The dataset

Our data mainly stems from the Million Song Dataset (hereafter: 1M), which is a freely available dataset covering roughly one million songs in contemporary music. The dataset records various information of the actual audio, like length, tempo and loudness, as well as metadata like release year, genre tags and artist information for each song. The basis of our project was to use these fields to construct a feature space in which the recommender could perform some kind of search.

In total, 1M provides over 50 fields for each song, most of which are not useful for our purposes. So in a first step, we selected -- by reasonable intuition -- those fields that might be predictive or informative for a recommender. // TODO first fields chosen // TODO some fields empty, cannot choose // TODO some more fields were left out to keep the scope contained, but they might actually be good

// TODO mention EchoNest dataset // TODO mention thisismyjam dataset (only briefly, because we didn't end up using it)

What didn't work

Empirical analysis did not show any results because the data is too sparse. // TODO Why exactly? How could fix? Maybe recommender bad?

// TODO using thisismyjam for empirical analysis didnt work, too sparse, also most IDs didnt match

Possible improvements

There are many ways to improve the recommendation algorithm. While our current approach is apt for finding songs with similar characteristics, given input songs that are all similar themselves, its limitations quickly show for users with varied taste. If a user likes both metal and chorales, it should not be assumed that they also like whatever the average of those two styles is. However, by choosing one of each as their input songs as their input songs, the song with least distance in the feature space to will be exactly that and thus not pose a good recommendation. To avoid this, we could cluster the input songs based on some maximum distance and then give separate recommendations for each cluster. On the other hand, we could embrace this nonlinearity with which taste maps onto the feature space and wholesale replace the nearest-neighbor recommender with e.g. a neural network. It could be trained on the input songs' characteristics, with gold labels acquired through a dataset like Thisismyjam or Echonest likes. Given how sparsely populated these datasets are in the whole 1M Song feature space, this would be a sizeable task requiring a lot of compute.