INTELIGENTNI SISTEMI

as. ms Vladimir Jocović
as. ms Adrian Milaković

UVOD U MAŠINSKO UČENJE

05

„What we want is a machine that can learn from experience.“
- Alan Turing

MAŠINSKO UČENJE

Šta je mašinsko učenje?

Mašinsko učenje je proces koji omogućava sistemima da automatski uče i unapređuju svoje znanje na osnovu prethodnog iskustva, bez potrebe za eksplicitnim programiranjem.

Gde se koristi mašinsko učenje?

- Kod problema za koje je teško ručno definisati kako se rešavaju (prepoznavanje lica, obrada prirodnih jezika, robotika, igranje igara...) te klasično programiranje nije moguće.
- Kod problema izvlačenja informacija iz velike količine sirovih podataka i predviđanja budućih trendova korišćenjem trenutno dostupnih podataka. (*Data mining*)
- Kod kompleksnih sistema koji se dinamički prilagođavaju okruženju.

MAŠINSKO UČENJE

Kako izgleda proces mašinskog učenja?

U procesu mašinskog učenja moguće je identifikovati sledeće korake (podela ne mora nužno izgledati ovako):

1. Formulisanje problema i čitanje podataka
2. Analiza podataka
3. Čišćenje podataka
4. Modifikacija i transformacija atributa
5. Izbor i treniranje modela mašinskog učenja
6. Validacija i testiranje podataka

MAŠINSKO UČENJE

Tipovi mašinskog učenja

- Nadgledano učenje (*Supervised learning*) – Svakom ulaznom podatku ***x*** je pridružena izlazna vrednost ***y***. Cilj učenja je da se na osnovu datih parova (*x*, *y*) pronađe optimalna funkcija koja mapira ulaz u izlaz. Koristi se u klasifikacionim (medicinska dijagnostika) i regresivnim (cena nekretnina) problemima.
- Nenadgledano učenje (*Unsupervised learning*) – Dati su samo ulazni podaci ***x***. Ne postoji izlazna vrednost ***y***. Potrebno je pronaći pravilnost u ulaznim podacima na osnovu kojih mogu da se generišu izlazne vrednosti. Algoritmi nenadgledanog učenja mogu se koristiti za klasterizaciju ulaznih podataka (pronalaženje sličnih podataka i njihovo grupisanje), detekciju anomalija (sumnjive transakcije na osnovu istorije kupovina), asocijaciju podataka (predikcija vrednosti drugih atributa na osnovu povezanosti datih atributa), itd.

MAŠINSKO UČENJE

Tipovi mašinskog učenja

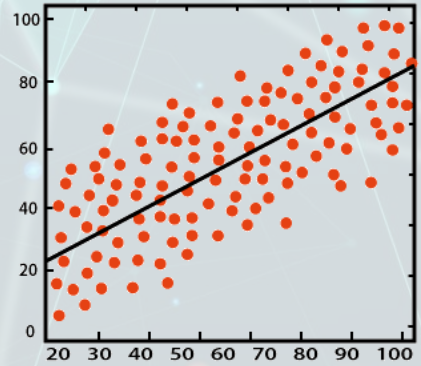
- Polunadgledano učenje (*Semi-supervised learning*) – Predstavlja kombinaciju nadgledanog i nenadgledanog učenja. Delu ulaznih podataka *x* pridružene su izlazne vrednosti *y*. Delu ulaznih podataka nisu pridružene izlazne vrednosti.
- Učenje sa podrškom (*Reinforcement learning*) – Primenjuje se na obučavanje softverskih agenata koji deluju u nekom prostoru akcija. Učenje se vrši na osnovu datih ulaznih podataka i signala podrške koji stiže na kraju nekog skupa akcija agenata i može biti pozitivan ili negativan. Cilj učenja je da iskoristi signal podrške da utvrdi koja tačno akcija ili skup akcija je dovela do pozitivnog signala podrške i shodno tome koriguje ponašanje agenta.

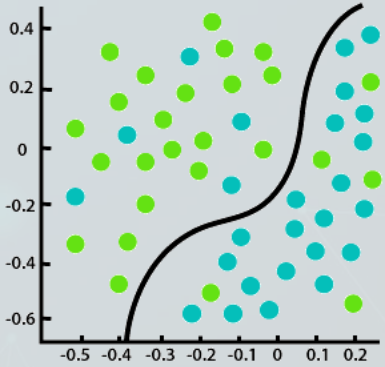
NADGLEDANO UČENJE

Tipovi izlazne vrednosti

Na osnovu tipa izlazne vrednosti ***y*** probleme nadgledanog učenja možemo podeliti na:

- **Probleme regresije** – izlazna vrednost je kontinualnog tipa (realna vrednost)
- **Probleme klasifikacije** – izlazna vrednost je kategoričkog tipa (diskretna vrednost iz skupa)





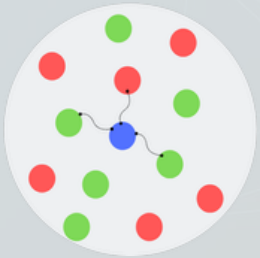
ALGORITAM K NAJBLIŽIH SUSEDA

Kako funkcioniše algoritam k najbližih suseda?

Algoritam k najbližih suseda (*k-nearest neighbors*) je tip nadgledanih algoritama mašinskog učenja. Koristi se kod problema klasifikacije (mada može da se koristi i kod problema regresije).

Algoritam k najbližih suseda polazi od pretpostavke da će se slične instance nalaziti bliže u prostoru. Algoritam klasifikuje određenu instancu na osnovu klasifikacije k najbližih instanci prostim prebrojavanjem svake klase.

U slučaju regresije, algoritam uzima srednju vrednost izlaznog parametra najbližih suseda.



Algoritam započinje biranjem vrednosti za k, potom računa rastojanje od posmatrane instance do svih instanci u skupu podataka, sortira ih i bira k instanci sa namanjim rastojanjem na osnovu kojih daje rezultat.

ALGORITAM K NAJBLIŽIH SUSEDA

Kako odabrati vrednost k?

Da bismo odabrali vrednost k, pokrećemo algoritam više puta sa različitim vrednostima i biramo onu koja najviše smanjuje broj grešaka prilikom testiranja skupa podataka.

Neka opažanja:

- Smanjivanjem vrednosti k do 1, predikcije postaju nestabilnije. Pretpostavimo da određujemo boju nekog oblika oko kojeg se nalazi mnoštvo crvenih oblika i jedan zeleni koji je ujedno i najbliži. Naravno, jasno je da je oblik koji ispitujemo najverovatnije crven, međutim kako je zeleni oblik najbliži, KNN netačno previđa da je oblik zelene boje.
- Sa druge strane, povećanjem vrednosti k, povećavamo i stabilnost predikcije, ali do neke granice. U nekom trenutku, vrednost k će da bude prevelika i obuhvata veći deo skupa podataka čime povećava broj grešaka.
- Najčešće se za vrednost k uzima neparan broj, kako bismo imali *tiebreaker*.
- Jako često se za vresnot k uzima kvadratni koren ukupnog broja podataka u skupu.

ALGORITAM K NAJBLIŽIH SUSEDA

Na koji način odrediti najbliže susede?

Za pronalaženje najbližih suseda između dve tačke P i Q u N-dimenzionalnom prostoru date koordinatama P = (p1, p2, ... , pN) i Q = (q1, q2, ... , qN) koristi se neka od sledećih metrika:

- Euklidska razdaljina - $\sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$
- Menhetn razdaljina - $\sum_{i=1}^{n}|p_i - q_i|$
- Čebiševa razdaljina - $\max(|p_i - q_i|)$

i mnoge druge.

Zadatak 1 – Mršava osoba ili ne?

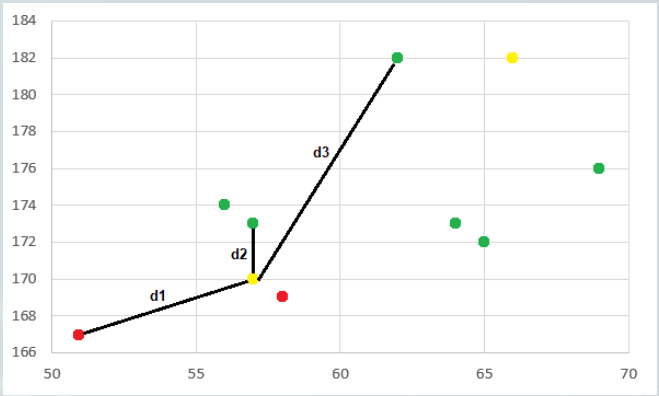
★★☆☆☆

Odrediti da li su posmatrane osobe mršave koristeći *knn* algoritam sa Euklidskim rastojanjem na posmatranom skupu podataka.

Nezavisni atributi		**Atribut odluke**
Masa (kg)	Visina (cm)	Mršava?
51	167	Da
62	182	Ne
69	176	Ne
64	173	Ne
65	172	Ne
56	174	Da
58	169	Ne
57	173	Ne
57	170	?
66	182	?

Zadatak 1 – Rešenje

Na grafiku je vizuelno prikazan skup podataka. Na x osi su prikazane mase osoba, na y osi visine osoba. Crvenim oznakama su prikazane mršave osobe, zelenim oznakama osobe koje nisu mršave, a žutim osobe koje nisu klasifikovane.



Kako računamo rastojanje između posmatrane instance i ostalih? Pr.

d1 = $\sqrt{(170-167)^2+(57-51)^2}$ = 6.71

d2 = $\sqrt{(170-173)^2+(57-57)^2}$ = 3.00

d3 = $\sqrt{(170-182)^2+(57-62)^2}$ = 13.00

Zadatak 1 – Rešenje

Za prvu nepoznatu osobu tražimo Euklidsku razdaljinu do svih drugih tačaka, pa na osnovu faktora *k* određujem koji su susedi najbliži.

Za *k* uzimamo kvadratni koren ukupnog broja podataka u skupu. ($k = \sqrt{8} \approx 3$)

Na osnovu klasifikacije najbližih suseda, određujemo da nepoznata osoba nije mršava.

Masa (kg)	Visina (cm)	Mršava?	Udaljenost
51	167	Da	6.71
62	182	Ne	13.00
69	176	Ne	13.42
64	173	Ne	7.62
65	172	Ne	8.25
56	174	**Da**	**4.12**
58	169	**Ne**	**1.41**
57	173	**Ne**	**3.00**
57	170	**Ne**	

Zadatak 1 – Rešenje

Za drugu nepoznatu osobu tražimo Euklidsku razdaljinu do svih drugih tačaka, pa na osnovu faktora *k* određujem koji su susedi najbliži.

Za *k* uzimamo kvadratni koren ukupnog broja podataka u skupu. ($k = \sqrt{8} \approx 3$)

Na osnovu klasifikacije najbližih suseda, određujemo da nepoznata osoba nije mršava.

Masa (kg)	Visina (cm)	Mršava?	Udaljenost
51	167	Da	21.21
62	182	**Ne**	**4.00**
69	176	**Ne**	**6.71**
64	173	**Ne**	**9.21**
65	172	Ne	10.05
56	174	Da	12.81
58	169	Ne	15.26
57	173	Ne	12.73
66	182	**Ne**	

Zadatak 2 – Crno ili belo vino?

★★★☆☆

Odrediti da li je posmatrano vino crno ili belo koristeći *knn* algoritam sa Euklidskim rastojanjem na posmatranom skupu podataka koristeći za *k* vrednosti: (a) $\sqrt{n}$, (b) 7

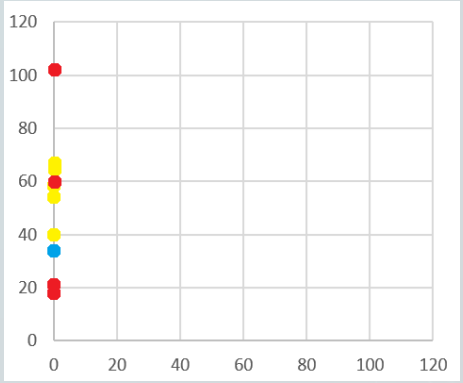
Nezavisni atributi		**Atribut odluke**
Nivo sumpor dioksida	Nivo hlora	Crno/Belo vino
34	0.076	Crveno
67	0.098	Belo
54	0.092	Belo
60	0.075	Crveno
59	0.089	Belo
40	0.075	Belo
59	0.069	Crveno
21	0.065	Crveno
18	0.073	Crveno
102	0.071	Crveno
65	0.097	Belo
34	0.084	?

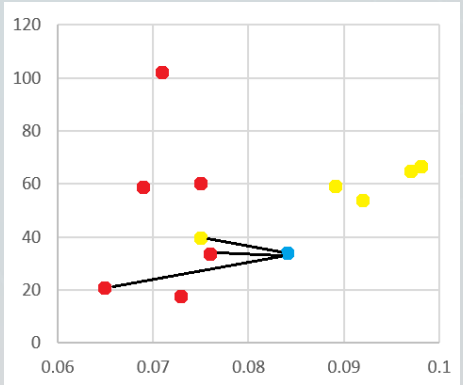
Zadatak 2 – Rešenje

Da li je dobro računati razdaljinu sa ovakvim podacima?

Nivo sumpor dioksida	Nivo hlora	Udaljenost
34	0.076	0.008
67	0.098	33
54	0.092	20
60	0.075	26
59	0.089	25
40	0.075	6.000007
59	0.069	25
21	0.065	13.00001
18	0.073	16
102	0.071	68
65	0.097	31
34	0.084	?

Nivo sumpor dioksida mnogo više utiče na udaljenost nego nivo hlora jer su razlike u vrednostima veće.





Zadatak 2 – Rešenje

Normalizacija je proces reskaliranja svih vrednosti u opseg od 0 do 1. Na taj način sve vrednosti jednako utiču na računanje distance između instanci.

$$x_{novo} = \frac{x_{staro} - x_{min}}{x_{max} - x_{min}}$$

Pr.

x_{min} = 1

x_{min} = 14

$$x_{novo1} = \frac{1-1}{14-1} = 0$$

$$x_{novo2} = \frac{5-1}{14-1} = \frac{4}{13}$$

$$x_{novo3} = \frac{14-1}{14-1} = 1$$

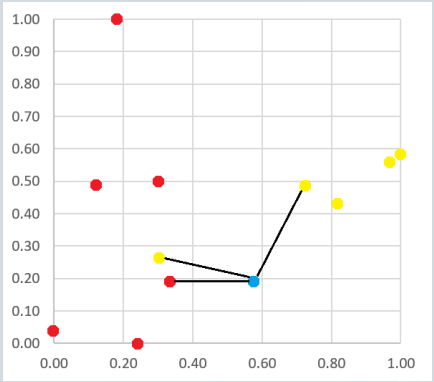
...

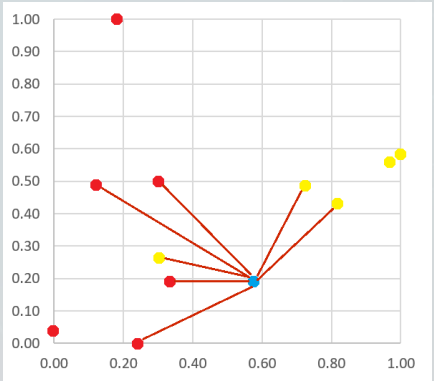
Stara vrednost	Nova vrednost
1	0
5	0.31
5	0.31
14	1
6	0.38
3	0.15
8	0.54
9	0.62
11	0.77
4	0.23
7	0.46

Zadatak 2 – Rešenje

Podaci nakon normalizacije i grafici za $k = \sqrt{11} \approx 3$ i $k = 7$.

Nivo sumpor dioksida	Nivo hlora	Udaljenost	Crno/Belo vino
0.19	0.33	0.242	Crveno
0.58	1	0.578	Belo
0.43	0.82	0.340	Belo
0.50	0.3	0.413	Crveno
0.49	0.73	0.334	Belo
0.26	0.3	0.282	Belo
0.49	0.12	0.543	Crveno
0.04	0	0.596	Crveno
0.00	0.24	0.384	Crveno
1.00	0.18	0.900	Crveno
0.56	0.97	0.540	Belo
0.19	0.58	-	?





Zašto smo dobili različite rezultate za različite vrednosti *k*?

Zadatak 2 – Rešenje

Da li nam normalizacija pomaže u situacijama kada imamo tzv. *outlier*-e?

Nezavisni atributi		Atribut odluke
Nivo sumpor dioksida	Nivo hlora	Crno/Belo vino
34	0.076	Crveno
67	0.098 → 0.980	Belo
54	0.092	Belo
60	0.075	Crveno
59	0.089	Belo
40	0.075	Belo
59	0.069	Crveno
21	0.065	Crveno
18	0.073	Crveno
102	0.071	Crveno
65	0.097	Belo
34	0.084	?

Zadatak 2 – Rešenje

Nivo hlora gubi na značaju zbog prevelikog odstupanja jedne instance.

Nivo sumpor dioksida	Nivo hlora	Udaljenost	Crno/Belo vino
0.19	0.33	0.242	Crveno
0.58	1	0.578	Belo
0.43	0.82	0.340	Belo
0.50	0.3	0.413	Crveno
0.49	0.73	0.334	Belo
0.26	0.3	0.282	Belo
0.49	0.12	0.543	Crveno
0.04	0	0.596	Crveno
0.00	0.24	0.384	Crveno
1.00	0.18	0.900	Crveno
0.56	0.97	0.540	Belo
0.19	**0.58**	-	?

Nivo sumpor dioksida	Nivo hlora	Udaljenost	Crno/Belo vino
0.19	0.01	0.009	Crveno
0.58	**1**	1.055	Belo
0.43	0.03	0.238	Belo
0.50	0.01	0.310	Crveno
0.49	0.03	0.298	Belo
0.26	0.01	0.072	Belo
0.49	0.01	0.298	Crveno
0.04	0	0.156	Crveno
0.00	0.01	0.191	Crveno
1.00	0.01	0.810	Crveno
0.56	0.03	0.369	Belo
0.19	**0.02**	-	?

Zadatak 3 – Filmovi

★★★★☆

Na osnovu skupa podataka, preporučiti korisniku 6 filmova koji su najsličniji filmu *The Imitation Game* koristeći *knn* algoritam sa Menhetn rastojanjem i uporediti rezultate ukoliko: (a) ne koristimo normalizaciju vrednosti atributa, (b) koristimo normalizaciju vrednosti atributa

Film	IMDB ocena	Žanr
The Imitation Game	8	Biografski, Drama, Triler
Ex Machina	7.7	Drama, Misterija
A Beautiful Mind	8.2	Biografski, Drama
Good Will Hunting	8.3	Drama
Forrest Gump	8.8	Drama
21	6.8	Drama
Gifted	7.6	Drama
Travelling Salesman	5.9	Drama, Misterija
Avatar	7.9	-
The Wolf of Wall Street	8.2	Biografski, Komedija
A Time To Kill	7.4	Drama, Triler
Interstellar	8.6	Drama
The Wind Rises	7.8	Biografski, Drama

Zadatak 3 – Rešenje

Neki algoritmi (kao KNN) ne podržavaju rad sa kategoričkim podacima, stoga je prvo potrebno transformisati tabelu. Transformacija kategoričkih vrednosti direktno u numeričke vrednosti ima smisla ukoliko su kategoričke vrednosti uporedive. Transformacija na ovakav način nije dobra jer se nekim kategoričkim vrednostima daje veći prioritet za neke algoritme.

Plata
Niska
Srednja
Visoka
Srednja

Plata
0
1
2
1

Boja
Plava
Zelena
Plava
Crvena

Boja
0
1
0
2

Zadatak 3 – Rešenje

Takve kategoričke vrednosti ćemo pretvoriti u numeričke koristeći *one-hot-encoding* tehniku koja za svaku moguću kategoričku vrednost nekog atributa kreira novi atribut čija je vrednost 0 ili 1.

Boja
Plava
Zelena
Plava
Crvena

Crvena	Zelena	Plava
0	0	1
0	1	0
0	0	1
1	0	0

Zadatak 3 – Rešenje

Za svaki žanr definišemo poseban atribut čija vrednost (1 ili 0) određuje prisustvo žanra u filmu.

Film	IMDB ocena	Biografski	Drama	Triler	Komedija	Misterija
The Imitation Game	8	1	1	1	0	0
Ex Machina	7.7	0	1	0	0	1
A Beautiful Mind	8.2	1	1	0	0	0
Good Will Hunting	8.3	0	1	0	0	0
Forrest Gump	8.8	0	1	0	0	0
21	6.8	0	1	0	0	0
Gifted	7.6	0	1	0	0	0
Travelling Salesman	5.9	0	1	0	0	1
Avatar	7.9	0	0	0	0	0
The Wolf of Wall Street	8.2	1	0	0	1	0
A Time To Kill	7.4	0	1	1	0	0
Interstellar	8.6	0	1	0	0	0
The Wind Rises	7.8	1	1	0	0	0

Zadatak 3 – Rešenje

Tabele najsličnijih filmova ukoliko se ne koristi (levo), odnosno koristi (desno) normalizacija.

Film	Sličnost
The Imitation Game	
Ex Machina	3.30
A Beautiful Mind	**1.20**
Good Will Hunting	**2.30**
Forrest Gump	2.80
21	3.20
Gifted	**2.40**
Travelling Salesman	5.10
Avatar	3.10
The Wolf of Wall Street	3.20
A Time To Kill	**1.60**
Interstellar	2.60
The Wind Rises	**1.20**

Film	Sličnost
The Imitation Game	
Ex Machina	3.10
A Beautiful Mind	**1.07**
Good Will Hunting	**2.10**
Forrest Gump	2.28
21	2.41
Gifted	**2.14**
Travelling Salesman	3.72
Avatar	3.03
The Wolf of Wall Street	3.07
A Time To Kill	**1.21**
Interstellar	2.21
The Wind Rises	**1.07**

Zadatak 4 – Medicinska dijagnostika

★★★★☆

Dat je skup podataka o obavljenoj analizi nad pacijentima i informacije da li boluju od posmatrane bolesti. Zbog privatnosti podataka, originalne vrednosti i opis atributa nije dat.

Pacijent	X1	X2	X3	Bolest
T1	0.74	0.02	0.44	da
T2	0.91	0.13	0.44	ne
T3	0.92	0.07	0.36	ne
T4	0.85	0.13	0.27	ne
T5	0.80	0.03	0.56	da
T6	0.87	0.11	0.17	ne
T7	0.75	0.17	0.39	ne
T8	0.86	0.06	0.41	ne
T9	0.84	0.15	0.47	ne
T10	0.79	0.11	0.50	ne
T11	0.81	0.09	0.54	da
T12	0.69	0.12	0.40	ne
T13	0.99	0.12	0.39	ne
T14	0.95	0.13	0.37	ne
T15	0.94	0.11	0.22	ne

Zadatak 4 – Medicinska dijagnostika

★★★★☆

Za sledeći test skup odrediti tačnost, preciznost i odziv koristeći *knn* algoritam sa *Manhattan* distancom i parametrom k=3.

Pacijent	X1	X2	X3	Bolest
T16	0.76	0.05	0.45	da
T17	0.79	0.06	0.51	da
T18	0.72	0.02	0.39	da
T19	0.82	0.11	0.38	ne
T20	0.88	0.09	0.25	ne
T21	0.91	0.13	0.27	ne
T22	0.87	0.09	0.29	ne
T23	0.86	0.14	0.33	ne

Zadatak 4 – Rešenje

U tabeli su date distance od tačke T16 i obeležena tri najbliža suseda. Analogno bi se radilo za preostale tačke skupa za testiranje.

Pacijent	X1	X2	X3	Bolest	Distanca
T1	0.74	0.02	0.44	da	0.06
T2	0.91	0.13	0.44	ne	0.24
T3	0.92	0.07	0.36	ne	0.27
T4	0.85	0.13	0.27	ne	0.35
T5	0.80	0.03	0.56	da	0.17
T6	0.87	0.11	0.17	ne	0.45
T7	0.75	0.17	0.39	ne	0.19
T8	0.86	0.06	0.41	ne	0.15
T9	0.84	0.15	0.47	ne	0.20
T10	0.79	0.11	0.50	ne	0.14
T11	0.81	0.09	0.54	da	0.18
T12	0.69	0.12	0.40	ne	0.19
T13	0.99	0.12	0.39	ne	0.36
T14	0.95	0.13	0.37	ne	0.35
T15	0.94	0.11	0.22	ne	0.47

Zadatak 4 – Rešenje

Na osnovu stvarnih izlaza i prediktovanih izlaza računamo svu potrebnu metriku.

Pacijent	X1	X2	X3	Bolest	Predikcija
T16	0.76	0.05	0.45	da	ne
T17	0.79	0.06	0.51	da	da
T18	0.72	0.02	0.39	da	ne
T19	0.82	0.11	0.38	ne	ne
T20	0.88	0.09	0.25	ne	ne
T21	0.91	0.13	0.27	ne	ne
T22	0.87	0.09	0.29	ne	ne
T23	0.86	0.14	0.33	ne	ne

True positive – Broj ispravno prediktovanih „da“ primeraka (Ima ih 1)
True negative – Broj ispravno prediktovanih „ne“ primeraka (Ima ih 5)
False positive – Primerci označeni sa „ne“ koji su pogrešno prediktovani kao „da“ (Ima ih 0)
False negative – Primerci označeni sa „da“ koji su pogrešno prediktovani kao „ne“ (Ima ih 2)

Tačnost = *broj_tačnih_predikcija / ukupan_broj_predikcija = (TP+TN) / (TP+TN+FP+FN)*
Tačnost = 6/8 = 0.75 (75%)

Zadatak 4 – Rešenje

Pacijent	X1	X2	X3	Bolest	Predikcija
T16	0.76	0.05	0.45	da	ne
T17	0.79	0.06	0.51	da	da
T18	0.72	0.02	0.39	da	ne
T19	0.82	0.11	0.38	ne	ne
T20	0.88	0.09	0.25	ne	ne
T21	0.91	0.13	0.27	ne	ne
T22	0.87	0.09	0.29	ne	ne
T23	0.86	0.14	0.33	ne	ne

Preciznost (da) = TP / (TP + FP) = 1 / 1 = 1.00 (100%)
Odziv (da) = TP / (TP + FN) = 1 / 3 = 0.33 (33%)

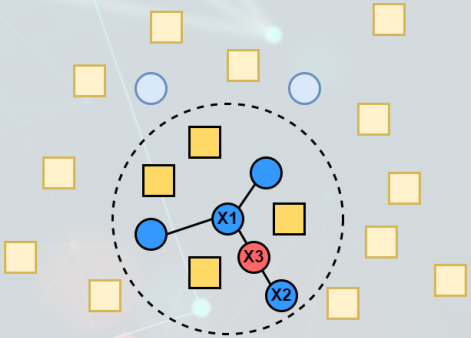
Preciznost (ne) = 5/7 = 0.71 (71%)
Odziv (ne) = 5/5 = 1.00 (100%)

Iako je tačnost važna, ona nije presudna u nekim primerima. U slučaju medicinske dijagnostike, čak i da imamo tačnost od 99%, to nam ništa ne znači ukoliko je greška od 1% prouzrokovana *false negative* primercima, jer nismo uspeli da otkrijemo bolest na vreme kod pacijenata koji su je zapravo imali.
Odziv za pozitivne primerke je vrlo važan u medicinskoj dijagnostici, proveri spama, prevarama u bankarskim transakcijama itd.

Zadatak 4 – Rešenje

Loš odziv za „da“ primerke je posledica neuravnoteženog skupa podataka (premali je procenat stvarno bolesnih u odnosu na one koji nisu bolesni – čest slučaj u praksi).
Da bismo učinili skup uravnoteženim primenjujemo jednu od sledeće dve tehnike:
- *oversampling* – sinteza novih primeraka klase u manjini
- *undersampling* – uklanjanje primeraka klase koja preovladava

Oversampling može da se uradi pomoću SMOTE algoritma. Ideja algoritma jeste da nasumično odabere primerak klase u manjini (x1) i pronađe njegovih *k* najbližih suseda klase u manjini. Potom, nasumično odabere jedan od *k* najbližih suseda (x2) i sintetiše nov primerak (x3) na nasumičnoj poziciji na putanji između odabrana dva primerka (između x1 i x2, najčešće na sredini). Algoritam se ponavlja dok se ne sintetiše željen broj primeraka.



Zadatak 4 – Rešenje

Primenom SMOTE algoritma možemo da povećamo skup pacijenata koji su bolesni i približimo ga broju pacijenata koji nisu bolesni kako bismo potencijalno napravili bolji model sa većim odzivom. Neka je parametar SMOTE algoritma k=1, a svaki novi primerak se sintetiše na sredini.

Pacijent	X1	X2	X3	Bolest
T1	0.74	0.02	0.44	da
T5	0.80	0.03	0.56	da
T11	0.81	0.09	0.54	da

Nasumično je odabran primerak T1. Traži se njegov najbliži sused:
Dist(T1-T5) = 0.06 + 0.01 + 0.12 = 0.19
Dist(T1-T11) = 0.07 + 0.07 + 0.10 = 0.24
Kako je primerak T5 najbliži, sintetiše se nov primerak, L1 na sredini između primeraka T1 i T5.
X1 = (0.74+0.80)/2 = 0.77
X2 = (0.02+0.03)/2 = 0.025
X3 = (0.44+0.56)/2 = 0.50
Bolest = da

Zadatak 4 – Rešenje

Ponavljanjem SMOTE algoritma, možemo da generišemo 9 novih primeraka, kako bi ukupan broj primeraka pacijenata sa i bez bolesti bio po 12 u oba slučaja, čime dobijamo uravnotežen skup.

Pacijent	X1	X2	X3	Bolest
L1	0.77	0.03	0.50	da
L2	0.78	0.06	0.49	da
L3	0.81	0.06	0.55	da
L4	0.77	0.04	0.50	da
L5	0.79	0.04	0.53	da
L6	0.79	0.06	0.52	da
L7	0.77	0.04	0.50	da
L8	0.79	0.06	0.52	da
L9	0.79	0.04	0.53	da

Zadatak 4 – Rešenje

Ponavljamo KNN algoritam sa povećanim k zbog povećanog broja primeraka i određujemo tačnost, preciznost i odziv. U nastavku su date distance od primerka T16.

Pacijent	Bolest	Distanca
T1	da	0.06
T2	ne	0.24
T3	ne	0.27
T4	ne	0.35
T5	da	0.17
T6	ne	0.45
T7	ne	0.19
T8	ne	0.15
T9	ne	0.20
T10	ne	0.14
T11	da	0.18
T12	ne	0.19
T13	ne	0.36
T14	ne	0.35
T15	ne	0.47

Pacijent	Bolest	Distanca
L1	da	0.08
L2	da	0.07
L3	da	0.16
L4	da	0.07
L5	da	0.11
L6	da	0.11
L7	da	0.07
L8	da	0.11
L9	da	0.11

Zadatak 4 – Rešenje

Na osnovu stvarnih izlaza i prediktovanih izlaza računamo svu potrebnu metriku.

Pacijent	X1	X2	X3	Bolest	Predikcija
T16	0.76	0.05	0.45	da	da
T17	0.79	0.06	0.51	da	da
T18	0.72	0.02	0.39	da	da
T19	0.82	0.11	0.38	ne	ne
T20	0.88	0.09	0.25	ne	ne
T21	0.91	0.13	0.27	ne	ne
T22	0.87	0.09	0.29	ne	ne
T23	0.86	0.14	0.33	ne	ne

True positive – Broj ispravno prediktovanih „da“ primeraka (Ima ih 3)
True negative – Broj ispravno prediktovanih „ne“ primeraka (Ima ih 5)
False positive – Primerci označeni sa „ne“ koji su pogrešno prediktovani kao „da“ (Ima ih 0)
False negative – Primerci označeni sa „da“ koji su pogrešno prediktovani kao „ne“ (Ima ih 0)

Tačnost = 8/8 = 1.00 (100%)
Preciznost (da) = TP / (TP + FP) = 3/3 = 1.00 (100%)
Odziv (da) = TP / (TP + FN) = 3/3 = 1.00 (100%)

Preciznost (ne) = 7/7 = 1.00 (100%)
Odziv (ne) = 5/5 = 1.00 (100%)

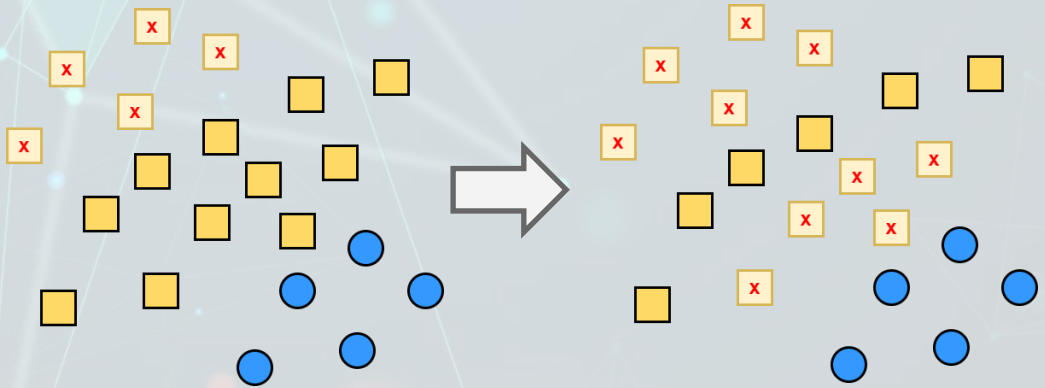
Zadatak 4 – Rešenje

U tabeli su date distance od tačke T16 i obeležena tri najbliža suseda. Analogno bi se radilo za preostale tačke skupa za testiranje.

Pacijent	X1	X2	X3	Bolest	Distanca
T1	0.74	0.02	0.44	da	0.06
T2	0.91	0.13	0.44	ne	0.24
T3	0.92	0.07	0.36	ne	0.27
T4	0.85	0.13	0.27	ne	0.35
T5	0.80	0.03	0.56	da	0.17
T6	0.87	0.11	0.17	ne	0.45
T7	0.75	0.17	0.39	ne	0.19
T8	0.86	0.06	0.41	ne	0.15
T9	0.84	0.15	0.47	ne	0.20
T10	0.79	0.11	0.50	ne	0.14
T11	0.81	0.09	0.54	da	0.18
T12	0.69	0.12	0.40	ne	0.19
T13	0.99	0.12	0.39	ne	0.36
T14	0.95	0.13	0.37	ne	0.35
T15	0.94	0.11	0.22	ne	0.47

Zadatak 4 – Rešenje

Undersampling može da se uradi pomoću *NearMiss* algoritma. Postoje tri verzije algoritma, a ovde će biti iskorišćena verzija 3. Algoritam se izvršava u dva koraka. Ideja algoritma jeste da za svaki primerak klase u manjini pronađe njegovih *k* najbližih suseda klase koja nije u manjini. Od svih takvih primeraka, zadržava se samo *n* primeraka čija je srednja udaljenost do *k*-najbližih suseda klase u manjini najveća, a svi ostali primerci se odbacuju.



Zadatak 4 – Rešenje

Primenom *NearMiss* algoritma možemo da smanjimo skup pacijenata koji nisu bolesni i približimo ga broju pacijenata koji su bolesni kako bismo potencijalno napravili bolji model sa većim odzivom. Neka su parametri *NearMiss* algoritma k=2 i n=3.

Za svaki primerak klase „da“ računamo distancu do svih primeraka klase „ne“ i tražimo k najbližih.

	T2	T3	T4	T6	T7	T8	T9	T10	T12	T13	T14	T15
T1	0.28	0.31	0.39	0.49	0.21	0.19	0.26	0.20	0.19	0.40	0.39	0.51

Kada i od ostalih primeraka (T5, T11) nađemo k najbližih (T2, T7, T9 i T10), završava se prvi korak algoritma.

Zadatak 4 – Rešenje

Od posmatranih primeraka čuva se *n* primeraka čija je srednja udaljenost do *k* najbližih suseda klase „da“ najveća, a svi ostali se odbacuju.

	T1	T5	T11	AVG
T2	0.28	0.24	0.33	0.26
T7	0.21	0.29	0.36	0.25
T8	0.19	0.21	0.24	0.20
T9	0.26	0.16	0.25	0.21
T10	0.20	0.08	0.15	0.12
T12	0.19	0.29	0.36	0.24

Zadatak 4 – Rešenje

Ponavljamo KNN algoritam i određujemo tačnost, preciznost i odziv. U nastavku su date distance od primerka T16.

Pacijent	X1	X2	X3	Bolest	Distanca
T1	0.74	0.02	0.44	da	0.06
T2	0.91	0.13	0.44	ne	0.24
T5	0.80	0.03	0.56	da	0.17
T7	0.75	0.17	0.39	ne	0.19
T11	0.81	0.09	0.54	da	0.18
T12	0.69	0.12	0.40	ne	0.19

Zadatak 4 – Rešenje

Na osnovu stvarnih izlaza i prediktovanih izlaza računamo svu potrebnu metriku.

Pacijent	X1	X2	X3	Bolest	Predikcija
T16	0.76	0.05	0.45	da	da
T17	0.79	0.06	0.51	da	da
T18	0.72	0.02	0.39	da	ne
T19	0.82	0.11	0.38	ne	ne
T20	0.88	0.09	0.25	ne	ne
T21	0.91	0.13	0.27	ne	ne
T22	0.87	0.09	0.29	ne	ne
T23	0.86	0.14	0.33	ne	ne

True positive – Broj ispravno prediktovanih „da“ primeraka (Ima ih 2)
True negative – Broj ispravno prediktovanih „ne“ primeraka (Ima ih 5)
False positive – Primerci označeni sa „ne“ koji su pogrešno prediktovani kao „da“ (Ima ih 0)
False negative – Primerci označeni sa „da“ koji su pogrešno prediktovani kao „ne“ (Ima ih 1)

Tačnost = 7/8 = 0.875 (88%)
Preciznost (da) = TP / (TP + FP) = 2/2 = 1.00 (100%)
Odziv (da) = TP / (TP + FN) = 2/3 = 0.67 (67%)

Preciznost (ne) = 5/6 = 0.83 (83%)
Odziv (ne) = 5/5 = 1.00 (100%)

Zadatak za samostalnu vežbu – Majice

★★★☆☆

Na osnovu sledeće tabele podataka o visini i masi osoba odrediti koja veličina majice odgovara označenim osobama na kraju tabele koristeći *knn* algoritam sa Menhetn rastojanjem i vrednošću k=3. Da li bi se rešenje promenilo ukoliko bi se podaci normalizovali?

Visina	Masa	Veličina majice
168	58	S
168	63	S
170	60	S
173	62	M
173	65	M
175	63	M
178	67	L
180	66	L
182	73	L
172	61	?
181	60	?

PITANJA?

http://ri4es.etf.rs/