Poslovna analitika u sustavu SQL Server 2019

Kraljić, Fabijan Josip

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Fabijan Josip Kraljić

Poslovna analitika u sustavu MS SQL Server 2019

DIPLOMSKI RAD

Varaždin, 2020.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Fabijan Josip Kraljić

Matični broj: 0016116913

Studij: Baze podataka i baze znanja

Poslovna analitika u sustavu MS SQL Server 2019

DIPLOMSKI RAD

Mentor:

Prof. dr. sc. Kornelije Rabuzin

Varaždin, rujan 2020.

Fabijan Josip Kraljić

Izjava o izvornosti

Izjavljujem da je moj diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

_____

Sažetak

U ovom radu će biti riječi o poslovnoj inteligenciji kao i pripadajućoj znanosti zvanoj poslovna analitika koja je u današnjem svijetu neophodna za donošenje pravovremenih i ispravnih poslovnih odluka. Također su pojašnjeni pojmovi usko vezani za poslovnu analitiku, kao što su rudarenje podacima (engl. *data mining*), skladište podataka (engl. *data warehouse*), Big Data kao i neki statistički pojmovi, odnosno tehnike i metode koje se koriste za izvlačenje korisnih informacija iz podataka. Drugi dio rada biti će posvećen praktičnoj primjeni teorije gdje su uzeti stvarni podaci koji će se obraditi uz pomoć sustava MS SQL Server 2019 uz dodatak R jezika. Dobiveni podaci daju određene informacije koje mogu poslužiti u donošenju poslovnih odluka. Radi lakšeg zaključivanja i donošenja odluka, brojčani podaci prikazani su grafički uz pomoć R jezika.

Ključne riječi: Poslovna analitika, Poslovna inteligencija, Big Data, Data warehouse, Data Mining, MS SQL Server 2019, R jezik, Strojno učenje

Sadržaj

1. Uvod

Podaci se nalaze svugdje oko nas. Čovjek od kada je znao pisati je zapisivao podatke, bili to podaci o pripremanju hrane, količine hrane, stvaranju lijekova, bilježenju događaja, broja stanovništva, vojske i sličnih stvari koje je on smatrao korisnim. Takve podatke je zapisivao na određena mjesta kako bi njima mogao pristupiti po potrebi i kako bi iduće generacije imale koristi od njih. Budući da su ti zapisi bili jedinstveni, čovjek je nastojao iste te podatke kopirati kako bi bili dostupni drugima i zbog toga što je smatrao kako su od velike važnosti. Za takvo što je bilo potrebno izdvojiti puno vremena i strpljenja. Kakve sad to veze ima sa Poslovnom analitikom?

Odgovor su podaci, odnosno informacije. Podaci se nalaze svugdje oko nas te su neizostavan dio naših života, kao aktivni korisnici interneta svakodnevno ostavljamo svoj digitalni trag koji može biti nekome od koristi. Bitno je napomenuti kako nepromišljeno prikupljanje podataka nije dovoljno, dapače, troši skupe resurse koji bi se mogli koristiti na drugi način. Podaci se trebaju pametno strukturirati, analizirati i koristiti kako bi od njih bilo neke koristi. Tako npr. nekom poduzeću možda i nije toliko bitno zašto su izgubili pokojeg klijenta te takvo poduzeće ne treba prikupljati i analizirati podatke svojih klijenata. Problem, tj. prilika se javlja kada konkurencija daje veliku važnost zadovoljstvu svojih klijenata, tada takvo poduzeće ima značajnu prednost te lakše može pratiti korak s ostalima.

U takvim slučajevima možemo reći da su podaci za poduzeće „zlata vrijedni“ i opstanak takvih poduzeća gotovo da ovisi isključivo o njima. Podaci koji se mogu pohraniti mogu biti raznoliki kao npr. industrijski trendovi, zadovoljstvo kupaca, povijesni događaji, proizvodi, usluge i sve ostalo što bi moglo biti od značaja nekoj organizaciji. Takvi podaci se danas automatski bilježe bez ljudske intervencije na razne računalne medije uz pomoć novih i naprednih tehnologija. Zabilježene podatke potrebno je staviti u određeni kontekst te je iz njih potrebno izvući korisne zaključke na temelju kojih se mogu donositi odluke. U nekim slučajevima se zna što se traži dok se u drugima može doći do korisnih informacija na sasvim slučajan i neočekivan način. Takve poslove danas obavljaju statističari i analitičari uz pomoć različitih alata, a o nekima će biti više riječi u nastavku ovog rada.

Gore navedeni poslovi se mogu sažeti u jedan pojam, a to je poslovna analitika (engl. *Business analytics*). Zanimljivu izjavu u vezi poslovne analitike je dao glavni ekonomist tvrtke Google, Hal Varian koji kaže sljedeće „*Stalno ponavljam da će u idućih 10 godina posao statističara biti vrlo privlačan i pritom se ne zafrkavam.*“. Njegovu izjavu podupire tržište rada gdje u SAD-u plaće statističara s doktoratom počinju od 125 000 dolara, a s vremenom sve više raste. (Winston, 2015)

Upravo će sve navedeno biti tema ovog diplomskog rada, a i više, gdje će se kroz razne aspekte i poglede dočarati važnost poslovne analitike. Iznad je navedeno kako se podaci nalaze svugdje oko nas i kako svakodnevno ostavljamo u svijetu svoj digitalni trag, te dvije činjenice zapravo predstavljaju termine *Veliki podaci* (*engl. Big Data*) i *Internet stvari* (*engl. Internet of Things*) koji će ovdje biti pobliže prikazani. Također je navedeno da se na temelju prikupljenih i analiziranih podataka donose poslovne odluke, a takvo nešto je predstavljeno terminom *Poslovna inteligencija* (*engl. Bussines inteligence*).

Time će se obuhvatiti teorijska podloga koja je potrebna za shvaćanje praktičnog dijela u kojemu će se za primjer uzeti nedavni događaji u SAD-u, gdje je glavna tema rasizam i nasilje policijskih službenika nad određenom skupinom ljudi. Za provedbu praktičnog dijela koristiti će se razne tehnologije i tehnike, gdje su najznačajnije MS SQL Server 2019 i R jezik. Cilj ovakvog primjera je prikazivanje važnosti poslovne analitike, kakve se korisne informacije mogu izvući iz podataka, tj. neočekivana saznanja i što se pomoću njih može postići.

2. Poslovna inteligencija

Kao što je prethodno navedeno u ovome poglavlju govoriti će se o *Poslovnoj inteligenciji* čije je razumijevanje neophodno za shvaćanje *Poslovne analitike* te su ta dva pojma usko vezana i teško je odrediti razlike među njima. Bitno je znati kako *Poslovna inteligencija* predstavlja samo jedan aspekt *Poslovne* analitike, no time ćemo se pozabaviti u nastavku. Osim što će se objasniti pojam *Poslovne inteligencije* također će biti razrađen termin *Skladišta podataka* koje je potrebno u nekakvom obliku za statističku obradu podataka. Pije nego što se obradi termin *Poslovne inteligencije* potrebno je razraditi pojam *Potpora upravljanja*.

2.1. Potpora upravljanja

Ponekad upravljanje organizacijom može biti mukotrpni i iscrpljujući posao, a da bi se takvo nešto postiglo koriste se upravljački procesi kojima se *analizira*, *planira*, *organizira* i *kontrolira* funkcionalnost i efikasnost poslovnog sustava. Upravljanje organizacijom se zapravo svodi na pravodobno i ispravno donošenje odluka[1], a da bi se one mogle donijeti potrebno je koristiti kvalitetne informacije. (Mladen Varga, 2016, str. 61)

Donositelji odluka često se nalaze u nepogodnim okolnostima, bilo to do nedostatka vremena, znanja ili dostupnih mogućnosti. Stoga se oslanjaju na informacijske sustave koji bi im trebali pomoći prilikom identifikacije, pronalaženju i analizi potrebnih podataka odnosno informacija na temelju kojih mogu donijeti pravovremene i ispravne odluke značajne organizaciji u kojima posluju.

Donošenje odluka odnosno odlučivanje „*je proces koji se obavlja kroz nekoliko koraka odnosno faza: (a) izviđanje odnosno prepoznavanje problema, (b) oblikovanje modela rješenja, (c) izbor rješenja i (d) implementacija rješenja.*“ (Mladen Varga, 2016, str. 62)

S druge strane, *potpora upravljanja* temelji se na pronalaženju i pripremi podataka relevantnih za problem o kojemu se odlučuje i na analizi podataka primjenom odgovarajuće metode obrade kojom će se dobiti informacije potrebne u procesu odlučivanja. (Mladen Varga, 2016, str. 62)

[1] Odluka – je rezultat mentalnog procesa koji rezultira izborom najpovoljnije inačice rješenja između više njih. Proces odlučivanja može biti više li manje racionalan, ali i intuitivan. (Mladen Varga, 2016)

U nastavku će se razmotriti razni načini upravljanja poslovnim sustavom, koliko oni ovise o podacima te kako oni, odnosno pripadajuće osobe koriste dostupne podatke i informacije u svoju korist.

2.1.1. Operativno upravljanje

Operativnim upravljanjem (odlučivanjem) bave se menadžeri niže razine kojima je to svakodnevni posao unutar organizacija, a često su to odluke koje određuju *taktički* menadžeri. Ne trebaju to biti samo ljudi koji donose odluke, mogu biti i računalni sustavi, pogotovo one odluke koje su repetitivnog tipa, imaju zacrtane smjernice i pravila (procedure, algoritme, ...). (Mladen Varga, 2016, str. 62)

Ljudi se svakodnevno susreću s time, npr. na Zavodu za zapošljavanje se za osobu unose osobni podaci, kakvo znanje posjeduju, gdje su sve radili, kakav posao traže i sl. Nadalje, informacijski sustav na temelju dobivenih i njemu dostupnih podataka treba odrediti koji bi poslovi bili idealni za kandidata. Radnicima u Zavodu za zapošljavanje je time značajno olakšan posao gdje tako bolje i brže mogu suziti izbor mogućih poslova, kako bi kandidatu preporučili prikladan posao. Drugi primjer bi bio isplata traženog iznosa na bankomatu. Bankomat, odnosno banka ima zadane smjernice za isplatu određenog iznosa klijentu, koje se mogu ponešto razlikovati od klijenta do klijenta (npr. dnevni limit), tada sustav banke za klijenta treba odrediti smije li traženi iznos isplatiti klijentu.

2.1.2. Taktičko upravljanje

Taktičkim upravljanjem bave se menadžeri srednjih razina koji mogu pratiti i do nekoliko različitih procesa, a ne samo jedan. Jedan takav primjer je praćenje prodaje određenih dobara za neko razdoblje na nekom području. Na sličan način funkcionira prognoziranje vremena, prate se različiti parametri koji su ključni za dobivanje valjane prognoze vremena. Takvi poslovi su polustrukturirani, zna se što se ustvari radi i traži no za takvo nešto ne postoje predefinirani koraci, odnosno procedure i algoritmi. Potrebna je određena razina iskustva, stručnog znanja i potrebno je imati širu sliku samog okruženja, a sustav može do određene razine pomoći (Mladen Varga, 2016, str. 63)

Može se zaključiti kako je potrebno značajno više informacija iz više izvora nego što je to kod *Operativnog upravljanja*. Potrebne su informacije iz različitih izvora za određeni period, općenito su potrebni značajno detaljniji i sveobuhvatniji podaci. Analitičar tako postavlja određene upite u bazu podataka ili skladište podataka putem informacijskog sustava gdje dobivene informacije može analizirati i konačno dati nekakav odgovor. U protivnome, ponavljaju se prethodni postupci gdje takvo nešto nazivamo *analitička obrada podataka* (engl. *On-line Analytical Processing*, OLAP). O navedenom pojmu će biti nešto više riječi kasnije.

2.1.3. Strateško upravljanje

Dolazimo do zadnjeg pojma vezan za *Potporu upravljanja*, a to je *Strateško upravljanje* kojim se bave gotovo isključivo osobe na najvišim razinama u organizacijama. Takve osobe donose dugoročne odluke te se oslanjaju na veliki set informacija koje im pruži informacijski sustav. Za takvo nešto ne postoje nikakva propisana pravila, procedure i algoritmi te se osoba oslanja isključivo na dobivene informacije i na svoj um. (Mladen Varga, 2016, str. 63)

U takvim slučajevima se koriste značajno naprednije metode obrade podataka zvane *dubinska analiza podataka* koje pripadaju područjima statistike, matematike te u novije vrijeme umjetnoj inteligenciji. Jedna takva metoda se zove *otkrivanje znanja iz podataka*, a postoje i razne druge metode koje djeluju nad *velikim podacima* (eng. *Big Data*).

2.2. Definicija poslovne inteligencije

Sve prethodno opisane metode odnosno koncepti zapravo su predstavljeni pojmom *Poslovna inteligencija* ili bolje poznato pod nazivom *Bussines Intelligence* (BI). *Poslovna inteligencija* nije nikakav novi koncept, već je evolucija prijašnjih pristupa, metoda i tehnika korištenja podataka, odnosno informacija radi boljeg razumijevanja poslovnih procesa na temelju kojeg se mogu donositi odluke. Za *Poslovnu inteligenciju* se može reći da je ono ustvari pojam koji opisuje nove, naprednije informacijske sustave za zrelije odlučivanje i upravljanje poslovnim procesima koji su danas sve kompleksniji i dinamičniji. Konkretna priznata definicija ne postoji, no jedna definicija koja zorno opisuje Poslovnu inteligenciju je:

> *„Business intelligence (poslovno-obavještajna aktivnost) je obavještajna aktivnost u poslovnom svijetu koju planiraju, organiziraju i provode poslovni subjekti, pri čemu ta aktivnost podrazumijeva proces legalnog prikupljanja javnih i svima dostupnih podataka etičnim sredstvima, njihovu analizu i pretvaranje u gotove poslovno-obavještajne analize („znanje“) radi pružanja potpore čelništvu poslovnog subjekta s ciljem donošenja i realizacije što kvalitetnijih poslovnih odluka usmjerenih na očuvanje postojeće pozicije poslovnog subjekta u poslovnom okruženju, izbjegavanje bilo kakvih prijetnji i u konačnici na ukupni kvalitativni napredak poslovnog subjekta.“* (Božidar Javorović, 2007, str. 205)

Na kraju krajeva, sama definicija nije ni toliko bitna, nego je bitan sam učinak koji se dobiva iza danog termina. Prije nekoliko godina se korištenjem informacijskog sustava moglo čekati i nekoliko dana kako bi se dobio određeni izvještaj za neki složeni upit, dok danas postoji veliki broj sustava koji može odmah dati odgovor na složenija pitanja.

3. Skladište podataka

Idući pojam koji je vrlo bitan za razumijevanje *Poslovne analitike* je pojam *Skladišta podataka*. Pojmovi *Poslovne analitike* i *Poslovne inteligencije* direktno se vežu za skup podataka, a oni se trebaju negdje čuvati, odnosno pohraniti kako bi se njima moglo po potrebi pristupiti. Upravo tu uskače *Skladište podataka* koje se brine o pohrani ali i lakšem pristupu velikoj količini podataka koje je neophodno za donošenje bilokakvih odluka. U nastavku će biti nešto više riječi o tome što je to *Skladište podataka*, kako je građeno, iz kakvih se izvora puni skladište i sl.

3.1. Definicija skladišta podataka

Prije svega, *Skladište podataka* je najbolje poznato pod engleskim nazivom *Data Warehouse* ili ponekad *Data Mart* [2]. Za razliku od *Poslovne inteligencije*, stručnjaci se slažu tko je otac pojma i definicije *Skladišta podataka*, a to je William H. Inmon. Prema Inmonu definicija *Skladišta podataka* je sljedeća:

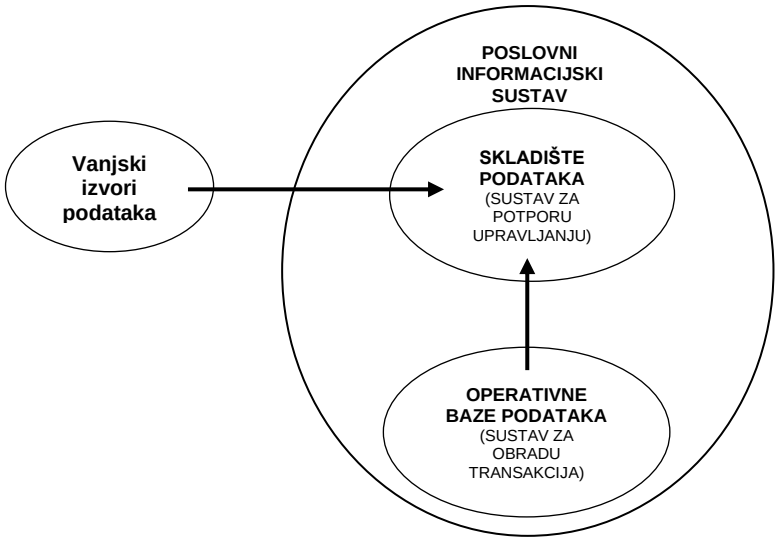
> „*Skladište podataka je području orijentiran, integriran, strukturom nepromjenjiv i vremenski dinamičan skup podataka namijenjen za potporu odlučivanja.*“ (Inmon, 1992)

Takva definicija je opće prihvaćena, a na njoj se temelje sve druge definicije do danas. U daljnjem dijelu teksta će biti pojašnjen svaki pojedini dio definicije i na što se odnosi.

- Području orijentiran – skladište podataka je orijentirano na glavna područja, odnosno procese u sustavu
- Integriran – podaci u skladištu podatka su prikupljeni sa nekoliko izvora te unutar nje tvore konzistentnu cjelinu. (Comp.nus.edu)
- Strukturno nepromjenjiv – podaci se unutar skladišta ne mijenjaju, povremeno (često u intervalima) se dodaju novi podaci, ali gotovo nikada se ne brišu.
- Vremenski dinamično – svim podacima unutar skladišta se pridodaje vremenska komponenta. (Comp.nus.edu)

[2] *Data Mart* – je manji oblik *Data Warehouse*-a, gdje je fokus na jedan subjekt te ima samo nekoliko izvora podataka. Često *Data Warehouse* uz ostale izvore crpi svoje podatke iz nekoliko *Data Mart*-ova. (Rabuzin, 2020.)

Postoji veliki broj drugih definicija istog termina, a prema Mladenu Vargi i Ivanu Strugaru *Skladište podataka „predstavlja izdvojeni, veliki skup podataka koji je oblikom i sadržajem pripremljen za zahtjevne analize, kako bi se iz njega izvukle informacije i znanje za potrebe odlučivanja i upravljanja*“. (Mladen Varga, 2016, str. 65) Podaci iz *Skladišta podataka* dolaze iz operativnih baza podataka, većinom transakcijskog tipa, koje sadrže podatke bitne za dnevno poslovanje. Svaka pojedinačna transakcija sadrži određeni skup podataka koji opisuje transakciju, a u skladište ulaze samo oni bitni. Potrebno je odrediti koji su to bitni podaci, ekstrahirati i transformirati te unijeti u skladište, no o tome će biti nešto više riječi kasnije.



Slika 1: Skladište podataka kao dio informacijskog sustava (Mladen Varga, 2016, str. 66)

Navedeni koraci predstavljaju termin ETL o kojem će se govoriti u nastavku, također će biti nešto više riječi o tome kako se gradi skladište podataka. Također slijedi i opis izvora podataka te kako oni mogu biti strukturirani.

3.1.1. Građa skladišta podataka i ETL

Građa *Skladišta podataka* nije previše kompleksna, no zato su to procesi koji obrađuju podatke. Imamo dvije razine, same podatke i mehanizme manipulacije nad podacima. Podaci mogu biti osnovni i agregirani u slučaju da se radi o vrlo velikom broju podataka (višedimenzionalni podaci), a mehanizme manipulacije čine procesi ekstrakcije, transformacije i unosa (engl. ETL). (Mladen Varga, 2016) Takozvani ETL postupak je potreban iz razloga jer se podaci u skladište slijevaju iz različitih izvora i nisu uvijek istog formata, ispravni, potpuni ili nisu svi podaci potrebni, stoga je potrebno bitne podatke ekstrahirati, transformirati, odnosno pravilno oblikovati i zatim pohraniti u *Skladište podataka*.

3.1.2. Izvori podataka

Kao što je rečeno, svijet je okružen podacima te se svaki naš korak, svaka naša akcija u digitalnom svijetu bilježi. Cilj toga je da organizacije koje prikupljaju takve podatke iz njih mogu izvući korisne informacije kako bi mogle pratiti korak s konkurencijom i eventualno postale lideri u svome području.

U stvarnome svijetu broj, veličina i kakvoća podataka ovisi o samoj djelatnosti s kojom se određena organizacija bavi, tj. što je veća organizacija to je veća vjerojatnost da imaju veći broj izvora podataka, pogotovo ako je organizacija sačinjena integracijom više manjih organizacija. Također se za izvor sve češće uzimaju polustrukturirani pa čak i nestrukturirani podaci kao npr. Excel, Word, HTML dokumenti i drugi, no to vrlo rijetko.

Svi ti izvori podataka se mogu svesti u dvije grupe, a to su unutarnji i vanjski izvori koji opskrbljuju skladište podataka, a iz pogleda poduzeća unutarnji izvori podataka su sljedeći (Mladen Varga, 2016, str. 69):

- Sustav za obradu transakcija
    - Financijski podsustavi
    - Logistički podsustavi
    - Podsustavi prodaje
    - Podsustavi proizvodnje
    - Podsustavi ljudske proizvodnje
- Sustav za potporu upravljanja (planovi, odluke, pravilnici, itd.)
- Sustav za potporu komunikaciji i suradnji (email i sl.)

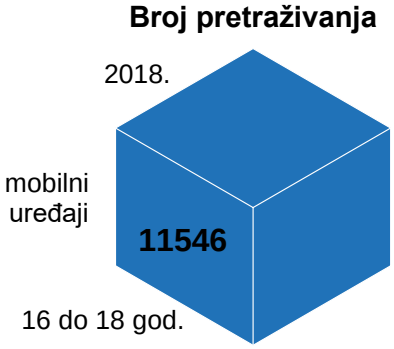
Što se tiče vanjskih izvora podataka kao što i samo ime govori, oni se nalaze izvan poduzeća, a oni su od izuzetne važnosti kako bi se poduzeće prilagodilo svojoj okolini. Mogu se prilagoditi konkurenciji kao i potencijalnim prilikama i prijetnjama. Vanjski izvori se mogu grupirati na sljedeći način (Mladen Varga, 2016, str. 69):

- Podaci o konkurentnosti (proizvodi, usluge, itd.)
- Ekonomski podaci (burzovni podaci, podaci valute, itd.)
- Strukovni podaci (tehnološki i marketinški trendovi)
- Robni podaci (cijene sirovina, poluproizvoda, ...)
- Ekonometrijski podaci (ponašanje kupaca/klijenata)
- Meteorološki podaci
- Psihometrijski podaci (profiliranje kupaca/klijenata)
- Marketinški podaci i
- Podaci o novim proizvodima

3.1.3. Dimenzijska struktura podataka

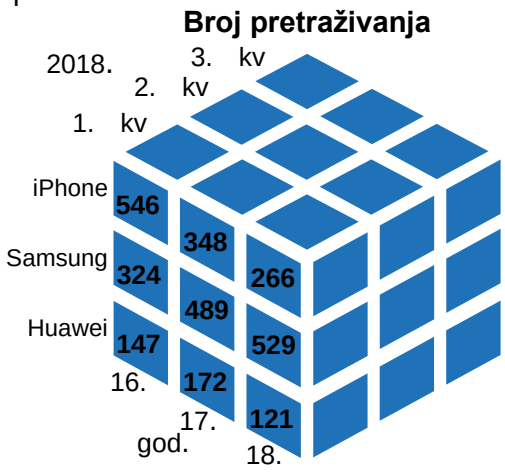
U skladištu podataka, podaci se mogu zamisliti kao dimenzije kocke, gdje svaka strana kocke predstavlja određenu vrstu podataka. Takav pristup se koristi radi lakše vizualizacije i razumijevanja same ovisnosti pojedinih podataka, bolje nego u dvije dimenzije. (Mladen Varga, 2016) U nastavku će biti prikazan jedan takav primjer.

Neko poduzeće koje posjeduje internetsku stranicu (npr. web prodaje) se pita sljedeće, „*Koliki broj ljudi u dobi od 16 do 20 godina je u 2018. godini pretraživalo, odnosno tražilo mobilne uređaje?*“. Poduzeću bi takva informacija bila korisna da sazna što je sada u trendu i što bi trebali imati u ponudi kako bi omogućili veću potencijalnu zaradu.



Slika 2: Prikaz višedimenzionalnosti podataka 1/2 (vlastita izrada)

Određeni podaci se mogu hijerarhijski prikazati, odnosno prikazati s još većom razinom zrnatosti. Tako npr. godine može prikazati kroz kvartale, mjesece i dane, mobilne uređaje prikazuje po markama ili modelima, a dobne skupine po točno određenim godinama. Na taj način dobijemo sljedeći prikaz.



Slika 3: Prikaz višedimenzionalnosti podataka 2/2 (vlastita izrada)

Naziv korištene tehnike je *drill-down*, a postoje i neke druge (*roll-up*, *slice*, *dice* i *pivoting*). Kod skladišta podataka i dimenzijskih struktura podataka razlikujemo činjenice i mjere, gdje su činjenice rezultati mjerenja kojima evaluiramo procese, a mjere izražavaju neku statiku (npr. količina zaliha, broj klikova, itd.). (Rabuzin, 2020)

4. Big Data

U idućem dijelu teksta slijedi opis pojma *veliki podaci* ili popularno zvano *Big Data*. U novije vrijeme je količina podataka toliko narasla da ih se više ne može pohranjivati klasičnim relacijskim bazama podataka. U te svrhe, potrebna su alternativna rješenja koja se svrstavaju pod pojmom *Big Data*. U nastavku teksta slijedi opširniji opis navedenog pojma.

4.1. Definicija Big Data

Potrebno je poznavati pojam *veliki podaci* budući da se ono koristi kod vrlo naprednih analitičkih tehnika gdje su u „igri“ goleme količine strukturiranih, polustrukturiranih i nestrukturiranih podataka s različitih izvora. Prema Gartneru definicija za pojam *veliki podaci* je sljedeća:

> *„Veliki podaci su velika količina, kontinuirano prikupljenih i/ili raznolikih informacijskih sredstava koje zahtijevaju cjenovno isplative i inovativne načine obrade koje omogućuju bolji uvid u promatrano područje, poslovno odlučivanje te automatizaciju poslovnih procesa.“* (Gartner, 2020.)

Definicija je poznata još kao *3V* definicija budući da izvorni zapis sadrži tri pojma, *volume, velocity* i *variety* (prijevod *volumen, brzina* i *raznolikost*) (Stepinac, 2020).

- ***Volume*** – velika količina podataka koja se prikuplja i stavlja na raspolaganje za analizu
- ***Velocity*** – kontinuirano prikupljanje velike količine podataka u realnom vremenu
- ***Variety*** – podaci su dostupni u različitim oblicima i izvorima, a zapravo su najčešće nestrukturirani

4.2. Povijest Big Data

Pojam *Big Data*, odnosno *Velikih podataka* je nešto što se smatra modernim i novim te ga gotovo svi veći „igrači“ implementiraju u nekakvom obliku, no to ustvari nije istina kako se radi o novoj tehnologiji. Rooseveltova administracija u SAD-u je koristila ogromne količine podataka kako bi mogla pratiti kome je sve potrebno isplaćivati mirovinu, tko sve ima neku vrstu beneficije, tko ima potporu tokom nezaposlenosti i sl. Također ima raznih drugih primjera kao što je npr. slučaj s dešifriranjem enigme gdje su se također koristile velike količine podataka koje su analizirane kako bi se došlo do odgovora. (DeWitt, 2020.)

Pojam *Big Data* i podatkovne analize kakav danas poznajemo javlja se u 60-tim godinama prošlog stoljeća. Tako je npr. Američka vlada odlučila napraviti sustav za vođenje brige o povratu poreza kao i otiscima prstiju cjelokupnog stanovništva SAD-a što je poprilično veliki broj ljudi kao i broj podataka koji je potrebno pohraniti. Zadnje bitne obrise *Big Data* i analiza podataka dobiva početkom 90-ih godina, začecima interneta kojim su se dnevno počele kretati ogromne količine podataka. (Zhang, 2017)

Termin *Big Data* se javlja tek 2005. godine, kojeg je spomenuo Roger Mougalas koji zapravo opisuje situaciju u kojoj se tradicionalni alati i tehnike ne mogu nositi s velikom količinom podataka. Upravo te 2005. godine počinju naglo rasti poduzeća poput Facebook-a, Google-a, YouTube-a i drugi. Te iste godine je stvoren Hadoop okvir (engl. *framework*) za indeksiranje interneta i analizu podataka koji je prije nekoliko godina postao pristupačan manjim poduzećima. (Zhang, 2017) (Oracle, 2020.)

Još jedan zanimljiv podatak je sljedeći, čovjek je od svojih početaka pa sve do 2003. godine prikupio jednaku količinu podataka kao što se to prikupi u dva dana u 2010. godini (oko 10 exabajta), a 10 godina kasnije je značajno premašena i ta brojka, pogotovo korištenjem IoT[3] tehnologijama. Možemo reći da je u današnjem svijetu najvrjednija roba podatak, odnosno podaci i informacije, a onaj koji njima zna dobro upravljati ima veliku moć. (Zhang, 2017) (Oracle, 2020.)

4.3. Big Data u primjeni

Dosad je bilo riječi o teoriji iza termina *Big Data* no bitna je sama primjena i korist koja se dobije iz njega. U nastavku slijede neki od mogućih primjena *Big Data* uz navođenje kratkih praktičnih primjera koji upotpunjuju sliku o navedenom pojmu.

- **Razvoj proizvoda** - Poduzeća poput Netflix-a i Procter & Gamble (kratko P&G, vlasnici Hugo Boss, Braun, Duracell, Gillette, Oral-B, Puma, itd.) koriste velike podatke kako bi predvidjeli potražnju svojih korisnika. Izrađuju razne prediktivne modele za svoje nove proizvode koje žele plasirati na tržište, gdje uzimaju razne atribute svih prijašnjih i trenutnih proizvoda kako bi ustanovili da li ima smisla uvoditi novi proizvod, kako ga uvesti, gdje uvesti itd. Kompanija P&G za dobivanje odgovora prilikom prikupljanja i analize podataka koristi razne izvore poput društvenih mreža, testnih uvođenja proizvoda, profiliranje korisnika kako bi mogli planirati, proizvoditi i izbaciti novi proizvod na tržište. (Oracle, 2020.)

[3] IoT – eng. *Internet of Things* je opisano u cjelini *4.7.*

- **Prediktivno održavanje** - Ljudi su okruženi raznim tehničkim tvorevinama, a neke od njih su od iznimne važnosti za očuvanje ljudskih života. U takvim slučajevima bitno je poznavati strukturu, godinu proizvodnje, modela takvih proizvoda. Podaci o tome se mogu nalaziti u strukturiranim izvorima, ali također i u nestrukturiranim izvorima kao što su to podaci senzora, povijesni podaci kvarova i grešaka, log podataka, itd. Takvi podaci prilikom njihove analize mogu dati veliki broj bitnih informacija, npr. kolika je dugovječnost i pouzdanost proizvoda, kako i kada vršiti servis i slično kako bi se izbjegle nesreće ili gubici prilikom zaustavljanja rada nekog sustava. (Oracle, 2020.)
- **Korisničko iskustvo** - Kao što je dosad rečeno, svaki naš korak u digitalnom svijetu se prati, prikupljaju se razni podaci bilo sa društvenih mreža, raznih pretraživanja interneta i internet kupovina, poziva itd. Ti svi podaci su „zlata vrijedni" za poduzeća jer onda mogu skrojiti točno određene proizvode/usluge za svoje kupce/klijente i time maksimizirati moguću zaradu, ali i zadovoljstvo kupaca/klijenata. (Oracle, 2020.)
- **Otkrivanje prevara** - U današnjem svijetu se svakodnevno vrši veliki broj novčanih transakcija, a ukoliko se ne nadziru sustavi i pripadajuće transakcije lako može doći do krađa, nezakonite dobiti i slično. U takvim slučajevima koriste se prediktivne metode za predviđanje mogućih točaka prevare te je takvo nešto vrlo kompleksno za realizirati. Kasnije će u tekstu biti riječi o prediktivnoj analitici. Računalni sustavi su puno bolji i brži u otkrivanju prevara nego ljudi budući da oni bolje uočavaju nelogičnosti i uzorke stoga sve veće organizacije imaju neku vrstu sustava implementiranog za otkrivanje i prepoznavanje prevara. (Oracle, 2020.) (Zhang, 2017, str. 128)
- **Strojno učenje**[4] - (engl. *Machine learning*) grana IT industrije koja naglo raste i koja pronalazi sve više mjesta u primjeni, a upravo je *Big Data* jedan od kotačića koji omogućuje razvoj te grane. Tako npr. za stvaranje vremenske prognoze potrebna je značajna količina podataka kao i obradbena moć kako bi se dobili relevantni podaci, u takvim slučajevima se može koristiti strojno učenje koje može postići bolje i točnije rezultate uz manji napor. (Oracle, 2020.)
- **Operativna učinkovitost** – točka primjene kojoj se pridaje najmanje pozornosti, a bavi se poslovnim procesima gdje *Big Data* ima najveći utjecaj. Uz pomoć *Big Data* moguće analizirati i procijeniti proizvodnju, prodaju, korisničko zadovoljstvo i sl. kako bi se umanjila mogućnost nestašica i kako bi se mogle prognozirati buduće potražnje. (Oracle, 2020.) Tako npr. avionske kompanije koriste analitiku s *velikim podacima* kako bi odredili koliko karata za određenu cijenu trebaju prodati da bi ostvarili profit, slično tako i Hotelski lanci, budući da su marže u takvim poslovima dosta male. (Zhang, 2017)

[4] Strojno učenje – „Strojno učenje grana je umjetne inteligencije koja se bave oblikovanjem algoritama koji svoju učinkovitost poboljšavaju na temelju empirijskih podataka" (fer.unizg.hr, 2020.)

4.4. Funkcioniranje Big Data

Big data je vrlo koristan u poslovnome svijetu te onaj tko ima priliku iskoristiti njene značajke to i radi, pruža nova saznanja i drugačiji pogled na okruženje. Big Data je poprilično sličan skladištima podataka, no postoje neke određene razlike odnosno akcije o kojima se u nastavku govori.

4.4.1. Integracija

U cjelini *Građa skladišta podataka i ETL* je bilo riječi o ETL procesima, kako postoji veliki broj mogućih izvora podataka te kako je te podatke potrebno obraditi prije nego što se pohrane. Takav tradicionalan način rada nije dorastao ogromnim količinama podataka koje se danas obrađuju i dosežu razinu od nekoliko terabajta[5] do nekoliko petabajta.[6] Potrebni su novi procesi i tehnologije koje bi se mogle nositi sa takvim količinama kako bi kasnije poslovni analitičar mogao donijeti zaključke iz njih. (Oracle, 2020.)

4.4.2. Upravljanje

Big Data kao i skladišta podataka zahtijevaju nekakav prostor za pohranu. Kod skladišta podataka to je lokalna pohrana na namjenskim poslužiteljima, dok *Big Data* pohranjuje svoje podatke u oblaku[7] (eng. *Cloud*) i/ili lokalno na poslužitelju, ustvari kod *Big Data* je moguće pohraniti podatke na bilo koji tehnološki način te se zapravo procesi obrade dovode do dataseta, a ne obrnuto. Način pohrane u oblaku sve više dobiva na važnosti te polako jača na strani obrade. (Oracle, 2020.)

4.4.3. Analiza

Cijela ta svrha pohrane podataka se vodi na jednu stvar, a to je njihova analiza. Iz takvih velikih izvora podataka je moguće izvući nove zaključke, uvide u poslovno okruženje kao i buduća saznanja. Do novih spoznaja se može doći na više načina, uz pomoć statističara, strojnog učenja (str. 11) i umjetne inteligencije, a mogu biti i od koristi drugim poduzećima i društvenim granama. (Oracle, 2020.)

[5] Terabajt – jedinica za količinu podataka, točnije, 1 terabajt jednak je 1024 gigabajta
[6] Petabajt – jedinica za količinu podataka, točnije, 1 petabajt jednak je 1024 terabajta
[7] Oblak – skup računala (server-a) koji služe za pohranu i pristup podacima s bilo koje lokacije. Vlasnik vodi brigu o održavanju i pružanju svoje usluge pohrane, a korisnici iznajmljuju servis.

4.5. Prednosti i nedostaci Big Data

U dosadašnjem dijelu je bilo riječi o tome što je to *Big Data*, od kud potječe, kako funkcionira i koje su mu primjene. Iz pojedinih točaka se moglo iščitati koje bi to prednosti mogle biti, a koji to nedostaci. U nastavku slijede opisi pojedinih prednosti i nedostataka *Big Data*.

Big Data omogućuje vlasnicima poduzeća da imaju novi, opširniji i sveobuhvatniji pogled na svoje poduzeće, ali i konkurente. Takvo nešto im omogućuje povećanje profita poduzeća pomoću kojeg radnicima omogućuju veću plaću, bolje radno okruženje kao i otvaranje novih radnih mjesta. Još jedan sektor ili razlog koji može imati beneficija od *Big Data* je zdravstvo. Podaci pacijenata mogu biti od koristi samim liječnicima, ali također mogu i biti korisni onima koji proizvode lijekove, odnosno traže nove lijekove. Na taj način imaju veći bazen podataka na raspolaganju. Također je jednostavnije i efikasnije proučavati svojstva genetike putem *Big Data*, tako liječnik može imati uvid u pacijentov genetički kod, a time i sve bolesti koje ima ili koje bi mogao imati, pa sve do toga da se „popravi" genetički kod. (Zhang, 2017, str. 49) Cijeli naš dnevni život se može promijeniti na bolje uz pomoć *Big Data*, npr. koliki je broj rasvjetnih stupova ili kanti za smeće optimalan, kako umanjiti postotak ubojstava i samoubojstava. Policijska uprava u Memphis-u (Tennessee) je smanjila broj zločina za 30% uz pomoć *Big Data.* (Zhang, 2017, str. 50) Upravo će sličan takav primjer biti obrađen u drugoj polovici rada (Praktični primjer). Sve u svemu, Big Data na kraju dana omogućuje dobivanje boljeg, bržeg i točnijeg odgovara, omogućuje nam da efikasnije i brže djelujemo i sl.

Nakon prednosti slijede neki značajni nedostaci Big Data koji se i u određenoj mjeri kose s prednostima te je potrebno vagati među njima. Prvi bitni nedostatak je sigurnost, budući da su podaci kod Big Data većinom pohranjeni u oblaku kod neke treće strane, a vezani su za korisnike, odnosno kupce nekog poduzeća što može biti „škakljivo". Također jedan od nedostataka je pojava problema *velikog brata*[8]*.* Ljudi se toga podosta boje, te zapravo pokušavaju izbjegavati dijeljenje osobnih podataka koliko god je to moguće, no uvijek postoji nekakav skriveni način koji ljudi previde (kamere na cestama, pametni telefoni, itd.). Još jedan nedostatak *Big Data* je njegova nedostupnost za manja poduzeća, a time im je uskraćen veliki dio podataka i informacija, dok sva veća poduzeća imaju njima pristup. Na taj način se mala poduzeća teško mogu izboriti i dokazati u poslovnome svijetu.

[8] Veliki brat – (engl. Big Brother) termin za svemoćnu i sveprisutna vlast, vlast koja nadzire i nastoji nadzirati sve podatke

4.6. Hadoop

Prethodno je bio naveden okvir (engl. *framework*) zvan Hadoop, a bez kojeg je *Big Data* nezamisliv. Hadoop je okvir (eng. *framework*) koji omogućuje distribuiranu obradu velikog seta podataka na nizu računala koristeći razne programske modele. Na tržište je izbačen 2005. godine od strane Apache Software Foundation (neprofitna organizacija). Okvir je dizajniran tako da sustav koji se gradi na njemu može biti vrlo skalabilan gdje podržava rad na jednom poslužitelju (eng. *server*) pa sve do nekoliko tisuća. Korisnici se ne oslanjaju samo na hardver za dostupnost podacima, nego i na sami Hadoop za detekciju, otkrivanje i rukovođenjem grešaka na aplikacijskoj razini, a time je postignuta visoka razina dostupnosti servisa.

Tri bitne značajke Hadoop-a su MapReduce, HDFS (eng. *The Hadoop Distributed Filesystem*) i YARN. MapReduce se koristi kod velikih seta podataka, a posjeduje dvije bitne tehnike, jedna zvana *map,* a druga *reduce.* Tehnika*,* odnosno procedura *map* provodi filtriranje i sortiranje dok *reduce* vrši sumiranje danih podataka. Korištenjem navedene značajke omogućen je brzi pristup podacima budući da su lokalni. Druga značajka HDFS, je optimizirani sustav pohrane podataka na distribuiranim sustavima koji vodi brigu o tome da se pojedini podaci mogu pronaći i koristiti iako se nalaze na različitim računalima na različitim lokacijama. (White, 2012) Zadnja značajka je YARN, koji upravlja resursima sistema na kojima su podaci pohranjeni i vodi analizu nad podacima. (Marr, 2020.)

S porastom broja podataka koji se može pohraniti na jednom hard disku, stručnjaci su uočili kako će se brzo nadmašiti veličine kapaciteta hard diska. Također počinje se javljati značajno kašnjenje prilikom čitanja zapisa, što nije prihvatljivo niti u jednoj vrsti pohrane podataka. Stručnjaci su odlučili odvojiti te podatke na više manjih jedinica za pohranu koje mogu paralelno raditi, a tako je i nastao Hadoop. (Marr, 2020.)

4.7. Internet stvari

U prethodnim lekcijama spomenut je pojam *internet stvari* (eng. *Internet of things, IoT*) koji se usko veže uz poslovnu analitiku i znanost podataka te je jedan od izvora podataka za *skladišta podataka* i *velike podatke*. U nastavku slijedi opširniji opis, primjena i značajnost pojma u promatranom području.

Prvo je potrebno ustanoviti što su to *internet stvari*, na što se one odnose i kako one funkcioniraju, odnosno kako se uklapaju u pojam poslovne analitike. U nastavku slijedi definicija *internet stvari*:

„Internet stvari podrazumijevaju povezanost različitih uređaja koji primjenom informacijsko-telekomunikacijskih tehnologija međusobno komuniciraju i dijele informacije.“ (Mladen Varga, 2016, str. 193)

Kada govorimo o *internet stvari* u većini slučajeva se umjesto *internet stvari* koristi skraćenica *IoT*. U stvarnome svijetu *internet stvari* se odnose na koncept kojim su uređaji međusobno povezani putem interneta, dijele razne podatke i informacije međusobno, većinom bez ljudske interakcije (čovjek je samo promatrač). Neki takvi uređaji su pametni hladnjaci, zvučnici, televizori, igračke, razne nosivi „gedžeti“, klime, termostati, pa čak i perilice rublja, ustvari sve što radi na električnu struju može biti i *IoT* uređaj, a to ovisi od proizvođača, ali i želji kupca.

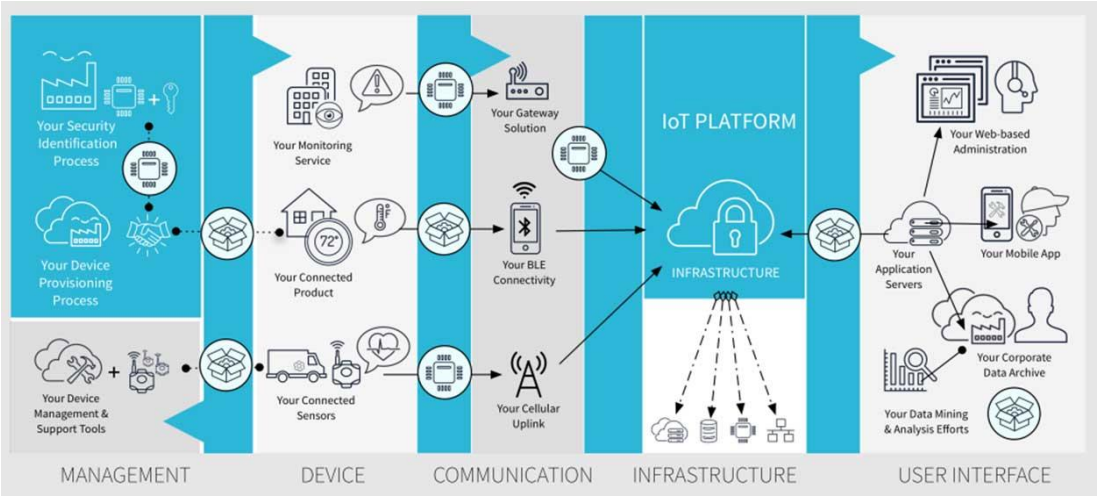
IoT uređaji često su povezani na internet putem WiFi veze, no postoje i razni drugi načini, kao što je to mobilne mreže (4G, 3G, HSPA, itd.), Bluetooth, RFID, LAN i razni drugi načini. Vrsta povezivanja većinom ovisi o samoj namjeni uređaja, tako npr. *IoT* omogućuje kategorizaciju uređaja unutar okruženja, pa čak i okruženje je moguće kategorizirati, npr. moguće je povezati termostat na mrežu koji onda prepoznaje kada je vlasnik u blizini ili ne. Time je moguće gasiti grijanje u kući kada se osoba kreće na posao i ponovno paljenje nakon što se osoba vrati u dom, što može značajno uštedjeti na režijama. Naravno mogući je niz drugih načina povezivanja i iskorištavanja *IoT* koncepta, postoje i primjeri gdje hladnjak prepoznaje nedostatak hrane te istu bilježi ili čak naručuje uz pomoć korisnika ukoliko je to potrebno što se naravno bilježi u obliku transakcije. Takve transakcije do određene granice opisuju kupca što je današnjim poduzećima korisno za plasiranje proizvoda.

Iz gornjeg teksta se lako vidi koliki broj podataka se može prikupiti o jednoj osobi s jednim uređajem u jednom danu, a danas ljudi imaju nekolicinu takvih uređaja oko sebe, stoga je lako zaključiti kako se radi o ogromnoj količini podataka koja se prikuplja.

Svi ti podaci su od velike koristi proizvođačima uređaja kao i onima koji crpe podatke iz njih te ih nastoje čim pametnije iskoristiti u svrhe oblikovanja, plasiranja proizvoda ili stvaranja imidža proizvoda (vidi 4.3). Količina podataka koja se prikuplja će i dalje naglo rasti, tako je npr. 2015. godine u svijetu bilo oko 25 milijardi takvih uređaja, a danas oko 50 milijardi. (Mladen Varga, 2016, str. 195).

Također prema nekakvim neslužbenim podacima 2025. godine će *IoT* uređaji generirati 79,4 zetabajta[9] podataka, što je prije nekoliko godina bila nezamisliva količina podataka, a sada je to stvarnost u kojoj ljudi žive. Za većinu podataka koje se prikuplja ljudi niti ne znaju koji bi to mogli biti, ali su nekome ili nekomu od velike koristi. (Estopace, 2019.)

[9] Zetabajt – jednak je 1 000 000 petabajta



Slika 4: Prikaz IoT okruženja (Izvor: iotondemand)[10]

Kao što je to bio slučaj s velikim podacima, javlja se problem privatnosti i sigurnosti kao i svugdje na internetu. U ovome slučaju je to nešto izraženije budući da se ljudi okružuju takvim uređajima unutar svojih kućanstava čime je izraženiji efekt velikog brata, a pogotovo unutar organizacija i poduzeća (vidi 4.5). Kako bi se današnja poduzeća zaštitila od zlouporabe takvih novih tehnologija potrebno je osigurati visoki stupanj zaštite unutar mrežne infrastrukture i pristupa internetu gdje je potrebno koristiti odgovarajuću razinu identifikacije i autentifikacije. (Mladen Varga, 2016)

Unatoč takvim nedostacima i nedoumicama, postoji veliki broj korisnih aspekata za korisnike *IoT* koncepta, bilo to za ljude, poduzeća ili općenito društvo. Kada o društvenim koristima govorimo, to se odnosi na izgradnje pametnih gradova s ciljem smanjenje potrošnje struje, plina i drugih resursa kao i smanjenjem stope zločina, itd.

4.8. Big Data i skladište podataka

Dosad su opisana oba pojma, *Big Data* i *Skladište podataka* gdje je moguće uočiti određene razlike. Jasno je vidljivo kako skladište podataka zagovara strukturirane podatke, tj. podatke koji mogu biti prikazani u nekakvom tabličnom obliku, dok na drugoj strani kod *Big Data* nije bitno jesu li podaci strukturirani ili ne. Također *Big Data* može imati značajno veći broj izvora podataka nego skladište podataka, a time i veću količinu podataka. U analitičkom i statističkom smislu, skladište podataka više odgovara za analizu podataka budući da su podaci već smisleno strukturirani i grupirani. Takvo nešto nije uvijek moguće, a da bi se to postiglo kod *Big Data*, potrebno je koristiti vrlo napredne tehnologije, što može biti dosta skupo.

[10] Preuzeto 02.04.2020. Izvor: https://iotondemand.com/wp-content/uploads/2019/03/what-iot.jpg

5. Poslovna analitika

Sve opisano je potrebno za postavljanje temelja koji su neophodni za shvaćanje samog termina *Poslovne analitike*, čemu služi, kako ono funkcionira, koji svi aspekti poslovne analitike postoje, što se uopće može dobiti korištenjem *Poslovne analitike* i slično. U nastavku rada se upravo o tome terminu govori, također će sve navedeno biti potkrijepljeno primjerima radi boljeg razumijevanja opisanog.

5.1. Opis poslovne analitike

Kada govorimo o *Poslovnoj analitici* (eng. *Business analytics*, BA) tada svi stručnjaci imaju određenu sliku o tome što je to, čemu služi i kako funkcionira, ali zapravo ne postoji jedinstvena definicija s kojom bi se mogao točno i jasno opisati navedeni termin. Većina stručnjaka smatra kako je *Poslovna analitika* ustvari produžetak, odnosno nadopuna *Poslovne inteligencije* budući da obje grane omogućuju jednu stvar – potporu odlučivanju. Bitno je napomenuti kako obje grane takvo nešto postižu na različite načine, tako npr. *Poslovna inteligencija* omogućuje davanje odgovora što se sve dogodilo u prethodnim razdobljima, koliko često se nešto događa/stvara/omogućuje, u kojoj količini, gdje se nalazi problem i slično, s ciljem da se odrede potrebne akcije, odnosno odluke. Na drugoj strani *Poslovna analitika* daje nešto sofisticiranije odgovore, npr. zašto se nešto uistinu događa, koji je razlog tomu, koje su buduće projekcije na temelju trenutnih trendova, što bi se sljedeće moglo dogoditi i koji su mogući učinci.

Jasno je vidljivo kako *Poslovna inteligencija* i *Poslovna analitika* daju slične odgovore, s time da *Poslovna analitika* više skreće pažnju na buduća događanja, ustvari nekakva predviđanja (prediktivno, preskriptivno), dok Poslovna inteligencija opisuje trenutno stanje (opisno). Na temelju dobivenih odgovora, tj. podataka baziraju se i moguće poslovne odluke poduzeća. (Resources, 2020.)

Unatoč svemu rečenome neka opće prihvaćena definicija *Poslovne analitike* bi bila sljedeća:

> „*Poslovna analitika je iterativan i metodičan proces istraživanja podataka organizacije, gdje se određena pažnja postavlja na statističku analizu. Poslovnu analitiku koriste kompanije čije su odluke bazirane i provedene na temelju podataka koje posjeduju. Takve kompanije tretiraju svoje podatke kao imovinu iz kojih mogu aktivno izvući kompetitivnu korist.*“ (Rouse, 2019)

5.2. Tipovi poslovne analitike

Prethodno, prilikom opisivanja relacije *Poslovne analitike* i *Poslovne inteligencije* navedeno je nekoliko pojmova, a to su opisni, prediktivni i preskriptivni pojam. Upravo ti navedeni pojmovi su jedni od mogućih tipova *Poslovne analitike* s time da nedostaje dijagnostička analitika. Svaki od navedenih tipova ima određenu svrhu i primjenu, a u nastavku slijedi malo više riječi o tome.

5.2.1. Deskriptivna analitika

Prvi i najjednostavniji oblik *Poslovne analitike* je deskriptivna ili opisna analitika (eng. *descriptive analytics*). Opisnu analitiku koriste sva poduzeća, velika i mala, gdje neki to rade svjesno, a neki nesvjesno. Često se opisni tip analitike poistovjećuje s *Poslovnom inteligencijom* budući da daje informacije o prošlom i trenutnom stanju. Opisna analitika daje uvid u prošlo stanje poduzeća/organizacije bilo to unazad nekoliko tjedana ili 50 godina. Koristi se kada je potrebno uočiti određeni trend ili predvidljivo ponašanje, odnosno uzorak unutar promatranih podataka i kada ga želimo vizualno prikazati i opisati radi boljeg i lakšeg razumijevanja. (Zhang, 2017)

Tako npr. kod sezonske prodaje umjetnih borova ili suncobrana, poduzeće bi moglo na temelju prijašnjih godina odrediti kolike zalihe proizvoda trebaju stvoriti prije nego što krene prodaja i tako se pravovremeno mogu prebaciti na proizvodnju drugog proizvoda za drugi dio sezone. Na taj način poduzeće može bolje postaviti svoje poslovne procese čime postaje efektivnije i efikasnije.

Kao što je to bilo navedeno u prijašnjim cjelinama (3 i 4), podaci mogu dolaziti s različitih izvora, bilo to interno, eksterno ili kombinirano. Kompanija ili neko poduzeće uzima postojeće podatke te ih upotpunjuje s obrazloženim nagađanjima, barem tamo gdje je to potrebno. Svi ti podaci se kombiniraju pa kasnije analiziraju. Drugim riječima, traže se razni uzorci, provode se algoritmi nad danim setom podataka i na kraju se dobiju određeni statistički podaci koji mogu pomoći u donošenju odluka. (Zhang, 2017, str. 92)

Kod deskriptivne analitike se najčešće koriste dva različita statistička pristupa, jedan je pod nazivom mjera centralne tendencije, a drugi mjera disperzije. Mjera centralne tendencije se svodi na promatranje jedne centralne, središnje, figure (mjere) unutar više njih. U same mjere je uključena aritmetička sredina, mod i medijan. Navedene tri vrijednosti unutar jedne brojke sažimaju veći skup podataka, što je savršeno za vodeće figure nekog poduzeća jer lakše mogu interpretirati podatke i donijeti odluku. (Zhang, 2017)

Opisi pojedinih mjera centralne tendencije (Jasminka Dobša, 2008):

- **Aritmetička sredina** – je vrijednost koja se uvijek izračunava, a ovisi o tome jesu li vrijednosti grupirane ili negrupirane. U matematičkom smislu, aritmetička sredina je omjer ukupne vrijednosti numeričkog obilježja jedinica u statističkom nizu i broja jedinica u tom nizu.
- **Mod** – ili zvano još najčešća vrijednost, kao što naziv kaže je vrijednost numeričkog obilježja koje se javlja kod najvećeg broja elemenata u statističkom skupu te ulazi u grupu položajnih srednjih vrijednosti ukoliko se radi o redoslijednom nizu. Ako jedinice skupa nisu grupirane već su date pojedinačno, modalna vrijednost obilježja je ona koja je najčešća.
- **Medijan** – je numeričko obilježje koje dijeli elemente određenog skupa na dva jednakobrojna dijela. Prva polovica ima obilježje manje ili jednako od medijana, a druga polovica označuje veće ili jednako od medijana. Kada je skup jedinica negrupiran, sortiran i neparan tada središnji element predstavlja medijan, a ukoliko je skup jedinica paran tada je medijan poluzbroj vrijednosti obilježja središnjih dvaju članova.

5.2.2. Prediktivna analitika

Idući odlomak se bavi *prediktivnom analitikom*, koja je značajno složenija od deskriptivne te zahtjeva veće tehničko i tehnološko znanje kako bi se provelo, koji efekt ima na poduzeće i slično.

Prvo je potrebno znati što radi, odnosno što omogućuje *prediktivna analitika* (engl. *Predictive analytics*). Ona omogućuje određivanje (predviđanje) vjerojatnosti izvršavanja budućih događanja na temelju različitih statističkih modela, algoritama kao i korištenjem strojnog učenja i umjetne inteligencije. Ustvari, *prediktivna analitika* je nadogradnja deskriptivne analitike s razlikom da ona daje značajno preciznije podatke za moguća buduća stanja. Za razliku od opisne analitike, *prediktivnu analitiku* koriste nešto veća poduzeća i organizacije jer zahtjeva više vremena, kao i naprednije tehnologije i tehnike. Bitno je napomenuti kako sve više manjih poduzeća ima pristup takvim resursima (ponajviše umjetnoj inteligenciji i strojnom učenju) stoga se može očekivati kako će i ona s vremenom početi iskorištavati svojstva *prediktivne analitike*. (Zhang, 2017, str. 97) Jedan takav primjer koji je ujedno i besplatno dostupan je *ML.NET* alat za strojno učenje za koji nije potrebno imati iskusnog matematičara i/ili statističara budući da alat većinu toga sam obavlja.

Jedan od najpoznatijih područja primjene *prediktivne analitike* je bankarstvo, odnosno otkrivanje raznih vrsta prevara, bilo to kod normalnih bankarskih transakcija, kredita i slično. Na temelju prijašnjih podataka kod kojih su se dogodile prevare, onih uobičajenih podataka te uz pomoć strojnog učenja i umjetne inteligencije moguće je predvidjeti prevare te ih suzbiti. Također je moguće pratiti kretanje tržišta gdje se onda uz pomoć *prediktivne analitike* mogu odrediti budući trendovi na tržištu na temelju kojih se mogu onda kupovati i/ili prodavati dionice, obveznice i slično. Još jedan primjer koji se teže može uočiti je kod Netflix-a, koji prilikom prve prijave od korisnika traži da odabere žanrove i filmove/serije koje ga zanimaju te na temelju njih servis može odrediti koje bi ga idući filmovi/serije mogli zanimati te mu iste preporučuje.

Prediktivna analitika se može dijeliti na tri načina, prediktivni model, opisni model kao i model odlučivanja. Prediktivni model je ustvari reprezentacija relacija između člana uzorka, poznatih karakteristika uzorka i kako se član ponaša. Cilj je odrediti kako bi se ponašao neki drugi, slični, član nekog drugog uzorka u tome, odnosno dokazuje se ili opovrgava da li se isto ponaša. Prediktivni model često se koristi u području marketinga gdje se tako može odrediti tip korisnika, tj. njegove preferencije. Takvo nešto je moguće jer su tipovi korisnika grupirani (uzorci) te se na temelju ponašanja članova unutar uzorka zna kako bi se ponašao neki novi član, u ovome slučaju korisnik. (Zhang, 2017, str. 99)

Opisni model opisuje relaciju između događaja i akcija koje su prouzročile događaj. Koriste ga različita poduzeća i organizacije koja si na taj način žele pomoći prilikom određivanja područja plasiranja kao i vrste oglašavanja svojih proizvoda/usluga. Također se u tom slučaju grupiraju i kategoriziraju korisnici, no potrebno je koristiti Opisni model u sklopu Prediktivnog modela kako bi se dobili željeni rezultati. (Zhang, 2017, str. 100)

Preostaje još model odlučivanja kojem zadnjih par godina raste popularnost. Ovaj model uz pomoć velike količine podataka i korištenjem složenih algoritama daje moguće odluke koje pomažu menadžerima prilikom odlučivanja. Model odlučivanja se orijentira samo na odluke/akcije koje se mogu ponavljati također i na one vrijednosti koje se egzaktno mogu mjeriti. (Zhang, 2017) Može se koristi npr. za određivanje najbržeg i najjeftinijeg načina slanja proizvoda kupcu.

Postoji veliki broj tehnika i tehnologija koji se koristi kod *prediktivne analitike*, a u nastavku će neke od njih biti prikazane i objašnjene. Prema A. Zhangu (2017.) neke od tih tehnika su sljedeće:

- **Strojno učenje** (engl. *Machine Learning*) – koristi algoritam koji stalno „uči“, usavršava i prilagođava se na temelju danog seta podataka bez da se fiksno ukodira algoritam.

- **Regresijske tehnike** – sastavljanje jednadžbi koje predstavljaju model u kojem su vidljive interakcije između varijabli.
- **Linearna regresija** – najbolje shvaćena tehnika koja računa odnos između zavisnih[11] i nezavisnih[12] varijabli koristeći ravnu liniju, zvanu regresijska linija, koja je često u obliku jednadžbe. Koristi se samo kada je zavisna varijabla ima neograničen raspon, u protivnome se koriste druge tehnike.
- **Logistička regresija** – također jedna od tehnika koja se vrlo često primjenjuje, a koja omogućuje računanje vjerojatnosti uspjeha ili neuspjeha nekog događaja.
- **Probit model** – slično linearnoj regresiji s razlikom da zavisna varijabla može poprimiti samo jednu od dvije vrijednosti (binarna vrijednost).
- **Neuronske mreže** – koristi se za komplicirane ulazno/izlazne podatke, kada se ne znaju točni odnosi između ulaza i izlaza. Neuronske mreže se uče, te sklapaju nove spojeve (spoznaje) kroz treniranje, slično kao ljudski mozak.
- **Ostalo**: Radijalna mreža funkcija (engl. *Radial basis function networks*), Metoda potpornih vektora (engl. *Support vector machines*), Naivi Bayesovi klasifikatori (engl. *Naive Bayes*), Učenje zasnovano na primjerima (engl. *Instance based learning*), Prediktivno prostorno modeliranje (engl. *Geospatial predictive modeling*), Hitachi-ev prediktivni analitički model (engl. *Hitachi predictive analytic model*), itd.

Svaka od navedenih tehnika ima svoje prednosti, nedostatke i područje primjene. Na temelju takvih karakteristika potrebno odrediti koja bi tehnika ili tehnike najbolje odgovarale za provođenje analize podataka.

5.2.3. Preskriptivna analitika

Iduća vrsta *Poslovne analitike* je *perskirptivna analitika* (engl. *prescriptive analytics*) koja je u odnosu na prethodne dvije najnaprednija, što ne čudi budući da je najnovija. Ukratko, poslovna analitika koristi razne simulacije, algoritme i statističke metode kako bi se odredile najbolje akcije čijom provedbom se može maksimizirati određena vrsta profita i rasta. (Zhang, 2017, str. 243) U prijevodu, *preskriptivna analitika* ne govori samo što neko poduzeće može čekati u budućnosti, nego što treba i poduzeti kako bi to poduzeće bilo spremno za to određeno stanje koje je jedno od nekoliko mogućih stanja. (Resources, 2020.)

[11] Zavisna varijabla – onaj podatak koji se promatra, koji se nastoji odrediti, a ovisan je o drugim parametrima, tj. varijablama

[12] Nezavisna varijabla – oni podaci koji su samostojeći i njihove vrijednosti ne ovise o drugim varijablama, ali određuju stanje neke varijable (Zavisne varijable)

Preskriptivna analitika zahtjeva pristup gotovo svim podacima s kojima poduzeće radi, pa tako i onim trenutnim, važećim, podacima. Također, algoritmi koji barataju takvim podacima rade s malim brojem parametara, odnosno sami podaci imaju relativno mali broj parametara. Budući da je takva vrsta analitike najnovija, nju koriste samo one veće kompanije i to samo određene. Jedan primjer *preskriptivne analitike* je Google-ov autonomni automobil. Upravo u takvom području su potrebne ogromne količine podataka, kako starih tako i trenutnih podataka. Stari podaci su tu kako bi automobil mogao jednostavno voziti u svojoj traci, a trenutni podaci, kako bi automobil mogao reagirati prilikom zagušenja prometa ispred sebe, naglih reakcija drugih vozača i slično. Cilj je da vozilo bude adaptivno svojoj okolini. Podaci u sustav ulaze sa nekoliko izvora, kao što su kamere, sonar senzori, drugi optički senzori i internet.

5.2.4. Dijagnostička analitika

Zadnja vrsta *Poslovne analitike* je dijagnostička analitika (engl. *diagnostic analytics*) koja se bavi pitanjem „Zašto se to dogodilo?“. *Opisna* i *prediktivna analitika* daju odgovore na to „Što se dogodilo?“ i „Što će se sljedeće dogoditi?“, ali u nekim slučajevima je potrebno znati zašto je došlo do određenog stanja.

Kako bi se takvo nešto postiglo, potrebno je odrediti anomalije iz podataka kao što je npr. povećana prodaja određenog proizvoda na tržištu gdje se nije posebno promovirao određeni proizvod. Nakon što su se anomalije izdvojile, analitičar pretražuje/istražuje dostupne setove podataka kako bi pronašli ovisnosti određenih podataka s anomalijama. Često se koristi strojno učenje u takvim slučajevima, zbog toga što su računala znatno brža i točnija od ljudi prilikom određivanja ponavljajućih uzoraka, otkrivanja anomalija i sl. Moguće je koristit druge tehnike i tehnologije, no čovjek, odnosno analitičar je uvijek prisutan prilikom vrednovanja i potvrđivanja rezultata ali i vizualizacije istih budući da ponekad takvi podaci mogu biti u neprikladnom obliku.

5.3. Poslovna analitika i Poslovna inteligencija

U ovome trenutku se može dati jasniji odgovor na razliku između oba pojma. Poslovna inteligencija prikuplja, pohranjuje i analizira podatke na temelju kojih se donose odluke, dok poslovna analitika radi isto sve to uz dodatak rudarenju podataka (data-mining), statističku analizu, prediktivno modeliranje, strojno učenje, itd.. Sama razlika se na kraju svodi na sami odgovor koji daju, a to je da poslovna inteligencija odgovara na pitanje *što* i *kako* se nešto dogodilo, dok poslovna analitika daje opširniji odgovor, npr. *zašto* se nešto dogodilo, *kada* će se ponovo dogoditi, na koji način. U prijevodu, poslovna analitika omogućuje uvid u budućnost do određene razine kao i detaljnije opise promatranog. (Tableau, 2020.)

6. Postavljanje okruženja

Prije nego što se *poslovna analitika* prikaže u praksi potrebno je postaviti radno okruženje i alate. Bitno je prikazati pojedinosti MS SQL Servera i drugih tehnologija kako bi praktični primjer bio razumljiv i lako replicirajući. U nastavku je prikazano postavljanje i korištenje sljedećih alata/tehnologija: SQL Server 19, SSMS 18 i R jezik.

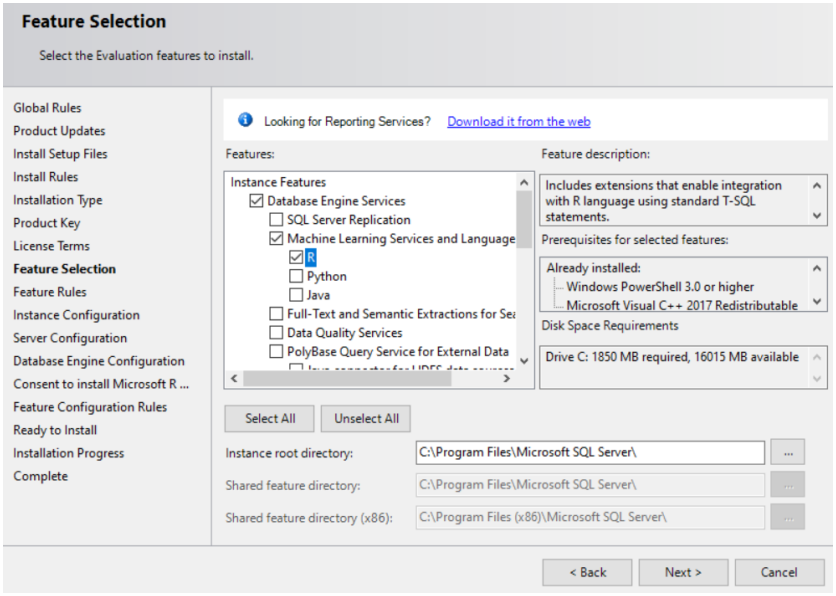
6.1. MS SQL Server i SSMS

Podaci se relativno lagano prikupe, no potrebno ih je negdje pohraniti. Kao pohrana će poslužiti MS SQL Server 2019[13], najnovija inačica, a alat za upravljanje bazom podataka koristit će se SSMS 18 (Microsoft SQL Server Management Studio 18[14]). Kroz daljnji tekst opisani su pojedini koraci instalacija.

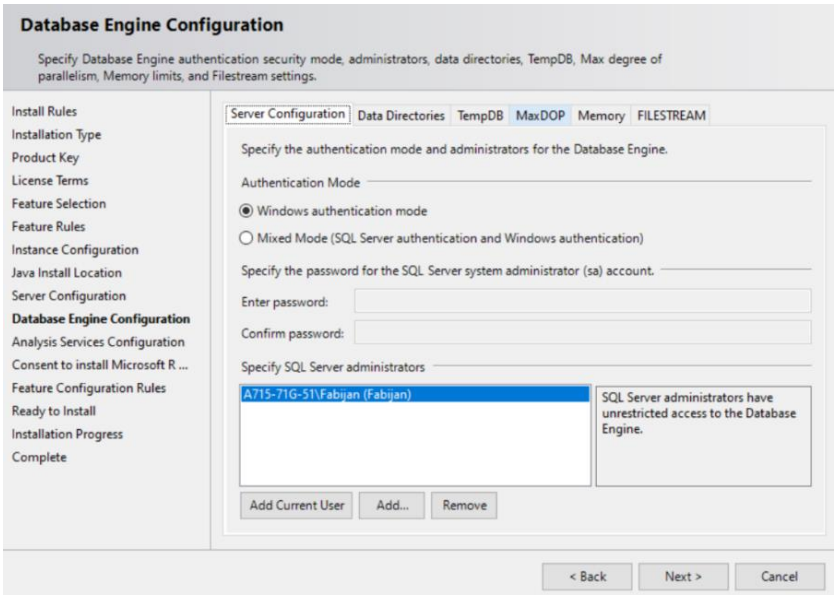
Prilikom pokretanja procesa instalacije prvih nekoliko koraka se brzo prođe, na tim mjestima se ostavljaju preporučene vrijednosti. Prvi bitan prozor na koji se naiđe je zvan „*Feature Selection“* u kojem je moguće izabrati dodatne funkcionalnosti dostupne za SQL Server. Iz prozora dodatnih funkcionalnosti potrebno je odabrati isključivo sljedeće: Instance *Features* > *Database EngineServices* > *Machine Learning Services and Language Extension* > *R* (Slika 5). Pomoću te funkcionalnosti moguće je koristiti, tj. pokretati R jezik unutar SQL Servera što je neizostavna komponenta za provedbu *poslovne analitike* nad bilo kojim podacima. U iste svrhe moguće je koristiti i Python, no on nije prikazan u ovome radu.

Tijekom instalacije potrebno je imenovati instancu SQL Servera, odnosno „lokaciju“ gdje će se nalaziti. U dano polje moguće je upisati proizvoljno ime ili je moguće odabrati već postavljeno ime (*MSSQLSERVER*). U ovome slučaju je to druga opcija. Na idućem koraku potrebno je odrediti pojedinosti SQL Server instance (engl. *Database Engine Configuration*). Pod karticom *Server Configuration* > *Authentication Mode* potrebno je odabrati prvu opciju (engl. *Windows authentication mode*) i na dnu istog zaslona se odabere *Add Current User* (Slika 6). Navedena opcija je vrlo korisna ukoliko se želi na jednostavan način testirati R kod o kojem će biti nešto više riječi kasnije. Ostatak instalacije se može privesti kraju sa preporučenim postavkama.

[13] Izvor (pristupano 31.07.2020): https://www.microsoft.com/en-us/sql-server/sql-server-downloads#
[14] Izvor (pristupano 31.07.2020): https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15



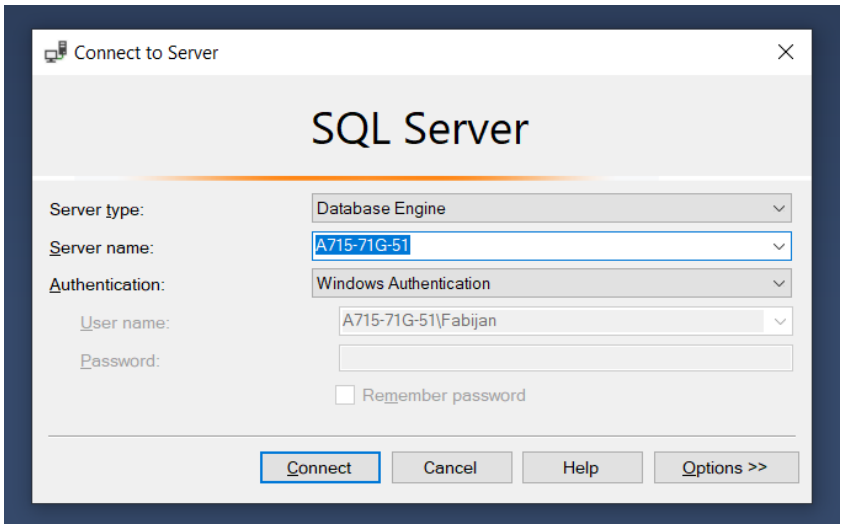
Slika 5. Odabir dodatnih funkcionalnosti SQL Server 2019 (Izvor: vlastita izrada)



Slika 6. Prikaz postavljanja „Database Engine Configuration“ (izvor: vlastita izrada)

Preostaje još instalacija SSMS 18 sustava za upravljanje bazom podataka. Instalacija je vrlo jednostavna, prate se samo upute prikazane unutar prozora, gdje je najbitnije odabrati odgovarajuće mjesto instalacije alata. Kada se alat instalira i otvori, korisnik se susreće s prozorom prikazanim na slici 7.

Korisnik ne treba unositi nikakve podatke, već odabere dostupnu SQL Server instancu i ukoliko je prijavljen u Windows operacijski sustav s ispravnim korisnikom tada će mu biti omogućen pristup odabranoj SQL Server instanci, odnosno bazi/bazama podataka. Na taj način je značajno olakšan pristup SQL Server instanci.



Slika 7. Spajanje na SQL Server instancu (izvor: vlastita izrada)

Time je riješena instalacija potrebnih sustava i alata. U nastavku slijede opisi na koji se način pristupa podacima, kako se R paketi unose, kako se generiraju grafovi i na što sve treba obratiti pozornost.

6.2. Unos podataka

Što se tiče učitavanja podataka u SQL Server, postoji nekoliko načina. U ovome slučaju je samo jedan dovoljan, a to je unos CSV datoteke. U alatu SSMS pod konekcijom na *Server –> Databases –> <Ime baze podataka>* se pritisne desnim klikom miša i iz izbornika se odabere *Tasks –> Import Data* gdje se onda otvara novi prozor. Moguće je i koristiti opciju *Import Flat File* budući da se radi o CSV datoteci, no prva opcija unosa nudi veću razinu fleksibilnosti.

U novome prozoru se pritisne *Next* opcija gdje se onda prijeđe na idući prikaz u kojem se odabire vrsta izvora podataka. U ovome slučaju je to CSV datoteka odnosno *Flat File Source*. Odabere se CSV datoteka i postave se sve dodatne opcije, bilo vrsta pisma, jezika, načina razdvajanja atributa, i sl. U istom prozoru pod *Columns* moguć je uvid u izgled podataka, odnosno kako bi podaci izgledali u tablici. Pod opcijom *Advanced* postoji odabir tipa podataka za pojedini atribut te izmjena naziva stupca kao i neke druge opcije. Pod *Preview* je moguć uvid u krajnji rezultat, tj. izgleda podataka u novoj tablici. Nakon što se odabere opcija *Next*, na idućem prozoru se odabire mjesto pohrane tablice. U ovome slučaju je to *SQL Server Native Client* i odabere se Server instanca za koju je potrebna određena vrsta prijave (Windows autentifikacija). Kao posljednje, odabere se baza podataka za odabranu instancu. Pritisne se *Next* i na idućem prozoru se može unijeti naziv tablice koja će se kreirati, a ostatak opcija se može ostaviti kakve jesu. U ovome trenutku podaci bi se trebali uspješno unijeti u novu tablicu.

Uvoz podataka zapravo predstavlja prvi korak ETL procesa. Podaci ne trebaju biti samo unutar određenih datoteka, već se mogu nalaziti na fizičkim dokumentima, raznim medijima i sl. Također mogu biti i trenutni podaci, odnosno operativne baze podataka koje se aktivno koriste. Unos podataka ili kako se još zove, ekstrakcija podataka, se može značajno razlikovati ovisno o lokaciji podataka.

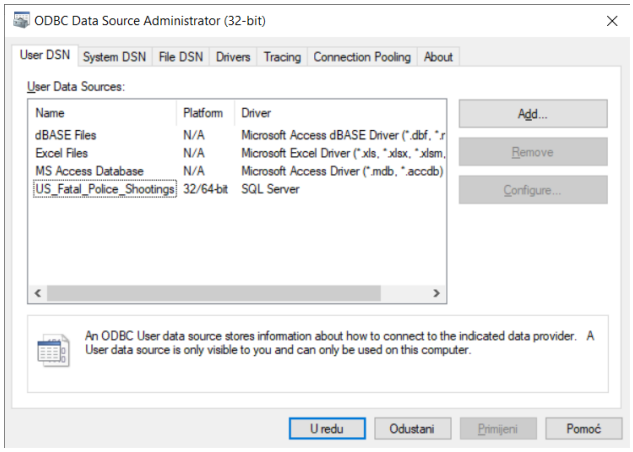
6.3. R jezik i R studio

Na početku cjeline navedeno je kako se uz *poslovnu analitiku* neslužbeno veže i R jezik, a time i R studio. Ukratko, R je programski jezik kojeg statističari, analitičari i drugi često koriste kako bi iz dostupnih podataka izvukli korisne informacije. Upravo je to najbitnija značajka u poslovnoj analitici.

Također je u radu navedeno kako SQL Server 2019 ima mogućnost korištenja R jezika unutar SQL koda, tj. upita, no ne preporučuje se takvo direktno programiranje jer lako može doći do logičkih pogrešaka. Osim toga, značajno je otežano i usporeno pisanje R koda. Stoga je dobro prije implementacije u SQL Server, napisati R kod u pripadajući R Studio radi lakše i brže provjere. R studio je moguće preuzeti na lokaciji stranice[15], a što se instalacije tiče, ona se provodi sa tvornički postavljenim opcijama. U nastavku su prikazani razni pristupi podacima iz baze podataka u R kodu.

6.3.1. Pristup podacima iz programa R studio

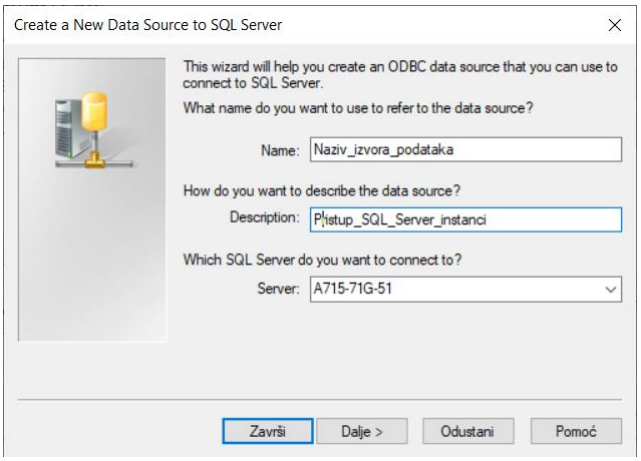
Da bi se iz R Studio-a moglo pristupiti podacima, potrebni su manji zahvati unutar postavki Windows operacijskog sustava. Potrebno je dodati „User DNS“ kod ODBC Data Source Administrator prozora. Do prozora se dolazi jednostavnim upisom ključne riječi „ODBC“ u pretraživač na programskoj traci u Windows OS-u.



Slika 8. Postavljanje User DNS (Izvor: vlastita izrada)

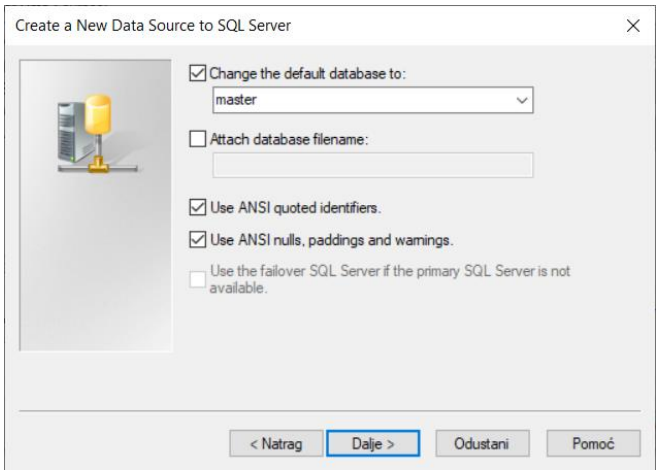
[15] Izvor (pristupano 31.07.2020): https://rstudio.com/products/rstudio/

Potrebno je pritisnuti na gumb „*Add*“, odabrati SQL Server iz liste i pritisnuti „*OK*“. Zatim se otvara novi prozor u kojem se zahtjeva unos naziva za izvor podataka (baza podataka), te konekcija na SQL Server instancu. Navedeni korak je prikazan na slici 9.



Slika 9. Dodavanje veze na SQL Server instancu (izvor: vlastita izrada)

Nakon što se polja ispune, odabere se dva puta opcija „*Dalje*“ gdje se onda dolazi na novi prozor (Slika 10.) u kojem se odabire konkretna baza podataka iz odabrane SQL Server instance. Iz padajućeg izbornika se odabere željena baza podataka.



Slika 10. Odabir baze podataka (izvor: vlastita izrada)

Nakon uspješno obavljenih koraka, korisnik ima postavljenu vezu na odabranu bazu podataka te sada može na jednostavan način pristupiti njenim podacima. Kako bi se iz R jezika moglo pristupiti podacima, potrebno je nekoliko linija koda. Ispod je prikazan minimalni broj linija koda koji su potrebni za pristup podacima iz baze podataka. Inače bi bio značajno otežan pristup podacima, a ovime se brže i preglednije može pristupiti podacima.

```
library(RODBC)
konekcija <- odbcConnect("Naziv_izvora_podataka", rows_at_time = 1)
podaci <- sqlQuery(konekcija, "SELECT * FROM table")
```

Unutar varijable *podaci* su dohvaćeni podaci iz baze podataka te korisnik sada može njima manipulirati kako mu odgovara.

6.3.2. Pristup podacima iz MS SQL Servera

Prvi način pristupa podacima je u većini slučajeva i više nego dovoljan, no postoje scenariji kada iz nekih razloga takav način nije adekvatan. Može postojati potreba za automatiziranim generiranjem statističkih rezultata za već predefinirani R kod ili se iz nekog servisa pristupa bazi podataka i želi se istovremena određena statistička obrada podataka (npr. strojno učenje u R jeziku). Tada je korisno ugraditi R kod unutar SQL procedura te ih pozivati po potrebi, a takvo nešto podržava SQL Server 19.

Da bi se takva značajka omogućila, potrebno je dati SQL Serveru prava za izvršavanje takvog koda, zvanog još eksterna skripta (engl. *sp_execute_external_script*). Eksterna skripta može biti pisana u R jeziku, ali i Python jeziku. U nastavku je prikazan SQL kod koji daje prava SQL Serveru za izvršavanje eksternih skripti.

```
sp_configure
GO
sp_configure 'external scripts enabled',1;
GO
RECONFIGURE
GO
```

Sada je omogućeno izvršavanje eksternih skripti, a u nastavku je prikazana SQL procedura koja sadrži i pokreće R programski kod. Sama procedura služi samo za objašnjenje pojedinih dijelova i načina rada R programskog koda u SQL-u.

```
CREATE PROCEDURE MojaProcedura (
    @param1 INT
    , @param2 INT
    , @param3 INT
    )
AS
EXECUTE sp_execute_external_script @language = N'R'
    , @script = N'
            podaci <- PodaciIzUpita
            OutputDataSet <- as.data.frame(rnorm(arg1, arg2, arg3));'
    , @input_data_1 = N'SELECT * FROM Tablica'
    , @input_data_1_name = N'PodaciIzUpita'
    , @params = N' @arg1 int, @arg2 int, @arg3 int'
    , @arg1 = @param1
    , @arg2 = @param2
    , @arg3 = @param3
      WITH RESULT SETS(([Rezultat] FLOAT NOT NULL));
```

Iz gornjeg isječka vidljivo je kreiranje procedure sa *EXECUTE* operacijom, gdje procedura ima ulazne parametre, no takvo nešto ne treba uvijek biti slučaj. Nadalje, kod *EXECUTE* operacije je definiran jezik (*@language*) koji će se koristiti unutar skripte (*@script*). Unutar *@script* se piše čisti R programski kod, a ostatak služi za definiranje imena ulaznih podataka ili definiranje unosa parametara u R kod.

Tako npr. *@input_data_1* služi za prihvat i unos ulaznog seta podataka (tablica ili SQL upit), a *@input_data_1_name* služi za definiranje imena, odnosno naziva varijable u kojem se nalaze ti podaci. Dohvat i korištenje takvih podataka je vidljivo unutar *@script,* gdje se dalje prate pravila i konvencije R jezika.

Nadalje, moguće je unositi zasebne parametre u R kod vidljivo po *@params* i *@arg1*, *@arg2*, *@arg3* izrazima*.* Unutar *@param* potrebno je definirati kojeg je tipa parametar koji se unosi u R jezik. Kao zadnje, sa *WITH RESULT SETS* se definira izlazni rezultat ukoliko on postoji, a on je pohranjen unutar varijable *OutputDataSet.* Moguće je imati i više izlaznih rezultata (npr. tablica) gdje je svaki pojedini rezultat odvojen zarezom.

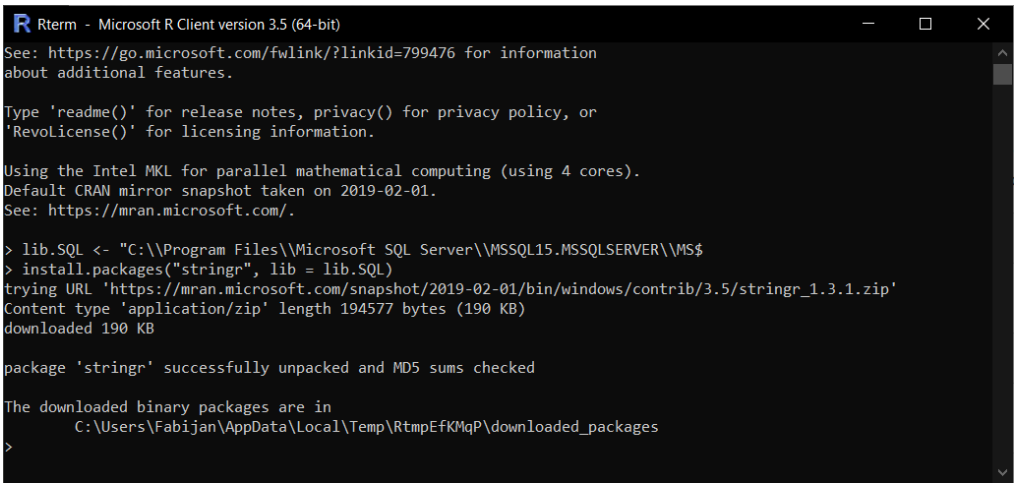
6.4. Preuzimanje i korištenje R paketa

Eksterna skripta R jezika ima nekoliko tvornički ugrađenih funkcija, a to su one koje se najčešće koriste. No u većini slučajeva takve funkcije nisu dovoljne kako bi se iskoristio puni potencijal samog jezika. U te svrhe se koriste vanjski paketi koji se trebaju preuzeti i pozvati kako bi se neke nove, naprednije funkcije mogle koristiti u sklopu R jezika. U nastavku slijedi opis preuzimanja i pozivanja R paketa kako bi se mogli koristiti unutar eksterne skripte u MS SQL Serveru 2019. Prilikom preuzimanja MS SQL Servera 2019 i R jezika, preuzela se tzv. *Microsoft R Client* verzija R-a, koju održava Microsoft i uobičajeno je niža inačica od one koju korisnici mogu samostalno preuzeti. Tako npr. *Microsoft R Client* je verzije 3.5.2, a uobičajeni R za Windows je inačice 4.0.1.

Kako bi se paketi mogli instalirati i koristiti unutar eksterne R skripte, potrebno je locirati izvršnu datoteku *Microsoft R Client*-a, a ona bi trebala biti u mapi sličnoj „*C:\Program Files\Microsoft\R Client\R_SERVER\bin*“. Tada se s administrativnim pravima pokrene *R.exe*. R paketi se preuzimaju, odnosno smještaju u tri moguće mape ("*C:\\Program Files\\Microsoft SQL Server\\MSSQL15.MSSQLSERVER\\MSSQL\\ExternalLibraries\\9\\1\\1*“, "*C:\\Program Files\\Microsoft SQL Server\\MSSQL15.MSSQLSERVER\\MSSQL\\ExternalLibraries\\9\\1\\1*“ ili "*C:\\Program Files\\Microsoft SQL Server\\MSSQL15.MSSQLSERVER\\R_SERVICES\\library*").

U nastavu su prikazane naredbe za preuzimanje R paketa.

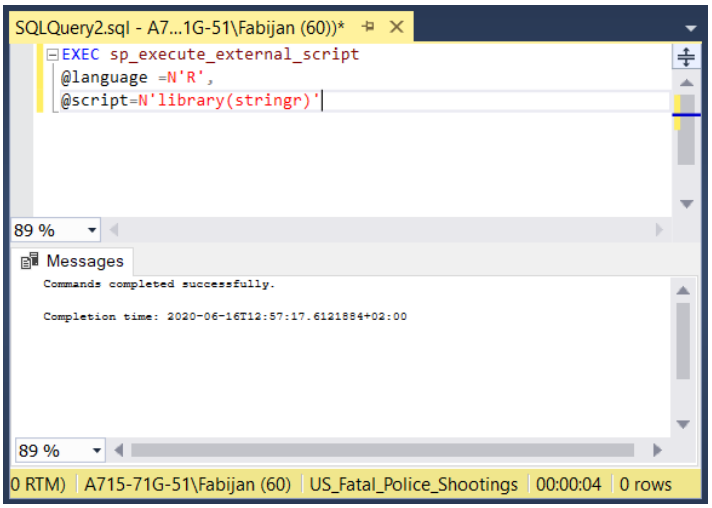
```
lib.SQL <- "C:\\Program Files\\Microsoft SQL
Server\\MSSQL15.MSSQLSERVER\\MSSQL\\ExternalLibraries\\9\\1\\1"

install.packages("stringr", lib = lib.SQL)
```



Slika 11. Preuzimanje R paketa (izvor: vlastita izrada)

U ovome trenutku je paket preuzet i spreman za korištenje unutar eksterne skripte. U nastavku će se prikazati način na koji se može pozvati paket ukoliko postoji potrebe za njegovim korištenjem unutar eksterne skripte.

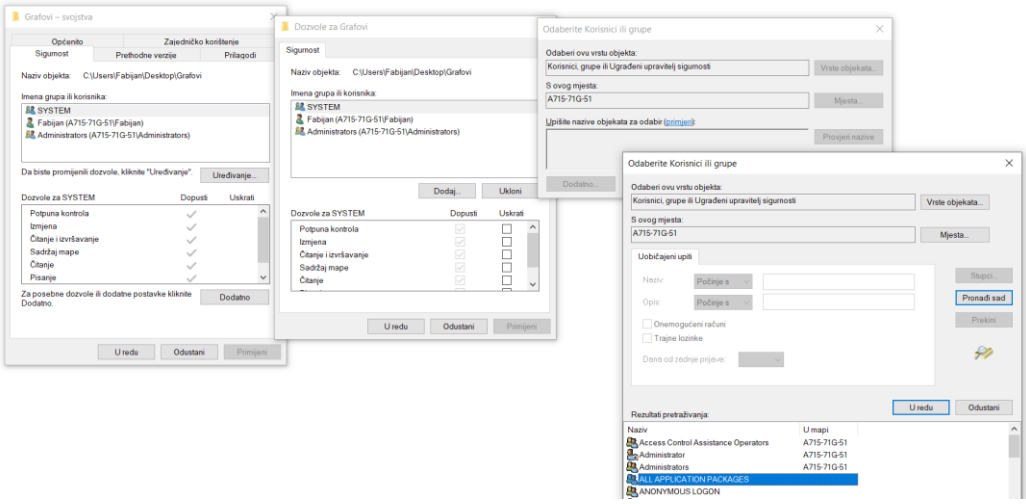
```
EXEC sp_execute_external_script
@language =N'R',
@script=N'library(stringr)'
```



Slika 12. Pokretanje R paketa (izvor: vlastita izrada)

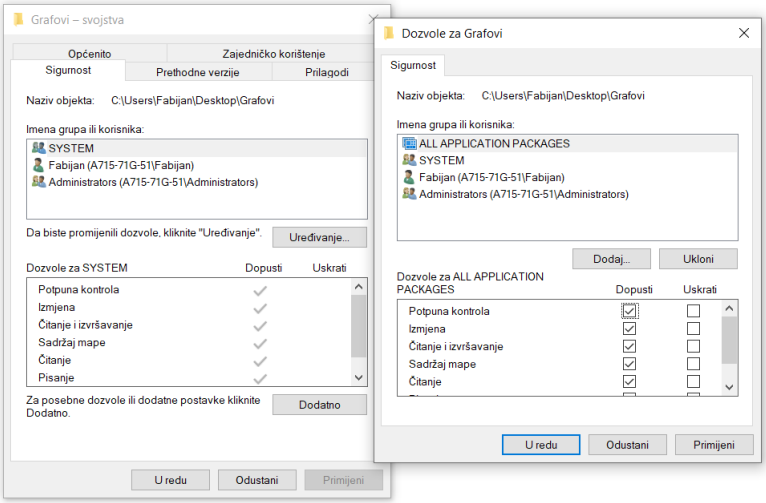
6.5. Generiranje grafova

Kroz rad je nekoliko puta navedeno kako informacijama dobivenim poslovnom analitikom najviše barataju šefovi, menadžeri, uprave, odnosno ljudi na višim razinama u firmama, poduzećima ili organizacijama. Takvim skupinama brojčani podaci mogu biti korisni, no većini je jednostavnije kada se takvi podaci stave u nekakav kontekst ili se grafički prikažu. Na taj način osoba sebi može jednostavnije predočiti značenje određene vrijednosti. Kao i u prethodnom dijelu, potrebno je dati određena prava eksternoj skripti, a to je pisanje i čitanje određenog foldera, odnosno mjesta gdje će se spremati grafovi.



Slika 13. Postavljanje dozvole za čitanje i pisanje u mapu 1/2 (izvor: vlastita izrada)

U nastavku su prikazani i opisani određeni koraci. Na odabranu mapu je potrebno pritisnuti desnim klikom miša te odabrati *Svojstva* (engl. *Properties*), a pod *Općenito -> Atributi* odznačiti *Samo za čitanje.* Nadalje, u istom prozoru pod karticom *Sigurnost* (engl. *Security*) potrebno je pritisnuti gumb *Uređivanje -> Dodaj -> Dodatno -> Pronađi sad*. Iz tog zadnjeg prozora dolje u izborniku je potrebno odabrati *ALL APPLICATION PACKAGES* (Slika 13.). Sada je potrebno potvrditi zadnja dva prozora i na drugom prozoru (Dozvole za <*Mapa*>) pod Dozvole za *ALL APPLICATION PACKAGES* označiti polje *Potpuna kontrola* (Slika 14.). Preostala dva prozora se također potvrde. Sada MS SQL Server, odnosno eksterna skripta ima pristup mapi za pisanje i čitanje.



Slika 14. Postavljanje dozvole za čitanje i pisanje u mapu 2/2 (izvor: vlastita izrada)

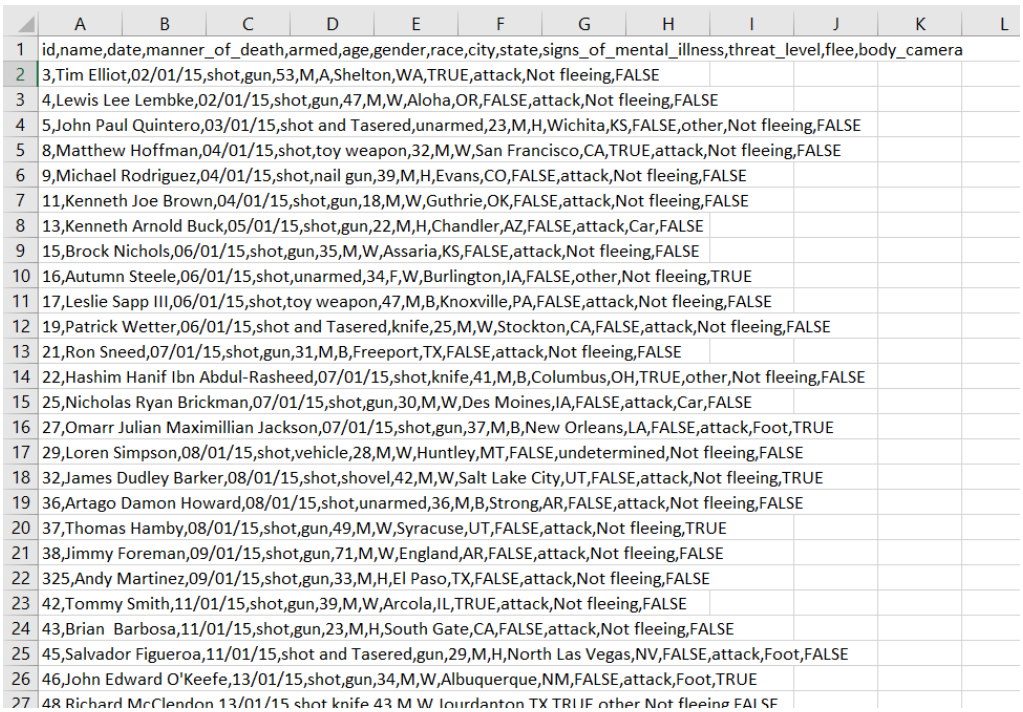
Nakon uspješno izvedenih koraka MS SQL Server 2019 kao i njegovo okruženje je spremo za analizu podataka, stvaranje grafova i predviđanje uz pomoć strojnog učenja. U nastavku slijedi praktičan primjer nad stvarnim podacima gdje će se podaci analizirati i vizualno prikazati. Pomoću njih predočiti će se određeni trendovi i predvidjeti određena stanja.

7. Praktični primjer

U nastavku rada slijedi detaljan i opširan primjer *poslovne analitike* koji će poslužiti za prikaz glavnih pojedinosti MS SQL Server-a 2019 kao i nekih drugih tehnologija koje su neizbježne ukoliko se žele izvući korisne informacije iz dostupnih podataka. Podaci koji će se analizirati prikazuju ubojstva civila od strane policijskih službenika u SAD-u. Takav primjer je odabran iz razloga što je to vrlo aktualna tema za vrijeme pisanja rada i iz razloga što *poslovna analitika* može doprinijeti smanjenju kriminaliteta s obje strane. Također bi analizirani podaci u stvarnome svijetu mogli poslužiti saveznim državama da uoče imaju li potrebe za restrukturiranjem policijskog školovanja i sl. Omogućio bi se pad broja ubojstava izazvanih od policijskih službenika, ali i općenito kriminalnih incidenata ili čak u nekim slučajevima bi se mogla modernizirati oprema ukoliko za time postoji potreba.

7.1. Analiza podataka

Primjer nad kojim će se provesti *poslovna analitika* prikazuje osobe koje su ubijene od strane policijskih dužnosnika. Originalni podaci su dostupni na stranici[16] i stranici[17]. Prvi izvor daje najbitnije podatke (podatke o ubojstvima), dok drugi izvor kontekstno nadopunjuje iste (npr. stopa školovanog stanovništva, rasa).



	A	B	C	D	E	F	G	H	I	J	K	L
1	id,name,date,manner_of_death,armed,age,gender,race,city,state,signs_of_mental_illness,threat_level,flee,body_camera											
2	3,Tim Elliot,02/01/15,shot,gun,53,M,A,Shelton,WA,TRUE,attack,Not fleeing,FALSE											
3	4,Lewis Lee Lembke,02/01/15,shot,gun,47,M,W,Aloha,OR,FALSE,attack,Not fleeing,FALSE											
4	5,John Paul Quintero,03/01/15,shot and Tasered,unarmed,23,M,H,Wichita,KS,FALSE,other,Not fleeing,FALSE											
5	8,Matthew Hoffman,04/01/15,shot,toy weapon,32,M,W,San Francisco,CA,TRUE,attack,Not fleeing,FALSE											
6	9,Michael Rodriguez,04/01/15,shot,nail gun,39,M,H,Evans,CO,FALSE,attack,Not fleeing,FALSE											
7	11,Kenneth Joe Brown,04/01/15,shot,gun,18,M,W,Guthrie,OK,FALSE,attack,Not fleeing,FALSE											
8	13,Kenneth Arnold Buck,05/01/15,shot,gun,22,M,H,Chandler,AZ,FALSE,attack,Car,FALSE											
9	15,Brock Nichols,06/01/15,shot,gun,35,M,W,Assaria,KS,FALSE,attack,Not fleeing,FALSE											
10	16,Autumn Steele,06/01/15,shot,unarmed,34,F,W,Burlington,IA,FALSE,other,Not fleeing,TRUE											
11	17,Leslie Sapp III,06/01/15,shot,toy weapon,47,M,B,Knoxville,PA,FALSE,attack,Not fleeing,FALSE											
12	19,Patrick Wetter,06/01/15,shot and Tasered,knife,25,M,W,Stockton,CA,FALSE,attack,Not fleeing,FALSE											
13	21,Ron Sneed,07/01/15,shot,gun,31,M,B,Freeport,TX,FALSE,attack,Not fleeing,FALSE											
14	22,Hashim Hanif Ibn Abdul-Rasheed,07/01/15,shot,knife,41,M,B,Columbus,OH,TRUE,other,Not fleeing,FALSE											
15	25,Nicholas Ryan Brickman,07/01/15,shot,gun,30,M,W,Des Moines,IA,FALSE,attack,Car,FALSE											
16	27,Omarr Julian Maximillian Jackson,07/01/15,shot,gun,37,M,B,New Orleans,LA,FALSE,attack,Foot,TRUE											
17	29,Loren Simpson,08/01/15,shot,vehicle,28,M,W,Huntley,MT,FALSE,undetermined,Not fleeing,FALSE											
18	32,James Dudley Barker,08/01/15,shot,shovel,42,M,W,Salt Lake City,UT,FALSE,attack,Not fleeing,TRUE											
19	36,Artago Damon Howard,08/01/15,shot,unarmed,36,M,B,Strong,AR,FALSE,attack,Not fleeing,FALSE											
20	37,Thomas Hamby,08/01/15,shot,gun,49,M,W,Syracuse,UT,FALSE,attack,Not fleeing,TRUE											
21	38,Jimmy Foreman,09/01/15,shot,gun,71,M,W,England,AR,FALSE,attack,Not fleeing,FALSE											
22	325,Andy Martinez,09/01/15,shot,gun,33,M,H,El Paso,TX,FALSE,attack,Not fleeing,FALSE											
23	42,Tommy Smith,11/01/15,shot,gun,39,M,W,Arcola,IL,TRUE,attack,Not fleeing,FALSE											
24	43,Brian Barbosa,11/01/15,shot,gun,23,M,H,South Gate,CA,FALSE,attack,Not fleeing,FALSE											
25	45,Salvador Figueroa,11/01/15,shot and Tasered,gun,29,M,H,North Las Vegas,NV,FALSE,attack,Foot,FALSE											
26	46,John Edward O'Keefe,13/01/15,shot,gun,34,M,W,Albuquerque,NM,FALSE,attack,Foot,TRUE											
27	48 Richard McClendon 13/01/15 shot knife 43 M W Jourdanton TX TRUE other Not fleeing FALSE											

Slika 15. Prikaz izgleda izvornih podataka (izvor: vlastita izrada)

[16] Izvor (pristupano 31.07.2020): https://github.com/washingtonpost/data-police-shootings
[17] Izvor (pristupano 31.07.2020): https://www.kaggle.com/kwullum/fatal-police-shootings-in-the-us

Prva i najbitnija datoteka „*fatal-police-shootings-data.csv*“ prikazuje osobe i okolnosti u kojima su one ubijene, bio to kriminalac ili nedužna osoba. U nastavku su prikazani i ukratko objašnjeni pojedini atributi navedenog dataseta:

- **id** – identifikacijska oznaka pojedine osobe
- **name** – ime i prezime ubijene osobe
- **date** – datum incidenta
- **manner_of_death** - na koji način je osoba ubijena
- **armed** - da li je osoba prilikom incidenta bila naoružana
- **age** - dob ubijene osobe
- **gender** - spol ubijene osobe
- **race** - rasa ubijene osobe (5 rasa: W-bijelac, B-crnac, A-azijat, H-amerikanci podrijetlom sa španjolskog govornog područja, N-nativni amerikanci, O-bliski istok, X-nije određeno)
- **city** – grad u kojem se incident dogodio
- **state** – savezna država incidenta
- **sings_of_mental_illness** – da li osoba ima mentalnih problema
- **thread_leve**l – do koje je razine osoba bila opasna prilikom incidenta
- **flee** – da li je osoba bila u bijegu
- **body_camera** – da li je policajac imao uključenu kameru na svom tijelu prilikom incidenta

Bitno je izdvojiti atribut *sings_of_mental_illness*. U praktičnom dijelu će se taj podatak koristiti u sklopu strojnog učenja, tj. on se predviđa. Time se mogu nadopuniti novi zapisi, a stari se mogu provjeriti. Čak je moguće ustanoviti greške, gdje osoba možda nema nikakve znakove mentalnih problema ili obrnuto.

Idući dataset je „*ShareRaceByCity.csv*“ gdje *Geographic area* označava saveznu državu, *City* označava grad. Ostali podaci predstavljaju postotke zastupljenosti pojedine rase u pojedinom gradu (*share_white, share_black, share_native_american, ...*)

Nakon toga, dolazi dataset „*MedianHouseholdIncome2015.csv*“ koji opisuje prosječnu zaradu po kućanstvu u određenom gradu. *Geographic Area* kao i prethodno, označava saveznu državu, *City* označava grad, a *Median Income* prosječna zarada po kućanstvu.

Na sličan način funkcionira dataset „*PercentagePeopleBelowPovertyLevel.csv*“ u kojem se isto nalazi atributi *Geographic Area* i *City*. Zadnji podatak, *poverty_rate,* opisuje postotak ljudi ispod granice siromaštva u određenom mjestu.

Zadnji bitan dataset je „*PercentOver25CompletedHighSchool*.csv“ koji ima iste atribute kao i prethodna dva dataseta, osim zadnjeg atributa, *percent_completed_hs* koji opisuje postotak ljudi preko 25 godina koji su završili srednju školu.

Svi ti podaci mogu dati jasan prikaz pozadine ubijenih osoba. Takvi podaci mogu pomoći prilikom identificiranja kritičnih lokacija na kojima se događaju takva ubojstva i na temelju kojih se mogu uvesti nove mjere, ustroji ili slično što bi moglo pomoći prilikom smanjenja budućih incidenata.

Izvorni podaci su dodatno prošireni podacima o populaciji pojedinih saveznih država gdje su izvori dataset[18] i stranica[19]. Sveukupno postoji 5417 zapisa o ubojstvima u razdoblju od 2015. do 2020. godine (18.06.2020). Podaci se iz .csv oblika mogu lako unijeti u bazu podataka iz koje onda slijedi daljnja obrada.

7.2. Unos, ispravak i dopuna podataka

Prethodno navedeni podaci se unesu u novokreiranu bazu podataka koracima opisanim u cjelini 6.2 te nakon toga slijedi ispravak greški. U nekim slučajevima mogu nedostajati zapisi kao npr. u datasetu „*ShareRaceByCity*“ gdje za određene gradove nisu navedeni postoci, a na takvo mjesto je postavljen znak „(X)“ umjesto postotka. U nastavku je prikazan SQL kod koji ispravlja takve greške.

```
UPDATE [dbo].[OrigigiShareRaceByCity] SET
share_hispanic='-1'
WHERE
share_hispanic='(X)'
```

Također se mogu pojaviti slučajevi gdje tip podatka treba zamijeniti radi jednostavnijeg korištenja kao npr. True/False podaci koji se mogu zapisati u binarnom obliku kao 0 ili 1. Ispod je prikazan takav primjer.

```
UPDATE [dbo].[Original_data]
SET signs_of_mental_illness='0'
WHERE signs_of_mental_illness='False'
```

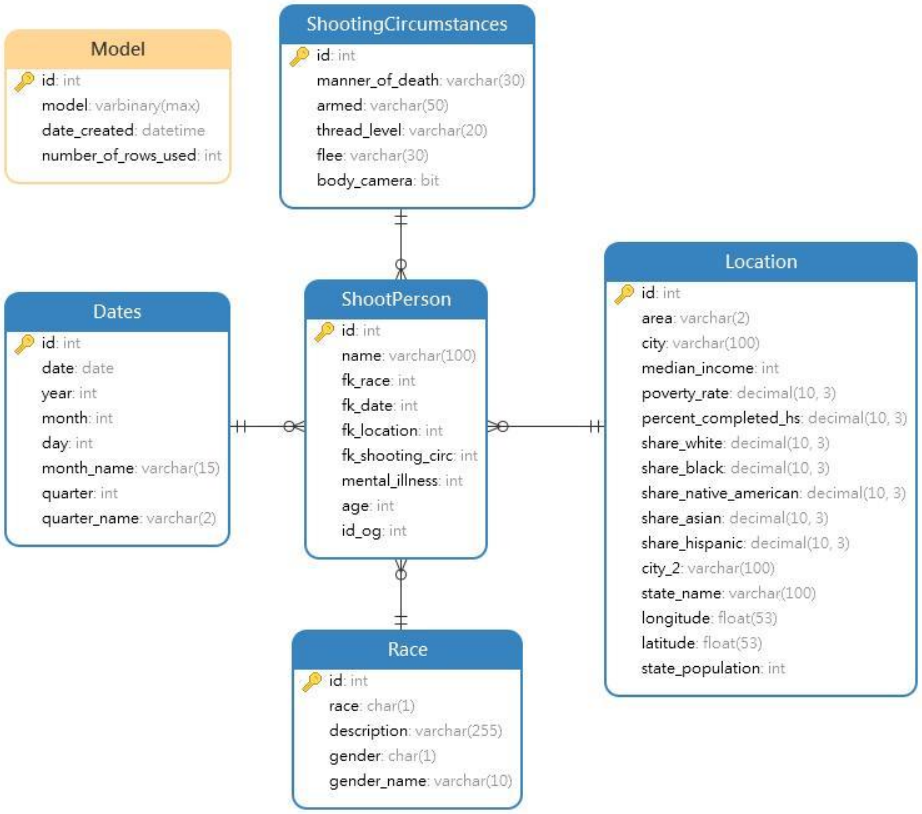
Na sličan način se svi ostali podaci isprave i/ili nadopune, no bitno je napomenuti kako se koraci mogu znatno razlikovati ovisno o samom sastavu podataka. Uglavnom se koriste osnovne SQL naredbe slične onima koje opisuje Rabuzin (Uvod u SQL, 2011), pretežito *UPDATE* naredbe. Kada bi se navedeni koraci promatrali kroz skladišta podataka tada bi oni predstavljali *Transform* korak kod ETL procesa.

[18] Izvor (pristupano 31.07.2020): https://www.kaggle.com/giodev11/usstates-dataset
[19] Izvor (pristupano 31.07.2020): https://worldpopulationreview.com/states/

7.3. Kreiranje skladišta podataka

Nakon što su izvorni podaci ispravljeni i nadopunjeni, oni se mogu strukturirati u skladište podataka kroz razne upite. U svrhe rada, skladište podataka biti će jednostavnije prirode radi održavanja performansi na prihvatljivoj razini. Povećanjem količine podataka i korištenjem određenih procesa, *Hardware* se počinje lošije nositi s postavljenim upitima. Kasnije je prikazana cjelina o strojnom učenju koja zahtjeva iznimno puno resursa.

Cjelokupno skladište podataka se sastoji od 4 dimenzijske tablice, jedne činjenične i jedne zasebne tablice za pohranu modela nastalog strojnim učenjem. U nastavku slijedi detaljan opis i prikaz skladišta podataka. (Rabuzin, 2014, str. 95-118)



Slika 16. Model skladišta podataka (izvor: vlastita izrada)

U nastavku slijedi opis pojedinih tablica kao i njenih atributa. Sve tablice sadrže jedinstveni identifikator odnosno id.

- **ShootPerson** – činjenična tablica koja sadrži podatke o ubijenoj osobi kao što su ime i prezime, broj godina, psihička bolest, originalni id iz izvornih podataka te vanjske ključeve na dimenzijske tablice.
- **Dates** – dimenzijska tablica koja sadrži podatke o svim potrebnim datumima (dan, mjesec, godina, kvartal) koji opisuju vrijeme događaja incidenta

- **Location** – dimenzijska tablica koja opisuje mjesto događaja incidenta, odnosno saveznu državu i grad (puni i skraćeni naziv). Također sadrži postotke za broj ljudi koji imaju srednjoškolsko obrazovanje, one koji su ispod granice siromaštva, te postotke odnosno udio pojedinih rasa na tim mjestima. Tablica također sadrži podatke o populaciji savezne države te središnju točku savezne države (longitude, latitude) koja je potrebna za ispis grafova.
- **Race** – dimenzijska tablica koja opisuje rasu ubijene osobe i spol.
- **ShootingCircumstances** – dimenzijska tablica koja opisuje okolnosti u kojima je osoba ubijena, na koji način, da li je osoba bila naoružana, je li bila u bijegu, koje je razine prijetnje osoba i je li policajac imao uključenu kameru.
- **Model** – zadnja tablica koja nije vezana za niti jednu tablicu. Sadrži model koji je u stanju opisati da li osoba ima mentalnih problema ili ne. Tablica se puni podacima nastalih strojnim učenjem.

7.4. Analiza podataka uz MS SQL Server 2019 i R jezik

Unutar navedene podcjeline slijedi analiza podataka uz pomoć MS SQL Server 2019 i R jezika gdje će rezultati biti prikazani u grafičkom obliku. Uz same grafove biti će priložena procedura, odnosno kod zaslužan za analizu podataka i stvaranje grafova. Procedure su pisane na temelju opće poznate strukture s nekoliko manjih preinaka vezanim za SQL Server bazu podataka (Rabuzin, 2014, str. 45-56). Nisu korištene nikakve napredne značajke procedura osim što se koristi R skripta unutar njih.

7.4.1. Prvi prikaz

U prvom prikazu predočeno je koje savezne države imaju najveći broj smrtnih slučajeva izazvanih od strane policijskih službenika. Takvi podaci su relativni i ne trebaju uvijek prikazivati stvarno stanje, budući da svaka savezna država ima različiti broj stanovništva.

Kako bi se stvorio prvi prikaz, potrebno je prikupiti određene podatke iz skladišta podataka i proslijediti u eksternu R skriptu. Svi potrebni koraci su napravljeni unutar SQL procedure. Takva bi se procedura, uz malo modifikacije, mogla koristiti unutar određenih sustava i aplikacija kako bi korisnik mogao dobiti uvid u najnovije podatke. Tako bi u ovome slučaju pripadajuće tijelo savezne države ili vlada SAD mogla putem svoje aplikacije zatražiti aktualni ispis podataka. Time bi se pozvala procedura i stvorili novi uvidi u analizirane podatke, bez potrebe za pisanjem novog R koda ili angažmana analitičara što uvelike pojednostavljuje i ubrzava proces odlučivanja. Također se kroz godine mogu dobiti konzistentni podaci koji bi se mogli međusobno uspoređivati.

U nastavku je prikazana i objašnjena SQL procedura i R skripta koje su zaslužne za analizu i ispis podataka. Potpuni izvorni kod je vidljiv na web stranici (str. 57). (eriqande.github.io, 2020)

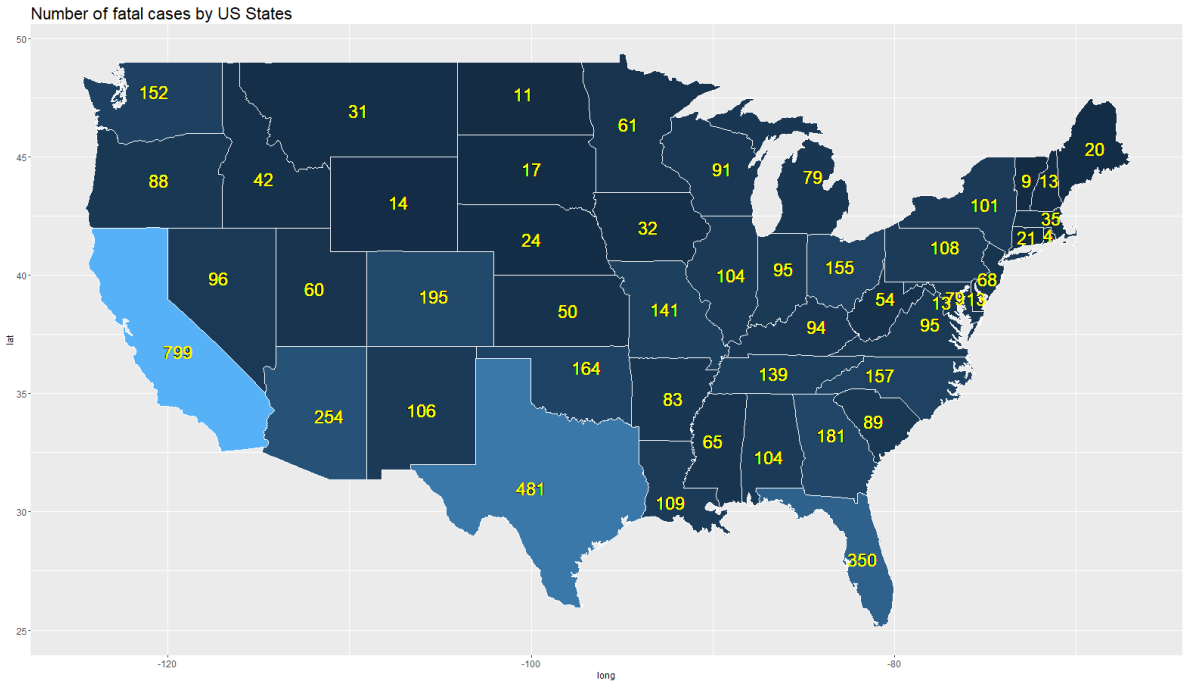
```
CREATE PROCEDURE [dbo].[MapaSADa]
AS
BEGIN
  SET NOCOUNT ON;
  DECLARE @query nvarchar(max) =
  N'   SELECT sp.mental_illness, sc.manner_of_death, sc.armed, sp.age, r.gender,
r.race, l.area, l.city, sc.thread_level,sc.flee,l.state_name
       FROM [dbo].[ShootPerson] sp
       JOIN [dbo].[ShootingCircumstances] sc ON sc.id=sp.fk_shooting_circ
       JOIN [dbo].[Race] r ON sp.fk_race=r.id
       JOIN [dbo].[Location] l ON l.id=sp.fk_location'

DECLARE
@Table1_Input NVARCHAR( MAX ) = 'SELECT DISTINCT area, state_name, longitude,
latitude FROM [dbo].[Location]',
@Table1_Data VARBINARY( MAX )

  EXECUTE sp_execute_external_script
    @language = N'R',
    @script = N'

        if( nrow(InputDataSet) == 0 )
            stop("Invalid data passed in")

        # Read in the sql table, serialize it to an output string
        Output <- serialize(InputDataSet, NULL)
    ',
    @input_data_1 = @Table1_Input,
    @params = N'@Output VARBINARY( MAX ) OUTPUT',
    @Output = @Table1_Data OUTPUT;

  EXECUTE sp_execute_external_script @language = N'R',
                                     @script = N'
      library(ggplot2)
      library("maps")
      library(ggmap)
      library(mapdata)

      #--- Ulazni podaci ---
      databaseTable <- InputDataSet
      locationDB <- unserialize(Table1_Data)

      print(databaseTable)
      print(locationDB)

      #--- Strukturiranje podataka ---
      table <- data.frame(
        databaseTable$age,
        databaseTable$manner_of_death,
        databaseTable$armed,
        databaseTable$gender,
        databaseTable$race,
        databaseTable$area,
        databaseTable$mental_illness,
        databaseTable$thread_level,
        databaseTable$flee,
        databaseTable$city,
        databaseTable$state_name
      )
```

```
        names(table) <- c("age", "manner", "armed", "gender", "race", "state",
        "sings", "thread", "flee", "city","state_name")
        #--- Analiza podataka ---
        data <- table(table$state_name)
        data <- data.frame(data)
        names(data) <- c("region", "number")

        #--- Dohvaćanje podataka o saveznim državama ---
        states <- map_data("state")
        states <- data.frame(states)

        #--- Središnje točke saveznih država ---
        locationDB <- data.frame(
          locationDB$area,
          locationDB$state_name,
          locationDB$longitude,
          locationDB$latitude
        )
        names(locationDB)<-c("state","region","longitude","latitude")

        #--- Kreiranje grafa ---
        numberOfCasesPerState <- merge(data,states,by="region")
        stateWithLongAndLat <- merge(locationDB,numberOfCasesPerState,by="region")
        head(states$region)
        p <- ggplot(data = numberOfCasesPerState) +
        geom_polygon(aes(x = long, y = lat, fill = number, group = group), color =
        "white") + geom_text( data=stateWithLongAndLat, aes(longitude, latitude,
        label = number), size=8,color="black", fontface = "bold") + geom_text(data =
        stateWithLongAndLat, aes(longitude, latitude, label = number),
        size=8,color="yellow")+ coord_fixed(1.3) + guides(fill=FALSE)

        #--- Ispis grafa ---
        print(getwd())
        png(filename="C:/Users/Fabijan/Desktop/Diplomski/US_Map_killings_by_state.png
        ", width=1600, height=900, units = "px")
        p <- p + labs(title = "Number of fatal cases by US States")
        p <- p + theme(plot.title = element_text(size=22))
        p <- p + theme(axis.text=element_text(size=12),
              axis.title=element_text(size=12))
        plot(p)
        dev.off();
    ',
    @input_data_1 = @query,
    @params = N'@Table1_Data VARBINARY(MAX)',
    @Table1_Data = @Table1_Data
 SET NOCOUNT OFF
END
GO
```

Kreirana procedura ne prima nikakve parametre niti ih prosljeđuje van budući da ne postoji potreba za time. Na početku procedure se deklariraju dva upita, jedan za uzimanje relevantnih podataka o broju slučajeva, a u drugome se uzimaju podaci o lokacijama. Također se deklarira jedna pomoćna varijabla za prihvat rezultata. U samoj proceduri postoje dvije eksterne R skripte iz razloga što jedna eksterna skripta ne može prihvatiti 2 ulazna dataseta, što je veliki nedostatak. Iz tog razloga se iz prve eksterne R skripte uzimaju podaci i stavljaju u pomoćnu varijablu koja se kao parametar prosljeđuje u drugu eksternu R skriptu. Jedna moguća modifikacija procedure je ta da se za ulazne parametre u procedure proslijede datumi za koje se želi napraviti ispis.

Druga skripta služi za analizu podataka i kreiranje grafa. Na početku druge eksterne R skripte je vidljivo pozivanje nužnih biblioteka koje su prethodno preuzete (vidi 6.4). Nakon toga slijedi prihvat ulaznih podataka, koji se onda strukturiraju u tablice (*data.frame*). Sama analiza podataka, odnosno u ovome slučaju grupiranje i prebrojavanje, odvija se u jednoj liniji (`table(table$state_name)`) koja broji pojavljivanje redova u tablici na temelju naziva savezne države SAD-a. Takav korak je moguće izvesti i uobičajenim SQL upitom. Nakon toga slijedi dohvat podataka o saveznim državama radi iscrtavanja grafa, a potrebni se podaci povlače iz uvezenog paketa (`library(mapdata)`).

Zadnjih nekoliko linija je zaslužno za stvaranje grafa, postavljanje potrebnih naziva na osi, naslove i sl. kao i za rezultat analize podataka. Za crtanje saveznih država potrebne su napredne funkcije koje su dostupne iz uvezenog paketa (`library(ggplot2)`). Nakon što je graf postavljen slijedi crtanje i pohrana na memorijski medij ili po potrebi u neku novu tablicu. Naredba za stvaranje slike je `png(filename="direktorij/naziv datoteke")` iza koje slijedi crtanje grafa (`plot(p)`) i pohrana (`dev.off()`). Osim `png` naredbe može se koristiti i `pdf`, `jpg`, `svg` i slične naredbe. Ispod je prikazan rezultat pokretanja skripte gdje je naredba „`EXEC dbo.MapaSADa`“.



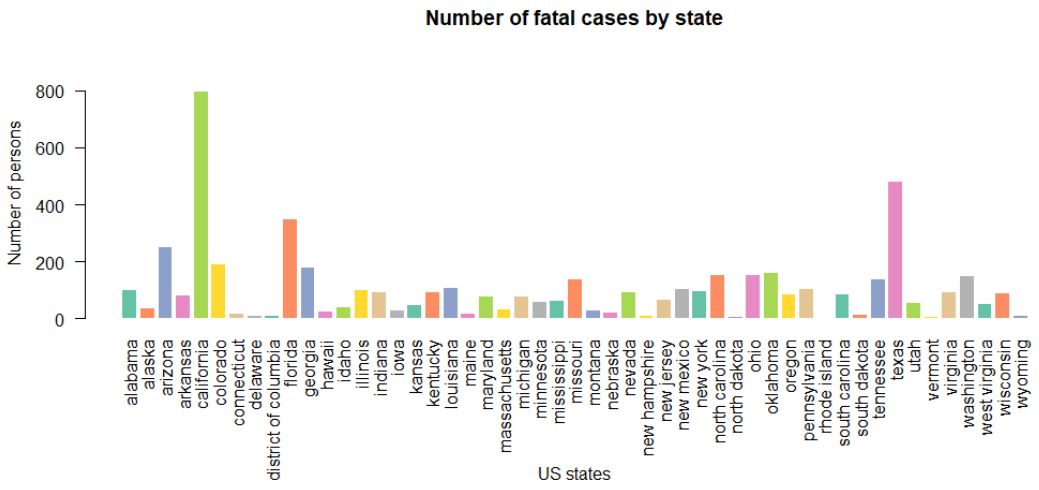
Slika 17. Prikaz rezultata procedure (izvor: vlastita izrada)

Tamno plava boja na grafu prikazuje one savezne države koje su u razdoblju od 2015. do 2020. godine imale najmanje smrtnih incidenata izazvane od strane policijskih službenika, a svijetloplave predstavljaju one koje su imale najviše slučajeva. Naravno brojevi unutar saveznih država predstavljaju konkretan broj takvih slučajeva.

Ukoliko korisnik želi uobičajeniji prikaz, tipa stupčasti graf, onda je procedura nešto jednostavnija. Nema potrebe za dvije eksterne R skripte, samo je jedna dovoljna. Također je dovoljno uvoziti manje R paketa (samo `library(RColorBrewer)`). Analiza podataka ostaje jednaka, a crtanje se zamijeni sljedećim R kodom.

```
library(RColorBrewer)
coul <- brewer.pal(8, "Set2")

par(mai=c(2,1,1,1)
barplot(data$number,
        main = "Number of fatal cases by US state",
        xlab = "",
        ylab="Number of persons",
        ylim = c(0, max(data$number) + 80),
        col=coul,
        border="white",
        names=data$region,
        las=2,
        cex.names = 1.00)


mtext("US states", side=1, line = 7, las=1)
```

Rezultat gore navedenog R koda vidljiv je na slici ispod:



Slika 18. Prikaz alternativnog rezultata procedure (izvor: vlastita izrada)

Ovakvi podaci su relativno slabo upotrebljivi budući da svaka savezna država ima drugačiji broj stanovnika. Stoga ne čudi da u Kaliforniji i Teksasu ima najviše slučajeva jer imaju najveću populaciju. Podaci se trebaju staviti u kontekst s populacijom kako bi se uočilo u kojoj saveznoj državi policija stvara najviše smrtonosnih incidenata (broj smrtnih slučajeva po glavi stanovnika). Iz tog razloga će se gornji graf replicirati, ali se u kontekst ubacuje populacija. Kada bi se stvarala SQL procedura, tada bi ona trebala sadržavati dvije eksterne R skripte gdje jedna služi za dohvat populacije, a u drugoj se vrši analiza podataka.

U nastavku je prikazan R kod za analizu, odnosno određivanje onih saveznih država koje imaju najviše smrtonosnih slučajeva po broju stanovnika izazvanih od strane policijskih službenika.

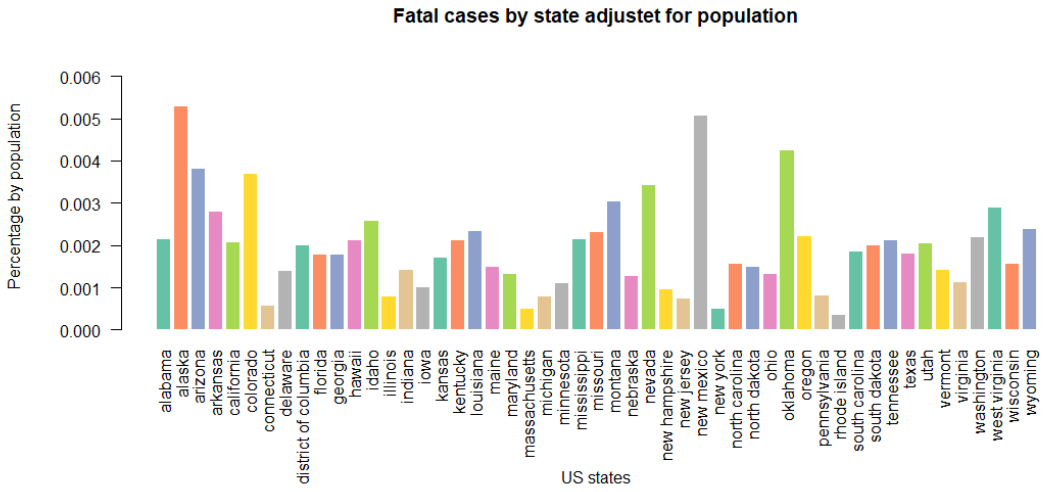
SQL upit:

```
DECLARE @query nvarchar(max) =
N'SELECT select distinct area,state_name,state_population FROM
[dbo].[Location]
```

R skripta unutar eksterne skripte:

```
# --- Analiza podataka ---
data <- table(table$state_name)
data <- data.frame(data)
names(data) <- c("region", "number")

# --- Strukturiranje podataka ---
populationDB <- unserialize(Table1_Data)
populationDB <- data.frame(
  populationDB$area,
  populationDB$state_name,
  populationDB$state_population
)
names(populationDB)<-c("state","region","population")

populationData <- merge(data,populationDB,by="region")
options("scipen" = 100, "digits" = 4)

# --- Prilagođavanje podataka stanovništvu ---
populationData <-
cbind(populationData,((populationData$number/populationData$population)*100))
names(populationData) <- c("region","number","state","population","perPop")

par(mai=c(2,1.5,1,1))
barplot(populationData$perPop,
        main = "Fatal cases by state adjustet for population",
        xlab = "",
        ylab="",
        ylim = c(0, max(populationData$perPop) + 0.001),
        col=coul,
        border="white",
        names=data$region,
        las=2,
        cex.names = 1.00)

mtext("US states", side=1, line = 7, las=1)
mtext("Percentage by population", side=2, line = 5, las=3)
```

Rezultat gore navedenog koda vidljiv je na slici 19. Vidljiva je velika razlika između prethodna dva grafa. Više nisu Kalifornija i Teksas one savezne države koje su najizraženije već su to New Mexico, Oklahoma i Alaska jer se u obzir uzima stanovništvo. Na temelju toga bi te savezne države ili druga organizacijska tijela mogla donijeti prijeko potrebne odluke, kao npr. nove mjere, školovanja ili slično. Time bi se omogućilo smanjenje broja smrtonosnih incidenata u navedenim saveznim državama. Također bi se novim mjerama mogao suzbiti eventualni kriminalitet što je jedna od točki kojima se bavi poslovna analitika, bilo to ovakav klasični kriminalitet sličan ovome ili cyber kriminalitet.



Slika 19. Prikaz alternativnog rezultata procedure prilagođenog stanovništvu (izvor: vlastita izrada)

Ovo je prvi praktični primjer od nekoliko njih. Dosadašnji primjeri su jednostavne prirode te daju uvid u određene brojeve incidenata po saveznim državama što zapravo predstavlja *opisnu analitiku*. Prikazuje i opisuje prošlo stanje što je zapravo skraćena definicija *opisne analitike*. U ovome slučaju je takav prikaz i više nego dovoljan, no nekada vodstvo može zahtjevati više informacija, tada nastupaju ostale grane, kao npr. prediktivna analitika koja je opisana u nastavku rada.

7.4.2. Drugi prikaz

Za vrijeme pisanja rada aktualna tema SAD-a pa čak i do neke mjere u svijetu je rasizam i nasilje izazvano od strane policijskih službenika i općenito rasizam među ljudima. Postavlja se pitanje da li je to ustvari istina ili ne, jesu li američki policijski službenici skloniji nasilju i ubojstvu nad osobom druge rasne skupine. U SAD-u se smatra da su tamnoputi državljani najčešća meta nasilja i ubojstva od strane policijskih službenika. Iz takvih razloga se u puku stvara nemir koji kasnije izaziva neželjene nerede.

U nastavku slijedi razrađeni primjer kojim se nastoji otkriti koja je rasna skupina najviše ugrožena u kojoj saveznoj državi. Ovakvi uvidi u podatke bi mogli poslužiti suzbijanju širenja lažnih informacija kako je određena skupina stanovništva više ugrožena od druge. Time bi se mogli suzbiti mogući neredi i nemiri. Također bi se mogla postići ravnopravnost među određenim skupinama stanovništva. U primjeru je korišten SQL upit kako bi se podaci relevantno strukturirali da bi se dobile one rasne skupine koje su najugroženije u pojedinim saveznim državama. Također je kreirana privremena pomoćna tablica u kojoj su ti rezultati zabilježeni i koji se kasnije koriste u eksternoj R skripti za izradu grafa, odnosno izvještaja. Izvorni kod SQL procedure je moguće preuzeti na web stranici (str. 57).

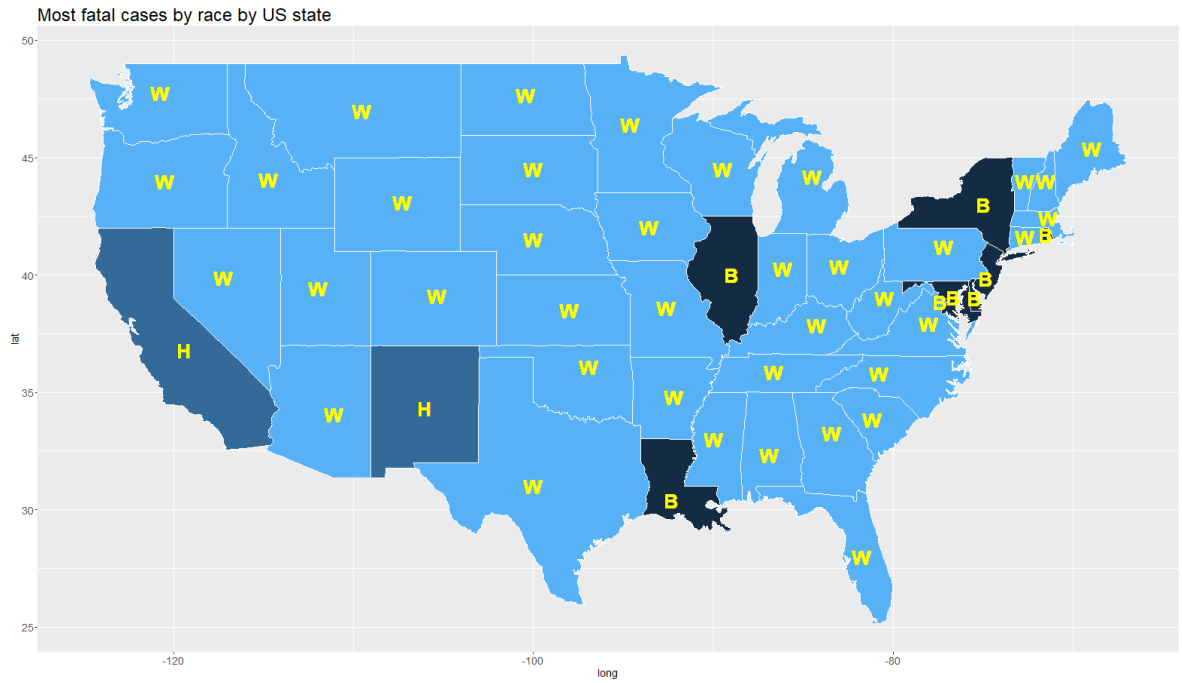
```
CREATE PROCEDURE [dbo].[KreiranjeTEMPTablice]
AS
BEGIN
  SET NOCOUNT ON;
       DROP TABLE IF EXISTS TEMP
       CREATE TABLE TEMP (
              ct INT,
              race CHAR(1),
              description VARCHAR(30),
              area VARCHAR(2),
              state_name VARCHAR(100),
              long FLOAT,
              lat FLOAT);

       ;WITH rezultat AS (
              SELECT  r.race, r.description, COUNT(*) as ct,l.area,
              l.state_name, l.longitude AS long, l.latitude AS lat,
              ROW_NUMBER() OVER(PARTITION BY l.area ORDER BY COUNT(*) DESC) AS
              rk
              FROM [dbo].[ShootPerson] sp JOIN [dbo].[Race] r
              ON r.id=sp.fk_race
              JOIN [dbo].[Location] l
              ON sp.fk_location=l.id
              GROUP BY r.race, r.description, l.area, l.state_name,
              l.longitude, l.latitude)

       INSERT INTO TEMP SELECT ct, race, description, area, state_name, long,
       lat FROM rezultat s1 WHERE s1.rk = 1 ORDER BY state_name;

  SET NOCOUNT OFF
END
```

U nastavku je prikazana eksterna R skripta pomoću koje se analizirani podaci grafički prikazuju. Procedura je znatno jednostavnija u odnosu na prijašnje budući da se analitički dio nalazi u SQL kodu, a dobiveni rezultati su izrazito zanimljivi. Izvorni kod je prikazan na web stranici (str. 57).

```
CREATE PROCEDURE [dbo].[USMapByPoliceRaceKilling]
AS
BEGIN
  SET NOCOUNT ON;
  DECLARE @query nvarchar(max) =  N'SELECT * FROM TEMP'

  EXECUTE sp_execute_external_script @language = N'R',
                                     @script = N'
       library(ggplot2)
       library("maps")
       library(ggmap)
       library(mapdata)

       raceDB <- InputDataSet

       # --- Dobavljanje podataka o saveznim državama ---
       states <- map_data("state")
       states <- data.frame(states)

       raceDB <- data.frame(
         raceDB$ct,
         raceDB$race,
         raceDB$description,
         raceDB$area,
         raceDB$state_name,
         raceDB$long,
         raceDB$lat)
```

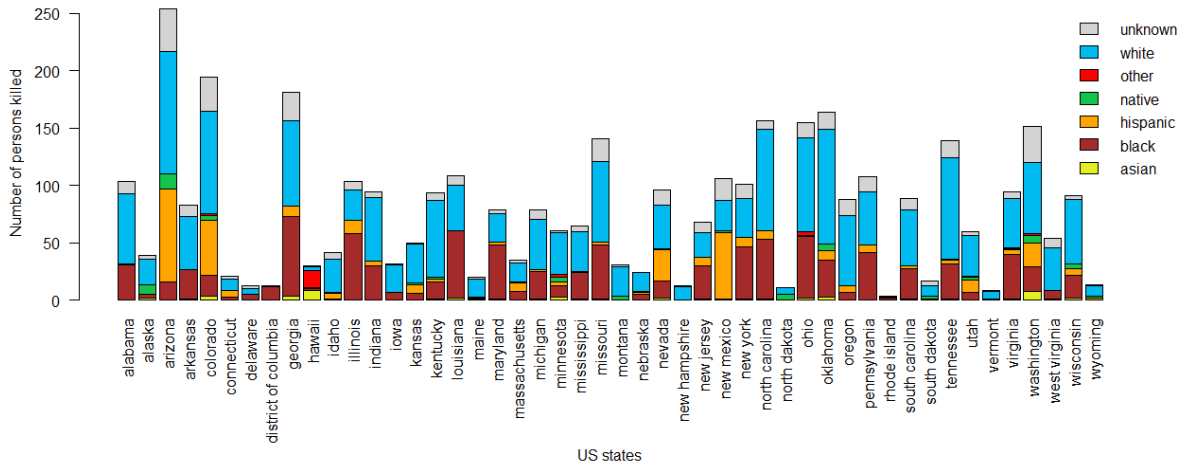
```
        names(raceDB)<-c("ct","race","desc","area","region","long1","lat1")

        numberOfCasesByRacePerState <- merge(raceDB,states,by="region")
        numberOfCasesByRacePerState <-
        numberOfCasesByRacePerState[order(numberOfCasesByRacePerState$group,numberOfC
        asesByRacePerState$order),]
        a <- as.numeric(factor(numberOfCasesByRacePerState$race))

        p <- ggplot(data = numberOfCasesByRacePerState) +
        geom_polygon(aes(x = long, y = lat, fill = a, group = group), color =
        "white") + geom_text(data=numberOfCasesByRacePerState, aes(long1, lat1, label
        = race), size=5,color="yellow", fontface = "bold") + geom_text(  data =
        numberOfCasesByRacePerState, aes(long1, lat1, label = race),
        size=5,color="yellow")+ coord_fixed(1.3) + guides(fill=FALSE)

        png(filename="C:/Users/Fabijan/Desktop/Diplomski/US_Map_killings_by_race_in_s
        tate.png", width=1600, height=900, units = "px")
        p <- p + labs(title = "Most fatal cases by race by US state")
        p <- p + theme(plot.title = element_text(size=22))
        p <- p + theme(axis.text=element_text(size=12),
              axis.title=element_text(size=12))
        plot(p)
        dev.off();
    ',
    @input_data_1 = @query
 SET NOCOUNT OFF
END
GO
```

Kao rezultat gore prikazanog SQL i R koda dobiven je graf sa slike ispod. Svjetlo plava boja i „*W*“ označavaju savezne države u kojima su policijski službenici najviše usmrtili osobe bjelačke rasne skupine, a tamno plava i „*B*“ označava one u kojima su najviše usmrćene tamnoputе osobe.



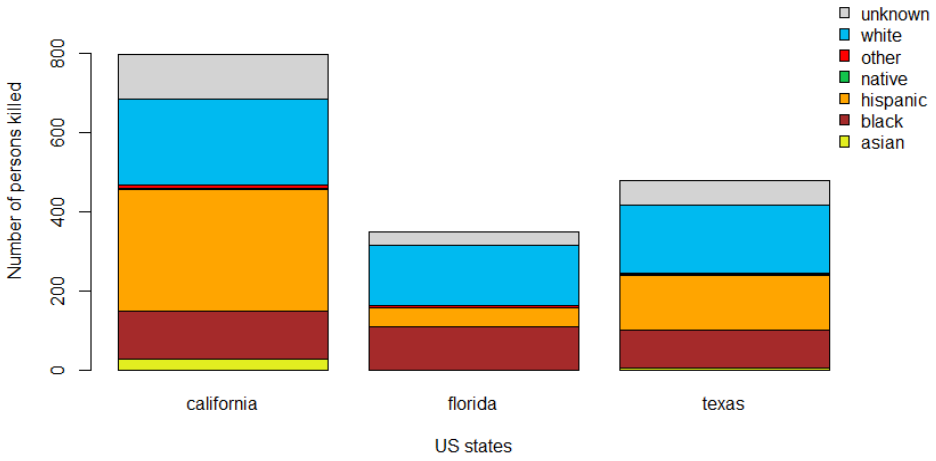
Slika 20. Prikaz najugroženijih rasnih skupina od strane policijskih službenika (izvor: vlastita izrada)

Jasno je vidljivo kako je u većini saveznih država najugroženija bjelačka rasna skupina, a u ostalim saveznim državama označenim sa „*B*“ najugroženija crnačka rasna skupina. Takvi podaci su izrazito važni prilikom javnog objavljivanja i educiranja stanovništva pomoću kojih se mogu izbjeći razni neredi i rasistički ispadi. Takvim podacima se može postići ravnopravnost među ljudima u jednoj državi koja je rasno raznolika kao što je to SAD. Točniji prikaz gore navedene grafike je vidljiv ispod, unutar dva stupčasta grafa



Slika 21. Prikaz najugroženijih rasnih skupina u saveznim državama (izvor: vlastita izrada)

Preostale tri savezne države su zasebno prikazane, budući da one imaju najveći broj prijavljenih slučajeva. Na oba grafa se jasno može uočiti kako je najugroženija bjelačka rasna skupina, a u rjeđem slučaju kada nije (stupac 1), tada je direktno na drugome mjestu.



Slika 22. Prikaz najugroženijih rasnih skupina u tri preostale savezne države (izvor: vlastita izrada)

U nastavku su prikazani SQL i R isječci koda koji se mogu koristiti kroz eksternu skriptu u SQL serveru ili zasebno unutar R Studio-a ili drugog alata.

SQL upit:

```
SELECT r.race,l.area,l.state_name FROM [dbo].[ShootPerson] sp
JOIN [dbo].[Race] r ON r.id=sp.fk_race
JOIN [dbo].[Location] l ON l.id=sp.fk_location
```

R kod:

```
raceAreaDB <- InputDataSet
raceAreaDB <- data.frame(

  raceAreaDB$race,
  raceAreaDB$area,
  raceAreaDB$state_name
)
names(raceAreaDB)<-c("race","area","s_name")

counts <- table( raceAreaDB$race,raceAreaDB$s_name)
counts <- counts[,-5]
counts <- counts[,-43]
counts <- counts[,-9]

rownames(counts) <-
c("unknown","white","other","native","hispanic","black","asisan")
counts

par(mai=c(2.5,1,1,1))
barplot(counts, main = "US Fatal Police Shootings by race in state",
        col = c( "#e2ee20", "brown", "orange", "#13c34c", "red",  "#00baf0",
"#D3D3D3"), ylab="Number of persons killed", legend = rownames(counts),
args.legend = list(x = "topright", bty = "n", inset=c(-0.08, 0)),
ylim = c(0, max(counts) + 150), las=2)
mtext("US states", side=1, line = 8, las=1)
```

Ovo je relativno jednostavan prikaz, općenito za sve savezne države. Još korisniji prikaz bi bio kada bi se u kontekst stavio broj stanovnika gdje bi se dobili postotni udjeli. Također bi bio koristan sličan takav prikaz na razini savezne države, tj. gradova. Onda bi vodstvo imalo jasniju sliku gdje se najviše incidenata događaja po broju stanovnika. Time bi se mogle provesti razne reforme na tim mjestima kako bi se mogao smanjiti broj smrtnih incidenata.

7.4.3. Treći prikaz

Dosad su bili prikazani primjeri zasnovani na *opisnoj analitici*, no bilo bi korisno iskoristiti dostupne podatke prilikom uočavanja trendova, odnosno prognoziranja bliže budućnosti. Tako se u ovome slučaju može uočiti dali iz godine u godinu broj ubijenih osoba od strane policijskih službenika opada, raste ili ostaje isti. U nastavku su prikazani pripadajući SQL upit i R kod zaslužni za generiranje prikaza, odnosno izvještaja.

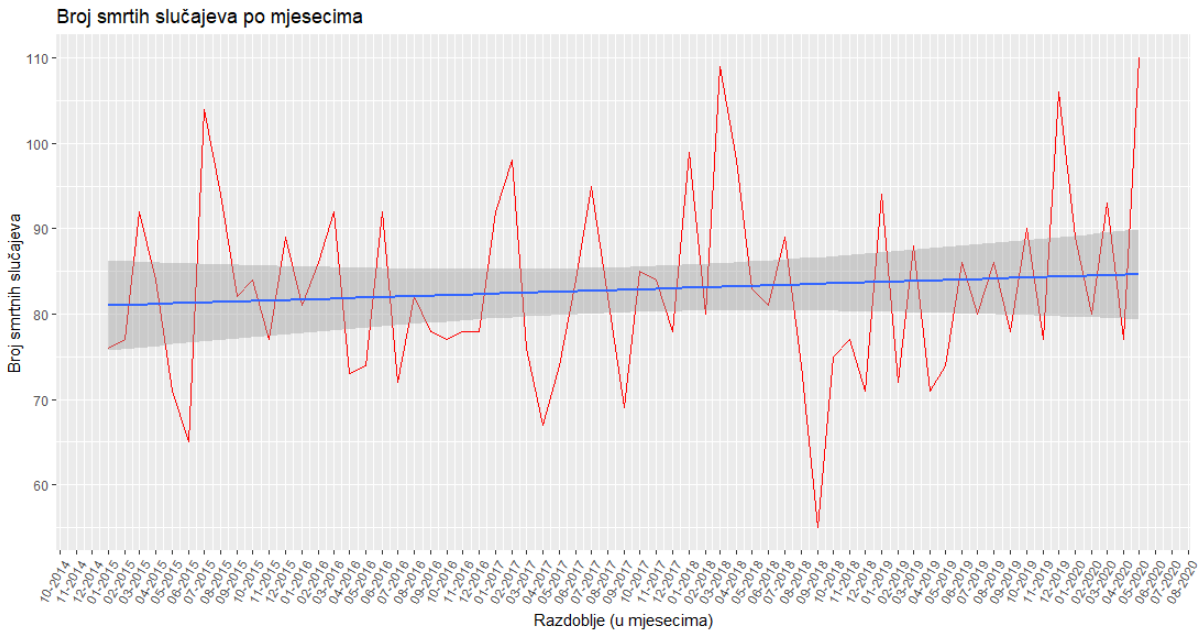
SQL upit:

```
SELECT DATEPART(month, d.date) AS month,DATEPART(YEAR, d.date) AS
year,CONCAT(DATEPART(YEAR, d.date),'-',DATEPART(month, d.date),'-1')
AS concat,COUNT(*) AS count FROM [dbo].[ShootPerson] sp
JOIN [dbo].[Dates] d
ON d.id=sp.fk_date
WHERE d.date < '2020-06-01'
GROUP BY DATEPART(month, d.date), DATEPART(YEAR,d.date)
ORDER BY year,month
```

R kod:

```
timeDB <- InputDataSet
timeDB <- data.frame(
  timeDB$month,
  timeDB$year,
  timeDB$concat,
  timeDB$count
)
names(timeDB)<-c("month","year","concat","count")

p <- ggplot(timeDB, aes(x=as.Date(concat), y=count)) +
  geom_line( color="red") +
  xlab("Razdoblje (u mjesecima)") +
  ylab("Broj smrtnih slučajeva") +
  ggtitle("Broj smrtih slučajeva po mjesecima")+
  geom_smooth(method = "lm")+
  theme(axis.text.x=element_text(angle=60, hjust=1))

p + scale_x_date(date_breaks = "1 month", date_labels = "%m-%Y")
```

Korištenjem ovog jednostavnog R koda i SQL upita dobije se vrlo koristan graf, odnosno informacija da je kroz zadnjih 4.5 godine broj ubijenih osoba od strane policijskih službenika porastao. Ustvari trend je rastući i može se očekivati kako će takav i dalje ostati, barem u bližoj budućnosti. Stvarni podaci su označeni crvenom linijom, a trend je prikazan plavom linijom. Takva informacija je izrazito korisna saveznim državama, odnosno SAD-u, koji na temelju toga može donijeti razne odluke ili mjere pomoću kojih bi se trend trebao ublažiti ili smanjiti čime bi se postiglo sigurno okruženje za stanovništvo i za policijske službenike. Bitno je napomenuti kako sivo područje označava druge moguće putanje trenda, tj. trend linije, a što je manje to područje to je točnija trend linija.



Slika 23. Prikaz trenda smrtnih incidenata izazvanih policijskim službenicima u SAD-u (izvor: vlastita izrada)

Iz grafa se vidi kako je sivo područje relativno veliko, tako da bi se dobiveni podaci trebali pažljivo koristiti. Razlog tomu je mogućnost brže i značajne izmijene novim smrtnim incidentima, odnosno ne incidentima. U takvim slučajevima gornji graf bi mogao imati veću uzlaznu putanju ili pak padajuću ukoliko su podaci takve prirode.

7.4.4. Primjer prediktivne analitike

U prethodnoj cjelini je prikazano određivanje trenda iz kojeg se lako može odrediti buduće stanje i moglo bi se reći kako je takvo nešto jednostavna vrsta *prediktivne analitike*. U ovoj cjelini biti će prikazan konkretan primjer *prediktivne analitike* zasnovan na strojnom učenju, koji uzima značajno više parametara u obzir prilikom davanja određenog rezultata.

Za primjer će se uzeti određivanje mentalnog stanja ubijene osobe, ima li problema ili ne. Također bi se strojno učenje moglo koristiti za određivanje mentalnog stanja osobe koja je u bijegu. Takve informacije su izrazito korisne za ustanovljavanje razloga iz kojih je osoba stupila u okršaj s policijskim službenikom ili ne, pa čak iako osoba nije ubijena. Podaci se mogu koristiti prilikom davanje dijagnoze osobi (u medicini) ili prilikom suđenja kako bi se eventualna kazna umanjila/uvećala. Kao što je navedeno u cjelini 5.2.2, postoji nekoliko tehnika pomoću kojih se može izvesti *prediktivna analitika*, a u ovome slučaju se koristi strojno učenje zasnovano na linearnoj regresiji. Linearna regresija provodi se u R jeziku pomoći specijalizirane funkcije `glm()` (engl. *generalized linear model*). Kao zavisna varijabla uzima se signs_of_*mentall_illness*, a za nezavisne varijable postavlja se *age*, *gender*, *race*, *threat_level*, *city*, *state*, *armed*, *manner_of_death*, *flee*. Za distribuciju podataka tu je *quasibinominal* tip. U nastavku je prikazana SQL procedura sa eksternom R skriptom koja se može preuzeti sa web stranice (str. 57).

```
CREATE PROCEDURE [dbo].[generiranje_modela] (@trained_model varbinary(max)
OUTPUT) AS

DECLARE @inquery nvarchar(max) = N'
SELECT
sp.mental_illness,sc.manner_of_death,sc.armed,sp.age,r.gender,r.race,l.area,l
.city,sc.thread_level,sc.flee
FROM [dbo].[ShootPerson] sp
JOIN [dbo].[ShootingCircumstances] sc ON sc.id=sp.fk_shooting_circ
JOIN [dbo].[Race] r ON sp.fk_race=r.id
JOIN [dbo].[Location] l ON l.id=sp.fk_location'

BEGIN
    EXEC sp_execute_external_script
    @language = N'R'
    , @script = N'
databaseTable <- InputData
```

```
table <- data.frame(
                databaseTable$age,
                databaseTable$manner_of_death,
                databaseTable$armed,
                databaseTable$gender,
                databaseTable$race,
                databaseTable$area,
                databaseTable$mental_illness,
                databaseTable$thread_level,
                databaseTable$flee,
                databaseTable$city)

names(table) <- c("age", "manner", "armed", "gender", "race", "state",
"sings", "thread", "flee", "city")

model <- glm(formula = sings ~ age + manner + armed + gender + city + state +
race + thread + flee, data = table, family = quasibinomial)

trained_model <- as.raw(serialize(model, NULL));'
       , @input_data_1 = @inquery
       , @input_data_1_name = N'InputData'
       , @params = N'@trained_model varbinary(max) OUTPUT'
       , @trained_model = @trained_model OUTPUT;
END;
```

Procedura kao izlazni rezultat daje generirani model koji se onda pohranjuje u zasebnu tablicu (vidi 7.3). Čim je više podataka u „*igri*“ to se može generirati bolji i točniji model. Za takvo nešto je potrebna nova procedura koja okida iznad navedenu proceduru, prima rezultat i pohranjuje model u tablicu. Druga procedura je vrlo jednostavna i prikazana je u nastavku, a preuzeti se može sa web stranice (str. 57).

```
CREATE PROCEDURE [dbo].[insert_model]
AS
BEGIN

DECLARE
       @num INT,
       @model varbinary(MAX);

SELECT @num=COUNT(*) FROM [dbo].[ShootPerson]
PRINT(@num)

EXEC generiranje_modela @model OUTPUT

INSERT INTO [dbo].[Model](model,date_created,number_of_rows_used)
VALUES(@model,GETDATE(),@num)
END
```

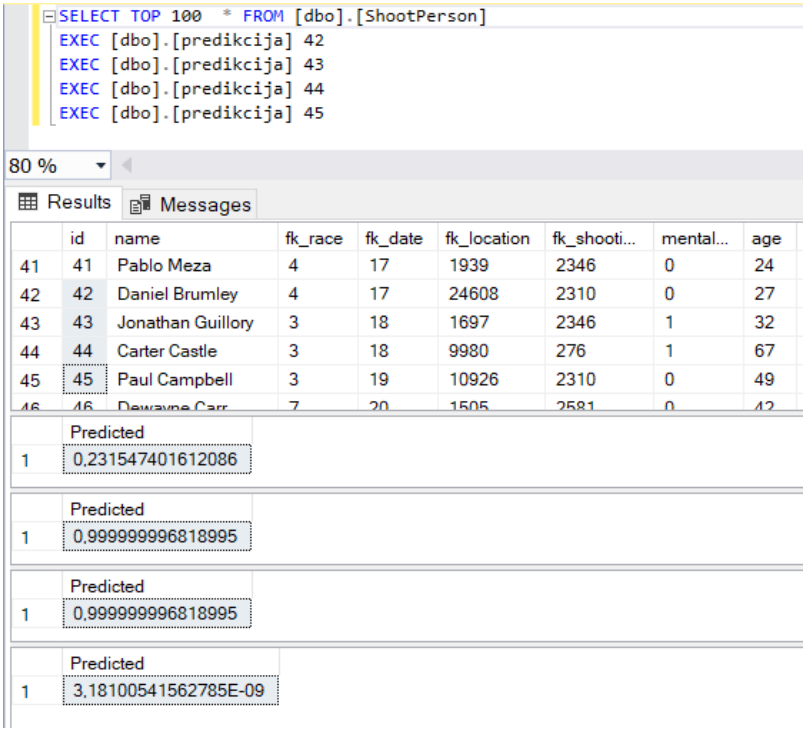
Pokretanje i izvođenje procedure zahtjeva relativno puno resursa (procesorske moći), u ovome slučaju se procedura na oko 5000 zapisa izvodi 3 minute. Može se očekivati kako na sve većim podacima izvođenje, odnosno generiranje modela, traje sve duže. Kada se govori o poslovnoj analitici, to može biti i do nekoliko milijuna zapisa. Sama R skripta se može ubrzati korištenjem isključivo brojčanih podataka, no u ovome slučaju su i ovakvi rezultati performansi prihvatljivi.

U ovome trenutku model je spreman za korištenje te se može vršiti predikcija nad novim podacima. U skladište podataka moguće je unijeti novi/e zapis/e za koje se ne znaju podaci o tome da li osoba ima mentalnih problema ili ne. Kroz novokreiranu proceduru se proslijedi podatak o kojem se zapisu radi, za koji zapis se želi odrediti ima li osoba mentalnih problema ili ne, odnosno kolika je vjerojatnost za to. Budući da je podatak izražen vrijednostima 0 ili 1, gdje 0 označava da osoba nema mentalnih problema, a 1 da ima, kao rezultat predikcije dobit će se decimalna vrijednost u tim granicama. U nastavku je prikazana SQL procedura sa eksternom R skriptom za predikciju s izvornim kodom sa web stranice (str. 57).

```
CREATE PROCEDURE [dbo].[predikcija] (@id INT)
AS
DECLARE @model varbinary(max) = (SELECT TOP 1 model FROM [dbo].[Model] ORDER
BY date_created DESC, number_of_rows_used DESC);
BEGIN
 EXEC sp_execute_external_script
    @language = N'R',
    @script = N'

       databaseTable2 <- PredictData
       table2 <- data.frame(
              databaseTable2$age, databaseTable2$manner_of_death,
              databaseTable2$armed, databaseTable2$gender,
              databaseTable2$race, databaseTable2$area,
              databaseTable2$thread_level, databaseTable2$flee,
              databaseTable2$city )

              names(table2) <- c("age", "manner", "armed", "gender", "race",
              "state", "thread", "flee", "city")

              current_model <- unserialize(as.raw(model));
              predicted <- predict(current_model, table2, type = "response");
              options("scipen" = 100, "digits" = 4)
              print(predicted)
              sings_of_mental_illness <- data.frame(predicted)
              OutputDataSet <- sings_of_mental_illness'

, @input_data_1 = N'SELECT sp.mental_illness, sc.manner_of_death, sc.armed,
sp.age, r.gender, r.race, l.area, l.city, sc.thread_level, sc.flee
       FROM [dbo].[ShootPerson] sp
       JOIN [dbo].[ShootingCircumstances] sc ON sc.id=sp.fk_shooting_circ
       JOIN [dbo].[Race] r ON sp.fk_race=r.id
       JOIN [dbo].[Location] l ON l.id=sp.fk_location
       WHERE sp.id=@ID'
, @input_data_1_name=N'PredictData'
, @params = N'@ID INT,@model VARBINARY(max)'
, @ID=@id
, @model = @model
WITH RESULT SETS ((Predicted FLOAT));
END
```

Za ulazne parametre u skriptu se proslijedi aktualni generirani model koji se prethodno dohvatio putem SQL upita i identifikator zapisa za koji se želi izvršiti predikcija. Kao ulazni set podataka se proslijedi SQL upit koji prihvaća parametar identifikatora pomoću kojeg se dobiju podaci o zapisu za koji se vrši predikcija.

Proces predikcije vrši R funkcija zvana `predict()` koja za parametre prihvaća podatke zapisa za koji se vrši predikcija (nezavisne varijable), model na temelju kojeg se vrši predikcija te kakvog je tipa predikcija. U nastavku je prikazano izvođenje predikcije na 4 zapisa, gdje su dva zapisa označena nulom, a druga dva jedinicom, odnosno dvije osobe nemaju naznake mentalnih problema, dok druge dvije imaju.



Slika 24. Rezultati predikcije (izvor: vlastita izrada)

Na slici je vidljivo kako R skripta u sva četiri slučajeva daje točan odgovor. Kod prvog i zadnjeg zapisa tj. osobe, nije dijagnosticirano da ima mentalnih problema, a to i predikcija potvrđuje jer su oba zapisa ispod 0.5 (50%). Druga dva zapisa su gotovo na 1 (100%), što se poklapa sa stvarnim podacima.

Iz gornjeg primjera se može zaključiti kako takva funkcionalnost može biti vrlo moćna ako se izvede na ispravan način, no ne smije se isključivo osloniti na takve informacije jer i dalje postoji određen postotak pogrešnih zaključivanja. Ovakav primjer može biti od velike pomoći policijskim službenicima da bolje procijene kritične situacije, može pomoći liječnicima da donesu točniju odluku o stanju osobe te pravosuđu da donesu ispravnu presudu.

8. Zaključak

Cilj rada je bio prikazati razlike između poslovne analitike i poslovne inteligencije, kako se provodi poslovna analitika te što ona može pružati svom korisniku. Kroz rad su opisane razne vrste poslovnih analitika i čemu koja služi te kakve odgovore daju. Također je bio cilj prikazati tehnički aspekt poslovne analitike putem MS SQL Server 2019 i kako se on može koristiti za dobivanje ključnih informacija za donošenje odluka.

Na početku rada je opisana poslovna inteligencija koja je neophodna za shvaćanje poslovne analitike, budući da je ona sadržana unutar nje. U cjelini Deskriptivna analitika najbolje se može uočiti kako je zapravo poslovna analitika nadogradnja poslovne inteligencije. Poslovna inteligencija daje odgovore na trenutno stanje onoga što promatramo, a takvo nešto omogućuje deskriptivna analitika kod poslovne analitike. Sve ostalo je ustvari nadopuna, prediktivna, dijagnostička i preskriptivna analitika koji daju još detaljnije informacije, odnosno odgovore.

U radu je također bilo riječi o razlikama između skladišta podataka i Big Data te koju svrhu imaju za poslovnu analitiku. Da bi se mogla provesti analiza podataka potrebni su nekakvi izvori podataka, a to pružaju i skladište podataka i Big Data. Skladište podataka omogućuje pristup strukturiranim i organiziranim podacima, dok Big Data omogućuje pristup većoj količini podataka koji u većini slučajeva nisu strukturirani. Oba pristupa imaju svoje prednosti i nedostatke, razlika je u tome da je skladište podataka jeftinije od Big Data jer se koriste relacijske baze podataka u odnosu na napredne okvire i algoritme. Također su zbog toga podaci pristupačniji kod skladišta podataka. Na drugoj strani Big Data omogućuje pristup značajno većem spektru podataka, nekad iz neočekivanih izvora što ima svoje koristi.

Svrha rada je također bila opisati kako se poslovna analitika provodu uz pomoć MS SQL Server 2019. Naime navedeni SUBP ima mogućnost korištenja R i Python skripti u sklopu SQL-a koji se koriste za analizu podataka. Za omogućavanje takvih funkcionalisti treba izdvojiti dosta vremena i strpljenja budući da sustavu treba dati veliki broj prava. Također je dosta teško koristiti R jezik unutar SQL ukoliko je potrebno koristiti biblioteke. Potrebno ih je ručno kroz konzolu uvesti, a to može potrajati. Kada se sve uspješno postavi, tada se dobiva vrlo moćan alat koji omogućuje automatizirano generiranje izvještaja putem procedura. Također je omogućena pohrana predikcijskog modela kao i vršenje predikcije s jednog mjesta, MS SQL Server-a 2019.

Kao zadnje, svrha rada je bila prikazati jedan praktični primjer poslovne analitike (deskriptivne i prediktivne analitike) uz pomoć MS SQL Servera. Analizom podataka se dobije potpuni uvid u podatke koji mogu pomoći prilikom donošenja zaključka/odluka. U ovome slučaju dobiveni rezultati se odnose na nedavne događaje u SAD-u što se tiče nasilja policijskih službenika nad građanstvom. Dobiveni podaci mogu poslužiti za edukaciju stanovništva kao i onih koji vode državu. Na temelju podataka se mogu donijeti razne zakonske odredbe koje bi mogle spriječiti takvo nasilje, ali i smanjiti kriminalitet što je također jedna stavka koja omogućuje poslovna analitika.

Popis literature

1. Božidar Javorović, M. B. (2007). *Poslovne informacije i business intelligence.* Zagreb: Golden marketing-Tehnička knjiga.
2. *Comp.nus.edu.* (n.d.). Preuzeto 10. 03 2020 iz https://www.comp.nus.edu.sg/~lingtw/cs4221/dw.pdf
3. DeWitt, L. (25. 03. 2020.). *ssa.gov*. Dohvaćeno iz The Development of Social Security in America: https://www.ssa.gov/policy/docs/ssb/v70n3/v70n3p1.html
4. *eriqande.github.io*. (24.. 08. 2020). Dohvaćeno iz Making Maps with R: https://eriqande.github.io/rep-res-web/lectures/making-maps-with-R.html
5. Estopace, E. (22. 05. 2019.). *futureiot.tech*. Dohvaćeno iz Big Data and Analytics: https://futureiot.tech/idc-forecasts-connected-iot-devices-to-generate-79-4zb-of-data-in-2025/
6. *fer.unizg.hr*. (27. 03. 2020.). Dohvaćeno iz Strojno učenje: https://www.fer.unizg.hr/predmet/su
7. Gartner. (20. 03. 2020.). *gartner.com*. Dohvaćeno iz gartner glossary: https://www.gartner.com/en/information-technology/glossary/big-data
8. Inmon, W. H. (1992). *Building the Data Warehouse.* New York: John Wiley & Sons, Inc. Preuzeto 10. 03 2020 iz https://www.comp.nus.edu.sg/~lingtw/cs4221/dw.pdf
9. Jasminka Dobša, K. K.-G. (2008). *Statistika deskriptivna i inferencijalna i vjerojatnost.* Varaždin: Tiskara Varteks d.o.o.
10. Marr, B. (29. 03. 2020.). *bernardmarr*. Dohvaćeno iz What is Hadoop: https://www.bernardmarr.com/default.asp?contentID=1080
11. *microstrategy*. (03. 04. 2020.). Dohvaćeno iz Business Analytics: Everything You Need to Know: https://www.microstrategy.com/us/resources/introductory-guides/business-analytics-everything-you-need-to-know
12. Mladen Varga, I. S. (2016). *Informacijski sustavi u poslovanju.* Zagreb: Sveučilište u Zagrebu.
13. *Oracle*. (25. 03. 2020.). Dohvaćeno iz What Is Big Data?: https://www.oracle.com/big-data/guide/what-is-big-data.html
14. Rabuzin, K. (2011). *Uvod u SQL.* Varaždin: Fakultet organizacije i informatike.
15. Rabuzin, K. (2014). *SQL - Napredne teme.* Varaždin: Fakultet organizacije i informatike.
16. Rabuzin, K. (03. 08. 2020.). *elfarchive1819.foi.hr.* Dohvaćeno iz SPPI: Predavanje 2: https://elfarchive1819.foi.hr/mod/resource/view.php?id=9651
17. Rabuzin, K. (21. 03. 2020.). *elfarchive1819.foi.hr.* Dohvaćeno iz SPPI: Predavanje 4: https://elfarchive1819.foi.hr/mod/resource/view.php?id=9659
18. Rouse, M. (06. 2019.). *searchbusinessanalytics.techtarget.com*. Dohvaćeno iz business analytics (BA): https://searchbusinessanalytics.techtarget.com/definition/business-analytics-BA
19. Stepinac, L. (20. 03. 2020.). *ictbusiness.info*. Dohvaćeno iz ICT Business : https://www.ictbusiness.info/poslovna-rjesenja/sto-je-to-zapravo-big-data-i-gdje-se-primjenjuje
20. *Tableau*. (04. 08. 2020.). Dohvaćeno iz Business Intelligence or Business Analytics: https://www.tableau.com/learn/articles/business-intelligence/bi-business-analytics
21. White, T. (2012). *Hadoop: The Definitive Guide.* Sebastopol: O'Reilly Media, Inc.
22. Winston, A. (2015). *Bussiness Analytics, Data Analysis and Decision Making.* Cenage Learning.
23. Zhang, A. (2017). *Data Analytics.* CreateSpace Independent Publishing Platform.

Popis slika

Prilozi

1. Izvorne datoteke:
   https://github.com/fkraljic/Poslovna_analitika_u_sustavu_MS_SQL_Server_2019