UNIVERZITET U BEOGRADU
MATEMATIČKI FAKULTET

Jovana J. Kovačević

STRUKTURNA PREDIKCIJA FUNKCIJE PROTEINA I ODNOS FUNKCIONALNIH KATEGORIJA PROTEINA I NJIHOVE NEUREĐENOSTI

doktorska disertacija

Beograd, 2015.

UNIVERSITY OF BELGRADE
FACULTY OF MATHEMATICS

Jovana J. Kovačević

STRUCTURED PREDICTION OF PROTEIN FUNCTION AND RELATIONSHIP BETWEEN FUNCTIONAL CATEGORIES OF PROTEINS AND THEIR DISORDER

doctoral dissertation

Belgrade, 2015.

Mentor:
dr Gordana Pavlović-Lažetić, redovni profesor
Univerzitet u Beogradu, Matematički fakultet

Članovi komisije:

dr Miloš Beljanski, naučni savetnik
Institut za opštu i fizičku hemiju, Beograd

dr Nenad Mitić, vanredni profesor
Univerzitet u Beogradu, Matematički fakultet

dr Predrag Radivojac, redovni profesor
Fakultet za informatiku i računarstvo, Univerzitet Indijana,
Blumington, Indijana, SAD

dr Mladen Nikolić, docent
Univerzitet u Beogradu, Matematički fakultet

Datum odbrane: _____

Mami,
Tatjana Jovašević-Kovačević (1961-2008)
Baji,
Nadežda-Nada Jovašević (1930-2007)
Vladi.
prof. dr Vladan Jovašević (1921-1998)

Prijatna mi je dužnost da se zahvalim svima koji su posredno ili neposredno učestvovali u izradi ove disertacije.

Na prvom mestu želim da se zahvalim svom mentoru, dr Gordani Pavlović-Lažetić, redovnom profesoru Matematičkog fakulteta u Beogradu, na velikoj pomoći i bezrezervnoj podršci tokom mojih doktorskih studija. Njena izuzetna profesionalnost sa jedne i ogromno strpljenje i razumevanje sa druge strane pomogli su mi da se izborim sa svim izazovima koje nosi doktorat.

Dr Predragu Radivojcu, redovnom profesoru Fakulteta za informatiku i računarstvo u Blumingtonu, SAD, zahvalna sam na ukazanom poverenju i pruženoj prilici za zajednički rad koji je u velikoj meri usmerio moje naučno usavršavanje. Iskreno se nadam da ću iskustva stečena u njegovoj istraživačkoj grupi uspeti da implementiram u Srbiji i time doprinesem povećanju kvaliteta naučnog i nastavnog rada u oblasti bioinformatike.

Zahvaljujem se dr Nenadu Mitiću, vanrednom profesoru Matematičkog Fakulteta u Beogradu i dr Milošu Beljanskom, naučnom savetniku Instituta za opštu i fizičku hemiju, na mnogim korisnim savetima tokom naše dugogodišnje saradnje u Bioinformatičkoj istraživačkoj grupi.

Naročitu zahvalnost dugujem dr Mladenu Nikoliću, docentu Matematičkog Fakulteta u Beogradu, za dragocenu pomoć u raznim fazama mog doktorskog rada.

Veoma sam zahvalna i dr Jeleni Graovac i dr Saši Malkovu, docentima Matematičkog Fakulteta u Beogradu, na podršci i ohrabrivanju za vreme pisanja disertacije.

Najveća zahvalnost ide mojoj porodici na beskrajnom razumevanju i podršci u toku izrade ove teze. Na kraju, posebno hvala mom Nikoli za ljubav, pažnju i veru u moj uspeh.

Naslov disertacije: Strukturna predikcija funkcije proteina i odnos funkcionalnih kategorija proteina i njihove neuređenosti

Rezime:

Proteini predstavljaju najvažniju grupu biomolekula u živom svetu. Različite funkcije koje imaju u svakom organizmu jedinstvene su i nezamenljive, počev od raznovrsnih ćelijskih procesa, preko njihove strukturalne uloge, uloge katalizatora, velikog broja metaboličkih funkcija i slično. Poznavanje i razumevanje funkcija proteina je stoga esencijalno u istraživanju bilo kog biološkog procesa, sa posebnim naglaskom na oboljenja ljudi, s obzirom da se mnoga od njih mogu pojaviti zbog funkcionalnih mutacija.
U ovom radu biće predstavljeno istraživanje ovog domena kroz dva različita pristupa. U prvom, funkcija proteina posmatrana je kroz GO ontologije, koje podrazumevaju predstavljanje funkcije proteina kroz tri velika usmerena aciklička grafa funkcija: jedan je vezan za biološke procese, drugi za ćelijske komponente, a treći za molekulsku funkciju. Svaki od njih sadrži više hiljada čvorova, pri čemu svaki čvor određuje specifičniju funkciju od svojih predaka. Zadatak ovog dela istraživanja je razvoj prediktora funkcije proteina na osnovu njene primarne sekvence primenom metode strukturalnih podržavajućih vektora koja predstavlja generalizaciju poznate metode podržavajućih vektora na strukturalni izlaz.
Jedno od osnovnih načela molekularne biologije predstavlja paradigma struktura-funkcija po kojoj je 3D struktura proteina blisko povezana sa njegovom ulogom u organizmu. Utvrđeno je da su neuređeni proteini (kojima nedostaje 3D struktura) kao i neuređeni delovi proteina u vezi sa teškim savremenim bolestima i kao takvi su predmet aktuelnih istraživanja. U drugom pravcu razmatrana je veza funkcionalnih kategorija proteina sa njihovom neuređenošću, kao i sa drugim fizičko-hemijskim karakteristikama proteina. Funkcija proteina ovde je posmatrana kroz 25 osnovnih funkcija proteina koje su grupisane u 4 funkcionalne grupe. U ovoj disertaciji su prikazani rezultati detaljne analize nad velikim skupom proteina za koje je neuređenost određena primenom javno dostupnih alata.

Ključne reči: strukturalna klasifikacija, bioinformatika, funkcija proteina, neuređenost proteina

Naučna oblast: Računarstvo

Uža naučna oblast: Istraživanje podataka

UDK broj: [[519.17+519.863]:[004.023+004.925.8]]:577.322(043.3)

Title of the dissertation: Structured prediction of protein function and relationship between functional categories of proteins and their disorder

Abstract:

Proteins represent the most important groups of biomoleculs. Different functions that they carry out in each organism are unique and irreplaceable, including versatile cellular processes, structural role of proteins, catalytic function, a number of metabolic functions and so on. Knowing and understanding protein function is therefore essential in investigation of any biological process, especially of human diseases since a lot of them are caused by functional mutations.

In this paper, we represent investigation of protein function domain through two different approaches. In the first one, protein function is represented by GO ontologies with the structure of a directed acyclic graph. There are three GO ontologies: one for functions regarding biological processes, one for functions regarding cellular components and one for molecular functions. Each ontology contains several thousands of nodes, where every node determines more specific function than his ascendants. The task of this part of research was to develop a software for predicting protein function from its primary sequence based on structural support vector machines method which represents generalization of well-known support vector machines method on structural output.

Structure-function paradigm is one of basic concepts in molecular biology, stating that 3D proten structure is closely connected to its role in organism. It has been detected that disordered proteins (the ones that lack 3D structure) and disordered regions of proteins are related with severe contemporary illnesses, which contributed to their popularity in modern research. In another aspect, we investigated the relationship between proteins' functional categories and their disorder, as well ad with other physico-chemical characteristics of proteins. Here, protein function has been observed through 25 elementary functions grouped in 4 functional groups. In this work, we present results of thorough analysis over large protein dataset where disorder has been determined computationally, using publicly available tools.

Key words: Structured prediction, bioinformatics, protein function, protein disorder

Scientific field : Computer Science

Scientific subfield : Data mining

UDK number: [[519.17+519.863]:[004.023+004.925.8]]:577.322(043.3)

Sadržaj

1. Uvod

Strukturna predikcija funkcija proteina

Određivanje funkcije proteina predstavlja važan korak neophodan za različita biološka istraživanja. S obzirom da mnoga oboljenja potiču od funkcionalnih mutacija, precizna detekcija funkcionalnog ponašanja proteina veoma je značajna u savremenoj bioinformatici. Aktuelne eksperimentalne metode za funkcionalnu anotaciju proteina su suviše vremenski i materijalno zahtevne za veliki priliv novootkrivenih proteinskih sekvenci čiji broj raste sa svakim sekvencionisanim genomom. Zbog toga je poslednjih godina intenziviran razvoj softverskih alata za automatsku predikciju funkcije proteina, koji mogu predstavljati prvi korak u usmeravanju skupih laboratorijskih resursa.

Klasičan pristup problemu predikcije funkcije podrazumeva takozvano prenošenje funkcionalne anotacije sa sličnih proteina, za koje su funkcije eksperimentalno utvrđene, na dati protein, pri čemu se sličnost može utvrđivati globalnim, lokalnim i višestrukim poravnanjem, na osnovu zajedničkih šablona u sekvenci proteina, na osnovu evolutivne povezanosti, na osnovu slične sekundarne strukture itd. Napredniji pristup se sastoji u primeni različitih algoritama mašinskog učenja koji se treniraju na skupovima već funkcionalno anotiranih proteina, gde na osnovu odabranih karakteristika proteina pokušavamo da zaključimo koje funkcije ima dati protein. U zavisnosti od toga da li koristimo nadgledane ili nenadgledane metode mašinskog učenja, problemu predviđanja funkcije možemo pristupiti kao problemu klasifikacije ili klasterovanja. U okviru prvog pristupa korišćeni su sledeći algoritmi: metod podržavajućih vektora [97], neuralne mreže [7], višeslojni perceptroni [51], Markovljeva slučajna polja [48], bajesovske mreže [107], kernel logistička regresija [43] i druge. U okviru drugog pristupa korišćeni su sledeći algoritmi: Markovljevo klasterovanje [57], hijerarhijsko klasterovanje [2], verovatnosni grafički modeli [66], model skrivenog modularnog slučajnog polja [67] i drugi. Naučna zajednica koja se bavi predviđanjem funkcije proteina korišćenjem tehnika mašinskog učenja svake dve godine održava

takmičenje najnovijih prediktora funkcije (CAFA – Critical Assessment of Function Annotation experiment)[1].

Uobičajeni način predstavljanja funkcije proteina definisan je kroz Gene Ontology (GO) projekat [47]. GO razdvaja sve moguće funkcije proteina na tri različita usmerena aciklička grafa: ontologija molekulskih funkcija, ontologija bioloških procesa i ontologija ćelijskih komponenti. Svaki čvor u ontologiji predstavlja jednu funkciju, a svaka grana predstavlja vezu između čvorova – funkcija koje povezuje. Svaki čvor definiše specifičniju funkciju nego njegov predak: na primer, čvor koji označava funkciju *Vezivanje miRNK* ima pretke redom *Vezivanje RNK*, *Vezivanje nukleinskih kiselina*, *Vezivanje* i *Molekulska funkcija*. Svaka ontologija ima nekoliko hiljada čvorova i u okviru svake anotirano je po nekoliko desetina hiljada proteina. U ovoj tezi razvijen je model i programski sistem za automatsko predviđanje funkcije proteina na osnovu njegove sekvence korišćenjem metoda strukturne klasifikacije, preciznije metode stukturalnih podržavajućih vektora koja predstavlja proširenje metode podržavajućih vektora za strukturalni izlaz.

Funkcionalne kategorije proteina i njihova neuređenost

Funkcionalne kategorije proteina predstavljaju rezultat klasterovanja proteina iz kompletnih genoma različitih organizama na osnovu njihove evolutivne povezanosti. Na ovaj način je nastala javno dostupna baza proteina pod nazivom COG (Cluster of Orthologous Groups). Proteini su tako podeljeni na 25 klastera, u svakom klasteru su proteini koji obavljaju istu funkciju (npr. nuklearna struktura, transkripcija, transport i metabolizam lipida,...), a oni su dalje grupisani u četiri funkcionalne kategorije: ćelijski procesi, skladištenje i obrada informacija, metabolizam i nedovoljno okarakterisani proteini.

Proteini u prostoru najčešće zauzimaju uređenu prostornu strukturu u obliku spirala ili ravni. Međutim, za neke proteine eksperimentalno je pokazano da su neuređeni, što znači da nemaju fiksiranu 3D strukturu, u pojedinim regionima proteina ili kompletno. Pored eksperimentalnih metoda, postoji veliki broj alata za automatsko određivanje neuređenosti proteinske sekvence [49].

[1] http://www.biofunctionprediction.org

Poznato je da postoji veza neuređenosti proteina sa njihovom funkcijom (ulogom u organizmu) kao i sa različitim genomičkim, metaboličkim i ekološkim karakteristikama organizma kom protein pripada, kao i veza sa lokacijom pojavljivanja neuređenosti u proteinu. Istraživanja nad neuređenim proteinima su od velikog značaja jer je utvrđeno da su oni povezani sa nekim od najtežih savremenih bolesti. U ovoj tezi biće izvršena analiza neuređenosti velikog skupa prokariotskih i eukariotskih proteina u odnosu na njihove funkcionalne kategorije i u odnosu na sastav neuređenih regiona.

Ciljevi i organizacija teze

U prvom delu istraživanja (glave 2 i 3) opisan je razvoj prediktora funkcije proteina metodom strukturalnih podržavajućih vektora.

- implementacija zajedničkog predstavljanja ulaza i izlaza trening podataka;
- implementacija algoritma za određivanje maksimuma funkcije cilja;
- razmatranje različitih funkcija za izbor funkcije gubitka;
- ispitivanje uticaja genetskog porekla proteina na kvalitet predviđanja.

Cilj ovog dela istraživanja bio je izgraditi alat za automatsko predviđanje funkcije proteina koji će realnom vremenu odrediti skup funkcija novog izolovanog proteina i koji će svojom pouzdanošću biti konkurentan sa postojećim metodama. Ovakav sistem može biti od koristi za usmeravanje laboratorijskih tehnika za određivanje funkcije proteina i može doprineti uštedi skupih eksperimentalnih resursa, što bi omogućilo masovnu analizu novootkrivenih proteina.

U drugom delu istraživanja (glava 4) opisana je analiza neuređenosti proteina i odnos prema funkcionalnim kategorijama. Neuređenost proteina posmatrana je iz dva ugla. Sa jedne strane, izvršena je analiza nad velikim skupom podataka kod kojih je neuređenost utvrđena primenom postojećeg softvera za predviđanje, u odnosu na funkcionalne kategorije proteina i različite osobine organizama kojima proteini pripadaju. Sa druge strane, izvršena je analiza nad malim skupom podataka kod kojih je neuređenost utvrđena eksperimentalno, u odnosu na lokaciju neuređenih regiona u proteinu i njihov sastav.

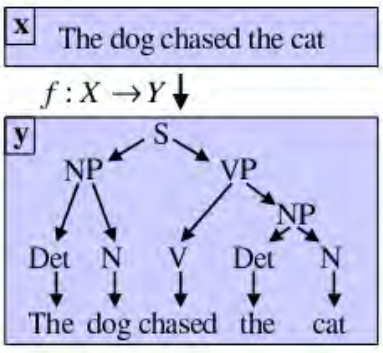
Na osnovu ovakve analiza neuređenosti proteina, za dati novi protein može se sugerisati njegova pripadnost određenoj funkcionalnoj kategoriji, što je provereno na proteinima za koje je poznata funkcija na osnovu GO anotacije (poglavlje 5.5). Pored toga, utvrđivanje stepena neuređenosti pojedinih aminokiselina, tj njihove zastupljenosti u neuređenim regionima, može da doprinese unapređenju metoda za automatsko predviđanje neuređenosti proteina.

2. Strukturna klasifikacija

Posmatrajmo opšti problem učenja preslikavanja skupa ulaznih vektora $\boldsymbol{x} \in \mathcal{X}$ u skup klasa $\boldsymbol{y} \in \mathcal{Y}$, na osnovu trening skupa $(\boldsymbol{x_1}, \boldsymbol{y_1}), \ldots, (\boldsymbol{x_n}, \boldsymbol{y_n}) \in \mathcal{X} \times \mathcal{Y}$ generisanog iz nepoznate, ali fiksirane raspodele verovatnoća. Za razliku od višeklasne klasifikacije, gde se prostor svih mogućih klasa $\mathcal{Y}$ sastoji od konačnog skupa oznaka klasa, $\mathcal{Y} = \{1, \ldots, N\}$, ili regresije, gde $\mathcal{Y} = \mathbb{R}$ a izlaz predstavlja skalarnu vrednost, razmatramo slučajeve kada su elementi skupa $\mathcal{Y}$ struktuirani objekti, kao što su nizovi, niske, stabla ili grafovi [100]. Ovakvi problemi sreću se u različitim oblastima:

1. obrada prirodnih jezika
    - automatsko prevođenje (izlaz: rečenica)
    - parsiranje rečenice (izlaz: stablo)
2. bioinformatika
    - predviđanje sekundarne strukture (izlaz: niz)
    - predviđanje funkcije proteina (izlaz: usmereni aciklički podgraf)
3. obrada glasa
    - automatska transkripcija (izlaz: rečenica)
    - čitanje teksta (izlaz: audio signal)
4. robotika
    - planiranje (izlaz: niz akcija)

Zadatak strukturne klasifikacije je da nauči preslikavanje $f : \mathcal{X} \to \mathcal{Y}$ da za nepoznati ulazni vektor odredi klasu kojoj pripada, pri čemu je ta klasa kompleksni, struktuirani objekat. Preslikavanje f zovemo funkcijom cilja za problem strukturne klasifikacije. Na primer, ako posmatramo problem parsiranja prirodnog jezika, funkcija cilja preslikava datu rečenicu $\boldsymbol{x}$ u stablo izvođenja $\boldsymbol{y}$ (slika 2.1).



Slika 2.1: *Primer funkcije cilja za problem određivanja stabla izvođenja u formalnoj gramatici. Slika je preuzeta iz [100].*

Rešavanju postavljenog problema pristupićemo na sledeći način: umesto da učimo funkciju cilja f, uvodimo diskriminantnu funkciju $F : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ koja predstavlja meru kompatibilnosti ulaza $\boldsymbol{x}$ sa izlazom $\boldsymbol{y}$. Što je $F(\boldsymbol{x}, \boldsymbol{y})$ veće, to ulazu $\boldsymbol{x}$ više odgovara klasa $\boldsymbol{y}$. Predviđanje klase za dati ulazni vektor $\boldsymbol{x}$ stoga možemo definisati kao određivanje one klase $\boldsymbol{y} \in \mathcal{Y}$ za koju diskriminantna funkcija $F(\boldsymbol{x}, \boldsymbol{y})$ ima maksimalnu vrednost, a funkciju cilja možemo zapisati na sledeći način:

$$f(\boldsymbol{x}; \boldsymbol{w}) = \underset{\boldsymbol{y} \in \mathcal{Y}}{\operatorname{argmax}}(F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}))$$

gde $\boldsymbol{w} \in \mathbb{R}^n$ označava vektor realnih koeficijenata koji se određuju treniranjem modela. Ograničićemo se na prostor linearnih funkcija, odnosno podrazumevamo da je funkcija F sledećeg oblika:

$$F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}) = \langle \Psi(\boldsymbol{x}, \boldsymbol{y}), \boldsymbol{w} \rangle$$

gde je $\Psi : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$ funkcija koja preslikava instancu $(\boldsymbol{x}, \boldsymbol{y})$ u vektor realnih vrednosti koji predstavlja zajednički zapis ulaznog vektora $\boldsymbol{x}$ i izlaznog vektora $\boldsymbol{y}$. Svakom elementu vektora $\Psi(\boldsymbol{x}, \boldsymbol{y})$ odgovara jedan element vektora $\boldsymbol{w}$ koji mu dodaje težinu, pozitivnu ili negativnu.

Izbor funkcije Ψ direktno zavisi od konkretnog problema koji rešavamo. Na primer, za problem parsiranja prirodnog jezika, funkcija Ψ se može definisati kao histogram vektor broja pravila koja su primenjena prilikom izvođenja $\boldsymbol{y}$ za rečenicu $\boldsymbol{x}$ (slika 2.2). Ukoliko se neko pravilo pojavljuje više puta u trening skupu, element vektora $\boldsymbol{w}$ koji odgovara takvom pravilu biće veći od onog koji odgovara pravilu koje se manje puta pojavljuje.

$$\Psi(\mathbf{x},\mathbf{y}) = \begin{pmatrix} 1 \\ 0 \\ 2 \\ 1 \\ \vdots \\ 0 \\ 2 \\ 1 \\ 1 \\ 1 \end{pmatrix} \begin{matrix} S \rightarrow NP\,VP \\ S \rightarrow NP \\ NP \rightarrow Det\ N \\ VP \rightarrow V\ NP \\ \\ Det \rightarrow dog \\ Det \rightarrow the \\ N \rightarrow dog \\ V \rightarrow chased \\ N \rightarrow cat \end{matrix}$$

Slika 2.2: *Primer funkcije Ψ za problem određivanja stabla izvođenja u formalnoj gramatici, za instancu $(\boldsymbol{x}, \boldsymbol{y})$ navedenu na slici 2.1. Vektor $\Psi(\boldsymbol{x}, \boldsymbol{y})$ označava da se u stablu izvođenja $\boldsymbol{y}$ rečenice $\boldsymbol{x}$ pravilo $S \rightarrow NPVP$ pojavilo jednom, pravilo $S \rightarrow SP$ nijednom, pravilo $NP \rightarrow DetN$ dva puta, ..., pravilo $N \rightarrow cat$ jednom. Slika je preuzeta iz [100].*

Kvalitet strukturne klasifikacije meri se pomoću funkcije $\Delta : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$, takozvane funkcije gubitka koja meri različitost dva izlazna vektora, tačnog $\boldsymbol{y}$ i predviđenog $\boldsymbol{y'}$. S obzirom da funkcija gubitka predstavlja grešku klasifikacije, obično se za ove funkcije gubitka biraju preslikavanja za koja važi $\Delta(\boldsymbol{y}, \boldsymbol{y}) = 0$ i $\Delta(\boldsymbol{y}, \boldsymbol{y'}) > 0\ \forall \boldsymbol{y'} \neq \boldsymbol{y}$. Kod binarne i multiklasifikacije rezultat se obično procenjuje $0-1$ funkcijom - 0, ako je primer tačno klasifikovan a 1 u suprotnom. Kod strukturnog izlaza upotreba ovakve funkcije gubitka nije pogodna. Na primer, ako poredimo različita stabla izvođenja kod problema parsiranja prirodnog jezika, istom vrednošću bismo ocenili predviđanje koje se od tačnog razlikuje u samo jednom gramatičkom pravilu i predviđanje koje je sasvim različito od tačnog. U ovom konkretnom slučaju, primerenija mera kvaliteta predviđanja bi bila npr. F_1 mera, harmonijska sredina preciznosti i odziva izračunata na osnovu preklapanja čvorova dva stabla izvođenja [80].

Strukturna klasifikacija se dakle svodi na određivanje optimalnog vektora $\boldsymbol{w}$ na osnovu datog skupa podataka koji će pravilno klasifikovati ne samo primere iz tog skupa već i nepoznate primere. Metode strukturne klasificije (CRF [38], M^3N [16], SSVM [100], strukturalni perceptron [8]) se međusobno razlikuju u načinu na koji treniraju vektor $\boldsymbol{w}$.

2.1 Minimizacija empirijskog rizika

Strukturna klasifikacija predstavlja jedan generalizovani oblik nadgledanog učenja. Pre nego što opišemo jednu metode strukturne klasifikacije, navodimo formalne osnove svih metoda nadgledanog učenja.

Kod nadgledanog učenja, cilj je na osnovu trening skupa $S = \{(\boldsymbol{x_1}, \boldsymbol{y_1}), \ldots, (\boldsymbol{x_n}, \boldsymbol{y_n})\}$ izvučenog iz fiksirane raspodele $P(\boldsymbol{x}, \boldsymbol{y})$ pronaći funkciju f koja najbolje moguće predviđa primere iz trening skupa, odnosno za koju je očekivanje funkcije gubitka Δ nad trening skupom minimalno. Očekivana vrednost funkcije gubitka Δ predstavlja rizik funkcije cilja f:

$$R_P^{\Delta}(f) = \mathbb{E}[\Delta(\boldsymbol{y}, f(\boldsymbol{x}; \boldsymbol{w}))] = \int_{\mathcal{X}\times\mathcal{Y}} \Delta(\boldsymbol{y}, f(\boldsymbol{x}; \boldsymbol{w}))dP(\boldsymbol{x}, \boldsymbol{y})$$

S obzirom da je raspodela P nepoznata, a skup trening primera konačan, rizik funkcije cilja uobičajeno se procenjuje empirijskim rizikom R_S^{Δ}:

$$R_S^{\Delta}(f) = \frac{1}{n}\sum_{i=1}^{n} \Delta(\boldsymbol{y_i}, f(\boldsymbol{x_i}; \boldsymbol{w})) \tag{2.1}$$

koji predstavlja očekivanje funkcije gubitka pri empirijskoj raspodeli indukovanoj trening skupom S. Kako bi se izbegao problem preprilagođavanja (eng. *overfitting*), obično se empirijskom riziku dodaje regularizaciona funkcija $R(\boldsymbol{w})$ čime dobijamo regularizovani empirijski rizik R_{SR}^{Δ}:

$$R_{SR}^{\Delta}(f) = \frac{1}{n}\sum_{i=1}^{n} \Delta(\boldsymbol{y_i}, f(\boldsymbol{x_i}; \boldsymbol{w})) + \lambda R(\boldsymbol{w}) \tag{2.2}$$

Uloga regularizacione funkcije je da smanji kompleksnost modela odnosno da osigura da male promene u vrednostima ulaznog vektora $\boldsymbol{x}$ neće imati veliki uticaj na performanse klasifikatora. Zbog toga je pogodno da su elemenati w_i vektora $\boldsymbol{w}$ male vrednosti, a idealno bi bilo da su većinom nule. Na primer, minimizacijom sume indikatorskih funkcija $\sum_{i=1}^{n} \mathbb{I}(w_i \neq 0)$ bismo postigli taj cilj. Ipak, indikatorska funkcija nije dobar izbor jer je NP-teška za minimizaciju, pa se stoga koristi l_1 norma kao njena aproksimacija: $||\boldsymbol{w}||_1 = \sum_{i=1}^{n} |w_i|$.

Parametar λ u (2.2) predstavlja parametar regularizacije koji određuje koji će od dva sabirka imati veći uticaj prilikom minimizacije funkcije $R_{SR}^{\Delta}(f)$ (da li očekivanje greške na trening skupu ili kompleksnost modela) i podešava

se prilikom treniranja modela (npr. unakrsnom validacijom).

Teorijski je moguće minimizovati direktno funkciju R_{SR}^{Δ}. Međutim, funkcija $\Delta(\boldsymbol{y_i}, f(\boldsymbol{x_i}; \boldsymbol{w})) = \Delta(\boldsymbol{y_i}, \text{argmax}_{\boldsymbol{y}\in\mathcal{Y}}(F(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}))$ nije neprekidna po $\boldsymbol{w}$ (zbog funkcije arg max) već deo po deo konstantna, što onemogućava korišćenje popularnih gradijentnih metoda za optimizaciju. Rezultati iz statističke teorije učenja pokazuju da je dovoljno minimizovati konveksnu funkciju $L(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w})$ koja odozgo ograničava R_{SR}^{Δ} bez umanjenja kvaliteta predviđanja [109]. Ovakve funkcije su poznate i kao surogat funkcije. Ako za regularizacionu funkciju izaberemo kvadratnu normu, treniranje modela se stoga svodi na minimizaciju funkcije

$$\lambda||\boldsymbol{w}||^2 + L(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{w}) \tag{2.3}$$

Izbor surogat funkcija za empirijski rizik i regularizaciona funkcija definišu različite metode nadgledanog učenja. Primeri surogat funkcija koje se koriste u strukturnoj klasifikaciji biće dati u poglavlju 2.3.

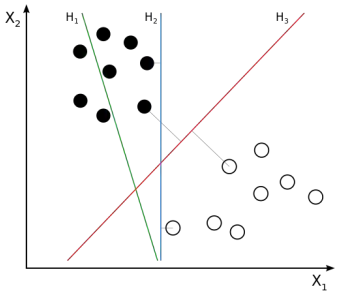
2.2 Pojam margine i njena maksimizacija

Jedan od načina rešavanja problema strukturne klasifikacije je kroz pristup maksimizacije margine. Najpre ćemo definisati metodu u jednostavnijem kontekstu, u binarnoj klasifikaciji, a potom ćemo ovaj pojam generalizovati na strukturnu klasifikaciju.

Pojam margine u binarnoj klasifikaciji - primer metode podržavajućih vektora

Binarna klasifikacija nam daje odgovor na pitanje da li dati primer $\boldsymbol{x}$ pripada nekoj klasi ili ne. Neka su dati trening primeri $\{(\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_n}, y_n)\}$, gde su ulazni podaci $\boldsymbol{x_i} \in \mathbb{R}^d$ d-dimenzioni vektori a izlazni podaci $y_i \in \{-1, 1\}$ označavaju vrednost koju želimo da predvidimo. Na primer, zadatak može biti da predvidimo da li će sutra padati kiša na osnovu rezultata raznih merenja prikupljenih danas. Vrednost $y_i = 1$ označava da će padati kiša, a $y_i = -1$ da neće. Pretpostavimo da smo sakupili podatke za svaki dan prethodne godine i da sada imamo 365 tačaka u d-dimenzionalnom vektorskom prostoru, kojima je dodeljena klasa 1 ili -1. Ideja SVM metode je jednostavna: konstruisati optimalnu hiperravan u ovom d-dimenzionalnom vektorskom prostoru koja razdvaja pozitivne i negativne primere. Kao što

možemo videti na slici 2.3, postoji više hiperravni koje razdvajaju trening primere, ali nisu sve jednako dobre. Očigledno je da hiperravan H_1 nije dobro rešenje jer ne razdvaja trening primere, kao i da hiperravan H_3 bolje razdvaja trening primere nego hiperravan H_2. Ovo opažanje možemo formalizovati na sledeći način: kod hiperravni H_3 rastojanje od najbližeg trening primera veće je od odgovarajućeg rastojanja kod hiperravni H_2.



Slika 2.3: *SVM - optimalna margina.*

Geometrijski, hiperravan predstavljamo jednačinom $\boldsymbol{w} \cdot \boldsymbol{x} - b = 0$, gde je vektor $\boldsymbol{w}$ normalan na hiperravan, a parametar b određuje rastojanje hiperravni od koordinatnog početka. Ukoliko su trening primeri linearno separabilni (kao na slici 2.4), onda možemo odabrati dve hiperravni koje ih razdvajaju tako da se između tih hiperravni ne nalazi nijedan trening primer:

$$\boldsymbol{w} \cdot \boldsymbol{x} - b = 1$$

i

$$\boldsymbol{w} \cdot \boldsymbol{x} - b = -1$$

Rastojanje između ovih hiperravni iznosi $\frac{2}{||\boldsymbol{w}||}$ i predstavlja *marginu* koju želimo da maksimizujemo. Da bismo osigurali da će svi pozitivni primeri ostati sa jedne strane hiperravni a svi negativni sa druge, uvodimo uslove:

$$\boldsymbol{w} \cdot \boldsymbol{x_i} - b \geq 1, \text{ako je } \boldsymbol{x_i} \text{ pozitivan trening primer}$$

i

$$\boldsymbol{w} \cdot \boldsymbol{x_i} - b \leq 1, \text{ako je } \boldsymbol{x_i} \text{ negativan trening primer}$$

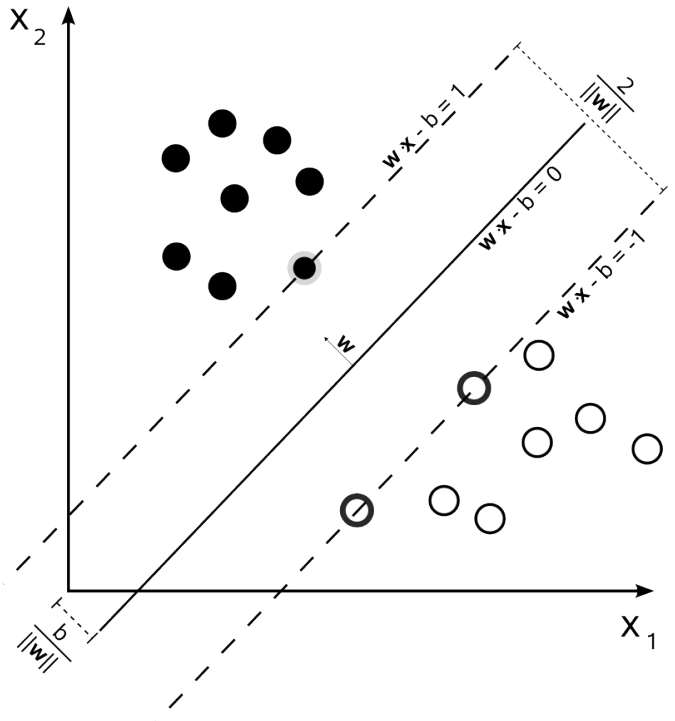
Ove uslove možemo zapisati jedinstvenom nejednakošću:

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x_i} - b) \geq 1 \tag{2.4}$$

Maksimizacija margine $\frac{2}{||\boldsymbol{w}||}$ ekvivalentna je minimizaciji funkcije $\frac{||\boldsymbol{w}||^2}{2}$, pri čemu kvadrat dodajemo kako bismo dobili konveksnu funkciju koju možemo

lako minimizovati. Uzevši u obzir uslove (2.4), dobijamo optimizacioni problem metode SVM sa čvrstom marginom:

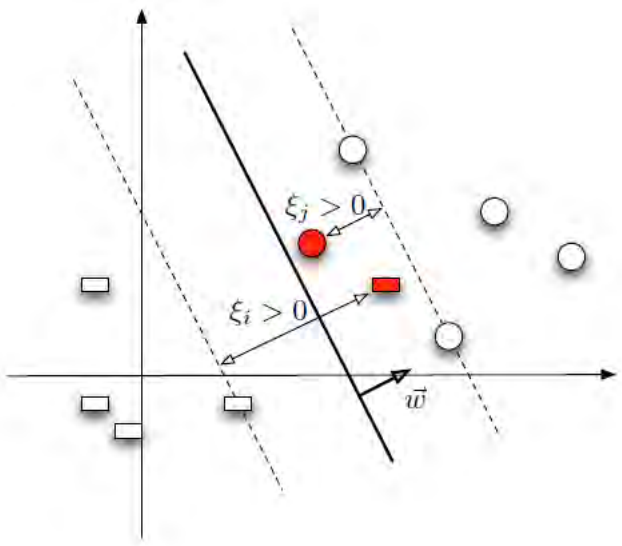
$$\min_{\boldsymbol{w},b} \frac{||\boldsymbol{w}||^2}{2}$$
$$\forall i \in \{1,\ldots,n\}\ y_i(\boldsymbol{w}\cdot\boldsymbol{x_i} - b) \geq 1$$



Slika 2.4: *SVM - čvrsta margina.*

U praktičnim primenama malo je primera kada su podaci linearno separabilni. Otud potreba da se prilikom konstruisanja optimalne hiperravni dopusti mogućnost da neki trening primeri budu unutar margine, a neki čak da budu pogrešno klasifikovani. Ovakvi slučajevi bi se kažnjavali pomoću takozvanih labavih (eng. *slack*) promenljivih ξ_i koje su nenegativne. Postoji po jedna labava promenljiva ξ_i za svaki trening primer $\boldsymbol{x_i}$. Ukoliko je $\boldsymbol{x_i}$ sa "prave" strane hiperravni, odgovarajuća labava promenljiva ξ_i biće jednaka nuli, a u suprotnom će biti veća od nule (slika 2.5). Optimizacioni problem u ovom slučaju možemo formulisati na sledeći način:

$$\min_{\boldsymbol{w},\boldsymbol{\xi},b} \frac{||\boldsymbol{w}||^2}{2} + C\sum_{i=1}^{n}\xi_i$$
$$\forall i \in \{1,\ldots,n\}\ y_i(\boldsymbol{w}\cdot\boldsymbol{x_i} - b) \geq 1 - \xi_i, \xi_i \geq 0$$

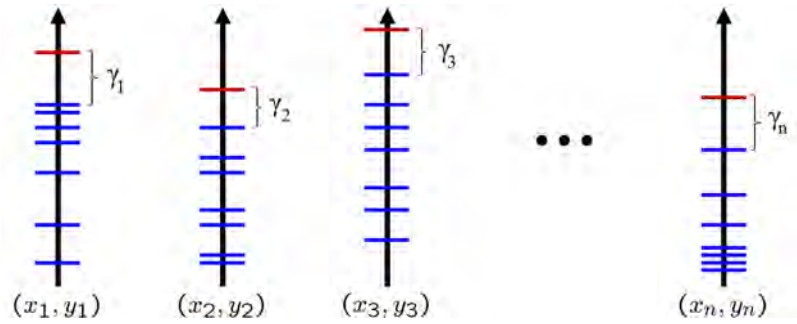


Slika 2.5: *SVM - meka margina. Trening primer i (crveni pravougaonik) je pogrešno klasifikovan (nalazi se sa pogrešne strane hiperravni) i zato je* $\xi_i > 0$. *Trening primer j (crveni krug) je tačno klasifikovan (nalazi se sa prave strane hiperravni) ali se nalazi u oblasti margine i zato je* $\xi_j > 0$.

Parametar C balansira između dva suprotstavljena cilja: da margina bude što šira i da greška na trening skupu, koja je predstavljena sumom labavih promenljivih, bude što manja. Manje vrednosti parametra C daju hiperravni sa širom marginom i sa više pogrešno klasifikovanih trening primera, dok je kod većih vrednosti parametra C obrnuto. Vrednost ovog parametra obično se određuje unakrsnom validacijom.

Pojam margine u strukturnoj klasifikaciji - primer metode strukturnih podržavajućih vektora

Metoda strukturnih podržavajućih vektora (eng. *Structured support vector machines*, skraćeno SSVM) predstavlja generalizaciju metode podržavajućih vektora na strukturni izlaz i jednu od metoda strukturne klasifikacije. Dok kod binarnih SVM pojam margine ima veoma intuitivnu, geometrijsku interpretaciju, kod SSVM to nije slučaj. U binarnom kontekstu postoji jedinstvena margina koja predstavlja rastojanje između onih trening primera koji su najbliži hiperravni koja razdvaja dve klase. U strukturnom kontekstu marginu definišemo na nivou svakog pojedinačnog trening primera kao razliku skora kompatibilnosti ulaznog podatka $\boldsymbol{x_i}$ i za njega poznatog izlaznog



Slika 2.6: *Margina u strukturnom kontekstu. Za svaki trening primer* $(\boldsymbol{x_i}, \boldsymbol{y_i})$ *prikazana je po jedna vertikalna osa na kojom su crticama obeležene vrednosti funkcije skora* $F(\boldsymbol{x_i}, \boldsymbol{y}) = \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle$*, i to crvenom bojom za* $\boldsymbol{y} = \boldsymbol{y_i}$ *i plavom za* $\boldsymbol{y} \neq \boldsymbol{y_i}$*.*

podatka $\boldsymbol{y_i}$ i maksimalnog skora kompatibilnosti istog ulaznog podatka i ostalih izlaznih podataka:

$$\gamma_i = \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle - \max_{\boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}} \{ \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \} \tag{2.5}$$

Želimo da odredimo koeficijente $\boldsymbol{w}$ tako da predviđanje trening primera bude tačno (za svaki trening primer $\boldsymbol{x_i}$ skor kompatibilnosti $F(\boldsymbol{x_i}, \boldsymbol{y})$ bude najviši upravo za $\boldsymbol{y} = \boldsymbol{y_i}$) kao i da margina γ_i bude maksimalna (razlika između skora kompatibilnosti $\boldsymbol{x_i}$ za tačno rešenje i za pogrešno rešenje sa najvećim skorom od svih pogrešnih rešenja je najveća moguća).

Na slici 2.6 grafički je predstavljen pojam margine u strukturnom kontekstu. Svaka vertikalna linija predstavlja y-osu na kojoj su plavim crtama obeležene vrednosti funkcije skora $F(\boldsymbol{x_i}, \boldsymbol{y})$ za ulazni vektor $\boldsymbol{x_i}$ i razne izlazne vektore $\boldsymbol{y} \in \mathcal{Y}$. Crvenim crtama obeležena je vrednost funkcije skora $F(\boldsymbol{x_i}, \boldsymbol{y_i})$ za ulazni vektor $\boldsymbol{x_i}$ i tačan izlazni vektor $\boldsymbol{y_i}$. U prikazanoj ilustraciji, svaka funkcija skora F za svaki trening primer $(\boldsymbol{x_i}, \boldsymbol{y_i})$ je maksimalna, odnosno veća od $F(\boldsymbol{x_i}, \boldsymbol{y})$ za bilo koje drugo $\boldsymbol{y}$, što znači da je uslov margine za svaki trening primer $(\boldsymbol{x_i}, \boldsymbol{y_i})$ ispunjen. Ovaj slučaj je analogan slučaju linearno separabilnih podataka kod SVM, što se i u strukturnim primenama retko sreće. Zbog toga i u strukturnom kontekstu postoji definisana meka margina, o čemu će biti reči u poglavlju 2.3.

2.3 Formulacija optimizacionih problema za SSVM

Kao što je bilo reči u poglavlju 2.2, potrebno je trenirati koeficijente $\boldsymbol{w}$ tako da funkcija $f(\boldsymbol{x};\boldsymbol{w}) = \text{argmax}_{\boldsymbol{y}\in\mathcal{Y}}(\langle\Psi(\boldsymbol{x},\boldsymbol{y}),\boldsymbol{w}\rangle)$ ima minimalan regularizovani empirijski rizik na trening skupu $S = \{(\boldsymbol{x_1},\boldsymbol{y_1}),\ldots,(\boldsymbol{x_n},\boldsymbol{y_n})\}$. Tsochantaridis predlaže nekoliko varijanti generalizacije metode podržavajućih vektora na strukturni izlaz [100]:

1. **Slučaj sa čvrstom marginom**
   Prvo je razmatran separabilan slučaj, odnosno situacija kada je moguće naći funkciju f takvu da je empirijski rizik jednak nuli. U ovoj situaciji vrednost funkcije gubitka za svaki trening primer je jednaka nuli, što se može zapisati preko skupa nelinearnih uslova:

$$\forall i \in \{1,\ldots,n\} : \max_{\boldsymbol{y}\in\mathcal{Y}\setminus\boldsymbol{y_i}} \{\langle\Psi(\boldsymbol{x_i},\boldsymbol{y}),\boldsymbol{w}\rangle\} \leq \langle\Psi(\boldsymbol{x_i},\boldsymbol{y_i}),\boldsymbol{w}\rangle \tag{2.6}$$

   Ove nejednakosti važe za bilo koji izbor funkcije gubitka, s obzirom na pretpostavke 1 i 2.

   Uslovi (2.6) tvrde da razlika $\langle\Psi(\boldsymbol{x_i},\boldsymbol{y_i}),\boldsymbol{w}\rangle - \max_{\boldsymbol{y}\in\mathcal{Y}\setminus\boldsymbol{y_i}}\{\langle\Psi(\boldsymbol{x_i},\boldsymbol{y}),\boldsymbol{w}\rangle\}$ mora biti veća ili jednaka od nule za svaki trening primer i. Intuitivno, želimo da skor za tačan izlaz $\boldsymbol{y_i}$ za trening primer $\boldsymbol{x_i}$ bude veći (ili jednak) od maksimalnog skora od svih ostalih mogućih izlaza $\boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}$. Ova razlika predstavlja razdvajajuću marginu, analognu slučaju klasične metode podržavajućih vektora.
   Svaki uslov (2.6) može se zameniti sa:

$$\forall i \in \{1,\ldots,n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i} : \langle\Psi(\boldsymbol{x_i},\boldsymbol{y_i}) - \Psi(\boldsymbol{x_i},\boldsymbol{y}),\boldsymbol{w}\rangle \geq 0 \tag{2.7}$$

   Na ovaj način, od n nelinearnih uslova dobijamo $n|\mathcal{Y}| - n$ linearnih uslova.
   Ako su podaci separabilni ($\gamma_i > 0\ \forall i$), tada skaliranjem vektora $\boldsymbol{w}$ možemo dobiti proizvoljno veliku marginu [75]. Ovaj problem možemo rešiti ili fiksiranjem margine (npr. $\min_i \gamma_i \geq 1$) ili fiksiranjem norme vektora (npr. $||\boldsymbol{w}||$=1) čime dobijamo sledeće ekvivalentne optimizacione probleme:

Optimizacioni problem 1.

$$SVM_0^1: \max_{\gamma, ||\boldsymbol{w}||=1} \gamma$$
$$\forall i \in \{1, \ldots, n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i} : \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \geq \gamma \quad (2.8)$$

Ovaj problem može biti ekvivalentno zapisan kao problem konveksne kvadratne optimizacije na sledeći način:

Optimizacioni problem 2.

$$SVM_0^2: \min_{\boldsymbol{w}} \frac{||\boldsymbol{w}||^2}{2}$$
$$\forall i \in \{1, \ldots, n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i} : \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \geq 1 \quad (2.9)$$

2. **Slučaj sa mekom marginom**

Prethodno opisani slučaj se retko pojavljuje u praksi i mnogo češće je neophodno dozvoliti grešku nad podacima u trening skupu, što kao i kod klasične metode podržavajućih vektora postižemo uvođenjem labavih (eng. *slack*) promenljivih. Da bismo kaznili slučaj kada trening primer nije pravilno klasifikovan, u funkciju koju minimizujemo dodajemo član koji je linearan po labavim promenljivim, čime dobijamo sledeći optimizacioni problem (slučaj sa tzv. mekom marginom):

Optimizacioni problem 3.

$$SVM_1^1: \min_{\boldsymbol{w}, \xi_i} \frac{||\boldsymbol{w}||^2}{2} + \frac{C}{n} \sum_{i=1}^{n} \xi_i$$
$$\forall i \in \{1, \ldots, n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i} : \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \geq 1 - \xi_i, \xi_i \geq 0 \quad (2.10)$$

Još jednu formulaciju problema sa mekom marginom možemo dobiti uvođenjem sume kvadrata labavih promenljivih u funkciju koju minimizujemo umesto linearne sume:

Optimizacioni problem 4.

$$SVM_1^2: \min_{\boldsymbol{w},\xi_i} \frac{||\boldsymbol{w}||^2}{2} + \frac{C}{2n}\sum_{i=1}^{n}\xi_i^2$$
$$\forall i \in \{1,\ldots,n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}:$$
$$\langle \Psi(\boldsymbol{x_i},\boldsymbol{y_i}) - \Psi(\boldsymbol{x_i},\boldsymbol{y}), \boldsymbol{w}\rangle \geq 1 - \xi_i, \xi_i \geq 0 \tag{2.11}$$

U oba slučaja, $C > 0$ je konstanta koja kontroliše nagodbu između minimizacije greške nad trening skupom i maksimizacije margine.

3. **Uvođenje funkcije gubitka u formulaciju problema - reskaliranje labavih promenljivih i reskaliranje margine**
Slučaj sa mekom marginom podrazumeva da se model trenira tako da može da napravi greške na pojedinim trening primerima, što je regulisano labavim promenljivim. Intuitivno, ako model pogrešno klasifikuje trening primer $(\boldsymbol{x_i},\boldsymbol{y_i})$ dodelivši mu klasu $\boldsymbol{y}$ koja se mnogo razlikuje od klase $\boldsymbol{y_i}$, kazna treba da bude veća nego da se dodeljena klasa malo razlikuje od stvarne. U literaturi su predložena dva načina za uvođenje funkcije gubitka u formulaciju problema i pokazano je da njihovo rešavanje dovodi do minimizacije regularizovanog empirijskog rizika [100].

Ukoliko u uslovima (2.10) fiksiranu dužinu margine zamenimo funkcijom gubitka, dobijamo formulaciju sa reskaliranjem margine.

Optimizacioni problem 5.

$$SVM_2^{\Delta m}: \min_{\boldsymbol{w},\xi_i} \frac{||\boldsymbol{w}||^2}{2} + \frac{C}{n}\sum_{i=1}^{n}\xi_i$$
$$\forall i \in \{1,\ldots,n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}:$$
$$\langle \Psi(\boldsymbol{x_i},\boldsymbol{y_i}) - \Psi(\boldsymbol{x_i},\boldsymbol{y}), \boldsymbol{w}\rangle \geq \Delta(\boldsymbol{y},\boldsymbol{y_i}) - \xi_i, \xi_i \geq 0 \tag{2.12}$$

Ako transformišemo uslove (2.12) na sledeći način:

$$\forall i \in \{1,\ldots,n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}:$$
$$\langle \Psi(\boldsymbol{x_i},\boldsymbol{y_i}), \boldsymbol{w}\rangle - \langle \Psi(\boldsymbol{x_i},\boldsymbol{y}), \boldsymbol{w}\rangle \geq \Delta(\boldsymbol{y},\boldsymbol{y_i}) - \xi_i,$$

možemo uočiti da oni obezbeđuju da za dati trening primer $(\boldsymbol{x_i}, \boldsymbol{y_i})$ razlika skora kompatibilnosti za tačno $\boldsymbol{y_i}$ i bilo koje drugo $\boldsymbol{y}$ bude široka bar koliko su $\boldsymbol{y_i}$ i $\boldsymbol{y}$ različiti, pri čemu labava promenljiva ξ_i dopušta da to ne mora važiti za svaki trening primer.

Radi potpunijeg razumevanja, uslove (2.12) tranformišemo ponovo:

$$\forall i \in \{1, \ldots, n\} : \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle - \max_{\boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}} \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \geq \Delta(\boldsymbol{y}, \boldsymbol{y_i}) - \xi_i,$$

Uvedimo oznaku $\boldsymbol{y_i'} = \arg\max_{\boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i}} \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle$. Daljom transformacijom dobijamo sledeci ekvivalentan zapis uslova (2.12):

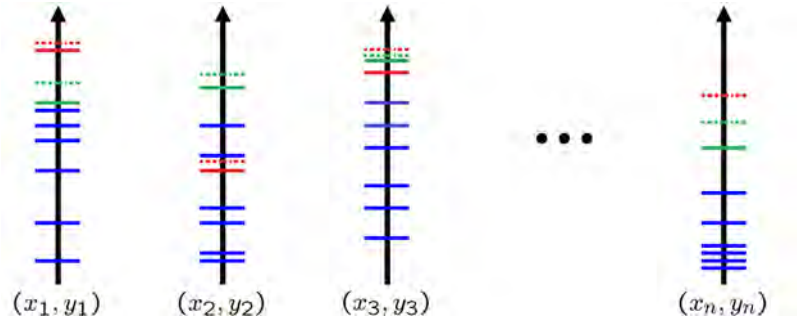
$$\forall i \in \{1, \ldots, n\} : \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle + \xi_i \geq \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i'}), \boldsymbol{w} \rangle + \Delta(\boldsymbol{y}, \boldsymbol{y_i}) \quad (2.13)$$

Na slici 2.7 prikazani su skorovi kompatibilnosti za svaki trening primer $(\boldsymbol{x_i}, \boldsymbol{y_i})$ i svako $\boldsymbol{y} \in \mathcal{Y}$ pri čemu su skorovi za $(\boldsymbol{x_i}, \boldsymbol{y_i})$ i $(\boldsymbol{x_i}, \boldsymbol{y_i'})$ označeni redom crvenom i zelenom linijom, a njihove "popravke" koje odgovaraju redom levoj i desnoj strani nejednakosti (2.13) označene su crvenom isprekidanom i zelenom isprekidanom linijom. Možemo uočiti da trening primer $(\boldsymbol{x_1}, \boldsymbol{y_1})$ ispunjava uslov uslov (2.12) (crvena je iznad zelene i bez popravki), trening primer $(\boldsymbol{x_2}, \boldsymbol{y_2})$ ne ispunjava uslov (2.12) (crvena je ispod zelene i bez popravki i sa njima), trening primer $(\boldsymbol{x_3}, \boldsymbol{y_3})$ ispunjava uslov (2.12) (crvena je ispod zelene, ali je nakon popravki crvena isprekidana iznad zelene isprekidane) i trening primer $(\boldsymbol{x_n}, \boldsymbol{y_n})$ takođe ispunjava uslov (2.12) (crvena je iznad zelene, i crvena puna se poklapa sa crvenom isprekidanom što znači da je $\xi_n = 0$).

Ukoliko u uslovima (2.10) labave promenljive pomnožimo recipročnom vrednošću funkcije gubitka, dobijamo takozvanu formulaciju sa reskaliranjem labavih promenljivih.

Optimizacioni problem 6.

$$SVM_2^{\Delta s}: \min_{\boldsymbol{w}, \xi_i} \frac{||\boldsymbol{w}||^2}{2} + \frac{C}{n} \sum_{i=1}^{n} \xi_i$$
$$\forall i \in \{1, \ldots, n\}, \forall \boldsymbol{y} \in \mathcal{Y} \setminus \boldsymbol{y_i} :$$
$$\langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \geq 1 - \frac{\xi_i}{\Delta(\boldsymbol{y}, \boldsymbol{y_i})}, \xi_i \geq 0 \quad (2.14)$$



Slika 2.7: *Margina u strukturnom kontekstu u formulaciji sa reskaliranjem margine.*

Intuitivno, uloga reskaliranja je da za one izlazne vektore $\boldsymbol{y}$ koji su sličniji tačnom izlaznom vektoru $\boldsymbol{y_i}$ (i za koje je $\Delta(\boldsymbol{y}, \boldsymbol{y_i})$ malo) bude omogućeno veće odstupanje od uslova (2.14) nego za izlazne vektore koji su različitiji od $\boldsymbol{y_i}$.

Zapišimo uslove (2.12) i (2.14) na sledeći način:

$$\begin{aligned} \textit{margin:}\ \xi_i &\geq \Delta(\boldsymbol{y}, \boldsymbol{y_i}) - \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle + \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle \\ \textit{slack:}\ \xi_i &\geq \Delta(\boldsymbol{y}, \boldsymbol{y_i})[1 - \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle + \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle] \end{aligned}$$

U formulaciji sa reskaliranjem labavih promenljivih, ukoliko je skor za tačan izlaz $\langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle$ veći od skora za bilo koji drugi izlaz $\langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle$ bar za 1 (ili bilo koju fiksiranu vrednost), tada labava promenljiva ξ_i mora biti jednaka nuli zbog uslova $\xi_i \geq 0$. To znači da je, da bi predviđanje za trening primer bilo tačno, dovoljno da razlika skorova bude veća od konstante, koja ne zavisi od same funkcije gubitka. Sa druge strane, kod formulacije sa reskaliranjem margine, da bi ξ bila jednaka nuli, razlika skora za tačan izlaz $\boldsymbol{y_i}$ i bilo koji drugi izlaz $\boldsymbol{y}$ mora biti bar $\Delta(\boldsymbol{y_i}, \boldsymbol{y})$. Ova zavisnost čini uslove u prvoj formulaciji strožim u odnosu na uslove u drugoj formulaciji.

Formulacija sa reskaliranjem labavih promenljivih pokazala je bolje performanse kod problema predviđanja sa strukturnim izlazom koji sadrže šum, s obzirom da, za razliku od formulacije sa reskaliranjem margine, izlazi $\boldsymbol{y}$ sa skorom većim od skora za tačan izlaz $\boldsymbol{y_i}$ imaju mali uticaj na labavu promenljivu ξ_i. Sa druge strane, reskaliranje labavih promenljivih obično dovodi do računski zahtevnije faze treninga [96]. U ovom istraživanju fokus je bio na reskaliranju margine upravo zbog računskih prednosti.

4. **Uvođenje jedne labave promenljive umesto n (1-*slack* formulacija)** Primetimo najpre da s obzirom da je $\Delta(\boldsymbol{y_i}, \boldsymbol{y_i}) = 0$ za sve $\boldsymbol{y_i}$ u trening skupu, optimizacioni problem (2.12) možemo ekvivalentno zapisati na sledeći način:

   Optimizacioni problem 7.

$$SVM_{2ref}^{\Delta m}: \min_{\boldsymbol{w},\xi_i} \frac{||\boldsymbol{w}||^2}{2} + \frac{C}{n}\sum_{i=1}^{n}\xi_i$$
$$\forall i \in \{1,\ldots,n\}, \forall \boldsymbol{y} \in \mathcal{Y} : \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w}\rangle \geq \Delta(\boldsymbol{y}, \boldsymbol{y_i}) - \xi_i \tag{2.15}$$

   U odnosu na prethodnu formulaciju, u novoj formulaciji kod uslova razmatramo sve $\boldsymbol{y} \in \mathcal{Y}$ uključujući i $\boldsymbol{y} = \boldsymbol{y_i}$ i isključujemo uslov $\xi_i \geq 0$. Nova formulacija problema je jednaka staroj baš zbog toga što za $\boldsymbol{y} = \boldsymbol{y_i}$ dobijamo uslov

$$\langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w}\rangle \geq \Delta(\boldsymbol{y_i}, \boldsymbol{y_i}) - \xi_i$$

   koji je zbog pretpostavke $\Delta(\boldsymbol{y_i}, \boldsymbol{y_i}) > 0$ ekvivalentan uslovu $\xi_i \geq 0$.

   Dalje, primetimo da optimizacioni problem (2.15) možemo ekvivalentno zapisati tako da koristi samo jednu labavu promenljivu umesto n, po jednu za svaki trening primer.

   Optimizacioni problem 8.

$$SVM_{1slack} : \min_{\boldsymbol{w},\xi} \frac{||\boldsymbol{w}||^2}{2} + C\xi$$
$$\forall(\bar{\boldsymbol{y}}_{\boldsymbol{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n :$$
$$\frac{1}{n}\sum_{i=1}^{n}\langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}) - \Psi(\boldsymbol{x_i}, \bar{\boldsymbol{y}}_{\boldsymbol{i}}), \boldsymbol{w}\rangle \geq \frac{1}{n}\sum_{i=1}^{n}\Delta(\boldsymbol{y_i}, \bar{\boldsymbol{y}}_{\boldsymbol{i}}) - \xi \tag{2.16}$$

   Ekvivalencija optimizacionih problema (2.15) i (2.16) se lako pokazuje [69]. Za dato $\boldsymbol{w}$, svako ξ_i u (2.15) se optimizuje individualno. Najmanje ξ_i za koje važe uslovi datog optimizacionog problema je

$$\xi_i = \max_{\boldsymbol{y}\in\mathcal{Y}}[\Delta(\boldsymbol{y}, \boldsymbol{y_i}) - \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w}\rangle + \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w}\rangle]$$

Sa druge strane, najmanje ξ koje ispunjava uslove optimizacionog problema (2.16) je

$$\xi = \max_{(\bar{\boldsymbol{y}}_1,\ldots,\bar{\boldsymbol{y}}_n)\in\mathcal{Y}^n} \left[\frac{1}{n}\sum_{i=1}^{n} \Delta(\boldsymbol{y_i},\bar{\boldsymbol{y}}_i) - \frac{1}{n}\sum_{i=1}^{n} \langle \Psi(\boldsymbol{x_i},\boldsymbol{y_i}) - \Psi(\boldsymbol{x_i},\bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle \right]$$

Pošto se funkcija koju maksimizujemo može linearno dekomponovati po $\bar{\boldsymbol{y}}_i$, za svako dato $\boldsymbol{w}$, svako $\bar{\boldsymbol{y}}_i$ možemo optimizovati nezavisno.

$$\xi = \frac{1}{n}\sum_{i=1}^{n} \max_{\bar{\boldsymbol{y}}_i\in\mathcal{Y}} [\Delta(\boldsymbol{y_i},\bar{\boldsymbol{y}}_i) - \langle \Psi(\boldsymbol{x_i},\boldsymbol{y_i}) - \Psi(\boldsymbol{x_i},\bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle] = \frac{1}{n}\sum_{i=1}^{n} \xi_i$$

Stoga, funkcije cilja za oba optimizaciona problema su jednake za proizvoljni vektor parametara $\boldsymbol{w}$ i date najmanje vrednosti promenljivih ξ i ξ_i.

2.4 Algoritam odsecajućih ravni za treniranje SVM sa strukturnim izlazom

U poglavlju 2.3 predstaviljena je SSVM metoda, generalizacija SVM metode za strukturni izlaz. Optimizacioni problem koji ona definiše je problem kvadratne optimizacije sa linearnim uslovima i može se rešavati različitim tehnikama konveksne optimizacije. Ovde će biti opisano rešavanje ovog problema metodom odsecajućih ravni, koja je ugrađena u SVMstruct[1] implementaciju, jednu od popularnih i javno dostupnih implementacija SSVM metode.

Optimizacioni problem (2.16) predstavlja minimizaciju kvadratne funkcije sa $|\mathcal{Y}|^n$ linearnih uslova, gde je n broj primera u trening skupu a $|\mathcal{Y}|$ broj svih mogućih izlaznih vektora. Kod velikog broja problema strukturne klasifikacije, postoji eksponencijalno mnogo klasa kao mogućih izlaza za dati skup ulaznih atributa, na primer kod problema gde klasa predstavlja podgraf zadatog grafa sa velikim brojem čvorova. Sa tako velikim brojem uslova, standardne metode za rešavanje problema konveksne optimizacije ne mogu se efikasno primenjivati kao što je to slučaj kod binarnih SVM. Jedan način

[1] http://www.cs.cornell.edu/people/tj/svm_light/svm_struct.html

za efikasno treniranje SSVM metode je pomoću optimizacione tehnike odsecajućih ravni (eng. *cutting-plane*) [81]. Složenost ovog algoritma za treniranje je linearna po broju trening primera i ne zavisi od kardinalnosti skupa $\mathcal{Y}$ [69].

Algoritam odsecajućih ravni zasnovan je na činjenici da je pri minimizaciji funkcije sa velikim brojem uslova samo mali broj uslova aktivan, odnosno ima uticaj na rezultat minimizacije. Na slici 2.8 prikazan je primer minimizacije funkcije sa ograničenjima (slika levo) gde su za određivanje tačke minimuma od značaja samo dva uslova (slika desno). Upravo aktivni uslovi smanjuju prostor pretrage za određivanje minimuma funkcije cilja, odakle i potiče naziv "odsecajuće ravni".

Slika 2.8: *Neaktivni(levo) i aktivni(desno) uslovi pri minimizaciji sa ograničenjima.*

Osnovna ideja algoritma odsecajućih ravni je da se umesto rešavanja optimizacionog problema sa $|\mathcal{Y}|^n$ uslova minimizuje ista funkcija cilja sa mnogo manjim brojem pažljivo odabranih uslova čije će se rešenje od rešenja originalnog problema razlikovati najviše za proizvoljno izabranu konstantu ϵ. Preciznije, u optimizacionom problemu (2.16) umesto uslova po svim n-torkama $(\bar{\boldsymbol{y}}_\mathbf{1}, \ldots, \bar{\boldsymbol{y}}_\boldsymbol{n}) \in \mathcal{Y}^n$ uzimaćemo u obzir samo neke n-torke $(\bar{\boldsymbol{y}}_\mathbf{1}, \ldots, \bar{\boldsymbol{y}}_\boldsymbol{n}) \in W \subset \mathcal{Y}^n$. Inicijalno, skup W je prazan. U svakoj iteraciji algoritma dodavaćemo po jednu n-torku u skup W i rešavati optimizacioni problem za sa smanjenim brojem uslova. Dodata n-torka predstavlja jedan aktivan uslov optimizacionog problema (2.16).

Skup W će biti sačinjen od n-torki skupa $\mathcal{Y}^n$ odabranih na sledeći način: u svakoj iteraciji, sa tekućim rešenjem optimizacionog problema $(\boldsymbol{w}, \xi)$, za svaki trening primer $(\boldsymbol{x_i}, \boldsymbol{y_i})$ tražimo ono $\bar{\boldsymbol{y}}_\boldsymbol{i}$ koje najviše krši uslove optimizacionog

problema (2.16). Podsetimo se tih uslova:

$$\forall(\bar{\boldsymbol{y}}_{\mathbf{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n :$$
$$\frac{1}{n}\sum_{i=1}^{n}(\langle\Psi(\boldsymbol{x}_i, \boldsymbol{y}_i), \boldsymbol{w}\rangle - \langle\Psi(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle) \geq \frac{1}{n}\sum_{i=1}^{n}\Delta(\boldsymbol{y}_i, \bar{\boldsymbol{y}}_i) - \xi$$

Zapišimo ih malo drugačije:

$$\forall(\bar{\boldsymbol{y}}_{\mathbf{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n :$$
$$\xi \geq \frac{1}{n}\sum_{i=1}^{n}\Delta(\boldsymbol{y}_i, \bar{\boldsymbol{y}}_i) - \frac{1}{n}\sum_{i=1}^{n}(\langle\Psi(\boldsymbol{x}_i, \boldsymbol{y}_i), \boldsymbol{w}\rangle - \langle\Psi(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle)$$

$$\forall(\bar{\boldsymbol{y}}_{\mathbf{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n :$$
$$\xi \geq \frac{1}{n}\sum_{i=1}^{n}\Delta(\boldsymbol{y}_i, \bar{\boldsymbol{y}}_i) - \frac{1}{n}\sum_{i=1}^{n}\langle\Psi(\boldsymbol{x}_i, \boldsymbol{y}_i), \boldsymbol{w}\rangle + \frac{1}{n}\sum_{i=1}^{n}\langle\Psi(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle$$

$$\forall(\bar{\boldsymbol{y}}_{\mathbf{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n :$$
$$\xi \geq \frac{1}{n}\sum_{i=1}^{n}(\Delta(\boldsymbol{y}_i, \bar{\boldsymbol{y}}_i) + \langle\Psi(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle) - \frac{1}{n}\sum_{i=1}^{n}\langle\Psi(\boldsymbol{x}_i, \boldsymbol{y}_i), \boldsymbol{w}\rangle$$

Ukoliko je za bilo koji trening primer $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ ovaj uslov prekršen, to možemo zapisati ovako:

$$\exists(\bar{\boldsymbol{y}}_{\mathbf{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n :$$
$$\xi < \frac{1}{n}\sum_{i=1}^{n}(\Delta(\boldsymbol{y}_i, \bar{\boldsymbol{y}}_i) + \langle\Psi(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle) - \frac{1}{n}\sum_{i=1}^{n}\langle\Psi(\boldsymbol{x}_i, \boldsymbol{y}_i), \boldsymbol{w}\rangle \quad (2.17)$$

Primetimo da u razlici sa desne strane nejednakosti (2.17) umanjilac ne zavisi od $\bar{\boldsymbol{y}}_i$. Sa druge strane, umanjenik zavisi od $\bar{\boldsymbol{y}}_i$ i što veći, to će i cela razlika biti veća. Dalje, razlika će biti maksimalna ukoliko je umanjenik maksimalan, što možemo postići ukoliko za $(\bar{\boldsymbol{y}}_{\mathbf{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}})$ odaberemo baš one izlazne vektore $\bar{\boldsymbol{y}}_i$ koji maksimizuju izraz

$$\Delta(\boldsymbol{y}_i, \bar{\boldsymbol{y}}_i) + \langle\Psi(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i), \boldsymbol{w}\rangle \quad (2.18)$$

Zbog toga ima smisla u tekućoj iteraciji, sa trenutnim vrednostima $(\boldsymbol{w}, \xi)$, odrediti za svaki trening primer $\boldsymbol{x_i}$ maksimalnu vrednost izraza 2.18 po svim $\bar{\boldsymbol{y}}_{\boldsymbol{i}} \in \mathcal{Y}, \forall i \in \{1, \ldots, n\}$. Dobijenu n-torku $(\bar{\boldsymbol{y}}_{\boldsymbol{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}}) \in \mathcal{Y}^n$ dodajemo u skup W aktivnih uslova. Na kraju svake iteracije proverava se da li tekuća n-torka $(\bar{\boldsymbol{y}}_{\boldsymbol{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}})$ krši uslove optimizacionog problema za više od unapred odabrane konstante ϵ. Ukoliko krši, to je znak da tekuća rešenja $(\boldsymbol{w}, \xi)$ nisu dovoljno dobra i postupak optimizacije se nastavlja. U suprotnom, algoritam se završava i vraća tekuća rešenja $(\boldsymbol{w}, \xi)$ kao konačno rešenje optimizacionog problema. Algoritam 1 prikazuje opisani postupak u pseudokodu.

Algoritam 1 Algoritam odsecajućih ravni za treniranje SSVM metode

1: Input: $(\boldsymbol{x_1}, \boldsymbol{y_1}), \ldots, (\boldsymbol{x_n}, \boldsymbol{y_n}), C, \epsilon$
2: $W = \emptyset$
3: **repeat**
4: $(\boldsymbol{w}, \xi) \leftarrow \arg\min_{\boldsymbol{w}, \xi \geq 0} \frac{||\boldsymbol{w}||}{2} + C\xi$ tako da

$$\forall (\hat{\boldsymbol{y}}_{\boldsymbol{1}}, \ldots, \hat{\boldsymbol{y}}_{\boldsymbol{n}}) \in W :$$
$$\xi \geq \frac{1}{n} \sum_{i=1}^{n} (\Delta(\boldsymbol{y_i}, \hat{\boldsymbol{y}}_{\boldsymbol{i}}) + \langle \Psi(\boldsymbol{x_i}, \hat{\boldsymbol{y}}_{\boldsymbol{i}}), \boldsymbol{w} \rangle) - \frac{1}{n} \sum_{i=1}^{n} \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle$$

5: **for** $i = 1, \ldots n$ **do**
6: $\bar{\boldsymbol{y}}_{\boldsymbol{i}} \leftarrow \arg\max_{\boldsymbol{y} \in \mathcal{Y}} (\Delta(\boldsymbol{y_i}, \boldsymbol{y}) + \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y}), \boldsymbol{w} \rangle)$
7: $W \leftarrow W \cup \{(\bar{\boldsymbol{y}}_{\boldsymbol{1}}, \ldots, \bar{\boldsymbol{y}}_{\boldsymbol{n}})\}$
8: **until** $\xi + \epsilon \geq \frac{1}{n} \sum_{i=1}^{n} (\Delta(\boldsymbol{y_i}, \bar{\boldsymbol{y}}_{\boldsymbol{i}}) + \langle \Psi(\boldsymbol{x_i}, \bar{\boldsymbol{y}}_{\boldsymbol{i}}), \boldsymbol{w} \rangle) - \frac{1}{n} \sum_{i=1}^{n} \langle \Psi(\boldsymbol{x_i}, \boldsymbol{y_i}), \boldsymbol{w} \rangle$
9: Output: $(\boldsymbol{w}, \boldsymbol{\xi})$

Joachims u [69] pokazuje da postupak konvergira, odnosno da je ukupan broj iteracija ograničen i ne zavisi ni od ukupnog broja mogućih klasa $|\mathcal{Y}|$ ni od broja trening primera n u slučaju da je funkcija gubitka Δ ograničena odozgo, odnosno $\forall y^* \; \exists \max_{y \in \mathcal{Y}} \Delta(y, y^*)$. U istom radu pokazano je da se optimizacioni problem u koraku 4 može rešiti u $\mathcal{O}(n)$ koraka, što čini ukupno vreme izvršavanja linearnim po broju trening primera.

2.5 Primeri primena SSVM

Prilagođavanje SSVM metode konkretnom problemu zahteva sledeće korake:

1. Izbor funkcije Ψ koja će predstaviti vektor ulaznih atributa i izlazni vektor u jednom
2. Izbor funkcije Δ koja će meriti različitost dva izlazna vektora
3. Definisanje algoritma za rešavanje optimizacionog problema u koraku 6 algoritma za treniranje:
$$\bar{\boldsymbol{y}}_{\boldsymbol{i}} = \arg\max_{\boldsymbol{y}\in\mathcal{Y}}(\Delta(\boldsymbol{y}_{\boldsymbol{i}}, \boldsymbol{y}) + \langle\Psi(\boldsymbol{x}_{\boldsymbol{i}}, \boldsymbol{y}), \boldsymbol{w}\rangle)$$
4. Definisanje algoritma za rešavanje optimizacionog problema prilikom predviđanja klase $\boldsymbol{y'}$ za nepoznati vektor ulaznih atributa $\boldsymbol{x'}$:
$$\boldsymbol{y'} = \arg\max_{\boldsymbol{y}\in\mathcal{Y}}(\langle\Psi(\boldsymbol{x'}, \boldsymbol{y}), \boldsymbol{w}\rangle)$$

Kod većine primena SSVM, algoritmi 3 i 4 su isti mada ne moraju biti. Nekoliko primera primena SSVM dato je u [69] i ovde će biti predstavljen njihov kratak pregled radi ilustracije koraka 1-4:

- **Višeklasna klasifikacija**
  ulaz $\mathcal{X} = \mathbb{R}^n$
  izlaz $\mathcal{Y} = \{1, \ldots, k\}$
  funkcija gubitka $\Delta(y, \bar{y}) = 1[y \neq \bar{y}] = \begin{cases} 0, & \text{ako } y == \bar{y} \\ 1, & \text{inače} \end{cases}$
  funkcija Ψ
  $$\Psi_{mult}(x, y) = \begin{pmatrix} 0 \\ \ldots \\ 0 \\ x \\ 0 \\ \ldots \\ 0 \end{pmatrix}$$
  Algoritmi Optimizacija u problemima 3 i 4 se rešava iterativno, eksplicitnom enumeracijom klasa.

- **Određivanje vrste reči u rečenici (*Part-of-speech (POS) tagging*)**
  ulaz $x = (x_1, \ldots, x_l)$ rečenica, po jedan atribut za svaku reč
  izlaz $y = (y_1, \ldots, y_l)$ vektor vrsta reči, po jedna za svaku reč
  funkcija gubitka broj pogrešno klasifikovanih vrsta reči
  $$\Delta((y_1, \ldots, y_l), (y'_1, \ldots, y'_l)) = \sum_{i=1}^{l} [y_i \neq y'_i]$$

funkcija Ψ kreiran je model izomorfan HMM modelu

$$\Psi_{POS}(x,y) = \begin{pmatrix} \Psi_{mult} \\ [y_i = 1][y_{i-1} = 1] \\ [y_i = 1][y_{i-1} = 2] \\ \dots \\ [y_i = k][y_{i-1} = k] \end{pmatrix}$$

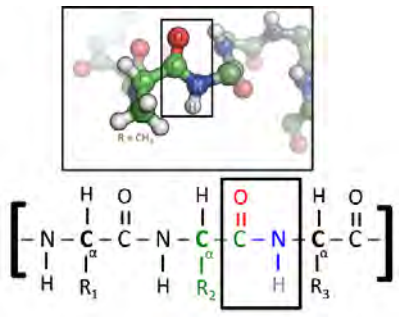
Algoritmi oba optimizaciona problema se rešavaju Viterbijevim algoritmom

- **Predviđanje stabla izvođenja rečenice u konteksno slobodnoj gramatici**
  ulaz $x = (x_1, \dots, x_l)$ rečenica, po jedan atribut za svaku reč
  izlaz y stablo izvođenja sa rečima x_i u listovima
  funkcija gubitka $\Delta(y, \bar{y}) = 1[y \neq \bar{y}]$
  funkcija Ψ po jedna pozicija za svako pravilo izvođenja koje se pojavljuje u trening skupu, vrednost predstavlja broj pojavljivanja određenog pravila u datoj rečenici (slika 2.1)
  Algoritmi oba optimizaciona problema se rešavaju korišćenjem CKY parsera

3. Predviđanje funkcije proteina

3.1 Proteini

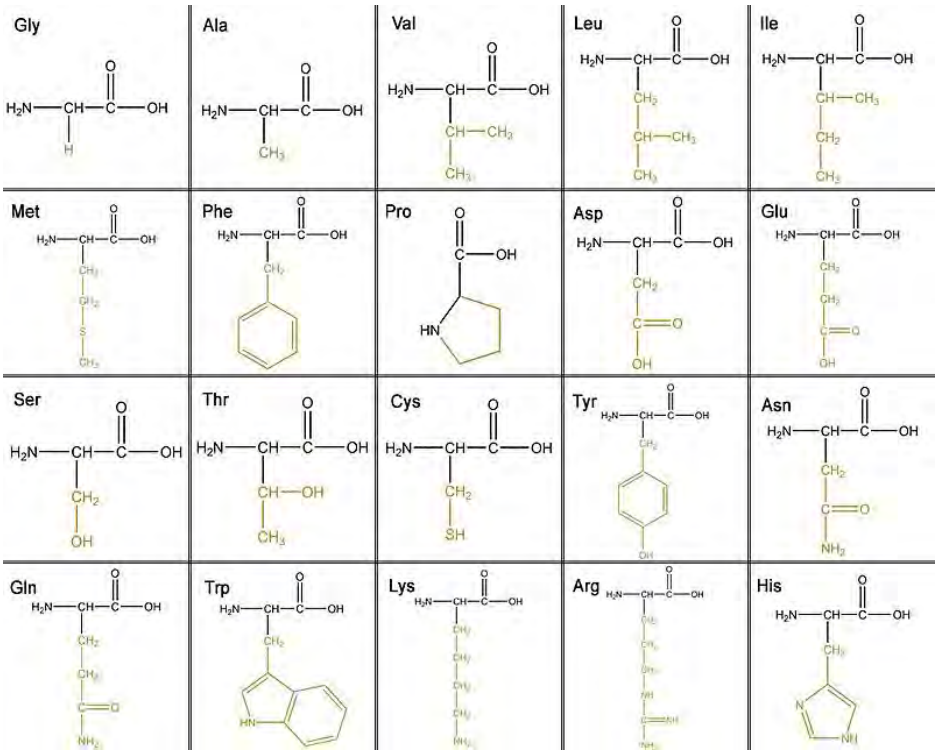
Proteini su biološki makromolekuli zaduženi za različite aktivnosti u našim ćelijama, tkivima, organima i celom organizmu. Sačinjavaju više od 50% suvog dela ćelije i esencijalni su za njenu izgradnju i pravilno funkcionisanje. U važne uloge koje proteini obavljaju spadaju katalitička aktivnost, kontrakcija mišića, strukturna podrška, ubrzavanje i usporavanje hemijskih reakcija, prenošenje signala, odbrana od virusa i bakterija. Na molekulskom nivou, funkcija proteina proizilazi iz interakcija koje oni uspostavljaju sa drugim molekulima. U kakve će interakcije molekuli proteina stupati zavisi prvensteno od njihove strukture, raspodele atoma i njihovih naelektrisanja u prostoru.



Slika 3.1: *Peptidna veza (C=O-NH) između dve susedne aminokiseline formira se povezivanjem karboksilne grupe sa azotom iz amino grupe uz oslobađanje molekula vode.*

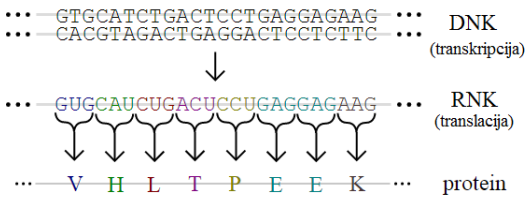
Po hemijskom sastavu proteini predstavljaju linearne lance aminokiselina koje su povezane peptidnim vezama (slika 3.1). Postoji preko 500 ami-

nokiselina dok u sastav proteina ulazi njih 20 i one se nazivaju osnovnim aminokiselinama. Aminokiseline se sastoje iz amino-grupe (-NH_2), karboksilne grupe (-COOH) i bočnog lanca koji je jedinstven za svaku aminokiselinu. Hemijske formule osnovnih aminokiselina prikazane su na slici 3.2. Zahvaljujući svom hemijskom sastavu, aminokiseline mogu da se povezuju u lance različitih dužine, od nekoliko desetina do nekoliko hiljada molekula. Dužina lanca ne utiče na funkciju proteina, npr. insulin se sastoji od svega stotinak aminokiselina dok je dužina titina preko 30 hiljada.



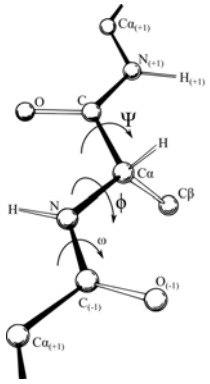
Slika 3.2: *Osnovne aminokiseline.*

Svaki molekul proteina nastaje u ćeliji živog organizma. Redosled aminokiselina u proteinskom lancu određen je redosledom nukleotida u dezoksiribonuleinskoj kiselini (DNK). Informacija o broju, vrsti i redosledu aminokiselina zapisana je na delovima DNK koji se nazivaju geni. Svakoj trojci nukleotida u genu na osnovu genetskog koda jedinstveno je pridružena po jedna aminokiselina. U procesu genske ekspresije, posredstvom glasničke (eng. *messenger*) ribonukleinske kiseline (RNK) i transportne (eng. *transfer*) RNK, enkodirana informacija iz DNK se prevodi u niz aminokiselina u proteinskom lancu. Na slici 3.3 ilustrovan je postupak sinteze proteina.

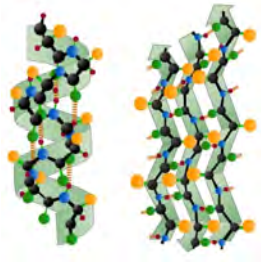


Slika 3.3: *Centralna dogma molekularne biologije.*

Broj, vrsta i redosled aminokiselina jednog proteina predstavlja njegovu primarnu strukturu. Pored primarne, proteinski lanci imaju i različite prostorne strukture. Osnova - kičma proteinskog polipeptidnog lanca (slika 3.4) može biti relativno fiksna i uređena (tri ili više ϕ i ψ uglova imaju iste vrednosti) i ona definiše sekundarnu (prvu trodimenzionalnu) strukturu proteina. Ovakve sekundarne strukture su stabilisane sekundarnim (nekovalentnim) vodoničnim vezama i mogu zauzmati dva osnovna oblika u prostoru: helikoidu (α heliks, π heliks, itd.) ili traku (β struktura) (slika 3.5). Pored navedena dva tipa fiksnih uređenih stuktura koje se u proseku sastoje od po desetak aminokiselina, proteini sadrže i druge oblike sekundarnih struktura kao što su okreti (eng. *turn*), koji se sastoje od 3 do 5 aminokiselina i mogu se posmatrati kao delovi helikoida i petlje (eng. *loop*), slabije uređene fiksne strukture. Pojedini proteini ili manji regioni unutar uređenih proteinskih sekundarnih struktura mogu biti neuređeni tako da tri ili više ϕ i ψ uglova kičme njegovog peptidnog lanca menjaju vrednosti u vremenu. Ovakvi proteini (proteinski regioni) se nazivaju neuređeni proteini (eng. *intrinsically disordered protein – IDP*) i o njima će biti više reči u poglavlju 5.2.

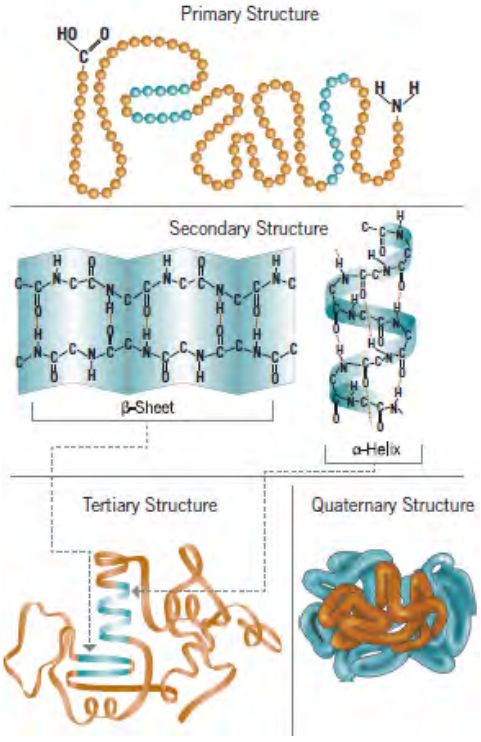


Slika 3.4: *Kičma proteinskog polipeptidnog lanca.*

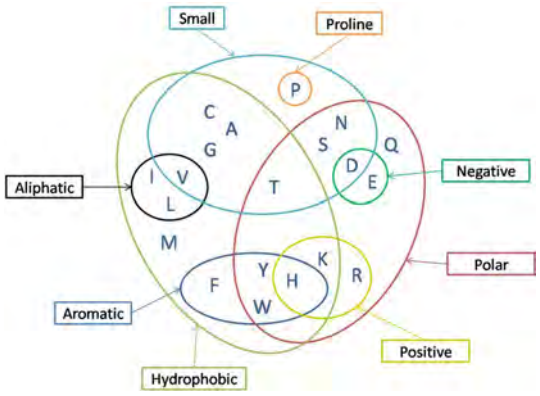


Slika 3.5: *Tipovi sekundarne strukture proteina: α heliks (levo) i β traka (desno).*

Pored primarne i sekundarne, proteini imaju i tercijarnu i kvaternarnu strukturu. Raspored u prostoru svih atoma koji čine protein predstavlja njegovu tercijarnu strukturu. Mnogi proteini su sastavljeni od više polipeptidnih lanaca, takozvanih podjedinica proteina koje čine proteinski kompleks. Kvaternarna struktura proteina podrazumeva trodimenzionalni oblik podjedinica proteinskog kompleksa, odnosno način na koji su podjedinice smeštene unutar kompleksa. Slika 3.6 ilustruje različite nivoe strukture proteina.



Slika 3.6: *Četiri nivoa strukture proteina.*



Slika 3.7: *Aminokiseline klasifikovane prema svojim fizičko-hemijskim karakteristikama.*

Prema svojim fizičko-hemijskim karakteristikama, aminokiseline se mogu grubo podeliti na hidrofobne (nerastvorljive u vodi), koje su više zastupljene unutar globularnih proteina i hidrofilne (rastvorljive u vodi), koje se nalaze na površini globularnih proteina (slika 3.7). Pokazano je da pojedine aminokiseline preferiraju određene forme sekundarne strukture proteina. Na primer, regioni proteina u kojima su prezastupljene aminokiseline Ala, Glu, Leu, Met, a podzastuplene Pro, Gly, Tyr i Ser, imaju tendenciju da formiraju helikoidne strukture. Nasuprot tome, aminokiseline Ala, Arg, Gly, Gln, Ser, Glu, Lys i Pro su više zastupljene u neuređenim proteinima/proteinskim regionima.

Predstavljanje proteina

Da bi automatska obrada proteina bila omogućena, neophodno ih je predstaviti u obliku podataka. Najjednostavniji način za reprezentaciju proteina je u obliku niski karaktera nad azbukom od 20 aminokiselina: Σ={A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y} (tabela 3.1). Na proteine predstavljene u ovom obliku mogu se primenjivati različiti algoritmi za rad sa niskama, na prvom mestu poravnanje niski kao i njegove varijante koje uključuju višestruko poravnanje, poravnanje jednog proteina ili familije proteina u odnosu na bazu proteina ili bazu familija proteina, itd.

Tabela 3.1: *Nazivi aminokiselina sa skraćenicama.*

naziv aminokiseline	naziv amino-kiseline (eng.)	3-slovni kod	1-slovni kod
Alanin	Alanine	Ala	A
Arginin	Arginine	Arg	R
Asparagin	Asparagine	Asn	N
Asparaginska kiselina	Aspartic acid (Aspartate)	Asp	D
Cistein	Cysteine	Cys	C
Glutamin	Glutamine	Gln	Q
Glutaminska kiselina	Glutamic acid (Glutamate)	Glu	E
Glicin	Glycine	Gly	G
Histidin	Histidine	His	H
Izoleucin	Isoleucine	Ile	I
Leucin	Leucine	Leu	L
Lizin	Lysine	Lys	K
Metionin	Methionine	Met	M
Fenilalanin	Phenylalanine	Phe	F
Prolin	Proline	Pro	P
Serin	Serine	Ser	S
Treonin	Threonine	Thr	T
Triptofan	Tryptophan	Trp	W
Tirozin	Tyrosine	Tyr	Y
Valin	Valine	Val	V

Drugi pogodan način za predstavljanje proteina je preko grafova ili mreža. PPI mreže (*protein-protein interaction*) opisuju interakcije između proteina. U takvoj mreži, svaki čvor odgovara jednom proteinu a svaka grana događaju da proteini koje ona povezuje međusobno reaguju da bi obavili neku biološki značajnu funkciju. Mreže genskih regulacija (eng. *gene regulatory networks*) formiraju se na osnovu ekspresivnih šablona između određenih grupa gena. I pojedinačni protein može biti predstavljen kao graf gde bi se u svakom čvoru nalazila po jedna aminokiselina njegove primarne sekvence a dužine grana bi određivale njihovo međusobno rastojanje.

Podaci o proteinima nalaze se u velikim biomedicinskim bazama podataka koje su često javno dostupne. Neke od njih su prikazane u tabeli 3.2.

Tabela 3.2: *Neke biološke baze podataka koje sadrže podatke o proteinima.*

Baza podataka	URL	Opis
UniProtKB	uniprot.org	Proteinske sekvence, funkcija proteina
PFAM	pfam.sanger.ac.uk	Proteinske familije
PDB	pdb.org	Strukture utvrđene eksperimentalno
ModBase	modbase.compbio.ucsf.edu	Strukture utvrđene predviđanjem
I2D	ophid.utoronto.ca	Interakcije između proteina
GEO	ncbi.nlm.nih.gov/geo	Podaci o genskoj ekspresiji
PRIDE	www.ebi.co.uk/pride	Podaci dobijeni masenom sprektometrijom

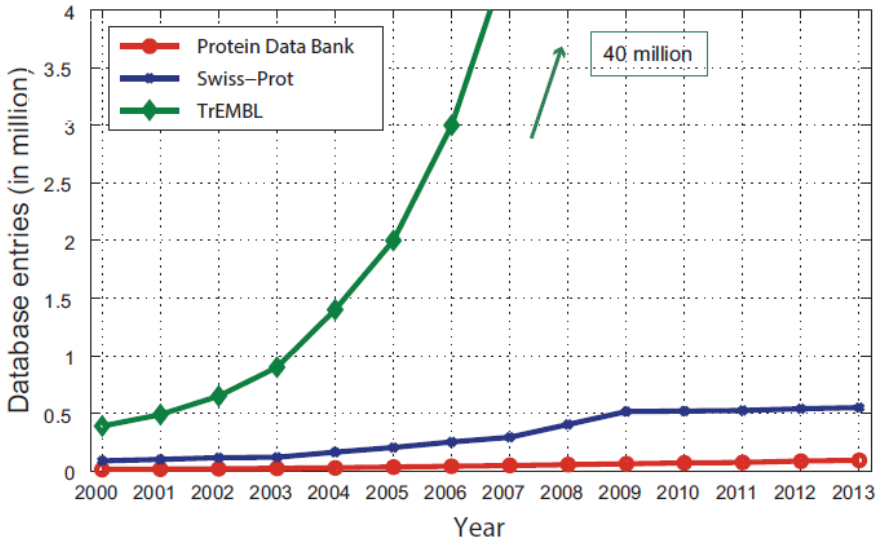
3.2 Funkcija proteina

Proteini obavljaju veoma značajne zadatke u živom svetu. Spektar svih aktivnosti koje jedan protein izvršava sačinjava njegovu funkciju. Međutim, pojam funkcije proteina je teško precizno definisati. Funkcija je kompleksan fenomen koji se može posmatrati na više nivoa: biohemijskom, ćelijskom, razvojnom, fiziološkom. Pomenuti aspekti funkcije proteina su međusobno isprepletani. Na primer, posmatrajmo sve uloge koje obavlja protein kinaza. Iz biohemijskog ugla, uloga kinaze je fosforilacija hidroksilne grupe određenog supstrata. Ovaj aspekt opisuje funkciju nezavisno od toga gde se protein nalazi i za koji ćelijski proces je fosforilacija od značaja. Iz fiziološkog ugla, kinaza učestvuje u prenošenju signala gde ovaj protein fosforizuje druge proteine, i do njegove fosforizacije dolazi u interakciji sa drugim proteinima. Mutacija ovog proteina može dovesti do poremećaja njegove funkcije a time i oboljenja, što predstavlja fenotipski i medicinski aspekt. Zbog toga, kada govorimo o funkciji proteina, neophodno je odrediti aspekt u okviru koga se ona posmatra [15, 27].

Funkcija proteina se, bez obzira na aspekt, određuje eksperimentalnim procedurama u laboratorijama. Utvrđenu funkciju biolozi opisuju prirodnim jezikom. U prirodnom jeziku funkciju možemo veoma precizno opisati. Ovakav opis je bogat i informativan ali nimalo pogodan za bilo kakvu au-

tomatsku obradu jer prirodni jezik kao način izražavanja nije struktuiran niti jednoznačan. Korišćenje sinonima, homonima, kao i izostavljanje značajnih informacija samo su neke karakteristike opisa funkcije proteina na prirodnom jeziku koje mogu zbuniti čak i čitaoce.

Eksperimentalno određivanje funkcije proteina se smatra najpouzdanijim načinom utvrđivanja uloge koju protein obavlja u ćeliji. Međutim, eksperimentalna anotacija je skupa i spora za uvećani priliv novih proteina koji dolaze sa svakim sekvencioniranim genomom. Zbog toga je razvoj metoda za automatsko predviđanje funkcija od velikog značaja. Slika 3.8 pokazuje veliki raskorak koji postoji između broja proteinskih sekvenci čija je funkcija eksperimentalno utvrđena i onih za koje funkcija nije poznata.



Slika 3.8: *Rast bioloških baza podataka u protekloj deceniji. Linija Protein Data Bank pokazuje broj proteinskih struktura koje su eksperimentalno utvrđene po godinama, linija Swiss-Prot daje informacije o broju proteina za koje je eksperimentalno ispitana funkcija, a linija TrEMBL pokazuje broj novih proteina koji su dodati u bazu proteina UniProt i za koje nema informacija ni o strukturi ni o funkciji. Slika je preuzeta iz [92].*

Da bi se omogućila računarska obrada funkcije proteina, javila se potreba za uvođenjem pravila za njeno opisivanje. Neophodnost uvođenja kontrolisanog rečnika i dobro definisanih odnosa između funkcija prvi su prepoznali biohemičari i predložili sistem za nomenklaturu enzima pod nazivom *Enzyme*

Commision Classification (EC)[1]. EC svakom enzimu dodeljuje jedinstveni kod koji se sastoji od četiri prirodna broja, razdvojena tačkom, pri čemu svaki broj respektivno predstavlja finiju klasifikaciju enzima. Na primer, enzim *tripeptidna aminopeptidaza* ima kod EC 3.4.11.4, gde EC 3 označava da ovaj enzim pripada grupi hidrolaza, EC 3.4 da pripada grupi hidrolaza koji deluju na peptidnim vezama, EC 3.4.11 da pripada grupi hidrolaza koji odvajaju amino-terminalnu aminokiselinu iz polipeptida i EC 3.4.11.4 da je taj polipeptid tripeptid. Paralelno sa projektima sekvencioniranja genoma različitih organizama, predlagane su genomske šeme za kategorizaciju proteina za određeni organizam [93, 106], koje su uglavnom pratile postavljeni trend uvođenja kontrolisanog rečnika i kategorija koje idu od opšte ka specifičnoj. Različite šeme za anotaciju funkcije proteina prikazane su u [94].

Sistem za predstavljanje funkcija proteina koji je trenutno dominantan u zajednici za njihovu automatsku predikciju je *Gene Ontology* (GO) [47]. GO se sastoji od tri ontologije u kojima je kontrolisanim rečnikom opisana funkcija proteina sagledana iz tri različita aspekta: molekulskih funkcija (MFO), bioloških procesa (BPO) i ćelijskih komponenti (CCO). Uz svaku funkciju zabeležen je i referenca koja potvrđuje da protein zaista ima tu funkciju i na koji način je ona dobijena (eksperimentalno, računarskom analizom, ...).[2] MFO opisuje elementarne funkcije koje protein obavlja na nivou hemijskih reakcija, kao što je funkcija vezivanja ili razne katalitičke funkcije. U BPO, funkcije su opisane na nivou bioloških procesa koje ta funkcija omogućava ili potpomaže unutar organizma, kao što su različiti metabolički ili regulacioni procesi. CCO ontologija sadrži opis funkcija na nivou lokacije u ćeliji gde se funkcija obavlja i u nazivu može sadržati naziv organele ili nekog drugog dela ćelije. Iako poslednja ontologija ne opisuje konkretne funkcije kao preostale dve, veoma je važna za razumevanje fenomena funkcije proteina jer proteini ne deluju izolovano već unutar živih ćelija. Svaka od GO ontologija sadrži po nekoliko hiljada čvorova što je prikazano na tabeli 3.3.

Sve tri GO ontologije su struktuirane u obliku usmerenog acikličkog grafa gde se u čvorovima nalaze funkcije a grane koje ih povezuju definišu relaciju "is-a". Na taj način, svaki čvor opisuje specifičniju funkciju od svog pretka, u korenu se nalazi najopštija funkcija koja nosi naziv ontologije (MFO, BPO ili CCO) a u listovima najspecifičnije funkcije. Na primer, u MFO ontologiji, funkcija koja opisuje vezivanje hormona tiroidne žlezde se nalazi u listu ontologije, a putanja od nje do korena je prikazana na slici 3.9. Struktura

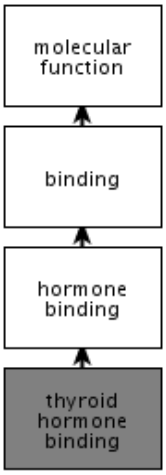
[1] http://www.chem.qmul.ac.uk/iubmb/enzyme/

[2] http://geneontology.org/page/guide-go-evidence-codes

Tabela 3.3: *Broj čvorova u usmerenim acikličkim grafovima GO ontologija.*

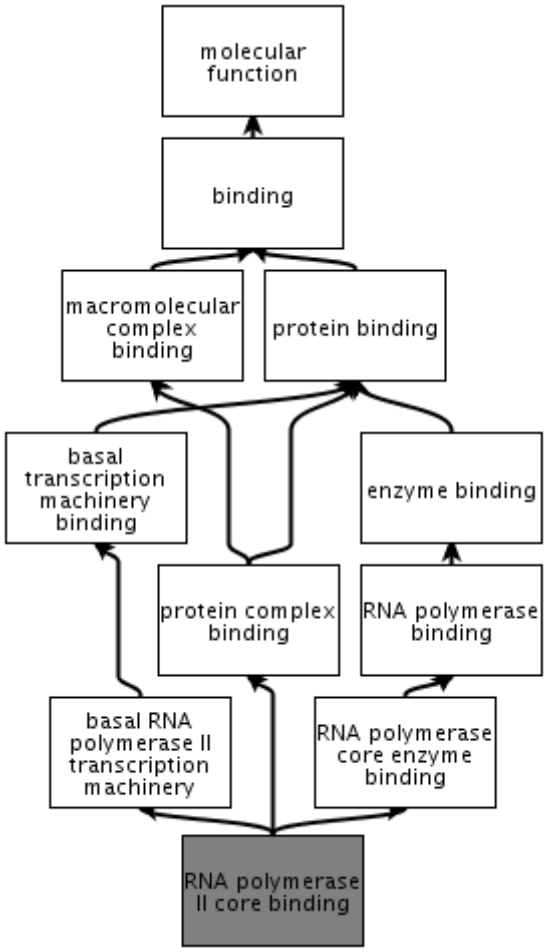
Ontologija	Broj čvorova
molekulska funkcija	9571
biološki procesi	24936
ćelijske komponente	3184

usmerenog acikličkog grafa podrazumeva da jedan čvor može imati više od jednog roditelja što je obično slučaj kod kompleksnih funkcija za čiji je opis neophodno više opštijih funkcija, kao što je prikazano na slici 3.10.



Slika 3.9: *Molekulska funkcija thyroid hormone binding.*

GO ontologija predstavlja funkciju proteina nezavisno od organizma u kom svaki od njih deluje. Za proučavanje funkcija proteina karakterističnih za određeni organizam, na primer za čoveka i njegove bolesti, razvijene su specifične ontologije, od kojih su najčešće korišćene *Human Phenotype Ontology* [95], *Unified Medical Language System* [3], *Disease Ontology* [42], itd.



Slika 3.10: *Primer čvorova koji imaju više od jednog roditelja.*

Pored GO ontologije, postoje i drugi načini označavanja funkcije proteina. Jedan od njih je i COG klasifikacija (eng. *Cluster of Orthologous Genes*), koja je takođe korišćena u ovom istraživanju i o kojoj će biti više reči u poglavlju 5.1. GO ontologija je bogatija od COG klasifikacije po broju funkcija koje sadrži. I pored ove razlike, za funkcije koje su zajedničke u oba sistema predstavljanja postoji bijektivno preslikavanje.[3]

[3]http://geneontology.org/external2go/cog2go

3.3 Postojeće metode predviđanja funkcije proteina

Osnovni pristup predviđanju funkcije proteina zasniva se na činjenici da proteini sa sličnom primarnom ili sekundarnom strukturom imaju sličnu funkciju. Proteini čije se sekvence ne razlikuju mnogo obično dele zajedničkog pretka, odnosno predstavljaju homologe, i pretpostavlja se da njihova mutacija tokom evolucije nije uticala na promenu funkcije koju obavljaju. Ovakav pristup je poznat pod nazivom prenos anotacije (eng. *transfer of annotation*) i predstavlja najčešći način za određivanje funkcije [74]. Mnoge studije su pokazale ograničenja hipoteze da slična primarna i/ili sekundarna struktura dva proteina implicira i sličnost u njihovim funkcijama [11,15], posebno zbog toga što mutacije na jednom nukleotidu i duplikacija gena mogu dovesti do promene funkcije iako sekvence i nakon ovih procesa ostaju međusobno slične [4,76].

Bez obzira na nedostatke polazne pretpostavke, postoji veliki broj metoda različitog kvaliteta koji predviđa funkciju proteina na osnovu sličnosti primarne ili sekundarne strukture. Postupak utvrđivanja nepoznate funkcije na osnovu sekvence obično počinje izračunavanjem mera sličnosti proteinske sekvence sa proteinima za koje je funkcija od ranije poznata, obično uz pomoć softverskih alata za poravnanja sekvenci kao što je BLAST [64]. BLAST izračuna skor za svako poravnanje kao i njegovu statističku značajnost izraženu preko veličine pod nazivom *e-vrednost*, koja predstavlja očekivanje da se u bazi proteina iste veličine čije su sekvence generisane na slučajan način pronađe sekvenca sa istom merom sličnosti. Što je e-vrednost manja, to je dobijeno poravnanje statistički značajnije. Neke metode, kao što je GOblet [65], odrede gornju granicu za e-vrednost i nepoznatom proteinu dodele funkcije svih proteina čiji je skor poravnanja iznad date granice. GOtcha metod [21] koristi prvi BLAST pogodak i predstavlja BLAST metod sa kojima su rezultati poređeni. GOtcha [21] i OntoBlast [108] koristi dobijene e-vrednosti kao težine za GO termine funkcija proteina iz baze podataka.

Pored primarne strukture, i atributi zasnovani na sekundarnoj i tercijarnoj strukturi proteina, kao što su 3D savijanje (eng. *3D-fold*), strukturni motivi i domeni, mogu se koristiti za određivanje homologa, a time i zajedničkih funkcija dva proteina. Javno dostupni veb servis ProKnow korišćenjem bajesovskih metoda na osnovu atributa zasnovanih na primarnoj i sekundarnoj strukturi proteina određuje koji je protein u bazi proteina sa anotiranom funkcijom najsličniji datom proteinu [89].

Osim navedenih metoda koje direktno koriste informacije dobijene poravnanjem, razvijane su i nove tehnike koje koriste ove informacije kao ulazni podatak za metode nadgledanog učenja [7,25,73,97,105]. Naprednije metode su uvodile i druge atribute u vektor ulaznih podataka kao što su fizičko-hemijske osobine [10,25,40,41], evolutivne relacije [14,20,22,24,50,53], mikronizove [17], mreže proteinskih interakcija [13,23,48,86], sekundarnu strukturu proteina [35, 85, 90, 91] ili njihovu kombinaciju [32, 34, 52, 72, 97].

Zajednica istraživača koji se bave predviđanjem funkcije proteina svake dve godine organizuje takmičenje prediktora pod nazivom CAFA - *Critical Assesment of Functional Annotation* [60]. Takmičari dobiju skup proteina za koje nije poznata funkcija i potrebno je da testiraju svoju metodu za predikciju na njima i pošalju rezultate organizatorima do propisanog roka. Nakon nekoliko meseci, neki proteini iz datog skupa biće eksperimentalno anotirani i za taj podskup se vrši evaluacija dobijenih rezultata. CAFA je jedno od mnogih sličnih takmičenja u biomedicinskim naukama [9].

3.4 Problem predviđanja funkcije proteina

Neka je dat skup podataka $\mathcal{D}_l = \{(p_1, T_1), \ldots, (p_n, T_n)\}$ gde je p_i vektorska reprezentacija i-tog proteina a T_i njegova eksperimentalno utvrđena funkcija u nekoj ontologiji $\mathcal{O}$. Svaki vektor p_i se sastoji od podataka o sekvenci proteina (npr. histogram n-grama) a može sadržati i druge podatke kao što su organizam kom protein pripada, njegova sekundarna struktura, interakcija sa drugim proteinima itd. Funkcionalna anotacija T_i koja odgovara i-tom proteinu predstavlja konzistentan podgraf ontologije $\mathcal{O}$, što znači da za svaki čvor $v \in T_i$ važi da su svi njegovi preci takođe u podgrafu T_i. Posledično, koren ontologije $\mathcal{O}$ se nalazi u svakom podgrafu T_i.

Problem predviđanja funkcije može se formulisati na sledeći način: za dati skup $\mathcal{D}$ funkcionalno anotiranih proteina i protein p koji ne pripada tom skupu, pronaći konzistentan podgraf $\hat{T} \subset \mathcal{O}$ za koji je najverovatnije da predstavlja njegovu eksperimentalnu anotaciju T:

$$\hat{T} = \underset{O \subset \mathcal{O}}{\operatorname{argmax}} \{Pr(O|p)\}$$

gde $Pr(O|p)$ predstavlja uslovnu verovatnoću da je proteinu p dodeljen graf O, pri čemu se maksimum računa nad svim konzistentnim podgrafovima O u ontologiji $\mathcal{O}$. Slika 3.11 prikazuje ilustraciju postavljenog problema. Možemo

primetiti da zahvaljujući uslovu konzistentnosti, svaki podgraf O jedinstveno je određen svojim listovima.

Slika 3.11: *Problem predviđanja funkcije. Obojeni čvorovi predstavljaju one čvorove ontologije koji ulaze u konzistentni podgraf kojim se opisuje funkcija datog proteina.*

3.5 Rešavanje problema predviđanja funkcije proteina metodom podržavajućih vektora za strukturni izlaz

Reprezentacija podataka

Kao što je navedeno u poglavlju 3.3, za funkcionalnu anotaciju koriste se različite informacije o proteinima. U ovom pristupu, funkciju proteina predviđamo samo na osnovu primarne sekvence, bez dodatnih podataka o proteinu (sekundarna struktura, iz kog organizma dolazi, itd.). Primarna sekvenca proteina u ovom radu kodirana je vektorom tetragrama. Naime, ulazni vektor x je dimenzije 20^4, čime je za svaki tetragram $(AAAA, AAAC, \ldots, YYYY)$ rezervisana jedinstvena pozicija. Vrednost koja će se nalaziti na toj poziciji odgovara broju pojavljivanja odgovarajućeg tetragrama u datoj proteinskoj sekvenci. Očigledno je da će vektor x biti redak, odnosno da će većinom sadržati nule. Na primer, ako je data sekvenca $MCAAAAAGHQR$, tada bi ona bila kodirana na sledeći način:

$$\boldsymbol{x} = \begin{bmatrix} 2(AAAA) \\ 0 \\ \vdots \\ 0 \\ 1(AAAG) \\ 0 \\ \vdots \\ 0 \\ 1(AAGH) \\ 0 \\ \vdots \\ 0 \\ 1(AGHQ) \\ 0 \\ \vdots \\ 0 \\ 1(CAAA) \\ 0 \\ \vdots \\ 0 \\ 1(GHQR) \\ 0 \\ \vdots \\ 0 \\ 1(MCAA) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.1}$$

Funkcija proteina predstavljena je kao podgraf GO ontologije. Vektorska reprezentacija svakog podgrafa ima dimenziju koja odgovara ukupnom broju čvorova u GO ontologiji, pri čemu je redosled čvorova u vektoru dobijen topološkim obilaskom grafa. Ukoliko se neki čvor nalazi u datom podgrafu,

tada će vrednost na poziciji tog čvora u vektoru biti 1, a u suprotnom 0. U ovom istraživanju ograničavamo se samo na ontologiju MFO koja predstavlja molekulsku funkciju proteina. Na primer, neka je u toj ontologiji funkcija proteina opisana podgrafom koji sadrži sledeće cvorove u topološkoj numeraciji: 8679, 9474, 9475, 9476, 9477, 9549, 9557, 9571. Tada bi njegova vektorska reprezentacija bila sledeća:

$$\boldsymbol{y} = \begin{bmatrix} 0(1) \\ \vdots \\ 0 \\ 1(8679) \\ 0 \\ \vdots \\ 0 \\ 1(9474) \\ 1(9475) \\ 1(9476) \\ 1(9477) \\ 0 \\ \vdots \\ 0 \\ 1(9549) \\ 0 \\ \vdots \\ 0 \\ 1(9557) \\ 0 \\ \vdots \\ 0 \\ 1(9571) \end{bmatrix} \tag{3.2}$$

Funkcija Ψ

Kao što je navedeno u poglavlju 2.5, za metodu podržavajućih vektora sa strukturnim izlazom od značaja je za konkretan problem definisati funkciju $\Psi : X \times Y \to \mathbb{R}^n$ koja predstavlja zajedničku vektorsku reprezentaciju ulaznog vektora $\boldsymbol{x}$ i izlaznog vektora $\boldsymbol{y}$. Funkcija Ψ se bira za svaki problem posebno na osnovu njegove prirode. Jedan primer izbora ove funkcije je prikazan na slici 2.2. Inspirisani ovim primerom, za problem predviđanja funkcije proteina koji rešavamo u ovom radu za funkciju Ψ odabrali smo tenzorski proizvod vektora $\boldsymbol{x}$ i $\boldsymbol{y}$:

$$\Psi(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x} \otimes \boldsymbol{y}$$

Dimenzija rezultujućeg vektora $\Psi(\boldsymbol{x}, \boldsymbol{y})$ jednaka je proizvodu dimenzija ulaznog vektora $\boldsymbol{x}$ i izlaznog vektora $\boldsymbol{y}$. U konkretnom slučaju, s obzirom da je $X = \mathbb{N}_0^{20^4}$, a $Y = \{0, 1\}^{9571}$, dimenzija vektora Ψ je oko 1 i po milijardu (precizno, 1 531 360 000). Vrednosti vektora Ψ dodeljene su na sledeći način: za svaki čvor ontologije funkcija rezervisan je blok koji je jednak vektoru $\boldsymbol{x}$, ukoliko se taj čvor pojavljuje u grafu $\boldsymbol{y}$, ili nula vektoru dimenzije jednake dimenziji vektora $\boldsymbol{x}$, u suprotnom. Na primer, neka je p dimenzija ulaznog vektora $\boldsymbol{x}$, q dimenzija izlaznog vektora $\boldsymbol{y}$ i neka $\boldsymbol{y}$ sadrži jedinice na pozicijama a, b i c, dok su na ostalim pozicijama nule. Tada bi vektor Ψ bio sledećeg oblika:

$$\mathbf{0} = [\underbrace{0, \ldots, 0}_{p \text{ puta}}]$$

$$\boldsymbol{x} = [x_1, \ldots, x_p]$$

$$\boldsymbol{y} = [0, \ldots, 0, \underbrace{1}_{a}, 0, \ldots, 0, \underbrace{1}_{b}, 0, \ldots, 0, \underbrace{1}_{c}, 0, \ldots, 0]$$

$$\Psi(\boldsymbol{x}, \boldsymbol{y}) = [\mathbf{0}, \ldots, \mathbf{0}, \underbrace{\boldsymbol{x}}_{a}, \mathbf{0}, \ldots, \mathbf{0}, \underbrace{\boldsymbol{x}}_{b}, \mathbf{0}, \ldots, \mathbf{0}, \underbrace{\boldsymbol{x}}_{c}, \mathbf{0}, \ldots, \mathbf{0}]$$

Ogromna dimenzionalnost ovako odabranog vektora Ψ svakako je njegov nedostatak. Međutim, vektor Ψ kodira sve informacije o primeru $(\boldsymbol{x}, \boldsymbol{y})$: za svaki tetragram proteinske sekvence i svaki čvor iz usmerenog acikličkog grafa proteinske funkcije postoji vrednost u vektoru Ψ kojoj se pridružuje težina iz vektora koeficijenata $\boldsymbol{w}$. Na taj način, postoji mogućnost da ukoliko nepoznati protein ima funkciju f sa pretkom g, pri čemu se funkcija f nije pojavila u trening skupu, a funkcija g jeste, model može da predvidi funkciju g za dati protein. Ovo ne bi bilo moguće da je npr. za vektor Ψ uzet

tenzorski proizvod vektora tetragrama i vektora listova usmerenog acikličkog grafa funkcija.

Funkcija Δ

Kao što je pomenuto u poglavlju 2.4, da bi se algoritam odsecajućih ravni završio u konačnom broju koraka, neophodno je da funkcija Δ bude nenegativna, da je jednaka nuli jedino kada su njeni operandi jednaki i da za svaki izlazni vektor $\boldsymbol{y^*}$ ograničena odozgo. U ovom istraživanju razmatrane su sledeće funkcije koje trivijalno ispunjavaju tražene uslove:

1. **Žakardovo rastojanje**

   Žakardovo rastojanje (*Paul Jaccard*) nad konačnim skupovima A i B definiše se na sledeći način:

   $$d_j(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

   Ovu meru primenjujemo na grafove $\boldsymbol{y}$ i $\boldsymbol{y'}$ tako što svaki graf svedemo na skup čvorova koje sadrži.

2. $\mathbf{1 - F_1}$

   Prilikom ispitivanja kvaliteta binarne klasifikacije, gde za dati test primer utvrđujemo da li pripada ili ne pripada datoj klasi, od značaja su sledeće veličine:

   - tp - broj tačno klasifikovanih pozitivnih test primera (koji pripadaju klasi)
   - tn - broj tačno klasifikovanih negativnih test primera (koji ne pripadaju klasi)
   - fp - broj lažno pozitivnih test primera koji predstavlja broj pogrešno klasifikovanih negativnih test primera (koji ne pripadaju klasi)
   - tn - broj lažno negativnih test primera koji predstavlja broj pogrešno klasifikovanih pozitivnih test primera (koji pripadaju klasi)

Preciznost p i odziv r se definišu kao mere kvaliteta binarnog klasifikatora na sledeći način:

$$p = \frac{tp}{tp + fp}, r = \frac{tp}{tp + fn}$$

F_1 mera definiše se kao mera tačnosti u binarnoj klasifikaciji kao harmonijska sredina preciznosti p i odziva r:

$$F_1 = 2 \cdot \frac{p \cdot r}{p + r}$$

U strukturnoj klasifikaciji, preciznost i odziv se mogu definisati na nivou pojedinačnog test primera. Na primer, ako je datom test primeru pridružen izlazni vektor $\boldsymbol{y} = [0, 0, 1, 0, 0, 1]$, bez obzira koju strukturu predstavlja (niz, graf, stablo, ...), a klasifikator je predvideo vektor $\boldsymbol{y'} = [1, 0, 0, 1, 0, 1]$, tada za par $(\boldsymbol{y}, \boldsymbol{y'})$ možemo odrediti veličine tp, tn, fp, fn:

$$\boldsymbol{y'} = [\underbrace{1}_{\in \text{fp}}, \underbrace{0}_{\in \text{tn}}, \underbrace{0}_{\in \text{fn}}, \underbrace{1}_{\in \text{fp}}, \underbrace{0}_{\in \text{tn}}, \underbrace{1}_{\in \text{tp}}]$$

Na taj način, možemo odrediti preciznost p i odziv r za jedan test primer, a time i vrednost mere F_1, što znači da meru F_1 možemo koristiti kao meru sličnosti tačnog i predviđenog vektora u strukturnoj klasifikaciji. U konkretnoj primeni za predviđanje funkcije proteina, neophodna nam je mera različitosti zbog čega koristimo $1 - F_1$:

$$d_{F_1}(\boldsymbol{y}, \boldsymbol{y'}) = 1 - F_1(\boldsymbol{y}, \boldsymbol{y'})$$

3. **Semantičko rastojanje**

Semantičko rastojanje s_k je mera uvedena u radu Klarka i Radivojca [6] kao mera različitosti dva grafa koji su podgrafovi iste ontologije koja opisuje funkciju proteina i razvijena je za potrebe evaluacije prediktora proteinske funkcije na CAFA takmičenju. Da bismo definisali meru s_k, neophodno je objasniti nekoliko osnovnih pojmova.

Pretpostavimo da je svaki čvor ontologije binarna slučajna promenljiva i da je njihova raspodela verovatnoća nad $X \times Y$ nepoznata ali fiksirana. Pretpostavimo da je svaki vektor funkcija $\boldsymbol{y}$ generisan ovom

raspodelom i da se njegova verovatnoća može faktorisati na osnovu strukture ontologije, odnosno da su vektori funkcija generisani bajesovskom mrežom. Na osnovu ove pretpostavke, svaki čvor je nezavisan od drugih predaka osim od svojih roditelja i verovatnoća da će biti generisan graf T može se faktorisati kao proizvod uslovnih verovatnoća njegovih čvorova [82]:

$$Pr(T) = \prod_{v \in T} Pr(v|\mathcal{P}(v))$$

gde v označava čvor u grafu a $\mathcal{P}(v)$ skup roditeljskih čvorova za čvor v. Uslovna verovatnoća pojedinačnog čvora $Pr(v|\mathcal{P}(v))$ predstavlja verovatnoću da se u grafu nađe čvor v ukoliko taj graf sadrži njegove roditelje $\mathcal{P}(v)$.

Klark i Radivojac u istom radu uvode i pojam informacionog sadržaja grafa funkcije proteina (u nastavku skraćeno graf funkcije). Informacioni sadržaj intuitivno predstavlja broj bitova informacije koje saznajemo o proteinu ukoliko je za njega predviđen određeni graf funkcije:

$$i(T) = \log \frac{1}{Pr(T)}$$

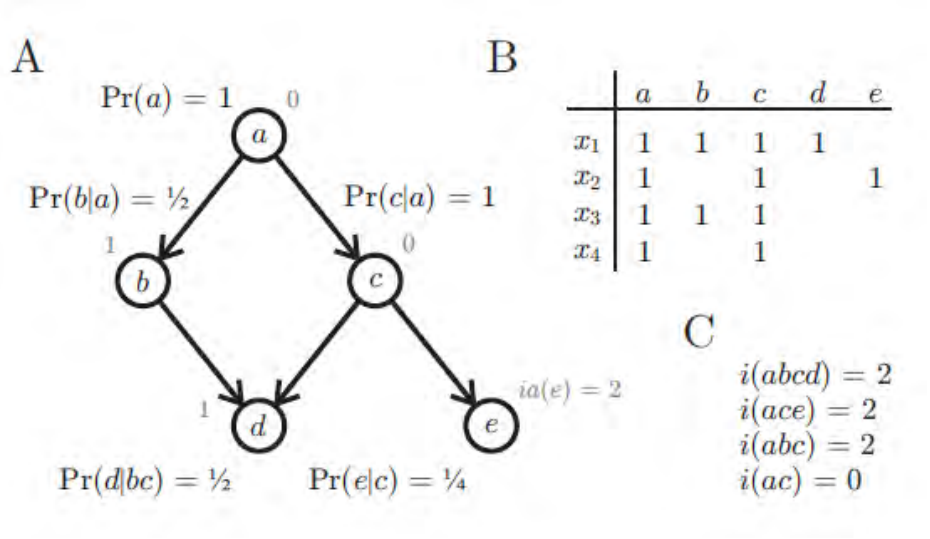
Po konvenciji, koristi se logaritam za osnovu 2. Kombinovanjem poslednje dve jednačine dobijamo:

$$\begin{aligned} i(T) &= \sum_{v \in T} \log \frac{1}{Pr(v|\mathcal{P}(v))} \\ &= \sum_{v \in T} ia(v) \end{aligned}$$

gde, radi pojednostavljivanja, uvodimo oznaku ia za negativni logaritam $Pr(v|\mathcal{P}(v))$. Za ovu veličinu uveden je naziv informacioni doprinos (*information accretion*) i možemo je shvatiti kao povećanje količine informacije dobijene dodavanjem čvora v u graf funkcije T.

U prethodnom teorijskom razmatranju, pretpostavljeno je da su čvorovi ontologije slučajne promenljive sa fiksiranim ali nepoznatim raspodelama verovatnoća. Ilustracije radi, na slici 3.12 dat je jednostavan primer u kome su za svaki čvor poznate njihove uslovne verovanoće, preciznije verovatnoće njihovog pojavljivanja u grafu funkcije proteina ako

se u grafu nalaze njihovi roditelji. Prikazana je ontologija sa 5 čvorova zajedno sa tabelom uslovnih verovatnoća i izračunatim informacionim doprinosom (A), zatim skup od 4 proteina prikazanih vektorima svojih funkcija koje su generisane na osnovu date raspodele verovatnoća (B) i informacioni sadržaj sa svaki graf (C).



Slika 3.12: *Primer ontologije, skupa podataka i računanja informacionog doprinosa. Slika je preuzeta iz [6] A-ontologija modelovana bajesovskom mrežom sa uslovnom verovatnoćom (unapred zadatom) i informacionim doprinosom pored svakog čvora (siva boja). B-skup proteinskih funkcija generisanih na osnovu pretpostavljenog verovatnosnog modela bajesovske mreže. C-informacioni sadržaj grafova koji predstavljaju funkcije proteina koji navedenih u delu B.*

Klark i Radivojac dalje uvode veličine preostala neizvesnost *ru* (eng. *remaining uncertainty*) i pogrešna informacija *mi* (eng. *misinformation*) kao analogne odzivu i preciznosti, redom. Za dati tačan graf T funkcije proteina i predviđeni graf P dobijen kao izlaz nekog prediktora *ru* i *mi* se formalno definisu na sledeći način:

$$ru(T,P) = \sum_{v \in T \setminus P} ia(v)$$

$$mi(T,P) = \sum_{v \in P \setminus T} ia(v)$$

Preostala neizvesnost odnosi se na informaciju koju predviđeni graf P nije pružio, dok se pogrešna informacija odnosi na informaciju koju

predviđeni graf P pružio a koja je netačna. U idealnom slučaju, kada je $T = P$, i ru i mi su jednaki nuli.

Konačno, semantičko rastojanje s_k definišemo kao

$$s_k(T, P) = (ru(T, P)^k + mi(T, P)^k)^{\frac{1}{k}}$$

Pored semantičkog rastojanja, postoji i normalizovano semantičko rastojanje koje je korišćeno u ovom istraživanju:

$$s_k^n(T, P) = \frac{(ru(T, P)^k + mi(T, P)^k)^{\frac{1}{k}}}{\sum_{v \in P \cup T} ia(v)}$$

Rešavanje optimizacionog problema

Kao što je bilo navedeno u poglavlju 2.5, jedan od koraka prilagođavanja SVM struct metode konkretnom problemu je konstrukcija algoritama za rešavanje sledećih optimizacionih problema:

1. prilikom treniranja:

$$\bar{\boldsymbol{y}}_{\boldsymbol{i}} = \arg\max_{\boldsymbol{y} \in Y} (\Delta(\boldsymbol{y}_{\boldsymbol{i}}, \boldsymbol{y}) + \langle \boldsymbol{w}, \Psi(\boldsymbol{x}_{\boldsymbol{i}}, \boldsymbol{y}) \rangle)$$

2. prilikom testiranja:

$$\boldsymbol{y'} = \arg\max_{\boldsymbol{y} \in Y} (\langle \boldsymbol{w}, \Psi(\boldsymbol{x'}, \boldsymbol{y}) \rangle)$$

Ovi optimizacioni problemi mogu se rešavati istim ili različitim algoritmima. U oba slučaja, traži se maksimum po svim elementima skupa Y, koji se sastoji od svih konzistentnih podgrafova ontologije funkcija i eksponencijalne je kardinalnosti. Posebno je važno pronaći efikasan algoritam za rešavanje optimizacionog problema prilikom treniranja, jer se on izvršava za svaki trening primer u svakoj iteraciji.

Ideja algoritma za optimizaciju predloženog u ovom radu je sledeća:

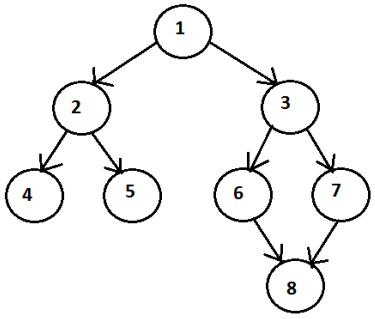
1. neka je H funkcija koju želimo da maksimizujemo, $\boldsymbol{y}_{max}$ promenljiva u kojoj ćemo čuvati graf koji ima maksimalnu vrednost funkcije H a $H_{max} = H(\boldsymbol{y}_{max})$; inicijalno, $\boldsymbol{y}_{max}$ je graf koji se sastoji samo od korena, a $H_{max} = -\infty$

2. neka je L lista grafova $\boldsymbol{y}$ sortiranih opadajuće po vrednosti $H(\boldsymbol{y})$; inicijalno, lista L sadrži jedan graf koji se sastoji samo od korena

3. Za potrebe algoritma uvodimo novi pojam *grafa potomka*. Najpre ćemo dati njegovu neformalnu definiciju a potom i formalnu. Postupak generisanja grafova potomaka zvaćemo *proširivanje grafa*.

   Neformalno, ako je dat graf $\boldsymbol{y}$, graf potomak je svaki konzistentni graf nastao dodavanjem jednog direktnog potomka nekom od čvorova grafa $\boldsymbol{y}$. Na taj način, graf $\boldsymbol{y}$ se proširuje za jedan čvor, osim u sledećem slučaju kada mu se dodaje više od jednog čvora: ukoliko postoji roditelj novog čvora koji ne pripada grafu $\boldsymbol{y}$, on se zbog konzistentnosti mora dodati u prošireni graf. U tom slučaju može se dogoditi i da datom grafu bude dodat čitav novi podgraf kome je pridodati roditelj list.

   Na primer, neka je data ontologija na slici 3.13 i neka je graf $\boldsymbol{y} = [1, 2, 3, 6]$ jedan njen podgraf. Njegovi grafovi potomci bili bi grafovi $\boldsymbol{y}_1 = [1, 2, 3, 4, 6]$, $\boldsymbol{y}_2 = [1, 2, 3, 5, 6]$, $\boldsymbol{y}_3 = [1, 2, 3, 6, 7]$ i $\boldsymbol{y}_4 = [1, 2, 3, 6, 7, 8]$. Možemo primetiti da se broj čvorova svih grafova potomaka od grafa $\boldsymbol{y}$ razlikuje za jedan, osim kod grafa $\boldsymbol{y}_4$ gde pored čvora 8 zbog konzistentnosti dodajemo i njegov drugi potomak, čvor 7, koji nije bio u polaznom grafu $\boldsymbol{y}$.



Slika 3.13: *Primer male ontologije.*

Formalno, neka su:

- t čvor ontologije
- T skup čvorova ontologije
- $Children(t)$ svi direktni potomci čvora t

- $ChildrenList(T) = \bigcup_{t \in T} Children(t)$
- $Parents(t)$ svi direktni preci čvora t
- $l(\boldsymbol{y})$ skup svih listova grafa $\boldsymbol{y}$

Za graf $\boldsymbol{z}$ kažemo da je *graf potomak* grafa $\boldsymbol{y}$ ako važi:

(a) $\boldsymbol{y} \subset \boldsymbol{z}$

(b) $(\forall v \in \boldsymbol{z}$ tako da $v \notin \boldsymbol{y})(v \in Children(l(\boldsymbol{y})) \vee (\exists t \in Children(l(\boldsymbol{y}))$ tako da $v \in Parents(t)))$

4. Za graf $\boldsymbol{y}_{head}$ smešten na početku liste L generišemo *grafove potomke*. Za svaki na taj način dobijeni graf izračunamo vrednost funkcije H, poredimo da li je veća od aktuelnog maksimuma H_{max}, po potrebi ažuriramo vrednosti H_{max} i $\boldsymbol{y}_{max}$, uklanjamo graf sa početka liste L a njegove potomke ubacujemo u listu L tako da ona očuva sortiranost.

5. prethodni korak ponavljamo dok ima elemenata u listi L ili do unapred određenog maksimalnog broja koraka. Po završetku algoritma, traženi graf se nalazi u promenljivoj $\boldsymbol{y}_{max}$

Da bismo ubrzali i usmerili izvršavanje algoritma, uvodimo sledeće parametre:

- *imax* - maksimalna vrednost informacionog sadržaja grafa
  Prilikom određivanja grafa $\boldsymbol{y}$ koji maksimizuje funkciju H, u obzir ćemo uzeti samo one grafove $\boldsymbol{y}$ čiji je informacioni sadržaj $i(\boldsymbol{y})$ manji od parametra *imax*. Time izbegavamo grafove koji nisu realni predstavnici grafa funkcije proteina, kao što su grafovi sa ogromnim brojem čvorova ili grafovi koji sadrže više čvorova sa velikim informacionim doprinosom. Za vrednost ovog parametra uzimana je maksimalna vrednost informacionog sadržaja grafova u trening skupu.

- *smax* - maksimalni broj koraka algoritma
  Vrednost ovog parametra određivana je za svaki klasifikacioni model u postupku validacije od ponuđenih vrednosti $64, 128, 256$. Suprotno intuitivnoj pretpostavci, za neke modele optimalne rezultate davala je manja vrednost parametra *smax*. Ovaj parametar takođe služi da ograniči maksimalnu dužinu liste L - s obzirom da se u svakom koraku algoritma koristi jedan (prvi) element liste L, nema potrebe da njena dužina bude veća od ukupnog broja koraka.

Algoritam 2 Algoritam za optimizaciju funkcije H

Ulaz: trening primer $(\boldsymbol{x_i}, \boldsymbol{y_i})$
Izlaz: $\boldsymbol{y_{best}}$ koji maksimizuje $H(\boldsymbol{x_i}, \boldsymbol{y})$ nad Y
Inicijalizacija: $L = \{\boldsymbol{y_{root}}\}, L_{used} = \emptyset, \boldsymbol{y_{best}} = \emptyset, H_{best} = -\infty$
repeat
    $\boldsymbol{y_{head}}$:= prvi element iz liste L
    generisati sve grafove potomke Y_{ext} grafa $\boldsymbol{y_{head}}$
    for each $\boldsymbol{y_{ext}} \in Y_{ext}$ **do**
        if $\boldsymbol{y_{ext}} \in L_{used}$ **or** $i(\boldsymbol{y_{ext}}) \geq imax$ **then**
            continue
        $H_{ext} := H(\boldsymbol{y_{ext}})$
        ubaciti $\boldsymbol{y_{ext}}$ u sortiranu listu L
        if $H_{ext} > H_{best}$ **then**
            ažurirati $\boldsymbol{y_{best}}, H_{best}$
    ukloniti $\boldsymbol{y_{head}}$ iz L
    ubaciti $\boldsymbol{y_{head}}$ u L_{used}
    uvećati *step*
    if $|L| > smax$ **then**
        ukloniti višak grafova sa kraja liste L
until *step* $> smax$ or L je prazna

S obzirom na korak generisanja grafova potomaka, vremenska složenost algoritma je eksponencijalna u odnosu na broj čvorova ontologije. Parametar *imax* usmerava pretragu da zaobiđe podgrafove sa velikim brojem čvorova, dok parametar *smax* osigurava završavanje algoritma u konačnom broju iteracija.

3.6 Eksperimenti

Cilj eksperimenata bio je višestruk:

1. Ispitati koliko izbor funkcije Δ ima uticaj na performanse klasifikacionog modela i posebno ispitati kakve rezultate daju klasifikatori koji za funkciju različitosti koriste semantičko rastojanje, meru koja do sada nije korišćena u fazi treniranja klasifikacionog modela

2. Ispitati koliko je informacija o organizmu iz kog protein dolazi značajna za kvalitet predviđanja, odnosno kakve će rezultati prediktora trenirani

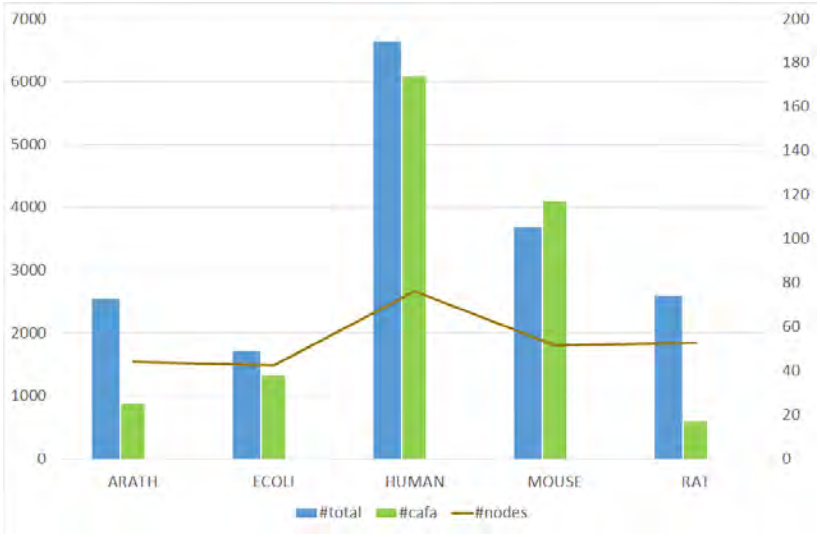
na proteinima jednog organizma dati kada se primene na proteine drugog organizma

3. Ispitati kvalitet prediktora u odnosu na prediktore koji su učestvovali na CAFA2 takmičenju

Skup funkcionalno anotiranih proteina koji je korišćen u ovom istraživanju preuzet je iz javno dostupne baze podataka Swiss-Prot iz juna 2013. godine.[4] S obzirom na postavljene ciljeve, eksperiment je dizajniran na sledeći način:

- Ceo skup proteina podeljen je na 5 disjunktnih podskupova, od kojih je svaki podskup predstavljao proteine jednog organizma:
  - *Arabidopsis thaliana* (ARATH)
  - *Escherichia coli* (ECOLI)
  - *Homo sapiens* (HUMAN)
  - *Mus musculus* (MOUSE)
  - *Rattus norvegicus* (RAT)

  Izabrani su organizmi koji su imali najveći broj proteina u osnovnom skupu podataka, pri čemu su to takođe organizmi za koje su evaluirani prediktori na CAFA takmičenju. Slika 3.14 prikazuje broj proteina u



Slika 3.14: *Broj proteina po organizmima, broj CAFA proteina po organizmima (sekundarna vertikalna osa) i broj čvorova MFO ontologije koji se pojavljuju u prvom skupu (linijski dijagram, primarna vertikalna osa).*

[4]http://biofunctionprediction.org/node/12

svakom od ovako generisanih skupova zajedno sa brojem čvorova MFO ontologije funkcija koji se u tim skupovima pojavljuju. Na taj način smo dobili 5 skupova proteina nad kojima smo mogli da ispitamo sve zadatke istraživanja.

- Za svaki od 5 skupova i svaku od 4 mere različitosti Δ ($jaccard, 1 - f1, s2, s1$), klasifikacioni model je kreiran na sledeći način:
  1. Svaki skup podataka je podeljen u odnosu 75%:25% na trening skup i test skup (tačan broj proteina naveden je u tabeli 4.3)
  2. Iz svakog trening skupa izdvojeno je 25% proteina za postupak validacije, u kom su određene optimalne vrednosti parametra C SVM struct metode i parametra $smax$ algoritma za optimizaciju funkcije H, za svaku od 4 izabrane mere Δ
  3. Nad svakim trening skupom treniran je klasifikacioni model sa vrednostima parametara određenim tokom validacije
  4. Dobijeni klasifikacioni modeli testirani su nad test skupovima za odgovarajući organizam. Dobijeni rezultati poređeni su međusobno po organizmima, čime je utvrđeno koja od izabranih mera Δ daje najbolje rezultate za koji organizam.
  5. Izvršeno je ukršteno testiranje: modeli svakog organizma testirani su i na drugim organizmima.
- Najbolji ovako dobijeni rezultati upoređeni su sa rezultatima prediktora funkcije proteina validiranim na poslednjem CAFA takmičenju [36]. Važno je napomenuti da CAFA za prediktore koji se takmiče ne obezbeđuje obavezan skup proteina za treniranje, već samo skup proteina za testiranje. Rezultati prediktora na skupu proteina za testiranje se potom porede na osnovu čega se takmičari rangiraju. Svakom učesniku dopušteno je da izabere proteine koje će koristiti za treniranje svog modela, a posebno karakteristike proteina koje će koristiti kao ulazne podatke. Karakteristike proteina koje se obično koriste za predviđanje funkcije su raznovrsne i uključuju informacije iz proteinske sekvence, filogenetske informacije, fizičko-hemijske osobine, podatke o strukturi proteina, podatke dobijene masenom spektrometrijom, podatke iz literature, itd. Na svakom učesniku je da na osnovu svog biološkog znanja izabere karakteristike koje će mu doneti optimalne rezultate.

  Cilj CAFA takmičenja je da utvrdi koji model daje najbolje rezultate na datom skupu proteina evaluirajući pritom i biološke veštine

takmičara u izboru pravih karakteristika proteina za ulazne podatke i njihovo računarsko umeće prilikom izgradnje modela za predikciju. S obzirom da učesnici CAFA takmičenja treniraju svoje modele na različitim podacima, CAFA ne pruža pouzdano poređenje algoritama na kojima su oni zasnovani u smislu osnovnih koncepata mašinskog učenja. Izuzetna kompleksnost domena na koji se predloženi algoritmi primenjuju otežava direktnu primenu ovih koncepata.

4. Rezultati eksperimenata i diskusija

4.1 Rezultati

Za svaki od trening skupova proteina podeljenih po organizmima trenirana su 4 klasifikaciona modela, po jedan za svaku meru različitosti grafova funkcije $(jaccard, 1 - f1, s_2, s_1)$. Na taj način, dobijeno je 20 klasifikacionih modela. Pored toga, na svaki test skup primenjeni su i sledeći jednostavni klasifikatori koji se obično koriste kao osnovne (*baseline*) metode pri poređenju rezultata:[1]

1. Naivni klasifikator *naive* dodeljuje svakom čvoru skor koji odgovara njegovoj frekvenciji pojavljivanja u trening skupu i tako formirani graf pridružuje svakom test primeru.

2. BLAST klasifikator *blast* funkcioniše na sledeći način: neka je $\boldsymbol{x}$ test primer, čvor v koji odgovara jednoj funkciji iz MFO ontologije i $S_v = \{s_1, s_2, \ldots, s_n\}$ sekvence trening skupa koje sadrže funkciju v u svom grafu funkcija; tada se funkciji v za test primer $\boldsymbol{x}$ dodeljuje skor

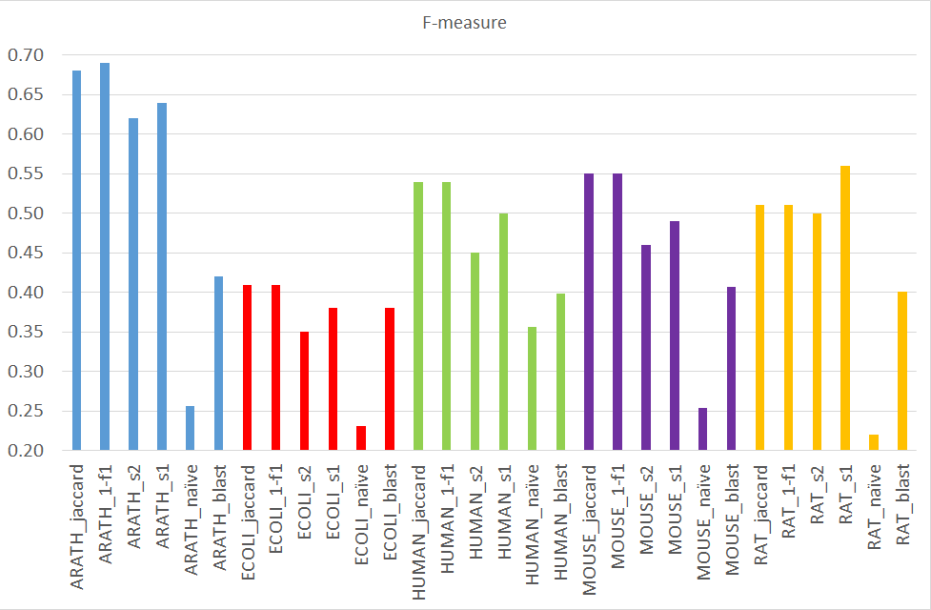
$$\max_{s \in S_v} \{sid(x, s)\}$$

   gde je $sid(x, s)$ maksimalna vrednost identičnosti sekvenci koju je vratio BLAST algoritam [64] pri poravnanju dve sekvence.

Prilikom svakog testiranja računata je prosečna vrednost funkcije gubitka, prosečna vrednost preciznosti i prosečna vrednost odziva po svim test primerima. Na osnovu prosečnih vrednosti preciznosti i odziva izračunata je F-mera koja je korišćena za međusobno poređenje performansi klasifikacionih modela nad istim organizmima.

[1] http://www.biofunctionprediction.org

Na slici 4.1 prikazani su rezultati testiranja svih klasifikacionih modela na test skupovima odgovarajućih organizama. Možemo uočiti da, u poređenju sa osnovnim metodama *naive* i *blast*, svi klasifikacioni modeli imaju bolje performanse, osim za model $ECOLI\_s2$ koji je slabiji od $ECOLI\_blast$. Dalje, u okviru istog organizma, mere *jaccard* i $1 - f1$ su približne i daju bolje rezultate od ostalih, osim kod organizma RAT gde je mera $s1$ bolja od ostalih. Ako posmatramo najbolje modele za svaki organizam, najviša F-mera je zabeležena za organizam $ARATH$, najniža za organizam $ECOLI$, koji je jedini prokariotska vrsta u ovom eksperimentu, dok su vrednosti F-mere za $HUMAN$, $MOUSE$ i RAT približni. Slabi rezultati kod $ECOLI$ se mogu objasniti činjenicom da je odnos broja proteina i broja čvorova MFO ontologije koji se pojavljuju u trening skupu manji nego kod svih ostalih organizama (tabela 4.3).



Slika 4.1: *Performanse dobijenih klasifikacionih modela i osnovnih modela (blast,naive) izražene F-merom. Klasifikacioni modeli istog organizma prikazani su istom bojom.*

Da bismo ispitali uticaj genetskog porekla proteina, primenili smo svaki klasifikacioni model na test proteine ostalih organizama. Kompletni rezultati prikazani su u tabeli 4.4, a u tabeli 4.1 izdvojene su vrednosti F-mere za najbolje klasifikatore po merama *jaccard,1-f1,s2,s1* zajedno sa oznakom koja mera je postigla najvišu vrednost. Modeli trenirani na trening skupu proteina iz organizama $ARATH, ECOLI, MOUSE, RAT$ bolje predviđaju

funkcije "svojih" proteine nego proteina iz drugih organizma, s tim što model $MOUSE$ daje približne rezultate za organizam RAT i za "svoje" proteine. Međutim, kod čoveka to nije slučaj: model $HUMAN$ bolje predviđa funkciju proteina iz organizama $MOUSE$ i RAT nego funkciju sopstvenih proteina. Ovo zapažanje nije iznenađujuće i može biti objašnjeno sledećim činjenicama:

- Organizmi $HUMAN$, $MOUSE$ i RAT su blisko evolutivno povezani i stoga je za očekivati da njihovi slični proteini (sa sličnom primarnom sekvencom) imaju slične funkcije
- $HUMAN$ model je treniran na većem broju proteina nego preostala dva modela, a s obzirom na evolutivnu bliskost, moguće je da taj trening skup pokriva neke funkcije iz test skupa $MOUSE$ i RAT organizama koje njihovi trening skupovi ne pokrivaju; zbog toga su RAT i $MOUSE$ proteini bili "lakši" za $HUMAN$ model lakši za predviđanje nego $HUMAN$ proteini

Posmatrano po merama, i u ukrštenom testiranju najbolje rezultate za sve organizme dale su mere *jaccard* i *1-f1*, izuzev modela RAT koji ponovo ima najveću vrednost F-mere za meru $s1$.

Tabela 4.1: *Ukršteno testiranje klasifikacionih modela na drugim organizmima, zajedno sa oznakom mere koja je dala najbolje rezultate (0-jaccard,1-1-f1,2-s2,3-s1). Maksimalne vrednosti po vrstama (proteine kog organizma model najbolje predviđa) su podebljane.*

best f1	ARATH	ECOLI	HUMAN	MOUSE	RAT
ARATH	**0.69 (1)**	0.4 (1)	0.39 (1)	0.4 (1)	0.35 (1)
ECOLI	0.41 (0,1)	**0.41 (0,1)**	0.34 (1)	0.35 (1)	0.31 (1,2)
HUMAN	0.44 (1)	0.36 (1)	0.54 (0,1)	0.6 (0)	**0.61 (0,1)**
MOUSE	0.45 (0)	0.39 (0)	0.52 (0,1)	**0.55 (0,1)**	0.55 (1)
RAT	0.41 (3)	0.36 (3)	0.51 (3)	0.52 (3)	**0.56 (3)**

Na osnovu dobijenih rezultata ukrštenog testiranja, za reprezentativne klasifikacione modele za primenu na CAFA proteine i za poređenje sa drugim prediktorima izabrani su modeli sa najvišom vrednošću F-mere, a u slučaju da su se maksimalne vrednosti F-mere za dve funkcije razlikovale za < 0.001, uzet je model koji je dao više kompletno tačno predviđenih test primera.

- za organizam $ARATH$, klasifikacioni model $ARATH\_1 - f1$

- za organizam $ECOLI$, klasifikacioni model $ECOLI\_1 - f1$
- za organizam $HUMAN$, klasifikacioni model $HUMAN\_jaccard$
- za organizam $MOUSE$, klasifikacioni model $HUMAN\_jaccard$
- za organizam RAT, klasifikacioni model $HUMAN\_jaccard$

Tabela 4.2 prikazuje rezultate testiranja na CAFA proteinima i poziciju dobijenih rezultata u rang listi CAFA prediktora. Rezultati na $ARATH$ proteinima su u samom vrhu, na RAT i $MOUSE$ u prvih 15% evaluiranih prediktora, dok su rezultati na $HUMAN$ i $ECOLI$ proteinima slabiji.

Tabela 4.2: *Rang naših modela na listi CAFA prediktora, od ukupno 125 prediktora funkcije proteina.*

organizam	F-mera predloženog prediktora	F-mera najbolje rangiranog prediktora	CAFA rang
ARATH	0.69	0.74	4
ECOLI	0.36	0.6	75
HUMAN	0.47	0.62	45
MOUSE	0.54	0.62	16
RAT	0.63	0.78	17

4.2 Zaključak i dalji rad

Rezultati ovog istraživanja dali su sledeće odgovore na postavljena pitanja:

1. Prema rezultatima prikazanim u tabeli 4.4, razlike u rezultatima klasifikacionih modela za različiti izbor funkcije Δ kreću se od 0.6 do 0.9 u zavisnosti od organizma, pa možemo zaključiti da funkcija Δ utiče na performanse klasifikacionog modela. Semantičko rastojanje najčešće nije bila mera za koju je postizana najviša vrednost F-mere (osim za kvalifikacione modele trenirane na proteinima organizma RAT)

2. Prilikom unakrsnog testiranja, očekivano je kod najvećeg broja klasifikacionih modela najbolji rezultat imao model treniran na proteinima istog organizma. Izuzetak su organizmi $MOUSE$ i RAT - funkciju

njihovih proteina je bolje predviđao klasifikacioni model treniran na humanim proteinima. Svi eukariotski modeli imali su loš rezultat na predviđanju funkcije proteina jedinog prokariotskog organizma, *ECOLI*, što potvrđuje sličnost funkcija proteina evulutivno bliskih organizama.

3. Rezultati predloženih prediktora uporedivi su sa aktuelnim rezultatima prikazanim na poslednjem CAFA takmičenju, najrelevantnijem takmičenju u ovoj oblasti. Važno je naglasiti da CAFA ne propisuje koji se proteini koriste za treniranje modela i da je većina takmičara trenirala svoje prediktore na mnogo većim skupovima podataka koji su sadržali proteine raznih organizama. Treniranje na većem skupu podataka koji se sastoji od različitih organizama bi svakako poboljšalo rezultate s obzirom na činjenicu da su proteini evolutivno povezanih organizama sa istim funkcijama sličnih sekvenci.

4.3 Dodatak

Tabela 4.3: *Broj proteina u korišćenim skupovima i broj čvorova MFO ontologije koji se u celom skupu (za trening i za testiranje) pojavljuju.*

	#ukupno	#trening skup	#test skup	#cafa skup	#cvorovi
ARATH	2538	1904	634	25	1546
ECOLI	1728	1296	432	38	1485
HUMAN	6635	4977	1658	174	2661
MOUSE	3683	2763	920	117	1809
RAT	2601	1951	650	17	1852

Tabela 4.4: *Testiranje klasifikacionih modela na test skupovima odgovarajućih organizama. Maksimalne vrednosti po vrstama (proteine kog organizma model najbolje predviđa) su podebljane.*

			test		
jaccard	ARATH	ECOLI	HUMAN	MOUSE	RAT
ARATH	**0.68**	0.39	0.36	0.37	0.33
ECOLI	0.41	**0.41**	0.32	0.32	0.3
HUMAN	0.44	0.36	0.54	0.6	**0.61**
MOUSE	0.45	0.39	0.52	**0.55**	0.54
RAT	0.35	0.32	0.46	0.48	**0.51**
1-f1	ARATH	ECOLI	HUMAN	MOUSE	RAT
ARATH	**0.69**	0.4	0.39	0.4	0.35
ECOLI	0.41	**0.41**	0.34	0.35	0.31
HUMAN	0.43	0.35	0.54	0.59	0.61
MOUSE	0.4	0.35	0.52	**0.55**	0.55
RAT	0.35	0.32	0.45	0.47	**0.51**
s2	ARATH	ECOLI	HUMAN	MOUSE	RAT
ARATH	**0.62**	0.3	0.37	0.39	0.33
ECOLI	0.32	**0.35**	0.32	0.34	0.31
HUMAN	0.33	0.29	0.45	0.53	**0.56**
MOUSE	0.35	0.29	0.44	0.46	0.47
RAT	0.4	0.35	0.47	0.47	**0.5**
s1	ARATH	ECOLI	HUMAN	MOUSE	RAT
ARATH	**0.64**	0.37	0.35	0.35	0.32
ECOLI	0.31	**0.38**	0.29	0.29	0.29
HUMAN	0.37	0.32	0.5	0.56	**0.57**
MOUSE	0.38	0.31	0.47	**0.49**	0.49
RAT	0.41	0.36	0.51	0.52	**0.56**

5. Funkcionalne kategorije i neuređenost proteina

5.1 Funkcionalne kategorije proteina - COG klasifikacija

Jedan način određivanja funkcije proteina je na osnovu pripadnosti jednoj od funkcionalnih kategorija proteina zadatih u okviru COG[1] baze proteina (eng. ***C****luster of* ***O****rthologous* ***Gr****oups*). Klasteri grupa ortologa (COGovi) formirani su na osnovu evolutivne povezanosti gena koji kodiraju proteine date u izvornom skupu od 136 711 proteina iz 50 bakterijskih genoma, 13 arhejskih genoma i 3 jednoćelijska eukariotska genoma [63]. Kao i svi organizmi, i ovi organizmi su evolutivno povezani, a time i geni iz njihovih genoma, kao i proteini koje kodiraju. Za gene u različitim organizmima koji imaju sličnu primarnu sekvencu smatra se da su potekli od pra-gena njihovog zajedničkog pretka, pra-organizma. Ovakvi geni se nazivaju ortolozima i obavljaju istu funkciju u ćeliji. Pojednostavljeno, klasteri su konstruisani na osnovu pretpostavke da ukoliko su 3 proteina sličnija međusobno nego sa ostalim proteinima, onda će oni vršiti istu funkciju i stoga sačinjavati *grupu ortologa* [83]. Sličnost dve proteinske sekvence određivana je na osnovu `gapped BLAST` algoritma [1]. Na taj način, dobijena su 4873 osnovna klastera (COG-a) koji su dalje klasterovani u 26 funkcionalnih kategorija a oni u 4 funkcionalne grupe:

1. ćelijski procesi (eng. *cellular processes*), koju čine:

   D - Kontrola ćelijskog ciklusa i mitoze
   M - Biogeneza ćelijskog zida
   N - Ćelijska pokretljivost
   O - Post-translaciona modifikacija, promet proteina, šaperonske funkcije
   T - Prenos signala

[1] http://www.ncbi.nlm.nih.gov/COG/

U - Međućelijski promet i sekrecija
Y - Nuklearna struktura
Z - Citoskelet
V - Odbrambeni mehanizmi
W - Vanćelijske strukture

2. skladištenje i obrada informacija (eng. *information storage and processing*), koju čine:

   A - RNK obrada i modifikacija
   B - Hromatinska struktura i dinamika
   J - Translacija, ribozomalna struktura i biogeneza
   K - Transkripcija
   L - DNK replikacija, rekombinovanje i popravke

3. metabolizam, koju čine:

   C - Proizvodnja i konverzija energije
   E - Metabolizam i transport aminokiselina
   F - Metabolizam i transport nukleotida
   G - Metabolizam i transport ugljenih hidrata
   H - Metabolizam koenzima
   I - Metabolizam lipida
   P - Transport i metabolizam neorganskih jona
   Q - Biosinteza, transport i katabolizam sekundarnih metabolita

4. nedovoljno okarakterisani proteini (eng. *poorly characterised*), koju čine:

   R - Predviđena samo opšta funkcija
   S - Nepoznata funkcija

Svaka funkcionalna kategorija sadrži više klastera od kojih svaki predstavlja posebnu funkciju. Specijalno, jedan klaster može potpadati pod jednu ili više funkcionalnih kategorija. Na primer, funkcionalna kategorija A sadrži između ostalih i sledeće klastere, od kojih pretposlednji klaster pripada dvema funkcionalnim kategorijama:

- COG0430 - pripada funkcionalnoj kategoriji A
- COG1949 - pripada funkcionalnoj kategoriji A
- COG4085 - pripada funkcionalnoj kategoriji A
- COG5180 - pripada funkcionalnoj kategoriji A
- COG5186 - pripada funkcionalnoj kategoriji A
- COG5238 - pripada funkcionalnim kategorijama A i T
- COG5239 - pripada funkcionalnoj kategoriji A

Na ovaj način je uspostavljen opis funkcije proteina preko klastera grupa ortologa. Novosekvencioniranim proteinima može biti dodeljena funkcionalna kategorija korišćenjem programa Cognitor [83], čime se COG baza podataka periodično dopunjava. Postoje i drugi alati za automatsko predviđanje COG funkcionalnih grupa (npr. [88]). Za potrebe ovog istraživanja korišćena je verzija COG baze podataka od 20. novembra 2009. godine.

5.2 Neuređenost proteina

Sa porastom broja proteina za koje je eksperimentalno utvrđena sekundarna struktura postalo je očigledno da značajan broj proteina, pod određenim fiziološkim uslovima, ne poseduje dobro definisanu, uređenu 3D strukturu. Različiti termini se trenutno koriste za opisivanje ovakvih proteina: prirodno/ suštinski neuređeni, nesavijeni, denaturisani proteini ili reomorfni proteini (eng. *intrinsically disordered/ unfolded/ unstructured*). U ovom radu biće korišćen samo kraći termin neuređeni proteini.

Neuređeni proteini mogu biti u celosti bez strukture ili se mogu sastojati iz uređenih i neuređenih regiona raznih dužina. Proteini za koje je do sada eksperimentalno utvrđeno da su neuređeni prikupljeni su u bazi podataka DisProt [2] i u verziji *DisProt Release 6.02* od 24.5.2013. godine nalazi se 694 takvih proteina [68] iz različitih organizama.[3]

[2] http://www.disprot.org

[3] Pored DisProt baze proteina, postoje i druge baze proteina koje sadrže informacije o eksperimentalno utvrđenoj neuređenosti, npr. PED (*eng. Protein Ensemble Database*, http://pedb.vib.be/)

Na osnovu podataka dobijenih eksperimentalno i automatskim predviđanjem neuređenosti, predloženo je više klasifikacija neuređenih regiona na osnovu dužine. Na primer, po jednoj podeli regioni se dele na kratke (dužine od 4 do 30 aminokiselina), duge (dužine od 31 do 200 aminokiselina) i vrlo duge (dužine preko 200 aminokiselina) [55], po drugoj su regioni razvrstani u 5 grupa (dužine 1-3, 4-15, 16-30, 31-100 i preko 100 aminokiselina) [56]. Neki autori prilikom svojih analiza neuređenosti razdvajaju proteine koji sadrže neuređene regione duže od 30, 40 i 50 aminokiselina [39], zatim duže od 1,11,21,31 i 41 aminokiselina [54], itd. Proteini koji su potpuno neuređeni predstavljaju posebnu klasu proteina i takođe mogu biti različitih dužina.

Do sada ne postoji opšteprihvaćena precizna definicija neuređenosti proteina. Oni se mogu pojavljivati u različitim oblicima, počev od sasvim nestruktuiranog nasumičnog klupka, preko pre-topljive globule do topljive globule. Svako od ovih stanja može biti prirodno stanje proteina, odnosno stanje relevantno za njegovo obavljanje funkcije u ćeliji. Neki neuređeni proteini mogu preći u uređeno stanje ili obrnuto, prilikom reakcije sa drugim molekulima, dok drugi ostaju stabilno neuređeni tokom svojih aktivnosti.

Na nivou primarne strukture, neuređene proteine karakteriše niska kompleksnost sekvence, odnosno da se sastoje iz kratkih segmenata koji se ponavljaju, zatim veća zastupljenost polarnih i šaržiranih aminokiselina kao i odsustvo velikih hidrofobnih i aromatičnih aminokiselina. Korišćenjem TOP-IDP skale, zasnovane na osobinama aminokiselina kao što su hidrofobnost, polarnost, zapremina itd. predloženo je sledeće rangiranje zastupljenosti aminokiselina u različito struktuiranim regionima, od uređenih ka neuređenim: Trp, Phe, Tyr, Ile, Met, Leu, Val, Asn, Cys, Thr, Ala, Gly, Arg, Asp, His, Gln, Lys, Ser, Glu, Pro [18].

Eksperimentalno, neuređenost se kod proteina može utvrditi pomoću preko 20 različitih biofizičkih i biohemijskih tehnika od kojih su neke kristalografija difrakcijom X-zraka, heteronuklearna multidimenzionalna nuklearna magnetna rezonanca, cirkularni dihroizam, optička rotaciona disperzija, infra crvena spektroskopija sa Furijeovom transformaciom, Ramanova optička aktivnost, itd. Eksperimentalno izučavanje neuređenih proteina je vrlo teško s obzirom da oni ne poseduju jedinstvenu strukturu u izolovanom stanju [99, 101], zbog čega je razvijen veliki broj alata za automatsko predviđanje neuređenosti proteina [31].

Programi za automatsko predviđanje neuređenih regiona kod proteina mogu se na osnovu principa rada podeliti na dve grupe [26, 31, 46, 99]:

1. prediktori koji su zasnovani na fizičko-hemijskim osobinama aminokiselina u proteinima (PONDR familija prediktora neuređenosti koja između ostalih uključuje VL-XT, VL3, VSL1 i VSL2, FoldUnfold, PreLINK, IUPred, GlobProt, FoldIndex, itd.)

2. prediktori koji su zasnovani na poravnanjima sličnih proteinskih sekvenci (RONN, DISOPRED)

Neke od javno dostupnih baza neuređenosti proteina gde je neuređenost utvrđena automatski, primenom prediktivnih alata su:

- D^2P^2 (eng. ***D**atabase of **D**isordered **P**rotein **P**redictions*)[4] koja sadrži predikcije neuređenosti za proteine iz 1765 kompletnih proteoma [49]. Za svaki protein prikazani su rezultati nekoliko prediktora neuređenosti: VL-XT, VSL2b, PrDOS, PV2, Espritz i IUPred.

- MobiDB [58] koja sadrži informacije o tome da li je neuređenost eksperimentalno utvrđena, kojim metodama kao i predikcije neuređenosti dobijene na osnovu 10 različitih prediktora [5]

Taksonomski, neuređeni proteini su zastupljeni u proteomima sva tri superkraljevstva (Arheje, Bakterije i Eukarije) i to daleko od zanemarljivog broja: još su rezultati prvih istraživanja pokazali da najmanje 25% sekvenci u bazi podataka *SwissProt* sadrži disorder regione dužine ≥ 40 [61, 62].

Funkciju neuređenih proteina prvi je istraživao Danker [12]. Analiza je izvršena nad više od 150 proteina sa neuređenim regionima dužim od 30 aminokiselina, pod prirodnim uslovima, iz različitih vrsta organizama. Pretragom tada dostupne literature, identifikovano je 28 različitih biohemijskih funkcija za 98 od 115 neuređenih regiona, među kojima su vezivanje između dva proteina, vezivanje između proteina i nuklenskih kiselina, modifikacija proteina itd. Na osnovu aktivnosti koju neuređeni proteini obavljaju, predložena je njihova podela na bar 4 klase:

1. molekulsko prepoznavanje

2. molekulsko sastavljanje i rastavljanje

3. modifikacije proteina

4. aktivnosti entropijskog lanca, odnosno aktivnosti zavisne od fleksibilnosti, savitljivosti i plastičnosti kičme [12, 68, 98, 101]

[4] http://d2p2.pro/
[5] http://mobidb.bio.unipd.it//

Za neke neuređene proteine eksperimentalno je pokazana povezanost sa raznim savremenim oboljenjima kao što su karcinom i neurodegenerativne bolesti. Pored toga, bioinformatičkim analizama otkriveno je da su mnogi ovakvi proteini relevantni kod dijabetesa i kardiovaskularnih bolesti [19, 33, 70, 102].

5.3 Odnos neuređenosti i pozicije u proteinu

Ispitivanje neuređenosti proteina na njegovim krajevima i u sredini prethodno su sprovodile istraživačke grupe Lija [44] i Lobanova [45]. Prva grupa [44] je podelila proteinski lanac na 3 dela: terminalne delove dužine po 15 aminokiselina i središnji deo. Za trening skup je uzeto 197 proteina iz PDB (*Protein Data Bank*[6]) baze proteina, sa ciljem da otkrivena saznanja primene u izgradnji prediktora sekundarne strukture proteina. Testirane su tri metode za predikciju na svakom od 3 dela proteina i svaka metoda je pokazala da je neuređenost veća na terminalnim delovima nego u sredini. Kod druge grupe [45], proteinski lanac je podeljen na sličan način, s tim što su terminalni delovi činili po 30 aminokiselina od svakog kraja, a srednji deo ostatak proteina. Na skupu sačinjenom od 28 727 jedistvenih proteinskih struktura iz PDB baze proteina pokazano je da je udeo neuređenih aminokiselina u terminalnim delovima proteina veći nego udeo neuređenih aminokiselina u celom proteinu, dok obrnuto važi za srednji deo: udeo uređenih aminokiselina u srednjem delu proteina veći je nego udeo uređenih aminokiselina u celom proteinu. Dobijeni zaključci poslužili su u izgradnji FoldUnfold [28] prediktora za određivanje neuređenosti proteina na osnovu njegove primarne sekvence.

Prethodna istraživanja su u ovoj disertaciji proširena na analizu DisProt baze proteina u odnosu na prisustvo neuređenih regiona u krajnjim delovima (N-kraj i C-kraj) i u središnjem delu proteinskog lanca. Dodatno, ispitana je raspodela aminokiselina u svakom delu (u svakom od 3 dela proteina, da li se neka aminokiselina češće pojavljuje u uređenom ili u neuređenom regionu), kao i veza između neuređenih aminokiselina u svakom od delova i njihovih fizičko-hemijskih karakteristika (u svakom od 3 dela proteina, da li se u neuređenim regionima češće pojavljuju aminokiseline sa određenim fizičko-hemijskim osobinama).

[6] http://www.pdb.org

Tabela 5.1: *Distribucija aminokiselina u različitim delovima proteinskog lanca.*

deo proteina	% svih AA	% neuređenih AA
N-deo (30 AA od N-terminala)	5.62	8.45
C-deo (30 AA od C-terminala)	5.62	8.85
oba terminalna dela	11.23	17.30
M-deo (sve ostale aminokiseline)	88.77	82.70

U ovom istraživanju korišćen je DisProt skup proteina, različit u odnosu na skupove proteina iz prethodnih radova po broju proteina, prosečnoj dužini proteinskog lanca i po prosečnom broju neuređenih regiona po proteinu. Značajnu razliku predstavlja i činjenica da je DisProt baza proteina sačinjene od proteina kod kojih je neuređenost detektovana različitim eksperimentalnim metodama.

Udeo svih aminokiselina i neuređenih aminokiselina u različitim delovima proteinskog lanca

Da bismo ispitali da li je neuređenost više zastupljena u krajevima proteinskog lanca ili u njegovom središnjem delu, uporedili smo udeo svih (uređenih i neuređenih) aminokiselina u N-, C- i središnjem delu u odnosu na ceo lanac sa udelom neuređenih aminokiselina u odgovarajućim delovima u odnosu na ceo lanac. Kao što je pokazano u tabeli 5.1, udeo neuređenih aminokiselina u terminalnim delovima proteina je viši od udela svih aminokiselina u ovim delovima proteina, što je konzistentno sa rezultatima Lobanova [45].

Udeo neuređenih aminokiselina u različitim delovima proteinskog lanca

Da bismo detaljnije ispitali neuređenost u krajevima, izvršili smo dodatnu podelu terminalnih delova proteina. I N-deo i C-deo podeljeni su na trećine, čime smo dobili podelu na 7 delova: N1-10, N11-20, N21-30, središnji M-deo,

Tabela 5.2: *Procenat neuređenih aminokiselina u različitim delovima proteinskog lanca.*

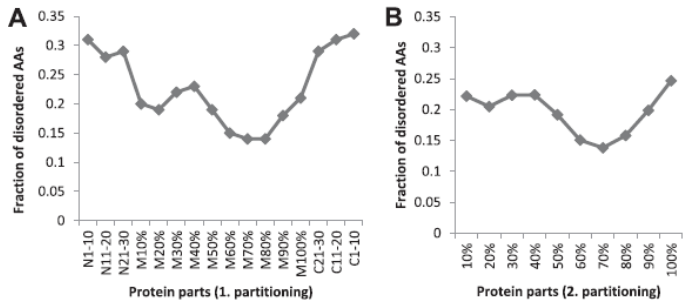
deo proteina	procenat neuređenih AA
N1-10	31.4
N11-20	27.69
N21-30	29.19
M	18.23
C21-30	29.46
C11-20	30.6
C1-10	32.42

C21-30, C11-20, C1-10. Tabela 5.2 prikazuje procenat neuređenih aminokiselina u svakom od ovih delova. Možemo primetiti da je razlika između susednih delova N21-30 i M, kao i M i C21-30 prilično visoka, čak 11% u oba slučaja, što može ukazivati na činjenicu da raspodela procenata neuređenosti duž M-dela nije uniformna. Da bismo to ispitali, predložili smo još dve podele proteinskog lanca i analizirali procenat neuređenosti na svakoj od njih. Prva podela podrazumevala je razdvajanje M-dela na manje delove od kojih svaki čini po 10% njegove dužine (N1-10, N11-20, N21-30, M10%,. . . , M100%, C21-30, C11-20, C1-10). U drugoj podeli se ukidaju terminalni delovi (N-deo i C-deo) i ceo proteinski lanac se deli na desetine (10%,. . . , 100%).

Procenat neuređenih aminokiselina po delovima proteina za obe podele prikazan je na slici 5.1. Na prvoj podeli između susednih delova N21-30 i M10% kao i M100% i C21-30 razlike su manje i iznose 7% i 8%, redom, dok razlike između susednih delova u središnjem delu ne prelaze 4% (slika 5.1A). U drugoj podeli najveća razlika između susednih delova iznosi 5%, između dela 90% i dela 100% (slika 5.1B).

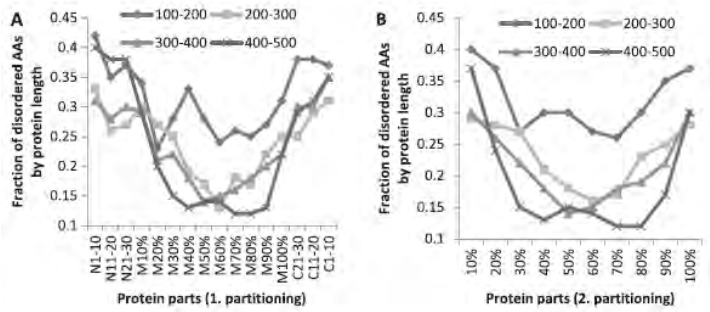
Udeo neuređenih aminokiselina u različitim delovima proteinskog lanca za proteine različitih dužina

S obzirom da dužina proteina DisProt baze varira od 92 do 18543, analizirali smo procenat neuređenih aminokiselina u proteinima grupisanim po njihovim dužinama. Na slici 5.2 prikazani su rezultati za obe podele, za 4 grupe pro-



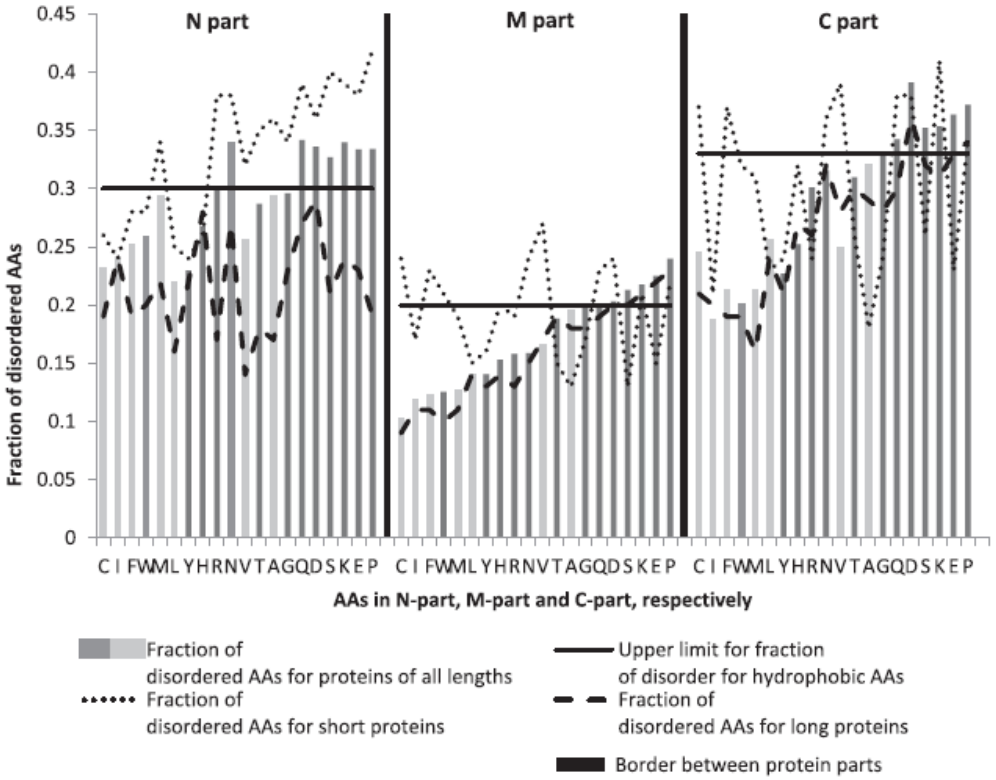
Slika 5.1: *Procenat neuređenih aminokiselina po delovima proteina za dve podele. A - podela (N1-10, N11-20, N21-30, M10%,. . . , M100%, C21-30, C11-20, C1-10), B - podela (10%,. . . , 100%).*

teina: proteini dužine od 100 do 200, zatim od 201 do 300, od 301 do 400 i 401 do 500.



Slika 5.2: *Procenat neuređenih aminokiselina za proteine različitih dužina za dve podele. A - podela (N1-10, N11-20, N21-30, M10%,. . . , M100%, C21-30, C11-20, C1-10), B - podela (10%,. . . , 100%).*

Za proteine dužine između 100aa i 500aa možemo primetiti strmiji pad procenta neuređenosti od N-dela ka središnjim delovima (5.2A) nego kod proteina svih dužina (5.1A). Štaviše, razlika između nekih susednih delova u M-delu je viša u poređenju sa proteinima svih dužina i iznosi 11% (na primer za proteine dužine 100-200, između delova M10% i M20%). U drugoj podeli evidentna je veća razlika na prelazima iz terminalnih u središnje delove 5.2B u poređenju sa svim proteinima (5.1B).



Slika 5.3: *Procenat neuređenih aminokiselina za proteine različitih dužina za dve podele. A - podela (N1-10, N11-20, N21-30, M10%,..., M100%, C21-30, C11-20, C1-10), B - podela (10%,..., 100%).*

Stepen neuređenosti za svaku od 20 aminokiselina

Da bismo svakoj aminokiselini dodelili meru neuređenosti u proteinskom lancu, uvodimo pojam *stepena neuređenosti*:

$$stepen\_neuredjenosti\_AA\_i\_u\_delu\_j = \frac{broj\_pojavljivanja\_AA\_i\_u\_neuredjenom\_regionu\_u\_delu\_j}{ukupan\_broj\_pojavljivanja\_AA\_i\_u\_delu\_j}$$

Za svaku aminokiselinu izračunat je stepen neuređenosti u svim delovima proteinskog lanca. Uz to, ispitivali smo i vezu hidrofobnosti svake aminokiseline prema Kajt-Dulitlovoj skali [84] i njihovog stepena neuređenosti. U ovoj analizi korišćena je osnovna podela proteinskog lanca na N-deo (30 aminokiselina od N-terminala), C-deo (30 aminokiselina od C-terminala) i M-deo (ostatak proteinskog lanca).

Tabela 5.3: *Nova skala zasnovana na stepenu neuređenosti u srednjem delu proteina u poređenju sa odgovarajućom skalom aminokiselina prikazanom u radu Lobanova [45].*

pozicija u novoj skali	1	2	3	4	5	6	7	8	9	10
aminokiseline	C	I	F	W	M	L	Y	H	R	N
pozicija na skali Lobanova	5	3	2	1	8	6	4	9	12	13
pozicija u novoj skali	11	12	13	14	15	16	17	18	19	20
aminokiseline	V	T	A	G	Q	D	S	K	E	P
pozicija na skali Lobanova	7	11	10	19	15	16	20	17	18	14

Slika 5.3 prikazuje stepen neuređenosti za svaku aminokiselinu posebno, u svakom od 3 dela. Uz to, stepen neuređenosti je označen tamnijim stubićem u slučaju da je aminokiselina hidrofilna a svetlijim u suprotnom. Redosled aminokiselina je u svakom delu isti i predstavlja redosled dobijen sortiranjem stepena neuređenosti za svih 20 aminokiselina u srednjem delu. Očigledno je da sve aminokiseline imaju veći stepen neuređenosti u srednjem delu u odnosu na terminalne delove. Dalje, u svakom delu možemo identifikovati gornju granicu stepena neuređenosti za hidrofobne aminokiseline. U srednjem delu, sve hidrofobne aminokiseline imaju stepen neuređenosti ispod 0.2 dok neke hidrofilne (P, Q, D, K, E i S) imaju stepen neuređenosti viši od 0.2 a ostale (W, Y, R, N, H i T) niži. U N-delu, sve hidrofobne aminokiseline imaju stepen neuređenosti ispod 0.3 dok neke hidrofilne (P, Q, D, K, E, S, R i N) imaju stepen neuređenosti viši od 0.3 a ostale (W, Y, H i T) niži. Konačno, u C-delu sve hidrofobne aminokiseline imaju stepen neuređenosti ispod 0.33 dok neke hidrofilne (P, Q, D, K, E, i S) imaju stepen neuređenosti viši od 0.33 a ostale (W, Y, H R, N i T) niži, isto kao u srednjem delu.

Na osnovu vrednosti prikazanih na slici 5.3, konstruisali smo skalu aminokiselina na osnovu njihovog stepena neuređenosti u srednjem delu proteinskog lanca. Dobijena skala je slična skali koju je predstavio Lobanov, na osnovu iste analize ali na različitom skupu [45]. Tabela 5.3 predstavlja novu skalu (2. red), pozicije aminokiselina u novoj skali (1. red) i pozicuju svake aminokiseline na skali Lobanova (3. red). Možemo videti da je razlika pozicija u dve skale za većinu aminokiselina ≤ 3 (osim za C, V, G i P).

5.4 Odnos funkcionalnih kategorija i neuređenosti

Jedan od ciljeva ove disertacije je i utvrđivanje odnosa neuređenosti i funkcije proteina. Analize su izvršene na skupovima proteina prokariotskih organizama (superkraljevstva arheja i bakterija) za koje je funkcija određena na osnovu pripadnosti odgovarajućem klasteru u COG sistematizaciji proteinskih funkcija (Clusters of Orthologous Groups), o kojoj je bilo reči u poglavlju 5.2. Skup proteina sačinjavali su proteini 25 arhejskih organizama iz 3 filuma i 271 bakterijskih organizama iz 17 filuma za koje je bila poznata COG anotacija. Svi proteini preuzeti su iz javno dostupne baze podataka američkog Nacionalnog centra za biotehnološke informacije (NCBI[7]). Za određivanje neuređenih regiona u proteinima korišćen je VSL2b prediktor.

Neuređenost skupa proteina posmatrana je kroz 4 različite mere:

1. prosečan broj neuređenih regiona po proteinu
2. prosečan broj neuređenih regiona na 100 aminokiselina
3. broj proteina koji sadrže bar jedan neuređeni region
4. prosečan procenat aminokiselina u neuređenim regionima u odnosu na ceo lanac

Svaka od ovih mera izračunata je za svaki COG i svaku funkcionalnu kategoriju za oba superkraljevstva kao i pojedinačno po filumima i organizmima.

Prosečan broj neuređenih regiona po proteinu

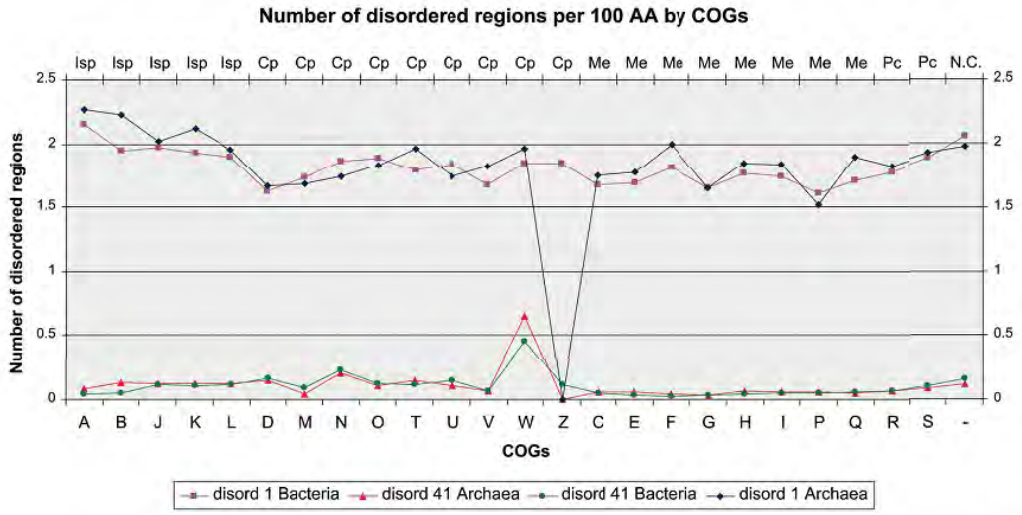
Najveći prosečni broj neuređenih regiona po proteinu sadrže proteini u funkcionalnoj grupi ćelijskih procesa (Cp, COGovi: D, M, N, O, T, U, V, W), zatim u funkcionalnoj grupi za skladištenje i obradu informacija (Isp, COGovi: A, B, J, K, L), potom u funkcionalnoj grupi za metabolizam (Me, COGovi: C, E, F, G, H, I, P, Q) i na kraju u funkcionalnoj grupi proteina sa slabo karakterisanom funkcijom (Pc, COGovi: R, S). Proteini koji nisu dodeljeni nijednom COGu imaju nizak broj neuređenih regiona bilo koje dužine. Najveći prosečan broj neuređenih regiona dužina $L \geq 11, 21, 31, 41$ aminokiselina, po proteinu, u najvećem broju filuma nalazi se u COGovima N(Cp) i

[7] http://www.ncbi.nlm.nih.gov/

L(Isp).

Prosečan broj neuređenih regiona na 100 aminokiselina

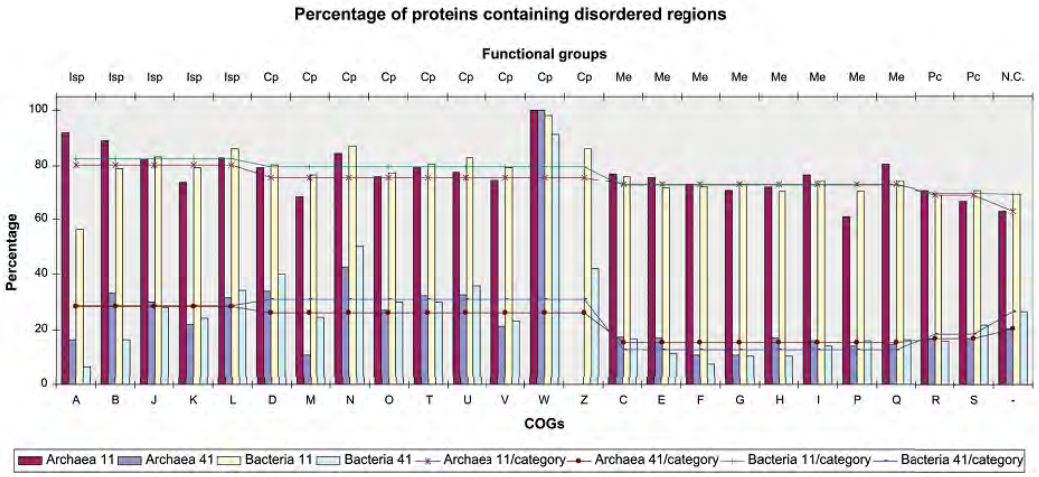
S obzirom da dugi proteini mogu imati više neuređenih regiona nego kratki, uvedena je mera koja bi neutralizovala ovaj efekat. Prosečan broj neuređenih regiona na 100 aminokiselina dobija se tako što se ukupan broj neuređenih regiona podeli sa $L/100$, pri čemu je L dužina proteina. Pritom, posebno je računat broj neuređenih regiona proizvoljne dužine kao i broj neuređenih regiona dužine ≥ 41, jer se ovoliko dugi regioni mogu u strogom slučaju smatrati regionima prave neuređenosti, koji nisu slučajni. Rezultati po COG-ovima prikazani su na slici 5.4. Za bakterije, prosečan broj neuređenih regiona neograničene dužine na 100 aminokiselina po COG-ovima iznosi 1.82 sa standardnom devijacijom 0.13, dok su odgovarajuće vrednosti za arheje, 1.88, 0.18, redom. COG-ovi koji odstupaju od prosečne vrednosti su W i N kod bakterija i jednočlani COG W kod arheja. Proteini metaboličke funkcionalne grupe i po ovoj meri pokazuju najmanju neuređenost, dok su proteini grupe Cp i Isp na prvom i drugom mestu u odnosu na broj neuređenih regiona.



Slika 5.4: *Prosečan broj neuređenih regiona na 100 aminokiselina. COG-ovi su poređani po funkcionalnim grupama kojima pripadaju. Prikazane su vrednosti za regione dužine ≥ 1 i ≥ 41, za oba superkraljevstva.*

Procenat proteina koji sadrže bar jedan neuređeni region

Gledano po superkraljevstvima, procenat proteina koji sadrže bar jedan neuređeni region dužine $L \geq 1, 11, 21, 31, 41$ iznosi redom 99.9%, 71%, 43%, 30% i 20% za arheje i 99.9%, 74%, 46% , 32% i 22% za bakterije. Distribucija po COG-ovima za regione dužine $L = 11, 41$ predstavljena je na slici 5.5. Linijama su prikazane vrednosti ove mere po funkcionalnim kategorijama COG-ova (Me, Isp, Cp, Nc).

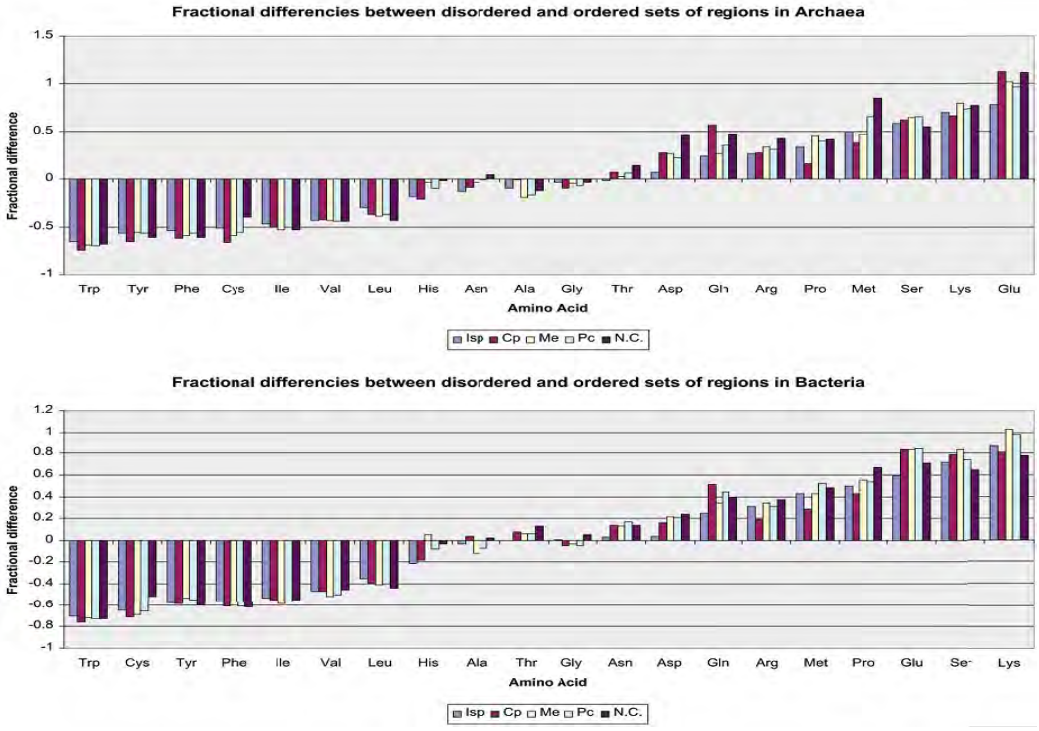


Slika 5.5: *Procenat proteina koji sadrže bar jedan neuređeni region, po COG-ovima i po funkcionalnim grupama. Predstavljeni su neuređenu regioni dužine $L \geq 11, 41$. Vrednosti za COG-ove su predstavljene stubićima a za funkcionalne kategorije linijama.*

Molske frakcije i frakcijske razlike

Za svaku aminokiselinu izračunata je molska frakcija po COG-ovima proteina za oba superkraljevstva kao i frakcijska razlika između neuređenih i uređenih regiona COG-ova. Molska frakcija za aminokiselinu j ($j = 1, \ldots, 20$) u skupu proteinskih sekvenci (jednog COG-a ili jedne funkcionalne kategorije), definiše kao $P_j = \frac{\sum n_i * P_{ji}}{\sum n_i}$, gde je n_i dužina i-te sekvence a P_{ji} učestalost pojavljivanja j-te aminokiseline u i-toj sekvenci. Frakcijska razlika računata je po formuli $\frac{P_j(a) - P_j(b)}{P_j(b)}$, gde je $P_j(a)$ molska frakcija j-te aminokiseline u skupu predviđenih neuređenih regiona za proteine iz datog COG-a, a $P_j(b)$ odgovarajuća molska frakcija za uređene regione istog COG-a.

Na slici 5.6 predstavljene su frakcijske razlike za aminokiseline po funkcionalnim kategorijama COG-ova, za arheje i bakterije. Aminokiseline su uređene rastuće po vrednosti frakcijske razlike u funkcionalnoj kategoriji Isp koja pokazuje najveću vrednost frakcijske razlike od svih funkcionalnih kategorija. Negativna vrednost frakcijske razlike za neku aminokiselinu odgovara nižem nivou neuređenosti, a pozitivna višem nivou neuređenosti. Rezultati pokazuju da je u oba superkraljevstva neuređenost niska u najvećem broju takozvanih *order-promoting* aminokiselina (koje promovišu uređenost), a visoka kod *disorder-promoting* aminokiselina. Rezultati drugih autora [18, 59, 71, 103] takođe prate ovu tendenciju. Frakcijska razlika za aminokiselinu asparagin (Asn) je nekonzistentna kod arheja i bakterija, dok su *disorder-promoting* aminokiseline glicin (Gly) i alanin (Ala) nešto više uređene. Rastuće uređenje aminokiselina u odnosu na frakcijsku razliku je blisko Vajonenovoj skali fleksibilnosti [104].



Slika 5.6: *Frakcijske razlike između skupa uređenih i skupa neuređenih regiona po funkcionalnim grupama COG-ova za superkraljevstva arheja i bakterija. Na x-osi predstavljene su aminokiseline; za datu aminokiselinu, na y-osi prikazana je frakcijska razlika za odgovarajuću funkcionalnu kategoriju.*

Redosled COG-ova i njihovih funkcionalnih kategorija u odnosu na različite mere neuređenosti

Analizirane mere neuređenosti proteina u pojedinačnim COG-ovima i u funkcionalnim kategorijama COG-ova dale su različite ali uporedive rezultate koji mogu biti predstavljeni kroz rastući redosled COG-ova po neuređenosti, zasnovano na ovim merama. Možemo primetiti da se redosled za neuređene regione neograničene dužine značajno razlikuje od ostanih i stoga je manje pouzdan od uređenja zasnovanih na dužim regionima.

Redosledi COG-ova u rastućem poretku, po različitim kriterijumima, za neuređene regione dužine od $L \geq 41$ ("-" je oznaka za proteine koji nisu dodeljeni nijednom COG-u):

1. Rastući redosled prosečnog broja neuređenih regiona po proteinu:
   arheje: F,G,M,Q,P,I,E,H,C,R,A,S,-,V,K,J,O,B,U,L,D,T,N,W
   bakterije: F,A,G,H,E,I,B,C,P,R,Q,S,V,K,J,M,-,O,T,L,U,D,Z,N,W

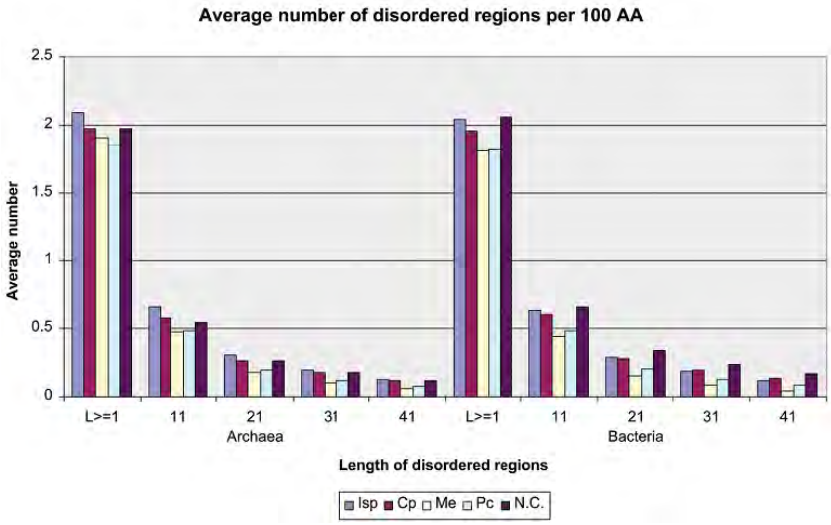
2. Rastući redosled prosečnog broja neuređenih regiona na 100 aminokiselina:
   arheje: G,M,F,Q,P,E,I,C,H,R,V,A,S,O,U,K,L,-,J,B,T,D,N,W
   bakterije: F,G,E,A,H,I,C,B,P,Q,R,V,M,K,S,T,J,Z,L,O,U,D,-,N,W

3. Rastući redosled procenta proteina koji sadrže bar jedan neuređeni region:
   arheje: G,M,F,P,Q,I,A,E,R,S,H,C,-,V,K,O,J,L,T,U,B,D,N,W
   bakterije: A,F,G,H,E,I,P,R,B,Q,C,S,V,K,M,-,J,T,O,L,U,D,Z,N,W

4. Rastući redosled prosečnog procenta aminokiselina u neuređenim regionima:
   arheje: G,M,F,Q,E,I,P,V,C,H,R,A,S,O,J,K,B,U,-,L,T,D,N,W
   bakterije: F,G,E,H,I,B,P,C,Q,A,V,R,M,K,J,L,S,T,Z,O,U,-,N,D,W

Opšti zaključak analize neuređenosti po COG-ovima je da COG-ovi funkcionalne kategorije Isp koji sadrže veliki broj proteina (J,K,L), kao i većina COG-ova funkcionalne kategorije Cp (D, V, T, M, N, Z, U, O) pokazuju viši nivo neuređenosti od prosečnog, dok metabolički COG-ovi (C, G, E, F, H, I, P, Q) pokazuju niži nivo neuređenosti od proseka. Na slici 5.7 prikazan je ovaj odnos između COG-ova u funkcionalnim kategorijama Isp, Cp i Me, kada je kriterijum neuređenosti prosečni broj neuređenih regiona na 100 aminokiselina. Slični odnosi važe i za ostale kriterijume neuređenosti.

Na osnovu dobijenih rezultata uočava se veza između povećanog nivoa neuređenosti proteina i funkcija koje oni imaju. Tako, možemo reći da veću neuređenost pokazuju proteini koji obavljaju funkcije translacije, transkripcije, replikacije i popravke nukleinskih kiselina, kontrole ćelijskog ciklusa i ćelijske pokretljivosti, kao i funkcije prenosa signala. Sa druge strane, manje su neuređeni proteini koji učestvuju u proizvodnji i čuvanju energije u ćeliji, u formiranju sekundarne strukuture, kao i u metabolizmu i transportu aminokiselina, nukleotida, ugljenih hidrata, masti, koenzima i neorganskih jona.



Slika 5.7: *prosečni broj neuređenih regiona na 100 aminokiselina po dužinama regiona i kategorijama COG-ova. Na x-osi prikazani su regioni različitih dužina; visina stubića za datu dužinu regiona i datu funkcionalnu kategoriju COG-a odgovara prosečnom broju neuređenih regiona na 100 aminokiselina, date dužine i u datoj funkcionalnoj kategoriji.*

5.5 GO-anotirani proteini i neuređenost

Kao što je bilo reči u poglavlju 3.2, funkcija proteina se može predstaviti na različite načine. U ovom istraživanju, funkcija proteina analizirana je kroz reprezentaciju preko COG-funkcionalnih grupa i kategorija i preko GO ontologija. Iako su ovi sistemi predstavljanja funkcija međusobno različiti, postoji jedinstveno preslikavanje jednog dela COG funkcionalnih grupa u

odgovarajuće GO funkcije.[8]

U poglavlju 5.4 izvršena je analiza veze neuređenosti prokariotskih proteina i funkcionalnih kategorija kojima pripadaju. Sa ciljem da proverimo da li povezanost postoji i u slučaju kada je odabrano predstavljanje funkcije preko GO ontologije, iz skupa proteina korišćenog za treniranje prediktora funkcije proteina (poglavlje 3.6) izdvojeni su proteini koji su obavljali one GO funkcije za koje postoji preslikavanje ka odgovarajućim COG funkcionalnim grupama. S obzirom da su prethodna zapažanja bila utvrđena za arhejske i bakterijske organizme, iz detektovanog skupa proteina izdvojeni su proteini iz organizama ova dva superkraljevstva. Proteina iz arheja je bilo svega 3 pa je dalja analiza nastavljena samo za proteine iz bakterija, većinom iz popularnog model-organizma *E.Coli*. Većina izdvojenih proteina imala je više funkcija koje su pripadale različitim funkcionalnim kategorijama. Kako bismo izbegli šum, ograničili smo se na proteine koji obavljaju ili jednu funkciju ili više funkcija koje pripadaju istim COG funkcionalnim kategorijama. Na taj način je dobijeno 508 proteinskih sekvenci koje su dalje podeljene na 3 grupe prema funkcionalnim kategorijama kojima nakon preslikavanja pripadaju: Me, Cp i Isp. Nedovoljno okarakterisanih proteina (kategorija Pc) nije bilo u skupu izdvojenih proteina.

Kao i u analizi prokariotskih proteina, na skup izdvojenih proteina primenjen je VSL2b prediktor za utvrđivanje neuređenosti. Potom su na osnovu dobijenih informacija o neuređenim regionima izračunate 4 mere neuređenosti uvedene u poglavlju 5.4:

1. prosečni procenat neuređenih aminokiselina po proteinu
2. prosečni broj neuređenih regiona po proteinu
3. prosečni broj neuređenih regiona na 100 aminokiselina po proteinu
4. broj proteina koji sadrže bar jedan neuređeni region

Računanja su, analogno prethodnoj analizi nad prokariotama, sprovedene za neuređene regione dužine $L \geq 1$ i $L \geq 41$. Dobijeni rezultati prikazani su u tabeli 5.4.

Možemo primetiti da proteini Isp funkcionalne kategorije pokazuju višu, a proteini Me funkcionalne kategorije nižu neuređenost od prosečne po sve 4 mere, što se poklapa sa rezultatima iz analize predstavljene u poglavlju 5.4. Proteini iz funkcionalne kategorije Cp pokazuju višu neuređenost od prosečne

[8] http://geneontology.org/external2go/cog2go

Tabela 5.4: *Neuređenost GO-anotiranih proteina.*

L¿=1	#proteina	mera1	mera2	mera3	mera4
Me	374	21.16	5.56	2.07	1
Cp	111	23.88	6.45	2.09	1
Isp	23	41.66	5.04	2.48	1
prosečno	169.33	22.68	5.73	2.09	1
L¿=41	#proteina	mera1	mera2	mera3	mera4
Me	374	5.33	0.24	0.07	0.18
Cp	111	7.1	0.34	0.11	0.27
Isp	23	21.41	0.52	0.38	0.48
prosečno	169.33	6.44	0.27	0.09	0.22

po svim merama, ali su za sve 4 mere vrednosti bliže manje neuređenim, Me proteinima, nego više neuređenim, Isp proteinima, što je u suprotnosti sa rezultatima dobijenim u prethodnoj analizi, gde je nivo neuređenosti Cp proteina visok kao i kod Isp proteina. Razlike u rezultatima mogu se opravdati različitim skupovima podataka: u ovoj analizi skup podataka dosta manji nego u prethodnoj i u ovoj analizi proteini uglavnom potiču iz jednog organizma (*E.coli*) dok su u prethodnoj iz raznih bakterijskih organizama.

Gledano po pojedinačnim funkcijama, na osnovu preslikavanja između funkcija GO ontologije i COG funkcionalnih kategorija, možemo uvideti da su više neuređeni proteini sa GO funkcijama translacije, transkripcije, ribozomalni proteini, kao i oni koji učestvuju u udvajanju, rekombinaciji i popravci DNK (preslikani u Isp kategoriju), kao i da su manje uređeni proteini sa GO funkcijama odgovornim za metabolizam i transport aminokiselina, nukleinskih kiselina i koenzima, proizvodnju i čuvanje energije u ćeliji i biosintezu različitih jedinjenja (preslikani u Me kategoriju). Proteini zaduženi za ćelijsku deobu i particionisanje hromozoma, ćelijsku pokretljivost i mehanizme signalne transdukcije preslikani su u Cp kategoriju i pokazuju veću neuređenost od prosečne, ali ipak bližu proteinima manje neuređene Me kategorije nego više neuređene Isp kategorije.

6. Zaključak

U ovom radu predstavljeno je istraživanje problema funkcije proteina iz dva različita aspekta. Glavni doprinos prvog aspekta sastoji se u definisanju nove metode za automatsko predviđanje funkcije proteina na osnovu njegove primarne sekvence korišćenjem metoda strukturne klasifikacije, konkretno metode strukturalnih podržavajućih vektora (eng. *Structured Support Vector Machines*, skraćeno SSVM), evaluaciji rezultata koji se postižu primenom ove metode kao i njihovo poređenje sa aktuelnim metodama predviđanja funkcije proteina. Da bismo prilagodili metodu konkretnom problemu bilo je, između ostalog, neophodno konstruisati algoritam za računanje maksimuma funkcije po svim mogućim konzistentnim podgrafovima za datu ontologiju, što s obzirom na veličinu ontologije i ukupan broj konzistentnih podgrafova nije bilo moguće rešiti eksplicitnom enumeracijom. Uz to, posebna pažnja posvećena je ispitivanju da li poreklo proteina nad kojima je treniran klasifikacioni model ima uticaja na njegove performanse i očekivano je pokazano da je odgovor potvrdan. Fokus u ovom istraživanju bile su i različite funkcije gubitka, gde smo došli do zaključka da su "starije" mere (Žakardovo rastojanje i $1 - f_1$) u ovoj primeni bolje od novouvedenih mera normalizovanog semantičkog rastojanja (s_1 i s_2). Ovaj rezultat bio je iznenađujući, s obzirom da s_1 i s_2 nose veću količinu informacije o sličnosti dva grafa nego preostale dve mere. Razlog zbog čega se u većini modela nisu pokazale jednako dobro kao Žakardovo rastojanje i $1 - f_1$ ostaje otvoreno pitanje. Jedan deo rezultata prikazanog dela istraživanja u glavi 3 predstavljen je u radovima [37, 79] izloženim na konferencijama, a u pripremi je rad sa rezultatima eksperimenata sa različitim reprezentacijama proteinskih sekvenci.

Važno je napomenuti da je SSVM metoda moćan alat sa širokom primenom koja prevazilazi bioinformatički domen prikazan u ovom radu. Na primer, jedna od oblasti u kojoj se SSVM može primeniti je hijerarhijska klasifikacija tekstova. Za pojedine korpuse tekstova postoje drvolike hijerarhije gde svaki list predstavlja kategoriju teksta, unutrašnji čvorovi su opštije kategorije a svaka putanja od korena ontologije do lista predstavlja jednu

klasu. Razvijeni su prediktivni modeli za nekoliko jezika (srpski, arapski, kineski i engleski) i rezultati su objavljeni u radu [30], prihvaćeni za štampu u radu [78] i u postupku recenzije u radu [29] .

Drugi aspekt istraživanja ove disertacije podrazumevao je analizu neuređenosti proteina u odnosu na njihove funkcionalne kategorije kao i u odnosu na poziciju neuređenih regiona u proteinu. Glavni doprinos ovog dela istraživanja predstavljaju skale neuređenosti pojedinačnih aminokiselina dobijene na različitim skupovima proteina: na velikom skupu prokariotskih proteina gde je neuređenost određena predikcijom i na malom skupu proteina iz raznih organizama gde je neuređenost određena eksperimentalno. Dobijeni rezultati mogu doprineti unapređenju metoda za automatsko predviđanje funkcije proteina, analizi interakcija neuređenih i uređenih regiona proteina i mogućoj zameni neuređenih regiona malim molekulima leka. Rezultati ovog dela istraživanja objavljeni su u radovima [77] (poglavlje 5.3) i [54] (poglavlje 5.4).

Planovi za dalji rad na polju predviđanja funkcije proteina uključuju:

- proširenje skupa korišćenih proteina na nove proteine koji su postali dostupni u javnim bazama podataka;
- eksperimentisanje sa različitim reprezentacijama podataka, pre svega vektora $\boldsymbol{x}$ (proteinske sekvence);
- izradu jedinstvenog modela koji bi bio treniran nad svim proteinima, uz mogućnost da se informacija o genetskom poreklu kodira unutar vektora Ψ;
- ugradnju mogućnosti da za dati test primer izlaz iz prediktora budu težine iz intervala $[0, 1]$ pridružene svakom čvoru ontologije, gde bi se konačan graf dobijao odbacivanjem svih čvorova težine manje od unapred određenog parametra;
- unapređenje predloženog algoritma za nalaženje maksimuma, pre svega njegovo ubrzanje.

Literatura

[1] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and Lipman Dj. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Res.*, 25:3389–3402, 1997.

[2] Rives A.W. and Galitski T. Modular organization of cellular networks. In *Proceedings of the National Academy of Sciences of the United States of America*, page 1128–1133, 2003.

[3] O. Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic Acids Research*, 32(Database issue):267–270, 2004.

[4] P. Bork and E. V. Koonin. Predicting functions from protein sequences - where are the bottlenecks? *Nature Genetics*, 18:313–318, 1998.

[5] W. Clark. *Understanding protein function through statistical inference and evolutionary analysis*. PhD thesis, School of Informatics and Computing, Indiana University, Indiana, USA, 2013.

[6] W. Clark and P. Radivojac. Information-theoretic evaluation of predicted ontological annotations. *Bioinformatics*, 29(13):53–61, 2013.

[7] W. T. Clark and P. Radivojac. Analysis of protein function and its prediction from amino acid sequence. *Proteins, Structure, Function and Bioinformatics*, 79(7):2086–2096, 2011.

[8] M. Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, 2002.

[9] J. C. Costello and G. Stolovitzky. Seeking the wisdom of crowds through challenge-based competitions in biomedical research. *Clin Pharmacol Ther*, 93(5):396–398, 2013.

[10] Cozzetto D. and Jones DT. The contribution of intrinsic disorder prediction to the elucidation of protein function. *Curr Opin Struct Biol.*, 23(3):467–72, 6 2013.

[11] D. Devos and A. Valencia. Practical limits of function prediction. *Proteins*, 41(1):98–107, 2000.

[12] A. K. et al Dunker. Intrinsic disorder and protein function. *Biochemistry*, 41(21):6573–6582, 2002.

[13] A. Vazquez et al. Global protein function prediction from protein-protein interaction networks. *Nature Biotechnology*, 21(6):697–700, 2003.

[14] B. E. Engelhardt et al. Protein molecular function prediction by Bayesian phylogenomics. *PLoS Computational Biology*, 1(5):45, 2005.

[15] B. Rost et al. Automatic prediction of protein function. *Cellular and Molecular Life Sciences*, 60:2637–2650, December 2003.

[16] B.Taskar et al. Max-margin markov networks. *Advances in Neural Information Processing Systems*, 2004.

[17] C. Huttenhower et al. A scalable method for integration and functional analysis of multiple microarray datasets. *Bioinformatics*, 22(23):2890–2897, 2006.

[18] Campen A. et al. Top- idp-scale: A new amino acid scale measuring propensity for intrinsic disorder. *Protein and Pept Lett*, 15(9):956–963, 2008.

[19] Cheng Y. et al. Abundance of intrinsic disorder in protein associated with cardiovascular disease. *Biochemistry*, 45:10448–10460, 2006.

[20] D. Bandyopadhyay et al. A structure-based function inference using protein family-specific fingerprints. *Protein Science*, 15(6):1537–1543, 2006.

[21] D. Martin et al. Gotcha: a new method for prediction of protein function assessed by the annotation of seven genomes. *BMC bioinformatics*, 5:1, 2004.

[22] E. M. Marcotte et al. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285(5428):751–753, 1999.

[23] E. Nabieva et al. Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. *Bioinformatics*, 21:302–310, 2005.

[24] F. Enault et al. Phydbac "gene function predictor": a gene annotation tool based on genomic context analusis. *BMC Bioinformatics*, 6:257, 2005.

[25] F. Minneci et al. Ffpred 2.0: Improved homology-independent prediction of gene ontology terms for eukaryotic protein sequences. *PLoS One*, 8(5), 2013.

[26] Ferron F. et al. A practical overview of protein disorder prediction methods. *PROTEINS: Structure, Function, and Bioinformatics*, 65:1–14, 2006.

[27] G. Bartlett et al. Infering protein function from structure. *Methods Biochem Anal.*, 44:387–407, 2003.

[28] Galzitskaya O. V. et al. Foldunfold: web server for the prediction of disordered regions in protein chain. *Bioinformatics*, 22:2948–9, 2006.

[29] Graovac J. et al. Hierarchical vs. flat n-gram-based text categorization: can we do better? (submitted).

[30] Graovac J. et al. Language independent n-gram-based text categorization with weighting factors: A case study. *Journal of Information and Data Management*, 6, 2015.

[31] He B. et al. Predicting intrinsic disorder in proteins: an overview. *Cell Research*, 19:929–949, 2009.

[32] I. Lee et al. A probabilistic functional network of yeast genes. *Science*, 306(5701):1555–1558, 2004.

[33] Iakoucheva L. M. et al. Intrinsic disorder in cell-signaling and cancer-associated proteins. *J Mol Biol*, 323(3):573–584, 2002.

[34] J. C. Costello et al. Gene networks in drosophila melanogaster: integrating experimental data to predict gene function. *Genome Biology*, 10(9):97, 2009.

[35] J. C. Hermann et al. Structure-based activity prediction for an enzyme of unknown function. *Nature*, 448(7155):775–779, 2007.

[36] Jiang Y. et al. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. (submitted).

[37] J.Kovacevic et al. On protein function prediction methods. In *Theoretical Approaches to BioInformation Systems-Book of Abstracts, Belgrade, Serbia, 17-22 September 2013.*, 2013.

[38] J.Lafferty et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, 2001.

[39] K. Dunker et al. Intrinsic protein disorder in complete genomes. *Genome Informatics*, 11:161–171, 2000.

[40] L. J. Jensen et al. Prediction of human protein function from post-translational modifications and localization features. *Journal of Molecular Biology*, 319(5):1257–1266, 2002.

[41] L. J. Jensen et al. Prediction of human protein function according to gene ontology categories. *Bioinformatics*, 19(5):635–642, 2003.

[42] L. M. Schriml et al. Disease ontology: a backbone for disease semantic integration. *Nucleic Acids Research*, 40(1):940–946, 2012.

[43] Lee H. et al. Diffusion kernel-based logistic regression models for protein function prediction. *OMICS: J Integr Biol*, 10:40–55, 2006.

[44] Li X. et al. Predicting protein disorder for n-, c- and internal regions. In *Genome Informormatics. Workshop on Genome Informatics*, pages 30–40, 1999.

[45] Lobanov M. Y. et al. Statistical analysis of unstructured amino acid residues in protein structures. *Biochemistry (Mosc)*, (75):192–200, 2010.

[46] Longhi S. et al. Conformational disorder. *Methods Mol Biol*, 609:307–325, 2010.

[47] M. Ashburner et al. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[48] M. Deng et al. Prediction of protein function using protein-protein interaction data. *Journal of Computational Biology*, 10(6):947–960, 2003.

[49] M. E. Oates et al. D2p2: Database of disordered protein predictions. *Nucleic Acids Research*, 41:508–516, 2013.

[50] M. Pellegrini et al. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. In *Proceedings of the National Academy of Sciences of the United States of America*, pages 4285–4288, 1999.

[51] Mateos A. et al. Systematic learning of gene functional classes from dna array expression data by using multilayer perceptrons. *Genome Research*, 12:1703–1715, 2002.

[52] O. G. Troyanskaya et al. A bayesian framework for combining heterogeneous data sources for gene function prediction (in saccharomyces cerevisiae). In *Proceedings of the National Academy of Sciences of the United States of America*, pages 8348–8353, 2003.

[53] P. Gaudet et al. Phylogenetic-based propagation of functional annotations within the gene ontology consortium. *Briefings in Bioinformatics*, 12(5):449–462, 2011.

[54] Pavlovic-Lazetic G. et al. Bioinformatics analysis of disordered proteins in prokaryotes. *BMC Bioinformatics*, 12(66), 2011.

[55] Peng K. et al. Optimizing long intriscic disorder predictors with protein evolutionary information. *J Bioinform and Comput Biol*, 3(1):35–60, 2005.

[56] Peng K. et al. Length-dependent prediction of protein intrinsic disorder. *BMC Bioinformatics*, 7(208):1–17, 2006.

[57] Pereira-Leal J.B. et al. Detection of functional modules from protein interaction networks. *Proteins*, 54:49–57, 2004.

[58] Potenza E. et al. Mobidb 2.0: an improved database of intrinsically disordered and mobile proteins. *Nucleic Acids Research*, pages 477–480, 2014.

[59] Radivojac P. et al. Intrinsic disorder and functional proteomics. *Biophys J*, 92(5):1439–1456, 2007.

[60] Radivojac P. et al. A large-scale evaluation of computational protein function prediction. *Nature Methods*, pages 221–227, 2013.

[61] Romero P. et al. Intelligent data analysis for protein disorder prediction. In *Proc IEEE Int Conf on Neural Networks: Houston TX 1*, pages 90–95, 1997.

[62] Romero P. et al. Thousands of proteins likely to have long disordered regions. In *Proc Pacific Symposium on Biocomputing*, pages 435–446, 1998.

[63] R.Tatusov et al. The cog database: a tool for genome-scale analysis of protein functions and evolution. *Nucleic Acids Research*, 28(1):33–36, 2000.

[64] S. F. Altschul et al. Basic local alignment search tool. *J. Mol. Biol*, 215(3):403–410, 1990.

[65] S. Hennig et al. Automated gene ontology annotation for anonymous sequence data. *Nucleic Acids Research*, 31(13):3712, 2003.

[66] Segal E. et al. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, 19:264–272, 2003.

[67] Shiga M. et al. Annotating gene function by combining expression data with a modular gene network. *Bioinformatics*, 23:468–478, 2007.

[68] Sickmeier M. et al. Disprot: the database of disordered proteins. *Nucleic Acids Res.*, 2007(35):786–93, Jan 2006.

[69] T. Joachims et al. Cutting-plane training of structural svms. *Machine Learning*, 77:27–59, 2009.

[70] Uversky V. N. et al. Intrinsically disordered proteins in human diseases: Introducing the d2 concept. *Ann Rev Biophys Biomol Structure*, 37:215–246, 2008.

[71] Vacic V. et al. Composition profiler: a tool for discovery and visualization of amino acid composition differences. *BMC Bioinformatics*, 8:211, 2007.

[72] Y. A. Kourmpetis et al. Bayesian markov random field analysis for protein function prediction based on network data. *PloS One*, 5(2):92–93, 2010.

[73] Y. A. Kourmpetis et al. Gene ontology consistent protein function prediction: the falcon algorithm applied to six eukaryotic genomes. *Algorithms for Molecular Biology*, 8(1):10, 2013.

[74] Y. Loewenstein et al. Protein function annotation by homology-based inference. *Genome Biology*, 10(2):207, 2009.

[75] Y.Altun et al. Support vectore machine learning for interdependant and structured output spaces. In *Predicting Structured Data*. MIT Press, 2007.

[76] M. Y. Galperin and E. V. Koonin. Sources of systematic error in functional annotation of genomes: domain rearrangement, non-orthologous gene displacement, and operon disruption. *In Silico Biology*, 1:55–67, 1998.

[77] Kovacevic J. Computational analysis of position-dependent disorder content in disprot database. *Genomics, Proteomics, Bioinformatics*, 10:158–165, 2012.

[78] Kovacevic J. and Graovac J. N-gram based text classification on serbian language based on structural support vector machines method. (in press).

[79] J.Kovacevic and G.Pavlovic-Lazetic. Predictive models based on support vector machinesfor structured outputs. In *Mathematical Data Science-Book of Abstracts, Belgrade, Serbia, 22. 6. 2015.*, 2015.

[80] M. Johnson. Pcfg models of linguistic tree representations. *Computational Linguistics*, 24(4):613–632, 1998.

[81] E. Kelley. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8(4):703–712, 1960.

[82] D. Koller and N. Friedman. *Probabilistic Graphical Models*. The MIT Press, Cambridge, MA., 2009.

[83] E. V. Koonin. *The Clusters of Orthologous Groups (COGs) Database: Phylogenetic Classification of Proteins from Complete Genomes*. NCBI handbook, 2003. chapter 22.

[84] J. Kyte and R. F. Doolittle. A simple method for displaying the hydropathic character of a protein. *J Mol Biol*, 157:105–132, 1982.

[85] R. A. Laskowski and J. M. Thornton. Understanding the molecular machinery of genetics through 3d structures. *Nature Review Genetics*, 9(2):141–151, 2008.

[86] S. Letovsky and S. Kasif. Predicting protein function from protein/protein interaction data: a probabilistic approach. *Bioinformatics*, 19:197–204, 2003.

[87] M. Levandowsky and D. Winter. Distance between sets. *Nature*, 234(5):34–35, 1971.

[88] Ulfeta Marovac. *Istraživanje obrazaca u određivanju karakteristika proteina*. PhD thesis, Matematički fakultet Univerziteta u Beogradu, 2015.

[89] D. Pal and D. Eisenberg. Inference of protein function from protein structure. *Structure*, 13(1):121–130, 2005.

[90] D. Pal and D. Eisenberg. Inference of protein function from protein structure. *Structure*, 13(1):121–130, 2005.

[91] F. Pazos and M. J. Sternberg. Automated prediction of protein function and detection of functional sites from structure. In *Proceedings of the National Academy of Sciences of the United States of America*, pages 14754–14759, 2004.

[92] P. Radivojac. *A not so Quick Introduction to Protein Function Prediction*. 2013.

[93] M. Riley. Functions of the gene products of escherichia coli. *Microbiol Rev*, 57:862–952, 1993.

[94] S. C. Rison, T. C. Hodgman, and Thornton Jm. Comparison of functional annotation schemes for genomes. *Funct Integr Genomics*, pages 56–69, 1 2000.

[95] P. N. Robinson and S. Mundlos. The human phenotype ontology. *Clinical Genetics*, 77(6):525–534, 2010.

[96] S. Sarawagi and R. Gupta. Accurate max-margin training for structured output spaces. In *Proceedings of the 25th International Conference on Machine Learning*, page 888–895. ACM, 2008.

[97] A. Sokolov and A. Ben-Hur. Hierarchical classifcation of gene ontology terms using the gostruct method. *Journal of Bioinformatics and Computational Biology*, 8(2):357–376, 2010.

[98] P. Tompa. Intrinsically unstructured proteins. *Trends Biochem Sci*, 27(10):527–533, 2002.

[99] P. Tompa and D. Kovacs. Intrinsically disordered chaperones in plants and animals. *Biochem Cell Biol*, 88:167–174, 2010.

[100] I. et al Tsochantaridis. Large margin methods for structured and interdependent output variables. *Machine Learning Research*, pages 1453–1484, 2005.

[101] V. N. Uversky. Natively unfolded proteins: a point where biology waits for physics. *Protein Sci*, 11:739–756, 2002.

[102] V. N. Uversky. Intrinsic disorder in proteins associated with neurodegenerative diseases. *Front Biosci (Landmark Ed)*, 1(14):5188–238, Jun 2009.

[103] V. N. Uversky and A. K. Dunker. Understanding protein non-folding. *Biochim Biophys Acta - Proteins and Proteomics*, 1804(6):1231–1264, 2010.

[104] M. Vihinen. Relationship of protein flexibility to thermostability. *Protein Eng*, 1:477–480, 1987.

[105] M. N. Wass and M. J. Sternberg. Confunc-functional annotation in the twilight zone. *Bioinformatics*, 24(6):798–806, 2008.

[106] Lesk AM. Whisstock JC. Prediction of protein function from protein sequence and structure. *Q Rev Biophys 2003*, 36:307–340, 2003.

[107] Chen Y. and Xu D. Global protein function annotation through mining genome-scale data in yeast saccharomyces cerevisiae. *Nucleic Acid Research*, 32:6414–6424, 2004.

[108] G. Zehetner. Ontoblast function: From sequence similarities directly to potential functional annotations by ontology terms. *Nucleic acids research*, 31(13):3799, 2003.

[109] T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *Annals of Statistics*, 32(1):56–85, 2004.

Biografija

Jovana Kovačević je rođena 6.5.1983. godine u Beogradu. Završila je Petu beogradsku gimnaziju i diplomirala na Matematičkom fakultetu, smer Računarstvo i informatika. Nakon završenih osnovnih studija, 2007. godine upisala je doktorske studije na istom smeru. Od iste godine angažovana je na Matematičkom fakultetu kao saradnik u nastavi, a od 2009. godine kao asistent. Do sada je držala vežbe iz 7 računarskih predmeta.

Osnovna oblast interesovanja joj je istraživanje podataka u bioinformatici. Član je Bioinformatičke istraživačke grupe. Pored toga, sarađuje i sa istraživačkom grupom prof. Predraga Radivojca na Fakultetu za informatiku i računarstvo na Univerzitetu Indijana, u Blumingtonu, SAD, gde je 2013. godine bila u istraživačkoj poseti. Učestvovala je u radu više međunarodnih radionica i letnjih škola posvećenih raznim bioinformatičnim temama. Objavila je veći broj naučnih radova i učestvovala na nekoliko međunarodnih i domaćih konferencija.

Прилог 1.

Изјава о ауторству

Потписани-а Јована Ковачевић

број индекса 2026/2009

Изјављујем

да је докторска дисертација под насловом

„Структурна предикција функције протеина и однос функционалних категорија протеина и њихове неуређености"

- резултат сопственог истраживачког рада,
- да предложена дисертација у целини ни у деловима није била предложена за добијање било које дипломе према студијским програмима других високошколских установа,
- да су резултати коректно наведени и
- да нисам кршио/ла ауторска права и користио интелектуалну својину других лица.

Потпис докторанда

У Београду, 10.11.2015.

Ковачевић

Прилог 2.

Изјава о истоветности штампане и електронске верзије докторског рада

Име и презиме аутора Јована Ковачевић

Број индекса 2026/2009

Студијски програм Информатика

Наслов рада „Структурна предикција функције протеина и однос функционалних категорија протеина и њихове неуређености"

Ментор проф. др Гордана Павловић-Лажетић

Потписани/а Јована Ковачевић

Изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предао/ла за објављивање на порталу **Дигиталног репозиторијума Универзитета у Београду.**

Дозвољавам да се објаве моји лични подаци везани за добијање академског звања доктора наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Универзитета у Београду.

Потпис докторанда

У Београду, 10.11.2015.

Прилог 3.

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

„Структурна предикција функције протеина и однос функционалних категорија протеина и њихове неуређености“

која је моје ауторско дело.

Дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигитални репозиторијум Универзитета у Београду могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио/ла.

(1.) Ауторство

2. Ауторство - некомерцијално

3. Ауторство – некомерцијално – без прераде

4. Ауторство – некомерцијално – делити под истим условима

5. Ауторство – без прераде

6. Ауторство – делити под истим условима

(Молимо да заокружите само једну од шест понуђених лиценци, кратак опис лиценци дат је на полеђини листа).

Потпис докторанда

У Београду, 10.11.2015.

Јовановић