Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Aleks Huč

Napovedovanje mesta na RNA v interakciji s proteinom

magistrsko delo

študijski program druge stopnje
računalništvo in informatika

Mentor: doc. dr. Tomaž Curk

Ljubljana, 2015

Izjava o avtorstvu magistrskega dela

Spodaj podpisani Aleks Huč, z vpisno številko **63090055**, sem avtor magistrskega dela z naslovom:

Napovedovanje mesta na RNA v interakciji s proteinom

S svojim podpisom zagotavljam, da:

- sem magistrsko delo izdelal samostojno pod mentorstvom doc. dr. Tomaža Curka,
- so elektronska oblika magistrskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko magistrskega dela,
- soglašam z javno objavo elektronske oblike magistrskega dela v zbirki "Dela FRI".

V Ljubljani, 23. junij 2015 Podpis avtorja:

Zahvaljujem se mentorju doc. dr. Tomažu Curku za vodenje in pomoč pri izdelavi magistrske naloge. Zahvalil bi se tudi družini in vsem ostalim, ki so mi v tem času nudili podporo.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
AUC	area under the curve	ploščina pod krivuljo ROC
HMM	hidden Markov model	skriti Markov model
MAE	mean absolute error	povprečna absolutna napaka
MC	Markov chain	Markova veriga
MFE	minimum free energy	princip minimalne proste energije
MLE	maximum likelihood estimation	ocena maksimalnega verjetja
RF	random forest	naključni gozdovi
ROC	receiver operating characteristic	krivulja ROC
SVM	support vector machine	metoda podpornih vektorjev
CLIP	UV cross linking and immunoprecipitation	povezovanje z UV in imunoprecipitacija
CLIPSEQ	CLIP sequencing	visokoprepustni CLIP
CTF	conjoint triad feature	kompromisni trojiški atribut
DNA	deoxyribonucleic acid	deoksiribonukleinska kislina
HITSCLIP	high throughput sequencing CLIP	visokoprepustni CLIP
hnRNP	heterogeneous ribonucleoprotein particle	heterogeni ribonukleoproteinski delec
iCLIP	individual UV cross linking and immunoprecipitation	CLIP posameznega nukleotida
mRNA	messenger RNA	informacijska RNA
ncRNA	non-coding RNA	nekodirajoča RNA
ORF	open reading frame	odprt bralni okvir
PARCLIP	photoactivatable ribonucleoside enhanced CLIP	fotoaktivni ribonukleinski CLIP
RBP	RNA binding protein	protein, ki se veže na RNA
RNA	ribonucleic acid	ribonukleinska kislina
RRM	RNA recognition motif	RNA regijski motiv
RT-PCR	reverse transcription polymerase chain reaction	obratne verižne reakcije transkripcijske polimeraze
UTR	untranslated Region	neprevedena regija

Povzetek

V magistrskem delu smo razvili modele za napovedovanje mest na RNA v interakciji s proteini, pri čemer smo uporabili metodo skritih Markovih modelov. Določili smo reprezentativne atribute in motive, na podlagi katerih smo nato zgradili modele za vsak posamezen eksperiment. Ugotovili smo, da različni proteini uporabljajo veliko skupnih atributov in motivov za prepoznavo mest interakcije z RNA. Modeli z dvema stanjema (prisotnost interakcije) boljše napovedujejo kot pa modeli s tremi stanji (prisotnost in intenziteta interakcije). Združevanje napovedi modelov posameznih eksperimentov ne izboljša zmogljivosti napovedovanja. Združeni modeli imajo dovolj dobro zmogljivost, da nam lahko služijo za sklepanje o relacijah med posameznimi proteini, in sicer njihovem sodelovanju, tekmovanju ali neodvisnosti. Skriti Markovi modeli predstavljajo primerno metodo za napovedovanje mest interakcij.

Ključne besede

skriti Markov model, Markove verige, Viterbijev algoritem, algoritem naprej, algoritem nazaj, aposteriorno dekodiranje, transkripcija, RNA, protein, gen

Abstract

The main goal of the master's thesis has been to develop prediction models for interactions between RNA and proteins. We have chosen hidden Markov models as our method for modelling and predicting interactions. From our initial data we have extracted representative features and motifs, which we used for building separate models for each experiment. Majority of proteins bind to the same or very similar features and motifs. We have compared the predictive accuracy of models build with two (presence of interaction) and three states (presence and intensity of interaction). Results show that models with two states perform better than models with three states. Merging predictions of multiple single experiment models in combined models, does not improve prediction accuracy. However, combined models perform with high accuracy, and can be used to determine the relations between proteins, such as competition, cooperation and independence with other proteins when interacting with RNA. We have presented hidden Markov models as viable method for predicting interactions between RNA and proteins.

Keywords

hidden Markov model, Markov chain, Viterbi algorithm, forward algorithm, backward algorithm, posterior decoding, transcription, RNA, protein, gene

Poglavje 1

Uvod

Interakcije med proteini in ribonukleinsko kislino (angl. ribonucleic acid - RNA) igrajo pomembno vlogo pri uravnavanju izražanja genov oziroma proizvodnji proteinov. Proteini neposredno vplivajo na večino biokemijskih procesov v celici ter posredno na delovanje celotnega organizma, na primer na uravnavanje, stabilizacijo in translacijo RNA [20, 21, 22], zdravje in bolezni [23, 25] ter na širitev virusov [24]. Opisovanje in razumevanje lastnosti zaporedij RNA na mestih, ki vstopajo v interakcijo s proteini, je tako ključnega pomena za razumevanje delovanja celic.

Interakcije med proteini in RNA so trenutno zelo raziskovano področje. Raziskovanje interakcij se je pričelo z razvojem eksperimentalnih metod za določanje mest interakcij. Nato pa je bilo potrebno metode izpopolniti ter izvesti zadostno količino eksperimentov in tako pridobiti dovolj dobrih podatkov za začetek gradnje napovedovalnih modelov in algoritmov. Interakcije lahko predstavimo z različnimi statističnimi in matematičnimi modeli, ki temeljijo na verjetnostnih porazdelitvah ter se jih naučimo z metodami strojnega učenja.

Skriti Markovi modeli (angl. Hidden Markov Model - HMM) se za napovedovanje že uporabljajo na področju bioinformatike, vendar zelo redko na področju napovedovanja interakcij med RNA in proteini. V magistrski nalogi smo se zato odločili, da poskusimo uporabiti HMM za modeliranje in napo-

vedovanje mest interakcij med proteini in RNA. Zgradili smo več različnih HMM, jih primerjali ter vrednotili njihovo zmogljivost napovedovanja mest interakcij.

V uvodu sledijo še motivacija, glavni cilji magistrske naloge ter pregled področja. V drugem poglavju predstavimo podatke, in sicer njihovo teoretično ozadje kot dejansko uporabljeno obliko podatkov. V tretjem poglavju predstavimo teoretično ozadje uporabljenih metod in algoritmov. V četrtem poglavju predstavimo pripravo podatkov in gradnjo posameznih HMM. V petem poglavju predstavimo vrednotenje HMM in rezultate magistrske naloge. V dodatku je predstavljena implementacija orodja za pripravo podatkov, gradnjo ter vrednotenje HMM.

1.1 Motivacija

Pri odkritju kemijske strukture v obliki dvojne vijačnice, imenovane deoksiribonukleinska kislina (angl. Deoxyribonucleic acid - DNA), v zgodnjih 50-tih letih prejšnjega stoletja [49] je postalo jasno, da so dedne informacije v celicah zapisane z zaporedjem štirih različnih nukleotidov. DNA ima dve ključni vlogi: prvič omogoča prenos dednega zapisa v nove celice pri delitvi celic in drugič vsebuje informacije za delovanje ter vzdrževanje celic. DNA vsebuje informacije za izgradnjo proteinov, vendar jih sama ne more proizvesti. Najprej se DNA prevede v novo kemijsko strukturo, imenovano RNA, z uporabo postopka transkripcije. Deli DNA, ki se prevedejo v RNA, se imenujejo geni, ki predstavljajo neposredna navodila za gradnjo proteinov. Iz RNA se nato zgradijo ciljni proteini z uporabo postopka translacije. Tok genskih informacij v celicah se začne v DNA in preko RNA zaključi v proteinih. To zaporedje kemijskih procesov imenujemo centralna dogma molekularne biologije, ki opisuje delovanje in razvoj vseh živih bitij. Interakcije med proteini in RNA predstavljajo enega izmed ključnih delov centralne dogme molekularne biologije. Boljše razumevanje le teh bo veliko pripomoglo k razumevanju delovanja celic in vplivov okolja na njih ter zdravljenju nepravilnega delovanja

celic [8].

Na področju DNA so tudi druge panoge biologije, vključno z raziskavami, v zadnjih desetletjih dosegle velik napredek. Z novim razumevanjem delovanja bioloških organizmov se je razširila uporaba njihovih principov in metod tudi v drugi znanstvenih panogah in seveda tudi v računalništvu. Podobno je tudi računalništvo v zadnjem času doseglo velik napredek in tako omogočilo drugim panogam uporabo vedno naprednejših in kompleksnejših metod. DNA sekvence in ostale informacije o DNA predstavljajo ogromne količine podatkov, ki jih brez uporabe računalnikov težko učinkovito hranimo in obdelujemo, zato je med biologijo in računalništvom nastala nova znanstvena panoga, ki se imenuje bioinformatika. Bioinformatika združuje področja matematike, statistike, informatike ter računalništva z namenom hranjenja in obdelave bioloških podatkov.

Strojno učenje (angl. Machine Learning) se ukvarja z avtomatičnim modeliranjem podatkov, kar se izvaja z uporabo različnih metod in algoritmov. Algoritmi iščejo splošna pravila na množici učnih podatkov in tako lahko odgovarjajo tudi na vprašanja, ki se niso pojavila pri učenju. Med znane metode strojnega učenja spadajo nevronske mreže (angl. Neural Network), metoda podpornih vektorjev (angl. Support Vector Machine - SVM) in HMM. V zadnjem času imamo na voljo ogromne količine podatkov iz različnih akademskih in gospodarskih, ki nam predstavljajo nove izzive, kako in na kakšen način iz danih podatkov potrditi že znane povezave ali pa odkriti čisto nove povezave v podatkih. Eno izmed takih področij predstavlja genetika oziroma bioinformatika.

HMM je metoda strojnega učenja, ki predvideva, da so skrita stanja procesa predstavljena z Markovo verigo (MC - angl. Markov chain). MC predstavlja proces, pri katerem je izbira trenutnega stanja odvisna od zgodovine stanj in ni neodvisna. HMM se uporabljajo na veliko področjih in tudi dobro opisujejo opazovane procese, če so podatki dobro predstavljeni. V bioinformatiki se HMM uporabljajo za prepoznavanje sekundarne strukture proteinov, za poravnavo več zaporedij DNA hkrati in za iskanje genov v zaporedjih

DNA. RNA je predstavljena kot množica dolgih zaporedij in tako predstavlja odlično obliko podatkov za HMM, saj delujejo na dolgih zaporedjih. Zato je naravno, da smo poskusili uporabiti HMM za napovedovanje interakcij med proteini in RNA. Vendar smo tako kot pri vsakem strojnem učenju morali za pravilno delovanje HMM prilagoditi, razširiti in jih združevati.

1.2 Cilji

Za izdelavo magistrske naloge smo si zastavili naslednje cilje in izdelali načrt poteka reševanja teh ciljev:

- Ali zaporedja RNA in nekaj dodatnih atributov določajo motive, ki predstavljajo mesta interakcije?
- Ali lahko s HMM modeliramo in napovedujemo mesta interakcij?
- Ali razširitev HMM iz dveh skritih stanj na več izboljša modeliranje in napovedovanje mest interakcij?
- Ali združevanje HMM posameznih proteinov izboljša napovedovanje mest interakcije posameznega proteina?
- Ali lahko iz združevanja HMM sklepamo kaj o relacijah sodelovanja, tekmovanja in neodvisnosti med proteini, ki se vežejo na RNA?

Izdelavo magistrske naloge smo pričeli z analizo in preoblikovanjem podatkov za delo z HMM. Pri analizi smo poiskali najbolj reprezentativne atribute in motive, ki predstavljajo mesta interakcij. Za vsak podani eksperiment posebej smo poiskali atribute in iz njih smo zgradili HMM, ki smo jih vrednotili. Najprej smo izgradili HMM za posamezne eksperimente z dvema stanjema, nato pa še s tremi ter jih primerjali. Nato smo primerjali napovedi vseh HMM za vsak posamezen eksperiment in združili nekaj tistih, ki najbolje napovedo mesta interakcij posameznega eksperimenta, v skupen model. Skupne modele smo nato vrednotili in iz njih razbrali interakcije med samimi proteini, ki se vežejo na RNA.

1.3 Pregled področja

Bioinformatika je zelo novo področje in se je začela hitro razvijati šele z velikim napredkom v računalništvu in genetiki. Za preučevanje proteinov, ki stopajo v interakcijo z RNA, je bilo najprej potrebno razviti postopke, ki omogočajo eksperimentalno določanje mest vezave. Na začetku so metode lahko napovedale le približno okolico mesta, kjer je prišlo do dejanske vezave. Prav tako so lahko zaznale vezavna mesta le majhne množice proteinov. Problem je bil tudi v tem, da so se metode izvajale *in vitro* in se je pri tem velik del informacij o vezavnih mestih že izgubil. Šele z razvitjem naprednejših *in vivo* metod se je povečalo število proteinov, ki smo jim lahko napovedali mesta vezave, izboljšali sta se tudi kvaliteta ter natančnost določitve mest vezave. Pregled nad eksperimentalnimi metodami za določitev mest vezave ponudi članek [3] in kot trenutno najboljšo skupino metod predstavi metode povezovanja z UV in imunoprecipitacije (angl. Crosslinking and immunoprecipitation - CLIP). Članki [10, 11, 12, 13, 14] definirajo metodo CLIP in njene izvedenke PARCLIP (angl. Photoactivatabl-Ribonucleoside Enhanced CLIP), HITSCLIP (angl. High Throughput Sequencing CLIP) in CLIP-SEQ. Predstavijo tudi rezultate uporabe CLIP metod za eksperimentalno določanje mest interakcije s posameznimi proteini ter interpretacijo mest vezave in njihovega vpliva na delovanje celic. Najnovejšo različico metod CLIP predstavlja metoda individualni CLIP (angl. Individual CLIP - iCLIP), ki pa omogoča napovedovanje mest vezave na nukleotid natančno. Definicijo metode iCLIP in njene prve uporabe pa predstavijo članki [4, 15, 16, 17, 18, 19]. Metode CLIP so tako omogočile natančno eksperimentalno določanje mest vezave in predstavljajo odlično osnovo za višjo nivojsko obdelavo podatkov in gradnjo modelov za napovedovanje mest vezave.

Pregled nad dosedanjimi metodami za napovedovanje mest interakcije in podatkov, uporabljenih pri njih, nam ponudi članek [26]. V članku [27] je predstavljen prvi algoritem za napovedovanje interakcij med RNA in proteini. Podatke so predstavljali z rezultatom CLIP eksperimentalne obdelave sekvenc RNA *S. cerevisiae* [28]. Uporabljena sta bila algoritma naključnih

gozdov (angl. Random Forest - RF) in SVM, ki sta napovedovala, s kakšno verjetnostjo se bo nek protein vezal na tarčno mRNA. Kot vhod so bili podani vektorji z več kot sto atributi, na primer genska ontologija, sekundarna zgradba in lastnosti RNA. Niso pa vsebovali eksperimentalno določenih motivov in drugih lastnosti. Za vrednotenje je bilo uporabljeno prečno preverjanje in v splošnem se je RF odrezal rahlo bolje od SVM. Glavna slabost algoritma je, da potrebuje veliko atributov za dobro delovanje. V članku [29] je predstavljen algoritem catRAPID, ki na podlagi fizično kemičnih lastnostih RNA in proteinov zgradi vektor interakcijskih nagnjenj za vsak par RNA in proteina. Za napovedovanje nato uporabi diskriminatorsko moč med nič in ena. Do sedaj je to edini algoritem, ki napoveduje mesta vezave tako na RNA kot na samem proteinu. Algoritem RPISeq je predstavljen v članku [30] in za predstavitev mest uporablja zgolj zaporedje nukleotidov ter za učenje uporablja algoritma SVM in RF. RNA zaporedje je predstavljeno z normaliziranimi frekvencami n-terk dolžine štiri, zaporedje aminokislin proteina pa je predstavljeno s kompromisnim trojiškim atributom (angl. Conjoint Triad Feature - CTF), ki je definiran v članku [31]. Algoritem ugotovi za zaporedje enega proteina in ene RNA, s kakšno verjetnostjo bo prišlo do vezave med njima. V člankih [32, 33, 34, 35] so predstavljeni še drugi algoritmi, ki pa se vsebinsko ne razlikujejo veliko od že omenjenih.

HMM so bili definirani v šestdesetih letih prejšnjega stoletja v člankih [36, 37, 38] in so se na začetku uporabljali predvsem v matematiki in ekonomiji. V računalništvu so se HMM najprej začeli uporabljati za modeliranje prepoznavanja govora [39]. Članek zelo dobro predstavi do takrat že dodobra izpopolnjene metode HMM in predstavi njihovo uporabo pri modeliranju prepoznave govora. V današnjem času so se HMM začeli uporabljati v bioinformatiki predvsem za poravnavo dveh ali več zaporedij, ki nam omogoča prepoznavo genov in drugih lastnosti DNA ter RNA zaporedij [40, 41, 42]. Knjiga [1] naredi splošni pregled nad HMM ter predstavi njihovo uporabo v bioinformatiki.

Uporaba HMM za napovedovanje mest interakcije med proteini in RNA je

zelo majhna. V članku [7] je predstavljena uporaba HMM za napovedovanje mest vezave proteinov NOVA in MBNL. Za opis mest so uporabili gruče eksperimentalno določenih motivov zaporedja nukleotidov YCAY in YGVY (Y predstavlja nukleotid C ali T). V članku [43] uporabijo nehomogeni HMM, ki določene povezave med stanji preprečuje. Za svoje napovedovanje uporablja zaporedja nukleotidov in mutacije, ki so nastale pri eksperimentih CLIP in so jih pridobili pri predhodni primerjavi zaporedij RNA. Predpostavljajo, da se mutacije pojavljajo manj pogosto na začetkih in koncih vezavnih zaporedij.

Združevanje modelov se pri strojnem učenju v splošnem prevede na večciljni ali večznačni problem, ki ga predstavijo naslednji članki [44, 45, 46, 47, 48]. V članku [6] pa je predstavljena metoda generalizacije kopice, ki je namenjena za združevanje več modelov skupaj v en model. Te metode se uporabljajo na različnih področjih in tudi na področju bioinformatike, vendar ne pogosto pa se za združevanje HMM.

Napovedovanje mest interakcije je trenutno še zelo okrnjeno, zato smo si v ciljih zastavili naravno izpopolnjevanje metod za napovedovanje mest interakcije. V naslednjih poglavjih najprej predstavimo podatke in metode, ki smo jih uporabili. Nato sledi opis gradnje modelov ter vrednotenje rezultatov. V zaključku delo in rezultate povzamemo ter predlagamo nadaljnje delo. V dodatku sledi še podrobnejša predstavitev implementacije.

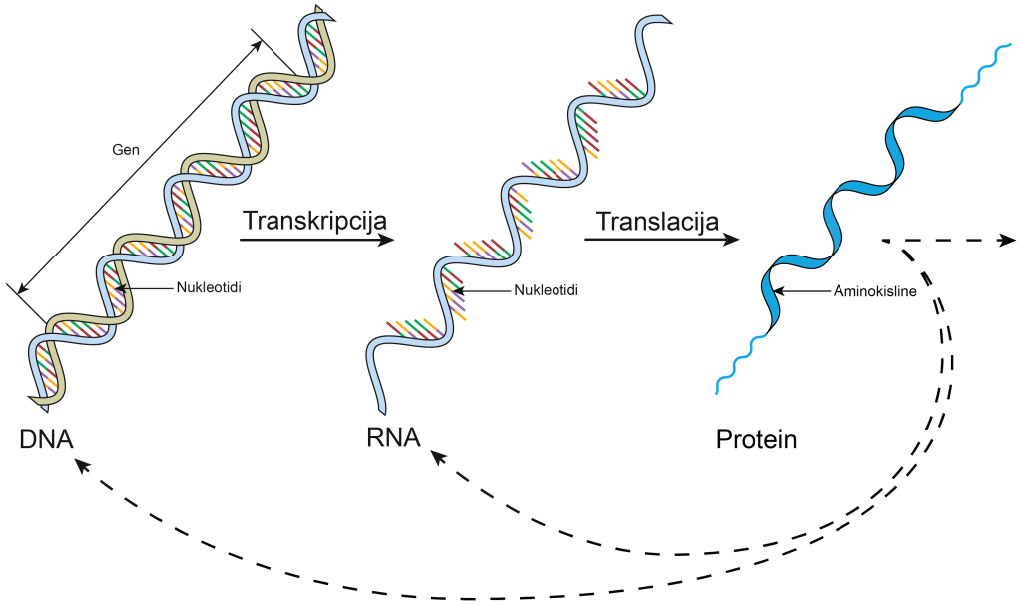
Poglavje 2

Podatki

V bioinformatiki podatkovno domeno predstavljajo podatki, ki opisujejo DNA, njene značilnosti in procese, v katerih sodeluje posredno in tudi neposredno. Pred začetkom gradnje modelov smo se seznanili s podatkovno domeno, da smo prepoznali njihove glavne značilnosti in podrobnosti, da smo nato lahko določili reprezentativne atribute. Pri gradnji našega modela smo uporabili zaporedja nukleotidov, ki predstavljajo gene v prvem kromosomu človeške DNA. Za njih smo imeli že podane informacije o sekundarni strukturi zaporedja ter funkcijske podenote gena, v katere spadajo posamezni deli zaporedja. Za vsak nukleotid zaporedja imamo tudi podatek o interakciji med RNA in proteini, ki je bil pridobljen z eksperimentalnimi metodami za določitev mest interakcij.

2.1 Centralna dogma molekularne biologije

Centralna dogma molekularne biologije poskuša razložiti tok genskih informacij v vseh živih bitjih. Avtor dogme Francis Crick jo je prvič omenil leta 1956 [64]. Dogma torej predstavlja ogrodje za razumevanje prenosa zaporednih informaciji med biopolimeri, ki so nosilci informacij. Poznamo tri glavne razrede biopolimerov: DNA, RNA in proteini. V osnovni obliki informacije prehajajo iz DNA preko transkripcije v RNA in iz RNA preko translacije



Slika 2.1: Centralna dogma molekularne biologije. Povzeto po viru [8].

v proteine (Slika 2.1). Tako kot v osnovnih tudi v ostalih smereh prenosa informacije in uravnavanja genske ekspresije, igrajo interakcije med proteini in RNA ključno vlogo [65, 8].

2.2 DNA

DNA je molekula, ki hrani genetska navodila za razvoj in delovanje vseh znanih živih bitij. DNA je nukleinska kislina, ki skupaj s proteini in ogljikovimi hidrati predstavlja tri glavne makromolekule, ki so nujno potrebne življenje. Večina molekul DNA je sestavljena iz dveh verig, ovitih druga okoli druge. Skupaj tvorita dvojno vijačnico. DNA verigi lahko rečemo tudi polinukleotid, saj je sestavljena iz enostavnejših gradnikov, imenovanih nukleotidi, ki so lahko adenin (A), gvanin (G), citozin (C) in timin (T). Vsak nukleotid je sestavljen iz sladkorja (pentoze), dušikove baze in fosfatne skupine. Sladkorna komponenta se med DNA in RNA razlikuje, in sicer v DNA je prisotna 2'-deoksiriboza, v RNA pa riboza. Na sladkor je preko N-glikozidne vezi vezana

dušikova baza. Ta je lahko adenin (A), timin (T), gvanin (G) in citozin (C). Za podatke smo uporabili zaporedja DNA namesto RNA, saj RNA nastane s postopkom transkripciji, kjer nukleotidi molekule RNA predstavljajo komplement nukleotidov na molekuli DNA z drugim sladkorjem. Pojavi se le ena izjema, in sicer RNA namesto dušikove baze timin (T) uporablja uracil (U), ki pa ima enake kemijske lastnosti kot timin in se pojavi na njegovih mestih. Sladkor in dušikova baza skupaj tvorita nukleozid, če pa je na sladkor vezana še fosfatna skupina, pa dobimo nukleotid. Nukleotidi se povezujejo v poli nukleotide s fosfodiesterskih vezi, natančneje: 3'-hidroksilna skupina enega nukleotida se poveže s 5'-fosfatno skupino naslednjega nukleotida. Dve verigi, ki skupaj tvorita dvojno vijačnico, se vežeta preko dušikovih skupin z vodikovimi vezmi. Adenin se vedno veže s timinom in citozin vedno z gvaninom; temu pravimo tudi Watson-Crickovo pravilo baznih parov [8, 66].

Ker je DNA sestavljen iz nukleotidov, se na eni strani konča s 5'-fosfatno skupino in na drugi strani s 3'-hidroksilno skupino. Krajše lahko zapišemo, da ima DNA 5'-konec in 3'-konec. Smer DNA je pomembna, saj se nukleinska kislina sintetizira *in vivo* le v smeri od 5' proti 3'. Protein polimeraza, ki gradi verige nukleotidov, lahko nov nukleotid pripne le na 3' konec molekule DNA. Da lahko ločimo med verigama DNA, uvedemo pozitivno verigo (+), ki predstavlja verigo v smeri od 5' proti 3', in njej komplementarno negativno verigo (-), ki predstavlja verigo v smeri od 3' proti 5'. Pozitivna veriga DNA ima enako zaporedje kot mRNA, ki se ustvari s transkripcijo negativne verige DNA, le da se namesto timina pojavi uracil (Tabela 2.1).

Smer DNA je poljubna in geni se nahajajo na obeh verigah, zato si pri analizi izberemo eno verigo za pozitivno in drugo za negativno. Zaradi učinkovitosti pri predstavitvi podatkov uporabljamo le eno verigo DNA in gene, predstavljane z lokacijo začetka gena, z lokacijo konca gena in na kateri, pozitivni ali negativni, verigi se nahaja. V našem primeru delamo z zaporedji, ki se berejo iz leve proti desni. Geni na pozitivni verigi so že v pravi smeri, medtem ko gene, ki so na negativni verigi, dobimo iz pozitivne verige, tako da nad zaporedjem nukleotidov uporabimo obratni komplement.

3’ CGCTATAGCGTTT 5’	- DNA	Primer DNA.
5’ GCGATATCGCAAA 3’	+ DNA	Komplement primera DNA.
5’ GCGAUAUCGCAAA 3’	+ mRNA	RNA veriga transkriptirana iz - DNA in je identična + DNA.
3’ CGCUAUAGCGUUU 5’	- mRNA	RNA veriga transkriptirana iz + DNA in je identična - DNA.

Tabela 2.1: Prikaz + in - DNA in njunih mRNA transkriptov.

Komplement nukleotidov naredimo, tako da timin zamenjamo z adeninom in obratno ter gvanin zamenjamo s citozinom in obratno. Komplementu nukleotidov nato še obrnemo vrstni red, tako da je zadnji nukleotid sedaj na prvem mestu, in tako naprej vse nukleotide [8, 67].

Primarna struktura RNA in DNA predstavlja zaporedje nukleotidov, ki so med seboj povezani s fosfodiesterskimi vezmi v verigo. Sekundarna struktura pa predstavlja interakcije med bazami nukleotidov oziroma kako se povežejo različne verige. Pri dvojni vijačnici DNA sta dve verigi vezani med seboj preko vodikovih vezi. Nukleotid na eni verigi se poveže z nukleotidom na drugi verigi tako, da tvori bazni par. Sekundarna struktura DNA je določena z vezavo baznih parov obeh verig nukleotidov, ki se ovijeta druga okoli druge. Sekundarno strukturo RNA predstavlja le ena veriga nukleotidov. Baze tvorijo vezi s komplementarnimi bazami na isti verigi in tako se pojavijo vijačnice, zanke, izbokline in križanja, ki lahko predstavljajo motive za interakcije med proteini in RNA [8, 68, 69].

2.3 Kromosom

Kromosom je organizirana struktura, ki vsebuje večino DNA živega organizma, strukturne proteine, ki se imenujejo tudi histoni, transkripcijske faktorje in druge makromolekule. Manjši del DNA, ki se ne nahaja v kromosomih, pa se nahaja v mitohondrijih, ki se dedujejo od matere. Kroma-

tin je kompleks DNA in proteinov, ki vzdržuje kromosome, ter ga najdemo v celičnem jedru evkariontskih celic. Struktura kromosomov se spreminja med različnimi življenjskimi stopnjami celice in se podreja zahtevam DNA. Število in zgradba kromosomov so značilni za vsako vrsto evkariontskega organizma. Človeške kromosome lahko razdelimo na avtosome in spolne kromosome. Določen genski zapis je povezan s spolom človeka in je dedovan preko spolnih kromosomov. Avtosomi vsebujejo dedni zapis, ki je neodvisen od spola. Vsi kromosomi se obnašajo enako pri delitvi celic. Človek ima 23 parov kromosomov, saj dobimo po en kromosom od vsakega starša, od tega je 22 parov avtosomov in 1 par kromosomov spola. V vsakem celičnem jedru se torej nahaja 46 kromosomov. Prav tako vsaka celica vsebuje več sto kopij mitohondrijske DNA [8].

2.4 Gen

Gen predstavlja zaporedje nukleotidov na DNA ali RNA, ki nosi navodilo za posamezno lastnost organizma ali posamezen protein, ki opravlja določeno funkcijo v živem organizmu. Gene vsak organizem podeduje od svojih prednikov, celica pa podeduje gene od svoje materinske celice med procesom delitve celice, v kateri se izvede replikacija DNA. Geni imajo zelo različne dolžine od nekaj sto pa do par milijonov nukleotidov. Geni so sestavljeni iz več funkcionalnih pod enot (Slika 2.2) [8, 70, 71, 72]:

- **Promotor** - Predstavlja lokacijo, ki jo prepoznajo proteini transkripcije in se vežejo nanje. Ponavadi se nahajajo pred samim genom ali na začetku gena.
- **Ojačevalec** - Predstavlja lokacijo, ki jo prepoznajo proteini transkripcije in se vežejo nanje. Ponavadi se nahajajo znotraj samega gena.
- **Zaključevalec** - Predstavlja lokacijo, ki jo prepoznajo proteini transkripcije in se vežejo nanje. Ponavadi se nahajajo na koncu samega gena ali pa za njim.

- **Intron** - Predstavlja del gena, ki se med transkripcijo obdrži, vendar se pred translacijo s procesom spajanja zavrže. Ne vsebuje informacij o proteinih.
- **Ekson** - Predstavlja del gena, ki se med transkripcijo obdrži in predstavlja vhod translacije. Vsebuje informacijo o proteinih.
- **5'UTR** (angl. 5' Untranslated Region) - Predstavlja začetni del oziroma 5' konec eksona, ki vsebuje tudi kodone za začetek translacije. Ne vsebuje informacij o proteinih, vendar služi kot lokacija, ki jo prepoznajo proteini translacije in se vežejo nanj.
- **ORF** (angl. Open Reading Frame) - Predstavlja del eksona, ponavadi med 5'UTR in 3'UTR delom, ki v kodonih nosi informacijo o aminokislinah, ki sestavljajo posamezen protein.
- **3'UTR** (angl. 3' Untranslated Region) - Predstavlja končni del oziroma 3' konec eksona, ki vsebuje tudi kodone za konec translacije. Ne vsebuje informacij o proteinih, vendar služi kot lokacija, ki jo prepoznajo proteini translacije in se vežejo nanj.
- **ncRNA** (angl. Non-coding RNA) - Predstavlja zaporedje nukleotidov, ki se pojavi med samim zaporedjem nukleotidov gena, vendar funkcijsko ne spada h genu. Ta zaporedja se pri procesu transkripcije ne upoštevajo.

2.5 Protein

Proteini so velike biološke molekule ali makromolekule, ki so sestavljene iz ene ali več verig aminokislin. Proteini opravljajo veliko različnih nalog znotraj živih organizmov, kot so kataliziranje metabolističnih reakcij, repliciranje DNA, odzivanje na dražljaje in prenašanje molekul po organizmu. Proteini se med seboj razlikujejo v zaporedju aminokislin, ki ga narekuje zaporedje nukleotidov v genih. Poznamo dvajset standardnih aminokislin. Kodon je

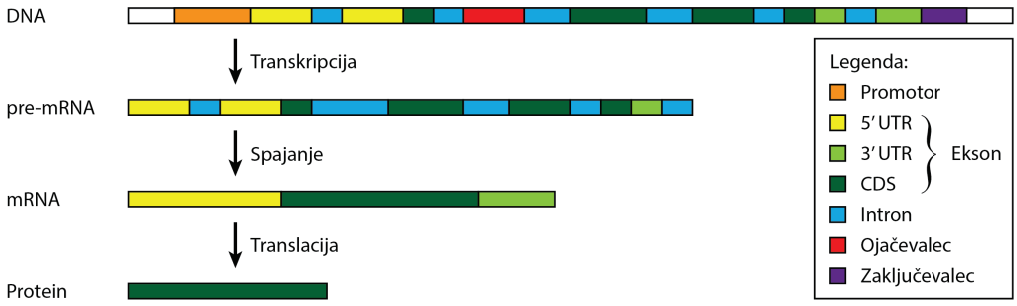
zaporedje treh nukleotidov, ki predstavljajo posamezno aminokislino ali drug kontrolni proces. Zaporedje aminokislin vpliva tudi na obliko proteina v tridimenzionalnem prostoru, ki vpliva na aktivnost proteina. Linearno verigo aminokislin imenujemo polipeptid. Polipeptidi, ki vsebujejo manj kot 20 do 30 aminokislin, le redko predstavljajo prave proteini in se imenujejo peptidi ali oligopeptidi. Vsak protein obstaja le določen čas in je potem degradiran in recikliran s strani same celice [8, 73, 74, 75].

Posebna podvrsto proteinov predstavljajo proteini, ki se vežejo na verige RNA (angl. RNA- Binding Protein - RBP). RBP vsebujejo različne strukturne motive, na primer RNA prepoznavni motivi (angl. RNA Recognition Motif - RRM). Nahajajo se tako v jedru kot v citoplazmi. Zrela RNA se hitro prenese iz jedra v citoplazmo, zato RBP v jedru obstajajo le kot kompleks proteina in pre-mRNA, ki se imenuje heterogeni ribonukleoproteinski delec (angl. Heterogeneous ribonucleoprotein particle - hnRNP). RBP imajo ključno vlogo pri delovanju celice, prenosu in lokalizaciji. Še posebej so pomembni pri upravljanju RNA verig, ki nastanejo pri transkripciji. Procesi, ki jih izvajajo, so: spajanje, poliadenilacije (angl. Polyadenylation) ter stabilizacija, lokalizacija in translacija mRNA. Evkariontske celice uporabljajo približno 900 RBP proteinov z različnimi RNA vezavnimi aktivnostmi in odnosi z drugimi proteini [8, 76, 77, 78, 80].

2.6 Transkripcija

Transkripcija je proces, pri katerem se informacija, shranjena v zaporedju DNA, prevede v novo sintetizirano prenosno RNA (mRNA) (Slika 2.2). Postopek transkripcije se izvede znotraj celičnega jedra. Encimi, ki sodelujejo pri transkripciji, so polimeraza RNA in transkripcijski faktorji.

Transkripcija se prične tako, da se eden ali več transkripcijskih faktorjev veže na holoencim RNA polimeraze, kar mu omogoči vezavo na DNA promotor. RNA polimeraza ustvari traskripcijski mehurček, ki razdre dvojno vijačnico DNA, tako da prekine vodikove vezi med komplementarnimi DNA



Slika 2.2: Prikaz posameznih delov gena pri prehodu iz DNA v protein. Povzeto po viru [8].

nukleotidi. RNA polimeraza nato doda nukleotide RNA, ki so komplementarni eni verigi DNA. Sladkorno-fosfatna hrbtenica RNA se ustvari s pomočjo RNA polimeraze in tako se ustvari RNA veriga. Vodikove vezi neovite vijačnice DNA-RNA se prekinejo in osvobodijo na novo ustvarjeno verigo RNA. Pri celicah z jedrom se RNA lahko še dodatno obdela s postopki spajanja, poliadenilacije in dodajanja 5' konca. RNA lahko ostane v jedru ali pa se prenese v citoplazmo skozi jedrne pore.

V evkariontskih celicah je primarni transkript pre-mRNA, ki mora biti pred translacijo še dodatno preurejena. Iz Pre-mRNA dobimo mRNA, tako da dodamo 5'konec in poli-A konec pre-mRNA in izvedemo proces spajanja. Spajanje je proces, ki izloči introne gena iz pre-mRNA. Lahko se pojavi tudi alternativno spajanje, kar poveča število proteinov, ki lahko nastanejo iz posamezne mRNA. Del DNA, ki se prevede v molekulo RNA, se imenuje transkripcijska enota, ki vključuje vsaj en gen. Rezultat transkripcije je zrela mRNA, ki predstavlja ekson gena. Zrela mRNA predstavlja vhodno informacijo za izdelavo proteina s postopkom translacije ali pa predstavlja nekodirajočo RNA, ribosomsko RNA, prenosno RNA in druge encime RNA [8, 79].

2.7 Podatki, uporabljeni pri gradnji modelov

Podatki, ki jih uporabljamo pri učenju modelov in napovedovanju, izhajajo iz zaporedja nukleotidov. Za vsak nukleotid imamo podano še intenziteto mesta interakcije, v katero funkcijsko podenoto gena spada, ali sodeluje pri vezavah sekundarne strukture in katera n-terka se začne na njegovem mestu.

Pred analizo podatkov, grajenjem modelov in napovedovanjem mest interakcij je bilo potrebno podatke preoblikovati v obliko, ki je primerna za HMM. Podatke smo organizirali kot množico matrik, kjer je bil vsak gen predstavljen s štirimi matrikami:

- Matrika z zaporedjem nukleotidov (dolžina gena x 1).
- Matrika z zaporedjem funkcijskih podenot (dolžina gena x 6).
- Matrika z zaporedjem intenzitet mest interakcij (dolžina gena x 32).
- Matrika z zaporedjem sekundarne strukture zaporedja (dolžina gena x 1)

2.7.1 Nukleotidi

Uporabili smo zaporedje nukleotidov prvega kromosoma človeka, ki je dolg približno 250 milijonov nukleotidov, ki predstavlja 8 % celotnega človeškega genoma. Iz zaporedja prvega kromosoma smo izločili 2,040 zaporedij, ki predstavljajo trenutne poznane gene, s skupno dolžino približno 110 milijonov nukleotidov. Uporabili smo kar zaporedje DNA, saj se od RNA razlikuje le v tem, da se namesto timina pojavlja uracil (Slika 2.3).

2.7.2 Mesta na RNA in intenzitete interakcij

Za vsak nukleotid v zaporedje nukleotidov genov imamo podano prisotnost in intenziteto interakcije za 32 eksperimentov. Prisotnosti in intenzitete interakcij so bile pridobljene z uporabo eksperimentalnih metod CLIPSEQ,

PARCLIP, HITSCLIP in iCLIP. Prve tri metode za posamezno mesto vezave v njegovi okolici podajo interval intenzitet z uniformno ali Gauss-ovo porazdelitvijo. Metoda iCLIP pa napove intenziteto interakcije za vsak nukleotid posebej. Imeli smo podatke za 32 CLIP eksperimentov, kjer je protein lahko predstavljen v enem ali več eksperimentih (Tabela 2.2). Intenziteta je predstavljena z zvezno vrednostjo, kjer 0 predstavlja, da na tem mestu ni prišlo do interakcije. Če pa je intenziteta vezave večja od 0, potem vrednost predstavlja dejansko intenziteto vezave (Slika 2.3).

Postopek metod CLIP se začne z izpostavitvijo transkriptomov v celicah UV svetlobi, ki povzroči *in vivo* tvorjenje kompleksov RNA-protein. UV svetloba povzroči tvorbo kovalentnih vezi med proteini in nukleinskimi kislinami, ki so blizu skupaj. Celice, v katerih je prišlo do vezav, se lizirajo. Protein, ki nas zanima, je izoliran s postopkom imunoprecipitacije. Da lahko pride do od zaporedja odvisne povratne transkripcije, se morajo RNA adapterji ligirati na 3' konec in radioaktivno označeni fosfati priključiti na 5' konec zaporedij RNA. Kompleksi RNA-protein so ločeni od prostih RNA zaporedij z uporabo elektroforeznega gela in prenosa skozi membrane. Proteinaza K povzroči, da se ločijo proteini od kompleksa RNA-protein. Ta korak pusti peptid na mestu vezave, kar omogoči prepoznavo nukleotida na mestu vezave. Po ligiranju RNA povezovalcev na 5' konec RNA se proizvede s postopkom obratne verižne reakcije transkripcijske polimeraze (angl. Reverse Transcription Polymerase Chain Reaction - RT-PCR) komplementarna DNA. Visoko prepustno sekvenciranje je nato uporabljeno, da ustvari branja zaporedja z enoličnimi črtnimi kodami, ki identificirajo zadnji nukleotid komplementarne DNA. Mesta vezave so nato identificirana s postopkom razporejanja prebranih črtnih kod nazaj na transkriptom [3, 4, 17].

2.7.3 Funkcijski deli gena

Za vsak nukleotid v zaporedju nukleotidov genov imamo podano oznako, in sicer; ali spada pod intron, ekson ali ncRNA. Če spada pod ekson, imamo nato še podano dodatno oznako, ali spada pod 3'UTR, ORF ali 5'UTR del

(Sliki 2.3 in 2.2).

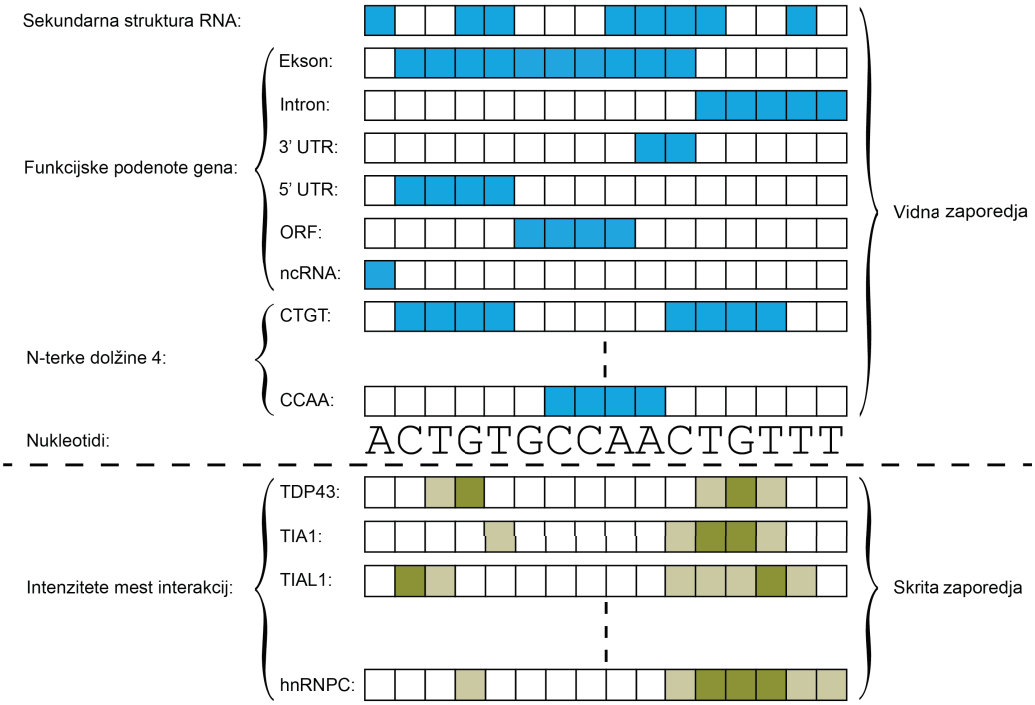
2.7.4 Sekundarna struktura RNA

Za vsak nukleotid v zaporedju nukleotidov genov imamo tudi podano, ali sodeluje pri vezavah sekundarne strukture RNA, saj ta mesta lahko predstavljajo specifične lastnosti RNA, ki jih lahko prepoznajo proteini, ki se vežejo z RNA (Slika 2.3).

Podatki o sekundarni strukturi RNA so bili eksperimentalno pridobljeni z metodo RNAFold, ki izhaja iz principa minimalne proste energije (angl. Minimum Free Energy - MFE). MFE predstavlja drugi zakon termodinamike, ki pravi, da v zaprtem sistemu, ki ima konstante zunanje parametre in entropijo, energija pade na minimalno vrednost, ko se približa ravnotežju. Vključuje implementacijo razdelitvene funkcije za izračun verjetnosti baznih parov in krožno RNA zlaganje [59, 60, 61, 62, 63].

2.7.5 N-terke zaporedij

N-terke predstavlja krajše zaporedje nukleotidov, s katerimi smo predstavili motive, ki jih uporabljajo proteini pri vezavi. Izbrali smo n-terke dolžine štiri, saj se v literaturi najbolj pogosto pojavijo. Skušali smo število možnih emisij v HMM obdržati kar se da nizko. Za vsak nukleotid v zaporedje nukleotidov genov imamo podano n-terko, ki se začne na mestu nukleotida, torej trenutni nukleotid predstavlja prvi nukleotid v n-terki (Slika 2.3).



Slika 2.3: Prikaz podatkov, ki smo jih uporabili pri gradnji HMM za posamezne eksperimente.

Št.	Protein	Metoda	Skupina replik	Interakcije na 1. kromosomu	Interakcije na vseh kromosomih	%	Funkcija
1	Ago/EIF2C1-4	PARCLIP	A	161,775	1,364,096	11.86	Protein potreben za zaviranje
2	Ago2-MNase	PARCLIP	A	460,732	3,013,785	15.29	genov. Veže se na kratke
3	Ago2 (1)	HITSCLIP	A	39,255	432,089	9.09	mRNA in zavira translacijo
4	Ago2 (2)	HITSCLIP	A	39,305	432,139	9.10	mRNA, ki so njim
5	Ago2 (3)	CLIPSEQ	A	398,563	2,766,476	14.41	komplementarne.
6	eIF4AIII (1)	CLIPSEQ	B	5,904,596	63,353,334	9.32	RNA helikaza, ki vpliva na
7	eIF4AIII (2)	CLIPSEQ	B	2,098,789	20,925,715	10.03	translacijo, lokalizacijo in upravljanje mRNA.
8	ELAVL1 (1)	PARCLIP	C	137,054	1,202,570	11.40	Protein, ki se veže v okolici
9	ELAVL1 (2)	CLIPSEQ	C	28,773	223,121	12.90	3' konca mRNA in poveča
10	ELAVL1A	PARCLIP	C	30,298	256,387	11.82	stabilnost mRNA.
11	ELAVL1-MNase	PARCLIP	C	1,007,000	7,940,664	12.68	
12	ESWR1	PARCLIP	D	64,485	543,116	11.87	Protein, ki zavira transkripcijo DNA.
13	FUS	PARCLIP	E	105,662	1,012,411	10.44	Protein, ki povzroči od ATP
14	Mut FUS	PARCLIP	E	47,512	380,345	12.49	neodvisno vezavo posameznih komplementarnih verig DNA v dvojno vijačnico, po tem ko je bila razbita na posamezni verigi.
15	IGF2BP1-3	PARCLIP	F	755,248	6,097,934	12.39	Protein, ki skrbi za transport in hranjenje mRNA v citoplazmi ter zavira ali pospešuje translacijo mRNA.
16	hnRNPC (1)	iCLIP	G	527,734	4,602,041	11.47	Protein, ki se veže pre-mRNA z
17	hnRNPC (2)	iCLIP	G	23,226	228,961	10.14	hnRNP delci. Veže se na 3' in 5' konce mRNA in spreminja stabilnost molekule in stopnjo translacije.
18	hnRNPL (1)	iCLIP	H	10,563	123,685	8.54	Protein, ki deluje kot faktor
19	hnRNPL (2)	iCLIP	H	117,019	1,125,304	10.40	spajanja pre-mRNA. Veže se na
20	hnRNPL-like	iCLIP	H	11,464	128,958	8.89	začetek in konce eksonov in intronov ter deluje, kot zaviralec ali pospeševalec vključitve eksona.
21	MOV10	PARCLIP	I	81,377	592,451	13.74	Protein, ki zavira ekspresijo genov DNA.
22	Nsun2	iCLIP	J	7,310	75,343	9.70	Protein, ki doda metilno skupino tRNA.
23	PUM2	PARCLIP	K	40,422	368,700	10.96	Protein, ki zmanjšuje stabilnost mRNA tako, da se veže na 3' konec mRNA.
24	QKI	PARCLIP	L	39,668	381,140	10.41	Protein, ki upravlja spajanje pre-mRNA, prenos in stabilnost ter translacijo mRNA.
25	RBPMS	PARCLIP	M	46,545	455,809	10.21	Protein, ki sodeluje kot aktivator transkripcije DNA.
26	SFRS1	CLIPSEQ	N	121,786	966,912	12.60	Protein, ki sodeluje pri preprečevanju izpuščanja eksonov, zagotavlja pravilnost spajanja in upravlja alternativno spajanje pre-mRNA.
27	TAF15	PARCLIP	O O	26,650	222,421	11.98	Protein, ki sodeluje pri začetku transkripcije DNA.
28	TDP-43	iCLIP	P	261,163	3,053,718	8.55	Protein, ki upravlja transkripcijo DNA in spajanje pre-mRNA.
29	TIA1	iCLIP	Q	47,671	393,111	12.13	Protein, ki sodeluje pri
30	TIAL1	iCLIP	Q	132,540	1,146,658	11.56	alternativnem spajanju pre-mRNA in upravljanju translacije mRNA.
31	U2AF2 (1)	iCLIP	R	1,008,472	9,147,292	11.03	Protein, ki sodeluje pri tvorbi
32	U2AF2 (2)	iCLIP	R	402,829	3,668,916	10.98	kompleksov z ribosomi in mRNA.

Tabela 2.2: Tabela eksperimentov CLIP, ki so bili uporabljeni pri gradnji modelov in opisi proteinov iz vira [50].

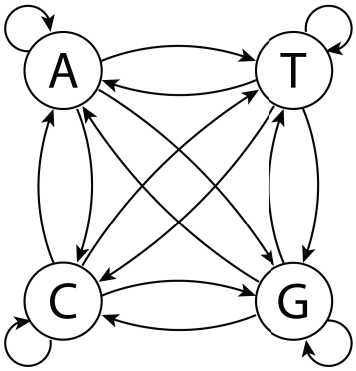
Poglavje 3

Metode

Najprej smo definirali Markove verige (MC) in jih razširili v skrite Markove modele (HMM). Nato smo definirali način določanja začetnih parametrov HMM iz zaporedij stanj in emisij. Definirali smo Viterbi algoritem, ki nam za zaporedje emisij vrne zaporedje stanj, ki predstavlja najbolj verjetno pot, kot bi jo ustvaril trenutni HMM. Aposteriorno dekodiranje pa nam za zaporedje emisij vrne zaporedje stanj, ki je sestavljeno iz najbolj verjetnih stanj v vsakem koraku. Algoritma imata pri zelo dolgih zaporedjih težave z matematično stabilnostjo, ki smo jo morali izboljšati. Na koncu smo še definirali metode, s katerimi smo vrednotili posamezne atribute, napovedi in rezultate.

3.1 Markove verige

Markove verige so modeli, s katerimi lahko ustvarimo zaporedja, kjer je verjetnost pojavitve simbola v zaporedju odvisna od predhodnih simbolov. Model si lahko predstavljamo kot množico stanj, kjer so prehodi med stanji določeni s puščicami (Slika 3.1). Vsak prehod med dvema stanjema je določen s puščico, ki ima tudi pripisano verjetnost, s katero se ta prehod izvrši. Verjetnosti na puščicah imenujemo *verjetnosti prehodov*, ki jih označimo s simbolom a_{st} in enačbo (3.1).



Slika 3.1: Markova veriga, s katero ponazorimo zaporedje nukleotidov DNA. Povzeto po viru [1].

$$a_{st} = P(x_i = t|x_{i-1} = s) \tag{3.1}$$

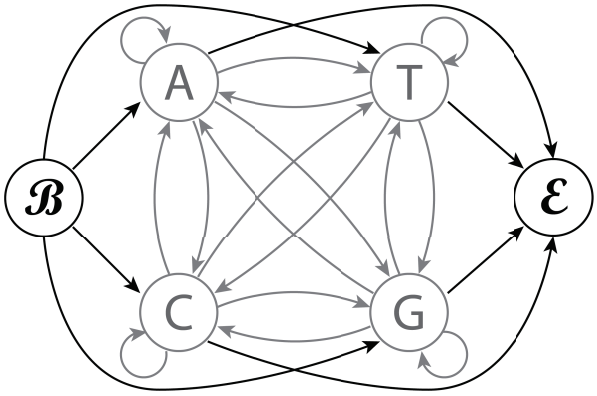
Za vsak verjetnostni model zaporedja simbolov lahko zapišemo verjetnost celotnega zaporedja z večkratno uporabo enačbe (3.2), da dobimo končno enačbo (3.3).

$$P(X, Y) = P(X|Y)P(Y) \tag{3.2}$$

$$P(x) = P(x_L, x_{L-1}, ..., x_1) = P(x_L|x_{L-1}, ..., x_1)P(x_{L-1}|x_{L-2}, ..., x_1)...P(x_1) \tag{3.3}$$

Glavna lastnost MC je, da je verjetnost za pojavitev določenega x_i simbola odvisna le od predhodnih simbolov $x_{i-1}, x_{i-2}...x_{i-n}$ in ne od celotnega predhodnega zaporedja. Če določimo, da je trenutni simbol odvisen samo od enega predhodnega simbola x_{i-1}, lahko enačbo (3.3) preoblikujemo v enačbo (3.4).

$$P(x) = P(x_L|x_{L-1}P(x_{L-1}|x_{L-2})...P(x_2|x_1)P(x_1) = P(x_1)\prod_{i=2}^{L} a_{x_{i-1}x_i} \tag{3.4}$$



Slika 3.2: Markova veriga, ki predstavlja zaporedje nukleotidov DNA, z začetnim in s končnim stanjem. Povzeto po viru [1].

Poleg definiranja prehodnih verjetnosti, moramo definirati tudi verjetnost $P(x_1)$, da začnemo v določenem vozlišču. Z uporabo začetnih verjetnosti in uvedbo dodatnega začetnega vozlišča $\mathcal{B}$ se lahko izognemo nehomogenosti enačbe (3.4). Definiramo $x_0 = \mathcal{B}$ kot začetek zaporedja in tako dobimo verjetnost za prvi simbol v zaporedju po enačbi (3.5) (Slika 3.2).

$$P(x_1 = s) = a_{\mathcal{B}s} \tag{3.5}$$

Podobno definiramo končno vozlišče $\mathcal{E}$, da lahko zaključimo zaporedje z verjetnostjo po enačbi (3.6).

$$P(\mathcal{E}|x_L = t) = a_{z\mathcal{E}} \tag{3.6}$$

Ni nujno, da eksplicitno ustvarimo začetno in končno stanje, vendar ju lahko obravnavamo le implicitno kot začetek in konec zaporedja. MC predvidevajo, da se lahko zaporedje konča kadarkoli. Če določimo konec zaporedja, potem s tem modeliramo porazdelitev dolžin zaporedij in tako model definira verjetnostno porazdelitev čez vsa možna zaporedja vseh možnih dolžin [1].

3.2 Skriti Markovi modeli

Skriti Markovi modeli se uporabljajo takrat, ko želimo ugotoviti zaporedje stanj iz zaporedja simbolov. Zaporedje stanj poimenujemo kot pot π. Sama pot sledi MC, torej je verjetnost, da se v določenem trenutku nahajamo v določenem stanju, odvisna le od predhodnega stanja. Verjetnost za poljubno lokacijo π_i na poti dobimo z enačbo (3.7).

$$a_{kl} = P(\pi_i = l|\pi_{i-1} = k) \tag{3.7}$$

Za definiranje začetka uporabimo začetno stanje in definiramo verjetnost prehoda a_{0k} iz začetnega stanja v stanje k, ki jo razumemo kot verjetnost, da začnemo v stanju k. Prav tako uporabimo verjetnost prehoda a_{k0} kot prehod v končno stanje in s tem zaključimo zaporedje. Tako začetno kot končno vozlišče lahko označimo z 0, saj ne pride do konfliktov, ker imamo iz začetnega vozlišča le izhodne povezave in v končnem stanju le vhodne povezave.

Ker skušamo ugotoviti v kakšnem stanju je sistem na določenem mestu v zaporedju, moramo razdvojiti verjetnosti prehodov med stanji in verjetnosti ustvarjanja simbolov v določenem stanju. Uvedemo verjetnost emisije $e_k(b)$, ki predstavlja verjetnost emisije simbola b, če se nahajamo v stanju k in dobimo enačbo (3.8).

$$e_k(b) = P(x_i = b|\pi_i = k) \tag{3.8}$$

Zaporedje si predstavljamo, kot da je ustvarjeno z MC. Najprej smo izbrali stanje π_1 z verjetnostjo a_{0i}. V tem stanju smo smo emitirali simbol glede na verjetnostno porazdelitev emisij e_{π_1}. Novo stanje π_2 je izbrano na podlagi prehodnih verjetnosti $a_{\pi_1 i}$ in tako dalje. Tako ustvarimo zaporedje naključnih umetnih simbolov in lahko določimo verjetnost $P(x)$, ki nam pove, s kakšno verjetnostjo je bilo ustvarjeno zaporedje simbolov x. Skupno verjetnost opazovanega zaporedja x in zaporedja stanje π zapišemo z enačbo (3.9).

$$P(x, \pi) = a_{0\pi_1} \prod_{i=1}^{L} e_{\pi_i} a_{\pi_i \pi_{i+1}} \tag{3.9}$$

Enačba (3.9) nam v praksi ne pride prav, saj ponavadi ne poznamo poti, lahko pa ocenimo najbolj verjetno ali pa uporabimo aposteriorno verjetnostno porazdelitev čez vsa stanja [1].

HMM rešujejo tri osnovne probleme [39]:

- **Evaluacija** - Če imamo dan HMM in zaporedje emisij, kakšna je verjetnost, da je HMM ustvaril dano zaporedje emisij?
- **Dekodiranje** - Če imamo dan HMM in zaporedje emisij, katero zaporedje stanj modela je z največjo verjetnostjo ustvarilo dano zaporedje emisij?
- **Učenje** - Če imamo dan HMM in zaporedje emisij, kako moramo prilagoditi parametre modela, da maksimiramo verjetnost, da je HMM ustvaril dano zaporedje emisij?

3.3 Inicializacija in določanje parametrov

Skriti Markov model inicializiramo tako, da mu določimo strukturo, to pomeni, da določimo, kaj posamezna stanja predstavljajo in kako so le ta povezana med seboj. Parametri modela določajo vrednosti, ki predstavljajo verjetnosti prehodov med stanji in verjetnosti emisij simbolov v posameznih stanjih. Za določitev parametrov imamo natančno določene postopke, medtem ko je pa določitev strukture odvisna od tega, kaj želimo z modelom predstaviti in za to nimamo določenih postopkov.

Iz množice učnih zaporedij simbolov in stanj določimo parametre tako, da preštejemo, kolikokrat se pojavi določena tranzicija A_{kl} ali emisija $E_k(b)$ v določenem stanju. Z uporabo enačbe (3.10) dobimo oceno maksimalnega verjetja (angl. Maximum Likelihood Estimation - MLE) posamezne tranzicije a_{kl} in z uporabo enačbe (3.11) dobimo MLE posamezne emisije $e_k(b)$ [1].

$$a_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}} \tag{3.10}$$

$$a_k(b) = \frac{E_k(b)}{\sum_{b'} E_k(b')} \tag{3.11}$$

3.4 Viterbijev algoritem

Za vsako zaporedje emitiranih simbolov lahko najdemo veliko zaporedij stanj, ki so zaporedja ustvarila. Razlikujejo se le po verjetnostih, s katerimi so se določena stanja pojavila v zaporedju in s kakšnimi verjetnostmi so se emitirali simboli v teh stanjih. Viterbijev algoritem nam poišče najbolj verjetno zaporedje stanj oziroma pot stanj za vhodno zaporedje emisij. Viterbijev algoritem spada pod algoritme dinamičnega programiranja. Najbolj verjetno zaporedje stanj π^* predstavimo z enačbo (3.12) in do nje pridemo z uporabo rekurzije.

$$\pi^* = argmax_\pi P(x, \pi) \tag{3.12}$$

Verjetnost $v_k(i)$ predstavlja najbolj verjetno zaporedje, ki se konča v stanju k z emisijo i. V vsakem koraku poznamo verjetnosti za vsa možna stanja k. Te verjetnosti lahko z emisijo x_{i+1} predstavimo z enačbo (3.13).

$$v_l(i+1) = e_l(x_{i+1}) max_k(v_k(i) a_{kl}) \tag{3.13}$$

Vsa zaporedja se začnejo z začetnim stanjem, zato imamo začetno verjetnost $v_0(0) = 1$. Pri izvajanju algoritma obdržimo kazalce na prejšnja stanja in tako lahko najdemo najbolj verjetno zaporedje z uporabo vračanja oziroma sestopanja po kazalcih od konca do začetka zaporedja [1].

3.5 Algoritem naprej

Ker veliko zaporedij stanj lahko ustvari enako zaporedje emitiranih simbolov za različno verjetnostjo, moramo sešteti verjetnosti vseh zaporedij, da

Viterbijev algoritem:	
Inicializacija $(i = 0)$:	$v_0(0) = 1, v_k = 0\ za\ k > 0$
Rekurzija $(i = 1...L)$:	$v_l(i) = e_l(x_i)max_k(v_k(i-1)a_{kl})$
	$ptr_i(l) = argmax_k(v_k(i-1)a_{kl})$
Terminacija:	$P(x, \pi_*) = max_k(v_k(L)a_{k0})$
	$\pi_L^* = argmax_k(v_k(L)a_{k0})$
Sestopanje $(i = L...1)$:	$\pi_{i-1}^* = ptr_i(\pi_i^*)$

dobimo celotno verjetnost zaporedja emisij. Število vseh možnih zaporedij stanj π narašča eksponentno z dolžino zaporedja emisij, zato naivna uporaba enačbe (3.14), ki gre skozi vse možna zaporedja stanj, ni praktična. Enega od pristopov predstavlja enačba (3.9), s katero izračunamo najbolj verjetno zaporedje π^*, ki predstavlja približek $P(x)$.

$$P(x) = \sum_{\pi} P(x, \pi) \tag{3.14}$$

Izračun približka je v tem primeru nepotreben, saj lahko izračunamo celotno verjetnost z dinamičnim programiranjem in algoritmom, podobnim Viterbiju, kjer zamenjamo iskanje maksimumov z vsotami. Nov algoritem imenujemo algoritem naprej. Verjetnost za opazovano zaporedje emisij do vključno mesta x_i, pri tem da se nahajamo v stanju $\pi_i = k$, dobimo z enačbo (3.15). Algoritem prav tako izvajamo rekurzivno po enačbi (3.16)[1].

$$f_k(i) = P(x_1...x_i, \pi_i = k) \tag{3.15}$$

$$f_l(i+1) = e_l(x_{i+1}) \sum_k f_k(i) a_{kl} \tag{3.16}$$

3.6 Algoritem nazaj

Verjetnost, da je emisija na mestu x_i prišla iz stanja k, predstavimo s $P(\pi = k|x)$. Tej verjetnosti rečemo aposteriorna verjetnost stanja k ob času i, pri

Algoritem naprej:	
Inicializacija $(i = 0)$:	$f_0(0) = 1, f_k = 0 \; za \; k > 0$
Rekurzija $(i = 1...L)$:	$f_l(i) = e_l(x_i) \sum_k f_k(i-1) a_{kl}$
Terminacija:	$P(x) = \sum_k f_k(L) a_{k0}$

tem da smo poznali zaporedje emisij. Najprej izračunamo verjetnost, da je bilo celotno zaporedje emisij ustvarjeno z $i - tim$ simbolom iz stanja k, ki jo predstavimo z enačbo (3.17), kjer drugo vrstico dobimo, ker je zaporedje po stanju k odvisno le od stanja k.

$$\begin{aligned} P(x, \pi = k) &= P(x_1...x_i, \pi_i = k) P(x_{x+1}...x_L | x_1...x_i, \pi_i = k) \\ &= P(x_1...x_i, \pi_i = k) P(x_{x+1}...x_L | \pi_i = k) \end{aligned} \tag{3.17}$$

V prvem delu enačbe prepoznamo $f_k(i)$ iz enačbe (3.15), ki smo ga izračunali z algoritmom naprej. Drugi del enačbe predstavimo s spremenljivko $b_k(i)$ in enačbo (3.18), ki je analogna spremenljivki algoritma naprej $f_k(i)$, le da jo izračunamo v vračanju rekurzije z začetkom na koncu zaporedja.

$$b_k(i) = P(x_{i+1}...x_L | \pi_i = k) \tag{3.18}$$

Algoritem nazaj:	
Inicializacija $(i = L)$:	$b_k(L) = a_{k0} \; za \; k > 0$
Rekurzija $(i = L - 1...1)$:	$b_l(i) = \sum_l a_{kl} e_l(x_{i+1}) b_l(i+1)$
Terminacija:	$P(x) = \sum_l a_{0l} e_l(x_1) b_l(1)$

Terminacija v algoritmu je nepotrebna, saj $P(x)$ izračunamo že z algoritmom naprej in je tukaj navedena le zaradi celovitosti.

Enačbo (3.17) lahko zapišemo kot $P(x, \pi_i = k) = f_k(i) b_k(i)$, iz katere lahko izpeljemo aposteriorno verjetnost z enačbo (3.19), kjer je $P(x)$ rezultat algoritma naprej ali nazaj [1].

$$P(\pi = k|x) = \frac{f_k(i)b_k(i)}{P(x)} \tag{3.19}$$

3.7 Aposteriorno dekodiranje

Glavna uporaba enačbe (3.19) je, da predstavlja dva alternativna postopka Viterbi algoritmu. Oba postopka sta uporabna predvsem v primerih, ko imamo veliko poti stanj z zelo podobnimi verjetnostmi, kjer ni vedno dobra praksa, da upoštevamo le najbolj verjetno.

Za prvi postopek definiramo zaporedje stanj $\hat{\pi}_i$ in dobimo enačbo (3.20). Iz enačbe vidimo, da je to zaporedje stanj bolj primerno, če nas zanima stanje pri določeni emisiji i, kot pa celotno zaporedje stanj. Zaporedje stanj $\hat{\pi}_i$ lahko tudi predstavlja zaporedje, ki ni nujno zelo verjetno, kot celotno zaporedje stanj ali pa celo ne predstavlja veljavnega celotnega zaporedja stanj.

$$\hat{\pi}_i = argmax_k P(\pi = k|x) \tag{3.20}$$

Drugi postopek izhaja iz tega, da nas v določenih primerih ne zanima zaporedje stanje, vendar kakšna druga lastnost, izpeljana iz njega. Na primer, da imamo za stanja zaporedja definirano funkcijo $g(k)$, potem lahko izpeljemo enačbo (3.21). Pomemben poseben primer uporabe postopka je, ko $g(k)$ vrne vrednost 1 za podmnožico množice stanj in 0 za ostala stanja. V tem primeru nam $G(i|x)$ predstavlja aposteriorno verjetnost, da se emitira simbol i iz stanja v podmnožici stanj.

$$G(i|x) = \sum_k P(\pi = k|x)g(k) \tag{3.21}$$

3.8 Numerična stabilnost algoritmov

Tudi sodobni procesorji imajo omejen interval realnih števil, nad katerimi izvajajo računske operacije z uporabo plavajoče vejice. Zato pri uporabi Viterbi algoritma in algoritma naprej ter nazaj mnogokrat množimo vedno

manjše verjetnosti med seboj. Tako pridemo hitro do najmanjšega možnega števila, ki ga je procesor še sposoben zapisati. Ko prekoračimo to mejo, procesor nastavi vrednost te spremenljivke na nenumerični znak (angl. Not a number), za katerega procesor nima definiranih računskih operacij, kar povzroči, da se program nepravilno zaključi.

Pri Viterbi algoritmu lahko namesto verjetnosti uporabimo logaritme verjetnosti, in sicer zato, ker velja zakonitost, da je logaritem produkta enak vsoti logaritmov posameznih faktorjev (3.22).

$$log(x) + log(y) = log(xy) \tag{3.22}$$

Tako vse operacije množenja zamenjamo z operacijo seštevanja in smo rešili problem numerične stabilnosti. Logaritem spremenljivke označimo z uporabo tilde $\tilde{a}_{kl} = log(a_{kl})$ in tako pridemo do definicije razširjene enačbe rekurzije Viterbi algoritma (3.23), kjer V predstavlja logaritem v. Baza logaritma ni pomembna, le da vedno uporabljamo enako in da je večja od ena. Za boljšo učinkovitost pred uporabo logaritemskega Viterbi algoritma pretvorimo vrednosti matrik prehodov in emisij v logaritemske in tako zmanjšamo število logaritmiranja pri vsakem koraku rekurzije.

$$V_l(i+1) = \tilde{e}_l(x_{i+1}) + max_k(V_k(i) + \tilde{a}_{kl}) \tag{3.23}$$

Pri algoritmu nazaj in naprej pa uporaba logaritmov ni praktična, saj logaritma vsote ne moremo izračunati brez uporabe eksponentnih in logaritemskih funkcij, kar močno poveča računsko zahtevnost algoritma. Zato pri alternativnem pristopu pri vsakem koraku i vrednosti f in b spremenljivk pomnožimo z dodatno spremenljivko s_i in tako dobimo nove vrednosti, ki so definirane z enačbo (3.24). Iz te enačbe lahko enostavno izpeljemo razširjeno enačbo za posamezen korak rekurzije algoritma naprej(3.25).

$$\tilde{f}_l(i+1) = \frac{f_l(i+1)}{\prod_{j=1}^{i} s_j} \tag{3.24}$$

$$\tilde{f}_l(i+1) = \frac{1}{s_{i+1}} e_l(x_{i+1}) \sum_k \tilde{f}_k(i) a_{kl} \tag{3.25}$$

Ta metoda deluje za različne definicije spremenljivke s_i, vendar praktična odločitev je, da velja $\sum_l \tilde{f}_l(i) = 1$. Enačbo (3.26), ki izpolnjuje naš pogoj, dobimo tako , da seštejemo nove vrednosti $f_l(i+1)$. Za vsa stanja znotraj koraka i izračunamo s_{i+1} ter z njo delimo nove vrednosti s spremenljivko $\tilde{f}_l(i+1) = \frac{f_l(i+1)}{s_{i+1}}$ in dobimo končne vrednosti v tem koraku.

$$s_{i+1} = \sum_l e_l(x_{i+1}) \sum_k \tilde{f}_k(i) a_{kl} \tag{3.26}$$

Vrednosti s_i v vsakem koraku shranimo, ker jih potrebujemo pri algoritmu nazaj, kjer jih uporabimo na enakih mestih v zaporedju ter tako dobimo razširjeno enačbo rekurzije algoritma nazaj (3.27). Ta metoda množenja vrednosti v vsakem koraku v praksi dobro deluje, vendar lahko v robnih primerih še vedno privede do matematične nestabilnosti, in sicer v primerih, kjer imamo veliko skritih stanj.

$$\tilde{b}_k(i) = \frac{1}{s_{i+1}} \sum_l a_{kl} \tilde{b}_l(i+1) e_l(x_i + 1) \tag{3.27}$$

3.9 Vrednotenje

Za vrednotenje smo uporabili prečno preverjanje, krivuljo ROC (angl. Receiver Operating Characteristic - ROC) in ploščino pod krivuljo ROC (angl. Area Under the ROC Curve - AUC) ter povprečno absolutno napako (angl. Mean Absolut Error - MAE).

Prečno preverjanje je metoda za vrednotenje modelov na neodvisni podmnožici podatkov. Uporablja se predvsem v primeru, ko je cilj modela napovedovanje in nam oceni, kako dobro se bo model obnesel v praksi. Učenje modela poteka na učni množici podatkov, vrednotenje naučenega modela se nato opravi na testni množici podatkov. Učna in testna množica sta neodvisni. Z uporabo prečnega preverjanja se izognemo problemu prekomernega

prilagajanja modela podatkom [51, 52, 53].

Krivulja ROC predstavlja graf, ki prikaže učinkovitost binarnega klasifikatorja za vse vrednosti diskriminacijskega praga. Krivuljo dobimo, tako da narišemo občutljivost (os y) v odvisnosti od 1 - specifičnost (os x). Omogoča nam primerjavo modelov in izbiro najboljšega, prav tako tudi izbiro optimalnega diskriminacijskega praga posameznega modela. V osnovni obliki krivulje ROC podpirajo le binarne klasifikatorje. AUC nam predstavlja ploščino pod krivuljo ROC, ter nam poda eno vrednost, ki nam opiše učinkovitost modela. Omogoča enostavnejšo primerjavo modelov kot pa krivulje ROC. Vrednost AUC je enaka verjetnosti, da klasifikator klasificira naključni pozitivni primer višje kot pa naključni negativni primer[54, 55, 56, 57].

MAE je mera, ki nam pove, povprečno absolutno razliko med napovedanimi in dejanskimi vrednostim. Definirana je z enačbo (3.28), kjer f_i predstavlja napovedano vrednost in y_i dejansko vrednost [58] razreda. Uporabimo jih lahko tudi za vrednotenje klasifikatorjev z več kot dvema razredoma.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| \tag{3.28}$$

3.10 Združevanje modelov

Združevanje modelov smo izvedli tako, da smo združili napovedi več HMM v emisije zaporedij. Od izbranih HMM smo za vsak nukleotid zaporedja dobili vektor napovedi. Vsak znak emisije predstavlja napoved enega HMM za trenuten nukleotid. Tako smo definirali večznačni problem, ki predstavlja problem pri katerem imamo za posamezen podatek podanih več oznak, na podlagi katerih ji določimo razred. Večznačni problem lahko rešujemo s transformacijskimi metodami ali pa z algoritmičnem prilagajanjem. Transformacijske metode večznačni problem razbijejo na množico binarnih klasifikacijskih problemov. Algoritmično prilagajanje pa direktno izvaja večznačno klasifikacijo [44, 45, 46, 47, 48]. Napovedovanje nismo razdelili na več binarnih problemov, smo pa generalizirali emisije tako, da določene emisije

predstavljajo mesta vezave, druge pa ne. Tako stanji HMM predstavljata razreda v katera razporedimo naše emisije. Generalizacijo emisij smo naredili po osnovnem postopku generalizacije kopice, ki več oznak posploši in predstavi kot eno stanje [6]. Iz postopkov večznačne klasifikacije in generalizacije kopice smo povzeli ideje in razvili svoj postopek osnovnega združevanja napovedi več HMM v združen HMM preko stanj in emisij zaporedij.

Poglavje 4

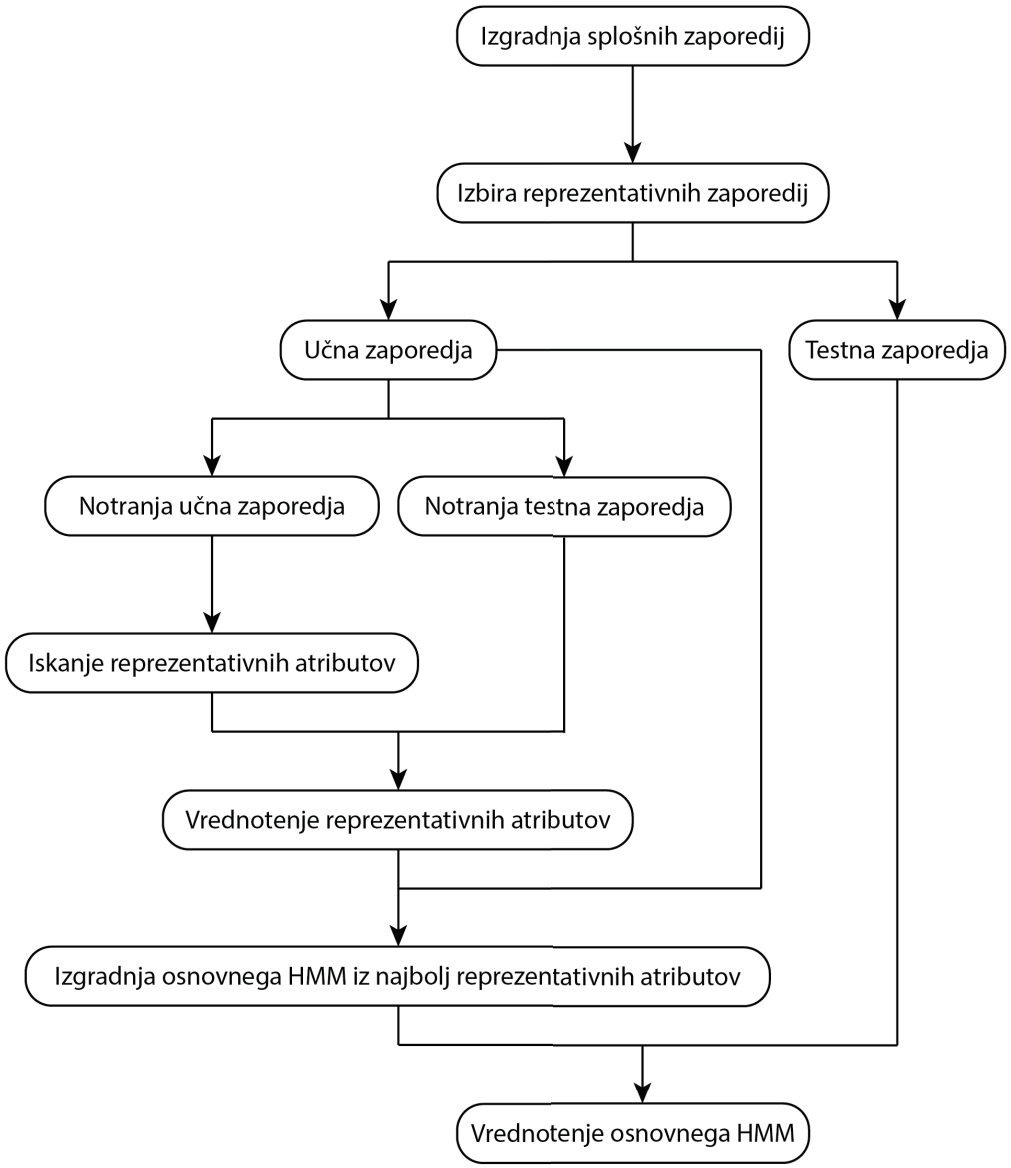
Gradnja modelov HMM

Pred samo gradnjo HMM smo najprej pregledali podatke, jih preoblikovali za delo s HMM ter zavrgli slabe. Nato smo določili reprezentativne atribute vsakega eksperimenta in iz njih zgradili HMM za vsak eksperiment posebej. HMM so bili najprej zgrajeni za preprosti binarni problem, in sicer, ali je na določenem nukleotidu prišlo do vezave ali ne. Nato smo zgradili še HMM s tremi stanji, s katerimi smo želeli opisati različne intenzitete vezave. Izgrajene modele smo vrednotili in primerjali njihove napovedi za vsak posamezen eksperiment. Napovedovanje mest vezave za posamezen eksperiment smo nato poskusili izboljšati z združevanjem napovedi HMM, ki so najbolje napovedali posamezni eksperiment.

4.1 HMM eksperimentov z dvema stanjema

Pred samo gradnjo modelov smo za vsak protein posebej **izgradili splošna zaporedja** (Slika 4.1), ki predstavljajo posamezne gene. Vsako zaporedje nukleotidov, ki predstavlja posamezni gen, smo predstavili z dvema zaporedjema. Prvo zaporedje je predstavljalo stanja, drugo zaporedje pa emisije.

Stanja smo predstavili z dvema vrednostma, in sicer 0 ter 1. Stanja predstavljajo intenzitete mest interakcij oziroma podatkov CLIP posameznega eksperimenta. Stanje 0 predstavlja nukleotid, na katerem ni prišlo do inte-



Slika 4.1: Shema postopka gradnje osnovnega HMM posameznega eksperimenta.

rakcije med RNA in proteinom. Stanje 1 pa predstavlja nukleotid na katerem je prišlo do interakcije med RNA in proteinom. Intenziteta interakcije je predstavljena z zvezno vrednostjo, vendar smo vse posplošili v enako stanje. Tako smo dobili binarni problem, saj za vsak nukleotid, vemo ali je stopal v interakcijo ali ne. Zaporedje stanj predstavlja skrito zaporedje HMM, saj ga HMM vidi le pri učenju, pri napovedovanju pa ga skuša napovedati.

Emisije smo predstavili z nizi znakov, kjer je prvi znak predstavljal nukleotid, naslednjih šest znakov je predstavljalo funkcijske podenote gena in zadnji znak je predstavljal sekundarno strukturo RNA. Nukleotidi so bili predstavljeni z znaki A, C, G, T in N, kjer N predstavlja katerikoli nukleotid. Za vsako funkcijsko podenoto gena smo imeli binarno stikalo, in sicer 0, če nukleotid ne pripada posamezni podenoti, in 1, če ji pripada. Za sekundarno strukturo smo imeli tudi binarno stikalo, in sicer 0, če se nukleotid ne povezuje z ostalimi nukleotidi, in 1, če se povezuje. Emisije predstavljajo vidno zaporedje HMM, saj ga HMM vidi tako pri učenju kot napovedovanju.

Iz vseh vhodnih zaporedij smo **izbrali reprezentativna zaporedja** (Slika 4.1) tako, da smo izločili tista, kjer se interakcije med RNA in proteini v zaporedju pojavijo manj kot 0.1 ‰ ter so zaporedja daljša od 400,000 nukleotidov. Ta zaporedja nam ne pomagajo veliko pri določitvi reprezentativnih atributov, saj HMM smatra zelo majhen delež pojavitev določenega stanja kot šum. Prav tako smo pridobili na času izvajanja učenja.

Množico najbolj reprezentativnih zaporedij smo nato naključno razdelili na **učna** (80 %) in **testna** (20 %) **zaporedja** (Slika 4.1). Učna zaporedja smo uporabili za iskanje najbolj reprezentativnih atributov tega eksperimenta. Učna zaporedja smo še enkrat naključno razdelili na **notranja učna** (80 %) in **notranja testna** (20 %) **zaporedja** (Slika 4.1).

Poiskali smo reprezentativne atribute (Slika 4.1) izbranega eksperimenta tako, da smo na notranjih učnih zaporedjih prešteli pojavitve posameznih atributov znotraj emisij skozi vsa stanja in izračunali njihove verjetnosti pojavitve. Enako smo naredili tudi za vsako stanje posebej. Za stanje 1 smo vzeli še okolico 100 nukleotidov pred in po nukleotidih v tem stanju. Nu-

kleotidi pred in po samem mestu interakcije lahko predstavljajo pomembne motive, ki omogočajo proteinu najti dejansko mesto interakcije. Poleg prej omenjenih atributov smo sproti ustvarili še n-terke dolžine štiri in prav tako prešteli njihove pojavitve za vsa stanja skupaj in posebej. N-terke dolžine štiri smo izbrali zato, ker se v literaturi pojavi najbolj pogosto in ker smo hoteli obdržati število vseh možnih stanj emisij kar se da nizko, saj so HMM z manj emisijami bolj obvladljivi. Emisije, ki so se pojavile z verjetnostjo manj kot 0.1 ‰, smo zavrgli. Delili smo verjetnost pojavitev ostalih emisij v posameznem stanju z verjetnost pojavitev te emisije skozi vsa stanja. Če je bilo razmerje večje od 1, potem se je emisija znotraj določenega stanja pojavila bolj pogosto, kot pa v vseh stanjih skupaj.

Za vsako stanje smo **vrednotili reprezentativne atribute** (Slika 4.1), in sicer tako, da smo zgradili nova zaporedja iz učnih zaporedij. Zaporedja stanj so ostala enaka, zaporedja splošnih emisij pa smo zamenjali z zaporedji emisij posameznega atributa. Pri podenotah gena in sekundarni strukturi RNA smo vzeli samo tiste, ki so imeli razmerje pojavitve za izbrano stanje večje od 1. N-terke smo razporedili po padajočem razmerju njihove pojavitve v posameznemu stanju. Vzeli smo od 2 do 25 n-terk, dokler je v vsakem koraku n-terka z najnižjim razmerjem imela razmerje večje od 1. Iz prilagojenih notranjih učnih zaporedij smo zgradili in naučili HMM, ki smo ga nato vrednotili na prilagojenih notranjih testnih zaporedjih. S HMM smo napovedali skrita stanja iz emisij notranjih testnih podatkov ter jih primerjali z dejanskimi skritimi stanji. Primerjavo smo izvedli z uporabo krivulje ROC in vrednosti AUC. Atribut smo dodali med reprezentativne le takrat, ko je bil AUC prilagojenih testnih zaporedij večji od 0.5. Če ima AUC vrednost 0.5 pomeni, da testirani klasifikator deluje enako dobro kot naključni klasifikator. Pri n-terkah smo izbrali tisto število najboljših n-terk, ki so nam dale najboljšo vrednost AUC. V primeru, da nismo našli reprezentativnih n-terk, smo uporabili posamezne nukleotide. V najslabšem primeru, če ne najdemo nobenih reprezentativnih atributov, uporabimo za emisije le nukleotide.

Reprezentativne atribute iz vsakega stanja smo nato združili v enotno

množico skupnih atributov, tako da smo tiste atribute, ki se pojavijo le v posameznemu stanju, preprosto dodali. Tiste, ki so se pojavili v več stanjih hkrati, pa smo primerjali njihovo razmerje pojavitve med stanji in če je bila razlika večja od 1.5 smo jih dodali v skupne atribute, drugače pa smo jih zavrgli.

Iskanje reprezentativnih atributov znotraj učne množice smo ponovili štirikrat in vsakič dodali izbrane atribute posamezne iteracije v množico končnih skupnih atributov. Učna in testna zaporedja smo prilagodili tako, da so zaporedja stanj ostala enaka, zaporedja splošnih emisij pa smo zamenjali z zaporedji emisij, ki smo jih sestavili iz izbranih končnih atributov. Vse nereprezentativne n-terke smo predstavili z eno emisijo. Iz prilagojenih učnih in testnih zaporedij smo pridobili vse možne emisije, ki se pojavijo. V primeru, da obstaja emisija, ki se pojavi le v testnih zaporedjih, smo jo morali zamenjati z drugim stanjem. V primeru n-terke smo zamenjali del emisije s stanjem, ki predstavlja nereprezentativne n-terke. Če se spremenjena emisija še vedno ni pojavila v učnih emisijah, pa smo emisijo zamenjali s predhodno emisijo in v robnih primerih z naslednjo emisijo v zaporedju. Enako smo naredili tudi v primeru, če se je pojavila emisija s kombinacijo ostalih atributov, ki se ni pojavila v učnih emisijah.

Iz prilagojenih učnih zaporedij smo **zgradili osnovni HMM** in ga **vrednotili** na prilagojenih testnih zaporedjih (Slika 4.1). Vrednotenje smo izvedli s krivuljami ROC in vrednost AUC nam je podala končno mero zmogljivosti našega HMM. Tako smo zgradili 32 HMM za vsak eksperiment posebej.

4.2 HMM eksperimentov z več stanji

Pri gradnji HMM z več stanji smo se odločili, da izberemo tri stanja, ki so predlagana tudi v literaturi [15]. Podobno kot pri gradnji HMM z dvema stanjema smo za vsak protein posebej zgradili splošna zaporedja. Omenili bomo le razlike v primerjavi z gradnjo HMM z le dvema stanjema.

Stanja smo predstavili s tremi vrednostmi, in sicer 0, 1 ter 2. Stanje 0

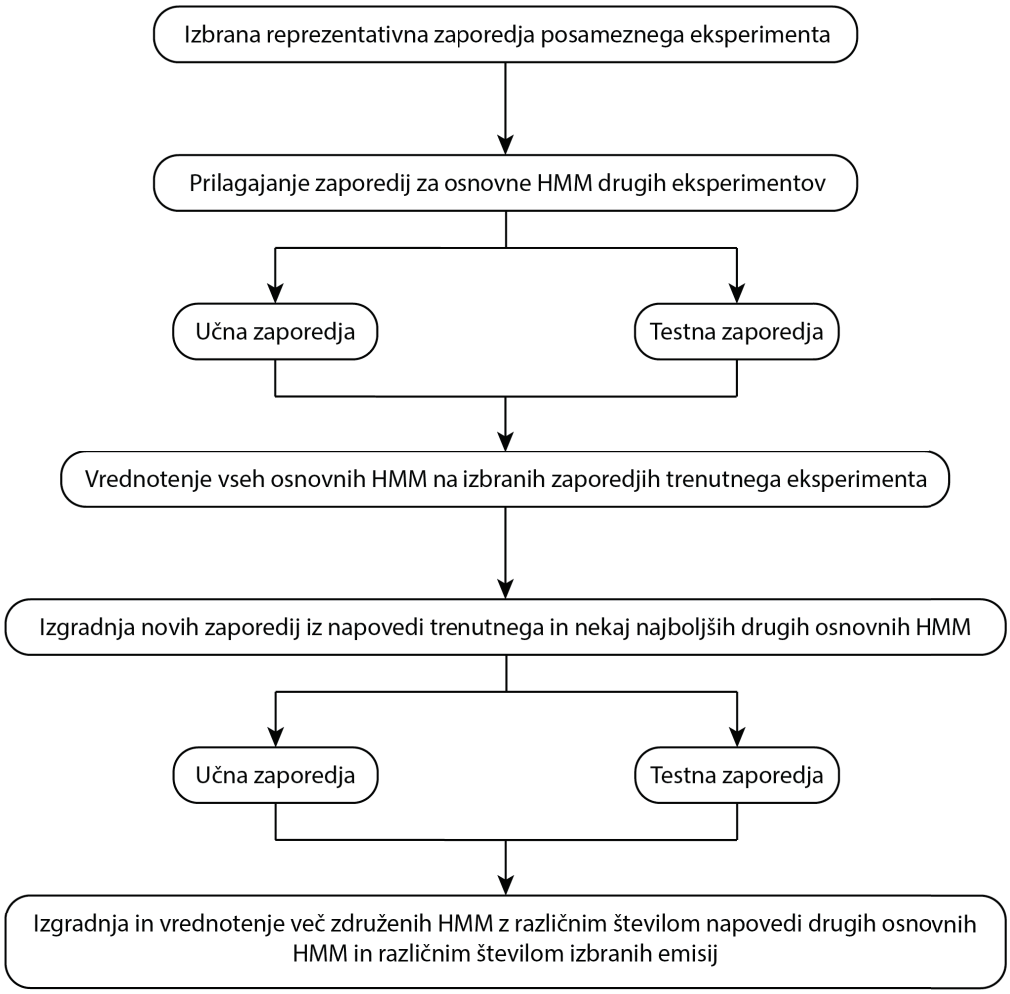
predstavlja nukleotid, na katerem ni prišlo do interakcije. Stanje 1 predstavlja nukleotid, na katerem je prišlo do interakcije z intenziteto med nič in pet. Stanje 2 pa predstavlja nukleotid, na katerem je prišlo do interakcije z intenziteto večjo ali enako pet. Z novimi stanji smo sedaj ponovno zgradili splošna zaporedja ter iz njih izbrali najbolj reprezentativna zaporedja. Podobno smo postavili mejo za maksimalno dolžino zaporedja na 400.000 nukleotidov in mejo za minimalno verjetnost pojavitve mesta vezave na 0.1 ‰, kjer je zadostovalo, da ima vsaj eno izmed stanj 1 in 2 minimalno verjetnost pojavitve večjo od dane meje.

Ker nimamo več binarnega klasifikatorja, smo sedaj atribute in HMM vrednotili z uporabo MAE. Vrednost, ki nam predstavlja zmogljivost klasifikatorja, smo dobili tako, da smo izračunali povprečje MAE vsakega testnega zaporedja. MAE pa smo izvedli tako, da smo primerjali napovedane in dejanske verjetnosti za vsako stanje za določen nukleotid. Za reprezentativne atribute smo izbrali le tiste, ki so vrnili vrednost MAE manjšo od 0.5.

4.3 Združevanje HMM

Pred samim združevanjem modelov smo za vsak eksperiment ugotovili, kako dobro ga napovedujejo modeli drugih eksperimentov. Za vsak eksperiment smo vzeli učna in testna zaporedja, ki smo jih uporabili pri gradnji posameznih HMM. Nato smo ta zaporedja poiskali v prvotnih splošnih zaporedjih, ki smo jih morali **prilagoditi za vsak HMM drugega eksperimenta posebej** (Slika 4.2). Sedaj smo posamezne modele naučili na prilagojenih **učnih zaporedjih** in jih **vrednotili** na prilagojenih **testnih zaporedjih** (Slika 4.2). Tako smo dobili krivulje ROC in vrednosti AUC za vsak eksperiment od HMM ostalih eksperimentov. Vrednosti AUC napovedi smo še uredili po velikosti, saj imajo boljši klasifikatorji višjo vrednot AUC kot slabši.

Kako dobro HMM drugih eksperimentov napovedujejo izbrani eksperiment, nam lahko pove nekaj o vrsti samih proteinov ter o njihovih odnosih.

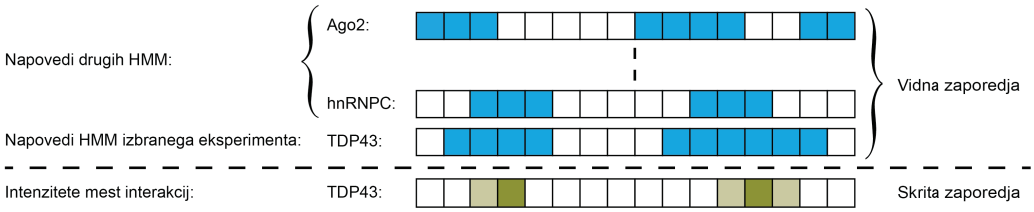


Slika 4.2: Shema postopka gradnje združenih HMM posameznega eksperimenta.

Za lažjo ponazoritev smo zgradili graf, kjer vozlišča predstavljajo posamezni eksperimenti. Usmerjene povezave so definirane z začetnim in končnim vozliščem ter utežjo. Začetno vozlišče predstavlja eksperiment, ki je med najboljših n eksperimentov pri napovedi eksperimenta, ki predstavlja končno vozlišče. Utež povezave predstavlja vrednost AUC, ki jo doseže HMM eksperimenta v začetnem vozlišču pri napovedi eksperimenta v končnem vozlišču. Vrednosti AUC so si lahko zelo blizu, zato smo jih preštevilčili tako, da ima najmanjši AUC vrednot 1, največji AUC pa 3. S tem smo dosegli boljšo razporeditev vozlišč v grafu, ki smo ga izrisali z uporabo silno usmerjenega algoritma.

Združevanje vseh modelov v skupen HMM ni praktično, saj število vseh možnih emisij postane preveliko. Zato smo združevali le po nekaj najboljših modelov skupaj. Pred izbiro najboljših smo izločili replike eksperimentov posameznega proteina, kot smo določili v tabeli (Tabela 2.2). Pri gradnji združenega HMM za posamezen eksperiment smo izbrali najboljše štiri HMM drugih eksperimentov. Sedaj smo **spremenili učna in testna zaporedja** (Slika 4.2), ki smo jih uporabili pri gradnji posameznih HMM, in sicer tako, da smo emisije zamenjali z napovedmi HMM, medtem ko pa je zaporedje stanj ostalo nespremenjeno (Slika 4.3). Za osnovo smo vzeli kar napovedi HMM eksperimenta, za katerega gradimo združeni model, ki za vsak nukleotid poda vrednost 0, če ni napovedana vezava, in 1, če je napovedana vezava. S tem smo predstavili atribute, ki smo jih uporabili za gradnjo posameznih HMM. Poskusili smo tudi z uporabo vseh atributov, vendar ni prišlo do izboljšanja HMM. Hkrati pa smo s tem tudi dosegli zmanjšanje števila vseh možnih emisij. Nato pa smo emisijam dodali še napovedi ostalih najboljših HMM drugih eksperimentov. Zgradili smo zaporedja z emisijami HMM eksperimenta in najboljšega HMM drugega eksperimenta, nato z najboljšima dvema HMM drugih dveh eksperimentov in tako naprej, dokler ni bila emisija dolga pet mest. Učna in testna zaporedja so predstavljala zaporedja, ki so bila učna in testna tudi pri gradnji posameznega HMM tega eksperimenta.

Iz spremenjenih učnih zaporedij smo nato **zgradili združene HMM ter**



Slika 4.3: Prikaz podatkov, ki smo jih uporabili pri gradnji združenih HMM za posamezne eksperimente.

ga vrednotili na spremenjenih testnih zaporedjih (Slika 4.2). Najprej smo prešteli in izračunali verjetnosti pojavitev emisij skozi vsa učna zaporedja. In nato še za vsako stanje posebej. Pri stanju 1 smo upoštevali še okolico sto nukleotidov pred in po mestu interakcije. Emisije, ki so se pojavile z verjetnostjo manj kot 0.1 ‰, smo zavrgli, za ostale emisije pa smo izračunali razmerje pojavitve v posameznem stanju. Nato smo se sprehodili skozi vsa stanja in vsakemu uredili emisije po razmerju pojavitve od največjega do najmanjšega. Vzeli smo po dve najboljši emisiji, nato tri, dokler ni razmerje pojavitve emisij padlo pod mejo 1.1 ali pa smo prišli do največ desetih najboljših emisij. Neizbrane emisije smo združili v eno stanje. Glede na izbrane emisije smo učna in testna zaporedja ponovno prilagodili ter iz njih zgradili, naučili in vrednotili združene HMM. Vrednotenje smo izvedli s krivuljami ROC in vrednostmi AUC. Ta korak smo ponovili za dva, tri in vse do pet izbranih eksperimentov, ki najbolje napovedujejo izbrani eksperiment. Iz vseh teh HMM smo nato na podlagi vrednosti AUC izbrali najboljšega, ki je predstavljal naš končni HMM izbranega eksperimenta. Ta postopek smo izvedli za vseh dvaintrideset eksperimentov.

Poglavje 5

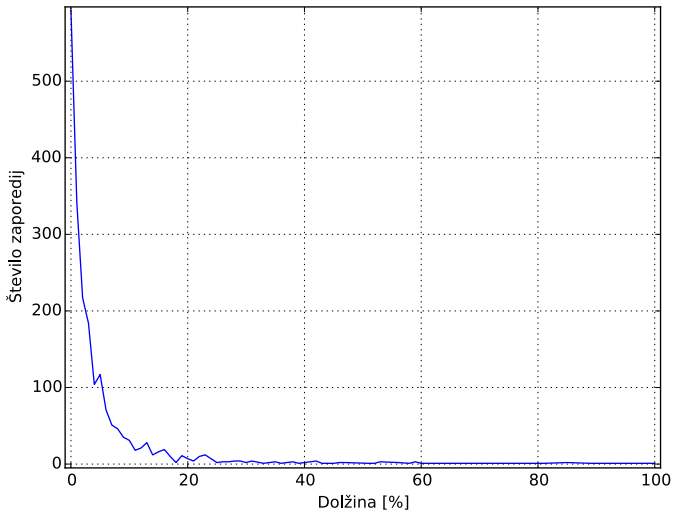
Rezultati

V tem poglavju prikažemo rezultate, ki smo jih dobili pri analizi podatkov, gradnji HMM za posamezne eksperimente, primerjavi posameznih HMM med seboj ter gradnji združenih HMM. Nato jih vrednotimo in skušamo iz njih sklepati o interakcijah med RNA ter proteini in o relacijah med različnimi proteini.

5.1 HMM eksperimentov z dvema stanjema

Po izgradnji splošnih zaporedij genov smo analizirani njihove dolžine ter verjetnosti pojavitve mest vezave znotraj zaporedij. Distribucija zaporedij je prikazana na (Slika 5.1), kjer os x predstavlja dolžino zaporedij v odstotkih, ki smo jih uporabili zaradi lepšega izrisa in os y pa predstavlja število zaporedij, ki imajo določeno dolžino. Najdaljše zaporedje je imelo dolžino 982,532 nukleotidov, kar predstavlja 100 % na grafu. Pri gradnji smo upoštevali vsa zaporedja z dolžino manjšo od 400,000 ali 41 % maksimalne dolžine. Dolga zaporedja smo zavrgli zato, ker občutno podaljšajo izvajanje algoritmov, pri tem pa ne izgubimo veliko reprezentativnih podatkov o vezavah, saj so mesta interakcije zelo redka. Iz grafa je razvidno, da smo upoštevali večino zaporedij, le nekaj ekstremno dolgih smo zavrgli.

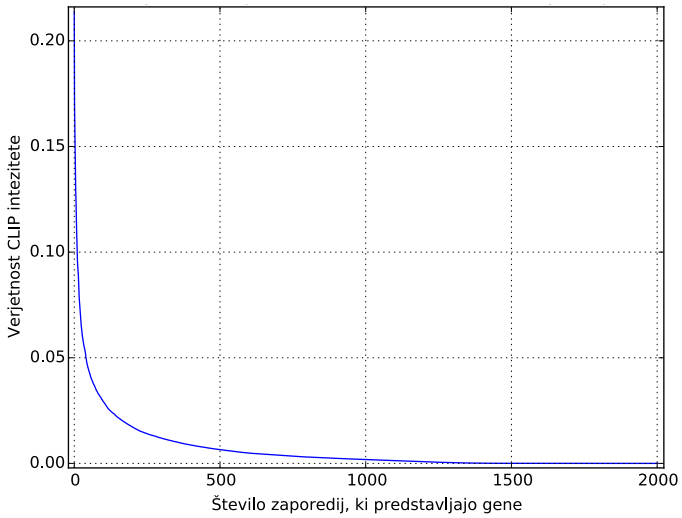
Verjetnosti pojavitve mest interakcij smo izračunali na vseh zaporedjih



Slika 5.1: Dolžine vseh zaporedij, ki smo jih uporabili pri gradnji, učenju in vrednotenju HMM. Dolžina je predstavljena v odstotkih, kjer 100 % predstavlja 982,532 nukleotidov.

za vsak eksperiment posebej. Verjetnosti pojavitev smo nato povprečili čez vse eksperimente in jih prikazali na grafu (Slika 5.2). Os x nam predstavlja posamezna zaporedja, ki so urejena po verjetnosti pojavitve mest interakcij, in sicer od največje do najmanjše. Na osi y so prikazane povprečne verjetnosti pojavitve mest interakcij skozi vseh 32 eksperimentov. Zavrgli smo zaporedja, ki imajo verjetnost pojavitve mest interakcije za določen eksperiment, nižjo od 0.1 ‰. S tem smo se znebili zaporedij, ki imajo zelo majhno število mest interakcij ali pa jih sploh nimajo. Iz grafa lahko sklepamo, da smo v povprečju zavrgli malo zaporedij.

V tabeli (Tabela 5.1) vidimo, število genov oziroma nukleotidov ter njihov odstotek smo uporabili pri gradnji HMM za posamezne eksperimente. To so zaporedja, ki najbolj reprezentativno predstavljajo posamezen eksperiment in so bila dobljena z uporabo meje za maksimalno dolžino zaporedja ter meje za minimalno verjetnost pojavitve mest interakcije tega eksperimenta. Na začetku smo imeli 2,040 zaporedij, ki predstavljajo gene in imajo skupno



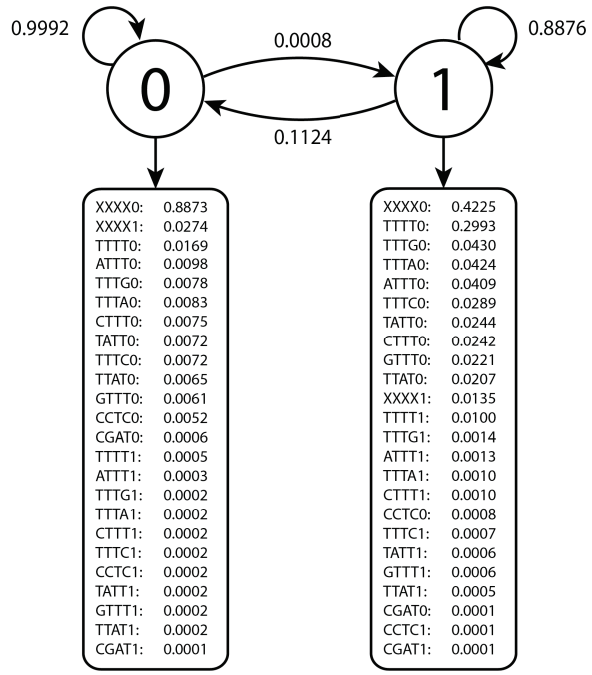
Slika 5.2: Povprečne verjetnosti pojavitve mest interakcij v vseh splošnih zaporedjih 32-tih eksperimentov z dvema stanjema.

dolžino približno 110 milijonov nukleotidov. Vidimo, da smo v določenih zaporedjih izločili veliko število zaporedij. Po podrobnejši analizi smo ugotovili, da nam zmanjša število zaporedij predvsem meja za minimalno verjetnost pojavitve mest interakcije. Mejo smo nato preprosto postavili na 0 in vzeli vsa zaporedja, kjer se je pojavila vsaj ena interakcija, vendar smo tako dobili le malo več izbranih zaporedij. Večino zavrženih zaporedij predstavljajo zaporedja, kjer posamezen eksperiment sploh ni imel mest interakcije. V povprečju smo za gradnjo HMM izbrali 665 genov oziroma 33 % genov, kar predstavlja 36 milijonov nukleotidov oziroma 33 % nukleotidov. Največ genov smo izbrali pri eksperimentu 6 (eIF4AIII 1), kjer smo vzeli 1,556 genov in 80 milijonov nukleotidov. Najmanj genov smo izbrali pri eksperimentu 8 (ELAVL1 1), kjer smo vzeli 248 genov in 11 milijonov nukleotidov.

Iz izbranih zaporedij smo sedaj zgradili HMM za posamezne eksperimente. Iz učnih zaporedij smo pridobili reprezentativne atribute, na podlagi katerih smo zgradili HMM in jih nato vrednotili na testnih zaporedjih. V tabeli (Tabela 5.2) vidimo izbrane atribute in vrednost AUC, ki smo jo dosegli

Št.	Protein	Št. genov	% genov	Št. nukleotidov	% nukleotidov
1	Ago/EIF2C1-4	909	44.56	51,636,823	47.16
2	Ago2-MNase	573	28.09	23,953,854	21.88
3	Ago2 (1)	304	14.90	22,708,241	20.74
4	Ago2 (2)	304	14.90	22,708,241	20.74
5	Ago2 (3)	548	26.86	22,795,827	20.82
6	eIF4AIII (1)	1,556	76.28	80,624,682	73.63
7	eIF4AIII (2)	1,401	68.68	75,134,857	68.61
8	ELAVL1 (1)	660	32.35	41,196,006	37.62
9	ELAVL1 (2)	248	12.16	11,622,498	10.61
10	ELAVL1A	263	12.89	11,949,840	10.91
11	ELAVL1-MNase	601	29.46	25,848,224	23.61
12	ESWR1	471	23.09	34,345,160	31.36
13	FUS	686	33.63	49,784,484	45.46
14	Mut FUS	538	26.37	30,879,196	28.20
15	IGF2BP1-3	790	38.73	44,099,380	40.27
16	hnRNPC (1)	971	47.60	54,436,131	49.71
17	hnRNPC (2)	417	20.44	25,704,067	23.47
18	hnRNPL (2)	554	27.16	19,548,810	17.85
19	hnRNPL (1)	1,251	61.32	62,067,294	56.68
20	hnRNPL-like	586	28.73	16,957,559	15.49
21	MOV10	419	20.54	23,612,231	21.56
22	Nsun2	433	21.23	10,206,824	9.32
23	PUM2	487	23.87	32,590,021	29.76
24	QKI	439	21.52	38,048,735	34.75
25	RBPMS	248	12.16	15,169,217	13.85
26	SFRS1	726	35.59	38,822,148	35.45
27	TAF15	360	17.65	19,900,893	18.17
28	TDP-43	756	37.06	56,027,002	51.16
29	TIA1	657	32.21	35,001,866	31.96
30	TIAL1	827	40.54	46,888,778	42.82
31	U2AF2 (1)	1,100	53.92	57,631,792	52.63
32	U2AF2 (2)	1,188	58.24	60,784,417	55.51
	Povprečje	665	32.58	36,333,909	33.18

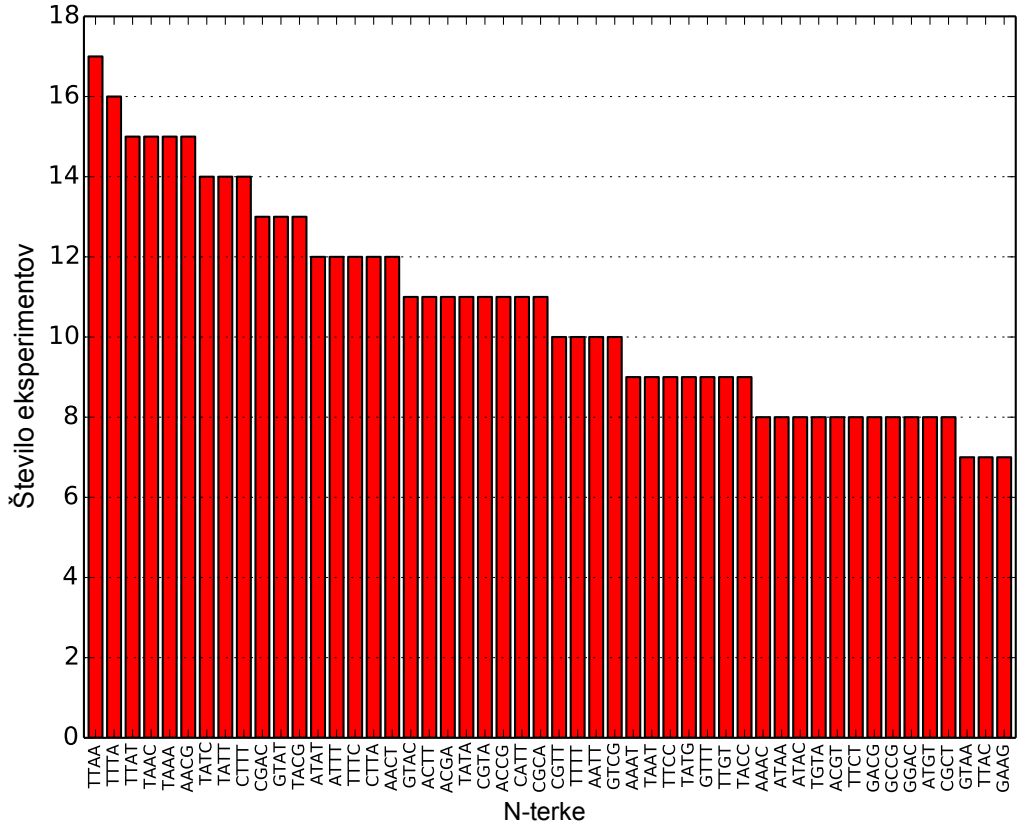
Tabela 5.1: Tabela števila in odstotkov izbranih genov ter nukleotidov iz množice vseh splošnih zaporedij z dvema stanjema genov na prvem kromosomu.



Slika 5.3: Shema prikazuje HMM eksperimenta 16 oziroma hnRNPC (1) s stanjema 0, 1 in emisijam ter pripadajočimi verjetnostmi prehodov med stanji in oddaje emisij. Emisije so urejene po padajoči verjetnosti pojavitve.

z njimi. Povprečni AUC, ki smo ga dosegli pri gradnji posameznih HMM, znaša 0.82. Najvišji AUC smo dosegli pri eksperimentu 17 (hnRNPC 2), in sicer 0.93, najnižji AUC pa pri eksperimentu 6 (eIF4AIII 1), in sicer 0.68. Na sliki (Slika 5.3) vidimo shemo HMM, ki smo ga zgradili za eksperiment 16 (hnRNPC 1).

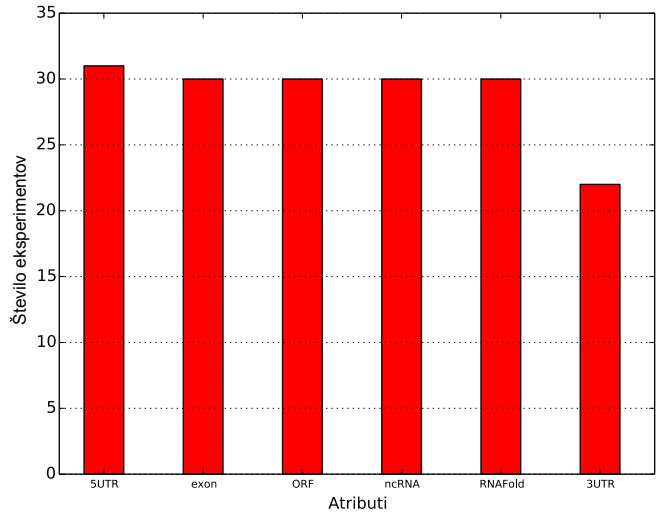
Skupaj smo za vse eksperimente izbrali 158 različnih reprezentativnih n-terk, od tega se jih 116 pojavi pri več kot enem eksperimentu. Na grafu (Slika 5.4) vidimo 50 n-terk, ki so bili reprezentativni pri največ eksperimentih. N-terka TTAA se je pojavila kot reprezentativna pri 17-tih eksperimentih, medtem ko pa se je v več kot 10-tih eksperimentih pojavilo 25 n-terk. Tudi če upoštevamo, da imamo več eksperimentov za posamezen protein, lahko še vedno sklepamo, da se določeno število proteinov veže na enake ali



Slika 5.4: Graf pojavitve 50-tih najbolj pogostih n-terk v HMM posameznih eksperimentov z dvema stanjema.

podobne motive zaporedja nukleotidov. Glede na to, da smo pri določanju reprezentativnih atributov upoštevali tudi okolico okoli mest vezave, nam n-terke lahko predstavljajo tudi motive v okolicah mest vezave. Iz grafa vidimo, da imamo veliko n-terk, ki se med seboj razlikujejo le v nukleotidu ali dveh. Iz tega sklepamo, da določena n-terka ponavadi ne predstavlja absolutnega motiva, saj je lahko določen nukleotid v motivu poljuben. S tem se poveča zanesljivost vezavnih mest na RNA. Lahko pa tudi predstavljajo motiv, ki je krajši ali pa celo daljši od dolžine n-terk.

Za ostale reprezentativne atribute smo uporabili funkcijske podenote in



Slika 5.5: Graf pojavitve ostalih atributov v HMM posameznih eksperimentov z dvema stanjema.

sekundarno strukturo genov. Na grafu (Slika 5.5) vidimo pojavitve posameznih atributov pri posameznih eksperimentih. Atribut 5UTR je bil reprezentativen pri 31-tih eksperimentih, atributi: exon, ORF, ncRNA in RNAFold pri 30-tih eksperimentih, atribut 3UTR pri 22-tih eksperimentih, atribut intron pa se ni pojavil pri nobenem eksperimentu. Vidimo, da se skoraj vsi proteini vežejo v okolici ali celo na sam ekson oziroma njegove podenote. Ekson in njegove podenote se ohranijo med postopkom transkripcije. Podenote ncRNA so tudi pogosto reprezentativni atribut, čeprav se ob sami transkripciji zavržejo. Lahko sklepamo, da se nahajajo v okolici mest vezave in vsebujejo motive, ki jih potrebujejo proteini za vezavo v okolici ali pa na samo ncRNA podenoto. Intron ni bil nikoli izbran za reprezentativni atribut, saj predstavlja podenoto gena, ki se med samo transkripcijo zavrže. Na grafu (Slika 5.6) vidimo prehajanje vrednosti posameznih atributov ter dejanskih in napovedanih mest vezave.

Št.	Protein	AUC	Izbrane n-terke	Izbrani ostali atributi
1	Ago/EIF2C1-4	0.85	AACG, ACCG, ACGA, ACGT, AGCG, ATCG, CCGA, CCGC, CGAA, CGAC, CGCA, CGCG, CGCT, CGGA, CGGC, CGGT, CGTC, CGTT, CTTC, GCCG, GGAC,	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD

Št.	Protein	AUC	Izbrane n-terke	Izbrani ostali atributi
			GTCG, TACG, TCCG, TCGA, TCGG, TCGT, TGCG, TTCG	
2	Ago2-MNase	0.85	AACG, ACCG, ACGA, CCCC, CCCT, CCGA, CCGC, CCGG, CGAA, CGAC, CGCG, CGCT, CGGA, CGGC, GACG, GCCG, GCGG, GGAC, GTCG, TACG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
3	Ago2 (1)	0.90	AACG, AAGA, AATG, ACCG, ACGA, ACGT, ATCG, ATGT, CCGA, CGAC, CGAG, CGCA, CGCT, CGTA, GAAG, GACG, GATG, GTAA, GTAC, GTAT, GTCG, GTTA, TAAA, TACG, TATG, TCGA, TCGT, TGTA, TTCG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
4	Ago2 (2)	0.89	AACG, AAGA, AATG, ACCG, ACGA, ACGG, ACGT, AGCG, ATCG, ATGT, CCGA, CGAA, CGAC, CGAG, CGCA, CGCT, CGGT, CGTC, GAAG, GACC, GACG, GATG, GCCG, GTAA, GTAC, GTAT, GTCG, TACG, TATG, TCGA, TCGT, TGTA, TTCG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
5	Ago2 (3)	0.78	AAAC, AACG, AACT, AAGA, AATG, ACCG, ACGA, ACGT, ACTT, AGAA, CAAC, CCCC, CCCT, CCTT, CGAA, CGAC, CGTA, CGTT, CTTC, CTTT, GAAG, GGAC, GTAC, TAAA, TAAC, TACC, TACG, TTAA, TTCC	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
6	eIF4AIII (1)	0.68	AACG, ACCG, ACGG, AGCG, AGGG, CCGC, CCGG, CGAC, CGCA, CGCG, CGCT, CGGA, CGGC, CGGG, CGGT, GAAG, GACG, GCCG, GCGC, GCGG, GGAC, GGGC, GGGG, GTCG, TGCG	EXON, 5UTR, ORF, ncRNA, RNAFOLD
7	eIF4AIII (2)	0.73	AACG, ACCG, ACGA, ACGG, AGCG, CCGC, CCGG, CGAC, CGCA, CGCG, CGCT, CGGA, CGGC, CGGG, CGGT, GAAG, GACG, GCCG, GCGC, GCGG, GGAC, GGCG, GGGC, GGGG, GTCG, TCGG, TGCG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
8	ELAVL1 (1)	0.83	AATT, ACTT, ATAT, ATTT, CATT, CTTA, CTTT, GTAT, GTTT, TAAA, TAAT, TATA, TATC, TATT, TCTT, TGTT, TTAA, TTAC, TTAT, TTCT, TTGT, TTTA, TTTC, TTTG, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
9	ELAVL1 (2)	0.92	AAAT, AACT, ATAT, ATTT, CATT, CCTT, CGTA, CTTT, GTTT, TAAA, TAAC, TACC, TATC, TATT, TCGT, TCTT, TGTA, TGTT, TTAA, TTAT, TTCC, TTCT, TTGT, TTTA, TTTC, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
10	ELAVL1A	0.92	AAAT, AATT, ATAT, ATTC, ATTT, CATT, CCTT, CGTA, CTAT, CTTA, CTTC, CTTT, GTTT, TAAA, TAAC, TACC, TATA, TATC, TATT, TCAT, TCTA, TCTT, TTAA, TTAC, TTAT, TTCC, TTCT, TTGT, TTTA, TTTC, TTTG, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
11	ELAVL1-MNase	0.72	AACG, ACCG, ACGA, ACGG, AGCG, CCCC, CCGA, CCGC, CCGG, CGAC, CGCA, CGCG, CGCT, CGGA, CGGC, CGGG, CGTC, GACG, GCCG, GCGC, GCGG, GGAC, GTCG, TACG, TCCG, TGCG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
12	ESWR1	0.79	AAAC, AAAT, AACA, AACT, AATA, AATT, ACAA, ACTT, ATAA, ATAC, ATAT, ATTA, ATTC, CATT, CGTT, CTTA, CTTT, TAAA, TAAC, TAAT, TACC, TACG, TACT, TATA, TATC, TATT, TTAA, TTAC, TTAT, TTTA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
13	FUS	0.78	AAAC, AAAT, AACT, AATA, AATT, ACTT, ATAA, ATAC, ATAT, ATTA, ATTT, CATA, CATT, CTAT, CTTA, GTAT, TAAA, TAAC, TAAT, TACT, TATA, TATC, TATT, TCAT, TTAA, TTAC, TTAT, TTTA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
14	Mut FUS	0.84	AAAC, AAAT, AACG, AACT, AATA, AATG, AATT, ACTT, ATAA, ATAC, ATAT, ATGT, ATTT, CATT, CGTA, CTTA, CTTT, GTAT, TAAA, TAAC, TAAT, TACT, TATA, TATC, TATT, TCTA, TTAA, TTAT, TTTA, TTTC	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
15	IGF2BP1-3	0.86	AACG, AACT, ACCG, ACGA, ACGT, CAAC, CAGA, CATC, CCAA, CCCC, CCCT, CCTT, CGAA, CGAC, CGCA, CGTC, CGTT, CTTC, GAAG, GCCC, GGAC, GTAC, GTCC, TACC, TACG, TCCA, TTCC, TTCG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
16	hnRNPC (1)	0.91	ATTT, CCTC, CGAT, CTTT, GTTT, TATT, TTAT,	5UTR

Št.	Protein	AUC	Izbrane n-terke	Izbrani ostali atributi
			TTTA, TTTC, TTTG, TTTT	
17	hnRNPC (2)	0.93	CGAT, CTCG, GCGA, TTTT	EXON, 5UTR, ORF, ncRNA, RNAFOLD
18	hnRNPL (1)	0.69	AAAC, AACG, AACT, ACAC, ACAT, ACGA, ACGT, ATAC, ATCA, ATCG, ATTC, CAAC, CACA, CATA, CATC, CATT, CCCC, CCCT, CCTT, CGAC, CGTA, CGTT, CTTC, CTTT, GTAC, TAAA, TAAC, TACC, TATC, TCAT, TTAA, TTCA, TTCC, TTCG, TTTC	EXON, 5UTR, ORF, ncRNA, RNAFOLD
19	hnRNPL (2)	0.69	AAAC, AAAT, AACA, AACG, ACAC, ACAT, ATAA, ATAC, ATAT, CACA, CATA, CATT, CGTA, CGTT, CTTA, GTAC, TAAA, TAAC, TACA, TACC, TACG, TATA, TATC, TTAA, TTAC, TTAT, TTCA, TTTA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
20	hnRNPL-like	0.70	AACG, AACT, ACAC, ACGA, ATCG, CACA, CATC, CCCC, CCCT, CCTT, CGAC, CGCA, CGTA, CGTT, CTCT, CTTC, CTTT, GTCC, GTCG, TACC, TATC, TCCA, TCCC, TCCT, TCGT, TCTA, TTCC, TTCT, TTTC	EXON, 5UTR, ORF, ncRNA, RNAFOLD
21	MOV10	0.90	AAAC, AAAG, AAAT, AACT, AATA, AATG, AATT, ACTT, ATAA, ATAT, ATGT, ATTT, CTTA, GAAA, GATA, GTAA, GTAC, GTAT, GTTA, TAAA, TAAC, TAAG, TACT, TATA, TATG, TATT, TGTA, TTAA, TTTA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
22	Nsun2	0.76	ACCG, AGCG, CCCG, CCGA, CCGC, CCGG, CGAC, CGAG, CGCA, CGCC, CGCG, CGCT, CGGA, CGGC, CGGG, CGGT, CGTC, GACG, GCCG, GCGA, GCGC, GCGG, GGCG, GTCG, TCCG, TCGC, TCGG, TTCG	EXON, 5UTR, ORF, ncRNA, RNAFOLD
23	PUM2	0.89	AAAT, AATA, AATT, ACAT, ATAA, ATAC, ATAT, ATGT, ATTA, ATTT, CATA, GTAA, GTAC, GTAT, TAAA, TAAT, TACA, TATA, TATG, TATT, TGTA, TTAA, TTAT, TTGT, TTTA, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
24	QKI	0.93	AACA, AACC, AACT, AATA, AATC, ACAA, ACAT, ACTA, ACTT, ATAA, ATAC, ATTA, CATA, CATT, CTAA, CTTA, TAAA, TAAC, TAAT, TACC, TACT, TATT, TCTA, TTAA, TTAC, TTAT, TTTA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
25	RBPMS	0.84	ACAC, ACCC, ACTA, ACTC, ACTT, ATCA, CACA, CACC, CACT, CATC, CCAC, CCCC, CCCT, CCTC, CCTT, CGCA, CGTC, CTCT, CTTC, CTTT, GTAC, GTCC, TCAC, TCCA, TCCC, TCCG, TCGT, TTCA, TTCC, TTTC	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
26	SFRS1	0.79	AACG, AAGA, ACCG, ACGA, ACGG, AGCG, CCGA, CCGC, CCGG, CGAA, CGAC, CGCA, CGCG, CGGA, CGGC, CGGG, GAAG, GACG, GCCG, GCGC, GCGG, GGAC, GTCG, TACG, TCGA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
27	TAF15	0.75	AAAC, AAAT, AACG, AACT, AATG, AATT, ACGT, ACTT, ATAA, ATAC, ATTC, CGTA, CGTT, CTAA, CTTA, GAAA, GGTA, GTAA, GTAT, TAAA, TAAC, TAAT, TACG, TACT, TATA, TATC, TGAA, TTAA, TTAT, TTTA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
28	TDP-43	0.78	AATG, ACGT, ATGA, ATGT, CATG, CGTG, CTGT, GAAT, GAGT, GCAT, GCGT, GTAT, GTGA, GTGC, GTGT, GTTG, GTTT, TATG, TCTG, TGAA, TGCA, TGCG, TGTA, TGTC, TGTG, TGTT, TTGT	ncRNA
29	TIA1	0.83	ACTT, ATAT, ATTT, CGTA, CGTT, CTAT, CTTT, GTAA, GTAC, GTAT, GTTT, TAAA, TAAC, TACG , TATA, TATC, TATG, TATT, TCTT, TGTA, TGTT, TTAA, TTAT, TTCT, TTGT, TTTA, TTTC, TTTG, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
30	TIAL1	0.77	AACT, ACTT, ATTT, CGTA, CTTA, CTTT, GGTA, GTAA, GTAC, GTAT, GTTT, TAAC, TACG, TATC, TATG, TATT, TCGT, TCTT, TGTA, TGTT, TTAA, TTAT, TTCT, TTGT, TTTA, TTTC, TTTG, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
31	U2AF2 (1)	0.86	AATT, ATAT, ATGT, ATTT, CATT, CTTA, CTTT,	EXON, 5UTR,

Št.	Protein	AUC	Izbrane n-terke	Izbrani ostali atributi
			GTAT, GTTA, GTTT, TAAC, TAAT, TATA, TATC, TATG, TATT, TCTT, TGTT, TTAA, TTAC, TTAT, TTCC, TTCT, TTGT, TTTA, TTTC, TTTT	ORF, RNAFOLD
32	U2AF2 (2)	0.84	AATT, ATAT, ATGT, ATTT, CATT, CGTT, CTTA, CTTT, GTAT, GTTT, TAAC, TAAT, TATC, TATG, TATT, TCTT, TGTT, TTAA, TTAT, TTCC, TTCT, TTGT, TTTA, TTTC, TTTG, TTTT	EXON, 5UTR, ORF, 3UTR,

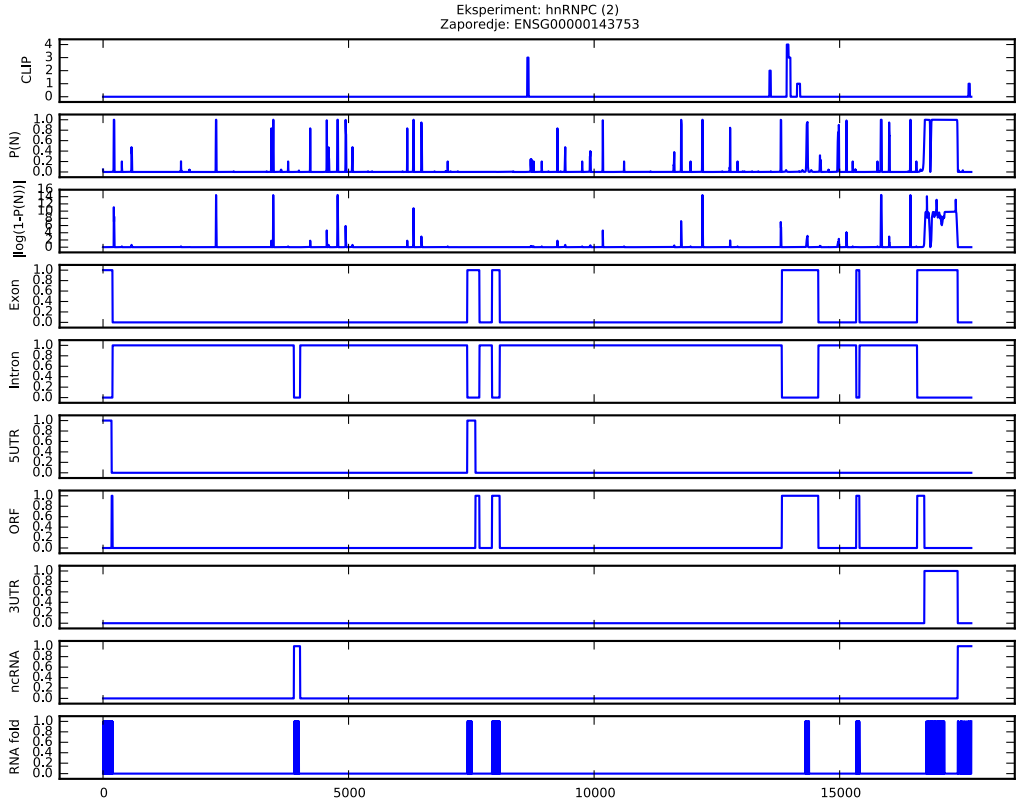
Tabela 5.2: Tabela izbranih atributov pri gradnji in vrednosti AUC pri vrednotenju posameznih HMM z dvema stanjema.

5.2 HMM eksperimentov z več stanji

Postopek gradnje osnovnih HMM za posamezne eksperimente s tremi stanji je enak gradnji HMM s samo dvema stanjema, le da smo za vrednotenje namesto krivulj ROC uporabili metodo MAE. Po izgradnji splošnih zaporedij genov smo analizirani njihove dolžine ter verjetnosti pojavitve mest vezave znotraj zaporedij. Dolžine zaporedij so enake kot pri gradnji HMM z dvema stanjema in so predstavljena z grafom (Slika 5.1). Prav tako je graf verjetnosti pojavitve mest interakcij (Slika 5.7) zelo podoben grafu HMM z dvema stanjema (Slika 5.2). Tako, da smo uporabili enake meje pri gradnji HMM.

V tabeli (Tabela 5.3) vidimo, število genov oziroma nukleotidov ter njihov odstotek smo uporabili pri gradnji HMM za posamezne eksperimente. Vidimo, da se veliko ne razlikuje od tabele HMM z dvema stanjema (Tabela 5.7). V povprečju smo izgubili manj kot en odstotek zaporedij pri gradnji HMM s tremi stanji. Ugotovimo, da se število izbranih genov pri določenih eksperimentih ni spremenilo. Devet eksperimentov ima intenzitete mest interakcij podane le z uniformnimi vrednostmi, ki se nahajajo znotraj stanja 1 ali 2. Ti eksperimenti so: 3, 4, 12, 13, 14, 15, 21, 26, 27.

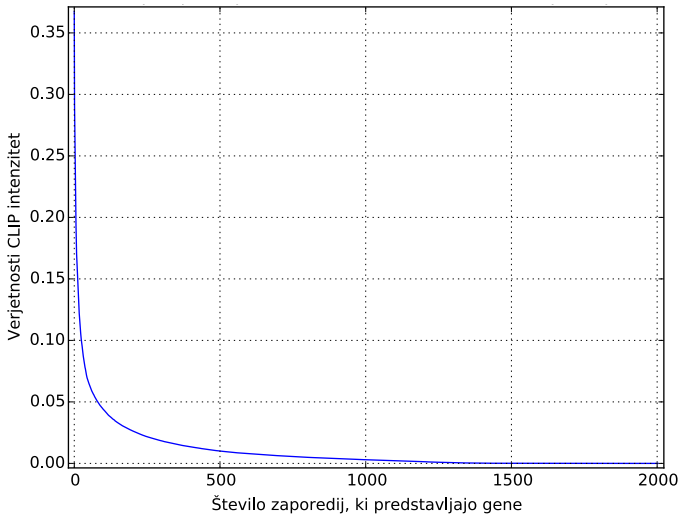
Iz izbranih zaporedij smo sedaj zgradili HMM s tremi stanji za posamezne eksperimente. V tabeli (Tabela 5.4) vidimo izbrane atribute in dosežene vrednost MAE. Povprečni MAE, ki smo ga dosegli za stanje 1, znaša 0.067, za stanje 2, pa 0.021. Najnižji MAE stanja 1 znaša 0.001 in smo ga dosegli pri eksperimentih 18, 20, 22, 25, 28, 29. Najnižji MAE stanja 2 znaša 0.001 in smo ga dosegli pri eksperimentih 1, 5, 8, 9, 10, 18, 20, 22, 23, 24.



Slika 5.6: Graf prikazuje dejanske, napovedane in logaritem napovedanih vrednosti CLIP ter ostale atribute podenot genov in sekundarne zgradbe za posamezno zaporedje posameznega eksperimenta.

Št.	Protein	Št. genov	% genov	Št. nukleotidov	% nukleotidov
1	Ago/EIF2C1-4	899	44.09	50,867,979	46.45
2	Ago2-MNase	551	27.01	23,338,303	21.31
3	Ago2 (1)	304	14.90	22,708,241	20.74
4	Ago2 (2)	304	14.90	22,708,241	20.74
5	Ago2 (3)	528	25.88	22,194,971	20.82
6	eIF4AIII (1)	1,550	75.98	79,959,880	73.02
7	eIF4AIII (2)	1,397	68.48	74,663,353	68.18
8	ELAVL1 (1)	650	31.86	40,766,126	37.23
9	ELAVL1 (2)	238	11.67	11,392,309	10.40
10	ELAVL1A	253	12.40	11,718,940	10.70
11	ELAVL1-MNase	579	28.38	25,232,673	23.04
12	ESWR1	471	23.09	34,345,160	31.36
13	FUS	686	33.63	49,784,484	45.46
14	Mut FUS	538	26.37	30,879,196	28.20
15	IGF2BP1-3	790	38.73	44,099,380	40.27
16	hnRNPC (1)	939	46.03	53,607,519	48.95
17	hnRNPC (2)	381	18.67	23,481,884	21.44
19	hnRNPL (1)	553	27.11	19,503,613	17.81
18	hnRNPL (2)	1,249	61.23	61,818,920	56.45
20	hnRNPL-like	585	28.67	16,876,945	15.41
21	MOV10	419	20.54	23,612,231	21.56
22	Nsun2	432	21.23	10,150,751	9.27
23	PUM2	480	23.53	32,228,740	29.43
24	QKI	430	21.07	37,539,926	34.28
25	RBPMS	218	10.69	13,089,827	11.95
26	SFRS1	726	35.59	38,822,148	35.45
27	TAF15	360	17.65	19,900,893	18.17
28	TDP-43	731	35.83	55,141,256	50.36
29	TIA1	601	29.46	31,896,472	29.13
30	TIAL1	799	39.17	44,875,236	40.98
31	U2AF2 (1)	1,083	53.09	56,055,413	51.19
32	U2AF2 (2)	1,179	57.79	59,606,371	54.43
	Povprečje	653	32.02	35,714,605	32.61

Tabela 5.3: Tabela števila in odstotkov izbranih genov ter nukleotidov iz množice vseh splošnih zaporedij s tremi stanji genov na prvem kromosomu.

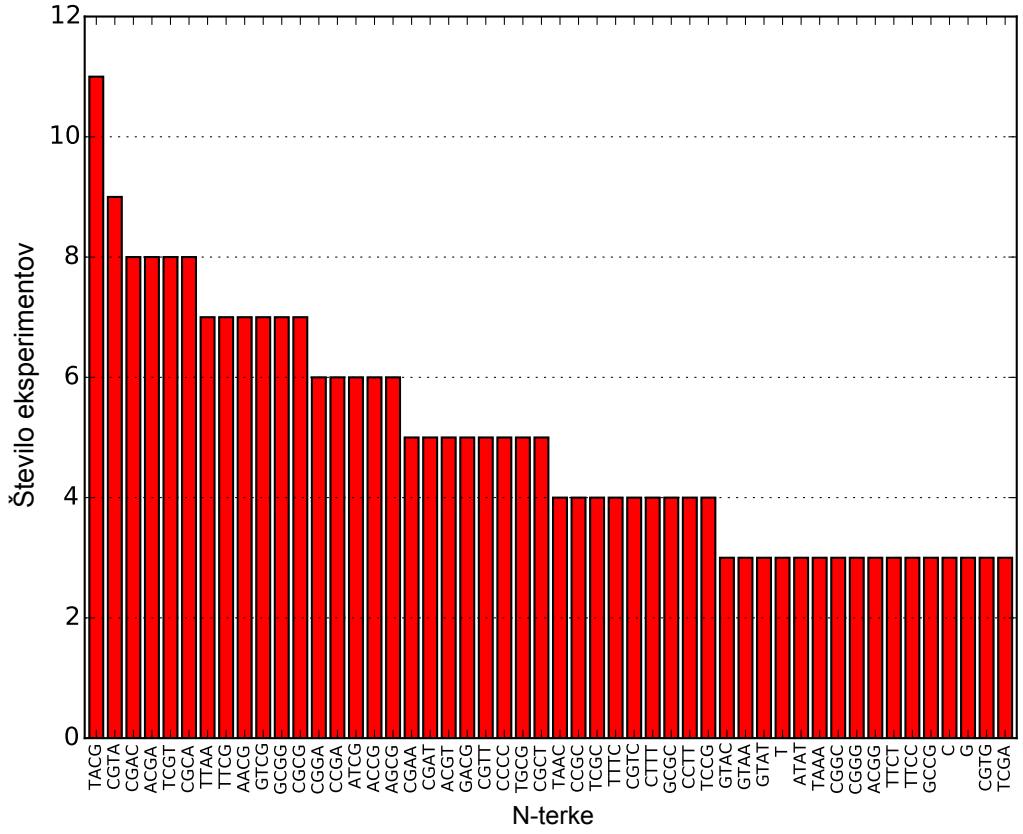


Slika 5.7: Povprečne verjetnosti pojavitve mest interakcij v vseh splošnih zaporedjih 32-tih eksperimentov s tremi stanji.

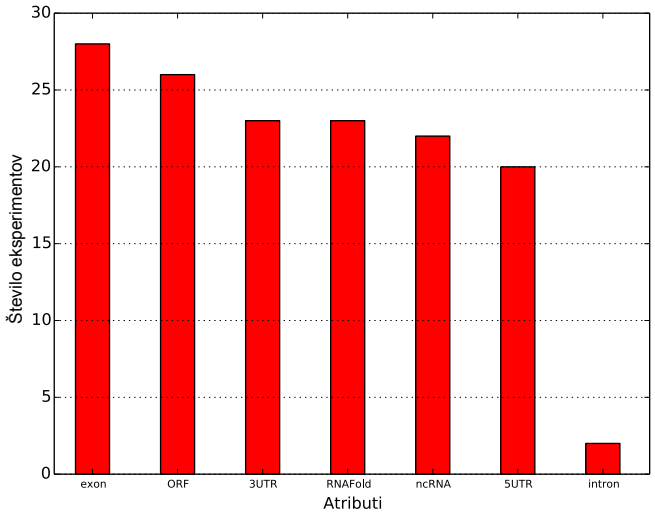
Najvišji MAE stanja 1 znaša 0.412 in smo ga dosegli pri eksperimentu 1 (Ago/EIF2C1-4). Najvišji MAE stanja 2 znaša 0.098 in smo ga dosegli pri eksperimentu 30 (TIAL1).

Skupaj smo za vse eksperimente izbrali 142 različnih reprezentativnih n-terk in nukleotidov, od tega se jih 85 pojavi pri več kot enem eksperimentu. Na grafu (Slika 5.8) vidimo 50 n-terk in nukleotidov, ki so bili reprezentativni pri največ eksperimentih. N-terka TACG se je pojavila kot reprezentativna pri 11-tih eksperimentih, medtem ko pa se je v več kot 4-tih eksperimentih pojavilo 25 n-terk. Najbolj reprezentativna n-terka dvostanjskih HMM je TTAA in se tukaj pojavi kot reprezentativna v 7 eksperimentih. Če primerjamo izbrane reprezentativne n-terk, se jih 114 pojavi pri tri in dvostanjskih HMM hkrati. Večina izbranih n-terk je torej enaka, vendar se razlikujejo po številu eksperimentov, ki jih reprezentativno predstavljajo.

Za ostale reprezentativne atribute smo uporabili funkcijske podenote in sekundarno strukturo genov. Na grafu (Slika 5.9) vidimo pojavitve posameznih atributov pri posameznih eksperimentih. Atribut ekson je bil repre-



Slika 5.8: Graf pojavitve 50-tih najbolj pogostih n-terk v HMM posameznih eksperimentov s tremi stanji.



Slika 5.9: Graf pojavitve ostalih atributov v HMM posameznih eksperimentov s tremi stanji.

zentativen pri 31 eksperimentih, atribut ORF pri 26-tih, atributa 3UTR in RNAFOLD pri 23-tih, atribut ncRNA pri 22-tih, atribut 5UTR pri 20-tih in atribut intron pri 2-eh. Vidimo, da se skoraj vsi proteini vežejo v okolici ali celo na sam ekson oziroma njegove podenote. Če primerjamo izbiro reprezentativnih atributov med tri in dvostanjskimi HMM, vidimo, da se vsi atributi razen introna zelo pogosto pojavijo kot reprezentativni, razlikujejo se le v številu eksperimentov, ki jih reprezentativno predstavijo.

Št.	Protein	MAE 1	MAE 2	Izbrane n-terke	Izbrani ostali atributi
1	Ago/EIF2C1-4	0.411	0.001	AACG, ACCG, ACGA, ACGG, ACGT, ATCG, CCCC, CCTT, CGAA, CGAT, CGCA, CGCG, ACTT, AGCG,CGCT, CGGA, CGTA, CGTC, CGTT, CTTT, GACG, GGAC, GTAC, GTCG, TACC, TACG, TATG, TCCG, TCGC, TCGG, TCGT, TTCG	3UTR
2	Ago2-MNase	0.058	0.007	AACG, ACCG, ACGA, AGCG, ATCG, CCGA, CCGC, CCGT, CGAA, CGAC, CGAG, CGCA, CGCG, CGCT, CGGA, CGGG, CGGT, CGTA, CGTC, CGTG, GACG, GCCG, GCGA, GCGC, GCGG, GGCG, GTCG, TCCG, TCGT, TGCG, TTCG	EXON, 3UTR,
3	Ago2 (1)	0.064		ACCG, ACGA, CGAT, CGCA, GACG, TACG	EXON, ORF 3UTR, ncRNA, RNAFOLD
4	Ago2 (2)	0.120		CGAC	EXON, 5UTR,

Št.	Protein	MAE 1	MAE 2	Izbrane n-terke	Izbrani ostali atributi
5	Ago2 (3)	0.141	0.001	ACCG, ACGA, ACGT, ATCG, CCGA, CCGC, CGAC, CGAT, CGCA, CGTA, CGTC, CTTC, GAAC, GAAG, GTCG, TAAC, TACG, TCGT, TGAA, TGCG, TTCG	ORF, 3UTR, ncRNA, RNAFOLD EXON, ORF, ncRNA, RNAFOLD
6	eIF4AIII (1)	0.026	0.011	AGCG, CGAC, GCGG	EXON, ORF, ncRNA,
7	eIF4AIII (2)	0.019	0.032	GCGG, CGAC	ORF
8	ELAVL1 (1)	0.061	0.001	AATT, ACTT, ATAT, ATTT, CATT, CTTA, CTTT, GTAT, GTTT, TAAA, TAAT, TATA, TATC, TATT, TCTT, TGTA, TGTT, TTAA, TTAT, TTCT, TTGT, TTTA, TTTC, TTTG	EXON, RNAFOLD
9	ELAVL1 (2)	0.111	0.001	TCGT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
10	ELAVL1A	0.093	0.001	TACG, TCGT, TTTT	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
11	ELAVL1-MNase	0.102	0.003	ACCG, ACGA, AGCG, ATCG, CCGA, CCGC, CCGG, CGAA, CGCA, CGCG, CGCT, CGGA, CGGC, CGGG,CGGT, CGTA, CGTC, GACG, GCCG, GCGA, GCGC,GCGG, GGCG, GGGG, GTCG, TCCG, TCGG, TCGT, TGCG, TTCG	EXON, 5UTR, ORF, 3UTR, ncRNA
12	ESWR1	0.107		TTAA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
13	FUS	0.087		TTAA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
14	Mut FUS	0.138		TTAA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
15	IGF2BP1-3	0.119		AACG	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
16	hnRNPC (1)	0.002	0.045	A, C, T, G	5UTR, 3UTR ncRNA, RNAFOLD
17	hnRNPC (2)	0.006	0.001	A, C, T, G	EXON, ORF, ncRNA, RNAFOLD
18	hnRNPL (1)	0.001	0.001	AACG, AAGG, ACAC, ACAT, ACGG, ACGT, ATAC, ATAT, ATCC, ATCG, CAAC, CAAG, CACA, CATA, CATC, CATT, CCAC, CCCC, CCCG, CCCT, CCGA, CCGC, CCTT, CGAC, CGAT, CGCA, CGCT, CGGC, CGGG, CGTA, CGTT, CTTC, GATC, GCCG, GCGG, GCGT, GGAA, GGTA, GTAT, TACG, TATC, TATG, TCAT, TCCA, TCGA, TCGC, TCGT, TCTC, TGCG, TTCA, TTCC, TTCG, TTTC	EXON, INTRON, 5UTR, ORF,
19	hnRNPL (2)	0.003	0.001	AACG, ACAC, ACAT, ACCC, ACCT, ACGA, ACGC, ACGT, AGCG, ATAC, ATAT, CAAT, CACA, CACC, CATA, CATC, CATT, CCAC, CCCA, CCCC, CCCT, CCTT, CGAA, CGAC, CGCA, CGCG, CGGA, CGTA, CGTT, CTTA, GCGC, GCGG, GGAC, GTCG, TAAA, TAAC, TACA, TACG, TATA, TCAA,	EXON,ORF, 3UTR, ncRNA, RNAFOLD

Št.	Protein	MAE 1	MAE 2	Izbrane n-terke	Izbrani ostali atributi
20	hnRNPL-like	0.001	0.001	TCAT, TCGC, TTAA, TTCC AACG, AAGT, ACAC, ACCC, ACCG, ACGA, ACGC, AGCG, ATCG, CACA, CACG, CATC, CCCC, CCCT, CCGA, CCGT, CCTT, CGAA, CGAC, CGAG, CGAT, CGCA, CGGA, CGTA, CGTG, CGTT, CTAC, CTCG, CTCT, CTTC, CTTT, GCAT, GCGG, GTAC, GTCC, GTCG, TACG, TCCA, TCCG, TCCT, TCGA, TCGC, TCGG, TCGT, TCTC, TGCG, TGCT, TTCC, TTCG, TTCT, TTTC	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
21	MOV10	0.110		TAAA, TTAA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
22	Nsun2	0.001	0.001	AACG, ACGA, CGCG, CGCT, CGGC, CGTT, GCGC, TACG, TCGA, TTCG	EXON, 5UTR, ORF, 3UTR,
23	PUM2	0.031	0.001	TATA, TGTA	EXON, ORF
24	QKI	0.037	0.001	ACTA, CGCG, GTCG, TAAC	EXON, ORF, 3UTR, ncRNA, RNAFOLD
25	RBPMS	0.001	0.096	AAGA, ACGG, ACGT, ACTG, AGAG, AGAT, AGTG, ATGA, ATGT, CACT, CCCC, CCGA, CGGA, CGTG, CTAC, CTAG, GACG, GATG, GGCT, GTAA, GTGA, GTGG, TACG, TAGT, TGAT, TGGC, TGGT	EXON, INTRON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
26	SFRS1	0.118		CGCG	EXON, 5UTR, ORF, ncRNA RNAFOLD
27	TAF15	0.127		AAAC, CGTA, TTAA	EXON, 5UTR, ORF, 3UTR, ncRNA, RNAFOLD
28	TDP-43	0.001	0.003	GTGT	EXON, 5UTR ncRNA, RNAFOLD
29	TIA1	0.001	0.055	CGTA, GGTA, GTAA, GTAC, TACG, TATT, TTAT, TTTT	EXON, ORF, 3UTR, RNAFOLD
30	TIAL1	0.002	0.098	CTTT, GGTA, GTAA, GTAT, TACG, TATT, TGTT, TTCT, TTTC, TTTT	EXON, 5UTR, ORF, 3UTR, RNAFOLD
31	U2AF2 (1)	0.009	0.060	TAAC	EXON, 5UTR, ORF, ncRNA, RNAFOLD
32	U2AF2 (2)	0.022	0.028	A, C, T, G	3UTR

Tabela 5.4: Tabela izbranih atributov pri gradnji in MAE vrednosti pri vrednotenju posameznih HMM.

Ker se vrednosti MAE in AUC ne da direktno primerjati, smo zgradili posamezne HMM z dvema in s tremi stanji in smo jih vrednotili z metodo MAE. Rezultate vidimo v tabeli (Tabela 5.5). Povprečni MAE HMM z dvema stanjema je 0.099, ki je rahlo višji od povprečnih MAE HMM s tremi stanji, ki znašata 0.067 za stanje 1 in 0.021 za stanje 2. Vrednosti pri posameznih eksperimentih so v večini primerov vseeno primerljive.

Pri gradnji HMM s tremi stanji, smo ugotovili MAE mera ni najbolj

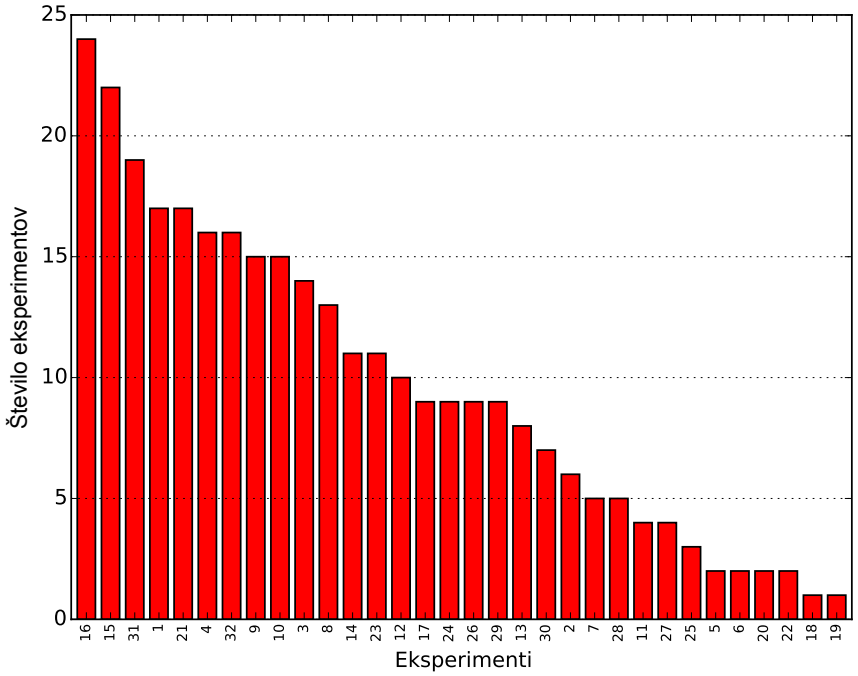
Št.	Protein	HMM z dvema stanjema	HMM s tremi stanji	
		MAE stanja 1	MAE stanja 1	MAE stanja 2
1	Ago/EIF2C1-4	0.149	0.411	0.001
2	Ago2-MNase	0.142	0.058	0.007
3	Ago2 (1)	0.111	0.064	
4	Ago2 (2)	0.095	0.120	
5	Ago2 (3)	0.129	0.141	0.001
6	eIF4AIII (1)	0.125	0.026	0.011
7	eIF4AIII (2)	0.099	0.019	0.032
8	ELAVL1 (1)	0.110	0.061	0.001
9	ELAVL1 (2)	0.168	0.111	0.001
10	ELAVL1A	0.110	0.093	0.001
11	ELAVL1-MNase	0.155	0.102	0.003
12	ESWR1	0.125	0.107	
13	FUS	0.108	0.087	
14	Mut FUS	0.124	0.138	
15	IGF2BP1-3	0.146	0.119	
16	hnRNPC (1)	0.077	0.002	0.045
17	hnRNPC (2)	0.077	0.006	0.001
18	hnRNPL (1)	0.001	0.001	0.001
19	hnRNPL (2)	0.003	0.003	0.001
20	hnRNPL-like	0.001	0.001	0.001
21	MOV10	0.121	0.110	
22	Nsun2	0.001	0.001	0.001
23	PUM2	0.162	0.032	0.001
24	QKI	0.040	0.037	0.001
25	RBPMS	0.110	0.001	0.096
26	SFRS1	0.140	0.118	
27	TAF15	0.144	0.127	
28	TDP-43	0.014	0.001	0.003
29	TIA1	0.084	0.001	0.055
30	TIAL1	0.132	0.002	0.098
31	U2AF2 (1)	0.050	0.009	0.060
32	U2AF2 (2)	0.132	0.022	0.028
	Povprečje	0.099	0.067	0.021

Tabela 5.5: Tabela z vrednostmi MAE napovedi za stanje 1 pri HMM z dvema stanjema ter stanji 1 in 2 pri HMM s tremi stanji.

primerna za vrednotenje naših podatkov, saj je pogosto čisto blizu vrednosti 1, kar pomeni, da je pojavitev mest interakcije tako zelo majhna, da ne vpliva dovolj na vrednost. Ta problem se je pojavil tudi pri določanju reprezentativnih zaporedij. Prav tako se je izbor reprezentativnih atributov in motivov poslabšal, saj smo redka mesta interakcij še nadaljnjo razdelili na dve podskupini, kar še dodatno zmanjša število mest iz katerih izluščimo reprezentativne atribute. Zaradi teh vzrokov in ker za 9 eksperimentov lahko zgradimo le HMM z dvema stanjema, smo se odločili, da za združevanje HMM uporabimo enostavnejše in bolj zanesljive modele z dvema stanjema.

5.3 Združevanje HMM

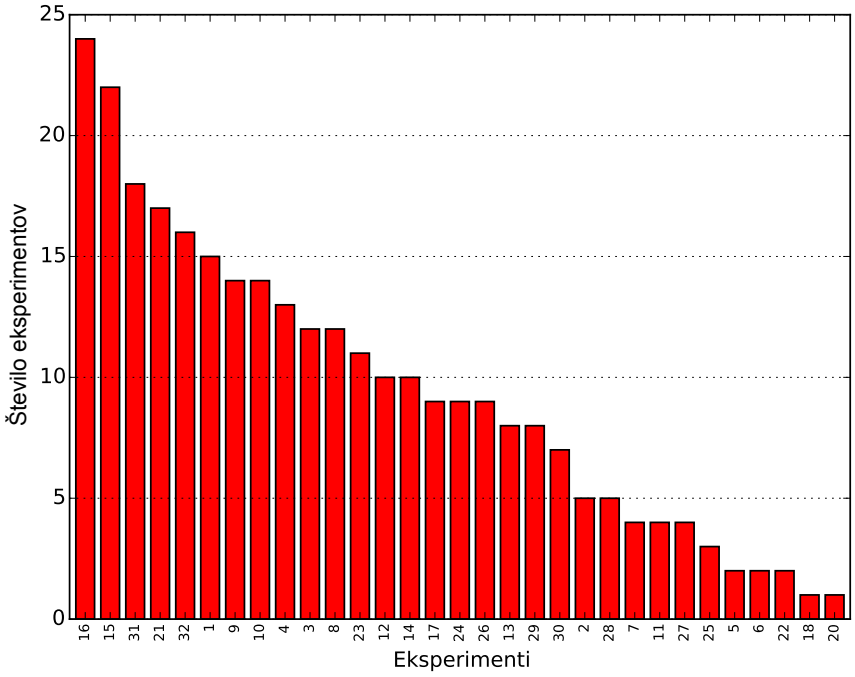
Preden smo začeli graditi združene HMM, smo vrednotili osnovne HMM vseh eksperimentov na testnih množicah zaporedij, ki smo jih uporabili pri vrednotenju posameznih HMM. Tako smo ugotovili, kako dobro HMM drugih eksperimentov napovedujejo mesta vezave drugih eksperimentov. Rezultati so podani v tabeli (Tabela 5.6). Na grafu (Slika 5.10) vidimo, koliko eksperimentov je HMM določenega eksperimenta napovedal bolje od pripadajočih HMM, ne da bi izločili replike eksperimentov posameznih proteinov. Nato smo odstranili HMM replike in tako dobili graf (Slika 5.11). Vidimo, da v prvem grafu HMM vsakega eksperimenta napove bolje vsaj en drug eksperiment. V drugem grafu pa vidimo, da samo HMM eksperimenta 19 (hnRNPL 2) ne napove bolje nobenega drugega eksperimenta. Najbolj izstopata eksperimenta 16 (hnRNPC 1) in 15 (IGF2BP1-3), ki bolje napovesta več kot 20 drugih eksperimentov. Na grafu (Slika 5.12) imamo prikazano mrežo, kjer vozlišča predstavljajo eksperimente, povezave pa predstavljajo, da izhodiščni HMM eksperimenta bolje napove končni eksperiment, kot njegov HMM. Pri mreži smo zavrgli napovedi replik eksperimentov posameznih proteinov in nato izrisali le povezave do eksperimentov, če je posamezni HMM eksperimenta med najboljšimi tremi HMM eksperimenti, ki napovedujejo mesta vezave tega eksperimenta bolje, kot pripadajoči HMM. Vrednosti AUC smo



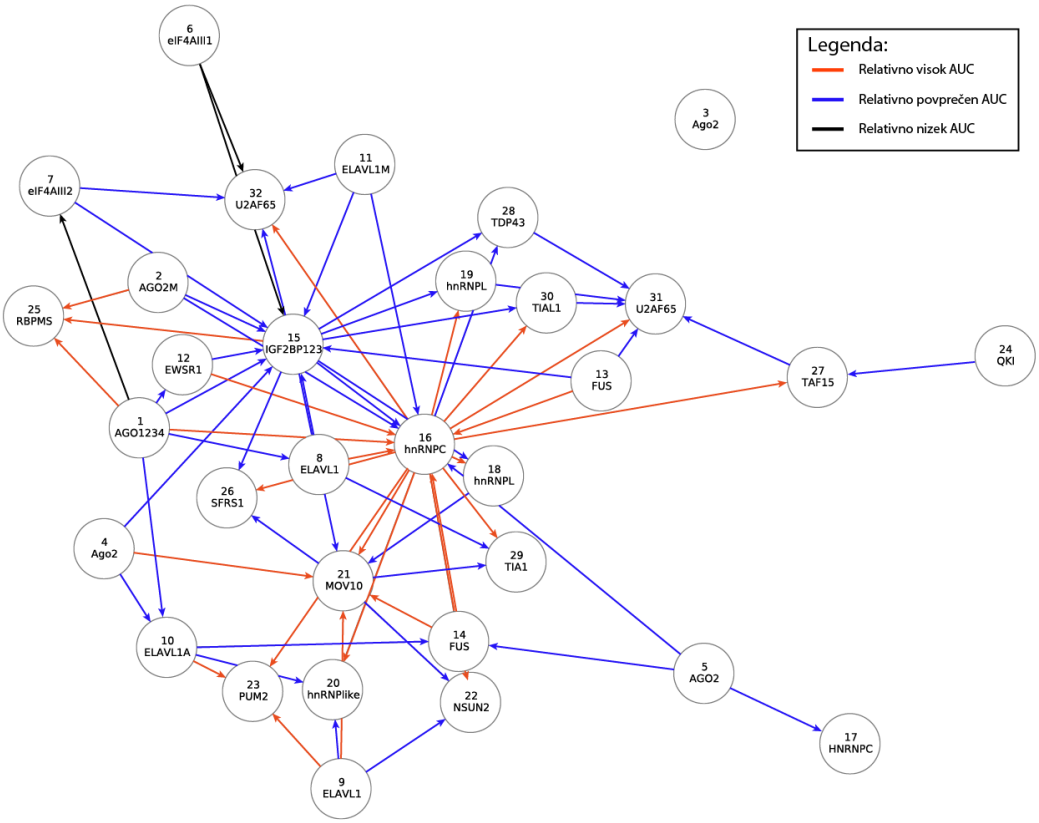
Slika 5.10: Graf HMM posameznih eksperimentov, ki boljše napovedujejo druge eksperimente, kot pripadajoči HMM.

diskretizirali v tri kategorije: relativno visok, povprečen in nizek AUC (glej poglavje 4.3). Na grafu se jasno vidi, da izstopata eksperimenta 16 (hnRNPC 1) in 15 (IGF2BP1-3), ki dobro napovedujeta veliko ostalih eksperimentov.

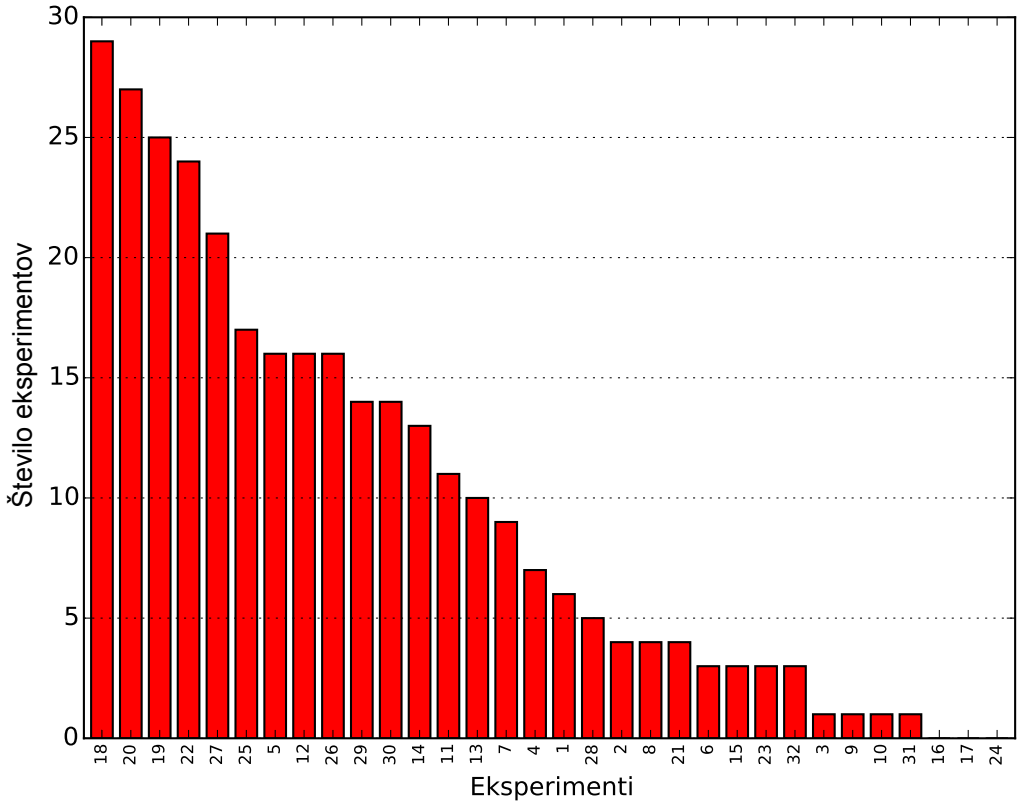
Na grafu (Slika 5.13) vidimo, koliko HMM drugih eksperimentov bolje napove izbrani eksperiment kot pripadajoči osnovni HMM. Nato izločimo HMM replike eksperimentov in dobimo graf (Slika 5.14). Iz prvega grafa vidimo, da eksperimente 16 (hnRNPC 1), 17 (hnRNPC 2) in 24 (QKI) ne napove boljše noben HMM drugega eksperimenta. Če izločimo replike eksperimentov, imamo poleg prejšnjih treh še eksperimente 3 (Ago2 1), 9 (ELAVL1 2) in 10 (ELAVL1A). Eksperiment 18 (hnRNPL 1) pa napove bolje večina eksperimentov, če upoštevamo replike eksperimente ali ne. Kar 13 eksperimentov je takih, da jih bolje napove več kot 10 HMM drugih eksperimentov.



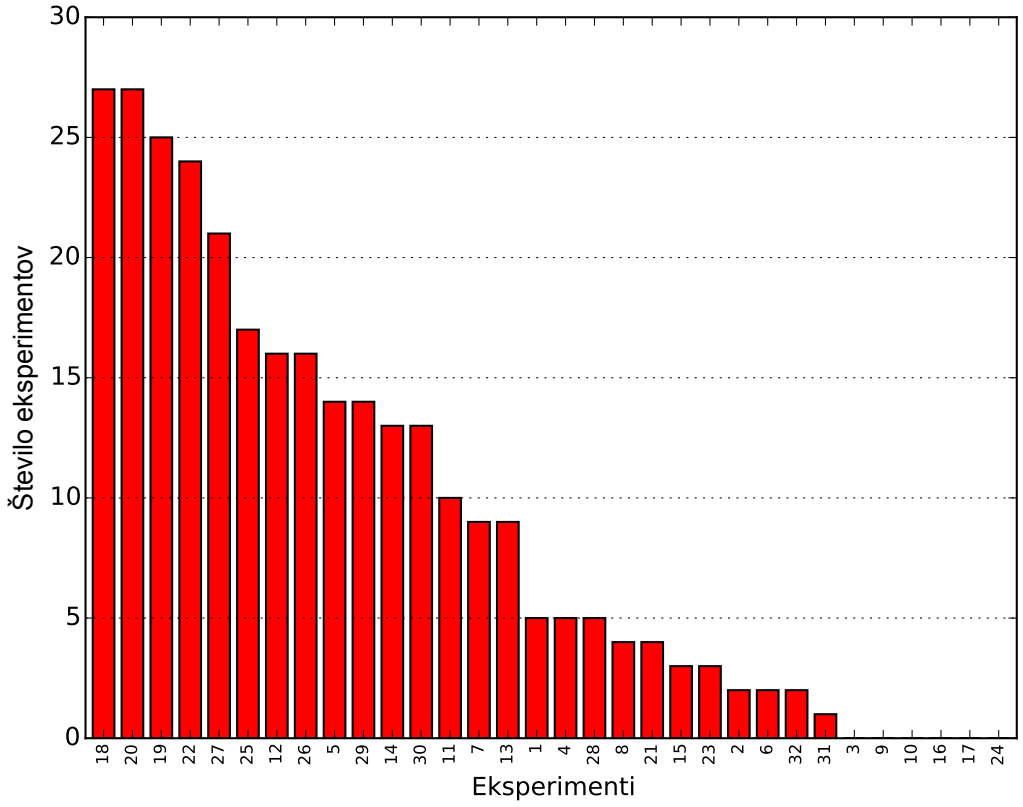
Slika 5.11: Graf HMM posameznih eksperimentov, ki boljše napovedujejo druge eksperimente, kot pripadajoči HMM. Izločili smo HMM eksperimentov, ki predstavljajo replike proteinov.



Slika 5.12: Mreža, kjer vozlišča predstavljajo eksperimente. Vozlišči sta povezani, če HMM izvornega eksperimenta boljše napove končni eksperiment, kakor to napove HMM končnega eksperimenta. Vrednosti AUC predstavljajo uteži povez: rdeča, relativno visok AUC, modra, relativno povprečen AUC in črna, relativno nizek AUC.



Slika 5.13: Graf, ki prikazuje koliko HMM ostalih eksperimentov boljše napove izbrani eksperiment.



Slika 5.14: Graf, ki prikazuje koliko HMM ostalih eksperimentov boljše napove izbrani eksperiment. Izločili smo HMM eksperimentov, ki predstavljajo replike proteinov.

Št.	Protein	AUC	HMM drugih eksperimentov, ki bolje napovedujejo izbrani eksperiment
1	Ago/EIF2C1-4	0.85	16 (0.90), 4 (0.86), 10 (0.86), 15 (0.86), 31 (0.86), 21 (0.85)
2	Ago2-MNase	0.85	16 (0.88), 4 (0.87), 3 (0.87), 15 (0.85)
3	Ago2 (1)	0.90	4 (0.91)
4	Ago2 (2)	0.88	21 (0.91), 15 (0.91), 1 (0.90), 10 (0.89), 9 (0.89), 3 (0.88), 23 (0.88)
5	Ago2 (3)	0.77	16 (0.89), 17 (0.89), 14 (0.88), 15 (0.88), 31 (0.85), 21 (0.85), 1 (0.84), 9 (0.84), 26 (0.84), 32 (0.83), 20 (0.82), 18 (0.81), 13 (0.81), 22 (0.79), 2 (0.78), 10 (0.78)
6	eIF4AIII (1)	0.68	32 (0.80), 15 (0.74), 7 (0.68)
7	eIF4AIII (2)	0.73	15 (0.83), 32 (0.83), 1 (0.79), 11 (0.77), 31 (0.76), 21 (0.75), 16 (0.75), 3 (0.74), 4 (0.74)
8	ELAVL1 (1)	0.83	16 (0.90), 15 (0.88), 1 (0.88), 31 (0.86)
9	ELAVL1 (2)	0.92	10 (0.93)
10	ELAVL1A	0.92	9 (0.94)
11	ELAVL1-MNase	0.72	32 (0.85), 16 (0.84), 8 (0.84), 15 (0.81), 30 (0.80), 2 (0.80), 26 (0.79), 31 (0.79), 13 (0.73), 7 (0.73), 12 (0.73)
12	ESWR1	0.79	16 (0.90), 15 (0.89), 1 (0.87), 24 (0.87), 31 (0.87), 21 (0.86), 8 (0.84), 32 (0.84), 9 (0.83), 10 (0.83), 14 (0.82), 4 (0.81), 3 (0.81), 23 (0.81), 17 (0.80), 11 (0.79)
13	FUS	0.77	16 (0.91), 31 (0.86), 15 (0.85), 32 (0.85), 24 (0.83), 1 (0.83), 8 (0.80), 21 (0.79), 12 (0.79), 14 (0.77)
14	Mut FUS	0.84	16 (0.90), 21 (0.90), 10 (0.89), 9 (0.89), 1 (0.89), 23 (0.89), 15 (0.89), 29 (0.88), 3 (0.87), 4 (0.87), 31 (0.86), 24 (0.86), 8 (0.86)
15	IGF2BP1-3	0.86	16 (0.89), 21 (0.87), 4 (0.87)
16	hnRNPC (1)	0.91	
17	hnRNPC (2)	0.93	
18	hnRNPL (1)	0.67	16 (0.91), 21 (0.89), 15 (0.88), 31 (0.87), 1 (0.87), 23 (0.86), 4 (0.86), 8 (0.86), 3 (0.86), 10 (0.86), 9 (0.86), 24 (0.86), 32 (0.85), 29 (0.83), 17 (0.83), 5 (0.82), 12 (0.81), 14 (0.81), 28 (0.81), 30 (0.80), 26 (0.79), 13 (0.79), 27 (0.77), 11 (0.75), 25 (0.74), 19 (0.73), 20 (0.72), 7 (0.71), 6 (0.69)
19	hnRNPL (2)	0.69	16 (0.90), 31 (0.87), 15 (0.86), 1 (0.85), 32 (0.84), 8 (0.84), 21 (0.81), 29 (0.80), 30 (0.79), 23 (0.78), 12 (0.77), 14 (0.76), 28 (0.76), 11 (0.75), 26 (0.75), 13 (0.75), 4 (0.74), 10 (0.74), 2 (0.73), 9 (0.73), 3 (0.73), 24 (0.72), 7 (0.72), 17 (0.71), 27 (0.70)
20	hnRNPL-like	0.70	16 (0.91), 9 (0.90), 10 (0.89), 21 (0.88), 1 (0.87), 31 (0.86), 15 (0.86), 17 (0.86), 24 (0.84), 23 (0.84), 14 (0.83), 8 (0.83), 32 (0.82), 26 (0.82), 4 (0.81), 3 (0.81), 28 (0.80), 12 (0.78), 13 (0.77), 29 (0.75), 30 (0.74), 7 (0.73), 5 (0.73), 22 (0.72), 27 (0.71), 6 (0.71), 25 (0.71)
21	MOV10	0.90	16 (0.91), 4 (0.90), 9 (0.90), 3 (0.90)
22	Nsun2	0.76	16 (0.92), 21 (0.89), 9 (0.89), 10 (0.88), 15 (0.87), 1 (0.86), 31 (0.86), 12 (0.85), 23 (0.85), 3 (0.84), 8 (0.84), 14 (0.84), 4 (0.84), 29 (0.84), 17 (0.83), 2 (0.82), 30 (0.82), 32 (0.81), 13 (0.80), 27 (0.80), 28 (0.79), 26 (0.78), 25 (0.77), 24 (0.77)
23	PUM2	0.89	16 (0.90), 10 (0.90), 9 (0.90)
24	QKI	0.93	
25	RBPMS	0.84	15 (0.96), 1 (0.94), 2 (0.94), 21 (0.92), 16 (0.91), 14 (0.91), 26 (0.90), 23 (0.89), 8 (0.88), 24 (0.88), 31 (0.87), 9 (0.87), 29 (0.87), 4 (0.86), 3 (0.86), 10 (0.86), 32 (0.85)
26	SFRS1	0.79	16 (0.90), 15 (0.87), 21 (0.87), 31 (0.86), 1 (0.85), 4 (0.85), 3 (0.84), 32 (0.84), 8 (0.83), 29 (0.82), 10 (0.82), 30 (0.82), 12 (0.81), 23 (0.81), 14 (0.81), 2 (0.80)
27	TAF15	0.75	16 (0.90), 24 (0.89), 31 (0.87), 15 (0.85), 9 (0.85), 10 (0.85), 8 (0.84), 32 (0.84), 1 (0.83), 21 (0.83), 17 (0.83), 14 (0.82), 23 (0.81), 12 (0.79), 3 (0.78), 13 (0.78), 4 (0.78), 26 (0.78), 29 (0.77), 30 (0.76), 28 (0.75)
28	TDP-43	0.78	16 (0.89), 15 (0.85), 31 (0.85), 32 (0.83), 1 (0.82)
29	TIA1	0.83	16 (0.90), 8 (0.87), 21 (0.86), 31 (0.86), 10 (0.86), 17 (0.86), 9 (0.85), 15 (0.85), 23 (0.85), 32 (0.84), 3 (0.83), 4 (0.83), 12 (0.83), 1 (0.83)
30	TIAL1	0.77	16 (0.91), 15 (0.87), 31 (0.86), 1 (0.85), 32 (0.83), 8 (0.82), 17 (0.81), 29 (0.80), 14 (0.80), 21 (0.80), 12 (0.79), 13 (0.78), 26 (0.77), 9 (0.77)
31	U2AF2 (1)	0.86	16 (0.91)
32	U2AF2 (2)	0.84	16 (0.91), 31 (0.87), 15 (0.86)

Tabela 5.6: Tabela eksperimentov, vrednosti AUC posameznih HMM in seznam HMM drugih eksperimentov, ki so vrnili boljše vrednosti AUC kot HMM samega eksperimenta.

Združili smo napovedi HMM posameznih eksperimentov v skupni HMM ter jih vrednotili. V tabeli (Tabela 5.7) imamo rezultate, ki smo jih dobili tako, da smo združili napovedi HMM samega eksperimenta in napovedi HMM najboljših štirih eksperimentov. Pred združevanjem smo izločili osnovne HMM, ki predstavljajo replike eksperimentov. Vrednosti AUC združenih HMM so nižje od vrednosti AUC HMM posameznih eksperimentov, razen pri eksperimentih 5 (Ago2 3), 18 (hnRNPL 1) in 26 (SFRS1). Zanimivo je tudi to, da se pri desetih združenih HMM, kadar uporabimo pet posameznih HMM, vrednost AUC občutno zmanjša v primerjavi z združitvijo manjšega števila posameznih HMM. V grafih (Slika 5.15, 5.16, 5.17, 5.18) vidimo pogostost pojavitve posameznih emisij glede na število združenih modelov. V vseh štirih grafih vidimo, da je najbolj pogosta emisija tista, ki je sestavljena iz samih števil ena. Torej emisija, kjer so vsi HMM v združenem HMM napovedali mesto vezave. Iz emisij lahko sklepamo tudi o interakcijah med proteini, katerih HMM so združeni v skupni model, in sicer, ali tekmujejo za enaka vezavna mesta, ali sodelujejo in so odvisni eden od drugega ter se mora določeni vezati pred drugim, ali pa se določen protein na mesta interakcij ne veže in tako omogoči vezave drugih proteinov.

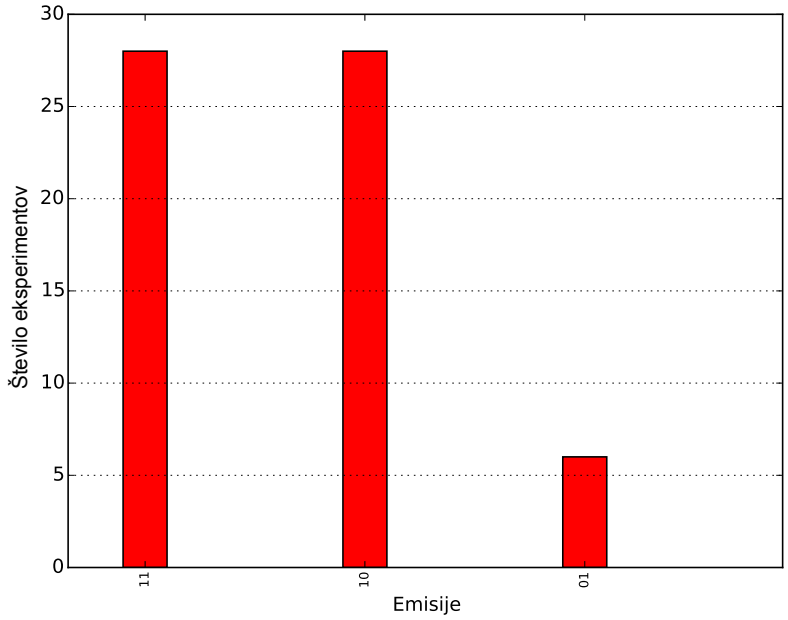
Št.	Protein	AUC	Izbrani eksperimenti	Združeni AUC	Izbrane emisije
1	Ago/EIF2C1-4	0.85	1, 16	0.80	11, 10
			1, 16, 10	0.81	101, 111, 100, 110
			1, 16, 10, 15	0.81	0101, 1011, 1111, 0111, 0001, 1001, 0011, 1101, 1010, 1000
			1, 16, 10, 15, 31	0.76	01011, 10111, 10110, 11111, 01111, 11110, 00011, 00111, 10011, 00010
2	Ago2-MNase	0.85	2, 16	0.77	10, 11
			2, 16, 15	0.77	011, 001, 101, 111
			2, 16, 15, 31	0.77	0111, 0011, 1011, 0110, 0010, 1111, 1010, 1110
			2, 16, 15, 31, 32	0.68	01111, 10101, 00111, 00101, 01101, 10111, 00100, 11111, 11101
3	Ago2 (1)	0.90	3, 8	0.85	11, 10
			3, 8, 21	0.85	111, 101, 100, 110
			3, 8, 21, 16	0.85	1110, 1111, 1010, 1000, 1101, 1100
			3, 8, 21, 16, 29	0.85	11010, 11000, 11101, 11111, 10000, 11100, 10100
4	Ago2 (2)	0.88	4, 21	0.82	11, 10
			4, 21, 15	0.82	111, 100, 101
			4, 21, 15, 10	0.82	1111, 1110, 1000, 1011, 1010, 0011
			4, 21, 15, 10, 9	0.82	11111, 11100, 11101, 10000, 01101
5	Ago2 (3)	0.77	5, 16	0.77	11, 10
			5, 16, 17	0.80	101, 111, 110, 100
			5, 16, 17, 14	0.75	1011, 0111, 1111, 0001, 1101, 1001, 0101, 0010, 1010
			5, 16, 17, 14, 15	0.54	01111, 10111, 11111, 01101, 01001,

Št.	Protein	AUC	Izbrani eksperimenti	Združeni AUC	Izbrane emisije
					00011, 01011, 10101, 00101, 00001
6	eIF4AIII (1)	0.68	6, 32	0.66	11, 10
			6, 32, 15	0.67	111, 110, 101, 001, 011
			6, 32, 15, 1	0.66	1110, 0001, 1111, 1100, 1010, 1000, 1101, 0010, 0101, 1011
			6, 32, 15, 1, 19	0.66	11100, 00010, 11110, 11000, 10100, 10000, 11010, 00100, 01010, 10110
7	eIF4AIII (2)	0.73	7, 15	0.73	11, 01, 10
			7, 15, 32	0.73	111, 101, 110, 010, 011
			7, 15, 32, 1	0.72	1110, 0001, 1111, 1100, 1000, 1010, 0100, 0011, 1011, 1101
			7, 15, 32, 1, 11	0.68	00011, 11101, 00001, 00101, 00010, 11111, 10000, 01001, 11001, 10100
8	ELAVL1 (1)	0.83	8, 16	0.80	10, 11
			8, 16, 15	0.80	111, 101, 011, 100, 110
			8, 16, 15, 1	0.78	1110, 1111, 1001, 1011, 1010, 0111, 1101, 0110, 1000
			8, 16, 15, 1, 31	0.73	11101, 10101, 11111, 10111, 10011, 01111, 01100, 11110
9	ELAVL1 (2)	0.92	9, 14	0.83	11, 10, 01
			9, 14, 21	0.83	111, 100, 110, 001, 101, 010
			9, 14, 21, 16	0.83	1111, 1110, 1000, 1100, 1011, 0010, 1001, 0011, 1101
			9, 14, 21, 16, 15	0.80	10001, 10011, 11000, 11111, 10111, 10101, 11101
10	ELAVL1A	0.92	10, 16	0.84	11, 10
			10, 16, 21	0.88	111, 101, 100, 011, 001, 110
			10, 16, 21, 15	0.87	1111, 1101, 1001, 1011, 0111, 0101, 0011, 1010, 1110, 1000
			10, 16, 21, 15, 23	0.85	11110, 11111, 11010, 10110, 10010, 01110, 10001
11	ELAVL1-MNase	0.72	11, 32	0.71	11, 10
			11, 32, 16	0.71	110, 111, 100
			11, 32, 16, 15	0.72	1101, 0111, 1111, 0001, 0101, 1100, 1001, 1011
			11, 32, 16, 15, 30	0.70	01110, 11011, 01111, 10110, 11110, 11010, 10010, 11001, 11111, 00010
12	ESWR1	0.79	12, 16	0.76	11, 10
			12, 16, 15	0.76	111, 101, 100, 011
			12, 16, 15, 1	0.76	1001, 1101, 1110, 0110, 1010, 1111, 1011, 0010, 1000
			12, 16, 15, 1, 24	0.66	10010, 10011, 11010, 11101, 11111, 11100, 00101, 10111
13	FUS	0.77	13, 16	0.68	10, 11
			13, 16, 31	0.70	101, 110, 100, 111, 001
			13, 16, 31, 15	0.68	1011, 1111, 1101, 1001, 1010, 1000, 1100, 0011, 1110
			13, 16, 31, 15, 32	0.68	10111, 11111, 11011, 10110, 10011, 10101, 10000, 10010, 10001
14	Mut FUS	0.84	14, 16	0.80	11, 10
			14, 16, 21	0.80	111, 101, 011, 001, 100, 110
			14, 16, 21, 10	0.80	1111, 1110, 1011, 1010, 0110, 0111, 0011, 0010, 1001
			14, 16, 21, 10, 9	0.80	11111, 11100, 10111, 11101, 10101, 11110, 01100, 10100, 10001, 00100
15	IGF2BP1-3	0.86	15, 16	0.80	10, 11
			15, 16, 21	0.82	111, 101, 100, 110
			15, 16, 21, 4	0.83	1111, 1011, 0101, 0001, 1010, 1110, 1001, 1101
			15, 16, 21, 4, 31	0.81	10111, 01010, 11111, 10110, 11110, 01011, 00011, 00010, 10101, 10100
16	hnRNPC (1)	0.91	16, 31	0.86	11, 10, 01
			16, 31, 32	0.86	111, 101, 110, 100, 011, 010

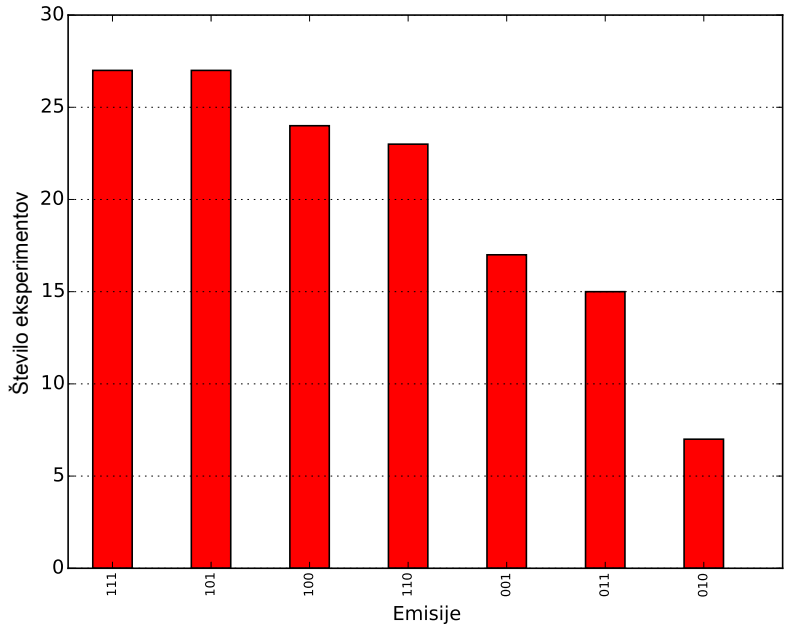
Št.	Protein	AUC	Izbrani eksperimenti	Združeni AUC	Izbrane emisije
			16, 31, 32, 15	0.86	1111, 1011, 1001, 1110, 1100, 1010, 0111, 1000, 0101, 0110
			16, 31, 32, 15, 1	0.83	11110, 11111, 10111, 10011, 11100, 11000, 10101, 10100, 11101, 01110
17	hnRNPC (2)	0.93	17, 8	0.82	11, 10, 01
			17, 8, 24	0.82	110, 111, 101, 100, 011, 010
			17, 8, 24, 31	0.85	1100, 1101, 1111, 1001, 1110, 1010, 0111, 1011, 0101, 0110
			17, 8, 24, 31, 15	0.80	11000, 11011, 11001, 11111, 11010, 10010, 01111, 10011, 11101, 10100
18	hnRNPL (1)	0.67	18, 16		
			18, 16, 21	0.66	011, 001
			18, 16, 21, 15	0.67	0111, 0011, 0101, 0001, 0110
			18, 16, 21, 15, 31	0.68	00111, 01111, 01010, 00110, 01110, 01011, 00010, 01100, 00011
19	hnRNPL (2)	0.69	19, 16		
			19, 16, 31		
			19, 16, 31, 15	0.64	0011, 0111, 0101, 0001
			19, 16, 31, 15, 1	0.64	00111, 01111, 01110, 01011, 00110, 00011, 00010, 00101
20	hnRNPL-like	0.70	20, 16		
			20, 16, 9	0.68	001, 011
			20, 16, 9, 10	0.68	0011, 0111, 0010, 0110, 0101, 0001
			20, 16, 9, 10, 21	0.69	01111, 00111, 01001, 00001, 00100, 01011, 01101, 00101, 00010, 00110
21	MOV10	0.90	21, 16	0.84	11, 10
			21, 16, 4	0.85	111, 101, 110, 100, 011, 001
			21, 16, 4, 9	0.84	1111, 1100, 1011, 1110, 1010, 1001, 1101, 0011, 0111, 0110
			21, 16, 4, 9, 3	0.83	11001, 11111, 00110, 10111, 11100, 10110, 01100, 11101, 01001, 10101
22	Nsun2	0.76	22, 16		
			22, 16, 21		
			22, 16, 21, 9	0.74	0011, 0010
			22, 16, 21, 9, 10	0.74	00111, 00100, 00101, 01100, 00110
23	PUM2	0.89	23, 16	0.86	11, 10
			23, 16, 10	0.86	111, 101, 110, 100, 001
			23, 16, 10, 9	0.86	1111, 1101, 1011, 1001, 1110, 1100, 1000, 0011, 0001
			23, 16, 10, 9, 21	0.82	11011, 11111, 11001, 11101, 10111, 10100, 10011
24	QKI	0.93	24, 16	0.85	10, 11
			24, 16, 31	0.85	101, 110, 100
			24, 16, 31, 14	0.85	1100, 1011, 1000, 1010, 1111, 1101, 1001, 1110, 0011
			24, 16, 31, 14, 32	0.84	01110, 11000, 10001, 11001, 10111, 10000, 10100, 11100, 10101, 10010
25	RBPMS	0.84	25, 15	0.74	11, 10
			25, 15, 1	0.74	101, 111, 110, 100, 001, 010
			25, 15, 1, 2	0.74	1010, 1100, 1111
			25, 15, 1, 2, 21	0.74	10100, 11001, 11111, 10111, 10000, 11011, 00100, 01111, 11110, 01011
26	SFRS1	0.79	26, 16	0.77	10, 11
			26, 16, 15	0.79	101, 001, 100, 111
			26, 16, 15, 21	0.80	1011, 0010, 1000, 0011, 1010
			26, 16, 15, 21, 31	0.80	11000, 10110, 00100, 00101, 10000, 10001, 00110, 10111, 10100
27	TAF15	0.75	27, 16	0.71	11, 10
			27, 16, 24	0.73	111, 101, 110, 100, 001,
			27, 16, 24, 31	0.73	1011, 1111, 1110, 1010, 1001, 1101, 1000, 1100, 0110, 0010
			27, 16, 24, 31, 15	0.66	01111, 11000, 10111, 11010, 10010, 11111, 01101, 11101, 00101, 10101

Št.	Protein	AUC	Izbrani eksperimenti	Združeni AUC	Izbrane emisije
28	TDP-43	0.78	28, 16	0.75	10, 11
			28, 16, 15	0.75	100, 101 ,111, 110
			28, 16, 15, 31	0.74	1000, 1010
			28, 16, 15, 31, 32	0.74	10000, 10100, 11001, 10101
29	TIA1	0.83	29, 16	0.76	10, 11
			29, 16, 8	0.79	101, 111, 001
			29, 16, 8, 21	0.79	1111, 1011, 0011, 0111, 1010, 0010, 0101, 0001, 1110, 1101
			29, 16, 8, 21, 31	0.74	11111, 10111, 01110, 11110, 00111, 10110, 00110, 10011, 01111, 10101
30	TIAL1	0.77	30, 16	0.73	10, 11
			30, 16, 15	0.72	111, 101, 011, 001, 100, 110
			30, 16, 15, 31	0.71	1011, 1111, 0011, 1110, 0111, 1010, 1000, 0010, 0110, 1001
			30, 16, 15, 31, 1	0.70	11110, 10110, 10111
31	U2AF2 (1)	0.86	31, 16	0.77	10, 11, 01
			31, 16, 15	0.77	101, 111, 100, 011, 001, 110, 010
			31, 16, 15, 8	0.75	1011, 1010, 0001, 1110, 1001, 1111, 0011, 1000
			31, 16, 15, 8, 30	0.62	10111, 11100, 10100, 10101, 00011, 00010, 10011, 10110, 00110, 10001
32	U2AF2 (2)	0.84	32, 16	0.77	10, 11, 01
			32, 16, 15	0.78	111, 101, 100, 011, 110, 001, 010
			32, 16, 15, 30	0.76	1010, 1110, 1111, 1011, 0001, 0110, 1001, 0010, 0111, 1000
			32, 16, 15, 30, 8	0.64	10100, 11100, 10101, 11101, 00101, 11111, 00011, 10111, 01100, 10010

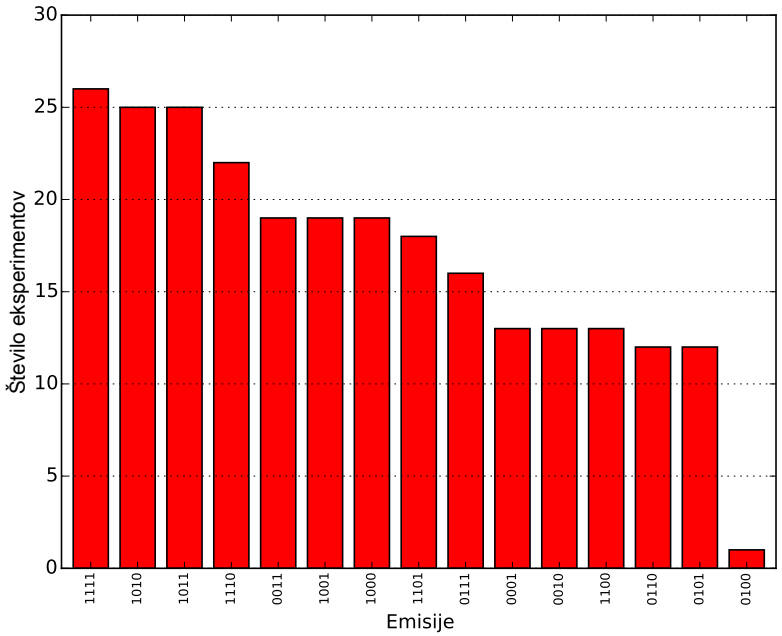
Tabela 5.7: Tabela izbranih atributov pri gradnji in AUC vrednosti pri vrednotenju posameznih HMM.



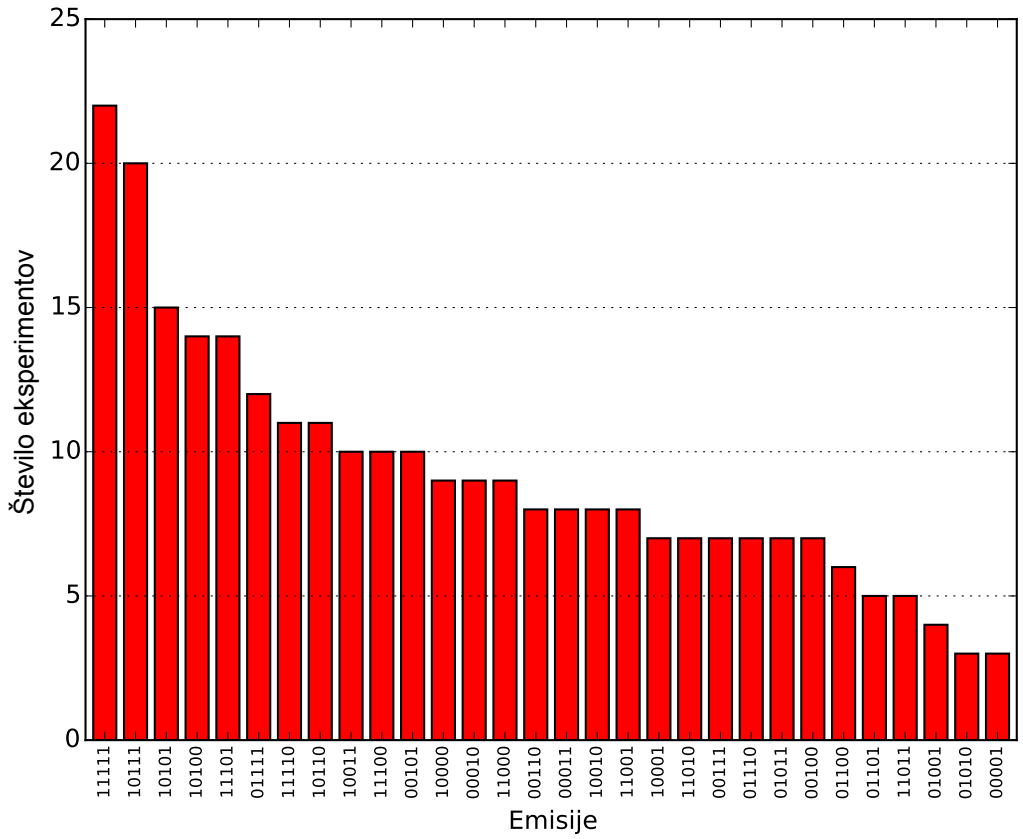
Slika 5.15: Graf, ki prikazuje pogostost emisij pri združitvi dveh HMM.



Slika 5.16: Graf, ki prikazuje pogostost emisij pri združitvi treh HMM.



Slika 5.17: Graf, ki prikazuje pogostost emisij pri združitvi štirih HMM.



Slika 5.18: Graf, ki prikazuje pogostost emisij pri združitvi petih HMM.

Poglavje 6

Zaključek

V sklopu magistrske naloge smo razvili metodo za napovedovanje mest interakcij med RNA in proteini, ki temelji na skritih Markovih modelih. Najprej smo analizirali podatke, jih preuredili in izluščili reprezentativne atribute in motive. Zgradili smo HMM za vsak eksperiment posebej ter jih vrednotili. Izvedli smo primerjavo HMM, in sicer tako, da smo z vsakim HMM napovedovali druge eksperimente. Na podlagi teh primerjav smo za vsak eksperiment zgradili združen model in ga vrednotili.

Iz zaporedij RNA in majhnega števila dodatnih atributov smo izluščili atribute in motive, ki opisujejo mesta interakcij. Izluščili smo jih tako, da smo za vse motive in atribute izračunali verjetnost njihove pojavitve na mestih interakcij in njihovi okolici. Za vsak eksperiment smo dobili množico reprezentativnih atributov in motivov. Ugotovili smo, da se določeni motivi in atributi pogosto pojavljajo pri več eksperimentih. Glavni preskus atributov in motivov pa smo naredili tako, da iz njih zgradimo modele ter jih vrednotimo.

Iz reprezentativnih atributov smo zgradili HMM za napovedovanja mest interakcij posameznih eksperimentov. Za emisije smo uporabili izbrane atribute in motive, ostale pa smo predstavili s splošno emisijo. Ko smo vrednotili HMM, smo dosegli zmogljivosti oziroma vrednosti AUC večje od 0.68 in v povprečju 0.82. Zmogljivosti HMM so veliko boljše od naključnega klasifika-

torja in potrdijo, da se iz podatkov da izluščiti reprezentativne atribute in motive. Prav tako se lahko uporabi HMM za napovedovanje mest interakcij proteinov in RNA.

Razširili smo HMM z več stanji in poskusili izboljšati napovedovanje mest interakcij. To smo preverili z izgradnjo osnovnih modelov najprej z dvema stanjema, nato pa s tremi. Po vrednotenju smo ugotovili, da so si HMM za posamezen eksperiment med seboj primerljivi. Ugotovili smo tudi, da metoda MAE ni primerna metoda za vrednotenje, saj imajo mesta interakcije zelo majhno verjetnost pojavitve in napake ne vplivajo dovolj na končno vrednost. Prav tako za 9 eksperimentov nismo imeli podatkov, ki bi se lahko razširili v tri stanja. Zato smo se odločili, da pri združevanju HMM uporabimo modele z dvema stanjema.

Združili smo osnovne HMM posameznih proteinov in poskusili izboljšati napovedovanje mest interakcij. To smo preverili tako, da smo zgradili HMM, ki so za vhodne podatke vzeli združene napovedi več osnovnih HMM. Ugotovili smo, da nam združeni HMM delujejo nekoliko slabše od posameznih HMM, vendar nam še vedno vračajo višje AUC vrednosti od 0.5.

Na koncu nas je zanimalo, ali lahko iz združevanja HMM posameznih proteinov sklepamo o interakcijah med samimi proteini. To smo preverili tako, da smo analizirali emisije združenih HMM. Če nam več osnovnih HMM za določen nukleotid vrne, da je na tem mestu vezava, potem ti proteini tekmujejo za določena vezavna mesta. Če nam vrnejo, da se na tem mestu ne vežejo, lahko sklepamo, da jih to mesto odbija in omogoča vezave drugih proteinov. Vezavna mesta proteinov se lahko tudi prekrivajo in iz tega lahko sklepamo o sodelovanju proteinov.

RNA in pripadajoči atributi so predstavljeni kot zelo dolga zaporedja, kar predstavlja določene izzive pri obdelavi. Pri algoritmih smo morali paziti na matematično stabilnost in na minimalno vrednost, ki jo lahko računalnik še shrani. To smo rešili z uporabo logaritmov in množenja vrednosti v vsakem koraku. Prav tako smo morali uporabiti 64-bitno različico Python-a, saj določenih zaporedij nismo mogli prebrati v pomnilnik računalnika. Paziti

smo morali tudi na način shranjevanje osnovnih podatkov in vmesnih rezultatov, saj zavzamejo veliko prostora in je njihovo branje posledično počasnejše. Uporabili smo shranjevanje redkih matrik, ki smo jih nato še dodatno stisnili. V pomnilniku pa smo hranili samo tiste podatke, ki so trenutno v obdelavi. Postopek gradnje trajal kar nekaj časa, in sicer gradnja HMM vseh 32-tih eksperimentov in njihovo vrednotenje je trajalo tudi po nekaj dni.

V magistrskem delu smo definirali osnovne metode za napovedovanje mest interakcij z uporabo HMM, kar predstavlja dobro izhodišče za nadaljnje delo, ter optimizacija in izboljšavo paralelizacije trenutnih metod. Dobro bi bilo poskusiti uporabiti nove dodatne atribute za boljši opis mest interakcije, na primer tri dimenzionalna oblika RNA. Za reprezentativne atribute bi poskusili uporabiti tudi zaporedja in lastnosti proteinov. Dodali in izboljšali bi lahko metode za iskanje reprezentativnih atributov in motivov, na primer z uporabo entropije in uporabo gručenja za določitev bolj splošnih atributov. Zgradili bi nove atribute, ki vsakemu nukleotidu določijo kontekst, tako da predstavijo njegovo okolico. Tukaj bi lahko uporabili nekaj predhodnih in naslednjih nukleotidov, razdaljo do drugih reprezentativnih motivov ter atributov ali pa razdaljo do drugih mest vezave, lahko pa tudi skupek teh atributov. Razširili bi stanja in emisije HMM na zvezne vrednosti, da bi lahko bolj natančno predstavili intenzitete mest interakcij. Tukaj bi bilo potrebno dodati tudi metode za reševanje integralov in odvodov. Izboljšali bi vrednotenje modelov z več stanji, na primer z uporabo večdimenzionalnih krivulj ROC ali pa s primerjavo eden proti vsem ostalim ali pa celo z drugimi merami. Združevali bi napovedi posameznih osnovnih HMM z drugimi metodami strojnega učenja, na primer SVM in RF. Razširili bi uporabo predhodnega znanja pri združevanju modelov, na primer z uporabo informacij o vrstah genov (angl. Gene Onthology).

Področje raziskovanja mest interakcij med RNA in proteini predstavlja dokaj novo in trenutno aktualno raziskovalno področje, na katerem se je že pojavilo nekaj metod za napovedovanje mest interakcij. V magistrski smo pokazali, da se za napovedovanje lahko uporabi tudi metodo HMM. Razvili

in predstavili smo kako, zgraditi HMM posameznih eksperimentov in kako jih združevati v skupne HMM. Verjamemo, da naše delo predstavlja dobro osnovo za nadaljnji razvoj uporabe HMM za napovedovanje mest interakcij med RNA in proteini.

Dodatek A

Implementacija

V okviru magistrske naloge smo se odločili, da bomo za pisanje programov uporabili programski jezik Python zaradi preprostosti programske kode, skriptne narave jezika. Programski jezik Python se na področju bioinformatike tudi pogosto uporablja. Programska koda je dosegljiva na spletnem naslovu: https://github.com/AleksHuc/RBPHMM

Za delo s podatki in skritimi Markovimi modeli smo implementirali lastno programsko orodje in ga strukturirali kot modul. Glavne funkcionalnosti ogrodja smo razvili sami, za lažje delo pa smo uporabili tudi nekaj prosto dostopnih knjižnic za:

- **matplotlib** - Izrisovanje grafov in slik.
- **numpy** - Kompleksne podatkovne strukture in metode za delo z njimi.
- **networkx** - Izdelavo in izris mrežnih struktur.
- **scipy** - Splošne numerične algoritme.
- **scikit-learn** - Analizo podatkov in rudarjenje podatkov.
- **Sphinx** - Avtomatsko ustvarjanje dokumentacije.
- **TableFactory** - Izpis v obliki tabel.

Naše ogrodje smo strukturirali kot modul in v njem določili več razredov, ki vsebujejo vsebinsko zaokrožene funkcionalnosti in omogočajo enostavni nadaljnji razvoj ogrodja:

- **AnalyticsManager** - Razred za vrednotenje HMM, analizo zaporedij in izpis ter izris rezultatov.
- **DiscreteEmission** - Implementacija objekta, ki predstavlja diskretno emisijo HMM.
- **DiscreteState** - Implementacija objekta, ki predstavlja diskretno stanje HMM.
- **Emission** - Vmesnik za objekt emisije HMM.
- **FileManager** - Razred za delo s podatki. Vsebuje metode za branje, urejanje in pisanje datotek ter za ustvarjanje in prilagajanje zaporedij.
- **HiddenMarkovModel** - Razred predstavlja stanje skritega Markovega modela in metode za incializacijo, gradnjo in učenje.
- **State** - Vmesnik za objekt stanja HMM.

Literatura

[1] R. Durbin, S. R. Eddy, A. Krogh, G. Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998.

[2] T. Grigorios, I. Katakis. *Multi-Label Classification: An Overview*. International Journal of Data Warehousing and Mining, 2007, str. 1-13.

[3] J. König, K. Zarnack, G. Rot, T. Curk, M. Kayikci, B. Zupan, D. J. Turner, N. M. Luscombe, J. Ule. *iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution*. Nature Structural & Molecular Biology, 2010, str. 909–915.

[4] J. König, K. Zarnack, N. M. Luscombe, J. Ule. *Protein–RNA interactions: new genomic technologies and perspectives*. Nature Reviews Genetics, 2012, str. 77–83.

[5] A. Re, T. Joshi, E. Kulberkyte, Q. Morris, C. T. Workman. *RNA–Protein Interactions: An Overview*. RNA Sequence, Structure, and Function: Computational and Bioinformatic Methods, Methods in Molecular Biology, Volume 1097, 2014, str. 491-521.

[6] D. H. Wolpert. *Stacked generalization*. Neural Networks, 1992, str. 241-259.

[7] C. Zhang, K. Y. Lee, M. S. Swanson, R. B. Darnell. *Prediction of clustered RNA-binding protein motif sites in the mammalian genome*. Nucleic Acids Research, 2013.

[8] B. Alberts, D. Bray, K. Hopkin, A. Johnson, J. Lewis, M. Raff, K. Roberts, P. Walter. *Essential Cell Biology, 4th Edition*. Garland Science, Taylor and Francis Group, LLC, 2014.

[9] S. R. Haynes. *RNA-Protein Interaction Protocols*. Methods in Molecular Biology, Volume 118, Humana Press, 1999.

[10] J. Ule, K. Jensen, A. Mele, R. B. Darnell. *CLIP: a method for identifying protein-RNA interaction sites in living cells*. Methods, Volume 37, 2005, str. 376-386.

[11] J. Ule, K. Jensen, M. Ruggiu, M. Mele, A. Ule, R. B. Darnell. *CLIP identifies NOVA-regulated RNA networks in the brain*. Science, Volume 37, 2003, str. 1212-1215.

[12] M. Hafner, M. Landthaler, L. Burger, M. Khorshid, J. Hausser, P. Berninger, A. Rothballer, M. Jr. Ascano, A. C. Jungkamp, M. Munschauer, U. Ulrich, G. S. Wardle, S. Dewell, M. Zavolan, T. Tuschi. *Transcriptome-wide identification of RNA-binding protein and microRNA target sites by PAR-CLIP*. Cell, Volume 141, 2010, str. 129-141.

[13] D. D. Licatalosi, A. Mele, J. J. Fak, J. Ule, M. Kayikci, S. W. Chi, T. A. Clark, A. C. Schweitzer, J. E. Blume, X. Wang, J. C. Darnell, R. B. Darnell. *HITS-CLIP yields genome-wide insights into brain alternative RNA processing*. Nature, Volume 456, 2008, str. 464-469.

[14] S. W. Chi, J. B. Zang, A. Mele, R. B. Darnell. *Argonaute HITS-CLIP decodes microRNA-mRNA interaction maps*. Nature, Volume 460, 2009, str. 479-486.

[15] J. König, K. Zarnack, G. Rot, T. Curk, M. Kayikci, B. Zupan, D. J. Turner, N. M. Luscombe, J. Ule. *iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution*. Nature structural and molecular biology, Volume 17, Issue 7, 2010, str. 909-915.

[16] Z. Wang, M. Kayikci, M. Briese, K. Zarnack, N. M. Luscombe, G. Rot, B. Zupan, T. Curk, J. Ule. *iCLIP Predicts the Dual Splicing Effects of TIA-RNA Interactions*. Plos biology 8 (10), 2010.

[17] I. Huppertz, J. Attig, A. D'Ambrogio, L. E. Easton, C. R. Sibley, Y. Sugimoto, M. Tajnik, J. König, J. Ule. *iCLIP: Protein–RNA interactions at nucleotide resolution*. Methods, Volume 65, 2013, str. 274-287.

[18] C. Yao, L. Weng, Y. Shi. *Global protein-RNA interaction mapping at single nucleotide resolution by iCLIP-seq*. Methods in molecular biology 2014, Volume 1126, 2014, str. 399-410.

[19] J. R. Tollervey, T. Curk, B. Rogelj, M. Briese, M. Cereda, M. Kayikci, J. König, T. Hortobágyi, A. L. Nishimura, V. Župunski, R. Patani, S. Chandran, G. Rot, B. Zupan, C. E. Shaw, J. Ule. *Characterizing the RNA targets and position-dependent splicing regulation by TDP-43*. Nature Neuroscience, Volume 14, 2011, str. 452-458.

[20] S. Kishore, S. Luber, M. Zavolan. *Deciphering the role of RNA-binding proteins in the post-transcriptional control of gene expression*. Briefings in functional genomics, Volume 9, 2010, str. 391-404.

[21] D. D. Licatalosi, R. B. Darnell. *RNA processing and its regulation: global insights into biological networks*. Nature Reviews Genetics, Volume 11, 2010, str. 75-87.

[22] R. Singh. *RNA-protein interactions that regulate pre-mRNA splicing*. Gene expression, Volume 10, 2002, str. 79-92.

[23] A. M. Khalil, J. L. Rinn. *RNA-protein interactions in human health and disease*. Gene expression, Volume 22, 2011, str. 356-365.

[24] Z. Li, P. D. Nagy. *Diverse roles of host RNA binding proteins in RNA virus replication*. RNA Biology, Volume 8, 2011, str. 305-315.

[25] A. S. Zvereva, M. M. Pooggin. *Silencing and innate immunity in plant defense against viral and non-viral pathogens.* Viruses, Volume 4, 2012, str. 2578-2597.

[26] U. K. Muppirala, B. A. Lewis, D. Dobbs. *Computational Tools for Investigating RNA-Protein Interaction Partners.* Journal of Computer Science and Systems Biology, Volume 6, 2013, str. 182-187.

[27] V. Pancaldi, J. Bähler. *In silico characterization and prediction of global protein-mRNA interactions in yeast.* Nucleic Acids research, Volume 39, 2011, str. 5826-5836.

[28] D. J. Hogan, D. P. Riordan, A. P. Gerber, D. Herschlag, P. O. Brown. *Diverse RNA-binding proteins interact with functionally related sets of RNAs, suggesting an extensive regulatory system.* Plos biology, Volume 6, 2008, e255.

[29] M. Bellucci, F. Agostini, M. Masin, G. G. Tartaglia. *Predicting protein associations with long noncoding RNAs.* Nature Methods, Volume 8, 2011, str. 444-445.

[30] U. K. Muppirala, V. G. Honavar, D. Dobbs. *Predicting RNA-protein interactions using only sequence information.* BMC Bioinformatics, Volume 12, 2011, str. 489.

[31] J. Shen, J. Zhang, X. Luo, K. Chen, Y. Li, H. Jiang. *Predicting protein-protein interactions based only on sequences information.* Proceedings of the National Academy of Sciences of the United States of America, Volume 104, 2007, str. 4337-4347.

[32] D. Cirillo, F. Agostini, G. G. Tartaglia. *Predictions of protein–RNA interactions.* Wiley Interdisciplinary Reviews: Computational Molecular Science, Volume 3, 2012, str. 161-175.

[33] Y. Wang, X. Chen, Q. Huang, Y. Wang, D. Xu, X. S. Zhang, R. Chen, L. Chen. *De novo prediction of RNA-protein interactions from sequence information.* Molecular bioSystems, Volume 9, 2013, str. 133-142.

[34] T. Puton, L. Kozlowski, I. Tuszynska, K. Rother, J. M. Bujnicki. *Computational methods for prediction of protein-RNA interactions.* Journal of structural biology, Volume 179, 2012, str. 261-268.

[35] R. R. Walia, C. Caragea, B. A. Lewis, F. Towfic, M. Terribilni, Y. El-Manzalawy, D. Dobbs, V. Honavar. *Protein-RNA interface residue prediction using machine learning: an assessment of the state of the art.* BMC Bioinformatics, Volume 13, 2012, str. 89.

[36] L. E. Baum, T. Petrie. *Statistical Inference for Probabilistic Functions of Finite State Markov Chains.* The Annals of Mathematical Statistics, Volume 37, 1966, str. 1554-1563.

[37] L. E. Baum, J. A. Eagon. *An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes and to a Model for Ecology.* Bulletin of the American Mathematical Society, Volume 73, 1967, str. 360-363.

[38] L. E. Baum, T. Petrie, G. Soules, N. Weiss. *A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains.* The Annals of Mathematical Statistics, Volume 41, 1970, str. 164-171.

[39] L. R. Rabiner. *A tutorial on Hidden Markov Models and selected applications in speech recognition.* Proceedings of the IEEE, Volume 77, 1989, str. 257-286.

[40] M. Stanke, S. Waack. *Gene prediction with a hidden Markov model and a new intron submodel.* Bioinformatics, Volume 19, 2003, str. 215-225.

[41] M. Gupta. *Generalized hierarchical markov models for the discovery of length-constrained sequence features from genome tiling arrays.* Biometrics, Volume 63, 2007, str. 797-805.

[42] X. Deng, J. Cheng. *Enhancing HMM-based protein profile-profile alignment with structural features and evolutionary coupling information.* BMC Bioinformatics, Volume 15, 2014, str. 252.

[43] J. Yun, T. Wang, G. Xiao. *Bayesian hidden Markov models to identify RNA-protein interaction sites in PAR-CLIP.* Biometrics, Volume 70, 2014, str. 430-440.

[44] S. Saftić. *Primerjava pristopov k večznačni in večciljni klasifikaciji.* Diplomsko delo, 2013.

[45] G. Madjarov, D. Kocev, D. Gjorgjevikj, S. Džeroski. *An extensive experimental comparison of methods for multi-label learning.* Pattern Recognition, Volume 45, 2012, str. 3084-3104.

[46] J. Read, B. Pfahringer, G. Holmes, E. Frank. *Classifier chains for multi-label classification.* Machine Learning, Volume 85, 2011, str. 333-359.

[47] C. M. Bishop. *Pattern Recognition and Machine Learning.* Springer, 2007.

[48] G. Tsoumakas , I. Katakis. *Multi-Label Classification: An Overview.* International Journal of Data Warehousing and Mining, Volume 3, 2007, str. 1-13.

[49] J. D. Watson, F. H. Crick. *A Structure for Deoxyribose Nucleic Acid.* Nature, Volume 171, 1953, str. 737-738.

[50] UniProt Consortium. The Universal Protein Resource (UniProt). http://www.uniprot.org/, 13.5.2015.

[51] S. Geisser. *Predictive Inference.* Chapman and Hall, New York, 1993.

[52] R. Kohavi. *A study of cross-validation and bootstrap for accuracy estimation and model selection.* Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, Volumne 2, 1995, str. 1137-1143.

[53] P. A. Devijver, J. Kittler. *Pattern Recognition: A Statistical Approach.* GB: Prentice-Hall, London, 1982.

[54] T. Fawcett. *An Introduction to ROC Analysis.* Pattern Recognition Letters, Volumne 27, 2006, str. 861–874.

[55] T. Hastie, R. Tibshirani, J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction (2nd ed.).* Springer-Verlag New York, New York, 2009.

[56] D. J. Till, R. J. Hand. *A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems.* Machine Learning, Volumne 45, 2012, str. 171–186.

[57] J. Hernandez-Orallo. *ROC curves for regression.* Pattern Recognition, Volumne 46, 2013, str. 3395-3411.

[58] R. J. Hyndman, A. B. Koehler. *Another look at measures of forecast accuracy.* International Journal of Forecasting, Volumne 22, 2006, str. 679-688.

[59] M. Zuker, P. Stiegler. *Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information.* Nucleic Acids Research, Volumne 9, 1981, str. 133-148.

[60] I. L. Hofacker, W. Fontana, P. F. Stadler, S. Bonhoeffer, M. Tacker, P. Schuster. *Fast Folding and Comparison of RNA Secondary Structures.* Monatshefte für Chemie, Volumne 125, 1994, str. 167-188.

[61] J. S. McCaskill. *The equilibrium partition function and base pair binding probabilities for RNA secondary structure*. Biopolymers, Volumne 29, 1990, str. 1105-1119.

[62] I. L. Hofacker, P. F. Stadler. *Memory efficient folding algorithms for circular RNA secondary structures*. Bioinformatics, Volumne 22, 2006, str. 1172-1176.

[63] A. F. Bompfünewerer , R. Backofen, S. H. Bernhart, J. Hertel, I. L. Hofacker, P. F. Stadler, S. Will. *Variations on RNA folding and alignment: lessons from Benasque*. Journal of mathematical biology, Volumne 56, 2008, str. 129-144.

[64] F. H. C. Crick. *On Protein Synthesis*. Symposium of the Society of Experimental Biology Journal, Volumne 12, 1956, str. 139-163.

[65] F. H. C. Crick. *Central dogma of molecular biology*. Nature, Volumne 227, 1970, str. 561-563.

[66] W. Saenger. *Principles of Nucleic Acid Structure*. Springer-Verlag New York, New York, 1984.

[67] R. A. Cartwright, D. Graur. *The Multiple Personalities of Watson and Crick Strands*. Biology Direct, Volumne 6, 2011, str. 7.

[68] E. Rivas, S. R. Eddy. *A dynamic programming algorithm for RNA structure prediction including pseudoknots*. Journal of molecular biology, Volumne 285, 1999, str. 2053-2068.

[69] J. L. Chen, C. W. Greider. *Functional analysis of the pseudoknot structure in human telomerase RNA*. Proceedings of the National Academy of Sciences, Volumne 102, 2005, str. 8080-8085.

[70] H. Pearson. *Genetics: what is a gene?*. Nature, Volumne 441, 2006, str. 398-401.

[71] E. Pennisi. *Genomics. DNA study forces rethink of what it means to be a gene*. Science, Volumne 316, 2007, str. 1556-1557.

[72] M. B. Gerstein, C. Bruce, J. S. Rozowsky, D. Zheng, J. Du, J. O. Korbel, O. Emanuelsson, Z. D. Zhang, S. Weissman, M. Snyder. *What is a gene, post-ENCODE? History and updated definition*. Genome Research, Volumne 17, 2007, str. 669-681.

[73] R. F. Murray, H. W. Harper, D. K. Granner, P. A. Mayes, V. W. Rodwell. *Harper's Illustrated Biochemistry*. Lange Medical Books/McGraw-Hill, New York, 2006.

[74] D. L. Nelson, M. M. Cox. *Lehninger's Principles of Biochemistry (4th ed.)*. W. H. Freeman and Company, New York, 2005.

[75] A. Gutteridge, J. M. Thornton. *Understanding nature's catalytic toolkit*. Trends in Biochemical Sciences, Volumne 30, 2005, str. 622-629.

[76] B. M. Lunde, C. Moore, G. Varani. *RNA-binding proteins: Modular design for efficient function*. Nature Reviews Molecular Cell Biology, Volumne 8, 2007, str. 479-490.

[77] D. J. Hogan, D. P. Riordan, D. Herschlag, P. O. Brown. *Diverse RNA-Binding Proteins Interact with Functionally Related Sets of RNAs, Suggesting an Extensive Regulatory System*. PLOS Biology (Public Library of Science), Volumne 6, 2008, str. 255.

[78] T. Glisovic, J. L. Bachorik, J. Yong, G. Dreyfuss. *RNA-binding proteins and post-transcriptional gene regulation*. FEBS Letters (Elsevier), Volumne 582, 2008, str. 1977-1986.

[79] P. H. Raven. *Biology (9th ed.)*. McGraw-Hill, New York, 2011.

[80] A. Castello, B. Fischer, K. Eichelbaum, R. Horos, B. M. Beckmann, C. Strein, N. E. Davey, D. T. Humphreys, T. Presiss, L. M. Steinmetz, J. Krijgsveld, M. W. Hentze. *Insights into RNA biology from an atlas*

of mammalian mRNA-binding proteins. Cell, Volumne 149, 2012, str. 1393-1406.