# Do You Want to Dance?

## 1. Size of Team

There are four (4) members in our team.

## 2. Motivation

Determining the danceability of a song has an interesting application to music recommendation, with well-known applications such as Pandora and Songza. We want to apply machine learning techniques to song meta-data to embark on our own introduction to the field of music recommendation.

## 3. Problem Statement

A tangible way of approaching music recommendation is understanding song features such as energy, danceability, and valence. Due to time constraints, we will focus on predicting the danceability of songs. Our goal is to answer the following questions:

1. What features (defined in the General Apporach) are most relevant to predicting the danceability of a song?
2. What is a representative similarity function that will reduce the impact of noise in our dataset?
3. What feature space mapping can we use to reduce the dimensionality of our input space for a quicker kNN invocation?
4. What is the best value of $k$ to use in this kNN approximation?

## 4. General Approach

We are using the Echo Nest API to populate a classless, multidimensional instance space with features like 'danceability', 'valence', 'loudness', 'energy', and 'tempo'. From this instance space, a k-Nearest-Neighbors (kNN) approach will be used to determine the danceability of the song. Further, this kNN model will be used to determine which features in the instance space are most relevant to the 'danceability' feature.

### 4.1. Adapting Feature Mapping for kNN

The primary focus of our "learning" will be to find a function that will map our instances into a reliable feature space that provides accurate prediction of the danceability feature. An additional objective of this function will be to reduce the feature space dimensionality, or to reduce the computational demand, as the kNN algorithm running time is heavily dependent on the feature dimensionality.

Thus, a major portion of this project will involve the research and development of an effective feature space mapping for our song feature data that minimizes dimensionality and maximizes accuracy.

In an effort to figure out the best value of $k$ for kNN, we will find results for $k$ values in the range [1..10], and choose the value of $k$ whose median neighbor yields the value closest to the true feature. The true feature value exists for the predicted instance (from the Echo Nest API), and thus will be used in comparison with the predicted feature. The overall algorithm and mapping function will be validated and classified by the amount of deviation the predicted value has from the value within the original data set.

#### 4.1.1. Instance Features

The majority of these features are floats that fall in the range [0..1]. For those features with floats in a different range, we will scale that range to fit in [0..1] for consistency.

Although we will be utilizing more features (such as lyrical content), we have thus far found that the following 6 features are the strongest indicators of danceability:

- Danceability: a combination of energy, rhythm, and tempo approximated using algorithmic estimation by Echo Nest (values in range: [0.0..1.0])
- Valence: Measure of the emotional content of a song (values in range: [0.0..1.0])
- Acousticness: Measure of how acoustic vs. electric a song is (values in range: [0.0..1.0])
- Energy: Energy from listener point of view (values in range: [0.0..1.0])

- Loudness: overall loudness in dB (values in range: [-100.0..100.0])

- Tempo: estimated tempo in beats-per-minute (BPM) (values in range: [0.0..500.0])

### 4.2. Final Product and Possible Extensions

At the conclusion of this project, we will have a system that accepts a song and predicts how danceable it is using only the meta-data provided by the Echo Nest API. As mentioned in the problem statement, this introduces the possibility of detecting song and listener mood, and subsequent music adaptation (playlist prediction).

Additionally, if time permits, we would also like to answer the following questions:

1. The Echo Nest developers have reported that valence, energy, and danceability are closely related, but have not published their exact relationship. We would like to examine the tradeoffs between valence, energy, and danceability in an effort to better understand a possible Echo Nest algorithm for determining each of these features.

2. What effect does leaving out one of the features have on our results? To answer this, we will use some variation of Leave-One-Out and do feature-wise exclusions instead of instance-wise exclusions, and compare the accuracy of our LOO results to those with the entire feature set. This research will demonstrate the importance these features have on the danceability of a song.

## 5. Resources

We will primarily be using the Echo Nest API for data, Matlab / Octave / R for data processing and visualization, and PHP and other web tools for creating an interface for our final product

## 6. Progress

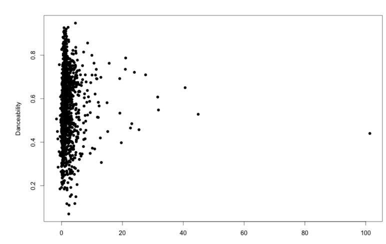
### 6.1. Progress and Feasibility

Noticeably, the project focus has changed once more. Only upon obtaining data from the Echo Nest API was it realized that there was no reliable access to enough features like sound files or chord progressions that we feel would be helpful in distinguishing the Beatles' songs from other artists. So from the data we have already obtained, we rethought what was within reach, and decided to determine the 'danceability' of a song instead, which we definitely see as more feasible than our original Beatles-related projects.

Within our new project, we have written code to submit GET requests to the Echo Nest API for song data, parse the json responses, and format the results into .csv files for easy access in subsequent training and use.

From the data we've extracted, we've begun investigating how its large input space could be converted into a higher dimensional space that would be linearly representable. This is what we have come up with so far, which we know is not perfect, because it maps too many instances to a very small range of x-values, but is a good starting point for now:

$$
\begin{aligned}
&2.218902 energy^2 + 1.389 valence \\
&-1.038269 energy^2 * liveness \\
&-0.01705197 energy^2 * tempo \\
&+247.3779 * speechiness^2 * liveness \\
&+2.447499 liveness * acousticness + \\
&4.269856 liveness * instrumentalness + \\
&0.01783731 tempo * instrumentalness \\
&+2.839705 energy^2 * instrumentalness * valence \\
&-207.259 speechiness^2 * liveness * instrumentalness \\
&-177.1677 * speechiness^2 * valence
\end{aligned}
$$




We have also started looking into incorporating song lyrics using Bayesian analysis. We might use Bayesian analysis to get a float value that represents lyrical mood, which we input into our vector in order to consider how some words suggest a song is more appropriate for dancing than others.

### 6.2. Problems

We clearly saw that a lack of data was a major problem in our first project, and in our current project potential problems still include dealing with the amount of time it takes to query data from the API and we are still working through how to include more mathematical analysis such as computed error bounds, that we need to more concretely plan

## 7. Schedule

| Date | Milestone |
|---|---|
| **11/11** | **Progress Report Due** |
| 11/13 | Solidify understanding of most relevant features |
| 11/15 | Research/training of music similarity functions |
| 11/15 | Finish implementation of kNN (Python) |
| 11/23 | Finish Testing various feature space mappings |
| 11/25 | Fine tune kNN algoirthm (determine $k$) |
| 11/28 | Evaluation of Findings |
| 12/02 | Finish Poster |
| **12/04** | **Poster Presentation** |
| 12/12 | Finish Final Report |
| **12/16** | **Final Project Report Due** |