UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Finančna matematika – 1. stopnja

Nastja Košir

# PRIPRAVA PODATKOV ZA DETEKCIJO GOLJUFIJ V TELEKOMUNIKACIJAH

Delo diplomskega seminarja

Mentorica: izred. prof. dr. Marjetka Knez

Ljubljana, 2019

## Zahvala

*Največja zahvala gre mentorici izred. prof. dr. Marjetki Knez za strokovno pomoč in uporabne napotke pri izdelavi diplomskega dela. Hvala tudi Fakulteti za matematiko in fiziko ter vsem njenim zaposlenim za vso pridobljeno znanje in dodatne razlage.*

*Iskrena hvala staršema za moralno in finančno podporo. Prav tako se zahvaljujem preostali družini, prijateljem in partnerju za vso spodbudo in ljubezen.*

*Hvala tudi vsem ostalim, ki ste mi vsa ta leta stali ob strani.*

*Znanje je zaklad, ki venomer spremlja svojega lastnika.*
*Kitajski pregovor*

# Kazalo

# PRIPRAVA PODATKOV ZA DETEKCIJO GOLJUFIJ V TELEKOMUNIKACIJAH


## Povzetek

Lahko bi rekli, da so telekomunikacijske goljufije kuga 21. stoletja: povzročajo ogromne izgube in se hitro širijo. Vsebinsko pomenijo krajo telekomunikacijske opreme ali storitev, prikrivanje obstoja, lokacije izvira ali destinacije uporabe storitve, razkrivanje informacij o računih strank in podobno, z namenom okoristiti se z oškodovanjem telekomunikacijskega operaterja, ponudnika storitve ali uporabnika telefonskega omrežja.

Že iz definicije telekomunikacijske goljufije lahko razberemo, da jih poznamo kar nekaj različnih vrst. Delimo jih po dveh kriterijih. Prvi je glede na način uporabe omrežja ali storitve z namenom zlorabe – tako ločimo več tipov goljufij. Alternativno prevare delimo po načinu dostopa do omrežja ali storitev z namenom prevare. Pravimo, da poznamo več metod goljufij.

V delu diplomskega seminarja goljufe obravnavamo kot redke primerke v množici vseh uporabnikov telefonskega omrežja. Ogromne CDR datoteke z metapodatki o vseh zabeleženih klicih preiskujemo s pomočjo podatkovnega rudarjenja. Pri eksperimentalnem delu se soočimo z mnogimi izzivi: nenavadnimi časovni formati, manjkajočimi vrednostmi, obsežnimi podatkovnimi bazami, redukcijo dimenzije, veliko časovno zahtevnostjo in težavnostjo ločevanja med sleparji in običajnimi uporabniki. Potencialne goljufe iščemo s pomočjo metode $k$-voditeljev in metode LOF.


## Data preparation for fraud detection in telecommunications


## Abstract

One could say that frauds in telecommunications are the plague of the 21st century, because of its huge losses and extremely quick spreading. They include the theft of goods or telecommunication services, hidding existence of using the service, location of origin or user's destination, disclosure of the information about clients' accounts and so on, with the purpose to benefit from the loss of telecommunication operator, provider of the service or user of the telephone network.

From the definition we can conclude that there are many different frauds. We can divide them by two criteria. The first one is by the type of the fraud or the way of using of the network or service with the purpose to misuse. Similarly different methods of fraud are known with respect to the way of access to the network or service with the purpose to misuse.

In the thesis the swindlers are seen as outliers in the set of all users of the telecommunications network. Huge CDR files, which include the metadata of calls, are analysed with data mining. While doing experimental work, we face many challenges, such as unusual time formats, missing values, huge data bases, reduction of the dimension, time complexity and difficulties with distinguishing the normal and fraudulent behaviour. We search for potential swindlers with $k$-means and LOF methods.



**Math. Subj. Class. (2010):** 68P01, 68W01, 62P30, 62H30, 15A18

**Ključne besede:** zaznavanje goljufij, odkrivanje osamelcev, veliko podatkovje, podatkovno rudarjenje, gručenje, modeliranje

**Keywords:** fraud detection, detection of outliers, large datasets, data mining, clustering, modeling

# Uvod

Telekomunikacijska industrija se je v zadnjem desetletju izjemno razširila zaradi razvoja cenovno dostopne mobilne tehnologije. Po podatkih Združenja globalnega sistema za mobilne telekomunikacije s kratico GSMA (ang. Global System for Mobile Communications Association) je oktobra 2018 mobilni telefon uporabljalo približno 5,1 milijarde ljudi (od tega 3,7 milijard na trgih v razvoju), medtem ko je svetovna populacija znašala okrog 7,7 milijarde. Z naraščajočim številom uporabnikov mobilnega telefona se povečuje tudi število goljufij v telekomunikacijah in z njimi povezanih izgub.

Finančne izgube v telefoniji, ki so posledica goljufij, predstavljajo od štiri do šest odstotkov dohodka telekomunikacijskih podjetij, pri novejših operaterjih z manj izkušnjami in sredstev pa tudi do 20 odstotkov ([23]). Lahko so povzročene direktno z goljufijami ali pa posredno s prebegom nezadovoljnih strank h konkurenci zaradi visokih računov ali nezaupanja v operaterja. Ker je lažje obdržati obstoječo kot pridobiti novo stranko, podjetja ogromno vlagajo v čim učinkovitejše odkrivanje goljufij. A svojega znanja žal ne želijo deliti z javnostjo, ker predstavlja njihovo konkurenčno prednost, goljufije pa še dlje ostanejo skrite in povzročajo nadaljnje izgube. Po drugi strani bi z objavo svojih modelov pomagala tudi goljufom, saj bi ugotovili, kakšne goljufije lahko odkrijejo.

Kaj je pravzaprav *telekomunikacijska goljufija*? Gre za krajo telekomunikacijske opreme ali storitev, prikrivanje obstoja, lokacije izvira ali destinacije uporabe storitve, razkrivanje informacij o računih strank in podobno, z namenom okoristiti se z oškodovanjem telekomunikacijskega operaterja, ponudnika storitve ali uporabnika telefonskega omrežja.

Omenjene goljufije lahko odkrivamo s *podatkovnim rudarjenjem* oziroma sistematičnim iskanjem informacij v veliki količini podatkov. Ta se v telekomunikacijski industriji sooča s številnimi izzivi ([22]) zaradi velikosti podatkovnih množic, zaporedne in časovne narave podatkov, zahteve uporabe v realnem času za mnogo aplikacij, težavnosti ločevanja normalnega in nezakonitega obnašanja in zakonov, ki omejujejo uporabo osebnih podatkov.

V nadaljevanju dela diplomskega seminarja bomo skušali odgovoriti na vprašanja kdo, zakaj in kako goljufa, predvsem pa kako odkriti sleparje. V začetnem delu obravnavamo goljufije v splošnem: v prvem poglavju predstavimo tipe goljufij, v drugem metode goljufij in v tretjem nekaj statističnih podatkov. V četrtem poglavju so na kratko opisane obsežne datoteke telekomunikacijskih podjetij, imenovane CDR datoteke. Peto poglavje razloži pojma podatkovno rudarjenje in strojno učenje ter uvede pripadajočo notacijo. V zadnjem, to je šestem poglavju, predstavimo še uporabo podatkovnega rudarjenja na realnih podatkih slovenskega telekomunikacijskega podjetija.

## 1. Tipi goljufij

*Tip goljufije* (ang. type of fraud) razvršča goljufije glede na način uporabe omrežja ali storitve z namenom zlorabe. V nadaljevanju poglavja so obravnavane goljufija s premijsko stopnjo, goljufija pri delitvi prihodkov v mednarodnem in domačem omrežju, goljufija v gostovanju, goljufija s prodajanjem klicev, kuponska goljufija ter kraja vsebine in opreme.

### 1.1. Goljufija s premijsko stopnjo.

*Telefonske številke s premijsko stopnjo* oziroma *premijske številke* (ang. premium rate phone number) se uporabljajo pri plačilu storitev, kot so tehnična podpora, TV glasovanje, vedeževanje, nagradne igre, vroče linije in podobno. V Sloveniji se ponavadi začnejo s številko 090. *Goljufija s premijsko stopnjo* ali *PRSF* (ang. premium rate service fraud) po [14] nastane, ko se uporabniki ne zavedajo dodatnih stroškov pri klicu na premijsko številko. Pogosto se pojavlja v kombinaciji z metodo goljufija naročnine (opisana v 2.1), kjer se iz lažnih računov generirajo klici na premijsko številko pri drugem operaterju. Tam goljufi pobirajo dobiček, računov pa ne plačujejo in jih zaprejo, ko operater odkrije prevaro.

### 1.2. Goljufija pri delitvi prihodkov v mednarodnem omrežju.

*Goljufija pri delitvi prihodkov v mednarodnem omrežju* ali *IRSF* (ang. international revenue share fraud) po [14] pomeni zlorabo pri sporazumih medsebojnega povezovanja med operaterji v mednarodnem okolju pri mednarodnih premijskih številkah. Izvaja se podobno kot goljufija s premijsko stopnjo, le da se pri tem generirajo klici v mednarodno okolje. Evidenca klicev v mednarodnem omrežju prispe z zakasnitvijo, kar goljufom izjemno olajša zaslužek. Najpogosteje je vključenih več držav, ki imajo visoke provizije pri mednarodnih klicih. Praviloma klici sploh ne dosežejo geografske lokacije, ampak so preusmerjeni do tretjega ponudnika, ki pobere dobiček. Ta goljufija se pogosto uporablja v kombinaciji z metodami goljufija naročnine (opisana v 2.1), vdiranje v omrežje (2.2.4), goljufija Wangiri (2.3) in kloniranje (2.2.2).

### 1.3. Goljufija pri delitvi prihodkov v domačem omrežju.

*Goljufija pri delitvi prihodkov v domačem omrežju* ali *DRSF* (ang. domestic revenue share fraud) po [14] deluje tako kot IRSF, le da se zlorabe zgodijo v domačem omrežju. Navezuje se torej na zlorabe pri sporazumih medsebojnega povezovanja med operaterji v domačem okolju pri domačih premijskih številkah.

### 1.4. Goljufija v gostovanju.

*Gostovanje* naročnikom *domačega javnega telefonskega omrežja* ali *HPMN* (ang. home public mobile network) omogoča uporabo *gostujočega javnega telefonskega omrežja* ali *VPMN* (ang. visited public mobile network), ko je domače omrežje nedosegljivo. Pred tem morata operaterja omenjenih omrežij skleniti pogodbo o gostovanju, s katero se HPMN obveže, da bo plačevalo storitve, ki jih bo zagotavljalo VPMN – ne glede na to ali je bilo gostovanje goljufivo ali ne. *Goljufija v gostovanju* (ang. roaming fraud) je podrobno opisana v [11]. Zgodi se, ko prevarant preko HPMN uporablja VPMN na način, da mu HPMN ne more zaračunati opravljenih storitev, a je obvezano poravnati račun pri VPMN. Tovrstne goljufije je težko pravočasno odkriti, saj podatki o gostujočih klicih prispejo z zakasnitvijo zaradi počasne izmenjave informacij med domačim in gostujočim omrežjem. Posledično so tudi finančne izgube veliko višje. Goljufijo v gostovanju omogoča prav počasna in nezadostna izmenjava informacij med HPMN in VPMN. Tako se lahko

plačilu izognejo uporabniki predplačniških paketov z ničelnim stanjem na računu ali naročniki, ki so dosegli limit.

Obseg te goljufije lahko zmanjšamo s postopno aktivacijo gostovanja, omejitvijo klicev na premijske številke v gostovanju, preprečevanjem preusmeritev mednarodnih klicev v gostovanju in omejitvijo največje dolžine klicev. Problem pa je, ker številne omejitve odbijajo (nove) stranke. Ključen je tudi hiter prenos podatkov o klicih.

### 1.5. **Drugi tipi goljufij.**

Naslednje goljufije so opisane v [9]. Pri *goljufiji s prodajanjem klicev* (ang. call selling) gre za prodajo visokotarifnih klicev, ponavadi mednarodnih, pod njihovo tržno ceno. Ponavadi kriminalci uporabijo ponarejene dokumente, da bi se izognili plačilu in identifikaciji. Gre torej za metodo goljufije naročnine, ki je opisana v 2.1. Za odkrivanje te vrste goljufije je potrebna analiza CDR datotek (glej poglavje 4) in strankinih podatkov ter seznam visoko tveganih destinacij.

Pri *kuponski goljufiji* (ang. voucher fraud) je slepar predplačniški uporabnik. Ti morajo, da napolnijo svoj račun, vpisati posebno kodo, ki jo dobijo z nakupom kupona za napolnitev računa. Goljufija se najpogosteje zgodi tako, da uporabnik skuša uganiti kodo z vpisovanjem naključnih številk, ali pa vpiše že uporabljeno kodo. Ta metoda običajno podjetjem ne povzroča velikih izgub. Nenavadno obnašanje lahko hitro opazimo tako, da analiziramo strankino zgodovino poskusov napolnitve računa.

Omenjena je še *kraja vsebine* (ang. content stealing), pri kateri prevaranti brezplačno pridobijo vsebino z visoko vrednostjo (na primer video posnetki in igre) s pomočjo izkoriščanja predplačniškega sistema, ki ne deluje v realnem času, ali z izogibanjem plačila izdanega računa. Sorodna goljufija je tudi *kraja opreme* (ang. stolen goods).

## 2. Metode goljufij

Če je prejšnje poglavje razvrščalo goljufije glede na način uporabe omrežja ali storitve z namenom zlorabe, jih bo to po načinu dostopa do omrežja ali storitev z namenom prevare. *Metode goljufij* (ang. method of fraud), ki so obraravnane, so goljufija naročnine, goljufija prekrivanja (zlasti goljufija kloniranja in vdiranje v zasebno telefonsko omrežje), goljufija Wangiri, kraja osebnih podatkov in notranja goljufija.

### 2.1. **Goljufija naročnine.**

*Goljufija naročnine* (ang. subscription fraud) nastane, ko stranka odpre račun brez namena plačila obveznosti. Po [11] jo povzroči prevarant, ki uporablja lažne račune ali kartice z nezadostnim stanjem na računu, da bi goljufivo dostopal do storitev. Pri goljufiji naročnine je nenavadno obnašanje prisotno tekom celotnega odprtja računa oziroma so vsi klici goljufivi. Lahko jo omejimo s preverjanjem informacij, ki jih posredujejo stranke, usposabljanjem zaposlenih za učinkovitejše spopadanje z goljufijami, pregledovanjem strankinih kreditov in s shemo za kaznovanje goljufij.

### 2.2. **Goljufija prekrivanja.**

*Goljufija prekrivanja* (ang. superimposition fraud) je natančno opisana v [22]. Slepar pridobi nezakonit dostop do računa negoljufive stranke. Tovrstno goljufivo obnašanje se pogosto zgodi na ravni posameznega klica in poteka vzporedno z normalnim obnašanjem oziroma je z njim prekrito. Najpogostejša tehnika za prepoznavanje te goljufije je primerjava strankinega trenutnega klicnega obnašanja s profilom

njene pretekle uporabe, pri čemer si lahko pomagamo s tehnikami za detekcijo odstopanj in anomalij. Ker novo obnašanje še ne pomeni nujno goljufije, je bolj smiselna primerjava novega obnašanja s profilom goljufivega. Tretji način odkrivanja goljufije prekrivanja izkorišča dejstvo, da kriminalci redko delajo sami. Pogosto se obnašajo kot borzni posredniki in prodajajo nezakonite storitve drugim, ali pa iz več računov kličejo eno telefonsko številko.

2.2.1. *Deskanje.*

Po [9] goljufija *deskanja* (ang. surfing) pomeni uporabo storitev druge osebe brez njenega soglasja, na primer preko podvojitve SIM kartice (kloniranje), nelegalne pridobitve podatkov klicne kartice ali vdiranja v zasebno telefonsko omrežje.

2.2.2. *Goljufija kloniranja.*

Vključen mobilni telefon periodično oddaja dve unikatni identifikacijski številki, ki označujeta strankin račun: *mobilno identifikacijsko številko* ali *MIN* (ang. mobile identification number) in *elektronsko serijsko številko* ali *ESN* (ang. electronic serial number). *Goljufija kloniranja* (ang. cloning fraud) se po [10] zgodi, ko sta strankini številki MIN in ESN programirani v mobilni telefon, ki ni njen. Med uporabo tega drugega telefona omrežje zazna strankini identifikacijski številki in ji posledično zaračuna storitve. Tako lahko slepar komunicira z veliko nižjimi stroški – klici so običajno precej dražji od cene kloniranega telefona. Druga pridobitev je ključnega pomena za kriminalce: kloniranje omogoča nesledljivo komunikacijo, saj njihove identitete ni mogoče povezati s kloniranim računom.

Goljufija kloniranja je škodljiva na veliko načinov. Prvič, goljufiva uporaba preobremeni omrežje in onemogoča uporabo normalnim strankam. Drugič, mobilni operaterji morajo plačati za gostujočo uporabo. Nenazadnje se stranke pogosto odločijo za drugega operaterja, če se ta smatra za manj dovzetnega za goljufije. Zato mobilni operaterji veliko vlagajo v odkrivanje te prevare.

Poznamo dva razreda metod za odkrivanje goljufije kloniranja. *Metode pred klicem* (ang. pre-call methods) skušajo identificirati in blokirati goljufive klice ob nastanku. *Metode po klicu* (ang. post-call methods) se osredotočajo na identifikacijo goljufij, ki so se že zgodile, da bi lahko preprečili nadaljnje goljufivo obnašanje.

Metode pred klicem vključujejo potrjevanje telefona ali njegovega uporabnika pred klicem. Pogosto je potrebno pred vsakim klicem vpisati *PIN* (ang. personal identification number), kar otežuje prevare, a jih žal ne prepreči. Druga metoda se imenuje *radio frekvenčni prstni odtis* (ang. radio frequency fingerprinting) in mobilne telefone identificira preko prenosa njihovih karakteristik. *Overitev* ali *potrditev verodostojnosti* (ang. authentication) je zanesljiva in varna metoda, ki uporablja šifriranje s privatnim ključem. Ta metoda stranki ne povzroča nobenih nevšečnosti, vendar na žalost potrebuje posebno strojno opremo.

Metode po klicu periodično analizirajo podatke o klicih na vsakem računu, da bi ugotovile, ali se je zgodilo kloniranje ali ne. Ena izmed takih metod se imenuje *detekcija konfliktov* (ang. collision detection) in temelji na tem, da so klici enega računa, ki se časovno prekrivajo, najbrž goljufivi. Podobno idejo uporabi tudi metoda *preverjanje hitrosti* (ang. velocity checking), ki analizira lokacije in čase zaporednih klicev, da bi ugotovila ali bi lahko en uporabnik prepotoval vmesno razdaljo z razumno hitrostjo. Če na primer klicu iz Kopra po 10 minutah sledi klic iz Maribora na istem računu, najverjetneje ta račun uporabljata dve različni osebi. Omenjeni metodi zanesljivo delujeta za uporabnike, ki pogosto kličejo, ne pa tudi za tiste, ki svoje telefone redko uporabljajo. Metoda z imenom *analiza klicanih številk*

(ang. dialed digit analysis) s podatkovnim rudarjenjem na podatkih o klicih zbere telefonske številke, ki so jih klicali sleparji med obdobji goljufivih aktivnosti. Metoda nato preveri, koliko klicev posamezne stranke je bilo naslovljenih na telefonske številke iz seznama, in za prevarante razglasi tiste, ki so nad določeno mejo.

Tudi *profiliranje uporabnikov* (ang. user profiling) spada med metode po klicu. Analizira klicno obnašanje, da bi odkrilo anomalije, ki pogosto kažejo na goljufijo. Običajno dobro deluje tudi na naročnikih z nizko uporabo, ker je nenavadno obnašanje dobro razpoznavno, zato ustrezno dopolnjuje metodi detekcije konfliktov in preverjanja hitrosti. Poleg tega ima prednost pred metodo preverjanja verodostojnosti, saj ne potrebuje posebne strojne opreme. Žal tudi ta metoda ni brez napak. Težavno je namreč to, da je neobičajen klic ene stranke lahko tipičen za neko drugo stranko (na primer glede na kraj bivanja). Odkriti moramo torej indikatorje sprememb obnašanja, ki označujejo goljufijo, oziroma poiskati profil običajnega obnašanja posamezne stranke. Vsaka sprememba obnašanja seveda ne pomeni nujno goljufije, zato se je potrebno odločiti, kdaj sprožimo alarm.

2.2.3. *Goljufija duhov.*

Vir [9] *goljufijo duhov* (ang. ghosting) opredeli kot pridobitev brezplačne ali poceni klicne tarife tako, da prevaranti v podatkovnih bazah spremenijo podatke o njihovih klicih in preslepijo omrežje, ki misli, da gre za brezplačen klic ali javni telefon. Drugi način je konfiguracija stikal tako, da se neka telefonska številka ne pojavlja v CDR datotekah (opisane v poglavju 4).

2.2.4. *Vdiranje v zasebno telefonsko omrežje.*

Tarče metode *vdiranja v zasebno telefonsko omrežje* (angl. private branch exchange hacking – PBX hacking) so po [17] predvsem podjetja s šibkimi uporabniškimi imeni in gesli, sploh tista, ki niso spremenila privzetih. Hekerji tako lahko opravljajo anonimne in brezplačne klice, tudi na premijske številke in v tujino. S poslušanjem telefonskih pogovorov se lahko dokopljejo do zaupnih informacij in ukradejo identiteto katerega od zaposlenih s pripadajočo PIN številko. Tako lahko dostopajo do spletnega direktorija podjetja in pridobijo PIN številke ostalih zaposlenih ter spreminjajo gesla. Glavni znak te goljufije je neobičajno obnašanje za panogo podjetja, kot na primer klici izven delovnega časa in ogromno število različnih klicanih destinacij. Sorodna metoda je *vdiranje v zasebno telefonsko omrežje IP* (angl. private branch exchange internet protocol hacking – PBX IP hacking), pri kateri napadalci vdirajo v zasebno telefonsko omrežje, ki temelji na internetnemu protokolu.

### 2.3. **Goljufija Wangiri.**

*Goljufijo Wangiri* ali *en klic in prekinitev* (ang. one ring and cut) obravnava [18]. Beseda Wangiri izhaja iz japonščine, in sicer *wan* pomeni *en*, *giri* pa *prekiniti*. Kriminalec običajno z računalnikom generira ogromno sočasnih klicev iz premijske na naključne telefonske številke in jih prekine po prvem zvonenju. Tako se na mobilnikih pojavijo neodgovorjeni klici, za katere prevarant upa, da jih bodo uporabniki vrnili nazaj.

### 2.4. **Kraja osebnih podatkov.**

V [6] se pojavi tudi *kraja osebnih podatkov* (ang. phishing), pri kateri se sleparji pretvarjajo, da so predstavniki nekega legitimnega podjetja in skušajo pridobiti uporabniška imena, gesla in druge občutljive informacije. Alternativno se lahko do podatkov dokopljejo preko pošiljanja elektronskih in SMS sporočil ali oglasov na internetu, ki preusmerjajo na lažno stran ter prosijo za vpis z uporabniškim imenom

in geslom. Včasih tako tudi okužimo računalnik ali telefon z virusi. Kadar napadalci iščejo podatke finančnih institucij, gre za goljufijo *vzemi račun* (ang. account takeover).

2.5. **Notranja goljufija.**

Po [11] *notranje goljufije* (ang. internal frauds) zagrešijo zaposleni v telekomunikacijskem podjetju, kar jim omogočajo pomanjkljivi obrambni notranji varnostni sistemi ali protokoli. Prvi način je preko kraje in aktivacije SIM kartice (ang. subscriber identity/identification module), drugi pa uporaba testne kartice za gostovanje. Včasih se zgodi tudi, da zunanjim osebam prodajajo kode za dostop. Goljufija je pogostejša v podjetjih, v katerih najemajo zunanje delavce.

## 3. Telekomunikacijske goljufije v številkah

V nadaljevanju si poglejmo še nekaj statističnih podatkov o goljufijah v telekomunikaciji. Tabela 1 prikazuje ocenjene globalne prihodke in izgube zaradi goljufij v telekomunikacijskih podjetjih po letih. Podatki so povzeti po virih [1], [3] in [26]. Opazen je trend nižanja deleža izgub v prihodkih kot posledica povečevanja prihodkov in zmanjševanja izgub zaradi goljufij. Razlog tiči v izboljšanem znanju in izkušnjah na področju odkrivanja goljufij, naprednejši tehnologiji in žal tudi temu, da nekatere goljufije morda še niso bile odkrite. Opozoriti je potrebno tudi na to, da so ti podatki nastali zgolj na podlagi poročil telekomunikacijskih podjetij, ki svoje izgube deloma želijo prikriti, saj škodujejo ugledu pri naročnikih in poslovnih partnerjih.

| | **2005** | **2008** | **2011** | **2013** | **2015** | **2017** |
|---|---|---|---|---|---|---|
| **Prihodki [bilijon $]** | 1,2 | 1,7 | 2,1 | 2,2 | 2,25 | 2,3 |
| **Izgube zaradi goljufij [milijarda $]** | 61,3 | 60,1 | 40,1 | 46,3 | 38,1 | 29,2 |
| **Delež izgube v prihodkih** [%] | 5,11 | 3,54 | 1,88 | 2,09 | 1,69 | 1,27 |

Tabela 1. Prikaz izgub zaradi goljufij v telekomunikacijah.

V [26] so predstavljene tudi izgube v letu 2015 po metodah in tipih goljufij, ki so bile navedene. Metode z največjimi izgubami so goljufija naročnine (8,04 milijard $), vdiranje v zasebno telefonsko omrežje (3,93 milijard $), vdiranje v zasebno telefonsko omrežje IP (3,53 milijard $) in goljufija vzemi račun (1,96 milijard $). Pri tipih goljufij vodi goljufija pri delitvi prihodkov v mednarodnem omrežju s kar 10,76 milijardami $ izgub, sledijo pa ji goljufija s premijsko stopnjo (3,77 milijard $), kraja opreme (2,84 milijard $), kraja vsebine (2,35 milijard $) in goljufija pri delitvi prihodkov v domačem omrežju (2,09 milijard $).

Raziskava Združenja za nadzor komunikacijskih goljufij s kratico CFCA (ang. Communications Fraud Control Association) iz leta 2017 ([15]) je pokazala, da so se v telekomunikacijah začeli pojavljati novi trendi. Ker se je cena mednarodnih klicev v zadnjem času drastično znižala, so se sleparji preusmerili v druge prevare. Izgube z IRSF, ki je bila še v letu 2015 vodilen tip goljufije, so se zmanjšale na 6 milijard $, nasprotno pa so se povečale izgube pri metodah goljufija naročnine in vzemi račun na kar 12 milijard $. Porast goljufij s slednjo metodo je posledica visokokonkurenčnih trgov, ki so prisiljeni omogočati svojim strankam enostaven in hiter dostop do njihovih računov preko aplikacij ali potrditvenih kod.

In kako so goljufije razporejene geografsko? Vir [26] navede 10 držav, ki so bile izvor največjega števila goljufivih klicev v letu 2015. Padajoče so urejene takole: Združene države Amerike, Pakistan, Španija, Kuba, Italija, Filipini, Somalija, Združeno Kraljestvo, Dominikanska republika in Egipt. Situacija se je od leta 2013 očitno spremenila, saj so takrat vodile ZDA, Indija in Združeno kraljestvo. Po drugi strani so na seznamu držav z največ destinacij goljufivih klicev večinoma afriške in evropske države. V samem vrhu sta Kuba in Somalija, sledijo pa ji Bosna in Hercegovina, Estonija, Latvija, Gvineja, Srbija, Sierra Leone, Združeno Kraljestvo in Litva.

## 4. CDR datoteke

Odkrivanje telekomunikacijskih goljufij s pomočjo podatkovnega rudarjenja ni mogoče brez velikih baz podatkov. Datoteke, ki jih vsebujejo, imenujemo *CDR datoteke* (ang. Call detail record). Opisane so v [5] in [24]. V njih so shranjeni *matapodatki* (ang. metadata) oziroma podatki o podatkih, natančneje, kako je nek uporabnik uporabljal storitve, ne pa tudi vsebina (na primer telefonskega pogovora ali SMS sporočila). Operaterji potrebujejo informacije o kraju, času in načinu klicev za obračunavanje svojih storitev, iskanje motenj v omrežju, morebitno obravnavo pritožb naročnikov, analizo lastnega poslovanja in upravljanje podjetja. Datoteke so uporabne tudi za kriminaliste, ki lahko z njihovo pomočjo odkrijejo osumljence na podlagi lokacij klicev in oseb, s katerimi komunicirajo. Metapodatki ponavadi vključujejo datum in čas nastanka klica ter njegovo trajanje, izvorno in ciljno telefonsko številko, vrsto (notranji ali mednarodni), ceno in status (ne)odgovorjenosti klica, oznako ali gre za premijsko številko in tako dalje.

## 5. Podatkovno rudarjenje in strojno učenje

V današnjem (poslovnem) svetu se vse pogosteje srečujemo z obsežnimi podatki. Podjetja lahko uporabijo podatkovno rudarjenje za izboljšanje prodaje: zbrani podatki o nakupih so lahko v pomoč pri odkrivanju trendov, preko katerih sestavijo ustrezno marketinško kampanjo, načrtujejo ustrezno proizvodnjo in morda najdejo novo tržno nišo. Strojno učenje je po drugi strani na primer tehnologija, ki se skriva za samovozečimi avtomobili, ki se lahko hitro prilagodi novim okoliščinam med vožnjo. Omogoča tudi takojšnja priporočila, ko nakupujemo na spletnih trgovinah.

In kaj pravzaprav sploh pomenita zgornji besedni zvezi? Kako sta povezani? *Podatkovno rudarjenje* (ang. data mining) je po [21] računalniško podprt proces zbiranja, čiščenja, obdelave, analize in pridobivanja koristnega znanja iz (običajno obsežnih) podatkov. *Strojno učenje* (ang. machine learning) je proučevanje algoritmov, ki svojo uspešnost izboljšujejo iz izkušenj. Predstavlja bistveni korak podatkovnega rudarjenja, saj algoritmi strojnega učenja omogočajo odkrivanje vzorcev in modelov iz podatkov. So pa tu ključni algoritmi za učenje in ne dejansko odkrivanje vzorcev in modelov.

Poznamo dve glavni vrsti strojnega učenja: nenadzorovano in nadzorovano. *Nenadzorovano* (ang. unsupervised) išče zanimive vzorce v podatkih. To so na primer skupine primerkov, ki so si med seboj najbolj podobni. Po drugi strani *nadzorovano strojno učenje* (ang. supervised machine learning) uporabljamo za gradnjo napovednih modelov.

Urejeno $p$-terico *napovednih* oziroma *vhodnih spremenljivk* (ang. input variables) $X_i$ z zalogami vrednosti $D_i, i = 1,...,p$, označimo z $X = (X_1, X_2,..., X_p)$. Spremenljivka $X_i$ je *numerična* oziroma *kvantitativna* (ang. numerical, quantitative), če je $D_i \subseteq \mathbb{R}$, in *diskretna* oziroma *kvalitativna* (ang. discrete, qualitative), če je $D_i$ diskretna končna množica. Posamezna vrstica v podatkih za nenadzorovano strojno učenje predstavlja *primerek* (ang. case) $e = x = (x_1, x_2,..., x_p)$, kjer je $x_i \in D_i$. Pri nadzorovanem strojnem učenju, kjer napovedujemo vrednosti *ciljne* oziroma *izhodne spremenljivke/atributa* (ang. target variable/feature) $Y$ z zalogo vrednosti $D_Y$, so ustrezni primerki oblike $e = (x, y) = (x_1, x_2,..., x_p, y)$, kjer $x_i \in D_i$ in $y \in D_Y$. *Podatkovna množica* (ang. data set) $S$ je množica vseh možnih primerkov $e$ (za vsako od omenjenih vrst strojnega učenja). Število vseh primerkov označimo z $n$.

V nadaljevanju ne bomo poskušali povečati prodaje ali narediti samovozečega avtomobila, ampak se bomo osredotočili na iskanje goljufov v telekomunikacijah.

## 6. Primer

Poglejmo si, kako se torej lotimo odkrivanja goljufij v telekomunikacijah na primeru realnih klicev, ki jih je zaznalo eno od vodilnih slovenskih telekomunikacijskih podjetij. CDR datoteke obsegajo klice med ponedeljkom 30. 4. 2018 (začetek prvega klica ob 7:48 po našem času) in 14. 5. 2018 (začetek zadnjega klica ob 10:44 po našem času). Za vsakega izmed 6804447 klicev je podanih 58 različnih vrednosti oz. spremenljivk. Podatki o telefonskih številkah kličočega in klicanega ter še nekaj drugih spremenljivk je zaradi varstva osebnih podatkov seveda zašifriranih. Na tem podatkovju bomo s pomočjo podatkovnega rudarjenja v programu *R* skušali najti goljufe. Ker so ti redki primerki v množici vseh uporabnikov mobilnega telefona, bomo iskali osamelce.

### 6.1. **Uvoz in transformacija podatkov.**

Podatki so shranjeni v SQL bazi, zato jih najprej uvozimo v program R. Pri tem si pomagamo z R-ovim paketom *odbc* in njegovima funkcijama *dbConnect* in *dbGetQuery*, s katerima vzpostavimo povezavo s strežnikom in napišemo ustrezno poizvedbo z SQL stavkom.

Uvoženo podatkovje je preobsežno za analizo (v razumnem času), poleg tega pa odvečni podatki škodujejo modelu, zato bomo uporabili ustrezno transformacijo. Hitro ugotovimo, da ima kar 19 spremenljivk same enake vrednosti, kar pomeni, da ne bodo izboljšale našega modela. Enako velja tudi za 2 spremenljivki, ki imata unikatne označbe. Opazimo, da imajo trije pari spremenljivk identične vrednosti pri vseh primerkih, zato vzamemo po eno iz vsakega para. Iz datoteke s pojasnili o podatkih razberemo, da lahko neko spremenljivko izrazimo z drugimi, zato je ne potrebujemo. Prav tako izpustimo dve neinterpretabilni kategorični spremenljivki, ki bi nam le otežili raziskavo. Seveda v nadaljevanju teh spremenljivk sploh ne uvozimo.

Naša prva uganka so časovni formati. Le kaj pomeni čas 15258013880000000000? Prvih 10 števk predstavlja *Unix timestamp*, ki se določa kot čas v sekundah od 1. 1. 1970 ob 0:00 (UTC). Iz zgornjega števila torej izluščimo število 1525801388, ki predstavlja čas 8. 5. 2018 ob 17:43 (UTC). Upoštevamo še naš časovni pas in prištejemo dve uri (eno zaradi premika ure na poletni čas) ter tako dobimo čas pri nas: 8. 5. 2018 ob 19:43. Ustrezno pretvorbo za nas naredi ukaz *from_unixtime* v programu *MySQL*.

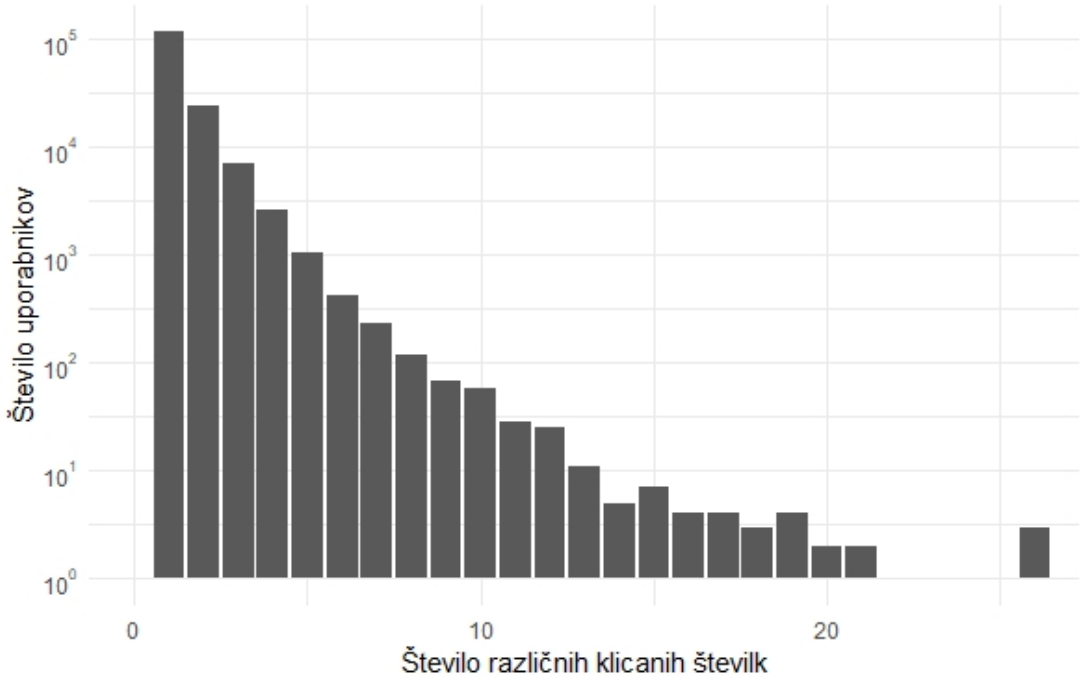
Iz časov začetka in konca klica izračunamo še njegovo trajanje in to vzamemo za novo spremenljivko, saj sta prejšnji dve preveč korelirani. Iz časa začetka klica izluščimo del dneva klica in oznako vikenda ali praznika, saj so takrat napadi pogostejši zaradi slabšega preiskovanja omrežja. Dan razdelimo na 5 delov, ki jih po vrsti označimo s števili od 1 do 5. Vrednost 1 označuje klice z začetkom med 5:00 in 7:59, 2 med 8:00 in 14:59, 3 med 15:00 in 18:59, 4 med 19:00 in 22:59 ter 5 med 23:00 in 4:59.

To seveda nista edini spremenljivki, ki ju skonstruiramo sami. V podatkih imamo podano telefonsko številko klicatelja in telefonsko številko, ki jo je sprejel klicani. Če se ti dve številki razlikujeta, je klic bolj verjetno goljufiv. Prevare se namreč pogosto zgodijo preko preusmeritev iz omrežja z višjo tarifo v omrežje z nižjo, klicanemu pa je vidno le slednje. Naša nova spremenljivka je torej indikator spremembe teh dveh številk.

V nadaljevanju nastavimo še pravilne razrede spremenljivk. Vse spremenljivke sicer predstavljajo faktorje, a kot taki označimo le telefonski številki kličočega in klicanega. Ostale s čim manjšo izgubo informacij pretvorimo v numerične spremenljivke, saj imamo zanje bistveno večji nabor algoritmov, poleg tega pa se ti tudi hitreje izvajajo.

6.1.1. *Agregacija po uporabnikih.*

Ker so goljufi običajno tisti, ki kličejo, podatke agregiramo po telefonski številki kličočega za neko izbrano časovno obdobje. Pri tem si pomagamo s funkcijo *ddply* iz R-ove knjižnice *plyr*. Za vsako tako telefonsko številko določimo število (različnih) klicanih telefonskih številk, število prejetih klicev in povprečno trajanje klicev, ostalim spremenljivkam pa v večini primerov poiščemo najpogostejšo vrednost. Lahko bi izračunali tudi povprečje, a je najpogostejša vrednost primernejša, saj v resnici gre za kategorične spremenljivke in želimo dobiti eno vrednost izmed že obstoječih.




Slika 1. Histogram števila različnih klicanih telefonskih številk za posameznega uporabnika na 1. 5. 2018.

Tako dobimo tabelo z 31 stolpci: *uporabnik*, *število različnih klicanih številk*, *število odhodnih klicev*, *število dohodnih klicev* (s strani uporabnikov obravnavanega telekomunikacijskega podjetja), *povprečno trajanje klica*, *nacionalna klicna koda kličočega* in *klicanega*, *časovni pas*, *verjetnost klica ob vikendu ali prazniku*, *del dneva*, *verjetnost preusmeritve*, *število preusmeritev*, *razlog preusmeritve*, *koda storitve*, *identifikacijska številka storitve*, *CDR kategorija*, *CDR status*, *tip poteka klica*, *identifikacijska številka scenarija*, *tip storitve*, *prejet MSC naslov*, *MSC nacionalna klicna koda*, *VLR številka*, *lokacijske informacije MCC*, *lokacijske informacije MNC*, *lokacijske informacije LAC*, *lokacijske informacije CI*, *zadnji BCSM dogodek*, *koda napake*, *transakcijska številka* in *kategorija kličoče skupine*.

Slika 1 prikazuje histogram števila različnih klicanih številk po uporabnikih na dan 1. 5. 2018. Razredi z manj kot dvema primerkoma niso prikazani. Obstajajo trije primerki, ki imajo po 130, 165 in 197 klicev (na prvi maj!). Glede na omenjen praznik, je obnašanje precej sumljivo, tako za posameznike, kot tudi za podjetja.

| | 30. 4. | 1. 5. | 2. 5. | 3. 5. | 4. 5. | 5. 5. | 6. 5. | 7. 5. |
|---|---|---|---|---|---|---|---|---|
| število klicev | 1271 | 326124 | 367839 | 593039 | 596841 | 432117 | 352311 | 659844 |
| število uporabnikov | 873 | 153709 | 176197 | 252469 | 253548 | 199509 | 169962 | 274542 |

| | 8. 5. | 9. 5. | 10. 5. | 11. 5. | 12. 5. | 13. 5. | 14. 5. | skupaj |
|---|---|---|---|---|---|---|---|---|
| število klicev | 623429 | 630600 | 639980 | 653492 | 463699 | 362251 | 101610 | 6804447 |
| število uporabnikov | 265362 | 268076 | 271580 | 275253 | 211814 | 174126 | 61381 | 926104 |

Tabela 2. Prikaz števila klicev in uporabnikov za posamezen dan.

Tabela 2 vsebuje število primerkov pred in po agregaciji podatkov oziroma število klicev in uporabnikov za določen dan. Na koncu je oboje prikazano še za vse dni skupaj. Opazimo, da smo na ta način količino podatkov v povprečju zmanjšali za kar 53 % na posamezen dan.

6.1.2. *Manjkajoče vrednosti.*

Težavo pri nadaljnjem modeliranju predstavljajo tudi manjkajoče vrednosti v kakšnem izmed stolpcev. Večina algoritmov za iskanje osamelcev namreč ne deluje (dobro) na podatkih, ki vsebujejo manjkajoče vrednosti, zato jih moramo nekako odstraniti. Različni pristopi so opisani v viru [20].

Najpreprostejša načina, ki ju lahko uporabimo, sta brisanje primerkov (*analiza popolnih primerkov*, ang. complete-case analysis) ali stolpcev (*analiza popolnih spremenljivk*, ang. available-case analysis) z manjkajočimi vrednostmi. Oba povzročita pristranskost v podatkih in modelih, posledica prvega pa je lahko tudi premajhna podatkovna množica za strojno učenje. Če brišemo spremenljivke, morda izbrišemo tudi pomembne.

Drugi pristop je *nadomeščanje manjkajočih vrednosti* (ang. imputation of missing values). Preprost način, ki žal povzroči zmanjšanje variance, je nadomeščanje s povprečjem pri numeričnih spremenljivkah in z najbolj zastopano vrednostjo pri diskretnih spremenljivkah. Variance ne bi zmanjšali, če bi nadomeščali z naključno vrednostjo iz domene, vendar to ne predstavlja dobro podatkov.

Alternativno začnemo s strojnim učenjem že pred izdelavo pravega modela. Prvi način je nadomeščanje z najbližjimi sosedi – manjkajočo vrednost nadomestimo s

povprečno vrednostjo $k$ najbližjih sosedov (pri čemer je lahko težavna nastavitev parametra $k$). Druga možnost je preko naslednjega iterativnega postopka:

(1) uporabimo enostavno metodo nadomeščanja,
(2) izberemo si spremenljivko $X_i$ (eno izmed tistih, ki so prej imele manjkajoče vrednosti) in se naučimo napovedni model za ciljno spremenljivko $X_i$, kjer so vse druge spremenljivke napovedne, ter z napovedmi tega modela nadomestimo neznane vrednosti. Ta korak ponavljamo dokler ne dosežemo maksimalnega števila iteracij ali pa se vrednosti spremenljivk ne spreminjajo več.

Na naših podatkih uporabimo funkcijo *missRanger* iz istoimenskega R-ovega paketa ([16]), ki hitro nadomešča vrednosti z veriženjem metode naključni gozd. *Naključni gozd* (ang. random forest) je algoritem nadzorovanega strojnega učenja, ki kombinira rešitve večih odločitvenih dreves, kar rezultira v bolj točno in stabilno napoved. Uporabimo ga lahko tako za regresijo kot tudi za klasifikacijo. Prej omenjena funkcija *missRanger* omogoča nastavitev parametra *pmm.k*. S tem pri drugem koraku iz zgornjega iteracijskega algoritma uporabimo *metodo PMM* (ang. predictive mean matching), ki deluje po algoritmu, opisanem v [2]. Za vsako od izbranih spremenljivk $X_i$ funkcija v drugem koraku zgornjega algoritma izvede naslednji postopek.

(1) Na primerkih brez manjkajočih vrednosti izvede linearno regresijo in tako dobi vektor koeficientov $b$.
(2) Naključno spremeni $b$ v $\hat{b}$ glede na neko porazdelitev (običajno multivariantno normalno s povprečjem $b$ in ocenjeno kovariančno matriko za $b$) in tako uvede dovoljšnjo variabilnost v nadomeščene vrednosti.
(3) S pomočjo vektorja $\hat{b}$ napove vrednosti $X_i$ za vse primerke (tudi tiste, ki nimajo manjkajočih vrednosti).
(4) Za vsak primerek $e_1$ z manjkajočo vrednostjo $X_i$ poišče množico s *pmm.k* primerki, pri katerih je napovedana vrednost najbližje napovedani vrednosti za $e_1$.
(5) Iz zgornje množice primerkov naključno izbere primerek $e_2$ in manjkajočo vrednost $e_1$ nadomesti s pravo vrednostjo primerka $e_2$.

Prednost uporabe metode PMM je, da so manjkajoče vrednosti nadomeščene z realnimi podatki oziroma z že obstoječimi vrednostmi (domena se ne poveča). Poleg tega se varianca dvigne na realno raven, kar omogoča večkratno uporabo metode naključni gozd.

Hitro ugotovimo, da imamo še vedno preobsežne podatke tako po številu dimenzij kot tudi po številu primerkov, zato se odločimo, da bomo modelirali na dnevni ravni, kasneje pa dnevne podatke agregirali na celotno obdobje. Izgradnja modela na dnevni ravni se zdi smiselna tudi iz vidika čim prejšnjega odkrivanja potencialnih prevarantov in zmanjševanja spremljajočih izgub. Tako postane stolpec z oznako vikenda ali praznika zaenkrat odveč, saj bo v enem dnevu vedno pokazal enako. V nadaljevanju bomo s pomočjo metode PCA zmanjšali še število spremenljivk.

6.1.3. *PCA.*

Veliko število napovednih spremenljivk lahko predstavlja velik problem. Večina algoritmov strojnega učenja namreč predpostavlja njihovo nekoreliranost, kar se v praksi zgodi zelo redko. Grajenje modelov na visokodimenzionalnem prostoru je

težavno tudi zaradi dolgotrajnega izvajanja metod. Želimo torej zmanjšati število dimenzij, a ohraniti čim več informacij iz naših podatkov.

*Redukcijo dimenzije* (ang. dimensionality reduction) lahko po viru [7] dosežemo z dvema pristopoma: z izbiro ali s konstrukcijo napovednih spremenljivk. *Izbira spremenljivk* (ang. feature elimination) ohrani samo nekaj najpomembnejših spremenljivk. Njena prednost je preprostost in ohranitev interpretabilnosti spremenljivk. Po drugi strani ne pridobimo nič informacij iz spremenljivk, ki smo jih izločili, čeprav bi te morda pripomogle k boljšemu modelu. Temu problemu se izognemo s *konstrukcijo spremenljivk* (ang. feature extraction). Iz $p$ napovednih spremenljivk konstruiramo novih $p$ **neodvisnih** spremenljivk, kjer je vsaka nova spremenljivka kombinacija vseh starih. Razporedimo jih glede na njihovo pomembnost oziroma glede na delež variance v podatkih, ki jo pojasnjujejo. Nato se odločimo, koliko spremenljivk želimo ohraniti, in vzamemo najboljše. Tako ohranimo najbolj koristne dele naših starih spremenljivk.

Ena izmed tehnik konstrukcije napovednih spremenljivk je *Analiza glavnih komponent* (ang. Principal component analysis – PCA), ki nove spremenljivke konstruira s pomočjo linearne transformacije. Deluje po naslednjem algoritmu.

(1) Vsakemu stolpcu matrike vhodnih podatkov $X \in \mathbb{R}^{n\times p}$ (pri nadzorovanem strojnem učenju odstranimo $Y \in \mathbb{R}^{n\times 1}$) odštejemo povprečje tega stolpca – tako imajo vsi stolpci povprečje 0.
(2) Če je pomembnost spremenljivk neodvisna od njihove variance, potem vsak stolpec delimo s standardno deviacijo tega stolpca. Tako skupaj s prvim korakom matriko $X$ transformiramo v matriko $Z \in \mathbb{R}^{n\times p}$, katere stolpci imajo povprečja enaka 0 in standardno deviacijo (in varianco) 1.
(3) Izračunamo matriko $Z^TZ \in \mathbb{R}^{p\times p}$, ki predstavlja kovariančno matriko matrike Z. Nadalje izračunamo še lastni razcep $Z^TZ = VDV^T$, kjer je $D \in \mathbb{R}^{p\times p}$ diagonalna matrika s padajoče urejenimi lastnimi vrednostmi po diagonali in $V \in \mathbb{R}^{p\times p}$ ortogonalna matrika s pripadajočimi lastnimi vektorji. Taka dekompozicija vedno obstaja, saj je $Z^TZ$ simetrična pozitivno semidefinitna matrika.
(4) Izračunamo matriko $W = ZV \in \mathbb{R}^{n\times p}$, ki je centrirana ali standardizirana (odvisno od odločitve v drugem koraku) verzija matrike $X$, dobljena z linearno kombinacijo prvotnih vhodnih spremenljivk. Ker so lastni vektorji matrike $V$ med seboj neodvisni, so taki tudi stolpci matrike $W$, ki jih imenujemo *glavne komponente* (ang. principal components).

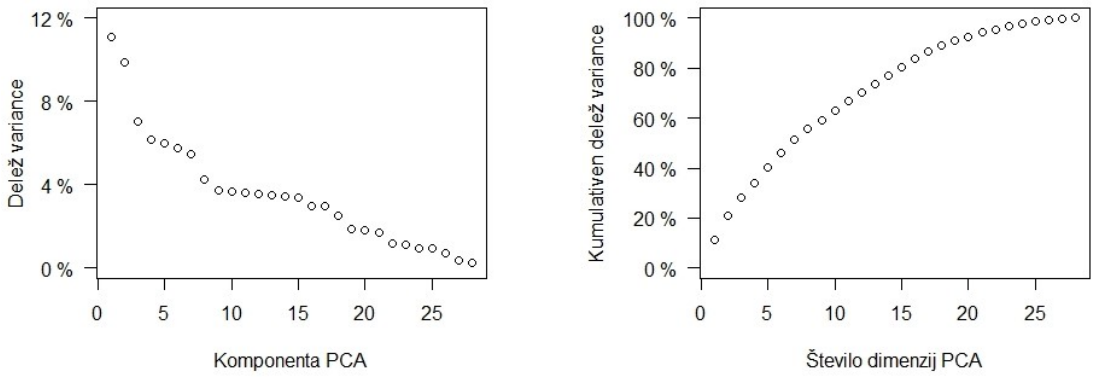
Ko izvedemo zgornji postopek, se moramo le še odločiti, koliko glavnih komponent bomo obdržali. Lahko vnaprej izberemo, koliko spremenljivk bomo uporabili – na primer, če želimo podatke grafično predstaviti, ohranimo dve ali tri dimenzije.

Drugi način je preko *deleža pojasnjene variance* (ang. proportion of variance explained). Vsaka lastna vrednost predstavlja pomembnost pripadajočega lastnega vektorja, zato je delež pojasnjene kumulativne variance enak vsoti lastnih vrednosti, ki jih obdržimo, deljeno z vsoto lastnih vrednosti vseh spremenljivk. Analogno izračunamo delež pojasnjene variance za posamezno spremenljivko tako, da delimo pripadajočo lastno vrednost z vsoto lastnih vrednosti vseh spremenljivk. Ena možnost uporabe deleža pojasnjene variance je, da obravnavamo kumulativen delež celotne variabilnosti vhodnih podatkov. Želimo na primer izbrati toliko spremenljivk, da skupaj pojasnijo vsaj 80 % variance. Alternativno si ogledamo delež pojasnjene variance za vsako od spremenljivk posebej. Tu si pomagamo z grafom

pojasnjenih varianc za vsako od glavnih komponent (urejenih padajoče glede na varianco) in poiščemo prvo "koleno" oziroma točko, po kateri ni več očitnega znižanja variance. Ta pristop je nekoliko subjektiven, saj ni natančno definiran.

Izberemo si torej prvih $q$ spremenljivk, kar pomeni, da ohranimo samo prvih $q$ ($q \leq p$) stolpcev matrike $V$, in dobljeno matriko označimo z $V_R \in \mathbb{R}^{p\times q}$. S transformacijo matrike $W$ v $q$-dimenzionalen podprostor dobimo novo podatkovno matriko $W_R = ZV_R \in \mathbb{R}^{n\times q}$, na kateri gradimo nadaljnje modele. Iz nje žal ne moremo več popolnoma rekonstruirati matrike $Z$, temveč dobimo zgolj njeno aproksimacijo $Z_R = W_R V_R^T \in \mathbb{R}^{n\times p}$.

Lastne vrednosti in lastni vektorji v sebi skrivajo globlji pomen. Slednji predstavljajo smeri, lastne vrednosti pa magnitudo oziroma pomembnost glavnih komponent. Večje lastne vrednosti so povezane s pomembnejšimi smermi. Predpostavili smo, da večja variabilnost v določeni smeri bolje pojasnjuje vhodni prostor $X$. Velika variabilnost namreč običajno predstavlja signal, majhna pa šum v podatkih. Zaznati hočemo le prvega, medtem ko želimo drugega čim bolj odpraviti. Pri razumevanju podatkov nam pomaga tudi kovariančna matrika $Z^TZ$, iz katere lahko razberemo, v kakšnem odnosu so naše prvotne vhodne spremenljivke.

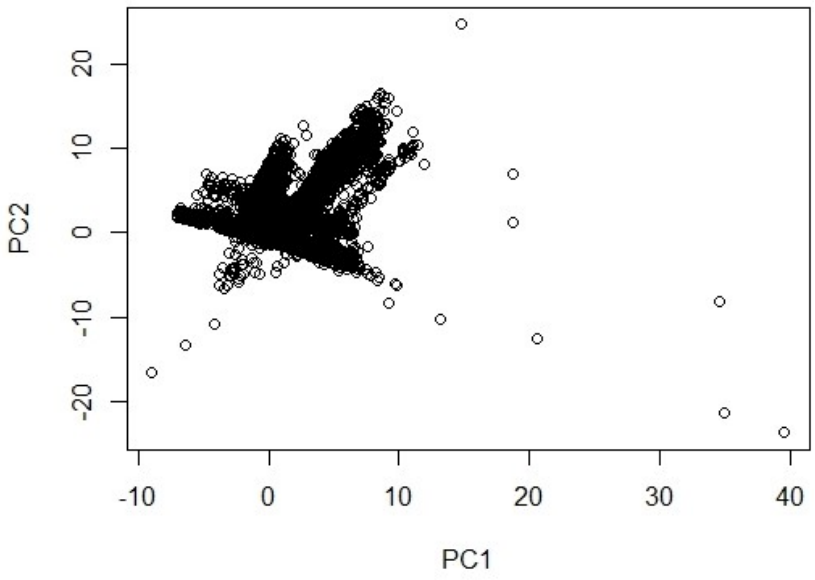



Slika 2. Delež pojasnjene variance in delež kumulativne variance na 1. 5. 2018.

V R-u za PCA na naših podatkih za dan 1. 5. 2018 uporabimo funkcijo *prcomp* iz paketa *stats*. Z uporabo ustreznih parametrov standardiziramo vrednosti spremenljivk. Na sliki 2 sta prikazana deleža pojasnjene variance in kumulativne variance. Če izbiramo število glavnih komponent glede na delež pojasnjene variance posamezne spremenljivke, poiščemo "koleno" in izberemo prvih sedem spremenljivk. Tako pojasnimo 51,4 % celotne variance. Alternativno se odločamo na podlagi pojasnjene kumulativne variance – želimo vsaj 80 % delež. Tako ohranimo najpomembnejših 15 glavnih komponent, s katerimi pojasnimo 80,6 % celotne variance vhodnih podatkov.

Pri določanju števila glavnih komponent se v našem primeru odločimo za kriterij vsaj 50 % pojasnjenega kumulativnega deleža variance in tako na posamezen dan dobimo od 5 do 8 glavnih komponent. S tem si zagotovimo primerljivo kakovost podatkov med različnimi dnevi in se izognemo subjektivnemu grafičnemu pristopu. Tako izračun tudi lažje implementiramo in avtomatiziramo.

Slika 3 prikazuje projekcijo naših podatkov na prvi dve glavni komponenti. Opazimo, da se nekaj primerkov precej razlikuje od ostalih, in prav te bomo v nadaljevanju skušali poiskati s pomočjo različnih metod.




Slika 3. Projekcija podatkov iz 1. 5. 2018 na prvi dve glavni komponenti.

6.2. **Iskanje osamelcev.**

Osamelce lahko iščemo na veliko različnih načinov, ki se med seboj ločijo po hitrosti in ustreznosti glede na naravo problema. V tem razdelku bomo preizkusili *metodo k-voditeljev* in *metodo LOF*.

6.2.1. *Metoda k-voditeljev.*

*Gručenje* (ang. clustering) je po [13] tehnika nenadzorovanega strojnega učenja, ki primerke razporeja v skupine tako, da so si znotraj skupine čim bolj podobni, med skupinami pa čim bolj različni. Gručenje lahko poteka na osnovi *centroidov* (ang. centroid), kjer je vsaka skupina predstavljena z osrednjim vektorjem oziroma centroidom, ki ni nujno primerek iz vhodne množice. Gre za iterativen algoritem gručenja, v katerem je podobnost določena z bližino primerka centroidu skupine.

Primer gručenja s centroidi je tudi *metoda k-voditeljev* (ang. $k$-means), ki dano podatkovno množico razdeli v $k$ skupin. Tekom algoritma minimizira celotno variacijo znotraj skupin, običajno kot vsoto kvadratov evklidskih razdalj med primerkom in pripadajočim centroidom:

$$W(C_\ell) = \sum_{e_i \in C_\ell} (e_i - \mu_\ell)^2,$$

kjer je $e_i$ primerek, ki pripada skupini $C_\ell$, in $\mu_\ell$ centroid skupine $C_\ell$. Celotno vsoto kvadratov znotraj skupin oziroma celotno variacijo znotraj skupin izračunamo z naslednjo formulo:

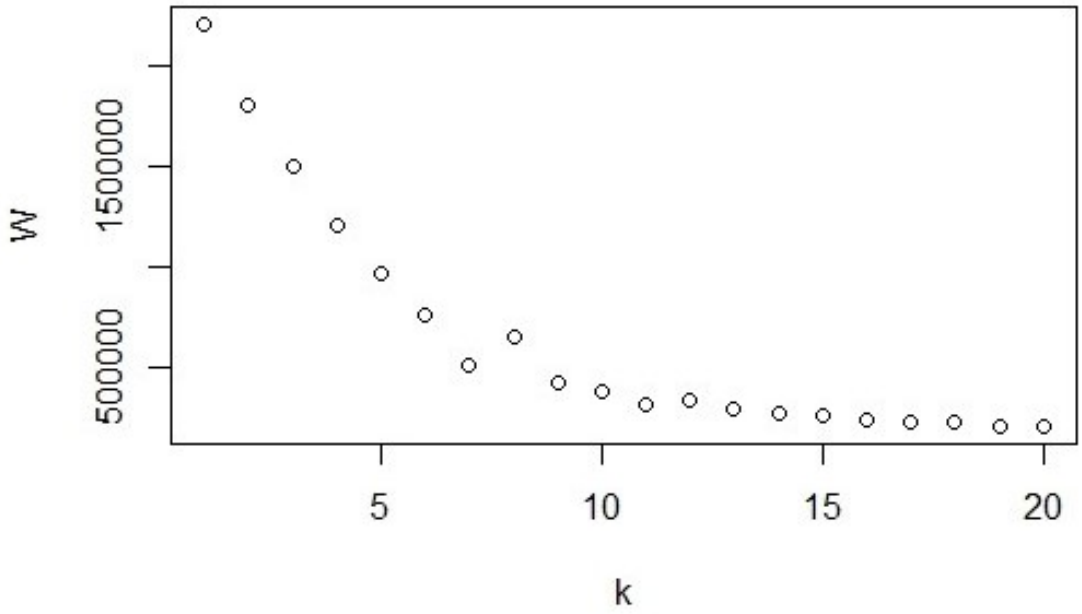
$$W = \sum_{\ell=1}^{k} W(C_\ell) = \sum_{\ell=1}^{k} \sum_{e_i \in C_\ell} (e_i - \mu_\ell)^2.$$

Poglejmo si sedaj še iteracijski postopek metode $k$-voditeljev.

(1) Naključno izberemo $k$ primerkov iz podatkovne množice in jih določimo za začetne centre skupin.
(2) Vsak primerek dodelimo najbližjemu centroidu glede na evklidsko razdaljo med njim in centroidom.

(3) Vsaki od $k$ skupin ponovno izračunamo centroid kot povprečje vseh primerkov v skupini.
(4) Koraka 2 in 3 ponavljamo, dokler se centroidi ne spreminjajo več ali je doseženo maksimalno število iteracij.

Največji problem metode $k$-voditeljev je, da moramo vnaprej izbrati, koliko skupin želimo. Če ne poznamo dobro podatkov, je to lahko zelo težko.




Slika 4. Celotna vsota kvadratov razdalj znotraj skupin na 1. 5. 2018.

Graf na sliki 4 prikazuje celotno vsoto kvadratov razdalj znotraj skupin na naših podatkih. Ta se zmanjšuje s številom skupin, kar je seveda pričakovano.

In kako si lahko pomagamo z metodo $k$-voditeljev pri iskanju osamelcev? Primerke razdelimo v skupine in za osamelce označimo tiste primerke, ki spadajo v skupine z malo primerkov (model $A$) ali pa osamelce določamo na podlagi njihove oddaljenosti od centra skupine, ki ji pripada (model $B$). Poleg tega bomo omenjena modela uporabili na dva načina. Pri prvem bomo model zgradili za vsak dan posebej, pri drugem pa na nekaj začetnih dneh in ga nato uporabili na novih podatkih. Tako dobimo 4 različne modele iskanja osamelcev s pomočjo metode k voditeljev: $1A$, $1B$, $2A$ in $2B$.

Poglejmo si model $1A$, pri katerem upoštevamo dejstvo, da goljufi redko delajo sami. Iščemo namreč skupine z malo primerki, ki odstopajo od ostalih. Za osamelce označimo tiste primerke, ki spadajo v skupine z manj kot $\frac{0.01 \times n}{k}$ primerkov. Prednost tega modela je, da ne vrne vedno osamelcev – v nekem časovnem oknu morda nihče ne goljufa. Rezultati omenjenega modela na naših podatkih so prikazani v tabeli 3. Podano je število osamelcev za vsako naravno število $k$ med 5 in 20 ter za vse datume, ki jih obravnavamo. Za $k \in \{1, 2, 3, 4\}$ je število osamelcev enako 0 pri vseh datumih, zato tega nismo posebej zapisovali v tabelo. V desnem delu tabele je razvidno še število različnih odkritih osamelcev, število vseh primerkov in delež različnih najdenih osamelcev glede na število vseh primerkov za vsak posamezen dan (izražen v procentih).

| k | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | os. | prim. | % |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30. 4. | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **1** | 873 | 0.11 |
| 1. 5. | 151 | 0 | 316 | 151 | 316 | 151 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 37 | 37 | 353 | 153709 | 0.23 |
| 2. 5. | 251 | 251 | 0 | 183 | 41 | 0 | 38 | 38 | 38 | 62 | 22 | 21 | 38 | 21 | 21 | 21 | 499 | 176197 | 0.28 |
| 3. 5. | 0 | 78 | 159 | 55 | 159 | 53 | 134 | 131 | 16 | 97 | 97 | 131 | 198 | 161 | 162 | 161 | **198** | 252469 | 0.08 |
| 4. 5. | 0 | 0 | 163 | 163 | 0 | 49 | 27 | 163 | 190 | 190 | 190 | 25 | 25 | 27 | 26 | 25 | 212 | 253548 | 0.08 |
| 5. 5. | 0 | 224 | 21 | 224 | 21 | 21 | 19 | 126 | 35 | 19 | 127 | 126 | 126 | 222 | 19 | 18 | 462 | 199509 | 0.23 |
| 6. 5. | 284 | 0 | 175 | 175 | 175 | 0 | 0 | 0 | 0 | 20 | 0 | 0 | 33 | 25 | 25 | 20 | 317 | 169962 | 0.19 |
| 7. 5. | 222 | 0 | 0 | 106 | 106 | 165 | 70 | 34 | 70 | 93 | 55 | 93 | 78 | 78 | 114 | 114 | 317 | 274542 | 0.12 |
| 8. 5. | 0 | 0 | 133 | 133 | 255 | 94 | 268 | 186 | 159 | 179 | 199 | 192 | 192 | 143 | 184 | 244 | **268** | 265362 | 0.10 |
| 9. 5. | 0 | 0 | 0 | 174 | 174 | 34 | 34 | 208 | 312 | 313 | 312 | 117 | 138 | 138 | 115 | 117 | **313** | 268076 | 0.12 |
| 10. 5. | 0 | 0 | 0 | 126 | 251 | 73 | 125 | 125 | 198 | 196 | 161 | 161 | 157 | 281 | 281 | 248 | **281** | 271580 | 0.10 |
| 11. 5. | 0 | 0 | 0 | 0 | 48 | 174 | 222 | 174 | 222 | 222 | 222 | 29 | 32 | 29 | 29 | 29 | 224 | 275253 | 0.08 |
| 12. 5. | 0 | 0 | 35 | 260 | 150 | 150 | 180 | 180 | 180 | 180 | 30 | 30 | 17 | 62 | 26 | 17 | 490 | 211814 | 0.23 |
| 13. 5. | 0 | 194 | 0 | 161 | 161 | 161 | 0 | 34 | 0 | 34 | 34 | 0 | 34 | 34 | 13 | 11 | 389 | 174126 | 0.22 |
| 14. 5. | 0 | 0 | 0 | 31 | 31 | 0 | 11 | 25 | 42 | 40 | 35 | 34 | 40 | 34 | 35 | 40 | **42** | 61381 | 0.07 |

TABELA 3. Prikaz števila najdenih osamelcev z uporabo modela $1A$.

Opazimo, da je pri večini datumov število osamelcev enako za več zaporednih $k$. Z nekaj manipulacije v programu R ugotovimo, da enako število osamelcev za različne $k$ na posamezen dan v našem primeru pomeni tudi enako skupino osamelcev. Pojav lahko interpretiramo kot močne skupine osamelcev oziroma primerkov, ki izrazito odstopajo od ostalih.

Zanimivo je tudi, da je pri šestih dnevih skupina z največjim številom osamelcev enaka številu različnih osamelcev. Slednje v tem primeru v tabeli označimo s krepko pisavo. Iz omenjenega pojava lahko sklepamo, da za različne $k$ algoritem najde iste osamelce.

| k | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | os. | prim. | % |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30. 4. | 2 | 2 | 2 | 2 | 1 | 3 | 3 | 3 | 3 | 7 | 3 | 4 | 2 | 2 | 1 | 2 | 2 | 4 | 11 | 2 | 16 | 873 | 1.83 |
| 1. 5. | 20 | 20 | 18 | 18 | 15 | 4 | 4 | 15 | 4 | 4 | 26 | 26 | 24 | 26 | 24 | 24 | 24 | 24 | 21 | 21 | 29 | 153709 | 0.02 |
| 2. 5. | 15 | 15 | 17 | 17 | 12 | 12 | 2 | 2 | 12 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | **17** | 176197 | 0.01 |
| 3. 5. | 11 | 10 | 10 | 11 | 11 | 10 | 24 | 10 | 14 | 9 | 12 | 12 | 9 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 24 | 252469 | 0.01 |
| 4. 5. | 30 | 29 | 28 | 29 | 17 | 17 | 28 | 61 | 18 | 25 | 32 | 18 | 33 | 21 | 33 | 33 | 31 | 32 | 31 | 31 | 75 | 253548 | 0.03 |
| 5. 5. | 2 | 2 | 2 | 2 | 2 | 2 | 10 | 2 | 10 | 10 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 12 | 2 | 2 | **12** | 199509 | 0.01 |
| 6. 5. | 20 | 17 | 17 | 18 | 17 | 17 | 8 | 11 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 20 | 20 | 20 | 9 | 20 | 26 | 169962 | 0.02 |
| 7. 5. | 15 | 14 | 14 | 15 | 17 | 16 | 12 | 11 | 11 | 14 | 8 | 14 | 8 | 15 | 9 | 15 | 21 | 21 | 14 | 14 | 25 | 274542 | 0.01 |
| 8. 5. | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 17 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | **17** | 265362 | 0.01 |
| 9. 5. | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | **2** | 268076 | 0.00 |
| 10. 5. | 22 | 19 | 19 | 19 | 15 | 14 | 15 | 12 | 17 | 13 | 17 | 17 | 17 | 18 | 24 | 24 | 23 | 23 | 23 | 23 | 28 | 271580 | 0.01 |
| 11. 5. | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 11 | 2 | 2 | 2 | **11** | 275253 | 0.00 |
| 12. 5. | 17 | 17 | 17 | 18 | 19 | 21 | 19 | 10 | 13 | 13 | 65 | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 15 | 68 | 211814 | 0.03 |
| 13. 5. | 11 | 11 | 12 | 12 | 14 | 12 | 14 | 8 | 8 | 8 | 10 | 8 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | **14** | 174126 | 0.01 |
| 14. 5. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | **1** | 61381 | 0.00 |

TABELA 4. Prikaz števila najdenih osamelcev z uporabo modela $1B$.

Poglejmo si sedaj model $1B$. Tokrat osamelce določamo na podlagi evklidske razdalje primerka do pripadajočega centroida. Po viru [4] določimo *mejno vrednost*

(ang. threshold value) po naslednji formuli:

$$mejna\ vrednost = \frac{maksimalna\ razdalja - minimalna\ razdalja}{2}.$$

Vrednost $k$ je tu implicitno upoštevana: za večji $k$ bo razlika razdalj manjša. Če je omenjena razdalja večja od mejne vrednosti, je primerek osamelec, sicer pa ne. Rezultate tega modela si lahko pogledamo v tabeli 4.

Hitro ugotovimo, da ta model najde manj osamelcev in da je to število bolj stabilno za posamezen datum. V večini primerov tudi tokrat enako število osamelcev na posamezen datum pomeni enako skupino osamelcev, a v nasprotju s prejšnjo metodo tu obstaja nekaj izjem. Ponovno s krepko pisavo označimo število osamelcev, ki tvorijo skupino, ki je enaka skupini z največjim številom osamelcev.

V nadaljevanju si natančneje oglejmo še modela $2A$ in $2B$. V obeh primerih model najprej zgradimo na dveh dneh, nato pa s pomočjo tega modela iščemo osamelce na preostalih dneh. V resnici vsak nov primerek primerjamo s tistimi, na podlagi katerih smo naučili naš model. S tem prihranimo na času na dva načina: algoritem ni le časovno učinkovitejši, temveč omogoča tudi iskanje osamelcev v realnem času, saj nam ni potrebno čakati na konec dneva za izgradnjo modela. Prednost tega pristopa je tudi uporaba informacij iz prejšnjih dni.

Ko se učimo na podatkih iz dveh dni, moramo pri transformaciji podatkov paziti, da ohranimo primerljivost s podatki na dnevni ravni. Pri agregaciji po uporabnikih namreč med drugim izračunamo tudi število različnih klicanih telefonskih številk ter število dohodnih in odhodnih klicev, ki bo za podatke iz dveh dni seveda v povprečju dvakrat večje kot na podatkih iz enega dneva. Problem enostavno rešimo tako, da vrednosti omenjenih spremenljivk delimo z 2. Spremenljivke, ki jih skonstruiramo s povprečji ali najpogostejšimi razredi očitno niso težavne, zato jih lahko ohranimo nespremenjene. Shranimo si tudi preslikavo iz PCA metode, ki preslika podatke iz dveh dni v nižjedimenzionalen podprostor, in jo kasneje uporabimo na novih podatkih.

| k | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | os. | prim. | % |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30. 4. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0** | 873 | 0.00 |
| **1. 5.** | 0 | 165 | 151 | 13 | 13 | 178 | 176 | 178 | 11 | 176 | 11 | 11 | 11 | 11 | 11 | 329 | 153709 | 0.21 |
| **2. 5.** | 0 | 183 | 249 | 23 | 23 | 206 | 205 | 206 | 22 | 205 | 22 | 22 | 21 | 21 | 21 | 455 | 176197 | 0.26 |
| 3. 5. | 80 | 80 | 59 | 80 | 80 | 59 | 243 | 59 | 140 | 324 | 322 | 325 | 324 | 216 | 320 | **325** | 252469 | 0.13 |
| 4. 5. | 72 | 72 | 49 | 72 | 72 | 49 | 234 | 49 | 212 | 397 | 393 | 397 | 397 | 300 | 390 | **397** | 253548 | 0.16 |
| **5. 5.** | 0 | 107 | 224 | 27 | 27 | 134 | 130 | 134 | 23 | 130 | 23 | 23 | 27 | 27 | 27 | 358 | 199509 | 0.18 |
| **6. 5.** | 0 | 109 | 175 | 8 | 8 | 117 | 117 | 117 | 8 | 117 | 8 | 8 | 8 | 8 | 8 | 292 | 169962 | 0.17 |
| 7. 5. | 87 | 87 | 53 | 86 | 86 | 52 | 258 | 52 | 111 | 317 | 315 | 318 | 317 | 219 | 315 | **318** | 274542 | 0.12 |
| 8. 5. | 89 | 89 | 60 | 89 | 89 | 60 | 267 | 60 | 154 | 361 | 359 | 361 | 361 | 259 | 358 | **361** | 265362 | 0.14 |
| 9. 5. | 80 | 80 | 53 | 80 | 80 | 53 | 215 | 53 | 225 | 387 | 386 | 387 | 387 | 299 | 385 | **387** | 268076 | 0.14 |
| 10. 5. | 77 | 77 | 51 | 77 | 77 | 51 | 222 | 51 | 176 | 347 | 345 | 348 | 347 | 264 | 344 | **348** | 271580 | 0.13 |
| 11. 5. | 66 | 66 | 45 | 66 | 65 | 45 | 223 | 45 | 219 | 397 | 396 | 397 | 397 | 297 | 392 | **397** | 275253 | 0.14 |
| **12. 5.** | 0 | 150 | 260 | 18 | 18 | 168 | 167 | 168 | 17 | 167 | 17 | 17 | 18 | 18 | 18 | 428 | 211814 | 0.20 |
| **13. 5.** | 0 | 194 | 161 | 8 | 8 | 202 | 200 | 202 | 6 | 200 | 6 | 6 | 8 | 8 | 8 | 363 | 174126 | 0.21 |
| 14. 5. | 3 | 3 | 1 | 3 | 3 | 1 | 9 | 1 | 27 | 35 | 35 | 35 | 35 | 33 | 35 | **35** | 61381 | 0.06 |

TABELA 5. Prikaz števila najdenih osamelcev z uporabo modela $2A$.

Upoštevamo tudi, da imamo ljudje drugačne navade ob vikendih in praznikih kot na običajne delovne dni, zato ločimo dva modela. Model za iskanje osamelcev ob delovnih dneh se naučimo na podatkih iz prvih dveh dni ob delavniku, ki obsegata celoten dan, torej 3. in 4. 5. 2018. Izbira se zdi smiselna, saj enkrat zgradimo model, nato pa v naslednjih dneh prihranimo čas in druge vire z uporabo istega modela, saj se v kratkem času podatki ne spremenijo veliko. Pri modelu za vikende in praznike vzamemo v ozir tudi to, da običajen vikend vseeno ni povsem enak prazniku, zato obravnavamo 1. in 5. 5. 2018, to je en praznični dan in običajno soboto.

Sedaj pa je že čas, da si pogledamo delovanje modelov $2A$ in $2B$. Pri obeh najprej s pomočjo modela, ki smo se ga naučili na "starih" podatkih, poiščemo skupino, v katero spada novi primerek. Pri modelu $2A$ v nadaljevanju preverimo, ali skupina, v kateri se nahaja novi primerek, morda ustreza kriteriju iz modela $1A$ in glede na to določimo, ali je osamelec ali ne. Na tem mestu moramo seveda povečati velikost omenjene skupine in število vseh primerkov za 1. Tudi model $2B$ je analogen modelu $1B$: podobno izračunamo evklidsko razdaljo primerka do pripadajočega centroida in po istem kriteriju kot prej označimo osamelce.

Rezultati modela $2A$ so prikazani v tabeli 5. Zaradi izgradnje dveh različnih modelov datume ob vikendih in praznikih označimo s krepko pisavo. Podatkov za $k \in \{1, 2, 3, 4, 5\}$ ne prikažemo v tabeli, saj za take $k$ ne najdemo osamelcev. Opazimo, da je kar pri devetih datumih vsota vseh različnih najdenih osamelcev za vse $k$ enaka največjemu številu osamelcev za posamezen $k$.

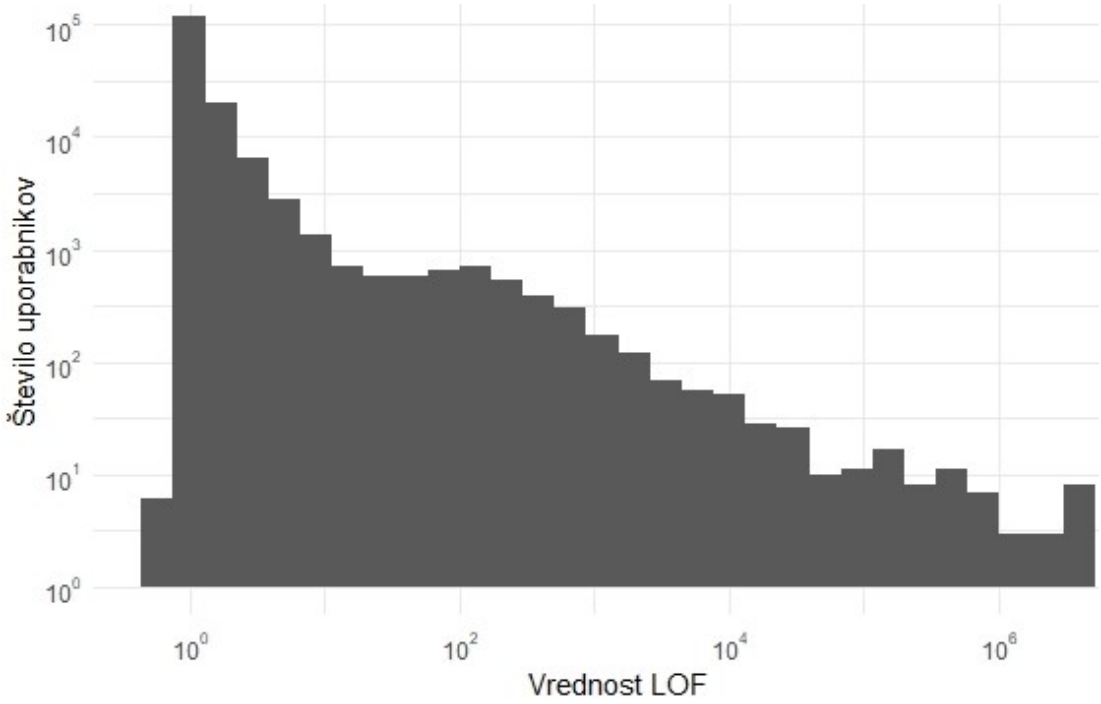
| k | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | os. | prim. | % |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30. 4. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | **1** | 873 | 0.11 |
| **1. 5.** | 21 | 19 | 19 | 18 | 18 | 15 | 23 | 15 | 12 | 12 | 23 | 23 | 23 | 23 | 30 | 19 | 23 | 23 | 21 | 23 | 36 | 153709 | 0.02 |
| **2. 5.** | 25 | 25 | 27 | 27 | 28 | 19 | 32 | 14 | 12 | 12 | 30 | 30 | 30 | 30 | 30 | 26 | 30 | 30 | 30 | 30 | 45 | 176197 | 0.03 |
| 3. 5. | 51 | 46 | 46 | 32 | 34 | 17 | 17 | 18 | 12 | 12 | 11 | 11 | 12 | 33 | 34 | 32 | 32 | 33 | 32 | 33 | 53 | 252469 | 0.02 |
| 4. 5. | 42 | 40 | 42 | 27 | 30 | 16 | 16 | 16 | 11 | 11 | 11 | 10 | 11 | 26 | 27 | 26 | 26 | 26 | 26 | 26 | **42** | 253548 | 0.02 |
| **5. 5.** | 21 | 20 | 21 | 21 | 21 | 16 | 13 | 16 | 9 | 9 | 11 | 11 | 11 | 11 | 11 | 11 | 11 | 22 | 22 | 23 | 27 | 199509 | 0.01 |
| **6. 5.** | 19 | 19 | 19 | 18 | 18 | 18 | 20 | 9 | 9 | 9 | 19 | 19 | 19 | 19 | 19 | 18 | 19 | 20 | 21 | 20 | 28 | 169962 | 0.02 |
| 7. 5. | 44 | 45 | 45 | 22 | 26 | 12 | 12 | 14 | 12 | 12 | 10 | 9 | 12 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | **45** | 274542 | 0.02 |
| 8. 5. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | **1** | 265362 | 0.00 |
| 9. 5. | 1 | 1 | 1 | 8 | 7 | 8 | 8 | 7 | 10 | 10 | 10 | 10 | 10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | **10** | 268076 | 0.00 |
| 10. 5. | 38 | 38 | 39 | 26 | 29 | 16 | 16 | 17 | 13 | 13 | 12 | 12 | 13 | 28 | 30 | 28 | 28 | 28 | 28 | 28 | **39** | 271580 | 0.01 |
| 11. 5. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | **1** | 275253 | 0.00 |
| **12. 5.** | 20 | 19 | 19 | 19 | 19 | 16 | 63 | 9 | 8 | 8 | 21 | 20 | 21 | 20 | 23 | 19 | 20 | 21 | 21 | 21 | 71 | 211814 | 0.03 |
| **13. 5.** | 11 | 10 | 9 | 9 | 9 | 12 | 15 | 6 | 6 | 6 | 12 | 12 | 12 | 12 | 13 | 10 | 12 | 12 | 11 | 12 | 21 | 174126 | 0.01 |
| 14. 5. | 10 | 10 | 10 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | **10** | 61381 | 0.02 |

TABELA 6. Prikaz števila najdenih osamelcev z uporabo modela $2B$.

V tabeli 6 si poglejmo še rezultate modela $2B$. Ponovno datume ob vikendih in praznikih označimo s krepko pisavo. Tokrat je vsota vseh različnih najdenih osamelcev za vse k enaka največjemu številu osamelcev za posamezen $k$ pri osmih datumih. V primerjavi z metodo $1B$ je najbolj očitna sprememba v deležu najdenih osamelcev na prvi dan, to je 30. 4. 2018.

6.2.2. *Metoda LOF.*

*Metoda LOF* (ang. local outlier factor) po [12] primerja lokalno gostoto točke z lokalnimi gostotami njenih $k$ najbližjih sosedov. Točke, ki imajo bistveno nižjo




Slika 5. Histogram vrednosti LOF na 1. 5. 2018 za $k = 4$.

gostoto kot točke v njeni soseščini, so prepoznane kot osamelci. Če ima neka točka vrednost LOF približno 1, pomeni, da je primerljiva s svojimi sosedi. Vrednosti, ki so bistveno večje od 1, kažejo na osamelce. Nasprotno točke, ki imajo vrednost LOF precej manjšo od 1 (a večjo od 0), predstavljajo primerke z večjo gostoto od njihovih sosedov.

Na naših podatkih uporabimo prej omenjeno metodo s pomočjo funkcije *lof* iz R-ovega paketa *dbscan*. Ta funkcija za vsak primerek izračuna vrednost LOF s pomočjo k-d drevesa, da pohitri iskanje najbližjih sosedov. Podroben opis k-d drevesa se nahaja v viru [25].

Glavna prednost metode LOF je, da ni občutljiva na porazdelitev podatkov: odkrije lokalne osamelce ob gostih gručah in primerkov v redkih skupinah ne prepozna kot osamelce. Poleg tega nam ne poda le indikatorja, ali je nek primerek osamelec ali ne, temveč tudi stopnjo "osamelosti".

Žal ima ta metoda tudi pomanjkljivost. Vrednost LOF namreč pokaže neskončno za točke, ki imajo vsaj $k$ ponovitev (glej vir [8]), čeprav vemo, da take točke zagotovo ne bi smele biti osamelci. S tem problemom se soočimo tudi mi, ko preslikamo naš vhodni prostor na nekaj glavnih komponent, zato pred uporabo metode odstranimo vse točke z več kot $k$ ponovitvami. Tem točkam kasneje pripišemo vrednost LOF enako 1.

Slika 5 prikazuje histogram vrednosti LOF na logaritmični skali za podatke iz 1. 5. 2018 in $k = 4$, iz česar želimo razbrati osamelce. Odločimo se, da je to 0,01 % primerkov z največjimi vrednostmi LOF in v tem primeru najdemo 16 osamelcev. Podatke za ostale datume in parametre $k \in \{1, 2, \ldots, 20\}$ razberemo iz tabele 7.

Glede na izbiro kriterija določanja potencialnih goljufov je očitno, da je število osamelcev za posamezen dan precej konsistentno za vse $k$. Posledično je skoraj konstanten tudi delež osamelcev, izstopa le na dan 30. 4. 2018, ko je 10-krat večji glede na preostale dni. V primerjavi z metodo $k$-voditeljev so sedaj skupine osamelcev

| k | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | os. | prim. | % |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30. 4. | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 6 | 873 | 0.69 |
| 1. 5. | 16 | 16 | 17 | 16 | 20 | 16 | 19 | 22 | 19 | 18 | 18 | 18 | 18 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 103 | 153709 | 0.07 |
| 2. 5. | 18 | 18 | 18 | 21 | 18 | 22 | 22 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 18 | 137 | 176197 | 0.08 |
| 3. 5. | 26 | 26 | 26 | 27 | 26 | 27 | 28 | 26 | 26 | 27 | 26 | 26 | 26 | 27 | 26 | 27 | 27 | 27 | 27 | 26 | 176 | 252469 | 0.07 |
| 4. 5. | 26 | 27 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 26 | 178 | 253548 | 0.07 |
| 5. 5. | 20 | 20 | 20 | 21 | 22 | 20 | 20 | 20 | 25 | 23 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 133 | 199509 | 0.07 |
| 6. 5. | 17 | 17 | 19 | 17 | 17 | 17 | 17 | 17 | 17 | 17 | 18 | 18 | 17 | 18 | 17 | 17 | 17 | 17 | 17 | 17 | 106 | 169962 | 0.06 |
| 7. 5. | 28 | 28 | 28 | 28 | 29 | 28 | 31 | 30 | 31 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 194 | 274542 | 0.07 |
| 8. 5. | 27 | 28 | 27 | 27 | 28 | 27 | 31 | 27 | 30 | 30 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 170 | 265362 | 0.06 |
| 9. 5. | 27 | 27 | 28 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 29 | 27 | 27 | 27 | 27 | 27 | 27 | 27 | 29 | 27 | 190 | 268076 | 0.07 |
| 10. 5. | 28 | 29 | 28 | 30 | 28 | 32 | 28 | 28 | 28 | 28 | 29 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 210 | 271580 | 0.08 |
| 11. 5. | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 28 | 191 | 275253 | 0.07 |
| 12. 5. | 22 | 22 | 23 | 22 | 22 | 23 | 23 | 23 | 22 | 22 | 22 | 22 | 23 | 23 | 23 | 22 | 22 | 22 | 22 | 22 | 141 | 211814 | 0.07 |
| 13. 5. | 18 | 18 | 18 | 18 | 20 | 18 | 19 | 18 | 18 | 18 | 18 | 21 | 19 | 25 | 25 | 23 | 22 | 18 | 18 | 18 | 120 | 174126 | 0.07 |
| 14. 5. | 7 | 7 | 7 | 8 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 35 | 61381 | 0.06 |

Tabela 7. Prikaz števila najdenih osamelcev z uporabo metode LOF.

bistveno bolj raznolike po sestavi primerkov, kar opazimo tudi iz bistveno večjega števila vseh najdenih osamelcev od števila osamelcev za posamezen $k$ na določen dan. Z nekaj manipulacije v programu R ugotovimo, da so najdeni osamelci za nekaj zaporednih $k$ pri posameznem datumu enaki. Uzremo tudi primerek, ki je bil razglašen za osamelca za vsak $k \in \{1, 2, \ldots, 20\}$.

### 6.3. **Analiza najdenih osamelcev.**

V tem razdelku se bomo posvetili vsem najdenim osamelcem v obdobju od 30. 4. 2018 do 14. 5. 2018. Zanimalo nas bo, koliko osamelcev je skupnih za več metod in po čem se razlikujejo od običajnih primerkov.

| | metoda $k$-voditeljev | | | | | LOF | skupaj (LOF in $k$-voditeljev) |
|---|---|---|---|---|---|---|---|
| | $1A$ | $1B$ | $2A$ | $2B$ | skupaj ($k$-voditeljev) | | |
| število osamelcev | 3547 | 342 | 3592 | 384 | 4029 | 2080 | 6101 |

Tabela 8. Prikaz števila različnih najdenih osamelcev v obdobju od 30. 4. 2018 do 14. 5. 2018 po metodah.

Iz tabel 8 in 9 hitro vidimo, da imata metodi $k$-voditeljev in LOF skupnih le 8 osamelcev od 6101. Opazimo, da so potencialni goljufi pri različnih modelih metode $k$-voditeljev dosti bolj enotni in imamo veliko več skupnih osamelcev.

Sedaj bi radi poiskali primerke, ki so prepoznani kot osamelci iz strani vseh petih metod – štirih modelov metode $k$-voditeljev in metode LOF. Razočarani ugotovimo, da takega primerka ni. Nadalje iščemo osamelce, ki jih najdejo vsaj štiri metode. Dobimo jih 178, a med njimi ni niti enega po metodi LOF. Gre torej za osamelce, ki so skupni vsem štirim modelom metode $k$-voditeljev.

Preverimo še, kako je za vsaj tri metode. Skupno odkrijemo 275 potencialnih prevarantov, od tega jih 8 pripada tudi metodi LOF. Tudi v primeru, ko nastavimo kriterij na vsaj dve metodi, še vedno najdemo le 8 osamelcev po metodi LOF. Skupnih osamelcev je tokrat 3391.

| | $1A$ | $1B$ | $2A$ | $2B$ | LOF |
|---|---|---|---|---|---|
| 1A | $*$ | 223 | 3293 | 276 | 8 |
| 1B | 223 | $*$ | 204 | 232 | 0 |
| 2A | 3293 | 204 | $*$ | 231 | 8 |
| 2B | 276 | 232 | 231 | $*$ | 0 |
| LOF | 8 | 0 | 8 | 0 | $*$ |

Tabela 9. Prikaz števila najdenih osamelcev v obdobju od 30. 4. 2018 do 14. 5. 2018, ki so skupni dvema metodama.

Natančneje si oglejmo število najdenih osamelcev, ki jih odkrijeta vsaj dve metodi. Iz tabel 8 in 9 sklepamo, da imata modela $1A$ in $2A$ kar 86 % skupnih osamelcev. Tudi modela $1B$ in $2B$ poiščeta veliko skupnih potencialnih goljufov, še več, pri kar 107 najdenih osamelcih označita nek primerek za osamelec za enako število različnih $k$.

V tabeli 10 si poglejmo še razlike med najdenimi osamelci po metodah LOF in $k$-voditeljev v primerjavi z vsemi primerki. Jasno je, da se ljudje obnašamo drugače med delavniki kot ob dela prostih dneh, zato statistike za vse primerke ločimo glede na ta kriterij. Ker metoda $k$-voditeljev najde zelo veliko osamelcev, ločeno zapišemo rezultate za vse osamelce skupaj in za tiste, ki se pojavijo pri vseh štirih modelih.

| | vsi primerki | | osamelci | | |
|---|---|---|---|---|---|
| | | | LOF | $k$-voditeljev | |
| | delavnik | vikend | | skupni | vsi |
| delež klicev ob vikendu ali prazniku | 36,00 % | | 34,32 % | 53,26 % | 46,23 % |
| delež preusmeritev | 43,15 % | 39,29 % | 70,58 % | 36,54 % | 39,80 % |
| klicna koda klicatelja = 386 | 98,16 % | 98,07 % | 96,12 % | 96,07 % | 96,10 % |
| povprečno število različnih klicanih številk | 1,24 | 1,14 | 1,15 | 6,99 | 3,93 |
| povprečno število odhodnih klicev | 1,51 | 1,23 | 1,49 | 18,10 | 6,92 |
| povprečno število dohodnih klicev | 0,93 | 0,72 | 0,47 | 3,33 | 3,35 |

Tabela 10. Primerjava nekaterih spremenljivk med osamelci in vsemi uporabniki. Povprečja so preračunana na dnevno raven.

Opazimo, da metoda LOF išče večinoma primerke, ki izvršijo preusmeritev. Po drugi strani je metoda $k$-voditeljev usmerjena v primerke, ki opravijo in prejmejo več klicev od običajnih uporabnikov, poleg tega pa kličejo tudi več različnih telefonskih številk. To je opazno zlasti na primerkih, ki so jih za osamelce označili vsi štirje modeli metode $k$-voditeljev. Zanimivo je tudi opažanje, da metoda $k$-voditeljev poišče velik delež potencialnih goljufov med vikendi in prazniki, čeprav v celotnih

podatkih klici med vikendi in prazniki obsegajo le 36 % vseh klicev. Pojav je skladen z našimi pričakovanji, saj vemo, da se večina prevar zgodi med vikendi in prazniki, ko je manj preiskav omrežja. Po drugi strani je nenavadno, da metoda LOF vrne manjši delež morebitnih goljufij v tem času. Pričakovano potencialni sleparji manj kličejo iz klicne kode za Slovenijo (386) kot ostala populacija.

| | | vsi primerki | | osamelci | | |
|---|---|---|---|---|---|---|
| | | | | LOF | $k$-voditeljev | |
| | | delavnik | vikend | | skupni | vsi |
| del dneva | 5:00 – 7:59 | 4,52 % | 3,22 % | 4,47 % | 1,69 % | 2,41 % |
| | 8:00 – 14:59 | 55,11 % | 52,26 % | 58,22 % | 66,29 % | 65,38 % |
| | 15:00 – 18:59 | 26,89 % | 25,14 % | 24,62 % | 16,85 % | 18,47 % |
| | 19:00 – 22:59 | 12,61 % | 15,90 % | 10,10 % | 12,36 % | 10,42 % |
| | 23:00 – 4:59 | 0,87 % | 3,48 % | 2,60 % | 2,81 % | 3,33 % |

Tabela 11. Delež klicev v določenem delu dneva za različne primerke.

Zanimivo si je ogledati tudi porazdelitev klicev preko dneva za različne skupine uporabnikov (tabela 11). Ob dela prostih dneh so časi klicev, prav nič presenetljivo, zamaknjeni na kasnejše ure. Nenavadna sprememba se zgodi pri metodi $k$-voditeljev: število klicev med 8. in 15. uro se poveča, med 15. in 19. uro pa zmanjša za približno 10 %. V primerjavi z delovnimi dnevi obe metodi najdeta primerke z večjim deležem klicev v nočnem času, kar je tudi posledica večjega deleža klicev med vikendi in prazniki.

## Zaključek

Telekomunikacijska industrija se v zadnjem času sooča z velikimi izzivi. Članek [19] iz leta 2018 napoveduje eksplozivno rast deleža goljufivih klicev: v letu 2017 je delež znašal 3,7 %, leta 2018 29 %, v letu 2019 pa naj bi dosegel celo 45 %. Pomen odkrivanja goljufij vsekakor ni zanemarljiv.

Ne glede na to, katero tehniko za detekcijo goljufij v telekomunikacijah uporabljamo, mora biti lahko prilagodljiva novim razmeram. Sleparji namreč neprestano iščejo nove načine za prevare, ki jih obstoječi modeli ne bi odkrili, zato se goljufivo obnašanje lahko občutno spremeni. Da preprečimo ogromne izgube, je potreben odziv v realnem času. Iz tega stališča sta izmed preizkušenih najprimernejša modela $2A$ in $2B$ metode $k$-voditeljev. Kar se tiče vsebinske ustreznosti smo v dvomih: metoda LOF je usmerjena v klice s preusmeritvami, metoda $k$-voditeljev pa v povečano število klicev. Za oboje vemo, da spodbudi sum na prevaro. Kaj torej izbrati?

Mislim, da ni enega samega odgovora – v nekaterih primerih je boljša ena metoda, v drugih pa druga. Verjetno je najboljša kombinacija obeh s primerjavo rezultatov in ugotovitvijo, v čem se goljufi razlikujejo od ostalih uporabnikov.

**account takeover** goljufija vzemi račun
**authentication** overitev/potrditev verodostojnosti
**available-case analysis** analiza popolnih spremenljivk
**call detail record** CDR datoteka
**call selling** goljufija s prodajanjem klicev
**case** primerek
**cloning fraud** goljufija kloniranja
**clustering** gručenje
**collision detection** detekcija konfliktov
**complete-case analysis** analiza popolnih primerkov
**content stealing** kraja vsebine
**data mining** podatkovno rudarjenje
**data set** podatkovna množica
**dialed digit analysis** analiza klicanih številk
**dimensionality reduction** redukcija dimenzije
**discrete/qualitative variable** diskretna/kvalitativna spremenljivka
**domestic revenue share fraud – DRSF** goljufija pri delitvi prihodkov v domačem omrežju
**electronic serial number – ESN** elektronska serijska številka
**feature elimination** izbira spremenljivk
**feature extraction** konstrukcija spremenljivk
**ghosting** goljufija duhov
**home public mobile network – HPMN** domače javno telefonsko omrežje
**imputation of missing values** nadomeščanje manjkajočih vrednosti
**input variable** napovedna/vhodna spremenljivka
**internal fraud** notranja goljufija
**international revenue share fraud – IRSF** goljufiija pri delitvi prihodkov v mednarodnem omrežju
**$k$-means** metoda $k$-voditeljev
**metadata** metapodatki
**method of fraud** metoda goljufije
**mobile identification number – MIN** mobilna identifikacijska številka
**numerical/quantitative variable** numerična/kvantitativna spremenljivka
**phishing** kraja osebnih podatkov
**post-call method** metoda po klicu
**pre-call method** metoda pred klicem
**premium rate phone number** telefonska številka s premijsko stopnjo ali premijska številka
**premium rate service fraud – PRSF** goljufija s premijsko stopnjo
**Principal component analysis – PCA** Analiza glavnih komponent
**private branch exchange hacking – PBX hacking** vdiranje v zasebno telefonsko omrežje
**proportion of variance explained** delež pojasnjene variance
**radio frequency fingerprinting** radio frekvenčni prstni odtis
**random forest** naključni gozd
**roaming fraud** goljufija v gostovanju
**stolen goods** kraja opreme

**subscription fraud** goljufija naročnine
**superimposition fraud** goljufija prekrivanja
**surfing** goljufija deskanja
**target variable/feature** ciljna/izhodna spremenljivka/atribut
**type of fraud** tip goljufije
**(un)supervised machine learning** (ne)nadzorovano strojno učenje
**user profiling** profiliranje uporabnikov
**velocity checking** preverjanje hitrosti
**visited public mobile network – VPMN** gostujoče javno telefonsko omrežje
**voucher fraud** kuponska goljufja
**Wangiri fraud – one ring and cut** goljufija Wangiri – en klic in prekinitev

## Literatura


[1] S. Alam, *2017 Global Fraud Loss Survey – CFCA*, 2018, [ogled 15. 11. 2018], dostopno na `https://www.scribd.com/document/368471387/2017-Global-Fraud-Loss-Survey-CFCA-pdf`.

[2] P. Allison, *Imputation by Predictive Mean Matching: Promise & Peril*, [ogled 6. 4. 2019], dostopno na `https://statisticalhorizons.com/predictive-mean-matching`

[3] R. Aronoff, CFCA, *2013 Global Fraud Loss Survey*, [ogled 15. 11. 2018], dostopno na `https://www.ntvoiceanddata.co.uk/Global_Fraud_Loss_Survey2013.pdf`

[4] A. Barai, L. Dey, *Outlier Detection and Removal Algorithm in K-Means and Hierarchical Clustering*, [ogled 16. 7. 2019], dostopno na `https://pdfs.semanticscholar.org/4c68/4a9ba057fb7e61733ff554fe2975a2c91096.pdf`

[5] K. Bartley, *What Are Call Detail Records (CDRs)?*, [ogled 17. 11. 2018], dostopno na `https://www.onsip.com/blog/what-are-call-detail-records-cdrs`

[6] N. Bhular, *Telecommunication Fraud and Detection Techniques: A Review*, International Journal on Recent and Innovation Trends in Computing and Communication, **4**(5), (2016), 493–495.

[7] M. Brems, *A One-Stop Shop for Principal Component Analysis*, [ogled 6. 4. 2019], dostopno na `https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c`

[8] M. M. Breunig, H. P. Kriegel, R. T. Ng, J. Sander, *LOF: Identifying density-based local outliers*, Proc. ACM SIGMOD 2000 Int. Conf. On Management of Data. ACM Press, Dalles, (2000), 93–104.

[9] L. Cortesão, F. Martins, A. Rosa, P. Carvalho, *Fraud Management Systems in Telecommunications: a practical approach*, 2015, [ogled 2. 11. 2018], dostopno na `https://www.researchgate.net/publication/228963798_Fraud_Management_Systems_in_Telecommunications_a_practical_approach`.

[10] T. Fawcett, F. Provost, *Adaptive Fraud Detection*, Data Mining and Knowledge Discovery Journal, Kluwer Academic Publishers, **1**(3), (1997), 291–316.

[11] G. M. Fernandez, P. G. Teodoro, J. D. Verdejo, *Fraud in Roaming Scenarios: An Overview*, IEEE Wireless Communications, **16**(6), (2010), 88–94.

[12] M. Hahsler, *lof, Local Outlier Factor Score*, [ogled 7. 4. 2019], dostopno na `https://www.rdocumentation.org/packages/dbscan/versions/1.1-3/topics/lof`

[13] S. Jaiswal, *K-Means Clustering in R Tutorial*, [ogled 7. 4. 2019], dostopno na `https://www.datacamp.com/community/tutorials/k-means-clustering-r`

[14] A. Kamnik, *Odkrivanje goljufij v telekomunikacijah z uporabo podatkovnega rudarjenja*, magistrsko delo, Fakulteta za matematiko in fiziko, Univerza v Ljubljani, 2018.

[15] J. Lane-Sellers, *The changing nature of fraud in telecommunications industry*, 2018, [ogled 17. 11. 2018], dostopno na `https://www.thepaypers.com/expert-opinion/the-changing-nature-of-fraud-in-telecommunications-industry/773807`

[16] M. Mayer, *Package 'missRanger'*, [ogled 6. 4. 2019], dostopno na `https://cran.r-project.org/web/packages/missRanger/missRanger.pdf`

[17] C. Pollard, *Telecom fraud: The cost of doing nothing just went up*, Computers & Security, **24**, (2005), 437–439.
[18] I. Rufferty, *The Greatest Telecommunication Frauds*, 2017, [ogled 2. 11. 2018], dostopno na `https://medium.com/bsg-sms/the-greatest-telecommunication-frauds-922d1413c760`.
[19] H. Shaban, *Nearly half of cellphone calls will be scams by 2019, report says*, [ogled 12. 8. 2019], dostopno na `https://www.washingtonpost.com/technology/2018/09/19/nearly-half-cellphone-calls-will-be-scams-by-report-says/`
[20] L. Todorovski, *Obravnava manjkajočih vrednosti in neenakomerne porazdelitve vrednosti ciljne spremenljivke*, verzija 7. 6. 2018, [ogled 5. 4. 2019], dostopno na `http://kt.ijs.si/~ljupco/lectures/itap-1718/itap-13-8pp.pdf`
[21] L. Todorovski, *Podatkovno rudarjenje in odkrivanje zakonitosti v podatkovnih bazah*, verzija 9. 5. 2018, [ogled 6. 4. 2019], dostopno na `http://kt.ijs.si/~ljupco/lectures/itap-1718/itap-10-8pp.pdf`
[22] G. Weiss, *Networking and Telecommunications: Concepts, Methodologies, Tools and Applications*, IGI Global, Hershey, 2010, 197.
[23] T. Žagar, *Doprinos vizualizacije pri boju z goljufijami v telefonskih sistemih*, magistrsko delo, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2008.
[24] *Call detail record*, [ogled 17. 11. 2018], dostopno na `https://en.wikipedia.org/wiki/Call_detail_record`
[25] *k-d tree*, [ogled 6. 8. 2019], dostopno na `https://en.wikipedia.org/wiki/K-d_tree`
[26] *2015 Global Fraud Loss Survey*, [ogled 15. 11. 2018], dostopno na `https://www.sysnettelematica.it/documenti-um-labs/antifrode/529-communication-fraud-control-association-survay-2015/file`