UNIVERZITET U BEOGRADU
MATEMATIČKI FAKULTET

Aleksandra S. Jovičić

RAZVOJ ALATA ZA DETEKCIJU ŠTETNIH AMINOKISELINSKIH SUPSTITUCIJA U PROTEINIMA UKLJUČENIM U NASTANAK KANCERA

master rad

Beograd, 2022.

Mentor:

dr Jovana Kovačević, docent
Univerzitet u Beogradu, Matematički fakultet

Članovi komisije:

dr Branislava Gemović, naučni saradnik
Institut za nuklearne nauke "Vinča"

dr Mladen Nikolić, vanredni profesor
Univerzitet u Beogradu, Matematički fakultet

Datum odbrane: _____

Zahvaljujem se mentorki profesorki Jovani Kovačević i profesoru Mladenu Nikoliću na saradnji, korisnim primedbama i sugestijama i posvećenom vremenu.
Zahvaljujem se dr Branislavi Gemović na predloženoj temi i na smernicama u vezi sa biološkim aspektima rada.
Zahvaljujem se porodici i prijateljima na ukazanoj podršci i strpljenju.

Posvećeno porodici i prijateljima

Naslov master rada: Razvoj alata za detekciju štetnih aminokiselinskih supstitucija u proteinima uključenim u nastanak kancera

Rezime: Veliki problem današnjice predstavljaju maligne bolesti. Njihova dijagnostika i lečenje mogu biti veoma zahtevni i teški po ljudski organizam. Različiti su uzroci nastanka ovih bolesti. U ovom radu fokus je na kancerogene bolesti nastale zbog prisustva nenaslednih aminokiselinskih supstitucija u sekvenci proteina. Zadatak razvijanog alata je da ispita da li je mutacija unutar date genomske sekvence štetna ili ne. Korišćeni su različiti modeli mašinskog učenja koji na osnovu dostupnih podataka predviđaju rezultat za nove mutacije. Podaci na osnovu kojih modeli uče preuzeti su iz javno dostupnih baza podataka. Rad sadrži pregled pripreme skupa podataka, korišćenih tehnika za razvoj nekoliko modela, njihovo testiranje i tumačenje rezultata.

Ključne reči: bioinformatika, geni, mutacije, proteinske sekvence

Sadržaj

Glava 1

Upoznavanje sa problemom

Problem koji se rešava u ovom radu biće predstavljen preko prikaza misaonog procesa, motivacije, kao i preko upoznavanja sa planom celog rada po koracima njegove izrade.

1.1 Motivacija i opis problema

Razvoj nauke je uvek motivisan ljudskim potrebama od kojih je najveća zdravlje. Rak, kancer ili zloćudni tumor, najsmrtonosnija je bolest današnjice. Godine 2020. oko 10 miliona ljudi je podleglo ovoj bolesti [1]. Karakteristika ove bolesti je nekontrolisano množenje abnormalnih ćelija bez ograničenja. Iz tog razloga, one se mogu proširiti na bilo koji organ i deo tela. Ovaj proces, zvan metastaziranje, uglavnom jeste razlog smrti. Nekontrolisano umnožavanje ćelija se dešava zbog određenih izmena genetičkog materijala unutar ćelije. Promene nad genetičkim materijalom, zvane mutacije, mogu biti nasleđene ili mogu nastati posle rođenja. Veći procenat oboljenja od raka su somatska, odnosno nastaju nakon rođenja, dok je od 5% do 10% nasleđeno [1, 2].

Vreme otkrivanja bolesti je bitno kod ovog oboljenja. Rano detektovanje zloćudnog tumora smanjuje procenat smrtnosti ove bolesti, što se postiže kroz ranu dijagnostiku i skrining. Rana dijagnostika uključuje rano primećivanje specifičnih simptoma od strane lekara koji se manifestuju na pacijentu i mogu voditi oboljenju, dok skrining označava testiranje ili izvršavanje nekog vida pregleda, bez prisutnih simptoma, u cilju otkrivanja neke bolesti [1].

Mutacije u nukleotidnim sekvencama uzrokuju poremećaj u sintezi proteina i samim tim mogu biti razlog bilo koje funcionalne ili nefunkcionlalne promene u

organizmu. Shodno tome, postoji velika motivacija za izradu bioinformatičkih alata koji bi na brz način, zahvaljujući sekvencama proteina sa supstitucijama, uspeo da otkrije da li je data mutacija neutralna ili kancerogena.

1.2 Pregled rada

Rad čine sledeća poglavlja:

- Prvo poglavlje upoznaje čitaoce sa osnovnim pojmovima koji se koriste u radu zarad njegovog boljeg razumevanja. U tom delu objašnjeni su biološki i bioinformatički pojmovi, kao i pojmovi iz mašinskog učenja.
- Drugim poglavljem je dat opis načina na koji su podaci prikupljani i pretprocesirani. Tokom ovog procesa korišćeno je više različitih baza za izdvajanje podataka, naveden je način filtriranja podataka sa određenim karakteristikama i njihovo objedinjavanje.
- Trećim poglavljem se vrši upoznavanje sa tehnikama i bibliotekama koje se koriste u radu.
- Četvrto poglavlje služi za prikaz razvoja modela. Predstavljeno je nalaženje parametara modela, njihova implementacija, kao i dobijeni rezultati.
- Petim poglavljem su prikazani rezultati *state-of-art* alata za rešavanje ovde posmatranog problema, usled čega je omogućeno njegovo poređenje sa razvijenim modelima.
- Šesto, poslednje poglavlje je zaduženo za navođenje ključnih elemenata rada i njihovo objedinjavanja, sa osvrtom na cilj i zaključak celog rada.

Glava 2

Osnovni pojmovi

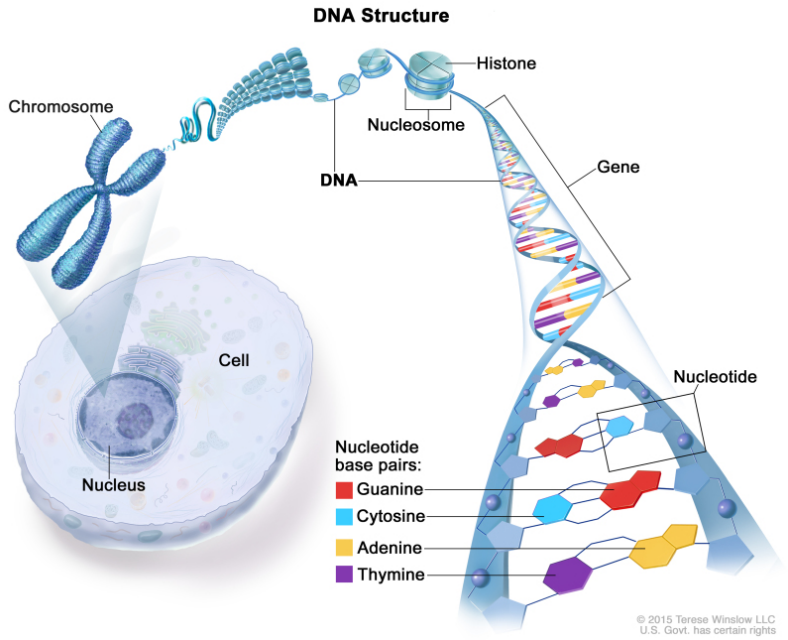
U ovom poglavlju biće opisani osnovni pojmovi koji se koriste u radu. Neki od pojmova se odnose na biološke procese i pojave, dok su drugi pojmovi iz oblasti mašinskog učenja.

2.1 Biološki pojmovi

Bioinformatika je nauka koja ima široku primenu u molekularnoj genetici i genomici. Različitim biološkim procesima dolazimo do podataka koji se prikupljaju, čuvaju, obrađuju i analiziraju. Iz tog razloga su u nastavku objašnjeni neki od značajnih bioloških procesa i pojmova koji su važni za razumevanje ovog rada.

2.1.1 DNK

Osnovna forma genetičkog materijala je dezoksiribonukleinska kiselina (u nastavku DNK) koja se sastoji od dva polinukleotidna lanca sastavljena od četiri tipa nukleotida: A (adenin), G (guanin), T (timin) i C (citozin), kao što je prikazano na slici 2.1.1. Genomska replikacija je jedan od najbitnijih procesa u ćeliji. Pre nego što se ćelija podeli na dva dela potrebno je da izvrši kopiranje svog genetičkog materijala kako bi obe ćerke ćelije dobile po kopiju. Tokom replikacije gena lanci se odmotavaju i igraju ulogu obrasca za sintezu novog lanca. Sinteza se izvršava po pravilu komplemetarnosti baza, gde adenin odgovara timinu i guanin citozinu. Kao rezultat, proces replikacije počinje sa parom komplementarnih lanaca DNK i završava se sa dva para komplementarnih lanaca [3].



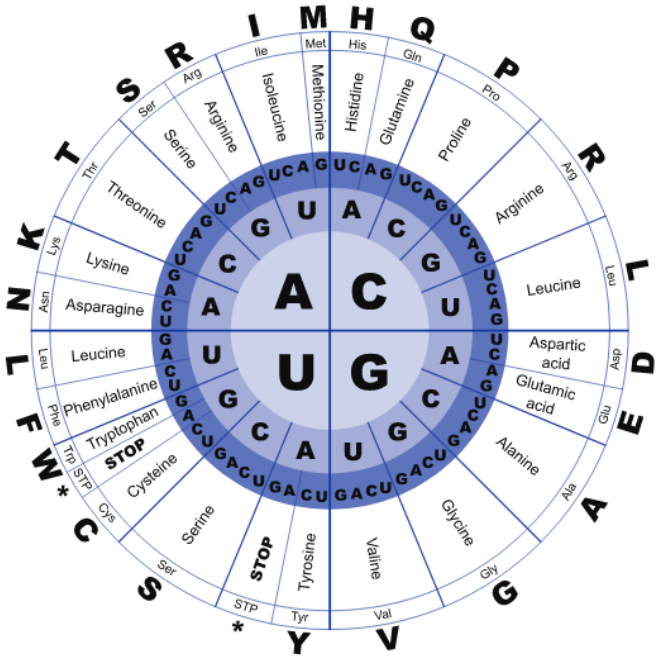
Slika 2.1: Struktura dvostrukog lanca DNK molekula [4]

2.1.2 Centralna dogma molekularne biologije

Centralna dogma molekulatne biologije, koju je prvi formulisao i objavio Francis Krik, navodi da "DNK formira RNK koja formira protein". Prema ovoj dogmi, gen iz genoma se prvo transkribuje u lanac ribonukleinske kiseline (u nastavku RNK) koji se sastoji od četiri ribonukleotida: A (adenin), G (guanin), U (uracil) i C (citozin). Zatim se transkript prevodi u aminokiselinsku sekvencu proteina.

U procesu transkripcije nastaje novi molekul RNK na osnovu postojeće DNK tako što se lanci DNK razdvoje i jedan od njih služi kao obrazac po kome seređaju komplementarni nukleotidi RNK. Naspram adenina DNK postavlja se uracil RNK, dok se naspram guanina DNK postavlja citozin RNK [3].

Posle transkripcije, procesom translacije se izvršava prevođenje iz RNK u aminokiselinski niz, odnosno protein. Tokom translacije, RNK lanac je podeljen na nepreklapajuće 3-grame zvane kodoni. Svaki kodon se prevodi u jednu od dvadeset aminokiselina koje su prisutne u ljudskom telu. Kao što je prikazano na slici 2.2, svaki od 4^3, odnosno 64 kodona kodira aminokiselinu (neki kodoni kodiraju istu aminokiselinu), sa izuzetkom od tri zaustavna kodona koji se ne prevode u aminokiseline nego služe kao oznaka za zaustavljanje prevođenja. Pomenuti sistem kodiranja se naziva genetski kod [3].



Slika 2.2: Genetski kod [5]

2.1.3 Proteini

Osnovne jedinice građe proteina su aminokiseline. Svaka aminokiselina sadrži amino (NH2) grupu i karboksilnu (COOH) grupu. Pri spajanju dve aminokiseline formira se peptidna veza (CO-NH) između amino grupe jedne i karboksilne grupe druge aminokiseline sa oslobađanjem molekula vode. Redosled aminokiselina sa njihovim peptidnim vezama predstavlja primarnu strukturu proteina [6].

Sekundarnu strukturu proteina definišu vodonične veze između atoma vodonika iz amino (NH) grupe jedne peptidne veze i atoma kiseonika iz karboksilne (CO) grupe druge, nesusedne peptidne veze. Vodonične veze koje nastaju na opisan način predstavljaju uzrok savijanja polipeptidnog lanca i mogu dovesti do nekoliko različitih oblika. Najčešći oblici su α-spirala, gde se vodonične veze obrazuju kod svake četvrte aminokiseline, i β-ploča, gde se vrši povezivanje udaljenih delova polipeptidnog lanca [6, 7].

Savijanjem lanca mogu se obrazovati oblici koji predstavljaju tercijarnu strukturu proteina. Protein se prema tercijarnoj strukturi može podeliti na fibrilarni, odnosno izduženi protein i globularni, odnosno loptasti protein. Kvaternarnu strukturu proteina čine dva ili više polipeptidna lanca sa svojom tercijarnom strukturom koji su povezani i međusobno interaguju. Većina proteina ima kvaternarnu struktu-

ru, ali je nemaju svi. Proteini mogu imati sličnu tercijarnu i kvaternarnu strukturu, što vodi sličnoj biološkoj funkciji, i pored različite primarne i sekundarne strukture. Iz tog razloga, mutacije koje menjaju primarnu strukturu ne moraju uticati na samu biološku funkciju proteina [6, 7].

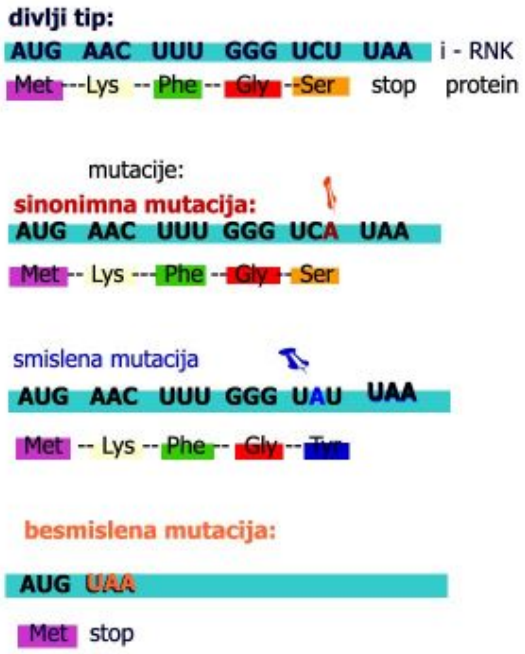
2.1.4 Mutacije

Mutacije predstavljaju promene u sekvenci DNK koje mogu biti rezultat grešaka koje nastaju tokom kopiranja DNK u procesima deobe ćelija, izlaganjem zračenju, izlaganjem hemikalijama zvanim mutageni ili kao posledica infekcije virusima. Germinativne mutacije javljaju se u polnim ćelijama i prenose se na potomstvo, dok se somatske mutacije javljaju u ćelijama tela i ne prenose se dalje. Genetske mutacije po delovanju na fenotip mogu biti veoma štetne, letalne, ali mogu biti i neutralne ili ne moraju uopšte da deluju na fenotip [8].

Mutacije obuhvataju sledeće vrste: supstitucija (zamena jednog nukleotida drugim), delecija (gubitak jednog nukleotida) i insercija (dodavanje jednog nukleotida). Supstitucije se mogu podeliti na tranzicije, transverzije, sinonimne i nesinonimne supstitucije. Tranzicija predstavlja zamenu jednog para purinske (A, G) i piramidinske (T, C) baze drugim purinsko-piramidinskim parom, dok je transverzija zamena jednog purinsko-piramidinskog para baza sa drugim piramidinsko-purinskim parom. Sinonimna supstitucija ne dovodi do promene primarne strukture proteina i samim tim ne uzrokuje velike posledice na organizam, dok se kod nesinonimnih supstitucija dešava suprotno [8].

Nesinonimne supstitucije mogu biti besmislene i smislene. Kod besmislenih mutacija[1] kodon se menja u zaustavni kodon i time se zaustavlja sinteza proteina. Samim tim rezultat ovih mutacija jeste nefukcionalan protein. Smislene mutacije (eng. *missense*) menjaju kodon u kodon koji određuje drugu aminokiselinu i kao takve mogu biti uzrok neutralnih promena u funkciji proteina, ali i ozbiljnih kancerogenih bolesti. Iz tog razloga, predstavljaju uzrok nastanka aminokiselinskih supstitucija (AKS), odnosno zamene jedne aminokiseline drugom. Primeri mutacija su dati na slici 2.3.

[1] Pojmovi (ne)sinonimne supstitucije i (ne)sinonimne mutacije će se u daljem tekstu ravnopravno koristiti.



Slika 2.3: Mutacije [8]

2.1.5 Mutacije u humanim kancerima

Kao što je prethodno rečeno, ne moraju sve mutacije doprineti razvoju malignih oboljenja, ali je od značaja detektovati one koje to rade. Mutacije koje omogućuju ubrzan rast ćelijama u kojima se nalaze i time utiču na razvoj kancera, nazivaju se vodeće (eng. *driver*) mutacije, a mutacije koje nemaju taj efekat prateće (eng. *passenger*) mutacija. Prateće mutacije jesu prisutne u ćeliji koja učestvuje u formiranju kancera, ali su biološki neutralne. Geni koji sadrže vodeće mutacije se nazivaju geni kancera, vodeći geni ili onkogeni [9, 10].

Stopa mutacije gena označava frekvenciju nastanka novih mutacija unutar tog gena tokom vremena. Geni postaju kandidati za vodeće gene ako u malignim uzorcima mutiraju u većoj meri nego što je to previđeno stopom mutacije. Mutacija sa većom frekvencijom sugeriše da ćelije sa mutacijom u tom genu mogu biti kancerogene i zato ti geni jesu kandidate za vodeće gene. Kada se na ovaj način odrede vodeći geni, njegove mutacije koje imaju predviđene efekte na aktivnost gena su vodeće mutacije [9, 10].

U nedavnim studijama je uočeno veće pojavljivanje pratećih mutacija u odnosu na vodeće mutacije. Takođe, broj somatskih tačkastih mutacija varira između različi-

tih klasa kancera, ali i između različitih uzoraka kancera iz iste klase. Iz tog razloga, teško je uočiti posebne karakteristike i razlike između vodećih i pratećih mutacija. Nesinonimne somatske mutacije imaju veće šanse da budu svrstane u vodeće mutacije zato što učestvuju u promeni funkcije i strukture proteina, dok se sinonimne mutacije uglavnom pojavljuju kao prateće mutacije. Nagomilavanje mutacija ovog tipa usled prekomernih deoba dovodi do nefunkcionalnog tkiva [9, 10].

2.2 Pojmovi mašinskog učenja

Nastanak mašinskog učenja je motivisan procesom učenja koji ljudi i životinje poseduju, kao i željom da se takav proces primeni u praktične svrhe. Cilj mašinskog učenja predstavlja proces generalizacije, odnosno da se od ograničenog broja podataka izvuče univerzalni zaključak. Prema prirodi problema mašinsko učenje možemo podeliti na tri grupe: nadgledano učenje, nenadgledano učenje i učenje potkrepljivanjem. Nadgledano učenje je jedna od najznačajnijih oblasti mašinskog učenja. Njegova glavna osobina je da su prisutni podaci, ne samo na osnovu kojih se uči, nego i ono što je potrebno naučiti. Drugim rečima, modelu su pored atributa koji opisuju instance na raspolaganju i njihove ciljne promenljive. Korišćenjem ovih podataka formirani model za nove instance zna da odredi nepoznatu vrednost ciljne promenljive. Međutim, kako je moguće izraditi više modela koji novim instancama pridružuje vrednosti ciljne promenljive, bitno je izdvojiti modele koji pri tome najmanje greše. Od značaja su modeli koji imaju najmanju razliku između pravih vrednosti ciljne promenljive i njihove predviđene vrednosti. Ovu razliku meri funkcija greške. Traženjem njenog optimuma može se naći model sa najmanjim brojem grešaka pri predviđanju [11].

Klasifikacija i regresija predstavljaju dva osnovna problema nadgledanog učenja. Predviđanje neprekidne ciljne promenljive je problem regresije, dok je predviđanje ciljne promenljive iz ograničenog broja ciljih vrednosti problem klasifikacije. Funkcija greške u slučaju klasifikacije se može prikazana uz pomoć indikatorske funkcije za koju veži:

$$I(F) = \begin{cases} 0 & \text{ako važi F} \\ 1 & \text{ako važi } \neg\text{F} \end{cases}$$

kada F označava tvrđenje. U tom slučaju, funkcija greške se naziva greška klasifikacije i ima oblik:

$$L(u, v) = I(u \neq v)$$

		Predviđene klase	
		Negativne	Pozitivne
Stvarne klase	Negativne	True negative-TN (stvarno negativne)	False positive-FP (lažno pozitivne)
	Pozitivne	False negative-FN (lažno negativne)	True positive-TP (stvarno pozitivne)

Tabela 2.1: Matrica konfuzije

gde u označava pravu vrednost klase, a v predviđenu. Jedna od često korišćenih funkcija greške u slučaju klasifikacije, a koja će biti korišćena u radu, je binarna unakrsna etropija (eng. *binary cross entropy*):

$$BUE = \frac{-1}{N} \sum_{i=1}^{N} y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log(1 - \hat{y}_i)$$

gde N predstavlja broj instanci, y_i ciljnu vrednost i $\hat{y}_i$ predviđenu vrednost od strane modela.

Ovaj rad se bavi klasifikacijom podataka i iz tog razloga mere kvaliteta modela koje će biti u fokusu su tačnost klasifikacije (eng. *classification accuracy*), preciznost (eng. *precision*), odziv (eng. *recall*), F1 mera, površina ispod ROC krive (eng. *area under the curve - AUC*) i balansirana tačnost (eng. *balanced accuracy*). Sve nabrojane mere kvaliteta modela počivaju na matrici konfuzije (eng. *confusion matrix*). Matrica konfuzije u slučaju binarne klasifikacije, koja lako može da se prevede na ostale slučajeve, prikazana je tabelom 2.1.

U narednom delu mere kvaliteta modela će biti opisane preko vrednosti matrice konfuzije:

- Tačnost klasifikacije:
$$\frac{TP + TN}{TP + TN + FP + FN}$$
- Preciznost:
$$\frac{TP}{TP + FP}$$
- Odziv:
$$\frac{TP}{TP + FN}$$

- F1 mera:
$$2 \cdot \frac{Preciznost \cdot Odziv}{Preciznost + Odziv}$$
- ROC kriva je kriva u koordinatnom sistemu gde y-osa predstavlja udeo TP instanci u skupu svih instanci i x-osa predstavlja udeo FP instanci u skupu svih instanci, čija je zapremina jedinični kvadrat. Kriva pokazuje kako se menja količina TP i FP instanci kada se prag odlučivanja koji se postavlja na skor klasifikatora menja od njegove minimalne do njegove maksimalne vrednosti.
- AUC predstavlja površinu ispod ROC krive.
- Balansirana tačnost:
$$\frac{Preciznost + Odziv}{2}$$

2.2.1 Neuronske mreže

Neuronske mreže predstavljaju jednu od najznačajnijih i najprimenjivijih ideja mašinskog učenja. Njihova upotreba se uočava u medicinskoj dijagnostici, kategorizaciji teksta, autonomnoj vožnji, prepoznavanju objekata sa slike i mnogim drugim oblastima. Postoje različiti tipovi neuronskih mreža: potpuno povezane, rekurentne, konvolutivne, rekurzivne i grafovske. Međutim, iako neuronske mreže imaju veliki uspeh i značaj, one se ne mogu koristiti za rešavanje svih problema. Neuronske mreže su posebno pogodne za probleme sa velikom količinom podataka. Učenje neuronskih mreža se može vrši i na osnovu sirove reprezentacije podataka [11].

Potpuno povezane neuronske mreže

Neuroni predstavljaju osnovnu jedinicu potpuno povezanih neuronskih mreža. Oni vrše izračunavanje linearne kombinacije ulaznih argumenata nad kojom se primenjuje neka aktivaciona funkcija. Najčešće aktivacione funkcije koje se upotrebljavaju su sledeće nelinearne funkcije:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$
$$tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$
$$relu(x) = max(0, x)$$

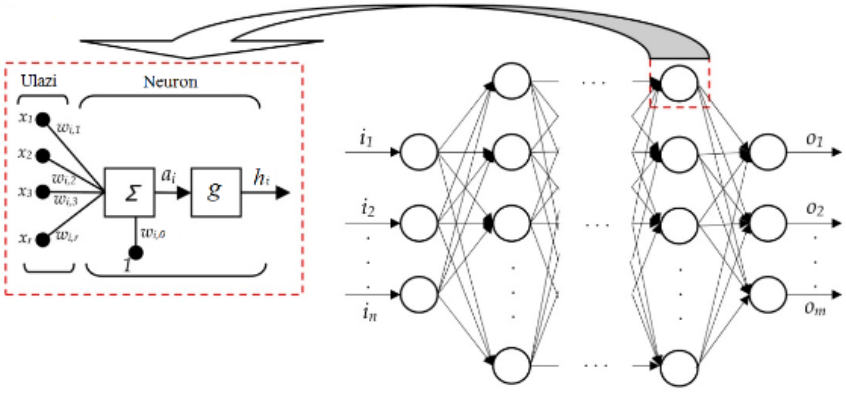
Ispravljena linearna jedinica (eng. *rectified linear unit - relu*) ima najveću popularnost zbog prikladnih svojstava za optimizaciju i osobine da je izvod u linearnom delu uvek 1, što otežava prerano zaustavljanje procesa optimizacije [11].

Neuroni su poređani po slojevima, gde svaki neuron kao ulaz prima izlaze svih neurona iz prethodnog sloja. Prvi sloj u mreži se zove *ulazni* i prima atribute mreže, dok je poslednji sloj *izlazni* nad čijim rezultatima neurona se može primeniti aktivaciona funkcija, ali i ne mora u zavisnosti da li se vrši klasifikacija ili regresija. Skriveni slojevi su slojevi između ulaznog i izlaznog sloja. Mreža koja ima više od jednog skrivenog sloja je duboka neuronska mreža. Model se može opisati na sledeći način:

$$h_0 = x$$

$$h_i = g(W_i h_i + w_{i0}) \qquad i = 1, 2, ...L$$

U ovoj formuli x je vektor ulaznih atributa, L je broj slojeva, g predstavlja aktivacionu funkciju, w_{i0} je vektor slobodnih članova i-tog sloja, W_i je matrica gde je k-ta vrsta vektor vrednosti parametara jedinice k u sloju i. Šema strukture potpuno povezane neuronske mreže je data na slici 2.4 [11].



Slika 2.4: Potpuno povezana neuronska mreža [11]

2.2.2 Ansambl

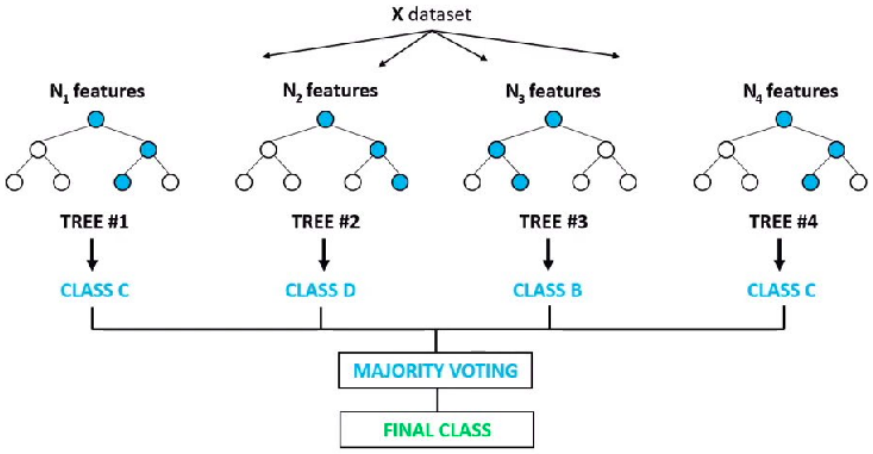
U slučajevima kada jedan model ne daje zadovoljavajuće rezultate i samostalno ne može da dođe do dobrih rešenja, a pritom daje relativno dobre predikcije, možemo koristiti princip *ansambla*. Ansambl predstavlja skup više modela koji zajedno

donose odluku. Pojedinačni modeli ne moraju biti previše precizni, ali njihove greške su nezavisne. Ukoliko se predviđanja svih pojedinačnih modela agregiraju, dobija se predviđanje ansambla [11].

Model slučajnih šuma

Slučajne šume (slika 2.5) predstavljaju ansembl stabala odlučivanja. Stabla odlučivanja se sastoje od čvorova, kojima su pridruženi različiti uslovi, i listova, kojima su pridružena predviđanja. U zavisnosti da li je uslov ispunjen ili ne, određenom granom se može ići iz trenutnog čvora u druge čvorove. Ovaj proces traje dok se ne stigne do lista stabla [11].

Broj stabala n, veličina podskupova instanci i atributa predstavljaju metaparametre za ovaj ansambl. Pojedinačno stablo može izabrati podskup instanci i podskup atributa za učenje. Povećavanjem broja stabala može se smanjiti prilagođenost modela na podatke nad kojima uči. Iz tog razloga broj stabala se može koristiti i kao regularizacioni parametar [11].



Slika 2.5: Slučajna šuma [12]

Balansirani *bagging* klasifikator

Bagging klasifikator predstavlja ansambl stabala odlučivanja, čiji modeli se uče na trening skupovima generisanim slučajnim izborom. Generisani trening skupovi su iste veličine kao i originalni trening skup, ali se oni međusobno razlikuju. Skup može

sadržati duplikate jedne instance, dok drugu instancu može u potpunosti izostaviti. Cilj ove tehnike je smanjenje šansi za preprilagođavanje [13].

Balansirani *bagging* klasifikator predstavlja jednu verziju *bagging* klasifikatora koji se često koristi u prisustvu nebalansiranih podataka. Način na koji klasifikator prevazilazi ovaj problem jeste balansiranje generisanih trening skupova. Tehnika koju balansirani *bagging* klasifikator koristi za balansiranje je *random undersampler*. Ova tehnika izbacivanjem slučajno izabranih instanci iz većinske klase, vrši smanjenje broja instanci većinske klase [13].

2.2.3 Naivni Bajesov algoritam

Naivni Bajesov algoritam koristi sledeću Bajesovu formulu:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

preko koje modeluje raspodelu ciljne promenljive y pri datim vrednostima promenljive x. Ovaj algoritam vrši modelovanje desne strane jednačine zbog lakšeg pristupa modelovanju. Sa desne strane nije teško odrediti raspodelu promenljive y zbog njene jednodimenzionalnosti, ali to ne važi za ocene $p(x|y)$ i $p(x)$. Međutim, pošto $p(x)$ ne zavisi od y i biće ista za sve njene vrednosti, ona se ne mora računati. Za izračunavanje ocene $p(x|y)$ pretpostvlja se uslovna nezavisnost atributa kada je data vrednost cilje promenljive:

$$p(x|y) = \prod_{i=1}^{n} p(x_i|y)$$

gde je x_i vrednost i-tog atributa. Sada su ove raspodele jednodimenzionalne i mogu se lako modelovati. Uslovna nezavisnost je slabija pretpostavka od nezavisnosti i nije poznato da li će biti ispunjena, zbog čega se pridev *naivni* nalazi u imenu ovog algoritma. Model se može predstaviti sledećom relacijom [11]:

$$p(y|x) \sim p(y) \prod_{i=1}^{n} p(x_i|y)$$

Komplementarni naivni Bajesov algoritam

Naivni Bajesov klasifikator ne funcioniše dobro u slučaju nebalansiranih podataka. Lako može doći do preprilagođavanja u korist većinske klase. U tom slučaju može se koristiti komplementarni naivni Bajesov klasifikator (eng. *complement naive*

Bayes - CNB). Ovaj klasifikator ne predviđa verovatnoću sa kojom instanca pripada nekoj klasi, već predviđa verovatnoće sa kojom instanca pojedinačno **ne** pripada svakoj od klasa. Od dobijenih vrednosti traži se najmanja, zato što su veće šanse da instanca pripada toj klasi. Model se može formalno predstaviti na sledeći način:

$$p(y)\prod_{i=1}^{n}\frac{1}{p(x_i|y)}$$

Kog naivnog Bajesovog algoritma najveća predviđena vrednost određuje klasu, dok je kod komplementarnog naivnog Bajesovog klasifikatora najmanja predviđena vrednost odgovorna za određivanje klase [14].

2.2.4 Logistička regresija

Logistička regresija (eng. *logistic regression - LR*) predstavlja model binarne klasifikacije koji korišćenjem Bernulijeve raspode određuje verovatnoću sa kojom instanca pripada jednoj ili drugoj klasi. Instanci se pridružuje klasa sa većom verovatnoćom pripadanja. Funkcija raspodela se predstavlja na sledeći način:

$$p(y|x) = \begin{cases} \lambda, & y = 1 \\ 1-\lambda & y = 0 \end{cases}$$

gde je y ciljna promenljiva i x vrednosti atributa. Međutim, ovakav model nije ispravan zato što predviđene vrednosti treba da predstavljaju verovatnoću i moraju biti unutar intervala $[0, 1]$. Kao rešenje ovog problema se koristi sigmoidna funkcija koja transformiše vrednosti iz intervala $[-\infty, \infty]$ u interval $[0, 1]$ i formuliše zavisnost parametra λ od vrednosti atributa x. Model se onda može predstaviti na sledeći način [11]:

$$p(y|x) = \sigma(w \cdot x)^y \cdot (1 - \sigma(w \cdot x))^{1-y}$$

2.2.5 Metod PCA

Algoritam analize glavnih komponenti ili algoritam PCA (eng. *principal component analysis*) pronalazi šablone u podacima nad kojima se primenjuje, preko kojih uočava njihove sličnosti i razlike. Zahvaljujući tome moguće je izvršiti kompresiju podataka na manje dimenzije bez velikog gubitka informacija koje podaci nose [15].

Pre primene algoritma PCA potrebno je obaviti standardizaciju podataka. Vrednosti različitih podataka mogu se nalaziti u različitim intervalima. Standardizacijom sve podatke prevodimo u vrednosti iz istog intervala kako bi podjednako doprineli istraživanju i analizi podataka. Način na koji se stadardizacija obavlja je nalaženje srednje vrednosti podataka za koju se oni umanje, a potom podele sa njihovom standaradnom devijacijom [15].

Da bi se uočile sličnosti u podacima potrebno je izračunati matricu kovarijanse. Ova matrica nam daje informaciju koliko su podaci međusobno u korelaciji. Na poziciji *(i, j)* stoji korelacija između podatka *i* i podatka *j*. Velika apsolutna vrednost korelacije između dva podatka predstavljaja veliku povezanost ta dva podatka. Uočavanje ponovljenih informacija nam pruža mogućnost da otklonimo iste [15].

Smanjenje dimenzionalnosti podataka se postiže izračunavanjem sopstvenih vektora i sopstvenih vrednosti dobijene matrice kovarijanske. Vektor koji se preslikava u samog sebe pomnoženim sa skalarom kada se na njega primeni neka linearna transformacije je sopstveni vektor, dok je skalar sopstvena vrednost tog vektora. Ispostavlja se da sopstveni vektor sa najvećom sopstvenom vrednošću predstavlja najznačajniju komponentu podataka. Korišćenjem ovog svojstva, svi sopstveni vektori se mogu opadajuće sortirati po svojim sopstvenim vrednostima nakon čega je moguće izabrati nekoliko početnih komponenti za predstavljanje podataka. Poslednjim korakom podatke reprezentujemo uz pomoć glavnih komponenti tako što se transponovana matrica sastavljena od vrednost izabranih sopstvenih vektora množi transponovanom vrednošću originalnih podataka [15].

2.3 Alati za predviđanje funkcionalnih efekata *missense* mutacija

Neki od postojećih alata za predviđanje funkcionalnih efekata *missense* mutacija koji se danas koriste su PolyPhen-2 i SIFT. Ovi alati predviđaju uticaj supstitucija aminokiselina na protein korišćenjem poznatih bioloških svojstava genomskih i proteinskih sekvenci. Međutim, i pored popularnosti ovih alata, oni poseduju određene mane. Naime, postoji određena količina osetljivosti u predviđanju funkcionalnih efekata mutacija u proteinskim regionima koji su izvan konzerviranog funkcionalnog domena (eng. *non Conserved Functional Domain - nCFD*). Ova informacija je od značaja zato što se većina malignih mutacija, kao i veliki broj neutralnih mutacija,

nalazi u nCFD. Još neki od alata ovog tipa su MutPred2, SNAP2 i MutationAssessor.

2.3.1 PolyPhen-2

PolyPhen-2 (eng. *Polymorphism Phenotyping v2*) predstavlja softver koji predviđa efekat mutacija metodama klasifikacije. Klasifikacija se obavlja korišćenjem naivnog Bajesovog klasifikatora koji koristi svojstva zasnovana na primarnoj i sekundarnoj strukturi proteina i kao rezultat daje verovatnoću da je neka mutacije neutralna ili štetna [16].

Izabrana svojstva koja se koriste pri klasifikaciji su svojstva zasnovana na sekvenci [16]:

1. Verovatnoća da se odabrana aminokiselina nalazi na određenom mestu kod normalne sekvence (eng. *position specific independent counts - PSIC*);
2. Razlika PSIC kod normalne i mutirane sekvence
3. Identičnost, odnosno broj poklapanja aminokiselina sekvence koja je mutirana i aminokiselina sekvence koja je najbliži homolog (sekvenca koja sadrži proizvoljnu aminokiselinu koja se nalazi na mestu razmatrane mutacije);
4. Podudarnost mutirane sekvence sa višestrukim poravnavanjem, odnosno najmanji broj operacija insercije, delecije i supstitucije da bi se od mutirane sekvence došlo do sekvence koja je rezultat višestrukog poravnavanja;
5. CpG (mesta gde se citozin nalazi pored guanina) kontekst tranzicionih mutacija (npr. kada C mutira u T);
6. Dubina poravnanja (bez praznina) na mestu mutacije;
7. Razlika u zapremini aminokiselina između normalne i mutirane sekvence;
8. Da li je mesto mutacije pozicionirano u anotiranom *Pfam domenu* (bazi proteinskih familija),

dok su svojstva zasnovana na strukturi proteina sledeća:

9. Dostupna površina aminokiseline kod normalne sekvence;
10. Promena hidrofobne sklonosti;

11. Kristalografski B faktor.

2.3.2 SIFT

SIFT (eng. *Sort Intolerant From Tolerant*) je alat koji na osnovu sekvence i informacija koje dobija poravnavanjem višestrukih sekvenci predviđa da li je mutacija na svakom mestu u sekvenci neutralna ili štetna. Predviđanje se vrši kroz sledeća četiri koraka [16]:

1. Izdvajanje sličnih sekvenci korišćenjem alata PSI-BLAST (eng. *Position-Specific Iterative Basic Local Alignment Search Tool*) nad bazom *UniRef90 2011 Apr*;

2. Odabir blisko povezanih sekvenci od dobijenih sekvenci u prethodnom koraku. Formiraju se grupe sekvenci koje imaju veliku sličnost (npr. postavljena granica može biti 90%), za svaku grupu se pravi njena konsenzus sekvenca i ona koja se najbolje poravna sa ulaznom sekvencom se bira. Računa se srednja vrednost kozerviranosti sekvence i ako je ona veća od date granične vrednosti konsenzus sekvenca se zadržava u poravnanju. Sve dok srednja vrednost konzerviranosti ne padne ispod granice, ovaj postupak se ponavlja;

3. Preuzimanje poravnanja. Poravnanja sekvenci izabranih u prethodnom koraku se mogu dobiti iz početnih rezulatata PSI-BLAST pretrage;

4. Izračunavanje verovatnoće. Na svakoj poziciji poravnanja svaka aminokiselina se pojavljuje sa frekvencijom n_i. Koristeći n_i, procenjuje se verovatnoća pojavljivanja aminokiselina prema *Dirichlet mixture* modelu.

Glava 3

Formiranje baze

Ovim poglavljem je prikazan proces prikupljanja podataka, njihovo filtriranje i formiranje skupa podataka koji će biti korišćen prilikom izrade alata *CancerMut* za detekciju štetnih aminokiselinskih supstitucija u proteinima uključenim u nastanak kancera. Prikazani su tipovi podataka, opisano je njihovo značenje kao i alati koji su korišćeni u ovom procesu.

Potrebno je formirati bazu sa neutralim mutacijama i mutacijama koje su asocirane sa malignim bolestima. Baza treba da sadrži informacije o genu u kom je prisutna mutacija, oznaku transkripta gena, oznaku mutacije i informaciju o klasi, odnosno da li je mutacija štetna ili ne. Podaci iz obe klase se prikupljaju na različite načine i na kraju se spajaju u jednu bazu.

3.1 Podaci o mutacijama asociranim sa kancerima

Podaci o mutacijama asociranim sa kancerima dostupni su na stranici COSMIC (eng. *Catalogue of Somatic Mutations in Cancer*) koja predstavlja najbogatiji izvor informacija o somatskim mutacijama u vezi sa ljudskim karcinomima [17]. Sa ovog sajta preuzete su dve datoteke.

3.1.1 Prva datoteka

Datoteka *CosmicMutantExport.tsv* predstavlja skup svih kancerogenih mutacija koje se javljaju samo na jednom mestu u genu. Ova datoteka ima 40 atributa i 1 048 566 instanci. Međutim, nama nisu potrebni svi atributi iz ove baze, zato izdvajamo sledeće:

1. `Gene name` - niska koja označava ime gena u kom je prisutna mutacija,

2. `Mutation AA` - niska koja daje informaciju koja zamena i gde je izvršena u sekvenci proteina. Na primer, tekst *p.P1903S* znači da se na 1903. mestu u proteinskoj sekvenci umesto aminokiseline P nalazi S.

3. `Mutation Description` - niska koja govori o tipu mutacije (supstitucija, insercija, delecija, nepoznata ili slično)

4. `Mutation somatic status` - niska koja ima vrednost *Confirmed Somatic* ako je mutacija testiranjem potvrđeno somatska, *Previously observed* ako imamo informaciju da je mutacija bila somatska u prethodnom radu, ali ne i u trenutnom i *Variant of unknown origin* kada pretpostavljamo da je mutacija somatska, ali nemamo potvrdu.

Dobijene podatke potrebno je filtrirati sa više uslova. Za početak, iz ove baze nama su potrebne mutacije koje su potvrđeno somatske, što znači da atribut `Mutation somatic status` ima vrednost *Confirmed Somatic*. Takođe, značajne su nam samo *missense* mutacije, zato uzimamo instance gde je vrednost atributa `Mutation Description` niska *Substitution - Missense*.

3.1.2 Druga datoteka

Datoteka *`proteinSeqAndSymbol.tsv`* formirana na osnovu podataka sa ovog sajta sadrži listu gena uključenih u nastanak kancera u slučajevima kada ti geni imaju *missense* mutacije. Pomenute gene dobijamo iz liste COSMIC cenzusa[1].

Pored imena gena, u datoteku su naknadno dodate sekvence proteina koje gen kodira i oznaka transkripta gena koji koristimo. Neki geni mogu imati više transkripata, dok neki imaju samo jedan. U oba slučaja, korišćen je takozvani *divlji tip* transkripta (eng. *wild type*). Ove oznake neće biti korišćene u kasnijem modelu, ali je važno znati oznaku gena koji je korišćen.

[1]COSMIC cenzus predstavlja listu gena povezanih sa malignim oboljenjima koji se koriste kao standard za osnovna istraživanja u medicini i farmakologiji. Međutim, lista gena u COSMIC cenzusu nije stalna i ažurira se na osnovu novih istraživanja [17].

3.1.3 Finalna datoteka mutacija asociranim sa kancerima

Kada imamo listu gena iz druge datoteke, potrebno je prvu datoteku filtrirati tako da ostanu samo geni koji se nalaze u drugoj datoteci. Ostali geni nam nisu od značaja. Potom se duplikati izbacuju iz podataka.

U ovako dobijenom skupu podataka nama su značajne kolone koje sadrže informaciju o imenu gena i oznaci mutacije. Ove podatke izdvajamo i kasnije spajamo sa podacima o neutralnim mutacijama. Sve instance dobijene na ovaj način će u koloni koja označava klasu imati vrednost *1*, što označava štetnu mutaciju.

3.2 Podaci o neutralnim mutacijama

Podaci o neutralnim mutacijama se nalaze u bazi ***dbSNP*** dostupnoj na sajtu Nacionalnog centra za biotehnološke informacije (eng. *The National Center for Biotechnology Information* - NCBI). NCBI predstavlja deo američke Nacionalne biblioteke za medicinu (eng. *National Library of Medicine* - NLM) u okviru Nacionalnog instituta za zdravlje (eng. *National Institutes of Health* - NIH). Uloga ovog centra je unapređivanje nauke i zdravlja kroz pristup biomedicinskim i genomskim informacijama.

Baza *dbSNP* (eng. *single nucleotide polymorphism database*) je besplatna, javno dostupna baza koja nam pruža informacije o varijacijama pojedinačnih nukleotida, kratkim tandemskim ponavljanjima gena, malim insercijama i delecijama zajedno sa publikacijama i molekularnim posledicama [18]. Pomenuta baza sadrži podatke o neutralnim mutacijama potvrđenim od strane eksperata koje možemo preuzeti pomoću veb interfejsa ***Entrez***.

Veb interfejs ***Entrez*** predstavlja primarni način za pretragu i pronalaženje potrebnih informacija koje se nalaze u bazama datim na prethodno pomenutom sajtu. ***Entrez*** nudi različite opcije za konstruisanje preciznih pretraga i upravljanje rezultatima. Pruža unapred podešene filtere čime olakšava fokusiranje na određene vrste rezultata. Date su mogućnosti odabira baze podataka koju želimo da pretražimo, kao i specijalizovana polja za filtriranje za svaku od njih [19].

Za filtriranje relevantnih podataka potrebno je navesti sledeći upit:

$$(Gene\,Name[GENE]\ OR\ Gene\,Name[GENE]\ OR\ ...)\ AND\ missense\,variant[Function\_Class]$$

pri čemu je umesto *Gene Name* potrebno navesti sve nazive gena koje smo dobili u drugoj datoteci u prethodno navedenom odeljku. Nakon toga, dobijeni podaci se

preuzimaju u vidu *xml* datoteke, koja se može dodatno filtrirati. Za svaku instancu u datoteci izdvajamo ime gena, oznaku mutacije i listu malih alelskih frekvencija (eng. *minor allele frequency* - MAF). MAF vrednosti pomažu u razlikovanju uobičajnih mutacija od retkih. Za izradu modela relevantne su sve mutacije koje imaju bar jednu MAF vrednost između *0.01* i *0.5*. Takve mutacije zadržavamo, dok ostale izbacujemo zajedno sa duplikatima.

Pored pomenute baze, korišćene su neutralne mutacije iz još tri izvora. Neutralne mutacije koje su učestvovale u izradi alata PolyPhen2 [20], alata MutPred2 [21] i neutralne mutacije iz baze *humsavar* [22] na strani *UniProt* (eng. *Universal Protein resource*), koja predstavlja skup baza podataka Švajcarskog bioinformatičkog instituta (eng. *Swiss Institute of Bioinformatics* - SIB), Evropskog bioinformatičkog instituta (eng. *European Bioinformatics Institute* - EBI) i Proteinskog informacionog resursa (eng. *Protein Information Resource* - PIR) [23].

Skup neutralnih mutacija mora biti u istom formatu kao i skup štetnih mutacija, sa istim kolonama u istom redosledu. Zato izdajamo ime gena, oznaku mutacije i dodajemo kolonu za klasu sa vrednošću *0*, kao oznaku neutralnih mutacija. Pre spajanja skupa štetnih i neutralnih mutacija potrebno je izbaciti mutacije koje se nalaze u oba skupa i proveriti ispravnost proteinskih sekvenci. Ispravne su one sekvence koje su usaglašene sa svojim oznakama (npr. *p.P1903S* je ispravna ako se u proteinskoj sekvenci divljeg tipa na 1903. mestu nalazi aminokiselina sa oznakom *P*). Kada ostanu samo ispravni podaci može se izvršiti spajanje podataka.

3.3 Finalni skup podataka

Nakon spajanja prve i druge datoteke, dobija se objedinjena datoteka pod nazivom `finalBase.csv`. Sledeći korak je mutacijama iz ove datoteke pridružiti osobine koje će biti od koristi prilikom izgradnje klasifikatora *CancerMut*. Potrebne osobine se dobijaju primenom alata *EpiMut* [24] koji izračunava *EpiMut* skor mutacija, zasnovan na biohemijskim i fizičko-hemijskim karakteristikama aminokiselina i pristupu obrade digitalnog signala pri analizi sekvence proteina. Alat radi na principu naivnog Bajesovog klasifikatora i daje rezultate za svaki gen pojedinačno. Lista pomenutih osobina data je u AAindex bazi [25]. Neke od osobina su:

1. `Hydrophobicity index` - indeks rastvorljivosti aminokiseline u vodi;

2. `Retention coefficient in HFBA` - koeficijent retencije u heptafluorobuterna kiselini;
3. `Partial specific volume` - promena u zapremini rastvora kada se izmerena količina rastvora doda;
4. `Solvation free energy` - promena u slobodnoj energiji povezanoj sa prelaskom molekula iz idealnog gasa u rastvor, na određenoj temperaturi i pod određenim pritiskom;
5. `pK-C` - proteinska kinaza C, predstavlja skup enzima koji učestvuju u kontroli proteina;
6. `Positive charge` - pozitivno naelektrisanje;
7. `Negative charge` - negativno naelektrisanje;
8. `Isoelectric point` - pH vrednost na kojoj je naelektrisanje neutralno;
9. `Bitterness` - gorčina;
10. `Refractivity` - indeks prelamanja svetlosti;

Rezultat *EpiMut* alata je predstavljen putem 553 skora. Modeli kreirani u ovom radu će koristiti bazu podataka koja sadrži pomenute skorove, odnosno oni će biti atributi za učenje modela.

Glava 4

Korišćene tehnologije i alati

Prilikom izrade ovog rada korišćeni su različiti alati i tehnologije. Iz tog razloga, naredni odeljak je posvećen njihovom detaljnijem opisu, kao i načinu korišćenja. Predstavljen je značaj izabranih alata i tehnologija, kao i prednosti koje oni pružaju.

4.1 Python

Rad na *Python-u* počeo je 1980-ih godina, dok se zvanično prvi put pojavio 1991. godine. Objavio ga je Gvido van Rosum motivisan programskim jezikom *ABC*. Danas je ovaj jezik jedan od najpopularnijih i najkorišćenijih programskih jezika. *Python* predstavlja interpretabilni, objektno-orjentisani programski jezik višeg nivoa. Pored svoje jednostavnosti i sintakse lake za učenje, *Python* pruža raznolike mogućnosti. Podržava rad sa klasama, modulima, izuzecima, dinamičkim povezivanjem i proverom tipova. Omogućava korišćenje velikog broja bibloteka i interfejs za korišćenje mnogih sistemskih poziva. Verzija *Python* jezika koja je korišćena prilikom izrade ovog rada je *Python 3.7.7*.

Zbog svoje fleksibilnosti, odličnih mogućnosti vizualizacije i korišćenja velikog broja biblioteka, *Python* predstavlja najpoželjniji jezik za mašinsko učenje. Podržavanjem više paradigmi daje mogućnost korisniku da izabere najlakši način za rešenje problema.

4.1.1 NumPy

NumPy (eng. *numerical python*) je *Python* biblioteka otvorenog koda. Koristi se pri rešavanju problema iz domena linearne algebre i pruža podršku za rad sa Fu-

Slika 4.1: Python logo

rijeovim transformacijama, nizovima i matricama. *NumPy* pruža različite metode vezane za objekat niza, *ndarray*, koje olakšavaju i ubrzavaju proces dobijanja rešenja. Razlog za ubrzanje leži u načinu čuvanja niza i listi. Elementi niza, za razliku od listi, se u memoriji čuvaju jedan do drugog, zbog čega su susedni elementi niza na susednim adresama. Iz tog razloga, omogućeno je lakše upravljanje nizovima i pristupanje njihovim elementima.

Slika 4.2: NumPy logo

4.1.2 Pandas

Pandas predstavlja *Python* biblioteku otvorenog koda koja olakšava upravljanje podacima. Nudi mogućnost lakog čitanja, pisanja, čišćenja i analiziranja podataka. Olakšava formatiranje podataka, kao i spajanje i pridruživanje više skupova podataka. *Pandas* se, zahvaljujući ovim osobinama, često koristi prilikom rešavanja problema iz oblasti mašinskog učenje i istraživanja podataka.

Slika 4.3: Pandas logo

4.1.3 Keras

Keras predstavlja interfejs za duboko učenje koji radi na platformi otvorenog koda za mašinsko učenje TensorFlow. Glavne osobine ovog interfejsa su njegova moć, jednostavnost i fleksibilnost. Ima odličnu dokumentaciju i korisnički vodič. Zbog svoje čvrste povezanosti sa TensorFlow platformom, pokriva sve aspekte mašinskog učenja, od upravljanja podacima, nalaženja modela i hiperparametara, do izrade rešenja za primenu. Keras koriste američka Nacionalna vazduhoplovna i svemirska organizacija (eng. *National Aeronautics and Space Administration* - NASA), Evropska organizacija za nuklearno istraživanje (eng. *European Organization for Nuclear Research* - CERN), američki Nacionalni institut za zdravlje (eng. *National Institutes of Health* - NIH) i mnoge druge naučne organizacije širom sveta.

Slika 4.4: Keras logo

4.2 Jupyter Notebook

Jupyter je neprofitni projekat otvorenog koda koji je nastao kao rezultat projekta IPython 2014. godine razvijenog na *GitHub-u*. Podržava interaktivno istraživanje podataka i naučno izračunavanje na različitim programskim jezicima. Jedan od rezultata ovog projekta predstavlja veb aplikacija otvorenog koda *Jupyter Notebook*. Neke od mogućnosti koje ovaj alat nudi su kreiranje i deljenje dokumenata sa kodovima, jednačinama, vizualizacijom podataka, narativan tekst i još mnogo toga. Prednosti koje *Jupyter Notebook* nudi, a koje su relevantne za ovaj rad, predstavljaju mogućnost čišćenja i transformisanja podataka, vizualizaciju podataka i mogućnost upotrebe algoritama mašinskog učenja.

4.3 Osobine računara

Karakteristike računara na kojem je rađen projekat su sledeće:

- Brend: *Lenovo ThinkPad*

Slika 4.5: Jupyter logo

- Operativni sistem: *Ubunto 18.04 LTS*, 64-bit
- Procesor: *Intel® Core™ i3-6100U CPU @ 2.30GHz × 4*
- Memorija: 8 *GB*
- Disk : SSD-240 *GB*

Glava 5

Razvoj modela i analiza performansi

U daljem tekstu je prikazan postupak pretprocesiranja podataka, formiranja modela i tumačenje njihovog rezultata. Ispitani su modeli koji rade na principu ansambla (slučajne šume, balansirani *bagging* klasifikatori, XGBoot ansambli), naivni Bajesovi klasifikatori, logistička regresija i potpuno povezane neuronske mreže. Određivanje hiperparametara pomenutih modela je izvršeno validacijom, dok se ocenjivanje modela vrši u odnosu na test skup.

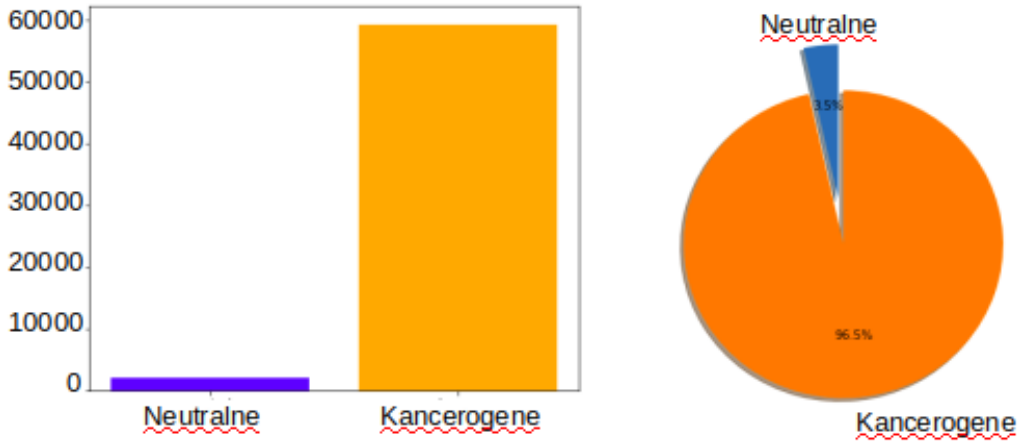
5.1 Pretprocesiranje

Po učitavanju baze, podaci se dela na trening, validacioni i test skup u razmeri 60:20:20, stratifikovano po ciljnoj promenljivoj. Vrši se skaliranje podataka skalarima dobijenim iz trening skupa. Posle provere, utvrđeno je da su mnogi atributi u velikoj međusobnoj korelaciji. Takođe, postoje i atributi koji predstavljaju različite verzije, odnosno definicije, istih karakteristika mutacija, što prouzrokuje da njihove vrednosti budu približno iste. Prilikom opisa atributa u AAindex bazi dato je ime autora i godina izdanja definicije na koju se atribut odnosi (npr. atributi u potpunoj korelaciji su *Electron-ion interaction potential values (Cosic, 1994)* i *Electron-ion interaction potential (Veljkovic et al., 1985)*, *Hydrophobicity index (Engelman et al., 1986)* i *Hydrophobicity (Prabhakaran, 1990)*, *Hydrophobicity index (Argos et al., 1982)* i *Hydrophobicity (Jones, 1975)*, *Optimal matching hydrophobicity (Sweet-Eisenberg, 1983)* i *SWEIG index (Cornette et al., 1987)*, *Residue volume (Bigelow, 1967)* i *Residue volume (Goldsack-Chalifoux, 1973)* i tako dalje).

Kako bi smanjili korelaciju između atributa, pre samog učenja modela upotrebljen je algoritam PCA. Sa ovim algoritmom dobijamo bazu manje dimenzionalnosti

sa novim međusobno linearno nekorelisanim atributima, gde prvi atribut ima najveću varijansu. Od 553 atributa sada imamo 300, pri čemu je zadržano oko 91% informacija.

Pre izrade modela uočen je problem nebalansiranih podataka. U bazi postoji veće prisustvo štetnih instanci u odnosu na neutralne (slika 5.1). Pre treniranja modela u nastavku biće pomenute korišćene metode za prevazilaženje ovog problema.



Slika 5.1: Udeo klasa u bazi podataka

5.2 Implementacija modela i rezultati testiranja

Postupkom validacije mogu se naći najbolji mogući hiperparametri modela za date podatke. Modeli tokom ovog postupka daju rezultate u odnosu na validacioni skup sa 534 neutralnih i 14 810 štetnih instanci. Postavljanjem niskih vrednosti za hiperparametre dobijaju se jednostavni modeli koji ne mogu iz podataka izvući informacije potrebne za razlikovanje klasa, dok sa visokim vrednostima hiperparametara dobijamo kompleksnije modele koji previše zavise od podataka nad kojima uče. Cilj je naći vrednosti hiperparametara za koje modeli imaju najuspešniju moć učenja iz prosleđenih podataka, ali se njima ne preprilagođavaju. Izabrani hiperparametri se koriste prilikom formiranja finalnog modela.

Ocenjivanje modela se vrši u odnosu na test skup sa 668 neutralnih i 18 512 štetnih instanci. Prilikom ocenjivanja i analize rezultata modela od značaja je izdvojiti mere kvaliteta koje su relevantne u slučaju nebalansiranih podataka. Iz tog razloga, rezultati svih obrađenih modela biće prikazani preko matrica konfuzije i balansirane tačnosti klasifikacije.

5.2.1 Slučajne šume

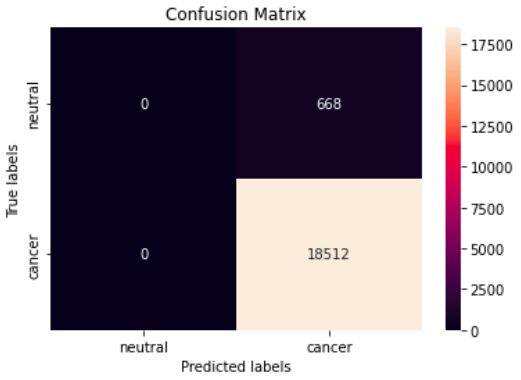
Prilikom formiranja modela slučajnih šuma potrebno je ispitati hiperparametre koji predstavljaju najveću dubina stabla i broj stabala odlučivanja. Tokom ovog procesa, uočeno je da se pri povećanju vrednosti ovih hiperparametara povećava balansirana tačnost klasifikacije samo na trening skupu, dok se na validacionom skupu ne zapaža velika promena. Ovo zapažanje je uočeno rano u validacionom procesu, čak i pri korišćenju niskih vrednosti za broj stabala i maksimalne dubine stabla. Kako promene hiperparametara nisu uticale na kvalitet modela, hiperparametri su podešeni na vrednosti sa kojima je balansirana tačnost pokazala mali rast, ali dovoljno niske da ne dođe do preprilagođavanja modela.

Posmatrani su rezultati modela treniranog sa podacima nad kojima su primenjene tehnike za balansiranje i modela koji tokom treniranja podešava težine klasa. Modeli koji koriste balansirane podatke imaju najveću dubinu stabla postavljenu na `10` i broj stabala postavljen na `30`. Tehnika *random oversampler* vrši umnožavanje broja instanci manjinske klase nasumičnim ponavljanjem. Kao parametar se ovoj metodi prosleđuje *sampling_strategy* što predstavlja željeni odnos klasa nakon umnožavanja. Korišćenjem ove metode dobija se model koji sve instance klasifikuje kao štetne (slika 5.2). Približne rezultate je dala i tehnika *random undersampler* koja smanjuje broj instanci većinske klase nasumičnim uklanjanjem njenih instanci dok se ne dođe do željenog odnosa klasa (slika 5.3).

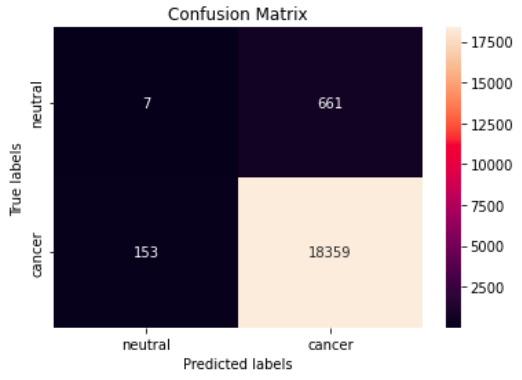
Kako pojedinačne tehnike nisu dale rezultat, ispitana je i kombinacija tehnika *SMOTE* i *random undersampler*. *SMOTE* (eng. synthetic minority oversampling technique) tehnika služi za generisanje instanci manjinske klase posmatrajući k najbližih suseda za svaku instancu ove klase koje, takođe, pripadaju manjinskoj klasi. Nove instance se interpoliraju duž pravca koji spaja izabranu instancu sa njenim susedima. Zato što nema ponavljanja instanci, manja je opasnost od preprilagođavanja, ali je i veća šansa za stvaranje šuma koji može pripadati većinskoj klasi [26]. Međutim, kako je vrednost balansirane tačnosti modela oko vrednosti 0.5, nijedna tehnika balansiranja podataka nije doprinela poboljšanju rezultata modela (slika 5.4).

Treniranje modela koji postavlja težine klasa se vrši postavljanjem hiperparametra *class_weight* na vrednost *balanced*. Prethodno podešavanje omogućava automatsko određivanje težina manjinske i većinske klase podataka formulom:

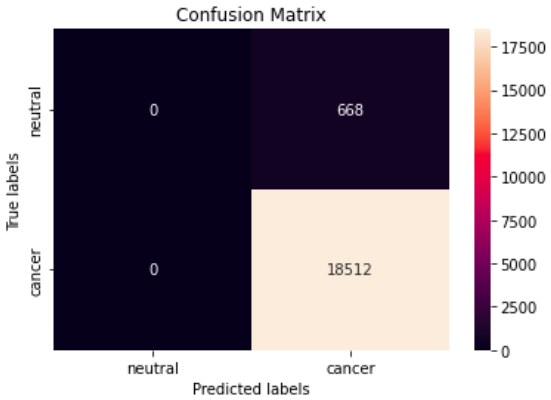
$$w_i = N/(n_c \cdot n_i)$$



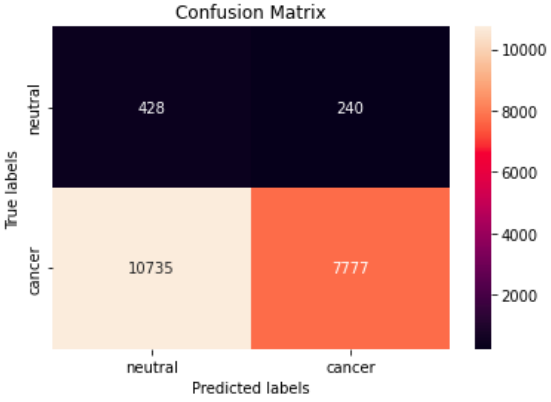
Slika 5.2: RF - *oversampling*



Slika 5.3: RF - *undersampling*



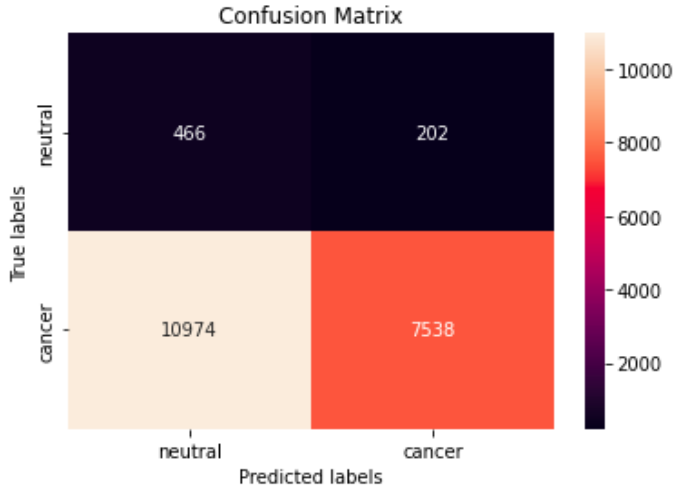
Slika 5.4: RF - kombinovana metoda



Slika 5.5: RF - podešene težine klasa

gde je N broj instanci, n_c broj klasa i n_i broj instanci koje pripadaju klasi i. Veća težina se dodeljuje manjinskoj klasi kako bi se obezbedila veća kazna pri njenoj pogrešnoj klasifikaciji, dok se suprotno radi sa većinskom klasom. Da bi se izbeglo preprilagođavanje modela potrebno je podesiti hiperparametre broj stabala na 5 i maksimalnu dubinu stabla na 3. Rezultat ovog modela daje nisku vrednost balansirane tačnosti koja iznosi 0.53 (slika 5.5).

Kako bi se ispitale sve mogućnosti formiran je i model slučajnih šuma koji kombinuje balansiranje podataka korišćenjem metode *random undersampler* i podešavanje težina klasa. Vrednost hiperparametara je ista kao i kod prethodnog modela iz pomenutih razloga. Iz matrice konfuzije se vidi da je vrednost balansirane tačnosti modela 0.53 (slika 5.6). Posmatran model ima najveći broj prepoznatih neutralnih instanci, ali je udeo grešake pri klasifikaciji štetnih instanci znatno veći. Dobijene vrednosti jasno ukazuju da model ne prepoznaje razliku između dve klase.



Slika 5.6: RF - *undersampling* i podešene težine klasa

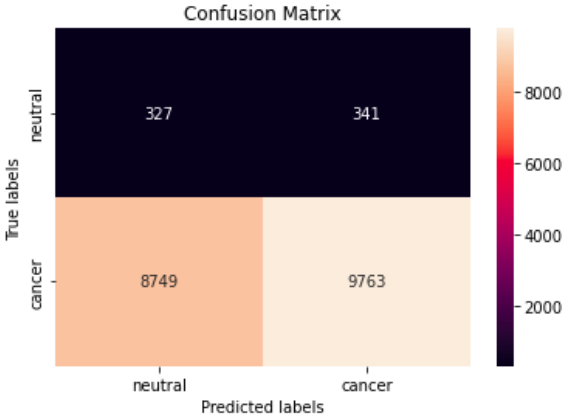
5.2.2 Primena praga na model slučajnih šuma

Model slučajnih šuma nije dao zadovoljavajuće rezultate prilikom korišćenja tehnika za balansiranje podataka. Iz tog razloga, pokušano je sa primenom praga na formiran model slučajnih šuma sa podešavanjem težina klasa. Ovaj princip predstavlja nalaženje geometrijske sredine čijom optimizacijom se traži balans između udela tačno klasifikovanih neutralnih instanci i udela tačno klasifikovanih štetnih instanci. Formula je sledeća [27]:

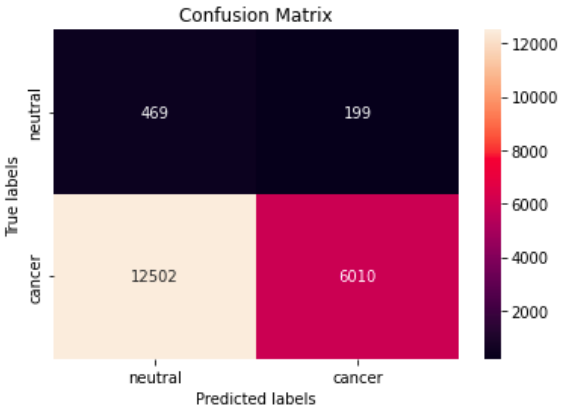
$$G\_sredina = \sqrt{TPR \cdot (1 - FPR)}$$

Izračunat indeks najveće vrednosti geometrijske sredine se koristi za nalaženje praga koji će biti upotrebljen. Ako je predviđena verovatnoća veća od izabranog praga, instanca se klasifikuje kao štetna, u suprotnom se ista svrstava u neutralnu klasu. Dobijeni model daje balansiranu tačnost koja ne prelazi vrednost 0.51 (slika 5.7).

Drugi način nalaženje praga je direktan, kada se od ponuđenih vrednosti za prag traži optimalna izračunavanjem balansirane tačnosti prilikom primene svih vrednosti za prag [27]. Prosleđene vrednosti praga su predstavljene kao niz vrednosti od 0 do 1 sa korakom 0.001. Preko ovog pristupa izabrana je vrednost od 0.623. Međutim, formirani model vrši pogrešno klasifikovanje oko 67% štetnih instanci (slika 5.8). Pored činjenice da model daje oko 70% tačno klasifikovanih neutralnih instanci, balansirana tačnost modela iznosi samo 0.513.



Slika 5.7: Matrica konfuzije - geometrijska sredina



Slika 5.8: Matrica konfuzije- direktan pristup

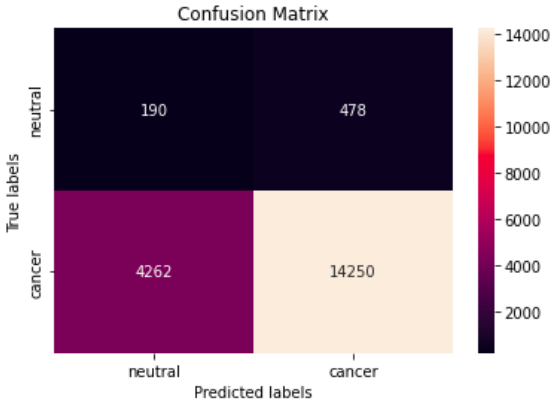
5.2.3 Balansirani *bagging* model

Drugi model koji radi na principu ansambla, a koji koristi nasumično generisane balansirane trening skupove je balansirani *bagging* model. Balansirani *bagging* modeli za hiperparametar imaju broj stabala odlučivanja. Ispitivanjem ovog hiperparametra uočeno je da promena njegovih vrednosti ne doprinosi poboljšanju relevantnih mera kvaliteta modela. Modeli brzo postaju preprilagođeni, dok je balansirana tačnost validacionog skupa uvek blizu vrednosti 0.5. Broj stabala odlučivanja je iz tog razloga postavljen na podrazumevanu vrednost `10`.

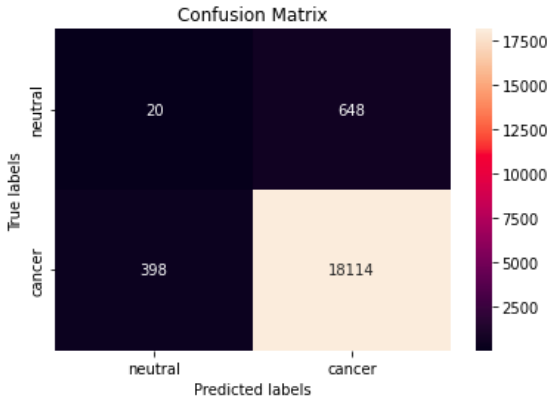
Rezultati dati matricama konfuzije predstavljaju rešenja modela treniranog sa nebalansiranim podacima (slika 5.9) i modela treniranog sa podacima posle upotrebe tehnika za balansiranje podataka (korišćenje *random oversampler* slika 5.10, korišćenje *random undersampler* slika 5.11 i korišćenje kombinacije metoda *SMOTE* i *random undersampler* slika 5.12). Vrednosti balansirane tačnosti modela nisu veće od 0.524. Modeli ne daju zadovoljavajuće rezultati, odnosno ne nalaze razliku između zadatih klase.

5.2.4 *XGBoost* klasifikator

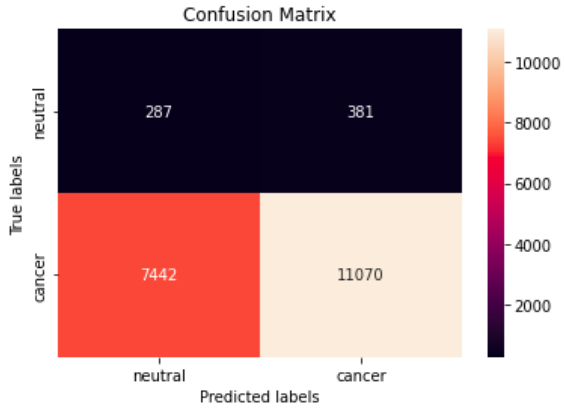
Poslednji ispitan model koji radi na principu ansambla je *XGBoost* (eng. *extreme gradient boosting - XGB*) klasifikator. Modeli koje *XGBoost* ansambl koristi su klasifikaciona i regresiona stabla (eng. *classification and regression trees - CART*) koja predstavljaju jednu vrstu stabala odlučivanja. Jedinstvena odlika CART-a je da su to binarna stabla koja uzimaju u obzir pogrešne predikcije iz prethodnog



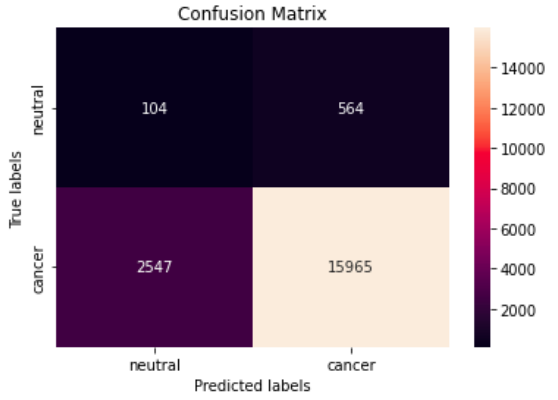
Slika 5.9: BBC bez balansiranja



Slika 5.10: BBC - *oversampling*



Slika 5.11: BBC - *undersampling*

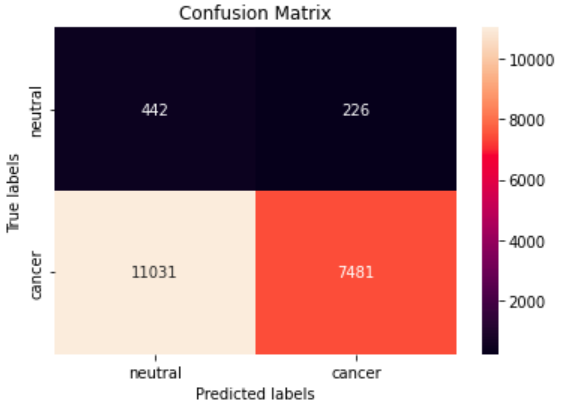


Slika 5.12: BBC - kombinovana metoda

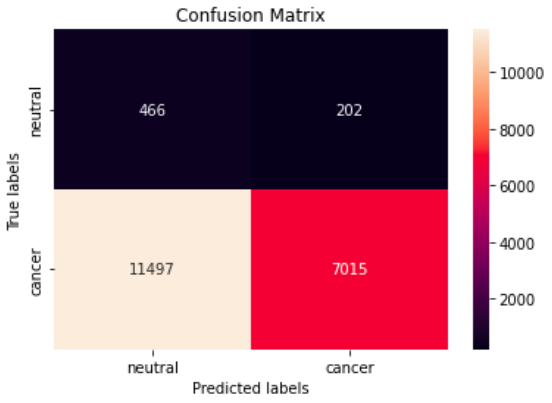
stanja kako bi se odredio najbolji uslov za podelu podataka [28].

Ispitane su različite vrednosti hiperparametara koji označavaju broj stabala i maksimalnu dubinu stabla. Kao i za prethodne modele koji rade na principu ansambla, uočeno je da se relevantne mere kvaliteta ne menjaju sa promenom vrednosti hiperparametara, dok se model brzo preprilagođava. Izabrane vrednosti su one koje daju najveću balansiranu tačnost bez preprilagođavanja modela. Hiperparametri klasifikatora su sledeći:

- *objective* : `binary:logistic` - funkcija greške
- *learning_rate* : `0.1` - koeficijent učenja
- *max_depth* : `3` - maksimalna dubina stabla
- *use_label_encoder* : `False` - ne koristi se enkoder za diskretne vrednosti



Slika 5.13: XGB bez balansiranja



Slika 5.14: XGB - *oversampling*

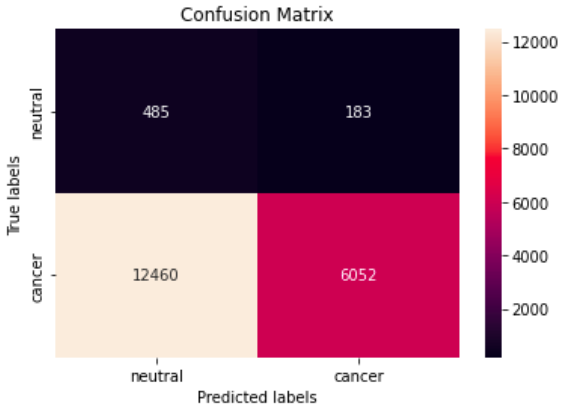
- *n_estimators* : `5` - broj CART modela
- *max_delta_step* : `1` - podešavanje koraka ažuriranja, pozitivna vrednost je potrebna zbog prisustva nebalansiranih podataka
- *scale_pos_weight* - kontrola balansa između težina klasa, koristi se kod prisustva nebalansiranih podataka. Hiperparametar se često postavlja na vrednost: `sum(negative instances)/sum(positive instances)`.

Prikazani su rezultati modela treniranog sa nebalansiranim skupom podataka (slika 5.13) i treniranog sa podacima posle primene metoda za balansiranje (primena *random oversampler* slika 5.14, primena *random undersampler* slika 5.15 i primena kombinacije pomenutih tehnika slika 5.16). Najveću vrednost balansirane tačnosti na test skupu ima model pre koga se primenjuje *random oversampler* tehnika i iznosi 0.538, što ne predstavlja pouzdanu vrednost za prepoznavanje različitih klasa među podacima.

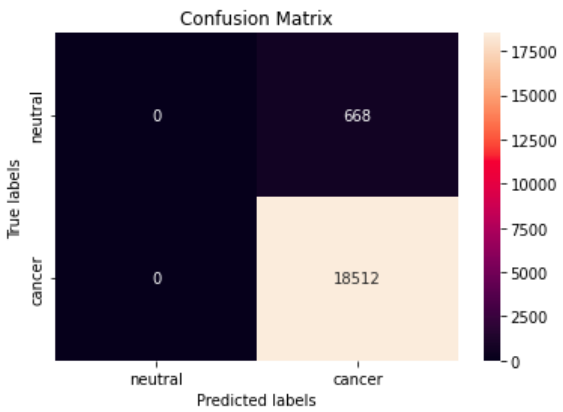
5.2.5 Komplementarni naivni Bajesov model

Naivni Bajesovi klasifikatori zbog velike nebalansiranosti podataka ne daju dobre rezultate. Iz tog razloga, korišćeni su CNB klasifikatori. CNB modeli, dostupni preko *naive_bayes* biblioteke *sklearn*-a, ne podržavaju korišćenje negativnih vrednosti podataka. Iz tog razloga, na trening skup koji CNB modeli koriste nije primenjen algoritam PCA koji proizvodi negativne vrednosti atributa.

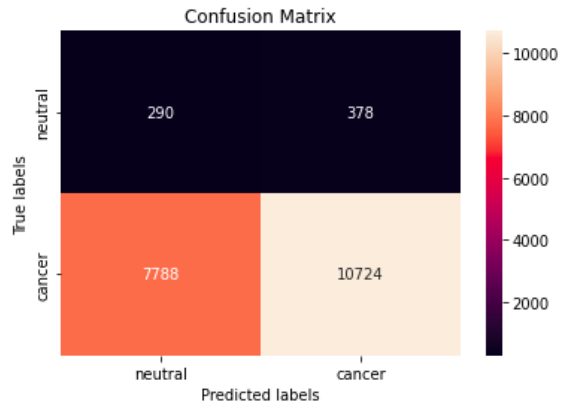
Posmatran je rezultat CNB modela treniranog sa nebalansiranim podacima (slike 5.17) i sa balansiranim podacima primenom pomenutih tehnika (primena *ran-*



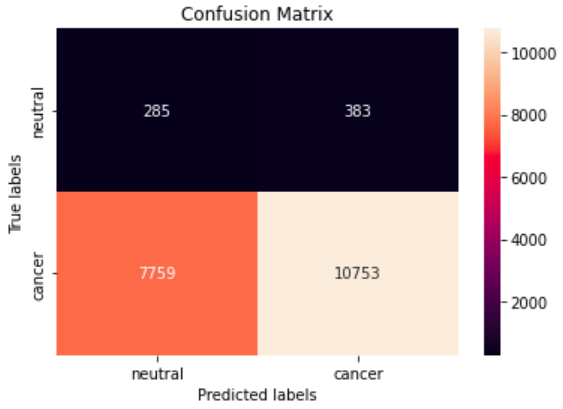
Slika 5.15: XGB - *undersampling*



Slika 5.16: XGB - kombinovana metoda



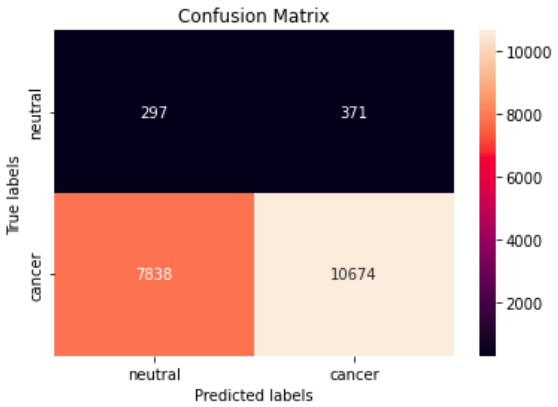
Slika 5.17: CNB bez balansiranja



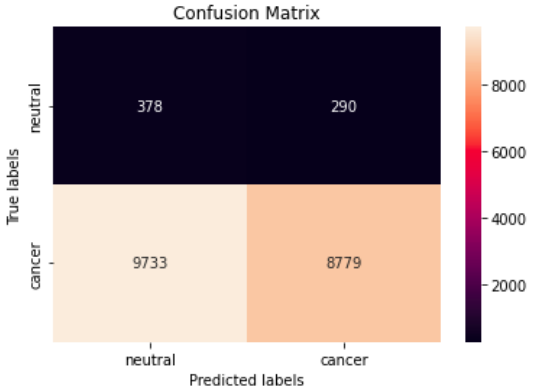
Slika 5.18: CNB - *oversampling*

dom undersampler slika 5.18, primena *random oversampler* slika 5.19 i kombinacije tehnika *SMOTE* i *random undersampler* slika 5.20). Prikazano je i rešenje CNB klasifikatora prilikom podešavanja težina instanci pri njegovom treniranju, što se realizuje postavljanjem *sample_weight* hiperparametra na vrednost dobijenu preko funkcije *compute_sample_weight* date u biblioteci *sklearn* (slika 5.21).

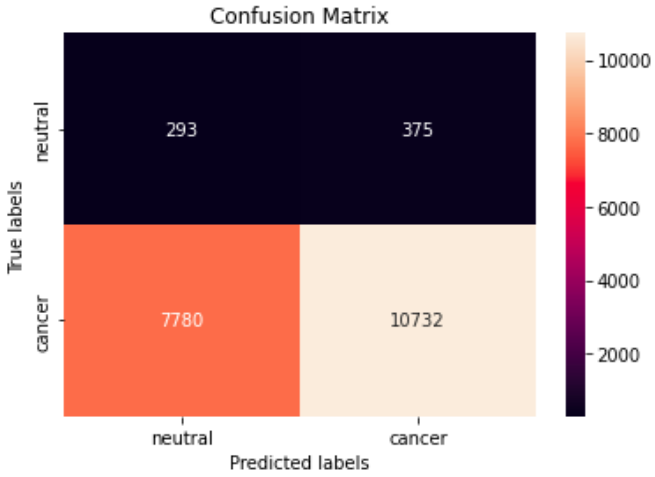
Dobijene vrednosti modela za balansiranu tačnost ne prelaze vrednost veću od 0.52. Dok modeli nisu podlegli uticaju nebalansiranih podataka, mere kvaliteta pokazale su da klasifikacija instanci nije bila uspešna, i pored balansiranja podataka i podešavanja težina instanci.



Slika 5.19: CNB - *undersampling*



Slika 5.20: CNB - kombinovana metoda

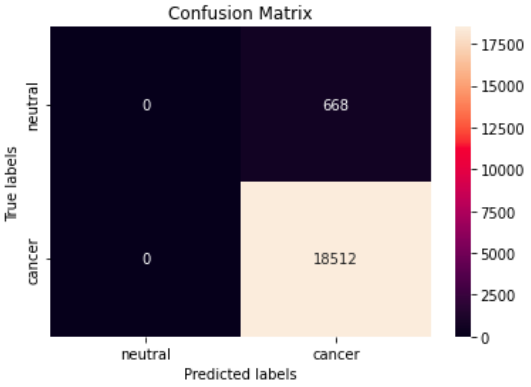


Slika 5.21: CNB - podešene težine instanci

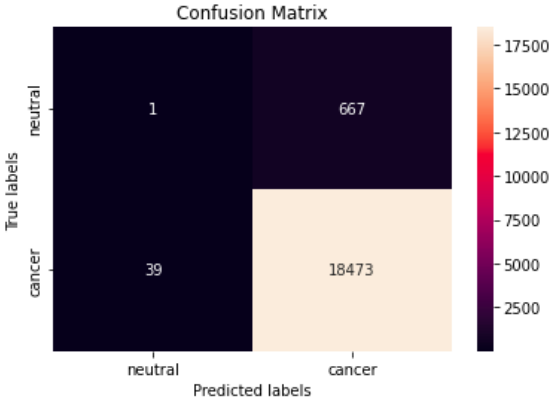
5.2.6 Logistička regresija

Model logističke regresije je, po sličnom pristupu kao i prethodni modeli, treniran sa nebalansiranim (slika 5.22) i balansiranim podacima (*random oversampler* slika 5.23, *random undersampler* slika 5.24, kombinacija *SMOTE* i *random oversampler* slika 5.25), kao i sa podešavanjem težina klasa pri treniranju modela (slika 5.26). Hiperparametar koji predstavlja maksimalan broj iteracija logističke regresije (eng. *max_iter*) je postavljen na `1000`, dok se hiperparametar *class_weight* prilikom treniranja modela koji izračunava težinu klasa postavlja na vrednost `balanced`.

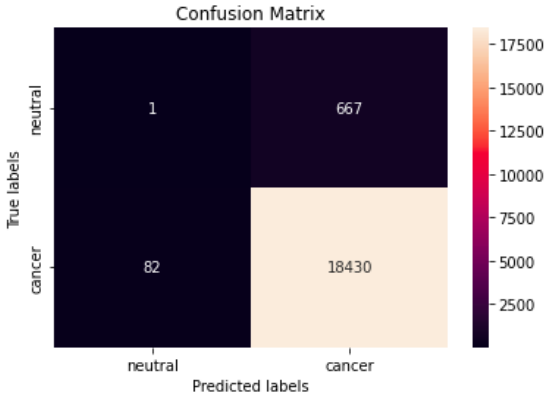
Do preprilagođavanja nije došlo, što se vidi iz podatka da je balansirana tačnost trening skupa približna balansiranoj tačnosti validacionog skupa. Međutim, kako balansirana tačnost modela na test skupu nije veća od vrednosti 0.51, modeli nisu



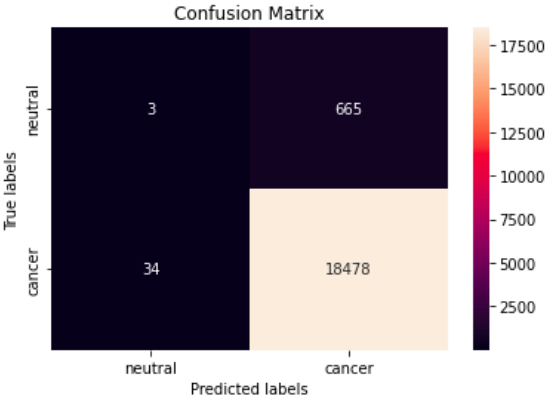
Slika 5.22: LR bez balansiranja



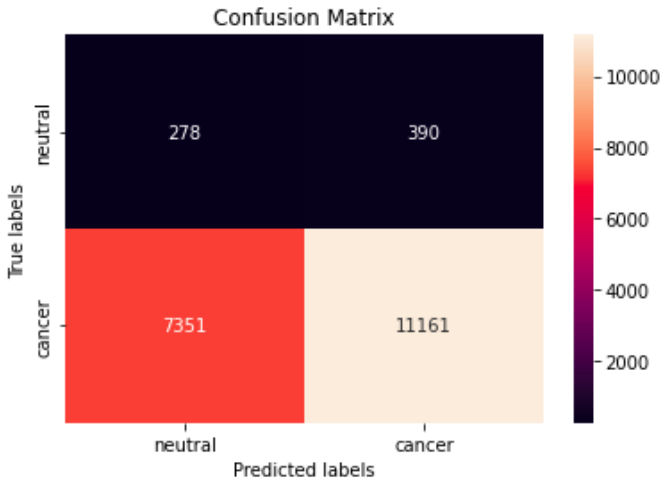
Slika 5.23: LR - *oversampling*



Slika 5.24: LR - *undersampling*



Slika 5.25: LR - kombinovana metoda



Slika 5.26: LR - podešene težine klasa

uspešni u predviđanju klase za prosleđene test instance.

5.2.7 Potpuno povezane neuronske mreže

Zbog velikog broja atributa formirani su modeli potpuno povezanih neuronskih mreža. Jedan model neuronskih mreža je treniran sa nebalansiranim podacima, dok je drugi treniran sa podacima na koje je primenjena kombinacija tehnika za balasiranje, *SMOTE* i *random undersampler*.

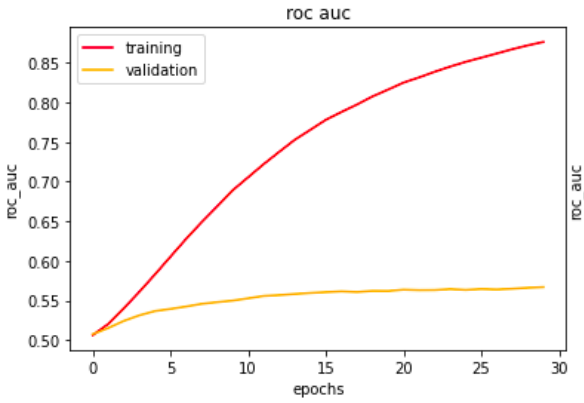
Prilikom formiranja mreža postavljeni su sledeći hiperparametri:

- ulazni sloj: `20` neurona sa ***`relu`*** aktivacionom funkcijom
- srednji sloj: `10` neurona sa ***`relu`*** aktivacionom funkcijom
- izlazni sloj: `1` neuron sa ***`sigmoidnom`*** aktivacionom funkcijom
- optimizator: *Adam* sa `0.0001` vrednošću za parametar učenja, odnosno dužina koraka
- funkcija gubitka: ***`Sigmoid Focal Cross Entropy`***[1]
- broj epoha: `30` - koliko se puta prolazi kroz ceo trening skup
- veličina *batch* skupa: `64` - broj instanci koji se obrađuje pre ažuriranja podataka

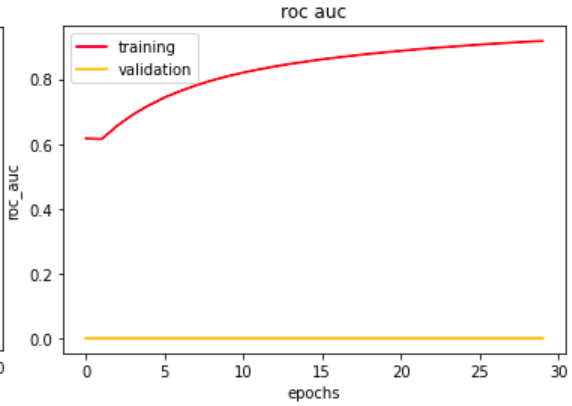
Kao optimizator neuronske mreže koji traži maksimum ili minimum funkcije gubitka izabran je *Adam*. *Adam* koristi ocene prvog i drugog momenta gradijenta kako bi odredio pravac i dužinu koraka kretanja ka optimumu funkcije. Upravo iz tog razloga ovaj optimizator ima sposobnost bržeg kretanja, odnosno većih koraka, u slučaju stabilnog silaska nizbrdicom bez velike promene pravca. Dok, ukoliko je korak prevelik i dolazi do preskakanja optimuma, veličina koraka se smanjuje i vrši se usporeno kretanje nizbrdo [11].

Prilikom posmatranja procesa validacije modela otkriveno je da se model brzo preprilagođava prosleđenim podacima. Povećanjem broja neurona po sloju mreža, broja epoha ili veličine *batch* skupa balansirana tačnost trening skupa naglo raste dok se promene ne javljaju na validacionom skupu. Posle prvih nekoliko epoha, čak i sa jednostavnom strukturom mreže preprilagođavanje se ne može izbeći. Prikazano je

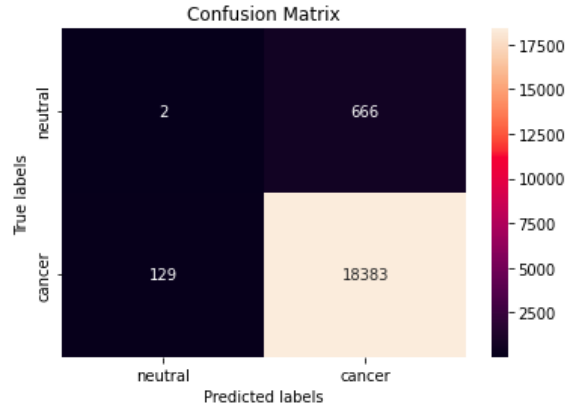
[1] Funkcija gubitka koje se često koristi u slučajevima kada postoji velika neuravnoteženost klasa među podacima. Fokusira se na podatke malobrojnije klase koje je teško klasifikovati, odnosno za njih je greška veća u odnosu na grešku pri klasifikaciji podataka iz većinske klase.



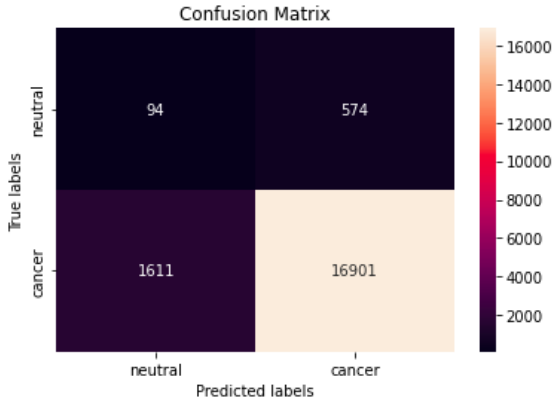
Slika 5.27: Validacioni i trening skup za vrednost AUC - nebalansirani podaci



Slika 5.28: Validacioni i trening skup za vrednost AUC - kombinacija metoda za balansiranje



Slika 5.29: Matrica konfuzije bez balansiranja

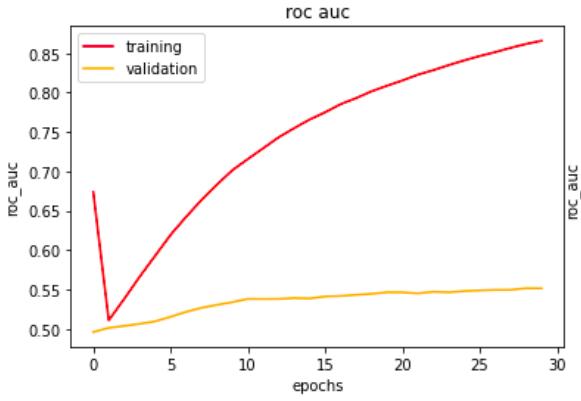


Slika 5.30: Matrica konfuzije sa balansiranjem

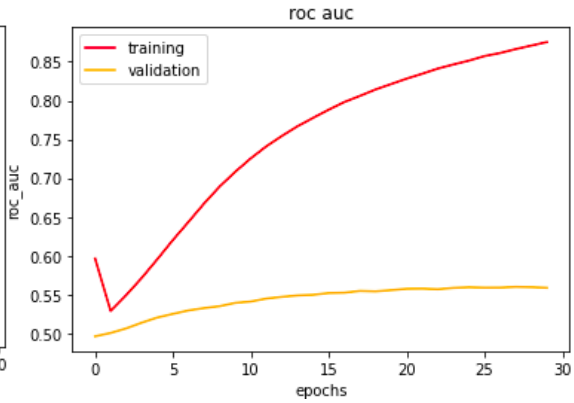
ponašanje modela koji dostiže svoj maksimum balansirane tačnosti na validacionom skupu, ali se vidi i njen veliki skok na trening skupu (slike 5.27 i 5.28).

Rezultati prethodnih modela su prikazani preko matrica konfuzije (slike 5.29 i 5.30). Korišćenje tehnika za balansiranje podataka nije doprinelo stvaranju boljeg modela. Iako je prisutan veći broj tačno klasifikovanih neutralnih instanci, značajno je veći broj štetnih instanci koje su pogrešno klasifikovane.

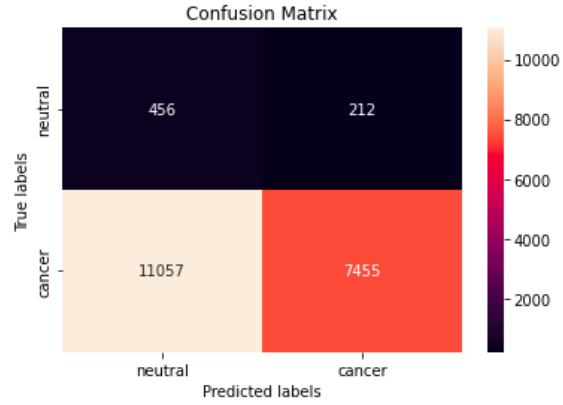
Treniranje neuronskih mreža je urađeno i sa podešavanjem težina klasa (slika 5.33), kao i sa podešavanjem težina instanci (slika 5.34). Za klase, težina se nalazi uz pomoć metode *compute_class_weight*, dok se težina instanci nalazi metodom *compute_sample_weight*. Oba modela na validacionom skupu ukazuju na rano preprilagođavanje, u prvim epohama učenja. Iz tog razloga, izabrani hiperparametri su



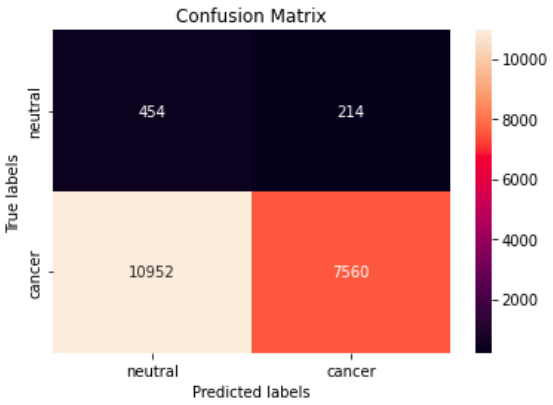
Slika 5.31: Validacioni i trening skup za vrednost AUC - podešavanje težina klasa



Slika 5.32: Validacioni i trening skup za vrednost AUC - podešavanje težina instanci



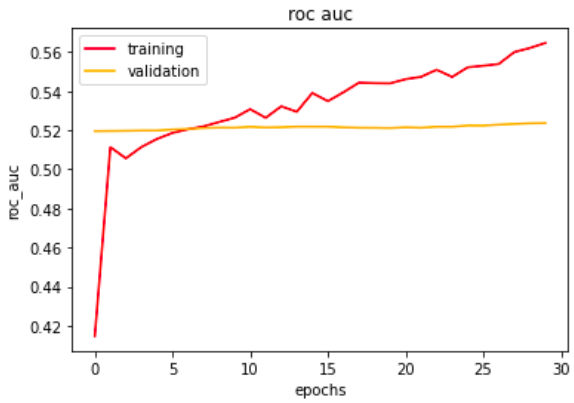
Slika 5.33: Sa težinama klasa



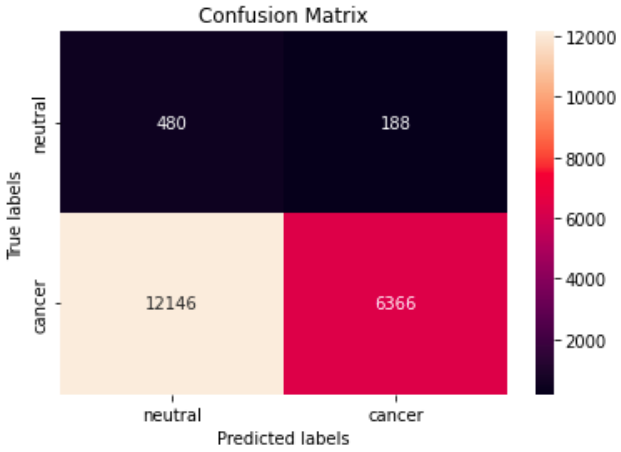
Slika 5.34: Sa težinama instanci

istih vrednosti kao i kod prethodnog modela. Zbog vrednosti balansirane tačnosti modela koja ne prelazi vrednost 0.5567 i preprilagođavanja u ranim fazama, modeli nisu korisni.

Ispitan je još jedan pokušaj balansiranja podataka upotrebom generatora koji balansira *batch* skup pri učenju neuronske mreže korišćenjem *random oversampler* metode. Zastupljenost klasa u svakom *batch* skupu je u ovom slučaju približno ista. Balansiranje *batch* podataka obavlja funkcija *make_generator* kojoj se prosleđuju trening podaci i veličina *batch* skupa, a koja je dostupna preko biblioteke *keras_balanced_batch_generator*. Izabrani hiperparametri ne utiču na ponašanje modela zato što preprilagođavanje nastaje u početnim fazama učenja i nije ga moguće izbeći (slika 5.35):



Slika 5.35: Validacioni i trening skup za vrednost AUC - generator



Slika 5.36: Sa generisanjem *batch*-a

- ulazni sloj: 20 neurona sa `relu` aktivacionom funkcijom
- srednji sloj: 10 neurona sa `relu` aktivacionom funkcijom
- izlazni sloj: 1 neuron sa `sigmoidnom` aktivacionom funkcijom
- optimizator: *Adam* sa 0.0001 vrednošću kao parametar učenja
- funkcija gubitka: `Sigmoid Focal Cross Entropy`
- veličina *batch* skupa prosleđen funkciji *make_generator*: 2000
- broj epoha: 30

Iz dobijenih rezultata koji su prikazani matricom konfuzije (slika 5.36) se očitavaju mere približne vrednosti kao i kod prethodno ispitanih modela, odnosno prisutne su niske vrednostima balansirane tačnosti od 0.531 na test skupu.

5.2.8 Ansambl neuronskih mreža

Ispitivanje pojedinačnih modela sa različitim pristupima za balansiranje podataka nije donelo dobre rezultate. Stoga, pokušano je sa kombinovanjem ansambla i drugih modela. Na prikazan klasifikacioni problem su primenjena dva modela: model sastavljen od više potpuno povezanih neuronskih mreža i model sastavljen od više slučajnih šuma sa različitim skupovima za treniranje.

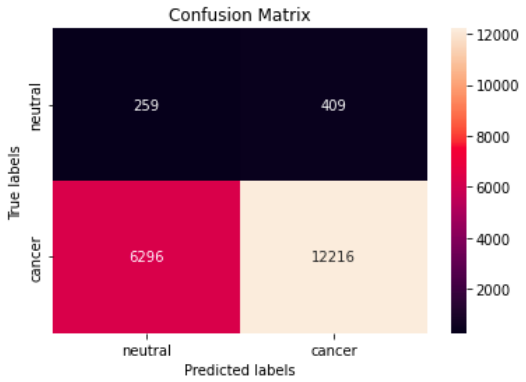
Balansiranje podataka se izvršava primenom dve tehnike. Prva tehnika je *CCMUT* (eng. *cluster centroid based majority under-sampling technique*) koja izbacuje instance većinske klase u redosledu njihove udaljenosti od centroida klase. Implementacija funkcije je prikazana na slici 5.37. Kao parametar se ovoj metodi prosleđuje procenat podataka koji želimo da izbacimo, u ovom slučaju izabrano je 5%. Procenat se određuje sa ciljem da bude što manji, da bi se izbeglo bespotrebno odbacivanje podataka, ali da ukloni najudaljenije instance [29]. Posle primene CCMUT funkcije, primenjujemo *random oversampler* metodu sa parametrom 0.1.

```
from math import sqrt
def CCMUT(X,f):
    cluster_centroid = np.sum(X,axis=0)/X.shape[0]
    euclidean = [None]*X.shape[0]
    for i in range(0,X.shape[0]):
        euclidean[i] = sqrt(sum((cluster_centroid-X[i])**2))
    indices = list(reversed(sorted(range(len(euclidean)), key = lambda j: euclidean[j])))
    X_f = np.delete(X, indices[:int(f/100*X.shape[0])], axis=0)
    return X_f
```

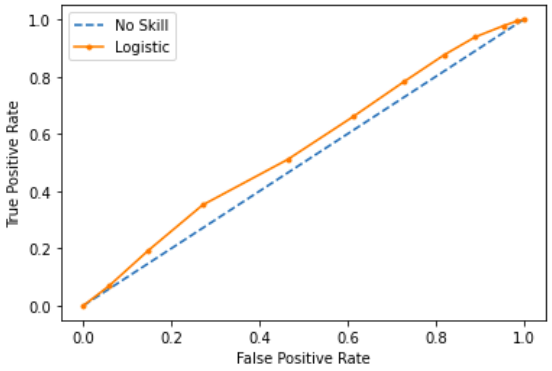
Slika 5.37: CCMUT metoda za balansiranje podataka

Nakon upotrebe metoda za balansiranje, razlika u zastupljenosti dve klase je smanjena, ali i dalje prisutna. Iz tog razloga vršimo implementiranje ansambla sastavljenog od 10 neuronskih mreža. Svaka mreža uzima sve neutralne instance i jedan deo (od ukupno deset) štetnih instanci trening skupa. Sve štetne instance trening skupa su pre podele izmešane.

Predikcija ansambla zavisi od pojedinačnih predikcija neuronskih mreža. Svaka mreža predviđa klasu instance, odnosno daje svoj glas. Svi glasovi se potom sabiraju i dele sa brojem neuronskih mreža u ansamblu. Za instancu se dobija srednja



Slika 5.38: Matrica konfuzije ansambla neuronskih mreža



Slika 5.39: ROC ansambla neuronskih mreža

vrednost glasova. Ako je pomenuta vrednost veća od 0.5, datu instancu posmatramo kao štetnu, a ako je manja, posmatramo je kao neutralnu.

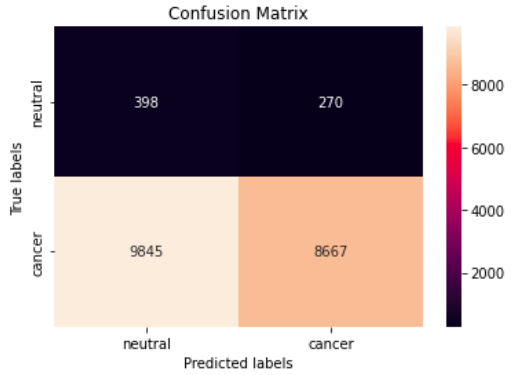
Arhitektura potpuno povezanih neuronskih mreža se sastoji od tri sloja. Ulazni sloj čine neuroni sa ulaznom dimenzijom `300` (broj atributa) i ***`relu`*** aktivacionom funkcijom, koja je aktivaciona funkcija i skrivenog sloja. Izlazni sloj ima jedan neuron i ***`sigmoidnu`*** aktivacionu funkciju. Izabrana funkcija gubitka je binarna unakrsna entropija.

Kao i za prethodne modele, validacija je izvršena za broj neurona po sloju, broj epoha i veličinu *batch* skupa. Izabrane vrednosti su:

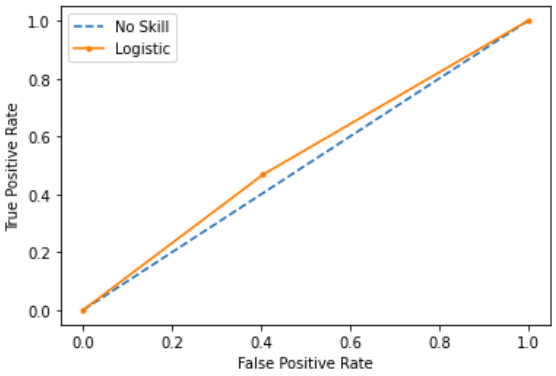
- ulazni sloj : `20`
- skriveni sloj : `10`
- broj epoha : `30`
- veličina *batch* skupa : `64`

Kako i kod prethodnih modela, uočeno je brzo preprilagođavanje modela prosleđenim podacima. Balansirana tačnost trening skupa brzo raste u odnosu na njenu vrednost prikazanu na validacionom skupu. I pored izabrane jednostavne strukture mreža pomenuti problem je i dalje prisutan.

Za model ansambla prikazana je matrica konfuzije (slika 5.38) i izgled ROC krive (slika 5.39), dok je dobijena vrednost balansirane tačnosti 0.524. Prikazane vrednosti ukazuju da kombinacija ansambla i neuronskih mreža nije doprinela stvaranju pouzdanog modela za razlikovanje prisutnih klasa.



Slika 5.40: Matrica konfuzije ansambla slučajnih šuma



Slika 5.41: ROC ansambla slučajnih šuma

5.2.9 Ansambl slučajnih šuma

Model sačinjen od više modela slučajnih šuma se formira na sličan način kao i u slučaju pomenutog ansambla neuronskih mreža. Jedina razlika je što se umesto neuronskih mreža koriste modeli slučajnih šuma, dok je proces određivanja predviđene klase isti. Glasanjem pojedinačnih modela slučajnih šuma dobijamo predikcije za prosleđene instance.

Prilikom formiranja modela hiperparametar *random_state* je postavljen na `0`, dok su vrednosti hiperparametara koji označavaju maksimalnu dubinu stabla i broj stabala odlučivanja ispitani u odnosu na validacioni skup. Niske vrednosti hiperparametara su izabrane kako bi se izbeglo preprilagođavanje modela. Izabrane vrednosti hiperparametara su:

- broj stabala : `5`;
- maksimalna dubina stabla : `3`.

Matrica konfuzije (slika 5.40), kao i izgled ROC krive (slika 5.41) pokazuju da model nepouzdano klasifikuje instance sa balansiranom tačnošću koja ne prelazi vrednost od 0.532.

5.3 Tumačenje

Izrada modela u ovom radu za cilj ima razvoj klasifikatora koji bi sa što većom pouzdanošću uspeo da prepozna razliku između štetnih i neutralnih aminokiselinskih supstitucija izazvanih *missense* mutacijama. Svi modeli u okviru *CancerMut*

alata su dali rezultate sa približno istim ishodom. Rezultati modela se analiziraju sa predznanjem o prisutnoj velikoj nebalansiranosti podataka. Stoga, posmatranjem relevantnih mere kvaliteta, balansirane tačnosti i matrica konfuzije, za sve ove modele se mogu naći rezultati približne vrednosti od 0.5 (tabela 5.1). Dobijene vrednosti ukazuju na modele koji ne klasifikuju različite klase na pouzdan način.

	Balansirana tačnost modela na test skupu
Slučajne šume sa *oversampling* tehnikom i podešavanjem težina klasa	0.532
Balansirani *bagging* model sa undersampling tehnikom	0.515
XGBoost model	0.532
Komplementarni naivni Bajesov model sa kombinacijom tehnika za balansiranje	0.517
Logistička regresija sa *oversampling* tehnikom nad podacima i podešavanjem težina klasa	0.509
Potpuno povezane neuronske mreže sa generatorom	0.531
Ansambl neuronskih mreža	0.524
Ansambl slučajne šume	0.514

Tabela 5.1: Balansirana tačnost modela na test skupu

Ispitivanjem korelacija ciljne promenljive sa svim atributima koji opisuju instancu se otkriva da su njihove vrednosti izrazito male. Maksimalna korelacija ciljne promenljive sa atributima je 0.0185, dok je najveća obrnuta korelacija -0.0092. Osim prethodno ispitanih korelacija, uočeno je preprilagođavanje modela u ranim fazama učenja tokom procesa validacije. Navedene vrednosti i rezultati ukazuju na modele koji iz postojećih podataka ne mogu izvući dovoljno potrebnih informacija za razlikovanje prosleđenih klasa. Iz tog razloga, dobijaju se modeli koji rade približno kao i modeli slučanog izbora. Odsustvo informacije može biti posledica velike nebalansiranosti podataka, kao i atributa koji nisu dovoljno informativni. Kako se u radu

posmatraju štetne i neutralne mutacije na jednoj poziciji, vrednosti atributa koji ih opisuju mogu imati približne vrednosti. Iz tog razloga, moguće je da pomenuti atributi nisu adekvati za uočavanje razlike između instanci iz različitih klasa.

5.4 Poređenje sa drugim alatima

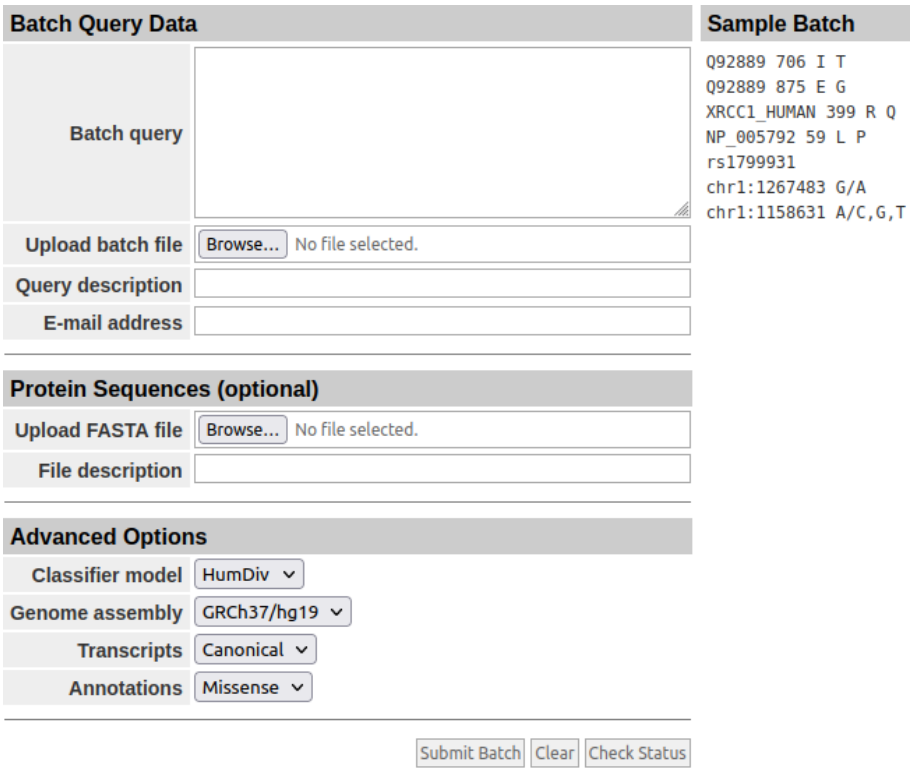
U sledećem poglavlju biće prikazi način slanja podataka *state-of-art* alatima i ocena njihovih rezultata za prosleđen test skup mutacija. Od alata koristiće se *PolyPhen2* i *SIFT*.

5.4.1 Poređenje sa alatom *PolyPhen2*

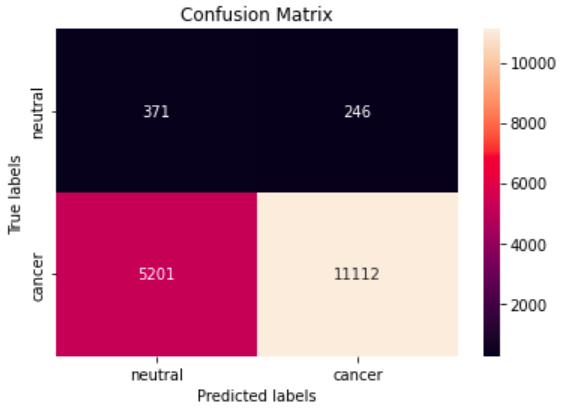
Test podaci izdvojeni iz baze podataka prosleđuju se alatu *PolyPhen2* dostupnom na njegovoj veb stranici. Podatke je potrebno uneti u formu (slika 5.42) čijom obradom se dobija rezultat u obliku tabele sa kolonom koja predstavlja predviđenu verovatnoću sa kojom instanca pripada klasi. Ako je pomenuta vrednost veća od 0.5, instanca je štetna, dok u suprotnom instancu posmatramo kao neutralnu. Zbog ispitivanja mera kvaliteta alata izbačene su mutacije sa nepoznatom verovatnoćom, odnosno mutacije koje umesto tražene verovatnoće imaju oznaku `?` (mutacija *G56D* kod gena *ARID1B* i mutacija *A22V* kod gena *PIM1*).

Dobijena je balansirana tačnost od 0.641, a matrica konfuzije i ROC kriva su prikazane redom na slikama 5.43 i 5.44. Vrednosti balansirane tačnosti i površine ispod ROC krive ukazuju na bolju rezultate u razlikovanju klasa od modela razvijenih u okviru *CancerMut* alata. Alat *PolyPhen2* je uspešniji za prosleđene podatke, i pored toga što vrednosti mera kvaliteta ne opisuju model koji sa visokom preciznošću zna da predvidi prirodu mutacije.

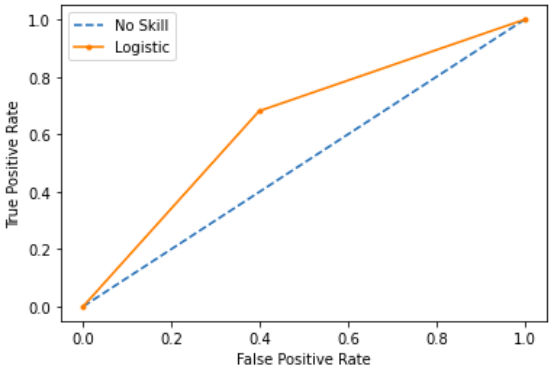
Posmatrajući mutacije grupisane po genima koje alat pogrešno klasifikuje, nije uočena ni jedna specifična karakteristika, odnosno udeo gena sa pogrešno klasifikovanim mutacijama je u korelaciji sa udelom datog gena u prosleđenom skupu podataka. Potrebno je pomenuti da alat *PolyPhen2* za 2 250 mutacija vraća grešku, od toga 51 neutralnih i 2 199 štetnih instanci. Gen *KMT2D* koji je u velikom broju prisutan u prosleđenom skupu u većini slučajeva sadrži mutacije za koje alat *PolyPhen2* proizvodi grešku.



Slika 5.42: Alat PolyPhen2 - forma za prosleđivanje podataka



Slika 5.43: Alat PolyPhen2 - matrica konfuzije



Slika 5.44: Alat PolyPhen2 - ROC kriva

5.4.2 Poređenje sa alatom *SIFT*

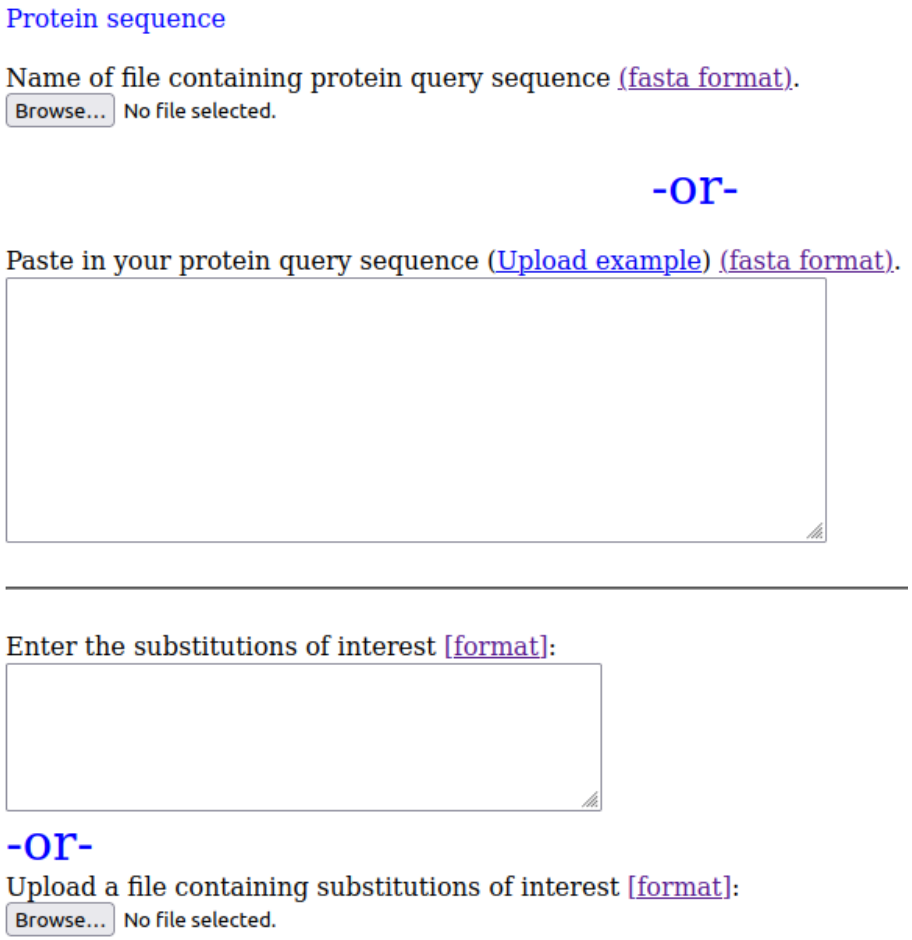
Prosleđivanje test skupa alatu *SIFT* se radi preko forme sa veb stranice alata (slika 5.45). Potrebno je proslediti posmatrane sekvence i spisak njihovih mutacija. To je moguće uraditi na dva načina: direktnim unosom potrebnih podataka ili slanjem datoteka odgovarajućeg formata. Rezultati alata se dobijaju u obliku datoteke sa tekstom koji pruža informacije koja mutacija je testirana i koja je predviđena vrednost alata. Ako se pomenuta vrednost nalazi u intervalu od 0.0 do 0.05 mutacija se posmatra kao štetna, dok ako se vrednost nalazi u intervalu od 0.05 do 1.0 mutacija se svrstava u neutralne.

Prilikom korišćenja alata *SIFT* za neke gene rezultati nisu bili dostupni ni posle 24 časa od njihovog slanja (*FAT1, NOTCH1, FAT4, KMT2D, CTCF, ZFHX3, LRP1B, NOTCH2, ATRX, COL2A1*). Iz tog razloga, ocenjivanje alata je izvršeno bez njih. Takođe, nisu korišćeni ni geni koji su izazvali grešku usled malog broja raznovrsnih sekvenci u bazi koju alat koristi, odnosno za koje alat nije mogao da proizvede rezultat (*FANCD2, FANCG, TSC1, PALB2, MEN1, FANCA*). Posle analiziranja dobijenih rezultata primećen je nedostatak od 817 mutacija, koje su bile prosleđene, ali ih alat nije obradio. Prilikom određivanja ocene alata ni ove mutacije nisu bile korišćene.

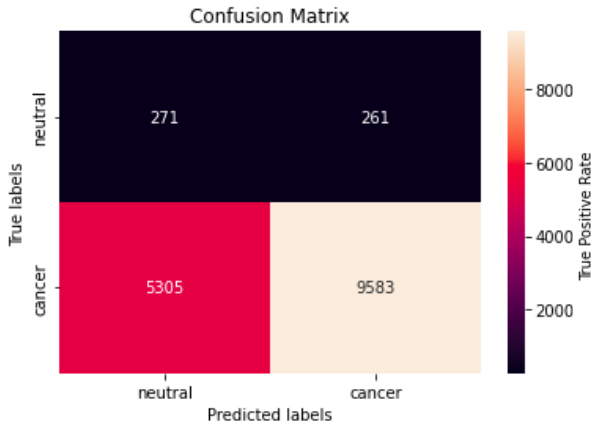
Balansirana tačnost alata *SIFT* je 0.577, a matrica konfuzije i ROC kriva su prikazane redom na slikama 5.46 i 5.47. Rezultati alata su bolji od formiranih modela u ovom radu, ali sam alat ne daje značajno veću pouzdanost. Iz balansirane tačnosti se vidi da alat uočava razliku između klasa, ali radi malo bolje od modela slučajnog izbora. Prilikom posmatranja pogrešno klasifikovanih mutacija uočeno je da su to greške usled malog broja raznovrsnih sekvenci u bazi alata.

5.5 Aplikacija *CancerMut*

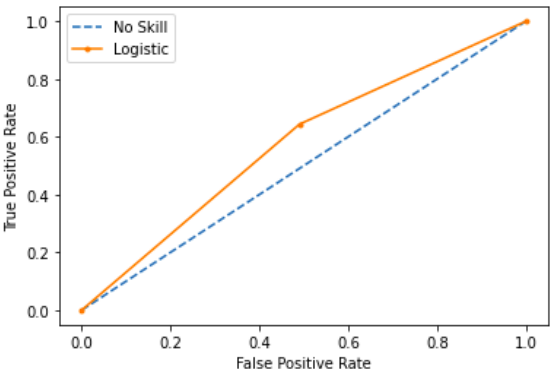
Aplikacija *CancerMut* sadrži formirane modele mašinskog učenja, pomenute u prethodno prikazanoj tabeli, za predviđanje prirode prosleđenih *missense* mutacija. Za pokretanje aplikacije je potrebno imati instaliran programski jezik *Python3* sa svim dodatnim modulima (kao što su *NumPy, Pandas, Tensorflow i Tkinder*), foldere sa uskladištenim modelima i datoteke koji čuvaju informaciju o parametrima za skaliranje i upotrebu algoritma PCA. Kod aplikacije, potrebne datoteke i korišćena baza podataka za učenje modela će biti otvoreni i javno dostupni na *gitlab* repozitorijumu: https://gitlab.com/AleksandraJovicic/materradmatf.



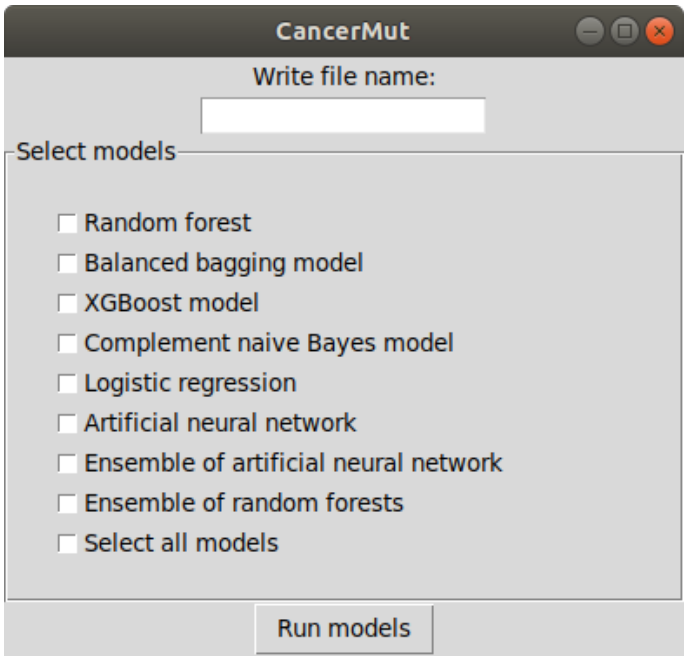
Slika 5.45: Alat SIFT - forma za prosleđivanje podataka



Slika 5.46: Alat SIFT - matrica konfuzije



Slika 5.47: Alat SIFT - ROC kriva



Slika 5.48: Aplikacija *CancerMut*

	Gene name	Variation	Random forest prediction	Random forest probability prediction	Balanced bagging prediction	Balanced bagging probability prediction
0	MTOR	M1083V	0	0.477055097804941	0	0.3
1	MTOR	A1134V	1	0.507831608624949	1	0.6
2	AKT1	V167A	0	0.477055097804941	1	0.6
3	MAP2K1	A366G	0	0.477055097804941	0	0.5
4	MAP2K1	G392S	0	0.477055097804941	1	0.6
5	MAP2K1	M274L	0	0.477055097804941	1	0.6
6	MAP2K1	V393I	0	0.477055097804941	0	0.3
7	PMS2	P470S	0	0.491303639440709	0	0.4

Slika 5.49: Deo *Results.csv* datoteke

Prilikom pokretanja aplikacije dolazi do učitavanja modela i parametara. Grafički korisnički interfejs aplikacije je prikazan na slici 5.48. Prozor aplikacije sadrži polje za unos teksta gde se navodi naziv datoteke sa mutacijama. Datoteka sadrži kolone koje predstavljaju ime gena, mutaciju i atribute dobijene preko alata *EpiMut*. Takođe, data je mogućnost izbora modela koji će vršiti predviđanje. Pritiskom na dugme *Run models* generiše sa nova datoteka sa rešenjima, *Results.csv* (slika 5.49). Pomenuta datoteka pored kolona koje označavaju ime gena i mutaciju, za svaki izabrani model ima po još dve kolone. U jednoj koloni su rezultati dati preko vrednosti *0* (u slučaju predviđene neutralne mutacije) i *1* (u slučaju predviđene štetne mutacije), dok druga kolona sadrži vrednosti predviđene verovatnoće sa kojom mutacija pripada štetnoj klasi.

Glava 6

Zaključak

U ovom radu je opisan razvoj alata *CancerMut* za detekciju aminokiselinskih supstitucija u sekvencama proteina koje učestvuju u nastanak kancera primenom metoda mašinskog učenja. Prikazani su postupci prikupljanja i filtriranja podataka, generisanje atributa modela upotrebom alata *EpiMut*, kao i formiranje i evaluiranje modela. Implementacija pomenutih procesa je izvršena u *Python* programskom jeziku sa odgovarajućom vizualizacijom podataka. Međutim, dobijeni su nezadovoljavajući rezultati zato što modeli nisu uspeli da uoče razliku između supstitucija iz različitih klasa.

Alati *PolyPhen2* i *SIFT* daju bolje rezultate od opisanih modela. I pored činjenice da alati ne daju veliku preciznost, način na koji se uočava razlika između klasa, zajedno sa izborom podataka koji učestvuju u formiranju alata doprinose obrazovanju robusnijeg modela.

Postoji značajan prostor za usavršavanje razvijenog alata. Prikupljanje novih podataka, odnosno obogaćivanje baze podataka može doprineti dobijanju boljih rezultata. Dodavanjem novih atributa ili korišćenjem drugačijih atributa za učenje modela mogli bi uspostaviti veću razliku između štetnih i neutralnih instanci. Moguće je isprobati druge klasifikacione modele mašinskog učenja i eksperimentisati sa njihovim parametrima. Upotrebom ovih metoda moguće je doći do unapređenog alata za razlikovanje supstitucija.

Bibliografija

[1] World Health Organisation. Cancer, 2022. na stranici: https://www.who.int/en/news-room/fact-sheets/detail/cancer.

[2] The American Cancer Society medical and editorial content team. Family Cancer Syndromes, 2020. na stranici: https://www.cancer.org/cancer/cancer-causes/genetics/family-cancer-syndromes.html.

[3] Vukosava Diklić, Marija Kosanović, Smiljka Dukić, and Jovanka Nikoliš. *Biologija sa humanom genetikom*. GRAFOPAN, 1997.

[4] PDQ Cancer Information Summaries. Bethesda (MD):National Cancer Institute (US), 2002-. na stranici: https://www.ncbi.nlm.nih.gov/books/NBK65951/figure/CDR0000044393__5/.

[5] Christopher VanLang. How do I memorize the codons and their corresponding amino acids? na stranici: https://www.quora.com/How-do-I-memorize-the-codons-and-their-corresponding-amino-acids/answer/Christopher-VanLang.

[6] Snežana Trifunović. Proteini, 2019. na stranici: https://www.bionet-skola.com/w/Proteini.

[7] Igor Balać, Branko Bugarski, Irena Ćosić, Miroslav Dramićanin, Drago Đorđević, Nenad Filipović, Nenad Ignjatović, Đorđe Janaćković, Miloš Kojić, Verica Manojlović, Zoran Marković, Bojana Obradović, Ivana Pajić Lijaković, Miodrag Pavlović, Milenko Plavšić, Vladimir Ranković, Boban Stojanović, Vladimir Trajković, Dragan Uskoković, Petar Uskoković, Dejan Veljković, Ivo Vlastelica, and Gordana Vunjak Novaković. *Biomaterijali*. Institut tehničkih nauka Srpske akademije nauka i umetnosti, 2010.

[8] Snežana Trifunović. Genske mutacije, 2018. na stranici: https://www.bionet-skola.com/w/Genske_mutacije.

[9] Julia R. Pon and Marco A. Marra. Expected computation time for Hamiltonian path problem. *Annual Review of Pathology: Mechanisms of Disease*, 10:25–50, 2015.

[10] Smith R. Greenman C. Stephens, P. Patterns of somatic mutation in human cancer genomes. *Nature 446*, pages 153–158, 2007.

[11] Mladen Nikolić and Anđelka Zečević. Mašinsko učenje. unpublished, 2019.

[12] Ankit Chauhan. Random Forest Classifier and its Hyperparameters. Analytics Vidhya, 2021. na stranici: https://medium.com/analytics-vidhya/random-forest-classifier-and-its-hyperparameters-8467bec755f6.

[13] Jason Brownlee. Bagging and Random Forest for Imbalanced Classification. Machine Learning Mastery, 2020. na stranici: https://machinelearningmastery.com/bagging-and-random-forest-for-imbalanced-classification/.

[14] Alakesh Bora. Complement Naive Bayes (CNB) Algorithm, 2020. na stranici: https://www.geeksforgeeks.org/complement-naive-bayes-cnb-algorithm/.

[15] L. I. Smith. A tutorial on principal components analysis(computer science technical report no. oucs-2002-12). Technical report, Department of Computer Science, University of Otago, New Zealand, 05 2002.

[16] Branislava S. Gemović. *Bioinformatička analiza proteina uključenih u patogenezu mijeloidnih maligniteta.* PhD thesis, Univerzitet u Beogradu Biološki fakultet, 2015.

[17] The Wellcome Sanger Institute. Catalogue of Somatic Mutations in Cancer, 2004. na stranici: https://cancer.sanger.ac.uk/cosmic.

[18] U.S. National Center for Biotechnology Information. Single Nucleotide Polymorphism Database, 1999. na stranici: https://www.ncbi.nlm.nih.gov/snp/.

[19] U.S. National Center for Biotechnology Information. Entrez Help, 2004. na stranici: https://www.ncbi.nlm.nih.gov/books/NBK3837/.

[20] Adzhubei IA, Schmidt S, Peshkin L, Ramensky VE, Gerasimova A, Bork P, Kondrashov AS, and Sunyaev SR. A method and server for predicting damaging missense mutations. *Nature methods*, 7:248—-249, 2010.

[21] Pejaver V, Urresti J, Lugo-Martinez J, Pagel KA, Lin GN, Nam H, Mort M, Cooper DN, Sebat J, Iakoucheva LM, Mooney SD, and Radivojac P. Inferring the molecular and phenotypic impact of amino acid variants with mutpred. *Nature Communications*, 11, 2020.

[22] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2021. Nucleic Acids Research. baza se može naći na stranici: https://www.uniprot.org/docs/humsavar.

[23] The Universal Protein Resource (UniProt), 2003. na stranici: https://www.uniprot.org/.

[24] Gemovic B., Perovic V., Davidovic R., and Veljkovic N. Epimut: Alignment-independent tool for functional annotation of amino acid substitutions in epigenetic factors. Technical report, Institut za nuklearne nauke Vinča, 2021. alat dostupan na stranici: https://www.vin.bg.ac.rs/180/tools/epimut.php.

[25] Shuichi Kawashima, Piotr Pokarowski, Maria Pokarowska, Andrzej Kolinski, Toshiaki Katayama, and Minoru Kanehisa. Aaindex: amino acid index database, progress report 2008. *Nucleic Acids Research*, 36:D202–D205, 11 2007.

[26] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal Of Artificial Intelligence Research*, 16:321–357, 2002.

[27] Jason Brownlee. A Gentle Introduction to Threshold-Moving for Imbalanced Classification. Machine Learning Mastery, 2020. na stranici: https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/.

[28] Jason Brownlee. Classification And Regression Trees for Machine Learning. Machine Learning Mastery, 2016. na stranici: https://machinelearningmastery.com/classification-and-regression-trees-for-machine-learning/.

[29] Navoneel Chakrabarty. Implementation of Cluster Centroid based Majority Under-sampling Technique (CCMUT) in Python. Towards Data Science, 2018.

na stranici: https://towardsdatascience.com/implementation-of-cluster-centroid-based-majority-under-sampling-technique-ccmut-in-python-f006a96ed41c.

Biografija autora

Aleksandra Jovičić (*Beograd, 16. jul 1996.*) je diplomirani informatičar. Pohađala je Osnovnu školu "Vuk Karadžić" u Surčinu od 2003. do 2011. godine i završila kao nosilac Vukove diplome. Zemunsku gimnaziju je upisala 2015. godine i završila 2015. godine sa odličnim uspehom. Godine 2019. završila je Matematički fakultet, Univerziteta u Beogradu, na smeru Informatika.