REPUBLIKA E SHQIPËRISË

UNIVERSITETI POLITEKNIK I TIRANËS

FAKULTETI I TEKNOLOGJISË SË INFORMACIONIT

DEPARTAMENTI I INXHINIERISË INFORMATIKE

BRUNELA KARAMANI

PËR MARRJEN E GRADËS

"DOKTOR"

NË "TEKNOLOGJITË E INFORMACIONIT DHE KOMUNIKIMIT"

DREJTIMI "INXHINIERI INFORMATIKE"

DISERTACION

INFORMATIZIMI I MODELIT PROBABILITAR TË ECURISË DEMOGRAFIKE TË JETËS DHE ANALIZA NËPËRMJET TEKNIKAVE TË DATA MINING NË FUSHËN E SIGURIMIT TË JETËS

Udhëheqësi Shkencor

Akademik NEKI FRASHERI

Tiranë, 2015

INFORMATIZIMI I MODELIT PROBABILITAR TË ECURISË DEMOGRAFIKE TË JETËS DHE ANALIZA NËPËRMJET TEKNIKAVE TË DATA MINING NË FUSHËN E SIGURIMIT TË JETËS

Disertacioni

i paraqitur në Universitetin Politeknik të Tiranës

për marrjen e gradës

"Doktor"

në

"Teknologjitë e Informacionit dhe Komunikimit"

Drejtimi Inxhinieri Informatike

Nga

Znj.Brunela Karamani

2015

Disertacioni i shkruar nga

Znj.Brunela Karamani

FTI, Universiteti Politeknik i Tiranës, 2015

I aprovuar nga

Juria

_____, Kryetari, Juria e doktoratës

_____, Anëtar, Juria e doktoratës

_____, Anëtar, Juria e doktoratës

_____, Anëtar, Juria e doktoratës

_____, Anëtar, Juria e doktoratës

I pranuar nga

_____, Dekan, Fakulteti i Teknologjisë së Informacionit

Miratuar nga

Akademik Jorgaq KAÇANI, Rektori i UPT

Vendimi nr._____datë_____, Këshilli i Profesorëve, FTI

ABSTRAKT

INFORMATIZIMI I MODELIT PROBABILITAR TË ECURISË DEMOGRAFIKE TË JETËS DHE ANALIZA NËPËRMJET TEKNIKAVE TË DATA MINING NË FUSHËN E SIGURIMIT TË JETËS

Industria e sigurimit të jetës është një industri e pasur me të dhëna ku pjesa më e madhe e tyre janë të papërdorshme. Këto të dhëna që posedojnë kompanitë e sigurimit të jetës janë përftuar nga operacionet e kryera çdo ditë dhe suksesi i tyre varet nga shkalla e përdorimit dhe njohjes së ketyre të dhënave. Specifika kryesore e kompanive të sigurimit të jetës është lidhja e pandashme me modelin probabilitar të ecurisë demografike të jetës, ku tendencat demografike janë gjithmonë në zhvillim dhe kërkojnë vrojtime periodike të fenomenit.

Data Mining mund të përcaktohet si proçesi i përzgjedhjes, zbulimit dhe modelimit të sasive të mëdha të të dhënave për të zbuluar modele të panjohura më parë. Në industrinë e sigurimit të jetës, Data Mining mund të ndihmojë kompanitë të fitojnë përparësi në biznes. Me anë të aplikimit të teknikave të saj, kompanitë mund të shfrytëzojnë plotësisht të dhënat në ndërtimin e modeleve për reduktimin e mashtrimit, në menaxhimin e riskut, në përftimin e klientëve të rinj, në ruajtjen e klientëve aktuale dhe në zhvillimin e produkteve të reja. Përdorimi i modelimeve nëpërmjet teknikave të klasifikimit, grupimit, analizës së shoqërimit, rrjetave neurale, algoritmave gjenetike mundësojnë përgjigje të shpejta dhe të sigurta për problemet kryesore në industrinë e sigurimit të jetës.

Në këtë disertacion janë modifikuar dhe eksperimentuar algoritmat e Data Mining, të cilat përdoren sot në fushën e sigurimit të jetës, me qëllim përmirësimin e tyre në mënyrë që të ndihmojnë kompanitë të fitojnë përparësi në biznes.

Algoritmat e përdorur në këtë disertacion janë algoritmi CART, algoritmi i fqinjësisë më të afërt, rrjetat nervore, algoritmi i k-mesatareve, algoritmi k-medianave, algoritmi përparësor dhe algoritmi gjenetik.

CART është një klasifikues të dhënash pemë vendimi vetëshpjegues që trajton atribute me vlerë numerike dhe nominale. Përmirësimi i performancës për algoritmin CART është bërë nëpërmjet vendosjes të një ndarësi zëvendësues për klasën e klientëve, të cilët janë mohuar për të marrë një sigurim jete. Rezultatet eksperimentale tregojnë se modifikimi i propozuar për algoritmin CART është më i mirë në terma të saktësisë 85.9% se modeli i tij standart me 64.6%. Ky algoritëm përdoret nga kompanitë e sigurimit të jetës për të parashikuar mundësinë e paracaktimit të vlerësimit për klientët e rinj.

Algoritmi i fqinjësisë më të afërt është një klasifikues i bazuar në të mësuarit me analogji dhe është efikas për grupe të mëdha trajnimi. Rezultatet eksperimentale tregojnë se modeli i ndërtuar sipas largësisë Euklidiane të ponderuar është më i mirë se modelet e tjera të ndërtuara sipas largësive Euklidiane e thjeshtë, Manhatan e thjeshtë dhe të ponderuar. Saktësia e modelit të ndërtuar sipas largësisë Euklidiane të ponderuar rezultoi 87% dhe faktorët e riskut që ndikojnë në vlerësimin e klientëve me risk në sigurimin e jetës janë mosha, gjinia dhe pesha.
Algoritmi i k-mesatareve përbën një metodë të thjeshtë grupimi të të dhënave, të përftuara sipas një numri k grupimesh të dhënë. Përmirësimi i tij konsiston në gjetjen e një mënyre të re përzgjedhjeje të centroideve fillestare. Rezultatet eksperimentale tregojnë se metoda e propozuar e përzgjedhjes së centroideve fillestare është 45% më e mirë sipas treguesit shuma e katrorëve të gabimit se metoda standarte.
Algoritmi përparësor bazohet në faktin e përdorimit të njohurive të mëparshme të cilësive në gërmimin e bashkësive të shpeshta për të nxjerrë rregulla shoqërimi. Për algoritmin përparësor propozojmë përmirësimin e tij nëpërmjet metodës së rritjes të modeleve të shpeshtë, e cila thjeshton termin e përftuar duke përshtatur një ndarje dhe duke vendosur termat e shpeshtë brënda një strukture. Rezultatet eksperimentale tregojnë se metoda e rritjes se modeleve të shpeshtë është 30% më e shpejtë në kohë proçesimi dhe gjeneron dyfishin e rregullave të shoqërimit se metoda standarte e algoritmit përparësor.

Fjalët kyçe: Data Mining; Sigurim Jete; CART; Algoritëm k-mesatare; Algoritëm Fqinjësia më e Afërt; Rrjetat Neurale; Algoritëm Përparësor; Algoritëm Gjenetik; Saktësia; Përmirësim; Performancë;

FALENDERIME

Realizimi i këtij punimi u bë i mundur pas një pune kërkimore disavjeçare, ku përveç kontributit tim personal nuk mund të lë pa përmendur ndihmën e çmuar të një grupi miqsh e kolegësh.

Në rradhë të parë, një falenderim të veçantë ia kushtoj udhëheqësit tim shkencor, Akademik Neki Frasheri, i cili me mbështetjen, orientimet, sugjerimet dhe profesionalizmin e treguar kontribuoi pa kursim për kurorëzimin me sukses të kësaj pune kërkimore.

Mirënjohje speciale kam gjithashtu për të gjithë ata ish-kolegë e miq në sistemin e sigurimeve që, në mënyrë formale apo informale, më kanë mundësuar baza të dhënash, informacione, vërejtje dhe sugjerime gjatë periudhës kërkimore.

Falenderime më të sinqerta u shkojnë atyre, që me mbështetjen e tyre kanë bërë të mundur realizimin e kësaj sipërmarrjeje.

Në fund do të doja të falenderoja nga zemra vajzën time Sara dhe familjen time për dashurinë, inkurajimin dhe mbështetjen që më kanë krijuar në jetë.

Brunela Karamani

Tiranë, 2015

TABELA E PËRMBAJTJES

LISTA E FIGURAVE

LISTA E TABELAVE

LISTA E GRAFIKEVE

FJALOR TERMINOLLOGJIK

DM - Data Mining; proçesi i analizës së volumeve të mëdha të të dhënave me qëllim gjetjen e informacioneve të reja mbi to, si në baza të dhënash, në data warehouse, në web dhe në depozitime të tjera masive të dhënash.

KDD - Knowledge Discovery in Databases; proçesi për identifikimin tek të dhënat i modeleve me karakteristika vlefshmërie, risie, potencial përdorimi dhe thjeshtësie.

DBMS - Database Management System; një grup programesh që manipulojnë bazat e të dhënave dhe paraqiten si ndërmjetësues mes bazave të të dhënave, përdoruesve dhe programeve aplikative

DWH - Data WareHouse; një depozitë informacioni të mbledhur nga burime të ndryshme, që përdoret për raportime dhe analizë të dhënash.

OLAP - Online Analytical Processing një strukturë të dhënash shumëpërmasore e cila lejon përpunimin paraprak ashtu si dhe akses më të shpejtë të të dhënave

TUPLE – ose rekorde, quhen objektet, kampionet, shembujt, pikat e të dhënave në kontekstin e klasifikimit të dhënave

IBM - International Business Machines

SPSS - Statistical Package for the Social Sciences

WEKA - Waikato Environment for Knowledge Analysis

GUI - Graphical User Interface

KNN - K Nearest Neighbors

NN - Neural Network

ML - Machine Learning

CART - Classification and Regression Trees

ROC - Receiver Operating Characteristic

SEC (Square Error Criteria) – kriter i bazuar në shumën e katrorëve të gabimeve

1 – HYRJE

1.1 PËRDORIMI I DATA MINING NË SIGURIMIN E JETËS

Industria e sigurimit të jetës është një industri e pasur me të dhëna ku pjesa më e madhe e tyre janë të papërdorshme. Këto të dhëna që posedojnë kompanitë e sigurimit të jetës janë përftuar nga operacionet e kryera çdo ditë. Suksesi i kompanive varet nga shkalla e përdorimit dhe njohjes së ketyre të dhënave, të cilat mund të mundësojnë përcaktimin e sjelljes së klientëve dhe preferencave të tyre, në mënyrë që t'ju ofrojnë shërbime më të mira dhe më fitimprurëse.
Specifika kryesore e kompanive të sigurimit të jetës është lidhja e pandashme me modelin probabilitar të ecurisë demografike të jetës, ku tendencat demografike janë gjithmonë në zhvillim dhe kërkojnë vrojtime periodike të fenomenit.
Data Mining mund të përcaktohet si proçesi i përzgjedhjes, zbulimit dhe modelimit të sasive të mëdha të të dhënave për të zbuluar ligjësi të panjohura më parë. Në industrinë e sigurimit të jetës, Data Mining mund të ndihmojë kompanitë e sigurimit të jetës të fitojnë përparësi në biznes. Me anë të aplikimit të teknikave të saj, kompanitë mund të shfrytëzojnë plotësisht të dhënat në ndërtimin e modeleve për reduktimin e mashtrimit, në menaxhimin e riskut, në përftimin e klientëve të rinj, në ruajtjen e klienteve aktuale dhe në zhvillimin e produkteve të reja.
Përdorimi i modelimeve nëpërmjet teknikave të klasifikimit, grupimit, analizës së shoqërimit, rrjetave neurale, algoritmave gjenetike mundësojnë zgjidhje të shpejta dhe më të sigurta për problemet kryesore në industrinë e sigurimit të jetës.

1.2 MOTIVIMI DHE QËLLIMI

Qëllimi i këtij studimi është përmirësimi i algoritmave të Data Mining në fushën e sigurimit të jetës për të përmirësuar cilësinë në menaxhimin e riskut, në përftimin e klientëve të rinj dhe në ruajtjen e klientelës ekzistuese.
Teknikat e përdorura në këtë studim janë klasifikimi nëpërmjet pemëve të vendimit, fqinjësisë më të afërt dhe rrjetave neurale; grupimi nëpërmjet algoritmit të k–mesatareve; analiza e shoqërimit nëpërmjet algoritmit përparësor dhe algoritmat gjenetikë.
Synojmë të përmirësojmë algoritmat duke përdorur bazën e të dhënave të një kompanie sigurimi jete, do të përcaktojmë metrikat e vlerësimit dhe do të krahasojmë dhe analizojmë rezultatet.
Nga kompanitë e sigurimit të jetës kërkohet nxitja e teknikave të data mining dhe për këtë arsye ngrihen pyetjet e mëposhtme:

1. Si i parashikojnë kompanitë e sigurimit të jetës faktorët e riskut që janë të rëndësishëm në përpilimin e çmimoreve?
2. Çfarë teknike përpunimi të të dhënave përdorin kompanitë e sigurimit të jetës në përftimin e klientëve të rinj?
3. Si përcaktohen nga kompanitë e sigurimit të jetës klientët që kanë mundësi të blejnë më shumë se një produkt?
4. Me çfarë teknike përcaktohet nga kompanitë e sigurimit të jetës maksimizimi i vlerës së jetëgjatësisë të klientit?
5. Si ndërtohen modelet për ruajtjen e klientëve ekzistues nga kompanitë e sigurimit të jetës?

Rezultatet e nxjerra nga eksperimentet e kryera i referohen realitetit shqiptar. Duke patur gjithmonë në qendër të studimit pyetjet e ngritura më sipër, kemi ngritur disa hipoteza, të cilat mbështeten në kërkime dhe fakte.

Hipoteza 1: Përmirësimi i teknikave të klasifikimit të data mining çon në vlerësimin dhe përcaktimin më të mirë të ndryshimeve demografike të klientëve në kompanitë e sigurimit të jetës.

Hipoteza 2: Vendosja e një ndarësi zëvëndësues në modelet pemë vendimi të algoritmit CART përmirëson performancën në klasifikimin e të dhënave që përdoren nga kompanitë e sigurimit të jetës.

Hipoteza 3: Zgjedhja e madhësisë së largësisë dhe përafrimi drejt kombinimit të emërtimeve të grupeve në modelet e ndërtuara me anë të algoritmit të fqinjësisë më të afërt përmirëson performancën, e cila ndikon në vlerësimin e klientëve në kompanitë e sigurimit të jetës.

Hipoteza 4: Përcaktimi i përzgjedhjes së centroidit fillestar të grupimit në modelet e ndërtuara me anë të algoritmit të k-mesatareve siguron një rritje të performancës të tij se metoda e rastit, kur përdoret për grupime të dhënash në kompanitë e sigurimit të jetës.

Hipoteza 5: Përmirësimi i algoritmin përparësor me anë të metodës së rritjes së modeleve të shpeshtë që thjeshton termin e përftuar duke përshtatur një ndarje, e cila jep strategjinë për vendosjen e të dhënave që përfaqësojnë terma të shpeshtë brënda një strukture. Duke përdorur teknikën analiza e shoqërimit dhe duke kryer

analiza të njëpasnjëshme në grupe të caktuara klientësh, kompanitë e sigurimit të jetës mund të zgjedhin se cilat shërbime të ofrojnë dhe ndaj cilëve klientë.

Hipoteza 6: Duke përdorur algoritmat gjenetike kompanitë e sigurimit të jetës mund të përmirësojnë dhe optimizojnë rrjetin e shitjes dhe mund të klasifikojnë dokumentat.

1.3 OBJEKTIVAT E STUDIMIT

Në këtë studim janë përmirësuar dhe eksperimentuar metodat e përdorura nga teknikat e Data Mining në sigurimin e jetës, të cilat mund të ndihmojnë kompanitë të fitojnë përparësi në biznes. Në ndërtimin e eksperimenteve janë përshkruar hapat që kryhen per t'ju përgjigjur pyetjeve kërkimore dhe janë vlerësuar rezultatet e gjetura në to. Objektivat kryesore të këtij studimi janë:

- Vlerësimi i situatës aktuale dhe zhvillimet e reja në përmirësimin e teknikave të Data Mining në fushën e sigurimit të jetës, nëpërmjet studimit të literaturës;
- Përmirësimi i performancës së algoritmit CART nëpërmjet vendosjes së një ndarësi zëvendësues;
- Përmirësimi i performancës së algoritmit KNN duke kombinuar dy problematikat zgjedhjen e madhësisë së largësisë dhe përafrimin drejt kombinimit të emërtimeve të grupeve;
- Për algoritmin e k–mesatareve propozojmë një metodë përzgjedhjeje për centroidin fillestar të grupimit në vend të një metode përzgjedhjeje të rastësishme, duke siguruar një vlerësim të performancës të metodës së propozuar mbi shumë bashkësi të dhënash me përmasa të ndryshme;
- Për algoritmin përparësor propozojmë një metodë përmirësimi të quajtur metoda e rritjes se modeleve të shpeshtë që thjeshton termin e përftuar duke përshtatur një ndarje, e cila jep strategjinë për vendosjen e të dhënave që përfaqësojnë terma të shpeshtë brënda një strukture;
- Për algoritmin gjenetik do të hulumtojmë klasifikimin e teksteve në sigurime duke rritur efikasitetin dhe efiçencën e tij.
- Analiza e eksperimenteve për verifikimin e saktësisë të rezultateve të modeleve të propozuara;
- Vlerësimi i performancës për zgjidhjet nga algoritmat e modifikuara duke përdorur të dhënat në fushën e sigurimit të jetës, në raport me performancën e arritur nga zgjidhjet standarte të tyre;
- Vlerësimi i përgjithshëm i rezultateve të arritura;

1.4 METODOLOGJIA E KËRKIMIT

Në metodologjinë e kërkimit kemi zgjedhur mjetet që duhen për analizën e metodave të Data Mining për të gjetur hapësirat për përmirësimit e algoritmave të përdorur në të, duke marrë në konsideratë të gjitha supozimet themelore dhe kriteret. Metodologjia e kërkimit në Data Mining bazohet në kriteret e mëposhtme: (1) Eksplorimi për të gjetur lidhjet e fshehura prapa një fenomeni; (2) Diferencimi i një fenomeni nga të tjerët; (3) Identifikimi i lidhjeve; (4) Gjetja e një zgjidhjeje; dhe (5) Vlerësimi i efektivitetit.

Jemi bazuar gjithashtu edhe në metodologjitë e njohura të kërkimit shkencor abstraksioni dhe projektimi, ku abstraksioni ngrihet mbi metodën eksperimentale shkencore dhe projektimi është metodologjia e cila përdoret në inxhinieri.

Në metodën e abstraksionit një fenomen shikohet dhe vlerësohet duke u mbështetur në rezultatet dhe analizat e eksperimenteve shkencore. Hapat që ndiqen janë ndërtimi i hipotezës, ndërtimi i modelit, ndërtimi i eksperimentit, mbledhja e të dhënave dhe analiza e rezultateve.

Ndersa projektimi ka si qëllim që t'i japë zgjidhje një problemi të caktuar. Hapat për ndërtimin e projektimit janë kërkesat bazë, specifikimet, projektimi, implementimi dhe testimi i sistemit.

1.5 METODIKA E STUDIMIT

Ky studim fillon duke dhënë një vështrim të literaturës së viteve të fundit për të studiuar dhe njohur teknikat e Data Mining që janë në dispozicion për zgjidhjen e problemeve në fushën e sigurimit të jetës. Jemi fokusuar në teknikat më të njohura dhe metodat e gjetura në literaturë. Janë analizuar konceptet e teknikave të klasifikimit, pemët e vendimeve dhe janë marrë në konsideratë përmirësimi i algoritmit CART nëpërmjet vendosjen e një ndarësi zëvendësues, rritja e shkallës së saktësisë së klasifikimit për algoritmin e fqinjësisë më të afërt nëpërmjet zgjedhjes së madhësisë së largësisë dhe të përafrimit drejt kombinimit të emërtimeve të grupeve. Për teknikën e grupimit kemi analizuar algoritmin e k-mesatareve, aplikimi i tij dhe përmirësimi i performancës nëpërmjet përzgjedhjes së centroidit fillestar. Për algoritmin përparësor është marrë në konsideratë metoda e rritjes së modeleve të shpeshtë për të përmirësuar performancën në kohë proçesimi. Nëpërmjet studimit të literaturës janë eksploruar metoda të ndryshme që janë në dispozicion për vlerësimin e teknikave të Data Mining. Bazuar në rezultatet e gjetura të kërkimit kemi modifikuar kritere të përshtatshme për vlerësimin e teknikave të ndryshme që do të përdorim. Në mënyrë që të kemi rezultate cilësore,

kemi bërë një kërkim në literaturë për të identifikuar se cilat metoda japin rezultate cilësore dhe janë të përshtatshme për studimin tonë. Janë hulumtuar software të ndryshme të Data Mining dhe për të kryer eksperimentet e studimit tonë kemi zgjedhur programet IBM SPSS Statistics v.20, Matlab v7.10 dhe Weka 3.7.6 që kanë zbatimin e të gjitha teknikave standarte të Data Mining.

1.6 ORGANIZIMI I STUDIMIT

Ky studim është e organizuar në tetë kapituj, ku secili ka një fokus të caktuar.

Kapitulli 1 – përshkruan në terma të përgjithshëm disa nga elementët kryesorë të dizertacionit. Bazat shkencore mbi të cilat bazohet ky studim, përmenden shkurt në terma të thjeshtuar. Jepet një tablo e përgjithshme e problemit, qëllimi, pyetjet kërkimore, hipotezat, objektivat, metodika dhe metodologjia e përdorur me të cilat lidhet puna konkrete e zhvilluar.
Kapitulli 2 – Në këtë kapitull prezantohen aspektet themelore në lidhje me Data Mining, të cilat do të përdoren në këtë punë kërkimore. Këto aspekte kanë të bëjnë me: çfarë është Data Mining, proçesi i zbulimit të njohurive në bazat e të dhënave, tipet e të dhënave, tipet e modeleve, matjet objektive dhe subjektive, kriteret e vlerësimit, përfshirjen e teknologjive si statistika dhe mekanizmi i të mësuarit në Data Mining.
Kapitulli 3 – Trajton gjerësisht problemin e klasifikimit të të dhënave në sigurimin e jetës, teknikat e saj dhe kriteret për përzgjedhjen e atributeve në ndërtimin e modeleve. Gjithashtu janë trajtuar algoritmi pemë vendimi CART, algoritmi i fqinjësisë më të afërt KNN dhe rrjetat neurale në ndërtimin e modeleve për problemet e klasifikimit në përcaktimin saktë të ndryshimeve demografike të klientëve në sigurime. Përmirësimi i performancës për algoritmin CART është realizuar nëpërmjet vendosjes të një ndarësi zëvendësues në modelet e klasifikimit të ndërtuara. Rritja e shkallës së saktësisë së klasifikimit për algoritmin e fqinjësisë më të afërt të rendit të k-të është zgjidhur duke kombinuar dy problematikat zgjedhjen e madhësisë së largësisë dhe numrit e fqinjëve më të afërt.
Kapitulli 4 - Në këtë kapitull është paraqitur analiza e grupimit, kriteret e saj dhe metodat për krahasim të saj. Janë trajtuar algoritmet e k–mesatareve, k-medianave dhe DBSCAN në ndertimin e modeleve për problemet e grupimit të të dhënave në sigurime. Për algoritmin e k-mesatareve në problemin e sigurimit të jetës kemi propozuar të përdoret metoda e përzgjedhjes së centroidin fillestar të grupimit në vend të metodës së përzgjedhjes së rastësishme dhe efektivitetin e algoritmit të grupimit e kemi vlerësuar nëpërmjet kriterit shuma e gabimeve në katror.

Kapitulli 5 - Në këtë kapitull është dhënë një vështrim mbi analizën e shportës së tregut, analizën e shoqërimit, rregullat e saj dhe kriteret e vlerësimit mbeshtetja e besueshmëria. Kemi hulumtuar algoritmat ekzistues të analizës së shoqërimit duke propozuar një përmirësim të algoritmit përparësor. Përmirësimi i performancës së tij e kemi zgjidhur nëpërmjet metodës së rritjes së modeleve të shpeshtë, duke përshtatur një ndarje, e cila jep strategjinë e vendosjes se të dhënave që përfaqësojnë termat e shpeshtë brënda një strukture.

Kapitulli 6 - Ky kapitull është fokusuar tek algoritmat gjenetikë duke përshkruar fillimisht ndërtimin dhe strukturën e tij, me pas kemi paraqitur parametrat përfaqësimi i individëve, popullata fillestare, funksioni i vlerësimit, përzgjedhja e prindërve dhe operatorët e ndryshimit, me qëllim përmirësimin e efikasitetit llogaritës të tij. Kemi hulumtuar një algoritëm të ri për problemin e klasifikimit të dokumentave në sigurime me anë të algoritmave gjenetike.

Kapitulli 7 - Ky kapitull fokusohet në përdorimin e programeve të DM, të cilat bazohen ne algoritmat standarte për përmirësimin e treguesve të algoritmave CART, KNN, K-mesatareve, algoritmit përparësor dhe algoritmit gjenetik. Janë trajtuar mjetet për ndërtim, analizim dhe paraqitje të të dhënave për të eksperimentuar algoritmat standarte dhe të propozuar.

Kapitulli 8 – Kapitulli i tetë paraqet një vlerësim të gjerë eksperimental të algoritmave të modifikuar. Eksperimentet janë zhvilluar mbi disa baza të dhënash të marra nga një kompani shqiptare në sigurimin e jetës, me anë të të cilave është testuar efikasiteti i algoritmave. Vlerësimi i performancës është bazuar në kritere të mirëpercaktuara. Rezultatet eksperimentale tregojnë se modifikimi i propozuar për algoritmin CART është më i mirë në terma të saktësisë 85.9% se modeli i tij standart me 64.6%. Saktësia e modelit të ndërtuar nga algoritmi KNN sipas largësisë Euklidiane të ponderuar është 87%. Përmirësimi i algoritmit te k-mesatareve konsiston në gjetjen e një mënyre të re përzgjedhjeje të centroideve fillestare, e cila rezultatet 45% më e mirë sipas treguesit shuma e gabimeve ne katror se metoda standarte. Për algoritmin përparësor metoda e rritjes se modeleve të shpeshtë është 30% më e shpejtë në kohë proçesimi dhe gjeneron dyfishin e rregullave të shoqërimit se metoda standarte e tij. Kemi paraqitur një aplikim real i algoritmave gjenetike në permiresimin dhe optimizimin e rrjetit te shitjes në një kompani sigurimi i jetës. Gjithashtu kemi përdorur algoritmat gjenetike në klasifikimin e dokumentave në sigurime duke e trajtuar si një problem optimizimi.

2 – DATA MINING

Në këtë kapitull prezantohen aspektet themelore në lidhje me Data Mining, të cilat do të përdoren në këtë punë kërkimore. Këto aspekte kanë të bëjnë me: çfarë është Data Mining (në vijim DM), proçesi i zbulimit të njohurive në bazat e të dhënave, tipet e të dhënave, tipet e modeleve, matjet objektive dhe subjektive, kriteret e vlerësimit, përfshirjen e teknologjive si statistika dhe mekanizmi i të mësuarit në DM.

2.1 ÇFARË ËSHTË DATA MINING?

Teknologjia harduerike e kompjuterave ka bërë progres të vazhdueshëm dhe të qëndrueshëm në dekadat e fundit dhe kjo ka sjellë si rezultat kompjutera të fuqishëm dhe të përballueshëm nga ana financiare për përdoruesit. Kjo teknollogji i jep një shtytje industrisë së bazave të të dhënave dhe mundëson përgatitjen për menaxhimin e transaksioneve, përftimin e informacionit dhe analizën e të dhënave të një numri të madh baza të dhënash [1].

Rritja e volumit të të dhënave të ruajtura kërkon teknika të reja dhe mjete të automatizuara, të cilët mund të ndihmojnë në mënyrë inteligjente në transformimin e sasive të mëdha të të dhënave në njohuri [2].

Kjo ka sjellë gjenerimin e një fushe në shkencat kompjuterike të quajtur DM si dhe të aplikacioneve të ndryshme të saj. DM është proçesi i ekzaminimit të volumeve të mëdha të të dhënave me qëllim gjetjen e informacioneve të reja mbi to, si në baza të dhënash, në data warehouse, në web dhe në depozitime të tjera masive të dhënash [3].

DM është përdorur gjerësisht nga sistemi bankar për nxitjen e konsumatorëve të kartës së kreditit [4], nga shoqëritë e sigurimit, instuticionet e tjera financiare dhe kompanitë e telekomunikacionit në zbulimin e mashtrimeve [5] dhe nga kompanitë telefonike në identifikimin e klientëve potencial më të mundshëm për t'u ofruar më shumë aplikacione të tjera [6].

2.2 PROÇESI KDD

DM bën pjesë në fushën më të zgjeruar të zbulimit të njohurive në bazat e të dhënave (Knowledge Discovery in Databases) ndryshe njohur si proçesi KDD, pra

aplikimi i një algoritmi të caktuar për të veçuar lidhjet, modelet, sekuencat e përsëritura dhe rregullsitë, të fshehura tek të dhënat.

KDD është proçesi për identifikimin tek të dhënat i modeleve me karakteristika vlefshmërie, risie, përdorimi potencial dhe thjeshtësie në kuptim [7].

Vlefshmëria qëndron në faktin që modeli i zbuluar do të jetë i vlefshëm edhe për të dhënat e reja me të njëjtin nivel qartësie. Modelet janë risi nëse japin informacione të reja që mund të vlerësohen duke vëzhguar ndryshimet e të dhënave. Përdorimi i mundshëm tregon që modelet e zbuluara duhet të çojnë në veprime të dobishme. Objektivi i fundit është që në përkufizimin e modeleve duhet të thjeshtëzohet dhe përmirësohet aftësia për t'u kuptuar e të dhënave.

Ky proçes është iterativ për shkak se fazat e ndryshme mund të përsëriten dhe është interaktiv në momentin që kërkohet pjesemarrja e analistit për shumë vendime dhe për zgjedhjen e disa parametrave që do t'i kalohen algoritmeve [8].
Qëllimi parësor i proçesit të KDD është njohja, e kuptuar si një bashkësi informacionesh që përftohen duke analizuar të dhënat. KDD është një fushë kërkimi që përfshin disiplina që shkojnë përtej Inteligjencës Artificiale, përftimit të njohurive, nga statistika tek vizualizimi i të dhënave etj. Proçesi KDD [9] mund të shihet si një sekuencë iterative e hapave të mëposhtëm në figurën 1 [10]:

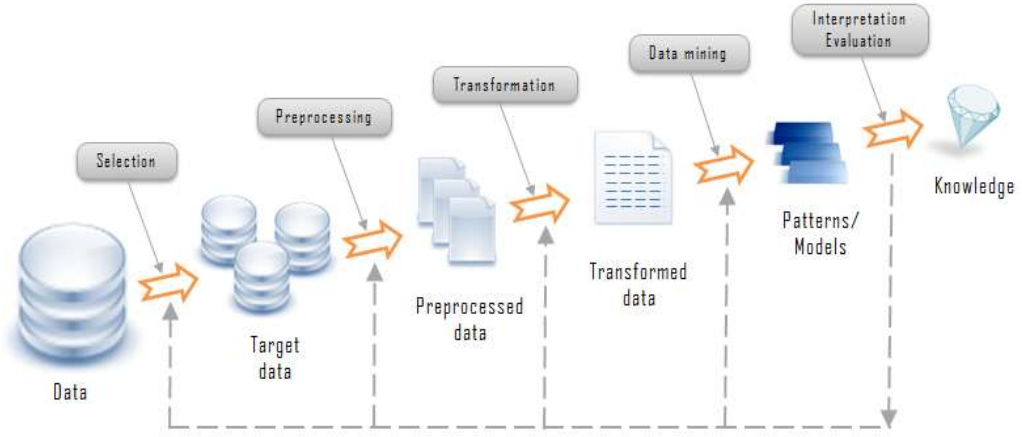


Figura 1 Proçesi KDD

Hapi_1: Pastrimi i të dhënave, i cili është proçesi i mënjanimit të zhurmave dhe të dhënave të paqëndrueshme.

Hapi_2: Integrimi i të dhënave, i cili është proçesi i kombinimit të burimeve të shumëfishta të të dhënave.
Hapi_3: Përzgjedhja e të dhënave, i cili është proçesi i marrjes së të dhënave të dobishme për analizë nga baza e të dhënave.
Hapi_4: Shndërrimi i të dhënave, i cili është proçesi ku të dhënat transformohen dhe konsolidohen në forma të përshtatshme për gërmim duke kryer veprimet e përmbledhjes dhe agregimit.
Hapi_5: Ekzaminimi dhe vlerësimi i modelit.
Hapi_6: Prezantimi i njohurive, i cili është proçesi ku vizualizimi dhe teknikat e prezantimit përdoren për t'i paraqitur njohuritë e përftuara përdoruesve.

2.2.1 PROÇESI I DM NË SIGURIME

Proçesi i Data Mining kalon nëpër disa faza [11], të cilat paraqiten si më poshtë:
- *Hyrja e të dhënave* e cila përfshin elementët kryesorë si hyrjen në një ose në të gjithë tipet e burimeve të të dhënave, pranimi i të dhënave pavarësisht platformës në të cilën ato janë vendosur dhe ruajtja e burimit të të dhënave përmes përdorimit të praktikave rutinë të sigurisë.
- *Ruajtja e të dhënave* bën të mundur të pranojmë me lehtësi të dhënat të cilat mund të ruhen në tabela të ndryshme të të dhënave ose grupeve të të dhënave nëpër një shumëllojshmëri platformash. Nëpërmjet ruajtjes së të dhënave, mund të shkrijmë dhe ti bashkojmë të dhënat në fusha për të cilat duam.
- *Analizimi i të dhënave* kryhet nëpërmjet përdorimit të GUI-ve që shfrytëzojnë algoritmat e sofistikuara të Data Mining dhe përfshin kampionimin, zbulimin, modifikimin, modelin dhe pranimin.

- Kampionimi përmban informacionin më të rëndësishëm dhe më përfaqësues të bazës së të dhënave. Përpunimi i një kampioni përfaqësues në vend të të gjithë vëllimit të saj e zvogëlon shumë kohën e përpunimit të kërkuar.
- Pas kampionimit të të dhënave, hapi tjetër është ti zbulojmë ato vizualisht ose numerikisht për trende ose grupime të pandara.
- Modifikimi i të dhënave ka të bëjë me krijimin, zgjedhjen dhe transformimin e njërës ose më tepër variablave për tu fokusuar në proçesin e zgjedhjes së modelit në një drejtim të veçantë.
- Krijimi i një modeli të dhënash, i cili përfshin përdorimin e një software-i të përpunimit të të dhënave për të kërkuar automatikisht për një kombinim të të dhënave që parashikojnë në mënyrë të sigurtë një rezultat të dëshiruar.
- Hapi i fundit në Data Mining është pranimi i modelit dhe përcaktimi i performancën së tij.

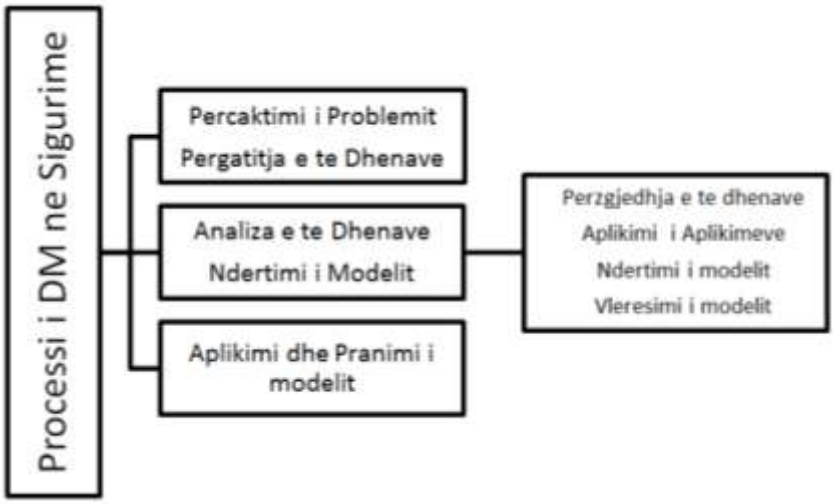


Figura 2 Proçesi i DM në Sigurime

2.3 TIPET E TË DHENAVE

Tipet më bazike të të dhënave për të zbatuar DM janë të dhënat nga bazat e të dhënave, nga baza të dhënash transaksionesh dhe nga data warehouse [12].

Bazat e të Dhënave

Bazat e të dhënave janë një nga depozitat më të pasura dhe të gjendshme dhe janë një formë mjaft e përhapur informacioni në DM. Kur gërmojmë bazat e të dhënave kërkojmë për prirje, trende dhe modele të dhënash. Për shembull, mund të analizojmë të dhënat e klientëve ose të detektojmë devijime te ndryshme. Aksesi i këtyre bazave të të dhënave realizohet nëpërmjet programeve të quajtur sisteme të menaxhimit të bazave të të dhënave. DBMS është një grup programesh që manipulojnë bazat e të dhënave dhe paraqitet si ndërmjetësues mes bazave të të dhënave, përdoruesve dhe programeve aplikative [13]. Modelimi i të dhënave kalon në tre faza:
Faza_1: Në projektimin llogjik kryhet identifikimi i relacioneve midis të dhënave, grupimi i tyre sipas një rendi dhe përfshirja e përdoruesve.
Faza_2: Në projektimin fizik bëhen modifikime të imtësishme, konsiderohen kostot dhe performanca, dhe në këtë fazë kemi njohje të thellë të DBMS.
Faza_3: Në modelimin e të dhënave kemi shtrimin e një problemi specifik dhe analizën e të dhënave, të cilat nevojiten për zgjidhjen e tij. Modelimi kryhet përmes diagramave të entiteteve dhe relacioneve mes tyre.

Një bazë të dhënash transaksionesh përbëhet nga disa tabela, të cilat përmbajnë informacione të nevojshme të lidhura me transaksionin. Çdo rekord në një bazë të dhënash transaksionesh i referohet një transaksioni, i cili përfshin një numër identiteti të vetëm dhe një listë të elementëve që përbëjnë transaksionin.

Data Warehouse

Një arkitekturë e depozitimit të të dhënave gjithnjë e më e përhapur është data warehouse (në vijim DWH). Një DWH [14] është një depozitë informacioni të mbledhur nga burime të ndryshme, që përdoret për raportime dhe analizë të dhënash.
Dallimi midis DWH dhe bazës së të dhënave është se ato janë multidimensionale, bazohet në të dhëna historike dhe përpunojnë volume shumë të medha të dhënash. Ato ndërtohen nëpërmjet një proçesi i cili përfshin pastrimin e të dhënave, integrimin, transformimin, ngarkimin e të dhënave dhe rifreskimin e vazhdueshëm të tyre.
Zakonisht një DWH modelohet nga një strukturë të dhënash shumë përmasore, e quajtur kubi i të dhënave, në të cilën çdo përmasë i përket një bashkësie cilësish ku çdo qelizë ruan një vlerë të caktuar. Një strukturë e tillë të dhënash jep mundësinë e një pamjeje shumëpërmasore të të dhënave dhe lejon përpunimin paraprak dhe akses më të shpejtë në përmbledhjen e të dhënave. Veprimet e përpunimit analitik online OLAP shfrytëzojnë njohuritë e fushës së të dhënave për të paraqitur të dhënat në nivele të ndryshme abstraksioni. DM shumëpërmasore kryen gërmime të dhënash në hapësirën shumë përmasore sipas metodës OLAP dhe nëpërmjet përpunimit paraprak dhe pamjeve shumë përmasore të të dhënave, sistemet DWH mundësojnë mbështetje për OLAP.

Tipe të tjera të Dhënash

Përmbajtja e të dhënave në DM është e shumëllojshme në forma dhe struktura të përshtatshme. Këto lloje të dhënash mund të vihen re në shumë aplikacione si: të dhëna sekuenciale apo të lidhura me kohën (rekorde historike, të dhëna mbi tregun e stoqeve, të dhëna kohore apo sekuenciale biologjike); stream-e të dhënash (të dhënat e vëzhgimeve me video, të dhënat e marra nga sensorë të ndryshëm të cilat transmetohen në mënyrë të vazhdueshme); të dhëna hapësinore (si për shembull hartat); të dhëna mbi projekte inxhinierike (projekte ndërtesash, përbërës sistemesh apo qarqe të integruar); hipertekste dhe të dhëna multimediale (përfshijnë tekste, figura, video dhe audio); të dhëna rrjeti dhe grafike (si psh të dhëna mbi rrjetet

informative dhe sociale); web-i (një depozitë e madhe e shpërndarë informacioni që ofrohet nga Interneti).

2.4 TIPET E MODELEVE

Në thelb të DM është ndërtimi i modeleve [15]. Një model është thjesht një algoritëm ose një bashkësi rregullash që lidhin një bashkësi inputesh me një output të caktuar dhe që tregojnë disa lidhje të cilat mund të ndikojnë në mënyrë domethënëse në disa lidhje të tjera. Ka dy mënyra për të përshtatur të dhënat me modelin e krijuar [16]:

1- *Mënyra Parashikuese* e cila përdoret për nxjerrjen e modeleve që përshkruajnë klasa të rëndësishme të të dhënave ose që parashikojnë tendencat e të dhënave në të ardhmen.
2- *Mënyra Përshkruese* lejon ose paraqet njohjen e lidhjeve që fshihen tek të dhënat dhe siguron rezultate të ndryshme.

Teknikat e përdorura në të dy mënyrat parashikuese dhe përshkruese kanë prerje me njëra-tjetrën.

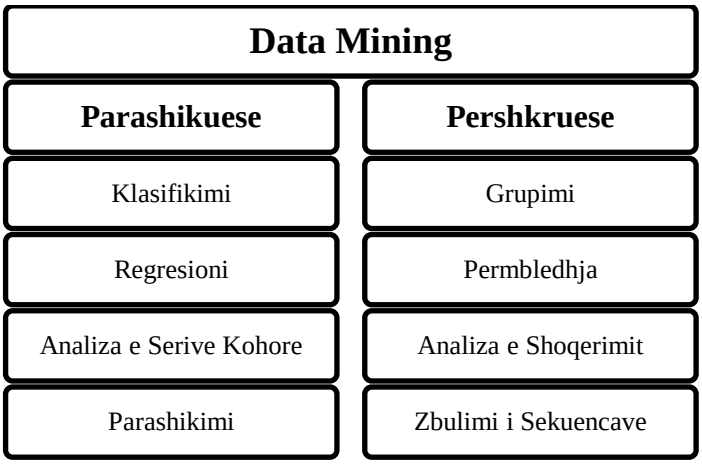


Figura 3 Teknikat e Data Mining

2.4.1 TEKNIKAT PARASHIKUESE

Teknikat parashikuese janë paraqitur shkurtimisht si më poshtë:

Klasifikimi është proçesi i gjetjes së një modeli, i cili parashikon klasat e të dhënave. Modeli derivohet në bazë të analizës së një bashkësie të dhënash trajnimi dhe përdoret për të parashikuar etiketën e klasës së objekteve për të cilat ajo është e panjohur. Modeli i derivuar mund të paraqitet në disa mënyra si rregulla

klasifikimi “nqs…atëherë”, pemë vendimi, formula matematikore, fqinjësia më e afërt dhe rrjeta nervore. Ajo zbatohet fillimisht në një bashkesi provë, për t’u caktuar një model klasifikimi i quajtur “klasifikues”. Pasi zhvillohet klasifikuesi, ai përdoret për të klasifikuar të dhënat e tjera në të njëjtat klasa të paracaktuara, që nuk ndodhen në bashkësinë provë.

Regresioni Ndërkohë që klasifikimi parashikon etiketa kategorike, diskrete, regresioni ka vlera të vazhduara funksionesh. Pra regresioni përdoret më shumë për të parashikuar vlera të dhënash numerike që mungojnë, sesa etiketa klasash diskrete. Analiza e regresionit eshtë një metodologji statistikore që shpesh përdoret për parashikimin numerik, megjithëse ekzistojnë edhe metoda të tjera. Regresioni gjithashtu përfshin identifikimin e shpërndarjes së trendeve bazuar në të dhënat e gatshme. Edhe këtu mund të përdoren pemët e regresionit, pemët e vendimmarrjes, nyjet e të cilave kanë vlera numerike në vend të vlerave kategorike. Regresioni linear është teknika matematikore që mund të përdoret për të gjeneralizuar një bashkësi të dhënash numerike nëpërmjet krijimit të një ekuacioni matematik. Regresioni logjik nga ana tjetër vlerëson probabilitetin e verifikimit të një ngjarjeje në rrethana të caktuara, duke përdorur faktorët e vëzhguar së bashku me ndodhjen apo jo të ngjarjes.

Seritë Kohore janë të ngjashme me regresionin, por përdorin cilësitë plotësuese të të dhënave të përkohshme për të parashikuar vlerat e ardhshme. Karakteristika e kësaj klase teknikash është pikërisht varësia e vlerave nga koha. Mund të jetë e rëndësishme për të ardhmen hierarkia e periudhave, sezonet, festat dhe pushimet, ditët e punës etj. Pasi krijohet modeli, përshtatet vazhdimi i testimit dhe në rast nevoje modifikohet edhe per të dhëna të reja.

Parashikimi përdor vlera të njohura për parashikimin e sasive të panjohura. Lidhet me analizën numerike dhe statistikën, për shkak se përdoren funksione interpolimi (lineare ose jo), të cilët përshkruajnë vlerat e parashikuara.

2.4.2 TEKNIKAT PERSHKRUESE

Teknikat përshkruese janë paraqitur shkurtimisht si më poshtë:

Grupimi (Clustering) Ndryshe nga klasifikimi dhe regresioni, të cilët analizojnë grupe të dhënash në klasa trajnimi, grupimi analizon objektet e të dhënave pa u konsultuar me etiketat e klasave. Në shumë raste, të dhënat e klasave të etiketuara

mund të mos ekzistojnë në fillim. Objektet grupohen në bazë të parimit të maksimizimit të ngjashmërisë brëndaklasore dhe minimizimit të ngjashmërisë ndërklasore. Pra grupet e objekteve janë formuar në mënyrë të tillë që objektet brënda një grupi të kenë ngjashmëri të madhe në krahasim me njëri-tjetrin, por dhe të jenë të dallueshëm nga objektet e grupeve të tjerë. Çdo grupim i formuar në këtë mënyrë mund të shihet si një klasë objektesh, nga e cila mund të derivohen rregulla.

Përmbledhja është një teknikë që nxjerr nga të dhëna shumë voluminoze vetëm informacione rilevante dhe kuptimplotë. Ajo jep një përshkrim të përmbledhur dhe të thjeshtë të një bashkësie ose nënbashkësie të dhënash, për shembull duke treguar karakteristikat e ndryshme të cilat e përbëjnë një produkt sigurimi jete.

Rregullat e Shoqërimit identifikojnë argumentet që gjenden së bashku me një të dhënë, ngjarje ose rekord: "prania e një bashkësie argumentesh sjell praninë e një bashkësie tjetër". Kështu identifikohen rregulla të tipit: "nëse argumenti A është pjesë e një ngjarjeje, atëherë për një probabilitet të caktuar edhe argumenti B është pjesë e ngjarjes".

Zbulimi i Sekuencave është i ngjashëm me analizën e shoqërimeve, përveç faktit se relacionet mes argumenteve janë të kushtëzuara nga koha. Zakonisht për të gjetur këto sekuenca, duhet te kapen jo vetëm detajet për cdo transaksion por edhe identiteti i personit që kryen transaksionin.

2.4.3 MATJET OBJEKTIVE DHE SUBJEKTIVE

Një model paraqet interes nëse është lehtësisht i kuptueshëm nga njerëzit, i vlefshëm për një grup të ri të dhënash me një shkallë të caktuar sigurie, potencialisht i përdorshëm dhe nëse vërteton një hipotezë që përdoruesi kërkon të konfirmojë. Një model i tillë përfaqëson njohje në DM.
Matjet objektive masin sasisë së interesit që paraqet një model dhe bazohen në strukturën e modeleve të zbuluar dhe statistikat që fshihen pas tyre. Megjithëse ato ndihmojnë në identifikimin e modeleve të dobishëm, ato shpesh janë të pamjaftueshme ndaj edhe kombinohen me matjet subjektive të cilat pasqyrojnë nevojat dhe interesat e veçanta të përdoruesit.
Matjet subjektive masin sasisë e interesit që paraqet një model dhe bazohet në besimin e përdoruesit tek të dhënat. Këto matje ofrojnë informacion strategjik mbi të cilin përdoruesi mund të veprojë. Modelet të cilat janë të pritshme mund të jenë të dobishëm nëse konfirmojnë një hipotezë që përdoruesi kërkon të vërtetojë.

Për të përqëndruar gërmimin e të dhënave nevojiten udhëzimet dhe matjet për shkallën e interesit nga përdoruesi. Modelet duhet të jenë efiçentë dhe të shkallëzueshëm me qëllim që të nxjerrin informacion me efiktivitet nga sasi të mëdha të dhënash në shumë depozita të dhënash apo në stream-e dinamike të dhënash. Koha e ekzekutimit të një modeli duhet të jetë e parashikueshme, e shkurtër dhe e pranueshme nga aplikacionet. Kriteret e vlerësimit më të rëndësishme që duhet të plotësoj një model i gjeneruar nëpërmjet DM që të jetë i dobishëm janë efiçenca, shkallëzueshmëria, saktesia dhe aftësia për t'u ekzekutuar në kohë reale.

2.5 TEKNOLLOGJITË E PËRDORURA

DM ka një natyrë ndërdisplinare dhe si e tillë shfrytëzon shumë teknika nga fusha të tjera si statistika, machine learning, sistemet e bazave të të dhënave dhe të DWH, vizualizimi, algoritmika, përdorimi i perpunimit ne paralel [17].

Metodat statistikore mund të përdoren për të verifikuar rezultatet e DM. Statistika studion mbledhjen, analizën, interpretimin, shpjegimin dhe prezantimin e të dhënave. Një model statistikor është një bashkësi funksionesh matematikorë që përshkruajnë sjelljen e objekteve në një klasë studimi në terma të ndryshoreve të rastësishme dhe shpërndarjen probabilistike të tyre. Kërkimi statistikor zhvillon mjete për parashikim duke përdorur të dhëna dhe modele statistikore. Metodat statistikore mund të përdoren për të përmbledhur apo përshkruar një bashkësi të dhënash; për të kuptuar mekanizmat që gjenerojnë dhë ndikojnë tek modelet. Shumë metoda statistikore kanë kompleksitet të lartë në përpunim kompjuterik dhe kur metoda të tilla zbatohen në grupe të mëdha të dhënash, algoritmet duhet të projektohen me kujdes duke patur parasysh reduktimin e kostos të përpunimit kompjuterik.

Machine Learning (në vijim ML) heton mënyrën se si kompjuterat mësojnë apo përmirësojnë performancën e tyre duke u bazuar në të dhëna. Një fushë e madhe kërkimore është ajo e mësimit të programeve që njohin automatikisht modele komplekse dhe që marrin vendime inteligjente duke u bazuar në të dhëna. Më poshtë do të ilustrohen disa mësime klasikë në ML që lidhen ngushtë me DM.

- Mësimi i mbikqyrur (supervised learning) është një sinonim për klasifikimin. Mbikqyrja në mësim vjen nga shëmbujt e etiketuar në bashkesi të dhënash provë.
- Mësimi jo i mbikqyrur (unsupervised learning) është një sinomin për grupimin. Proçesi i mësimit nuk është i mbikqyrur duke qenë se shembujt e dhënë nuk

janë të etiketuar apo të grupuar në klasa. Zakonisht përdoret metoda e grupimit për të zbuluar klasa mes të dhënave.

- Mësimi gjysmë i mbikqyrur (semi-supervised learning) është një klasë teknikash e ML, e cila shfrytëzon si modelet e etiketuara ashtu dhe ato jo të etiketuara kur mësohet një model.
- Mësimi aktiv është një qasje e ML që lejon përdoruesit të luajnë një rol aktiv në proçesin e të mësuarit.

Kërkuesit vazhdojnë të zhvillojnë metodologji të reja për proçesin e zbulimit të njohurive të dobishme nga të dhënat duke marrë në konsideratë pasiguritë, zhurmat dhe paplotësinë e të dhënave.

3 – TEKNIKA E KLASIFIKIMIT

Në këtë kapitull paraqitet një vështrim i përgjithshëm i klasifikimit të të dhënave, teknikave të tij, klasifikuesit pemë vendimi dhe kriteret për përzgjedhjen e atributeve në ndërtimin e modelit. Për algoritmin pemë vendimi CART kemi propozuar vendosjen e një ndarësi zëvendësues, i cili përmirëson performancën e tij. Kemi kombinuar dy teknikat zgjedhjen e madhësisë së largësisë dhe përafrimin drejt kombinimit të emërtimeve të grupeve për të përmirësuar performancën dhe rritur shkallen e saktësisë së klasifikimit për algoritmin e fqinjësisë më të afërt. Gjithashtu kemi paraqitur edhe aplikimin e rrjetave nervore në klasifikimin e të dhënave që përdoren në sigurimit e jetës. Për vlerësimin e modeleve kemi përdorur kriteret si saktësia, ndjeshmëria, shkalla e gabimit, analizën e kostove dhe përfitimeve dhe grafikun ROC.

3.1 ÇFARË ËSHTË KLASIFIKIMI?

Klasifikimi zbulon modele të cilat përshkruajnë klasa të rëndësishme të dhënash dhe parashikojnë etiketa klase kategorike diskrete dhe të parenditura. Sistemet që ndërtojnë klasifikues janë një prej mjeteve më të përdorura gjërësisht në përpunimin e të dhënave [18]. Klasifikimi i të dhënave është një proçes dy hapësh [19], i cili përbëhet nga hapi mesimi - ku ndërtohet modeli dhe hapi klasifikimi - ku modeli përdoret për të parashikuar etiketat e klasave për të dhënat.

Në hapin e parë, ndërtohet modeli, i cili përshkruan një bashkësi klasash të paracaktuara me të dhëna. Ky hap quhet ndryshe faza e trajnimit, ku algoritmi i klasifikimit ndërton klasifikuesin duke analizuar apo "duke mësuar" nga një bashkësi provë e përbërë nga tuple baza të dhënash dhe etiketa klasore shoqëruese.

Tuple e të dhënave quhen objektet, kampionet, shembujt, pikat e të dhënave në kontekstin e klasifikimit.

Një tuple [20] X, përfaqësohet nga një vektor atributesh n-përmasor, $X=(x_1,x_2,...,x_n)$, që përfaqësojnë n masa të marra në tuple nga n atribute baza të dhënash, përkatësisht, $A_1,A_2,...,A_n$. Çdo tuple X, supozohet se i përket një klase të paracaktuar siç është vendosur nga një atribut tjetër i bazës së të dhënave i quajtur atributi i etiketës së klasës, i cili merr vlera diskrete dhe të parenditura; është kategorik për shkak se çdo vlerë shërben si një klasë.

Tuplet individuale të cilat përbëjnë bashkësinë provë quhet tuple provë dhe kampionohen nga baza e të dhënave që po analizohet. Për shkak se njihet etiketa e klasës së çdo tuple, ky hap njihet si mësimi i mbikqyrur. Ndryshon nga mësimi jo i

mbikqyrur ku etiketa e klasës nuk njihet dhe bashkësia e klasave nuk njihet më parë.
Hapi i parë i proçesit të klasifikimit mund të shihet si mësimi i një funksioni, *y=f(X),* i cili mund të parashikojë etiketën përkatëse të klasës *y* kur jepet një tuple *X*. Pra duam të mësojmë një funksion që ndan klasat e të dhënave. Zakonisht ky funksion jepet në formën e rregullave të klasifikimit, pemëve të vendimeve apo formulave matematikore.
Rregullat mund të përdoren për të kategorizuar tuple të ardhshme të dhënash, të japin një pamje më të gjerë të përmbajtjes së të dhënave dhe një paraqitje të kompresuar të të dhënave. Saktësia e një klasifikuesi në një bashkësi test të dhënë, është përqindja e tupleve të bashkësisë test që janë klasifikuar drejt nga klasifikuesi.
Etiketa klasore përkatëse e çdo tuple test krahasohet me parashikimin e klasës për atë tuple të kryer nga klasifikuesi i mësuar. Nëse saktësia e klasifikuesit gjykohet e pranueshme, klasifikuesi mund të përdoret për të klasifikuar tuple të ardhshme të dhënash për të cilat etiketa e klasës nuk njihet [21].

3.2 PËRZGJEDHJA E ATRIBUTEVE

Përzgjedhja e atributive është proçesi që zgjedh kriterin ndarës, i cili veçon më mirë një bashkësi të dhënash *D,* prej tuplesh të etiketuara në klasa të veçanta. Nëse duhet të ndajmë *D* në pjesë më të vogla sipas rezultateve të kriterit të ndarjes, idealisht çdo pjesë duhet të jetë e pastër. Kriteri më i mirë ndarës është ai që kryen saktë këtë proçes dhe përcakton ndarjen e tupleve nga një nyje.

Përzgjedhja e atributeve seleksionon atributet që përshkruajnë tuplet. Atributi i renditur më lart zgjidhet si atributi ndarës për tuplet e të dhënave. Nëse atributi ndarës ka vlerë të vazhdueshme ose nëse jemi të detyruar të përdorim pemët binare, atëherë duhet të përcaktohen edhe pika e ndarjes apo nënbashkësia e ndarjes si pjesë e kriterit ndarës. Nyja e pemës etiketohet me kriterin ndarës, degët dalëse përmbajnë rezultatin e kriterit dhe në bazë të tij ndahet tuplet.

Kriteret për përzgjedhjen e atributeve janë: përfitimi i informacionit, koefiçienti i përfitimit dhe indeksi Gini.

Le të jetë *D,* bashkësia e trajnimit të të dhënave të tupleve të etiketuara. Supozojmë se atributi për etiketën e klasës ka *m* vlera të ndryshme, duke caktuar *m* klasa, C_i *(i=1,..., m).* Le të jetë $C_{i,D}$ bashkësia e tupleve të klasës C_i në *D* dhe le të jenë $|D|$ dhe $|C_{i,D}|$ numri i tupleve përkatësisht në *D* dhe $C_{i,D}$.

3.2.1 PËRFITIMI I INFORMACIONIT

Le të kemi nyjen *N* si nyjen ku ndodhen tuplet e bashkësisë *D*. Atributi me më shumë informacion zgjidhet si atributi ndarës për nyjen *N* dhe klasifikon tuplet në bashkësitë rezultante duke pasqyruar papastërtinë në këto bashkësi [22].

Në këtë mënyrë minimizohet numri i testeve të nevojshëm për të klasifikuar një tuple dhe jep si rezultat një pemë të thjeshtë. Informacioni i nevojshëm për të klasifikuar një tuple në *D* jepet nga shprehja:

$$Info(D) = -\sum_{i=1}^{m} p_i \, log_2(p_i) \qquad (3.1)$$

Ku:

p_i - është probabiliteti jozero që një tuple çfarëdo në D, bën pjesë në klasën C_i, i cili llogaritet nga formula $p_i = |C_{i,D}|/|D|$

$log_2(p_i)$ - funksioni logaritmik me bazë 2, i cili përdoret pasi informacioni është binar

$Info(D)$ - është vlera mesatare e informacionit të nevojshëm për të identifikuar etiketën e klasës së një tuple në *D*, e cila njihet ndryshe edhe si entropia e *D*.

Supozojmë se duhet të ndajmë tuplet në *D* në bazë të një atributi *A* që ka ***v*** vlera të ndryshme $\{a_1, a_2, \dots, a_v\}$. Nëse *A* ka vlera diskrete, këto vlera i takojnë një testi *A*, i cili i jep vlerë *v* rezultateve. Atributi *A* përdoret për të ndarë *D* në *v* particione apo nënbashkësi $\{D_1, D_2, \dots, D_v\}$, ku D_j përmban tuplet në *D* që kanë si rezultat nga *A*, a_j. Këto degë i pershtaten particioneve që dalin nga nyja *N*. Sigurisht që do të donim që kjo ndarje të japë një klasifikim të saktë të tupleve, por ka shumë mundësi që particionet të përmbajnë dhe tuple nga klasa të ndryshme dhe jo vetëm nga një klasë e vetme. Sasia e informacionit qe duhet pas ndarjes për të arritur në një klasifikim të saktë matet nga shprehja:

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} Info(D_j) \qquad (3.2)$$

Termi $|D_j|/|D|$ përfaqëson peshën e bashkësisë së j-të. $Info_A(D)$ është informacioni i nevojshëm që kërkohet për të klasifikuar një tuple nga D bazuar në ndarjen nga A. Sa më i vogël informacioni që kërkohet, aq më të pastra janë particionet. *Perfitimi i informacionit përcaktohet si diferenca mes kërkesës së informacionit fillestar dhe kërkesës së re i përfituar pas ndarjes sipas A.* Pra kemi:

$$Gain(A) = Info(D) - Info_A(D) \qquad (3.3)$$

Ku:

$Gain(A)$ - tregon sa informacion përfitohet nga ndarja në bazë të *A-së*. Atributi *A* me perfitimin më të madh të informacionit dhe $Gain(A)$ zgjidhet si atributi ndarës në nyjen *N* [23]. Kjo do të thotë që duam të kryejmë një ndarje sipas atributit *A* që do të kryejë klasifikimin më të mirë, në mënyrë që sasia e informacionit që kërkohet më tutje për përfundimin e klasifikimit të jetë minimale.

Nëse *A* ka vlera të vazhdueshme, duhet të përcaktohet një pikë ndarëse për *A*. Fillimisht renditen vlerat e ndeshura te tuplet dhe mesatarja midis çdo çifti vlerash të njëpasnjëshme merret si një pikë ndarëse e mundshme. Ndaj kur jepen v vlera të *A*, atëherë shihen *v-1* ndarje të mundshme. Nëse vlerat e *A* janë renditur më parë, atëherë përcaktimi i ndarjes më të mirë kërkon vetëm një kalim të vlerave. Për çdo vlerë të mundshme për *A*, vlerësohet $Info_A(D)$, ku numri i particioneve është dy, pra *v=2 (apo j=1,2).* Pika me informacionin minimal të pritshëm është kërkesë për A-në që të përzgjidhet si pika_ndarëse. D_1 - është bashkësia e tupleve në *A* që kënaqin mosbarazimin *A≤ pika_ndarëse* dhe D_2 - bashkësia e tupleve në *A* që nuk e kënaqin mosbarazimin.

3.2.2 KOEFIÇIENTI I PËRFITIMIT

Përfitimi i informacionit kërkon teste me shumë rezultate, pra priret të përzgjedhë atribute që kanë një numër të madh vlerash ndërsa koefiçienti i përfitimit nga ana tjetër zbaton një lloj normalizimi në perfitimin e informacionit duke përdorur një vlerë "informacion ndarje", e cila përcaktohet në mënyrë analoge me *Info(D):*

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} log_2\left(\frac{|D_j|}{|D|}\right) \quad (3.4)$$

Kjo vlerë përfaqëson informacionin e mundshëm të gjeneruar nga ndarja e bashkësisë së të dhënave, *D*, në *v* particione, që i përkasin *v* rezultateve të një testi mbi atributin A.

Duhet pasur parasysh se për çdo rezultat, konsiderohet numri i tupleve që kanë atë rezultat, në krahasim me numrin total të tupleve në D. Koeficienti i perftimit përcaktohet në këtë mënyrë:

$$GainRatio(A) = \frac{Gain(A)}{SplitInfoA(D)} \quad (3.5)$$

Atributi me koefiçientin maksimal të përfitimit përzgjidhet si atributi ndarës. Për të evituar një emërues zero, përfitimi i informacionit të testit të përzgjedhur duhet të jetë të paktën sa përftimi mesatar i të gjithë testeve.

3.2.3 INDEKSI GINI

Indeksi Gini mat papastërtinë e *D* [24], e cila eshte një bashkësi tuplesh provë. Ai llogaritet si më poshtë:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2 \quad (3.6)$$

Indeksi Gini merr në konsideratë një ndarje binare për çdo atribut. Le të marrim fillimisht rastin kur *A* merr vlera diskrete dhe ka *v* vlera të ndryshme $\{a_1,a_2,...,a_v\}$ në *D*. Për të përcaktuar ndarjen binare më të mirë në *A*, shihen të gjithë nënbashkësitë e mundshme që mund të formohen.

Çdo nënbashkësi S_A, mund të konsiderohet si një test binar për atributin *A* te formës "$A \in S_A$*?"*. Për një tuple të dhënë, ky test kalohet nëse vlera e *A* për këtë tuple është mes vlerave të S_A*-së*. Nëse *A* ka *v* vlera të mundshme, atëherë ka 2^v*-2* mënyra të ndryshme për të formuar dy particione të dhënash në *D*, bazuar në një ndarje binare sipas *A*. Kur konsiderohet një ndarje binare, llogaritet një shumë e peshuar e papastërtisë të secilit particion. Për shembull, nëse kemi një ndarje binare sipas *A*, ku *D* ndahet në D_1 dhe D_2, Indeksi Gini i *D* në këtë rast llogaritet si më poshtë:

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \quad (3.7)$$

Për çdo atribut, konsiderohet çdo ndarje binare e mundshme. Për një atribut me vlera diskrete, nënbashkësia që jep një indeks Gini minimal për atë atribut, zgjidhet si nënbashkësia ndarëse. Për atributet me vlera të vazhdueshme, duhet të konsiderohet çdo pikë ndarëse e mundshme.

Strategjia është e njëjta me atë të përshkruar më parë në përftimin e informacionit, ku mesatarja midis çdo çifti vlerash të renditura të njëpasnjëshme merret si një pikë ndarëse e mundshme.

Pika që jep indeksin Gini minimal, për një atribut të dhënë, merret si pika ndarëse për atë atribut. Reduktimi i papastërtisë per një ndarje binare sipas një atributi me vlera diskrete apo të vazhdueshme llogaritet si më poshtë:

$$\Delta Gini(A) = Gini(D) - Gini_A(D) \quad (3.8)$$

Atributi që maksimizon reduktimin e papastërtive, apo ka vlerën minimale të indeksit Gini, përzgjidhet si atributi ndarës. Ky atribut dhe nënbashkësia e tij ndarëse (për një atribut me vlera diskrete) apo pika ndarëse (për një atribut me vlera të vazhdueshme) përbëjnë kriterin ndarës.

3.3 PEMËT E VENDIMIT

Pemët e vendimit janë një metodë klasifikimi e përdorur gjerësisht për klasifikimin e të dhënave. Ato janë të njohura për shkak se në ndërtimin e tyre nuk është e nevojshme vendosja e ndonje parametri apo kriteri paraprak. Klasifikuesit pemë vendimi janë vetëshpjegues, trajtojnë atribute me vlerë numerike dhe nominale, bëjnë pjesë në grupin e klasifikuesve me vlere diskrete, janë të afta të trajtojnë baza të dhënash me gabime dhe mungese të dhënash dhe janë joparametrike [25].
Struktura e tyre është e ngjashme me një pemë dhe zhvillimi i saj kalon në dy hapa:
Hapi i parë: Faza e ndërtimit
Për të ndërtuar një pemë atributet duhet të jenë te njohura. Kjo përzgjedhje attributesh bëhet në bazë të kritereve si perftimi i informacionit, koeficienti i perfitimit dhe indeksi Gini etj. Këto kritere janë të dobishme për të zgjedhur një atribut më të mirë pasi dallon tuplet qe i përkasin një klasë. Për të vendosur ndarjen më të mirë të atributeve më sipër vazhdojme derisa të gjitha të dhënat ti përkatësin të njëjtës klasë. Nyjet e klasës quhen nyje gjethe.
Hapi i dytë: Faza e thjeshtimit
Faza e thjeshtimit përdoret për të shmangur anomalite në pemë. Parathjeshtimi, pasthjeshtimi, kostot komplekse te thjeshtimit janë metodat për të thjeshtuar pemën dhe si rrjedhim përftojmë një pemë të optimizuar [26].

3.3.1 ALGORITMI PEMË VENDIMI

Pranimi i pemëve të vendimit është mësimi i pemëve të vendimit nga tuplet provë të etiketuara. Një pemë vendimi është një strukturë peme si flowchart, ku çdo nyje e brëndshme shënon një test në një atribut, çdo degë përfaqëson rezultatin e testit dhe çdo gjethe përmban një etiketë klase. Nyja e parë e pemës është rrënja. Disa algoritme pemë vendimi prodhojnë vetëm pemë binare (ku çdo nyje e brendshme degëzohet në vetëm dy nyje të tjera), ndërkohë që të tjera mund të prodhojnë pemë jo binare [27].
Shumica e algoritmeve për pranimin e pemës së vendimit fillojnë me një bashkësi provë dhe më pas ndahet në mënyrë rekursive në bashkësi më të vogla ndërkohë që ndërtohet pema. Disa algoritme pemësh vendimi prodhojnë vetëm pemë binare (ku çdo nyje e brendshme degëzohet në vetëm dy nyje të tjera), ndërkohë që të tjera mund të prodhojnë pemë jo binare.
Algoritmet të zhvilluara sipas kësaj metode përfshijnë *ID3, C4.5, CART, CHAID*. Më poshtë jepet algoritmi për gjenerimin e një pemë vendimi:

***Algoritmi**: GPV- Gjenerimi i një peme vendimi nga një bashkësi të dhënash, D.*
Të dhëna hyrëse:

- *Bashkësia e të dhënave D, e cila është një bashkësi tuple provë dhe etiketat e tyre përkatëse.*
- *Attribute_list - bashkësia e atributeve kandidate; Attribute_selection_method - një procedurë për të përcaktuar kriterin ndarës që ndan më mirë tuple e të dhënave në klasa të veçanta. Ky kriter përbëhet nga splitting_attribute dhe një split_point ose splitting_subset.*

***Rezultati**: Një pemë vendimi.*
***Metoda[ ]**:*

Krijo nje nyje N;
if** tuplet ne D jane te gjtha ne te njejten klase C, **then
Kthen N si nje nyje gjethe etiketuar me klasen C;
if** attribute_list eshte bosh **then
kthen N si nje nyje gjethe etiketuar me klasen shumice ne D;
*aplikojme **Attribute_selection_method** (D,attribute_list) per te gjetur "me te mirin" splitting_criterion;*
Nyja etiketike N me splitting_criterion;
if** splitting_attribute eshte is vlere diskrete-dhe ndarjet shume dege jane lejuar **then
attribute_list←attribute_list-splitting_attribute;
***for each** rezultat j nga spitting_criterion*
le te jete D_j tuple te dhenash te caktuara ne D qe kenaqin rezultatin j;
if** D_j eshte bosh **then
bashkangjit nje gjethe etiketuar me klasen shumice ne D te nyja N;
***else** bashkengjit nyjen e kthyer nga **GPV** (D_j,attribute_list) te nyja N;*
***end for** kthe N;*

Figura 4 Algoritmi për gjenerimin e një peme vendimi

Pema fillon me një nyje të vetme, *N*, që përfaqëson tuplet provë në *D*. Nëse tuplet në *D* janë të gjitha të një klase, atëherë nyja *N* bëhet gjethe dhe etiketohet me atë klasë, nëse jo, algoritmi thërret Attribute_selection_method për të për të përcaktuar kriterin ndarës, i cili tregon se kush atribut testohet në nyjen *N* për të përcaktuar mënyrën më të mirë të ndarjes të tupleve në *D* në klasa të veçanta. Algoritmi përdor të njëjtin proçes në mënyrë rekursive për të formuar një pemë vendimi për tuplet në secilën nënbashkësi D_j të *D* [28]. Ndarja rekursive ndalon atëherë kur plotësohet një nga kushtet përfunduese:

1. Të gjitha tuplet në bashkësinë *D* i përkasin së njëjtës klasë.

2. Nuk ka më atribute sipas të cilave mund të ndahen tuplet. Në këtë rast zbatohet votimi me shumicë. Kjo do të thotë shndërrimi i nyjes *N* në gjethe dhe etiketimi i saj me klasën më të përhapur në *D*.
3. Nuk ka më tuple për një degë të caktuar, pra një bashkësi D_j është bosh. Në këtë rast krijohet një gjethe me klasën e shumicës në *D*.

Kompleksiteti i algoritmit kur jepet bashkësia *D* është $O(n|D|log(|D|))$ ku n është numri i atributeve që përshkruajnë tuplet në *D* dhe $|D|$ është numri i tupleve provë në *D*.

3.3.2 PËRDORIMI I RREGULLAVE NQS-ATËHERË

Një klasifikues i bazuar në rregull përdor një bashkësi rregullash Nqs-atëherë për klasifikimin. Një rregull Nqs-atëherë shprehet në formën: Nqs kushti Nqs-atëherë përfundimi. Psh marrim rregullin *R1*:

Nqs mosha=iri ***dhe*** *arsimi=ilarte* ***atëherë*** *blen_sigurim_jete=po.*

Pjesa e nqs, njihet si parakushti; ndërsa pjesa atëherë është pasoja e rregullit. Në parakusht, kushti përbëhet nga një apo më shumë teste atributesh, të cilat lidhen me njëra-tjetrën nëpërmjet lidhëzës dhe (në mënyrë logjike). Pasoja e kushtit përman një parashikim klase. *R1* mund të shkruhet edhe si:

$$R1: (mosha=iri) \wedge (arsimi=ilarte) \Rightarrow (blen\_sigurim\_jete=po)$$

Nëse kushti është i vërtetë për një tuple të caktuar, atëherë themi se parakushti plotësohet dhe rregulli mbulon atë tuple. Një rregull *R* mund të gjykohet nga mbulimi dhe saktësia dhe përcaktohen si më poshtë:

për një tuple të dhënë *X*, nga bashkësia e të dhënave të etiketuara, *D*, janë dhënë:

n_{mb} - numri i tupleve të mbuluara nga *R*

n_{klas} - numri i tupleve të klasifikuara drejt sipas *R*

$|D|$ - numri i tupleve në *D*

Mbulimi i një rregulli është përqindja e tupleve që mbulohen nga rregulli.

$$mbulimi(R)=n_{mb}/|D| \tag{3.9}$$

Saktësia e një rregulli është pjesa e tupleve që mbulohen, të cilat klasifikohen drejt rregullit.

$$saktësia(R)=n_{klas}/n_{mb} \tag{3.10}$$

Nëse rregulli plotësohet nga *X*, thuhet se rregulli trigërohet. Nëse rregulli *R* është i vetmi që plotësohet, ai rregull kthen parashikimin e klasës për *X*. Fakti që një tuple plotëson një rregull, jo domosdoshmërisht do të thotë parashikimin e klasës për të.

Ai mund të plotësojë më shumë se një rregull (të cilët mund të parashikojnë klasa të ndryshme) apo mund të mos plotësojë asnjë rregull.
Nëse plotësohet më shumë se një rregull kërkohet një strategji për zgjidhje konflikti për të zgjedhur rregullin përcaktues dhe për t'i caktuar klasën atij tupli që plotëson këtë rregull. Një strategji është renditja sipas madhësisë, e cila i cakton prioritet më të lartë, rregullit të trigëruar, i cili ka më shumë kërkesa pra ka madhësi më të madhe të parakushtit, thënë ndryshe është rregulli i cili ka më shumë teste atributesh. Strategjia tjetër është renditja sipas rregullit, e cila i jep prioritet rregullit më parë. Renditja mund të jetë e bazuar në klasë apo në rregull. Renditja e bazuar në klasë, rendit klasat sipas rëndësisë- në rendin rënës, ose sipas frekuencës së shfaqjes-në rendin rënës. Pra rregullat me klasat më të shpeshta janë në fillim të ndjekura nga rregullat me klasa më pak të shpeshta.

3.3.3 NXJERRJA E RREGULLIT NGA NJË PEMË VENDIMI

Klasifikuesit pemë vendimi janë një metodë e përhapur e klasifikimit për shkak se mënyra se si pemët e vendimit funksionojnë është lehtësisht të kuptueshme dhe njihen për saktësinë e tyre. Megjithatë ato mund të bëhen të mëdha dhe të vështira për t'u interpretuar. Për të nxjerrë rregulla nga një pemë vendimi, një rregull krijohet për çdo dege nga rrënja në një nyje gjethe. çdo kriter ndarës gjatë një dege, futet në një DHE llogjike për të formuar parakushtin (pjesën "Nqs"). Gjethja përmban parashikimin e klasës duke formuar pasojën e rregullit (pjesën "atëherë"). Mes rregullave të nxjerra zbatohet një Ose llogjike. Për shkak se rregullat nxirren direkt nga pema, kemi një rregull për gjethe dhe një tuple-i i shoqërohet vetëm një gjethe, gjithashtu kemi një rregull për çdo kombinim vlerë-atribut.

3.3.4 ALGORITMI CART

Algoritmi CART (Classification and Regresion Trees) është zhvilluar nga Breiman, Friedman, Olshen, Stone në fillim të viteve '80. Pema e vendimit CART është një pemë vendimi binare që ndërtohet duke ndarë një nyje në dy nën-nyje në mënyrë të përsëritur, duke filluar nga nyja rrënjë.
Të dhënat trajtohen në formën e tyre fillestare të papërpunuar. Pemët rriten deri në një madhësi maksimale pa qënë nevoja e përdorimit të një rregulli ndalues përpunimi të së dhënës, më pas ndodh proçesi i thjeshtësimit të termave të cilët marrin pjesë. Ndarja e rradhës, e cila mbart thjeshtimin e termave është njëra procedurë, e cila kontribon ndoshta më pak në drejtim të performancës së përgjithshme të pemës për trajnimin e së dhënës. Procedura këtu prodhon pemë, të

cilat janë të pandryshueshme sipas një rendi ruajtës për transformim të cilësive dhe vetive parashikuese të të dhënave.
Mekanizmi CART i përpunimit të së dhënës ka për qëllim të prodhojë jo një, por një seri pemësh fole të thjeshtimit të termave, të cilët marrin pjesë në përpunimin e saj, ku secila prej tyre është pemë optimale e përftimit të termave apo kufizave që marrin pjesë në modelin e përpunimit. Pema "me madhësi të saktë" ose pema "e pastër" identifikohet nëpërmjet vlerësimit të performancës parashikuese të secilës pemë në një rradhë thjeshtimi [29].
CART nuk ofron madhësi të brendshme performance për zgjedhjen e pemës së vendimit që bazohet në të dhënat e marra për trajnim, të cilat janë vlerësuar si madhësi të dyshimta. Në vend të saj mundëson performancën e pemës që bazohet në të dhenat test të pavarura dhe zgjedhja e pemës së të dhënave ndjek atë rrugë vetëm pas vlerësimit të bazuar në të dhënat për testimin e saj. CART përqëndrohet në pjesën më të madhe mbi indeksi Gini. CART në mungesë të vlerave të parametrit, gjithmonë llogarit shpeshtësitë në grup në secilën nyje që ka të bëjë me shpeshtësinë e grupit në qelizën bazë [30]. Kjo është ekuivalente me ripeshimin automatik të të dhënës për të drejtëpeshuar grupet e të dhënave dhe për të siguruar faktin që pema e zgjedhur si më optimalja minimizon gabimin e drejtëpeshuar të grupit. Ripeshimi është i pashtjelluar përsa i përket llogaritjes së të gjithë probabiliteteve dhe përmirësimeve; seritë e raportuara të shembujve në secilën nyje reflektojnë kësisoj të dhënën e paponderuar. Për një synim binar (0/1) secila nyje klasifikohet ne grupin 1 nëse:

$$N_1(nyje)/N_1(rrënjë) > N_0(nyje)/N_0(rrënjë) \quad (3.11)$$

Kjo mënyrë e përpunimit me mungesë të të dhënave merret si "e barabartë me përparësi", e cila lejon përdoruesit e CART të punojnë posaçërisht me çdo të dhënë të padrejtëpeshuar, e cila nuk kërkon vlera speciale brënda klasës në paraqitjen e ponderimeve të ndërtuara me dorë. Riponderimi i pashtjelluar mund të shmanget nëpërmjet zgjedhjes së opsionit të "së dhënave të përftuara paraprakisht" dhe modeluesi mundet gjithashtu të zgjedhë të specifikojë një bashkësi arbitrare vlerash prioriteteve për të pasqyruar kostot ose ndryshimet e mundshme ndërmjet të dhënës për trajnim dhe shpërndarjeve synuese të ardhshme të së dhënës brënda grupit.
Ne CART mekanizmi i përpunimit të vlerës me humbje është plotësisht automatik dhe lokalisht i përshtatur kundrejt çdo nyje. Në secilën nyje në pemë, ndarësi i zgjedhur i vlerave bën një ndarje binare të të dhënave [31].

Kostoja për një klasifikim të gabuar me anë të algoritmit CART për një vlerë *i,j* të grupit emerohet *C(i, j)* dhe është e barabartë me 1. Për vlera *i=j* atëherë *C(i, i)=0*. Çdo pemë klasifikimi mund të përmbajë një kosto të plotë të llogaritur për klasifikimet fundore të nyjes nëpërmjet grupimit të kostove në kufirin e të gjithë klasifikimeve të gabuara të bëra. Problemi në instruktimin me orientim koston është të nxirret një pemë që të përmbajë të gjithe koston në llogari gjatë grupimit të tyre dhe fazave të thjeshtimit të të dhënave. Pemët rriten derisa ato ndeshen me një kusht ndalues dhe pema rezultante mund të jetë përfundimtarja. Asnjë rregull që ka për qëllim të ndalojë rritjen e pemës mund të garantojë faktin se ajo nuk do të humbasë strukturën e të dhënave. Pema rezultante shumë e madhe siguron materialin e papërpunuar të të dhënave nga i cili nxirret një model optimal. Mekanizmi i thjeshtimit mbështetet rigorozisht në të dhënat për trajnim dhe funksionon me një vlerë komplekse kostoje e përcaktuar si më poshtë:

$$R_a(T) = R(T) + a|T| \quad (3.12)$$

Ku:

- *R(T)* është kosto e tuples për trajnim të pemës
- $|T|$ është numri i nyjeve fundore në pemë
- *a* përbën një penalitet të vënë mbi secilën nyje

Nëse $a = 0$ atëherë pema me vlerë minimale komplekse të kostos është saktësisht më e madhja e mundshmja.

Nëse parametri *a* rritet progresivisht nga *0* deri në një vlerë të mjaftueshme për të thjeshtuar të gjitha ndarjet në klasifikim, pema optimale përcaktohet si pemë në radhitjen e thjeshtuar të madhësisë, e cila arrin koston minimale në të dhënat test.

3.3.5 Modifikimi i Propozuar në Përmirësimin e Performancës

Vendosja e një ndarësi zëvendësues, i cili përbën një vlerë unike *k* që mund të parashikojë këtë ndarje në të cilën vetë zëvendësuesi ndodhet në formën e një ndarësi binar të së dhënës. Me fjalë të tjera çdo ndarës bëhet një synues i ri, i cili parashikohet me një pemë ndarëse unike binare. Zëvendësuesit janë të klasifikuar me anë të pikëve të grumbulluara që japin përparësi zëvendësuesit për rregullin e përftimit në mungesë të vlerave të madhësisë që parashikojnë se të gjitha vlerat shkojnë drejt nyjes më të madhe fëminore. Për të cilesuar një zevendësues, variabli duhet të tejkalojë këtë rregull në zbatim të perftimit të vlerave në mungese të madhësisë. Kur një vlerë me humbje gjëndet në një pemë CART madhësia spostohet majtas ose djathtas në përputhje kjo me zëvendësuesin e renditur më në krye.

Rëndësia e një vlere madhësi bazohet në shumën e përmirësimeve të bëra me të gjitha nyjet në të cilat vlera e madhësisë shfaqet si një ndarës. Zëvendësuesit janë të përfshirë gjithashtu në llogaritjet e peshës. Kjo i lejon klasifikimet e ndryshueshëm të peshës të zbulojnë fshehjen e ndryshueshme të vlerave të madhësisë dhe bashkëlidhjen jolineare ndërmjet të gjitha vetive të parametrit lidhës. Pikët sipas të cilave vlerësohet rëndesia ose pesha e madhësisë mundet të kufizohen në mënyrë optimale në ndarësit dhe krahasimin vetëm të ndarësve si edhe klasifikimet e plota të peshës apo të rëndësisë që në këtë rast përbëjnë diagnozën e duhur brënda grupit të madhësisë.

3.4 ALGORITMI I FQINJËSISË MË TË AFËRT TË RENDIT TË K-TË

Algoritmi i fqinjësisë më të afërt të rendit të k-të [32] (ne vijim KNN), u përshkrua për herë të parë në fillim të viteve 1950. Ai është efikas per grupe të mëdha trajnimi dhe është përdorur gjerësisht në fushën e njohjes se modeleve. Klassifikuesit KNN janë të bazuara në të mësuarit me analogji, dmth krahason një tuple provë nëse eshte e ngjashme me nje tuple trajnimi. Tuplet e trajnimit janë përshkruar nga *n-atribute*. Çdo tuple përfaqëson një pikë në një hapësirë *n-dimensionale*. Në këtë mënyrë, të gjitha tuplet trajnimit janë të ruajtura në një model hapësirë *n-dimensionale*. Kur jepet një tuple e panjohur, një klasifikues KNN kërkon në hapësirë për *k* - tuple trajnimi që janë më të afërt me tuplen e panjohur. Këto *k* tuple trajnimi janë *k* "fqinjët më të afërt" të tuples së panjohur [33]. "Afërsia" është përcaktuar në termat e një largësie metrike Euklidiane ose Manhatan.

Le të jenë dhënë $i = (x_{i1}, x_{i2},.., x_{ip})$ and $j = (x_{j1}, x_{j2},..., x_{jp})$ dy objekte të përshkruara nga *p-atribute* numerike.

Largësia Euklidiane midis dy objekteve i dhe j përcaktohet nga formula e mëposhtme:

$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ip} - x_{jp})^2} \quad (3.13)$$

Ndërsa largësia Manhatan llogaritet nga formula si më poshtë:

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + |x_{ip} - x_{jp}| \quad (3.14)$$

Të dyja largësitë gëzojnë vetitë:

- Largësia nuk është një numër negativ: $d(i, j) \geq 0$
- Largësia e një objekti i të dhënë me veten e vet është zero: $d(i, i) = 0$
- Largësia është një funksion simetrik: $d(i, j) = d(j, i)$

Ekzistojnë tre elemente kryesore të algoritmi të fqinjësisë më të afërt:

- një bashkësi objektesh të emërtuara
- një largësi metrike ngjashmërie për tu llogaritur ndërmjet objekteve
- dhe vlera e k-së, numri i fqinjëve më të afërt

Të dhëna hyrëse:

D bashkësia e k objekteve trajnuese dhe objekti I testit $z = (x', y')$

Përpunimi:

Llogarit $d(x', x)$ *, largësia ndërmjet çdo objekti* $(x, y) \in D$*;*

Zgjidh $D_z \subseteq D$ *, bashkësinë e k objekteve më të afërta trajnuese për z.*

Rezultati:

$$y' = \arg\max_{v} \sum_{(x_i, y_i \in D_z)} I(v = y_i)$$

Figura 5 Algoritmi i fqinjësisë më të afërt të rendit të k-të

Më sipër është paraqitur një përmbledhje e algoritmit të klasifikimit për kufizën më të afërt. Jepet një bashkësi trajnuese D dhe një objekt testi ose prove $x = (x', y')$ algoritmi llogarit largësinë (ose ngjashmërinë) ndërmjet z dhe të gjithë objekteve trajnues $(x, y) \in D$ për të përcaktuar listën me kufizën më të afërt, D_z.(x është e dhëna e objektit të trajnimit, vlera y është klasa e tij, ndersa x' përfaqëson të dhënën e objektit të testit dhe y' klasën e tij.) Pasi lista me kufizën më të afërt të përftohet, objekti i testit klasifikohet duke qënë i bazuar në klasën më të madhe të fqinjit te tij më të afërt [34]:

$$\text{Votimi i Shumicës: } y' = \arg\max_{v} \sum_{(x_i, y_i \in D_z)} I(v = y_i) \qquad (3.15)$$

Ku:

v është një emërtim grupi.

y_i është emërtimi i grupit për kufizën e i-të më të afërt.

$I(\cdot)$ është një funksion tregues i cili rikthen vlerën 1 nëse argumenti i tij është i vërtetë dhe 0 një vlerë tjetër e tij.

3.4.1 MODIFIKIMI I PROPOZUAR PËR PËRMIRËSIMIN E PERFORMANCËS

Problemet që ndikojnë në performancën e KNN janë zgjedhja e k, përafrimi drejt kombinimit të emërtimeve të grupeve dhe zgjedhja e madhësisë së largësisë.

Zgjedhja e k: Nëse k është shumë e vogël atëherë rezultati mund të jetë i ndjeshëm ndaj pikave problematike. Nga ana tjetër nëse k është shumë e madhe, atëherë zona mund të përfshijë shumë pika nga grupe të tjera të të dhënës [35].

Përafrimi drejt kombinimit të emërtimeve të grupeve [36]: Metoda më e thjeshtë është të përftosh votën e shumicës, por ky mund të përbëjë një problem nëse kufizat më të afërta ndryshojnë shumë përsa i përket largësisë së tyre dhe kufizat më të afërta tregojnë në mënyrë më bindëse grupin e objektit.

Zgjedhja e madhësisë së largësisë: Ndonëse madhësi të ndryshme të kësaj largësie mund të përdoren për të llogaritur largësinë ndërmjet dy pikave, madhësia më e dëshiruar e largësisë është ajo për të cilën një largësi më e vogël ndërmjet dy pikave tregon për një mundësi më të madhe të të paturit të të njëjtit grup madhësie. Disa madhësi largësie mund të ndikohen gjithashtu nga përmasat e larta të të dhënave [37].

Kemi kombinuar dy problematikat zgjedhjen e madhësisë së largësisë dhe përafrimin drejt kombinimit të emërtimeve të grupeve për të përmirësuar performancën dhe rritur shkallën e saktësisë së klasifikimit te algoritmit KNN. Ponderojmë çdo votë të objektit me largësinë e tij, ku faktori i ponderimit është i anasjellti i largësisë katrore:

$$w_i = 1/d\left(x^{'}, x_i\right)^2 \tag{3.16}$$

Këto shuma zëvëndësohen në hapin final të algoritmit KNN si më poshtë:

Të dhëna hyrëse:
D bashkësia e k objekteve trajnuese dhe objekti I testit $z = (x', y')$
Përpunimi:
Llogarit $d\left(x^{'}, x\right)$ *, largësia ndërmjet çdo objekti* $(x, y) \in D$ *;*
Zgjidh $D_z \subseteq D$ *, bashkësinë e k objekteve më të afërta trajnuese për z.*
Rezultati:

$$y^{'} = \arg\max_{v} \sum_{\left(x_i, y_i \in D_z\right)} w_i \times I\left(v = y_i\right)$$

Figura 6 Algortmi i fqinjësisë më të afërt të rendit të k-të i modifikuar

Algoritmi KNN është një mënyrë e lehtë për tu kuptuar dhe zbatuar për sa i perket teknikës për klasifikim. Eshte i thjeshtë, ekzekutohet mirë në shumë situata dhe i lehtë për të shpjeguar rezultatet [38].
Por, algoritmi KNN ka nevojë për më shumë hapësirë për të ruajtur të gjitha shembujt dhe merr më shumë kohë për të klasifikuar një shembull të ri. KNN paraqitet veçanërisht e përshtatshme për grupet multi-modale si dhe në aplikimet në të cilët një objekt mund të ketë shumë emërtime brënda grupit [39].

3.5 RRJETAT NERVORE

Një rrjetë nervore përbëhet nga elementët proçesues, të cilët shumëzojnë inputet me pesha të ndryshme, për të nxjerrë një rezultat output të vetëm. Pika e fortë e rrjetave nervore qendron në përpunimin paralel të njësive të ndërlidhura dhe natyrës së përshtatshme të peshave [40]. Vështirësitë nga ana tjetër qëndrojnë në lidhjet e njësive me njera-tjetrën, në vendosjen e peshave dhe në vlerat kufi. Për më tepër ndërkohë që një rrjetë nervore është duke mësuar mund të bëjë dhe gabime. Në këto raste ndryshohen peshat dhe vlerat kufi për të kompensuar gabimin [41].
Rrjetat nervore që mund të quhen ndryshe edhe sisteme përshtatëse kur përdoret një bazë të dhënash të madhe shembujsh të mëparshëm nga ana e sistemit për të mësuar nga eksperiencat e mëparshme. Zbatime të kësaj teknologjie janë kontrolli i cilësisë, parashikimi i motit, parashikime financiare, njohja e të folurit dhe shkrimit, zbulimi i naftës dhe gazit, ulja e kostos së shërbimit shëndetësor, parashikimi i falimentimit [42].

Shembulli më thjeshtë i një rrjeti nervor është një perceptron, një neuron i vetëm i stërvitshëm. Një perceptron prodhon një vlerë buleane nga inputet $x_1,x_2,...,x_n$ të cilat mund të jenë ose jo vlera buleane. Z është një vlerë buleane; P_i-të përfaqësojnë peshat dhe janë vlera reale dhe K është vlera kufi reale. Këtu rezultati Z është i vërtetë, nëse inputi $p_1x_1+p_2x_2+...+p_nx_n$ është më i madh se vlera kufi, përndryshe Z është zero.

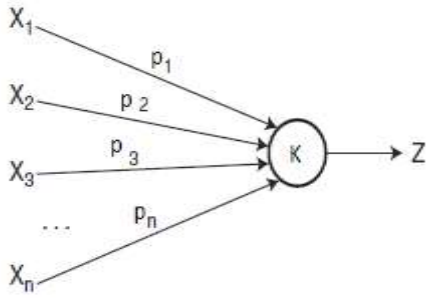


Figura 7 Nje perceptron

Një perceptron [43] prodhon outpute për inpute të veçantë, sipas mënyrës që është trajtuar. Nëse stërvitja ka shkuar mirë, duhet të presim një rezultat të saktë, me kuptim për çdo input. Perceptroni duhet të jetë i aftë të përcaktojë një output të arsyeshëm edhe nëse nuk ka parë një bashkësi vlerash inputi të veçantë më parë.

Perceptronet stërviten për të përdorur si algoritme të mësimit të mbikqyrur ashtu edhe të mësimit jo të mbikqyrur. Mësimi i mbikqyrur merr parasysh njohuritë e mëparshme të cilat i jepen rrjetit gjatë fazës së stërvitjes. Gjatë kohës së mësimit i thuhet rrjetit nëse rezultati i tij përfundimtar është i saktë, nëse jo ndryshohen peshat input për të prodhuar rezultatin e dëshiruar. Në mësimin jo të mbikqyrur nuk jepet rezultati korrekt i rrjetit gjatë stërvitjes. Rrjeti përshtatet vetëm në bazë të përgjigjes ndaj inputeve, ku mëson të njohë modelet dhe strukturat në bashkësitë e të dhënave.

Hipoteza e paraqitur nga një rrjet nervor i stërvitur përcaktohet nga topologjia e rrjetit, funksionet e transferimit për njësitë e daljes dhe parametrat me vlera reale që i shoqërohen lidhjeve të rrjetit (peshat). Një rrjet i zakonshëm mund të përmbajë dhe qindra apo mijëra parametra të tillë dhe në rrjetat me disa shtresa këto parametra mund të përfaqësojnë lidhje jolineare apo jo monotone. Ndaj është e vështirë të përcaktohet efekti i një elementi në vlerën përfundimtare në mënyrë të izoluar sepse ky efekt mund të ndikohet nga vlerat e elementëve të tjerë. Proçesi i zbatimit të rrjetave nervore në DM kalon në tre faza:

Faza 1 - Ndërtimi i rrjetit dhe stërvitja
Kjo fazë ndërton një rrjet me x shtresa bazuar në numrin e atributeve, numrin e klasave dhe në metodën e zgjedhur të kodimit të inputit.
Faza 2 - Thjeshtimi i rrjetit
Kjo fazë synon të eliminojë lidhje dhe njësi të panevojshme pa ndikuar në gabimin e klasifikimit të rrjetit. Pjesa e mbetur e njësive dhe lidhjeve mundëson nxjerrjen e rregullave të saktë dhe të kuptueshëm.
Faza 3 - Nxjerrja e rregullit
Kjo fazë nxjerr rregullat e klasifikimt nga rrjeti. Këto rregulla janë të formës:

$$\text{“nëse } (a_1\ \theta\ v_1) \text{ dhe } (x_2\ \theta\ v_2) \ldots \text{dhe } (x_n\ \theta\ v_n) \text{ atëherë } C_j\text{”} \qquad (3.17)$$

Ku:
a_i - janë atributet
v_i - janë konstantet
θ - janë operatorët relacionalë ($=, \leq, \geq$)
C_j - është një etiketë klase

Nxjerrja e Rregullit

Vështirësia e nxjerrjes së rregullave qëndron në lidhjet e shumta (edhe me një rrjet ku lidhjet e panevojshme janë eliminuar), që e bëjnë të vështirë shprehjen e një

lidhjeje mes një inputi dhe etiketës së klasës përkatëse në formën e rregullave "if...then". Nëse një rrjet ka n - lidhje në hyrje me vlera binare mund të kemi 2^n modele inputi. Gjithashtu rregullat mund të jenë të gjata dhe komplekse edhe për një numër n të vogël. Nga ana tjetër, vlerat kufi të një njësie të fshehur mund të jenë numër çfarëdo në intervalin]-1,1]. Është e vështirë të nxirret një rregull mes vlerave kufi të vazhdueshme të njësive të fshehura dhe vlerave në dalje të një njësie në shtresën dalëse. Rregullat që nxirren japin kushtet që elementët në hyrje duhet të përmbushin për të dhënë një output të vlefshëm.

$$a_y=\begin{cases} 1 \text{ nese } \sum_i p_i a_i+\theta>0 \\ 0 \quad \text{tjeter} \end{cases} \tag{3.18}$$

ku:

a_y - aktivizimi i daljes dhe a_i - është aktivizimi i njësisë së i-të

p_i - pesha e njësisë së i-të

θ - vlera kufi e njësisë së daljes

X_i i referohet vlerës së elementit të i-të dhe a_i- aktivizimit të elementit hyrës përkatës, pra nëse x_i=true atëherë a_i=1.

Nxjerrja e rregullave nga rrjetat nervore përbëhet nga katër hapa:

Hapi 1: Zbatohet një algoritëm grupimi për të gjetur grupime të vlerave kufi të njësive të fshehura. Në këtë hap algoritmi grupon vlerat kufi të njësive të fshehura në një numër të caktuar vlerash diskrete pa sakrifikuar shkallën e saktësisë së rrjetit. Pas grupimit kemi disa vlera aktivizimi në secilën nyje të fshehur.

Hapi 2: Diskretizohen këto vlera kufi dhe llogariten daljet e rrjetit. Gjenerohen rregulla që përshkruajnë daljet e rrjetit sipas vlerave diskrete kufi të njësisë së fshehur. Këto vlera të diskretizuara me vlerat kufi të shtresës dalëse, shoqërohen me etiketat e klasave.

Hapi 3: Për çdo njësi të fshehur, emërohen vlerat e hyrjes që të sjellin tek to dhe gjenerohen rregulla për të përshkruar vlerat kufi të njësive të fshehura sipas hyrjeve.

Hapi 4: Bashkohen këto dy bashkësi rregullash për të përftuar rregulla që lidhin hyrjet dhe daljet.

Përdorimi i rrjetave nervore në sigurimin e jetës ka avantazhet si: (i) ato janë të pandjeshme ndaj zhurmave; (ii) rregullat e klasifikimit e nxjerrë nga rrjetat nervore kanë një nivel gabimi të krahasueshëm me pemët e vendimit.

Gjithashtu rrjetat nervore kanë edhe disavantazhet e tyre në sigurime si [44]: (i) ato mund të mësojnë rregullat e klasifikimit nëpërmjet shumë hapave mbi bashkësinë e të dhënave ndaj koha e mësimit është e gjatë; (ii) një rrjet nervor është një graf me

disa shtresa, ku dalja e një nyje mund të ndahet në një ose disa nyje të tjera në shtresën tjetër. Proçesi i klasifikimit bazohet në strukturën e grafit dhe në peshat që i shoqërohen lidhjeve mes nyjeve, si rrjedhojë artikulimi i rregullave të klasifikimit bëhet i vështirë.
Ky algoritëm është zhvilluar dhe aplikuar në një bazë të dhënash në sigurimin e jetës, me qëllim identifikimin e karakteristikave të klientëve që janë të siguruar, nëse janë me risk të lartë ose të ulët dhe për t'i përdorur më pas këto karakteristika tek klientët e rinj.

3.6 KRITERET E VLERËSIMIT TË MODELEVE TË KLASIFIKIMIT

Pasi ndërtojmë një model klasifikimi, duhet të kryejmë një vlerësim të saj për të parë saktësinë e tij. Për të kryer ndarjen e etiketuar të të dhënave në modelet e klasifikimit kemi përdorur metodën mbështetëse dhe metodën e vlerësimit kryq të k-fishtë.
Ndërsa për vlerësimin e një modeli klasifikimi janë llogaritur dhe vlerësuar kriteret si saktësia, ndjeshmëria, specifikimi dhe shkalla e gabimit. Zgjedhja e modelit më të mire është kryer duke përdorur analizën e kostove dhe përfitimeve dhe grafikun ROC.

3.6.1 METODAT PËR NDARJE TË ETIKETUARA

Metoda mbështetëse *(holdout method)*

Në këtë metodë të dhënat fillestare ndahen rastësisht në dy grupe të pavarura, në një bashkësi trajnimi dhe në një bashkësi test. Dy të tretat e të dhënave caktohen për bashkësinë e trajnimit, dhe një e treta është ndarë për bashkësinë test. Bashkësia e trajnimit është përdorur për të nxjerrë modelin, ndërsa saktësia e modelit është vlerësuar më pas me bashkësinë test. Vlerësimi me anë të kësaj metode është pesimist për shkak se vetëm një pjesë e të dhënave fillestare është përdorur për të nxjerrin modelin.

Metoda e vlerësimit kryq e k-fishtë *(k-fold cross-validation)*

Kjo metodë i referohet një procedurë të përdorur gjerësisht ne testimin e modeleve, ku baza e të dhënave është e ndarë rastësisht në *k* nenbashkesi, atëherë algoritmi i DM perdor si bashkesi trajnimi *k-1* nënbashkësi dhe nënbashkësinë e mbetur e përdor si bashkesi test per të matur performancën e algoritmit, ky proçes përsëritet k-herë, dhe në fund, matjet e regjistruara mesatarizohen. Zgjedhja e *k=10* është standarte, por mund të ndryshohet në varësi të madhësisë së bazës së të dhënave.

Për një model klasifikimi, vlerësimi saktësia është numri i përgjithshëm i klasifikimeve saktë nga *k - përsëritje*, pjesëtuar me numrin total të tupleve në të dhënat fillestare.

3.6.2 KRITERET E VLERËSIMIT PËR MODELIN

Saktësia është kriteri kryesor për vlerësimin e modelit, i cili është koefiçienti i rasteve të klasifikuara në rregull në përqindje, ose ndryshe njihet si norma e përgjithshme e njohjes së klasifikuesit [45]. Saktësia është rezultat i matricës së dhënë më poshtë (tabela 1) dhe llogaritet nga formula (3.19):

$$Saktësia = TP+TN / (TP + FP + TN + FN) \quad (3.19)$$

Ku:

TP - i referohet tupleve pozitive, të cilat janë etiketuar në mënyrë korrekte nga ana e klasifikuesit.

TN - i referohet tupleve negative, të cilat janë etiketuar në mënyrë korrekte nga ana e klasifikuesit.

FP - i referohet tupleve negative që janë etiketuar gabimisht si pozitive nga ana e klasifikuesit.

FN - i referohet tupleve pozitive që janë etiketuar gabimisht si negative nga ana e klasifikuesit.

		Vlerat e parashikuar nga modeli	
		Klasa 1	Klasa 2
Vlerat Aktuale	Klasa 1	True positive (TP)	False positive (FP)
	Klasa 2	False negative (FN)	True negative (TN)

Tabela 1 Klasifikimi matrica e saktësisë

Ndjeshmëria është koefiçienti pozitiv i njohjes ose përqindja e tupleve pozitive që janë identifikuar siç duhet dhe shprehet me formulën e mëposhtme:

$$Ndjeshmëria = TP / (TP + FN) \quad (3.20)$$

Specifikimi paraqet përqindjen e tupleve negative që janë identifikuar siç duhet dhe shprehet si më poshtë:

$$Specifikimi = TN / (FP + TN) \quad (3.21)$$

Shkalla e gabimit në klasifikimit matet me formulat e meposhtme:

$$T1\_SHGK = FP/N \quad (3.22)$$

$$T2\_SHGK = FN/N \quad (3.23)$$

Analiza e kostos dhe përfitimit (risk and gain) gjithashtu përdoret për të llogaritur saktësinë e një model klasifikimi [46].

Grafiku ROC eshte një mjet i dobishëm vizual për të krahasuar dy modele klasifikimi, i cili është një paraqitje grafike e marrëdhënies midis normës së njohjes së klasifikuesit për tuplet pozitive dhe negative [47].
Për të vlerësuar saktësinë e një modeli, masim zonën nën kurbë dhe një model me saktësi shumë të mirë ka sipërfaqen e zonës nën kurbë të barabartë me një [48]. Një udhëzues i përafërt për saktësinë e klasifikimin e një modeli paraqitet në tabelën 2, si më poshtë:

Nr	***Vlerat e sipërfaqes së zonës nën kurbë***	***Saktesia e klasifikimit***
1	0.90 - 1.00	I shkëlqyer/shumë i mire
2	0.80 - 0.90	I mire
3	0.70 - 0.80	I drejtë
4	0.60 - 0.70	I dobët
5	0.50 - 0.60	I dështuar

Tabela 2 ROC vlerat e saktësisë së klasifikimit

4 – TEKNIKA E GRUPIMIT

Grupimi i të dhënave (Clustering) është një nga çështjet më të rëndësishme në DM dhe ML. Ai është proçesi i ndarjes të një bashkësie të dhënash në disa nënbashkësi në mënyrë që objektet brënda një nënbashkësie të kenë ngjashmëri të madhe dhe të jenë pak të ngjashëm me objektet në nënbashkësitë e tjera. Shkalla e ngjashmërisë përcaktohet në bazë të vlerave të atributeve që përshkruajnë objektet duke përfshirë edhe matjen e largësisë midis tyre. Gjatë viteve të fundit është shtuar interesi në zhvillimin e algoritmeve të grupimit nga shumë studiues.
Problemi i grupimit në një bashkësi të dhënash është se ne nuk kemi njohuri paraprake në zgjedhjen e parametrave si numri i grupimeve dhe të faktorëve të tjerë në këto algoritme, duke e bërë grupimin nje temë shumë interesante. Çdo zgjedhje e gabuar e këtyre parametrave jep rezultate të gabuara në grupim. Grupimi si një mjet i DM i ka rrënjët në shumë fusha zbatimi si financa, biologjia, siguria, inteligjenca në biznes dhe kërkimi në web.
Në këtë kapitull paraqitet një përshkrim mbi analizën e grupimit, kriteret e saj dhe metodat për krahasim të grupimit. Për metodën e ndarjes kemi zgjedhur algoritmet e k–mesatareve dhe k-medianave, ndërsa për metodën e bazuar në dënduri algoritmin dbscan.
Për algoritmin e k-mesatareve në problemin e sigurimit të jetës propozohet të përdoret metoda e përzgjedhjes së centroidin fillestar të grupimit në vend të metodës së përzgjedhjes së rastësishme që përdoret zakonisht në këto raste. Gjithashtu jepet një vlerësim i detajuar në përmirësimin e performancës së metodës së inicializimit të propozuar mbi shumë bashkësi të dhënash me përmasa të ndryshme, numër të ndryshëm vëzhgimesh dhe kompleksitete të ndryshme grupimi. Efektivitetin e algoritmit të grupimit e kemi vlerësuar nëpërmjet kriterit shuma e katrorëve të gabimit, i cili është i thjeshtë në përdorim. Rezultatet eksperimentale tregojnë se metoda e inicializimit të propozuar është më efektive dhe konvergjon drejt rezultateve më të sakta të grupimit sesa ato të metodës së inicializimit në mënyrë të rastësishme.

4.1 ANALIZA E GRUPIMIT

Analiza e grupimit është proçesi i ndarjes të një bashkësie të dhënash në nënbashkësi. Çdo nënbashkësi është një grupim dhe brënda tyre zbatohet parimi i ngjashmërisë. Bashkësia e grupeve që rrjedh nga analiza e grupimit mund të quhet grupim. Metoda të ndryshme grupimi mund të gjenerojnë grupe të ndryshëm për të

njëjtën bashkësi të dhënash. Gjatë grupimit automatik mund të gjënden edhe grupime me ngjashmëri të panjohura [49].
Si një funksion i DM, analiza e grupimit mund të përdoret si një mjet i veçantë për të vëzhguar shpërndarjen e të dhënave, karakteristikat e grupeve të ndryshëm dhe për t'u përqëndruar në një bashkësi grupesh për t'i analizuar më tej. Gjithashtu mund të shërbejë si një hap parapërpunimi për algoritme të tjerë, si karakterizimi, klasifikimi etj. Për shkak se një grupim është një grumbull objektesh të dhënash që janë të ngjashëm mes tyre brënda grupimit, por jo të ngjashëm me objektet në grupimet e tjera, një grupim objektesh të dhënash mund të trajtohet edhe si një klasë. Në këtë kuptim, grupimi ndonjëherë quhet edhe klasifikim automatik. Gjithashtu ai quhet edhe segmentim të dhënash në disa aplikacione, sepse ai copëzon bashkësi të mëdha të dhënash në grupe sipas ngjashmërisë së tyre. Grupimi përdoret edhe për detektimin e përjashtimeve, ku përjashtimet apo vlerat jashtë rangut, mund të jenë më interesantet. Aplikacione të rëndësishme që shfrytëzojnë këtë cilësi përfshijnë detektimin e mashtrimeve me kartat e kreditit dhe monitorimin e aktiviteteve kriminale në blerjet elektronike [50].
Grupimi është një fushë sfiduese kërkimi, ku potenciali i tij në aplikime të ndryshme paraqet kërkesa për realizimin e grupimit të veçanta [51].
Në vijim janë dhënë shkurtimisht disa kërkesa për realizimin e grupimit në DM:

Shkallëzueshmëria: Aftësia e algoritmave të grupimit për të analizuar të dhënat në bashkesi të vogla dhe të mëdha. Shumë algoritme grupimi funksionojnë mirë në bashkësi të vogla, që përmbajnë më pak se disa qindra objekte të dhënash; megjithatë, një bazë te dhenash e madhe mund të përmbajë miliona apo miliarda objekte. Zbatimi i analizës vetëm në një kampion të dhënash sjell rezultate jo të drejta dhe për këtë arsye nevojiten algoritme grupimi të shkallëzueshëm [52].

Aftësia për të trajtuar tipe të ndryshme atributesh: Shumë algoritme projektohen për të grupuar të dhëna numerike, por shumë aplikacione kërkojnë edhe grupimin e tipeve të tjera të dhënash, si të dhëna binare, tekste apo edhe përzierje të tyre. Gjithnjë e më shumë aplikacionet kanë nevojë për teknika grupimi të cilat trajtojnë tipe komplekse të dhënash si grafe, sekuenca, imazhe apo dokumente [53].

Zbulimi i grupeve me formë arbitrare: Algoritmet të cilat bazohen në disa matje largesie të caktuar priren të gjejnë grupe sferikë me madhësi dhe denduri të përafërt. Por një grupim mund të ketë një formë të çfarëdoshme, ndaj është e rëndësishme zhvillimi i algoritmave që detektojnë edhe grupe me formë të çfarëdoshme [54].

Kërkesa për njohuri të fushës për të përcaktuar parametrat hyrës: Shumë algoritme grupimi kërkojnë nga përdoruesi të japin njohuri mbi fushën në formën e parametrave hyrës (input), për shëmbull numri i dëshiruar i grupimeve dhe për rrjedhojë rezultatet e grupimit mund të jenë të ndjeshëm ndaj parametrave të tillë. Parametrat mund të jetë të vështirë për t'u përcaktuar, sidomos për bashkësi të dhënash shumë përmasore ku përdoruesve i duhet të kuptojnë fillimisht të dhënat. Kërkesa për njohuri të fushës jo vetëm i ngarkon përdoruesit por edhe vështirëson kontrollin e cilësisë së grupimit [55].

Aftësia për të trajtuar të dhënat e zhurmshme: Shumë bashkësi të dhënash përmbajnë vlera të gabuara, të panjohura apo boshe. Algoritmet e grupimit mund të jenë të ndjeshëm ndaj zhurmave të tilla dhe prodhojnë grupime jo shume cilësore, për këtë arsye nevojiten metoda që përballojnë zhurmën.

Grupimi rritës dhe pandjeshmëria ndaj renditjes së të dhënave hyrëse: Në shumë aplikacione të dhëna të reja mund të vijnë vazhdimisht. Shumë algoritme grupimi nuk mund ti përfshijnë këto të dhëna në strukturat e grupimit ekzistuese dhe fillojnë një proçes grupimi nga e para për një bashkësi objektesh të dhënash.

Aftësia për të grupuar të dhëna shumë përmasore: Një bashkësi të dhënash mund të përmbajë shumë përmasa apo atribute. Për shembull kur grupohen dokumentet, çdo fjalë kyçe mund të merret si një përmasë dhe shpesh kemi mijëra fjalë kyçe. Shumë algoritme grupimi janë të përshtatshëm për trajtimin e të dhënave me përmasa te vogla me dy ose tre atribute. Gjetja e grupimeve të objekteve të të dhënave është një sfidë më vete sidomos nëse të dhënat janë shumë përmasore. Më poshtë do të trajtohen disa nga metodat e grupimit më të rëndësishme, të cilat përdoren në sigurime.

4.2 METODAT E NDARJES

Shënojmë me *D*, bashkësinë e të dhënave të *n - objekteve* për t'u grupuar. Një objekt përshkruhet nga *d – ndryshore* (attribute), ndaj mund t'i referohemi si një pikë në një hapësirë *d-përmasore*. Metodat e ndarjes i ndajnë objektet e të dhënave në grupime që përjashtojnë njëri-tjetrin. Pra nëse një objekt bën pjesë në një grupim ai nuk mund të bëjë pjesë në asnjë grupim tjetër [56].
Pika fillestare e metodave të ndarjes është dhënia e numrit të grupimeve. Për një bashkësi të dhënë *D* me *n - objekte* dhe *k – numër grupimi*, një algoritëm grupimi sipas metodës së ndarjes grupon objektet në *k - nënbashkesi (k≤n)*, ku çdo nënbashkësi përfaqëson një grupim.

Grupimet:{C_1,...,C_k} ku: $C_i \subset D$ dhe $C_i \cap C_j = \emptyset$ për ($1 \leq i, j \leq k$) (4.1)

Ndarja në grupime bëhet duke përdoret një funksion objektiv, i cili synon një ngjashmëri të madhe brënda grupimit dhe ngjashmëri të vogël ndërmjet objekteve.
Teknika ndarëse bazohet në pikën qëndrore, përdor qendrën e grupimit c_i, për të përfaqësuar grupimin. Largësia ose distanca midis një objekti $p \in C_i$ dhe përfaqësuesit të grupimit c_i, matet nëpërmjet distancës euklidiane $dist(p,c_i)$.
Cilësia e grupimit C_i, matet nga variacioni brënda grupimit, i cili llogaritet nga shuma e katrorëve të gabimit mes të gjithë objekteve në C_i dhe qendrës c_i:

$$\text{SEC} = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, c_i)^2 \quad (4.2)$$

Ku:
SEC - është shuma e katrorëve të gabimeve për të gjithë objektet në bashkësinë e të dhënave;
p - është pika në hapësirë që përfaqëson një objekt të dhënë $\in C_i$;
c_i - është qendra e grupimit C_i ose përfaqësuesi i grupimit C_i
Pra, për çdo objekt në secilin grupim llogaritet katrori i distancës nga qendra dhe më pas të gjitha vlerat mblidhen. Ky funksion objektiv përpiqet të bëjë grupimet përfundimtare sa më kompakt brenda tyre dhe të ndarë nga njëri-tjetri.
Algoritmat kryesorë që përfaqësojnë metodat e ndarjes janë: algoritmi i k-mesatareve (k-means) dhe algoritmi i k-medianave (k-medoids).

4.2.1 ALGORITMI I K-MESATAREVE

Algoritmi i k-mesatareve përbën një metodë të thjeshtë grupimi dhe synon të ndajë *n – objekte* në *k – grupime*, ku ndarja bazohet në një funksion objektiv të caktuar. Ai është i thjeshtë, i lehtë, i kuptueshëm, i shkallëzueshëm, dhe mund të adaptohet që të veprojë me bashkësi të mëdha të dhënash [57]. Ndarja e një bashkësie të dhënash *D* në *k-grupime* që nuk kanë pika bashkimi, kryhet bazuar në mosngjashmëritë ndërmjet objekteve të të dhënave dhe centroidëve të grupimit.
Ky algoritëm është zbuluar nga disa kërkues për disiplina të ndryshme përfaqësuar nga Lloyd (1957, 1982), Forgey (1965), Friedman, Rubin (1967) dhe McQueen (1967) [58]. Një përshkrim i detajuar për algoritmin e k-mesatareve është dhënë në punimin [59], i cili lidhet me ndryshimet që shoqërojnë madhësitë e marra në shqyrtim për llogaritje të ndryshme dhe mbi gjetjen e hapave të duhur për interpretimin e problemeve në disiplina të ndryshme. Gray dhe Neuhoff në punimin e tyre [60] sigurojnë një paraqitje më të mirë të tipeve të të dhënave për

algoritmin e k-mesatareve, duke e vendosur në një kontekst më të gjërë për bashkesi të mëdha të dhënash.

Si funksionon algoritmi i k-mesatareve?

Ky algoritëm fillimisht përcakton qendrën e grupimit si vlera mesatare e pikave brënda tij. Më pas zgjidhen në mënyrë të çfarëdoshme *k - objekte* në *D*, ku secili përfaqëson një qendër grupimi. Për secilin nga objektet e mbetur, një objekt i caktohet grupimit më të ngjashëm, bazuar në distancën mes objektit dhe qendrës së grupit[61]. Algoritmi i k-mesatareve përmirëson iterativisht variacionin brenda grupimit. Për çdo grupim, llogaritet mesatarja e re, duke përdorur objektet që i janë shtuar grupimit në ciklin e mëparshëm. Më pas të gjithë objektet ricaktohen duke përdorur mesataret e reja si qendra të reja për grupimet. Ciklet vazhdojnë deri sa grupimet e formuara në atë cikël janë të njëjta me ato të formuara në ciklin paraardhës.

Algoritmi i k - mesatareve
Të dhëna: *k* - numri i grupimeve;
D - bashkësia e të dhënave që përmban *n* – objekte
Metoda:
(1) përzgjidhen në mënyrë të çfarëdoshme k - objekte nga D, si qendra fillestare grupimi
(2) përsërit
(3) (ri)cakto çdo objekt ne grupin më të ngjashëm me objektin, bazuar në vlerën mesatare të objekteve në grupim;
(4) rillogarit qendrën e grupit për çdo grupim;
(5) derisa s'ka më ndryshim;
Rezultati: Një bashkësi me k – grupime

Figura 8 Algoritmi i k-mesatareve metoda standarte

Për të marrë rezultate të mira, algoritmi i k-mesatareve mund të ekzekutohet disa herë me qendra grupi të ndryshme. Kompleksiteti i algoritmit te k-mesatareve është *O(nkt)*, ku *n* - është numri i objekteve, *k* - është numri i grupimeve dhe *t* - është numri i përsëritjeve. Normalisht, $k<<n$ dhe $t<<n$, ndaj metoda është relativisht e shkallëzueshme dhe efiçiente në proçesimin e bashkësive të mëdha të të dhënave.

Në rastet kur kemi të dhëna me atribute tekst përdoret një variant tjetër i k-mesatareve, i cili është algoritmi i k-modeve, ku vlera mesatare zëvendësohet nga moda. Të dy këto variante mund të kombinohen në rastet kur kemi të dhëna me vlera numerike dhe tekst.

Algoritmi i k-mesatareve është një prej algoritmave të grupimit më i mirë [62].për shkak se:

- Kohët e fundit është zgjedhur dhe listuar ndër 10 algoritmat më ndikues në përpunimin e të dhënave.
- Në të njëjtën kohë është shumë i thjeshtë dhe relativisht i shkallëzueshëm, pasi ai ka kohë ekzekutimi asimptotikisht lineare kundrejt secilit variabël të problemit.

Algoritmit i k-mesatareve paraqet edhe disa kufizime, të cilat janë si më poshtë:

- *Shkallëzueshmëria:* Ai shkallëzohet shumë pak në mënyrë të llogaritur.
- *Kuptimi fillestar:* Rezultati i grupimit është ekstremisht shumë i ndjeshëm ndaj vleres fillestare.
- *Zhurma:* Zhurma ose ngacmimet frenojnë ose prishin cilësinë e rezultatit të grupimit.
- *Numri i Grupimeve:* Numri i grupimeve duhet të përcaktohet në fillim.
- *Minimumi lokal*: Ai gjithmonë konvergjon drejt minimumit lokal.
- *Paaftësia* për të grumbulluar bashkësi të dhënash të ndara në mënyrë jo-lineare.

4.2.2 PUNIME TË NGJASHME NË PËRMIRËSIMIN E PERFORMANCËS

Kërkues të ndryshëm kanë propozuar disa metoda, të cilat zvogëlojnë ose minimizojnë ndjeshmërinë ndaj zhurmave, rrisin saktësinë e algoritmit e k – mesatareve duke përzgjedhur vendndodhjen më të mirë të centroidit brënda bashkësisë së të dhënave ekzistuese.

- Në 2007, David Arthur dhe Sergei Vassilvitski ne kërkimin e titulluar "Algoritmi i k-mesatareve++: avantazhet e futjes së kujdesshme", ata propozuan një mënyrë specifike të zgjedhjes së centroidëve fillestarë. Në kërkimin e tyre, centroidët fillestarë janë zgjidhur rresht me probabilitet propocional me distancën e centroidit më të afërt. Ata prezantuan një mënyrë të re për të futur algoritmin e k-mesatareve si *O(logk)* konkurrues me grupimin optimal [63]. Më poshtë është dhënë algoritmi i k-mesatareve++:

    1. *Zgjidhet një centroid inicializues c_1=x, në mënyrë të rastësishme nga X.*
    2. *Vendos D(x) si distancë Euklidiane më të shkurtër nga një pikë e dhënë x me centroidin më të afërt.*
    3. *Zgjidhet një centroidi tjetër c_i, duke përzgjedhur $c_i = x' \in X$ me probabilitet*
    $$\frac{D(x')^2}{\sum D(x)^2}$$
    4. *Përsëriten hapat 2 dhe 3 derisa të kemi zgjedhur një total prej k- centroidesh.*
    5. *Veprojmë si me algoritmin standart të k-mesatareve.*

- Ndërsa algoritmat inicializues për optimizimin e ndarjes kanë propozuar një rrugë me hapa për të specifikuar vlerat inicializuese duke gjetur fillimisht një numër të madh grupimesh dhe më pas kryhet reduktimi i tyre për të përftuar k-grupimet e dëshiruar [64].

- Efiçenca e algoritmit te k-mesatareve nëpërmjet teknikave evolucionare për të zhvilluar aplikimin janë më efikase gjatë llogaritjes sesa rrugët sistematike që tentojnë të sillen rrotull të metave të tij [65].

4.2.3 METODA E PROPOZUAR PËR PROÇESIN E INICIALIZIMIT

Problemi i përmirësimit të proçesit të përzgjedhjes së centroideve fillestarë ka disa përfitime:
1. Ul sasinë e përllogaritjes;
2. Optimizon performancën e algoritmit;
3. Minimizon funksionin objektiv duke na çuar në rezultate më të mira.
Ky përmirësim kryhet kur centroidet fillestarë janë shumë larg njëri-tjetrit dhe secili prej tyre ndjek një grupim të ndryshëm.
Metoda e re e propozuar për procesin e inicializimit në sigurime fillon duke zgjedhur centroidet fillestare në mënyrë të rastit. Pas proçesit të përzgjedhjes së pikënisjes rastësisht; kryejmë disa llogaritje për të gjetur nëse pika është e përshtatshme për t'u marrë në konsideratë si një centroid i parë fillestar apo jo. Një vendim i tillë e bazojmë në procesin e llogaritjes së distancave ndërmjet centroidit të zgjedhur dhe pikave të tjera brenda bashkësisë së të dhënave. Kemi përdorur njësinë matëse Euklidiane të largësisë për të llogaritur largësinë midis të dhënave.

Supozojmë se numri i objekteve në një grup është i afërt ose i barabartë me numrin e objekteve në grupimet e tjera.

Këtë supozim e bazojmë në faktin se algoritmi i k-mesatareve gjithmonë nxjerr rezultate më të mira me bashkësi të dhënash të cilat janë të ngjashme në densitete dhe të afërt me numrin e objekteve në secilin grup. Si rrjedhim, supozimi i ngritur është i vlefshëm për një numër të madh të bashkësive të të dhënave.
Testojmë pikën e përzgjedhur si centroid nëse është zhurmë ose jo. Më pas, vlerën mesatare të numrit *N - pika më të afërt e centroidit aktual*, e ruajmë si centroidi i parë i pranuar, i cili është injoruar në llogaritjet e mëparshme. Kjo metodë përsëritet derisa identifikohet numri i kërkuar i centroideve fillestarë.
Procesi i llogaritjes së distancave midis pikës së përzgjedhur dhe pikave të mbetura është boshti drejtues i kësaj metode, pasi vlerat e distancës mes centroidit të

zgjedhur dhe pikës më të afërt me të është përdorur për të llogaritur vlerën e ε ' dhe është krahasuar me vlerën e ε, e cila është e barabartë me vlerën mesatare të distancave midis çdo çifti të pikave N. Përcaktojmë numrin e pikave më të afërta me centroidin e zgjedhur në varësi të supozimit të bërë më lart, ku numri është i barabartë me 80% deri në 90 % të numrit të llogaritur nga pjesëtimi i numrit total të objekteve të bashkësisë së të dhënave me numrin e grupimeve të dhëna nga ana e përdoruesit. Nëse pika e parë e zgjedhur është zhurmë, dmth ε ' > ε; kjo pikë injorohet dhe një pikë tjetër zgjidhet rastësisht si centroid fillestar derisa të gjendet centroidi i parë. Pastaj, centroidi i ardhshëm duhet të zgjidhet si pikat më të largëta nga centroidi i parë. Nëse pika e dytë e zgjedhur është zhurmë, ajo injorohet bashkë me pikën më të afërt të saj.

Algoritmi i Propozuar

Le të jetë $X = \{ x_1 , x_2 , .. , x_n \}$ një bashkësi të dhënash me n - objekte, dhe k - parametri hyrës, i cili është i barabartë me numrin e grupimeve.

1. *Zgjedhim një centroid fillestar $c_i = xr$, ku $0 < i \leq k$ dhe xr një pikë rastësore nga X.*
2. *Llogaritim distancën midis centroid të zgjedhur c_i dhe çdo pike në X, dhe pastaj renditim pikat e të dhënave bazuar në distancat rezultuese.*
   $D = d(c_i, p_j)$ ku D : zakonisht është zgjedhur si distanca Euklidiane , $0 < j \leq n$.
3. *Marrim një nënbashkësi të të dhënave të klasifikuara me një numër të pikave të barabartë me N- numri i të dhënave më të afërta me centroidin c_i të zgjedhur, σ është një numër $1 < \sigma \leq 2$.*
4. *Llogaritim distancën mesatare ndërmjet çdo çifti të pikave N*
5. *Nëse {ε' >ε dhe i = 1} ; injoro c_i dhe shko në hapin 1 për të zgjedhur një c_i të ri*
6. *Nëse { ε' >ε dhe i> 1} ; injoro c_i, përzgjidh një c_i të ri me vlerë të barabartë me pikën më të afërt të c_i së mëparshme, dhe shko në hapin 2.*
7. *Zgjidh centroidin e ardhshëm c_{i+1} që të jetë pika më e largët nga c_i .*
8. *Vlera mesatare e N pikave më të afërta të ci është identifikuar si centroid dhe është ruajtur si "centroidi i pranimit C_i".*
9. *Injoro N pikat të cilat janë më të afërta ose më afër c_i.*
10. *Shko tek hapi 2 me vlerën e $c_i = c_i + 1$.*
11. *Përsërit hapat derisa një total prej K - centroidesh janë zgjedhur.*

Figura 9 Algoritmi i k-mesatareve i përmirësuar në proçesin e inicializimit

Testimi i performancën për algoritmin e k-mesatareve në sigurime për bashkësi të dhënash me përmasa të ndryshme paraqitet në kapitujt më poshtë, ku rezultatet eksperimentale krahasohen midis algoritmin standart të k-mesatareve dhe atij të propozuar.

4.2.4 ALGORITMI I K-MEDIANAVE

Algoritmi i k-mesatareve është i ndjeshëm ndaj përjashtimeve, për shkak se ato ndodhen larg pjesës më të madhe të të dhënave dhe në çastin që i caktohen një grupimi, prishin vlerën mesatare të tij. Kjo ndikon edhe në caktimin e objekteve të tjerë nëpër grupime. Për të shmangur këtë problem, në vend që të marrim një vlerë mesatare të objekteve në grupim si pikë referimi, merret një objekt për të përfaqësuar grupimet. Metoda ndarëse zbatohen në bazë të parimit të minimizimit të shumës së jo-ngjashmërive mes çdo objekti p dhe objektit përkatës përfaqësues. Pra përdoret një kriter gabimi absolut që përcaktohet si:

$$SEC = \sum_{i=1}^{k} \sum_{p \in c_i} dist(p, o_i) \qquad (4.3)$$

Ku: *SEC* - është shume e gabimeve absolute për të gjithë objektet *p* në bashkësinë e të dhënave, dhe o_i është objekti përfaqësues i c_i.

Ky është thelbi i metodës i k-medianave (k-medoids), e cila grupon n - objekte në k - grupime duke minimizuar gabimin absolut [66].

Algoritmi PAM (Partitioning Around Medoids) është një zbatim i përhapur i metodës së k-medianave. Ashtu si në algoritmin e k-mesatareve, objektet fillestarë përfaqësues zgjidhen në mënyrë të rastësishme. Merret parasysh nëse zëvendësimi i një objekti përfaqësues me një objekt jo-përfaqësues do të rrisë cilësinë e grupimit dhe provohen të gjithë zëvendësimet e mundshme. Proçesi iterativ i zëvendësimit të objekteve vazhdon derisa cilësia e grupimit rezultant nuk përmirësohet nga ndonjë zëvendësim. Kjo cilësi matet nga një funksion i mesatares së jo-ngjashmërisë mes një objekti dhe përfaqësuesit të grupit ku ai ndodhet [67].

Le të kemi, $o_1,...,o_k$ objektet përfaqësues të grupimeve. Për të përcaktuar nëse një objekt jo-përfaqësues, i shënuar o_{cf}, është një zëvendësues i mirë për përfaqësuesin aktual o_j $(1 \leq j \leq k)$, llogarisim distancën nga çdo objekt *p* në objektin më të afërt nga bashkësia $\{o_1,..,o_{j-1},o_{cf},o_{j+1},...,o_k\}$ dhe përdorim distancën për të ndryshuar funksionin. Ricaktimi i objekteve në $\{o_1,..,o_{j-1},o_{cf},o_{j+1},...,o_k\}$ është i thjeshtë. Supozojmë se objekti *p* i caktohet një grupimi që përfaqësohet nga o_j. Objekti *p* i ricaktohet o_{cf} ose ndonjë grupimi tjetër që përfaqësohet nga o_i $(i \neq j)$ dhe që është më afër tij.

***Algoritmi**: k-medianave. PAM, një algoritëm i k-medianave i bazuar në objekte qendrorë.*
***Të dhënat:** k - numri i grupimeve; D - bashkësia e të dhënave që përmban n- objekte*
Metoda:
(1) zgjidhen në mënyrë të çfarëdoshme k objekte në D si objekte fillestarë përfaqësues;
(2) përsërit, cakto çdo objekt të mbetur në grupimin me objektin përfaqësues më të afërt;
(3) zgjidh në mënyrë të çfarëdoshme një objekt jo përfaqësues, o_{cf};

(4) Llogariten kostot totale, S, të këmbimit të objekteve përfaqësues, o_j me o_{cf};
(5) Nëse S<0 atëherë këmbe o_j me o_{cf} për të formuar bashkësinë e re të k objekteve të rinj përfaqësues; derisa s'ka më ndryshim;
Rezultati: *Një bashkësi me k - grupime.*

Figura 10 Algoritmi i k-medianave PAM i bazuar në objekte qendrorë

Algoritmi i k-medianave është më pak i ndjeshëm se algoritmi i k-mesatareve në rastet e zhurmave dhe përjashtimeve, sepse një objekt përfaqësues ndikohet më pak nga përjashtimet apo vlerat e tjera ekstreme sesa mesatarja. Megjithatë kompleksiteti i çdo iteracioni në algoritmin e k-medianave është $O(k(n-k)^2)$. Për vlera të mëdha të *n* dhe *k*, një llogaritje e tillë bëhet edhe me e kushtueshme se algoritmi i k-mesatareve.

Algoritmi PAM funksionon mirë për bashkësi të vogla të dhënash, për bashkësi të mëdha të dhënash përdoret një metodë tjetër e quajtur CLARA (Clustering LARge Applications). Kjo metodë përdor një kampion të çfarëdoshëm nga bashkësia e të dhënave. Pas kësaj përdoret algoritmi PAM për të llogaritur përfaqësuesit më të mirë nga kampioni. CLARA ndërton modele grupimi nga shumë kampionë të çfarëdoshëm dhe kthen grupimin më të mirë si output. Kompleksiteti i llogaritjes së përfaqësuesve në një kampion çfarëdo është $O(ks^2+k(n-k))$, ku s - është madhësia e kampionit, k - është numri i grupimeve dhe n - numri total i objekteve. Ndërkohë që PAM kërkon për përfaqësuesit më të mirë në një bashkësi të dhënë të dhënash, ndërkohë që CLARA kërkon për përfaqësuesit më të mirë në një kampion të zgjedhur të bashkësisë së të dhënave [68].

4.3 METODAT HIERARKIKE

Metodat hierarkike grupojnë objektet e të dhënave në grupime hierarkikë apo "pemë" grupimesh. Një metodë hierarkike grupimi mund të jetë aglomerative apo ndarëse. Një metodë grupimi aglomerative përdor një strategji nga poshtë lart. Fillimisht çdo objekt krijon një grupim të vetin dhe në mënyrë iterative bashkohen grupet në grupime gjithnjë e më të mëdhenj deri sa të gjithë objektet janë në një grupim të madh apo takohet një kusht përfundimi, grupimi i vetëm bëhet rrënja e hierarkisë. Për hapin e bashkimit, gjen dy grupimet që janë më afër njëri tjetrit për të formuar një grupim të vetëm. Duke qenë se në çdo iteracion bashkohen dy grupime dhe çdo grupim ka të paktën një objekt, metoda aglomerative kërkon maksimumin n-cikle [69].
Metoda e grupimit hierarkike ndarëse zbaton një strategji nga lart poshtë. Fillimisht të gjithë objektevt vendosen në një grupim, i cili është rrënja e

hierarkisë. Më pas ndahet rrënja në nën-grupe më të vegjël dhe në mënyrë rekursive këto grupe ndahen në grupe më të vegjël e kështu me rradhë deri kur një grup përmban një objekt të vetëm apo takohet një kusht përfundimi. Si në metodën aglomerative ashtu edhe në atë ndarëse përdoruesi mund të caktojë numrin e grupeve si kusht përfundimi.
Një metodë grupimi hierarkike është AGNES (Aglomerative Nesting), ndërsa DIANA (Divisive Analysis) është një metodë grupimi hierarkike ndarëse.
Një strukturë pemë e quajtur dendrogram përdoret për të përfaqësuar proçesin e grupimit hierarkik. Kjo strukturë tregon si grupohen objektet së bashku (në metodën aglomerative) apo si ndahen nga njëri-tjetri (në metodën ndarëse) hap pas hapi.
Për arsye të eficiencës, metodat ndarëse nuk i rikthehen vendimeve të marra mbi ndarjen e grupeve. Pasi një grupim ndahet nuk merren parasysh alternativa të tjera ndarjeje. Për shkak të sfidave të metodave ndarëse ka më shumë metoda aglomerative sesa ndarëse.

4.4 METODAT E BAZUARA NË DENDËSI

Metodat ndarëse dhe hierarkike janë projektuar për të gjetur grupime në formë sferike dhe e kanë të vështirë të gjejnë grupime të formave të çfarëdoshme. Për të gjetur grupime të tillë, mund të përdoren modele që i tregojnë grupimet si rajone me dendësi të ndryshueshme. Kjo teknikë përdoret në metodat grupimi të bazuara në dendësi. Disa nga teknikat më të rëndësishme të saj janë DBSCAN, OPTICS dhe DENCLUE [70].

Algortimi DBSCAN

Dendësia e një objekti *o* mund të matet nga numri i objekteve afër këtij objekti. Algoritmi gjen objektet që kanë zona të dendura përreth dhe më pas bashkon këto objekte bërthamë dhe rrethinat e tyre për të formuar rajone të dendura si grupim. Një parameter i caktuar nga përdoruesi *eps* me vlerë më të madhe se zero përdoret për të specifikuar rrezen e rrethinës që merret në konsideratë për çdo objekt. Rrethina *eps* e një objekti *o* është hapësira brenda një rrezeje *eps* me qendër *o*. Dendësia e rrethinës mund të matet nga numri i objekteve që ndodhen në të. Për të përcaktuar nëse një rrethinë është apo jo e dendur shihet numri i objekteve në atë rrethinë dhe përdoret një parametër i dhënë nga përdoruesi *MinEl*, cili specifikon vlerën minimale të dendësisë [71].

Një objekt është objekt bërthamë nëse në rrethinën me rreze *eps*, ndodhen të paktën *MinEl* objekte. Për një bashkësi të dhënash *D*, mund të gjejmë të gjithë objektet bërthame me anë të këtyre dy parametrave. Ndaj thjesht përdoren objektet bërthamë dhe rrethinat e tyre për të formuar rajonet e dendura-grupimesh. Për një objekt bërthamë *q* dhe një objekt *p* thuhet se: *p* është dendësisht i arritshëm nga *q*, nëse *p* ndodhen në rrethinën me rreze *eps* të *q*. Për të lidhur objektet bërthamë në një zonë të dendur algoritmi përdor lidhshmërinë e dendësisë. Dy objekte p_1 dhe $p_2 \in D$ janë të lidhura me njëra-tjetrën në lidhje me parmetrat *eps* dhe *MinEl* nëse kemi një objekt $q \in D$, i tillë që p_1 dhe p_2 të jetë të arritshme nga *q* në lidhje me parametrat *eps* dhe *MinEl*.

Proçesi i grupimit nëpërmjet teknikës DBSCAN

Fillimisht të gjithë objektet e bashkësisë *D* të të dhënave shënjohet "të pavizituar". Algoritmi në mënyrë të çfarëdoshme zgjedh një objekt *p*, e shenjon si "të vizituar" dhe sheh nëse rrethina me rreze *eps* e *p* përmban të paktën *MinEl* objekte. Nëse jo, *p* shenjohet si zhurmë. Përndryshe, krijohet një grupim *C* për *p* dhe të gjithë objektet në rrethinën me rreze *eps* me qendër *p* i shtohen bashkësisë së kandidatëve, *N*. Algoritmi shton iterativisht në C objektet e N që nuk bëjnë pjesë në asnjë grupim. Nëse rrethina e këtij objekti ka minimalisht *MinEl* objekte, këto objekte i shtohen bashkësisë *N*. Ai vazhdon t'i shtojë objekte *C*, derisa ai nuk mund të zgjerohet më, pra bashkësia *N* është bosh dhe grupimi *C* është i plotësuar. Nëse përdorim një indeks hapësinor, kompleksiteti i algoritmit është *O(nlogn)*, ku n - numri i objekteve të bazës së të dhënave, përdryshe kompleksiteti është $O(n^2)$.

5 – ANALIZA E SHOQËRIMIT

Modelet e shpeshtë janë modele (bashkësi objektesh, nënsekuenca, apo nënstruktura) që shfaqen shpesh në një bashkësi të dhënash. Për shembull një kombinim si sigurim jete dhe sigurim shëndeti, që shfaqen shpesh së bashku në një bashkësi të dhënash është një kombinim i shpeshtë. Një nënsekuencë, si psh blerja në fillim e një sigurimi për makinën, më pas të një sigurimi jete dhe më pas një sigurimi shëndeti, nëse shfaqen shpesh në bazën e të dhënave të blerjeve është një model i shpeshtë sekuencial. Një nënstrukturë i referohet formave të ndryshme strukturore, si nëngraf, nënpemë apo nënshtresë, e cila mund të kombinohet me bashkësi objektesh apo nënsekuencash.
Një shoqërim është një rregull i tipit nëse *X* atëherë *Y*. Gjetja e modeleve të shpeshtë luan një rol thelbësor në gërmimet e shoqërimeve, korrelimeve dhe shumë lidhjeve të tjera interesante mes të dhënave në sigurime. Gjithashtu ndihmon në klasifikimin, regresionin, grupimin e të dhënave dhe detyrave të tjera të DM [72].
Në këtë kapitull do të paraqesim një vështrim mbi analizën e shportës së tregut, rregullat e shoqërimit, konceptet e bashkësisë së shpeshtë të mbyllur, bashkësisë së shpeshtë maksimale dhe kriteret e vlerësimit si mbeshtetja, besueshmëria dhe korrelacioni. Gjithashtu do të hulumtojmë algoritmat ekzistues të analizës së shoqërimit dhe do propozojmë një përmirësim të algoritmit përparësor duke rritur performancën e tij në sigurime nëpërmjet metodës së rritjes së modeleve të shpeshtë. Metoda përshtat një ndarje duke dhënë strategjinë për vendosjen e të dhënave që përfaqësojnë termat e shpeshtë brënda një strukture, e cila përmban të gjithë informacionin kryesor.

5.1 ANALIZA E SHPORTËS SË TREGUT

Me rritjen e të dhënave të mbledhura në industrinë e sigurimit të jetës, është rritur interesi për t'i analizuar këto të dhëna për të nxjerrë përfundime, të cilat ndikojnë në vendimet menaxheriale. Analiza e shportës së tregut shërben për të parë më nga afër zakonet e blerjes së klientëve duke gjetur shoqërime apo lidhje mes objekteve të ndryshëm që klientët vendosin në shportë [73].
Zbulimi i këtyre lidhjeve ndihmon kompanitë e sigurimit të jetës të zhvillojnë strategji marketingu duke parë se cilët produkte blihen shpesh së bashku. Nëse marrim parasysh tërësinë e produkteve që janë nëpër shporta, secilit mund t'i shoqërojmë një ndryshore buleane që përfaqëson praninë apo mungesën e një objekti. Çdo shportë mund të përfaqësohet nga një vektor bulean vlerash që i janë shoqëruar këtyre ndryshoreve. Këto vektorë mund të analizohen për modele

blerjesh që pasqyrojnë objektet që lidhen apo blihen shpesh së bashku dhe mund të paraqiten në formën e rregullave të shoqërimit.

5.2 BASHKËSITË E SHPESHTA, RREGULLAT E SHOQËRIMIT

Le të kemi një bashkësi objektesh $L=\{I_1,I_2,....,I_m\}$. Shënojmë me D, të gjithë të dhënat që do të merren në shqyrtim, një bashkësi transaksionesh në bazën e të dhënave, ku çdo transaksion T është një bashkësi jo boshe e tillë që $T \subseteq L$ dhe i shoqërohet një identifikator. Le të jetë A, B dy bashkësi objektesh dhe një transaksion T thuhet se përmban A nëse $A \subseteq T$.
Një rregull shoqërimi është një implikim i formës $A \Rightarrow B$, ku $A \subset L$, $B \subset L$, $A \neq \emptyset$, $B \neq \emptyset$ dhe $A \cap B = \emptyset$.
Rregulli $A \Rightarrow B$ ka në bashkësinë D të transaksionit **mbështetje s**, ku **s** është përqindja e transaksioneve në D që përmbajnë $A \cup B$. Ky merret si probabiliteti $P(A \cup B)$, i cili është probabiliteti që një transaksion të përmbajë ose A ose B:

$$s=mbështetja\ (A \Rightarrow B) = P(A \cup B) \qquad (5.1)$$

Rregulli $A \Rightarrow B$ ka një **besueshmëri c** në bashkësinë D, ku ***c*** është përqindja e transaksioneve në D që përmbajnë A dhe B. Ky merret si si probabiliteti P(B|A).

$$c=besueshmëri\ (A \Rightarrow B) = P(B|A) \qquad (5.2)$$

Rregullat që kënaqin një vlerë minimale mbështetjeje *min(s)* dhe një një vlerë minimale besueshmërie *min(c)* quhen të forta [74].
Shpeshtësia e shfaqjes së një bashkësie objektesh është *numri* i transaksioneve që përmbajnë një bashkësi objektesh. Kjo njihet thjesht si *frekuenca*, apo *sasia* e bashkësisë së objekteve. Mbështetja e përcaktuar më sipër njihet shpesh si mbështetja relative, ndërkohë që shpeshtësia e shfaqjes quhet *mbështetja absolute*. Nëse mbështetja relative e një bashkësie objektesh L kënaq një *vlerë mbështetje minimale* të paracaktuar, atëherë L është një bashkësi e shpeshtë objektesh.

$$Besueshmëri\ (A \Rightarrow B)=P(B|A)=\frac{mbështetja\ (A \cup B)}{mbështetja\ (A)}=\frac{sasia\ e\ mbështetjes\ (A \cup B)}{sasia\ e\ mbështetjes(A)} \qquad (5.3)$$

Ky ekuacion tregon që besueshmëria e rregullit $A \Rightarrow B$ mund të derivohet nga sasia e mbështetjes së A dhe $A \cup B$. Pra me gjetjen e sasisë së mbështetjes së A, B dhe $A \cup B$ mund të derivohen direkt rregullat e shoqërimit $A \Rightarrow B$ dhe $B \Rightarrow A$ dhe të shihet nëse janë të forta.

Në këtë mënyrë problemi i gërmimit të rregullave të shoqërimit mund të reduktohet në gërmimin e bashkësive të shpeshta objektesh. Në përgjithësi, gërmimi për rregulla shoqërimi mund të shihet si një proçes dy hapësh:

Hapi_1: Gjetja e të gjitha bashkësive të shpeshta të objekteve, ku secili nga bashkësitë do të shfaqet të paktën aq shpesh sa është vlera minimale e mbështetjes së paracaktuar *min(s).*

Hapi_2: Gjenerimi i rregullave të shoqërimit të forta nga bashkësitë e shpeshta të objekteve. Rregullat e gjeneruara duhet të kënaqin një mbështetje minimale *min(s)* dhe një besueshmëri minimale *min(c).* Për shkak se hapi i dytë është shumë më pak i kushtueshëm se i pari, performanca e përgjithshme e gërmimit për rregulla shoqërimi përcaktohet nga hapi i parë.

5.3 BASHKËSITË E SHPESHTA TË MBYLLURA DHE MAKSIMALE

Në gërmimin e bashkësive të shpeshta të objekteve nga një bashkësi e madhe të dhënash, shpesh gjenerohet një numër shumë të madh bashkësish që kënaqin mbeshtetjen *min(s).* Për të kapërcyer këtë vështirësi, jepen konceptet e bashkësisë së shpeshtë të mbyllur dhe bashkësisë së shpeshtë maksimale.

Një bashkësi *X* është e mbyllur në një bashkësi *D* nëse nuk ka një superbashkësi *Y* (*Y* është një superbashkësi e *X*, nëse *X* është një nënbashkësi e *Y*, pra $X \subset Y$. Pra çdo objekt i *X* bën pjesë në *Y*, por ka të paktën një objekt të *Y* që nuk bën pjesë në *X*), e tillë që *Y* ka të njëjtën sasi mbështetje me *X* në *D*. Një bashkësi *X* është një bashkësi e shpeshtë e mbyllur në bashkësinë *D*, nëse *X* është njëkohësisht e mbyllur dhe e shpeshtë në *D*.
Një bashkësi *X* është një bashkësi e shpeshtë maksimale në bashkësinë *D*, nëse *X* është e shpeshtë dhe nuk ka asnjë superbashkësi *Y* e tillë që $X \subset Y$ dhe *Y* është i shpeshtë në *D* [75].
Le të jetë *C* bashkësia e shpeshtë e mbyllur për një bashkësi të dhënash *D* që kënaq një vlerë minimale mbështetje *min(s).* Le të jetë *M* bashkësia e shpeshtë maksimale në *D* që kënaq *min(s).* Supozojmë që kemi sasinë e mbështetjes për çdo objekt në *C* dhe *M*. Duhet pasur parasysh që bashkësia *C* dhe informacioni mbi sasinë e mbështetjes mund të përdoren për të derivuar të gjithë bashkësinë e bashkësive të shpeshta. Në këtë mënyrë themi se *C* përmban informacion të plotë sa i përket bashkësive të shpeshta të veta. Nga ana tjetër, *M* mban vetëm mbështetjen për bashkësitë maksimale dhe jo të gjithë informacionin për mbështetjen e të gjitha bashkësive të veta [76].

5.4 VLERËSIMI I MODELEVE, RREGULLAT E FORTA

Kriteret mbështetje dhe besueshmëri janë dy njësi matje të interesit që paraqet një rregull dhe përfaqësojnë dobinë e rregullave të zbuluara. Shumë algoritme për gërmimin e rregullave të shoqërimit përdorin një mjedis të bazuar në mbështetje dhe besueshmëri, ku vlerat minimale të tyre ndihmojnë të pastrohen një pjesë e mirë rregullash jo interesante, të cilat ende nuk përbëjnë interes për përdoruesit. Kjo dukuri ndodh sidomos në rastet kur vlerat minimale të besueshmërisë dhe mbështetjes janë shumë të ulta [77].

Sasia e interesit që paraqet një rregull mund të gjykohet objektivisht apo subjektivisht. Është përdoruesi ai që ka fjalën e fundit nëse një rregull është apo jo interesante. Ky mbetet sidoqoftë një gjykim subjektiv dhe jo të gjithë mund të vijnë në një përfundim. Matjet objektive të sasisë së interesit që paraqet një rregull, të cilat bazohen në statistika mund të përdoren si një hap drejt pastrimit të rregullave jo interesante që mund t'i ishin paraqitur përdoruesit [78].

Nga analiza e shoqërimeve në analizën e korrelimeve mbështetja dhe besueshmëria mund të mos jenë të mjaftueshme për të filtruar rregullat e shoqërimit jo interesante. Për të kapërcyer këtë dobësi, përdorim koefiçientë korrelacioni, te cilat sjellin rregulla korrelacioni ne formen si më poshtë:

$$A \Rightarrow B \ [mbështetje, besueshmëri, korrelacion] \tag{5.4}$$

Kjo do të thotë se një rregull shoqerimi nuk matet vetëm nëpërmjet mbështetjes dhe besueshmërisë por edhe nëpërmjet korrelacionit mes bashkësive A dhe B. Ekzistojnë disa koefiçiente për matjen e korrelacionit, ndër të cilat mund te përmëndim:

Koefiçienti Lift

Shfaqja e bashkësisë A është e pavarur nga shfaqja e bashkësisë B nëse $P(A \cup B)=P(A)P(B)$; përndryshe bashkësitë A dhe B janë ngjarje të ndërvarura dhe të korreluara.

$$Lift\ (A,B)=\frac{P(A \cup B)}{P(A)P(B)} \tag{5.5}$$

Përkufizimi mund të zgjerohet edhe për më shumë se dy bashkësi. Nëse vlera që del nga ekuacioni i mësipërm është <1, atëherë shfaqja e *A* është e korreluar negativisht me shfaqjet e *B*, çka do të thotë se shfaqja e një bashkësie mund të sjellë mungesën e tjetrës. Nëse vlera rezultante është >1, atëherë *A* dhe *B* janë të korreluara pozitivisht, që do të thotë se shfaqja e njërës bashkësi implikon edhe

shfaqjen e tjetrës. Nëse vlera=1, atëherë A dhe B janë të pavarura dhe nuk ka korrelacion mes tyre. Gjithashtu ai është ekuivalent me *P(B|A)/P(B)* apo *besueshmëria(A⇒B)/mbështetjen(B)*, i cili quhet lift-i i rregullit të shoqërimit *A⇒B*. Pra me fjalë të tjera gjykon gradën e rritjes së shfaqeve të njërës bashkësi nga tjetra.

Koefiçienti All_confidence

$$all\_confidence(A,B)=\frac{mbështetja(A\cup B)}{max\{mbështetja(A),mbështetja(B)\}}=min\{P(A|B),P(B|A)\} \qquad (5.6)$$

ku: *max{mbështetja(A),mbështetja(B)}* është maksimumi i mbështetjes i bashkësive *A* dhe *B*. Gjithashtu ky koefiçient është besueshmëria minimale e dy rregullave shoqëruese të lidhura me *A* dhe *B*, pikërisht "*A ⇒B*" dhe "*B ⇒A*".

Koefiçienti max_confidence

$$max\_conf(A,B) = max\{P(A|B),P(B|A)\} \qquad (5.7)$$

Koeficienti max_confidence është besueshmëria maksimale e dy rregullave shoqëruese "*A ⇒B*" dhe "*B ⇒A*".

Koefiçienti Kulczynski

$$Kulc(A,B)=\frac{P(A|B)+P(B|A)}{2} \qquad (5.8)$$

Ky koefiçient shihet si mesatarja e dy matjeve të besueshmërisë, pra mesatarja e dy probabiliteteve të kushtëzuar: probabiliteti të bashkësisë *A* kur jepet bashkësia *B* dhe probabilitetit të bashkësisë *B* kur jepet bashkësia *A*.

Koefiçienti Cosinus

$$cosinus(A,B)=\frac{P(A\cup B)}{\sqrt{P(A)P(B)}}=\frac{mb(A\cup B)}{\sqrt{mb(A)mb(B)}}=\sqrt{P(A|B)P(B|A)} \qquad (5.9)$$

Koefiçienti Cosinus shihet si një koefiçient lift e harmonizuar. Të dy formulat janë të ngjashmë përveç faktit së në cosinus merret rrënja katrore e prodhimit të probabiliteteve të *A* dhe *B*. Ky është një ndryshim i rëndësishëm, sepse duke marrë rrënjën katrore, vlera e cosinus varet nga mbështetjet a *A, B* dhe *A∪B* dhe jo nga numri i përgjithshëm i transaksioneve.
Secili nga koefiçientët varet nga probabilitetet e kushtëzuara të *P(A|B)* dhe *P(B|A)* dhe jo nga numri i përgjithshëm i transaksioneve. Vlerat e tyre janë të përcaktuara në segmentin [0, 1] dhe sa më e madhe të jetë vlera aq më të lidhura janë *A* dhe *B*.

5.5 ALGORITMI PËRPARËSOR

Algoritmi Përparësor (Apriori) është algoritmi thelbësor për gërmimin e bashkësive të shpeshta për rregulla shoqërimi buleane. Emri i algoritmit bazohet në faktin se algoritmi përdor njohuri të mëparshme të cilësive të bashkësive të shpeshta. Ky algoritëm zbaton një qasje iterative, ku bashkësitë me *k* - elementë përdoren për të zbuluar bashkësitë me *(k+1)* elementë [79].
Fillimisht, gjenden bashkësitë e shpeshta me nga një element duke skanuar bazën e të dhënave për të numëruar çdo objekt dhe mbledhur këto objekte që kënaqin mbështetjen minimale. Bashkësia e përftuar shënohet L_1. Më pas L_1 përdoret për të gjetur L_2, pra bashkësinë e bashkësive të shpeshta me 2 elementë, e cila përdoret për të gjetur L_3 dhe kështu me rradhë, derisa nuk gjenden më bashkësi të shpeshta me k - elementë. Gjetja e çdo L_k kërkon një skanim të plotë të bazës së të dhënave. Struktura e algoritmin mund ta paraqitet edhe si më poshtë [80]:
(1) - Kërkohet për të gjithë elementët individualë (bashkësitë me 1 element) që kanë një mbështetje minimale min(s).
(2) - Përsëritet procedura

- *Nga rezultati i kërkimit paraardhës për bashkësitë me k - elementë, kërkohet për bashkësitë me k+1 elementë që kanë një mbështetje minimale min(s).*
- *Kjo bëhet bashkësia e të gjithë bashkësive të shpeshta me (k+1) elementë që janë interesante.*

(3)- Derisa madhësia e bashkësisë të arrijë maksimumin

Pra supozojmë se kemi një bashkësi me n - elementë. Kërkojmë elementët që kanë një mbështetje minimale *min(s).* Këto përbëjnë bashkësitë e shpeshta me një element interesant (pra që kanë një *mbështetje* $\geq s$). Nga kombinimi i këtyre bashkësive formohen bashkësi me dy elementë, nga të cilat përzgjidhen ato që kanë një mbështetje minimale s. Kështu vazhdohet derisa të gjendet bashkësia me numrin maksimal të elementëve që ka *një mbështetje s*.
Për të përmirësuar efiçencën e gjenerimit të bashkësive të shpeshta, përdoret një karakteristikë e rëndësishme që quhet ***karakteristika përparësore*** për të reduktuar hapësirën e kërkimit. Karakteristika përparësore pohon se: *"Të gjithë nënbashkësitë joboshe të një bashkësie të shpeshtë duhet të jenë të shpeshta."dhe* bazohet në vëretjen e mëposhtme:

- Me përkufizim nëse një bashkësi *I* nuk kënaq një vlerë minimale mbështetje *min(s)*, atëherë *I* nuk është bashkësi e shpeshtë, pra *P(I)<min(s).*

- Nëse një objekt *A*, i shtohet bashkësisë *I*, atëherë bashkësia rezultante *(I∪A)* nuk mund të shfaqet më shpesh se *I*. Ndaj, *I∪A* nuk është e shpeshtë, pra *P(I ∪A)<min(s)*.

Kjo karakteristikë i përket një kategorie të veçantë karakteristikash të quajtur ***antimonotoniciteti*** në kuptimin që nëse një bashkësi nuk e kalon provën, të gjithë superbashkësitë e saj nuk do ta kalojnë atë provë. Quhet *antimonotonicitet* për shkak se është monotone në faktin që nuk kalon provën.

5.5.1 PËRDORIMI I KARAKTERISTIKËS PËRPARËSORE

Përdorimi i karakteristikës përparësore në algoritëm bëhet duke ndjekur një proçes dy hapësh: **bashkimi** dhe **krasitja.**

Bashkimi: Për të gjetur L_k, një grup bashkësish kandidate me k - elementë gjenerohet duke bashkuar L_{k-1} me vetveten. Kjo bashkësi kandidatesh shënohet C_k. Le të jenë l_1 dhe l_2 objekte në L_{k-1}. Shënimi $l_i[j]$ i referohet objektit të *j-t*ë në l_i (psh. $l_1[k-2]$ i referohet objektit të parafundit në l_1). Për një implementim efiçent, algoritmi përparësor supozon që objektet në transaksione apo bashkësi janë të renditur në rend leksikografik. Për bashkësinë *(k-1)*, l_i, kjo do të thotë që objektet janë renditur në mënyrë që $l_i[1]<l_i[2]<....<l_i[k-1]$. Bashkimi $L_{k-1}xL_{k-1}$ kryhet kur pjesëtarët e L_{k-1} janë të bashkueshëm-nëse *(k-2)* elementët e parë janë të përbashkët. Pra anëtarët l_1 dhe l_2 të L_{k-1} bashkohen nëse $(l_1[1]=l_2[1]) \wedge (l_1[2]=l_2[2]) \wedge ....\wedge (l_1[k-2]=l_2[k-2]) \wedge (l_1[k-1]<l_2[k-1])$. Kushti $l_1[k-1]<l_2[k-1]$ siguron që nuk gjenerohen duplime. Bashkësia rezultante e formuar nga bashkimi i l_1 dhe l_2 është $\{l_1[1],l_1[2],...,l_1[k-2],l_1[k-1],l_2[k-1]\}$.

Krasitja*:* C_k është një superbashkësi e L_k, pra anëtarët e saj mundet ose jo të jenë të shpeshtë, por të gjitha bashkësitë e shpeshta me k elementë janë të përfshirë në C_k. Një skanim i bazës së të dhënave për të përcaktuar sasinë e çdo kandidati në C_k çon në përcaktimin e L_k (të gjitha bashkësitë kandidate që janë në sasi jo më pak se vlera minimale e mbështetjes janë të shpeshta me përkufizim, ndaj bëjnë pjesë në L_k). Bashkësia C_k mund të jetë e madhe dhe kjo mund të sjellë shumë llogaritje. Për të reduktuar madhësinë e C_k, karakteristika përparësore përdoret si në vijim: Çdo bashkësi me *(k-1)* elementë që nuk është e shpeshtë nuk mund të përdoret si nënbashkësi e një bashkësie të shpeshtë me k - elementë. Ndaj, nëse një nënbashkësi me *(k-1)* elementë e një bashkësie kandidate me k - elementë nuk është në L_{k-1}, atëherë bashkësia kandidate nuk mund të jetë e shpeshtë dhe mund të eliminohet nga C_k. Ky testim nënbashkësish mund të bëhet shpejt duke mbajtur një pemë hash të të gjitha bashkësive të shpeshta.

5.5.2 FUNKSIONIMI I ALGORITMIT PËRPARËSOR

Gjetja e bashkësive të shpeshta duke përdorur një qasje iterative bazuar në gjenerimin e kandidatëve është thelbi i funksionimit per algoritmin përparësor dhe procedurave të lidhura me të.

Hapi 1 i algoritmit gjen bashkësitë e shpeshta me 1-element, L_1. Në hapat 2 deri 10 L_{k-1} përdoret për të gjeneruar kandidatët C_k për të gjetur L_k për $k \geq 2$. Proçedura përparësor_gen gjeneron kandidatët dhe më pas përdor karakteristikën përparësore për të eliminuar ato që kanë nënbashkësi jo të shpeshta (hapi 3). Pasi janë gjeneruar të gjithë kandidatët, skanohet baza e të dhënave (hapi 4). Për çdo transaksion, një funksion nënbashkësi përdoret për të gjetur të gjithë nënbashkësitë e transaksionit që janë kandidatë (hapi 5) dhe sasia e këtyre kandidatëve mblidhen (hapat 6 dhe 7). Në fund të gjithë kandidatët që kënaqin mbështetjen minimale (hapi 9) formojnë bashkësinë e bashkësive të shpeshta, L (hapi 11).

Algoritmi Përparësor. Gjetja e bashkësive të shpeshta duke përdorur një qasje iterative bazuar në gjenerimin e kandidatëve.

Inputi: (D, një bazë të dhënash transaksionesh; min(s), vlera minimale e mbështetjes)
Outputi: L, bashkësitë e shpeshta në D.
Metoda:

```
(1)  L1=find_frequent_1-itemsets(D);
(2)  for(k=2; Lk-1≠Ø; k++){
(3)          Ck=Perparesor_gen(Lk-1);
(4)          for each transaction t ∈ D {//skanohet D për sasinë
(5)                  Ct=subset(Ck,t); //merr nënbashkësitë e t që janë kandidate
(6)                  for each candidate c ∈ Ct
(7)                          c.count++;
(8)          }
(9)          Lk={c ∈ Ck|c.count ≥ min(s)}
(10)         }
(11)         return L=∪kLk;
```

procedura Perparesor_gen (L_{k-1}: bashkësitë e shpeshta me k-1 elementë)

```
(1)  for each itemset l1 ∈ Lk-1
(2)          for each itemset l2 ∈ Lk-1
(3)            if(l1[1]=l2[1]) ∧(l1[2]=l2[2]) ∧....∧(l1[k-2]=l2[k-2]) ∧(l1[k-1]<l2[k-1])
     then {
(4)            c=l1xl2; //bashkimi: gjenerimi i kandidatëve
(5)             if has_infrequent_subset (c,Lk-1) then
(6)                  delete c; //krasitja: hiqen kandidatët jo të frytshëm
(7)            else add c to Ck;
```

(8) }
(9) return C_k;

procedura has_infrequent_subset
(c:bashkësia kandidate me k-elementë;L_{k-1}:bashkësitë e shpeshta me k-1 elementë);
//përdoret njohuria e mëparshme
(1) for each (k-1) subset s of c
(2) if s $\notin L_{k-1}$ then
(3) return TRUE;
(4) return FALSE;

Figura 11 Algoritmi përparësor gjetja e bashkësive të shpeshta duke përdorur një qasje iterative bazuar në gjenerimin e kandidatëve

Një procedurë mund të thërritet për të gjeneruar rregulla shoqërimi nga bashkësitë e shpeshta. Procedura Perparesor_gen kryen dy lloj veprimesh, bashkimin dhe krasitjen. Gjatë bashkimit, L_{k-1} bashkohet me L_{k-1} për të gjeneruar kandidatë potencialë (hapat 1-4). Gjatë krasitjes (hapat 5-7) zbatohet karakteristika perparesore për të eliminuar kandidatët që kanë një nënbashkësi që nuk është e shpeshtë. Gjithashtu paraqitet testi për nënbashkësitë jo të shpeshta nëpermjet procedurës has_infrequent_subset.

5.5.3 METODA PËRMIRËSIMI PËR ALGORITMIN PËRPARËSOR

Janë paraqitur shumë metoda përmirësimi për algoritmin përparësor, që përqëndrohen në rritjen e efiçencës së metodës standarte të tij [81]. Më poshtë kemi prezantuar disa prej tyre:

Teknika e Bazuar në Hash

Një teknikë bazuar në hash mund të përdoret për të reduktuar madhësinë e bashkësive kandidate me k elementë, C_k, për $k>1$. Për shembull, kur skanohet çdo transaksion në bazën e të dhënave për të gjeneruar bashkësitë e shpeshta me një element, L_1, ne mund të gjenerojmë të gjithë bashkësitë me dy elementë për çdo transaksion, të cilat i nënshtrohen teknikës hash, pra hartëzohen në enë të ndryshme të një strukture tabele hash dhe rritet sasia përkatëse në enë. Një bashkësi me 2 elementë me një numërues përkatës në enë në tabelën hash, që ka një vlerë mbështetje më të vogël se vlera minimale nuk mund të jetë e shpeshtë dhe ndaj duhet të eliminohet nga bashkësia e kandidatëve. Një teknikë e tillë e bazuar në hash mund të reduktojë ndjeshëm numrin e bashkësive kandidate me k - elementë që testohen (sidomos kur $k=2$).

Reduktimi i numrit të transaksionit

Një transaksion që nuk ka ndonjë bashkësi të shpeshtë me *k* elementë, nuk mund të përmbajë bashkësi të shpeshta me *(k+1)* elementë. Ndaj një transaksion i tillë mund të shenjohet apo të eliminohet nga konsiderimi i mëtejshëm për shkak se skanimet e mëvonshme të bazës së të dhënave për bashkësitë me *j* elementë, ku *j>k*, nuk kanë nevojë të konsiderojnë një transaksion të tillë.

Copëzimi

Copëzimi i të dhënave për të gjetur bashkësitë kandidate. Mund të përdoret një teknikë copëzimi që kërkon dy skanime databaze për të gërmuar bashkësitë e shpeshta. Konsiston në dy faza. Së pari, algoritmi ndan tranaksionet e *D* në n-pjesë të cilat nuk priten me njëra tjetrën. Nëse vlera minimale e mbështetjes për transaksionet në *D* është *min(s)* dhe sasia minimale e mbështetjes për një pjesë është *min(s)* e x, ku x është numri i transaksioneve në atë pjesë. Për çdo pjesë, gjenden të gjithë bashkësitë e shpeshta lokale (bashkësitë e shpeshta brenda asaj pjese). Një bashkësi lokale e shpeshtë mund të jetë ose jo e shpeshtë në krahasim me të gjithë bazën e të dhënave, *D*. Megjithatë, çdo bashkësi që mund të jetë e shpeshtë në krahasim me *D* duhet të jetë e shpeshtë në të paktën një nga pjesët. Ndaj, të gjithë bashkësitë lokale të shpeshta janë bashkësi kandidate në krahasim me *D*. Bashkësia e bashkësive të shpeshta nga të gjitha pjesët përbën bashkësitë kandidate globale në raport me *D*. Së dyti, kryhet një skanim i dytë i *D* sipas të cilit vlerësohet mbështetja për çdo kandidat për të përcaktuar bashkësitë e shpeshta globale. Madhësia e pjesëve apo copave dhe numri i tyre vendosen në mënyrë të tillë që çdo pjesë të mund të vendoset në memorjen kryesore dhe kështu të lexohet vetëm një herë në çdo fazë.

Figura 12 Algoritmi Përparësor përmirësimi i efiçencës nëpërmjet teknikës së Copëzimit

Kampionimi - Gërmimi në një nënbashkësi të të dhënave

Idea thelbësore e kampionimit është të zgjidhet një kampion i çfarëdoshëm *S*, nga baza e të dhënave *D* dhe më pas të kërkohet për bashkësi të shpeshta në *S* në vend të *D*. Kështu, sakrifikohet pak saktësi në kurriz të efiçencës. Madhësia e kampionit *S* është e tillë që kërkimi për bashkësi të shpeshta në *S* mund të zhvillohet në memorjen kryesore ndaj edhe kërkohet vetëm një skanim në total i transaksioneve në *S*. Për shkak se po kërkojmë për bashkësi të shpeshta në *S* dhe jo në *D*, mund të ndodhë që humbim disa nga bashkësitë globale të shpeshta. Për të reduktuar këtë mundësi, përdorim një vlerë mbështetjeje më të vogël sesa minimumi, për të gjetur bashkësitë e shpeshta lokale në *S* (shënohet L^S). Pjesa tjetër e bazës së të dhënave përdoret për të llogaritur shpeshtësitë e çdo bashkësie në L^S. Përdoret një mekanizëm për të përcaktuar nëse të gjitha bashkësitë e shpeshta globale janë të përfshira në L^S. Nëse L^S vërtet përmban të gjitha bashkësitë e shpeshta në *D*, atëherë kërkohet vetëm një skanim i *D*. Në të kundërt, kryhet edhe një skanim tjetër për të gjetur bashkësitë e shpeshta që ishin anashkaluar në hapin e parë. Kampionimi është veçanërisht me përfitim kur eficienca është me rëndësi të madhe si në rastet e aplikacioneve intesive kompjuterike që duhet të ekzekutohen shpesh.

Numërimi dinamik i bashkësive

Shtimi i bashkësive kandidate në pika të ndryshme gjatë skanimit. Një teknikë e numërimit dinamik i bashkësive e propozuar, konsiston në copëzimin e bazës së të dhënave në blloqe të shenjuar në pika fillimi. Në këtë rast, bashkësitë kandidate të reja mund të shtohen në çdo pikë fillimi, ndryshe nga karakteristika përparësore, e cila përcakton bashkësi të reja kandidate vetëm para çdo skanimi të plotë të bazës së të dhënave. Kjo teknikë përdor numërimin deri në atë pikë si kufiri i poshtëm i numërimit të vërtetë. Nëse numërimi deri në atë pikë kalon mbështetjen minimale, bashkësia i shtohet grupit të bashkësive të shpeshta dhe mund të përdoret për të gjeneruar kandidate më të gjata. Kjo sjell më pak skanime të bazës së të dhënave sesa me karakteristikën përparësore për gjetjen e të gjithë bashkësive të shpeshta.

Rritja e modeleve të shpeshtë

Në shumë raste algoritmi përparësor redukton ndjeshëm madhësinë e bashkësive kandidate, e cila çon në një përfitim të mirë të performancës [82]. Megjithatë ka edhe dy kosto domethënëse:

- Mund të ketë ende nevojë të gjenerojë në numër të madh bashkësish kandidate. Për shembull, nëse kemi 10^4 bashkësi të shpeshta me *1* element, algoritmi

përparësor do të duhet të gjenerojë më shumë se 10^7 bashkësi kandidate me *2* elementë.

- Mund të ketë nevojë të skanojë vazhdimisht të gjithë bazën e të dhënave dhe të kontrollojë një bashkësi të madhe kandidatësh duke i lidhur me modelet. Është e kushtueshme të kalosh çdo transaksion në bazën e të dhënave për të përcaktuar mbështetjen e të gjitha bashkësive kandidate.

Rritja e modeleve të shpeshtë në fillim kompreson bazën e të dhënave duke paraqitur objektet e shpeshta në një pemë të modeleve të shpeshta, e cila përmban informacionin mbi shoqërimin e objekteve. Më pas ndan bazën e të dhënave të kompresuar në një bashkësi bazash të dhënash kondicionale, ku secila është shoqëruar me një objekt të shpeshtë apo "fragment modeli" dhe gërmon çdo bazë të dhënash në mënyrë të veçantë. Për çdo "fragment modeli", duhet të kontrollohen vetëm të dhënat e shoqëruara me të. Ndaj kjo metodë mund të reduktojë ndjeshëm madhësinë e bashkësive të të dhënave që do të kërkohen, së bashku me "rritjen" e modeleve që do të kontrollohen.

Gjithashtu metoda e rritjes se modeleve të shpeshtë transformon problemin e gjetjes së modeleve të gjatë të shpeshtë në kërkimin për modele të shpeshtë më të shkurtër, në mënyrë rekursive në baza të dhënash më të vogla kondicionale dhe më pas në lidhjen më prapashtesën. Kjo metodë përdor objektet më pak të shpeshta si prapashtesa, duke ofruar përzgjedhshmëri të mirë dhe më e rëndësishme ul ndjeshëm koston e kërkimit. Kur baza e të dhënave është e madhe, nuk mund të ndërtojmë një pemë te modeleve të shpeshtë të bazuar në memorjen kryesore [83]. Një mundësi do të ishte ndarja e bazës së të dhënave në një bashkësi bazash të dhënash të projektuara dhe më pas të ndërtohet një pemë dhe të gërmohet në çdo bazës të dhënash të projektuar. Ky proçes mund të zbatohet në mënyrë rekursive për çdo bazë të dhënash të projektuar nëse për pemën e saj nuk ka vend në memorje. Një studim i performancës së kësaj metode tregon se është efiçente dhe e shkallëzueshme për gërmimin e modeleve të gjatë dhe të shkurtër. Algoritmi që del nga metoda e rritjes së modeleve të shpeshtë funksionon në një peme nëpërmjet zgjedhjes së një termi sipas rendit rritës të shpeshtisë dhe përftimit të objekteve me të shpeshtë që përmbajnë termin e zgjedhur.

5.5.4 METODA E PROPOZUAR PËR PROBLEMET NË SIGURIME

Që me prezantimin fillestar të algoritmit janë bërë shumë përpjekje për të shpikur me shumë algoritme të sigurta të përpunimit të bashkësive të objekteve të shpeshtë. Pjesa me e madhe ndajnë të njëjtin mendim me algoritmin përparësor në faktin e

gjenerimit të termave. Këto përfshijnë teknikën e bazuar në thjeshtimin e të dhënave, ndarjes, kampionimit dhe përdorimit të formatit vertikal të të dhënave. Teknika e përpunimit këtu mund të zvogëlojë madhësinë e gruptermave në seri. Meqënëse një skedar mund të përmbajë grupterma të ndryshem nëse vargu është më i vogël sesa një vlerë minimale, keto grup terma në skedarë mund të eleminohen nga grupet seriale të të dhënës. Nje ndarje mund të përdoret për të thjeshtuar problemin e plotë të përpunimit në *n*- probleme më të vogla. Grupi i të dhënave ndahet në nënndarje jo të mbivendosura të tillë që çdo klasë të bie në memorien kryesore dhe secila ndarje të përpunohet veçmas [84].

Përmirësimi i propozuar për algoritmin përparësor në sigurime është metoda e rritjes së modeleve të shpeshtë që thjeshton termin e perftuar. Metoda përshtat një ndarje duke dhënë strategjinë për vendosjen e të dhënave që përfaqësojnë termat e shpeshtë brënda një strukture, e cila përmban të gjithë informacionin kryesor. Metoda e rritjes së modeleve të shpeshtë e ndan strukturën e krijuar në dy pjesë dhe përpunimi i secilit prej tyre kryhet në mënyrë të veçantë. Metoda e skanon bazën e të dhënave vetëm 2 herë. Në skanimin e parë përftohen të gjithë termat e shpeshtë dhe zgjidhjet që shoqërojnë ato në seri dhe ndahen sipas një rradhe në zbritje të përftimit të një serie zgjidhjesh për secilin kalim të bërë. Në skanimin e dyte llogariten termat në secilin kalim, shkrihen në një pemë prefiksi dhe termat (nyjet) që shfaqen sëbashku në kalime të ndryshme.

6 - ALGORITMET GJENETIKE

Ky kapitull është fokusuar kryesisht në algoritmat gjenetikë dhe rolin e tyre në vendimmarrje në fushën e sigurimit të jetës. Fillimisht kemi prezantuar se çfarë janë algoritmat gjenetikë, strukturën e tij dhe parametrat kryesorë në funksionimin e tij. Gjithashtu janë dhënë problemet e performancës që hasen tek algoritmat gjenetike. Kemi hulumtuar një algoritëm gjenetik të ri për problemin e klasifikimit të teksteve ne sigurime.

6.1 ÇFARË JANË ALGORITMAT GJENETIKË

Algoritmat Gjenetikë (në vijim AGj) janë zhvilluar nga Holande në vitin 1975 si teknika kërkimi dhe optimizimi. AGj janë shumë efiçente në kërkimin e zgjidhjeve në hapësira shumë të mëdha, janë metoda kërkimi bazuar në popullatë të përbërë nga individë dhe imitojnë disa prej proçeseve të evolucionit dhe selektimit natyror [85]. Në natyrë, secila specie ka nevojë që të përshtatet ndaj një ambjenti të komplikuar dhe është gjithnjë e në ndryshim në mënyrë që të maksimizojë apo të rrisë mundësitë e mbijetesës së saj [86]. AGj janë metoda optimizimi të forta dhe efiçente për problemet komplekse të karakterizuara si: nga optimizues të shumëfishtë ose kritere të tjerë të parregullt. Për të gjetur vlerën optimale AGj përdor një trajtim përsëritës numerik, ndryshe nga teknikat e tjera të optimizimit të cilat përdorin trajtimin analitik [87].
AGj janë një formë e Inteligjencës Artificiale, e cila bazohet mbi idenë e simulimit njerëzor, si psh. aftësia e vendimmarrjes duke përdorur kompjuterin. Ato i përkasin klasës së algoritmeve evolucionare, duke zgjidhur probleme te ndryshme nga evolucioni i një popullate fillestare të zgjidhjeve të mundshme drejt zgjidhjeve më të mira, nëpërmjet një proçesi përsëritës [88].
AGj gjejnë aplikim në bioinformatikë, gjenetikë, shkenca kompjuterike, inxhinieri, ekonomi, kimi, prodhim, matematikë, fizikë dhe fusha të tjera.

6.2 STRUKTURA E ALGORITMAVE GJENETIKË

Evolucioni zakonisht fillon nga një popullatë e krijuar me individë rastësor dhe që ndodhin në breza. Në çdo brez, vlerësohet përshtatshmëria e çdo individi në popullatë, ku individë të shumtë përzgjidhen në mënyrë statistikore nga popullata aktuale dhe modifikohen për të formuar një popullatë të re. Popullata e re përdoret më pas në përsëritjen e ardhshme të algoritmit. Algoritmi mbaron kur është

prodhuar një numër maksimal i brezave, ose kur është arritur një nivel i kënaqshëm i përshtatshmërisë së popullatës [89].
Kromozomet janë individët e popullatës të paraqitur si vargje gjenesh ose stringjesh të koduara në 0 dhe 1. Paraqitja në pseudokod e AGj është dhënë në figurën 13, ku përshkruhet procedura e funksionimit të tij nëpërmjet parametrave.

Procedura e Algoritmit Gjenetik
{t - Koha, P(0) – Popullata fillestare në kohën t=0; P(t) - Popullata në kohën t}
Fillim
(1) t=0;
(2) INICIALIZO popullatën me individe kandidatë të rastësishëm P(t);
(3) VLERESO secilin kandidat P(t);
(4) Përsërit
(5) t=t+1;
(6) PERZGJIDH P(t) nga P(t-1);
(7) RIPRODHO çiftet në P(t);
(8) VLERESO P(t) ;
(9) Përderisa KUSHTI I MBARIMIT të plotësohet
Fund

Figura 13 Procedura e funksionimit të Algoritmit Gjenetik

Kjo procedurë futet në kategorinë e algoritmave *gjenero dhe testo*. Funksioni objektiv përfaqëson një vlerësim orientues për cilësinë e zgjidhjes dhe proçesi i kërkimit në procedurën e AGj-së është drejtuar nga përzgjedhja e prindërve dhe operatorët e ndryshimit.

6.3 PARAMETRAT KRYESORE TE ALGORITMIT GJENETIK

Parametrat kryesore të funksionimit të një algoritmi AGj janë [90]: (1) Përfaqësimi i individëve: (2) Popullata fillestare; (3)Funksioni Objektiv dhe Funksioni i Fitnesit; (4) Përzgjedhja e prindërve; (5) Operatorët e ndryshimit.

6.3.1 PËRFAQËSIMI I INDIVIDËVE

Paraqitja binare për një individ kandidat paraqitet në formën e një vektori binar, ku çdo vlerë e secilit prej *n - elementëve* është 0 ose 1. Kjo paraqitje bazohet në sistemin binar të numrave (formula 6.1).

$$x = [x_1, x_2, x_3, x_4, \cdots, x_n] \qquad (6.1)$$

Kromozomi përfaqëson një zgjidhje të problemit dhe është i përbërë nga një varg gjenesh ose stringjesh me gjatësi të fundme. Elementët e një kromozomi janë quajtur gjene dhe vlera e një gjeni është quajtur alel. Struktura e të dhënave të një kromozomi paraqitet me një matricë të thjeshtë me madhësi *N x M* elemente, ku *N* është numri i individëve në popullatë dhe *M* është gjatësia e gjenotipit që paraqitet nga këta individë (formula 6.2).

$$Kromozomi = \begin{bmatrix} g_{1,1} & g_{1,2} & \cdots & \cdots g_{1,M} \\ g_{2,1} & g_{2,2} & \cdots & \cdots g_{2,M} \\ \cdots & \cdots & \cdots & \cdots \ \cdots \\ g_{N,1} & g_{N,2} & \cdots & \cdots g_{N,M} \end{bmatrix} \quad (6.2)$$

Individët që formojnë zgjidhje të mundshme brënda kontekstit kryesor të problemit në AGj janë quajtur fenotipe. Hapi i përfaqësimit të individëve specifikon hartëzimin nga fenotipet në një bashkësi gjenotipesh. Fenotipet dhe individët përdoren për të paraqitur zgjidhjet e mundshme në hapësirën e fenotipeve [91].

6.3.2 Popullata Fillestare

Sapo vendoset një përfaqësim i përshtatshëm për kromozomet (individët), është e nevojshme që të krijohet një popullatë fillestare që shërben si pika e nisjes për AGj-në. Kjo popullatë fillestarë zakonisht krijohet në mënyrë të rastësishme, madhësia e saj zakonisht merret 30 dhe 100 individë. Roli i popullatës është që të mbajë zgjidhjet e mundshme dhe madhësia e popullatës është konstante dhe nuk ndryshon gjatë kërkimit evolutiv.

6.3.3 Funksioni Objektiv dhe Funksioni i Fitnesit

Funksioni objektiv përdoret për të siguruar një masë matjeje për rritjen e përshtatjes se individëve "më të mirë" dhe përshtatjen e të gjithë popullatës në një të tërë. Në rastin e një popullate minimale, individët më të mirë kanë vlerën më të ulët numerike të funksionit objektiv. Kjo masë përdoret vetëm si një fazë e ndërmjetme në përcaktimin e performancës relative të individëve në një AGj. Në rastin e rutinave të optimizimit, përshtatshmëria është vlera e fuksionit objektiv që duhet optimizuar.

Ndërsa funksioni i fitnesit përdoret normalisht për të transformuar vlerën e funksionit objektiv në një vlerë pozitive relative (formula 6.3).

$$F(x) = g\big(f(x)\big) \quad (6.3)$$

Ku: f - funksioni objektiv

g - transformon vlerën e funksionit objektiv në numër jonegativ

Në shumë raste, vlera e funksionit të fitnesit korrespondon me numrin e pasardhësve që një individ mund të riprodhojë në gjeneratën e ardhshme. Një transformim i përdorur zakonisht është ai i caktimit proporcional të fitnesit.
Fitnesi individual $F(x_i)$ për secilin individ llogaritet si performanca e parë e tij në raport me të gjithë popullsinë (formula 6.4):

$$F(x_i) = \frac{f(x_i)}{\sum_{i=1}^{N} f(x_i)} \qquad (6.4)$$

Ku: N - madhësia e popullatës

x_i - vlera e fenotipit për çdo individ i

Funksioni i fitnesit siguron që çdo individ ka një probabilitet riprodhimi sipas vlerës së fitnesit të tij më të afërt.

6.3.4 PËRZGJEDHJA E PRINDËRVE

Përzgjedhja e prindërve nga popullata aktuale eshte i nevojshëm për proçesin e riprodhimit. Prindërit e përzgjedhur duhet të jenë individë të përshtatshëm nga popullata. Efekti ose ndikimi i përzgjedhjes është që të kthejë një prind të përzgjedhur në mënyrë probabilistike. Tre teknikat më të zakonshme të përzgjedhjes janë [92]:

a) Përzgjedhja Bazuar në Përshtatshmëri

Metoda origjinale për përzgjedhjen e prindërve është përzgjedhja bazuar në përshtatshmërinë e individit. Në këtë lloj përzgjedhje të prindërve, secili kromozom ka një shans përzgjedhjeje, e cila është proporcionale në mënyrë direkte me përshtatshmërinë e tij. Efekti apo ndikimi i tij varet nga hapësira e vlerave të përshtatjes në popullatën që analizojmë.

b) Përzgjedhja e Ruletës Rrotulluese

Ideja që fshihet pas teknikës së përzgjedhjes së prindërve me anë të ruletës rrotulluese është ajo që secilit individ i jepet një mundësi që të bëhet prind në përpjesëtim me vlerësimin e përshtatshmërisë së tij. Është quajtur përzgjedhja e ruletës rrotulluese pasi shancet e përzgjedhjes së një prindi mund të shihen si rrotullimi i një rulete me madhësinë e slotit për secilin prind që është në përpjesëtim me përshtatshmërinë e tij. Si pasojë, ata me përshtatshmëri më të madhe kanë më shumë shanse që të përzgjidhen.

c) Përzgjedhja me Dyluftim

Në përzgjedhjen me dyluftim, prindërit potencialë janë përzgjedhur dhe dyluftimi mbahet për të vendosur se cili prej individëve do të jetë prind. Përzgjedhja origjinale me dyluftim është që të zgjedhim k - prindër në mënyrë rastësore dhe të kthejmë më të përshtatshmin prej tyre.

6.3.5 OPERATORET E NDRYSHIMIT

a) Operatori i Mbikalimit

Mbikalimi është proçesi i cili jep rikombinimet e biteve të gjeneve nëpërmjet shkëmbimit të segmenteve në çiftet e kromozomeve [93]. Operatori i mbikalimit është shumë i rëndësishëm për AGj-në.

Mbikalimi me shumë-pika

Elementët kryesore për një mbikalim me shumë pika janë përcaktuar si më poshtë:
m – pozicione të mbikalimit, $k_i \in \{1, 2, 3, \dots, l-1\}$ - pikat e mbikalimit dhe l - gjatësinë e kromozomit;

Proçesi i funksionimit të tij konsiston në shkëmbimin e biteve ndërmjet pikave të njëpasnjëshme të mbikalimit midis të dy prindërve për të prodhuar dy pasardhës të rinj, ku seksioni midis pozitës së parë alele dhe pikës se parë te mbikalimit nuk shkëmbehet midis individëve. Ky proçes është ilustruar në figurën e mëposhtme:

Figura 14 Algoritmi Gjenetik mbikalimi me shumë pika (m=5)

Sapo përzgjidhet një çift kromozomesh, mbikalimi vepron për të prodhuar një rezultat. Kur probabiliteti i një mbikalimi ka vlerën 1.00 tregon se të gjithë kromozomet e përzgjedhur janë përdorur në riprodhim. Megjithatë, studimet empirike tregojnë se rezultatet më të mira arrihen nga mbikalimi kur vlera e probabilitetit është ndërmjet vlerave 0.65 dhe 0.85.

Mbikalimi në 1-pikë

Proçedura e mbikalimit në 1-pikë është që të gjenerojë në mënyrë rastësore një numër pozicioni të mbikalimit. Më pas, mbajmë bitet përpara numrit të pandryshuara dhe i ndërrojmë vendin biteve pas pozicionit të mbikalimit ndërmjet dy prindërve.

Mbikalimi në 2-pika

Në një mbikalim me 2-pika, kromozomet shihen si cikle duke bashkuar fundet e njëri-tjetrit. Proçedura e mbikalimit në 2-pika është e ngjashme me mbikalimin në 1-pikë përveç faktit se duhet të zgjedhim dy pozicione dhe vetëm bitet ndërmjet dy pozicioneve shkëmbehen ose ndryshohen. Kjo metodë mbikalimi mund të ruajë pjesët e fillimit dhe të mbarimit të një kromozomi dhe mundet thjesht që të shkëmbejë pjesën që ndodhet në mes.

Mbikalimi Uniform

Proçedura e mbikalimit uniform ku secili gjen i prindit të parë ka një probabilitet prej 0.5 për shkëmbim me gjenin korrespondues të prindit të dytë. Mbikalimi uniform përfshin një mesatare pikash të mbikalimit l/2 për kromozome me gjatësi l. Në mbikalimin uniform, secili gjen tek pasardhësi është krijuar duke kopjuar gjenin korrespondues nga të dy prindërit sipas një maske të mbikalimit të gjeneruar në mënyrë të rastësishme.

b) Operatori i Mutacionit

Në qoftë se ne përdorim vetëm veprimin e mbikalimit për të prodhuar një rezultat, një problem i rëndësishëm që mund të dalë është që nëse të gjitha kromozomet në popullatën fillestare kanë të njëjtën vlerë ndaj një pozicioni të caktuar, të gjitha rezultatet në të ardhmen do të kenë këtë vlerë të njëjtë për këtë pozicion. Psh: nëse të gjitha kromozomet kanë një 0 në pozicionin e dytë, atëherë të gjitha rezultatet në të ardhmen do të kenë 0 në pozicionin e dytë.
Për të kontrolluar këtë situatë të padëshirueshme, përdorim operatorin e mutacionit, i cili parashikon futjen e disa alternativave të rastësishme të gjeneve psh: 0 të bëhet 1 dhe anasjelltas. Kryesisht kjo gjë ndodh rrallë kështu që mutacioni është i rendit prej një biti ndryshimi në një mijë të testuar. Secili bit në secilin kromozom është kontrolluar për mutacione të mundshme duke gjeneruar një numër të rastësishëm 0 dhe 1 dhe nëse ky numër është më i vogël ose i barabartë me një probabilitet mutacioni të dhënë psh: 0.001 atëherë vlera e bitit ndryshohet. Ky operator kompleton ciklin e punës të AGj-së, ku përshtatja e secilit kromozom në popullatën e re vlerësohet dhe e gjithë proçedura përsëritet [94].

6.4 Problemet e Performancës të Algoritmave Gjenetikë

a) Problemi i Operatorit të Mutacionit

Të qënurit i sigurtë se të gjitha kromozomet e mundshme janë të arritshëm përbën një problem per operatorin e mutacionit. Operatorët e mbikalimit e kryjnë kërkimin të kufizuar, vetëm ndaj aleleve të cilët ekzistojnë në popullatën fillestare. Ndërsa operatori i mutacionit është një proçes modifikimi i rastësishëm të vlerës së një kromozomi me probabilitet të vogël. Ai siguron që probabiliteti i kërkimit për një zonë në hapësirën e problemit nuk është kurrë zero, duke parandaluar humbjet totale të materialit gjenetik të marrë nëpërmjet riprodhimit dhe mbikalimit. Operatori i mutacionit mund ta kapërcejë këtë thjesht duke përzgjedhur në mënyrë rastësore pozicionin e secilit bit në një kromozom dhe duke e ndryshuar atë. Kjo është shumë e përdorshme pasi

operatori e mbikalimit mund të mos jetë i aftë që të prodhojnë alele të reja nëse ata nuk shfaqen në gjeneratën e parë [95].

b) *Problemi i Madhësisë së Popullatës*
Zgjedhja e madhësisë së popullatës për algoritmit AGj ndikon në efiçencën e tij. Në qoftë se ne kemi popullatë të vogël, ajo do të mbulojë vetëm një hapesirë kërkimi të vogël, e cila mund të rezultojë me përformancë të ulët. Një popullatë më e madhe do të mbulonte më shumë hapësirë dhe do të parandalonte konvergjencën e parakohshme të zgjidhjeve lokale. Në të njëjtën kohë, një popullatë më e madhe ka nevojë për më shumë vlerësim për gjenerimet dhe mund të ngadalësojë shkallën e konvergjencës. Rritja e kompleksitetit të algoritmit AGj na drejton gjithmonë e më shumë drejt nevojës për të patur një popullate me madhësi të madhe [96].

6.5 PËRDORIMI I AGJ NE KLASIFIKIMIN TË TEKSTEVE

Shoqeritë e sigurimeve të jetës përveç të dhënave të strukturuara përdorin edhe të dhëna tekst, të cilat fokusohen në kërkim sipas një query me fjalë kyçe [97]. Ne jemi përqëndruar në studimin tonë në rritjen e efektivitetit dhe efiçencës për problemet e klasifikimit të të dhënave tekst në sigurime.

6.5.1 PROBLEMI I KLASIFIKIMIT TE TEKSTIT

Klasifikimi i tekstit është detyra e caktimit të teksteve të gjuhës natyrore të një ose më shumë kategorive tematike mbi bazën e përmbajtjes së tyre. Një numër metodash të ML janë propozuar në vitet fundit, duke përfshirë algoritmin e fqinjësisë më të afërt, rrjetat nervore, algoritmin gjenetik etj. [98] Ne do të trajtojmë problemin e futjes së klasifikuesve të tekstit në sigurime në formën:

$$c \leftarrow (t_i \in d \; V \cdots V \; t_n \in d) \wedge \neg (t_{n+1} \in d \; V \cdots V \; t_{n+m} \in d) \qquad (6.5)$$

ku:
c - Një bashkësi e fundme të kategorive, të quajtura skema të klasifikimit;
d - është një tekst
D - Një bashkësi e fundme dokumenta tekst, të quajtur përmbledhje (korpus);
t_i - është një term i marrë nga një fjalor i dhënë
Kc(Pozitiv, Negativ) - është një klasifikues për kategorinë c
Pozitiv = $\{ t_1 , \cdot , t_n \}$ - bashkësia e trajnimit
Negativ = $\{ t_{n+1} \cdot t_{n+m} \}$ - bashkësia test
Gjithashtu për të zgjidhur problemin e futjes së klasifikuesve të tekstit supozojmë:

-Një marrëdhënie binare e cila i cakton çdo teksti $d \in D$ një numër kategorish në C.
-Një bashkësi $\varphi = \{ f_1, \cdot, f_k \}$ e kritereve per përzgjedhjes e atributeve, siç janë: përfitimi i informacionit, χ^2, koefiçienti i përfitimit, indeksi Gini etj.
-Një fjalor *V(k, f)*, i cili është një bashkësi e *k - termave* që ndodhen në tekstin *d* dhe që janë përzgjedhur nga kriteri *f*.
-Çdo term në bashkësinë *Pozitiv* = $\{ t_1,\cdots,t_n \}$ është një term pozitiv dhe çdo term në bashkësinë *Negativ* = $\{ t_{n+1},\cdots,t_{n+m} \}$ është një term negativ.
-Klasifikuesi *Kc(Pozitiv, Negativ)* bazohet në kushtin: nëse ndonjë nga termat $t_1,\cdots,t_n$ gjënden në *d* dhe asnjë prej termave $t_{n+1},\cdots,t_{n+m}$ nuk gjenden në *d* atëherë klasifikojmë *d* në kategorinë *c*". Kjo tregon se gjetja e një termi pozitiv në tekstin *d* kërkon mungesën kontekstuale të termave negative në mënyrë që *d* të klasifikohet në *c*.
Për të vlerësuar performancën e klasifikuesit tekst, në eksperimentet tona kemi përdorur kriteret: Precision, Recall dhe F-measure të përcaktuara si më poshtë:

- Precision është raporti i numri të fjalëve kyçe termave të përshtashëm që janë nxjerrë nga algoritmi automatik, me numrin total të termave të nxjerra nga algoritmi automatik:

$$Precision = \frac{a}{a+b}; \tag{6.6}$$

- Recall është raporti i numrit të termave të përshtashëm që janë nxjerrë nga algoritmi automatik, me numrin total të termave të cilësuara të përshtashëm.

$$Recall = \frac{a}{a+e}; \tag{6.7}$$

Ku:
a - numri i termave të përshtashëm, që janë klasifikuar në mënyrë korrekte në c
b - numri i termave të përshtashëm, që janë klasifikuar gabimisht si pozitive në c
e - numri i termave të përshtashëm, që janë klasifikuar gabimisht si negative në c
Njësia e matjes së performancës për klasifikuesit tekst është F-measure, i cili përdor kriteret Precision dhe Recall dhe përcaktohet si më poshtë:

$$F\text{-}measure = \frac{2a}{2a+b+e}; \tag{6.8}$$

6.5.2 Algoritmi AGJ për Klasifikimin e Teksteve

Problemi i të mësuarit per klasifikuesin *Kc (Pozitiv, Negativ)* është formuluar si një detyrë optimizimi, me synimin gjetjen e bashkësive përfaqësuese, kur klasifikuesi aplikohet mbi bashkësinë e trajnimit. Bashkësia e trajnimit përfaqësohet si një problem kombinatorik optimizimi për qëllim gjetjen e një kombinimi më të mirë të

termave të marra nga një fjalor të caktuar. Përdorimi i AGj-së rezulton të jetë një metodë zgjidhjeje efikase per klasifikimin e teksteve. Metoda e propozuar është një algoritëm i të mësuarit me një hap, e cila nuk ka nevojë për asnjë lloj optimizimi të mëpasshëm për të përmirësuar bashkësinë e rregullave të gjetura.

Algoritmi AGj për klasifikimin e teksteve
Të dhëna: fjalori V(f, k) mbi bashkësinë e trajnimit; n - numri i brezave;
Rezultati: klasifikuesi më i "mirë" Kc(Pozitiv, Negativ) i c mbi bashkesine e trajnimit;
- Fillim
- Vlerëso bashkësinë e kandidatëve të termave pozitive dhe negative nga V(f, k);
- Krijo popullatën P(0) dhe inicializo çdo kromozom;
- Per i=1 deri n kryej
- Vlerëso përshtatshmërinë e secilit kromozom në P(i);
- P(i+1) = Ø;
- Kopjo në P(i+1) kromozomet më të mira të P(i)
Perderisa madhësia (P(i+1)) < madhësia (P(i))
- Zgjidh prind1 dhe prind2 në P(t) nëpërmjet ruletës rrotulluese
- Gjenero fëmi1, fëmi2 nëpërmjet mbikalimit (prind1 , prind2)
- Apliko mutacionin dhe operatorin riparues;
- Shto fëmi1 dhe fëmi2 në P(i+1);
- fund perderisa
- P(i)= P(i+1);
- Fund per deri kryej;
- Zgjedhim kromozomin K më të mirë në P(i);
- Eliminojme tepricat nga kromozomi K;
- paraqesim klasifikuesin Kc(Pozitiv, Negativ) lidhur me kromozomin K.

Figura 15 Procedura e AGj-së për problemin e klasifikimit të të dhënave

Ndërsa AGj është në kërkim të klasifikuesit më të"mirë" për kategorinë c mbi bashkësinë e trajnimit, për një fjalor hyrës të caktuar, procesi i të mësuarit është në kërkim të një klasifikuesi të "mirë" për c mbi bashkësinë test, për të gjithë fjalorët e dhënë. Më saktësisht, procesi i të mësuarit vazhdon si më poshtë (figura 16):

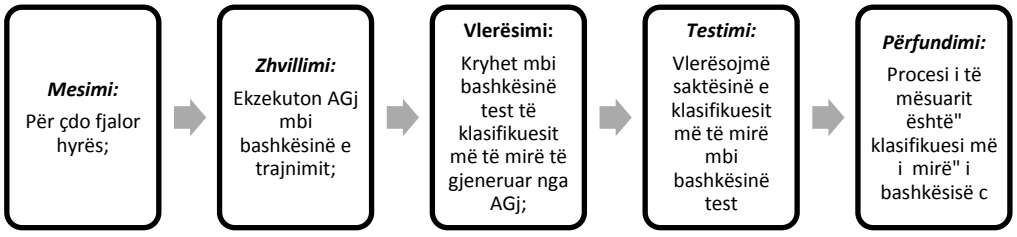


Figura 16 Proçesi i të mësuarit të AGj-së për problemin e klasifikimit të teksteve

KAPITULLI 7- MJETET PËR ANALIZË DHE PARAQITJE TË TË DHËNAVE

Ky kapitull fokusohet në përdorimin e programeve të DM, të cilat bazohen ne algoritmat standarte për përmirësimin e treguesve të algoritmave CART, KNN, K-mesatareve, algoritmit përparësor dhe algoritmit gjenetik. Janë trajtuar mjetet për ndërtim, analizim dhe paraqitje të të dhënave për të eksperimentuar algoritmat standarte dhe të propozuar. Per realizimin e ketyre eksperimenteve kemi përdorur të dhënat nga një shoqëri sigurimi.

7.1 SPSS

IBM SPSS është një mjet i fuqishëm për analizimin dhe paraqitjen e të dhënave. Për ilustrimin e algoritmeve të ndryshëm do të përdoret një bazë të dhënash që përmban të dhënat e klientëve dhe produkteve te tyre në një shoqëri sigurimi. Versionet aktuale (2014) të saj janë statistika, mbledhja e të dhënave, modelimi dhe analizimi i teksteve. Duke filluar nga version 16 IBM SPSS punon vetëm në sistemet operativ Windows, Mac dhe Linux dhe ndërfaqja e saj GUI është shkruar në Java.

Figura 17 Ambjenti i punës në SPSS

Për të aplikuar dhe analizuar teknikat e klasifikimit në sigurimin e jetës, ne kemi përdorur modulet Decision Tree, Nearest Neighbors dhe Neural Networks.

Moduli Decision Tree (Peme Vendimi) ndihmon për të identifikuar më mirë grupet, për të zbuluar marrëdhëniet ndërmjet tyre dhe per të parashikuar ngjarjet e ardhshme. Ky modul përmban klasifikimin nëpërmjet pemëve të vendimit, të cilat mund të paraqesin rezultatet kategorike në mënyrë intuitive. Ai përfshin katër algoritme pemë vendimi te cilat po i paraqesim shkurtimisht më poshtë:

- CHAID dhe Exhaustive CHAID (një modifikim i CHAID) janë algoritme pemë vendimi të shpejtë dhe statistikore që shqyrtojne të dhënat shpejt dhe me efikasitet, ndërtojnë klasa dhe profile në lidhje me rezultatin e dëshiruar.
- CART është një algoritëm pemë vendimi i plotë binare që ndan të dhënat dhe prodhon nënbashkësi të sakta homogjene.
- QUEST është një algoritëm statistikor që zgjedh variablat pa paragjykim dhe ndërton pemë te sakta binare shpejt dhe me efikasitet.

Moduli Nearest Neighbors (Algoritmi i fqinjësisë më të afërt) përmban klasifikimin nëpërmjet algoritmit KNN, i cili bazohet në ngjashmërinë e rasteve dhe nga largësia midis tyre. Rastet e klasifikikuara vendosen në kategorinë që përmban numrin më të madh të fqinjëve të afërt.

Moduli Neural Networks (Rrjeta Nervore) ofron procedura jolineare modelimi dhe mundëson zbulimin e marrëdhënieve më komplekse në bazën e të dhënave. Gjithashtu ky modul mund të zhvillojë modele të sakta, efektive dhe parashikuese për të gjetur në bazën e të dhënave marrëdhënie te fshehura duke përdorur procedurën multilayer perceptron (MLP) ose funksionin radial base (RBF).

7.2 WEKA

Weka (Waikato Environment for Knowledge Analysis) është një mjet mësimi i prezantuar nga universiteti Waikato, Zelanda e Re dhe përdoret për kërkim, edukim dhe projekte të ndryshme. Ndërfaqja e saj GUI është shkruar në Java dhe është e aplikueshme për sistemet operativ Windows, Mac dhe Linux. Përmban koleksione të mëdha algoritmesh të DM dhe mjete për teknikat si regresioni, klasifikimi, grupimi, rregullat e shoqerimit dhe vizualizimi. Versioni që ne kemi përdorur për të analizuar të dhënat është 3.7.4 dhe të dhënat që përpunohen mund të jenë të formateve .arff, .csv, .lib, .svm, URL ose baza të dhënash. Weka është burim i hapur dhe pa kosto, një platformë e pavarur, i lehtë në përdorim dhe fleksibël për shkrimin e eksperimenteve.
Zgjedhësi Weka GUI (klasa weka.gui.GUIChooser) ofron një pikënisje për nisjen e aplikacioneve kryesore ne WEKA dhe mjeteve te saj dhe përbëhet nga katër menu kryesore:
Explorer: Mjedis për eksplorimin e të dhënave.
Experimenter: Një mjedis për kryerjen e eksperimenteve dhe kryerjen e testeve statistikore ndërmjet skemave të të mësuarit.
KnowledgeFlow: Ky mjedis mbështet në thelb të njëjtat funksione si Explorer, por me një ndërfaqe të drag-and-drop.

SimpleCLI: Ofron një ndërfaqe të thjeshtë command-line që lejon ekzekutimin e drejtpërdrejtë e komandave ne WEKA për sistemet operative.

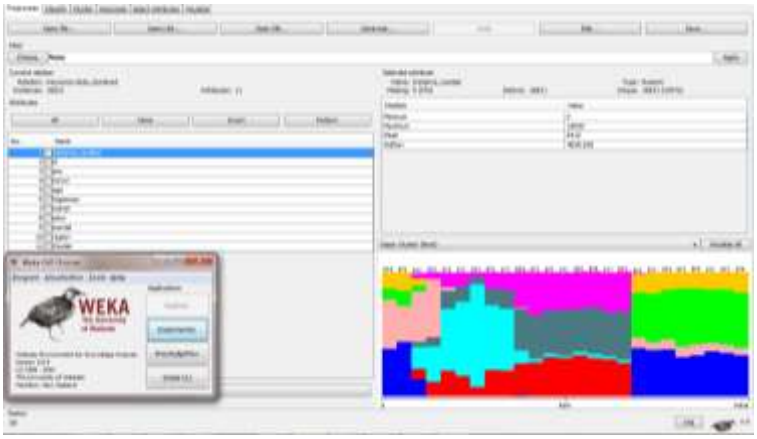


Figura 18 Ambjenti i punës në WEKA

Për të zbatuar algoritmin e grupimit të k-mesatareve në sigurimin e jetës kemi përdorur modulin Cluster në Weka. Algoritmet e grupimit që përfshihen në këtë modul janë: Dbscan, EM, Hierarchical dhe K-means. Implementimi i eksperimentit është përmbledhur në kodimin e k-mesatareve dhe në përmirësimin e procesit të përzgjedhjes së centroideve fillestare duke përdorur gjuhën e programimit Java dhe në këtë mënyrë algoritmi ekzekutohet në çdo platform.
Analiza e shoqërimit në sigurimin e jetës është aplikuar në Weka për të gjetur ato produkte që janë të lidhur me njeri - tjetrin.

7.3 MATLAB

MATLAB është një program që ofron një gjuhë me performancë shumë të lartë. Ai bën të mundur veprimet informatike, vizualizimin dhe programimin në një mjedis të thjeshtë ku problemet dhe zgjidhjet janë shprehur me simbole të thjeshta matematikore. MATLAB-in e përdorëm si framework për të ekzekutuar algoritmat gjenetike për probleme e optimizimit në sigurime.

7.4 AMBJENTI I TESTIMIT

Parametrat e kompjuterit në të cilin kemi bërë ekzekutimin e modeleve janë: HP me procesor Intel core Duo 2.4 Ghz; RAM 4 GB; OS Windows7 64 bit.

8- APLIKIMI DHE ANALIZA E TESTIMEVE

Në këtë kapitull do të paraqesim një vlerësim të gjerë eksperimental të algoritmave të modifikuar. Eksperimentet janë zhvilluar mbi disa baza të dhënash të marra nga një kompani shqiptare në sigurimin e jetës, me anë të të cilave është testuar efikasiteti i algoritmave të propozuar. Vlerësimi i performancës është bazuar në kritere të mirëpercaktuara.

Për realizimin e këtyre eksperimenteve janë përdorur funksionet dhe ndërhyrjet përkatëse. Struktura e përgjithshme e eksperimenteve është dhënë si më poshtë:

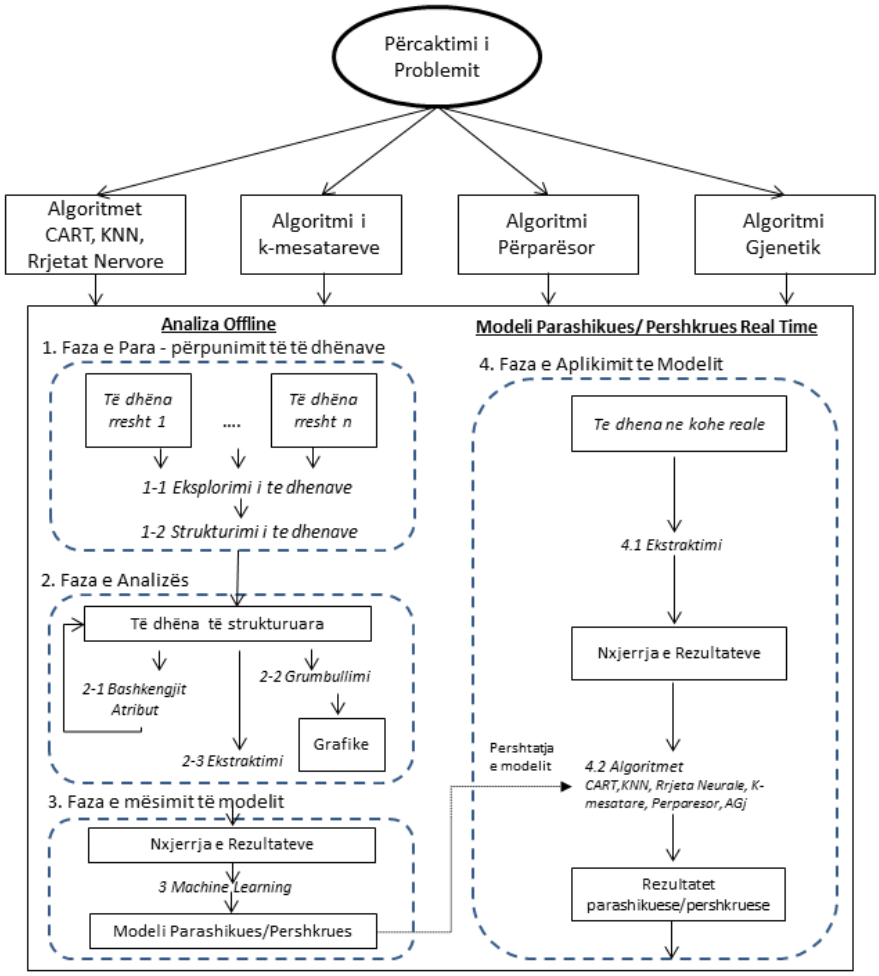


Figura 19 Struktura e organizimit të eksperimenteve

8.1 APLIKIMI I MODIFIKIMIT TË ALGORITMIT CART

Procedura e pemës së vendimit ofron disa metoda për krijimin e pemëve të vendimit. Metoda e përdorur në këtë rast është CART (Classification And Regression Trees). Në çdo hap CART zgjedh variablin e pavarur (parashikuesin), i cili ka ndërveprimin më të madh me variablin e varur. Kategoritë e çdo parashikuesi bashkohen nëse nuk janë shumë të ndryshëm sipas variablit të varur.
Të dhenat janë marrë nga një shoqëri sigurimi jete mbi klientët që janë siguruar pranë saj për një periudhë të caktuar. Grupi i synuar per tu analizuar në këto të dhëna janë klientët, të cilët kanë aplikuar për një sigurim jete. Vlerat e klasës i mirë ose i keq ju korrespondojnë respektivisht klientëve, të cilët janë pranuar ose mohuar për të marrë një sigurim jete.
Fillimisht do të analizojmë karakteristikat e klasës së dhënë dhe më pas do të ndërtojmë modelin për të parashikuar mundësinë e paracaktimit të vlerësimit për klientët e rinj. Atributet që janë perdorur paraqiten në tabelën 3 më poshtë:

Emri	Type	Vlerat	Njesia matese
Vleresimi i siguruesit	Numerike	0 (I keq); 1(I mire); 9 (pa histori ne sigurime)	Nominale
Mosha	Numerike	E vazhduar	Scale
Paga (Te ardhurat personale)	Numerike	1(I ulet); 2(Mesatar); 3(I larte)	Ordinal
Nr.Femijeve (numri i pjestarëve të familjes)	Numerike	1 (me pak se= 3); 2 (me shume se 3)	Nominale
Arsimi	Numerike	1 (I larte); 2 (I mesem)	Nominale
Gjinia	Numerike	1 (mashkull); 2 (femer)	Nominale

Tabela 3 Klasifikimi CART lista e atributeve

Kemi marrë në shqyrtim 2’464 rekorde dhe përdorëm programin IBM SPSS Statistics për të ndërtuar modelin dhe për të interpretuar rezultatet kemi perdorur IBM SPSS statistics Viewer.
Kemi zgjedhur kategorinë i keq si kategorinë target të interesit domethënë të gjithë klientët që kanë një vlerësim të keq nga siguruesi (nuk kanë aftësi paguese për primin) do të jenë objektivi ynë.

Tabela përmbledhëse e paraqitur më poshtë përmban informacion të gjerë përsa i përket specifikimeve të përdorura për ndërtimin e modelit.
Pjesa e specifikimeve jep informacion mbi karakteristikat e përdorura për të gjeneruar modelin e pemës duke përfshirë variablat e përdorur në analizë.
Ndërsa pjesa e rezultateve paraqet informacionin mbi numrin e nyjeve totale dhe terminale; lartësinë e pemës dhe variablat e pavarur të përfshirë në modelin përfundimtar.
Variablat që nuk kanë kontribut domethënës në modelin përfundimtar lihen jashtë tabelës përmbledhëse.

Specifications	Growing Method	CART
	Dependent Variable	Vleresimi i Siguruesit
	Independent Variables	Mosha, Niveli i pages, Numri i femijeve, Arsimi, Gjinia
	Validation	None
	Maximum Tree Depth	5
	Minimum Cases in Parent Node	400
	Minimum Cases in Child Node	200
Results	Independent Variables Included	Niveli i pages, Numri i femijeve, Gjinia, Mosha
	Number of Nodes	13
	Number of Terminal Nodes	7
	Depth	4

Tabela 4 Klasifikimi CART tabela përmbledhëse

Nga paraqitja e pemës se vendimit në tabelën 5shohim disa rezultate paraprake:
Duke perdorur algoritmin CART niveli i pagës është parashikuesi më i mirë për vlerësimin e klientit nga ana e siguruesit.
Për nivele pagash mesatare dhe të lartë parashikuesi më i mirë është numri i fëmijëve ose numri i pjestarëve të familjes.
82% e klientëve me nivel page mesatare dhe me kriter moshe më të vogël se 28 kanë një vlerësim të keq nga ana e siguruesit.
Gjithashtu në tabelën 5 janë paraqitur të gjitha nyjet e pemës me numrin dhe përqindjen për secilën kategori të variablit të varur.

Node	I keq		I mire		Total		Predicted Category	Parent Node
	N	Percent	N	Percent	N	Percent		
0	1020	41.40%	1444	58.60%	2464	100.00%	I mire	
1	454	82.10%	99	17.90%	553	22.40%	I keq	0
2	566	29.60%	1345	70.40%	1911	77.60%	I mire	0
3	502	41.90%	697	58.10%	1199	48.70%	I mire	2
4	64	9.00%	648	91.00%	712	28.90%	I mire	2
5	422	56.70%	322	43.30%	744	30.20%	I keq	3
6	80	17.60%	375	82.40%	455	18.50%	I mire	3
7	39	18.90%	167	81.10%	206	8.40%	I mire	4
8	25	4.90%	481	95.10%	506	20.50%	I mire	4
9	205	82.30%	44	17.70%	249	10.10%	I keq	5
10	217	43.80%	278	56.20%	495	20.10%	I mire	5
11	19	8.70%	199	91.30%	218	8.80%	I mire	8
12	6	2.10%	282	97.90%	288	11.70%	I mire	8

Tabela 5 Klasifikimi CART pema e vendimeve në formë tabelore

Në tabelën 6 janë paraqitur atributet sipas rëndësisë në modelin e krijuar, të cilët shërbejnë si kritere ndarëse në pemë. Kriteri i parë ndarës është niveli i pagës, më pas numri i fëmijëve dhe më pas mosha.

Tree Table			
Node	Primary Independent Variable		
	Variable	Improvement	Split Values
1	Niveli i pages	0.096	<= I Ulet
2	Niveli i pages	0.096	> I Ulet
3	Numri i femijeve	0.039	>3
4	Numri i femijeve	0.039	<=3
5	Niveli i pages	0.035	<= Mesatar
6	Niveli i pages	0.035	> Mesatar
7	Mosha	0.002	<= 30.243
8	Mosha	0.002	> 30.243
9	Mosha	0.02	<= 27.893
10	Mosha	0.02	> 27.893
11	Mosha	0	<= 37.359
12	Mosha	0	> 37.359

Tabela 6 Klasifikimi CART vlerat e kriterit ndarës

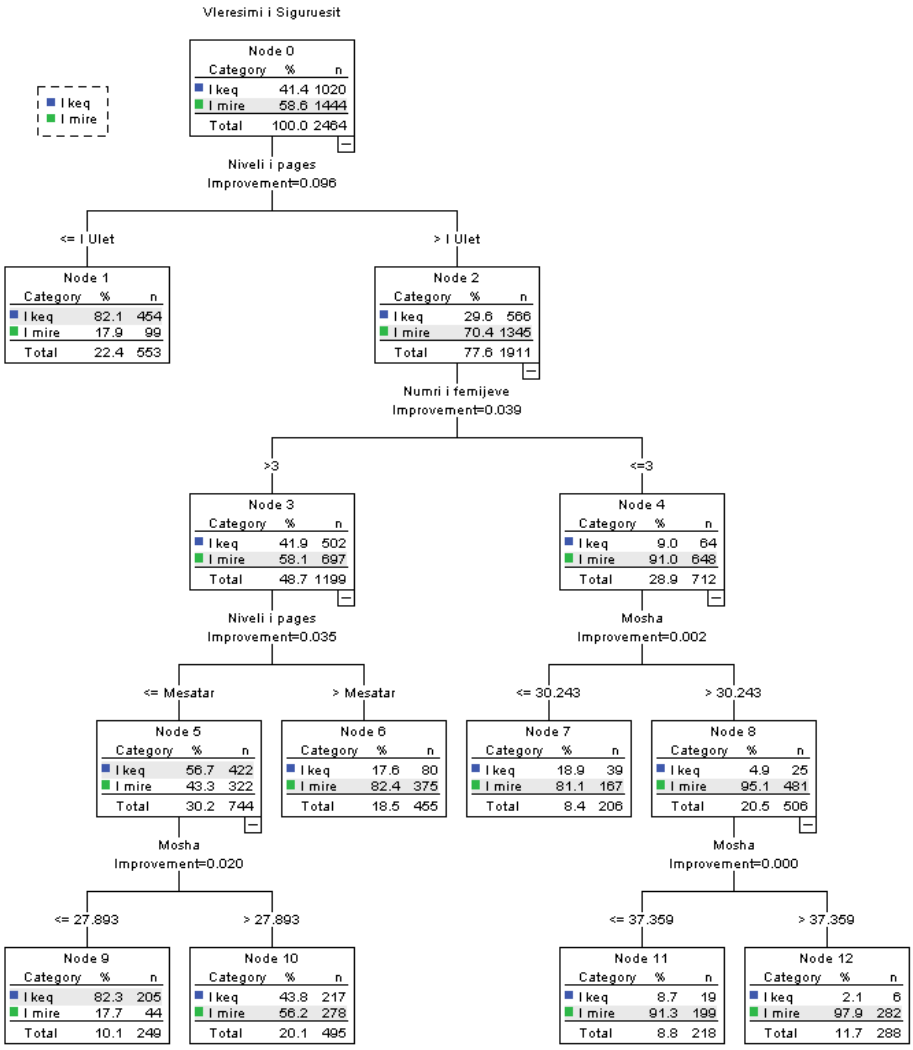
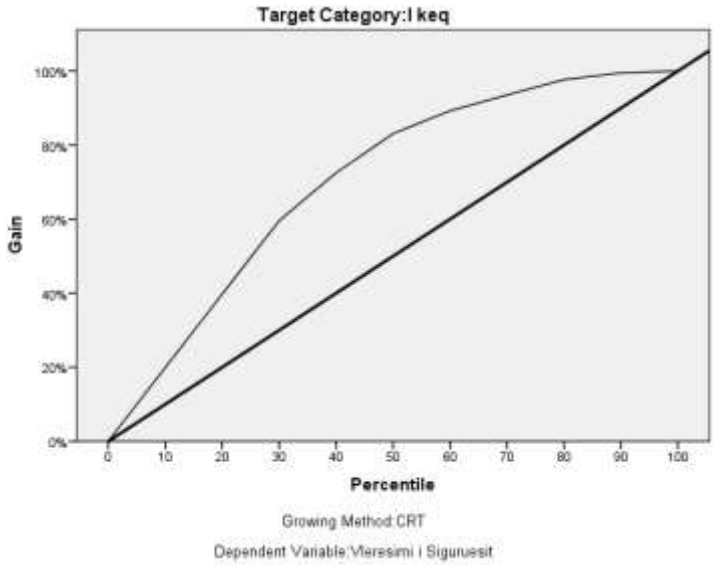


Figura 20 Klasifikimi CART paraqitja chartflow i pemës së vendimit

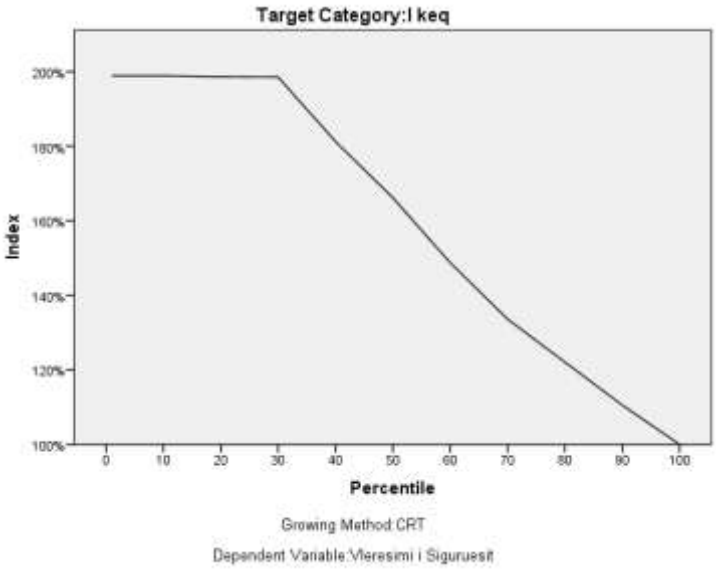
Gains for Nodes						
Node	Node		Gain		Response	Index
	N	Percent	N	Percent		
9	249	10.10%	205	20.10%	82.30%	198.90%
1	553	22.40%	454	44.50%	82.10%	198.30%
10	495	20.10%	217	21.30%	43.80%	105.90%
7	206	8.40%	39	3.80%	18.90%	45.70%
6	455	18.50%	80	7.80%	17.60%	42.50%
11	218	8.80%	19	1.90%	8.70%	21.10%
12	288	11.70%	6	0.60%	2.10%	5.00%

Tabela 7 Klasifikimi CART Perfitimi i vlerave për çdo nyje



Grafiku 1 Klasifikimi CART Grafiku Gain per target kategorine: I keq

Grafiku Gain tregon se modeli është i mjaft i mirëpasi vlerat kumulative fillojnë nga 0% dhe përfundojnë në 100%.



Grafiku 2 Klasifikimi CART grafiku Index per target kategori: I keq

Një vlerë indeksi më e madhe se 100% do të thotë se ka më shumë raste në target kategorinë se përqindja e përgjithshme e target kategorisë. Në anën tjetër, një vlerë

indeksi më e vogel se 100% do të thotë që ka më pak raste në target kategorinë se përqindja e përgjithshme.
Modeli për vlerësimin e klasës target i keq është vlerësuar edhe nga grafiku 2, i cili tregon se modeli është i mirë. Vlerat kumulative kanë tendencë për të filluar mbi 100% dhe gradualisht zbresin deri sa të arrijë në 100%.
Sipas analizës së kostos, e cila rezulton me vlerë 0.205 shohim se kategoria e parashikuar nga modeli është e gabuar vetëm në 20.5% të rasteve. "Risku" për klasifikim të gabuar për një klient te ri ne sigurime është përafërsisht 20.5%. Rezultatet në tabelën e klasifikimit janë në përputhje me vlerësimin e rrezikut. Tabela tregon se modeli i klasifikon rreth 79.5% e klientëve saktë.

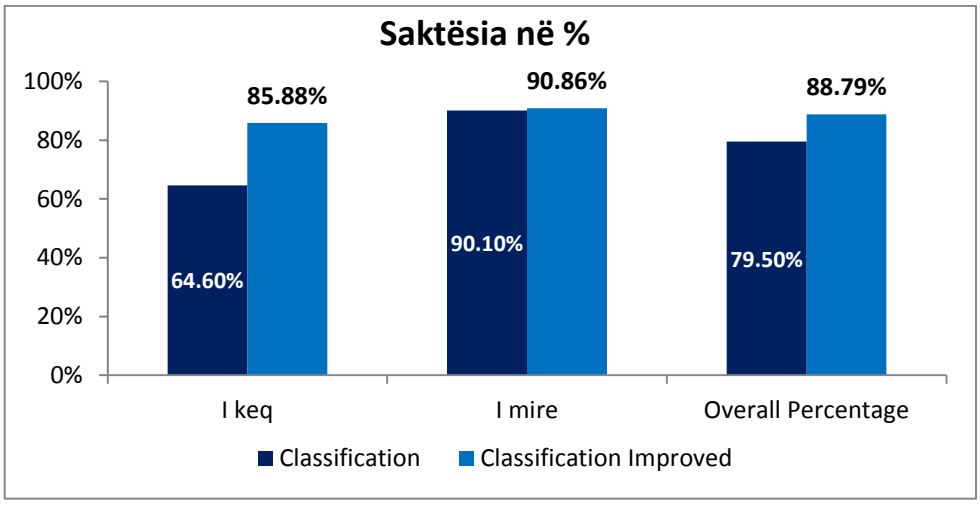
Classification			
Observed	Predicted		
	I keq	I mire	Percent Correct
I keq	659	361	64.60%
I mire	143	1301	90.10%
Overall Percentage	32.50%	67.50%	79.50%

Tabela 8 Klasifikimi CART vlerat e klasifikimit sipas metodes standarte

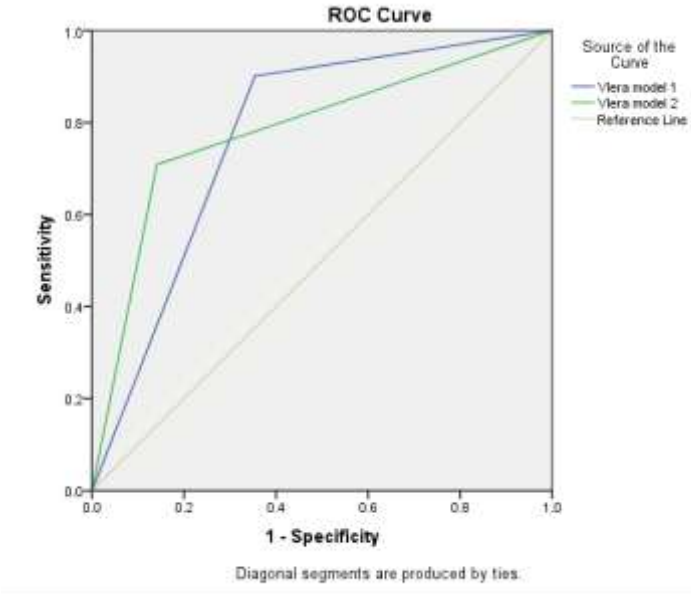
Në modelin e ndërtuar sipas tabelës se klasifikimit te mesiperme rezulton një problem:
Për ata klientë me një vlerësim nga siguruesi të keq, ajo parashikon një vlerësim të keq vetëm për 64.6% të tyre, që do të thotë se 34.4% e klienteve me një vlerësim të keq nga siguruesi janë të pasaktë dhe janë klasifikuar si klientë të "mirë".
Duke ditur që vlerat e variablit vlerësimi i siguruesit ka dy vlera 0-i keq dhe 1 – i mirë, shtojmë një vlerë tjetër për të gjithë klientët e vlerësuar të këqinj nga siguruesi (vlera reale) dhe të mirë nga modeli i ndërtuar (vlera e parashikuar).
Rezultati tregon se modeli i përmirësuar i klasifikon rreth 88.8% të klientëve saktë dhe për ata klientë me një vlerësim nga siguruesi të keq rezultati është përmirësuar nga 64.6% në 85.9% të klasifikuar saktë.

Classification Improved			
Observed	Predicted		
	I keq	I mire	Percent Correct
I keq	876	144	85.88%
I mire	132	1312	90.86%
Overall Percentage	40.9%	59.1%	88.79%

Tabela 9 Klasifikimi CART vlerat e klasifikimit sipas modelit të përmirësuar



Grafiku 3 Klasifikimi CART Saktësia në % për të dy metodat (metodës standarte dhe metodës së përmirësuar)



Grafiku 4 Klasifikimi CART grafiku ROC, modeli 1 – metoda standarte, modeli 2 – metoda e përmirësuar

Sipas kriterit Grafiku ROC shohim se të dy modelet janë të mirë pasi siperfaqet e zonës ndodhen midis vlerave 0.8<0.814<0.884<0.9 dhe modeli i propozuar rezulton më i mire se modeli standart për të klasifikuar të dhënat në sigurime.

8.2 ANALIZA KNN NËPËRMJET ZGJEDHJES SË LLOGARITJES SË LARGËSISË

Analiza KNN është një metodë për klasifikimin e rasteve bazuar në ngjashmërinë e tyre me rastet e tjera. Në ML, ajo u zhvillua si një mënyrë për të njohur modelet e të dhënave pa kërkuar një krahasim për çdo rast. Rastet e ngjashme janë pranë njëri-tjetrit dhe rastet jo të ngjashme janë larg nga njëri-tjetrit. Kështu largësia mes dy rasteve është një masë e pangjashmërisë së tyre. Rastet që janë pranë njëri-tjetrit janë quajtur "fqinjë". Kur një rast i ri shfaqet, llogaritet largësia e tij nga të gjithë rastet e tjera në model. Klasifikimet e rasteve më të ngjashme - fqinjët më të afërt - shenohen dhe rast i ri vendoset në kategorinë që përmban numrin më të madh të fqinjëve më të afërt.

Në bazën e të dhënave për klientët që janë siguruar pranë saj shoqeria e sigurimeve gjithashtu mban edhe te dhenat mbi pyetësorin shëndetësor. Modeli që do të ndërtojmë me anë të algoritmin KNN në IBM SPSS Statistics v.20 përcakton se kush është mënyra më e mirë për llogaritjen e largësisë dhe cilët janë atributet më me rëndësi që ndikojnë në vlerësimin e siguruesit ndikuar nga të dhënat e klientit. Janë proçesuar 70 raste të vlefshme.

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	Riskfaktor	Numeric	4	0	Vleresues Risku	None	None	11	Right	Nominal	Input
2	Mosha	Numeric	4	0	Mosha ne Vite	None	None	5	Right	Scale	Input
3	Gjinia	Numeric	4	0	Gjinia	{0, Mashkull...	None	6	Right	Nominal	Input
4	Gjatesia	Numeric	5	1	Gjatesia ne cm	None	None	6	Right	Scale	Input
5	Pesha	Numeric	5	1	Pesha ne kg	None	None	6	Right	Scale	Input
6	Duhanpires	Numeric	4	0	Pi Duhan	{0, jo duhan...	None	9	Right	Nominal	Input
7	Szemar	Numeric	4	0	Semundje Zemra	{0, jo}...	None	6	Right	Nominal	Input
8	STumor	Numeric	4	0	Semundje Tumori	{0, jo}...	None	6	Right	Nominal	Input
9	SGjaku	Numeric	4	0	Semundje Gjaku	{0, jo}...	None	6	Right	Nominal	Input
10	SMushkeri	Numeric	4	0	Semundje Mushkeri	{0, jo}...	None	6	Right	Nominal	Input
11											

Tabela 10 Klasifikimi KNN Lista e atributeve

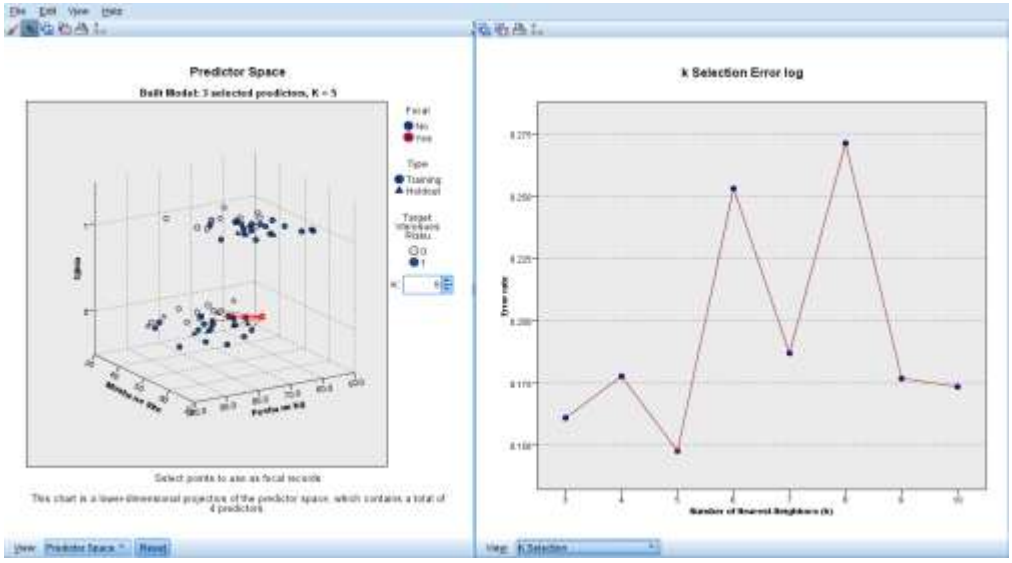
8.2.1 MODELI I - SIPAS LARGËSISË EUKLIDIANE TË PONDERUAR

Sintaksa e Klasifikimit KNN sipas largësisë euklidiane të ponderuar është paraqitur si më poshtë:

```
*Nearest Neighbor Analysis
KNN Riskfaktor (MLEVEL=N) BY Gjinia Duhanpires Szemer STumor SGjaku SMushkeri Mosha
Pesha
  /RESCALE COVARIATE=ADJNORMALIZED
```

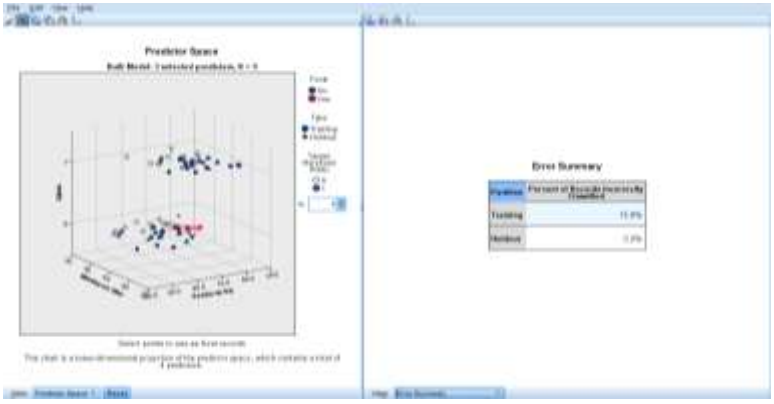
```
/MODEL NEIGHBORS=AUTO (KMIN=3, KMAX=10) METRIC=EUCLID FEATURES=ALL
/CRITERIA WEIGHTFEATURES=YES
/PARTITION TRAINING=70 HOLDOUT=30
/CROSSVALIDATION FOLDS=10
/PRINT CPS /VIEWMODEL DISPLAY=YES
/MISSING USERMISSING=EXCLUDE.
```

Atributetet më me rëndësi në këtë model janë Mosha, Pesha dhe Gjinia tre dimensione e grafikut në figurën 21 (Predictor Space).



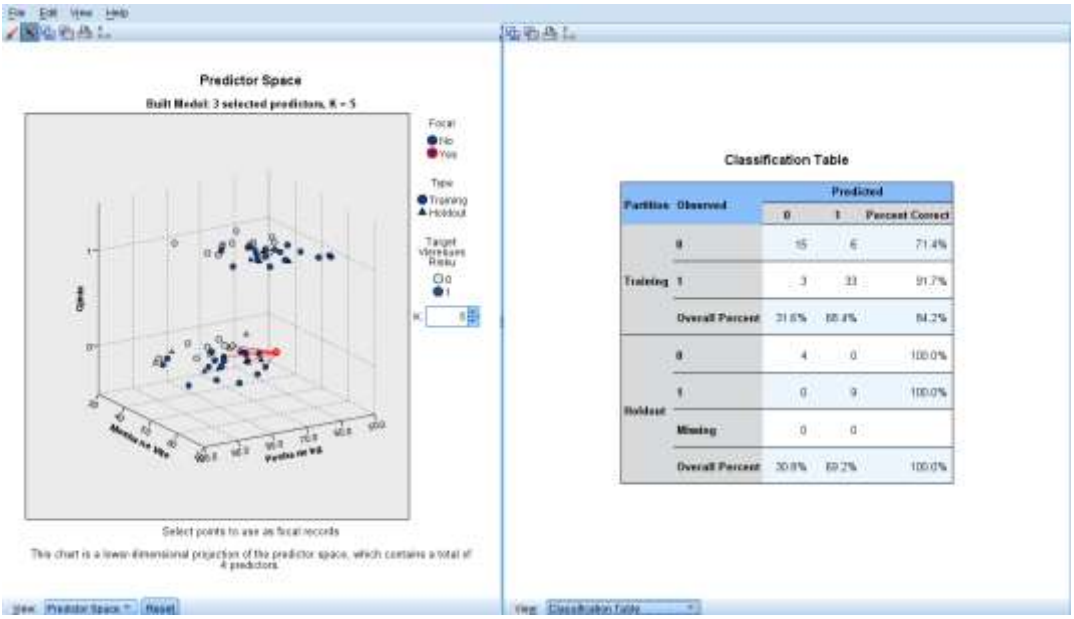
Grafiku 5 Klasifikimi KNN zgjedhja e k-se per modelin I Sipas largësisë euklidiane të ponderuar

Pikat në figurën 21 të mësipërme (k-Selection error log) paraqesin shkallën e gabimit (boshti y) të modelit në varësi të numrit të fqinjëve më të afërt (boshti x). Modeli për k=5 fqinjë më të afërt ka shkallën më të ulët të gabimit. Për k të ndryshëm nga 5 shkalla e gabimit per modelin rritet në mënyrë dramatike.



Grafiku 6 Klasifikimi KNN shkalla e gabimit per modelin I Sipas largësisë euklidiane të ponderuar

Përqindja e vlerave që janë klasifikuar gabim ne modelin I është për të dhënat training 15.8% dhe për të dhënat holdout 0%. Tabela e mëposhtme tregon klasifikimin e vlerave të vëzhguara kundrejt vlerave të parashikuara të atributit target "vlerësues risku" për çdo vlerë të tij. Saktësia e klasifikimit është për të dhënat training 84.2% dhe për të dhënat holdout 100%.



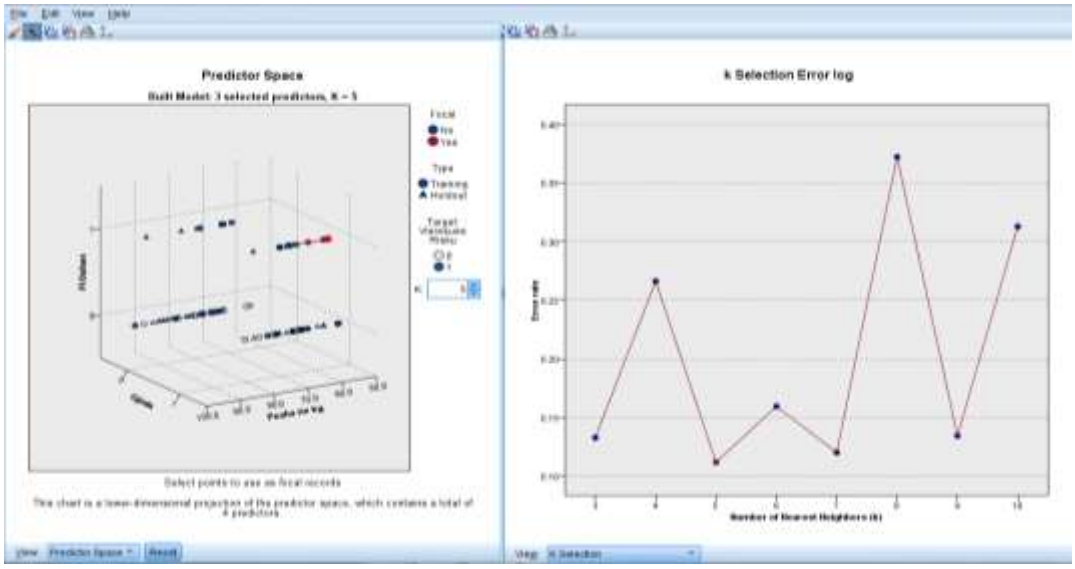
Grafiku 7 Klasifikimi KNN tabela e klasifikimit per modelin I Sipas largësisë euklidiane të ponderuar

8.2.2 MODELI II - SIPAS LARGËSISË EUKLIDIANE TË THJESHTË

Sintaksa e Klasifikimit KNN sipas largësisë euklidiane të thjeshtë është paraqitur si më poshtë:

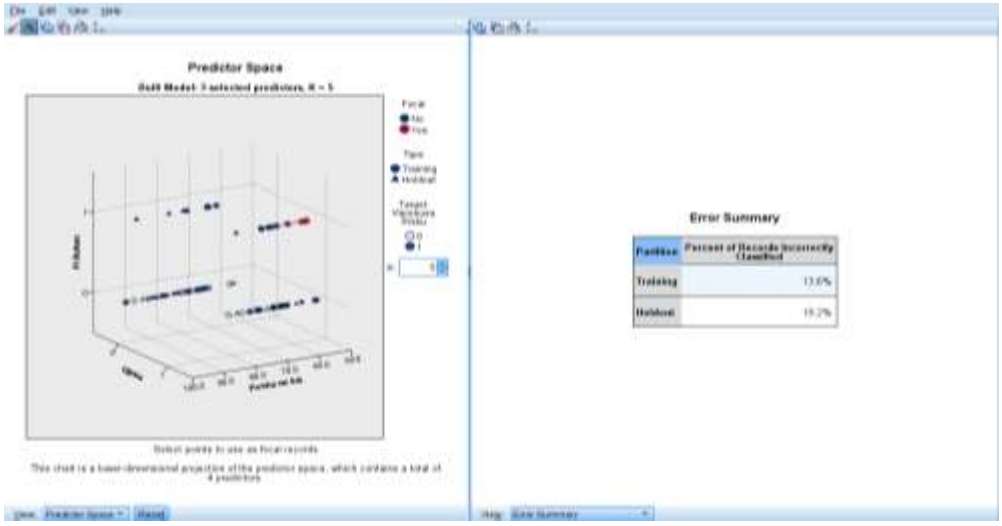
```
*Nearest Neighbor Analysis.
KNN Riskfaktor (MLEVEL=N) BY Gjinia Duhanpires Szemer STumor SGjaku SMushkeri Mosha
Pesha
  /RESCALE COVARIATE=ADJNORMALIZED
  /MODEL NEIGHBORS=AUTO (KMIN=3, KMAX=10) METRIC=EUCLID FEATURES=ALL
  /CRITERIA WEIGHTFEATURES=NO
  /PARTITION TRAINING=70  HOLDOUT=30
  /CROSSVALIDATION FOLDS=10
  /PRINT CPS
  /VIEWMODEL DISPLAY=YES
  /MISSING USERMISSING=EXCLUDE.
```

Atributetet më me rëndësi ne kete model janë Pesha, Gjinia dhe Pi duhan tre dimensione e grafikut të mëposhtëm.



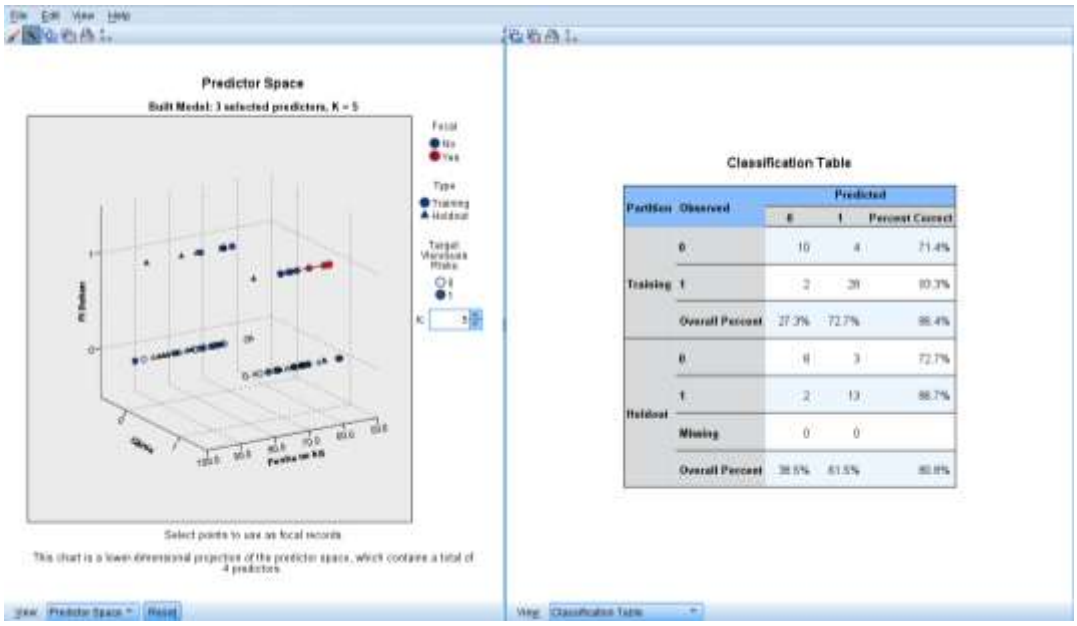
Grafiku 8 Klasifikimi KNN zgjedhja e k-se per modelin II sipas largësisë euklidiane të thjeshtë

Pikët në grafikun e mësipërm (k-Selection error log) paraqesin shkallën e gabimit (boshti y) të modelit në varësi të numrit të fqinjëve më të afërt (boshti x). Modeli për k=5 fqinjë më të afërt ka shkallën më të ulët të gabimit. Përqindja e vlerave që janë klasifikuar gabim në modelin II është për të dhënat training 13.6% dhe për të dhënat holdout 19.2%.



Grafiku 9 Klasifikimi KNN shkalla e gabimit për modelin II sipas largësisë euklidiane të thjeshtë

Tabela e mëposhtme tregon klasifikimin e vlerave të vëzhguara kundrejt vlerave të parashikuara të atributit target "vlerësues risku" për çdo vlerë të tij. Saktësia e klasifikimit është për të dhënat training 86.4% dhe për të dhënat holdout 80.8%.



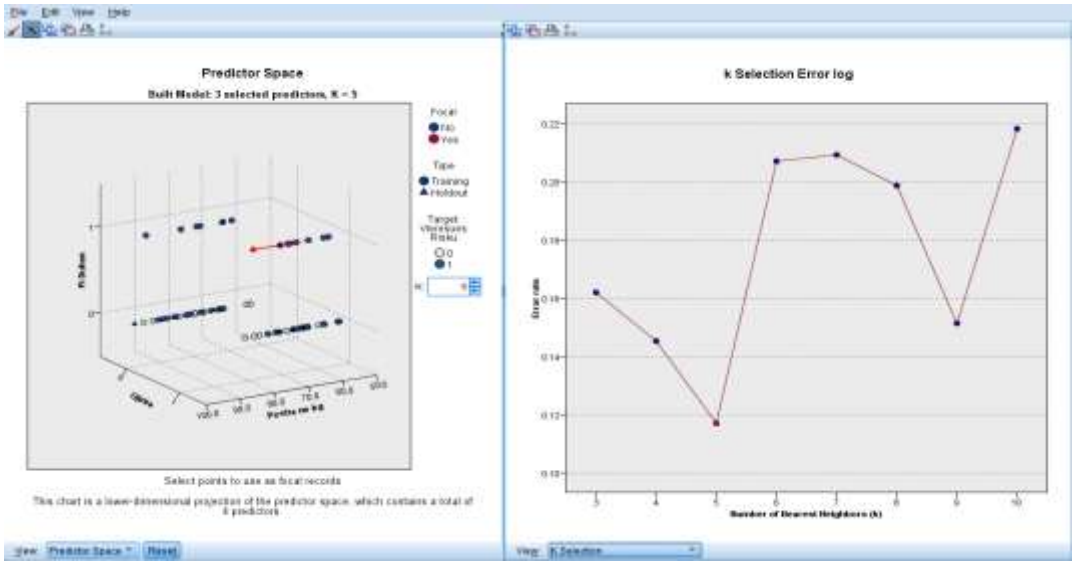
Grafiku 10 Klasifikimi KNN tabela e klasifikimit për modelin II sipas largësisë euklidiane të thjeshtë

8.2.3 MODELI III-SIPAS LARGËSISË MANHATAN TË THJESHTË

Sintaksa e Klasifikimit KNN sipas largësisë Manhatan të thjeshtë eshte paraqitur si më poshtë:

```
*Nearest Neighbor Analysis.
KNN Riskfaktor (MLEVEL=N) BY Gjinia Duhanpires Szemer STumor SGjaku SMushkeri Mosha
Pesha
  /RESCALE COVARIATE=ADJNORMALIZED
  /MODEL NEIGHBORS=AUTO(KMIN=3, KMAX=10) METRIC= CITYBLOCK
FEATURES=ALL
  /CRITERIA WEIGHTFEATURES=NO
  /PARTITION  TRAINING=70  HOLDOUT=30
  /CROSSVALIDATION FOLDS=10
  /PRINT CPS
  /VIEWMODEL DISPLAY=YES
  /MISSING USERMISSING=EXCLUDE.
```

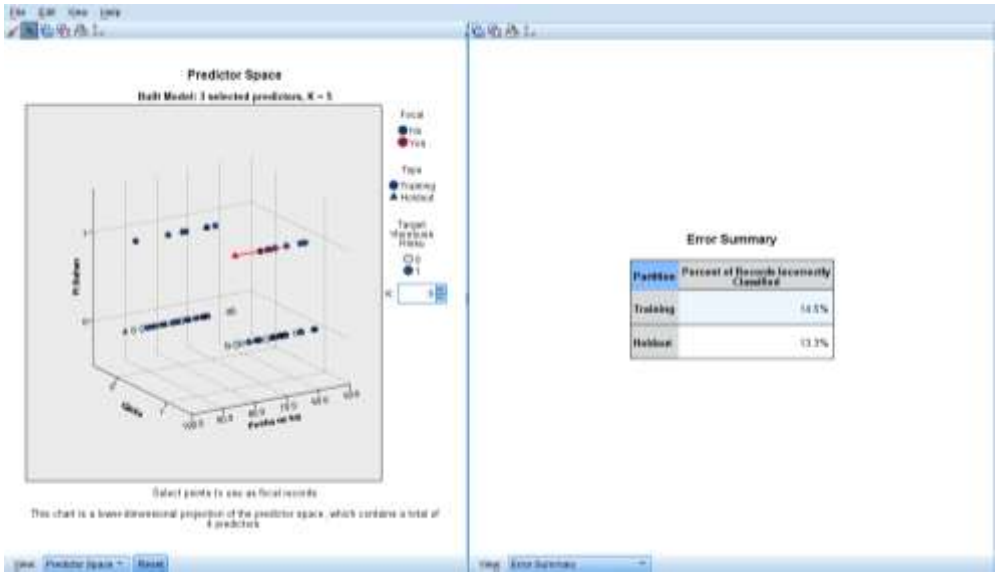
Atributetet më me rëndësi në këtë model janë Pesha, Gjinia dhe Pi duhan tre dimensione e grafikut të mëposhtëm.



Grafiku 11 Klasifikimi KNN zgjedhja e k-se per modelin III sipas largësisë manhatan të thjeshtë

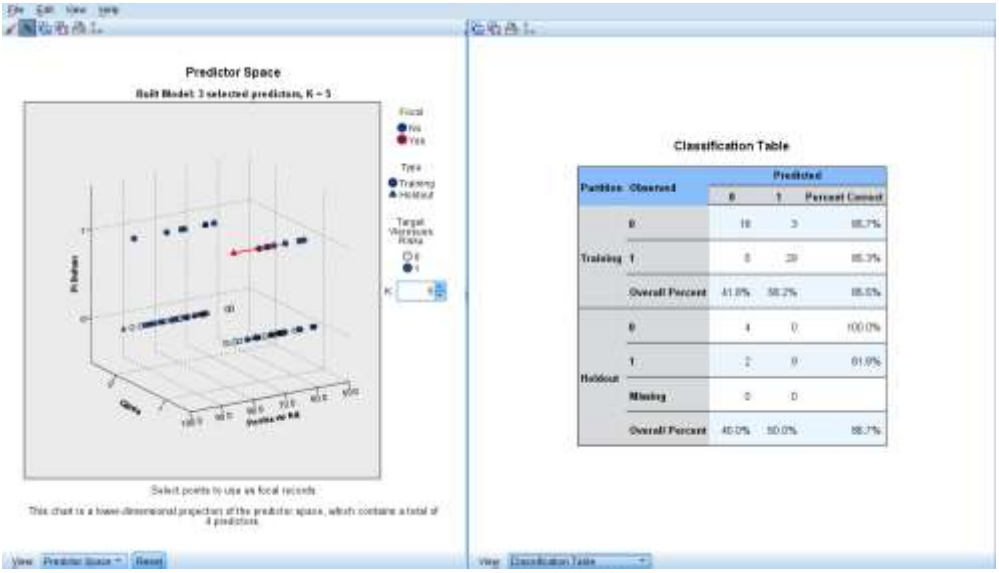
Pikët në grafikun e mësipërm (k-Selection error log) paraqesin shkallën e gabimit (boshti y) të modelit në varësi të numrit të fqinjëve më të afërt (boshti x). Modeli për k=5 fqinjë më të afërt ka shkallën më të ulët të gabimit.

Përqindja e vlerave që janë klasifikuar gabim në modelin III është për të dhënat training 14.5% dhe për të dhënat holdout 13.3%.



Grafiku 12 Klasifikimi KNN shkalla e gabimit per modelin III sipas largësisë manhatan të thjeshtë

Tabela e mëposhtme tregon klasifikimin e vlerave të vëzhguara kundrejt vlerave të parashikuara të atributit target "vlerësues risku" për çdo vlerë të tij. Saktësia e klasifikimit është për të dhënat training 85.5% dhe për të dhënat holdout 86.7%.



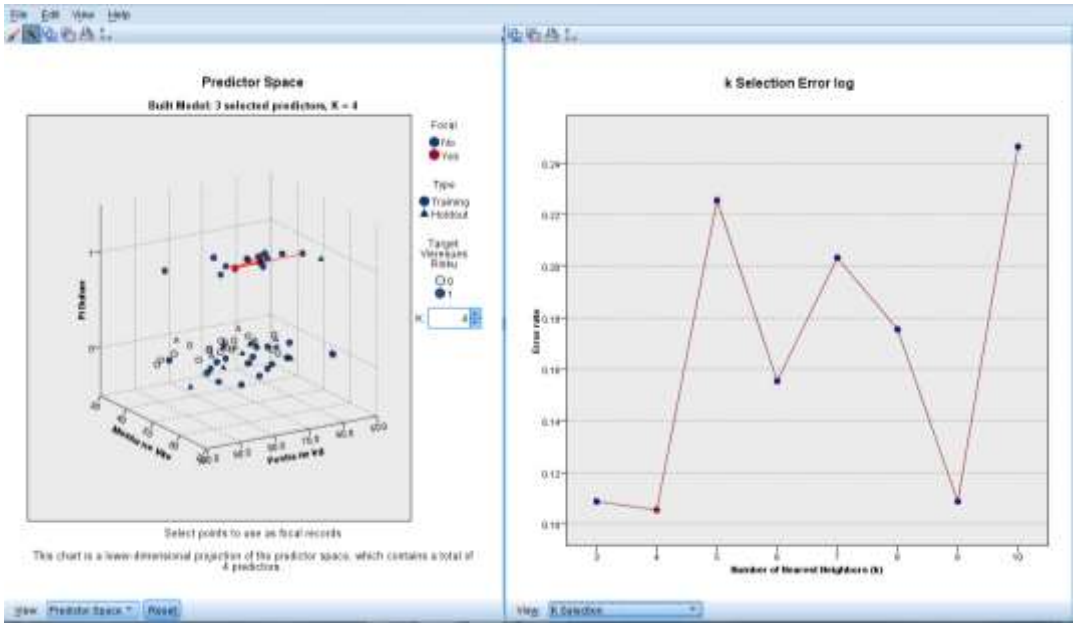
Grafiku 13 Klasifikimi KNN tabela e klasifikimit per modelin III sipas largësisë manhatan të thjeshtë

8.2.4 MODELI IV- SIPAS LARGËSISË MANHATAN TË PONDERUAR

Sintaksa e Klasifikimit KNN sipas largësisë manhatan të ponderuar eshte paraqitur si me poshte:

```
*Nearest Neighbor Analysis.
KNN Riskfaktor (MLEVEL=N) BY Gjinia Duhanpires Szemer STumor SGjaku SMushkeri  Mosha Pesha
  /RESCALE COVARIATE=ADJNORMALIZED
  /MODEL NEIGHBORS=AUTO (KMIN=3, KMAX=10) METRIC= CITYBLOCK
FEATURES=ALL
  /CRITERIA WEIGHTFEATURES=YES
  /PARTITION TRAINING=70 HOLDOUT=30
  /CROSSVALIDATION FOLDS=10
  /PRINT CPS
  /VIEWMODEL DISPLAY=YES
  /MISSING USERMISSING=EXCLUDE.
```

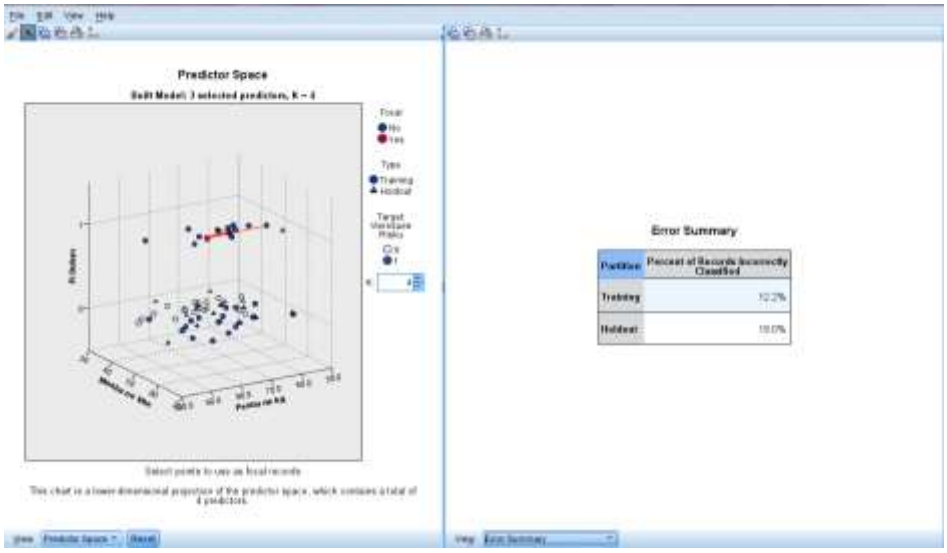
Atributetet më me rëndësi në këtë model janë Pesha, Gjinia dhe Pi duhan tre dimensione e grafikut të mëposhtëm.



Grafiku 14 Klasifikimi KNN zgjedhja e k-se per modelin IV sipas largësisë manhatan të ponderuar

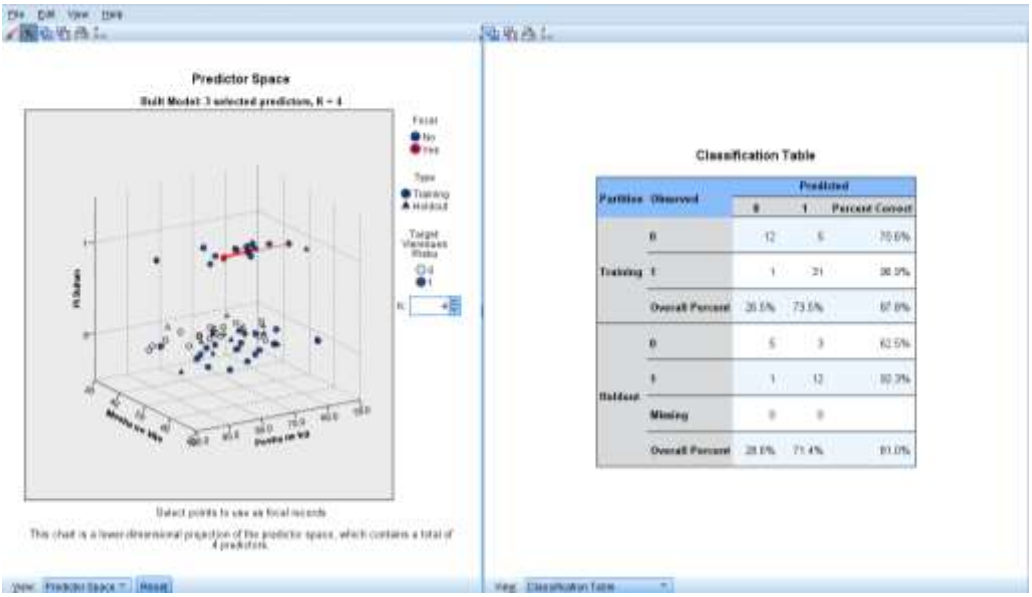
Pikët në grafikun e mësipërm (k-Selection error log) paraqesin shkallën e gabimit (boshti y) të modelit në varësi të numrit të fqinjëve më të afërt boshti x. Modeli për k=4 fqinjë më të afërt ka shkallën më të ulët të gabimit.

Përqindja e vlerave që janë klasifikuar gabim në modelin IV është për të dhënat training 12.2% dhe për të dhënat holdout 19%.



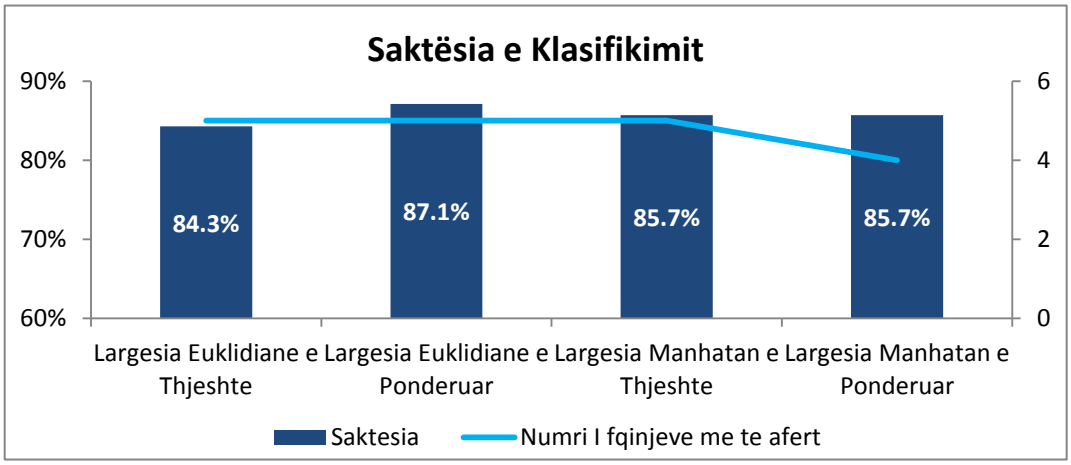
Grafiku 15 Klasifikimi KNN shkalla e gabimit per modelin IV sipas largësisë manhatan të ponderuar

Tabela e mëposhtme tregon klasifikimin e vlerave të vëzhguara kundrejt vlerave të parashikuara të atributit target "vlerësues risku" për çdo vlerë të tij. Saktësia e klasifikimit është për të dhënat training 87.8% dhe për të dhënat holdout 81%.



Grafiku 16 Klasifikimi KNN tabela e klasifikimit per modelin IV sipas largësisë manhatan të ponderuar

Për të dhënat numerike të analizuara nëpërmjet algoritmit të fqinjësisë më të afert KNN modeli me saktesi më të madhe rezulton sipas largesisë Euklidiane të ponderuar me 87.1% dhe me numer fqinjësh më të afërt 5.



Grafiku 17 Klasifikimi KNN Saktësia e klasifikimit për modelet I, II, III dhe IV

8.3 KLASIFIKIMI NEPERMJET RRJETAVE NERVORE NE SIGURIME NJE METODE ALTERNATIVE

Një shoqëri sigurimi jete duhet të jetë në gjendje të identifikojë karakteristikat e klientëvë që janë të siguruar dhe ti përdor këto karakteristika për të identifikuar klintët e rinj për sigurim nëse janë me risk të lartë ose të ulët.
Baza e të dhënave qe do te analizojme përmban informacion për 1’050 klientë ekzistues. Klientët të cilët kishin më parë sigurime do te përdoren si një tuple e rastit për të krijuar një perceptron multilayer, ndërsa klientët e mbetur do të përdoren për të vërtetuar analizën duke përdorur modelin për ti klasifikuar klientët e ardhshëm në sigurime me risk të lartë dhe të ulët.
Fillimisht vendosim pasardhësit e rastit duke gjeneruar numra rasti. Më pas vendosim një kriter ndarës për të saktësuar bashkësinë provë (training) dhe bashkësinë mbështetesë (holdout).
Multilayer Perceptron Network prodhon një model parashikues për një ose më shumë variabla të varura bazuar në vlerat e variablave parashikuese. Sintaka e klasifikimit nepermjet rrjetave nervore në IBM SPSS statistics është si më poshtë:

```
*Multilayer Perceptron Network.
MLP default (MLEVEL=N) BY Edukimi WITH Mosha Vjetersia Adresa Teardhura Shpenzime
Kredi Shp_tjera
/RESCALE COVARIATE=STANDARDIZED
/PARTITION  VARIABLE=partition
/ARCHITECTURE   AUTOMATIC=YES (MINUNITS=1 MAXUNITS=50)
/CRITERIA TRAINING=BATCH OPTIMIZATION=SCALEDCONJUGATE
LAMBDAINITIAL=0.0000005 SIGMAINITIAL=0.00005 INTERVALCENTER=0
INTERVALOFFSET=0.5 MEMSIZE=1000
/PRINT CPS NETWORKINFO SUMMARY CLASSIFICATION IMPORTANCE
/PLOT NETWORK ROC GAIN LIFT PREDICTED
/STOPPINGRULES ERRORSTEPS= 1 (DATA=AUTO) TRAININGTIMER=ON
(MAXTIME=15) MAXEPOCHS=AUTO ERRORCHANGE=1.0E-4 ERRORRATIO=0.0010
```

Tabela e mëposhtme jep informacion mbi rrjetën dhe është e rëndësishme sepse siguron që specifikimet janë të sakta. Numri i shtresave në hyrje është numri i kovariateve dhe disa faktorëve. Krijohen njësi të veçanta për çdo kategori Niveli i Edukimit, Mosha ne vite, Vjetersia ne punen aktuale, Vjetersia ne adresen aktuale, Te ardhurat ne mije ALL, Shpenzimet ne mije ALL, Sigurime ne mije ALL, Shpenzime te tjera, ku asnjë kategori nuk konsiderohet e padobishme.
Gjithashtu një njësi dalje krijohet për cdo kategori Histori ne Sigurime më parë duke pasur keshtu rezultat 2 njësi dalëse.

Network Information

Input Layer	Factors	1	Niveli i Edukimit
	Covariates	1	Mosha ne Vite
		2	Vjetersia ne punen aktuale
		3	Vjetesia ne adresen aktuale
		4	Te ardhurat ne mije ALL
		5	Shpenzimet ne mije ALL
		6	Sigurime ne mije ALL
		7	Shpenzime te tjera
	Number of Units[a]		12
	Rescaling Method for Covariates		Standardized
Hidden Layer(s)	Number of Hidden Layers		1
	Number of Units in Hidden Layer 1[a]		3
	Activation Function		Hyperbolic tangent
Output Layer	Dependent Variables	1	Histori ne Sigurime me pare
	Number of Units		2
	Activation Function		Softmax
	Error Function		Cross-entropy

a. Excluding the bias unit

Tabela 11 Klasifikimi NN tabela Informacioni i rrjetit

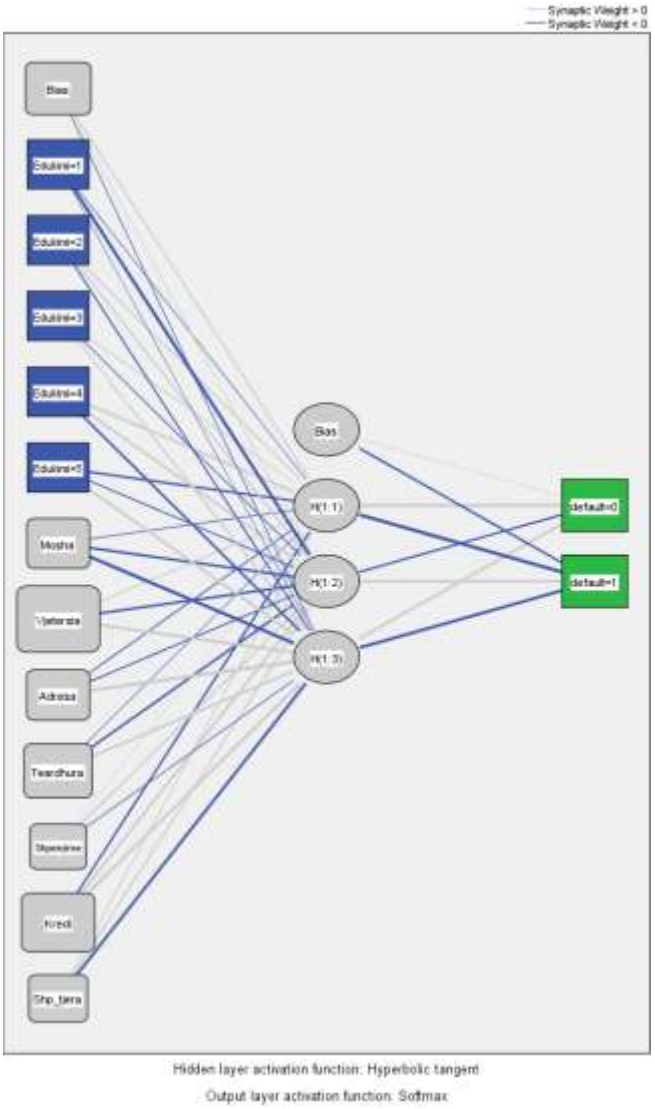
Tabela informacioni i rrjetit tregon informacion në lidhje me rrjetat nervore dhe siguron që specifikimet janë të sakta.

Training	Cross Entropy Error	179.429
	Percent Incorrect Predictions	16.2%
	Stopping Rule Used	Maximum number of epochs (100) exceeded
	Training Time	0:00:00.44
Holdout	Percent Incorrect Predictions	20.9%

Dependent Variable: Histori ne Sigurime me pare

Tabela 12 Klasifikimi NN Përmbledhja e modelit

Përmbledhja e modelit (tabela 12) tregon informacion në lidhje me rezultatet e trajnimit (training) dhe aplikimin e rrjetit përfundimtar të kampionit nga mbështetësit (Holdout). Cross Entropy Error paraqitet në këtë tabelë, sepse shtresa dalëse përdorin funksionin softmax, i cili është funksion gabimi që rrjeti përpiqet për ta minimizuar gjatë trajnimit. Përqindja e parashikimeve të pasakta është marrë nga tabela e klasifikimit.



Grafiku 18 Klasifikimi NN Struktura e rrjetit Diagrame

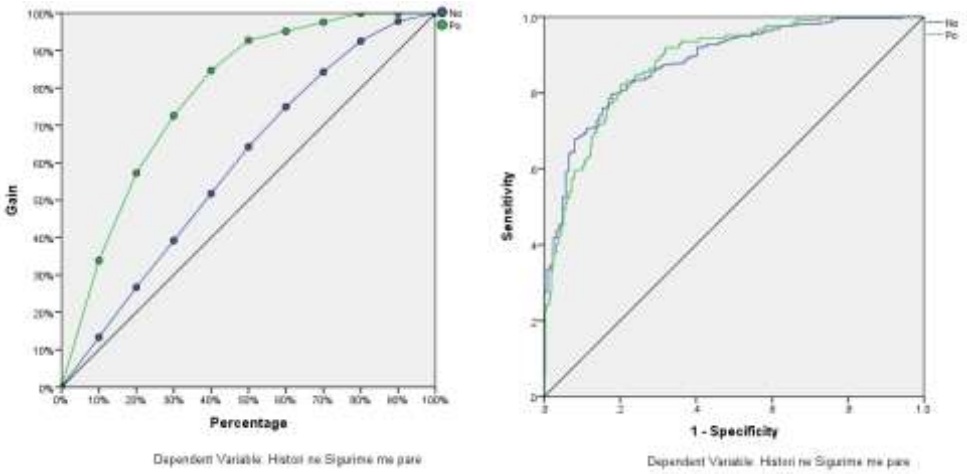
Classification

Sample	Observed	Predicted		
		No	Po	Percent Correct
Training	No	346	29	92.3%
	Po	52	72	58.1%
	Overall Percent	79.8%	20.2%	83.8%
Holdout	No	127	15	89.4%
	Po	27	32	54.2%
	Overall Percent	76.6%	23.4%	79.1%

Dependent Variable: Histori ne Sigurime me pare

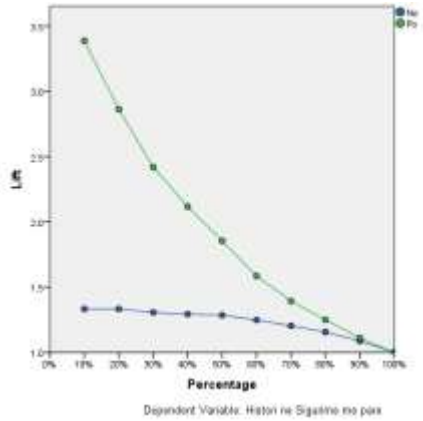
Tabela 13 Klasifikimi NN rezultatet e klasifikimit

Tabela e klasifikimit tregon rezultatet praktike të përdorimit të rrjetit. Për secilin rast, përgjigja e parashikuar është kategoria me pseudo-probabilitetin e parashikuar më të madh. Qelizat në diagonale janë parashikimet e sakta, ndërsa ato jashtë diagonales janë jo të sakta. Kështu me të dhënat e vëzhguara, modeli pa parashikues do t'i klasifikonte saktë të gjithë klientët në 82.4% të rasteve.



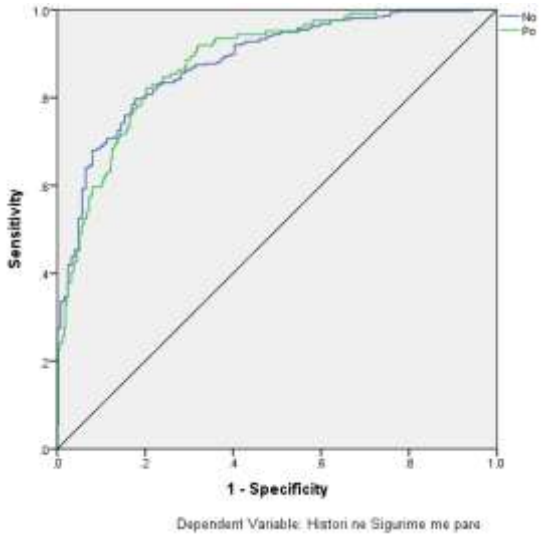
Grafiku 19 Klasifikimi NN grafiku i perfitimit

Grafiku i perfitimit (Gain) tregon përqindjen e rasteve në një kategori të caktuar të "fituar" duke synuar një përqindje të numrit të përgjithshëm të rasteve. Për shembull, pika e parë në kurbë për kategorinë "Ka patur histori ne sigurime me pare" ndodhet përafërsisht në kordinatat (10%, 30%) ç'ka nënkupton se nëse rendisim të gjitha rastet në bazë të pseudoprobabiliteti të parashikuar të kategorisë përkatëse, mund të presim që 10% i parë të përmbajë rreth 30% të të gjitha rasteve që kanë patur histori në sigurime.



Grafiku 20 Klasifikimi NN grafiku i njësisë matëse lift

Vlerat në boshtin y i përgjigjen raportit të fitimit për cdo kurbë me diagonalen. Kështu lift-i në 10% për kategorinë ka pasur histori ne sigurime është rreth 30%/10%=3.0.



Grafiku 21 Klasifikimi NN grafiku ROC

Nga grafiku ROC i mësipërm rezulton se modeli i ndërtuar është i mirë dhe sipërfaqja e zonës nën kurbë është 0.881. Në disa raste rrjetat nervore japin një gabim më të vogël klasifikimi sesa pemët e vendimit, por kërkojnë më shumë kohë mësimi. Një aspekt tjetër pozitiv i rrjetave nervore është pandjeshmëria ndaj zhurmave. Rregullat e klasifikimit të nxjerrë nga rrjetat nervore kanë një nivel gabimi të krahasueshëm me pemët e vendimit.

8.4 ANALIZA E GRUPIMIT ME ANË TË K-MESATAREVE

Të dhënat e përdorura në eksperimente janë marrë nga një kompani sigurimi jete. Për përzgjedhjen e të dhënave kemi përdorur bashkësi të ndryshme të dhënash, kemi përshkruar llojet e atributeve dhe numrin e rasteve të depozituara. Në tabelën e mëposhtme kemi paraqitur të gjitha karakteristikat e të dhënave të cilat janë përdorur në eksperimentet e modeleve të ndërtuara.

Tipi i file-it	Nr.atribute	Nr.rasteve	Baza e te dhenave	Vlera qe mungojne
CSV(comma separated value)	9	16’831	Multivariat	Po

Nr	Emri	Tipi i te dhenes	Emri i plote	Pershkrimi
Attribute 1	ID	numerical	Numer identifikues per rastin	
Attribute 2	Gjinia	qualitative	Gjinia e personit qe do te sigurohet	M-mashkull and F-femer
Attribute 3	Shumasig	numerical	Shuma qe do te sigurohet	vlerat jane te vazhduara
Attribute 4	Mosha	numerical	Mosha e personit qe do te sigurohet	vlerat e kufizuara nga 19 deri ne 65
Attribute 5	Nrpers	qualitative	numri i personave ne ngarkim	{1,2,3,4}
Attribute 6	Histsig	qualitative	nese ka patur sigurime te meparshme	po/jo
Attribute 7	Arsimi	qualitative	arsimi i personit qe do te sigurohet	I larte/tjeter
Attribute 8	Martuar	qualitative	statusi i gjendjes civile	po/jo
Attribute 9	Rajoni	qualitative	rajoni ku jeton personi qe do te sigurohet	brenda qytetit; qytet; fshat; periferi

Tabela 14 Algoritmi i k-mesatareve lista e atributeve qe jane perzgjedhur

Për çdo vlerë të parametrit hyrës kemi analizuar sesi ndryshojnë rezultatet sa herë që ndryshojmë metodën e testimit. Në eksperimentet me anë të algoritmeve të grupimit performanca vlerësohet gjatë testimeve të ndryshme nga kriteret e vleresimit si: koefiçienti i korrelacionit, mesatarja e gabimit absolut, rrënja katrore e mesatares se gabimit, gabimi relative absolut, rrënja katrore e gabimit relativ. Efektivitetin e një algoritmi grupimi e kemi vlerësuar nëpërmjet treguesit shuma e gabimeve në katror (SEC), i cili është i thjeshtë dhe i përdorur gjerësisht. Rezultatet e analizës së grupimit janë: numri i grupimeve te gjeneruara, kohën e marra për të ndërtuar modelet dhe numri i të dhënave te pagrupuara. Kemi krahasuar rezultatet e grupimit të arritura nga algoritmi i k-mesatareve duke përdorur metoden standarte te inicializimit me rezultatet që rrjedhin nga algoritmi i propozuar, ku numri total i ekzekutimeve është 36.

Bashkesia e te dhenave	Metoda	K	Distanca	SEC	Numri i perseritjeve
N=16’831	E rastit	5	Euklidiane	67,575	5
	E propozuar			55,708	4
	E rastit		Manhatan	70,646	4
	E propozuar			58,671	3

Tabela 15 Algoritmi i k-mesatareve rezultatet per parametrin hyres k =5

Tabela 15 tregon krahasimin e rezultateve të performancës duke përdorur metodën e rastit dhe metodën e propozuar për centroidet fillestare të grupimit, pasi është aplikuar në bashkësitë e të dhënave. Këto rezultate mesatare të grupimit janë nxjerrë për parametrin hyrës K=5, i cili përfaqëson numrin e grupimeve dhe jepet nga përdoruesi.
Ndërsa në tabelën 16 është treguar krahasimi i rezultateve të performancës për K=50, ku treguesi shuma e gabimeve në katror ka vlerën 28’177 për metodën e propozuar, vlera minimale e arritur që rezulton nga eksperimentet e kryera.

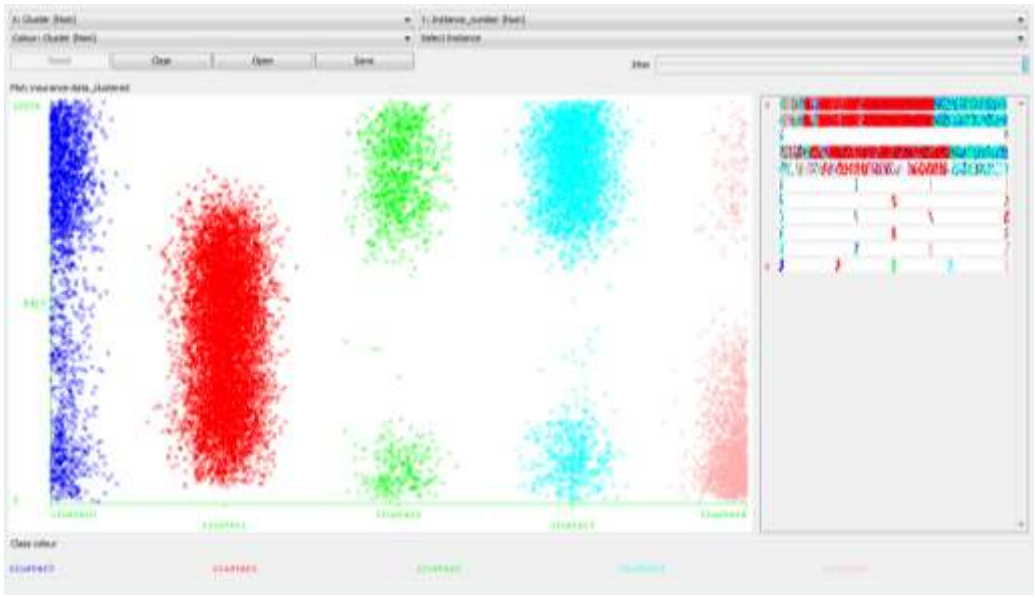
Bashkesia e te dhenave	Metoda	K	Distanca	SEC	Numri i perseritjeve
N=16’831	E rastit	50	Euklidiane	51,036	10
	E propozuar			**28,177**	**8**
	E rastit		Manhatan	70,646	4
	E propozuar			58,671	3

Tabela 16 Algoritmi i k-mesatareve rezultatet per parametrin hyres k =50

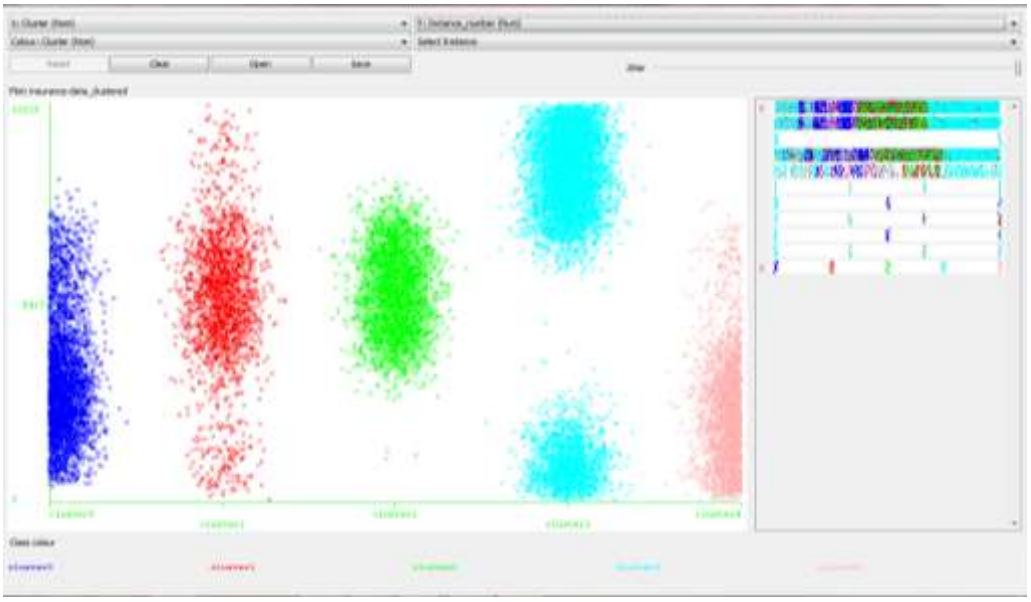
Vërejmë se përzgjedhja e centroideve sipas metodës së inicializimit të propozuar arrin vlera më të vogla për çdo numër grupimesh të dhënë sesa rezultatet e metodës së inicializimit të rastit. Kjo dëshmon se metoda e propozuar është më e qëndrueshme sesa metoda e rastit dhe paraqet rezultate më të mira.
Figurat 32, 33, 34 dhe 35 tregojnë rezultatet e ekzekutimit të algoritmit të k-mesatareve ne bashkësinë e të dhënave për parametrin hyres te barabarte me 5.

- Figura 32: Rezultatet e algoritmit të k-mesatareve duke përdorur inicializimin e rastit (distanca Euklidiane)
- Figura 33: Rezultatet e algoritmit të k-mesatareve duke përdorur metodën e propozuar të inicializimit (distanca Euklidiane)
- Figura 34: Rezultatet e algoritmit të k-mesatareve duke përdorur inicializimin e rastit (distanca Manhatan)
- Figura 35: Rezultatet e algoritmit të k-mesatareve duke përdorur metodën e propozuar të inicializimit (distanca Manhatan)

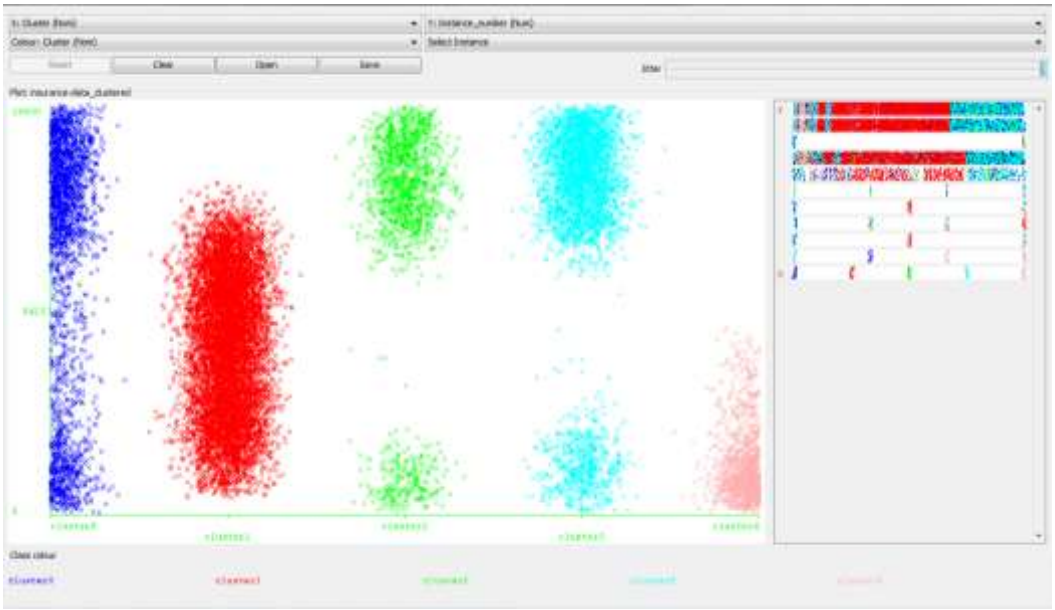
Çdo grafik paraqet grupime të identifikuara duke i dalluar me nga një karakter të ndryshëm pikat dhe ngjyrat, ku shënojmë rezultate gjerësisht divergjente për çdo metode.



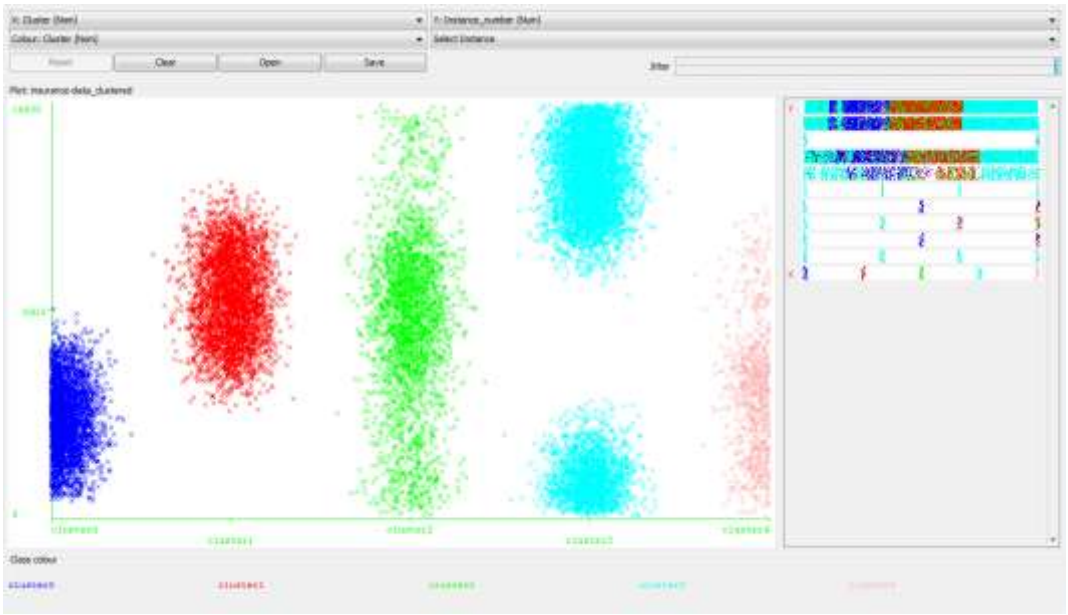
Grafiku 22 Rezultatet e grupimit sipas algoritmit K-mesatare për k=5 me metodën e inicializimit të rastit dhe distance Euklidiane



Grafiku 23 Rezultatet e grupimit sipas algoritmit K-mesatare per k=5 me metoden e inicializimit te propozuar dhe distanca Euklidiane



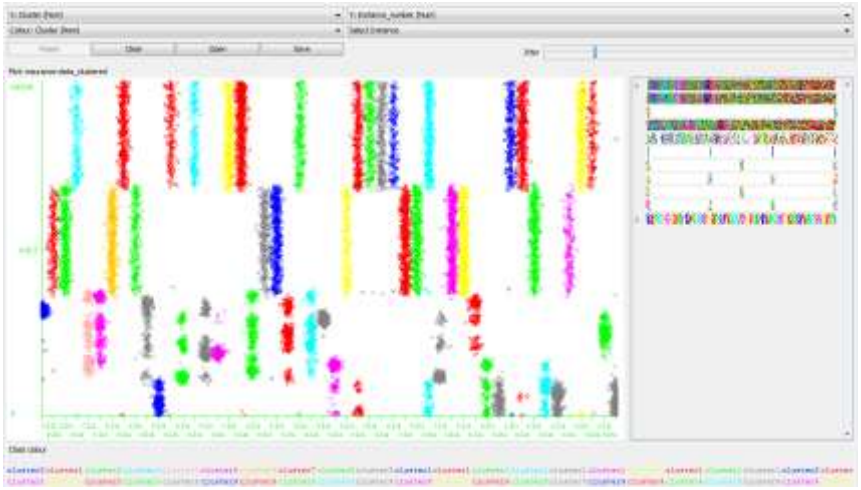
Grafiku 24 Rezultatet e grupimit sipas algoritmit K-mesatare për k=5 me metodën e inicializimit të rastit dhe distanca Manhatan



Grafiku 25 Rezultatet e grupimit sipas algoritmit K-mesatare për k=5 me metodën e inicializimit të propozuar dhe distanca Manhatan

Në figurën 36 tregojmë rezultatet e ekzekutimit të algoritmit të k-mesatareve në bashkësinë e të dhënave për 50 grupime me metodën e propozuar. Edhe në rastin për k=50 u vëzhgua se metoda për inicializim e rastit jep rezultatet joefektive pasi

ajo ngatërron dy grupime së bashku, e ndan ose e copëton një prej grupimeve të vërtetë në dy grupime të ndryshme. Ndërsa metoda e propozuar për inicializim është më efektive dhe më e saktë në identifikimin e secilit grupim shumë afër grupimeve të vërtetë.



Grafiku 26 Rezultatet e grupimit sipas algoritmit K-mesatare për k=50 me metodën e inicializimit të propozuar

8.5 ALGORITMI PERPARESOR NË GJETJEN E PRODUKTEVE TË LIDHURA

Për të kryer eksperimentin tonë kemi konsideruar 300 veprime që përbëhen nga produktet e shitura në një shoqëri sigurimi jete tek klientët e saj. Çdo veprim ka të specifikuar listën e produkteve të marra nga klienti. Në këtë eksperiment rezultatet janë pasqyruar në WEKA duke përdorur algoritmin përparësor. Skedari ARFF i paraqitur më poshtë përmban informacionin për çdo veprim në mënyrë të detajuar.

```
@attribute SJKredi {TRUE,FALSE}
@attribute SJKursim {TRUE,FALSE}
@attribute SJMartese {TRUE,FALSE}
@attribute SJVdekje {TRUE,FALSE}
@attribute SJFemije {FALSE,TRUE}
@attribute SJFamilje {TRUE}

@data
TRUE,TRUE,TRUE,TRUE,FALSE,TRUE
TRUE,TRUE,TRUE,TRUE,TRUE,TRUE
FALSE,TRUE,TRUE,TRUE,TRUE,TRUE
FALSE,TRUE,FALSE,FALSE,TRUE,TRUE
TRUE,TRUE,FALSE,TRUE,TRUE,TRUE
TRUE,FALSE,TRUE,FALSE,FALSE,TRUE
FALSE,TRUE,FALSE,TRUE,TRUE,TRUE
TRUE,FALSE,TRUE,TRUE,TRUE,TRUE
FALSE,TRUE,TRUE,TRUE,TRUE,TRUE
```

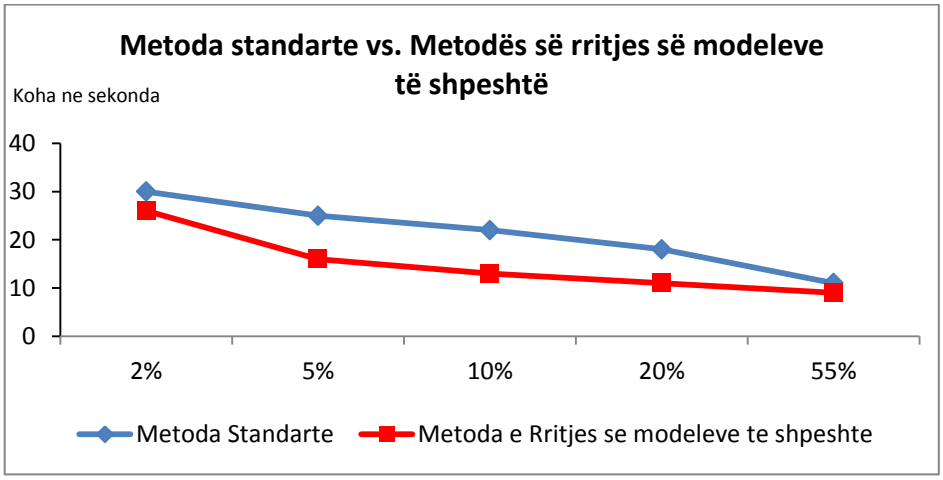
TRUE,FALSE,TRUE,FALSE,TRUE,TRUE
FALSE,FALSE,TRUE,FALSE,TRUE,TRUE
TRUE,FALSE,FALSE,TRUE,TRUE,TRUE
FALSE,TRUE,TRUE,FALSE,TRUE,TRUE
TRUE,TRUE,TRUE,FALSE,FALSE,TRUE
TRUE,TRUE,FALSE,FALSE,TRUE,TRUE
TRUE,TRUE,TRUE,TRUE,FALSE,TRUE
TRUE,TRUE,TRUE,TRUE,TRUE,TRUE
FALSE,TRUE,TRUE,TRUE,TRUE,TRUE
FALSE,TRUE,FALSE,FALSE,TRUE,TRUE
TRUE,TRUE,FALSE,TRUE,TRUE,TRUE
TRUE,FALSE,TRUE,FALSE,FALSE,TRUE
……

Figura 21 Algoritmi Perparesor skedari ARFF

Eksperimentet janë kryer duke përdorur vlerat për minimum mbështetje për të dy metodat: {2%, 5%, 10%, 20%, 55%}. Në tabelën 17 janë paraqitur kohët e proçesimit për çdo vlerë të mbështetjes minimum dhe në grafikun 10 është paraqitur krahasimi i rezultateve të nxjerra nga dy metodat e përdorura.

Mbështetja minimum	**2%**	**5%**	**10%**	**20%**	**55%**
Metoda Standarte (koha në sekonda)	30	25	22	18	11
Metoda e rritjes së modeleve të shpeshtë (koha ne sekonda)	26	16	13	11	9

Tabela 17 Algoritmi Përparësor rezultatet e nxjerra për çdo vlerë mbeshtetje minimum të metodës standarte dhe metodës së rritjes së modeleve të shpeshtë



Grafiku 27 Algoritmi Përparësor krahasimi midis metodës standarte dhe metodës së rritjes së modeleve të shpeshtë për çdo vlerë mbështetje minimum

Në figurën 39 janë paraqitur rezultatet e nxjerra mbi të dhënat hyrëse duke përdorur metodën standarte të Algoritmit Përparësor në Weka.

```
=== Run information ===
Scheme:       weka.associations.Perparesor -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Instances:    300
Attributes:   6
              SJKredi
              SJKursim
              SJMartese
              SJVdekje
              SJFemije
              SJFamilje
=== Associator model (full training set) ===
Apriori
=======
Minimum support: 0.55 (16 instances)
Minimum metric <confidence>: 0.5
Number of cycles performed: 90
Best rules found:
 1. SJFemije=TRUE 240 ==> SJFamilje=TRUE 240 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 2. SJKursim=TRUE 200 ==> SJFamilje=TRUE 200 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 3. SJMartese=TRUE 200 ==> SJFamilje=TRUE 200 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 4. SJKredi=TRUE 180 ==> SJFamilje=TRUE 180 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 5. SJVdekje=TRUE 160 ==> SJFamilje=TRUE 160 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 6. SJKursim=TRUE SJFemije=TRUE 160 => SJFamilje=TRUE 160 <conf:(1)> lift:(1) lev:(0) [0]
conv:(0)
 7. SJFamilje=TRUE 300 ==> SJFemije=TRUE 240 <conf:(0.8)> lift:(1) lev:(0) [0] conv:(0.86)
 8. SJKursim=TRUE 200 ==> SJFemije=TRUE 160  <conf:(0.8)> lift:(1) lev:(0) [0] conv:(0.8)
 9. SJKursim=TRUE SJFamilje=TRUE 200 ==> SJFemije=TRUE 160 <conf:(0.8)> lift:(1) lev:(0)
[0] conv:(0.8)
10. SJKursim=TRUE 200 ==> SJFemije=TRUE SJFamilje=TRUE 160 <conf:(0.8)> lift:(1) lev:(0)
[0] conv:(0.8)
```

Figura 22 Algoritmi Përparësor rezultatet sipas metodës standarte në WEKA

Në figurën 40 janë paraqitur rezultatet e nxjerra mbi të dhënat hyrëse duke përdorur metodën e rritjes së modeleve të shpeshtë të Algoritmit Përparësor në Weka.

```
=== Run information ===
Scheme:       weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 0.1
Instances:    300
Attributes:   6
              SJKredi
              SJKursim
              SJMartese
```

SJVdekje
SJFemije
SJFamilje
=== Associator model (full training set) ===
FPGrowth found 18 rules (displaying top 10)
1. [SJFemije=TRUE]: 240 ==> [SJFamilje=TRUE]: 240 <conf:(1)> lift:(1) lev:(0) conv:(0)
2. [SJVdekje=FALSE]:140 ==> [SJFamilje=TRUE]: 140 <conf:(1)> lift:(1) lev:(0) conv:(0)
3. [SJKredi=FALSE]: 120 ==> [SJFamilje=TRUE]: 120 <conf:(1)> lift:(1) lev:(0) conv:(0)
4. [SJMartese=FALSE]:100 ==> [SJFamilje=TRUE]: 100 <conf:(1)> lift:(1) lev:(0) conv:(0)
5. [SJKursim=FALSE]: 100 ==> [SJFamilje=TRUE]: 100 <conf:(1)> lift:(1) lev:(0) conv:(0)
6. [SJKredi=FALSE]: 120 ==> [SJFemije=TRUE]: 120<conf:(1)> lift:(1.25) lev:(0.08) conv:(2.4)
7. [SJMartese=FALSE]: 100 ==> [SJFemije=TRUE]: 100 <conf:(1)> lift:(1.25) lev:(0.07) conv:(2)
8. [SJFemije=TRUE, SJVdekje=FALSE]: 100 ==> [SJFamilje=TRUE]: 100 <conf:(1)> lift:(1) lev:(0) conv:(0)
9. [SJKredi=FALSE]: 120 ==> [SJFamilje=TRUE, SJFemije=TRUE]: 120 <conf:(1)> lift:(1.25) lev:(0.08) conv:(2.4)
10. [SJFamilje=TRUE, SJKredi=FALSE]: 120 => [SJFemije=TRUE]: 120 <conf:(1)> lift:(1.25) lev:(0.08) conv:(2.4)

Figura 23 Algoritmi Përparësor metoda e rritjes së modeleve të shpeshtë në WEKA

Duke përdorur metodën standarde dhe metodën e rritjes së modeleve të shpeshtë në algoritmin përparësor për të gjeneruar rregulla shoqërimi shohim se rezultatet më të mira i marrim në rastin kur përdorim metodën e rritjes së modeleve të shpeshtë.

8.6 Implementimi i AGJ në MATLAB për të Optimizuar rrjetin e shitjes në sigurime

Për të plotësuar optimizimin e manaxhimit të rrjetit të njësive të biznesit kemi ndërtuar një eksperiment në MATLAB. Në ketë eksperiment do të tregojmë se si algoritmi gjenetik i aplikuar në MATLAB ndihmon në gjetjen e distancës minimale midis koordinatave të objekteve të dhënë, për të optimizuar rrjetin e shitjes në një kompani sigurimi. Kjo kompani për të përmbushur kërkesat e klientëve kërkon të zgjerojë rrjetin e shitjes se saj edhe në qytete të tjera. Kështu përballë këtij problemi kompania duhet të gjejë fillimisht pikën në të cilën do të ndërtojë degën e saj të re. Kjo pikë duhet të jetë afër një degë tjetër ekzistuese të kompanise në mënyrë që për cdo lloj veprimi, komunikimi midis dy degëve të jetë me kosto të ulët. E më pas të përcaktojë të gjitha veprimet që dega e re të funksionojë në rregull.

Fillimisht ne kemi përdorur të gjitha koordinatat e degëve ekzistuese të kompanisë së sigurimit për qytetin e Tiranës. Supozojmë që kjo kompani kërkon të hapë një

degë të re në Durrës. Fillimisht kemi përcaktuar parametrat hyrës për AGj, që do të thotë përcaktimi i funksionit të fitnesit, i cili në rastin tonë përshkallezohet në gjetjen e minimumit të distancës midis dy degëve. Funksioni i fitnesit do permbajë formulën e llogaritjes së distancës minimum. Për të gjetur këtë funksion ne jemi bazuar në përfundime të gatshme që na i ofron matematika. Tasku optimizues që është përdorur në këtë eksperiment është analiza e grupimeve. Qëllimi kryesor i tij është grupimi i objekteve në grupe në mënyrë që dy objekte brënda të njëjtit grup janë më shumë të ngjashëm se objektet e një grupi-i tjetër.

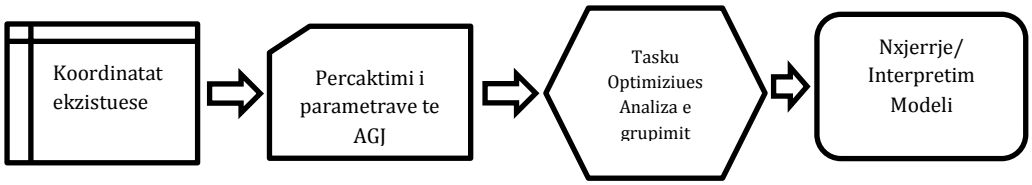


Figura 24 AGj Skema e ndërtimit të optomizimit te rrjetit e shitjes

Kompania e marrë në studim nga ne ka M-degë, të cilat duke u bazuar në llogjikën e mësipërme të analizës së grupimit do të ndahen në N-grupe. Cdo degë është e karakterizuar nga k-variabla, ku k përfaqëson dimensionet e vektorëve që paraqesin koordinatat e degëve. Koordinatat e një dege paraqiten nga tre dimensione x, y, z ku x- është gjatësia gjeografike (longitude), y- është gjerësia gjeografike (latitude) dhe z - lartësia mbi nivelin e detit (elevation).
Ideja e këtij eksperimenti konsiston në ndarjen e degëve ekzistuese të kompanisë në grupime dhe më pas minimizimin e ndryshueshmërisë së objekteve brënda grupit. Më pas vazhdohet me percaktimet matematikore që do të na cojnë drejt percaktimit të funksionit të fitnesit.
Përcaktojmë i dhe j si më poshtë:

$$i = 1, 2.....M \qquad \text{dhe} \qquad j = 1, 2.... N$$

dhe për keto variabla do përcaktojmë peshën si më poshtë:

$w_{ij}=1$ *nëse objekti i i-të është pjesë e grupit të j-të*
$w_{ij}=0$ *nëse objekti i i-të nuk është pjesë e grupit të j-të*

Matrica e peshave $\boldsymbol{W} = [w_{ij}]$ merr vlerat nga 0 ne 1 dhe shuma e të gjitha peshave është 1. Pasi realizohen disa veprime aritmetike, duke përfshirë këtu dhe përdorimin e formulës së distancës Euklidiane për gjetjen e distancen minimale midis një objekti dhe një centroidi. Funksionin e fitnesit e implementojmë në

scripte Matlab-i në mënyrë që të përdoret nga pajisja e algoritmit gjenetik e inkorporuar në Matlab, për të gjetur distancen minimum midis dy pikave. Më poshtë paraqitet scripti kryesor që bën implementimin e funksionit të mësiperm të fitnesit në Matlab:

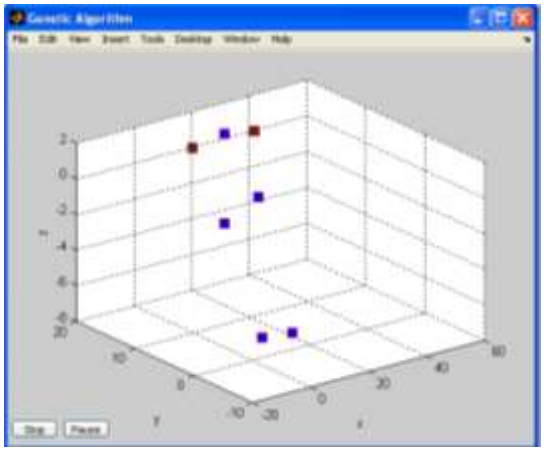
```
num=input(Numri i grupeve):');
num=3*num;
PopSize=input('Përmasa e Popullatës:');
FitnessFcn = @Distances;
numberOfVariables = num;
LOCATION=(xlsread('Distanca',Vendodhja))
my_plot = @(Options,state,flag)
Draw3(Options,state,flag,LOCATION,num);
Options =
gaoptimset('PlotFcns',my_plot,'PopInitRange',[0;1],'PërmasaPopullatës',Po
pSize);
[x,fval] = ga(FitnessFcn,numberOfVariables,Options);
assign=zeros(1,size(LOCATION,1));
for i=1:size(LOCATION,1)
distances=zeros(num/3,1);
for j=1:(size(x,2)/3)
distances(j)=sqrt((LOCATION(i,1)-x(j))^2+(LOCATION(i,2)-
x(size(x,2)/3+j))^2+(LOCATION(i,3)-x(2*size(x,2)/3+j))^2);
end
[min_distance,assign(i)]=min(distances)
```

Figura 25 AGj implementimi i funksionit të fitnesit në Matlab

Për të paraqitur rezultatin e eksperimentit tonë, kemi përdorur dhe një script tjetër në matlab që realizon vizatimin e një ndërfaqeje të përshtatshme për të paraqitur rezultatin e algoritmit gjenetik për funksionin e fitnesit të mësipërm.

Për të hapur paisjen e algoritmit gjenetik në matlab ekzekutohet në command line e MATLAB komanda >> gatool dhe brënda panelit të GAtool do të insertojmë funksionin e fitnesit @DistanceMinimum.

Algoritmi Gjenetik ekzekuton fillimisht funksionin e fitnesit mbi të dhënat hyrëse, të cilat përmbajnë informacion për degët e kompanisë në Tiranë. Këto të dhëna janë ruajtur në një skedar exceli. Bëhet importi i të dhënave nga exceli dhe rezultati që do marrim në përfundim paraqitet si në figurën më poshtë:



Grafiku 28 Algoritmi Gjenetik rezultati ne Matlab per N=5 dhe M=50

Në eksperimentin tonë rezultatet më të mira janë marrë për numër grupimesh të barabartë me 5 (N=5), ku përmasa e popullsisë është e barabartë me 50 (M=50). Nga këto të dhëna kemi përfituar distancën minimum të barabartë me 132.6080. Koha e përcaktimit të këtij rezultati është 110 sekonda, ndërsa saktësia qëndron në kufijtë 90%.

8.7 EFEKTIVITETI I AGJ NE KLASIFIKIMIN E TEKSTEVE NE SIGURIME

Ne kemi vlerësuar eksperimentalisht AGj-në për problemin e klasifikimit të të dhënave tekst duke përdorur të dhënat për 1'500 dokumenta (dosje polica sigurimi jete) të një kompanie sigurimi jete.

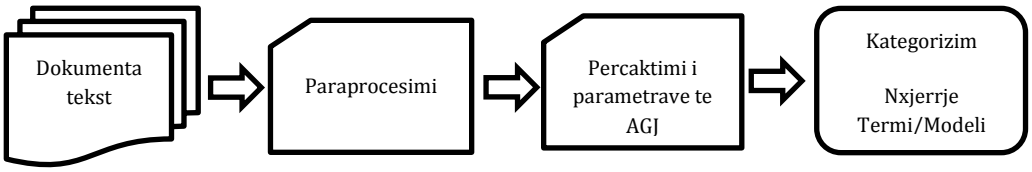


Figura 26 AGj Skema e ndërtimit të funksionimit për klasifikimin e teksteve

Paraprakisht, të dhënat janë nënshtruar hapave të para - përpunimit të mëposhtëm:

- Së pari, janë hequr të gjitha shenjat e pikësimit dhe numrat;
- Së dyti, janë nxjerrë të gjitha n-germat, të përcaktuara si sekuenca maksimale e tre fjalëve rresht që ndodhin brenda një teksti;
- Së treti, kemi ndarë rastësisht bashkësinë e të dhënave (70%) 1'050 dokumenta në një bashkësi të trajnimit, në të cilën do të ekzekutojmë AGj, dhe (30%) 450 dokumenta në një bashkësi test në të cilin do të vlerësojmë saktësinë e klasifikuesit më të mirë të gjeneruar nga AGj.

- Së katërti, për secilën kategori c ϵ C, janë shënuar të gjitha fjalët kyçe që ndodhen në bashkësisë së trajnimit që plotësojnë kriteret e përzgjedhjes së atributeve *f.*

Objektivi ynë është që të hulumtojmë efektivitetin e AGj-së së propozuar. Nga 25 kategori kemi konsideruar 10 prej tyre me numrin e termave më të përshtashëm ose që janë klasifikuar në mënyrë korrekte nga klasifikuesi. Janë kryer një numër i caktuar eksperimentesh për futjen e klasifikuesve më të mirë, ku për çdo fjalor hyrës *V(f, k)* kemi përcaktuar vlerat e *k* dhe *f* si më poshtë:

- $k \epsilon \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$
- $f \epsilon \{ Gain(A), \chi^2 \}$

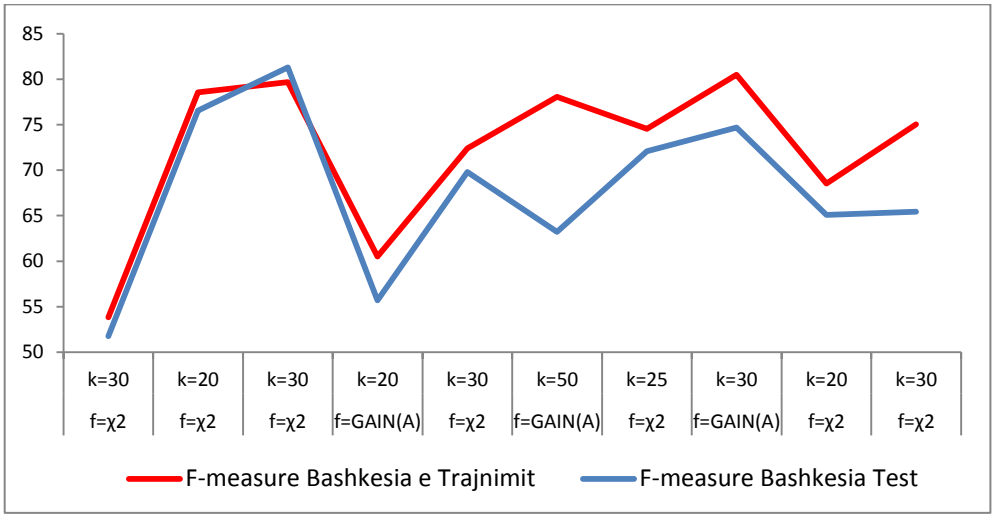
Për çdo vlerë të fjalorit hyrës ekzekutojmë algoritmin AGj tri herë, me vlerat e parametrave të tij të përcaktuara: Madhësia e popullatës 100 individë; Numri i brezave 50; Probabiliteti i mbikalimit 1.00 dhe probabiliteti i mutacionit 0,001;

Tabela e mëposhtme paraqet performancën e klasifikuesve për 10 kategoritë me numrin e termave më të përshtashëm dhe për secilin prej tyre kemi dhënë: vlerat e *f* dhe *k* të fjalorit, vlerat e njësisë F-measure për bashkësinë e trajnimit dhe bashkësinë test, numrin e termave positive dhe negative që gjënden në klasifikues.

Kategoria ID	*F*	*k*	Traning F-measure	Test F-measure	Klasifikuar pozitiv	Klasifikuar negativ
C17	χ2	30	53.83	51.77	7	4
C02	χ2	20	78.57	76.56	9	2
C11	χ2	30	89.69	84.28	19	5
C06	GAIN(A)	20	60.53	55.7	7	7
C09	χ2	30	72.41	69.79	28	1
C20	GAIN(A)	50	78.05	63.21	28	1
C18	χ2	25	74.54	72.08	11	1
C14	GAIN(A)	30	80.49	74.7	16	2
C08	χ2	20	68.53	65.09	12	6
C01	χ2	30	75.04	65.42	28	1

Tabela 18 AGj Rezultatet për 10 kategori që gjenerojnë klasifikuesit më të mirë

Klasifikuesi më i mirë është C11 me F-measure të barabartë me 89.69 në bashkësinë e trajnimit dhe 84,28 në bashkësinë test.



Grafiku 29 AGj Vlerat e F-measure për bashkësinë e trajnimit dhe test për 10 kategoritë më të mira

PËRMBLEDHJE

Data Mining është një fushë në zhvillim të vazhdueshëm, pjesë e një proçesi që quhet KDD-zbulimi i njohurisë në bazat e të dhënave, e cila përmbledh një sërë fushash studimi të cilat japin kontributin e tyre si statistika, machine learning, inteligjenca artificiale, sistemet e bazave të të dhënave dhe data warehouse. Gërmimi i të dhënave mund të kryhet për arsye të ndryshme, në varësi të të cilave zgjidhen dhe metodat e kërkimit.

Për teknikën e klasifikimin paraqitëm konceptin e saj, përdorimin e pemëve të vendimit si struktura të rëndësishme për klasifikimin e të dhënave në sigurimin e jetës. Për algoritmin pemë vendimi treguam proçesin e përftimit të rregullave. Aspektet pozitive të përdorimit të pemëve të vendimit në sigurimin e jetës: (i) janë vetëshpjegues; (ii) mund të trajtojnë të dhëna numerike dhe nominale. (iii) mund të trajtojnë të dhëna që mund të përmbajnë gabime apo vlera që mungojnë. Përmirësimin e algoritmit pemë vendimi CART e bëmë nëpërmjet vendosjes së një ndarësi zëvendësues, ku çdo ndarës bëhet një synues i ri, i cili parashikohet me një pemë ndarëse unike binare.

Identifikuam problemet që ndikojnë në performancën e algoritmit KNN dhe kombinuam dy problematikat zgjedhjen e madhësisë së largësisë dhe përafrimin drejt kombinimit të emërtimeve të grupeve për të përmirësuar performancën dhe rritur shkallën e saktësisë së klasifikimit.

Rrjetat nervore janë struktura mjaft të rëndësishme në teknikën e klasifikimit me një nivel gabimi të krahasueshëm me atë të pemëve të vendimit dhe ndjeshmëri të vogël ndaj zhurmave. Këtë algoritëm e aplikuam në një bazë të dhënash në sigurimin e jetës, si një metodë alternative klasifikimi, me qëllim identifikimin e karakteristikave të klientëve që janë të siguruar, nëse janë me risk të lartë ose të ulët dhe për t'i përdorur më pas këto karakteristika tek klientët e rinj.

Për teknikën e grupimin paraqitëm analizën e grupimit, metodat e ndarjes nëpërmjet algoritmeve k-mesatareve dhe k-mediana, metodat hiearkike dhe metodat e bazuara në denduri. Prezantuam një mënyrë të re për të përzgjedhur centroidet fillestare në algoritmin e k-mesatareve. Kjo metodë inicializimi është aq sa e shpejtë dhe e thjeshtë sa dhe vetë algoritmi i k-mesatareve. Arsyeja kryesore e kësaj arritje është që ta bënim algoritmin e k-mesatareve më pak të ndjeshëm ndaj

procesit të inicializimit dhe që të merrnim rezultate të qëndrueshme sa herë që ai ekzekutohet.

Për teknikën analiza e shoqerimit trajtuam analizën e shportës së tregut, analizën e shoqërimit, dhe rregullat e saj. Hulumtuam algoritmat ekzistues të analizës së shoqërimit duke propozuar një përmirësim për algoritmit përparësor. Për algoritmin përparësor propozuam përmirësimin e tij nëpërmjet metodës së rritjes të modeleve të shpeshtë, e cila thjeshton termin e përftuar duke përshtatur një ndarje dhe duke vendosur termat e shpeshtë brënda një strukture.

Hulumtuam një algoritëm të ri AGj për klasifikimin e dokumentave të bazuar në rregulla të formës "në qoftë se teksti d përfshin një prej termave t_1 ose...ose t_n, por jo termat t_{n+1} dhe ... t_{n+m}, atëherë klasifikoje d sipas kategorisë c". Problemi i të mësuarit është formuluar si një detyrë optimizimi, ku bashkësia e trajnimit përfaqësohet si një problem kombinatorik optimizimi për qëllim gjetjen e një kombinimi më të mirë të termave të marra nga një fjalor të caktuar.

VERTETIMI I HIPOTEZAVE DHE KONKLUZIONET

Duke u bazuar tek rezultatet dhe analizat e bëra në eksperimentet më sipër u munduam të vërtetojmë hipotezat e ngritura në mbështetje të këtij punimi shkencor, ku rezultoi se:

Hipoteza 1: Përmirësimi i teknikave të klasifikimit të data mining çon në vlerësimin dhe përcaktimin më të mirë të ndryshimeve demografike të klientëve në kompanitë e sigurimit të jetës.

- Përmiresimi i teknikave të klasifikimit CART, KNN rezultoi efikas në vlerësimin dhe përcaktimin e ndryshimeve demografike të klientëve në industrinë e sigurimit të jetës.
- Algoritmi CART i përmirësuar rezultoi të përdoret si një parashikues për mundësinë e paracaktimit të vlerësimit për klientët e rinj.
- Përmirësimi i algoritmit KNN çoi në përcaktimin e faktorëve që ndikojnë në vlerësimin e klientëve me risk në sigurimin e jetës si mosha, gjinia dhe pesha.

Hipoteza 2: Vendosja e një ndarësi zëvëndësues në modelet pemë vendimi të algoritmit CART përmirëson performancën në klasifikimin e të dhënave që përdoren nga kompanitë e sigurimit të jetës.

- Përmirësimi i performancës për algoritmin CART është bërë nëpërmjet vendosjes të një ndarësi zëvendësues për klasën e klientëve, të cilët janë mohuar për të marrë një sigurim jete. Rezultatet eksperimentale tregojnë se modifikimi i propozuar për algoritmin CART rezultoi më i mirë në terma të saktësisë 85.9% se modeli i tij standart me 64.6%.

Hipoteza 3: Zgjedhja e madhësisë së largësisë dhe përafrimi drejt kombinimit të emërtimeve të grupeve në modelet e ndërtuara me anë të algoritmit të fqinjësisë më të afërt përmirëson performancën, e cila ndikon në vlerësimin e klientëve në kompanitë e sigurimit të jetës.

- Algoritmi i fqinjësisë më të afërt është një klasifikues i bazuar në të mësuarit me analogji dhe është efikas për grupe të mëdha trajnimi. Rezultatet eksperimentale tregojnë se modeli i ndërtuar sipas largësisë Euklidiane të ponderuar është më i mirë se modelet e tjera të ndërtuara sipas largësive

Euklidiane e thjeshtë, Manhatan e thjeshtë dhe të ponderuar. Saktësia e modelit të ndërtuar sipas largësisë Euklidiane të ponderuar rezultoi 87%.

Hipoteza 4: Përcaktimi i përzgjedhjes së centroidit fillestar të grupimit në modelet e ndërtuara me anë të algoritmit të k-mesatareve siguron një rritje të performancës të tij se metoda e rastit, kur përdoret për grupime të dhënash në kompanitë e sigurimit të jetës.

- Algoritmi i k-mesatareve përbën një metodë të thjeshtë grupimi të të dhënave, të përftuara sipas një numri k grupimesh të dhënë. Përmirësimi i tij konsiston në gjetjen e një mënyre të re përzgjedhjeje të centroideve fillestare. Rezultatet eksperimentale tregojnë se metoda e propozuar e përzgjedhjes së centroideve fillestare rezulton 45% më e mirë sipas treguesit shuma e gabimeve ne katror se metoda standarte. Gjithashtu rezultatet eksperimentale treguan se modifikimi jep përformancë me eficente kur ka të bëjë me bashkësi të dhënash me permasa të ndryshme.

Hipoteza 5: Përmiresimi i algoritmin përparësor me anë të metodës së rritjes së modeleve të shpeshtë që thjeshton termin e përftuar duke përshtatur një ndarje, e cila jep strategjinë për vendosjen e të dhënave që përfaqësojnë terma të shpeshtë brënda një strukture. Duke përdorur teknikën analiza e shoqërimit dhe duke kryer analiza të njëpasnjëshme në grupe të caktuara klientësh, kompanitë e sigurimit të jetës mund të zgjedhin se cilat shërbime të ofrojnë dhe ndaj cilëve klientë.

- Metoda e rritjes së modeleve të shpeshtë e përdorur nga algoritmi përparësor gjeneron më shumë rregulla shoqërimi në një kohë shumë më të vogël se metoda standarte duke patur të njëjtën bazë të dhënash dhe mbështetje minimum të njëjtë.
- Rezultatet eksperimentale tregojnë se metoda e rritjes se modeleve të shpeshtë është 30% më e shpejtë në kohë proçesimi dhe gjeneron dyfishin e rregullave të shoqërimit se metoda standarte e algoritmit përparësor.
- Gjithashtu kompanitë e sigurimit të jetës nga rregullat e gjeneruara nga modelet e ndërtuara zbulojnë produktet, të cilat shiten gjithmonë bashkë dhe mund të ndërtojnë strategji për klientët që kanë të njëjta karakteristika për t'ju ofruar shërbimet e tyre.

Hipoteza 6: Duke përdorur algoritmat gjenetike kompanitë e sigurimit të jetës mund të përmirësojnë dhe optimizojnë rrjetin e shitjes dhe mund të klasifikojnë dokumentat.

- Algoritmat gjenetikë të implementuar nëpërmjet analizës së grupimit në Matlab mund të përdoren për gjetjen e distancës minimum midis dy koordinatave. Gjithashtu për kompaninë kjo rezulton zgjidhja optimale dhe mjafton të përdor njërën nga rezultatet e gjetura që të vendosë për pozicionimin e degës së saj të re.
- Përdorimi i algoritmave gjenetikë rezulton të jetë një metodë zgjidhjeje efikase per klasifikimin e teksteve. Metoda e propozuar është një algoritëm i të mësuarit me një hap, e cila nuk ka nevojë për asnjë lloj optimizimi të mëpasshëm për të përmirësuar bashkësinë e rregullave të gjetura.
  Koha mesatare e ekzekutimit për algoritmin AGj në klasifikimin e teksteve rezultoi afërsisht 10 sekonda për secilën kategori.

LITERATURA

[1] Dawei, J. The Application of Date Mining in Knowledge Management. International Conference on Management of e-Commerce and e-Government, IEEE Computer Society, 2011.

[2] Berson, A., Smith, S.J. &Thearling, K. Building Data Mining Applications for CRM. New York: McGraw-Hill, 1999.

[3] Gorunescu, F. Data Mining: Concepts, Models, and Techniques, Springer, 2011.

[4] S. Weng, R.K. Chiu, B.J. Wang, S.-H. Su. The study and verification of mathematical modeling for customer purchasing behavior, Journal of Computer Information Systems 47:2, f. 46–50, 2007.

[5] R.M. Rejesus, B.B. Little, A.C. Lovell. Using data mining to detect crop insurance fraud: Is there a role for social scientists? Journal of Financial Crime 12:1, f. 24–32, 2004.

[6] G.S. Linoff. Survival data mining for customer insight, Intelligent Enterprise 7:12, f.28–33, 2004.

[7] Han, J.; Kamber, M.; Pei.J.; Data Mining: Concepts and Techniques. 3rd.ed. Boston: Morgan Kaufmann Publishers, 2012.

[8] Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P. From Data Mining to Knowledge Discovery in Databases.AI Magazine, 17(3), f. 37-54, 1996.

[9] ACM SIGKDD:www.acm.org/sigkdd & KDD Nuggets:www.kdnuggets.com

[10] http://www.rithme.eu/?m=home&p=kdprocess&lang=en

[11] Brunela Karamani, Shkëlqim Kuka: Data Mining in Life Insurance Industry. Information Systems and Technologies and their importance in the Economic Development, 2011.

[12] Tan, P.-N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison-Wesley, Reading, 2005.

[13] Database Management System [DBMS] Tutorial; Simply Easy Learning by www.tutorialspoint.com.

[14] Ralph, Kimball, the DataWarehouse Toolkit: Practical Technique for Buiding Dimensional Data Warehouses. New York: John Wiley, 1996.

[15] Bhatnagar V. and Gupta S.K. Modeling the KDD Process. Encyclopedia of Data Warehousing and Mining, Second Edition. Information Science Reference, Hershey, New York, f.1337 – 1344, 2008.

[16] Usama, M.Fayyad, et al., Advances in Knowledge Discovery and Data Mining Cambridge, Mass.MIT Press, 1996.

[17] Smyth P. "Data Mining: Data analysis on a grand scale?" Stat Methods Med Res, 9, f. 309–327, 2000.

[18] T.Hastie, R.Tibshirani, and J. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd Edition), Springer, 2009.

[19] Duda, R.O., Hart, P.E., and Stork D.G Pattern classification, John Wiley and Sons, New York, NY, 2001.

[20] Han J. and Kamber M. Data Mining Concept and Techniques. London: Morgan Kaufmann Publishers, 2001.

[21] E. Alpaydin. Introduction to Machine Learning (2nd ed.). Cambridge, MA: MIT Press, 2011.

[22] J. R. Quinlan. Induction of decision trees, Machine Learning, f. 81–106, 1986.

[23] J. R. Quinlan.Programs for Machine Learning. Morgan Kaufmann, 1993.

[24] L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and Regression Trees. Wadsworth International Group, 1984.

[25] Berry M.J., Linoff G., "Data Mining Techniques for Marketing, Sales and Customer Support", John Wiley & Sons Inc., USA, 1997.

[26] S. Marsland. Machine Learning: An Algorithmic Perspective. Chapman & Hall /CRC, 2009.

[27] S. K. Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. Data Mining and Knowledge Discovery f. 345–389, 1998.

[28] M. Ankerst, C. Elsen, M. Ester, and H.-P. Kriegel. Visual classification: An interactive approach to decision tree construction. Int.Conf.Knowledge Discovery and Data Mining (KDD'99), f. 392–396, 1999.

[29] S. L. Crawford. Extensions to the CART algorithm. Int. J. Man-Machine Studies, f. 197–217, 1989.

[30] Bloch DA, Olshen RA, Walker MG Risk estimation for classification trees. J Comput Graph Stat, f. 263–288, 2001.

[31] Brunela Karamani, Esteriana Haskasa, Josef Bushati: Classification of a Life Insurance Database with SPSS".The 1st International Conference on Research and Educatıon – Challenges toward the Future, ICRAE 2013.

[32] Tan P-N, Steinbach M, Kumar V. Introduction to data mining. Pearson Addison-Wesley, 2006.

[33] B. V. Dasarathy. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, 1991.

[34] F.P.Preparata and M. I. Shamos. Computational Geometry: An Introduction. Springer- Verlag, 1985.

[35] X. D. Wu, V. Kumar et al, Top 10 algorithems in data mining, Knowledge Information System, 14, 2008.

[36] J.Wang, P.Neskovic, L.N.Cooper, Neighborhood size selection in the k-nearest-neighbor rule using statistical confidence, f. 417-423, 2006.

[37] Han E. Text categorization using Weight adjusted k-nearest neighbor classification. University of Minnesota, 1999.

[38] Brunela Karamani, Esteriana Haskasa: K-Nearest Neighbours Algorithm in Insurance. International Journal of Science, Innovation and New Technology;Nentor 2012.

[39] Kuramochi M, Karypis G. Gene Classification using Expression Profiles: A Feasibility Study. Int J Artif Intell Tools 14(4), f. 641–660, 2005.

[40] From Data Mining to Knowledge Discovery in Databases- Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, 2012.

[41] Effective Data Mining Using Neural Networks- Hongjun Lu, Rudy Setiono (IEEE Computer Society) ,1996.

[42] B.Widrow, D. E. Rumelhart, and M. A. Lehr. Neural networks: Applications in industry, business and science. Comm. ACM, 37, f. 93–105, 1994.

[43][43] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65, f. 386–498, 1958.

[44] M.W. Craven and J.W. Shavlik.Using neural networks in data mining. Future Generation Computer Systems, 13, f. 211–229, 1997.

[45] I. H. Witten and E. Frank. Data Mining: Practical Machine Learning Tools and Techniques (2nd ed.). Morgan Kaufmann, 2005.

[46] D. Freedman, R. Pisani, and R. Purves. Statistics (4th ed.). W. W. Norton & Co., 2007.

[47] Alicja J.Michał P. Springerlink.com, Nonparametric estimation of the ROC curve based on smoothed empirical distribution functions; Volume 23, f.703-712, 2013.

[48] M. Vuk, T. Curk. ROC curve, lift chart and calibration plot, f. 89–108, 2006.

[49] Tan, P.-N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison-Wesley, Reading , 2005.

[50] Ince, E.A., Ali, S.A.: Rule based segmentation and subject identification using fiducial features and subspace projection methods. Journal of Computers 2(4), f. 68–75, 2007.

[51] Han, J.;Kamber, M.; Pei.J.; Data Mining: Concepts and Techniques. 3rd.ed. Boston: Morgan Kaufmann Publishers, f.445-448, 2012.

[52] T. Lee and J. J. Mody. Behavioral Classification. In EICAR Conference, 2006.

[53] C.Li, G.Biswas “Unsupervised Learning with mixed Numeric and Nominal Data.IEEE Transactions on Knowledge and Data Engineering. Vol.14, No.4, 2002.

[54] Magdy, A.; Yousri, N.A.; El-Makky, N.M.; “Discovering Clusters with Arbitrary Shapes and Densities in Data Streams” IEEE Conference publications, 2011.

[55] Kumar, V. Knowledge discovery from database using an integration of clustering and classification. IJACSA - International Journal of Advanced Computer Science and Applications; 2(3), f. 29-33, 2011.

[56] A. Jain and R. Dubes, "Algorithms for Clustering Data," Prentice Hall, 1988.

[57] S. Kalyani and K.S. Swarup, "Particle swarm optimization based K-means clustering approach for security assessment in power systems," Expert Systems with Applications, vol. 30, f. 10839–10846, 2011.

[58] Lloyd SP; Least squares quantization in PCM. Unpublished Bell Lab. Tech. Note, portions presented at the Institute of Mathematical Statistics Meeting Atlantic City, 1957.

[59] Jain AK, Dubes RC; Algorithms for clustering data. Prentice-Hall, Englewood Cliffs, 1988.

[60] Gray RM, Neuhoff DL; Quantization. IEEE Trans Inform Theory 44(6), f. 2325–2384, 1988.

[61] Brunela Karamani: Decision Making to Predict Customer Preferences in Life Insurance; IJERT International Journal of Engineering Research and Technology Issues, Vol. 2, Issue 9, September 2013; ISSN: 2278-0181; IJERT Impact Factor 1.76.

[62] Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand, Dan Steinberg; Top 10 algorithms in data mining; Knowledge and Information Systems; Volume 14, Issue 1,f. 1-37, 2008.

[63] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," ACM-SIAM Symposium on Discrete Algorithms (SODA 2007) Astor Crowne Plaza, New Orleans, Louisiana, f. 1–11, 2007.

[64] R. Maitra, "Initializing partition-optimization algorithms," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 6, f. 144–157, 2009.

[65] M.C. Naldi, R.J.G.B. Campello, E.R. Hruschka, and A.C.P.L.F. Carvalho, "Efficiency issues of evolutionaryk-means," Applied Soft Computing, vol. 11, f. 1938–1952, 2011.

[66] H.S. Park, C.H. Jun, A simple and fast algorithm for K-medoids clustering, Expert Systems with Applications, 36(2), f.3336–3341, 2009.

[67] A. P. Reynolds, G. Richards, and V. J. Rayward-Smith The Application of K-medoids and PAM to the Clustering of Rules,Springer, 2004.

[68] Han, J.; Kamber, M.; Pei.J.; Data Mining: Concepts and Techniques. 3rd.ed. Boston: Morgan Kaufmann Publishers, f. 451- 457, 2012.

[69] Székely, G. J. and Rizzo, M. L. Hierarchical clustering via joint between within Distances: Extending Ward's Minimum Variance Method, Journal of Classification 22, f.151-183, 2005.

[70] Campello, R. J. G. B.; Moulavi, D.; Sander, J. Density-Based Clustering Based on Hierarchical Density Estimates. Proceedings of the 17th Pacific-Asia Conference on Knowledge Discovery in Databases, 2013.

[71] Kriegel, H.; Kröger, P.; Sander, J.; Zimek, A. Density-based Clustering; WIREs Data Mining and Knowledge Discovery 1, f. 231–240, 2011.

[72] Piatetsky-Shapiro, G. Discovery, analysis, and presentation of strong rules. In: Piatetsky-Shapiro, G., Frawley, W. (eds.) Knowledge Discovery in Databases, AAAI/MIT Press, Cambridge/MA ,1991.

[73] Agrawal, R., Imielinski, T., Swami, A.,Mining association rules between sets of items in large databases. In: Proc. 1993 ACM SIGMOD Conference, Washington, f. 207– 216, 1993.

[74] Han, J., Pei, J., Yin, Y., Mao, R.: Mining frequent patterns without candidate generation. Data Mining and Knowledge Discovery 8, f.53–87, 2004.

[75] Adamo, J.M.: Data Mining for association rules and sequential patterns: sequential and parallel algorithms. Springer, Heidelberg, 2001.

[76] Webb, Geoffrey I.; Discovering Significant Patterns, Machine Learning 68(1), Netherlands: Springer, f. 1-33, 2007.

[77] Tan, P.-N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison-Wesley, Reading , 2005.

[78] Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules in Large Databases. In: Proc. 20th International Conference on Very Large Data Bases, f. 487–499. Morgan Kaufmann, San Francisco, 1994.

[79] Zhang, C., Zhang, S.: Association Rule Mining. Models and Algorithms. LNCS (LNAI), vol. 2307, Springer, Heidelberg, 2002.

[80] Brunela Karamani, Shkëlqim Kuka, Akli Fundo: Një zbatim i Algoritmit Përparesor për Gjetjen e Rregullave të Shoqërimit në baza të mëdha të dhënash. Studime të avancuara në Inxhinierine Matematike, Fizike dhe Kimike; FIMIF, UPT, 2011.

[81] Jiawei Han, Micheline Kamber, Jian Pei Data Mining: Concepts and Techniques; f.255-256, 2011.

[82] D. Bhalodiya, K. M. Patel and C. Patel. An Efficient way to Find Frequent Pattern with Dynamic Programming Approach, Nirma University International Conference On Engineering, NUiCONE-2013, 2013.

[83] Z. H. Deng, Z. Wang, and J. Jiang. A New Algorithm for Fast Mining Frequent Itemsets Using N-Lists. Science China Information Sciences, 55(9); f.2008 - 2030, 2012.

[84] Han J, Pei J, Yin Y Mining frequent patterns without candidate generation. In: Proceedings of ACM SIGMOD international conference on management of data, 2012.

[85] HOLLAND, J.H., Adaptation in natural and artificial systems. University of Michigan Press, 1975.

[86] Mitchell, M., An Introduction to Genetic Algorithms, Massachusetts Institute of Technology, 1996.

[87] Dorsey, R. E. and W. J. Mayer: Genetic Algorithms for Estimation Problems with Multiple Optima, Non-Differentiability, and other Irregular Features, Journal of Business and Economic Statistics 13(1), 53-66, 1995.

[88] K.Youksel, B.Bozkurt, H.Ketabdar: A Software platform for Genetic Algorithms based Parameter Estimation on Digital Sound Synthesizer, ACM, 2011.

[89] Goldberg, D. E. "Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, Massachusetts", Addison-Wesley, 1989.

[90] K.Youksel, B.Bozkurt, H.Ketabdar: A Software platform for Genetic Algorithms based Parameter Estimation on Digital Sound Synthesizers, SAC'11, ACM, 2011.

[91] A.H. Wright, "Genetic algorithms for real parameter optimization, in Foundations of Genetic Algorithms, J.E. Rawlins (Ed.), Morgan Kaufmann, f. 205-218, 2011.

[92] Kim, D.S., and Park, G.S.: 'Modeling network instruction detection system using feature selection and parameters optimization', Ieice Transactions on Information and Systems, E91D, (4), f. 1050-1057, 2008.

[93] W. M. Spears and K. A. De Jong, "An Analysis of Multi-Point Crossover", In Foundations of Genetic Algorithms, J. E. Rawlins (Ed.), pp. 301-315, 1991.

[94] K.F.Man, K.S and Tang, S.Kwong, "Genetic Algorithms: Concepts and Designs" 2010.

[95] N. Suguna, and K. Thanushkodi, "A Novel Rough Set Reduct Algorithm Based on Bee Colony Optimization", International Journal of Granular Computing, Rough Sets and Intelligent Systems, 2010.

[96] Hristakeva, M., Shrestha, D.: Solving the 0/1 Knapsack Problem with Genetic Algorithms. In: Midwest Instruction and Computing Symposium Proceedings, 2004.

[97] Aggarwal, Ch.C. and Zhai. Ch. Mining text data, Springer, 2012.

[98] Shumeyko A.A., Sotnik S.L., Lysak M.V. - Using genetic algorithms in texts classification problems, In VI International Conference "Mathematical methods and Programming inIntelligence Systems", 2008.

[99] Jiawei Han, Micheline Kamber, Jian Pei: Data Mining: Concepts and Techniques- 3rd Edition Morgan Kaufmann, ISBN 978-0-12-381479-1, 2011.

[100] Lior Rokach, Oded Maimon, Data mining with Decision Trees: Theory and Applications, ISBN 978-9812771711, 2008.

[101] Linda Null, Julia Lobur, The essentials of computer organization and architecture, ISBN 0-7637-0444-X, 2010.

Botimet e mia

1. Brunela Karamani, Shkëlqim Kuka: Data Mining in Life Insurance Industry. Information Systems and Technologies and their importance in the Economic Development; 10 - 11 June 2011; Tirana, Albania.

2. Brunela Karamani, Shkëlqim Kuka, Akli Fundo: Një zbatim i Algoritmit Përparesor për Gjetjen e Rregullave të Shoqërimit në baza të mëdha të dhënash. Studime të avancuara në Inxhinierine Matematike, Fizike dhe Kimike; FIMIF, UPT, 28 Tetor 2011; Tirana, Albania.

3. Brunela Karamani, Esteriana Haskasa, Josef Bushati: Classification of a Life Insurance Database with SPSS".The 1st International Conference on Research and Educatıon – Challenges toward the Future, ICRAE 2013.

4. Brunela Karamani, Esteriana Haskasa: K-Nearest Neighbours Algorithm in Insurance. International Journal of Science, Innovation and New Technology; Nentor 2012; Tirana, Albania.

5. Brunela Karamani, Esteriana Haskasa, Shkelqim Kuka: Using Data Mining for Life Insurance Network Optimization, IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 3, No 2, May 2013; ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784.

6. Brunela Karamani: Decision Making to Predict Customer Preferences in Life Insurance; IJERT International Journal of Engineering Research and Technology Issues, Vol. 2, Issue 9, September 2013; ISSN: 2278-0181; IJERT Impact Factor 1.76.