Analiza "Million Songs" skupa podataka

Seminarski rad u okviru kursa
Istraživanje podataka
Matematički fakultet

Milana Kovačević
Ivan Ristović

jun 2018.

Sažetak

U ovom radu su dati rezultati istraživanja skupa podataka *Million Songs Dataset*. Nakon kratkog opisa strukture samog skupa, opisan je način na koji je on obradjen kako bi se prilagodio korišćenim alatima. Uočene su značajne zavisnosti izmedju atributa. Neke od ovih informacija su dobijene vizuelizacijom skupa atributa, dok su druge dobijene kao izlazi odredjenih algoritama. Postupak analize podataka je detaljno opisan, a najbolje rezultate analize je dao algoritam *Apriori*.

Sadržaj

1 Opis skupa podataka

The Million Song Dataset [8] je skup od milion slogova koji sadrže informacije o popularnim pesmama. S obzirom da ovaj skup preveliki za udoban rad, ograničićemo se na podskup od deset hiljada slogova izdvojen od strane autora originalnog skupa.

Svaki slog pomenutog skupa podataka sadrži informacije o jednoj pesmi: detalje o izvodjaču, segmentima, tempu kao i ID pesme na raznim online servisima (*Echo Nest* [6], *7digital* [1], *MusicBrainz* [4] i *PlayMe* [5]). Detaljne informacije o atributima se mogu videti na slici 3.1. Primer jednog sloga se može videti u dodatku A.

Skup u svojoj originalnoj formi je organizovan u *HDF5* format [9]. Mi ćemo izdvojiti informacije iz datog modela i podatke organizovati u CSV format, zarad lakšeg ubacivanja u alate koje ćemo opisati kasnije. Ova transformacija je izvršena korišćenjem Python skripti iz MSongsDB repozitorijuma [3], modifikovanih za naše potrebe. Kompletne skripte se mogu naći u dodatku B.

2 Korišćeni alati

Za obradu podataka, korišćeni su alati *Knime Analytics Platform* [11] i *IBM SPSS Modeler* [10]. *IBM SPSS Modeler* je pretežno korišćen za vizuelizaciju, dok je *KNIME AP* korišćen za manipulisanje podacima, vizuelizaciju i primenu algoritama. Jupyter svesku [2] smo koristili za dodatnu vizuelizaciju podataka.

3 Preprocesiranje i vizualizacija podataka

U ovom odeljku ćemo pokušati da čitaoca upoznamo sa skupom podataka. Uz vizualne prikaze raznovrsnosti skupa i analize njegovih specifičnosti došli smo do bitnih zaključaka koji su kasnije uticali na dalje istraživanje skupa i njegovih karakteristika.

Pre samog preprocesiranja podataka koje je neophodno za istraživanje, izvršili smo analizu statističkih podataka dobijenih na osnovu skupa. Dobijene statistike se mogu videti u dodatku C.

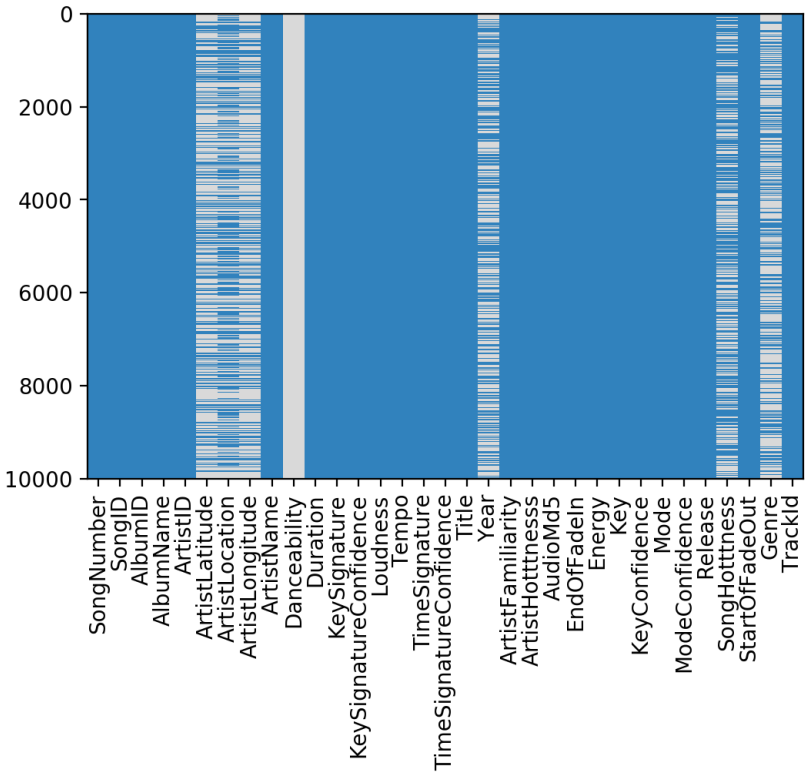
Neki od atributa koji su vizuelizovani kasnije u ovom odeljku nisu u potpunosti prisutni u skupu, tako da je analiza takvih atributa radjena samo nad slogovima gde nema nedostajućih vrednosti za te atribute.

Jedna od opštih transformacija je nad atributom koji sadrži informacije o žanru. Žanr je podatak koji je originalno dat kao niz niski. Medjutim, sadržaj ovog niza nije tačno definisan, već on ponekad u sebi sadrži čitavu rečenicu koja opisuje žanr. Jednostavna, a neophodna transformacija je bila da iz ovog niza izbacimo pojavljivanje reči *and*, čije je često pojavljivanje remetilo rezultate.

Atribut	Tip podatka	Kratki opis
analysis sample rate	float	učestalost uzorkovanja
artist 7digitalid	int	*7digital* ID izvodjača ili -1
artist familiarity	float	algoritamska aproksimacija
artist hotttnesss	float	algoritamska aproksimacija
artist id	string	*Echo Nest* ID izvodjača
artist latitude	float	geografska širina
artist location	string	lokacija autora
artist longitude	float	geografska dužina
artist mbid	string	MusicBrainz ID izvodjača
artist mbtags	array string	niz MusicBrainz tagova
artist mbtags count	array int	broj MusicBrainz tagova
artist name	string	ime autora
artist playmeid	int	*PlayMe* ID izvodjača ili -1
artist terms	array string	niz *Echo Nest* tagova
artist terms freq	array float	frekvencije *Echo Nest* tagova
artist terms weight	array float	težina *Echo Nest* tagova
audio md5	string	MD5 heš kod audio zapisa
bars confidence	array float	pouzdanost takta
bars start	array float	niz početaka taktova
beats confidence	array float	pouzdanost ritma
beats start	array float	niz početaka ritmova
danceability	float	algoritamska aproksimacija
duration	float	trajanje audio zapisa (u sekundama)
end of fade in	float	vreme u odnosu na pocetak u kom prestaje fade-in efekat (u sekundama)
energy	float	algoritamska aproksimacija energije pesme od strane slušaoca
key	int	tonalitet u kojem je audio zapis
key confidence	float	pouzdanost tonaliteta
loudness	float	prosečna jačina (u dB)
mode	int	mod - dur ili mol
mode confidence	float	pouzdanost moda
release	string	ime albuma
release 7digitalid	int	*7digital* ID albuma ili -1
sections confidence	array float	niz pouzdanosti stihova
sections start	array float	počeci stihova
segments confidence	array float	niz pouzdanosti segmenata
segments loudness max	array float	nig maksimalnih jačina unutar segmenata (u dB)
segments loudness max time	array float	niz vremena dostizanja maksimalne jačine unutar segmenata
segments loudness max start	array float	niz jačina na počecima segmenata
segments pitches	2D array float	niz jačina po segmentima, jedna vrednost za svaku notu
segments start	array float	počeci segmenata
segments timbre	2D array float	informacije o teksturi ($MFCC + PCA$)
similar artists	array string	niz *Echo Nest* sličnih izvodjača
song hotttnesss	float	algoritamska aproksimacija
song id	string	*Echo Nest* ID pesme
start of fade out	float	vreme u odnosu na pocetak u kom počinje fade-out efekat (u sekundama)
tatums confidence	array float	pouzdanost najmanjih elemenata ritma
tatums start	array float	niz najmanjih elemenata ritma
tempo	float	procenjen tempo (u BPM)
time signature	int	procenjen broj ritmova u taktu, npr. 4
time signature confidence	float	pouzdanost procene broja ritmova u taktu
title	string	naziv pesme
track id	string	*Echo Nest* ID pesme
track 7digitalid	int	ID *7digital* ID pesme ili -1
year	int	godina izdavanja uzeta sa *MusicBrainz* ili 0

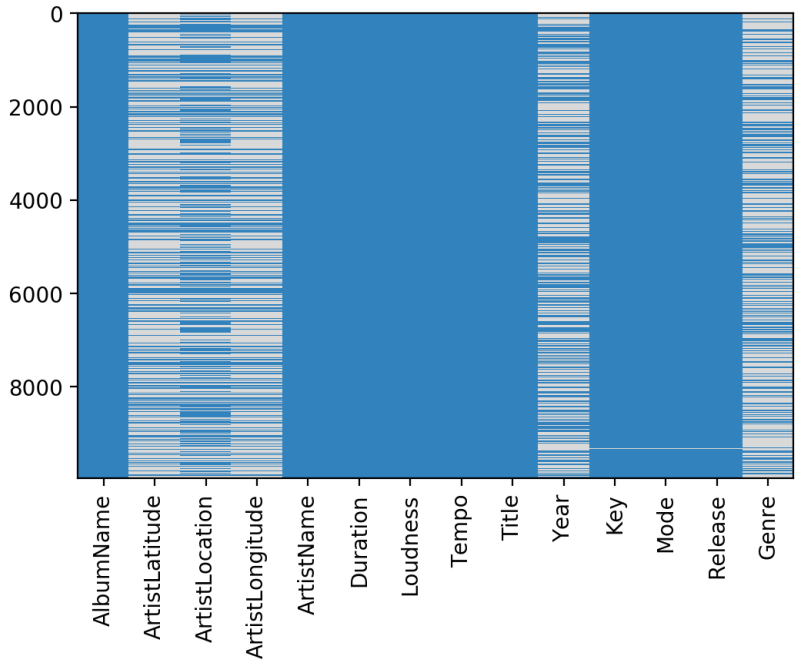
Slika 3.1: Svi atributi prisutni u *The Million Song Dataset* skupu podataka

Vizuelizacija originalnog skupa podataka je prikazana na slici 3.2. Nedostajuće vrednosti su prikazane belom bojom, dok su plavom bojom predstavljene postojuće vrednosti atributa.



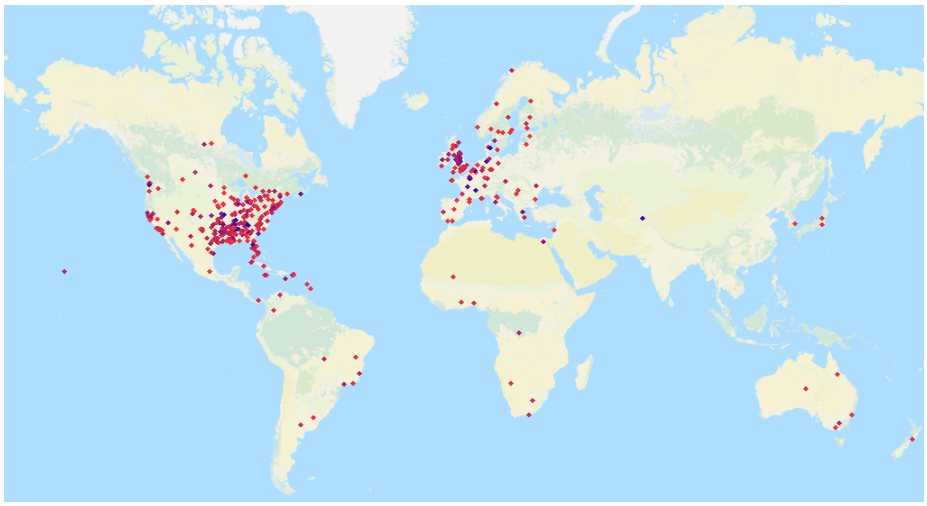
Slika 3.2: Nedostajuće vrednosti u originalnom skupu podataka

Sa prikaza skupa podataka se vidi da atribut koji opisuje plesnu moć pesme *eng. dancability*, ni u jednom slučaju nema postavljenu vrednost, te da je on neupotrebljiv. Takodje, godina, lokacija autora, geografska širina, geografska dužina i žanr su atributi koji u velikom broju slučajeva nemaju vrednost. Medjutim, zbog njihove važnosti, mi ćemo svoje istraživanje vršiti nad onim slogovima za koje su ove vrednosti poznate. Nakon izdvajanja relevantnih atributa za naše istraživanje, pripremljen skup podataka je prikazan na slici 3.3.



Slika 3.3: Izdvojeni atributi korišćeni u istraživanju

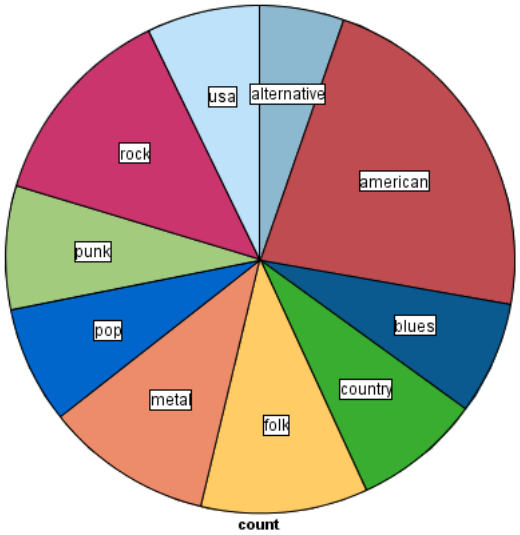
Geografska rasprostranjenost autora čije su se pesme našle u skupu podataka se može videti na slici 3.4. Različite boje predstavljaju vizuelizaciju godine izdavanja pesme - gradijentni prelaz od plave (1950) do crvene (2010).



Slika 3.4: Geografska rasprostranjenost autora

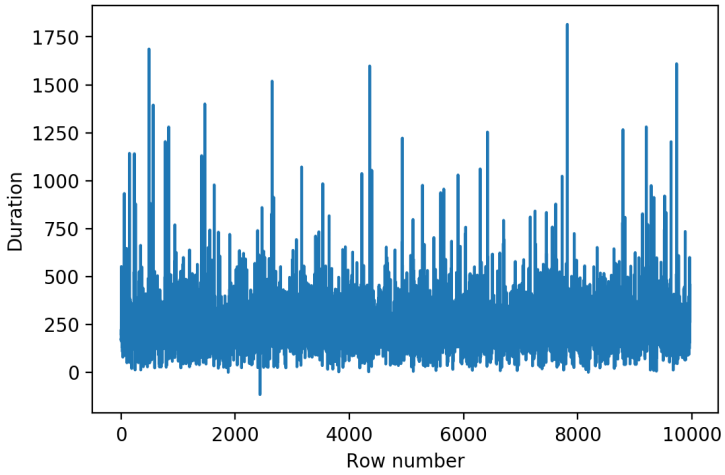
Spisak najzastupljenijih zanrova u skupu se može videti na slici 3.5. Žanr je atribut sa velikom stopom nedostajućih vrednosti, uz dodatni

problem rečenica prisutnih u nizu (podsećamo na problem naveden prilikom preprocesiranja, poglavlje 3), tako da je analiza vršena nad veoma ograničenim skupom od oko tri hiljade slogova. Smatramo, da su ovi rezultati u velikoj meri slični rezultatima koji bi se dobili da je potpuni skup analiziran - ukoliko ne bi bilo nedostajućih vrednosti za atribut žanr.



Slika 3.5: Zastupljenost žanrova

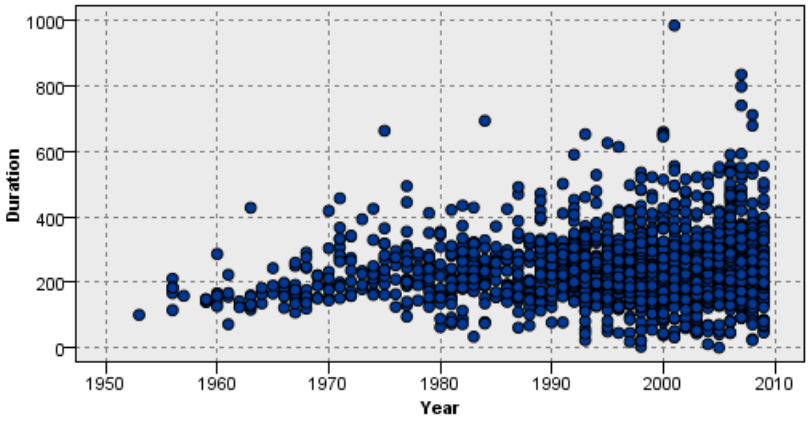
Vrednosti atributa *duration* prikazane su na grafiku 3.6. Postoji jedna pesma koja ima negativnu dužinu pa smo najpre izbacili ovu pesmu iz skupa koji obradjujemo. Kako postoje izuzetno duge pesme, nismo postavili gornje ograničenje za trajanje pesme.



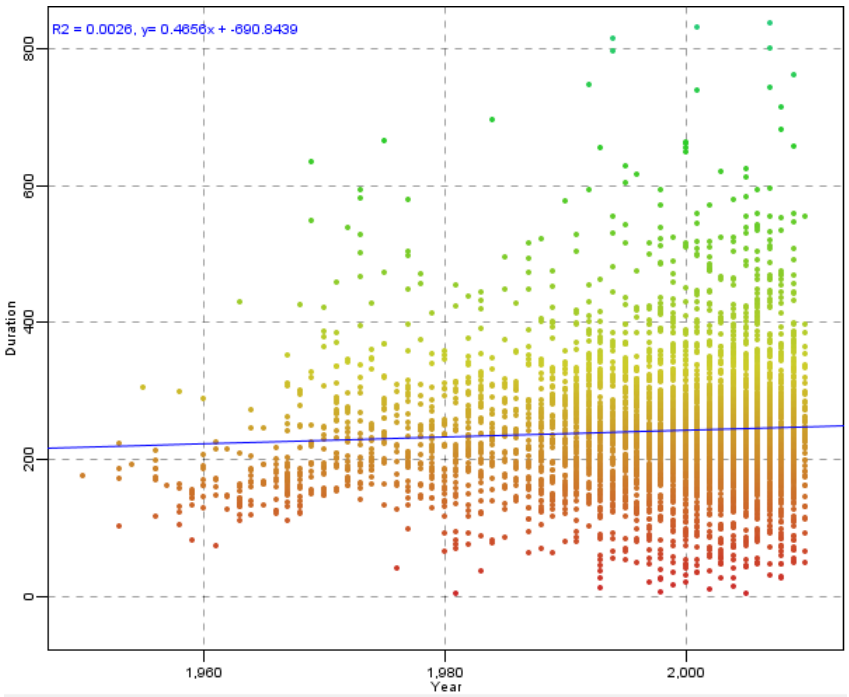
Slika 3.6: Vrednosti atributa *duration*

Grafik zavisnosti godine i trajanja pesme se može videti na slici 3.7.

Jedan zanimljiv zaključak koji se nameće, je da se prosečno trajanje pesama povećava kroz vreme, sa razlikom od oko 20 sekundi u odnosu na 50-te godine prošlog veka. Detaljniji prikaz promene proseka trajanja se može videti na slici 3.8. Takodje, jasno je da se i raznovrsnost pesama mnogo veća danas - prisutne su i veoma kratke ali i veoma dugačke pesme.



Slika 3.7: Odnos godine izdavanja i dužine pesme

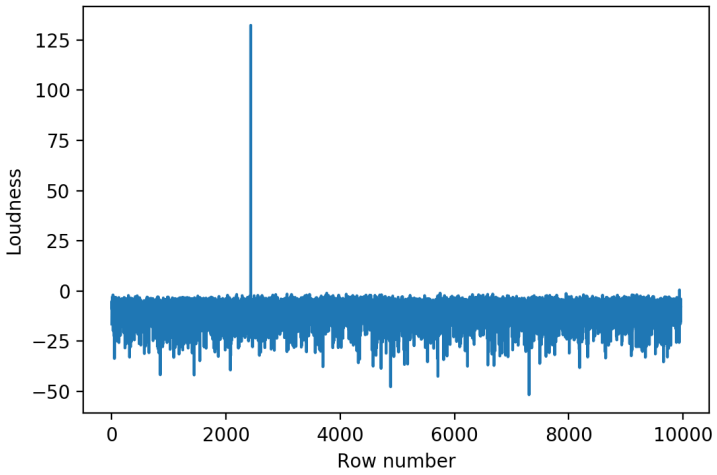


Slika 3.8: Promena proseka dužine pesama kroz vreme

Prosečna dužina pesama po državama je prikazana na slici 3.9. Nažalost, nisu svi slogovi imali informaciju o nazivu države. Iz ovog razloga, dobijenim rezultatima ne treba pridavati veliki značaj.

Slika 3.9: Prosečna dužina pesama na raznim lokacijama

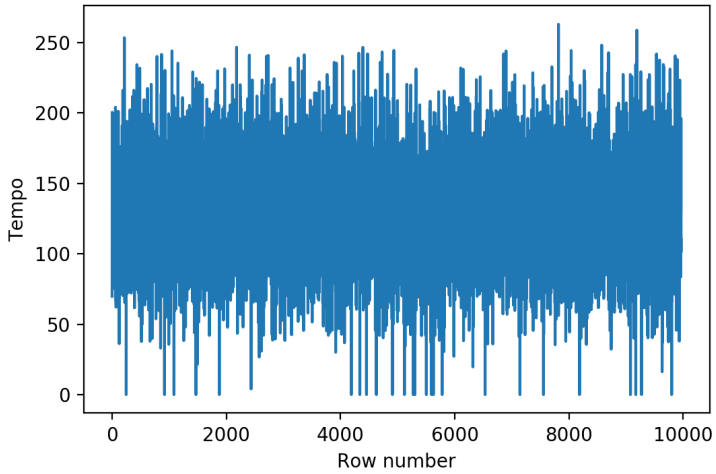
Na slici 3.10 prikazane su vrednosti atributa *loudness*. Ove vrednosti su dobijene na osnovu njihovih proseka tokom trajanja pesme, kao i nekim dodatnim transformacijama. Vise o ovome se može nać na zvaničnom sajtu skupa [8]. Sa grafika se vidi da za atribut *loudness* postoje vrednosti koje izuzetno odstupaju od proseka. Iz toga razloga smo prilikom istraživanja koristili pesme sa vrednošću atributa iz skupa $[-40, 0]$.



Slika 3.10: Vrednosti atributa *loudness*

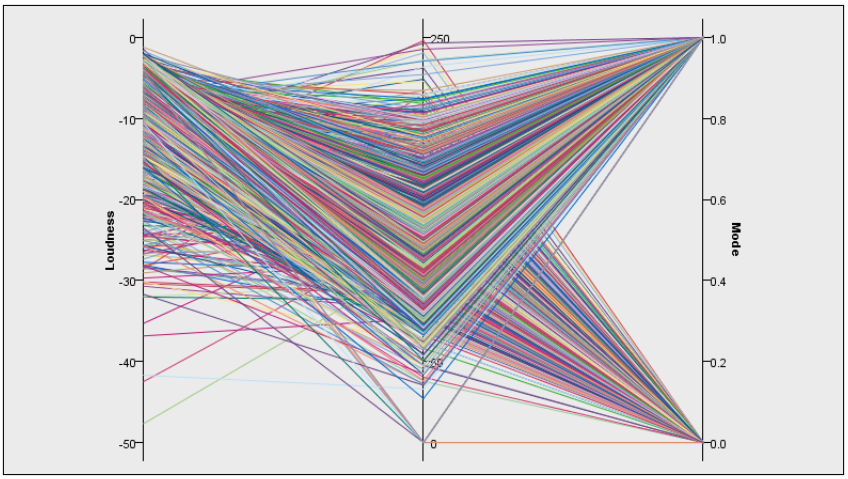
Na slici 3.11 prikazane su vrednosti atributa *tempo*. Ovaj atribut sadrži nekoliko vrednosti koje su dosta ispod proseka. Medjutim, postoji vero-

vatnoća da su njima predstvaljene nadprosečno spore pesme te zbog toga nismo vršili nikakvo odsecanje.



Slika 3.11: Vrednosti atributa *tempo*

S obzirom da ćemo atribute *loudness*, *tempo* i *mode* [1] koristiti u daljoj analizi (predikcija žanra pesme) u poglavlju 6, prikazaćemo i grafički prikaz njihove zavisnosti na slici 3.12. Može se videti da iako pesme oba moda uzimaju raznolike vrednosti za tempo i jačinu, moguće je uočiti da se pesme sa ekstremnim vrednostima za tempo lepše dele po modu. Takodje, mnogo tihe pesme obično pripadaju jednom modu, što je i za očekivati jer molske pesme su nekada jako tihe.



Slika 3.12: Prikaz tempa, jačine i moda (dur ili mol)

[1] loudness, tempo i mode predstavljaju prosečnu jačinu, tempo i mod (dur ili mol), redom.

4 Pravila pridruživanja

Uočavanje zavisnosti izmedju atributa je uradjeno korišćenjem algoritma *Apriori*. Najpre, kao što je opisano u poglavlju 3, bilo je potrebno eleminisati slogove koji sadrže nedostajuće vrednosti na relevantnim atributima. Sledeći korak je zavisio od toga koje su zavisnosti analizirane. Pokušali smo da pronadjemo zavisnosti izmedju zanrova, drzava i vremena kreiranja pesme. Rezultati rada Apriori algoritma slede u nastavku. Svi rezultati prikazani ispod su samo deo čitavog skupa rezultata, izabrani jer smo smatrali da su najinteresantnija.

4.1 Zavisnosti izmedju žanrova

Nakon prilagodjavanja atributa *žanr* algoritmu, dobijeni su rezultati prikazani na slici 4.1. Na njoj je prikazano 10 najizražajnijih pravila sortirana prvo po podršci, a zatim po lift meri, opadajuće. Ovi rezultati su relevantni iako su dobijeni na relativno malim skupom slogova.

Podrška	Pouzdanost	Lift mera	Glava pravila	Telo pravila
0.2646	0.7404	1.5567	rock	pop
0.2646	0.5563	1.5567	pop	rock
0.2248	0.9930	2.0878	rock	classic
0.2163	0.8176	3.6114	classic	pop, rock
0.2163	0.9623	2.6926	pop	rock, classic
0.2163	1.0000	2.1026	rock	pop, classic
0.1612	0.9967	2.0957	rock	indie
0.1405	0.9170	5.9225	british	uk
0.1405	0.9075	5.9225	uk	british
0.1373	0.5582	1.1736	rock	american, classic

Slika 4.1: Rezultati Apriori algoritma koji pokazuju zavisnost medju žanrovima

4.2 Zavisnost žanra od decenije

Istraživanje koji je žanr bio zastupljen u nekoj deceniji je dalo nočekivane rezultate. Na tabeli 4.2 prikazana su dobijena pravila zavisnosti sortirana prvo po podršci, a zatim po lift meri, opadajuće. Kako se ranije pokazalo na slici 3.7, najveći deo skupa obradjenih podataka pripada dvehiljaditim godinama. Zbog ovoga se dobijeni rezulati koncentrišu na ovoj deceniji. Korišenje većeg i raznovrsnijeg skupa podataka bi doneo drugačije rezultate.

Podrška	Pouzdanost	Lift mera	Glava pravila	Telo pravila
0.0716	0.8133	1.6369	00s	pop, chart
0.0525	0.7857	1.5815	00s	rnb, hop, hall, dance, hip
0.0764	0.7385	1.4864	00s	dance
0.0615	0.6824	1.3734	00s	rnb
0.0615	0.6667	1.3419	00s	hop hip
0.0541	0.6000	1.2077	00s	metal
0.0530	0.5917	1.1910	00s	rock, alternative
0.0578	0.5619	1.1309	00s	alternative
0.0896	0.5541	1.1153	00s	indie
0.0891	0.5526	1.1123	00s	rock, indie

Slika 4.2: Rezultati Apriori algoritma koji pokazuju zavisnost izmedju žanrova i decenija tokom koje su bili popularni

4.3 Zavisnost žanra od lokacije

Lokacija autora je atribut čije ja vrednost često nepostojuća. A ukoliko jeste, iste lokacije su na različitim pesmama drugačije napisane. Na primer, jedna pesma ima lokaciju *London*, a druga *London, UK*. Ovakvi podaci su doveli do loših rezultata istraživanja - slika 4.3. Nazalost, jedina pravila koja imaju dovoljno veliku pouzdanost su oblika:

$$\{uk,\ british\} \rightarrow England$$

Ovo nam nije od preteranog značaja jer je očekivano da se u Engleskoj slušaju pesme žanra *british*.

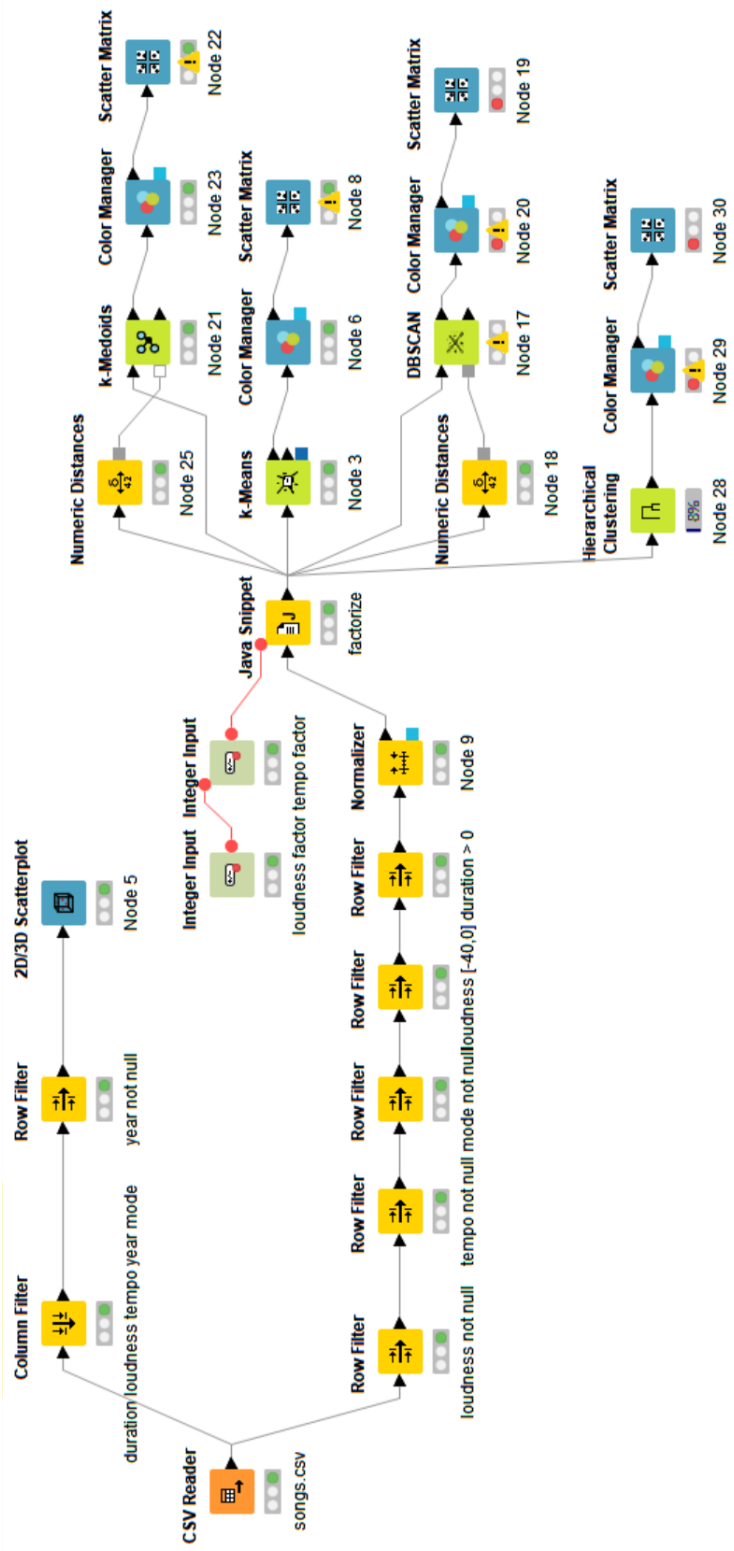
Podrška	Pouzdanost	Lift mera	Glava pravila	Telo pravila
0.0525	0.8684	7.6535	England	pop, rock, classic, uk, english, british
0.0541	0.7786	6.8621	England	pop, rock, classic, uk, british
0.0546	0.8729	7.6928	England	pop, rock, classic, and, english
0.0546	0.8729	7.6928	England	rock, classic, uk, english, british
0.0557	0.8268	7.2865	England	pop, and, uk, english, british

Slika 4.3: Rezultati Apriori algoritma koji pokazuju zavisnost izmedju žanrova i lokacije

5 Klasterovanje

Isprobali smo različite algoritme klasterovanja. Početna ideja je bila da na taj način izdvojimo grupe pesama koje su slične i prema tome formiramo liste pesama. Ovu ideju smo sproveli algoritmima K-sredina, K-medioida, DBSCAN i algoritmom hijerarhijskog klasterovanja. Medjutim, dobijeni rezultati nisu bili zadovoljavajući te nećemo ulaziti u njihovu dublju analizu.

Na slici 5.1 prikazan je KNIME workflow koji smo koristili za isprobavanje algoritama.



Slika 5.1: KNIME workflow za algoritme klasterovanja

6 Klasifikacija

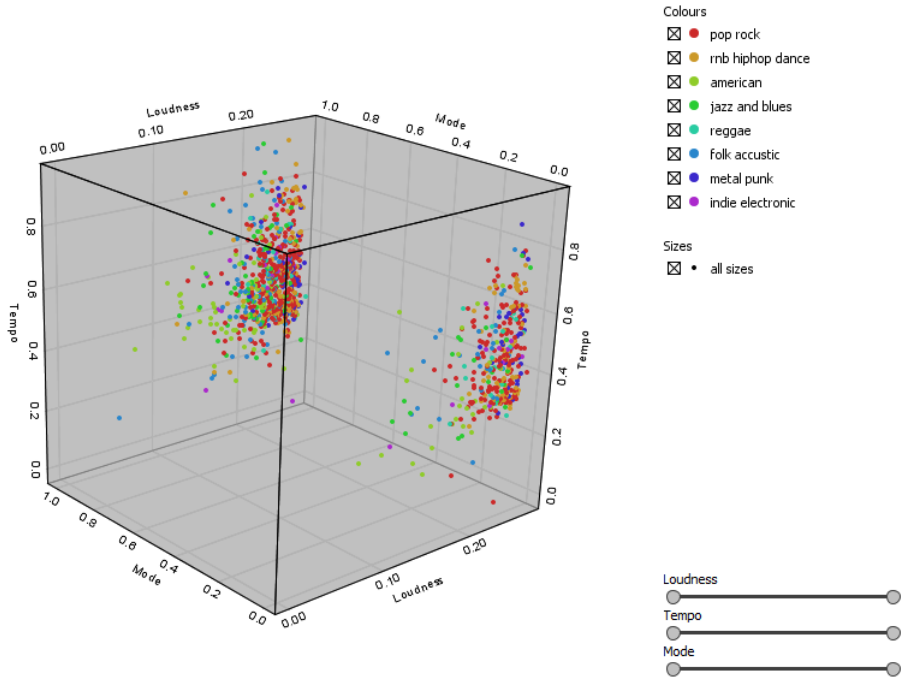
Jedan od glavnih problema u čitavom skupu je veliki broj nedostajućih vrednosti za žanr. Stoga smo pokušali da napravimo klasifikator koji će klasifikovati instance sa nepoznatim vrednostima za žanr, treniran nad onim podacima gde su te informacije dostupne.

Prvi problem na koji smo naišli su nestandarne vrednosti za žanr (videti poglavlje 3), stoga smo korišćenjem jednostavnih transformacija izvukli slogove sa nedvosmislenom vrednošću za žanr, a eliminisali one koji su za žanr imali vrednosti koje nisu bile od značaja za analizu (neki slogovi su imali više različtih žanrova). Takodje smo neke slične žanrove spojili u jedan, zarad jednostavnijeg rada (na primer *jazz* i *blues* se često pojavljuju zajedno ih ima smisla posmatrati kao jedan žanr).

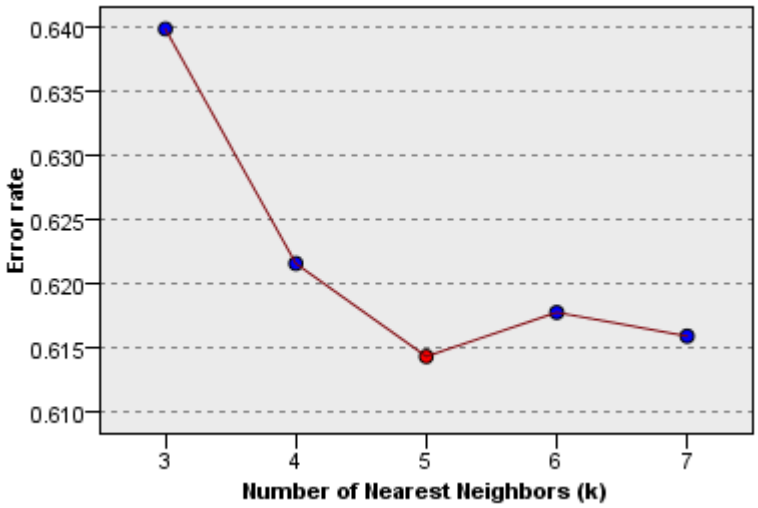
Za predikciju vrednosti žanra smo koristili atribite *loudness*, *tempo* i *mode*. Nažalost, iz njihove prostorne rasprostranjenosti se vidi da ne postoji jednostavni separator instanci raznih klasa - slika 6.1.

6.1 Klasifikacija metodom K najbližih suseda

Prva stvar koju smo odlučili da isprobamo bila je *KNN* algoritam u nadi da će male grupe pesama istog žanra lepo klasifikovati bliske instance. Očekivano, najbolji model koji smo dobili je imao preciznost od 52.65%. Na slici 6.2 se može videti kako se greška menjala sa različitim vrednostima za parametar k.

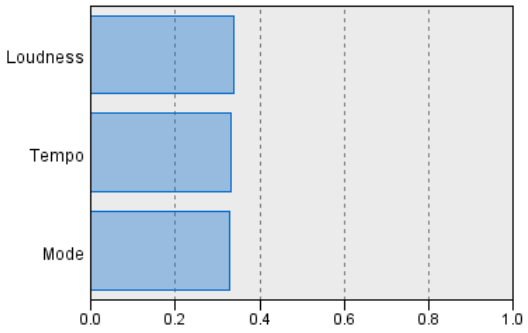


Slika 6.1: 3D prikaz pesama različitih žanrova u prostoru sa koordinatama *loudness*, *tempo* i *mode*



Slika 6.2: Greške dobijene za kreirane modele k parametrom u opsegu $[3, 7]$

Ono što nismo očekivali je jednaka važnost atributa prilikom pravljenja klasifikatora - slika 6.3. Intuicija nekako kaže da su jače pesme podložnije da budu žanra *metal*, ali rezultati su pobili tu intuiciju.



Slika 6.3: Važnost atributa prilikom klasifikacije

Sledći korak koji smo uradili bila je da isptobamo isti algoritam u alatu Knime [11] (prethodni zaključci su dobijeni u alatu SPSS Modeler [10]). Ovo nam je pružilo mogućnost da ručno pridamo atributima veću ili manju važnost. Iterativnim postupkom smo došli do faktora kojima je potrebno pomnožiti vrednosti *loudness* i *tempo* kako bi se povećala preciznost klasifikatora. Ovo transformacija utiče na izračunavanje različitih euklidskih rastojanja izmedju pesama i dovela je do poboljšanja preciznosti na 58.6%. Korišćenje Menhetn rastojanja je dovelo do neznatno boljih rezultata, sa preciznošću od 58.8%

Dodatno, jedno od mogućih unapredjenja bi bila da se u obzir uzme i atribut *ModeConfidence*. Informacija o pouzdanosti vrednosti atributa koji predstavlja tonalitet pesme bi mogla da dovede do bolje klasifikacije.

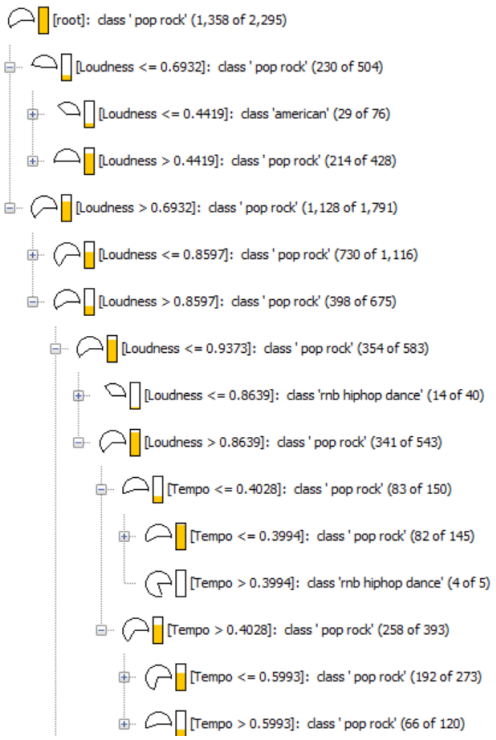
6.2 Klasifikacija stablom odlučivanja

Predvidjanje žanra koristeći stablo odlučivanja je takodje dalo lose rezultate. Preciznost dobijenog modela je 44.6%, te je greška 55.4%. Matrica konfuzije je prikazana na slici 6.4.

Prediction	pop rock	hiphop rnb dance	metal punk	USA	reggae	jazz blues	accustic folk	indie electro
pop rock	225	53	18	8	5	17	13	0
hiphop rnb dance	48	10	11	0	1	3	1	0
metal punk	32	11	7	0	0	0	1	0
american	14	0	1	8	0	1	2	0
reggae	7	2	0	0	0	1	3	0
jazz blues	21	2	1	1	1	4	1	0
folk accustic	18	4	2	1	1	3	2	0
indie electro	6	1	0	1	0	0	0	0

Slika 6.4: Matrica konfuzije dobijena klasifikacijom stablom odlučivanja

Na slici 6.5 prikazano je dobijeno sablo odlučivanja. Kako više od polovine pesama iz skupa pripada žanru *pop rock*, a vrednosti atributa nisu dobre za klasifikaciju, stablo daje loše rezultate.



Slika 6.5: Stablo odlučivanja za atrbut žanr

7 Zaključak

Dobijeni rezultati se oslanjaju na podskup skupa podataka koji u sebi sadrži milion pesama. Korišćenje celog skupa podataka bi donelo još pouzdanije rezultate. Medjutim, i sam podskup podataka je bio dovoljan da se uoče prethodno navedeni zaključci.

Rezultati imaju praktičnu primenu. Uočavanje zavisnosti muzičkih žanrova i godina njihove popularnosti, ili njihove popularnosti na različitim podnevljima bi se moglo iskoristiti za automatsko generisanje lista pesama. Ovo bi predstavljao zanimljiv i koristan dodatak za muzički plejer.

Sajtovi kao što su YouTube [7] koriste prethodno prikupljene podatke o pesmama. Njih, u kombinaciji sa saznanjem o korisnikovim prethodno preslušanim pesmama, koristi kako bi korisniku pružio što bolje preporuke za sledeću pesmu, a samim tim ga i zadržao na sajtu. Implementacija sličnog, ali dosta jednostavnijeg, sistema bila bi moguća dobijenim rezultatima koji su predstavljeni u ovom radu.

Literatura

[1] 7digital. on-line at:
https://www.7digital.com/.

[2] Jupyter notebook. on-line at:
http://jupyter.org/.

[3] Million Song Database GitHub repository. on-line at:
https://github.com/tbertinmahieux/MSongsDB.

[4] MusicBrainz. on-line at:
https://www.musicbrainz.org/.

[5] Play.me. on-line at:
https://www.playme.com/.

[6] The Echo Nest. on-line at:
http://the.echonest.com/.

[7] YouTube. on-line at:
https://www.youtube.com/.

[8] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The Million Song Dataset, 2011. on-line at: https://labrosa.ee.columbia.edu/millionsong/.

[9] The HDF Group. The HDF5 data model, 2011. on-line at: https://support.hdfgroup.org/HDF5/.

[10] IBM. IBM SPSS Modeler. on-line at:
https://www.ibm.com/analytics/data-science/predictive-analytics/spss-statistical-software.

[11] KNIME AG. KNIME Analytics Platform. on-line at:
https://www.knime.com/knime-analytics-platform.

Dodatak A Primer sloga

```
analysis_sample_rate: 22050
artist_7digitalid: 61424
artist_familiarity: 0.5467275539627645
artist_hotttnesss: 0.3861804160792181
artist_id: ARE26EG1187B990AEF
artist_latitude: 51.77045
artist_location: Essex, England
artist_longitude: 0.64255
artist_mbid: de212b3a-2f54-4def-a13d-5a877bfaeaf7
artist_mbtags: shape = (6,)
artist_mbtags_count: shape = (6,)
artist_name: Sunscreem
artist_playmeid: 19156
artist_terms: shape = (44,)
artist_terms_freq: shape = (44,)
artist_terms_weight: shape = (44,)
audio_md5: c2f7f92e66d18e86af3752478d3be966
bars_confidence: shape = (123,)
bars_start: shape = (123,)
beats_confidence: shape = (497,)
beats_start: shape = (497,)
danceability: 0.0
duration: 232.4371
end_of_fade_in: 0.0
energy: 0.0
key: 11
key_confidence: 0.625
loudness: -8.955
mode: 0
mode_confidence: 0.558
release: Looking At You: The Club Anthems
release_7digitalid: 196929
sections_confidence: shape = (6,)
sections_start: shape = (6,)
segments_confidence: shape = (1045,)
segments_loudness_max: shape = (1045,)
segments_loudness_max_time: shape = (1045,)
segments_loudness_start: shape = (1045,)
segments_pitches: shape = (1045, 12)
segments_start: shape = (1045,)
segments_timbre: shape = (1045, 12)
similar_artists: shape = (100,)
song_hotttnesss: nan
song_id: SOICLQB12A8C13637C
start_of_fade_out: 232.437
tatums_confidence: shape = (993,)
tatums_start: shape = (993,)
tempo: 130.201
time_signature: 4
time_signature_confidence: 0.0
title: Exodus
track_7digitalid: 2140010
track_id: TRBBBLA128F424E963
year: 1995
```

Slika A.1: Primer sloga

Dodatak B Konverzija iz HDF5 u CSV format

```
"""
Alexis Greenstreet (October 4, 2015) University of Wisconsin-Madison
"""
class Song:
    songCount = 0

    def __init__(self, songID):
        self.id = songID
        Song.songCount += 1
        self.albumID = None                          # string
        self.albumName = None                        # string
        self.artistFamiliarity = None                # float
        self.artistHottnesss = None                  # float
        self.artistID = None                         # string
        self.artistLatitude = None                   # float
        self.artistLocation = None                   # string
        self.artistLongitude = None                  # float
        self.artistName = None                       # string
        self.audioMd5 = None                         # string
        self.danceability = None                     # float
        self.duration = None                         # float
        self.endOfFadeIn = None                      # float
        self.energy = None                           # float
        self.genre = None                            # string
        self.genreList = []                          # list of strings
        self.key = None                              # int
        self.keyConfidence = None                    # float
        self.keySignature = None                     # float
        self.keySignatureConfidence = None           # float
        self.loudness = None                         # float
        self.mode = None                             # int
        self.modeConfidence = None                   # float
        self.release = None                          # string
        self.songHotttness = None                    # float
        self.songId = None                           # string
        self.startOfFadeOut = None                   # float
        self.tempo = None                            # float
        self.timeSignature = None                    # int
        self.timeSignatureConfidence = None          # float
        self.title = None                            # string
        self.trackId = None                          # string
        self.year = None                             # int
```

Slika B.1: Klasa korišćena za deserijalizaciju podataka.

```
"""
Alexis Greenstreet (October 4, 2015) University of Wisconsin-
    Madison
"""
outputFile1 = open('SongCSV.csv', 'w')
csvRowString = ""

csvRowString = ("SongID,AlbumID, ...")
    csvAttributeList = re.split('\W+', csvRowString)
for i, v in enumerate(csvAttributeList):
    csvAttributeList[i] = csvAttributeList[i].lower()
outputFile1.write("SongNumber,");
outputFile1.write(csvRowString + "\n");
csvRowString = ""

#####################################################
#Set the basedir here, the root directory from which the
    search
basedir = "/home/m/Documents/MillionSongSubset/data"
ext = ".h5"
#####################################################

csvRowStringTotal = ""

for root, dirs, files in os.walk(basedir):
    files = glob.glob(os.path.join(root,'*'+ext))
    for f in files:
        print f

        songH5File = hdf5_getters.open_h5_file_read(f)
        song = Song(str(hdf5_getters.get_song_id(songH5File)))

        song.artistID = str(hdf5_getters.get_artist_id(
    songH5File))
        # Isto za ostala polja

        artistMbtags = np.array(hdf5_getters.get_artist_mbtags
    (songH5File))
        song.genre = ' | '.join(artistMbtags)

        csvRowString += str(song.songCount) + ","
        csvRowString += song.id + ","
        # Isto za ostala polja

        csvRowString += song.trackId + ","
        csvRowString += song.genre + "\n"
        csvRowStringTotal += csvRowString
        csvRowString = ""

        songH5File.close()

outputFile1.write(csvRowStringTotal)
outputFile1.close()
```

Slika B.2: Uprošćena verzija programa korišćenog za konvertovanje iz HDF5 u CSV format.

Dodatak C Statistika skupa podataka

Column	ArtistLatitude	Duration	Loudness	Tempo	Year	Energy	Mode
Min	-41.281	-114	-51.643	0	1	0	0
Max	151246	1815.222	261.538	262.828	2010	0.399	1
Mean	94.001	238.379	-10.444	122.868	1996.381	7.614E-05	0.691
Std. deviation	2641.518	113.117	6.217	35.216	42.910	0.005	0.462
Variance	6977616.187	12795.546	38.644	1240.182	1841.236	2.893E-05	0.21363
Skewness	52.760	3.20372	8.711	0.404	-43.0545	70.828	-0.826
Kurtosis	2922.508	24.813	396.106	0.487	1997.926	5029.175	-1.319
Overall sum	352221.307	2375926.299	-104109.426	1224748.712	9317112	0.758	6876
No. missings	6251	31	30	30	331	44	44
No. NaNs	0	0	0	0	0	0	0

Slika C.1: Statistike nama relevantnih atributa iz skupa