



Univerzitet u Beogradu -Elektrotehnički fakultet
Katedra za Signale i sisteme



Sistemi odlučivanja u medicini

- Projektni zadatak -

Neuralne mreže 1

STUDENTI

**Kristina Nikolić 2018/0245,
Marko Macura 2018/0261**

Januar 2022.

Sadržaj

1	Uvod i baza podataka	3
2	Podela podataka	4
3	Formiranje neuralne mreže	5
3.1	Kriterijumska funkcija	5
3.2	Aktivacione funkcije	5
3.3	Optimizacioni algoritam	6
4	Kros validacija	6
4.1	Merenje performansi	6
5	Finalni model	7
5.1	Dodatak	8

1 Uvod i baza podataka

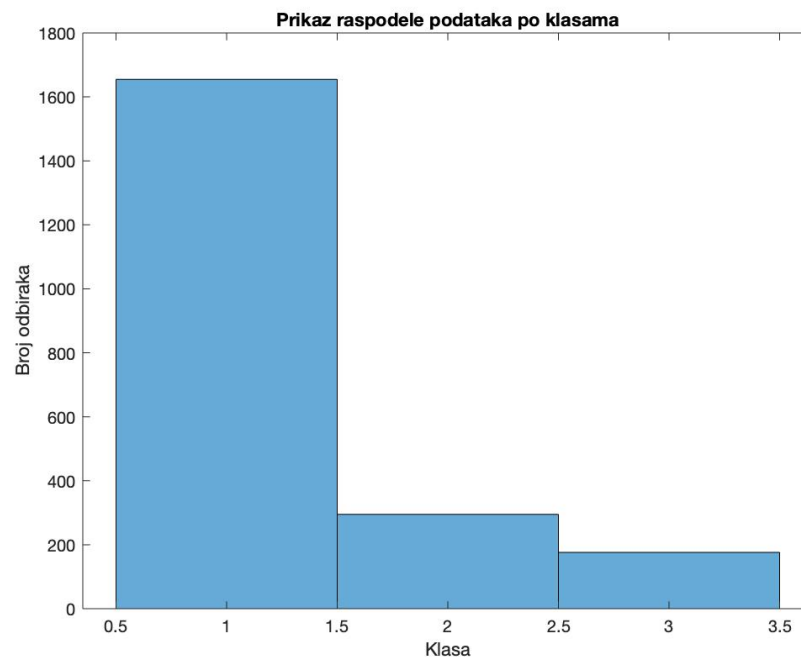
U ovom projektu korišćena je baza podataka: "Fetal Cardiotocography Data". Ovaj set podataka sadrži obeležja koja predstavljaju parametre dobijene procesiranjem 2126 kardiograma fetusa. Ova metoda se primarno koristi za praćenje otkucaja srca fetusa tokom trudnoće. Snimljeni podaci su takođe obrađeni od 3 eksperta akušera koji su snimke klasifikovali po moršfološkim obrascima na 10 klasa i na osnovu stanja fetusa na standardno, sumnjivo i patološko stanje. Ovaj set se može koristiti za klasifikaciju na spomenutih 10 klasa, kao i na spomenute 3 (standardno, sumnjivo i patološko). U ovom projektu formiraćemo veštačku neuralnu mrežu za klasifikaciju na 3 klase.

Obeležja data u bazi podataka su:

- b - instant start
- e - instant stop
- LBE - osnovna vrednost (medicinski stručnjak)
- LB - osnovna vrednost (SisPorto)
- AC - ubrzanja (SisPorto)
- FM - kretanje fetusa (SisPorto)
- UC - kontrakcija materice (SisPorto)
- ASTV - procenat vremena sa abnormalnom kratkorocnom varijabilnošću (SisPorto)
- mSTV - srednja vrednost kratkoročne varijabilnosti (SisPorto)
- ALTV - procenat vremena sa abnormalnom dugotrajnom varijabilnošću (SisPorto)
- mLTV - srednja vrednost dugoročne varijabilnosti (SisPorto)
- DL - blaga usporavanja
- DS - nagla usporavanja
- DP - produžena usporavanja
- DP - ponovljena usporavanja
- Width - širina histograma
- Min - minimalna frekvencija histograma
- Max - maksimalna frekvencija histograma
- Nmax - broj pikova histograma
- Nzeros - broj nula na histogramu
- Mode - moda histograma
- Mean - srednja vrednost histograma
- Median - median histograma
- Variance - varisansa histograma
- Tendency - tendencija histograma: -1 - na levo, 1 - na desno, 0 - simetrično
- A - miran san

- B - REM san
- C - smirena budnost
- D - aktivna budnost
- SH - promenjeljiv obrazac (A ili SUSP sa promenama)
- DE - usporavajući obrazac (vagalna stimulacija)
- LD - dominantno usporavajući obrazac
- FS - ravno-sinusoidalno obrazac (patološko stanje)
- SUSP - sumnjiv obrazac
- CLASS - kod (1-10) za klase od A to SUSP
- NSP - Normalno stanje: 1, Sumnjivo stanje: 2, Patološko stanje: 3

U skupu podataka se nalazi 2126 odbiraka koji su potpuni. Klasi 1 (normalno stanje) pripada 1655 odbiraka, klasi 2 (sumnjivo stanje) pripada 295 odbiraka i klasi 3 (patološko stanje) pripada 173 odbiraka. Primećujemo da skup nije balansiran i da je prva klasa znatno dominantija u odnosu na ostale dve. Na slici 1 prikazana je raspodela odbiraka po klasama.



Slika 1: Raspodela podataka po klasama. 1 - Normalno stanje; 2 - Sumnjivo stanje; 3 - Patološko stanje

2 Podela podataka

Podatke smo delili na obučavajući, validacioni i testirajući skup, jer smo radili metod ranog zaustavljanja. Podelili smo podatke na 60% za obučavajući, 20% za validacioni i 20% za testirajući skup. Kako podaci nisu balansirani bilo je neophodno prvo separatisati sve podatke kako bismo imali isti udeo svih ishoda u sva tri skupa. To je jako bitno kako bi se neuralna mreža naučila na sve moguće ishode.

Najviše podataka smo dali obučavajućem skupu jer na tom skupu neuralna mreža uči. Na osnovu izlaza iz ovog skupa ona menja težine.

Validacioni skup je tu da bi se sprečilo preobučavanje neuralne mreže. Sve dok greška na validacionom skupu podataka opada, znamo da nije došlo do preobučavanja. Onog trenutka kada rezultati na validacionom skupu postaju lošiji određeni broj epoha, to je znak za pretreniranost i neuralna mreža prestaje da se trenira i za izlaz se uzima mreža sa najboljim rezultatima na validacionom skupu.

Testirajući skup podataka je za testiranje da li nam neuralna mreža daje dobre rezultate i u opštem slučaju (da li dobro generalizuje), ne samo na trenirajućem skupu podataka ili validacionom koji na indirektan način učestvuje u obučavanju neuralne mreže (obučavanje prestaje kad su rezultati na validacionom najbolji). Ukoliko su nam rezultati loši na testirajućem skupu, naš model se ne može upotrebljavati u upštem slučaju.

3 Formiranje neuralne mreže

Ulazi u neuralnu mrežu su formirani kao matrica AXB gde je A broj atributa $A = 35$, a B broj odbiraka $B = 2126$. Ova matrica je kasnije podeljena na trenirajući, testirajući i validirajući skup na način objašnjen u prethodnoj tački i korišćenjem odgovarajućih parametara *patternet* mreže u MatLab-u. Pošto vršimo klasifikaciju na 3 klase potrebno je da izlaz predstavimo u obliku OneHot encoding-a. Formiramo matricu $3 \times B$ gde će se u odgovarajućem redu matrice nalaziti 1 u zavisnosti od klase konkretnog odbirka. Na ovaj način, paket *patternet* takođe prepoznaje da će naša mreža imati 3 izlazna neurona.

3.1 Kriterijumska funkcija

Za kriterijumsku funkciju koristili smo *Cross – Entropy*. Kriterijumska funkcija nam govori o tome koliko mreža greši prilikom klasifikacije. Zbog toga, mi želimo da minimizujemo ovu funkciju promenama pojačanja mreže. *Cross – Entropy* se često koristi u problemima klasifikacije i pogodna je za probleme sa više od 2 klase. Ova funkcija uzima u obzir za koliko je mreža pogrešila i penalizuje velike greške više nego male. Takođe, ova funkcija očekuje one-hot zapis željenog izlaza i verovatnoće koje mreža predviđa za svaku od klasa. Ove verovatnoće dobijamo primenom *softmax* aktivacione funkcije u poslednjem sloju. Kriterijumsku funkciju definišemo na sledeći način:

$$J = -\frac{1}{N} \sum_{i=1}^N p_i \log q_i,$$

gde je N broj odbiraka; p_i je 1 za željenu klasu, inače 0; q_i izlaz iz neurlane mreže za konkretnu klasu.

3.2 Aktivacione funkcije

Za aktivacione funkcije u skrivenim slojevima izabrali smo *tansig* funkciju. Pokušavali smo sa finckijom rampe koja se dosta često koristi, ali je *tansig* dao bolje rezultate. Za skrivene slojeve potrebno je izabrati aktivacionu funkciju koja uvodi nelinearnost kako ne bi sveli celu mrežu na jednu linearnu funkciju koja može da rešava samo linearne separabilne probleme. Bipolarni *tansig* u MatLab-u ima sledeći oblik:

$$\text{tansig}(x) = \frac{2}{1 + e^{-2x}} - 1$$

Za aktivacionu funkciju u izlaznom sloju korišćena je *softmax* funkcija. Ova funkcija se često koristi kod problema sa više klasa jer omogućava da suma izlaza iz krajnjih neurona bude jednaka 1 i da njihove vrednosti budu u opesegu od 0 do 1. Tada, izlaze iz odgovarajućih neurona možemo da posmatramo kao verovatnoće sa kojima mreža tvrdi da konkretni odbirak pripada određenoj klasi. Takođe, izlaz iz svakog neurona normalizovan je sumom eksponenata svih izlaza i time se svi izlazni neuroni dovode u vezu jedni s drugim. Ovakav izlaz je kasnije pogodan za određivanje vrednosti *Cross – entropy* kriterijumske funkcije.

3.3 Optimizacioni algoritam

Kao optimizacioni algoritam koristili smo propagaciju u nazad sa momentumom. Ovaj algoritam se razlikuje od standardne metode propagacije u nazad po tome što pored izvoda kriterijumske funkcije po težinama mreže uzima u obzir i pravac kretanja ka minimumu iz prethodnog koraka. Promenu pojačanja možemo definisati sledećim izrazom:

$$\Delta W(t) = -\eta \nabla_w J + \alpha \Delta W(t-1),$$

gde je α momentum, a η konstanta obučavanja.

4 Kros validacija

Kako bismo izabrali najbolje hiperparametre modela primenili smo metodu kros validacije. Model smo trenirali za sve moguće kombinacije 4 različita hiperparametara koji su uzimali vrednosti iz predefinisanih skupova. U ovom koraku, model smo trenirali na trenirajućem skupu i posmatrali njegove performanse na validacionom skupu.

Parametri koje smo varirali su sledeći: arhitektura mreže, konstanta obučavanja, težina klase od interesa i regularizacija.

Arhitektura mreže. Trenirali smo model sa 4 različite arhitekture koje se razlikuju po broju slojeva i po broju neurona u slojevima. Arhitekture koje smo testirali su sledeće: $arh \in \{[10, 5], [12, 6, 3], [4, 5, 6]\}$. Najbolji rezultat dala je arhitektura: $[12, 6, 3]$.

Konstanta obučavanja. Konstanta obučavanja η može značajno da utiče na brzinu konvergencije modela. Velika konstanta obučavanja vodi u bržu konvergenciju, ali može da dovede i do neželjenog preskakanja lokalnog minimuma kriterijumske funkcije zbog velikog koraka. S druge strane, mala konstanta obučavanja neće dovesti do ovog preskakanja, ali značajno usporava konvergenciju modela. Želeli smo da odredimo optimalnu vrednost ovog parametra i u postupku kros validacije on je uzimao vrednosti iz sledećeg skupa: $\eta \in \{0.005, 0.05, 0.5, 1\}$. Najbolje rezultate je dalo $\eta = 0.5$.

Težine za određenu klasu. Pošto je skup podataka nebalansiran, pretpostavili smo da možemo da sprečimo zanemarivanje manje brojnih klasa ukoliko im povećamo važnost. Ovo se radi tako što se dodeli težina željenim klasama koja je veća od jedan i time se mreža više kažnjava ukoliko pogrešno klasifikuje odbirke iz važnije klase. Ovaj postupak se često koristi u medicini kada želimo da imamo najmanji mogući broj lažno negativnih slučajeva. Vrednosti težina smo uzimali iz sledećeg skupa: $w \in \{1, 1.2, 1.5, 2\}$. Najbolji rezultat dala je vrednost $w = 1$.

Regularizacija Poslednji parametar koji smo varirali je konstanta kojom se množi l2 regularizacija pri računanju kriterijumske funkcije. Regularizacija sprečava da se mreža preobuči zbog prevelikih težina jer se njihov rast kažnjava. Vrednosti koje smo varirali su iz skupa: $reg \in \{0.01, 0.1, 0.3\}$. Najbolji rezultat dala je vrednost $reg = 0.01$.

4.1 Merenje performansi

Tokom procesa korsvalidacije pratili smo tačnost modela i F_1 skor za svaku klasu. Tačnost smo definisali na sledeći način:

$$acc = \frac{N_{11} + N_{22} + N_{33}}{N}$$

gde je N_{ii} broj odbiraka koji su tačno klasifikovani u i -tu klasu, a N ukupan broj odbiraka.

Kako su klase nebalansirane informativnije je posmatrati F_1 metriku. Ova metrika daje podjednak značaj svim klasam dok se u slučaju tačnosti može desiti da se ekstremno loša klasifikacija manjebrojnih klasa

zanemari jer dobro klasifikovane brojčano jače klase dovoljno podižu ovu metriku i prikrivaju pomenuti nedostatak modela. F_1 skor smo računali za svaku klasu ponaosob, a zatim ga usrednjili. Sledećim izrazima definisani su preciznost, osetljivost i F_1 skor za svaku od klasa.

$$Prec_i = \frac{N_{ii}}{N_{i1} + N_{i2} + N_{i3}}$$

$$Sens_i = \frac{N_{ii}}{N_{1i} + N_{2i} + N_{3i}}$$

$$F_{1i} = 2 \frac{Prec_i * Sens_i}{Prec_i + Sens_i}$$

gde je N_{ij} broj odbiraka i -te klase klasifikovanih u j -tu klasu, a F_{1i} je F_1 skor računat za i -tu klasu. Konačno, usrednjićemo F_1 metrike za sve tri klase u finalnu metriku koju želimo da maksimizujemo:

$$F_1 = \frac{F_{11} + F_{12} + F_{13}}{3}$$

U slučaju da za ovaj model postoji klasa od interesa, maksimizirali bismo F_1 skor te klase kako bi model imao što bolje performanse pri klasifikaciji odbiraka iz ove klase. Kako u ovom skupu podataka nije naznačena klasa od interesa, već se pridaje podjednak značaj svim klasama, nismo izdvajali jednu od klasa.

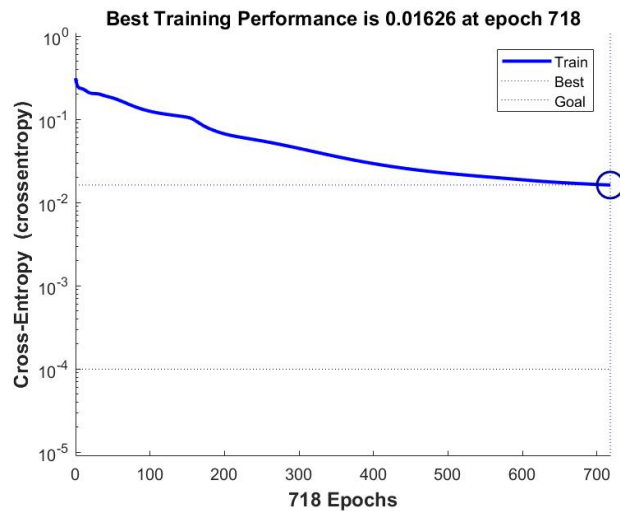
5 Finalni model

Kao rezultat kros-validacije odabrani su parametri prikazani u tabeli 1:

Parametar	Vrednost
learning rate	0.5
weights	1
architecture	[12 6 3]
regularization	0.01

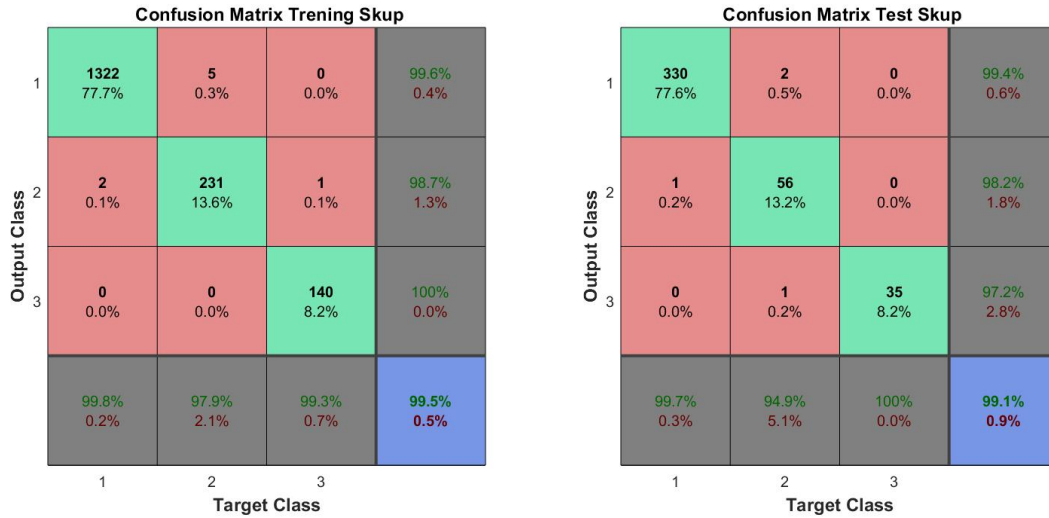
Tabela 1: Parametri modela.

Grafik performansi modela kroz epohe prikazan je na slici 2.



Slika 2: Performanse finalnog modela kroz epohe.

Na slikama 3 prikazane su konfuzione matrice modela na trenirajućem i testirajućem skupu.



Slika 3: Konfuzione matrice na trening skupu (levo) i na test skupu (desno).

F_1 skor je na trening skupu 0.9923 dok je na test skupu 0.9823. Ovaj parametar je izuzetno visok. Takodje, primećujemo da model na oba skupa podataka ima najmanju preciznost i senzitivnost za 2. klasu. Tj. često se dešava da odбирke 2. klase model klasifikuje u 1. klasu. Ukoliko nam je ova klasa od izuzetnog značaja, možemo popraviti performanse modela za ovu klasu ukoliko prilikom kros validacije posmatramo F_1 skor samo za klasu 2. Pojedinačne vrednosti za preciznost, senzitivnost i F_1 skor su prikazane u tabelama 5 i 6.

Klasa	Preciznost	Senzitivnost	F_1
1	0.9962	0.9985	0.9974
2	0.9872	0.9788	0.9830
3	1.0	0.9925	0.9964

Tabela 2: Preciznost, senzitivnost i F_1 skor po klasi na trening skupu.

Klasa	Preciznost	Senzitivnost	F_1
1	0.9940	0.9970	0.9955
2	0.9825	0.9492	0.9655
3	0.9722	1.0	0.9859

Tabela 3: Preciznost, senzitivnost i F_1 skor po klasi na test skupu.

5.1 Dodatak

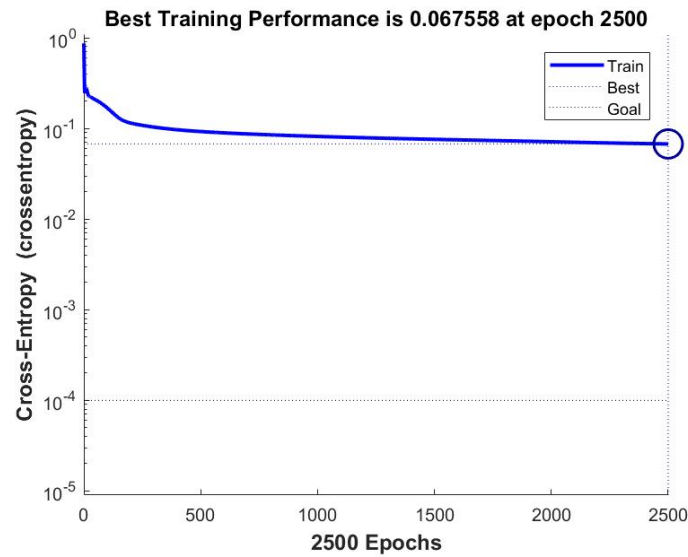
Kako je ovaj skup podataka sadrži podelu na 10 klasa i podelu na 3 klase koje ne predstavljaju širu i užu podelu istih karakteristika, do sada smo uzimali u obzir podatke klasifikaciju na 10 klasa kako bismo klasifikovali podatke u neutralan, sumnjiv i patloški slučaj. U ovom dodatku želeli smo da treniramo model koji neće imati informacije o podeli na 10 klasa pri odlučivanju izlaza. Ponovili smo ceo postupak kros-validacije za isti skup parametara, iste aktivacione funkcije i isti optimizacioni algoritam. Rezultate ovakvog treniranja prikazujemo u nastavku.

Kao rezultat kros-validacije odabrani su parametri prikazani u tabeli 4:

Parametar	Vrednost
learning rate	0.5
weights	1
architecture	[10 5]
regularization	0.01

Tabela 4: Parametri modela.

Grafik performansi modela kroz epohe prikazan je na slici 4.



Slika 4: Performanse finalnog modela kroz epohe.

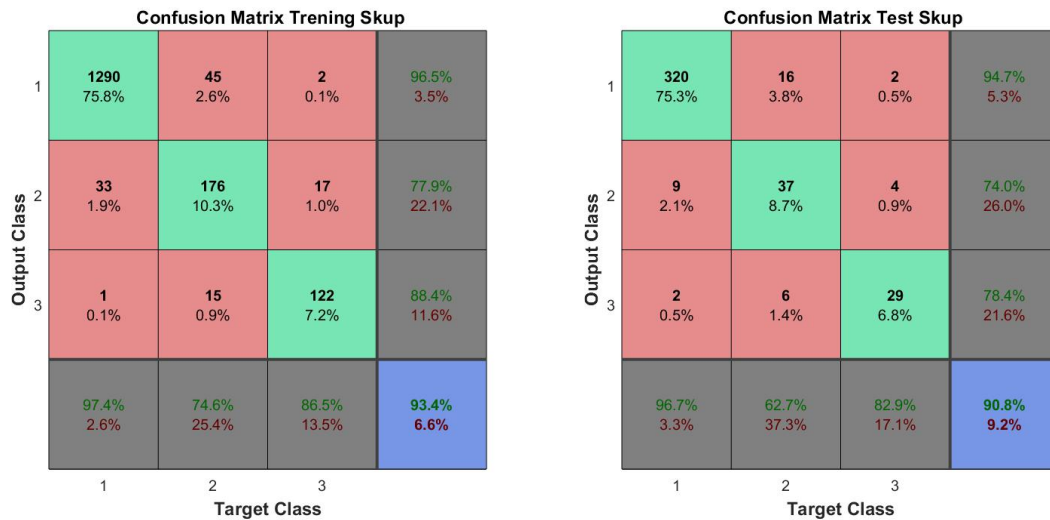
Na slikama 5 prikazane su konfuzione matrice modela na trenirajućem i testirajućem skupu.

Klasa	Preciznost	Senzitivnost	F_1
1	0.9648	0.9743	0.9696
2	0.7788	0.7458	0.7619
3	0.8841	0.8652	0.8746

Tabela 5: Preciznost, senzitivnost i F_1 skor po klasi na trening skupu.

Klasa	Preciznost	Senzitivnost	F_1
1	0.9467	0.9668	0.9567
2	0.7400	0.6271	0.6789
3	0.7828	0.8286	0.8056

Tabela 6: Preciznost, senzitivnost i F_1 skor po klasi na test skupu.



Slika 5: Konfuzione matrice na trening skupu (levo) i na test skupu (desno).

Primećujemo da je tačnost ovakvog modela dosta niža. Zbog toga zaključujemo da je prethodna podela na 10 klasa od strane stručnjaka izuzetno značajna pri podeli na naše 3 klase, tj. poslednjih 10 atributa u našem skupu podataka su imali veliku važnost i značajno poboljšali performanse modela. U ovom slučaju se takođe dešava da se veliki broj odbiraka 2. klase klasifikuje u 1., ali su performanse modela da tačno klasifikuje 3. klasu takođe značajno smanjene.