

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Charleen Konsa

**Patsientide ravimikasutuse klasterdamine ATC
koodide alusel**

Bakalaureusetöö (9 EAP)

Juhendaja: Markus Haug, MSc

Tartu 2024

Patsientide ravimikasutuse klasterdamine ATC koodide alusel

Lühikokkuvõte:

Tervishoiuandmed annavad võimaluse patsientide ravimitrajektoore uurimiseks. Käesoleva lõputöö eesmärk on luua töövoog, mis klasterdab patsiendid nende ravimikasutuse alusel kasutades ATC koodi. Lisaks arendatakse kasutajaliides, mille abil saab töövoogu interaktiivselt kasutada. Valminud töövoog koosneb 5 osast: filtreerimine, ravimitrajektoore koostamine, ravimitrajektoore võrdlemine, ravimitrajektoore klasterdamine ja klastrite analüüsimine. Töövoo tulemusena on patsiendid jaotatud klastritesse, mille kohta on antud lihtne ülevaade. Tulemusi saab kasutada edasiseks uurimiseks, et leida erinevate ravimitrajektoore tekkepõhjuseid. Kasutajaliides koosneb 4 osast. Selle külgribal kuvatakse põhilised kasutajasisendeid, millega saab mõjutada patsientide valikut, klasterdamist ja analüüsi. Vahelehtedel kuvatakse klasterdamise ning analüüsi tulemusi. Kasutajaliidese tulemusi ning kasutatud parameetreid saab alla laadida RDS- ja CSV-failina.

Võtmesõnad:

R, klasterdamine, aegridade analüüs, ravimikasutus, astma

CERCS:

P170 Arvutiteadus, süsteemid;

P160 Statistika, operatsioonanalüüs, programmeerimine, finants- ja kindlustusmatemaatika.

Clustering patients' drug usage based on ATC codes

Abstract:

Healthcare data provides an opportunity to study patients' drug trajectories. This thesis aims to create a workflow that clusters patients based on their drug use using ATC codes. In addition, a user interface is developed that can be used to interactively run the workflow. The workflow consists of 5 parts: filtering, drug trajectories compilation, drug trajectories comparison, drug trajectories clustering, and cluster analysis. As a result of the workflow, patients are divided into clusters, which are given a simple overview. The results can be used for further research to find the reasons for the different drug trajectories. The user interface consists of 4 parts. Its sidebar displays the main user inputs that can influence patient selection, clustering, and analysis. Tabs display the results of clustering and analysis. The results and the used parameters in the user interface can be downloaded as RDS and CSV files.

Keywords:

R, clustering, time series analysis, drug use, asthma

CERCS:

P170 Computer science, systems;

P160 Statistics, operations research, programming, financial and actuarial mathematics.

Sisukord

Sissejuhatus	5
1. Põhimõisted	6
1.1 ATC koodid	6
1.2 Klasterdamine	7
2. Varasemad katsetused ATC koodidega	9
3. Töövoo arendamiseks kasutatav andmestik	10
4. Implementatsioon	12
4.1 Kasutajaliides	12
4.2 Filtreerimine	14
4.3 Ravimitrajektooride koostamine	14
4.4 Ravimitrajektooride võrdlemine	15
4.4.1 Eukleidiline kaugus	16
4.4.2 Dünaamiline ajadeformatsioon	17
4.5 Ravimitrajektooride klasterdamine	19
4.6 Klastrite analüüs	20
5. Programmi analüüs	21
5.1 Programmi ajakulu	21
5.2 Tulemuste kirjeldamine	23
5.2.1 Klastrite loomine	24
5.2.2 Klastrite kirjeldamine	26
5.3 Edasiarendus	29
Kokkuvõte	31
Viidatud kirjandus	32
Lisad	35
1. Külgriba	35
2. Vaheleht „Trajektooride klastrid”	36
3. Vaheleht „Ravimid klastrites”	37
4. Vaheleht „Abi”	38
5. Kasutajaliidese sisendfaili näidis	39
6. GitHubi repositoorium	40
7. Litsents	41

Sissejuhatus

Patsiendi andmete ja visiitide digitaliseerimine on kaasa toonud tervishoiuandmete pideva proportsionaalse kasvu digitaalsete andmete hulgas. Aastaks 2025 ennustatakse, et 36% kõigist digitaalsetest andmetest on seotud tervishoiuga (RBC Capital Markets, 2020).

Suured andmemahud ja nende kättesaadavus pakuvad laialdasi uurimisvõimalusi, mis parandaks tulevikus patsientide tervise väljavaateid. Kõikide andmete käsitlemine nõuab aga palju spetsialistide tööd ja aega. Seetõttu tuleb leida uusi võimalusi tehtava töö automatiseerimiseks, mis aitaks kiiremini olemasolevatest andmetest informatsiooni leida ning järeldusi teha.

Järjest enam kasutatakse tervishoiuandmetest informatsiooni kogumiseks juhendamata masinõppe meetodeid (Khalid ja Prieto-Alhambra, 2019). Varem on edukalt uuritud patsientide haigustrajektoore kasutades RHK koodi (Giannoula jt, 2018). Samuti on Brandon Loorits oma lõputöös vaadelnud astma patsientide ravimitrajektoore, kus astmaravimite grupid on juhuslikult järjestatud. Tema töös loodi töövoog, kus prooviti klasterdada patsientide ravimitrajektooreid dünaamilise ajadeformatsiooni algoritmiga, mis osutus edukaks (Loorits, 2022). Siinne töö on Looritsa töö mõtteline edasiarendus, kus ravimitrajektoore klasterdamiseks kasutatakse ATC koodidel põhinevat kaugusmõõtu.

Lõputöö eesmärk on luua töövoog, mis klasterdab sarnaste ravimitrajektooredega patsiendid, ja kasutajaliides, mis võimaldab töövoogu mugavalt kasutada. Töövoogu tulemusi on võimalik kasutada edasiseks uurimiseks, et leida erinevate ravimitrajektoorede tekkepõhjuseid.

Käesolev lõputöö koosneb viiest peatükist. Esimeses peatükis selgitatakse põhimõisteid, mida lugeja peab teadma edasise töö mõistmiseks. Teises peatükis kirjeldatakse varasemaid artikleid, milles on kasutatud ravimikoodi masinõppe. Kolmandas peatükis tutvustatakse töövoogu arendamiseks kasutatavat andmestikku. Neljandas peatükis süüvitakse töövoogu ja selle rakendamisse programmis ning kasutajaliideses. Viimases peatükis analüüsitakse programmi tulemusi ja selle ajalist efektiivsust. Lisades on kuvatõmmised loodud kasutajaliidesest, kasutajaliidese sisendfaili näidis ja GitHubi repositooriumi link programmi koodile.

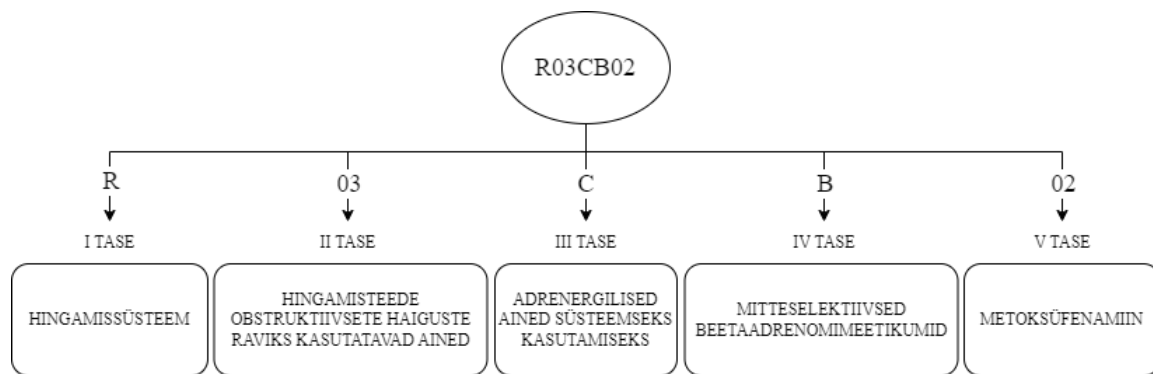
1. Põhimõisted

Selles peatükis antakse ülevaade töös kasutatavatest põhimõistetest nagu ATC koodid ja klasterdamine. Nendest arusaamine on vajalik, et hiljem mõista, kuidas toimub programmis patsientide ravimite ning ravimitrajektoride võrdlemine ja klasterdamine.

1.1 ATC koodid

World Health Organization Collaborating Centre for Drug Statistics Methodology (WHO CC) veebilehe „History“ alamlehel tutvustatakse anatoomilis-terapeutilis-keemilise (ATC, ingl *anatomical therapeutic chemical*) klassifikatsiooni teket. Selle järgi nägi WHO kuuekümnendate lõpus vajadust rahvusvaheliseks ravimite klassifikatsioonisüsteemiks, et edendada ravimite uurimist ja kasutamist. ATC klassifikatsioon töötati välja Norras ja inspiratsiooniks oli selleaegne European Pharmaceutical Market Research Association (EphMRA) klassifikatsioonisüsteem. Samuti tuuakse WHO lehel välja, et nii EphMRA kui ka ATC klassifikatsiooni puhul on kasutusel tasemeline struktuur ja praegugi võib klassifikatsioonide tasemete vahel leida sarnasusi, kuid ATC klassifikatsioon on oma eelkäijast detailsem ja seetõttu laiemalt levinud (WHO CC, 2018).

Ravimite ATC koodide tasemeline struktuur on abiks ravimite vahelise erinevuste arväärtuse arvutamiseks, mistõttu tuleb teada, kuidas need koodid on üles ehitatud. WHO CC alamlehel tuuakse täpselt välja koodide ülesehitus. Selle järgi pannakse ATC kood ainult ravimi toimeainele. ATC koodide struktuur on viietasemeline, kus esimene tase on kõige üldisem ja viimane tase kõige spetsiifilisem. Esimest, kolmandat ja neljandat taset tähistatakse ühe suurtähega. Teist ja viiendat taset märgitakse kahekohalise numbriga. Esimese taseme jaotusi on 14 ja selle määrab elundkond, millele ravim mõjub. Teine tase määratakse ravimi terapeutiliste omaduste järgi, kolmas terapeutilise-farmakoloogiliste omaduste järgi, neljas keemilise, terapeutilise ja farmakoloogiliste omaduste järgi ning viies toimeaine enda järgi. Koos ATC koodiga on võetud kasutusele päevase doosi (DDD, ingl *defined daily dose*) ühik, mis näitab keskmist ravimiannust päevas täiskasvanu kohta (WHO CC, 2022). Joonisel 1 on näha ATC koodi struktuuri toimeaine metoksuüfenamiin näitel.



Joonis 1. Metoksüfenamiini ATC koodi struktuur.

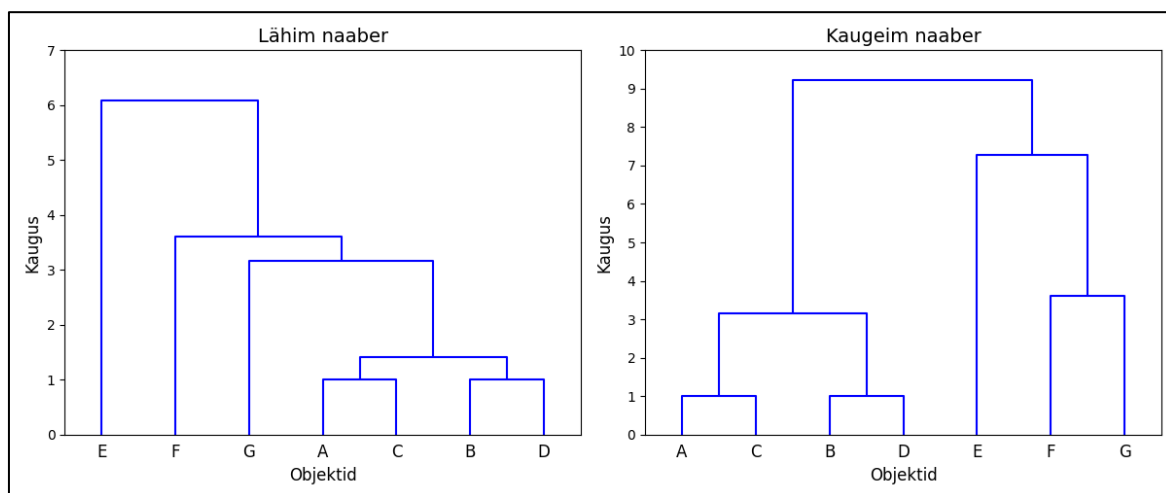
Hetkel tegeleb WHO CC peamiselt uute ravimite klassifitseerimisega ATC/DDD süsteemis, klassifikatsioonisüsteemi ülevaatamisega ja ATC klassifikatsiooni kasutamise edendamisega (WHO CC, 2023). ATC klassifikatsioon on rahvusvaheliselt üks levinumaid, kuid on ka muid süsteeme, mille abil ravimeid klassifitseeritakse. Kõige sarnasemad ATC süsteemile on varem mainitud EphMRA klassifikatsioon ja Põhja-Ameerikas kasutatav American Hospital Formulary Service (AHFS) klassifikatsioon, mille on loonud American Society of Health-System Pharmacist. Kõigi kolme klassifikatsiooni puhul on ravimitele määratud koodid tasemeliselt struktureeritud. Käesolevas töös ei ole koos ATC koodiga DDD käsitletud, sest kogu vajalik informatsioon ravimite vahelise erinevuste arvutamiseks tuleb ATC koodist.

1.2 Klasterdamine

Meelis Kulli õppematerjalide järgi on klasterdamine masinõppe meetod, mille sihiks on sarnaste objektide grupeerimine. Hea klasterdamise tulemusel on ühes klastris objektid, mis on üksteisega sarnased ja teistest väljaspool klastrit asuvatest objektidest erinevad (Kull, 2023). Viise klasterdamiseks on mitmeid, kuid täpsemalt süüvitakse siinses töös hierarhilise klasterdamise meetodisse, sest see on bioinformaatikas üks populaarseimaid meetodeid (Larrañaga jt, 2006).

Hierarhilise klasterdamise algfaasis kujutatakse kõiki objekte eraldi klastrites, igal sammul leitakse kaks kõige väiksema omavahelise kaugusega klastrit ja ühendatakse need kuniks alles on ainult üks suur klaster (Bock, 2018). Klastrite omavahelise kauguse arvutamiseks on erinevaid viise, millest tuntumad on eukleidiline ja linnakaugus (ingl *Manhattan distance*).

Hierarhilise klasterdamise tulemusi kuvatakse üldjuhul graafikul nimetusega dendrogramm. Graafikult on võimalik lugeda, millised elemendid on omavahel klasterdatud ja mis järjekorras on klasterdamine toimunud. Enamasti kuvatakse y-teljel kahe klatri kaugus teineteisest ühendamise hetkel. Näitena on joonisel 2 välja toodud kaks dendrogrammi, kus on seitse objekti hierarhiliselt klasterdatud.



Joonis 2. Dendrogrammid.

Hierarhilise klasterdamise juures on tähtis valida, milliste objektide järgi arvutatakse klasterite kaugus teineteisest olukorras, kus klasterites on rohkem kui üks objekt. See mõjutab, millised klasterid esmalt kokku pannakse ja seetõttu ka klasterdamise tulemust tervikuna. Peamiselt eristatakse kahte meetodit. Lähima naabri meetodi puhul leitakse kaugus kahe klatri lähimate elementide vahel, aga kaugeima naabri meetodis leitakse kaugus klasterite kaugeimate elementide vahel (Tooding, 2020). Joonisel 2 on näha, kuidas ühe või teise meetodi kasutamine võib klasterdamise protsessi muuta.

2. Varasemad katsetused ATC koodidega

ATC koode on varemgi masinõppes kasutatud. Groza jt käsitlevad artiklis uute võimaluste leidmist ravimite taaskasutamiseks klasterdamise abil. See on oluline ravivõimaluste leidmiseks haruldastele või kiiret ravi vajavatele haigustele, sest uute ravimite väljatöötamine on kallis ja pikaldane protsess. Välja tuuakse, kuidas esmalt klasterdatakse ravimid funktsionaalsuste ja omaduste baasil. Edasi minnes võetakse kasutusele ravimite ATC koodid, et leida klastrist ravimid, mille ATC koodi esimene tase erineb klatri enamuse omast. Nii moodi on võimalik leida ravimeid, millel on mõne teise kategooria ravimitega sarnased omadused või funktsionaalsused ja avastada uusi võimalusi juba olemasolevate ravimite kasutamiseks (Groza jt, 2021). Käesolevas töös leitakse võimalusi klasterdamiseks ainult ATC koodide baasil, sest iga ravimi funktsionaalsuste ja omaduste kaasamine oleks arvutuslikult ebaefektiivne.

Chen ja Jiang kirjutasid artikli, kus kasutati ravimite koostist ja ATC koodide hierarhiat, et ennustada vastava ravimi ATC koodi. ATC koodi ennustamine on abiks uute ravimite klassifitseerimisel kui ka võimaluste leidmiseks olemasolevate ravimite taaskasutamiseks. ATC koodide sarnasuse leidmiseks tuuakse välja kaks meetodit. Esimene neist kasutab ATC koodide tasemelist struktuuri ja teise puhul võrreldakse ATC koodide hindamisskoore. ATC koodide tasemelise struktuuri alusel arvutatakse sarnasus tõenäosusliku mudeli abil. Hindamisskoori arvutamiseks käsitlevad Chen ja Jiang ATC koode koostisosade hulgana. Kahe ATC koodi sarnasus leitakse vastavate ravimite koostishulkade sarnasuse arvutamisega. Kokkuvõtlikult toovad nad oma tulemustes välja, et ainult ühe meetodi kasutamine sarnasuse leidmiseks ei pruugi olla piisav heade tulemuste saavutamiseks ATC koodide ülesehituse keerulisuse tõttu (Chen ja Jiang, 2015).

Välja toodud artiklites on kasutatud ATC koode masinõppes ja neist saab järeldada, kuidas on varem ATC koodide kasutamine toiminud. Esimesest artiklist saab lugeda, et ravimeid on võimalik edukalt klasterdada kasutades nende funktsionaalsusi ja omadusi, kuid antud töös oleks nende kaasamine arvutuslikult kulukas. Teisest artiklist tuleb välja, et ATC koodide keerulisuse tõttu võib sarnasuse arvutamine olla raskendatud. Seda tuleb käesolevas töös arvestada, et leida efektiivne lahendus ATC koodide vahelise sarnasuse arvutamiseks.

3. Töövoo arendamiseks kasutatav andmestik

Andmestik koosneb Eesti astma patsientide terviseandmetest ajavahemikus 2011-2019, mis pärinevad Tervisekassa väljastatud raviarvetelt ja digireseptidelt. Andmed on väljastatud Tartu Ülikooli inimuuringute eetikakomitee loaga 299/T-21. Igale patsiendile on määratud ID, mille abil on võimalik tuvastada patsiendi ravimite ajalugu. Patsientide andmed on pseüdonümiseeritud ning nende sünniaeg on antud kuu täpsusega.

Andmestik valiti programmi arendamiseks järgnevate omaduste alusel: terviklikkus, kättesaadavus ja suurus. See tähendab, et andmestikus pole puuduvaid andmeid, sest kasutatakse registriandmeid, sellele on võimalik ligi pääseda ja sellel on piisavalt andmeid, et teha üldistusi ning leida sarnaseid ravimitrajektoore.

Selsamal andmestikul leitud tulemusi kasutasid Markus jt oma artiklis, mis tõi välja ravimite mustrid äsja diagnoositud astma ja kroonilise obstruktiivse kopsuhaiguse¹ (KOK) patsientidel (Markus jt, 2024). Selles artiklis raporteeritud ravimitrajektoorid on antud töös olulised, sest neid saab kasutada loodud töövoo valideerimiseks.

Suur osa andmestikus olemasolevast informatsioonist pole antud töös vajalik, sest loodava töövoo kasutamiseks on vaja ainult ravimitega seotud ATC koodi. Andmestikust on välja võetud alamandmestik, mis koosneb kolmest veerust: patsiendi ID, määratud ravimi ATC kood ja ravimi määramise alguskuupäev. Vaatluse alla võetud patsiendid moodustavad kohordi, kuhu on kaasatud esmase astma diagnoosi saanud inimesed. Selle kontrolliks on rakendatud astma diagnoosile üheaastast tagasivaate perioodi. Sama kohordi definitsiooni kasutasid ka Markus jt (Markus jt, 2024). Iga patsiendi puhul kasutatakse tema kogu kohorti kuulumise perioodi ravimiandmeid. Raviväliste ravimite vältimiseks on vaatluse alla võetud ravimid, kus ravimi ATC kood algab klassifikaatoriga „R03“, sest see viitab hingamisteedega seotud haiguste ravimitele. Ridu on kasutatavas andmestikus 49226 ja selles on 6273 patsiendi raviandmed. Erinevaid ravimikoode on andmestikus 34 tükki.

Andmed asuvad andmemudelil The Observational Medical Outcomes Partnership (OMOP) Common Data Model (CDM), mille on loonud organisatsioon Observational Health Data

¹ Progresseeruv kopsuhaigus, mille korral süveneb õhupuudustunne tingituna bronhide ahenemisest ja kopsukoe kahjustusest (Ivanova jt, 2021).

Sciences and Informatics (OHDSI). OHDSI kodulehe järgi on see 2014. aastal loodud organisatsioon, mille eesmärk on parandada inimeste tervist. Selleks kogutakse terviseandmeid, mis aitaks teha tuleviku terviseküsimustes paremaid otsuseid. Terviseandmete hoiustamine erineb riikide kui ka riigisiseste süsteemide vahel oluliselt ehk andmete jagamiseks ja ühteks kasutamiseks oli vaja luua ühtne standard. Hetkel on OHDSI-l üle 2000 koostööpartneri 74-st riigist. OMOP CDM on standard, mille abil hoiustatakse terviseandmeid, et neid oleks võimalik igapäev analüüsida ja kasutada. OMOP CDM koosneb 37-st tabelist ja 394-st atribuudist ning terves andmestikus kasutatakse standardiseeritud sõnavara (OHDSI, 2024). Käesolevas töös loodav töövoog on mõeldud kasutamiseks OHDSI kommuuni vabavalalise tööriistana.

4. Implementatsioon

Programmi loomiseks on kasutatud programmeerimiskeelte R versiooni 4.0.3 (R Core Team, 2020) ja C++ versiooni C++17 (ISO/IEC, 2017). C++ on võetud kasutusele, et kiirendada suuremahuliste tsüklite jooksumist ja selle võrra vähendada programmi ajakulu. Mõlemad programmeerimiskeeled on omavahel ühendatud R-paketi *Rcpp* versiooniga 1.0.6 (Eddelbuettel ja Balamuta, 2018), mis võimaldab lihtsat integratsiooni R-i ja C++-i funktsioonide vahel. Kasutajaliidese loomiseks kasutatakse R-paketi *shiny* versiooni 1.8.0 (Chang jt, 2024). Pakett *shiny* võimaldab hõlpsasti luua reaktiivseid veebirakendusi.

4.1 Kasutajaliides

Loodud kasutajaliides koosneb külgribast ja kolmest vahelehest: „Trajektoorie klastrid“, „Ravimid klastrites“ ning „Abi“. Järgnevates lõikudes antakse lühiülevaade iga kasutajaliidese osa ülesehitusest.

Külgribal asuvad peamised kasutajasisendid, millega on võimalik mõjutada patsientide valikut, klasterdamist ja klastrite analüüsi. Ülaservas on võimalik lisada oma andmefail, mille peal töövoogu jooksutatakse. Selle puudumisel kasutatakse töötulemuste näitamiseks andmete näidisfaili. Külgriba allservas kuvatakse küünarnuki graafikut (ingl *elbow plot*), mis annab kasutajale aimu, milline klastrite arv oleks klasterdamisel optimaalne. Külgriba välimust saab näha lisas 1.

Vahelehel „Trajektoorie klastrid“ kuvatakse graafikut, kuidas on kõigi patsientide ravimitrajektooriid klasterdatud. Graafikult on lisaks võimalik näha umbkaudselt trajektoorie vahelisi kauguseid vastavalt kõrvalasuvale legendile. Graafiku kohal asuva allalaadimisnupu abil on võimalik alla laadida viimase klasterdamise tulemused. Antud vahelehe väljanägemist näidatakse lisas 2.

Vahelehel „Ravimid klastrites“ kuvatakse iga klatri kohta graafikut selle patsientide kirjetest leiduvate andmestiku kõige sagedasemate ravimite koguseid. Graafikust allpool asub tabel klastrite lisainfoga. Igas veerus näidatakse klatri numbrit, selles leiduvate patsientide arvu, kõige sagedasemat ravimit klatri patsientide seas ja keskmist patsientide ravimitrajektoori pikkust. Vahelehe ülesehitust on võimalik vaadata lisast 3.

Vahelehel „Abi“ selgitatakse kasutajale kasutajaliidese ülesehitust ja erinevate kasutajasisendite tähendust. Vahelehte saab näha lisas 4.

Kasutajaliidese on võimalik lisada oma andmestik, milles olevaid patsiente soovitakse klasterdada. Programmi sees ei tehta andmestikule enne kasutamist eeltötlust ehk kasutaja antud andmestik peab olema juba sobival kujul, mis on välja toodud tabelis 1. Lisaks on lisas 5 välja toodud sisendfaili näidis. Andmestik ei pea olema sorteeritud, sest sellega tegeletakse programmisiseselt.

Tabel 1. Andmestiku veerud.

Nimi	Tüüp	Sisu
ID	Täisarv	Kood patsiendi identifitseerimiseks
DRUG_SOURCE_VALUE	Sõne	Määratud ravimi ATC kood
DRUG_EXPOSURE_START_DATE	Kuupäev	Ravimi määramise kuupäev

Kasutajaliidese on võimalik saadud tulemusi arvutisse salvestada. Kui soovitakse salvestada ainult viimasena kasutatud parameetrid lihtsasti jagataval ja loetaval kujul, siis on seda võimalik teha külgribal asuva allalaadimisnupuga. Sel juhul laetakse alla CSV-fail, milles on kirjas sisendparameetrite väärtused.

Kui soovitakse salvestada informatsiooni parameetrite, graafikute ja klastrite kohta, siis on võimalik seda teha esimesel vahelehel graafiku kohal asuva allalaadimisnupuga. Sel juhul laetakse alla RDS-fail, kus on olemas kasutatud parameetrid, esimesel ja teisel vahelehel asuvate graafikute binaarkoodid ja patsientide ID-d klastrite kaupa. Graafikud on muudetud binaarkoodiks, et lihtsustada nende jagamist R-objektina. RDS-failiga on võimalik hõlpsasti jagada R-objekte ja kasutada tulemusi teistes R-i programmides.

Töövoog on jaotatud osadeks, kus iga järgnev osa toetub eelneva tulemustele. Iga osa vajalikkusesse ja toimimisse süüvitakse järgmistes peatükkides. Programmi töövoog jaguneb viieks osaks:

- filtreerimine;
- ravimitrajektooride koostamine;

- ravimitrajektooride võrdlemine;
- ravimitrajektooride klasterdamine;
- klastrite analüüs.

4.2 Filtreerimine

Patsientide filtreerimist on võimalik teostada patsiendi kirjete koguse järgi. Saab valida minimaalse ja maksimaalse kirjete arvu, mis peab iga patsiendi kohta andmestikus olema. Vahel on mõistlik välja filtreerida näiteks patsiendid, kes on alles oma raviteekonna algusjärgus ja kelle kohta on andmestikus vähe kirjeid, sest sel juhul pole veel piisavalt andmeid, et teha ravimitrajektoori põhjal häid järeldusi. Samuti võib olla mõistlik välja filtreerida näiteks patsiendid, kellel on äärmuslikult palju kirjeid, sest niimoodi võivad ebaharilikud ravimitrajektoorid mõjutada klasterdamise tulemust. Filtreerides saab kasutaja valida klasterdamise protsessi enda jaoks kõige huvipakkuvamad ja sobilikumad patsiendid.

Programmis on patsientide filtreerimine rakendatud R-paketi *dplyr* versiooniga 1.0.4 (Wickham jt, 2021). Pakett *dplyr* võimaldab filtreerida, sorteerida ja muudel viisidel käsitleda andmestikke. Sellega leitakse patsientide ID-de esinemiste arv ja jäetakse alles ainult ID-d, mille kogus jääb soovitud vahemikku.

Kasutajaliideses on lisatud patsiendi kirjete vahemiku määramiseks külgribale liugur (ingl *slider*). Liuguri miinimumväärtus on 1 ja maksimumväärtus on määratud etteantud andmestiku kõige suurema kirjete arvuga patsiendi järgi. Niimoodi on kasutajal võimalik soovi korral võtta valimisse kõik patsiendid, valides terve liuguri piirkonna. Vaikeväärtusena valitakse kirjete vahemik, millega on vähemalt 50% patsientidest kaasatud.

4.3 Ravimitrajektooride koostamine

Järgmisena koostatakse igale sobivale patsiendile tema ravimitrajektoor. Vaadeldes ühe patsiendi ravimeid, siis pole tarvidust alles hoida järjest väljakirjutatud samasuguseid ravimeid, sest sel juhul on ravimit pikendatud ning ravim toimib. Antud juhul tunneme huvi eelkõige, kuidas on patsiendi ravimid aja jooksul muutunud ehk juhud, kus ravim on eelnevast erinev.

Ravimitrajektoori koostamiseks leitakse kõik patsiendi kirjed ja järjestatakse need väljakirjutamise aja järgi alates kõige varasemast. Edaspidi võime käsitleda loodud trajektoore ka

aegridadena, sest nendes olevad andmed on kronoloogiliselt järjestatud. Kirjetest jäetakse alles ainult ravimite ATC koodid, millest eemaldatakse järjestikku korduvad koodid. Edasise võrdlemise puhul on tähtis, et kõik ravimitrajektorid oleksid võrdse pikkusega, sest muul juhul ei ole võimalik rakendada eukleidilise kauguse algoritmi (vt ptk 4.4.1). Kui patsiendi ravimitrajektor on liiga pikk, siis eemaldatakse trajektoori lõpust ebavajalikud ravimikoodid. Kui patsiendi ravimitrajektor on liiga lühike, siis lisatakse trajektoori lõppu iga puuduoleva ravimi märkimiseks -1.

Kasutajaliideses on võimalik kasutajal ravimitrajektoori pikkust valida külgribal asuva liuguriga. Minimaalne trajektoori pikkus on 1, sel juhul vaadatakse ainult iga patsiendi esimest ravimit. Maksimalne trajektoori pikkus on 50% maksimaalsest patsientide kirjade arvust. Vaikiväärtusena vaadatakse trajektoore, mille pikkus on võrdne patsientide kirjade arvu keskmisega.

4.4 Ravimitrajektoride võrdlemine

Kahe ravimitrajektoori võrdlemisel on eesmärk saada numbriline väärtus, mis iseloomustaks trajektoore sarnasust. Saadud väärtust võib nimetada kauguseks. Mida väiksem on kaugus, seda sarnasemad on trajektorid omavahel. Ravimitrajektoride võrdlemisel on võimalik valida kahe algoritmi vahel: eukleidiline kaugus ja dünaamiline ajadeformatsioon (vt ptk 4.4.1 ja 4.4.2). Ravimitrajektoride võrdlemise tulemusena luuakse maatriks, kus on iga patsiendi trajektoori kaugus enda ja kõigi teiste patsientide trajektoridest.

Mõlema algoritmi puhul võrreldakse omavahel trajektorides esinevaid ATC koode. Ravimikoodide võrdlemisel kasutatakse ära ATC koodide tasemelist ehitust. Igale tasemele on määratud maksumus, mis on arvuline väärtus, mida rakendatakse ainult juhul kui kahe koodi sama taseme vahel on erinevus. Kahe ATC koodi vaheline kaugus on rakendatud maksumuste summa. Iga taseme kaupa maksumus väheneb, sest mida varem on koodide vahel erinevus, seda erinevamad need on. Näiteks kui koodid erinevad juba esimesel tasemel, siis ei mõju need samale elundkonnale ehk tegu on kahe väga erineva ravimiga. Samas erinevus ainult viimasel tasemel tähendaks, et kahel ravimil on samasugused terapeutilised-farmakoloogilised ja keemilised omadused. Näide kahe ATC koodi omavahelisest võrdlemisest ja kauguse arvutamisest on joonisel 3.

Maksumused	R	03	A	K	10	
I 14						
II 11	0	+	0	+	7	+
III 7					4	+
IV 4						0
V 2						= 11
	R	03	C	C	10	

Joonis 3. Kahe ATC koodi võrdlemine.

Kasutajaliidese külgribal on kasutajal võimalik valida ravimitrajektooride võrdlemisel kasutatav algoritm kui ka ATC koodide maksumused tasemete kaupa. Vaikimisi kasutatakse eukleidilise kauguse algoritmi. Dünaamilise ajadeformatsiooni algoritmi valimiseks saab kasutada sellenimelist lülitit (ingl *toggle switch*). ATC koodide maksumuste muutmiseks saab kasutada viite numbrilist sisestusvälja (ingl *entry field*), kus iga väli määrab ühele tasemele käiva maksumuse. Iga maksumus peab olema mittenegatiivne arv. Kõik sisestatud negatiivsed arvud asendatakse 0-ga. Vaikeväärtusena on ATC koodide maksumused vastavalt 14, 11, 7, 4 ja 2.

4.4.1 Eukleidiline kaugus

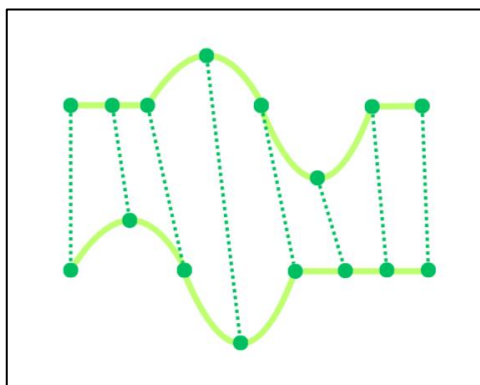
Eukleidilise kauguse algoritm on oma olemuselt kõige intuitiivsem. Algoritmi kasutamisel võrreldakse omavahel aegrea osi, mis on toimunud kronoloogiliselt samal ajal. Kahe ravimitrajektoori võrdlemisel tähendab see, et omavahel võrreldakse esimesi ravimeid, teisi ravimeid ja nii kuni trajektoori lõpuni. Joonisel 4 kuvatakse, kuidas võrreldakse kahe aegrea elemente eukleidilise kauguse algoritmiga. Joonisel kujutatakse heleroheliste joontega aegridu, tumeroheliste täppidega aegrea osi ja tumerohelised katkendlikud jooned näitavad, milliseid aegrea osi on omavahel võrreldud. Kahe trajektoori vaheline kaugus on kõigi võrdluste summa.

Mario Filho kirjutatud artikli järgi on eukleidilise kauguse algoritmi hea kasutada aegridade analüüsis, sest see on kergesti arusaadav ja seda on lihtne rakendada. Filho toob välja, et algoritmi on mõistlik kasutada eelkõige suurte võrdluste arvu puhul, sest võrreldes teiste algoritmidega, ei nõua selle läbimine palju arvutamist ja on selle võrra efektiivsem. See-eest on algoritm palju tundlikum erandite suhtes ega ole võimeline leidma aegridades mustreid, mis ei toimu ajaliselt samal ajal (Filho, 2023).

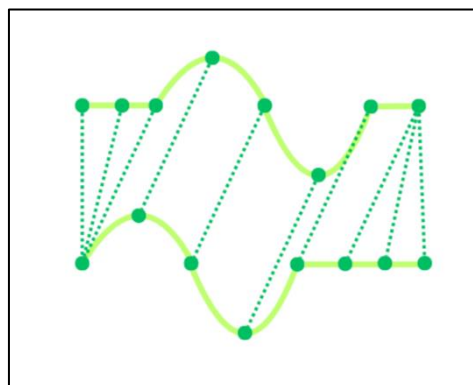
4.4.2 Dünaamiline ajadeformatsioon

Dünaamilise ajadeformatsiooni (DTW, ingl *dynamic time warping*) algoritm on laialt kasutusel aegridade analüüsis. Algoritm võimaldab leida optimaalse ehk kõige lühema kauguse kahe aegrea vahel. DTW algoritmi kasutatakse eelkõige informaatikas ja majanduses. Näiteks on algoritm leidnud kasutust nii kõnetuvastuses kui ka finantsturu analüüsis (Alizadeh, 2022).

Käesolevas töös kasutatakse DTW algoritmi patsientide ravimitrajektooride omavaheliseks võrdlemiseks. Niimoodi leitakse trajektooridest sarnasusi, mis ei toimu ajaliselt samal ajal, kuid ajaliselt samas järjestuses. Joonisel 5 kuvatakse, kuidas kahe aegrea elemente omavahel vastavusse seatakse kasutades DTW algoritmi. Edasi kirjeldatakse algoritmi tööd, mis toetub DTW algoritmi kirjeldavale artiklile (Chauhan, 2022).



Joonis 4. Eukleidilise kauguse algoritm kahel aegreal.



Joonis 5. DTW algoritm kahel aegreal.

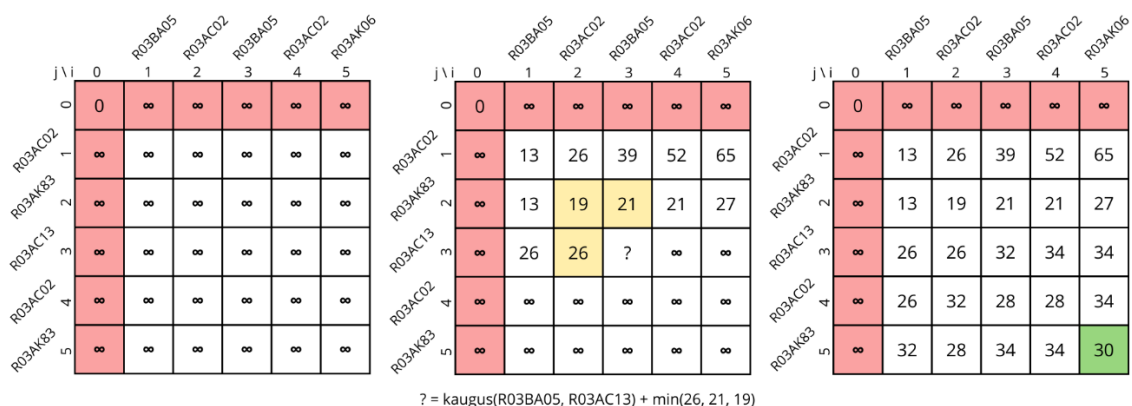
DTW algoritm järgib nelja piirangut:

1. Iga esimese aegrea indeks peab vastama vähemalt ühele teise aegrea indeksile ja vastupidi.
2. Mõlema aegrea esimesed indeksid peavad vastama teineteisele, aga esimese indeksiga vastavusi võib olla rohkem.
3. Mõlema aegrea viimased indeksid peavad vastama teineteisele, aga viimase indeksiga vastavusi võib olla rohkem.

4. Aegride indeksite vastavused peavad olema monotoonselt kasvavas järjestuses. Kui esimese aegria indeksid on i ja j , kus $j > i$, ja teise aegria indeksid on m ja n , kus $n > m$, siis ei saa samaaegselt olla omavahel vastavuses i ja n ning j ja m ega vastupidi.

Kui oletada, et on kaks aegrida, mille pikkused on vastavalt n ja m , siis DTW algoritmi tööd antud aegridadel on võimalik kirjeldada järgmiste sammudena, mida kujutakse ka joonisel 6:

1. Luuakse maatriks kõrgusega $n + 1$ ja laiusega $m + 1$. Maatriksi kõik elemendid väärtustatakse lõpmatusena.
2. Elemendile indeksil $(0, 0)$ antakse väärtus 0, sest aegride võrdlust alustatakse mõlema aegria algusest.
3. Läbitakse maatriksi elemendid (i, j) , kus $i = [1, n + 1]$ ja $j = [1, m + 1]$. Igal sammul võrreldakse omavahel esimese aegria indeksil $i - 1$ asuvat väärtust ja teise aegria indeksil $j - 1$ asuvat väärtust.
 - 3.1. Leitakse mõlema aegria vastavate elementide omavahelist kaugust iseloomustav väärtus.
 - 3.2. Leitakse varasem kõige lühem kaugus. Selleks võetakse minimaalne väärtus indeksitelt $(i - 1, j)$, $(i, j - 1)$ ja $(i - 1, j - 1)$.
 - 3.3. Maatriksi indeksil (i, j) väärtustatakse eelmises kahes punktis leitud väärtuste summaga.
4. Optimaalne kahe aegria vaheline kaugus asub indeksil $(n + 1, m + 1)$.



Joonis 6. DTW algoritmi töö. Vasakul on maatriksi seisund pärast 2. sammu. Keskel on maatriksi seisund 3. sammu poole peal. Paremal on maatriksi lõppseisund.

DTW on hea algoritm aegridade analüüsiks, sest see suudab leida kõige lühema aegridade vahelise kauguse. Läbi käiakse kõikvõimalikud teekonnad, mistõttu leitakse üles sarnased mustrid, mis muidu oleks jäänud avastamata. Algoritmi negatiivseks küljeks on selle suur ajaline keerukus ja arvutuste kogus. Seetõttu on algoritm aeglasem ehk seda pole soovitatav kasutada suurtel andmestikel.

4.5 Ravimitrajektooride klasterdamine

Antud töös sarnaste ravimitrajektooride leidmiseks klasterdatakse need hierarhiliselt. Ravimitrajektooride klasterdamiseks kasutatakse eelnevalt loodud maatriksit, kus on patsientide trajektooride kaugused üksteisest. Klasterdamiseks kasutatakse R-paketi *flashClust* versiooni 1.01-2 (Langfelder ja Horvath, 2012), mis implementeerib optimaalse hierarhilise klasterdamise. Klasterdamine tehakse lähima naabri meetodiga.

Klasterdamisel loodavate klastrite arvu saab muuta külgribal asuva liuguri abil. Minimaalne väärtus on 1 ehk luuakse klaster, milles on kõik patsiendid. Käesolevas töös on määratud klastrite arvu maksimaalseks väärtuseks 10. Luues üle 10 klatri, muutuvad klastrid liiga väikseks ning kasutajal on neid keerulisem hoomata. Vaikeväärtusena luuakse 4 klastrit.

Optimaalse klastrite arvu valimiseks on kasutajale loodud küünarnuki graafik. Küünarnuki graafik sõltub valitud patsientidest ja nende trajektooride võrdlusest. Graafikul kuvatakse klastrite keskmise standardhälbe sõltuvust klastrite arvust. Eesmärgiks on saada standardhälve võimalikult väikseks, sest see näitab, et klattris olevate patsientide trajektoorid on üksteisele võimalikud sarnased. Graafiku pealt saab kasutaja valida klastrite arvu, mis on tema jaoks kõige sobivam.

Kui kasutaja soovib läbi kasutajaliidese klasterdada suurel hulgal patsientide ravimitrajektoore, siis tehakse töövoog läbi ainult 600 juhuslikult valitud patsiendiga. Soovides klasterdada üle 600 patsiendi, võib kasutajaliides võtta enne tulemuste kuvamist liiga pikalt aega ning graafikute mõistmine on raskendatud. Selle asemel tehakse töövoog läbi juhusliku valimiga suurusega 600 patsienti. Niimoodi antakse kasutajale siiski ülevaade patsientide trajektooridest ja klasterdamisest. Kui kasutaja soovib saada klastreid kõigi soovitud patsientidega, siis tulemusi alla laadides tehakse klasterdamise protsess läbi kõigi patsientide hulgal.

4.6 Klastrite analüüs

Moodustunud klastritest kiire ülevaate andmiseks analüüsitakse igat klastrit. Igas klastris leitakse andmestiku kõige sagedasemate ravimite kogused iga patsiendi kirjetes ja kuvatakse need graafikul. See võimaldab kasutajal näha, millised sagedased ravimid esinevad klastris kõige enam ja kuidas erineb sagedaste ravimite osakaal klastrite vahel. Lisaks luuakse tabel, mis kuvab arusaadaval kujul patsientide arvu klastrites, kõige sagedasemat ravimit igas klastris ja keskmist trajektoori pikkust klastris.

Kasutajaliideses on võimalik klastrite analüüsi näha vahelehel „Ravimid klastrites“. Kahes veerus kuvatakse graafikud iga klatri kohta, mis võimaldab kasutajal saada kiire ülevaate ravimite sagedustest kõigis klastrites. Algselt kuvatakse graafikul ravimite sagedusi normaliseerimata kujul ehk graafikult on täpselt näha, mitu korda on mõni ravim patsiendi kirjetes esinenud. See muudab aga klastrite omavahelise võrdlemise keeruliseks, sest graafikute vahemikud võivad olla väga erinevad. Normaliseerimiseks on graafikute kohal lüüti, mis kasutab graafikute normaliseerimiseks \log_{10} funktsiooni. Graafikute all asub tabel, kust saab lugeda eelnimetatud lisainfot klastrites.

Andmestiku kõige sagedasemate ravimite hulga suurust, mida iga patsiendi kohta klastrites vaadatakse, on võimalik muuta külgreal asuva liuguriga. Minimaalselt on võimalik vaadelda 1 ravimit ja maksimaalselt 20 ravimit. Vaikeväärtusena vaadatakse 5 kõige sagedasemat ravimikoodi andmestikus.

Pärast klastrite analüüsi on töövoog lõppenud ja kasutajaliideses kuvatakse tulemused kasutajale. Programm on leidnud sobivad patsiendid, loonud nende ravimitrajektooreid, leidnud ravimitrajektooreid vahelised kaugused, ravimitrajektooreid klasterdanud ja analüüsinud põgusalt tekkinud klastreid. Edasi saab kasutaja tulemusi vaadata, analüüsida kui ka alla laadida. Kasutaja saab parameetreid vastavalt oma soovidele muuta ja töövoog andmestiku peal uuesti käivitada. Tekkinud klastreid saab OHDSI süsteemis sügavamalt uurida Tartu Ülikooli terviseinformaatika uurimisrühma GitHubi lehel² asuvate pakettidega, näiteks *CohortContrast*, *Cohort2Trajectory*, *TrajectoryMarkovAnalysis* ja *TrajectoryViz*.

² <https://github.com/HealthInformaticsUT>

5. Programmi analüüs

Viimases peatükis analüüsitakse kirjutatud programmi ajakulu sõltuvust muutujatest ja kirjeldatakse andmestikust leitud klastreid kasutades Aniek F. Markus jt artiklis (Markus jt, 2024) defineeritud ravimigruppe. Lõpuks tuuakse välja programmi ning kasutajaliidese võimalikud edasiarendused.

5.1 Programmi ajakulu

Järgnevalt leitakse, kuidas mõjutavad kasutaja valitud muutujad programmi ajakulu. Muutuja ajalise mõju uurimiseks defineeritakse kõik teised muutujad ja samm-sammult suurendatakse vaadeldavat muutujat. Iga uue väärtuse korral mõõdetakse töövoa käitamise kestvust. Niimoodi nähakse, kuidas muutub ajakulu vastavalt muutuja väärtustele.

Lõputöö programmi põhieesmärk on klasterdada patsiendid nende ravimitrajektooride alusel. Klasterdamisele järgnev analüüs ja küünarnuki graafiku koostamine on lisatud selleks, et aidata kasutajal hinnata tekkinud klastreid ja valida sobivaim klastrite arv. Kuna need töövoa osad pole ravimitrajektooride klasterdamise jaoks vajalikud, siis vaadatakse järgnevalt ainult põhieesmärgi saavutamise ajakulu.

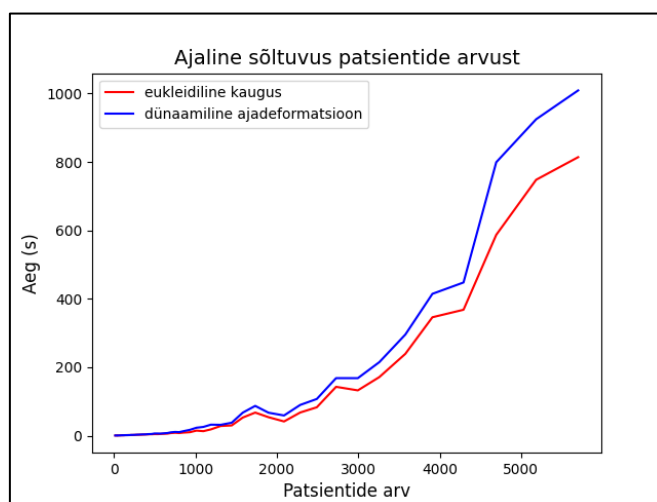
Klasterdamist mõjutavad kuus muutujat:

- andmefail;
- patsiendi kirjete vahemik;
- ravimitrajektoori pikkus;
- klastrite kogus;
- ATC koodide maksumused;
- kasutatav algoritm.

Neist kolme – andmefail, klastrite kogus ja ATC koodide maksumused – edaspidi ei käsitleta, sest need ei mõjuta programmi ajalist keerukust, kuna nende muutujate sisust ei sõltu ükski programmis jooksutatav tsüklil. Seevastu vaadatakse, kuidas mõjuvad ülejäänud kolm muutujat programmi käitusajale.

Patsiendi kirjete vahemiku ehk patsientide arvu ja kasutatava algoritmi mõju programmi ajale vaadatakse koos, sest need on omavahel tugevalt seotud. Patsientide arvust sõltub palju trajektoore tuleb klasterdada ja algoritmi valikust sõltub kaua aega võtab trajektooride võrdlemine. Trajektoori pikkuseks on valitud 5 ja teiste muutujate väärtused on vaikeväärtused. Kirjete vahemiku maksimumväärtus on 200 ja selle miinimumväärtust vähendatakse järjest 1 võrra 150-st 10-ni. Niimoodi läbib töövoogu järjest enam patsiente ning tuleb välja, kuidas mõjub kirjete vahemik programmi ajale. Algoritmi mõju nägemiseks tehakse töövoog läbi nii eukleidilise kauguse kui ka DTW algoritmiga iga kirjete vahemiku väärtusega.

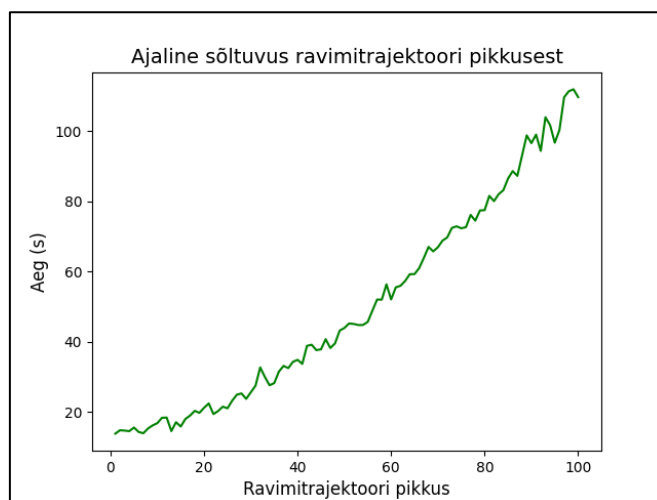
Tulemusi kuvatakse joonisel 7. Kirjete vahemik ehk patsientide arv mõjutab programmi ajakulu olenemata algoritmist sarnaselt. Mõlemal juhul on ajaline keerukus $O(n^2)$. DTW algoritmi kasutades on märgata, kuidas alates 900-st patsiendist on käitusaeg võrreldes eukleidilise kauguse algoritmiga suurem. DTW keerukus on eukleidilise kauguse algoritmi omast suurem ehk oli teada, et selle kasutamine nõuab suuremat ajalist ressursi. Mida suurem on patsientide arv, seda suuremaks muutub algoritmide vaheline ajaline erinevus. Seda tasub programmi käivitamisel arvesse võtta, kui soovitakse klasterdada suurel hulgal patsiente.



Joonis 7. Programmi ajaline sõltuvus patsientide arvust ja valitud algoritmist.

Trajektoori pikkuse mõju vaatamiseks programmi ajale valitakse patsiendi kirjete vahemikuks 30-200 ehk klasterdamine toimub 932 patsiendiga. Ülejäänud muutujad on jäetud vaikeväärtusteks. Trajektoori pikkust suurendatakse järjest 1 võrra 1-st 100-ni.

Tulemusi on kujutatud joonisel 8. Ajakulu kasvab järk-järgult trajektoori pikkuse suurene-
misel ajalise keerukusega $O(n)$. Eeldada võib, et ajaline kasv on seda järsem, mida suurem
on patsientide arv. Sel juhul on ravimitrajektoore rohkem ja trajektoori pikkuse mõju tuleb
rohkem välja. Graafikult on näha, et trajektoori pikkuse mõju pole nii suur kui kirjete vahe-
mikul. Trajektoori pikkuse valimisel tasub leida optimaalne väärtus, mille puhul saadakse
kätte vajalik informatsioon patsiendi kohta hoides samal ajal väärtust piisavalt madalana, et
vähendada programmi ajakulu.



Joonis 8. Programmi ajaline sõltuvus ravimitrajektoori pikkusest.

Uuriti, kuidas mõjutavad kasutajasisendid – patsiendi kirjete vahemik, algoritm ja trajek-
toori pikkus - programmi käitamisaega. Tulemustest sai järeldada, et kirjete vahemik mõju-
tab aega kõige rohkem, sest sellest sõltub klasterdavate patsientide arv. Vaadeldud vahemi-
kus oli näha algoritmide vahelist ajalist erinevust, mis patsientide arvu kasvul suurenes. Tra-
jektoori pikkuse ajaline mõju polnud nii drastiline võrreldes kirjete vahemikuga, kuid siiski
arvestatav. Muutujate ajalise mõju teadmine on kasutajale tähtis, et valida väärtused, mille
puhul ei muutu käitamisaeg liiga suureks.

5.2 Tulemuste kirjeldamine

Tulemuste kirjeldamiseks võetakse eeskuju Aniek F. Markuse jt artiklist, milles uuritakse
astma ja KOK patsientide ravi erinevates riikides (Markus jt, 2024). Selles käsitletakse kuue
riigi digitaalseid terviseandmeid, sealhulgas ka Eesti omi, mida kasutati ka käesolevas töös.
Artiklis uuriti patsientide ravimitrajektoore, kus ravimid on jaotatud kümnesse klassi: ICS

(ingl *inhaled corticosteroids*), SABA (ingl *short-acting β_2 agonists*), LABA (ingl *long-acting β_2 agonists*), SAMA (ingl *short-acting muscarinic antagonists*), LAMA (ingl *long-acting muscarinic antagonists*), LTRA (ingl *leukotriene receptor antagonists*), kantiinid (ingl *xanthines*), suukaudsed süsteemsed glükokortikoidid (ingl *oral systemic glucocorticoids*), PDE4 (ingl *phosphodiesterase-4*) ja biosimilarid (ingl *biologics*). Lisaks toodi eraldi välja neli ravimikombinatsiooni: SABA-LAMA, LABA-LAMA, LABA-ICS ja LABA-LAMA-ICS. Iga käsitletud riigi puhul kuvati diagrammina nende andmetest tulenevad ravimitrajektoorid.

Klastrite põhjalikuks analüüsiks ja kirjeldamiseks kasutati siinses lõputöös juhendaja välja pakutud töövoogu. Patsientide analüüsimiseks kasutati Haugi loodud R-paketti *CohortContrast*³ ja *Cohort2Trajectory*⁴. Ravimitrajektooride jaotuste paremaks kuvamiseks on kasutatud *sunburst* diagramme, millele vajalik eeltöötlus on tehtud Pajusalu R-pakettiga *TrajectoryViz*⁵. Selgema visualiseerimise huvides on *sunburst* diagrammidel kuvatud iga patsiendi esimesed viis ravimit.

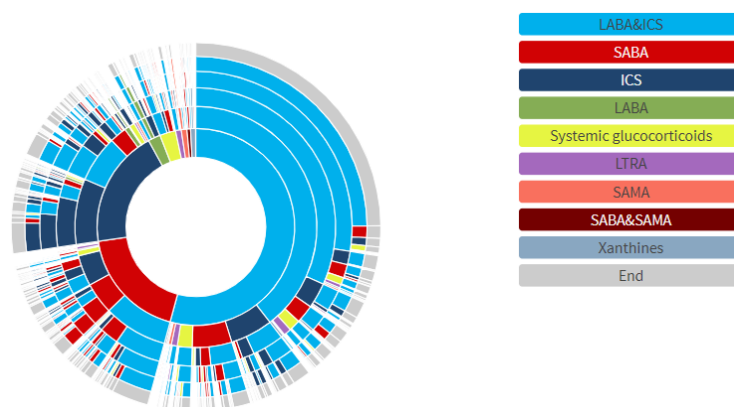
5.2.1 Klastrite loomine

Tulemuste saamiseks klasterdati arendamisandmestiku patsiendid käesolevas töös loodud töövooga. Seejärel grupeeriti patsientide ravimid Markuse jt artikliga (Markus jt, 2024) samasugustesse ravimigruppidesse. Kõigi patsientide grupeeritud ravimite trajektoorid on näha joonisel 9, mille tulemused on sarnased Aniek F. Markuse jt artiklis (Markus jt, 2024) leitud trajektooridele ja nende osakaaludele andmestikus.

³ <https://github.com/HealthInformaticsUT/CohortContrast>

⁴ <https://github.com/HealthInformaticsUT/Cohort2Trajectory>

⁵ <https://github.com/HealthInformaticsUT/TrajectoryViz>



Joonis 9. *Sunburst* diagramm kõigi patsientide trajektoorigega.

Kui klasterdamine on hästi toimunud, siis võiks tekkinud klastreid kirjeldades välja tulla seda klastrit iseloomustav ravimitrajektor või põhjus, miks antud klastris patsiendid teistest erinevad. See aitaks põhjendada, miks on patsiendid sattunud ühte klastrisse ja kuidas on nende ravimitrajektorid üksteisele sarnased.

Klasterdamisel filtreeriti välja patsiendid, kellel oli andmestikus kirjeid vähem kui 10, et eemaldada raviteekonna alguses olevad patsiendid. Ravimitrajektorid loodi pikkusega 20. ATC koodide maksumused olid vastavalt 32, 16, 8, 4 ja 2. Sellega kindlustati, et ravimid, mille erinevus on varasemates tasemetes, eristuvad rohkem ravimitest, mille erinevus on hilisemates tasemetes. Klastrite arv valiti küünarnuki graafiku järgi, kus klastrite keskmine standardhälve oli kõige madalam. Trajektoore võrreldi mõlema algoritmiga. Eukleidilise kauguse ja DTW algoritmiga tekkinud klastrite suurused on välja toodud tabelis 2.

Tabel 2. Eukleidilise kauguse ja DTW algoritmiga tekkinud klastrid patsientide arvuga.

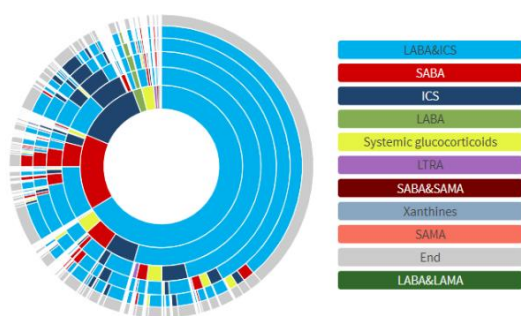
Klaster	1	2	3	4	5	6	7	8
Eukleidiline kaugus	674	581	160	174	113	21	-	-
DTW	955	534	118	56	21	19	14	5

Järgnevalt süüvitakse DTW algoritmiga tekkinud klastritesse, sest võrreldes eukleidilise algoritmiga tuli sellega rohkem välja üksteisest eristuvaid klastreid. Iga tekkinud klastris puhul kirjeldatakse selle erinevust teistest klastritest ja tuuakse välja selle patsientide ravimitrajektorid *sunburst* diagrammiga.

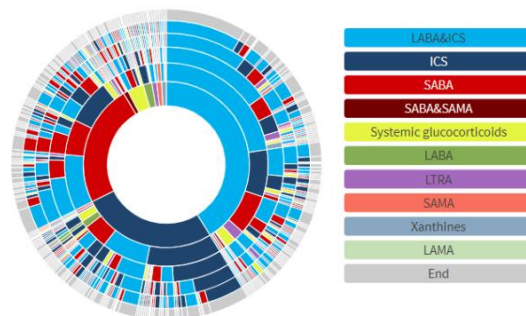
5.2.2 Klastrite kirjeldamine

Järgnevalt kirjeldatakse igat klastrit ja selles olevaid patsiente. Esmalt tuuakse välja, mis ravimid või diagnoosid on klastris ülesindatud, sest see võib anda võimalikke põhjuseid, miks on neid patsiente teistest erinevalt ravitud. Seejärel kuvatakse patsientide ravimitrajektoolid *sunburst* diagrammil. Nende pealt saab vaadata, millised ravimitrajektoolid on enimesindatud. Tuleb tähele panna, et grupeeritud ravimite trajektoories on esindatud ainult ravimid, mis esinesid Markuse jt artiklis (Markus jt, 2024) ehk välja jäänud ravimeid ei ole joonistel kuvatud. Seega võivad erinevatesse klastritesse sattuda näiliselt samasuguste ravimitrajektoorige patsiendid, kellele on tegelikult määratud erinevaid ravimeid, sest klasterdamine on toimunud ravimite ATC koodide alusel.

Klaster 1 patsientide andmetest ei tule välja midagi eripärast. Kuna klaster on teistest kõige suurem ja selle patsiente ei ole võimalik eristada, siis saab järeldada, et klastrisse on sattunud patsiendid, kelle ravimitrajektoor on tervikuna olnud kõige tavapärasem. Jooniselt 10 tuleb välja, et klatri ravimitrajektoolid koosnevad peamiselt LABA-ICS ravimitest, mis on terve andmestiku kõige sagedasem ravimigrupp.



Joonis 10. Klaster 1 ravimitrajektoolid.

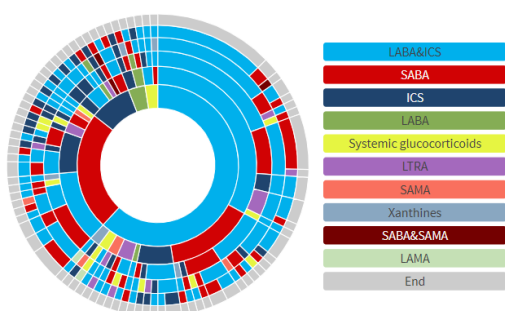


Joonis 11. Klaster 2 ravimitrajektoolid.

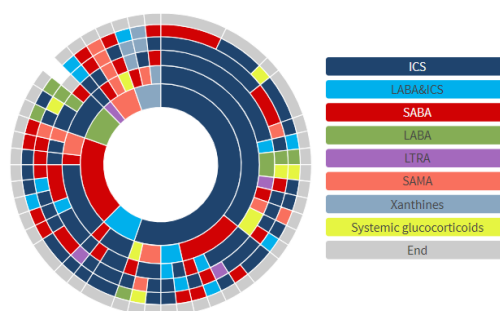
Klastris 2 tulevad esile ravimid, mida kasutatakse peale astma ka allergilise nohu leevendamiseks. Lisaks on klatri patsientidele välja kirjutatud ravimit teofülliin, mida kasutatakse kui teised astmaravimid pole piisavalt tõhusad. Sellest võib järeldada, et klastrisse on sattunud patsiendid, kelle puhul tavapärased astmaravimid ei toimi ehk sobiva ravi leidmiseks katsetatakse erinevate ravimitega. Joonisel 11 on näha, kuidas ravimitrajektoolid on juba pärast teist taset väga erisugused ja puudub kindel eristuv ravimitrajektoor või ravimigrupp.

Klaster 3 patsientidele on sarnaselt klasterile 2 välja kirjutatud teofülliin. Samuti on patsientidele rohkem antud põletikuvastaseid ravimeid. Tundub, et klasteris on patsiendid, kelle

terviseajaloos on rohkem põletikke ja kelle puhul pole tavapärane ravi olnud piisavalt tõhus. Joonisel 12 saab näha, et esimese ja teise ravimi puhul domineerib eelkõige LABA-ICS.



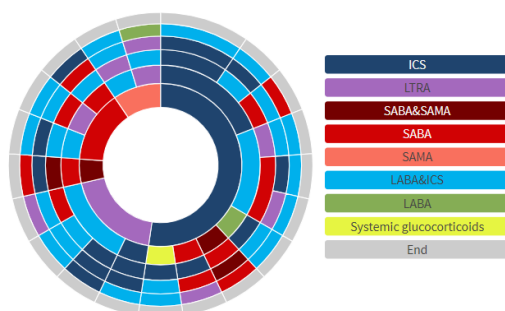
Joonis 12. Klaster 3 ravimitrajektorid.



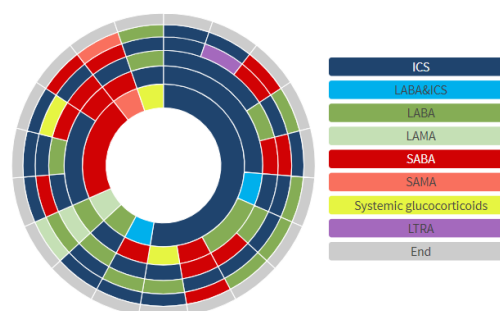
Joonis 13. Klaster 4 ravimitrajektorid.

Klaster 4 patsientidele on rohkem antud astmaravimit salbutamool. Seda eristab selle lühiajaline toime ning eesmärk leevendada astma ägedaid sümptomeid. Lisaks tulevad klastris esile ICS ravimid. Nii on ka näha joonisel 13, et ravimitrajektorides domineerib tavapärase LABA-ICS asemel ICS ravimigrupp. Selle alusel saab eeldada, et patsientidele ravi määramisel on keskendutud eelkõige astma põletikuliste sümptomite kontrollimisele ja vähendamisele.

Klastris 5 patsientide ravimitest tuleb esile astmaravimit montelukast, mis kuulub LTRA gruppi. See takistab leukotrieenide teket vähendades hingamisteede põletikku. Joonisel 14 tuleb välja, et LTRA ravimigrupp on trajektorides rohkem esindatud kui see oli teistes klastrites. Võib eeldada, et LTRA ülesindatus ravimitrajektorides on põhjustanud patsientide sattumise samasse klastrisse.



Joonis 14. Klaster 5 ravimitrajektorid.

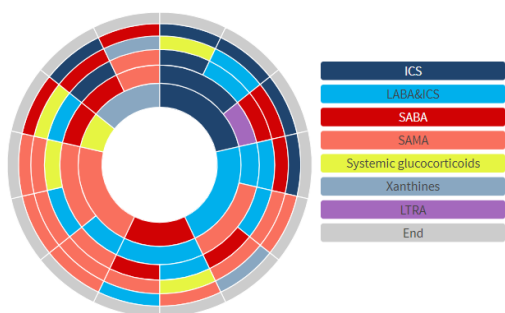


Joonis 15. Klaster 6 ravimitrajektorid.

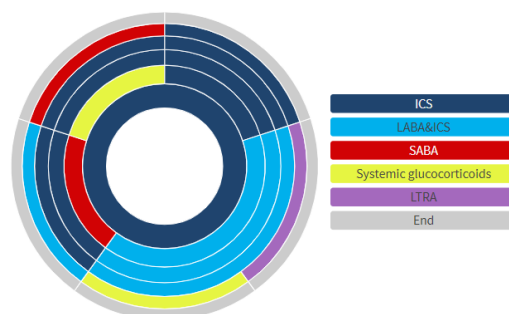
Klastris 6 on ülesindatud LABA ravimid. Samuti on klastris patsientidel rohkem diagnoositud katarakti, silma klaaskeha hägusust ja muid silmahaigusi. See võib tuleneda klastris

patsientide kõrgemast vanusest, sest katarakt on peamiselt vanemas eas esinev haigus, kuid on ka võimalik, et katarakt ja muud silmahaigused on tekkinud astmaravimi kõrvalmõjuna. Joonisel 15 on näha klasteri ravimitrajektoore, kus võrreldes eelnevate klasteritega on rohkem esindatud LABA ravimid.

Klastris 7 ilmneb suuremas koguses astmaravimit teofülliin. Seetõttu võib järeldada, et klasteris olevatele patsientidele pole tavapärased ravimid piisavalt hästi toiminud. Jooniselt 16 on näha klasteri ravimitrajektoorid, kus on ülesindatud SAMA ravimid. Selles võib järeldada, et klaster koosneb patsientidest, kelle ravi koosneb eelkõige SAMA ravimitest.



Joonis 16. Klaster 7 ravimitrajektoorid.



Joonis 17. Klaster 8 ravimitrajektoorid.

Klastris 8 on ainult 5 patsienti, mistõttu on keeruline välja tuua patsientide vahelisi sarnasusi. Peamiselt ilmneb klasteris autoimmuune türeoidiit, mis esineb 3 patsiendil. Selle haiguse tõttu ründab inimese immuunsüsteem enda kilpnääret. Joonisel 17 ei tule patsientide trajektooridest midagi eripärast välja.

Kõiki tekkinud klastreid kirjeldati ja üritati leida, mis eristab neid teistest klasteritest. Iga klasteri puhul ei tulnud hästi välja nende erinevus ja selge klasterdamise põhjendus. Siiski oli klastreid, mille puhul oli näha, kuidas nendes olevad patsiendid omavahel sarnanevad. Klasterite 5, 6 ja 7 puhul olid ravimitrajektoorides tavapärasest rohkem esindatud vastavalt LTRA, LABA, ja SAMA ravimigrupid. Ehk tulevad välja klasterid, kus ravimitrajektoorid on teistsugused teiste klasterite trajektooridest. Klasterisse 1 sattusid patsiendid, kelle ravimitrajektoor oli kõige standardsem. Sellele vastupidiselt sattusid klasteritesse 2 ja 3 patsiendid, kelle ravimitrajektoorid koosnesid väga erinevatest ravimigruppidest. Üldiselt võib pidada klasterdamist edukaks, sest enamikel juhtudel tulid klasteritest välja patsientide ja nende ravimitrajektooride eripärasused, mida oligi tarvis.

5.3 Edasiarendus

Valminud töövoog on kasulik patsientide ravimitrajektooride klasterdamisel ATC koodide alusel, kuid seda on võimalik veel edasi arendada. Käesolevaga pakutakse välja ja selgitatakse erinevaid töövoog ja kasutajaliidese edasiarendamise võimalusi.

Hetkel klasterdatakse patsiendid nende ravimitrajektooride sarnasuse alusel. Pärast seda ei põhjendata ega selgitada klasterdamist ehk ei uurita, miks on patsiendid selliselt klastritesse sattunud. Seetõttu saaks programmi tulemuste analüüsimiseks luua uue töövoog, mis kirjeldaks klastritesse sattunud patsiente ja leiaks sarnaste ravimitrajektooride tekkepõhjuseid. Põhjuste leidmine annaks aimu, mis mõjutab patsiendile määratavaid ravimeid ja mis ravimeid millistele patsientidele välja kirjutatakse. See aitaks kaasa paremate valikute tegemisele tulevaste patsientide ravi määramisel.

Programmis toimub ravimite võrdlemine ATC koodide alusel, kuid seda funktsionaalsust on võimalik laiendada. Hetkel keskendutakse võrdlemisel ATC koodidele, kasutades ära nende tasemelist ehitust. Samas pole ATC klassifikatsioon ainuke kasutusel olev ravimiklassifikatsioon. Peatükis 1.1 on välja toodud ravimiklassifikatsioonid EphMRA ja AHFS, mille ehitus sarnaneb ATC klassifikatsioonile. Seega ei pea programmis piirduma ainult ATC koodidega, vaid see võiks aktsepteerida ka teisi ravimiklassifikatsioone.

Programmis vaadatakse ainult ravimitrajektoore ja klasterdamine toimub nende sarnasuse alusel. Edasi arendades saaks programmiga vaadelda ka muude meditsiiniliste andmete trajektoore. Näiteks saaks uurida patsientide diagnoose ja leida nende trajektooride sarnasusi. Eestis kasutatakse diagnooside klassifikatsioonisüsteemi RHK-10. Seegi, sarnaselt ATC klassifikatsiooniga, on üles ehitatud tasemete kaupa. Kuna RHK-10 ja ATC ehitused on sarnased, siis saaks lõputöös loodud programmiga uurida ka patsientide diagnoose. See võimaldaks programmi kasutada laiemalt kui ainult ravimitrajektooride klasterdamiseks.

Kasutajaliidese on vaadeldavate patsientide arv piiratud, sest suurema patsientide arvu tõttu võtaks programmi jooksumine liiga pikalt aega ning graafikute mõistmine oleks raskendatud. Selle parandamiseks saaks programmis leida võimalusi optimeerimiseks, et vähendada selle ajalist keerukust. Lisaks tuleks leida praegustele graafikutele alternatiive, mis suudaks ka suurel andmehulgal tulemusi mõistlikult kuvada. Niimoodi saab kasutaja kaasata kasutajaliidese klasterdamisesse rohkem patsiente ja tulemusi suudetaks selgemini kuvada.

Kasutajaliideses antakse põgus ülevaade loodud klastritest, et anda kasutajale aimu klastrite sisust ja olemusest. Ülevaade koosneb andmestiku sagedaste ravimite kogusest klastrite patientide kirjetes kui ka informatsioonist klastrite suuruse, kõige sagedasema ravimi ja keskmise trajektoori pikkuse kohta. Soovides pakkuda kasutajale sügavamat arusaamist loodud klastritest, saaks edasiarendusena lisada kasutajaliidesesse muid võimalusi klastrite iseloomustamiseks ja vaatlemiseks. See võimaldaks kasutajal kiiremini ja süvendatult tutvuda klastrite sisuga.

Käesolevaga toodi välja erinevad võimalused, kuidas saaks programmi edasi arendada. Võimalik on analüüsida programmi tulemusi, laiendada programmi funktsionaalsust, kas aktsepteerimaks teisi ravimiklassifikatsioone või täiesti uusi meditsiinilisi andmeid, optimeerida programmi ja lisada kasutajaliidesesse süvendatud klastrite analüüs. Kõik need edasiarendused annaksid tulemustele, programmile ja kasutajaliidesele lisandväärtust ning edendaksid juba loodut.

Kokkuvõte

Iga aastaga tekib järjest enam digitaalseid tervishoiuandmeid (RBC Capital Markets, 2020). Andmete suurehulgaline kogumine pakub uusi võimalusi nende uurimiseks ja avastuste tegemiseks. Seejuures nõuaks nende andmete käsitlemine suurt spetsialistide hulka ja see muudaks andmete käsitlemise aja- ja rahakulukas. Seetõttu tuleb leida lahendusi, mis aita-
vad meditsiinilisi andmeid uurida ja neist järeldusi teha.

Meditšiiniandmete töötlemiseks on juba varem kasutatud erinevaid masinõppe meetodeid. Loorits kasutas oma töös klasterdamist sagedaste ravimitrajektooride leidmiseks, kus trajektoories on astmaravimid grupeeritud ja juhuslikult järjestatud (Loorits, 2022). Kuna töö oli edukas, siis sellest juhinduvalt otsustati siinses töös luua töövoog, mis klasterdab patsientide ravimitrajektoore kasutades ATC koodidel põhinevat kaugusmõõtu.

Lõputöö eesmärgist lähtuvalt kirjutati programm ja selle kasutamist lihtsustav kasutajaliides. Koodi saab vaadata ja alla laadida GitHubi repositoorimis (vt lisa 6). Programm klasterdab etteantud andmestikus olevad patsiendid nende ravimitrajektooride alusel ja analüüsib põgusalt tekkinud klastreid. Selle jaoks läbitakse 5-osaline töövoog: patsientide filtreerimine, ravimitrajektooride koostamine, ravimitrajektooride võrdlemine, ravimitrajektooride klasterdamine ja klastrite analüüsimine. Läbi kasutajaliidese on võimalik töövoogu lihtsasti käivitada kasutades endale sobivaid parameetreid. Kasutajaliides koosneb kolmest vahelehest ja külgribast. Külgriba peal asuvad peamised kasutajasisendid ning vahelehtedel kuvatakse klasterdamise ning analüüsi tulemusi. Saadud tulemusi saab alla laadida RDS ja kasutatud parameetreid CSV-vormingus.

Tööle lisandväärtuse andmiseks on välja toodud edasiarendamise võimalused. Tulemusi saab süvendatult analüüsida, uurida tekkinud klastreid ja leida põhjendusi klastrite tekkeks. See aitaks selgitada, millised ravimitrajektoolid klastrites esile tulevad ja miks on need ravimitrajektoolid tekkinud. Samuti saaks laiendada programmi üldist funktsionaalsust. Seda annaks modifitseerida, et see aktsepteeriks teisi ravimiklassifikatsioone peale ATC ja et sellega saaks klasterdada teisigi meditsiinilisi andmeid, näiteks diagnooside trajektore. Loodud programmi ja kasutajaliidest saaks täiendada, et kasutajatel oleks võimalik klasterdada rohkem patsiente ning saada põhjalikumad ülevaadet tekkinud klastritest.

Viidatud kirjandus

Alizadeh, E. (2022). An Introduction to Dynamic Time Warping. *Built In*. <https://builtin.com/data-science/dynamic-time-warping> (15.05.2024).

Bock, T. (2018). What is Hierarchical Clustering? *Displayr*. <https://www.displayr.com/what-is-hierarchical-clustering/> (15.05.2024).

Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A., Borges, B. (2024). shiny: Web Application Framework for R (R package version 1.8.0.9000). <https://shiny.posit.co/>.

Chauhan, N. S. (2022). Dynamic Time Warping(DTW) Algorithm in Time Series. *The AI Dream*. <https://www.theaidream.com/post/dynamic-time-warping-dtw-algorithm-in-time-series> (15.05.2024).

Chen, F.-S., Jiang, Z.-R. (2015). Prediction of drug's Anatomical Therapeutic Chemical (ATC) code by integrating drug–domain network. *Journal of Biomedical Informatics*, 58, 80-88, doi: 10.1016/j.jbi.2015.09.016

Eddelbuettel, D., Balamuta, J. (2018). Extending R with C++: A Brief Introduction to Rcpp. *The American Statistician*, 72(1), 28-36. doi:10.1080/00031305.2017.1375990.

Filho, M. (2023). How To Measure Time Series Similarity in Python. *Forecastegy*. <https://forecastegy.com/posts/how-to-measure-time-series-similarity-in-python/> (15.05.2024)

Giannoula, A., Bravo, Á., Sanz, F., Furlong, L. I. (2018). Identifying temporal patterns in patient disease trajectories using dynamic time warping: A population-based study. *Scientific Reports*, 8(1), 1-14. doi: 10.1038/s41598-018-22578-1

Groza, V., Udrescu, M., Bozdog, A., Udrescu, L. (2021). Drug Repurposing Using Modularity Clustering in Drug-Drug Similarity Networks Based on Drug–Gene Interactions. *Pharmaceutics*, 13(12), 2117, doi: 10.3390/pharmaceutics13122117

ISO/IEC. (2017). ISO/IEC 14882:2017 - Programming languages - C++. <https://www.iso.org/standard/68564.html>

Ivanova, V., Kliiman, K., Meister, T. (2021). Krooniline obstruktiivne kopsuhaigus ehk KOK. *Ravijuhend*. <https://www.ravijuhend.ee/patsiendivarav/juhendid/168/krooniline-obstruktiivne-kopsuhaigus-ehk-kok> (15.05.2024).

RBC Capital Markets. (2020). The healthcare data explosion. *Healthcare Digitization*, episode 1. https://www.rbccm.com/en/gib/healthcare/episode/the_healthcare_data_explosion (15.05.2024)

Khalid, S., Prieto-Alhambra, D. (2019). Machine Learning for Feature Selection and Cluster Analysis in Drug Utilisation Research. *Current Epidemiology Reports*, 6, 364–372. doi:10.1007/s40471-019-00211-7

Kull, M. (2023). Relations of attributes, clustering and dimensionality reduction. Introduction to Data Science, Lecture 05. https://courses.cs.ut.ee/2023/ids/fall/Main/Lectures?action=download&upname=IDS_lecture_5_part_2_clustering.pdf (15.05.2024).

Langfelder, P., Horvath, S. (2012). Fast R Functions for Robust Correlations and Hierarchical Clustering. *Journal of Statistical Software*, 46(11), 1–17. <http://www.jstatsoft.org/v46/i11/>.

Loorits, B. (2022). „Patsientide enim levinud ravitrajektoorde leidmine DTW meetodil“, TÜ arvutiteaduse instituudi bakalaureusetöö. https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=74494 (15.05.2024)

Markus, A. F., Rijnbeek, P. R., Kors, J. A., Burn, E., Duarte-Salles, T., Haug, M., Kim, C., Kolde, R., Lee, Y., Park, H.-S., Park, R. W., Prieto-Alhambra, D., Reyes, C., Krishnan, J. A., Brusselle, G. G., Verhamme, K. MC. (2024). Real-world treatment trajectories of adults with newly diagnosed asthma or COPD. *BMJ Open Respiratory Research*, 11(1), 2127, doi: 10.1136/bmjresp-2023-002127

Observational Health Data Sciences and Informatics homepage. (2024). <https://www.ohdsi.org/data-standardization/> (15.05.2024).

Larrañaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano J. A., Armañanzas, R., Santafé, G., Pérez, A., Robles, V. (2006) Machine learning in bioinformatics. *Briefings in Bioinformatics*, 7(1), 86–112. doi: 10.1093/bib/bbk007

R Core Team. (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>.

Tooding, L.-M. (2020). Sissejuhatus klasteranalüüsi. M. Beilmann, A. Roots ja K. Rootalu (toim), *Sotsiaalse analüüsi meetodite ja metodoloogia õpibaas*. <https://samm.ut.ee/sissejuhatus-klasteranaluuksi/> (15.05.2024).

Wickham, H., François, R., Henry, L., Müller, K., Vaughan, D. (2021). dplyr: A Grammar of Data Manipulation (R package version 1.0.4). <https://dplyr.tidyverse.org>

World Health Organization Collaborating Centre for Drug Statistics Methodology homepage. (2024). <https://www.whocc.no/> (15.05.2024).

Lisad

1. Külgriba

Vali andmefail

Browse...

No file selected

Kirjete vahemik

0

3

13

100

Trajektoori pikkus

1

10

50

Klastrite arv

1

4

10

Vaadeldavate sagedaste ravimite arv

1

5

20

Kasuta DTW algoritmi

☐

ATC koodide maksumused tasemete kaupa

1

2

3

4

5

14


11

7

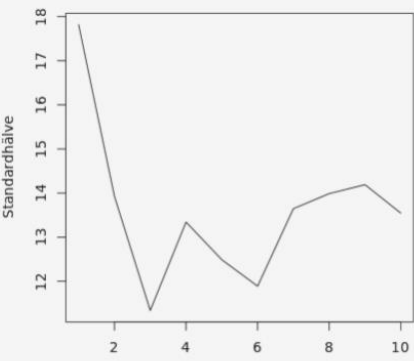
4

2

Salvesta

 Parameetrid

Standardhälve



Klastrite arv

2. Vaheleht „Trajektooriide klastrid“



3. Vaheleht „Ravimid klastrites”



4. Vaheleht „Abi”

[Trajektoore klastrid](#)[Ravimid klastrites](#)[Abi](#)

Vahekaardid

Vahekaart "Trajektoore klastrid". Kuvatakse diagramm, kuidas toimub patsientide klasterdamine nende ravimitrajektoore alusel. Diagrammi värvus annab aimu kui sarnased või erinevad on klasterdatud ravimitrajektoorid. Suurem väärtus tähendab suuremat erinevust kahe trajektoori vahel. Juhul kui klasterdatavaid patsiente on rohkem kui 600, siis valitakse klasterdamiseks juhuslikult 600 patsienti. See on vajalik, et diagrammid oleksid loetavad ja rakendus ei võtaks tulemuste kuvamiseks liiga pikalt aega. Tulemuste salvestamisel nupu "Tulemused" abil klasterdatakse ravimitrajektoorid kõigi soovitud patsientidega. Tulemused salvestatakse RDS-faili. Lisaks klastritele on failis kasutatud parameetrid ja vahelehtedel "Trajektoore klastrid" ning "Ravimid klastrites" olevad diagrammid.

Vahekaart "Ravimid klastrites". Kuvatakse lisainformatsiooni loodud klastrite kohta. Iga klatri puhul luuakse diagramm, millelt saab lugeda, mitu korda on patsientidele välja kirjutatud andmestiku kõige sagedasemaid ravimeid. Diagrammi väärtusi on võimalik normaliseerida diagrammi kohal asuva lüliti "Normaliseeri" abil. Normaliseerimiseks kasutatakse funktsiooni log10. Normaliseerimine võimaldab diagramme omavahel hõlpsamini võrrelda. Diagrammide all kuvatakse iga klatri kohta selles olevate patsientide arv, sagedaseim ravim ja keskmine trajektoori pikkus.

Vahekaart "Abi". Kuvatakse informatsiooni vahekaartide ja külgriba osade kohta.

Külgriba

Andmefail. Võimaldab lisada andmefaili, milles olevaid patsiente soovitakse klasterdada ravimitrajektoore alusel. Vaikeväärtusena kasutatakse näidisfaili, milles pole tegelike patsientide andmeid.

Kirjete vahemik. Võimaldab filtreerida klasterdatavaid patsiente. Kasutajal on võimalik valida minimaalne ja maksimaalne kirjete arv, mis peab igal patsiendil andmestikus olema.

Trajektoori pikkus. Võimaldab reguleerida, mitut ravimit patsiendi kohta vaadeldaks ehk kui pikk on iga patsiendi ravimitrajektoor.

Klastrite arv. Võimaldab reguleerida, mitmeks klastriks patsiendid jaotatakse.

Vaadeldavate sagedaste ravimite arv. Võimaldab reguleerida, mitut andmestikus enimesinenud ravimi kogust iga patsiendi puhul vaadatakse.

DTW algoritm. Võimaldab kasutada ravimitrajektoore võrdlemisel dünaamilise ajadeformatsiooni algoritmi. Vaikeväärtusena kasutatakse eukleidilise kauguse algoritmi.

ATC koodide maksumused. Võimaldab määrata maksumuse igale ATC koodi tasemele. Maksumused peavad olema mittenegatiivsed.

Parameetrid. Võimaldab alla laadida viimasena kasutatud parameetrid CSV-failina.

Diagramm "Keskmise standardhälbe sõltuvus klastrite arvust". Kuvab, kuidas klastrite keskmine standardhälve sõltub klastrite arvust. See võimaldab kasutajal valida optimaalse klastrite arvu.

5. Kasutajaliidese sisendfaili näidis

SUBJECT_ID	DRUG_SOURCE_VALUE	DRUG_EXPOSURE_START_DATE
1	R03AL06	15/07/2016
1	R03AK10	15/05/2017
3	R03AK10	14/08/2017
3	R03AK07	05/02/2018
3	R03BA05	01/10/2018
4	R03DC03	16/06/2017
4	R03AL01	07/07/2017
6	R03AK08	30/07/2019
6	R03AK08	23/09/2019
6	R03AK07	12/12/2019
6	R03BA05	03/01/2020
7	R03BB04	27/07/2018
7	R03AK06	27/11/2018
7	R03AK10	23/04/2019
7	R03DA04	19/06/2019
9	R03BA05	16/11/2018
9	R03AC13	17/11/2018
10	R03AK08	29/08/2019
10	R03AK10	20/09/2019
10	R03AL09	13/10/2019

6. GitHubi repositoorium

Käesolevas lõputöös loodud töövoo ja kasutajaliidese kood asub isiklikus repositooriumis, millele pääseb ligi lingil <https://github.com/Ckonsa/ClusterPatientsTrajectories>.

7. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Charleen Konsa,

annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose **Patsientide ravimikasutuse klasterdamine ATC koodide alusel**, mille juhendaja on **Markus Haug**, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Charleen Konsa

15.05.2024