# Klasifikacija pokretnih korisničkih profila telekomunikacijske mreže metodologijom strojnog učenja

**Kušić, Kristina**



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

# DIPLOMSKI RAD

**Kristina Kušić**

Zagreb, 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

# DIPLOMSKI RAD

Mentor:

Doc. dr. sc. Tomislav Stipančić, dipl. ing.

Student:

Kristina Kušić

Zagreb, 2020.

*Izjavljujem da sam ovaj rad izradila samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.*

*Zahvaljujem se svima koji su mi na bilo koji način pomogli prilikom izrade ovog rada.*

*Posebno se zahvaljujem svom mentoru doc. dr. sc. Tomislavu Stipančiću na ukazanom povjerenju, svoj pruženoj pomoći i savjetima tijekom izrade ovog rada te velikom razumijevanju. Također, zahvaljujem se kolegi dr. sc. Kristianu Skračiću na mnogim korisnim savjetima i pomoći pri izradi rada.*

*I na kraju, najveću zaslugu za ono što sam postigla pripisujem svojoj majci Jeli, koja je uvijek bila uz mene, bez obzira da li se radilo o teškim ili sretnim trenucima i bez čije podrške bi sve ovo bilo puno teže.*

*Veliko HVALA svima!*

Kristina Kušić

SVEUČILIŠTE U ZAGREBU
**FAKULTET STROJARSTVA I BRODOGRADNJE**
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske radove studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment,
inženjerstvo materijala te mehatronika i robotika

| Sveučilište u Zagrebu<br>Fakultet strojarstva i brodogradnje | |
|---|---|
| Datum: | Prilog: |
| Klasa: | |
| Ur. broj: | |

# DIPLOMSKI ZADATAK

Student: **KRISTINA KUŠIĆ** Mat. br.: 0035185396

Naslov rada na hrvatskom jeziku: **Klasifikacija pokretnih korisničkih profila telekomunikacijske mreže metodologijom strojnog učenja**

Naslov rada na engleskom jeziku: **Mobile user profile classification of a telecommunication network based on the machine learning methodology**

Opis zadatka:

Telekomunikacijski prijemnici i predajnici za mobilnu telefoniju sastoje se od većeg broja usmjerenih ćelija koje pokrivaju određena geoprostorna područja. Ovisno o fizičkom kretanju korisnika u mreži, svaka ćelija može imati drugačije vrijednosti parametara koji su za nju idealni, odnosno koji mogu rezultirati kvalitetnijim karakteristikama rada. Pretpostavka je da postoji konačan broj pokretnih (engl. mobility) korisničkih profila koji se koriste uslugama mreže te koji se ne mogu razlučiti jednostavnom geoprostornom analizom.
U radu je potrebno odrediti, prilagoditi i koristiti barem dva algoritama strojnog učenja te njima analizirati dostupnu bazu podataka s ciljem automatizirane identifikacije pokretnih profila pojedinih ćelija u mreži. Potrebnc je uzeti u obzir učestalost pojavnosti profila s obzirom na doba dana i dostupnost usluga mreže. S obzirom na to da se profili mogu razlikovati ovisno o vremenskoj komponenti brzine, na taj je način potrebno temeljiti njihovu klasifikaciju (npr. spori, srednje brzi, brzi, jako brzi profili).
Dobivene rezultate je potrebno usporediti s rezultatima provođenja vanjske i neovisne evaluacije rada mreže.
U radu je potrebno navesti korištenu literaturu te eventualno dobivenu pomoć.

| Zadatak zadan: | Rok predaje rada: | Predviđeni datum obrane: |
|---|---|---|
| 14. studenog 2019. | 16. siječnja 2020. | 20. siječnja do 24. siječnja 2020. |

Zadatak zadao: Predsjednica Povjerenstva:

doc. dr. sc. Tomislav Stipančić prof. dr. sc. Biserka Runje

# SADRŽAJ

## POPIS SLIKA

# POPIS TABLICA

# POPIS OZNAKA

| Oznaka | Jedinica | Opis |
|---|---|---|
| $\alpha$ | [°] | Mjera kuta |
| d | [m] | Duljina |
| $v_1$ | [bps] | Brzina prijenosa podataka u računalstvu |
| f | Hz | Frekvencija |
| $v_1$ | [km/h] | Brzina |
| t | [s] | Vrijeme |

## SAŽETAK

Tema ovog diplomskog rada je „Klasifikacija pokretnih korisničkih profila telekomunikacijske mreže metodologijom strojnog učenja“. U ovom radu opisani su koncepti rada mobilne telekomunikacijske mreže. Dvjema metodama strojnog učenja analizirani su određeni parametri pokretnih korisničkih profila između dva europska grada. Korištena je dostupna baza podataka s ciljem identifikacije i grupacije pojedinih ćelija unutar zadanog područja. Cilj je pronaći ćelije visokog stupnja mobilnosti, odnosno ćelije čije je područje pokrivanja usmjereno na autoput i prugu. Korištene metode su K-means i Gaussian Mixture metoda. Provedena je usporedba rezultata dobivenih objema metodama. Također, dobiveni rezultati uspoređeni su s podacima o radu mreže iz drugog vremenskog razdoblja kako bi se dokazala ponovljivost i sezonalnost rezultata.

Ključne riječi: ćelija, bazna stanica,, korisnički profil, grupiranje, klaster.

## SUMMARY

The topic of this master thesis is “Mobile user profiles classification of a telecommunication network based on the machine learning”. This project describes the concept of how mobile network works. This research uses two unsupervised machine learning methods in order to analyze cell *mobility* patterns. For this purpose, cell trace data from two European cities are used. The goal is to find “*high-mobility*” cells, or cells whose area of coverage is focused on the highway and track. The methods used were K-means and Gaussian Mixture method. A comparison of the results obtained with both methods is performed. Furthermore, the results were compared with the network performance data from other time interval to prove the repeatability and seasonality of the results.

Key words: cell, base station, user profile, clustering, cluster.

# 1. UVOD

Kako tehnologija napreduje, a svjetska populacija raste, tako raste i količina podataka koje svakodnevno stvaramo. Njihovo prikupljanje, sortiranje i analiziranje može biti težak zadatak. Istovremeno, razvoj mobilnih tehnologija i korištenje istih svakim danom je sve zastupljenije, kako u privatnim, tako i u poslovnim sferama.

Mobilne mreže danas su dio svakodnevnog života, vrlo su važne i značajne u svakodnevnom funkcioniranju stanovništva. U ovoj vrsti djelatnosti svoju potrebu i korist u različitom obujmu nalaze svi spektri društva, od pojedinaca do malih i velikih kompanija. Danas se gotovo svi služe nekim vidom mobilne komunikacije. Upravo radi potreba društva u čitavom svijetu (privatno, poslovno i globalno) mobilne telekomunikacije bilježe značajan razvoj unazad nekoliko desetljeća.

Mobilna mreža u svom značenju obuhvaća sučelje (eng. *hardver*) i programsku infrastrukturu (eng. *softver*). Do danas su razvijene i implementirane četiri ključne mobilne tehnologije – 1G, 2G, 3G i 4G. 5G mreža trenutno je u postupku razvoja i implementacije. Bazne stanice dio su hardverske infrastrukture mobilne mreže koje se sastoje se od većeg broja ćelija usmjerenih određenom geoprostornom području. Svaka se ćelija može podesiti na određene vrijednosti parametara u ovisnosti koji su za područje pokrivenosti idealni (npr. ruralno, planinsko, urbano područje, ili područje brzih cesta, vlakova, itd.). Time se postiže bolji i kvalitetniji rad pojedine ćelije što automatski rezultira boljim performansama cijele mreže. Pretpostavka je da postoji konačan broj pokretnih (eng. *mobility*) korisničkih profila ćelija te da se isti ne mogu razlučiti jednostavnom geoprostornom analizom.

Cilj ovog rada jest identificirati „*mobility*“ profile pojedinih ćelija u mreži između dva europska grada, kako bi se definirale „*high-mobility*“ ćelije. Ovaj cilj će se ostvariti analizom vremena zadržavanja korisnika u području pokrivanja pojedine ćelije. U radu su korištene dvije metode strojnog učenja pomoću kojih su se analizirala prikupljena mjerenja. S obzirom na to da se profili mogu razlikovati ovisno o vremenskom periodu provedenom u području neke ćelije, potrebno je same profile podijeliti u odgovarajuće vremenske periode i na temelju toga provesti analizu.

Dobiveni rezultati uspoređeni su s neovisnim mjerenjima iz drugog vremenskog perioda za iste ćelije i isto geoprostorno područje.

# 2. MOBILNA MREŽA

Mobilna mreža je komunikacijska mreža u kojoj je posljednja veza bežična. Mreža se rasprostire kroz površine koje pokrivaju bazne stanice, od kojih svaka koristi najmanje jedan (najčešće 3) primopredajnik signala s mobilnim uređajem i drugom baznom stanicom. Bazna stanica, cjelokupna infrastruktura unutar koje se nalaze i sve ćelije, na jednoj lokaciji pruža ćeliji mrežnu pokrivenost koja se može koristiti za prijenos glasa, podataka i drugih vrsta sadržaja. Stanica obično koristi drugačiji skup frekvencija od susjednih stanica, kako bi izbjegla smetnje i osigurala zajamčenu kvalitetu usluge unutar svake ćelije. Kada se spoje zajedno, te stanice pružaju radijsku pokrivenost na širokom zemljopisnom području što omogućuje velikom broju prijenosnih primopredajnika (npr. mobilni telefon, tablet i prijenosno računalo opremljeno mobilnim širokopojasnim modemima, *pager* itd.) međusobnu komunikaciju, s fiksnim primopredajnicima i telefonima bilo gdje u mreži, putem baznih stanica, čak i ako se neki od primopredajnika tijekom prijenosa kreću kroz više ćelija. [6]

Bazne stanice omogućuju niz poželjnih značajki unutar mobilne mreže. Jedna od njih je veći kapacitet od jednog velikog odašiljača, budući da se ista frekvencija može koristiti za višestruke veze sve dok su u različitim ćelijama. Mobilni uređaji troše manje energije u komunikaciji s baznom stanicom nego u komunikaciji s jednim odašiljačem ili satelitom, budući da su bazne stanice (konstrukcije) na maloj udaljenosti od korisnika, u bilo kojoj njegovoj poziciji tijekom kretanja. Veća površinska pokrivenost jednog zemaljskog odašiljača postiže se mogućnošću neograničenog dodavanja baznih stanica, prema potrebi, mogućnostima i odluci operatera – pružatelja telekomunikacijskih usluga.

Glavni pružatelji telekomunikacijskih usluga razmjestili su govorne i podatkovne mobilne mreže preko većine naseljenog kopnenog područja Zemlje. To omogućuje povezivanje mobilnih telefona i mobilnih računalnih uređaja s javnom komutiranom telefonskom mrežom i javnim internetom. Privatne mobilne mreže mogu se koristiti za istraživanje ili za velike organizacije, kao što je otprema za lokalne agencije za javnu sigurnost ili pojedine kompanije.

Povećani kapacitet u mobilnoj mreži, u usporedbi s mrežom s jednim odašiljačem, omogućio je višestrukim pozivateljima u istom području da koriste istu frekvenciju prebacivanjem poziva s iste frekvencije na najbližu dostupnu bazni stanicu koja ima tu istu

dostupnu frekvenciju. Ista se radiofrekvencija može ponovno upotrijebiti u drugom području za potpuno različit prijenos. Neizbježno, postoji određena razina smetnji od signala iz drugih stanica koje koriste istu frekvenciju, no ako postoji jedan običan odašiljač, samo se jedan prijenos može koristiti na bilo kojoj danoj frekvenciji. [4]

Ključna značajka mobilne mreže je mogućnost ponovnog korištenja frekvencija kako bi se povećala pokrivenost i kapacitet. Kao što je gore opisano, susjedne ćelije moraju koristiti različite frekvencije, međutim nema problema s dvije ćelije koje su dovoljno udaljene da djeluju na istoj frekvenciji, pod uvjetom da oprema korisnika (uređaj) mobilne mreže ne emitira previše energije.

Prva komercijalna mobilna mreža, generacija 1G, lansirana je u Japanu od strane Nippon Telegraph i Telephone (NTT) 1979. godine, najprije u gradskom području Tokija. U roku od pet godina, NTT mreža je proširena kako bi pokrila područje cijelog stanovništva Japana i postala prva 1G mreža u zemlji.

Mobilni telefoni rade slanjem i primanjem radio signala male snage. Signali se šalju i primaju od antena koje su priključene na radioodašiljače i prijemnike, tj. bazne stanice. Bazne stanice su povezane s ostalim mobilnim i fiksnim telefonskim mrežama i prenose signal, odnosno poziv na te mreže.

Najčešći primjer mobilne mreže je mreža mobilnog telefona (mobitela). Mobilni telefon je prijenosni telefon koji prima ili upućuje pozive putem stanice (bazne stanice) ili odašiljačkog tornja. Radio valovi koriste se za prijenos signala do i od mobitela. Moderne mreže mobilnih telefona koriste ćelije jer su radio frekvencije ograničene zajedničkim resursima. Svaka bazna stanica ima brojne radio stanice ili frekvencije za komunikaciju s mobilnim telefonima. Budući da je ovaj broj frekvencija ograničen, frekvencije se često ponovno koriste u susjednim stanicama. To se postiže smanjenjem razine snage bazne stanice kako bi se osiguralo minimalno ili nikakvo preklapanje pokrivenosti između ćelija. Stanice i uređaji mijenjaju frekvenciju pod kontrolom računala i koriste odašiljače male snage, tako da se obično ograničeni broj radio frekvencija može istovremeno koristiti od strane mnogih pozivatelja s manje smetnji. [2] [1]

Kada korisnik mobilnim uređajem pokušava uspostaviti poziv, prvi korak u postupku je da telefon provjeri postoji li pokrivenost u području u kojem je poziv upućen. Kada telefon potvrdi da postoji dovoljna jačina signala za upućivanje poziva, telefon uspostavlja vezu s

obližnjom baznom stanicom mobilnog telefona. Ova bazna stanica zatim uspostavlja poziv i zadržava poziv sve dok korisnik telefona ostane na pozivu i u dometu (području pokrivanja) te bazne stanice.

Praktično svaki mobilni sustav ima neku vrstu mehanizma emitiranja. To se može izravno koristiti za distribuciju informacija na više mobitela. Uobičajeno, na primjer u sustavima mobilne telefonije, najvažnija upotreba informacija emitiranja je postavljanje kanala za komunikaciju jedan-na-jedan između mobilnog primopredajnika i bazne stanice (eng. *paging*). Tri različite procedure pozivanja koje su općenito usvojene su sekvencijalni, paralelni i selektivni *paging*.

Pojedinosti o procesu *paginga* ponešto se razlikuju od tehnologije do tehnologije, ali obično znamo ograničen broj ćelija u kojima se telefon nalazi (GSM-2G, UTMS-3G i LTE-4G). *Paging* se vrši slanjem emitirane poruke svim tim ćelijama. *Paging* poruke mogu se koristiti za prijenos informacija. U ovakvom sustavu, dok se distribuirani mobilni korisnici kreću od ćelije do ćelije tijekom kontinuirane komunikacije, prebacivanje s jedne stanične frekvencije na drugu staničnu frekvenciju vrši se elektronski bez prekida i bez operatora bazne stanice ili ručnog prebacivanja. To se naziva primopredaja. Obično se novi komunikacijski kanal (frekvencija) automatski bira za mobilnu jedinicu na novoj baznoj stanici koja će ga poslužiti. Mobilna jedinica se zatim automatski prebacuje s trenutnog kanala na novi kanal i komunikacija se nesmetano nastavlja.

Mobilni operater koristi mobilnu mrežu kako bi postigao pokrivenost i kapacitet svojih pretplatnika. Velika geografska područja podijeljena su u skupine manjih ćelija, tzv. klastere, kako bi se izbjegao gubitak signala u vidnom polju i podržao veliki broj aktivnih telefona u tom području. Sve su stanice spojene na telefonske centrale, koje se pak povezuju s javnom telefonskom mrežom. [2]

Budući da gotovo svi mobilni telefoni koriste mobilnu tehnologiju, pojam "mobitel" je u nekim regijama, osobito u SAD-u, zamjenjiv s "mobilnim telefonom". Međutim, satelitski telefoni su mobilni telefoni koji ne komuniciraju izravno sa zemaljskim baznim stanicama, ali to mogu učiniti posredno putem satelita.

Prijelaz s postojećeg analognog na digitalni standard slijedio je vrlo različit put u Europi i SAD-u. Kao posljedica toga, u SAD-u su se pojavili višestruki digitalni standardi, dok su Europa i mnoge druge zemlje izabrale GSM standard. [7]

## 2.1. Bazna stanica

Bazna stanica mobilne telekomunikacije (BSMT) je jedinstven naziv za lokaciju na kojoj se nalaze primopredajni uređaji i odgovarajuća telekomunikacijska oprema, koja služi za povezivanje bazne stanice s ostalim dijelovima javne mobilne mreže. Ona je pozicionirana u točki dodira svih ćelija koje sadrži, odnosno središtu kopnene površine unutar koje se distribuira mreža.

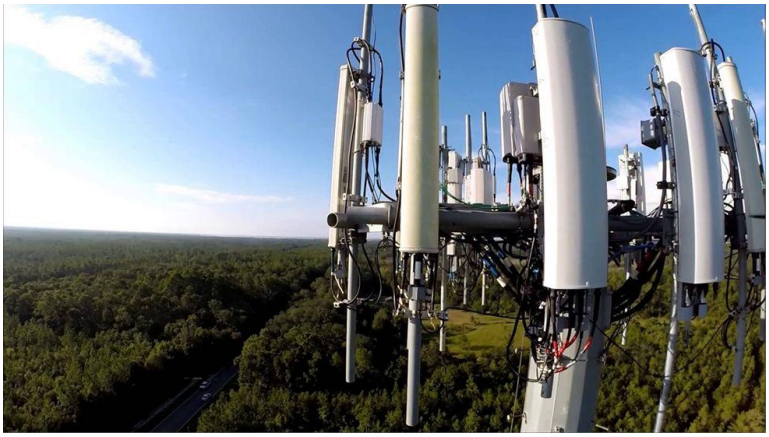
Slika 1. Primjer bazne stanice

Neki gradovi zahtijevaju da bazne stanice budu neprimjetne, moraju se „stopiti“ s okolinom, montirati na građevine ili reklamne konstrukcije. Ove se instalacije uglavnom nazivaju prikrivenim baznim stanicama ili nevidljivim stanicama.

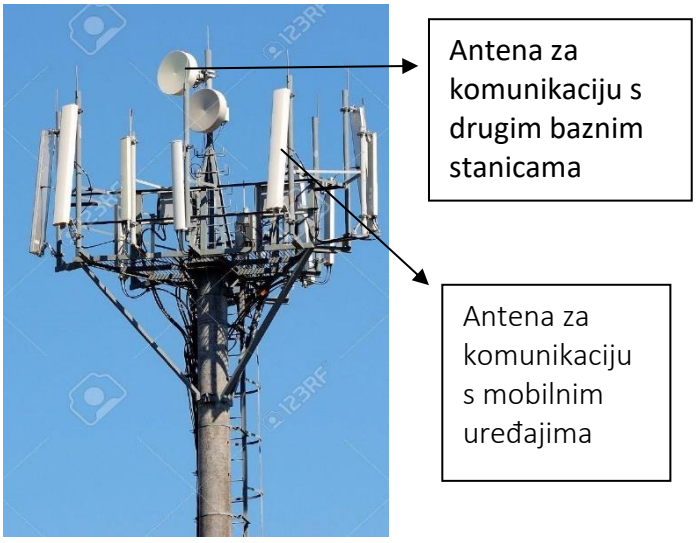
Sastoji se od dva glavna dijela, radio uređaja i antene (*Slika 3*.). Radio uređaj služi za prijem i slanje signala mobilnim uređajima korisnika. Prijem i slanje signala vrše se na različitim frekvencijama. Stanica može istovremeno raditi na sve tri mobilne tehnologije jedne mreže, ali će se razlikovati opseg frekvencije koju koristi.

Na mjesto stanice postavljaju se antene i oprema za elektroničku komunikaciju - obično na radio jarbol, toranj ili drugu uzdignutu strukturu - radi stvaranja ćelije u mreži (*Slika 1*.). Bazna stanica konstruirana na tornju sadrži i jedan ili više skupa primopredajnika / prijemnika,

digitalne procesore signala, upravljačku elektroniku, GPS prijemnik za vremenski raspon, primarni i rezervni izvor električne energije i skloništa.



Slika 2. Prikaz antena za komunikaciju s mobilnim uređajima

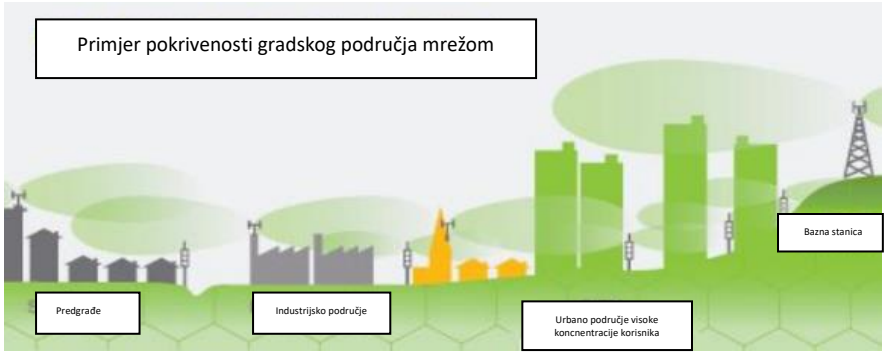



Slika 3. Prikaz antena za komunikaciju

Svaka bazna stanica može nositi konačan broj poziva. U područjima visoke uporabe mobilnih telefona, kao što su središnji poslovni prostori i područja visoke gustoće, potrebno je više baznih stanica za upravljanje prometom poziva. U područjima visoke iskoristivosti često

postoji niz baznih stanica, od vrlo specifičnih rješenja u zgradi (dizajniranih da daju kvalitetnu pokrivenost unutar određene zgrade), do vrlo malih baznih stanica poznatih kao "*mikroćelije*". Mikroćelije pokrivaju malo geografsko područje i često se nalaze na raskrižjima i na područjima s višim protokom korisnika. U ruralnim područjima ili područjima gdje uporaba mobilnih telefona nije tako visoka, bazne stanice često će se nalaziti na brdima ili visokim strukturama kako bi se maksimiziralo područje pokrivenosti.

Položaj bazne stanice unutar ćelije određen je brojnim čimbenicima, uključujući topografiju i druga fizička ograničenja kao što su drveće i zgrade, kapacitet ćelije ili broj poziva za koje se očekuje da će biti napravljeni u ćeliji, te radio frekvencija na kojoj će raditi bazna stanica. Po obrascu koji ovisi o karakteristikama terena i karakteristikama prijema, ćelije mogu biti šesterokutnih, kvadratnih, kružnih ili nekih drugih pravilnih oblika, iako se šesterokutne najčešće primjenjuju. Mobilni telefon mora imati ''pristup'' do bazne stanice mobilnog telefona. Drugim riječima, radio signal s telefona na baznu stanicu mora biti neprekinut (*Slika 4*.). Brda, drveće i visoke građevine mogu zakloniti ovu liniju pristupa, tako da bazne stanice moraju biti vrlo pažljivo locirane kako bi se povećala dostupnost pokrivenosti. Svakoj ćeliji dodijeljene su višestruke frekvencije (f1 - f6) koje imaju odgovarajuće radio baze. Skupina frekvencija može se ponovno upotrijebiti u drugim stanicama, pod uvjetom da iste frekvencije nisu korištene u susjednim stanicama, jer bi to uzrokovalo smetnje na zajedničkim kanalima. [12]

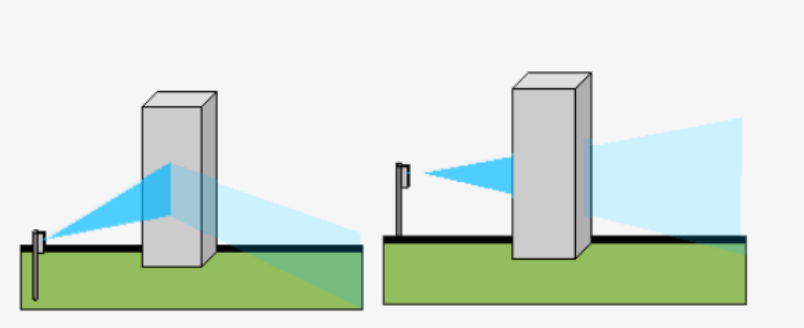



Slika 4. Ilustrativni prikaz pokrivenosti gradskog područja mrežom

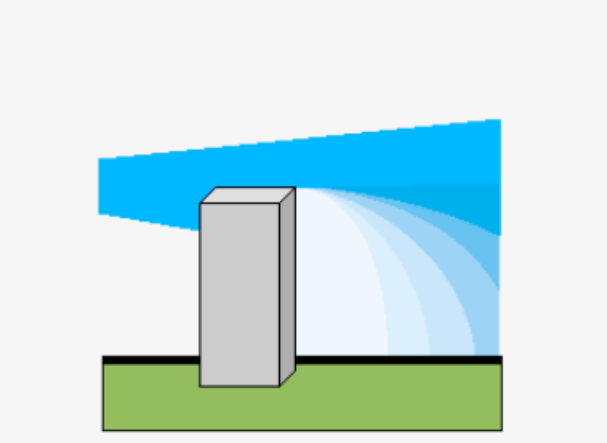
Stanična mreža je mreža ručnih mobilnih uređaja (mobitela) u kojoj svaki mobitel komunicira telefonskom mrežom radio valovima putem lokalne antene na staničnoj baznoj stanici.

Područje pokrivanja u kojem se pruža usluga podijeljeno je u mozaik malih geografskih područja koja se nazivaju "ćelije", a svako se od njih koristi zasebnim višekanalnim primopredajnikom i antenom na baznoj stanici. Svi mobiteli unutar stanice komuniciraju sa sustavom preko antene te stanice, na odvojenim frekvencijskim kanalima koje dodijeli bazna stanica iz zajedničkog baze frekvencija koje sustav koristi.

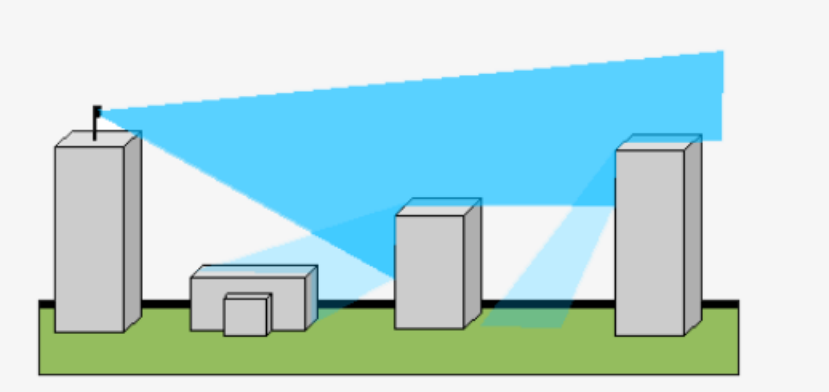
Mobitel se ponekad neće moći koristiti u svrhu komunikacije jer je predaleko od bazne stanice ili zato što je telefon na mjestu gdje su mobilni signali prigušeni i ograničeni zidovima zgrada, brežuljcima ili drugim građevinama. Signalima nije potrebna jasna linija vida, ali veća radio smetnja degradirat će ili eliminirati prijem. Kad mnogi ljudi pokušavaju istovremeno koristiti baznu stanicu, npr. tijekom zastoja u prometu ili sportskog događaja, na zaslonu telefona pojavit će se signal, ali onemogućeno je pokretanje nove veze. Drugi ograničavajući faktor za mobilne uređaje je mogućnost slanja signala iz baterije s malim naponom na mjesto ćelije. Neki mobiteli djeluju bolje od ostalih pod niskom potrošnjom energije ili slabom baterijom, obično zbog mogućnosti slanja dobrog signala s telefona na baznu stanicu. [13]



Slika 5. Prikaz signalnih valova bazne stanice koje se dijelom reflektiraju od zgradu, a dijelom prolaze kroz nju



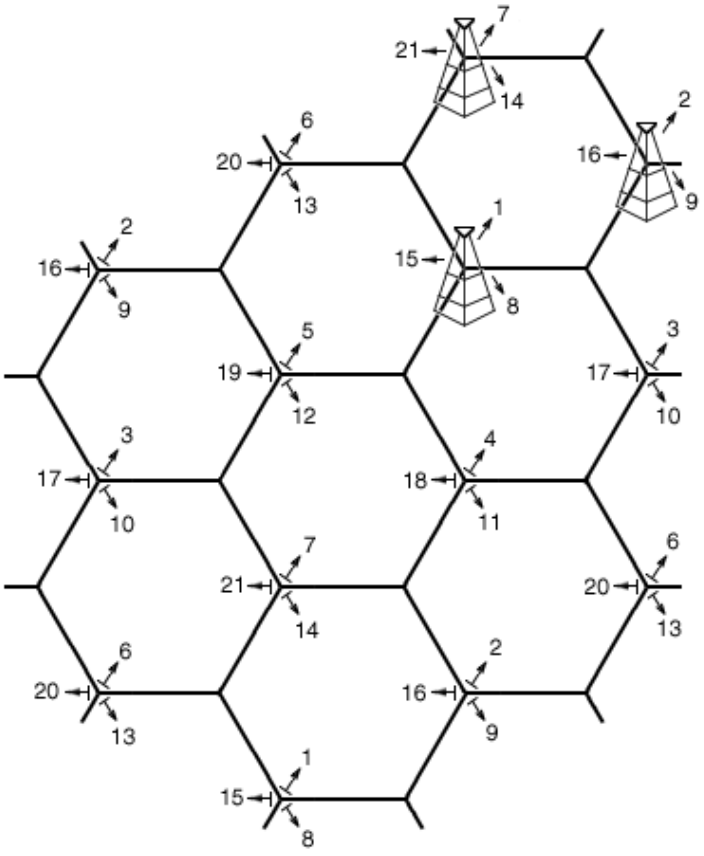
Slika 6. Primjer distrakcije signalnog vala



Slika 7. Prikaz moguće refleksije vala od više građevina

Svrha dobre organizacije baznih stanica je očuvanje radio opsega frekvencijskom upotrebom, radio signali male snage koji se koriste unutar svake ćelije ne putuju daleko izvan stanice, tako da se radio kanali mogu ponovo upotrijebiti u geografski razdvojenim ćelijama. Kada se mobilni korisnik prebaci iz jedne stanice u drugu, njegov se telefon automatski "predaje" anteni nove stanice i dodjeljuje mu novi skup frekvencija, a nakon toga komunicira s tom antenom. Ovaj postupak primopredaje pregledan je za korisnika i može se odvijati usred telefonskog poziva bez ikakvog prekida usluge. Svaki mobitel ima automatizirani dupleksni digitalni primopredajnik i komunicira sa staničnom antenom preko dva digitalna radio kanala.

Tipično se bazna stanica nalazi na rubu jedne ili više ćelija i prekriva više ćelija koristeći usmjerene antene. Uobičajena geometrija je locirati stanično mjesto na sjecištu triju susjednih ćelija, s tri antene pod kutom 120 °, a svaka pokriva jednu ćeliju. Vrsta antene koja se koristi za bazne stanice (vertikalni bijeli pravokutnici na *Slici 3*.), naziva sektorska antena, obično se sastoji od vertikalnog kolinearnog niza dipola. Ima plosnati oblik zračenja u obliku ventilatora, koji je blago nagnut prema dolje kako bi prekrivao stanično područje bez zračenja pod većim kutovima u daljnje ćelije koje ponovno koriste iste frekvencije. Visinski kut antene mora biti pažljivo podešen, tako da snop pokriva cijelu ćeliju bez zračenja predaleko. U modernim sektorskim antenama nagib snopa obično se može podesiti elektroničkim putem, kako bi se izbjegla potreba da se tehničar penje na toranj i mehanički naginje antenu kada je potrebno podešavanje.




Slika 8. Skica tornjeva baznih stanica i relacija prikazani šesterokutom

Iako su izvorne bazne stanice (tornjevi) stvorile ravnomjeran, višesmjerni signal, bazno stanična karta se može nacrtati pomoću baza koji se nalaze u kutovima šesterokuta gdje se konvergiraju tri stanice. Svaki bazna stanica ima tri seta antena usmjerenih u tri različita smjera sa 120° za svaku ćeliju (ukupno 360°) i primanje / prijenos u tri različite stanice na različitim frekvencijama. To osigurava najmanje tri kanala i tri baze za svaku ćeliju i uvelike povećava šanse primanja upotrebljivog signala iz najmanje jednog smjera.

U mrežama globalnog sustava za mobilnu komunikaciju (GSM), točan je naziv za bazu primopredajnika (BTS), a kolokvijalni sinonimi su "toranj mobilnog telefona" ili "bazna stanica". Više mobilnih davatelja često štedi novac montirajući svoje antene na zajednički toranj. Budući da zasebni sustavi koriste različite frekvencije, antene se mogu nalaziti blizu jedna bez druge. Neke tvrtke pružatelji usluga (mobilni operateri) imaju više mobilnih mreža i na sličan način koriste obojene bazne stanice za dvije ili više mobilnih mreža (npr. 3G i 4G). [12]

### *2.1.1. Domet ćelija*

Radni raspon ćelijskog mjesta (raspon u kojem se mobilni uređaji pouzdano povezuju s baznom stanicom) ovisi će o nizu faktora, a neki od njih su:

- visina antene nad okolnim terenom,
- učestalost signala koji se koristi,
- nazivna snaga odašiljača,
- potrebna brzina podataka veze pretplatničkog uređaja, odnosno korisnika,
- karakteristike usmjerenja nizova antene na mjestu,
- odbijanje i apsorpcija radio energije od zgrade ili vegetacije.

Također, domet može biti ograničen lokalnim geografskim ili regulatornim čimbenicima i vremenskim uvjetima.

Slika 9. Pokrivenost područja jedne bazne stanice sa tri sektorske antene

Uz to postoje ograničenja vremena u nekim tehnologijama (npr. u slobodnom prostoru GSM bi bio ograničen na 35 km iako je mogući domet čak 70 km uz posebnu opremu). Koncept "maksimalnog" dometa ćelije ipak je u praksi teško ostvariv. U gradovima, svaka stanica može imati domet do otprilike 0.8 km - 1,5 km, dok u ruralnim područjima raspon može biti i do 12

km. Moguće je da korisnik na jasnim otvorenim površinama može primati signale od mjesta na kojem je udaljen čak 40 km od bazne stanice.

Svakako je nužno osigurati dovoljno preklapanja ćelija za "primopredaju", tj. prebacivanje signala za mobilni uređaj s jedne bazne na drugu baznu stanicu, kao što je prijenos glasa putem GSM telefonskog poziva dok se korisnik kreće u automobilu ili vlaku. Potrebno područje preklapanja ne treba biti veliko kako bi se osigurala nesmetana „primopredaja“ uređaja.

U praksi se ćelije grupiraju u područjima velike gustoće stanovništva s najviše potencijalnih korisnika. Promet mobitela kroz jednu stranicu ograničen je kapacitetom bazne stanice, te postoji ograničen broj poziva ili podatkovnog prometa koji bazna stanica može istovremeno podnijeti. U prigradskim područjima bazne stanice su obično udaljene 2–3 km, a u gustim gradskim područjima bazne mogu biti udaljene čak 400–800 m.

Bazne stanice mogu podržati brojne razgovore s ograničenim brojem radio kanala (odsječci radio frekvencijskog spektra koji su potrebni za obavljanje jednog razgovora) licencirano operatera mobilnih usluga. Da bi se prevladalo ovo ograničenje, potrebno je ponoviti i ponovno koristiti iste kanale na različitim lokacijama. Kao što se autoradio mijenja s jedne lokalne postaje u potpuno drugu lokalnu postaju s istom frekvencijom kada putujete u drugi grad, isti se radio kanal ponovo koristi na baznoj stanici samo nekoliko kilometara dalje. Da bi to bilo moguće, signal ćelije namjerno se emitira malom snagom i u mnogim slučajevima je nagnut prema podu kako bi se ograničio njegov domet. To omogućava pokrivanje područja dovoljno malog da ne mora podržavati više razgovora nego što to mogu izdržati kanali. Zbog sektoriziranog rasporeda antena na baznoj stanici, moguće je mijenjati snagu i kut za svaki sektor, ovisno o pokrivenosti s drugim ćelijama na tom području. To su samo neki od parametara kojima operateri kreiraju kvalitetnu i visoko pristupačnu mobilnu mrežu. [13]

## 2.2. Mobilne tehnologije

"G" označi "generaciju" mobilne tehnologije. Svaka generacija je brža, sigurnija i pouzdanija. Faktor pouzdanosti je najteža prepreka koju je potrebno savladati. No važnije, svaka je generacija definirana kao skup standarda telefonske mreže, koji detaljno opisuju tehnološku implementaciju određenog sustava mobilne telekomunikacije. Brzina se povećava i tehnologija koja se koristi za postizanje te brzine također se mijenja. Na primjer, 1G nudi brzinu do 2,4 kbps, 2G do 64 kbps i temelji se na GSM-u, 3G do 144 kbps - 2 Mbps, dok 4G nudi brzinu prijenosa podataka do 100 Mbps - 1 Gbps i temelji se na LTE (eng. *long term evolution*) tehnologiji. Koju tehnologiju uređaj koristi u određenom trenutku moguće je vidjeti dok je uređaj spojen na internet, jer brzina interneta ovisi o jačini signala koja je prikazana kao 2G, 3G, 4G itd. tik do trake sa signalima na početnom zaslonu mobilnog uređaja, a to ujedno i označava korištenu tehnologiju. [3]

Cilj bežične komunikacije je pružiti visokokvalitetnu, pouzdanu komunikaciju poput žičane komunikacije, a svaka nova generacija usluga predstavlja veliki korak u tom smjeru.

Ovo evolucijsko putovanje započelo je 1979. godine iz 1G-a i još uvijek traje do 5G. Svaka od generacija ima standarde koji se moraju ispuniti kako bi službeno koristili G terminologiju. Postoje institucije zadužene za standardizaciju svake generacije mobilne tehnologije. Svaka generacija ima zahtjeve koji određuju parametre kao što su propusnost ili kašnjenje koje tehnologija treba ispuniti da bi se smatrala dijelom te generacije. Svaka je generacija izgrađena na istraživanju i razvoju koji se dogodio od posljednje generacije. 1G nije korišten za identifikaciju bežične tehnologije sve dok 2G, ili druga generacija, nije zaživjela među korisnicima. To je bio veliki skok u tehnologiji kada su bežične mreže prešle s analognog signala na digitalni. 3G je omogućio brži prijenos podataka za multimedijsku upotrebu (najmanje 200 kbps) i dugo je držao standard za bežične prijenose podataka. [5]

### *2.2.1. 1G - prva generacija*

To je bila prva generacija tehnologije mobilne mreže. Jednostavni telefonski pozivi bili su sve što je tehnologija omogućavala. Prva generacija komercijalne mobilne mreže uvedena je krajem 1970-ih godina s potpuno implementiranim standardima koji su uspostavljeni tijekom 1980-ih. 1G je analogna tehnologija i telefoni su općenito imali slabi vijek trajanja baterije. Kvaliteta glasa bila je velika, no bez sigurnosti zadržavanja poziva, a nerijetko bi pozivi bili prekinuti. Radi se o telekomunikacijskim standardima koji su se koristili sve do zamjene digitalnim telekomunikacijama unutar 2G.

Prednosti ove tehnologije su komunikacijski sustavi s potpunim pristupom, no uz spomenute nedostatke, 1G tehnologija nije omogućavala korištenje „roaming“ sustava, postojao je ograničen broj korisnika i pokrivenost ćelija (znatno manji nego u tehnologijama viših generacija), mala je sigurnost korisnika u smislu da se pozivi lako detektiraju, mobilni uređaji su bili preveliki i s malim kapacitetom baterija. [5]

Svojstva 1G tehnologije:

- frekvencija 800 MHz i 900 MHz,
- širina pojasa: 10 MHz,
- tehnologija: analogno prebacivanje,
- usluga: samo glas,
- modulacija: frekvencijska modulacija (FM).

### *2.2.2. 2G - druga generacija*

GSM (Globalni sustav za mobilne komunikacije) je standard koji je razvijen od strane Europskog instituta za telekomunikacijske standarde (ETSI) kako bi opisao protokole za drugu generaciju (2G) mreža koje koriste mobilni uređaji poput mobilnih telefona i tableta. Druga generacija prvi put je razmještena u Finskoj u prosincu 1991. Od 2014. godine postala je globalni standard za mobilne komunikacije - s preko 90% tržišnog udjela, a djeluje u više od 193 zemlje. GSM tehnologija kasnije je postala osnovni standard za daljnji razvoj bežičnih

standarda. Ovaj je standard mogao podržavati do 14,4 do 64 kbps (maksimalnu) brzinu prijenosa podataka koja je dovoljna za SMS i e-mail usluge.

Prva velika promjena jest nadogradnja s 1G na 2G. Glavna razlika između dvaju mobilnih telefonskih sustava (1G i 2G) je u tome što su radio signali koje koristi 1G mreža analogni, dok su 2G mreže digitalne. Glavni motiv ove generacije bio je osigurati siguran i pouzdan komunikacijski kanal. Implementirao je koncept CDMA (eng. *code-division multiple access*) i GSM. Pružena je mala podatkovna usluga kao što su SMS i MMS. Napredak u tehnologiji od 1G do 2G uveo je mnoge temeljne usluge koje i danas koristimo, kao što su SMS, roaming, konferencijski pozivi, zadržavanje poziva i naplata na temelju usluga, npr. naknade temeljene na međugradskim pozivima i naplati u stvarnom vremenu. Maksimalna brzina 2G-a s GPRS-om je 50 kbps ili 1 Mbps s poboljšanom brzinom prijenosa podataka za GSM evoluciju EDGE (Enhanced Data Rate for GSM Evolution ili EGPRS). Prije nego što je napravio veliki skok od 2G do 3G bežičnih mreža, manje poznati 2.5G i 2.75G bili su privremeni standardi. Kako bi se podržala veća brzina prijenosa podataka, uvedena je i uspješno implementirana GPRS (General Packet Radio Service). GPRS je imao mogućnost prijenosa podataka do 171 kbps (maksimalno). EDGE (Enhanced Data GSM Evolution) je također razvijen za poboljšanje brzine prijenosa podataka za GSM mreže. EDGE je bio sposoban podržati do 473.6 kbps (maksimalno). [8]

Svojstva 2G tehnologije:

- digitalni sustav,
- omogućene SMS usluge,
- omogućen roaming,
- šifrirani prijenos glasa,
- prvi internet na nižim brzinama prijenosa podataka,
- ograničen broj korisnika i mogućnost infrastrukture,
- ograničena mobilnost korisnika.

### *2.2.3. 3G - treća generacija*

Univerzalni sustav mobilnih telekomunikacija (UMTS) je mobilni sustav treće generacije za mreže temeljene na GSM standardu, te predstavlja standarde mreže i usluga na koji su korisnici i danas naviknuti. Pretraživanje weba, e-pošta, preuzimanje videozapisa, dijeljenje slika i druge tehnologije pametnih telefona uvedeni su u treću generaciju. Komercijalno je uvedena 2001. godine. Ciljevi postavljeni pred mobilnu komunikaciju treće generacije bili su olakšavanje većeg kapaciteta glasa i podataka, podrška širokom rasponu aplikacija i povećanje prijenosa podataka uz niže troškove.

Ova mreža kombinira aspekte 2G mreže s nekim novim tehnologijama i protokolima kako bi pružila znatno višu brzinu prijenosa podataka.

Svojstva 3G tehnologije:

- veća brzina prijenosa podataka,
- omogućen video poziv,
- viša sigurnost korisnika,
- veći broj korisnika,
- veća pokrivenost područja,
- podrška za mobilne aplikacije,
- podrška za multimedijske poruke,
- lakše i brže pregledavanja internet stranica,
- TV streaming.

UMTS (Univerzalni sustav mobilnih telekomunikacija) koristi tehnologiju radijskog pristupa za širokopojasni pristup s višestrukim pristupom (WCDMA) radi pružanja veće učinkovitosti i propusnosti za mobilne mrežne operatore. UMTS specificira cjeloviti mrežni sustav koji uključuje radio pristupnu mrežu (UMTS - zemaljska radijska pristupna mreža ili UTRAN), jezgrenu mrežu (Mobile Application Part ili MAP) i provjeru autentičnosti korisnika putem SIM kartica (modul identiteta pretplatnika). Tehnologija opisana u UMTS-u ponekad se naziva i sloboda pristupa multimedijskim sadržajima (FOMA) ili 3GSM.

Za razliku od EDGE (IMT Single-Carrier, na temelju GSM-a) i CDMA2000 (IMT Multi-Carrier), UMTS zahtijeva nove bazne stanice i nove raspodjele frekvencija, što je iziskivalo određene troškove. Samim time su i mobilni uređaji bili skuplji.

3G podržava konvencionalne mobilne glasovne, tekstualne i MMS usluge, ali također može prenositi podatke velikom brzinom, omogućujući mobilnim operaterima da isporuče aplikacije, uključujući streaming i širokopojasni pristup internetu. [10]

3G ili mreže treće generacije rade na drugačiji način od 2G mreža. Kada se poziv uspostavi na 2G, linija se drži otvorenom za razgovor korisnika tijekom cijelog trajanja poziva. S 3G mrežama, podaci poslani preko njih podijeljeni su u male „pakete podataka“ koji se ponovno sastavljaju u ispravnom redoslijedu na kraju primitka. Ovo „pametno“ kodiranje znači da se može poslati više podataka i da se šalje učinkovitije i brže. Osim toga, 3G telefoni mogu biti u kontaktu s više od jedne bazne stanice u isto vrijeme i to osigurava bolje performanse u kvaliteti glasa i brzinama prijenosa podataka, odnosno više istovremenih poziva se može dogoditi u istom frekvencijskom području.

3.5G do 3.75G sustavi razvijeni su kako bi se povećala brzina prijenosa podataka u postojećim 3G mrežama, uvedena su još dva tehnološka poboljšanja u mreži. HSDPA (eng. *High Speed Downlink*) paketni pristup i HSUPA (eng. *High Speed Uplink Packet Access)*, razvijen je i implementiran u 3G mreže. 3.5G mreža može podržavati brzinu prijenosa podataka do 2 Mbps. 3.75G sustav je poboljšana verzija 3G mreže s HSPA + High Speed paketnim pristupom. Kasnije će se ovaj sustav razviti u snažniji 3.9G sustav poznat kao LTE (*Long Term Evolution*), koji danas smatramo 4G tehnologijom. [9]

### *2.2.4. 4G – četvrta generacija*

4G je vrlo različita tehnologija u usporedbi s 3G-om i omogućena je praktično samo zahvaljujući napretku tehnologije u posljednjih 10 godina. Ideja iza razvoja 4G bila je pružiti korisnicima veliku brzinu, visoku kvalitetu i visoki kapacitet, a istovremeno povećati sigurnost i smanjiti troškove glasovnih i podatkovnih usluga, multimedije i interneta putem IP-a. Potencijalne i trenutne aplikacije uključuju izmijenjeni pristup mobilnom webu, usluge igranja, mobilni TV visoke razlučivosti, video konferencije, 3D televiziju, itd.

Dva važna 4G standarda su WiMAX (do danas se prestao koristiti) i LTE (proširena s implementacija). LTE je serija nadogradnji postojeće UMTS tehnologije uvedena na postojeći

Telstra frekvencijski pojas od 1800 MHz. Maksimalna brzina 4G mreže kada se uređaj kreće je 100 Mbps ili 1 Gbps za komunikaciju niske mobilnosti neovisno o kretanju korisnika, latencija smanjena s oko 300 ms na manje od 100 ms i postignuta znatno niža zagušenja. Također, potrebno je dijeliti mrežne resurse kako bi podržao više istodobnih veza na ćeliji. Kako se razvija, 4G bi mogao nadmašiti brzinu prosječne bežične širokopojasne kućne internetske veze. Nekoliko uređaja bilo je sposobno za punu brzinu kada je tehnologija prvi put izdana. Stvarna pokrivenost 4G u praksi bila je ograničena na velika gradska područja. Izvan pokrivenih područja, 4G telefoni su se povukli prema 3G standardima. Kada je 4G prvi put postao dostupan, jednostavno je bio malo brži od 3G. 4G nije isto što i 4G LTE koji je vrlo blizu ispunjavanju navedenih kriterija standarda za 4G tehnologiju.

Simultani prijenos glasa i podataka moguć je uz LTE sustav koji značajno poboljšava brzinu prijenosa podataka. Sve usluge, uključujući glasovne usluge, mogu se prenijeti putem IP paketa (paket podataka).

Poput 2G-a, 3G se razvio u 3.5G i 3.75G jer je uvedeno više značajki kako bi se donijela 4G tehnologija. 3G telefon ne može komunicirati putem 4G mreže, ali novije generacije telefona praktički su uvijek dizajnirane da budu kompatibilne unatrag, tako da 4G telefon može komunicirati putem 3G i 2G ako je potrebno. LTE napredna bežična tehnologija koja se koristi u sustavima 4G ima kompatibilnost s prethodnom verzijom, tako da je moguća lakša implementacija i nadogradnja LTE naprednih mreža u već postojeću infrastrukturu. Ipak široka primjena i nadogradnja oduzimaju mnogo vremena i novaca. Također, zakup frekvencijskog spektra u državama diljem svijeta je skup. [11][12]

Svojstva 4G tehnologije:

- veća brzina prijenosa podataka, do 1Gbps
- poboljšana sigurnost i mobilnost
- VoLTE putem LTE mreže (koristite IP paketa za glas)
- prijenos uživo (eng. *streaming*) videozapisa visoke kvalitete
- omogućen video poziv
- podrška za mobilne aplikacije
- podrška za multimedijske poruke

## 2.3. Key Performance Indicators - KPI

KPI-evi se smatraju ključnim pokazateljima mjerenja koja se koriste za identifikaciju i klasificiranje poslovnih rezultata, rada nekog sustava ili pogona. Iako se u ovom radu ne koriste direktno KPI podaci baznih stanica i cijele mreže, važno je objasniti iste radi lakšeg razumijevanja i kontrole rada bazne stanice, jer oni u konačnici pokazuju mogu li parametri kojima su definirane bazne stanice dati zadovoljavajuću kvalitetu mreže krajnjem korisniku.

Da bi se identificirali i uspostavili ključni KPI-i za određen sustav, proces koji se opisuje mora biti kreiran da zadovolji sljedeće zahtjeve:

- mora sadržavati jasne ciljeve
- proces mora biti mjerljiv, kvantitativno i kvalitativno
- mora biti moguće identificirati i riješiti organizacijska odstupanja

Ideja uvođenja KPI-eva u praćenje rada određenih procesa javlja se potrebnom da se isti lakše prate, kontroliraju i mijenjaju. Ono što se ne može izmjeriti i usporediti, ne može se niti poboljšati. Iako je ovaj pojam suštinski poslovno orijentiran, ima široku primjenu u opisu rada i kvalitete mobilnih mreža unutar svih korištenih tehnologija.

KPI se koriste u mnogim različitim djelatnostima kako bi se odredila učinkovitost poduzeća. Izbor i organizacija tih mjerenja su u djelokrugu uprave. Određeni KPI-evi mogu biti pregledani od strane operativnog osoblja kako bi se napravila poboljšanja. Druge se dostavljaju poslovnim jedinicama radi obavljanja izvršnih radnji. KPI-evi se također koriste za provjeru da su ugovori s kupcima pravilno ispunjeni. U bilo kojoj organizaciji te formule i vrijednosti mogu biti prilagodljive i imaju različite oblike i namjene.

KPI mjerenja nude kvantificirane podatke koji će pomoći da se donesu odluke o eventualnim promjenama.

Mrežni pružatelji usluga mjere učinkovitost kako bi osigurali najbolju moguću uslugu. Koriste ključne pokazatelje uspješnosti (KPI). To su vrijednosti utvrđene za kvantificiranje specifičnih aspekata funkcionalne mreže. KPI izvješća također su učinkoviti alati za inženjerske odjele u projektiranju, planiranju i upravljanju kapacitetima postojećih i budućih mreža.

KPI-evima se smatraju skup izračunatih pokazatelja rada mreže, prema unaprijed definiranim formulama. Svi parametri rada mreže se mjere i određenim formulama u

konačnici prikazuju kvantitetu i kvalitetu same mreže na određenom području. Svakodnevnim praćenjem KPI-eva i njihovih promjena omogućen je efektivan rad na optimizaciji mreže.

KPI koji se prate za rad mobilnih mreža u osnovi se dijele u 5 grupa:

- *Accessibility*
- *Retainability*
- *Mobility*
- *Integrity*
- *Availability*

„*Accessibility*“ predstavlja sposobnost mreže da se uspostavi poziv iz perspektive korisnika.

„*Retainbility*“ jest sposobnost sustava da zadrži poziv i da ga održava uz što manje smetnji i komplikacija.

„*Mobility*“ predstavlja sposobnost mreže da prebacuje pozive između frekvencija i ćelija (prijenos glasa ).

„*Intergrity*“ jest skupina KPI-eva koji daju podatke o brzini mreže te sposobnosti mreže da prenosi paketne podatke.

„*Availability*“ je zadnja, ali moglo bi se reći najvažnija skupina KPI-eva. Pokazuje dostupnost mreže krajnjem korisniku u svakom trenutku. Teži se vrijednosti od 100%, no uvijek se dopušta odstupanje od 5%. Dakle, 95% bi bila donja granica tolerancije zadovoljavajuće dostupnosti mreže, u ovisnosti o korištenju mreže, geografskom položaju, protoku korisnika kroz područje dometa bazne stanice i sl.

Upravljanje mrežom je složen pothvat i pojavljuju se problemi. Mrežni upravitelji - optimizatori često postavljaju osnovicu za operacije. Mrežni dizajneri određuju optimalne vrijednosti i pragove učinkovitosti, koji se zatim integriraju u alate za upravljanje mrežom.

Važno je zadržati mrežu u radu bez poteškoća i smetnji paralelno ju pokušavajući i unaprijediti. Tu dolazimo do optimizacije mreže. Dugoročna profitabilnost ovisi o pružanju najbolje moguće usluge kupcu na dogovorenoj razini. Iako timovi za kontrolu kvalitete ne

mogu biti pod stresom stalnog nadzora mreže, često su zaduženi za dubinsku analizu kako bi odredili najbolji put do poboljšanja mreže i visoku učinkovitost iste.

Bitno je napomenuti da vrijednost KPI poprilično varira ako se uzme u obzir u kojoj je fazi mreža koju pratimo, tako će npr. u fazi podizanja sustava i mreže KPI vrlo vjerojatno biti loši dok se mreža adekvatno ne optimizira i dok se ne ustabili njezin rad. Također, promjena KPI na nekom području mreže može se očitovati u specifičnim situacijama poput Nove godine, nogometnih utakmica, koncerata. Tamo gdje se očekuje veliki broj ljudi u isto vrijeme, a mreža je konfigurirana za prosječan (manji) broj ljudi, očekuju se privremeno lošije vrijednosti. Situacijama koje se mogu predvidjeti pristupa se poboljšanjem kapaciteta mreže unaprijed. [4]

# 3. STROJNO UČENJE I OBRADA PODATAKA

Strojno učenje grana je umjetne inteligencije koja se bavi oblikovanjem algoritama koji svoju učinkovitost poboljšava na temelju empirijskih podataka, odnosno skupina algoritama za automatsku obradu podataka. Strojno učenje tvori temelj današnje računalne znanosti. Obrada podataka metodama strojnog učenja rezultira prediktivnim modelom, no primjene su puno šire od same predikcije, tako da se strojno učenje koristi za bilo koje preslikavanje ulaznih i izlaznih podataka koje je preteško ručno unijeti ili za koje ne postoje jasno precizirana pravila koja bi se unijela, ili se pak ta pravila prečesto mijenjaju.

Interdisciplinarnost strojnog učenja vidljiva je iz istovremene primjene računalne znanosti, statistike i vjerojatnosti, optimizacije i upravljanja, numeričke matematike i linearne algebre, teorije informacija, filozofije, psihologije, neurobiologije, kognitivne znanosti i dr.

Strojno učenje jedno je od danas najaktivnijih i najuzbudljivijih područja računske znanosti, ponajviše zbog brojnih mogućnosti primjene koje se protežu od raspoznavanja uzoraka i dubinske analize podataka do robotike, računalnog vida, bioinformatike, računalne lingvistike, itd.

Upoznatost s metodama nadziranog i nenadziranog strojnog učenja te zadatcima klasifikacije i regresije, razumijevanje generativnih, diskriminativnih, parametarskih i neparametarskih modela, razumijevanje teorijskih osnova ovih modela, ugrađenih pretpostavki, njihovih prednosti i nedostataka omogućuje sposobnost oblikovanja i implementacije sustava za klasifikaciju, regresiju ili grupiranje te provođenja vrednovanja takvog sustava. [15]

Računala su u prošlosti mogla raditi samo ono za što su bila programirana. Strojno učenje omogućava im da uče na sličan način kako to rade ljudi, tako da stroj prikuplja znanje bazirano na prošlom iskustvu. Umjesto da mu se stalno mora ažurirati softverski kod, on je, kako vrijeme prolazi, samostalno sposoban poboljšavati svoj rad.

Za razliku od statistike koja se obično primjenjuje na „manji“ skup podataka i gdje je uloga samog statističara znatno veća od uloge računala i pomoćnih alata, strojno učenje se zasniva na automatiziranom korištenju i otkrivanju pravilnosti u podacima. Cijeli proces obrade podataka znatno ovisi o računalu i pomoćnim alatima, te o karakteriziranju podataka

u smislu što je „naučljivo“ i pod kojim uvjetima. Cilj je izgraditi model koji objašnjava podatke i omogućuje predviđanje i zaključivanje.

Učenjem se smatra memoriranje podataka s istovremenim zaključivanjem. Može se reći da je to učenje stroja (programiranje računala) da optimizira neki kriterij uspješnosti temeljem podatkovnih primjera ili iz prethodnih iskustva. Raspolaže se modelom koji je definiran parametrima (slobodnim) i optimizacijom tih parametara model mora moći predvidjeti svojstva novih, do sada neviđenih podataka.

Metodu, odnosno algoritam strojnog učenja, čine funkcija vrijednosti slobodnih parametara, algoritam pretraživanja i optimiziranja te metoda procjene greške. Funkcija može biti linearna, polinom, skup pravila, stabla odluka ili neuronska mreža. Evaluacija će se dobiti metodom procjene greške, odnosno skalarnom funkcijom koja će kvalificirati kvalitetu naučenog modela, tj. mjeru greške.

Opća podjela strojnog učenja jest na nadzirano i nenadzirano učenje. Pod nadzirano učenje (eng. *supervised learning*) smatra se obrada podataka kojoj je cilj napraviti model koji će raditi predviđanja za neke nove podatke (npr. klasifikacija objekata, regresija, predviđanje prodaje, cijena ili rizika, itd.). Ovim metodama će se eksplicitna informacija o danim primjerima i vrijednost njihove ciljne varijable koristiti za dobivanje cilja. [15]

Nenadzirano učenje (eng. *unsupervised learning*) bez anotacije ili povratne informacije o kategorizaciji ulaznih podataka, grupira prema nekoj strukturnoj pravilnosti unutar podataka koja će se odrediti ovisno o metodi. Nema učitelja niti kritičara, stoga je potrebno pronaći pravilnosti prema ulaznim podacima. Metoda neće otkriti što podaci znače i koja im je primjena, samo će ih organizirati prema izračunatoj pravilnosti.

Većina razvoja i probijanja u strojnom učenju jest u domeni nenadziranog učenja.

## 3.1. Primjena strojnog učenja

Primjena je vrlo široka. Rješavaju se složeni problemi koje nije moguće riješiti na klasičan algoritamski način, kao što su raspoznavanje uzorka, računalni vid, obrada prirodnog jezika, raspoznavanje emocija ili govora, robotika, itd. Koristi se kod sustava koji se moraju dinamički mijenjati i prilagođavati korisničkim sučeljima ili kada je potrebno obraditi velike količine podataka i provjeriti ima li znanja u njima.

Učenje se primjenjuje na različite baze podataka u raznim granama djelovanja kako bi se otkrile nepravilnosti implicitno sadržane u podacima.

Najčešće primjene algoritama strojnog učenja su u programskim implementacijama koje nije moguće napraviti na klasičan način, otkrivanje znanja unutar skupova te primjena u programima kojima je potrebno osigurati dinamičnu promjenu i prilagođavati ih određenim uvjetima.

Primjeri upotrebe strojnog učenja su klasa pacijenta s visokim rizikom za carski rez ili analizirati rizik banke kod dodjele kredita građanima. Neki od svakodnevnih primjera s kojima se većina današnjice susreće:

- *Google* koristi strojno učenje da poboljša preciznost rezultata traženja,
- *Facebook* prikazuje postove ovisno o vašim interesima i prošlom ponašanju na društvenoj mreži,
- *Netflix* vam predlaže popis sadržaja koje stvara zahvaljujući strojnom učenju,
- „samovozeći“ automobili prate objekte iz okruženja i koriste te podatke da bi poboljšali svoje vozačke sposobnosti,
- digitalni pomoćnici koriste strojno učenje da bi unaprijedili tehnologiju prepoznavanja govora.

Zanimljivo je napomenuti da i zdravstvena industrija eksperimentira s velikim brojem primjena, uključujući predviđanje životnog vijeka, organiziranje pacijentovih podataka, čak i dijagnosticiranje određenih bolesti. Prikupljanjem podataka poput rendgenskih slika, genetskih profila i krvnih testova, pokušava se postići da računala daju bržu i sigurniju dijagnozu.

Poslovni svijet također koristi benefite strojnog učenja. Razna poduzeća se odlučuju za prikupljanje i obradu podataka u svrhu boljeg poslovanja i boljeg razumijevanja tržišta.

Strojno učenje posjeduje vrlo široku primjenu metoda (umjetne inteligencije) za rješavanje složenih problema na gotovo svim područjima ljudske djelatnosti. Toliko je postalo zanimljivo da neki jednostavno individualno eksperimentiraju s tom vrstom znanja samo kako bi preispitali vlastite dosege.

Osim očitih prednosti razvoja strojnog učenja, paralelno s vrlo brzim razvojem istog, postavljaju se razna pitanja etičke prirode. Ljudi su zabrinuti za svoja radna mjesta i brinu da će zbog strojnog učenja gubiti poslove. Takve promjene značajno ovise o industriji, ali potencijal za dramatičnu promjenu mnogih poslova, poput vozača, bankara, čak i određenih liječnika, zasigurno postoji.

Također, često se postavljaju pitanja moralnog karaktera u vezi strojnog učenja i naglašava se njegov potencijal da naruši ljudsku privatnost. Primjerice, ne bi bilo teško stvoriti program koji bi prikupljao bilo čije poruke, podatke i informacije čime se otvara mogućnost zlouporabe istih. Strojno učenje može biti vrlo moćan alat, no važno je imati na umu i njegove negativne posljedice i efekte. [15][16]

## 3.2. Grupiranje (eng. *clustering*)

Grupiranje je proces podjele populacije ili podatkovnih točaka u više skupina tako da su podatkovne točke u istim skupinama sličnije drugim točkama podataka u istoj skupini od onih u drugim skupinama. Jednostavnim riječima, cilj je razdvojiti skupine sa sličnim osobinama i dodijeliti ih klasterima. Ideja grupiranja jest stvaranja klastera podataka, tako da svaki pojedinačni klaster ima najsličnije točke. To je u osnovi zbirka predmeta na temelju sličnosti i različitosti među njima. Ovo je najpopularnija tehnika nenadziranog učenja. Može koristiti dovoljno jednostavne algoritme za lako shvaćanje i razumijevanje, ali samo grupiranje ima mnogo težinskih slojeva. Grupiranje nije ograničeno na jednostavne algoritme.

Grupiranje se može podijeliti u „tvrdo“ i „meko“. Razlika je u tome što se u tzv. „tvrdom “ grupiranju svaka podatkovna točka (podatak) nastoji dodijeliti nekoj grupi tako da joj potpuno pripada. Dok u „mekom“ grupiranju umjesto svrstavanja svake podatkovne točke u zaseban klaster, dodjeljuje se vjerojatnost da će određena točka pripasti određenom klasteru.

Ne postoje kriteriji za dobro grupiranje. Ovisno o korisniku, samoj prirodi podataka i očekivanim rezultatima, mogu se koristiti sve metode koji zadovoljavaju njihove potrebe. [18]

Metode grupiranja:

1. Metode temeljene na gustoći

   Ove metode povezuju guste regije područja koja imaju neku sličnost i razlikuje se od druge guste regije prostora. Ova metoda ima dobru točnost i sposobnost spajanja dva klastera.

   Primjeri: Prostorno grupiranje aplikacija zasnovanih na gustoći, Točke za određivanje radi identificiranja strukture klastera, itd.

2. Hijerarhijske metode

   Klasteri formirani ovom metodom formiraju strukturu tipa stabla koja se temelji na hijerarhiji. Novi klasteri se formiraju koristeći prethodno formirani klaster.

   Primjeri: grupiranje pomoću zastupnika, uravnoteženo iterativno smanjenje grupiranje i korištenje hijerarhije) itd.

3. Metode particioniranja

   Ove metode dijele objekte na „k“ klastera i svaki dio čini jedan klaster. Spada pod „tvrde“ metode.

4. Metode temeljene na mreži

   U ovoj se metodi prostor podataka formulira u konačni broj ćelija koje tvore mrežnu strukturu. Sve operacije klastera izvedene na ovim mrežama su brze i neovisne o broju objekata podataka.

   Primjeri: Statistička informacijska mreža, valni klaster, CLIQUE (*eng. clustering in quest*), itd.

### *3.2.1. K-means grupiranje*

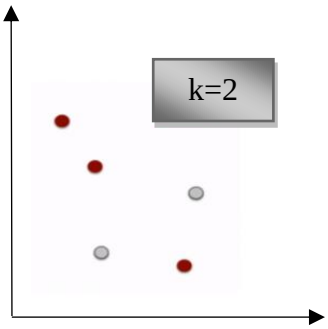
Budući da je zadatak grupiranja subjektivan, sredstava koja se mogu koristiti za postizanje tog cilja je mnogo. Svaka metodologija slijedi drugačiji skup pravila za definiranje "sličnosti" među točkama podataka. Poznato je više od 100 algoritama grupiranja, no pojedini se algoritmi upotrebljavaju vrlo često u praksi. „K-means“ grupiranje je vrlo često korišten i relativno jednostavan algoritam strojnog učenja bez nadzora temeljen na daljini. Za razliku od tradicionalnih nadziranih algoritama strojnog učenja, K-means pokušava klasificirati podatke bez da se prethodno obučio s označenim podacima. Nakon što je algoritam pokrenut i grupe definirane, svi novi podaci mogu se dodijeliti najrelevantnijoj grupi. Podaci se mogu grupirati na temelju centroida, distribucije, gustoće, itd. K-means to čini pomoću centroida.

Centroidni modeli su iterativni algoritmi grupiranja u kojima je pojam sličnosti izveden iz blizine podatkovne točke u središtu klastera. K-means algoritam grupiranja popularan je algoritam koji spada u ovu kategoriju. U tim modelima, broj klastera potrebnih na kraju moraju biti prethodno spomenuti, zbog čega je važno imati prethodno znanje o skupu podataka. Ovi se modeli iterativno prikazuju kako bi pronašli lokalni optimumi.

Minimiziranjem sume kvadrata udaljenosti unutar klastera, K-means pokušava rasporediti „n“ broj mjerenja u „k“ broj klastera tako da se svako mjerenje usmjeri u klaster s najbližom srednjom vrijednošću. K-means je iterativni algoritam grupiranja kojem je cilj pronaći lokalne maksimume u svakoj iteraciji. [17]

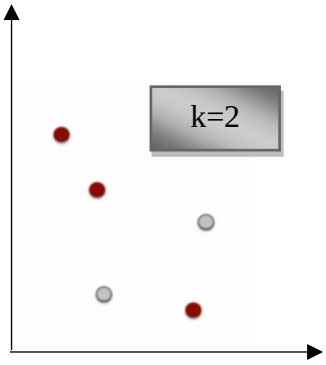
Provodi se kroz sljedećih 5 koraka:

1. Za određen broj podataka, potrebno je odabrati željeni broj klastera, odnosno grupa kao što je prikazano na *Slici 10*.



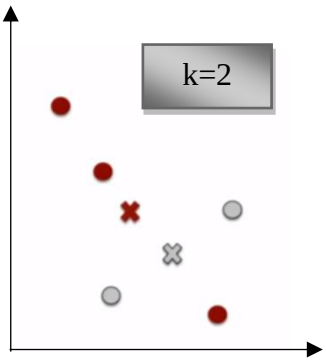

Slika 10. Prikaz „n“ broja podataka i željeni „k“ broj klastera

2. Nasumično svaku podatkovnu točku dodjeljuje jednom od klastera kao što je prikazano sa *Slici 11*.



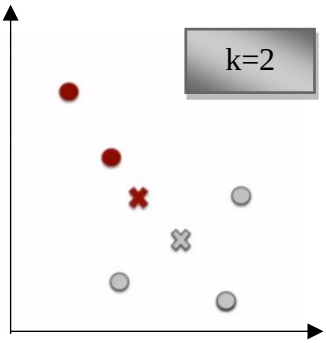

Slika 11. Prikaz nasumično dodijeljenih podatkovnih točaka klasteru

3. Izračunavanje centroida klastera. Oni su označeni „X“ na *Slici 12*. :



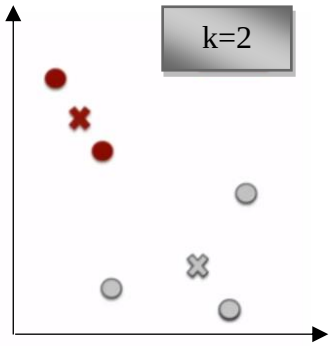

Slika 12. Prikaz izračunatih centroida klastera

4. Ponovo se vrši dodjela svake podatkovne točke, ali ovaj put isključivo najbližem, prethodno određenom središtu klastera.




Slika 13. Prikaz ponovljene podjele podatkovne točke najbližem od svih određenih centara klaster

5. Ponovo izračunavanje centroida za svaki od „k“ klastera

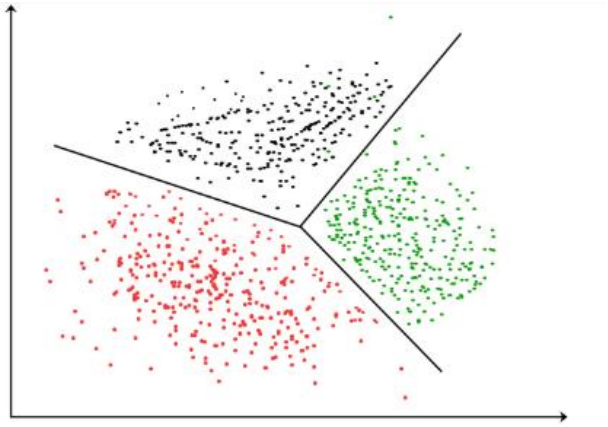



Slika 14. Prikaz ponovo izračunatih centroida klastera

Podatkovne točke dodjeljuju se najbližem centroidu i formira se klaster. Centroidi se zatim ažuriraju, a podatkovne točke preraspodjeljuju.

Zadnja dva koraka se ponavljaju dok se ne dođe do minimalnih promjena unutar kojih više poboljšanja nisu moguća, odnosno dok se postigne globalna optimala. Taj se proces nastavlja iterativno sve dok se lokacija centroida više ne promijeni. To je trenutak kada se više neće mijenjati položaj podatkovne točke između dva klastera za dva uzastopna ponavljanja.

Kreirani klasteri imaju tendenciju prema kružnom obliku radi korištenja srednje vrijednosti tijekom iteracije, što je potrebno uzimati u obzir pri odabiru ove metode grupiranja.[21]



Slika 15. Prikaz primjera grupiranja s većom količinom podataka

Neke praktične primjene K-means metode :

- profiliranje kupca - koristiti za karakterizaciju i otkrivanje korisničkih segmenata u marketinške svrhe,
- segmentacija tržišta,
- računalni vid,
- osiguranje: koristi se za priznavanje kupaca, njihovih namjera,identificiranje prijevara i izračun rizika ulaganja,
- internet tražilice,
- astronomija,
- biologija: koristiti se za razvrstavanje među različite vrste biljaka i životinja,
- knjižnice: koristi se u grupiranju različitih knjiga na temelju tema i informacija,
- geografija: učenjem područja pogođenih potresom možemo utvrditi opasne zone,
- itd.

K-menas konvergira u ograničenom broju iteracija. Budući da algoritam ponavlja funkciju čija je domena konačni skup, iteracijom se postiže konvergiracija. U usporedbi s drugim metodama grupiranja, tehnika grupiranja K-means brza je i učinkovita s obzirom na njezine formule. Teško je predvidjeti optimalan broj klastera, tj. vrijednost k. Za određivanje broja klastera mora pokrenuti algoritam grupiranja za raspon k vrijednosti, osim ako se unaprijed ne zna željeni broj klastera.

Metoda je široko korištena za analizu klastera, lako ju je razumijeti i vrlo se brzo nauči koristiti. Međutim, izračunata udaljenost nije dovoljno precizna u mnogim praktičnim primjenama, metoda ne može konkurirati najboljim metodama grupiranja. [18]

### *3.2.2. Odabir odgovarajućeg broja klastera*

Vrlo često podaci s kojima se radi imaju višestruke dimenzije koje znatno otežavaju preglednosti, a time i određivanje optimalnog broja klastera. Postoji više matematičkih metoda za određivanje broja klastera. Jedna od njih je kreiranje „WCSS“ grafa preko formule:

$$WSS = \sum_{n=1}^{\infty}(x_i + x_i)^2 \quad \text{(F1)}$$

WCSS se definira kao zbroj kvadratne udaljenosti između svakog člana klastera i njegovog centroida, a metoda je dobila ime po samom konačnom izgledu grafa.

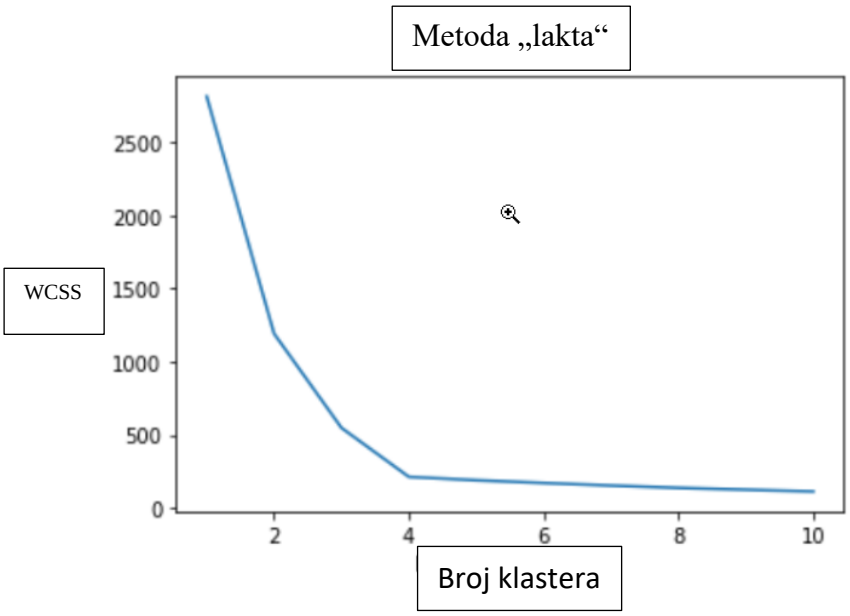
Za implementaciju WCSS grafa „metodom lakta“, potrebno je koristiti niže navedeni kôd u nekim od programskih jezika (Python):

```
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300,
n_init=10, random_state=0)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss)
```

```
plt.title('Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```

WCSS graf prikazuje odnos između podatkovnih točaka i zbroja klastera unutar klastera (WCSS), zatim odabiremo broj klastera kod kojih promjena WCSS-a počinje izravnati („metoda lakta“ – radi izgleda samog grafa). Nagla promjena na grafu koja ukazuje na optimalan broj klastera, formirat će se upravo na rednom broju točke zaključno s predloženim brojem klastera. U primjeru na *Slici 14.* izračunat je optimalan broj klastera k=3, sukladno ranije navedenom kôdu.

„*Elbow* metoda“, tj „metoda lakta“ promatra ukupni WSS kao funkciju broja klastera: potrebno je odabrati niz klastera tako da dodavanje drugog klastera ne poboljšava znatno ukupni WSS. Iz primjera na *Slici 14.* odgovarajući broj klastera bi bio 4. [18]




Slika 14. Primjer izračunavanja optimalnog broja klastera „metodom lakta“

Programski kôd K-means metode koja generira podatke funkcije “make_blobs“ iz modula „sklearn.datasets“, a parametar centra određuje broj klastera:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.datasets.samples_generator import make_blobs
from sklearn.cluster import Kmeans
X, y = make_blobs(n_samples=300, centers=4, cluster_std=0.60,
random_state=0)plt.scatter(X[:,0], X[:,1])

wcss = []for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300,
n_init=10, random_state=0)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss)
plt.title('Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()

kmeans = KMeans(n_clusters=4, init='k-means++', max_iter=300,
n_init=10, random_state=0)
pred_y = kmeans.fit_predict(X)plt.scatter(X[:,0], X[:,1])
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:,
1], s=300, c='red')
plt.show()
```

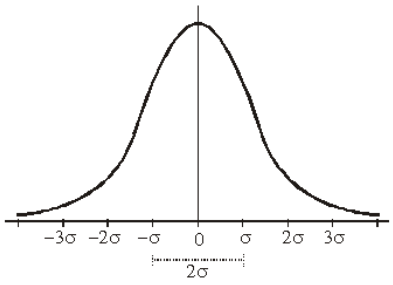
### *3.2.3. Gaussian Mixture metoda (GMM)*

Gaussian Mixture model smatra se vrlo moćnim algoritmom grupiranja. GMM pretpostavlja da su sve podatkovne točke nastale iz mješavine konačnog broja Gaussovih distribucija s nepoznatim parametrima. Svaka od tih distribucija predstavlja tzv. „grozd“, stoga GMM nastoji zajedno grupirati podatkovne točke koje pripadaju jednoj distribuciji. GMM može se smatrati generaliziranjem K-means metode.

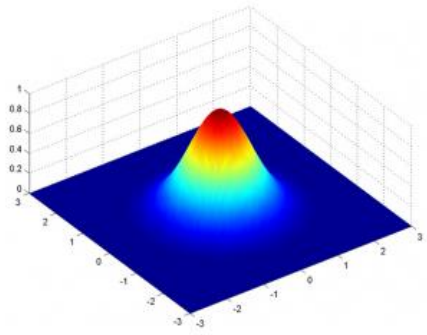
Ukoliko se pretpostavi da postoje tri Gaussove distribucije (GD1, GD2 i GD3) i da one imaju određenu srednju vrijednost (μ1, μ2, μ3) i varijancu (σ1, σ2, σ3), za dani skup podatkovnih točaka, GMM identificira vjerojatnost da svaka podatkovna točka pripada svakoj od tih distribucija. GMM su modeli temeljeni na vjerojatnosti i koriste postupke „mekog“ grupiranja raspodjelu točaka po klasterima. [24]

Na primjer, pretpostavka je da postoje tri skupine podataka označene trima bojama – plavom, zelenom i crvenom. Podatkovna točka označena crvenom bojom imat će vjerojatnost 1 da pripadne „crvenom“ klasteru, a za ostala dva klastera vjerojatnost će biti 0. No, ako podatkovna točka sadrži podatke koji ne pripadaju 100% niti jednom klasteru, kao npr. ljubičasta boja. Ta točka će imati vjerojatnost 0,5 za „plavi“ i 0,5 za „crveni“ klaster, a za „zeleni“ 0.

Gaussova distribucija može se shvatiti kao 3D krivulja u obliku zvona stvora prema uzoru na „normalnu Gaussovu razdiobu “, 2D krivulja u obliku zvona.




Slika 16. Krivulja normalne Gaussove razdiobe



Slika 17. 3D prikaz Gaussove krivulje

Funkcija gustoće vjerojatnosti jest:

$$f(x \mid \mu, \sum) = \frac{1}{\sqrt{2\pi|\sum|}} exp\left[-\frac{1}{2}(x-\mu)^t \sum^{-1}(x-\mu)\right] \quad \text{(F2)}$$

gdje je x ulazni vektor, μ je 2D srednji vektor, a Σ 2 koverzijska matrica 2 × 2. Kovarijancija definira oblik ove krivulje. Na taj način moguće je generirati vektor za d-dimenzije. Dakle, GMM sadrži x i μ kao vektore duljine d, a Σ bi bio kovarijantna matrica d x d. Za skup podataka s d značajkama GMM s „k" brojem raspodjela, gdje je k ekvivalent broju klastera, a svaka ima određenu matricu srednjeg vektora i varijance. Srednji vektor i varijacija određuju se *Expectation-Maximization* (EM) metodom.

EM je statistički algoritam koji pronalazi ispravne parametre modela, a osnova je mnogih algoritama. Ova metoda obično se koristi kada nedostaju podaci ili su nepotpuni. Takve varijable koje nedostaju nazivaju se latentne (skrivene) varijable. Teško je odrediti ispravne parametre modela kada postoje varijable koje nedostaju. Problematika leži u tome što latentnim varijablama nije poznata matica srednjih vrijednosti i varijacije.

Algoritam EM sastoji se od 2 koraka, E-korak i M-korak. [24]

E-korak pomoću postojećih podataka određuje optimalne vrijednosti varijabli koje nedostaju, a M-korak na temelju procijenjenih vrijednosti generiranih u E-koraku kreira potpune odgovarajuće parametre modela.

Za svaku točki $x_i$ potrebno je izračunati vrijednost pripadanja klaster/distribuciji $c_1, c_2, c_3, \ldots c_k$ preko omjera vjerojatnosti da $x_i$ pripada klasteru c i sume vjerojatnosti da $x_i$ pripada svim klasterima.

Nakon toga, ažuriraju se vrijednosti Π, μ i Σ. Nova gustoća definira se omjerom broja dodijeljenih klastera podatkovnoj točki i ukupnog broja klastera, preko sljedeće formule:

$$\Pi = \frac{Broj\ točaka\ dodjeljenih\ klasteru}{Ukupan\ broj\ klastera} \quad \text{(F3)}$$

Srednja vrijednost i matrica kovarijacije ažuriraju se na temelju vrijednosti dodijeljenih distribucijom, razmjerno s vrijednostima vjerojatnosti za tu podatkovnu točku. Dakle, podatkovna točka koja ima veću vjerojatnost da će biti dio te distribucije, bit će joj i dodijeljena.

Srednja vrijednost i matrica kovarijacije računaju se preko sljedećih formula:

$$\mu = \frac{1}{Broj\ točaka\ dodijeljenih\ klasteru} \sum_i r_{ic} x_i \quad \text{(F4)}$$

$$\sum_{ic} = \frac{1}{Broj\ točaka\ dodijeljenih\ klasteru} \sum_i r_{ic} (x_i - \mu_c)^T (x_i - \mu_c) \quad \text{(F5)}$$

Na temelju vrijednosti dobivenih iz opisanih koraka, izračunavaju se nove vjerojatnosti za svaku točku. Postupak se iterativno ponavlja i sukladno tome ažuriraju podatke. [25] [26]

*Sklearn.mixture* je programski paket koji omogućuje učenje i istovremeno korištenje kombinacija Gaussovih modela (podržane dijagonalne, sferne, vezane i pune kovarijantne matrice). Uzorkovanje i procjena dostupni su iz ulaznih podataka (podatkovnih točaka). Omogućeni su i objekti za određivanje odgovarajućeg broja klastera.

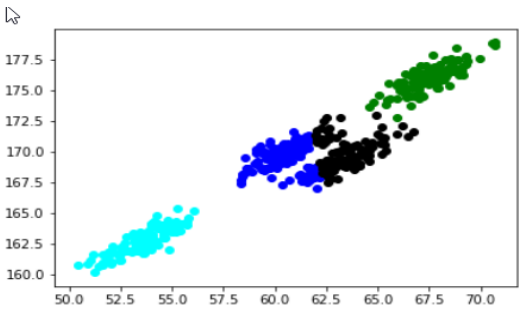
Primjer kôda:

```
import pandas as pd
data = pd.read_csv('Clustering_gmm.csv')

plt.figure(figsize=(7,7))
plt.scatter(data["Weight"],data["Height"])
plt.xlabel('Weight')
plt.ylabel('Height')
plt.title('Data Distribution')
plt.show()



# training gaussian mixture model
from sklearn.mixture import GaussianMixture
gmm = GaussianMixture(n_components=4)
gmm.fit(data)

#predictions from gmm
labels = gmm.predict(data)
frame = pd.DataFrame(data)
frame['cluster'] = labels
frame.columns = ['Weight', 'Height', 'cluster']

color=['blue','green','cyan', 'black']
for k in range(0,4):
    data = frame[frame["cluster"]==k]
    plt.scatter(data["Weight"],data["Height"],c=color[k])
plt.show()
```

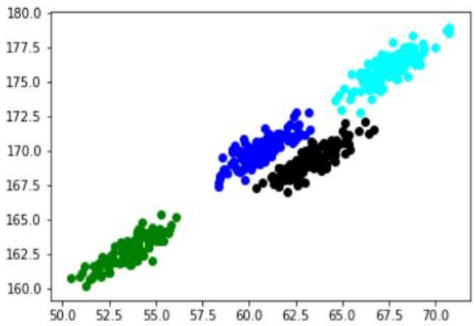
### *3.2.4. Usporedba K-menas i GMM metode*

Razlika između K-means i GMM metode je u tome što K-means uzima u obzir srednju vrijednost kako bi se izračunao centroid, a GMM uz srednju vrijednost uzima u obzir i varijacije podataka. Stoga je potreban drugačiji način dodjeljivanja klastera podatkovnim točkama. Umjesto korištenja modela temeljenog na daljini, GMM koristi model temeljen na distribuciji. [22]

Iz *Slike 18.* i *Slike 19.* vidljivo je kako se u praksi očituje razlika između ove dvije metode. K-means metodom dobiveni su klaster oblika elipse, iako za prikazani slučaj to nije praktično. GMM metodom došlo se do „točnije“ i prihvatljivije podjele u klastere oblika elipse, kako je vizualno i predvidljivo. Različite boje predstavljaju različitu raspodjelu točaka u rezultantne klastere, iako je ilustrativno vidljivo koja je grupacija „točnija“ u odnosu na sam položaj točaka. Za iste ulazne podatke, metode su dale različitu podjelu unutar klastera.




Slika 18. Primjer razultata grupiranja K-means metodom




Slika 19. Primjer rezultata grupiranja GMM metodom

# 4. OPIS ZADATKA

Zadatak je identificirati „*mobility*“ profile ćelija u mreži. Ovisno o prirodi i potrebama kretanja korisnika unutar mreže, svaka ćelija može imati drugačije vrijednosti parametara koji su za nju najpovoljniji. Težnja tome da svaka ćelija bude efektivno optimizirana rezultira boljim performansama mreže, što svaki operater nastoji pružiti svojim korisnicima. Uz pretpostavku da postoji konačan broj profila, te da se isti ne mogu grupirati jednostavnom geoprostornom analizom, cilj projekta je određivanje „*mobility*“ profila pojedinih ćelija u mreži.

Potrebno je pronaći „*high-mobility*“ ćelije, odnosno ćelije koje pokrivaju područja brzog prolaska korisnika kroz nju na području između dva grada. Analizirat će se područje rada mreže između dva europska grada udaljena oko 160 km, povezana autoputom, brzom cestom, lokalnim cestama i prugom. S obzirom na to da je to područje raznolikih geografskih i demografskih svojstava, ne može se jednostavnim metodama odrediti koje ćelije su usmjerene k brzim cestama i pruzi. Gradovi su povezani brzom cestom i prugom, ideja je izdvojiti „*high-mobility*“ ćelije koje pokrivaju to područje kako bi se lakše optimizirale (promjena parametara u svrhu boljeg rada). Analiza uključuje 2099 ćelija.

Potreba za ovim pojavila se kada se ustanovilo da korisnicima na tom području kretanja vrlo često „puca“ veza, iz razloga što se oni jednostavno prebrzo kreću kroz mrežu i ćelije ne stignu nesmetano obaviti primopredaju korisnika. Korisnici se kreću brzinama između 100 km/h i 200 km/h na autoputu, a u vlakovima između 200 km/h i 300 km/h. Zbog ćelijskih parametara se kretanje korisnika može drugačije odraziti na primopredaju korisnika između ćelija (eng. *handover*). Ono što je bitno jest posljedica kretanja korisnika na količinu primopredaje, a ne samo kretanje istog.

Na temelju ulaznih podataka i provedbom dvije metode strojnog učenja pokušat će se dobiti grupacija svih ćelija u 4 klastera. Nema potrebe da se traži grupacija u više klastera, jer je poznata priroda podataka koji se analiziraju. Iz znanja o mobilnim mrežama, baznim stanicama i ćelijama te optimizaciji istih, 4 klastera će biti dovoljna za olakšavanje promjene parametara mreže u svrhu njezinog boljeg rada.

Skup podataka koji se analizira jest vrijeme (s) korisnika provedeno u nekoj ćeliji.

## 4.1. Obrada podataka

U Tablici 1. prikazan je dio ulazni podaci. U lijevom stupcu nalaze se imena ćelija u anonimiziranom prikazu, a u desnom stupcu vrijeme koje je korisnik proveo u ćeliji. S obzirom na to da se radi o 1 048 576 uzoraka, u tablici su reprezentativno prikazani samo neki od njih. Podaci su prikupljeni na radni dan u siječnju 2019. godine, između 6:00 h i 8:00 h.

| Ćelija | Vrijeme (s) |
|---|---|
| cell-1 | 7.863 |
| cell-2 | 11.923 |
| cell-1 | 13.214 |
| cell-2 | 7.133 |
| cell-3 | 3.906 |
| cell-1 | 2.35 |
| cell-4 | 1.581 |
| cell-1 | 5.379 |
| cell-5 | 27.947 |
| cell-3 | 5.932 |
| cell-2 | 20.18 |
| cell-4 | 2.264 |
| cell-1 | 11.268 |
| cell-2 | 1.968 |
| cell-4 | 2.302 |
| cell-5 | 3.004 |
| cell-1 | 6.43 |
| cell-5 | 12.383 |
| cell-1 | 7.104 |
| cell-4 | 7.488 |
| cell-5 | 0.384 |
| cell-2 | 11.837 |
| cell-1 | 1.378 |
| cell-2 | 2.123 |
| cell-5 | 3.855 |
| cell-1 | 15.558 |
| cell-2 | 0.403 |
| cell-2 | 9.659 |
| cell-1 | 5.924 |
| cell-1 | 1.056 |
| cell-2 | 1.954 |
| cell-2 | 2.553 |
| cell-0 | 49.384 |
| cell-1 | 2.399 |
| cell-1 | 9.261 |
| cell-1 | 10.718 |
| cell-5 | 0.624 |
| cell-1 | 1.447 |
| cell-0 | 7.357 |
| cell-0 | 2.5 |
| cell-1 | 2.01 |
| cell-0 | 1.205 |
| cell-5 | 2.653 |
| cell-5 | 20.536 |
| cell-1 | 5.839 |

| Ćelija | Vrijeme (s) |
|---|---|
| cell-1 | 2.249 |
| cell-5 | 3.276 |
| cell-1 | 2.089 |
| cell-5 | 0.321 |
| cell-0 | 0.679 |
| cell-4 | 0.542 |
| cell-1 | 2.026 |
| cell-4 | 3.968 |
| cell-1 | 15.579 |
| cell-1 | 8.8 |
| cell-1 | 1.462 |
| cell-4 | 26.947 |
| cell-5 | 0.697 |
| cell-5 | 10.668 |
| cell-5 | 4.243 |
| cell-1 | 5.409 |
| cell-5 | 4.492 |
| cell-1 | 9.823 |
| cell-0 | 3.763 |
| cell-1 | 2.224 |
| cell-3 | 1.787 |
| cell-2 | 12.302 |
| cell-1 | 24.242 |
| cell-1 | 15.175 |
| cell-5 | 2.847 |
| cell-1 | 2.899 |
| cell-3 | 3.841 |
| cell-5 | 7.73 |
| cell-1 | 3.121 |
| cell-0 | 8.574 |
| cell-1 | 1.499 |
| cell-4 | 6.847 |
| cell-1 | 1.544 |
| cell-4 | 2.116 |
| cell-5 | 27.029 |
| cell-4 | 2.677 |
| cell-5 | 1.932 |
| cell-1 | 6.599 |
| cell-5 | 4.81 |
| cell-3 | 4.263 |
| cell-5 | 11.781 |
| cell-0 | 9.735 |
| cell-5 | 1.604 |
| cell-2 | 1.998 |
| cell-1 | 1.049 |

| Ćelija | Vrijeme (s) |
|---|---|
| cell-4 | 1.34 |
| cell-1 | 11.884 |
| cell-5 | 5.323 |
| cell-4 | 2.117 |
| cell-1 | 5.643 |
| cell-2 | 2.869 |
| cell-3 | 29.887 |
| cell-1 | 13.617 |
| cell-5 | 13.057 |
| cell-0 | 9.476 |
| cell-3 | 3.485 |
| cell-5 | 3.423 |
| cell-2 | 3.294 |
| cell-4 | 14.122 |
| cell-2 | 6.517 |
| cell-0 | 1.116 |
| cell-5 | 2.877 |
| cell-2 | 8.194 |
| cell-1 | 3.237 |
| cell-2 | 1.535 |
| cell-5 | 1.687 |
| cell-1 | 3.214 |
| cell-1 | 3.568 |
| cell-0 | 4.448 |
| cell-1 | 5.778 |
| cell-0 | 1.933 |
| cell-1 | 18.31 |
| cell-0 | 1.355 |
| cell-3 | 3.367 |
| cell-1 | 1.321 |
| cell-3 | 1.764 |
| cell-0 | 4.219 |
| cell-1 | 20.748 |
| cell-1 | 5.473 |
| cell-3 | 5.751 |
| cell-1 | 4.35 |
| cell-1 | 4.215 |
| cell-4 | 10.758 |
| cell-5 | 3.782 |
| cell-3 | 3.596 |
| cell-1 | 2.02 |
| cell-4 | 1.599 |
| cell-1 | 1.203 |
| cell-3 | 2.57 |
| cell-1 | 6.107 |

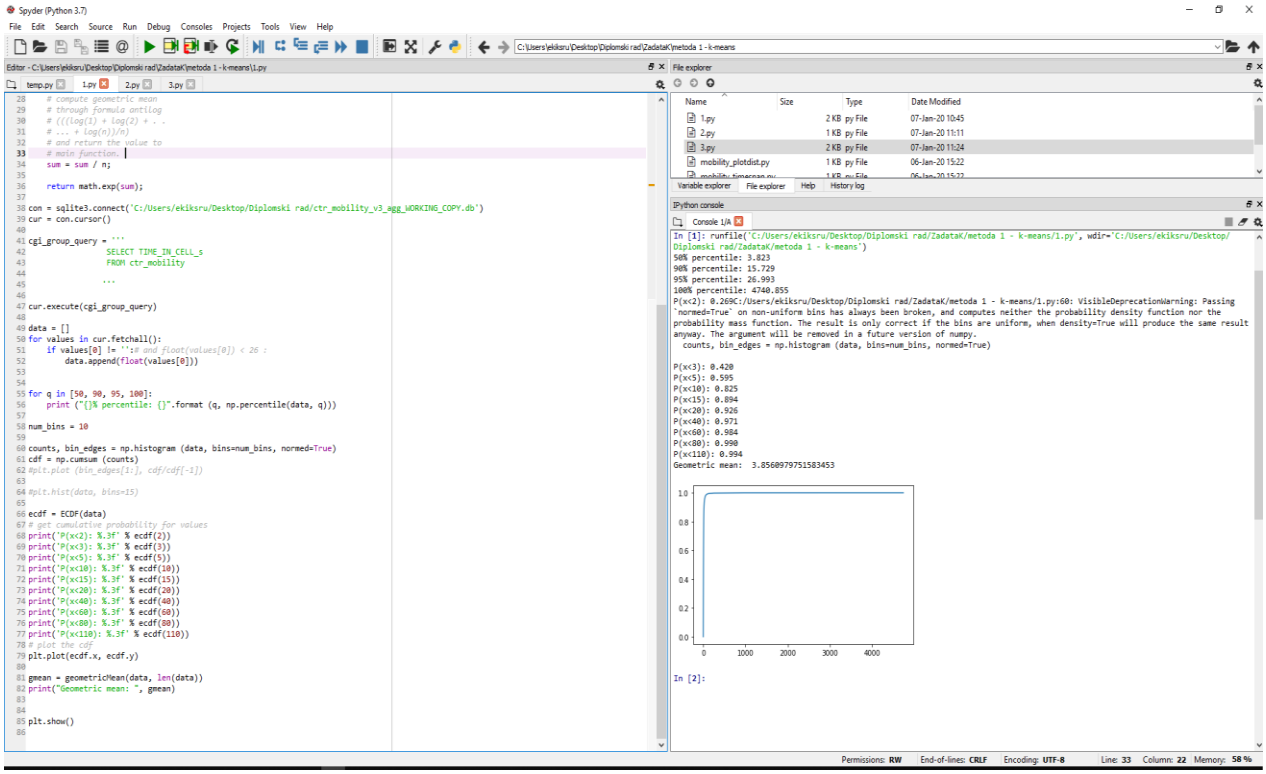
| | |
|---|---|
| cell-5 | 5.094 |
| cell-0 | 1.113 |
| cell-1 | 9.424 |
| cell-4 | 1.34 |
| cell-1 | 11.884 |
| cell-5 | 5.323 |
| cell-4 | 2.117 |
| cell-1 | 5.643 |
| cell-2 | 2.869 |
| cell-3 | 29.887 |
| cell-1 | 13.617 |
| cell-5 | 13.057 |
| cell-0 | 9.476 |
| cell-3 | 3.485 |
| cell-5 | 3.423 |
| cell-2 | 3.294 |
| cell-4 | 14.122 |
| cell-2 | 6.517 |
| cell-0 | 1.116 |
| cell-5 | 2.697 |
| cell-4 | 7.221 |
| cell-4 | 0.705 |
| cell-1 | 11.114 |
| cell-1 | 1.645 |
| cell-0 | 2.229 |
| cell-1 | 9.25 |
| cell-3 | 5.223 |
| cell-4 | 0.774 |
| cell-5 | 0.257 |
| cell-1 | 0.816 |
| cell-3 | 6.723 |
| cell-4 | 2.937 |
| cell-5 | 7.159 |
| cell-1 | 2.577 |
| cell-4 | 0.673 |
| cell-1 | 1.932 |
| cell-1 | 7.473 |
| cell-4 | 3.276 |
| cell-2 | 1.631 |
| cell-2 | 4.329 |
| cell-5 | 0.97 |
| cell-5 | 4.598 |
| cell-1 | 2.941 |
| cell-0 | 5.426 |
| cell-0 | 6.284 |
| cell-1 | 3.96 |
| cell-4 | 12.775 |
| cell-4 | 11.402 |

| | |
|---|---|
| cell-2 | 8.364 |
| cell-3 | 3.338 |
| cell-5 | 2.877 |
| cell-2 | 8.194 |
| cell-1 | 3.237 |
| cell-2 | 1.535 |
| cell-5 | 1.687 |
| cell-1 | 3.214 |
| cell-1 | 3.568 |
| cell-0 | 4.448 |
| cell-1 | 5.778 |
| cell-0 | 1.933 |
| cell-1 | 18.31 |
| cell-0 | 1.355 |
| cell-3 | 3.367 |
| cell-1 | 1.321 |
| cell-3 | 1.764 |
| cell-0 | 4.219 |
| cell-1 | 20.748 |
| cell-1 | 5.473 |
| cell-3 | 5.751 |
| cell-1 | 4.35 |
| cell-1 | 4.215 |
| cell-4 | 10.758 |
| cell-5 | 3.782 |
| cell-3 | 3.596 |
| cell-1 | 2.02 |
| cell-2 | 0.257 |
| cell-4 | 1.599 |
| cell-1 | 1.203 |
| cell-3 | 4.238 |
| cell-3 | 2.57 |
| cell-1 | 6.107 |
| cell-1 | 0.537 |
| cell-5 | 0.322 |
| cell-2 | 2.192 |
| cell-1 | 4.786 |
| cell-2 | 2.98 |
| cell-1 | 6.832 |
| cell-1 | 6.2 |
| cell-4 | 0.901 |
| cell-4 | 1.037 |
| cell-5 | 9.966 |
| cell-1 | 16.221 |
| cell-5 | 3.765 |
| cell-4 | 0.658 |
| cell-1 | 3.134 |
| cell-2 | 0.285 |

| | |
|---|---|
| cell-1 | 0.537 |
| cell-5 | 0.322 |
| cell-1 | 0.342 |
| cell-1 | 21.953 |
| cell-1 | 4.474 |
| cell-1 | 2.486 |
| cell-4 | 7.824 |
| cell-5 | 3.161 |
| cell-1 | 12.284 |
| cell-1 | 0.622 |
| cell-1 | 1.685 |
| cell-0 | 3.419 |
| cell-5 | 2.063 |
| cell-2 | 0.6 |
| cell-1 | 3.526 |
| cell-2 | 5.703 |
| cell-0 | 7.993 |
| cell-1 | 1.692 |
| cell-4 | 3.333 |
| cell-4 | 0.879 |
| cell-1 | 5.745 |
| cell-4 | 3.07 |
| cell-1 | 6.666 |
| cell-4 | 1.225 |
| cell-2 | 4.057 |
| cell-3 | 6.027 |
| cell-0 | 5.996 |
| cell-0 | 5.506 |
| cell-5 | 2.056 |
| cell-5 | 6.302 |
| cell-0 | 30.246 |
| cell-5 | 6.259 |
| cell-0 | 2.331 |
| cell-2 | 0.298 |
| cell-5 | 7.04 |
| cell-2 | 16.461 |
| cell-4 | 0.718 |
| cell-3 | 4.424 |
| cell-0 | 1.744 |
| cell-3 | 7.74 |
| cell-1 | 4.569 |
| cell-3 | 4.213 |
| cell-4 | 12.597 |
| cell-1 | 5.102 |
| cell-1 | 11.335 |
| cell-1 | 5.134 |
| cell-5 | 19.59 |
| cell-5 | 25.038 |

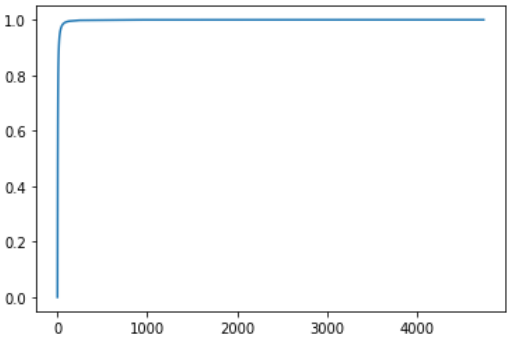
Tablica 1. Reprezentativni uzorak ulaznih podataka

Ulazni podaci prikazani su u Tablici 1.. Posljedica kratkog zadržavanja unutar područja pokrivenosti ćelije (iz nekog razloga), jest povećanje količine signalizacije u mreži. Stoga je prvi korak upravo određivanje razumnog raspona vremena. [Prilog I]




Slika 20. Određivanje raspona vremena

Statistika ulaznih podataka pokazala je da se više od 92% korisnika kretalo u ćeliji manje od 20 sekundi, što je vidljivo u *Slici 21.* [Prilog I]

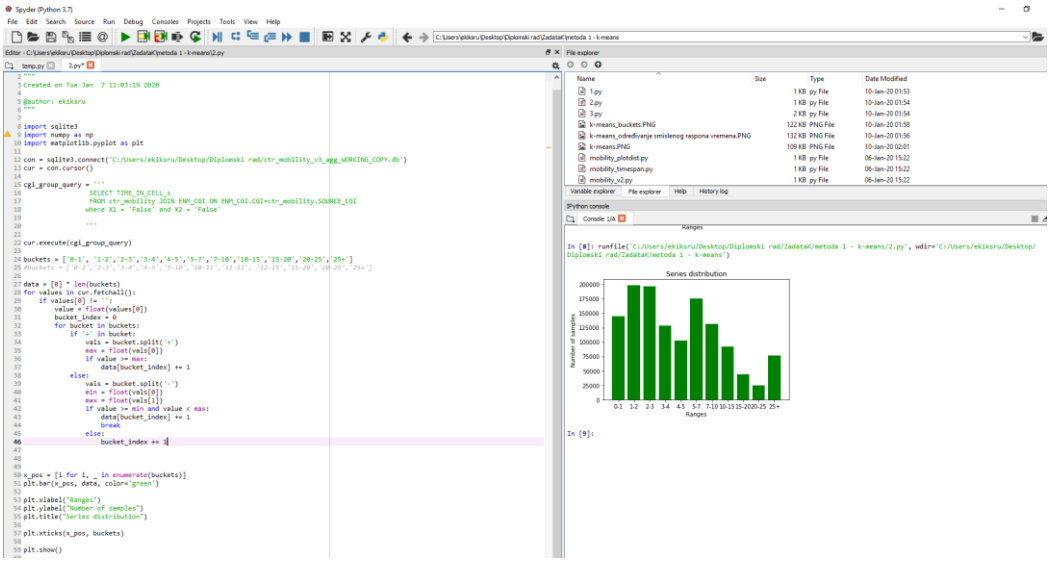



Slika 21. CDF graf

Iz *Slike 21.* vidi se da u području pokrivanja ćelije postoje korisnici koji se u njoj nalaze po 3000, 4000 sekundi. Zaključak je da se radi o korisnicima koji se ne kreću unutar istraživanog vremena.

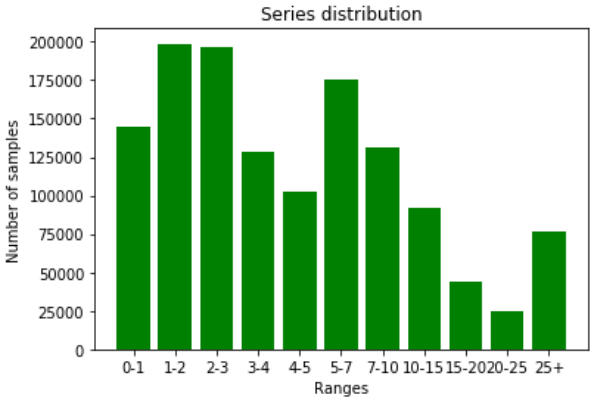
S obzirom na to da je 20 sekundi vrijeme koje većina korisnika provedu na području određene ćelije, raspon podataka odabran je logički, s obzirom na poznavanje prirode istih, a on je:

- *['0-1', '1-2','2-3','3-4','4-5','5-7','7-10','10-15','15-20','20-25','25+']*



Slika 22. Provjera smislenosti zadanog raspona

Na *Slici 23.* vidljivo je na koji način uzorci padaju u rasponima [Prilog II].

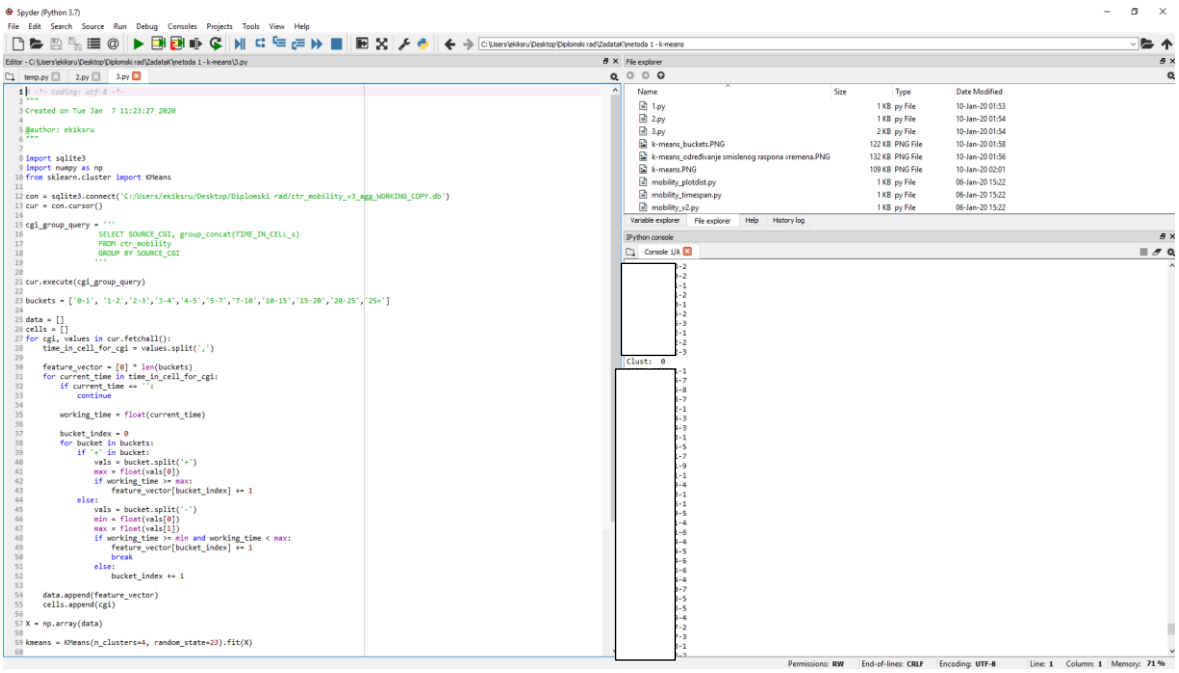



Slika 23. Raspodjela korisnika unutar zadanih vremenskih raspona

Iz histograma je vidljivo kako se najveći broj uzoraka ( u konkretnom zadatku – korisnika) kreće u rasponu od 1-2, 2-3 i 5-7 sekundi. Određivanjem logičnog raspona i potvrdom da je većina korisnika unutar njega, pristupa se trećem koraku analize.

## 4.2. Analiza K-means metodom

S obzirom na poznavanje opisa i značenja ulaznih podataka, unaprijed će se odabrati broj klastera. Četiri klastera su dovoljna grupacija prema zahtjevima optimizacijskog tima. [Prilog III]



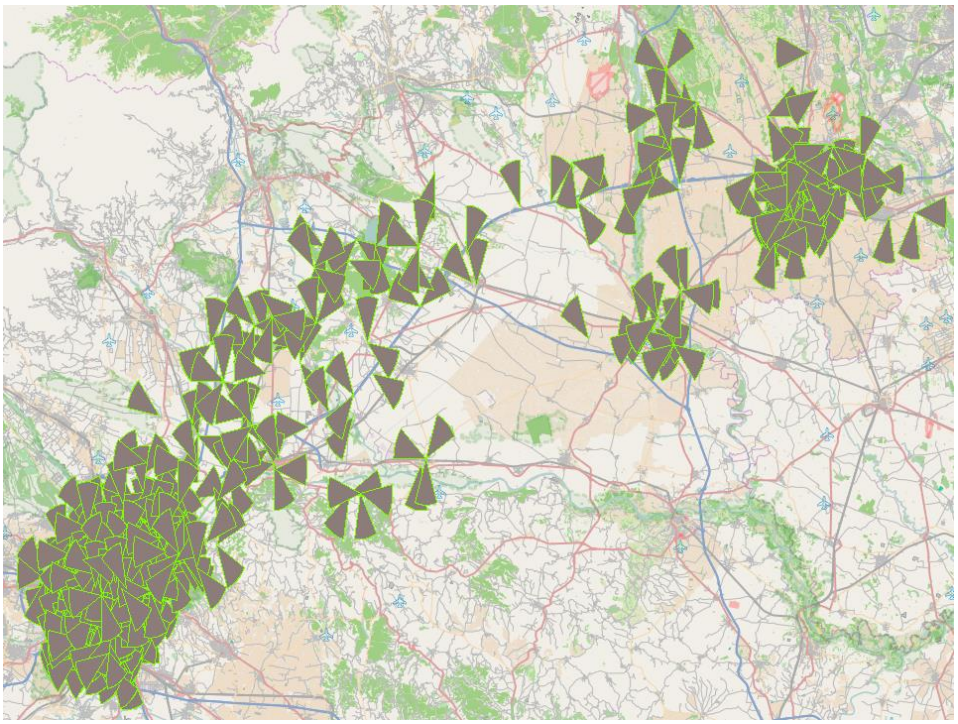
```
# -*- coding: utf-8 -*-
"""
Created on Tue Jan  7 11:23:27 2020

@author: ekiksru
"""

import sqlite3
import numpy as np
from sklearn.cluster import KMeans

con = sqlite3.connect('C:/Users/ekiksru/Desktop/Diplomski rad/ctr_mobility_v3_agg_WORKING_COPY.db')
cur = con.cursor()

cgi_group_query = '''
                  SELECT SOURCE_CGI, group_concat(TIME_IN_CELL_s)
                  FROM ctr_mobility
                  GROUP BY SOURCE_CGI
                  '''


cur.execute(cgi_group_query)

buckets = ['0-1', '1-2','2-3','3-4','4-5','5-7','7-10','10-15','15-20','20-25','25+']

data = []
cells = []
for cgi, values in cur.fetchall():
    time_in_cell_for_cgi = values.split(',')

    feature_vector = [0] * len(buckets)
    for current_time in time_in_cell_for_cgi:
        if current_time == "":
            continue

        working_time = float(current_time)

        bucket_index = 0
        for bucket in buckets:
            if '+' in bucket:
                vals = bucket.split('+')
                max = float(vals[0])
                if working_time >= max:
                    feature_vector[bucket_index] += 1
            else:
                vals = bucket.split('-')
                min = float(vals[0])
                max = float(vals[1])
                if working_time >= min and working_time < max:
                    feature_vector[bucket_index] += 1
                    break
                else:
                    bucket_index += 1

    data.append(feature_vector)
    cells.append(cgi)

X = np.array(data)

kmeans = KMeans(n_clusters=4, random_state=23).fit(X)
```

Slika 24. Određivanje klastera K-means metodom

### *4.2.1. Rezultati analize K-means metode*

Rezultati K-means metode i podjela ćelija u 4 tražena klastera prikazani su na *Slici 25.*, *Slici 26.*, *Slici 27.* i *Slici 28.*

Dobivene ćelije unutar klastera prikazane na karti:



Slika 25 . Klaster – 0 / K-means metoda

Slika 26. Klaster – 1 / K-means metoda

Slika 27. Klaster – 2 / K-means metoda

Slika 28. Klaster – 3 / K-means metoda

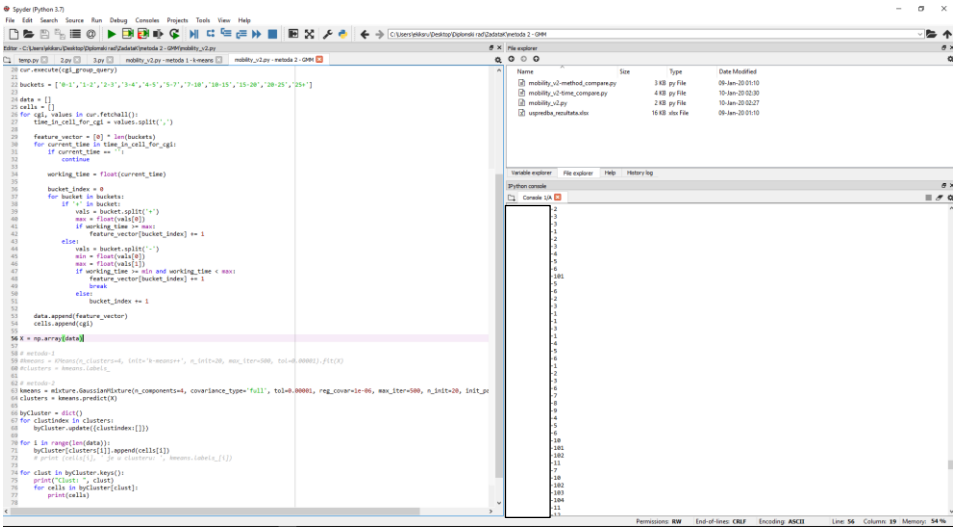
Na *Slici 25*. i *Slici 27*. vidljivo je kako se klasificiranje „*high-mobility*“ ćelije nalaze duž autoputa i pruge. Vizualno se prepoznaje upravo ta putanja.

Ostale grupacije mogu se objasniti na sljedeći način. Vidljivo je koliko je veliki broj „*high-mobility*“ ćelija u svakom klasteru pozicionirano u samim gradovima. Obzirom na ideju ovog zadatka, taj podatak se ignorira i objašnjava na sljedeći način. Bazne stanice u gradovima su znatno bliže pozicionirane jedna drugoj nego što je to duž autoputa, pruga i manje naseljenih područja. Radi veće koncentracije korisnika u gradovima i veće opterećenosti mreže, one mogu biti postavljene na udaljenost od 200 m do 400 m, a više im se preklapaju područja pokrivenosti. Korisnici kroz grad ne moraju se nužno brzo kretati da bi brzo prelazili iz jedne ćelije u drugu. Dovoljno je kretati se dozvoljenom brzinom u automobilu i kroz par kilometara vrlo brzo mijenjati područja pokrivenosti ćelija. Također, istu situaciju je moguće postići bržom šetnjom kroz centar grada gdje su bazne stanice gusto raspoređene, ili vožnjom biciklom ili nekim drugim prijevoznim sredstvom.

## 4.3. Analiza Gaussian Mixture metodom

Isti podaci odrađeni su i GMM algoritmom. Prva dva koraka se ne razlikuju u odnosu na K-means metodu, s obzirom na to da oni nisu dio metode grupiranja, nego priprema podataka za provedbu samih metoda.

Zadnji korak analize, korištenje GMM metode dao je podjelu ćelija po klasterima prikazanu na *Slici 29*. [Prilog IV]

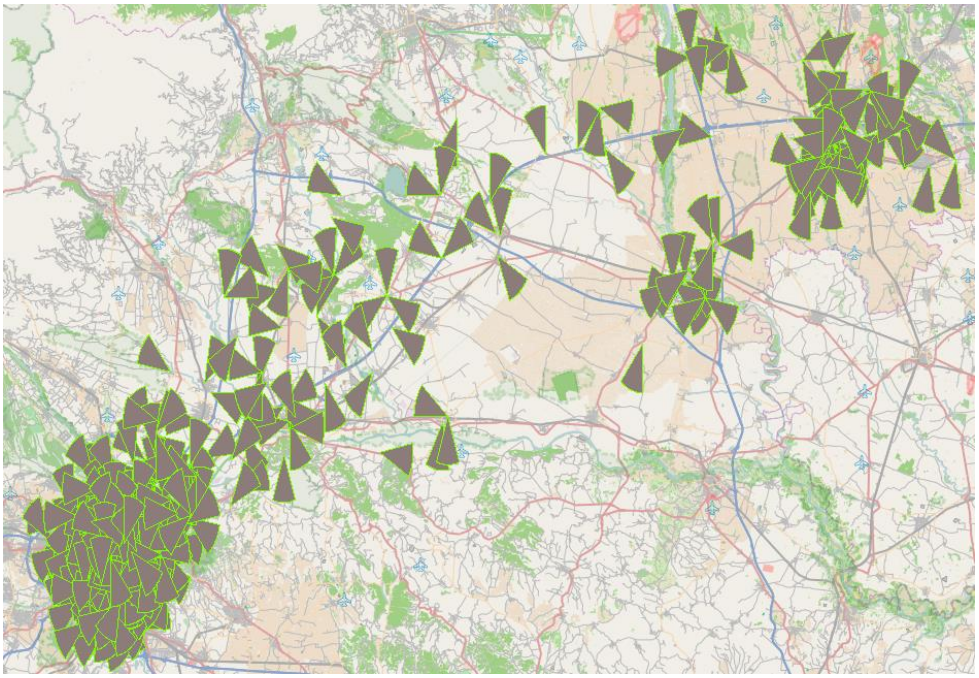


Slika 29. Grupacija klastera GMM metodom

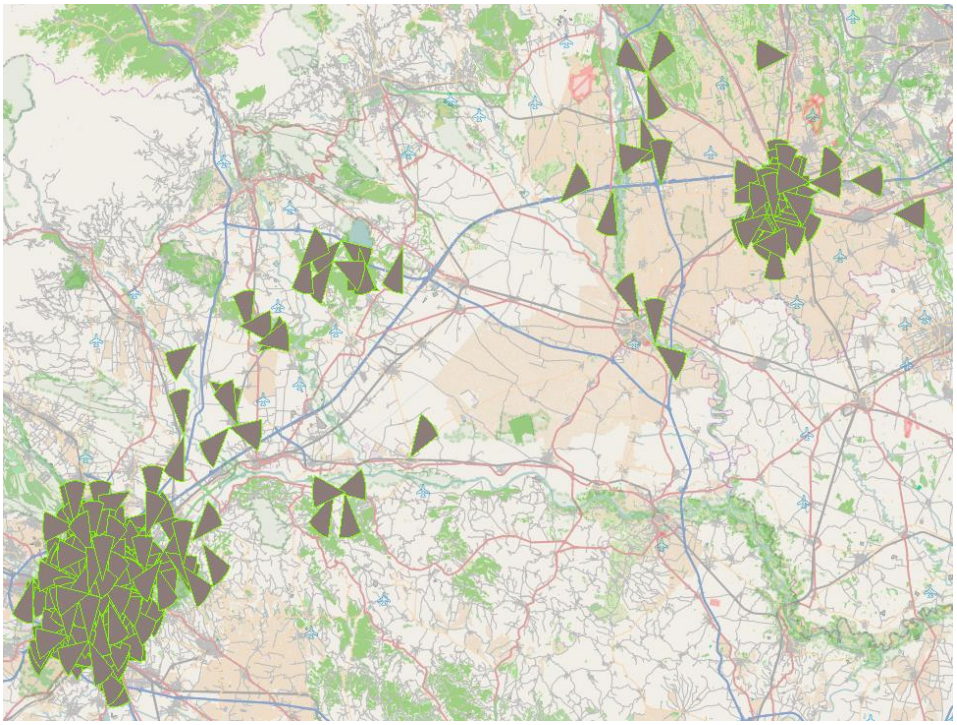
### *4.3.1. Rezultati analize GMM metode*

Geografski prikaz dobivenih klastera slijedi na *Slici 30*., *Slici 31*., *Slici 32*. i *Slici 33*.

Gusta raspodjela klasificiranih „*high-mobility*“ ćelija unutar samih gradova dogodila se iz istog razloga kao i u K-means metodi. Priroda ulaznih podataka objašnjava ovu pojavu. Prema *Slici 32*. može se primijetiti klasificiranje ćelija duž prometnica, što je i bio cilj analize.

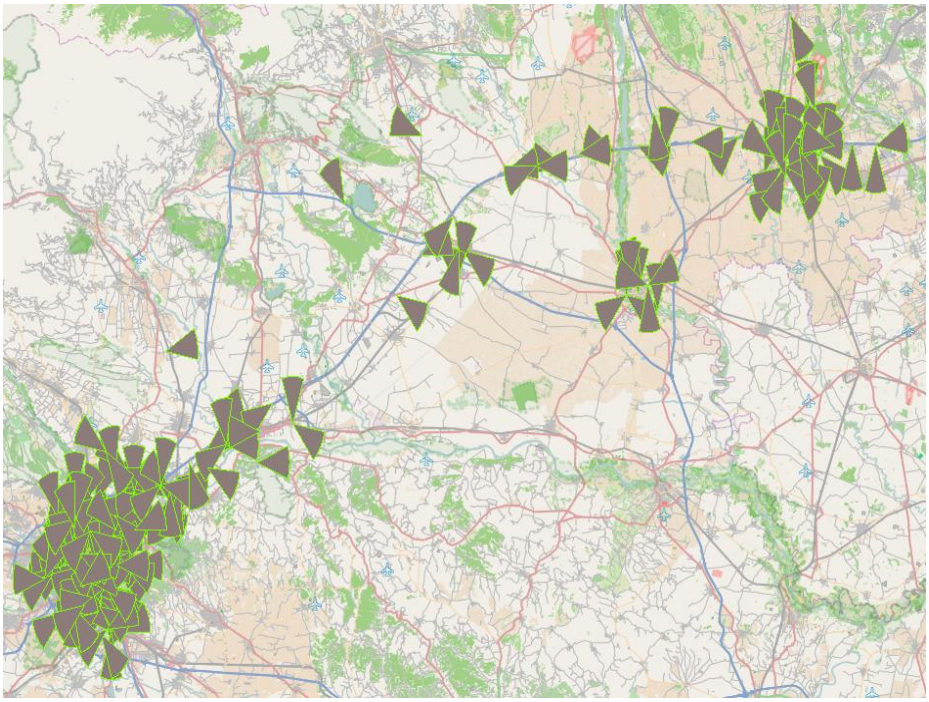


Slika 30. Klaster – 0 / GMM metoda



Slika 31. Klaster – 1 / GMM metoda

Slika 32. Klaster – 2 / GMM metoda



Slika 33. Klaster – 3 / GMM metoda

## 4.4. Usporedba rezultata K-means i GMM metode

Obje metode su dale isti rezultat, ali ne i istu geografsku raspodjelu, što se može objasniti činjenicom da su broj klastera i maksimalan broj iteracija postavljeni isto. Usporedbom rezultata prikazanom u Tablici 2., vidljivo je da su obje jednak broj ćelija raspodijelile u zadane klastere. Tablica 2. prikazuje broj ćelija koje su svrstane u istu grupaciju klastera različitim metodama, što znači da su obje metode jednak broj ćelija svrstale u zadana 4 klastera. [Prilog IV]

| **K-menas klaster / GMM klaster** | GMM-3 | GMM-1 | GMM-0 | GMM-2 |
|---|---|---|---|---|
| K-means-3 | 624 | 0 | 0 | 0 |
| K-means-1 | 0 | 1152 | 0 | 0 |
| K-means-0 | 0 | | 281 | 0 |
| K-means-2 | 0 | 0 | 0 | 42 |

Tablica 2. Usporedba rezultata različitih metoda

## 4.5. Usporedba rezultata s neovisnim mjerenjima

Za provjeru sezonalnosti i ponovljivost rezultata, provedena je analiza istih ćelija na istom području, ali u drugo vrijeme. Ulazni podaci prikazani u Tablici 3. sadrže periode korisnika unutar ćelije izražena u sekundama, za isti dan, u razdoblju od 16:00 h do 18:00 h.

| Ćelija | Vrijeme (s) |
|---|---|
| cell-1 | 1.66 |
| cell-2 | 8.712 |
| cell-2 | 27 |
| cell-3 | 8.815 |
| cell-2 | 2.476 |
| cell-1 | 1.921 |
| cell-0 | 2.906 |
| cell-1 | 1.877 |
| cell-1 | 1.91 |
| cell-3 | 12.497 |
| cell-1 | 5.016 |
| cell-1 | 3.626 |
| cell-1 | 5.913 |
| cell-3 | 5.608 |
| cell-3 | 10.773 |
| cell-2 | 10.872 |
| cell-1 | 10.624 |
| cell-1 | 9.36 |
| cell-4 | 5.002 |
| cell-0 | 3.477 |
| cell-1 | 1.482 |
| cell-0 | 1.198 |
| cell-1 | 9.915 |
| cell-4 | 6.847 |
| cell-0 | 2.496 |
| cell-5 | 1.578 |
| cell-4 | 34.693 |
| cell-1 | 6.924 |
| cell-1 | 5.447 |
| cell-1 | 1.205 |
| cell-0 | 0.92 |
| cell-2 | 2.516 |
| cell-3 | 4.434 |
| cell-5 | 2.821 |

| Ćelija | Vrijeme (s) |
|---|---|
| cell-347 | 6.284 |
| cell-351 | 6.55 |
| cell-349 | 1.13 |
| cell-349 | 0.684 |
| cell-349 | 0.77 |
| cell-352 | 9.389 |
| cell-350 | 21.779 |
| cell-352 | 0.76 |
| cell-349 | 1.323 |
| cell-349 | 14.109 |
| cell-349 | 3.112 |
| cell-347 | 1.735 |
| cell-348 | 0.347 |
| cell-352 | 3.925 |
| cell-352 | 0.767 |
| cell-352 | 22.889 |
| cell-352 | 42.8 |
| cell-351 | 0.869 |
| cell-349 | 0.949 |
| cell-347 | 1.191 |
| cell-350 | 0.432 |
| cell-352 | 1.854 |
| cell-352 | 6.402 |
| cell-349 | 5.981 |
| cell-352 | 8.204 |
| cell-352 | 5.43 |
| cell-352 | 2.074 |
| cell-351 | 5.029 |
| cell-351 | 1.628 |
| cell-349 | 5.519 |
| cell-348 | 1.171 |
| cell-352 | 1.615 |
| cell-350 | 1.205 |
| cell-352 | 12.111 |

| Ćelija | Vrijeme (s) |
|---|---|
| cell-1148 | 0.546 |
| cell-1142 | 20.083 |
| cell-1140 | 1.626 |
| cell-1145 | 5.467 |
| cell-1142 | 6.687 |
| cell-1144 | 16.243 |
| cell-1146 | 2.408 |
| cell-1142 | 22.879 |
| cell-1144 | 8.976 |
| cell-1143 | 5.903 |
| cell-1148 | 4.942 |
| cell-1145 | 1.477 |
| cell-1147 | 19.454 |
| cell-1142 | 57.013 |
| cell-1145 | 4.674 |
| cell-1145 | 18.883 |
| cell-1145 | 14.257 |
| cell-1145 | 1.79 |
| cell-1148 | 3.783 |
| cell-1146 | 8.879 |
| cell-1148 | 0.902 |
| cell-1144 | 6.95 |
| cell-1144 | 2.235 |
| cell-1141 | 3.867 |
| cell-1146 | 7.421 |
| cell-1145 | 9.182 |
| cell-1144 | 2.712 |
| cell-1146 | 7.679 |
| cell-1147 | 7.015 |
| cell-1142 | 2.905 |
| cell-1147 | 15.727 |
| cell-1146 | 8.022 |
| cell-1146 | 1.268 |
| cell-1145 | 3.327 |

| | | | | | |
|---|---|---|---|---|---|
| cell-0 | 21.729 | cell-349 | 0.977 | cell-1144 | 1.813 |
| cell-4 | 0.459 | cell-352 | 4.332 | cell-1142 | 1.577 |
| cell-3 | 12.362 | cell-351 | 3.886 | cell-1145 | 10.278 |
| cell-5 | 2.689 | cell-349 | 1.271 | cell-1147 | 23.541 |
| cell-5 | 0.753 | cell-350 | 2.086 | cell-1144 | 1.083 |
| cell-3 | 4.693 | cell-348 | 16.419 | cell-1147 | 4.959 |
| cell-4 | 1.17 | cell-351 | 11.654 | cell-1142 | 2.575 |
| cell-1 | 0.715 | cell-351 | 1.815 | cell-1142 | 2.317 |
| cell-3 | 2.032 | cell-349 | 7.827 | cell-1140 | 59.875 |
| cell-3 | 4.702 | cell-347 | 13.926 | cell-1146 | 1.937 |
| cell-3 | 2.492 | cell-347 | 2.119 | cell-1147 | 1.827 |
| cell-1 | 5.068 | cell-348 | 0.708 | cell-1144 | 3.62 |
| cell-0 | 15.303 | cell-347 | 4.499 | cell-1146 | 2.113 |
| cell-3 | 5.89 | cell-352 | 2.789 | cell-1145 | 8.757 |
| cell-1 | 2.728 | cell-352 | 16.571 | cell-1146 | 2.328 |
| cell-3 | 2.227 | cell-352 | 14.795 | cell-1144 | 2.001 |
| cell-3 | 17.087 | cell-347 | 6.739 | cell-1148 | 0.916 |
| cell-3 | 1.964 | cell-347 | 7.802 | cell-1145 | 6.363 |
| cell-5 | 13.996 | cell-352 | 4.72 | cell-1145 | 30.47 |
| cell-3 | 9.68 | cell-352 | 2.251 | cell-1146 | 5.67 |
| cell-0 | 3.71 | cell-350 | 7.72 | cell-1144 | 10.981 |
| cell-0 | 1.593 | cell-350 | 3.994 | cell-1148 | 4.262 |
| cell-3 | 3.99 | cell-351 | 57.864 | cell-1144 | 3.077 |
| cell-3 | 11.199 | cell-348 | 17.917 | cell-1146 | 4.047 |
| cell-3 | 1.068 | cell-351 | 63.91 | cell-1144 | 1.133 |
| cell-1 | 3.537 | cell-351 | 9.803 | cell-1145 | 3.852 |
| cell-0 | 4.263 | cell-349 | 1.888 | cell-1146 | 5.992 |
| cell-3 | 0.845 | cell-351 | 3.233 | cell-1145 | 2.252 |
| cell-3 | 11.392 | cell-349 | 0.585 | cell-1140 | 3.831 |
| cell-1 | 132.395 | cell-352 | 5.562 | cell-1147 | 10.695 |
| cell-5 | 1.048 | cell-348 | 5.547 | cell-1144 | 4.451 |
| cell-4 | 8.938 | cell-350 | 7.916 | cell-1146 | 10.896 |
| cell-0 | 9.071 | cell-349 | 2.276 | cell-1141 | 1.98 |
| cell-4 | 17.52 | cell-351 | 2.743 | cell-1151 | 6.967 |
| cell-1 | 9.198 | cell-348 | 1.265 | cell-1140 | 0.633 |
| cell-1 | 2.353 | cell-348 | 6.908 | cell-1146 | 1.573 |
| cell-1 | 1.39 | cell-349 | 2.961 | cell-1143 | 17.149 |
| cell-1 | 72.305 | cell-349 | 0.753 | cell-1142 | 2.332 |
| cell-0 | 92.48 | cell-349 | 1.779 | cell-1148 | 1.386 |
| cell-1 | 1.793 | cell-347 | 2.717 | cell-1147 | 1.777 |

Tablica 3. Reprezentativni uzorci popodnevnih mjerenja

Usporedba jutarnjih i popodnevnih analiza prikazana je u Tablici 4. [Prilog IV] :

| | UJUTRO-0 | UJUTRO-3 | UJUTRO-2 | UJUTRO-1 |
|---|---|---|---|---|
| POPODNE-1 | 497 | 52 | 98 | 0 |
| POPODNE-2 | 74 | 255 | 3 | 9 |
| POPODNE-3 | 57 | 2 | 972 | 0 |
| POPODNE-0 | 1 | 7 | 0 | 46 |

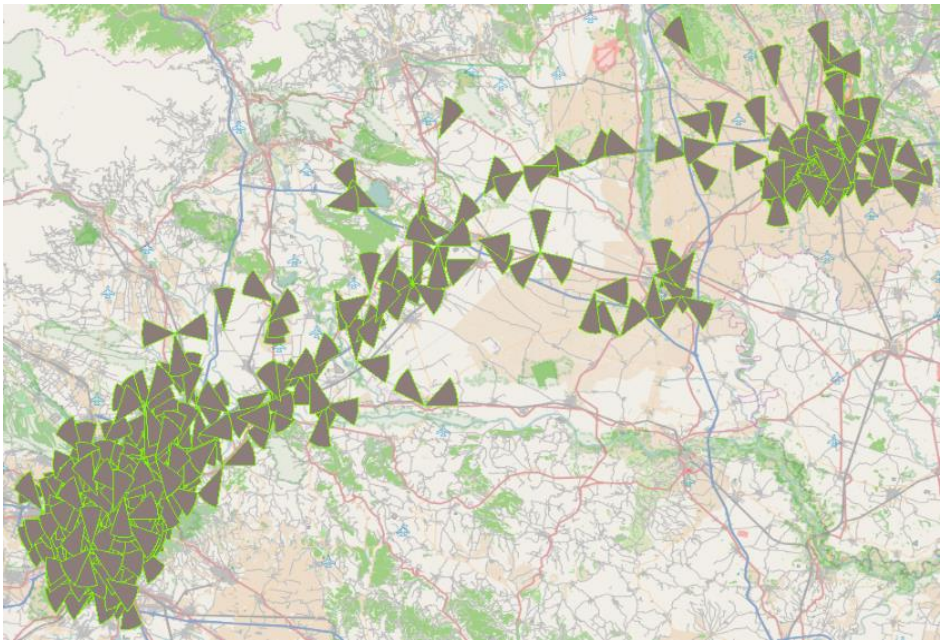
Tablica 4. Usporedba rezultata jutarnjih i popodnevnih mjerenja

U tablici su prikazane količine jednako klasificiranih ćelija u različitom periodu. Neke ćelije su različito klasificirane (označeno narančasto).

Usporedbom različitih perioda rada ćelija potvrđuje se pretpostavka da ćelija ne mora biti tijekom cijelog dana klasificirana kao „*high-mobility*“ ćelija. Njezino stanje ovisi o količini i načinu kretanja korisnika, što se mijenja kroz period sata, dana, tjedna, itd.

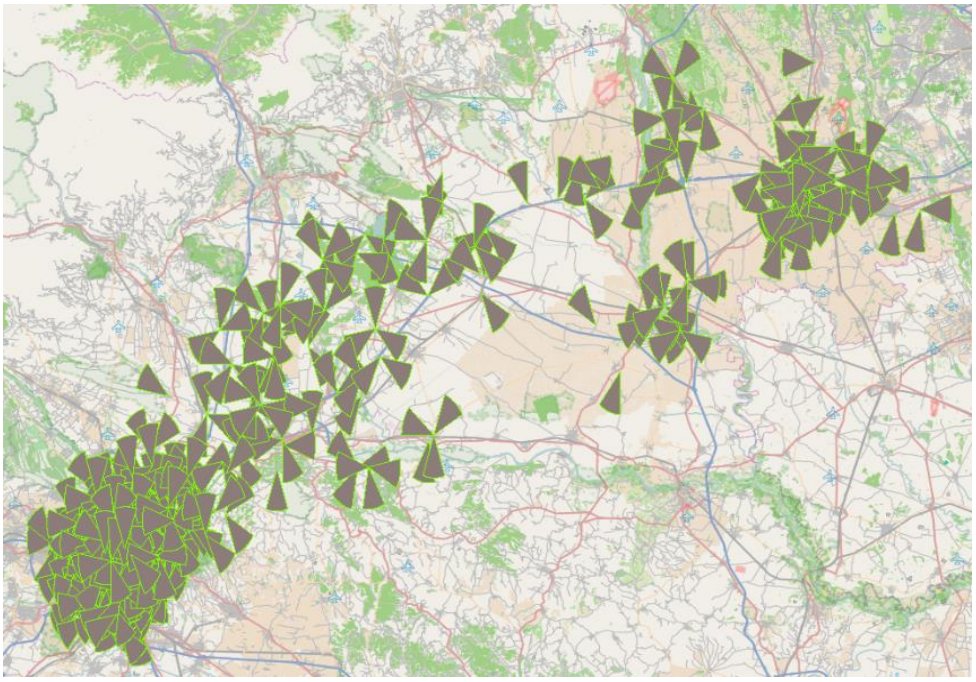
Ipak, istaknuta dijagonala (označena zeleno) unutar Tablice 4. pokazuje kako je veći dio ćelija ispravno klasificiran kao ćelija visokog stupnja mobilnosti, čime se dokazala sezonalnost i ponovljivost rezultata.

Rezultati popodnevnog mjerenja prikazani na karti:



Slika 34. Klaster – 0 / K-means, popodne

Slika 35. Klaster – 1 / K-means, popodne



Slika 36. Klaster – 2 / K-means, popodne

Slika 37. Klaster – 2 / K-means, popodne

# 5. ZAKLJUČAK

Zadatak ovog diplomskog rada je klasifikacija profila ćelija iz podataka korisničkih profila telekomunikacijske mreže, što je i realizirano. Implementirane su i uspoređene dvije metode strojnog učenja (grupiranja): K-means i GMM metoda. Usporedbom rezultata dobivenih putem obje metode može se zaključiti da između njih gotovo i nema razlike. Obje metode rezultirale su podjelom jednakog broja ćelije u klasterima, iako se vidi neznatna razlika u geografskom prikazu. Bez obzira na različit način obrade podataka, kojih i nije bilo „puno" u usporedbi s podacima na projektima za koje se ove metode koriste u praksi, obje metode su trajale oko 2 – 3 min, pri čemu je programski kôd K-means metode neznatno jednostavniji. Prema tome, ne može se ocijeniti koja je od tih dviju metoda bolja. Kao što je i navedeno u radu, ne postoji nužno bolja i lošija metoda, sve ovisi o potrebama korisnika i prirodi podataka. Iz vizualizacije rezultata na karti, intuitivno se može zaključiti da je K-means metoda dala „točnije rezultate", iako se u svim rezultatima vizualno istaknuo barem jedan klaster ćelija koji pokriva autoput i prugu, kako je i bilo očekivano. Pod „točnije rezultate" smatraju se rezultati bliski očekivanima. U ovom slučaju, bilo bi zanimljivo pokušati s hijerarhijskim grupiranjem ili MeanShift metodom. Može se reći da je u ovom zadatku klasificiranja bilo vrlo jednostavno izabrati metodu, jer je bio poznat broj željenih klastera.

Ćelije se ne mogu striktno definirati kao „*high-mobility*". „*Mobility*" svojstvo ćelije je vremenski zavisno. Ćelije koje su svrstane u „*high-mobility*" profil, ne moraju nužno imati „high-*mobility*" ponašanje 24/7 jer se primjerice vikendima manje putuje, postoje i tzv. „špice" kada je povećanje svojstva mobilnosti uzrokovano korisnicima koji odlaze i dolaze s posla. „*Mobility*" svojstvo ovisi o tome koje je doba dana, kreće li se kroz ćeliju veći ili manji broj korisnika. Dakle, postoji sezonalnost u ponašanju. Jedini način da se potpuno točno provjeri metoda jest ručna analiza svake ćelije zasebno, direktnim pogledom u mrežu, u njezine performanse i statistiku rada, što je praktički nemoguće i nitko to ne bi radio na razini mreže.

Kako bi se sa sigurnošću moglo reći da je ćelija „*high-mobility*", podaci bi se trebali periodički prikupljati, analizirati i uspoređivati.

Ovaj rad je rezultirao uspješnim klasificiranjem profila, što može poslužiti kao osnova za daljnju analizu čime bi se identificirao postotak stanja visoke mobilnosti pojedine ćelije. Daljnjom analizom moglo bi se definirati postotak vremena ćelije (npr. 30-50%, 50-70%, 70-

90%, 90%+) kada ona ima svojstvo „*high-mobility*“ i koji je to interval dana ili tjedna. Na taj način bi se došlo do podataka o tome kakav je obrazac ponašanja pojedinih ćelija.

# 6. LITERATURA

[1] R. C. Raciti: *Cellular teghnology*, New Southeastern University, 2012.

[2] D. Yu. Ignatov, A. N. Filippov, A. D. Ignatov, X. Zhang: *Homogenous Network Optimization* , 2016.

[3] Golio, Mike: *RF and Microwave Passive and Active Technologies*, 2018.

[4] Guowang Miao, Jens Zander, Ki Won Sung, Ben Slimane: *Fundamentals of Mobile Data Networks*, Cambridge University, 2016.

[5] N. Nasser, A. Hasswa, H. Hassanein: *Handovers in fourth generation heterogeneous networks*, IEEE Commun. Mag., vol. 44, Oct. 2006.

[6] G. L. Stuber: *Principles of Mobile Communication*, 2nd ed. Norwell, MA: Kluwer Academic Publishers, 2001.

[7] Leslie D. Fife, Le Gruenwald: *Research Issues for Data Communication in Mobile Ad-Hoc Network Database Systems*, Brigham Young University – Hawaii, The University of Oklahoma, School of Computer Science, Norman, 2013.

[8] K. Tutschku: *Demand-based Radio Network Planning of Cellular Mobile Communication Systems*, University of Wurzburg Institute of Computer Science Research Report Series, 1997.

[9] Yufei Wu and Samuel Pierre: O*ptimization of 3G Mobile Network Design Using a Hybrid Search Strategy*, Journal of communications and networks, 2005.

[10] D. Varoutas, D. Katsianis, Th. Sphicopoulos, K. Stordahl, I. Welling: *On the Economics of 3G Mobile Virtual Network Operators*, Wireless Personal Communications, 2006.

[11] Suk Yu Hui, Kai Hau Yeung: *Challenges in the Migration to 4G Mobile Systems*, City University of Hong Kong, 2003.

[12] Bao-Shuh P. Lin, Wen-Hsiang Tsai, C.C. Wu, P.H. Hsu, J.Y. Huang, Tsai-Hwa Liu: *The Design of Cloud-based 4G/LTE for Mobile Augmented Reality with Smart Mobile Devices*, National Chiao Tung University, 2013.

[13] Pol Blasco, Deniz Gunduz: *Learning-Based Optimization of Cache Content in a Small Cell Base Station*, Imperial College London, UK, 2014.

[14] Juan Carrasquilla, Roger G. Melko: *Machine learning phases of matter*, University of Waterloo, Ontario, Canada, 2016.

[15] Alpaydin, Ethem: *Introduction to Machine Learning*, MIT Press, 2010.

[16] Mohri Mehryar, Rostamizadeh Afshin, Talwalkar Ameet: *Foundations of Machine Learning*, The MIT Press, 2012.

[17] Kanungo Tapas, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Silverman Ruth, Angela Y. Wu: *An efficient k-means clustering algorithm: Analysis and implementation*, IEEE Transactions on Pattern Analysis and Machine Intelligence. 2002.

[18] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery: *Section 16.1. Gaussian Mixture Models and k-Means Clustering, Numerical Recipes: The Art of Scientific Computing(3rd ed.),* New York (NY): Cambridge University Press., 2007.

[19] Coates Adam, Lee Honglak, Andrew Y. Ng: *An analysis of single-layer networks in unsupervised feature learning*, International Conference on Artificial Intelligence and Statistics, 2011.

[20] Cepy Slamet, Ali Rahman, Muhammad Ali Ramdhani, Wahyudin Darmalaksana: *Clustering the Verses of the Holy Qur'an using K-Means Algorithm,* Department of Informatics Engineering, UIN Sunan Gunung Djati Bandung, Indonesia, 2016.

[21] M. P. Bhatia, D. Khurana: *Analysis of Initial Centers for K-Means Clustering Algorithm,* International Journal Computer Application, 2013.

[22] Vijayarani S., Ilamathi M. J., Nithya, M.: *Preprocessing Techniques for Text Mining: An Overview*, International Journal Computer Science and Communication Network, 2015.

[23] Marco Capó, Aritz Pérez, Jose A. Lozano: *An efficient approximation to the K-means clustering for Massive Data*, University of the Basque Country UPV/EHU, 2016.

[24] Yu, Guoshen: *Solving Inverse Problems with Piecewise Linear Estimators: From Gaussian Mixture Models to Structured Sparsity*, IEEE Transactions on Image Processing., 2012.

[25] Permuter H., Francos, J., Jermyn I.H.: *Gaussian mixture models of texture and colour for image database retrieval*, IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003.

[26] B. G. Lindsay B. G.: *Mixture Models: Theory, Geometry, and Applications*, NSF-CBMS Regional Conference Series in Probability and Statistics, Hayward: Institute of Mathematical Statistics., 2015.

# 7. PRILOZI

I. Programski kôd za određivanje raspona vremena

```
import sqlite3
     import numpy as np
     import scipy
     import matplotlib.pyplot as plt
     from statsmodels.distributions.empirical_distribution import ECDF
     import math

     def geometricMean(arr, n):

         # declare sum variable and
         # initialize it to 1.
         sum = 0;

         # Compute the sum of all
         # the elements in the array.
         for i in range(n):
             sum = sum + math.log(arr[i]);

         # compute geometric mean
         # through formula antilog
         # (((log(1) + log(2) + . .
         # ... + log(n))/n)
         # and return the value to
         # main function.
         sum = sum / n;

         return math.exp(sum);

     con=sqlite3.connect('C:/Users/data.db')
     cur = con.cursor()

     cgi_group_query = '''
                      SELECT TIME_IN_CELL_s
                      FROM ctr_mobility

                     '''

     cur.execute(cgi_group_query)

     data = []
     for values in cur.fetchall():
         if values[0] != '':# and float(values[0]) < 26 :
             data.append(float(values[0]))


     for q in [50, 90, 95, 100]:
         print ("{}% percentile: {}".format (q, np.percentile(data, q)))

     num_bins = 10

     counts, bin_edges = np.histogram (data, bins=num_bins, normed=True)
```

```
        cdf = np.cumsum (counts)
        #plt.plot (bin_edges[1:], cdf/cdf[-1])

        #plt.hist(data, bins=15)

        ecdf = ECDF(data)
        # get cumulative probability for values
        print('P(x<2): %.3f' % ecdf(2))
        print('P(x<3): %.3f' % ecdf(3))
        print('P(x<5): %.3f' % ecdf(5))
        print('P(x<10): %.3f' % ecdf(10))
        print('P(x<15): %.3f' % ecdf(15))
        print('P(x<20): %.3f' % ecdf(20))
        print('P(x<40): %.3f' % ecdf(40))
        print('P(x<60): %.3f' % ecdf(60))
        print('P(x<80): %.3f' % ecdf(80))
        print('P(x<110): %.3f' % ecdf(110))
        # plot the cdf
        plt.plot(ecdf.x, ecdf.y)

        gmean = geometricMean(data, len(data))
        print("Geometric mean: ", gmean)


    plt.show()
```

II. Programski kôd za razvrstavanje korisnika unutar ćelija u zadanom vremenskom rasponu

```
import sqlite3
     import numpy as np
     import matplotlib.pyplot as plt

con=sqlite3.connect('C:/Users/data.db')
     cur = con.cursor()

     cgi_group_query = '''
                       SELECT TIME_IN_CELL_s
                       FROM ctr_mobility JOIN ENM_CGI ON
ENM_CGI.CGI=ctr_mobility.SOURCE_CGI
                      where X1 = 'False' and X2 = 'False'

                      '''

     cur.execute(cgi_group_query)

     buckets   =   ['0-1',   '1-2','2-3','3-4','4-5','5-7','7-10','10-15','15-
20','20-25','25+']
     #buckets = ['0-2','2-3','3-4','4-5','5-10','10-11','11-12', '12-15','15-
20','20-25','25+']
```

```
        data = [0] * len(buckets)
        for values in cur.fetchall():
            if values[0] != '':
                value = float(values[0])
                bucket_index = 0
                for bucket in buckets:
                    if '+' in bucket:
                        vals = bucket.split('+')
                        max = float(vals[0])
                        if value >= max:
                            data[bucket_index] += 1
                    else:
                        vals = bucket.split('-')
                        min = float(vals[0])
                        max = float(vals[1])
                        if value >= min and value < max:
                            data[bucket_index] += 1
                            break
                        else:
                            bucket_index += 1




        x_pos = [i for i, _ in enumerate(buckets)]
        plt.bar(x_pos, data, color='green')

        plt.xlabel("Ranges")
        plt.ylabel("Number of samples")
        plt.title("Series distribution")

        plt.xticks(x_pos, buckets)

    plt.show()
```

III. Programski kôd za provedbu K-means metode

```
import sqlite3
import numpy as np
from sklearn.cluster import KMeans

con=sqlite3.connect('C:/Users/data.db')
cur = con.csor()

cgi_group_query = '''
                    SELECT SOURCE_CGI, group_concat(TIME_IN_CELL_s)
             FROM ctr_mobility
             GROUP BY SOURCE_CGI
            '''


  cur.execute(cgi_group_query)
```

```
buckets = ['0-1', '1-2','2-3','3-4','4-5','5-7','7-10','10-15','15-20','20-
25','25+']

data = []
cells = []
for cgi, values in cur.fetchall():
    time_in_cell_for_cgi = values.split(',')

    feature_vector = [0] * len(buckets)
    for current_time in time_in_cell_for_cgi:
        if current_time == '':
            continue

        working_time = float(current_time)

        bucket_index = 0
        for bucket in buckets:
            if '+' in bucket:
                vals = bucket.split('+')
                max = float(vals[0])
                if working_time >= max:
                    feature_vector[bucket_index] += 1
            else:
                vals = bucket.split('-')
                min = float(vals[0])
                max = float(vals[1])
                if working_time >= min and working_time < max:
                    feature_vector[bucket_index] += 1
                    break
                else:
                    bucket_index += 1

    data.append(feature_vector)
    cells.append(cgi)

X = np.array(data)
kmeans = KMeans(n_clusters=4, random_state=23).fit(X)

byCluster = dict()
for clustindex in kmeans.labels_:
    byCluster.update({clustindex:[]})

for i in range(len(data)):
    byCluster[kmeans.labels_[i]].append(cells[i])
    # print (cells[i], ' je u clusteru: ', kmeans.labels_[i])

for clust in byCluster.keys():
    print("Clust: ", clust)
    for cells in byCluster[clust]:
        print(cells)
```

## IV. Programski kôd za provedbu GMM metode i usporedbu K-means i GMM metode

```
  # jutarnji dataset
  con=sqlite3.connect('C:\Users\data.db')

  cur = con.cursor()

  cgi_group_query = '''
                    SELECT SOURCE_CGI, group_concat(TIME_IN_CELL_s)
                    FROM ctr_mobility
                    GROUP BY SOURCE_CGI
                   '''

  cur.execute(cgi_group_query)

  buckets  =  ['0-1','1-2','2-3','3-4','4-5','5-7','7-10','10-15','15-20','20-
25','25+']

  data = []
  cells = []
  for cgi, values in cur.fetchall():
      time_in_cell_for_cgi = values.split(',')

      feature_vector = [0] * len(buckets)
      for current_time in time_in_cell_for_cgi:
          if current_time == '':
              continue

          working_time = float(current_time)

          bucket_index = 0
  for bucket in buckets:
              if '+' in bucket:
                  vals = bucket.split('+')
                  max = float(vals[0])
                  if working_time >= max:
                      feature_vector[bucket_index] += 1
              else:
                  vals = bucket.split('-')
                  min = float(vals[0])
                  max = float(vals[1])
                  if working_time >= min and working_time < max:
                      feature_vector[bucket_index] += 1
                      break
                  else:
                      bucket_index += 1

      data.append(feature_vector)
      cells.append(cgi)

  X = np.array(data)

  # metoda-1
  kmeans=KMeans(n_clusters=4,   init='k-means++',   n_init=20,   max_iter=500,
tol=0.00001).fit(X)
  clusters_kmeans = kmeans.labels_
```

```
byCluster_kmeans_1 = dict()
for clustindex in clusters_kmeans:
    byCluster_kmeans_1.update({clustindex:[]})

for i in range(len(data)):
    byCluster_kmeans_1[clusters_kmeans[i]].append(cells[i])

# popodnevni dataset
con=sqlite3.connect(''C:\Users\Data/ctr_mobility.db')

cur = con.cursor()

cgi_group_query = '''
                  SELECT SOURCE_CGI, group_concat(TIME_IN_CELL_s)
                  FROM ctr_mobility
                  GROUP BY SOURCE_CGI
                 '''

cur.execute(cgi_group_query)

buckets = ['0-1', '1-2','2-3','3-4','4-5','5-7','7-10','10-15','15-20','20-
25','25+']

data = []
cells = []
for cgi, values in cur.fetchall():
    time_in_cell_for_cgi = values.split(',')

    feature_vector = [0] * len(buckets)
    for current_time in time_in_cell_for_cgi:
        if current_time == '':
            continue
           working_time = float(current_time)

      bucket_index = 0
       for bucket in buckets:
            if '+' in bucket:
                vals = bucket.split('+')
                max = float(vals[0])
                if working_time >= max:
                    feature_vector[bucket_index] += 1
            else:
                vals = bucket.split('-')
                min = float(vals[0])
                max = float(vals[1])
                if working_time >= min and working_time < max:
                    feature_vector[bucket_index] += 1
                    break
                else:
                    bucket_index += 1

    data.append(feature_vector)
    cells.append(cgi)

X = np.array(data)
```

```
# metoda-1
kmeans      =      KMeans(n_clusters=4,      init='k-means++',      n_init=20,
max_iter=500tol=0.00001).fit(X)
clusters_kmeans = kmeans.labels_

byCluster_kmeans_2 = dict()
for clustindex in clusters_kmeans:
    byCluster_kmeans_2.update({clustindex:[]})
V.
for i in range(len(data)):
    byCluster_kmeans_2[clusters_kmeans[i]].append(cells[i])



# print header
for gmmCluster in byCluster_kmeans_1.keys():
    print("UJUTRO-"+str(gmmCluster), sep=' \t ', end=' \t ', flush=True)

for kmeansCluster in byCluster_kmeans_2.keys():
    kmeansCells = set(byCluster_kmeans_2[kmeansCluster])
    print("\n POPODNE-"+str(kmeansCluster), sep='\t', end=' \t ', flush=True)
    for gmmCluster in byCluster_kmeans_1.keys():
        gmmCells = set(byCluster_kmeans_1[gmmCluster])

        commonElements = len(kmeansCells.intersection(gmmCells))
        diff = len(list(kmeansCells - gmmCells))
        diff += len(list(gmmCells - kmeansCells))
        print(commonElements, sep=' \t ', end=' \t ', flush=True)
```