UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

RAZVRŠČANJE PORABNIKOV NA PODLAGI VZORCEV TIPKANJA

Ljubljana, februar 2024 ANDREJ MOHORIČ

IZJAVA O AVTORSTVU

Podpisani Andrej Mohorič, študent Ekonomske fakultete Univerze v Ljubljani, avtor predloženega dela z naslovom Razvrščanje porabnikov na podlagi vzorcev tipkanja, pripravljenega v sodelovanju s svetovalcem doc. dr. Denisom Marinškom in sosvetovalko red. prof. dr. Barbaro Čater

I Z J A V L J A M

1. da sem predloženo delo pripravil samostojno;

2. da je tiskana oblika predloženega dela istovetna njegovi elektronski obliki;

3. da je besedilo predloženega dela jezikovno korektno in tehnično pripravljeno v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani, kar pomeni, da sem poskrbel da so dela in mnenja drugih avtorjev oziroma avtoric, ki jih uporabljam oziroma navajam v besedilu, citirana oziroma povzeta v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani;

4. da se zavedam, da je plagiatorstvo – predstavljanje tujih del (v pisni ali grafični obliki) kot mojih lastnih – kaznivo po Kazenskem zakoniku Republike Slovenije;

5. da se zavedam posledic, ki bi jih na osnovi predloženega dela dokazano plagiatorstvo lahko predstavljalo za moj status na Ekonomski fakulteti Univerze v Ljubljani v skladu z relevantnim pravilnikom;

6. da sem pridobil vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v predloženem delu in jih v njem jasno označil;

7. da sem pri pripravi predloženega dela ravnal v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobil soglasje etične komisije;

8. da soglašam, da se elektronska oblika predloženega dela uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom članice;

9. da na Univerzo v Ljubljani neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve predloženega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja predloženega dela na voljo javnosti na svetovnem spletu preko Repozitorija Univerze v Ljubljani;

10. da hkrati z objavo predloženega dela dovoljujem objavo svojih osebnih podatkov, ki so navedeni v njem in v tej izjavi;

11. da sem preveril verodostojnost informacij, ki izhajajo iz zapisov na podlagi uporabe orodij umetne inteligence.

V Ljubljani, dne _____ Podpis študenta: _____

KAZALO

KAZALO SLIK

KAZALO PRILOG

SEZNAM KRATIC

angl. – angleško
CMU – (angl. Carnegie Mellon University); Univerza Carnegie Mellon
IDE – (angl. Integrated Development Environments); Integrirani razvoj okolja
IOT – (angl. Internet of Things); splet stvari
PPC – (angl. Pay-Per-Click Advertising); oglaševanje s plačilom na klik
SEM – (angl. Search Engine Marketing); trženje prek iskalnikov
SEO – (angl. Search Engine Optimization); optimizacija spletnih strani za iskalnike

1 UVOD

V zadnjih letih smo priča nenehnemu vzponu digitalne dobe, saj število uporabnikov spleta dramatično narašča. Na začetku 4. četrtletja leta 2023 je ta nepretrgan trend dosegel novo višino, saj je 5,30 milijarde ljudi po vsem svetu, kar znaša 65,7 % svetovnega prebivalstva, redno vključenih v svetovni splet. V preteklih 12 mesecih se je število spletnih uporabnikov povečalo za impresivnih 189 milijonov, kar jasno kaže, da se število ljudi, ki vstopajo v virtualni svet, še naprej vztrajno povečuje. Ta razmah spleta odraža globalno premikanje k digitalni povezanosti in poudarja vlogo spleta kot nepogrešljivega orodja v vsakdanjem življenju (DataReportal, brez datuma).

Za številna podjetja je ključnega pomena, da identificirajo svojo primarno skupino porabnikov, ki se pogosto imenuje ciljni trg, ter razumejo potrebe in želje ciljnega trga, ko razvijajo nove izdelke ali storitve ali ko vstopajo na nov trg. Razumevanje ciljnega trga je koristno ne le v procesu razvoja izdelka, temveč tudi pri izvajanju trženjskih načrtov in pri izbiri ustreznih prodajnih mest. Podrobna slika ciljnega trga pripomore pri izvajanju učinkovite in ciljno usmerjene promocije, skrb za najdragocenejše porabnike, pri oblikovanju novih izdelkov, ki najbolj ustrezajo potrebam porabnikov, pri izbiri ustreznih prodajnih mest za izdelke ter zagotavljanju storitev in podpori potrebam/zahtevam izbranega trga (Curtis in Allen, 2018).

Ciljni trg sestavljajo porabniki, ki želijo določen izdelek ali storitev in so zanj pripravljeni plačati donosno ceno. Vsak segment porabnikov ima edinstvene lastnosti, ki vplivajo na odločitve porabnikov. Podjetja lahko pri segmentaciji uporabijo različne osnove, ki jih delimo v štiri večje skupine: demografske, geografske, psihografske in vedenjske. Demografske oziroma sociodemografske značilnosti porabnika opisujejo njegovo starost, stopnjo izobrazbe, dohodek itd. Psihografske značilnosti vključujejo hobije, interese in cilje ciljnega trga.

Prav tako je pomembno poznati posebne potrebe ali želje ciljnega trga. Obstajajo številne metode, ki lahko pomagajo podjetjem pri opredelitvi njihovih ciljnih trgov. Primarne raziskovalne metode vključujejo ankete porabnikov, preizkušanja cen, osebne intervjuje, fokusne skupine itd. Sekundarni podatki, ki jih zagotovijo vladne agencije, svetovalna podjetja itd., lahko nadomestijo primarne podatke ali jih dodatno dopolnijo. Bolje ko podjetje ali organizacija opredeli svoj ciljni trg in zahteve ter potrebe nika, učinkovitejše bo pri razvoju izdelka ter pri izvajanju uspešne promocijske in trženjske strategije (Curtis in Allen, 2018).

Dejstvo je, da v sodobnem času preživimo vse več časa na spletu, zato je izjemno pomembno, da so spletne strani e-trgovine učinkovite in uspešne, saj to vpliva na rast in uspešnost podjetja. Učinkovitost spletnih strani je odvisna od različnih dejavnikov, kot sta preglednost uporabniškega vmesnika ter količina in kakovost informacij, ki so na voljo na

spletni strani. Medtem ko so se včasih pogosteje uporabljali kvalitativni pristopi z dejavniki ali kriteriji za oceno splošne kakovosti spletnega mesta z e-poslovanjem, pa imamo danes na voljo tudi kvalitativni pristop s pomočjo spletnih analitik, pri čemer je ena najbolj priljubljenih Google Analitika (Glavan, 2022).

Razumevanje zmogljivosti ciljanja na občinstvo s strani Googla se lahko razdeli v štiri ključne kategorije. Prva kategorija vključuje Googlove segmente občinstva, ki temeljijo na podrobnih demografskih podatkih, pripravljenosti za nakup, afiniteti in življenjskih dogodkih. Druga kategorija obsega segmente podatkov porabnikov, ki vključujejo ciljne skupine za ponovno trženje in podobne ciljne skupine. Tretja kategorija zajema segmente po meri, ki jih določajo iskalni izrazi, spletna mesta in aplikacije. Zadnja, četrta kategorija vključuje druge segmente, kot so kombinirani segmenti, optimizirano ciljanje in širjenje občinstva. Te različne kategorije ponujajo trženjskim strokovnjakom raznolike možnosti za prilagajanje svojih pristopov in dosego želenih ciljnih skupin v okviru Googlovih oglaševalskih platform (Gales, 2022).

Pri kategoriji segmenti po meri je še veliko prostora za napredek. Podatki se lahko zbirajo na različne načine. Eden od načinov so tudi posebej izdelana programska orodja, ki vključujejo integrirani razvoj okolja (angl. Integrated Development Environments, v nadaljevanju IDE) , orodja za vizualizacijo programa, samodejno ocenjevanje orodja in spletna učna gradiva. Razdrobljenost podatkov, zbranih s takšnimi orodji, se razlikuje. Na spletu npr. podatki vključujejo študentske oddaje vaj, medtem ko lahko podrobni podatki vključujejo dejanja učencev znotraj IDE, kot je na primer urejanje izvorne kode. Primer zelo natančnih podatkov so podatki o pritisku tipk, ki po navadi vključujejo vsako tipko, pritisnjeno med tipkanjem skupaj s časovnim žigom, ki pove, kdaj točno je bila tipka pritisnjena (Leinonen, 2019).

Vzorec tipkanja po mobilni napravi ali tipkovnici računalnika se razlikuje od posameznika do posameznika. Vsak izmed nas ima svoj slog, ki se lahko kaže na različne načine. Nekateri se izogibajo določenim znakom, spet drugi uporabljajo določene besedne zveze pogosteje in jih posledično tudi natipkajo hitreje kot drugi. Ritem tipkanja lahko z beleženjem in analiziranjem določenih lastnosti tipkanja postane prepoznaven kot ročna pisava ali podpis. Lastnosti tipkanja, ki jih lahko beležimo, so npr. časovni razmik med določenimi tipkami (vmesni čas), število napak in časovni razmik med pritiskom na tipko in sprostitvijo tipke (zadrževalni čas) (Zhao, 2006).

Identifikacija posameznika na podlagi njegove hitrosti in ritma tipkanja se lahko izvede z merjenjem tipkanja na kateri koli tipkovnici in primerjanjem teh podatkov z že zbranimi podatki. Vendar pa je ta tehnika omejena, saj ritem tipkanja posameznika ni odvisen samo od hitrosti njegovega tipkanja, temveč tudi od drugih dejavnikov, kot sta utrujenost ali morebitna alkoholiziranost (Epp in drugi, 2011).

Namen magistrskega dela je s pomočjo strokovne in znanstvene literature opisati obstoječe tehnike klasifikacije porabnikov na podlagi vzorcev tipkanja po računalniški tipkovnici in predstaviti primere uporabe omenjene tehnologije v vsakdanjem življenju. Magistrsko delo je namenjeno podjetjem in oglaševalskim organizacijam za podrobnejšo analizo porabnikov ter bralcem, ki jih tematika zanima. Magistrsko delo zajema več ciljev. Prvi cilj je razviti natančen in zanesljiv merski instrument, ki bo omogočal sistematično analizo vzorcev tipkanja. Drugi cilj je zbrati primerno veliko množico podatkov o tipkanju uporabnikov, ki se bo lahko uporabila še za druge raziskave. S tem želimo zbrati relevantne informacije o vzorcih tipkanja, ki bodo služile kot temelj za nadaljnjo analizo. Glavni cilj je preveriti, ali je mogoče na podlagi teh vzorcev tipkanja uspešno razvrstiti porabnike v specifične skupine ali segmente.

V magistrskem delu se bomo osredotočili na dve raziskovalni vprašanji:

- Ali lahko na podlagi vzorca tipkanja porabnika predvidimo njegove sociodemografske lastnosti?
- Ali lahko oblikujemo skupine (segmentov) porabnikov glede na vzorec tipkanja?

Metodologija raziskave bo temeljila na analizi vzorcev tipkanja, s poudarkom na povezavi med temi vzorci in različnimi značilnostmi uporabnikov, z namenom razvoja naprednih metod ločevanja in analize uporabniških profilov na podlagi tipkanja. Lahko si predstavljamo praktično uporabo takšnih spoznanj na primeru Googlove oglaševalske platforme. Google Ads bi lahko izkoriščal vzorce tipkanja porabnikov v trženske namene, tako da bi analiziral način tipkanja in hitrost, s katero porabniki vnašajo besedilo. Te informacije bi lahko uporabili za boljše razumevanje vedenja in interesov porabnikov, kar bi imelo veliko vrednost pri ciljanju oglasov, prilagojenih njihovim potrebam. Na primer, če porabnik tipka počasi in naredi veliko napak, bi Google lahko predvideval, da je manj samozavesten ali manj informiran o določeni temi. V takem primeru bi lahko Google Ads ciljal oglase, ki ponujajo izobraževalno ali pojasnjevalno vsebino, s čimer bi pomagal porabniku bolje razumeti temo. Enako velja v primeru, ko porabnik tipka hitro in natančno; Google bi lahko sklepal, da ima ta oseba več izkušenj ali znanja o določeni temi, in ciljal oglase, ki so bolj napredni ali specializirani, da bi pritegnili njihovo strokovno znanje. Skratka, vzorci tipkanja nudijo dragocene vpoglede v vedenje in interese porabnikov, kar bi Google Ads lahko izkoristil za izboljšanje učinkovitosti svojih oglaševalskih kampanj.

Magistrsko delo je sestavlja več poglavij, ki sledijo logični strukturi. Začenjamo z opisom obstoječih raziskav in literature na področju zaznavanja vzorcev tipkanja. V tem poglavju predstavljamo relevantno teoretično ozadje ter preučujemo dosedanje dosežke in ugotovitve s tega področja. Sledita opis razvoja merskega instrumenta, ki ga uporabljamo za beleženje vzorcev tipkanja, in postopek pridobivanja podatkovne množice, ki je osnova naše analize. V tem delu predstavljamo metodologijo, uporabljeno za izvedbo eksperimenta in pridobivanje podatkov. Na koncu se osredotočamo na analizo pridobljenih podatkov s pomočjo vizualizacij. Nadalje sledi ovrednotenje rezultatov in izpeljevanje sklepov na

podlagi analize podatkov. Celotna struktura magistrskega dela je zasnovana tako, da bralca vodi skozi celoten proces raziskave od teoretičnega ozadja do praktičnih rezultatov ter ponudi celovit vpogled v obravnavano temo.

2 PREGLED RAZISKOVALNEGA PODROČJA

Računalniki predstavljajo velik tehnološki in znanstveni napredek zadnje polovice dvajsetega stoletja. Računalnik je spremenil način, kako delamo, kako organiziramo in shranjujemo informacije, kako komuniciramo med seboj, ter celo način, kako razmišljamo o vesolju in človeškem umu. Računalniki so olajšali utrujajoče računanje in pisarniško delo ter postali bistvena orodja v vseh vejah tehnoloških industrij in vsakdanjem življenju. Postali so vseprisotni v mnogih vidikih poslovanja, rekreacije in vsakdanjega življenja in trend je, da postajajo močnejši, pogostejši in enostavnejši za uporabo (Swedin in Ferro, 2007).

Sodobna družba, ki je vse bolj odvisna od tehnologije in globalno povezana, se sooča z naraščajočimi grožnjami na področju varnosti računalniških sistemov. V zadnjem času smo priča izjemnemu porastu elektronskih napadov prek spleta, pri čemer strokovnjaki napovedujejo, da bodo napadalci v prihodnosti uporabljali še bolj inovativne strategije. Ti napadi niso omejeni zgolj na poskuse vdora v sisteme, temveč so usmerjeni v poškodovanje računalniških infrastruktur, nezakonit dostop do občutljivih podatkov ter prestrezanje ključnih informacij. Ta kompleksna problematika ima obsežne posledice, saj se organizacije in posamezniki znajdejo v vse bolj ogroženem položaju, postajajo tarče napadalcev in izpostavljeni izgubi kritičnih informacij. To poudarja nujnost stalnega izboljševanja varnostnih strategij ter nenehnega prilagajanja novim izzivom, s katerimi se soočamo v digitalnem okolju. Zato je ključno, da se vzpostavijo inovativne in učinkovite rešitve, ki bodo zagotavljale zanesljivo zaščito pred sodobnimi grožnjami, ter da se razvijejo mehanizmi za hitro odzivanje in obvladovanje morebitnih varnostnih incidentov (Mijwil in drugi, 2023).

Biometrija, ki vključuje fizične in vedenjske značilnosti posameznika, postaja priljubljena metoda za preverjanje identitete. Ena glavnih prednosti biometričnih značilnosti je v njihovi neizgubljivosti ali neukradljivosti, kar jih naredi potencialno zanesljive za določanje identitete. Na primer, prstni odtisi, kot fiziološke značilnosti, so edinstveni v velikem delu populacije in so primerni za preverjanje. Poleg prstnih odtisov se uporabljajo tudi drugi biometrični znaki, kot so oblika roke, termalni vzorci v obrazu, vzorci krvnih žil v mrežnici in roki, glasovni odtisi ter pisni podpisi. Biometrija se vse bolj kombinira z drugimi metodami, kot so gesla ali identifikacijske kartice, kar povečuje stopnjo varnosti, na primer pri odklepanju mobilnih telefonov. Kljub temu so nekatere biometrične tehnike predrage ali dovzetne za prevaro. Na primer, dinamika tipkanja, kot vedenjska biometrija, analizira način tipkanja in se lahko uporablja za identifikacijo posameznikov. Časovne informacije o tipkanju lahko zbiramo programsko, kar omogoča stroškovno učinkovito avtentikacijo in klasifikacijo uporabnikov, vendar pa nestabilnost te tehnologije predstavlja izziv, saj je način

tipkanja odvisen od različnih nevropsihioloških dejavnikov, kot so čustva, stres, dremavost in podobno (Idrus in drugi, 2013).

2.1 Obstoječe raziskave na področju zaznavanja vzorcev tipkanja

Uporabnikova dinamika pritiska na tipke predstavlja inovativno področje, ki že ima praktične aplikacije na različnih področjih. S hitrim napredkom raziskav na tem področju lahko pričakujemo še številne inovacije, ki bodo dodatno obogatile različna področja uporabe dinamike pritiska na tipke. V tem poglavju bomo temeljito preučili že izvedene raziskave na področju tipkanja uporabnikov, pri čemer se bomo osredotočili na štiri ključne vidike: avtentikacijo in klasifikacijo, zaznavanje čustev, sledenje zdravju ter uporabo zaznavanja tipkanja uporabnikov v kontekstu trženja. Avtentikacija predstavlja ključen vidik uporabe dinamike pritiska na tipke za zagotavljanje varnosti in identifikacije posameznikov. Raziskave na tem področju se osredotočajo na razvoj naprednih modelov, ki temeljijo na značilnostih tipkanja, s čimer omogočajo zanesljivo prepoznavo uporabnikov in preprečujejo nepooblaščen dostop (Magnusson, 2023).

Klasifikacija razvršča uporabnike v predhodno določene skupine na podlagi njihovih lastnosti (Jain in drugu, 2004). Poleg tega se v zadnjem času vse bolj razvija raziskovanje zaznavanja čustev prek tipkanja, pri čemer se analizira ritmičnost in slog tipkanja kot kazalnika čustvenega stanja posameznika. S sledenjem zdravja prek tipkanja se raziskujejo povezave med fiziološkimi stanji in načinom tipkanja, kar odpira možnosti za neinvazivno spremljanje zdravstvenih parametrov (Tripathi in drugi, 2022).

Na področju trženja pa se zaznavanje tipkanja uporablja za analizo vedenja porabnikov, prilagajanje trženjskih strategij ter izboljšanje personalizacije storitev (Patel, 2023). S tem poglobljenim pregledom raziskav želimo razumeti širši kontekst in potencial, ki ga dinamika pritiska na tipke uporabnikov prinaša na omenjenih področjih, ter identificirati morebitne smeri nadaljnjih raziskav in inovacij.

2.1.1 Avtentikacija uporabnikov

Avtentikacija je postopek potrditve, da je uporabnik tisti, za katerega se izdaja. Preden uporabnik poskuša dostopati do informacij v omrežju, mora zagotoviti poverilnice (angl. credentials), da dokaže svojo identiteto. S tem postopkom zagotavljamo varnost sistemov. Preverjanje pristnosti nam omogoča, da z zaupanjem dodelimo dostop pravemu uporabniku ob pravem času. Preverjanje prisotnosti za dostop do digitalnih virov je sestavljen iz naslednjih treh korakov. Prvi korak je identifikacija, kjer uporabnik pove za koga se izdaja. Avtentikacija ni izvedljiva samo s pomočjo uporabniškega imena, saj sistem nima zagotovila, ali to določeno uporabniško ime resnično pripada uporabniku, za katerega se izdaja. Drugi korak je preverjanje pristnosti, ko uporabnik dokazuje svojo identiteto. Uporabnik navede uporabniško ime skupaj z geslom ali drugimi poverilnicami za

preverjanje. Tretji in zadnji korak pa predstavlja pooblastilo, kjer uporabnik dokazuje, da ima dovoljenje za dostop. V zadnjih nekaj letih pa so se grožnje kibernetske varnosti drastično povečale, zato večina organizacij za večplastno varnost uporablja dodatne kriterije preverjanja (Magnusson, 2023).

Ena od njih je tudi avtentikacija z dinamiko pritiska na tipko, ki je večinoma dosežena z uporabo modelov za prepoznavanje vzorcev. Najpogostejše uporabljeni modeli so statični modeli, nevronske mreže, fuzzy logika (angl. Fuzzy logic) in metode podpornega vektorskega stroja (angl. support-vector machines) (Buza in Farou, 2019).

Na področju dinamike pritiska na tipke v povezavi z avtentikacijo je bilo izpeljanih že mnogo raziskav. Gaines in drugi (1980) so izvedli poskuse s sedmimi sekretarji, ki so morali v obdobju štirih mesecev dvakrat prepisati iste tri odstavke. Pri poskusih so zbirali podatke o časih zakasnitve pri tipkanju za omejeno število digrafov (angl. digraphs) ter jih nato primerjali in analizirali. Osnovo analize so predstavljali digrafi, ki so se pojavili več kot 10-krat. Pri tem so izvedli preizkus dvojic. Glavna domneva je temeljila, da sta srednji vrednosti časov digrafov enaki pri obeh sejah in da sta varianci enakovredni. Kljub spodbudnim rezultatom so ugotovili, da je bila velikost vzorca premajhna ter da je bila količina potrebnih podatkov za izgradnjo referenčnih profilov nesprejemljiva (Gaines in drugi, 1980).

Podoben eksperiment sta izvedla Leggett in Williams (1988), ki je potrdil rezultate raziskave Gaines in drugi. Podobno kot pri prejšnjih raziskavah pa je glavno omejitev predstavljala količina potrebnih podatkov. Odkrila sta, da je pri vsakem udeležencu potrebnih več kot 1000 besed za dovolj natančno identifikacijo. Tak statični avtentikacijski sistem se v praksi ne bi dobro obnesel (Leggett in Williams, 1988).

Monrose in Rubin (1997) sta preučevala učinkovitost avtomatskih metod za prepoznavanje uporabnikov na podlagi dinamike tipkanja. Pri svojem pristopu sta uporabila tri različne klasifikatorje. Prvi je temeljil na kvadratni evklidski razdalji, pri kateri so neznan profil povezali z referenčnim profilom v podatkovni zbirki, ki ima najkrajšo razdaljo do vseh referenčnih profilov v podatkovni zbirki. Drugi klasifikator je temeljil na verjetnostni oceni. Predpostavila sta normalno porazdelitev za vsako značilnost v vzorcu in jo uporabila za določanje najbolj verjetnega referenčnega profila za neznan profil. Zadnji klasifikator je vključeval optimizacijo klasifikatorja, ki je temeljil na verjetnostni oceni, s poudarkom na najbolj zanesljivih značilnostih s pomočjo uteži (angl. weights). Uteži so poudarjale klasifikacijsko moč zanesljivejših značilnosti. Izkazalo se je, da je bila pravilnost prepoznavanja z uporabo uteženega verjetnostnega klasifikatorja približno 90 %. To je predstavljalo izboljšanje v primerjavi z drugimi klasifikatorji. Rezultati so bili prilagodljivi in odvisni od frekvenčne porazdelitve posameznih značilnosti (Monrose in Rubin, 1997).

Prepoznavanje določenega uporabnika in razlikovanje med različnimi uporabniki na podlagi statistike njihovega tipkanja so poskusili z arhitekturo nevronskih mrež razrešiti Maheshwary in drugi (2017). Preizkusili so njihov model z uporabo podatkovne zbirke

univerze Carnegie Mellon (angl. Carnegie Mellon University – CMU) merila dinamike pritiskov tipk (angl. keystroke dynamics benchmark) in primerjali njegovo učinkovitost z obstoječimi metodami. Njihov model je dosegel stopnjo napake 0,03 in celotno natančnost 93,59 %, kar kaže na visoko učinkovitost (Maheshwary in drugi, 2017).

Tudi na trgu obstajajo različni izdelki, ki uporabljajo za avtentikacijo uporabnikov dinamiko pritiska na tipke, a zaradi zaprtosti virov ni veliko podatkov o njih. Spodaj je seznam nekaterih znanih izdelkov, za katere so znane nekatere informacije o njihovih izvedbah (Buza in Farou, 2019):

- TypeWATCH, ki ga je izdala programska oprema Watchful, brezplačna programska oprema za tipkanje besedilnih vzorcev,
- Intensity analytics uporablja statistične uteži in meritve,
- BioTracker, ki ga je izdal Pluriloc, ki hkrati spremlja tudi gibanje miške,
- KeyTrac, ki analizira vsak vnos besedila v ozadju.

2.1.2 Klasifikacija uporabnikov

Klasifikacija je proces razvrščanja objektov ali podatkov v predhodno določene kategorije na podlagi njihovih lastnosti in značilnosti. Glavni cilj klasifikacije je zgraditi model, ki lahko samodejno določi kategorijo, v katero spada določeni objekt glede na njegove lastnosti. To se doseže z uporabo algoritmov, ki se naučijo razlikovati med različnimi kategorijami na podlagi podatkov za učenje. Klasifikacija se uporablja za različne namene. Pogosto je uporabljena v aplikacijah, kot so prepoznavanje spola, starosti, obrazov, klasifikacija besedil, prepoznavanje zvoka in celo pri diagnozi zdravljenja. V tem poglavju bomo predstavili raziskave, v katerih so uporabnike klasificirali v določene skupine glede na način tipkanja po tipkovnici.

Razlog, zakaj je učinkovitost dinamike tipkanja po tipkovnici slabša v primerjavi z drugimi standardnimi modalitetami biometričnih sistemov, je v variabilnosti uporabnikovega vedenja. Na uporabnikovo počutje lahko vplivajo različni dejavniki, kot so stres, strah, utrujenost, uživanje substanc in še mnogo drugih. To povzroči, da se uporabnikovo tipkanje lahko razlikuje v različnih situacijah, kar lahko vpliva na natančnost prepoznavanja in s tem zmanjša učinkovitost sistema. Ena rešitev za spopadanje s to variabilnostjo je preučevanje mehkih biometričnih značilnosti, ki so jih prvič predstavili Jain in drugi (2004). Mehke biometrične značilnosti so opredelili kot značilnosti, ki lahko zagotavljajo nekaj informacij o posamezniku, vendar nimajo dovolj razločnosti in trajnosti, da bi bile same po sebi zadostne za razlikovanje med dvema posameznikoma. Obravnavajo jih kot dopolnilne podatke pri biometričnem sistemu, ki temelji na odtisu prsta, kot so spol, etnična pripadnost in višina. Mehke biometrične značilnosti lahko izboljšajo učinkovitost iskanja pravega uporabnika v bazi podatkov, kar lahko zmanjša čas izračuna. Na primer, če se mehki

biometrični modul ujema z moškim, lahko standardni biometrični avtentikacijski sistem omeji iskanje le na moške uporabnike, ne da bi upošteval ženske (Jain in drugi, 2004).

Prepoznavanje spola je obravnavano v raziskavi Giot in Rosenberger (2012), kjer sta pokazala, da je mogoče z načinom tipkanja fiksnega besedila zaznati spol posameznika. Stopnja natančnosti prepoznavanja spola je več kot 90 % (Giot in Rosenberger, 2012).

Delo Idrus in drugi (2013) prikazuje, da je mogoče z enim prstom, z eno roko, in več kot enim prstom, z obema rokama, prepoznati način tipkanja uporabnikov z 80 % natančnostjo prepoznavanja na nizu podatkov s tremi gesli. Predstavili so nov pristop k razumevanju načina tipkanja, ki tudi uporablja mehke biometrične značilnosti. Te biometrične značilnosti so dostopne vsem, kar jih naredi varne za zasebnost. Pristop vključuje zbiranje informacij o tipkanju, kot so roka, spol, starostna kategorija in rokopis uporabnika pri vnašanju določenega gesla na tipkovnici. Poskusi so bili izvedeni na temelju dinamike tipkanja 110 uporabnikov. Iz rezultatov je razvidno, da prepoznavanje števila uporabljenih rok deluje s 94 % natančnostjo, prepoznavanje spola deluje z 78 % natančnostjo, prepoznavanje starosti pod in nad 32 let deluje z 69 % natančnostjo in prepoznavanje rokopisa uporabnika deluje s 73 % natančnostjo (Idrus in drugi, 2013).

Uzun in drugi (2015) so z uporabo biometričnih značilnosti za prepoznavanje posameznikov dokazali, da je dinamika pritiska tipk uspešno uporabljena za napovedovanje, ali je starost uporabnika manjša od 15 let. Rezultati raziskave kažejo, da so dosegljive natančnosti nad 90 % za predvideno izbiro metodologije klasifikacije (Uzun in drugi, 2015). Namen raziskave je bil predvsem zaščita otrok pred grožnjami na spletu. V raziskavi, izvedeni v Indiji, je bilo ugotovljeno, da je 67 % otrok, mlajših od 10 let, imelo račun na Facebooku in 82 % od njih jih je prejelo neprimerna sporočila (Variyar, 2013).

V nadaljevanju so opisane raziskave, ki so se osredotočale na prepoznavanje čustev na podlagi načina tipkanja na tipkovnici. Te raziskave so se srečevale številnimi izzivi, kot so ustvarjanje čustev za namen raziskave, zbiranje in označevanje podatkov, definiranje in izračunavanje značilnih lastnosti ter na koncu usposabljanje in preizkušanje modelov. Različne metode so razvite za prepoznavanje različnih čustvenih stanj, v nekaterih primerih pa se prepozna več čustev hkrati. Druge raziskave se osredotočajo na prepoznavanje le enega izbranega čustvenega stanja.

Vizer in drugi (2009) so izvedli eksperiment prepoznavanja stresnih stanj na podlagi značilnosti pritiskov na tipkovnico. Za sprožanje stresa so udeleženci dobili določene stresne naloge. Pridobljene podatke so kategorizirali z uporabo različnih tehnik strojnega učenja, kot so SVM, k-NN, nevronske mreže in odločitvena drevesa. Rezultati so pokazali, da je bilo mogoče prepoznati kognitivni in fizični stres s 75 % oziroma 62,5 % točnostjo. Poleg tega so avtorji pokazali, da obstaja močna povezava med čustvenim stanjem in uporabo določenih tipk, kot so tipke za vračanje, brisanje, konec, puščice, ter dolžino premora in časom na pritisk (Vizer in drugi, 2009).

Epp in drugi (2011) so preučevali prepoznavanje čustvenih stanj med običajnimi računalniškimi dejavnostmi, kot so pisanje besedil in pošiljanje e-pošte. Stres je bil le eno od petnajstih čustvenih stanj, ki so jih prepoznali. Z uporabo odločitvenih dreves so lahko z visoko natančnostjo (do 87,8 %) prepoznali nekatera čustva, kot so zaupanje, oklevanje, živčnost, sproščenost, žalost in utrujenost, vendar stres ni bil med najbolj prepoznavnimi čustvenimi stanji (Epp in drugi, 2011).

2.1.3 Sledenje zdravju

Parkinsonova bolezen je druga najpogostejša nevrodegenerativna motnja na svetu. Ob pravočasnem diagnosticiranju bolnika je mogoče izvesti klinične raziskave za nevroprotektivne terapije, ki lahko upočasnijo napredovanje bolezni. Nedavna raziskovanja so razkrila slikovne in krvne označevalce, ki bi lahko bili v pomoč pri prepoznavanju Parkinsove bolezni pri bolnikih. Idiopatska narava Parkinsove bolezni je otežila uporabo testov v splošni populaciji, kar je bil razlog za raziskovanje v smeri izdelave enostavnega orodja, ki bi omogočilo presojanje ključnih znakov bolezni pri splošni populaciji.

Tripathi in drugi (2022) so preučevali uporabo dinamike pritiska na tipkovnici za odkrivanje Parkinsove bolezni. Predlagani model doseže od 80 % do 83 % natančnost na skupini subjektov, ki so aktivno uporabljali svoje računalnike vsaj 5 mesecev, pri čemer so se redno zbirali njihovi podatki o dinamiki pritiska na tipke (Tripathi in drugi, 2022).

2.1.4 Uporaba zaznavanja vzorcev tipkanja v trženju

Trženje je eden od ključnih poslovnih procesov, ki vključuje raziskovanje, razvoj in izvajanje strategij za promocijo izdelkov ali storitev ter pridobivanje in ohranjanje porabnikov. Cilj trženja je zadovoljiti potrebe in želje ciljne skupine, ustvariti vrednost za porabnike ter doseči uspeh na trgu, kar vključuje različne aktivnosti, kot so trženjske raziskave, oglaševanje, prodaja in management odnosov s porabniki (Kotler in drugi, 2015).

Trženje in spletno trženje sta tesno povezana koncepta v sodobnem poslovnem svetu. Spletno trženje, kot del širšega trženjskega okvira, izkorišča spletna orodja za promocijo izdelkov ali storitev. To dinamično področje nudi podjetjem številne možnosti za dosego ciljnega občinstva prek spleta in učinkovito izvajanje trženjskih strategij. Med različnimi oblikami spletnega trženja so (Patel, 2023):

- optimizacija spletnih strani za iskalnike (angl. Search Engine Optimization, v nadaljevanju SEO),
- trženje prek e-pošte,
- trženje prek iskalnikov (angl. Search Engine Marketing, v nadaljevanju SEM),
- oglaševanje s plačilom na klik (angl. pay-Per-Click Advertising, v nadaljevanju PPC),
- plačano oglaševanje prek Googla ter trženje prek družbenih medijev.

SEO povečuje promet prek organskih iskalnih rezultatov, zagotavlja relevantnost vsebine za ciljno občinstvo in se pogosto izvaja s pomočjo Googla. Trženje prek e-pošte vključuje pošiljanje sporočil porabnikom za ohranjanje obstoječih in pridobivanje novih porabnikov. SEM omogoča hitro promocijo s plačilom za prikaz rezultatov iskanja, pri čemer se oglašuje na platformah, kot sta Google in Facebook. PPC je plačana oblika oglaševanja, kjer se trženjski stroški merijo po številu klikov na oglas. Oglaševanje na Googlu je plačana kampanja, ki se prikaže ob iskanju določenih ključnih besed. Spletno trženje prek družbenih medijev vključuje organski in plačani pristop, kjer gradnja odnosov na platformah, kot sta Facebook in Twitter, povečuje zvestobo, medtem ko plačane kampanje omogočajo ciljanje določenega občinstva. Različne oblike spletnega trženja omogočajo podjetjem prilagodljive strategije za dosego ciljev in ciljnega občinstva (Patel, 2023).

Uporaba biometrične tehnologije je postala izjemno pomembna na spletu, saj omogoča zanesljivo in varno preverjanje identitete posameznikov. S tem pa se odpirajo nove možnosti in potenciali za uporabo biometrije pri vsakodnevnih aktivnostih. Z uporabo te tehnologije bi tržniki lahko ciljali na porabnike, saj se lahko natančno beležijo in analizirajo porabnikove aktivnosti na spletni strani. To ima velike posledice za razvoj učinkovitih tržnih strategij za spletne porabnike in koristi porabnikom tako, da se zmanjša količina nepotrebnih in neprimernih informacij, ki jih prejmejo. Natančne, personalizirane informacije bi tržnikom omogočile natančnejši profil posameznikov in zmanjšanje napačne oglaševalske vsebine. Če se to izvaja spoštljivo in etično, bi lahko celotna trženjska strategija postala bolj preprosta, kar bi prineslo večjo vrednost za obe strani transakcije. Uspeh upravljanja s porabniki v zadnjem desetletju je močan dokaz, da lahko osebno obravnavanje izboljša konkurenčno prednost organizacije na trgu. Biometrična tehnologija je visoko robustna, stroškovno učinkovita, preprosta za uporabo in varnost, a na drugi strani jo zaznamujeta nizka sprejetost in slaba prenosljivost. V bližnji prihodnosti pa bodo te težave premagane, saj lahko miška in tipkovnica vključujeta standardne biometrične naprave, ki bi znižale stroške in povečale prenosljivost teh naprav. Robustnost oz. zaupanje pa se bo širilo, ko se bodo zaupanja vredne tretje osebe uveljavile med porabniki in prodajalci (Pons, 2006).

Kot je opisano v prejšnjih raziskavah, se da na podlagi vzorca tipkanja dokaj natančno določiti spol, starost, število uporabljenih rok, ali je posameznih levičar ali desničar in celo čustva posameznika. Monem (2021) je raziskovala učinkovitost personalizacije oglaševanja. Raziskava je dokazala, da personalizacija pozitivno vpliva na učinkovitost oglaševanja glede na starost, potrebe in interese porabnikov. Privlačnost, prepoznavnost, zapomnljivost in simpatije se povečajo, ko je sporočilo dostavljeno neposredno in personalizirano. Čas in kontekst vplivata na interakcijo udeležencev z oglasi (Monem, 2021).

3 RAZVOJ PROGRAMSKE OPREME

V tem delu bomo opisali tehnologije, programe in koncepte, ki smo jih uporabili pri izdelavi merskega instrumenta za zajemanje podatkov pri beleženju dinamike pritiska na tipke. Pri izbiri teh tehnologij smo upoštevali njihovo združljivost in primernost za vse funkcije, ki jih je zahtevala uporaba merskega instrumenta.

3.1 Programski jeziki in orodja

Prvi programski jezik, ki smo ga uporabili pri implementaciji merskega instrumenta, je Java. Java je široko uporabljen programski jezik za kodiranje spletnih aplikacij. Danes se v praksi uporablja več milijonov javanskih aplikacij. Java je večplatformni, objektno orientiran in mrežno naravnan jezik, ki se lahko uporablja kot platforma sam po sebi. Gre za hiter, varen in zanesljiv programski jezik za kodiranje vsega, od mobilnih aplikacij in poslovne programske opreme do aplikacij za obdelavo velikih podatkov in tehnologij na strežniški strani. Programski jezik Java je vsestranski in brezplačen jezik, ki se lahko uporablja za različne namene. Pogosto se uporablja za razvoj iger, razvoj aplikacij, računalništvo v oblaku, obdelavo velikih količin podatkov, umetno inteligenco in splet stvari (angl. Internet of Things – IOT) (Amazon Web Services, brez datuma). V magistrskem delu smo ga uporabili kot jezik za razvoj merskega instrumenta za zajemanje podatkov načina tipkanja. Aplikacijo smo razvili v orodju Eclipse, ki je integrirano razvojno okolje za razvijanje aplikacij z uporabo programskih jezikov Java in drugih programskih jezikov, kot so C/C++, Python, PERL, Ruby itd. Platforma Eclipse, ki predstavlja temelj za Eclipse IDE, je sestavljena iz vtičnikov (angl. plug-ins). Zasnovana je tako, da se lahko razširi z dodatnimi vtičniki (Tutorialspoint, brez datuma).

Drugi jezik, ki smo ga uporabili, je programski jezik Python. Python je interpretiran, objektno usmerjen, visokonivojski programski jezik z dinamično semantiko, ki ga je razvil Guido van Rossum. Izvorno je bil objavljen leta 1991. Ime "Python" je bilo izbrano kot poklon britanski komični skupini Monty Python. Python je programski jezik, namenjen začetnikom, in nadomešča programski jezik Java kot najpogosteje uporabljeni uvodni jezik, saj se uporabniki lahko osredotočijo na popolno razumevanje programskih konceptov namesto podrobnosti v kodi. Python se uporablja za različne namene, kot so spletni razvoj na strežniku, razvoj programske opreme, matematika in sistemske skripte, ter je priljubljen za hitri razvoj aplikacij. Stroški vzdrževanja programov se s Pythonom zmanjšujejo zaradi enostavne sintakse in poudarka na berljivosti. Poleg tega Pythonova podpora modulom in paketom omogoča modularne programe in ponovno uporabo kode. Python je jezik skupnosti z odprtokodno licenco, zato številni neodvisni programerji neprestano gradijo knjižnice in funkcionalnosti za njega (Teradata, brez datuma).

V magistrski nalogi smo ga uporabili za pripravo podatkov v okolju Visual Studio Code. Visual Studio Code je brezplačen, lahek, vendar zmogljiv urejevalnik izvorne kode, ki deluje

na namizju in na spletu ter je na voljo za Windows, macOS, Linux in Raspberry Pi OS. Ima vgrajeno podporo za JavaScript, TypeScript in Node.js ter bogato ekosistemsko razširitev za druge programske jezike, kot so C++, C#, Java, Python, PHP in Go, okolja, kot sta Docker in Kubernetes, in oblaki, kot so Amazon Web Services, Microsoft Azure in Google Cloud Platform (Heller, 2022).

Zadnji jezik, ki smo ga uporabili, je programski jezik R. Jezik R je programska platforma, namenjena predvsem statističnemu računalništvu in ustvarjanju grafik. Ena od njegovih ključnih značilnosti je njegova brezplačna dostopnost, kar uporabnikom omogoča, da ga namestijo in izvajajo na različnih operacijskih sistemih, vključno z Windows, Mac OS X in Linux. R se razlikuje od bolj tradicionalnih programskih jezikov, kot sta C++ ali Java, in izstopa kot nekonvencionalen jezik. Posebna značilnost je tudi njegovo delovanje kot interaktivno statistično okolje, kar uporabnikom omogoča učinkovito analizo podatkov. Poleg tega R omogoča uporabo podčrtajev kot spremenljivk, kar je drugače od drugih jezikov, ki za to uporabljajo operaterje dodeljevanja. Zaradi svojih lastnosti je R postal priljubljen med znanstveniki podatkov, ki ga pogosto uporabljajo pri raziskavah in analizah (R (programski jezik), 2022). V magistrskem delu smo programski jezik R uporabljali za analizo pripravljenih podatkov.

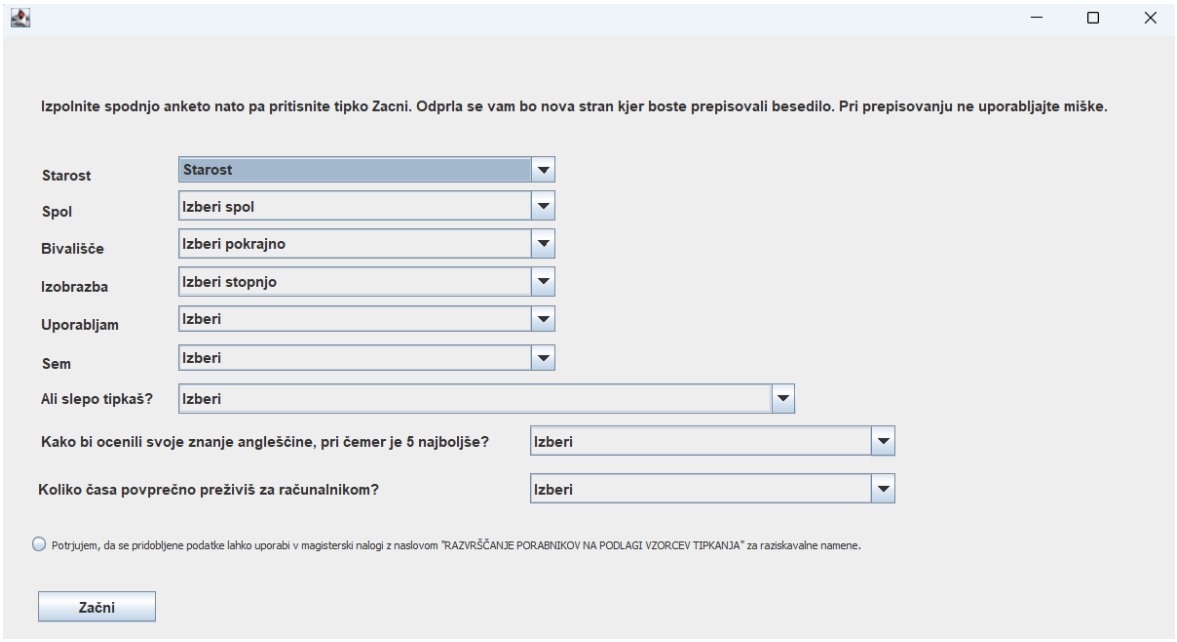
3.2 Razvoj merskega instrumenta

Prvi korak, ki smo ga naredili, je bil, da smo preverili, ali podoben merski instrument že obstaja in nam je dostopen. Odkrili smo, da merski instrument, ki ga potrebujemo za namen pridobivanja podatkov, še ne obstaja ali pa ni dostopen prek spleta, zato smo merski instrument implementirali sami. V nadaljevanju sledi opis merskega instrumenta in njegovega delovanja ter opis besedila, ki smo ga pripravili za prepisovanje.

3.2.1 Opis merskega instrumenta

Merski instrument smo razvili z orodjem Eclipse v jeziku Java. Celotna koda je dostopna na Github repozitoriju, ki je dostopen prek spletne povezave na naslovu https://github.com/AndrejMohoric/Keyboard-keystroke-dynamics-.git. Merski instrument smo uporabili za zajem dovolj velike količine podatkov o tem, kako uporabniki tipkajo. Program sestavlja uporabniški vmesnik, ki je razdeljen na dve strani. Vprašalnik je zajemal 9 vprašanj. Na prvi strani je uporabnik odgovarjal na sociodemografska vprašanja, kot so starost, spol, bivališče, stopnja izobrazbe, koliko časa uporabnik povprečno preživi za računalnikom dnevno, ali uporabnik slepo tipka, ali pri tipkanju uporablja eno ali dve roki, ali je desničar ali levičar in osebna ocena znanja angleščine. To je razvidno iz slike 1.

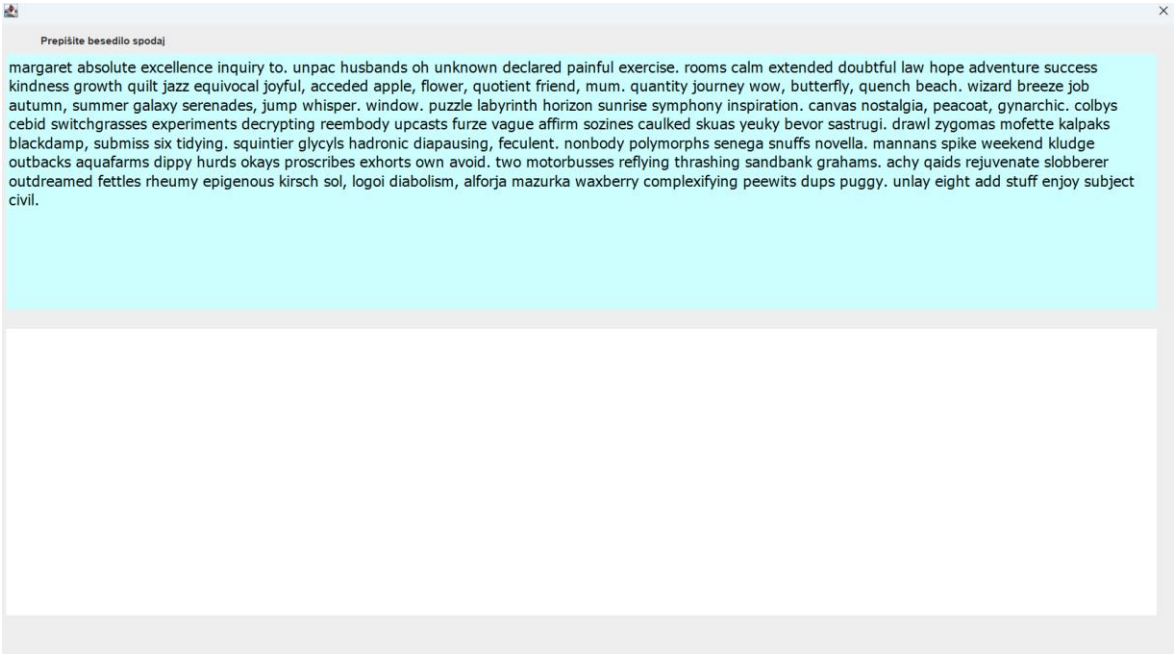
Slika 1: Prva stran uporabniškega vmesnika merskega instrumenta



Vir: lastno delo.

Na drugi strani uporabniškega vmesnika pa je bilo od uporabnika zahtevano, da pretipka predhodno definirano besedilo. Vsi uporabniki so prepisovali enako besedilo, ki je bilo v angleščini. Uporabniški vmesnik druge strani je prikazan na sliki 2.

Slika 2: Druga stran uporabniškega vmesnika merskega instrumenta

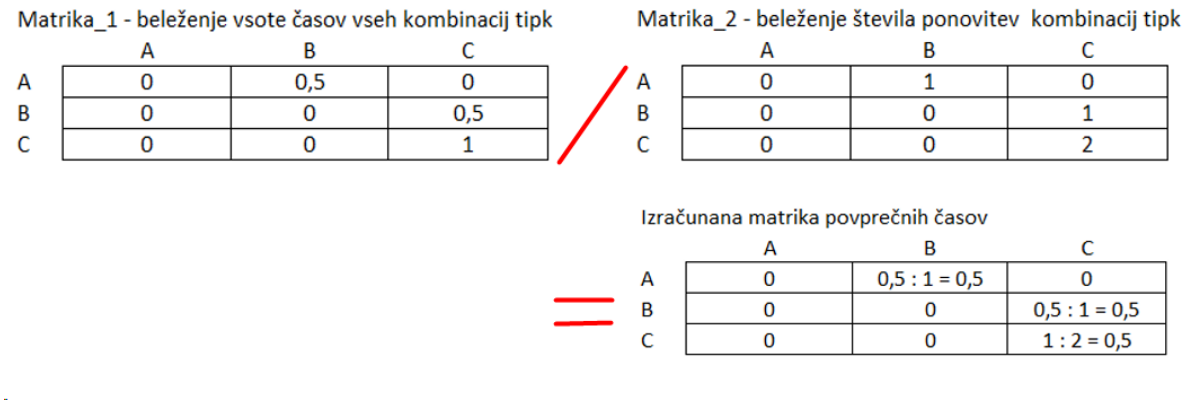


Vir: lastno delo.

Več o izbiri in vsebini besedila bo predstavljeno v poglavju Besedilo za prepisovanje. Program je beležil časovne razmike posameznih kombinacij črk angleške abecede (26 črk) in znakov, kot so pika vejica in presledek. Torej vse skupaj 29 tipk na tipkovnici. Časovne vrednosti so se za vsakega uporabnika posebej zapisovale v 29 x 29 veliko matriko. Lokacija kombinacije dveh zaporednih pritiskov tipk v matriki je bila določena tako, da je vsaka vrstica predstavljala prvo pritisnjeno tipko, vsak stolpec pa je predstavljal drugo pritisnjeno tipko iz dane kombinacije tipk. V določeno polje se je beležil čas med pritiskom prve tipke in pritiskom druge tipke. Ob večkratni ponovitvi pritiska enake kombinacije se je čas v polju seštevал. To matriko bomo v nadaljevanju imenovali »Matrika 1«. Hkrati je program beležil dodatno 29 x 29 veliko matriko, kjer so imeli stolpci in vrstice enak pomen kot v »Matriki 1«, le da smo namesto časovnih razmikov med tipkami beležili število ponovitev posamezne kombinacije. To matriko bomo v nadaljevanju imenovali »Matrika 2«. Iz matrik nato izračunamo povprečno matriko uporabnika tako, da delimo polja »Matrike 1« s polji »Matrike 2«.

Za poenostavljen primer si lahko predstavljamo matriko 3 x 3, kjer stolpci predstavljajo črke A, B in C in vrstice predstavljajo črke A, B in C, v enakem vrstnem redu. Če uporabnik pretipka črke ABCCC v enakem časovnem razmiku (npr. 0,5 sekunde), se bodo izpolnila naslednja štiri polja, za kombinacije AB, BC, CC in CC v »Matriki 1« in »Matriki 2«. Nato se izračuna še povprečna matrika, kot je prikazano na sliki 3.

Slika 3: Poenostavljeni primer izračuna matrike povprečnih časov



Vir: lastno delo.

Zadnja meritev, ki jo merimo pri posameznem uporabniku, pa je število napak. Program deluje tako, da se gumb »Končaj«, ki uporabniku omogoči, da konča preizkus, prikaže šele tedaj, ko uporabnik pravilno pretipka celotno podano besedilo. Med tipkanjem se uporabnik lahko zmoti in svojo napako pobriše s pritiskom na tipko Backspace. Prvi način, s katerim program beleži število napak, je število, kolikokrat je porabnik pritisnil na tipko Backspace, saj to odraža število napak. Poleg tega pa beležimo tudi število pritiskov vsake tipke posebej v matriki velikosti 1 x 29. Vsi uporabniki prepisujejo enako besedilo, torej bomo lahko iz zadnje matrike pridobili podatke, kolikokrat se je posamezni uporabnik zmotil pri določeni

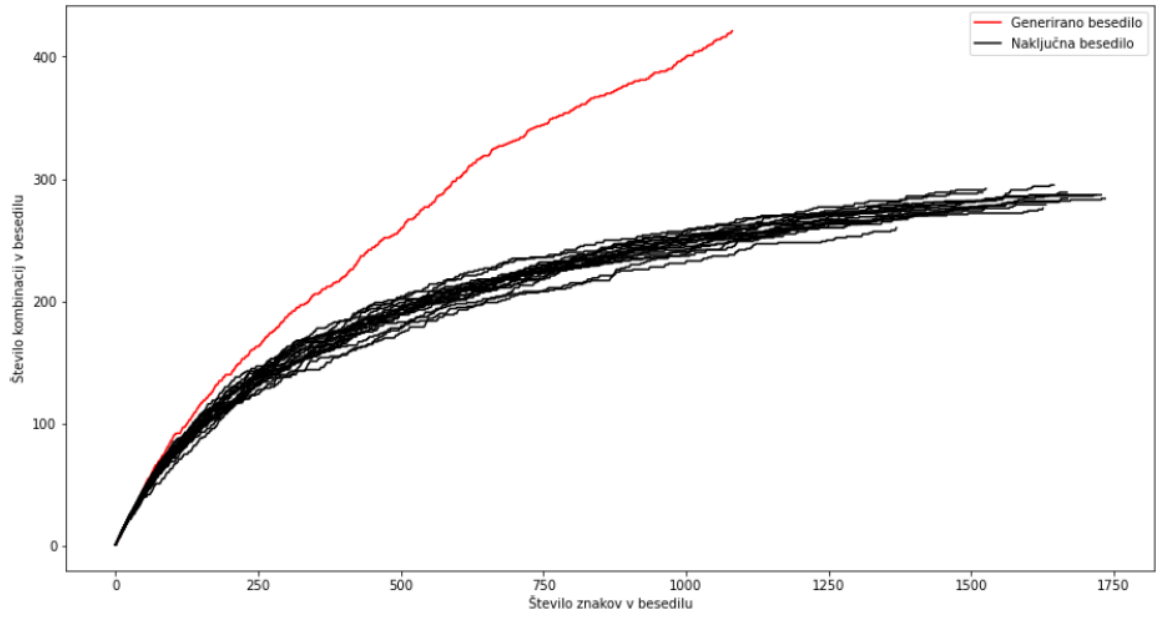
tipki. Za primer lahko predpostavljamo, da je v besedilu 60 ponovitev črke A. V matriki je pri določenem uporabniku vrednost v prvem stolpcu, ki predstavlja število pritiskov tipke A, enaka 64. Na podlagi teh podatkov lahko sklepamo, da se je uporabnik štirikrat zmotil pri pritisku na tipko A, saj je število pritiskov večje od pričakovanega števila ponovitev črke A v besedilu.

3.2.2 Besedilo za prepisovanje

Besedilo, ki ga uporabnik prepisuje v programu, je sestavljeno iz 137 angleških besed. Besedilo smo generirali sami s pomočjo Python skripte, ki smo jo zaganjali v Visual Studio Code orodju. Cilj generiranega besedila je bil, da je besedilo čim krajše in zajema čim višje število različnih kombinacij tipk. V končni različici besedilo zajema 421 različnih kombinacij tipk. To predstavlja 50,06 % vseh kombinacij tipk, ki jih merski instrument beleži (29 x 29 = 841 je vseh polj). Python skripto lahko najdete na spletni povezavi https://andrejmohoric.github.io/generate-text-with-a-lot-of-char-combinations/. Gre za preprost algoritem, ki v zanki kliče API za pridobitev naključne angleške besede. Če beseda zajema novo kombinacijo znakov, jo doda v seznam. Ko se zanka konča, nam skripta vrne vse besede, ki smo jih dodali v seznam. Seznam besed nato združimo v spremenljivko tipa beseda in ročno dodamo ločila, kot so pike in vejice na ustreznih mestih, tako da povečamo število kombinacij tipk v besedilu. Generirano besedilo smo primerjali z 20 drugimi naključnimi besedili, ki smo jih pridobili s pomočjo spletne aplikacije Random text generator (Random text generator, brez datuma).

Iz rezultatov na sliki 4 je razvidno, da ima naše generirano besedilo (označeno z rdečo barvo) višje število različnih kombinacij in je hkrati krajše od preostalih besedil.

Slika 4: Primerjava besedil po številu kombinacij in število znakov v besedilu

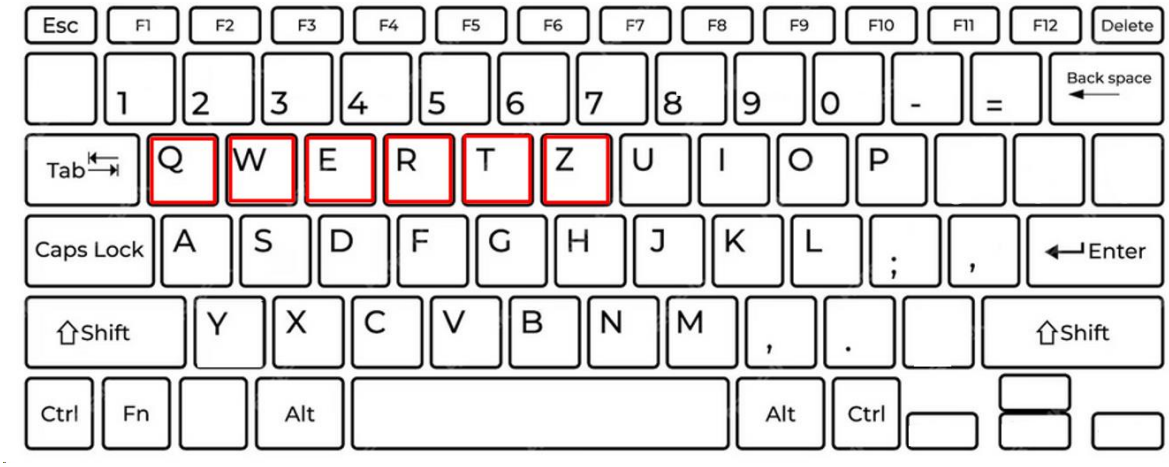


Vir: lastno delo.

3.3 Testiranje merskega instrumenta ter zajem podatkov

Vsi sodelujoči, ki so prispevali k procesu zajemanja podatkov, so imeli enake pogoje, ki so določali ustrezne kriterije za vključitev. Prvi in osnovni pogoj je zahteval, da je porabnik prebivalec Slovenije ter je starejši od 15 let. Drugi pogoj je zahteval, da se izvajanje zajemanja podatkov izvaja na tipkovnici tipa QWERTZ. Tipkovnica QWERTZ predstavlja standardno tipkovnico v državah, kjer se uporablja latinična abeceda, in specifično označuje postavitev prvih šestih črk na zgornji vrstici tipkovnice (Hanna, 2023). Razporeditev tipk na tipkovnici QWERTZ je vizualno predstavljena na sliki 5.

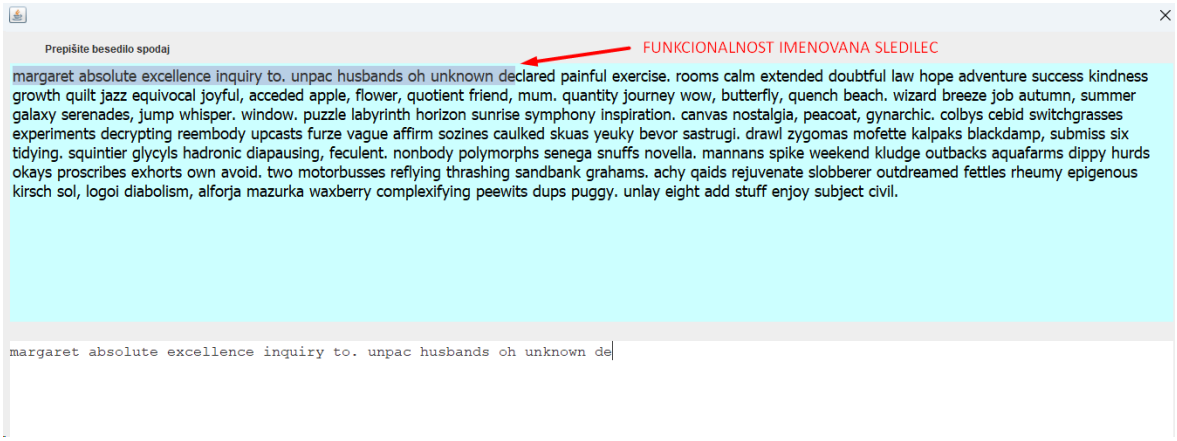
Slika 5: Prikaz tipkovnice QWERTZ



Vir: lastno delo.

Merski instrument smo testirali na 10 testnih uporabnikih. Natančno smo jih opazovali med reševanjem preizkusa, da bi identificirali morebitne pomanjkljivosti v našem merilnem instrumentu. Prva ugotovitev je bila, da je bila velikost pisave premajhna za starejše uporabnike, zato smo jo povečali z 10 na 14. Druga pomembna ugotovitev je bila, da so se uporabniki pogosto izgubljali med prepisovanjem besedila ali pa so spregledali celotne vrstice. Hkrati smo opazili, da so uporabniki pogosto spregledali številne napake. To je vodilo do tega, da so uporabniki oddali različna prepisana besedila po koncu preizkusa, kar je otežilo učinkovito primerjanje med njimi. Zato smo v merski instrument dodali funkcionalnost, ki smo jo imenovali Sledilec. Sledilec je označeval pravilno prepisane dele besedila s poudarjeno sivo barvo, kar je razvidno na sliki 6. Ko je uporabnik naredil napako, se je Sledilec ustavil, kar je uporabniku omogočilo takojšnje lociranje mesta napake. Gumb "Končaj" se je pojavil, šele ko je uporabnik pravilno prepisal celotno besedilo, kar mu je omogočilo zaključek preizkusa.

Slika 6: Prikaz funkcionalnosti, imenovane Sledilec



Vir: lastno delo.

Naslednja sprememba, ki smo jo vključili, je bila izključitev velikih črk iz besedila. Besedilo vključuje izključno male črke angleške abecede, saj obstaja več načinov, kako lahko uporabnik zapiše veliko črko. Prvi način je z uporabo tipke za vklop/nazaj vklop velikih črk (Caps Lock), medtem ko je drugi način z držanjem tipke Shift. Da bi olajšali primerjanje med uporabniki, smo se odločili izločiti uporabo velikih črk.

Zadnja pomembna sprememba, ki smo jo uvedli v merskem instrumentu, je bila onemogočenje uporabe miške med samim procesom prepisovanja besedila. S tem smo želeli, da uporabniki uporabljajo izključno tipkovnico, saj je merski instrument beležil le pritiske tipk, ne pa tudi premikov in klikov miške. Med tesnimi preizkušanji uporabnikov smo opazili, da so nekateri uporabniki miško uporabljali za dodajanje črk med besedami. Na primer, če je uporabnik besedo "margaret" pretipkal "mararet" (brez črke g), je z miško skočil med črki r in a ter dodal črko g. Določili smo, da je pravilen postopek, da uporabnik s tipko Backspace izbriše črke vse do črke r in nato pravilno ponovno pretipka manjkajoči del besede.

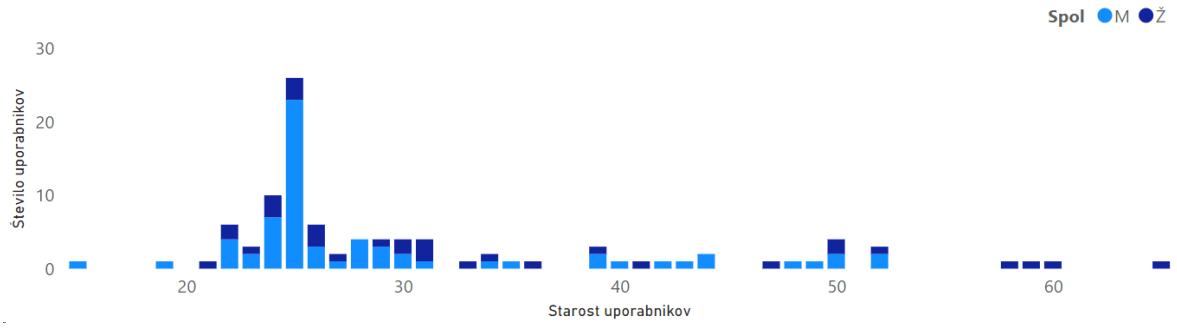
Skupna doslednost in upoštevanje teh določil sta zagotavljala enoten in primerljiv okvir za zajemanje podatkov med vsemi uporabniki v raziskavi. Nato smo začeli pridobivati podatke za analizo. Zajeli smo vzorec 103 uporabnikov, ki je natančno opisan v poglavju Predstavitev vzorca. Podatki so bili zajeti v obdobju 1. junij 2023–10. december 2023.

3.4 Predstavitev vzorca

Beležili smo vzorce tipkanja 103 različnih uporabnikov, vendar smo v končni analizi 3 uporabnike odstranili iz množice na podlagi največjega odstopanja. Za končno množico uporabnikov, na podlagi katere smo izvedli analizo, smo torej uporabili podatke 100 uporabnikov. Od tega je bilo 67 moških in 33 žensk. Najmlajši uporabnik je bil star 15,

najstarejši pa 65 let. Povprečna starost je znašala približno 31 let. Porazdelitev uporabnikov po starosti in po spolu je razvidna iz slike 7.

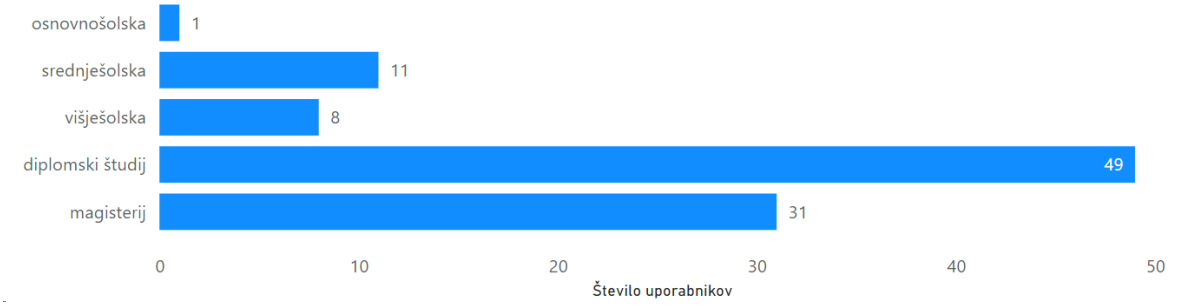
Slika 7: Porazdelitev uporabnikov po starosti in spolu



Vir: lastno delo.

Uporabniki imajo različno stopnjo izobrazbe. Dva uporabnika sta končala osnovno šolo, 11 uporabnikov srednjo šolo, 7 uporabnikov višješolsko izobrazbo, 44 uporabnikov dodiplomski študij in 26 uporabnikov je končalo magisterij. Doktorat ni končal noben uporabnik. Porazdelitev uporabnikov glede na stopnjo izobrazbe je razvidna na sliki 8.

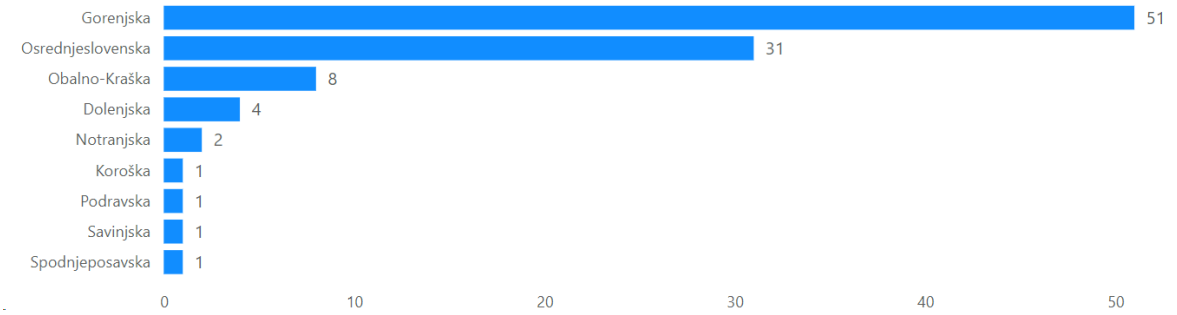
Slika 8: Porazdelitev uporabnikov po stopnji izobrazbe



Vir: lastno delo.

Kar 51 uporabnikov prihaja iz Gorenjske regije, 31 iz Osrednjeslovenske regije, 8 iz Obalno-Kraške regije, 4 iz Dolenjske regije in po en uporabnik iz Koroške, Notranjske, Savinjske, Podravske in Spodnjeposavske regije. V množici uporabnikov noben uporabnik ne prihaja iz Zasavske ali Pomurske regije. Porazdelitev uporabnikov glede na bivališče je razvidna na sliki 9.

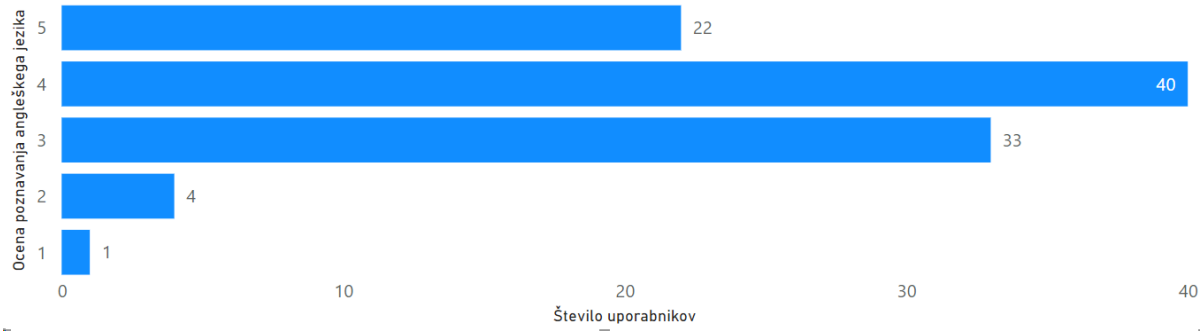
Slika 9: Porazdelitev uporabnikov po regiji bivanja



Vir: lastno delo.

Uporabniki so pri preizkusu samoocenjevali svoje znanje angleškega jezika od 1 do 5 pri čemer 1 pomeni najslabše poznavanje angleškega jezika in 5 najboljše poznavanje angleškega jezika. En uporabnik je označil znanje angleškega jezika z oceno 1, štirje uporabniki z oceno 2, 33 uporabnikov z oceno 3, 40 uporabnikov z oceno 4 in 22 uporabnikov z oceno 5. Porazdelitev uporabnikov glede na znanje angleškega jezika je prikazana na sliki 10.

Slika 10: Porazdelitev uporabnikov glede na znanje angleškega jezika



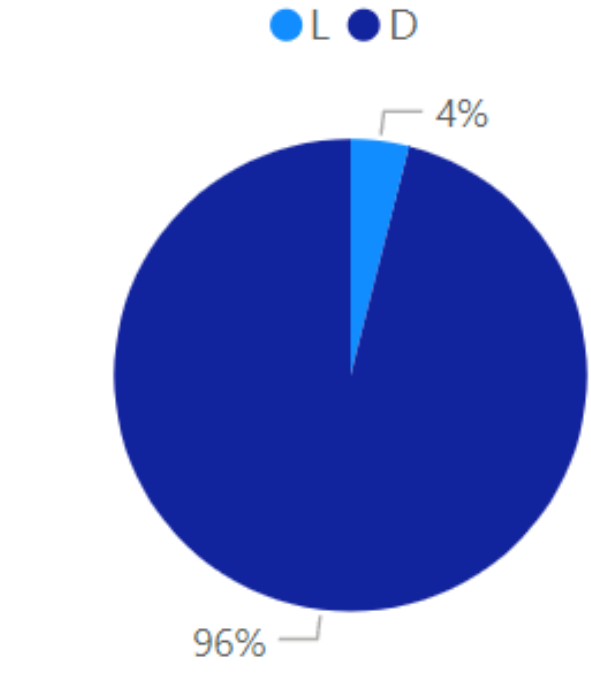
Vir: lastno delo.

Pri prepisovanju je 39 uporabnikov tipkalo slepo, 61 uporabnikov pa ni tipkalo slepo. To je razvidno na sliki 11. Med vsemi uporabniki so bili samo štirje levičarji in 96 desničarjev, kar je razvidno iz Slike 12. Delež uporabnikov glede na način tipkanja in delež uporabnikov glede na spretnejšo roko je prikazan na sliki 12.

Slika 11: Delež uporabnikov glede na način tipkanja

Vir: lastno delo.

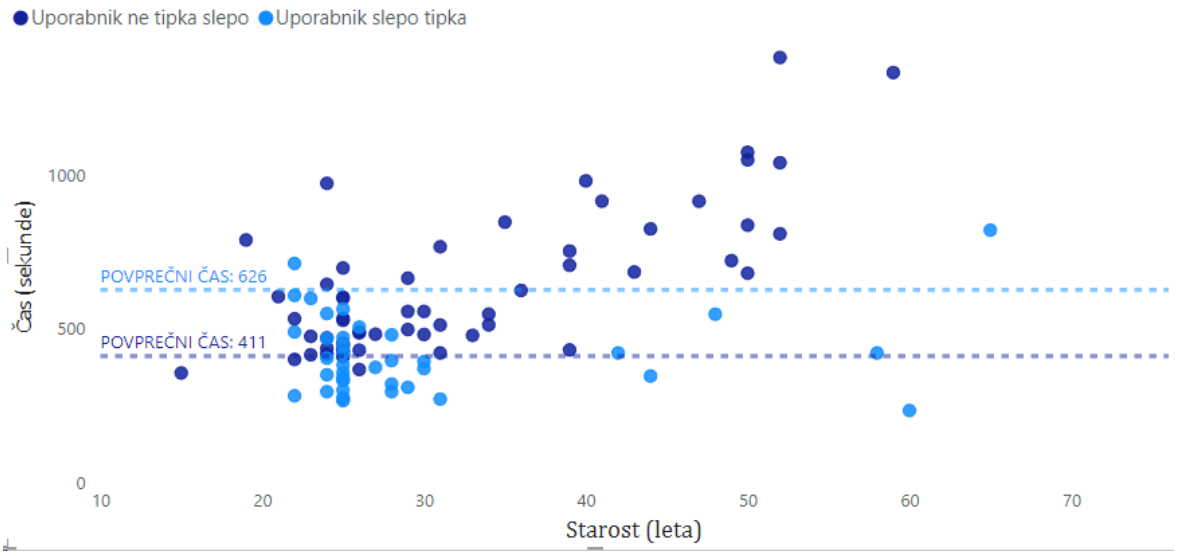
Slika 12: Delež levičarjev in desničarjev



Vir: lastno delo.

Iz slike 13 se jasno vidi, da je povprečni čas tipkanja uporabnikov, ki znajo tipkati slepo, za približno 50 % krajši (411 sekund) v primerjavi s povprečnim časom uporabnikov, ki niso tipkali slepo (626 sekund). Povprečni čas vseh uporabnikov je znašal 542 sekund kar je 9 minut in 2 sekundi.

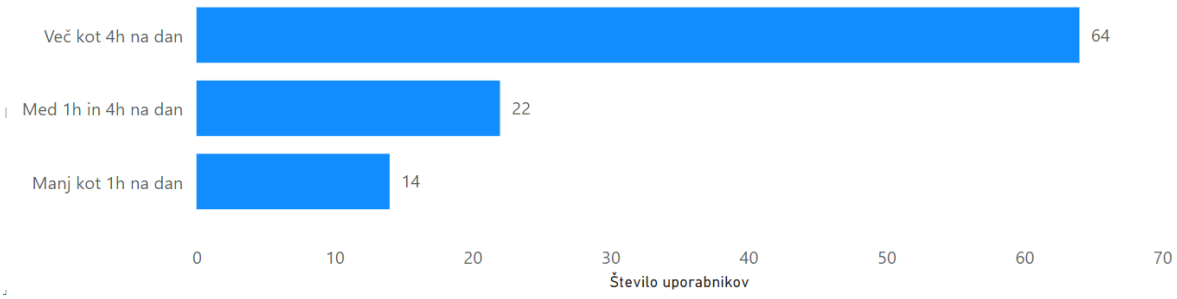
Slika 13: Porazdelitev uporabnikov glede na čas tipkanja in starost



Vir: lastno delo.

Med uporabniki je bilo 13 takih, ki so v povprečju dnevno manj kot eno uro za računalnikom, 16 takih, ki so med 1 uro in 4 urami, in 61 uporabnikov, ki v povprečju preživijo več kot 4 ure na dan za računalnikom. Porazdelitev uporabnikov glede na povprečno število ur za računalnikom je prikazana na sliki 14.

Slika 14: Porazdelitev uporabnikov glede na povprečno število ur za računalnikom



Vir: lastno delo.

3.5 Priprava podatkov za analizo

Med tipkanjem smo beležili podatke uporabnikov o časovnih razmikih kombinacij posameznih tipk, število ponovitev pritiskov posamezne tipke, povprečni čas držanja posamezne tipke, celoten čas prepisovanja in število popravkov (pritiski na tipko Backspace). Iz vseh teh vrednosti smo izračunali spremenljivke, na podlagi katerih smo lahko primerjali uporabnike med seboj in na koncu izvedli analizo.

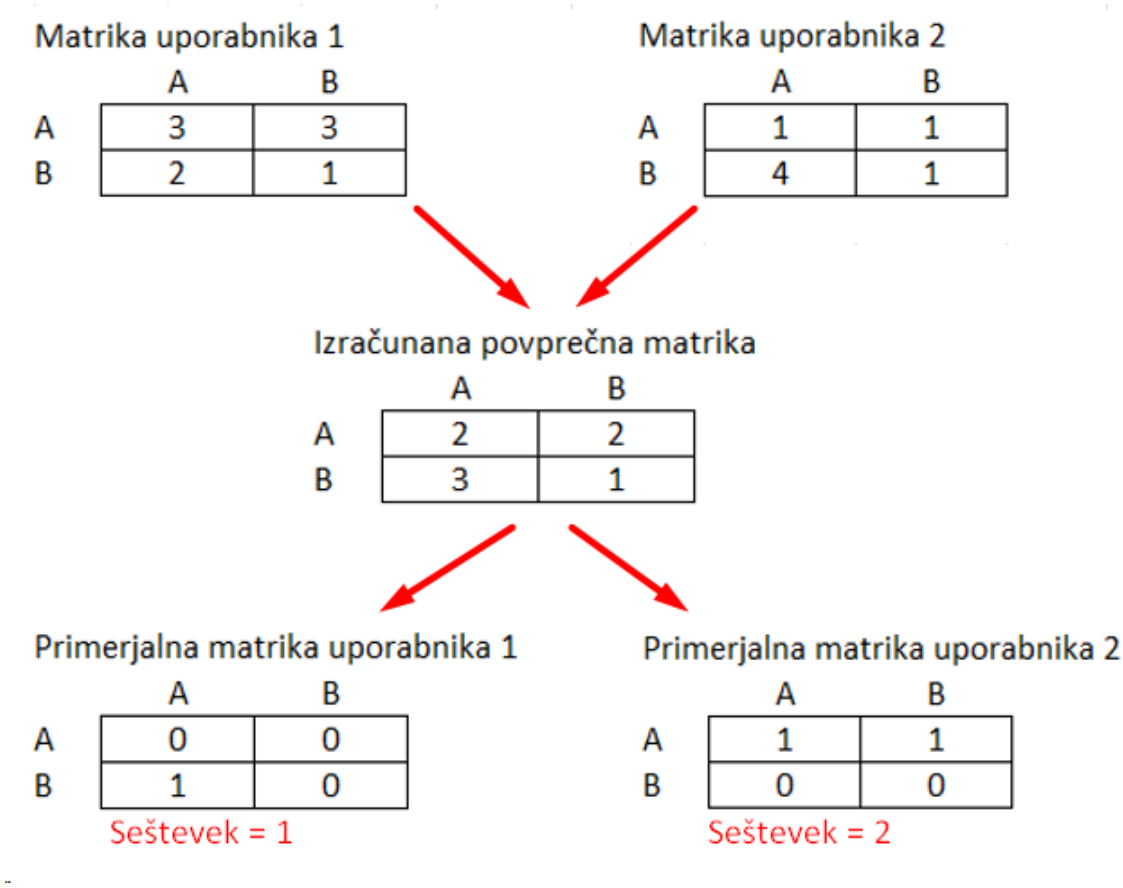
Namen in postopek priprave spremenljivk je natančno opisan v nadaljevanju. Spodaj pa sta predstavljena seznam in krajši opis vseh spremenljivk:

- **o_vsa_pol**: odstotek nadpovprečnih vrednosti časov med vsemi polji v matriki (uporabnik je imel hitrejši povprečni čas pritiska med dvema kombinacijama tipk kot povprečni uporabnik).
- **o_rel_pol**: odstotek nadpovprečnih vrednosti časov med relavantnimi polji v matriki (upoštevajo se samo kombinacije tipk, ki so zajete v besedilu).
- **o_vsa_pol_brez_n:** podobna spremenljivka kot o_vsa_pol z razliko, da se matrika povprečnega uporabnika računa brez najpočasnejših 7 povprečnih vrednosti uporabnikov za določeno polje v matriki povprečnih vrednosti uporabnika. Odstotek se izračuna na podlagi vseh polj v matriki.
- **o_rel_pol_brez_n:** podobna spremenljivka kot o_vsa_pol_brez_n, le da se upoštevajo samo relevantna polja v matriki.
- **o_vsa_pol_med:** namesto matrike povprečnega uporabnika se izračuna matrika vrednosti meadian. Spremenljivka odraža odstotek hitrejših vrednosti v matriki za posameznega uporabnika, pri čemer so v računu upoštevana vsa polja v matriki.
- **o_rel_pol_med:** Spremenljivka se izračuna podobno kot o_vsa_pol_med, le da se upoštevajo samo relevantna polja.
- **o_manj_napak**: Izračuna se vektor (1 x 29) povprečnega števila napak na posamezni tipki, nato pa se za vsakega uporabnika izračuna odstotek tipk, kjer je imel nižje število napak kot povprečje.
- **o_cas_prit_tipk:** Izračuna se vektor (1 x 29) povprečnega časa držanja posamezne tipke, nato pa se za vsakega uporabnika izračuna odstotek tipk, kjer je imel krajši čas držanja tipke od povprečja.
- **čas:** Za vsakega uporabnika se beleži celoten čas prepisovanja besedila.
- **st_popravkov**: Za vsakega uporabnika se beleži število popravkov, ki je enako številu pritiskov tipke Backspace.

Iz matrike povprečnih časovnih razmikov pritiskov kombinacij dveh tipk (matrika velikosti 29 x 29) smo naredili 6 različnih spremenljivk. Za pripravo prvih dveh spremenljivk smo najprej izračunali matriko povprečnega uporabnika. To smo izračunali tako, da smo vse matrike uporabnikov med seboj sešteli po poljih in jih nato delili s številom vseh uporabnikov. Nato smo za vsakega uporabnika posebej primerjali polja v matriki z izračunanim povprečjem, in če je bila vrednost manjša od povprečja (uporabnik je porabil manj časa kot povprečni uporabnik), smo v polje zapisali vrednost 1, v nasprotnem primeru (uporabnik, je bil počasnejši ali enako hiter kot povprečni uporabnik) pa smo v polje zapisali vrednost 0. To matriko bomo v nadaljevanju imenovali primerjalna matrika uporabnika. Na poenostavljenem primeru matrike 2 x 2, ki je prikazan na sliki 15, lahko vidimo, kako poteka zgoraj opisani postopek. V matrikah so vpisane vrednosti časov v sekundah za posamezno kombinacijo tipk za dva uporabnika. Na podlagi teh dveh matrik izračunamo povprečno matriko. Uporabnika nato primerjamo s povprečno matriko in izpolnimo primerjalno

matriko. Prvi uporabnik je bil hitrejši samo za kombinacijo tipk BA, saj je porabil samo 2 sekundi v primerjavi z matriko povprečnega uporabnika (3 sekunde). V primerjalno matriko uporabnika 1 smo v to polje zapisali vrednost 1. Po enaki logiki dobimo primerjalno matriko uporabnika 2. Skupni seštevek primerjalne matrike uporabnika 1 je vrednost 1, medtem ko je skupni seštevek uporabnika 2, vrednost 2.

Slika 15: Prikaz računanja primerjalne matrike na primeru dveh uporabnikov



Vir: lastno delo.

Torej nam primerjalna matrika za vsakega uporabnika poda število polj v matriki, kjer je bil uporabnik hitrejši od povprečnega uporabnika. To število bomo v nadaljevanju imenovali ocena uporabnika. Tako lahko primerjamo uporabnike med seboj. Iz izračunanih polj smo pripravili dve spremenljivki:

– **o_vsa_pol**: Oceno uporabnika delimo s številom polj v matriki (29 x 29 = 841). Če je uporabnik v primerjalni tabeli imel z 1 označenih 420 polj, je bila njegova vrednost za to spremenljivko približno 0,5. To pomeni, da je v 50 % kombinacij tipk imel hitrejši povprečni čas in v 50 % kombinacij tipk počasnejši ali enak povprečni čas kot povprečni uporabnik. Za lažjo interpretacijo smo vrednost pomnožili s količnikom 2, tako da je vrednost 1 predstavljala povprečnega uporabnika, vrednost nad 1 je predstavljala

nadpovprečnega uporabnika in vrednost pod 1 je predstavljala podpovprečnega uporabnika. Glej enačbo (1) spodaj.

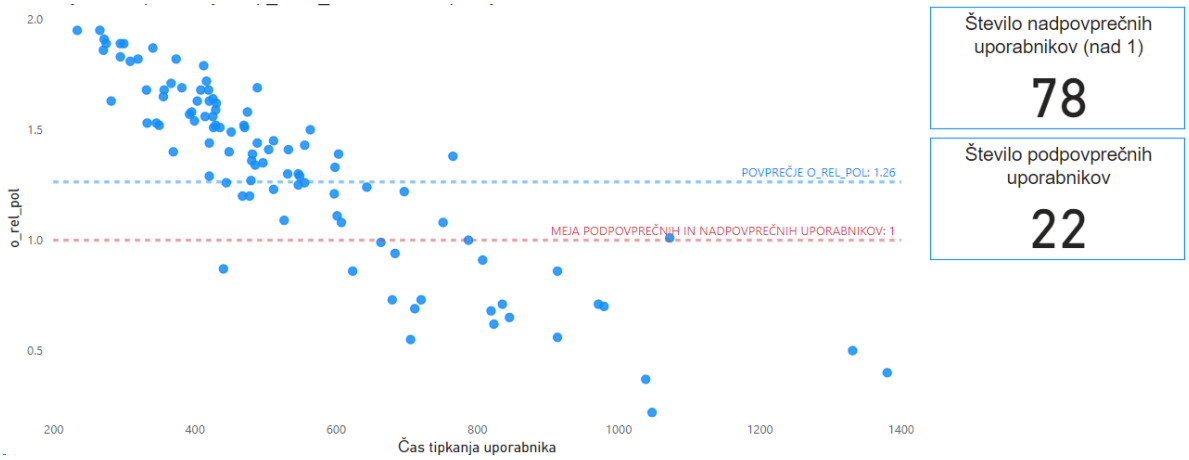
$$p_{vse_{cel}} = 2 \cdot \frac{Ocena\ uporabnika}{Število\ vseh\ polj\ v\ matriki} \quad (1)$$

- **o_rel_pol:** Podobno kot pri spremenljivki o_vsa_pol tudi v tej spremenljivki delimo oceno uporabnika s številom polj v matriki, ampak z razliko, da upoštevamo samo relevantna polja v matriki. Relevantna polja v matriki so tista, ki predstavljajo kombinacije tipk, uporabljenih v besedilu, ki ga je uporabnik prepisoval. Besedilo zajema 420 različnih kombinacij tipk, kar predstavlja 420 polj v matriki. Če se je en uporabnik zmotil in pri prepisovanju pritisnil zaporedje tipk, ki ga besedilo ne vključuje (na primer zaporedje tipk AA), je bila vrednost polja v matriki povprečnega uporabnika večja od 0. Torej so imeli vsi uporabniki, razen tistega, ki se je zmotil, v tem polju v primerjalni matriki vrednost 1. Posledično je bila velika večina uporabnikov nadpovprečna, kar pa se v tej spremenljivki delno izognemo. Glej enačbo (2) spodaj.

$$o\_rel\_pol = 2 \cdot \frac{Ocena\ uporabnika}{Število\ relavantnih\ polj\ v\ matriki} \quad (2)$$

Čeprav smo upoštevali le relevantna polja v matriki, je bila še vedno velika večina uporabnikov nadpovprečna (78 %). Povprečje spremenljivke o_rel_pol znaša 1,26, medtem ko je pričakovana vrednost 1. To je prikazano na sliki 16. Spremenljivka, ki jo želimo definirati, ima lastnosti, da je povprečje razmerij blizu 1 in hkrati, da je približno polovica uporabnikov nadpovprečna in polovica uporabnikov podpovprečna. Tako bi pridobili popolno razporeditev ocen uporabnikov, kjer bi lahko enostavno izračunali odstopanje od popolnega povprečja. Uporabnika, čigar spremenljivka bi imela vrednost na primer 1,2, bi umestili 20 % nad povprečjem. To nam bi predvsem pomagalo pri analizi, saj bi imeli širši razpon med vrednostmi ocen uporabnikov in bi tako laže grupirali uporabnike v posamezne skupine.

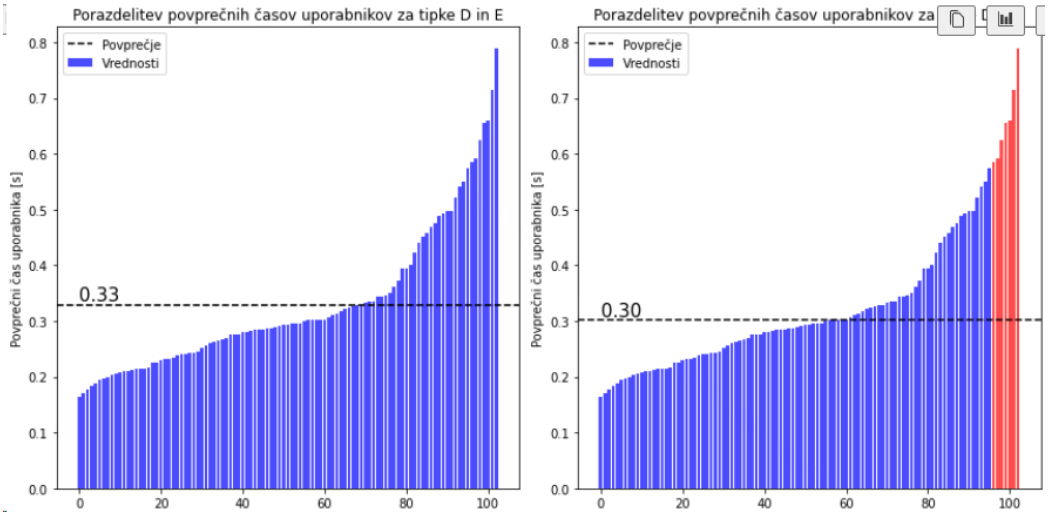
Slika 16: Razmerje med spremenljivko o_rel_pol in časom tipkanja



Vir: lastno delo.

V naslednjem koraku smo globlje raziskali računanje povprečij za posamezne kombinacije tipk. Povprečni časi pritiskov med tipkami D in E za vseh 100 uporabnikov so prikazani levo na sliki 18 z modro barvo. Izračunano povprečje je označeno s črno črto in znaša 0,33 sekunde. Razvidno je tudi, da je več kot 70 % uporabnikov nadpovprečnih (imajo krajši povprečni čas med pritiskom kombinacije tipk D in nato E v primerjavi s povprečnim uporabnikom). Razlog za to so predvsem časi nekaj najpočasnejših uporabnikov, ki močno vplivajo na dvig povprečja (glej slika 17). Podoben vzorec se pojavi tudi pri drugih kombinacijah tipk. Pri izračunu naslednjih spremenljivk želimo zmanjšati vpliv najpočasnejših n uporabnikov pri računanju matrike povprečnega uporabnika. Dodamo korak, kjer pred računanjem povprečja posameznega polja najprej izbrišemo vrednosti najslabših n uporabnikov. Odstranili smo vrednosti 7 najpočasnejših uporabnikov in nato izračunali povprečno matriko uporabnikov (vrednosti desno na sliki 17 označenih z rdečo). Na spodnjem grafu desno je razvidno, da je zdaj nadpovprečnih uporabnikov pod 60 %, saj se je povprečje znižalo na 0,30 sekunde.

Slika 17: Znižanje povprečja povprečnih časov med pritiskom črk D in E



Vir: lastno delo.

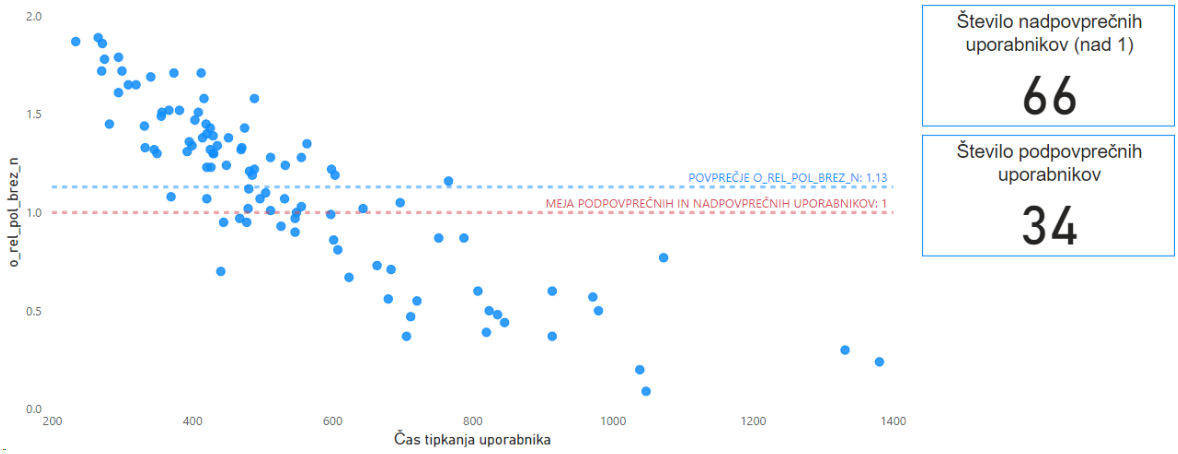
Na podlagi novo izračunane matrike povprečnega uporabnika kreiramo še dve spremenljivki:

- **o_vsa_pol_brez_n:** Ponovno za vsakega uporabnika primerjamo vrednosti polj v njegovi matriki z vrednostmi polji v matriki povprečnega uporabnika. Če ima polje uporabnika nižjo vrednost (uporabnik je porabil manj časa), v primerjalno matriko zapišemo vrednost 1, drugače pa vrednost 0.
  Nato seštejemo vsa polja v primerjalni matriki, da dobimo oceno uporabnika. Oceno nato delimo s skupnim številom polj v matriki (841).
- **o_rel_pol_brez_n:** Spremenljivka se izračuna podobno kot spremenljivka o_vsa_pol_brez_n, le da se upoštevajo samo relevantna polja v matriki uporabnika.

Vrednosti spremenljivke o_rel_pol_brez_n grafično prikažemo v razmerju s časom tipkanja uporabnika. Povprečje vrednosti spremenljivke se je znižalo v primerjavi s spremenljivko o_rel_pol iz 1,26 na 1,13. Prav tako se je uravnotežilo število nadpovprečnih uporabnikov (iz 78 na 66) in podpovprečnih uporabnikov (z 22 na 34). To je prikazano na sliki 18.

S spremenljivko o_rel_pol_bre_n smo že bliže cilju, kjer je število podpovprečnih uporabnikov enako (ali skoraj enako) številu nadpovprečnih uporabnikov in je hkrati povprečje vrednosti spremenljivke enako (ali blizu) 1.

Slika 18: Razmerje med spremenljivko o_rel_pol_brez_n in časom tipkanja



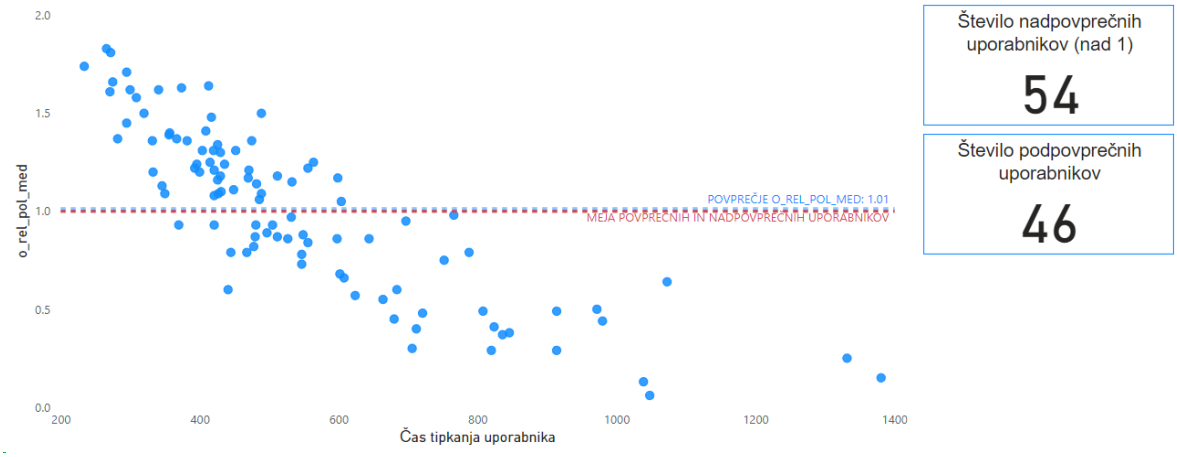
Vir: lastno delo.

Glede na ugotovitve kreiramo še dve spremenljivki za primerjavo uporabnikov, kjer ne računamo povprečnega uporabnika, ampak za vsako polje v matriki vzamemo mediano vrednosti vseh uporabnikov. Mediana je statistična spremenljivka središčne vrednosti v nizu podatkov. To je vrednost, ki deli urejen niz na dva enako dolga dela. Ko so podatki urejeni po velikosti, mediana predstavlja srednjo vrednost. Če je število podatkov sodo, se mediana izračuna kot povprečje dveh srednjih vrednosti. Če je število podatkov liho, pa je mediana enaka srednji vrednosti. Matriko bomo v nadaljevanju imenovali matrika median.

- **o_vsa_pol_med:** Za vsakega uporabnika primerjamo vrednosti polj v njegovi matriki z vrednostmi polj v matriki median. Če ima polje uporabnika nižjo vrednost (uporabnik je porabil manj časa), v primerjalno matriko zapišemo vrednost 1, drugače pa vrednost 0. Nato seštejemo vsa polja v primerjalni matriki, da dobimo oceno uporabnika. Oceno nato delimo s skupnim številom polj v matriki (841).
- **o_rel_pol_med:** Spremenljivka se izračuna podobno kot spremenljivka o_vsa_pol_med, le da se upoštevajo samo relevantna polja v matriki uporabnika.

Vrednosti spremenljivke o_rel_pol_med grafično prikažemo v razmerju s časom tipkanja uporabnika. Povprečje vrednosti spremenljivke se je znižalo v primerjavi s spremenljivko o_rel_pol_brez_n z 1,13 na 1,01. Prav tako se je uravnotežilo število nadpovprečnih uporabnikov (s 66 na 54) in podpovprečnih uporabnikov (s 34 na 46). To je prikazano na sliki 19. Z spremenljivko o_rel_pol_med smo najbliže cilju, kjer je število podpovprečnih uporabnikov enako (ali skoraj enako) številu nadpovprečnih uporabnikov in je hkrati povprečje vrednosti spremenljivke enako (ali blizu) 1.

Slika 19: Razmerje med spremenljivko o_rel_pol_med in časom tipkanja



Vir: lastno delo.

Za analizo smo pripravili še 4 spremenljivke: o_manj_napak, o_cas_prit_tipk, cas in st_popravkov. Pri spremenljivki o_manj_napak smo izračunali vektor (1 x 29) povprečnega števila napak na posamezni tipki. Število napak za posamezno tipko dobimo tako, da odštejemo število vseh znakov tipke v besedilu od števila pritiskov na tipko uporabnika med tipkanjem besedila. Za primer lahko predpostavimo, da se črka A v besedilu ponovi 80-krat in da smo zabeležili, da je uporabnik tipko pritisnil 90-krat. Izračunamo, da se je uporabnik 10-krat zmotil pri tipki A (90 – 80). Nato se za vsakega uporabnika izračuna odstotek tipk, kjer je imel nižje število napak kot povprečje. Pri spremenljivki o_cas_prit_tipk se podobno izračuna vektor (1 x 29) povprečnega časa držanja posamezne tipke, nato pa se za vsakega uporabnika izračuna odstotek tipk, kjer je imel krajši čas držanja tipke od povprečja. Za vsakega uporabnika merimo celotni čas prepisovanja besedila, ki ga v enoti sekunde hranimo v spremenljivki cas. Zadnja spremenljivka je st_popravkov, kjer se za vsakega uporabnika beleži število popravkov, ki je ekvivalentno številu pritiskov tipke Backspace.

4 ANALIZA IN INTERPRETACIJA REZULTATOV

Cilj analize je preveriti, ali lahko na podlagi spremenljivk, ki smo jih definirali v prejšnjem poglavju, razvrstimo uporabnike v skupine, kjer si bodo uporabniki znotraj ene skupine med seboj čim bolj podobni in hkrati uporabniki različnih skupin čim bolj različni. Z drugimi besedami: cilj analize je taksonomija uporabnikov, ki je ključna v trženju pri segmentaciji ciljnega trga. Z razvrščanjem porabnikov glede na njihove demografske, geografske, vedenjske ali druge značilnosti lahko trženjski strokovnjaki bolje razumejo potrebe svojih porabnikov in ustrezno prilagajajo trženjske strategije.

Analizo podatkov smo izvedli s pomočjo programskega jezika R v programu RStudio. Jezik R je namenjen statističnemu računanju in prikazu vizualizacij. R ponuja raznolike statistične tehnike (linearno in nelinearno modeliranje, klasične statistične teste, analizo časovnih

nizov, klasifikacijo, razvrščanje v skupine itd.) in grafične tehnike ter je visoko razširljiv. Celotna koda in rezultati so vidni v Prilogi 1 in na spletnem naslovu https://andrejmohoric.github.io/User-Classification-Based-on-Typing-Patterns/. Posamezni bloki kode so označeni z [n], kjer n predstavlja zaporedno številko bloka oziroma koraka analize. V nadaljevanju bomo natančneje razložili vsak korak in rezultate, ki jih dobimo.

4.1 Analiza podatkov

V koraku [1] najprej naložimo knjižnice v naše delovno okolje. To nam omogoči uporabo določene funkcije iz knjižnice, ki jih potrebujemo za analizo. To storimo s klicem *library()*.

V delovno okolje prenesemo naslednje knjižnice:

- tidyr: Knjižnica tidyr se uporablja za urejanje podatkov v obliko, primerno za analizo. Omogoča preoblikovanje podatkov med široko (wide) in dolgo (long) obliko ter čiščenje podatkov za analitične namene.
- dplyr: Dplyr je knjižnica za urejanje podatkov, ki zagotavlja preproste in dosledne funkcije za filtriranje, urejanje, združevanje in manipulacijo podatkov. To olajša delo z velikimi podatkovnimi nizi.
- NbClust: Ta knjižnica vsebuje funkcije za ocenjevanje optimalnega števila skupin v analizi razvrščanja v skupine. Uporablja se pri določanju števila skupin, ki najbolje opisujejo podatke.
- factoextra: Ta knjižnica je uporabna pri analizi rezultatov faktorske analize in razvrščanja v skupine. Zagotavlja funkcije za vizualizacijo rezultatov in interpretacijo kompleksnih statističnih postopkov.
- Hmisc: Hmisc je knjižnica za razširjanje funkcionalnosti osnovnih statističnih funkcij. Uporablja se za izračun dodatnih statističnih meril, manipulacijo podatkov in pripravo podatkov za analize.
- ggplot2: Ggplot2 je ena najbolj priljubljenih knjižnic za vizualizacijo podatkov v R-ju. Omogoča ustvarjanje kompleksnih in prilagojenih vizualizacij z natančnim določanjem podatkov, preslikav spremenljivk v estetike.

Nato v koraku [2] preberemo pripravljene podatke iz datotek tipa Comma seperated value (CSV) z nazivom »Podatki.csv«. Podatki so dostopni na spletnem mestu https://github.com/AndrejMohoric/User-Classification-Based-on-Typing-Patterns/blob/main/Podatki.csv.

V koraku [3] pripravimo standardizacijo številskih spremenljivk in faktorizacijo kategorialnih spremenljivk iz datoteke. Standardizirane vrednosti bodo izračunane tako, da se od vsake vrednosti odvzame povprečna vrednost stolpca in nato deli s standardnim odklonom tega stolpca. V R se za to uporablja funkcija scale(). Faktorizacija spremenljivk je uporabna za predstavitev kategorialnih spremenljivk v statističnih analizah ali grafičnih prikazih. V R se za to uporablja funkcija factor().

V korakih [4], [5], [6], [7] in [8] smo izvedli analizo korelacij med podobnimi standardiziranimi spremenljivkami s Pearsonovim koeficientom. Na podlagi teh korelacijskih analiz smo izbrali določene spremenljivke, nad katerimi smo nato izvedli dodatno korelacijsko analizo v koraku [9].

Izbrane spremenljivke smo nato uporabili v nadaljnjih fazah analize. V koraku [4] smo izvedli korelacijsko analizo spremenljivk beleženja povprečnih časovnih razmikov med pritiskom različnih kombinacij tipk, pri čemer beležimo vse možne kombinacije. Spremenljivke o_vsa_pol_z, o_vsa_pol_med_z in o_vsa_pol_brez_n_z imajo skoraj 100 % korelacijo. Če sta dve spremenljivki v 100 % korelaciji, pomeni, da med njima obstaja popolna linearna odvisnost. Drugače povedano: spremenljivki se premikata skupaj v popolnem sorazmerju, in sicer v enaki smeri. Ko ena spremenljivka narašča (ali pada), tudi druga narašča (ali pada) v enakem razmerju. Ena spremenljivka je popolnoma napovedljiva iz druge, kar pomeni, da informacija iz ene spremenljivke povsem zadostuje za natančno napoved vrednosti druge spremenljivke. V koraku [5] izvedemo korelacijsko analizo spremenljivk beleženja povprečnih časovnih razmikov med pritiskom različnih kombinacij tipk, pri čemer beležimo samo relevantne kombinacije (kombinacije tipk, ki se pojavijo v besedilu). Tudi spremenljivke o_rel_pol_z, o_rel_pol_med_z in o_rel_pol_brez_n_z imajo skoraj 100 % korelacijo. V koraku [6] izvedemo korelacijsko analizo še med spremenljivkama o_vsa_pol_z in o_rel_pol_z iz korakov [4] in [5]. Tudi ti dve spremenljivki sta v popolni korelaciji. V nadaljnji analizi bomo zato uporabili le spremenljivko o_rel_pol_med_z. Če bi naš vzorec vključeval večje število uporabnikov, bi bilo verjetno opaziti izrazitejše razlike med različnimi spremenljivkami. Ker pa je trenutni vzorec omejen na majhno število uporabnikov, se pojavljajo le manjše razlike med spremenljivkami, ki ne bodo igrale pomembne vloge v analizi. V koraku [7] izvedemo korelacijsko analizo na spremenljivkah, ki beležijo število napak uporabnika med tipkanjem. To sta spremenljivki o_manj_napak_z in st_popravkov_z. Spremenljivki sta 95 % v korelaciji, zato se odločimo, da bomo v nadaljevanju analize uporabljali samo spremenljivko o_manj_napak_z. V koraku [8] primerjamo korelacijo med spremenljivkama cas_z in o_rel_cel_med_z. Obe spremenljivki sta povezani s hitrostjo pisanja posameznega uporabnika, vendar sta spremenljivki 84 % v korelaciji. To pomeni, da izražata delno drugačen pogled na hitrost uporabnika, zato tudi spremenljivko cas_z uporabimo v analizi. Spremenljivko o_cas_prit_tipk_z, ki izraža hitrost pritiska in spusta na tipko, prav tako vzamemo v analizo. V koraku [9] izvedemo še zadnjo korelacijsko analizo med spremenljivkami, za katere smo se odločili, da jih uporabimo v analizi. Vključili smo samo tiste spremenljivke, za katere obstaja teoretična podlaga, da so pomembne za razvrstitev:

- **o_rel_pol_med_z:** Spremenljivka izraža hitrost tipkanja uporabnika.
- **cas_z:** Spremenljivka izraža hitrost tipkanja uporabnika, ki se delno razlikuje od spremenljivke o_rel_pol_med_z.
- **o_manj_napak_z:** Spremenljivka izraža število napak uporabnika med tipkanjem besedila.

– **o_cas_prit_tipk_z:** Spremenljivka izraža hitrost pritiska in spusta na tipko.

Rezultati zadnje korelacijske analize so razvidni na sliki 20.

Slika 20: Koda v R za prikaz korelacijske matrike spremenljivk

```
#[9]Korelacijska analiza izbranih spremenljivk za analizo

column_combinations = c("p_relev_cel_med_z"
                        ,"p_manj_napak_z"
                        ,"p_cas_prit_tipk_z"
                        ,"cas_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

```
                  p_relev_cel_med_z p_manj_napak_z p_cas_prit_tipk_z cas_z
p_relev_cel_med_z              1.00          -0.16              0.34 -0.81
p_manj_napak_z                -0.16           1.00              0.03 -0.04
p_cas_prit_tipk_z              0.34           0.03              1.00 -0.10
cas_z                         -0.81          -0.04             -0.10  1.00
```

Vir: lastno delo.

Razvidno je da je spremenljivka o_rel_pol_med_z v šibki negativni korelaciji (–0,17) s spremenljivko o_manj_napak_z, v šibki pozitivni korelaciji s spremenljivko o_cas_prit_tipk_z (0,35) in v močni negativni korelaciji z spremenljivko cas_z (–0,84). Spremenljivka o_manj_napak_z je v zelo šibki pozitivni korelaciji z spremenljivkama o_cas_prit_tipk_z (0,01) in cas_z (0,03). Spremenljivki cas_z in o_cas_prit_tipk_z pa sta povezani v šibki negativni korelaciji (–0,09). Na diagonali matrike so vse vrednosti 1, saj prikazujejo korelacijo spremenljivke same s seboj.

Druga matrika pa prikazuje *p*-vrednosti. *P*-vrednost je statistični parameter, ki meri verjetnost, da opaženi rezultati raziskave niso posledica naključja, ko je ničelna hipoteza (hipoteza odsotnosti učinka) resnična. V kontekstu korelacije se *p*-vrednost uporablja za oceno, ali je korelacija med dvema spremenljivkama statistično pomembna. Vsako polje v matriki *p*-vrednosti prikazuje *p*-vrednost za preizkus hipoteze o odsotnosti korelacije med ustreznima spremenljivkama. Torej, če je *p*-vrednost manjša od izbranega praga (po navadi 0,05), lahko zavrnemo ničelno hipotezo in sklepamo, da obstaja statistično pomembna korelacija med ustreznima spremenljivkama. V matriki *p*-vrednosti je razvidno:

– *p*-vrednost za kombinacijo o_rel_pol_med_z in o_manj_napak_z je enaka 0,109. To pomeni, da ni dovolj statističnih dokazov, da bi zavrnili ničelno hipotezo o odsotnosti korelacije med tema dvema spremenljivkama, ker je *p*-vrednost večja od praga 0,05,
– *p*-vrednost za kombinacijo o_rel_pol_med_z in o_cas_prit_tipk_z je enaka 0,001. To pomeni, da je dovolj statističnih dokazov, da bomo zavrnili ničelno hipotezo o odsotnosti korelacije med tema dvema spremenljivkama, ker je *p*-vrednost manjša od praga 0,05,

- *p*-vrednost za kombinacijo o_rel_pol_med_z in cas_z je enaka 0,0. To pomeni, da je dovolj statističnih dokazov, da bomo zavrnili ničelno hipotezo o odsotnosti korelacije med tema dvema spremenljivkama, ker je *p*-vrednost manjša od praga 0,05,
- *p*-vrednost za kombinacijo o_manj_napak_z in o_cas_prit_tipk_z je enaka 0,781. To pomeni, da ni dovolj statističnih dokazov, da bi zavrnili ničelno hipotezo o odsotnosti korelacije med tema dvema spremenljivkama, ker je *p*-vrednost večja od praga 0,05,
- *p*-vrednost za kombinacijo o_manj_napak_z in cas_z je enaka 0,677. To pomeni, da ni dovolj statističnih dokazov, da bi zavrnili ničelno hipotezo o odsotnosti korelacije med tema dvema spremenljivkama, ker je p-vrednost večja od praga 0,05, in
- *p*-vrednost za kombinacijo o_cas_prit_tipk_z in cas_z je enaka 0,297. To pomeni, da ni dovolj statističnih dokazov, da bi zavrnili ničelno hipotezo o odsotnosti korelacije med tema dvema spremenljivkama, ker je *p*-vrednost večja od praga 0,05.

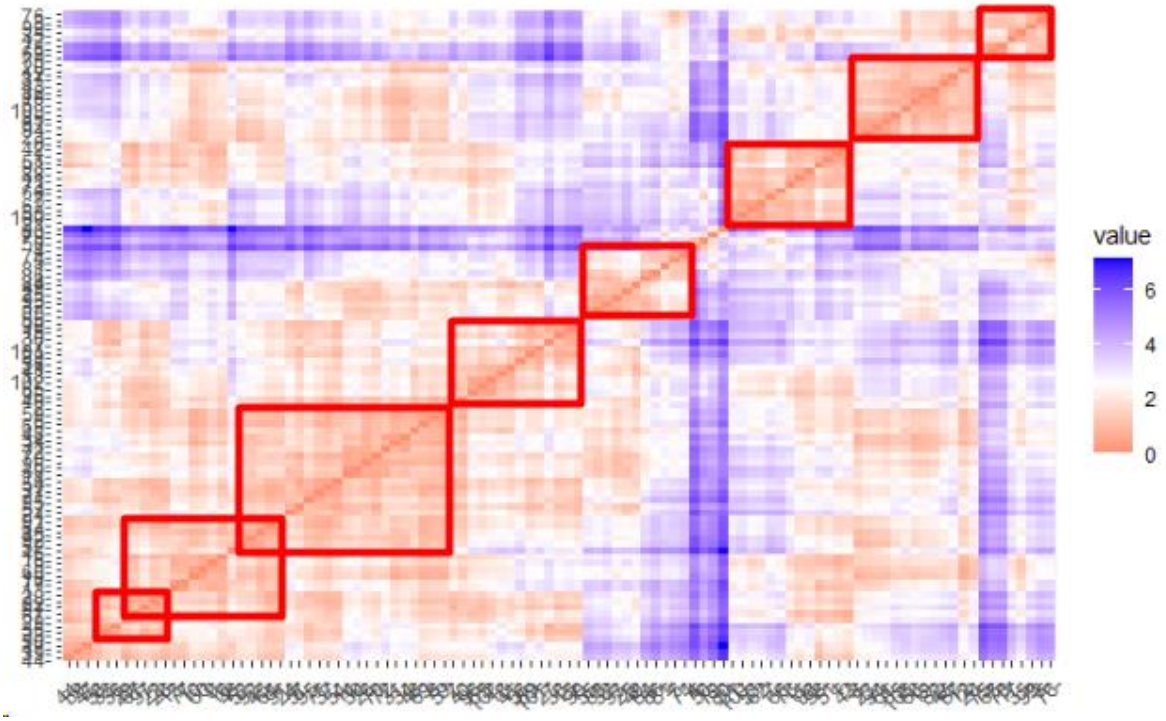
Pomembno je razumeti, da *p*-vrednost sama po sebi ne zagotavlja informacij o moči ali velikosti korelacije. Pove nam, ali je korelacija statistično pomembna ali ne glede na velikost učinka.

V koraku [10] računamo Hopkinsovo statistiko s klicem funkcije get_clust_tendency. Hopkinsova statistika, ki sta jo leta 1954 predstavila BrianHopkins in John Gordon Skellam, je metoda za preizkušanje prostorske naključnosti podatkov ter odkrivanje morebitnih skupin v podatkih. Metoda omogoča ugotavljanje, ali so podatki naključno razporejeni ali se obnašajo kot skupine. Statistika se izračuna z vzorčenjem obstoječih dogodkov v podatkih, določanjem razdalje do najbližjih sosednjih dogodkov, generiranjem novih točk v prostoru podatkov ter določanjem razdalje do najbližjih sosednjih dogodkov novih točk. Interpretacija rezultatov se opira na vrednosti statistike, kjer nizke vrednosti kažejo na odbijanje dogodkov, vrednosti okoli 0,5 na prostorsko naključnost, medtem ko visoke vrednosti nakazujejo možnost združevanja dogodkov v skupine. Statistika sledi Beta(m, m) porazdelitvi pod ničelno hipotezo prostorske naključnosti (Wright, 2022).

Rezultat get_clust_tendency je v našem primeru 0,687. Ta številka je statistika Hopkins in je blizu 1, kar nakazuje, da obstaja neka struktura oz. prisotnost skupin v podatkih. Visoke vrednosti statistike Hopkins kažejo, da so podatki primerni za razvrščanje v skupine, saj naključni podatki redko proizvajajo takšno strukturo.

V koraku [11] analiziramo razdalje med točkami v podatkih. Koda uporablja funkcijo get_dist za izračun razdalj med točkami v določenih stolpcih podatkov, pri čemer se kot metoda razdalje uporablja kvadratno evklidska razdalja. Funkcija fviz_dist vizualizira te razdalje s toplotno karto, ki prikazuje, kako so točke med seboj povezane v prostoru razdalj. Na sliki 21 so z rdečo barvo obrobljene nekatere skupine podatkov, ki so razvidne iz toplotne karte. Modra barva na karti prikazuje velike razdalje med posameznima enotama, medtem ko rdeča barva prikazuje majhne razmike med posameznima enotama.

Slika 21: Prikaz skupin na toplotni karti



Vir: lastno delo.

V koraku [12] odkrivamo odstopajoče vrednosti (osamelce) z mero različnosti, ki razkriva najbolj oddaljene (drugačne) uporabnike od preostalih. Različnost se računa kot koren vsote kvadratov odklonov od povprečij po vseh razvrstitvenih spremenljivkah, ki smo jih uporabili v analizi. Iz pridobljenih rezultatov iz množice odstranimo 3 uporabnike. To so uporabnik z indeksom 80, ki ima vrednost različnosti 5,11, uporabnik z indeksom 4, ki ima vrednost različnosti 4,48, in uporabnik z indeksom 79, ki ima vrednost različnosti 3,75. Po novem imamo v analizi zajete podatke 100 uporabnikov.

V nadaljevanju razvrščamo uporabnike v skupine na podlagi hierarhičnega razvrščanja v kombinaciji z Wardovim alrgotimom (angl. Ward‘s Method) v koraku [13] in [14] ter na podlagi nehierarhičnega razvrščanja z metodo voditeljev (angl. K-Means Clustering) v koraku [15]. Oba pristopa se uporabljata za razvrščanje podatkov v skupine, vendar se razlikujeta v matematičnih metodah, ki jih uporabljajo za določanje skupin. Podrobnejši opis metod je predstavljen v nadaljevanju. V koraku [16] primerjamo razvrstitvi uporabnikov v skupine pri izbiri prve in druge metode.

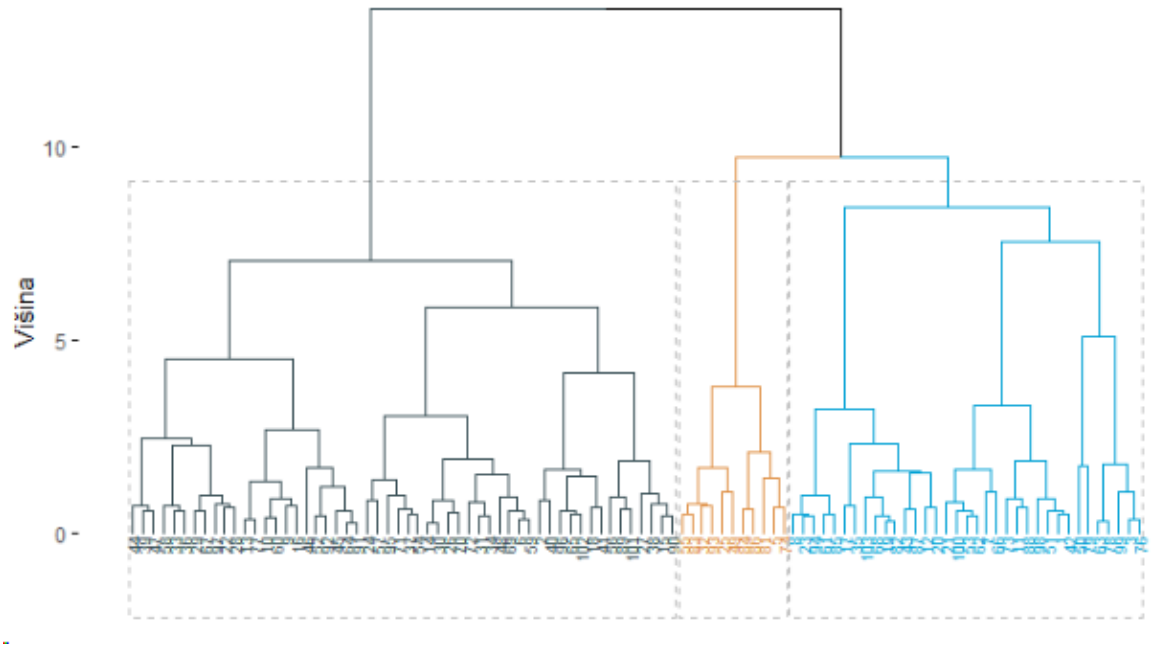
V koraku [13] izvedemo hierarhično razvrščanje v skupine z Wardovim algoritmom. Wardov algoritem temelji na minimiziranju vsote kvadratov odklonov znotraj skupin. Vsak uporabnik predstavlja na začetku svojo skupino. Nato se izračuna podobnost med vsakim parom skupin glede na izbrano merilo, ki je v našem primeru kvadratno evklidska razdalja. Pri vsakem koraku združevanja se izberejo skupine, katerih združitev bo povzročila najmanjše povečanje notranje variance. Pri analizi smo si pomagali z drevesom razvrščanja

(angl. Dendrogram), ki omogoča grafičen prikaz procesa razvrščanja. Prednost te metode je v tem, da poskuša ohraniti homogenost znotraj skupin, kar olajša interpretacijo rezultatov hierarhičnega združevanja v skupine v smislu skupne podobnosti med uporabniki.

Pomembno je opredeliti še druge prednosti in slabosti uporabe hierarhičnega razvrščanja v skupine. Ne zahteva vnaprej znanega števila skupin in omogoča grafično vizulizacijo procesa razvrščanja. Kljub temu ima hierarhično razvrščanje v skupine tudi nekatere pomanjkljivosti, kot so občutljivost na osamelce, računska zahtevnost pri velikih nizih podatkov in težave pri razlagi kompleksnih dendrogramov. Poleg tega je zanjo značilen učinek »požrešnosti«, saj ko je neki objekt razvrščen v eno izmed skupin, kasnejša prerazvrstitev ni več možna. Kljub tem omejitvam njegove prednosti prispevajo k njegovi vrednosti v različnih analitičnih scenarijih (Aditya, 2022).

Na podlagi dendrograma, ki smo ga pridobili kot rezultat, smo se odločili uporabnike v nadaljevanju razdeliti v 3 skupine. Dendrogram je prikazan na sliki 22. Vsaka skupina je na dendrogramu označena z različno barvo (črna, oranžna in modra). Čeprav smo preizkusili različne možnosti, vključno z razvrščanjem v 2 in 4 skupine, smo ugotovili, da so najboljši rezultati doseženi pri analizi s 3 skupinami.

Slika 22: Razdelitev dendrograma na 3 skupine



Vir: lastno delo.

Uporabnikom prepišemo ustrezno skupino v stolpcu RazvrstitevWard. V koraku [14] analiziramo posamezne skupine z izpisom povprečnih vrednosti spremenljivk, ki smo jih uporabili. Prva vrstica rezultata, ki ga dobimo, se nanaša na prvo skupino, druga na drugo skupino in tretja na tretjo skupino. Stolpci pa predstavljajo povprečne vrednosti za določene

spremenljivke v vsaki skupini. Prva skupina ima negativno povprečno vrednost –0,75 za spremenljivko o_rel_pol_med_z in prav tako tretja skupina z vrednostjo –0,61. Druga skupina ima medtem pozitivno povprečno vrednost za spremenljivko o_rel_pol_med_z, kar nam vsebinsko pove, da so uporabniki v drugi skupini v povprečju hitrejši kot v prvi in tretji skupini. Pri spremenljivki o_manj_napak_z ima prva skupina pozitivno povprečje vrednosti 0,79, medtem ko imata druga in tretja skupina negativno povprečno vrednost –0,19 in –1,4 zaporedno. Iz tega lahko razberemo, da so v povprečju uporabniki prve skupine naredili manj napak v primerjavi z drugo in tretjo skupino. Pri spremenljivki o_cas_prit_tipk_z pa je v drugi skupini pozitivno povprečje v vrednosti 0,36. V prvi in tretji skupini pa sta negativni povprečji spremenljivke z vrednostima –0,21 in –1,02, kar pomeni, da so uporabniki druge skupine v povprečju imeli krajši čas med pritiskom iz spustom tipk v primerjavi s prvo in tretjo skupino. Zadnji stolpec prikazuje povprečne vrednosti po skupinah za spremenljivko cas_z. V tem primeru pa negativno povprečje pomeni, da so uporabniki skupine v povprečju porabili manj časa za prepisovanje besedila. Druga skupina je edina, ki ima negativno povprečje spremenljivke cas_z z vrednostjo –0,53. To nam še dodatno potrdi razlago pri povprečju spremenljivke o_rel_pol_med_z. Prva in tretja skupina pa imata pozitivno povprečno vrednost v vrednosti 0,41 in 0,36 zaporedno. Interpretacija se osredotoča na analizo posebnosti posamezne skupine ter poudarja ključne razlike med skupinami. V našem primeru so iz analize razvidne jasne razlike med povprečji posamezne spremenljivke med tremi skupinami. Če interpretacija ne bi omogočala smiselnega razlikovanja med skupinami, bi morali spremeniti končno število skupin.

V koraku [15] pa razvrstimo uporabnike s pomočjo nehierarhične metode razvrščanja v skupine, imenovane metoda voditeljev. Pri tej metodi vnaprej določimo končno število skupin, ki je v našem primeru 3, in začetne voditelje, ki jih določimo na podlagi hierarhičnega pristopa z uporabo Wardovega alrgotima. V prvi iteraciji nehierarhičnega razvrščanja z metodo voditeljev enote pridružimo najbolj podobnemu začetnemu voditelju, nato pa ponovno izračunamo centre tako oblikovanih skupin, ki predstavljajo nove voditelje. Postopek združevanja se nadaljuje, pri čemer so nekatere enote lahko prerazvrščene med skupinami, kar pomeni, da se položaji voditeljev spreminjajo. Ko se voditelji ustalijo, se postopek zaključi, kar privede do optimizirane končne razvrstitve. Metoda voditeljev ima številne prednosti. Prvič, omogoča prerazvrščanje enot, kar pomeni, da se lahko v postopku razvrščanja spremenijo pripadnosti enot med skupinami. Drugič, končna razvrstitev je manj odvisna od prisotnosti osamelcev, načina izračuna razdalje med enotami in uporabe neprimernih ali nerelevantnih spremenljivk. Tretjič, metoda se lahko uporablja tudi za razvrščanje zelo velikega števila enot, kar prispeva k njegovi prilagodljivosti. Metoda pa ima tudi nekaj slabosti. Ena izmed njih je dejstvo, da je razvrstitev odvisna od vnaprej določenega števila skupin in izbire začetnih voditeljev, kar omejuje njeno prilagodljivost. Poleg tega je analiza večjega števila potencialnih razvrstitev težja, saj metoda ne omogoča grafičnega prikaza odstotka razvrščanja in generira sferične, podobno velike skupine, kar lahko omeji interpretacijo rezultatov. Nazadnje, metoda ne zagotavlja, da smo našli

optimalno razvrstitev, kar pomeni, da obstaja možnost, da je končna razvrstitev suboptimalna glede na specifične cilje analize (Baeldung, 2023).

Izpisani rezultat metode predstavlja povprečne vrednosti posameznih uporabljenih spremenljivk, ki so izračunane na podlagi uporabnikov posamezne skupine. Prva vrstica predstavlja povprečne vrednosti spremenljivk prve skupine, druga vrstica povprečne vrednosti druge skupine in tretja vrstica povprečne vrednosti tretje skupine. Vrednosti v matriki rahlo odstopajo od vrednosti, ki smo jih dobili v koraku [14], a to ne igra velike vloge pri interpretaciji uporabnikov v posamezni skupini. Pomemben je predvsem predznak vrednosti. Predznaki se ujemajo pri vseh vrednostih nove matrike z matriko, izpisano v koraku [14]. To nam še dodatno potrdi smiselnost razporeditve uporabnikov v skupine.

V koraku [16] izvajamo primerjavo razporeditve uporabnikov v skupine po Wardovem algoritmu in po metodi voditeljev. Z uporabo Wardovega algoritma smo v prvi skupini identificirali 35 uporabnikov, v drugi 54 uporabnikov in v tretji 11 uporabnikov. Po metodi voditeljev pa smo v prvi skupini identificirali 33 uporabnikov, v drugi skupini 54 uporabnikov in v tretji skupini 13 uporabnikov. Zadnja matrika rezultatov nam prikazuje, kako so se enote prerazvrstile po razvrščanju uporabnikov v skupine po obeh metodah. Vrednosti na diagonali matrike, od zgornjega levega kota proti spodnjemu desnemu, kažejo, da sta obe metodi razvrščanja enako razporedili 31 uporabnikov v prvo skupino, 51 uporabnikov v drugo skupino in 11 uporabnikov v tretjo skupino. Nadaljujemo z interpretacijo po posameznih vrsticah. Prva vrstica nam sporoča, da so bili trije uporabniki iz prve skupine po Wardovem algoritmu prerazvrščeni v drugo skupino, po metodi voditeljev in en uporabnik v tretjo skupino. V drugi vrstici matrike opazimo, da sta bila dva uporabnika iz druge skupine prerazvrščena v prvo skupino po metodi voditeljev in en uporabnik v tretjo skupino. Iz zadnje vrstice je razvidno, da noben uporabnik ni bil prerazvrščen iz tretje skupine, ne glede na metodo razvrščanja.

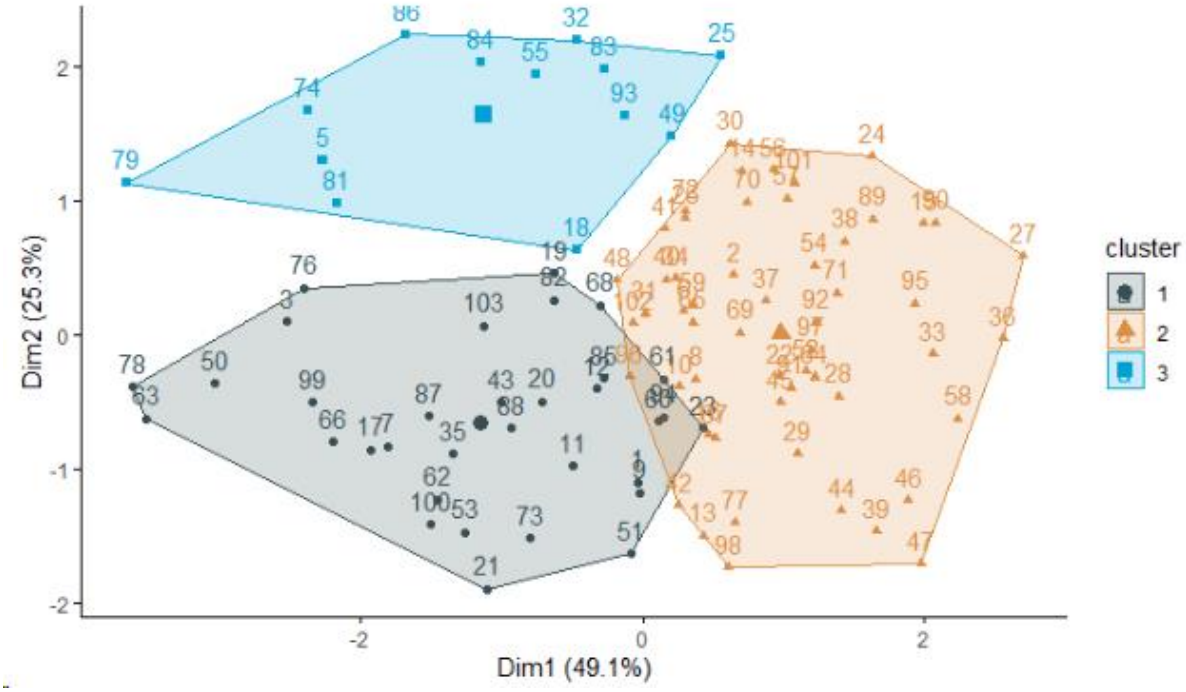
4.2 Prikaz rezultatov

V koraku [17] izrišemo vizualizacijo razvrščanja v skupine, kjer so skupine označene s specifičnimi barvami iz palete »jama«. To je razvidno iz slike 23. Enote (uporabniki) prve skupine so na vizualizaciji označene s črno barvo, enote druge skupine so na vizualizaciji označene z oranžno barvo in enote tretje skupine so označene z modro barvo. Vsaka enota je predstavljena s piko in izpisom zaporedne številke enote v seznamu nad njo, ki predstavlja identifikator enote. Robne enote posamezne skupine so med seboj povezane za lažjo predstavo raztega skupine. V vsaki skupini je razviden končni centroid skupine. V prvi skupini je predstavljen kot krog, v drugi skupini kot trikotnik in v tretji skupini kot kvadrat. Na grafu z oznako DIM1 (49,1 %) in DIM2 (25,3 %) sta označeni osi x in y. Te vrednosti predstavljajo odstotek variance v podatkih za vsako dimenzijo (DIM1 in DIM2):

- **DIM1 (49,1 %):** To pomeni, da prva dimenzija (os x) pojasni 49,1 % variance v podatkih. Višji odstotek pomeni, da ta dimenzija vsebuje več informacij o variabilnosti podatkov.
- **DIM2 (25 %):** Druga dimenzija (os y) pojasni 25,3 % variance v podatkih. Nižji odstotek kaže, da druga dimenzija vsebuje manj informacij o variabilnosti podatkov v primerjavi s prvo dimenzijo.

Odstotek variance predstavljajo delež celotne variabilnosti podatkov, ki ga je mogoče razložiti z vsako posamezno dimenzijo. Večji odstotek pomeni večjo pomembnost te dimenzije pri razložitvi variabilnosti v podatkih.

Slika 23: Vizualizacija skupin



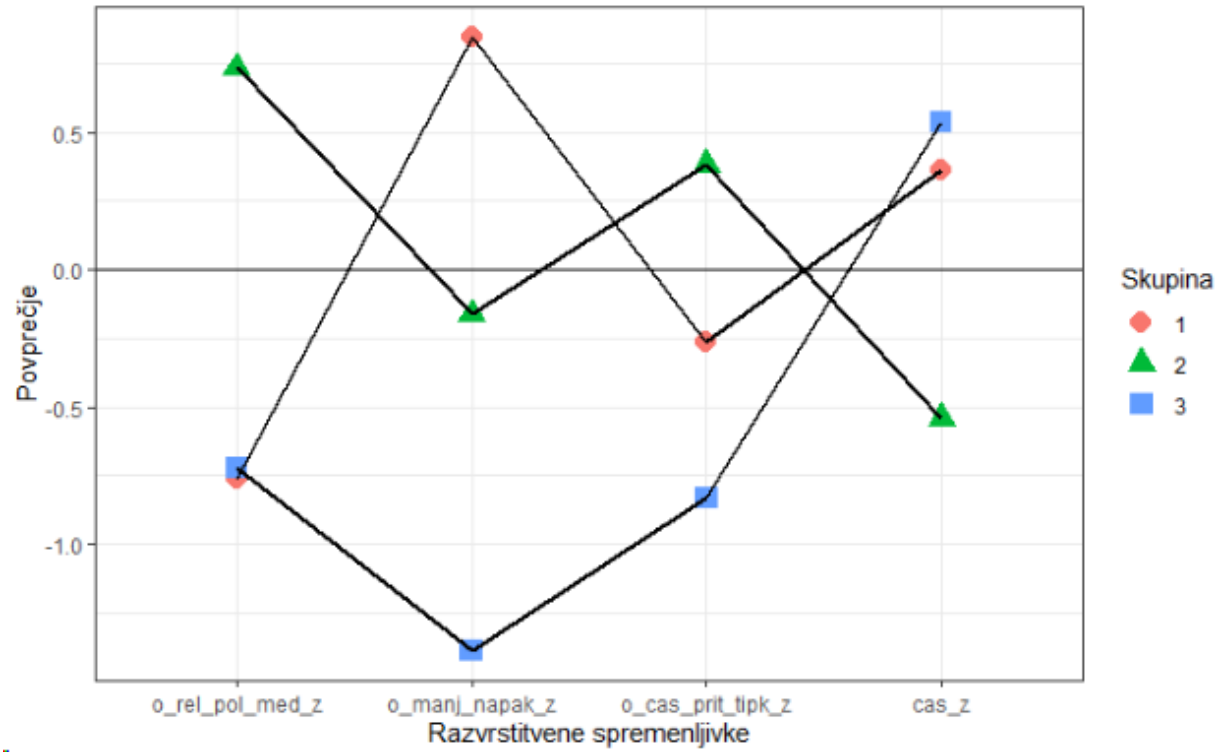
Vir: lastno delo.

V koraku [18] prikažemo povprečja uporabljenih spremenljivk po skupinah, ki smo jih dobili pri razporeditvi glavnih voditeljev (glej 24). Povprečja prve skupine so predstavljena z rdečim krogom, povprečja skupine dve so predstavljena z zelenim trikotnikom in povprečja skupine 3 so označena z modrim kvadratom. Spremenljivke so predstavljene na x-osi in si od leve proti desni sledijo: o_rel_pol_med_z, o_manj_napak_z, o_cas_prit_tipk_z in cas_z. Iz slike 24 je razvidno, da so med posameznimi skupinami velike razlike. Če interpretacija ne bi omogočala smiselnega razlikovanja med skupinami, bi morali spremeniti končno število skupin. Na podlagi vizualizacije lahko zaključimo naslednje:

- Uporabniki v drugi skupini so v povprečju hitrejši kot tisti v prvi in tretji skupini. Uporabniki v tretji skupini so v povprečju malo hitrejši od uporabnikov v prvi skupini.

- Uporabniki prve skupine so v povprečju naredili manj napak v primerjavi z drugo in tretjo skupino. Največ napak so v povprečju naredili uporabniki v tretji skupini.
- Uporabniki druge skupine imajo v povprečju krajši čas med pritiskom in spustom tipk v primerjavi s prvo in tretjo skupino. Najpočasnejši povprečni čas med pritiskom in spustom so imeli uporabniki tretje skupine.
- Uporabniki druge skupine porabijo v povprečju manj časa za prepisovanje besedila v primerjavi s preostalima skupinama. V povprečju je največ časa porabila tretja skupina, malo hitrejša pa je bila v povprečju prva skupina.

Slika 24: Predstavitev povprečij spremenljivk za posamezno skupino



Vir: lastno delo.

Prva skupina uporabnikov (33 % vseh uporabnikov) se izkaže za podpovprečno glede hitrosti tipkanja, kar je razvidno iz analize spremenljivke o_res_pol_med_z (povprečna vrednost manjša od 0) in spremenljivke cas_z (višja povprečna vrednost pomeni daljši čas prepisovanja besedila). Hkrati je opaziti, da je prva skupina podpovprečna tudi pri spremenljivki o_cas_prit_tipk_z, kar nakazuje na počasnejše tipkanje. Kljub temu pa je ta skupina nadpovprečna glede natančnosti, kar je razvidno iz spremenljivke o_manj_napak_z. V skupnem povzetku lahko trdimo, da prva skupina sicer tipka počasneje, vendar je bolj natančna in dela manj napak.

Uporabniki druge skupine (54 % vseh uporabnikov) predstavljajo nasprotje prvi skupini. Gre za skupino, ki v povprečju najhitrejša pri tipkanju, kar je razvidno iz analize spremenljivk p_rel_pol_med_z (nadpovprečna), o_cas_prit_tipk_z (nadpovprečna) in cas_z (vrednost pod

0, kar nakazuje na krajši čas prepisovanja besedila). Je pa druga skupina podpovprečna glede števila popravkov, kar pomeni, da kljub visoki hitrosti tipkanja ohranja nizko stopnjo natančnosti.

Tretja skupina uporabnikov (13 % vseh uporabnikov) je manj spretna pri tipkanju, kar se jasno kaže v rezultatih analize treh spremenljivk: o_res_pol_med_z, o_cas_prit_tipk_z in cas_z, kjer višja povprečna vrednost pomeni daljši čas pri prepisovanju besedila. Hkrati pa tudi v primerjavi z drugimi skupinami kaže podpovprečno število popravkov. Za to skupino je torej značilno, da počasi tipka in hkrati naredi veliko napak.

4.3 Analiza skupin

V nadaljevanju bomo analizirali značilnosti posamezne skupine. Spremenljivke, ki jih bomo opazovali, se klasificirajo v več tipov glede na naravo podatkov, ki jih predstavljajo. Osnovne kategorije so numerične (kvantitativne) in kategorialne (kvalitativne) spremenljivke. Obstajajo pa tudi drugi tipi spremenljivk, kot so ordinalne, binarne, diskretne in zvezne. V našem primeru imamo 2 numerični spremenljivki – starost in znanje angleščine (spremenljivka znanje_anglescine). Numerične spremenljivke predstavljajo količine in so merljive. Lahko so diskretne ali zvezne. V našem primeru sta obe diskretni spremenljivki. Pri omenjenih spremenljivkah nas bo predvsem zanimalo povprečje v posamezni skupini. Hkrati pa bomo izvedli še analizo variance (ANOVA) na podlagi linearnega modela, kjer bosta odvisni spremenljivki starost in znanje_anglescine (vsaka posebej). Neodvisna spremenljivka pa je faktor RazvrstitevK_MEANS, ki predstavlja rezultate razvrščanja v skupine z metodo voditeljev.

Namen analize ANOVA je predvsem ugotoviti, ali obstaja statistično značilna variabilnost v numerični spremenljivki med različnimi skupinami. Poleg numeričnih spremenljivk pa imamo 8 kategorialnih spremenljivk, s pomočjo katerih bomo še dodatno opisali skupine, pri tem pa bomo uporabili Pearsonov hi-kvadrat preizkus. S preizkusom bomo ugotavljali, ali obstaja statistično značilna povezanost med izbrano kategorialno spremenljivko in razvrstitvijo v skupino (RazvrstitevK_MEANS).

Kategorialne spremenljivke, o katerih bomo izvedli analizo, so:

- Spremenljivka spol (spol_f): Raziskujemo, ali obstaja statistično značilna razlika med skupinami, oblikovanimi z RazvrstitevK_MEANS, glede na spol uporabnikov.
- Spremenljivka, ki opisuje povprečen čas uporabe računalnika na dan (cas_za_racunalnikom_f): Želimo ugotoviti, ali obstaja statistično značilna razlika v času uporabe računalnika med skupinami, ki so bile oblikovane z RazvrstitevK_MEANS.
- Ali uporabnik uporablja računalnik več kot 4 ure povprečno na dan (vec_kot_4_ure_f): Preverjamo, ali obstaja statistično značilna razlika v razporeditvi med skupinami, ki so bile ustvarjene z RazvrstitevK_MEANS, glede na to, ali uporabniki v povprečju uporabljajo računalnik več kot 4 ure na dan ali ne.

- Spremenljivka, ki opisuje stopnjo izobrazbe uporabnika (stopnja_sole_f): Raziskujemo, ali obstaja statistično značilna razlika v stopnji izobrazbe med skupinami, ki so bile oblikovane z RazvrstitevK_MEANS.
- Spremenljivka, ki opisuje, ali ima uporabnik dokončano dodiplomsko stopnjo izobrazbe ali ne (ima_diplomo_f): Preverjamo, ali obstaja statistično značilna razlika v razporeditvi med skupinami, oblikovanimi z RazvrstitevK_MEANS, glede na to, ali imajo uporabniki dokončano dodiplomsko stopnjo izobrazbe ali ne.
- Spremenljivka, ki opisuje, ali ima uporabnik dokončano dodiplomsko stopnjo izobrazbe ali višjo stopnjo (ima_diplomo_visjo_izobrazbo_f): Želimo ugotoviti, ali obstaja statistično značilna razlika med skupinami, oblikovanimi z RazvrstitevK_MEANS, glede na to, ali imajo uporabniki dokončano dodiplomsko stopnjo izobrazbe ali celo višjo.
- Spremenljivka slepega tipkanja (slepo_tipkanje_f): Preverjamo, ali obstaja statistično značilna razlika med skupinami, ki so bile ustvarjene z RazvrstitevK_MEANS, glede na to, ali uporabniki tipkajo slepo ali ne.
- Spremenljivka, ki nam pove, ali je uporabnik levičar ali desničar (spretnejsa_roka_f): Raziskujemo, ali obstaja statistično značilna razlika v razporeditvi med skupinami, ki so bile oblikovane z RazvrstitevK_MEANS, glede na to, ali uporabniki preferirajo levo ali desno roko.

4.3.1 Analiza numeričnih spremenljivk

V koraku [19] izvedemo analizo variance (ANOVA) in izračun povprečnih starosti za posamezno skupino. Ugotovimo, da ima najvišjo povprečno starost prva skupina, in sicer povprečna starost znaša 37,2 leta. Najnižjo povprečno starost ima druga skupina, kjer povprečna starost znaša 27,3 leta. Ostane še tretja skupina, katere povprečna starost znaša 31,8 leta.

Rezultati analize variance (ANOVA) so razčlenjeni na dva dela: učinek faktorja RazvrstitevK_MEANS (prva vrstica rezultata) in preostanek Residuals (druga vrstica rezultata).

Slika 25:Rezultati analize variance (ANOVA)

```
fit <- aov(starost ~ as.factor(RazvrstitevK_MEANS), data = podatki)
summary(fit)
```

```
##                             Df Sum Sq Mean Sq F value   Pr(>F)
## as.factor(RazvrstitevK_MEANS)  2   2035  1017.6   10.98 5.03e-05 ***
## Residuals                     97   8990    92.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vir: lastno delo.

Učinek faktorja RazvrstitevK_MEANS:

- Df (Degrees of Freedom): 2 stopinji prostosti, saj imamo tri skupine v faktorju RazvrstitevK_MEANS.
- Sum Sq (Sum of Squares): 2035, to je vsota kvadratov variabilnosti, ki jo povzroča faktor RazvrstitevK_MEANS.
- Mean Sq (Mean Square): 1017,6, povprečna vsota kvadratov za faktor RazvrstitevK_MEANS.
- F value (F-test statistic): 10,98, vrednost testne statistike F, ki primerja variabilnost, razloženo s faktorjem z variabilnostjo preostanka.
- Pr(>F) (*p*-vrednost): 5,03e-05 (0,001), zelo majhna *p*-vrednost, kar pomeni, da obstaja statistično značilna razlika med skupinami v faktorju RazvrstitevK_MEANS.

Ostanek (Residuals):

- Df (Degrees of Freedom): 97 stopinj prostosti, kar predstavlja preostanek variabilnosti v podatkih.
- Sum Sq (Sum of Squares): 8990, to je vsota kvadratov preostale variabilnosti, ki ni razložena s faktorjem.
- Mean Sq (Mean Square): 92,7, povprečna vsota kvadratov za preostanek variacije.

- Rezultati poudarjajo statistično značilno razliko v povprečjih med skupinami, ki so bile oblikovane z razvrščanjem RazvrstitevK_MEANS, v kontekstu starostne spremenljivke (age). *P*-vrednost, ki je zelo majhna (manjša od 0,05), kaže na to, da je učinek faktorja statistično značilen.

V koraku [20] postopek ponovimo za numerično spremenljivko ang. Iz povprečij ocen znanja angleščine ugotovimo, da ima najboljšo oceno znanja angleškega jezika druga skupina (4,04), najnižjo povprečno oceno pa prva skupina (3,45). Tretja skupina ima povprečno oceno znanja angleškega jezika 3,54. Rezultati analize variance (ANOVA) kažejo, da obstaja statistično značilna razlika v povprečnih ocenah znanja angleškega jezika med skupinami, ki so bile oblikovane z razvrščanjem RazvrstitevK_MEANS. *P*-vrednost 0,005 (pod 0,05) kaže na to, da so vsaj nekatere skupine različne glede na povprečne ocene znanja angleškega jezika.

4.3.2 Analiza kategorialnih spremenljivk

Pri izvedbi Pearsonovega hi-kvarat preizkusa je treba preveriti nekaj predpostavk. Prvič, opazovanja so med seboj neodvisna. Drugič, vse pričakovane frekvence so višje od 5. Tretja predpostavka predpostavlja, da je v kontingenčnih tabelah, ki imajo več kot dve kategoriji, lahko največ 20 % pričakovanih frekvenc nižjih od 5, vendar to privede do zmanjšane moči preizkusa, vse pa vedno višje od 1

Prva spremenljivka, s katero bomo izvedli Pearsonov hi-kvarat preizkus, je spremenljivka spol_f. To naredimo v koraku [21]. Izpisani rezultat je sestavljen iz štirih delov. V prvem delu nas zanima *p*-vrednost, v drugem delu imamo kontingenčno tabelo, v kateri so predstavljene empirične frekvence za vsako kategorijo po skupinah. Nato sledi izpis matrike pričakovanih frekvenc in na koncu še standardizirani ostanki. Standardizirani ostanki merijo, kakšna je razlika med dejansko in pričakovano frekvenco. Izračunani so po formuli (3).

$$Std.ostanek = \frac{Empirična\ frekvenca - Pričakovana\ frekvenca}{\sqrt{Pričakovana} frekvenca} \quad (3)$$

Iz rezultatov lahko razberemo, da je p-vrednost enaka 0,005. To kaže na statistično značilno povezavo med spolom in RazvrstitevK_MEANS. Iz standardiziranih ostankov je razvidno, da je v prvi skupini mnogo več uporabnikov ženskega spola, kot je bilo pričakovano, saj je vrednost ostanka 2,15. V drugi in tretji skupini pa je več kot pričakovano število uporabnikov moškega spola. Metoda nas hkrati opozori, da p-vrednost ni zanesljiva, z opozorilom »Warning: Chi-squared approximation may be incorrect«. Razlog za to je predvsem velikost vzorca, ki je premajhen. Podrobnejše analize kontingenčne tabele razkrivajo, da v tretji skupini obstajata le 2 uporabnika ženskega spola. Čeprav Pearsonov hi-kvarat preizkus dopušča, da je do 20 % pričakovanih frekvenc nižjih od 5 v tabelah z več kot dvema kategorijama, to vpliva na moč preizkusa. V koraku [22] izvedemo še Fisherjev natančni preizkus neodvisnosti (angl. Fisher's Exact test). Fisherjev test je statistični test, ki se uporablja za ugotavljanje povezanosti med dvema kategorialnima spremenljivkama v primerih, ko je število opazovanj majhno. Ime testa izhaja iz imena britanskega statistika Sira R. A. Fisherja, ki ga je razvil leta 1935 (Upton, 1992). Zavrnemo ničelno domnevo pri p = 0,006, kar pomeni, da med spol_f in RazvrstitevK_MEAN obstaja statistična povezanost.

V drugem testiranju spremenljivke cas_za_racunalnikom_f smo v koraku [23] zabeležili *p*-vrednost 0,041, kar je nižje od 0,05. To nakazuje, da obstajajo statistično značilne razlike med vsaj nekaterimi skupinami glede na povprečni čas uporabe računalnika na dan. Kljub temu je treba omeniti, da se v kontingenčni tabeli pojavlja več vrednosti blizu 5 ali celo 0, zlasti v kombinaciji tretje skupine uporabnikov, ki porabijo manj kot 1 uro dnevno. Tudi pri spremenljivki cas_za_racunalnikom_f izvedemo Fisherjev test, ki nam vrne *p*-vrednost 0,052. Ne moremo trditi, da bi bili spremenljivki cas_za_racunalnikom_f in RazvrstitevK_MEANS statistično povezani, saj je *p*-vrednost večja od 0,05 in hkrati je kršena predpostavka Fisherjevega testa.

Nato testiramo povezavo med RazvrstitevK_MEANS in spremenljivko vec_kot_4_ure_f v koraku [25]. *P*-vrednost 0,179 v koraku [25] presega običajni prag 0,05, zato ne moremo potrditi statistično značilne povezave med tema dvema spremenljivkama. Izvedemo še Fisherjev test v koraku [26], ki nam vrne *p*-vrednost 0,167. Ne moremo trditi, da bi bili spremenljivki vec_kot_4_ure_f in RazvrstitevK_MEANS statistično povezani.

Iz rezultatov je še razvidno, da večina enot povprečno preživi več kot štiri ure na dan za računalnikom, in sicer v prvi skupini 18 % vseh enot, v drugi skupini 39 % vseh enot in v tretji skupini 7 % vseh enot.

Pri testiranju spremenljivke stopnja_sole_f smo opazili, da imamo 6 kategorij, kar pomeni, da so vrednosti v kontingenčni tabeli pod 5. Zaradi tega test ni zanesljiv. Večina enot ima končan diplomski študij, in sicer 17 % vseh enot v prvi skupini, 28 % v drugi skupini in 4 % v tretji skupini. Ponovili smo preizkus s poenostavljenimi spremenljivkami ima_diplomo_f in ima_diplomo_visjo_izobrazbo_f v korakih [29] in [31], ki izpolnjujeta predpostavke, vendar *p*-vrednosti za obe spremenljivki (0,37 in 0,928 zaporedno) nista dovolj nizki, da bi potrdili statistično značilno povezavo z RazvrstitevK_MEANS. V korakih [30] in [32] še potrdimo hipotezo s Fisherjevim testom, kjer dobimo *p*-vrednost enako 0,422 za spremenljivko ima_diplomo_f in *p*-vrednost 0,9405 za spremenljivko ima_diplomo_visjo_izobrazbo_f.

Test za spremenljivko slepo_tipkanje_f v koraku [33] je pokazal *p*-vrednost 0,034, kar kaže na statistično značilno povezavo med slepim tipkanjem in RazvrstitevK_MEANS. V prvi skupini je več uporabnikov, ki ne tipkajo slepo, medtem ko je v drugi in tretji skupini več slepih tipkarjev, kot smo pričakovali. Iz rezultatov je hkrati razvidno, da je izrazito manj kot pričakovano število uporabnikov v prvi skupini, ki tipkajo slepo, saj je vrednost ostankov – 1,64. Pomembno je še izpostaviti, da je 25 % vseh enot, ki slepo tipkajo, in 29 % enot, ki slepo ne tipkajo, v drugi skupini. V prvi skupini pa je kar 26 % vseh enot, ki ne znajo slepo tipkati.

Pri spremenljivki spretnejsa_roka_f smo v koraku [35] dobili *p*-vrednost 0,004, vendar so bile kršene druga in tretja predpostavka. Zaradi majhnega števila levičarjev v celotni množici (štirje uporabniki) test ni zanesljiv. Izvedemo še Fisherjev test v koraku [36], ki nam vrne vrednost 0,02747. To kaže na statistično značilno povezavo med spretnejsa_roka_f in RazvrstitevK_MEANS.

5 OVREDNOTENJE REZULTATOV

5.1 Uporabna vrednost raziskave

Pri analizi smo uporabnike razdelili v tri skupine z metodo voditeljev, kar nam je omogočilo smiselno razlikovanje med njimi. V povprečju so uporabniki druge skupine hitrejši od uporabnikov prve in tretje skupine, medtem ko so uporabniki tretje skupine v povprečju malenkost hitrejši od uporabnikov prve skupine. Glede na število napak so uporabniki prve skupine v povprečju naredili manj napak v primerjavi z drugo in tretjo skupino, pri čemer so uporabniki tretje skupine naredili največ napak. Povprečni čas med pritiskom in spustom tipk je krajši pri uporabnikih druge skupine v primerjavi s prvo in tretjo skupino, medtem ko so uporabniki tretje skupine imeli najdaljši povprečni čas med pritiskom in spustom.

Uporabniki druge skupine v povprečju porabijo manj časa za prepisovanje besedila kot uporabniki preostalih skupin, pri čemer tretja skupina porabi največ časa, medtem ko je prva skupina malo hitrejša v povprečju. Glede starosti ima prva skupina najvišjo povprečno starost (37,2 leta), druga skupina najnižjo (27,3 leta), tretja pa ima povprečno starost 31,8 leta. Raziskava jasno kaže, da obstaja pomembna statistična razlika med skupinami glede na starost.

Ti rezultati so v skladu z ugotovitvami raziskovalcev Uzun in drugi (2015), ki so prav tako preučevali klasifikacijo uporabnikov glede na starost. Povprečne ocene znanja angleščine kažejo, da ima druga skupina najboljšo oceno (4,04), prva skupina najnižjo povprečno oceno (3,45), medtem ko tretja skupina dosega povprečno oceno 3,54. Rezultati analize variance (ANOVA) poudarjajo statistično značilno razliko v povprečnih ocenah znanja angleškega jezika med skupinami, ki so bile oblikovane z razvrščanjem RazvrstitevK_MEANS. *P*-vrednost 0,005 kaže, da so vsaj nekatere skupine različne glede na povprečne ocene znanja angleškega jezika. Iz standardiziranih ostankov lahko razberemo, da je v prvi skupini izrazito večje kot pričakovano število uporabnikov ženskega spola. Iz rezultatov je hkrati razvidno, da je izrazito manjše kot pričakovano število uporabnikov v prvi skupini, ki tipkajo slepo, saj je vrednost ostankov –1,64. *P*-vrednost 0,005 jasno kaže, da obstaja statistično pomembna povezava med spolom in RazvrstitevK_MEANS. Ti rezultati so v skladu z ugotovitvami raziskovalcev Giot in Rosenberger (2012), ki sta pokazala, da je mogoče prepoznati spol posameznika glede na način tipkanja fiksnega besedila.

Rezultate analize lahko nadgradimo tako, da za vsako skupino opišemo, kako bi lahko izkoriščali njihovo znanje v trženjske namene. V našem primeru smo ugotovili, da so v povprečju uporabniki prve skupine najpočasnejši z vidika tipkanja, a so pri tem naredili tudi najmanj napak. Prva skupina ima najvišjo povprečno starost, ki znaša 37,2 leta, in najnižjo povprečno oceno znanja angleščine. V prvi skupini je izrazito večje kot pričakovano število uporabnic ženskega spola. Oglaševanje za prvo skupino bi lahko bilo prilagojeno na način, da so oglasi v slovenskem jeziku ali pa oglašujejo izobraževalne programe za izboljšanje znanja angleščine. Za doseganje ciljev v oglaševanju pri ciljni množici porabnikov ženskega spola, starih okoli 37 let, ki tvorijo pomembno skupino v tej starostni kategoriji, bi bilo smiselno trženjske dejavnosti usmeriti proti raznolikemu naboru izdelkov in storitev, ki jih glede na rezultate trženjskih raziskav kupuje ta ciljna skupina. S takšnim pristopom bi bilo mogoče zadovoljiti različne potrebe in interese ciljne skupine, hkrati pa izkoristiti njihovo izrazito številčnost v tržnem prostoru. Ključno pri tem pa je razumevanje njihovih specifičnih potreb in interesov. Na tej podlagi bi jim lahko ponudili izdelke, ki odražajo njihov življenjski slog in vrednote.

V povprečju so uporabniki druge skupine pri tipkanju hitrejši od uporabnikov prve in tretje skupine. Druga skupina ima najnižjo povprečno starost, ki znaša 27,3 leta. Povprečne ocene znanja angleščine kažejo, da ima druga skupina najvišjo oceno, ki znaša 4,04. Pri oglaševanju, usmerjenem na drugo skupino porabnikov bi se lahko usmerili na modno

industrijo oblačil in dodatkov v skladu z aktualnimi modnimi trendi za mlajše generacije, medtem ko bi ponudniki potovanj lahko privabljali s posebnimi paketi za mlade popotnike. V digitalni tehnologiji se lahko oglašujejo pametne naprave in aplikacije, ki olajšajo vsakodnevno življenje mladih. Športne in rekreacijske ponudbe, glasba ter koncerti, skupaj z digitalnimi produkti, kot so aplikacije za organizacijo opravkov ali ustvarjanje vsebin, predstavljajo dodatne priložnosti za nagovarjanje te ciljne skupine.

Tretjo skupino sestavlja 13 % celotnega vzorca uporabnikov, od tega 11 % enot moškega spola. Povprečna starost skupine znaša 32,8 leta. Tretja skupina je izziv za natančno opredelitev primerne oglaševalske vsebine zaradi svoje relativno majhne velikosti. Kljub temu bi bilo smiselno raziskati njihove specifične interese, da bi razvili prilagojene trženjske strategije, usmerjene v doseganje uspešnega oglaševanja v tej skupini.

Poleg eksperimentalnih ugotovitev menimo, da je še en pomemben prispevek te študije množica zbranih podatkov. Čeprav obstaja več nizov podatkov, ki jih je mogoče uporabiti za preverjanje porabnikov na podlagi dinamike tipkanja, je naš niz podatkov prvi, ki vsebuje informacije o starosti, spolu, bivališču, stopnji izobrazbe, koliko časa uporabnik povprečno preživi za računalnikom dnevno, ali uporabnik slepo tipka, ali pri tipkanju uporablja eno ali dve roki, ali je desničar ali levičar in oceno znanja angleščine uporabnika, ter je javno dostopen. Poleg tega objavljamo tudi merski instrument za zbiranje podatkov in kodo analize, tako da lahko raziskavo enostavno ponovijo tudi drugi raziskovalci ali pa podjetja z oglaševalskimi nameni.

5.2 Omejitve raziskave in priložnosti za prihodnje raziskovanje

Raziskava kljub dosedanjim ugotovitvam še vedno ponuja veliko priložnosti za izboljšave, ki jih je pomembno izpostaviti. Prva pomembna pomanjkljivost je omejeno število zajetih enot, saj smo spremljali le 103 uporabnike, v končno analizo pa smo jih vključili 100. Za večjo zanesljivost bi bilo nujno razširiti vzorec na vsaj 500 ali več uporabnikov. Tudi besedilo, ki so ga uporabniki prepisovali, bi lahko bilo daljše in bi lahko zajemalo večji odstotek vseh kombinacij tipk za natančnejše beleženje vrednosti. Trenutno besedilo je namreč zajemalo samo 50 % vseh kombinacij tipk, ki smo jih beležili v matriki. Dodatno bi lahko vključili sledenje miškinega gibanja in časovnega razmika med kliki ter sledenje, s katerim prstom v večini klikamo na posamezno tipko. Pri predhodni anketi bi dodali dodatna vprašanja o delovni dobi uporabnika, stroki v kateri uporabnik dela, višini uporabnika, športnih aktivnostih, s katerimi se uporabnik ukvarja, povprečno število ur spanja uporabnika, itd. Povečali bi lahko tudi raznolikost nalog tipkanja. Naloge bi se lahko osredotočale na hitrost, natančnost in uporabo tipkovnice v različnih jezikih ter s posebnimi znaki. Razširitev vzorca na globalno raven in vključitev mlajših od 15 let bi prav tako prispevala k celovitosti analize. Metoda voditeljev pri razvrščanju uporabnikov ima prav tako svoje pomanjkljivosti, kot so odvisnost rezultata od števila skupin in izbire začetnih voditeljev, ki se jih določi vnaprej. Hkrati metoda ne zajema vizualne predstavitve procesa

razvrščanja. Alternativne metode razvrščanja bi lahko prinesle dodatno perspektivo in primerjalno analizo rezultatov.

6 SKLEP

Za številna podjetja predstavlja ključno vodilo prepoznati svojo osrednjo skupino porabnikov, znano tudi kot ciljni trg, saj je to nepogrešljiv korak pri razvoju novih izdelkov ali storitev ter pri vstopanju na nove trge. Razumevanje ciljnega trga ne predstavlja zgolj izhodišče pri oblikovanju produkta, temveč igra ključno vlogo tudi pri izvajanju učinkovitih trženjskih strategij ter izbiri primernih prodajnih kanalov. Temeljito poznavanje potreb in želja ciljnega trga je ključno za uspešno izvajanje ciljno usmerjenih promocij, skrb za ključne porabnike, oblikovanje izdelkov, ki so prilagojeni njihovim specifičnim potrebam, izbiro ustrezne distribucijske poti ter zagotavljanje storitev in podpore, ki so prilagojene zahtevam izbrane ciljne skupine. S tem pristopom podjetje ne le zadovoljuje potrebe trga, temveč gradi trajne odnose s svojimi strankami, kar lahko vodi do dolgoročne lojalnosti in konkurenčne prednosti na trgu. Celovito razumevanje ciljnega trga torej ne le oblikuje uspešen izdelek, temveč predstavlja temelj za vzpostavitev trajnostnega in uspešnega poslovnega modela.

Na temelju preučitve strokovne in znanstvene literature, razvoja programske opreme merskega instrumenta ter analize in interpretacije rezultatov lahko sklepamo, da smo v okviru magistrskega dela pridobili pomembne ugotovitve in vpogled v različne vidike uporabe tehnologije zaznavanja vzorcev tipkanja. Razvili smo zanesljiv merilni instrument, ki se lahko uporabi tudi v drugih podobnih raziskavah. Z uporabo merskega instrumenta smo pridobili podatke o vzorcih tipkanja 103 uporabnikov, pri čemer smo končno analizo izvedli na vzorcu 100 uporabnikov.

V uvodu smo si postavili dve osnovni vprašanji za raziskavo. Prvo vprašanje se nanaša na to, ali je mogoče na podlagi načina tipkanja porabnika napovedati njegove sociodemografske značilnosti. Rezultati naše analize kažejo, da je mogoče določiti nekatere sociodemografske značilnosti, kot so spol, starost in raven znanja angleščine, na podlagi vzorca tipkanja. Vendar pa nismo uspeli zanesljivo določiti drugih značilnosti, kot so povprečen čas, ki ga porabnik preživi za računalnikom dnevno, stopnja izobrazbe, roka (levičar ali desničar) in spretnost slepega tipkanja. Pri tem se moramo zavedati, da je možen razlog za neuspeh premajhno število enot vključenih v analizo.

Drugo raziskovalno vprašanje se osredotoča na možnost oblikovanja manjšega števila skupin (segmentov) porabnikov glede na vzorec tipkanja. Rezultati analize kažejo, da je mogoče porabnike razdeliti v manjše skupine. V našem primeru smo identificirali 3 skupine, ki se bistveno razlikujejo med seboj. Prva skupina, ki predstavlja 33 % vseh enot vključenih v analizi, je značilna po tem, da tipka podpovprečno hitro, vendar z nadpovprečno natančnostjo in posledično manjšim številom popravkov. Druga skupina, ki obsega 54 %

enot vključenih v analizi, pa je nadpovprečno hitra pri tipkanju, vendar ima podpovprečno število popravkov. Tretja skupina uporabnikov, ki predstavlja 13 % enot vključenih v analizi, je podpovprečno hitra pri tipkanju in hkrati podpovprečna pri številu popravkov. Tretja skupina uporabnikov, ki predstavlja 13 % enot v analizi, je podpovprečno hitra pri tipkanju in hkrati podpovprečna pri številu popravkov.

Na podlagi teh ugotovitev je analiza prinesla pomembne prednosti na področju trženja. Podjetja bi lahko prilagodila oglaševalske strategije, ciljajoč na specifične skupine porabnikov z ustrezno demografsko usmerjenostjo. Razumevanje razlik v hitrosti in natančnosti tipkanja lahko omogoča oblikovanje bolj prilagojenih trženjskih kampanj, ki nagovarjajo posamezne skupine z ustreznimi oglasi. Personalizirane trženjske strategije bi se lahko osredotočile na določene demografske značilnosti posameznih skupin, izhajajoč iz analize tipkanja. To bi lahko pripomoglo k bolj ciljanim in učinkovitejšim trženjskim pristopom, ki bolje zadovoljujejo potrebe posameznih ciljnih skupin porabnikov. Te ugotovitve predstavljajo pomemben korak naprej, zlasti za področje trženja.

Raziskava ponuja različne priložnosti za izboljšave. Ključne pomanjkljivosti vključujejo omejen vzorec (le 100 uporabnikov), krajše besedilo za beleženje tipk (50 % kombinacij), pomanjkanje sledenja miškinega gibanja in časovnega razmika med kliki. Dodatne možnosti za izboljšave obsegajo vprašanja o uporabnikovi delovni dobi, stroki, višini, športnih aktivnostih itd. Raznolikost nalog tipkanja bi lahko vključevala hitrost, natančnost, uporabo tipkovnice v različnih jezikih ter s posebnimi znaki. Razširitev vzorca na globalno raven in vključitev mlajših od 15 let bi prispevala k celovitosti analize. Metoda voditeljev pri razvrščanju ima omejitve, vključno z odvisnostjo od števila skupin in izbire začetnih voditeljev, brez vizualne predstavitve procesa razvrščanja. Alternativne metode razvrščanja bi lahko prispevale k celovitejši analizi rezultatov.

LITERATURA IN VIRI

1. Monem, A. H. (2021). The effectiveness of advertising personalization. *Journal of Design Sciences and Applied Arts, 2*(1), 114–121.
2. Aditya. (2022, 9. december). *Hierarchical clustering: applications, advantages, and disadvantages*. Pridobljeno 31. novembra 2023 s https://codinginfinite.com/hierarchical-clustering-applications-advantages-and-disadvantages/
3. Amazon Web Services. (brez datuma). *What is Java? - Enterprise Java Beginner's Guide*. Pridobljeno 24. aprila 2023 s https://aws.amazon.com/what-is/java/
4. Baeldung. (2023, julij). *The drawbacks of K-Means algorithm. Baeldung on Computer Science*. Pridobljeno 11. decembra 2023 s https://www.baeldung.com/cs/k-means-flaws-improvements
5. Buza, K. in Farou, Z. (2019). *Fraud Detection based on Keystroke Dynamics*. Pridobljeno 18. aprila 2023 s http://t-labs.elte.hu/wp-content/uploads/MSc_Thesis_FZFXIH.pdf

6. Curtis, K. R. in Allen, S. (2018). *Target Market Identification and Data Collection Methods*. Pridobljeno 28. decembra 2022 s https://extension.usu.edu/apec/files/uploads/Target_Market_Identification.pdf
7. Daignault, M., Shepherd, M., Marche, S. in Watters, C. (2002, oktober). Enabling trust online. V *Proceedings. Third International Symposium on Electronic Commerce* (str. 3–12). IEEE.
8. DataReportal. (brez datuma). *Digital around the world*. Pridobljeno 16. novembra 2023 s https://datareportal.com/global-digital-overview
9. Epp, C., Lippold, M. in Mandryk, R. L. (2011, 7. maj). Identifying emotional states using keystroke dynamics. *Proceedings of the sigchi conference on human factors in computing systems.* Association for Computing Machinery (str. 715–724). Association for Computing Machinery.
10. Gaines, R., Lisowski, W., Press, S. in Shapiro, N. (1980, maj). *Authentication by Keystroke Timing: Some Preliminary Results. 52*. Pridobljeno 12. aprila 2023 s https://www.researchgate.net/publication/235089478_Authentication_by_Keystroke_Timing_Some_Preliminary_Results
11. Gales, J. S. (2022, november). *Google Ads Audience Targeting: 15 Powerful & Underused Strategies*. Pridobljeno 20. marca 2023 s https://www.wordstream.com/blog/ws/2022/09/21/google-ads-audience-targeting-cheat-sheet
12. Giot, R. in Rosenberger, C. (2012). A new soft biometric approach for keystroke dynamics based on gender recognition. *International Journal of Information Technology and Management, 11*(1-2), 35–49.
13. Glavan, T. (2022). *Uporaba google analitike za povečanje prodaje v spletnih trgovinah v Sloveniji* (magistrsko delo). Ekonomska fakulteta Univerze v Ljubljani.
14. Hanna, K. T. (2023). What is the QWERTY keyboard?. *TechTarget.* Pridobljeno 18. junija2023 s https://www.techtarget.com/whatis/definition/QWERTY-keyboard
15. Heller, M. (2022, julij). What is Visual Studio Code? Microsoft's extensible code editor. *InfoWorld*. Pridobljeno 28. aprila 2023 s https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html
16. Leggett, J. in Williams, G. (1988). Verifying identity via keystroke characterstics. *International Journal of Man-Machine Studies, 28*(1), 67–76. doi:10.1016/S0020-7373(88)80053-1.
17. Leinonen, J. (2019). University of Helsinki, Finland. *Keystroke Data in Programming Courses*. Pridobljeno 28. decembra 2022 s https://tuhat.helsinki.fi/ws/files/138592784/Keystroke.pdf
18. Idrus, S. Z. S., Cherrier, E., Rosenberger, C. in Bours, P. (2013). Soft Biometrics for Keystroke Dynamics. V M. Kamel in A. Campilho (ur.), *Image Analysis and Recognition. ICIAR 2013. Lecture Notes in Computer Science, vol 7950* (str. 11-18). Springer.

19. Jain, A. K., Dass, S. C. in Nandakumar, K. (2004). Soft Biometric Traits for Personal Recognition Systems. V D. Zhang in Jain, A. K. (ur.), *Biometric Authentication. ICBA 2004. Lecture Notes in Computer Science, vol 3072* (str. 731-738). Springer.
20. Kotler, P., Burton, S., Deans, K., Brown, L. in Armstrong, G. (2015). *Marketing*. Pearson Higher Education AU.
21. Magnusson, A. (2023, februar). *The Definitive Guide to Authentication*. Pridobljeno 30. aprila 2023 s https://www.strongdm.com/authentication
22. Maheshwary, S., Ganguly, S. in Pudi, V. (2017). Deep secure: A fast and simple neural network based approach for user authentication and identification via keystroke dynamics. *International Joint Conference on Artificial Intelligence*.
23. Mijwil, M., Unogwu, O. J., Filali, Y., Bala, I. in Al-Shahwani, H. (2023). Exploring the top five evolving threats in cybersecurity: An in-depth overview. *Mesopotamian journal of cybersecurity*, 2023, 57–63.
24. Monrose, F. in Rubin, A. (1997, april). Authentication via keystroke dynamics. *Proceedings of the 4th ACM Conference on Computer and Communications Security* (str. 48–56).
25. Patel, B. (2023). *What is web marketing and why it is essential for businesses?* [objava na blogu]. Pridobljeno 16. novembra 2023 s https://www.brainvire.com/blog/web-marketing/#What_Is_Web_Marketing
26. Pons, A. P. (2006). Biometric marketing: targeting the online consumer. *Communications of the ACM*, *49*(8), 60–66.
27. Random text generator. (brez datuma). *Get random text for web or typography*. Pridobljeno 18. junija s https://randomtextgenerator.com/
28. Swedin, E. G. in Ferro, D. L. (2007). Computers: the life story of a technology. *JHU Press*. Pridobljeno 29. decembra 2023 s https://books.google.si/books?hl=en&lr=&id=IJXYoPiwvOMC&oi=fnd&pg=PR5&dq=How+Did+Computers+Change+The+World&ots=uKQl13C6_F&sig=IT3H-uyb92p7gFKtqpC0F81WhZQ&redir_esc=y#v=onepage&q=How%20Did%20Computers%20Change%20The%20World&f=false
29. Teradata. (brez datuma). *What is Python?*. Pridobljeno 22. aprila 2023 s https://www.teradata.com/Glossary/What-is-Python
30. Tripathi, S., Arroyo-Gallego, T. in Giancardo, L. (2022). Keystroke-dynamics for Parkinson's disease signs detection in an at-home uncontrolled population: a new benchmark and method. *IEEE Transactions on Biomedical Engineering*, *70*(1), 182-192.
31. Tutorialspoint. (brez datuma). *Eclipse – Overview*. Pridobljeno 25. aprila 2023 s https://www.tutorialspoint.com/eclipse/eclipse_overview.htm
32. Upton, G. J. (1992). Fisher's exact test. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *155*(3), 395-402.
33. Uzun, Y., Bicakci, K. in Uzunay, Y. (2015). *Could we distinguish child users from adults using keystroke dynamics?*. Pridobljeno 3. januarja 2024 s https://arxiv.org/abs/1511.05672

34. Variyar, M. (2013, februar). *82% children on Facebook receive vulgar messages*. Pridobljeno 24. oktobra 2023 s https://www.hindustantimes.com/mumbai/82-children-on-facebook-get-vulgar-messages/story-0d532SUH4E2kYDN4o1ja8H.html
35. Vizer, L. M., Zhou, L. in Sears, A. (2009). Automated stress detection using keystroke and linguistic features: An exploratory study. *International Journal of Human-Computer Studies*, 67(10), 870-886.
36. R (programski jezik). (2022). V *Wikipedija*. Pridobljeno 17. december 2023 s https://hr.wikipedia.org/wiki/R_(programski_jezik)
37. Wright, K. (2022). Will the Real Hopkins Statistic Please Stand Up?. *R Journal*, *14*(3), 282-292.
38. Zhao, Y. (2006, december). Learning user keystroke patterns for authentication. *Proceeding of World Academy of Science, Engineering and Technology* (Vol. 14, str. 65–70). Pridobljeno 9. februarja 2022 s https://www.researchgate.net/publication/228646671_Learning_User_Keystroke_Patterns_for_Authentication

PRILOGE

Priloga 1: Koda in rezultati analize podatkov

```
#[1] Knjižnice
library(tidyr)
library(dplyr)
library(NbClust)
library(factoextra)
library(Hmisc)
library(ggplot2)

#[2] preberemo podatke iz csv datoteke
podatki <- read.table('C:/Podatki.csv', header = TRUE, sep= ";", dec=",")
str(podatki)
```

```
## 'data.frame':    103 obs. of  25 variables:
##  $ ID                          : int  601175745 602093026 604075007
604082217 604092930 604162527 604164815 604202532 604204512 607072421 ...
##  $ spol                        : int  0 0 1 0 0 0 0 0 1 1 ...
##  $ starost                     : int  25 29 65 67 35 34 40 28 30 31 ...
##  $ stopnja_sole                : int  3 4 2 3 3 3 3 4 4 4 ...
##  $ ima_diplomo                 : int  1 0 0 1 1 1 1 0 0 0 ...
##  $ ima_diplomo_visjo_izobrazbo: int  1 1 0 1 1 1 1 1 1 1 ...
##  $ regija                      : int  0 5 0 0 0 0 0 5 0 5 ...
##  $ iz_osrednja_slovenia        : int  0 1 0 0 0 0 0 1 0 1 ...
##  $ iz_gorenjska                : int  1 0 1 1 1 1 1 0 1 0 ...
##  $ cas_za_racunalnikom         : int  2 2 0 1 1 2 2 2 1 2 ...
##  $ vec_kot_4_ure               : int  1 1 0 0 0 1 1 1 0 1 ...
##  $ slepo_tipkanje              : int  1 1 0 1 1 1 1 0 1 1 ...
##  $ spretnejsa_roka             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ stevilo_rok                 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ znanje_anglescine           : int  4 5 3 1 3 3 4 3 3 3 ...
##  $ st_popravkov                : int  66 130 70 78 121 81 122 56 36 ...
##  $ cas                         : int  527 556 820 1605 846 512 980  ...
##  $ o_vsa_pol                   : num  1.15 1.33 0.93 0.69 0.91 1.34 ...
##  $ o_rel_pol                   : num  1.09 1.43 0.68 0.19 0.65 1.45 ...
##  $ o_manj_napak                : num  1.66 0.83 1.59 1.38 0.9 1.24  ...
##  $ o_cas_prit_tipk             : num  1.72 1.59 0 2 0 0.9 2 0.41    ...
##  $ o_vsa_pol_med               : num  0.43 0.61 0.14 0.04 0.19 0.59 ...
##  $ o_rel_pol_med               : num  0.86 1.22 0.29 0.08 0.38 1.18 ...
##  $ o_vsa_pol_brez_n            : num  0.52 0.7 0.25 0.13 0.28 0.71  ...
##  $ o_rel_pol_brez_n            : num  0.93 1.28 0.39 0.13 0.44 1.28 ...
```

```
#[3] Standardizacijo in kategorizacija spremenljivk
#standardizacija
podatki$cas_z <- scale(podatki$cas)
podatki$st_popravkov_z <- scale(podatki$st_popravkov)
podatki$o_vsa_pol_z <- scale(podatki$o_vsa_pol)
podatki$o_rel_pol_z <- scale(podatki$o_rel_pol)
podatki$o_manj_napak_z <- scale(podatki$o_manj_napak)
podatki$o_cas_prit_tipk_z   <- scale(podatki$o_cas_prit_tipk)
podatki$o_vsa_pol_med_z   <- scale(podatki$o_vsa_pol_med)
podatki$o_rel_pol_med_z   <- scale(podatki$o_rel_pol_med)
podatki$o_vsa_pol_brez_n_z   <- scale(podatki$o_vsa_pol_brez_n)
podatki$o_rel_pol_brez_n_z   <- scale(podatki$o_rel_pol_brez_n)

#kategorizacija
podatki$spol_f <- factor(podatki$spol, levels = c(0,1), labels =
c("M","Ž"))
```

```
podatki$stopnja_sole_f <- factor(podatki$stopnja_sole, levels =
c(0,1,2,3,4,5), labels= c("osnovnošolska","srednješolska", "višješolska",
"diplomski študij", "magisterij", "doktorat"))
podatki$ima_diplomo_f <- factor(podatki$ima_diplomo, levels = c(0,1),
labels = c("Ne","Da"))
podatki$ima_diplomo_visjo_izobrazbo_f <-
factor(podatki$ima_diplomo_visjo_izobrazbo, levels = c(0,1), labels =
c("Ne","Da"))
podatki$iz_osrednja_slovenia_f <- factor(podatki$iz_osrednja_slovenia,
levels = c(0,1), labels = c("Ne","Da"))
podatki$iz_gorenjska_f <- factor(podatki$iz_gorenjska, levels = c(0,1),
labels = c("Ne","Da"))
podatki$vec_kot_4_ure_f <- factor(podatki$vec_kot_4_ure, levels = c(0,1),
labels = c("Ne","Da"))
podatki$cas_za_racunalnikom_f <-
factor(podatki$cas_za_racunalnikom,levels = c(0,1,2), labels= c("Manj kot
1h na dan","Med 1h in 4h na dan","Več kot 4h na dan"))
podatki$slepo_tipkanje_f <- factor(podatki$slepo_tipkanje,levels =
c(0,1), labels= c("Da","Ne"))
podatki$spretnejsa_roka_f <-factor(podatki$spretnejsa_roka,levels =
c(0,1), labels= c("L","D"))
podatki$stevilo_rok_f <- factor(podatki$stevilo_rok, levels = c(0,1),
labels= c("Eno","Dve"))

#[4] Korelacijska analiza spremenljivk beleženja povprečnih časovnih
razmikov med pritiskom različnih kombinacij tipk pri čemer beležimo vse
kombinacije s Peasronovim koeficientom

column_combinations = c("o_vsa_pol_z"
                        ,"o_vsa_pol_med_z"
                        ,"o_vsa_pol_brez_n_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

```
##                     o_vsa_pol_z o_vsa_pol_med_z o_vsa_pol_brez_n_z
## o_vsa_pol_z                1.00            0.98                0.99
## o_vsa_pol_med_z            0.98            1.00                1.00
## o_vsa_pol_brez_n_z         0.99            1.00                1.00
##
## n= 103
##
##
## P
##                     o_vsa_pol_z o_vsa_pol_med_z o_vsa_pol_brez_n_z
## o_vsa_pol_z                     0               0
## o_vsa_pol_med_z     0                           0
## o_vsa_pol_brez_n_z  0           0
```

```
#[5]  Korelacijska analiza spremenljivk beleženja povprečnih časovnih
razmikov med pritiskom različnih kombinacij tipk  pri čemer beležimo samo
relavantne kombinacije (kombinacije tipk, ki se pojavijo v besedilu) s
Peasronovim koeficientom

column_combinations = c("o_rel_pol_z"
                        ,"o_rel_pol_med_z"
                        ,"o_rel_pol_brez_n_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

```
##                     o_rel_pol_z o_rel_pol_med_z o_rel_pol_brez_n_z
## o_rel_pol_z                1.00            0.98                0.99
## o_rel_pol_med_z            0.98            1.00                1.00
## o_rel_pol_brez_n_z         0.99            1.00                1.00
##
## n= 103
##
##
## P
##                     o_rel_pol_z o_rel_pol_med_z o_rel_pol_brez_n_z
## o_rel_pol_z                     0               0
## o_rel_pol_med_z     0                           0
## o_rel_pol_brez_n_z  0           0
```

```
#[6]  Korelacijska analiza spremenljivk beleženja povprečnih časovnih
razmikov med pritiskom različnih kombinacij tipk za vse celice in samo
relevantne celice s Peasronovim koeficientom

column_combinations = c("o_vsa_pol_z"
                        ,"o_rel_pol_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

```
##              o_vsa_pol_z o_rel_pol_z
## o_vsa_pol_z            1           1
## o_rel_pol_z            1           1
##
## n= 103
##
##
## P
##              o_vsa_pol_z o_rel_pol_z
## o_vsa_pol_z              0
## o_rel_pol_z  0
```

```
#[7]Korelacijska analiza spremenljivk beleženja števila napak uporabnika
s Peasronovim koeficientom

column_combinations = c("o_manj_napak_z"
                        ,"st_popravkov_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

```
##                 o_manj_napak_z st_popravkov_z
## o_manj_napak_z            1.00          -0.95
## st_popravkov_z           -0.95           1.00
##
## n= 103
##
##
## P
##                 o_manj_napak_z st_popravkov_z
## o_manj_napak_z                  0
## st_popravkov_z  0
```

```
#[8]Korelacijska analiza spremenljivk beleženja števila napak uporabnika
s Peasronovim koeficientom

column_combinations = c("o_rel_pol_med_z"
                        ,"cas_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

```
##                  o_rel_pol_med_z cas_z
## o_rel_pol_med_z             1.00 -0.81
## cas_z                      -0.81  1.00
##
## n= 103
##
##
## P
##                  o_rel_pol_med_z cas_z
## o_rel_pol_med_z                  0
## cas_z             0
```

```
#[9] Korelacijska analiza izbranih spremenljivk za analizo

column_combinations = c("o_rel_pol_med_z"
                        ,"o_manj_napak_z"
                        ,"o_cas_prit_tipk_z"
                        ,"cas_z")
rezultati_korelacije <- rcorr(as.matrix(podatki[,column_combinations]),
                              type = "pearson")
rezultati_korelacije
```

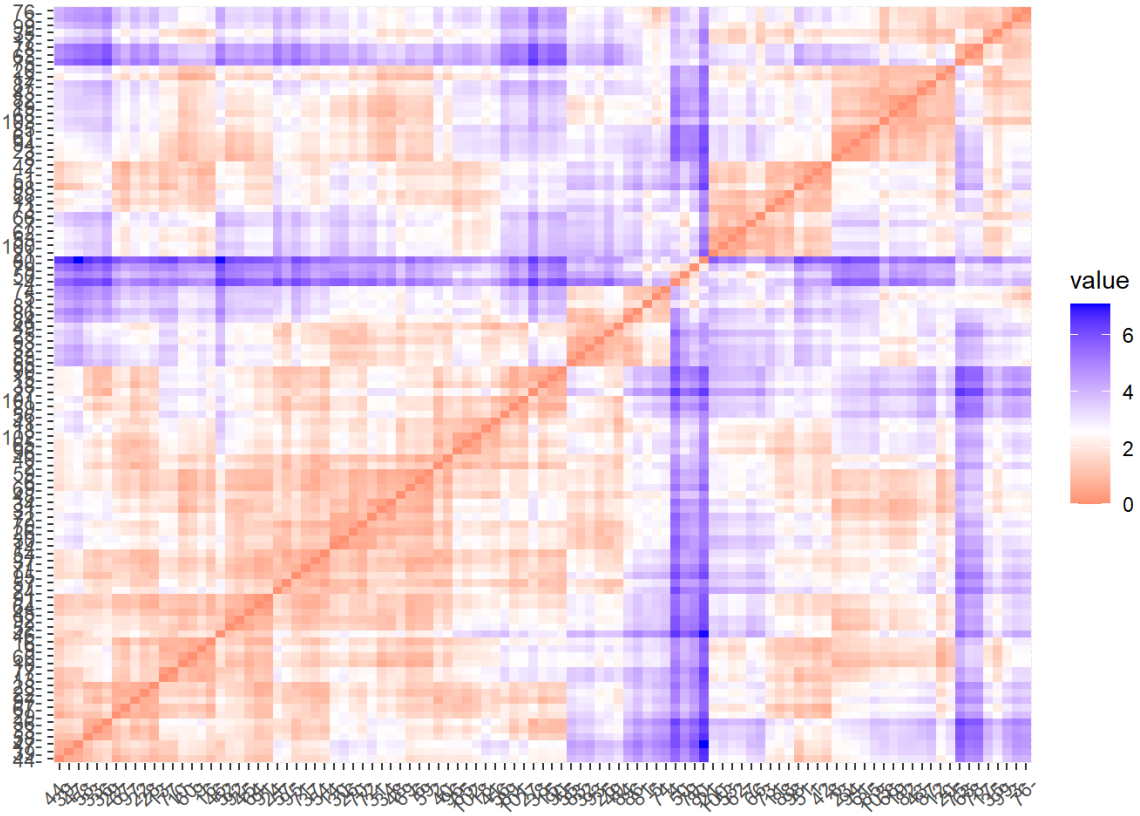
```
## A = o_rel_pol_med_z
## B = o_manj_napak_z
## C = o_cas_prit_tipk_z
## D = cas_z
##
##         A             B           C           D
## A      1.00         -0.16        0.34       -0.81
## B     -0.16          1.00        0.03       -0.04
## C      0.34          0.03        1.00       -0.10
## D     -0.81         -0.04       -0.10        1.00
##
## n= 103
##
##
## P
##         A             B           C           D
## A                     0.1086      0.0005      0.0000
## B      0.1086                     0.7810      0.6768
## C      0.0005        0.7810                   0.2968
## D      0.0000        0.6768      0.2968
```

```
#[10] Hopkinsova statistika.

column_combinations1 = c("o_rel_pol_med_z"
                        ,"o_manj_napak_z"
                        ,"o_cas_prit_tipk_z"
                        ,"cas_z")
```

```
get_clust_tendency(podatki[column_combinations1], n=nrow(podatki)-1,
graph = FALSE)
```

```
## $hopkins_stat
## [1] 0.6865057
##
## $plot
## NULL
```

```
#[11] Računanje razdalj med uporabniki in  prikaz z toplotno karto

Razdalje <- get_dist(podatki[column_combinations1], method = "euclidian")
fviz_dist(Razdalje)
```

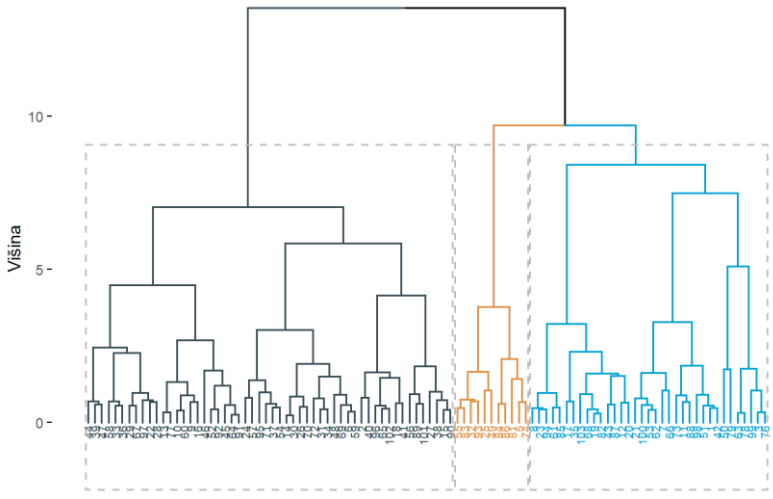



```
#[12] Računanje različnosti med uporabniki in izločitev 3 uporabnikov iz
analize

podatki$Različnost_z = sqrt(podatki$o_rel_pol_med_z^2+
                            podatki$st_popravkov_z^2+
                            podatki$o_cas_prit_tipk_z^2 +
                            podatki$cas_z^2
                              )
#stolpec postavi na začetek
podatki <- podatki[,c("Različnost_z", names(podatki)[1:(ncol(podatki)-
1)])]
head(podatki[order(-podatki$Različnost_z),],10)

podatki <-podatki[c(-80,-4, -75), ]
```

```
## A = Različnost_z
## B = ID
## C = spol
## D = starost
## E = stopnja_sole
## F = ima_diplomo
## G = ima_diplomo_visjo_izobrazbo
##
##
```

	A	B	C	D	E	F	G
80	5.108867	927205858	1	44	3	1	1
4	4.481639	604082217	0	67	3	1	1
79	3.745037	927205343	0	52	3	1	1
75	3.685649	926174105	1	51	0	0	0
50	3.490520	921140307	1	59	2	0	0
63	3.292293	923192134	1	52	3	1	1
78	3.277306	927163737	1	50	1	0	0
83	2.835124	928085848	0	29	4	0	1
32	2.793000	919081918	0	23	1	0	0

```r
#[13] Hierarhičo gručenje in dendrogram

WARD <- podatki[column_combinations1] %>%
  get_dist(method="euclidean") %>%
  hclust(method = "ward.D2")

fviz_dend(WARD,
          k=3,
          cex = 0.5,
          palette = "jama",
          color_labels_by_k = TRUE,
          rect = TRUE) +
          ggtitle("") +
          ylab("Višina")
```



```
set.seed(1)

podatki$RazvrstitevWard <-cutree(WARD, k=3)

#[14] Analiza razvrstitve z Wardovo metodo

Zac_voditelji <-aggregate(podatki[,column_combinations1]
                          ,by = list(podatki$RazvrstitevWard)
                          ,FUN = mean)
Zac_voditelji
```

```
## A = o_rel_pol_med_z
## B = o_manj_napak_z
## C = o_cas_prit_tipk_z
## D = o_cas_prit
##
            A             B             C
1  -0.7507249     0.7917266    -0.2102363
2   0.7064516    -0.1984222     0.3677730
3  -0.6110885    -1.4365532    -1.0184457
```

```
#[15] Nehierarhično gručenje voditeljev z metodo ward.D2

K_MEANS <-hkmeans(podatki[column_combinations1]
                  ,k = 3
                  ,hc.metric = "euclidean"
                  ,hc.method = "ward.D2" )

podatki$RazvrstitevK_MEANS <-K_MEANS$cluster

K_MEANS$centers
```

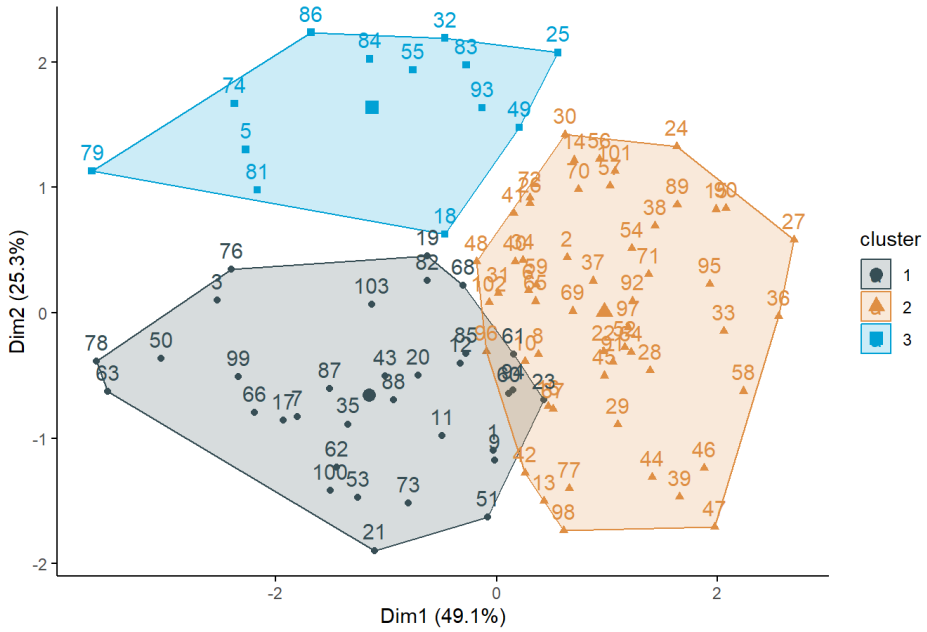
```
##   o_rel_pol_med_z o_manj_napak_z o_cas_prit_tipk_z      cas_z
## 1      -0.7635343      0.8466262       -0.2615257  0.3661073
## 2       0.7360939     -0.1619374        0.3839210 -0.5413955
## 3      -0.7231842     -1.3846537       -0.8309860  0.5369347
```

```
# [16] Primerjava razvrstve uporabnikov v skupine

cat("Razvrstitev WARD")
```

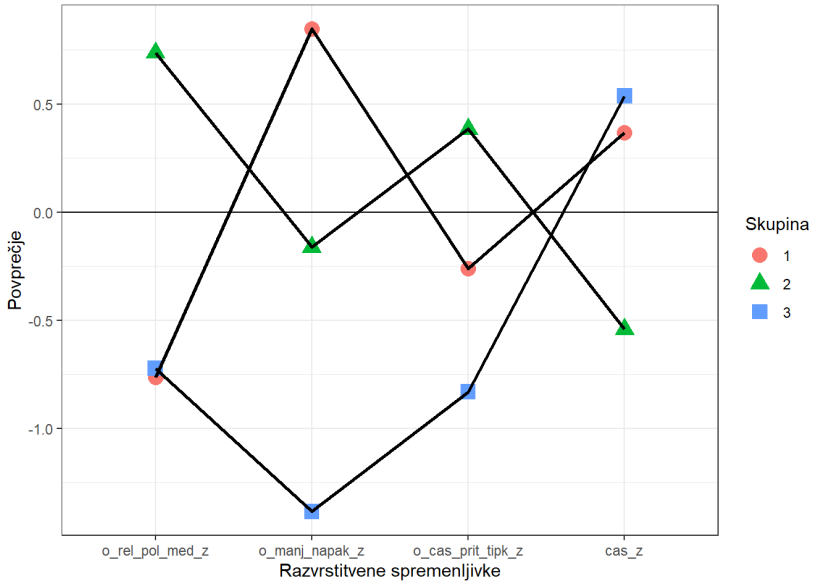
```
## Razvrstitev WARD
table(podatki$RazvrstitevWard)
##
##  1  2  3
## 35 54 11

cat("Razvrstitev K_MEANS")
## Razvrstitev K_MEANS
table(podatki$RazvrstitevK_MEANS)
##
##  1  2  3
## 33 54 13

cat("Razvrstitev po WARD in KMEANS")
## Razvrstitev po WARD in KMEANS
table(podatki$RazvrstitevWard, podatki$RazvrstitevK_MEANS)
##
```

```
##       1  2  3
##    1 31  3  1
##    2  2 51  1
##    3  0  0 11
```

```
# [17] Grafična predstavitev skupin

fviz_cluster(
  K_MEANS,
  palette = "jama",
  repel = FALSE,
  ggtheme = theme_classic()
) +
ggtitle("")
```




```
# [18] Grafična predstavitev povprečji uporabljenih spremenljivk po
skupinah

Povprečja <- K_MEANS$centers
Slika <- as.data.frame(Povprečja)
Slika$Id <- 1:nrow(Slika)
Slika <-pivot_longer(Slika, cols = c(o_rel_pol_med_z, o_manj_napak_z,
o_cas_prit_tipk_z, cas_z))
Slika$Skupina <- factor(Slika$Id, levels = c(1,2,3,4), labels =
c("1","2","3","4"))
Slika$nameFactor <- factor(Slika$name,
                           levels = column_combinations1,
                           labels = column_combinations1)


ggplot(Slika, aes(x = nameFactor, y = value)) +
  geom_hline(yintercept = 0) +
  theme_bw() +
  geom_point(aes(shape = Skupina, col = Skupina), size = 4) +
  geom_line(aes(group = Id), linewidth = 1) +
  ylab("Povprečje") +
  xlab("Razvrstitvene spremenljivke")
```




```
# [19] Povprečje starosti v posameznih skupinah in analizo variance
(ANOVA)

aggregate(podatki$starost ,
          by = list(podatki$RazvrstitevK_MEANS ),
          FUN = mean)
```

```
## Group     avg.
##  1       37.21212
##  2       27.25926
##  3       31.76923
```

```
fit <- aov(starost ~ as.factor(RazvrstitevK_MEANS), data = podatki)
summary(fit)
```

```
##                                Df Sum Sq Mean Sq F value   Pr(>F)
## as.factor(RazvrstitevK_MEANS)  2   2035  1017.6   10.98 5.03e-05 ***
## Residuals                     97   8990    92.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# [20] Povprečje ocene znanja angleščine v posameznih skupinah in analizo
variance (ANOVA)

aggregate(podatki$znanje_anglescine,
          by = list(podatki$RazvrstitevK_MEANS ),
          FUN = mean)
```

```
## Group     avg.
##   1      3.454545
##   2      4.037037
##   3      3.538462
```

```
fit <- aov(znanje_anglescine ~ as.factor(RazvrstitevK_MEANS), data =
podatki)
summary(fit)
```

```
##                               Df Sum Sq Mean Sq F value  Pr(>F)
## as.factor(RazvrstitevK_MEANS)  2   7.82   3.911   5.633 0.00485 **
## Residuals                     97  67.34   0.694
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#[21] Pearsonov test za spremenljivko spol_f

chisq <- chisq.test(podatki$spol_f,
as.factor(podatki$RazvrstitevK_MEANS))
## Warning in chisq.test(podatki$spol_f,
as.factor(podatki$RazvrstitevK_MEANS)):
## Chi-squared approximation may be incorrect
print(chisq)
##
##  Pearson's Chi-squared test
##
## data:  podatki$spol_f and as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 10.699, df = 2, p-value = 0.004751

#Kontingenčna tabela - empirične frekvence za vsako kategorijo
addmargins(chisq$observed)
##
## podatki$spol_f   1   2   3 Sum
##            M    15  41  11  67
##            Ž    18  13   2  33
##            Sum  33  54  13 100

#pričakovana frekvenca
addmargins(round(chisq$expected, 2))
##
## podatki$spol_f     1     2     3 Sum
##            M   22.11 36.18  8.71  67
##            Ž   10.89 17.82  4.29  33
##            Sum 33.00 54.00 13.00 100

#standardizirani ostanki (če je - in + potem je razlika)
round(chisq$res,2)
##
## podatki$spol_f     1     2     3
##              M -1.51  0.80  0.78
##              Ž  2.15 -1.14 -1.11

#[22] Fisherjev neparametrični preizkus za spremenljivko spol_f
fisher.test(podatki$spol_f, as.factor(podatki$RazvrstitevK_MEANS))
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$spol_f and as.factor(podatki$RazvrstitevK_MEANS)
## p-value = 0.005341
## alternative hypothesis: two.sided

#[23] Pearsonov test za spremenljivko cas_za_racunalnikom_f
```

```
chisq <- chisq.test(podatki$cas_za_racunalnikom_f ,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
## Warning in chisq.test(podatki$cas_za_racunalnikom_f,
## as.factor(podatki$RazvrstitevK_MEANS)): Chi-squared approximation may
be
## incorrect
```

```
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$cas_za_racunalnikom_f and
as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 10.01, df = 4, p-value = 0.04025
```

```
#[24] Fisherjev neparametrični preizkus za spremenljivko
cas_za_racunalnikom_f
fisher.test(podatki$cas_za_racunalnikom_f ,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$cas_za_racunalnikom_f and
as.factor(podatki$RazvrstitevK_MEANS)
## p-value = 0.05126
## alternative hypothesis: two.sided
```

```
#[25] Pearsonov test za spremenljivko vec_kot_4_ure_f
chisq <- chisq.test(podatki$vec_kot_4_ure_f ,
as.factor(podatki$RazvrstitevK_MEANS))
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$vec_kot_4_ure_f and
as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 3.4465, df = 2, p-value = 0.1785
```

```
prop_table <- chisq$observed / sum(chisq$observed)
prop_table
```

```
##
## podatki$vec_kot_4_ure_f    1    2    3
##                      Ne 0.15 0.15 0.06
##                      Da 0.18 0.39 0.07
```

```
#[26] Fisherjev neparametrični preizkus za spremenljivko vec_kot_4_ure_f
fisher.test(podatki$vec_kot_4_ure_f ,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$vec_kot_4_ure_f and
as.factor(podatki$RazvrstitevK_MEANS)
```

```
## p-value = 0.1668
## alternative hypothesis: two.sided
```

```
#[27] Pearsonov test za spremenljivko stopnja_sole_f

chisq <- chisq.test(podatki$stopnja_sole_f,
as.factor(podatki$RazvrstitevK_MEANS))
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$stopnja_sole_f and
as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 6.1928, df = 8, p-value = 0.6256
```

```
prop_table <- chisq$observed / sum(chisq$observed)
prop_table
```

```
##
## podatki$stopnja_sole_f    1    2    3
##       osnovnošolska    0.00 0.01 0.00
##       srednješolska    0.03 0.05 0.03
##       višješolska      0.03 0.05 0.00
##       diplomski študij 0.17 0.28 0.04
##       magisterij       0.10 0.15 0.06
```

```
#[28] Fisherjev neparametrični preizkus za spremenljivko stopnja_sole_f
fisher.test(podatki$stopnja_sole_f,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$stopnja_sole_f and
as.factor(podatki$RazvrstitevK_MEANS)
## p-value = 0.6442
## alternative hypothesis: two.sided
```

```
#[29] Pearsonov test za spremenljivko ima_diplomo_f

chisq <- chisq.test(podatki$ima_diplomo_f,
as.factor(podatki$RazvrstitevK_MEANS))
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$ima_diplomo_f and as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 1.9882, df = 2, p-value = 0.37
```

```
#[30] Fisherjev neparametrični preizkus za spremenljivko ima_diplomo_f
fisher.test(podatki$ima_diplomo_f, as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$ima_diplomo_f and as.factor(podatki$RazvrstitevK_MEANS)
```

```
## p-value = 0.422
## alternative hypothesis: two.sided
```

```
#[31] Pearsonov test za spremenljivko ima_diplomo_visjo_izobrazbo_f

chisq <- chisq.test(podatki$ima_diplomo_visjo_izobrazbo_f,
as.factor(podatki$RazvrstitevK_MEANS))
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$ima_diplomo_visjo_izobrazbo_f and
as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 0.14973, df = 2, p-value = 0.9279
```

```
#[32] Fisherjev neparametrični preizkus za spremenljivko
ima_diplomo_visjo_izobrazbo_f
fisher.test(podatki$ima_diplomo_visjo_izobrazbo_f,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$ima_diplomo_visjo_izobrazbo_f and
as.factor(podatki$RazvrstitevK_MEANS)
## p-value = 0.9405
## alternative hypothesis: two.sided
```

```
#[33] Pearsonov test za spremenljivko slepo_tipkanje_f
chisq <- chisq.test(podatki$slepo_tipkanje_f,
as.factor(podatki$RazvrstitevK_MEANS))
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$slepo_tipkanje_f and
as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 6.8018, df = 2, p-value = 0.03334
```

```
round(chisq$res,2)
```

```
##
## podatki$slepo_tipkanje_f     1     2     3
##                       Da -1.64  0.86  0.86
##                       Ne  1.31 -0.69 -0.69
```

```
prop_table <- chisq$observed / sum(chisq$observed)
prop_table
```

```
##
## podatki$slepo_tipkanje_f    1    2    3
##                       Da 0.07 0.25 0.07
##                       Ne 0.26 0.29 0.06
```

```
#[34] Fisherjev neparametrični preizkus za spremenljivko slepo_tipkanje_f
```

```
fisher.test(podatki$slepo_tipkanje_f,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$slepo_tipkanje_f and
as.factor(podatki$RazvrstitevK_MEANS)
## p-value = 0.02683
## alternative hypothesis: two.sided
```

```
#0 - pisal je slepo

#[35] Pearsonov test za spremenljivko spretnejsa_roka_f

chisq <- chisq.test(podatki$spretnejsa_roka_f,
as.factor(podatki$RazvrstitevK_MEANS))
print(chisq)
```

```
##
##  Pearson's Chi-squared test
##
## data:  podatki$spretnejsa_roka_f and
as.factor(podatki$RazvrstitevK_MEANS)
## X-squared = 7.0027, df = 2, p-value = 0.03016
```

```
#addmargins(chisq$observed)
#addmargins(round(chisq$expected, 2))
#round(chisq$res,2)

prop_table <- chisq$observed / sum(chisq$observed)
prop_table
```

```
##
## podatki$spretnejsa_roka_f    1    2    3
##                         L 0.02 0.00 0.02
##                         D 0.31 0.54 0.11
```

```
#[36] Fisherjev neparametrični preizkus za spremenljivko
spretnejsa_roka_f
fisher.test(podatki$spretnejsa_roka_f,
as.factor(podatki$RazvrstitevK_MEANS))
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  podatki$spretnejsa_roka_f and
as.factor(podatki$RazvrstitevK_MEANS)
## p-value = 0.02747
## alternative hypothesis: two.sided
```