SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1868

MODELI GRUPIRANJA I KLASIFIKACIJE KRIVULJA MIKCIOMETRIJE

Josip Renić

Zagreb, lipanj 2019.

Sadržaj

1. Uvod

Mikciometrija je vrlo jednostavna i neinvazivna pretraga mjerenja mlaza urina. Provodi se kod pacijenata sa smetnjama mokrenja (učestalo ili otežano mokrenje) te daje osnovni uvid u funkciju mokraćnog mjehura i mokraćne cijevi. Pacijent/ica mora imati puni mokraćni mjehur i potrebu za mokrenjem te mokri spontano u poseban aparat, tzv. mikciometar, koji mjeri protok urina u mililitrima po sekundi. Tijekom normalnog mokrenja, strujanje mokraće počinje polako, ubrzava, a zatim se konačno opet usporava. Mikciometar može zabilježiti bilo kakve razlike u odnosu na normu kako bi pomogao liječniku da postavi dijagnozu.

Razmatraju se biološki vremenski nizovi krivulja mikciometrije snimanih kod djece. Anonimizirani zapisi ovih vremenskih nizova dobiveni su od Klinike za dječje bolesti Zagreb. Disfunkcija donjeg mokraćnog sustava u djece češća je od one koju bi u početku očekivali. Čak 7 do 10 posto djece školske dobi ima urinarnu inkontinenciju i / ili povratne infekcije mokraćnog sustava. Taj postotak čini ju jednom od najčešćih kroničnih bolesti u pedijatrijskoj dobi. [1]

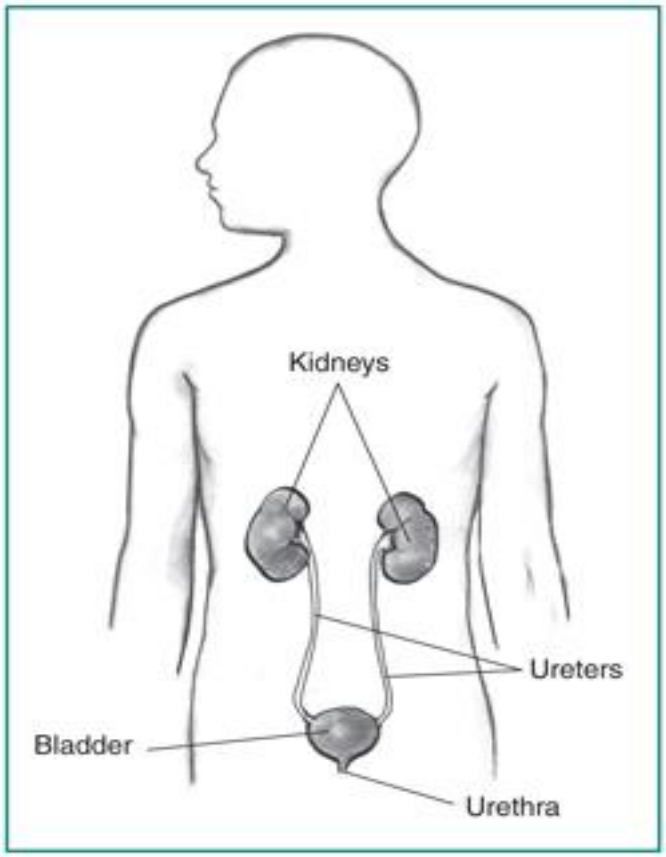
Ideja rada je pružiti platformu za klasifikaciju mikciometrijskih krivulja u optimalni broj grupa dobiven za dostupan skup podataka. U ovom radu promatramo opisivanje krivulja mikciometrije ekspertnim značajkama uz pomoć kojih će se pokušati odrediti optimalan broj grupa krivulja odabranim metodama grupiranja (postupak K-srednjih vrijednosti, postupak EM i postupak DBSCAN). Odabir optimalnog broja grupa za dane podatke procijenit ćemo temeljem mjera: koeficijent siluete, indeks Calinski-Harabasz i indeks Davies-Bouldin. Nakon određivanja broja grupa želja nam je omogućiti klasifikaciju novo viđenih primjera a to ćemo ostvariti treniranjem klasifikacijskih postupaka nad dobivenim grupama iz odabranog postupka grupiranja. Također želimo koristiti klasifikacijske postupke strojnog učenja s jasnim tumačenjem te iz tog razloga odabiremo postupke temeljene na izgradnji klasifikacijskih pravila (postupak RIPPER) i stablima odluke (postupak C4.5). Sav postupak se ujedinjuje u web aplikaciji koja pruža učitavanje i klasifikaciju novog skupa mikciometrijskih krivulja kako bi se liječnicima olakšala dijagnoza različitih tipova poremećaja donjeg urinarnog trakta kod djece.

2. Medicinski opis problema

Prvo ćemo dati jednostavan opis anatomije i fiziologije nastanka mokraće, kako bismo bolje razumjeli proces mikciometrije i promatrane smetnje koje možemo očekivati u dobivenim krivuljama.

2.1 Anatomija

Na slici 1 vidimo osnovni prikaz mokraćnog sustava. Ulogu proizvodnje i izlučivanja mokraće obavljaju bubrezi. Bubrezi su dva tijela u obliku graha, svaki veličine šake. Nalaze se odmah ispod prsnog koša, jedan na svakoj strani kralježnice. Svaki dan, bubrezi filtriraju oko 120 do 150 litara krvi da bi proizveli oko 1 do 2 litre urina. Bubrezi rade bez prestanka; osoba ne kontrolira ono što rade. Bubrezi su odgovorni za tri funkcije: reguliranje koncentracije tekućina i iona, kiselo-baznu homeostazu i hormonsku zadaću u proizvodnji eritropoetina i renina. [1]



Slika 1. Osnovni prikaz mokraćnog sustava [2]

Mokraća, koja dolazi iz bubrega, prenosi se preko mokraćovoda u mokraćni mjehur. Ovdje će se pohraniti mokraća, dok se ne isprazni. Mokraćni mjehur smješten je u zdjelici između zdjeličnih kostiju, šupalj je, mišićav, organ u obliku balona koji se širi kako se ispunjava mokraćom. Iako osoba ne kontrolira funkciju bubrega, osoba kontrolira kada se mokraćni mjehur prazni. Pražnjenje mjehura je poznato kao mokrenje. Mjehur pohranjuje mokraću dok osoba ne pronađe odgovarajuće vrijeme i mjesto za mokrenje. Normalan mjehur djeluje kao rezervoar i može sadržavati 1,5 do 2 šalice mokraće. Koliko često osoba treba mokriti ovisi o tome koliko brzo bubrezi proizvode mokraću koji ispunjava mjehur. Mišići stijenke mjehura ostaju opušteni dok se mjehur puni mokraćom. Dok se mjehur puni do kapaciteta, signali koji se šalju u mozak govore osobi da uskoro pronađe zahod. Tijekom mokrenja, mokraćni se mjehur prazni kroz mokraćnu cijev, koja se nalazi na dnu mjehura. [2]

Tri skupa mišića rade zajedno kao brana, zadržavajući mokraću. Prvi skup su mišići same mokraćne cijevi. Područje gdje se mokraćna cijev pridružuje mjehuru je vrat mokraćnog mjehura. Vrat mokraćnog mjehura, sastavljen od drugog skupa mišića poznatih kao unutarnji sfinkter koji pomaže da urin ostane u mjehuru. Treći skup mišića je mišić dna zdjelice, također poznat kao vanjski sfinkter, koji okružuje i podržava mokraćnu cijev. Za mokrenje, mozak signalizira mišićnom zidu mjehura da se stegne, istisnuvši mokraću iz mjehura. U isto vrijeme, mozak signalizira sfinkterima da se opuste. Kako se sfinkteri opuštaju, mokraća izlazi iz mjehura kroz mokraćnu cijev. [2]

Sva tri skupa mišića sudjeluju zajedno kako bi se omogućilo kontrolirano mokrenje. Kontrolirano mokrenje je moguće tek kada smo sposobni kontrolirati mišić dna zdjelice, što se najčešće događa pri starosti od oko dvije godine. Nama su interesantni slučajevi disfunkcije mikcije kod djece koja kontrolirano mokre. Disfunkcija mikcije može biti uzrokovana neurološkim defektom, gdje takvi pacijenti imaju jasan uzrok za probleme s mikcijom. U neurološki normalnoj djeci, međutim, uzrok je nejasan. Kada nema anatomskih abnormalnosti, poteškoće s mikcijom se nazivaju funkcionalna inkontinencija. Disfunkcionalno pražnjenje je čest oblik simptoma donjeg urinarnog trakta (engl. *Lower Urinary Tract Symptoms*, kraće: LUTS). LUTS se može manifestirati u različitim oblicima, osobito: hitnost, učestalost, inkontinencija ili ponavljajuće infekcije

mokraćnog sustava. Hitnost je snažna, nagla, preplavljujuća, osjećaj potrebe za pražnjenjem. Nenormalno česta pražnjenja nazivaju se frekvencijom. Različiti oblici LUTS-a su preaktivan mokraćni mjehur, sindrom nedovoljno aktivnog mokraćnog mjehura i disfunkcionalna pražnjenja. Inkontinencija zbog disfunkcije faze punjenja je sindrom poriva, a inkontinencija zbog disfunkcije faze pražnjenja naziva se disfunkcionalno pražnjenje. [1]

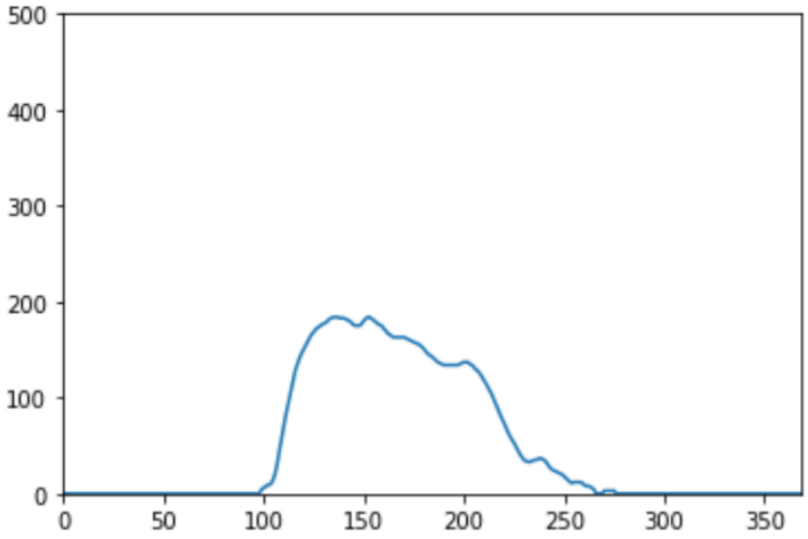
2.2 Mikciometrijske krivulje

Mikciometrijske krivulje nastaju mjerenjem brzine protoka (ml/sec) mokraće prilikom mokrenja u posudu za skupljanje. Dobiveni podatci se pohranjuju u jednodimenzionalni vremenski niz zabilježenog protoka (Q). Te se informacije zatim pretvaraju u graf zajedno s izračunatim mikciometrijskim parametrima. Oblik grafikona zajedno s izračunatim parametrima pomaže liječniku da protumači i ocijeni funkciju donjeg mokraćnog sustava, a zatim utvrđuje postoji li opstrukcija normalnog izlučivanja mokraće. Prema [3], mikciometrijski parametri koje doktori promatraju su:

- Količina izmokrenog volumena
- Prosjek ukupnog protoka
- Prosjek protoka nad kontinuiranim protokom
- Maksimalni protok
- Maksimalni protok nad kontinuiranim protokom
- Vrijeme do maksimalnog protoka
- Vrijeme pražnjenja
- Trajanje protoka
- Vrijeme odgode do protoka mokraće
- Ukupno vrijeme s pauzama između protoka mokraće

Protok mokraće ovisi o mnoštvu čimbenika: volumenu mokraće, psihološkom statusu, dobi i korištenju abdominalne muskulature za istiskivanje mokraće. Od svih njih najvažniji je volumen mokraće te tako minimalna količina mokraće za prihvatljivu interpretaciju rezultata ne smije biti manja od 150 ml, a poželjno je i postupak ponoviti nekoliko puta. [4]

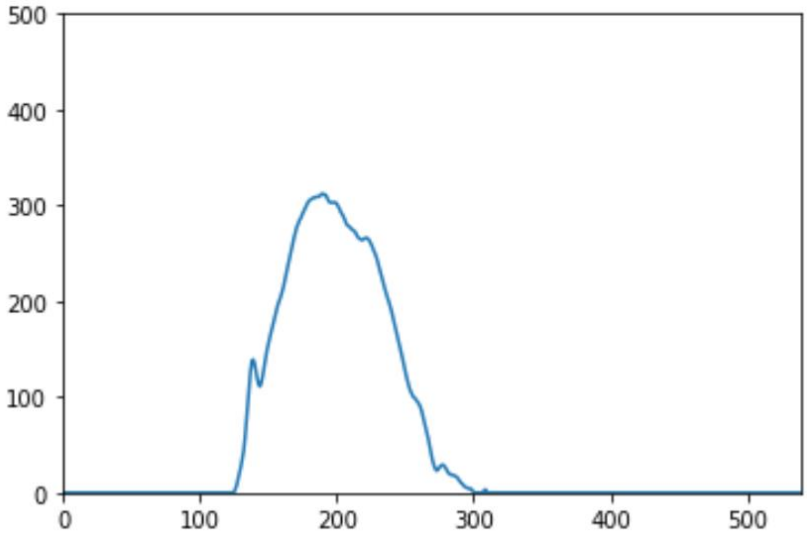
Mikciometrijske krivulje dijelimo na normalne i abnormalne oblike. Normalan oblik ima oblik zvona, slika 2.



Slika 2. Normalni oblik krivulje

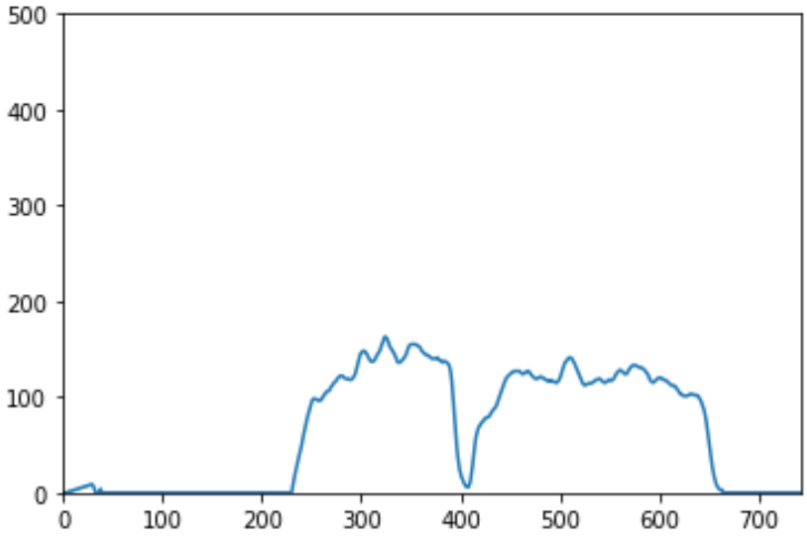
Abnormalne oblike možemo podijeliti u četiri grupe [5]:

1) krivulja oblika tornja – krivulja visoke amplitude s niskim trajanjem, upućuje na prekomjerno aktivan mokraćni mjehur, slika 3.



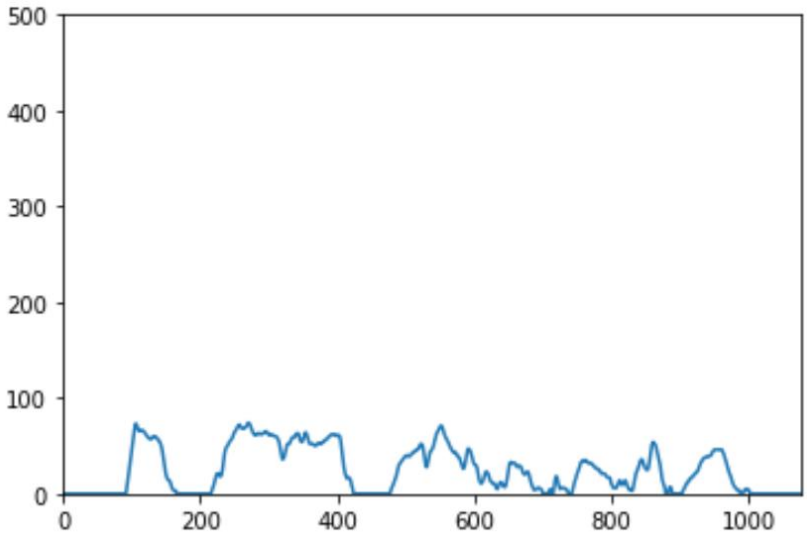
Slika 3. Krivulja oblika tornja

2) nazubljena (*staccato*) krivulja – karakterizira ju pulsirajući neregularan protok, upućuje na lošu usklađenost u radu sfinktera i mokraćnog mjehura, slika 4.



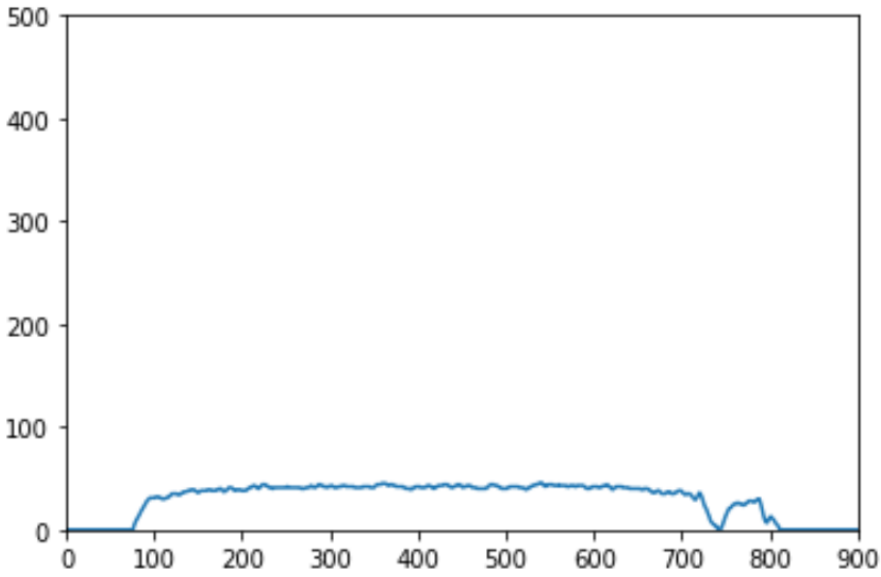
Slika 4. Nazubljen oblik krivulje

3) isprekidana krivulja – krivulja s više od jednog potpunog prekida mokrenja, upućuje na nedovoljnu aktivnost mokraćnog mjehura tj. da se pacijent služi ostalim mišićima kako bi potaknuo mokrenje, slika 5.



Slika 5. Isprekidan oblik krivulje

4) zaravnjena krivulja – produljena krivulja niske amplitude, upućuje na mogućnost opstrukcije unutar mokraćne cijevi, slika 6



Slika 6. Zaravnjeni oblik krivulje

3. Srodna istraživanja

J. Kortenbout i sur. [1] u svom radu analiziraju učinkovitost sustava za treniranje pravilne mikcije djece sa smetnjama u pražnjenju mokraće, zvanog „talking toilet“. Sustav je namijenjen kućnoj uporabi pacijenata. Korišteno je 20 značajki za treniranje ekspertnog sustava baziranog na neuronskim mrežama koji ocjenjuje mikciometrijsku krivulju sa 4 ocjene, od 1 do 5: sličnost krivulje krivulji oblika tornja, nazubljenoj krivulji, isprekidanoj krivulji i zaravnjenoj krivulji. Pokazno je da treniranje mikcije pomoću sustava „talking toilet“ nije poboljšao mikciju kod djece.

S. Abdovic i sur. [6] u svom radu analiziraju model duboke neuronske mreže za predikciju PUV-a (engl. *posterior urethral valve*) kod dječaka sa simptomima u donjem urinarnom traktu. Za predikciju je korištena četveroslojna neuronska mreža s ulaznim parametrima: dob, maksimalni protok, vrijeme do maksimalnog protoka, volumen mokraće, vrijeme pražnjenja, ukupno vrijeme, prosjek protoka. Mikciometrijske krivulje korištene u ovom istraživanju su uzete od pacijenata dijagnosticiranim s PUV-om i skupine pacijenata za koje je provedena kontrola i utvrđen izostanak PUV-a. U ovom radu pokazane su značajke koje pozitivno utječu na klasifikaciju PUV-a, postignuta točnost modela iznos 92.7%.

4. Obrada mikciometrijskih krivulja

Mikciometrijski zapisi su prikupljeni u suradnji s Klinikom za dječje bolesti Zagreb. Prikupljeno je 210 anonimnih mikciometrijskih zapisa. Izlučene su 33 ekspertne značajke određene u suradnji doc. Alana Jovića i dr. Slavena Abdovića. Popis značajki dan je u tablici 1. Značajke možemo podijeliti na značajke vremenske domene (1-19) te značajke frekvencijske domene (20-33). Za analizu frekvencijske domene korištena je diskretna Fourierova transformacija. Program za izlučivanje značajki napisao je doc. Jović u programskom jeziku Java. Program podržava učitavanje značajki iz datoteka zapisanih u *csv* formatu. Prva linija datoteke ne sadrži mikciometrijski zapis a svaka iduća počinje prikladnim identifikatorom zapisa i nakon toga nizom očitanih vrijednosti. Izračunate značajke zapisuju se u dvije nove datoteke, jednu *csv* formata drugu *arff* formata. Također, moguće je unijeti pri pokretanju programa i frekvenciju uzorkovanja mikciometrijske krivulje.

Tablica 1. Popis korištenih značajki s kratkim opisom

Naziv značajke	Pojašnjenje
PeakFlowRate	Maksimalni zabilježeni protok
Volume	Ukupni izmokreni volumen
TimeToPeakFlowRate	Vrijeme potrebno do maksimalnog protoka
TimeFromPeakFlowRateTillFlowEnd	Vrijeme od maksimalnog protoka do kraja protoka
Acceleration	Ubrzanje do maksimalnog protoka
Decceleration	Usporavanje od maksimalnog protoka do kraja
PeakFlowDuration	Ukupno vrijeme za koje je protok bio veći od 90% maksimalnog
MaximumInterruptionTime	Trajanje najduljeg prekida protoka
MinimumInterruptionTime	Trajanje najkraćeg prekida protoka
NumberOfInterruptions	Broj prekida protoka
FlowDuration	Trajanje protoka
VoidingDuration	Ukupno vrijeme mikcije
AverageFlowRate	Prosječna vrijednost protoka
TopMiddleScore	Provjera je li maksimalni protok postignut u sredini krivulje
IncreaseDecrease10perc	Broj podizanja i spuštanja amplitude protoka od 10% maksimalnog protoka

IncreaseDecrease30perc	Broj podizanja i spuštanja amplitude protoka od 30% maksimalnog protoka
IncreaseDecrease50perc	Broj podizanja i spuštanja amplitude protoka od 50% maksimalnog protoka
ZeroDerivative	Ukupno vrijeme protoka za koje je derivacija jednaka 0
SteepnessScore	Ocjena strmosti krivulje nakon maksimalnog protoka
FourierCutScore0_3	Razlika između originalnog signala i filtriranog niskopropusnim filterom 0.3 Hz.
FourierCutScore0_5	Razlika između originalnog signala i filtriranog niskopropusnim filterom 0.5 Hz.
FourierCutScore0_7	Razlika između originalnog signala i filtriranog niskopropusnim filterom 0.7 Hz.
FourierCutScore0_3N	Normalizirana vrijednost *FourierCutScore0_3*
FourierCutScore0_5N	Normalizirana vrijednost *FourierCutScore0_5*
FourierCutScore0_7N	Normalizirana vrijednost *FourierCutScore0_7*
ZeroDerivativeAccent	Isto kao *ZeroDerivative*, ali nad filtriranim signalom niskopropusnim filterom od 0.3 Hz
IncreaseDecrease10percAccent	Isto kao *IncreaseDecrease10perc*, ali nad filtriranim signalom niskopropusnim filterom od 0.3 Hz
IncreaseDecrease30percAccent	Isto kao *IncreaseDecrease30perc*, ali nad filtriranim signalom niskopropusnim filterom od 0.3 Hz
IncreaseDecrease50percAccent	Isto kao *IncreaseDecrease50perc*, ali nad filtriranim signalom niskopropusnim filterom od 0.3 Hz
FourierPSD0_02Percent	Omjer između PSD-a* vrijednosti manje od 0.02 Hz i ostatka spektra
FourierPSD0_05Percent	Omjer između PSD-a vrijednosti manje od 0.05 Hz i ostatka spektra
FourierPSD0_1Percent	Omjer između PSD-a vrijednosti manje od 0.1 Hz i ostatka spektra
FourierPSD0_2Percent	Omjer između PSD-a vrijednosti manje od 0.2 Hz i ostatka spektra

*spektralna gustoća snage (engl. *Power Spectral Density,* kraće: PSD)

Računanje značajki provodimo prema danim uputama:

- *PeakFlowRate* se računa kao najveća vrijednost u vremenskom zapisu.
- *Volume* se računa kao površina ispod krivulje.
- *TimeToPeakFlowRate* se računa kao vrijeme od početka protoka do maksimalnog protoka.
- *TimeFromPeakFlowRateTillFlowEnd* se računa kao vrijeme od maksimalnog protoka do kraja protoka.
- *Acceleration* se računa kao omjer vrijednosti *PeakFlowRate i TimeToPeakFlowRate*
- *Decceleration* se računa kao omjer vrijednosti *PeakFlowRate i TimeFromPeakFlowRateTillFlowEnd*
- *PeakFlowDuration* se računa kao vrijeme za koje je protok bio iznad 90% *PeakFlowRate.*
- *MaximumInterruptionTime* se računa kao vrijeme trajanja najduljeg prekida.
- *MinimumInterruptionTime* se računa kao vrijeme trajanja najkraćeg prekida.
- *NumberOfInterruptions* se računa kao broj zabilježenih prekida protoka.
- *FlowDuration* se računa kao ukupno trajanje protoka bez prekida.
- *VoidingDuration* se računa kao ukupno trajanje mikcije. Od prvog trenutka u kojem je zabilježen protok do zadnjega.
- *AverageFlowRate* se računa kao prosječna vrijednost protoka
- *TopMiddleScore* se računa kao provjera nalazi li se *TimeToPeakFlowRate* imeđu 40% i 60% krivulje protoka. Inače se računa kao:

  $$100 - d * 100 \quad (1)$$

  gdje je *d* omjer udaljenosti maksimalnog protoka do sredine krivulje i polovice krivulje.
- *IncreaseDecrease10perc* se računa kao:

  $$\frac{broj\ prelazaka}{trajanje\ protoka} * 100 \quad (2)$$

  gdje se broj prelazaka računa kao broj prelazaka iznad ili ispod 10% maksimalnog protoka.

- *IncreaseDecrease30perc* se računa prema formuli (2), gdje se broj prelazaka računa kao broj prelazaka iznad ili ispod 30% maksimalnog protoka.
- *IncreaseDecrease50perc* se računa prema formuli (2), gdje se broj prelazaka računa kao broj prelazaka iznad ili ispod 50% maksimalnog protoka.
- *ZeroDerivative* se računa kao omjer vremena za koje je derivacija krivulje jednaka nuli i ukupnog vremena protoka bez prekida.
- *SteepnessScore* se računa kao omjer vremena potrebnog da protok završi od kada se spusti ispod 20% maksimalnog protoka i vremena potrebnog da protok završi od pojave maksimalnog protoka.
- *FourierCutScore0_3* se računa prema formuli:

  $$sum(FourierCut\text{^}2)/100 \quad (3)$$

  gdje je *FourierCut* razlika između protoka i filtriranog protoka niskopropusnim filterom od 0.3 Hz.
- *FourierCutScore0_5* se računa prema formuli (3), gdje je *FourierCut* razlika između protoka i filtriranog protoka niskopropusnim filterom od 0.5 Hz.
- *FourierCutScore0_7* se računa prema formuli (3), gdje je *FourierCut* razlika između protoka i filtriranog protoka niskopropusnim filterom od 0.7 Hz.
- *FourierCutScore0_3N* se računa kao vrijednost *FourierCutScore0_3* normalizirana duljinom protoka.
- *FourierCutScore0_5N* se računa kao vrijednost *FourierCutScore0_5* normalizirana duljinom protoka.
- *FourierCutScore0_7N* se računa kao vrijednost *FourierCutScore0_7* normalizirana duljinom protoka.
- *ZeroDerivativeAccent* se računa kao *ZeroDerivative* nad filtriranim signalom niskopropusnim filterom od 0.3 Hz.
- *IncreaseDecrease10percAccent* se računa kao *IncreaseDecrease10perc* nad filtriranim signalom niskopropusnim filterom od 0.3 Hz.
- *IncreaseDecrease30percAccent* se računa kao *IncreaseDecrease30perc* nad filtriranim signalom niskopropusnim filterom od 0.3 Hz.

- *IncreaseDecrease50percAccent* se računa kao *IncreaseDecrease50perc* nad filtriranim signalom niskopropusnim filterom od 0.3 Hz.
- *FourierPSD0_02Percent* se računa kao omjer između PSD-a vrijednosti manje od 0.02 Hz i ostatka spektra.
- *FourierPSD0_05Percent* se računa kao omjer između PSD-a vrijednosti manje od 0.05 Hz i ostatka spektra.
- *FourierPSD0_1Percent* se računa kao omjer između PSD-a vrijednosti manje od 0.1 Hz i ostatka spektra.
- *FourierPSD0_2Percent* se računa kao omjer između PSD-a vrijednosti manje od 0.2 Hz i ostatka spektra.

5. Dubinska analiza

Krajnji cilj dubinske analize je pronaći optimalni broj grupa mikciometrijskih krivulja za dani skup podataka s ciljem precizne i objektivne kvantifikacije tipova obrazaca mikciometrijskih krivulja te opisati proces odluke kojoj grupi pripada novi zapis. Za pronalazak optimalnog broja grupa odabrani su postupci: k-srednjih vrijednosti, EM i DBSCAN. Za klasifikaciju novog zapisa odabrani su postupci strojnog učenja s jasnim tumačenjem: stablo odluke C4.5 i postupak klasifikacijskih pravila RIPPER. Na ovaj način postiže se točno i automatizirano razvrstavanje vremenskih nizova mikciometrije u nekoliko grupa, te se time liječnicima olakšava dijagnoza različitih tipova poremećaja donjeg urotrakta kod djece.

5.1 Postupak K-srednjih vrijednosti

Postupak K-srednjih vrijednosti grupira podatke u *K* (unaprijed zadan broj) grupa. Postupak pokušava razdvojiti uzorke u skupine jednake varijance, minimizirajući kriterij kvadratne sume unutar grupa. Postupak dijeli skup primjera u nepovezane grupe, od kojih je svaka opisana srednjom vrijednošću primjera u grupi. Sredine se obično nazivaju „centroidi“ grupa. Najčešće, dobiveni centroidi ne pripadaju početnom skupu primjera. Postupak ima cilj odabrati centroide koji minimiziraju kvadratne sume unutar grupa odnosno formulu 4:

$$\sum_{k=1}^{K} \sum_{i=1}^{N} b_k^{(i)} ||x^{(i)} - \mu_k||^2 \qquad (4)$$

Varijabla *K* predstavlja broj grupa, *N* predstavlja ukupan broj primjera, *b* predstavlja pripadnost promatranog primjera grupi *k*, μ predstavlja trenutni centroid grupe *k*.

Varijanca se može promatrati kao mjera koherentnosti dobivenih grupa. Ova mjera ima nedostatak. Varijanca čini pretpostavku da su grupe konveksne i izotropne, što nije uvijek slučaj. Loše reagira na izdužene nakupine ili nepravilne oblika značajki.

Damo li dovoljno vremena, postupak će uvijek konvergirati, no to može biti u lokalni optimum. To uvelike ovisi o inicijalnim centroidima. Kao rezultat toga, izračun se često radi nekoliko puta, s različitim inicijalizacijama centroida.

Prvi korak postupka je odabir početnih centroida, a najosnovnija metoda je odabir uzoraka iz skupa podataka. To možemo učiniti slučajnim odabirom, no bolje je koristiti postupak koji uzima u obzir da je bolje odabrati međusobno udaljene primjere za centroide, poput postupka *K-means*++. Nakon inicijalizacije, postupak se sastoji od petlje između dva koraka. Prvi korak dodjeljuje svaki uzorak najbližem središtu. Drugi korak stvara nove centroide, uzimajući srednju vrijednost svih uzoraka dodijeljenih svakom prethodnom središtu. Izračunava se razlika između starog i novog centroida i postupak ponavlja ova posljednja dva koraka sve dok ova vrijednost ne bude manja od praga. Drugim riječima, ponavlja se sve dok se centroidi ne pomaknu značajno.

Vremenska kompleksnost ovog postupka je: $O(nKdi)$, gdje *n* predstavlja broj primjera, *K* predstavlja broj grupa, *i* predstavlja broj iteracija potrebnih do konvergencije, a *d* predstavlja dimenzionalnost primjera odnosno broja značajki.

Pseudokod postupka preuzet je iz 19. predavanja predmeta Strojno učenje profesora Jana Šnajdera. [7]

1: **inicijaliziraj** centroide $\mu_k, \mathrm{k} = 1, \ldots, \mathrm{K}$

2: **ponavljaj**

3: za svaki $x^{(i)} \in D$

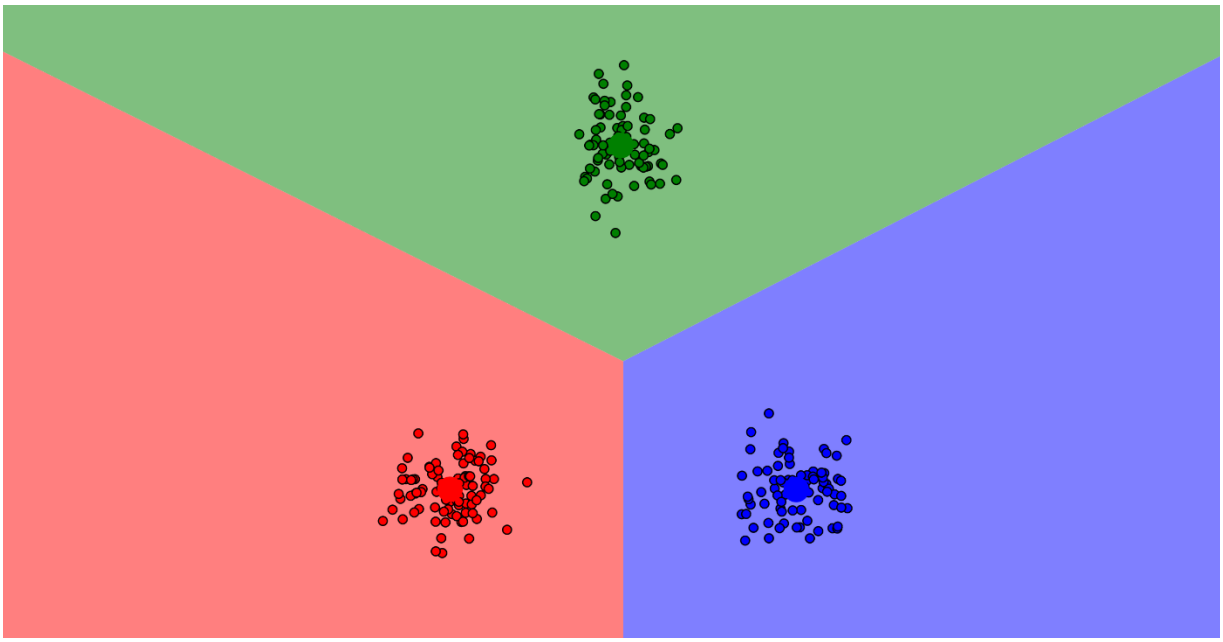
4: $b_k^{(i)} \leftarrow \begin{cases} 1 & ako\ k = argmin_j \left|\left|x^{(i)} - \mu_j\right|\right| \\ 0 & inače \end{cases}$

5: $za\ svaki\ \mu_k, k = 1, \ldots, K$

6: $\mu_k \leftarrow \frac{\sum_{i=1}^{N} b_k^{(i)} x^{(i)}}{\sum_{i=1}^{N} b_k^{(i)}}$

7: $\boldsymbol{dok\ \mu_k}\ ne\ konvergiraju$

Vizualizaciju kako postupak *k*-srednjih vrijednosti dijeli prostor prema dobivenim grupama s prikazanim centroidima za svaku grupu možemo vidjeti na primjeru prikazanom na slici 7.



Slika 7. Vizualizacija postupka K-sredina

5.2 Postupak EM

Postupak EM (engl. *expectation–maximization*) je učinkovit iterativni postupak za izračunavanje maksimalne vjerojatnosti (engl. *maximum likelihood*, kraće: ML) u prisutnosti nepoznatih ili skrivenih podataka. U ML-procjeni želimo procijeniti parametre modela koji su najvjerojatniji za promatrane podatke. Svaka iteracija postupka EM sastoji se od dva procesa: E-korak i M-korak.

- E-korak (engl. *expectation*), nepoznati podaci se procjenjuju na temelju promatranih podataka i trenutačne procjene parametara modela. To se postiže izračunavanjem očekivanja potpune izglednosti uz fiksirane parametre u iteraciji.[8]
- U M-koraku, funkcija vjerojatnosti je maksimirana pod pretpostavkom da su podaci koji nedostaju poznati. Procjena nedostajućih podataka iz E-koraka koristi se umjesto stvarnih nedostajućih podataka. Konvergencija je osigurana jer postupak jamči povećanje vjerojatnosti u svakoj iteraciji. [8]

U ovom radu je korišten model Gaussove mješavine (engl. Gaussian mixture model), postupak iz paketa *sklearn.mixutre* iz pythonove biblioteke *scikit-learn* namijenjene strojnom učenju. Postupak Gaussove mješavine pretpostavlja da svi primjeri dolaze iz mješavine konačnog broja Gaussovih distribucija s nepoznatim parametrima. Gaussovu mješavinu možemo promatrati kao poopćenje postupka k-srednjih vrijednosti gdje se zamjenjuje varijabla pripadnosti $b_k^{(i)} \in \{0,1\}$ s varijablom odgovornosti $h_k^{(i)} \in [0,1]$. Odgovornost predstavlja vjerojatnost da je primjer $x^{(i)}$ generirala grupa *k*. [9]

Pseudokod kombinacije GMM i EM postupaka preuzet je iz 20. predavanja predmeta Strojno učenje profesora Jana Šnajdera. [9]

1: **inicijaliziraj** parametre $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^{K}$

2: **ponavljaj** do konvergencije $\log -$ izglednosti ili parametara

3: $\mathbf{E} - \mathbf{korak}$

4: Za svaki primjer $x^{(i)} \in D$ i za svaku komponentu $k = 1, \ldots, K$:

5: $h_k^{(i)} \leftarrow \frac{p\left(x^{(i)} \middle| \mu_k, \Sigma_k\right)\pi_k}{\sum_{j=1}^{K} p\left(x^{(i)} \middle| \mu_j, \Sigma_j\right)\pi_j}$

6: $\mathbf{M} - \mathbf{korak}$

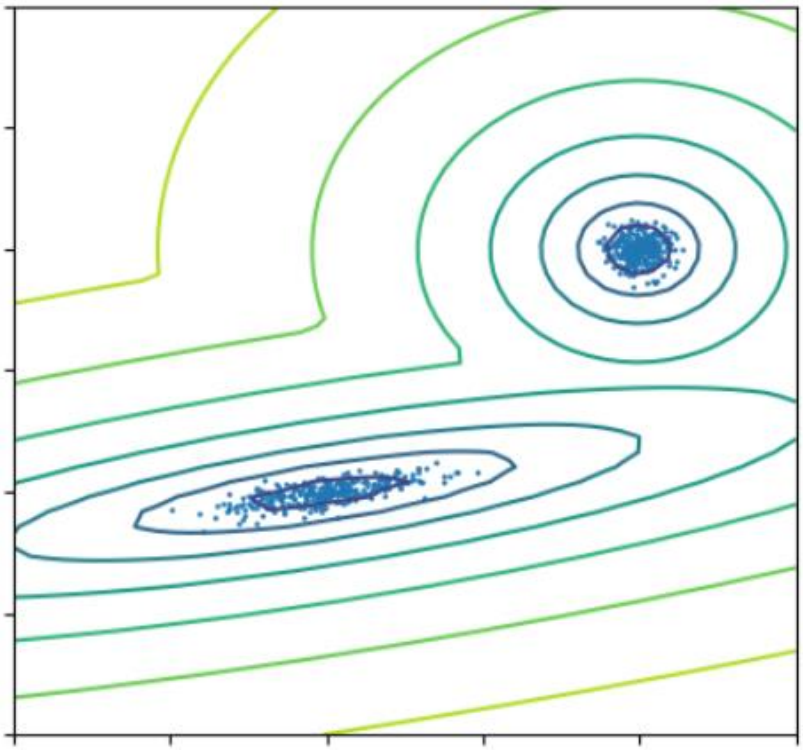
7: Za svaku komponentu $k = 1, \ldots, K$:

8: $\mu_k \leftarrow \frac{\sum_i h_k^{(i)} x^{(i)}}{\sum_i h_k^{(i)}}, \boldsymbol{\Sigma}_k \leftarrow \frac{\sum_i h_k^{(i)} \left(x^{(i)} - \mu_k\right)\left(x^{(i)} - \mu_k\right)^T}{\sum_i h_k^{(i)}}, \pi_k \leftarrow \frac{1}{N}\sum_{i=1}^{N} h_k^{(i)}$

9: Izračunaj trenutačnu vrijednost $\log -$ izglednosti

10: $\ln L(\theta|D) = \sum_{i=1}^{N} \ln \sum_{k=1}^{K} \pi_k \, p(x^{(i)}|\mu_k, \Sigma_k)$

Vizualizaciju kako postupak GMM-EM dijeli prostor prema dobivenim grupama s prikazanim plohama jednake vjerojatnosti možemo vidjeti na slici 8.



Slika 8. Vizualizacija postupka GMM-EM [17]

5.3 Postupak DBSCAN

Postupak DBSCAN (engl. *Density-Based Spatial Clustering of Applications with Noise*) koristi jednostavnu procjenu minimalne gustoće na temelju praga za broj susjeda, *minPts*, unutar radijusa ε (s proizvoljnom mjerom udaljenosti). Primjeri s više od *minPts* susjeda unutar tog radijusa (uključujući i točku upita) smatraju se jezgrenom točkom. Ideja postupka DBSCAN je pronaći ona područja koja zadovoljavaju ovu minimalnu gustoću i koja su odvojena područjima manje gustoće. Zbog povećane učinkovitosti, DBSCAN ne provodi procjenu gustoće između točaka. Umjesto toga, svi susjedi unutar polumjera ε jezgre točke smatraju se dijelom iste grupe kao i središte (naziva se izravna dosezljiva gustoća). Ako je bilo koji od ovih susjeda opet ključna točka, njihova susjedstva su uključena tranzitivno (dostupna po gustoći). Točke s manje

od *minPts* susjeda u ovom postupku nazivaju se graničnim točkama, a sve točke unutar iste grupe su dostupne po gustoći. Točke koje nisu dostupne po gustoći iz bilo koje točke jezgre smatraju se šumom i ne pripadaju niti jednoj grupi. [10]

Dok parametar *minPts* prvenstveno kontrolira koliko je postupak tolerantan prema šumu (na šumovitim i velikim skupovima podataka može biti poželjno da se poveća ovaj parametar), parametar $\boldsymbol{\varepsilon}$ je presudan za funkciju udaljenosti i obično ga se treba optimirati na promatranom skupu podataka. $\boldsymbol{\varepsilon}$ kontrolira lokalno susjedstvo točaka. Kada je odabrana premala vrijednost, većina podataka uopće neće biti grupirana (tj. bit će označena kao šum). Kada je odabrana prevelika vrijednost, onda ona uzrokuje spajanje bliskih grupa u jednu grupu, a u najgorem slučaju se cijeli skup podataka vraća kao jedna grupa. Za razliku od prošla dva postupka, postupku DBSCAN ne treba unaprijed zadati broj grupa. Također, ovaj postupak posjeduje svojstvo prepoznavanja šuma u podatcima i ne radi pretpostavku o distribuciji grupa tj. obliku koji mogu poprimiti u prostoru. Pseudokod postupka preuzet je iz rada. [10]

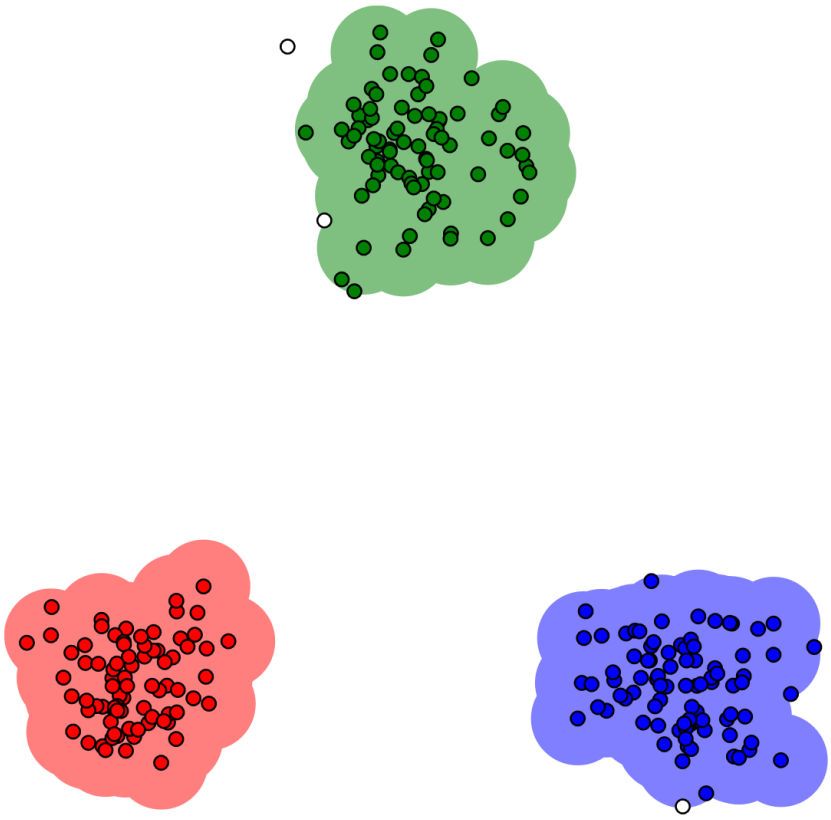
1: $DBSCAN(\boldsymbol{DB}: baza\ podataka,$ **ε**: polumjer, **minPts**: prag gustoće, **dist**: funkcija udaljenosti$)$:
2: $za\ svaki\ primjer\ \boldsymbol{p}\ iz\ DB$:
3: $ako\ \boldsymbol{oznaka}(p)\ \neq nedefiniran\ onda\ nastavi$
4: $susjedi\ \boldsymbol{N}\ \leftarrow RangeQuery(DB, dist, p, \varepsilon)$
5: $ako\ |N| < minPts\ onda$:
6: $oznaka(p) \leftarrow Šum$
7: $nastavi$
8: $\boldsymbol{c}\ \leftarrow oznaka\ iduće\ grupe$
9: $oznaka(p) \leftarrow c$
10: $\boldsymbol{S}\ \leftarrow N\ \{p\}$
11: $za\ svaki\ \boldsymbol{q}\ iz\ S$:
12: $ako\ oznaka(q) = Šum\ onda\ oznaka(q) \leftarrow c$
13: $ako\ oznaka(q) \neq nedefiniran\ onda\ nastavi$
14: $susjedi\ \boldsymbol{N}\ \leftarrow\ RangeQuery(DB, dist, q, \varepsilon)$
15: $oznaka(q)\ \leftarrow c$
16: $ako\ |N| < minPts\ onda\ nastavi$

17: $S \leftarrow S \cup N$

Metoda pozivana u pseudokodu postupka može se implementirati kao:

1: $RangeQuery(\boldsymbol{DB}: baza\ podataka,$ **dist**: funkcija udaljenosti, **q**: primjer, **ε**: polumjer):
2: $susjedi = prazna\ lista$
3: $za\ svaki\ primjer\ \boldsymbol{P}\ iz\ DB$:
4: $ako\ dist(q, P) \leq \varepsilon\ onda$:
5: $susjedi = susjedi \cup \{P\}$
6: $vrati\ susjedi$

Vizualizaciju kako postupak DBSCAN dijeli prostor prema dobivenim grupama s prikazanim primjerima označenima kao šum (bijeli kružići) možemo vidjeti na slici 9.



Slika 9. Vizualizacija postupka DBSCAN

5.4 Mjere vrednovanja grupiranja

Vrednovanje izvedbe postupaka grupiranja nije trivijalno kao što je to slučaj s brojanjem broja pogrešaka ili određivanja preciznosti nadgledanog klasifikacijskog postupka. Popularni pristupi vrednovanju grupiranja uključuju „internu“ procjenu, gdje je grupiranje sažeto na jednu ocjenu kvalitete i „vanjsku“ procjenu, gdje se grupiranje uspoređuje s postojećom klasifikacijom „temeljne istine“, koja je „ručna“ procjena od strane stručnjaka. [11]

Dobiveni podatci mikciometrijskih krivulja ne sadrže skup primjera „temeljne istine“, odnosno skup primjera unaprijed određenih od strane stručnjaka. Temeljem te činjenice u ovom radu promatrat ćemo samo „internu“ procjenu grupiranja. Odabrane su 3 mjere: koeficijent siluete, indeks Calinski-Harabasz i indeks Davies-Bouldin.

5.4.1 Koeficijent siluete

Koeficijent siluete je mjera koliko je primjer sličan svojoj grupi u usporedbi s drugim grupama. Silueta se kreće u rasponu od -1 do +1, gdje visoka vrijednost ukazuje da je objekt dobro usklađen s vlastitom grupom i loše se slaže sa susjednim grupama. Vrijednosti oko nule ukazuju na preklapanje grupa. Nedostatak koeficijenta siluete je privrženost konveksnim grupama za koje općenito daje veće rezultate. [12]

Kako bismo dobili koeficijent siluete, za svaki primjer iz skupa podataka definiramo prosječnu udaljenost između promatranog primjera i svih ostalih primjera iste grupe.

$$a(i) = \frac{1}{|C_i|-1} \sum_{j \in C, i \neq j} d(i,j) \tag{5}$$

Možemo interpretirati *a(i)* kao mjeru koliko dobro je primjer *i* pridružen svojoj grupi. Zatim definiramo prosječnu različitost primjera *i* od svih drugih primjera iz iduće najbliže grupe.

$$b(i) = min_{i \neq j} \frac{1}{|C_j|} \sum_{j \in C} d(i,j) \tag{6}$$

Koeficijent siluete za jedan primjer računamo po formuli (7). Za usporedbu postupaka promatramo prosječnu vrijednost koeficijenata siluete svih primjera. [7]

$$s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}} \quad (7)$$

5.4.2 Indeks Calinski-Harabasz

Indeks Calinski-Harbasz je mjera poznata kao kriterij omjera varijance (engl. *Variance Ratio Criterion*). Ukazuje na omjer disperzije podataka između grupa i disperzije podataka unutar grupa. Mjera se računa prema formuli (8). [13]

$$C = \frac{N-K}{K-1}\frac{BGSS}{WGSS} \quad (8)$$

N predstavlja ukupan broj primjera dok je *K* broj grupa. Vrijednost WGSS predstavlja varijancu unutar svake grupe, BGSS predstavlja ukupnu varijancu između grupa. $C_q$predstavlja skup primjera iz grupe *q*, a $c_q$predstavlja centar grupe *q.* [13]

$$WGSS = \sum_{q=1}^{k}\sum_{x \in C_q}(x - c_q)(x - c_q)^T \quad (9)$$

$$BGSS = \sum_q n_q(c_q - c)\,(c_q - c)^T \quad (10)$$

Visoka vrijednost BGSS nam govori da su centroidi grupa međusobno udaljeni a znamo da će se vrijednost WGSS smanjivati povećanjem broja grupa. Indeks Calinski-Harbasz je veći što su grupe gušće grupirane i bolje razdvojene. [13]

5.4.3 Indeks Davies-Bouldin

Indeks Davies-Bouldin je definiran formulom (11). *N* predstavlja broj primjera u skupu podataka, c_i predstavlja centroid grupe *i*, σ_i predstavlja prosječnu udaljenost svih elemenata unutar grupe *i* do centroida te grupe. Nedostatak indeksa Davies-Bouldin je privrženost konveksnim grupama za koje općenito daje veće rezultate. Također, da bi mjera imala smisla, potrebno je koristiti mjeru udaljenosti *d* istu kao i za postupak grupiranja. [14]

$$DB = \frac{1}{N}\sum_{i=1}^{N} max_{j \neq i}\left(\frac{\sigma_i+\sigma_j}{d(c_i,c_j)}\right) \quad (11)$$

5.5 Klasifikator C4.5

C4.5 je postupak koji se koristi za generiranje stabla odluke. C4.5 je nadogradnja na postupak ID3 koji uklanja ograničenje da značajke moraju biti kategorijske varijable. Postupak kao izlaz daje stablo gdje je svaki krajnji čvor (list) odluka (grupa) i svaki nekonačni čvor predstavlja test. Svaki list predstavlja odluku o pripadnosti grupi podataka Postupak gradi inicijalno stablo tehnikom podijeli pa vladaj (engl. *divide-and-conquer*): ako svi primjeri pripadaju istoj grupi ili je skup promatranih primjera malen, dodaj list označen s najčešćom grupom iz skupa promatranih primjera. Inače, odaberi test nad jednom značajkom. Dodaj novi čvor s odabranim testom s po jednom granom za svako moguće rješenje testa, podijeli promatrani skup podataka prema testu i ponovi postupak za svaki podskup. Zatim se nad stablom obavlja podrezivanje (engl. pruning) kako bi se izbjegla prenaučenost. Postupak se može sažeti u pseudokod: [5]

$C4.5(\boldsymbol{D}\ skup\ podataka)$:
1: $Stablo = \{\quad\}$
2: $ako\ je\ D$ "čisto" $ili\ |D| < graničniBroj\ onda$:
3: $vrati\ list(D)$;
4: $za\ svaku\ značajku\ a \in D$:
5: $izračunaj$ "*Information gain ratio" ako bismo podijelili stablo po* a
6: $a_{best} = a\ s\ najvećom\ vrijednosti$ "*Information gain ratio"*
7: $Stablo =$ čvor s a_{best} *testom*
8: $D_v = podijeli\ D\ prema\ a_{best}\ testu$
9: $za\ svaki\ D_i\ iz\ D_v$:
10: $Stablo_i = C4.5(D_i)$
11: $dodaj\ Stablo_i\ u\ granu\ Stabla$
12: $vrati\ Stablo$

5.6 Klasifikator RIPPER

RIPPER (engl. *Repeated Incremental Pruning to Produce Error Reduction*) je postupak za izgradnju skupa indukcijskih pravila. Prednost skupa pravila za klasifikaciju je jasno tumačenje dolaska do odluke. Klasifikacijska pravila imaju izražen problem prenaučenosti, a RIPPER da bi izbjegao taj problem za razliku koristi tehniku REP (engl. *reduced error pruning*). Tehnika REP dijeli podatke na skup za učenje i skup za podrezivanje. Pravila se izgrade nad skupom za učenje, a zatim se dobiveni skup pravila pojednostavljuje primjenom operatora podrezivanja. U svakom koraku rezidbe odabire se operator koji daje najveće smanjenje pogreške nad skupom za rezidbu. Rezidba završava kada svi operatori daju povećanje pogreške nad skupom za rezidbu. RIPPER predstavlja optimiranu inačicu postupka IREP (engl. *Incremental Reduced Error Pruning*). [16]

Pseudokod postupka dan je u nastavku: [16]

$RIPPER(\boldsymbol{E}: skup\ primjera)$:

1: $za\ svaku\ grupu\ \boldsymbol{C}, od\ najmanje\ prema\ najvećoj$

2: $\boldsymbol{GRADI}$:

3: $Podijeli\ E\ u\ set\ za\ \boldsymbol{TRENING}\ \ i\ set\ za\ \boldsymbol{REZIDBU}\ omjer\ 2:1$

4: $Ponavljaj\ dok$:

5: $a)\ nisu\ pokriveni\ svi\ primjeri\ iz\ C$

6: $b)\ DL\ mjera\ je\ za\ 64\ veća\ od\ najmanje\ vrijednosti\ DL\ dosad\ pronađene$

7: $c)\ stupanj\ pogreške > 50\%$

8: $faza\ UČENJA: izgradi\ pravilo\ pohlepnom\ tehikom\ dodavanjem\ antecedenata$

9: $dok\ pravilo\ ne\ postane\ 100\%\ precizno\ testiranjem\ svih\ vriednosti\ značajki$

10: $i\ odabirom\ antecedenata\ s\ najvećom\ vrijednosti\ mjere\ p\left[\log\left(\frac{p}{t}\right) - \log\left(\frac{P}{T}\right)\right]$

11: $faza\ PODREZIVANJA: primjenjuj\ tehnike\ podrezivanja\ od\ zadnjeg\ antecententa\ prema\ prv$

12: $sve\ dok\ vrijednost\ pravila\ raste\ prema\ mjeri\ \frac{p+1}{t+2}$

13: $\boldsymbol{OPTIMIRAJ}$:

14: $GENERIRAJ\ VARIJANTE$:

15: *za svako pravilo R za grupu C*:

16: *podijeli E u nove skupove za UČENJE i PODREZIVANJE*

17: *izbaci iz skupa za ODREZIVANJEP sve primjere koji su pokriveni drugim pravilima*

18: *ponovi fazu UČENJA i fazu PODREZIVANJA kako bi se izgradila dva nova pravila*

19: *nad novo podijeljenim podatcima*:

20: R_1 *je nanovo izgrađeno pravilo*

21: R_2 *je pravilo izgrađeno pohlepnim dodavanjem antecedenata na pravilo R*

22: *odradi fazu PODREZIVANJA koristeći mjeru* $\frac{p + n'}{T}$

23: *ODABERI NAJBOLJE*: *zamijeni R sa* R, R_1 *ili* R_2 *pravilom koje ima najmanji DL*

24: ***BRISANJE***:

25: *ako postoje nepokriveni primjeri za grupu C vrati se na korak GRADI*

26: *ČIŠĆENJE*:

27: *izračunaj DL za svko pravilo iz skupa i pobriši ga ako povećava DL*

28: *pobriši primjere iz E pokrivene izgrađenim pravilima*

Varijabla *p* predstavlja broj pozitivnih primjera pokrivenih pravilom („true positive“), varijabla *n* predstavlja broj negativnih primjera pokrivenih pravilom („false negative“), varijabla *t* predstavlja ukupan broj primjera pokriven pravilom ($t = p + n$), varijabla n' predstavlja broj negativnih primjera koji nisu pokriveni ovim pravilom („true negative), varijabla *P* predstavlja broj pozitivnih primjera promatrane grupe, varijabla *N* predstavlja broj negativnih primjera promatrane grupe, varijabla *T* predstavlja ukupan broj primjera promatrane grupe.

5.7 Programsko rješenje

Programsko rješenje izlučivanja značajki (UroflowFeatureExtractor_1_0) iz mikciometrijskih krivulja napisano je u programskom jeziku Java. Autor rješenja je doc. Alan Jović. Programsko rješenje je napravljeno u stilu konzolne aplikacije koju kontroliramo uz pomoć argumenata iz komandne linije. Aplikacija prima putanju do datoteke s zapisima mikciometrijskih krivulja u *csv* formatu gdje se prva linija preskače a svaka iduća sadrži po jedan zapis u formatu „id, niz vrijednosti.“ Također prima i putanju do izlazne datoteke u koju zapisuje izlučene značajke. Zapisuju se dvije izlazne datoteke, jedna u *csv* formatu, druga u *arff* formatu. Kao treći argument moguće je predati frekvenciju uzorkovanja pri snimanju mikciometrijskih krivulja, inače se koristi frekvencija od 10 Hz.

Za provođenje postupka grupiranja korišteni su paketi iz pythonove biblioteke *scikit-learn* u verziji *0.21.1*. Programski kod grupiranja napisan je u razvojnom okruženju *Jupyter Notebook* i programskom jeziku Python 3. Za učitavanje podataka korištena je bilbioteka *pandas*, a za vizualizaciju grafova korištena je biblioteka *matplotlib*.

5.7.1 Opis postupka grupiranja

Dobivenih 210 zapisa mikciometrijskih krivulja predano je programu UroflowFeatureExtractor_1_0 da bismo dobili 33 izlučene značajke za kvantitativan opis krivulja. Zatim su dobiveni zapisi značajki podijeljeni na skupove za učenje i testiranje u omjeru 2:1. Zbog uporabe euklidske udaljenosti unutar promatranih postupaka grupiranja potrebno je skalirati dobivene značajke kako bismo izbjegli privrženost prema značajkama velikih vrijednosti. Za skaliranje korišten je postupak *MinMaxScaler* iz paketa *sklearn.preprocessing* koji radi skaliranje po svakoj značajki prema formuli (12).

$$\frac{x_i - \max(x)}{\max(x) - \min(x)} \tag{12}$$

Postupak *MinMaxScaler* je učen na primjerima za učenje, a zatim su skalirana oba skupa primjera za učenje i primjera za testiranje.

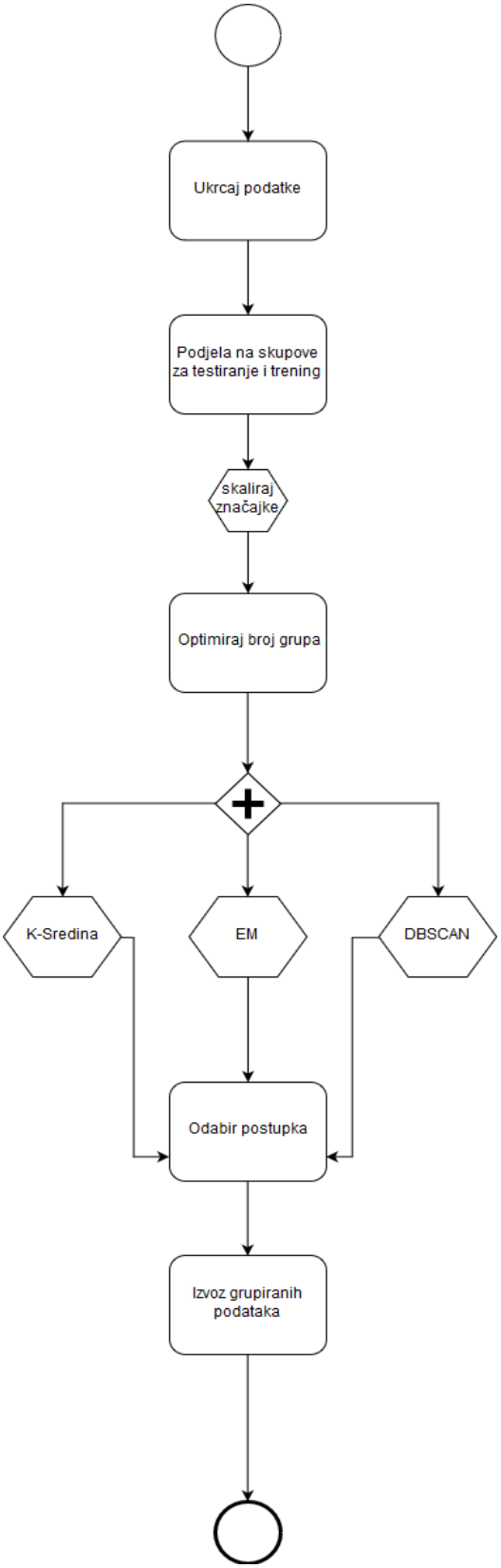
Nakon provođenja postupka skaliranja značajki provodimo postupak odabira optimalnog broja grupa za svaki promatrani postupak grupiranja. Postupcima K-srednjih vrijednosti i EM trebamo zadati broj grupa prije pokretanja postupka dok postupak DBSCAN sam zaključuje optimalan broj grupa za podatke. Promatrani raspon grupa je [2, 10]. Vizualnom subjektivnom inspekcijom primjera možemo ustvrditi nisku zastupljenost krivulja oblika tornja kao i isprekidanih mikciometrijskih krivulja. Za svaki broj grupa pokrećemo postupak 1000 puta zbog različitog odabira početnih centroida (postupak K-sredina) i različitog odabira početnih težina (postupak EM) pri svakom pokretanju postupka. Odabire se najbolji model od 1000 učenja kao predstavnik modela za zadani broj grupa. Nad najboljim modelom se računaju 3 mjere interne procjene grupiranja ranije opisane u radu.

Postupak DBSCAN ima dva hiperparametra koja možemo optimirati. Najmanji broj susjeda (*minPts*) kojeg primjer mora imati da bi se smatrao jezgrom optimiramo kao i parametar *eps* koji govori o potrebnoj udaljenosti između dva primjera da se smatraju susjedima. Promatrane vrijednosti parametra *eps* su: linearno raspoređenih 15 vrijednosti u rasponu [0.5, 1.5], a vrijednosti parametra *minPts* su u rasponu [3,5].

Predloženo je postaviti parametar *minPts* na broj 2 puta veći od broja značajki. Za podatke velikog broja značajki i podatke koje sadrže velik šum u značajkama predlaže se postavljanje parametra *minPts* i iznad predložene vrijednosti. Predložena vrijednost za ovaj skup podataka s 33 značajke je 66, no za tu vrijednost dobivamo rezultate od 0 grupa i svi primjeri su označeni kao šum. Parametar *eps* je predloženo odabrati čim manjim. *Eps* uveliko ovisi o korištenoj funkciji udaljenosti. Jedna od predloženih heuristika za odabir ovog parametra je udaljenost do (2 * broj značajki – 1) najbližeg susjeda. [10]

Nakon odabira broja grupa i jednog od postupaka potrebno je odrediti grupe primjerima iz skupova za učenje i testiranje. Također, potrebno je iskrcati skupove u *arff* format kako bismo ih mogli koristiti dalje za postupak klasifikacije u programskom alatu *Weka*.

Korake postupka grupiranja možemo vidjeti na slici 10.



Slika 10. Koraci postupka grupiranja

5.7.2 Rezultati grupiranja

Cilj grupiranja mikciometrijskih krivulja je odrediti optimalan broj grupa nad promatranim skupom podataka. Promatrani skup se sastoji od 210 primjera mikciometrijskih krivulja. Već ukazanom subjektivnom vizualnom inspekcijom primjećujemo nedovoljnu zastupljenost pojedinih tipova krivulja kao i prevladavanje normalnog tipa krivulja. Uz slabu zastupljenost nakon podjele skupa podataka na skupove za učenje i testiranje očekujemo da će se povećanjem očekivanog broja grupa drastično smanjivati broj primjera koji pripadaju pojedinim grupama. Odbacit ćemo sve grupe s kardinalnim brojem manjim od 10.

Za početak ćemo pogledati rezultate grupiranja postupkom DBSCAN. Korištena je implementacija postupka *DBSCAN* iz paketa *sklearn.cluster*. Ovaj postupak, za razliku od postupaka K-srednjih vrijednosti i EM pruža ugrađeno otkrivanje šuma u podatcima. Tako da ćemo promatrati broj grupa, broj primjera označenih kao šum, kao i tri mjere koje su prije opisane. Rezultate postupka možemo vidjeti u tablici 2.

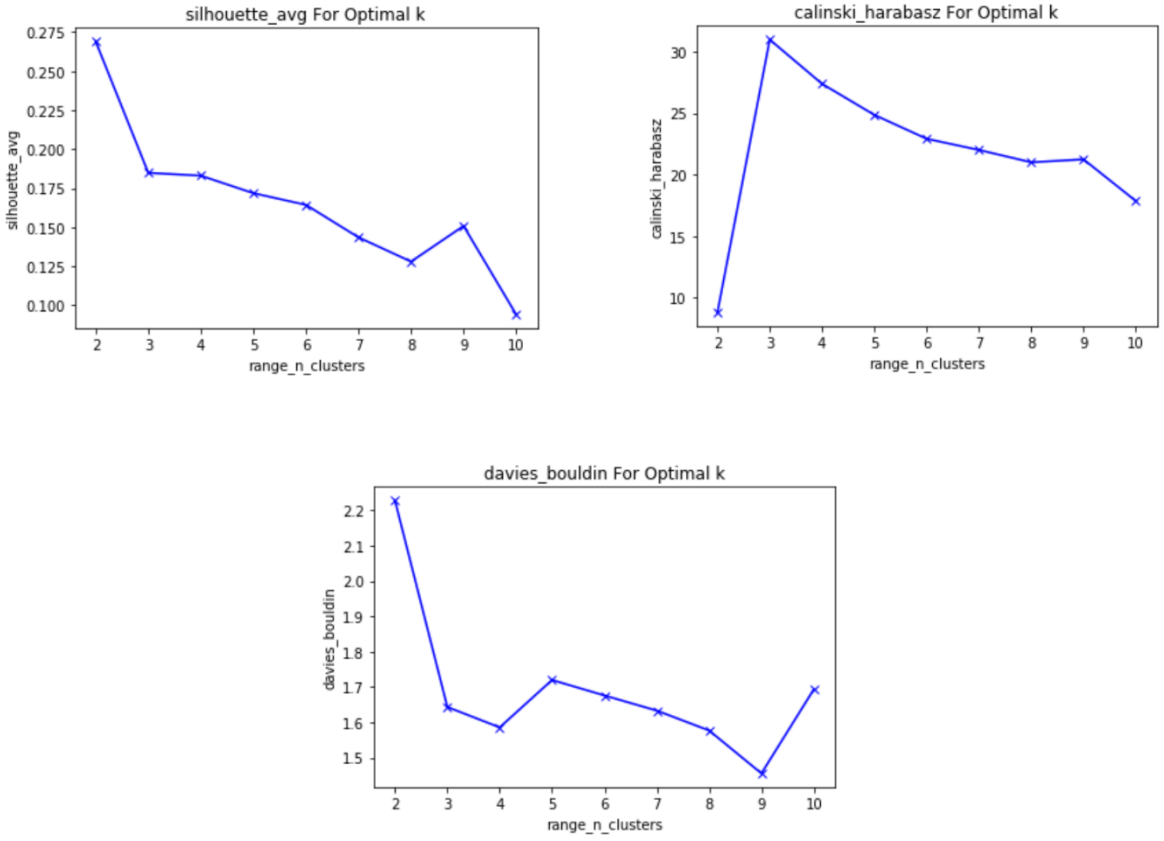
Tablica 2. Rezultati postupka DBSCAN

eps	*minPts*	Broj grupa	Broj primjera šuma	Koeficijent siluete	Indeks Calinski-Harabasz	Indeks Davies-Bouldin
0.5	3	4	61	-0.131	5.844	2.578
0.5714	3	2	40	0.167	8.967	3.101
0.6428	3	2	28	0.221	9.438	2.805
0.7142	3	2	19	0.289	10.039	2.520
0.7857	3	2	16	0.304	10.076	2.406
0.8571	3	2	14	0.319	10.552	2.229
0.9285	3	1	10	-	-	-

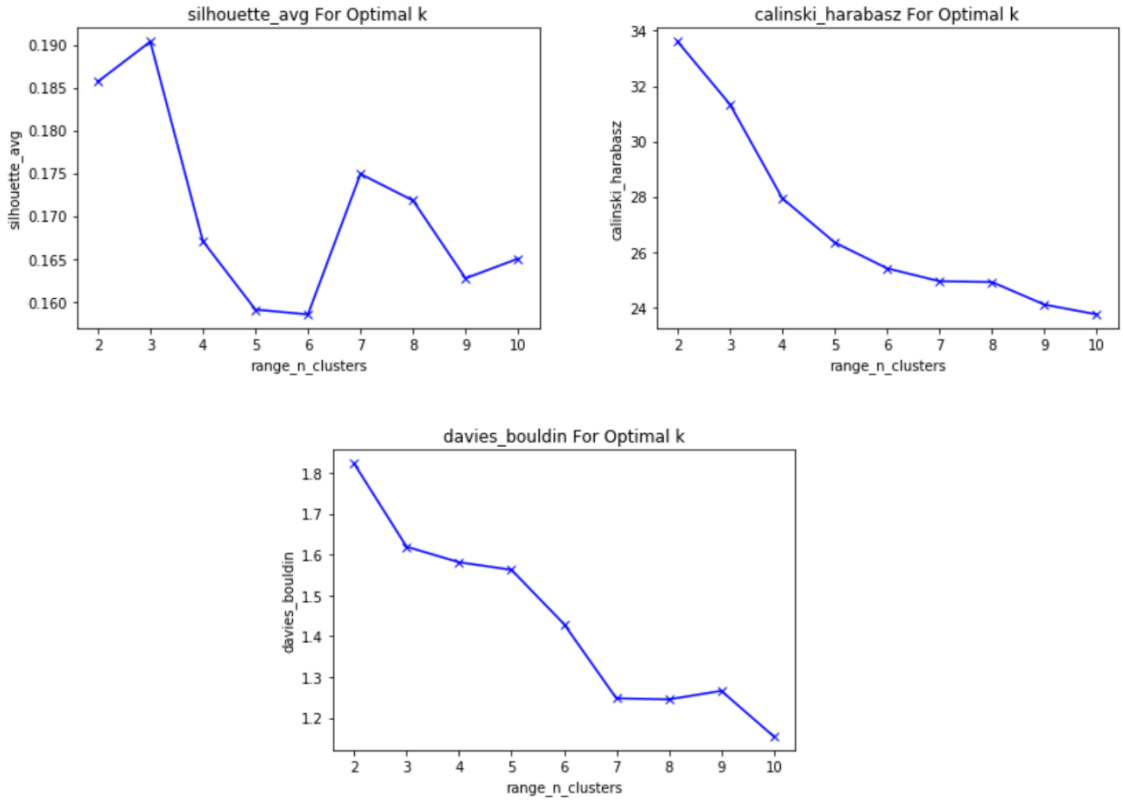
U tablici 2 možemo vidjeti rezultate samo za 7 kombinacija parametara postupka. Za ostalih 38 kombinacija, ovaj postupak grupira sve primjere u jednu grupu kao što vidimo na zadnjem primjeru parametara. Za prethodno predložene vrijednosti parametra *minPts* (oko 66) postupak ne daje niti jednu grupu već označava sve primjere kao šum. Iz tog razloga odbacujemo DBSCAN postupak kao kandidata za odabir optimalnog broja grupa mikciometrijskih krivulja.

Za postupak K-srednjih vrijednosti korištena je implementacija *KMeans* iz paketa *sklearn.cluster*. Za postupak EM korištena je implementacija *GaussianMixture* iz paketa *sklearn.mixture*.

Za odabir optimalnog broja grupa računamo tri mjere: koeficijent siluete (viša vrijednost povlači bolje grupiranje), indeks Davies-Bouldin (vrijednosti bliže nuli povlače bolje grupiranje) i indeks Calinski-Harabasz (viša vrijednost povlači bolje grupiranje). Promatramo promjene mjera u odnosu na povećavanje broja grupa. Na slici 11 možemo vidjeti kretanje mjera u promatranom rasponu grupa za postupak EM, a na slici 12 možemo vidjeti za postupak K-srednjih vrijednosti. Iz danih grafova možemo iščitati potencijalne optimalne brojeve grupa. Tražimo vršne vrijednosti kao što je vrijednost indeksa Calinski-Harabasz za 3 grupe postupka EM. Također, optimalan broj grupa može predstavljati početak promjene nagiba krivulje kao što su vrijednosti indeksa Davies-Bouldin za brojeve grupa 3 i 7 postupka K-srednjih vrijednsoti.



Slika 11. Rezultati postupka EM



Slika 12. Rezultati postupka K-srednjih vrijednosti

Za promatrane grafove predložit ćemo dvije potencijalne vrijednosti optimalnog broja grupa. Za postupak EM predložene vrijednosti možemo vidjeti u tablici 3, a za postupak K-srednjih vrijednosti u tablici 4. Kao optimalni broj grupa uzimamo najčešći prijedlog u tablici. Iz tablica možemo iščitati da je optimalan broj grupa za promatrani skup podataka 3.

Tablica 3. Vrijednosti optimalnih broja grupa po mjerama za postupak EM

Postupak	Prvi prijedlog broja grupa	Drugi prijedlog broja grupa
Koeficijent siluete	2	3
Indeks Calinski-Harabasz	3	-
Indeks Davies-Bouldin	3	4

Tablica 4. Vrijednosti optimalnih broja grupa po mjerama za postupak K-sredina

Postupak	Prvi prijedlog broja grupa	Drugi prijedlog broja grupa
Koeficijent siluete	3	7
Indeks Calinski-Harabasz	3	-
Indeks Davies-Bouldin	3	7

Nakon odabira optimalnog broja grupa nastavljamo s odabirom najboljeg postupka grupiranja. Za ovaj korak promatramo dva postupka, K-srednjih vrijednosti i EM. Ponovno učimo postupke s prethodno odabranim brojem grupa i računamo sve 3 mjere. Odabiremo postupak koji daje bolje rezultate u dvije ili više mjera. Rezultate mjera za usporedbu postupka možemo vidjeti u tablici 5. Iz tablice možemo iščitati da je postupak K-srednjih vrijednosti dao bolje rezultate za sve tri promatrane mjere i time ga odabiremo kao postupak grupiranja u daljnjim koracima.

Tablica 5. Usporedba postupaka grupiranja

Postupak	Koeficijent siluete	Indeks Calinski-Harabasz	Indeks Davies-Bouldin
EM	0.1655	14.8900	1.7811
K-sredina	0.2111	19.1475	1.4600

Nakon odabira postupka dobivamo grupiran skup podataka za Također, potrebno je iskoristiti odabrani postupak za grupiranje skupa podataka za testiranje. Zatim je potrebno izvesti podatke u *arff* formatu kako bismo ih mogli ukrcati u programski alat *Weka*.

U tablici 6 možemo vidjeti zastupljenost pojedinih grupa u podatcima. Promatramo skupove podataka za treniranje i testiranje kao i njihovu uniju. Sve grupe po veličini prelaze minimalni broj primjera da bi ih smatrali dobrim kandidatom za grupu.

Tablica 6. Zastupljenost grupa

Grupa	Skup za treniranje	Skup za testiranje	Unija skupova
1	45.8% (64)	41.4% (29)	44.2% (93)
2	45% (63)	44.3% (31)	44.7% (94)
3	9.2% (13)	14.3% (10)	11.1% (23)

5.7.3 Opis postupka klasifikacije

Provedbom postupka grupiranja dobivamo dva skupa: skup za treniranje od 140 primjera i skup za testiranje od 70 primjera u *arff* formatu. Promatrani skupovi podataka su skalirani postupkom *MinMaxScaler* i grupirani postupkom K-srednjih vrijednosti u tri grupe.

Za daljnju analizu korišten je alat *Weka.* Ovaj alat, po značenju *Waikato Environment for Knowledge Analysis* (Weka), je program napisan u programskom jeziku Java namijenjen obradi podataka pomoću strojnog učenja. Ovaj alat je odabran jer sadrži implementacije postupaka *RIPPER* i *C4.5.* Implementacija postupka *RIPPER* se u alatu *Weka* naziva *JRip*, a implementacija postupka *C4.5* se naziva *J48*. Korištena je inačica alata 3.8.

Optimizaciju parametara postupaka provodimo na skupu za učenje pomoću postupka *k*-struke unakrsne provjere. Odabran broj preklopa *k* je 5. *K*-struka unakrsna provjera dijeli podatke na *k* preklopa s ciljem da se promatrani postupak uči na 4 preklopa dok se odabrana statistika vrednovanja modela ispituje na preostalom preklopu. Postupak učenja i testiranja nad preklopima se obavlja *k* puta tako da se nad svakim od preklopa jednom provede testiranje modela. Kao najbolji promatrani model odabire se onaj s najboljom statistikom vrednovanja.

Parametri postupka *JRip* koje optimiramo su *minNo* i *optimizations.* Parametar *minNo* predstavlja najmanju težinu primjera koju novostvoreno pravilo mora pokriti kako bi se prihvatilo kao novo pravilo. Kako svi promatrani primjeri imaju težinu 1, parametar *minNo* označava najmanji broj primjera koje pravilo mora pokriti. Promatrane vrijednosti parametara *minNo* su u intervalu [2,6]. Parametar *optimizations* predstavlja broj optimizacija odrađenih u svakom koraku postupka. Promatrane vrijednosti parametra *optimizations* su u intervalu [2,4].

Parametri postupka *J48* koje optimiramo su *confidenceFactor* i *minNumObj*. Parametar *confidenceFactor* predstavlja razinu povjerenja u rezultate nad podatcima za učenje. Parametar dolazi do izražaja u fazi podrezivanja. Niže vrijednosti parametra povlače veću vjerojatnost da će se dogoditi podrezivanje tj. postupak podrezivanja

pesimistično gleda pogrešku nad skupom za učenje. Čvorovi do kojih je došlo u vrlo malom broju slučajeva iz skupa podataka za učenje se kažnjavaju, jer ne možemo pouzdano pretpostaviti njihovu stvarnu klasifikacijsku pogrešku. Promatrane vrijednosti parametra su [0.1, 0.3, 0.5].

Parametar *minNumObj* predstavlja najmanji broj primjera u listu stabla. Veće vrijednosti ovog parametra pospješuju generalizaciju i smanjuju prenaučenost. Također, veće vrijednosti parametra daju jednostavnija stabla. Pretpostavljena vrijednost parametra je 2. Promatrane vrijednosti parametra su u rasponu [2,5].

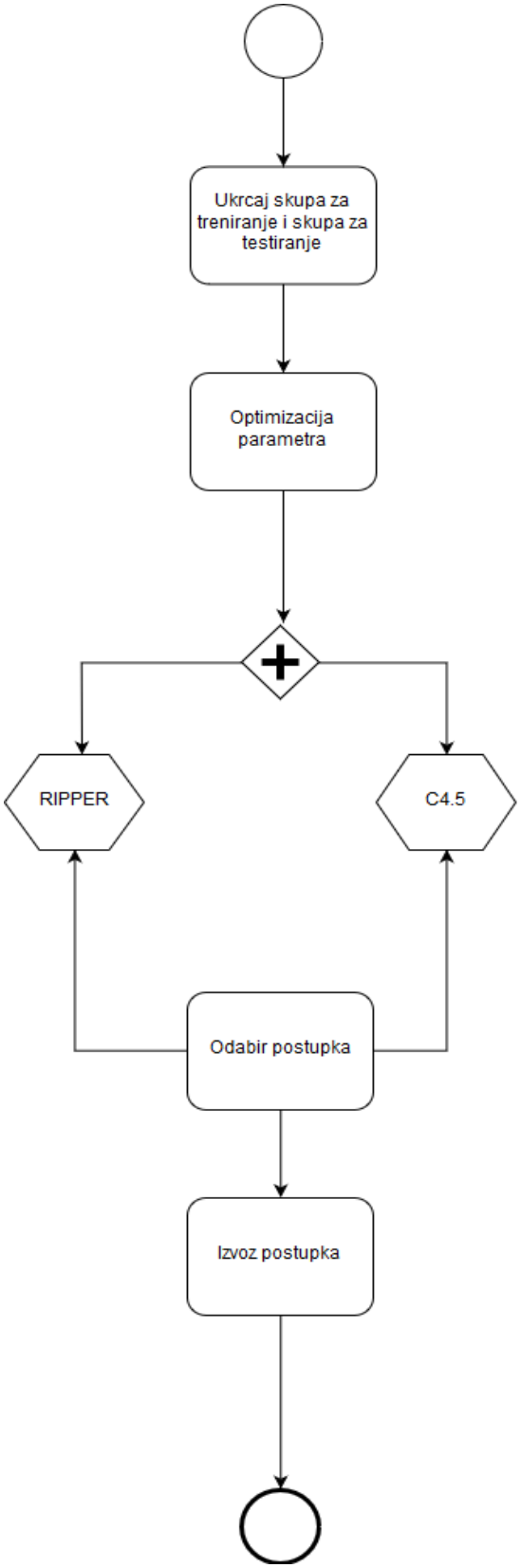
Odabrana statistika vrednovanja modela je F-mjera. F-mjera predstavlja harmonijsku sredinu preciznosti i odziva, a računa se po formuli (13). F-mjera se računa za svaku od grupa zasebno, a zatim se promatra srednja vrijednost F-mjere za sve tri grupe.

$$F = \frac{2PR}{P+R} \tag{13}$$

Nakon pronalaska optimalnih parametara postupaka sam odabir postupka obavljamo računanjem F-mjere nad skupom za testiranje. Za ugradnju u web aplikaciju odabiremo postupak s najvećom F-mjerom, a također u interesu nam je imati čim jednostavniju interpretaciju rezultata, što znači manje pravila i jednostavnija stabla.

Izvoz postupka obavljamo putem alata *Weka* u formatu *model* koji se kasnije može koristiti jednostavnim uvozom u web aplikaciju pomoću ugrađenih funkcija alata *Weka* pisanih za korištenje u programskom jeziku Java.

Korake postupka klasifikacije možemo vidjeti na slici 13.



Slika 13. Koraci postupka klasifikacije

5.7.4 Rezultati klasifikacije

Cilj klasifikacije mikciometrijskih krivulja je pronaći čim bolji klasifikacijski postupak s jasnim tumačenjem kako bismo mogli izgraditi web aplikaciju za uspješno predviđanje grupa novih mikciometrijskih krivulja.

Rezultate optimizacije postupka *JRip* možemo vidjeti u tablici 7. Za optimalne parametre postupka *optimizations* odabiremo 2, a za *minNo* 5. Iako su vrijednosti F mjere jednake za vrijednosti parametra *minNo* 4 i 5 odabiremo pesimističniju, veću vrijednost parametra.

Tablica 7. Rezultati optimizacije JRip postupka

optimizations	*minNo*	F mjera	Broj pravila
2	2	0.850	4
2	3	0.864	4
2	4	0.871	4
2	5	0.871	4
2	6	0.864	4
3	2	0.851	4
3	3	0.851	4
3	4	0.865	4
3	5	0.858	4
3	6	0.864	4
4	2	0.836	5
4	3	0.851	5
4	4	0.843	4
4	5	0.851	4
4	6	0.858	4

Rezultate optimizacije postupka *J48* možemo vidjeti u tablici 8. Za optimalne vrijednosti parametra *confidenceFactor* odabiremo 0.1, a za *minNumObj* 3. Zanimljivo je primijetiti da parametar *confidenceFactor* nema utjecaja na rezultat, no odabire se pesimistična vrijednost parametra.

Nakon odabira optimalnih parametara odabir postupka radimo ponovnim učenjem nad cijelim skupom za učenje te uspoređivanjem kvalitete modela nad skupom za testiranje.

Tablica 8. Rezultati optimizacije postupka J48

confidenceFactor	*minNumObj*	F-mjera	Veličina stabla
0.1	2	0.835	13
0.3	2	0.835	13
0.5	2	0.835	13
0.1	3	0.842	13
0.3	3	0.842	13
0.5	3	0.842	13
0.1	4	0.828	11
0.3	4	0.828	11
0.5	4	0.828	11
0.1	5	0.828	11
0.3	5	0.828	11
0.5	5	0.828	11

Postupak *JRip* kao rezultat izgradi skup pravila odluke pomoću kojih radi klasifikaciju novi primjera. Naučena pravila nad skupom za učenje možemo vidjeti u nastavku:

$(INCREASE\_DECREASE\_30\_PREC\_ACCENT >= 0.343591) => CLASS = 2.0\ (15.0/2.0)$

$(FLOW\_DURATION\_IN\_s <= 0.106838) => CLASS = 1.0\ (54.0/3.0)$

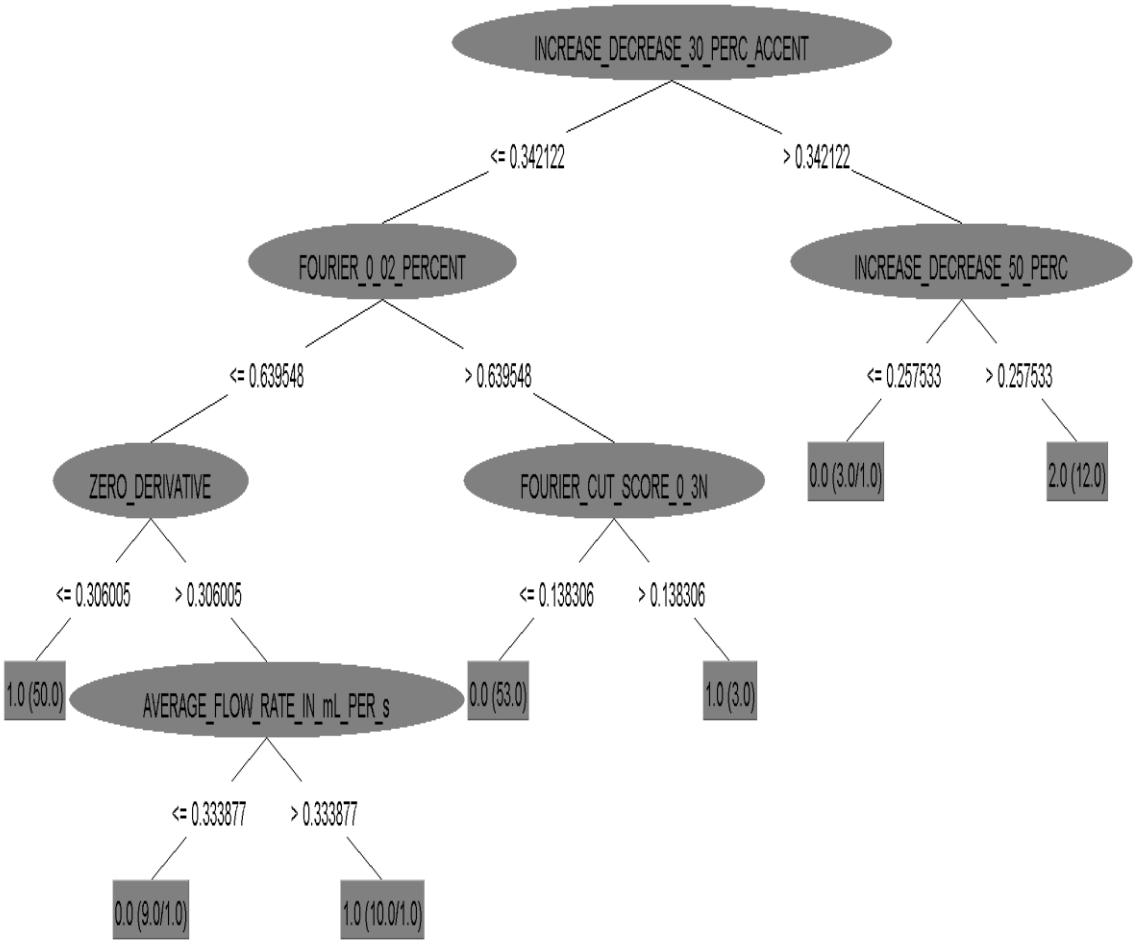
$(FOURIER\_CUT\_SCORE\_0\_3 >= 0.256124) => CLASS = 1.0\ (12.0/1.0)$

$=> CLASS = 0.0\ (59.0/1.0)$

Možemo primijetiti da u klasifikaciji sudjeluju samo 4 pravila, što doprinosi vrlo jasnom tumačenju odluke. Također, i sama pravila su jednostavna i sastoje se od samo jedne značajke.

Postupak *J48* kao rezultat izgrađuje stablo odluke pomoću kojeg radi klasifikaciju novih primjera. Naučeno stablo odluke nad skupom za učenje možemo vidjeti na slici 14.

Stablo odluke se sastoji od 13 čvorova, od kojih su 7 listovi. Za razliku od postupka *JRip*, u stablu odluke sudjeluje 6 značajki što ga čini kompleksnijim od skupa pravila.



Slika 14. Stablo odluke J48 postupka

Rezultate F-mjera i ostalih standarnih mjera za usporedbu postupaka nad skupom za testiranje možemo vidjeti u tablici 9. Ne vidimo značajnu razliku mjere u postupcima. Za odabir postupka koji ćemo koristiti u web aplikaciji poslužit ćemo se brojem značajki korištenih u modelu. Tim argumentom odabiremo postupak *JRip*, čiji model izvozimo u *model* formatu za danju uporabu.

Tablica 9. Usporedba postupaka nad skupom za testiranje

Postupak	F-mjera	ACC (Accuracy)	SENS (Sensitivity)	SPEC (Specificity)
JRip	0.913	0.914	0.914	0.953
J48	0.914	0.914	0.914	0.952

6. Web aplikacija

U okviru ovog rada, izrađena je prikladna web aplikacija u svrhu klasificiranja novih primjera mikciometrijskih krivulja. Aplikacija se sastoji od poslužiteljske i klijentske komponente. Poslužiteljska komponenta pisana je u programskom jeziku Java unutar razvojnog okvira *Spring boot*. Također, potrebna je podrška jezika Python jezika i paketa iz biblioteke *sklearn*. Funkcionalnost poslužiteljske komponente se sastoji od pružanja sučelja za obradu mikciometrijskih krivulja. Sučelje se koristi putem arhitektonskog stila *REST*. Omogućeno je učitavanje datoteke sa zapisima mikciometrijskih krivulja u *csv* formatu. Mikciometrijski zapisi dolaze u jednom redu, s tim da prva linija ne sadrži zapis. Svaki idući red sadrži zapis oblika „(id), N izmjerenih vrijednosti.“

Koraci potrebni za klasifikaciju mikciometrijskih krivulja prikazani su na slici 15. Izlučivanje značajki se provodi putem programa *UroflowFeatureExtractor_1_0* . Skaliranje značajki se provodi istim postupkom primijenjenim u koraku grupiranja podatka, učenim nad skupom za učenje. Kao klasifikator odabran je postupak *JRip*. Rješenje se izvozi s informacijama o grupi mikciometrijske krivulje, distribuciji pouzdanosti klasifikacije po grupama te nizom izlučenih značajki.

Klijentska komponenta aplikacije pisana je u programskom jeziku *JavaScript* unutar razvojnog okvira *Angular*. Funkcija klijentske komponente je pružiti sučelje za učitavanje i konzumaciju klasifikacije krivulja poslužiteljske komponente. Također, sučelje sadrži opis provedenog postupka grupiranja i klasifikacije mikciometrijskih krivulja, prikaz stabla odluke postupka *J48* i skupa pravila postupka *JRip*, te uputa za formatiranje i ukrcavanje podataka. Provedbom klasifikacije rezultati su prikazani unutar tablice, gdje prva kolona prikazuje vrijednost grupe, iduće tri pouzdanost klasifikacije po grupama, a zatim niz naziva izlučenih značajki i njihovih vrijednosti.

Na slici 16 moguće je vidjeti početni ekran aplikacije. Nakon učitavanja i provedbe postupka klasifikacije krivulja prikaz rezultata korisniku se može vidjeti na slici 17.

Slika 15. Koraci klasifikacije mikciometrijskih krivulja

Klasifikacija krivulja mikciometrije

Značajke

Provedena je analiza krivilja mikciometrije, njih 210. Iz krivulja izračunate su 33 značajke. Nad dobivenim značajkama proveden je postupak grupiranja pomoću 3 postupka: K-Means, EM i DBSCAN, korištene su sklearn implementacije. Zaključeno je da je optimalan broj grupa 3. Najbolji rezultati dobiveni su K-Means postupkom prema mjerama: Silhouette Coefficient, Calinski-Harabasz Index i Davies-Bouldin index.

Klasifikacija

Klasifikacija krivulja trenirana je na implementacijama postupaka iz programskog alata weka a korišteni su postupci J48 i RIPPER.

RIPPER:
(INCREASE_DECREASE_30_PERC_ACCENT >= 0.343591) => CLASS=2.0 (15.0/2.0)
(FLOW_DURATION_IN_s <= 0.106838) => CLASS=1.0 (54.0/3.0)
(FOURIER_CUT_SCORE_0_3 >= 0.256124) => CLASS=1.0 (12.0/1.0)
=> CLASS=0.0 (59.0/1.0)

Stablo odluke:
INCREASE_DECREASE_30_PERC_ACCENT <= 0.342122
|___FOURIER_0_02_PERCENT <= 0.639548
|___|___ZERO_DERIVATIVE <= 0.306005: 1.0 (50.0)
|___|___ZERO_DERIVATIVE > 0.306005
|___|___|___AVERAGE_FLOW_RATE_IN_mL_PER_s <= 0.333877: 0.0 (9.0/1.0)
|___|___|___AVERAGE_FLOW_RATE_IN_mL_PER_s > 0.333877: 1.0 (10.0/1.0)
|___FOURIER_0_02_PERCENT > 0.639548
|___|___FOURIER_CUT_SCORE_0_3N <= 0.091189: 0.0 (51.0)
|___|___FOURIER_CUT_SCORE_0_3N > 0.091189: 1.0 (5.0/2.0)
INCREASE_DECREASE_30_PERC_ACCENT > 0.342122: 2.0 (15.0/2.0)

Učitavanje skupa podataka

Format podataka mora biti .csv datoteka gdje prva linija ne sadrži mikciometrijski zapis. Svaki idući red sadrži mikciometrijski zapis oblika (id), N izmjerenih vrijednosti. Svaki zapis mora imati isti broj izmjerenih vrijednosti (N), ako neki zapisi kraće traju ne upisuje se vrijednost. Primjer jednog zapisa:
502301,0,8,20,37,55,71,82,87,86,84,81,81,83,88,95,103,110,
118,126,131,134,133,131,129,127,127,129,132,133,133,0,0,,,

Choose a file... *

Slika 16. Početni ekran aplikacije

CLASS	A %	B %	C %	PEAK_FLOW_RATE_mL_PER_s	FLOW_VOLUME_mL	TIM
B	0.06	0.94	0.00	0.323383	0.132391	0.05
B	0.06	0.94	0.00	0.557214	0.296468	0.06
C	0.13	0.00	0.87	0.402985	0.065022	0.04
C	0.13	0.00	0.87	0.31592	0.047618	0
B	0.08	0.92	0.00	0.71393	0.628158	0.12
B	0.08	0.92	0.00	0.992537	0.652221	0.42
B	0.06	0.94	0.00	0.462687	0.199125	0.03

Slika 17. Ekran rezultata mikciometrijske analize

7. Zaključak

U ovom radu predstavljena je analiza grupiranja i klasifikacije mikciometrijskih krivulja. Ispostavilo se da grupiranje krivulja predstavlja složen problem za koje je potrebno ekspertno znanje pri oblikovanju značajki. Predstavio se problem otkrivanja šuma u podatcima koji je potrebno detaljnije analizirati, jer se postupak *DBSCAN* nije pokazao učinkovitim za otklanjanje šuma.

Klasifikacijske metode s jasnim tumačenjem su se pokazale dobrim odabirom zbog jednostavnosti interpretacije postupka dolaska do grupiranja, posebice na primjeru postupka *JRip* koji je izgradio modela sa samo četiri pravila i tri značajke.

U daljnjim istraživanjima, potrebno je prikupiti značajno veći skup mikciometrijskih krivulja, implementirati veći skup ekspertnih značajki za izlučivanje iz krivulja, a posebice nelinearnih značajki i analizirati kvalitetu značajki pri klasifikaciji krivulja.

Josip Renić

_____

Literatura

[1] J. Kortenbout i sur. „The effect of home uroflowmetry on the uroflow curve of children with dysfunctional voiding“, kolovoz 2016.

[2] „The Urinary Tract & How It Works”, https://www.niddk.nih.gov/health-information/urologic-diseases/urinary-tract-how-it-works, lipanj 2019.

[3] N. Sancheti. „Presentation on Understanding Uroflowmetry”, Urology & Nephrology Open Access Journal, Volume 3 Issue 6 - 2016.

[4] M. Trošelj, N. Rubinić, I. Vukelić i D. Markić, „Urodinamika i njezina klinička primjena“, medicina fluminensis 2017, Vol. 53, No. 3, p. 351-358

[5] A. N. Idsardi, .P. Dik, A. Nieuwenhof- Leppink, B. ten Haken i M. Groenier. „Effect of biofeedback training on the uroflow curve of children with dysfunctional voiding“ studeni 2016.

[6] S. Abdovic i sur. „Predicting posterior urethral obstruction in boys with lower urinary tract symptoms using deep artificial neural network“, studeni 2018.

[7] J. Šnajder. „Strojno učenje:19. Grupiranje, Natuknice s predavanja od 19. 1. 2019., v1.1“, https://www.fer.unizg.hr/_download/repository/SU-2018-19-Grupiranje.pdf, lipanj 2019.

[8] S. Borman. „The Expectation Maximization AlgorithmA short tutorial“, srpanj 2004.

[9] J. Šnajder. „Strojno učenje:20. Grupiranje II, Natuknice s predavanja od 11. 1. 2019., v1.1“, https://www.fer.unizg.hr/_download/repository/SU-2018-20-Grupiranje2.pdf, lipanj 2019.

[10] E. Schubert i sur. „DBSCAN Revisited, Revisited:Why and How You Should (Still) Use DBSCAN“, srpanj 2017.

[11] R. Feldman i J. Sanger. „The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Dana“, Cambridge Univ. Press., 2007.

[12] P. J. Rousseeuw. „"Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis”, Computational and Applied Mathematics, 1987.

[13] „2.3.10.6. Calinski-Harabasz Index“, https://scikit-learn.org/stable/modules/clustering.html#calinski-harabasz-index, lipanj 2019.

[14] „2.3.10.7. Davies-Bouldin Index“, https://scikit-learn.org/stable/modules/clustering.html#davies-bouldin-index, lipanj 2019.

[15] X. Wu i V. Kumar. „The Top Ten Algorithms in Data Mining“, travanj 2009.

[16] I. H. Witten, E. Frank i M. A. Hall. „Data Mining: Practical Machine Learning Tools and Techniques“, 2nd ed., Morgan Kaufmann, 2011.

[17] „2.1. Gaussian mixture models“, https://scikit-learn.org/stable/modules/mixture.html, lipanj 2019.

Popis slika

Popis tablica

Modeli grupiranja i klasifikacije krivulja mikciometrije

Sažetak: U ovom radu promatrani su anonimizirani biološki vremenski nizovi krivulja mikciometrije snimanih kod djece. Provedena je dubinska analiza optimalnog broja grupa za dani skup podataka, analiza postupaka grupiranja i analiza klasifikacijskih postupaka s jasnim tumačenjem. Prikazan je opis razvijenog programa za izlučivanje značajki iz vremenskog niza mikciometrijske krivulje, također opisan je postupak izračuna značajki. Dobiven je optimalan broj od tri grupe, temeljen na internim mjerama kvalitete grupiranja. Naučeni su postupci klasifikacije s jasnim tumačenjem temeljeni na pravilima i stablom odluke. Cijeli postupak grupiranja i klasifikacije objedinjen je u web aplikaciji koja omogućuje učitavanje anonimiziranih zapisa vremenskih nizova mikciometrije te dobivanje odgovora o kategoriji obrasca krivulje, uz prikazani određeni postotak pouzdanosti kategorizacije.

Ključne riječi: Mikciometrija, grupiranje, klasifikacija, interne mjere kvalitete grupiranja.

Clustering and classification models of uroflow curves

Abstract: In this paper anonymised biological time series of uroflow curves recorded in children were observed. A depth analysis of the optimum number of groups for a given data set, analysis of grouping procedures, and analysis of classification procedures with clear interpretation was performed. A description of the developed program for extracting features from the time series of the uroflow curve is also described, a feature calculation process is also described. An optimal number of three groups was obtained, based on internal grouping quality measures. The classification procedures with clear interpretation based on the rules and the decision tree have been learned. The entire grouping and classification process are combined into a web application that allows user to load anonymous uroflow curves and to obtain a response to the uroflow category, along with a certain percentage of categorization reliability.

Key words: Uroflow, clustering, classification, internal clustering evaluation