

Domaći 2 iz Veštačke inteligencije

Nemanja Saveski 2019/0056

2. jun 2023.

Zadatak 1: Potrebno je prvo definisati mrežu za sledeće uslovne raspodele:

A	$P(b^- A)$
a^-	0.7
a^+	0.2

A	$P(c^- A)$
a^-	0.5
a^+	0.7

A	$P(d^- A)$
a^-	0.3
a^+	0.8

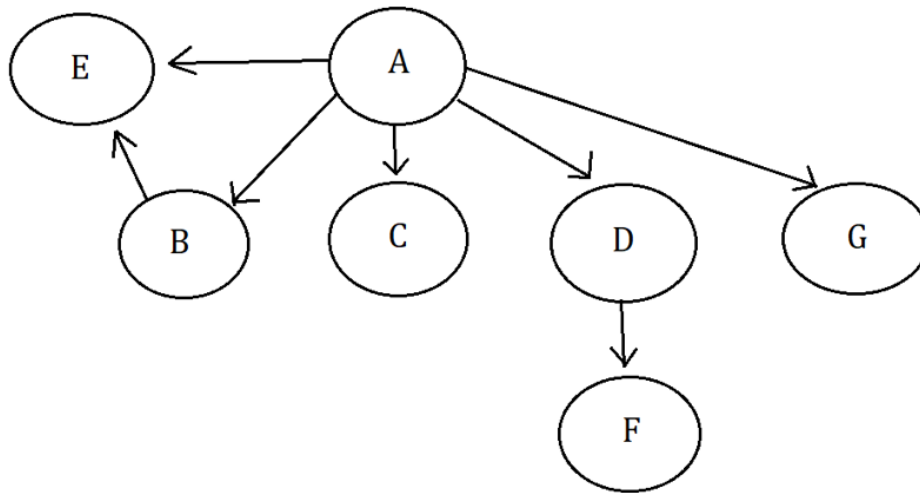
A	$P(g^- A)$
a^-	0.4
a^+	0.6

D	$P(f^- D)$
d^-	0.8
d^+	0.3

$$P(a^-) = 0.3$$

A	B	$P(e^- A, B)$
a^-	b^-	0.6
a^-	b^+	0.5
a^+	b^-	0.3
a^+	b^+	0.8

Mreža izgleda ovako:



Zatim ćemo pronaći uslovnu verovantoću $P(e^- | f^+)$ teorijski, metodom eliminacije. Imamo da je:

$$\begin{aligned}
 P(e^- | f^+) &= \beta P(e^-, f^+) = \alpha P(a, b, c, d, e^-, f^+, g) \\
 &= \alpha \sum_{a,b,c,d,g} P(a)P(b|a)P(e^-|a,b)P(c|a)P(d|a)P(g|a)P(f^+|d) \\
 &= \alpha \sum_{a,b,d} P(a)P(b|a)P(e^-|a,b)P(d|a)P(f^+|d)
 \end{aligned}$$

Iz gornje jednačine su nestale promenljive C i G jer su one "nebitne" promenljive (naime, pri izračunavanju uslovne raspodele neke promenljive, bitno je uzeti samo njene pretke i pretke dokaznih promenljivih, a kako se po slici vidi, ni C ni G nisu preci promenljivih E i F)

Dalje imamo grupisanje po promenljivoj D (gramzivo, sa najmanjim brojem faktora, 2):

$$= \alpha \sum_a \sum_b P(a)P(b|a)P(e^-|a, b) \underbrace{\sum_d P(d|a)P(f^+|a)}_{f_D(a)},$$

gde se dobija $f_D(a^+) = 0.3$ i $f_D(a^-) = 0.55$. Zatim grupišemo po B:

$$P(E|f^+) = \alpha \sum_a P(a)f_D(a) \underbrace{\sum_b P(b|a)P(e|a, b)}_{f_B(e, a)},$$

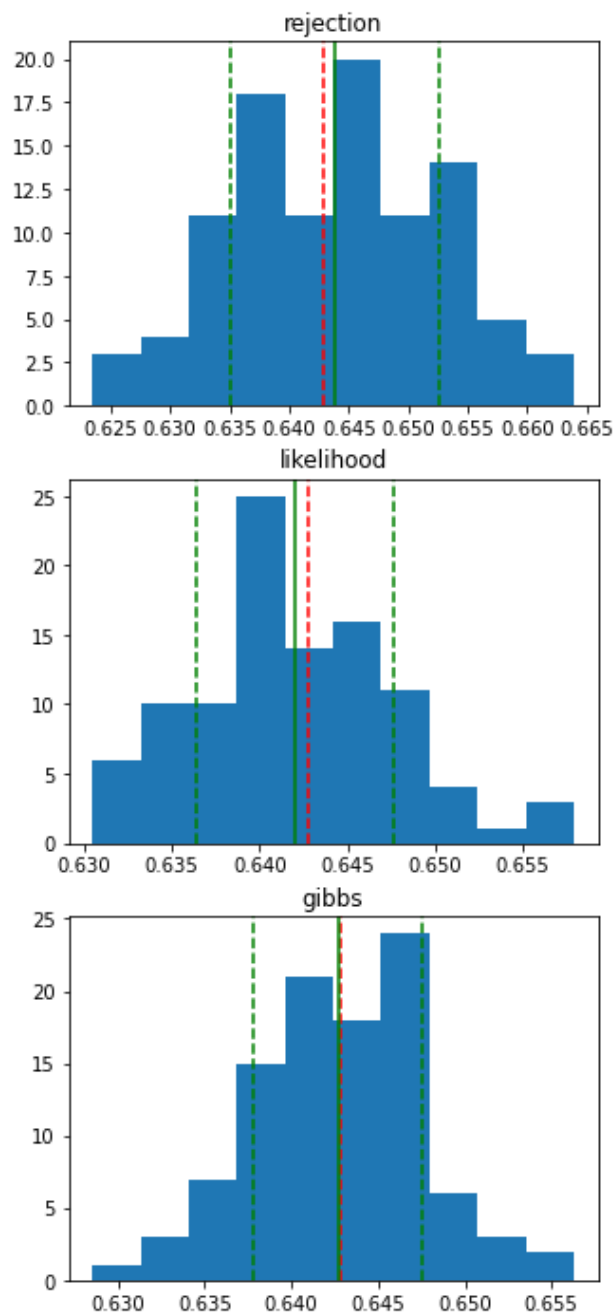
odakle se dobija $f_B(e^-, a^-) = 0.57$, $f_B(e^-, a^+) = 0.7$, $f_B(e^+, a^-) = 0.43$, i $f_B(e^+, a^+) = 0.3$. Poslednje imamo eliminaciju promenljive A:

$$= \alpha \underbrace{\sum_a P(a)f_D(a)f_B(e, a)}_{f_A(E)},$$

gde je $f_A(e^+) = 0.13395$, a $f_A(e^-) = 0.24105$. Za α se dobija $1/0.375 = 2.67$, pa konačno, procena za $P(e^-|f^+)$ iznosi **0.6428**. Dalje, potrebno je proceniti traženu verovatnoću Monte Karlo metodama: uzorkovanje sa odbacivanjem, metoda ponderisanja verodostojnošću, Gibbsovo uzorkovanje. Uzorkovanje sa odbacivanjem se vrši tako što procenjujemo broj povoljnih ishoda uzimajući samo one uzorke, gde dokazne promenljive uzimaju tražene vrednosti, i iz tog skupa uzoraka dalje uzimamo one gde upitne promenljive uzimaju tražene vrednosti. Dakle, ako nađemo na neki uzorak gde nam se prvobitno ne poklapaju vrednosti dokaznih promenljivih sa željenim vrednostima, taj uzorak odbacujemo. Metod ponderisanja verodostojnošću radi na sličan način kao prethodni, ali ovde ne odbacujemo uzorke, nego svakom uzorku dajemo neku težinu na osnovu verodostojnosti, a zatim na osnovu dobijenih težina odredimo ponderisane ishode i konačno dobijamo tražene procene nakon normalizacije ishoda. Gibbsovo uzorkovanje radi na malo drugačiji način. Na početku se nasumično uzorkuju nedokazne promenljive, a zatim se bira jedna po jedna od njih i uzorkuje na osnovu ostalih starih odbiraka, što čini jedan ciklus. Ciklusa ćemo imati onoliko koliko smo mi sami zadali. Posle svakog koraka u ciklusu je napravljen novi uzorak (tj. pre biranja nove promenljive za uzorkovanje). Tabela sa traženim očekivanjima i standardnim devijacijama:

Metod	mean	std
Uzorkovanje sa odbacivanjem	0.6438	0.0088
Ponderisanje verodostojnošću	0.6419	0.0056
Gibbsovo uzorkovanje	0.6427	0.0049

Sve Monte Karlo simulacije su pokrenute 100 puta, sa brojem uzoraka $N = 10000$. Histogrami su prikazani na sledećoj slici:



Zadatak 2: Potrebno je napraviti čestični filter za metu koja se kreće duž fiksiranog pravca, pravolinijski. Brzina je modelovana na sledeći način:

$$v[t + 1] = v[t] + 0.2(\mu - v[t]) + 0.32\epsilon[t],$$

gde svaka trajektorija ima svoje $\mu \sim \mathcal{U}[-2, 2]$, koje je konstantno tokom cele te trajektorije, dalje $\epsilon[t] \sim \mathcal{N}(0, 1)$, a početni uslov $v[0] \sim \mathcal{N}(0, 1)$. Pozicija u svakom trenutku se dobija na sledeći način:

$$x[t + 1] = x[t] + v[t]$$

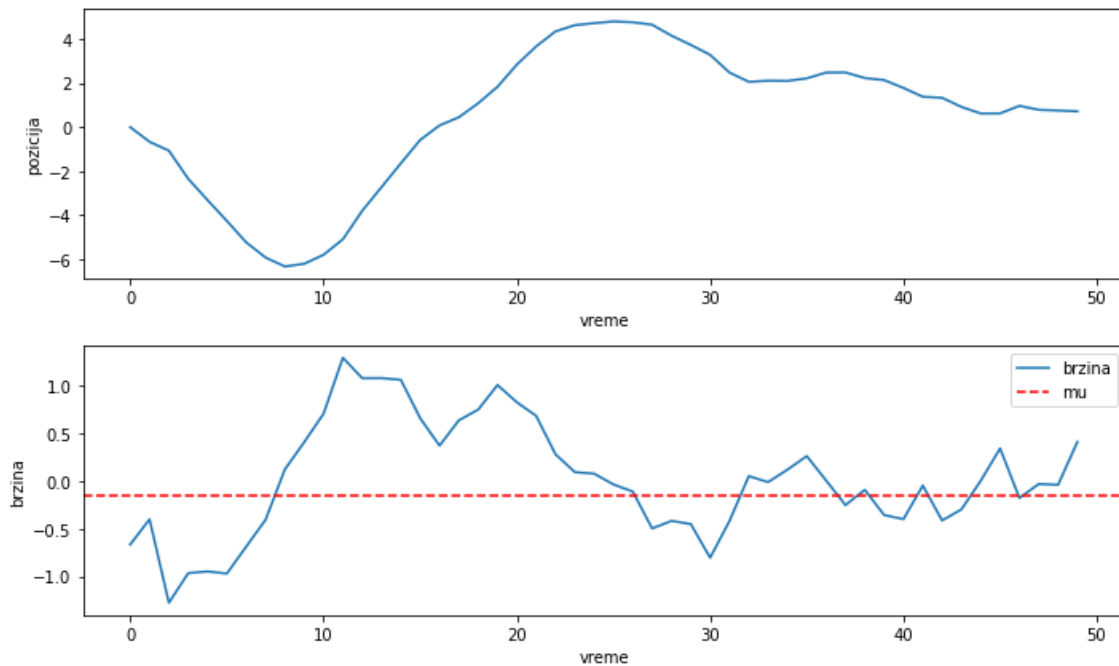
Model merenja je dat sa:

$$e[t] = (-1)^{\theta[t]}x[t] + n[t]$$

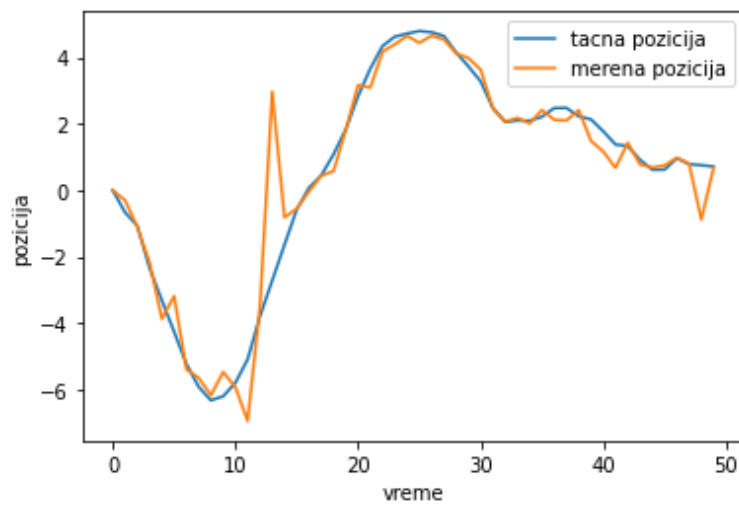
gde je $\theta[t] \sim \text{Bernoulli}(0.05)$ (u 5% slučajeva dolazi do promene znaka $x[t]$), a n , šum merenja je predstavljen kao beli Laplaceov proces:

$$p(n|x) = \frac{1}{2b(x)}e^{-\frac{|n|}{b(x)}}, \quad b(x) = \frac{\sqrt{|x|}}{5}$$

Prvo je potrebno simulirati jednu trajektoriju i prikazati pozicije tokom te jedne trajektorije:



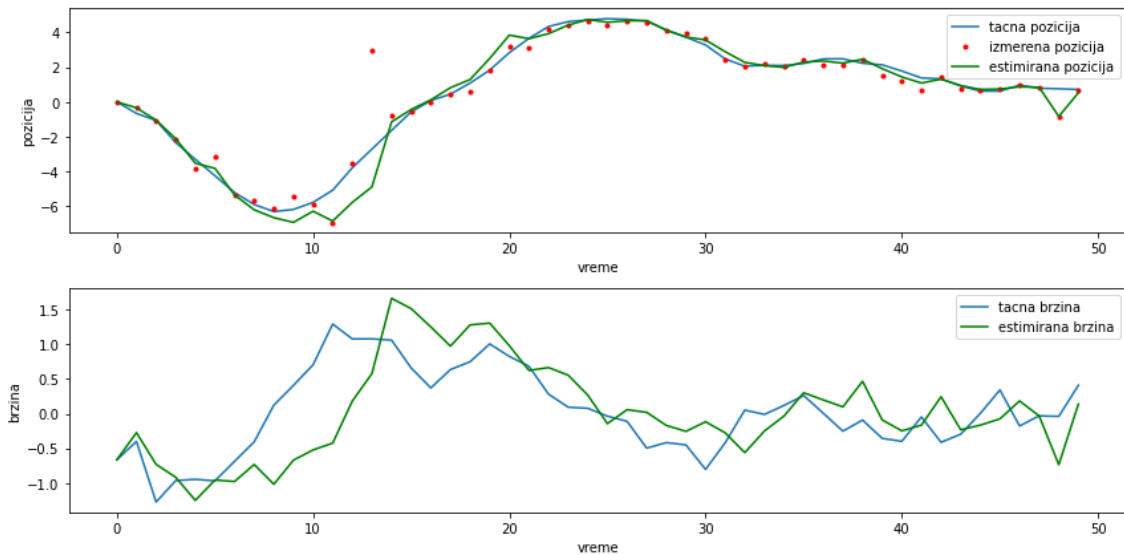
Zatim na osnovu simulirane trajektorije (tačnih pozicija), napravićemo opservacije:



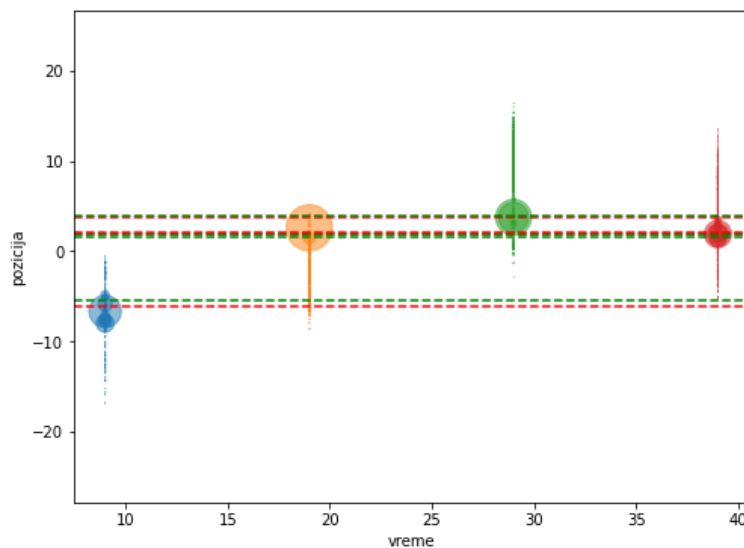
I konačno implementirati čestični filtar. Uzeto je $N = 1000$ čestica, gde je početna težina svake čestice iznosila $1/N$, a zatim, u svakoj iteraciji, se menjala po zakonu (za i -tu česticu):

$$w[t + 1] \propto w[t]p(\text{ops}|x)$$

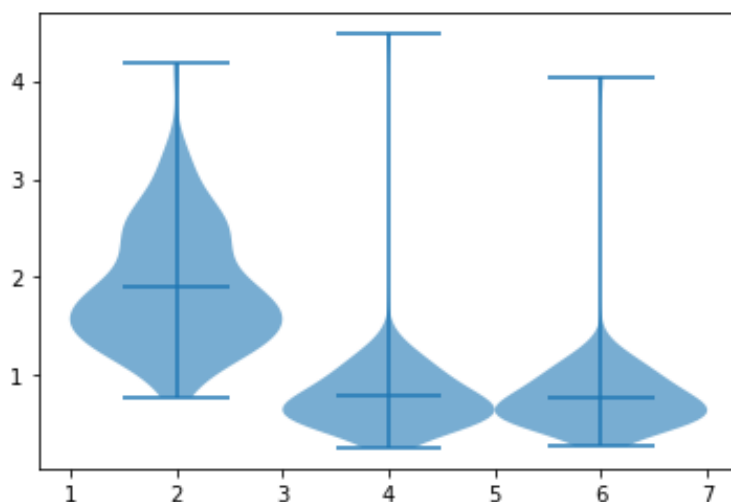
$P(\text{ops}|x)$ predstavlja beli Laplaceov proces dat u zadatku $p(n|x)$, samo smo iskoristili da je $n[t] = \text{ops}[t] - x[t]$ i zamenili u datu raspodelu. Novodobijene težine normalizujemo posle računanja istih. Dobijamo sledeće rezultate:



Treba napomenuti da je za prethodni zahtev urađeno reuzorkovanje u vremenskim trenucima $t = [10, 20, 30, 40]$, kako bi rešenja u sledećem zahtevu zadatka bila konzistentna sa rešenjima iz prošlih zahteva. Grafik čestica u različitim trenucima, pre nego što su reuzorkovane, gde je veličina određene čestice proporcionalna svojoj težini u datom trenutku:



Poslednje što treba ispitati je kako različite metode reuzorkovanja (bez reuzorkovanja, reuzorkovanje u svakoj iteraciji i uslovno reuzorkovanje) utiču na koren srednje kvadratne greške. Na sledećem grafiku s leva na desno su predstavljene redom navedene metode, kao raspodele korena srednje kvadratnih grešaka sa označenim srednjim vrednostima i ekstremnim vrednostima (minimum i maksimum (outlieri)):



Možemo primetiti da se dobijaju slične vrednosti za srednju RMSE druge dve metode, dok bez reuzorkovanja imamo veću srednju RMSE, takođe postoje outlieri u metodama sa reuzorkovanjem.