Analiza poslovnih procesa u R razvojnom okruženju

Lenić, Monika

Undergraduate thesis / Završni rad

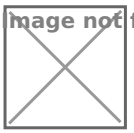
2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:137:474303

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

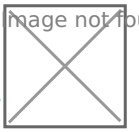
Download date / Datum preuzimanja: **2024-03-29**



Repository / Repozitorij:

Digital Repository Juraj Dobrila University of Pula





Sveučilište Jurja Dobrile u Puli

Fakultet Informatike

MONIKA LENIĆ

ANALIZA POSLOVNIH PROCESA U R RAZVOJNOM OKRUŽENJU

Završni rad

Pula, srpanj 2019. godine

Sveučilište Jurja Dobrile u Puli

Fakultet Informatike

MONIKA LENIĆ

ANALIZA POSLOVNIH PROCESA U R RAZVOJNOM OKRUŽENJU

Završni rad

JMBAG: 0303069685, redovita studentica

Studijski smjer: Informatika

Kolegij: Poslovni informacijski sustavi

Znanstveno područje: Društvene znanosti

Znanstveno polje: Informacijske i komunikacijske znanosti

Znanstvena grana: Informacijski sustavi i informatologija

Mentor: doc.dr.sc. Darko Etinger

Pula, srpanj 2019. godine

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisana Monika Lenić, kandidat za prvostupnika informatike, ovime izjavljujem da je ovaj Završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Studentica:

Monika Lenić

U Puli, srpanj 2019. godine

IZJAVA

o korištenju autorskog djela

Ja, Monika Lenić dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom „Analiza poslovnih procesa u R razvojnom okruženju“ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, srpanj 2019. godine

Potpis:

Monika Lenić

Sadržaj:

1. Uvod 1

2. R razvojno okruženje 2

2.1. Povijest R-a 3

2.2. Značajke R-a 4

2.2.1. Prednosti R-a 4

2.2.2. Nedostaci R-a 5

2.3. Paketi 6

2.3.1. Instalacija paketa 7

2.3.2. Najpoznatiji paketi 8

2.4. RStudio 10

2.5. Korisničko sučelje RStudio 11

2.6. Analiza poslovnih procesa 13

2.7. Važnost analize poslovnih procesa 13

2.8. Tehnike analize poslovnog procesa 14

2.9. Razina detaljnosti analize poslovnog procesa 16

3. Paketi za analizu poslovnih procesa u R-u 17

3.1. bupaR 17

3.2. edeaR 20

3.2.1. Metrike za istraživanje i opisivanje podataka o događajima 20

3.2.2. Filteri za podatkovne podatke 23

3.3. xesreadR 25

3.4. eventdataR 27

3.5. processmonitR 28

4. Primjeri primjene bupaR paketa za analizu poslovnih procesa u R-u 29

5. Zaključak 39

6. Popis slika i tablica 41

7. Literatura 43

Sažetak:

Današnje organizacije koriste velik raspon različitih sustava informacijske podrške kako bi podržale svoje poslovne procese. Takvi sustavi potpore bilježe obilje podataka koji sadrže različite događaje. Kako bi korisnici mogli realizirati i na što bolji način pregledati događaje, te analizirati sam poslovni proces baziraju se na današnji tradicionalni programski jezik R.

U ovom završnom radu je opisano kako se koristi RStudio korisničko sučelje, odnosno kako se analiziraju poslovni procesi pomoću RStudia uz korištenje R programskog jezika. Na početku rada je objašnjen R programski jezik te korisničko sučelje koje koristi za rad. Nadalje, u ovom radu bit će objašnjeno što su poslovni procesi, te koje tehnike postoje za analizu poslovnih procesa, kao što su: identificiranje procesa, uspostavljanje tima, i drugi. Naglasak rada je više na paketu bupaR koji se koristi za obradu i analizu podataka u R-u, te njemu slični paketi poput: edeaR i xesreadR. Osnovni cilj ovoga rada je pojasniti važnost analize poslovnih procesa, te mogućnost korištenja R programskog okruženja za vizualizaciju i prikaz analize poslovnih procesa.

Ključne riječi: poslovni procesi, RStudio, R, bupaR, analiza poslovnih procesa, edeaR, xesreadR, eventdataR, processmonitR, processmapR

Abstract:

Today's organizations employ a wide range of different information support systems to support their business processes. Such support systems record a wealth of data that contains differnet events. In order for users to realize and in a better way view the events, and analyze the business process themselves, they are based on today's traditional programming language R.

This final paper describes how to use RStudio user interface, how to analyze business processes using RStudio using the R programming language. At the beginning of this paper, the R language and user interface are explained. Furthermore, this paper will explain what business processes are, and what techniques exist for business process analysis, such as: process identification, team building, and others.

The focal point of the paper is package bupaR, it used to process and analyze data in R and similar packages like edeaR, xesreadR, will be described. The main objective of this paper is to clarify the importance of business process analysis and the possibility of using R's programming environment for visualization and presentation of business process analysis.

Keywords: buisness procesess, RStudio, R, bupaR, business process analysis, edeaR, xesreadR, eventdataR, processmonitR, processmapR

1. Uvod

Ovaj završni rad bavi se analizom poslovnih procesa u R razvojnom okruženju. Ideja i tema rada motivirana je željom da se prikažu mogućnosti programskog jezika R. Naime, R je programski jezik za statističku obradu podataka, smatra se dobro razvijenim i jednostavnim programskim jezikom te je jedan od najraširenijih svjetskih programskih jezika za statističku analizu podataka što će ovaj rad i potvrditi.

Valja također sagledati kakva je prošlost iza nas kako bismo shvatili razvojni put ove implementacije. U nekoliko zadnjih desetljeća svjedočimo iznimno brzom razvoju svih vrsta informacijskih znanosti koje donose promjene u sve aspekte ljudskog života. Povijesni razvoj R implementacije seže u davnu 1993. godinu pa sve do 2000. godine kada je izašla R verzija 1.0.0. koja se smatrala spremnom za javnost. Obzirom da bi direktan prelazak sa R razvojnog okruženja na analizu poslovnih procesa u R-u možda izgledao nejasan, bit će objašnjen i sam pojam poslovnog procesa, te važnost i tehnika analize poslovnih procesa.

Ova cijelina se može svrstati u prvi dio završnog rada, unutar koje je objašnjeno ono najosnovnije kako bi se lakše razumjelo na koji način kreirati proces pomoću programskog jezika R.

Završni rad nadalje opisuje paket bupaR koji je temelj u R programskom okruženju za izradu analize poslovnih procesa, stoga će unutar trećeg poglavlja rada biti obuhvaćeno i pojašnjeno sve u vezi bupaR-a te paketa koji ga čine, poput edeaR paketa, processmonitR paket i drugih paketa. Ovaj dio rada stoga možemo smatrati glavnim dijelom završnog rada.

Zadnja cijelina rada su dva primjera analize poslovnog procesa na temelju kojih bi svakom tko čita ovaj rad trebalo biti jasno na koji način je moguće doći do vizualizacije i analize poslovnog procesa, te kako pomoću R programskog jezika prikazati sam taj proces.

2. R razvojno okruženje

U ovom dijelu rada biti će pojašnjeno što je i čemu služi programski jezik R, tako će se dobiti uvid u osnovne karakteristike programskog jezika, također će biti navedene prednosti i nedostaci rada u R razvojnom okruženju, nešto podrobnije će biti opisani najpoznatiji paketi te njegovo korisničko sučelje.

Što možemo reći za R? R je programski jezik i softversko okruženje koji je nastao kao implementacija S programskog jezika te se koristi za statističko računanje i grafiku. Osim toga R sadrži djelotvorno rukovanje i skladištenje podataka, pakete operatora za izračune na nizovima. Smatra se dobro razvijenim i jednostavnim programskim jezikom koji uključuje uvjetovane petlje, rekurzivne funkcije te ulazne i izlazne mogućnosti. R je integrirani paket softverskih sredstava za obradu podataka koji uključuje:

- djelotvorno rukovanje i skladištenje podacima
- skup operatora za izračune na nizovima, posebno matrice
- veliku, koherentnu, integriranu zbirku posrednih alata za analizu podataka
- dobro je razvijen, jednostavan i učinkovit programski jezik

Velika prednost R okruženja je to što je besplatan programski jezik, tj. programska podrška R-a je slobodnog dohvata (eng. Open-source software). Ovakva vrsta programa omogućuje svakom istraživaču da sudjeluje u njegovom razvoju, tako je pokrivena velika paleta matematičkih i statističkih modela koji postaju dostupni krajnjim korisnicima. Danas mnoge poznate tvrtke kao što su Facebook, Google, Bing, Mozilla i mnogi drugi koriste R.

2.1. Povijest R-a

Kao što je već navedeno u prethodnom poglavlju, R je implementacija S programskog jezika kojeg je razvio John Chambers s kolegama u Bell Labaratoriesu te Robert Gentleman sa sveučilišta u Aucklandu. Početak rada na R programskom jeziku seže u davnu 1993. godinu, kada su Robert Gentleman i Ross Ihaka proizveli nešto što je izgledalo kao potencijalno koristan dio softvera te su ga odlučili koristiti u nastavničkom laboratoriju na studijima statistike. S obzirom na to da su bili izuzetno zadovoljni svojim radom odlučili su postaviti na [1]StatLib neke binarne kopije R-a. Među ljudima koji su preuzimali njihove binarne kopije i ostavljali povratne informacije, istaknuo se Martin Mächler koji im je predložio da objave izvorni kod R-a kao slobodni softver. Kako je Mächler imao odlične argumente, uspio ih je nagovoriti da 1995. godine, objave izvorni kod koji je bio tada dostupan putem [2]FTP-a pod određenim uvjetima GNU licence. Sam proces razvoja R-a tada je bio relativno „zatvoren proces“, osnivači bi dobili određene informacije o pogreškama i s vremena na vrijeme bi radili na održavanju R-a. Kako se kroz povijest R stvarno brzo razvijao i ljudi su sve više prenosili svoje aplikacije na R programski jezik, postalo je sasvim jasno da će u skoroj budućnosti biti potreban bolji mehanizam raspodjele. Tada na red dolazi Kurt Hornik iz TU Wien, koji preuzima zadatak osnivanja arhiva. Uvođenjem mailing listi razvoj R je abnormalno brzo rastao, djelomično zbog toga što je bilo sve više povratnih informacija, no djelomično i zbog zaprimanja zakrpi na kodu. Razina doprinosa nad kodom bila je toliko velika, da Robert Gentleman, Martin Mächler i Ross Ihaka nisu mogli uvijek mijenjati stopu koja je bila zadovoljavajuća za one koji traže promjene. Stoga je 1997. godine uspostavljena veća skupina koja je imala mogućnost mijenjati CVS arhivu izvornog koda. U toj grupi su sada: Doug Bates, Peter Dalgaard, Robert Gentleman, Kurt Hornik, Ross Ihaka, Friedrich Leisch, Thomas Lumley, Martin Mächler, Paul Murrell, Heiner Schwarte te Luke Tierney, tada nazvan, R-Core tim. Hornik i Leisch su 1997. godine dali Herkulov doprinos R projektu kada su izgradili [3]CRAN, mrežu koja je omogućila preuzimanje bitnih informacija i datoteka R-a na jednom mjestu. R funkcionalnosti su sadržane uglavnom u paketima, koji su pohranjeni na CRAN-u, zbog tih paketa R je zapravo postao jedan od svestranijih statističkih softvera. Paketi su isključivo korisni kako

[1] Arhiva za statističke softvere, skupove podataka i informacije
[2] Standardni mrežni protokol koji se koristi za premještanje datoteka s jednog hosta na drugi putem mreže temeljene na TCP-u, kao što je Internet
[3] Mrežna arhiva paketa za R programski jezik

korisnici ne bi trošili vrijeme na pisanje tisuća i tisuća redaka koda, već bi samo pozvali određeni paket i riješili probleme.

2000. godine R Project je izdao R verziju 1.0.0. te je to bila prva verzija koja se smatrala spremnom za javnost.

2.2. Značajke R-a

R je jedan od najraširenijih svjetskih programskih jezika za programiranje. Uz mnoge prednosti, R ima i svojih nedostataka. U nastavku će biti navedene prednosti i nedostaci korištenja R programskog jezika.

2.2.1. Prednosti R-a

Za početak, glavna prednost R programskog jezika je njegova dostupnost, naime R je dostupan kao slobodan softver pod uvjetima FSF GNU opće javne licence u obliku izvornog koda. Što znači da dozvoljava:

a) slobodu uporabe programa u bilo koje svrhe
b) slobodu proučavanja načina rada programa i mogućnost prilagodbe vlastitim potrebama
c) slobodu kopiranja i distribucije programa
d) slobodu poboljšavanja programa
e) mogućnost dijeljenja poboljšanog programa u zajednici

Osim tih činjenica, važno je naglasiti da je R dostupan putem interneta na raznim UNIX platformama i drugim sličnim sustavima. To je jezik koji je neovisan o platformi te se može koristiti bilo kada i bilo gdje. Prema dokumentu Ross Ihake jedna od najvećih prednosti i zadovoljstava u radu s R je: „...jedna od najboljih stvari u radu na R je prilika za rad s tako velikom skupinom naroda“ (Ihaka, str. 4).

Neke od glavnih prednosti R-a u odnosu na druge programske jezike su:

a) proizvod je međunarodne suradnje vrhunskih statističara i dizajnera programskih jezika
b) dozvoljava statističke analize i vizualizaciju te njihovo neograničeno unapređivanje

c) svaki korak u analizi je sačuvan, pa se takva povijest (datoteka .Rhistory) može naknadno koristiti u novim analizama ili kao dokumentacija postojećoj
d) zbog svoje organizacije potiče kritičko razmišljanje tijekom provedbe analize
e) u potpunosti je programabilan, a u podlozi sadrži vrlo sofisticirani jezik S
f) programski kod svake funkcije je javan, pa je vidljiv svaki korišteni algoritam.

2.2.2. Nedostaci R-a

Iako se u prethodnom poglavlju čini da je R programski jezik nadomak savršenstva, potrebno je naglasiti da ipak postoji i dovoljno nedostataka koji ga udaljavaju od savršenog programskog jezika. Naime, većina ljudi navodi kako je najveći nedostatak ovog programskog jezika njegovo održavanje, tj. teško je održavati kod koji kontinuirano nadograđujete.

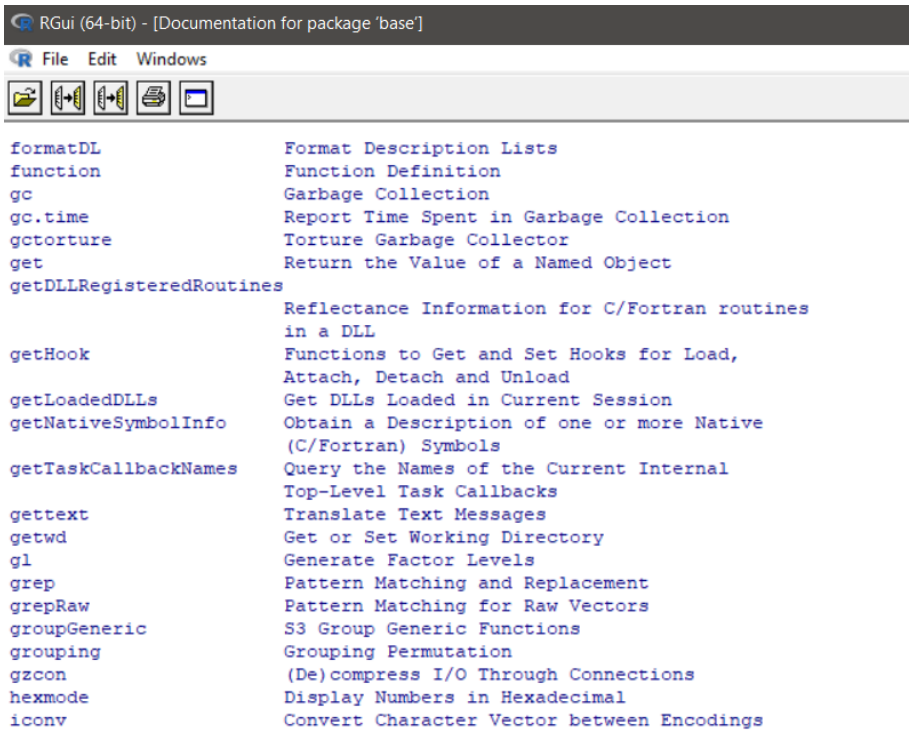
Neki od nedostataka su:

a) R nije jednostavan za početnike
b) temelji se na 40 godina staroj tehnologiji
c) postoji slaba podrška analizama podataka putem grafičkih korisničkih sučelja
d) R ne može ispravno skalirati s velikim skupovima podataka
e) nema konzistentnu sintaksu
f) jako loša dokumentacija, dokumentacija algoritama je nepotpuna, te nema prikazan primjer kako riješiti kanonski oblik ili kako koristiti određenu implementaciju
g) algoritmi se razlikuju u složenosti na temelju njihove implementacije i opisa
h) ovisi o memoriji te može potrošiti svu dostupnu memoriju

2.3. Paketi

Funkcionalnosti koje pruža R proširene su pomoću određenih paketa. Pakete uglavnom razvijaju korisnici koji razvojem paketa omogućuju uporabu grafičkih uređaja, uvoz/izvoz podataka i slično. Na internetu se može pronaći nešto više od 6400 dodatnih paketa unutar arhive sustava R (kako bilježe podaci iz 2015. godine). Samom instalacijom sustava R dobivaju se osnovni paketi koji podržavaju programski jezik S, osnovnu grafiku i osnovnu sintaksu. Neki od tih paketa su: base, compiler, datasets i dr. Drugi skup paketa koji je također dio same instalacije sustava je skup paketa koji imaju mogućnost nadograđivanja između verzija i zbog toga ih se može pronaći na CRAN-u, što nije u slučaju osnovnih paketa. Neki od paketa ove skupine su: boot, MASS, matrix i sl.

Paket koji se ističe je base, to je paket koji čini osnovu sustava R. Paket base sadržava osnovne funkcije poput aritmetike i osnovne programske potpore. Ako korisnik želi pogledati sve funkcije koje pruža paket base može upisati: library(help = "base") i dobit će popis svih funkcija koje base može raditi. S obzirom na to da je lista funkcija za paket base iznimno duga, na slici 2 je prikazan dio funkcija koje se dobivaju pozivanjem naredbe za ispis liste svih funkcija paketa base.



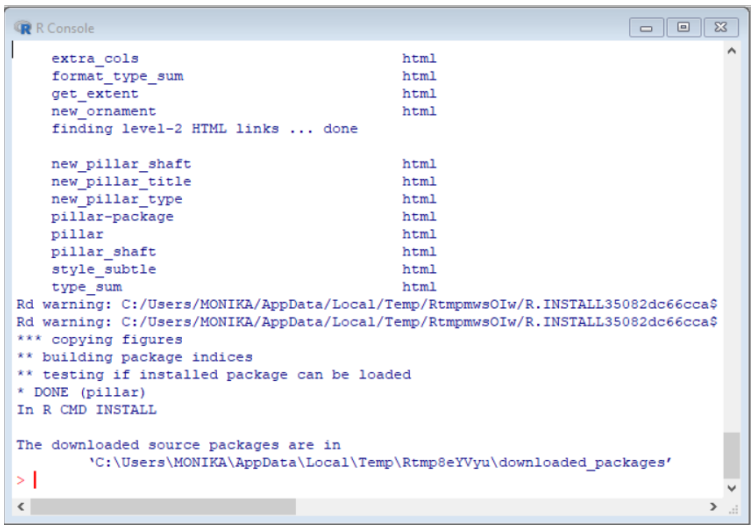
Slika 1 Dokumentacija za paket base

Izvor slike: Vlastiti rad

2.3.1. *Instalacija paketa*

Mogućnosti sustava R mogu se nadograditi instalacijom dodatnih paketa te postavljanjem njihovih funkcionalnosti u sustav. Postoje dvije metode za instalaciju paketa u R okruženju.

***Prva metoda*:** nakon pokretanja R u izborniku odabire se metoda „Packages“ te se u padajućem izborniku kod paketa može odabrati nekoliko funkcionalnosti, jedna od funkcionalnosti je „Install package(s)...“, tada sustav daje mogućnost odabira „Secure CRAN mirrors“ u kojem valja odabrati ogledalo koje je najbliže našem zemljopisnom položaju. Kada je odabrano sve potrebno sustav daje mogućnost odabira paketa koji treba instalirati. U slučaju da korisnik želi odjednom instalirati više od jednog paketa, ima mogućnost kliknuti na svaki paket koji želi dok pri odabiru drži tipku CTRL. Ako je sve uspješno napravljeno na kraju instalacije će se pojaviti znak „>“, kao što je prikazano na slici 3.



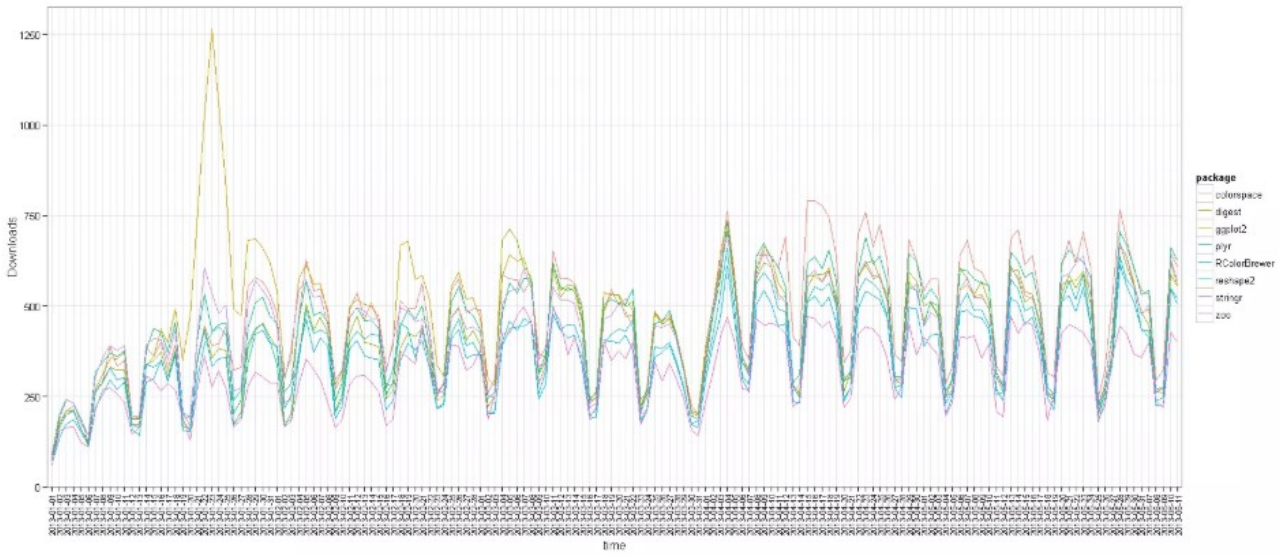
Slika 2 Uspješna instalacija paketa "ggplot2"

Izvor slike: Vlastiti rad

Druga metoda: puno je jednostavnija od prethodne. Potrebno je u konzolu upisati naredbu za instalaciju paketa „install.packages("ggplots2") te se pritiskom na tipku enter paket instalira.

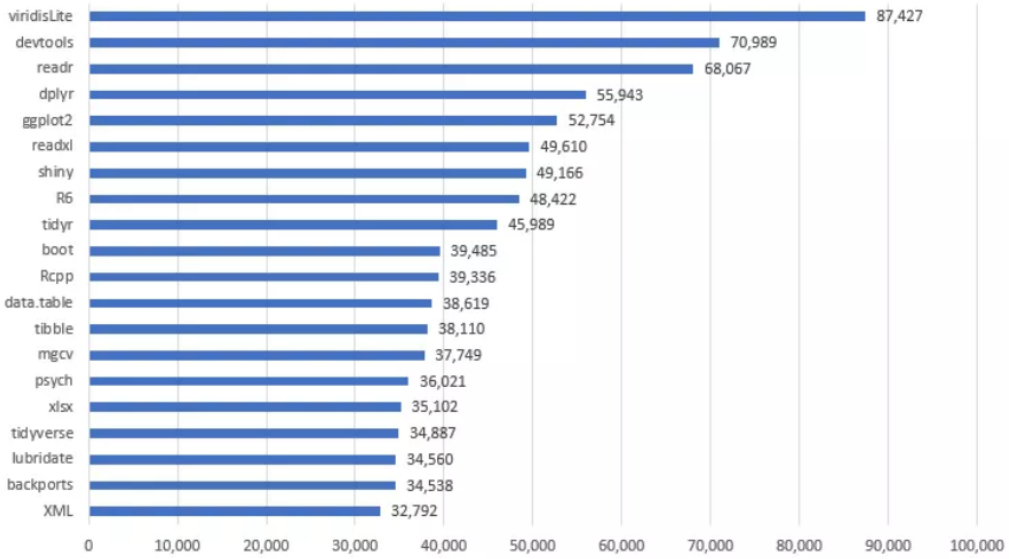
2.3.2. Najpoznatiji paketi

Nakon brojnih istraživanja i želje za saznanjem koji su R paketi najviše preuzimani s CRAN-a, zaključak je očekivan. Prikupljeni rezultati iz 2013. godine uspoređeni su sa rezultatima iz 2017. godine. Na temelju prikupljenih rezultata zaključeno je da je u 2013. godini od mjeseca siječnja do svibnja na bazi dnevnog preuzimanja paketa najviše preuziman plyr, digest, ggplot2 te colorspace. Rezultati prikupljanja ovih podataka jasno su vidljivi na dijagramu prikazanom na slici slici 3.



Slika 3 Statistika preuzimanja R paketa 2013. godine

Izvor slike: https://www.r-statistics.com/2013/06/top-100-r-packages-for-2013-jan-may/



Slika 4 Statistika preuzimanja paketa 2017. godine

Izvor slike: http://makemeanalyst.com/20-most-popular-r-packages/

Uspoređujući sa 2017. godinom na slici 4 da se zaključiti kako se na prva četiri mjesta po broju preuzimanja i dalje nalazi ggplot2. Uzevši u obzir popularnost ovog paketa, te podrobnijim istraživanjem da se zaključiti kako je njegova današnja popularnost i dalje velika, točnije i danas je na vrhu ljestvice preuzimanih paketa.

Najpreuzimaniji paketi i njihove karakteristike:

a) plyr – skup alata koji rješava zajednički niz problema
b) digest – stvaranje kriptografske skripte s R objektima
c) ggplot2 – omogućuje stvaranje grafova koji predstavljaju jednolične i multivarijatne numeričke i kategorijske podatke na jednostavan način
d) colorspace – paket koji omogućuje mapiranje između različitih boja, kao što su RGB, HSV, HCL i druge

Na slici pet se nalazi primjer korištenja ggplot2 paketa.

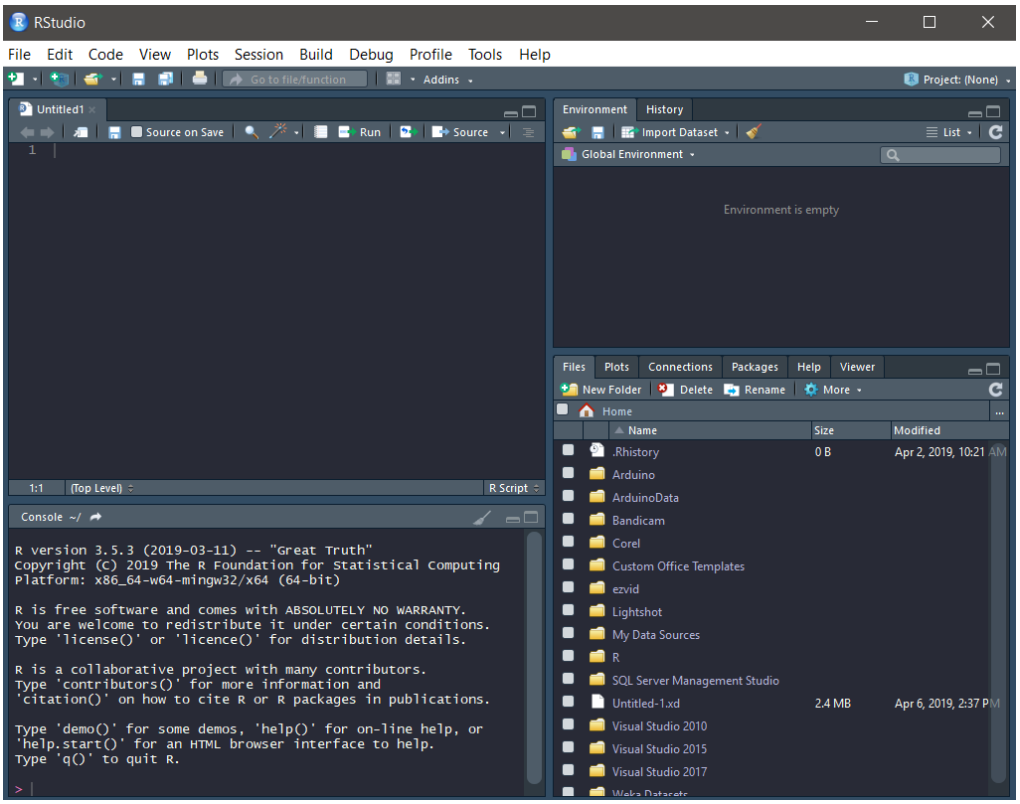


Slika 5 Area vs Population - ggplot2

Izvor slike: http://r-statistics.co/Complete-Ggplot2-Tutorial-Part1-With-R-Code.html

2.4. RStudio

RStudio je integrirano razvojno okruženje koje omogućuje lakšu interakciju s R-om. On je poput R-a besplatan, a programska podrška mu je slobodnog dohvata. Svoju povijest započeo 2011. godine kada je službeno objavljena beta verzija, te je danas jedan od popularnih razvojnih okruženja za statističke analize podataka. Pri pokretanju RStudia korisnik može uočiti tri dostupna prozora, četvrti prozor je skriven, no može se pokrenuti tako što korisnik ode na File -> New File -> R script. Kada korisnik pokrene i četvrti prozor dobiva pregled okruženja kao što je na slici šest.



Slika 6 RStudio okruženje

Izvor slike: Vlastiti rad

Također, vrijedi naglasiti da je RStudio dostupan u dva izdanja:

- ***PRVO IZDANJE:*** RStudio za radnu površinu, program se pokreće lokalno, kao jednostavna desktop aplikacija.
- ***DRUGO IZDANJE:*** RStudio Server, je Linux poslužiteljska aplikacija koja pruža sučelje temeljeno na web pregledniku za R verziju koja se izvodi na poslužitelju.

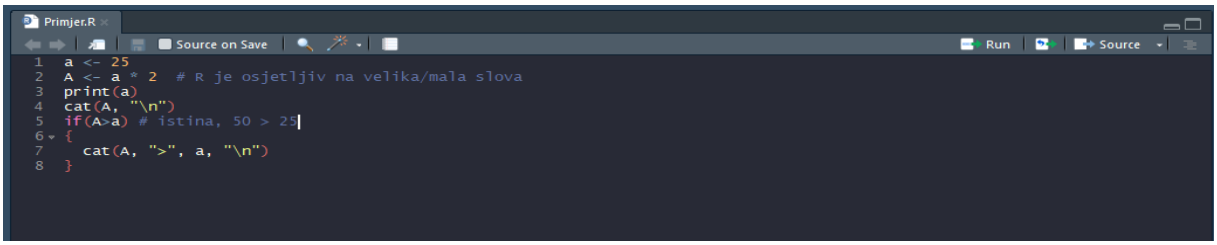
2.5. Korisničko sučelje RStudio

Obzirom da je RStudio iznimno dobar za obradu podataka i aktivnosti, primjeri ovoga rada bit će kreirani unutar njegovog sučelja. Stoga je korisničko sučelje iznimno jednostavno za korištenje. Kao što je navedeno u poglavlju RStudio sučelje se sastoji od četiri glavna prozora.

Korisničko sučelje se sastoji od:

1) R Source

Prozor koji služi za kreiranje, izvođenje i spremanje skripti koje se sastoje od niza naredbi koje se izvršavaju u nizu kao paket, ovaj prozor se također koristi za pregled skupova podataka. Prilikom pisanja složenijih kodova iznimno je dobro koristiti R Source jer omogućuje bolju preglednost zbog numeriranja redova. Također, bitno je naglasiti da R ima izuzetnu osjetljivost na velika i mala slova, kao što vidimo u ovom primjeru na slici sedam, malo slovo a i veliko slovo A predstavljaju različite vrijednosti. Napisani kod u source pokreće se opcijom Run, a rezultat je vidljiv u konzoli.



Slika 7 R Source prozor

Izvor slike: Vlastiti rad

2) Konzola

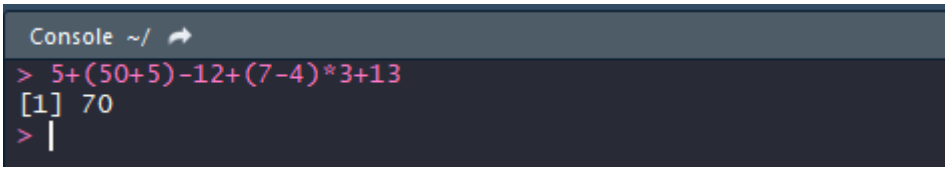
Konzola je prozor u koji se unose i izvršavaju pojedinačne naredbe, ali služi i za prikaz rezultata. Kod pokretanja skripte iz prethodnog djela, unutar konzole će biti dobiven rezultat. Na slici osam je prikazan ispis rezultata, koji je prethodno bio pokrenut unutar skripte.



Slika 8 Primjer ispisa rezultata u konzoli

Izvor slike: Vlastiti rad

Također, unutar konzole se mogu računati jednostavni matematički izrazi kao npr. $5 + (50 + 5) - 12 + (7 - 4) * 3 + 13$, rješenje se dobiva unosom ovoga primjera u konzolu.



Slika 9 Rješavanje zadataka u konzoli

Izvor slike: Vlastiti rad

Iz ovoga primjera se vidi kako program jednostavno računa matematičke operacije poput zbrajanja, oduzimanja, množenja te grupiranja brojeva. Isto vrijedi i za sve ostale elementarne matematičke funkcije, kao što su dijeljenje, trigonometrijske funkcije i slično.

3) Environment, History

Odabirom kartice „Environment“ mogu se pregledati skupovi podataka i trenutni objekti. Kod kartice „History“ je logično da korisnik može vidjeti koje je naredbe prethodno odradio.

4) Dokumenti

Ovaj prozor prikazuje sve dokumente, mape kao i grafičke prikaze te dostupne pakete na radnoj površini. Također, u ovom prozoru korisnik može pronaći dio za pomoć s detaljnom dokumentacijom gdje se može istražiti nejasne dijelove programa.

2.6. Analiza poslovnih procesa

Poslovni proces je strukturiran, analitičan skup aktivnosti koji zahtjeva neprestano poboljšavanje. „Riječ je o aktivnostima s jasno utvrđenim početkom i završetkom, tijekom kojih se u više ili manje stalnim intervalima stvara vrijednost za potrošače“ (Bosilj Vukšić, Hernaus i Kovačić, 2007., str. 17).

Za razliku od same definicije poslovnog procesa, važno je naglasiti što je zapravo analiza poslovnog procesa. Analiza poslovnog procesa je metoda analize koja pomaže povećati učinkovitost i djelotvornost procesa. Analiza poslovnih procesa identificira i ispituje svaki dio strukture uključujući sam proces, korisnike koji sudjeluju, razmjenu informacija i mnoge druge.

Analiza poslovnih procesa uključuje:

a) ***Dokumentiranje procesa*** – razumijevanje kako posao teče kroz procese
b) ***Pridruživanje vlasnika*** – uspostavljanje menadžerske odgovornosti
c) ***Upravljanje procesima*** – optimizacija performansi procesa
d) ***Poboljšanje procesa*** – povećanje kvalitete proizvoda ili mjerenih pokazatelja

2.7. Važnost analize poslovnih procesa

Danas postoje mnoge provjerene metodologije i alati koji se koriste za analizu organizacijske strukture poslovnih procesa, poslovnih funkcija te komunikacijskih oblika organizacije, neki od poznatijih alata za analizu poslovnih procesa su: Appian, Kissflow i mnogi drugi, također jedna od poznatih metodologija koja se koristi za analizu poslovnih procesa radi provođenja kontinuiranog unaprijeđenja je DMAIC metodologija. Neovisno koju metodologiju ili alat koristimo, treba se naglasiti da je potrebno analizirati poslovne procese kako bi se osigurala potpora ključnim poslovnim ciljevima. Danas svako poduzeće mora neprestano vrednovati svoje procese i razmišljati o tome kako ih poboljšati. Vremenom postoje sve novije tehnologije koje as-is proces čine zastarjelim ili lakim za poboljšanje, te je iz toga razloga izuzetno bitna analiza poslovnih procesa. „Analiza poslovnih procesa posebice je korisna za ostvarivanje ciljeva vezanih za potrošače, učinkovitost i za profitabilnost“ (Bosilj Vukšić, Hernaus i Kovačić, 2007., str. 84).

Analizu poslovnih procesa je dobro koristiti u slučajevima kada:

a) Sudionici procesa nejasno izražavaju na koji način treba provesti proces
b) Tim želi zamijeniti proces s novom verzijom
c) Prije uvođenja automatizacije kako bi se osiguralo da je proces optimiziran

2.8. Tehnike analize poslovnog procesa

Ako želimo kontinuirano poboljšanje kvalitete poslovnog procesa potrebno je držati se nekoliko koraka, a to su:

A) Identificiranje procesa

Za početak je potrebno utvrditi koji se procesi trebaju poboljšati, ovo je jedna bitna stavka koju je potrebno dobro naučiti i razumjeti, važno je uvijek razmišljati o krajnjem cilju poslovanja i koji procesi doprinose tom cilju. No, uvijek postoji pitanje: „Kako prepoznati najvažnije procese?“, to su procesi koji najviše doprinose postizanju organizacijskih ciljeva i dodaju veću vrijednost konačnoj realizaciji procesa. S obzirom na to da je Peter Drucker predložio metodologiju poslovne analize koja može pomoći u razumijevanju strateških ciljeva tvrtke, a potom i definirati koji su procesi potrebni. Njegova metodologija nalaže pet koraka analize poslovnih procesa, a to su:

1) Što je naša misija?

Ako kao poslodavac ne razumijemo razlog postojanja tvrtke, nećemo biti u mogućnosti odrediti koji su procesi temeljni za ispunjavanje našeg cilja, procesi su promjenjivi no cilj tvrtke traje i mora se postići kroz određene poslovne procese.

2) Tko je naš klijent?

Važno je identificirati procese koji našim kupcima (od svih vrsta kupaca) dodaju najviše percipirane vrijednosti, tako da je bitno u popis procesa uključiti i ovu stavku.

3) Koja je vrijednost kupca?

Jedan od načina utvrđivanja vrijednosti kupaca je temeljito istraživanje tržišta. Samo precizno prikupljanje informacija može dati prave odgovore na ovo pitanje.

4) Koji je naš ključni rezultat?

Četvrta faza analize poslovnih procesa govori o ključnim pokazateljima uspješnosti, potrebno je provjeriti koji procesi ne ispunjavaju navedene ciljeve i zašto.

5) Koji je naš plan?

Posljednja faza naglašava koje je aktivnosti organizacija definirala kako bi ostvarila svoje strateške ciljeve? Svi procesi koji su povezani s tim radnjama ispunit će popis.

B) Uspostavljanje tima

Bitno je uspostaviti tim koji će pomoći u analizi poslovnih procesa, najbolji ljudi su oni koji već svakodnevno rade s procesom. Kada je u pitanju sastavljanje tima, bitno je razmotriti koji će stručnjaci znati učinkovito odraditi svoj zadatak u analizi poslovnih procesa.

C) Stvaranje dijagrama (flowchart)

Tehnika analize procesa jako je korisna i može pružiti uvid u to kako se proces odvija. Standardnim simbolima i alatima može predstaviti proces na jasan i praktičan način.

D) Definiranje AS-IS procesa

Nakon što je kreiran dijagram, potrebno je definirati kako će se proces odvijati sada, važno je napredovati i držati se stvarne situacije. To znači da je potrebno razumjeti kako se proces odvija u trenutnom vremenu, a ne ga zamišljati na način koji bi ste htjeli. U ovom koraku da bi ste ispravno pratili odvijanje procesa potrebno je:

- Razgovarati s akterima – predstavljaju aktivnosti procesa, njegov redoslijed izvršavanja, tko je odgovoran
- Analizirati model procesa – saznati svrhu procesa, postoji li interakcija s klijentima, koja se poslovna pravila primjenjuju

E) Točke poboljšanja

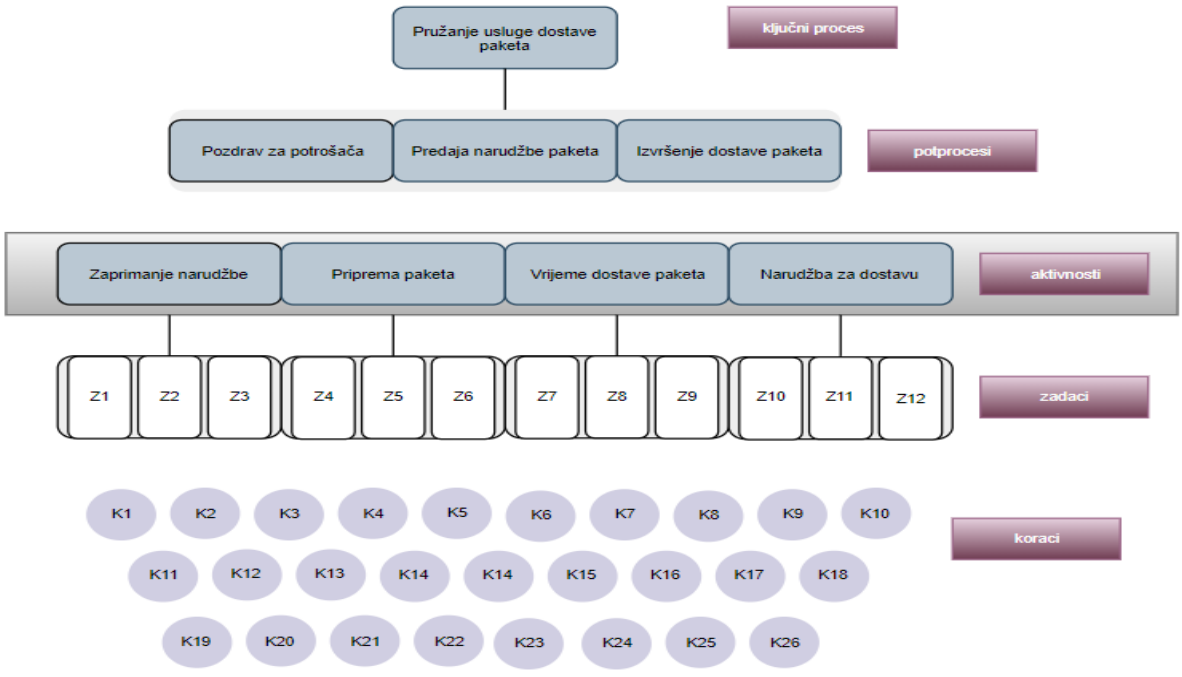
Sada je važno odrediti koja su poboljšanja poželjna, pri tome je bitno voditi računa o strateškim ciljevima tvrtke, budući da ih svi procesi i akcije moraju imati kao ciljeve.

F) Model TO BE procesa

Posljednji korak se sastoji od osmišljavanja novog poboljšanog procesa, koji će učinkovitije i djelotvornije ostvariti cilj organizacije.

2.9. Razina detaljnosti analize poslovnog procesa

Neke organizacije pojedini poslovni proces mogu analizirati do najsitnijih detalja, do njegovim najmanjih gradivnih blokova. Ako organizacija želi takav pristup, onda je potrebno puno više razumijevanja i vremena, a troškovi provođenja same analize su puno veći. „Sam je odabir razine detaljnosti arbitraran, odnosno ovisi o svrsi analize poslovnih procesa u konkretnom slučaju, ali i o raspoloživom vremenu“ (Bosilj Vukšić, Hernaus i Kovačić, 2008., str. 86). Dekompoziciju poslovnog procesa treba prekinuti kada je visoka razina detaljnosti više nije korisna za analizu poslovnog procesa, točnije najčešće je to na razini aktivnosti, jer ako ulazimo u veće analize idemo prema pojedinostima koje donose više štete, veće troškove već neku značajnu korist. Tako gledajući s razine kvantitete, potrebno je stopostotno znanje o poslovnom procesu, no za njegovo modeliranje potrebno je tek 20% ukupnog znanja. Problem modeliranja poslovnog procesa prikazan je na skici koja prikazuje dostavu paketa.



Slika 10 Model poslovnog procesa na različitim razinama detaljnosti

Izvor slike: Vlastiti rad

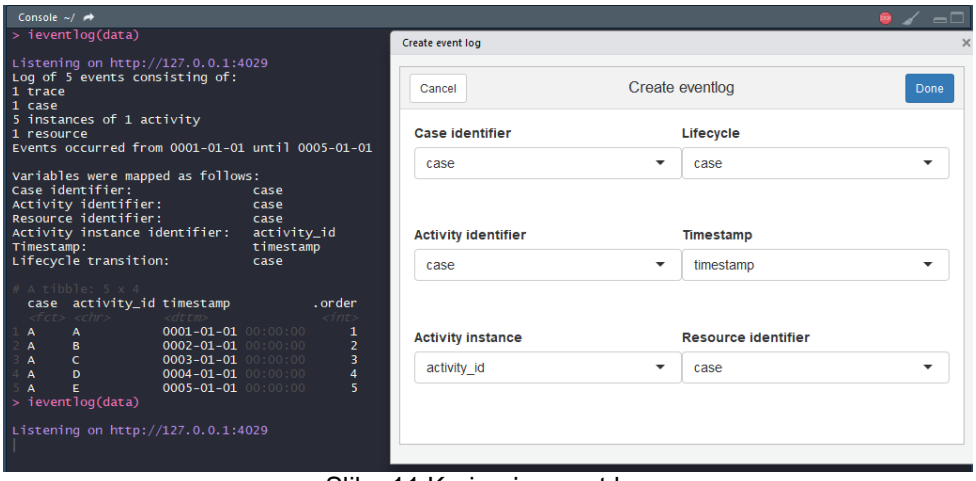
3. Paketi za analizu poslovnih procesa u R-u

bupaR (Buisness Process Analysis with R) se sastoji od različitih R paketa od kojih svaki ima svoju određenu svrhu. Paket bupaR predstavlja središnje srce paketa te pruža osnovnu funkcionalnost za obradu podataka o događajima koje koriste drugi paketi. U sljedećim paragrafima su ukratko predstavljeni ostali paketi:

- edeaR – za istraživačku i desktiptivnu analizu podataka o događajima
- xesreadR – za čitanje i pisanje xes datoteka
- processmapR – za kreiranje vizualizacije procesa
- processmonitR – za stvaranje nadzornih ploča procesa
- eventdataR – spremište podataka, sadrži stvarne i umjetne događaje

3.1. bupaR

bupaR je open-source paket za obradu i analizu podataka o poslovnim procesima u R razvojnom okruženju. Razvijen je od strane istraživačke grupe Business Informatics sa Sveučilišta Hasselt u Belgiji. Prema Gert-u „Centralni paket bupaR uključuje osnovne funkcionalnosti za kreiranje eventlog-objects u R“ (Janssenswillen, str. 2). Sadrži nekoliko funkcija za dobivanje informacija o zapisniku događaja i također pruža specifične verzije dnevnika događaja generičkih funkcija R. bupaR ima cilj podržati svaki korak u analizi podataka događaja s programskim jezikom R, od uvoza podataka do online praćenja procesa. U nastavku ću prikazati kako kreirati objekte zapisnika događaja.



Slika 11 Kreiranje event log

Izvor slike: Vlastiti rad

Funkcija koja je korištena u ovom primjeru, kao na slici 11 je ieventlog. Slovo „i“ predstavlja oznaku kojom se prikazuje da će se otvoriti korisničko sučelje (interface) kao što je prikazano.

Argumenti koje sam koristila za ovaj prikaz su:

- ieventdata (data) - data predstavlja podatkovni objekt koji će se koristiti kao zapisnik događaja, može biti npr. tablica
- case identifier - predstavlja klasifikator slučaja dnevnika, to je vektor znakova koji sadrži nazive varijabli duljine 1 ili više
- lifecycle – predstavlja životni ciklus zapisnika (event log)
- activity identifier – ovaj argument zapravo predstavlja isto što i case identifier, vektor znakova koji sadrži nazive varijabli duljine 1 ili više
- activity instance identifier – klasifikator instance aktivnosti kod event log
- timestamp – predstavlja vremensku oznaku, odnosi se na Date ili [4]POSIXct
- resource identifier – isto kao i activity identifier i case identifier, vektor znakova koji sadrži nazive varijabli duljine 1 ili više

Svaki događaj se odnosi na jedan koncept aktivnosti. Ako uzmemo za primjer aktivnosti mogu biti: prijava, provjera te mnoge druge. Kada se aktivnost provodi to znači da se kreira instanca aktivnosti, stoga možemo navesti da ako jedna osoba prolazi specifične radnje u određenom trenutku u nekom vremenu tada se to naziva aktivnost. Obzirom da je događaj povezan s instancom aktivnosti, to znači da sadrži samo jednu vremensku oznaku, osim toga svaki događaj bi trebao sadržavati prijelaz na životni ciklus. Točnije, višestruki događaji mogu opisati različite prijelaze životnog ciklusa jedne instance aktivnosti. Naprimjer, jedan događaj može zabilježiti kada je određeni korisnik prijavljen, drugi kada je je korisnik obavio svoj posao i odjavio se i tako dalje. Da bi se mogli povezati događaji koji pripadaju istoj instanci aktivnosti važno je odrediti identifikator instance aktivnosti, tako uzmemo za primjer da je korisnik odradio različite poslove, svaki posao je imao zakazan, započet i završen događaj. Identifikator aktivosti će tada omogućiti da se razlikuju koji događaji pripadaju zajedno i koji ne.

[4] Predstavlja reprezentaciju datuma u kalendaru te vremena

Na slici 12 se nalazi primjer pacijenta te njegove aktivnosti, vremenska oznaka, identifikator aktivnosti i status.

pacijent	aktivnost	vremenska oznaka	status	activity_instance
John Doe	prijava	2017-05-10 08:33:26	potpun	1
John Doe	kirurgija	2017-05-10 08:38:21	raspored	2
John Doe	kirurgija	2017-05-10 08:53:16	početak	2
John Doe	kirurgija	2017-05-10 09:25:19	potpun	2
John Doe	liječenje	2017-05-10 10:01:25	početak	3
John Doe	liječenje	2017-05-10 10:35:18	potpun	3
John Doe	kirurgija	2017-05-10 10:41:35	početak	4
John Doe	kirurgija	2017-05-10 11:05:56	potpun	4

Slika 12 Kreiranje pojedinosti za zapisnik događaja

Izvor slike: https://www.bupar.net/creating_eventlogs.html

U konačnici svaki redak podataka treba biti događaj s najmanje šest različitih potrebnih informacija, poput: vremenske oznake, identifikatora slučaja, oznaka aktivnosti i druge.

Objekt dnevnika događaja se može kreirati također i pomoću eventlog funkcije unutar R razvojnog okruženja, kod za kreiranje objekta dnevnika događaja se nalazi na slici 13.

```
library(bupaR)

example_log_1 %>% #a data.frame with the
information in the table above
    eventlog(
        case_id = "patient",
        activity_id = "activity",
        activity_instance_id =
"activity_instance",
        lifecycle_id = "status",
        timestamp = "timestamp",
        resource_id = "resource"
    )
```

Slika 13 Kreiranje objekta dnevnika događaja pomoću funkcije eventlog

Izvor slike: https://www.bupar.net/creating_eventlogs.html

3.2. edeaR

edeaR (Exploratory and Descriptive Eventdata Analysis in R) ja paket koji je prvi puta predstavljen na SIMPDA 2016. godine i najstariji je član "obitelji" bupaR. Svrha ovog paketa je izvršiti dublje analize zapisnika događaja, u osnovi sadrži dva skupa funkcija, a to su:

3.2.1. Metrike za istraživanje i opisivanje podataka o događajima

Metrike za istraživanje i opisivanje podataka o događajima koje su dostupne, temelje se na literaturi iz područja operativne izvrsnosti i organizirane su u sljedeće kategorije:

a) Vremenska perspektiva

vrijeme mirovanja (idle_time)

Vrijeme mirovanja predstavlja vrijeme kada nema aktivnosti u slučaju ili resursu, može se računati samo kada postoje početni i završni vremenski žigovi za instance aktivnosti. Mogućnost računanja je na više načina, npr. na razini praćenja, resursa, slučaja i dnevnika ili korištenjem različitih vremenskih jedinica.

To je vidljivo iz primjera sa pacijentima koji slijedi, dakle kada pomoću RStudio pozovemo određene pakete i funkciju kao što je na slici 14:

```
library(bupaR)

library(edeaR)
library(eventdataR)
patients %>%
    idle_time("resource", units = "days")
```

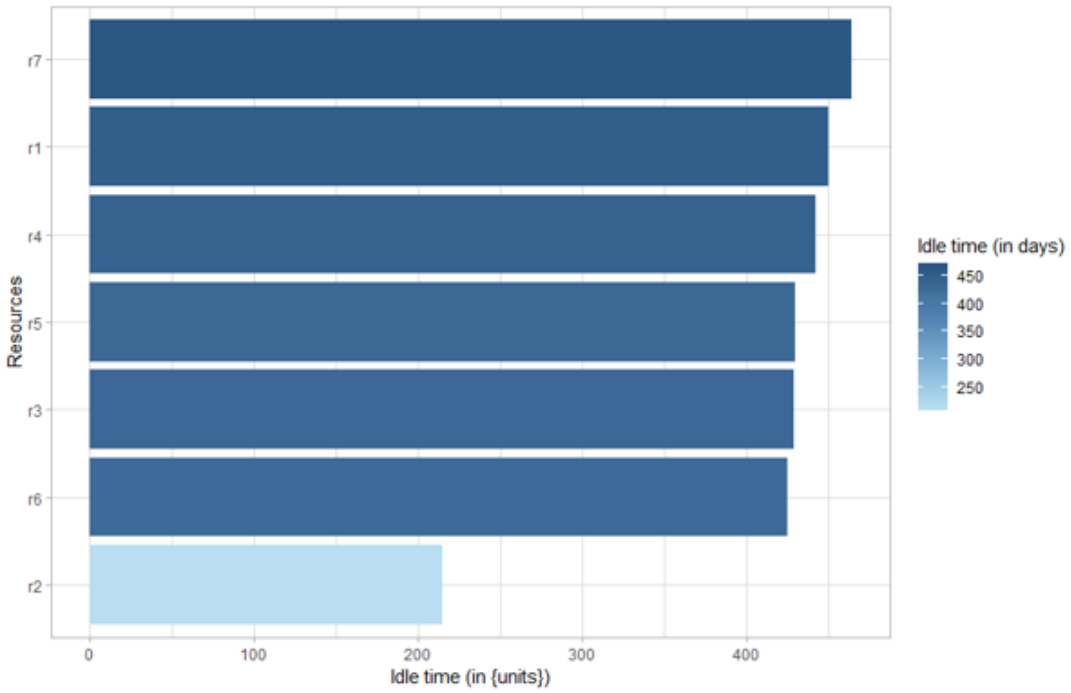
Slika 14 Vrijeme mirovanja - funkcija idle_time

Za ispis dobijemo:

```
# A tibble: 7 x 2
  employee idle_time
  <fct>         <dbl>
1 r7             464.
2 r1             450.
3 r4             443.
4 r5             430.
5 r3             429.
```

Slika 15 Ispis nakon pokretanja idle_time funkcije

Ispis svih metrika dobivenih pomoću paketa edeaR moguće je i vizualizirati, to se postiže samo dodavanjem plot() funkcije, vizualizacija je prikazana na sljedećoj slici 16.

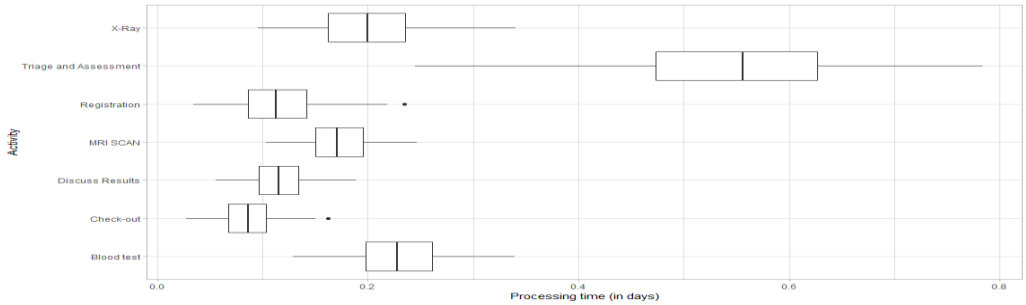


Slika 16 Histogram ispisa metrike

Izvor slike: https://www.bupar.net/exploring.html

vrijeme procesiranja (processing_time)

Vrijeme procesiranja je vrijeme koje se može računati na razini dnevnika, praćenja, slučaja, aktivnosti i na razini resursne aktivnosti, ova metrika se može samo računati u slučajevima kada postoje početni i završni vremenski žigovi za instance aktivnosti. Na slici 17 je prikazano vrijeme procesiranja na temelju primjera iz bolnice. Prikazuje određene aktivnosti i njihovo vrijeme izvršavanja.

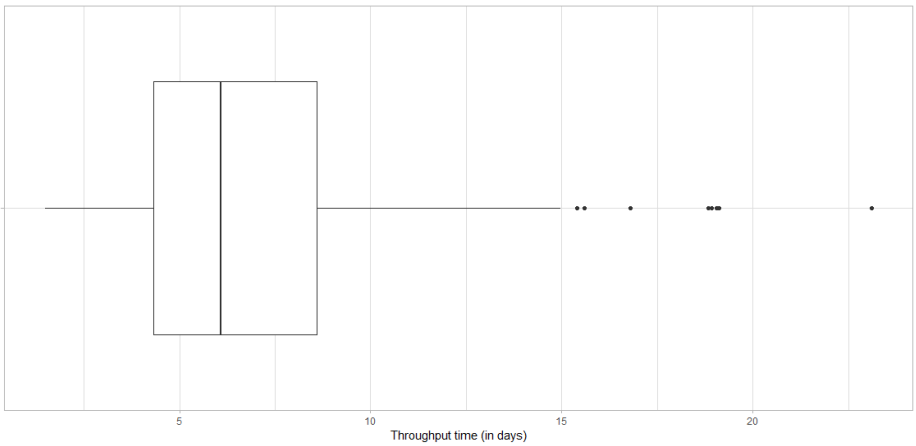


Slika 17 Vrijeme procesiranja vizualizacija

Izvor slike: https://www.bupar.net/exploring.html

vrijeme propusnosti (throughput_time)

Vrijeme propusnosti je vrijeme od prvog događaja do posljednjeg događaja koji se izvršava, razine na kojima se može izračunati su zapisnik, trag ili slučaj. Na slici 18 u nastavku je primjer propusnosti vremena.



Slika 18 Vrijeme propusnosti vizualizacija

Izvor slike: https://www.bupar.net/exploring.html

b) Organizacijska perspektiva

frekvencija resursa (resource frequency)

Metrika za frekvenciju resursa omogućuje izračun broja/učestalosti resursa na razini dnevnika, slučaja, aktivnosti te resursa i aktivnosti resursa.

uključenost resursa (resource involvement)

Sudjelovanje resursa odnosi se na pojam broja slučajeva u kojima je resurs uključen, može se izračunati na razini slučaja, resursa te na razini aktivnosti resursa.

specijalizacija resursa (resource specialization)

Metrika za specijalizaciju resursa pokazuje jesu li resursi specijalizirani za određene aktivnosti ili ne, može se računati na razini dnevnika, slučaja, resursa i aktivnosti.

3.2.2. Filteri za podatkovne podatke

Filtri za podatkovne podatke o događajima mogu se podijeliti u dva tipa:

a) Filteri događaja

Filteri događaja grupirat će dijelove slučajeva na temelju kriterija koji se primjenjuje na događaje (npr. resurs koji ga je izvršio).

Filteri događaja dijele se na:

- Filtriranje aktivnosti (filter activities) – funkcija filtra aktivnosti može se koristiti za filtriranje aktivnosti prema nazivu, sadrži tri argumenta: zapisnik događaja, vektor aktivnosti i obrnuti argumenti.
- Filtriranje na frekvenciji aktivnosti (filter on activity frequency) – pomoću ovog filtera se pregledavaju aktivnosti koje su najčešće dok se ne postigne potreban postotak događaja, dakle 80-postotno odsustvo od postotka gledat će na aktivnosti koje su potrebne da bi predstavljale 80% događaja.
- Filtriranje na atributima (filter on attributes) – generička funkcija, omogućuje filtriranje atributa događaja ili slučajeva, unutar ove funkcije mogu se navesti višestruki uvjeti odvojeni zarezom te se u tom slučaju zarez tretira kao "i".
- Filtriranje resursa (filter resources) – filtriranje resursa je zapravo slično filterima aktivnosti, može se koristiti za filtriranje događaja navođenjem na jedan ili više resursa.
- Slučajevi podrezivanja (trim cases) – poseban filter događaja jer uzima u obzir pojam slučajeva, slučajevi se smanjuju tako da počinju s određenim aktivnostima dok ne završe određenom djelatnošću, zahtjeva dva potpisa (jedan za moguće početne aktivnosti te jedan za krajnje aktivnosti)

b) Filteri slučajeva

Filteri slučajeva mijenjaju kompletne slučajeve, na temelju kriterija koji se primjenjuje na slučajeve (npr. dužina praćenja).

Filteri slučajeva dijele se na:

- Filtriranje prisutnosti aktivnosti (filter activity presence) – omogućuje filtriranje slučajeva koji sadrže određene aktivnosti

- Filter slučaja (filter case) – omogućuje podskup skupa identifikatora slučaja, kao argumente zahtjeva samo vektor identifikatora slučaja (odabir se može zanemariti pomoću reverse = T)
- Filtriranje krajnje točke (filter end points) – može se koristiti na dva načina: određivanjem vektora s dopuštenim početnim aktivnostima ili dopuštenim krajnjim aktivnostima ili određivanjem percentila
- Prednost filtra (filter precedence) – ovaj filter se koristi kako bi se izdvojio podskup dnevnika događaja koji je u skladu s pravilima prioriteta
- Vrijeme obrade filtera, vrijeme propusnosti i duljina traga (filter processinf time, throughput time and trace length) – postoje tri različita filtera koja u obzir uzimaju dužinu slučaja, a to su:
  a) *Vrijeme obrade (processing time)* – zbroj trajanja aktivnosti koje su sadržane u predmetu
  b) *Propusno vrijeme (throughput time)* – vrijeme između prvog događaja i posljednjeg događaja slučaja
  c) *Duljina tragova (trace length)* – broj istanci aktivnosti sadržanih u slučaju
- Filtriranje vremenskog razdoblja (filter time period) – slučajevi se mogu filtrirati tako da se metodi dostavi vremenski prozor
- Filtriranje frekvencije praćenja (filter trace frequency) – koristi se za filtriranje slučajeva na temelju frekvencije odgovarajućeg traga, trag predstavlja slijed oznaka aktivnosti

3.3. xesreadR

Iz samog naziva ovog paketa se može jasno zaključiti koja je njegova svrha, naime paket xesreadR je namijenjen za čitanje i pisanje xes-datoteka. Na taj način bupaR postaje kompatibilan s [5]IEEE standardom za dijeljenje i spremanje podataka o događajima, te samim time i drugim alatima za procesno rudarenje.

Korištenjem funkcije read_xes se može čitati [6]XES – datoteka te se pomoću ove funkcije može pretvarati podatke u objekt dnevnika događaja za R. Funkciji je potrebno dati samo jedan argument koji se zove xesfile, to može biti lokalna putanja do datoteke s ekstenzijom .xes. Na slici u nastavku se nalazi primjer jedne .xes datoteke.



Slika 19 XES datoteka

Izvor slike: https://bupar.net/eventdata/exercise1.xes

Pomoću R-a je lako uvesti .xes datoteke, samo je potrebna putanja do određene .xes datoteke, te se pomoću funkcije data <- read_xes (" ") može pročitati. Tako npr. ako se uzme neki URL na kojem se nalazi .xes datoteka i ako se upiše unutar navodnika taj URL može se pročitati što se nalazi u dokumentu. Ako korisnik želi iščitati što se nalazi na slici 18 potrebno je upisati u RStudio dio koda kao na slici 20.

```
data <- read_xes("C:\Users\MONIKA\Documents\monika.xes")

data
```

Slika 20 Funkcija za čitanje xes datoteke

Izvor slike: Vlastiti rad

[5] Profesionalna udruga koja razvija, definira i ocjenjuje standarde elektronike i informatike
[6] Format utemeljen na XML-u

Te kada se pokrene kod unutar konzole se može iščitati xes datoteku, kao što je na slici 21.

```
>   data
Log of 11 events consisting of:
3 traces
3 cases
11 instances of 5 activities
1 resource
Events occurred from 2008-12-09 07:20:01 until 2008-12-09 07:23:01

Variables were mapped as follows:
Case identifier:                CASE_concept_name
Activity identifier:            activity_id
Resource identifier:            resource_id
Activity instance identifier:   activity_instance_id
Timestamp:                      timestamp
Lifecycle transition:           lifecycle_id

# A tibble: 11 x 7
   CASE_concept_name activity_id lifecycle_id resource_id timestamp           activity_instance_id .order
   <chr>             <fct>       <fct>        <fct>       <dttm>              <chr>                <int>
 1 Case3.0           A           complete     UNDEFINED   2008-12-09 07:20:01 1                        1
 2 Case3.0           E           complete     UNDEFINED   2008-12-09 07:21:01 2                        2
 3 Case3.0           D           complete     UNDEFINED   2008-12-09 07:22:01 3                        3
 4 Case2.0           A           complete     UNDEFINED   2008-12-09 07:20:01 4                        4
 5 Case2.0           C           complete     UNDEFINED   2008-12-09 07:21:01 5                        5
 6 Case2.0           B           complete     UNDEFINED   2008-12-09 07:22:01 6                        6
 7 Case2.0           D           complete     UNDEFINED   2008-12-09 07:23:01 7                        7
 8 Case1.0           A           complete     UNDEFINED   2008-12-09 07:20:01 8                        8
 9 Case1.0           B           complete     UNDEFINED   2008-12-09 07:21:01 9                        9
10 Case1.0           C           complete     UNDEFINED   2008-12-09 07:22:01 10                      10
11 Case1.0           D           complete     UNDEFINED   2008-12-09 07:23:01 11                      11
>
```

Slika 21 Čitanje xes datoteke pomoću RStudio

Izvor slike: Vlastiti rad

Ovdje je dobro napomenuti da postoji funkcija i za kreiranje xes datoteke, args (write_xes). Funkcija write_xes zahtjeva minimalno dva argumenta, a to su:

a) objekt zapisnika događaja
b) putanja datoteke gdje će se pohraniti datoteka

Nakon određivanja oba argumenta može se pokrenuti funkcija kao što je na slici 22.

```
args(write_xes)

write_xes(patients, "monika.xes")
```

Slika 22 Funkcija za kreiranje .xes datoteke

Izvor slike: Vlastiti rad

Nakon što je pokrenuto kreiranje .xes datoteke u konzoli se može pratiti kada je kreirano, tako da se prikaže kada je izrada gotova, slika 23 prikazuje izradu .xes datoteke u konzoli.

```
> write_xes(patients, "monika.xes")
  |=============================================================| 100%
```

Slika 23 Izvođenje funkcije write_xes

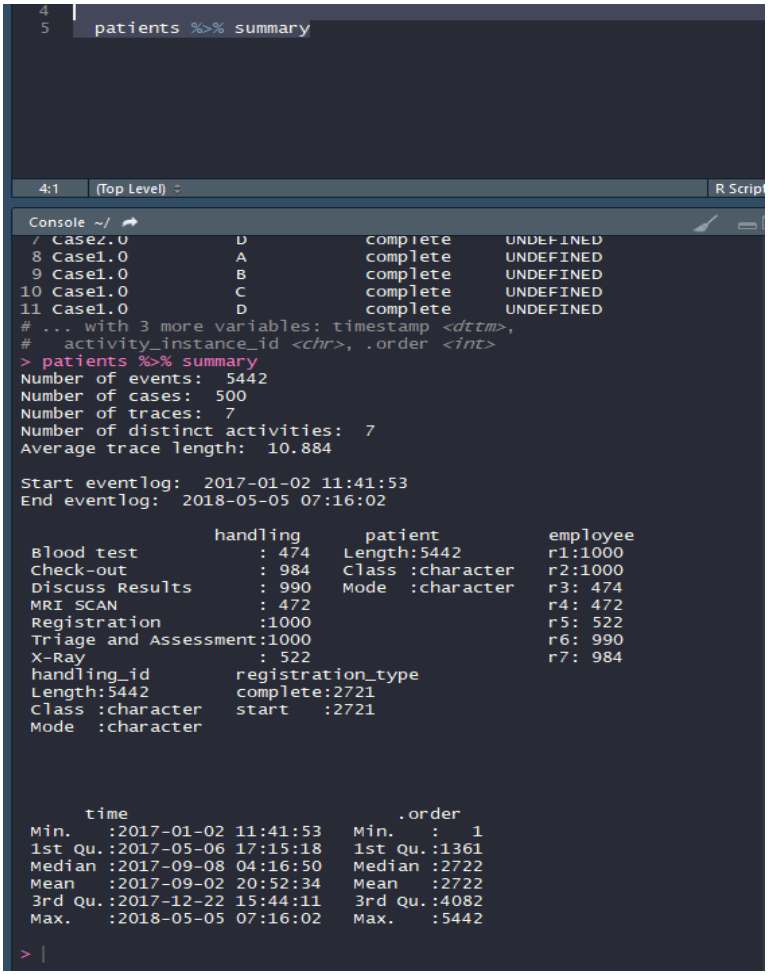
Izvor slike: Vlastiti rad

3.4. eventdataR

Paket eventdataR se koristi s ciljem testiranja novih tehnika i algoritama, eventdata R sadrži spremište podataka za umjetne i za stvarne dnevnike događaja. Svako spremište podataka se može učitati pomoću data funkcije. Trenutno postoje ovi zapisi događaja:

a) Umjetni podaci

Kod umjetnih podataka možemo navesti primjer o pacijentima koji dolaze u hitnu službu, dnevnik događaja je korišten kao primjer u časopisu pod nazivom "Retrieving batch organisation of work insights from event logs."



Slika 24 Prikaz umjetnih podataka – eventdataR

Izvor slike: Vlastiti rad

Na slici 24 se može vidjeti koliko je događaja izvršeno, koliko je slučajeva bilo, broj aktivnosti i mnoge druge pojedinosti vezane uz pacijente.

b) Podaci iz stvarnog života

Za primjer iz stvarnog života upotrebljeno je upravljanje cestovnim prometom, tako se sa slike 25 može kao i u primjeru sa umjetnim podacima iščitati podatke kao što je broj događaja, broj slučajeva, te se vidi kako funkcionira poslovni proces upravljanja cestovnim prometom.

```
> traffic_fines %>% summary
Number of events:  34724
Number of cases:  10000
Number of traces:  44
Number of distinct activities:  11
Average trace length:  3.4724

Start eventlog:  2006-06-17
End eventlog:  2012-03-26

   case_id                              activity
 Length:34724       Create Fine              :10000
 Class :character   Send Fine                : 6570
 Mode  :character   Payment                   : 4910
                    Add penalty              : 4635
                    Insert Fine Notification  : 4635
                    Send for Credit Collection: 3387
                    (Other)                  :  587
    lifecycle        resource       timestamp
 complete:34724   559     : 1278   Min.   :2006-06-17 00:00:00
                  557     : 1181   1st Qu.:2007-07-19 00:00:00
                  541     : 1119   Median :2007-10-29 00:00:00
                  561     : 1002   Mean   :2007-12-06 15:34:51
                  558     :  948   3rd Qu.:2008-02-29 00:00:00
                  (Other): 4491   Max.   :2012-03-26 00:00:00
                  NA's    :24705
     amount          article       dismissal
 Min.    : 210.0   Min.    :  7.0   Length:34724
 1st Qu.: 360.0   1st Qu.:157.0   Class :character
 Median : 360.0   Median :157.0   Mode  :character
 Mean   : 459.3   Mean   :131.8
 3rd Qu.: 740.0   3rd Qu.:157.0
 Max.   :1480.0   Max.   :158.0
 NA's   :20089    NA's   :24724
    expense         lastsent          matricola
 Min.   :110.0   Length:34724       Length:34724
 1st Qu.:130.0   Class :character   Class :character
 Median :130.0   Mode  :character   Mode  :character
 Mean   :131.9
```

Slika 25 Podaci iz stvarnog života – eventdataR

Izvor slike: Vlastiti rad

3.5. processmonitR

Paket processmonitR pruža nekoliko unaprijed definiranih nadzornih ploča za praćenje procesa na temelju zapisnika događaja. Mogu se pokrenuti pomoću sljedećih funkcija:

a) performance_dashboard – može se pregledati vrijeme protoka, vrijeme obrade, vrijeme mirovanja
b) activity_dashboard – mogu se pregledati frekvencije aktivnosti i prisutnosti
c) rework_dashboard – može se pregledati prerada (selfloops)
d) resource_dashboard – može se pregledati učestalost resursa, uključenost ili specijalizacija

4. Primjeri primjene bupaR paketa za analizu poslovnih procesa u R-u

Paket processmapR pruža jednostavan način za stvaranje vizualizacije procesa pomoću prilagodljivih profila. Postoji pet profila karata, a to su:

a) Profil frekvencije
b) Profil izvedbe
c) Prilagođeni profil
d) Kombinacija različitih profila
e) Prilagodba izgleda

U nastavku se nalaze dva primjer, prvi je preuzet sa interneta, drugi koji se temelji na dataset-u je preuzet sa BPI challange-a s Kaggle.com 2017. godine.

PRVI PRIMJER:

Za početak je potrebno učitati library (bupaR) te pozvati naredbu process_map(), kao što se nalazi na slici 26.

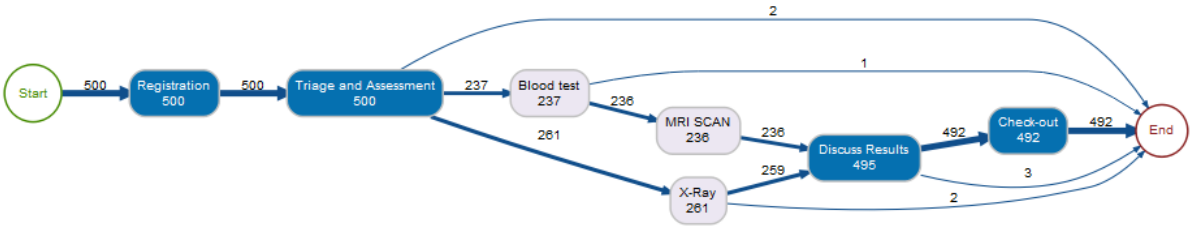
```
library(bupaR)

patients %>%
    process_map()
```

Slika 26 Učitavanje library(bupaR) i naredbe process_map()

Izvor slike: https://www.bupar.net/processmaps.html

Nakon što se pozove naredba, unutar Viewer-a možemo pogledati proces koji nam program ispisuje, kao što je na slici 27.

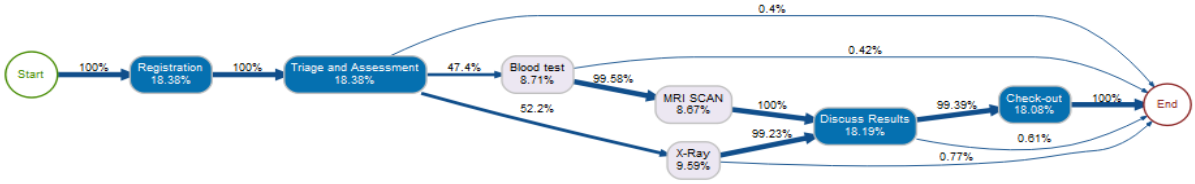


Slika 27 Vizualizacija procesa za pacijente

Izvor slike: https://www.bupar.net/processmaps.html

a) Profil frekvencije

Pri izradi procesne mape vidljivo je kako je karta označena frekvencijama aktivnosti i tokova, to je ono što se naziva profil frekvencije, te se može kreirati eksplicitno korištenjem frequency funkcije. Funkcija frequency ima value argument koji se može koristiti za podešavanje prikazanih frekvencija, primjerice ako se koriste relativne frekvencije umjesto zadanih apsolutnih frekvencija dobije se procesna mapa kao na slici 28.



Slika 28 Vizualizacija procesa uz korištenje relative frequency

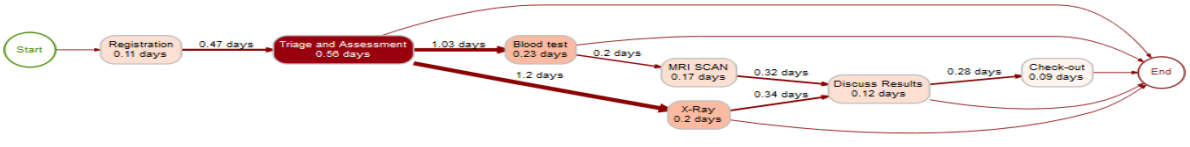
Izvor slike: https://www.bupar.net/processmaps.html

Postoje različite vrijednosti frekvencija, kao što su:

- Absolute frequency – apsolutni broj slučajeva i tokova aktivnosti
- Absolute_case frequency – apsolutni broj slučajeva iza svake aktivnosti i toka
- Relative frequency – relativni broj primjeraka po aktivnosti
- Relative_case frequency – relativni broj slučajeva po aktivnosti

b) Profil izvedbe

Profil izvedbe temelji se na naglasku vremena obrade aktivnosti. Profil izvedbe ima dva argumenta: agrument FUN za određivanje funkcije koja se primjenjuje na vrijeme obrade, te drugi argument koji predstavlja vremensku jedinicu koja će se koristiti. Na slici 29 prikazana je procesna mapa koja sadrži prvi argument kao medijan te drugi argument, točnije vremensku jedinicu u danima.

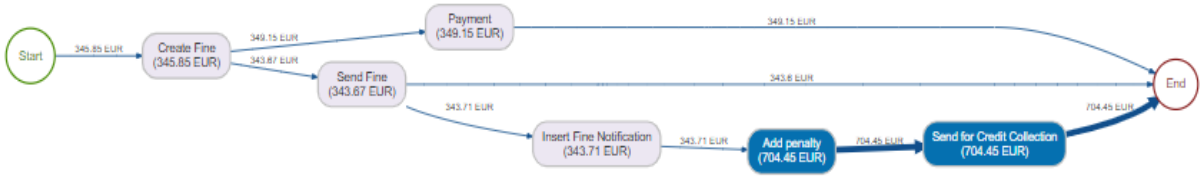


Slika 29 Vizualizacija procesa korištenjem profila izvedbe

Izvor slike: https://www.bupar.net/processmaps.html

c) Prilagođeni profil

Osim specifičnih profila, kao što su profil pregleda i profil frekvencija, moguće je projicirati prilagođeni numerički atribut, takvo projiciranje moguće je postići pomoću prilagođenog profila. Ovaj profil zahtjeva agregaciju (npr. srednja vrijednost, suma, minimum, medijan).

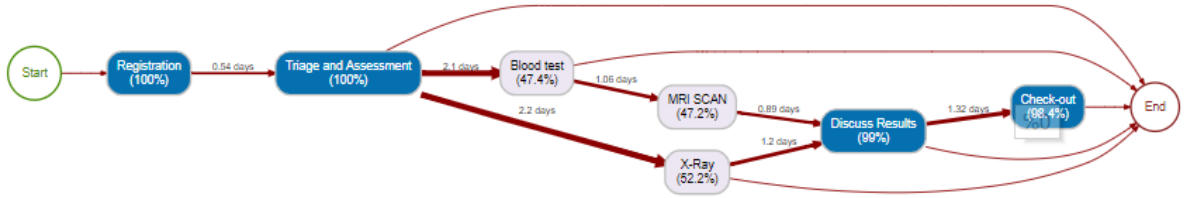


Slika 30 Vizualizacija procesa korištenjem prilagođenog profila

Izvor slike: https://www.bupar.net/processmaps.html

d) Kombinacija različitih profila

Kombinirajući profil se koristi za razlikovanje rubova i čvorova pomoću type_edges i type_nodes atributa umjesto type argumenta, na ovaj način korisnik može informacije o frekvencijama i performansama ili bilo kojoj drugoj vrijednosti kombinirati u istom grafikonu. Primjer kombinacije različitih profila prikazan je na slici 31.



Slika 31 Vizualizacija procesa korištenjem kombiniranih različitih profila

Izvor slike: https://www.bupar.net/processmaps.html

e) Prilagodba izgleda

Izgled pojedinih karata procesa može se dodatno prilagoditi, npr. umjesto rasporeda lijevog-desnog, rankdir funkcija može postaviti na TB točnije na vrh-dno, BT na vrh-dno ili RL na desno-lijevo, također je moguće onemogućiti promjenu širine ruba.

Drugi primjer:

Prije same obrade drugog primjera potrebno je naglasiti da je važna obrada podataka. Obrada podataka je tehnika data mininga pomoću koje možemo pretvarati neobrađene podatke u razumljiv format, podaci koje susrećemo u stvarnom svijetu su većinom neobrađeni. Tako da prije same izrade modela, podatke je potrebno obraditi.

Za početak ovog primjera treba naglasiti što konkretno on definira, koji poslovni procesi se analiziraju. Dakle, skup podataka je preuzet sa BPI challange-a s stranice Kaggle.com 2017. godine. Pomoću ovih podataka izrađen je dnevnik događaja koji se odnosi na procese prijave zajma nizozemskog financijskog instituta, podaci sadrže sve prijave koje su podnesene putem internetskog sustava 2016. godine te sve njihove naknadne događaje do 01. veljače 2017. godine. Tvrtka koja dostavlja podatke i proces koji se razmatra isti je kao [7]DOI: 10.4121 te [8]UUID: 3926db30-f712-4394-aebc-75976070e91f. Sustav koji podupire proces se u međuvremenu promijenio, konkretno sustav sada omogućuje više ponuda po aplikaciji te se ponude mogu pratiti kroz njihove ID-ove u zapisniku. U zapisnik događaja bilježi se mnogo događaja. Za svaki događaj bilježi se zaposlenik koji je uzrokovao događaj, kao i datum i informacije o životnom ciklusu.

Stoga, na početku je potrebno učitati i instalirati velik broj paketa, na slici 32 nalazi se popis svih korištenih paketa.

```
install.packages("bupaR")
install.packages("edeaR")
install.packages("processmapR")
install.packages("eventdataR")
install.packages("readr")
install.packages("DiagrammeR")
install.packages("stringr")
install.packages("lubridate")
install.packages("DiagrammeRsvg")
install.packages("rsvg")
install.packages("V8")
install.packages('rvest')
install.packages("tidyverse")


library(bupaR)
library(edeaR)
library(processmapR)
library(eventdataR)
library(readr)
```

Slika 32 Instalacija paketa potrebnih za primjer procesne mape

Izvor slike: Vlastiti rad

[7] Digitalni identifikator objekta, predstavlja jedinstveni alfanumerički niz koji dodjeljuje agencija kako bi identificirala sadržaj i osigurala trajnu vezu na svoje mjesto na Internetu, izdavač dodjeljuje DOI kada je članak objavljen i dostupan na internetu

[8] Univerzalni jedinstveni identifikator, predstavlja 128-bitni broji kojise koristi za identifikaciju podataka u računalnim sustavima

Neki od paketa nisu spomenuti u radu, stoga u nastavku slijedi kratki opis pojednih paketa:

a) readr – obzirom da je primjer temeljen na csv datoteci, kako bi ju mogli pročitati potrebno je koristiti paket readr, osmišljen je za fleksibilno analiziranje puno vrsta podataka
b) DiagrammeR – paket namijenjen izgradnji grafikona pomoću funkcija za postupno dodavanje i brisanje čvorova i rubova
c) Lubridate – funkcija za rad s datumskim vremenima i vremenskim rasponim: služi za brzo i jednostavno prilagođavanje podataka datuma i vremena

Nakon instalacije svih paketa, napokon je potrebno izvršiti obradu podataka, obzirom da je korišten određeni dataset (loan_application.csv) potrebno ga je učitati, za čitanje csv datoteke potrebno je koristiti paket readr, kao što se vidi na slici 33.

```
data <-
readr::read_csv('/Users/MONIKA/Desktop/zavrsni_rad/loan_applications.csv',
                locale = locale(date_names='en', encoding = 'ISO-8859-1'))
```

Slika 33 Učitavanje podataka

Izvor slike: Vlastiti rad

Na slici 34 je prikazano kako promijeniti timestamp u željeni format kako bi mogli raditi s tim podacima, POSIXct je funkcija koja je već spomenuta na početku ovoga rada, ona označava mogućnost reprezentacije datuma u kalendaru, također je navedeno brisanje nepotrebnih znakova poput " " , = te _ .

```
data$starttimestamp <- as.POSIXct(data$'starttimestamp', format =
"%Y/%m/%d %H:%M:%S")


data$endtimestamp <- as.POSIXct(data$'endtimestamp', format = "%Y/%m/%d
%H:%M:%S")

names(data) <- str replace all(names(data), c(" " = " ", "," = ""))
```

Slika 34 Korištenje funkcije POSIXct

Izvor slike: Vlastiti rad

Srce ovog primjera je kreiranje zapisnika događaja. Dakle, kada smo napravili sva formatiranja, potrebno je kreirati zapisnik događaja. Pomoću zapisinika događaja na jednostavan način možemo analizirati što određeni korisnik radi, koje su mu vremenske komponente te na koji način i koje aktivnosti izvršava.

Na slici 35 je prikazana izrada zapisnika događaja svih podataka koje imamo.

```
events <- bupaR::activities_to_eventlog(

  head(data, n = 10000),
  case_id = 'Case_ID',
  activity_id = 'Activity',
  resource_id = 'Resource',
  timestamps = c('starttimestamp', 'endtimestamp')
)

events <- bupaR::activities_to_eventlog(
  data,
  case_id = 'Case_ID',
  activity_id = 'Activity',
  resource_id = 'Resource',
  timestamps = c('starttimestamp', 'endtimestamp')
)

events %>%
  summary
```

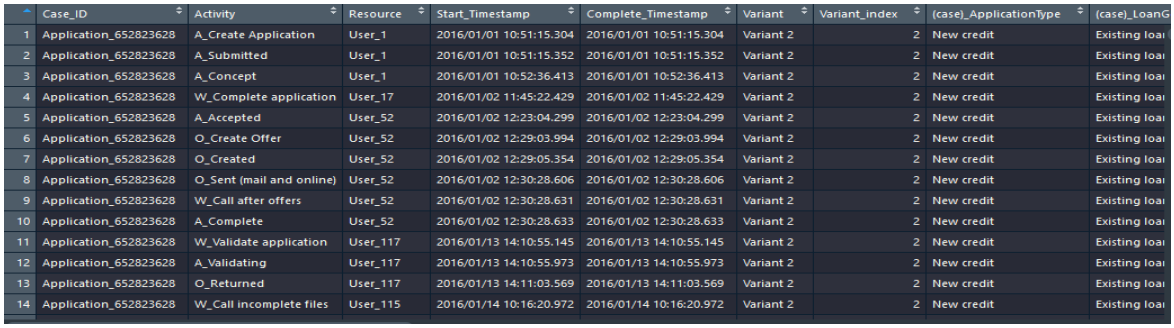
Slika 35 Kreiranje events-a

Izvor slike: Vlastiti rad

Kao što je već u poglavlju 3.1. bupaR spomenuto o svim argumentima ovdje se može vidjeti konkretno primjer vezan za pojedini argument. Stoga, kada korisnik pokrene dio koda kod izrade zapisnika događaja, dobit će informacije kao na slici 36.

Na ovoj slici se svaki red zapravo odnosi na određeni događaj, kako bi se događaji koji pripadaju istoj instanci aktivnosti mogli povezati, potreban je identifikator instance aktivnosti. Iz ove slike se mogu vidjeti točno aktivnosti koje se analiziraju u ovom poslovnom procesu, naime User_117 izvršio je aktivnost „A_Validating“ . Dakle, aktivnost „A_Validating“ predstavlja procjenu primljenih dokumenata od strane banke. Kao što je navedeno na početku za svaki događaj koji je izvršen bilježi se i zaposlenik koji je izvršio taj događaj. Podaci su kao što se može vidjeti anonimni. Također, na slici se može vidjeti complete_timestamp ova varijabla prikazuje vrijeme završetka izvršene aktivnosti, te start_timestamp koji prikazuje vrijeme početka aktivnosti, osim toga može se vidjeti application_type pomoću kojeg se očituje je li aplikacija namijenjena novom kreditu ili povišici za postojeći kredit.

Obzirom da je slika velika pojedine varijable se ne vide, stoga se popis korištenih varijabli nalazi na tablici 1 ispod slike.



	Case_ID	Activity	Resource	Start_Timestamp	Complete_Timestamp	Variant	Variant_index	(case)_ApplicationType	(case)_LoanG
1	Application_652823628	A_Create Application	User_1	2016/01/01 10:51:15.304	2016/01/01 10:51:15.304	Variant 2	2	New credit	Existing loai
2	Application_652823628	A_Submitted	User_1	2016/01/01 10:51:15.352	2016/01/01 10:51:15.352	Variant 2	2	New credit	Existing loai
3	Application_652823628	A_Concept	User_1	2016/01/01 10:52:36.413	2016/01/01 10:52:36.413	Variant 2	2	New credit	Existing loai
4	Application_652823628	W_Complete application	User_17	2016/01/02 11:45:22.429	2016/01/02 11:45:22.429	Variant 2	2	New credit	Existing loai
5	Application_652823628	A_Accepted	User_52	2016/01/02 12:23:04.299	2016/01/02 12:23:04.299	Variant 2	2	New credit	Existing loai
6	Application_652823628	O_Create Offer	User_52	2016/01/02 12:29:03.994	2016/01/02 12:29:03.994	Variant 2	2	New credit	Existing loai
7	Application_652823628	O_Created	User_52	2016/01/02 12:29:05.354	2016/01/02 12:29:05.354	Variant 2	2	New credit	Existing loai
8	Application_652823628	O_Sent (mail and online)	User_52	2016/01/02 12:30:28.606	2016/01/02 12:30:28.606	Variant 2	2	New credit	Existing loai
9	Application_652823628	W_Call after offers	User_52	2016/01/02 12:30:28.631	2016/01/02 12:30:28.631	Variant 2	2	New credit	Existing loai
10	Application_652823628	A_Complete	User_52	2016/01/02 12:30:28.633	2016/01/02 12:30:28.633	Variant 2	2	New credit	Existing loai
11	Application_652823628	W_Validate application	User_117	2016/01/13 14:10:55.145	2016/01/13 14:10:55.145	Variant 2	2	New credit	Existing loai
12	Application_652823628	A_Validating	User_117	2016/01/13 14:10:55.973	2016/01/13 14:10:55.973	Variant 2	2	New credit	Existing loai
13	Application_652823628	O_Returned	User_117	2016/01/13 14:11:03.569	2016/01/13 14:11:03.569	Variant 2	2	New credit	Existing loai
14	Application_652823628	W_Call incomplete files	User_115	2016/01/14 10:16:20.972	2016/01/14 10:16:20.972	Variant 2	2	New credit	Existing loai

Slika 36 Zapisnik događaja

Izvor slike: Vlastiti rad

Primjer koji se nalazi u tablici vođen je CaseID-om: Application_652823628, te aktivnost „O_Create Offer“ (aktivnost ponude).

Tablica 1 Prikaz dodatnih varijabli i primjer

Varijabla	Objašnjenje	Primjer
Requested Amount	Traženi iznos kredita	20.000
Credit Score	Opis je li podnositelj zahtjeva pouzdan	979
OfferID	Identifikator za svaku ponudu	NA
Offered Amonut	Iznos novca koji se nudi podnositelju zahtjeva	20.000

Slika 35 je također prikazivala funkciju summary, to je funkcija koja se koristi za dobivanje rezultata gdje se ispisuje sažetak korištenja različitih funkcija ugradnje modela. Na slici 37 je prikazana analiza poslovnih procesa za korišteni skup podataka, na njemu se može iščitati case_id koji predstavlja klasifikator slučaja dnevnika i on sadrži nazive varijabli, također se mogu iščitati aktivnosti kao što su prihvaćanje, slanje i mnoge druge. Na slici je vidljivo da postoji čak 800000 događaja koji su vezani za 22509 zahtjeva za kredit.

```
Number of events:  800000
Number of cases:  22509
Number of traces:  3096
Number of distinct activities:  26
Average trace length:  35.54134

Start eventlog:  2016-01-01 09:51:15
End eventlog:  2017-01-26 09:11:10

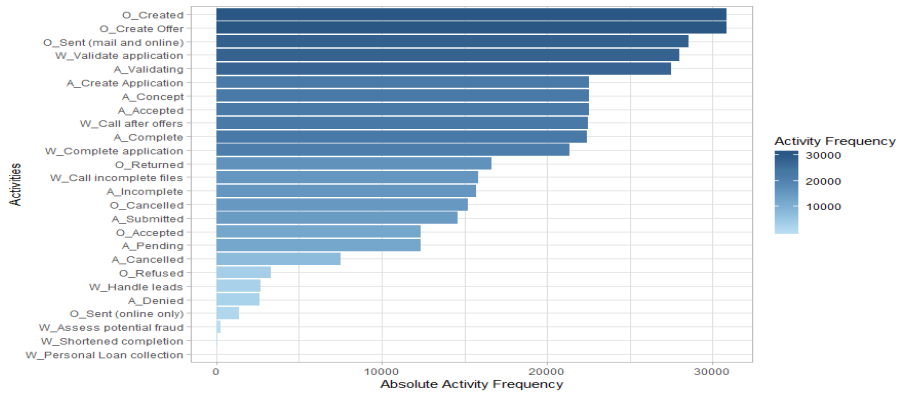
   Case_ID                           Activity         Resource      Start_Timestamp    Complete_Timestamp
 Length:800000      O_Create Offer          : 61706   User_1 :108886  Length:800000      Length:800000
 Class :character   O_Created               : 61706   User_49: 16624  Class :character   Class :character
 Mode  :character   O_Sent (mail and online): 57130   User_29: 15224  Mode  :character   Mode  :character
                    W_Validate application  : 55986   User_10: 14736
                    A_Validating            : 54952   User_3 : 14584
                    A_Accepted              : 45018   User_27: 14132
                    (Other)                 :463502   (Other):615814
   Variant           Variant_index    (case)_ApplicationType (case)_LoanGoal    (case)_RequestedAmount  Accepted
 Length:800000      Min.    :   1.0   Length:800000          Length:800000     Min.    :     0         Mode :logical
 Class :character   1st Qu.:   7.0   Class :character       Class :character  1st Qu.:  6000         FALSE:18602
 Mode  :character   Median :  25.0   Mode  :character       Mode  :character  Median : 12000         TRUE :43104
                    Mean   : 385.5                                           Mean   : 15877         NA's :738294
                    3rd Qu.: 249.0                                           3rd Qu.: 20000
                    Max.   :3159.0                                           Max.   :450000

    Action          CreditScore       EventID          EventOrigin        FirstWithdrawalAmount  MonthlyCost
 Length:800000      Min.    :   0.0   Length:800000      Length:800000      Min.    :    0         Min.    :  43.0
 Class :character   1st Qu.:   0.0   Class :character   Class :character   1st Qu.:    0         1st Qu.: 150.0
 Mode  :character   Median :   0.0   Mode  :character   Mode  :character   Median : 5000         Median : 232.8
                    Mean   : 317.5                                        Mean   : 7886         Mean   : 275.5
                    3rd Qu.: 849.0                                        3rd Qu.:11000         3rd Qu.: 345.2
```

Slika 37 Sažetak podataka

Izvor slike: Vlastiti rad

Nakon što je sve kreirano poziva se funkcija plot() koja služi za crtanje R objekata, nakon pokretanja funkcije, unutar prozora Environment->Plots možemo pregledati histogram sa slike 38. Na histogramu se mogu vidjeti aktivnosti koje se analiziraju za ovaj poslovni proces, te na temelju histograma se može zaključiti kako je aktivnost W_Personal Loan collection aktivnost s najmanjom frekvencijom.



Slika 38 Histogram podataka

Izvor slike: Vlastiti rad

```
events %>%

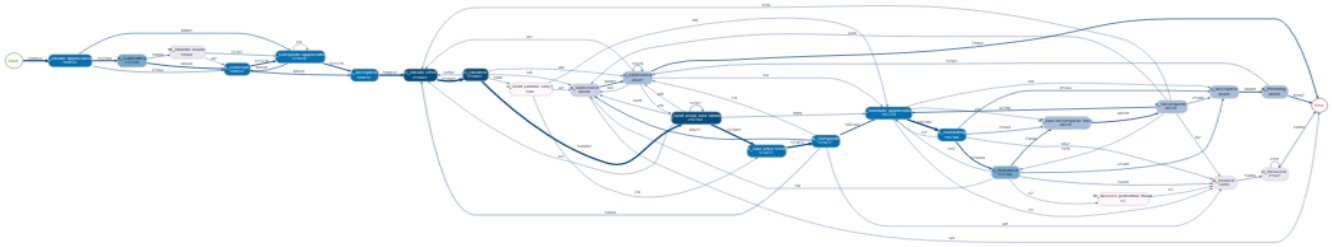
  filter_activity_frequency(percentage = 1.0) %>%
  filter_trace_frequency(percentage = .80) %>%
  process_map(type = frequency("absolute"))
```

Slika 39 Kreiranje procesne mape

Izvor slike: Vlastiti rad

Na kraju se nalazi zapravo najbitniji dio ovog primjera, a to je procesna mapa, koja prikazuje nakon ove analize poslovnog procesa i njegovu vizualizaciju. Korištena su dva filtera, koja su prethodno već pojašnjena. Filter_activity_frequency se koristi za pregledavanje aktivnosti koje su najčešće sve dok se ne postigne potreban postotak događaja, te filter_trace_frequency pomoću kojeg se filtriraju slučajevi na temelju frekvencije odgovarajućeg traga. Na slici 39 se vidi kod pomoću kojeg je kreirana u konačnici procesna mapa, kao što je na slici 40.

U konačnici ovoga primjera dobivamo procesnu mapu, točnije vizualiziran jedan poslovni proces. Kada bi ga bolje pogledali, možemo lako uvidjeti sve aktivnosti koje ima, te analizom ishoda svake aktivnosti možemo doći do ideja kako poboljšati poslovni proces.



Slika 38 Procesna mapa

Izvor slike: Vlastiti rad

Dakle, na slici 40 se vide aktivnosti poput primjene zajmova kredita, potvrde zahtjeva i odluka o tome hoćete li dati ponudu ili ne, odgovor podnositelja zahtjeva. Podaci za analizu podataka sadrže 18012 različitih instanci procesa. Aktivnosti su podijeljene u tri kategorije, a to su:

a) Aplikacijske aktivnosti (A_)
b) Aktivnosti ponude(O_)
c) Aktivnosti tijeka rada(W_)

Primjer u daljnjem opisu se bazira na aplikacijskim aktivnostima te aktivnostima ponude. Postupak podnošenja zahtjeva za zajam započinje kreiranjem aplikacije

(A_Create Application) i završava sa jednom od sljedećih točaka, npr. A_Pending (da je određena ponuda prihvaćena), A_Cancelled (ponuda je bila otkazana) i A_Denied (ponuda je odbijena). U analizi je primjer usredotočen na vrijeme čekanja između tih aktivnosti.

Vrijeme koje se čeka između svake aktivnosti je vrijeme koje banka koristi kod unosa podataka podnositelja zahtjeva ili pak zahtjev čeka određeno vrijeme da ga banka obradi. Procesi koji utječu na ukupno vrijeme protoka procesa su:

- Proces od aktivnosti A_Concept do aktivnosti A_Accepted - vrijeme koje prikazuje čekanje nakon stvaranja aplikacije i vrijeme prije nego što proces kreiranja ponude zahtjeva pokrene zaposlenik banke
- Proces od aktivnosti O_Returned do aktivnosti O_Accepted – vrijeme čekanja nakon postupka validacije je završeno te je pokrenut odgovarajući postupak za nedovršene aplikacije, ako je ponuda prihvaćena status se mijenja iz O_Returned u O_Accepted
- Proces od aktivnosti A_Incomplete do aktivnosti O_Accepted – vrijeme čekanja prije prihvaćanja nedovršene ponude

Tokom pregledanja cijelokupnog poslovnog procesa da se zaključiti da je postupak zahtjeva pokrenuo ili podnositelj zahtjeva u banci ili putem internetske stranice banke. Svaki kandidat će dobiti barem jednu ponudu koja je ili odbijena ili prihvaćena. Te se time da zaključiti kako je vrijeme čekanja na odobrenje ili odbijanje iznimno dugo. Također, iz podataka se može zaključiti da vrijeme čekanja može povećati nepotpunost dokumenata ili nedovoljna komunikacija između podnositelja zahtjeva i banke.

Ukupnom analizom poslovnog procesa ideja za poboljšanje poslovnog procesa je zapravo da bi bilo dobro skratiti vremenski period između podnošenja prijave putem web-lokacije i završetka zahtjeva, te bi bilo dobro implementirati određeni alat za umjetnu inteligenciju kojim bi se automatiziralo prihvaćanje ponude, te na mjestima gdje je to moguće automatizirati i proces stvaranja ponude.

5. Zaključak

R je najrasprostranjeniji programski jezik za statističku analizu podataka. Ovaj programski jezik se smatra dobro razvijenim i jednostavnim programskim jezikom koji uključuje uvjetovane petlje, rekurzivne funkcije, ulazne i izlazne mogućnosti te nudi mnoštvo korisnih mogućnosti koje uvelike pomažu pri analizi poslovnih procesa.

R je jedan od najraširenijih svjetskih programskih jezika za statističku analizu podataka. Velika prednost R-a je između ostalog, njegova podrška slobodnog dohvata, te je dostupan krajnjim korisnicima. Treba istaknuti da ga danas koriste velike poznate tvrtke kao što su: Facebook, Google, Bing i Mozilla. Od početka rada R programskog jezika iz davne 1993., te verzije 1.0.0. koja je 2000. godine puštena u javnost R se razvijao i napredovao do današnjih dana, no verziju 3.6.1. koja se danas koristi još ne možemo smatrati savršenom jer sa novim tehnologijama 21. stoljeća razvija se sve više alata za statističku analizu podataka.

Prema primjerima koji su korišteni u ovom radu, može se zaključiti kako korištenjem R programskog jezika nije teško raditi analizu poslovnih procesa, korištenjem ovog jezika jednostavno je doći do kompletnih podataka koji prikazuje što pojedini poslovni proces radi. Naime, korištenjem procesnog rudarstva lako dolazimo do bitnih odgovora na relevantna poslovna pitanja, razumijevanje podataka, tumačenje podataka i što je naravno najvažnije, izvođenje mjera za poboljšanje analiziranog procesa. bupaR je alat koji podržava ovaj iterativni postupak procesnog rudarstva te omogućuje besplatnu analizu podataka. U konačnici za drugi primjer je izrađena još i vizualizacija poslovnog procesa te je jasno vidljivo na koji način se dolazi do vizualizacije procesa.

Na temelju navedenog vidljivo je kako je jednostavno razumijeti tijek analize poslovnog procesa od paketa potrebnih za kreiranje modela, preko kreiranja zapisnika događaja koji je ujedno i središte primjera do histograma koji je kreiran pomoću funkcije plot().

U konačnici se da zaključiti da je analiza poslovnih procesa iznimno bitna, analiza je bitna prvenstveno jer danas svako poduzeće mora neprestano vrednovati svoje procese i razmišljati o tome kako ih poboljšati, kako vrijeme prolazi, nova tehnologija postaje dostupna te sam as-is proces čini zastarjelim, a ponekad takav proces jednostavno više nije učinkovit jer troši previše resursa. Analiza poslovnih procesa svakom korisniku omogućuje da ima temeljit i detaljan pogled na proces i njegove komponente. Koristeći programsko R okruženje za analizu i vizualizaciju poslovnih

procesa, korisnik lako dobiva uvid u svaku aktivnost koja se izvršava, tko izvršava određenu aktivnost te sve ostale pojedinosti poput vremenskih kompnenti, statusa na temelju kojih je puno jednostavnije doći do relevantnih rješenja za poboljšanje. Temeljem primjera lako se može zaključiti koja poboljšanja bi dobro došla unutar poslovanja banke, naime skraćivanjem vremenskog perioda čekanja između podnošenja prijave i završetka zahtjeva za kredit cijeli proces izvođenja bi se uvelike smanjio i bilo bi praktičnije izvršiti zahtjeve za kredit.

6. Popis slika i tablica

Slike:

Slika 1 Dokumentacija za paket base 6
Slika 2 Uspješna instalacija paketa "ggplot2" 7
Slika 3 Statistika preuzimanja R paketa 2013. godine 8
Slika 4 Statistika preuzimanja paketa 2017. godine 8
Slika 5 Area vs Population - ggplot2 9
Slika 6 RStudio okruženje 10
Slika 7 R Source prozor 11
Slika 8 Primjer ispisa rezultata u konzoli 11
Slika 9 Rješavanje zadataka u konzoli 12
Slika 10 Model poslovnog procesa na različitim razinama detaljnosti 16
Slika 11 Kreiranje event log 17
Slika 12 Kreiranje pojedinosti za zapisnik događaja 19
Slika 13 Kreiranje objekta dnevnika događaja pomoću funkcije eventlog 19
Slika 14 Vrijeme mirovanja - funkcija idle_time 20
Slika 15 Ispis nakon pokretanja idle_time funkcije 20
Slika 16 Histogram ispisa metrike 21
Slika 17 Vrijeme procesiranja vizualizacija 21
Slika 18 Vrijeme propusnosti vizualizacija 22
Slika 19 XES datoteka 25
Slika 20 Funkcija za čitanje xes datoteke 25
Slika 21 Čitanje xes datoteke pomoću RStudio 26
Slika 22 Funkcija za kreiranje .xes datoteke 26
Slika 23 Izvođenje funkcije write_xes 26
Slika 24 Prikaz umjetnih podataka – eventdataR 27
Slika 25 Podaci iz stvarnog života – eventdataR 28
Slika 26 Učitavanje library(bupaR) i naredbe process_map() 29
Slika 27 Vizualizacija procesa za pacijente 29
Slika 28 Vizualizacija procesa uz korištenje relative frequency 30
Slika 29 Vizualizacija procesa korištenjem profila izvedbe 30
Slika 30 Vizualizacija procesa korištenjem prilagođenog profila 31
Slika 31 Vizualizacija procesa korištenjem kombiniranih različitih profila 31
Slika 32 Instalacija paketa potrebnih za primjer procesne mape 32
Slika 33 Učitavanje podataka 33
Slika 34 Korištenje funkcije POSIXct 33

Slika 35 Kreiranje events-a....34
Slika 36 Zapisnik događaja....35
Slika 37 Sažetak podataka....36
Slika 39 Kreiranje procesne mape....36
Slika 38 Histogram podataka....36
Slika 40 Procesna mapa....37

Tablica:

Tablica 1 Prikaz dodatnih varijabli i primjer 35

7. Literatura

1. Ihaka, R., „R: Past and Future History“, A Draft of a Paper for Interface '98, https://www.stat.auckland.ac.nz/~ihaka/downloads/Interface98.pdf , (12.05.2019.)

2. Thieme, N., „R generation – The story of a statistical programming language that became a subcultural phenomenon“, 11 August, 2018., https://statfr.blogspot.com/2018/08/r-generation-story-of-statistical.html , (12.05.2019.)

3. Tutorialspoint, „Learn R Programming“, https://www.tutorialspoint.com/r/r_overview.htm ,(12.05.2019.)

4. „R – PROS AND CONS“, 13 May, 2017., https://economictheoryblog.com/2017/05/13/r-pros-and-cons/ ,(13.05.2019.)

5. The R Base Package, https://rdrr.io/r/base/base-package.html , (13.05.2019.)

6. RStartHere, 29 April, 2016. https://github.com/rstudio/RStartHere/blob/master/top_downloads_2016/top_packages , (18.05.2019.)

7. Pylvainen, I., „RStudio Support“, 10 May, 2019., https://support.rstudio.com/hc/en-us/articles/217799198-What-is-the-difference-between-RStudio-Desktop-and-RStudio-Server- ,(18.05.2019.)

8. The Extensive Guide to Business Processes(2018.), https://kissflow.com/bpm/business-process/ ,(29.05.2019.)

9. Enterprise architect's guide to optimal BPM workflow(2018.), https://searchcio.techtarget.com/definition/business-process ,(29.05.2019.)

10. Analiza poslovnih procesa(2017.), https://www.bizpro.hr/usluge/analiza-poslovnih-procesa/ ,(29.05.2019.)

11. The Basics of Documenting and Analyzing Your As-Is Process(2018.), https://www.lucidchart.com/blog/as-is-process-analysis ,(29.05.2019.)

12. Understand and apply the business process analysis methodology(2015.), https://www.heflo.com/blog/process-mapping/business-process-analysis-methodology/ ,(29.05.2019.)

13. Janssenswillen, G. „Filters“, 26 June, 2018., http://gertjanssenswillen.github.io/edeaR/articles/filters.html ,(06.07.2019.)

14. Janssenswillen, G. „Metrics“, 22 February, 2019., https://cran.r-project.org/web/packages/edeaR/vignettes/metrics.html ,(06.07.2019.)

15. „Read XES-files“, https://www.bupar.net/read_xes.html ,(06.07.2019.)

16. „Write XES-files“, http://bupar.net/write_xes.html ,(06.07.2019.)

17. xesreadR documentation, „Write XES-files“, 02 May, 2019., https://rdrr.io/cran/xesreadR/man/write_xes.html ,(06.07.2019.)

18. „processmonitR“, https://www.bupar.net/processmonitR.html (07.07.2019.)

19. Janssenwillen, G., „bupaR: Business Process Analysis in R“ https://pdfs.semanticscholar.org/1bdb/a1b21dba26929da12f36cf2581e730f04872.pdf?_ga=2.70688528.1850292205.1560798583-2120704751.1559674133 (07.07.2019.)

20. Mannhardt, F., „processanimateR“, https://fmannhardt.github.io/processanimateR/index.html (07.07.2019.)

21. Janssenwillen, G., „bupaR: Business Process Analysis in R“, 29 September, 2017., https://www.r-bloggers.com/bupar-business-process-analysis-with-r/ (07.07.2019.)

22. Gour, R., G., „A Complete Guide to Learn R“, 25 April, 2019., https://towardsdatascience.com/a-complete-guide-to-learn-r-29e691c61d1 (08.07.2019.)

23. Hernaus, T., Kovačić, A. i Bosilj-Vukšić, V. (2008.) *Upravljanje poslovnim procesima organizacijski i informacijski pristup.* Zagreb: Školska knjiha, d.d.

24. Kdnuggets, 2019., https://www.kdnuggets.com/2017/11/process-mining-r-introduction.html (10.07.2019.)

Izvor podataka za primjer:

1. Dataset iz BPI challange-a s Kaggle.com 2017. godine, https://raw.githubusercontent.com/scheithauer/processmining-bupaR/master/00-data/loanapplicationfile.csv?fbclid=IwAR1ozRZXEK0_EBZ9-sWocyMFePM7jYMblb633aDt0WrqQRtVcB3MH_OeQC0 (06.07.2019.)
2. Data Pre Processinf Techniques You Should Know, 02 Dec, 2018., https://towardsdatascience.com/data-pre-processing-techniques-you-should-know-8954662716d6 (10.07.2019.)
3. Creating event logs, https://www.bupar.net/creating_eventlogs.html (10.07.2019.)
4. Scheithauer,G., Henne, R., Kerciku, A., Waldenmaier, R., i Riedel, U., Suggestions for Improving a Bank's Loan Application Process based on a Process Mining Analysis, https://www.win.tue.nl/bpi/lib/exe/fetch.php?media=2017%3Abpi2017_paper_19.pdf&source=post_page---------- (22.07.2019.)