SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

ZADANIE 2 : ANALÝZA DÁT A REGRESORY
SEMINÁRNA PRÁCA

2021 Ladislav Rajcsányi

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

ZADANIE 2 : ANALÝZA DÁT A REGRESORY
SEMINÁRNA PRÁCA

Študijný program:	Aplikovaná informatika
Predmet:	I-SUNS – Strojové učenie a neurónové siete
Prednášajúci:	prof. Dr. Ing. Miloš Oravec
Cvičiaci:	Ing. Zuzana Bukovčiková
	Ing. Vanesa Andicsová
	Ing. Dominik Sopiak, PhD.

Bratislava 2021 **Ladislav Rajcsányi**

Obsah

Zoznam obrázkov a tabuliek

Zoznam skratiek

API Application Programming Interface

ML Machine Learning

NaN Not a Number

NS Neurónová Sieť

Úvod

Hlavným cieľom tohto zadania je predpovedanie hlasitosti piesne pomocou rôznych regresných modelov. Na vypracovanie zadania sme použili poskytnuté dátové súbory, ktoré sú rozdelené na trénovacie a testovacie dáta. Trénovacie dáta použijeme na natrénovanie našich regresných modelov a v prípade potreby môžeme z nich vybrať aj validačné dáta, ktoré môžeme používať na monitorovanie úspešnosti na predtým nevidených dát počas fázy trénovania.

Dáta sú uložené v CSV súboroch. Trénovacie (spotify_train.csv) aj testovacie (spotify_test.csv) dáta majú 27 stĺpcov, pričom jednotlivé stĺpce reprezentujú jednotlivé vlastnosti piesne na streamovacej službe Spotify, ktoré sú nasledovné:

1. ID : ID služby Spotify pre skladbu.
2. ID Umelca (angl.: artist_id) : ID služby Spotify pre umelca.
3. Umelec (angl.: artist) : Meno umelca.
4. Názov (angl.: name) : Názov skladby.
5. Popularita (angl.: popularity) : Popularita skladby je hodnota medzi 0 a 100, pričom 100 je najpopulárnejšia. Obľúbenosť sa vypočítava pomocou algoritmu a z väčšej časti sa zakladá na celkovom počte prehrávaní skladby a na tom, ako nedávno sa tieto prehrávania uskutočnili. Vo všeobecnosti platí, že skladby, ktoré sa teraz veľa hrajú, budú mať vyššiu popularitu ako skladby, ktoré sa veľa hrali v minulosti. Duplicitné skladby (napr. tá istá skladba zo singla a albumu) sa hodnotia nezávisle. Popularita interpretov a albumov sa odvodzuje matematicky od popularity skladieb. Poznámka: hodnota popularity môže zaostávať za skutočnou popularitou o niekoľko dní: hodnota sa neaktualizuje v reálnom čase.
6. Dátum vydania(angl.: release_date) : Dátum prvého vydania albumu.
7. Trvanie v milisekundách (angl.: duration_ms) : Trvanie skladby v milisekundách.
8. Explicitné (angl.: explicit) : Či skladba obsahuje alebo neobsahuje explicitný text (true = áno, obsahuje; false = nie, neobsahuje alebo neznáme).
9. Tanečnosť (angl.: danceability) : Tanečnosť opisuje vhodnosť skladby na tanec na základe kombinácie hudobných prvkov vrátane tempa, stability rytmu, sily rytmu a celkovej pravidelnosti. Hodnota 0,0 je najmenej tanečná a 1,0 je najviac tanečná.

10. Energia (angl.: energy) : Energia je miera od 0,0 do 1,0 a predstavuje vnímanie intenzity a aktivity. Energické skladby sú zvyčajne rýchle, hlasné a hlučné. Napríklad death metal má vysokú energiu, zatiaľ čo Bachovo prelúdium má na stupnici nízke skóre. Medzi percepčné vlastnosti, ktoré prispievajú k tomuto atribútu, patria dynamický rozsah, vnímaná hlasitosť, farba zvuku, rýchlosť nástupu a všeobecná entropia.

11. Kľúč (angl.: key) : Kľúč, v ktorom sa skladba nachádza. Celé čísla sa mapujú na výšky tónov pomocou štandardnej notácie Pitch Class. Napr. 0 = C, 1 = C#, 2 = D atď.

12. **Hlasitosť (angl.: loudness) :** Celková hlasitosť skladby v decibeloch (dB). Hodnoty hlasitosti sú spriemerované pre celú stopu a sú užitočné na porovnanie relatívnej hlasitosti stôp. Hlasitosť je kvalita zvuku, ktorá je primárnym psychologickým korelátom fyzickej sily (amplitúdy). Hodnoty sa zvyčajne pohybujú v rozmedzí od -60 do 0 db. Zisťuje prítomnosť publika v nahrávke. Vyššie hodnoty živosti predstavujú zvýšenú pravdepodobnosť, že skladba bola vykonaná naživo. Hodnota nad 0,8 poskytuje veľkú pravdepodobnosť, že skladba je živá.

13. Mód / Režim (angl.: mode) : Mód označuje modalitu (dur alebo mol) skladby, typ stupnice, z ktorej je odvodený jej melodický obsah. Dúr je reprezentovaný hodnotou 1 a mol je 0.

14. Rečnosť (angl.: speechiness) : Funkcia Speechiness zisťuje prítomnosť hovorených slov v skladbe. Čím viac sa nahrávka podobá výlučne reči (napr. talk show, zvuková kniha, poézia), tým bližšie k hodnote 1,0 je hodnota atribútu. Hodnoty nad 0,66 opisujú skladby, ktoré sú pravdepodobne zložené výlučne z hovorených slov. Hodnoty medzi 0,33 a 0,66 opisujú skladby, ktoré môžu obsahovať hudbu aj reč, a to buď v častiach, alebo vrstve, vrátane takých prípadov, ako je rap. Hodnoty pod 0,33 s najväčšou pravdepodobnosťou predstavujú hudbu a iné skladby, ktoré nie sú podobné reči.

15. Akustickosť (angl.: acousticness) : Miera spoľahlivosti od 0,0 do 1,0, či je stopa akustická. Hodnota 1,0 predstavuje vysokú istotu, že stopa je akustická.

16. Inštrumentálnosť (angl.: instrumentalness) : Predpovedá, či skladba neobsahuje vokály. Zvuky 'Ooh' a 'Aah' sa v tomto kontexte považujú za inštrumentálne. Rapové skladby alebo skladby s hovoreným slovom sú jednoznačne 'vokálne'. Čím bližšie je

hodnota inštrumentálnosti k hodnote 1,0, tým väčšia je pravdepodobnosť, že skladba neobsahuje vokálny obsah. Hodnoty nad 0,5 majú predstavovať inštrumentálne skladby, ale dôvera je vyššia, keď sa hodnota blíži k 1,0.

17. Živosť (angl.: liveness) : Zisťuje prítomnosť publika v nahrávke. Vyššie hodnoty živosti predstavujú zvýšenú pravdepodobnosť, že skladba bola vykonaná naživo. Hodnota nad 0,8 poskytuje veľkú pravdepodobnosť, že skladba je živá.

18. Valencia (angl.: valence) : Miera od 0,0 do 1,0, ktorá opisuje hudobnú pozitívnosť skladby. Skladby s vysokou valenciou znejú pozitívnejšie (napr. šťastné, veselé, euforické), zatiaľ čo skladby s nízkou valenciou znejú negatívnejšie (napr. smutné, depresívne, nahnevané).

19. Tempo : Celkové odhadované tempo skladby v úderoch za minútu (BPM). V hudobnej terminológii je tempo rýchlosť alebo tempo danej skladby a odvodzuje sa priamo od priemerného trvania úderov.

20. Žánre Umelca (angl.: artist_genres) : Zoznam žánrov, s ktorými je umelec spojený. Ak ešte nie je zaradený, pole je prázdne.

21. Nasledovníci Umelca (angl.: artist_followers) : Informácie o nasledovníkoch umelca.

22. URL : URI služby Spotify pre skladbu.

23. ID Zoznamu Skladieb (angl.: playlist_id) : ID služby Spotify zoznamu skladieb.

24. Popis Zoznamu Skladieb (angl.: playlist_description) : Hodnota pre popis zoznamu skladieb, ako sa zobrazuje v klientoch Spotify a vo webovom rozhraní API.

25. Názov Zoznamu Skladieb (angl.: playlist_name) : Názov Zoznamu Skladieb.

26. URL Zoznamu Skladieb (angl.: playlist_url) : URL Zoznamu Skladieb.

27. Dotaz (angl.: query) : Vyhľadávací dotaz.

1 Použité technológie

1.1 Pandas

Pandas je rýchly, výkonný, flexibilný a ľahko použiteľný open source nástroj na analýzu a manipuláciu s údajmi, postavený na programovacom jazyku Python [1].

1.2 Matplotlib

Matplotlib je komplexná knižnica na vytváranie statických, animovaných a interaktívnych vizualizácií v jazyku Python [2].

1.3 Seaborn

Seaborn je knižnica na vizualizáciu údajov v jazyku Python založená na matplotlib. Poskytuje vysokoúrovňové rozhranie na kreslenie atraktívnej a informatívnej štatistickej grafiky [3].

1.4 Scikit-Learn

- Jednoduché a efektívne nástroje na prediktívnu analýzu údajov
- Prístupné pre každého a opakovane použiteľné v rôznych kontextoch
- Postavené na NumPy, SciPy a matplotlib
- Open Source, komerčne použiteľný - licencia BSD [4]

1.5 TensorFlow

TensorFlow je komplexná open source platforma pre strojové učenie (angl.: Machine Learning - ML). Má komplexný, flexibilný ekosystém nástrojov, knižníc a komunitných zdrojov, ktorý umožňuje výskumníkom posúvať najnovšie poznatky v oblasti ML a vývojárom ľahko vytvárať a nasadzovať aplikácie využívajúce ML [5].

1.6 Keras

Keras je API určené pre ľudí, nie pre stroje. Keras sa riadi osvedčenými postupmi na zníženie kognitívnej záťaže: ponúka konzistentné a jednoduché API, minimalizuje počet činností používateľa potrebných pre bežné prípady použitia a poskytuje jasné a použiteľné chybové hlásenia. Má tiež rozsiahlu dokumentáciu a príručky pre vývojárov [6].

2 Implementácia

2.1 Chýbajúce Údaje (Missing Data)

Riešenie tohto zadania začneme načítaním vopred nachystaných trénovacích aj testovacích dát pomocou metódy **read_csv** [7], ktorý nám ukladá načítané dáta do Pandas DataFrame-u.

Následne si môžeme zobraziť jednoduché informácie o trénovacích a testovacích dátach pomocou metódy **info** [8]. Táto metóda nám vráti nasledujúce výsledky:

```
Information about the Training Data Set

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 44776 entries, 0 to 44775
Data columns (total 27 columns):
 #   Column               Non-Null Count Dtype
---  ------               -------------- -----
 0   id                   44776 non-null object
 1   artist_id            44776 non-null object
 2   artist               44776 non-null object
 3   name                 44776 non-null object
 4   popularity           44776 non-null int64
 5   release_date         44776 non-null object
 6   duration_ms          44776 non-null int64
 7   explicit             44776 non-null bool
 8   danceability         44776 non-null float64
 9   energy               44776 non-null float64
 10  key                  44776 non-null int64
 11  loudness             44776 non-null float64
 12  mode                 44776 non-null int64
 13  speechiness          44776 non-null float64
 14  acousticness         44776 non-null float64
 15  instrumentalness     44776 non-null float64
 16  liveness             44776 non-null float64
 17  valence              44776 non-null float64
 18  tempo                44776 non-null float64
 19  artist_genres        44776 non-null object
 20  artist_followers     44775 non-null float64
 21  url                  44776 non-null object
 22  playlist_id          44776 non-null object
 23  playlist_description 30974 non-null object
 24  playlist_name        44755 non-null object
 25  playlist_url         44776 non-null object
 26  query                44776 non-null object
dtypes: bool(1), float64(10), int64(4), object(12)
memory usage: 8.9+ MB
None
```

```
Information about the Testing Data Set

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8893 entries, 0 to 8892
Data columns (total 27 columns):
 #   Column               Non-Null Count Dtype
---  ------               -------------- -----
 0   id                   8893 non-null  object
 1   artist_id            8893 non-null  object
 2   artist               8893 non-null  object
 3   name                 8893 non-null  object
 4   popularity           8893 non-null  int64
 5   release_date         8893 non-null  object
 6   duration_ms          8893 non-null  int64
 7   explicit             8893 non-null  bool
 8   danceability         8893 non-null  float64
 9   energy               8893 non-null  float64
 10  key                  8893 non-null  int64
 11  loudness             8893 non-null  float64
 12  mode                 8893 non-null  int64
 13  speechiness          8893 non-null  float64
 14  acousticness         8893 non-null  float64
 15  instrumentalness     8893 non-null  float64
 16  liveness             8893 non-null  float64
 17  valence              8893 non-null  float64
 18  tempo                8893 non-null  float64
 19  artist_genres        8893 non-null  object
 20  artist_followers     8893 non-null  float64
 21  url                  8893 non-null  object
 22  playlist_id          8893 non-null  object
 23  playlist_description 6176 non-null  object
 24  playlist_name        8888 non-null  object
 25  playlist_url         8893 non-null  object
 26  query                8893 non-null  object
dtypes: bool(1), float64(10), int64(4), object(12)
memory usage: 1.8+ MB
None
```

Z týchto údajov vidíme, že niektoré stĺpce/riadky obsahujú prázdne hodnoty tzv. NaN alebo Null Value. Môžeme vypísať aj to, že aké percento tvoria NaN hodnoty v jednotlivých stĺpcoch. Jednoduchý príkaz nám vráti nasledujúce údaje:

Number of NaN Values in the Training Data Set before dealing with NaN Values

```
Column id has 0 (0.0 %) NaN value(s)
Column artist_id has 0 (0.0 %) NaN value(s)
Column artist has 0 (0.0 %) NaN value(s)
Column name has 0 (0.0 %) NaN value(s)
Column popularity has 0 (0.0 %) NaN value(s)
Column release_date has 0 (0.0 %) NaN value(s)
Column duration_ms has 0 (0.0 %) NaN value(s)
Column explicit has 0 (0.0 %) NaN value(s)
Column danceability has 0 (0.0 %) NaN value(s)
Column energy has 0 (0.0 %) NaN value(s)
Column key has 0 (0.0 %) NaN value(s)
Column loudness has 0 (0.0 %) NaN value(s)
Column mode has 0 (0.0 %) NaN value(s)
Column speechiness has 0 (0.0 %) NaN value(s)
Column acousticness has 0 (0.0 %) NaN value(s)
Column instrumentalness has 0 (0.0 %) NaN value(s)
Column liveness has 0 (0.0 %) NaN value(s)
Column valence has 0 (0.0 %) NaN value(s)
Column tempo has 0 (0.0 %) NaN value(s)
Column artist_genres has 0 (0.0 %) NaN value(s)
Column artist_followers has 1 (0.002 %) NaN value(s)
Column url has 0 (0.0 %) NaN value(s)
Column playlist_id has 0 (0.0 %) NaN value(s)
Column playlist_description has 13802 (30.825 %) NaN value(s)
Column playlist_name has 21 (0.047 %) NaN value(s)
Column playlist_url has 0 (0.0 %) NaN value(s)
Column query has 0 (0.0 %) NaN value(s)
```

Number of NaN Values in the Testing Data Set before dealing with NaN Values

```
Column id has 0 (0.0 %) NaN value(s)
Column artist_id has 0 (0.0 %) NaN value(s)
Column artist has 0 (0.0 %) NaN value(s)
Column name has 0 (0.0 %) NaN value(s)
Column popularity has 0 (0.0 %) NaN value(s)
Column release_date has 0 (0.0 %) NaN value(s)
Column duration_ms has 0 (0.0 %) NaN value(s)
Column explicit has 0 (0.0 %) NaN value(s)
Column danceability has 0 (0.0 %) NaN value(s)
Column energy has 0 (0.0 %) NaN value(s)
Column key has 0 (0.0 %) NaN value(s)
Column loudness has 0 (0.0 %) NaN value(s)
Column mode has 0 (0.0 %) NaN value(s)
Column speechiness has 0 (0.0 %) NaN value(s)
Column acousticness has 0 (0.0 %) NaN value(s)
Column instrumentalness has 0 (0.0 %) NaN value(s)
Column liveness has 0 (0.0 %) NaN value(s)
Column valence has 0 (0.0 %) NaN value(s)
Column tempo has 0 (0.0 %) NaN value(s)
Column artist_genres has 0 (0.0 %) NaN value(s)
```

```
Column artist_followers has 0 (0.0 %) NaN value(s)
Column url has 0 (0.0 %) NaN value(s)
Column playlist_id has 0 (0.0 %) NaN value(s)
Column playlist_description has 2717 (30.552 %) NaN value(s)
Column playlist_name has 5 (0.056 %) NaN value(s)
Column playlist_url has 0 (0.0 %) NaN value(s)
Column query has 0 (0.0 %) NaN value(s)
```

Teraz nastáva otázka, čo by sme mali robiť s chýbajúcimi dátami. Existujú rôzne stratégie pre túto situáciu. Môžeme jednoducho vymazať / ignorovať tie vzorky (riadky), ktoré obsahujú NaN hodnotu. Druhá možnosť je, že vymažeme / ignorujeme príznaky (stĺpce), ktoré obsahujú príliš veľa prázdnych hodnôt a moc neovplyvňujú náš výsledok. V tomto zadaní, sme sa rozhodli riešiť túto situáciu druhou možnosťou. Vynechali sme všetky ID-čka, a stĺpce, ktoré sa týkajú zoznamu skladieb, pretože vo viacerých prípadoch používatelia nevyplnili popis toho zoznamu, ako vidíme v trénovacích aj testovacích dátach skoro jedna tretina toho stĺpca je prázdna.

```
Information about the Training Data Set after ignoring unnecessary columns

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 44776 entries, 0 to 44775
Data columns (total 19 columns):
 #   Column           Non-Null Count Dtype
---  ------           -------------- -----
 0   artist           44776 non-null object
 1   name             44776 non-null object
 2   popularity       44776 non-null int64
 3   release_date     44776 non-null object
 4   duration_ms      44776 non-null int64
 5   explicit         44776 non-null bool
 6   danceability     44776 non-null float64
 7   energy           44776 non-null float64
 8   key              44776 non-null int64
 9   loudness         44776 non-null float64
 10  mode             44776 non-null int64
 11  speechiness      44776 non-null float64
 12  acousticness     44776 non-null float64
 13  instrumentalness 44776 non-null float64
 14  liveness         44776 non-null float64
 15  valence          44776 non-null float64
 16  tempo            44776 non-null float64
 17  artist_genres    44776 non-null object
 18  artist_followers 44775 non-null float64
dtypes: bool(1), float64(10), int64(4), object(4)
memory usage: 6.2+ MB
None
```

Information about the Testing Data Set after ignoring unnecessary columns

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8893 entries, 0 to 8892
Data columns (total 19 columns):
 #   Column           Non-Null Count Dtype
---  ------           -------------- -----
 0   artist           8893 non-null  object
 1   name             8893 non-null  object
 2   popularity       8893 non-null  int64
 3   release_date     8893 non-null  object
 4   duration_ms      8893 non-null  int64
 5   explicit         8893 non-null  bool
 6   danceability     8893 non-null  float64
 7   energy           8893 non-null  float64
 8   key              8893 non-null  int64
 9   loudness         8893 non-null  float64
 10  mode             8893 non-null  int64
 11  speechiness      8893 non-null  float64
 12  acousticness     8893 non-null  float64
 13  instrumentalness 8893 non-null float64
 14  liveness         8893 non-null  float64
 15  valence          8893 non-null  float64
 16  tempo            8893 non-null  float64
 17  artist_genres    8893 non-null  object
 18  artist_followers 8893 non-null float64
dtypes: bool(1), float64(10), int64(4), object(4)
memory usage: 1.2+ MB
None
```

Number of NaN Values in the Training Data Set after dealing with NaN Values

```
Column artist has 0 (0.0 %) NaN value(s)
Column name has 0 (0.0 %) NaN value(s)
Column popularity has 0 (0.0 %) NaN value(s)
Column release_date has 0 (0.0 %) NaN value(s)
Column duration_ms has 0 (0.0 %) NaN value(s)
Column explicit has 0 (0.0 %) NaN value(s)
Column danceability has 0 (0.0 %) NaN value(s)
Column energy has 0 (0.0 %) NaN value(s)
Column key has 0 (0.0 %) NaN value(s)
Column loudness has 0 (0.0 %) NaN value(s)
Column mode has 0 (0.0 %) NaN value(s)
Column speechiness has 0 (0.0 %) NaN value(s)
Column acousticness has 0 (0.0 %) NaN value(s)
Column instrumentalness has 0 (0.0 %) NaN value(s)
Column liveness has 0 (0.0 %) NaN value(s)
Column valence has 0 (0.0 %) NaN value(s)
Column tempo has 0 (0.0 %) NaN value(s)
Column artist_genres has 0 (0.0 %) NaN value(s)
Column artist_followers has 1 (0.002 %) NaN value(s)
```

```
Number of NaN Values in the Testing Data Set after dealing with NaN Values

Column artist has 0 (0.0 %) NaN value(s)
Column name has 0 (0.0 %) NaN value(s)
Column popularity has 0 (0.0 %) NaN value(s)
Column release_date has 0 (0.0 %) NaN value(s)
Column duration_ms has 0 (0.0 %) NaN value(s)
Column explicit has 0 (0.0 %) NaN value(s)
Column danceability has 0 (0.0 %) NaN value(s)
Column energy has 0 (0.0 %) NaN value(s)
Column key has 0 (0.0 %) NaN value(s)
Column loudness has 0 (0.0 %) NaN value(s)
Column mode has 0 (0.0 %) NaN value(s)
Column speechiness has 0 (0.0 %) NaN value(s)
Column acousticness has 0 (0.0 %) NaN value(s)
Column instrumentalness has 0 (0.0 %) NaN value(s)
Column liveness has 0 (0.0 %) NaN value(s)
Column valence has 0 (0.0 %) NaN value(s)
Column tempo has 0 (0.0 %) NaN value(s)
Column artist_genres has 0 (0.0 %) NaN value(s)
Column artist_followers has 0 (0.0 %) NaN value(s)
```

Po vynechaní týchto stĺpcov jediná prázdna hodnota bude v trénovacích dátach v stĺpci artist_followers. Keď nad tým logicky zamýšľame, tak je zrejmé, že ak daný umelec nemá uvedený počet fanúšikov, tak to môže znamenať len to, že zatiaľ nemá žiadnych fanúšikov a tým pádom môžeme túto prázdnu hodnotu nahradiť nulou. Po odstránení prázdnych hodnôt môžeme prejsť na duplicitné hodnoty. V tomto zadaní, sme nemohli len jednoducho ignorovať duplicitné hodnoty, lebo sme zistili, že niektorí umelci majú rôzne verzie toho istého piesňa s rôznymi atribútmi, a kvôli tomu sme sa rozhodli, že v tomto zadaní za duplicitných hodnôt budeme považovať vzorky, ktoré majú identické meno, umelca a rok vydania. Pre túto operáciu, sme museli prerobiť formát dátumu vydania, aby sme mohli s jednotlivými atribútmi pracovať ako číselnými hodnotami. Preto sme rozdelili dátum na 3 stĺpce (deň, mesiac, rok). Po vykonaní týchto funkcií sme dosiahli nasledujúce výsledky.

Information about the Training Data Set after ignoring Duplicate Values

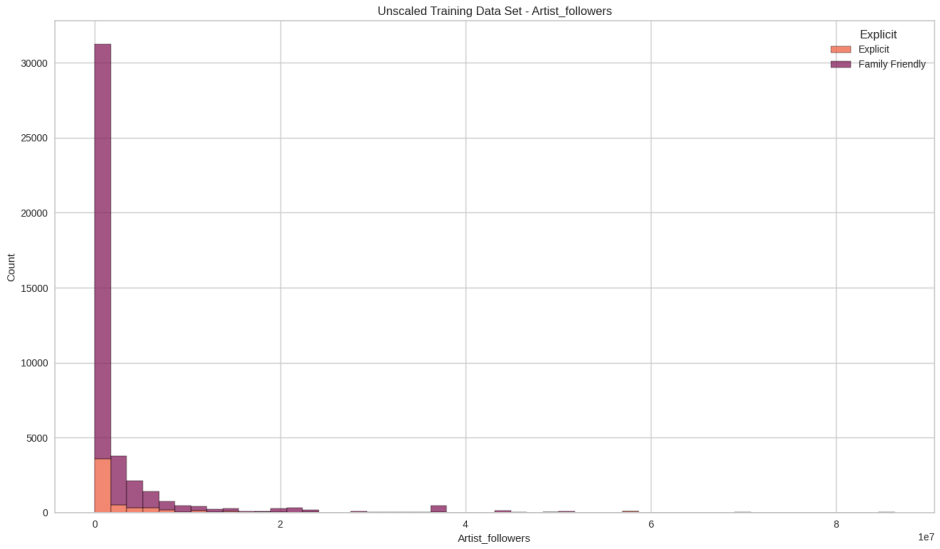
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 43084 entries, 0 to 43083
Data columns (total 21 columns):
 #   Column            Non-Null Count Dtype
---  ------            -------------- -----
 0   artist            43084 non-null object
 1   name              43084 non-null object
 2   popularity        43084 non-null int64
 3   duration_ms       43084 non-null int64
 4   explicit          43084 non-null bool
 5   danceability      43084 non-null float64
 6   energy            43084 non-null float64
 7   key               43084 non-null int64
 8   loudness          43084 non-null float64
 9   mode              43084 non-null int64
 10  speechiness       43084 non-null float64
 11  acousticness      43084 non-null float64
 12  instrumentalness  43084 non-null float64
 13  liveness          43084 non-null float64
 14  valence           43084 non-null float64
 15  tempo             43084 non-null float64
 16  artist_genres     43084 non-null object
 17  artist_followers  43084 non-null float64
 18  release_date_year 43084 non-null int64
 19  release_date_month 43084 non-null int64
 20  release_date_day  43084 non-null int64
dtypes: bool(1), float64(10), int64(7), object(3)
memory usage: 6.6+ MB
None
```

```
Information about the Testing Data Set after ignoring Duplicate Values

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8825 entries, 0 to 8824
Data columns (total 21 columns):
 #   Column            Non-Null Count Dtype
---  ------            -------------- -----
 0   artist            8825 non-null  object
 1   name              8825 non-null  object
 2   popularity        8825 non-null  int64
 3   duration_ms       8825 non-null  int64
 4   explicit          8825 non-null  bool
 5   danceability      8825 non-null  float64
 6   energy            8825 non-null  float64
 7   key               8825 non-null  int64
 8   loudness          8825 non-null  float64
 9   mode              8825 non-null  int64
 10  speechiness       8825 non-null  float64
 11  acousticness      8825 non-null  float64
 12  instrumentalness  8825 non-null  float64
 13  liveness          8825 non-null  float64
 14  valence           8825 non-null  float64
 15  tempo             8825 non-null  float64
 16  artist_genres     8825 non-null  object
 17  artist_followers  8825 non-null  float64
 18  release_date_year 8825 non-null  int64
 19  release_date_month 8825 non-null int64
 20  release_date_day  8825 non-null  int64
dtypes: bool(1), float64(10), int64(7), object(3)
memory usage: 1.4+ MB
None
```

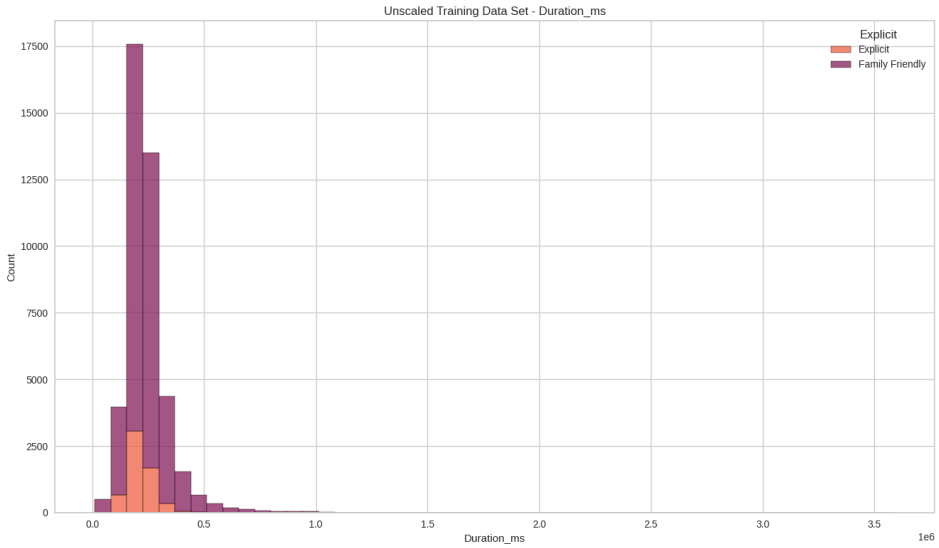
Už máme skoro vyčistené dáta, jediné čo nám ostalo, je odstránenie outlier hodnôt. Pre lepšiu vizualizáciu to budeme riešiť na začiatku EDA.

2.2 Prieskumná Analýza Údajov (Exploratory Data Analysis)

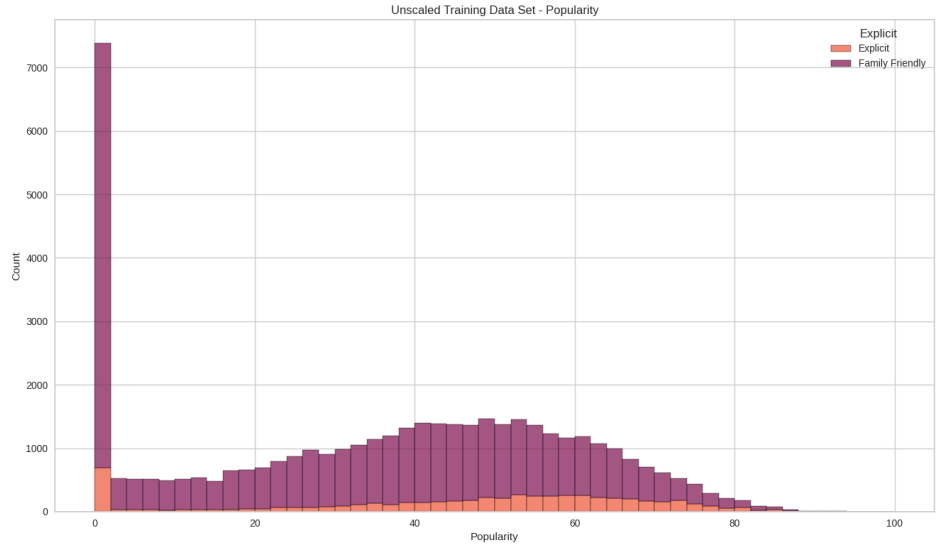
V tejto sekcií prejdeme niektoré vlastnosti trénovacích aj testovacích dát. Ako sme už spomínali, začneme s odstránením outlier hodnôt. EDA začneme zobrazením **Pair plotu** [9] (ktorá nám bohužiaľ nezmestila do tejto dokumentácie kvôli nadmernej veľkosti, ale samotný plot Vám vygeneruje program). Na tomto grafe sme boli schopný identifikovať niektoré problematické stĺpce, ktoré v sebe zahŕňajú potencionálnych outlier hodnôt. Pre lepšie znázornenie sme sa rozhodli vykresľovať histogramy pomocou metódy **histplot** [10].



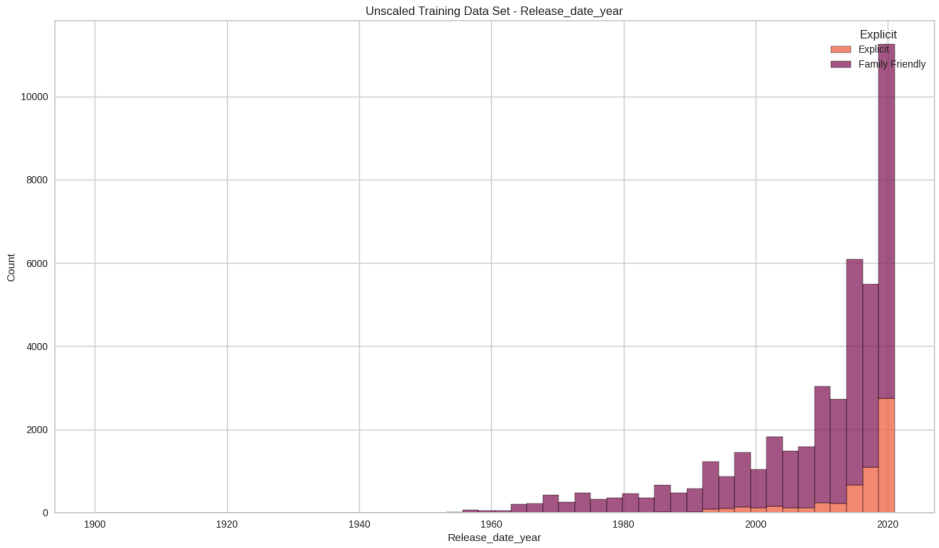
Obr. 1: Stĺpec Artist Followers v Trénovacích Dátach so znázorneným vulgarizmu pred škálovaním



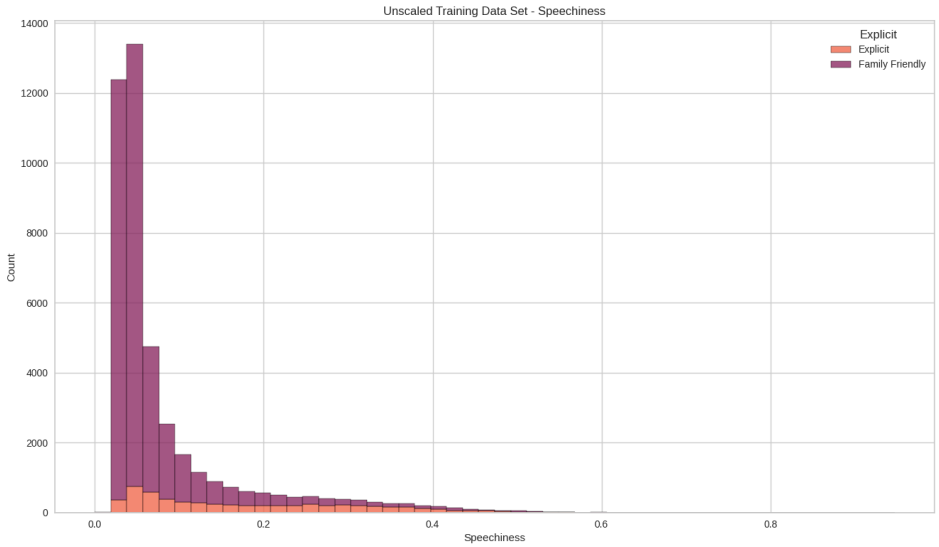
Obr. 2: StĺpecDuration (ms) v Trénovacích Dátach so znázorneným vulgarizmu pred škálovaním



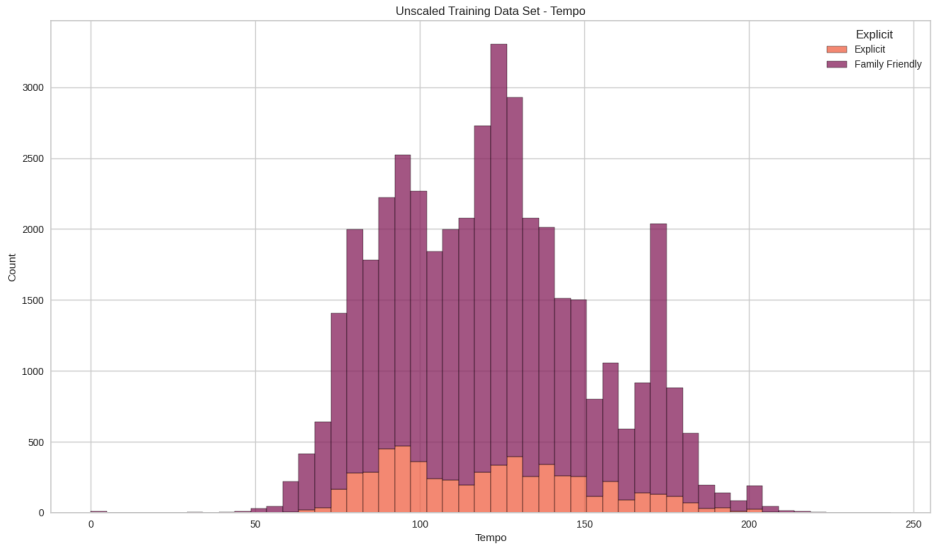
Obr. 3: StĺpecPopularity v Trénovacích Dátach so znázorneným vulgarizmu pred škálovaním



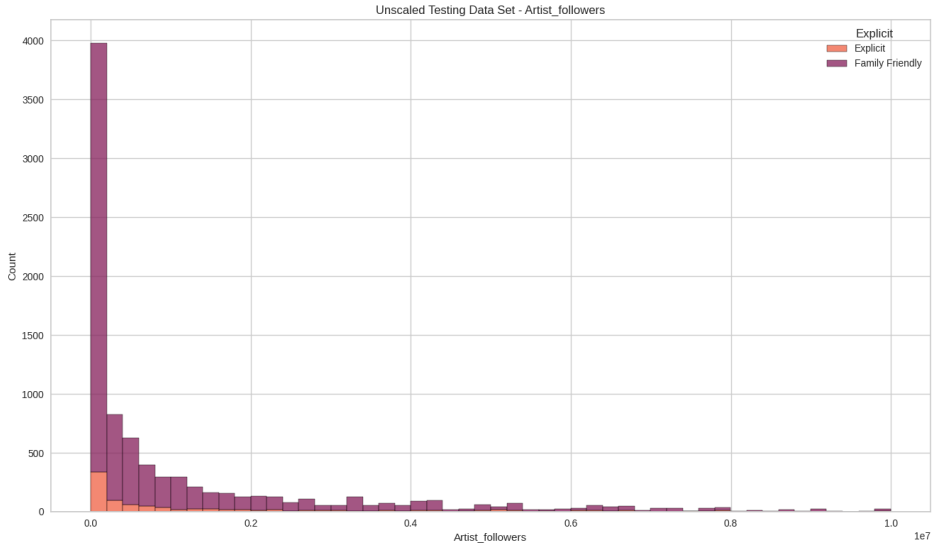
Obr. 4: StĺIpec Release Date (Year) v Trénovacích Dátach so znázorneným vulgarizmu pred škálovaním



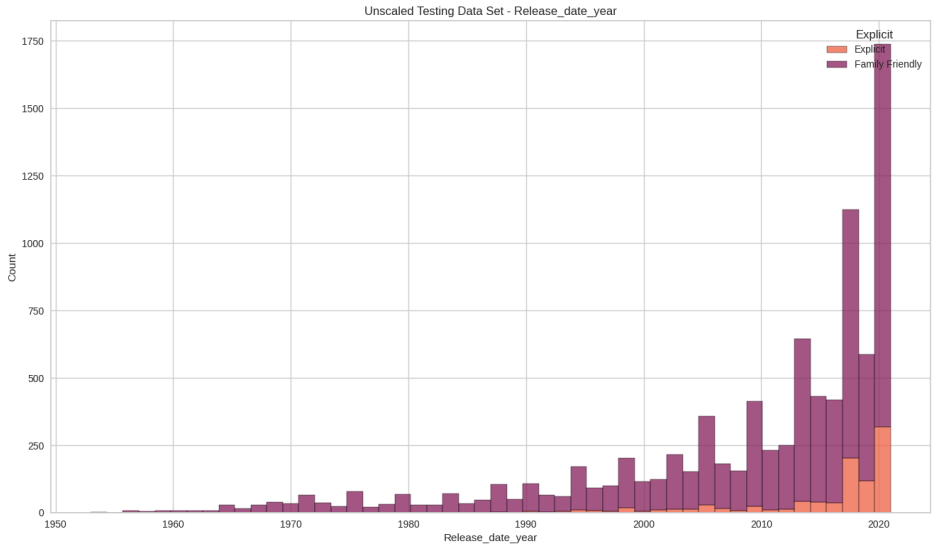
Obr. 5: StĺIpec Speechiness v Trénovacích Dátach so znázorneným vulgarizmu pred škálovaním



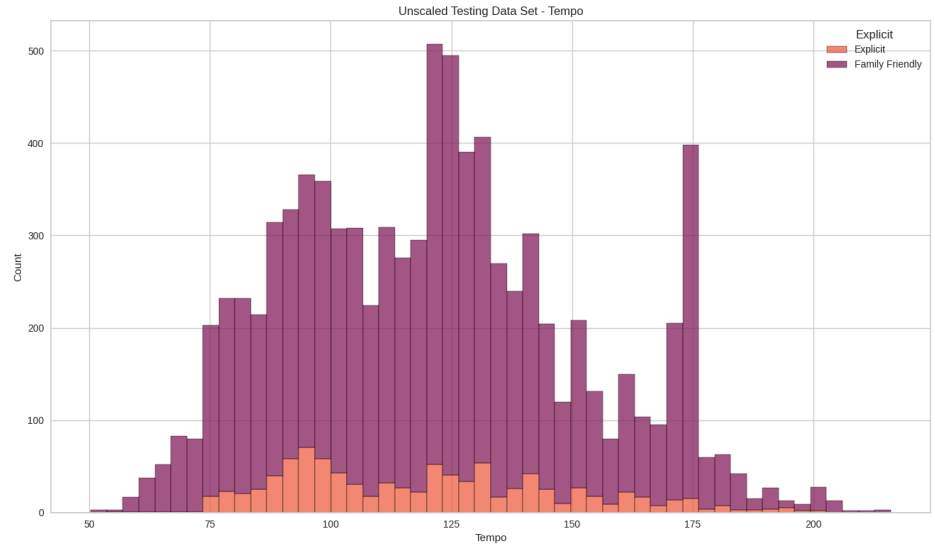
Obr. 6: Stĺpec Tempo v Trénovacích Dátach so znázorneným vulgarizmu pred škálovaním



Obr. 7: Stĺpec Artist Followers v Testovacích Dátach so znázorneným vulgarizmu pred škálovaním



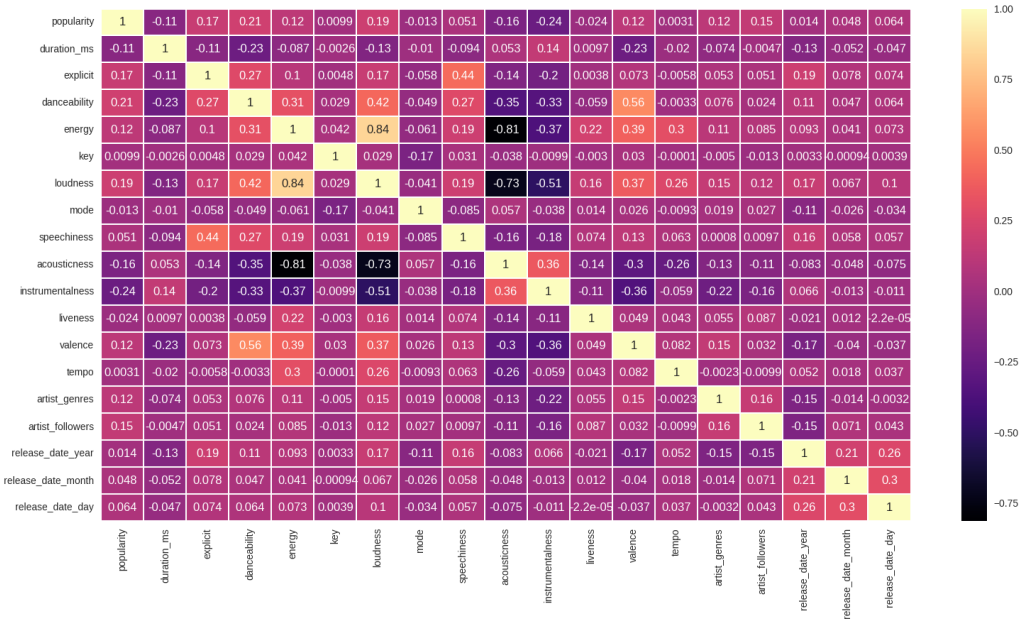
Obr. 8: Stĺpec Release Date (Year) v Testovacích Dátach so znázorneným vulgarizmu pred škálovaním



Obr. 9: Stĺpec Tempo v Testovacích Dátach so znázorneným vulgarizmu pred škálovaním

Na vyššie uvedených obrázkoch vidíte rozloženie dát v jednotlivých stĺpcoch. Vo viacerých prípadoch si môžete všimnúť, že na pravej alebo ľavej strane existuje malá množina vzoriek, ktoré sú ďaleko od ostatných vzoriek, tieto hodnoty nazývame outlier hodnoty, ktoré nám môžu znižovať úspešnosť predikcie. Preto je odporúčané ich ignorovať, alebo v niektorých prípadoch odstrániť. Na odstránenie týchto hodnôt na základe odporúčania [11, 12] sme vybrali technológiu **IsolationForest** [13]. Táto funkcia po natrénovaní vráti hodnoty 1 (riadok obsahuje inlier hodnoty) a -1 (riadok obsahuje outlier hodnoty). Na základe týchto hodnôt sme boli schopní ignorovať riadky, ktoré obsahujú outlier hodnoty.

Po vynechaní outlier hodnôt sme vygenerovali korelačnú maticu a podobne sme zobrazili aj korelačné hodnoty pre stĺpec Loudness. Túto stratégiu sme si vybrali z toho dôvodu, aby sme znížili čas trénovania našich modelov. Ako vidíte v korelačných hodnotách pre stĺpec Loudness naše dáta obsahujú stĺpce, ktoré sú slabo korelované s hlasitosťou, preto sme sa rozhodli ich vynechať. Vynechali sme hodnoty $< -0.15, 0.15 >$. Tým pádom sme vynechali stĺpce Duration (ms), Mode, Key, Release Date - Month, Release Date - Day a Artist Followers.



Obr. 10: Heatmap pre naše dáta

```
Correlation for Loudness

acousticness       -0.724899
instrumentalness   -0.511669
duration_ms        -0.150212
mode               -0.039109
key                 0.027298
release_date_month  0.070068
release_date_day    0.104378
artist_followers    0.120839
liveness            0.155252
speechiness         0.164955
explicit            0.168268
release_date_year   0.171954
popularity          0.189633
tempo               0.264097
valence             0.365620
danceability        0.417483
energy              0.838364
loudness            1.000000
Name: loudness, dtype: float64
```

Po vynechaní outlier hodnôt a menej relevantných stĺpcov naše dáta vyzerali nasledovne.

```
Information about the Training Data Set after ignoring Outlier Values and Not Strongly Correlated Columns

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38775 entries, 0 to 38774
Data columns (total 12 columns):
 #   Column           Non-Null Count Dtype
---  ------           -------------- -----
 0   popularity       38775 non-null object
 1   explicit         38775 non-null object
 2   danceability     38775 non-null object
 3   energy           38775 non-null object
 4   speechiness      38775 non-null object
 5   acousticness     38775 non-null object
 6   instrumentalness 38775 non-null object
 7   liveness         38775 non-null object
 8   valence          38775 non-null object
 9   tempo            38775 non-null object
 10  artist_genres    38775 non-null object
 11  release_date_year 38775 non-null object
dtypes: object(12)
memory usage: 3.6+ MB
None
```

```
Information about the Testing Data Set after ignoring Outlier Values and Not Strongly Correlated Columns

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7942 entries, 0 to 7941
Data columns (total 12 columns):
 #   Column           Non-Null Count Dtype
---  ------           -------------- -----
 0   popularity       7942 non-null  object
 1   explicit         7942 non-null  object
 2   danceability     7942 non-null  object
 3   energy           7942 non-null  object
 4   speechiness      7942 non-null  object
 5   acousticness     7942 non-null  object
 6   instrumentalness 7942 non-null  object
 7   liveness         7942 non-null  object
 8   valence          7942 non-null  object
 9   tempo            7942 non-null  object
 10  artist_genres    7942 non-null  object
 11  release_date_year 7942 non-null object
dtypes: object(12)
memory usage: 744.7+ KB
None
```

Následne môžeme zobraziť aj popis dát. Popis dát môžeme vygenerovať pomocou metódy **describe** [14]. Tento popis obsahuje informácie o jednotlivých stĺpcoch, hlavne čo sa týka minimálnej/maximálnej hodnoty, priemernej hodnoty a štandardnej odchýlky. Pre lepšie grafické znázornenie si transponujeme popis pomocou metódy **transpose** [15]. Po spustení príkazu dostaneme nasledujúci výpis pre trénovacie dáta:

	count	mean	std	min	25%	50%	75%	max
popularity	38775.0	36.069529335912314	23.334149952439876	0.0	17.0	39.0	54.0	100.0
explicit	38775.0	0.11901998710509348	0.3238162040653917	0.0	0.0	0.0	0.0	1.0
danceability	38775.0	0.56122178207608	0.18117987087854423	0.0	0.442	0.573	0.696	0.988
energy	38775.0	0.6063431720696326	0.2679868955872795	0.000355	0.434	0.658	0.827	1.0
speechiness	38775.0	0.0796450986460348	0.08011013213956729	0.0	0.0359	0.0473	0.0818	0.935
acousticness	38775.0	0.3069316460451321	0.3423029834422245	1.13e-06	0.0163	0.145	0.563	0.996
instrumentalness	38775.0	0.17109631016840746	0.3174206370710301	0.0	0.0	0.000188	0.11	0.997
liveness	38775.0	0.17236773178594458	0.13573121330246407	0.0104	0.0929	0.119	0.207	1.0
valence	38775.0	0.4537296840747905	0.2594408321306317	0.0	0.236	0.44	0.659	0.999
tempo	38775.0	121.00477462282399	29.74707714339013	0.0	97.0965	120.047	139.966	220.099
artist_genres	38775.0	30.89036750483559	9.211802215267499	0.0	26.0	33.0	37.0	47.0
release_date_year	38775.0	2009.4830174081237	12.851012263455063	1933.0	2004.0	2014.0	2019.0	2021.0

Tabuľka 2: Neškálovaný výstup metódy **describe().transpose()** na trénovacích dátach

	count	mean	std	min	25%	50%	75%	max
popularity	7942.0	35.72777637874591	22.190085188804897	0.0	19.0	39.0	53.0	79.0
explicit	7942.0	0.07088894485016368	0.25665540415158083	0.0	0.0	0.0	0.0	1.0
danceability	7942.0	0.5486039410727777	0.17739583741290213	0.0593	0.433	0.561	0.679	0.968
energy	7942.0	0.5993172297909846	0.27589363094906316	0.000634	0.413	0.655	0.83	1.0
speechiness	7942.0	0.06178475195164945	0.045219335400928526	0.0224	0.0351	0.0445	0.0679	0.299
acousticness	7942.0	0.3202791207668093	0.35303894325986784	2.23e-06	0.0151	0.15	0.616	0.996
instrumentalness	7942.0	0.1958129268660287	0.3378053064912411	0.0	1.08e-06	0.000468	0.23175	0.995
liveness	7942.0	0.1523109166456812	0.09309202461901556	0.0172	0.091	0.115	0.186	0.498
valence	7942.0	0.4539270838579703	0.2651700218997727	0.0	0.233	0.437	0.669	1.0
tempo	7942.0	121.14611231427854	29.162198358622515	51.737	98.00425	120.0645	139.58275	215.993
artist_genres	7942.0	30.656509695290858	9.66388253604179	0.0	26.0	33.0	39.0	47.0
release_date_year	7942.0	2009.4007806597833	12.826083344024164	1956.0	2004.25	2014.0	2019.0	2021.0

Tabuľka 3: Neškálovaný výstup metódy **describe().transpose()** na testovacích dátach

Teraz nasleduje škálovanie dát, na čo použijeme **StandardScaler** [16]. Po škálovaní si spustíme tie isté príkazy a pozrieme sa na dáta.

	count	mean	std	min	25%	50%	75%	max
popularity	38775.0	2.428031269844314e-17	1.000012895155856	-1.5407967654099024	-0.8106945322778186	0.13414365177546622	0.7783515045390695	2.7539222530141196
explicit	38775.0	-3.2985757541281244e-17	1.0000128951557192	-0.34301694299322577	-0.34301694299322577	-0.34301694299322577	-0.34301694299322577	2.9153078890896573
danceability	38775.0	-2.2026366689455587e-16	1.0000128951559335	-3.0704260132659646	-0.6478125741274328	0.06407155719117884	0.742844798681018	2.3818338452052643
energy	38775.0	-6.192304352496533e-15	1.0000128951559162	-2.2423154627313	-0.6560605168725286	0.19707382948371815	0.8276513898339879	1.4656475097177901
speechiness	38775.0	-1.381687228273674e-15	1.000012895155932	-1.0246873735129653	-0.5458043183159481	-0.3986614801269205	0.03607872361338836	11.616220089089863
acousticness	38775.0	4.348649816201918e-15	1.0000128951559395	-0.9054454536484993	-0.8569946529149279	-0.47023245057518465	0.7802889988503918	1.9843873782276875
instrumentalness	38775.0	-7.817344450683655e-16	1.0000128951560192	-0.5461344846488055	-0.5461344846488055	-0.545504126676337	-0.15798302376894374	2.549761567328972
liveness	38775.0	1.6705221631814894e-14	1.0000128951559186	-1.1344248118224103	-0.5728442141788127	-0.3969683075407332 6	0.7879840589313594	2.1032475614221577
valence	38775.0	1.4652481534849157e-15	1.0000128951559388	-1.7387063537483334	-0.8387892204651556	-0.05424607862853876	0.7879840589313594	2.1032475614221577
tempo	38775.0	6.581155850957637 5e-15	1.0000128951559355	-4.056777814702577	-0.798858461114727	-0.03194900273723467	0.6360898358341314	3.3165095535588014
artist_genres	38775.0	-3.880726959783242e-16	1.0000128951559824	-3.6750736468563163	0.09633550846305466	0.2106206343818235	0.7820462639756676	1.6963272713258182
release_date_year	38775.0	5.46444471447226e-15	1.0000128951558909	-7.2144213086881885	-0.383713296069379	0.3669139580645561	0.7422275851315236	0.8923530359583107

Tabuľka 4: Škálovaný výstup metódy **describe().transpose()** na trénovacích dátach

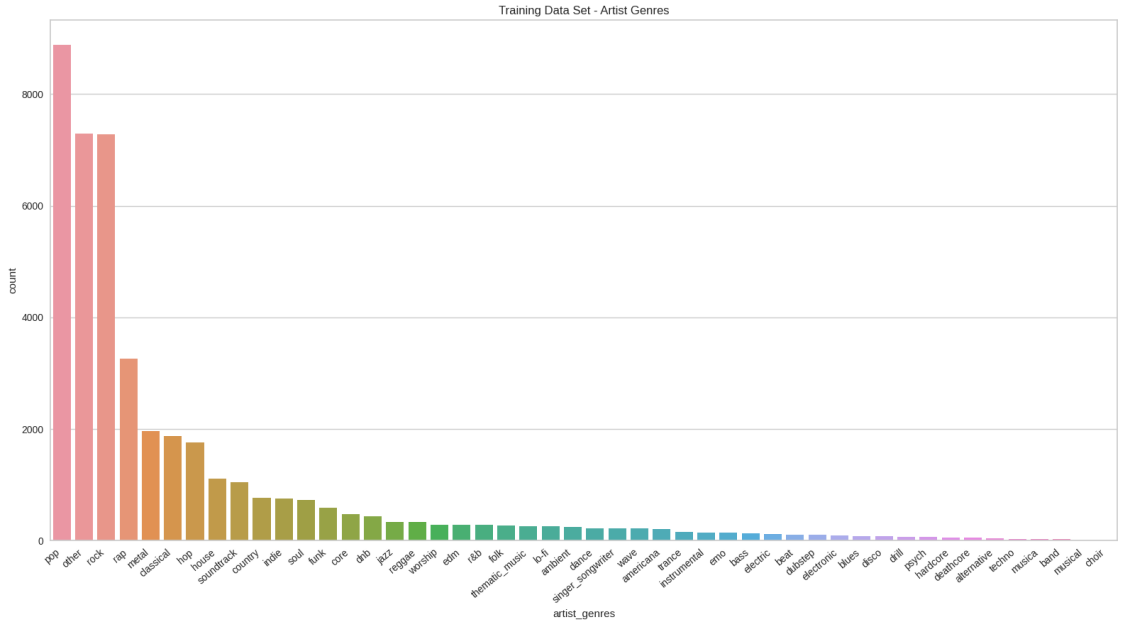
	count	mean	std	min	25%	50%	75%	max
popularity	7942.0	-0.004577611104956143	0.9598015246829911	-1.5407967654099024	-0.7570105445475184	0.13414365177546622	0.7354043143548293	1.8520312591450752
explicit	7942.0	-0.0755241464031931	0.8944980556501224	-0.34301694299322577	-0.34301694299322577	-0.34301694299322577	-0.34301694299322577	2.9153078890896573
danceability	7942.0	-0.00367221720070985	0.9567724951625346	-2.727728303538168	-0.6478125741274328	0.04199763063866372	0.6986969455759878	2.2714642124426887
energy	7942.0	0.004949259527988197	1.0054888293848034	-2.2412805736763715	-0.6671883561728276	0.202637749133868	0.8461977886678194	1.4656475097177901
speechiness	7942.0	-0.17763815584848788	0.6314701396527233	-0.7250510484734908	-0.5551679534734317	-0.42407706126866157	-0.09501216859138156	2.9615458792443285
acousticness	7942.0	0.0009339632452177142	1.0133032635134087	-0.9054422620624335	-0.8641757215630215	-0.4905425437213077	0.8180077432646204	1.9843873782276875
instrumentalness	7942.0	0.0313088158099567	1.0259538501456913	-0.5461344846488055	-0.5461344846488055	-0.5448986103973641	-0.0003935306517198785	2.5435511439548946
liveness	7942.0	-0.16040994703913591	0.6395575531046509	-1.088248898110482	-0.585067250161382	-0.4173400341783493	0.07837198067029 18	2.169869248798806
valence	7942.0	0.03125715739970195	1.0059122645134075	-1.6521758601634124	-0.796485423601416	-0.02732548062434067	0.84567105465464	2.1070933611370437
tempo	7942.0	0.023482704287044572	0.9687759474407661	-2.1291527601914866	-0.7375295084933792	-0.01754922952369423	0.6356958955890588	3.1847826409720708
artist_genres	7942.0	-0.04576542561499145	1.041658900226797	-3.6750736468563163	-0.2465198692932518	0.09633550846305466	0.7820462639756676	1.6963272713258182
release_date_year	7942.0	0.018895867511549865	0.9628825513626175	-3.9867241159122675	-0.383713296069379	0.3669139580645561	0.7422275851315236	0.8923530359583107

Tabuľka 5: Škálovaný výstup metódy **describe().transpose()** na testovacích dátach

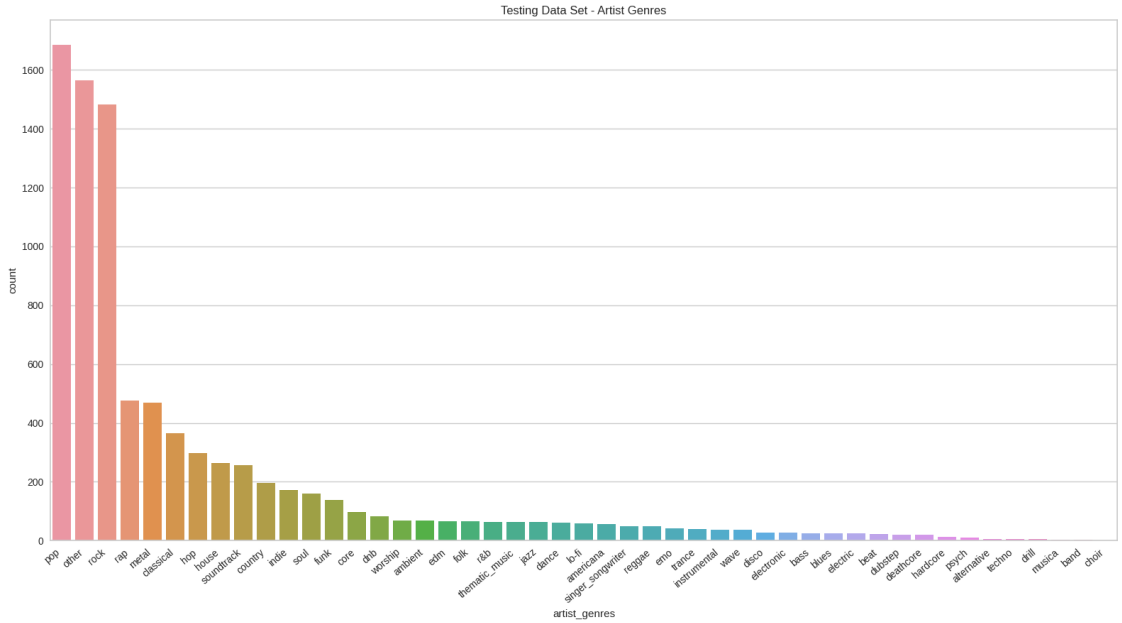
V kóde vidíme, že na trénovacích dátach sme použili fit + transform (alebo fit_transform) a na testovacích len transform. Nemôžeme používať fit na testovacích dátach, aby náhodou nenastal Data Leakage [17], aby nám program nevedel dopredu aké sú presné dáta.

Následne sme vykreslili rôzne grafy, z ktorých sme dozvedeli rôzne informácie o naších dátach.

Najprv sme zistili rozloženie rôznych zakódovaných žánrov v trénovacích a testovacích dátach. Na túto úlohu sme používali **Count Plot** [18].



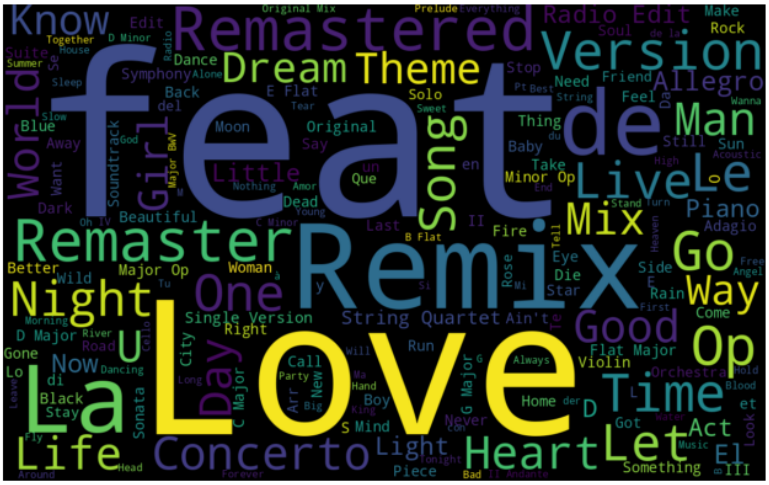
Obr. 11: Žánre v trénovacích dátach



Obr. 12: Žánre v testovacích dátach

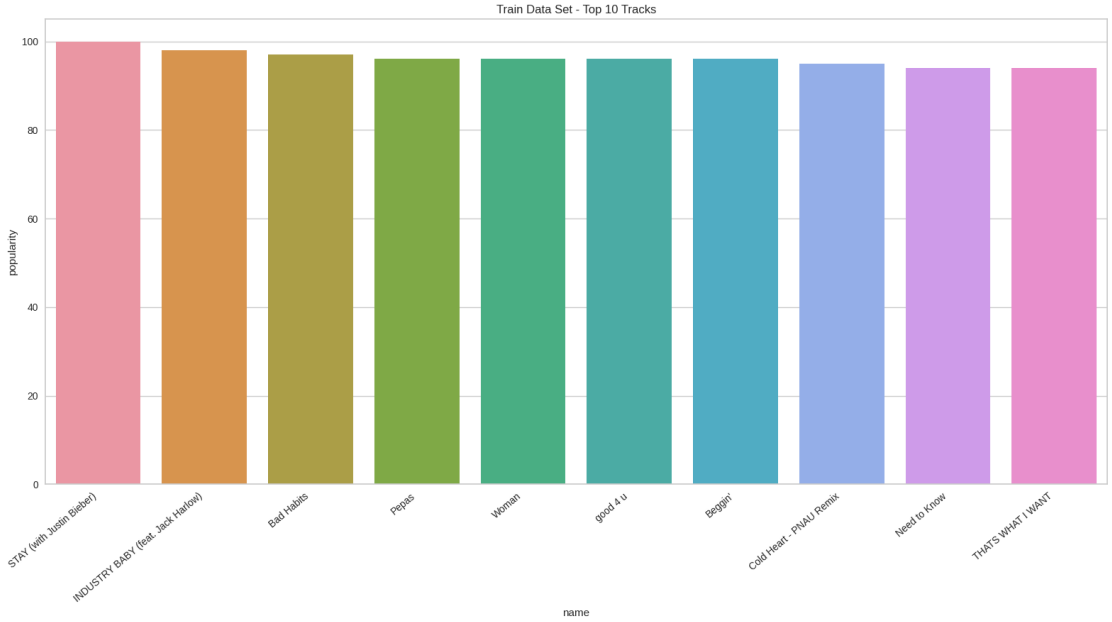
Potom nás zaujímalo, ktoré sú najčastejšie použité slová v názvu skladieb. Na túto úlohu sme používali **Word Cloud** [19, 20].

Obr. 13: Word Cloud pre trénovacie dáta

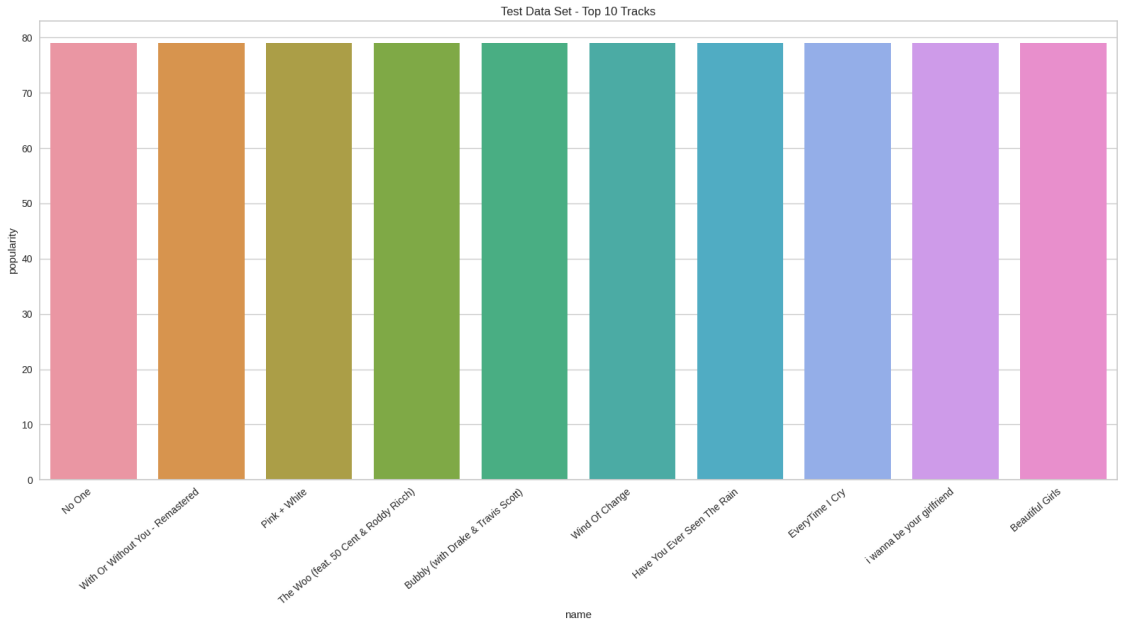


Obr. 14: Word Cloud pre testovacie dáta

Boli sme zvedaví, ktoré sú najpopulárnejšie skladby, kvôli tomu sme usporiadali dáta podľa popularity a následne na základe počtu fanúšikov daného umelca. Na túto úlohu sme použili **Bar Plot** [21].



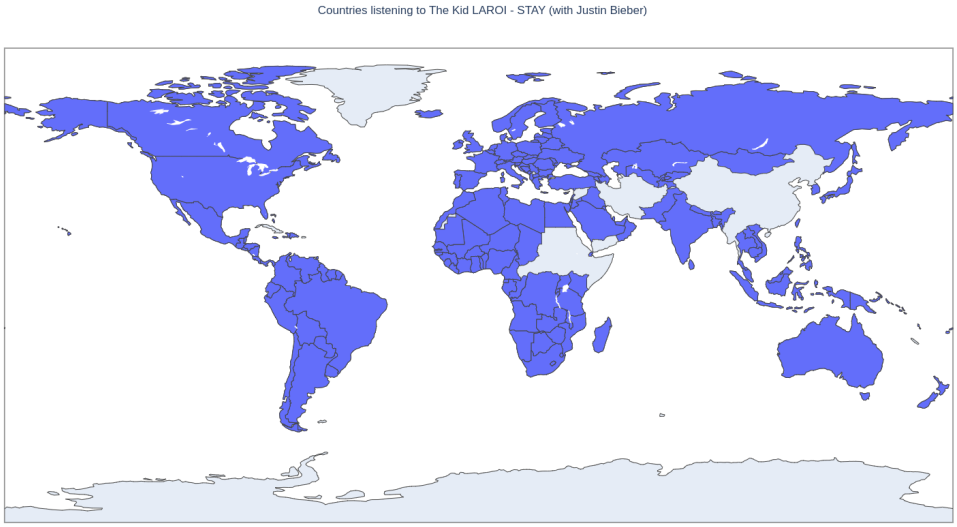
Obr. 15: Najpopulárnejšie skladby v Trénovacích Dátach



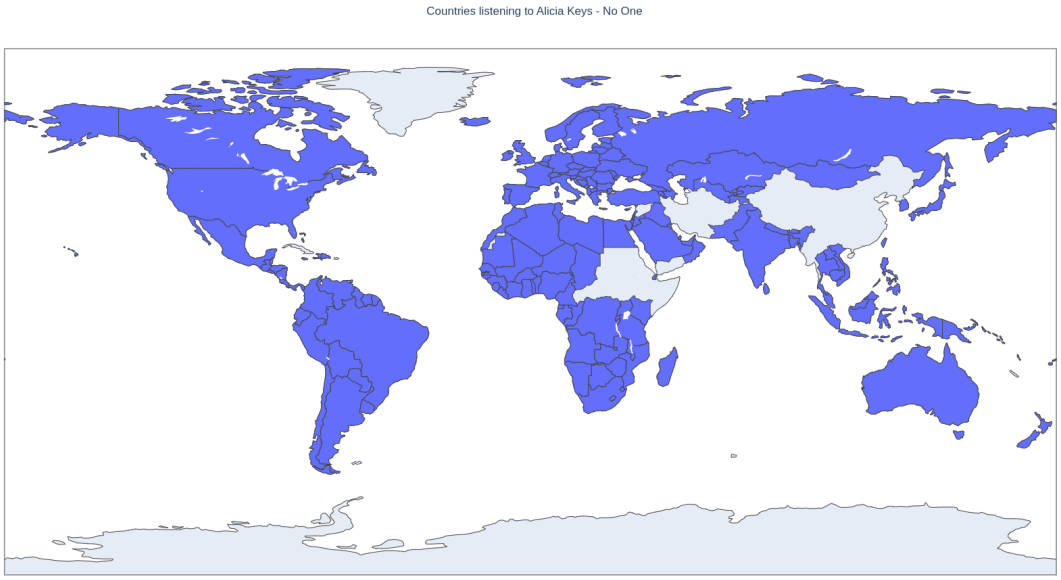
Obr. 16: Najpopulárnejšie skladby v Testovacích Dátach

Po zobrazení najpopulárnejších skladieb nás zaujalo, kde všade je dostupná najpopulárnejšia skladba z trénovacích a testovacích dát. Kvôli tomu pomocou balíčka **Spotipy** [22]

sme stiahli príslušné informácie. Dosiahnuté informácie sme preformátovali na štandardu **ISO 3166-1 alpha-3**, čo je vlastne reprezentácia krajín tromi písmenami. Následne tieto údaje sme zobrazili na **Choropleth Mape** [23].



Obr. 17: Dostupnosť najpopulárnejšej pesničky z trénovacích dát



Obr. 18: Dostupnosť najpopulárnejšej pesničky z testovacích dát

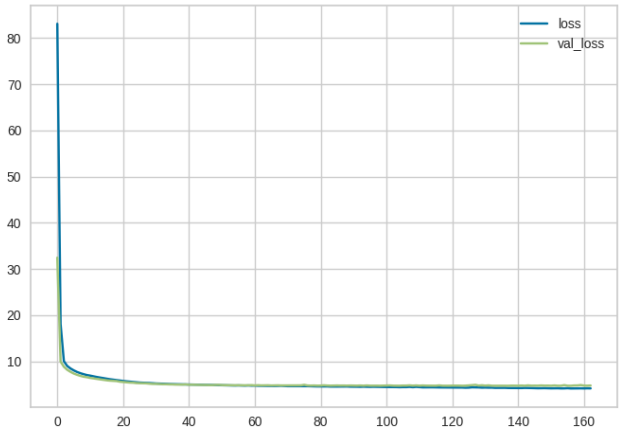
Tým pádom sme spoznali naše dáta a môžeme prejsť na rôzne modely strojového učenia a môžeme predpovedať hlasitosť skladieb.

2.3 Neurónová Sieť (Neural Network)

Z prvého zadania už máme skúsenosti s Neurónovými sieťami, takže aj v tomto prípade sme ich vyskúšali. Ale v tomto zadaní, sem ich používali na regresnú úlohu, takže menili sme aktivačné a evaluačné funkcie. Na túto úlohu, sme používali nasledujúcu štruktúru.

Learning Rate	Activation Function	Layers	Neurons	Batch Size	Patience
0.001	Relu	3	64	1024	25

Na nižšie uvedenom obrázku vidíte priebeh trénovania. Keďže sme mali regresnú úlohu, úspešnosť sme ani nemerali, totiž úspešnosť sa počíta len pri klasifikačných úlohách.



Obr. 19: Priebeh trénovania

Dosiahli sme nasledujúce výsledky.

```
Neural Network - Mean Absolute Error (MAE):
1.6577391135526884

Neural Network - Mean Squared Error (MSE):
5.076282240881297

Neural Network - Root Mean Square Error (RMSE):
2.253060638527356

Neural Network - R2 Score:
0.8748374898282059
```

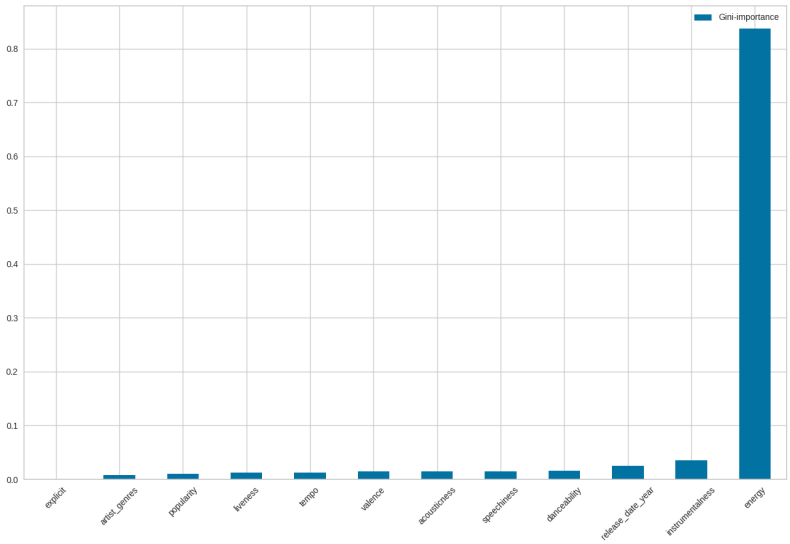
2.4 Náhodný Les Regressor (Random Forest Regressor)

V tejto časti zadania sme vyskúšali prácu s Náhodným Les Regressorom. Na túto úlohu sme použili **RandomForestRegressor** [24]. Vlastnosť **n_estimators** (čo je počet

stromov v danom lese) sme nastavili na 100. Po vytvorení a natrénovaní sme si vybrali jeden strom, ktorý sme potom exportovali do svg súboru (ktorá nám tiež kvôli veľkosti sa nezmestí do dokumentácie, ale program Vám to vygeneruje). Dosiahli sme nasledujúce výsledky.

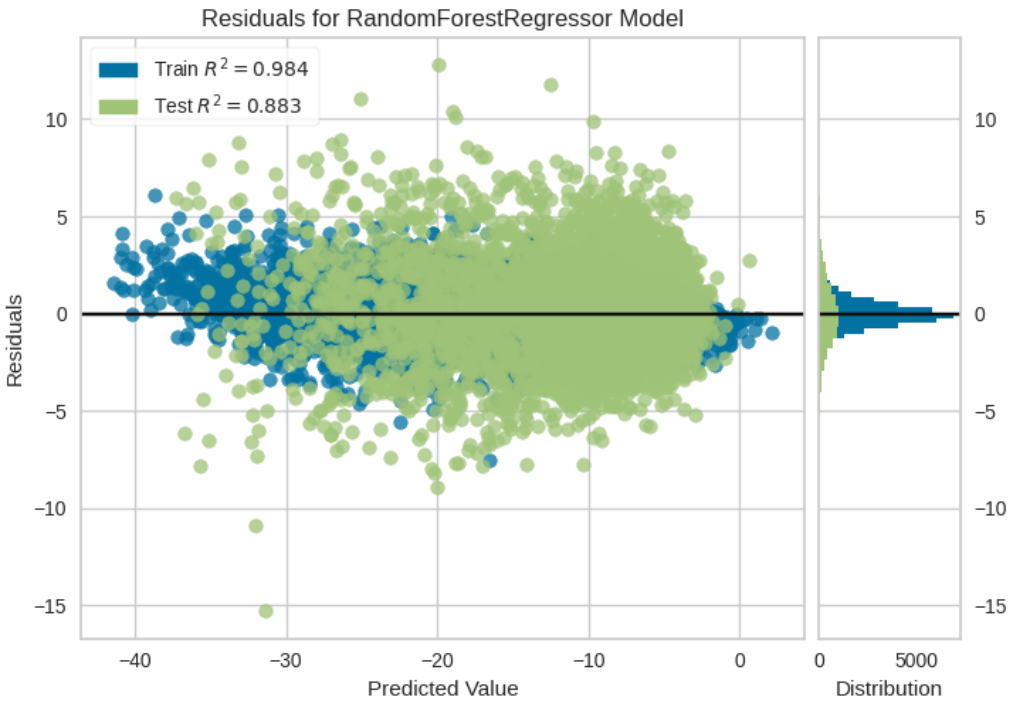
```
Random Forest Regressor - Mean Squared Error (MSE) from Cross Validation:
[4.17396339 4.27635781 4.33212575 4.37007134 4.4098394 ]

Random Forest Regressor - R2 Score from Cross Validation:
[0.88963626 0.87576482 0.88051305 0.87967785 0.87945343]


Random Forest Regressor - Error Values for Predicted Values:

Mean Absolute Error (MAE):
1.5618058587257617

Mean Squared Error (MSE):
4.4612442852317935

Root Mean Square Error (RMSE):
2.112165780716986

R2 Score:
0.889030182732353
```

Po exporte sme boli zvedaví, že ktorá vlastnosť našich dát je najsilnejšia v tejto časti zadania. Na nižšie uvedenom obrázku vidíte výsledky.



Obr. 20: Sila vstupných príznakov

```
Random Forest Regressor Feature Importance:

Feature energy: 0.8338375191705992
Feature instrumentalness: 0.03529426358218532
Feature release_date_year:  0.026503656515352844
Feature danceability: 0.015750342350729668
Feature speechiness: 0.015243671765562142
Feature valence: 0.014848327887322148
Feature acousticness: 0.014362140266718719
Feature tempo: 0.01320078253366416
Feature liveness: 0.01223401589071116
Feature popularity: 0.010373246846301113
Feature artist_genres: 0.008003874431146328
Feature explicit: 0.0003481587597071529
```



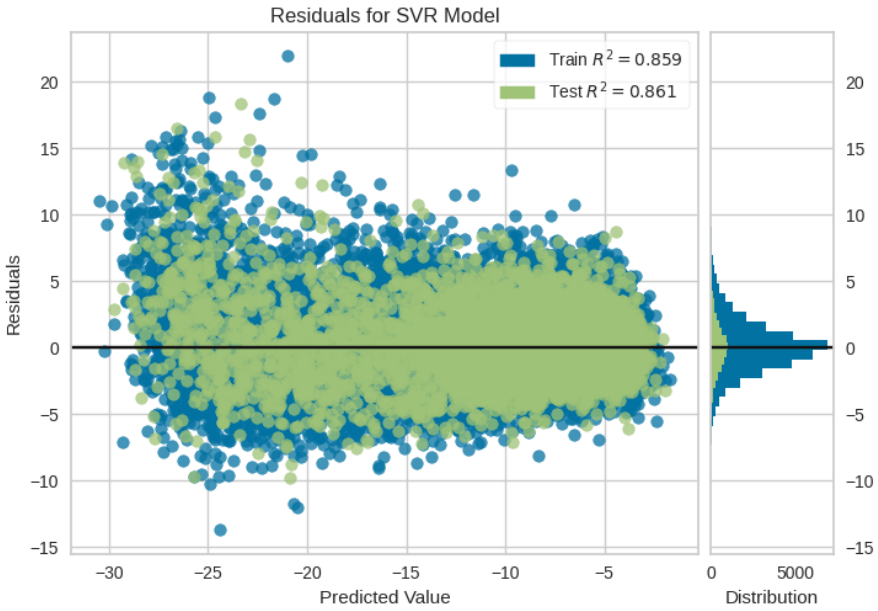
Obr. 21: Reziduály pre Náhodný Les Regressor

2.5 Mechanizmus podporných vektorov (Support Vector Machine)

2.5.1 Predvolené nastavenia

Na začiatku sme vyskúšali zo zaujímavosti, že bez nastavovania aké hodnoty môžeme dosiahnuť. Teda všetky hodnoty sú defaultné, s ktorými sme dosiahli nasledujúce výsledky.

```
Mean Squared Error (MSE) from Cross Validation:
[5.10440214 5.13760479 5.2198723 5.06404368 5.17586937]

R2 Score from Cross Validation:
[0.86503453 0.85074418 0.85602758 0.86057055 0.85851337]

Error Values for Predicted Values:

Mean Absolute Error (MAE) :
1.6775507309368922

Mean Squared Error (MSE) :
5.469948907128103

Root Mean Square Error (RMSE) :
2.338792189812533

R2 Score :
0.863939476998213
```



Obr. 22: Reziduály pre Mechanizmus podporných vektorov

2.5.2 Grid Search

Grid Search je metóda, ktorá sa používa na vyhľadanie najoptimálnejších nastavení pre modely. V tomto zadaní, sme využili už vopred vytvorenú metódu **GridSearchCV** [25]. Použili sme nasledujúce parametre.

```
'C': [0.1, 1, 100]
'gamma': [0.1, 0.01, 'scale']
'epsilon': [0.1, 0.01, 0.001]
```

```
Grid Search Completed in: 5:29:12.299380
```

Potom pomocou tejto metódy sme našli konfiguráciu, s ktorým sme dosiahli 0.875 R2 Score. Upravený výpis celého procesu nájdete v súbore **grid_search_output.txt**.

C	gamma	epsilon
100	'scale'	0.1

Tabuľka 6: Výstup metódy **SupportVectorMachine.grid_search**

Podrobnejšie výsledky nájdete nižšie.

```
Mean Squared Error (MSE) from Cross Validation:
[4.7869756 4.81861617 4.78283611 4.76204603 4.99659337]

R2 Score from Cross Validation:
[0.87342761 0.86001132 0.86808174 0.86888552 0.86341403]

Error Values for Predicted Values:

Mean Absolute Error (MAE) :
1.6386806462472463

Mean Squared Error (MSE) :
5.012881014241868

Root Mean Square Error (RMSE) :
2.238946407183939

R2 Score :
0.8753086684859772
```

2.5.3 Súborové učenie

Na konci tohto zadania sme vyskúšali rôzne formy súborového učenia. Vyskúšali sme metódy Bagging a Boosting.

2.5.3.1 Bagging

Pre túto podúlohu sme používali **BaggingRegressor** [26]. Používali sme predvolené nastavenia, s ktorými sme dosiahli nasledujúce výsledky.

```
Bagging Regression Completed in: 0:01:51.206870


Support Vector Machine Bagging - Mean Squared Error (MSE) from Cross
    Validation:
[5.1114392 5.13021003 5.21567702 5.07464629 5.18232111]

Support Vector Machine Bagging - R2 Score from Cross Validation:
[0.86484847 0.85095901 0.85614329 0.86027863 0.85833701]


Support Vector Machine Bagging - Error Values for Predicted Values:

Mean Absolute Error (MAE):
1.678529484387738

Mean Squared Error (MSE):
5.471103727304714

Root Mean Square Error (RMSE):
2.339039060662458

R2 Score:
0.8639107517870875
```

2.5.3.2 Boosting

Pre túto podúlohu sme používali **AdaBoostRegressor** [27]. Používali sme predvolené nastavenia, s ktorými sme dosiahli nasledujúce výsledky.

```
Boosting Regression Completed in: 2:01:05.449453


Support Vector Machine Boosting - Mean Squared Error (MSE) from Cross
   Validation:
[5.15004508 5.53150892 5.43760894 5.6269076 5.35181158]

Support Vector Machine Boosting - R2 Score from Cross Validation:
[0.86382769 0.83930063 0.85002205 0.8450731 0.85370385]

  Support Vector Machine Boosting - Error Values for Predicted Values:

  Mean Absolute Error (MAE):
  1.7885010064546343

  Mean Squared Error (MSE):
  5.728641134946084

  Root Mean Square Error (RMSE):
  2.3934579868771637

  R2 Score:
  0.857504718573405
```

Záver

V tomto zadaní sme sa snažili dosiahnuť čo najlepšie výsledky s rôznymi spôsobmi. Snažili sme sa dosiahnuť čo najnižšiu MSE hodnotu a naopak čo najvyššiu R2 hodnotu. Nižšie v tabuľke vidíte sumár dosiahnutých výsledkov. Na základe týchto hodnôt vieme povedať, že zo všetkých možností Náhodný Les Regressor bol najúspešnejší model zo všetkých.

Model	**Mean Squared Error**	**R2 Score**
Neural Network	5.076282240881297	0.8748374898282059
Random Forest Regressor	4.4612442852317935	0.889030182732353
Support Vector Machine (Default Settings)	5.469948907128103	0.863939476998213
Support Vector Machine (Grid Search)	5.012881014241868	0.8753086684859772
Support Vector Machine (Bagging)	5.471103727304714	0.8639107517870875
Support Vector Machine (Boosting)	5.728641134946084	0.857504718573405

Zoznam použitej literatúry

1. *pandas* [online] [cit. 2021-11-17]. Dostupné z : https://pandas.pydata.org/.
2. *Matplotlib* [online] [cit. 2021-11-17]. Dostupné z : https://matplotlib.org/.
3. *Seaborn - Statistical Data Visualization* [online] [cit. 2021-11-17]. Dostupné z : https://seaborn.pydata.org/.
4. *scikit-learn* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/.
5. *Tensorflow* [online] [cit. 2021-11-17]. Dostupné z : https://www.tensorflow.org/.
6. *Keras - Simple. Flexible. Powerful.* Dostupné tiež z: https://keras.io/.
7. *pandas.read_csv* [online] [cit. 2021-11-17]. Dostupné z : https://pandas.pydata.org/docs/reference/api/pandas.read_csv.html.
8. *pandas.DataFrame.info* [online] [cit. 2021-11-17]. Dostupné z : https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.info.html.
9. *seaborn.pairplot* [online] [cit. 2021-11-17]. Dostupné z : https://seaborn.pydata.org/generated/seaborn.pairplot.html.
10. *seaborn.histplot* [online] [cit. 2021-11-17]. Dostupné z : https://seaborn.pydata.org/generated/seaborn.histplot.html.
11. BROWNLEE, Jason. *4 automatic outlier detection algorithms in Python* [online]. 2020-08-17 [cit. 2021-11-17]. Dostupné z : https://machinelearningmastery.com/model-based-outlier-detection-and-removal-in-python/.
12. BADR, Will. *5 ways to detect outliers that every data scientist should know (python code)* [online]. Towards Data Science, 2019-03-05 [cit. 2021-11-17]. Dostupné z : https://towardsdatascience.com/5-ways-to-detect-outliers-that-every-data-scientist-should-know-python-code-70a54335a623.
13. *sklearn.ensemble.IsolationForest* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html.
14. *pandas.DataFrame.describe* [online] [cit. 2021-11-17]. Dostupné z : https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.describe.html.
15. *pandas.DataFrame.transpose* [online] [cit. 2021-11-17]. Dostupné z : https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.transpose.html.

16. *sklearn.preprocessing.StandardScaler* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html.

17. BROWNLEE, Jason. *Data Leakage in Machine Learning* [online]. 2020-08-15 [cit. 2021-11-17]. Dostupné z : https://machinelearningmastery.com/data-leakage-machine-learning/.

18. *Seaborn.countplot* [online] [cit. 2021-11-17]. Dostupné z : https://seaborn.pydata.org/generated/seaborn.countplot.html.

19. *Wordcloud* [online] [cit. 2021-11-17]. Dostupné z : https://pypi.org/project/wordcloud/.

20. *Python word clouds: How to create a word cloud* [online] [cit. 2021-11-17]. Dostupné z : https://www.datacamp.com/community/tutorials/wordcloud-python.

21. *Seaborn.barplot* [online] [cit. 2021-11-17]. Dostupné z : https://seaborn.pydata.org/generated/seaborn.barplot.html.

22. *Welcome to spotipy!* [Online] [cit. 2021-11-17]. Dostupné z : https://spotipy.readthedocs.io/en/2.19.0/.

23. PLOTLYGRAPHS. *Choropleth maps* [online]. plotlygraphs [cit. 2021-11-17]. Dostupné z : https://plotly.com/python/choropleth-maps/.

24. *sklearn.ensemble.RandomForestRegressor* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html.

25. *sklearn.model_selection.GridSearchCV* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html.

26. *sklearn.ensemble.BaggingRegressor* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingRegressor.html.

27. *sklearn.ensemble.AdaBoostRegressor* [online] [cit. 2021-11-17]. Dostupné z : https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostRegressor.html.

28. KARAGIANNAKOS, Sergios. *Best practices to write Deep learning code: Project structure, OOP, type checking and documentation* [online]. Sergios Karagiannakos, 2020-06-17 [cit. 2021-11-17]. Dostupné z : `https://theaisummer.com/best-practices-deep-learning-code/`.

Prílohy

A Štruktúra projektu

Inšpiráciu pre projektovú štruktúru sme našli na webovej stránke AI Summer [28].

configs

· Konfiguračné súbory

/config.py

· Hlavný konfiguračný súbor

data

· Dátové súbory

/spotify_test.csv

· Testovacie dáta

/spotify_train.csv

· Trénovacie dáta

dataloader

· Čítač dát

/dataloader.py

· Čítač dát

executor

· Spúšťač

/support_vector_machine_project.py

· Spúšťač zadania

models

· Modely Strojového Učenia

/neural_network.py

· Neurónová Sieť

/support_vector_machine.py

· Mechanizmus podporných vektorov

ops

· Operácie

/api_caller.py

· Zavolávač API

/evaluator.py

· Evaluátor

/plotter.py

· Vykresľovač grafov

output

· Výstupy

/plots

· Grafy

/bar_plots

· Stĺpcové Grafy

/choropleth

· Choropleth Mapy

/count_plots

· Count Ploty

/decision_trees

· Rozhodovacie Stromy

/heatmaps

· Heatmapy

/histograms

· Histogramy

/neural_network

· Neurónové Siete

/pair_plots

· Pair Ploty

/residual

· Reziduály

/word_clouds

· Word Cloudy

/processing_steps

· Pomocné tabuľky

/grid_search_output.txt

· Výstup funkcie Grid Search

utils

· Utilitné funkcie

/setup.py

· Setup metódy

/I-SUNS_-_Support_Vector_Machine_Project.pdf

· Dokumentácia - tento dokument

/main.py

· Hlavný program

/Support_Vector_Machine_Project

· Bash Script

/Support_Vector_Machine_Project.ps1

· PowerShell Script

/requirements.txt

· Zoznam požiadnaých balíčkov

B Používateľská príručka

V tejto časti práce prejdeme spôsoby, ktoré nám umožňujú spúšťať túto aplikáciu. Treba špecifikovať, ktorý skript chceme spustiť na základe operačného systému.

Linux

```
$ ./Support_Vector_Machine_Project
```

Windows

```
> .\Support_Vector_Machine_Project.ps1
```

Po behu týchto skriptov by som Vám odporúčal projekt spúšťať v PyCharme s vygenerovaným virtuálnym prostredím a nastaveným Spotify Client ID-m a Secretom.