SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2981

Usporedba algoritama grupiranja u postupcima otkrivanja anomalija

Jelena Nemčić

Zagreb, lipanj 2022.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 11. ožujka 2022.

DIPLOMSKI ZADATAK br. 2981

Pristupnica:	**Jelena Nemčić (0036497921)**
Studij:	Računarstvo
Profil:	Računarska znanost
Mentor:	izv. prof. dr. sc. Goran Delač

Zadatak: **Usporedba algoritama grupiranja u postupcima otkrivanja anomalija**

Opis zadatka:

Odabrati, proučiti i opisati algoritme za grupiranje primjerene obradi velikih skupova podataka. Opisati obilježja algoritama i objasniti njihov princip rada nad pokaznim primjerima. Proučiti postojeće metrike za vrednovanje uspješnosti algoritama grupiranja. Odabrati primjeren skup podataka za postupak otkrivanja anomalija. Programski ostvariti i provesti vrednovanje odabranog podskupa algoritama nad odabranim skupom podataka. Opisati programsko ostvarenje sustava, rezultate vrednovanja algoritama te navesti korištenu literaturu i primljenu pomoć.

Rok za predaju rada: 27. lipnja 2022.

Sadržaj

1. Uvod

Svaki dan stvara se velika količina podataka koja se zatim obrađuje kako bi se iz nje saznale nove informacije. Jedan od načina korištenja podataka jest otkrivanje neobičnog ponašanja i pronalaženje anomalija.

Anomalijom se smatra svaki događaj ili opažanje koje značajno odstupa od većine podataka i ne ponaša se na očekivan način. Takvi primjeri mogu izazvati sumnju da ih proizvodi drugačiji mehanizam ili se činiti nedosljednima s ostatkom tog skupa podataka.

Otkrivanje anomalija pronalazi primjenu u mnogim domenama uključujući kibernetičku sigurnost, medicinu, računalni vid, statistiku, neuroznanost i oružane snage. Koristi se također i za otkrivanje financijskih prijevara, industrijskih oštećenja i poremećaja u ekosustavu. Anomalije mogu predstavljati problem te su tada tražene radi namjernog izostavljanja iz skupa podataka kako bi se dobila točnija statistička analiza ili bolje predviđanje nekog modela strojnog učenja. Međutim, u mnogim su primjenama anomalije najzanimljiviji dio skupa podataka i predstavljaju novu pojavu koju je potrebno prepoznati i dalje istražiti.

Jedna od tehnika otkrivanja anomalija jest korištenje algoritama grupiranja s ciljem pronalaženja elemenata koji ne pripadaju ni jednoj grupi. U ovom radu dano je objašnjenje problema pronalaska anomalija, opis različitih algoritama grupiranja i korištenih skupova podataka te usporedba izvedbe tih algoritama u postupcima otkrivanja anomalija. Algoritmi odabrani za usporedbu su algoritmi K-sredina, DBSCAN i Gaussova mješavina, a ispitivani su na problemima otkrivanja prijevara kreditnim karticama, detekcije upada u mrežu i detekcije raka.

2. Anomalije

2.1. Pojava anomalija i njeni uzroci

Postoji više pokušaja definiranja anomalija, a većina njih opisuje anomaliju kao opažanje čiji se obrazac ponašanja razlikuje od očekivanog, najčešće se pojavljuje vrlo rijetko u skupu podataka i njegova su obilježja značajno drugačija od onih većine preostalih opažanja. Također, anomalijom se može smatrati podatak koji se čini nedosljedan i relativno udaljen od drugih podataka iz skupa ili izaziva sumnju da ga proizvodi drugačiji mehanizam.

Anomalije se mogu pojaviti u bilo kojem skupu podataka i njihovo otkrivanje može biti od izuzetne važnosti. Često se detekcija anomalija provodi u predobradi kako bi se mogle ukloniti iz skupa podataka. Time se dobiva točnija statistika podataka, bolje predviđanje modela strojnog učenja i bolja vizualizacija podataka. S druge strane, anomalije mogu biti najvažnija i najzanimljivija opažanja i tada se otkrivanje anomalija provodi radi njih samih. Primjeri takve primjene su otkrivanje upada u području kibernetičke sigurnosti, otkrivanje financijskih prijevara i lažnih informacija, otkrivanje kvarova i pogrešaka, praćenje stanja sustava i vremenskih serija, detekcija događaja u senzorskim mrežama, otkrivanje poremećaja u ekosustavu, otkrivanje nedostataka na slikama pomoću računalnog vida te postavljanje medicinske dijagnoze i provođenje zakona.

Mogući uzroci pojave anomalija su:

1. Podaci pripadaju različitim razredima.
   - Anomalije se razlikuju od ostalih podataka jer pripadaju drugom razredu, koji ima drugačija obilježja.
   - Primjer takvih anomalija su financijske prijevare, strani upad u sustav i pojava bolesti.
2. Prirodna varijacija.

- Neki skupovi podataka mogu se modelirati normalnom distribucijom, gdje su anomalije oni događaji koji imaju vrlo malu vjerojatnost pojavljivanja.

3. Pogreške u mjerenju ili prikupljanju podataka.
   - Do pojave anomalija može doći ako podaci sadrže šum, ako postoji kvar u mjernim instrumentima ili zbog ljudske pogreške.
   - Krajnji je cilj eliminirati ovakve anomalije jer smanjuju kvalitetu podataka.

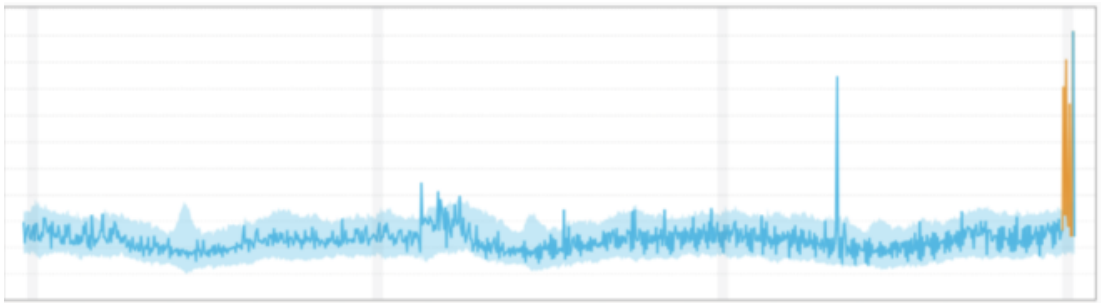
U ovom radu razmatrat će se samo anomalije koje se javljaju kao posljedica činjenice da podaci prirodno pripadaju različitim razredima.

2.2. Klasifikacija anomalija

Kako bi sustav za otkrivanje anomalija mogao točno prepoznati potencijalna odstupanja nužno je znati koja vrsta anomalije se očekuje. Anomalije se mogu podijeliti u tri glavne kategorije:

1. Globalne anomalije

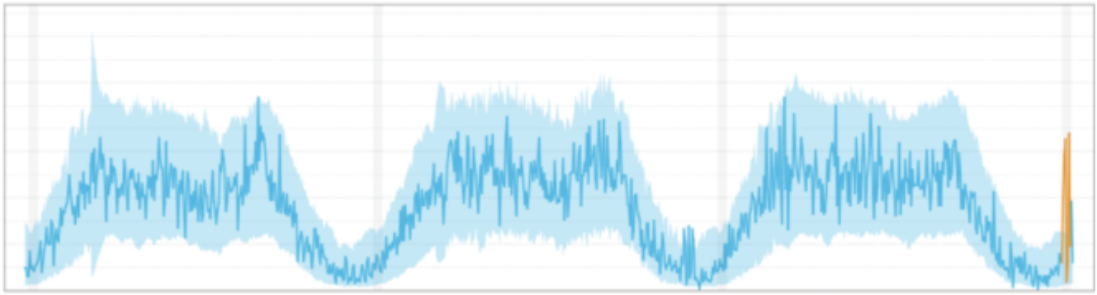
   Opažanje se smatra globalnim odstupanjem ili globalnom anomalijom ako se njegova vrijednost ili vrijednost nekih njegovih obilježja značajno razlikuje od vrijednosti cjelokupnog skupa podataka. Gledano u n-dimenzionalnom prostoru, taj se podatak nalazi daleko od svih ostalih podataka iz skupa. Primjer globalne anomalije dan je na slici 2.1.



Slika 2.1: Globalna anomalija. Preuzeto s [5]

2. Kontekstualne anomalije

Kontekstualne ili uvjetne anomalije su opažanja čije se vrijednosti znatno razlikuju od ostalih opažanja koja postoje u istom kontekstu. Takve vrijednosti ne moraju biti izvan globalnih očekivanja, ali odudaraju od konteksta u kojem se nalaze. Također, jedan podatak koji je anomalija u jednom kontekstu ne mora biti anomalija u drugom kontekstu u istom skupu podataka. Ovakva odstupanja najčešća su u podacima vremenskih serija jer takvi skupovi podataka sadrže zapise ovisne o vremenskom razdoblju. Slika 2.2 prikazuje primjer takve anomalije.



Slika 2.2: Kontekstualna anomalija. Preuzeto s [5]

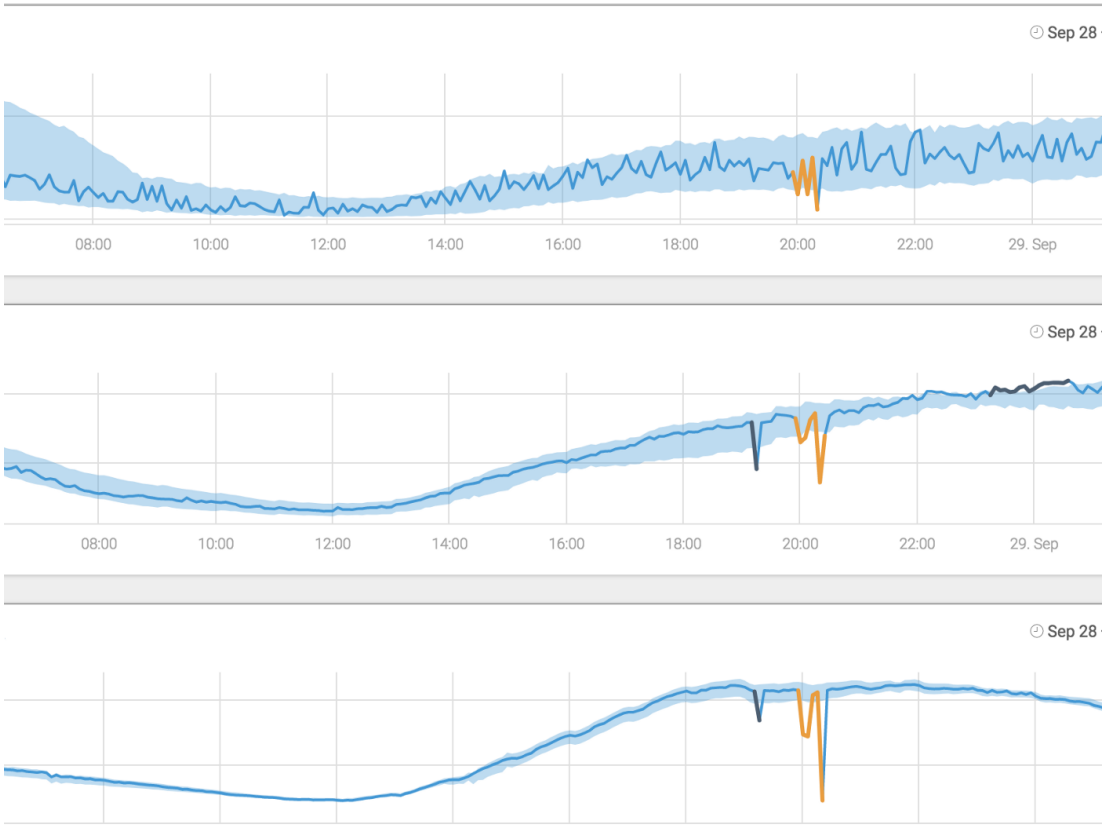
3. Kolektivne anomalije

   Podskup podataka smatra se kolektivnom anomalijom ako njihove vrijednosti kao grupa značajno odstupaju od cijelog skupa podataka, ali vrijednosti pojedinačnih podataka nisu same po sebi anomalne ni u globalnom ni u kontekstualnom smislu. U podacima vremenskih serija kolektivne anomalije mogu se manifestirati kao vrhovi i doline koje se javljaju izvan vremenskog okvira kada je takvo ponašanje normalno, kao što se vidi na slici 2.3.

Ovisno o vrsti anomalije primjenjuju se različite metode i načini detekcije. Ovaj rad fokusira se na globalne anomalije i njihovo pronalaženje.

2.3. Problem otkrivanja anomalija

Otkivanje anomalija može se shvatiti kao problem definiranja očekivanog ponašanja podataka ili granica unutar kojih se podaci smatraju normalnima te prepoznavanja točaka koje se ne nalaze unutar njih. Postoji nekoliko čimbenika koji čine ovaj problem vrlo teškim.



Slika 2.3: Kolektivna anomalija. Preuzeto s [5]

– Učinkovito modeliranje normalnih vrijednosti i ponašanja može biti vrlo izazovan problem. Često je teško nabrojati sva moguća normalna ponašanja nekog objekta i klasificirati neki podatak kao anomaliju. Također, granica između normalnih podataka i anomalija može biti vrlo nejasna.

– Svaki problem zahtijeva specifičan način detekcije anomalija jer su odabir mjere sličnosti i modeliranje odnosa ovisni o svojstvima tog problema. Zbog toga nije moguć razvoj univerzalno primjenjive metode otkivanja anomalija.

– Prikupljeni podaci često sadrže šum koji može imati vrijednosti koje znatno odstupaju od normalnih ili čak nedostaju. Šum smanjuje kvalitetu podataka i otežava definiranje granica između normalnih podataka i anomalija te se često šum može pogrešno odrediti kao anomalija i obrnuto.

– Mnogi načini otkrivanja anomalija postaju neučinkoviti u slučaju velike dimenzionalnosti skupa podataka. Podaci su tada rijetki i udaljenosti među podacima su sve veće te se puno točaka može pogrešno klasificirati kao anomalija.

– U nekim primjenama korisnik ne želi samo prepoznati anomalije već i razumjeti zašto su ti podaci detektirani kao abnormalni. Zbog toga metoda otkrivanja anomalija mora biti razumljiva, smislena i pružiti opravdanje detekciji.

2.4. Metode otkrivanja anomalija

Postoji puno različitih tehnika otkrivanja anomalija i one se mogu podijeliti u četiri glavne kategorije.

1. Statističke metode

   Statistički pristup naziva se još i pristup zasnovan na modelu jer sadrži model koji opisuje obilježja skupa podataka. Model najčešće sadrži distribuciju vjerojatnosti podataka i za svaki podatak računa se vjerojatnost njegova pojavljivanja u tom modelu. Ako je ta vjerojatnost vrlo mala, podatak se proglašava anomalijom.

2. Metode zasnovane na blizini

   (a) Metode zasnovane na udaljenosti

   Metode zasnovane na udaljenosti pretpostavljaju da je podatak anomalija ako mu se najbliži susjedi nalaze daleko u prostoru značajki, odnosno ako blizina njegovih susjeda značajno odstupa od blizine većine drugih objekata i njihovih susjeda u istom skupu podataka.

   (b) Metode zasnovane na gustoći

   Metode zasnovane na gustoći koriste broj podataka koji se nalaze unutar definiranog prostora ispitivanog podatka za definiranje lokalne gustoće. Što je lokalna gustoća objekta manja, veća je vjerojatnost da je on anomalija.

3. Metode zasnovane na grupiranju

   Metode koje se zasnivaju na grupiranju pretpostavljaju da normalni podaci pripadaju velikim i gustim grupama, dok anomalije pripadaju malim i rijetkim grupama ili ne pripadaju ni jednoj. Razlika između grupiranja i metoda zasnovanih na gustoći je u tome što grupiranje dijeli podatke u grupe, dok metode zasnovane na gustoći dijele podatkovni prostor.

U ovom radu za detekciju anomalija koristit će se metode zasnovane na grupiranju, odnosno algoritmi grupiranja. Za usporedbu su izabrani algoritam K-sredina, algoritam DBSCAN i model Gaussove mješavine.

3. Algoritmi grupiranja

3.1. O algoritmima grupiranja

Grupiranje je podjela skupa podataka u grupe na način da su podaci u istoj grupi sličniji jedni drugima nego podacima iz ostalih grupa. Cilj jest pronalaženje intrinzičnih grupa u skupu podataka. Algoritmi grupiranja pripadaju u skupinu nenadziranih metoda strojnog učenja jer su ulazni podaci dani bez ciljnih vrijednosti, odnosno nisu označeni.

3.1.1. Podjela

Grupiranje se može podijeliti u dvije kategorije:

1. Tvrdo grupiranje - podatak ili pripada grupi ili ne pripada
2. Meko grupiranje - podatak pripada svakoj grupi s određenom vjerojatnošću

Osim po tipu grupiranja koje provode, algoritmi grupiranja razlikuju se i po tome kako definiraju pojam grupe i sličnost podataka. Svaki algoritam pretpostavlja specifičan model grupe, a najčešći modeli su:

1. Modeli povezanosti - na osnovi udaljenosti podataka stvara se hijerarhijsko stablo grupa
2. Centroidni modeli - podaci se organiziraju u nehijerarhijske grupe ovisno o udaljenosti od centra te grupe
3. Modeli distribucije - grupe se modeliraju pomoću vjerojatnosti da podaci pripadaju istoj statističkoj distribuciji
4. Modeli gustoće - područja veće gustoće povezuju se u grupe

Ne postoji objektivno najbolji algoritam grupiranja, već odabir algoritma ovisi o problemu koji se rješava. Algoritam se može odabrati na osnovi modela grupe ili eksperimentalno. Također, algoritam dizajniran za jednu vrstu modela grupe općenito neće raditi na skupu podataka koji sadrži drugačiji tip grupa.

U ovom radu uspoređivat će se tri različita modela: algoritam K-sredina kao predstavnik centroidnih modela, DBSCAN algoritam kao model gustoće i model Gaussove mješavine koji pripada modelima distribucije.

3.1.2. Vrednovanje

Rezultati algoritama grupiranja mogu se vrednovati na dva načina. Prvi je vredovanje korištenjem podataka za koje su poznate oznake grupa. Takvo vrednovanje mjeri koliko je dobiveno grupiranje blizu unaprijed određenoj podjeli. Metode vrednovanja tada su često prilagođene varijante metoda koje se koriste za vrednovanje klasifikacije. Neke od tih metrika su:

1. Matrica zabune (engl. *Confusion Matrix*)

   Matrica koja opisuje uspješnost modela prikazom broja istinski pozitivnih (engl. *True Positive - TP*), lažno pozitivnih (engl. *False Positive - FP*), istinski negativnih (engl. *True Negative - TN*) i lažno negativnih (engl. *False Negative - FP*) primjera.

2. Točnost (engl. *Accuracy*)

   Točnost je udio točno klasificiranih primjera u skupu svih primjera.

   $$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

3. Preciznost (engl. *Precision*)

   Preciznost predstavlja udio točno klasificiranih primjera među onima koje je model deklarirao kao pozitivne.

   $$P = \frac{TP}{TP + FP}$$

4. Odziv (engl. *Recall*)

   Odziv je udio točno klasificiranih primjera u skupu svih stvarno pozitivnih primjera.

   $$R = \frac{TP}{TP + FN}$$

5. F1-mjera (engl. *F1-score*)

   F1-mjera jest harmonijska sredina preciznosti i odziva i najčešće korištena mjera za usporedbu klasifikatora.

$$F1 = \frac{2PR}{P + R}$$

6. AUC mjera

   ROC krivulja (engl. *Receiver Operating Characteristic curve*) jest graf koji prikazuje odnos stope istinski pozitivnih primjera (engl. *True Positive Rate - TPR*), odnosno odziva i stope lažno pozitivnih primjera (engl. *False Positive Rate - FPR*), koja se računa kao: $\frac{FP}{FP+TN}$. Njihov odnos prikazuje se na svim mogućim pragovima klasifikacije. AUC mjera (engl. *Area under the ROC curve*) predstavlja površinu ispod cijele ROC krivulje.

7. Randov indeks

   Randov indeks računa u kojoj mjeri dobiveno grupiranje odgovara referentnom grupiranju, odnosno točnost na razini parova primjera. Za svaki mogući par iz skupa primjera gleda se jesu li ta dva primjera završila u istoj grupi ili nisu.

$$R = \frac{a + b}{\binom{N}{2}}$$

   gdje je:

   - a - broj jednako označenih parova u istim grupama
   - b - broj različito označenih parova u različitim grupama

Drugi način vrednovanja algoritama grupiranja jest korištenje metrika koje ne zahtijevaju oznake podataka kako bi izračunale učinkovitost algoritma. Najčešće korištene metrike su:

1. Koeficijent siluete (engl. *Silhouette Coefficient*)

   Koeficijent siluete definira se na osnovi udaljenosti unutar grupe i između različitih grupa i računa se kao:

$$S = \frac{1}{N} \sum_{i=1}^{N} \frac{b_i - a_i}{max(a_i, b_i)}$$

   gdje je:

- a - srednja udaljenost između uzorka i i svih ostalih podataka u toj grupi
- b - srednja udaljenost između uzorka i i svih ostalih podataka u drugoj najbližoj grupi

Vrijednost koeficijenta siluete nalazi se u skupu $[-1, 1]$ i što je ona veća grupe su jasnije odijeljene i grupiranje se smatra točnijim.

2. Dunnov indeks

   Dunnov indeks zahtijeva da su udaljenosti primjera unutar grupe male, a udaljenosti između različitih grupa što veće. Računa se kao:

$$D = \frac{\min_{1 \leq i < j \leq m} \delta(C_i, C_j)}{\max_{1 \leq k \leq m} \Delta_k}$$

   gdje je:

   - $\delta(C_i, C_j)$ - udaljenost između grupa C_i i C_j (udaljenost između dva najbliža primjera, dva najudaljenija primjera ili prosječna udaljenost)
   - Δ_k - udaljenost primjera unutar iste grupe (najveća udaljenost između dva primjera, prosječna udaljenost ili udaljenost primjera od centroida grupe)

   Što je vrijednost Dunnovog indeksa veća, bolje je grupiranje.

3. Davies-Bouldin indeks

   Davies-Bouldin indeks računa se kao prosjek sličnosti svake grupe s grupom koja joj je najsličnija:

$$DB = \frac{1}{K} \sum_{i=1}^{K} \max_{j \neq i} \frac{\Delta_i + \Delta_j}{\delta(C_i, C_j)}$$

   Razlikuje se od ostalih metrika jer manja vrijednost ovog indeksa označava bolje grupiranje.

4. DBCV (engl. *Density-Based Clustering Validation*)

   Metrika DBCV računa gustoću unutar grupe i gustoću između grupa. Visoka gustoća unutar grupe i niska gustoća između njih ukazuju na dobro grupiranje.

Za vrednovanje algoritama u ovom radu koristit će se sve navedene metode osim točnosti, Randovog indeksa, Dunnovog indeksa i metode DBCV. Točnost i Randov indeks postaju nepouzdani u slučaju neuravnoteženih razreda, kao što je to slučaj u detekciji anomalija, a Dunnov indeks i DBCV su metode bez svojstva razmjernog rasta, čije računanje postaje računski vrlo zahtjevno već za nekoliko tisuća primjera.

3.2. Algoritam K-sredina

Algoritam K-sredina (engl. *K-Means*) najpoznatiji je algoritam grupiranja koji se zasniva na centroidnom modelu. U ovom algoritmu svaka grupa ima centroid koji se računa kao srednja vrijednost članova grupe i predstavlja tu grupu. Primjeri se iz neoznačenog skupa podataka grupiraju u K grupa na način da svaki podatak pripada onoj grupi čijem je centroidu najbliži.

Algoritam očekuje broj grupa K kao hiperparameter i njegova se vrijednost može odrediti na više načina, a najpoznatiji su:

- Metoda lakta (engl. *Elbow method*)

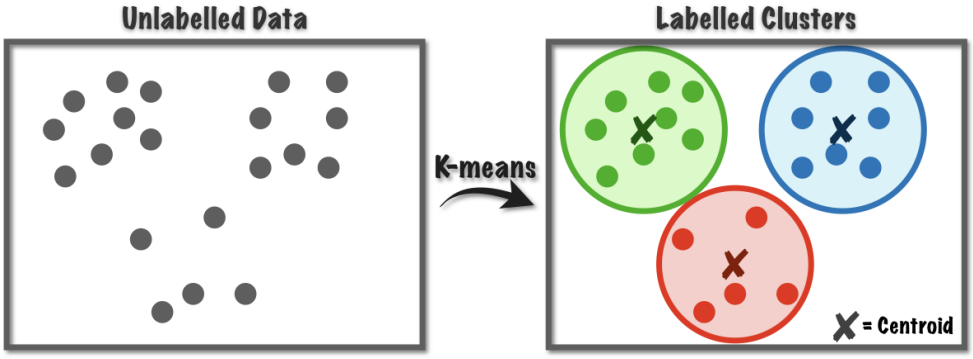
  U metodi lakta grafički se prikazuje ovisnost funkcije gubitka o broju grupa K. S porastom broja grupa vrijednost funkcije će se smanjivati te je cilj pronaći "lakat" funkcije, odnosno broj grupa nakon kojeg se vrijednosti funkcije počinju smanjivati vrlo sporo.

- Analiza siluete (engl. *Silhouette analysis*)

  Ova je metoda grafička metoda koja se zasniva na ranije objašnjenom koeficijentu siluete. Njegova vrijednost prikaže se za svaki primjer iz skupa podataka ovisno o grupi u koju je primjer raspoređen te se izabere onaj broj grupa za koji svi primjeri imaju približno jednak koeficijent siluete.

Osim odabira broja grupa, potrebno je definirati i način odabira početnih centroida. Neki od mogućih pristupa su:

- Nasumičan odabir K primjera.
- Nasumična dodjela grupe svakom primjeru i izračun centroida na osnovi primjera u grupi.
- Izračun srednje vrijednosti sviju primjera i dodavanje K slučajnih vektora toj vrijednosti.



Slika 3.1: Primjer izvođenja algoritma K-sredina uz $K = 3$. Preuzeto s [14]

– Nasumičan odabir prvog centroida, nakon čega se svaki sljedeći bira na način da bude što dalje od postojećih. Verzija algoritma koja ostvaruje ovakav pristup zove se *K-sredine++*.

Postupak grupiranja algoritma K-sredina je iterativan. Nakon inicijalizacije početnih centroida svi se primjeri stavljaju u onu grupu čiji im je centroid najbliži. U sljedećem se koraku, na osnovi razvrstanih primjera, ponovno računaju novi centroidi za svaku grupu. Dalje se ponavljaju ova dva koraka sve do konvergencije, odnosno do trenutka kad više nema promjene u podjeli primjera po grupama i u vrijednostima centroida. Ovaj postupak prikazan je pseudokodom 3.1, a rezultat takvog grupiranja vidljiv je na slici 3.1.

Pseudokod 3.1: Pseudokod algoritma K-sredina

```
definiraj broj grupa K
inicijaliziraj centroide μ_k, k = 1, ..., K
ponavljaj
        za svaki x_i ∈ D
                pronađi najbliži centroid
                dodjeli x_i toj grupi
        za svaki μ_k, k = 1, ..., K
                ažuriraj vrijednost centroida
dok svi μ_k ne konvergiraju
```

Bitna karakteristika algoritma K-sredina jest da pripada algoritmima tvrdog grupiranja, što znači da će svaku točku dodijeliti jednoj i točno jednoj grupi. Algoritam se dobro nosi s velikim skupovima podataka jer ima linearnu vremensku složenost $O(nkdi)$, gdje je:

– n - veličina skupa podataka

– k - broj grupa

– d - dimenzionalnost podataka

– i - broj iteracija algoritma

Međutim, algoritam K-sredina uvijek traži grupe sfernog oblika te ne može prepoznati nekonveksne grupe. Također, jako je osjetljiv na prisutnost anomalija i šuma u podacima.

3.3. DBSCAN algoritam

DBSCAN algoritam (engl. *Density-based spatial clustering of applications with noise*) pripada skupini algoritama grupiranja zasnovanih na gustoći. On grupira zajedno točke koje su blizu jedna drugoj, odnosno točke s mnogo susjednih točaka. Primjere koji nisu svrstani ni u jednu grupu i nalaze se u područjima niske gustoće algoritam označava kao anomalije. Kao i algoritam K-sredina, i DBSCAN je algoritam tvrdog grupiranja.

Glavna ideja algoritma DBSCAN jest da grupa mora sadržavati određeni minimalni broj točaka unutar definiranog polumjera. Zato algoritam zahtijeva dva parametra:

1. *minPts*

   Parametar *minPts* predstavlja najmanji broj točaka u grupi da bi se ona smatrala gusto popunjenom. Za njegovu procjenu može se primijeniti općenito pravilo $minPts \geq D + 1$, gdje je D broj dimenzija skupa podataka. Također, što je veći skup podataka potrebno je odabrati veći *minPts* i tada se može koristiti pravilo $minPts = 2 * D$. Veće vrijednosti obično daju bolje rezultate kada je u podacima prisutan šum.

2. ϵ

   Parametar ϵ jest polumjer unutar kojeg se traže susjedne točke. Pri odabiru vrijednosti ϵ nema općenitog pravila. Vrijednost ne smije biti ni prevelika ni premala i mora biti sukladna udaljenostima među podacima.

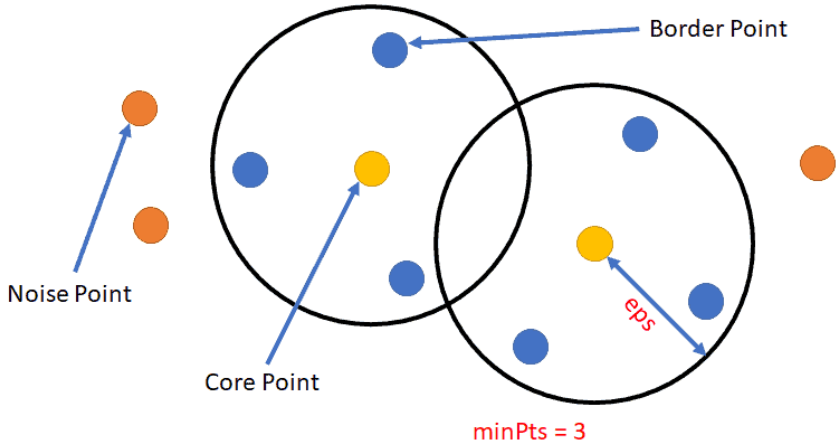
DBSCAN algoritam pridodaje svakoj točki jednu od tri moguće oznake:

1. Središnja točka (engl. *Core point*) - točka oko koje se nalazi minimalno *minPts* drugih točaka unutar udaljenosti ϵ

2. Granična točka (engl. *Border point*) - točka koja ima barem jednu središnju točku na udaljenosti manjoj od ϵ, ali se nalazi na rubu grupe i broj točaka oko nje manji je od *minPts*

3. Točka šuma (engl. *Noise point*) - točka koja nije ni središnja ni granična točka; točka od koje DBSCAN nije znao oblikovati grupu te je proglašava anomalijom

Na slici 3.2 prikazane su grafički različite vrste točaka.



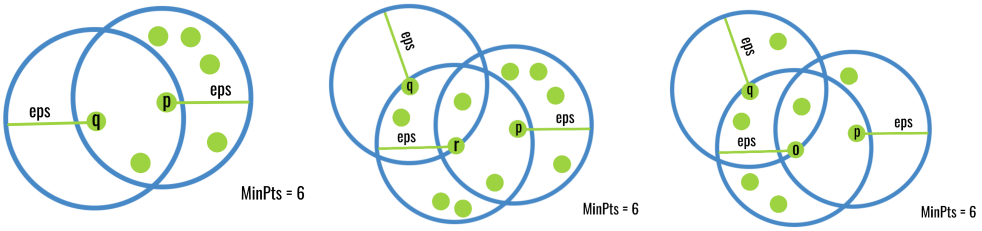
Slika 3.2: Primjer središnje točke, granične točke i točke šuma uz $minPts = 3$. Preuzeto s [27]

Središnja točka p formira grupu zajedno sa svim središnjim i graničnim točkama koje su iz nje dohvatljive. Točka q može biti:

- Izravno dohvatljiva - ako se nalazi unutar udaljenosti ϵ od točke p
- Dohvatljiva - ako postoji put $p_1, ..., p_n$, pri čemu je $p_1 = p$ i $p_n = q$ i svaka točka p_{i+1} izravno je dohvatljiva iz točke p_i

Dohvatljivost nije simetrična relacija, već samo središnje točke mogu dohvatiti granične. Zbog toga je uveden pojam povezanosti, kojim se formalno definira opseg grupe. Dvije točke p i q povezane su ako postoji točka o takva da su i p i q dohvatljive iz o. Ova relacija je simetrična i grupa tada ispunjava sljedeća svojstva:

- Sve točke unutar grupe međusobno su povezane.
- Ako je točka dohvatljiva iz bilo koje točke koja pripada grupe, tada ona također pripada grupi.



(a) Točka q je izravno dohvatljiva iz točke p

(b) Točka q je dohvatljiva iz točke p preko točke r

(c) Točka p i točka q su povezane preko točke o

Slika 3.3: Prikaz izravne dohvatljivosti, dohvatljivosti i povezanosti na primjeru gdje je $minPts = 6$. Slike su preuzete s [8]

Svojstva izravne dohvatljivosti, dohvatljivosti i povezanosti prikazana su grafički na slici 3.3.

Na početku algoritma odabire se nasumična točka od koje se formira grupa ako ona ispunjava navedeni kriterij. Grupe se zatim rekurzivno proširuju obilaženjem ostalih članova grupe. Kad se grupa više ne može proširiti, odabire se nasumično nova točka iz skupa podataka i algoritam se ponavlja dok sve točke nisu obiđene. Pseudokod 3.2 prikazuje pseudokod algoritma DBSCAN.

Pseudokod 3.2: Pseudokod algoritma DBSCAN

```
definiraj minimalan broj članova grupe minPts i udaljenost ε
inicijaliziraj početni broj grupa: C = 0
za svaku točku p ∈ D
        ako je točka p već označena
                nastavi
        pronađi skup točaka S izravno dohvatljivih iz p
        ako je |S| < minPts onda
                označi p kao točku šuma
                nastavi
        C = C + 1
        dodaj p u grupu C
        za svaku točku q ∈ S
                ako točka q već pripada grupi
                        nastavi
                dodaj q u grupu C
                pronađi skup točaka N izravno dohvatljivih iz q
                ako je |N| ≥ minPts onda
                        dodaj točke iz N u skup S
```

Ako je skup podataka spremljen tako da se upiti o susjedstvu mogu izvoditi u logaritamskom vremenu, složenost DBSCAN algoritma je $O(nlogn)$, gdje je n broj podataka. Ako nema strukture indeksiranja, složenost raste na $O(n^2)$.

Prednosti algoritma DBSCAN su što ne zahtijeva definiranje broja grupa unaprijed, može pronaći grupe proizvoljnog oblika i otporan je na prisutnost šuma i anomalija. Nedostatak mu je što ne može dobro grupirati skup podataka u kojem je prisutna velika razlika u gustoći među grupama jer se tada ne može odabrati kombinacija parametara *minPts* i ϵ koja bi bila prikladna za sve grupe.

3.4. Model Gaussove mješavine

Gaussova je mješavina model distribucije koji pretpostavlja da su sve točke iz skupa podataka stvorene iz mješavine konačnog broja Gaussovih distribucija s nepoznatim parametrima. Model zatim grupira podatke na način da svaka grupa sadrži podatke iz jedne distribucije.

Gaussova distribucija još se naziva normalnom distribuciju i ima karakterističan zvonolik oblik. Funkcija gustoće vjerojatnosti Gaussove distribucije u jednoj dimenziji glasi:

$$\mathcal{N}(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{3.1}$$

pri čemu je:

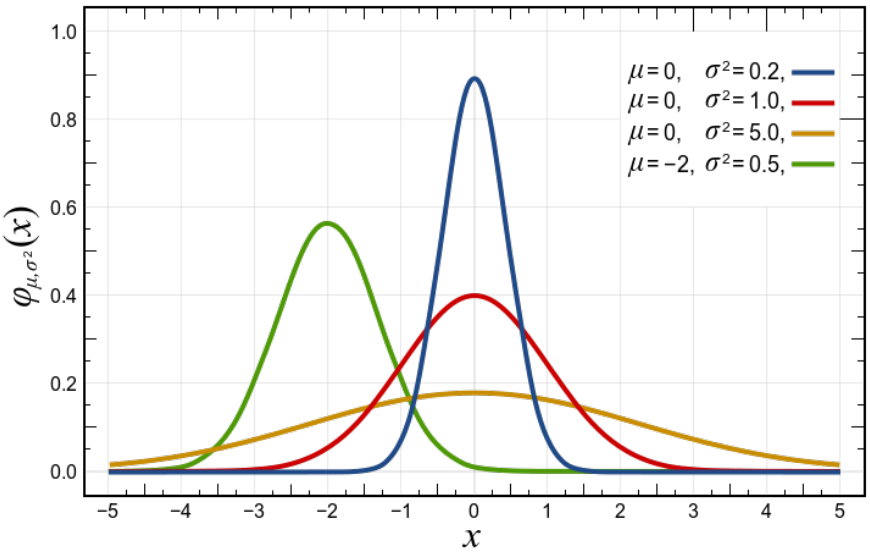
- μ - srednja vrijednost skupa podataka; određuje "visinu" krivulje
- σ - standardna devijacija podataka; određuje "širinu" krivulje

Funkcija gustoće vjerojatnosti daje vjerojatnost dobivanja podatka x u slučaju kada imamo normalnu distribuciju s parametrima μ i σ. Slika 3.4 prikazuje funkciju gustoće za različite vrijednosti parametara μ i σ.

U slučaju više dimenzionalnog skupa podataka, formula multivarijatne (višedimenzijske) Gaussove razdiobe glasi:

$$\mathcal{N}(X;\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^k|\Sigma|}} e^{-\frac{1}{2}(X-\mu)^T\Sigma^{-1}(X-\mu)} \tag{3.2}$$

U tom je slučaju k broj dimenzija skupa podatka, μ k-dimenzionalni vektor srednjih vrijednosti, a Σ kovarijacijska matrica veličine $k \times k$. Kovarijacijska matrica prikazuje, osim varijance svake dimenzije podatka, i odnos između različitih dimenzija.



Slika 3.4: Funkcija gustoće vjerojatnosti Gaussove distrubucije za različite vrijednosti parametara μ i σ. Preuzeto s [41]

Gaussova je mješavina funkcija koja se sastoji od onoliko Gaussovih funkcija koliko ima grupa u skupu podataka. Broj grupa K jest hiperparametar algoritma. Svaka Gaussova funkcija u mješavini ima sljedeće parametare:

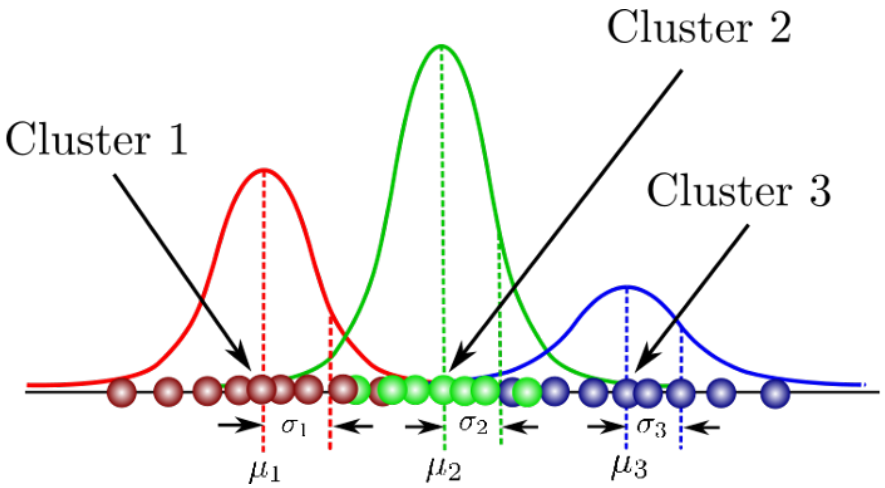
- Srednju vrijednost μ koja definira središte.
- Kovarijancu Σ koja određuje "širinu" funkcije.
- Vjerojatnost miješanja π koja definira vjerojatnost pripadanja primjera toj distribuciji.

Slika 3.5 prikazuje kako model Gaussove mješavine grupira podatke u slučaju kada imamo tri grupe.

Model Gaussove mješavine radi na načelu generativnog modeliranja i pretpostavlja se da se ulazni podaci ravnaju po Gaussovoj distribuciji. Iako to nije uvijek slučaj, centralni granični teorem iz statistike kaže da ako se prikuplja sve više i više uzoraka iz skupa podataka, oni imaju tendenciju nalikovati Gaussovoj funkciji, čak i kad izvorna distribucija skupa podataka nije Gaussova.

Cilj ovog modela jest pronaći parametre Gaussove funkcije koji maksimiziraju vjerojatnost dobivanja tih podataka. Svaku točku promatra se kao mješavinu više različitih Gaussovih funkcija te ona ima vjerojatnost:

$$p(x_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k) \tag{3.3}$$



Slika 3.5: Primjer Gaussove mješavine za $K = 3$. Preuzeto s [6]

$$\sum_{k=1}^{K} \pi_k = 1 \tag{3.4}$$

Jednadžba 3.3 govori da je određena točka x_i linearna kombinacija K Gaussovih funkcija. $p(x_i)$ još se naziva i izglednost primjera x_i (engl. *likelihood*). Varijabla π_k predstavlja vjerojatnost miješanja te Gaussove funkcije, odnosno njenu jačinu. Ograničenje vjerojatnosti miješanja jest da njihova suma mora biti 1, kao što se vidi u jednadžbi 3.4. Cilj algoritma jest maksimizirati logaritamsku izglednost svih podataka u skupu veličine N koja glasi:

$$\ln p(X; \mu, \Sigma, \pi) = \sum_{i=1}^{N} p(x_i) \tag{3.5}$$

Potrebno je za svaku Gaussovu funkciju odrediti vrijednosti vjerojatnosti miješanja π_k, srednje vrijednosti μ_k i kovarijance Σ_k, takve da maksimiziraju izraz 3.5. Vrijednosti ovih parametara dobiju se primjenom algoritma maksimizacije očekivanja (engl. *Expectation-Maximization Algorithm - EM*). Taj algoritam ima dva koraka:

1. Očekivanje

   U prvoj iteraciji algoritma inicijaliziraju se vrijednosti parametara slučajnim odabirom ili pomoću rezultata grupiranja nekog drugog algoritma. Zatim se u ovom koraku računa vjerojatnost da je svaki podatak stvoren svakom od K Gaussovih funkcija. Vjerojatnost da je primjer x_i stvoren pomoću Gaussove funkcije j računa se kao:

$$W_{ij} = \frac{\pi_j \mathcal{N}(x_i; \mu_j, \Sigma_j)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)} \tag{3.6}$$

Dijeljenje s $p(x_i)$ provodi se radi normalizacije, odnosno kako bi suma svih vjerojatnosti bila 1. U algoritmu K-sredina ovaj korak algoritma odgovara podjeli primjera u grupe.

2. Maksimizacija

   Korak maksimizacije ažurira vrijednosti vjerojatnosti miješanja, srednje vrijednosti i kovarijance na sljedeći način:

$$\pi_j = \frac{1}{N}\sum_{i=1}^{N} W_{ij} \tag{3.7}$$

$$\mu_j = \frac{\sum_{i=1}^{N} W_{ij} x_i}{\sum_{i=1}^{N} W_{ij}} \tag{3.8}$$

$$\Sigma_j = \frac{\sum_{i=1}^{N} W_{ij}(x_i - \mu_j)(x_i - \mu_j)^T}{\sum_{i=1}^{N} W_{ij}} \tag{3.9}$$

   Vjerojatnost π_j računa se kao vjerojatnost da je svaka točka stvorena Gaussovom funkcijom j podijeljeno s ukupnim brojem točaka. Parametri μ_j i Σ_j računaju se kao srednja vrijednost i kovarijacija svih točaka koje su pomnožene s vjerojatnošću W_{ij}. Ovaj korak analogan je pomicanju centroida grupa u algoritmu K-sredina.

Ova dva koraka ponavljaju se do konvergencije logaritamske izglednosti podataka, a pseudokod je dan kao pseudokod 3.3.

Pseudokod 3.3: Pseudokod modela Gaussove mješavine

```
definiraj broj grupa K
inicijaliziraj parametre π_k, μ_k i Σ_k, k = 1, ..., K
ponavljaj
   za svaku točku x_i ∈ N
      za svaku distribuciju k ∈ K
          izračunaj vjerojatnost W_ik da je x_i nastao iz k
   za svaku distribuciju k ∈ K
      ažuriraj vjerojatnost miješanja π_k
      ažuriraj srednju vrijednost μ_k
      ažuriraj kovarijancu Σ_k
dok ln p(X; μ, Σ, π) ne konvergira
```

Model Gaussove mješavine model je mekog grupiranja i svakoj točki dodijeljuje vektor vjerojatnosti pripadanja svakoj grupi. Iako sporije konvergira od algoritma K-sredina, za razliku od njega može se koristiti i na malim skupovima

podataka te kada grupe nisu jasno razdvojene. Također, može pronaći grupe različitih oblika i otporan je na prisutnost anomalija ili loše definiranih podataka.

4. Korišteni skupovi podataka

4.1. Otkrivanje prijevare s kreditnim karticama

Važno je da banke mogu prepoznati neobične transakcije kreditnim karticama kako bi se otkrila prijevara i spasili novci korisnika. Prvi skup podataka koristi se za otkrivanje prijevara kreditnim karicama (engl. *Credit Card Fraud Detection*) i preuzet je s [28]. Sadrži transakcije koje su europski vlasnici kartica izvršili kreditnim karticama u rujnu 2013. godine u 2 dana. Skup podataka izrazito je neuravnotežen te prijevare čine samo 0,172% svih transakcija.

Skup podataka sadrži numeričke značajke koje su rezultat analize glavnih komponenata (engl. *Principal component analysis*). Zbog povjerljivosti nisu dostupni originalni podaci i imena značajki već samo rezultat PCA transformacije. Dobivene glavne komponente su značajke V1, V2, ..., V28. Značajke koje nisu transformirane i koje su zadržale svoje vrijednosti i ime su značajke *Vrijeme* i *Iznos*. Značajka *Vrijeme* sadrži sekunde koje su protekle između svake transakcije i prve transakcije u skupu podataka, a značajka *Iznos* je iznos transakcije. Na kraju, značajka *Klasa* je varijabla oznake koja poprima vrijednost 1 u slučaju prijevare, a vrijednost 0 pri normalnim transakcijama.

Skup podataka koji se koristi već je obrađen, maknuta je značajka *Vrijeme* jer ne pomaže u problemu detekcije prijevara i provedena je normalizacija podataka.

4.2. Detekcija upada u mrežu

Skup podataka *KDD Cup 1999* umjetno je stvoren skup koji se može koristiti za detekciju mrežnog upada ili napada. Taj skup podataka korišten je za *Treće međunarodno natjecanje u otkrivanju znanja i alatima za rudarenje podataka*. Zadatak natjecanja bio je izgraditi prediktivni model koji će detektirati upade u mrežu i razlikovati ih od normalnih veza. Svaka veza je slijed TCP paketa koji počinje i završava u nekom definiranom vremenu, u kojem se podaci šalju od

izvorne IP adrese na ciljnu IP adresu prema nekom definiranom protokolu.

Ovaj skup podataka sadrži standardni skup podataka za provjeru modela i preuzet je s [28]. Nastao je tako što je *Lincoln Labs* postavio okolinu za prikupljanje neobrađenih TCP podataka za lokalnu mrežu (LAN) tijekom devet tjedana simulirajući tipični LAN američkih zračnih snaga. Upravljali su LAN-om kao da se radi o pravoj okolini zračnih snaga uz dodatak višestrukih napada. Prikupljeno je oko 7 milijuna zapisa veza, pri čemu svaka veza ima 41 značajku i označena je kao normalna veza ili kao napad, s točno specificiranom vrstom napada. U skupu za treniranje korišteno je ukupno 24 različita tipa napada.

Za potrebe ovog rada korištena su dva podskupa skupa podataka *KDD Cup 1999*: *U2R* i *Probe*.

Napad *U2R* (engl. *User to Root*) jest napad u kojem napadač na početku pristupa normalnom korisničkom računu, a kasnije dobiva pristup korijenskom korisniku (engl. *root*) iskorištavanjem ranjivosti sustava ili greške korisnika. *Probe* napad skenira mrežu sa svrhom praćenja mreže ili prikupljanja podataka o mreži i mrežnoj aktivnosti.

Novi skupovi podataka izvedeni su iz skupa podataka *KDD Cup 1999* koristeći *U2R* ili *Probe* napad kao anomaliju naspram normalnih veza. Ti skupovi su znatno manji i broj značajki smanjen je na 6 osnovnih:

1. Vrsta transportnog protokola
2. Vrsta aplikacijskog protokola
3. Zastavica (status povezivanja)
4. Prijava uspješna
5. Prijava ostvarena kao domaćin
6. Prijava ostvarena kao gost

Prve tri značajke su kategoričkog tipa, a druge tri tipa boolean (točno/netočno).

4.3. Detekcija raka

Jedna od važnih primjena otkrivanja anomalija jest u medicini gdje one mogu signalizirati pojavu bolesti. Detekcija anomalija korisiti se za analizu medicinskih

slika kako bi se otkrile abnormalne stanice ili tumori. Primjer toga jest predikcija je li tumor dojke dobroćudan ili zloćudan i za tu svrhu korišten je skup podataka preuzet s [10].

Svaki primjer skupa podataka ima 32 značajke, među kojima su i značajke *Identifikacijski broj* i *Dijagnoza*, koja klasificira tumor kao dobroćudni ili zloćudni. Preostale značajke su zračunate iz digitalizirane slike aspiracijske biopsije finom iglom (FNA) tkiva dojke i opisuju karakteristike staničnih jezgri prisutnih na slici. Za svaku jezgru izračunato je deset vrijednosti kao što su radijus, tekstura, područje i glatkoća. Značajke su zatim dobivene iz tih podataka kao srednja vrijednost, standardna pogreška i najgora (odnosno najveća) vrijednost mjerenja.

Pregled svih korištenih skupova podataka dan u tablici 4.1.

Tablica 4.1: Podaci o korištenim skupovima podataka.

Skup podataka	**Broj primjera**	**Dimenzionalnost**	**Udio anomalija**
Kreditne kartice	284 807	29	0.17%
U2R	60 821	6	0.38%
Probe	64 759	6	6.88%
Rak dojke	569	32	37.26%

5. Programsko ostvarenje

5.1. Obrada podataka

Na početku programskog ostvarenja provedena je obrada i priprema podataka kako bi se nad njima mogli koristiti algoritmi grupiranja.

Prvi skup podataka, prijevare s kreditnim karticama, nije bilo potrebno mijenjati niti obraditi budući da je na njemu već prethodno provedena normalizacija i maknuta je značajka *Vrijeme*.

Skupovi podataka *U2R* i *Probe* sadrže kategoričke značajke koje su se morale kodirati jer svi navedeni algoritmi rade isključivo s numeričkim podacima. Provedeno je *One-hot* kodiranje u kojem se svaku kategoričku značajku prikazuje vektorom binarnih vrijednosti. Svaka pozicija u vektoru označava jednu vrijednost te značajke i svaka značajka ima jedinicu samo na mjestu koje odgovara njezinoj vrijednosti te nulu na svim ostalim mjestima. Time se broj dimenzija podataka poveća za broj svih različitih vrijednosti značajki. Budući da značajka *Vrsta aplikacijskog protokola* može poprimiti 65 različitih vrijednosti, odabrano je 5 najčešćih vrijednosti dok su preostale označene kao *ostalo*. Najčešće vrijednosti u oba skupa podataka su: *http*, *private*, *smtp*, *domain_u* i *ftp_data*. Značajka *Zastavica* poprima ukupno 9 različitih vrijednosti, od čega 94.86% u jednom, odnosno 99.51% podataka u drugom skupu ima vrijednost *SF*. Zbog toga je samo ta vrijednost zadržana, a preostale su označene s *ostalo*. Značajka *Vrsta transportnog protokola* ima samo tri moguće vrijednosti (*tpc*, *udp* i *icmp*) te ovdje reduciranje broja vrijednosti nije bilo potrebno.

U obradi skupa podataka tumora dojke maknute su značajke *Identifikacijski broj* i *Dijagnoza* budući da nam prva ne daje nikakvu informaciju, dok nam druga daje rješenje problema grupiranja. Zbog toga se značajka *Dijagnoza* koristi u vrednovanju kao oznaka grupe kojoj primjer pripada. Također, provedena je normalizacija podataka koristeći robusno skaliranje (engl. *Robust Scaler*). Takvo skaliranje uklanja medijan i skalira podatke prema interkvartilnom rasponu (engl.

Interquartile Range - IQR), što je raspon između 25. i 75. kvantila. Korištena je ova vrsta skaliranja jer je najotpornija na stršeće vrijednosti te anomalije ostaju prisutne i nakon skaliranja.

5.2. Smanjenje broja uzoraka

Zbog ograničene snage računalnog procesora bilo je potrebno smanjiti broj uzoraka za neke skupove podataka.

Za svaki skup podataka definirao se udio normalnih (negativnih) primjera i udio anomalija (pozitivnih primjera) koji se želi zadržati u skupu. Kako bi rezultati bili što vjerodostojniji, skup podataka je svaki put prvo izmiješan te su primjeri koji ostaju izabrani slučajnim odabirom.

U skupu podataka o kreditnim karticama zadržano je 10% normalnih primjera i sve anomalije, kako bi se povećao udio anomalija u skupu. U slučaju detekcije upada u mrežu u oba skupa podataka *U2R* i *Probe* ispitivanje je provedeno nad 50% normalnih primjera i 50% anomalija.

Skup podataka za detekciju raka nije zahtijevao smanjenje broja uzoraka budući da je skup sam po sebi malen. Međutim, provedeno je smanjenje broja anomalija u skupu jer je njihov originalni udio bio prevelik da bi se takvi primjeri smatrali odstupanjima. Iz tog je razloga zadržano 10% anomalija i svi normalni primjeri.

U tablici 5.1 dan je pregled svih korištenih skupova podataka nakon njihove obrade i smanjenja broja primjera.

Tablica 5.1: Korišteni skupovi podataka nakon obrade.

Skup podataka	**Broj primjera**	**Dimenzionalnost**	**Udio anomalija**
Kreditne kartice	28 924	29	1.70%
U2R	30 410	14	0.37%
Probe	32 379	14	6.43%
Rak dojke	378	30	5.56%

5.3. Algoritmi i ostvarenje detekcije anomalija

Za algoritme K-sredina, DBSCAN i Gaussovu mješavinu, kao i za sve metrike, korištena su programska ostvarenja iz biblioteke *scikit-learn* [31].

Algoritam K-sredina očekuje hiperparametar K, odnosno broj grupa. Budući da u korištenim skupovima podataka uvijek postoji samo jedna grupa, koja sadrži određeni postotak odstupanja, broj grupa fiksiran je na 1. Algoritam je isproban i za veći broj grupa, međutim za svaki skup podataka dobije se najbolje rješenje kada postoji samo jedna grupa. Zadatak algoritma jest računanje vrijednosti centroida te grupe nakon čega se računa udaljenost svake točke od centroida. One točke koje su najudaljenije smatraju se anomalijama, a granična udaljenost dobiva se kao određeni percentil distrubucije svih udaljenosti. Odabrani percentil najčešće ovisi o očekivanom udjelu anomalija u skupu podataka. Na primjer, ako se očekuje oko 10% anomalija u skupu, uzima se 90. percentil kao prag udaljenosti. To znači da će 10% točaka s najvećom udaljenošću biti označeno kao anomalije.

Algoritam DBSCAN sam po sebi otkriva anomalije jer proglašava anomalijom svaku točku koja nije ni središnja ni granična točka. Algoritam zahtijeva dva hiperparametra: minimalan broj točaka *minPts* i udaljenost ϵ. Njihovo određivanje pokazalo se kao težak zadatak te je za svaki skup podataka provedeno pretraživanje po rešetci (engl. *Grid Search*). Na taj način otkrivena je kombinacija parametara za koju algoritam dalje najbolju detekciju anomalija. Budući da DBSCAN algoritam sam određuje broj grupa, on može biti i veći od 1. U tom je slučaju prije vrednovanja provedeno označavanje točaka gdje je svaka točka koja nije prepoznata kao anomalija označena kao normalna točka. Ovaj korak nije nužan i zapravo mijenja rezultat grupiranja, ali je proveden kako bi se vredovanje fokusiralo na detekciju anomalija, a ne na otkrivanje grupa.

Za model Gaussove mješavine potrebno je definirati broj grupa K, odnosno broj Gaussovih komponenata u skupu podataka. Kao i u slučaju algoritma K-sredina, model Gaussove mješavine ispitan je za različite brojeve komponenata, ali najbolji rezultat ostvaren je kada postoji samo jedna grupa. Korišten je model koji ima sferni tip kovarijance, u kojem svaka komponenta ima vlastitu varijancu koja je ista po svim osima te se zato dobije kružni oblik u grafičkom prikazu. Za inicijalizaciju srednjih vrijednosti, kovarijance i vjerojatnosti miješanja korišten je algoritam K-sredina. Ti su parametri izabrani jer je za njih dobivena najtočnija detekcija anomalija. Algoritam za svaki primjer računa njegovu logaritamsku

izglednost u tom skupu podataka te se primjeri s najmanjom izglednošću proglašavaju anomalijama. Granična logaritamska izglednost uzeta je kao određeni percentil svih logaritamskih izglednosti, koji također ovisi o očekivanom udjelu anomalija. Ako se očekuje 10% anomalija, 10. percentil uzima se kao granični i sve točke s manjom logaritamskom izglednošću od njegove označuju se kao anomalije.

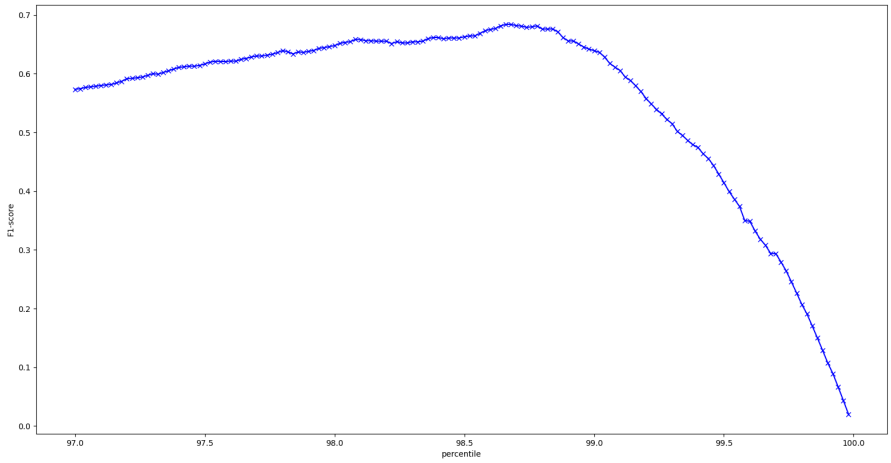
Prilikom određivanja hiperparametara algoritma i graničnih vrijednosti odabrane su one vrijednosti koje maksimiziraju F1-mjeru.

6. Rezultati

6.1. Anomalije u kartičnim transakcijama

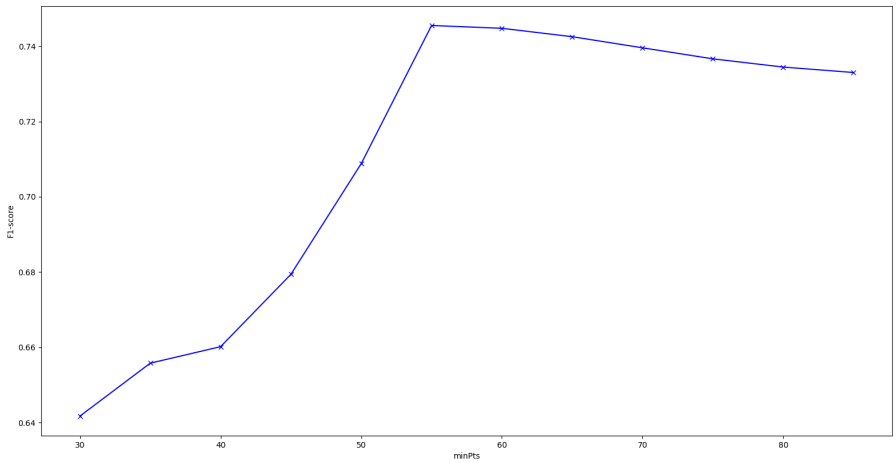
Prva usporedba algoritama grupiranja provedena je na skupu podataka o kreditnim karticama s ciljem detektiranja prijevara.

Za algoritam K-sredina unaprijed je određen hiperparametar K, odnosno broj grupa je fiksiran na 1. Bilo je potrebno odrediti granični percentil udaljenosti od centroida te je ovisnost F1-mjere o odabranom percentilu prikazana na slici 6.1. Kao granični percentil odabran je percentil 98.65 jer je za njega dobivena maksimalna vrijednost F1-mjere.



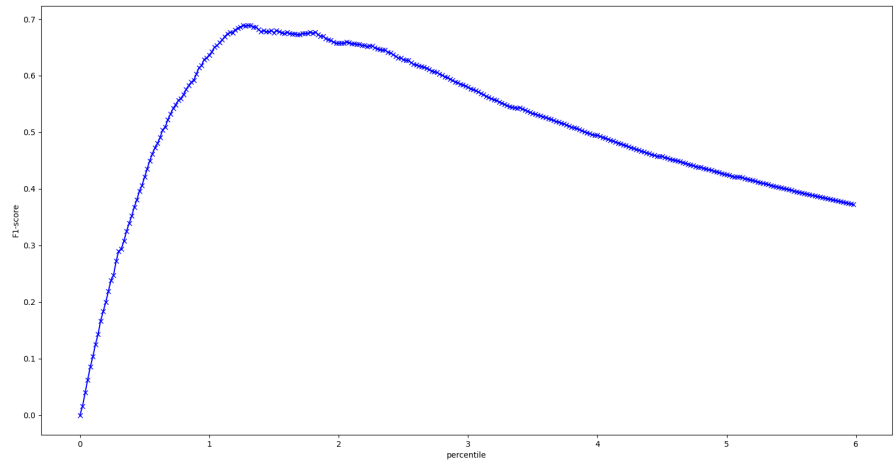
Slika 6.1: Graf ovisnosti F1-mjere o percentilu udaljenosti za algoritam K-sredina i problem detekcije kartičnih prijevara.

Kako bi se definirali hiperparametri algoritma DBSCAN, provedeno je pretraživanje po rešetci te su najveće vrijednosti F1-mjere dobivene za $\epsilon = 0.25$ i $minPts = 55$. Slika 6.2 prikazuje promjenu F1-mjere ovisno o minimalnom broju točaka $minPts$ uz fiksnu udaljenost $\epsilon = 0.25$.



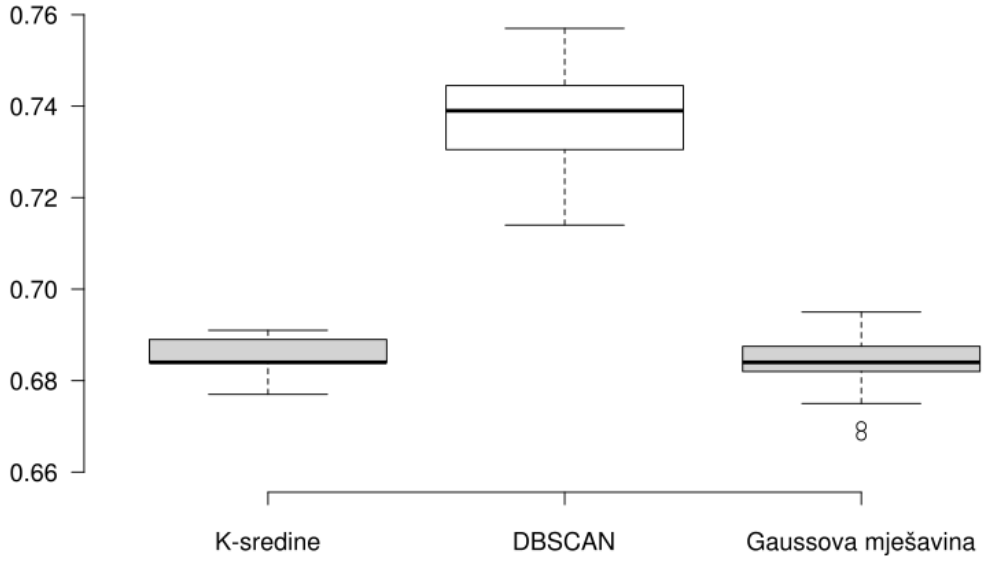
Slika 6.2: Graf ovisnosti F1-mjere o minimalnom broju točaka $minPts$ uz $\epsilon = 0.25$ za algoritam DBSCAN i problem detekcije kartičnih prijevara.

U slučaju modela Gaussove mješavine broj komponenata unaprijed je postavljen na 1 te je bilo potrebno odrediti percentil logaritamske izglednosti ispod kojeg se točke proglašavaju anomalijiama. Ovisnost F1-mjere o tom percentilu prikazana je na slici 6.3 te je percentil 1.35 uzet kao granični percentil. Slučajno je i u algoritmu K-sredina i u modelu Gaussove mješavine odabran granični percentil koji proglašava jednak postotak primjera anomalijama, odnosno 1.35%.



Slika 6.3: Graf ovisnosti F1-mjere o percentilu logaritamske izglednosti za model Gaussove mješavine i problem detekcije kartičnih prijevara.

Svaki algoritam pokrenut je 20 puta na svakom skupu podataka. Na slici 6.4 prikazana je grafička usporedba uspješnosti svih triju algoritama na osnovi dobivenih F1-mjera, a u tablici 6.1 dana je usporedba algoritama po svim metrikama, pri čemu je za svaku metriku uzeta srednja vrijednost u 20 mjerenja.



Slika 6.4: Usporedba F1-mjere algoritama na problemu detekcije kartičnih prijevara.

Tablica 6.1: Usporedba algoritama na skupu podataka o kreditnim karticama.

	K-sredine	**DBSCAN**	**Gaussova mješavina**
Preciznost	**0.774**	0.721	0.771
Odziv	0.615	**0.753**	0.613
F1-mjera	0.685	**0.737**	0.683
AUC mjera	0.806	**0.874**	0.805
Koeficijent siluete	**0.653**	0.611	**0.653**
Davies-Bouldin indeks	**1.212**	1.414	1.215

Iz grafičkog prikaza, kao i iz tablice rezultata, vidi se da je algoritam DBSCAN ostvario najbolju detekciju anomalija na skupu podataka o kreditnim karticama, dok su algoritmi K-sredina i Gaussove mješavine ostvarili približno isti rezultat. Također, F1-mjera rješenja algoritma DBSCAN više se razlikuje između različitih izvođenja nego F1-mjera rješenja preostala dva algoritma, odnosno rješenja

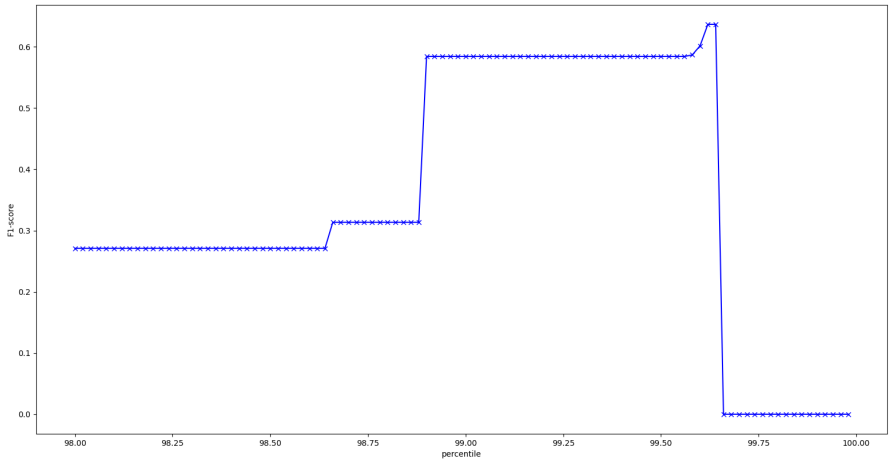
algoritma DBSCAN više variraju. Međutim, usprkos dobroj detekciji i najvećem iznosu F1-mjere, metrike koje ne zahtijevaju oznake podataka ocijenile su algoritam K-sredina i Gaussovu mješavinu kao uspješnije.

6.2. Otkrivanje anomalija u mreži

U sljedeća dva skupa podataka zadatak je bio prepoznati mrežni napad kao anomaliju. Algoritmi su uspoređeni na skupovima podataka koji sadrže *U2R* i *Probe* mrežne napade.

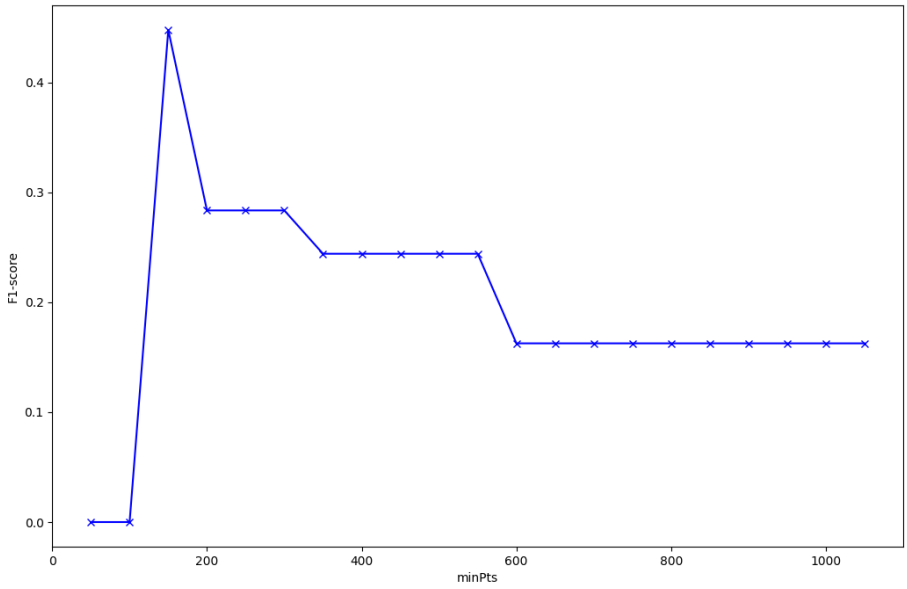
6.2.1. Napad *U2R* kao anomalija

U slučaju skupa podataka *U2R* za algoritam K-sredina određen je percentil 99.62 kao granični percentil udaljenosti od centroida jer on maksimizira vrijednost F1-mjere, kao što se vidi na slici 6.5.

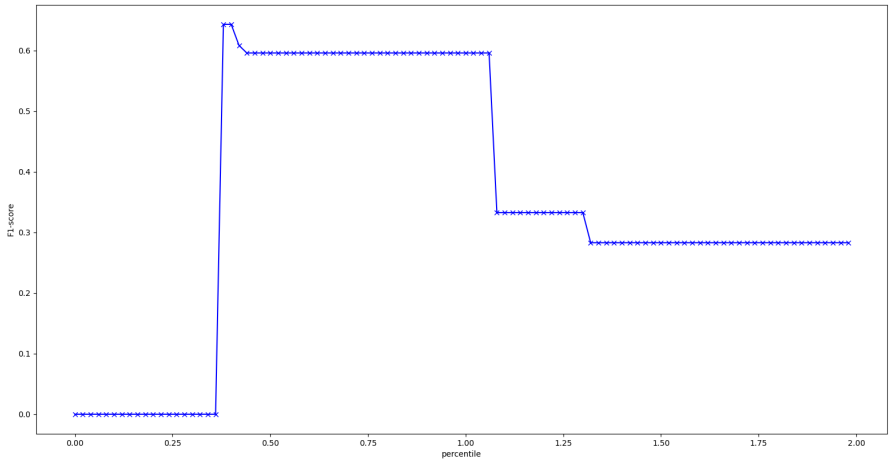


Slika 6.5: Graf ovisnosti F1-mjere o percentilu udaljenosti za algoritam K-sredina i skup podataka *U2R*.

Za algoritam DBSCAN izabrani su hiperparametri $\epsilon = 0.1$ i $minPts = 150$. Na slici 6.6 vidi se da je F1-mjera najveća za $minPts = 150$, iako ni za jednu vrijednost hiperparametra $minPts$ nije dobiven zadovoljavajuć iznos F1-mjere, koji bi bio iznad 0.5. Mijenjanje udaljenosti ϵ nije imalo utjecaja na rezultat jer bi nakon takve promjene algoritam ili prepoznao cijeli skup kao anomaliju ili ne bi pronašao ni jednu anomaliju u skupu. Tome je mogući razlog primjena *One-hot*



Slika 6.6: Graf ovisnosti F1-mjere o minimalnom broju točaka $minPts$ uz $\epsilon = 0.1$ za algoritam DBSCAN i skup podataka *U2R*.



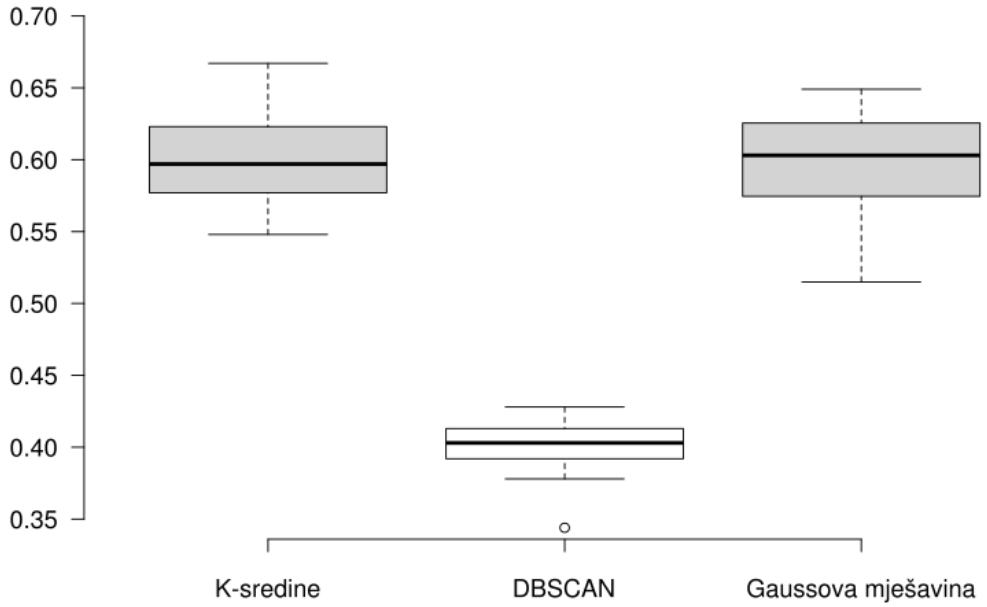
Slika 6.7: Graf ovisnosti F1-mjere o percentilu logaritamske izglednosti za model Gaussove mješavine i skup podataka *U2R*.

kodiranja zbog kojeg velik broj značajki ima vrijednost 0 te se udaljenosti među podacima čine manjima.

Slika 6.7 prikazuje promjenu F1-mjere modela Gaussove mješavine ovisno o odabranom graničnom percentilu logaritamske izglednosti te je, sukladno slici,

odabran percentil 0.38. Time je i u algoritmu K-sredina i u modelu Gaussove mješavine odabran postotak anomalija koji upravo odgovara stvarnom udjelu napada u skupu podataka.

Algoritmi su vrednovani pomoću F1-mjere i na slici 6.8 dan je grafički prikaz rezultata nakon 20 izvođenja. U tablici 6.2 prikazane su srednje vrijednosti svih mjerenih metrika.



Slika 6.8: Usporedba F1-mjere algoritama na problemu detekcije mrežnog napada *U2R*.

Tablica 6.2: Usporedba algoritama na skupu podataka s mrežnim napadom *U2R*.

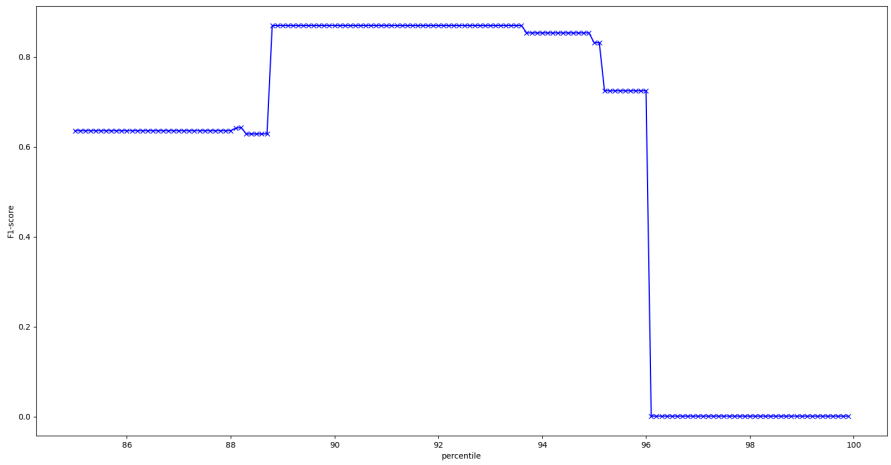
	K-sredine	**DBSCAN**	**Gaussova mješavina**
Preciznost	**0.609**	0.266	0.603
Odziv	0.591	**0.832**	0.596
F1-mjera	**0.600**	0.403	0.599
AUC mjera	0.795	**0.911**	0.797
Koeficijent siluete	**0.526**	0.467	**0.526**
Davies-Bouldin indeks	**0.552**	1.699	0.585

U slučaju detekcije mrežnog napada *U2R* algoritmi K-sredina i Gaussove mješavine ostvarili su značajno bolji rezultat nego algoritam DBSCAN. Iako je on

postigao puno veći odziv, što je često u problemima detekcije anomalija i poželjno, po vrijednostima ostalih metrika zaostaje za druga dva algoritma. U ovom primjeru vidljivo je i da AUC mjera nije uvijek mjerodavna prilikim otkrivanja anomalija, budući da algoritam DBSCAN ima najveći iznos AUC mjere, a najlošiju sveukupnu izvedbu. Druga dva algoritma ostvarila su sličan rezultat, pri čemu je algoritam K-sredina imao malo bolju izvedbu od modela Gaussove mješavine.

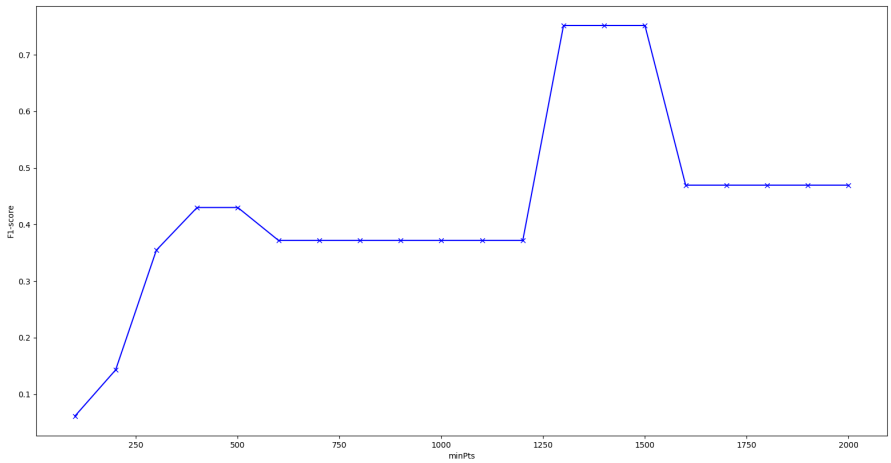
6.2.2. Napad *Probe* kao anomalija

Za skup podataka koji kao anomaliju imaju mrežni napad *Probe* slika 6.9 prikazuje promjenu F1-mjere algoritma K-sredina pri promjeni graničnog percentila udaljenosti od centroida. Kao granični percentil odabran je percentil 93.5, iako je to mogao biti i bilo koji u rasponu $[89, 93.5]$ jer za sve njih F1-mjera postiže maksimum. Međutim, odabran je najveći percentil kako bi što manji broj primjera bio označen kao anomalija.

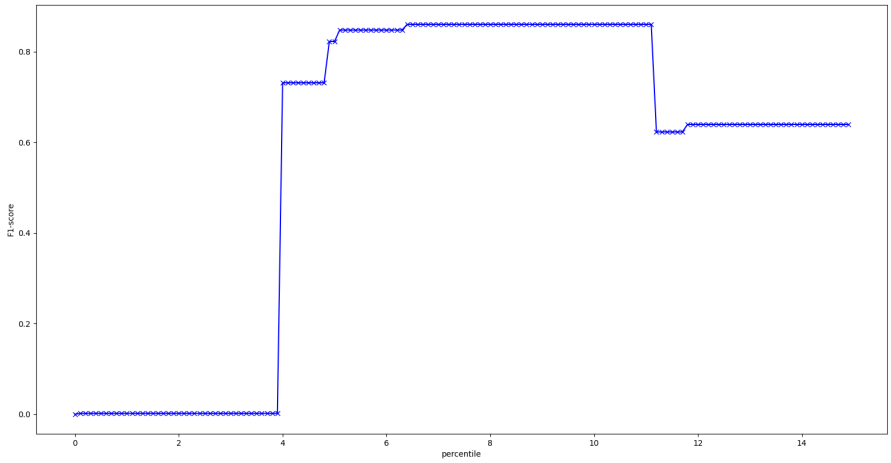


Slika 6.9: Graf ovisnosti F1-mjere o percentilu udaljenosti za algoritam K-sredina i skup podataka *Probe*.

Kao i kod skupa podataka s napadom *U2R*, manje promjene udaljenosti ϵ nisu imale utjecaja na rezultate algoritma DBSCAN, dok su veće promjene imale neželjeni učinak, pa je udaljenost fiksirana na $\epsilon = 0.1$. Slika 6.10 prikazuje ovisnost F1-mjere o minimalnom broju točaka $minPts$ te je odabrano $minPts = 1300$, jer se za tu vrijednost postiže maksimum F1-mjere.



Slika 6.10: Graf ovisnosti F1-mjere o minimalnom broju točaka $minPts$ uz $\epsilon = 0.1$ za algoritam DBSCAN i skup podataka *Probe*.

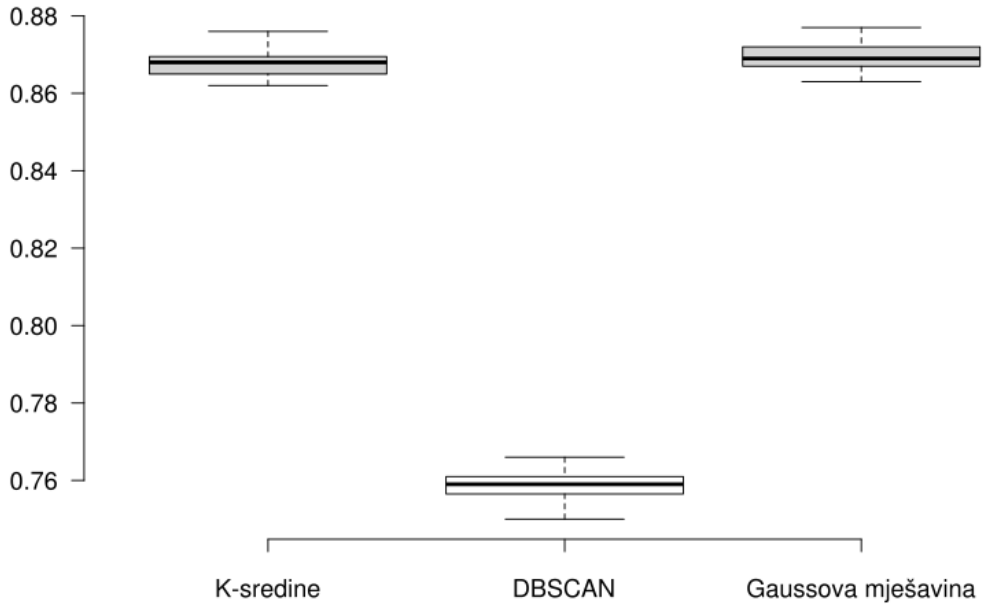


Slika 6.11: Graf ovisnosti F1-mjere o percentilu logaritamske izglednosti za model Gaussove mješavine i skup podataka *Probe*.

Slika 6.11 prikazuje promjenu F1-mjere modela Gaussove mješavine ovisno o graničnom percentilu logaritamske izglednosti. Vidi se da dobiveni graf sliči zrcalnoj slici grafa za algoritam K-sredina te se, isto kao i kod tog algoritma, maksimum postiže za više vrijednosti, odnosno za svaki percentil iz skupa $[6.5, 11]$. Kao i kod K-sredina, izabran je percentil 6.5 kako bi se što manji udio podataka proglasio anomalijom.

Rezultati F1-mjere svih algoritama nakon 20 izvođenja prikazani su na slici

6.12, a u tablici 6.3 dana je usporedba srednjih vrijednosti svih mjerenih metrika.



Slika 6.12: Usporedba F1-mjere algoritama na problemu detekcije mrežnog napada *Probe*.

Tablica 6.3: Usporedba algoritama na skupu podataka s mrežnim napadom *Probe*.

	K-sredine	**DBSCAN**	**Gaussova mješavina**
Preciznost	**0.881**	0.634	**0.881**
Odziv	0.859	**0.944**	0.858
F1-mjera	**0.870**	0.759	0.869
AUC mjera	0.925	**0.953**	0.925
Koeficijent siluete	**0.518**	0.497	**0.518**
Davies-Bouldin indeks	**1.031**	1.417	1.035

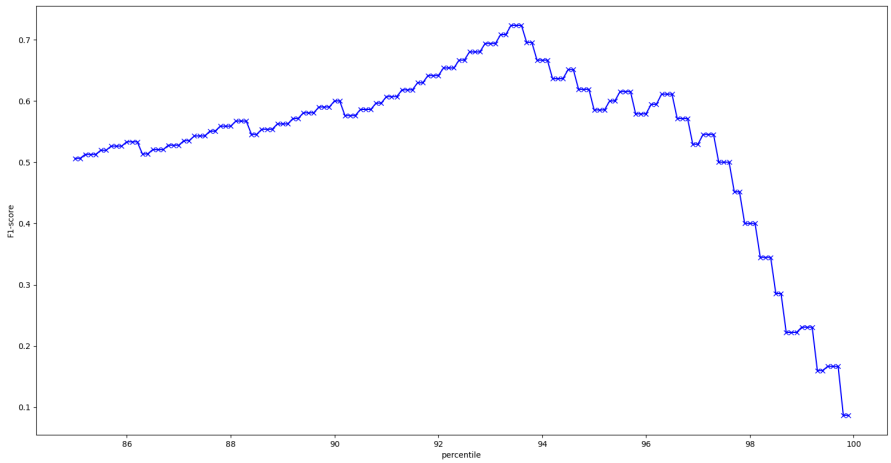
Rezultati detekcije napada *Probe* grafički nalikuju rezultatima detekcije napada *U2R*. Kao i tada, algoritmi K-sredina i Gaussove mješavine ostvarili su gotovo pa jednak rezultat i nadmašili su algoritam DBSCAN. Međutim, ovaj je put razlika između algoritma DBSCAN i ostalih algoritama manja i svi su algoritmi postigli bolje rezultate nego na skupu podataka *U2R*. Tome je mogući razlog veći ukupni udio anomalija u ovom skupu podataka. Algoritam DBSCAN ponovno je

prvi po broju točno detektiranih anomalija, odnosno ima najveći odziv, ali zato ima manje vrijednosti ostalih metrika.

6.3. Detekcija raka kao anomalije

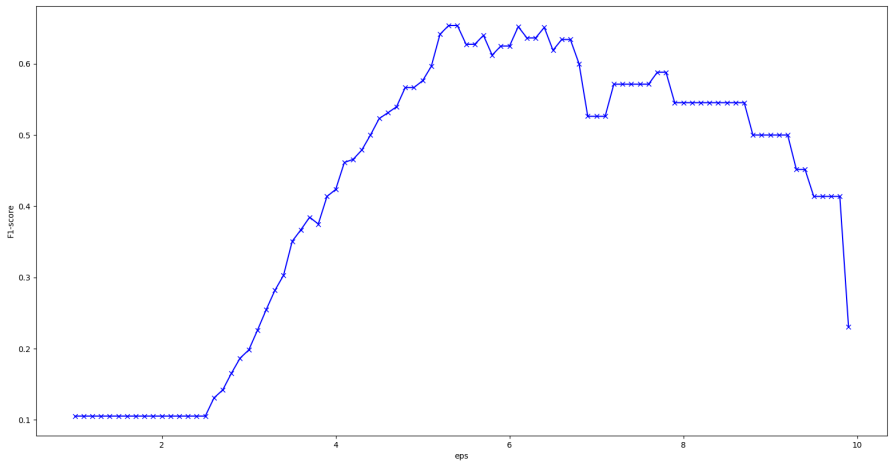
Cilj posljednjeg eksperimenta bio je detekcija zloćudnog tumora na malom skupu podataka koji sadrži mjerenja više dobroćudnih i zloćudnih tumora dojke.

Za otkrivanje anomalija pomoću algoritma K-sredina odabran je granični percentil udaljenosti 93.5 jer je za njega dobivena najbolja detekcija anomalija, kao što se vidi na slici 6.13.

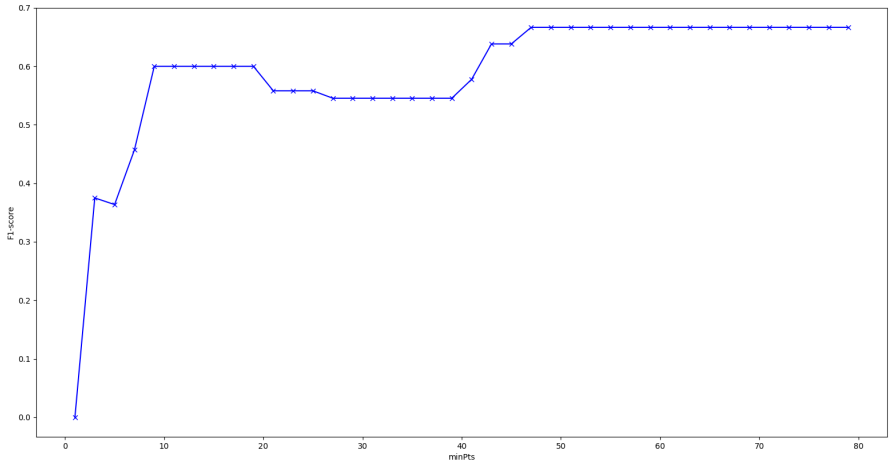


Slika 6.13: Graf ovisnosti F1-mjere o percentilu udaljenosti za algoritam K-sredina i problem detekcije raka.

Za algoritam DBSCAN odabran je minimalni broj točaka $minPts$ po pravilu $minPts = 2 * D$, gdje je broj dimenzija D jednak 30. Iz slike 6.14 vidi se da optimalna udaljenost uz $minPts = 60$ iznosi $\epsilon = 5.3$. Slika 6.15 prikazuje i promjenu F1-mjere za različite vrijednosti minimalnog broja točaka $minPts$ uz $\epsilon = 5.3$ te je vidljivo da se maksimalna vrijednost F1-mjere postiže, među ostalim, i za $minPts = 60$. Isto ispitivanje provedeno je i za druge početne vrijednosti hiperparametra $minPts$ i najbolji rezultati ostvareni su korištenjem ove kombinacije hiperparametara algoritma. Takva detaljna analiza nije provedena za preostale skupove podataka zbog zahtjevnosti izvođenja algoritma DBSCAN na velikim skupovima podataka, naročito uz veće variranje udaljenosti ϵ.



Slika 6.14: Graf ovisnosti F1-mjere o udaljenosti ϵ uz $minPts = 60$ za algoritam DBSCAN i problem detekcije raka.

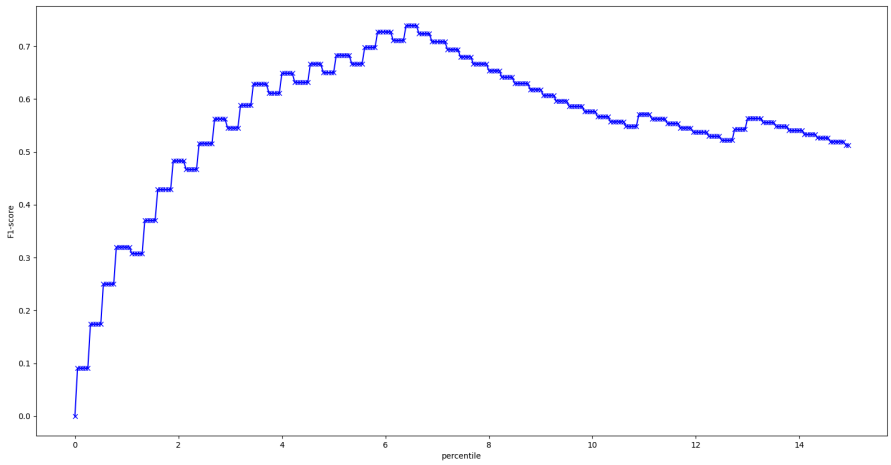


Slika 6.15: Graf ovisnosti F1-mjere o minimalnom broju točaka $minPts$ uz $\epsilon = 5.3$ za algoritam DBSCAN i problem detekcije raka.

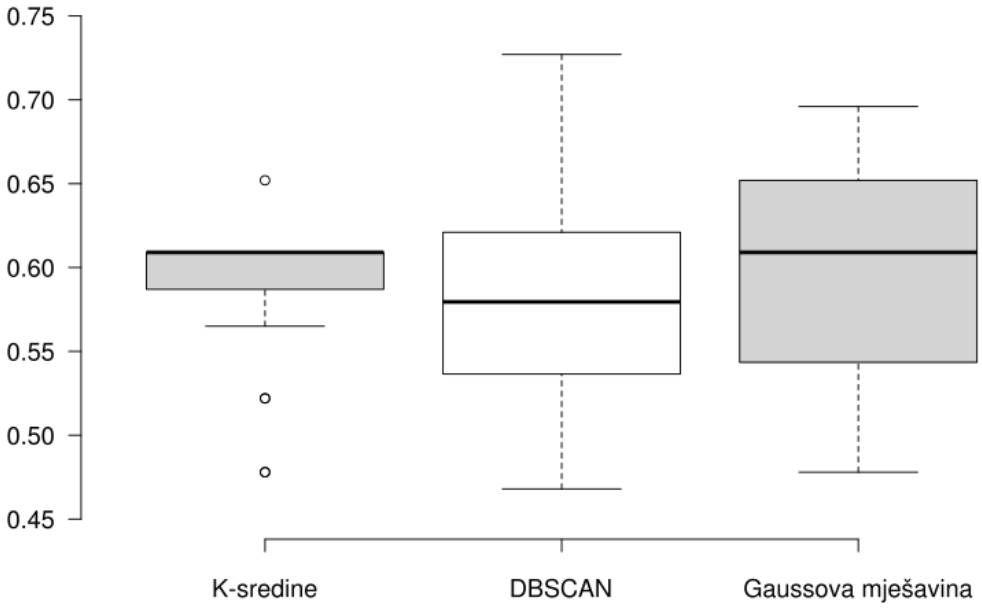
Ovisnost F1-mjere o graničnom percentilu logaritamske izglednosti za model Gaussove mješavine prikazana je na slici 6.16 te je kao granični percentil odabrana vrijednost 6.5.

Grafička usporedba rezultata F1-mjere nakon 20 pokretanja svakog algoritama dana je na slici 6.17. Tablica 6.4 sadrži usporedbu algoritama na osnovi srednjih vrijednosti svih mjerenih metrika.

Na problemu detekcije raka sva tri algoritma ostvarila su vrlo slične rezultate,



Slika 6.16: Graf ovisnosti F1-mjere o percentilu logaritamske izglednosti za model Gaussove mješavine i problem detekcije raka.



Slika 6.17: Usporedba F1-mjere algoritama na problemu detekcije raka.

što je vidljivo i iz grafičkog prikaza i iz srednjih vrijednosti različitih metrika. Gledajući F1-mjeru, model Gaussove mješavine daje nešto bolje rezultate od preostala dva algoritma. Algoritam DBSCAN ima najveću srednju vrijednost odziva i postigao je najveću vrijednost F1-mjere od svih algoritama na ovom skupu podataka. Međutim, on također ima i najveću varijaciju vrijednosti F1-mjere u

Tablica 6.4: Usporedba algoritama na skupu podataka za detekciju raka.

	K-sredine	**DBSCAN**	**Gaussova mješavina**
Preciznost	0.544	0.503	**0.548**
Odziv	0.648	**0.712**	0.652
F1-mjera	0.591	0.589	**0.596**
AUC mjera	0.808	**0.836**	0.810
Koeficijent siluete	**0.591**	0.565	0.582
Davies-Bouldin indeks	1.434	1.517	**1.415**

različitim izvođenjima te mu je zato srednja vrijednost niža od ostalih. S druge strane, algoritam K-sredina ima najmanju promjenu F1-mjere između izvođenja, ali je i maksimalna vrijednost koju postiže najniža. Stoga bi u ovom slučaju odabir najboljeg algoritma ovisio o problemu koji se rješava i okolnostima izvođenja. Ponekad je mala varijacija rješenja pri više pokretanja poželjno svojstvo, ali postoje i slučajevi kada je bitnije od toga dobiti jedno odlično rješenje, čak i ako je za to potrebno više puta pokrenuti algoritam.

7. Zaključak

U ovom radu napravljena je usporedba nekoliko algoritama grupiranja pri rješavanju problema otkrivanja anomalija. Anomalije su opažanja koja se znatno razlikuju od drugih i čije otkrivanje može biti izuzetno važno radi proučavanja novog obrasca ponašanja ili izvlačenja određenih zaključaka.

Ispitane su mogućnosti detekcije anomalija tri različita algoritma grupiranja: algoritma K-sredina, algoritma DBSCAN i modela Gaussove mješavine. Za ispitivanje su korištena četiri skupa podataka koji u sebi sadrže određena odstupanja. Algoritmi su ispitani na problemu detekcije kartičnih prijevara, detekcije upada u mrežu, odnosno otkrivanja dva tipa mrežnog napada te na problemu detekcije zloćudnog tumora.

Algoritam DBSCAN postigao je daleko najbolji rezultat u slučaju otkrivanja anomalija u kartičnim transakcijama, dok su se algoritmi K-sredina i Gaussove mješavine na svim skupovima podataka vrlo slično ponašali. Nadmašili su algoritam DBSCAN na problemima detekcije mrežnih napada, a algoritam K-sredina imao je neznatno bolje vrijednosti metrika. U slučaju problema otkrivanja raka sva tri algoritma postigla su približno jednake rezultate, s time da je model Gaussove mješavine bio nešto bolji od preostala dva.

Iz ovih rezultata može se zaključiti da je algoritam DBSCAN prikladan za velike i normalizirane skupove podataka, veće dimenzionalnosti i s oko 1-2% anomalija. Algoritmi K-sredina i Gaussove mješavine općenito daju slične rezultate i uspješniji su od algoritma DBSCAN na velikim skupovima podataka nešto manje dimenzionalnosti koji sadrže *One-hot* kodirane značajke. Na manjem skupu podataka sva tri algoritma daju približno jednako dobra rješenja, pri čemu algoritam DBSCAN postiže najbolji ukupni rezultat. Međutim, algoritam DBSCAN ima i veliku varijaciju rješenja, dok algoritam K-sredina ima najmanju varijancu i uvijek daje približno isto rješenje.

Odabir algoritma grupiranja uvijek ovisi o problemu otkivanja anomalija koji se rješava i njegovim svojstvima te o okolnostima eksperimenta. U obzir se mora

uzeti veličina skupa podataka, dimenzionalnost, vrsta značajki i očekivani udio anomalija. Također, u ispitivanjima koja nije moguće ponavljati ili se moraju događati u stvarnom vremenu veliku ulogu igra i varijanca rješenja algoritma, koja bi tada trebala biti što niža.

Daljnja analiza algoritama grupiranja u postupcima detekcije anomalija može se provesti ispitivanjem na više novih skupova podataka, dodavanjem šuma u podatke ili korištenjem skupa podataka koji ima više od jedne grupe.

Literatura

[1] Anton Andrésen Adam Håkansson. Comparing unsupervised clustering algorithms to locate uncommon user behavior in public travel data. 2020. URL https://www.diva-portal.org/smash/get/diva2:1439878/FULLTEXT01.pdf.

[2] Indraneel Dutta Baruah. K-Means, DBSCAN, GMM, agglomerative clustering — mastering the popular models in a segmentation problem, 2020. URL https://towardsdatascience.com/k-means-dbscan-gmm-agglomerative-clustering-mastering-the-popular-models-in-a-segmentation-c891a3818e29.

[3] Jason Brownlee. Ordinal and one-hot encodings for categorical data, 2020. URL https://machinelearningmastery.com/one-hot-encoding-for-categorical-data/.

[4] Nagesh Singh Chauhan. DBSCAN clustering algorithm in machine learning, 2022. URL https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html.

[5] Ira Cohen. Outliers explained: a quick guide to the different types of outliers, 2019. URL https://towardsdatascience.com/outliers-analysis-a-quick-guide-to-the-different-types-of-outliers-e41de37e6bf6.

[6] Oscar Contreras Carrasco. Gaussian Mixture Models explained, 2019. URL https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95.

[7] Mohit Deshpande. Clustering with Gaussian Mixture Models, 2020. URL https://pythonmachinelearning.pro/clustering-with-gaussian-mixture-models/.

[8] Debomit Dey. ML | DBSCAN reachability and connectivity, 2019. URL https://www.geeksforgeeks.org/ml-dbscan-reachability-and-connectivity/.

[9] Debomit Dey. Dunn index and DB index – cluster validity indices | set 1, 2022. URL https://www.geeksforgeeks.org/dunn-index-and-db-index-cluster-validity-indices-set-1/?ref=lbp.

[10] Dheeru Dua i Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

[11] Agasti Kishor Dukare. Anomaly detection in Python with Gaussian Mixture Models, 2020. URL https://towardsdatascience.com/understanding-anomaly-detection-in-python-using-gaussian-mixture-model-e26e5d06094b.

[12] Ernst. Outlier detection: data preparation, 2019. URL https://donernesto.github.io/blog/outlier-detection-data-preparation/.

[13] Ernst. Outlier detection: DBSCAN, 2019. URL https://donernesto.github.io/blog/outlier-detection-with-dbscan/.

[14] Luigi Fiori. K-Means clustering using Python, 2020. URL https://medium.com/@luigi.fiori.lf0303/k-means-clustering-using-python-db57415d26e6.

[15] Ginni. What are the causes of anomalies?, 2022. URL https://www.tutorialspoint.com/what-are-the-causes-of-anomalies.

[16] Ginni. What are the challenges of outlier detection?, 2022. URL https://www.tutorialspoint.com/what-are-the-challenges-of-outlier-detection.

[17] María García Gumbao. Best clustering algorithms for anomaly detection, 2019. URL https://towardsdatascience.com/best-clustering-algorithms-for-anomaly-detection-d5b7412537c8.

[18] Tufan Gupta. Gaussian Mixture Model, 2021. URL https://www.geeksforgeeks.org/gaussian-mixture-model/.

[19] Sauravkaushiki Kaushik. An introduction to clustering and different methods of clustering, 2016. URL https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/.

[20] Bora Kizil. Introduction to anomaly detection in time-series data and K-Means clustering, 2020. URL https://medium.com/swlh/introduction-to-anomaly-detection-in-time-series-data-and-k-means-clustering-5832fb33d8cb.

[21] Vincenzo Lavorini. Gaussian Mixture Model clustering: how to select the number of components (clusters), 2018. URL https://towardsdatascience.com/gaussian-mixture-model-clusterization-how-to-select-the-number-of-components-clusters-553bef45f6e4.

[22] G Sandhya Madhuri i Dr. M. Usha Rani. Anomaly detection techniques causes and issues. *International Journal of Engineering & Technology*, 2018. URL https://www.sciencepubco.com/index.php/ijet/article/view/22791.

[23] Cory Maklin. Gaussian Mixture Models clustering algorithm explained, 2019. URL https://towardsdatascience.com/gaussian-mixture-models-d13a5e915c8e.

[24] Seiichi Uchida Markus Goldstein. A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. 2016. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4836738/pdf/pone.0152173.pdf.

[25] Davoud Moulavi, Pablo A. Jaskowiak, Ricardo J. G. B. Campello, Arthur Zimek, i Jörg Sander. Density-based clustering validation. *Society for Industrial and Applied Mathematics*, 2014. URL https://epubs.siam.org/doi/pdf/10.1137/1.9781611973440.96.

[26] Tara Mullin. DBSCAN parameter estimation using Python, 2020.

URL https://medium.com/@tarammullin/dbscan-parameter-estimation-ff8330e3a3bd.

[27] Pallavi Pandey. DBSCAN clustering, 2020. URL https://machinelearninggeek.com/dbscan-clustering/.

[28] Guansong Pang, Chunhua Shen, Longbing Cao, i Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)*, 54(2):1–38, 2021.

[29] Animesh Patcha i Jung-Min Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *ScienceDirect*, 2007. URL https://www.sciencedirect.com/science/article/abs/pii/S138912860700062X.

[30] Manish Pathak. Quick guide to evaluation metrics for supervised and unsupervised machine learning, 2020. URL https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/.

[31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, i E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

[32] Ramiz Aliguliyev Rasim Alguliyev i Lyudmila Sukhostat. Anomaly detection in big data based on clustering. *Statistics Optimization & Information Computing*, 2017. URL https://www.researchgate.net/publication/321448608_Anomaly_Detection_in_Big_Data_based_on_Clustering.

[33] scikit-learn developers. Compare the effect of different scalers on data with outliers, 2022. URL https://scikit-learn.org/stable/auto_examples/preprocessing/plot_all_scaling.html.

[34] Ajay Sreenivasulu. Evaluation of cluster based anomaly detection. 2019. URL https://www.diva-portal.org/smash/get/diva2:1382324/FULLTEXT01.pdf.

[35] Jiong Jin Srikanth Thudumu, Philip Branch i Jugdutt (Jack) Singh. A comprehensive survey of anomaly detection techniques for high dimensional big data. *Journal of Big Data*, 2020. URL https://doi.org/10.1186/s40537-020-00320-x.

[36] Dilip Valeti. DBSCAN algorithm for fraud detection & outlier detection in a data set, 2021. URL https://medium.com/@dilip.voleti/dbscan-algorithm-for-fraud-detection-outlier-detection-in-a-data-set-60a10ad06ea8.

[37] Wikipedia contributors. DBSCAN — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/wiki/DBSCAN. [Online; accessed 05-June-2022].

[38] Wikipedia contributors. Anomaly detection — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/wiki/Anomaly_detection. [Online; accessed 05-June-2022].

[39] Wikipedia contributors. Cluster analysis — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/wiki/Cluster_analysis. [Online; accessed 05-June-2022].

[40] Wikipedia contributors. k-means clustering — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/wiki/K-means_clustering. [Online; accessed 05-June-2022].

[41] Wikipedia contributors. Normal distribution — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/wiki/Normal_distribution. [Online; accessed 05-June-2022].

[42] Jan Šnajder. Strojno učenje: 19. grupiranje. 2021. URL https://www.fer.unizg.hr/_download/repository/SU-2020-19-Grupiranje[1].pdf.

Usporedba algoritama grupiranja u postupcima otkrivanja anomalija

Sažetak

U ovom se radu uspoređuju algoritmi grupiranja na različitim problemima otkrivanja anomalija u skupu podataka. Opisana je pojava anomalija, njeni uzroci, klasifikacija i moguće metode detekcije. Detaljno su opisani algoritmi K-sredina, DBSCAN i Gaussove mješavine te postojeće metrike za vrednovanje uspješnosti algoritama grupiranja. Odabrana su i opisana četiri skupa podataka s odstupanjima nad kojima je provedena usporedba algoritama. Objašnjeno je programsko ostvarenje sustava i detalji vrednovanja algoritama. Dobiveni rezultati prikazani su grafički i u obliku tablice te analizirani.

Ključne riječi: algoritmi grupiranja, anomalije, usporedba, k-sredine, dbscan, gaussova mješavina

Comparison Of Clustering Algorithms in Anomaly Detection Procedures

Abstract

This paper provides a comparison of clustering algorithms in different anomaly detection procedures. The occurrence of anomalies, their causes, classification, and possible detection methods are described. A detailed description of algorithms K-Means, DBSCAN, and Gaussian mixture model is given, as well as of the existing metrics for evaluating the performance of clustering algorithms. Four data sets with outliers were selected for the algorithm comparison and described. The implementation of the system and the details of algorithm evaluation are explained. The obtained results are shown both graphically and in the form of a table and analyzed.

Keywords: clustering algorithms, anomalies, comparison, k-means, dbscan, gaussian mixture