Izrada SQL upita za analizu bioloških podataka

Plec, Andrea

SVEUČILIŠTE U ZAGREBU
PREHRAMBENO-BIOTEHNOLOŠKI FAKULTET

DIPLOMSKI RAD

Andrea Plec
1024/MB

Zagreb, prosinac 2019.

IZRADA SQL UPITA ZA ANALIZU BIOLOŠKIH PODATAKA

Rad je izrađen u Laboratoriju za bioinformatiku na Zavodu za biokemijsko inženjerstvo Prehrambeno-biotehnološkog fakulteta Sveučilišta u Zagrebu pod mentorstvom dr. sc. Janka Diminića, docenta Prehrambeno-biotehnološkog fakulteta Sveučilišta u Zagrebu.

Z a h v a l a

Prvenstveno se želim zahvaliti svojem mentoru, doc. dr. sc. Janku Diminiću, na pruženoj mogućnosti izrade ovog diplomskog rada i neprocjenjivoj motivaciji, kao i na svim korisnim savjetima i pruženoj pomoći pri pisanju. Hvala Vam jer ste uvijek bili dostupni i spremni pomoći, čak i za najmanje sitnice.

Hvala kolegici Katarini na kasnonoćnim raspravama, ohrabrivanju i bezuvjetnoj pomoći u borbi s programiranjem i nemogućim greškama.

Od srca hvala svim mojim prijateljima na pomoći, razgovorima i poticajima u trenutcima kad bi najradije bacila laptop kroz prozor. Spasili ste mi živce, a nebrojeno puta i laptop.

I za kraj, najveće hvala ide mojoj obitelji. Hvala mom bratu srećkoviću što je uvijek dijelio dio svoje sreće sa mnom – dobro mi je došla. Posebno hvala roditeljima na razumijevanju i potpori tijekom studiranja, a najviše na beskonačnom strpljenju. Ohrabrivali ste me pri svakom usponu, i pomagali mi ustati svaki put kad sam pala. Bez vas ništa od ovoga ne bi bilo moguće.

TEMELJNA DOKUMENTACIJSKA KARTICA

Diplomski rad

Sveučilište u Zagrebu
Prehrambeno-biotehnološki fakultet
Zavod za biokemijsko inženjerstvo
Laboratorij za bioinformatiku

Znanstveno područje: Biotehničke znanosti
Znanstveno polje: Biotehnologija

IZRADA SQL UPITA ZA ANALIZU BIOLOŠKIH PODATAKA

Andrea Plec, 1024/MB

Sažetak: Zahvaljujući mnogim istraživanjima, crijevna mikrobiota se pokazala kao važna komponenta fiziologije čovjeka s kojim živi u mutualističkom odnosu. S obzirom da je riječ o kompleksnoj zajednici mikroorganizama na čiji sastav utječu mnogobrojni unutarnji i vanjski čimbenici, svaki pojedinac posjeduje jedinstveni sastav crijevne mikrobiote. Disbioza crijevne mikrobiote može dovesti do raznih upalnih i metaboličkih poremećaja, što može rezultirati bolestima. Utvrđivanje sličnosti i razlika među zdravim i bolesnim skupinama pojedinaca pruža mogućnost određivanja povezanosti sastava crijevne mikrobiote s tim poremećajima i bolestima. U tu svrhu se provodi sekvenciranje mnogobrojnih uzoraka crijevne mikrobiote, a dobivena očitanja se zatim bioinformatički obrađuju te se rezultati pohranjuju u baze podataka. U ovom radu je u sklopu računalnog paketa MySQL korištena baza podataka koja sadrži brojnosti pojedinih taksona za 1639 uzoraka crijevne mikrobiote. Korištenjem računalnog jezika SQL izrađeni su pohranjeni postupci za analizu raznolikosti sastava pojedinih uzoraka i procjenu prosječnog sastava za sve uzorke.

Ključne riječi: bioinformatika, baza podataka, SQL, crijevna mikrobiota, bioraznolikost

Rad sadrži: 60 stranica, 14 slika, 4 tablice, 87 literaturnih navoda, 4 priloga

Jezik izvornika: hrvatski

Rad je u tiskanom i elektroničkom (pdf format) **obliku pohranjen u:** Knjižnica Prehrambeno-biotehnološkog fakulteta, Kačićeva 23, Zagreb

Mentor: *doc. dr. sc. Janko Diminić*

Stručno povjerenstvo za ocjenu i obranu:

1. Izv. prof. dr. sc. Jurica Žučko
2. Doc. dr. sc. Janko Diminić
3. Doc. dr. sc. Andreja Leboš Pavunc
4. Prof. dr. sc. Jasna Novak (zamjena)

Datum obrane: 12. prosinca 2019.

BASIC DOCUMENTATION CARD

Graduate Thesis

University of Zagreb
Faculty of Food Technology and Biotechnology
Department of Biochemical Engineering
Laboratory for Bioinformatics

Scientific area: Biotechnical Sciences
Scientific field: Biotechnology

BUILDING SQL QUERIES FOR BIOLOGICAL DATA ANALYSIS

Andrea Plec, 1024/MB

Abstract: According to numerous studies, gut microbiota lives in a mutualistically beneficial relationship with humans and has proven to be an important component of their physiology. Each individual has a unique gut microbiota composition, given that it's a complex community of microorganisms whose composition is affected by many internal and external factors. Dysbiosis of the gut microbiota can lead to various inflammatory and metabolic disorders, which can result in illness. Determining similarities and differences among healthy and affected cohorts provides an opportunity to establish correlations between gut microbiota composition and various afflictions. Such research requires sequencing of numerous intestinal microbiota samples. The acquired sequence reads are then bioinformatically processed and the results are stored in databases. In order to build several stored procedures using SQL, a database containing taxa abundance of 1639 gut microbiota samples was used. The stored procedures were built within MySQL database management system, and used to analyze the composition diversity of individual samples and to estimate the average composition for all samples.

Keywords: bioinformatics, database, SQL, gut microbiota, biodiversity

Thesis contains: 60 pages, 14 figures, 4 tables, 87 references, 4 supplements

Original in: Croatian

Graduate thesis in printed and electronic (pdf format) **version is deposited in:** Library of the Faculty of Food Technology and Biotechnology, Kačićeva 23, Zagreb

Mentor: *PhD, Janko Diminić, Assistant professor*

Reviewers:

1. PhD, Jurica Žučko, Associate professor
2. PhD, Janko Diminić, Assistant professor
3. PhD, Andreja Leboš Pavunc, Assistant professor
4. PhD, Jasna Novak, Associate professor (substitute)

Thesis defended: 12th December 2019

SADRŽAJ

1. UVOD

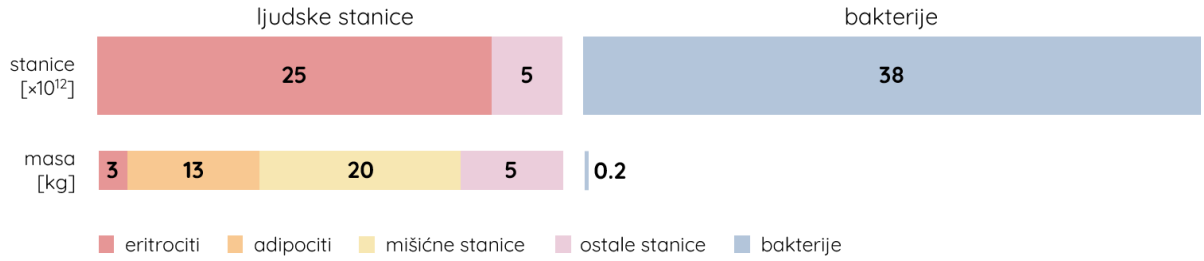
Koevolucijski simbiotski odnos između mikroorganizama i višestaničnih organizama istaknuta je karakteristika života na Zemlji, a upravo ti mikroorganizmi čine nenadmašnu komponentu bioraznolikosti svih živih organizama. Stoga nije iznenađujuće da su ljudi, kao i ostali sisavci, kolonizirani ogromnom, kompleksnom, i nadasve dinamičnom zajednicom mikroorganizama. Pri tome ih se najviše nalazi u gastrointestinalnom traktu, posebice u crijevima. Svaki pojedinac posjeduje svoj vlastiti i jedinstveni sastav crijevne mikrobiote, poput otiska prsta. Varijabilnost je najizraženija na razini vrste i soja, dok na višim razinama taksonomske klasifikacije postoje određene sličnosti među pojedincima. U tom smislu može se govoriti o temeljnoj zajednici mikrobiote, najčešće na razini koljena ili roda. Unatoč mutualističkom suživotu s crijevnom mikrobiotom, koja generalno za domaćina nosi koristi poput zaštite od patogena i olakšanog dobivanja energije iz hrane, taj odnos ponekad može postati patološki, te uzrokovati upalne i metaboličke poremećaje (Rosenberg, 2017; Lynch i Pedersen, 2016; Bäckhed, 2005; Xu i Gordon, 2003; Hooper i sur., 2002). Prema tome, uspostavljanje i održavanje poželjnih interakcija između domaćina i njegove crijevne mikrobiote ključni su uvjeti za održavanje zdravlja domaćina. Istraživanjima se pokušavaju utvrditi razlike u sastavu crijevne mikrobiote među zdravim i bolesnim skupinama pojedinaca, što se pokazalo izuzetno zahtjevnim s obzirom da na sastav utječe mnogo unutarnjih i vanjskih čimbenika, počevši od genetike domaćina do njegovog stila života (Cani, 2018; Sommer i Bäckhed, 2013; Xu i Gordon, 2003).

Mogućnosti proučavanja raznolikosti crijevne mikrobiote uvelike su se poboljšala uvođenjem visokopropusnih metoda sekvenciranja koje omogućuju lako razlikovanje vrsta. Međutim, sekvenciranjem se dobiva ogroman broj podataka koje je potrebno obraditi kako bi ih se moglo analizirati u svrhu nekog istraživanja. Trenutno se istraživanja crijevne mikrobiote sastoje od dva glavna koraka: (i) sekvenciranja mikrobne DNA i (ii) bioinformatičke obrade i analize dobivenih podataka. Bioinformatički projekti često uključuju rad s mnogo tabličnih podataka, najčešće organiziranih u bazu podataka. Za njihovu obradu mogu se koristiti razni sustavi za upravljanje bazama podataka, među kojima je MySQL jedan od najpopularnijih. Računalni jezik SQL pruža sposobnost pregledavanja, manipuliranja, filtriranja i sažimanja tih podataka. Još bitnije, lako ga je uključiti u bilo koji niz programa za analizu podataka (Thursby i Juge, 2017; Jandhyala i sur., 2015; Bessant i sur., 2014). Glavni cilj ovog rada je izrada upita pomoću SQL računalnog jezika, kako bi se u sklopu sustava MySQL omogućile različite analize sastava i raznolikosti crijevne mikrobiote pojedinaca, iz podataka dobivenih metagenomičkim *shotgun* sekvenciranjem.

2. TEORIJSKI DIO

2.1. MIKROBIOTA

Mikroorganizmi koloniziraju sve površine višestaničnih organizama koje su izložene okolišu. Stoga izraz ‘mikrobiota’ obuhvaća sve mikroorganizme, uključujući bakterije, arheje, funge, protozoe i viruse, koji žive u i/ili na domaćinu, ili određenom dijelu organizma domaćina (Sommer i Bäckhed, 2013; Clemente i sur., 2012). U tom smislu može se govoriti o ljudskom holobiontu, odnosno superorganizmu sastavljenom od čovjeka i stanica mikroorganizama. Štoviše, u i na čovjeku živi čak oko $3{,}8\times10^{13}$ bakterija (Slika 1), što je otprilike jednako ukupnom broju ljudskih stanica u tijelu (Lynch i Pedersen, 2016; Sender i sur., 2016). S obzirom da se vrste mikroorganizama i broj stanica pojedinih vrsta razlikuju od domaćina do domaćina, mikrobiota predstavlja kompleksnu i raznoliku zajednicu mikroorganizama s određenim genetičkim kapacitetom. Stoga je kolektivni genomički sadržaj mikrobiote nazvan ‘mikrobiom’ (Lloyd-Price i sur., 2016; Clemente i sur., 2012; Tremaroli i Bäckhed, 2012).



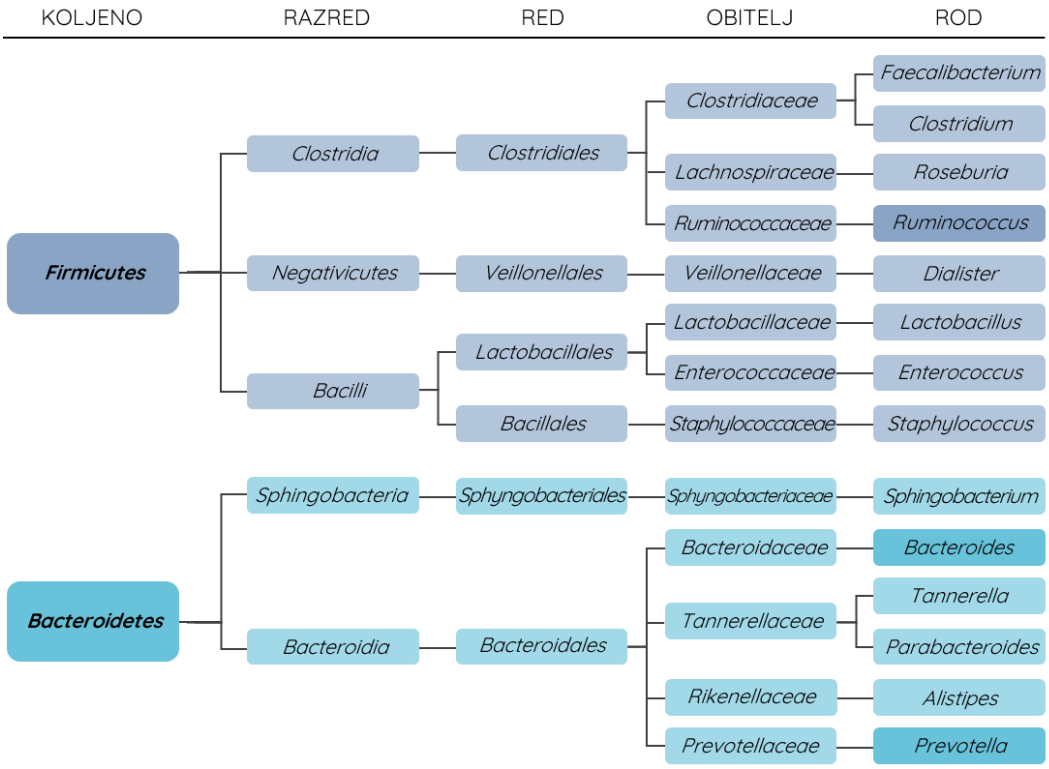
Slika 1. Usporedba broja i mase ljudskih stanica i bakterija u ljudskom tijelu na primjeru odraslog muškarca mase 70 kg. Zanimljivo je da sve bakterijske stanice u tijelu zajedno teže približno 0,2 kg, što je 0,29 % ukupne tjelesne mase (prilagođeno prema Sender i sur., 2016).

Kolonizacija ljudskog organizma započinje već tijekom rođenja kao posljedica izlaganja okolišu, a većina mikroorganizama koji ga koloniziraju nije patogena za imunokompetentne domaćine (Sommer i Bäckhed, 2013; Smith i sur., 2007). Pri tome ljudski gastrointestinalni trakt predstavlja jedno od najvećih mjesta kontakta okolišnih čimbenika s ljudskim antigenima, i to površine od 250 do čak 400 m^2 (Bengmark, 1998). Kao dio gastrointestinalnog trakta, crijeva su posebno poželjna niša za kolonizaciju, s obzirom da su puna izvora ugljika, minerala, i drugih otopljenih supstanci, te se uz to održavaju na stabilnoj temperaturi (Smith i sur., 2007). Zajednica bakterija, arheja i eukariota koja kolonizira gastrointestinalni trakt nazvana je ‘crijevnom mikrobiotom’ (eng. *gut microbiota*), i koevoluirala je s domaćinom kao posljedica raznih promjena tijekom tisuća godina (Bäckhed, 2005).

2.1.1. Temeljni crijevni mikrobiom i enterotipovi

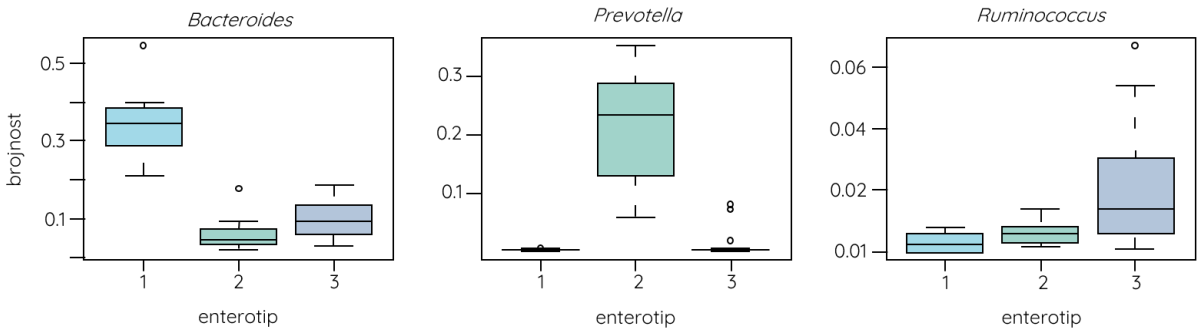
U gastrointestinalnom traktu postoje dva gradijenta mikrobne raspodjele: mikrobna gustoća i bakterijska raznolikost. S obzirom na građu gastrointestinalnog trakta, mikrobna gustoća povećava se od želuca prema debelom crijevu, a proporcionalno raste i bakterijska raznolikost (Sommer i Bäckhed, 2013). Primjerice, u jednjaku i stomaku ima 10 bakterija po gramu sastava, dok ih u debelom crijevu ima čak 10^{12} po gramu sastava (O'hara i Shanahan, 2006). Mnoge bakterijske vrste prisutne su u lumenu, dok neke imaju sposobnost adhezije na mukozni sloj (Sommer i Bäckhed, 2013). Crijevna mikrobiota prvo prolazi kroz proces sazrijevanja od rođenja do odrasle dobi, nakon čega se dodatno mijenja tijekom čitavog života domaćina, budući da promjene u prehrambenim navikama, higijeni, korištenju antibiotika, i općenito stilu života domaćina zapravo čine crijeva vrlo dinamičnim staništem koje je podložno brzim promjenama fizioloških parametara. Osim što takve promjene utječu na sastav mikrobne zajednice ili individualnih mikrobnih genoma, one u konačnici dovode do modifikacije mikrobioma, a samim time i transkriptoma, proteoma i metaboloma domaćina (Rosenberg, 2017; Sommer i Bäckhed, 2013; Yatsunenko i sur., 2012).

Bakterijska komponenta mikrobiote postala je predmet velikih i opširnih istraživanja tijekom posljednjih 15-ak godina, uključujući *The Human Microbiome Project* (Human Microbiome Project Consortium, 2012; Peterson i sur., 2009; Turnbaugh i sur., 2007) i *Metagenome of the Human Intestinal Tract (MetaHIT)* (Qin i sur., 2010). Uvođenjem novih, jeftinijih i visokopropusnih metoda sekvenciranja koje omogućuju lako razlikovanje vrsta uvelike se poboljšala mogućnost proučavanja raznolikosti crijevne mikrobiote (Thursby i Juge, 2017), pa je stoga uslijedilo još velikih projekata fokusiranih na crijevnu mikrobiotu, poput *Flemish Gut Flora Project*-a (Falony i sur., 2016) i *LifeLines-DEEP* projekta (Zhernakova i sur., 2016). Iako je utvrđeno da je crijevna mikrobiota među pojedincima jedinstvena i drugačija, ta različitost je izražena zahvaljujući varijabilnosti na nižim razinama taksonomske klasifikacije, posebice na razini vrste i soja, dok varijabilnost na višim taksonomskim razinama nije specifična za pojedince (Human Microbiome Project Consortium, 2012; Jeffrey i sur., 2012; Yatsunenko i sur., 2012). S obzirom da više od 90% bakterija u crijevima pripada u isključivo dva koljena, *Bacteroidetes* i *Firmicutes* (Slika 2), primijećen je kontinuirani gradijent unutar ljudske populacije prema kojem kod nekih pojedinaca prevladava jedno, a kod nekih drugo koljeno (Human Microbiome Project Consortium, 2012; Claesson i sur, 2011).



Slika 2. Primjer rodova koji se često nalaze u sastavu crijevne mikrobiote (Prilagođeno prema Rinninella i sur., 2019).

Arumugam i sur. su u sklopu MetaHIT projekta ukazali na postojanje tri izražena sastava mikrobiote ovisno o rodu bakterija koji prevladava unutar zajednice (Slika 3). Nazvali su ih 'enterotipovima': enterotip 1 je onaj gdje prevladava rod *Bacteroides*, kod enterotipa 2 prevladava *Prevotella*, a kod enterotipa 3 *Ruminococcus* (Arumugam i sur., 2011). Ti enterotipovi povezani su s dugoročnim načinom prehrane, pri čemu je *Bacteroides* vezan uz prehranu bogatu proteinima i životinjskim mastima, a *Prevotella* uz prehranu bogatu ugljikohidratima (Wu i sur., 2011).



Slika 3. Kutijasti dijagrami brojnosti dominantnih rodova crijevne mikrobiote, među kojima svaki doprinosi izraženosti odgovarajućeg enterotipa (Prilagođeno prema Arumugam i sur., 2011).

2.1.2. Funkcije zdravog mikrobioma

Mikrobiom nosi ogromnu važnost za ljudski organizam, do te mjere da ga možemo smatrati našim sekundarnim genomom. Posjeduje preko 400 puta više različitih gena u odnosu na ljudski genom, što bi značilo da bakterije nose više od 99 % genetičkih informacija u ljudima (Rosenberg, 2017; Grice i Segre, 2012). Mikrobiom sadrži više od 5 milijuna gena, od kojih mnogi nose informacije za biosintetske enzime, proteaze i glikozidaze, što značajno proširuje biokemijsku i metaboličku sposobnost domaćina (Sommer i Bäckhed, 2013). Stoga je odnos između čovjeka i crijevne mikrobiote simbiotički, odnosno mutualistički. Gastrointestinalni trakt pruža mikrobioti povoljne uvjete za rast i razmnožavanje, dok je ona esencijalna za normalnu fiziologiju čovjeka, te utječe na širok raspon procesa i karakteristika, često kroz modulaciju imunosnog sustava (Honda i Littman, 2016; Smith i sur., 2007).

Geni mikrobiote koji kodiraju za specifične funkcije slični su među pojedincima, što dodatno potvrđuje postojanje osnovne funkcionalne jezgre mikrobioma, odnosno temeljnog seta gena za koji mikrobiota kodira (Turnbaugh i sur., 2009). Predloženo objašnjenje zdravog mikrobioma je da je to funkcionalna jezgra koja na određenim staništima u sklopu domaćina provodi komplementarne metaboličke i druge molekularne funkcije. Pri tome je zanimljivo da brojnost molekularnih funkcija ne mora nužno odgovarati brojnosti vrsta unutar određenih enterotipova. Uz to, za iste funkcije mogu biti zaslužni potpuno različiti mikroorganizmi kod različitih pojedinaca (Jandhyala i sur., 2015; Shafquat i sur., 2014).

Stoga je zdrav mikrobiom zapravo idealna kolekcija gena i biokemijskih puteva, a ne specifičnih populacija. Među ostalim, 'zdrava' crijevna mikrobiota pruža domaćinu komplementarne genetičke resurse, poput biokemijskih puteva dobivanja energije i biosinteze esencijalnih vitamina, omogućuje razvoj i diferencijaciju crijevnog epitela, pruža zaštitu od invazije patogena, provodi metabolizam ksenobiotika, te ima ključnu ulogu u održavanju homeostaze tkiva (Lynch i Pedersen, 2016; Shin i sur., 2015; Nicholson i sur., 2012; Smith i sur., 2007). Jedinstven set tolerancijskih imunoregulacijskih mehanizama sprječava nepotrebnu aktivaciju imunosnog sustava protiv neškodljivih antigena, uključujući one koje eksprimiraju članovi zajednice mikrobiote, čime se ograničavaju upalni procesi, te je omogućena 'suradnja' mikrobiote s imunosnim sustavom (Johansson i sur., 2011; Rivas i sur., 2011; Vaishnava i sur., 2011; Macpherson i sur., 2009).

2.1.3. Mikrobna disbioza

Kao što već navedeno, zdrava mikrobiota čovjeka sastoji se od dva glavna koljena, *Bacteroidetes* i *Firmicutes*, a uz njih su uglavnom među brojnijima *Actinobacteria, Cyanobacteria, Fusobacteria, Proteobacteria* i *Verrucomicrobia* (Jandhyala i sur., 2015; Human Microbiome Project Consortium, 2012; Qin i sur., 2010). Iako je mikrobiota generalno stabilna za svakog pojedinca tijekom vremena, sastav joj se ipak može mijenjati ovisno o utjecaju određenih vanjskih i unutarnjih čimbenika. Opsežna analiza 1135 uzoraka u sklopu projekta LifeLinesDEEP pokazala je povezanost između sastava mikrobiote i čak 126 vanjskih i unutarnjih čimbenika, uključujući genetiku domaćina, prehranu i nutritivni status, infekcije i medicinske intervencije, te korištenje antibiotika i drugih lijekova. S ciljem poboljšanja zdravstvenog stanja povezanog s mikrobiomom, potencijalno se može manipulirati tim čimbenicima kako bi se utjecalo na promjene taksonomskog i funkcijskog sastava mikrobiote (Zhernakova i sur., 2016; Shin i sur., 2015; Petersen i Round, 2014; Clemente i sur., 2012).

Kao što je već spomenuto, imunosne stanice crijeva su hipoosjetljive na stanice mikrobiote ili čak mutualistički odgovaraju na mikrobnu stimulaciju (Geuking i sur., 2011). Neprikladni imunosni odgovor stoga lako uništava crijevnu homeostazu, izaziva disbiozu, i doprinosi lokalnim i sistemskim upalama te metaboličkim poremećajima (Shin i sur., 2015). Općenito, disbioza podrazumijeva bilo kakvu promjenu u sastavu zajednica mikrobiote domaćina u usporedbi sa zajednicama zdravih pojedinaca, a jedna od glavnih karakteristika disbioze je smanjena raznolikost članova zajednice mikrobiote (Bäumler i Sperandio, 2016; Petersen i Round, 2014). Pretpostavlja se kako promjene u sastavu mikrobnih zajednica doprinose inicijaciji i/ili perzistenciji mnogih bolesti, pa je tako disbioza u mikrobiomu povezana s mnogo njih, uključujući upalnu bolest crijeva, multiplu sklerozu, dijabetes tipa 1 i 2, rak, alergije, astmu, celijakiju, anoreksiju, pretilost, autoimune bolesti, autizam, te čak i neke psihičke poremećaje (Garret, 2015; Petersen i Round, 2014; Hsiao i sur., 2013; Bäckhed i sur., 2012; Clemente i sur., 2012).

2.2. SEKVENCIRANJE MIKROBIOMA

Jedan od najvažnijih koraka u analizi crijevne mikrobiote je taksonomska identifikacija svih članova pojedinih zajednica. Postoji nekoliko metoda koje se koriste za taksonomsku klasifikaciju mikroorganizama, pa tako i organizama članova mikrobiote, bez ikakve potrebe za kultiviranjem tih mikroorganizama. Mikrobiota se na razini DNA obično proučava pomoću 16S rDNA sekvenciranja i *shotgun* (hrv. sačmarica) sekvenciranja, ili pak na razini RNA, pri čemu se konkretna metoda odabire ovisno o potrebama samog istraživanja (Singh i sur., 2017; D'Argenio, 2018). Sveobuhvatniju analizu pruža multiomički pristup koji, kao što sam naziv kaže, obuhvaća nekolicinu takozvanih 'omika': metagenomiku, metatranskriptomiku, metaproteomiku i metametabolomiku (Singh i sur., 2017; Jandhyala i sur., 2015; Morgan i Huttenhower, 2014). Zahvaljujući razvoju biomedicinske tehnologije, za sekvenciranje se koriste visokopropusne platforme sekvenciranja sljedeće generacije (eng. *next generation sequencing, NGS*) (Eurofin Genomics, 2019; Jandhyala i sur., 2015).

Često se sekvenciranje bakterijske DNA temelji na umnožavanju 16S rDNA direktno iz izolirane DNA pomoću specifičnih univerzalnih početnica za PCR, pri čemu se produkti odmah sekvenciraju putem neke od NGS platformi. Regija je duljine 1,5 Kb, a uz visoko konzervirane regije sadrži i 9 hipervarijabilnih regija (V1-V9) koje sadrže sekvence specifične za pojedine vrste, pa stoga služe za taksonomsku identifikaciju (Eurofins Genomics, 2019; Singh i sur., 2017; Thurby i Juge, 2017; Peterson i sur., 2008). Većina NGS platformi daje očitanja koja ne pokrivaju sve konzervirane i varijabilne regije. Zato se obično sekvenciraju samo određene regije potrebne za određivanje mikrobne raznolikosti. Najčešće regije za identifikaciju su V3, V4, V6 i V8, a s obzirom da se u stanici nalazi do 15 kopija gena za 16S rRNA, mogu se umnožavati direktno iz degradiranog uzorka (Singh i sur., 2017; Hamady i sur., 2008).

Za metagenomičko *shotgun* sekvenciranje prvo se ukupna DNA uzorka pocijepa na manje fragmente, koji se zatim direktno sekvenciraju putem neke od NGS platformi. Najčešće se u tu svrhu koristi Illumina HiSeq ili MiSeq. Za razliku od sekvenciranja par odabranih marker lokusa, taksonomska identifikacija se u ovom slučaju temelji na stotinama lokusa, pa se dobivaju informacije o najnižim razinama taksonomske klasifikacije – vrsti i soju (Quince i sur., 2017; Singh i sur., 2017; Segata i sur., 2011).

2.3. BIOINFORMATIČKA ANALIZA MIKROBIOMA

Iako je sekvenciranje DNA ogromnog broja mikroorganizama prestalo biti limitirajući faktor u istraživanjima, snalaženje među dobivenim podacima zapravo predstavlja još veći izazov. Ovisno o metodi sekvenciranja i svrsi istraživanja, za obradu i analizu dobivenih podataka potrebni su specifični bioinformatički alati i baze podataka. Stoga postoji stalna potreba za kontinuiranim ažuriranjem i proširivanjem baza podataka te za razvijanjem novih bioinformatičkih alata (D'Argenio, 2018; Singh i sur., 2017).

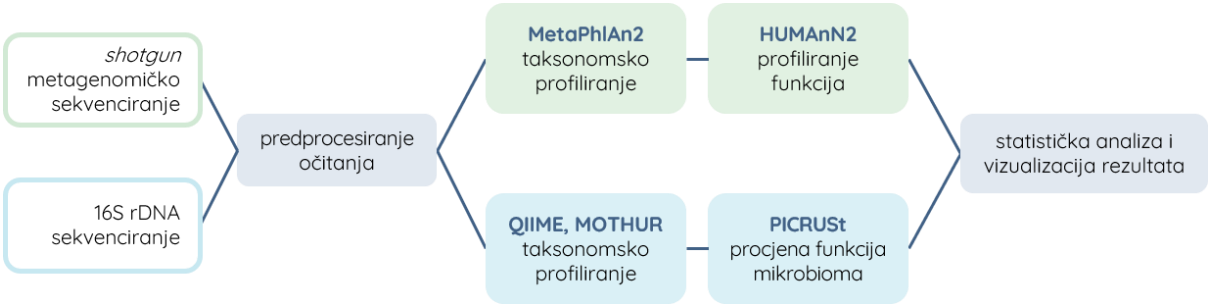
2.3.1. Obrada podataka 16S rDNA sekvenciranja

Sekvenciranjem 16S rDNA dobivaju se opsežna 'sirova' očitanja, koja su često vrlo fragmentirana, preklapaju se, te sadrže kontaminacije i šumove. Bioinformatička obrada omogućuje pročišćavanje takvih 'sirovih' podataka (Jandhyala i sur., 2015; Haas i sur., 2011; Schloss i sur., 2011). Nakon kontrole kvalitete slijedi taksonomska identifikacija očitanja dobivenih 16S rDNA sekvenciranjem. Provodi se pomoću zasebnih programa ili kao dio *pipeline*-a (hrv. cjevovod), odnosno kao dio definiranog niza alata i programa za obradu tih podataka. Temelji se na usporedbi sličnosti očitanja sekvenci s poznatim sekvencama 16S rDNA koje se nalaze u referentnim bazama podataka. Neke od najkorištenijih baza su EzBioCloud (Yoon i sur., 2017), GenBank (Benson i sur., 2005), Greengenes (DeSantis i sur., 2006), Ribosomal Database Project (RDP) baza (Cole i sur., 2014), i SILVA (Quast i sur., 2013). Očit nedostatak ovakvog pristupa je činjenica da se mogu identificirati samo one bakterijske vrste koje su poznate i anotirane u navedenim bazama podataka, pa neće nužno sva očitanja biti uspješno taksonomski identificirana (D'Argenio, 2018; Singh i sur., 2017). Dobivena očitanja sekvenci također se mogu grupirati na temelju međusobnih filogenetskih odnosa ili grupiranjem operacijskih taksonomskih jedinica (eng. *Operational Taxonomic Unit, OTU*) (Singh i sur., 2017).

Iako se 16S rDNA sekvenciranjem dobivaju samo podaci o taksonomskoj strukturi analiziranih zajednica mikroorganizama, postoji sve više specifičnih *pipeline*-ova koji omogućuju predviđanje funkcija vezanih uz te zajednice (D'Argenio, 2018). PICRUSt je bioinformatički alat koji omogućuje procjenu funkcionalnog sastava mikrobioma tako što predviđa prisutne obitelji gena i njihove relativne brojnosti (Langille i sur., 2013).

2.3.2. Obrada podataka *shotgun* sekvenciranja

Obrada podataka dobivenih *shotgun* sekvenciranjem mnogo je kompliciranija i zahtjevnija, jer je često riječ o ogromnoj količini podataka s nepotpunom pokrivenosti, te se uz to dobivaju očitanja sekvenci koje originalno pripadaju čovjeku (< 1 %), ali i arhejama, bakterijama, fungima, te virusima (D'Argenio, 2018; Morgan i Huttenhower, 2014). *Pipeline* za analizu predprocesiranih metagenomičkih podataka uključuje sastavljanje sekvenci, taksonomsko profiliranje, predviđanje gena i metaboličko profiliranje. Međutim, često se koriste i programi koji preskaču fazu sastavljanja sekvenci, kao što su MetaPhlAn2 i HUMAnN2 (Slika 4) (Singh i sur., 2017).



Slika 4. Pojednostavljeni primjer usporedbe tijeka rada bioinformatičke obrade podataka dobivenih *shotgun* metagenomičkim sekvenciranjem (bez sastavljanja) i podataka dobivenih 16S rDNA sekvenciranjem (Prilagođeno prema Jandhyala i sur., 2015; Morgan i Huttenhower, 2014).

Jedan od glavnih ciljeva metagenomičkog *shotgun* sekvenciranja čitavog genoma je identifikacija svih mikroorganizama u uzorku, i to na najnižoj taksonomskoj razini. Postoji nekoliko programskih paketa koji grupiraju sekvence dobivene *shotgun* sekvenciranjem na temelju njihovog sastava ili sličnosti referentnim sekvencama. Najbržu usporedbu sličnosti omogućuju AMPHORA2 (Wu i Scott, 2012), AmphoraNet (Kerepesi i sur., 2014) i MetaPhlAn (Segata i sur., 2012), jer vrše pretragu samo dijelova genoma koji su informativniji za taksonomsku identifikaciju, umjesto pretrage cijelog genoma (Singh i sur., 2017).

MetaPhlAn2 (*Metagenomic Phylogenetic Analysis*) je bioinformatički alat za precizno taksonomsko profiliranje zajednica mikroorganizama. Koristi ograničen broj jedinstvenih marker gena koji su specifični za pojedine podgrupe, odnosno klade na filogenetskom stablu, što olakšava brzu i direktnu identifikaciju taksona te određivanje brojnosti pojedinih taksona (McIver i sur., 2018; Segata i sur., 2012).

Iz sastavljenih i nesastavljenih genoma mogu se odrediti funkcije zajednica mikrobiote pomoću referentnih baza podataka poput dbCAn (Yin i sur., 2012), HUMAnN (Abubucker i sur., 2012), IMG (Markowitz i sur., 2014), i MetaRef (Huang i sur., 2014). HUMAnN2 (*The HMP Unified Metabolic Analysis Network*) je bioinformatički alat za utvrđivanje odsutnosti ili prisutnosti biokemijskih puteva, te određivanje njihove brojnosti u zajednicama mikrobiote. U funkcijske profile integrira taksonomske profile, tako da se dobivene brojnosti biokemijskih puteva automatski povezuju s poznatim i nepoznatim vrstama koje doprinose tim putevima (Franzosa i sur., 2018; McIver i sur., 2018).

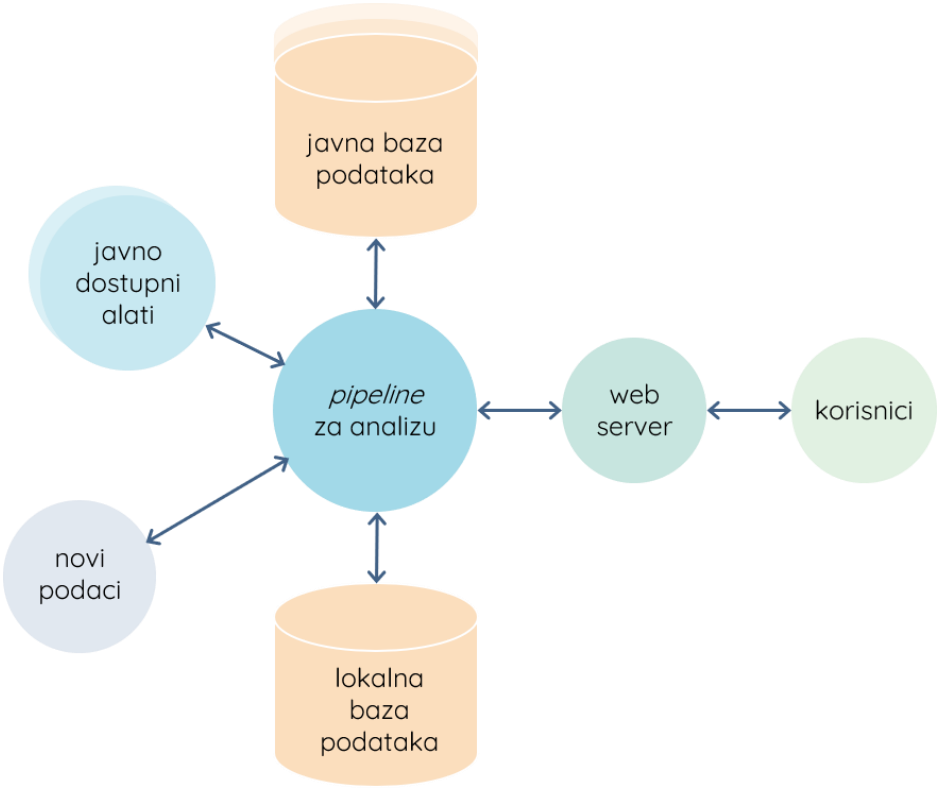
2.3.3. Određivanje bioraznolikosti

Statističke analize zajednica mikrobiote uglavnom su se fokusirale na deskriptivne ekološke mjere, u svrhu određivanja njihove bioraznolikosti. Podaci dobiveni sekvenciranjem nakon inicijalne obrade koriste se za određivanje bogatstva vrsta (eng. *species richness*), brojnosti vrsta (eng. *species abundance*), relativne brojnosti (eng. *relative abundance*), alfa i beta raznolikosti, i drugih parametara. Izračun relativne brojnosti nakon *shotgun* sekvenciranja čitavog genoma temelji se na jedinstvenom markeru, pa je mnogo točniji nego izračun s podacima 16S rDNA sekvenciranja (Singh i sur., 2017; Jandhyala i sur., 2015; Morgan i Huttenhower, 2014; Poretsky i sur., 2014).

Alfa raznolikost podrazumijeva mikrobnu raznolikost unutar uzorka, ekološke zajednice, ekosustava, ili bilo koje prostorne cjeline. Tipično je kvantificirana direktno u vidu broja taksona, najčešće vrste bakterija, ili putem nekih od indeksa alfa raznolikosti. Može se koristiti za kvantifikaciju smanjenja, odnosno gubitka bioraznolikosti, što je slučaj pri upotrebi antibiotika ili u slučaju određenih bolesti. Dva najčešće korištena indeksa za izračunavanje alfa raznolikosti na temelju brojnosti vrsta su Shannonov indeks i Simpsonov indeks (Veech, 2018; Singh i sur., 2017; Morgan i Huttenhower, 2014). Beta raznolikost podrazumijeva mikrobnu raznolikost među različitim uzorcima koji se uspoređuju. Omogućuje procjenu mjere u kojoj su taksoni ili funkcije zajednički za dva odabrana uzorka. Visoka beta raznolikost znači da su uzorci različiti, a niska da su vrlo slični s obzirom na vrste koje sadrže (Singh i sur., 2017; Morgan i Huttenhower, 2014). Popularni metagenomički *pipeline*-ovi poput mothur (Schloss i sur., 2009) i QIIME 2 (Boylen i sur., 2019) uključuju različite metode koje računaju alfa i beta raznolikost na temelju relativne brojnosti taksona ili brojnosti OTU (Jandhyala i sur., 2015; Morgan i Huttenhower, 2014).

2.4. BAZE PODATAKA

Istraživanjima se gomilaju podaci kao što su sekvence genoma, strukture proteina, metabolomički profili, i slično. Osim što uključuje obradu takvih podataka, bioinformatika zapravo omogućuje izvlačenje novih bioloških znanja iz dobivenih bioanalitičkih podataka. Ključne komponente generičkog bioinformatičkog rješenja su (Slika 5): *pipeline* za analizu (Perl, Python, R), lokalna baza podataka (npr. MySQL) i web sučelje (HTML5, Apache). Podaci se ubacuju u strukturiranu bazu podataka, a zatim se primjerice koriste Perl ili Python za automatizaciju manipuliranja podacima. Dodatno se može koristiti i R za sofisticirane analize i sposobnost vizualizacije kako bi se dobile korisne informacije. Korištenje takvih alata otvorenog koda (eng. *open source*) je poželjno, jer je zbog velike zajednice korisnika lakše dobiti podršku, a uz to se stalno stvaraju dodatni moduli za Perl i paketi za R koji provode razne bioinformatičke zadatke. Postoje i javno dostupne platforme za bioinformatički tijek rada, kao Taverna, Knime, i Galaxy (Bessant i sur., 2014).



Slika 5. Prikaz tipičnih komponenti generičkog bioinformatičkog rješenja. Lokalna baza podataka obično je potrebna za pohranu novo dobivenih podataka, radi njihove lakše organizacije i analize (Prilagođeno prema Bessant i sur., 2014).

2.4.1. Sustavi za upravljanje bazama podataka

Baza podataka je strukturirana kolekcija podataka koji su pohranjeni u nizovima bajtova u memoriji računala. Unosi podataka mogu biti odvojene datoteke ili jedna jedinstvena datoteka. Pri tome je bitan format takvih datoteka, te aplikacije koje se mogu koristiti za kodiranje (pisanje) i dekodiranje (čitanje) podataka. U bioinformatici se najčešće koriste relacijske baze podataka, jer pružaju sustavnost i sigurnost podataka. Drugi tip baza s kojima se često susreće u bioinformatici su *flat* baze podataka (Bessant i sur., 2014).

U svakodnevnom govoru se pojam 'baza podataka' zapravo odnosi na kolekciju podataka kojom se upravlja pomoću specijalizirane računalne aplikacije – sustava za upravljanje bazom podataka (eng. *Database Management System, DBMS*). DBMS je skup programa koji služi za stvaranje novih baza podataka, unošenje i pretraživanje podataka, te pruža mogućnost modificiranja podataka. Pomoću specijaliziranog jezika za definiranje podataka (eng. *Data Definition Language, DDL*) specificira se shema baze podataka koja daje podacima logičku strukturu, a pomoću specijaliziranog jezika za manipulaciju podacima (eng. *Data Manipulation Language, DML*) moguće je provođenje upita i izmjena podataka. Podaci su zahvaljujući DBMS-u istovremeno dostupni za više korisnika i programa. Uz to, pruža kontrolu pristupa podacima, kao i trajnost te mogućnost oporavka baze podataka ukoliko je potrebno (Bessant i sur., 2014; Westhead i sur., 2002; Date, 2000).

2.4.1.1. 'Flat' baze podataka

Flat baze podataka sastoje se od jedne ili više datoteka, a u bioinformatici su obično strukturirane kao kolekcije unosa koji opisuju specifične cjeline podataka. Obične tekstualne datoteke imaju *flat* format, sa znakovima iz proširenog ASCII seta ili Unicode seta. Među češće *flat* formate pripada CSV (eng. *Comma-Separated Values*), u kojem su vrijednosti odijeljene zarezima, i često služi za prijenos podataka među heterogenim platformama. Što se tiče bioinformatike, jedan od češćih *flat* formata je FASTA, za pohranjivanje sekvenci proteina i nukleotida. Neke od trenutnih online dostupnih *flat* baza bioloških podataka, poput EMBL (Kanz i sur., 2005) i GenBank (Benson i sur., 2005), sadrže stotine gigabajta teksta. Nedostatak je što je u datotekama s puno podataka teže i vremenski zahtjevnije pronaći traženi podatak. Stoga je razvijen XML (eng. *eXtensible Markup Language*) s namjerom da postane internacionalni standard. Dodaje sintaksu tekstualnim datotekama radi lakšeg pronalaženja podataka, pa su prošireni alati i knjižnice za čitanje i pohranjivanje podataka u XML formatu.

Flat bazama podataka mogu se smatrati i jednostavne proračunske tablice, poput onih napravljenih računalnim programom Microsoft Excel, te čak i linearna skladišta NoSQL podataka. NoSQL je pojam koji obuhvaća mnogo različitih 'skladišta' podataka. Naziv je akronim s doslovnim značenjem 'ne samo SQL' (eng. *Not Only SQL*), budući da kod takvih baza podataka SQL nije nužan. U neki slučajevima nije potrebna ni shema baze, i podržavaju mnogo različitih struktura podataka. Većina takvih baza služi za široku distribuciju podataka. (Grolinger i sur., 2013; Damian, 2009).

2.4.1.2. Relacijske baze podataka

Relacijska baza podataka je prema relacijskom modelu organizirana u zasebne tablice, odnosno 'relacije'. Svaka tablica ima vlastito ime po kojem se razlikuje od ostalih u bazi, i zapravo je svaka set redaka i stupaca. Svaki redak u tablici je jedan zapis i sadrži jednak broj stupaca, a svaki stupac ima definiranu vrstu vrijednosti i može se smatrati specifičnim svojstvom. Često postoji i stupac koji sadrži jedinstveni identifikator (ID) za svaki redak u tablici. Jedan redak može biti jedan primjerak unosa, ili samo bilježiti vezu između više unosa, ali isti unosi se ne smiju ponoviti unutar jedne tablice. Tablice u relacijskoj bazi podataka međusobno su povezane tako da se može pristupiti podacima iz jedne uz vezane informacije iz druge. Odnosi i ograničenja među tablicama su definirani kako bi se osigurala dosljednost podataka.

Za pristup, dodavanje i procesiranje podataka pohranjenih u relacijsku bazu podataka potreban je specijalizirani sustav za upravljanje relacijskim bazama podataka (eng. *Relational Database Management System, RDBMS*). Pomoću RDBMS je moguće definirati tablice i formate pojedinih stupaca. Također služi za definiranje odnosa među tablicama te proces normalizacije podataka, kojim se uklanjaju viškovi, primjerice nenamjerno duplicirani podaci. Modeliranjem se stvara shema baze, koja služi kao funkcionalna mapa strukture te baze podataka. Među najčešće korištenim RDBMS su Oracle, MySQL, PostgreSQL i Microsoft SQL server. Takvi sustavi pružaju značajnu izvedbu i funkcionalne prednosti, a većina pruža funkcije grupiranja i raspodjele podataka među više uređaja kako bi se lakše nosilo s rastućom količinom bioinformatičkih podataka (Bessant i sur., 2014; Damian, 2009; Ullman, 2003).

2.4.2. Računalni jezik SQL

SQL (eng. *Structured Query Language*) je računalni jezik za upravljanje relacijskim bazama podataka, koji istovremeno služi kao DML i DDL, a uz to i kao jezik za kontrolu podataka (eng. *Data Control Language, DCL*) te kao jezik za kontrolu transakcija (eng. *Transaction Control Language, TCL*). Razvili su ga Chamberlin i Boyce, i prvi puta je objavljen 1972. pod nazivom SEQUEL (eng. *Structured English Query Language*). Kasnije je naziv promijenjen u SQL te se kao takav proširio zahvaljujući univerzalnosti koju mu je pružila standardizacija. Standardizacija programskog jezika olakšava programerima korištenje baza podataka među različitim platformama. Prvi standard SQL-a objavljen je 1986. godine, i taj dijalekt poznat je kao SQL-86. Nakon toga su provedene daljnje revizije standarda, pa tako postoje i dijalekti SQL-89, SQL-92, SQL:1999 (SQL3), SQL:2003, SQL:2006, SQL:2008, SQL:2011 i SQL:2016. Najnoviji i trenutno aktualan standard je ISO/IEC 9075-15:2019 (SQL:2019), objavljen u lipnju 2019. godine. Međutim, svaki sustav za upravljanje relacijskim bazama podataka koristi proširenu verziju SQL standarda, s vlastitim naredbama i funkcijama. Osnovnim naredbama računalnog jezika SQL moguće je provesti gotovo sve akcije potrebne za upravljanje bazom podataka, a naprednim kombinacijama naredbi moguće je provoditi i aritmetičke operacije, filtrirati podatke prema zadanim uvjetima, sortirati podatke, i još mnogo toga (ISO, 2019; ANSI, 2018; Ullman i Widom, 2007; Suehring, 2002; Chamberlin i Boyce, 1972).

2.4.2.1. Provođenje SQL query-ja

Query je doslovno upit, odnosno pitanje o podacima u bazi, te se kao rješenje dobivaju rezultati, odnosno odgovori na postavljeno pitanje. Većina interakcija s DBMS svodi se na to da korisnik ili određena aplikacija provodi akciju koristeći DML, čime se ne utječe na shemu podataka. Naredbe za modifikaciju mogu utjecati na sastav podataka u bazi, dok *query* akcije mogu izvući željene podatke iz baze. Pri provođenju upita postoji određeni redoslijed kojim se upit obrađuje kako bi se dobili rezultati (Prilog 2.1). Prevoditelj upita parsira i optimizira upit te stvara plan, odnosno redoslijed akcija koje DBMS provodi kako bi odgovorio na upit. Plan se dalje predaje nizu upravljača i mehanizama kako bi u konačnici izvukao potrebne podatke iz diska. Korisnik se u praksi najviše susreće s procesorom upita, sastavljenim od prevoditelja i mehanizma za upravljanje izvršavanjem upita (Ullman i Widom, 2007; Grune i Jacobs, 1990).

2.4.2.2.Pohranjeni postupak

Pohranjeni postupak (eng. *stored procedure*) je rutina pohranjena u bazi podataka. Svaka rutina ima jasno definirano ime i parametre, te je moguće postavljati različite ulazne varijable s kojima će se provoditi. Sastavljena je od SQL izjava koje sačinjavaju specifičan upit, a mora se pozvati pomoću naredbe `CALL`. Jednom kad se rutini predaju parametri, provesti će se ovisno o uvjetima upita na unaprijed definiran način. Rutina uvijek ostaje ista, bez obzira koliko je puta korištena, pa se može opetovano koristiti uz izmjene ulaznih varijabli prema potrebi. Pri tome nije potrebno svaki put sastavljati čitav SQL upit, već je dovoljno pozvati rutinu i zadati joj parametre. Prednost korištenja pohranjenih postupaka je to što se nalaze unutar baze podataka i mogu se koristiti s bilo kojeg računala s pristupom toj bazi podataka (MySQL, 2019).

2.4.3. Programski paket MySQL

Od svih sustava za upravljanje relacijskim bazama podataka, MySQL nudi najbolju kombinaciju funkcija. MySQL je računalna aplikacija otvorenog koda koja se može slobodno nadograđivati, a u vlasništvu je Oracle korporacije. Napisan je programskim jezikom C++. Ima više verzija, kako bi bio dostupan za više platformi, prvenstveno za popularne Windows i Linux operativne sustave. Uz to ima mnogo sučelja za programiranje aplikacija (eng. *Application Programming Interface, API*), čime pruža mogućnost pristupa i modifikacija baze podataka putem različitih programskih jezika. Primjerice, dostupni su API za Python, PHP, JavaScript, C i C++. Pri tome su Python i PHP najpopularniji za programiranje web sučelja aplikacija, pogotovo za bioinformatičke potrebe. MySQL baza podataka podržava uvoz podataka iz delimitiranih tekstualnih datoteka, kao što su CSV datoteke. Mnogi programi imaju opciju izvoza podataka u CSV formatu, zahvaljujući čemu ih je jednostavno uvesti u MySQL bazu podataka. Još jedna korisna opcija MySQL baze podataka je mogućnost brzog izvoza čitavih baza podataka u `mysqldump` datoteku, iz koje se ponovo mogu uvesti u MySQL uz automatsko strukturiranje tablica. Sama aplikacija, odnosno programski paket, postoji u više izdanja dostupnih za korisnike, među kojima je Community Edition besplatno i dostupno svima. Ovisno o izdanju, MySQL se sastoji od više komponenti, i mnogo programa i alata. Glavni program je MySQL Server, uz MySQL Client potreban za spajanje na server. Zatim postoje MySQL Database, MySQL Workbench, MySQL Shell, MySQL for Excel, MySQL for Visual Studio, i druge programske komponente (MySQL, 2019; Bessant i sur., 2014; Suehring, 2002).

2.4.3.1. MySQL Server

MySQL Server, poznat i kao `mysqld`, je glavni program za upravljanje pristupom katalogu podataka u kojem se nalaze tablice i baze podataka. Uz to, u katalogu podataka nalaze se i druge informacije poput datoteka sa zapisima i datoteka statusa. Server može biti lokalan, na računalu koje se koristi, ili se može povezati s udaljenim serverom i pristupiti udaljenoj bazi podataka. Također, server može imati neograničen broj korisnika, a za svakoga se definiraju privilegije i način pristupa (MySQL, 2019).

2.4.3.2. MySQL Workbench

MySQL Workbench je besplatan alat s grafičkim korisničkim sučeljem (eng. *graphical user interface, GUI*) dostupan u sklopu MySQL Community Edition-a i MySQL Enterprise Edition-a. Služi za provođenje četiri glavne grupe funkcija: razvoj baze, modeliranje podataka, i administraciju servera, i migraciju baze podataka. Sadrži vizualni SQL uređivač sastavljen od specijaliziranih uređivača za upite, shemu, tablice, i slično. Služi za sastavljanje, uređivanje i provođenje upita, putem kojih se među ostalim mogu stvarati i uređivati podaci u bazi podataka. Rezultati provedenih upita se prikazuju grafički, i mogu se izvesti iz baze podataka u željenom formatu. MySQL Workbench uz to olakšava kreiranje sheme baze podataka jer ju je moguće kreirati u obliku EER (eng. *enhanced entity-relationship*) dijagrama, gdje su tablice i njihovi međusobni odnosi prikazani grafički. Uz sve navedeno, putem MySQL Workbench-a je moguće putem grafičkog sučelja obavljati sve uobičajene administrativne zadatke, poput upravljanja korisničkim računima, te stvaranja sigurnosnih kopija i vraćanja baze podataka. Nudi i mogućnost migracije podataka iz drugih sustava za upravljanje relacijskim bazama podataka u MySQL sustav (MySQL, 2019; Bessant i sur., 2014).

2.4.3.3. Računalni alat SQLyog

SQLyog je profesionalan GUI alat za MySQL sustav za upravljanje relacijskim bazama podataka. Napisan je C++ programskim jezikom, a komunicira s MySQL aplikacijom pomoću njenog C API-ja. Dostupan je za nekoliko platformi, kao i sama MySQL aplikacija, i to u nekoliko izdanja. Dizajniran je kako bi nadoknadio neke nedostatke MySQL Workbench-a i dodatno olakšao administraciju baza podataka. Dva vrlo korisna svojstva SQLyog alata su pametno automatsko popunjavanje naredbi, te napredniji vizualni uređivači upita i shema. (Laursen, 2017).

3. EKSPERIMENTALNI DIO

3.1. MATERIJALI

Za izradu ovog diplomskog rada korišteno je prijenosno računalo s instaliranim potrebnim programima i alatima, dok su upotrijebljeni biološki podaci već prethodno uneseni i organizirani u bazu podataka prema određenoj shemi.

3.1.1. Sklopovlje (hardware)

Korišteno je prijenosno računalo Asus serije ROG, model GL752VW, sljedećeg sklopovlja (eng. *hardware*):

- procesor Intel® Core™ i7-6700HQ od 2,6 GHz
- radna memorija 8 GB
- čvrsti disk SSD (eng. *solid-state drive*) kapaciteta 118 GB
- čvrsti disk HDD (eng. *hard-disk drive*) kapaciteta 1 TB
- grafička kartica Nvidia GeForce GTX 960M.

3.1.2. Operativni sustav

Na SSD čvrstom disku korištenog prijenosnog računala instaliran je 64-bitni operativni sustav Microsoft Windows 10 Home[1], verzija 1903.

3.1.3. Programska podrška (software)

U ovom radu korišten je računalni jezik SQL, te nekoliko programa i alata potrebnih za upravljanje bazom podataka, analize i grafički prikaz rezultata. Među te programe spadaju:

- MySQL Community Version
- SQLyog Community Version (64-bit)
- Microsoft Office Excel
- MySQL for Excel.

[1] Microsoft Windows 10 Home Edition: <https://www.microsoft.com/hr-hr/p/windows-10-home/d76qx4bznwk4?activetab=pivot:overviewtab>

3.1.3.1. Računalni jezik SQL

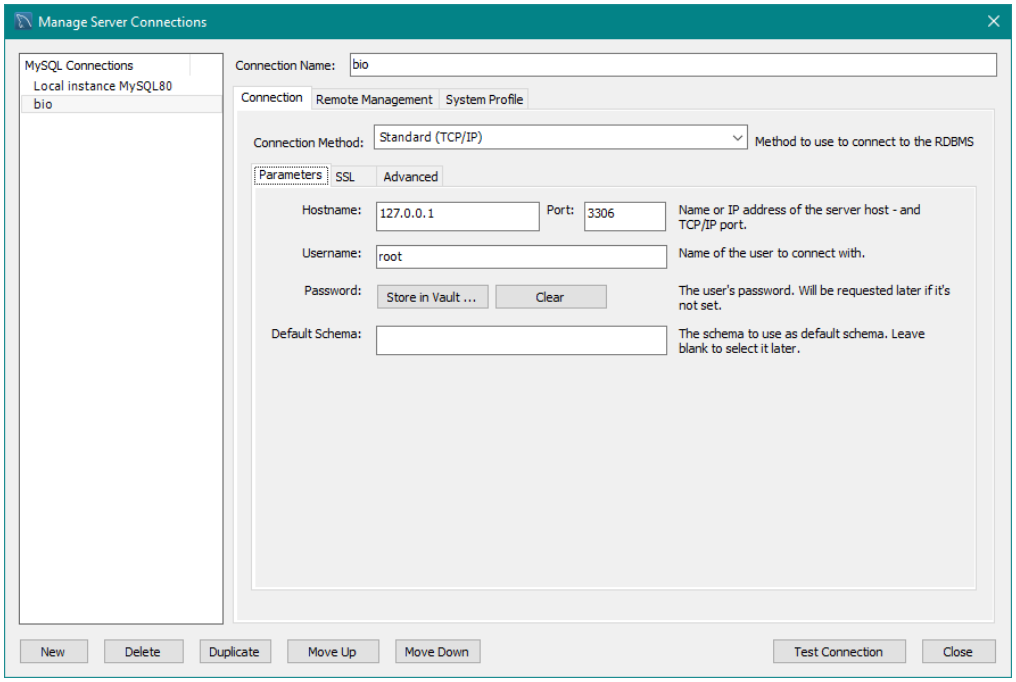
Računalni jezik SQL u ovom je radu upotrijebljen za pisanje i uređivanje upita pomoću kojih su provedene različite analize bioloških podataka organiziranih u bazu podataka. Pomoću njega je vrlo jednostavnim naredbama moguće provoditi sve osnovne akcije nužne za održavanje i upravljanje bazom podataka, poput kreiranja novih tablica, dodavanja ili brisanja podataka, premještanja podataka, i slično. Također je moguće sortirati, filtrirati i uspoređivati podatke, čak i provoditi aritmetičke operacije, ukoliko se slože naprednije kombinacije naredbi.

Računalni jezik SQL u suštini nema mnogo naredbi, te ima vrlo jednostavnu sintaksu, ali njime je moguće provesti relativno zahtjevne akcije. Sustav za upravljanje bazama podataka MySQL koristi vlastitu verziju SQL standarda, te je u opširnim uputstvima za upotrebu[2] dostupan pun popis svih naredbi s vezanim objašnjenjima. Za potrebe ovog rada većinom su korišteni operatori i funkcije, uključujući logičke, aritmetičke i matematičke, a uz to su korištene i neke izjave za definiranje podataka (eng. *Data Definition Statements*) te izjave za manipuliranje podataka (eng. *Data Manipulation Statements*).

3.1.3.2. Programski paket MySQL

Sa službenih stranica MySQL-a[3] preuzet je MySQL Installer for Windows, koji omogućuje čarobnjačku (eng. *wizard-based*) instalaciju najnovijih verzija svih željenih programa u sklopu programskog paketa MySQL Community Edition (GPL) za Windows operativni sustav. Iako je sam MySQL Installer 32-bitni, pomoću njega je moguće instalirati i 64-bitnu verziju MySQL Community Edition-a, što je u ovom slučaju učinjeno. Prilikom instalacije paketa odabran je ponuđeni set postavki za programere (eng. *developer setup*), te je instalirana 8.0.18. verzija svih programskih komponenti. Dvije najvažnije komponente za izradu ovog rada su MySQL Community Server i MySQL Workbench. MySQL Workbench je alat s grafičkim korisničkim sučeljem koji pojednostavljuje korištenje MySQL servera i pristup bazi podataka, a ujedno i omogućuje uređivanje sheme baze podataka te pisanje upita. Pomoću MySQL Workbench alata odabire se jedna od definiranih veza (eng. *connection*) za spajanje na server ili se stvara nova veza. Za potrebe izrade ovog rada korištena je veza s lokalnim serverom nazvana `bio` (Slika 6), a kao korisničko ime odabrano je `root`, jer ima pune administratorske privilegije.

[2] MySQL 8.0. Reference Manual: <https://dev.mysql.com/doc/refman/8.0/en/>
[3] MySQL Community Edition: <https://www.mysql.com/products/community/>



Slika 6. Prozorčić postavki veze `bio` s lokalnim MySQL serverom.

3.1.3.3. Računalni alat SQLyog

Unatoč praktičnosti MySQL Workbench-a, dodatno je s GitHub-a[4] preuzet i instaliran računalni alat SQLyog Community Edition, verzija MySQL GUI 13.1.2. (64-bit). Radi preglednijeg i jednostavnijeg grafičkog korisničkog sučelja, ovaj je računalni alat primarno korišten za pisanje, uređivanje i testiranje upita za analizu bioloških podataka iz baze podataka unesene u MySQL. Alat SQLyog spojen je na lokalni MySQL server putem veze `bio` koja je prethodno stvorena pomoću MySQL Workbench-a. Time je dobiven pristup bazi podataka sa svim popratnim privilegijama, kao što je slučaj kod direktnog spajanja unutar MySQL Workbench-a. Nakon što je omogućen pristup, moguće je korištenje uređivača upita.

3.1.3.4. Računalni program Microsoft Excel i dodatak MySQL for Excel

Podaci su grafički prikazani pomoću programa Microsoft Office[5] Excel 2016, već prethodno instaliranog na korištenom računalu. Dodatno je sa službenih stranica MySQL-a preuzet i instaliran MySQL for Excel[6]. Dodatak MySQL for Excel omogućuje spajanje na vezu `bio` i direktan pristup podacima u bazi kroz program Excel.

[4] SQLyog Community Edition: <https://github.com/webyog/sqlyog-community/wiki/Downloads>
[5] Microsoft Office: <https://products.office.com/hr-hr/home>
[6] MySQL for Excel: <https://dev.mysql.com/downloads/windows/excel/>

3.1.4. Biološki podaci dobiveni sekvenciranjem

Biološki podaci na čijoj se analizi temelji izrada ovog rada dobiveni su sekvenciranjem velikog broja uzoraka crijevne mikrobiote. Zatim su podaci obrađeni te oblikovani prema shemi i pohranjeni u bazu podataka. Čitav skup tako dobivenih podataka kategoriziran je kao jedan projekt, a unutar projekta se nalaze podaci za 1639 različitih uzoraka, od kojih svaki pripada različitom pojedincu. Budući da je baza podataka izrađena na engleskom jeziku, u daljem tekstu, slikama i tablicama navode se neki izvorni podaci napisani na engleskom jeziku uz njihova pojašnjenja na hrvatskom jeziku. U prilogu se na CD-u nalazi datoteka `bio.sql` koja sadrži shemu i sve podatke iz baze `bio`.

3.1.4.1. Protokol za uzorkovanje, sekvenciranje, i obradu podataka

Od dobrovoljaca su prikupljeni zamrznuti uzorci stolice, te je iz njih izolirana DNA prema već uspostavljenom protokolu (Imhann i sur., 2019; Tigchelaar i sur., 2015). Provedeno je metagenomičko sekvenciranje pomoću Illumina HiSeq platforme. Obrada sirovih očitanja provedena je pomoću KneadData (verzija 0.5.1.) (McIver i sur., 2018) i Bowtie2 (verzija 2.3.4.1.) alata (Clausen i sur., 2018), a kvaliteta procesiranih podataka provjerena je korištenjem FastQC alata (verzija 0.11.7.) (Andrews, 2010). Taksonomsko profiliranje metagenoma provedeno je pomoću MetaPhlAn2 alata (verzija 2.7.2.) prema marker genima iz MetaPhlAn baze mpa_v20_m200 (Segata i sur., 2012). Određivanje biokemijskih puteva provedeno je koristeći HUMAnN2 *pipeline* (verzija 0.11.1.) (Truong i sur., 2015). Dio podataka je nakon obrade i profiliranja parsiran i unesen u bazu podataka nazvanu `bio`, te su podaci kao takvi izvezeni u obliku `bio-data.sql` datoteke.

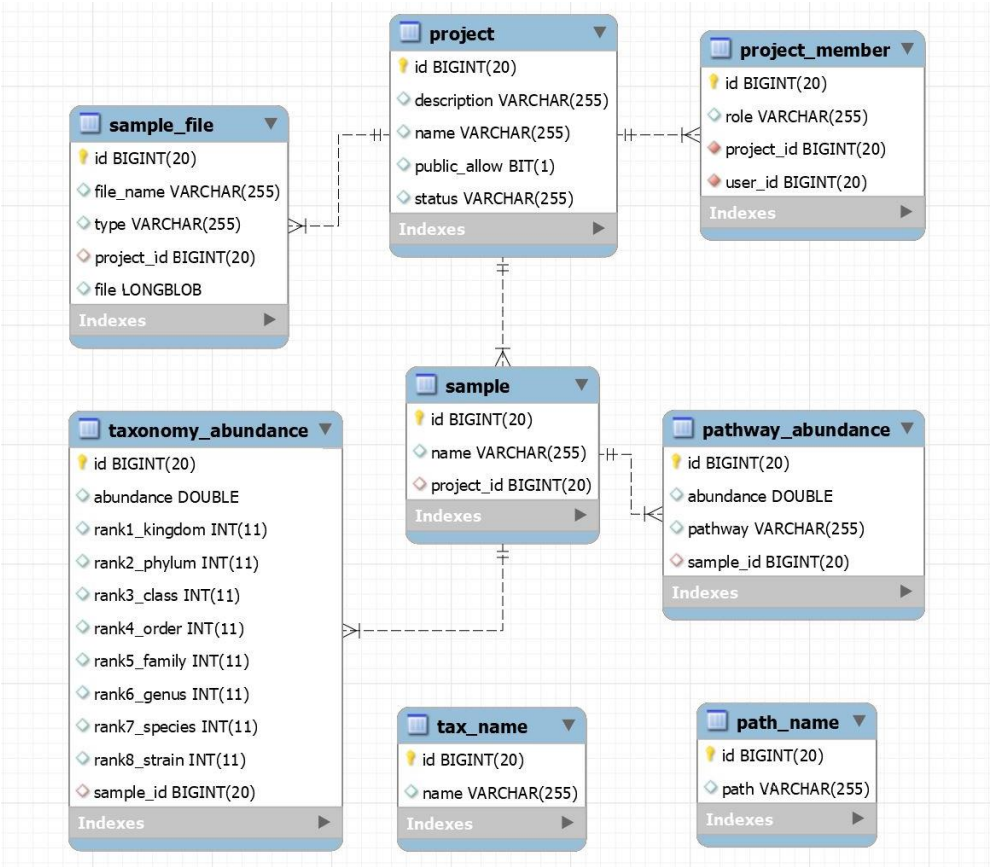
3.1.4.2. Shema baze podataka u MySQL-u

Za potrebe izrade ovog rada dobivena je gotova datoteka `bio-data.sql`, te je uvezena u MySQL Workbench kao gotova shema `bio` s podacima (Slika 7). Za izradu su korišteni podaci unutar tri tablice: `sample`, `taxonomy_abundance` i `tax_name`.

Tablica `sample` (hrv. uzorak) sadrži podatke o uzorcima u tri stupca: `id`, `name` i `project_id`. Svi redovi u stupcu `project_id` za potrebe izrade ovog rada sadrže samo jednu vrijednost – broj 1, budući da svi podaci u bazi podataka `bio` pripadaju jednom jedinom projektu. Stupac `name` sadrži jedinstvena imena uzoraka sačinjena od kombinacije slova i brojeva.

Najvažniji stupac je `id`, jer pojednostavljuje provođenje raznih analiza i filtriranje uzoraka. Sadrži pozitivne cijele brojeve, pri čemu je svaki broj strogo vezan uz samo jedan uzorak. To je osigurano korištenjem naredbe `AUTO_INCREMENT`, koja za svaki novi redak generira pozitivan cijeli broj za jedan veći od prethodnog. Zbog toga je stupac `id` određen kao `PRIMARY_KEY` za `sample` tablicu, jer sadrži jedinstvene vrijednosti.

Tablica `taxonomy_abundance` (hrv. taksonomska brojnost) sadrži podatke o brojnosti taksona u pojedinim uzorcima. Pomoću stupca `sample_id` svaka je vrijednost brojnosti dobivena pomoću MetaPhlAn2 alata povezana s odgovarajućim uzorkom na temelju njegovog identifikatora iz tablice `sample`. Preostali stupci predstavljaju sve razine taksonomske klasifikacije redom: kraljevstvo, koljeno, razred, red, obitelj, rod, vrsta i soj. Ispunjeni su tako da svaka razina ima odgovarajuću brojnost za pojedini uzorak. Brojevi koji se nalaze u tim stupcima poveznice su s tablicom `tax_name` gdje svaki odgovara jednom od imena pojedinih taksona.



Slika 7. EER dijagram (eng. *Enhanced Entity-Relationship*) sheme `bio` s prikazanim relacijama među pojedinim tablicama, definiranim vrstama vrijednosti te označenim jedinstvenim identifikatorima (zlatni ključ pokraj imena stupca).

3.2. METODE

Osmišljeno je 5 glavnih analiza podataka o crijevnoj mikrobioti za 1639 uzoraka unesenih u bazu podataka `bio`. Svaka analiza pisana je kao zaseban upit unutar uređivača u sklopu računalnog alata SQLyog. Upiti su sastavljeni kao pohranjeni postupci, odnosno rutine s odgovarajućim brojem ulaznih varijabli. Iako je kompliciranija pri sastavljanju upita, ovakva metoda je odabrana jer omogućuje jednostavno i brzo provođenje analiza uz lake izmjene varijabli prema potrebi. Uz to nema potrebe za dodatnim pohranjivanjem napisanih upita, s obzirom da se rutine pospremaju unutar same baze podataka, gdje su uvijek dostupne.

3.2.1. Bogatstvo vrsta

Bogatstvo vrsta (eng. *species richness*) jedna je od standardnih mjera bioraznolikosti zahvaljujući svojoj jednostavnosti i lakoj interpretaciji, a podrazumijeva broj različitih vrsta prisutnih u određenoj prostornoj cjelini koja je uzorkovana. U slučaju ovog rada, bogatstvo vrsta pojedinog uzorka odnosi se na broj različitih vrsta unutar zajednica crijevne mikrobiote pojedinaca. Može se prikazati na određenoj razini taksonomske klasifikacije za odabrani uzorak, ili pak kao srednja vrijednost na razini cijelog projekta, što obuhvaća svih 1639 uzoraka. Uz to, pomoću upita je čak moguće dobiti podatke potrebne za konstrukciju kutijastog dijagrama (eng. *box plot*).

3.2.1.1. Bogatstvo vrsta odabranog uzorka

Pohranjeni postupak nazvan `sample_top_richness` ima tri ulazne varijable: identifikator uzorka, razinu taksonomske klasifikacije, i limit za prikazivanje rezultata. Unosom identifikatora uzorka osigurano je da se prikazuju rezultati koji se odnose samo na taj uzorak, a unosom razine taksonomske klasifikacije da se rezultati prikazuju samo na toj razini. Limitom je ograničen broj redaka rezultata, što je posebno korisno na nižim taksonomskim razinama poput roda ili vrste, gdje uzorci s velikim bogatstvom vrsta daju i mnogo rezultata. Također, zahvaljujući limitu, može se uspoređivati primjerice 5 najzastupljenijih koljena ili razreda među različitim uzorcima. Pažljivim odabirom sve tri ulazne varijable dobivaju se rezultati od interesa u tabličnom obliku, koji se onda mogu koristiti za konstrukciju prikladnog grafičkog prikaza.

Kako bi se osiguralo da ne dođe do greške ukoliko u bazi podataka već postoji postupak s istim imenom, prvo se daje naredba za njegovo brisanje:

```
DROP PROCEDURE IF EXISTS sample_top_richness;
```

Zatim se kreira rutina uz definiranje ulaznih varijabli, pri čemu su sve definirane kao INT (eng. *integer*) tip podataka, odnosno kao cijeli broj koji može imati negativne ili pozitivne vrijednosti, ili biti nula. Varijabla `input_sample` određuje za koji se uzorak provodi analiza, varijabla `tax_rank` za koju taksonomsku razinu, a varijabla `top` je limit broja redaka koji će biti prikazani kao rezultat analize:

```
DELIMITER //
CREATE PROCEDURE sample_top_richness
    (
    IN input_sample INT,
    IN tax_rank INT,
    IN top INT
    )
BEGIN
```

Kako bi se omogućilo dobivanje rezultata za određenu razinu taksonomske klasifikacije, može se odabrati varijabla `tax_rank` u rasponu od 1 do 7, pri čemu 1 odgovara najvišoj razini – kraljevstvu, dok 7 odgovara vrsti. Za svaku razinu postoji poseban slučaj unutar rutine, koji će biti proveden kada je odabran odgovarajući cijeli broj za varijablu `tax_rank`. Za rezultate na razini kraljevstva vrijedi sljedeći slučaj:

```
CASE
WHEN tax_rank = 1 THEN
```

Nakon definiranja početka slučaja slijedi glavna `SELECT` naredba koja odabire koji će se stupci prikazivati u tablici s rezultatima, a koristi izraze koji su definirani i privremeno imenovani u tri zasebna podupita:

```
(SELECT (@row_number:=@row_number + 1) AS '#', total.*FROM
(
SELECT
       taxon_a AS 'kingdom',
       sample_richness AS richness,
       average,
       FORMAT(STD(project_richness),4) AS st_dev
FROM
```

Prvi podupit određuje bogatstvo vrsta pomoću `COUNT` naredbe i grupira rezultate prema imenima taksona, u ovom slučaju kraljevstva. Provodi `INNER JOIN` tablice `taxonomy_abundance` i `tax_name` kako bi u rezultatima mogla biti prikazana imena taksona uz odgovarajuće bogatstvo vrsta. Pomoću `WHERE` uvjeta osigurava se da se određuje bogatstvo vrsta samo za odabrani uzorak, pa je tu korištena varijabla `input_sample`.

```
(SELECT
	COUNT(taxonomy_abundance.rank1_kingdom) AS 'sample_richness',
	tax_name.name AS 'taxon_a'
FROM bio.taxonomy_abundance
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank1_kingdom = tax_name.id)
WHERE sample_id = input_sample AND rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY taxon_a
	) AS A,
```

Drugi podupit određuje srednju vrijednost bogatstva vrsta za sve taksone na razini kraljevstva unutar projekta, tako što određuje ukupno bogatstvo vrsta i dijeli ga s brojem uzoraka:

```
(SELECT
	((COUNT(taxonomy_abundance.rank1_kingdom)) / (SELECT COUNT(DISTINCT sample_id)
		FROM bio.taxonomy_abundance)) AS 'average',
	tax_name.name AS 'taxon_b'
FROM bio.taxonomy_abundance
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank1_kingdom = tax_name.id)
WHERE rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY taxon_b
	) AS B,
```

Treći podupit određuje bogatstvo vrsta i grupira rezultate prvo prema uzorcima, a zatim prema imenima taksona, za cijeli projekt. Pomoću tih podataka u glavnoj `SELECT` naredbi računa se standardna devijacija bogatstva vrsta:

```
(SELECT
	COUNT(taxonomy_abundance.rank1_kingdom) AS 'project_richness',
	tax_name.name AS 'taxon_c',
	taxonomy_abundance.sample_id AS 'sample'
FROM bio.taxonomy_abundance
	INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank1_kingdom = tax_name.id)
WHERE rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY sample, taxon_c
	) AS C
```

Rezultati slučaja grupiraju se prema imenima taksona, te se redci sortiraju silazno prema bogatstvu vrsta. Pomoću korisničke varijable `@row_number` numeriraju se redci u tablici s rezultatima, a pomoću varijable `top` postavlja se limit za prikaz rezultata:

```
WHERE taxon_a=taxon_b AND taxon_a=taxon_c
GROUP BY taxon_a
ORDER BY richness DESC
)
total CROSS JOIN (SELECT @row_number:=0) AS row_num
LIMIT top
);
```

Slijedi još šest slučajeva za preostale razine taksonomske klasifikacije, nakon čega je definiran kraj svih slučajeva, te kraj rutine. Time završava cijeli upit, nakon čega se kao graničnik ponovno postavlja točka i zarez:

```
END CASE;
END //
DELIMITER ;
```

Provođenjem prethodnog upita u bazu podataka pohranjuje se rutina. Kako bi se dobili rezultati, potrebno ju je pozvati uz unos vrijednosti za svaku od ulaznih varijabli. Primjerice, za određivanje bogatstva vrsta uzorka 26 na razini razreda i prikaz 10 najbrojnijih taksona koristi se upit:

```
CALL sample_top_richness (26, 3, 10);
```

3.2.1.2. Bogatstvo vrsta projekta

Pohranjeni postupak nazvan `project_top_richness` opširnija je verzija prethodnog upita. Nema ulaznu varijablu za identifikator uzorka niti sadrži prvi podupit za svaki od slučajeva. Zahvaljujući tome kao rezultat daje srednje vrijednosti i standardne devijacije bogatstva vrsta svih taksona odabrane razine taksonomske klasifikacije, s limitom ograničenim brojem redaka. Tako se primjerice može odrediti koji su, u prosjeku, najzastupljeniji razredi unutar čitavog projekta.

3.2.1.3. Podaci o bogatstvu vrsta za kutijasti dijagram

Pohranjeni postupak nazvan `box_plot_richness` ima dvije ulazne varijable: razinu taksonomske klasifikacije i limit za prikazivanje rezultata. Osmišljen je tako da na razini čitavog projekta prema bogatstvu vrsta svakog pojedinog uzorka pronalazi najmanju i najveću vrijednost, donji i gornji kvartil, te medijan. Vrijednosti su razdijeljene prema taksonima odabrane razine taksonomske klasifikacije, te je na temelju njih moguće konstruirati kutijasti dijagram.

Kreira se rutina i definiraju se dvije ulazne varijable INT tipa. Varijabla `tax_rank` određuje za koju taksonomsku razinu se provodi rutina, a varijabla `top` je limit broja redaka koji će biti prikazani kao rezultat analize:

```
DROP PROCEDURE IF EXISTS box_plot_richness;

DELIMITER //
CREATE PROCEDURE box_plot_richness
    (
    IN tax_rank INT,
    IN top INT
    )
BEGIN
```

Za svaku razinu taksonomske klasifikacije postoji poseban slučaj unutar rutine, koji će biti proveden kada je odabran odgovarajući cijeli broj za varijablu `tax_rank`. Za rezultate na razini razreda vrijedi sljedeći slučaj:

```
CASE
WHEN tax_rank = 3 THEN
```

Unutar slučaja dodan je dio izraza potreban za numeriranje redaka u tablici s rezultatima:

```
(SELECT (@rn:=@rn + 1) AS '#', tot.*FROM (
```

Zatim je definirano nekoliko CTE (eng. *Common Table Expression*), privremenih setova rezultata koji se mogu koristiti u daljnjim izrazima. Razlika između korištenja CTE i podupita je to što se CTE mogu koristiti rekurzivno, i više puta, unutar izraza koji slijede nakon što je CTE definiran. Podupiti se mogu koristiti samo unutar glavne `SELECT` naredbe, te se ne može nigdje drugdje pozivati na njihove rezultate. CTE se definira jednom i lako može zamijeniti nekoliko istih podupita koje bi trebalo ugnijezditi na različitim mjestima unutar upita kako bi se dobili isti rezultati.

Prvi CTE unutar slučaja nazvan je `raw_data` i u njemu su definirani svi podaci iz baze koji su potrebni za daljnji račun:

```
WITH
raw_data AS
	(SELECT
		tax_name.name AS 'taxon',
		COUNT(taxonomy_abundance.rank1_kingdom) AS 'richness',
		taxonomy_abundance.sample_id AS 'sample'
	FROM bio.taxonomy_abundance
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank3_class = tax_name.id)
	WHERE rank7_species IS NOT NULL AND rank8_strain IS NULL
	GROUP BY sample, taxon
		),
```

Zatim je definiran drugi CTE u kojem je provedeno razdjeljivanje podataka prema taksonima:

```
partitioned_data AS
	(SELECT
		taxon,
		richness,
		ROW_NUMBER() OVER (PARTITION BY taxon ORDER BY richness) AS row_num,
		SUM(1) OVER (PARTITION BY taxon) AS total
	FROM raw_data
		),
```

U trećem CTE pronađene su vrijednosti potrebne za izračunavanje donjeg i gornjeg kvartila, te medijana:

```
quartiles AS
	(SELECT taxon, richness,
		AVG(CASE WHEN row_num >= (FLOOR(total/2.0)/2.0)
			AND row_num <= (FLOOR(total/2.0)/2.0) + 1
			THEN richness/1.0 ELSE NULL END
				) OVER (PARTITION BY taxon) AS 'q1',
		AVG(CASE WHEN row_num >= (total/2.0)
			AND row_num <= (total/2.0) + 1
			THEN richness/1.0 ELSE NULL END
				) OVER (PARTITION BY taxon) AS 'med',
		AVG(CASE WHEN row_num >= (CEIL(total/2.0) + (FLOOR(total/2.0)/2.0))
			AND row_num <= (CEIL(total/2.0) + (FLOOR(total/2.0)/2.0) + 1)
			THEN richness/1.0 ELSE NULL END
				) OVER (PARTITION BY taxon) AS 'q3'
	FROM partitioned_data
		)
```

Nakon definiranja svih privremenih setova podataka slijedi glavna `SELECT` naredba kojom se uzimaju podaci iz `quartiles` privremenog seta i s njima računaju minimum, maksimum, kvartili te medijan. Donji kvartil je onaj od kojeg je 25% podataka manje ili jednako, dok je gornji kvartil onaj od kojega je 75% podataka manje ili jednako, pa se računaju srednje vrijednosti prethodno dobivenih podataka. Za pronalaženje minimuma i maksimuma postoje SQL funkcije `MIN i MAX`:

```
SELECT
      taxon AS 'class',
      MIN(richness) AS 'minimum',
      ROUND(AVG(q1)) AS 'q1',
      ROUND(AVG(med)) AS 'med',
      ROUND(AVG(q3)) AS 'q3',
      MAX(richness) AS 'maximum'
FROM quartiles
```

Rezultati slučaja grupiraju se prema imenima taksona, te se redci sortiraju silazno prema vrijednosti medijana. Redci u tablici s rezultatima se numeriraju, a pomoću varijable `top` postavlja se limit za prikaz rezultata:

```
GROUP BY 1
ORDER BY 4 ASC
)
tot CROSS JOIN (SELECT @rn:=0) AS rn
LIMIT top);
```

Unutar rutine postoji još šest slučajeva za preostale razine taksonomske klasifikacije, nakon čega je definiran kraj svih slučajeva, te kraj rutine:

```
END CASE;
END //
DELIMITER ;
```

Provođenjem prethodnog upita u bazu podataka pohranjuje se rutina. Kako bi se dobili rezultati, potrebno ju je pozvati uz unos vrijednosti za svaku od ulaznih varijabli. Primjerice, za određivanje podataka za kutijasti dijagram na razini koljena i prikaz 5 najbogatijih taksona koristi se upit:

```
CALL box_plot_richness (2, 5);
```

3.2.2. Relativna brojnost vrsta

Bogatstvo vrsta je mjera koja ne uzima u obzir brojnost vrsta (eng. *species abundance*) unutar uzorka, pa uvijek postoji mogućnost da više uzoraka ima jednako bogatstvo vrsta, čak i jednake vrste. Ako se mjeri brojnost vrsta, rezultati mogu biti sasvim drugačiji. U većini slučajeva su neke vrste dominantnije, odnosno brojčano zastupljenije nego druge, što je odmah vidljivo prema brojnosti vrsta. Relativna brojnost vrsta (eng. *relative species abundance*) dobiva se dijeljenjem brojnosti svake pojedine vrste u uzorku s ukupnom brojnosti svih vrsta u uzorku. Vrijednosti brojnosti unesene u bazu podataka `bio` već su normalizirane i predstavljaju relativne brojnosti, pa se na temelju njih mogu provoditi analize najzastupljenijih vrsta unutar uzorka ili projekta.

3.2.2.1. Brojnost taksona odabranog uzorka

Pohranjeni postupak nazvan `sample_top_abundance` vrlo je sličan prethodno opisanom postupku `sample_top_richness`. Ima jednako definirane tri ulazne varijable, te se sastoji od ukupno sedam slučajeva kako bi se analiza mogla provoditi na odabranoj razini taksonomske klasifikacije. Međutim, glavna razlika je to što kao podatke za račun izravno koristi relativnu brojnost pojedinih taksona iz tablice `taxonomy_abundance`.

Kreira se rutina uz definiranje ulaznih varijabli. Varijabla `input_sample` određuje za koji se uzorak provodi analiza, varijabla `tax_rank` za koju taksonomsku razinu, a varijabla `top` je limit broja redaka koji će biti prikazani kao rezultat analize:

```
DROP PROCEDURE IF EXISTS sample_top_abundance;

DELIMITER //
CREATE PROCEDURE sample_top_abundance
    (
    IN input_sample INT,
    IN tax_rank INT,
    IN top INT
    )
BEGIN
```

Za rezultate na razini koljena vrijedi sljedeći slučaj:

```
CASE
WHEN tax_rank = 2 THEN
```

Unutar slučaja slijedi glavna `SELECT` naredba koja odabire koji će se stupci prikazivati u tablici s rezultatima, a koristi izraze koji su definirani i imenovani u tri zasebna podupita:

```
(SELECT (@row_number:=@row_number + 1) AS '#', total.*FROM
(
SELECT
	taxon_a AS 'phylum',
	tax_abundance,
	FORMAT(project_average,6) AS average,
	FORMAT(STD(project_abundance),4) AS st_dev
FROM
```

Prvi podupit u bazi podataka pronalazi brojnosti pojedinih taksona odabrane razine taksonomske klasifikacije, u ovom slučaju koljena. Provodi `INNER JOIN` tablice `taxonomy_abundance` i `tax_name` kako bi u rezultatima brojnosti mogle biti grupirane prema imenima pojedinih taksona. Pomoću `WHERE` uvjeta osigurava se da se pronalaze podaci o brojnosti samo za odabrani uzorak i samo za određenu razinu taksonomske klasifikacije:

```
(SELECT
	taxonomy_abundance.abundance AS 'tax_abundance',
	tax_name.name AS 'taxon_a'
FROM bio.taxonomy_abundance
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank2_phylum = tax_name.id)
WHERE sample_id = input_sample
	AND rank3_class IS NULL AND rank4_order IS NULL AND rank5_family IS NULL
	AND rank6_genus IS NULL AND rank7_species IS NULL AND rank8_strain IS NULL
GROUP BY taxon_a
	) AS A,
```

Drugi upit određuje srednju vrijednost brojnosti vrsta za sve taksone na razini koljena unutar projekta, tako što određuje zbroj brojnosti svih taksona i dijeli ga s brojem uzoraka:

```
(SELECT
	(SUM(taxonomy_abundance.abundance)
		/ (SELECT COUNT(DISTINCT sample_id) FROM bio.taxonomy_abundance))
		AS 'project_average',
	tax_name.name AS 'taxon_b'
FROM bio.taxonomy_abundance
	INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank2_phylum = tax_name.id)
WHERE rank3_class IS NULL AND rank4_order IS NULL AND rank5_family IS NULL
	AND rank6_genus IS NULL AND rank7_species IS NULL AND rank8_strain IS NULL
GROUP BY taxon_b
	) AS B,
```

Treći podupit pronalazi brojnosti pojedinih taksona i grupira rezultate prvo prema uzorcima, a zatim prema imenima taksona, za cijeli projekt. Pomoću tih podataka u glavnoj `SELECT` naredbi računa se standardna devijacija brojnosti:

```
(SELECT
       taxonomy_abundance.abundance AS 'project_abundance',
       tax_name.name AS 'taxon_c',
       taxonomy_abundance.sample_id AS 'sample'
FROM bio.taxonomy_abundance
       INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
       INNER JOIN bio.tax_name ON (taxonomy_abundance.rank2_phylum = tax_name.id)
WHERE rank3_class IS NULL AND rank4_order IS NULL AND rank5_family IS NULL
       AND rank6_genus IS NULL AND rank7_species IS NULL AND rank8_strain IS NULL
GROUP BY sample, taxon_c
       ) AS C
```

Rezultati slučaja grupiraju se prema imenima taksona, te se redci numeriraju i sortiraju silazno prema brojnosti taksona:

```
WHERE taxon_a=taxon_b AND taxon_a=taxon_c
GROUP BY taxon_a
ORDER BY tax_abundance DESC
)
total CROSS JOIN (SELECT @row_number:=0) AS row_num
LIMIT top
);
```

Unutar rutine postoji još šest slučajeva za preostale razine taksonomske klasifikacije, nakon čega je definiran kraj svih slučajeva i kraj rutine:

```
END CASE;
END //
DELIMITER ;
```

Provođenjem prethodnog upita u bazu podataka pohranjuje se rutina. Kako bi se dobili rezultati potrebno ju je pozvati uz unos vrijednosti za svaku od ulaznih varijabli. Primjerice, za određivanje brojnosti vrsta uzorka 154 na razini koljena i prikaz 10 najbrojnijih taksona koristi se upit:

```
CALL sample_top_abundance(154, 2, 10);
```

3.2.2.2. Brojnost taksona projekta

Pohranjeni postupak nazvan `project_top_abundance` opširnija je verzija prethodnog upita. Nema ulaznu varijablu za identifikator uzorka niti sadrži prvi podupit za svaki od slučajeva. Kao rezultat daje srednje vrijednosti i standardne devijacije relativne brojnosti svih taksona odabrane razine taksonomske klasifikacije. Pomoću takve rutine može se odrediti koje su primjerice najzastupljenije obitelji unutar čitavog projekta.

3.2.2.3. Podaci o brojnosti taksona za kutijasti dijagram

Pohranjeni postupak nazvan `box_plot_abundance` ima dvije ulazne varijable: razinu taksonomske klasifikacije i limit za prikazivanje rezultata. Služi za pronalaženje minimuma, donjeg kvartila, medijana, gornjeg kvartila, i maksimuma, pomoću kojih se iz podataka o relativnoj brojnosti može konstruirati kutijasti dijagram na razini čitavog projekta.

Upit za izradu ove rutine gotovo je identičan `box_plot_richness` upitu, s tri CTE i jednakim načinom određivanja svih konačnih vrijednosti. Razlika je u prvom CTE, gdje se kao sirovi podaci koristi relativna brojnost pojedinih taksona odabrane razine taksonomske klasifikacije, umjesto bogatstva vrsta. Uz to je drugačije definiran `WHERE` uvjet, pa za rezultate na razini razreda vrijedi ovakav prvi CTE:

```
WITH
raw_data AS
	(SELECT
		tax_name.name AS 'taxon',
		taxonomy_abundance.abundance AS 'tax_ab'
	FROM bio.taxonomy_abundance
	INNER JOIN bio.tax_name ON (taxonomy_abundance.rank3_class = tax_name.id)
	WHERE rank4_order IS NULL AND rank5_family IS NULL
	AND rank6_genus IS NULL AND rank7_species IS NULL AND rank8_strain IS NULL
		),
```

Ostatak slučaja jednak je `box_plot_richness` upitu, te se sve ponavlja kroz još 6 slučajeva. Provođenjem upita u bazu podataka pohranjuje se rutina. Kako bi se dobili rezultati potrebno ju je pozvati uz unos vrijednosti za svaku od ulaznih varijabli. Primjerice, za određivanje podataka za kutijasti dijagram na razini razreda i prikaz 5 najbrojnijih taksona koristi se upit:

```
CALL box_plot_abundance (3, 5);
```

3.2.3. Shannonov indeks

Shannonov indeks raznolikosti ili Shannon-Weaver indeks je statistička mjera kojom se u ekologiji uobičajeno karakterizira raznolikost vrsta u zajednici. U obzir uzima relativnu brojnost i podjednakost raspodjele vrsta. Računa se prema formuli:

$$H = -\sum_{i=1}^{S}(p_i \cdot \ln p_i) \quad [1]$$

gdje je: H – Shannonov indeks, p_i – brojnost vrste i podijeljena s ukupnim brojem vrsta u uzorku, S – bogatstvo vrsta u uzorku.

Shannonova podjednakost (eng. *evenness*) je mjera raspodjele vrsta unutar uzorka s obzirom na njihovu relativnu brojnost. Može poprimiti vrijednosti od 0 do 1, pri čemu je 1 potpuna podjednakost raspodjele i podrazumijeva da sve vrste unutar uzorka imaju jednaku brojnost, odnosno da su u podjednakoj mjeri zastupljene u uzorku (Veech, 2018; Whittaker, 1972).

$$E_H = \frac{H}{H_{max}} = \frac{H}{\ln S} \quad [2]$$

gdje je: E_H – Shannonov indeks podjednakosti, H – Shannonov indeks, S – bogatstvo vrsta u uzorku.

Pohranjeni postupak nazvan `shannon_index_comparison` ima dvije ulazne varijable. Obje su identifikatori uzorka, tako da se odabirom manjeg i većeg identifikatora određuje raspon uzoraka za koji se određuju Shannonov indeks i Shannonova podjednakost. Rutina dodatno izračunava srednje vrijednosti i standardne devijacije Shannonovog indeksa i podjednakosti na razini čitavog projekta. Dobivaju se rezultati koji se mogu koristiti za usporedbu bogatstva vrsta pojedinih uzoraka sa Shannonovim indeksom i podjednakosti.

Upit započinje kreiranjem rutine pri čemu se definiraju dvije ulazne varijable INT tipa, `input_sample_1` i `input_sample_2`:

```
DROP PROCEDURE IF EXISTS shannon_index_comparison;

DELIMITER //
CREATE PROCEDURE shannon_index_comparison
	(
	IN input_sample_1 INT,
	IN input_sample_2 INT
	)
BEGIN
```

Nakon definiranja početka rutine slijedi glavna `SELECT` naredba, koja odabire koji će se stupci iz prvog podupita prizivati u tablici s rezultatima. Unutar te naredbe također se izračunavaju srednje vrijednosti i standardne devijacije za Shannonov indeks i Shannonovu podjednakost, koristeći podatke iz drugog podupita:

```
SELECT
	sample,
	richness,
	shannon_index,
	shannon_evenness,
	FORMAT(AVG(project_shannon_index),4) AS 'average_shannon_index',
	FORMAT(STD(project_shannon_index),4) AS 'index_st_dev',
	FORMAT(AVG(project_shannon_evenness),4) AS 'average_evenness',
	FORMAT(STD(project_shannon_evenness),4) AS 'evenness_st_dev'
FROM
```

Prvi podupit određuje bogatstvo vrsta, te Shannonov indeks i Shannonovu podjednakost za odabran raspon uzoraka, što je određeno pomoću `WHERE` uvjeta:

```
(SELECT
	sample_id AS 'sample',
	COUNT(abundance) AS 'richness',
	FORMAT(-SUM(abundance * LN(abundance)),4) AS 'shannon_index',
	FORMAT(((-SUM(abundance * LN(abundance))) / LN(COUNT(abundance))),4)
		AS 'shannon_evenness'
FROM bio.taxonomy_abundance
	INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
WHERE sample_id >= input_sample_1 AND sample_id <= input_sample_2
	AND rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY sample
	) AS A,
```

Drugi podupit određuje Shannonov indeks i Shannonovu podjednakost za sve uzorke unutar projekta, na temelju relativnih brojnosti vrsta grupiranih prema uzorcima:

```
(SELECT
	FORMAT(-SUM(abundance * LN(abundance)),4) AS 'project_shannon_index',
	FORMAT(((-SUM(abundance * LN(abundance))) / LN(COUNT(abundance))),4)
		AS 'project_shannon_evenness'
FROM bio.taxonomy_abundance
	INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
WHERE rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY sample_id
	) AS B
```

Konačno se rezultati glavnog upita grupiraju prema uzorcima, nakon čega završava cijeli upit i pohranjuje se rutina:

```
GROUP BY sample;
END //
DELIMITER ;
```

Rutina se poziva uz unos vrijednosti za ulazne varijable. Primjerice, za određivanje Shannonovog indeksa i Shannonove podjednakosti za raspon od uzorka 7 do uzorka 27 koristi se upit:

```
CALL shannon_index_comparison (7, 27);
```

3.2.4. Inverzni Simpsonov indeks

Inverzni Simpsonov indeks raznolikosti je statistička mjera kojom se u ekologiji uobičajeno karakterizira dominantnost vrsta, odnosno u kojoj je mjeri neka vrsta brojčano dominantna u uzorku na temelju relativne brojnosti. Računa se prema formuli:

$$D = \frac{1}{\sum_{i=1}^{S} p_i^2} \quad [3]$$

gdje je: D – Inverzni Simpsonov indeks, p_i – brojnost vrste i podijeljena s ukupnim brojem vrsta u uzorku, S – bogatstvo vrsta u uzorku.

Simpsonova nepristranost (eng. *equitability*) je mjera raspodjele vrsta unutar uzorka. Izračunava se kao razmjer maksimalne vrijednosti koju Simpsonov indeks može poprimiti kad bi sve vrste u uzorku bile podjednako raspoređene, što je jednako bogatstvu vrsta ukoliko je prisutan po jedan predstavnik svake vrste. Može imati vrijednosti od 0 do 1, pri čemu je 1 potpuna podjednakost raspodjele (Veech, 2018; Whittaker, 1972).

$$E_D = \frac{D}{D_{max}} = D \cdot \frac{1}{S} \quad [4]$$

gdje je: E_D – Simpsonova nepristranost, D – inverzni Simpsonov indeks, S – bogatstvo vrsta u uzorku.

Pohranjeni postupak nazvan `simpson_index_comparison` ima dvije ulazne varijable. Obje su identifikatori uzorka, tako da se odabirom manjeg i većeg identifikatora određuje raspon uzoraka za koji se određuju inverzni Simpsonov indeks i Simpsonova nepristranost, uz srednje vrijednosti i standardne devijacije na razini čitavog projekta.

Postupak započinje vrlo slično kao i prethodni, a razlikuju se prema podupitima koji vrše sam izračun. Prvi podupit određuje bogatstvo vrsta, te Simpsonov indeks i Simpsonovu nepristranost za odabran raspon uzoraka, što je određeno pomoću `WHERE` uvjeta:

```
(SELECT
       sample_id AS 'sample',
       COUNT(abundance) AS 'richness',
       FORMAT((1 / SUM(SQRT(abundance))),4) AS 'simpson_index',
       FORMAT(((1 / SUM(SQRT(abundance))) / COUNT(abundance)),4)
              AS 'simpson_equitability'
FROM bio.taxonomy_abundance
       INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
WHERE sample_id >= input_sample_1 AND sample_id <= input_sample_2
       AND rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY sample
       ) AS A,
```

Drugi podupit određuje inverzni Simpsonov indeks i Simpsonovu nepristranost za sve uzorke unutar projekta, na temelju relativnih brojnosti vrsta grupiranih prema uzorcima:

```
(SELECT
       FORMAT((1 / SUM(SQRT(abundance))),4) AS 'project_simpson_index',
       FORMAT(((1 / SUM(SQRT(abundance))) / COUNT(abundance)),4)
              AS 'project_simpson_equitability'
FROM bio.taxonomy_abundance
       INNER JOIN bio.sample ON (taxonomy_abundance.sample_id = sample.id)
WHERE rank7_species IS NOT NULL AND rank8_strain IS NULL
GROUP BY sample_id
       ) AS B
```

Konačno se rezultati glavnog upita grupiraju prema uzorcima, nakon čega završava cijeli upit i pohranjuje se rutina:

```
GROUP BY sample;
END //
DELIMITER ;
```

Rutina se poziva uz unos vrijednosti za ulazne varijable. Primjerice, za raspon od uzorka 711 do uzorka 740 koristi se upit:

```
CALL simpson_index_comparison (711, 740);
```

3.2.5. Jaccardov koeficijent sličnosti

Jaccardov koeficijent ili indeks sličnosti je statistička mjera za određivanje beta raznolikosti. Uspoređuje vrste prisutne u dva uzorka, kako bi se ustvrdilo koje vrste su zajedničke tim uzorcima, a koje su jedinstvene za svaki uzorak. Može poprimiti vrijednosti od 0 do 1, pri čemu veća vrijednost znači da su dva uzorka sličnija prema sadržaju vrsta. Računa se prema formuli:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A|+|B|-|A \cap B|} \quad [5]$$

gdje je: J – Jaccardov koeficijent sličnosti, $|A \cap B|$ – broj vrsta zajedničkih za oba uzorka, $|A \cup B|$ – ukupan broj jedinstvenih vrsta unutar oba uzorka.

Oduzimanjem Jaccardovog koeficijenta sličnosti od 1 dobiva se suprotna vrijednost, Jaccardova udaljenost. Što ima veću vrijednost, to su dva uzorka udaljenija, odnosno različitija prema sadržaju vrsta (Whittaker, 1972).

Pohranjeni postupak nazvan `jaccard_index` ima dvije ulazne varijable za identifikatore uzorka, pomoću kojih se bira za koja dva uzorka će se određivati Jaccardov koeficijent sličnosti i Jaccardova udaljenost. Upit započinje kreiranjem rutine uz definiranje ulaznih varijabli `input_sample_1` i `input_sample_2`:

```
DELIMITER //
CREATE PROCEDURE jaccard_index
	(
	IN input_sample_1 INT,
	IN input_sample_2 INT
	)
BEGIN
```

Slijedi glavna `SELECT` naredba kojom se iz podataka dobivenih prvim i drugim podupitom izračunavaju Jaccardov koeficijent i udaljenost. Naredba `CONCAT` omogućuje povezivanje više nizova u jedan niz (eng. *string*) kako bi se prikazalo za koja dva uzorka je proveden račun:

```
SELECT
	CONCAT(input_sample_1, ' and ', input_sample_2) AS 'samples',
	shared AS species_in_common,
	(total_species - shared) AS 'all_species',
	(shared / (total_species - shared)) AS 'jaccard_index',
	1-(shared / (total_species - shared)) AS 'jaccard_distance'
FROM
```

Prvi podupit je ključan za određivanje Jaccardovog koeficijenta, jer iz podataka o bogatstvu vrsta mora pronaći vrste zajedničke za dva odabrana uzorka. S obzirom da bi uobičajena `SELECT` naredba uz `WHERE` uvjet gdje su navedena oba uzorka zapravo pronalazila samo bogatstvo vrsta jednog ili drugog uzorka, a ne broj zajedničkih vrsta, potrebno je izvesti takozvani `'SELF' JOIN`. Koristi se uobičajena `INNER JOIN` naredba, ali se tablica pridružuje sama sebi tako što joj se daju dva aliasa, u ovom slučaju `a` i `b`. Time se omogućuje pronalaženje samo onih vrsta koje pripadaju tablici `a` i `b`, odnosno i prvom i drugom uzorku:

```
(SELECT COUNT(a.rank7_species) AS 'shared'
FROM bio.taxonomy_abundance AS a
      INNER JOIN bio.taxonomy_abundance AS b ON a.rank7_species = b.rank7_species
WHERE a.sample_id = input_sample_1 AND a.rank8_strain IS NULL
      AND b.sample_id = input_sample_2 AND b.rank8_strain IS NULL
      ) AS C,
```

U drugom podupitu računa se zbroj bogatstva vrsta oba uzorka. Kako bi se dobio ukupan broj jedinstvenih vrsta u oba uzorka, u glavnoj `SELECT` naredbi od tog se zbroja oduzima prethodno izračunat broj vrsta zajedničkih za oba uzorka:

```
(SELECT COUNT(rank7_species) AS 'total_species'
FROM bio.taxonomy_abundance
WHERE sample_id = input_sample_1 AND rank8_strain IS NULL
      OR sample_id = input_sample_2 AND rank8_strain IS NULL
      ) AS D;
```

Nakon drugog podupita završava cijela rutina. S obzirom da uvijek daje samo jedan redak u tablici s rezultatima, nije potrebno sortiranje ni numeriranje rezultata:

```
END //
DELIMITER ;
```

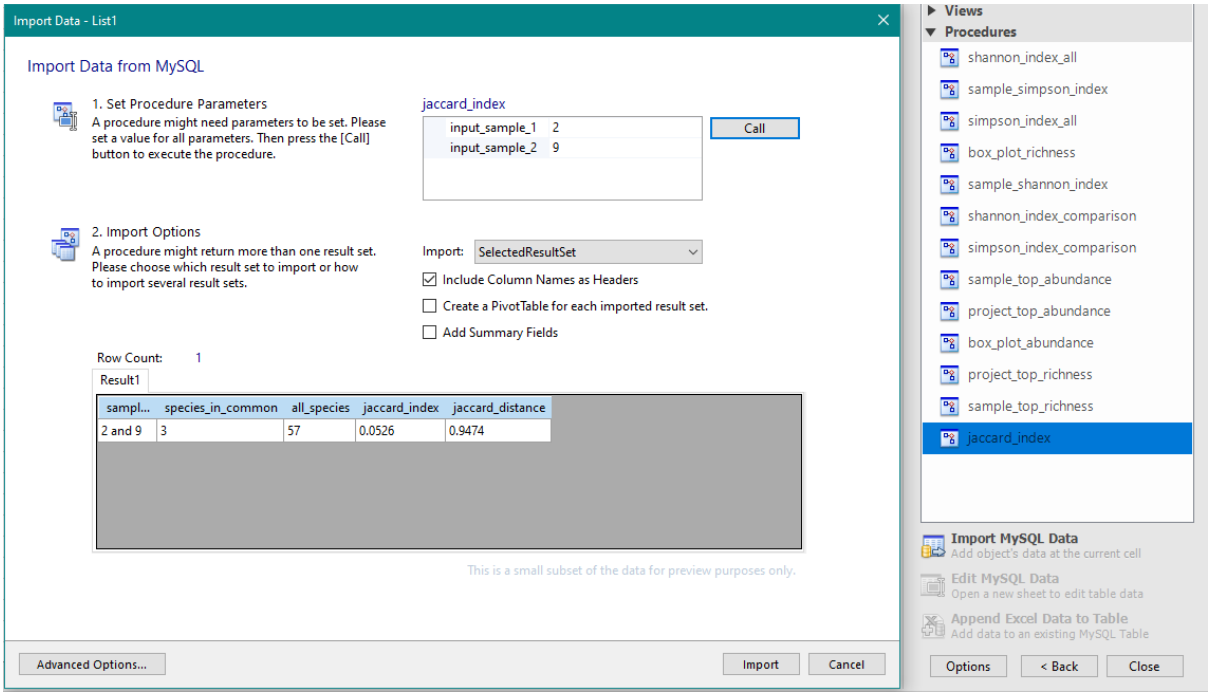
Rutina se poziva uz unos vrijednosti za ulazne varijable. Primjerice, za izračunavanje Jaccardovog koeficijenta sličnosti, odnosno Jaccardove udaljenosti uzoraka 2 i 9 koristi se upit:

```
CALL jaccard_index (2, 9);
```

Ovim pohranjenim postupkom moguće je usporediti bilo koja dva uzorka od svih 1639 uzoraka koji se nalaze unutar projekta.

3.2.6. Grafički prikaz rezultata

S obzirom da su svi upiti za analize sastavljeni u obliku pohranjenih postupaka, kroz korisničko sučelje programa Excel moguće ih je pozvati zahvaljujući dodatku MySQL for Excel. Za stvaranje veze s MySQL bazom podataka koristi se kartica 'Podaci' gdje se nalazi gumb za pokretanje MySQL for Excel. Unutar korisničkog sučelja pokreće se dodatan prozorčić MySQL for Excel, putem kojega se stvara veza s postojećom bazom podataka `bio` korištenjem istih podataka za pristup kao i u MySQL Workbench i SQLyog alatima. Nakon uspostavljanja veze, moguće je unutar prozorčića MySQL for Excel odabrati shemu baze podataka za koju se provode analize. Zatim se odabire željena rutina za provođenje određene analize na tim podacima (Slika 8). Unosom ulaznih varijabli dobiva se tablica s rezultatima direktno unutar odabranog radnog lista.



Slika 8. Primjer pozivanja pohranjenog postupka `jaccard_index` uz unos potrebnih ulaznih varijabli putem MySQL for Excel dodatka u Excel-u. S desne strane vidi se dio prozorčića s uspostavljenom vezom s bazom podataka `bio`, dok je s lijeve strane aktivan prozorčić za unos rezultata rutine u Excel.

Pripremom odgovarajućeg grafičkog prikaza za pojedine setove rezultata omogućen je određen stupanj automatizacije prikaza. Tako je za svaku analizu zasebno pripremljen radni list za grafički prikaz rezultata, te se izmjenom ulaznih varijabli rutine svaki put dobiva odgovarajući grafički prikaz novog seta rezultata.

4. REZULTATI I RASPRAVA

S obzirom da je glavni cilj ovog rada izvedba SQL upita za analizu podataka o sastavu crijevne mikrobiote, u rezultatima je prikazano tek nekoliko primjera kako bi se ilustrirala uloga svake analize. Budući da se baza podataka sastoji od 1639 uzoraka i vezanih podataka o brojnosti pojedinih taksona, uz pomoć dodatka MySQL for Excel su grafički ili tablično prikazani rezultati pohranjenih postupaka s nasumično odabranim ulaznim varijablama.

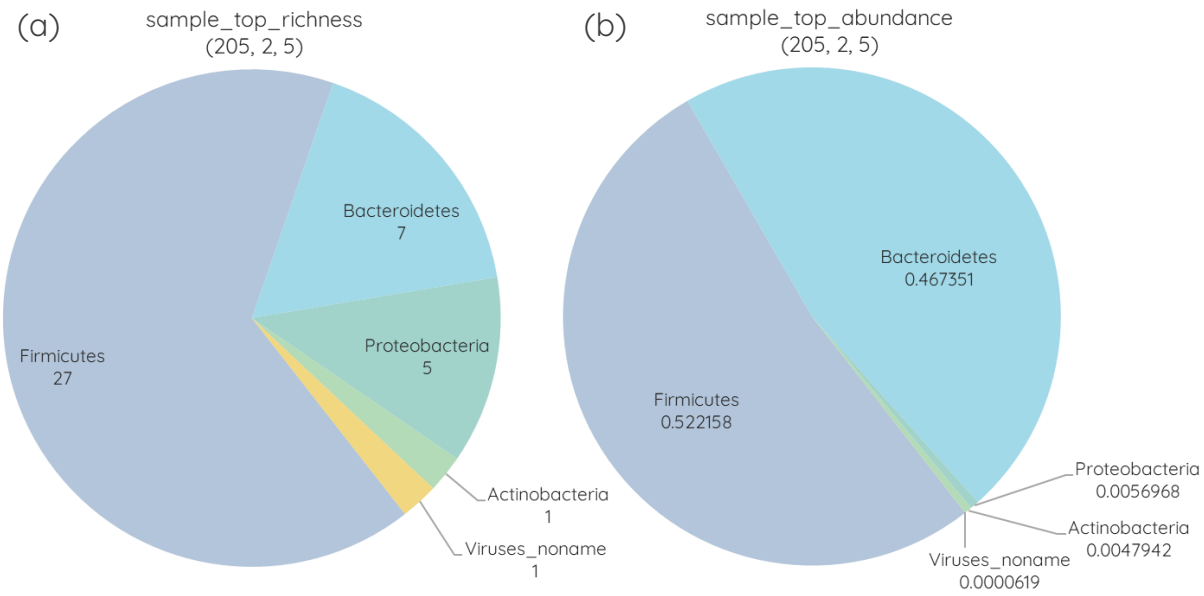
Primjeri su osmišljeni kako bi bili međusobno usporedivi, te su grupirani ovisno o tome provodi li se rutina za pojedini uzorak ili na razini čitavog projekta. Imena taksona u grafičkim prikazima istovjetna su onima unutar baze podataka, zbog čega često sadrže znak podvlačenja umjesto razmaka. Tablični prikazi su istovjetni onima koji nastaju kao rezultat rutina u bazi podataka MySQL, zbog čega su nazivi stupaca također pisani na engleskom jeziku i sa znakovima podvlačenja.

4.1. PRIMJERI REZULTATA PROVEDENIH RUTINA

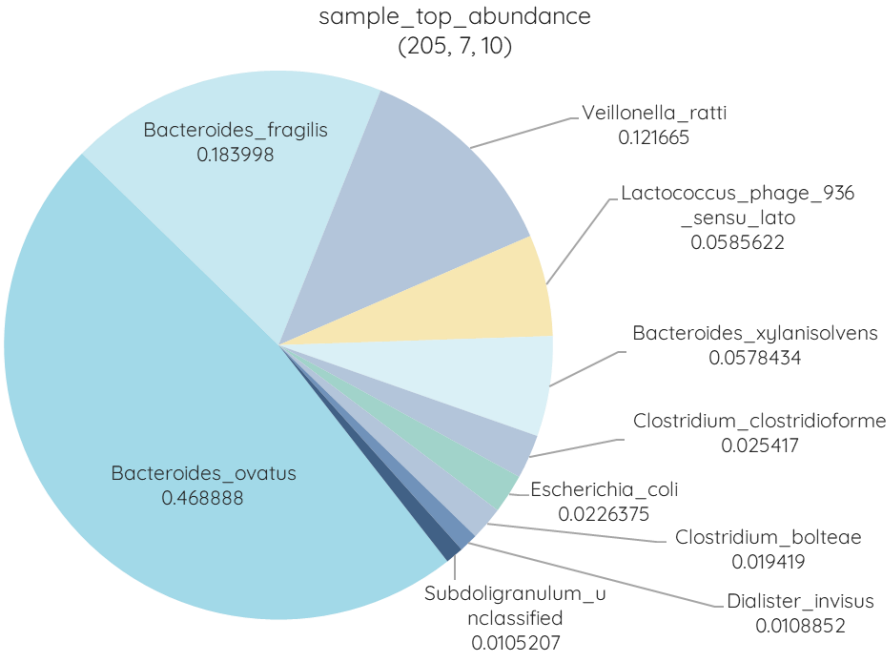
4.1.1. Bogatstvo vrsta i relativna brojnost taksona uzorka

Pozivanjem rutina za određivanje bogatstva i relativne brojnosti vrsta za odabrani uzorak moguće je usporediti raznolikost istog uzorka prema ta dva kriterija, i to na odabranoj taksonomskoj razini. Ove rutine mogu se provesti za bilo koji uzorak u projektu, te se rezultati za pojedine uzorke mogu međusobno uspoređivati. Rezultati svakog uzorka mogu se usporediti i sa srednjim vrijednostima za projekt koje se također određuju u sklopu navedenih rutina.

Kao primjer su za uzorak 205 provedene rutine `sample_top_richness` i `sample_top_abundance`. Iz dobivenih rezultata na razini koljena vidljivo je kako uzorak sadrži 40 različitih vrsta bakterija i jedan virus (Slika 9a), pri čemu je *Firmicutes* koljeno s najviše vrsta, dok su *Bacteroidetes* i *Proteobacteria* značajno manje raznovrsni. Međutim, ukoliko se u obzir uzme i relativna brojnost vrsta (Slika 9b), *Firmicutes* i *Bacteroidetes* su podjednaki i zajedno čine 98,95% brojnosti uzorka. Prema tome, iako *Bacteroidetes* ne sadrži mnogo različitih vrsta, neke od tih vrsta su i dalje brojčano vrlo zastupljene unutar uzorka. To se može potvrditi ponovnim provođenjem rutine `sample_top_abundance`, ali ovaj put na razini vrste (Slika 10). Iz rezultata je vidljivo kako bakterija *Bacteroides ovatus*, koja pripada koljenu *Bacteroidetes*, čini skoro polovicu brojnosti u uzorku.



Slika 9. Grafički prikaz pet najzastupljenijih koljena u uzorku 205 prema bogatstvu vrsta i relativnoj brojnosti. Rezultati su dobiveni provođenjem rutina **(a)** `sample_top_richness` i **(b)** `sample_top_abundance` uz zadane varijable: `205` – uzorak broj 205, `2` – razina taksonomije je koljeno, `5` – prikaz prvih pet rezultata. Izvorni tablični rezultati nalaze se u prilozima 3.1 i 3.2.

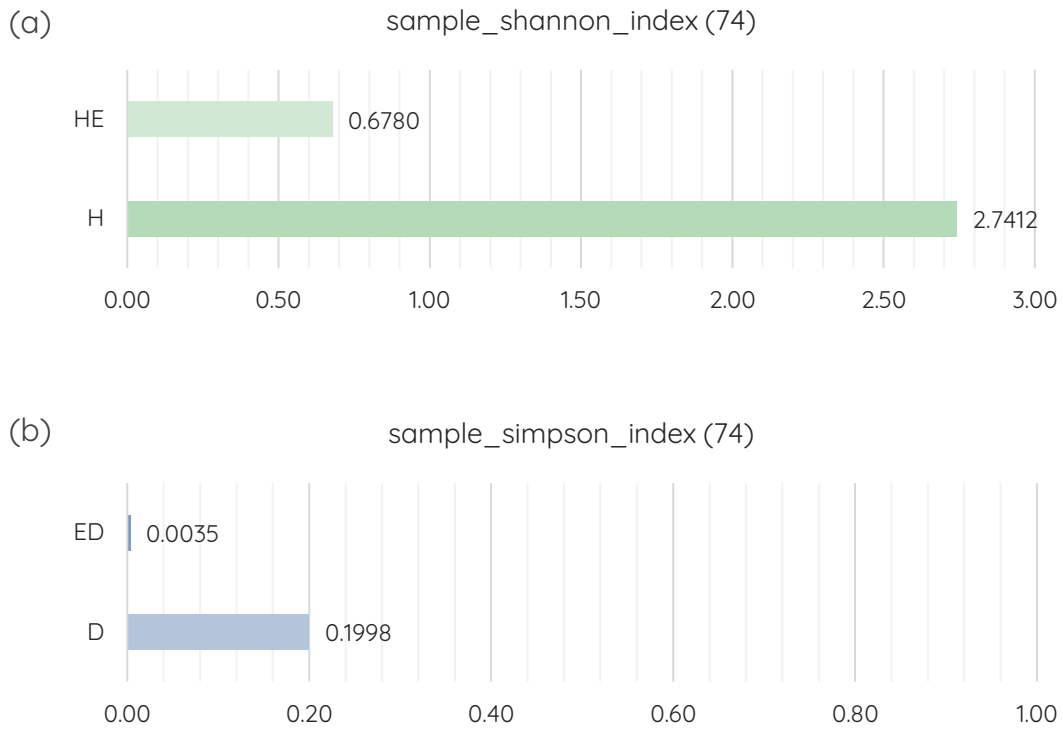


Slika 10. Grafički prikaz deset najzastupljenijih vrsta u uzorku 205 prema relativnoj brojnosti. Rezultati su dobiveni provođenjem rutine `sample_top_abundance` uz zadane varijable: `205` – uzorak broj 205, `7` – razina taksonomije je vrsta, `10` – prikaz prvih 10 rezultata. Izvorni tablični rezultati nalaze se u prilogu 3.3.

4.1.2. Shannonov i inverzni Simpsonov indeks uzorka

Pozivanjem rutina za Shannonov indeks i inverzni Simpsonov indeks moguće je odrediti raznolikost i podjednakost raspodjele vrsta unutar odabranog uzorka. Ove rutine korisne su za analizu ukoliko se žele saznati vrijednosti tih indeksa za točno određen uzorak, a najbolje ih je koristiti uz paralelnu analizu bogatstva i relativne brojnosti vrsta.

Kao primjer su za uzorak broj 74 provedene rutine `sample_shannon_index` i `sample_simpson_index`. Dobiveni rezultati (Slika 11) prikazuju vrijednosti oba indeksa te pripadajuće vrijednosti podjednakosti, odnosno nepristranosti.



Slika 11. Grafički prikaz **(a)** Shannonovog indeksa H i podjednakosti H_E dobivenih provođenjem rutine `sample_shannon_index` te **(b)** inverznog Simpsonovog indeksa D i Simpsonove nepristranosti E_D dobivenih provođenjem rutine `sample_simpson_index`. Za obje rutine korištena je samo varijabla `74` za određivanje uzorka. Izvorni tablični rezultati nalaze se u prilozima 3.4 i 3.5.

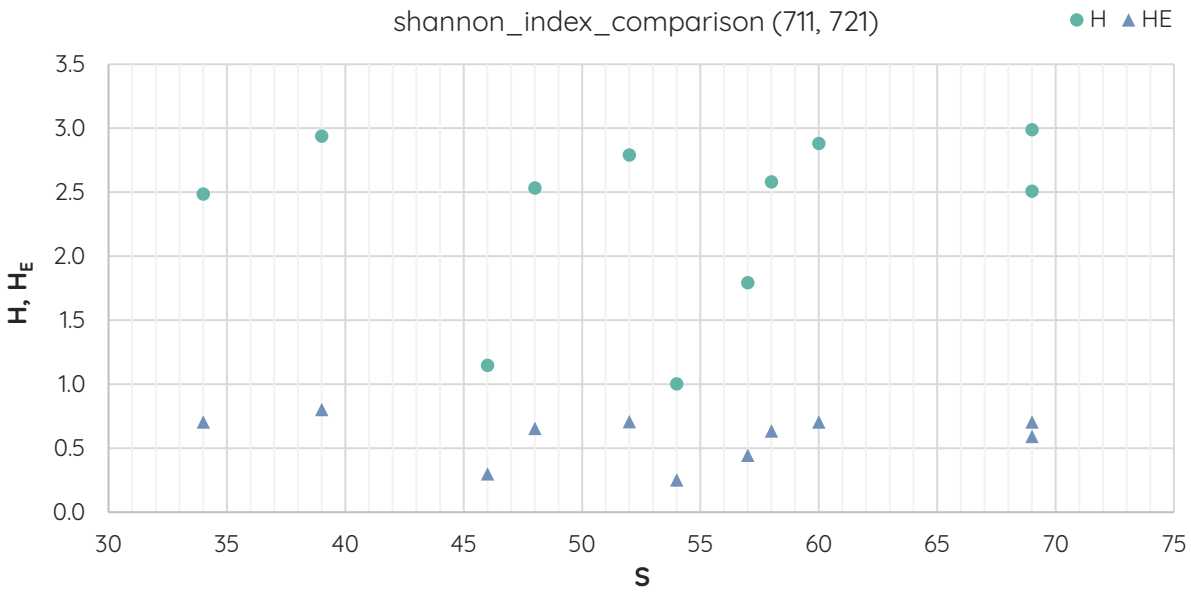
4.1.3. Shannonov i inverzni Simpsonov indeks za raspon uzoraka

Više informacija o raznolikosti sastava crijevne mikrobiote može se dobiti uspoređivanjem indeksa raznolikosti više različitih uzoraka, odnosno više različitih zajednica. Korištenjem rutine `simpson_index_comparison` dobiva se tablica s izračunatim Simpsonovim indeksima i pripadajućim nepristranostima za odabran raspon uzoraka, te se dodatno prikazuju srednje vrijednosti indeksa i nepristranosti za čitav projekt, uz standardne devijacije. Već se u tabličnim rezultatima može uočiti ukoliko neki uzorak odstupa od prosjeka i drugih uzoraka, kao što je primjerice slučaj s uzorcima 711 i 712 (Tablica 1).

Tablica 1. Tablica s rezultatima rutine `simpson_index_comparison` za zadane varijable: `711` – prvi uzorak u rasponu, `721` – zadnji uzorak u rasponu. Imena stupaca jednaka su izvornim imenima u tablici s rezultatima u MySQL-u. Izvorni tablični rezultati nalaze se u prilogu 3.6.

uzorak	S	D	E_D	$\overline{D}$	$\sigma(\overline{D})$	$\overline{E_D}$	$\sigma(\overline{E_D})$
711	46	0,3514	0,0076	0,2370	0,0645	0,0070	0,0373
712	54	0,3252	0,0060				
713	60	0,1906	0,0032				
714	69	0,1773	0,0026				
715	52	0,1967	0,0038				
716	34	0,2318	0,0068				
717	58	0,2164	0,0037				
718	69	0,2019	0,0029				
719	57	0,2571	0,0045				
720	39	0,1988	0,0051				
721	48	0,2188	0,0046				

Korištenjem rutine `shannon_index_comparison` dobiva se slična tablica, koja se može dodatno prikazati grafički kao odnos Shannonovog indeksa i/ili podjednakosti s brojem različitih vrsta u uzorku. Tablični prikaz sadrži više informacija, no na grafičkom prikazu lakše je uočiti ukoliko neki uzorak odstupa od ostatka. Na primjeru s jednakim rasponom kao za prethodnu rutinu vidljivo je da prema Shannonovom indeksu također 2 uzorka odstupaju od ostatka (Slika 12). Usporedbom bogatstva vrsta ta dva uzorka s podacima o bogatstvu vrsta u tablici 2 jasno je da je riječ o uzorcima 711 i 712.



Slika 12. Grafički prikaz odnosa Shannonovog indeksa H i Shannonove podjednakosti H_E s bogatstvom vrsta. Rezultati su dobiveni provođenjem rutine `shannon_index_comparison` uz zadane varijable: `711` – prvi uzorak u rasponu, `721` – zadnji uzorak u rasponu. Izvorni tablični rezultati nalaze se u prilogu 3.7.

4.1.4. Jaccardov koeficijent sličnosti dva uzorka

Rutina za određivanje Jaccardovog koeficijenta postoji u samo jednoj verziji, `jaccard_index`, s obzirom da služi za određivanje sličnosti dva uzorka. Može se provoditi za bilo koje uzorke unutar projekta, pri čemu se svaki put dobivaju tablični rezultati s izračunatim vrijednostima Jaccardovog koeficijenta sličnosti i Jaccardove udaljenosti. Kao primjer je prikazana tablica sličnosti uzoraka 21 i 510 (Tablica 2).

Tablica 2. Tablica s rezultatima rutine `jaccard_index` za zadane varijable: `21` – prvi uzorak za usporedbu, `510` – drugi uzorak za usporedbu. Izvorni tablični rezultati nalaze se u prilogu 3.8.

uzorci	zajedničke vrste	ukupno vrsta	Jaccardov koeficijent sličnosti	Jaccardova udaljenost
21 i 510	19	88	0,2159	0,7841

4.1.5. Zastupljenost taksona u projektu prema bogatstvu vrsta

Osim određivanja bogatstva vrsta odabranih uzoraka, također je moguće odrediti srednje vrijednosti bogatstva vrsta na temelju ukupnih podataka u bazi. Rutina `project_top_richness` računa srednje vrijednosti bogatstva taksona odabrane razine taksonomske klasifikacije, uz pripadajuće standardne devijacije. Budući da se na nižim taksonomskim razinama povećava broj taksona prisutnih u projektu, preporuča se ograničavanje prikaza rezultata samo na one najzastupljenije. Na taj način se za svaku razinu mogu odrediti najzastupljeniji taksoni. Prikazan je primjer tablice sa deset najvećih srednjih vrijednosti bogatstva vrsta $\bar{S}$ na razini koljena (Tablica 3). Prema dobivenim rezultatima, četiri najzastupljenija koljena u projektu su *Firmicutes, Bacteroidetes, Proteobacteria* i *Actinobacteria*, točno tim redoslijedom. Usporedbom s rezultatima bogatstva vrsta na razini koljena na slici 9a vidljivo je da taksonomski profil uzorka 205 odgovara prosječnom taksonomskom profilu projekta, barem na toj razini.

Drugi način prikaza bogatstva vrsta na odabranoj razini taksonomske klasifikacije je određivanje raspodjele vrijednosti unutar seta, na temelju čega se može sastaviti kutijasti dijagram. Stoga je potrebno za svaki set, odnosno takson, odrediti najmanju i najveću vrijednost, te izračunati donji kvartil, medijan, i gornji kvartil. Rutina `box_plot_richness` razdjeljuje podatke u projektu prema taksonima odabrane razine, sortira ih od najmanjeg od najvećeg za svaki set, te izračunava sve potrebne vrijednosti. Kao rezultat se dobivaju podaci za kutijasti dijagram u tabličnom obliku (Tablica 4).

Tablica 3. Tablica s rezultatima rutine `project_top_richness` za zadane varijable: `2` – razina taksonomije je koljeno, `10` – prikaz prvih 10 rezultata. Izvorni tablični rezultati nalaze se u prilogu 3.9.

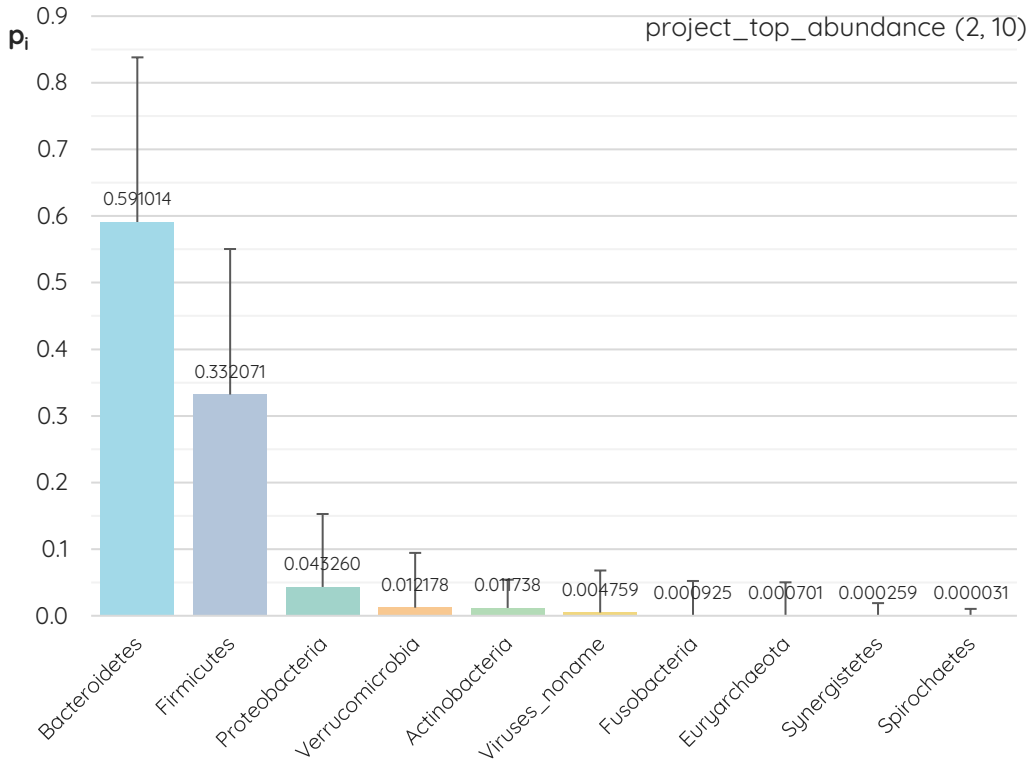
#	koljeno	$\bar{S}$	$\sigma(\bar{S})$
1	Firmicutes	31,6007	9,5084
2	Bacteroidetes	15,8346	7,3892
3	Proteobacteria	4,8565	2,5126
4	Actinobacteria	2,9603	1,7973
5	Viruses_noname	0,4444	0,5162
6	Verrucomicrobia	0,3553	0
7	Euryarchaeota	0,1154	0,4231
8	Fusobacteria	0,0604	0,2244
9	Deinococcus_Thermus	0,0226	0
10	Ascomycota	0,0220	0,4761

Tablica 4. Tablica s rezultatima rutine `box_plot_richness` za zadane varijable: `2` – razina taksonomije je koljeno, `4`– prikaz prva 4 rezultata. Izvorni tablični rezultati nalaze se u Prilogu 3.10.

#	koljeno	minimum	donji kvartil (Q1)	medijan	gornji kvartil (Q3)	maksimum
1	Firmicutes	1	26	32	37	63
2	Bacteroidetes	1	10	16	21	37
3	Proteobacteria	1	3	5	6	17
4	Actinobacteria	1	2	3	4	12

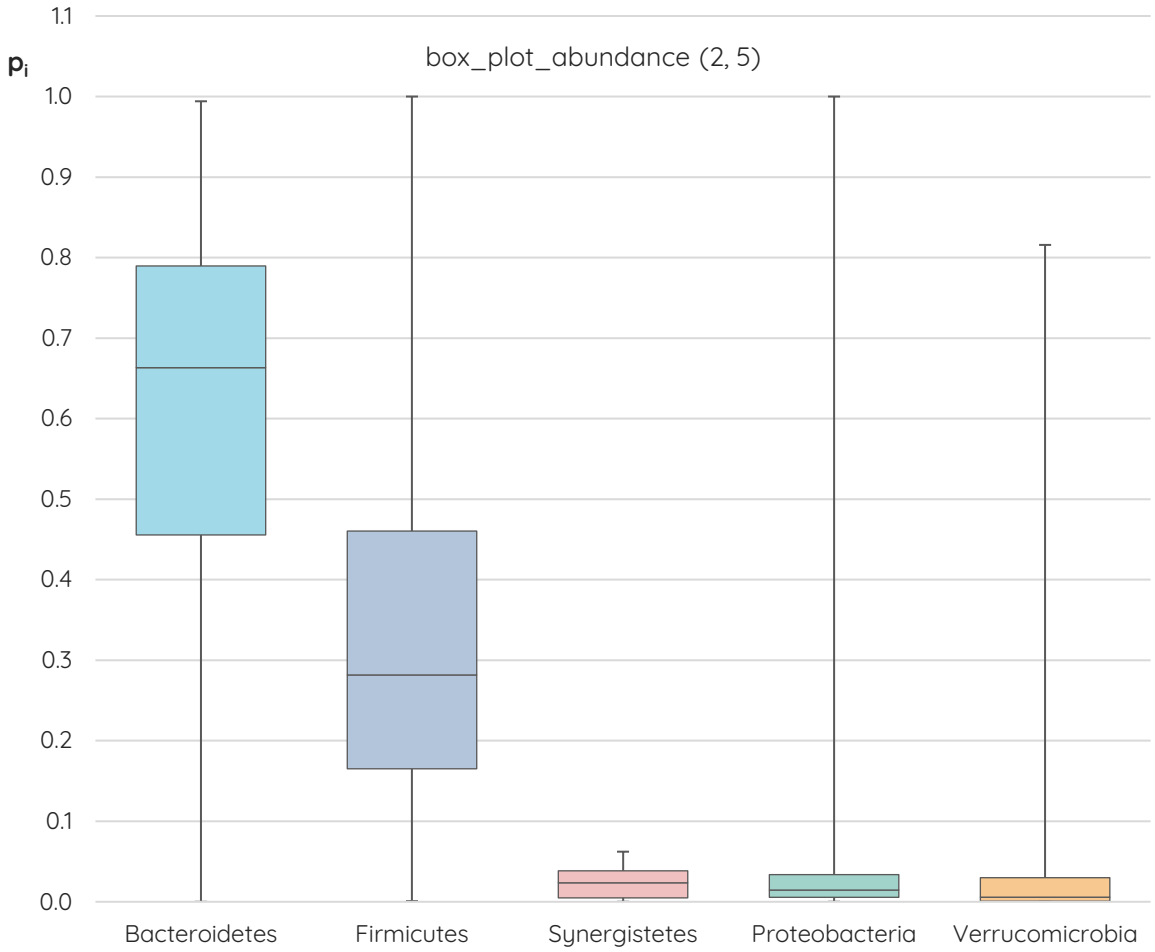
4.1.6. Zastupljenost taksona u projektu prema relativnoj brojnosti

Slično dvjema prethodno navedenim rutinama, `project_top_abundance` koristi sve podatke relativne brojnosti u bazi za izračun srednjih vrijednosti i pripadajućih standardnih devijacija za taksone odabrane razine klasifikacije. Daje tablične podatke slične onim prikazanim u tablici 3. Međutim, takvi podaci mogu se dalje koristiti za grafički prikaz kao što je napravljeno na primjeru za razinu koljena (Slika 13).



Slika 13. Grafički prikaz deset najzastupljenijih koljena u projektu prema srednjoj vrijednosti relativne brojnosti. Rezultati su dobiveni provođenjem rutine `project_top_abundance` uz zadane varijable: `2` – razina taksonomije je koljeno, `10` – prikaz prvih 10 rezultata. Izvorni tablični rezultati nalaze se u prilogu 3.11.

Kao što se pomoću bogatstva vrsta mogu izračunati sve potrebne vrijednosti za kutijasti dijagram, rutina `box_plot_abundance` vrši jednake proračune pomoću relativne brojnosti vrsta za odabranu razinu taksonomske klasifikacije. Iako se već iz tablice mogu vidjeti sve vrijednosti, putem računalnog programa Excel može se napraviti dodatna tablica iz koje je, iako nekonvencionalnim metodama, moguće napraviti kutijasti dijagram. Kao primjer je sastavljen kutijasti dijagram za 5 koljena koja imaju najvišu vrijednost medijana u projektu (Slika 14). Prema relativnoj brojnosti to su *Bacteroidetes*, *Firmicutes*, *Synergistetes*, *Proteobacteria* i *Verrucomicrobia*.



Slika 14. Kutijasti dijagram s 5 koljena koja imaju najveće medijane u projektu. Rezultati su dobiveni provođenjem rutine `box_plot_abundance` uz zadane varijable: `2` – razina taksonomije je koljeno, `5` – prikaz prvih 5 rezultata. Izvorni tablični rezultati nalaze se u prilogu 3.12.

4.2. RASPRAVA

Kao stanovnici ovog svijeta punog mikroorganizama, ljudi su domaćini velikog broja bakterija, a samim time i domaćini izvanredne bioraznolikosti (Xu i Gordon, 2003). Važno je naglasiti da uz bakterije u mikrobiotu spadaju i drugi mikroorganizmi, koji nude dodatnu dimenziju istraživanjima interakcija između domaćina i njegove mikrobiote (Cani, 2018). U nekim rezultatima provedenih rutina stoga se među bakterijama uobičajeno nalaze i arheje, ali može se naići i na neimenovane viruse. Za viruse ne postoje toliko razvijeni pristupi za analize, kao ni dovoljno specifičnih baza podataka, pa se uobičajeno ne uzimaju u obzir, iako to znači da je u konačnici zanemaren njihov utjecaj na zdravlje domaćina (D'Argenio i sur., 2018). S obzirom kako je cilj ovog rada zapravo bila sama izrada SQL upita za analizu sastava crijevne mikrobiote, nije bilo potrebe za provođenjem nikakvog dodatnog filtriranja podataka koji su se već nalazili u bazi.

Baza `bio` sadrži samo osnovne podatke dobivene sekvenciranjem crijevne mikrobiote pojedinaca, bez ikakvih popratnih podataka. Iz tog razloga se provođenjem osmišljenih rutina dobivaju samo rezultati o taksonomskom sastavu mikrobiote, odnosno raznolikosti pojedinih uzoraka, uz prosječne rezultate za cijeli projekt. Ove rutine se mogu provoditi za bilo koji uzorak na bilo kojoj taksonomskoj razini, što omogućuje procjenu varijabilnosti na pojedinim razinama. Primjeri rezultata izrađenih analiza svakako potvrđuju kako je sastav crijevne mikrobiote među pojedincima jedinstven, što je posebice izraženo na nižim taksonomskim razinama, dok na višim taksonomskim razinama postoje određene sličnosti i češće zastupljeni taksoni. Tako je iz primjera rezultata za prosječno bogatstvo vrsta i brojnost na razini koljena vidljivo da *Firmicutes* ima najveću raznolikost i bogatstvo vrsta unutar projekta (Tablica 3), no *Bacteroidetes* je koljeno s vrstama najveće brojnosti (Slika 13). Ista rutina može se provesti na razini roda, ako se želi utvrditi 'temeljni' sastav mikrobiote, odnosno koji se rodovi najčešće i u najvećoj brojnosti pojavljuju u projektu. Što se tiče rutina na razini pojedinih uzoraka, zbog nedostatka dodatnih podataka mogu se provoditi samo nasumice. Njima se dobivaju informacije o najzastupljenijim taksonima za odabrani uzorak, te je omogućeno izračunavanje Shannonovog indeksa i inverznog Simpsonovog indeksa, kao i uspoređivanje sličnosti dva nasumično odabrana uzorka pomoću Jaccardovog koeficijenta sličnosti. Međutim, na temelju takvih rezultata ne mogu se donijeti nikakvi stvarni zaključci o povezanosti sastava crijevne mikrobiote s bilo kojim unutarnjim i vanjskim čimbenicima.

Postoji nekoliko smjerova u kojima bi se dalje mogla razvijati baza podataka `bio`, a zajedno s njom i izrađeni pohranjeni postupci. Osnovna i jednostavnija proširenja uključuju: (i) optimizaciju postojećih rutina, (ii) osmišljanje i dodatak novih rutina, te (iii) dodatak novih podataka u postojeće tablice baze `bio`. Izvedbeno i vremenski zahtjevnija proširenja i unaprjeđenja uključuju: (i) dodatak podataka o raznim vanjskim i unutarnjim čimbenicima vezanim uz pojedine uzorke i razvoj novih rutina s obzirom na dodane podatke, (ii) vizualizaciju rezultata uz pomoć dodatnih računalnih alata i programskih jezika, uz mogućnost automatizacije, te (iii) razvoj vlastite sveobuhvatne web aplikacije za automatiziranu analizu podataka i vizualizaciju rezultata.

Pohranjeni postupci su već sami po sebi mnogo efikasnije rješenje od korištenja uobičajenih upita. Sam cilj korištenja pohranjenih postupaka je unaprjeđenje izvedbi uzastopnih akcija, za što je potreban optimiziran i kompiliran kôd. S obzirom da je izvedba rutine ovisna o odabiru i pohrani plana u predmemoriju, njezina efikasnost uvelike ovisi o optimalnosti plana, na što često utječe i odabir parametara. Prema tome, ovisno o potrebama i okolnostima, potrebno je optimizirati kôd kako bi MySQL Server efikasno provodio pozvane rutine, u što kraćem vremenu. To je posebice bitno kod izrade web aplikacija povezanih s bazom podataka. U svrhu optimizacije upiti izrađenih rutina morali bi se skratiti i jednostavnije organizirati u skladu s procesom i hijerarhijom provođenja akcija MySQL Server-a.

Uz već postojeće, moguće je izraditi i dodati nove pohranjene postupke koji se temelje na postojećim podacima u bazi `bio`. Za potrebe ovog diplomskog rada izrađene su rutine za tri indeksa koji služe kao statističke mjere bioraznolikosti: Shannonov indeks, inverzni Simpsonov indeks i Jaccardov koeficijent sličnosti. Međutim, postoji još nekolicina indeksa raznolikosti koji se često upotrebljavaju u analizi bioraznolikosti. Tako bi bilo moguće sastaviti rutine za dodatne indekse, poput Berger-Parker indeksa, Smith-Wilsonovog indeksa podjednakosti, Sorensonovog koeficijenta sličnosti, i drugih.

Shema `bio` već je osmišljena i za pohranu podataka o biokemijskim putevima za koje kodira mikrobiom, a koji se iz podataka sekvenciranja dobivaju HUMAnN2 alatom. U tu svrhu se u shemi nalaze tablice `path_name` (hrv. ime puta) i `pathway_abundance` (hrv. brojnost puta), koje su trenutno prazne. Izradom upita koji se odnose na te podatke mogao bi se pobliže utvrditi funkcijski sastav mikrobiote. Tada bi također bilo moguće uspoređivati funkcijski profil s taksonomskim profilom određenog uzorka, te općenito uspoređivati povezanost pojedinih funkcija s određenim sastavom mikrobiote.

U shemu `bio` moguće je dalje dodavati stupce i tablice prema potrebi, i povezati ih s već postojećima. Kada bi uz uzorke u bazi podataka bili vezani i podaci o spolu, dobi, prehrani, raznim bolestima i stanjima, medicinskim intervencijama, korištenju lijekova, dodataka prehrani, probiotika, i drugim čimbenicima, postojeći upiti bi se mogli proširiti za provođenje usmjerenijih analiza koje uzimaju bilo koje od željenih čimbenika u obzir. Tada bi također bilo moguće izraditi dodatne pohranjene postupe prema potrebama istraživanja koje se želi provesti. Primjerice, uz minimalne izmjene rutina za određivanje prosječno najzastupljenijih taksona na temelju bogatstva vrsta i relativne brojnosti vrsta za projekt bilo bi moguće usporediti najzastupljenije taksone skupina pretilih i skupina mršavih ljudi, ili pak skupina ljudi koji su se liječili antibioticima i kontrole koja nije. Također bi se pružio sažetiji uvid u najzastupljenije taksone na temelju bogatstva vrsta i relativne brojnosti vrsta te indekse raznolikosti za pojedine uzorke, pri čemu bi se uzorci mogli odabirati na temelju željenog broja definiranih parametara, odnosno na temelju odabranih čimbenika.

Primjeri rezultata provedenih rutina u ovom radu grafički su prikazani uz pomoć računalnog alata Excel i dodatka MySQL for Excel. Međutim, korištenjem programskih jezika za koje MySQL sustav sadrži API i konektor, kao što su Perl ili Python (MySQL, 2019), omogućila bi se veća automatizacija postupka pozivanja rutina i grafičkog prikaza rezultata nakon što se provede rutina. Takvo unaprjeđenje zahtjeva poznavanje još najmanje jednog programskog jezika uz poznavanje računalnog jezika SQL, te je potrebno znanje modula za izradu grafičkih prikaza u Perlu ili Pythonu.

Prema sličnom principu, MySQL baza podatka može se koristiti pri izradi vlastite sveobuhvatne aplikacije za analizu i vizualizaciju podataka, za što su također potrebni programski jezici za koje MySQL sustav sadrži API i konektor, kao što su Python ili PHP (MySQL, 2019). Kombinacijom jednog od tih programskih jezika s HTML, CSS i JavaScript jezicima bilo bi moguće napraviti potpuno funkcionalnu web aplikaciju za unos, obradu, analizu, i vizualizaciju podataka. Tu su naročito korisni pohranjeni postupci, jer su mnogo brži od uobičajenih upita, nude veću sigurnost, te uvijek ostaju organizirani i dostupni za ponovnu upotrebu. Međutim, u ovakvu svrhu je poželjno koristiti optimizirane rutine, a uz to je potrebno znanje većeg broja računalnih i programskih jezika te jezika za označavanje.

MySQL je jedan od najpopularnijih sustava za upravljanje relacijskim bazama podataka, zbog čega je i odabran za izradu ovog rada. Iako već postoje različiti bioinformatički alati i *pipeline*-ovi za analizu podataka dobivenih sekvenciranjem mikroorganizama, SQL je odabran kako bi se unutar istog programskog paketa mogle odraditi pohrana podataka i njihova analiza, što olakšava rukovanje velikom količinom podataka koja se dobiva nakon sekvenciranja i inicijalne bioinformatičke obrade.

Međutim, postoje mnogi već razvijeni bioinformatički alati i platforme koji su u svakodnevnoj upotrebi. Primjerice, QIIME 2 je bioinformatička platforma koja se često koristi za analize mikrobioma, uključujući analize bioraznolikosti, a može koristiti napredne dodatke za vizualizaciju. Koristi programski jezik Python te dopušta unos podataka u bilo kojoj fazi analize, pri čemu ih pretvara u poseban QIIME format, takozvane artefakte. Može se nativno instalirati na računala s Mac i Linux operativnim sustavima, dok je za instalaciju na Windows operativni sustav potreban virtualni okoliš (eng, *virtual box*) te su instalacija i korištenje nešto kompliciraniji (Boylen i sur., 2019).

Velika prednost korištenja alata u sklopu takve bioinformatičke platforme je što nudi dodatke za mnogo različitih analiza. Konkretno, dodatak za bioraznolikost nudi izračun preko 20 različitih indeksa alfa raznolikosti, uključujući Shannonov i Simpsonov indeks, te čak 23 indeksa beta raznolikosti, uključujući Jaccardov koeficijent sličnosti. Isti paket sadrži još neke dodatne analize bioraznolikosti, te nudi nekoliko različitih načina vizualizacije rezultata. U usporedbi s tim impresivnim odlikama QIIME 2 platforme, korištenje MySQL-a mnogo je nezgrapnije, čak i ako se baza podataka proširi i dodaju se rutine za izračun još mnogo indeksa raznolikosti (Boylen i sur., 2019, Caporaso i sur., 2010). Problem s korištenjem MySQL-a, onako kako je upotrijebljen u ovom radu, je u ograničenim mogućnostima, odnosno nemogućnosti potpune automatizacije vizualizacije rezultata. Daljnjim radom na razvijanju baze i rutina svakako se može dostići funkcionalnost QIIME 2 platforme što se tiče analiza bioraznolikosti, no za unaprjeđenje vizualizacije potrebno je korištenje još najmanje jednog programskog jezika što može dodatno zakomplicirati daljnji razvoj. Međutim, gledano iz perspektive istraživača koji nisu nužno obrazovani za rad s Pythonom i najčešće imaju Windows operativni sustav, rutine razvijene u MySQL-u uz vizualizaciju u računalnom programu Microsoft Excel relativno su jednostavan pothvat za analizu bioloških podataka. Besplatna verzija MySQL programskog paketa široko je dostupna, a instalacija je vrlo jednostavna. MySQL Workbench ima intuitivno grafičko sučelje, pa nije komplicirano uvesti već pripremljenu bazu podataka s pohranjenim postupcima za analizu u vlastiti MySQL Server.

Dodatak MySQL for Excel također je jednostavno instalirati i koristiti u sklopu Excel računalnog programa, a dotični većina istraživača već koristi za rad s tablicama i grafovima. Dakle, iako je grafički prikaz rezultata u računalnom programu Excel koji je korišten u ovom radu potencijalno inferioran prikazima većine dostupnih bioinformatičkih alata, u većini slučajeva mogao bi biti dostupniji i jednostavniji za korištenje. Svakako je jasno da grafički prikaz često olakšava interpretaciju rezultata, te čak pruža novu perspektivu u mnogim istraživanjima. Međutim, jedna velika prednost korištenja MySQL sustava za pohranu i analizu podataka je i u lakoći uvida u rezultate, iako u tabličnom obliku. Sintaksa računalnog jezika SQL prilično je jednostavna ukoliko se žele provoditi osnovni upiti za sortiranje i filtriranje podataka, što omogućuje lak pregled željenih rezultata. Štoviše, pomoću jednostavnih naredbi moguće je odabrati samo određene rezultate prema željenim kriterijima, izvesti ih iz baze u određenom formatu, te zatim uvesti u većinu bioinformatičkih *pipeline*-ova za daljnju analizu.

Zadnje što valja spomenuti su koristi ovakvih analiza i istraživanja u kojima bi mogla poslužiti. U ovom radu se mnogo puta spominju temeljni mikrobiom te povezanost sastava mikrobiote s raznim vanjskim i unutarnjim čimbenicima. Analize bioraznolikosti te usporedbe rezultata za što veći broj uzoraka crijevne mikrobiote, pogotovo ako se u obzir uzima i prisutnost pojedinih čimbenika, daju uvid u utjecaj tih čimbenika na bioraznolikost mikrobiote. Sa sve opširnijim projektima i istraživanjima stječu se sve konkretniji podaci o mogućem temeljnom mikrobiomu te povezanosti određenih promjena sastava mikrobiote s pojedinim uvjetima. Gorvitovskaia i sur. proveli su opsežna istraživanja objedinjujući uzorke s nekoliko kontinenata te na temelju rezultata predložili da se temeljni taksoni povezani sa životnim stilom, prehranom, bolestima i drugim čimbenicima zapravo trebaju smatrati biomarkerima, jer bi mogli olakšati utvrđivanje promjena u sastavu mikrobiote, te njihove uzroke i posljedice (Gorvitovskaia i sur., 2016). Prema tome, zahvaljujući određivanju temeljnih taksona i analizama raznolikosti mikrobiote, u budućnosti bi se moglo intervencijama na sastav utjecati na razna stanja povezana s disbiozom ili nepovoljnim sastavom kod pojedinaca, što je dio jedne velike priče o personaliziranim lijekovima.

U svakom slučaju, baza `bio-complete.sql` koja sadrži sve izrađene pohranjene postupke za analizu bioraznolikosti u trenutnom stanju predstavlja samo osnovu i početak jednog drugačijeg pristupa analizi bioloških podataka. Moguće ju je, čak i poželjno dalje razvijati, dodavati tablice, unositi nove podatke i proširivati postojeće, dodavati nove i unaprijediti postojeće pohranjene postupke, te je u konačnici uklopiti u *pipeline* za analizu prema potrebi.

5. ZAKLJUČCI

1. Pomoću sustava za upravljanje bazama podataka MySQL i računalnog alata SQLyog u sklopu baze podataka `bio` su upotrebom računalnog jezika SQL uspješno izrađeni i provjereni upiti sastavljeni u obliku pohranjenih postupaka (rutina) za:
    a. određivanje bogatstva vrsta i relativne brojnosti taksona odabrane razine taksonomske klasifikacije za pojedini uzorak,
    b. izračunavanje Shannonovog indeksa i Shannonove podjednakosti, te inverznog Simpsonovog indeksa i Simpsonove nepristranosti za pojedini uzorak,
    c. izračunavanje Shannonovog indeksa i inverznog Simpsonovog indeksa za veći raspon uzoraka,
    d. uspoređivanje sličnosti dva uzorka izračunavanjem Jaccardovog koeficijenta sličnosti i Jaccardove udaljenosti,
    e. određivanje srednjih vrijednosti bogatstva vrsta i relativne brojnosti taksona odabrane razine taksonomske klasifikacije za čitav projekt,
    f. pronalaženje i izračunavanje vrijednosti bogatstva vrsta i relativne brojnosti taksona odabrane razine taksonomske klasifikacije potrebnih za konstrukciju kutijastog dijagrama za čitav projekt.
2. Pomoću dodatka MySQL for Excel i računalnog programa Microsoft Excel uspješno su grafički prikazani primjeri rezultata navedenih pohranjenih postupaka.

6. LITERATURA

Andrews, S. (2010) FastQC: A quality control tool for high throughput sequence data. < https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>. Pristupljeno 5. studenog 2019.

ANSI (2018) The SQL standard – ISO/IEC 9075:2016, *ANSI – American National Standards Institute*, <https://blog.ansi.org/2018/10/sql-standard-iso-iec-9075-2016-ansi-x3-135/#gref>. Pristupljeno 30. listopada 2019.

Arumugam, M., Raes, J., Pelletier, E., i sur. (2011) Enterotypes of the human gut microbiome. *Nature*, **473**, 174-180.

Bäckhed, F. (2005) Host-bacterial mutualism in the human intestine. *Science*, **307**: 1915-1920.

Bäckhed, F., Fraser, C. M., Ringel, Y., i sur. (2012) Defining a healthy human gut microbiome: current concepts, future directions, and clinical applications. *Cell Host Microbe*, **12**: 611-622.

Bäumler, A. J., Sperandio, V. (2016) Interactions between the microbiota and pathogenic bacteria in the gut. *Nature*, **535**, 85-93.

Bengmark, S. (1998) Ecological control of the gastrointestinal tract. The role of probiotic flora. *Gut*, **42**, 2-7.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., i sur. (2005) GenBank. *Nucleic Acids Res.*, **33** (Database issue), D34-D38.

Bessant, C., Oakley, D., Shadforth, I. (2014) Building Bioinformatics Solutions with Perl, R, and SQL, 2. izd., *Oxford University Press*, Oxford, UK.

Boylen, E., Rideout, J. R., Dillon, M. R., i sur., (2019) Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nat. Biotechnol.*, **37**(8), 852-857.

Caporaso, J. Z., Kuczynski, J., Stombaugh, J., i sur. (2010) QIIME allows analysis of high-throughput community sequencing data. *Nat. Methods*, **7**, 335-336.

Claesson, M. J., Cusack, S., O'Sullivan, i sur. (2011) Composition, variability, and temporal stability of the intestinal microbiota of the elderly. *Proc. Natl Acad. Sci. USA*, **108** (Suppl. 1), 4586-4591.

Clausen, P. T. L. C., Aarestrup, F. M., Lund, O. (2018) Rapid and precise alignment of raw reads against redundant databases with KMA. *BMC Bioinformatics*, **19**, 1-8.

Clemente, J. C., Ursell, L. K., Wegener Parfrey, L., i sur. (2012) The impact of the gut microbiota on human health: an integrative view. *Cell*, **148**, 1258-1270.

Cole, J. R., Wang, Q., Fish, J. A., i sur. (2014) Ribosomal Database Project: data and tools for high throughput rRNA analysis. *Nucleic Acids Res.*, **42** (Database issue), D633-D642.

Damian, D. (2009) Data and databases. U: Bioinformatics: tools and applications (Edwards, D., Stajich, J., Hansen, D., ured.). *Springer*, New York, USA, str. 381-401.

Date, C. J. (2000) An introduction to database systems, vol. 1, *Addison-Wesley Publishing Company*, Reading, Massachusets, SAD.

DeSantis, T. Z., Hugenholtz, P., Larsen, N., i sur. (2006) Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl. Environ. Microbiol.*, **72**(7), 5069-5072.

Eurofins Genomics (2019) Material and methods: microbiome sequencing, <https://www.eurofinsgenomics.eu/en/eurofins-genomics/material-and-methods/microbiome-sequencing/>. Pristupljeno 22. listopada 2019.

Falony, G., Joossens, M., Vieira-Silva, S., i sur. (2016) Population-level analysis of gut microbiome variation. *Science*, **352** (6285), 560-564.

Franzosa, E. A., McIver, L. J., Rahnavard, G., i sur. (2018) Species-level functional profiling of metagenomes and metatranscriptomes. *Nat. Methods*, **15**, 962-968.

Geuking, M. B., Cahenzli, J., Lawson, M. A. E., i sur. (2011) Intestinal bacterial colonization induces mutualistic regulatory T-cell responses. *Immunity*, **34**(5), 697-699.

Grice, E. A., Segre, J. A. (2012) The human microbiome: our second genome. *Annu. Rev. Genomics Hum. Genet.*, **13**, 151-170.

Grolinger, K., Higashino, W. A., Tiwari, A., i sur. (2013) Data management in cloud environments: NoSQL and NewSQL data stores. *J. Cloud Comp.*, **2**, 22.

Grune, D., Jacobs, C. J. H. (1990) Parsing techniques – A practical guide. *Ellis Horwood*, Chichester, England.

Haas, B. J., Gevers, D., Earl, A. M., i sur. (2011) Chimeric 16S rRNA sequence formation and detection in Sanger and 454-pyrosequenced PCR amplicons. *Genome Res.*, **21**(3), 494-504.

Hamady, M., Walker, J. J., Harris, J. K., i sur. (2008) Error-correcting barcoded primers for pyrosequencing hundreds of samples in multiplex. *Nat. Methods*, **5**, 235-237.

Honda, K., Littman, D. R. (2016) The microbiota in adaptive immune homeostasis and disease. *Nature*, **535**, 75-84.

Hooper, L. V., Midtvedt, T., Gordon, J. I. (2002) How host-microbial interactions shape the nutrient environment of the mammalian intestine. *Annu. Rev. Nutr.*, **22**, 283-307.

Hsiao, E. Y., McBride, S. Q., Hsien, S., i sur. (2013) Microbiota modulate behavioral and physiological abnormalities associated with neurodevelopmental disorders. *Cell*, **155**, 1451-1463.

Huang, K., Brady, A., Mahurkar, A., i sur. (2014) MetaRef: A pan-genomic database for comparative and community microbial genomics. *Nucleic Acids Res.*, **42**, D617-D624.

Human Microbiome Project Consortium (2012) Structure, function and diversity of the healthy human microbiome. *Nature*, **486**, 207-214.

Imhann, F., Van Der Velde, K. J., Barbieri, R., i sur. (2019) The 1000IBD project: multi-omics data of 1000 inflammatory bowel disease patients; data release 1. *BMC Gastroenterol.*, **19**, 1-10.

ISO (2019) ISO/IEC 9075-15:2019. *ISO- International Organization for Standardization*, <https://www.iso.org/standard/67382.html>. Pristupljeno 30. listopada 2019.

Jandhyala, S. M., Talukdar, R., Subramanyam, C., i sur. (2015) Role of the normal gut microbiota. *World J. Gastroenterol.*, **21**(29), 8787-8803.

Jeffrey, I. B., Claesson, M. J., O'Toole, P. W., i sur. (2012) Categorization of the gut microbiota: enterotypes or gradients? *Nat. Rev. Microbiol.*, **10**, 591-592.

Johansson, M. E., Larsson, J. M., Hansson, G. C. (2011) The two mucus layers of colon are organized by the MUC2 mucin, whereas the outer layer is a legislator of host-microbial interactions. *Proc. Natl. Acad. Sci. USA*, **108** (Suppl. 1), 4659-4665.

Kanz, C., Aldebert, P., Althorpe, N., i sur. (2005) The EMBL nucleotide sequence database. *Nucleic Acids Res.*, **33** (Database issue), D29-D33.

Kerepesi, C., Banky, D., Grolmusz, V. (2014) AmphoraNet: the webserver implementation of the AMPHORA2 metagenomic workflow suite. *Gene*, **533**, 538-540.

Langille, M. G. I., Zaneveld, J., Caporaso, J. G., i sur. (2013) Predictive functional profiling of microbial communities using 16S rRNA marker gene sequences. *Nat. Biotechnol.*, **8**, 1-10.

Laursen, P. (2017) Manage your MySQL databases with SQLyog. *Webyog*, <http://blog.sqlyog.com/manage-mysql-databases-sqlyog/>. Pristupljeno 31. listopada 2019.

Lloyd-Price, J., Abu-Ali, G., Huttenhower, C. (2016) The healthy human microbiome. *Genome Med.*, **8**, 51.

Lynch, S. V., Pedersen, O. (2016) The human intestinal microbiome in health and disease. *N. Engl. J. Med.*, **375**, 2369-2379.

Macpherson, A. J., Slack, E., Geuking, M. B., i sur. (2009) The mucosal firewalls against commensal intestinal microbes. *Semin. Immunopathol.*, **31**, 145-149.

Markowitz, V. M., Chen, I. M., Chu, K., i sur. (2014) IMG/M 4 version of the integrated metagenome comparative analysis system. *Nucleic Acids Res.*, **42**, D568-D573.

McIver, L. J., Abu-Ali, G., Franzosa, E. A., i sur. (2018) BioBakery: A Meta'omic analysis environment. *Bioinformatics*, **34**, 1235-1237.

Morgan, X. C., Huttenhower, C. (2014) Meta'omic analytic techniques for studying the intestinal microbiome. *Gastroenterol.*, **146**, 1437-1448.

MySQL (2019) MySQL 8.0 Reference Manual. *Oracle*, <https://dev.mysql.com/doc/refman/8.0/en/>. Pristupljeno 31. listopada 2019.

Nicholson, J. K., Holmes, E., Kinross, J., i sur. (2012) Host-gut microbiota metabolic interactions. *Science*, **336**, 1262-1267.

O'Hara, A. M., Shanahan, F. (2006) The gut flora as a forgotten organ. *EMBO Rep.*, **7**, 688-693.

Petersen, C., Round J. L. (2014) Defining dysbiosis and its influence on host immunity and disease. *Cell Microbiol.*, **16**, 1024-1033.

Peterson, D. A., Frank, D. N., Pace, N. R., i sur. (2008) Metagenomic approaches for defining the pathogebnesis of inflammatory bowel diseases. *Cell Host Microbe*, **3**, 417-427.

Peterson, J., Garges, S., Giovanni, M., i sur.; NIH HMP Working Group (2009) The NIH Human Microbiome Project. *Genome Res.*, **19**, 2317-2323.

Qin, J., Li, R., Raes, J., Arumugam, M., i sur.; MetaHIT Consortium (2010) A human gut microbial gene catalogue established by metagenomic sequencing. *Nature*, **464**, 59-65.

Quast, C., Pruesse, E., Yilmaz, P., i sur. (2013) The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic Acids Res.*, **41** (Database issue), D590-D596.

Quince, C., Walker, A. W., Simpson, J. T., i sur. (2017) Shotgun metagenomics, from sampling to analysis. *Nat. Biotechnol.*, **35**, 833-844.

Rinninella, E., Raoul, P., Cintoni, M., i sur. (2019) What is the healthy gut microbiota composition? A changing ecosystem across age, environment, diet and diseases. *Microorganisms*, **7**(1), 14.

Rivas, M. A., Beaudoin, M., Gardet, A., i sur. (2011) Deep resequencing of GWAS loci identifies independent rare variants associated with inflammatory bowel disease. *Nat. Genet*., **43**, 1066–1073.

Rosenberg, E. (2017) Human microbiome: We are not alone. U: It's in your DNA (Rosenberg, E., ured.), *Academic Press*, str. 105-114.

Schloss, P. D., Gevers, D., Westcott, S. L. (2011) Reducing the effects of PCR amplification and sequencing artifacts on 16S rRNA-based studies. *PLoS One*, **6**, e27310.

Schloss, P. D., Westcott, S. L., Ryabin, T., i sur. (2009) Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Appl. Environ. Microbiol.*, **75**(23), 7537-7541.

Segata, N., Izard, J., Waldron, L., i sur. (2011) Metagenomic biomarker discovery and explanation. *Genome Biol.*, **12**(6), R60.

Segata, N., Waldron, L., Ballarini, A., i sur. (2012) Metagenomic microbial community profiling using unique clade-specific marker genes. *Nat. Methods*, **9**(8), 811-814.

Sender, R., Fuchs, S., Milo, R. (2016) Revised estimates for the number of human and bacteria cells in the body. *PLoS Biology*, **14**, e1002533.

Shafquat, A., Joice, R., Simmons, S. L., i sur. (2014) Functional and phylogenetic assembly of microbial communities in the human microbiome. *Trends Microbiol.*, **22**, 261–266.

Shin, N. R., Whon, T. W., Bae, J. W. (2015) *Proteobacteria*: microbial signature of dysbiosis in gut microbiota. *Trends Biotechnol.*, **33**(9), 496-503.

Singh, B., Crippen, T. L., Tomberlin, J. K. (2017) An introduction to metagenomic data generation, analysis, visualization, and interpretation. U: Forensic Microbiology, 1. izdanje (Carter, D. O., Tomberlin, J. K., Benbow, M. E., Metcalf, J. L., ured.), *John Wiley & Sons Ltd.*, str. 94-126.

Smith, K., McCoy, K. D., Macpherson, A. J. (2007) Use of axenic animals in studying the adaptation of mammals to their commensal intestinal microbiota. *Semin. Immunol.*, **19**, 59-69.

Sommer, F., Bäckhed, F. (2013) The gut microbiota – masters of host development and physiology. *Nat. Rev. Microbiol.*, **11**(4), 227-238.

Suehring, S. (2002) MySQL Bible. *Wiley Publishing, Inc.*, New York, USA.

Tigchelaar, E. F., Zhernakova, A., Dekens, J. A. M., i sur. (2015) Cohort profile: LifeLines DEEP, a prospective, general population cohort study in the northern Netherlands: study design and baseline characteristics. *BMJ Open*, **5**, e006772.

Truong, D. T., Franzosa, E. A., Tickle, T. L., i sur. (2015) MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nat. Methods*, **12**, 902-903.

Turnbaugh, P. J., Hamady, M., Yatsunenko, T., i sur. (2009) A core gut microbiome in obese and lean twins. *Nature*, **457**, 480-484.

Turnbaugh, P. J., Ley, R. E., Hamady, M., i sur. (2007) The human microbiome project: exploring the microbial part of ourselves in a changing world. *Nature*, **449**(7164), 804-810.

Ullman, L. (2003) MySQL: Visual QuickStart Guide. *Peachpit Press*, Berkley, Michigan, SAD.

Ullman, J. D., Widom, J. (2007) A First Course in Database Systems, 3. izd., *Pearson.*

Vaishnava, S., Yamamoto, M., Severson, K. M., i sur. (2011) The antibacterial lectin RegIIIgamma promotes the spatial segregation of microbiota and host in the intestine. *Science*, **334**, 255-258.

Veech, J. A. (2018) Measuring biodiversity. U: Encyclopedia of the Anthropocene, 3. dio (Dellasala, D. A., Goldstein, M. I., ured.), *Elsevier.*

Westhead, D. R., Parish, J. H., Twyman, R. M. (2002) Instant notes: Bioinformatics. *BIOS Scientific Publishers Ltd.*, Oxford, UK.

Whittaker, R. H. (1972) Evolution and measurement of species diversity. *Taxon*, **21**(2/3), 213-251.

Wu, G. D., Chen, J., Hoffmann, C., i sur. (2011) Linking long-term dietary patterns with gut microbial enterotypes. *Science*, **334** (6052), 105-108.

Wu, M., Scott, A. J. (2012) Phylogenomic analysis of bacterial and archaeal sequences with AMPHORA2. *Bioinformatics,* **28**, 1033-1034.

Xu, J., Gordon, J. I. (2003) Honor thy symbionts. *Proc. Natl. Acad. Sci. USA*, **100**: 10452-10459

Yatsunenko, T., Rey, F. E., Manary, M. J., i sur. (2012) Human gut microbiome viewed across age and geography. *Nature*, **486**, 222-227.

Yin, Y., Mao, X, Jang, J., i sur. (2012) dbCAN: A web resource for automated carbohydrate-active enzyme annotation. *Nucleic Acids Res.*, **40**, W445-W451.

Yoon, S. H., Ha, S. M., Kwon, S., i sur. (2017) Introducing EzBioCloud: a taxonomically united database of 16S rRNA gene sequences and whole-genome assemblies. *Int. J. Syst. Evol. Microbiol.*, **67**(5), 1613-1617.

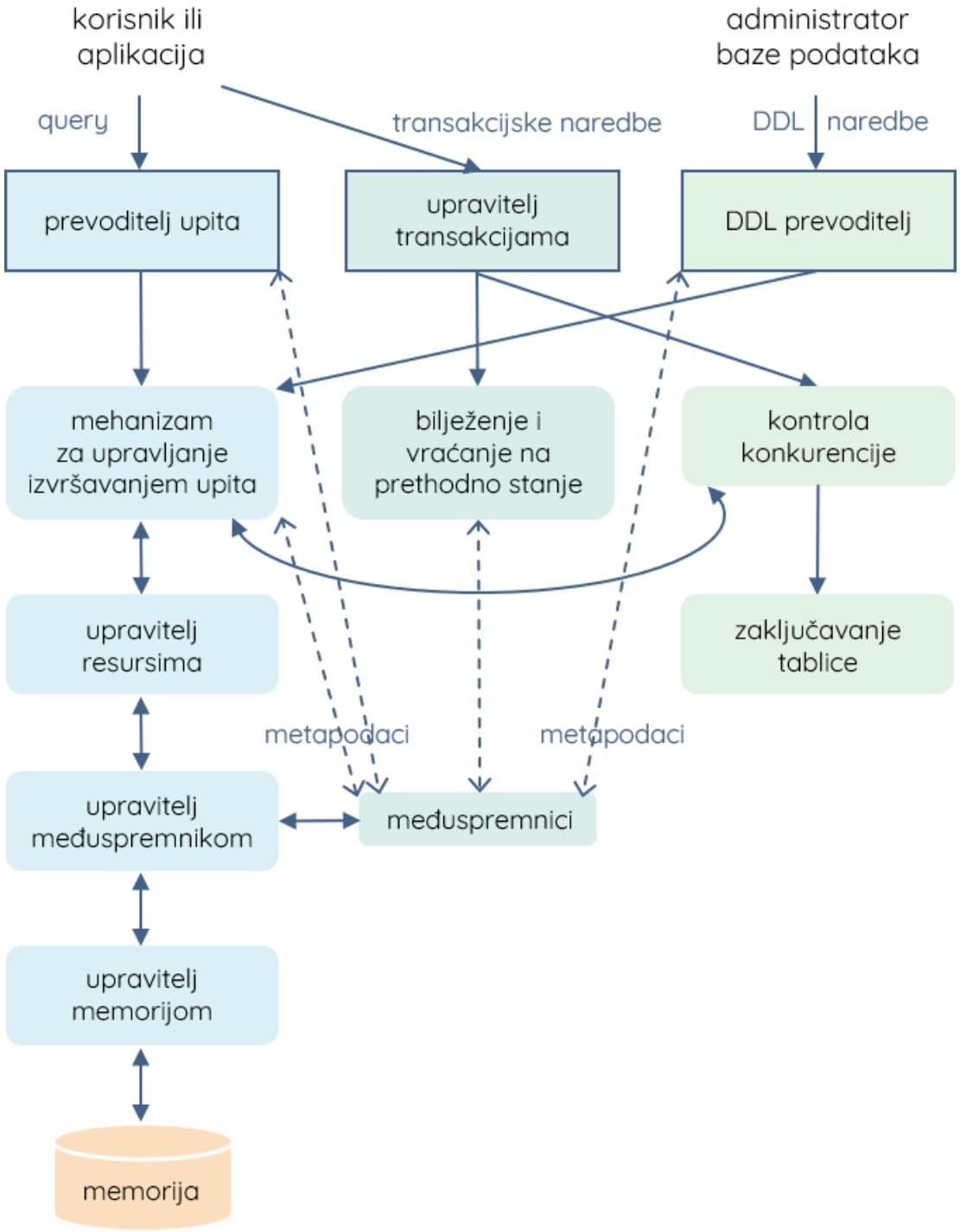
Zhernakova, A., Kurilshikov, A., Bonder, M. J., i sur. (2016) Population-based metagenomics analysis reveals markers for gut microbiome composition and diversity. *Science*, **352** (6285), 565-569.

7. PRILOZI

PRILOG 1: POPIS KORIŠTENIH KRATICA

API	Sučelje za programiranjea aplikacija; akronim za *Application Programming Interface*
CSV	*Flat* format datoteka s vrijednostima razdvojenim zarezom; akronim za *Comma-Separated Values*.
DBMS	Sustav za upravljanje bazama podataka; akronim za *Database Management System*.
DDL	Računalni jezik; akronim za *Data Definition Language*.
DML	Računalni jezik; akronim za *Data Manipulation Language*.
HUMAnN	Bioinformatički alat; akronim za *The HMP Unified Metabolic Analysis Network*.
MetaPhlAn	Bioinformatički alat; akronim za *Metagenomic Phylogenetic Analysis*.
NGS	Sekvenciranje sljedeće generacije; akronim za *Next Generation Sequencing*
OTU	Operacijska taksonomska jedinica; akronim za *Operational Taxonomic Unit*.
PCR	Polimerazna lančana reakcija; akronim za *Polymerase Chain Reaction*.
RDBMS	Sustav za upravljanje relacijskim bazama podataka; akronim za *Relational Database Management System*.
RDP	Baza podataka rRNA gena; akronim za *Ribosomal Database Project*.
SQL	Računalni jezik; akronim za *Structured Query Language*.
XML	Računalni jezik; akronim za *Extensible Markup Language*.

PRILOG 2: DODATNE SLIKE



Prilog 2.1. Shematski prikaz komponenti sustava za upravljanje bazama podataka (Prilagođeno prema Ullman i Widom, 2007).

PRILOG 3: IZVORNI TABLIČNI REZULTATI

Prilog 3.1. Rezultat pozivanja rutine `sample_top_richness (205, 2, 5).`

#	phylum	richness	average	st_dev
1	Firmicutes	27	31.6007	9.5084
2	Bacteroidetes	7	15.8346	7.3892
3	Proteobacteria	5	4.8565	2.5126
4	Actinobacteria	1	2.9603	1.7973
5	Viruses_noname	1	0.4444	0.5162

Prilog 3.2. Rezultat pozivanja rutine `sample_top_abundance (205, 2, 5).`

#	phylum	tax_abundance	average	st_dev
1	Firmicutes	0.522158	0.332071	0.2184
2	Bacteroidetes	0.467351	0.591014	0.2471
3	Proteobacteria	0.0056968	0.043260	0.1095
4	Actinobacteria	0.0047942	0.011738	0.0424
5	Viruses_noname	0.0000619	0.004759	0.0632

Prilog 3.3. Rezultat pozivanja rutine `sample_top_abundance (205, 7, 10).`

#	species	tax_abundance	average	st_dev
1	Bacteroides_ovatus	0.468888	0.036180	0.0665
2	Bacteroides_fragilis	0.183998	0.029832	0.0992
3	Veillonella_ratti	0.121665	0.000465	0.0406
4	Lactococcus_phage_936_sensu_lato	0.0585622	0.000390	0.0372
5	Bacteroides_xylanisolvens	0.0578434	0.005694	0.0169
6	Clostridium_clostridioforme	0.025417	0.002304	0.0217
7	Escherichia_coli	0.0226375	0.016836	0.0835
8	Clostridium_bolteae	0.019419	0.003929	0.0156
9	Dialister_invisus	0.0108852	0.010119	0.0427
10	Subdoligranulum_unclassified	0.0105207	0.031214	0.0510

Prilog 3.4. Rezultat pozivanja rutine `sample_shannon_index (74).`

sample	richness	shannon_index	average_shannon_index	index_st_dev	shannon_evenness	average_evenness	evenness_st_dev
74	57	2.7412	2.3477	0.5128	0.6780	0.5881	0.1061

Prilog 3.5. Rezultat pozivanja rutine `sample_simpson_index (74).`

sample	richness	simpson_index	average_simpson_index	index_st_dev	simpson_equitab.	average_equitab.	equitab._st_dev
74	57	2.7412	2.3477	0.5128	0.6780	0.5881	0.1061

Prilog 3.6. Rezultat pozivanja rutine `simpson_index_comparison (711, 721).`

sample	richness	simpson_index	simpson_equitab.	average_simpson_index	index_st_dev	average_equitab.	equitab._st_dev
711	46	0.3514	0.0076	0.2370	0.0645	0.0070	0.0373
712	54	0.3252	0.0060	0.2370	0.0645	0.0070	0.0373
713	60	0.1906	0.0032	0.2370	0.0645	0.0070	0.0373
714	69	0.1773	0.0026	0.2370	0.0645	0.0070	0.0373
715	52	0.1967	0.0038	0.2370	0.0645	0.0070	0.0373
716	34	0.2318	0.0068	0.2370	0.0645	0.0070	0.0373
717	58	0.2164	0.0037	0.2370	0.0645	0.0070	0.0373
718	69	0.2019	0.0029	0.2370	0.0645	0.0070	0.0373
719	57	0.2571	0.0045	0.2370	0.0645	0.0070	0.0373
720	39	0.1988	0.0051	0.2370	0.0645	0.0070	0.0373
721	48	0.2188	0.0046	0.2370	0.0645	0.0070	0.0373

Prilog 3.7. Rezultat pozivanja rutine `shannon_index_comparison (711, 721).`

sample	richness	shannon_index	shannon_evenness	average_shannon_index	index_st_dev	average_evenness	evenness_st_dev
716	34	2.4857	0.7049	2.3477	0.5128	0.5881	0.1061
720	39	2.9379	0.8019	2.3477	0.5128	0.5881	0.1061
711	46	1.1474	0.2997	2.3477	0.5128	0.5881	0.1061
721	48	2.5334	0.6544	2.3477	0.5128	0.5881	0.1061
715	52	2.7911	0.7064	2.3477	0.5128	0.5881	0.1061
712	54	1.0015	0.2511	2.3477	0.5128	0.5881	0.1061
719	57	1.7925	0.4434	2.3477	0.5128	0.5881	0.1061
717	58	2.5811	0.6357	2.3477	0.5128	0.5881	0.1061
713	60	2.8808	0.7036	2.3477	0.5128	0.5881	0.1061
714	69	2.9873	0.7055	2.3477	0.5128	0.5881	0.1061
718	69	2.5091	0.5926	2.3477	0.5128	0.5881	0.1061

Prilog 3.8. Rezultat pozivanja rutine `jaccard_index (21, 510).`

samples	species_in_common	all_species	jaccard_index	jaccard_distance
21 and 510	19	88	0.2159	0.7841

Prilog 3.9. Rezultat pozivanja rutine `project_top_richness (2, 10).`

#	phylum	average_richness	st_dev
1	Firmicutes	31.6007	9.5084
2	Bacteroidetes	15.8346	7.3892
3	Proteobacteria	4.8565	2.5126
4	Actinobacteria	2.9603	1.7973
5	Viruses_noname	0.4444	0.5162
6	Verrucomicrobia	0.3553	0.0000
7	Euryarchaeota	0.1154	0.4231
8	Fusobacteria	0.0604	0.2244
9	Deinococcus_Thermus	0.0226	0.0000
10	Ascomycota	0.022	0.4761

Prilog 3.10. Rezultat pozivanja rutine `box_plot_richness (2, 4).`

#	phylum	minimum	q1	med	q3	maximum
1	Firmicutes	1	26	32	37	63
2	Bacteroidetes	1	10	16	21	37
3	Proteobacteria	1	3	5	6	17
4	Actinobacteria	1	2	3	4	12

Prilog 3.11. Rezultat pozivanja rutine `project_top_abundance (2, 10).`

#	phylum	average	st_dev
1	Bacteroidetes	0.591014	0.2471
2	Firmicutes	0.332071	0.2184
3	Proteobacteria	0.043260	0.1095
4	Verrucomicrobia	0.012178	0.0822
5	Actinobacteria	0.011738	0.0424
6	Viruses_noname	0.004759	0.0632
7	Fusobacteria	0.000925	0.0513
8	Euryarchaeota	0.000701	0.0495
9	Synergistetes	0.000259	0.0187
10	Spirochaetes	0.000031	0.0103

Prilog 3.12. Rezultat pozivanja rutine `box_plot_abundance (2, 5).`

#	phylum	minimum	q1	med	q3	maximum
1	Bacteroidetes	0.000003	0.455547	0.663270	0.789584	0.994077
2	Firmicutes	0.000827	0.165137	0.281686	0.460372	1.000000
3	Synergistetes	0.000019	0.004826	0.023581	0.038448	0.062196
4	Proteobacteria	0.000016	0.005551	0.014561	0.033711	1.000000
5	Verrucomicrobia	0.000003	0.000795	0.005744	0.029917	0.815731

PRILOG 4: SADRŽAJ PRILOŽENOG CD-a

Na CD-u nazvanom *Diplomski rad – Andrea Plec* u mapi `prilozi` nalaze se svi navedeni prilozi, uključujući i cjelovit tekst Diplomskog rada (`diplomski rad AP.pdf`):

1. Datoteka `bio.sql` s izvornom shemom baze podataka.
2. Datoteka `bio-data.sql` s podacima oblikovanim prema `bio` shemi.
3. Datoteka `bio-complete.sql` s podacima oblikovanim prema `bio` shemi, te s pohranjenim postupcima za analizu podataka.
4. Tekstualne datoteke s cjelovitim upitima za kreiranje pohranjenih postupaka, unutar mape `upiti`. Svaka datoteka imenovana je prema odgovarajućem pohranjenom postupku, a sveukupno ih ima 11.
5. Datoteka `grafički_prikaz.xlsm` za prikaz rezultata u Microsoft Excel-u uz pomoć MySQL for Excel dodatka. Sadrži izvorne rezultate pohranjenih postupaka koji su korišteni kao primjeri, pri čemu svaki ima odgovarajući radni list.

IZJAVA O IZVORNOSTI

Izjavljujem da je ovaj diplomski rad izvorni rezultat mojeg rada te da se u njegovoj izradi nisam koristio/la drugim izvorima, osim onih koji su u njemu navedeni.

Andrea Plec

Ime i prezime studenta