DOES TRACK SEQUENCE IN USER-GENERATED PLAYLISTS MATTER?

Harald Schweiger **Emilia Parada-Cabaleiro** **Markus Schedl**
Institute of Computational Perception, Johannes Kepler University Linz (JKU), Austria
Human-centered AI Group, AI Lab, Linz Institute of Technology (LIT), Austria
harald.schweiger@jku.at emilia.parada-cabaleiro@jku.at markus.schedl@jku.at

ABSTRACT

The extent to which the sequence of tracks in music playlists matters to listeners is a disputed question, nevertheless a very important one for tasks such as music recommendation (e. g., automatic playlist generation or continuation). While several user studies already approached this question, results are largely inconsistent. In contrast, in this paper we take a data-driven approach and investigate 704,166 user-generated playlists of a major music streaming provider. In particular, we study the consistency (in terms of variance) of a variety of audio features and metadata between subsequent tracks in playlists, and we relate this variance to the corresponding variance computed on a position-independent set of tracks. Our results show that some features vary on average up to 16% less among subsequent tracks in comparison to position-independent pairs of tracks. Furthermore, we show that even pairs of tracks that lie up to 11 positions apart in the playlist are significantly more consistent in several audio features and genres. Our findings yield a better understanding of how users create playlists and will stimulate further progress in sequential music recommenders.

1. INTRODUCTION

Over the last decade, online streaming services have substantially changed the way people consume music. As a result, research on automatic playlist generation (APG) and automatic playlist continuation (APC) has gained attraction and contributed to improving machine-based creation and extension of item sequences (most commonly, music tracks), respectively. All the more as users nowadays spend over 36 % of their online music listening time on user-generated playlists, 17 % on playlists personalized by recommendation engines, and 15 % on the ones created by professional playlist curators. [1] Together with the fact that users create and share massive amounts of playlists on music streaming platforms, [2] this raises the question of how well current research understands the underlying semantics of user-generated playlists.

Most APG and APC approaches include algorithms that are capable of learning sequences [1–5] while other focus on smooth transitions [6]. However, contradictory findings from user-centered studies [7, 8], as well as from offline evaluations of sequence-aware recommenders [9, 10], impair a clear understating of whether tracks' sequential order has a meaningful role in users' listening experiences.

To narrow this research gap, the work at hand investigates directly, in a multifaceted manner, various properties shared across subsequent tracks in user-generated playlists. In contrast to other works, we argue that our conducted in-depth statistical analysis of a large set of real user-generated playlists complement findings over conclusions previously drawn from other indirect approaches, such as:

- measuring differences in recommendation accuracy for shuffled playlists [9, 10],
- comparing different machine learning approaches such as sequence aware vs. only context-aware recommenders [3],
- analyzing the effects of adding an additional re-ranking stage to the model [2, 4],
- evaluating feedback from user studies [7, 8, 11].

Against this background, we investigate the following research questions:

RQ1: Does the sequence of tracks matter in user-generated playlists? We approach this question by comparing the variance of subsequent tracks to the overall playlist variance, in terms of a variety of properties, concerning track metadata and audio features.

RQ2: For how long do the properties of one track persist on its successors? We study this question by evaluating the number of tracks that are affected by the previous ones concerning the aforementioned properties.

2. RELATED WORK

Related work can be categorized into (i) user studies investigating the quality criteria of user-generated playlists, (ii) research analyzing the difference between sequential and order-agnostic algorithms for APG or APC, and (iii) works that consider APG and APC as sequential problems, thereby, indirectly assuming the importance of track order.

[1] https://www.goodwatercap.com/thesis/understanding-spotify

[2] For instance, Spotify reports having over 4 billion playlists (https://newsroom.spotify.com/company-info/).

Concerning user studies, in the works by Kamehkhosh et al. [8, 11] users were asked to identify quality criteria of playlists. In both works participants ranked (out of 7 options) the track order as the fourth and sixth most important criteria, respectively. Although this might indicate that track order has less relevance to users than other properties, one third of the participants reordered their tracks during one of the experiments by Kamehkhosh et al. [11], which shows that (even unconsciously) track order is, to some extent, relevant to the users. Differently, in the user study conducted by Tintarev et al. [7], participants did not experience their track recommendations to be ordered. Some participants even reported that they generally use randomization for listening to their songs.

Concerning sequence-aware music recommender systems, Bonnin and Jannach [3] showed that algorithms based on sequential patterns outperform association rules. Chen et al. [12] trained a Latent Markov Embedding capable of reproducing coherency of playlists, thereby, outperforming n-gram models. Yang et al. [13] proposed an autoencoder architecture which performed better when track order was not manipulated. In contrast, Vall et al. [9, 10] investigated a recurrent neural network trained once on actual playlists and once on shuffled playlists. They showed that rank-based accuracy did not significantly change between the two settings.

Furthermore, some research acknowledged the importance of track order by directly implementing methods capable of learning track sequences. Bittner et al. [5] identified a vast support for the creation of smooth transitions in commercial DJing software, which led them to implement a system that fosters such transitions. Amongst other works related to the topic [6, 14], Jannach et al. [4] presented a two-stage approach for APC to re-rank candidates coherently with recent tracks. Similarly, Volkovs et al. [2] used a two-stage model including temporal and pairwise interactions which achieved the best score in the 2018 ACM Recommender Systems Challenge. [3]

Finally, since previous work on APG and APC mostly focus on western music, considering theoretical principles from tonal music is important when investigating tracks' transitions. Yet, in previous works the *mode* is typically considered [15] while the *key* (essential to represent *tonality* besides the *mode*), is often disregarded. Indeed, the role of *tonality*, despite its importance in the hierarchical relationships inherent of Western music, [4] has been rarely considered in the context of playlist sequentiality [5, 18].

3. DATA AND METHODOLOGY

3.1 Dataset

In order to answer the research questions, we considered the *Million Playlist Dataset* (MPD) provided by Spotify for the ACM Recommender Systems Challenge 2018. It encompasses one million user-generated playlists from US-citizens, with a length between 5 and 250 tracks, and an average length of 66.35 tracks. Overall, the playlists in the dataset contain about 2.3M unique tracks by 296K artists. The dataset includes only publicly shared playlists with at least 5 followers; thus, minimizing the risk of including collections of tracks without any musical theme which are just enjoyed by the creator.

One additional advantage of using this dataset is the coverage of high-level audio features, i. e., descriptors derived from low-level acoustic properties, that can be retrieved by the Spotify API. [5] These features have been used frequently in the literature [19–22] to analyze or recommend music. In this work, we investigate the following audio features: *acousticness* (confidence that a track contains non electronic instruments); *danceability* (how suitable a track is for dancing); *energy* (measure representing tracks' intensity and activity according to perceptual features such as dynamic range or loudness); *instrumentalness* (probability that a track does not contain vocals); *key* (indicates the tonality of the track without referring to the mode, i. e., the pitch-class); *liveness* (confidence value that indicates whether the track has been performed in presence of an audience); *loudness* (average loudness of the track in decibel); *speechiness* (measures the presence of spoken words); *mode* (indicates the scale of the track, i. e., major or minor, to which the key refers to); *tempo* (pace of the track in beats per minute); *valence* (indicates a track's hedonistic value, i. e., whether it sounds positive or negative).

In addition to the described audio features, we also take into account other three related to metadata, i. e., *artist*, *genre*, and *popularity*. As MPD provides only the main artist per track, we enrich the set of artists by retrieving for each of the 2.3M tracks, also through the Spotify API, all artists which have contributed to a track. This has been done to account for artist collaborations as possible effect of smooth transitions inside playlists. For $136,854$ of the $402,867$ artists in the enriched artist set, a set of *genres* is available. [6] We link these genres to the tracks of the playlists in order to analyze whether a shift in genres over time can be observed. Finally, the *popularity* of a track, which describes the recent average number of listening events, is retrieved by the same query as the artists.

All in all, 9 continuous features, i. e., *acousticness*, *danceability*, *energy*, *instrumentalness*, *liveness*, *loudness*, *speechiness*, *tempo*, and *valence*, as well as 5 discrete, i. e., *key*, *mode*, *genre*, *artist*, and *popularity*, are considered.

Note that some features, i. e., *acousticness*, *instrumentalness*, *liveness*, and *speechiness*, describe confidence levels rather than meaningful musical characteristics. Nevertheless, we include these features as they might still be insightful, even with their skewed distribution, towards values of 0 and 1.

From the one million playlists provided by the dataset, we filter out all playlists which have less than 30 tracks:

[3] http://www.recsyschallenge.com/2018

[4] From a music theory perspective tonal functionality models listeners' expectations, within and across songs, as shown by the tonal relationship between the different movements of unique compositions, e. g., sonatas (cf. Sonata A in [16]), whose movements' tonalities are typically related in terms of dominant, subdominant, relative, or modal relationships [17].

[5] https://developer.spotify.com/documentation/web-api/reference/#category-tracks

[6] https://everynoise.com/

this yields 704, 166 playlists with 2, 165, 065 unique tracks to be analyzed. The filtering is mainly done for **RQ2**, so that we can analyze tracks dependencies that lie up to 15 tracks apart, which is necessary since our method requires twice of this number of tracks to assure that all tracks are covered in the variance calculation presented in Section 3.2.2. As a side effect of the filtering procedure, we also minimize random noise caused by small playlists.[7]

3.2 Definitions

3.2.1 Playlist Variance

Let T be a list of n tracks $[t_1, \dots, t_n]$ forming an arbitrary playlist of our dataset. Each track t_i is assigned to a set of features where x_i denotes a single feature value, representing any of the considered Spotify features (i. e., the audio features and *popularity*). Besides, the genres and artists of each track are defined as discrete feature sets, G_i and A_i, respectively, through a bag-of-words representation.

The variance according to a feature **x** across all tracks, independently of the track order inside a playlist, is calculated as the sum of differences between the average of the feature $\overline{x}$ and all feature values x_i, as given by Equation (1). To avoid ambiguity, from now on we call this the *playlist variance*.

$$\text{pl\_var}(T) = \frac{1}{n-1}\sum_{i}^{n}(x_i - \overline{x})^2 \tag{1}$$

This formula works in cases for which the mean can be computed. However, calculating the overlap between genres and artists, e. g., with the Jaccard distance, does not provide a mean value. Similarly, the discrete features *key* and *mode* need also a different distance measurement to capture the similarities across tracks' tonalities. In these cases, the *playlist variance* can be calculated by averaging the differences of all pairwise combinations. This has been demonstrated by Zhang and Cheng [23] and it is shown in Equation (2).

$$\text{var}(T) = \frac{1}{n}\sum_{i}^{n}(x_i - \overline{x})^2 = \frac{1}{2n^2}\sum_{i}^{n}\sum_{j}^{n}(x_i - x_j)^2 \tag{2}$$

Equation (2) can now be extended by any arbitrary distance measurement $\mathcal{D}$ and since the distance w. r. t. the same track is always zero, a degree of freedom $n - 1$ is considered to compute the variances, as shown in Equation (3).

$$\text{pl\_var}(T) = \frac{1}{2n(n-1)}\sum_{i \neq j}^{n}\mathcal{D}(x_i, x_j)^2 \tag{3}$$

In order to eliminate possible correlations between repeating artists, we prevent some pairwise track combinations to be considered for calculating the playlist variance. The corresponding filter function $\mathcal{F}(A_i, A_j)$ returns 1 if all artists of A_i are different from those of A_j and 0 otherwise.

Adding the filter function $\mathcal{F}(A_i, A_j)$ to the playlist variance results in the *constrained playlist variance*, as defined by Equation (4).

$$\text{cpl\_var}(T) = \frac{\sum_{i \neq j}\mathcal{F}(A_i, A_j)\mathcal{D}(x_i, x_j)^2}{2\sum_{i \neq j}\mathcal{F}(A_i, A_j)} \tag{4}$$

3.2.2 Sequential Variance

To answer the RQs we need to compare the playlist variance with a variance eligible to account for the track order inside playlists. We will refer to this as *sequential variance*, which is the variance of a pair of tracks occurring at a fixed distance (number of tracks) apart in a given playlist. The sequential variance for all track combinations that lie d tracks apart is defined by Equation (5). Note that $d = 1$ means that the two tracks are direct neighbors.

$$\text{seq\_var}(T) = \frac{1}{2(n-d)}\sum_{i}^{n-d}\mathcal{D}(x_i, x_{i+d})^2 \tag{5}$$

Similarly as for the constrained playlist variance, to compute the *constrained sequential variance* we apply again the filter function $\mathcal{F}$ on the sequential variance, thus ignoring pairs of tracks by the same artist(s), as defined in Equation (6).

$$\text{cseq\_var}(T) = \frac{\sum_{i}^{n-d}\mathcal{F}(A_i, A_{i+d})\mathcal{D}(x_i, x_{i+d})^2}{2\sum_{i}^{n-d}\mathcal{F}(A_i, A_{i+d})} \tag{6}$$

3.2.3 Proportional Variance

To analyze the aggregated differences between playlist and sequential variance for all considered playlists in the dataset, we calculate for each track list $T \in \mathbf{D}$, where $\mathbf{D}$ denotes to the dataset, the ratio of the playlist variance to the sequential variance. From now on, we refer to it as the *unconstrained proportional variance* (UPV), by this denoting that repeating artists were not excluded. As a minor part of the tracks might present very homogeneous features, sequential variances with values close to zero can occur. Since dividing the playlist variance through these variances may yield proportional variances converging to infinity, we use the median instead of the mean to reduce the UPV values of all playlists to one average value, as shown in Equation (7).

$$\text{prop}(\mathbf{D}) = \text{median}_{T \in \mathbf{D}}\frac{\text{pl\_var}(T)}{\text{seq\_var}(T)} \tag{7}$$

Note that the *constrained proportional variance* (CPV) is calculated as shown in Equation (7) but considering the constrained versions of the playlist and sequential variance instead.

3.2.4 Feature-specific Distance Measurement

In this section, we summarize the three different distance measurements, previously denoted as $\mathcal{D}$, to calculate the variances, both the playlist variance and the sequential one:

[7] It seems that playlists in the dataset are stratified by their track size. The smallest playlists is 5 tracks long and the largest 250.

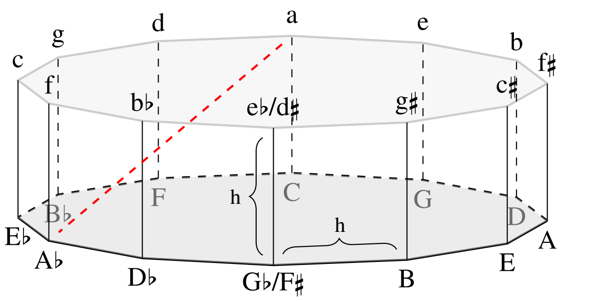

Figure 1. Visualization of the Euclidean distance (marked with the red dashed line) between 'A♭' and 'a' in $\mathbb{R}^3$. Major and minor tonalities are indicated with upper- and lowercase, respectively; ♯ stand for sharps, ♭ for flats.

(i) For all the continuous features and the discrete *popularity*, the variance is calculated by Euclidean distance.

(ii) For the discrete features *key* and *mode*, related in terms of tonality from a music theory perspective, these were considered together, as proposed by Bittner et al. [5], who maps *mode* and *key* into the three dimensional space $\mathbb{R}^3$. Keys, i. e., the pitch classes, are mapped according to the circle of fifths and represented as points onto a two dimensional unit circle. The third dimension is added by the mode (major or minor) so that *key* and *mode* are equidistant. In Figure 1 the representation of *key* and *mode* in the three dimensional space is shown. From now on, we will refer to the combination of these tow features as *tonality*.

(iii) For the overlap in artists (or genre) between two tracks, the variance is calculated by the Jaccard distance, as shown in Equation (8), where A_i and A_j represent the artist (or genre) sets of track t_i and t_j, respectively.

$$\mathcal{J}(A_i, A_j) = \frac{|A_i \cup A_j| - |A_i \cap A_j|}{|A_i \cup A_j|} \tag{8}$$

3.3 Method

To investigate the RQs, we first calculate for each playlist in the dataset the playlist variance, as defined in Section 3.2.1. The playlist variance is our baseline, which represents the variance of features irrespective of the order of the tracks. Then, the sequential variance is computed for each playlist as defined in Section 3.2.2. In contrast to the playlist variance, the sequential variance considers only tracks which lie exactly d tracks apart from each other inside the playlist.

To answer **RQ1** we choose $d = 1$, so that only features of direct neighbors, i. e., $(x_1, x_2), \ldots (x_{n-1}, x_n)$, are considered for the variance calculation. If for the majority of playlists the sequential variance is lower than the playlist variance, thus yielding a high proportional variance, i. e., above 1.0, we can conclude that users, consciously or unconsciously, create playlists with smooth transitions between tracks for the given feature under investigation. In contrast, if the sequential variance is higher than the playlist variance for a certain feature, thus yielding a low proportional variance, i. e., below 1.0, we can conclude that users tend to prefer a more rapid change for that feature. Reasons for rapid changes can be multifarious. For instance, in playlists with the purpose of dancing, a change towards slower or different music style might be used to give listeners a recovery break.

We also investigate the effects of repeating or partially overlapping artists across tracks. Assuming that artists tend to produce tracks with similar features, sequences of tracks by the same artist might bias the variances of other features, especially when correlations between artists and features are strong. Therefore, we adapted the sequential variance and playlist variance as defined by Equation (4) and Equation (6) with the constraint of excluding subsequent tracks for which artists repeat.

To answer **RQ2** we compare the playlist variance with the sequential variance for different track distances d. This enables us to assess how the features of a given track persist on the neighboring ones in relation to the distance between them. We interpret the changes in the UPV w. r. t. different track distances as defined in Section 3.2.2 and Section 3.2.3. We also compute a series of Welch's two-tailed t-tests between the playlist variances and sequential variances to identify how many consecutive tracks of a given track are affected w. r. t. the feature under consideration. Generally, track distance and significance are inversely proportional, i. e., when the former increases, the latter decreases. As soon as the two-tailed t-test returns a p-value larger than .001, we conclude that there is no significant difference between sequential and playlist variance. [8]

4. RESULTS AND DISCUSSION

For **RQ1** we first investigate in Section 4.1 the variation of subsequent tracks in comparison to the order-independent playlist variance. Next, in Section 4.2, we focus on the distribution of *loudness*, i. e., the audio feature with the largest UPV. For **RQ2** the number of tracks affected by the previous one, i. e., those for which properties characteristic of previous tracks still persist, are assessed in Section 4.3.

4.1 Quantitative Analysis of Proportional Variances

In Figure 2, the unconstrained and constrained proportional variances, i. e., UPV and CPV, respectively, as defined in Section 3.2.3, are shown.

The statistical analysis shows that *genre* seem to be the most important property influencing users in the selection of neighboring tracks, as displayed by the highest UPV, i. e., 1.159 (meaning that playlist variance exceeds sequential variance by 15.88 %); cf. UPV for *genre* in Figure 2. A high UPV indicates a low variance for neighboring tracks in comparison to the overall playlist variance for a given feature. However, as explained in Section 3.1, the tracks' genres, being the union of the corresponding artists' gen-

[8] Note that the reported results are comparable to those obtained from the non-parametric alternatives Mann-Whitney U rank and Wilcoxon signed-rank test.

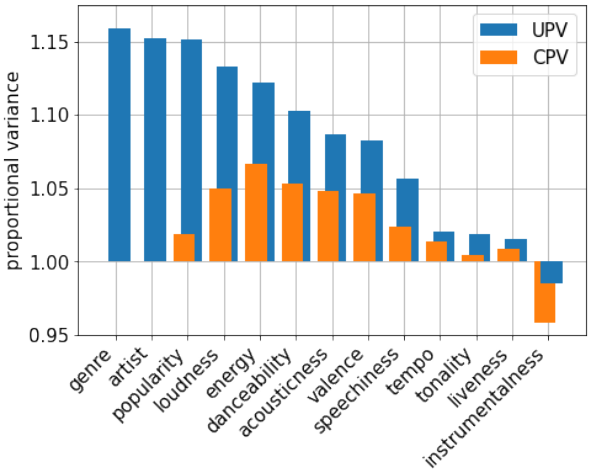

Figure 2. Bar chart representing the unconstrained proportional variances (UPV) and the constrained proportional variances (CPV) for all the analyzed features.

res, [9] are often sparse, leading to large Jaccard distances when repeating artists are filtered out, which yields a very low CPV (cf. no *visible* CPV bar for *genre* in Figure 2).

The next most important properties for low sequential variance (in relation to playlist variance) are *artist* and *popularity*, as shown by their high UPV: 1.153 and 1.151, respectively (cf. UPV for *artist* and *popularity* in Figure 2). This indicates that user-generated playlists often repeat the same artists in subsequent tracks. The proportional variance for *popularity* drops considerably, i. e., -0.133, after filtering out repeating artists: compare UPV (1.151) w. r. t. CPV (1.019) for *popularity* in Figure 2. [10] We assume this is due to tracks from famous artists, which are more common in playlists, being typically more popular; therefore, repeating artists have a large effect on tracks' popularity.

Several audio features (*loudness*, *energy*, *danceability*, *acousticness*, and *valence*) have lower but still considerable proportional variances, both in the unconstrained and the constrained setting: all of them show a UPV ≥ 1.082 and a CPV ≥ 1.046; while *tempo*, *tonality*, and *liveness* show UPV ≤ 1.021, CPV ≤ 1.013; and *speechiness* falls in between with UPV $= 1.056$, CPV $= 1.024$ (cf. UPV and CPV in Figure 2). This shows that concerning *loudness*, *energy*, *danceability*, *acousticness*, and *valence*, the majority of playlists tend to have smooth transitions between directly neighboring tracks even in cases where all artists are different from one song to another. Differently, for *tempo*, *tonality*, *liveness*, and to a lesser extent for *speechiness*, no substantial differences are displayed. Nevertheless, a deeper evaluation focusing on specific genres, such as 'classical' or 'rap', should be performed in order to understand whether the importance of these features is biased by the effect of predominant genres, e. g., 'pop' or 'rock', in which they might not have a prominent role.

Interestingly, for *instrumentalness* it can be observed that the proportional variances are below the line of equality (i. e., 1.0 on the y-axis), meaning that the sequential variance is on average larger than the playlist variance. More precisely, the UPV is 0.985 (or $-1.49\,\%$ in relation to the playlist variance). The effect is even stronger for CPV: 0.958 (or $-4.23\,\%$). This is an unexpected outcome, which might be explained by the very skewed distribution (skewness $= 3.593$) of this feature.

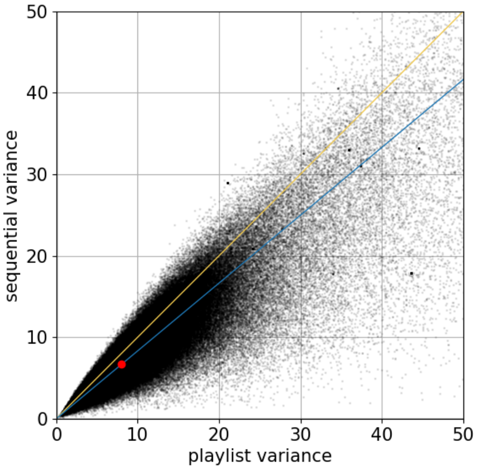

Figure 3. Scatter plot of the loudness feature. Each point represents one playlist with the playlist variance on the x-axis and the sequential variance on the y-axis. The red point marks the center of gravity. The lower line (in blue) visualizes the general differences between both variances in comparison to the line of equality (in yellow).

4.2 Visualization of Proportional Feature Differences

To visually explore the relationship between playlist and sequential variance over all playlists in the dataset, we represent each playlist as a point on a scatter plot with the x-axis corresponding to the playlist variance and the y-axis to the sequential variance of a chosen feature. Figure 3 shows the distribution of the feature *loudness*. We chose loudness as example as it is the audio feature with the largest UPV. For completeness, we provide the plots for all features as well as the source code to reproduce the experiments. [11]

Figure 3 displays that the scattered points are not symmetrically distributed along the line of equality, i. e., the diagonal (upper line) considered as reference. Most of the points fall below the line of equality, as shown by the general trend of the distribution, indicated by the lower line crossing the center of mass (large dot), which has a slope of 0.83, i. e., 39.77 degrees. This indicates that directly neighboring tracks vary less arbitrary than other tracks in the playlist, in other words, there is a large imbalance between sequential and playlist variance.

Furthermore, the effect seems to be even stronger for playlists with generally large playlist variances (cf. empty area in the upper left part compared to the lower right part of Figure 3). Thus, we conclude that the majority of user-generated playlists have a smooth change in *loudness*.

[9] There are 5, 145 genres across the whole dataset with an average of 3.17 genres per track.

[10] For obvious reasons there is no bar referring to the constrained proportion for *artist* in Figure 2.

[11] `https://gitlab.cp.jku.at/haralds/spv_analysis`

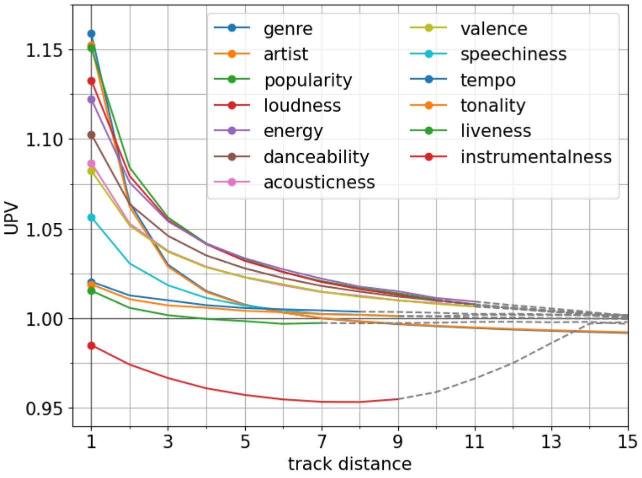

Figure 4. Visualization of the change in UPV according to track distance. The feature labels of the legend are sorted by the starting values of UVP in descending order. Gray dashed lines mark the point at which t-test between playlist and sequential variance return a p-value $\geq .001$.

4.3 Proportional Variances for Increasing Track Distances

Unlike in Section 4.1, where the sequential variance was only considered for every track and its direct neighbor, in order to answer **RQ2**, we compute now the sequential variance of tracks which lie a predefined distance d apart from each other. Figure 4 depicts along the x-axis the increasing track distance considered, whereas the y-axis shows the drop in features persistence according to the UPV defined in Section 3.2.3. Dashed lines indicate non-significant results on the t-test: as significance threshold, we consider $p \leq .001$.

It can be seen that the UPV of the analyzed features, excluding *instrumentalness*, *genre*, and *artist*, drop in a similar fashion. The larger the initial UPV (i. e., the UPV at track distance $d = 1$), the longer specific characteristics of a given feature prevail on the upcoming tracks. Generally, *energy*, *loudness*, *danceability*, *accousticness*, and *valence* are properties that significantly persist on tracks which lie up to 11 tracks apart (cf. solid lines for these features in Figure 4). Differently, the UPV drop faster for *genre* and *artist* than for the audio features, which indicates that repeating artists and overlapping genres are only important for neighboring tracks lying close to each other, i. e., within a track distance of 2 or 3. Interestingly, after around 8 tracks the lines for *genre* and *artist* drop below 1.0. This suggests that after 8 tracks it is more likely that artists and genres differ than they do not.

As mentioned in Section 4.1, the audio features *tempo*, *tonality*, and *liveness* present an initial UPV $\leq$ 1.021, which drops even further with increasing track distance (cf. Figure 4). Nevertheless, although these features do not generally show a high UPV for any track distance, they are still significant: especially *tonality*, whose characteristics persist even 9 tracks apart (cf. solid line for *tonality* in Figure 4). This suggest that these features might be important for specific genres or themes but not for the dominant ones, i. e., the most popular, whose weight could have hidden the role of these features for concrete genres in the investigated scenario. A similar trend (persisting up to 8 tracks) is shown for *speechiness*, falling in between audio features with high UPV and low UPV. The exact reason for the persisting significance but low UPV values is an open research question which will be investigated in future work.

The only outlier feature in this assessment is again, as expected by the findings described in Section 4.1, *instrumentalness*. Unexpectedly, the UPV continues to drop until a track distance of 8 is reached, afterwards it increases again. This might be explained by pronounced overlaps between artists or between genres, as well as by the skewed distribution (skewness = 3.6) of this feature. Investigating this behavior further will also be part of our future work.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we investigated to which extent audio and metadata characteristics of *subsequent* tracks in user-generated playlists differ, and we related this difference to the difference of arbitrary tracks in the playlist. For this purpose, we defined variance measures on the level of subsequent tracks (sequential variance) and on the level of an entire playlist (playlist variance). Using these measures, we analyzed both direct neighbors and tracks up to a certain distance apart in the playlist. Our major findings can be summarized as follows. (i) Metadata, i. e., *genre*, *artist*, and *popularity*, vary on average by 15.10 % more for the overall playlist variance than for order dependent sequential variance. (ii) The audio features *loudness*, *energy*, *danceability*, *acousticness*, and *valence* persist stronger over subsequent tracks at larger distances in the playlist than the metadata aspects *genre*, *artists*, and *popularity*. This effect is particularly pronounced for track distances $\geq$ 3, and specially marked for *energy*, *danceability*, *acousticness*, and *valence*, which significantly persist on average up to 11 subsequent tracks. (iii) Filtering tracks by the same artist(s) shows similar, but less pronounced results for all features, except for *genre* and *popularity*, where the difference between playlist and sequential variance almost vanishes.

Future work will include research about the content of playlists for which very large or very small UPV values are measured. This will enables us to identify possible patterns inside playlists as well as the ‘themes’ that the creator might have had in mind. We will also focus on a more profound explanation about correlations between features and will further investigate the reasons of certain outliers, e. g., *instrumentalness*. Since we are aware that the sequential relationship between tracks for some of the evaluated features, such as *key* and *mode*, might strongly depend on the musical genre, [12] a deeper evaluation on selected musical genres will also be carried out. We will ultimately leverage our findings to improve APG and APC algorithms.

[12] For instance, in classical music the sequential relationship between pieces in terms of *tonality* is stronger than in other genres, e. g., rock.

6. ACKNOWLEDGMENTS

This research received support by the Austrian Science Fund (FWF): P33526.

7. REFERENCES

[1] H. Zamani, M. Schedl, P. Lamere, and C. Chen, "An analysis of approaches taken in the ACM recsys challenge 2018 for automatic music playlist continuation," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 5, pp. 57:1–57:21, 2019. [Online]. Available: https://doi.org/10.1145/3344257

[2] M. Volkovs, H. Rai, Z. Cheng, G. Wu, Y. Lu, and S. Sanner, "Two-stage model for automatic playlist continuation at scale," in *Proceedings of the ACM Recommender Systems Challenge, RecSys Challenge 2018, Vancouver, BC, Canada, October 2, 2018*. ACM, 2018, pp. 9:1–9:6. [Online]. Available: https://doi.org/10.1145/3267471.3267480

[3] G. Bonnin and D. Jannach, "Automated generation of music playlists: Survey and experiments," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 26:1–26:35, 2014. [Online]. Available: https://doi.org/10.1145/2652481

[4] D. Jannach, L. Lerche, and I. Kamehkhosh, "Beyond "hitting the hits": Generating coherent music playlist continuations with the right tracks," in *Proceedings of the 9th ACM Conference on Recommender Systems, RecSys 2015, Vienna, Austria, September 16-20, 2015*, H. Werthner, M. Zanker, J. Golbeck, and G. Semeraro, Eds. ACM, 2015, pp. 187–194. [Online]. Available: https://doi.org/10.1145/2792838.2800182

[5] R. M. Bittner, M. Gu, G. Hernandez, E. J. Humphrey, T. Jehan, H. McCurry, and N. Montecchio, "Automatic playlist sequencing and transitions," in *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*, S. J. Cunningham, Z. Duan, X. Hu, and D. Turnbull, Eds., 2017, pp. 442–448. [Online]. Available: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/86_Paper.pdf

[6] A. Flexer, D. Schnitzer, M. Gasser, and G. Widmer, "Playlist generation using start and end songs," in *Proceedings of the 9th International Conference on Music Information Retrieval, ISMIR 2008, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*, J. P. Bello, E. Chew, and D. Turnbull, Eds., 2008, pp. 173–178. [Online]. Available: http://ismir2008.ismir.net/papers/ISMIR2008_143.pdf

[7] N. Tintarev, C. Lofi, and C. C. S. Liem, "Sequences of diverse song recommendations: An exploratory study in a commercial system," in *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, UMAP 2017, Bratislava, Slovakia, July 09 - 12, 2017*, M. Bieliková, E. Herder, F. Cena, and M. C. Desmarais, Eds. ACM, 2017, pp. 391–392. [Online]. Available: https://doi.org/10.1145/3079628.3079633

[8] I. Kamehkhosh, D. Jannach, and G. Bonnin, "How automated recommendations affect the playlist creation behavior of users," in *Joint Proceedings of the ACM IUI 2018 Workshops co-located with the 23rd ACM Conference on Intelligent User Interfaces (ACM IUI 2018), Tokyo, Japan, March 11, 2018*, ser. CEUR Workshop Proceedings, A. Said and T. Komatsu, Eds., vol. 2068. CEUR-WS.org, 2018. [Online]. Available: http://ceur-ws.org/Vol-2068/milc1.pdf

[9] A. Vall, M. Quadrana, M. Schedl, and G. Widmer, "Order, context and popularity bias in next-song recommendations," *Int. J. Multim. Inf. Retr.*, vol. 8, no. 2, pp. 101–113, 2019. [Online]. Available: https://doi.org/10.1007/s13735-019-00169-8

[10] ——, "The importance of song context and song order in automated music playlist generation," *CoRR*, vol. abs/1807.04690, 2018. [Online]. Available: http://arxiv.org/abs/1807.04690

[11] I. Kamehkhosh, G. Bonnin, and D. Jannach, "Effects of recommendations on the playlist creation behavior of users," *User Model. User Adapt. Interact.*, vol. 30, no. 2, pp. 285–322, 2020. [Online]. Available: https://doi.org/10.1007/s11257-019-09237-4

[12] S. Chen, J. L. Moore, D. Turnbull, and T. Joachims, "Playlist prediction via metric embedding," in *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012*, Q. Yang, D. Agarwal, and J. Pei, Eds. ACM, 2012, pp. 714–722. [Online]. Available: https://doi.org/10.1145/2339530.2339643

[13] H. Yang, Y. Jeong, M. Choi, and J. Lee, "MMCF: multimodal collaborative filtering for automatic playlist continuation," in *Proceedings of the ACM Recommender Systems Challenge, RecSys Challenge 2018, Vancouver, BC, Canada, October 2, 2018*. ACM, 2018, pp. 11:1–11:6. [Online]. Available: https://doi.org/10.1145/3267471.3267482

[14] B. McFee and G. R. G. Lanckriet, "The natural language of playlists," in *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, Florida, USA, October 24-28, 2011*, A. Klapuri and C. Leider, Eds. University of Miami, 2011, pp. 537–542. [Online]. Available: http://ismir2011.ismir.net/papers/PS4-11.pdf

[15] Z. Duan, L. Lu, and C. Zhang, "Audio tonality mode classification without tonic annotations," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2008, pp. 1361–1364.

[16] W. Apel, *The Harvard dictionary of music*. Cambridge, MA, USA: Harvard University Press, 2003.

[17] D. J. Grout and C. V. Palisca, *A history of Western music*. New York, NY, USA: Norton, 2001.

[18] A. M. Sarroff and M. Casey, “Modeling and predicting song adjacencies in commercial albums,” *Proc. SMC*, 2012.

[19] D. Kowald, P. Müllner, E. Zangerle, C. Bauer, M. Schedl, and E. Lex, “Support the underground: characteristics of beyond-mainstream music listeners,” *EPJ Data Sci.*, vol. 10, no. 1, p. 14, 2021. [Online]. Available: https://doi.org/10.1140/epjds/s13688-021-00268-9

[20] E. Zangerle, M. Pichl, and M. Schedl, “User models for culture-aware music recommendation: Fusing acoustic and cultural cues,” *Trans. Int. Soc. Music. Inf. Retr.*, vol. 3, no. 1, pp. 1–16, 2020. [Online]. Available: https://doi.org/10.5334/tismir.37

[21] J. S. Andersen, “Using the echo nest’s automatically extracted music features for a musicological purpose,” in *Proceedings of the 4th International Workshop on Cognitive Information Processing, CIP 2014, Copenhagen, Denmark, May 26-28, 2014*. IEEE, 2014, pp. 1–6. [Online]. Available: https://doi.org/10.1109/CIP.2014.6844510

[22] M. McVicar, T. Freeman, and T. D. Bie, “Mining the correlation between lyrical and audio features and the emergence of mood,” in *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, Florida, USA, October 24-28, 2011*, A. Klapuri and C. Leider, Eds. University of Miami, 2011, pp. 783–788. [Online]. Available: http://ismir2011.ismir.net/papers/OS9-2.pdf

[23] Y. Zhang, H. Wu, and L. Cheng, “Some new deformation formulas about variance and covariance,” in *Proceedings of International Conference on Modelling, Identification and Control*, 2012, pp. 1042–1047.