SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1167

Pouzdana klasifikacija tumorskih markera

Nikolina Očić

Zagreb, lipanj 2015.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA DIPLOMSKI RAD PROFILA

Zagreb, 9. ožujka 2015.

DIPLOMSKI ZADATAK br. 1167

Pristupnik: **Nikolina Očić (0036457079)**
Studij: Računarstvo
Profil: Računalno inženjerstvo

Zadatak: **Pouzdana klasifikacija tumorskih markera**

Opis zadatka:

Rana i pouzdana detekcija tumora od presudne je važnosti za uspješno liječenje. SELDI tehnika je ionizacijska metoda u masenoj spektrometriji koja se koristi za analizu proteinskih mješavina (SELDI - surface enhanced laser desorption/ionization). Uzorci tkiva, krvi ili urina se kapnu na posebno pripremljene površine, ioniziraju laserom te ubrzavaju i razvrstavaju masenim spektrometrom. Rezultat su maseni spektri koji se mogu koristiti za detekciju tumora.
U okviru diplomskog rada potrebno je analizirati masene spektre dobivene SELDI metodom radi pouzdane klasifikacije pacijenata s benignim ili malignim tumorom, odnosno zdravih pacijenata. Koristiti napredne metode digitalne obradbe signala za dobivanje relevantnih značajki te odgovarajuće statističke metode za izbor relevantnog podskupa značajki. Za klasifikaciju koristiti nadzirane i nenadzirane metode strojnog učenja. Rezultate usporediti na većem broju dostupnih uzoraka.

Zadatak uručen pristupniku: 13. ožujka 2015.
Rok za predaju rada: 30. lipnja 2015.

Mentor:
Prof. dr. sc. Damir Seršić

Djelovođa:
Prof. dr. sc. Danko Basch

Predsjednik odbora za diplomski rad profila:
Prof. dr. sc. Mario Žagar

Posebna zahvala kolegicama Ankici Gogić i Mirni Domančić koje su razvile temelje za ovaj rad i dale dopuštenje za korištenje programske potpore.

Sadržaj

Popis slika

Popis tablica

Popis formula

POPIS KORIŠTENIH KRATICA

engl.	Engleski
FN	False Negative
FP	False Positive
FS	Fisher Score
GDI	Gini Diversity Index
MALDI	Matrix-Assisted Laser Desorption/Ionization
PSA	Prostate-Specific Antigen
SELDI-TOF	Surface-Enhanced Laser Desorption/Ionization Time-of-flight
SVM	Support Vector Machine
TN	True Negative
TOF	Time of Flight
TP	True Positive

1. Uvod

Jedan od najvećih problema današnjice povećani je broj tumorskih oboljenja. Kako bi borba protiv istih bila što uspješnija, jedna od ključnih stavki je rano i brzo otkrivanje tumora. Trenutno se u medicini u tu svrhu preporučuju razni preventivni pregledi te u slučaju nepovoljnih rezultata pregleda daljnje mjerenje tumorskih biljega. No spomenute metode imaju svojih nedostataka. Sam pregled ovisi o subjektivnoj procjeni liječnika dok mjerenje tumorskih biljega nije uvijek precizno. Kako su tumorski biljezi zapravo različiti proteini, hormoni, enzimi, receptori i drugi stanični produkti koji se pojačano stvaraju u malignim stanicama (Bilušić, 2013), samim time oni u neku ruku diskriminiraju benigne stanice koje također mogu predstavljati tumorska oboljenja. S druge strane, ne luče svi tumori svoj specifičan biljeg te se može dobiti i značajan broj lažno negativnih rezultata (Bilušić, 2013). Također, značajan nedostatak nekih tumorskih biljega je i vrlo zahtjevan, odnosno složen postupak njihovog određivanja što uz visoku cijenu, onemogućuje primjenu mjerenja tumorskih biljega u svakodnevnoj praksi (Bilušić, 2013).

Razvitak tehnologije omogućio je razvijanje bržih i jeftinijih metoda obrade podataka. Upravo te metode mogle bi se primijeniti za otkrivanje novih tumorskih biljega koji bi omogućili brže i bolje otkrivanje tumora. Cilj ovog rada je razvitak programske potpore koja bi pomogla pronalasku pouzdane metode detekcije tumora korištenjem spomenutih novih metoda prikupljanja i obrade podataka. Sama programska potpora sastavljena je od učitavanja i obrade već dostupnih podataka te njihovim daljnjim korištenjem u metodama strojnog učenja kako bi se dobio model koji sa zadovoljavajućom točnošću određuje da li pacijent boluje od tumora.

U nastavku slijedi kratak pregled rada.
U drugom poglavlju ukratko je opisano što je tumor te je predstavljena problematika tumora jajnika i prostate.
U trećem poglavlju opisana je metoda prikupljanja medicinskih podataka, prikazani su

podaci korišteni u radu te su ukratko opisani koraci rada.
Četvrto poglavlje govori o korištenoj metodi obrade podataka te o samim metodama evaluacije dobivenih značajki u svrhu njihovog izvlačenja.
Peto poglavlje ukratko objašnjava odabranu metodu strojnog učenja te prikazuje njezino korištenje u radu.
U šestom poglavlju prikazani su dobiveni rezultati i usporedba između inačica odabrane metode strojnog učenja.

Važno je napomenuti da se ovaj rad djelomično ili u potpunosti nadovezuje na radove drugih kolegica (Gogić, 2013) i (Domančić, 2013) te su zbog toga pojedini dijelovi preuzeti iz citiranih radova, uz dopuštenje autora.

2. Tumor

Tumor, ujedno poznat i kao neoplasma, je abnormalna nakupina tkiva koja može biti kruta ili ispunjena tekućinom (Nordqvist, 2014). Postoje mnoge vrste tumora kao i nazivi za njih. Najčešće naziv predstavlja oblik tumora i vrstu tkiva u kojem nastaje (Nordqvist, 2014). Tumori mogu biti benigni ili maligni. Benigni tumori su razne izrasline, kvržice ili otekline koje ne predstavljaju nužno zdravstvene rizike. Oni ne mogu metastazirati odnosno ne mogu se širiti u okolna tkiva i organe. S druge strane, maligni tumori, ujedno poznati kao rak, veliki su zdravstveni problem. Kao što i sam naziv kaže, to su zloćudne nakupine abnormalnih stanica koje vrlo brzo rastu i šire se. Ovisno o vrsti malignog tumora ali i imunološkog sustava bolesnika, proces širenja može biti vrlo brz te su zbog toga česte pojave smrtonosnih ishoda za bolesnika. U znanstvenom svijetu i dalje se traga za potpunim lijekom za rak te je jedina mogućnost prevencije smrtonosnog ishoda vrlo rana detekcija i tretmani.

Iako ovaj rad klasificira i benigne i maligne i zdrave pacijente, naglasak je na otkrivanju malignih tumora te su u nastavku ukratko objašnjeni maligni tumori odnosno rak jajnika i prostate.

2.1. Rak jajnika

Rak jajnika je zloćudna bolest koja najčešće pogađa žene koje su pred ili u menopauzi (Cybermed, 2013). Po učestalosti zloćudnih bolesti kod žena, rak jajnika je na drugom mjestu. Često je dijagnosticiran u kasnijoj fazi, kada je već proširen na područje zdjelice i trbušne šupljine što ga čini smrtonosnijim i težim za liječenje.

Kao i kod drugih malignih tumora, dijagnoza raka jajnika nije uvijek pravovremena. Razne pretrage obavljaju se tek kada se posumnja na prisustvo raka. Metode korištene u tu svrhu su:

1. Ultrazvuk
   Pomoću ultrazvuka provjerava se veličina i oblik jajnika te se mogu vidjeti strane izrasline na jajnicima koje mogu upućivati na rak.

2. Pretrage krvi
   Vađenjem krvi kontrolira se razina bjelančevine CA 125 koja se nalazi na površini stanica raka jajnika ali i nekih zdravih tkiva. To je svojstveni tumorski biljeg za rak jajnika no nije uvijek vjerodostojan. Neke benigne bolesti mogu prouzročiti povišenu razinu CA 125 dok s druge strane može se dogoditi da u ranijim fazama raka jajnika razina CA 125 bude normalna.
   Zbog navedenih nedostataka pretraga CA 125 se obično koristi za praćenje napretka liječenja a ne za dijagnosticiranje (Cybermed, 2013).

3. Uzimanje uzoraka tkiva
   Nakon dijagnoze drugim metodama, kirurškim uzimanjem uzoraka tkiva može se potvrditi postojanje raka. Ovim postupkom uzimaju se uzorci abdominalne tekućine te se može izvaditi jedan jajnik kako bi se patološki pregledao. Ako se potvrdi postojanje raka, može se odmah započeti sa operacijom kako bi se uklonilo što više zahvaćenog tkiva (Cybermed, 2013).

2.2. Rak prostate

Rak prostate je zloćudni tumor koji najčešće pogađa muškarce starije dobi. Ubraja se u najčešće oblike karcinoma kod muškaraca i drugi je po redu najčešćih uzroka smrti muškaraca od zloćudnih bolesti. U većini slučajeva rak prostate raste sporo i ima veću uspješnost liječenja od raka jajnika ako je otkriven u ranijoj fazi (Cybermed, 2013)(Cybermed, 2014).

Nakon 50. godine života prostata se postepeno povećava te je preporučeno da muškarci stariji od 50 godina redovito odlaze na preglede prostate. Kako u većini slučajeva rak prostate sporo raste, redoviti pregledi prostate mogu dijagnosticirati rak u ranoj fazi što pospješuje liječenje. Metode korištene u dijagnosticiranju raka prostate uključuju:

1. Digitorektalni pregled
   Pregled se izvodi kažiprstom kako bi se mogla napipati prostata. Ovim pregledom može se utvrditi sumnja na postojanje raka prostate te uputiti na daljnje pretrage.

2. Određivanje serumskog PSA

   Serumski PSA je specifični antigen prostate čije povećane razine mogu upućivati na rak. Može se smatrati nekom vrstom tumorskog biljega za rak prostate. No poput drugih biljega, nije u potpunosti vjerodostojan jer također benigna oboljenja mogu imati povišeni PSA dok s druge strane, iako je prisutan rak prostate, razina PSA može biti normalna. Kod povećane razine PSA upućuje se na daljnju biopsiju kako bi se potvrdilo postojanje raka.

3. Biopsija

   Kao i kod raka jajnika, jedina metoda potvrđivanja postojanja raka je uzimanje uzorka tkiva u svrhu patološkog pregleda.

4. Ostale metode koje mogu pomoći u dijagnosticiranju raka prostate su kompjuterizirana tomografija, magnetska rezonanca, scintigrafija kosti i slično (Cybermed, 2014).

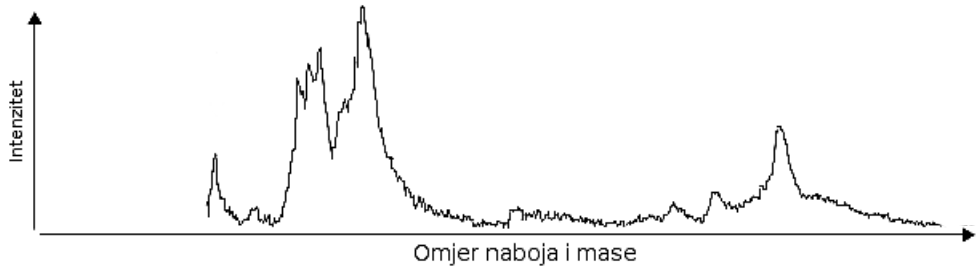
3. Problematika zadatka

Podaci korišteni za potrebe ovog rada prikupljeni su od strane Centra za istraživanje tumora (Bethesda, MD, SAD) (CCR, 2002) i vezani su uz radove (Petricoin, 2002a) i (Petricoin, 2002b). Podijeljeni su u dvije kategorije: u svrhu detekcije raka jajnika te u svrhu detekcije raka prostate. Obje kategorije podataka prikupljene su korištenjem SELDI-TOF metode niske rezolucije.

3.1. SELDI-TOF metoda

Surface-Enhanced Laser Desorption/Ionization Time-of-flight (Seibert, 2003) metoda je za profiliranje populacije proteina u uzorku na temelju veličine i električnog naboja pojedinačnih proteina. Sam SELDI je varijacija MALDI (Matrix-Assisted Laser Desorption/Ionization) masene spektrometrije u kojoj se uzorak proteina ili peptida pomiješa s matricom molekule u otopini te se mala količina mješavine stavlja na površinu i pušta da se osuši. Uzorak i matrica zajedno se kristaliziraju kako otopina isparava. Kod SELDI, početna mješavina proteina se razmazuje na površinu modificiranu s kemijskim značajkama. Neki proteini u uzorku se vežu uz površinu dok se drugi odstranjuju ispiranje. Nakon ispiranja, na površinu se primjenjuje matrica molekule koja se kristalizira sa peptidima iz uzorka. Povezivanje na SELDI površinu korak je koji rastavlja skup proteina te su oni proteini koji su ostali na površini lakši za daljnju analizu.
Razmazani uzorci na SELDI površini dalje se analiziraju TOF (engl. *Time of Flight*) masenom spektrometrijom. Laser ionizira peptide iz kristala mješavine uzorka i matrice. Ioni se ubrzavaju kroz cijev te se mjeri njihov omjer mase i naboja. Ovime se dobiva spektrogram u kojemu je iskazan omjer mase i naboja za svaki protein iz uzorka. Uobičajene SELDI površine koje se koriste uključuju **CM10** (izmjenjivač slabo-pozitivnih iona), **H50** (hidrofobna površina), **IMAC30** (površina za vezanje metala) i **Q10** (izmjenjivač jakih aniona). Djelovanje površine može se nadograditi drugim proteinima, antitijelima ili DNK.

Upravo zbog mogućnosti korištenja raznih vrsta površina, SELDI metoda ima veliku prednost jer je omogućena selekcija proteina vezanih uz pojedini problem. Također, od velike su važnosti i jednostavnost i brzina metode koje omogućuju da ulazni uzorak ne mora biti složen, odnosno pouzdane podatke moguće je iščitati i iz seruma manje koncentracije te s druge strane, SELDI metodom moguće je obraditi veliki broj uzoraka dnevno.



Slika 3.1.1: Primjer spektrograma
(Petricoin, 2002a)

3.2. Korišteni podaci

3.2.1. Skupina podataka za jajnike

Podaci su dobiveni uzimanjem uzoraka seruma nad 216 ispitanica te obrađivanjem uzetih uzoraka SELDI-TOF metodom. Nakon obrade, za svaku ispitanicu, dobiven je maseni spektrogram sa 15154 značajke. Svaka značajka predstavlja položaj i omjer mase i naboja. Podaci su podijeljeni u tri podskupine: 100 uzoraka dobiveno od zdravih žena, 16 uzoraka dobiveno od žena s benignim oboljenjima te 100 uzoraka dobiveno od žena s malignim tumorom.

3.2.2. Skupina podataka za prostatu

Slično kao i u prethodnom odjeljku, podaci su dobiveni uzimanjem uzoraka seruma nad 322 ispitanika te obrađivanjem SELDI-TOF metodom. Nakon obrade, također su dobiveni spektrogrami sa 15154 značajke gdje svaka značajka predstavlja položaj i

omjer mase i naboja. Podaci su podijeljeni u četiri podskupine: 63 uzorka dobivenih od zdravih muškaraca, 190 uzoraka dobivenih od muškaraca sa benignim oboljenjima, 26 uzoraka dobivenih od muškaraca s malignim oboljenjem i nivoom PSA između 4 i 10 te 43 uzorka dobivenih od muškaraca s malignim oboljenjem i nivoom PSA većim od 10. Posljednje dvije podskupine u radu su spojene u jednu s obzirom da je cilj bio odijeliti maligne od ostalih oboljenja.

3.3. Rješenje zadatka

Kao što je već spomenuto, ideja rada je razvitak programske podrške za klasifikaciju medicinskih podataka između benignih i malignih tumora te zdravih osoba. Zbog složenosti zadatka, rad je podijeljen u dvije velike cjeline:

1. Učitavanje i obrada podataka
   Prva cjelina odnosi se na učitavanje početnih podataka, kako za tumore jajnika tako i za tumore prostate. Sljedeći korak je obrada podataka i evaluacija dobivenih podataka u svrhu pronalaska najvažnijih značajki unutar pojedinog medicinskog nalaza. U tu svrhu za obradu podataka odabrana je transformacija Haarovim valićnim paketom kojom se lako određuju one značajke koje imaju najveće vrijednosti omjera mase i naboja u odnosu na druge značajke a potencijalno mogu predstavljati proteinske uzorke specifične za pojedinu klasifikaciju podataka. Nakon određivanja takvih značajki potrebno ih je evaluirati kako bi se odredile one koje su najbitnije za pojedine klasifikacije. Za to su odabrane tri različite mjere - entropija, Gini Diversity Indeks te Fisher Score.

2. Klasifikacija podataka strojnim učenjem
   Nakon spomenutih obrada i evaluacija, prije klasifikacije, potrebno je odrediti optimalan broj najvažnijih značajki kako bi model klasifikacije bio što efikasniji. Za klasifikaciju je od metoda strojnog učenja odabran stroj potpornih vektora (engl. *Support Vector Machine*) (u nastavku SVM) koji je korišten u tri različite inačice: linearni SVM, polinomni SVM te radijalni SVM. Korištenjem SVM-a je u istom algoritmu izveden odabir broja značajki te evaluacija tako stvorenog modela SVM-a.

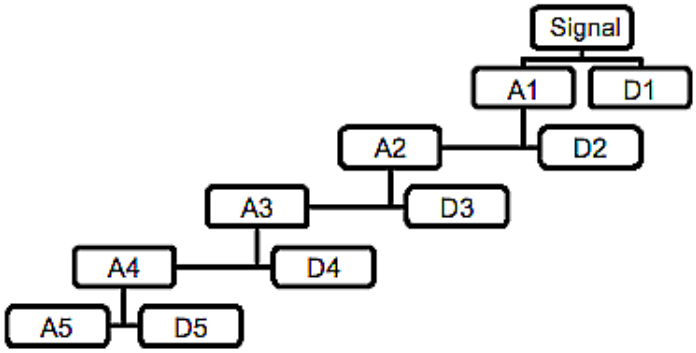
4. Obrada podataka

Kao što je spomenuto, prije izrade klasifikatora za zadani problem, ulazne podatke potrebno je obraditi kako bi izvlačenje najbitnijih značajki bilo što točnije. U tu svrhu, najprije je za svaki uzorak izgrađen Haarov valićni paket te je zatim primijenjena metoda evaluacije svakog člana paketa. Evaluacija je potrebna kako bi se odredili najvažniji valićni paketi, odnosno najvažnije značajke za klasifikaciju. Korištene metode evaluacije uključuju računanje entropije svakog člana paketa, računanje Gini Diversity Index-a svakog člana paketa te računanje Fisher Score-a svakog člana paketa. Ovime se dobivaju tri skupa značajki, za svaku metodu jedan, koji se dalje svaki posebno koriste u svrhu klasifikacije. Razlog višestrukim evaluacijama je usporedba metoda te određivanje najefikasnije.

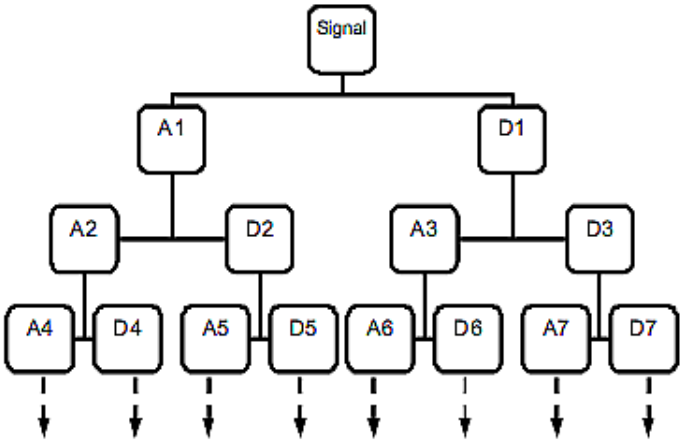
4.1. Transformacija valićnim paketom

Transformacija valićnim paketom jedna je od vrsta valićnih transformacija (engl. *wavelet transform*). **Valićna transformacija** jedna je od mnogih transformacija signala koja omogućava analizu signala i u frekvencijskoj i u vremenskoj domeni.

Primjena valićne transformacije je vrlo jednostavna. Početni signal se propušta kroz različite visoko i nisko frekvencijske filtre te se tako dijeli na dva dijela nazvana **aproksimacije** (aproksimacijski koeficijenti) i **detalji** (koeficijenti detalja). Ovdje se vidi razlika između valićne transformacije i transformacije valićnim paketom. Naime, kod valićne transformacije, daljnji koraci uključuju propuštanje samo aproksimacija kroz željene filtre dok se kod transformacije valićnim paketima propuštaju i aproksimacije i detalji. Postupak se ponavlja sve do unaprijed određene razine dekompozicije. Rezultat je tzv. *valićno stablo*.



Slika 4.1.1: Stablo valićne transformacije
(Stack Exchange)



Slika 4.1.2: Stablo transformacije valićnim paketom
(Stack Exchange)

4.1.1. Haarov valićni paket

Haarov valićni paket (Gogić, 2013) jedna je od vrsta transformacija valićnim paketom. Nastao kao posljedica Haarove sekvence, predložene od strane matematičara Alfreda Haara 1909. godine. Iako je Haarova sekvenca predstavljena daleko prije pojave valićnih paketa, jedna je od najčešće upotrebljavanih reprezentacija paketa. Kada se govori o Haarovom paketu, najčešće se pritom misli na Haarovu matricu - reprezentaciju valićnih paketa koja je napravljena jednostavnim transformacijama ulaznog signala. Postupak kreiranja Haarove matrice je sljedeći:

1. Kreira se prazna matrica dimenzija $2^N \times N$ gdje je 2^N duljina ulaznog signala

proširena na prvu sljedeću potenciju broja 2 (ako je potrebno).

2. Ulazni signal se postavlja u prvi stupac matrice na način da se prvi član signala stavi u prvi redak, drugi član u drugi redak i tako dalje sve do kraja signala.

3. Zatim se uzimaju po dva člana prvog stupca počevši od početka stupca, zbroje se i stave u sljedeći stupac.

4. Ponovno se uzimaju po dva člana prvog stupca počevši od početka, samo što se oni sada oduzimaju i spremaju u drugi stupac od mjesta gdje je stao prethodni korak.

5. Koraci se ponavljaju na sljedećim stupcima sve dok se ne popuni matrica

U prvom koraku, proširivanje broja ulaznih podataka vrši se zbog načina izvođenja algoritma izgradnje matrice koji na završetku za 2^n podataka "napravi" još n stupaca sa valićnim koeficijentima.

Korišteni kod priložen je u Dodatku A.

A	A+B	(A+B)+(C+D)	(A+B)+(C+D)+ (E+F)+(G+H)
B	C+D	(E+F)+(G+H)	((A+B)+(C+D))- ((E+F)+(G+H))
C	E+F	(A+B)-(C+D)	((A+B)-(C+D))+ ((E+F)-(G+H))
D	G+H	(E+F)-(G+H)	((A+B)-(C+D))- ((E+F)-(G+H))
E	A-B	(A-B)+(C-D)	((A-B)+(C-D))+ ((E-F)+(G-H))
F	C-D	(E-F)+(G-H)	((A-B)+(C-D))- ((E-F)+(G-H))
G	E-F	(A-B)-(C-D)	((A-B)-(C-D))+ ((E-F)-(G-H))
H	G-H	(E-F)-(G-H)	((A-B)-(C-D))- ((E-F)-(G-H))

Slika 4.1.3: Izgradnja Haarove matrice
(Gogić, 2013)

4.2. Evaluacija značajki

Evaluacija značajki provodi se zbog određivanja važnosti svakog od valićnih paketa kako bi se oni mogli sortirati. Ovo je veoma bitno za određivanje optimalnog broja značajki prilikom izrade klasifikatora. U nastavku su objašnjene korištene metode evaluacije.

4.2.1. Entropija

Entropija kao pojam ima nekoliko vrlo sličnih značenja ovisno u kojem području se koristi. Područja primjene su (Hrvatska Enciklopedija):

1. Fizika
   Entropija u fizici označava termodinamičku funkciju stanja (oznaka *S*) između dva beskonačno bliska ravnotežna stanja sustava.

2. Informatika
   Entropija u informatici je mjera neodređenosti informacija.

3. Komunikacije
   Entropija u komunikacijama je prosječna vrijednost količine informacija koje odašilje izvor ili koje prolaze komunikacijskim kanalom.

4. Geologija
   U geologiji, entropija je mjera za stupanj uniformnosti taložnih stijena.

5. Povijest umjetnosti
   U povijesti umjetnosti, entropija označava stupanj neuređenosti zatvorenog sustava, proces njegove razgradnje ili "termičke smrti".

Iako donekle različite, gore navedene definicije mogu se skupiti u jednu - entropija je mjera broja specifičnih načina složenosti sustava, odnosno mjera nereda sustava.
U ovom radu korištena je definicija iz komunikacija, odnosno teorije informacija koja kaže da je entropija prosječna vrijednost količine informacija. Još poznata kao i Shannonova entropija, nazvana je prema američkom znanstveniku Claudeu Shannonu koji je člankom *"A mathematical theory of communication"* iz 1948. godine stvorio temelj

teorije informacija. Shannonova entropija izražena je nad diskretnim skupom podataka x i može se izraziti formulom :

$$H(x) = -\sum_{i=1}^{n} p_i log_2 p_i \ \ [bit] \tag{4.2.1}$$

gdje su p_i vjerojatnosti pojavljivanja svakog od n podataka u skupu x.

Uz dopuštenje autora, iskorištena je programska podrška razvijena u (Gogić, 2013). Kako je opisano u (Gogić, 2013), algoritam je osmišljen tako da je entropija za svaki skup valićnih koeficijenata računata odmah prilikom transformacije valićnih paketa. Prvotno su skupovi paketa podijeljeni u dvije vrste - otac i dijete. Otac je bio onaj skup nad kojim je izvršena valićna transformacija u koraku $n-1$, dok su skupovi djece bili oni skupovi koji su dobiveni transformacijom skupa otac u koraku n tako da je prvo dijete bio skup suma a drugo dijete skup razlika.
Postupak kreće od prvog stupca koji je uzet kao otac te drugog stupca čija je gornja polovica predstavljala prvo dijete a donja polovica drugo dijete. Opisani postupak proveden je nad matricama valićnih paketa svih pacijenata tako da su svi prikupljeni očevi stavljeni u jedan zajednički vektor a djeca u svoje vektore. Nad tako dobivenim vektorima izračunate su entropije korištenjem ugrađene MATLAB funkcije *wentropy* koje su služile za određivanje optimalnih valićnih koeficijenata. Odluka se temeljila na provjeri koja je entropija manja - entropija oca ili suma entropija djece. Ukoliko je manja entropija oca algoritam nastavlja dalje na sljedećeg oca u stupcu, u suprotnom svako se dijete tretira kao novi otac svojoj djeci. Postupak se ponavlja sve dok su moguća daljnja grananja, odnosno dokle god trenutni član ima djece.
Kao krajnji rezultat dobiven je vektor duljine iste kao i vektor početnih značajki proširen na prvu potenciju broja 2 koji se dalje, nakon određivanja optimalnog broja značajki, sažima na tu duljinu i koristi pri klasifikaciji.

Važno je spomenuti da je u opisanom algoritmu korišteno svojstvo aditivnosti entropije koje kaže kako je suma entropija dva različita vektora jednaka entropiji unije ta dva vektora. Također, korišteno je i svojstvo zbijenosti entropije koje kaže kako je entropija zbijenijeg vektora veća od entropije manje zbijenog vektora iste sume koeficijenata. Kako je cilj pronaći značajke čiji su intenziteti u odnosu na susjedne značajke bitno veći, ovim svojstvom razmatraju se upravo paketi koji sadrže tražene značajke jer su vektori tih paketa zbijeniji od drugih.

Korišteni kod priložen je u Dodatku B.

4.2.2. Gini Diversity Index

Gini Diversity Index kombinacija je indeksa raznolikosti i Gini koeficijenta.
Indeks raznolikosti (Jost, 2006) kvantitativna je mjera koja označava koliko je različitih tipova unutar nekog skupa istovremeno uzimajući u obzir koliko jednako su raspoređeni bazni entiteti među tim tipovima. Indeks raste s porastom broja tipova podataka ali i sa porastom jednakosti rasporeda entiteta među tipovima. Dostiže svoju maksimalnu vrijednost kada su svi entiteti jednoliko raspoređeni među tipovima.
Gini koeficijent,poznat i kao **Gini indeks** (World Bank; Investopedia), mjera je distribucije dohotka stanovnika neke zemlje. Pomaže odrediti procjep između bogatih i siromašnih s vrijednostima između 0 i 1 gdje 0 predstavlja savršenu jednakost a 1 savršenu nejednakost. Gini indeks mjeri površinu između Lorentzove krivulje [1] i hipotetske linije koja predstavlja potpunu jednakost. Najčešće se vrijednost izražava postotkom.

Kombiniranjem Gini indeksa i indeksa raznolikosti dobiven je **Gini Diversity Index** (u nastavku GDI) koji ne mjeri samo nejednakost između pojedinaca već određuje i njihove tipove. Što je vrijednost GDI-a manja to je podjela bolja i rezultat se smatra optimalnijim.

Kao i kod entropije, korištena je programska podrška za računanje GDI-a razvijena u (Gogić, 2013). Za izračun GDI-a potrebno je znati kojoj klasi pripada pojedini entitet, odnosno u ovom slučaju pacijent. Algoritmom je za svaku poziciju paketa, GDI iz paketa na toj poziciji od svakog pacijenta stavljen u poseban vektor koji je potom sortiran i zapamćeni su indeksi nakon sortiranja. Dobiveni indeksi izmijenjeni su ovisno o odabiru tipa GDI. Tip GDI predstavlja odabir klase za koju se GDI računao, odnosno GDI je moguće računati za benigne tumore (podjela na ima/nema benigni tumor), maligne tumore (podjela ima/nema benigni tumor) te zdravlje (podjela na zdravi/bolesni). Indeksi sortiranja izmijenjeni su tako da su na pozicijama koje odgovaraju odabranoj klasi stavljene jedinice dok su na ostalim pozicijama nule. Nad novodobivenim indeksima proveden je izračun GDI-a koji je zatim zapisan na istu poziciju u novoj matrici. Ovime su dobivene tri nove matrice, po jedna za svaki odabir tipa. Dobivene matrice istih su dimenzija kao i matrice valićnih paketa. Pri daljnjoj obradi koja uključuje odabir optimalnog broja značajki i klasifikaciju, formiran je sažeti vektor za svaku matricu koji sadrži onoliko najmanjih GDI vrijednosti koliko je određeno kao optimalan broj

[1] U ekonomiji, Lorentzova krivulja je grafički prikaz kumulativnog postotka prihoda naspram kumulativnog broja korisnika, počevši od najsiromašnijeg pojedinca ili kućanstva.

značajki. Kao krajnji korak, vektori su spojeni u jedan koji predstavlja indekse optimalnih valićnih paketa za klasifikaciju.

Korišteni kod priložen je u Dodatku C.

4.2.3. Fisher Score

Fisher Score (u nastavku FS), poznat još i pod imenom **F-test**, statistička je mjera koja se koristi za odabir značajki u strojnom učenju. Mjera je nazvana prema engleskom statističaru Ronaldu Aylmeru Fisheru koji je razvitkom analize varijance stvorio temelj za FS.
Glavna ideja FS-a je pronalazak podskupa značajki takvih da su, unutar podataka predstavljenih odabranim značajkama, udaljenosti između podataka različitih klasa najveće moguće dok su udaljenosti između podataka iste klase najmanje moguće (Gu, 2012).
FS je moguće računati na sljedeći način:

$$F(x_j) = \frac{\sum_{k=1}^{c} n_k(\mu_k^j - \mu^j)^2}{(\sigma^j)^2} \tag{4.2.2}$$

uz

$$(\sigma^j)^2 = \sum_{k=1}^{c} n_k(\sigma_k^j)^2 \tag{4.2.3}$$

gdje su

- x_j *j*-ta značajka za koju se računa FS
- n_k veličina *k*-te klase
- μ_k^j srednja vrijednost *j* značajke u *k*-toj klasi
- μ^j srednja vrijednost *j*-te značajke u cjelokupnom skupu podataka
- σ_k^j standardna devijacija *j*-te značajke unutar *k*-te klase
- σ^j standardna devijacija *j*-te značajke unutar cjelokupnog skupa podataka.

Što je FS neke značajke veći, to je bolja efikasnost značajke u klasifikaciji. Ovime za odabir značajki uzimamo samo one najvećih vrijednosti.

FS ima i svojih nedostataka. Kako se vrijednost za pojedinu značajku računa neovisno o drugima, dobiveni rezultat nije u potpunosti optimalan jer ne uzima u obzir međusobnu povezanost značajki. Naime, algoritam računanja ne odabire značajke koje u kombinaciji s nekom drugom značajkom daju iznimno velik rezultat ako su pojedinačno dale vrlo mali rezultat. Također, algoritam ne provjerava redundantnost

značajki. Usprkos svemu, FS je jednostavna i efikasna te se smatra vrlo dobrom metodom odabira značajki.

Algoritam izračuna FS-a, razvijen u ovom radu korištenjem formule (4.2.2), vrlo je jednostavan. Nad podacima transformiranim valićnom transformacijom, računa se FS za svaku pojedinu značajku. Računanje se vrši tako da se kod razmatranja *j*-te značajke kreiraju pomoćni vektori i varijable koje se koriste u formuli. Navedeni vektori i varijable su:

- vektor koji sadrži vrijednosti *j*-te značajke prikupljene iz valićnih paketa svih pacijenata
- vektor koji sadrži vrijednosti *j*-te značajke prikupljene od pacijenata klase 1, odnosno pacijenata s benignim tumorom
- vektor koji sadrži vrijednosti *j*-te značajke prikupljene od pacijenata klase 2, odnosno pacijenata s malignim tumorom
- vektor koji sadrži vrijednosti *j*-te značajke prikupljene od pacijenata klase 3, odnosno zdravih pacijenata
- pomoćne varijable koje predstavljaju srednje vrijednosti i standardne devijacije spomenute u formulama (4.2.2) i (4.2.3)

Rezultat dobiven algoritmom je matrica koja sadrži vrijednosti FS-a za svaku značajku, odnosno za svaki valićni paket, istih dimenzija kao i rezultantna matrica kod valićne transformacije. Dobivena matrica se dalje koristi u odabiru optimalnog broja značajki za klasifikaciju.

Razvijeni kod priložen je u Dodatku D.

5. Klasifikacija podataka

Kao što je već spomenuto, za klasifikaciju podataka odabrana je metoda strojnog učenja *Support Vector Machine* (u nastavku SVM). Korištene su tri inačice odabrane metode - linearni SVM, polinomni SVM te radijalni SVM. U nastavku slijedi općenito o SVM-u te ukratko o svakoj od inačica.

5.1. Stroj potpornih vektora

Stroj potpornih vektora (engl. *Support Vector Mashine*) (u nastavku SVM), još poznat i kao *Support Vector Network*, model je nadziranog [2] strojnog učenja baziran na algoritmima učenja koji analiziraju ulazne podatke i prepoznaju uzorke važne za klasifikaciju.

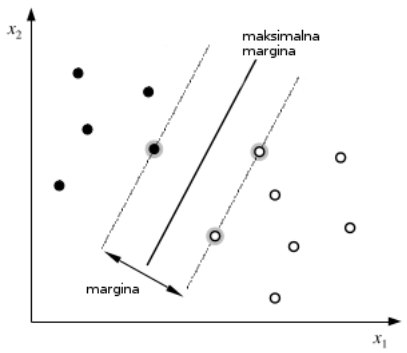
Naziv *Support Vector Machine* dolazi od podataka koji se koriste za izgradnju. Podaci koji se nalaze najbliže liniji klasifikacije nazivaju se **potporni vektori** (engl. *support vectors*) i najkorisnije su točke podataka jer to su oni podaci za koje postoji mogućnost krive klasifikacije. Iz ovog proizlazi najzanimljivije svojstvo SVM-a: nakon treniranja modela, svi podaci osim *support vectora* mogu se odbaciti a za klasifikaciju se mogu koristiti samo potporni vektori (Marsland, 2009).

5.1.1. Linearni SVM

Linearni SVM radi na pretpostavci da su ulazni podaci linearno odvojivi. Ako su podaci za treniranje linearno odvojivi, odaberu se dvije hiper-ravnine takve da klasificiraju podatke ali nema nikakvih podataka između njih i maksimizira se udaljenost između tih hiper-ravnina. Područje koje one zatvaraju nazivaju se **margina** (engl. *mar-*

[2] Nadzirano učenje je vrsta strojnog učenja gdje se podaci nad kojima se trenira model učenja sastoje od parova ulazne vrijednosti i željenog izlaza. Drugim riječima, model uz pomoć poznatih klasifikacija (izlaznih vrijednosti) ulaznih podataka pronalazi uzorke kojima klasificira nove podatke.

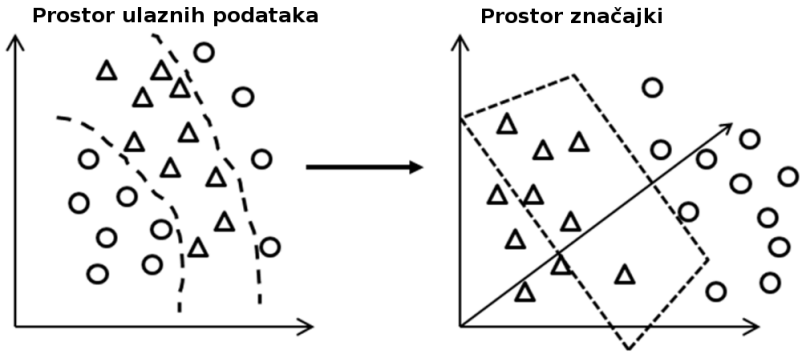
gin). Unutar strana margine pronalazi se hiper-ravnina čija je udaljenost do najbližih podataka sa svake strane maksimalna. Takva hiper-ravnina naziva se **hiper-ravnina maksimalne margine** (engl. *maximum-margin hyperplane*). Najbliži podaci sa svake strane čija se udaljenost maksimizirala su upravo potporni vektori i oni leže na ranije spomenutim hiper-ravninama koje čine marginu.



Slika 5.1.1: Hiper-ravnine margine i maksimalne margine
(Šnajder, 2014)

5.1.2. Nelinearni SVM

Iako je SVM u suštini linearni klasifikator, moguće je provoditi i nelinearnu klasifikaciju. Dizajniranje nelinearnog SVM-a temelji se na sljedećem principu (Huang, 2006): ulazni vektori podataka $x_1 \in \mathbb{R}^m$ preslikavaju se u vektore $\phi(x_i) \in \mathbb{R}^s$ višedimenzijskog prostora značajki S i rješava se linearni klasifikacijski problem u novonastalom prostoru značajki. Preslikavanje ϕ je unaprijed postavljena funkcija.



Slika 5.1.2: Preslikavanje podataka
(Pei, 2012)

Iako ovakvo preslikavanje omogućuje rješavanje nelinearnih problema, postoje dva osnovna problema (Huang, 2006):

1. Odabir preslikavanja ϕ trebalo bi rezultirati linearno odvojivim problemom u prostoru značajki.

2. Računanje skalarnog produkta $\phi^T(x_i)\phi(x_i)$ može biti računski veoma zahtjevno ako su dimenzije prostora značajki S velike.

Problemi se mogu riješiti tako da se, uzevši u obzir da se preslikani vektori $\phi(x)$ uvijek pojavljuju u obliku skalarnog produkta, učini takozvani **trik jezgre** (engl. *kernel trick*) (Šnajder, 2014): umnožak dvaju primjera x_i i x_j u prostoru značajki možemo zamijeniti funkcijom

$$\kappa(x_i, x_j) = \phi^T(x_i)\phi(x_J) \tag{5.1.1}$$

koja se naziva **jezgrena funkcija** (engl. *kernel function*). Jezgrena funkcija mjeri sličnost dvaju vektora u nekom prostoru značajki. Moguće je koristiti razne jezgrene funkcije kako bi se konstruirali modeli SVM-a koji rade u bilo kojoj dimenziji prostora.

Vrijednosti jezgrenih funkcija za sve parove primjera iz skupa podataka za učenje mogu se izračunati unaprijed i pohraniti u simetričnu matricu dimenzija $N \times N$ gdje N predstavlja veličinu skupa podataka za učenje:

$$\mathbf{K} = \begin{pmatrix} \kappa(x^{(1)}, x^{(1)}) & \kappa(x^{(1)}, x^{(2)}) & \dots & \kappa(x^{(1)}, x^{(N)}) \\ \kappa(x^{(2)}, x^{(1)}) & \kappa(x^{(2)}, x^{(2)}) & \dots & \kappa(x^{(2)}, x^{(N)}) \\ \vdots & \vdots & \ddots & \vdots \\ \kappa(x^{(N)}, x^{(1)}) & \kappa(x^{(N)}, x^{(2)}) & \dots & \kappa(x^{(N)}, x^{(N)}) \end{pmatrix} = \mathbf{\Phi}\mathbf{\Phi}^T \tag{5.1.2}$$

gdje je $\mathbf{\Phi}$ dizajn-matrica oblika

$$\mathbf{\Phi} = \begin{pmatrix} 1 & \phi_1(x^{(1)}) & \dots & \phi_m(x^{(1)}) \\ 1 & \phi_1(x^{(2)}) & \dots & \phi_m(x^{(2)}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \phi_1(x^{(N)}) & \dots & \phi_m(x^{(N)}) \end{pmatrix} \tag{5.1.3}$$

a gdje je m broj značajki ulaznih podataka. Matrica $\mathbf{K}$ naziva se **Gram-matrica** ili **jezgrena matrica** (Šnajder, 2014). Da bi jezgreni trik funkcionirao, jezgrena funkcija mora biti Merzerova jezgra, odnosno mora zadovoljavati Merzerov teorem: *"Ako je*

jezgrena matrica $\mathbf{K}$ *pozitivno semidefinitna* [3] *za svaki skup podataka za učenje, onda je jezgrenu funkciju* κ *uvijek moguće rastaviti na skalarni produkt vektora* $\kappa(x_i, x_j) = \phi^T(x_i)\phi(x_j)''$.

Polinomni SVM

Kao i kod drugih nelinearnih modela SVM-a, naziv potječe od korištene jezgrene funkcije. U ovom slučaju to je polinom n-tog stupnja, odnosno jezgrena funkcija je oblika

$$\kappa(x_i, x_j) = (x_i^T x_j + 1)^n \tag{5.1.4}$$

Radijalni SVM

Potpuno ime jezgrene funkcije ovog modela je **radijalna bazna funkcija** ili **homogena jezgra**. Jezgrena funkcija je oblika

$$\kappa(x_i, x_j) = \kappa(|x_i - x_j|) \tag{5.1.5}$$

a ovisi samo o udaljenosti između primjera. Poseban slučaj radijalne bazne funkcije jest **Gaussova jezgra** koja je i korištena u ovom radu:

$$\kappa(x_i, x_j) = exp\{-\frac{|x_i - x_j|^2}{2\sigma^2}\} = exp\{\gamma|x_i - x_j|^2\} \tag{5.1.6}$$

Gaussova jezgra mjeri sličnost dvaju primjera temeljem njihove udaljenosti u ulaznom prostoru gdje parametar γ kontrolira kojom brzinom $\kappa(x_i, x_j)$ teži k nuli u ovisnosti o udaljenosti (Šnajder, 2014). Ovaj parametar vrlo je važan jer njegove različite vrijednosti daju različita preslikavanja.

5.2. Primjena SVM-a u radu

Kako bi nova implementacija modela SVM-a bila prezahtjevna za okvir ovog rada, korištena je gotova biblioteka *LIBSVM* (Chang, 2011). Iako je SVM u suštini binarni klasifikator, kombinacijom više SVM-ova moguće je realizirati klasifikator za više klasa. U ovoj biblioteci, klasifikacija više klasa realizirana je metodom **"jedan-naprema-jedan"** (engl. *one-againts-one*). Metoda **"jedan-naprema-jedan"** radi na sljedećem principu (Chang, 2011): ako se problem sastoji od *k* klasa, tada se gradi $\frac{k(k-1)}{2}$ klasifikatora gdje svaki trenira podatke samo dviju klasa. Ovdje se primjenjuje osnovno

[3] Matrica je pozitivno semidefinitna ako vrijedi: $\forall x \neq 0, x^T\mathbf{K}x \leq 0$

binarno klasificiranje SVM-om. Nakon klasifikacije, koristi se strategija glasanja gdje se svaki binarni klasifikator smatra glasanjem za cjelokupni set podataka. Na kraju se za pojedini podatak određuje ona klasa koja je dobila najviše glasova.

Funkcije korištene iz biblioteke *LIBSVM* (Chang, 2011) su:

– Funkcija za treniranje **svmtrain** koja je oblika:

```
model = svmtrain (vektor_klasa_podataka_za_treniranje,
matrica_podataka_za_treniranje, 'libsvm_opcije')
```

gdje *'libsvm_opcije'* mogu biti:

- **-s svm_type** : tip korištenog SVM-a
- **-t kernel_type** : tip jezgre SVM-a
- **-d degree** : stupanj u polinomnoj jezgrenoj funkciji
- **-g gamma** : gama u polinomnoj jezgrenoj funkciji
- **-r coef0** : koeficijent u polinomnoj jezgrenoj funkciji
- **-c cost**
-n nu
-p epsilon } parametri ovisni o tipu korištenog SVM-a
- **-m cachesize** : veličina priručne memorije u MB
- **-e epsilon** : osjetljivost kriterija zaustavljanja
- **-h shrinking** : optimizacija
- **-b probability_estimates** : da li je potrebno trenirati model za procjenu vjerojatnosti
- **-wi weight** : postavljanje parametra C klase *i* na weight*C (za model C-SVC)
- **-v n** : n-struka unakrsna validacija
- **-q** : tihi način rada (bez ispisa međukoraka)

– Funkcija za predikciju/testiranje **svmpredict**

```
[vektor_predvidenih_klasa, tocnost, vrijednosti_odluke/
procijenjene_vjerojatnosti] = svmpredict (
vektor_klasa_podataka_za_testiranje,
matrica_podataka_za_testiranje, model, 'libsvm_opcije')
```

ili

```
[vektor_predvidenih_klasa] = svmpredict(
vektor_klasa_podataka_za_testiranje,
matrica_podataka_za_testiranje, model, 'libsvm_opcije')
```

gdje je **model** struktura SVM modela dobivena sa **svmtrain** a *'libsvm_opcije'* mogu biti:

- **-b probability_estimates** : da li je potrebno predvidjeti procjene vjerojatnosti
- **-q** : tihi način rada (bez ispisa međukoraka)

Izlazi koji mogu biti su:

- **vektor_predviđenih_klasa** dobiven predikcijom SVM modela
- **točnost** : vektor sa točnošću, srednjom kvadratnom pogreškom i koeficijentom kvadratne korelacije
- **procjene_vjerojatnosi** : vektor procijenjenih vjerojatnosti, samo ako je odabrana opcija -b

Prije same klasifikacije, potrebno je odrediti koji broj značajki je optimalan kako bi se dobili sto bolji rezultati klasifikacije. Kako je objašnjeno u (Dalbelo-Bašić, 2011), za odabir modela (u ovom slučaju to je predstavljeno odabirom broja značajki) radi se unakrsna provjera nad tri disjunktna skupa:

- skup za učenje **S1**
- skup za provjeru **S2**
- skup za ispitivanje **S3**

Model se uči na skupu **S1**, a pogreška generalizacije računa se na skupu **S2**. To se ponavlja sve dok se ne pronađe optimalan model (u ovom slučaju optimalan broj značajki). Kada se odabere optimalan model na skupu D2, tada se taj model uči na skupu **S1**$\cup$**S2** a pogreška generalizacije tako naučenog modela računa se na skupu **S3**. Ova pogreška je ona koja se objavljuje kao rezultat klasifikacije.
Uz navedeni postupak, koristi se još i k-struka unakrsna provjera. k-struka unakrsna provjera (engl. *k-folded cross validation*) je metoda procjene pogreške gdje se skup primjera podijeli na k disjunktnih particija/preklopa tako da se klasifikator trenira na $k-1$ preklopu dok se ispitivanje vrši na k-tom preklopu. To se ponavlja k puta s pomicanjem ispitnog skupa. Konačna procjena pogreške je prosječna pogreška na k preklopa.

Kombinacijom opisanih postupaka dobiva se **k-struka ugniježđena unakrsna provjera** (engl. *nested k-fold cross validation*) (Dalbelo-Bašić, 2011) jer se u kodu dobivaju dvije ugniježđene petlje:

1. Vanjska petlja za učenje i testiranje
2. Unutarnja petlja za odabir modela (za učenje i provjeru)

Ovim postupkom dobiva se točnija procjena pogreške od drugih metoda jer se odabir modela i konačna pogreška računaju na temelju prosjeka pogreške.
Za kreiranje skupova podataka potrebnih za gore navedeni postupak, korištena je ugrađena MATLAB funkcija *cvpartition(group, 'Kfold', k)*(Mathworks) koja kreira k preklopa takvih da su veličine svakog preklopa iste a omjeri klasa unutar preklopa isti kao i u vektoru *group* koji predstavlja klase svih podataka.

Važno je napomenuti da se opisani postupak posebno pokreče za svaku metodu evaluacije te za svaki odabrani model SVM-a što na kraju rezultira sa osamnaest različitih pokretanja postupka (devet za jajnike i devet za prostatu).

Razvijeni kod priložen je u dodatku E.

6. Rezultati i usporedba metoda

Cjelokupna programska podrška razvijena je na 64-bitnom računalu s dvojezgrenim procesorom Intel® Core™ i5-2410M 2.30GHz i RAM memorijom od 4GB, u programu MATLAB R2014a (verzija 8.3.0.532) na platformi Linux Debian 7.8 (Wheezy).

6.1. Evaluacija modela klasifikacije

Kako bi se mogla odrediti uspješnost pojedinog klasifikacijskog modela, potrebno je provesti evaluaciju. Evaluacija modela klasifikacije se sastoji od nekoliko mjera no prije prikaza spomenutih mjera potrebno je objasniti nekoliko pojmova. **Matrica zabune** (engl. *confusion matrix*) posebna je matrica kojom je moguće vizualizirati koliko se dobro ponaša klasifikacijski model. Ona pokazuje točne i netočne predikcije modela uspoređene s točnim vrijednostima podataka.

STVARNA VRIJEDNOST	PREDVIĐENA VRIJEDNOST: 0	PREDVIĐENA VRIJEDNOST: 1
0	TN	FP
1	FN	TP

Slika 6.1.1: Matrica zabune
(Badgerati, 2010)

Vrijednosti unutar matrice zabune su:

- **TN** (engl. *true negative*) - broj podataka iz klase 0 koji su klasificirani kao klasa 0
- **FP** (engl. *false pozitive*) - broj podataka iz klase 0 koji su klasificirani kao klasa 1

– **FN** (engl. *false negative*) - broj podataka iz klase 1 koji su klasificirani kao klasa 0

– **TP** (engl. *true pozitive*) - broj podataka iz klase 1 koji su klasificirani kao klasa 1

Navedene vrijednosti koriste se za izračun ranije spomenutih mjera evaluacije klasifikatora. Te mjere su (Dalbelo-Bašić, 2011):

– **Točnost** (engl. *accuracy*) - udio točno klasificiranih primjera u skupu svih primjera

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.1.1}$$

– **Preciznost** (engl. *precision*) - udio točno klasificiranih primjera u skupu pozitivno klasificiranih primjera

$$P = \frac{TP}{TP + FP} \tag{6.1.2}$$

– **Odziv** (engl. *recall*), poznat i kao **osjetljivost** (engl. *sensitivity*) - udio točno klasificiranih primjera u skupu svih pozitivnih primjera

$$R = \frac{TP}{TP + FN} \tag{6.1.3}$$

– **Specifičnost** (engl. *specificity*) - udio točno klasificiranih primjera u skupu svih negativnih primjera

$$SPEC = \frac{TN}{TN + FP} \tag{6.1.4}$$

– **F-mjera** (engl. *F-mean*) - harmonijska sredina preciznosti i odziva

$$F = \frac{2}{\frac{1}{P} + \frac{1}{R}} = \frac{2PR}{P + R} \tag{6.1.5}$$

U općem slučaju važnost preciznosti i odziva kontrolira se parametrom β (ako je važniji odziv koristi se veća vrijednost parametra β)

$$F_\beta = \frac{(a + \beta^2)PR}{\beta^2 P + R} \tag{6.1.6}$$

Za potrebe evaluacije klasifikacijskih modela, razvijena je funkcija koja računa gore spomenute mjere uzimajući u ozbir višeklasno klasificiranje. Spomenuta funkcija prikazana je u Dodatku E.

6.2. Dobiveni rezultati

Svi rezultati su predstavljeni u obliku *srednja vrijednost* $\pm$ *standardna devijacija*. Kako je spomenuto u (Dalbelo-Bašić, 2011), kod višeklasne klasifikacije uvijek vrijedi **FP=FN** što povlači **P=R=F-mjera** što se može i primijetiti u rezultatima prikazanim u nastavku.

6.2.1. Jajnici

Entropija

Tablica 6.2.1: Rezultati za evaluaciju značajki raka jajnika uz pomoć entropije

Model	Točnost [%]	Preciznost [%]	Odziv [%]	Specifičnost [%]	F-mjera [%]
Linearni SVM	93.22 ± 2.77	89.83 ± 4.16	89.83 ± 4.16	94.92 ± 2.08	89.83 ± 4.16
Polinomni SVM	91.67 ± 2.8	87.51 ± 4.2	87.51 ± 4.2	93.75 ± 2.1	87.51 ± 4.2
Radijalni SVM	74.36 ± 3.37	61.54 ± 5.06	61.54 ± 5.06	80.77 ± 2.53	61.54 ± 5.06

Gini Diversity Index

Tablica 6.2.2: Rezultati za evaluaciju značajki raka jajnika uz pomoć GDI-a

Model	Točnost [%]	Preciznost [%]	Odziv [%]	Specifičnost [%]	F-mjera [%]
Linearni SVM	94.16 ± 4.2	91.25 ± 6.29	91.25 ± 6.29	95.62 ± 3.15	91.25 ± 6.29
Polinomni SVM	95.07 ± 2.97	92.6 ± 4.45	92.6 ± 4.45	96.3 ± 2.23	92.6 ± 4.45
Radijalni SVM	92.9 ± 2.34	89.36 ± 3.51	89.36 ± 3.51	94.68 ± 1.76	89.36 ± 3.51

Fisher Score

Tablica 6.2.3: Rezultati za evaluaciju značajki raka jajnika uz pomoć FS-a

Model	Točnost [%]	Preciznost [%]	Odziv [%]	Specifičnost [%]	F-mjera [%]
Linearni SVM	95.06 ± 2.03	92.59 ± 3.04	92.59 ± 3.04	96.29 ± 1.52	92.59 ± 3.04
Polinomni SVM	96.92 ± 2.17	95.38 ± 3.26	95.38 ± 3.26	97.69 ± 1.63	95.38 ± 3.26
Radijalni SVM	84.86 ± 4.99	77.29 ± 7.49	77.29 ± 7.49	88.65 ± 3.74	77.29 ± 7.49

6.2.2. Prostata

Entropija

Tablica 6.2.4: Rezultati za evaluaciju značajki raka prostate uz pomoć entropije

Model	Točnost [%]	Preciznost [%]	Odziv [%]	Specifičnost [%]	F-mjera [%]
Linearni SVM	95.86 ± 2.45	93.78 ± 3.67	93.78 ± 3.67	96.89 ± 1.84	93.78 ± 3.67
Polinomni SVM	79.1 ± 1.19	68.64 ± 1.78	68.64 ± 1.78	84.32 ± 0.89	68.64 ± 1.78
Radijalni SVM	72.67 ± 0.33	59.01 ± 0.5	59.01 ± 0.5	79.5 ± 0.25	59.01 ± 0.5

Gini Diversity Index

Tablica 6.2.5: Rezultati za evaluaciju značajki raka prostate uz pomoć GDI-a

Model	Točnost [%]	Preciznost [%]	Odziv [%]	Specifičnost [%]	F-mjera [%]
Linearni SVM	96.9 ± 1.03	95.35 ± 1.54	95.35 ± 1.54	97.67 ± 0.77	95.35 ± 1.54
Polinomni SVM	96.69 ± 1.52	95.04 ± 2.28	95.04 ± 2.28	97.52 ± 1.14	95.04 ± 2.28
Radijalni SVM	83.44 ± 2.26	75.16 ± 3.39	75.16 ± 3.39	87.58 ± 1.69	75.16 ± 3.39

Fisher Score

Tablica 6.2.6: Rezultati za evaluaciju značajki raka prostate uz pomoć FS-a

Model	Točnost [%]	Preciznost [%]	Odziv [%]	Specifičnost [%]	F-mjera [%]
Linearni SVM	95.44 ± 1.21	93.16 ± 1.81	93.16 ± 1.81	96.58 ± 0.91	93.16 ± 1.81
Polinomni SVM	88.82 ± 3.83	83.24 ± 5.74	83.24 ± 5.74	91.62 ± 2.87	83.24 ± 5.74
Radijalni SVM	82.20 ± 1.63	73.30 ± 2.44	73.30 ± 2.44	86.65 ± 1.22	73.30 ± 2.44

7. Zaključak

Promatranjem dobivenih rezultata, može se zaključiti da su korištene metode vrlo dobre no dobrodošla su poboljšanja samog algoritma. Kao što se vidi u prethodnom poglavlju, dobiveni rezultati u klasifikaciji raka prostate u nekim situacijama znatno su lošiji od dobivenih rezultata u klasifikaciji raka jajnika. Razlog tome mogao bi biti u različitim intenzitetima značajki vezanih za rak prostate te shodno tome potrebno bi bilo modificirati korištene metode obrade signala i evaluacije značajki kako bi se bolje razlikovale takve značajke.

Također, postoji mogućnost proširenja rada tako da se razvije interaktivna aplikacija, samostalna ili kao web sučelje, gdje bi novi pacijent mogao učitati svoj spektrogram dobiven SELDI-TOF metodom te pokrenuti klasifikaciju kako bi se dobila prognoza zdravlja. Ovakva aplikacija ima potencijala za korištenje pri ranom otkrivanju raka zbog svoje brzine. Zbog brzine SELDI-TOF metode i uzoraka koje koristi, pacijent dobiva spektrogram kroz kratko vrijeme te pokretanjem aplikacije vrlo brzo može dobiti prognozu. Ovisno o prognozi, preporučile bi se daljnje pretrage za potvrđivanje ili opovrgavanje dobivenih rezultata. Ovime bi se ubrzao početak liječenja jer se isto može početi provoditi u roku par dana od početne dijagnoze za razliku od sada korištenih pretraga gdje se rezultati mogu čekati i po nekoliko tjedana što povlači da i liječenje počinje tek za nekoliko tjedana.

Nikolina Očić

Vlastoručni potpis

LITERATURA

Badgerati. Machine Learning - Confusion Matrix, 2010. URL https://computersciencesource.wordpress.com/2010/01/07/year-2-machine-learning-confusion-matrix/. posljednji pristup: Lipanj 2015.

Bilušić, M. Tumorski biljezi (tumorski markeri), Travanj 2013. URL http://www.cybermed.hr/clanci/tumorski_biljezi_tumorski_markeri. posljednji pristup: Svibanj 2015.

CCR. Clinical proteomics program databank, 2002. URL http://home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp. posljednji pristup: Svibanj 2015.

Chang, C.-C. i Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Cybermed. Rak jajnika, Kolovoz 2013. URL http://www.cybermed.hr/centri_a_z/rak_jajnika. posljednji pristup: Lipanj 2015.

Cybermed. Rak prostate, Siječanj 2014. URL http://www.cybermed.hr/centri_a_z/rak_prostate. posljednji pristup: Lipanj 2015.

Dalbelo-Bašić, B. i Šnajder, J. Vrednovanje klasifikatora, 2011. URL http://www.fer.unizg.hr/_download/repository/SU-12-VrednovanjeKlasifikatora.pdf. posljednji pristup: Lipanj 2015.

Domančić, M. Sustav za analizu tumorskih biljega. Završni rad br. 3315, Fakultet elektrotehnike i računarstva, Zagreb, 2013.

Fisher, R. A. *Statistical methods for research workers*. Oliver & Boyd, Edinburgh, 1925.

Gogić, A. Sustav za dijagnosticiranje tumora. Završni rad br. 3312, Fakultet elektrotehnike i računarstva, Zagreb, 2013.

Gu, Q., Li, Z., i Han, J. Generalized Fisher Score for feature selection. *CoRR*, abs/1202.3725, 2012. URL http://arxiv.org/abs/1202.3725.

Hrvatska Enciklopedija. Entropija. URL http://www.enciklopedija.hr/Natuknica.aspx?ID=18042. posljednji pristup: Lipanj 2015.

Huang, T., Kecman, V., i Kopriva, I. *Kernel Based Algorithms for Mining Huge Data Sets: Supervised, Semi-supervised, and Unsupervised Learning*. Studies in Computational Intelligence. Springer, 2006. ISBN 9783540316817.

Intel. Wavelet transform functions. URL https://software.intel.com/en-us/node/502357. posljednji pristup: Lipanj 2015.

Investopedia. Gini index. URL http://www.investopedia.com/terms/g/gini-index.asp. posljednji pristup: Lipanj 2015.

Jost, L. Entropy and diversity. *Oikos*, 113(2):363–375, 2006. URL http://dx.doi.org/10.1111/j.2006.0030-1299.14714.x.

Marsland, S. *Machine learning : an algorithmic perspective*. CRC Press, 2009.

Mathworks. cvpartition. URL http://www.mathworks.com/help/stats/cvpartition.html. posljednji pristup: Lipanj 2015.

Nordqvist, C. What is a tumor?, Rujan 2014. URL http://www.medicalnewstoday.com/articles/249141.php. posljednji pristup: Lipanj 2015.

Pei, L., Liu, J., Guinness, R., Chen, Y., Kuusniemi, H., i Chen, R. Using LS-SVM Based Motion Recognition for Smartphone Indoor Wireless Positioning. *Sensors*, 12(5):6155–6175, 2012. poglavlje 4.1: Support Vector Machines.

Petricoin, E. F., Ardekani, A., Hitt, B. A., Levine, P. J., Fusaro, V. A., Steinberg, S. M., Mills, G. B., Simone, C., Fishman, D. A., Kohn, E. C., i Liotta, L. A. Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet*, 359(9306):572–577, 2002a.

Petricoin, E. F., Ornstein, D. K., Paweletz, C. P., Ardekani, A., Hackett, P. S., Hitt, B. A., Velassco, A., Trucco, C., Wiegand, L., Wood, K., Simone, C. B., Levine, P. J., Linehan, W. M., Emmert-Buck, M. R., Steinberg, S. M., Kohn, E. C., i Liotta, L. A. Serum proteomic patterns for detection of prostate cancer. *Journal of the National Cancer Institute*, 94(20):1576–1578, 2002b.

Seibert, V., Wiesner, A., Buschmann, T., i Meuer, J. Surface-enhanced laser desorption ionization time-of-flight mass spectrometry (seldi tof-ms) and proteinchip technology in proteomics research. *Pathology, research and practice*, 200(2):83–94, 2003.

Sokolova, M. i Lapalme, G. A systematic analysis of performance measures for classification tasks. *Information Processing and Management*, 45(4):427–437, 2009.

Stack Exchange. Difference between a wavelet transform and a wavelet decomposition. URL `http://dsp.stackexchange.com/questions/10675/difference-between-a-wavelet-transform-and-a-wavelet-decomposition`. posljednji pristup: Lipanj 2015.

Ueltschi, D. Introduction to Statistical Mechanics. Chapter 6 - Shannon Entropy, 2006. URL `http://www.ueltschi.org/teaching/chapShannon.pdf`.

World Bank. GINI Index. URL `http://data.worldbank.org/indicator/SI.POV.GINI`. posljednji pristup: Lipanj 2015.

Šnajder, J. i Dalbelo-Bašić, B. Strojno učenje, 2014. Skripta za predavanja.

Pouzdana klasifikacija tumorskih markera

Sažetak

Predstavljena je i opisana problematika ranog otkrivanja tumora te je opisana problematika tumora jajnika i prostate kao središta razvijenog sustava. Opisana je važnost razvijenog sustava kao pomoći za rano otkrivanje tumora. Prikazan je format ulaznih podataka te objašnjena metoda za dobivanje istih. Razvijen je sustav koji uz pomoć metoda za obradu signala te strojnog učenja klasificira ulazne podatke u kategorije zdravlja. Objašnjena je metoda korištena za obradu podataka te metode korištene za evaluaciju istih. Također je temeljito objašnjena metoda korištena za izgradnju modela klasifikacije. Prikazani su i objašnjeni dobiveni rezultati u obje grupe tumora.

Ključne riječi: tumori, SELDI-TOF, strojno učenje, Haarov valićni paket, entropija, Gini Diversity Index, Fisher Score, stroj potpornih vektora

Reliable classification of tumor markers

Abstract

The problem of early cancer detection is presented and described. The problems of ovary and prostate cancer, as they are the center of developed system, are also described. Importance of developed system as a helping tool for early cancer detection is presented. Format of input data and corresponding method of data acquiring are presented. Developed system, which is based upon methods of signal processing and machine learning, classifies input data into health categories. Method for signal processing and multiple methods for evaluating data are explained. Also, model for classification is thoroughly explained. Obtained results are shown and explained for both cancer groups.

Keywords: tumors, SELDI-TOF, machine learning, Haar's wavelet packet, entropy, Gini Diversity Index, Fisher Score, support vector machine

A. Obrada ulaznog signala

A.1. Učitavanje podataka za jajnike

```
function [S,K] = Ucitaj()
    %funkcija ucitava podatke iz datoteka .cvs
    %izlazne vrijednosti su:
    %   S - matrica ulazni podataka
    %   K - vektor labela odnosno klasa za svaki ulazni podataka

    S1=UcitajPodatke('jajnici/benign/');
    S2=UcitajPodatke('jajnici/cancer/Group A/');
    S3=UcitajPodatke('jajnici/cancer/Group B/');
    S4=UcitajPodatke('jajnici/control/Group C/');
    S5=UcitajPodatke('jajnici/control/Group D/');
    S=[S1 S2 S3 S4 S5]';
    K=1:1:216;
    K(1:16)=1;
    K(17:116)=2;
    K(117:216)=3;
    K=K';
end

function S=UcitajPodatke(dirName)
    S = [];
    lista = dir(dirName);
    lista = lista(3:end);
    for i = 1 : length(lista)
        x = strcat(dirName,lista(i).name);
        Z = csvread(x,1,0);
        S(:,i)=Z(:,2);
    end
end
```

A.2. Učitavanje podataka za prostatu

```
function [S,K] = UcitajP()
    %funkcija ucitava podatke iz datoteka .cvs
    %izlazne vrijednosti su:
    %   S - matrica ulazni podataka
    %   K - vektor labela odnosno klasa za svaki ulazni podataka

    S1=UcitajPodatke('prostata/benign/');
    S2=UcitajPodatke('prostata/cancer/');
    S3=UcitajPodatke('prostata/control/');
    S=[S1 S2 S3]';
    K=1:1:322;
    K(1:190)=1;
    K(191:259)=2;
    K(260:322)=3;
    K=K';

end

function S=UcitajPodatke(dirName)
    S = [];
    lista = dir(dirName);
    lista = lista(3:end);
    for i = 1 : length(lista)
        x = strcat(dirName,lista(i).name);
        Z = csvread(x,1,0);
        S(:,i)=Z(:,2);
    end
end
```

A.3. Haarova valićna transformacija

```
function [ m ] = Wavelet_k( x )
    %funkcija racuna Haarovu valicnu transformaciju nad
    %vektorom ulaznih podataka x
    %izlaz je matrica valicnih paketa m dimenzija 2^N x N
    %(gdje je N prva potencija broja 2 koja odgovara
    %duljini ulaznog vektora x)

    depth = ceil(log2(length(x)));
    len = 2^depth;
    data=[];
    for i=1:len
        if i<length(x)
            data(i)=x(i);
        else
            data(i)=0;
        end
    end

    m=zeros(len,depth+1);
    for j=1:len
        m(j,1)=data(j);
    end

    dokud=len;
    k=1;
    for i=2:depth+1
        pom=k;
        redak=1;
        otkud=1;
        dokud=len;
        koef=dokud/k;
        dokud=koef;
        pos=1;
        while pom>0
            while pos<dokud
                m(redak,i)=(m(pos,i-1)+m(pos+1,i-1)) / sqrt(2);
                pos=pos+2;
                redak=redak+1;
            end
```

```
            pos=otkud;
            while pos<dokud
                m(redak,i)=(m(pos,i-1)-m(pos+1,i-1)) / sqrt(2);
                pos=pos+2;
                redak=redak+1;
            end;
            otkud=otkud+koef;
            dokud=dokud+koef;
            pom=pom-1;
        end
        k=k*2;
    end
end
```

B. Evaluacija značajki entropijom

```
function [ vector1, donjaG1, gornjaG1, stupac1, sve_matrice] =
Entropy_vector( M )
    %funkcija racuna bazni vektor za odabir valicnih
    %paketa pogodnih za klasifikaciju
    %kao ulaz prima matricu podataka svih pacijenata
    %kao izlaz vraca bazni vektor, vektor gornjih
    %granica paketa, vektor donjih granica paketa
    %te vektor stupaca vaznih paketa

    sve_matrice = napravi_valicne_matrice( M );
    map = [];
    i=1;
    vanjskaD = 1;
    vanjskaG = 16384;
    dG=[];
    gG=[];
    st = [];
    k=1;

    while i < 15
        otkud = vanjskaD;
        dokud = vanjskaG;
        korak = dokud/k;
        dokud = korak;
        while dokud<=vanjskaG
            if length(dG)>0
                nasao=0;
                for l=1:length(dG)
                    if (dG(l)==otkud || gG(l)==dokud ||
                    (dG(l)<=otkud && gG(l)>=dokud))
                        nasao=1;
                        break;
```

```
                end
            end
            if nasao==1
                otkud=otkud+korak;
                dokud=dokud+korak;
                continue
            end
        end

        [en1, en2, en3] = racunaj_entropije_prosjek
        (sve_matrice, otkud, dokud, korak, i);

        if en1 < en2 + en3
            map = [map en1];
            dG = [dG otkud];
            gG = [gG dokud];
            st = [st i];
            if i==1
                vector1 = map;
                donjaG1=dG;
                gornjaG1=gG;
                stupac1=st;
                return
            else
                otkud=otkud+korak;
                dokud=dokud+korak;
            end
        else
             if i==14
                map = [map en2 en3];
                dG = [dG otkud otkud+korak/2];
                gG = [gG otkud+korak/2-1 dokud];
                st = [st i+1 i+1];
            end
            otkud = otkud + korak;
            dokud = dokud + korak;
        end
    end
    i=i+1;
    k=k*2;
end
```

```
    [vector1,donjaG1,gornjaG1,stupac1] = sredi_vektor
    (map,dG,gG,st);

end

function [ matrice ] = napravi_valicne_matrice( S )
   %funkcija sluzi za racunanje valicnih matrica
   %svih pacijenata te kao izlaz daje trodimenzionalnu
   %matricu koja sadrzi matrice valicnih paketa
   %svih pacijenata

    matrice=zeros(16384,15,216);
    for i=1:216
        pacijent = S(i,:);
        val=Wavelet_k(pacijent);
        matrice(:,:,i)=val;
    end
end

function [e_o, e_d1, e_d2] = racunaj_entropije_prosjek(matrice,
otkud, dokud, korak, s)
    %funkcija racuna entropije paketa "oca" i
    %paketa "djece" koje se koriste za
    %odredivanje baznog vektora paketa

    e_o = 0.0;
    e_d1 = 0.0;
    e_d2 = 0.0;
    for i = 1 : 216
        mat = matrice(:,:,i);
        e_o = e_o + wentropy(mat(otkud:dokud,s),'shannon');
        e_d1 = e_d1 + wentropy(mat(otkud:otkud+korak/2-1,s+1),
        'shannon');
        e_d2 = e_d2 + wentropy(mat(otkud+korak/2:dokud,s+1),
        'shannon');
    end
    e_o = e_o / 216.0;
    e_d1 = e_d1 / 216.0;
    e_d2 = e_d2 / 216.0;
end
```

```
function [ v_b, d, g, stupac ] = sredi_vektor ( map, dG, gG, st)
    %funkcija sortira ulazni vektor te shodno
    %tome sortira vektore donjih i gornjih granica

    v_b=[];
    d=[];
    g=[];
    stupac=[];
    dokud=length(dG);
    for i = 1 : dokud
        [value,position]=min(dG);
        d(i)=value;
        dG(position)=[];
        g(i)=gG(position);
        gG(position)=[];
        stupac(i)=st(position);
        st(position)=[];
        v_b(i)=map(position);
        map(position)=[];
    end
end
```

C. Evaluacija značajki Gini Diversity Index-om

```
function [ B, M, Z ] = Gini_matrix ( S )
    %funkcija racuna vrijednosti Gini Diversity Indexa
    %kao ulaz prima matricu podataka svih pacijenata
    %kao izlaz daje tri matrice s GDI vrijednostima, za
    %svaku klasifikaciju po jednu matricu

    B = zeros(16384,15);
    M = zeros(16384,15);
    Z = zeros(16384,15);
    index = randperm(216);
    S = S(index,:);
    sve_matrice = napravi_valicne_matrice( S );

    for i=1:16384
        for j=1:15
            niz = zeros(1,216);
            for k=1:216
                niz(k) = sve_matrice(i,j,k);
            end

            [niz indexi]=sort(niz);
            b = promjeni(indexi, 'benigni',index);
            B(i,j)=gini_index(b);

            m = promjeni(indexi, 'maligni',index);
            M(i,j)=gini_index(m);

            z = promjeni(indexi, 'zdravi',index);
            Z(i,j) = gini_index(z);
```

```
        end
    end
end

function [ matrice ] = napravi_valicne_matrice( S )
   %funkcija sluzi za racunanje valicnih matrica
   %svih pacijenata te kao izlaz daje trodimenzionalnu
   %matricu koja sadrzi matrice valicnih paketa
   %svih pacijenata

    matrice=zeros(16384,15,216);
    for i=1:216
        pacijent = S(i,:);
        val=Wavelet_k(pacijent);
        matrice(:,:,i)=val;
    end
end

function [ ind ] = promjeni (in, klasa, index)
    %funkcija podesava vektor klasa ovisno o
    %klasifikaciji za koju se racunaju vrijednosti GDI

    ind = zeros(1,216);
    if strcmp(klasa,'benigni')
        for i=1:216
            if index(i)<17
                a=find(in==i);
                ind(a)=1;
            end
        end
    end
    if strcmp(klasa,'maligni')
        for i=1:216
            if index(i)>16 && index(i)<117
            a=find(in==i);
            ind(a)=1;
            end
        end
    end
    if strcmp(klasa,'zdravi')
        for i=1:216
```

```
            if index(i)>116
            a=find(in==i);
            ind(a)=1;
            end
        end
    end
end

function [ g ] = gini_index ( x )
    %funkcija racuna samu vrijednost pojedinog
    %elementa GDI matrice

    nx = length(x);
    gini_all = zeros(nx-1, 2);
    for split = 1:nx-1
        gini_all(split,1) = sum(x(1:split)) * (split -
        sum(x(1:split))) /split / nx;
        gini_all(split,2) = sum(x(split+1:end)) * ((nx-split) -
        sum(x(split+1:end))) / (nx-split) / nx;
    end
    sum_of_ginis = sum(gini_all,2);
    g = min(sum_of_ginis);
end
```

D. Evaluacija značajki Fisher Score-om

```
function [ F ] = Fisher_matrix ( S , K )
    %funkcija racuna vrijednosti Fisher Score-a
    %kao ulaz prima matricu podataka svih
    %pacijenata te vektor labela/klasa svih pacijenata
    %kao izlaz daje matricu s vrijednostima FS
    %dimenzija istih kao i matrica valicnih paketa

    F = zeros(16384,15);
    num_ben = sum(K==1);
    num_mal = sum(K==2);
    num_zdr = sum(K==3);
    num_p = length(K);
    vec_all = zeros(num_p,1);
    vec_feat1 = zeros(num_ben,1);
    vec_feat2 = zeros(num_mal,1);
    vec_feat3 = zeros(num_zdr,1);

    mat_all = napravi_valicne_matrice( S );

    for i=1:16384
        for j=1:15
            br1 = 1;
            br2 = 1;
            br3 = 1;

            for k=1:num_p
                vec_all(k) = mat_all(i,j,k);
                switch K(k)
                    case 1
```

```
                        vec_feat1(br1) = mat_all(i,j,k);
                        br1 = br1 + 1;
                    case 2
                        vec_feat2(br2) = mat_all(i,j,k);
                        br2 = br2 + 1;
                     case 3
                        vec_feat3(br3) = mat_all(i,j,k);
                        br3 = br3 + 1;
                     otherwise
                end
            end

            mean_all = mean(vec_all);
            mean1 = mean(vec_feat1);
            std1 = std(vec_feat1);
            mean2 = mean(vec_feat2);
            std2 = std(vec_feat2);
            mean3 = mean(vec_feat3);
            std3 = std(vec_feat3);

            pom = (num_ben*(mean1 - mean_all).^2 + num_mal*(mean2
            - mean_all).^2 + num_zdr*(mean3 - mean_all).^2) /
            (num_ben*(std1).^2 + num_mal*(std2).^2 +
            num_zdr*(std3).^2);

            if isnan(pom)
                F(i,j) = 0;
            else
                F(i,j) = pom;
            end
        end
    end
end

function [ matrice ] = napravi_valicne_matrice( S )
   %funkcija sluzi za racunanje valicnih matrica
   %svih pacijenata te kao izlaz daje trodimenzionalnu
   %matricu koja sadrzi matrice valicnih paketa
   %svih pacijenata

    matrice=zeros(16384,15,216);
```

```
    for i=1:216
        pacijent = S(i,:);
        val=Wavelet_k(pacijent);
        matrice(:,:,i)=val;
    end
end
```

E. Odabir broja značajki i SVM

```
function [ev_gen] = znacajke_evaluacijaP (Kp,Fp,MatAllp,jezgra)
  %funkcija koja provodi k-struku ugnijezdenu unakrsnu provjeru
  %kao ulaz prima vektor labela, matricu iznosa Fisher Score-ova
  %za pojedini valicni paket, matricu svih valicnih paketa
  %te odabir jezgre SVM modela
  %kao izlaz funkcija daje strukturu, odnosno vektor koji sadrzi
  %mjere evaluacije SVM modela

  err_gen = zeros(5,1);
  ev_gen = zeros(5,6);
  br_znacajki = zeros(5,1);
  cvpart = cvpartition(Kp,'Kfold',5);

  for i=1:5
      x = zeros((16384/16),1);
      m = 1;
      err_znacajke = zeros((16384/16),1);

      trIdx = cvpart.training(i);
      ind_pod = zeros(nnz(trIdx),1);
      w = 1;

      for k=1:length(trIdx)
          if(trIdx(k) == 1)
              ind_pod(w) = k;
              w = w+1;
          end
      end
      teIdx = cvpart.test(i);

      for n=16:16:16384
          err_unutarnja = 0.0;
```

```
vF = Odaberi_Fisher(Fp,n);
izvuceni2 = izvuci(vF, MatAllp);
izv2 = scale(izvuceni2,1,-1);

cvpart2 = cvpartition(Kp(trIdx,:),'Kfold',10);

for j=1:10
    train2 = cvpart2.training(j);
    ind_pod2 = zeros(nnz(train2),1);
    w = 1;
    for k=1:length(train2)
        if(train2(k) == 1)
            ind_pod2(w) = ind_pod(k);
            w = w+1;
        end
    end

    test2 = cvpart2.test(j);
    ind_neg2 = zeros(nnz(test2),1);
    w = 1;
    for k=1:length(test2)
        if(test2(k) == 1)
            ind_neg2(w) = ind_pod(k);
            w = w+1;
        end
    end

    testInd = zeros(length(Kp),1);
    trainInd = zeros(length(Kp),1);

    for p=1:length(trainInd)
        if( ismember(p,ind_pod2))
            trainInd(p) = 1;
        else
            trainInd(p) = 0;
        end
    end

    for p=1:length(testInd)
        if( ismember(p,ind_neg2))
```

```
                testInd(p) = 1;
            else
                testInd(p) = 0;
            end
        end

        trInd2 = logical(trainInd);
        teInd2 = logical(testInd);

        Xtrain2=izv2(trInd2);
        Ytrain2=Kp(trInd2);
        Xtest2=izv2(teInd2);
        Ytest2=Kp(teInd2);

        switch(jezgra)
            case 0
                svm=libsvmtrain(Ytrain2,Xtrain2,
                '-s 0 -t 0 -q');
                [~, acc, ~] = libsvmpredict(Ytest2,
                Xtest2,svm);
            case 1
                svm=libsvmtrain(Ytrain2,Xtrain2,
                '-s 0 -t 1 -q');
                [~, acc, ~] = libsvmpredict(Ytest2,
                Xtest2,svm);
            case 2
                svm=libsvmtrain(Ytrain2,Xtrain2,
                '-s 0 -t 2 -q');
                [~, acc, ~] = libsvmpredict(Ytest2,
                Xtest2,svm);
            otherwise
        end
        err_unutarnja = err_unutarnja + acc(2);
    end

    x(m) = n;
    err_znacajke(m) = err_unutarnja/10;
    m = m + 1;

end
```

```
        [~, indexi] = sort(err_znacajke,'ascend');
        br_znacajki(i) = x(indexi(1));

        vF = Odaberi_Fisher(Fp,br_znacajki(i));
        izvuceni3 = izvuci(vF, MatAllp);
        izv3 = scale(izvuceni3,1,-1);

        Xtrain=izv3(trIdx,:);
        Ytrain=Kp(trIdx,:);
        Xtest=izv3(teIdx,:);
        Ytest=Kp(teIdx,:);

        switch(jezgra)
            case 0
                svm1=libsvmtrain(Ytrain,Xtrain, '-s 0 -t 0 -q');
                [res1, acc1, ~] = libsvmpredict(Ytest,Xtest,svm1);
            case 1
                svm1=libsvmtrain(Ytrain,Xtrain, '-s 0 -t 1 -q');
                [res1, acc1, ~] = libsvmpredict(Ytest,Xtest,svm1);
            case 2
                svm1=libsvmtrain(Ytrain,Xtrain, '-s 0 -t 2 -q');
                [res1, acc1, ~] = libsvmpredict(Ytest,Xtest,svm1);
            otherwise
        end

        err_gen(i) = acc1(2);
        ev_gen(i,:) = Evaluate_Multi(Ytest, res1);
    end
end
```

E.1. Evaluacija SVM modela

```
function eval = Evaluate_Multi(ACTUAL, PREDICTED)
    %funckija racuna vrijednosti mjera za evaluaciju
    %modela klasifikacije
    %kao ulaz prima vektor stvarnih klasa podataka
    %te vektor predvidenih klasa podataka
    %kao izlaz daje vektor, odnosno strukturu koja
    %sadrzi vrijednosti svih mjera
```

```
    klase = length(unique(ACTUAL));
    TP = zeros(klase,1);
    FP = zeros(klase,1);
    TN = zeros(klase,1);
    FN = zeros(klase,1);

    for i=1:klase
        for j=1:length(PREDICTED)
            if(PREDICTED(j) == i && ACTUAL(j) == i)
                TP(i) = TP(i) + 1;
            elseif(PREDICTED(j) == i && ACTUAL(j) ~= i)
                FP(i) = FP(i) + 1;
            elseif(PREDICTED(j) ~= i && ACTUAL(j) ~= i)
                TN(i) = TN(i) + 1;
            elseif(PREDICTED(j) ~= i && ACTUAL(j) == i)
                FN(i) = FN(i) + 1;
            end
        end
    end

    TP_uk = sum(TP);
    FP_uk = sum(FP);
    FN_uk = sum(FN);
    TN_uk = sum(TN);

    accuracy = (TP_uk + TN_uk) / (TP_uk + FP_uk + FN_uk + TN_uk);
    precision = TP_uk / (TP_uk + FP_uk);
    recall = TP_uk / (TP_uk + FN_uk);
    specificity = TN_uk / (TN_uk + FP_uk);

    F_mjera = 2 * precision * recall / (precision + recall);

    eval = [accuracy precision recall specificity F_mjera ];
end
```