Experimental Project Report
Machine Learning Course a.y. '22-23
Done by: Arina Lopukhina
July 22, 2023

Ridge Regression Modelling: Spotify Song Popularity Prediction

1. Introduction

The goal of this project was to use Python to recreate ridge regression, which was later employed to predict how numerical and categorical features of a song may (or may not) have an impact on its popularity on Spotify. This is an exciting topic to explore since defining a way to predict a song's success not only can have direct financial gains but also shed some light on the tastes of millions of Spotify listeners.

As per assignment, the model was first trained using only numerical features like duration, tempo, valence, danceability, and other characteristics of the sound, as well as applying 5-fold cross-validation (CV) after. Then, encoded categorical features were added to the model and then the model was cross validated as well. Additionally, an alternative cross-validation approach was applied with grid search CV to experiment with parameter tuning. Also, a few additional models were created with different sets of features in order to see if there are any patterns regarding feature importance.

2. Theoretical approach

The model class uses the closed form solution of ridge regression which can be described as follows:

$$\widehat{\beta} = (X'X + \lambda I)^{-1}X'Y,$$

where X is the matrix containing observations, Y is the target vector, I is an identity matrix, and λ is the penalty term ranging from $[0, +\infty)$.

At the beginning, it was also attempted to write the class using gradient descent, however, the data required a high number of iterations at a small learning rate, and the code would take up to ten minutes to fit the model and calculate the relevant metrics, so for efficiency purposes the closed form was preferred.

In order to evaluate how suitable the model parameters are and later compare models among each other, several metrics have been introduced:

- Root Mean Squared Error: the loss function ought to be minimized to select the best regularization parameter.
- R-squared: to account for how well the observations explain the target variable.

Finally, to check for the best regularization coefficient, two forms of cross-validation were applied (using Scikit learn library):

- 5-fold
- Grid search

3. Data preprocessing

The original dataset consists of 20 variables which range from song descriptions such as name, artist, genre to more technical features such as tempo, key, time signature, and so on. From the theoretical point of view, the following ones were defined as categorical (before encoding):

- *artists*
- *album_name*
- *track_name*
- *explicit*
- *track_genre*
- *key*
- *mode*

Although the last two, *key* and *mode*, were present already in a numerical format, those were merely dummy variables for subcategories (types of keys and True/False for mode), and hence counted with the rest of the categorical variables. Nonetheless, explicit, key, and mode did not follow the same encoding procedures as the rest of the defined categorical variables since they did not have a myriad of subcategories. The remaining 13 features (*'popularity','duration_ms','danceability','energy','loudness','speechiness','acousticness','instrumentalness','liveness','valence','tempo','time_signature'*) were treated as numerical, with *popularity* being identified as the target variable.

The dataset was then checked for missing values, which was not the case, and the distribution of each numerical variable was examined. Most of the variables did not resemble Gaussian distribution and seemed to have required further normalization to be then rescaled with zero mean and 1 standard deviation. Therefore, a different technique has been chosen, MinMax rescaling to the range [-1,1] as there were some negative values present.

In terms of categorical variables, they were more tedious to deal with. Initially, only target encoding was applied to the four categorical variables (artists, albums, track name, genres) and then they were translated to the same range as the numerical variables. However, even with smoothing, the model seemed to derive the information about the target from the encoded variables. This was spotted in a correlation matrix, where encoded features were extremely significant with respect to popularity are artist name (0.7), album name (0.8), song track name (0.7), as well as genre (0.5), and their correlation coefficients rose almost up to 1 once duplicated track id's were erased. Also, the models with such encoded data showed at se overfittied test and train curves which were nearly identical.

Therefore, the encoding method had to be balanced-out by introducing other types of data encoding. Also, it had to be accounted for over 20 thousand duplicated track id's because there was only one genre per row, however thousands of songs belonged to more than one genre. So, after splitting the data into test and train sets, data encoding was performed on both separately.

After discovering that genres were evenly present, they were encoded with One Hot Encoding (OHE) based on their corresponding popularity range (<26 – 'low', 26<x<41– 'medium', 41< - 'high') in order to get the accurate distribution between the three dummy categories.

Low popularity	alt-rock, black-metal, bluegrass, classical, detroit-techno, disco, drum-and-bass, dub, electronic, funk, jazz, latin, malay, metalcore, minimal-techno, mpb, opera, reggae, ska, soul, tango
Medium popularity	acoustic, alt-rock, alternative, blues, brazil, cantopop, chill, comedy, country, dance, dancehall, death-metal, deep-house, disco, edm, electro, emo, folk, forro, french, garage, german, gospel, goth, groove, guitar, happy, hardcore, hardstyle, heavy-metal, hip-hop, honky-tonk, house, indie, indie-pop, industrial, j-dance, j-idol, j-pop, j-rock, kids, latino, mandopop, metal, minimal-techno, new-age, party, power-pop, psych-rock, punk, punk-rock, r-n-b, rockabilly, salsa, samba, ska, sleep, songwriter, spanish, study, swedish, techno, trance, trip-hop, turkish,
High popularity	anime, french, grunge, indian, k-pop, mandopop, pagode, piano, pop, pop-film, progressive-house, reggaeton, rock-n-roll, sertanejo, singer-songwriter, synth-pop, world-music

After performing OHE, the duplicate records were erased so that track id would be unique from then on.

Then, artist and album names were subjected to frequency encoding as the initial logic behind was that if an artist or an album have a lot of appearances, they are supposedly more popular. This assumption was quickly defeated after learning that the artist with most recordings was George Jones and not The Weeknd, or Taylor Swift, or any other famous pop artist of the 21st century. Thus, frequency can rather be interpreted as whether the length of one's career matters for popularity.

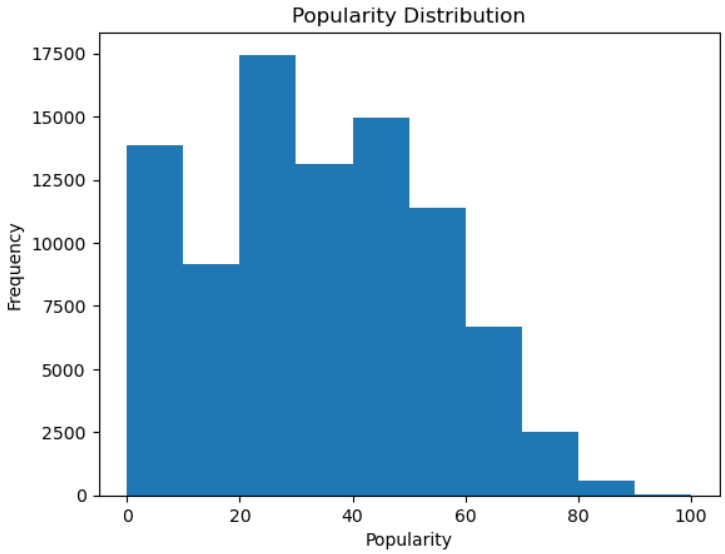
Alternatively, another metric was created for artist popularity. After splitting artist column (since sometimes there were two strings per cell) and counting unique instances of them, mean popularity of their songs was calculated and split into three categories ('low', 'medium', 'high') based on the quartiles. Consequently, this variable has been OHE-ed and added to the dataset.

Lastly, track names were encoded using smoothed target encoding as breaking them into OHE categories like genre would likely make them more prone to causing overfitting and frequency encoding does not make much sense in terms of interpretation.

4. Data: at first glance

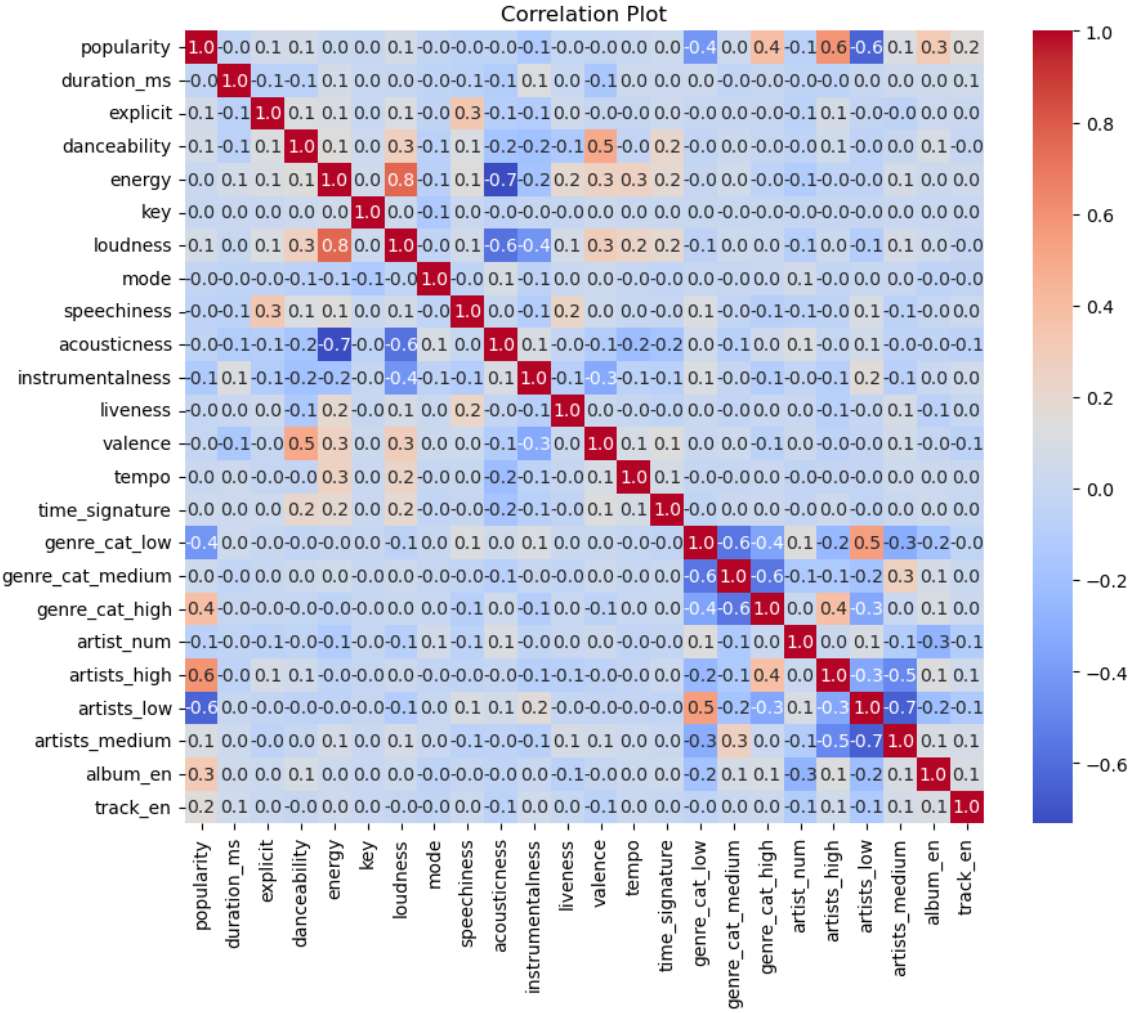
As many other numeric variables, *popularity* was not normally distributed, with most songs falling withing the lower popularity scores. Notably, most duplicate tracks were found in lowest popularity group (0-10), perhaps, because classifying a song into multiple genres can help ensure higher exposure to different groups of listeners.

Figure 1:Popularity distribution by number of tracks



After the encoding and re-scaling adjustments, this was the initial correlation between the variables:

Figure 2: Pearson Correlation matrix - all features



As can be seen on the matrix, the numerical variables all have little to no correlation with the target variable *popularity*. However, as expected, genres belonging to high popularity category possess higher correlation, and, surprisingly, so do certain albums. However, artist or song name frequency seem to have little to no contribution as well.

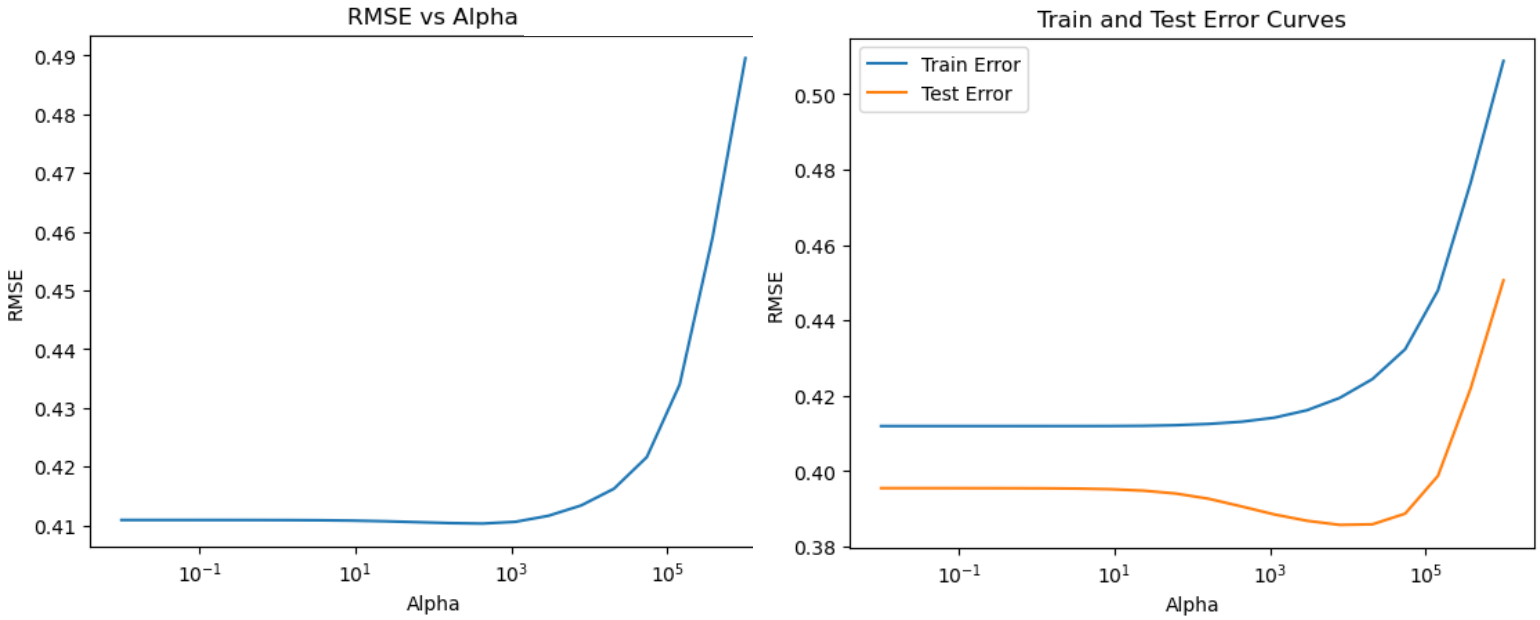
5.1. Model 1: Numerical variables only

As previously mentioned, not all columns in numerical format have been attributed to numerical features as some of them represented Boolean True/False values or previously encoded categories (*key, mode*). It was initially presumed that the model with only numerical variables would perform poorly since, as seen earlier, the correlation coefficients between them and the target were extremely low. Nonetheless, Pearson correlation suggested a minimal presence of association between some of these variables, so excluding all of them completely would be incorrect.

Without cross-validation, the model has identified an incredibly large regularization coefficient of approximately 10826 for the range of alphas [10^{-2}, 10^{5}], meaning that the model applied a very high penalty to the variables and, in fact, this resulted in zeroing out of the variable coefficients (see Figure). Adjusting the ranges of alpha did not quite change the situation as with smaller ranges the 'best alpha' targeted was the highest available value, and increasing the range led to alpha continuing to rise.

At the same time, the obtained best RMSE was quite high, 0.39, and most importantly, the R-squared was negative, which implies that the selected variables have no necessary information to identify the target variable. Such behavior was demonstrated both in the train and test sets (see Figure 3).

Figure 3: Model 1, Cross-validated

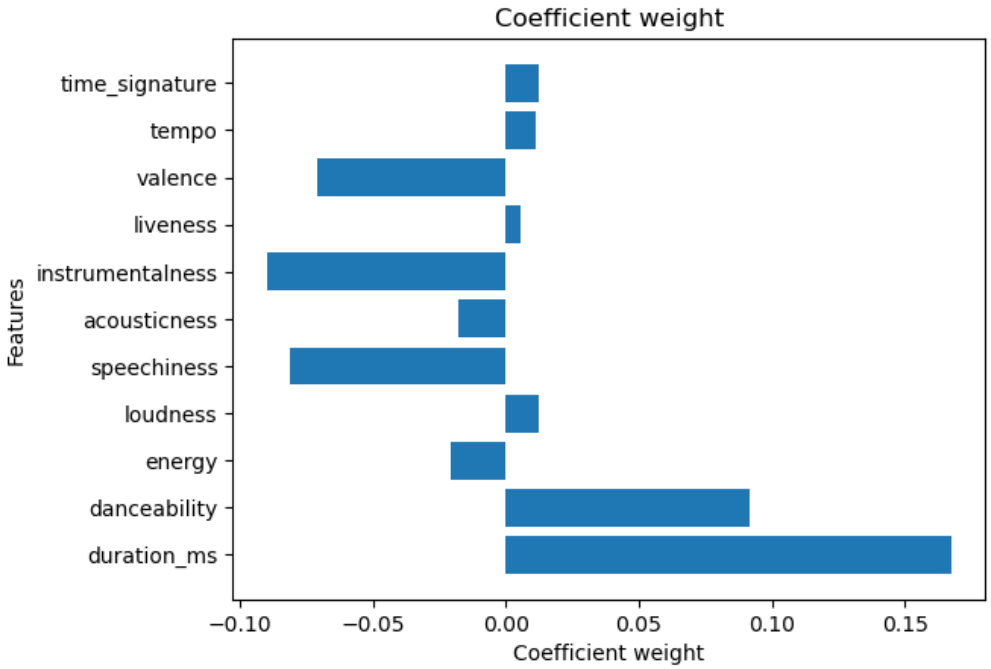


Using 5-fold cross-validation has produced very similar results, apart from the fact that the model chose a smaller value of the best alpha (approx. 428) since it was averaged between the five folds. Nonetheless, the RMSE remained as high as 0.41 and the R-squared stayed

negative, which supports the assumption that the numerical features have very little to do with song popularity.

Although ridge regression is biased when it comes to feature importance (compared to linear regression), looking at the variable coefficients produced by the model leads to the assumption that the song duration slightly contributes to the song popularity (longer songs are associated with the higher the popularity). Also, danceability might slightly add to a song's popularity. As for the remaining features, the model has zeroed out *time signature, tempo, liveness, loudness*, and *energy*, implying their lack of influence on the target. Interestingly, valence, instrumentalness and speechiness have a negative effect on popularity, implying that people prefer songs that contain a moderate portion of vocals. Ironically, higher valence (associated with more positive-sounding songs) also does not contribute to increasing popularity.

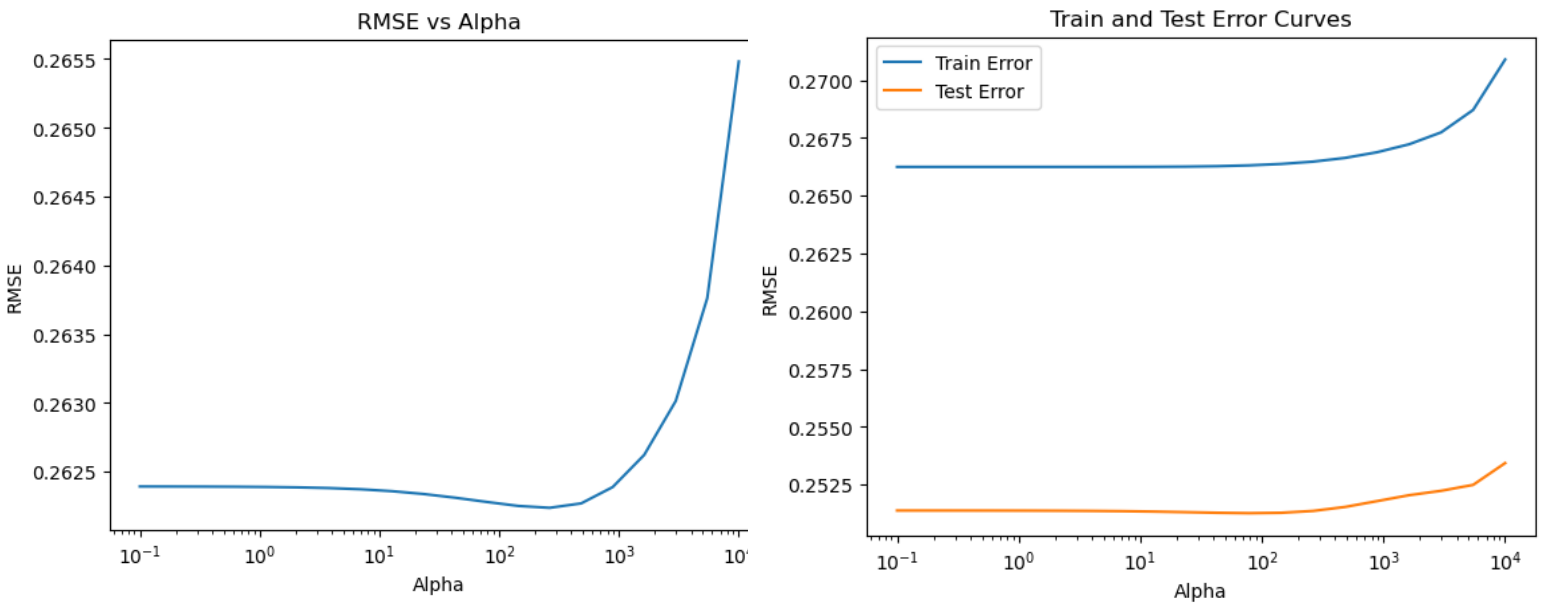
Figure 4: Weights by feature of Model 1



5.2.Model 2: All variables

Introducing categorical features to the model has largely improved the results since there were more relevant factors to the target variable. Before applying cross-validation, the model's RMSE has gone down to 0.26, and the R-squared has risen to 0.55, with the best alpha being roughly 78.

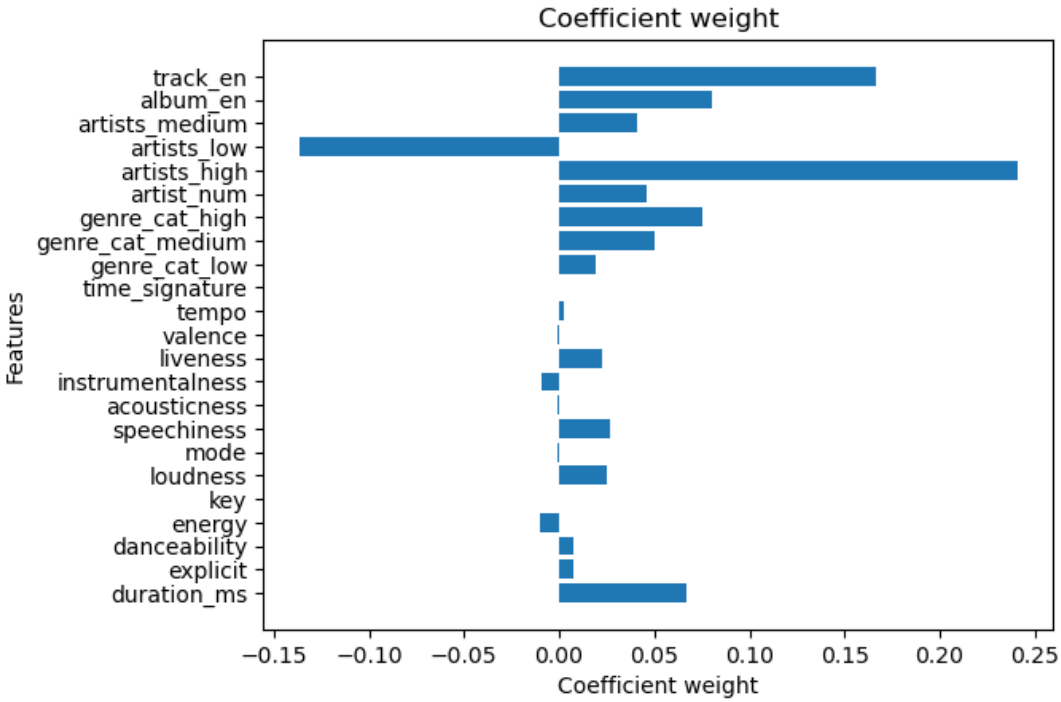
Figure 5: Model 2, cross-validated



After applying cross-validation, the model's RMSE and R-squared remained unchanged, however, the value of alpha has risen to 263, implying the need for a higher penalty largely due to the presence of numerical variables. Changing the range of alpha did not result in an improvement of the model behavior. Notably, the model has performed better on test set, hence it was not overfitted.

In terms of the variable weights, as suspected, categorical features have higher coefficients in the model. Namely, artists with higher mean popularity and track name stand out, but also genre's popularity and album's frequent appearance in the ranking positively contributed to the song's popularity. However, they must be reviewed with caution since genre and artist categories were derived according to the mean popularity of unique entries, hence there is a subtle dependence between them and the target. As for the numerical song qualities, all but duration were almost completely zeroed out.

Figure 6: Model 2 feature weights

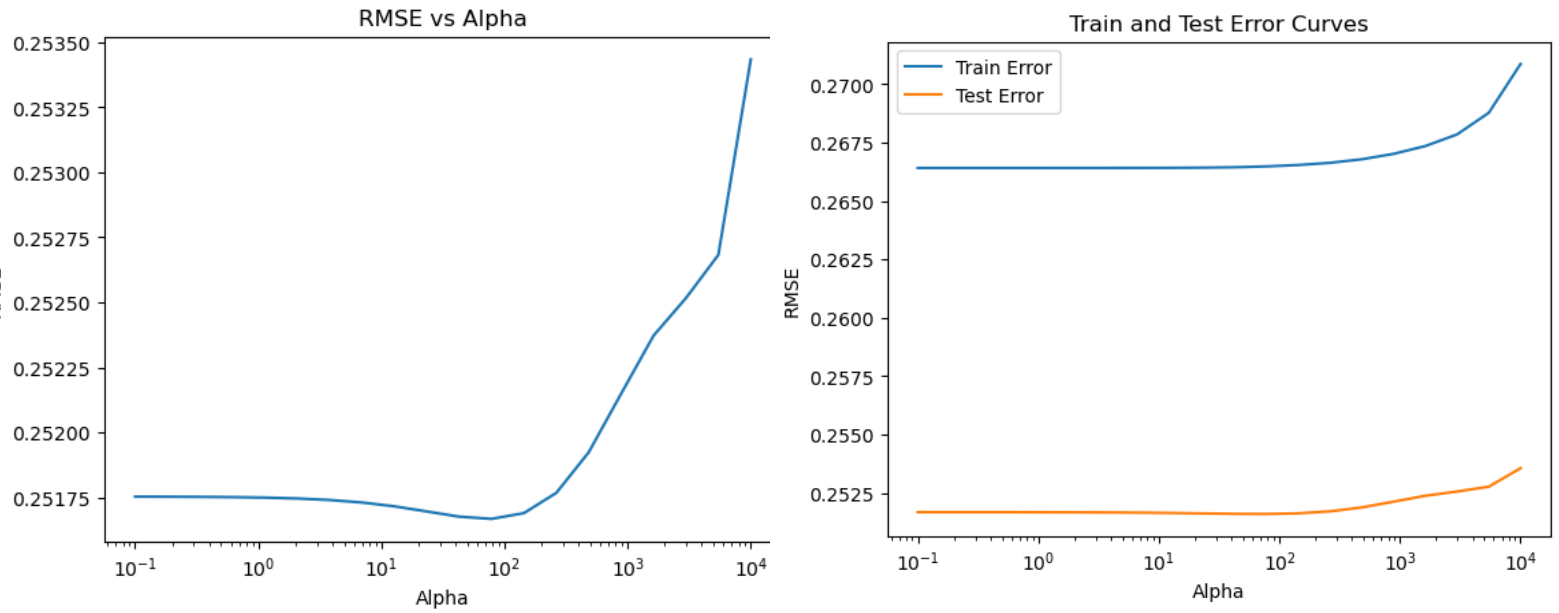


5.3. Model 3: Excluding loosely relevant numerical features

Although ridge regression produces biased weights, an alternative version of the model which excluded most of the numerical variables was attempted to see whether the R-squared and RMSE could be further improved. After analyzing the results of the previous models, only *duration, loudness, and speechiness* were kept in the model, as well as names of tracks and albums and genre and artist categories.

After applying cross-validation on Model 3, it was noted that the results barely differed from the ones of cross-validated Model 2. The R-squared and the alpha penalty have not changed, while RMSE decreased rather insignificantly, from 0.26 to 0.25. As expected, the weights of the remaining numerical values remained fairly close to 0.

Figure 7: Model 3, cross-validated



5.4. Model 4: Only categorical features

Leaving only the categorical features produced by encoding procedures has drastically changed the model behavior. With no numerical ones left, the model's best alpha was identified as 0.01, which is the lowest possible value in the range. Expanding the range of alpha to include lower values resulted in the model choosing the least value, which suggests that the penalty term ought to be discarded. Hence, Model 4 essentially began to function as a simple linear regression model. As for the R-squared and RMSE, they slightly improved (0.56 and 0.25 respectively) but remained on roughly the same level as models 2 and 3 (see Figure 8). Lastly, the behavior of test and train curves did not change as well as the overall model behavior when it came to choosing the best alpha.

Figure 8: Model 4, cross-validated

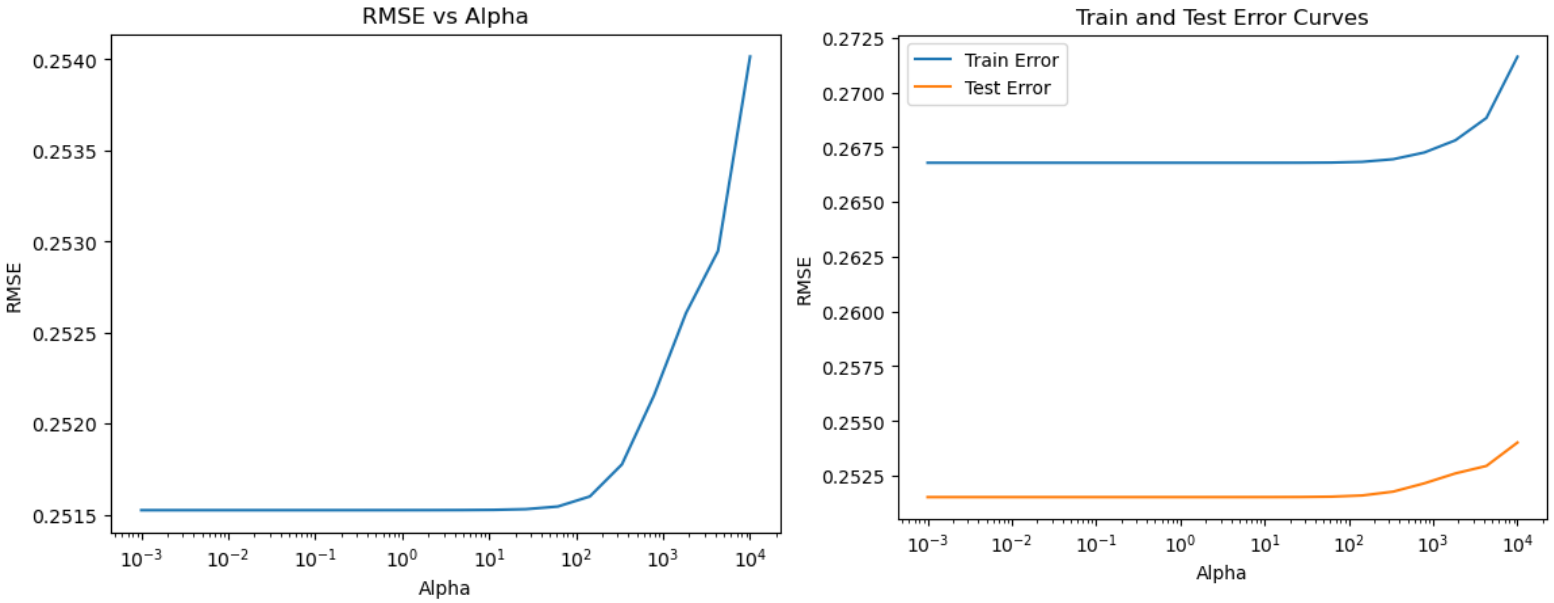
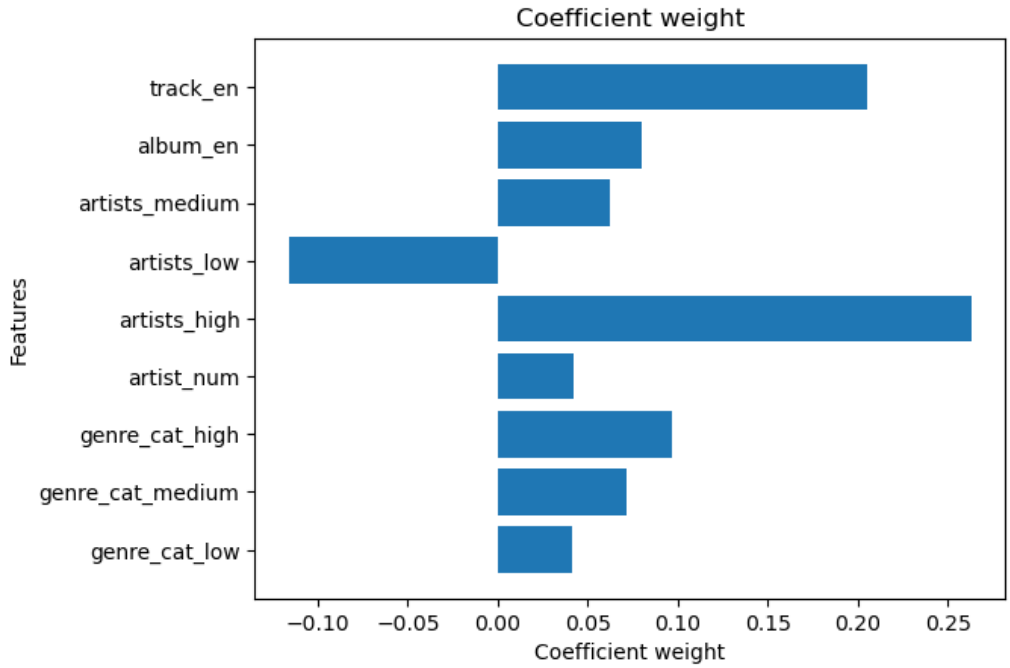


Figure 9: Model 4 feature weights

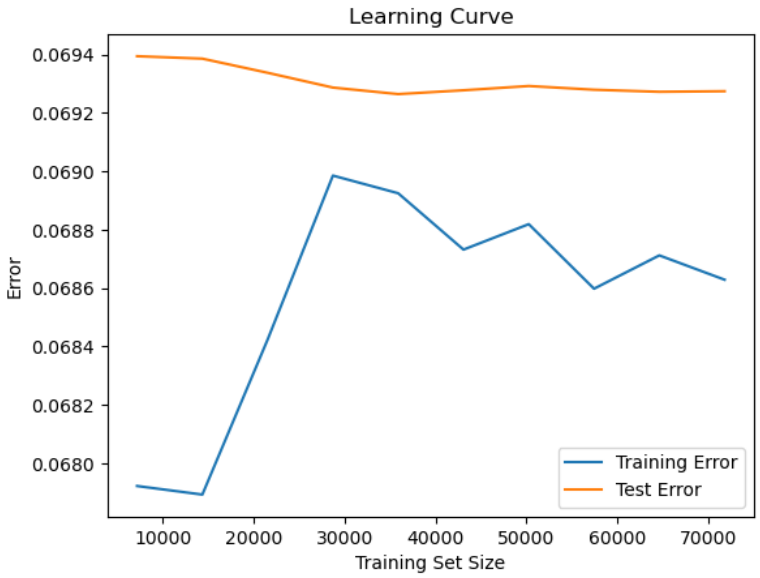


5.5. Using Grid search Cross validation

To experiment further, an alternative cross validation method was explored. Grid search is considered to be better at finding the best hyperparameters than simply using k-fold cross-validation, so it could help assess the previous parameter choices. Overall, using all the features, the grid search identified an alpha of 293, which is slightly larger than in model 2 (it was 263) and produced an outcome with the same RMSE and a slightly better R-square of 0.59 (used to be 0.56).

Consequently, it was applied to only categorical variables where the model achieved the same RMSE and R-squared but with a much smaller alpha of 16. This is an interesting outcome as although alpha is small, the grid search model insists on penalizing also the encoded categorical variables opposed to the model with k-fold cross validation that nearly took a form of linear regression. However, it is worth noting that in both cases grid search model has produced a slightly overfitted outcome, although the difference between train and test error is only 0,001. Also, the model did not seem to need a set size of bigger than 30,000 observations.

Figure 10: Model 4, grid search cross-validated



6. Conclusion

To summarize, in terms of model performance, the numerical-variables-only model did not yield satisfactory results due to the features' loose (or non-existent) correlation with the target. So, no matter the model tuning, it would always be associated with a very large alpha and could not arrive anywhere near decent ranges of R-squared or RMSE. On the other hand, categorical features were responsible for explaining the track's popularity and hence produced better model results. With less numerical features, in both k-fold and grid search cross validation, the value of alpha decreased significantly, thus indicating that such features indeed were moderately responsible for a song's popularity.

Overall, it has been a long and tedious journey, but in the end, we obtained better insights of the Spotify song popularity and how various features impact it. Firstly, we seem to have reached a rather sad conclusion – people do not seem to bother about the sound qualities but rather pay attention to features like artists', albums', or song names. Hence a good PR campaign could be much more responsible for a song's popularity than actual song quality. Duration of a song and qualities like *danceability* and *speechiness* were also slightly contributive to the song's popularity, however, not nearly as influential as the above-mentioned ones.

However, these results need to be viewed along with the data limitations such as the encoding techniques and variables interdependence. With so many unique categories, it was rather hard to find a method that would reflect the variable's relation to popularity without creating blunt linear dependence which could produce biased results. Also, duplicated records made the encoding process more challenging as well as corrupted the initial model tests.

I/We declare that this material, which I/We now submit for assessment, is entirely my/our own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my/our work. I/We understand that plagiarism, collusion, and copying are grave and serious offences in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion or copying. This assignment, or any part of it, has not been previously submitted by me/us or any other person for assessment on this or any other course of study.

Citations:

Cesa-Bianchi, N. (2023, March 21). Hyperparameter tuning and risk estimates. https://cesa-bianchi.di.unimi.it/MSA/Notes/crossVal.pdf

MaharshiPandya. (2022, October 22). *Spotify tracks dataset*. Kaggle. https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset

Vprokopev. (2018, October 7). *Mean (likelihood) encodings: A comprehensive study*. Kaggle. https://www.kaggle.com/code/vprokopev/mean-likelihood-encodings-a-comprehensive-study/notebook