Spotify Prediction Final Report

CS-GY 6053 Team 9

Varshitha Chennamsetti, vc2209

Claudia Shao, zs1282

Winnie Zheng, wz1509

What is the problem?

Spotify is the world's biggest music streaming platform with over 430 million active listeners. To keep users constantly engaged, an accurate recommendation system is needed to suggest songs that cater to users' interests. But how can we personalize recommendations for millions of users with varying music tastes?

Upon discovering Spotify's public datasets for song tracks' audio features and corresponding user interactions, we decided to examine the predictability of a user's behavior, mainly song skipability, based on the given track and session features. We will utilize RNN to predict a song's skipability – either being skipped early on, or enjoyed till the end. These predictions will help recommend the right songs and keep users streaming for longer hours on the app.

How will you learn the background?

To understand user behavior, publications on recommendation systems and user interaction will be studied to understand the approaches used in this domain. For instance, there are viewing sessions as a sequential classification problem where user behavior on the current track is predicted based on the previous user session interaction. Some methods for this approach include decision trees, boosting, Recurrent Neural Networks etc.

What kinds of data will you use?

The dataset used for this project is structured and split into two tables: track features and session features.

Variable	Type	Scale
session_id	unique string ID	n/a
session_position	discrete ordinal	{1,20}
session_length	discrete ordinal	{10,20}
track_id_clean	unique string ID	n/a
skip_1	binary	{0,1}
skip_2	binary	{0,1}
skip_3	binary	{0,1}
not_skipped	binary	{0,1}
context_switch	binary	{0,1}
no_pause_before_play	binary	{0,1}
short_pause_before_play	binary	{0,1}

long_pause_before_play	binary	{0,1}
hist_user_behavior_n_seekfwd	discrete ordinal	{0,60}
hist_user_behavior_n_seekback	discrete ordinal	{0,151}
hist_user_behavior_is_shuffle	ordinal	{0,1}
hour_of_day	discrete ordinal	{0,23}
date	ordinal	n/a
premium	binary	{0,1}
context_type	categorical	n/a
hist_user_behavior_reason_start	categorical	n/a
hist_user_behavior_reason_end	categorical	n/a

* Categorical data features will be converted to numerical data via one-hot encoding. Features that are currently booleans will be converted into binary values.

Variable	**Type**	**Scale**
track_id	unique string ID	n/a
duration	continuous ordinal	{30.013330, 1787.760986}
release_year	continuous oridinal	{1950, 2018}
us_popularity_estimate	continuous ordinal	{90.018900, 100}
acousticness	continuous ordinal	{0, 1}
beat_strength	continuous ordinal	{0, 1}
bounciness	continuous ordinal	{0, 1}
danceability	continuous ordinal	{0, 1}
dyn_range_mean	continuous ordinal	{0, 32.342781}
energy	continuous ordinal	{0, 1}
flatness	continuous ordinal	{0, 1.103213}
instrumentalness	continuous ordinal	{0, 1}
key	nominal	{0,11}

liveness	continuous ordinal	{0, 1}
loudness	continuous ordinal	{-60, 1.634}
mechanism	continuous ordinal	{0,1}
mode	categorical	n/a
organism	continuous ordinal	{0, 1}
speechiness	continuous ordinal	{0, 1}
tempo	continuous ordinal	{0,218.774994}
time_signature	nominal	{0,5}
valence	continuous ordinal	{0, 1}
acoustic_vector_0	continuous ordinal	{-1.122792, 0.932165}
acoustic_vector_1	continuous ordinal	{-1.084360, 0.812837}
acoustic_vector_2	continuous ordinal	{-0.752231, 0.605288}
acoustic_vector_3	continuous ordinal	{-0.809136, 1.074504}
acoustic_vector_4	continuous ordinal	{-1.029858, 0.895769}
acoustic_vector_5	continuous ordinal	{-0.942461, 0.380279}
acoustic_vector_6	continuous ordinal	{-0.736395, 0.96670}
acoustic_vector_7	continuous ordinal	{-0.975647, 1.152213}

* Mode can be converted into binary data using one-hot encoding. Ordinal numerical features such as tempo will be normalized due to its relatively huge scale.

What kind of model will you build?

The data didn't need much change as it was clean. There was no missing data or duplicates. There were also dummy variables created due to the presence of categorical variables. Data reduction methods such as pairwise correlation and decision tree feature importance were used for feature selection and were performed to reduce the dimensionality of the dataset.

The target variable will be defined as a three class skipability outcome variable:

	skip _1	**skip_2**	**skip_3**	**not_skipped**	**Indicated interaction**	**skipped (custom target var)**
1	False	False	False	False	not played at all	**1 True**

2	False	False	False	True	played until the end	**0 False**
3	False	False	True	False		**0 False**
4	False	True	True	False	skipped before 1/3rd of the song is played	**1 True**
5	True	True	True	False		**1 True**

To implement the prediction of user skipability, the dataset was trained and explored on a series of classification models with different regularization techniques, feature transformations as needed, different hyperparameter values, and normalization methods. These models include Decision trees, multinomial logistic regression, support vector machines (SVM), neural networks (NN), and recurrent neural networks (RNN). AUC and accuracy scores were used as performance metrics for all models.

For the logistic regression, SVM, and NN models, each row indicates a track that was played by a user during a Spotify listening session. While these models are simpler, RNN will be used as a deep learning method as it incorporates temporal information. In this scenario, user behavior is dependent on when they skip. Specifically, a SimpleRNN was used along with dense layers. The dataset will be restructured for RNN so that each row indicates an entire listening session, the data is restructured in such a way that one row in the input would contain track features of the entire session (20 tracks). The output would be the skipped variable for all 20 tracks. Since some sessions do not have 20 tracks, they are padded with zeros.

What assumptions are safe to make?

The following assumptions will be made:

- Different sessions record the interactions of different users.
- Our target variable will be created by amalgamating 4 original dataset features into one target outcome variable. Variable unions will not lead to significant information loss.
- Since Spotify's premium membership includes an unlimited skips feature, premium users may skip more nonchalantly, while non-premium users are more cautious with their scarce amount of skips. The importance of this feature will be examined in determining skipability.
- The genre of the track is a confounding variable, since it is not an original track feature variable and is an intuitive predictor of user skipability
- The type of user session is a confounding variable; the user could be playing their own customized playlist, an album, listening to someone else's playlist, or listening to a Spotify's generated recommended songs playlist. The nature of the session is not described in the dataset, but could be a big factor in determining how often a user skips.

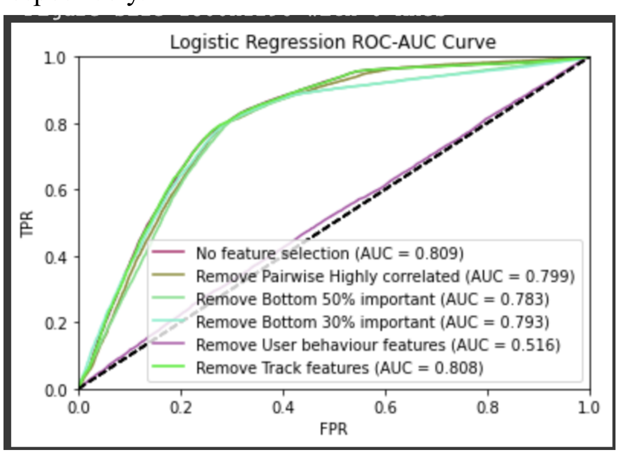
Results

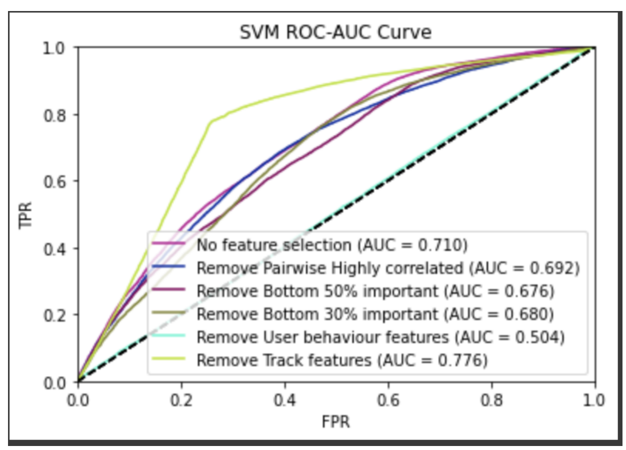
One of the main research questions was to determine the feature importance of the data in determining skipability. Because of this, a total of six subsets of data (with different selections of variables) were created: one with all the features from the original data from Spotify's API, one with highly pairwise correlated variables removed from the dataset, one with the top 50% of the important features determined through decision tree, one with the top 70% of the important featured determined through decision tree, one with just the user behavior related features, and one with just track related features.

K-fold cross validation was performed to tune the hyperparameters for logistic regression and SVM. For logistic regression, a C value of 0.1 performed best for regularization. For SVM, a regularization C value of 0.01 and a linear kernel performed accuracy. For RNN, the setting of random seed seems to be important in how the model gets trained. These hyperparameters were then used to train the models with each of the subsets of data. Additionally, session and track identification values were removed for the non temporal models as they did not indicate a difference in the test accuracy or AUC values. The performance of each model (test accuracy) can be shown in the figure below.

	CNN	Logistic Regression	SVM	RNN
No feature selection	75.911367	75.991780	64.668215	0.873275
Remove Pairwise Highly correlated	76.149631	75.488444	64.864784	0.855175
Remove Bottom 50% important	74.723017	75.762449	62.115797	0.830875
Remove Bottom 30% important	76.075172	75.899452	63.870026	0.832800
Remove User behaviour features	53.517395	53.469740	50.515249	0.836975
Remove Track features	76.414698	76.033476	75.759471	0.888500

The figure below shows the AUC - ROC curve for logistic regression and SMV of all six datasets respectively.





Conclusion and Next Steps

RNN was the highest performing model, with a prediction test accuracy of above 80% using only user interaction features. This suggested the importance of the sequential and temporal relations between the data variables. Additionally, all subsets of the data performed poorly when only track features were considered, suggesting that track feature are less significant in playing a part in predicting skipability. In contrast, the dataset of only user interaction features performed the best for all models, suggesting further research on immediate user interaction within one listening session. Future stages for this project include potential clustering of tracks to create pseudo genres to investigate the importance of genre in user interaction behavior, as well as the possible correlation between similar tracks in a session and how that could predict skipability.

References
[1] Hurtado, A., Wagner, M., & Mundada, S. (2019). Thank you, Next: Using NLP Techniques to Predict Song Skips on Spotify based on Sequential User and Acoustic Data.

[2] Jeunen, O., & Goethals, B. (2019). Predicting sequential user behaviour with session-based recurrent neural networks: our approach to the 2019 WSDM Cup Sequential Skip Prediction Challenge. In Proceedings of the 2019 WSDM Cup Workshop, February 15th 2019, Melbourne, Australia (pp. 1-4).

[3] Ferraro, A., Bogdanov, D., & Serra, X. (2019). Skip prediction using boosting trees based on acoustic features of tracks in sessions. arXiv preprint arXiv:1903.11833.

[4]https://machinelearningmastery.com/clustering-algorithms-with-python/

[5]https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/

Team 9 Point Division
Claudia Shao - 4
Varshitha Chennamsetti - 4
Winnie Zheng - 4