Factor Analysis for Predicting Song Release Year

Michael Kronovet, Tanishq Kancharla

1 Introduction

For this project, we explored the capabilities of musical note patterns and song lyrics to predict the year a song was released. For our dataset we used a 10,000 song subset of the Million Song Dataset as well as the Musixmatch dataset for the corresponding 10,000 songs. The 10,000 song subset is a high-dimensional data set, with many features for each song. In contrast, the Musixmatch dataset only contains bag-of-words lyrics for each song. Our goal is to determine whether models predicting a song's release year trained on the Musixmatch dataset can exhibit similar testing performance as models trained on the full feature-set of the 10k subset.

2 Background/Literature

An important part of making our models robust is choosing the features our models will train on carefully. One of the papers that helped us decide is "Feature Selection for High Dimensional Data: A Fast Correlation-Based Filter Solution" by L. Yu and H. Liu. This paper describes assigning a measure of goodness to each feature to maximize relevance to the class concept and minimize redudance to every other relevant feature. To quantize these measures, they introduce a new term, symmetrical uncertainty (SU), based on information gain between two random variables. Exploring various methods of feature selection will help streamline our model.

The paper "Lyric Text Mining in Music Mood Classification" by X. Hu, J. Downie, and A. Ehmann compares and contrasts music mood classification models trained on audio features vs. lyric text features. Their support vector machine model gave us valuable information that we could work into our own models. For example, they performed their own cleaning on the bag-of-words data to make sure each song had associated words that were representative of the song. How to do this in our case can be explored in several ways: for example, we can naively remove all occurrences of "meaningless" words like "the", "of", etc.

3 Methods/Model

The 10,000 song subset of the Million Song Dataset stored the data for each song in an HDF5 file which we scripted through to convert to a pandas dataframe. We removed every song without its year recorded, dropped features that were unrelated to the musical qualities of the songs, removed confidence values for song features, and took the weighted average of the 12 dimensional timbre and pitch vectors for each point in the song. The transformed pitch values ended up being highly correlated with each of the corresponding transformed timbre values, but both seemed provide useful information so we retained them. In addition, when a song did not have information for a certain variable, we input the mean of that feature so that we didn't throw out too much data.

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

3.1 Preliminary Results

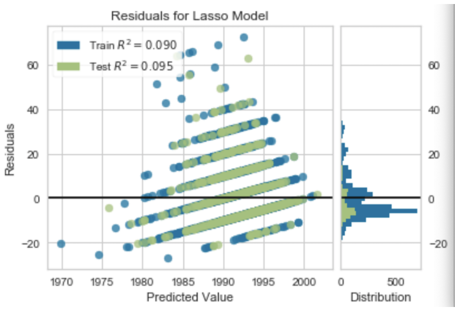
We tested both regression and classification models on the data, where the regression models attempted to classify the exact year a song was released while the classifiers were built to predict the decade a song was released.

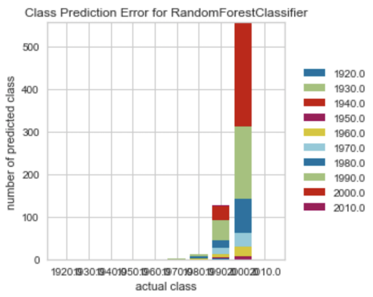
Model	Classifier Accuracy on Test Set	Regression Test MSE
KNN Classifier	0.510684	
Logistic Regression	0.530983	
RBF Kernel Classifier	0.541667	
Linear Kernel Classifier	0.354701	
Gaussian Naive Bayes	0.231838	
Random Forest Classifier	0.557692	
KNN Regression		134.405
Linear Ridge Regression		125.113
Linear Lasso Regression		122.999
SVM Regression		180.668

The linear lasso model performed the best of regression, and the random forest classifier performed the best of our classification models, so we will further investigate these two models.

3.2 Evaluation of Preliminary Work

We proceed to make diagnostic plots for each of the best models:





The residuals for the lasso model are mainly centered at 0, although some values seem to be skewed where the predicted years were way under the actual release year. After further investigation it seems that the lasso model predicts about every release year between 1970 and 2005. This could be due to the database containing mostly songs from this period, but this is definitely not good for our model.

The random forest classifier doesn't seem to do that well either. It's apparent that the random forest model predict mostly decades between 1980 and 2000. As a result, the classifier only does slightly better than if you were to naively guess that every song was released in 2000.

When using mean squared error to compare the lasso and random forest model on the same scale (where predicting the decade 1930 for a song released in 1940 is equivalent to being 10 years off for a regression prediction), we get the following results.

Model	Classifier Accuracy on Test Set	Regression Test MSE
Linear Lasso Regression		122.999
Random Forest classifier	0.557692	149.024
Always Predicting 2000	0.540598	192.628

It seems as though regression may suit this problem more since the proportion of songs in the validation set with correctly classified song release decades is very low, even for the best classifier. In addition, when comparing the mean squared error of the classifiers and the regression model, it seems as though the classifier was not doing considerably better despite having an easier problem to solve with only 10 decades to choose from.

4 Future Work

For our baseline models we performed a very naive form of feature extraction and selection, where we manually decided if a feature would be relevant to a model's performance or not. For example, features like "artist_familiarity", "artist_id", "artist_location" are very loosely related to a song's release year, and so we decided to remove them. One problem with this approach that we completely disregard non-numeric features such as "title", "similar_artists", "artist_name" and "artist_terms" which are theoretically highly correlated with a song's release year. A sophisticated model could develop a "cache" of artist names that are correlated with song release years/decades and that would highly increase accuracy. Similarly, it could create a cache for terms that are associated with artists and use the similar artists list to compare with its cache and see if all the similar artists are clustered around a particular time, narrowing down its choices. These ideas, however, only apply if our goal is to create a model that uses all dimensions as effectively as possible rather than solely by features that are directly contrived from the musical qualities of the song.

Another way we could improve our feature extraction is to use the correlation-based feature extraction mentioned in the background section of this paper. This will help us focus on key features that maximize correlation with the class but also minimize redudance with each other. One particular example where this would come in handy is with the pitch and timbre vectors, which were highly correlated. Therefore it may make sense to remove one of the timbre or pitch vectors, and to decide, we have to perform the aforementioned symmetric uncertainty filtration.

One of the problems with our data seems to be its concentration in the 2000-decade. This is seen most clearly in our random forest classifier model, which, despite being the best classifier, only does marginally better than a naive model which predicts the 2000 decade every time. To prevent this, additional analysis must be done to see which features tend to separate the 2000 decade from other decades.

4.1 Timeline

Moving forward, we expect to start work on the bag-of-words problem and modify our baseline models. Michael will explore some naive models trained on the bag-of-words dataset and Tanishq will start work on modifying current models to employ some of the techniques learned from reasearch and preliminary analysis by the end of spring break.

Around the beginning of April, we hope to have narrowed our models for each problem to one or two, and proceed to work on optimizing those. After our final models are decided for each problem, we will work on analysis of the results.

5 Teammates and work division

Michael Kronovet did much of the data processing and feature selection. He also fit the various predictive models and created the evaluation plots for the best classifier and best regressor models.

Tanishq Kancharla did much of the research of similar problems and helped select features for baseline models. Incorporating research articles and baseline model data, he decided the direction of improvement for models in the final report.

5.1 References and Citations

1. Hu, Xiao, J. Stephen Downie, and Andreas F. Ehmann. "Lyric text mining in music mood classification." American music 183.5,049 (2009): 2-209.
2. Martineau, Justin, AND Finin, Tim. "Delta TFIDF: An Improved Feature Space for Sentiment Analysis" International AAAI Conference on Web and Social Media (2009): n. pag. Web. 11 Mar. 2019
3. Yu, Lei, and Huan Liu. "Feature selection for high-dimensional data: A fast correlation-based filter solution." Proceedings of the 20th international conference on machine learning (ICML-03). 2003.
4. Singhi, Abhishek, and Daniel G. Brown. "Hit song detection using lyric features alone." Proceedings of International Society for Music Information Retrieval (2014).