TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Informaatika eriala

Mari-Liis Kruup

Klasterduspõhine motiiviotsing lühikestel peptiididel

Magistritöö (30 EAP)

Juhendajad: Meelis Kull, PhD
Prof. Jaak Vilo

Tartu 2015

Klasterduspõhine motiiviotsing lühikestel peptiididel

Lühikokkuvõte

Uute sekveneerimistehnoloogiate abil genereeritakse palju erineva taustaga bioloogilisi andmeid. Olulise info leidmiseks tuleb neid andmeid analüüsida. Antud töös koostame meetodi, mis suudab tuvastada motiive suurest hulgast lühikestest aminohapete järjestustest ehk peptiididest, mis sisaldavad infot konkreetse inimese organismis olevate antikehade kohta. On alust arvata, et leitud motiivide abil võib olla võimalik tuvastada, milliseid haiguseid inimene on põdenud. Kuna ükski uuritud olemasolevatest tööriistadest selle probleemi lahendamiseks ei sobinud, koostasime motiivide tuvastamiseks uue meetodi. Meetodi esimene osa, sarnaste peptiidigruppide tuvastamine, põhineb hierarhilisel klasterdamisel ning sisaldab kahte erinevat võimalust hierarhilise klasterduse puust automaatselt klastrite eraldamiseks. Meetodi teine osa on sarnaste peptiidide klastritest motiivide tuvastamine. Kuna pärisandmetes olevad motiivid ei ole teada, genereerisime sünteetilised andmed, mille peal koostatud meetodit valideerida. Koostatud meetod suutis vastavalt sünteetiliste andmete omadustele tuvastada 50% kuni 100% sinna sisestatud motiividest, pärisandmetele eeldatavalt kõige sarnasema andmestiku peal 86%. Motiivide lugemise meetod töötas samamoodi hästi, etteantud müra klastrite pealt suudetakse tuvastada 100% motiividest ning müraga klastrite pealt 90% motiividest. Koostatud meetodit on võimalik rakendada ka teistest bioloogilistest andmetest motiivide otsimiseks. Sel juhul peaks muutma teatud parameetreid, mis selles töös kasutatava andmestiku jaoks on seatud. Edaspidiseks tööks võiks olla meetodi töötamise valideerimine teiste omadustega andmete peal.

Võtmesõnad: motiiviotsing, hierarhiline klasterdamine, peptiidid

Clustering-based motif discovery from short peptides

Abstract

With the help of new sequencing technologies we can generate a lot of biological data of different backgrounds. These data need to be analysed in order to extract the most important information from them. In this work we develop a method for extracting motifs from a large amount of short amino acid sequences called peptides that contain information about antibodies in that organism. Motifs found from these peptides could be linked to diseases that a person has had. Since none of the tested existing methods were suitable for solving this problem, we developed our own method that consists of two parts. First part, finding groups of similar peptides, is based on hierarchical clustering and has two different options for automatically extracting clusters from the hierarchical clustering tree. Second part is reading motifs from groups of similar peptides. Since we cannot validate the method on real data due to the lack of knowledge about the true motifs in them, we generate synthetic datasets that we validate the developed method on. The percentage of motifs the developed method could identify from synthetic data with different properties ranged from 50% to 100%, with 86% on the data that should be most similar to the real data. Method that reads motifs from group of similar peptides worked also very well. It could identify 100% of motifs from groups of peptides where no noise was added and 90% of motifs from noisier peptide groups. The developed method could be also used for motif discovery on different biological datasets. In that case we would have to change some parameters that were specifically chosen for this problem. Future work could be to test how well this method performs on different biological datasets.

Keywords: motif discovery, hierarchical clustering, peptides

Tänusõnad

Tahaksin tänada oma juhendajaid Meelis Kulli ja Jaak Vilo ideede, nõuannete ja motiveerimise eest töö käigus tekkinud probleemide lahendamisel. Olen väga tänulik ettevõttele Protobios, kelle välja töötatud meetodi abil saadud andmete analüüsimiseks on antud töös kirjeldatud meetod koostatud. Eriline tänu Arno Pihlakule ja Anri Kivilile, kes lisaks teistele nõuannetele aitasid aru saada andmete bioloogilisest taustast. Tahan tänada ka uurimisrühma BIIT liikmeid, eriti Sven Lauri ja Balaji Rajashekari, nõuannete ja abi eest.

Sisukord

Sissejuhatus

Personaalmeditsiin on eesmärk, mille suunas on tehtud ning tehakse palju bioinformaatikaalast teadustööd. Idee seisneb selles, et teades inimese meditsiinilist tausta, geneetilist ja muud liiki informatsiooni, saavad arstid teha paremaid otsuseid tema haiguste ravimisel ja ennetamisel. Inimese kohta info saamiseks tehakse teiste uuringute seas erinevaid analüüse, näiteks tuvastatakse vereproovist teatud diagnostilise tähtsusega osakesi.

Inimese immuunsüsteemi jäävad salvestised põetud või põetavatest haigustest ning kui oleks võimalik neid salvestisi tuvastada, saaks koguda informatsiooni inimese haigusloo kohta või isegi haigusi diagnoosida. Hiljuti on loodud tehnoloogia, mis tuvastab inimese vereproovi abil sellised osakesed ning esitab need lühikeste aminohapete järjestuste ehk peptiididena [1]. Antud töös keskendume nendest peptiididest olulise signaali ehk sagedasti korduvate motiivide otsimisele. Leitud motiivid võivad sisaldada infot, mille edasisel analüüsimisel oleks võimalik tuvastada, milliseid haiguseid inimene põeb või on põdenud. Kuna ükski uuritud olemasolevatest tööriistadest probleemi lahendamiseks ei sobinud, kirjeldame antud töös autori poolt koostatud meetodit, mis hierarhilisele klasterdamisele põhinedes suudab kümnetest tuhandetest peptiididest tuvastada sagedased motiivid ning esitada need bioloogidele edasitöötamiseks sobivas formaadis. Koostatud meetodit on võimalik rakendada ka teistest sarnastest bioloogilistest andmetest motiivide tuvastamiseks, kuid selles töös meetodi töötamist teiste omadustega andmestikel ei valideerita.

Töö koosneb viiest peatükist. Esimeses peatükis tutvustame probleemi bioloogilist tausta ning termineid nagu peptiid ja motiiv, mida edaspidi kasutame. Teises peatükis anname ülevaate olemasolevatest vahenditest, mille abil sarnaseid probleeme lahendada on võimalik. Kolmandas peatükis räägime sünteetiliste andmete genereerimisest, mille peal hiljem koostatavat töövoogu testima hakkame. Neljandas peatükis kirjeldame täpselt koostatud töövoogu, mis põhineb hierarhilisele klasterdamisele. Viiendas peatükis näitame, kuidas koostatud töövoog sünteetiliste andmete peal töötab ning teeme järeldusi, millisel juhul töövoo erinevad variatsioonid paremini ja halvemini töötavad.

1 Bioloogiline taust

1.1 Bioloogiline meetod

Üks personaalse meditsiini andmete kogumise allikas on immunoloogia. Immuunsüsteem on inimese kaitsemehhanism erinevate võõrkehade, näiteks inimese normaalset funktsioneerimist häirivate viiruste ja bakterite vastu. Immuunsüsteem kaitseb inimest ka iseenda eest, leides ebatavalisi rakke, mis ei peaks meie kehas olema, näiteks vähirakke. Ühe tähtsa osa inimese immuunsüsteemist moodustavad antikehad. Nende eesmärgiks on märgistada organismi sattunud võõrkehad ehk antigeenid, nii et immuunsüsteem teaks need hävitada. Kui antikehad on mõne antigeeni vastu võidelnud, jätab immuunsüsteem selle meelde, nii et järgmine kord kui sama antigeen organismi satub, on antikehad valmis tegutsema. Sellele põhimõttele on üles ehitatud ka vaktsineerimine. Kui oleks teada, milliste antigeenide vastu inimese immuunsüsteem on valmis võitlema ehk antikehasid tootma, teaksime, milliseid haigusi on see inimene põdenud või põeb antud momendil.

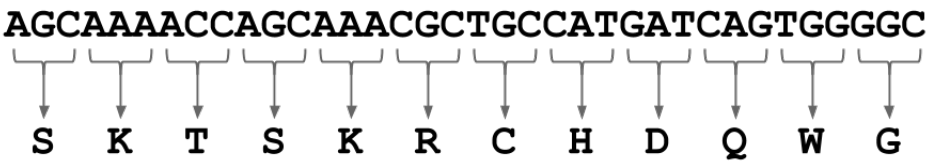
Seda osa antigeenist, mille järgi antikeha antigeeni ära tunneb, ehk millele antikeha seondub, nimetatakse epitoobiks. Välja on töötatud meetod nimega Mimotope Variance Analysis (MVA), mis organismi vereproovi põhjal suudab tuvastada, milliseid epitoope selle organismi antikehad ära tunnevad [1]. Lõpptulemusena peaks MVA abil leitud epitoopide abil suutma ära kirjeldada, milliste antigeenide vastu indiviidil antikehad on ning seega suutma kirjeldada tema põetud või põetavaid haigusi. MVA protsessi käigus filtreeritakse vereproovist välja antikehad, mis segatakse lühikeste järjestustega ehk mimotoopidega. Mimotoobid on kunstlikult genereeritud juhuslikud järjestused, mis matkivad antigeenidel olevaid epitoope. MVA protokollis segatakse mimotoobid antikehadega faagidisplei [2] kujul, mis kujutab endast hulka viiruseosakesi, kus iga osakese DNAsse on sisestatud erinev 36 nukleotiidist koosnev mimotoop. Vastavalt sisestatud mimotoobile seonduvad antikehad sobivate viiruseosakestega. Antikehaga seondunud viiruseosakesed eraldatakse ning nende DNA sekveneeritakse, et kätte saada mimotoobid, millega antikehad seondusid. Need mimotoobid on sisendiks edasisele töövoole.

1.2 Andmed

1.2.1 Peptiidid

Andmed, mis tulevad MVA analüüsist, jõuavad bioinformaatilisse analüüsi lühikeste, 36 nukleotiidist koosnevate järjestuste kujul. Need järjestused esindavad mimotoope, millega organismi antikehad seondusid. Ühe vereproovi MVA analüüsi käigus suudetakse tuvastada ligikaudu 3 miljonit järjestust. Algsed järjestused filtreeritakse, nii et sellised järjestused, mis sisaldavad vigu või on teadaolevad meetodi poolt tekitatud kõrvalnähud ja kontrolljärjestused ei satuks edasisse analüüsi. Järelejäänud järjestused transleeritakse aminohapete järjestuseks ehk peptiidideks nagu on demonstreeritud joonisel 1.1. Igale kolmesele nukleotiidijärjestusele ehk koodonile vastab üks aminohape. Seega tekivad transleerimisel peptiidid pikkusega 12. Peptiididеks transleerimine on vajalik, sest epitoobid ise on aminohapete järjestused. Antud töös kasutame mõistet peptiid kui järjestust, mis mimikeerib epitoopi, kuigi tegelikult võivad peptiidid esindada ka teistsuguseid bioloogilisi järjestusi.

Kuna ühe epitoobi äratundmiseks on organismis palju samasuguseid antikehi, võib andmestikku tekkida ühest peptiidist mitu koopiat. Iga peptiidi esinemiste arv loetakse kokku ning edasisse analüüsi jäetakse vaid sellised peptiidid, mis esinevad vähemalt kaks korda. Üksikud peptiidid eemaldatakse, sest tõenäoliselt on need MVA meetodi kõrvalnähud ja pole bioloogiliselt olulised. Üksikute peptiidide eemaldamise tagajärjel jääb erinevaid peptiide alles ligikaudu pool miljonit, kuid see arv võib sadades tuhandetes varieeruda kuna peptiidide mitmekesisus sõltub inimese immuunsüsteemist. Peptiidide esinemise arvud erinevad samuti palju, kahest kuni kümnete tuhandeteni. Väga sagedasi peptiide on vähe, ligi 90% peptiididest esineb vähem kui 10 korda. Seega on võimalik vähendada peptiidide arvu paarikümne tuhandeni, kui analüüsi jätta vaid väga sagedased peptiidid. Sagedased peptiidid eeldatakse olevat olulisemad, sest nad mimikeerivad epitoope, mille vastu on organismis palju antikehi, seega võivad nad viidata põetud või põetavale haigusele.



Joonis 1.1: Nukleotiidijärjestuse transleerimine peptiidiks.

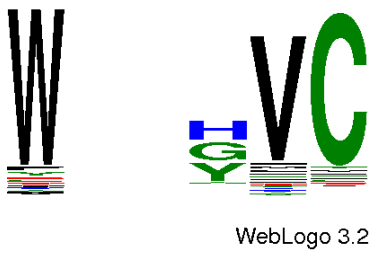
1.2.2 Motiivid

Antikeha seondub antigeenil asuva epitoobiga mitte ainult fikseeritud järjestuse alusel, vaid tihti on epitoobiks mingisugune muster ehk motiiv. Seega saavad sama epitoopi mimikeerida mitu erinevat peptiidi. Motiivi kirjeldatakse siinkohal kindlate reeglitega regulaaravaldise abil nagu on näidatud joonisel 1.2. Regulaaravaldise igal positsioonil saab olla üks kolmest variandist: kas konkreetne aminohape, aminohapete grupp või punkt, mis tähendab, et antud positsioonil võib olla ükskõik milline aminohape. Siinkohal seame reegliks ka selle, et motiiv ei tohi alata ega lõppeda punktiga. Seega on eesmärgiks leida mitte ainult sagedasti esinevaid peptiide, vaid ka sagedasti esinevaid motiive, kuna näiliselt erinevad peptiidid võivad siiski sisaldada sama motiivi ning sageli esinevad motiivid kirjeldavad sellisel antigeenil olevat epitoopi, mille vastu on konkreetsel organismil palju antikehasid.

```
W..[HGY]VC
```

Joonis 1.2: Näide regulaaravaldise kujul olevast motiivist.

Regulaaravaldise kujul olev motiiv ei suuda kirjeldada, kui oluline mõni positsioon motiivis on ning kas grupis olevad aminohapped on erineva olulisusega. Iga motiivi positsiooni kohta täpsema informatsiooni edastamiseks võib kasutada maatriksit, mille read tähistavad motiivi positsioone ning veerud erinevaid aminohappeid. Üks viis selline maatriks täita, on koostada tõenäosusmaatriks, kus iga element esitab tõenäosust, et antud positsioonil on vaadeldav aminohape. Tekkinud maatriksit on võimalik ka visualiseerida nagu on näidatud joonisel 1.3 ning kuigi lõpuks on vaja esitada motiivid regulaaravaldiste abil, aitab nende visualiseerimine bioloogidel leitud motiive paremini mõista. Tõenäosusmaatriksi koostamisest ja visualiseerimisest on pikemalt juttu osas 4.3.4.



Joonis 1.3: Näide tööriistaga WebLogo [3] visualiseeritud kaalumaatriksist.

Leides peptiididest sagedasti esinevad motiivid, on võimalik paarikümnes tuhandes peptiidis peituvat epitoopide infot esitada kompaktsemalt paarikümne või paarisaja motiiviga.

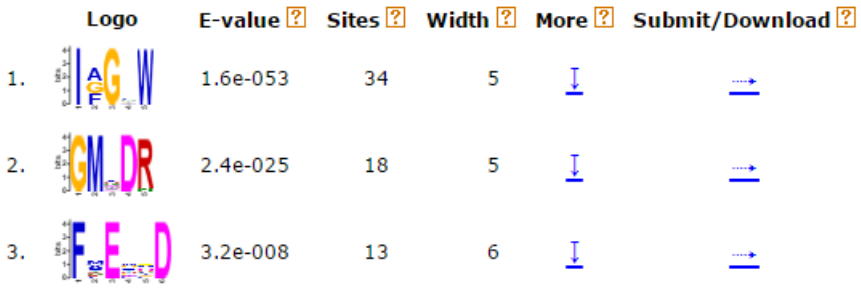
1.3 Bioloogilised küsimused

Leides inimese vereproovile tehtud MVA analüüsi käigus saadud peptiididest sagedased motiivid, saab vastata erinevatele bioloogilistele küsimustele. Esiteks saab kaardistada, millised motiivid peptiidides esinevad ning püüda teada saada, kas mõni leitud motiiv sobitud juba tundud haigusetekitaja valguga. Kui leida motiivid erinevatelt inimestelt, kellest ühel grupil on mõni haigus ning teisel mitte, saaks võrrelda, kas haigust põdevatel inimestel esineb motiive, mida haigust mittepõdevatel inimestel ei esine (või vastupidi). Sel viisil saaks leida markereid, mis kirjeldaksid erinevaid haigusi. Nendele markeritele põhinedes saaks välja töötada diagnostilisi meetodeid, mis vereproovi põhjal suudaksid tuvastada, kas inimene võib põdeda mõnda haigust.

Nende eesmärkide saavutamiseks on vaja meetodit, mis suudab etteantud peptiididest leida sagedased motiivid. Meetod peaks suutma töötada võimalikult paljude peptiididega ning tuvastama motiivid võimalikult täpselt. Bioloogide eesmärk on peptiididest kätte saada motiivid regulaaravaldiste ja visualiseeritud kaalumaatriksite kujul ning teada oleks vaja ka peptiide, kust motiiv leiti.

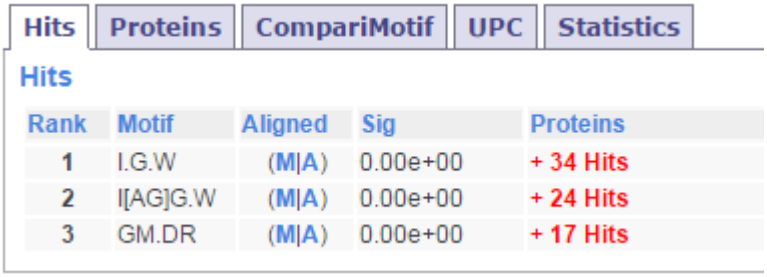
2 Kirjanduse ülevaade

Bioloogilistest järjestustest motiivide otsimiseks on olemas erinevaid tööriistu. Üks bioinformaatika valdkonnas sagedasti kasutatavatest tööriistadest on MEME [4]. MEME on tööriist, mis otsib järjestustest motiive ning esitab need regulaaravaldiste ning kaalumaatriksite kujul, näidates ära ka järjestused, mis iga motiivi alla kuuluvad. Seega sobib MEME ideeliselt väga hästi antud ülesandega. Lisaks on MEME veebitööriist seotud paljude erinevate tööriistadega nagu näiteks MAST [5], mis MEMEga leitud motiivide abil otsib sobivaid järjestusi etteantud järjestuste andmebaasist. MEME kasutab motiivide tuvastamiseks EM-algoritmi, mis on sisendjärjestuste arvu suhtes kuupkeerukusega. Paari tuhande järjestuse analüüsimiseks kulub ligikaudu päev, 10 000 järjestuse analüüsimiseks nädal [6]. Seega ei ole MEME suurte andmete puhul väga kasutatav. MEMEt saab siiski kasutada kõige sagedasemate peptiidide peal, võttes analüüsi näiteks 1000 kõige sagedasemat peptiidi. Näide MEME väljundist on toodud joonisel 2.1, kust on näha, et lihtsa testandmestiku peal tuvastab MEME täpselt etteantud motiivid. Hiljuti on avaldatud programm EXTREME [7], mis on MEMEga sarnane, kuid töötab mitmeid kordi kiiremini. EXTREME siiski antud probleemi lahendamiseks ei sobi, sest töötab vaid nukleotiidijärjestustega.



Joonis 2.1: MEME väljund andmete peal, kuhu on sisestatud kolm vigadeta motiivi: I[AGF]G.W, mida sisaldab 35 peptiidi, GM.DR, mida sisaldab 17 peptiidi ja F.E..D, mida sisaldab 17 peptiidi. Andmete genereerimisest on kirjutatud peatükis 3.

Erinevaid nukleotiidijärjestustest motiivide leidmise tööriistu on mitmeid, kuid proteiinidega töötavaid programme on vähem ning need on tihti väga spetsiifilised mõne kindla ülesande jaoks. Näiteks programm MUSI [8] eeldab, et andmetes on üks motiiv, mille erinevad variatsioonid tuleb üles leida. MUSI on küll väga kiire, kuid nii erinevate motiividega ja müraga, mis antud andmetes on, hakkama ei saa. The Gibbs Motif Sampler on samuti proteiini motiivide tuvastamise tööriist, kuid soovib, et täpsustatakse eraldi iga otsitava motiivi pikkus [9]. Dilimot leiab ainult ilma gruppideta regulaaravaldisi [10]. SLiMFinder on samuti üks tööriist, mis proteiinijärjestustest lühikesi lineaarseid motiive otsib [11]. See meetod põhineb BLASTil [12] ja kahest fikseeritud aminohappest koosnevate motiivide leidmisel ja kombineerimisel. SLiMFinder näib aga leidvat palju erinevaid versioone samast motiivist. Lihtsa testandmestiku peal ei saanud SLiMFinder nii hästi hakkama kui näiteks MEME. SLiMFinderi väljund on demonstreeritud joonisel 2.2.


Hits | Proteins | CompariMotif | UPC | Statistics

Hits

Rank	Motif	Aligned	Sig	Proteins
1	I.G.W	(M|A)	0.00e+00	+ 34 Hits
2	I[AG]G.W	(M|A)	0.00e+00	+ 24 Hits
3	GM.DR	(M|A)	0.00e+00	+ 17 Hits

Joonis 2.2: Näide SLiMFinderi väljundist samadel sisendandmetel, mida on kasutatud joonise 2.1 juures. Parima kolme motiivi hulka mahtus kaks varianti samast motiivist ning üks motiiv ei ole parimate hulka jõudnud.

Veel üks lähenemine järjestustest sagedaste motiivide leidmisele on SPEXS [13]. SPEXS võtab sisendiks kaks erinevat järjestuste komplekti: ühe, mille järjestused peaksid sisaldama otsitavaid motiive ning teise taustandmestiku, mis neid motiive sisaldama ei peaks. SPEXS püüab tuvastada motiivid, mis esimeses andmestikus olemas on ning teises mitte. Seda tehakse, genereerides kõikvõimalikud erinevad motiivid, mis esimese andmestiku järjestuste peal on ning sealt sobivaid välja filtreerides. SPEXS leiab üles küll olulised motiivid, kuid need motiivid esitatakse kui regulaaravaldised ilma grupisümboliteta. Grupisümboleid saab lisada vaid neid eelnevalt defineerides, kuid uusi gruppe otsingu jooksul moodustada ei saa. Teiseks probleemiks on see, et SPEXS tuvastab palju motiive, mis on omavahel veidi ülekattes või sarnased. Seepärast tuleks motiividele teha järeltöötlus, et sealt lõplik hulk mitteülekattes olevaid motiive kätte saada. Lühikeste motiivide puhul on aga keeruline öelda, et millised motiivid on piisavalt sarnased, et need

ühendada ning millised piisavalt erinevad. Näide SPEXSi väljundist on toodud joonisel 2.3.

MOTIIV	ESINEMISI SISENDIS	ESINEMISI TAUSTAS	SUHE	P-VÄÄRTUS
I.G.W	34	1	17.5	5.223818703076247e-12
F.E..D	17	0	18	2.435891603033169e-06
G..DR	17	0	18	2.435891603033169e-06
GM.DR	17	0	18	2.435891603033169e-06
GM..R	17	0	18	2.435891603033169e-06
M.DR	17	0	18	2.435891603033169e-06

Joonis 2.3: Näide SPEXSi väljundist samadel sisendandmetel, mida on kasutatud joonise 2.1 juures.

Lisaks mainitud tööriistadele oleme proovinud probleemi lahendamiseks koostada erinevaid meetodeid. Üks lihtne meetod on sorteerida peptiidid esinemise arvu järgi ning leida kõige sagedasemale peptiidile teised sarnased peptiidid ning siis seda korrata kuni kõik peptiidid on läbi käidud. See lähenemine tekitab aga olukorra, kus algpeptiidist sõltub liiga palju ning leitavad motiivid ei ole piisavalt üldised. Teine lahendus, mida siiamaani ka kasutanud oleme, on tuvastada motiivid põhinedes hierarhilisele klasterdusele [14]. See meetod toimib seni proovitutest kõige paremini, kuid suureks puuduseks on sarnastest peptiidigruppidest motiivide välja lugemise täpsus. Lisaks vajab meetod hetkel palju parameetreid. Antud töös koostatav meetod on varasema meetodi edasiarendus, kus püüame parandada motiivide lugemise täpsust ning asendada mitmed fikseeritud parameetrid dünaamiliselt leitud parameetritega. Samuti valideerime koostatavat töövoogu sünteetiliste andmete peal.

3 Sünteetiliste andmete loomine

3.1 Sünteetiliste andmete omadused

Kuna pärisandmete kohta ei ole teada, millised tulemused on õiged, on töövoo testimiseks vaja genereerida sünteetilised andmed, mille puhul on teada soovitav tulemus. Seejärel saab töövoogu valideerida sünteetiliste andmete peal ning oletades, et pärisandmed on neile sarnased, võime eeldada, et meetod peaks töötama ka reaalsete andmete peal. Andmete genereerimisel kasutatud parameetrid on valitud nii, et sünteetilised andmed oleksid sarnased tegelikele andmetele, kuid oleksid siiski lihtsama struktuuriga.

Andmete genereerimise juures on vaja teada, milliste omadustega on pärisandmed. Bioloogilistest taustast sõltuvalt teame, et andmetes peaksid olema motiivid pikkusega 3-8. Teame, et väga vähestes peptiidides olevad motiivid ei paku eriti palju huvi, sest võivad moodustuda juhuslikult. On alust arvata, et andmetes on juhuslikke peptiide, mille kohta ei saa küll öelda, et nad oleksid ebaolulised, aga mis ei sisalda ühtegi väga sagedast motiivi ning sellise analüüsi kontekstis võib neid käsitleda mürana. Nende peptiidide suhe olulistesse peptiididesse ei ole teada, kuid mida rohkem neid on, seda keerulisemaks teeb see oluliste motiivide tuvastamise.

Erinevad sünteetilised andmed on genereeritud kolme omaduse abil: motiivide arv andmestikus, juhuslike peptiidide protsent andmestikus ning motiivi täpsus peptiidides.

3.2 Motiivide arv ja suurus

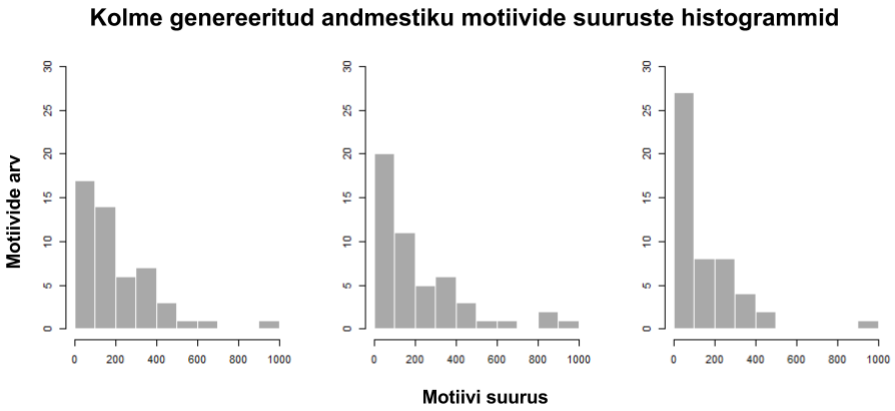
Esimene samm andmestiku genereerimisel on fikseerida motiivide arv. Testandmestikud said loodud 50 motiiviga, kuna katsetused pärisandmetel on esialgu näidanud, et andmetes on vähemalt paarkümmend motiivi.

Teine samm on igale motiivile vastavate peptiidide arvu ehk motiivi suuruse fikseerimine. Motiivi suurused said valitud vahemikust 30-1000. Alumine piir on valitud selle järgi, et motiiv peab sisaldama piisavalt palju peptiide, et bioloogid

saaksid sellega edasi töötada. Väga vähestes peptiidides esinevad ehk väga väikese signaaliga motiivid võivad samuti bioloogilises mõttes olulised olla, kuid nende olulisust on keerulisem tõestada ning seega on nendega keerulisem edasi töötada. Motiivide suuruste ülemist piiri päris andmete puhul teada ei ole. Kindlasti võib seal olla oluliselt suurema peptiidide arvuga motiive kui 1000, kuid eeldame, et meetodil ei tohiks suuremate motiivide leidmisega raskusi tulla, sest mida tugevam on signaal, seda kergem on seda tuvastada. Motiivide suuruste jaotus ühe andmestiku sees on genereeritud eksponentsiaalsest jaotusest parameetriga 0,5 ning teisendatud valemi 3.1 põhjal selliseks, et minimaalne väärtus oleks 30 ja maksimaalne 1000. Muutuja r tähistab vektorit, kus on n eksponentsiaalsest jaotusest genereeritud arvu. Vektor m saadakse vektori r teisendamisel sellisele kujule, et minimaalne väärtus vektoris m oleks 0. Vektor s sisaldab lõplikke suuruseid, kus minimaalne suurus on 30 ja maksimaalne suurus 1000. Eksponentsiaalne jaotus sai valitud eelnevate eksperimentide põhjal, mille tulemustest järeldasime, et päris andmetes näib olevat vähem sagedasi motiive ja rohkem väikese peptiidide arvuga motiive. Parameeter 0,5 tagab, et suurte motiivide hulk oleks siiski märgatav.

$$
\begin{aligned}
r_i &\sim Exp(0.5), \quad i = 1, ..., n \\
m_i &= r_i - min(r) \\
s_i &= \frac{m_i \cdot (1000 - 30)}{max(m)} + 30
\end{aligned}
\tag{3.1}
$$

Näide kolme 50 motiiviga andmestiku jaoks genereeritud motiivide suurustest on joonisel 3.1. Nagu näha, siis enamus motiive on väiksemad kui 200 ning suuremad klastrid on harvemad.



Joonis 3.1: Kolme genereeritud andmestiku motiivide suuruste histogrammid.

3.3 Motiivide genereerimine

Järgmiseks sammuks on motiividele vastavate regulaaravaldiste genereerimine. Motiivid genereerisime pikkustega vahemikust 3-8. Eelnevate analüüside põhjal nägime, et enamik motiive oli pikkusega 5 või 6 ning lühemad ja pikemad motiivid olid haruldasemad. Motiivide pikkused on genereeritud tõenäosustega, mis on näidatud tabelis 3.1.

Tabel 3.1: Motiivi pikkuse tõenäosused.

Motiivi pikkus	3	4	5	6	7	8
Genereerimise tõenäosus	0,10	0,17	0,23	0,23	0,17	0,10

Järgmisena peab määrama, mitu positsiooni motiivis on fikseeritud, ehk mitme koha peal on kindel aminohape. Fikseeritud positsioone peaks olema üldjuhul vähemalt 3 (suurima tõenäosusega) või 4, harva ka 2 või 5. Alumine piir tuleb sellest, et muidu on motiiv liiga üldine. Ülemine piir võiks ka suurem olla, kuid see teeks motiivi leidmise lihtsamaks ning loodav töövoog peaks selliste motiividega hakkama saama. Fikseeritud positsioonide arv genereeritakse tõenäosustega, mis on näidatud tabelis 3.2. Kui genereeritud fikseeritud positsioonide arv on suurem kui motiivi pikkus, siis võetakse fikseeritud positsioonide arvuks motiivi pikkus. Seetõttu tekivad erinevad genereerimise tõenäosused ja esinemise tõenäosused.

Tabel 3.2: Fikseeritud positsioonide arvu genereerimise tõenäosused ja esinemise tõenäosused.

Fikseeritud positsioonide arv	2	3	4	5
Genereerimise tõenäosus	0,05	0,50	0,40	0,05
Esinemise tõenäosus	0,05	0,55	0,37	0,04

Määrata tuleb ka juhuslike positsioonide arv ehk mitme koha peal motiivis võib olla suvaline aminohape. Selliste positsioonide arv genereeritakse tõenäosuste alusel, mis on toodud tabelis 3.3. Eesmärgiks sai, et juhuslikke positsioone ei oleks väga palju, kuid enamikes motiivides oleksid need siiski olemas. Kui genereeritud juhuslike positsioonide arv on suurem kui vabade positsioonide arv, siis võetakse juhuslike positsioonide arvuks vabade positsioonide arv.

Tabel 3.3: Juhuslike positsioonide arvu genereerimise tõenäosused ja esinemise tõenäosused.

Juhuslike positsioonide arv	0	1	2	3	4
Genereerimise tõenäosus	0,15	0,35	0,35	0,10	0,05
Esinemise tõenäosus	0,31	0,38	0,25	0,05	0,01

Ülejäänud positsioonide arv tähistab gruppide arvu motiivis, grupipositsioonide arvu esinemise tõenäosused on näidatud tabelis 3.4.

Tabel 3.4: Grupipositsioonide arvu tõenäosused.

Grupipositsioonide arv	0	1	2	3	4	5
Esinemise tõenäosus	0,50	0,19	0,16	0,10	0,04	0,01

Kui on teada erinevat tüüpi positsioonide arvud, tuleb määrata, kus asuvad motiivis juhuslikud positsioonid, fikseeritud positsioonid ja grupipositsioonid. Esimesena valitakse juhuslikud positsioonid nii, et motiiv ei algaks ega lõppeks juhusliku positsiooniga. Järgmisena valitakse fikseeritud positsioonid vabade kohtade hulgast ning ülejäänud positsioonid on grupisümbolite jaoks.

Pärast seda genereeritakse iga mittejuhusliku positsiooni jaoks aminohape või aminohapete grupp. Kui tegemist on fikseeritud positsiooniga, valitakse selle koha peale üks juhuslik aminohape. Kui tegemist on grupipositsiooniga, genereeritakse kõigepealt juhuslikult grupi suurus vahemikust 2-4 ning seejärel valitakse gruppi juhuslikud aminohapped.

3.4 Peptiidide genereerimine

Kui iga motiivi jaoks on olemas regulaaravaldis, saab hakata genereerima peptiide. Motiivi suuruse järgi teame, mitu peptiidi seda motiivi sisaldama peaks. Peptiid genereeritakse nii, et motiiv oleks selles peptiidis juhusliku koha peal. Kõik fikseeritud positsioonid täidetakse vastava aminohappega, grupipositsioonide jaoks valitakse antud grupist juhuslik aminohape ning juhuslike positsioonide ning motiivist väljapoole jäävate positsioonide jaoks valitakse juhuslik aminohape.

Kirjeldatud viisil peptiidide genereerimine tagab, et iga peptiid sisaldab täpselt otsitavat motiivi. Et teha andmed reaalsemaks, tuleks osadesse peptiididesse tekitada vigu, nii et kõik peptiidid ei sisaldaks motiivi täpselt, vaid et mõne fikseeritud või grupisümboli juures oleks sellele mittevastav aminohape. Iga peptiidi korral valitakse vigade arv Poissoni jaotusest parameetriga $\lambda = 1$. See tagab, et palju

peptiide on 0 või 1 veaga, mõned 2 või 3 veaga ning rohkemate vigadega peptiidid on harvad. Vigade tõenäosus Poissoni jaotusest parameetriga 1 on näha tabelis 3.5. Vigu tehakse ainult fikseeritud positsioonidesse ja grupipositsioonidesse ning kui juhuslikult genereeritud vigade arv on suurem kui vastavate positsioonide arv, siis võetakse vigade arvuks fikseeritud positsioonide ja grupipositsioonide summa. Seetõttu tekivad ka siin eraldi genereerimise ja esinemise tõenäosused.

Tabel 3.5: Vigade arvu tõenäosused Poissoni jaotusest, kus $\lambda = 1$.

Vigade arv	0	1	2	3	4
Genereerimise tõenäosus	0,37	0,37	0,18	0,06	0,02
Esinemise tõenäosus	0,37	0,37	0,18	0,07	0,01

Genereerisime ka rohkemate vigadega andmestikke, mis genereeriti Poissoni jaotusest parameetriga $\lambda = 2$. See tagab, et vigadeta peptiide on vähem, 1 ja 2 veaga peptiide on kõige rohkem, 3 ja 4 veaga veidi vähem ning rohkemate vigadega peptiidid on harvad. Nende vigade arvu jaotust kirjeldab tabel 3.6.

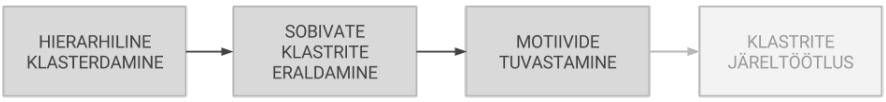
Tabel 3.6: Vigade arvu tõenäosused Poissoni jaotusest, kus $\lambda = 2$.

Vigade arv	0	1	2	3	4	5	6
Genereerimise tõenäosus	0,14	0,27	0,27	0,18	0,09	0,04	0,01
Esinemise tõenäosus	0,14	0,27	0,27	0,22	0,08	0,02	0,00

Viimasena lisasime andmestikku müra juhuslike peptiidide sisestamise teel. Juhusliku peptiidi genereerimiseks valiti igale peptiidi positsioonile juhuslik aminohape. Müra tasemed varieerusid nii, et 0%, 25%, 50% või 75% andmetest moodustaksid juhuslikud peptiidid. Kokkuvõttes genereeriti 12 erinevat andmestikku 50 motiiviga, mis olid vigade tasemetega 0, 1 või 2 ning juhuslike peptiidide protsendiga 0%, 25%, 50% või 75%. Andmestike suurused varieerusid vastavalt müra tasemele 10000 peptiidist 50000 peptiidini. Edaspidi kasutame konkreetsele sünteetilisele andmestikule viitamiseks lühendit stiilis J25-V1, mis tähendab, et andmestikus on 25% juhuslikke peptiide ja vigade tase on 1. Kõige sarnasemad eeldame päris andmetele olevat andmestikke, kus vigade tase on 1 ning müra vähemalt 50%.

4 Motiivide leidmise töövoog

Peptiididest motiivide leidmiseks koostasime töövoo, mis põhineb hierarhilisel klasterdamisel. Klasterdamise käigus tuvastatakse sarnaste peptiidide grupid ehk klastrid. Leitud klastrite seast valitakse sobivad ning iga sobiva klastri puhul loetakse välja klastri peptiidides peituv sarnane osa ehk motiiv. Motiivid esitatakse regulaaravaldiste ja tõenäosusmaatriksite kujul. Pärast esialgsete klastrite leidmist on võimalik teha ka klastrite järeltöötlus. Ülevaadet motiivide tuvastamise töövoost kirjeldab joonis 4.1.

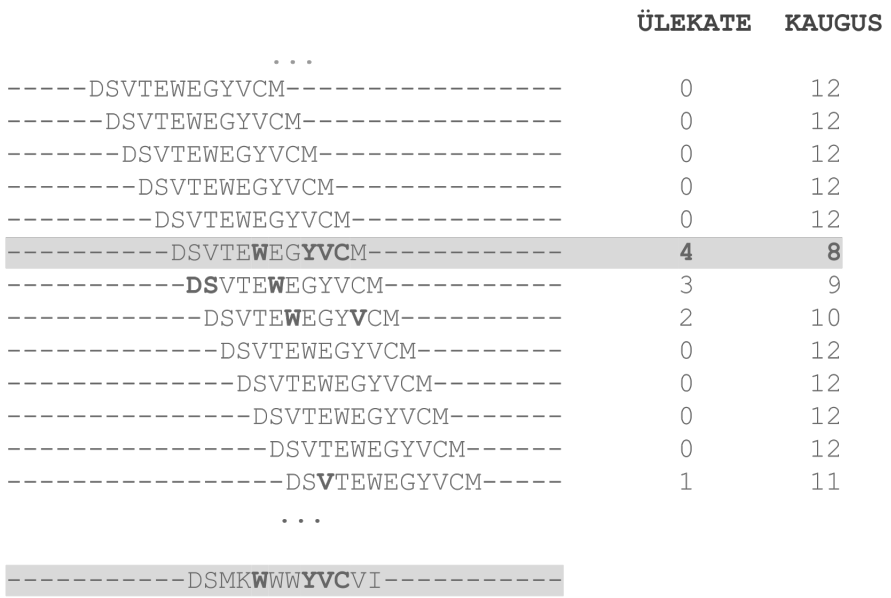


Joonis 4.1: Ülevaade motiivide leidmise töövoo põhietappidest.

4.1 Hierarhiline klasterdamine

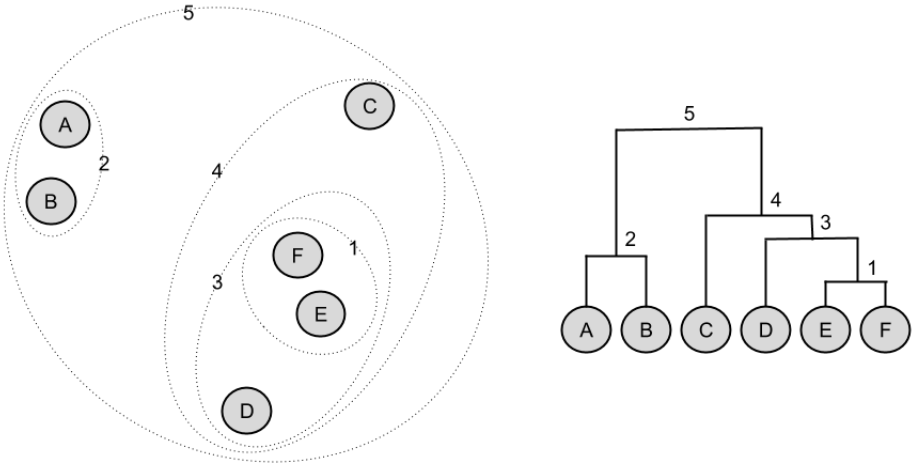
Sarnaste peptiidiklastrite tuvastamiseks otsustasime kasutada hierarhilist klasterdamist, kuna see meetod ei vaja töötamiseks leitavate klastrite arvu, nagu näiteks k-keskmiste meetod, ning on näidanud võrreldavate meetodite seast kõige paremaid tulemusi näidisandmetel. Alternatiivselt saaks kasutada ka teistsuguseid lähenemisi klastrite tuvastamiseks, kuid antud töös keskendume just hierarhilise klasterdamise abil saadud klastrite töötlemisele.

Aglomeratiivse ehk ühendava hierarhilise klasterdamise puhul käsitletakse alguses iga peptiidi kui eraldi klastrit. Kõikide peptiidide vahel on välja arvutatud nendevahelised kaugused. Antud andmetel arvutatakse kahe peptiidi vaheline kaugus, paigutades peptiide üksteise suhtes kõikvõimalikesse erinevatesse positsioonidesse ning leides, mitu aminohapet on parima positsiooni puhul samad. Kauguseks on peptiidi pikkusest lahutatud kattuvate aminohapete arv. Sellise kauguse leidmine on demonstreeritud joonisel 4.2. Vastavalt andmete bioloogilisele päritolule võib kauguse defineerida ka teisiti.



Joonis 4.2: Peptiidide DSMKWWWYYVCVI ja DSVTEWEGYVCM vahelise kauguse arvutamine. Esimene tulp näitab ülekatet võrreldava peptiidiga ning teine tulp kaugust võrreldavast peptiidist. Lõplikuks kauguseks saab vähim võimalik kaugus.

Klasterdamise esimese sammuna leitakse kaks kõige sarnasemat klastrit (alguses üksikut peptiidi) ja ühendatakse need uueks klastriks. Seejärel ühendatakse järgmised kaks kõige sarnasemat klastrit. Seda protsessi jätkatakse, kuni on alles ainult üks suur klaster, mis koosneb kõikidest elementidest. Kahe klastri omavaheline kaugus leitakse kui kahe klastri peptiidide kauguste keskmine. Klastritevahelise kauguse võiks defineerida ka kui kahe lähima või kahe kaugeima peptiidi vahelise kauguse, kuid eksperimendid näidisandmetel on näidanud, et kui andmestikus on juhuslikke elemente, töötab keskmine kaugus kõige paremini, sest teised meetodid sõltuvad liiga palju üksiku peptiidi järjestusest. Hierarhilise klasterdamise tulemusena tekib puustruktuur võimalikest klastritest ehk dendrogramm nagu on näidatud joonisel 4.3. Kahe klastri ühendamise kõrgus puus näitab, mis on nende kahe klastri omavaheline kaugus. Seega kasutame edaspidi mõisteid kaugus ja kõrgus samaväärselt.



Joonis 4.3: Näide hierarhilisest klasterduse olemusest. Vasakpoolne pilt kujutab elemente kahemõõtmelises ruumis. Parempoolne pilt kujutab samu elemente hierarhilise klasterduse tulemusena tekkinud dendrogrammis. Numbrid näitavad, millises järjekorras klastreid ühendatakse.

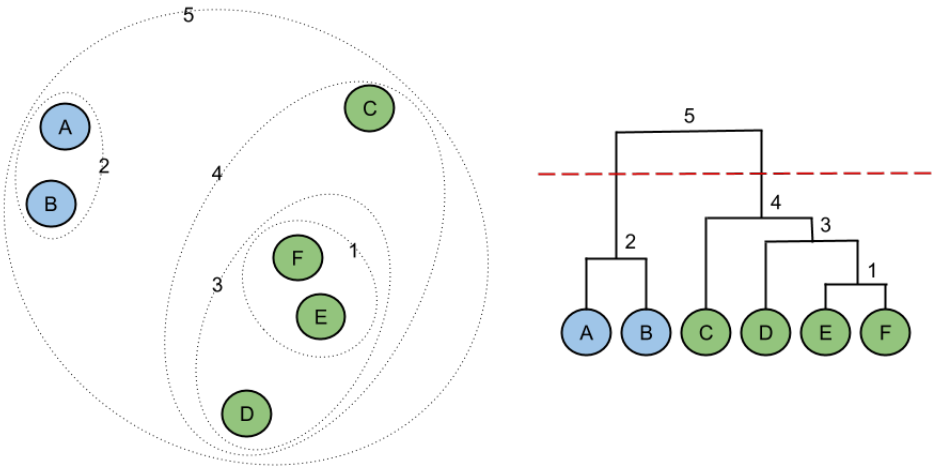
4.2 Sobivate klastrite eraldamine

Hierarhilise klasterdamise tulemusena tekkivast dendrogrammist ei ole võimalik kohe lõplikke klastreid kätte saada. Selleks peab dendrogrammist eraldama harud, mis klastriteks sobivad. Tihti tehakse seda tööd käsitsi ning valitakse sobivaid klastreid dendrogrammi visuaalselt inspekteerides, kuid häid, erinevates olukordades hästi töötavaid automaatseid lõikamismeetodeid selle probleemi lahendamiseks ei leidu. Üks lihtne viis oleks lõigata puu ühelt kõrguselt ning iga tekkiv haru moodustaks klastri. Teine viis oleks puu läbi käia ning klastreid eraldada selle järgi, kas parasjagu vaadatav haru vastab klastriks olemise tingimustele. Ühelt kõrguselt lõikamine on arvutuslikult vähem keerukas, kuid proovime mõlemat meetodit, et leida nende erinevused ning teada saada, millises olukorras erinevaid lõikamismeetodeid kasutada.

4.2.1 Ühelt kõrguselt lõikamine

Kõige lihtsam viis puust klastreid eraldada on ühelt kõrguselt lõikamine, kus klastriteks võetakse kõik harud, mis sellelt kõrguselt lõigates tekivad nagu on demonstreeritud joonisel 4.4. Meetod peaks töötama hästi, kui soovitavate klastrite kaugus on ligikaudu teada ning kui peptiidides on vähe vigu ja andmetes vähe juhuslikke elemente. Ühelt kõrguselt lõigates ei pruugi aga alati leida õigeid

klastreid näiteks juhul kui klastrid on väga erinevate kaugustega. Siis võib juhtuda, et mõni klaster on lõigataval kõrgusel kasvanud liiga suureks või mõni klaster pole veel lõigataval kõrgusel lõpuni moodustunud.



Joonis 4.4: Näide ühelt kõrguselt lõikamisest. Punktiirjoonega lõigates tekib kaks klastrit, ühte kuuluvad A ja B ning teise C, D, E ja F.

Kui on teada minimaalne otsitava motiivi pikkus, nagu antud hetkel on, siis on võimalik ligikaudselt välja arvestada lõikamise kõrgus. Näiteks hetkel otsime peptiididest pikkusega 12 motiive, kus oleks enamasti vähemalt kolm fikseeritud aminohapet. Seega on otsitava klastri enamikel peptiididel vähemalt kolm aminohapet ühised ning järelikult on maksimaalne klastrisisene ligikaudu 9. Tegelikkuses on see kaugus ilmselt küll veidikene suurem, kuna lubame peptiididesse vigu. Kuid niimoodi saab arvestada ligikaudse lõikamise kõrguse.

Puu lõikamisel filtreeritakse välja need klastrid, millest leitud motiiv ei ole pikkusega vahemikus 3-8 ja mis ei sisalda vähemalt 2 fikseeritud positsiooni ning mille fikseeritud ja grupisümbolite summa ei ole vähemalt 3. See tagab, et esitatakse ainult sobiva motiiviga klastrid.

4.2.2 Dünaamiline lõikamine

Kuna ühelt kõrguselt lõigates sõltub üsna palju valitud kõrgusest ning osad klastrid ei pruugi olla seal veel moodustunud või on kasvanud liiga suureks, püüame lahendada klastrite eraldamise probleemi ka dünaamilise meetodiga.

Ka dünaamilise meetodi puhul seatakse klastritele maksimaalse kõrguse piir, kuid klastriteks ei võeta kõiki ühelt kõrguselt lõigates tekkinud harusid, vaid puu käiakse läbi ülevalt allapoole liikudes ning alles siis, kui leitakse sobivate omadustega klaster, lisatakse see leitud klastrite hulka. See tagab näiteks, et liiga lühikese

motiiviga klastri puhul ei eemaldata seda analüüsist vaid vaadatakse tema alamklastreid, mis võivad olla juba piisavalt pika ja täpsema motiiviga. Lisaks peaks dünaamiline meetod vähendama olukordi, kus tulemusse satuvad mitme erineva motiiviga klastri ühendamisel tekkinud motiivid, kuna meetod kontrollib, et vaadeldava klastri otsesed alamklastrid oleksid temaga sarnased. Puud ülevalt alla läbi käies kontrollitakse iga tipu puhul klastriks olemise tingimuste kehtimist. Klastriks olemise tingimused on:

1. Peptiidide arv klastris peab olema suurem või võrdne minimaalse lubatud peptiidide arvuga.

   See tingimus on vajalik, et mitte leida väga väikeseid klastreid, mida ei olegi võimalik mõistlikult edasi töödelda.

2. Klastrisisene kaugus ei tohi olla suurem etteantud kaugusest.

   See parameeter tagab, et puud ei pea alates juurest läbi vaatama.

3. Klastri motiiv vastab etteantud tingimustele.

   Antud andmetes on tingimuseks, et motiivi pikkus ei tohi olla suurem kui 8, peab olema vähemalt 2 fikseeritud aminohappega positsiooni ning fikseeritud positsioonide ja grupipositsioonide summa peab olema vähemalt 3. See tagab, et valitakse ainult sobiva motiiviga klastrid.

4. Klastri motiiv peab olema sarnane oma otseste alamklastrite motiividega.

   Kontrollitakse, kas klastri motiiv on sarnane oma piisavalt suurte (võrreldakse minimaalse klastri suuruse parameetri abil) otseste alamklastrite motiividega. See tagab, et kui klaster erineb oma alamklastritest, siis järelikult on tema alamklastrid erinevad ning edasi tuleks vaadata hoopis neid. Motiivid loetakse sarnaseks juhul kui mõlemal on vähemalt kolm mittejuhuslikku positsiooni ning üks motiiv sobitub teisega nii, et motiivide vastavad positsioonid on sarnased. Positsioone loetakse sarnaseks kui

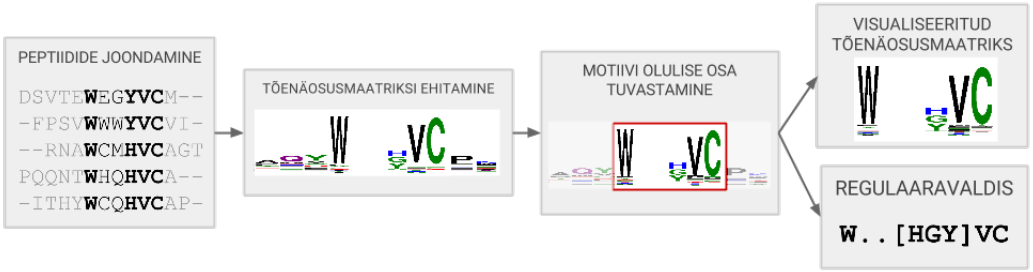
   - mõlema motiivi positsioonid tähistavad suvalist aminohapet;
   - ühe motiivi positsioonil on suvaline aminohape ja teise motiivi positsioonil on vähemalt kolmest elemendist koosnev grupp;
   - ühe motiivi positsioonil olevad aminohapped on teise motiivi positsioonil olevate aminohapete alamhulgaks.

   Sama meetodi alusel võrdleme ka edaspidi motiivide sarnasust.

Dendrogramm vaadatakse läbi ülevalt allapoole liikudes ning iga tipu korral kontrollitakse nende tingimuste kehtimist. Kui vaadeldav klaster ei vasta tingimustele, otsitakse sobivat klastrit alamklastrite hulgast. Kui vaadeldav klaster vastab etteantud tingimustele, siis määratakse see klaster sobivaks ning tema alamklastreid edasi ei vaadata.

4.3 Motiivide tuvastamine

Sobivatest klastritest on peale nende tuvastamist vaja leida motiivid ning esitada need regulaaravaldiste ning visualiseeritud tõenäosusmaatriksite kujul. Motiivi leidmise põhisammud on näidatud joonisel 4.5.



Joonis 4.5: Motiivi tuvastamise etapid.

4.3.1 Peptiidide joondamine

Sarnaste peptiidide klastris võib ühine motiiv paikneda peptiidides erinevatel positsioonidel. Motiivi tuvastamiseks on vaja peptiidid paigutada nii, et peptiidide sarnane osa asuks ühel positsioonil ehk peptiidid on vaja joondada. Näide peptiididest enne ja pärast joondamist on toodud joonisel 4.6.

```
DSVTEWEGYVCM      DSVTEWEGYVCM--
FPSVWWWYVCVI      -FPSVWWWYVCVI-
RNAWCMHVCAGT      --RNAWCMHVCAGT
PQQNTWHQHVCA      PQQNTWHQHVCA--
ITHYWCQHVCAP      -ITHYWCQHVCAP-
```

Joonis 4.6: Näide joondamata (vasakul) ning joondatud (paremal) peptiididest.

Kahe peptiidi joondamiseks kasutatakse dünaamilise programmeerimise meetodeid, et paigutada kaks järjestust selliselt, et nende ülekate oleks maksimaalne. Peptiidide puhul saab lisaks arvesse võtta ka seda, et mõned aminohapped on teineteisele sarnasemad. Antud ülesande puhul on oluline arvestada, et peptiidide joondamisel peptiidide keskele lünki tekitada ei tohi, kuna see ei oleks bioloogiliselt põhjendatud, lünki tohib lisada ainult peptiidi algusesse ja lõppu. Näide lünkadeta ja lünkadega joondusest on joonisel 4.7.

```
DSVTEWEGYVCM-      DSVTEWEGYVCM-
-DSMKWWWYVCVI      DS-MKWWWYVCVI
```

Joonis 4.7: Näide lünkadeta (vasakul) ning lünkadega (paremal) joondusest.

Joondamiseks kasutame programmi MAFFT, sest MAFFT on antud analüüsi jaoks piisavalt kiire (ning vajadusel paralleliseeritav) ning samas hea täpsusega [15]. MAFFT kasutab joondamiseks progressiivset meetodit, mis tähendab, et kõigepealt leitakse kõikide järjestuste omavahelised kaugused, mille põhjal ehitatakse esialgne puu kasutades hierarhilist klasterdamist. Seejärel hakatakse tekkinud puu põhjal joondust ehitama, lisades tulemusjoondusesse ükshaaval uusi klastreid või peptiide.

4.3.2 Tõenäosusmaatriksi ehitamine

Joondatud peptiidides on otsitav motiiv kõikide peptiidide puhul samal positsioonil. Niimoodi on võimalik kokku lugeda, millised aminohapped igal positsioonil esinevad ning tugevalt ülekaalus olevad aminohapped mingil positsioonil viitavad sellele, et antud positsioon on motiivis oluline ning peaks koosnema just nendest aminohapetest. Selliste positsioonide leidmiseks saab koostada tõenäosusmaatriksi. Tõenäosusmaatriks T on $n \times m$ maatriks, kus n tähistab positsioonide arvu ning m tähistab võimalike sümbolite arvu, mis on antud juhul aminohapete arv ehk 20. Maatriksi reas n ja veerus m olev element tähistab seda, kui tõenäoline on, et antud positsioonil esineb just see aminohape. Positsioonil i oleva aminohappe j tõenäosus T_{ij} leitakse kui aminohappe j protsent positsioonil i olevatest kõikidest aminohapetest (lünki arvestamata). Näide tõenäosusmaatriksist on joonisel 4.8.

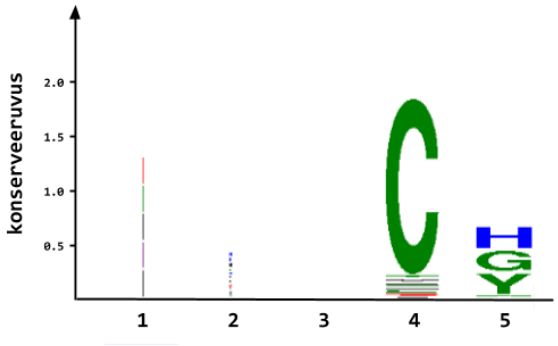
	A	C	D	E	F	G	H		Q	R	S	T	V	W	Y	KAAL	KONS
1	0,00	0,00	0,00	0,20	0,00	0,20	0,00		0,20	0,00	0,00	0,00	0,20	0,00	0,00	0,01	1,39
2	0,04	0,00	0,04	0,04	0,00	0,08	0,13		0,00	0,08	0,00	0,04	0,08	0,08	0,00	0,03	0,46
3	0,03	0,06	0,05	0,06	0,04	0,05	0,05	...	0,06	0,05	0,04	0,05	0,08	0,03	0,05	0,83	0,03
4	0,01	0,80	0,01	0,01	0,01	0,01	0,01		0,01	0,01	0,01	0,02	0,02	0,01	0,01	0,98	1,91
5	0,01	0,02	0,02	0,02	0,02	0,24	0,25		0,02	0,01	0,02	0,03	0,02	0,01	0,24	1,00	0,84

Joonis 4.8: Näide tõenäosusmaatriksist, mis esitab motiivi pikkusega 5.

Lisaks leitakse iga positsiooni kohta kaal, ehk mitu protsenti moodustavad sellest positsioonist aminohapped. Kui positsiooni kaal on 1, siis sellel positsioonil ei ole ühtegi lünka. Kui positsiooni kaal on 0, koosneb antud positsioon ainult lünkadest. Iga positsiooni kohta arvutatakse ka konserveeruvus, mis on vahe maksimaalse võimaliku entroopia ja antud positsiooni entroopia vahel (valem 4.1). Seega on minimaalne võimalik konserveeruvus 0, mis tähendab, et kõik aminohapped esinevad positsioonil võrdselt, ning maksimaalne $\log_2(20) \approx 4{,}32$, mis tähendab, et positsioonil on esindatud ainult üks kindel aminohape.

$$C = \log_2(20) - (-\sum_{n=1}^{20} p_n \log_2 p_n) \tag{4.1}$$

Tõenäosusmaatriksi kujul oleva motiivi arvutamiseks ja visualiseerimiseks kasutame tööriista WebLogo [3]. Visualiseerimisel kujutab WebLogo iga motiivi positsiooni järgmiselt. Positsiooni kõrguse määrab positsiooni konserveeruvus ning laiuse positsiooni kaal. Iga aminohappe kõrguse tulba sees määrab tema tõenäosus sellel positsioonil. Joonisel 4.9 on demonstreeritud joonise 4.8 positsioonide visualiseerimine WebLogo abil.

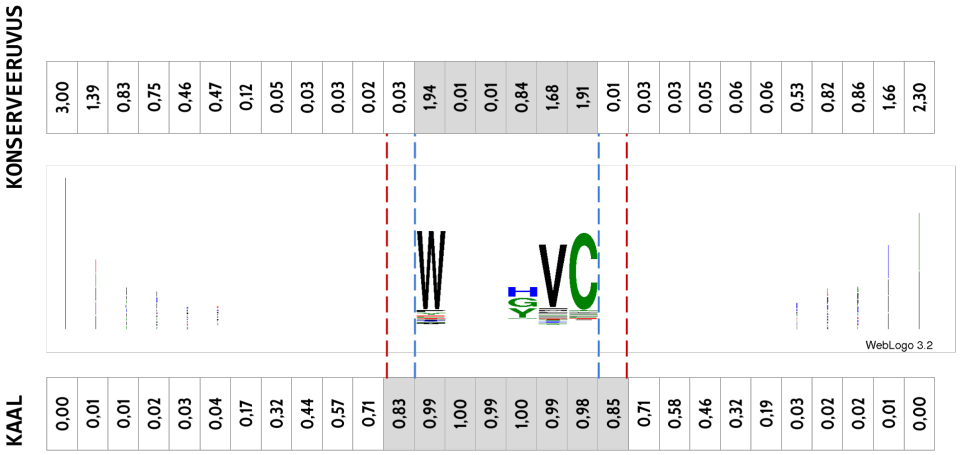


Joonis 4.9: Tõenäosusmaatriksi visualiseerimine.

4.3.3 Motiivi olulise osa tuvastamine

Joondamisel tekivad joonduse äärde positsioonid, mis koosnevad suures osas lünkadest ja ei ole motiivi leidmisel olulised. Motiivi esitamise teeks seega paremaks nende äärepositsioonide eemaldamine. Seda tehakse kahes osas.

Esialgne lõikamine toimub positsioonide kaalude põhjal. Motiivi mõlemast äärest eemaldatakse positsioone kuni esimese kaaluni, mis on $\geqslant 0{,}8$ ehk positsioonini, kus on vähemalt 80% aminohappeid. 80% piir sai valitud kui pigem madal piir - sellest suurema lünkade protsendiga positsioonid tavaliselt motiivi olulist osa ei sisalda.



Joonis 4.10: Motiivi lõikamise näide. Peale kaalu järgi lõikamist näeme, et alles jäänud konserveeruvuste järjestamisel on madalaim piisavalt suur hüpe 0,03 pealt 0,84 peale. Seega võetakse piirideks esimesed äärepositsioonid, mis on kaaluga $\geqslant$ 0,8 ja konserveeruvusega $\geqslant$ 0,84.

Kaalude järgi lõigatud motiivi äärtesse võib jääda siiski ebaolulisi positsioone, mis on küll piisavalt suure kaaluga, aga liiga väikese konserveeruvusega, ehk positsioonil ei ole ükski aminohape ega piisavalt väike aminohapete grupp tugevamalt esindatud. Ka sellised äärepositsioonid oleks hea eemaldada. Et leida konserveeruvuse piir, mille järgi motiivi ääred eemaldada, sorteeritakse kõikide positsioonide konserveeruvuse väärtused ning alustades vähimast väärtusest leitakse selline konserveeruvus, mille juurest hüpe järgmise konserveeruvuse väärtuseni on $\geqslant$ 0,5. Kui ükski konserveeruvus, mis oleks $\leqslant$ 1 sellist tingimust ei rahulda, siis võetakse konserveeruvuse piiriks 1. Selline dünaamiline piiri valimine on kasutusel seetõttu, et üldiselt motiivi positsioon kas on oluline või mitte. Olulised positsioonid on suure konserveeruvusega ja mitteolulised väga väikese

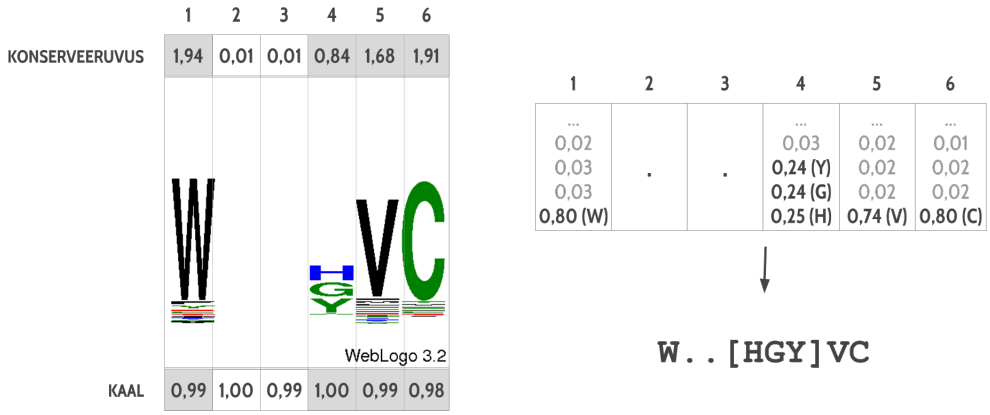
konserveeruvusega. Seega otsimegi esimest kohta, kus toimub väärtustes piisavalt suur hüpe. Alumist kohta otsime seetõttu, et ülemiste parimate väärtuste vahe võib olla samuti suur, sest fikseeritud positsioonid on suurema konserveeruvusega kui grupipositsioonid. Hüppe minimaalne suurus 0,5 on valitud kui väikene hüpe, et võtta motiivi pigem rohkem positsioone kui vähem. Kui madalal kohal sellist hüpet ei ole, siis järelikult on kõik positsioonid tähtsad ning piiriks võetakse 1. Viimane olukord ei näi olevat väga sage ning kui see juhtub, siis tavaliselt seetõttu, et kõik positsioonid on motiivis olulised ning seejuures on konserveeruvus tavaliselt $\geqslant 1$. Motiivi lõikamist demonstreerib joonis 4.10.

4.3.4 Regulaaravaldise lugemine

Eelmise sammu tulemusena tekkis tõenäosusmaatriksi kujul olev lõigatud motiiv, mis visualiseeritud kujul ongi üks motiivi lõplikest esitustest. Motiivi esitamiseks regulaaravaldisena tuleb tõenäosusmaatriksist leida olulised positsioonid ja esitada need sobival kujul. Selleks määratakse igale motiivi positsioonile üks kolmest võimalikust variandist: suvaline aminohape, grupp või kindel aminohape.

- Positsioonid, mille kaal on väiksem kui 0,8 ja konserveeruvus väiksem kui eelmises sammus leitud konserveeruvuse piir, on suvalised positsioonid ning nende kohale regulaaravaldises määratakse punkt.
- Ülejäänud positsioonide puhul vaadatakse eraldi aminohapete tõenäosusi. Kui kõige sagedasema aminohappe tõenäosus on $< 0{,}1$, siis ei ole antud positsioon oluline ja ühtegi aminohapet motiivi ei lisata ning tulemusse kirjutatakse samuti punkt.
- Kui leidub aminohape, mille tõenäosus on $\geqslant 0{,}1$, sorteeritakse aminohapete tõenäosused ja leitakse nende vahed. Kui maksimaalse vahe suurus on $\geqslant 0{,}1$, siis võetakse tõenäosuse piiriks vahe suurem väärtus. Teisel juhul saab piiriks 0,1. Pärast tõenäosuse piiri leidmist valitakse välja kõik aminohapped, mis on sellest suuremad või sellega võrdsed.
  - Kui selliseid aminohappeid ei olnud üldse või oli liiga palju (antud juhul oleme võtnud maksimaalseks grupi suuruseks 4), siis lisatakse regulaaravaldisse punkt.
  - Kui selliseid aminohappeid oli 2-4, lisatakse motiivi nendest aminohapetest koosnev grupp
  - Kui sellised aminohappeid oli üks, siis lisatakse motiivi see aminohape.

Suurima tõenäosuste erinevuse leidmist kasutame, sest hea positsiooni puhul on üldjuhul hästi eristatavad olulised aminohapped, mis on suure tõenäosusega, ning ebaolulised aminohapped, mis on väikese tõenäosusega. Regulaaravaldise leidmise näide on joonisel 4.11.



Joonis 4.11: Regulaaravaldise lugemise näide. Kõigepealt valitakse välja positsioonid, mille konserveeruvus ja kaal on sobivad (vasakul). Seejärel sorteeritakse iga sobiva positsiooni tõenäosused, leitakse sobivad aminohapped ja moodustatakse regulaaravaldis (paremal).

4.4 Järeltöötlus

Esialgsed klastrid ei pruugi olla veel lõplikud, sest võib juhtuda, et klastrite sobivuse parameetrite tõttu on jäänud osad klastri motiivi sisaldavad peptiidid klasterdamata. See võib näiteks juhtuda liiga madalalt lõikamisel, kus mõni klastri haru ei ole veel suurema klastriga, mis seda motiivi sisaldab, ühendatud. Kuna on oluline, et võimalikult palju motiivi sisaldavatest peptiididest oleks õigesse klastrisse klasterdatud, rakendame klasterdamata peptiididele järelklasterdust.

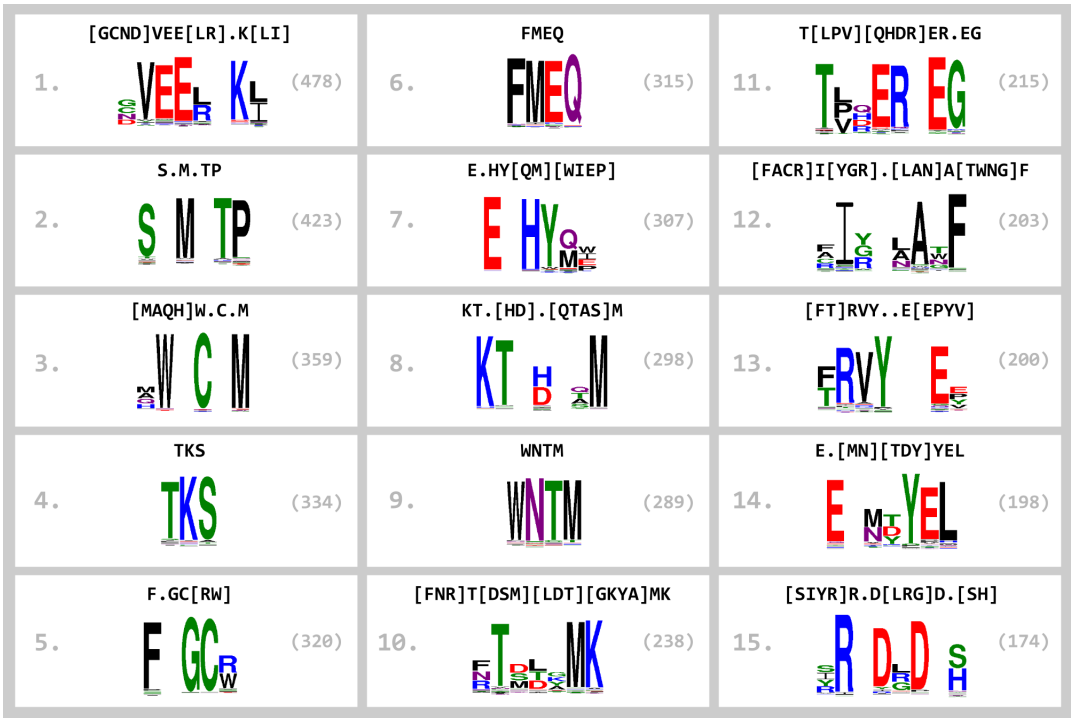
Järelklasterdusel vaadatakse läbi kõik klasterdamata peptiidid ning iga peptiidi puhul kontrollitakse, kas see sobib mõne leitud klastri motiiviga. Peptiidi ja motiivi sobivuse hindamiseks vaadatakse, kas peptiid sisaldab antud motiivi. Soovi korral võib lubada motiivi ja peptiidi võrdlemisel ka vigu. Kuid siiski peaks peptiid sisaldama vähemalt kolme mittejuhuslikku motiivi positsiooni (nende hulgas vähemalt kahte fikseeritud positsiooni). Kui peptiid sobib mitme motiiviga, valitakse nende hulgast see, millega on kõige rohkem ühiseid mittejuhuslikke positsioone. Kui selliseid motiive on mitu, valitakse motiiv, millel on peptiidiga enim

ühiseid fikseeritud positsioone. Kui ka neid motiive on mitu, siis lisatakse peptiid suurema motiivi klastrisse.

Järeltöötlusesse võib soovi korral veel erinevaid filtreid või korrektuure lisada. Näiteks võib kontrollida, kas on tekkinud sarnaseid klastreid ning püüda neid ühendada. Antud meetodis piirdume aga peptiidide järelklasterdusega.

5 Tulemused ja võrdlused

Koostatud töövoo tulemused esitatakse motiividena, mis on sorteeritud motiivide suuruse järjekorras. Iga motiivi kohta on välja toodud regulaaravaldis ja visualiseeritud tõenäosusmaatriks. Lisaks on teada ka igale motiivile vastavad peptiidid. Näide töövoo poolt väljastatud suurimatest motiividest sünteetilisel andmestikul J50-V1 on joonisel 5.1.



Joonis 5.1: Loodud töövoo poolt leitud 15 suurimat motiivi andmestikul J50-V1 (kõrguse parameetriga 9,5). Kõik leitud 15 suurimat motiivi vastavad täpselt mõnele sisestatud motiivi regulaaravaldisele. Sulgudes olev number tähistab leitud klastri suurust.

Valideerimaks loodud töövoogu, katsetame seda sünteetilistel andmetel. Vaatame, kuidas erinevad kaks lõikamise meetodit ning mis on nende eelised ja puudused. Vaatame ka seda, kui hästi motiivi regulaaravaldise tuvastamise meetod töötab ning kas ja kuidas järelklasterdamine parandab klasterdatud peptiidide protsenti. Töövoo puhul on võimalik muuta kahte peamist parameetrit: minimaalse motiivi suuruse parameetrit ning maksimaalse klastrisisese kauguse ehk kõrguse parameetrit (ühelt kõrguselt lõikamise puhul on see lõikamise kõrguseks). Edaspidi kutsume neid vastavalt suuruse ja kõrguse parameetriteks. Töövoo katsetamisel valisime suuruse parameetriks 20. Kuigi päriselt teame, et minimaalne sisestatud motiivi suurus on 30, vaatame juhtu, kus lubame ka väiksemaid motiive. See teeb ülesande keerulisemaks, kuna juhuslikest ja vigadega peptiididest võivad moodustuda motiivid, mida pole tegelikult andmetesse sisestatud. Paralleelselt vaatame ka seda, kuidas tulemused erinevad, kui minimaalse suuruse parameetrit suurendada. Kõrguse parameeter sai kolm võimalikku väärtust: 9, 9,5 ja 10. Kõige sobivamaks kõrguseks arvame olevat kõrguse 9,5, sest see kõrgus võiks ligikaudselt arvestades esitada sobivat maksimaalset klastrisisest kaugust. Aga vaatame ka seda, mis juhtub, kui lõigata puud kõrgemalt ja madalamalt.

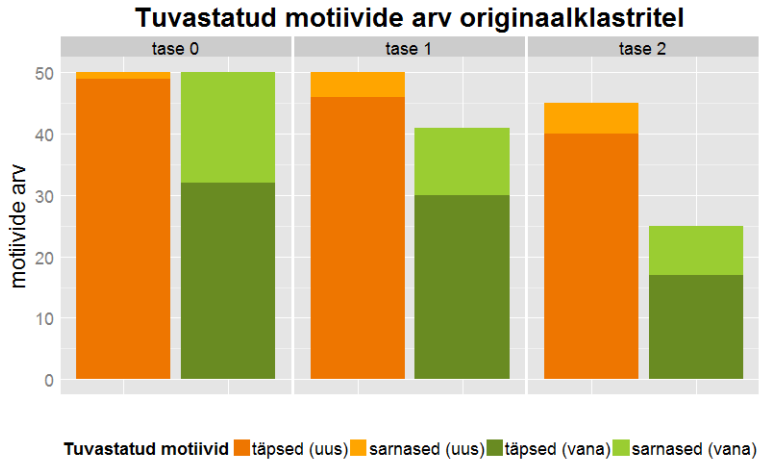
Sünteetilised andmestikud on joonistel reastatud nii, et andmete keerukus kasvab vasakult paremale. Neli vasakpoolset tulpa tähistavad nelja vigadeta peptiididega andmestikku kasvava juhuslike peptiidide protsentide järjekorras (J0-V0, J25-V0, J50-V0, J75-V0). Neli keskmist tulpa on vigade tasemega 1 (J0-V1, J25-V1, J50-V1, J75-V1) ja viimased neli vigade tasemega 2 (J0-V2, J25-V2, J50-V2, J75-V2).

5.1 Motiivide tuvastamine

Enne klasterdamise tulemuste valideerimist vaatame, kui täpne on motiivide tuvastamise meetod, sest sisestatud motiivide leidmise kontrollimiseks kasutame regulaaravaldisi ning seega peame olema kindlad, et klastritest motiivide lugemine töötab hästi.

Motiivide lugemise täpsuse kontrollimiseks võtsime ühe testandmestiku 50 klastrit ning püüdsime lugeda igast klastrist seal oleva motiivi. Paralleelselt püüdsime motiive lugeda ka varasemas meetodis [14] kasutusel olnud motiivi tuvastamise meetodiga, mis põhimõttelt on sarnane praegusega, kuid dünaamilise konserveeruvuse ja aminohapete tõenäosuse piiride asemel kasutatakse eeldefineeritud piire. Jooniselt 5.2 näeme, et uue meetodi abil suudame tuvastada tunduvalt rohkem motiive kui vana meetodiga. Tumedamalt on joonisel märgitud täpselt tuvastatud motiivide arv ning heledamalt motiivide arv, mis olid küll piisavalt sarnased originaalmotiivile, kuid mitte täpsed. Vigadeta peptiidide puhul suudavad

mõlemad meetodid leida kõik motiivid, kuid uus meetod suudab peaaegu kõik tuvastada täpselt. Vigade suurenedes suudab uus meetod saada väga head tulemused (tuvastamata jääb kuni 5 motiivi) kuid varasem meetod leiab eriti vigade taseme 2 juures üsna vähe motiive. Seega võib öelda, et motiivide tuvastamise meetod on vana meetodiga võrreldes tunduvalt täpsem.



Joonis 5.2: Motiivide tuvastamise täpsuse võrdlus.

5.2 Klasterduse headuse hindamine

Järgmiseks vaatame, kui hästi suudetakse erinevate sünteetiliste andmestike korral tuvastada neisse sisestatud motiivid ning kuidas erinevad ühelt kõrguselt lõikamise ja dünaamilise lõikamise tulemused.

Hea klasterduse korral on tähtsad järgmised aspektid:

1. Leitakse üles võimalikult palju õigeid motiive.

   Selle mõõtmiseks loeme ära, mitmele originaalmotiivile vastab vähemalt üks leitud motiiv. Motiivide sarnasust mõõdetakse juba eelnevalt kirjeldatud regulaaravaldiste võrdlemise meetodi abil.

   Siin vaatame eraldi ka seda, mitu motiivi tuvastatakse täpselt.

2. Motiivid on järjestatud olulisuse järgi.

   Klasterdamise tulemusena ei teki ainult soovitud motiivid, vaid kuna andmetesse ja peptiididesse on sisestatud müra, võivad tekkida ka motiivid, mida eraldi sisestatud pole. Motiivid järjestatakse suuruse alusel ning kui originaalmotiividele vastavad motiivid asuvad järjestatud motiivide nimekirja alguses, on võimalik võtta leitud motiividest suurimad ning vaid neid edasi vaadata.

3. Iga motiiv esineb ainult ühe korra.

   Selleks vaadatakse, mitmel leitud motiivil on olemas duplikaat.

4. Originaalpeptiidid on võimalikult hästi klasterdunud ja leitud õiged motiivid sisaldavad õigeid peptiide.

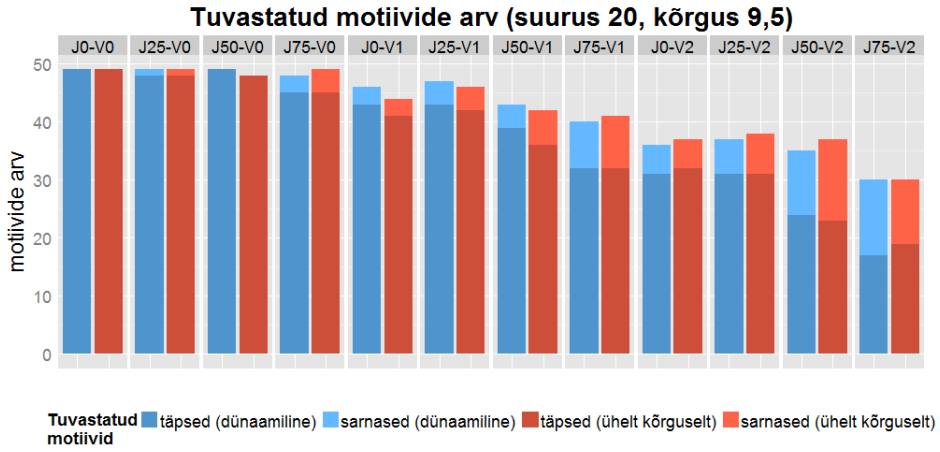
   Selle hindamiseks vaatame, mitu protsenti mittejuhuslikest peptiididest on klasterdunud õigetesse motiividesse ja mitu protsenti juhuslikesse motiividesse.

   Teise osa hindamiseks vaatame, mitu protsenti leitud õigete motiivide peptiididest on õigesti klasterdunud ning mitu protsenti on õigetesse motiividesse sattunud valesid peptiide.

5.2.1 Tuvastatud motiivide arv

Esimesena vaatame, mitu originaalmotiivi suudame mõlema lõikamismeetodiga tuvastada. Vaatame nii seda, mitu motiivi tuvastatakse täpselt kui ka seda, mitu motiivi ei olnud täpsed, kuid siiski piisavalt sarnased originaalmotiivile.

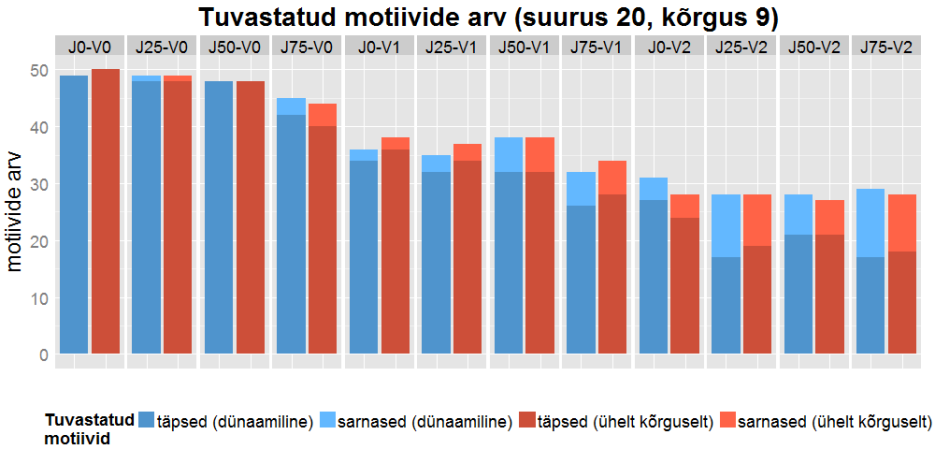
Joonis 5.3 näitab tuvastatud motiivide arvu, mis on leitud eeldatavalt sobivaima kõrguse parameetriga 9,5. Näeme, et enim mõjutab leitud motiivide arvu vigade tase ning vähem juhuslike peptiidide protsent. Vigadeta andmete korral saadakse peaaegu ideaalne tulemus ning suurima müratasemе korral leitakse ligi 30 motiivi. Ka leitud motiivide täpsus kahaneb müratasemе suurenedes. Lõikamismeetodite vahel leitud motiivide arvu puhul suurt erinevust ei ole.



Joonis 5.3: Tuvastatud motiivide arv (kõrgus 9,5).

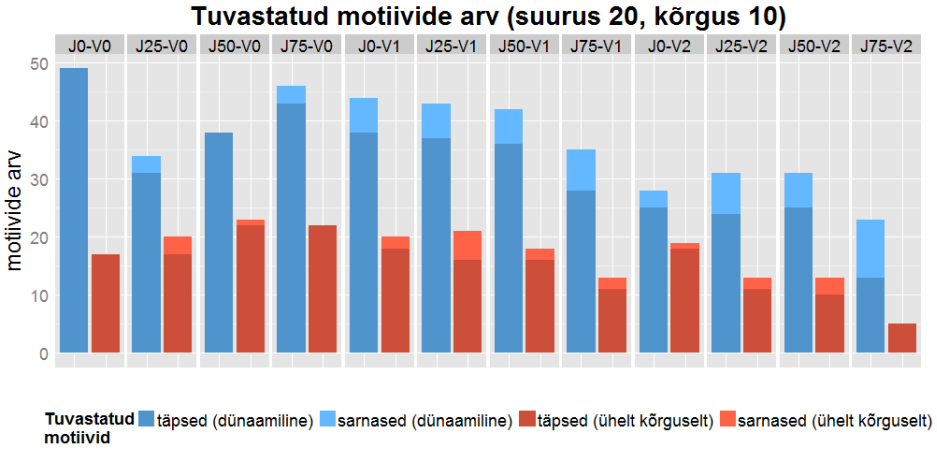
Joonis 5.4 näitab leitud motiivide arvu madalama kõrguse parameetriga. Siin on samuti mõlema lõikamise meetodi tulemused sarnased, kuid leitud motiivide

arv on väiksem kui parameetriga 9,5. Ilmselt ei ole kõik motiivid madalamalt lõigates lõpuni moodustunud ning seega ei suudeta neid tuvastada, eriti kui vigade arv peptiidides on suurem.



Joonis 5.4: Tuvastatud motiivide arv (kõrgus 9).

Joonis 5.5 näitab leitud motiivide arvu suurema kõrguse parameetriga. Siin näeme mõlema meetodi puhul tuvastatud motiivide arvu langemist, kuid dünaamiline lõikamine suudab siiski üsna head tulemused anda. Ühelt kõrguselt lõikamine ei ole liiga suure kõrguse puhul enam sobiv, sest lõigates saadakse mitme motiivi segunemisel tekkinud motiivid, mis ei sarnane enam originaalmotiividega. Dünaamiline lõikamine töötab paremini, sest liiga üldise motiivi leidmisel liigutakse puus allapoole kuni leitakse sobivad motiivid.



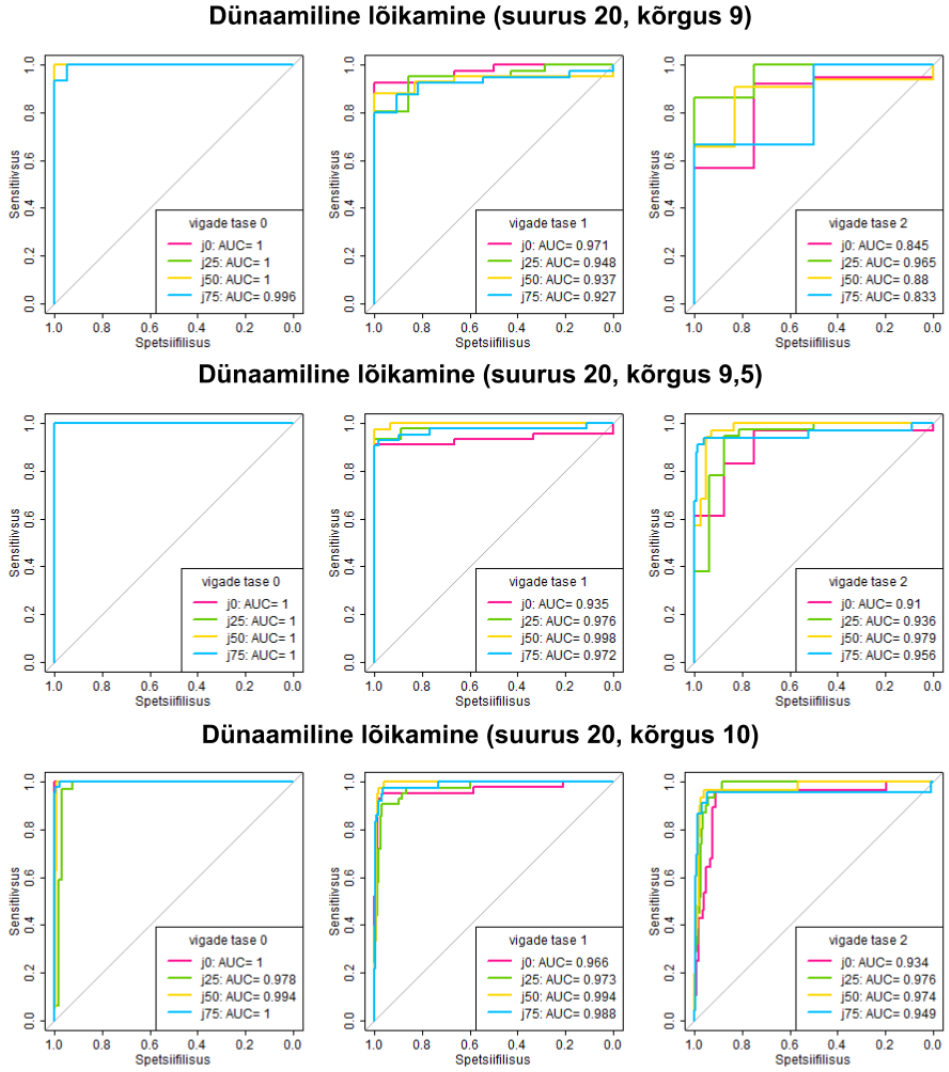
Joonis 5.5: Tuvastatud motiivide arv (kõrgus 10).

Vaadates erinevate kõrguse parameetritega tuvastatud motiivide arvu, näeme, et enamik leitud motiividest on suudetud tuvastada täpselt samasugusena nagu nad sisestati ning väiksel osal tuvastatud motiividest on mõni viga, kuid motiiv on siiski sarnane originaalmotiiviga. Kokkuvõtvalt võib öelda, et õigelt kõrguselt lõigates töötavad meetodid sarnaselt, kuid kui lõikamise kõrgust on eelnevalt raske välja arvestada, tuleks eelistada dünaamilist lõikamist.

5.2.2 Motiivide järjestus

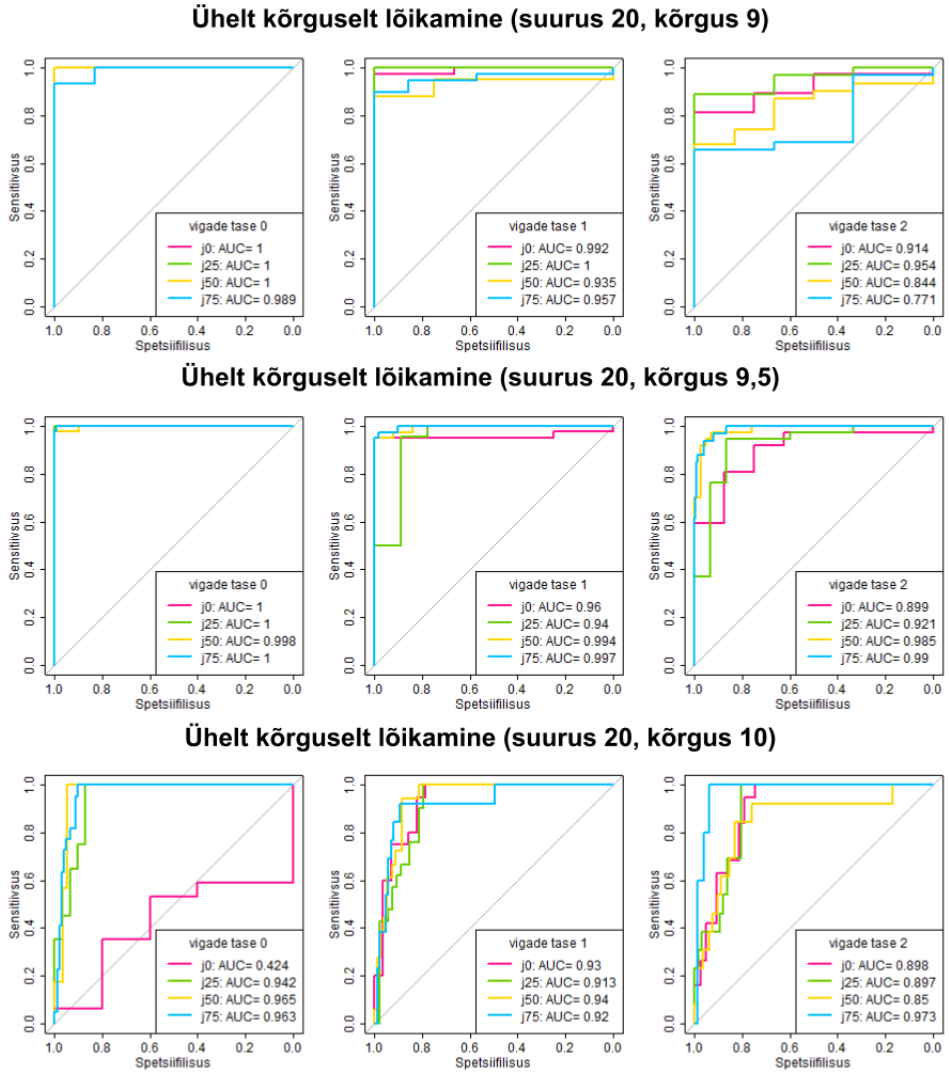
Kuna andmetesse on sisestatud müra, siis leitakse ka originaalmotiividele mittevastavaid motiive, mis on moodustunud juhuslikest ja vigadega peptiididest. Hea tulemuse korral on oluline, et järjestades motiivid suuruse alusel, oleks õiged motiivid kõige suuremate seas ja juhuslikult tekkinud motiivid väiksemate motiivide seas. Seega vaatame, kuidas on sünteetilistelt andmestikelt leitud motiivid järjestatud. Järjestuse hindamiseks joonistame iga tulemuse kohta ROC-kõvera. Järjestuse kohta joonistatud ROC-kõver kirjeldab, kui hästi erineva suuruse piiri järgi õigeid ja juhuslikke motiive on võimalik eraldada. Sensitiivsus näitab, mitu protsenti originaalmotiividest tuvastatakse ehk on suuruseks valitud piirist suuremad. Spetsiifilisus näitab, mitu protsenti valedest ehk juhuslikest motiividest tuvastatakse kui valed motiivid, ehk mitu protsenti valedest motiividest jääb valitud suuruse piirist allapoole. Ideaalse järjestuse korral peaks kõik originaalmotiividele vastavad motiivid olema enne juhuslikke motiive ehk peaks olema võimalik valida selline motiivi suuruse piir, et sellest väiksemad motiivid on juhuslikud ning suuremad mittejuhuslikud. Sel juhul oleksid sensitiivsus ja spetsiifilisus 100%. ROC-kõvera headust hinnatakse ROC-kõvera aluse pindala abil (AUC), mis 100% sensitiivsuse ja spetsiifilisuse puhul on 1.

Joonisel 5.6 on dünaamilise lõikamise tulemustele vastavad ROC-kõverad. Iga rida tähistab erinevat kõrgust ning iga veerg erinevat vigade taset. Ühel graafikul on koos kõik sellelt kõrguselt ja vigade tasemelt saadud tulemused (4 erinevat juhuslike motiivide protsenti). Näeme, et vigade tasemega 0 leitud motiivide järjestused vastavad peaaegu ideaalsele järjestusele ning mõnel juhul ongi võimalik suuruse järgi eristada kõik juhuslikud motiivid mittejuhuslikest. Mida suurem vigade tase, seda rohkem satub suuremate motiivide hulka ka juhuslikke motiive, kuid järjestus on siiski piisavalt hea ja eespool on suuremas osas õiged motiivid.



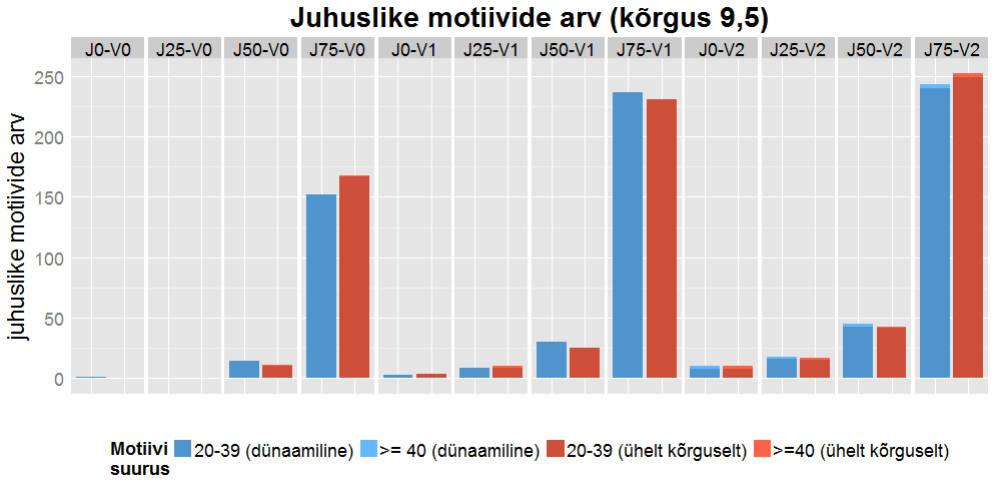
Joonis 5.6: ROC-kõverad dünaamilise lõikamisega saadud motiivide järjestuse hindamiseks.

Ühelt kõrguselt lõikamise tulemuste ROC-kõverad on joonisel 5.7. Tulemused on sarnased dünaamilise lõikamise tulemustega, kuid kõrguse parameetriga 10 saadud tulemused on veidi halvemad, sest nagu eespool mainitud, tekivad ühelt kõrguselt lõikamisel sel puhul suured klastrid, kus mitu motiivi on ühendatud. Seega leitakse ka rohkem suuri juhuslikke motiive.



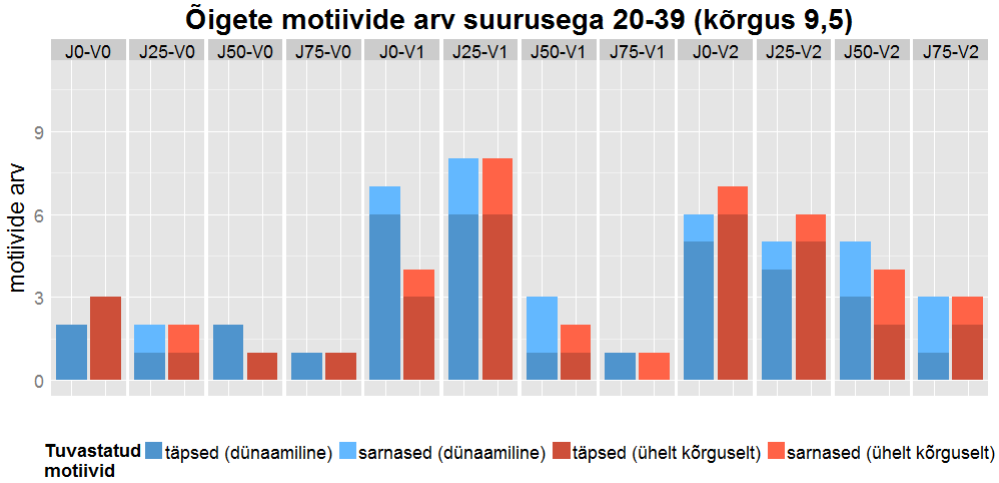
Joonis 5.7: ROC-kõverad ühelt kõrguselt lõikamisega saadud motiivide järjestuse hindamiseks.

Vaatame ka seda, kui palju juhuslikke motiive erinevatel sünteetilistel andmetel tekib ning mil määral saaksime vähendada nende arvu kui jätaksime alles ainult suuremad motiivid (suurus $\geqslant 40$). Joonis 5.8 näitab juhuslike motiivide arvu lõigates kõrguselt 9,5. Heledama värviga on näidatud juhuslike motiivide arv, mis on $\geqslant 40$ ning tumedamaga juhuslike motiivide arv, mille suurus jääb vahemikku 20-39. Nagu näeme, sõltub juhuslike motiivide arv tugevalt juhuslike peptiidide sisaldusest, kuid meetodite vahel suurt erinevust pole. Kõrgusega 9,5 tekib juhuslike motiive, mis oleksid $\geqslant 40$ väga vähe.



Joonis 5.8: Juhuslike motiivide arv (kõrgus 9,5).

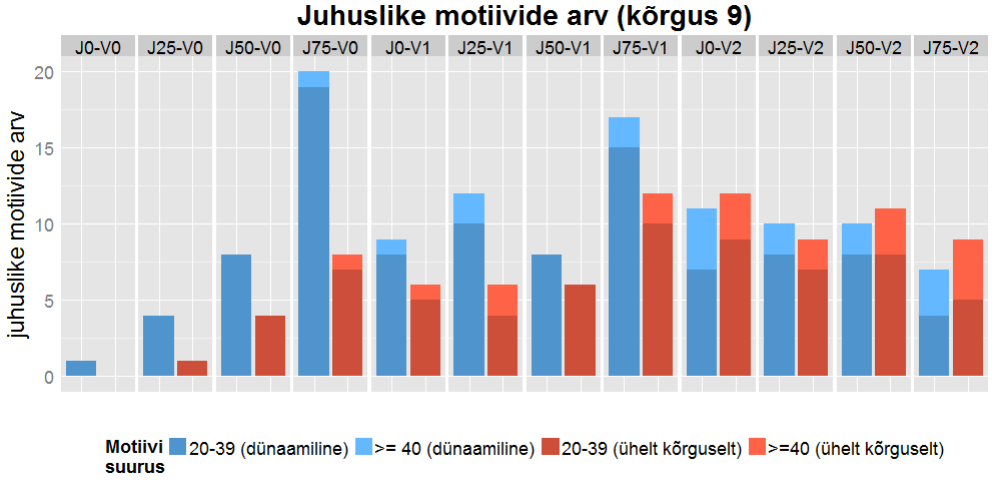
Suuruse järgi filtreerides suudame küll välja filtreerida üsna palju juhuslikke motiive, kuid selle käigus võivad kaduda ka mõned õigeid motiivid. Jooniselt 5.9 näeme kui palju häid motiive kaotaksime lubatud minimaalse suuruse tõstmisega. Kaotatud motiivide arv ei ole küll väga suur, kuid mida rohkem on peptiidides vigu, seda rohkem motiive niimoodi kaotame, sest ka õiged motiivid on peptiidides olevate vigade tõttu väiksemad.



Joonis 5.9: Õigete motiivide arv suurusega 20-39 (kõrgus 9,5).

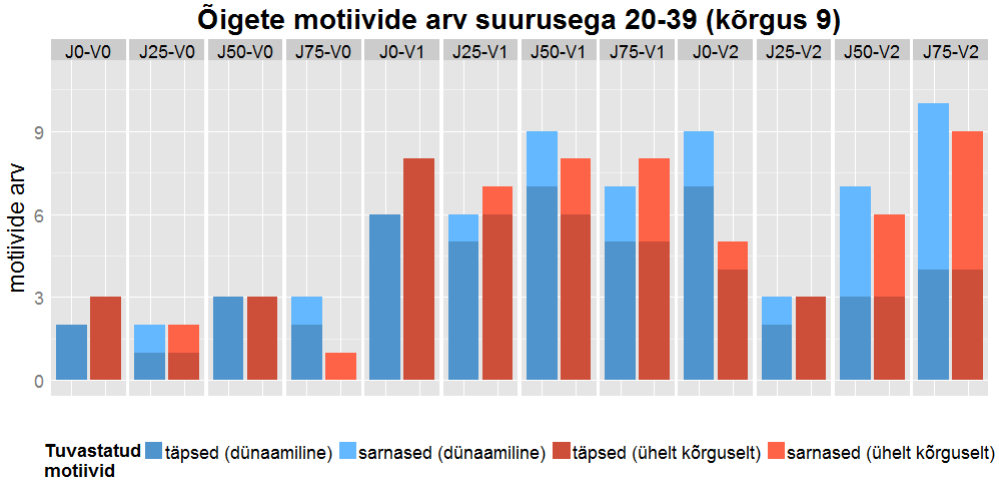
Madalamalt lõigates tekib juhuslikke motiive vähem nagu on näha jooniselt 5.10, sest madalamalt lõigates ei ole juhuslikud motiivid saanud piisavalt suureks

kasvada. Siin näeme ka erinevust ühelt kõrguselt lõikamise ja dünaamilise lõikamise vahel, kus viimane näib leidvat üldiselt rohkem juhuslikke motiive, eriti väiksema müratasemega andmete puhul.



Joonis 5.10: Juhuslike motiivide arv (kõrgus 9).

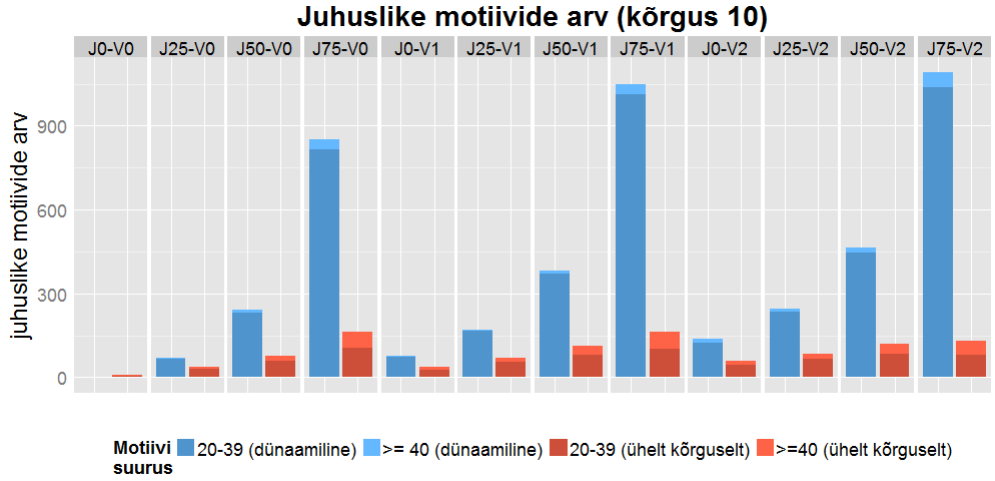
Jooniselt 5.11 näeme, et kaotatud motiivide arv on madalamalt lõigates suurem, sest madalamalt lõigates on ka õiged motiivid väiksemad.



Joonis 5.11: Õigete motiivide arv suurusega 20-39 (kõrgus 9).

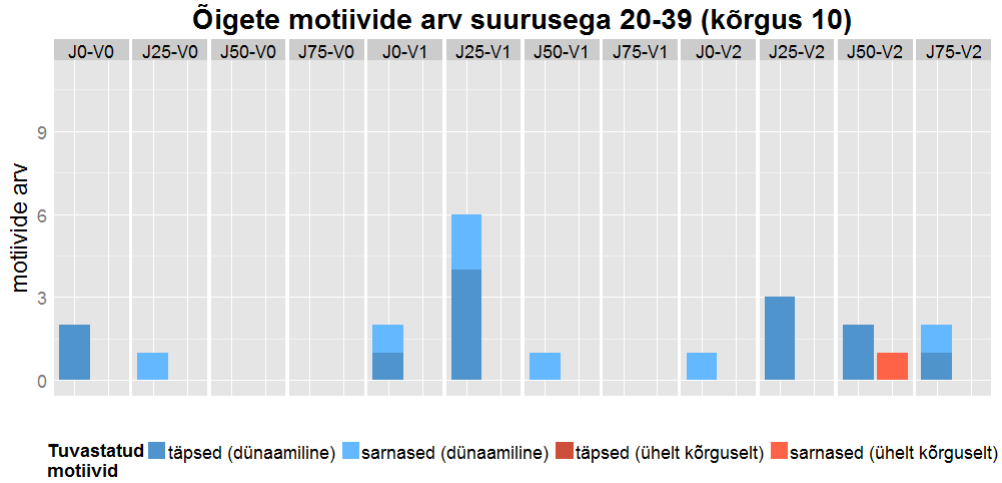
Kõrgemalt lõigates tekib juhuslikke motiive palju rohkem, sest tekkida saavad suuremad motiivid (joonis 5.12). Kui jätta analüüsi ainult suuremad motiivid, väheneb juhuslike motiivide arv märgatavalt, kuid suuremate juhuslike peptiididega

andmestike puhul jääb paarkümmend juhuslikku motiivi siiski alles. Siin näeme ka suurt erinevust kahe lõikamismeetodi vahel, mis tuleb sellest, et ühelt kõrguselt lõikamisel tekkivad liiga suured ja üldise motiiviga klastrid visatakse lihtsalt välja, aga dünaamiline lõikamine käib need harud läbi kuni leiab sobivad klastrid.



Joonis 5.12: Juhuslike motiivide arvu (kõrgus 10).

Joonisel 5.13 näidatud kaotatud motiivide arv on väga väike, sest head klastrid on sellelt kõrguselt lõigates piisavalt suured.

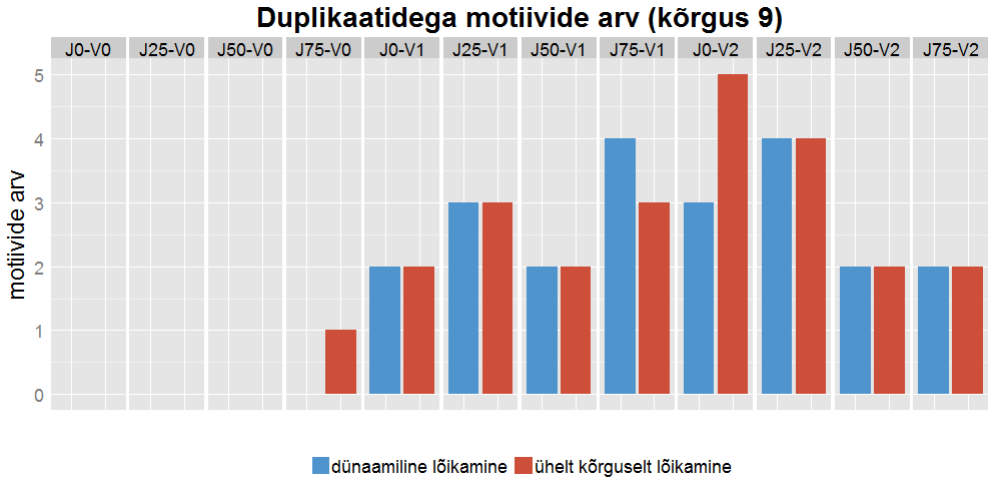


Joonis 5.13: Õigete motiivide arv suurusega 20-39 (kõrgus 10).

Kokkuvõttes võib öelda, et motiivide järjestus on sobiv ning kõige olulisemad motiivid asuvad üldjuhul eespool. Kui leitud motiivide arv tundub siiski liiga suur, saab välja filtreerida suuremad motiivid. Selle tagajärjel ei kaotata eriti palju õigeid motiive, kuid juhuslike motiivide arv väheneb märgatavalt.

5.2.3 Duplikaatide arv

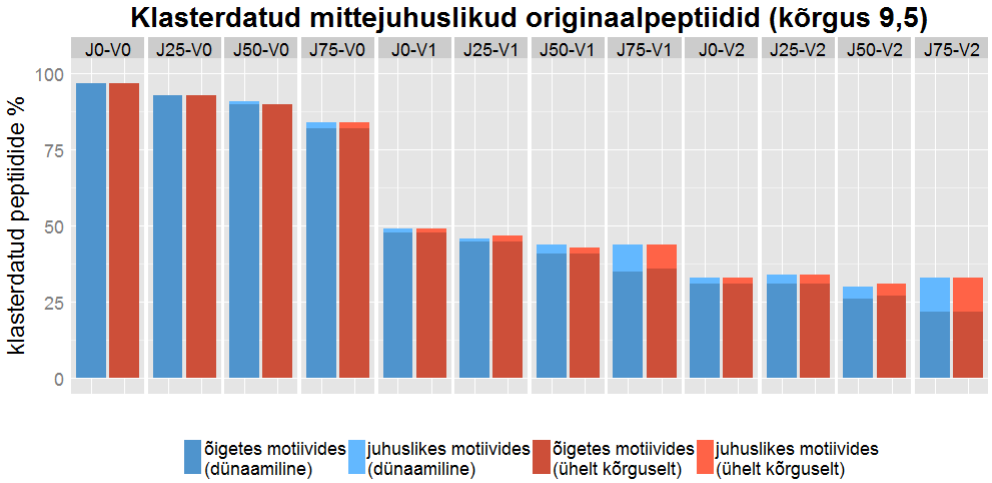
Klasterduse headuse hindamisel võtame arvesse ka seda, mitu duplikaati samast motiivist leitakse. Kõrgustega 9,5 ja 10 tekkis mõlema meetodiga väga vähe duplikaate, ainult mõne testandmestiku puhul tekkis kuni 2 sarnast motiivi. Kõige rohkem duplikaate tekkis lõigates kõrguselt 9 (joonis 5.14). Seda seetõttu, et madalamalt lõigates võivad mõned sama motiivi sisaldavad harud olla veel ühendamata. Nii väike duplikaatide arv aga ei ole väga suur probleem ning seda saab püüda järeltöötluse või visuaalse inspekteerimise abil veelgi vähendada.



Joonis 5.14: Duplikaatidega motiivide arv (kõrgus 9).

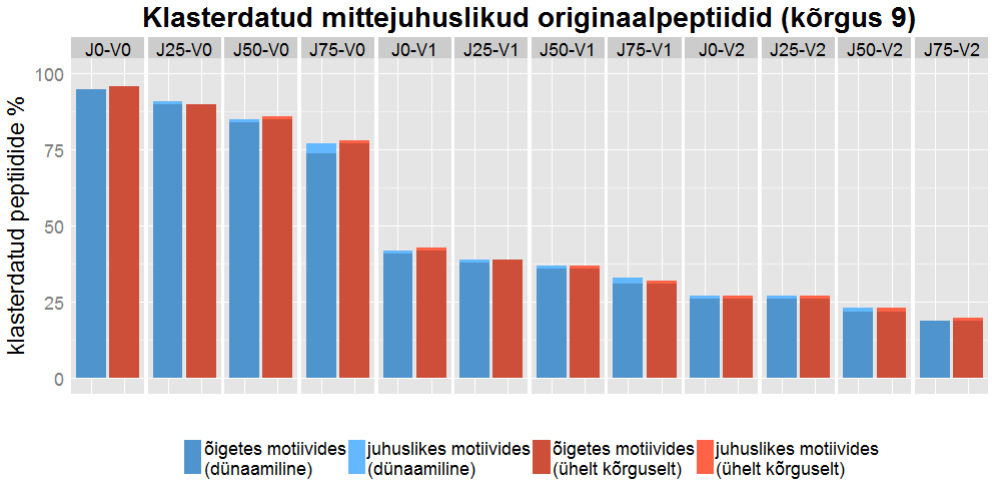
5.2.4 Klasterduse täpsus

Viimaseks hindame, kui hästi on sünteetilistes andmetes olevad motiive sisaldavad peptiidid klasterdunud ning kui täpsed on leitud klastrid. Joonised 5.15, 5.16 ja 5.17 näitavad klasterdatud peptiidide protsenti erinevatelt kõrgustelt lõigates. Heledamalt on märgitud, mitu protsenti on klasterdunud juhuslikesse motiividesse ja tumedamaga, mitu protsenti on klasterdunud õigetesse motiividesse. Näeme, et õigesti klasterdatud peptiidide protsent sõltub palju vigade tasemest. Kõrguselt 9,5 lõigates on vigadeta andmete puhul klasterduse protsent üle 80%. Vaadates tasemete 1 ja 2 klasterduse protsente, näeme, et need sarnanevad tabelites 3.5 ja 3.6 toodud vigadeta peptiidide genereerimise tõenäosusega.



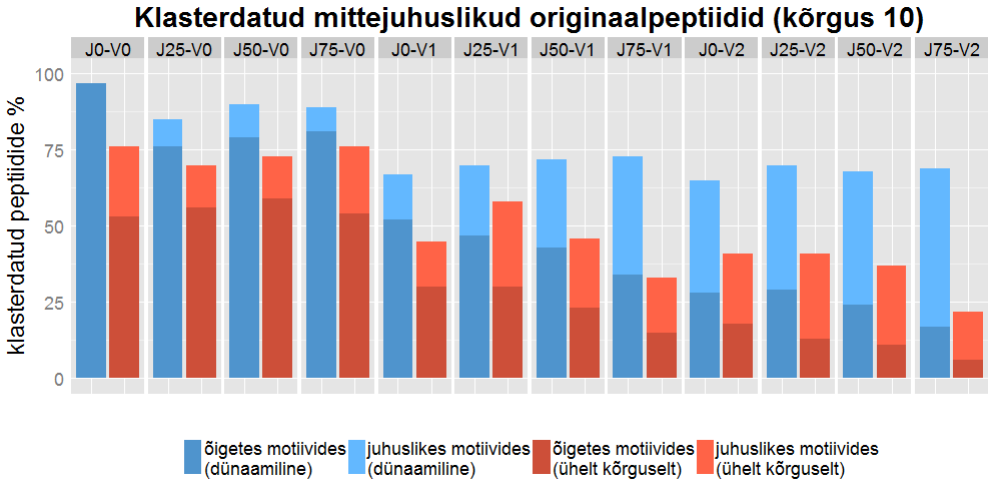
Joonis 5.15: Klasterdatud peptiidide protsent (kõrgus 9,5).

Madalamalt lõigates näeme, et klasterduse protsent on sarnane, kuid veidikene väiksem.



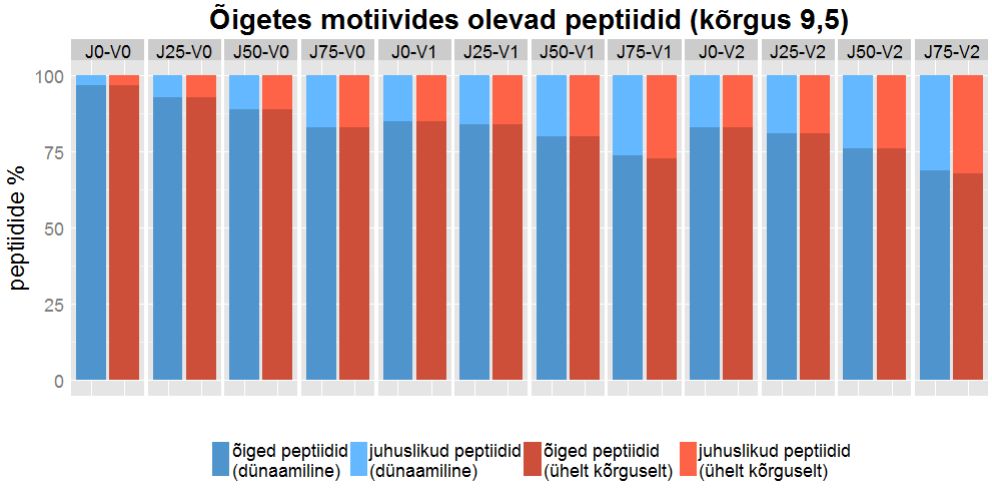
Joonis 5.16: Klasterdatud peptiidide protsent (kõrgus 9).

Kõrgemalt lõigates on klasterduse protsent suurema vigadega andmete puhul küll suurem, kuid kasv tuleb selle arvelt, et rohkem motiive sisaldavaid peptiide satub juhuslikesse motiividesse. Kui juhuslikud motiivid eemaldada, on protsendid jällegi sarnased eelnevatele kõrgustele. Ühelt kõrguselt lõigates klasterdub vähem peptiide, sest antud kõrgus selle lõikamismeetodiga ei sobi.

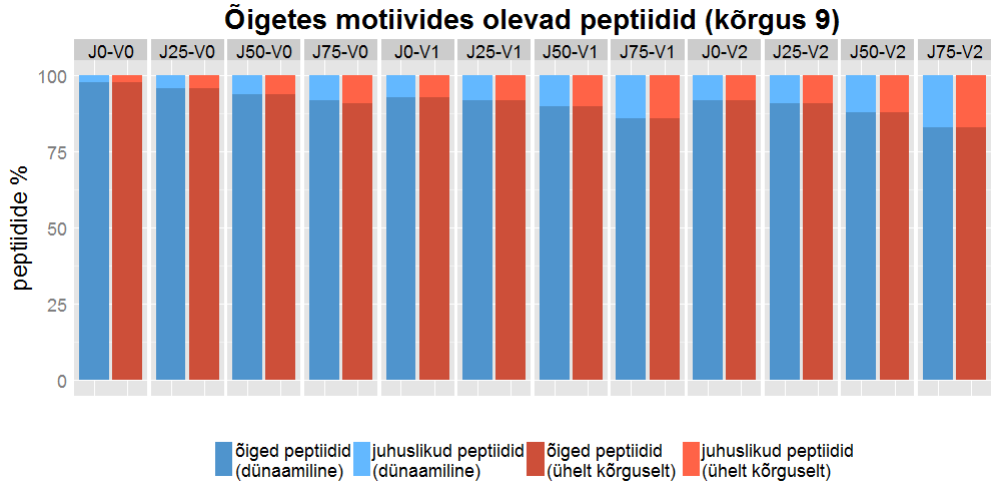


Joonis 5.17: Klasterdatud peptiidide protsent (kõrgus 10).

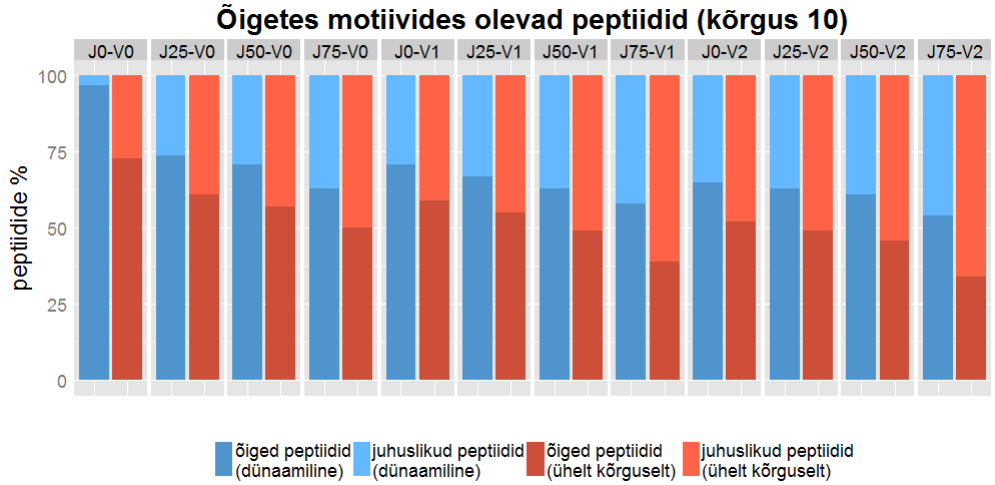
Joonised 5.18, 5.19 ja 5.20 näitavad, mitu protsenti leitud õigetes motiivides olevatest peptiididest on need, mis ka vastavasse originaalmotiivi kuulusid ning mitu protsenti on nendes motiivides juhuslikke või valesse motiivi sattunud peptiide. Nagu näeme, sõltub sobivate peptiidide protsent lõikamise kõrgusest, sest mida kõrgemalt lõigata, seda rohkem satub klastritesse juhuslikke või klastri motiiviga vähem sarnaseid peptiide.



Joonis 5.18: Õigetes motiivides olevate õigete ja juhuslike peptiidide protsent (kõrgus 9.5).



Joonis 5.19: Õigetes motiivides olevate õigete ja juhuslike peptiidide protsent (kõrgus 9).



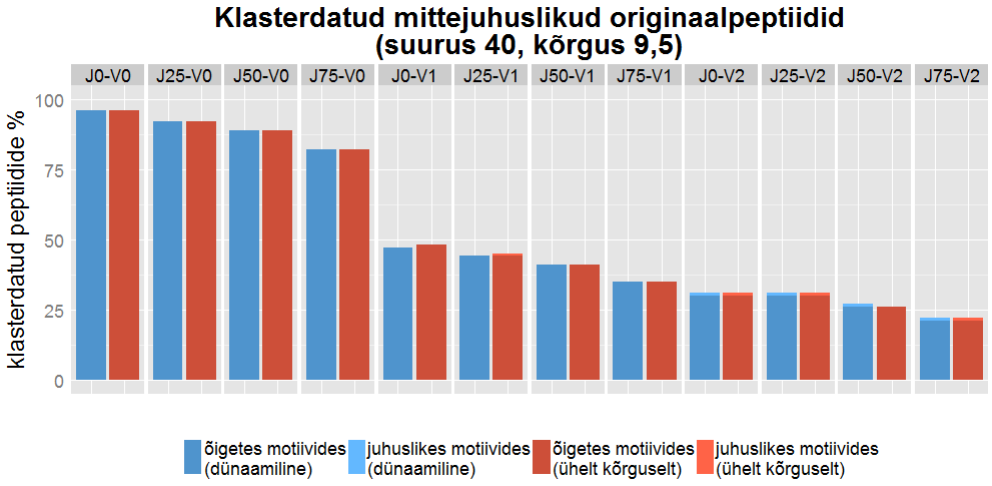
Joonis 5.20: Õigetes motiivides olevate õigete ja juhuslike peptiidide protsent (kõrgus 10).

5.3 Järeltöötlus

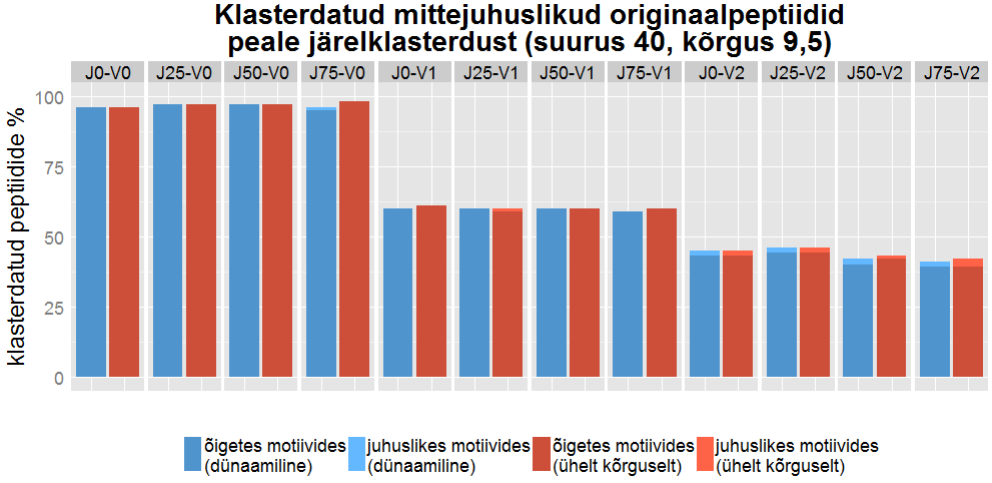
Testisime järeltöötlust motiivide peal, mis on leitud kõrguse parameetriga 9,5 ning suuruse parameetriga 40. Suuruse parameeter on valitud põhjusel, et järelklasterdus ei pruugi anda häid tulemusi, kui andmetesse jätta juhuslikud motiivid. Seega on parem enne järelklasterdust välja valida olulised (suurimad) motiivid. Valisime katsetamiseks lõikamise kõrguse 9,5, sest sellelt kõrguselt saime

kõige paremad motiivid. Niimoodi näitame, kui palju on heade motiivide puhul võimalik klasterdust parandada. Järelklasterdusel lugesime peptiidi motiivile sobivaks, kui peptiid sisaldas motiivi maksimaalselt ühe veaga.

Joonised 5.21 ja 5.22 kirjeldavad klasterdatud peptiidide protsenti enne ja pärast järelklasterdust. Peale järelklasterdust on motiividesse klasterdunud rohkem peptiide. Klasterdatud peptiidide protsent ei parane väga palju, kuid on kooskõlas andmetesse sisestatud vigadeta ja ühe veaga peptiidide osakaaluga.



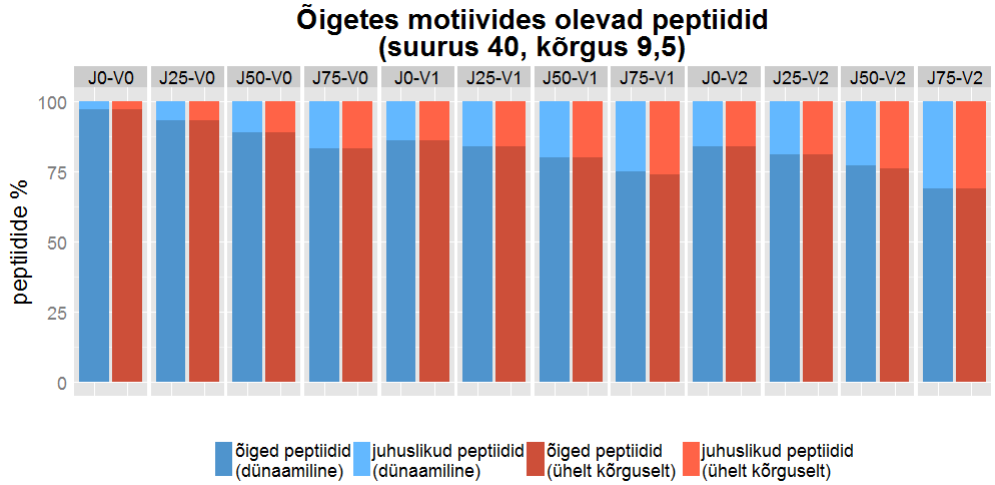
Joonis 5.21: Klasterdatud peptiidide protsent (suurus 40, kõrgus 9,5).



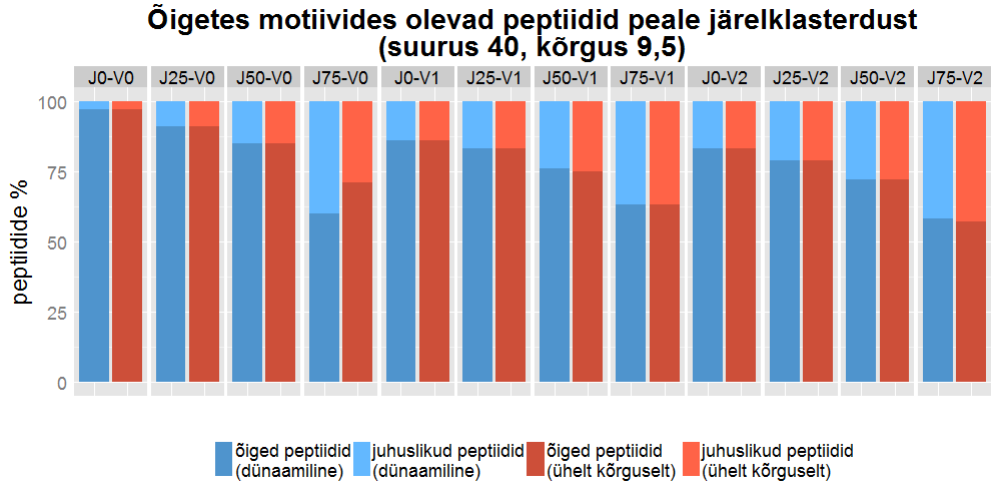
Joonis 5.22: Klasterdatud peptiidide protsent peale järelklasterdust (suurus 40, kõrgus 9,5).

Joonised 5.23 ja 5.24 kirjeldavad leitud õigetes motiivides olevate õigete ja

juhuslike peptiidide osakaalu. Järelklasterduse tagajärjel kasvab õigetes peptiidides olev juhuslike peptiidide osakaal. Eriti suur kasv toimub suurema juhuslike peptiidide protsentidega andmestike puhul, mis on oodatav, sest motiividele tekib rohkem sarnaseid juhuslikke peptiide.



Joonis 5.23: Õigetes motiivides olevate õigete ja juhuslike peptiidide protsent (suurus 40, kõrgus 9,5).



Joonis 5.24: Õigetes motiivides olevate õigete ja juhuslike peptiidide protsent peale järelklasterdust (suurus 40, kõrgus 9,5).

Järelklasterdust on seega soovitatav teha juhul, kui on oluline, et võimalikult suur osa originaalpeptiididest saaksid klasterdatud ning andmetes ei ole väga palju juhuslikke peptiide.

5.4 Järeldused

Testid sünteetilistel andmetel näitasid, et koostatud meetod suudab sobiva kõrguse parameetriga tuvastada vastavalt müratasemele 50%-100% andmetesse sisestatud motiividest. Vigadeta peptiidide puhul suudetakse tuvastada peaaegu kõik motiivid isegi juhul kui andmetest 75% moodustab müra. Mida rohkem on peptiidides vigu, seda vähem motiive suudetakse tuvastada.

Tuvastatud motiivide järjestus on samuti sobiv, suuremad leitud motiivid vastavad originaalmotiividele ja väiksemad motiivid juhuslikult moodustunud motiividele. Järjestus on peaaegu ideaalne kui peptiidides pole vigu, vigade suurendamisel satub lõpptulemusse veidi rohkem juhuslikke motiive, kuid järjestus on siiski piisavalt hea. Juhuslikud motiivid võib välja filtreerida suuruse järgi, sest mida suuremad motiivid alles jätta, seda vähem on nende hulgas juhuslikke motiive.

Õigesti klasterdatud originaalpeptiidide protsent varieerub andmestikes üsna palju, sest sõltub peptiidides olevatest vigadest. Vigadeta peptiidide puhul klasterdatakse õigesti üle 80% peptiididest ning suurima vigade taseme puhul ligikaudu 25% peptiididest. Valides välja suuremad motiivid ning klasterdamata peptiidid järelklasterdada, saab õigesti klasterdatud peptiidide protsenti tõsta vigadeta peptiidide puhul peaaegu 100% ning suurima vigade taseme puhul jääb õigesti klasterdunud peptiidide protsent veidi alla 50%. Järelklasterduse tegemisel peab aga arvestama, et mida rohkem on andmetes müra, seda rohkem suureneb ka juhuslike peptiidide protsent, mida motiividesse lisatakse.

Kahe lõikamismeetodi vahel ei olnud väga suuri erinevusi kui lõikamise kõrgus oli sobiv, kuid liiga kõrgelt lõigates tuleks eelistada dünaamilist lõikamist, sest ühelt kõrguselt lõikamine ei suuda sellel juhul tuvastada enam õigeid klastreid. Kuna teistelt kõrgustelt lõikamine annab meetodite puhul võrreldavaid tulemusi, tuleks üldjuhul eelistada dünaamilist lõikamist, mis töötab stabiilsemalt erinevate parameetritega.

Üks suurimaid edasiminekuid võrreldes varasema meetodiga [14] on motiivide lugemise täpsuse parandamine. Uus meetod suutis suurima vigade arvuga klastritest tuvastada 90% motiividest, 80% täpselt sellisel kujul nagu need on sisestatud, samas kui vana meetod suutis sellisel puhul tuvastada vaid 50% motiividest. Vigadeta klastrite puhul suudetakse tuvastada kõik motiivid ning peaaegu ideaalse täpsusega.

Pärisandmed arvame olevat sarnased andmestikule J50-V1, kust sobivalt kõrguselt lõigates suutsime tuvastada 43 motiivi 50st dünaamilise lõikamisega ja 42 motiivi ühelt kõrguselt lõikamisega. Motiivide kadu võib põhjendada sellega, et kuna peptiididesse tehakse vigu, ei pruugi väiksemates motiivides olla piisavalt palju sarnaseid peptiide, et moodustada originaalmotiivile sarnane motiiv. Motiivide järjestus oli väga hea, AUC väärtus oli 0.998 dünaamilise lõikamisega ja

0.994 ühelt kõrguselt lõikamisega, ehk peaaegu kõik õiged motiivid olid suuremad kui juhuslikult tekkinud motiivid. Õigesti klasterdatud peptiidide protsent jäi veidi alla 50% ning õigetes motiivides oli veidi üle 75% õigeid peptiide. Näide J50-V1 andmestikult leitud suurimatest motiividest on toodud joonisel 5.1.

Kokkuvõte

Antud töö raames valmis hierarhilisel klasterdamisel põhinev meetod lühikestest peptiididest sageli esinevate motiivide leidmiseks. Uue meetodi koostamine oli vajalik, kuna uuritud olemasolevad meetodid probleemi lahendamiseks ei sobinud. Meetod põhineb varem koostatud töövool [14], mille puuduseid püüti selles töös parandada. Meetodi kaks põhilist osa olid klastrite eraldamine hierarhilise klasterdamise käigus tekkinud dendrogrammist ning leitud klastritest motiivide lugemine. Klastrite eraldamiseks proovisime kahte erinevat meetodit: ühelt kõrguselt lõikamist ning dünaamilist lõikamist. Meetodi töötamise aeg oli väiksemate andmete puhul (ligi 10000 peptiidi) paarikümne minuti ringis ning suuremate andmete puhul (ligi 50000 peptiidi) paari tunni ringis. Meetodi tööaega saab edaspidi ka vähendada näiteks teatud ülesannete paralleliseerimise teel, kuid antud töös me optimeerimisele ei keskendunud.

Koostatud meetodi testimiseks genereerisime erinevate omadustega sünteetilisi andmeid. Klasterdusmeetodite võrdlemisel selgus, et kui andmete puhul on võimalik välja arvestada dendrogrammi lõikamiseks sobiv parameeter, annavad mõlemad meetodid sarnaseid tulemusi. Kui aga lõikamise parameetrit ei ole võimalik väga lihtsalt leida, peaks eelistama dünaamilist lõikamist, sest see meetod töötab erinevate parameetrite puhul stabiilsemalt. Sobiva kõrguse puhul suutsid mõlemad meetodid tuvastada andmetesse sisestatud motiive olenevalt andmetes olevale vigade ja müra tasemele 50% - 100% ulatuses. Eeldatavalt päris andmetele kõige sarnasemast sünteetilistest andmestikust, kus vigade ja müra tase on keskmine, suudeti tuvastada 86% motiividest.

Võrreldes varem kasutusel olnud meetodiga [14] oli kõige suurem paranemine sarnaste peptiidide klastritest motiivide tuvastamise meetodi puhul. Vigadega andmete puhul suutis uus meetod korrektselt lugeda 90% motiividest ning vana meetod 50% motiividest. Uue meetodi motiivide tuvastamise täpsus oli 90% - 100%.

Kokkuvõttes suutsime muuta uue meetodi võrreldes varasema kasutusel olnud meetodiga [14] täpsemaks, üldisemaks ja dünaamilisemaks ning vähendada vajalike parameetrite arvu. Edasiarendusena võiks uurida võimalusi juhuslikult tekkinud motiivide automaatseks eraldamiseks õigetest motiividest. Samuti võiks

meetodile koostada kasutajaliidese, et teha selle kasutamine ja tulemuste kuvamine bioloogidele võimalikult mugavaks. Kasutajaliidesesse võib lisada ka erinevaid manuaalseid järeltöötluse võimalusi, näiteks visuaalse inspekteerimise teel ühendada sarnased klastrid.

Eeldades, et pärisandmed on sarnased siin töös genereeritud sünteetiliste andmetega, peaks koostatud meetod suutma ühe inimese MVA analüüsi käigus tekkinud peptiididest tuvastada seal olevad motiivid. Need motiivid võivad edasise uurimise käigus olla abiks inimese haigusloo kirjeldamisel ning haiguste diagnoosimisel. Kuna motiiviotsingut kasutatakse ka teiste bioinformaatiliste probleemide lahendamiseks, võib edasise tööna koostatud meetodit testida teist tüüpi bioloogiliste järjestuste ja teiste omadustega andmete peal.

Kirjandus

[1] K. Palm, L. Kasak, A. Kivil, A. K. Lend, T. Neuman, A. Pihlak, A. Alman, M.-L. Kruup, M. Kull, B. Rajashekar, S. Reisberg, M. Sauk, G. Viikmaa, J. Vilo, "Peptiidide profileerimine ja humoraalse immuunsuse monitooring," Patent US 14/079,626, 11 13, 2013.

[2] G. P. Smith, V. A. Petrenko, "Phage display," *Chemical reviews*, vol. 97, no. 2, pp. 391–410, 1997.

[3] G. E. Crooks, G. Hon, J.-M. Chandonia, S. E. Brenner, "WebLogo: a sequence logo generator," *Genome research*, vol. 14, no. 6, pp. 1188–1190, 2004.

[4] T. L. Bailey, C. Elkan *et al.*, "Fitting a mixture model by expectation maximization to discover motifs in biopolymers," in *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, 1994, pp. 28–36.

[5] T. L. Bailey, M. Gribskov, "Combining evidence using p-values: application to sequence homology searches." *Bioinformatics*, vol. 14, no. 1, pp. 48–54, 1998.

[6] "MEME: Multiple Em for Motif Elicitation," May 2015. [Internetis]. Saadaval: http://meme.nbcr.net/meme/doc/meme.html

[7] D. Quang, X. Xie, "EXTREME: an online EM algorithm for motif discovery," *Bioinformatics*, vol. 30, no. 12, pp. 1667–1673, 2014.

[8] T. Kim, M. S. Tyndel, H. Huang, S. S. Sidhu, G. D. Bader, D. Gfeller, P. M. Kim, "MUSI: an integrated system for identifying multiple specificity from very large peptide or nucleic acid data sets," *Nucleic acids research*, p. 1294, 2011.

[9] W. A. Thompson, L. A. Newberg, S. Conlan, L. A. McCue, C. E. Lawrence, "The Gibbs centroid sampler," *Nucleic acids research*, vol. 35, no. suppl 2, pp. W232–W237, 2007.

[10] V. Neduva, R. Linding, I. Su-Angrand, A. Stark, F. De Masi, T. J. Gibson, J. Lewis, L. Serrano, R. B. Russell, "Systematic discovery of new recognition peptides mediating protein interaction networks," *PLoS biology*, vol. 3, no. 12, p. e405, 2005.

[11] R. J. Edwards, N. E. Davey, D. C. Shields, "SLiMFinder: a probabilistic method for identifying over-represented, convergently evolved, short linear motifs in proteins," *PloS one*, vol. 2, no. 10, p. e967, 2007.

[12] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, D. J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," *Nucleic acids research*, vol. 25, no. 17, pp. 3389–3402, 1997.

[13] J. Vilo, "Pattern Discovery from Biosequences," doktoritöö, Helsinki Ülikool, 2002.

[14] M.-L. Kruup, "Finding motifs from short peptides," bakalaureusetöö, Tartu Ülikool, 2013.

[15] K. Katoh, K. Misawa, K. Kuma, T. Miyata, "MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform," *Nucleic acids research*, vol. 30, no. 14, pp. 3059–3066, 2002.

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, Mari-Liis Kruup (sünnikuupäev: 14.10.1991),

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose:

Klasterduspõhine motiiviotsing lühikestel peptiididel,
mille juhendajad on Meelis Kull ja Jaak Vilo

1.1 reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2 üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartu, 21.05.2015