UNIVERZITET U BEOGRADU

MATEMATIČKI FAKULTET

MASTER RAD

Utvrđivanje N-glikozilovanosti proteina metodama mašinskog učenja

Autor:
Aleksandar ANŽEL

Mentor:
doc. dr Jovana Kovačević

Članovi komisije:
dr Nevena Veljković
doc. dr Mladen Nikolić

Katedra za računarstvo i informatiku

Beograd, januar 2020.

Izjava o akademskoj čestitosti

Ja, Aleksandar ANŽEL, student master akademskih studija Matematičkog fakulteta Univerziteta u Beogradu, potvrđujem da je rad pod naslovom „Utvrđivanje N-glikozilovanosti proteina metodama mašinskog učenja" kao i čitav njegov sadržaj, u potpunosti moj. Takođe, potpisivanjem izjavljujem:

- da je rad isključivo rezultat mog sopstvenog istraživačkog rada;
- da sam rad i mišljenja drugih autora koje sam koristio u ovom radu naznačio ili citirao u skladu sa Uputstvom;
- da su svi radovi i mišljenja drugih autora navedeni u spisku literature/referenci koji su sastavni deo ovog rada i pisani u skladu sa Uputstvom;
- da sam dobio sve dozvole za korišćenje autorskih dela koja se u potpunosti/celosti unose u predati rad i da sam to jasno naveo;
- da sam svestan da je plagijat korišćenje tuđih radova u bilo kom obliku (kao citata, parafraza, slika, tabela, dijagrama, dizajna, planova, fotografija, filma, muzike, formula, veb sajtova, kompjuterskih programa i sl.) bez navođenja autora ili predstavljanje tuđih autorskih dela kao mojih, kažnjivo po zakonu (Zakon o autorskom i srodnim pravima, Službeni glasnik Republike Srbije, br. 104/2009, 99/2011, 119/2012), kao i drugih zakona i odgovarajućih akata Univerziteta u Beogradu;
- da sam da sam svestan da plagijat uključuje i predstavljanje, upotrebu i distribuiranje rada predavača ili drugih studenata kao sopstvenih;
- da sam svestan/na posledica koje kod dokazanog plagijata mogu prouzrokovati na predati master rad i moj status;
- da je elektronska verzija master rada identična štampanom primerku i pristajem na njegovo objavljivanje pod uslovima propisanim aktima Univerziteta.

Potpis:

Datum:

UNIVERZITET U BEOGRADU

Sažetak

Matematički fakultet
Katedra za računarstvo i informatiku

Master matematičar

Utvrđivanje N-glikozilovanosti proteina metodama mašinskog učenja

Aleksandar ANŽEL

U cilju obavljanja svojih funkcija, većina eukariotskih proteina podleže različitim posttranslacionim modifikacijama. Jedna od najvažnijih takvih modifikacija je N-vezana glikozilacija, koja predstavlja vezivanje ugljenih hidrata (glikana), pomoću glikozidnih veza, za molekule proteina. Promene procesa N-vezane glikozilacije mogu dovesti do razvitka mnogih bolesti kod različitih organizama. Utvrđivanje da li će protein uopšte podleći procesu N-glikozilacije (tj. biti N-glikozilovan) predstavlja prvi korak procesa nalaženja tačne pozicije njenog dešavanja na proteinu. Eksperimentalne, laboratorijske metode za određivanje N-glikozilovanosti proteina i nalaženja tačnog mesta dešavanja glikozilacije su skupe i vremenski zahtevne. Ova činjenica je dala motivaciju za razvoj nekoliko računarskih alata koji rešavaju problem nalaženja tačnog mesta dešavanja procesa, ali ne i onih koji rešavaju problem egzistencije procesa. Takođe, većina ovih alata je bazirana na skupu proteina čoveka, a vrlo malo na skupovima podataka drugih organizama. Poznato je da proces N-glikozilacije sadrži specifičnosti među različitim organizmima, te se javlja potreba za razvijanjem i u tom smislu specifičnih klasifikatora. U cilju rešavanja ovog problema, razvijeni su različiti modeli zasnovani na metodama mašinskog učenja čiji je cilj klasifikovanje proteina u zavisnosti od njegovog svojstva N-glikozilovanosti. Radom su izložene i tehnike prevazilaženja problema nebalansiranosti klasa korišćenog skupa podataka, kao i uspešnosti različitih klasifikatora.

Ključne reči: bioinformatika; mašinsko učenje; glikozilacija; N-vezana glikozilacija.

UNIVERITY OF BELGRADE

Abstract

Faculty of Mathematics
Department of Computer Science and Informatics

Master of Mathematics

Determining protein N-glycosylation with machine learning methods

Aleksandar ANŽEL

Most of protein functions are dependent on post-translational modifications (PTMs). One of the most often PTM in eukaryotes is N-linked glycosylation, which represents the process of attaching an oligosaccharide, sometimes also referred to as glycan, to a protein molecule. Changes in N-linked glycosylation have been associated with various diseases in different organisms. Determining whether a protein will be N-glycosylated or not is the first step of finding an accurate position of an attachment. Wet lab experiments for determining protein N-glycosylation and finding the attachment position are expensive and time-consuming. Several computational tools were created for automatic determinating of the exact position of N-glycosylation, but there are none that predict the existence of process. Furthermore, most existing tools are based on human or mammalian proteomes and very few are using protein data sets of other organisms. It is known that N-glycosylation has distinctive characteristics between different organisms, therefore an organism-specific tools are much in need. For this thesis, different machine learning classifiers were developed for determining protein N-glycosylation. Also, different techniques were used to overcome unbalanced data problem that is existent in used data set.

Keywords: bioinformatics; machine learning; glycosylation; N-linked glycosylation; data science.

Zahvalnica

Najsrdačniju zahvalnost dugujem svojoj mentorki doc. dr Jovani Kovačević na posvećenom vremenu, savetima, idejama i velikoj podršci tokom izrade master rada. Takođe joj se zahvaljujem i na povećanju inspiracije za nastavak školovanja i bavljenja naučno-istraživačkim radom u domenima bioinformatike.

Veliku zahvalnost dugujem i doc. dr Mladenu Nikoliću na korisnim sugestijama i plodonosnim savetima koji su doprineli rešavanju mnogih problema sa kojima sam se susreo, kao i savetima tokom kurseva koje mi je držao tokom osnovnih i master studija.

Zahvalnost dugujem i dr Neveni Veljković, na korisnim savetima i pomoći pri izradi rada.

Na kraju, zahvalnost dugujem i svojoj porodici na neizmernoj podršci i razumevanju tokom celokupnog školovanja.

Sadržaj

Slike

Tabele

Lista skraćenica

A	**A**denin
ADASYN	**A**daptive **S**ynthetic
AK	**A**mino-**k**iselina
AUC	eng. *A**U**C*
C	**C**itozin
CDMB	**C**entralna **d**ogma **m**olekularne **b**iologije
D^2P^2	eng. *D**P**P*
DisProt	eng. *D**P***
DNK	**D**ezoksi**r**ibo**n**ukleinska **k**iselina
EBI	eng. *E**B**I*
EMBL	eng. *E**M**B**L***
FN	eng. *F**N***
FP	eng. *F**P***
G	**G**uanin
MMDB	eng. *M**M**D**b***
MPV	**M**etod/**M**odel **P**otpornih **V**ektora
MU	**M**ašinsko **u**čenje
NCBI	eng. *N**C**B**I***
PDB	eng. *P**D**b***
PTM	**P**ost**t**ranslacione **m**odifikacije
PTP	**P**ost**t**ranslacioni **p**roces
RNK	**R**ibo**n**ukleinska **k**iselina
ROC	eng. *R**O**C***
SGD	lat. *S** eng. *G**D***
SMOTE	eng. *S**M**O**T***
T	**T**imin
TAIR	eng. *T**A**I**R***
TN	eng. *T**N***
TP	eng. *T**P***
U	**U**racil
UniProt	eng. *U**P***

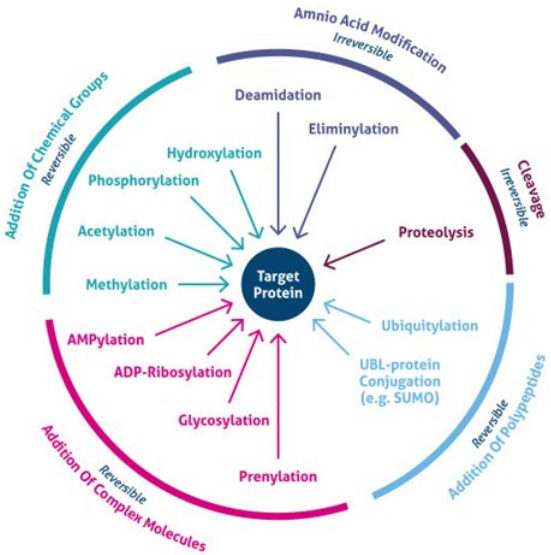
Mojim roditeljima i bratu...

Glava 1

Upoznavanje sa problemom

1.1 Posttranslacione modifikacije proteina

Proteini spadaju u najvažnije biomolekule i predstavljaju ključne komponente mnogih bioloških procesa. U cilju obavljanja svojih funkcija, većina proteina u ćelijama eukariota podleže kovalentnim posttranslacionim modifikacijama (skraćeno PTM). Kao što i samo ime implicira, ove kovalentne modifikacije se dešavaju nakon što se molekul DNK transkripcijom prevede u molekul RNK, a zatim molekul RNK translacijom prevede u protein [67]. PTM imaju uticaja kako na fizičke, tako i na hemijske osobine proteina, što direktno uslovljava načine savijanja proteina, njihovu konformaciju, aktivnost, kao i funkciju. To znači da one indirektno utiču na mnoge bolesti kod eukariotskih organizama, te je njihovo izučavanje izuzetno važno kako u teorijske, tako i u praktične svrhe. Uprkos tome što imaju ogromnu važnost u biološkim funkcijama, izučavanje PTM proteina veoma dugo stagnira usled nedostatka odgovarajućih metoda. Zbog toga pun značaj modifikacija proteina u vezi sa radom ćelije nije poznat [48]. Postoje više različitih PTM proteina, pri čemu ćemo se u ovom radu fokusirati na glikozilaciju i to poseban vid glikozilacije - N-vezanu glikozilaciju. Različite vrste PTM su ilustrovane slikom 1.1.



SLIKA 1.1: Različite vrste posttranslacionih modifikacija [59].

1.2 Motivacija

Poslednjih decenija, rešavanje mnogih bioloških problema se obavlja računarskim metodama koje zamenjuju tradicionalne, laboratorijske metode. Rešavanje problema eksperimentalnim putem je skoro uvek veoma težak i zahtevan proces. Pored toga što zahtevaju veliko iskustvo i stručnost, eksperimentalne metode su često i veoma skupe. Sa druge strane, nedavna eksplozija moći računara dovela je do toga da su se višestruko efikasnije računarske metode, po pouzdanosti rezultata, znatno približile eksperimentalnim.

Nekoliko ovakvih metoda je razvijeno upravo u cilju rešavanja problema koji se tiču PTM, pa samim tim i glikozilacije. Većina razvijenih alata rešavaju veoma opšte probleme, bazirajući se na ljudske proteine ili čak proteine čitavih klasa organizama. Postavlja se pitanje da li proces glikozilacije ima klasne specifičnosti, ili čak specifičnosti među samim organizmima? Pokazano je da je proces glikozilacije evolucijom razvio veliki broj uočljivih razlika među različitim organizmima, pa samim tim i između organizama istih taksonomskih nivoa [36], što daje motivaciju kreiranja alata specifičnih za pojedinačne organizme, tj. familije organizama.

Pored prethodnog, postojeći alati rešavaju problem nalaženja specifičnog mesta dešavanja procesa glikozilacije, na nekom proteinu. U trenutku pisanja ovog rada nema alata koji rešavaju opštiji problem: provera da li je protein uopšte podložan procesu glikozilacije ili ne. Rešavajući ovaj problem se u obzir uzimaju globalne osobine proteina. Samim tim se izbegava donošenje odluka na osnovu lokalnih proteinskih informacija (posmatrajući kratke sekvence aminokiselina) koje često mogu ispustiti bitne međuzavisnosti udaljenih delova tog proteina.

Ideja ovog rada predstavlja razvoj klasifikatora koji utvrđuju N-glikozilovanost proteina na osnovu njegovih fizičko-hemijskih osobina. U radu je korišćen skup proteina organizma *Arabidopsis Thaliana*, korovske biljke kratkog životnog veka koja se često koristi kao model u molekularnoj biologiji i genetici. Usled svojih specifičnih osobina, *Arabidopsis Thaliana* je postala jedna od najizučavanijih biljaka u biološkoj zajednici (zajedno sa kukuruzom i duvanom). Od svih biljaka, genom *Arabidopsis Thaliana* je prvi sekvenciran i intenzivno korišćen u različitim istraživanjima. Zbog svog kratkog genoma, a samim tim i malog skupa proteina koje on kodira, ovaj organizam se pokazao pogodnim za korišćenje u ovom radu.

1.3 Pregled koraka rešavanja problema

Sprovedeno istraživanje se sastojalo od sledećih koraka:

1. **Prikupljanje podataka**: Korišćeni podaci predstavljaju proteinske sekvence u FASTA formatu, preuzete iz *UniProt* baze podataka. Proteinske sekvence pripadaju organizmu *Arabidopsis Thaliana* i dobijaju se unošenjem odgovarajućeg upita bazi. Pored prethodnog, upitom su izabrani samo proteini koji su ekspertski pregledani, tj. provereni od strane stručnjaka. Proteini za koje se zna da podležu N-vezanoj glikozilaciji su takođe izdvojeni upitom (više o tome u 3.1.1), i čine skup pozitivnih podataka. Skup negativnih podataka koji predstavlja proteine za koje se zna da ne podležu N-vezanoj glikozilaciji, kreiran je izbacivanjem prethodnog pozitivnog skupa iz skupa svih ekspertski proverenih proteina ovog organizma.

2. **Preprocesiranje**: Pomoću ovih skupova se zatim pripremaju ulazne datoteke za metode mašinskog učenja. Za svaki protein se računaju njegove fizičko-hemijske osobine pomoću dostupnih biblioteka. Ulaz metoda mašinskog učenja predstavlja matrica, čiji redovi predstavljaju različite proteine, a kolone osobine svakog od tih proteina.

3. **Razvoj različitih modela**: Nakon obezbeđivanja ulaznih podataka, sledeći korak predstavlja razvoj modela zasnovanih na dva metoda mašinskog učenja: metod potpornih vektora i potpuno povezana neuronska mreža. Za oba modela se tokom validacije određuju vrednosti hiperparametara koje daju najbolje rezultate.

4. **Diskusija**: Na kraju se razmatraju prednosti i mane oba modela, kao i mogućnosti poboljšanja.

Šematski prikaz koraka rešavanja problema dat je dijagramom na slici 1.2 koji je kreiran veb alatom *draw.io* [29].

1.4 Pregled rada

Nakon uvodne prve glave, u drugoj glavi ovog rada su predstavljeni osnovni pojmovi potrebni za razumevanje podataka, metoda, kao i bioloških procesa koji se izučavaju. Na kraju poglavlja je dat i kratak pregled postojećih alata koji se bave istim ili sličnim problemom koji se izučava u ovom radu.

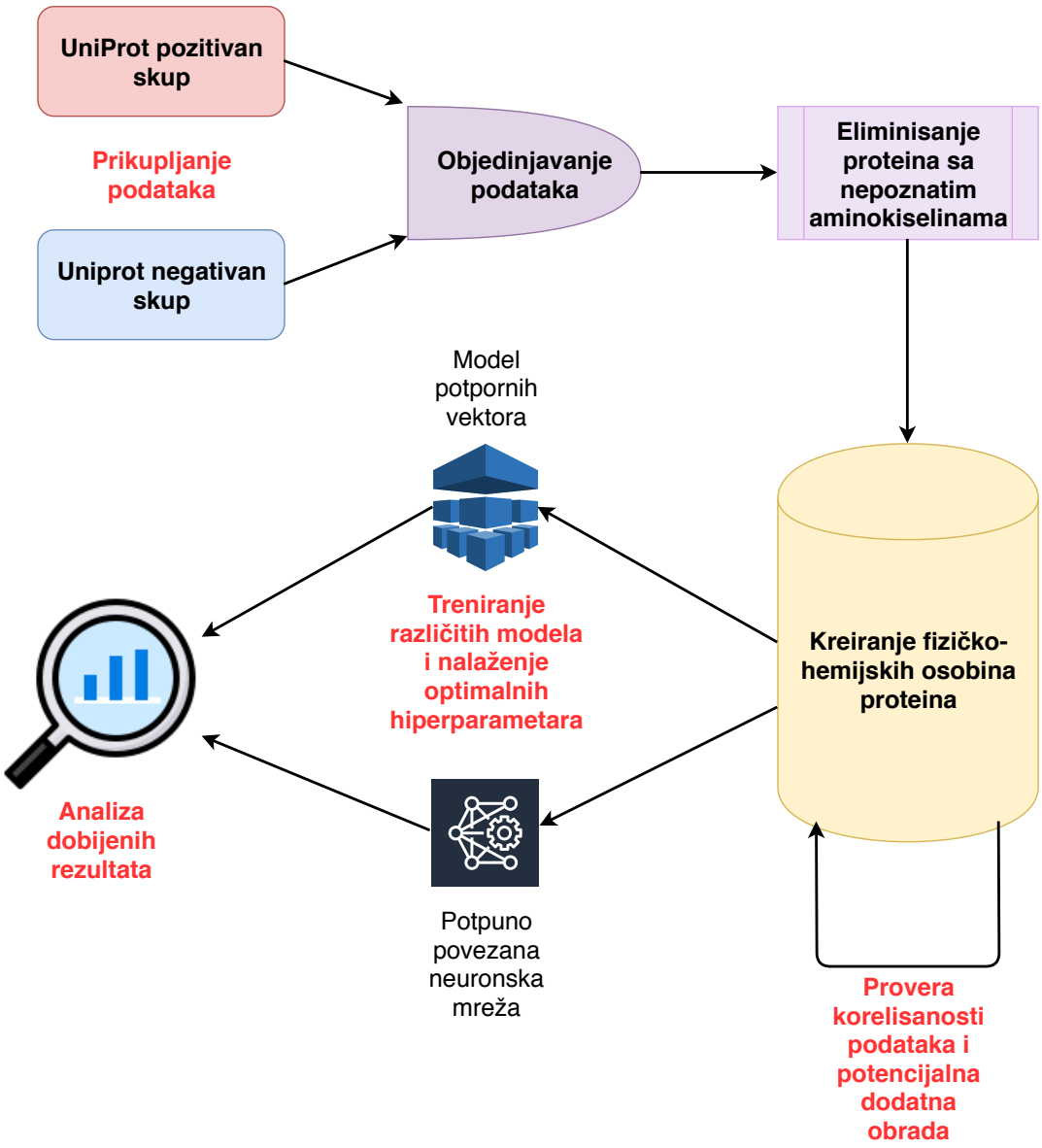
Treća glava daje detaljniji prikaz podataka koji su korišćeni kao i postupak njihovog prikupljanja. Zatim su sažeto navedeni alati koji su korišćeni za sprovođenje ideja rada, kao i metode koje su korišćene za rešavanje problema.

Četvrta glava opisuje rezultate različitih modela mašinskog učenja nad različitim skupovima podataka. Na kraju glave se nalazi i poglavlje o poređenju dobijenih rezultata sa rezultatima ranijih radova.

Peta glava sadrži diskusiju o mogućnostima poboljšanja rezultata.

Šesta glava sadrži zaključak na osnovu postignutih rezultata.

U dodatku A su izloženi prilozi rezultatima koji su prezentovani u četvrtoj glavi.



SLIKA 1.2: Koraci u rešavanju problema.

Glava 2

Osnovni pojmovi

2.1 Biomolekuli

Biomolekuli su supstance kreirane od strane ćelija i živih organizama. Predstavljaju molekule različitih veličina i oblika koji obavljaju niz esencijalnih funkcija za život jednog organizma. Četiri glavna tipa biomolekula su **ugljeni hidrati**, **lipidi**, **nukleinske kiseline** i **proteini**.

Nukleinske kiseline, DNK i RNK, izgrađene su od nukleotida čiji raspored određuje sekvence aminokiselina koje čine proteine. Prethodno opisan tok prenosa genetskih informacija unutar jednog organizma se naziva centralnom dogmom molekularne biologije (više u narednom poglavlju).

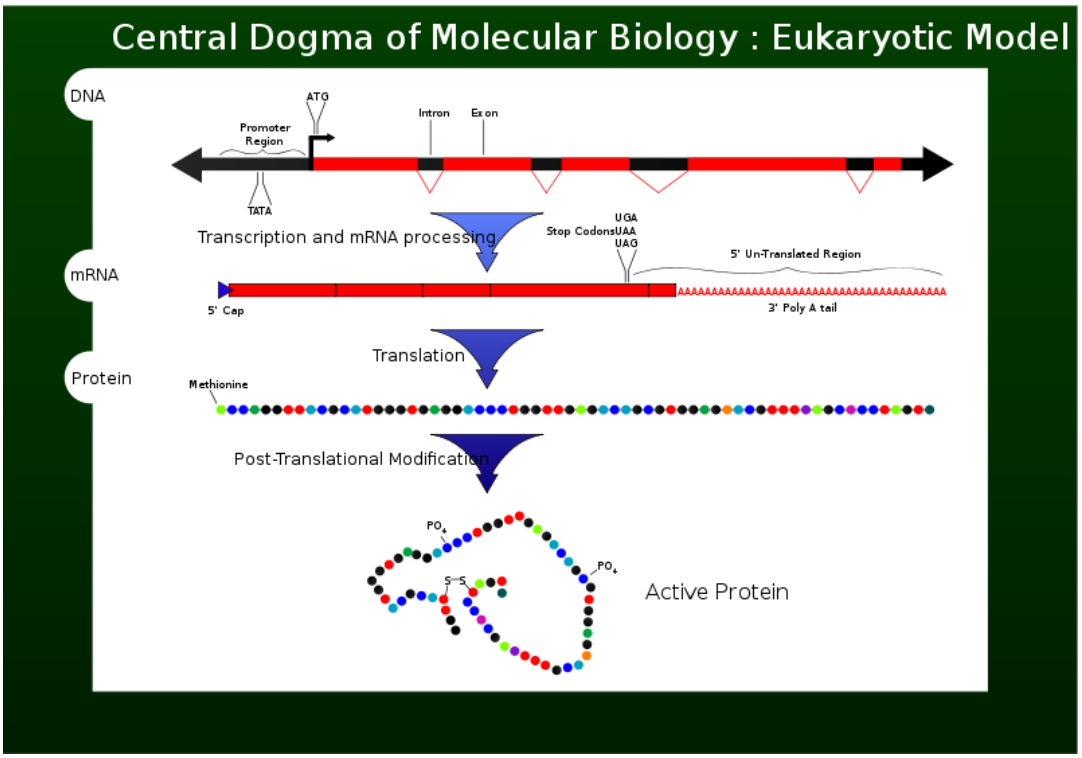
Sa druge strane, ugljeni hidrati su sačinjeni od molekula koji sadrže atome ugljenika, vodonika i kiseonika. Kao takvi, predstavljaju jedinstvene energetske izvore i strukturne komponente čitavog živog sveta i jedni su od najrasprostranjenijih biomolekula na Zemlji. Prema stepenu složenosti dele se na: monosaharide, disaharide, oligosaharide i polisaharide.

Lipidi, još jedna grupa ključnih biomolekula, imaju mnoštvo različitih funkcija neophodnih za normalno funkcionisanje živih organizama. Predstavljaju osnovnu komponentu bioloških membrana i utiču na njihovu propustljivost, učestvuju u prenošenju nervnih impulsa, u izgradnji energetskih rezervi i slično [60].

2.2 Centralna dogma molekularne biologije

Godine 1958., Fransis Krik (eng. *Francis Crick*) je pokušao da objedini suptilne poveznosti molekula DNK, RNK i proteina pomoću formalizma koji je on nazvao **centralnom dogmom molekularne biologije (CDMB)**. Reč dogma koja se često koristi u kontekstu religijske doktrine u koju pravi vernik ne sme da sumnja, iskorišćena je u nazivu usled nesporazuma. Kada je Krik formulisao CDMB, bio je pod utiskom da dogma zapravo znači „*ideja za koju ne postoji bilo kakav zdravorazumski dokaz*".

CDMB objedinjava nekoliko procesa koji određuju tok informacija unutar organizama: replikaciju, transkripciju, translaciju, kao i posttranslacione modifikacije. **Replikacija** predstavlja proces kreiranja dve identične kopije DNK molekula, na osnovu izvornog DNK molekula [1]. **Transkripcija** označava proces kreiranja RNK lanca pomoću enzima **RNK polimeraze**, prepisivanjem jednog dela lanca DNK. RNK polimeraza spaja redom nukleotide jednog dela DNK lanca sa njihovim odgovarajućim parovima (kreirajući RNK lanac), pri čemu se umesto nukleotida T koristi nukleotid U prilikom uparivanja. **Translacijom** se ovako kreiran RNK lanac čita redom tri po tri nukleotida, stvarajući aminokiseline koje odgovaraju tim tripletima.



SLIKA 2.1: Centralna dogma molekularne biologije (grafički prikaz) [17].

Te aminokiseline se istovremeno i spajaju, formirajući polipeptidni lanac. Ovo je ilustrovano slikom 2.1.

Opširnije rečeno, pojedini delovi DNK molekula (koji se nazivaju genima) bivaju transkribovani u oblik RNK koji nazivamo glasničkom RNK (eng. *messenger RNA*) koja sadrži kodirane informacije neophodne za sintezu proteina. Te informacije zapravo predstavljaju niz nukleotida A, C, G i U koji se dalje prenosi na obradu do ribozoma.[1] Glasničku RNK ribozom dekodira čitajući trojke nukleotida (tzv. kodone) i prevodeći ih u neku od 20 aminokiselina koje se još nazivaju i standardnim. Pored prethodnih, među kodonima ima i onih koji inicijalizuju proces translacije (START kodoni) tj. terminišu taj proces (STOP kodoni). Pomenuto preslikavanje naziva se **genetskim kodom** i opisano je tabelom 2.1.

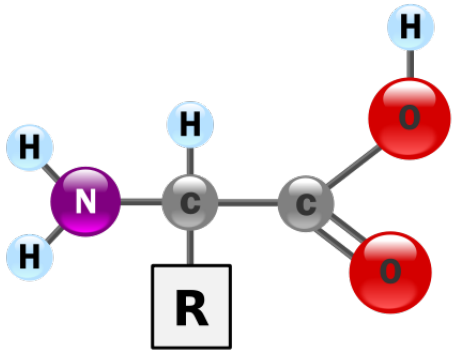
2.3 Aminokiseline

Aminokiseline (skraćeno AK) koje pomoću procesa translacije ulaze u sastav proteina nazivamo **proteinogenim** AK. Proteinogene AK čine 20 standardnih AK, datih u tabeli 2.1 i još dodatne dve AK koje specijalnim translacionim mehanizmima ulaze u sastav proteina [2]. Sa druge strane, neproteinogene AK su one koje ili ne ulaze u sastav proteina ili se ne proizvode direktno u izolaciji standardnim ćelijskim procesima (već najčešće nekim posttranslacionim modifikacijama proteina).

[1] Ribozom predstavlja makromolekularnu mašinu koja se nalazi u svim živim ćelijama, čija je glavna uloga sinteza proteina uz pomoć glasničke RNK [1].

Aminokiselina	Troslojna oznaka	Kodon
Alanin	Ala	GCU, GCC, GCA, GCG
Arginin	Arg	CGU, CGC, CGA, CGG, AGA, AGG
Asparagin	Asn	AAU, AAC
Asparaginska kiselina	Asp	GAU, GAC
Cistein	Cys	UGU, UGC
Glutamin	Gln	CAA, CAG
Glutaminska kiselina	Glu	GAA, GAG
Glicin	Gly	GGU, GGC, GGA, GGG
Histidin	His	CAU, CAC
Izoleucin	Ile	AUU, AUC, AUA
Leucin	Leu	UUA, UUG, CUU, CUC, CUA, CUG
Lizin	Lys	AAA, AAG
Metionin	Met	AUG
Fenilalanin	Phe	UUU, UUC
Prolin	Pro	CCU, CCC, CCA, CCG
Serin	Ser	UCU, UCC, UCA, UCG, AGU, AGC
Treonin	Thr	ACU, ACC, ACA, ACG
Triptofan	Trp	UGG
Tirozin	Tyr	UAU, UAC
Valin	Val	GUU, GUC, GUA, GUG
-	Start	AUG, GUG
-	Stop	UAG, UGA, UAA

TABELA 2.1: Genetski kôd.



SLIKA 2.2: Struktura aminokiseline u svojoj nejonizovanoj formi [15].

2.3.1 Klasifikacija aminokiselina

Posmatrano iz hemijskog ugla, AK su jedinjenja koja sadrže amino grupu $(-NH_2)$, karboksilnu grupu $(-COOH)$ i bočni lanac (R-grupu, radikal ili R-ostatak). Šematski prikaz AK dat je slikom 2.2. Dok su amino i karboksilna grupa zajedničke za sve AK, R-grupa je specifična za svaku. Jedna od podela AK je upravo prema prirodi R-grupe, gde se one izdvajaju u 7 celina [1]:

1. Aminokiseline sa nepolarnim (hidrofobnim) bočnim nizom: alanin, valin, leucin, izoleucin, glicin i prolin;
2. Aminokiseline sa aromatičnim bočnim nizom: fenilalanin tirozin triptofan;
3. Aminokiseline sa baznim bočnim nizom: lizin, arginin i histidin;
4. Aminokiseline sa kiselinskim ostatkom u bočnom nizu: asparaginska kiselina i glutaminska kiselina;
5. Aminokiseline sa amidnim ostatkom u bočnom nizu: asparagin i glutamin;
6. Aminokiseline sa hidroksilnom grupom u bočnom nizu: serin i treonin;
7. Aminokiseline sa sumporom u bočnom nizu: metionin i cistein.

2.4 Proteini

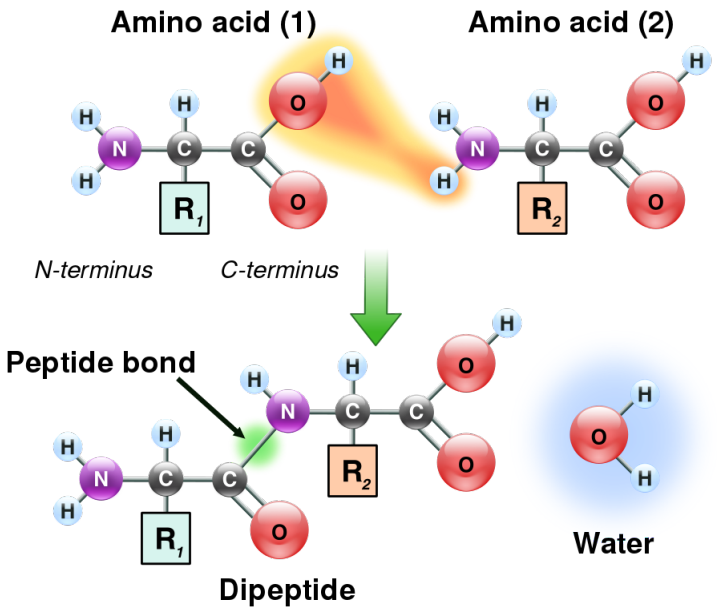
Proteini su prirodni molekuli izgrađeni od aminokiselina koje su međusobno povezane peptidnim vezama[2] između ugljenikovog atoma jedne AK i amino grupe druge AK [66]. Uobičajena podela je da se molekuli koji sadrže od dve do pedeset AK nazivaju peptidima ili polipeptidima, dok se proteinima nazivaju molekuli koji su duži od pedeset AK. Primer dipeptida, tj. polipeptida sastavljenog od dve aminokiseline, izložen je na slici 2.3. Naziv proteina potiče od grčke reči *proteios* koja prevedena znači prvi tj. glavni, što oslikava njihovu važnost, a i važnost procesa u kojima oni učestvuju.

2.4.1 Funkcija proteina

Jedinstvena sekvenca AK koja čini jedan protein obezbeđuje savijanje tog proteina u jedinstven trodimenzioni oblik koji se drugačije naziva konformacijom.[3] Sa druge strane, proteini ne predstavljaju rigidne objekte. Većina njih može imati pokretne delove čija mehanička dejstva direktno utiču na određene hemijske reakcije. Upravo ova povezanost između fizičkih pokreta i hemijskih procesa obezbeđuje neverovatne mogućnosti proteina koje se manifestuju u mnogim dinamičkim procesima unutar živih ćelija [1]. Neke od najznačajnijih funkcija proteina su date tabelom 2.2.

[2]Peptidna veza (amidna veza) je kovalentna hemijska veza koja se formira između dva molekula kada karboksilna grupa jednog molekula reaguje sa amino grupom drugog molekula, uz otpuštanje molekula vode [1].

[3]Konformacija predstavlja prostornu raspodelu atoma u tri dimenzije unutar makromolekula kao što je protein ili neukleinska kiselina.



SLIKA 2.3: Kreiranje dipeptida spajanjem dve aminokiseline peptidnom vezom [20].

Funkcija	Opis funkcije	Primer proteina
Antitela	Antitela štite organizam vezujući se za specifične strane čestice kao što su virusi i bakterije.	Imunoglobin G (IgG)
Enzimi	Enzimi su neophodni u sprovođenju velikog broja ćelijskih hemijskih reakcija. Takođe pomažu pri kreiranju novih molekula.	Fenilalanin-hidroksilaza
Glasnici	Proteini glasnici, kao što su neke vrste hormona, prenose signale zarad koordinisanog sprovođenja međućelijskih procesa, kao i procesa između tkiva i organa.	Hormon rasta
Struktura	Ovi proteini obezbeđuju strukturu ćelija ali i omogućavaju telu da se kreće u prostoru.	Aktin
Transport Skladištenje	Proteini takođe mogu da se vezuju i prenose atome i manje molekule unutar ćelija ali i unutar čitavog organizma.	Feritin

TABELA 2.2: Primeri funkcija proteina [68].

2.4.2 Struktura proteina

Da bi bili u mogućnosti da obavljaju svoje biološke funkcije, proteini se savijaju u jednu ili više različitih prostornih konformacija usled nekovalentnih interakcija. Zarad razumevanja funkcija proteina na molekularnom nivou, potrebno je odrediti njihovu trodimenzionu strukturu. Pojam strukture u kontekstu proteina ima dosta složenije značenje nego u kontekstu nekih manjih molekula. Proteini, kao što je ranije rečeno, predstavljaju makromolekule kod kojih razlikujemo četiri različitih nivoa struktura:

1. Primarna struktura;
2. Sekundarna struktura;
3. Tercijarna struktura;
4. Kvaternarna struktura.

Šematski prikaz različitih nivoa struktura proteina izložen je slikom 2.4.

Primarna struktura proteina

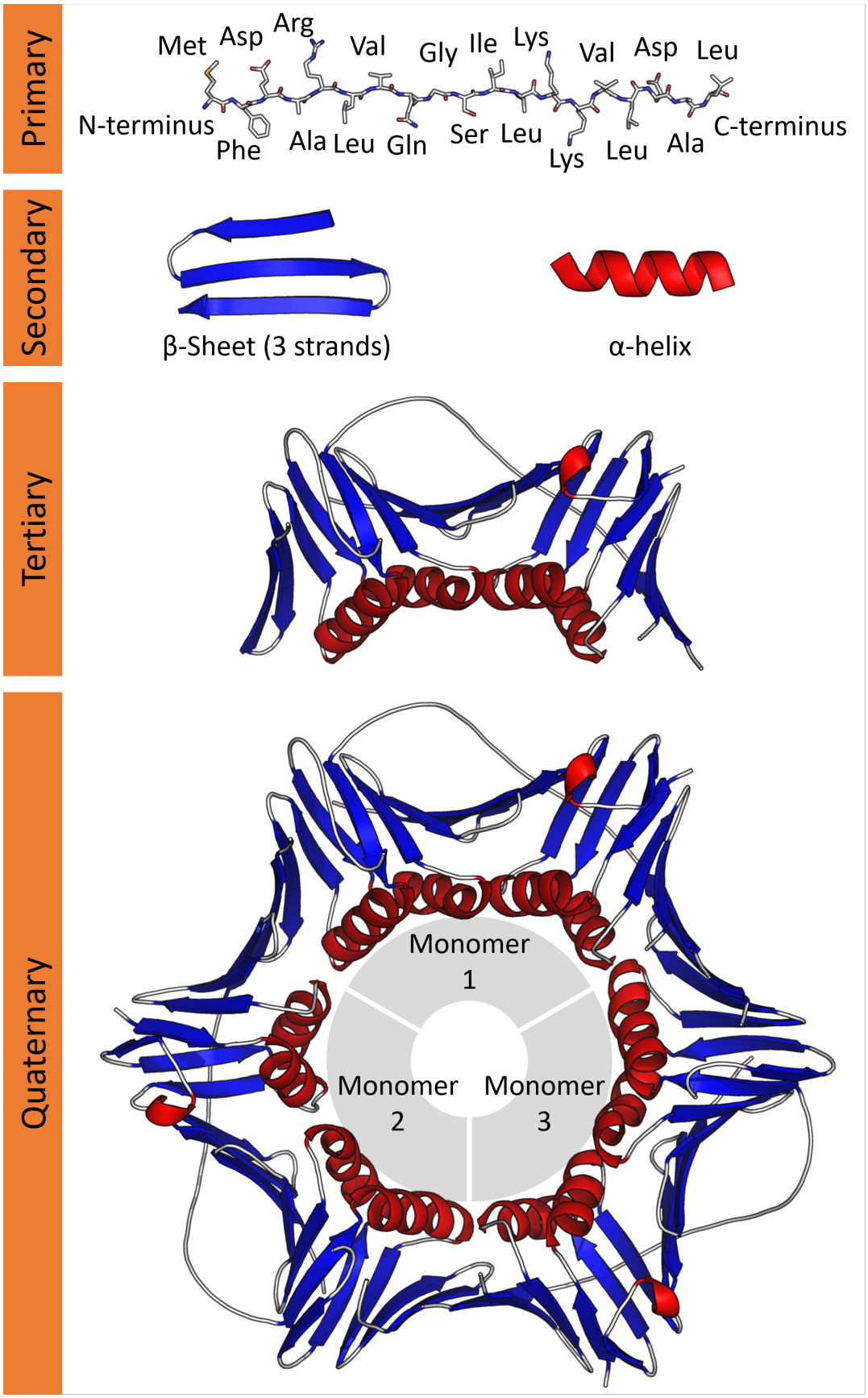
Kao što je naglašeno u poglavlju 2.4, svaki protein je izgrađen od određenog broja AK. Ako posmatramo 20 standardnih AK, lako je primetiti da postoji 400 različitih potencijalnih dipeptida koji mogu nastati od tih dvadeset AK. Analogno, postoji 8000 potencijalnih tripeptida. Broj mogućih kombinacija postaje ogroman kada u formiranju polipeptidnog lanca učestvuje veći broj AK. Poznavanje primarne strukture proteina upravo predstavlja poznavanje tačne sekvence AK koje ga sačinjavaju [3].

Sekundarna struktura proteina

Sekundarna struktura se zasniva na vodoničnim vezama između amino i karboksilne grupe u sekvenci AK. Često, ove slabe interakcije između sukcesivnih R-ostataka AK dovode do uvijene (helične) strukture. To znači da se polipeptidni lanac uvija tako da obrazuje tzv. **α-zavojnicu** (**α-heliks**), što je direktno uzrokovano interakcijama između grupa na rastojanjima 3-4 R-ostataka. Specijalno, neke AK se „krive" u ovim regionima zavojnice što dovodi do savijanja (eng. *folding*) čitavog lanca i kreiranja globularnije molekularne strukture. Delovi α-zavojnice se često mogu videti u većim polipeptidima, čak iako je ostatak lanca u potpunosti neuređen. Neki kraći proteini su često celi organizovani u obliku α-zavojnice. Primer takvog proteina je glukagon.

Drugi čest oblik, pored α-zavojnice, predstavlja **β-ploča**. Ovaj oblik polipeptidnog lanca gotovo je potpuno izdužen, za razliku od α-zavojnice koja je usko savijena. Takođe, R-ostaci susednih AK su suprotno usmereni. **β-naborana struktura** se formira povezivanjem dve ili više β-ploča vodoničnim vezama. Susedni lanci ove strukture se mogu prostirati u istom smeru (paralelne β-strukture) ili u suprotnom (antiparalelne β-strukture[4]) [40].

[4] Antiparalelna β-struktura je stabilnija od paralelne usled većeg broja bolje poravnatih vodoničnih veza.



SLIKA 2.4: Strukture proteina [21].

Tercijarna struktura proteina

Tercijarna struktura opisuje globalnu konformaciju proteina, odnosno trodimenzioni raspored atoma u jednom proteinu. Tericijarna struktura je zavisna i od primarne i od sekundarne strukture, jer je definisana interakcijama između AK koje mogu biti veoma udaljene u polipeptidnom nizu, ali se usled savijanja lanca one nađu jedna blizu druge. Pod tercijarnom strukturom podrazumevamo prostorne koordinate svih atoma polipeptida. Često, slični polipeptidni lanci imaju i slične trodimenzione strukture. Iako trodimenzioni oblik proteina često izgleda nasumičan, on direktno predstavlja posledicu kako vodoničnih veza, tako i drugih vidova interakcije R-ostataka AK (npr. elektrostatičke i hidrofobne interakcije) [7].

Kvaternarna struktura proteina

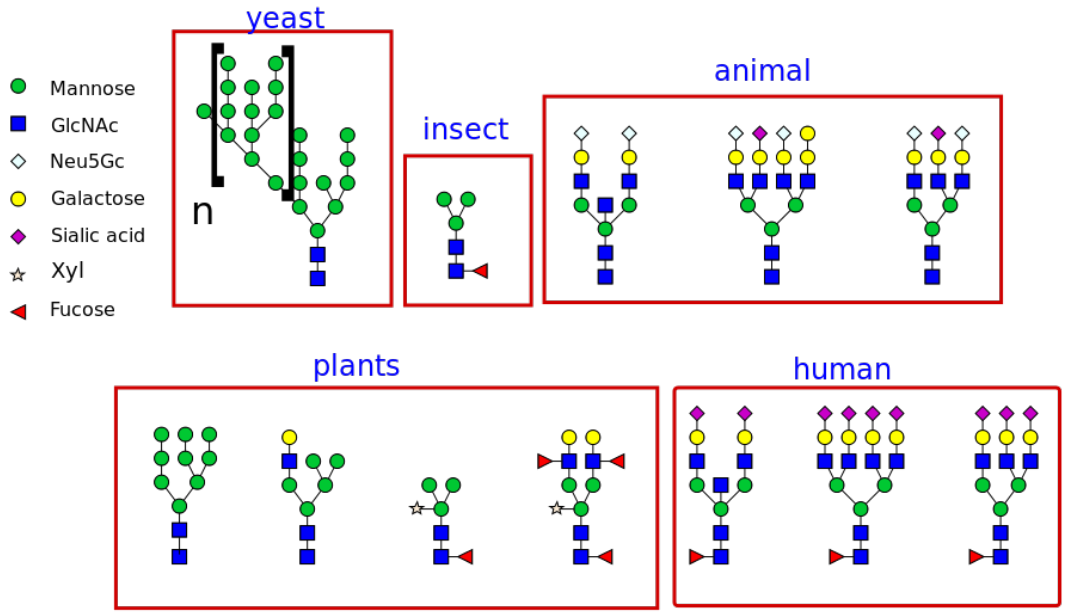
Mnogi proteini su sačinjeni od višestrukih polipeptidnih lanaca koji se često nazivaju *proteinskim podjedinicama*. Kvaternarna struktura opisuje kako ove jedinice međusobno interaguju i kako se raspoređuju u prostoru, kreirajući proteinski kompleks. Finalni oblik proteinskog kompleksa je opet određen različitim interakcijama, kao što su vodonične i disulfidne veze. Sa druge strane, postoje proteini čija kompleksnost nije dovoljno velika da bi sadržali i kvartenarnu strukturu, što znači da ona nije definisana za sve proteine [3].

2.4.3 Glikozilacija kao PTP

Novosintetisani polipeptidi često zahtevaju dodatne procese da bi postali funkcionalni. Ti posttranslacioni procesi (skraćeno PTP) modifikuju polipeptidni lanac i dovode do proširenja opsega funkcija proteina. Modifikacije obično predstavljaju vezivanje drugih biohemijskih funkcionalnih grupa (npr. acetat, fosfat, lipidi, ugljeni hidrati) ili cepanje lanca u nekom od njegovih regiona. Jedan od PTP je **glikozilacija** koja predstavlja vezivanje ugljenih hidrata (glikana), pomoću glikozidnih veza, na lipide, proteine ili neke druge organske molekule. Nadalje ćemo se baviti samo slučajem vezivanja glikana na proteine, pri čemu se kao rezultat te interakcije dobijaju **glikoproteini**. Glikozilacija proteina je specifična za proteine u eukariotskim ćelijama, dok se kod prokariotskih ne javlja. Najčešću vrstu proteina ćelija jednog eukariotskog organizma čine upravo glikozilovani proteini tj. glikoproteini [66]. Usled svoje fundamentalne prirode unutar ćelije, glikozilacija proteina ima uticaja i na mnoge bolesti kod životinja, ali i kod ljudi (alkoholizam [53], Alchajmerova bolest [61], rak [57] itd.).

Skoro svi sekratorni i proteini sa strukturnom funkcijom eukariotskih ćelija su glikozilovani [65]. Procesom glikozilacije se mogu dobiti 5 različitih tipova glikana [65]:

- N-vezani glikani dodati na azot bočnih lanaca asparagina ili arginina;
- O-vezani glikani dodati na hidroksilni kiseonik bočnih lanaca serina, treonina, tirozina, hidroksi-lizina, ili hidroksi-prolina, ili na kiseonike na lipidima kao što je keramid;
- Fosfo-glikani vezani putem fosfata na fosfo-serinu;
- C-vezani glikani, retka forma glikozilacije gde je šećer dodat na ugljenik bočnog lanca triptofana;
- Dodavanje GPI ankera koji povezuje proteine sa lipidima glikanskom vezom.



SLIKA 2.5: Različite vrste glikana sintetisane od strane različitih organizama [25].

N-vezana glikozilacija

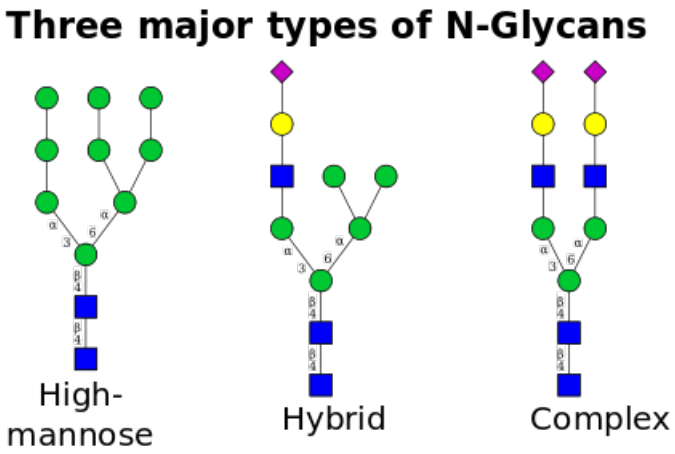
N-vezana glikozilacija (skraćeno N-glikozilacija) je ključna za strukturu, ali i za funkciju mnogih proteina eukariotskih ćelija [39, 54]. Priroda N-vezanih glikana određena je kako proteinom, tako i ćelijom u kome se nalaze [63]. Proces se razlikuje i među vrstama s obzirom da različite vrste sintetišu različite tipove N-vezanih glikana. Neki od njih su prezentovani slikama 2.5 i 2.6.

Postoje dva tipa veza koje učestvuju u formiranju glikoproteina: veze između saharidnih ostataka u glikanu i veze između glikanskog lanca i molekula proteina. Šećerni monomeri su međusobno povezani u glikanskom lancu glikozidnim vezama. Te veze se tipično formiraju između ugljenika 1 i 4 šećernih molekula. Sa druge strane, vezivanje glikanskog ostatka na protein zahteva prepoznavanje konsenzus niske (tj. motiva). N-vezani glikani su skoro uvek vezani za atom azota R-ostatka asparagina (skraćeno Asn) koji je prisutan kao deo **Asn–X–Ser/Thr** konsenzus niske, gde je X bilo koja aminokiselina izuzev prolina (skraćeno Pro) [63].

Pojava novih potencijalnih mesta izvršavanja procesa N-vezane glikozilacije na polipeptidnom lancu (skraćeno mesta N-glikozilacije) može promeniti funkciju proteina na pozitivan ili negativan način. Sa druge strane, nestanak mesta N-glikozilacije, usled određenih mutacija, skoro uvek dovodi do poremećenog savijanja proteina ili poremećenih transportnih ili drugih funkcija [70, 72].

2.5 Klasifikacija šablona

Eksponencijalno uvećavanje računarske moći početkom drugog milenijuma, kao i pojava ogromnog broja podataka, dovodi do naglog rasta uspešnosti metoda mašinskog učenja (skraćeno MU). Za probleme određenih domena, za koje se smatralo da će ljudska uspešnost u njihovom rešavanju biti nenadmašna, ubrzo se došlo do rezultata koji su superiorni u odnosu na rezultate eksperata. Problemi koji se mogu rešiti pomoću MU su raznorodni i mogu se razvrstati na osnovu metoda MU koje se



SLIKA 2.6: Različite vrste N-glikana [24].

koriste za njihovo rešavanje. Te metode se zatim mogu podeliti na više načina, pri čemu bi njihova osnovna podela bila prema prirodi problema učenja. Tako razlikujemo 3 vrste problema:

- Problemi nadgledanog učenja (eng. *supervised learning*);
- Problemi nenadgledanog učenja (eng. *unsupervised learning*);
- Problemi učenja potkrepljivanjem (eng. *reinforcement learning*).

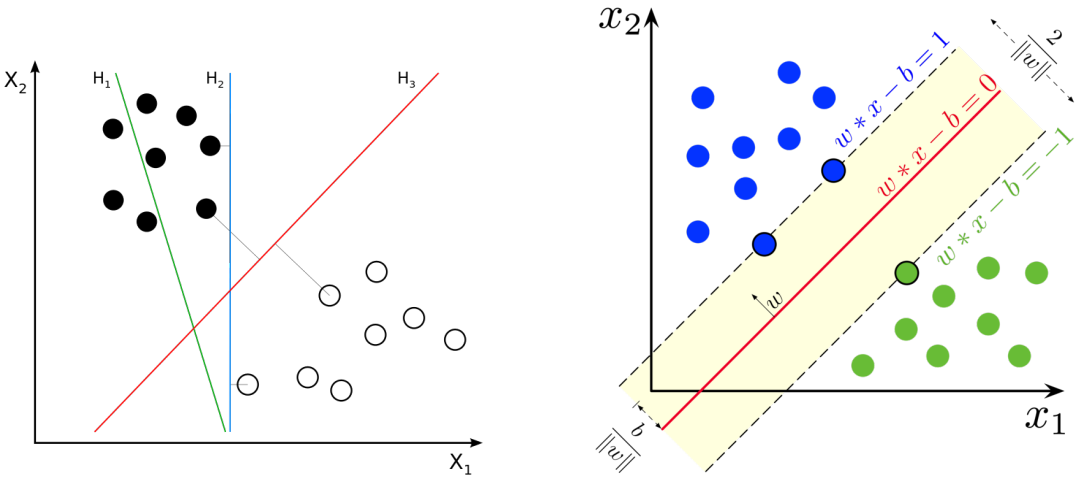
Nadgledano učenje predstavlja jedan od najkorišćenijih vidova mašinskog učenja. Kao što i samo ime kaže, osnovna karakteristika mu je da se podaci sastoje iz parova opisa onoga na osnovu čega se uči i onoga što je potrebno naučiti, tj. iz osobina podataka koje se uče i željenih rezultata. Dve osnovne vrste problema nadgledanog učenja su regresioni i klasifikacioni problemi. Regresioni problemi se odnose na predviđanje kontinualne tj. neprekidne ciljne promenljive, dok se klasifikacioni odnose na predviđanje kategoričke cilje promenljive [51].

Ljudi se konstantno suočavaju sa klasifikacionim problemima u svakodnevnom životu. Čitanje tekstova, prepoznavanje lica, traženje ključeva po džepovima, pa čak i razlikovanje svežeg od pokvarenog voća predstavljaju rešavanje klasifikacionih problema. Odatle dolazi stalna potreba za usavršavanjem metoda mašinskog učenja, kao i želja za nadmašivanjem ljudi u rešavanju raznih problema.

2.5.1 Metod potpornih vektora

Jedna od najkorišćenijih metoda (tj. modela) mašinskog učenja za rešavanje kako regresionih, tako i klasifikacionih problema, predstavlja metod potpornih vektora (eng. *Support Vector Machines*), ili skraćeno MPV. Ovaj model MU kao cilj ima nalaženje optimalne hiperravni,[5] tj. one hiperravni koja što bolje razdvaja predstavnike različitih klasa nekog skupa podataka. Ovo znači da MPV nalazeći optimalnu hiperravan zapravo maksimizuje udaljenost svih mogućih hiperravni do najbližih predstavnika svake od klasa. Idealno, na ovaj način se oko optimalne hiperravni kreira pojas u kome se ne nalazi instanca nijedne od klasa i koji nazivamo najširim pojasom ili pojasom optimalne hiperravni.

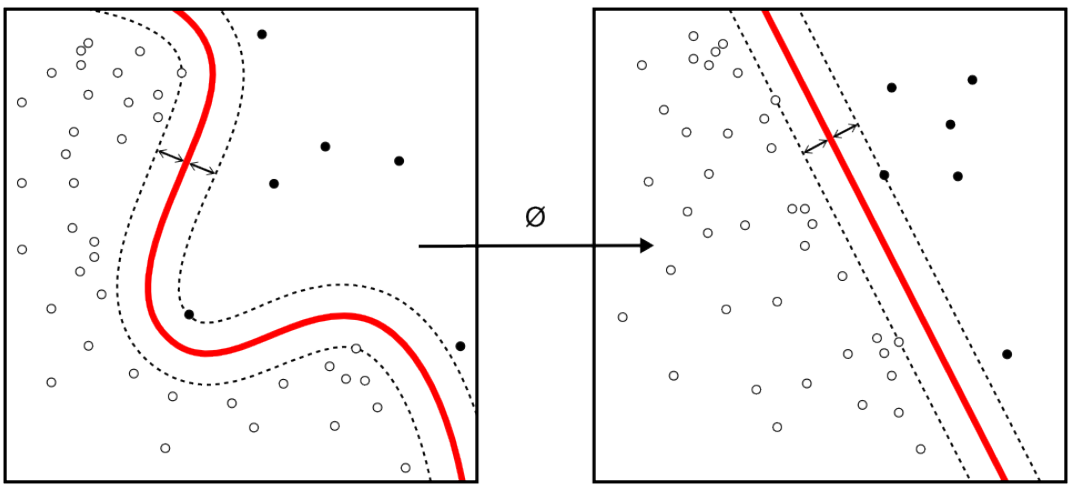
[5]Hiperravan predstavlja bilo koji potprostor nekog prostora dimenzije n, čija je dimenzija $n-1$.



(A) Različite hiperravni [23]. (B) Najširi pojas [22].

SLIKA 2.7: Različite hiperravni i optimalna hiperravan sa najširim pojasom.

MPV se može koristiti za rešavanje klasifikacionih, ali i regresionih problema. Kako se u ovom radu rešava klasifikacioni problem, u nastavku ćemo razmatrati samo takvu verziju MPV. Problemi klasifikacije mogu imati dva oblika: u prvom su podaci linearno razdvojivi (primer je dat na slici 2.7b), dok u drugom nisu (primer je dat na slici 2.8). U prvom slučaju, MPV nalazi optimalnu hiperravan, kao što je i ranije rečeno, dok u drugoj to nije moguće. Pri radu sa realnim podacima, češće se javlja drugi slučaj, gde hiperravan ne predstavlja adekvatnu granicu između klasa, već je ta granica nekog proizvoljnog oblika. Da bi se klasifikovali takvi podaci, pribegava se korišćenju kernela koji zapravo predstavljaju preslikavanja polaznog vektorskog prostora, u kome se podaci nalaze, u neki novi vektorski prostor. Kerneli zatim omogućavaju preraspodelu polaznih podataka u tom novom prostoru tako da su oni linearno razdvojivi, te se onda klasifikacija lako rešava prethodno opisanim postupkom [37]. Primer kernela je prezentovan slikom 2.8.



SLIKA 2.8: Primer kernela [18].

Hiperparametri metoda potpornih vektora

Metode MU često omogućavaju promenu vrednosti određenih parametara modela koje nazivamo hiperparametrima, zarad kontrolisanja prilagodljivosti modela skupu podataka. Jako prilagođavanje modela skupu podataka dovodi do toga da model veoma mnogo greši na novim, neviđenim podacima, dok sa druge strane slabo prilagođavanje modela skupu podataka može dovesti do prevelike generalizacije na novim podacima, te samim tim i neefikasnom klasifikacionom ili regresionom modelu.

Jedan od hiperparametara MPV predstavlja izbor nekog od kernela, ili pak korišćenje linearnog MPV (onog u kome se traži linearna hiperravan). Zatim, ukoliko se izvrši izbor kernela, često je moguće i taj kernel optimizovati i samim tim menjati prilagodljivost MPV podacima. Na primer, izborom često korišćenog Gausovog kernela omogućava se kontrolisanje njegovog oblika variranjem metaparametra γ koji menja širinu Gausovog zvona definisanog ovim kernelom [51]. Pored prethodnog, MPV dozvoljava i da određen broj instanci podataka bude unutar najšireg pojasa, te samim tim obezbeđuje i mogućnost postojanja „loših" podataka. Pod „lošim" podacima podrazumevaju se oni podaci koji su nastali usled nekakve greške prilikom merenja, unosa i slično, te samim tim drastično odstupaju od ostalog dela skupa podataka. Pri tome, često je moguće kontrolisati koliko instanci može upasti unutar najšireg pojasa, čime se kontroliše odlučivost klasifikatora, te je i to jedan hiperparametar koji je moguće varirati.

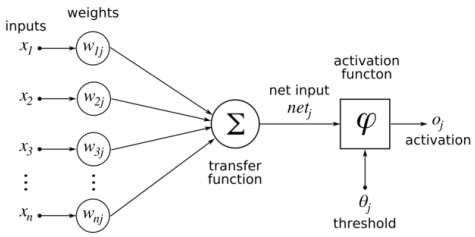
2.5.2 Neuronske mreže

Neuronske mreže predstavljaju metodu mašinskog učenja koja se posebno proslavila poslednjih godina, usled svojih uspeha u rešavanju, do nedavno, veoma teških problema. Prethodno je u velikoj meri posledica nagle porasti moći računara, kao i pada cena određenih računarskih komponenti koje uveliko smanjuju vreme obučavanja neuronskih mreža. Postoji više vrsta neuronskih mreža, pri čemu je svaka od njih specijalizovana za rešavanje problema različitih tipova. U ovom radu se baziramo na potpuno povezanim neuronskim mrežama (u nastavku samo - neuronska mreža) za rešavanje klasifikacionih problema, te će se na dalje govoriti samo o njima.

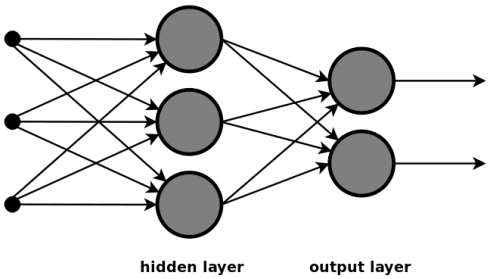
Neuronska mreža se može posmatrati kao niz slojeva koji se sastoje od neurona. Neuron predstavlja jednostavnu parametrizovanu funkciju, te se obučavanje neuronske mreže svodi upravo na određivanje parametara neurona, tako da je greška na novim podacima što manja. Izlaz svakog neurona predstavlja primenu neke nelinearne transformacije (tzv. aktivacione funkcije) na linearnu kombinaciju svojih ulaza. Grafički prikaz neurona dat je slikom 2.9a. Slojevi neurona su međusobno povezani tako da neuroni jednog sloja primaju kao ulaze izlaze prethodnog sloja, a svoje izlaze prosleđuju svim neuronima narednog sloja. Prvi sloj mreže se naziva ulaznim slojem, poslednji izlaznim, dok se slojevi između nazivaju skrivenim slojevima. Mreže koje sadrže više od jednog skrivenog sloja često se nazivaju dubokim neuronskim mrežama [51]. Izgled neuronske mreže sa jednim skrivenim slojem prezentovan je slikom 2.9b.

Hiperparametri neuronskih mreža

Neuronske mreže predstavljaju vrlo prilagodljive modele, te je njihovo treniranje izuzetno zahtevan posao koji iziskuje dosta iskustva. Prevelika prilagodljivost podacima se izbegava nalaženjem onih vrednosti hiperparametara za koje mreža ima



(A) Izgled neurona [16].



(B) Izgled neuronske mreže sa jednim skrivenim slojem [19].

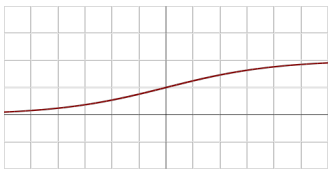
SLIKA 2.9: Struktura neuronske mreže.

najbolje ponašanje na test skupu, tj. na mreži neviđenim podacima. Hiperparametri neuronskih mreža su brojni, te će se u nastavku opisati samo oni koji su korišćeni u radu.

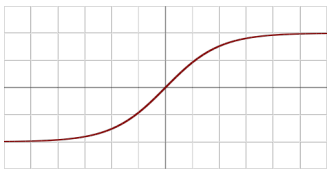
Nakon izbora neuronskih mreža kao modela kojim će se rešavati neki problem, dolazi se do prvog važnog pitanja - kakvu arhitekturu izabrati tj. kreirati? Upravo izbor arhitekture predstavlja jedan od važnijih hiperparametara i on se svodi na izbor broja skrivenih slojeva, kao i izbor broja neurona u svakom sloju te mreže. Dobar izbor arhitekture može značajno poboljšati željene rezultate.

Sa izabranom arhitekturom neuronske mreže, postavlja se pitanje koje polazne vrednosti parametara koristiti za svaki od neurona. Inicijalizacija parametara takođe predstavlja hiperparametar koji može uzimati različite vrednosti. U praksi se pokazuje da se najbolji rezultati postižu uzimanjem vrednosti parametara iz uniformne ili normalne raspodele, pri čemu postoje razne modifikacije ove tehnike u zavisnosti od vrste problema koji se rešava.

Pored prethodnog, neuronske mreže omogućavaju i izbor aktivacionih funkcija neurona po slojevima. Najčešće korišćene aktivacione funkcije su: sigmoidna (u oznaci σ), tangens hiperbolički (u oznaci *tanh*) i ispravljena linearna jedinica (u oznaci *relu*). Grafici ovih aktivacionih funkcija su izloženi na slici 2.10.



(A) Sigmoidna funkcija [12].



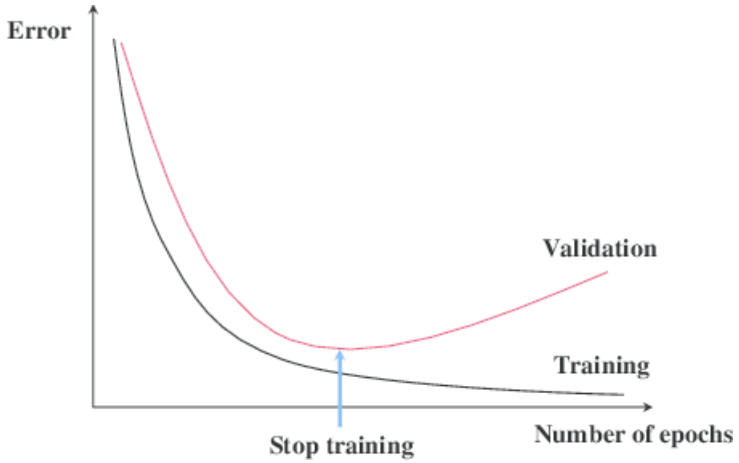
(B) Tangens hiperbolički [14].

(C) Ispravljena lin. jedinica [13].

SLIKA 2.10: Aktivacione funkcije.

Obučavanje neuronskih mreža, kao i ostalih modela mašinskog učenja, obavlja se pomoću različitih tehnika optimizacije koje zapravo minimizuju grešku modela na skupu podataka nad kojima se trenira. Izbor optimizatora takođe predstavlja hiperparametar, pri čemu većina njih sadrži dodatne metaparametre koji menjaju njihovo ponašanje, tj. najčešće brzinu kojom model uči. Neke od najpoznatijih tehnika optimizacije su gradijentni spust, metod inercije, Adam, itd.

Da bi se izbegla prevelika prilagodljivost modela podacima, često se pribegava tehnikama regularizacije. Ove tehnike imaju za cilj da smanje fleksibilnost modela



SLIKA 2.11: Tehnika ranog zaustavljanja [62].

i sačuvaju određen nivo moći generalizacije. Uobičajeno je da se tehnike regularizacije implementiraju u sklopu modela mašinskog učenja, i da u sebi sadrže mehanizam kontrolisanja regularizovanosti modela (najčešće u obliku hiperparametra modela). Pored opštih vrsta regularizacije (kao što su npr. l_1 regularizacija, l_2 regularizacija, ...) koje se intezivno koriste u različitim modelima mašinskog učenja, neuronske mreže sadrže i specifične vidove regularizacije prilagođene velikoj fleksibilnosti ovih modela. Jedna od njih je rano zaustavljanje (eng. *early stopping*) koja koristi specifičnost treniranja neuronskih mreža. Prilikom obučavanja neuronskih mreža, primetan je trend konstantnog opadanja greške na skupu za obučavanje, dok sa druge strane, na nekom odvojenom skupu (koji nazivamo validacionim), ta greška će prvo opadati (usled učenja), a zatim rasti (usled preprilagođavanja). Rano zaustavljanje prati grešku na validacionom skupu i prekida obučavanje onda kada greška krene da raste pri svakom novom obučavanju (slika 2.11).

2.5.3 Rad sa nebalansiranim podacima

Rešavanje klasifikacionih problema kod podataka gde jedna od klasa ima značajno više podataka od drugih klasa smatramo radom sa nebalansiranim podacima. U daljem tekstu podrazumevaćemo da radimo sa binarnim problemom klasifikacije, tj. onim problemom gde su podaci podeljeni u dve klase. Obično jednu klasu nazivamo pozitivnom (i označavamo brojem 1), a drugu negativnom (sa oznakom 0). Tada, za klasu koja ima manje podataka kažemo da je **manjinska klasa**, dok onu sa više podataka nazivamo **većinskom klasom**. Nebalansiranost podataka može dovesti do velikih problema kod većine klasifikacionih modela MU. Mali broj podataka koji pripadaju manjinskoj klasi dovodi do toga da klasifikator stvori sklonost ka većinskoj klasi, tako da i one instance koje pripadaju manjinskoj klasi klasifikuje kao podatke većinske. Ovo ponašanje može biti posebno nezgodno u određenim domenima. Na primer, ako se radi sa medicinskim podacima vezanim za određenu bolest, verovatnije je da će manjinska klasa predstavljati pacijente koji su oboleli od te bolesti, a čak se u praksi pokazuje da će odnos klasa u određenim slučajevima biti drastičan. Pri klasifikaciji, ovo može dovesti do velikog broja loše klasifikovanih pacijenata koji su zapravo oboleli, što može imati značajne posledice.

Da bi se prevazišli ovi problemi, razvijene su brojne tehnike kako na nivou samih podataka, tako i na algoritamskom nivou (u sklopu modela MU). Na nivou

podataka, zarad prevazilaženja problema nastalih nebalansiranošću klasa, mogu se koristiti tehnike **nadsempliranja** (eng. *oversampling*) ili **podsempliranja** (eng. *undersampling*). Nadsempliranje označava dodavanje podataka manjinskoj klasi tako da ona dostigne brojnost većinske klase, dok podsempliranje označava eliminisanje određenih podataka većinske klase (skoro uvek nasumično izabranih) dok se ne dostigne brojnost manjinske klase [30].

SMOTE i ADASYN

Dok je kod podsempliranja plan eliminisanja podataka većinske klase jasan (eliminišu se elementi nasumično, osim ako to nije drugačije naglašeno), tehnike nadsempliranja su komplikovanije i mogu u određenim slučajevima dovesti do poboljšanja rezultata. Naivni pristup nadsempliranju bi bilo korišćenje nasumičnog nadsempliranja, pri čemu je ovaj pristup u većini slučajeva sasvim pogrešan. Na ovaj način se uvode podaci koji su van prostora atributa (eng. *feature space*) izvornih podataka,[6] čime se gubi svaka implicitna veza između njih. Samim tim se „raspršuje" prostor atributa, što čini bilo kakav proces učenja uzaludnim.

Drugi prilaz rešavanju ovog problema bi bio pažljivo dodavanje sintetičkih podataka, tako da se u što manjoj meri naruši prostor atributa izvornih podataka. Jedna takva tehnika je **SMOTE** (skraćeno od eng. *Synthetic Minority Over-sampling Technique*) nadsempliranje, pomoću koje se generišu novi, sintetički podaci nasumičnom interpolacijom parova najbližih suseda. SMOTE tehnika se u praksi pokazala veoma dobro, a često se koristi i u kombinaciji sa tehnikom nasumičnog podsempliranja većinske klase zarad dobijanja još boljih rezultata [6].

ADASYN (skraćeno od eng. *Adaptive Synthetic*) predstavlja prirodno proširenje SMOTE tehnike. Kreiranje sintetičkih podataka se obavlja slično kao i kod SMOTE tehnike, pri čemu se kao mera udaljenosti najbližih suseda eksplicitno koristi njihovo euklidsko rastojanje. Glavna razlika SMOTE i ADASYN tehnika nadsempliranja jeste ta da SMOTE generiše isti broj novih, sintetičkih podataka za svaki izvorni podatak manjinske klase. Sa druge strane, ADASYN generiše različit broj sintetičkih podataka za svaki izvorni podatak manjinske klase u zavisnosti od njegovih suseda. Ovo se postiže tako što se za svaki element manjinske klase računa udeo elemenata većinske klase među njegovim susedima. Na ovaj način se kreiraju težine svakom elementu manjinske klase na osnovu kojih se donosi odluka o broju sintetičkih podataka koje je za taj element potrebno kreirati. Elementi manjinske klase kod kojih su među susedima brojniji elementi većinske klase (tj. elementi manjinske klase čije osobine klasifikator teško uči) zahtevaju i veći broj novih, sintetičkih elemenata. Samim tim, glavna osobina ove tehnike nadsempliranja predstavlja smanjenje naklonosti klasifikatora uzrokovane nebalansiranošću klasa, kao i prilagodljivo pomeranje granice klasa unutar prostora atributa (ka podacima koji se teže uče tj. za koje je klasifikator neodlučan) [35].

2.5.4 Evaluacija

Evaluacija modela u MU nam omogućava testiranje sposobnosti predviđanja našeg modela pomoću podataka koji do sada nisu viđeni tokom treniranja. Evaluirati model znači kvantifikovati njegovu moć predviđanja [51]. Na ovaj način želimo da dobijemo okvirnu reprezentaciju moći našeg modela pri njegovom korišćenju nad realnim podacima.

[6] Pod izvornim podacima ćemo u daljem tekstu podrazumevati podatke nad kojima nisu primenjene tehnike sempliranja.

Stvarno\Predviđeno	**Negativni**	**Pozitivni**
Negativni	stvarno negativni (TN)	lažno pozitivni (FP)
Pozitivni	lažno negativni (FN)	stvarno pozitivni (TP)

TABELA 2.3: Matrica konfuzije problema binarne klasifikacije.

Skoro svaka mera kvaliteta počiva na tzv. **matrici konfuzije**[7] koja predstavlja izveštaj predikcija klasifikacionog modela. Matrica konfuzije daje uvid ne samo u greške koje je klasifikator napravio, već i u tačne tipove grešaka koje su napravljene, što omogućava građenje složenih mera uspešnosti klasifikatora. Izgled matrice konfuzije za binarni problem klasifikacije je dat tabelom 2.3.

Različite mere kvaliteta modela se koriste za različite tipove problema. Najčešće mere kvaliteta koje se koriste za klasifikacione probleme jesu:

- Tačnost klasifikacije (eng. *classification accuracy*);
- Preciznost i odziv (eng. *precision and recall*);
- F_1 mera;
- Površina ispod ROC (skraćeno od eng. *receiver operating characteristic*) krive (eng. *area under the curve – AUC*).

Tačnost klasifikacije predstavlja udeo tačno klasifikovanih instanci od ukupnog broja instanci i u slučaju binarne klasifikacije je oblika:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

Preciznost predstavlja udeo izvorno pozitivnih instanci među svim klasifikovanim instancama koje su proglašene pozitivnim:

$$Prec = \frac{TP}{TP + FP}$$

Odziv predstavlja udeo izvorno pozitivnih instanci među svim klasifikovanim instancama koje su izvorno pozitivne:

$$Rec = \frac{TP}{TP + FN}$$

F1 mera uzima istovremeno u obzir i preciznost i odziv (jer samostalno nisu dovoljno informativne mere) i predstavlja njihovu harmonijsku sredinu:

$$F_1 = 2 * \frac{Prec * Rec}{Prec + Rec}$$

Površina ispod ROC krive je mera koja se najčešće koristi kod binarnih problema klasifikacije i njen smisao predstavlja proveru da li model MU dodeljuje manje težine elementima jedne klase, a veće elementima druge klase i data je formulom:

$$AUC = \frac{Rec + Tnr}{2}$$

[7] Matrica konfuzije vuče naziv iz mogućnosti identifikovanja konfuzije između klasa, tj. uvida u to koliko je neka klasa pogrešno klasifikovana kao neka druga klasa.

gde je $Tnr = \frac{TN}{FP+TN}$ [51].

Evaluacija pri radu sa nebalansiranim podacima

Prethodno definisane mere kvaliteta klasifikacionog modela nisu podjednako informativne u svim situacijama. Ako posmatramo probleme u kojima imamo nebalansiranost klasa, kao što su detekcija retkih bolesti, prevara sa kreditnim karticama i slično, tačnost klasifikacije neće biti odgovarajuća mera kvaliteta našeg modela MU. Ako jednoj klasi pripada 99% instanci, a drugoj 1%, tačnost od 0.99 se postiže klasifikovanjem svih instanci u prvu klasu. Na ovaj način se potpuno zanemaruju instance kritičnije, a i važnije druge klase, te je klasifikator samim tim beskorisan.

Preciznost i odziv, posmatrani pojedinačno, takođe ne predstavljaju dobre mere kvaliteta modela u slučaju nebalansiranih klasa. Klasifikujući sve instance kao pozitivne dobijamo maksimalan odziv, a sve kao negativne maksimalnu preciznost. Upravo F_1 mera predstavlja sponu između preciznosti i odziva, te najbolje (od svih prethodno pomenutih mera) oslikava kvalitet modela u slučaju nebalansiranih klasa [51].

U praksi se takođe dosta koristi i površina ispod ROC krive kao mera kvaliteta modela MU pri radu sa nebalansiranim klasama, ali ona u ovom radu nije od velikog značaja te se neće dalje pominjati.

2.6 Postojeći radovi

Sa porastom računarske moći, rastu i mogućnosti obrade visokodimenzionih podataka, što dovodi do efikasnog korišćenja metoda mašinskog učenja za rešavanje mnogih bioloških problema. Ova činjenica omogućava naučnicima da drastično smanje korišćenje standardnih eksperimentalnih procedura koje su skupe i vremenski zahtevne, u korist računarskih metoda koje u najmanju ruku mogu značajno da smanje prostor pretrage a vrlo često mogu dati veoma pouzdana rešenja. Pored toga, pomoću ovih metoda se mogu uočiti skrivene zakonitosti koje možda nisu očigledne u velikim skupovima podataka, pa samim tim i doći do novih saznanja o funkcionisanju raznih bioloških procesa.

Posmatrajući proces N-glikozilacije, postoje dva osnovna problema koja se mogu rešavati: **ispitivanje da li je protein podložan N-glikozilaciji** i **nalaženje pozicije u proteinu na kome je velika verovatnoća dešavanja procesa N-glikozilacije**. Evidentno je da drugi problem direktno zavisi od prvog, tj. razumno je tražiti potencijalno mesto N-glikozilacije jedino ako je protein podložan N-glikozilaciji. Drugi problem je rešavan pomoću različitih modela mašinskog učenja, kao što su slučajne šume (eng. *Random Forests*), metodi potpornih vektora, pa čak i pomoću neuronskih mreža. U svim slučajevima su korišćeni različiti skupovi proteina, kao i različite osobine proteina koje predstavljaju ulaz za svaki od modela, te je nemoguće adekvatno upoređivanje rezultata kreiranih alata. Takođe, u prethodnim radovima se često problem rešavao za ljudski proteom[8] ili za više različitih vrsta (što još više otežava upoređivanje rezultata); malo radova se baziralo isključivo na biljkama. Pregled najvažnijih prethodnih radova (tj. alata) za nalaženje mesta glikozilacije na proteinu je dat tabelom 2.4.

[8]Proteom je celokupan set proteina izraženih genomom, ćelijom, tkivom ili organizmom [69].

Ime alata i referentni rad	Datum	Vrsta glikozilacije	Organizmi	Modeli MU
NetNGlyc [32]	2002	N-vezana	Čovek	Neuronska mreža
EnsembleGly [8]	2007	C, N, O-vezana	Nekoliko različitih	Ansambl modela potpornih vektora
GPP [34]	2008	O-vezana	Sisari	Slučajne šume
NGlycPred [10]	2012	N-vezana	Eukarioti	Slučajne šume
GlycoMine [42]	2014	C, N, O-vezana	Čovek	Slučajne šume
GlycoMine_PU [43]	2019	C, N, O-vezana	Čovek	Pozitivno i neobeleženo učenje[9]

TABELA 2.4: Pregled prethodnih alata za rešavanje problema nalaženja mesta glikozilacije na proteinu.

[9]Od engleskog termina *PU learning* kreiranog u radu [44].

Glava 3

Podaci i metode

3.1 Nalaženje podataka

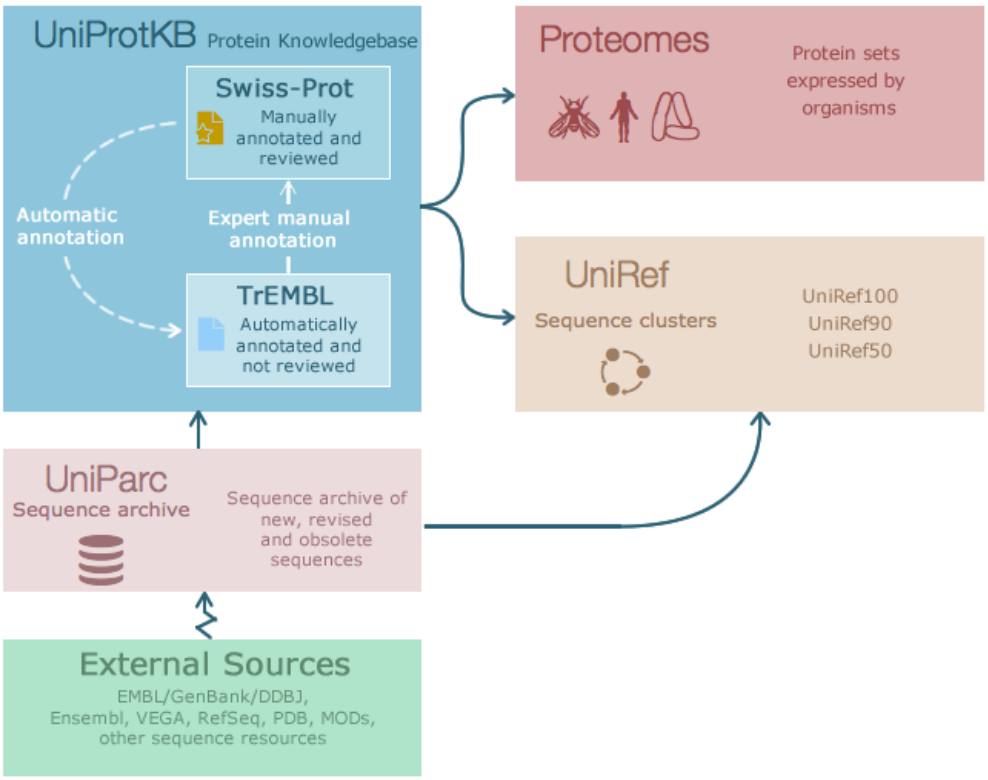
Ogromna količina podataka je postala dostupna početkom 21. veka uvođenjem novih mašina u biološke i hemijske istraživačke laboratorije, smanjujući cenu većine analiza i omogućavajući efikasno rešavanje dotad teških, skupih i vremenski zahtevnih problema. Sa druge strane, povećanje računarskih performansi omogućilo je pouzdanu obradu velikog broja podataka za vrlo kratko vreme. Ovaj ekponencijalni rast podataka je polako uveo bioinformatiku u oblast *Big Data*, a posebno njene discipline koje se nazivaju omikama (npr. genomika, epigenomika, proteomika i sl.), te je doveo i do potrebe za skladištenjem i strukturiranjem bioloških podataka. Ova potreba je rezultovala pojavom različitih baza bioloških podataka od kojih su najpoznatije navedene u tabeli 3.1.

Sadržani podaci	Baze podataka
Nukleotidne i proteinske sekvence	NCBI, EMBL-EBI, UniProt, MMDB
Strukture proteina	DisProt, RCSB PDB, D^2P^2
Genomi	ENSEMBL, SGD, TAIR
Literatura	PubMed, Web of Science

TABELA 3.1: Neke od najpoznatijih bioloških baza podataka.

3.1.1 UniProtKB

U ovom istraživanju je u potpunosti korišćena ***UniProtKB*** (ili samo *UniProt*) baza podataka. *UniProt* predstavlja zajednički rad 3 organizacije: Evropskog instituta za bioinformatiku (*EMBL-EBI*), Švajcarskog instituta za bioinformatiku (*SIB*) i Izvora proteinskih informacija (*PIR*). *EMBL-EBI* i *SIB* su zajedno kreirali ***Swiss-Prot*** i ***TrEMBL*** baze podataka [5], dok je *PIR* kreirao ***PIR-PSD*** [71] bazu podataka. Prve dve baze podataka su postojale nezavisno jedna od druge i nisu bile ekvivalentne, što znači da su se neke sekvence nalazile u obe baze a neke u samo jednoj od njih [26]. Godine 2002., spajanjem ovih dveju baza podataka nastaje najveća, neredundantna, javno dostupna proteinska baza podataka pod nazivom *UniProt*. Kako je sačinjena od dve različite baze podataka, proteinske sekvence i odgovarajuće dodatne informacije su različitog kvaliteta. Glavna razlika je ta što su svi unosi izvorno iz *Swiss-Prot* baze ekspertski pregledani i verifikovani, dok *TrEMBL* predstavlja nadskup sekvenci *Swiss-Prot* baze koji sadrži i podatke koji su računarskom analizom obrađeni, ali ne i provereni ljudskom rukom. Šematski prikaz funkcionisanja *UniProt* baze dat je slikom 3.1.



SLIKA 3.1: Šema *UniProt* baze podataka [28].

Uvid u podatke sadržane u *UniProt* bazi je moguće ostvariti pomoću upita. Upiti se mogu graditi iterativno unošenjem teksta unutar polja za pretragu ili pomoću ugrađenog alata za kreiranje upita koji se otvara preko dugmeta *Advanced*. Upitom je moguće izvući specifične informacije proteinskih sekvenci koje se kasnije mogu i preuzeti u željenom formatu. Pored prethodnog, rezultate upita je moguće oblikovati po želji, eliminisanjem ili dodavanjem kolona rezultujuće tabele, zarad funkcionalnijeg uvida u osobine dobijenog skupa proteina. Preuzimanje podataka se ostvaruje pritiskom na dugme *Download* (iznad rezultujuće tabele), nakon čega je moguće izabrati odgovarajući format za preuzimanje dobijenih rezultata.

Jedna od osobina proteinskih sekvenci koja je dostupna upitom, pomoću *Advanced* polja, jeste i pozicija N-glikozilacije na tim sekvencama.

3.2 Korišćeni alati

U nastavku će biti sažeto opisani alati korišćeni tokom ovog istraživanja.

3.2.1 JupyterLab

Project Jupyter je neprofitna organizacija kreirana sa ciljem pružanja servisa za interaktivan razvoj softvera u više različitih programskih jezika. Nastao je kao nadogradnja *IPython* projekta i predstavlja softver otvorenog koda i sa otvorenim standardima [56, 58]. Trenutno je jedan od najkorišćenijih alata u domenima naučnog izračunavanja i proteže se kako u akademskim, tako i u industrijskim granama.

JupyterLab je jedan od projekta *Project Jupyter* organizacije i predstavlja interaktivno, integrisano okruženje naredne generacije za razvoj softvera. Pored toga što omogućava interaktivnu obradu podataka, *JupyterLab* obezbeđuje fleksibilan i moćan način reprezentacije rezultata, te je zato i korišćen za implementaciju različitih ideja ovog rada. Više o projektu na adresi `https://jupyter.org`.

3.2.2 Python

Za obradu podataka, treniranje modela za predikciju, kao i za prikaz rezultata, korišćen je programski jezik *Python*. Python predstavlja programski jezik opšte namene, visokog nivoa i otvorenog koda (eng. *open source*). Pored toga podržava više različitih programskih paradigmi, kao i različite oblike struktuiranja koda (objektno-orijentisan, funkcionalan, proceduralan). Uobičajeno se koristi za kreiranje aplikacija u velikom broju različitih domena i predstavlja jedan od najkorišćenijih programskih jezika na svetu. Neke od karakterističnih osobina *Python* programskog jezika jesu čitljivost koda, bogate biblioteke funkcija, kao i dizajn koji pospešuje produktivnost programera, kvalitet softvera, njegovu prenosivost i mogućnost integracije [46].

Rezultati ovog rada su proizvod korišćenja više različitih biblioteka u okviru programskog jezika Python. U nastavku će biti izložene neke od intenzivnije korišćenih, zajedno sa glavnim funkcionalnostima koje obezbeđuju.

NumPy

NumPy je biblioteka otvorenog koda programskog jezika *Python* koja obezbeđuje podršku za rad sa velikim, višedimenzionim nizovima i matricama, zajedno sa matematičkim funkcijama visokog nivoa koje se koriste za manipulaciju sa njima [52]. Pored očigledne mogućnosti korišćenja u naučne svrhe, *NumPy* se takođe može koristiti i za efikasno skladištenje proizvoljnih visokodimenzionih podataka [64].

Pandas

Pandas je biblioteka otvorenog koda programskog jezika *Python* koja obezbeđuje podršku za rad sa numeričkim tabelama i vremenskim serijama, korišćenjem struktura podataka koje su istovremeno i visokih performansi i jednostavne za korišćenje. Najčešće se koristi kao dopunska biblioteka prilikom rešavanja problema mašinskog učenja jer omogućava uvoz, izvoz i obradu podataka različitih formata [49, 50].

(A) *NumPy* logo. (B) *Pandas* logo.

SLIKA 3.2: Logo biblioteke *NumPy* i logo biblioteke *Pandas*.

SciKit-Learn

SciKit-Learn je biblioteka otvorenog koda programskog jezika *Python* namenjena za razvoj softvera za rešavanje različitih problema mašinskog učenja (klasifikacija, regresija, klasterovanje, izbor modela, itd.). Pored prethodnog, važna osobina ove biblioteke je i njeno jednostavno korišćenje sa drugim bibliotekama koje se koriste u svrhe naučnog izračunavanja, kao što su *NumPy*, *SciPy*, *Matplotlib* i slične [55].

Matplotlib

Matplotlib je biblioteka otvorenog koda programskog jezika *Python* koja pomoću objektno-orijentisanog interfejsa obezbeđuje iscrtavanje grafika. Pored toga što je cela biblioteka napisana u programskom jeziku *Python*, u velikoj meri se oslanja na podršku *NumPy* biblioteke zarad dobrih performansi čak i pri radu sa velikim nizovima. Glavna ideja biblioteke je kreiranje jednostavnih i funkcionalnih grafika pomoću par linija koda, kao i čuvanje rezultujućih grafika u visokoj rezoluciji [38].

(A) *SciKit-Learn* logo.

(B) *Matplotlib* logo.

SLIKA 3.3: Logo biblioteke *SciKit-Learn* i logo biblioteke *Matplotlib*[1].

Biopython

Biopython je biblioteka otvorenog koda programskog jezika *Python* koja predstavlja veliki skup bioinformatičkih alata za rešavanje raznih bioloških problema. Sadrži podršku za rad sa različitim biološkim sekvencama i mogućnost čitanja i pisanja podataka u različitim formatima. Pored prethodnog, pruža i mogućnosti pristupa spoljnim bazama podataka, iscrtavanja filogenetskih stabala, genomskih dijagrama, makromolekularnih struktura i još mnogo drugih biološki specifičnih elemenata [11].

3.3 Detalji implementacije

U ovom poglavlju će se detaljnije izložiti koraci rešavanja problema, dati u 1.3, počevši od prikupljanja podataka, pa sve do načina prikaza rezultata dobijenih različitim metodama mašinskog učenja.

[1]Logo svake biblioteke je preuzet sa njihovih zvaničnih internet stranica.

SLIKA 3.4: Logo biblioteke *Biopython*.

3.3.1 Preuzimanje podataka

Kao što je i ranije rečeno, podaci u celosti dolaze iz *UniProt* baze podataka. Pristup bazi je izvršen pomoću zvanične stranice i dostupnog veb alata. Upit za izdvajanje proteina za koje se zna da podležu procesu glikozilacije (tj. upit za dobijanje pozitivnog skupa), kreiran je sledećim redosledom:

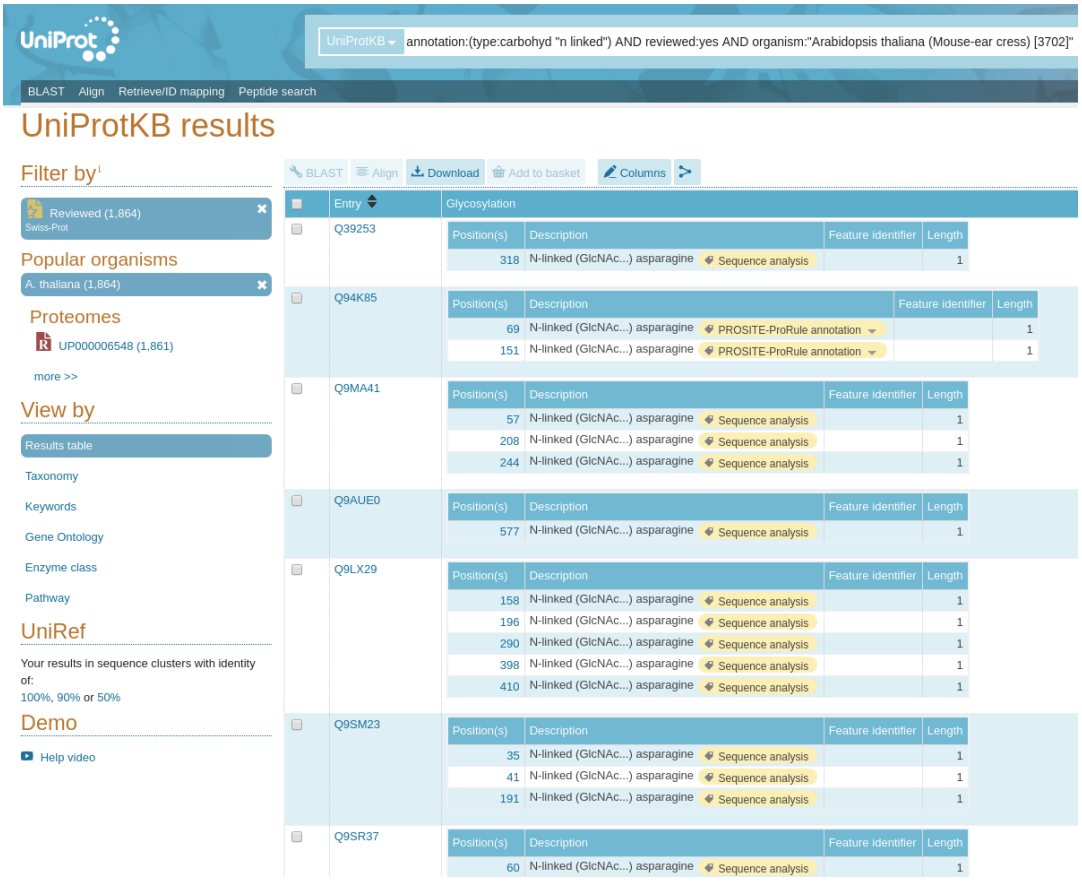
- U polje za pretragu se unosi niska *organism:"Arabidopsis thaliana (Mouse-ear cress) [3702]"* koja iz celokupne baze izdvaja samo proteine organizma *Arabidopsis thaliana*;
- Zatim se pomoću dugmeta *Advanced* dodaje konjunkt prethodnom upitu, pritiskom na dugme **+** pored trenutnog upita, i izborom *AND* opcije iz opadajuće liste. Nakon toga se iz opadajuće liste konjunkta izabere opcija *Reviewed* i podopcija *Reviewed*. Na ovaj način se u obzir uzimaju samo proteini koji su provereni od strane eksperata;
- Nakon što se iz upita izdvoje ekspertski pregledani proteini organizma *Arabidopsis thaliana*, potrebno je izdvojiti samo one proteine koji podležu N-vezanoj glikozilaciji. To se svodi na dodavanje još jednog konjunkta postojećem upitu, na ranije opisan način, pri čemu se iz opadajuće lista bira polje *PTM/Processing*, pa zatim iz podliste odabere *Glycosylation [FT]*. Zatim se u polju *Term* unese niska *n linked*, čime se kompletira upit bazi.

Prethodnim postupkom se dobija pozitivan skup proteina, tj. onaj skup proteina za koje je provereno da podležu N-vezanoj glikozilaciji. Negativan skup proteina, tj. onaj skup proteina za koje je provereno da ne podležu N-vezanoj glikozilaciji, dobija se jednostavnom modifikacijom prethodnog upita. Potrebno je samo konjunkt upita koji se tiče izbora proteina koji su N-glikozilovani, staviti u negaciju. To je moguće jednostavno odraditi menjanjem *AND* vrednosti ispred tog konjunkta u *NOT* vrednost, izborom iz opadajuće liste. Ovim se izdvajaju proteini organizma *Arabidopsis thaliana* koji su ekspertski pregledani i koji ne podležu procesu N-vezane glikozilacije. Na ovaj način se kompletira čitav skup proteina korišćen u ovom radu.

Prethodno kreirane skupove je moguće lako preuzeti u različitim formatima. Da bi se smanjila kasnija obrada podataka, za potrebe ovog rada su izmenjene podrazumevane kolone rezultujuće tabele proteina. To je ostvareno pritiskom na dugme *Columns* i uklanjanjem podrazumevano izabranih kolona sa nazivima: *Entry name*, *Reviewed/Unreviewed*, *Protein names*, *Gene names*, *Organism* i *Length*, i zatim dodavanjem jedne nove kolone, sekcije *PTM/Processing*, sa nazivom *Glycosylation*. Ovim

se ostvaruje da rezultujuća tabela sadrži samo jedinstvene identifikatore proteina i informacije o njihovoj glikozilaciji (mesto glikozilacije, njen tip i eksperimentalni proces kojim je to otkriveno). Naravno, u slučaju negativnog skupa, kolona sa informacijama o glikozilaciji je prazna.

Na kraju, rezultujuće skupove je potrebno preuzeti pomoću dugmeta *Download* u dva formata - FASTA (canonical) i Excel. Prvi format sadrži protein u FASTA obliku (koji će se koristiti u nastavku), dok drugi sadrži jedinstveni identifikator proteina i informacije o njegovoj glikozilaciji, u tabelarnom obliku. Prethodnim postupkom se za ***2019_09*** verziju baze dobija ukupno 1864 proteina pozitivnog skupa i 14032 negativnog skupa.



SLIKA 3.5: Deo rezultujuće *UniProt* tabele pozitivnog skupa proteina [27].

Rezultujuće skupove korišćene u radu moguće je videti i na javno dostupnom *GitHub* repozitorijumu, u svojim izvornim formatima, na sledećoj adresi: `https://github.com/AAnzel/Master_rad/tree/master/data`.

3.3.2 Kreiranje osobina podataka

Nakon preuzimanja, svaku proteinsku sekvencu je potrebno dodatno obraditi. Neki proteini mogu u svojoj sekundarnoj strukturi sadržati AK sa oznakom **X** koja označava nepoznatu AK. Pre kreiranja osobina, potrebno je eliminisati one proteine koji u svojoj sekundarnoj strukturi sadrže takvu oznaku. Rezultati obrade pokazuju da negativan skup ima dva proteina koji u sebi sadrže **X** oznaku, dok pozitivan skup nema takvih proteina.

Tek po završetku prethodnog postupka moguće je krenuti sa kreiranjem fizičko-hemijskih osobina svakog od proteina. Za kreiranje jednog dela osobina korišćena je biblioteka *Biopython*, dok je za kreiranje drugog dela korišćeno javno dostupno uputstvo istraživačke grupe *Systems Biology Research Group* Univerziteta u San Dijegu [31]. Ukupno je kreirano 43 osobina, pri čemu je važno napomenuti da ovaj broj sadrži 20 atributa koji odgovaraju osobini „Udeo svake AK u proteinu" (po jedan atribut za udeo svake AK), kao i 3 atributa za osobinu „Udeo sekundarne strukture". Ostali atributi su navedeni po stavkama kao osobine. U nastavku će biti navedene kreirane osobine, kao i objašnjenja nekih od njih.

Osobine kreirane pomoću Biopython biblioteke [4] su:

- **Molekularna težina proteina**;
- **GRAVY (skraćeno eng. *Grand average of hydropathicity*)**: predstavlja jednu od mera hidrofobičnosti proteina, razvijena u radu [41];
- **Aromatičnost**: računa vrednost aromatičnosti proteina, po formuli kreiranoj u radu [45];
- **Indeks nestabilnosti**: testira stabilnost proteina, na osnovu formule kreirane u radu [33];
- **Udeo svake AK u proteinu**;
- **Izoelektrična tačka**;
- **Udeo sekundarne strukture**: predstavlja trojku udela AK koje su najčešće na heliksu, zavojnici ili ploči.

Osobine kreirane na osonovu javno dostupnog uputstva (funkcija *broj(α)* u nastavku računa broj pojavljivanja AK α u proteinu, gde je α njena jednoslovna oznaka):

- **Naelektrisanje**: računa se po formuli

$$broj(K) + broj(R) - broj(D) \tag{3.1}$$

- **Apsolutno naelektrisanje**: apsolutna vrednost formule 3.1;
- **Prosečno naelektrisanje**: prosečna vrednost formule 3.1;
- **Apsolutno prosečno naelektrisanje**: apsolutna vrednost prethodne formule;
- **Udeo alifatičkih AK**: alifatičke AK su one čije su jednoslovne oznake A, G, I, L, P i V;
- **Udeo nenaelektrisanih polarnih AK**: nenaelektrisane polarne AK su one čije su jednoslovne oznake S, T, N i Q;
- **Udeo polarnih AK**: polarne AK su one čije su jednoslovne oznake Q, N, H, S, T, Y, C, M i W;
- **Udeo hidrofobnih AK**: hidrofobne AK su one čije su jednoslovne oznake A, G, I, L, P, V i F;
- **Udeo pozitivno naelektrisanih AK**: pozitivno naelektrisane AK su one čije su jednoslovne oznake H, K i R;

- **Udeo sumpornih AK**: sumporne AK su one čije su jednoslovne oznake C i M;
- **Udeo negativno naelektrisanih AK**: negativno naelektrisane AK su one čije su jednoslovne oznake D i E;
- **Udeo amidnih AK**: amidne AK su one čije su jednoslovne oznake N i Q;
- **Udeo alkoholnih AK**: alkoholne AK su one čije su jednoslovne oznake S i T;

Pored ovih osobina (koje možemo posmatrati i kao kolone rezultujućih tabela) dodaju se i još tri dodatne:

- **Postojanje motiva glikozilacije**: sadrži vrednosti 0 ili 1 u zavisnosti od postojanja motiva Asn–X–Ser/Thr (jednoslovno: N-X-S/T)[2] u proteinskoj sekvenci (X predstavlja bilo koju AK osim Prolina);
- **Broj motiva glikozilacije**: sadrži ukupan broj motiva Asn–X–Ser/Thr (jednoslovno: N-X-S/T) u proteinskoj sekvenci;
- **Klasa**: sadrži vrednosti 0 ili 1 koje predstavljaju pripadnost proteina negativnom (u oznaci 0), tj. pozitivnom (u oznaci 1) skupu.

Kolona „Postojanje motiva glikozilacije" omogućava identifikovanje onih proteina koji uopšte ne sadrže motiv N-glikozilacije, što se naravno može desiti samo kod proteina iz negativnog skupa. Samim tim, može se posmatrati skup podataka koji ne sadrži takve proteine, jer se svakako proces N-glikozilacije ne može desiti nad njima. Ovim se smanjuje izvorni skup podataka eliminišući nepotrebne podatke, u cilju lakšeg procesa učenja klasifikatora. Dalji rad je dakle sproveden nad skupom podataka iz kog su izbačeni proteini bez motiva N-glikozilacije, kao i bez kolone pod nazivom „Postojanje motiva glikozilacije". Eksperimentalno je utvrđeno da svi proteini bez motiva N-glikozilacije pripadaju negativnom skupu proteina, kao i da ih je ukupno 3186. Možemo primetiti da izbacivanje kolone „Postojanje motiva glikozilacije" ne smanjuje informativnost skupa podataka, jer je ona međuzavisna sa kolonom „Broj motiva glikozilacije". Ovo je moguće uočiti na slici 3.6.

Sledeći korak pripreme podataka predstavlja računanje i unošenje prethodno navedenih osobina proteina, kreirajući finalnu tabelu pozitivnog i finalnu tabelu negativnog skupa. Redovi ovih tabela su proteinske sekvence, indeksirane na osnovu njihovih identifikatora, sa kolonama koje sadrže sračunate osobine odgovarajućih sekvenci. Deo finalne tabele pozitivnog skupa je predstavljen tabelama 3.2 i 3.3.

Nakon kreiranja tabela, vrši se analiza dobijenih osobina ispitivanjem njihove korelisanosti. Ispitivanje korelisanosti osobina nam ukazuje na informativnost neke osobine - ako je neka osobina u visokoj korelaciji sa nekom drugom osobinom, onda se jedna od njih može izbaciti iz razmatranja bez velikog gubljenja informativnosti podataka [51]. Slika 3.6 predstavlja matricu korelacije kreiranih atributa. Vidimo da matrica ima veliku korelisanost samo po dijagonali, što zapravo ukazuje na nisku korelisanost kreiranih osobina.

3.3.3 Priprema podataka za metode mašinskog učenja

Rezultat prethodno opisanog postupka su dve tabele koje odgovaraju pozitivnom i negativnom skupu proteina, zajedno sa njihovim osobinama. Kreiranje ulaza za modele mašinskog učenja se dalje sprovodi konkatenacijom ove dve tabele u

[2] Ovaj motiv ćemo u nastavku zvati motivom N-glikozilacije, ili samo motivom glikozilacije.

Identifikator proteina	Molekularna tezina	Gravy	Aromaticnost	Indeks nestabilnosti	Udeo „A" u proteinu	Udeo „C" u proteinu	...
Q8VWF6	44036.0068	-0.239231	0.117949	35.844359	0.048718	0.012821	...
Q94F09	49850.3731	-0.155000	0.106818	34.868682	0.065909	0.009091	...
Q9SRM3	45619.3058	-0.539250	0.082500	47.761025	0.035000	0.010000	...
Q9SLC4	24422.1884	-0.344700	0.082949	59.852995	0.046083	0.027650	...
Q9LZV7	104177.5685	0.064840	0.065150	35.501448	0.052740	0.020683	...
C0LGD6	94979.5541	-0.192840	0.098592	41.125012	0.041080	0.015258	...
Q8H0W9	38741.5363	-0.179586	0.094675	34.552071	0.044379	0.011834	...
Q9LIF1	11055.9367	0.453922	0.078431	42.627549	0.107843	0.029412	...
Q9FMG1	40479.4328	-0.463812	0.074586	46.033978	0.077348	0.033149	...
Q4V3E0	33151.0884	-0.375472	0.047170	55.311950	0.062893	0.006289	...
Q8LBS4	19124.6860	-0.010615	0.055866	39.008939	0.055866	0.022346	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

TABELA 3.2: Prvi deo finalne tabele pozitivnog skupa.

...	Udeo hidrofobnih ak.	Udeo pozitivnih ak.	Udeo supornih ak.	Udeo negativnih ak.	Udeo amidnih ak.	Udeo alkoholnih ak.	Broj mesta glikozilacije
...	0.471795	0.146154	0.033333	0.123077	0.061538	0.112821	2
...	0.463636	0.138636	0.027273	0.156818	0.047727	0.122727	1
...	0.425000	0.137500	0.035000	0.135000	0.107500	0.110000	2
...	0.410138	0.142857	0.036866	0.133641	0.064516	0.184332	3
...	0.495346	0.103413	0.031024	0.098242	0.085832	0.163392	18
...	0.433099	0.111502	0.034038	0.098592	0.103286	0.173709	13
...	0.446746	0.162722	0.029586	0.118343	0.068047	0.127219	1
...	0.529412	0.117647	0.068627	0.078431	0.049020	0.117647	1
...	0.414365	0.149171	0.063536	0.116022	0.091160	0.127072	1
...	0.503145	0.116352	0.044025	0.078616	0.075472	0.169811	2
...	0.441341	0.089385	0.050279	0.067039	0.106145	0.212291	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

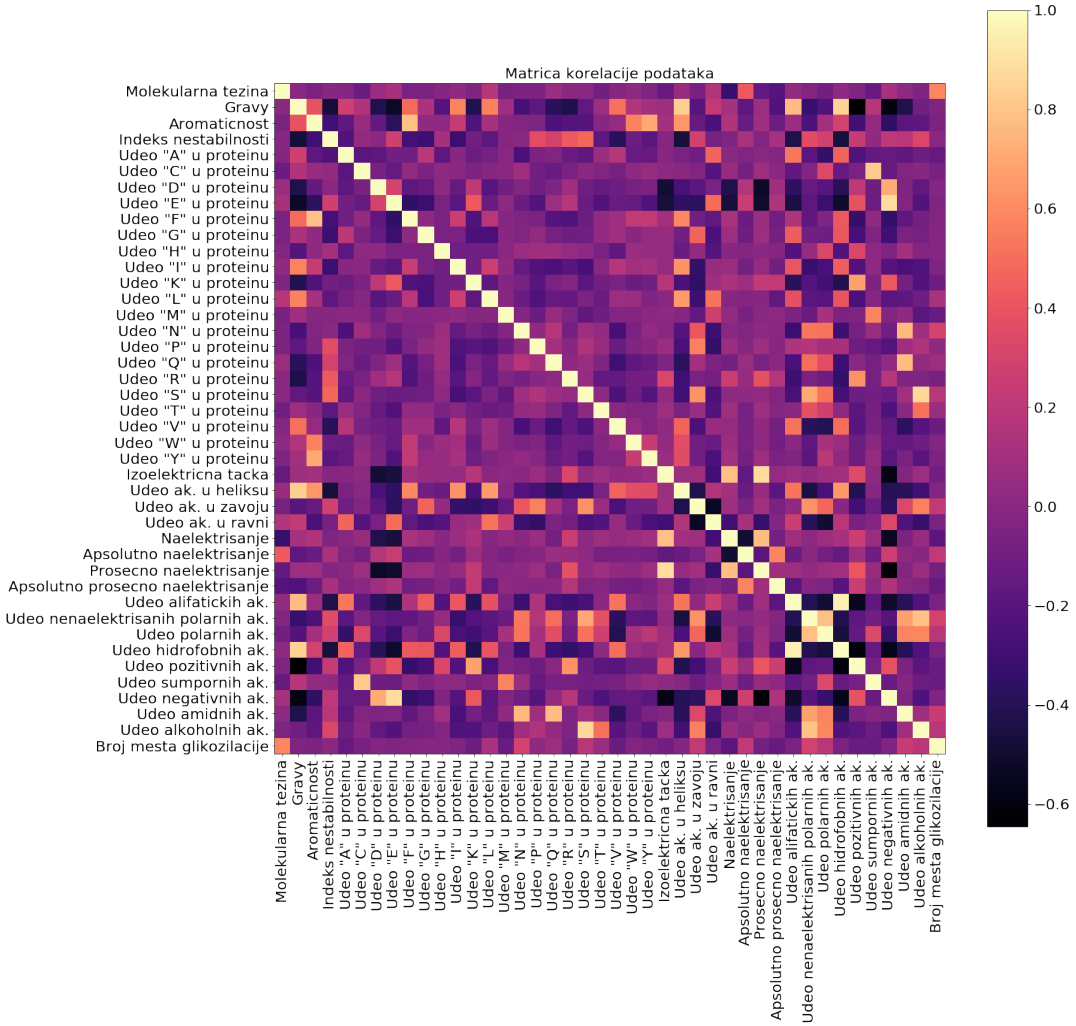
TABELA 3.3: Drugi deo finalne tabele pozitivnog skupa.

jednu, i preraspodelom redova rezultujuće tabele. Na ovaj način se dobija tabela koja sadrži proizvoljno alternirajuće redove proteina pozitivnog i negativnog skupa, čiji su indeksi jedinstveni identifikatori proteina definisani *UniProt* bazom podataka. Balansiranost klasa, tj. odnos negativnih naspram pozitivnih proteina je predstavljen slikom 3.7. Na slici je jasno uočljiva nebalansiranost klasa kako pre, tako i nakon izbacivanja proteina bez motiva glikozilacije. Razlog nebalansiranosti klasa nakon ove obrade je taj što se izbacuje mali deo skupa negativnih proteina (njih 3186) koji ne utiče značajno na polaznu balansiranost. Broj od 3186 negativnih proteina predstavlja približno 22% celokupnog negativnog skupa proteina (kojih izvorno ima 14032), te odnos pozitivne i negativne klase u oba slučaja ostaje približno isti (slika 3.7):

1. **Pre izbacivanja proteina bez motiva N-glikozilacije**: broj pozitivnih = 1864, broj negativnih = 14032
   BROJ POZITIVNIH : BROJ NEGATIVNIH = 1 : 7.52;

2. **Nakon izbacivanja proteina bez motiva N-glikozilacije**: broj pozitivnih = 1864, broj negativnih = 10846
   BROJ POZITIVNIH : BROJ NEGATIVNIH = 1 : 5.82.

Jedna interpretacija odnosa bi mogla biti ta da je pre obrade približno svaki 7.

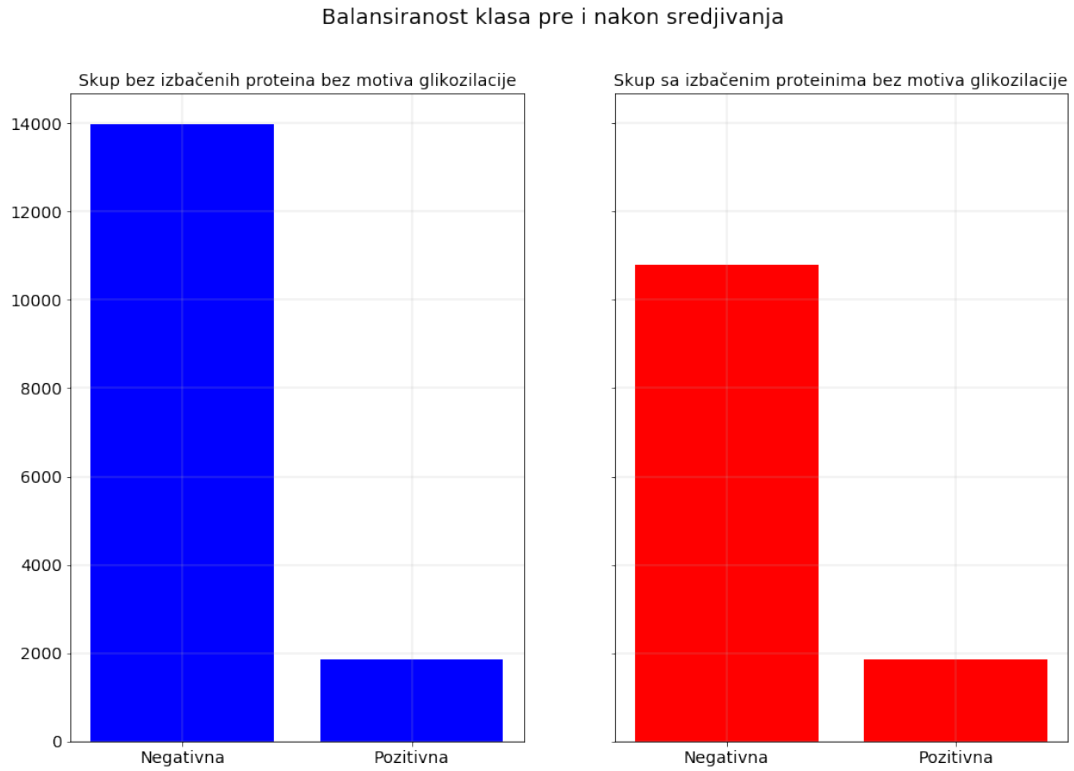


SLIKA 3.6: Matrica korelacije kreiranih osobina objedinjenih finalnih tabela.

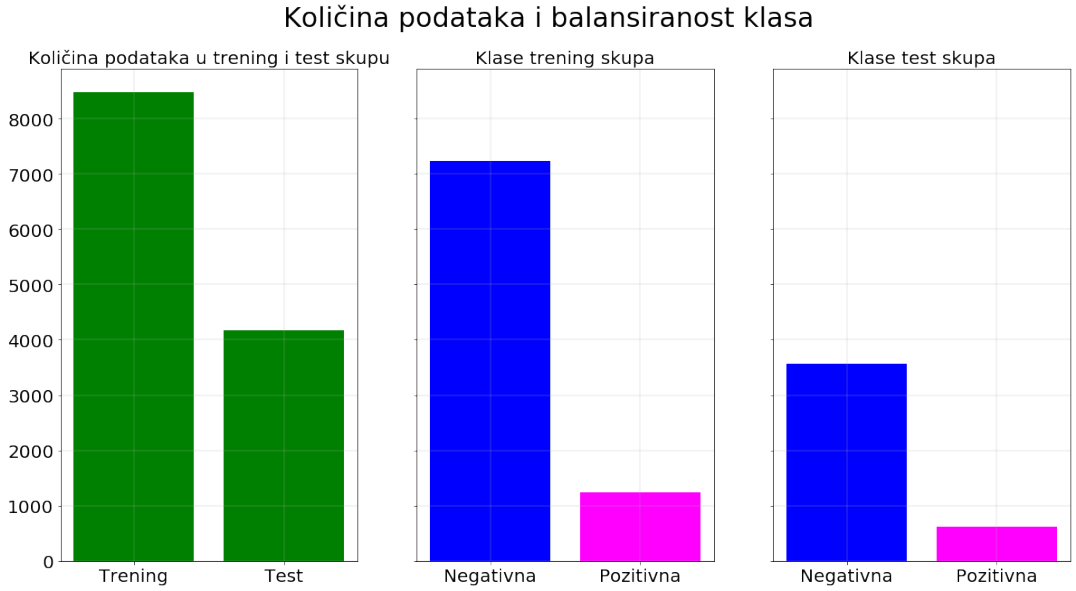
protein iz pozitivnog skupa, dok je nakon obrade približno svaki 6. protein iz pozitivnog skupa. Dakle, izbacivanjem proteina bez motiva N-glikozilacije se smanjuje skup negativnih proteina, ali nedovoljno da ukloni polaznu nebalansiranost klasa. Ovim se opravdava korišćenje tehnika balansiranja podataka u svrhe dobijanja efikasnijih modela MU.

Sledeći korak predstavlja odvajanje kolone pod nazivom „Klasa" od ostatka tabele, jer ona predstavlja niz ciljnih vrednosti klasifikatora. Ostatak tabele se deli na 3 skupa: trening skup, validacioni skup i test skup. Trening skup se koristi u svrhe treniranja modela, validacioni u svrhe nalaženja optimalnih hiperparametara, a test skup (u kombinaciji sa prethodna dva skupa) u svrhe evaluacije modela. Deljenje skupova se vrši uz metodu stratifikacije po ciljnoj vrednosti (klasi) [51], poštujući odnos 2:1 pri svakom deljenju. Ovo je predstavljeno slikom 3.8. Takođe, podaci svih skupova se standardizuju pomoću metoda dostupnog u *SciKit-Learn* biblioteci, zarad bržeg i efikasnijeg procesa učenja modela [51].

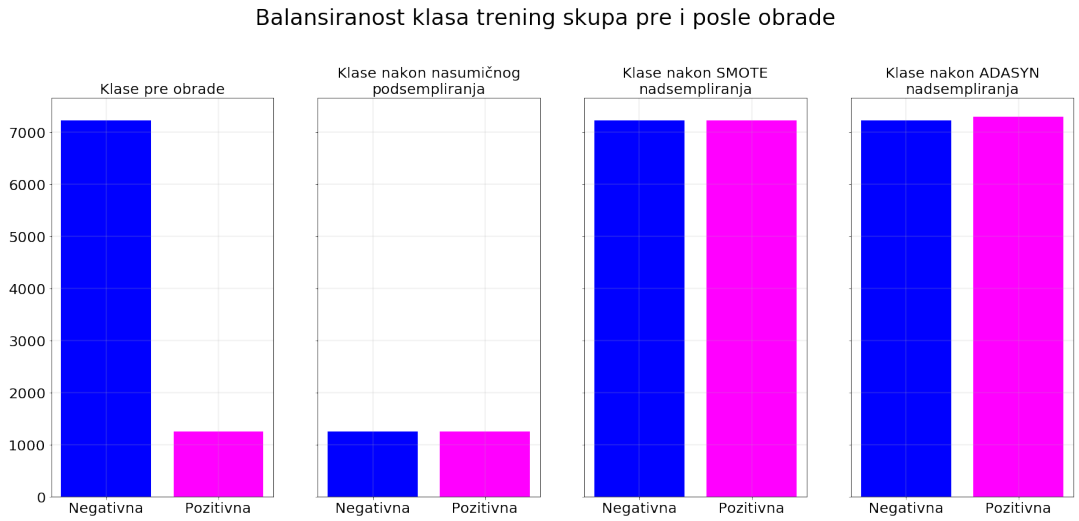
Sledeći korak je primena različitih metoda balansiranja podataka na osnovu odnosa klasa. Redom se primenjuju tri različite metode, pomoću bibliotečkih funkcija, i to na sledeći način:



SLIKA 3.7: Balansiranost klasa skupa podataka pre i nakon izbacivanja proteina bez motiva N-glikozilacije.



SLIKA 3.8: Prva slika prikazuje razmeru 2:1 nakon deljenja izvornog skupa na trening i test skup. Druga slika prikazuje balansiranost klasa trening skupa, a treća test skupa. Polazna balansiranost klasa je očuvana usled korišćenja metode stratifikacije po klasi.



SLIKA 3.9: Balansiranost klasa trening skupa pre obrade, nakon podsempliranja, nakon SMOTE nadsempliranja i nakon ADASYN nadsempliranja.

1. Balansiranje nasumičnim podsempliranjem trening skupa (koji se koristi za treniranje) i trening_valid skupa[3] (koji se koristi pri evaluaciji);
2. Balansiranje SMOTE nadsempliranjem trening i trening_valid skupa;
3. Balansiranje ADASYN nadsempliranjem trening i trening_valid skupa.

Rezultat primena ovih metoda nad izvornim skupom podataka predstavljen je slikom 3.9.

3.3.4 Nalaženje hiperparametara

U cilju otkrivanja da li je protein N-glikozilovan ili ne, koriste se klasifikacione verzije MPV i potpuno povezane neuronske mreže (u nastavku NM). Ovi modeli se treniraju nad različito balansiranim skupovima podataka i sa različitim hiperparametrima, u cilju nalaženja onih hiperparametara sa kojima modeli daju najbolje rezultate. Skup hiperparametara jednog modela se drugačije naziva i njegovom konfiguracijom.

Nalaženje hiperparametara MPV

Za nalaženje optimalnih hiperparametara MPV korišćen je metod ugnježdene unakrsne validacije, dostupan u sklopu *Scikit-Learn* biblioteke pod nazivom *GridSearchCV*. U zavisnosti od kompleksnosti modela, a samim tim i vremenskog trajanja njegovog treniranja, koriste se različite ugnježdene unakrsne validacije i to: pri radu sa linearnim MPV koristi se 10-ostruka, a pri radu sa kernelizovanim MPV 5-ostruka ugnježdena unakrsna validacija. Redom se isprobavaju linearni MPV pa kernelizovani, pri čemu se koriste njihove implementacije dostupne bibliotekom *Scikit-Learn*. Hiperparametri koji se variraju, zajedno sa vrednostima koje uzimaju su:

[3]Trening_valid skup predstavlja objedinjen trening skup sa validacionim skupom koji služi za treniranje modela za koji su pronađeni optimalni hiperparametri. Zatim se pomoću test skupa vrši evaluacija takvog modela.

- **kernel** $\in$ {poly, rbf, sigmoid} za kernelizovane modele;
- **C** $\in \{10^{-5}, 10^{-4}, ..., 10^{5}\}$ za linearne i kernelizovane modele;
- **gamma** $\in \{10^{-5}, 10^{-4}, ..., 10^{5}\}$ za kernelizovane modele.

Zatim se isprobavaju različiti modeli (ukupno 10) i to sledećim redosledom:

1. Linearni MPV **bez** i **sa** uključenim *class_weight*[4] parametrom nad skupom podataka kome klase **nisu eksplicitno balansirane** nekom od metoda;
2. Kernelizovani MPV **bez** i **sa** uključenim *class_weight* parametrom nad skupom podataka kome klase **nisu eksplicitno balansirane** nekom od metoda;
3. Linearni i kernelizovani MPV nad skupom podataka kome su klase izbalansirane **nasumičnim podsempliranjem**;
4. Linearni i kernelizovani MPV nad skupom podataka kome su klase izbalansirane **SMOTE nadsempliranjem**;
5. Linearni i kernelizovani MPV nad skupom podataka kome su klase izbalansirane **ADASYN nadsempliranjem**.

Nalaženje hiperparametara NM

Za nalaženje optimalnih hiperparametara NM neadekvatno je koristiti metod ugnježdene unakrsne validacije, usled dužine procesa učenja ovih modela. Zato se nalaženje hiperparametara ovih modela sprovelo nasumičnim izborom validacionog skupa i evalucijom modela nad njim, prilikom variranja hiperparametara. U nastavku su opisane karakteristike nalaženja optimalnih hiperparametara NM. Veći deo naredne implementacije oslanjao se na biblioteku *Keras*.

Svaka od mreža ima ulaznu dimenziju jednaku broju osobina proteina (tj. tačno 42) a izlaznu dimenziju jednaku 1. Jedan od hiperparametara koji se varira je i arhitektura mreže, tj. broj skrivenih slojeva, kao i broj neurona u svakom od njih.

Pored prethodnog, aktivaciona funkcija svakog skrivenog sloja je ispravljena linearna jedinica, a poslednjeg sloja je sigmoidna. Kao funkcija greške koristi se binarna unakrsna entropija, a metrika koja se prati je tačnost binarne klasifikacije. Između skrivenih slojeva se koristi unutrašnja standardizacija podataka[5] (sa podrazumevanim vrednostima parametara). Veličina grupe podataka koja se istovremeno trenira je 32, što znači da se ovaj hiperparametar ne varira. Maksimalan broj epoha dat za treniranje je 200, pri čemu on može biti i manji jer se kao vid regularizacije koristi i rano zaustavljanje, sa strpljenjem od 13 epoha, prilikom praćenja funkcije greške na skupu za validaciju. Ostali hiperparametri se određuju pomoću nasumično odabranog validacionog skupa, prateći F1 meru (na kraju svake epohe), pri čemu važi sledeće:

- **kernel_initializer** $\in$ {he_normal, he_uniform, glorot_normal, glorot_uniform};
- **optimizer** $\in$ {RMSprop, Nadam};

[4] *Class_weight* parametar je takođe jedan od argumenata bibliotečkog MPV koji se često koristi pri postojanju nebalansiranosti klasa. Njegova uloga je da naglasi modelu da prilikom odlučivanja uzme u obzir postojeću nebalansiranost. Ovaj metaparametar predstavlja vid balansiranja klasa na algoritamskom nivou.

[5] Omogućava standardizaciju podataka između unutrašnjih slojeva. U praksi se pokazuje da ova tehnika dovodi do boljih rezultata.

- **broj_neurona_1** $\in$ {30, 25, 20};
- **broj_neurona_2** $\in$ {18, 15, 13}.

Zatim se isprobavaju različiti modeli (ukupno 10) i to sledećim redosledom:

1. Potpuno povezana neuronska mreža **bez** uključenog *class_weight* parametra nad skupom podataka kome klase **nisu eksplicitno balansirane** nekom od metoda, sa **jednim i dva skrivena sloja**;
2. Potpuno povezana neuronska mreža **sa** uključenim *class_weight* parametrom nad skupom podataka kome klase **nisu eksplicitno balansirane** nekom od metoda, sa **jednim i dva skrivena sloja**;
3. Potpuno povezana neuronska mreža nad skupom podataka kome su klase izbalansirane **nasumičnim podsempliranjem**, sa **jednim i dva skrivena sloja**;
4. Potpuno povezana neuronska mreža nad skupom podataka kome su klase izbalansirane **SMOTE nadsempliranjem**, sa **jednim i dva skrivena sloja**;
5. Potpuno povezana neuronska mreža nad skupom podataka kome su klase izbalansirane **ADASYN nadsempliranjem**, sa **jednim i dva skrivena sloja**.

Takođe, za gore navedene modele, važi sledeće:

- Prilikom kreiranja mreže sa jednim skrivenim slojem, broj neurona tog sloja se uzima iz skupa *broj_neurona*_1 $\cup$ *broj_neurona*_2;
- Prilikom kreiranja mreže sa dva skrivena sloja, broj neurona prvog sloja se uzima iz skupa *broj_neurona*_1, a drugog iz skupa *broj_neurona*_2.

Izgled mreže sa dva skrivena sloja i optimalnim vrednostima hiperparametara, u formatu *Keras* biblioteke, dat je u nastavku.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 30)                1260
_________________________________________________________________
batch_normalization_1 (Batch (None, 30)                120
_________________________________________________________________
activation_1 (Activation)    (None, 30)                0
_________________________________________________________________
dense_2 (Dense)              (None, 18)                558
_________________________________________________________________
batch_normalization_2 (Batch (None, 18)                72
_________________________________________________________________
activation_2 (Activation)    (None, 18)                0
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 19
_________________________________________________________________
batch_normalization_3 (Batch (None, 1)                 4
_________________________________________________________________
activation_3 (Activation)    (None, 1)                 0
=================================================================
Total params: 2,033
Trainable params: 1,935
Non-trainable params: 98
_________________________________________________________________
```

Hiperparametri koji se variraju kod NM predstavljaju deo većeg skupa hiperparametara koji se koristio inicijalno. Veći skup hiperparametara (prikazujući samo one hiperparametre čiji se skupovi razlikuju od redukovanih) je bio oblika:

- **kernel_initializer** ∈ {uniform, normal, lecun_normal, lecun_uniform, he_normal, he_uniform, glorot_normal, glorot_uniform};
- **optimizer** ∈ {SGD, RMSprop, Adagrad, Adadelta, Adam, Adamax, Nadam};
- **vel_grupe** ∈ {32, 64};
- **broj_neurona_1** ∈ {30, 25, 20};
- **broj_neurona_2** ∈ {18, 15, 13};
- **broj_neurona_3** ∈ {10, 8, 5}.

Ovo znači da je na početku isprobavan veliki broj NM, među kojima su i one sa 3 skrivena sloja, gde se broj neurona trećeg sloja uzimao iz skupa *broj_neurona_3*. Kako ove mreže nisu dale zavidne rezultate, izbačene su iz daljeg razmatranja, zajedno sa hiperparametrom *broj_neurona_3*. Skupovi ostalih hiperparametara su takođe redukovani na osnovu zapažanja inicijalnih rezultata - one vrednosti hiperparametara koje se nikada nisu javile kao optimalne su izbačene iz razmatranja. Ovim su u procesu traženja optimalnih vrednosti ostale samo one koje su najperspektivnije, te se drastično smanjio proces treniranja NM pri očuvanju mogućnosti izbora različitih vrednosti hiperparametara prilikom učenja.

3.3.5 Prikaz rezultata

U sklopu svake evaluacije svih modela, iscrtava se matrica konfuzije a zatim i ispisuje klasifikacioni izveštaj. Izveštaj klasifikacije omogućava jasan prikaz uspešnosti modela tako što za svaki model ispisuje njegovu preciznost, odziv i F1 meru. Poslednja kolona izveštaja sadrži tačan broj proteinskih sekvenci u svakoj od klasa. Za NM se zajedno sa izveštajem iscrtavaju i grafici binarne tačnosti i greške tokom učenja. Primer izveštaja klasifikacije, u formatu *Scikit-Learn* biblioteke, dat je ispod.

```
              precision    recall  f1-score   support

           0       0.97      0.81      0.88      4612
           1       0.36      0.81      0.50       615

    accuracy                           0.81      5227
   macro avg       0.67      0.81      0.69      5227
weighted avg       0.90      0.81      0.84      5227
```

Glava 4

Rezultati

U ovoj glavi biće prezentovani rezultati modela, redosledom datim u poglavlju 3.3.4, na način opisan u poglavlju 3.3.5. Prvo će biti izloženi klasifikacioni izveštaji u obliku tabele, a zatim i matrice konfuzija najboljih modela. Posebno u slučaju NM, biće dati i grafici učenja najboljih modela. Zarad lakšeg praćenja rezultata, matrice konfuzija i grafici učenja najboljih modela su izloženi u dodatku A.

Kao posledica nebalansiranosti klasa izvornog skupa podataka, modeli lako uče koje karakteristike čine da protein pripada negativnom skupu. Usled znatno manjeg broja proteina pozitivnog skupa, klasifikatori imaju poteškoća pri učenju karakteristika tih proteina. Iz ovog razloga, u tabelama klasifikacionih izveštaja ove glave, crvenom bojom biće naglašeni redovi koji se odnose na uspešnost klasifikatora u radu sa manjinskom klasom (koja predstavlja skup pozitivnih proteina, sa oznakom 1). Takođe, isti redovi koji se odnose na najbolje modele biće naglašeni podebljanim tekstom.

Celokupan rad, zajedno sa rezultatima prezentovanim u ovoj glavi, moguće je videti i na javno dostupnom *GitHub* repozitorijumu, u obliku *Jupyter notebook* datoteka sa ekstenzijom **.ipynb**, na adresi: `https://github.com/AAnzel/Master_rad/tree/master/src`.

4.1 Rezultati modela potpornih vektora

Tabelama 4.1, 4.2 i 4.3 su prezentovani klasifikacioni izveštaji svih MPV.

Najbolje rezultate daju kernelizovani modeli nad polaznim ali i nad nadsempliranim skupovima podataka. U nastavku su navedeni optimalni hiperparametri ovih modela, a u dodatku A i odgovarajuće matrice konfuzija (slike A.1, A.2, A.3 i A.4):

1. **Kernelizovan model, nebalansiran, bez *class_weight***: C = 10, gamma = 0.1, kernel = rbf;
2. **Kernelizovan model, nebalansiran, sa *class_weight***: C = 1, gamma = 0.1, kernel = rbf;
3. **Kernelizovan model, balansiran, SMOTE**: C = 10, gamma = 0.1, kernel = rbf;
4. **Kernelizovan model, balansiran, ADASYN**: C = 10, gamma = 0.1, kernel = rbf.

Možemo primetiti još da su optimalni hiperparametri jednaki, osim kod drugog navedenog modela (hiperparametar C), te je moguće dobiti potencijalno bolje rezultate traženjem novih hiperparametara u okolini dobijenih optimalnih.

Model	Info	Prec.	Odziv	F1 mera	Broj instanci
Linearan model model nebalansiran bez class_weight	0	0.93	0.86	0.90	3561
	1	0.45	0.65	0.53	614
	tačnost			0.83	4175
	makro prosek	0.69	0.76	0.71	4175
	otežan prosek	0.86	0.83	0.84	4175
Linearan model nebalansiran sa class_weight	0	0.95	0.82	0.88	3561
	1	0.42	0.74	0.53	614
	tačnost			0.81	4175
	makro prosek	0.68	0.78	0.71	4175
	otežan prosek	0.87	0.81	0.83	4175
Kernelizovan model nebalansiran bez class_weight	0	0.94	0.99	0.97	3561
	1	**0.90**	**0.66**	**0.76**	**614**
	tačnost			0.94	4175
	makro prosek	0.92	0.82	0.86	4175
	otežan prosek	0.94	0.94	0.94	4175
Kernelizovan model nebalansiran sa class_weight	0	0.95	0.97	0.96	3561
	1	**0.81**	**0.73**	**0.77**	**614**
	tačnost			0.94	4175
	makro prosek	0.88	0.85	0.87	4175
	otežan prosek	0.93	0.94	0.93	4175

TABELA 4.1: Klasifikacioni izveštaji MPV sa skupovima podataka koji nisu eksplicitno balansirani.

Model	Info	Prec.	Odziv	F1 mera	Broj instanci
Linearan model balansiran podsempliranje	0	0.95	0.79	0.87	3561
	1	0.39	0.76	0.52	614
	tačnost			0.79	4175
	makro prosek	0.67	0.78	0.69	4175
	otežan prosek	0.87	0.79	0.81	4175
Kernelizovan model balansiran podsempliranje	0	0.97	0.90	0.93	3561
	1	0.58	0.82	0.68	614
	tačnost			0.89	4175
	makro prosek	0.77	0.86	0.81	4175
	otežan prosek	0.91	0.89	0.89	4175

TABELA 4.2: Klasifikacioni izveštaji MPV sa skupom podataka koji je balansiran podsempliranjem.

Model	Info	Prec.	Odziv	F1 mera	Broj instanci
Linearan model balansiran SMOTE	0	0.95	0.80	0.87	3561
	1	0.39	0.76	0.52	614
	tačnost			0.79	4175
	makro prosek	0.67	0.78	0.69	4175
	otežan prosek	0.87	0.79	0.82	4175
Kernelizovan model balansiran SMOTE	0	0.94	0.98	0.96	3561
	1	**0.87**	**0.67**	**0.76**	**614**
	tačnost			0.94	4175
	makro prosek	0.91	0.82	0.86	4175
	otežan prosek	0.93	0.94	0.93	4175
Linearan model balansiran ADASYN	0	0.97	0.63	0.77	3561
	1	0.29	0.89	0.44	614
	tačnost			0.67	4175
	makro prosek	0.63	0.76	0.60	4175
	otežan prosek	0.87	0.67	0.72	4175
Kernelizovan model balansiran ADASYN	0	0.94	0.98	0.96	3561
	1	**0.87**	**0.66**	**0.75**	**614**
	tačnost			0.94	4175
	makro prosek	0.91	0.82	0.86	4175
	otežan prosek	0.93	0.94	0.93	4175

TABELA 4.3: Klasifikacioni izveštaji MPV sa skupovima podataka koji su balansirani nadsempliranjem.

4.2 Rezultati potpuno povezanih neuronskih mreža

Tabelama 4.4, 4.5 i 4.6 su prezentovani klasifikacioni izveštaji svih NM.

U slučaju NM najbolje rezultate daju modeli nad izvornim skupom podataka. Ovi modeli daju čak i bolje rezultate od modela koji rade nad nadsempliranim skupovima. Mogući razlog tome je potencijalna preosetljivost mreža na sintetičke podatke, što dovodi do teškog učenja karakterističnosti izvornog skupa. U nastavku su navedeni optimalni hiperparametri nekih od modela, a u dodatku A i odgovarajuće matrice konfuzija sa graficima učenja (slike A.5, A.6, A.7 i A.8):

1. **Dva skrivena sloja, nebalansirana, bez *class_weight***: kernel_initializer = glorot_uniform, optimizer = Nadam, batch_size = 32, broj_neurona_1 = 20, broj_neurona_2 = 13;

2. **Dva skrivena sloja, nebalansirana, sa *class_weight***: kernel_initializer = glorot_uniform, optimizer = Nadam, batch_size = 32, broj_neurona_1 = 30, broj_neurona_2 = 13.

Model	Info	Prec.	Odziv	F1 mera	Broj instanci
Jedan skriveni sloj nebalansirana bez class_weight	0	0.93	0.97	0.95	3561
	1	0.78	0.60	0.68	614
	tačnost			0.92	4175
	makro prosek	0.85	0.78	0.81	4175
	otežan prosek	0.91	0.92	0.91	4175
Dva skrivena sloja nebalansirana bez class_weight	0	0.93	0.97	0.95	3561
	1	**0.77**	**0.61**	**0.68**	**614**
	tačnost			0.92	4175
	makro prosek	0.85	0.79	0.82	4175
	otežan prosek	0.91	0.92	0.91	4175
Jedan skriveni sloj nebalansirana sa class_weight	0	0.94	0.97	0.95	3561
	1	0.77	0.62	0.68	614
	tačnost			0.92	4175
	makro prosek	0.85	0.79	0.82	4175
	otežan prosek	0.91	0.92	0.91	4175
Dva skrivena sloja nebalansirana sa class_weight	0	0.94	0.97	0.96	3561
	1	**0.81**	**0.62**	**0.70**	**614**
	tačnost			0.92	4175
	makro prosek	0.87	0.80	0.83	4175
	otežan prosek	0.92	0.92	0.92	4175

TABELA 4.4: Klasifikacioni izveštaji NM sa skupovima podataka koji nisu eksplicitno balansirani.

Model	Info	Prec.	Odziv	F1 mera	Broj instanci
Jedan skriveni sloj balansirana podsempliranje	0	0.96	0.86	0.91	3561
	1	0.50	0.79	0.61	614
	tačnost			0.85	4175
	makro prosek	0.73	0.83	0.76	4175
	otežan prosek	0.89	0.85	0.86	4175
Dva skrivena sloja balansirana podsempliranje	0	0.96	0.87	0.91	3561
	1	0.51	0.80	0.62	614
	tačnost			0.86	4175
	makro prosek	0.73	0.83	0.77	4175
	otežan prosek	0.89	0.86	0.87	4175

TABELA 4.5: Klasifikacioni izveštaji NM sa skupom podataka koji je balansiran podsempliranjem.

Model	Info	Prec.	Odziv	F1 mera	Broj instanci
Jedan skriveni sloj balansirana SMOTE	0	0.96	0.89	0.92	3561
	1	0.55	0.76	0.64	614
	tačnost			0.87	4175
	makro prosek	0.76	0.83	0.78	4175
	otežan prosek	0.90	0.87	0.88	4175
Dva skrivena sloja balansirana SMOTE	0	0.96	0.90	0.93	3561
	1	0.57	0.79	0.66	614
	tačnost			0.88	4175
	makro prosek	0.77	0.84	0.80	4175
	otežan prosek	0.90	0.88	0.89	4175
Jedan skriveni sloj balansirana ADASYN	0	0.96	0.87	0.91	3561
	1	0.51	0.76	0.61	614
	tačnost			0.86	4175
	makro prosek	0.73	0.82	0.76	4175
	otežan prosek	0.89	0.86	0.87	4175
Dva skrivena sloja balansirana ADASYN	0	0.96	0.90	0.93	3561
	1	0.56	0.77	0.65	614
	tačnost			0.88	4175
	makro prosek	0.76	0.83	0.79	4175
	otežan prosek	0.90	0.88	0.88	4175

TABELA 4.6: Klasifikacioni izveštaji NM sa skupovima podataka koji su balansirani nadsempliranjem.

4.3 Poređenje sa rezultatima ranijih radova

Kao što je navedeno pri kraju poglavlja 2.6, direktno poređenje sa postojećim alatima nije moguće. Glavni razlog tome predstavlja odsustvo odgovarajućeg *benchmark* skupa podataka zbog čega su u svim postojećim radovima korišćeni različiti skupovi.

Glava 5

Diskusija i budući rad

Rezultati prezentovani u glavi 4 jasno pokazuju bolje performanse MPV naspram NM. Ovo se u današnje vreme može smatrati izuzetkom, jer NM često prevazilaze „tradicionalnije" modele u rešavanju mnogih problema. Ipak, ovo nije pravilo i postoje slučajevi kada se dešava baš obratno. Razlozi tome mogu biti dosta različiti, ali se često, bez greške, mogu izdvojiti dva glavna razloga - mala količina podataka i rad sa obrađenim podacima. NM se pokazuju posebno uspešnim pri velikim količinama podataka i onda kada su ti podaci dati u svom sirovom obliku (primer ovog bi bilo korišćenje samo proteinske sekvence kao ulaza modela, bez računanja ikakvih fizičko-hemijskih osobina) [51]. Kako ništa od prethodnog ne važi u slučaju ovog rada, veća uspešnost MPV nije iznenađenje.

Moguća poboljšanja ovog istraživanja uključuju sledeće:

- Povećati izvorni skup podataka;
- Uvesti dodatne, informativnije atribute;
- Koristiti sirove podatke.

Povećanje izvornog skupa podataka je moguće jedino u slučaju korišćenja dodatnog skupa proteina nekog organizma srodnog organizmu *Arabidopsis Thaliana*, ili eventualno korišćenjem skupa proteina viših taksonomskih kategorija kojem pripada *Arabidopsis Thaliana*.

Pristup kojim bi se povećao izvorni skup podataka a istovremeno sačuvala specifičnost problema, bio bi učenje više specifičnosti odjednom. Ovim pristupom bi se mogli uključiti skupovi proteina velikog broja srodnih organizama, pri čemu bi se sačuvale specifičnosti svakog od njih. Ideja ove tehnike je kreiranje po jednog modela koji rešava problem N-glikozilovanosti proteina, za svaki organizam ponaosob, pri čemu se ti modeli zajednički obučavaju. Na ovaj način, modeli koji odgovaraju organizmima sa malim skupovima podataka mogu da se oslone na modele sa većim skupovima podataka. Primetiti da primena ove tehnike na problem N-glikozilovanosti proteina ima smisla, jer iako postoje specifičnosti N-glikozilovanosti među organizmima, ovaj proces je dosta sličan kod svih njih. Tehnika učenja više poslova odjednom detaljnije je opisana radom [51].

Uvođenje dodatnih atributa je daleko jednostavniji proces koji može znatno poboljšati rezultate. U ovom radu su se za potrebe rešavanja problema koristile odabrane fizičko-hemijske osobine proteina. Izbor je kreiran na osnovu dostupnosti alata koji implementiraju funkcije izračunavanja ovih osobina nad nekim proteinom, kao i vremenskog trajanja izvršavanja tih alata. Postoji puno alata specijalno razvijenih za određivanje različitih osobina proteina (primeri takvih alata su kreirani u radovima [9] i [47]), pri čemu su problemi korišćenja tih alata njihova dostupnost i trajanje izvršavanja. Mnoge alate je moguće koristiti samo pod specijalnim uslovima, a ako se sa druge strane oni i mogu lako koristiti, izvršavanje tih alata nad

korišćenim skupom proteina bi trajalo nedeljama. Zbog toga su u ovom radu korišćene one osobine koje su se mogle sračunati pomoću *Biopython* biblioteke ili bez korišćenja ikakvih dodatnih alata.

Uzevši u obzir prethodno, verovatno su radom obuhvaćene sve osobine koje zadovoljavaju gore navedene uslove. Tokom izrade rada su se razmatrale i osobine proteina koje nisu fizičko-hemijske; postojala je ideja o korišćenju čitave proteinske sekvence u obliku atributa. Kako proces N-vezane glikozilacije dosta zavisi od redosleda AK, mišljenje je da bi uvođenje ovog atributa značajno poboljšalo rezultate klasifikatora. Glavna prepreka implementacije ove ideje je varijabilna dužina proteina. Modelima koji se koriste u radu (MPV i NM) se moraju obezbediti podaci fiksnih dužina.

Jedan od načina prevazilaženja ovog problema je enkodiranje proteinske sekvence varijabilne dužine u neku numeričku sekvencu fiksne dužine. Ovo bi bilo moguće implementirati korišćenjem enkoderske neuronske mreže (npr. rekurentne neuronske mreže) u sklopu postojećih modela, za dobijanje vektorskih reprezentacija proteina, fiksne dužine. Druga mogućnost bi bila korišćenje gotovog modela, kreiranog u radu [73], za dobijanje ovakvih reprezentacija.

Korišćenje sirovih podataka odnosi se na direktnu upotrebu sekvenca AK kao ulaznih podataka klasifikatora. Klasifikatori koji se koriste u ovakvim situacijama jesu rekurentne ili konvolutivne neuronske mreže. Ove neuronske mreže, između ostalog, imaju mogućnost obrade podataka varijabilnih dužina, datim u njihovom sirovom obliku. Dok su se prethodno pomenuta poboljšanja odnosila na rešavanju problema pomoću skupa podataka kreiranog računanjem fizičko-hemijskih osobina proteina, kod ovog poboljšanja skup podataka predstavlja proteine bez njihove prethodne obrade.

Glava 6

Zaključak

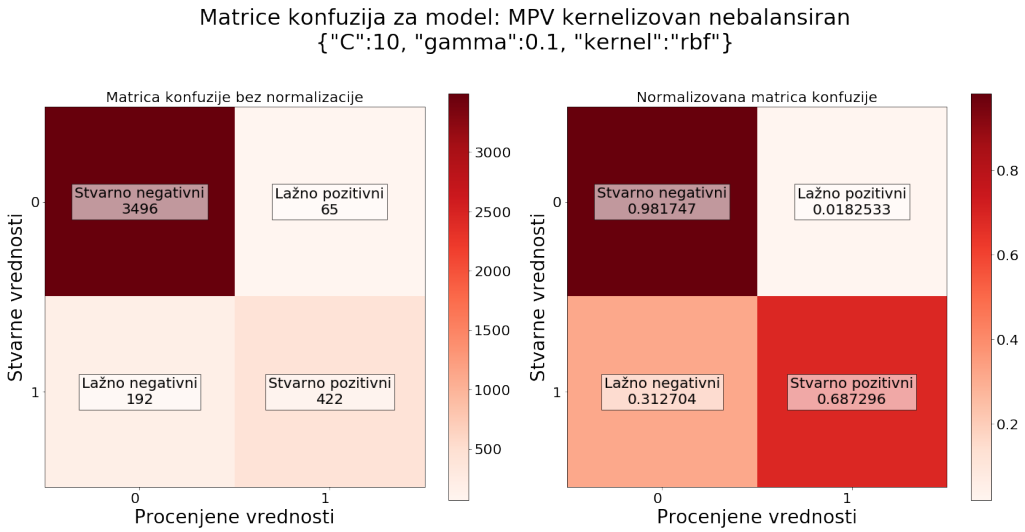
Cilj ovog istraživanja predstavlja razvoj klasifikatora zasnovanih na različitim metodama MU za utvrđivanje N-glikozilovanosti proteina organizma *Arabidopsis Thaliana*, uz korišćenje savremenih tehnika balansiranja podataka. Pored prethodnog, namera autora je bila i korišćenje javno dostupnih alata zarad implementacije raznih ideja, kao i korišćenje jedne od najpoznatijih proteinskih baza podataka. Ovim je omogućena jednostavna ponovljivost svih prezentovanih rezultata, kao i nezavisnost od alata zatvorenog koda.

Izloženi rezultati ukazuju da je za utvrđivanje N-glikozilovanosti proteina organizma *Arabidopsis Thaliana* pogodno koristiti MPV u kombinaciji sa nekom od tehnika nadsempliranja manjinske klase skupa podataka. Odlični rezultati se postižu i bez tehnika nadsempliranja, pri čemu je onda potrebno koristiti informaciju balansiranosti klasa prilikom procesa učenja (tj. koristiti *class_weight* metaparametar modela).

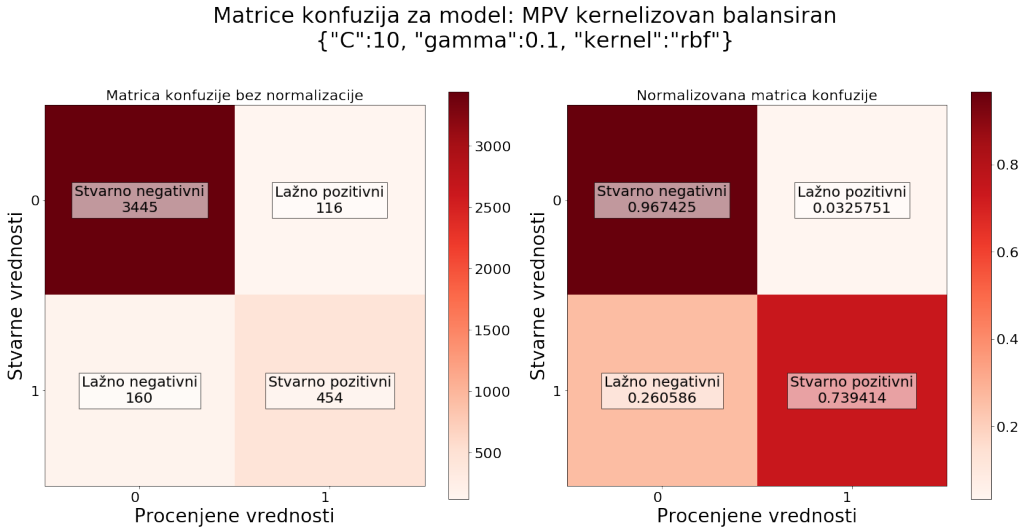
Moguća poboljšanja dobijenih rezultata su izložena u prethodnoj glavi, zajedno sa eventualno boljim idejama za rešavanje problema N-glikozilovanosti.

Dodatak A

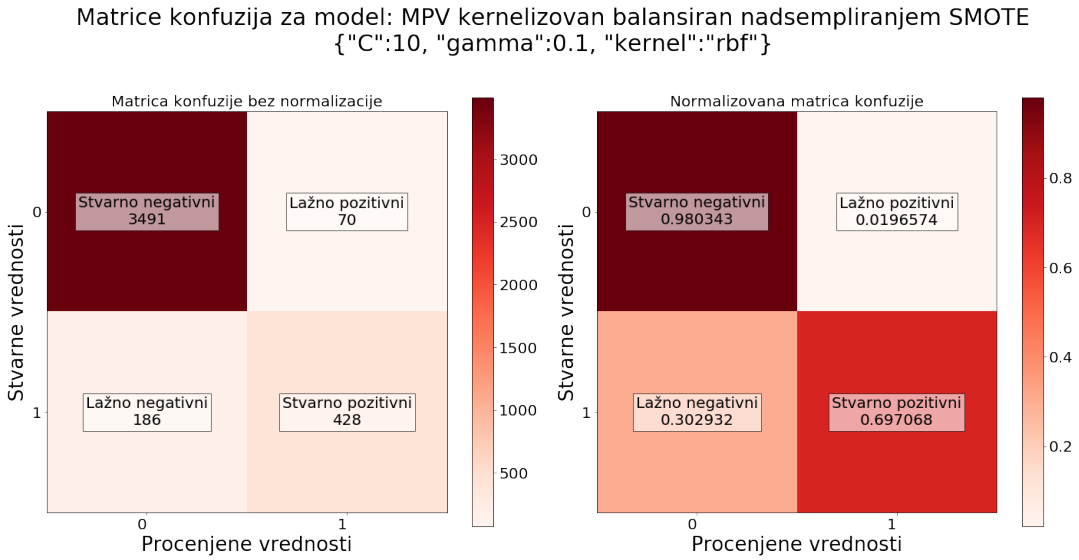
Prilog uz rezultate



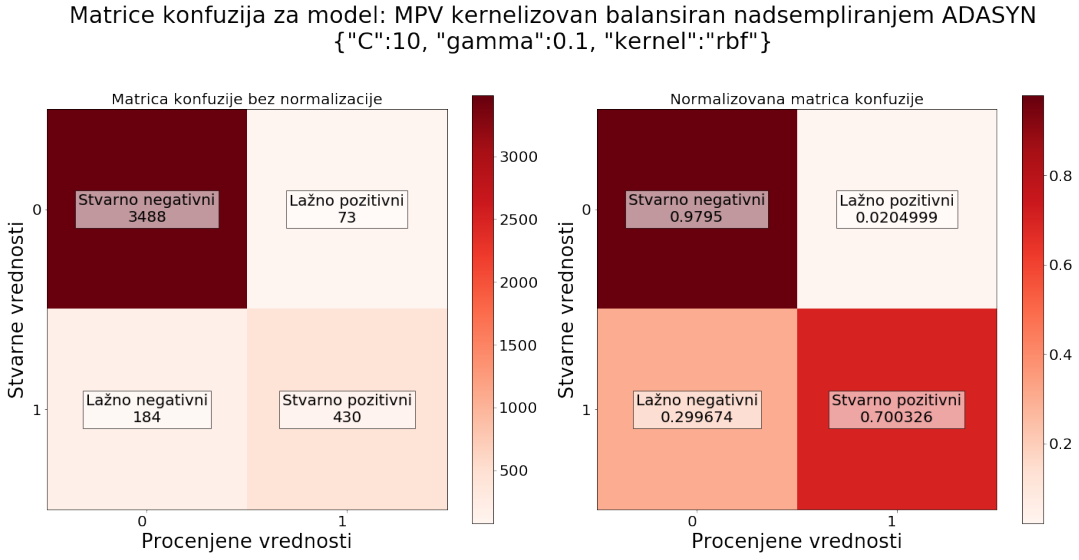
SLIKA A.1: Matrica konfuzije kernelizovanog MPV, nad nebalansiranim skupom podataka i bez uključenog *class_weight* parametra.



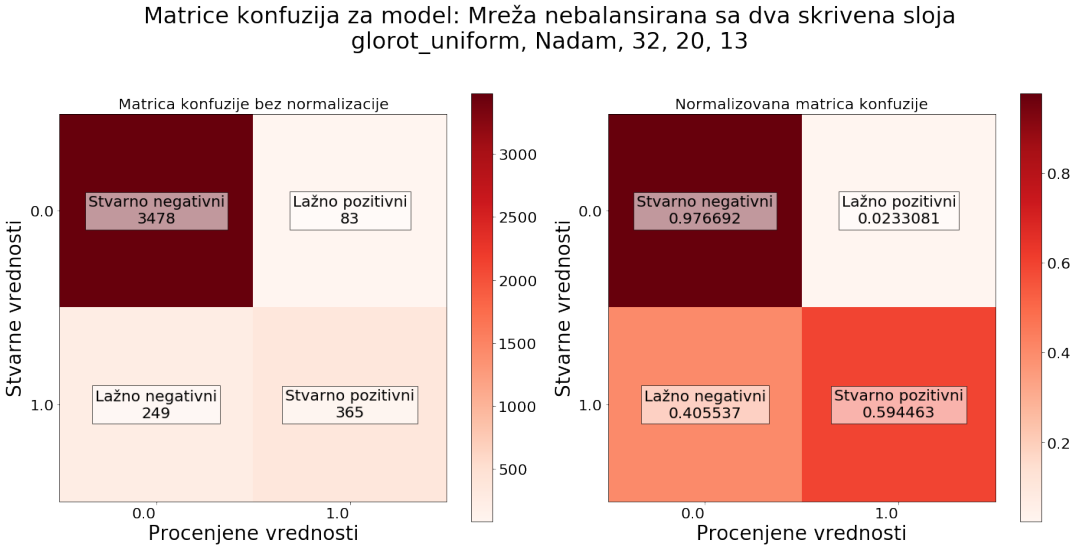
SLIKA A.2: Matrica konfuzije kernelizovanog MPV, nad nebalansiranim skupom podataka i sa uključenim *class_weight* parametrom.



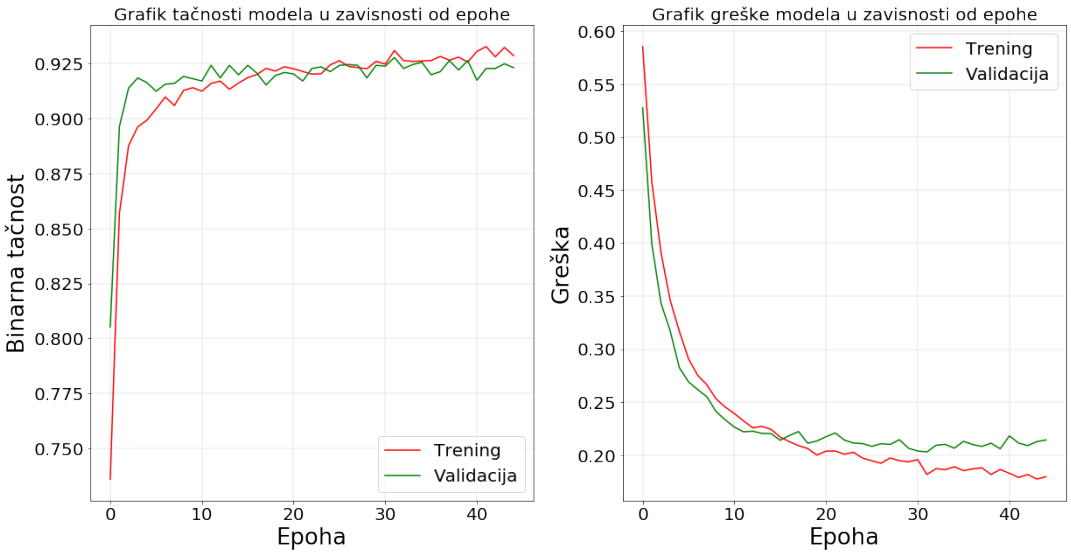
SLIKA A.3: Matrica konfuzije kernelizovanog MPV, nad SMOTE nadsempliranim skupom podataka.



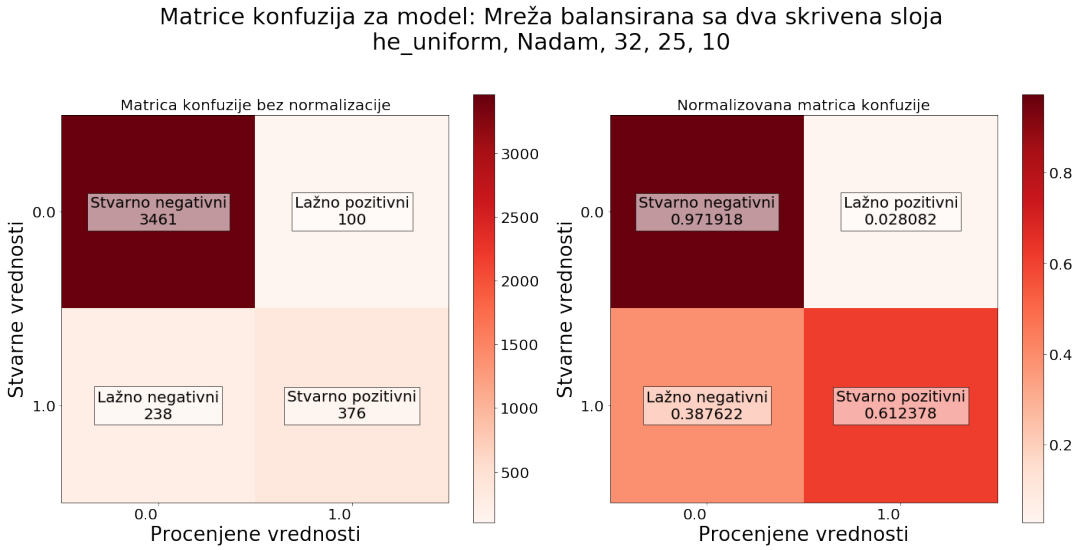
SLIKA A.4: Matrica konfuzije kernelizovanog MPV, nad ADASYN nadsempliranim skupom podataka.



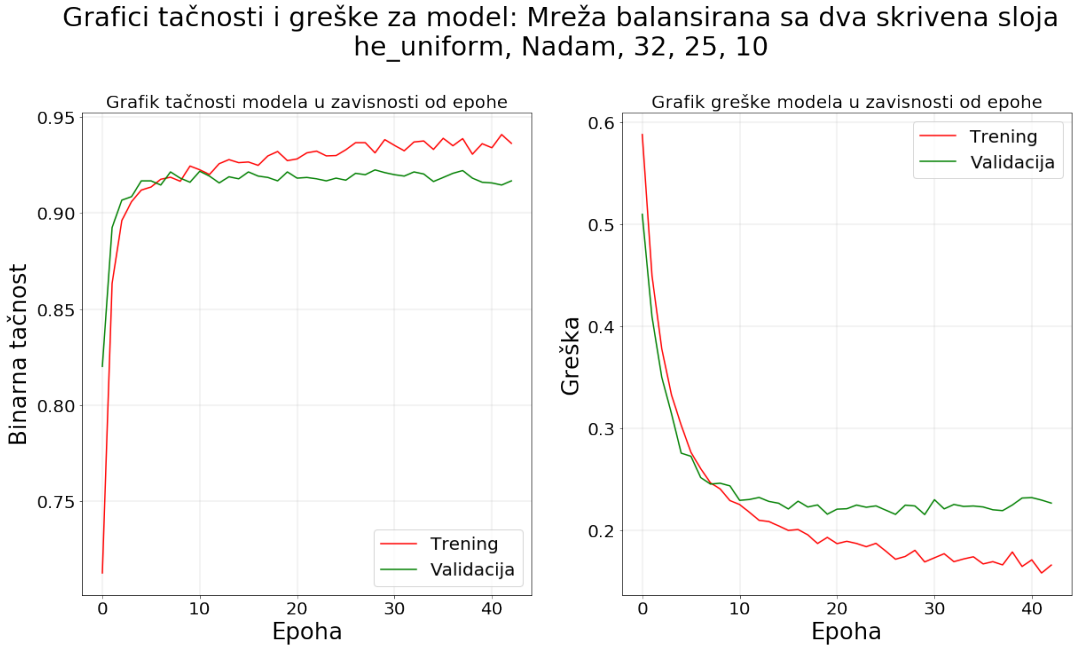
SLIKA A.5: Matrica konfuzije NM sa dva skrivena sloja, nad nebalansiranim skupom podataka i bez uključenog *class_weight* parametra.



SLIKA A.6: Grafici binarne tačnosti i greškle NM sa dva skrivena sloja, nad nebalansiranim skupom podataka i bez uključenog *class_weight* parametra.



SLIKA A.7: Matrica konfuzije NM sa dva skrivena sloja, nad nebalansiranim skupom podataka i sa uključenim *class_weight* parametrom.



SLIKA A.8: Grafici binarne tačnosti i greškle NM sa dva skrivena sloja, nad nebalansiranim skupom podataka i sa uključenim *class_weight* parametrom.

Literatura

1. Alberts, B. *i dr. Molecular Biology of the Cell. 4th edition* <https://www.ncbi.nlm.nih.gov/books/NBK26911/> (New York: Garland Science, 2002).
2. Ambrogelly, A., Söll, D. & Palioura, S. Natural expansion of the genetic code. *Nature Chemical Biology*. <https://doi.org/10.1038/nchembio847> (2007).
3. Andersen, N. H. Protein Structure, Stability, and Folding. Methods in Molecular Biology. Volume 168 Edited by Kenneth P. Murphy (University of Iowa College of Medicine). Humana Press: Totowa, New Jersey. 2001. ix + 252 pp. $89.50. ISBN 0-89603-682-0. *Journal of the American Chemical Society* **123,** 12933–12934 (2001).
4. Biopython. *ProtParam module* [Accessed 15-November-2019]. 2019. <ht%5Ctps://biopython.org/wiki/ProtParam>.
5. Boeckmann, B. *i dr.* The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic acids research* **31.** PMC165542[pmcid], 365–370. ISSN: 1362-4962 (2003).
6. Bowyer, K. W., Chawla, N. V., Hall, L. O. & Kegelmeyer, W. P. SMOTE: Synthetic Minority Over-sampling Technique. *CoRR* **abs/1106.1813.** arXiv: 1106.1813. <http://arxiv.org/abs/1106.1813> (2011).
7. Buxbaum, E. *Fundamentals of Protein Structure and Function* 1–367. doi:10.1007/978-0-387-68480-2 (2007).
8. Caragea, C., Sinapov, J., Silvescu, A., Dobbs, D. & Honavar, V. Glycosylation site prediction using ensembles of Support Vector Machine classifiers. *BMC bioinformatics* **8.** 1471-2105-8-438[PII], 438–438. ISSN: 1471-2105 (2007).
9. Cheng, J., Randall, A. Z., Sweredoski, M. J. & Baldi, P. SCRATCH: a protein structure and structural feature prediction server. *Nucleic Acids Research* **33,** W72–W76. ISSN: 0305-1048 (2005).
10. Chuang, G.-Y. *i dr.* Computational prediction of N-linked glycosylation incorporating structural properties and patterns. *Bioinformatics (Oxford, England)* **28.** bts426[PII], 2249–2255 (2012).
11. Cock, P. J. A. *i dr.* Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25,** 1422–1423. ISSN: 1367-4803 (2009).
12. Commons, W. *File:Activation logistic.svg — Wikimedia Commons, the free media repository* [Online; accessed 27-October-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Activation_logistic.svg&oldid=185144157>.
13. Commons, W. *File:Activation rectified linear.svg — Wikimedia Commons, the free media repository* [Online; accessed 27-October-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Activation_rectified_linear.svg&oldid=185144126>.

14. Commons, W. *File:Activation tanh.svg — Wikimedia Commons, the free media repository* [Online; accessed 27-October-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Activation_tanh.svg&oldid=185144140>.

15. Commons, W. *File:AminoAcidball.svg — Wikimedia Commons, the free media repository* [Online; accessed 20-September-2019]. 2017. <https://commons.wikimedia.org/w/index.php?title=File:AminoAcidball.svg&oldid=256827719>.

16. Commons, W. *File:ArtificialNeuronModel english.png — Wikimedia Commons, the free media repository* [Online; accessed 27-October-2019]. 2017. <https://commons.wikimedia.org/w/index.php?title=File:ArtificialNeu%5C-ronModel_english.png&oldid=233886112>.

17. Commons, W. *File:Central Dogma Model.png — Wikimedia Commons, the free media repository* [Online; accessed 20-September-2019]. 2019. <https://commons.wikimedia.org/w/index.php?title=File:Central_Dogma_Model.png&oldid=344651833>.

18. Commons, W. *File:Kernel Machine.png — Wikimedia Commons, the free media repository* [Online; accessed 26-October-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Kernel_Machine.png&oldid=215993536>.

19. Commons, W. *File:Multi-Layer Neural Network-Vector.svg — Wikimedia Commons, the free media repository* [Online; accessed 27-October-2019]. 2016. <https://commo%5C-ns.wikimedia.org/w/index.php?title=File:Multi-Layer_Neural_Network-Vector.svg&oldid=220121168>.

20. Commons, W. *File:Peptidformationball.svg — Wikimedia Commons, the free media repository* [Online; accessed 20-September-2019]. 2017. <https://commons.wikimedia.org/w/index.php?title=File:Peptidformationball.svg&oldid=251535610>.

21. Commons, W. *File:Protein structure (full).png — Wikimedia Commons, the free media repository* [Online; accessed 20-September-2019]. 2018. <https://commons.wikimedia.org/w/index.php?title=File:Protein_structure_(full).png&oldid=325901327>.

22. Commons, W. *File:SVM margin.png — Wikimedia Commons, the free media repository* [Online; accessed 26-October-2019]. 2019. <https://commons.wikimedia.org/w/index.php?title=File:SVM_margin.png&oldid=356523875>.

23. Commons, W. *File:Svm separating hyperplanes (SVG).svg — Wikimedia Commons, the free media repository* [Online; accessed 26-October-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Svm_separating_hyperplanes_(SVG).svg&oldid=217578095>.

24. Commons, W. *File:Types of glycans.svg — Wikimedia Commons, the free media repository* [Online; accessed 21-September-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Types_of_glycans.svg&oldid=222378856>.

25. Commons, W. *File:Variety of glycans.svg — Wikimedia Commons, the free media repository* [Online; accessed 21-September-2019]. 2016. <https://commons.wikimedia.org/w/index.php?title=File:Variety_of_glycans.svg&oldid=221872873>.

26. Consortium, T. U. UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Research* **47,** D506–D515. ISSN: 0305-1048 (2018).

27. Consortium, U. *UniProt:Extracted positive set* [Online; accessed 5-November-2019]. 2019. <https://www.uniprot.org/uniprot/?query=annotation:(type:carbohyd%20%22n%20linked%22)&fil=reviewed%5C%3Ayes+AND+organism%5C%3A%22Arabidopsis+thaliana+(Mouse-ear+cress)+[3702]%22%5C&sort=score>.
28. Consortium, U. *UniProt:Overview* [Online; accessed 24-September-2019]. 2019. <https://www.uniprot.org/help/about>.
29. Draw.io. *draw.io - Online Diagramming* [Accessed 30-November-2019]. 2019. <https://www.draw.io>.
30. Ganganwar, V. An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering* **2,** 42–47 (2012).
31. Group, S. B. R. *Protein_ML* https://github.com/SBRG/Protein_ML/blob/master/create_feature_matrix.ipynb. 2016.
32. Gupta, R., Jung, E. & Brunak, S. Prediction of N-glycosylation sites in human proteins. **46,** 203–206 (2004).
33. Guruprasad, K., Reddy, B. & Pandit, M. W. Correlation between stability of a protein and its dipeptide composition: a novel approach for predicting in vivo stability of a protein from its primary sequence. *Protein Engineering, Design and Selection* **4,** 155–161. ISSN: 1741-0126 (1990).
34. Hamby, S. E. & Hirst, J. D. Prediction of glycosylation sites using random forests. *BMC bioinformatics* **9.** 1471-2105-9-500[PII], 500–500. ISSN: 1471-2105 (2008).
35. He, H., Bai, Y., Garcia, E. A. & Li, S. *ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning* (2008), 1322–1328. doi:10.1109/IJCNN.2008.4633969.
36. Henquet, M. N-glycosylation in plants: science and application (2009).
37. Hill, T. & Lewicki, P. (2006). ISBN: 1884233597.
38. Hunter, J. D. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering* **9,** 90–95 (2007).
39. Imperiali, B. & O'Connor, S. E. Effect of N-linked glycosylation on glycopeptide and glycoprotein structure. *Current Opinion in Chemical Biology* **3,** 643–649. ISSN: 1367-5931 (1999).
40. Kabsch, W. & Sander, C. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* **22,** 2577–2637 (1983).
41. Kyte, J. & Doolittle, R. F. A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology* **157,** 105–132. ISSN: 0022-2836 (1982).
42. Li, F. *i dr.* GlycoMine: a machine learning-based approach for predicting N-, C- and O-linked glycosylation in the human proteome. *Bioinformatics* **31,** 1411–1419 (2015).
43. Li, F. *i dr.* Positive-unlabelled learning of glycosylation sites in the human proteome. *BMC bioinformatics* **20.** PMC6404354[pmcid], 112–112. ISSN: 1471-2105 (2019).
44. li, X. & Liu, B. *Learning from Positive and Unlabeled Examples with Different Data Distributions* (1970), 218–229. doi:10.1007/11564096_24.

45. Lobry, J. & Gautier, C. Hydrophobicity, expressivity and aromaticity are the major trends of amino-acid usage in 999 Escherichia coli chromosome-encoded genes. *Nucleic Acids Research* **22,** 3174–3180. ISSN: 0305-1048 (1994).

46. Lutz, M. *Python pocket reference* 5ed. ISBN: 9781449357016 (O'Reilly, 2014).

47. Magnan, C. N. & Baldi, P. SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics* **30.** ISSN: 1367-4803. doi:10.1093/bioinformatics/btu352. eprint: http://oup.prod.sis.lan/bioinformatics/article-pdf/30/18/2592/7509988/btu352.pdf. <https://doi.org/10.1093/bioinformatics/btu352> (2014).

48. Mann, M. & Jensen, O. Mann, M. and Jensen, O.N. Proteomic analysis of post-translational modifications. Nat. Biotechnol. 21, 255-261. *Nature biotechnology* **21,** 255–61 (2003).

49. McKinney, W. *Data Structures for Statistical Computing in Python Proceedings of the 9th Python in Science Conference* (ur. van der Walt, S. & Millman, J.) (2010), 51–56.

50. McKinney, W. pandas: a Foundational Python Library for Data Analysis and Statistics.

51. Nikolić, M. & Zečević, A. *Mašinsko učenje, skripta* <http://ml.matf.bg.ac.rs/readings/ml.pdf> (2019).

52. NumPy.org. *NumPy: About Us* [Online; accessed 26 - September - 2019]. 2019. <https://numpy.org>.

53. Park, D. S., Poretz, R. D., Stein, S., Nora, R. & Manowitz, P. Association of Alcoholism with the N-Glycosylation Polymorphism of Pseudodeficient Human Arylsulfatase A. *Alcoholism: Clinical and Experimental Research* **20,** 228–233 (1996).

54. Patterson, M. C. Metabolic Mimics: The Disorders of N-Linked Glycosylation. *Seminars in Pediatric Neurology* **12.** Metabolic Mimics–Disorders of N-linked Glycosylation, 144–151. ISSN: 1071-9091 (2005).

55. Pedregosa, F. *i dr.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12,** 2825–2830 (2011).

56. Pérez, F. & Granger, B. E. IPython: a System for Interactive Scientific Computing. *Computing in Science and Engineering* **9,** 21–29. ISSN: 1521-9615 (2007).

57. Pinho, S. S. & Reis, C. A. Glycosylation in cancer: mechanisms and clinical implications. *Nature Reviews Cancer* **15.** Review Article, 540 (2015).

58. Project Jupyter. *Project Jupyter: About Us* [Online; accessed 26-September-2019]. 2019. <https://jupyter.org/about>.

59. Proteintech. *POST TRANSLATIONAL MODIFICATIONS: AN OVERVIEW* [Online; accessed 02-October-2019]. 2017. <https://www.ptglab.com/news/blog/post-translational-modifications-an-overview/#References>.

60. Rogers, K. *Biomolecule* Online; accessed September 14, 2019. 2019. <https://www.britannica.com/science/biomolecule>.

61. Schedin-Weiss, S., Winblad, B. & Tjernberg, L. O. The role of protein glycosylation in Alzheimer disease. *The FEBS Journal* **281,** 46–62 (2014).

62. Shin, S., Hoang, T.-T., Le, T.-H. & Lee, M.-Y. A New Robust Design Method Using Neural Network. *Journal of Nanoelectronics and Optoelectronics* **11,** 68–78 (2016).

63. Taylor, M. E. & Drickamer, K. Structural insights into what glycan arrays tell us about how glycan-binding proteins interact with their ligands. *Glycobiology* **19,** 1155–1162. ISSN: 1460-2423 (2009).

64. van der Walt, S., Colbert, S. C. & Varoquaux, G. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science Engineering* **13,** 22–30 (2011).

65. Varki, A. *i dr. Essentials of Glycobiology. 2nd edition.* <https://www.ncbi.nlm.nih.gov/books/NBK1908/> (Cold Spring Harbor (NY): Cold Spring Harbor Laboratory Press, 2009).

66. Voet, D. & Voet, J. G. *Biochemistry 4th edition* (2005).

67. Walsh, C. T., Garneau-Tsodikova, S. & Gatto Jr., G. J. Protein Posttranslational Modifications: The Chemistry of Proteome Diversifications. *Angewandte Chemie International Edition* **44,** 7342–7372 (2005).

68. *What is a cell?* Online; accessed September 16, 2019. 2019. <https://ghr.nlm.nih.gov/primer/howgeneswork/protein>.

69. Wilkins, M. Proteomics data mining. *Expert Review of Proteomics* **6.** doi:10.1586/epr.09.81. eprint: https://doi.org/10.1586/epr.09.81. <https://doi.org/10.1586/epr.09.81> (2009).

70. Winterpacht, A. *i dr.* A novel mutation in FGFR-3 disrupts a putative N-glycosylation site and results in hypochondroplasia. *Physiological Genomics* **2.** PMID: 11015576, 9–12 (2000).

71. Wu, C. H. *i dr.* The Protein Information Resource: an integrated public resource of functional annotation of proteins. *Nucleic acids research* **30.** PMC99125[pmcid], 35–37. ISSN: 1362-4962 (2002).

72. Wujek, P., Kida, E., Walus, M., Wisniewski, K. E. & Golabek, A. A. N-Glycosylation Is Crucial for Folding, Trafficking, and Stability of Human Tripeptidyl-peptidase I. *Journal of Biological Chemistry* **279,** 12827–12839 (2004).

73. Zhang, S., Jiang, H., Xu, M., Hou, J. & Dai, L. *A Fixed-Size Encoding Method for Variable-Length Sequences with its Application to Neural Network Language Models* 2015. arXiv: 1505.01504 [cs.NE].