Универзитет у Београду
Електротехнички факултет

Препознавање облика

Домаћи задатак

Ђорђе Маљковић 2017/0256

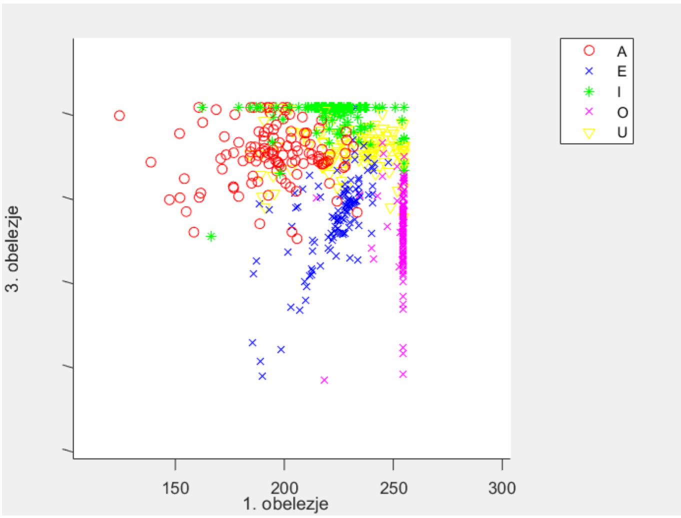
Београд, Август 2022.

1. Први задатак

1.1. За базу руком писаних самогласника пројектовати иновативни систем за препознавање свих пет самогласника заснован на теорији хипотеза. Резултате приказати у облику матрице конфузије. Извештај треба да садржи кратки опис пројектованог система, образложен избор обележја, као и карактеристичне примере правилно и неправилно класификованих слова.

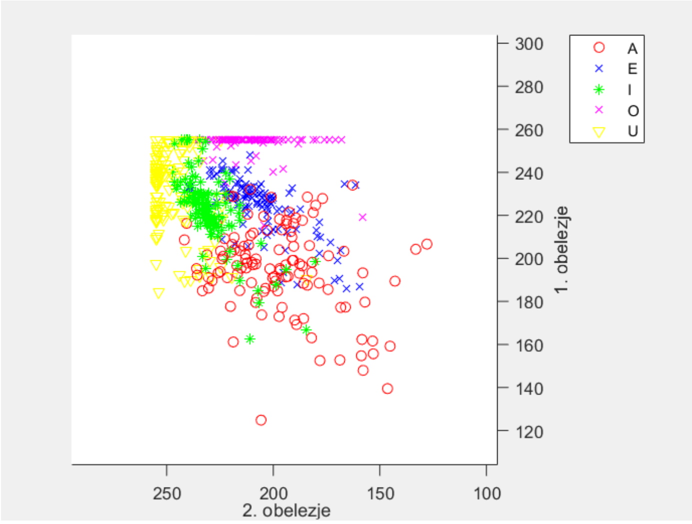
База слова састоји се од по 120 примерака ручно писаних самогласника ћириличним писмом. Након учитавања самогласника било је потребно да се изврши предобрада слике, како би се слика бинаризовала, то јест ако вредност пиксела има вредност мању од датог прага, подесити његову вредност на 0 (постају црни), а ако је вредност већа од датог прага, подесити његову вредност на 255 (постају бели). За праг је узета вредност од 80% максималног интезитета боје пиксела. Следећи корак у предобради је уклањање црних оквира који се налазе на сликама, као и евентуалне беле линије које се налазе на црним оквирима. За крај обраде слике, треба уклонити и све белине око слова како бисмо добили облик који се налази на целој површини слике, и тако избацили зависност процене од величине написаног слова.

Након предобраде слике, потребно је изабрати обележја. Обележја ће бити приказана у тродимензионалном простору, како би грешка при одлучивању била мања.



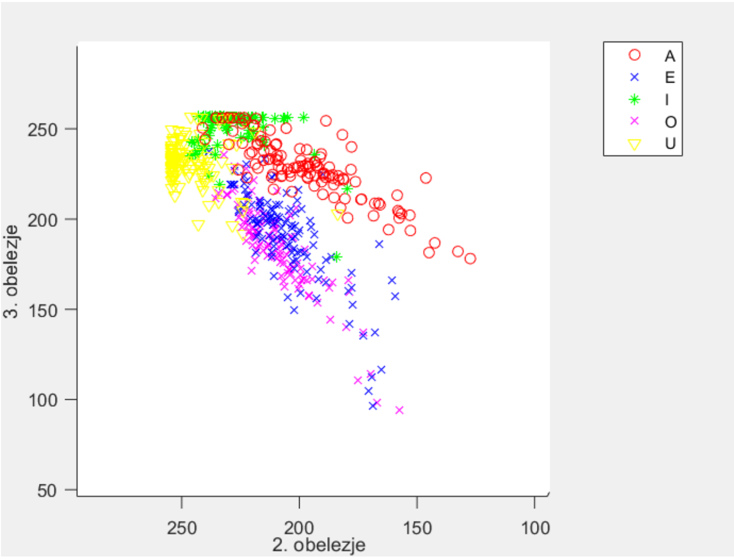
Слика 1. Прво обележје

Прво обележје представља средњу вредност интезитета пиксела средњег квадрата слике. Са слике се виде да се највише издваја слово **О** (максимална вредност) и делимично издваја слово **А** (минимална вредност).



Слика 2. Друго обележје

Друго обележје представља средњу вредност интезитета пиксела на два сегмента истовремено. Два посматрана сегмента је прва петина слике гледајући вертикално, а други сегмент је горња лева половина слике. Ово обележје преставља посебно одваја слово **У**, које у та два сегмента има белине (зато и има вредност обележја 255).



Слика 3. Треће обележје

Треће обележје представља средњу вредност интезитета пиксела на два сегмента истовремено, овог пута горња средња хоризонтална седмина и доња средња хоризонтална седмина. Од свих слова, слово **И** садржи највише белина на овом сегменту, као и слово **А** делимично. Такође се и слово **У** са малом мањим вредностима издваја у оквиру неког интервала вредности.

Након одређивања обележја, потребно је испројектовати систем одлучивања путем теста више хипотеза. Прво се скуп подели на обучавајући и тестирајући. Првих 100 одбирака сваког слова је узето као обучавајући скуп и на основу тих одбирака се рачуна средња вредност и коваријациона матрица. Узето је да су расподеле нормалне. Преосталих 20 одбирака се користи као тестрајући скуп, тако што за сваки, тако што се за свако слово одреди функција густине вероватноће, па се донесе одлука којој класи припада одбирак, на основу тога где функција густине вероватноће има максималну вредност. На основу одабраних класа, и стварних припадности класа, добија се матрица конфузије.

```
Mk =

    19     0     1     0     0
     0    20     0     0     0
     1     0    13     0     6
     0     0     0    20     0
     0     0     0     0    20

Greska iznosi: 0.08
```

Слика 4. Матрица конфузија

Слово које је имало највише погрешних класификација јесте слово **И** (чак 7). На основу неких одбирака, може се закључити да неки примерци слова **И** личе на слово **А** и слово **У**, због чега су и класификовани као такви.

2. ДРУГИ ЗАДАТАК

2.1. Генерисати по 500 одбирака из двеју дводимензионалнх бимодалних класа. Параметре класа самостално изабрати.

На дијаграму приказати одбирке.

Генерисати геометријско место тачака са константном вредношћу функција густина вероватноће, па их приказати на дијаграму у простору облика.

Пројектовати Бајесов класификатор минималне грешке и на дијаграму, заједно са одбирцима, скицирати класификациону линију, па проценити вероватноћу грешке.

Поновити претходну тачку за други класификатор.

За класе облика генерисаних у претходним тачкама, пројектовати Wald-ov секвенцијални тест па скицирати зависност броја потребних одбирака од усвојене вероватноће грешака првог, односно другог типа.

$$\Omega_1 \sim P_{11} \cdot N(M_{11}, \Sigma_{11}) + P_{12} \cdot N(M_{12}, \Sigma_{12}),$$

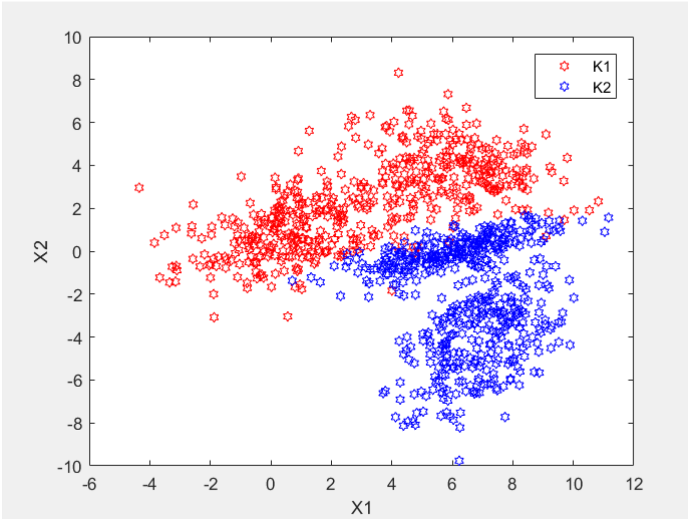
$$\Omega_2 \sim P_{21} \cdot N(M_{21}, \Sigma_{21}) + P_{22} \cdot N(M_{22}, \Sigma_{22}).$$

Параметри класа су дати у наставку:

$P_{11} = 0.6$, $P_{12} = 0.4$, $M_{11} = \begin{bmatrix}1\\1\end{bmatrix}$, $\Sigma_{11} = \begin{bmatrix}4 & 1.1\\1.1 & 2\end{bmatrix}$, $M_{12} = \begin{bmatrix}6\\4\end{bmatrix}$, $\Sigma_{12} = \begin{bmatrix}3 & -0{,}8\\-0.8 & 1.5\end{bmatrix}$

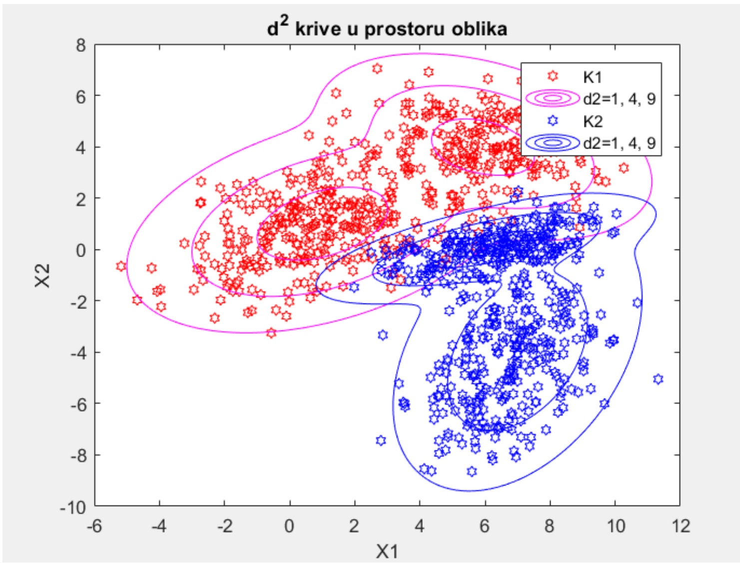
$P_{21} = 0.55$, $P_{22} = 0.45$, $M_{21} = \begin{bmatrix}7\\-4\end{bmatrix}$, $\Sigma_{21} = \begin{bmatrix}2 & 1.1\\1.1 & 4\end{bmatrix}$, $M_{22} = \begin{bmatrix}6\\0\end{bmatrix}$, $\Sigma_{22} = \begin{bmatrix}3 & 0.8\\0.8 & 0.5\end{bmatrix}$

Генерисани одбирци су приказани на наредној слици:



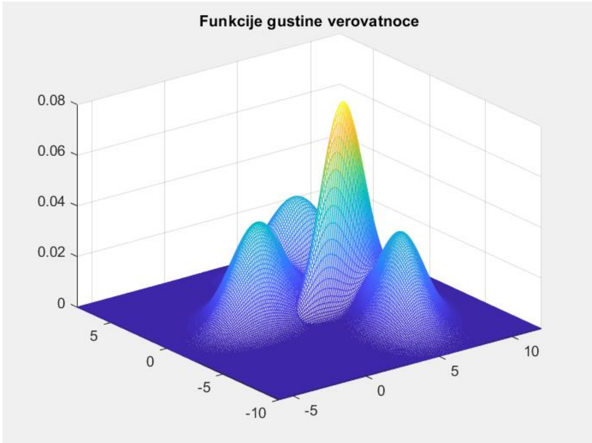
Слика 5. Одбирци две бимодалне класе

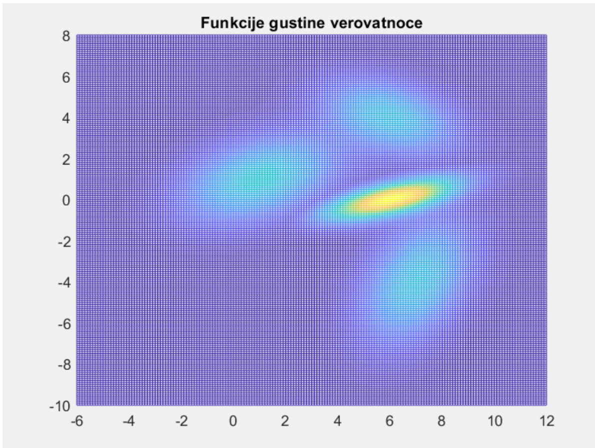
Следећа ставка у задатку је била да се генерише геометријско место тачака са константном вредношћу функција густине вероватноће (d^2 криве) и да се прикажу на дијаграму у простору облика.



Слика 6. d^2 криве у простору облика

Такође, приказане су и функције густине вероватноће датих класа.





Слика 7. Функције густине вероватноће

Следећи захтев задатка је да се пројектује Бајесов класификатор минималне грешке. Први корак је да се пројектује дискриминациона функција:

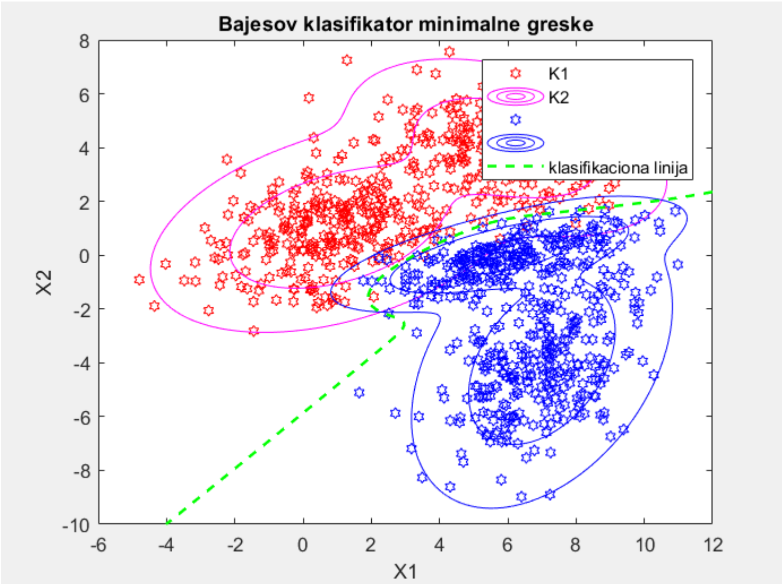
$$h(x) = -\,ln\frac{f_1(x)}{f_2(x)}$$

Уколико је њена вредност већа од одређеног прага, онда одбирак класификујемо у другу класу, а ако је вредност мања од одређеног прага, онда одбирак класификујемо у прву класу. Праг се одређује као:

$$T = -\,ln\frac{P_2}{P_1}$$

P1 и P2 представљају априорне вероватноће појављивања одбирака из појединих класа и с обзиром на то да имамо подједнак број одбирака из обе класе, усвојићемо да су ове вредности једнаке и износе по 0.5, па је праг једнак нули. и износе по 0.5, па је праг једнак нули.

Овакав класификатор је представљен на следећем графику:



Слика 8. Бајесов класификатор минималне грешке

Вероватноћа грешке се одређује по формули:

$$\varepsilon = P_1 \cdot \varepsilon_1 + P_2 \cdot \varepsilon_2$$

ε_1 представља вероватноћу грешке првог типа (вероватноћа да одбирак из прве класе буде погрешно класификован), а ε_2 представља вероватноћу грешке другог типа (вероватноћа да је одбирак из друге класе погрешно класификован). Ове две грешке се процењују методом процене помоћу израза:

$$\varepsilon_1=\sum_i \sum_j f_1(i,j)\Delta x\Delta y$$
$$\varepsilon_2=\sum_i \sum_j f_2(i,j)\Delta x\Delta y$$

Добијене вредности су :

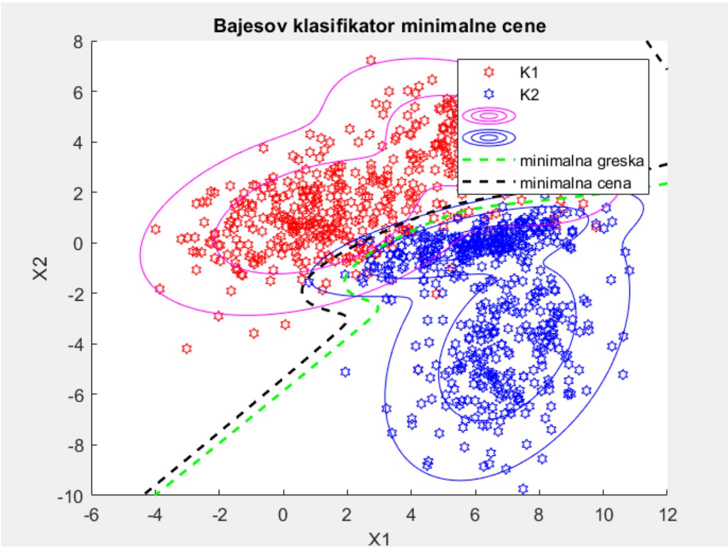
```
Eps1 =

    0.0285


Eps2 =

    0.0170


Eps =

    0.0228
```

Следећи класификатор који се пројектује је Бајесов класификатор минималне цене. Разлика у односу на претходни класификатор је та што сада праг одлуке не зависи само од априорних вероватноћа појављивања датих класа, већ и од цена које се претходно одређују. Праг и унапред изабране цене су:

$$T = -\ln\frac{(c_{12}-c_{22})P_2}{(c_{21}-c_{11})P_1}$$

$c_{11}=0$, $c_{12}=0.9$, $c_{22}=0$ i $c_{21}=0.1$

Пошто је знатно већа цена која је одабрана за погрешну класификацију одбирака из друге класе, класификациона линија ће се померити даље у односу на другу класу, како би се обухватило што више њених примерака.



Слика 9. Бајесов класификатор минималне цене

У овом случају, грешке износе:

```
Eps1 =

    0.0659


Eps2 =

    0.0030
```

У односу на претходни класификатор, сада је вероватноћа грешке другог типа значајно мања јер је постављена висока цена за њене одбирке, али се сходно томе повећава грешка првог типа, што је било и очекивано.

Последњи захтев овог задатка је пројектовање Wald-овог секвенцијалног теста, где се секвенцијално обрађују одбирци и одређује којој класи они припадају, тако да вероватноћа грешке буде што мања, али и да се што пре донесе одлука, уз што мањи број потребних одбирака. Овај тест се примењује на дате класе тако што се 500 пута понављају следећи кораци:

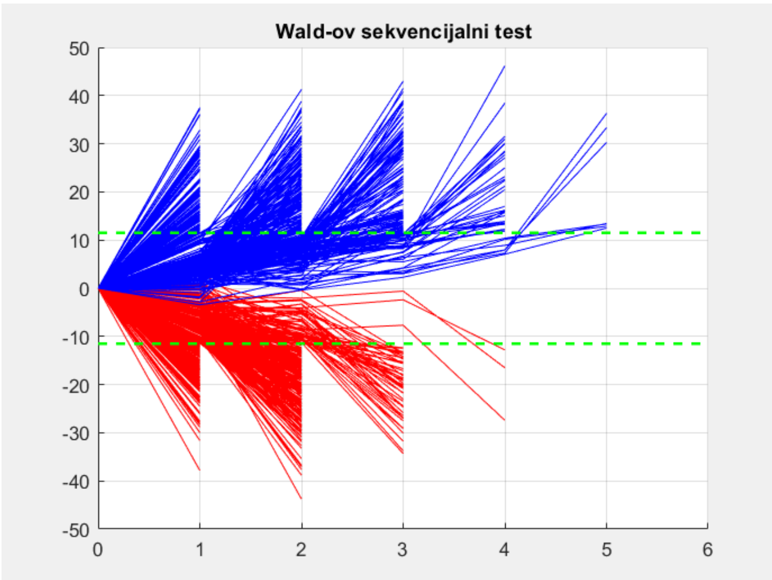
1. На случајан начин се помешају сви одбирци.
2. Узима се један по један одбирак.
3. Доноси се одлука на основу збирне дискриминационе функције Sm, коју упоређујемо са праговима a и b. Те вредности прагова се рачунају помоћу унапред одређених вредности за вероватноћу грешке првог и другог типа. Узето је да вредности грешке износе 10^{-5}.

Sm>b , odbirci pripadaju drugoj klasi
Sm<a, odbirci pripadaju prvoj klasi
Sm∈[a, b] , uzimamo sledeći odbirak

$$Sm=\sum_{i=1}^{m} h(X_i)\ ,\ m=1, 2...$$

$$a = -\ln\frac{1-\varepsilon_1}{\varepsilon_2},\ b = -\ln\frac{\varepsilon_1}{1-\varepsilon_2}$$

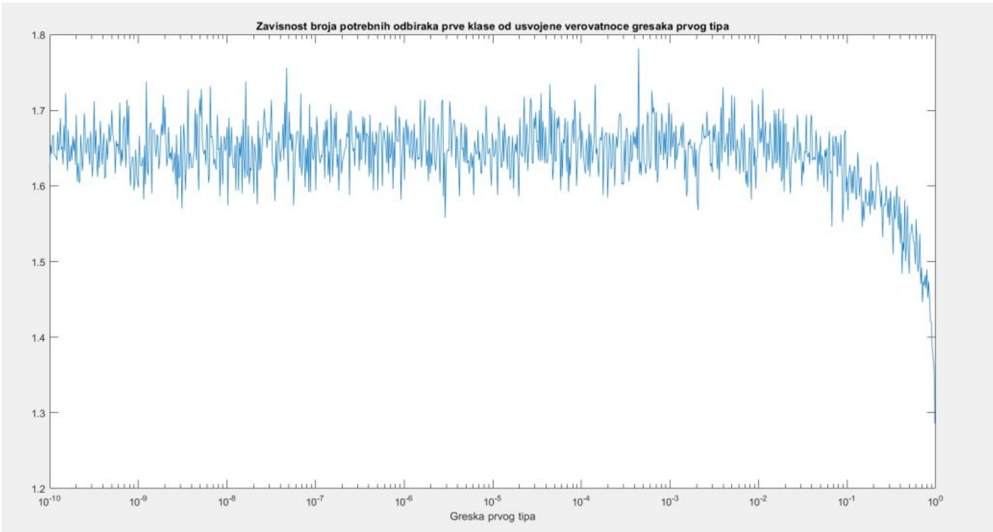
Резултат овог секвенцијалног теста се налази у наставку:



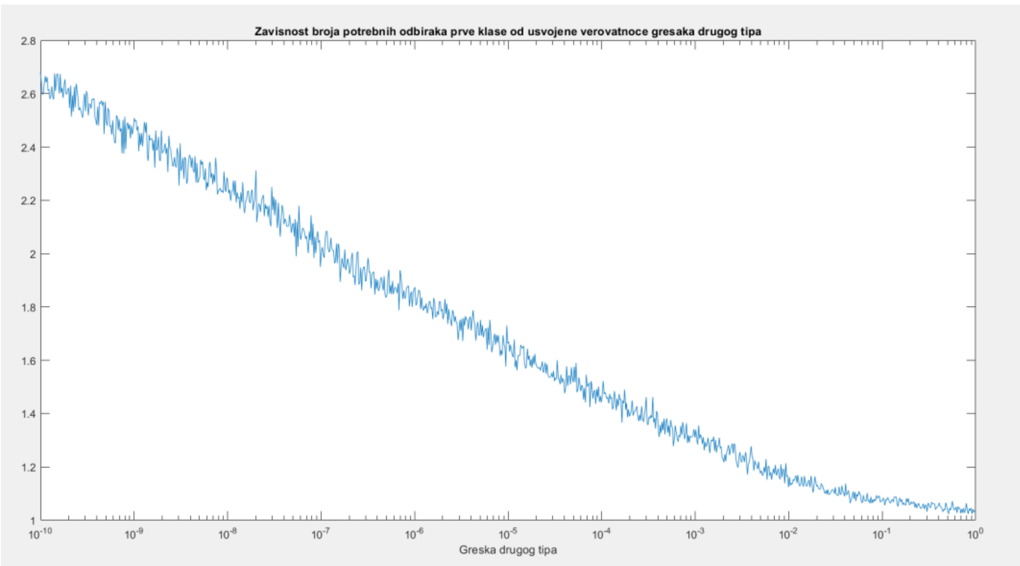
Слика 10. Wald-ов секвенцијални тест

Црвеном бојом су представљени одбирци прве класе, а плавом одбирци друге класе. Доња испрекидана линија је праг a, горња испрекидана линија је праг b. Са слике се види да је потребно углавном 3 одбирка да се донесе одлука о класификацији, док се некад деси да се алгоритам заустави после 5 одбирака.

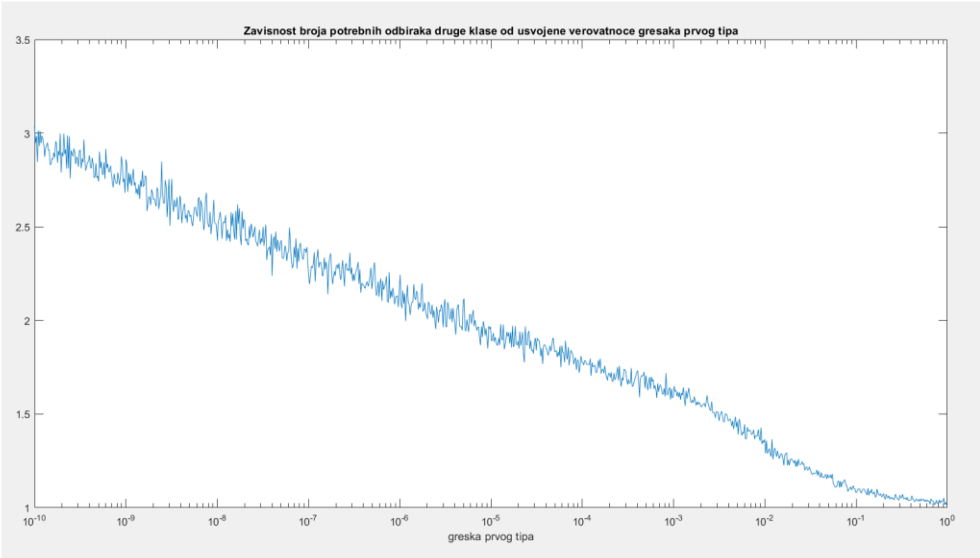
Последња ствар коју треба урадити у овом задатку је да се графички престави број потребних одбирака дате класе од усвојене вероватноће грешке првог и другог типа, у логаритамској размери:



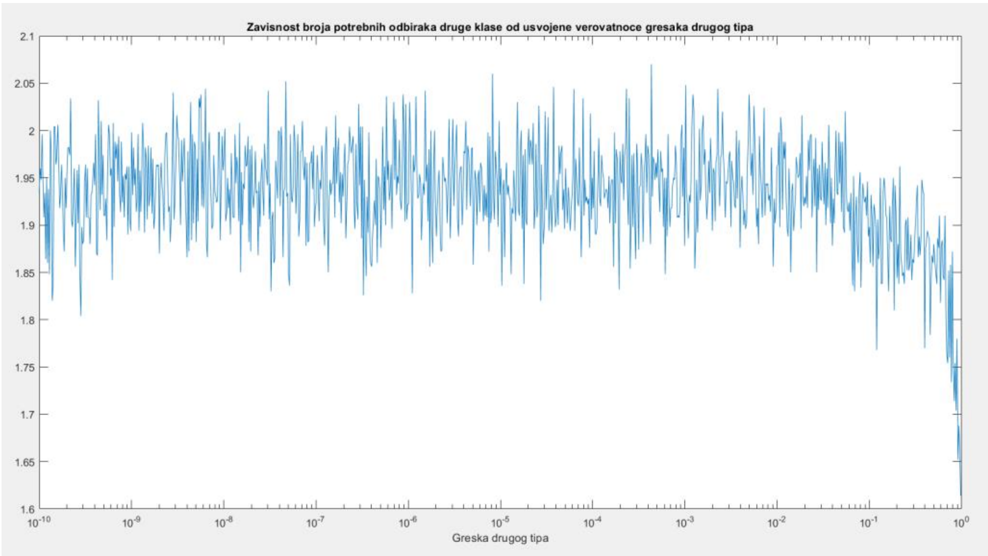
Слика 11. Зависност потребних одбирака прве класе од вероватноће грешке првог типа



Слика 12. Зависност потребних одбирака прве класе од вероватноће грешке другог типа



Слика 13. Зависност потребних одбирака друге класе од вероватноће грешке првог типа



Слика 14. Зависност потребних одбирака друге класе од вероватноће грешке другог типа

Са датих слика се може закључити да број одбирака неће зависити од вероватноће грешке истог типа, док опада уколико расте расте вероватноћа грешке различитог типа.

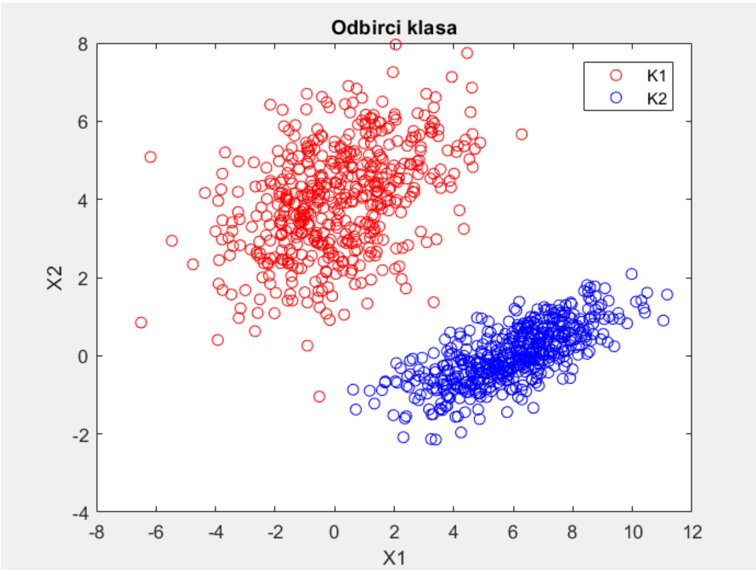
3. Трећи задатак

3.1. Генерисати две класе дводимензионалних облика. Изабрати функцију густине вероватноће тако да класе буду линеарно сепарабилне. За тако генерисане облике извршити пројектовање линеарног класификатора једном од три итеративне процедуре. Поновити претходни поступак коришћењем методе жељеног излаза. Анализирати утицај елемената у матрици жељених излаза на коначну форму линеарног класификатора.

Генерисати две класе дводимензионалних облика које јесу сепарабилне, али не линеарно, па за такве класе испројектовати квадратни класификатор методом по жељи.

За функције густине вероватноћа линерно сепарабилних класа изабране су унимодалне Гаусове расподеле са следећим параметрима:

$$M_1 = \begin{bmatrix} 0 \\ 4 \end{bmatrix},\ \Sigma_1 = \begin{bmatrix} 4 & 1.1 \\ 1.1 & 2 \end{bmatrix}, \qquad M_2 = \begin{bmatrix} 6 \\ 0 \end{bmatrix},\ \Sigma_2 = \begin{bmatrix} 3 & 0.8 \\ 0.8 & 0.5 \end{bmatrix}$$



Слика 15. Одбирци линеарно сепарабилних класа

За пројектовање линеарног класификатора, употребљена је hold-out метода. Она подразумева да се скуп одбирака подели на обучавајући и тестирајући скуп. На основу обучавајућег скупа процењују се математичка очекивања M1 и M2 и коваријационе матрице $\Sigma 1$ и $\Sigma 2$, а затим помоћу тестирајућег скупа рачуна се број грешака и тестира линеарни класификатор. У овом случају скупови су подељени према односу 70:30.

Hold-out метода се реализује као:

- Желимо да пронађемо оптимално s, V и v_0 за свако s између 0 и 1, са кораком од 0.01.
- Рачунамо параметар V на основу формуле:

$$V = (s \cdot \Sigma_1 + (1-s) \cdot \Sigma_2)^{-1} \cdot (M_2 - M_1)$$

- Затим је потребно израчунати:

$$y_1(i) = V^T \cdot X_1(i)$$
$$y_2(i) = V^T \cdot X_2(i)$$

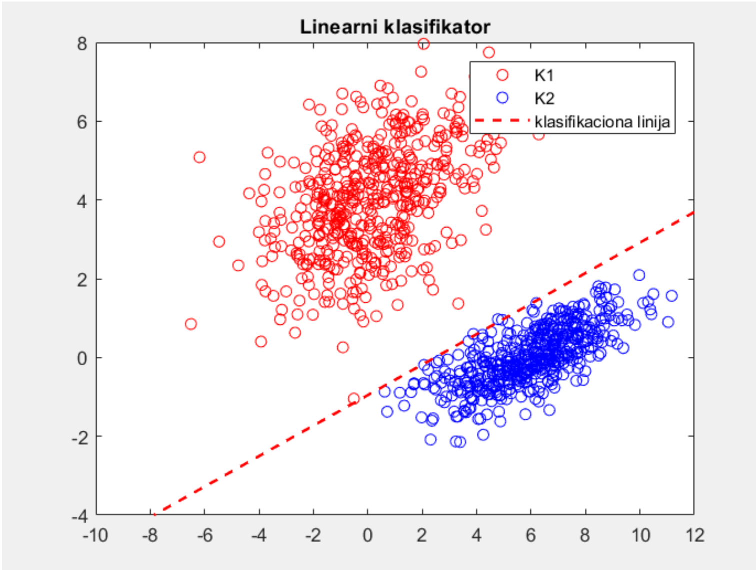
$i = 1, \dots N_t$, где је N_t број одбирака тестирајућег скупа (овај број у овом случају износи 150)

- Како бисмо на крају могли да одредимо и оптимално v_0, њега генеришемо помоћу петље, док су границе у којима га тражимо:

$$v_{0min} = -\max(\max_{i=1,Nt} y_1(i),\ \max_{i=1,Nt} y_2(i))$$
$$v_{0max} = -\min(\min_{i=1,Nt} y_1(i),\ \min_{i=1,Nt} y_2(i))$$

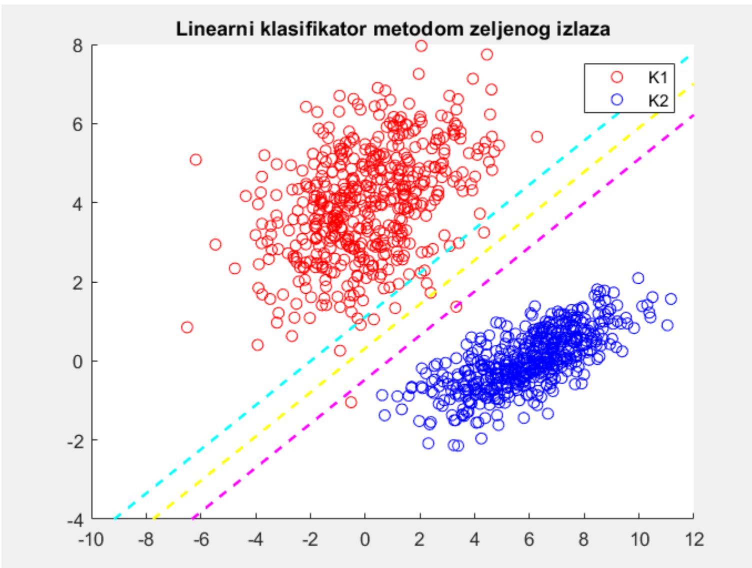
- Свако добијено v_0 поредимо са y_1 и y_2, како бисмо одредили колико пута је дошло до погрешне класификације при тренутно изабраном v_0 (уколико је $y_1 > -v_0$, тада је дошло до погрешне класификације одбирка из класе X_1, а уколико је $y_2 < -v_0$, онда је дошло до погрешне класификације одбирка из класе X_2.
- Грешке се кумулативно додају и усваја се оно v_0, за које је добијено најмањи број погрешно класификованих одбирака из обе класе.

Овај поступак се понавља за свако s и помоћу добијених оптималних параметара се пројектује оптималан линеарни класификатор.



Слика 16. Оптимални линеарни класификатор

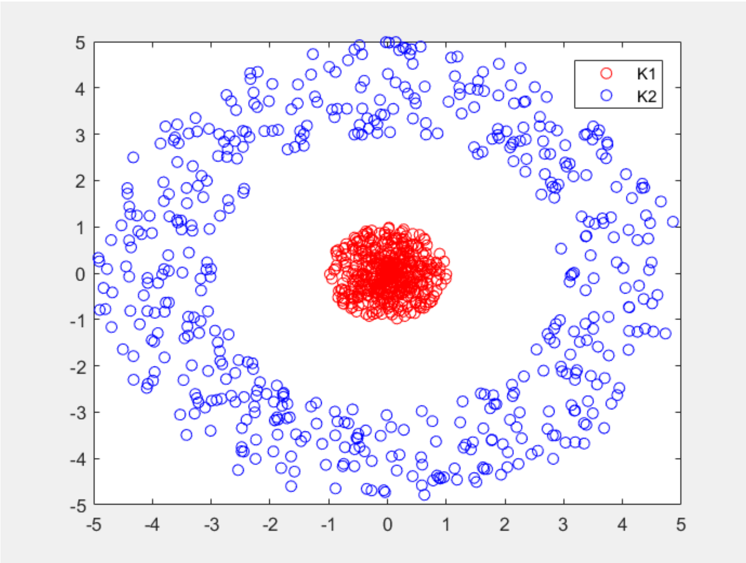
У наредном захтеву задатка, потребно је да класификујемо одбирке класа методом жељеног излаза. Формирамо матрицу U_{3x2N}, такву да првих N колона представља одбирке из прве класе, а наредних N из друге. У првој врсти се налази -1 уколико је одбирак из прве класе, а 1 уколико је одбирак из друге класе. У другој врсти се налазе одбирци из обе класе, с тим што су они из прве класе помножени са -1. Након тога, формирамо вектор W на основу гама критеријума $\varepsilon^2 = \frac{1}{N}\sum_{i=1}^{N}(W^T Z_i - \gamma_i)^2$ који желимо да минимизујемо. Методом псеудоинверзије добијамо израз $W = (U \cdot U^T)^{-1} \cdot U \cdot \Gamma$, где Γ представља матрицу жељених излаза. Уколико не желимо да обратимо посебну пажњу на неку од класа, све вредности у тој матрици су постављене на јединицу. Та класификациона линија је обојена жутом бојом на наредној слици.



Слика 17. Линеарни класификатор жељеног излаза

Уколико пак желимо да обратимо пажњу на одбирке из прве класе, потребно је да првих N чланова матрице Г буду помножени неком константом, па тиме класификациону линију померамо даље од прве класе да обухвати више одбирака, што је приказано ружичастом линијом на претходној слици. А ако је потребно да се више обрати пажња на одбирке друге класе, аналогно ће се других N чланова матрице Г помножити неким коефицијентом, па се линија помера даље од друге класе, што је приказано на прошлој слици плавом линијом.

Последњи захтев у овом задатку тражи да се генеришу две класе које неће бити линеарно сепарабилне, као што се види из наредне слике.



Слика 18. Линеарно несепарабилне класе

Квадратни класификатор има форму:

$$X^T \cdot Q \cdot X + V^T \cdot X + v_0 < 0, \text{ X припада класи } \omega_1$$

$$X^T \cdot Q \cdot X + V^T \cdot X + v_0 > 0, \text{ X припада класи } \omega_2$$

Q, V и v_0 се добијају тако што се овај класификатор своди на линеарни. Метода којом ће се овај класификатор пројектовати је метода жељених излаза. Прво се формира матрица жељених излаза, а затим у прву врсту уписујемо -1 уколико је одбирак из прве класе, а +1 уколико је одбирак из друге класе, и то редоследом да прво убацујемо одбирке из прве а затим из друге класе. У следеће две врсте исписујемо вредности координата генерисаних одбирака, тако да су одбирци прве класе у првој врсти помножени са -1, а одбирци друге класе у другој врсти помножени са +1. Следеће две врсте су попуњене аналогно, с тим што су овог пута у првој врсти квадриране прве координате одбирака, а у другој врсти производ прве и друге координате сваког одбирка посебно, с тим што је сада коефицијент уз одбирке прве класе -2, а коефицијент друге класе +2. У последњој врсти се налазе квадриране друге координате одбирака, уз коефицијенте -1/+1.

Даљи поступак се своди на исти као код метода жељеног излаза:

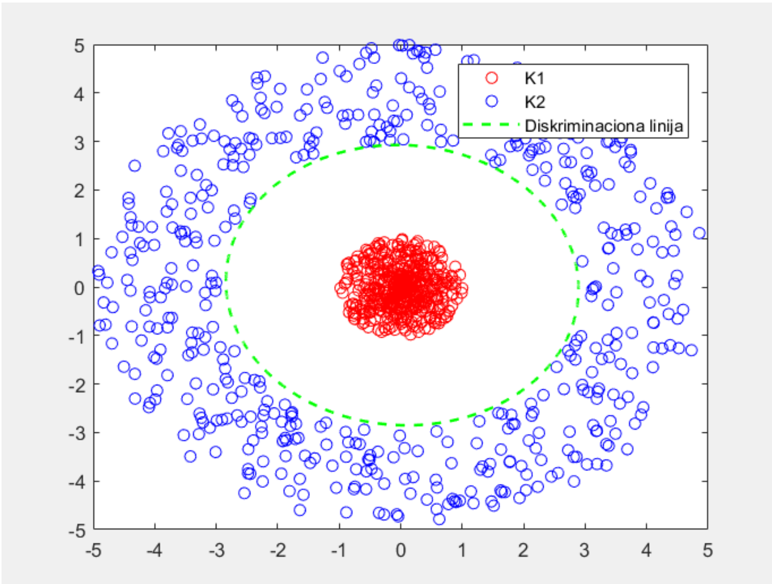
$$W = (U \cdot U^T)^{-1} \cdot U \cdot \Gamma$$

$$v_0 = W(1)$$

$$V = W(2{:}3)$$

$$Q = \begin{bmatrix} W(4) & W(5) \\ W(5) & W(6) \end{bmatrix}$$

На следећој слици је представљен овакав класификатор, заједно са одбирцима из класа. Примећујемо да је класификациона линија доста ближа другој класи, уколико бисмо желели да исправимо овај резултат, могу се даље подешавати параметри у матрици жељених израза на исти начин као код линеарног класификатора.



Слика 19. Квадратни класификатор

4. ЧЕТВРТИ ЗАДАТАК

4.1. Генерисати по N=500 дводимензионих одбирака из четири класе које ће бити линеарно сепарабилне. Препорука је да то буду Гаусовски расподељени дводимензиони облици. Изабрати једну од метода за кластеризацију (c-mean метод, метод квадратне декомпозиције) и применити је на формиране узорке класа. Извршити анализу осетљивости изабраног алгоритма на почетну кластеризацију као и средњи број потребних итерација. Такође извршити анализе случаја када се априорно не познаје број класа.

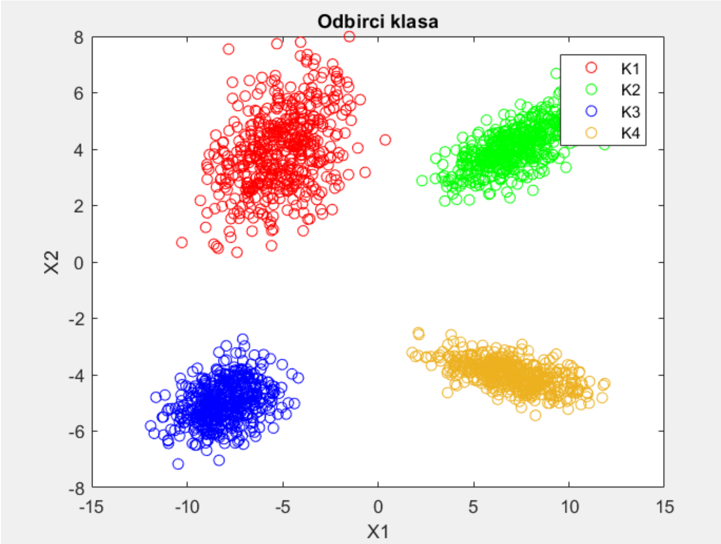
4.2. На одбирцима из претходне тачке изабрати једну од метода кластеризације (метод максималне веродостојности или метод грана и граница) и применити је на формиране узорке класа. Извршити анализу осетљивости изабраног алгоритма на почетну кластеризацију као и средњи број потребних итерација. Такође извршити анализе случаја када се априорно не познаје број класа.

4.3. Генерисати по N=500 дводимензионих одбирака из две класе које су нелинеарно сепарабилне. Изабрати једну од метода за кластеризацију које су примењиве за нелинеарно сепарабилне класе (метод квадратне декомпозиције или метод максималне веродостојности) и поновити анализу из претходних тачака.

У првој тачки је потребно генерисати по 500 одбирака из четири Гаусовски расподељене дводимензионалне класе, које су линеарно сепарабилне. Њихови параметри и одбирци изгледају овако:

$$M_1 = \begin{bmatrix}-5\\4\end{bmatrix},\ \Sigma_1 = \begin{bmatrix}3.5 & 1.1\\1.1 & 2\end{bmatrix},\qquad M_2 = \begin{bmatrix}7\\4\end{bmatrix},\ \Sigma_2 = \begin{bmatrix}3 & 0.8\\0.8 & 0.5\end{bmatrix}$$

$$M_3 = \begin{bmatrix}-8\\-5\end{bmatrix},\ \Sigma_3 = \begin{bmatrix}2 & 0.3\\0.3 & 0.5\end{bmatrix},\qquad M_4 = \begin{bmatrix}7\\-4\end{bmatrix},\ \Sigma_4 = \begin{bmatrix}4 & -0.6\\-0.6 & 0.23\end{bmatrix}$$

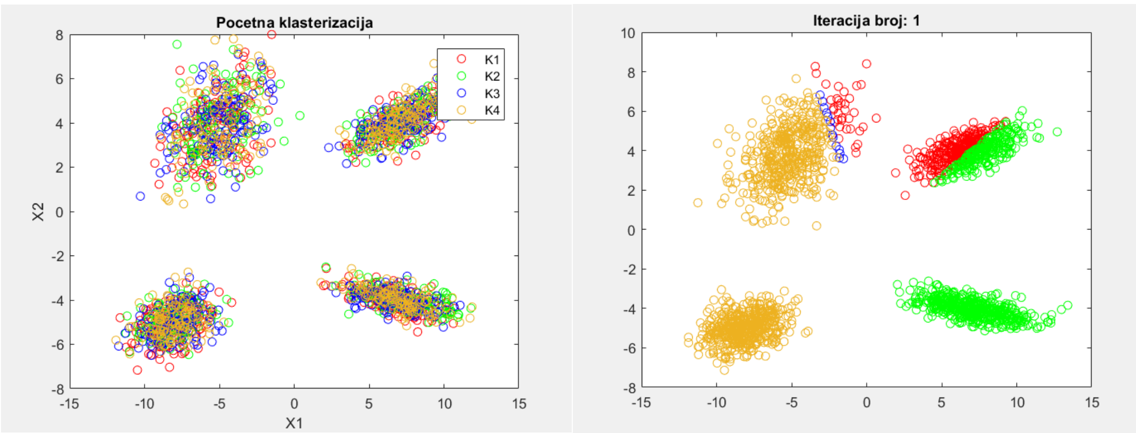


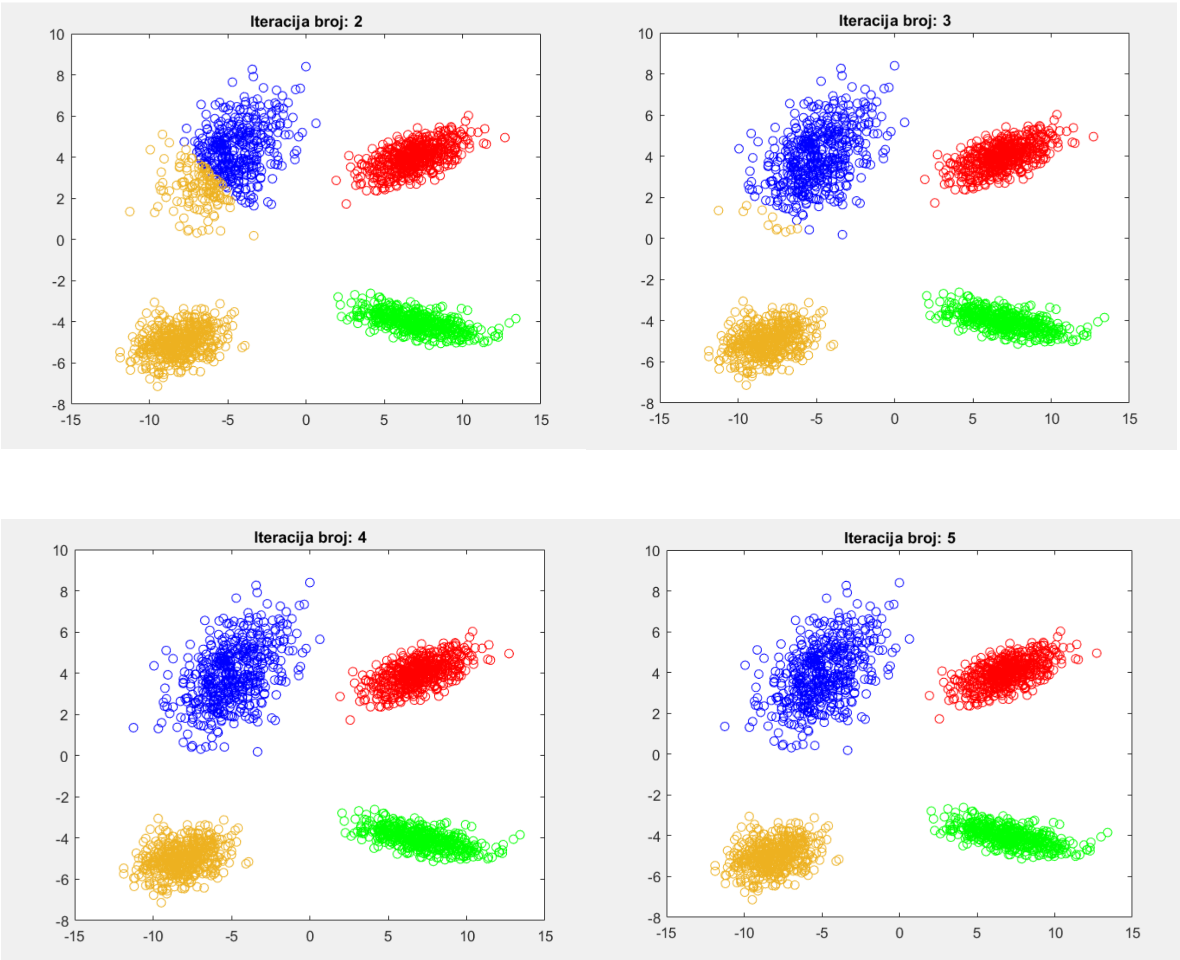
Слика 20. Одбирци класа

Ове четири класе ћемо најпре кластеризовати применом c-mean кластеризације, пошто је она једноставна и класе јесу линеарно сепарабилне. За почетак се одреди почетна кластеризација свих одбирака, која ће бити потпуно произвољна. Како ова метода подразумева да се унапред зна број кластера L, претпоставићемо тачан број класа, тј. кластера, четири. Касније ће бити показано како ће различити број класа и кластера негативно утицати на резултат, као и другачија почетна кластеризација.

Након што смо одредили почетну кластеризацију, процењују се њихова математичка очекивања, тј. центри кластера. Сада је потребно за сваки одбирак у свакој класи одредити Еуклидско растојање од свог центра кластера. Уколико је Еуклидско растојање сваког одбирка најмање за ону класу у којој и јесте (не треба ниједан одбирак рекласификовати), алгоритам се завршава. У супротном се врши рекласификација тог одбирка у класу, с чијим центром има најмање растојање. Битно је да се напомене да пре почетка нове итерације, треба изнова израчунати математичка очекивања за све кластере. Такође, како алгоритам може да се нађе у бесконачној петљи где ће два одбирка да мењају своје кластере наизменично, потребно је ограничити број итерација (у овом случају $l_{max} = 100$).

Резултат овог поступка је дат у наставку:

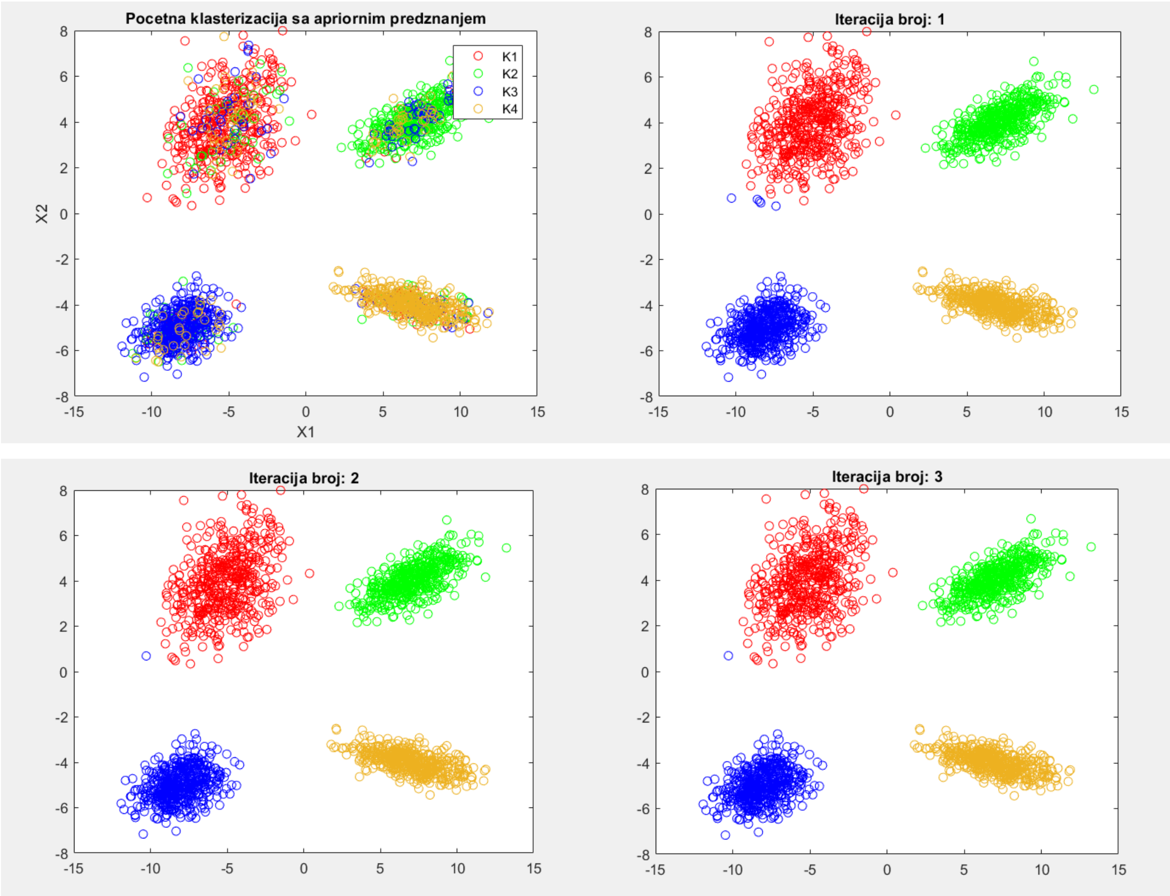




Слика 21. c-mean кластеризација

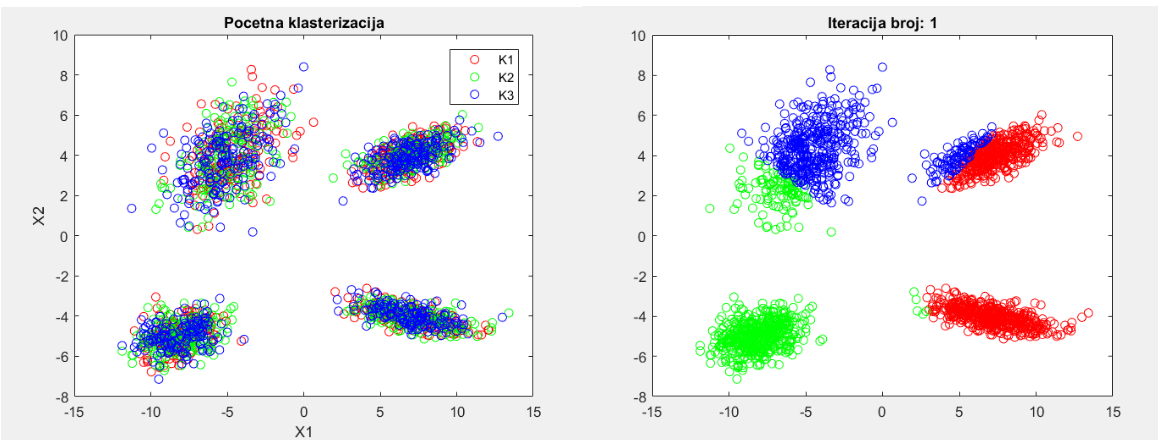
Са слика се види да је коначна кластеризација добијена у просеку већ у четвртој итерацији, што се може утврдити поновним покретањем алгоритма. Код овакве кластеризације, може се пронаћи неки одбирак који је класификован у класу која им је знатно даља у односу на неку другу, којој заправо и припада. Ово се дешава управо зато што овај метод разматра само удаљеност од ценрара кластера (математичких очекивања), а не и коваријационе матрице, што представља и главни недостатак ове методе.

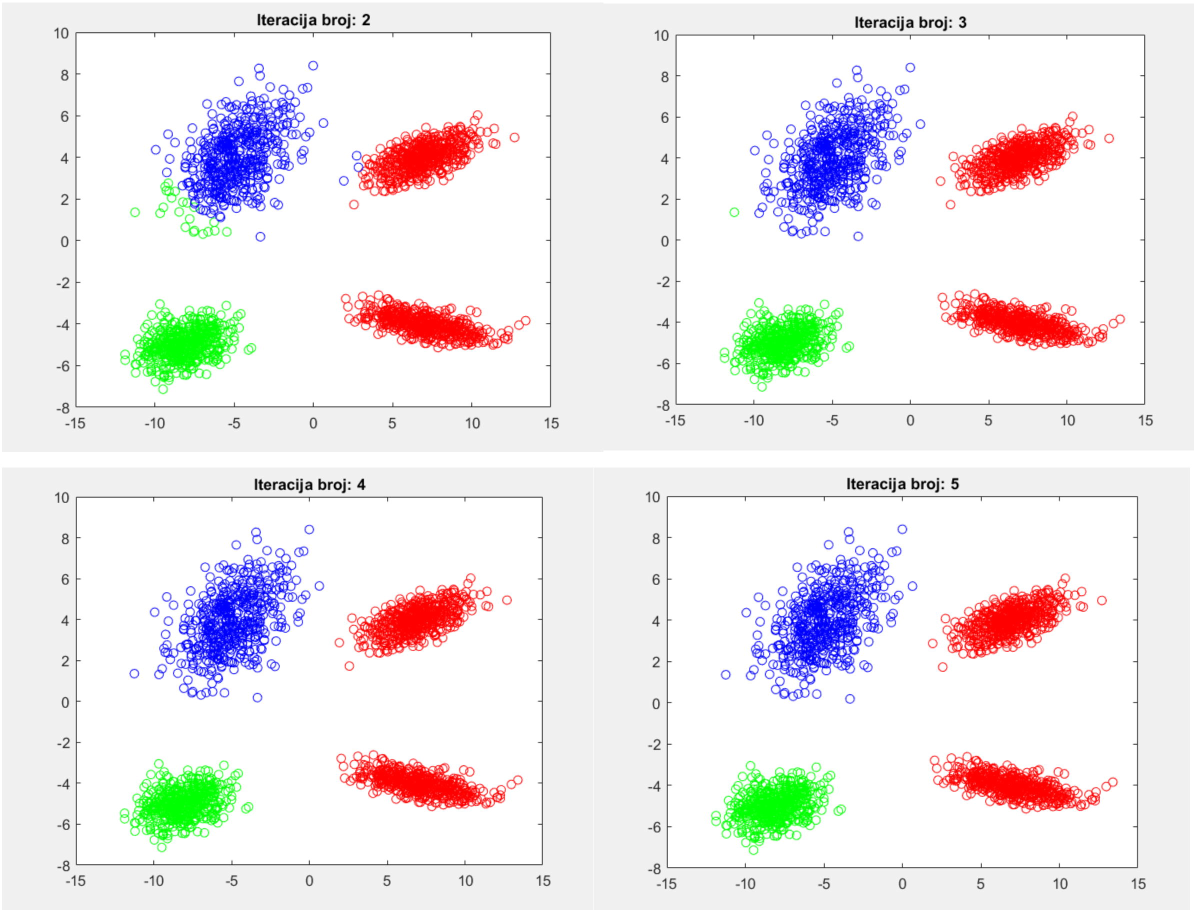
Сада је потребно поновити ову методу на случај где се другачије бира почетна кластеризација. Претпостављено је да за 70% одбирака постоји априорно знање о томе којој класи припада, док су преостали насумично расподељени по кластерима. За сада и даље постоји тачна информација о броју класа. Са наредних слика види се да је сада потребна једна мање итерација у просеку. Дакле, априорно познавање којим класама припадају неки одбирци позитивно утиче на временску сложеност алгоритма, али није пресудно за добру кластеризацију, јер и даље постоји главни недостатак ове методе описан у претходном пасусу.



Слика 22. c-mean кластеризација са априоним предзнањем

Као последњу модификацију, проверићемо како погрешан број кластера у односу на стваран број класа утиче на резултат алгоритма. Прво ћемо претпоставити да је број класа 3, а затим и 5 и упоредићемо добијене резултате.

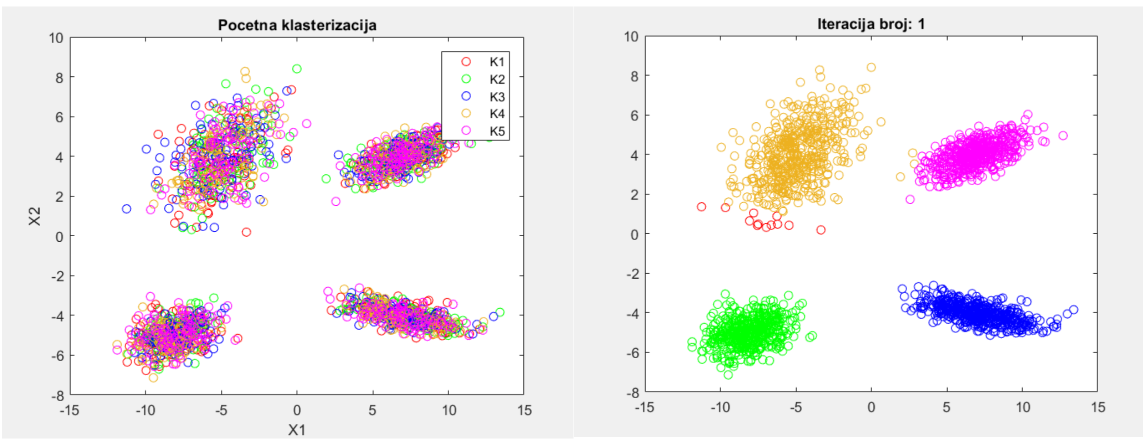


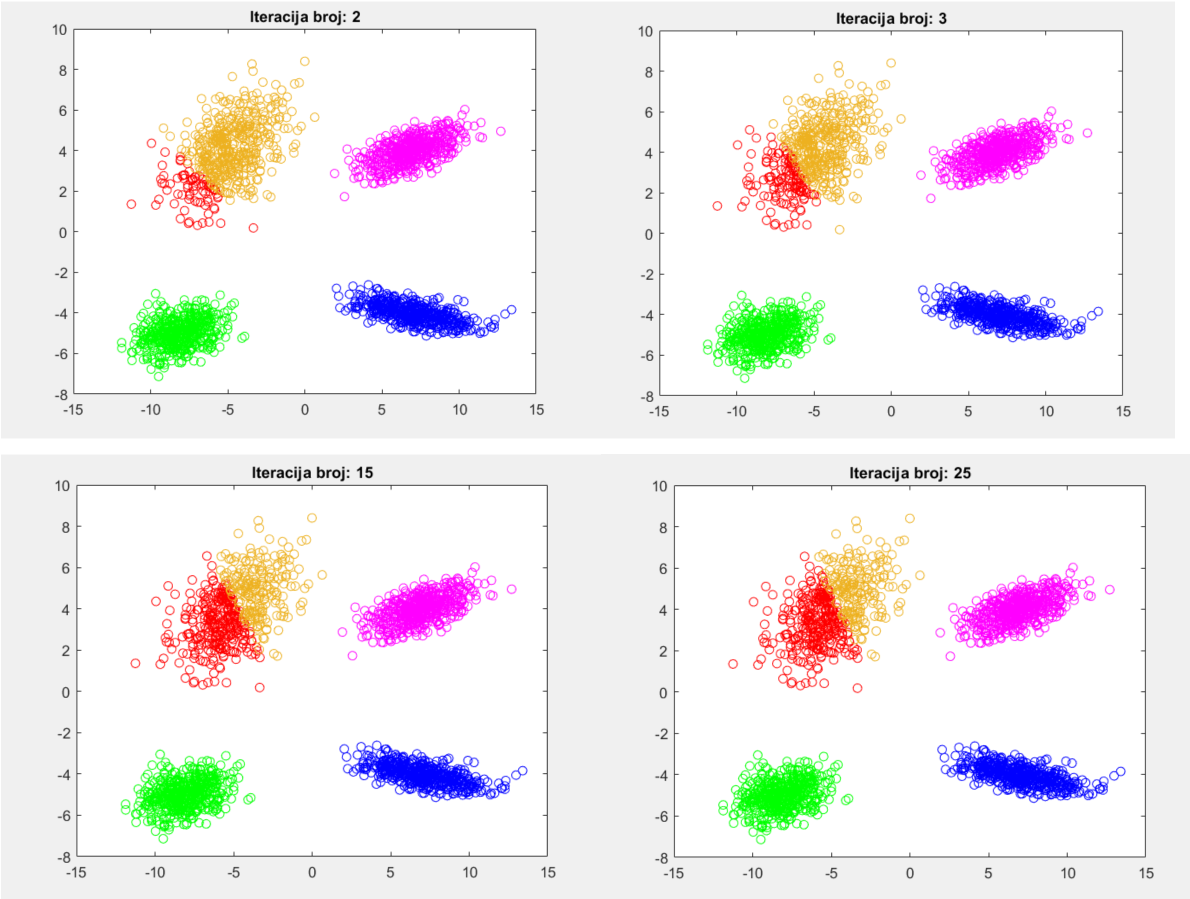


Слика 23. c-mean кластеризација са три кластера

С обзиром на то да је претпостављени број класа мањи од стварног, две класе које су међусобно једна другој најближе ће се спојити. У овом случају потребно је четири или пет итерација да се алгоритам заврши.

Ако се сада претпостави да је број класа једнак пет, добијају се следећи резулати:

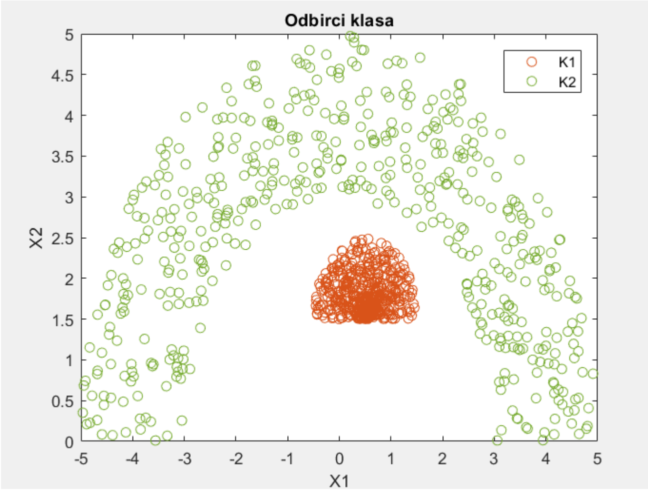




Слика 24. c-mean кластеризација са пет кластера

Како је сад претпостављен већи број класа од онолико колико заправо постоји, догодило се да се једна класа подели на две. Како је потребно да се густо груписани подаци једне класе разделе на две класе, интуитивно је да ће бити потребно много више итерација да се алгоритам изврши. У овом покретању било је потребно 25 итерација да се алгоритам прекине, мада у просеку број итерација знатно варира између ове вредности.

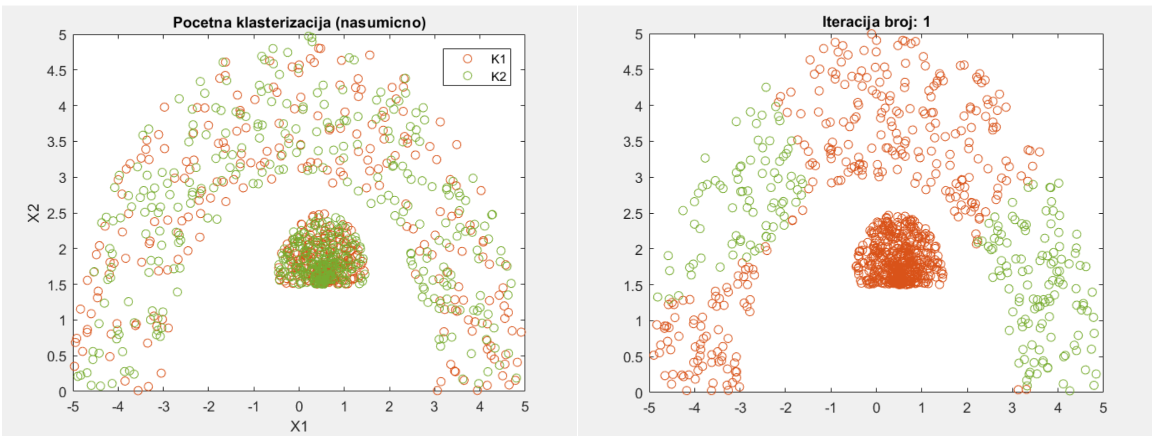
Као последњи захтев, треба спровести исту анализу над нелинеарно сепарабилним класама. Одбирци генерисани су приказани на следећој слици:

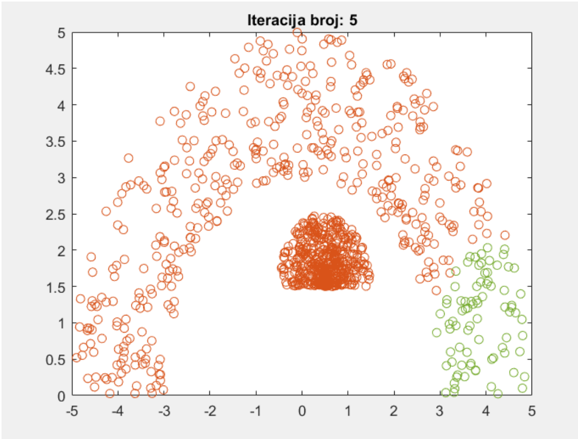


Слика 25. Нелинеарно сепарабилни одбирци

За нелинеарно сепарабилне класе се може употребити метод квадратне декомпозиције. Разлика у односу на c-mean кластеризацију је у томе што се осим математичког очекивања посматра и статистичко одстојање, тако да је самим тим потребно познавати и коваријационе матрице кластера, које се такође прорачунавају после сваке итерације. Такође, како кластери немају подједнак број одбирака, у свакој итерацији је потребно израчунати и априорну вероватноћу појављивања кластера, P_i. Остатак поступка је идентичан.

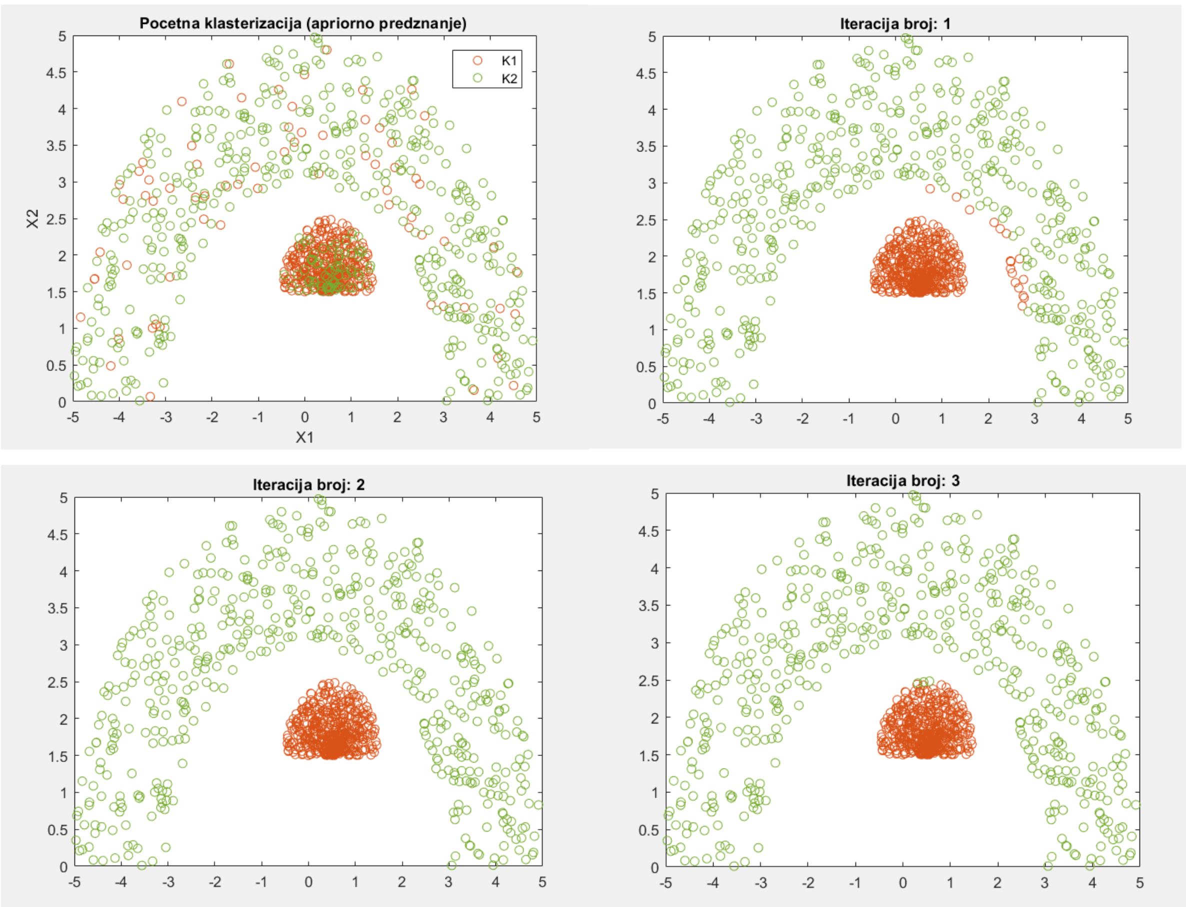
За почетак се посматра најједноставнији случај, када је почетна кластеризација потпуно насумична и када је претпостављено да је број кластера једнак броју класа, а то је два. Резулатати овог алгоритма указују на то да је код квадратне декомпозиције врло важно имати неко априорно знање о припадности одбирака класа одређеним класама. Ово се може закључити на основу више покретања алгоритма и потпуно другачије кластеризације одбирака.

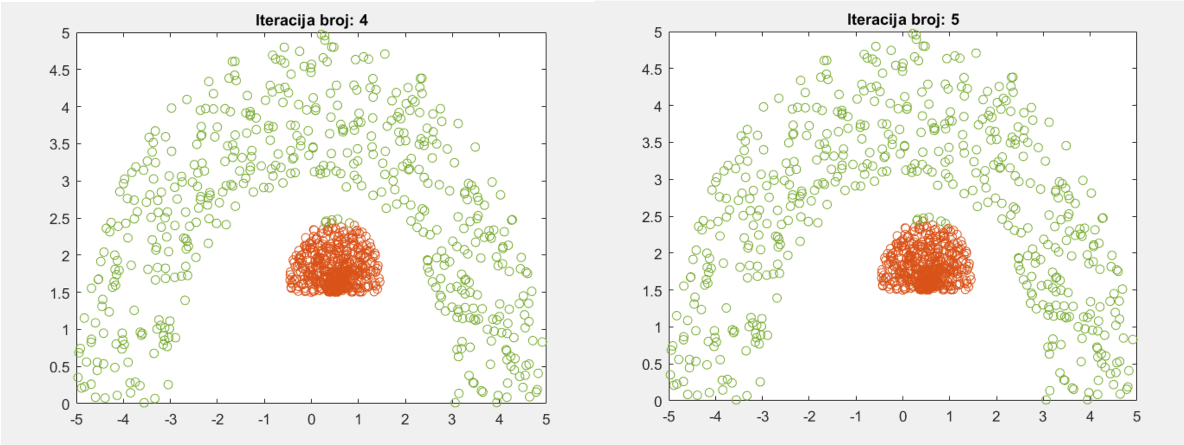




Слика 26. Квадратна кластеризација, лош исход

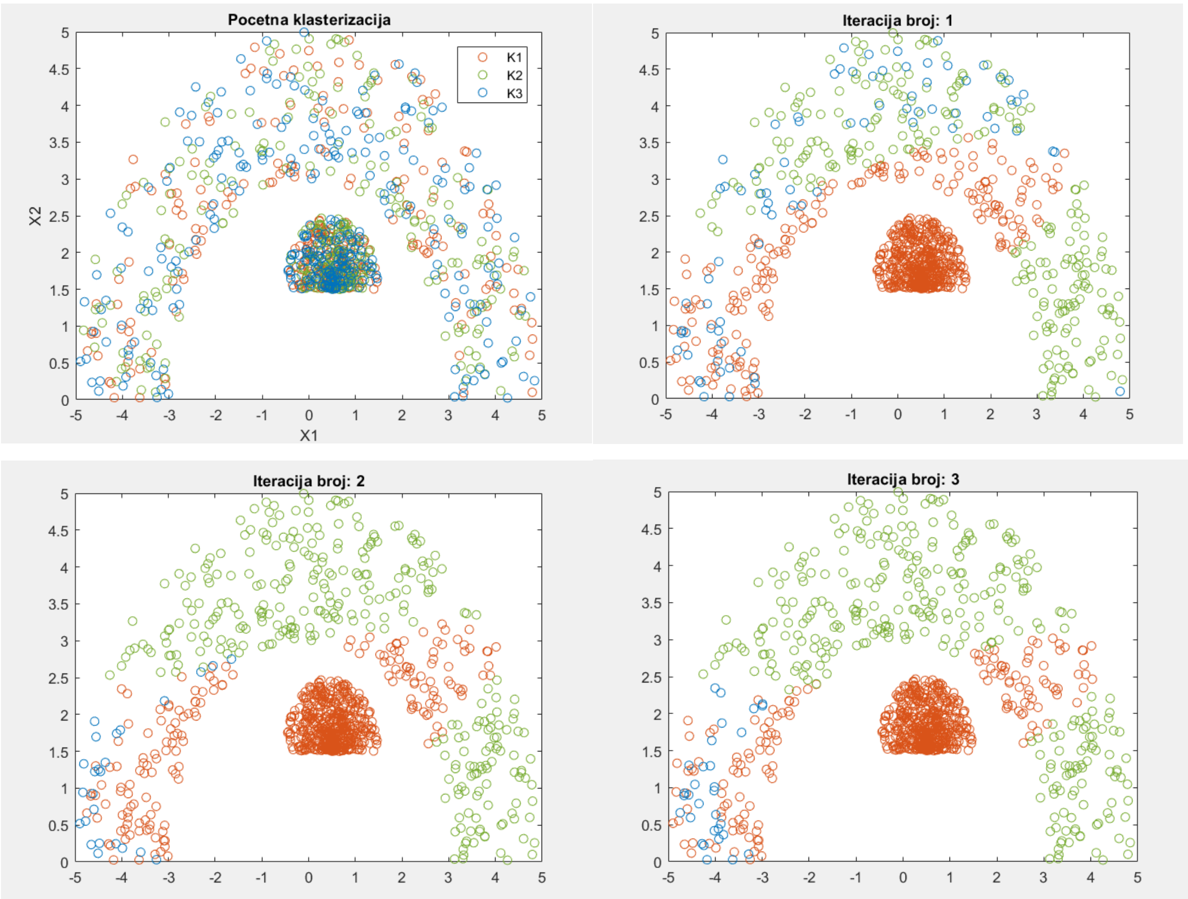
Како је утврђено да је почетна кластеризација итекако утиче на резултат, уводи се претпоставка да је познато априорно припадност одбирака класама, и то за 70% одбирака. Сада се може приметити да је већ прва кластеризација веома слична оној коју треба добити, а до коначне кластеризације се долази после већ неколико итерација, што се може видети са наредних слика.



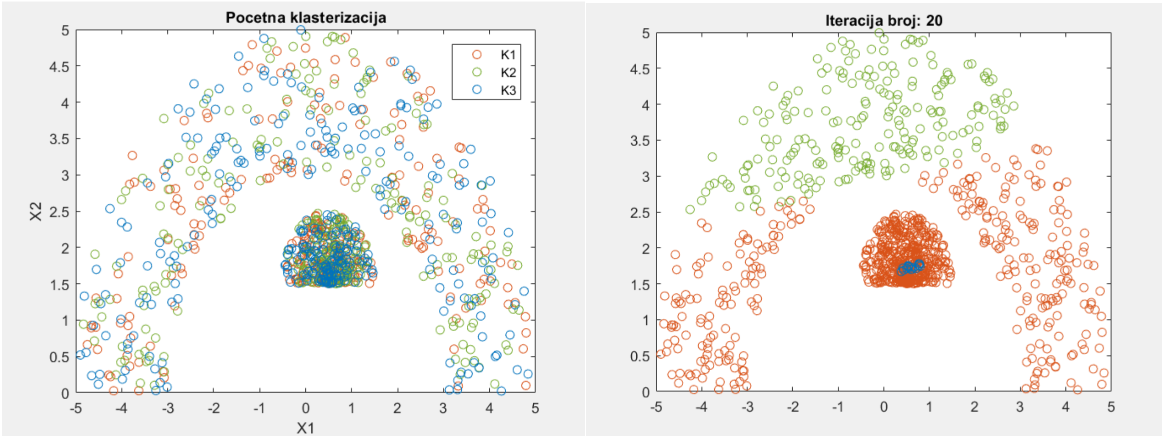


Слика 27. Квадратна кластеризација, добар исход

Преостало је само још да се утврди како број претпостављених класа утиче на резулатат квадратне кластеризације. Како у овом примеру постоје две класе, уколико би се број кластера смањио на један, било би потпуно извесно како би изгледао коначан резултат. Уколико се претпостави да је број кластера једнак три, може се десити или да се изврши свих 100 итерација и одбирци се на неки начин расподеле, или да једна класа у потпуности нестане, па се алгоритам прекине.



Слика 28. Квадратна кластеризација, један кластер ишчезава



Слика 29. Квадратна кластеризација, дивергенција алгоритма