Does a song's key or danceability affects a song's popularity?

causal inference (097400)

Shoval Zandberg 205791700

Noa Shmulevich 205737935

Introduction

Music has been an integral part of our culture throughout human history. The success of musicians depends heavily on the popularity of their songs. The top 10 worldwide artists in 2019 generated a combined 1 billion dollars in revenue. Our project focuses on finding how a song's characteristics affect its popularity, focusing on the song's key and danceability. This task is particularly important in keeping businesses competitive within the growing music industry.

The ability to determine whether danceability/key affects a song's popularity can have a valuable effect on the entire music industry. Therefore, we will focus on two research questions:

R1: Does a song's danceability affect its popularity?

R2: Does a song's key affect its popularity?

To understand what categorize as a popular song, we used the "19,000 Spotify songs" dataset from Kaggle. In this dataset, the popularity of a song is represented by a continuous variable (0-100) and we will treat it accordingly. Our treatments are the song's key and danceability, which we will refer to as categorical variables.

We measured ATE score for each desired pair of treatments as described later with three different methods- S-learner, T-learner, and IPW.

Our research hypothesis was that there will be some treatment that will be globally better than others for the keys and that higher danceability leads to higher popularity. By that, we will learn what causes a song to be popular and determine if some treatments are better than others.

We discovered that for some treatment pairs we could not determine if there was a causal effect (the CIs contain the value 0, or there was a discrepancy between the methods). Therefore, we decided to measure CATE for these pairs by conditioning our data. We found that by conditioning, we could infer some additional causal effects over some pairs. We also found that for some pairs, the ATE and CATE values contradicted each other.

Dataset and Features

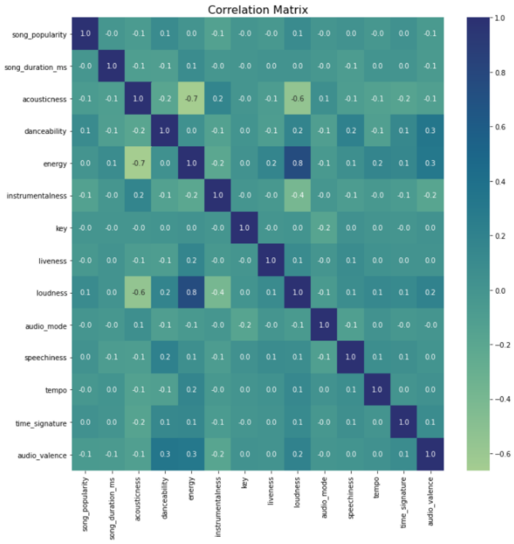
We used the '19,000 Spotify songs' dataset from Kaggle. Our data contains both acoustic features and metadata such as artist name, playlist, duration, acoustics, danceability, energy, loudness, "speechiness" (presence of a spoken word in track), audio valence, tempo, and liveness. The data set contains 16 features. These features gave us insights into the audio qualities of the track and might affect the popularity of the song. Full features explanation and primary analysis can be found in the appendix (Table 17).

Our code is available in https://github.com/shoval-z/Causal_Inference-Spotify_Songs

As part of the primary data analysis, we discovered almost no correlation between the different features to the song popularity feature (Figure 1).

Some of our features were non-numeric features and in order to properly use them in the algorithms, we transformed them into a one-hot vector. In addition, since every feature is measured on a different scale, we used Min-Max Scaler to linearly scale the data. After preprocessing the data, we were left with an additional 7564 features for the artist's name and 300 features for the playlist (a total of 7780 features).

We extracted information on the song's release date by using Genius API. For each song in our data set, we extracted the day, month, and year of release.



	song_popularity	song_duration_ms	acousticness	danceability	energy	instrumentalness	key	liveness	loudness	audio_mode	speechiness	tempo	time_signature	audio_valence
song_popularity	1.0	-0.0	-0.1	0.1	0.0	-0.1	-0.0	-0.0	0.1	-0.0	0.0	-0.0	0.0	-0.1
song_duration_ms	-0.0	1.0	-0.1	-0.1	0.1	-0.0	-0.0	0.0	0.0	-0.0	-0.1	0.0	0.0	-0.1
acousticness	-0.1	-0.1	1.0	-0.2	-0.7	0.2	-0.0	-0.1	-0.6	0.1	-0.1	-0.1	-0.2	-0.1
danceability	0.1	-0.1	-0.2	1.0	0.0	-0.1	0.0	-0.1	0.2	-0.1	0.2	-0.1	0.1	0.3
energy	0.0	0.1	-0.7	0.0	1.0	-0.2	0.0	0.2	0.8	-0.1	0.1	0.2	0.1	0.3
instrumentalness	-0.1	-0.0	0.2	-0.1	-0.2	1.0	-0.0	-0.0	-0.4	-0.0	-0.1	-0.0	-0.1	-0.2
key	-0.0	-0.0	-0.0	0.0	0.0	-0.0	1.0	-0.0	0.0	-0.2	0.0	0.0	-0.0	0.0
liveness	-0.0	0.0	-0.1	-0.1	0.2	-0.0	-0.0	1.0	0.1	-0.0	0.1	0.0	0.0	0.0
loudness	0.1	0.0	-0.6	0.2	0.8	-0.4	0.0	0.1	1.0	-0.1	0.1	0.1	0.1	0.2
audio_mode	-0.0	-0.0	0.1	-0.1	-0.1	-0.0	-0.2	-0.0	-0.1	1.0	-0.1	0.0	-0.0	-0.0
speechiness	0.0	-0.1	-0.1	0.2	0.1	-0.1	0.0	0.1	0.1	-0.1	1.0	0.1	0.1	0.0
tempo	-0.0	0.0	-0.1	-0.1	0.2	-0.0	0.0	0.0	0.1	0.0	0.1	1.0	0.0	0.0
time_signature	0.0	0.0	-0.2	0.1	0.1	-0.1	-0.0	0.0	0.1	-0.0	0.1	0.0	1.0	0.1
audio_valence	-0.1	-0.1	-0.1	0.3	0.3	-0.2	0.0	0.0	0.2	-0.0	0.0	0.0	0.1	1.0

Figure 1- Feature correlation

Some statistics of the dataset:

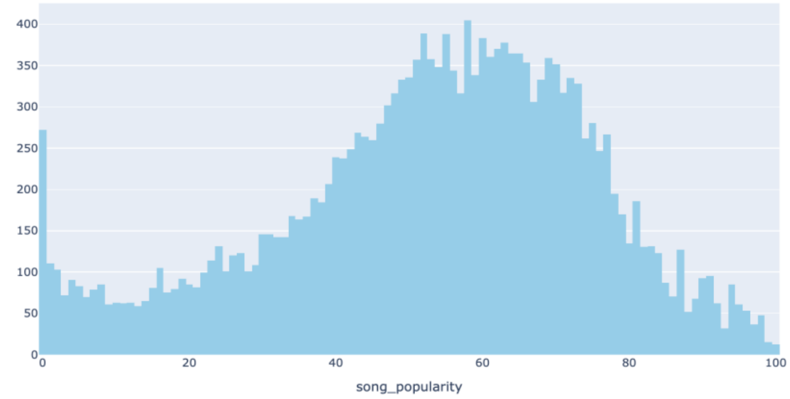


Figure 2- Song's popularity histogram

Mean popularity: 53

Median popularity: 56

We can see that the distribution is similar to the Normal distribution, most of the song's popularity is around the mean value.

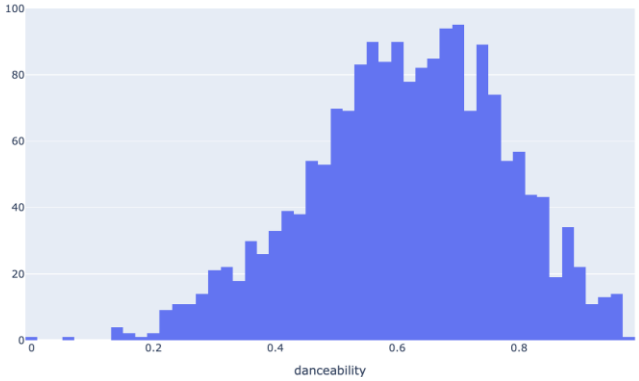


Figure 3- Song's danceability histogram

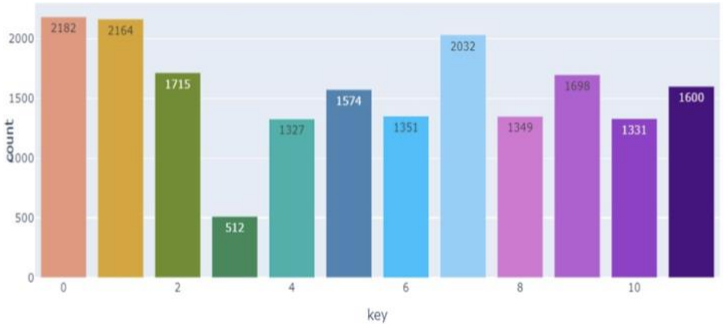


Figure 4- Song's key histogram

Challenges:

The first challenge while pre-processing the data was the size of our feature space. As part of the pre-processing, we changed the 'artist_name' and 'playlist' features into one-hot vectors, and therefore, we increased our feature space significantly. To deal with this challenge, we dropped the non-informative features (features that contain only a single value) while examining each pair (there was a different model for each pair and we used only the needed slice of the data with respect to the treatments). We also used Ridge regression that can handle large feature space well.

Another challenge was that one of our treatments (danceability), is a continuous variable (0-1). We transformed it into an ordinal variable. First, we removed the significant outliers that can be seen in Figure 3. Then, we divided danceability into different groups by looking at the deciles of the empirical distribution, as can be seen in Figure 5.

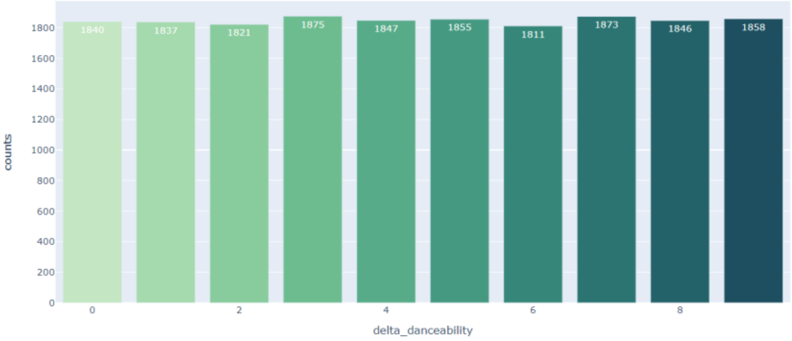


Figure 5- Song’s deciles danceability histogram

An additional challenge was that we have unbalanced data for our second treatment (key). As we can see in Figure 4. To overcome this problem, we weighted our sample while calculating the propensity score.

Moreover, we believed that the fact we have no correlation between our covariates might be a challenge as well.

Causal graph and causal inference assumptions:

As we mentioned earlier, the causal effect we are trying to measure is the influence of danceability/ key on the song’s popularity. To do so, we draw the causal graphs of our problems.

The causal graph of the first research question can be seen in Figure 6. We have two measured confounders - artist name and playlist (we believe that some artists may prefer specific danceability and the popularity of a song might be influenced by the artist - popular artists will have popular songs). Moreover, we assume that we have a hidden confounder - genre (e.g., we believe that pop is a genre with high danceability, and it is a more popular genre with popular songs). We assume this confounder depends only on the artist's name, and that the genre can influence the measured features. Also, we believe that some additional unmeasured features influence the song's popularity but do not influence the treatment (e.g., the country the song was released in, the song's writer, promotion level, etc.).

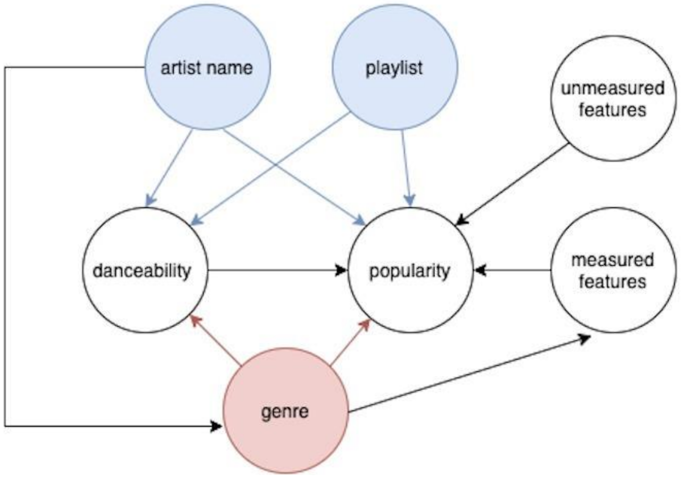


Figure 6 - Causal graph R1 (danceability)

The causal graph for the second research question can be seen in Figure 7. We have one measured confounder - artist name (we believe that Most singers have their preferred key while singing, and as mentioned before, the popularity of a song might be determined according to the artist). Again, we believe that some additional unmeasured features can influence the song's popularity.

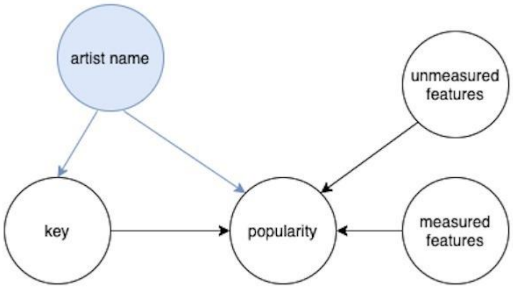


Figure 7- Causal graph R2 (key)

In our research questions we are dealing with categorical treatment and we will treat this accordingly.

In the first research question, our treatment (danceability) is an ordinal variable, therefore we compared consecutive pairs of treatment. In the second research question (the treatment is key) we compared all possible pairs of treatments.

To justify our results, we will assure that the following assumptions are plausible in our setting:

SUTVA (Stable Unit Treatment Value Assumption)- assumption holds because the outcome of a specific song does not depend on the treatment assigned to other songs, and every treatment has a single version and leads to only one possible outcome for each song.

Ignorability- the assumption means there are no hidden confounders. For R1 this assumption does not hold. In addition, we will check the back-door criterion to understand which paths are not blocked. We will mark the following nodes:

danceability: T, popularity: Y, artist name: A, playlist: B, genre: H, measured features: M, unmeasured features: U
As one can see in Figure 6, the blocked paths between popularity to danceability that end with an arrow pointing to danceability are:
$Y \leftarrow A \rightarrow T$; $Y \leftarrow B \rightarrow T$; $Y \leftarrow A \rightarrow H \rightarrow T$; $Y \leftarrow H \rightarrow A \rightarrow T$; $Y \leftarrow M \leftarrow H \rightarrow T$
(The above paths are blocked since we observed A, B, M). Unfortunately, the path $Y \leftarrow H \rightarrow T$ is not blocked since H is our hidden confounder. However, we assume that most artists have songs that belong to a limited number of genres, and therefore, we can partially infer the genre. In addition, we measure other song's features that can partially indicate the wanted genre as well.

For R2 the assumption holds, since there is only one path between popularity to the key that ends with an arrow pointing to key ($Y \leftarrow A \rightarrow T$), and this path is blocked (since we observe A).

Common support (overlap)- assumption $P(T = t|X = x) > 0\ \forall t, x$.

For both our research questions we plotted the propensity score graphs for each wanted treatments pairs as explained before. We saw that for some of the pairs this assumption is not fully met as you can see in Figure 8. To overcome this problem, we trimmed the data. We took only the samples that were in the overlap area and dropped the other samples. After

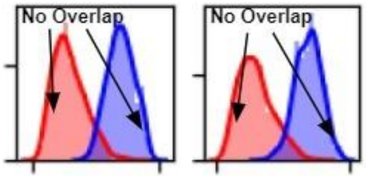


Figure 8- Propensity score example before trimming.

trimming, this assumption is fully met as you can see in Figures 9 and 10 for key and danceability correspondingly.

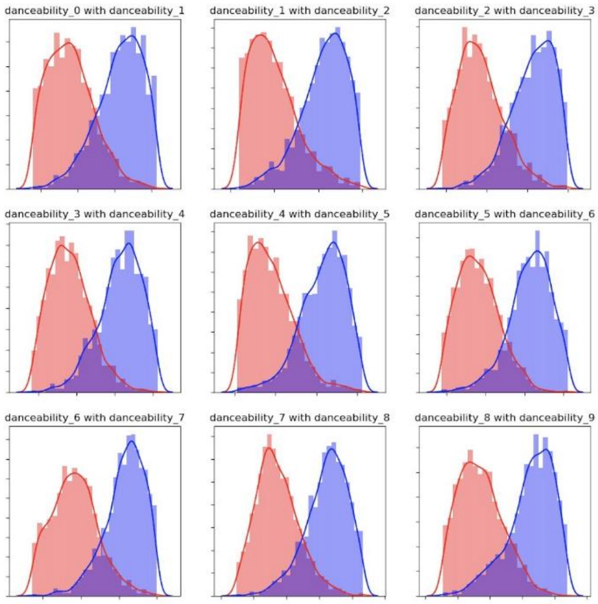


Figure 9- Propensity score after trimming (danceability)

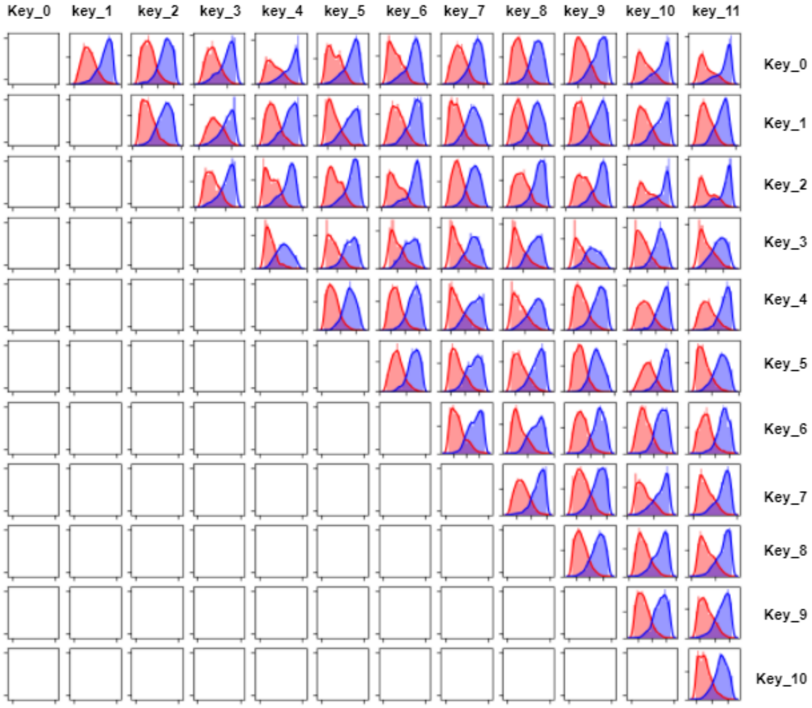
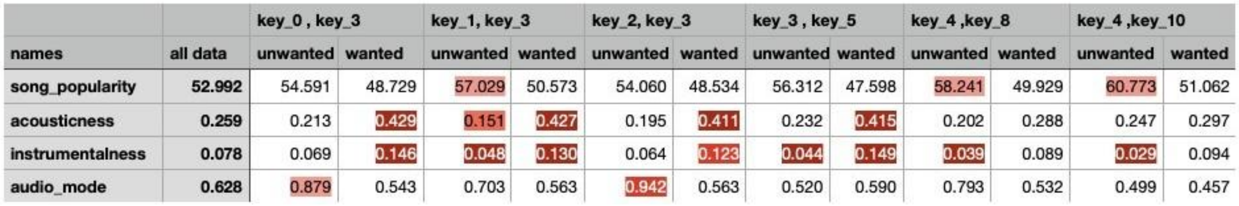


Figure 10- Propensity score after trimming (key)

We analyzed the data we dropped in the trimming process in order to see what differs the data with the extreme propensity values from the data we kept. We show the results for some of the treatment pairs as can be seen in Tables 1,2. First, we present the mean value for some of the features over the full dataset. For each pair, we present the same values for the unwanted data (the data that was removed), and for the wanted data (the data that we kept). We marked the interesting values that were not in line with the overall values. We will note that we present only the interesting features, even though we checked the entire feature space (including artist name and playlist that we initially thought would have a great influence).

Key

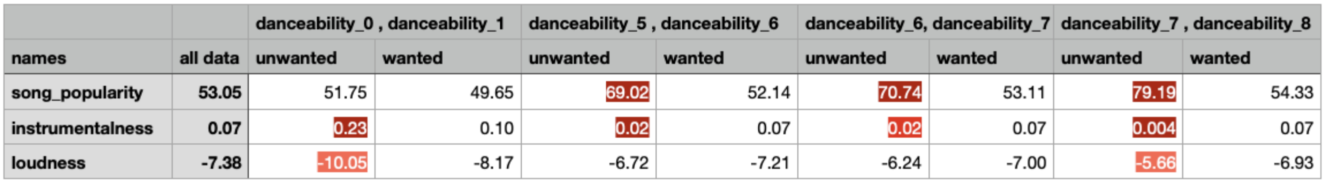


		key_0 , key_3		key_1, key_3		key_2, key_3		key_3 , key_5		key_4 ,key_8		key_4 ,key_10	
names	all data	unwanted	wanted	unwanted	wanted	unwanted	wanted	unwanted	wanted	unwanted	wanted	unwanted	wanted
song_popularity	52.992	54.591	48.729	57.029	50.573	54.060	48.534	56.312	47.598	58.241	49.929	60.773	51.062
acousticness	0.259	0.213	0.429	0.151	0.427	0.195	0.411	0.232	0.415	0.202	0.288	0.247	0.297
instrumentalness	0.078	0.069	0.146	0.048	0.130	0.064	0.123	0.044	0.149	0.039	0.089	0.029	0.094
audio_mode	0.628	0.879	0.543	0.703	0.563	0.942	0.563	0.520	0.590	0.793	0.532	0.499	0.457

Table 1- Trimming statistic (key)

In Table 1 we can see that 'acousticness' and 'Instrumentalness' are interesting features with different values. Overall, we can see that for most pairs the data that was dropped has lower 'acousticness' and 'Instrumentalness' than the data that we kept.

Danceability



		danceability_0 , danceability_1		danceability_5 , danceability_6		danceability_6, danceability_7		danceability_7 , danceability_8	
names	all data	unwanted	wanted	unwanted	wanted	unwanted	wanted	unwanted	wanted
song_popularity	53.05	51.75	49.65	69.02	52.14	70.74	53.11	79.19	54.33
instrumentalness	0.07	0.23	0.10	0.02	0.07	0.02	0.07	0.004	0.07
loudness	-7.38	-10.05	-8.17	-6.72	-7.21	-6.24	-7.00	-5.66	-6.93

Table 2- Trimming statistic (danceability)

In Table 2 we can clearly see that the dropped data's values differ in a significant manner from the mean of the entire data and from the mean of the kept data in each pair. Although we did not take the song's popularity into account in our propensity score models, we chose the present it since we can see that most of the dropped data have much higher popularity.

Methods and Results:

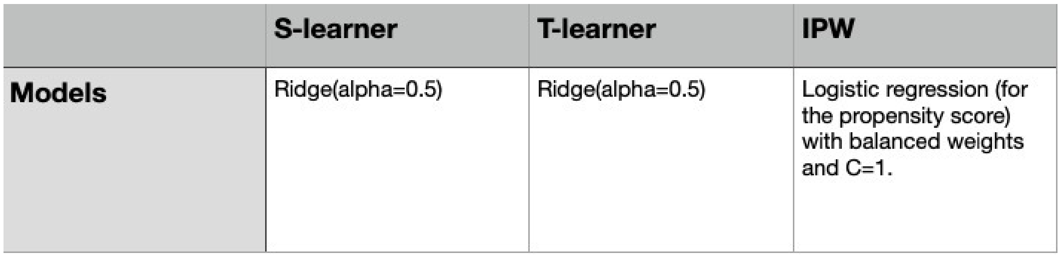
ATE

We will attempt to find the ATE value define as $ATE \coloneqq E[Y_1 - Y_0]$. We will estimate it by $\widehat{ATE} = E[Y_{obs}|T = 1] - E[Y_{obs}|T = 0]$.

Notice that:

- If $\widehat{ATE} > 0$, we conclude that the treatment causes higher popularity.
- if $\widehat{ATE} < 0$, we conclude that the treatment causes lower popularity.
- If $\widehat{ATE} = 0$, we cannot conclude any causal effect.

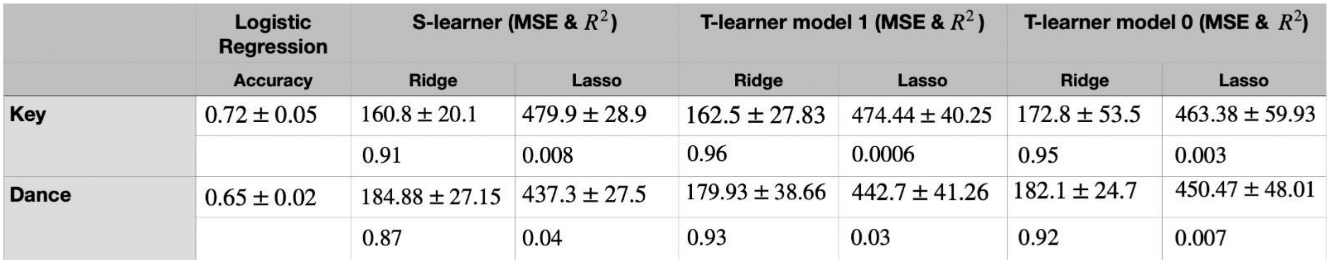
We will try to calculate the ATE for the treatment presented above. For the first research question (the treatment is danceability), we will find the ATE for each pair of following treatments (first decile with the second decile, second with the third, etc.), meaning 9 estimators. For the second research question (the treatment is key) we will find the ATE for each pair- meaning 12 choose 2 estimators. We will use several methods to calculate ATE estimators that will be presented next (chosen models and parameters are presented in Table 3).



	S-learner	T-learner	IPW
Models	Ridge(alpha=0.5)	Ridge(alpha=0.5)	Logistic regression (for the propensity score) with balanced weights and C=1.

-

Table 3 - Chosen models and parameters.



	Logistic Regression	S-learner (MSE & R^2)		T-learner model 1 (MSE & R^2)		T-learner model 0 (MSE & R^2)	
	Accuracy	Ridge	Lasso	Ridge	Lasso	Ridge	Lasso
Key	0.72 ± 0.05	160.8 ± 20.1	479.9 ± 28.9	162.5 ± 27.83	474.44 ± 40.25	172.8 ± 53.5	463.38 ± 59.93
		0.91	0.008	0.96	0.0006	0.95	0.003
Dance	0.65 ± 0.02	184.88 ± 27.15	437.3 ± 27.5	179.93 ± 38.66	442.7 ± 41.26	182.1 ± 24.7	450.47 ± 48.01
		0.87	0.04	0.93	0.03	0.92	0.007

Table 4 – Models' evaluation

We calculated the accuracy/MSE and R^2 measures in order to choose the best models for our estimators. The models' results were calculated with 5-fold Cross-validation (80% train and 20% validation) for each pair and were average over the full dataset. The results are presented in Table 4. As we can see, both MSE and R^2 suggest that Ridge is a better model than Lasso and it expaines most of the variance.

Also, to gain more confidence in the results, we calculated ATE's estimators with bootstrap and created confidence intervals based on a normal distribution with a probability of 95%.
We define $CI \coloneqq mean\ \pm z_{0.025} \cdot \frac{std}{\sqrt{n}}\ ;\quad s.t\ z_{0.025} \approx 2$.

<u>S learner-</u>

S-learner estimates the treatment effect using a single machine learning model. It defines the estimator as follow:

$$\widehat{ATE}_{S-Learner} \coloneqq \frac{1}{n}\sum_{i} f(x_i, 1) - f(x_i, 0)$$

Where f is a model for all observations when The treatment indicator is included as a feature similar to all the other features. We chose to use Ridge model (since the MSE was lower than Lasso as can be seen in Table 4).

In Tables 5 and 8, you can see the confidence intervals for both treatments.

<u>T-learner-</u>

T-learner estimates the treatment effect using two models- it does not combine the treated and control groups. It defines the estimator as follow:

$$\widehat{ATE}_{T-Learner} \coloneqq \frac{1}{n}\sum_{i} f_1(x_i) - f_0(x_i)$$

Where $f_1(x), f_0(x)$ are two separate models on treated and control samples. We chose to use Ridge model for both models (since the MSE was lower than Lasso as can be seen in Table 4).

In Tables 6 and 9 you can see the confidence intervals for treatments.

IPW-

As describe in [1], Inverse-probability weighting removes confounding by creating a “pseudo-population” in which the treatment is independent of the measured confounders. The goal is to make the sample look more like the population, it is based on the idea that the sample is not quite representative of the broader population, which might be true in our case since the data was drawn only from a few playlists from Spotify.

This approach suggests that individuals who were assigned to the treatment group, even though they were much more likely to be assigned to the control group, are rare and valuable. We want to give their outcomes as much weight as possible, whereas the much larger group of individuals who were placed in the expected treatment group need less weight, simply because we have much more information on individuals like this. It defines the estimator as follow:

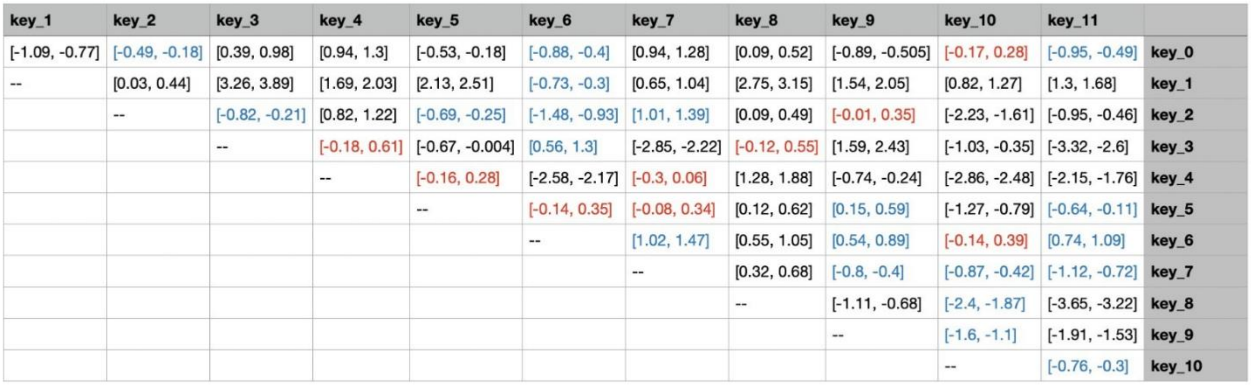
$$\widehat{ATE}_{IPW} := \frac{1}{n}\sum_i \frac{t_i \cdot y_i}{\widehat{e(x)}} - \frac{1}{n}\sum_i \frac{(1-t_i)\cdot y_i}{1-\widehat{e(x)}}$$

Where $\widehat{e(x)}$ is the propensity score which is define as $\widehat{e(x)} = P(T=1|X)$.

In Tables 7 and 10 you can see the confidence intervals for both treatments.

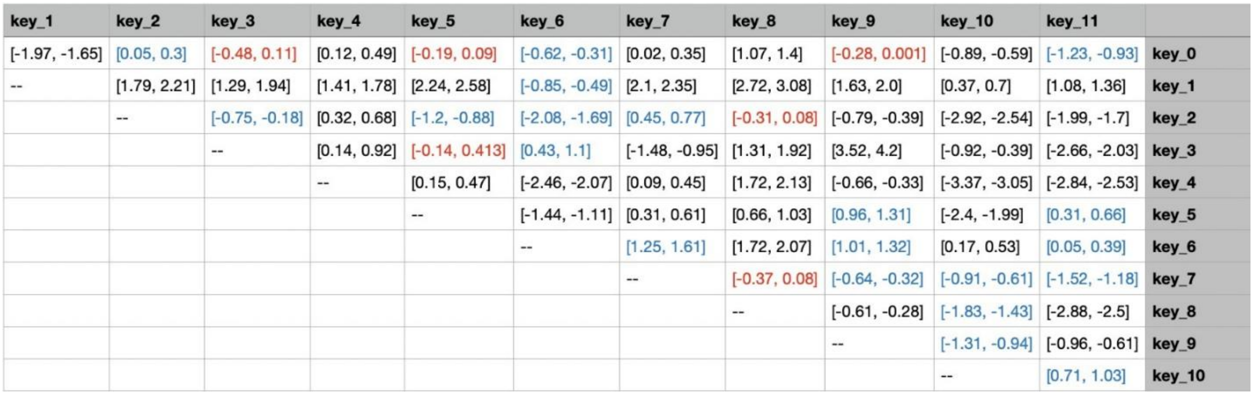
We marked the CIs that contain the value 0 in red- which means we cannot determine whether there is a causal effect. In addition, we marked the CIs of the treatments that their effect was not unanimous among methods (for example, in one method the CI was all negative, while in others it was positive).

ATE Results for treatment=key:



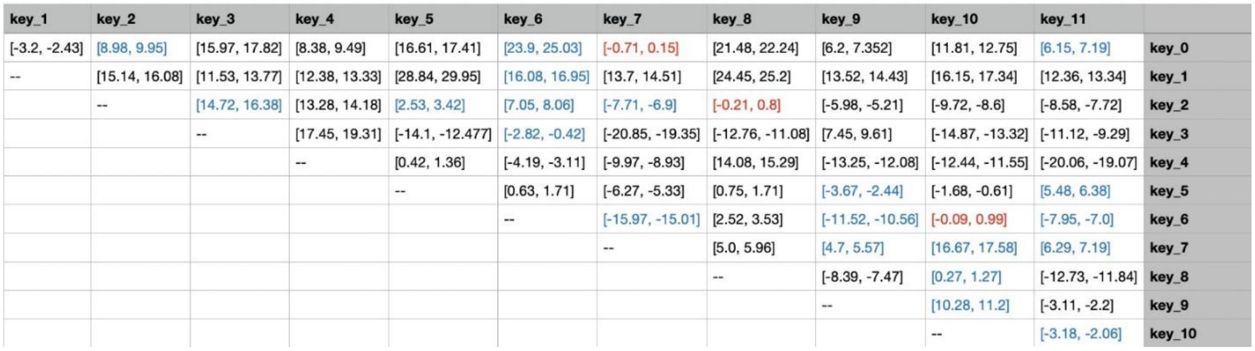
key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
[-1.09, -0.77]	[-0.49, -0.18]	[0.39, 0.98]	[0.94, 1.3]	[-0.53, -0.18]	[-0.88, -0.4]	[0.94, 1.28]	[0.09, 0.52]	[-0.89, -0.505]	[-0.17, 0.28]	[-0.95, -0.49]	key_0
--	[0.03, 0.44]	[3.26, 3.89]	[1.69, 2.03]	[2.13, 2.51]	[-0.73, -0.3]	[0.65, 1.04]	[2.75, 3.15]	[1.54, 2.05]	[0.82, 1.27]	[1.3, 1.68]	key_1
	--	[-0.82, -0.21]	[0.82, 1.22]	[-0.69, -0.25]	[-1.48, -0.93]	[1.01, 1.39]	[0.09, 0.49]	[-0.01, 0.35]	[-2.23, -1.61]	[-0.95, -0.46]	key_2
		--	[-0.18, 0.61]	[-0.67, -0.004]	[0.56, 1.3]	[-2.85, -2.22]	[-0.12, 0.55]	[1.59, 2.43]	[-1.03, -0.35]	[-3.32, -2.6]	key_3
			--	[-0.16, 0.28]	[-2.58, -2.17]	[-0.3, 0.06]	[1.28, 1.88]	[-0.74, -0.24]	[-2.86, -2.48]	[-2.15, -1.76]	key_4
				--	[-0.14, 0.35]	[-0.08, 0.34]	[0.12, 0.62]	[0.15, 0.59]	[-1.27, -0.79]	[-0.64, -0.11]	key_5
					--	[1.02, 1.47]	[0.55, 1.05]	[0.54, 0.89]	[-0.14, 0.39]	[0.74, 1.09]	key_6
						--	[0.32, 0.68]	[-0.8, -0.4]	[-0.87, -0.42]	[-1.12, -0.72]	key_7
							--	[-1.11, -0.68]	[-2.4, -1.87]	[-3.65, -3.22]	key_8
								--	[-1.6, -1.1]	[-1.91, -1.53]	key_9
									--	[-0.76, -0.3]	key_10

Table 5- S-learner CIs (key). The rows represent treatment = 1 and the columns represent treatment = 0



key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
[-1.97, -1.65]	[0.05, 0.3]	[-0.48, 0.11]	[0.12, 0.49]	[-0.19, 0.09]	[-0.62, -0.31]	[0.02, 0.35]	[1.07, 1.4]	[-0.28, 0.001]	[-0.89, -0.59]	[-1.23, -0.93]	key_0
--	[1.79, 2.21]	[1.29, 1.94]	[1.41, 1.78]	[2.24, 2.58]	[-0.85, -0.49]	[2.1, 2.35]	[2.72, 3.08]	[1.63, 2.0]	[0.37, 0.7]	[1.08, 1.36]	key_1
	--	[-0.75, -0.18]	[0.32, 0.68]	[-1.2, -0.88]	[-2.08, -1.69]	[0.45, 0.77]	[-0.31, 0.08]	[-0.79, -0.39]	[-2.92, -2.54]	[-1.99, -1.7]	key_2
		--	[0.14, 0.92]	[-0.14, 0.413]	[0.43, 1.1]	[-1.48, -0.95]	[1.31, 1.92]	[3.52, 4.2]	[-0.92, -0.39]	[-2.66, -2.03]	key_3
			--	[0.15, 0.47]	[-2.46, -2.07]	[0.09, 0.45]	[1.72, 2.13]	[-0.66, -0.33]	[-3.37, -3.05]	[-2.84, -2.53]	key_4
				--	[-1.44, -1.11]	[0.31, 0.61]	[0.66, 1.03]	[0.96, 1.31]	[-2.4, -1.99]	[0.31, 0.66]	key_5
					--	[1.25, 1.61]	[1.72, 2.07]	[1.01, 1.32]	[0.17, 0.53]	[0.05, 0.39]	key_6
						--	[-0.37, 0.08]	[-0.64, -0.32]	[-0.91, -0.61]	[-1.52, -1.18]	key_7
							--	[-0.61, -0.28]	[-1.83, -1.43]	[-2.88, -2.5]	key_8
								--	[-1.31, -0.94]	[-0.96, -0.61]	key_9
									--	[0.71, 1.03]	key_10

Table 6- T-learner CIs (key), The rows represent treatment = 1 and the columns represent treatment = 0



key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
[-3.2, -2.43]	[8.98, 9.95]	[15.97, 17.82]	[8.38, 9.49]	[16.61, 17.41]	[23.9, 25.03]	[-0.71, 0.15]	[21.48, 22.24]	[6.2, 7.352]	[11.81, 12.75]	[6.15, 7.19]	**key_0**
--	[15.14, 16.08]	[11.53, 13.77]	[12.38, 13.33]	[28.84, 29.95]	[16.08, 16.95]	[13.7, 14.51]	[24.45, 25.2]	[13.52, 14.43]	[16.15, 17.34]	[12.36, 13.34]	**key_1**
	--	[14.72, 16.38]	[13.28, 14.18]	[2.53, 3.42]	[7.05, 8.06]	[-7.71, -6.9]	[-0.21, 0.8]	[-5.98, -5.21]	[-9.72, -8.6]	[-8.58, -7.72]	**key_2**
		--	[17.45, 19.31]	[-14.1, -12.477]	[-2.82, -0.42]	[-20.85, -19.35]	[-12.76, -11.08]	[7.45, 9.61]	[-14.87, -13.32]	[-11.12, -9.29]	**key_3**
			--	[0.42, 1.36]	[-4.19, -3.11]	[-9.97, -8.93]	[14.08, 15.29]	[-13.25, -12.08]	[-12.44, -11.55]	[-20.06, -19.07]	**key_4**
				--	[0.63, 1.71]	[-6.27, -5.33]	[0.75, 1.71]	[-3.67, -2.44]	[-1.68, -0.61]	[5.48, 6.38]	**key_5**
					--	[-15.97, -15.01]	[2.52, 3.53]	[-11.52, -10.56]	[-0.09, 0.99]	[-7.95, -7.0]	**key_6**
						--	[5.0, 5.96]	[4.7, 5.57]	[16.67, 17.58]	[6.29, 7.19]	**key_7**
							--	[-8.39, -7.47]	[0.27, 1.27]	[-12.73, -11.84]	**key_8**
								--	[10.28, 11.2]	[-3.11, -2.2]	**key_9**
									--	[-3.18, -2.06]	**key_10**

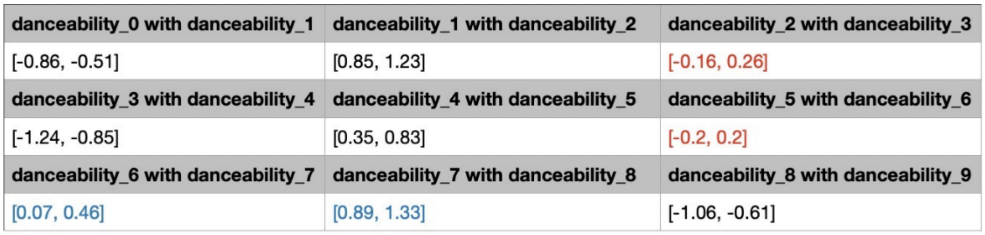
Table 7- IPW CIs (key), The rows represent treatment = 1 and the columns represent treatment = 0

Following the above results (Tables 5-7), we will attempt to find which treatments are better than others. As we can see in Tables 5 and 6, the S-learner and T-learner methods indicate similar results for most of the keys. We can see that 'key_1' and 'key_6' are better than almost all other treatments. On the other hand, we can see that 'key_8' might cause a song's popularity to decrease.

When looking at Table 7, we can see the results for the IPW method. We can see that the overall trend is slightly different from the other 2 methods, which makes sense since the other 2 methods work similarly, while the IPW calculation is very different. The CIs scale is more extreme (most of the CIs are farther from 0 than the CIs in the other 2 methods). With that in mind, we can still see that for 'key_1' and 'key_8' the overall trend is similar. In addition, IPW suggests that 'key_0' and 'key_3' are better than most of the other treatments, and 'key_4' and 'key_7' are usually worse.

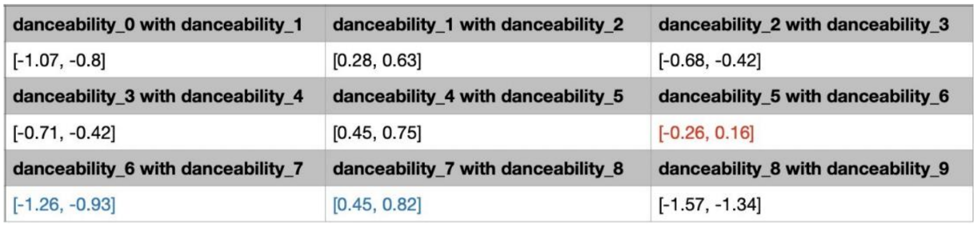
The treatment pairs marked in red (meaning it is not possible to determine unequivocally whether there is a causal effect) are different between all methods. Overall, although each method found some superior keys, we can contradict some of these keys' influence with another method.

ATE Results for treatment=danceability:



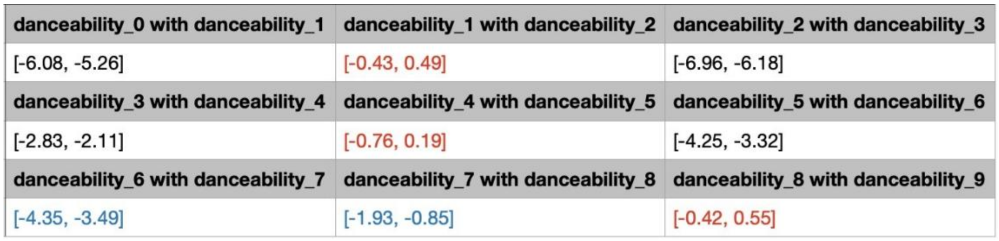
danceability_0 with danceability_1	**danceability_1 with danceability_2**	**danceability_2 with danceability_3**
[-0.86, -0.51]	[0.85, 1.23]	[-0.16, 0.26]
danceability_3 with danceability_4	**danceability_4 with danceability_5**	**danceability_5 with danceability_6**
[-1.24, -0.85]	[0.35, 0.83]	[-0.2, 0.2]
danceability_6 with danceability_7	**danceability_7 with danceability_8**	**danceability_8 with danceability_9**
[0.07, 0.46]	[0.89, 1.33]	[-1.06, -0.61]

Table 8- S-learner CIs (danceability) for each couple, the first one represents treatment = 1



danceability_0 with danceability_1	**danceability_1 with danceability_2**	**danceability_2 with danceability_3**
[-1.07, -0.8]	[0.28, 0.63]	[-0.68, -0.42]
danceability_3 with danceability_4	**danceability_4 with danceability_5**	**danceability_5 with danceability_6**
[-0.71, -0.42]	[0.45, 0.75]	[-0.26, 0.16]
danceability_6 with danceability_7	**danceability_7 with danceability_8**	**danceability_8 with danceability_9**
[-1.26, -0.93]	[0.45, 0.82]	[-1.57, -1.34]

Table 9- T-learner CIs (danceability), for each couple, the first one represents treatment = 1



danceability_0 with danceability_1	danceability_1 with danceability_2	danceability_2 with danceability_3
[-6.08, -5.26]	[-0.43, 0.49]	[-6.96, -6.18]
danceability_3 with danceability_4	**danceability_4 with danceability_5**	**danceability_5 with danceability_6**
[-2.83, -2.11]	[-0.76, 0.19]	[-4.25, -3.32]
danceability_6 with danceability_7	**danceability_7 with danceability_8**	**danceability_8 with danceability_9**
[-4.35, -3.49]	[-1.93, -0.85]	[-0.42, 0.55]

Table 10 - IPW CIs (danceability) for each couple, the first one represents treatment = 1

Following the above results (Tables 8-10), we will attempt to find which treatments are better than others. Overall, we did not identify a trend that suggests that higher\lower danceability is better. The 3 methods agree when it comes to 'danceability_0' with 'danceability_1' and 'danceability_3' with 'danceability_4'- ' danceability_1' and 'danceability_4' leads to higher popularity, respectively.

However, there are 2 pairs that the conclusion is inconclusive throughout the methods. We can see that for 'danceability_7' it is very hard to determine whether it is a good treatment or not. If we are looking at the pair 'danceability_6' and 'danceability_7', we can conclude from the S-learner estimator that the 'danceability_6' is better than 'danceability_7', while the T-learner and IPW estimates suggest the opposite conclusion. Similarly, we can conclude from the IPW estimator that the 'danceability_8' is better than 'danceability_7', while the T-learner and S-learner estimates suggest the opposite conclusion.

For the other pairs, the overall trend among methods is similar (even though some cannot determine whether there is an effect or not). The treatment pairs marked in red (meaning It is not possible to determine unequivocally whether there is a causal effect) are similar between the S-learner and T-learner methods but differ from the IPW method.

CATE:

After looking at the ATE results, we decided to move forward with the treatments pairs that we could not determine unequivocally if there is a causal effect (meaning that the pair's CI contains 0 for at least one of the methods), or for those with different conclusions from the different methods. We wanted to check if, by conditioning on the data, we will get a causal effect. We present the histogram of the number of artists with X songs in the dataset in Figure 10. We decided that we will look only at artists with at least 3 songs since it looks like a reasonable cut, and by that, we can remove a lot of "noise" from our dataset (rare artist) and significantly decrease our feature space dimension (decrease the amount of artist from ~7560 to ~1210). We will use the same methods as before to calculate CATE estimators, so we can compare the results of the ATE estimators to the CATE estimators.

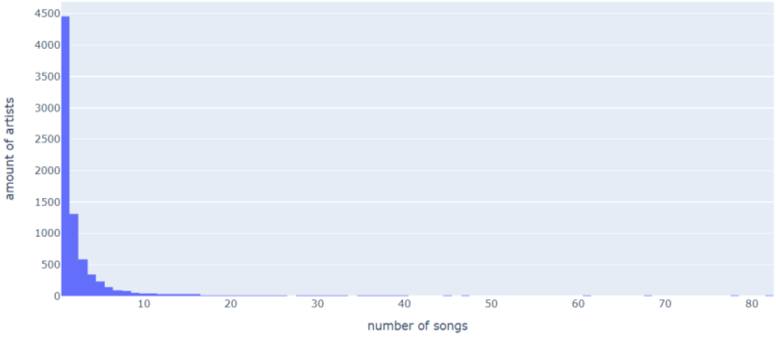


Figure 11 – Number of artists by amount of songs in the dataset

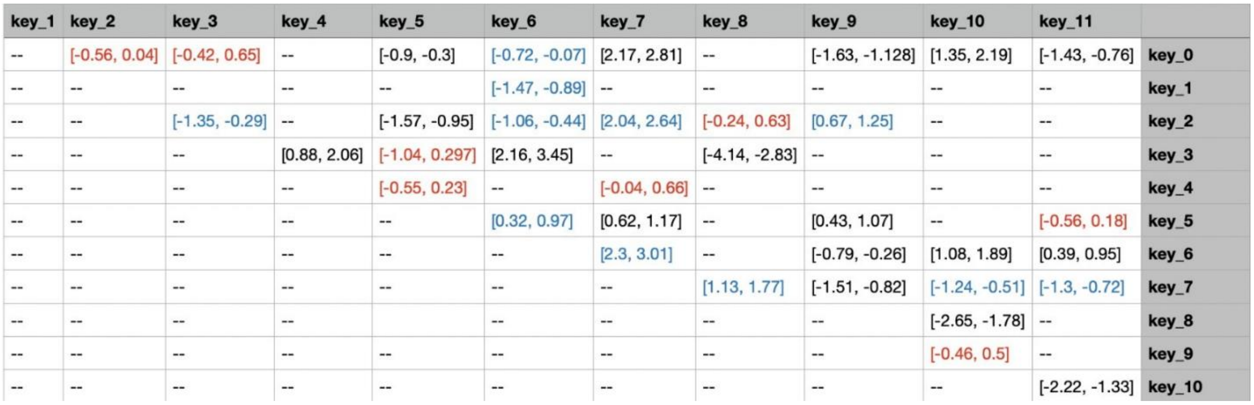
CATE will be calculated by the next formula:

$$CATE \coloneqq E[Y_1 - Y_0|X], \widehat{CATE} = E[Y_{obs}|T = 1, X] - E[Y_{obs}|T = 0, X]$$

We will note that the preprocessing for this estimator is the same as before (including the trimming to ensure that the common support assumption holds).

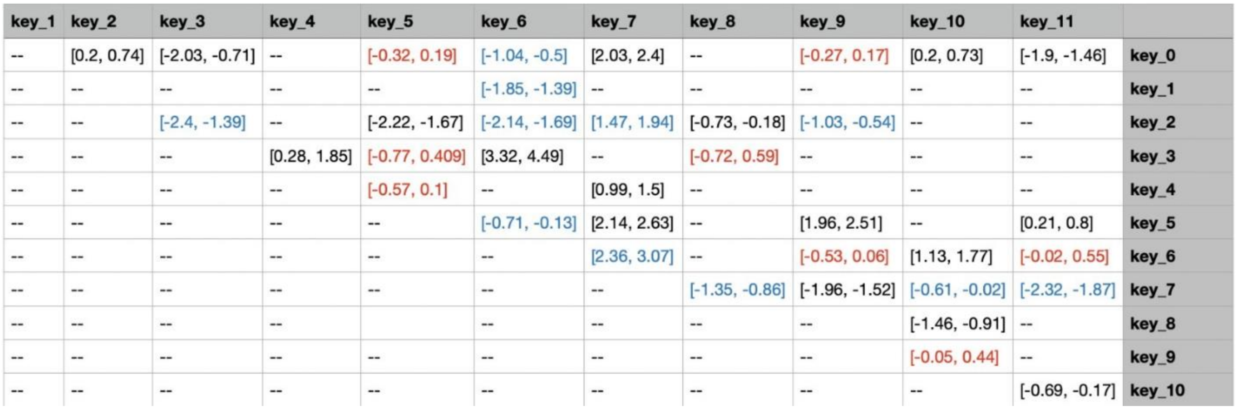
We first examine the size of the remaining dataset for each wanted treatment pairs, to make sure that we will not attempt to conclude insights on very little data. Overall, we have enough data for most of the pairs to conclude whether there is an effect. However, there are a few pairs with little data, and we will consider it while concluding (full table sizes for both key and danceability can be found in the appendix- Tables 18 and 19). The different colors have the same meaning as in the previous section.

CATE Results for treatment=key:



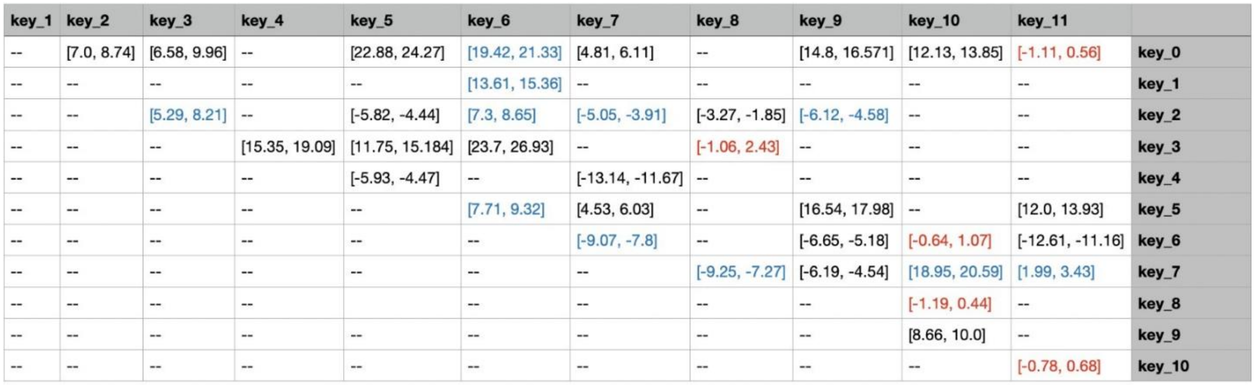
key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
--	[-0.56, 0.04]	[-0.42, 0.65]	--	[-0.9, -0.3]	[-0.72, -0.07]	[2.17, 2.81]	--	[-1.63, -1.128]	[1.35, 2.19]	[-1.43, -0.76]	key_0
--	--	--	--	--	[-1.47, -0.89]	--	--	--	--	--	key_1
--	--	[-1.35, -0.29]	--	[-1.57, -0.95]	[-1.06, -0.44]	[2.04, 2.64]	[-0.24, 0.63]	[0.67, 1.25]	--	--	key_2
--	--	--	[0.88, 2.06]	[-1.04, 0.297]	[2.16, 3.45]	--	[-4.14, -2.83]	--	--	--	key_3
--	--	--	--	[-0.55, 0.23]	--	[-0.04, 0.66]	--	--	--	--	key_4
--	--	--	--	--	[0.32, 0.97]	[0.62, 1.17]	--	[0.43, 1.07]	--	[-0.56, 0.18]	key_5
--	--	--	--	--	--	[2.3, 3.01]	--	[-0.79, -0.26]	[1.08, 1.89]	[0.39, 0.95]	key_6
--	--	--	--	--	--	--	[1.13, 1.77]	[-1.51, -0.82]	[-1.24, -0.51]	[-1.3, -0.72]	key_7
--	--	--	--		--	--	--	--	[-2.65, -1.78]	--	key_8
--	--	--	--	--	--	--	--	--	[-0.46, 0.5]	--	key_9
--	--	--	--	--	--	--	--	--	--	[-2.22, -1.33]	key_10

Table 11 - CATE S-learner CIs (key), The rows represent treatment = 1 and the columns represent treatment = 0



key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
--	[0.2, 0.74]	[-2.03, -0.71]	--	[-0.32, 0.19]	[-1.04, -0.5]	[2.03, 2.4]	--	[-0.27, 0.17]	[0.2, 0.73]	[-1.9, -1.46]	key_0
--	--	--	--	--	[-1.85, -1.39]	--	--	--	--	--	key_1
--	--	[-2.4, -1.39]	--	[-2.22, -1.67]	[-2.14, -1.69]	[1.47, 1.94]	[-0.73, -0.18]	[-1.03, -0.54]	--	--	key_2
--	--	--	[0.28, 1.85]	[-0.77, 0.409]	[3.32, 4.49]	--	[-0.72, 0.59]	--	--	--	key_3
--	--	--	--	[-0.57, 0.1]	--	[0.99, 1.5]	--	--	--	--	key_4
--	--	--	--	--	[-0.71, -0.13]	[2.14, 2.63]	--	[1.96, 2.51]	--	[0.21, 0.8]	key_5
--	--	--	--	--	--	[2.36, 3.07]	--	[-0.53, 0.06]	[1.13, 1.77]	[-0.02, 0.55]	key_6
--	--	--	--	--	--	--	[-1.35, -0.86]	[-1.96, -1.52]	[-0.61, -0.02]	[-2.32, -1.87]	key_7
--	--	--	--		--	--	--	--	[-1.46, -0.91]	--	key_8
--	--	--	--	--	--	--	--	--	[-0.05, 0.44]	--	key_9
--	--	--	--	--	--	--	--	--	--	[-0.69, -0.17]	key_10

Table 12 - CATE T-learner CIs (key), The rows represent treatment = 1 and the columns represent treatment = 0



key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
--	[7.0, 8.74]	[6.58, 9.96]	--	[22.88, 24.27]	[19.42, 21.33]	[4.81, 6.11]	--	[14.8, 16.571]	[12.13, 13.85]	[-1.11, 0.56]	key_0
--	--	--	--	--	[13.61, 15.36]	--	--	--	--	--	key_1
--	--	[5.29, 8.21]	--	[-5.82, -4.44]	[7.3, 8.65]	[-5.05, -3.91]	[-3.27, -1.85]	[-6.12, -4.58]	--	--	key_2
--	--	--	[15.35, 19.09]	[11.75, 15.184]	[23.7, 26.93]	--	[-1.06, 2.43]	--	--	--	key_3
--	--	--	--	[-5.93, -4.47]	--	[-13.14, -11.67]	--	--	--	--	key_4
--	--	--	--	--	[7.71, 9.32]	[4.53, 6.03]	--	[16.54, 17.98]	--	[12.0, 13.93]	key_5
--	--	--	--	--	--	[-9.07, -7.8]	--	[-6.65, -5.18]	[-0.64, 1.07]	[-12.61, -11.16]	key_6
--	--	--	--	--	--	--	[-9.25, -7.27]	[-6.19, -4.54]	[18.95, 20.59]	[1.99, 3.43]	key_7
--	--	--	--		--	--	--	--	[-1.19, 0.44]	--	key_8
--	--	--	--	--	--	--	--	--	[8.66, 10.0]	--	key_9
--	--	--	--	--	--	--	--	--	--	[-0.78, 0.68]	key_10

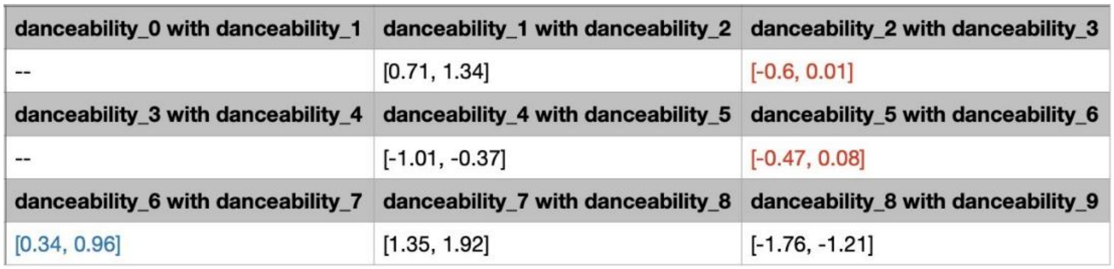
Table 13- CATE IPW CIs (key), The rows represent treatment = 1 and the columns represent treatment = 0

In Tables 11-13, we can see that by conditioning on the data, we were able to conclude causal effect over some treatment pairs that were not unanimous before. Once again, we can see that from both T-learner and S-learner we draw similar conclusions, while the IPW leads to a slightly different conclusion. As before, The CIs scale for IPW method is more extreme (most of the CIs are farther from 0 than the CIs in the other 2 methods).

In addition, we can see that it is hard to conclude that some treatment is superior to others in this part, which makes sense since we looked only at the uncertain pairs.

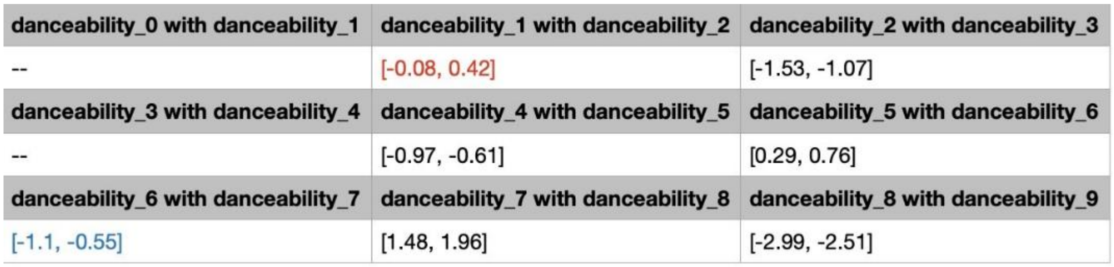
We will note that for the pairs that contain 'key_3' the conclusions should not be taken with the same confidence as the other conclusions since the datasets are much smaller.

CATE Results for treatment= danceability:



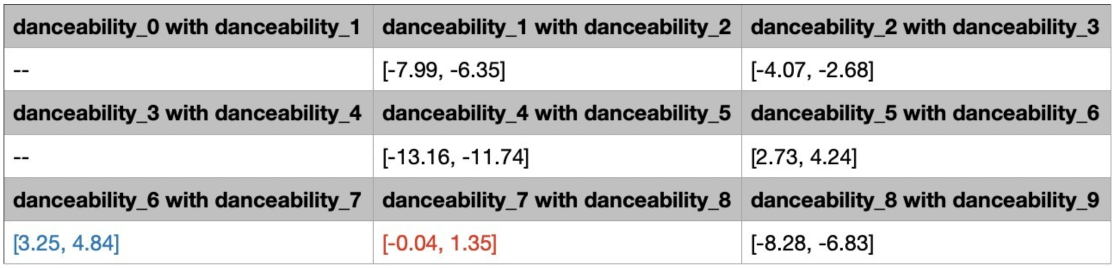
danceability_0 with danceability_1	danceability_1 with danceability_2	danceability_2 with danceability_3
--	[0.71, 1.34]	[-0.6, 0.01]
danceability_3 with danceability_4	**danceability_4 with danceability_5**	**danceability_5 with danceability_6**
--	[-1.01, -0.37]	[-0.47, 0.08]
danceability_6 with danceability_7	**danceability_7 with danceability_8**	**danceability_8 with danceability_9**
[0.34, 0.96]	[1.35, 1.92]	[-1.76, -1.21]

Table 14- CATE S-learner CIs (danceability), for each couple, the first one represents treatment = 1



danceability_0 with danceability_1	danceability_1 with danceability_2	danceability_2 with danceability_3
--	[-0.08, 0.42]	[-1.53, -1.07]
danceability_3 with danceability_4	**danceability_4 with danceability_5**	**danceability_5 with danceability_6**
--	[-0.97, -0.61]	[0.29, 0.76]
danceability_6 with danceability_7	**danceability_7 with danceability_8**	**danceability_8 with danceability_9**
[-1.1, -0.55]	[1.48, 1.96]	[-2.99, -2.51]

Table 15 - CATE T-learner CIs (danceability), for each couple, the first one represents treatment = 1



danceability_0 with danceability_1	danceability_1 with danceability_2	danceability_2 with danceability_3
--	[-7.99, -6.35]	[-4.07, -2.68]
danceability_3 with danceability_4	**danceability_4 with danceability_5**	**danceability_5 with danceability_6**
--	[-13.16, -11.74]	[2.73, 4.24]
danceability_6 with danceability_7	**danceability_7 with danceability_8**	**danceability_8 with danceability_9**
[3.25, 4.84]	[-0.04, 1.35]	[-8.28, -6.83]

Table 16 - CATE IPW CIs (danceability), for each couple, the first one represents treatment = 1

In Tables 14-16, we can see that by conditioning on the data, we were able to conclude for both 'danceability_ 4' with ' danceability_5' and ' danceability_8' with ' danceability_9', that higher danceability is better (leads to higher popularity). T-learner and IPW were more certain about pairs that they were not certain about before, while S-learner behaves as before.

- *For both treatments key and danceability, by comparing the ATE results from Tables 5-10 to the current CATE results from Tables 11-16, we can see that all methods changed their conclusions for some pairs. This might be because some of the 'certain' CIs were close to 0, meaning the difference between $E[Y_{obs}|T=1]$ to $E[Y_{obs}|T=0]$, is not very significant. Therefore, after filtering some data, we got a new conclusion.*

Possible drawbacks-

A possible drawback is the models and hyperparameters that were chosen for the different methods. The model selection might influence our conclusions and results, and as we know, it is hard to evaluate how accurate the models are in causal inference. After partitioning the data for each treatment pairs and filtering it, we were left with a much smaller dataset, which made it hard to examine a more complex algorithm such as neural network.

In addition, our outcome is a continuous variable that comes from a normal distribution, which means that most of the samples' popularity concentrates around the mean value and it may affect our results and predictions. We used Ridge regression, which is a variant of linear regression. This model relies on the linearity assumption that might not hold in our case.

One of the unaddressed issues while analyzing the result for the second treatment (danceability), was our hidden confounder. We made some assumptions that helped us believe that the hidden confounder's influence on our result will not be significant, but our assumptions might be wrong.

Moreover, we might have made the wrong modeling of the causal graph. Although we measured correlation and tried to think of all possible variables, the process was done manually with no professional domain knowledge.

Discussion-

Throughout this work, we showed various methods for estimating ATE and CATE for the research questions presented. The propensity-based method (IPW) implied a more significant causal effect for some keys and danceability. A reasonable explanation for this result can be the significant difference in the propensity scores between the treatments. This difference might lead to greater differences in the expectations $E[Y_{obs}|T=1]$ and $E[Y_{obs}|T=0]$ since in the IPW formula we divide our results by these scores. Moreover, if our treatment is unbalanced it might affect the IPW estimator significantly.

In addition, the other methods rely on the different models that tried to predict the song's popularity, which as we saw did not give promising results. As described in [2], we will prefer to use S-learner if we do not have a treatment effect since it sometimes does not split on the treatment indicator at all and it tends to be biased toward zero. On the other hand, we will prefer to use T-learner when there is nothing to be learned from the treatment group about the control group and vice versa. It is hard to determine which method is preferred in this case, but there are some treatment pairs that the methods agreed on.

Future work-

Overall, we can say that what causes a song to be more popular than others remains mostly a mystery to us. There was no global treatment that was always better than others, but there are some treatments that were locally preferred over others.

As future work, we would like to expand this project by adding an external dataset that will add some additional features such as our hidden confounder genre. We would also like to add additional samples that will help us represent the "songs population" better. The additional samples will also help us to apply powerful models such as neural networks.

In addition, we only investigated two possible treatments, but we can expand our work by examining other known features as treatments and their causal effect on the song's popularity. We could also try to add additional methods that might better suit this task (such as R-learner or X-learner).

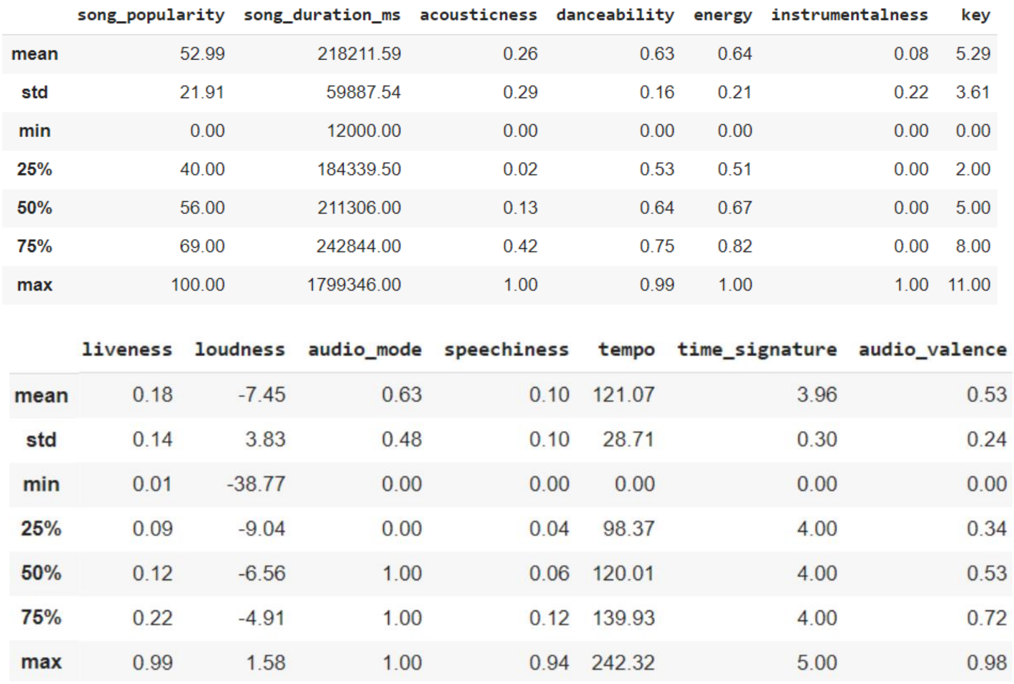
References:

[1] The intuition behind inverse probability weighting in causal inference

[2] Meta-learners for Estimating Heterogeneous Treatment Effects using Machine Learning Kunzel et al. 2019

Appendix:

Initial data analysis:

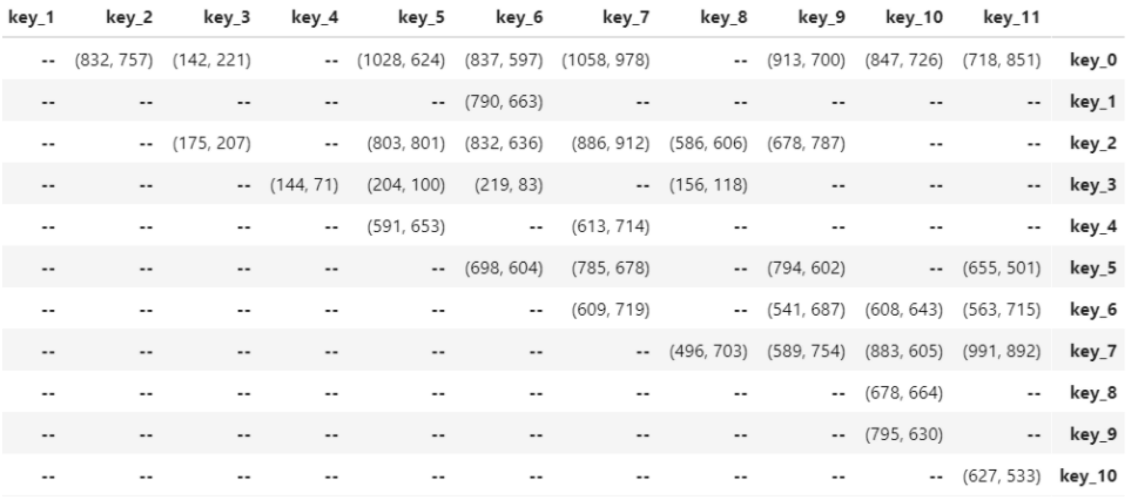


	song_popularity	song_duration_ms	acousticness	danceability	energy	instrumentalness	key
mean	52.99	218211.59	0.26	0.63	0.64	0.08	5.29
std	21.91	59887.54	0.29	0.16	0.21	0.22	3.61
min	0.00	12000.00	0.00	0.00	0.00	0.00	0.00
25%	40.00	184339.50	0.02	0.53	0.51	0.00	2.00
50%	56.00	211306.00	0.13	0.64	0.67	0.00	5.00
75%	69.00	242844.00	0.42	0.75	0.82	0.00	8.00
max	100.00	1799346.00	1.00	0.99	1.00	1.00	11.00

	liveness	loudness	audio_mode	speechiness	tempo	time_signature	audio_valence
mean	0.18	-7.45	0.63	0.10	121.07	3.96	0.53
std	0.14	3.83	0.48	0.10	28.71	0.30	0.24
min	0.01	-38.77	0.00	0.00	0.00	0.00	0.00
25%	0.09	-9.04	0.00	0.04	98.37	4.00	0.34
50%	0.12	-6.56	1.00	0.06	120.01	4.00	0.53
75%	0.22	-4.91	1.00	0.12	139.93	4.00	0.72
max	0.99	1.58	1.00	0.94	242.32	5.00	0.98

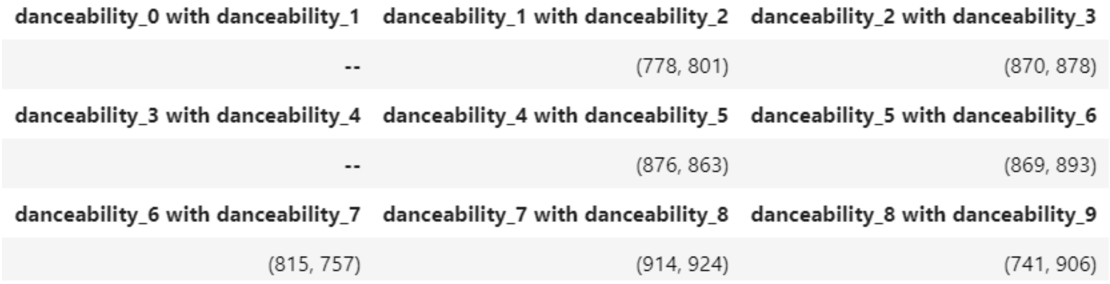
Dataset size for CATE:

Table 17 – Initial data analysis



key_1	key_2	key_3	key_4	key_5	key_6	key_7	key_8	key_9	key_10	key_11	
--	(832, 757)	(142, 221)	--	(1028, 624)	(837, 597)	(1058, 978)	--	(913, 700)	(847, 726)	(718, 851)	**key_0**
--	--	--	--	--	(790, 663)	--	--	--	--	--	**key_1**
--	--	(175, 207)	--	(803, 801)	(832, 636)	(886, 912)	(586, 606)	(678, 787)	--	--	**key_2**
--	--	--	(144, 71)	(204, 100)	(219, 83)	--	(156, 118)	--	--	--	**key_3**
--	--	--	--	(591, 653)	--	(613, 714)	--	--	--	--	**key_4**
--	--	--	--	--	(698, 604)	(785, 678)	--	(794, 602)	--	(655, 501)	**key_5**
--	--	--	--	--	--	(609, 719)	--	(541, 687)	(608, 643)	(563, 715)	**key_6**
--	--	--	--	--	--	--	(496, 703)	(589, 754)	(883, 605)	(991, 892)	**key_7**
--	--	--	--	--	--	--	--	--	(678, 664)	--	**key_8**
--	--	--	--	--	--	--	--	--	(795, 630)	--	**key_9**
--	--	--	--	--	--	--	--	--	--	(627, 533)	**key_10**

Table 18- Dataset size for each pair. the first value represent treatment 1 (row)



danceability_0 with danceability_1	danceability_1 with danceability_2	danceability_2 with danceability_3
--	(778, 801)	(870, 878)

danceability_3 with danceability_4	danceability_4 with danceability_5	danceability_5 with danceability_6
--	(876, 863)	(869, 893)

danceability_6 with danceability_7	danceability_7 with danceability_8	danceability_8 with danceability_9
(815, 757)	(914, 924)	(741, 906)

Table 19- Dataset size for each pair. the first value represent treatment 1 (the left treatment)