double izracunajRastojanje(Podatak p1, Podatak p2)`

koji izračunava euklidsko rastojanje između podataka `p1` i `p2`.

$$d(p1, p2) = \sqrt{(p1.visina - p2.visina)^2 + (p1.tezina - p2.tezina)^2}$$

4. Napraviti apstraktnu klasu `Klasifikator` koja sadrži atribut `podaci (List<Podatak>)`. Implementirati konstruktor klase koji prihvata listu podataka i postavlja atribut `podaci` na prosleđenu listu. Implementirati metod `double greska(List<Pol> labele)` koji računa udeo pogodnih polova (lista `labele`) u odnosu na ukupan broj podataka.

Na primer ako su pogodna 3 pola od 5, funkcija treba da vrati $\frac{3}{5}$ odnosno 0.6.

Klasa poseduje apstraktan metod `List<Pol> klasifikuj()`.

5. Napraviti klasu `NasumicniKlasifikator` koja nasleđuje klasu `Klasifikator`. Potom implementirati metod `List<Pol> klasifikuj()` tako da sa jednakim verovatnoćama dodeljuje labele podacima (dakle za nepoznat podatak, sa verovatnoćom 0.5 dodeljuje pol `muški` i sa verovatnoćom 0.5 dodeljuje pol `ženski`). Koristiti klasu `Random` za generisanje pseudo-slučajnih brojeva.

NAPOMENA: Metod `setSeed` možete iskoristiti da postavite koren generatora (eng. *seed*) na 42 kako bi vam se rezultati poklopili sa test primerom sa slike 1.

6. Napraviti klasu `KSusedaPar`. Klasa sadrži atribute:

- `indeksUPodacima (int)` - indeks podatka u originalnim podacima
- `rastojanje (double)` - rastojanje podatka od fiksiranog podatka

Napraviti konstruktor koji prihvata vrednosti za oba atributa.

7. Napraviti klasu `KSuseda` koja nasleđuje klasu `Klasifikator` i implementira algoritam `K NAJBLIŽIH SUSEDA` (eng. *K nearest neighbours - KNN*). Klasa poseduje atribut `brojSuseda (int)` koji predstavlja broj suseda koji će biti razmatran pri određivanju pola za podatak.

Implementirati metod `List<Pol> klasifikuj()` tako da za podatke nadklase `podaci` određuje pol za svaki podatak, tako što pol dodeljuje na osnovu `brojSuseda` najbližih (koristiti klasu `Rastojanje`) suseda (podataka). Prethodno implementirana klasa `KSusedaPar` Vam može olakšati implementaciju metoda za klasifikaciju.

Na primer, neka su za podatak $d = (160, 61)$ 3 najbliža suseda $(162, 66, m)$ i $(170, 69, z)$, $(164, 62, z)$. Algoritam dodeljuje pol žensko na osnovu toga što je više ženskih suseda od muških (2 naspram 1).

8. U klasi `Podatak` implementirati statički metod `List<Podatak> ucitajPodatke()` koji iz datoteke `podaci.txt` učitava podatke o visini, težini i polu ljudi i vraća ih u listi. Ukoliko ne uspe učitavanje datoteke, metod vraća `null`.
9. Napraviti klasu `Main` koja nasleđuje `Application` klasu biblioteke `javafx` i izgleda kao na slikama 1 i 2. Obezbediti da je unapred selektovano prvo radio dugme (Nasumični klasifikator) i da u svakom trenutku može biti selektovano tačno jedno radio dugme.

Na klik dugmeta `Ucitaj podatke` iz datoteke `pokemoni.txt` učitavaju se podaci koristeći statički metod `ucitajPodatke()` iz klase `Podatak` i prikazuju se u `TextArea` elementu. Ukoliko ne uspe učitavanje, u `TextArea` element je potrebno ispisati poruku `Neuspesno ucitavanje!`.

Na klik dugmeta `Klasifikuj`, u zavisnosti od odabranog radio dugmeta, vrši se klasifikacija podataka koristeći odabrani klasifikator, a nakon toga se izračunava greška (videti slike 1 i 2). U `TextArea` element pored zapisa podatka, potrebno je u zagradi dopisati pol koji je odredio klasifikator.

Ukoliko je korisnik odabrao `KNN` klasifikator, kao parametar za broj suseda mu postaviti vrednost koju je korisnik prosledio u `TextField` elementu ispod labele `Broj suseda`. Ukoliko je korisnik pokušao klasifikaciju bez unetog validnog broja za broj suseda, u `TextField` element upisati poruku `Morate uneti broj suseda!`.

Dozvoljeno je proširiti klase dodatnim atributima i metodama kako biste realizovali prethodno navedene zahteve.

[podaci.txt]
visina, tezina, pol

```
160,50,z
161,52,z
163,55,z
162,53,z
170,60,z
160,50,z
173,73,m
175,74,m
172,99,m
179,81,m
185,83,m
192,93,m
```

Ucitaj podatke

160 50 z (z)
161 52 z (z)
163 55 z (m)
162 53 z (m)
170 60 z (z)
160 50 z (z)
173 73 m (m)
175 74 m (m)
172 99 m (m)
179 81 m (z)
185 83 m (z)

Nasumični klasifikator
 KNN klasifikator

Klasifikuj

Broj suseda

Preciznost

0.6666666666666666

Slika 1: Nasumični klasifikator

Ucitaj podatke

160 50 z (z)
161 52 z (z)
163 55 z (z)
162 53 z (z)
170 60 z (z)
160 50 z (z)
173 73 m (z)
175 74 m (m)
172 99 m (m)
179 81 m (m)
185 83 m (m)

Nasumični klasifikator
 KNN klasifikator

Klasifikuj

Broj suseda

10

Preciznost

0.9166666666666666

Slika 2: K najbližih suseda