SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 5391

Analiza molekularnog potpisa tumora uz pomoć strojnog učenja

Josip Jukić

Zagreb, lipanj 2018.

Hvala svima koji su mi pomogli u stjecanju znanja.

SADRŽAJ

POPIS SLIKA

POPIS TABLICA

1. Uvod

U dosadašnjoj prošlosti pokazalo se da čovječanstvo neprestano napreduje promatra li se opći dojam krivulje uspona i padova. Uz veliki napredak, javila se i raznolikost područja ljudskog interesa. Sami primjeri mogu se uočiti u područjima znanosti, tehnologije, sporta te društvenim istraživanjima. Ogromna količina ljudskih aktivnosti utjecala je na okolinu koja se posljedično značajno promijenila. Evolucija ljudi i njihovih aktivnosti popraćena je razvojem ostalih živih bića.

Premda živimo u trenutcima kada su razvijene vrlo sofisticirane metode i lijekovi u medicini, ljudski životi i dalje bivaju ugroženi zbog nepoznavanja uzroka bolesti. Uspješnost patogena, odnosno bioloških agenata koji su odgovorni za bolest organizama, održala se zbog napretka njihovih mehanizama i vrsta. Goruć problem u dijagnostici preraznolika je paleta mogućih uzročnika i medicinskih stanja. Odsustvo dijagnoze sprječava adekvatnu terapiju koja bi mogla pomoći pacijentu. Pogrešne dijagnoze dodatno mogu znatno pogoršati pacijentovo stanje.

Jedan od načina da se doskoči navedenom problemu je razvoj pouzdanih metoda za dijagnostiku uz pomoć računala. U ovom radu se koncentrira na prepoznavanje poteškoća u fiziološkoj regulaciji kontrolnih mehanizama rasta stanica ljudskog organizma. Patološke tvorbe koje nastaju zbog poremećaja tih mehanizama uslijed prekomjernog umnažanja abnormalnih stanica nazivaju se novotvorine ili tumori (lat. tumor - oteklina). Pronalaženje točne vrste tumora najbitniji je korak u procesu liječenja jer su terapije većinom ustaljene. U slučaju ranog prepoznavanja uzroka, omogućuje se korištenje manje invazivnih metoda i povećava se uspješnost liječenja. Ako se na intuitivnoj razini promatraju tumorske stanice, može ih se poistovjetiti s prijetnjom koja se povećava ili usložnjava s vremenom. Sasvim prirodno je zaključiti da se s takvom vrstom opasnosti bolje suočiti što ranije je moguće.

Svojstva tumora skrivena su potencijalno u njihovom genotipu. Zbog brojnih mutacija tijekom vremena, geni odgovorni za reprodukciju i regulaciju rasta stanice izmi-

jenjeni su te ne mogu provoditi ustaljene mehanizme na ispravan način. Mogućnost prepoznavanja tumora ostvariva je analizom genskog materijala. Navedena spoznaja usmjerila je ideju metode ka klasifikaciji tumorskih stanica koja polaže temelje upravo na svojstvima njihovih genskih sekvenci.

Zbog ogromnog broja i veličine sekvenci gena (ljudski genom [1] sadrži oko $30\,000$ gena, dok je prosječna duljina gena između 600 i 1800 parova nukleotida) potrebno je odabrati moćan klasifikator. Kao pobjednik izašao je stroj potpornih vektora (SVM). Pojedine sastavnice klasifikatora dodatno su optimizirani genetskim algoritmom kako bi se izašlo na kraj s veličinom i složenošću konkretnih podataka u problemu. Složenost cjelokupnog zadatka leži u suptilnoj razlici između sekvenci pojedinih tumorskih stanica, stoga je od presudne važnosti uporaba strojnog učenja.

Krajnji cilj je ispravna klasifikacija tumora na temelju uzorka, primjerice dobivenog biopsijom[2]. Analiza svojstava može se vršiti naknadno, jednom kad su pojedini tumori razdvojeni u pripadne razrede.

U nastavku rada u poglavlju 2 opisani su biološki koncepti potrebni za upoznavanje s problemom. Nakon uvodnih pojmova, predstavljen je stroj potpornih vektora u poglavlju 3. Detaljnije je prikazana pozadina rada stroja te slijed razvoja same ideje. U sadržajno povezanom poglavlju 4 predstavljena je izgradnja klasifikatora tumorskih stanica. Postupak je rastavljen na manje dijelove koji se kasnije povezuju u jednu cjelinu. Poslije konstrukcije prikazani su rezultati i uspješnost klasifikatora u pogavlju 5. Izvršena je analiza korištene metode, a pripadni rezultati povezani su s određenim značajkama metode. Naglašene su prednosti i mane rabljenih algoritama. U posljednjem poglavlju 6 sadržaj rada zaokružen je u jednu cjelinu te su kratko predstavljene buduće mogućnosti i nadogradnje.

[1] ukupni genetski materijal nekog organizma

[2] medicinska tehnika uzimanja stanica ili tkiva radi ispitivanja

2. Biološki koncepti

2.1. Sekvenciranje

Prvi korak u analizi velikog broja bioloških interakcija u organizmu je sekvenciranje gena. Ekstrakcijom genetske informacije iz lanaca DNA ili RNA, moguće je pronaći poveznice i obrasce nasljednih bolesti, patogenih toksina, infekcija, tumora te ostalih brojnih međudjelovanja organizama [1]. Sekvenciranje je proces određivanja pojedinih nukleinskih baza unutar DNA ili RNA lanca. Među bazama razlikujemo: adenin (A), citozin (C), gvanin (G), timin (T) i uracil (U). Rezultati sekvenciranja mogu se iskoristiti za utvrđivanje samog genoma kao i za povezivanje sekvenci s pojedinim vrstama tumora.

Kao jedna od učestalijih strategija danas koristi se sekvenciranje sačmaricom (engl. *shotgun*). U tom pristupu DNA lanac lomi se na slučajan način u velik broj manjih fragmenata [1]. Dobiveni dijelovi se slažu u sljedove nukleinskih baza. Višestruka očitanja DNA dobivaju se ponavljanjem fragmentacije i sekvenciranja. Na kraju postupka preklapajući krajevi očitanja koriste se za sastavljanje kontinuirane sekvence.

2.2. Poravnanje genskih sljedova

Nakon što su genski sljedovi sekvencirani, potrebno je izvršiti postupak poravnanja. To često predstavlja težak zadatak. Dvije sekvence poravnavaju se tako da postoji mogućnost utvrđivanja sličnosti pojedinih područja. Spomenute sličnosti mogu biti posljedica funkcijske, strukturalne ili evolucijske veze između sekvenci. Poravnati sljedovi mogu se uspoređivati i pritom se pojedinačne razlike u bazama mogu tumačiti kao točke mutacije, a praznine kao mutacije brisanja ili stvaranja.

Za sam postupak koriste se metode dinamičkog programiranja [1]. Problem poravnanja nije karakterističan samo za područje bioinformatike, već se koristi u računanju

cijena udaljenosti uređenja (engl. *edit distance cost*) znakovnih nizova u prirodnom jeziku ili financijskim podatcima.

2.3. Izražajnost gena

Proces u kojem se informacije iz gena koriste za sintezu funkcionalnog produkta gena naziva se izražajnost gena. Najčešće se sinteza proteina odvija preko mRNA [2]. Prvi korak izražajnosti je transkripcija, odnosno prepisivanje DNA u mRNA. Nakon toga izvršava se prevođenje (translacija) određenih nizova nukleotida mRNA u protein.

2.4. Molekularni potpis

Molekularni potpis oblik je profiliranja genskog izražaja koji može ukazati na obilježja biološkog ponašanja (primjerice stanica tumora) [2]. Obrasci molekularnog potpisa grade se na temelju jedinstvenih nakupina gena i proteina koji prikazuju razlikovne razine izražajnosti.

Potpisi mogu pojačati razumijevanje bioloških mehanizama te se mogu iskoristiti u dijagnostičke svrhe. Ideja počiva u postupku izdvajanja uzoraka gena koji su koordinirano izraženi prema nekim značajkama. Neki od kriterija usporedbe mogu biti tip stanice, stadij diferencijacije [1] ili signalni odaziv [2]. Takva interpretacija genskih sljedova omogućuje da se bitne značajke pretoče u numeričke podatke. To će biti vrlo važno pri stvaranju modela nad kojim se vrši analiza. Naglasak se stavlja na mehanizme koji najbolje opisuju pojedinu sekvencu. U idealnom slučaju, svaka tumorska stanica moći će se identificirati jedinstvenim molekularnim potpisom.

2.5. Mehanizmi prijenosa informacije

Geni sadrže upute za izgradnju bjelančevina, molekula koje određuju obilježja organizma. Protok informacija odvija se u nekoliko koraka. Prvo se DNA prepisuje u glasničku RNA. Nakon toga dekodira se RNA kako bi se proizveo aminokiselinski lanac koji će se kasnije složiti u aktivni protein. Postupak nastanka proteina naziva se „centralnom dogmom“ molekularne biologije:

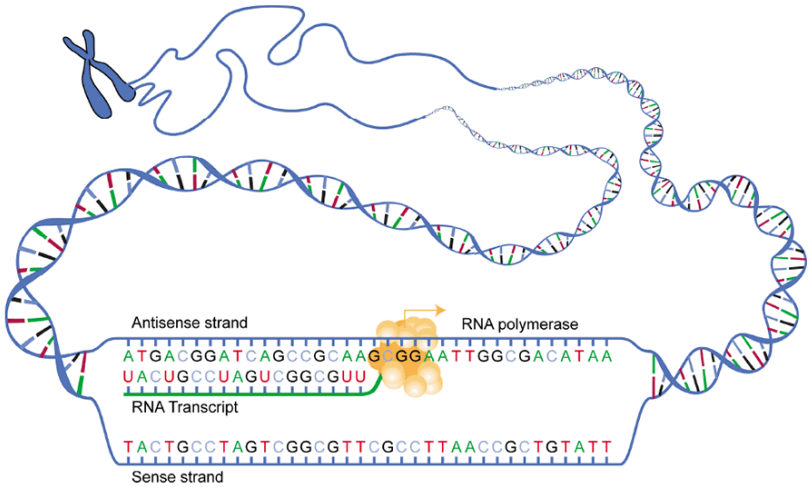
$$DNA \rightarrow RNA \rightarrow proteini. \tag{2.1}$$

[1] promjena stanica iz jednog tipa u drugi

[2] komunikacijski procesi u aktivnosti biološke stanice

2.5.1. Transkripcija

Tijekom prepisivanja ili transkripcije nastaje mRNA. S obzirom na to da su dva lanca DNA komplementarna (lanci imaju međusobno uparene nukleotide $A-T$, $G-C$), nije svejedno koji lanac se prepisuje. Uvijek se transkribira jedan lanac koji se naziva kalup ili nekodirajuća DNA [2]. Prema kalupu se stvara komplementarna mRNA slično kao kod drugog lanca DNA, samo što se u ovom slučaju u lancu umjesto timina nalazi uracil. Novi parovi su sada $A-T$, $G-U$. Postupak prepisivanja prikazan je na slici 2.1.



Slika 2.1: Transkripcija (*oerpub.github.io)*

2.5.2. Translacija

Prevođenje ili translacija je proces u kojem se aminokiseline povezuju u peptidni lanac. Navedeni postupak odvija se na temelju informacije zapisane u mRNA (glasnička RNA). Prevođenje se odvija u trima fazama:

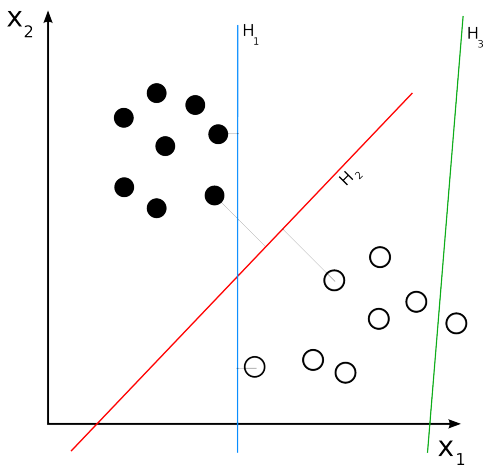
1. početak ili inicijalizacija
2. produživanje ili elongacija
3. završetak ili terminacija

3. Stroj potpornih vektora za klasifikaciju

3.1. Motivacijski problem

U velikom broju životnih situacija imamo potrebu raspoređivanja određenih objekata u pojedine kategorije ili razrede utemeljene na zajedničkim karakteristikama. Takvi primjeri se mogu poistovjetiti s klasifikacijskim problemima.

Neka je zadan skup uređenih parova $(\mathbf{x}_i, y_i)$, $i = 1, 2, ..., n$, gdje je $\mathbf{x}_i \in \mathbb{R}^d$ te predstavlja značajku prikazanu u vektorskom obliku. Uz to vrijedi da su $y_i \in \{-1, +1\}$ i reprezentiraju oznake razreda. Ovim pravilima opisan je binarni klasifikacijski problem, odnosno zadatak razdvajanja značajki u dva različita pripadna razreda. Cilj je postići ispravno raspoređivanje nove značajke x u jedan od dva razreda. Metoda ili algoritam koja omogućuje navedeni cilj naziva se klasifikator. Ako se značajke (uzorci) mogu odvojiti linearnom funkcijom, tada govorimo o konstrukciji linearnog klasifikatora [3].



Slika 3.1: Linearna klasifikacija (*wikipedia.org*)

Na slici 3.1 prikazane su tri mogućnosti linearne klasifikacije. Zeleni pravac H_3 primjer je neuspješnog razdvajanja jer je samo jedan bijeli uzorak dobro klasificiran. Plavi pravac H_1 ispravno razdjeljuje sve uzorke, ali nije optimalno postavljen. Cilj je pronaći pravac koji će biti maksimalno udaljen od oba razreda uzoraka. Takav primjer je crveni pravac H_2 koji predstavlja optimalno rješenje za dani problem. Uzorci iz oba razreda koji su najbliži razdvajajućoj hiperravnini (pravac u dvodimenzionalnom problemu) zovu se **potporni vektori**.

Vladimir Vapnik razvio je klasifikacijski algoritam motiviran problemom linearne binarne klasifikacije. Inicijalna ideja krenula je od želje da se pozitivni i negativni uzorci razdvoje po principu najšire ceste. Algoritam se ustalio pod nazivom stroj potpornih vektora (engl. *Support Vector Machine*, SVM) i spada u model nadziranog učenja.

U dvodimenzionalnom sustavu klasifikator određuje pravac koji razdvaja uzorke. Kod trodimenzionalnih sustava uzorci se razdvajaju ravninom, a u višedimenzionalnim sustavima govori se o hiperravnini[1] kao razdvajajućem mediju. Nakon što je određen potprostor razdvajanja, jednostavno se mogu rasporediti novi uzorci u vlastite razrede.

Ispostavlja se da ovako postavljen problem prelazi u traženje ekstrema funkcije s ograničenjima [3]. Neka je $\mathbf{w}$ vektor okomit na traženu hiperravninu. Tada se širina ceste d_{cesta} koju želimo maksimizirati može izraziti kao:

$$d_{cesta} = \frac{2}{\|\mathbf{w}\|} \ . \tag{3.1}$$

Problem se može pretvoriti u problem minimiziranja recipročnog izraza kojeg još dodatno možemo preoblikovati tako da ne utječemo na traženi maksimum. Manipulacija izraza se vrši radi matematičke praktičnosti što će kasnije olakšati račun. Problem se svodi na rješavanje sjedećeg izraza:

$$min\{\frac{1}{2}\|\mathbf{w}\|^2\} \tag{3.2}$$

s ograničenjima

$$y_i(\mathbf{w}^T x_i + b) \geqslant 1 \ .$$

[1] simetrala spojnice konveksnih ljusaka dviju klasa

Funkciju s ograničenjima[2] možemo minimizirati pronalaženjem pripadnih Lagrangeovih multiplikatora. Nakon raspisivanja Lagrangeove funkcije za problem 3.2 dobije se:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{N} \alpha_i \{y_i(\mathbf{w}^T x_i + b) - 1\} \tag{3.3}$$

gdje je $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_N),\ \alpha_i \geqslant 0$.

Dobivena Lagrangeova funkcija 3.3 može se prikazati u dualnoj formi. Minimizacijom po primarnim varijablama $\mathbf{w}$ i b dolazimo do pripadne dualne funkcije. Imamo sljedeće uvjete:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial \mathbf{w}} = 0 \ \Rightarrow \ \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i \ ,$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial b} = 0 \ \Rightarrow \ \sum_{i=1}^{N} \alpha_i y_i = 0 \ .$$

Nakon uvrštavanja u funkciju L danu izrazom 3.3 dobivamo:

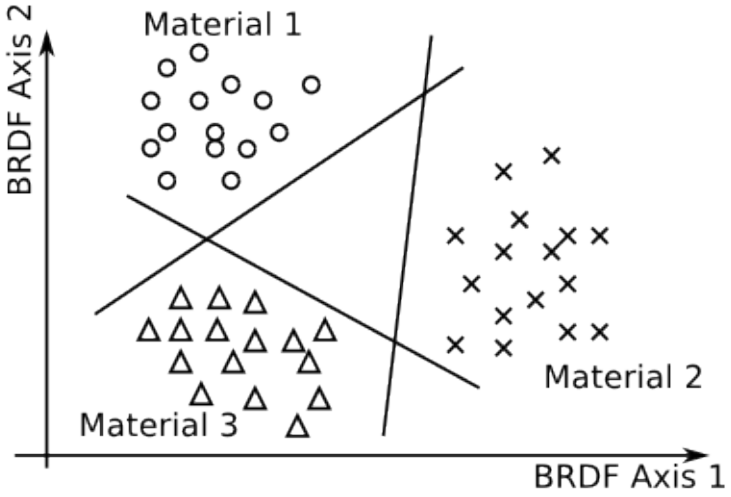
$$\tilde{L}(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \ . \tag{3.4}$$

Funkciju $\tilde{L}$ danu izrazom 3.4 potrebno je maksimizirati. Ovaj oblik predstavlja specifičan optimizacijski problem koji se dodatno svrstava u skupinu linearno ograničenih kvadratnih optimizacijskih problema. Grana programiranja pod nazivom kvadratno programiranje (engl. *quadratic programming*) bavi se rješavanjem upravo takvih problema.

[2] uzorci se ne smiju nalaziti unutar ceste

3.2. Višerazredna klasifikacija

Ozbiljni problemi iz svakodnevnog života najčešće nisu linearno razdvojivi. Uz to se pojavljuje više mogućih razreda u koje se uzorci mogu smjestiti. Binarna klasifikacija se može proširiti na višerazrednu klasifikaciju tako da se omogući da oznake y_i poprimaju vrijednosti u skupu $\{0, ..., N-1\}$, pri čemu je N broj različitih razreda [4]. Na slici 3.2 prikazan je primjer višerazredne klasifikacije.



Slika 3.2: Višerazredna linearna klasifikacija (*www.cs.cmu.edu*)

Za analizu tumorskih stanica svakako će biti potrebno razdvajati tumore u više razreda ovisno o njihovoj vrsti. Višerazrednost ovog problema može se promatrati iz nekoliko perspektiva. Klasifikacijom se primarno cilja na razdvajanje glavnih tipova tumora ovisno o organu ili području tkiva koje je pogođeno bolešću. Nakon što se odredi glavni tip, moguće je klasificirati podtip tumora (npr. u prvoj rundi klasifikacije je identificiran tumor štitnjače, a u drugoj želimo odrediti podvrstu tumora). S obzirom na agresivnost rasta, tumori se dodatno mogu razdvajati na benigne[3], premaligne[4] i maligne[5].

Kako bi se omogućila višerazredna klasifikacija pomoću algoritma stroja potpornih vektora, potrebno je koristiti više zasebnih strojeva. Cjelokupni zadatak se razbija na klasifikacijske podzadatke tako da je za svaki zadužen po jedan stroj potpornih vektora. Konkretne strategije višerazrednog razvrstavanja opisane su u nadolazećim poglavljima.

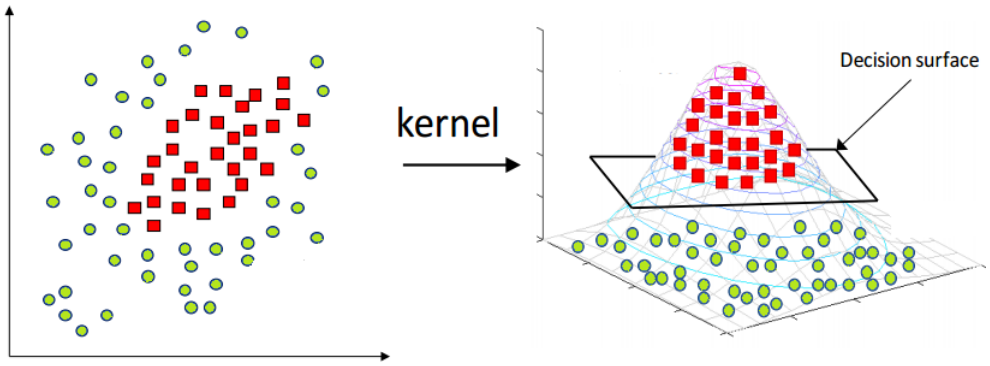
[3] dobroćudni
[4] u postupku prijelaza iz dobroćudne u zloćudnu narav
[5] zloćudni

3.3. Jezgreni trik

Za potrebe rješavanja nelinearnih klasifikacijskih problema uvodi se transformacijska funkcija [5]. Umjesto da se koristi samo skalarni produkt značajki $\mathbf{x}_i^T \mathbf{x}_j$ u izrazu 3.4, odabire se funkcija koja ih dodatno transformira:

$$\kappa : \chi \times \chi \rightarrow \mathbb{R} \ . \tag{3.5}$$

Izraz 3.5 naziva se jezgrena funkcija (engl. *kernel function*), dok se sama metoda naziva jezgreni trik (engl. *kernel trick*). Ako je pronađena prikladna jezgrena funkcija, uzorci se mogu klasificirati u transformiranom prostoru u kojem je moguće pronaći hiperravninu koja će ih razdvojiti na ispravan način. Ispostavilo se da postoji nekoliko jezgrenih funkcija koje za širok spektar problema uspješno transformiraju uzorke. Iz spomenutog razloga među često korištenim funkcijama su radijalna (Gaussova), polinomijalna te sigmoidalna. Na slici 3.3 prikazana je transformacija značajki uz korištenje jezgrenog trika.

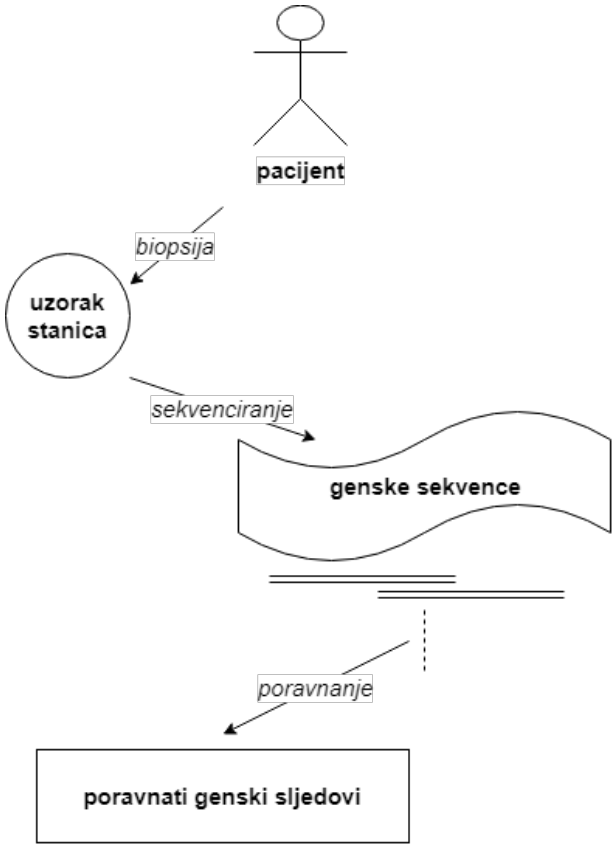


Slika 3.3: Jezgreni trik (*www.hackerearth.com*)

4. Konstrukcija klasifikatora tumorskih stanica

4.1. Ekstrakcija podataka

Nakon što se biopsijom (ili nekom drugom metodom) pribavi uzorak stanica, potrebno ga je pretočiti u genske sljedove postupkom sekvenciranja. Dobiveni sljedovi još nisu upotrebljivi jer su nukleinske baze razbacane unutar njih. Poravnavanjem sekvenci dolazi se do njihovog konačnog oblika koji će se direktno koristiti u samoj analizi.



Slika 4.1: Postupak pripreme podataka

4.2. Inkrementalni razvoj klasifikatora

Temelji korištenog klasifikatora počivaju na metodi stroja potpornih vektora. Na samom početku izgrađen je binarni linearni SVM. Dualni problem pronalaska Lagrangeovih multiplikatora riješen je pomoću sekvencijalne minimalne optimizacije (SMO), algoritma koji je razvijen u kvadratnom programiranju za navedeni problem.

Ako se držimo strogih uvjeta navedenih uz izraz 3.4, riskiramo prenaučenost klasifikatora. Taj problem se pojavljuje kad je skup podataka za učenje takav da tvori iskrivljenu sliku razreda. Posljedica je da se izvrsno klasificira ulazni skup podataka, dok se novi uzorci odvajaju s manje uspjeha. Neželjenom učinku možemo doskočiti uvođenjem meke margine. Dopustit će se ulazak uzorka unutar margine i pogrešna klasifikacija. Ovom žrtvom štitimo se od nepredvidljivosti budućih uzoraka. Uz meku marginu mijenjamo ograničenja:

$$0 \leqslant \alpha_i \leqslant C, \;\; i = 1, ..., N \;\; ,$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0, \;\; i = 1, ..., N$$

pri čemu je C meka margina.

Analogno kao u slučaju bez meke margine, vektori za koje vrijedi $0 \leqslant \alpha_i \leqslant C$ su **potporni vektori** (oni s $\alpha_i = C$ su unutar margine). Prije nego što se opiše algoritam SMO, potrebno je definirati Karush-Kuhn-Tucker (KKT) uvjete za konkretan problem maksimizacije izraza 3.4:

$$\alpha_i = 0 \;\; \Leftrightarrow \;\; y_i u_i \geqslant 1 \;\; , \tag{4.1}$$

$$0 < \alpha_i < C \;\; \Leftrightarrow \;\; y_i u_i = 1 \;\; , \tag{4.2}$$

$$\alpha_i = C \;\; \Leftrightarrow \;\; y_i u_i \leqslant 1 \tag{4.3}$$

gdje je u_i izlaz koji klasifikator daje za i-ti uzorak iz skupa za učenje, a y_i stvarna oznaka razreda. Korisno svojstvo KKT uvjeta je mogućnost pojedinačne evaluacije po uzorcima što će biti značajno u postupku izgradnje algoritma [6]. Cjelokupni postupak razbija se na podzadatke u kojima se zasebno optimiziraju parovi multiplikatora, umjesto da se odjednom pokušaju pronaći svi traženi multiplikatori. Dalje se situacija razrješava analitički spajanjem optimiziranih parova. U nastavku prikazan je pseudokod jedne iteracije algoritma SMO, odnosno recept za rješenje jednog podzadatka.

Algorithm 1 iterationSMO

Prvi parametar: $supportVectors$ – skup trenutnih potpornih vektora.
Drugi parametar: $threshold$ – tolerancija, uvjet zaustavljanja.
$nonKKT := [\varnothing]$
for $(i := 0;\ i < n;\ inc(i))$ **do**
  $\alpha = supportVectors[i].multiplier$
  if $\neg satisfiesKKT(\alpha)$ **then**
    $nonKKT.add(\alpha)$
  end if
end for
repeat
  $\alpha_1, \alpha_2 := choosePair(nonKKT, supportVectors)$
  $\xi := optimize(\alpha_1, \alpha_2)$
until $\xi > threshold$
return α_1, α_2

Funkcija $choosePair$ iz pseudokoda 1 podrazumijeva odabir dva multiplikatora od kojih je prvi sigurno iz skupa $nonKKT$. U navedenom skupu nalaze se multiplikatori koji ne zadovoljavaju KKT uvjete. Drugi multiplikator može biti proizvoljan. Obično se u tom slučaju primjenjuju heuristike zbog velikog broja potpornih vektora koji predstavljaju izvor multiplikatora. Za potrebe ovog problema, izabrana je heuristika koja uzima u obzir razliku između grešaka i-tog uzorka u procesu učenja.

Ako je ξ_1 greška multiplikatora α_1, a ξ_2 greška multiplikatora α_2, tada se nastoje odabrati multiplikatori za koje vrijedi da je izraz [6]:

$$|\xi_1 - \xi_2| \tag{4.4}$$

minimalan.

Greške multiplikatora pohranjuju se u internim priručnim memorijama objekata koji modeliraju potporne vektore. Osim što se tako nastoji omogućiti brzo dohvaćanje vrijednosti grešaka, eliminirana je potreba za ponovnim računanjem greške.

Jednom kad je napisan, algoritam SMO inkorporira se u logiku SVM klasifikatora. Prva razina je dovršena pa se klasifikator može isprobati na nekim problemima binarne linearne klasifikacije. U svrhu ispitivanja valjanosti i kvalitete rješenja, generirani su fiktivni problemi razdvajanja pozitivnih i negativnih uzoraka u višedimenzionalnom prostoru. Nakon što se utvrdi mogućnost pronalaženja hiperravnine koja će ispravno razdvojiti uzorke u spomenutim primjerima, prelazi se na sljedeći korak konstrukcije klasifikatora – nadogradnja za višerazredne probleme.

Kao što je opisano u poglavlju 3.2, pri klasifikaciji uzoraka u više razreda, potrebno je uvesti i više strojeva potpornih vektora.

4.2.1. Strategije višerazredne klasifikacije

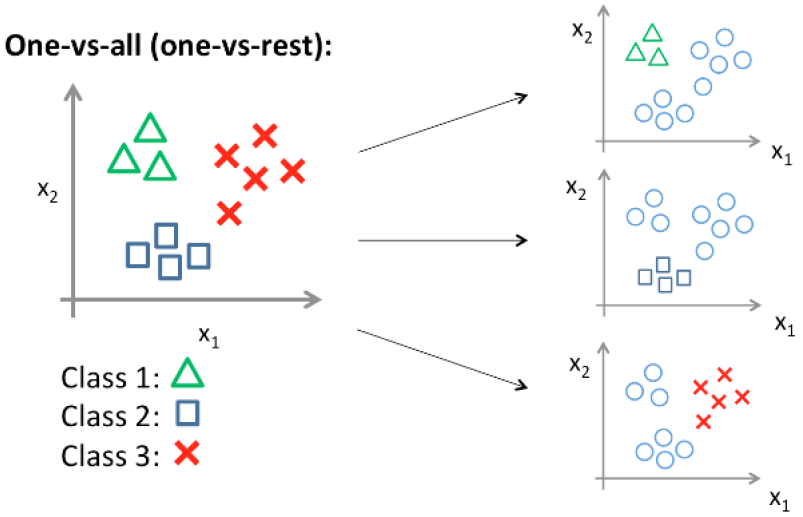
Dvije često korištene strategije višerazredne klasifikacije su:

1. strategija jedan protiv svih (engl. *One-vs-All*)
2. strategija jedan na jedan (engl. *One-vs-One*)

Strategija jedan protiv svih (OVA)

Problem s više razreda potrebno je svesti na osnovni slučaj kada postoje samo dva razreda jer SVM može riješiti takav zadatak. U strategiji jedan protiv svih, pojedinačno i slijedno uzimaju se razredi tako da isti zauzima jedno mjesto u binarnom izboru. Od ostalih neizabranih razreda stvara se nadrazred koji obuhvaća sve iz navedenog ostatka. Sada kao drugi izbor stoji simulirani nadrazred. Daljnji postupak nalaže da se upogoni SVM koji može naučiti klasificirati odabrani razred naspram svih ostalih.

Ponavljanjem metoda za svaki pojedini razred, kao posljedica će nastati N binarnih SVM-ova kod kojih je svaki odgovaran za prepoznavanje jednog od N razreda. Navedeno je ilustrirano na primjeru na slici 4.2



Slika 4.2: Strategija jedan protiv svih (*houxianxu.github.io*)

Strategija jedan na jedan (OVO)

U drukčijoj postavci u odnosu na strategiju jedan protiv svih, stvara se SVM za svaki mogući par razreda. Ukupno se može načiniti $\binom{N}{2}$, odnosno $\frac{N(N-1)}{2}$ parova. Strategija jedan protiv svih pridonosi veću značajnost lokaliziranoj klasifikaciji. Svaki SVM uspoređuje samo dvije klase od postojećih N, pri čemu se ostale u samom postupku zanemaruju.

Svaka od strategija višerazredne klasifikacije može biti bolja u nekom konkretnom problemu. Strategija jedan na jedan ima veću vremensku složenost jer podrazumijeva konstrukciju većeg broja SVM-ova. U daljnjem tekstu će se metoda koja enkapsulira logiku određivanja pripadnosti uzoraka za više razreda nazivati MSVM (skraćenica za multi-SVM).

Još je ostao problem nelinearnosti podataka koji će se nastojati riješiti pomoću jezgrenog trika. Postavlja se pitanje odabira jezgrene funkcije. Ako se posegne za uporabom ustaljenih jezgrenih funkcija, kolika će biti njihova uspješnost? Transformacija značajno utječe na ishod klasifikacije, stoga potrebno je posvetiti posebnu pažnju odabiru funkcije.

4.2.2. Izgradnja jezgrene funkcije

Umjesto odabira fiksne jezgrene funkcije, provest će se konstrukcija nove temeljene na samim podatcima. Ako su K_1 i K_2 jezgrene funkcije, tada vrijedi da su:

$$K^+ = K_1 + K_2 \ , \tag{4.5}$$

$$K^* = K_1 \cdot K_2 \tag{4.6}$$

također valjane jezgrene funkcije.

Posljedično se pravila 4.5 i 4.6 mogu kombinirati i primijeniti na bilo kojem broju baznih funkcija. Uzimajući spomenuto u obzir, moguće je postići transformaciju koja u sebi sadrži raznovrsna svojstva. To ne bi bilo ostvarivo uz korištenje pojedinačnih fiksnih transformacija. U ovom slučaju upotrijebljena je baza koja se sastoji od pet fiksnih jezgrenih funkcija [7].

Tablica 4.1: Bazne jezgrene funkcije

Naziv	Jezgrena funkcija	Broj parametara
Radijalna (K_{rad})	$K_{rad}(\mathbf{x}_i, \mathbf{x}_j) = exp(-\alpha\|\mathbf{x}_i - \mathbf{x}_j\|^2)$	1
Polinomijalna (K_{poly})	$K_{poly}(\mathbf{x}_i, \mathbf{x}_j) = [\alpha(\mathbf{x}_i \cdot \mathbf{x}_j) + \beta]^\gamma$	3
Sigmoidalna (K_{sig})	$K_{sig}(\mathbf{x}_i, \mathbf{x}_j) = tanh[\alpha(\mathbf{x}_i \cdot \mathbf{x}_j) + \beta]$	2
Inverzna multikvadratna (K_{imq})	$K_{imq}(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{\sqrt{\|\mathbf{x}_i - \mathbf{x}_j\|^2 + \beta^2}}$	1
Sferna (K_{sph})	$K_{sph}(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{3}{2}\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\alpha} + \frac{1}{2}(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\beta})^3$	2

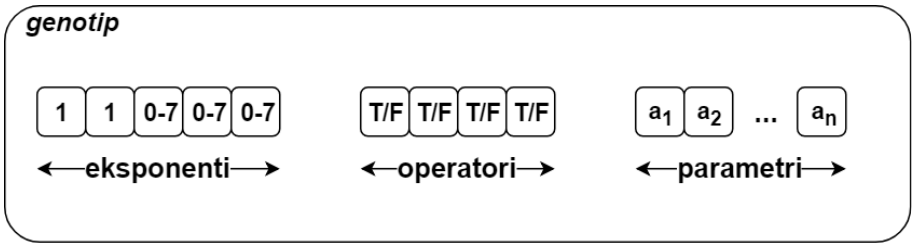
Kako bi se iskoristile sve mogućnosti kombiniranja baznih jezgrenih funkcija, nužno je kombiniranu transformaciju računati koristeći četiri spojne operacije. Neka operacija bude predstavljena simbolom $\otimes$. Kombinirana jezgrena funkcija može se opisati izrazom [7]:

$$\tilde{K} = K_{rad}^{\kappa_1} \otimes_1 K_{poly}^{\kappa_2} \otimes_2 K_{sig}^{\kappa_3} \otimes_3 K_{sph}^{\kappa_4} \otimes_4 K_{imq}^{\kappa_5} \tag{4.7}$$

pri čemu je $\otimes_i \in \{+; \cdot\} \ \forall \, i = 1, 2, 3, 4$. Eksponenti $(\kappa_1, ..., \kappa_5)$ su cijeli brojevi kojima se potencira vrijednost zasebnih komponenti (baznih jezgrenih funkcija).

Uporaba genetskog algoritma (GA) za pronalazak jezgrene funkcije

U želji da se tražene jezgrene funkcije što više determiniraju samim podatcima, koristi se metaheuristika čiji je cilj pronaći što bolje slobodne parametre u izrazu 4.7. Na taj način optimiziraju se elementi samog klasifikatora, odnosno proces se privremeno spušta na nižu razinu. Konkretno cilj je pronaći transformaciju uz koju će klasifikator imati najveću uspješnost u vidu postotka točnosti predviđanja razreda zadanog uzorka. Za navedeni zadatak koristi se genetski algoritam (GA) inspiriran biološkom evolucijom. Kako bi se GA upogonio, potrebno je osmisliti genotip koji će vjerno oslikavati izraz 4.7.



Slika 4.3: Genotip jedinke

Na slici 4.3 može se uočiti kako je struktura genotipa morfološki razdvojena na tri dijela [7]. Slijedi opis po komponentama.

Eksponenti - prva sastavnica

U prvoj komponenti sadržano je pet cijelih brojeva. S obzirom na to da radijalna i polinomijalna jezgrena funkcija već sadrže parametre koju omogućuju potenciranje, njihovi eksponenti su postavljeni na jedan. Ispostavlja se da je mijenjanje navedenih eksponenata matematički redundantno pa se izostavljaju u konkretnom izračunu. Na slici 4.3 prva dva eksponenta prikazana su radi potpunosti opisa genotipa. Ostali eksponenti su cijeli brojevi u rasponu $0 - 7$. Ograničenje je postavljeno kako ne bi došlo do prevelikih vrijednosti ukupne jezgrene funkcije, što bi posljedično dovelo do aritmetičkog preljeva. Gornja ograda od 7 doima se dovoljno malenom da se takvo što ne bi dogodilo, a opet može pružiti fleksibilnost u izračunu koja je poželjna za postizanje raznolikosti jedinki.

Operatori - druga sastavnica

Drugi dio predstavlja operatore. Kako bi se postigao elegantniji računalni zapis, učinjeno je jednostavno preslikavanje:

$$+ \quad \rightarrow \quad True(T) \ ,$$
$$\cdot \quad \rightarrow \quad False(F) \ .$$

Sada se operatorski dio može prikazati poljem istinitosnih vrijednosti. U modelu kombinirane jezgrene funkcije, ukupno četiri elementa otpadaju na prikaz operatora.

Parametri - treća sastavnica

Treći odjeljak genotipa reprezentira slobodne parametre. Uzimajući u obzir broj slobodnih parametara u baznim jezgrenim funkcijama iz tablice 4.1, dostiže se brojka od 9 parametara. Pri tome je svaki parametar predstavljen realnim brojem što posljedično omogućava prikaz trećeg dijela genotipa pomoću polja realnih brojeva.

Promatrajući cijeli genotip, model jedinke će se sastojati od $(5-2)+4+9=16$ elemenata. Od toga su tri cijeli brojevi koji predstavljaju eksponente, četiri istinitosne vrijednosti koje se interpretiraju kao operacije te 9 realnih brojeva koji modeliraju slobodne parametre.

Operatori i parametri genetskog algoritma

Nakon što je izgrađen model genotipa kombinirane jezgrene funkcije, potrebno je odabrati operatore korištene u genetskom algoritmu [8].

Jedinke se pri stvaranju početne populacije grade tako da se sastavnice genotipa popunjavaju nasumično. Konkretno će prvi dio biti popunjen nasumičnim cijelim brojevima u rasponu $0-7$, a drugi dio će poprimiti nasumične istinitosne vrijednosti. Vrijednosti za parametre u trećoj komponenti genotipa generiraju se po normalnoj razdiobi. Mjera kvalitete jedinke koja predstavlja kombiniranu jezgrenu funkciju izraču-

nava se na temelju točnosti klasifikacije na skupu za provjeru uporabom te konkretne jezgrene funkcije. To za posljedicu ima pokretanje konstrukta MSVM svaki put kad je potrebna evaluacija jedinke. Klasifikator će prolaskom po ispitnim uzorcima vratiti točnost na ispitnom skupu kao ocjenu.

Za odabir jedinki koje se koriste u postupku križanja, koristi se k-turnirska selekcija. Navedena metoda podrazumijeva nasumičan odabir k jedinki iz cjelokupne populacije. U svakom iteraciji postupak se provede dva puta kako bi se izabrane jedinke mogle proslijediti operatoru križanja. Zbog složene strukture genotipa, prirodno je razdvojiti postupak križanja i mutacije po komponentama. Za svaku sastavnicu odabran je prikladni operator.

Kod komponente s eksponentima upotrebljava se jednostavno križanje koje stvara novu jedinku tako što svaki eksponent preuzima nasumično od jednog ili drugog roditelja. U postupku mutacije za svaki eksponent se s određenom vjerojatnošću generira nova vrijednost u rasponu $0 - 7$. Tako je očuvana restrikcija raspona eksponenata.

Križanje za drugu sastavnicu, odnosno za operatore, vrši se slično kao kod eksponenata. Jedina je razlika što se u ovom slučaju preuzimaju istinitosne vrijednosti od roditelja. Operator mutacije negira pojedinu komponentu, ako je pripadni operator izabran u postupku. Navedeno je posljedica vjerojatnosnih parametara mutacije.

Za potrebe križanja slobodnih parametara, upotrebljen je BLX-α operator križanja. Mutacija se odvija tako da se konkretnom parametru pribraja vrijednost izvučena iz normalne distribucije. Tako se postiže smanjenje i povećanje parametara s obzirom na to da generirane vrijednosti mogu biti negativne i pozitivne.

U nastavku je prikazan pseudokod opisanog genetskog algoritma. Kao krajnji rezultat nastoji se pronaći jezgrena funkcija koja će dati najbolju točnost klasifikacije. Algoritam je elitistički jer se uvijek čuva najbolja jedinka.

Algorithm 2 kernelGA

Prvi parametar: $data$ – ulazni podatci za učenje.

Drugi parametar: $labels$ – pripadni razredi za ulazne podatke.

$pop := createPopulation()$

for ($i := 0$; $i < maxIter$; $inc(i)$) **do**

  for ($kernel$: pop) **do**

    $MSVM = initMSVM(kernel)$

    $MSVM.learn(data, labels)$

    $kernel.setFitness(MSVM.accuracy())$

  end for

  $newPop := [\varnothing]$

  $best = pop.getBest()$

  $newPop.add(best)$

  for ($j := 1$; $j < pop.size()$; $inc(j)$) **do**

    $mother = Ktournament(pop)$

    $father = Ktournament(pop \setminus \{mother\})$

    $child = crossover(mother, father)$

    $child.mutate()$

    $newPop.add(child)$

  end for

  $pop = newPop$

end for

return $pop.getBest()$

4.2.3. Konačna inačica klasifikatora

MSVM je uz posljednju nadogradnju spreman za teže klasifikacijske probleme nelinearne naravi. Genetski algoritam omogućava korištenje značajki samih podataka jer na temelju njih konstruiraju se jezgrene funkcije.

Trenutni klasifikator isproban je na nekoliko ispitnih skupova podataka. Analiza i rezultati klasifikacije opisani su detaljnije u poglavlju 5. Nakon potvrde ispravnog funkcioniranja nadograđenog MSVM-a, može se krenuti u izgradnju konkretnog modela za klasifikaciju tumorskih stanica. U sažetku dosadašnjeg postupka konstrukcije klasifikatora mogu se izdvojiti sljedeći koraci:

1. priprema i prilagođavanje podataka
2. izgradnja linearnog binarnog SVM-a
3. ispitivanje funkcionalnosti na jednostavnijim primjerima
4. nadogradnja za višerazrednu klasifikaciju
5. ispitivanje na višerazrednim linearnim problemima
6. uvođenje jezgrenog trika za nelinearnu klasifikaciju
7. implementacija metaheuristike – GA za optimiziranje kombinirane jezgrene funkcije
8. spajanje višerazrednog SVM-a s optimizacijskom metodom
9. ispitivanje kvalitete klasifikacije na složenijim problemima

Kao rezultat je nastao MSVM koji predstavlja generički klasifikator. Za rješavanje konkretnih problema, potrebno je samo pripremiti model podataka koji će se koristiti u klasifikaciji.

4.2.4. Model uzoraka tumorskih stanica

Sekvencirani i poravnani genski sljedovi tumorskih stanica još nisu prikladni za postupak razvrstavanja. Potrebno je prikaz pretočiti u numeričke podatke s kojima će proces

strojnog učenja uz pomoć načinjenog SVM-a moći baratati. U tu svrhu upotrijebit će se metoda RNA sekvenciranja (RNA-seq).

Glavna ideja je numerička interpretacija sličnosti izdvojenih genskih sekvenci s nekim dobro utvrđenim konstruktom. U ovom slučaju radi se o sekvenciranom ljudskom genomu.

Pojedini genski sljedovi u genomi imaju specifične uloge koje su većinom poznate. Najbitnije promatrane značajke na temelju kojih se provodi usporedba su:

1. spojeni genski transkripti
2. post-transkripcijske modifikacije
3. fuzija gena
4. mutacija gena
5. jednonukleotidni polimorfizam

Pomnim uspoređivanjem svakog gena iz ljudskog genoma, prepoznaju se navedena zajednička obilježja s ispitnim genskim slijedom potencijalne tumorske stanice ako postoje. Tim postupkom moguće je konstruirati polje cijelih brojeva čije će vrijednosti biti redom ocjene genske sličnosti s pojedinim poredbenim sekvencama.

U genskim značajkama krije se izražajnost gena koja utječe na molekularni potpis. Veličina i raznolikost ljudskog genoma koji sadrži oko $30\,000$ gena omogućuje vrlo detaljnu provedbu uspoređivanja. RNA sekvenciranjem pokušava se numerički iskazati molekularni potpis tumora koji će poslužiti kao temeljni klasifikacijski kriterij.

Kao konačni model nastaje polje cijelih brojeva čija veličina kruži oko $22\,000$ ovisno o detaljima usporedbe. Takva reprezentacija direktno se koristi za ulazne podatke u klasifikaciji. Kako je veličina modela poprilično velika, u nekim situacijama bit će potrebno izvršiti sažimanje kako bi se osiguralo prihvatljivo vrijeme trajanja provedbe postupka. Konkretni primjeri opisani su u poglavlju 5.

4.2.5. Postupak učenja i predviđanje razreda novih uzoraka

Za određeni klasifikacijski problem, potrebno je pribaviti uzorak za učenje. Na temelju tih podataka MSVM gradi model, odnosno pronalazi odgovarajuću hiperravninu [9]. Podrazumijeva se da za svaki uzorak za učenje znamo pripadni razred. Takva vrsta

učenja naziva se nadzirano učenje. Nakon provedbe tog postupka, moguće je koristiti MSVM za klasifikaciju. Potrebno je predati uzorak čiji se razred želi predvidjeti. Ispituje se pripadnost na temelju izgrađenog modela. Predviđanje rezultira konkretnim razredom.

4.2.6. Izgradnja vjerojatnosnog modela klasifikacije

Klasifikator opisan u prethodnom poglavlju spreman je za primjenu. Zamislimo situaciju u kojoj želimo postaviti dijagnozu odabranom pacijentu. Izdvojen je uzorak stanica štitnjače metodom biopsije kako bi se mogla pomnije analizirati njihova struktura. Sumnja se da pacijent ima tumor na području štitnjače i želi se utvrditi konkretan tip ako je to uistinu slučaj. To će omogućiti da pacijent primi adekvatnu terapiju. Pretpostavimo da postoji pet različitih tipova tumora koji se mogu identificirati. Uz dodatnu mogućnost da je pacijent zdrav, ukupno razlikujemo šest razreda klasifikacijskog problema. Nakon pripreme podataka i pokretanja klasifikatorskog programa, dobili smo rezultat. Klasifikator je odabrao određeni tip kao razred uzorka. U ovom trenutku se možemo zapitati koliko je pouzdano rješenje koje smo dobili. Program će uvijek ponuditi neko rješenje, ali nisu sva predviđanja jednako pouzdana. U nekim slučajevima uzorak će prema klasifikacijskom modelu imati značajke više razreda. Jednom kad je razred odabran, gube se informacije o mogućoj povezanosti s ostalim klasama. Sljedeći cilj je dobiti širu sliku klasifikacije te na neki način odrediti udjele pripadnosti uzorka pojedinim razredima.

Spomenuta ideja će se provesti izgradnjom distribucije vjerojatnosti nad razredima. Za taj zadatak koristi se metoda pod nazivom **Plattovo** [1] **skaliranje** ili **Plattova kalibracija** [10]. Umjesto predviđanja konkretnog razreda uzorka, aproksimiraju se aposteriorne vjerojatnosti pripadnosti pojedinoj klasi. Uporabom sigmoidalne funkcije na temelju uzoraka za učenje SVM-a moguće je izvršiti procjenu distribucije.

Posteriorna vjerojatnost pripadnosti uzorka određenom razredu $P(y = 1|x)$ može se aproksimirati na sljedeći način [9]:

$$P(y = 1|\mathbf{x}) = P_{A,B}(f) = \frac{1}{1 + exp(Af + B)} \tag{4.8}$$

pri čemu je $f = f(\mathbf{x})$. Neka je f_i procjena za $f(\mathbf{x}_i)$. Najbolja postavka parametara

[1] John Platt - američki znanstvenik na području računarstva

$\mathbf{z} = (A, B)$ određena je rješavanjem pripadnog regulariziranog problema maksimalne izglednosti (s S_+ kao pozitivnim uzorkom i S_- kao negativnim uzorkom):

$$\min_{\mathbf{z}=(A,B)} F(\mathbf{z}) = -\sum_{i=1}^{N} (t_i ln(p_i) + (1 - t_i) ln(1 - p_i)) \tag{4.9}$$

$$\text{za } p_i = P_{A,B}(f_i), \quad t_i = \begin{cases} \frac{S_+ + 1}{S_- + 2} & \text{ako } y_i = +1 \\ \frac{1}{S_- + 2} & \text{ako } y_i = -1 \end{cases}, i = 1, ..., N.$$

Funkcija maksimalne izglednosti može se pronaći Newtonom metodom numeričke optimizacije. U višerazrednoj klasifikaciji, potrebno je procijeniti vjerojatnosti za svaki pojedini razred. Kao rezultat sada imamo distribuciju vjerojatnosti po razredima za promatrani uzorak. Vratimo se na primjer utvrđivanja točnog tipa tumora štitnjače. Sada za svaki tip možemo dobiti i vjerojatnosnu ocjenu. To omogućava sigurniju odluku i opisuje kontekst situacije. Više nije svejedno hoće li klasifikator predvidjeti razred sa sigurnošću od 60% ili primjerice 95%. Analiza se može vršiti i među ostalim razredima koji nisu odabrani. Ako je ostatak vjerojatnosnog iznosa (ono što je preostalo oduzme li se od jedinice ocjena vjerojatnosti odabranog razreda) jednoliko raspoređen među ostalim klasama, tada je naše rješenje pouzdanije. U drugom slučaju se može dogoditi da je jedan razred vrlo blizu izboru klasifikatora. Tada se izbor može suziti na ta dva tipa. Brojne su druge varijacije u kojima se može puno ozbiljnije promišljati o potencijalnim odnosima i konačnom odabiru.

4.2.7. Paralelizacija višerazredne klasifikacije

Pri klasifikaciji velike količine podataka, javlja se problem vremenske složenosti izračuna. Numerički prikazi uzoraka s kojima klasifikator barata sadrže preko $20\,000$ atributa što dodatno usložnjava situaciju. Zbog spomenutih razloga, implementirana je paralelizacija posla pri učenju MSVM-a za višeklasne probleme.

Zadatci su rastavljeni na zasebne komponente tako da se sami poslovi mogu kreirati i odrađivati neovisno. Razlikuje se nekoliko vrsta poslova:

1. obrada
2. učenje

3. Plattovo skaliranje

Svi podzadatci slažu se u red iz kojeg pojedine dretve uzimaju poslove i odrađuju ih. Sve skupa je kontrolirano od strane objekta koji na početku stvara onoliko dretvi koliko je slobodnih procesora (ili procesorskih jezgara) te ih drži u fiksnom bazenu. Objekt upravljač dalje raspoređuje dretve koje stavljaju obavljene poslove u poseban red. Rezultati se spajaju i završava se učenje klasifikatora.

Bilo je potrebno razlučiti slučajeve ovisno o konkretnoj strategiji višerazredne klasifikacije. Kada se koristi strategija jedan na jedan, nužno je stvoriti poslove za svaki od $\binom{N}{2}$ SVM-ova. Analogno se stvara N strojeva potpornih vektora i pripadni poslovi kod strategije jedan protiv svih.

5. Analiza rezultata

5.1. Primjena klasifikatora za razdvajanje raznovrsnih podataka

MSVM opisan u poglavlju 4.2 konstruiran je tako da može klasificirati uzorke neovisno o njihovom tipu. Za svaku pojedinu vrstu uzorka napisan je prilagodni model koji povezuje klasifikator i sirove podatke. U daljnjoj analizi točnost učenja predstavlja postotak ispravno klasificiranih uzoraka na skupu za učenje, dok točnost ispitivanja podrazumijeva uspješnost klasifikacije uzoraka iz skupa za provjeru.

U želji da se isproba kvaliteta razdvajanja višedimenzionalnih uzoraka, koristio se skup podataka [A] koji prikazuje pacijente koji boluju od hepatitisa. Uz svakog pacijenta je pripremljen niz medicinskih značajki. Cilj je na temelju pacijentovih obilježja procijeniti stanje i izglednost ozdravljenja. Skup za učenje se sastoji od 155 opisa. Svaki uzorak sadrži 20 atributa od kojih su neki s diskretnim vrijednostima (npr. prisutnost umora kod pacijenta - da ili ne), dok su drugi kontinuirane vrijednosti u nekom specifičnom intervalu (npr. koncentracija bilirubina u krvi). Uspješnost klasifikacije na nova 122 podatka iz skupa za provjeru bila je 90.98%.

Testovi su se proveli i nad skupom podataka [B] koji sadrži opis otrovnih i neotrovnih gljiva. Baza posjeduje ukupno 8124 uzorka. Za opis svake gljive upotrebljena su 22 atributa s višestrukim kategorijskim vrijednostima. Točnost razvrstavanja nova 1243 uzorka iznosila je 96.62%.

Višerazredna klasifikacija isprobala se nad skupom podataka [C] koji opisuje različite vrste stakala. Skup za učenje sadrži 214 uzoraka, a ukupno je 7 različitih razreda kojima pojedino staklo može pripadati. Svaki podatak ima 9 atributa (kombinirano numeričkih i kategorijskih) koji ga opisuju. Klasifikacija je provedena s uspješnošću od 95.45% nad novih 198 uzoraka.

Sažetak rezultata za tri spomenuta skupa podataka [A], [B], [C] prikazan je tablično u nastavku.

Tablica 5.1: Rezultati klasifikacije

Skup podataka	Broj ulaznih uzoraka	Broj atributa	Broj novih uzoraka	Točnost
Domena hepatitisa	155	20	122	90.98%
Gljive	8124	22	1243	96.62%
Stakla	214	9	198	95.45%

5.2. Utjecaj jezgrene funkcije na kvalitetu klasifikacije

Transformacija podataka jedan je od ključnih faktora u višerazrednoj klasifikaciji. Posebna pažnja posvećena je jezgrenim funkcijama koje se grade genetskim algoritmom. Usporedba se vršila nad skupom koji opisuje vrste sunčevih zraka te podatcima o genskim sljedovima primata.

Baza podataka [D] o sunčevim zrakama sadrži 1389 uzoraka pri čemu svaki podatak ima 13 atributa. Četiri su moguće klase pripadnosti ovisno o tipu zrake. Testirana je uspješnost klasifikatora s različitim transformacija. Kada se koristi fiksna radijalna jezgrena funkcija, točnost ispitivanja na 812 uzoraka iznosi 84%. Uporabom genetskom algoritma i kombinirane jezgrene funkcije, točnost na skupu za provjeru popela se na 94.08%.

Skup podataka [E] koji sadrži opise genskih sljedova primata sastoji se od 3190 uzoraka. Svaki slijed u svom opisu sadrži 8 numeričkih atributa. Ukupno postoje tri različita razreda pripadnosti ovisno o tipu spojnih čvorišta gena:

- egzon/intron (EI)
- intron/egzon (IE)
- nijedan od prva dva razreda

Testiranje se izvršilo nad novih 1.560 uzoraka. Uspješnost bez kombinirane jezgrene funkcije iznosila je 82.37% na skupu za provjeru, dok se korištenjem genetskog algoritma za optimizaciju popela na 93.07%.

U sljedećoj tablici prikazana je kratka usporedba rezultata klasifikatora bez i s korištenjem kombinirane jezgrene funkcije (JF).

Tablica 5.2: Fiksna JF vs. kombinirana JF - točnost

Skup podataka	Fiksna JF - točnost	Kombinirana JF - točnost	Poboljšanje
Sunčeve zrake	84%	94.08%	+10.08%
Genski sljedovi primata	82.37%	93.07%	+10.7%

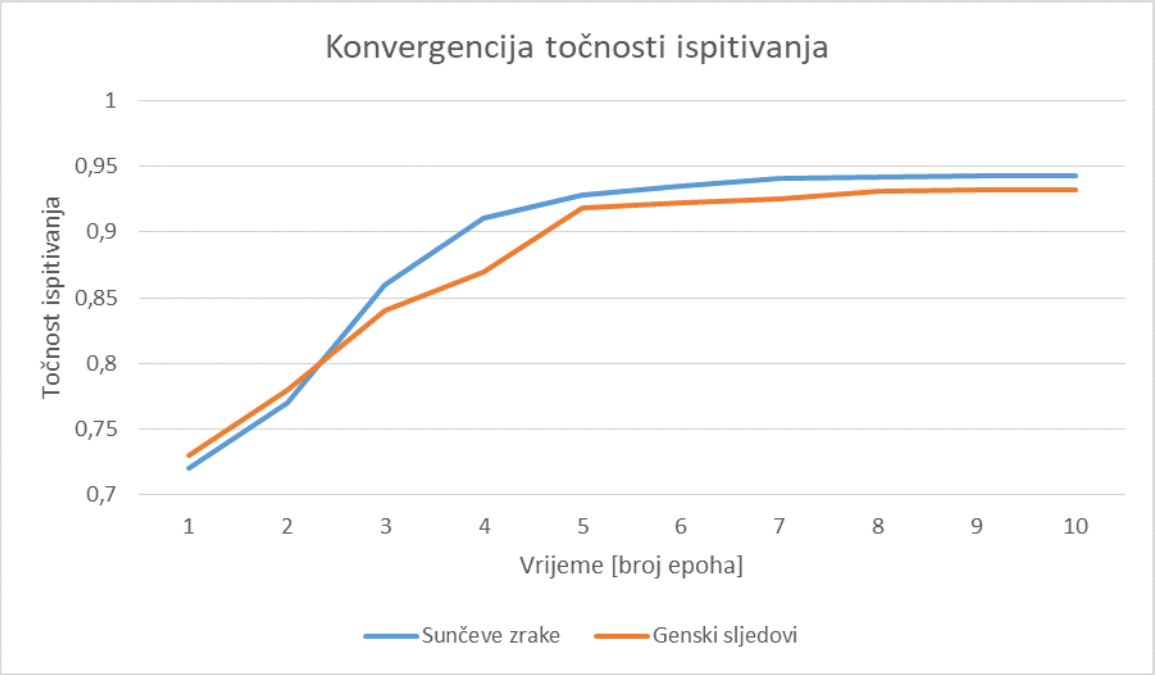
Korištenjem GA postignuti su bolji rezultati točnosti ispitivanja kao što se može vidjeti u tablici 5.2. Cijena kvalitetnijih rezultata je veća vremenska složenost. U tablici u nastavku prikazan je konkretan odnos vremena učenja klasifikatora bez i s postupkom optimizacije jezgrene funkcije uz pomoć genetskog algoritma. Usporedba se vršila nad istim skupovima podataka kao i kod analize točnosti. Učenje je pokretano 20 puta kako bi se dobila što bolja vremenska ocjena računanjem prosjeka. Kao uvjet zaustavljanja koristio se određen broj prolaza kroz skup za učenje. U terminologiji učenja klasifikatora, jedno predstavljanje cjelokupnog skupa za učenje naziva se **epoha**. U ovom slučaju broj epoha je ograničen na pet što je obrazloženo u sljedećim poglavljima.

Tablica 5.3: Fiksna JF vs. kombinirana JF - trajanje učenja

Skup podataka	Fiksna JF - trajanje [min]	Kombinirana JF - trajanje [min]
Sunčeve zrake	0.2	11.4
Genski sljedovi primata	0.4	28.1

5.3. Značajke klasifikatora

Sljedeća bitna značajka je doprinos točnosti ovisno o tome koliko epoha uči MSVM. Na slici sljedećoj prikazan je odnos točnosti dvaju skupova podataka iz prošlog odjeljka - skup sunčevih zraka [D] i genski sljedovi primata [E].

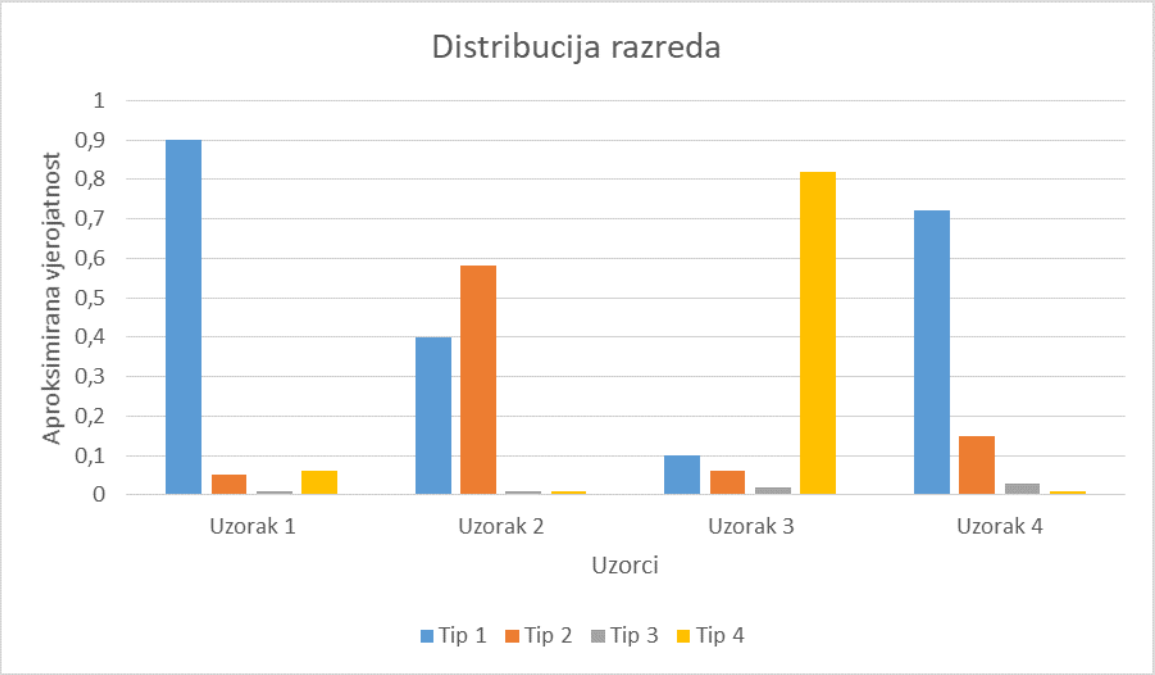


Slika 5.1: Konvergencija točnosti ispitivanja na skupovima [D] i [E]

Na slici 5.1 može se uočiti značajan porast točnosti u prvih nekoliko epoha kod oba skupa podataka. Rast se polako stabilizira nakon četvrte epohe, a nakon pete primjećuje se fiksiranje iznosa točnosti klasifikatora. U nastavku od šeste pa do desete epohe točnost konvergira k maksimalnoj vrijednosti. Na temelju ovih empirijskih rezultata, razumno je učiti MSVM pet epoha jer će se tako ostvariti gotovo maksimalna točnost. S druge strane, pet epoha prihvatljiva je brojka iz vremenskog aspekta. Prosječno vrijeme učenja klasifikatora uz broj atributa koji pruža dovoljnu izražajnost (primjerice 100) iznosi otprilike 20 minuta.

Plattovo skaliranje vrlo je moćan alat pri donošenju klasifikacijskih odluka. Za skup podataka sunčevih zraka izvučena su četiri nove uzorka koja su potom klasificirana. Iscrtane su vrijednosti procijenjene vjerojatnosti za svaki od moguća četiri razreda.

Na slici 5.2 se mogu vizualno interpretirati pouzdanosti klasifikacijskog odabira. Kod prvog uzorka, izbor je vrlo jasan. S velikom sigurnošću se može odabrati tip 1, odnosno prvi razred. Situacija nije kristalno čista u slučaju drugog uzorka. Tu se pojavljuje tip 1 kao potencijalna prijetnja sigurnosti odabira tipa 2. U ovoj konkretnoj situaciji izbor se može suziti na dva razreda. Tip 2 je s druge strane relativno siguran

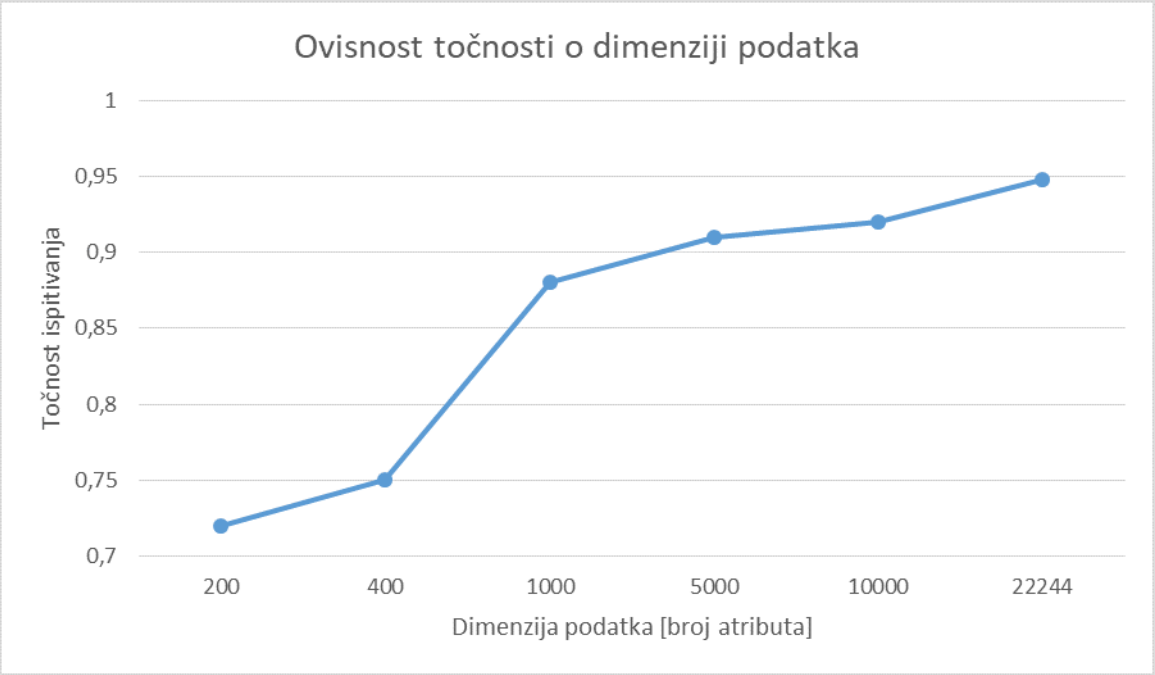


Slika 5.2: Distribucija razreda

odabir jer prednjači s vjerojatnošću 0.58 nad tipom 1 čija je aproksimirana vjerojatnost 0.4. Kod trećeg i četvrtog uzorka odabir je jasan - redom tip 4 i tip 1.

5.4. Uspješnost klasifikacije tumorskih stanica

Konačni model podataka za genske sljedove tumorskih stanica sadrži $22\,244$ atributa. Značajna veličina utječe na vremensku zahtjevnost učenja klasifikatora. Moguće je primijeniti neku od metoda smanjivanja dimenzionalnosti uzoraka kako bi se skratilo vrijeme izvođenja programa [11]. Navedeno podrazumijeva postupak poznat pod nazivom odabir značajki (engl. *feature selection*). U ovom konkretnom problemu varijable su se eliminirale na temelju kriterija korelacije. To znači da se snažno korelirane varijable (pojedini atributi u uzorku) smatraju redundantnima te se uklanjaju. Korištenjem koreliranih atributa ne dobiva se novo znanje o uzorku u postupku učenja. Na slici 5.3 prikazana je točnost ispitivanja u ovisnosti o broju korištenih atributa, odnosno o dimenziji jednog podatka. Ocjenjivanje se vršilo nad 2000 novih uzoraka. Skup za učenje sadrži 4122 uzorka.



Slika 5.3: Dimenzionalnost podataka

Za uzorke s 200 i 400 uzoraka učenje je trajalo dvadesetak minuta. Korištenje maksimalnog broja ekstrahiranih atributa (22 244) proizvelo je konačnu točnost klasifikacije od 94.8% nad 2000 uzoraka za provjeru. Trajanje samog učenja odužilo se na dva dana.

Navedeni rezultati podrazumijevaju prvu razine klasifikacije. To znači da se odjeljuju glavni tipovi tumora u ovisnosti o području nastajanja. Druga razina klasifikacije, odnosno određivanje specifičnih tipova pojedinih tumora zahtjeva učenje novog klasifikatora. Ponovno se koristi kombinirana jezgrena funkcija te genetski algoritam za njeno optimiranje. S obzirom na veliku brojnost različitih vrsta tumora, klasifikacija se na ovaj način razbija u dva dijela. Drugi dio nezavisan je od prvog. Razlog razdvajanja procedure krije se u svojstvima atributa. Kada se svi specifični tumori (podtipovi tumora štitnjače, debelog crijeva, bubrega, limfnih čvorova itd.) žele klasificirati zajedno u samo jednoj razini, velik broj atributa poprima visoku međusobnu koreliranost što rezultira sniženom točnošću ispitivanja. Klasifikator druge razine naučen je na istim podatcima koji su se koristili na prvoj razini. U ovom slučaju cijeli skup se rastavlja na četiri dijela čije su konkretne veličine prikazane u tablici 5.4. Dobiveni podskupovi su za svake pojedine tumore rastavljeni na dio za učenje koji sadrži 70% uzoraka i dio za provjeru koji obuhvaća preostalih 30%. Nakon što se provede postupak učenja,

ponovno je moguće primijeniti Plattovo skaliranje kako bi se dobila ocjena vjerojatnosti. U ovoj konkretnoj situaciji aproksimira se vjerojatnost ispravne klasifikacije specifičnog tipa tumora.

Tablica 5.4: Točnost ispitivanja specifičnih tumora

Područje tumora	Veličina cijelog skupa	Veličina skupa za provjeru	Točnost ispitivanja
Štitnjača	1422	426	87.26%
Debelo crijevo	1680	504	96.44%
Bubreg	1217	365	91.12%
Limfni čvorovi	1803	541	89.42%

6. Zaključak

Glavni klasifikator temeljen je na stroju potpornih strojeva. Empirijski rezultati pokazali su značajan doprinos genetskog algoritma u optimizaciji parametara SVM-a. Transformacija podataka ključan je element klasifikacije nelinearnih podataka. Uz pomoć dobre metaheuristike, omogućen je pronalazak kombinirane jezgrene funkcije koja se razvija posebno za ulazne podatke. Naglasak je na korištenju samih uzoraka pri konstrukciji transformacije što samo može pospješiti točnost klasifikacije.

Ispostavlja se da je analizu molekularnog potpisa tumora moguće vršiti nad komprimiranim modelom. To podrazumijeva smanjenje dimenzionalnosti uzoraka. Pokazano je kako se pristojni rezultati klasifikacije mogu dobiti već korištenjem 200 atributa. Važnost svih atributa modela očituje se u iznimno visokoj točnosti klasifikacije kada se koristi sve $22\,244$ značajke. Molekularni potpis može se dokučiti tako što se postigne dovoljno dobro predviđanje pripadnosti uzoraka. Konkretno se to odnosi na određivanje tipova i podtipova tumora.

Pomoć računala u dijagnostici bit će sve potrebnija. Velike odluke poput određivanja terapije nose sa sobom ogromne rizike. Uz razvoj dovoljno dobrog klasifikatora, moguće je gotovo iskorijeniti te probleme. Procjene opasnosti mogu se računati na temelju pouzdanosti klasifikacije.

Daljnji rad moguće je usmjeriti k razvoju još specifičnijih klasifikatora. Tako će se moći ponuditi veća količina informacija koja naposljetku može biti presudan faktor u trenutcima odluka.

LITERATURA

[1] Mile Šikić & Mirjana Domazet-Lošo. *Bioinformatika*.

[2] Bruce Alberts & Alexander Johnson & Julian Lewis & Martin Raff & Keith Roberts & Peter Walter. *Molecular Biology of the Cell*.

[3] Steve R. Gunn. *Support Vector Machines for Classification and Regression*. 1998.

[4] Dmitriy Fradkin & Ilya Muchnik. *Support Vector Machines for Classification*.

[5] Jan Šnajder. *Predavanja s kolegija Strojno učenje, FER*.

[6] John C. Platt. *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*.

[7] Stefan Lessmann & Robert Stahlbock & Sven F. Cronel. *Genetic Algorithms for Support Vector Machine Model Selection*.

[8] Marko Čupić. *Prirodom inspirirani optimizacijski algoritmi. Metaheuristike.*

[9] Chih-Wei Hsu & Chih-Chung Chang & and Chih-Jen Lin. *A Practical Guide to Support Vector Classification*.

[10] Hsuan-Tien Lin & Chih-Jen Lin & Ruby C. Weng. *A Note on Platt's Probabilistic Outputs for Support Vector Machines*.

[11] Isabelle Guyon & Andre Elisseeff. *An Introduction to Variable and Feature Selection*.

Dodatak A
Izvori skupova podataka

A. Domena hepatitisa: *Carnegie-Mellon University, www.cs.cmu.edu*

B. Otrovne i neotrovne gljive: *archive.ics.uci.edu*

C. Vrste stakla: *B. German, Home Office Forensic Science Service, github.com.*

D. Sunčeve zrake: *Gary Bradshaw, www.ngdc.noaa.gov*

E. Genski sljedovi primata: *genbank.bio.net.*

Analiza molekularnog potpisa tumora uz pomoć strojnog učenja

Sažetak

U uvodnim poglavljima izložena je motivacija problema. Kratko su opisani biološki koncepti čije je poznavanje važno u kasnijim fazama. Glavna tema rada je razvoj generičkog klasifikatora. Naglasak se stavlja na mogućnost određivanja tipova i podtipova tumorskih stanica. U inkrementalnom postupku razvoja koristi se algoritam stroja potpornih vektora (SVM). Prikazan je način optimizacije parametara SVM-a uz pomoć genetskog algoritma. Opisana je implementirana paralelizacija posla pri učenju. Izgrađeni klasifikator ispitan je na različitim vrstama skupova podataka. Ukratko su opisane strategije višerazredne klasifikacije. Predstavljen je način procjene vjerojatnosti pripadnosti uzoraka pojedinim razredima klasifikacije u vidu Plattovog skaliranja. Izloženi su rezultati i uspješnost konačnog modela.

Ključne riječi: klasifikacija, jezgrene funkcije, genetski algoritam, tumorske stanice

Analysis of molecular signature of tumors with machine learning

Abstract

Motivation for this assignment's problem is presented in introductory chapters. Biological concepts, key to understanding later work, are briefly described. Main topic in this assignment is development of generic classifier. Emphasis is put on possibility of determining types and subtypes of tumor cells. Algorithm Support Vector Machine is used in incremental building procedure. Optimization of SVM's parameters with genetic algorithm is described thoroughly. Implementation of parallel computing for work needed for training is also presented. Built classifier is tested on various types of datasets. Strategies of multiclass classification are shortly described. A way of creating a probability of class association approximation is shown by method of Platt scaling. Results and achievements of final model are stated in the end.

Keywords: classification, kernel functions, genetic algorithm, tumor cells