

I-SUNS: Zadanie č.1

NEURÓNOVÉ SIETE

Vo vybranom programovacom jazyku implementujte program, ktorý bude kategorizovať piesne z platformy [Spotify](#) do kategórií podľa nálady. V tomto zadaní budete pracovať s dátami z AIS. Čas odovzdania je určený časom vloženia do AIS. Deadline pre získanie 15 bodov je **26.10.2022 o 14:00/15:00/16:00 (pred vaším cvičením)**. Každý týžden omeškania je penalizovaný stratou dvoch bodov. Zadanie je rozdelené na 3 časti:

- Načítajte dáta, predspracujte ich, natrénujte jednoduchú sieť a vyhodnoťte ju (**spolu 5b**):
 - Prezrite si stĺpce v databáze (popis stĺpcov na konci tohto zadania). Podľa popisu zistite, či sa v jednotlivých stĺpcoch nachádzajú outliery (neobvyklé hodnoty) a odstráňte ich. **0.5b**
 - Odstráňte stĺpce, ktoré sa nedajú použiť pri ďalšom spracovaní a *null* hodnoty (pozor na odstraňovanie *null* hodnôt - aby ste neprišli o príliš veľa dát, ak v stĺpci chýba príliš veľa hodnôt, zvážte, či nie je rozumnejšie odstrániť celý stĺpec). **0.5b**
 - Nečíselné stĺpce vhodne zakódujte. **0.5b**
 - Vytvorte vstupné (X) a výstupné (y) dátové množiny. Vo vhodnom pomere rozdeľte dáta na trénovaciu, validačnú a testovaciu množinu. **1b**
 - Dáta správne normalizujte alebo škálujte. **0.5b**
 - Natrénujte jednoduchú neurónovú sieť (tu Vám stačí Sklearn). **1b**
 - Natrénovanú sieť vyhodnoťte na trénovacej **A** testovacej množine:
 - * dosiahnutá úspešnosť **0.5b**,
 - * konfúzna matica **0.5b**.
- Analyzujte dataset cez EDA. Pracujte s upraveným aj pôvodným datasetom, aby ste mali k dispozícii všetky údaje (pracujte s dátami po odstránení outlierov a chýbajúcich hodnôt, ale použite hodnoty pred kódovaním, aby ste vedeli použiť slovné hodnoty). Niektoré vzťahy sa Vám budú hľadať lepšie predtým, než dáta upravíte. Nájdite aspoň 5 zaujímavých vzťahov v dátach (každý za 1b), vizualizujte ich pomocou grafov, inšpirujte sa ukážkami z iných projektov (**spolu 5b**):
 - grafy bez slovného popisu nebudú hodnotené plným počtom bodov,

- histogramy, minimálne a maximálne hodnoty nebudú hodnotené plným počtom bodov,
 - ak budú všetky nájdené vzťahy len voči cieľovej premennej, nebudú hodnotené plným počtom bodov.
- Natrénujte neurónovú sieť. Pre splnenie tejto časti zadania odporúčame použiť sofistikovanejšiu knižnicu (Keras, Pytorch, ...), prípadne doplniť funkcionalitu knižnice Sklearn tak, aby ste boli schopní splniť všetky nasledovné body (**spolu 5b**):
 - Zvoľte architektúru a nastavenia hyperparametrov tak, aby ste dosiahli pretrénovanie. Demonštrujte pomocou grafov priebehu trénovania, vyhodnotenia úspešností a konfúznej matice. **1b**
 - Odstráňte pretrénovanie tak, že do trénovania zavediete EarlyStopping pre skoré zastavenie trénovania. Demonštrujte pomocou grafov priebehu trénovania, vyhodnotenia úspešností a konfúznej matice. **1b**
 - V tomto bode by ste už mali mať správne natrénovanú sieť¹. Skúste zmeniť niektoré hyperparametre, prípadne architektúru siete (aspoň 2) tak, aby ste natrénovali sieť aspoň 5x (napr. v prvých 3 experimentoch zmeníte hodnotu parametra rýchlosti učenia, v ďalších 2 počet neurónov v skrytej vrstve siete). Body budú udelené nasledovne:
 - * Jednotlivé konfigurácie sú prehľadne zapísané v tabuľke (stačí zapísať rozdiely v experimentoch, netreba pre každý experiment uvádzať všetky parametre, ak ostali zachované). **1b**
 - * Pre všetky trénovania sú v tabuľke vyhodnotené dosiahnuté úspešnosti pre trénovaciu aj testovaciu množinu. **1b**
 - * Pre najlepšie a najhoršie trénovanie je navyše zobrazený priebeh trénovania a konfúzne matice. **2b**

Na čo si dať pozor!

- Nezabudnite zo vstupnej množiny odstrániť sledovanú výstupnú hodnotu, vo vašom prípade stĺpec *emotion* (príp. iný, ak budete robiť nepovinné úlohy).
- Pri normalizácii/škálovaní dbajte na to, aby ste pri nastavení scaler-a (prípadne manuálnom výpočte použitých minimálnych/maximálnych hodnôt, priemeru a smerodajnej odchýlky) použili trénovacie dáta a následne už nastavený scaler (príp. vypočítané hodnoty) použili pre validačnú a testovaciu množinu.

¹Správne natrénovaná sieť - nepozorovať známky pretrénovania, úspešnosť presahuje náhodnú úspešnosť (v tomto prípade 25%), na konfúznej matici sa dá pozorovať správne vyfarbená diagonála.

- Na analýzu výsledkov používame vyhodnotenie úspešnosti, konfúznú maticu, ... pre trérovacie aj testovacie dáta. Keď je v zadaní požiadavka na vyhodnotenie úspešnosti alebo konfúznej matice (prípadne inej metriky v ďalších zadaniach) vždy chceme tieto výsledky pre trérovaciu aj testovaciu množinu (nie validačnú, tú používame len na vyhodnotenie priebehu trérovania).
- Každý bod zadania musí byť zdokumentovaný, t.j. ak máte napr. demonštrovať pretrénovanie a následne ho odstrániť pridaním EarlyStopping-u, je potrebné dať do dokumentácie aj priebeh trérovania, dosiahnuté výsledky a nastavenia pred jeho použitím, inak nemáme možnosť vyhodnotiť, či ste splnili daný bod zadania.

Nepovinné úlohy

- Dobré parametre hľadajte pomocou Grid-searchu, prehľadávajte aspoň 10 rôznych kombinácií parametrov - dosiahnuté výsledky analyzujte (najlepší/najhorší výsledok, pretrénovanie, ...). **1b**
- Spracujte a použite pri EDA aj ostatné žánre zo stĺpca *genres* alebo *filtered_genres*. **1b**
- Natrénujte vašu sieť na riešenie iného problému - klasifikácia žánrov alebo obľúbenosti (využité stĺpec *popularity* na vytvorenie vlastných kategórií, netrénujte sieť na číselnú hodnotu). Natrénovanú sieť vyhodnoťte a zobrazte úspešnosti pomocou grafov. **1-3b**

Upresnenie stĺpcov

Podľa [SpotifyAPI](#):

- *danceability* $\langle 0, 1 \rangle$ - Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
- *energy* $\langle 0, 1 \rangle$ - Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.

- *loudness* $\sim \langle -60, 0 \rangle$ - The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typically range between -60 and 0 db.
- *speechiness* $\langle 0, 1 \rangle$ - Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.
- *acousticness* $\langle 0, 1 \rangle$ - A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
- *instrumentalness* $\langle 0, 1 \rangle$ - Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.
- *liveness* $\langle 0, 1 \rangle$ - Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.
- *valence* $\langle 0, 1 \rangle$ - A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).
- *tempo* - The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.
- *duration_ms* - The duration of the track in milliseconds.
- *popularity* $\langle 0, 100 \rangle$ - The popularity of a track is a value between 0 and 100, with 100 being the most popular. The popularity is calculated by algorithm and is based, in the most part, on the total number of plays the track has had and how recent

those plays are. Generally speaking, songs that are being played a lot now will have a higher popularity than songs that were played a lot in the past.

- *number_of_artists* - The number of artists who performed the track.
- *explicit* {True, False}- Whether or not the track has explicit lyrics (true = yes it does; false = no it does not OR unknown).
- *name* - The name of the track.
- *url* - The Spotify URL for the track.
- *genres* - A list of the genres the artists of the track are associated with. If not yet classified, the array is empty.
- *filtered_genres* - A list of the genres filtered based on the list of available genres seed parameter values for recommendations [linka](#).
- *top_genre* - The most frequent genre of the artists of the track.
- *emotion* {sad, happy, energetic, calm} - The emotion of the track.