

# Spotify Recommender



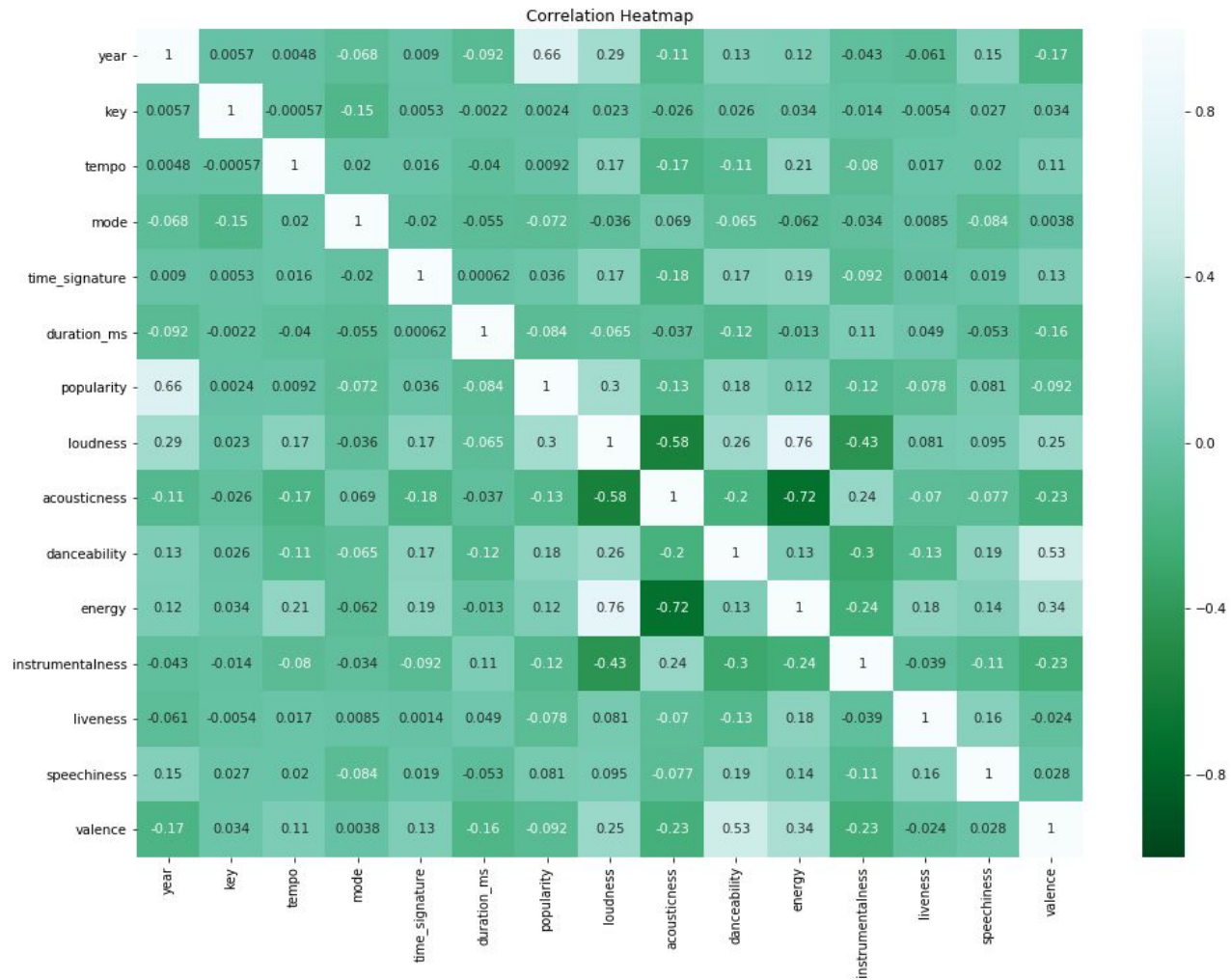
By: Brendan Lo

**258 million active users**

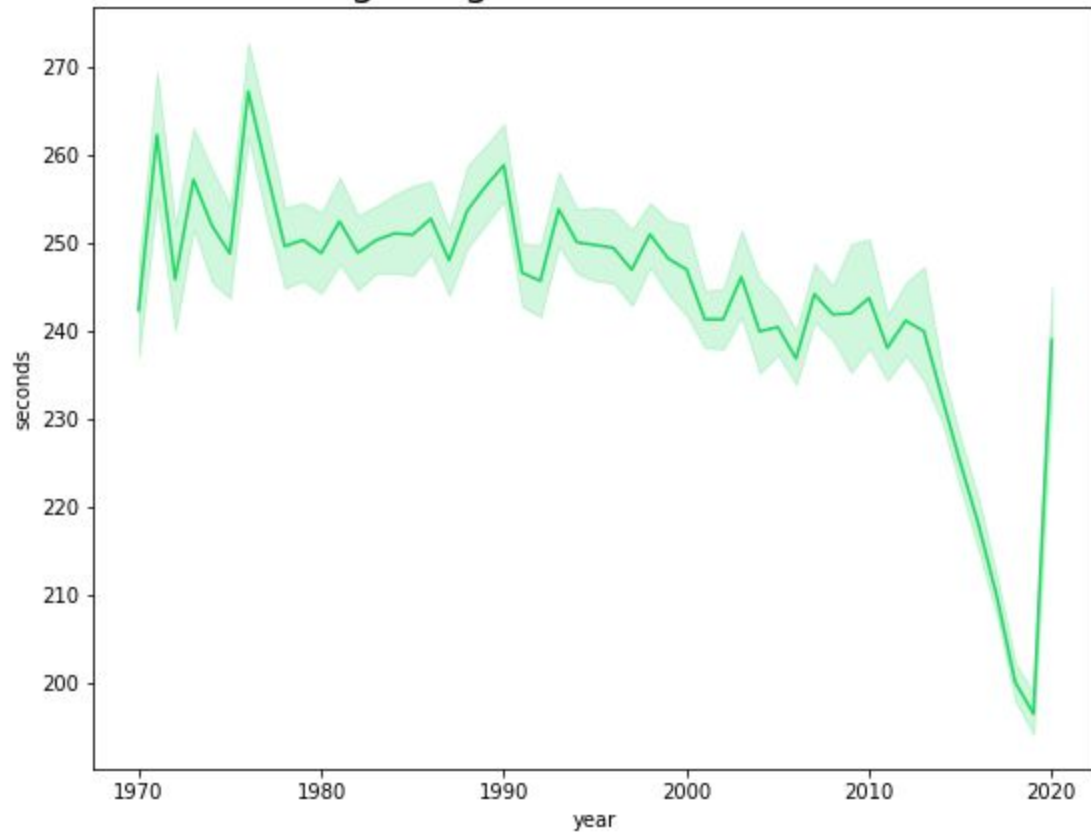
**113 million subscribers**

# Getting Data from Spotify's Web API

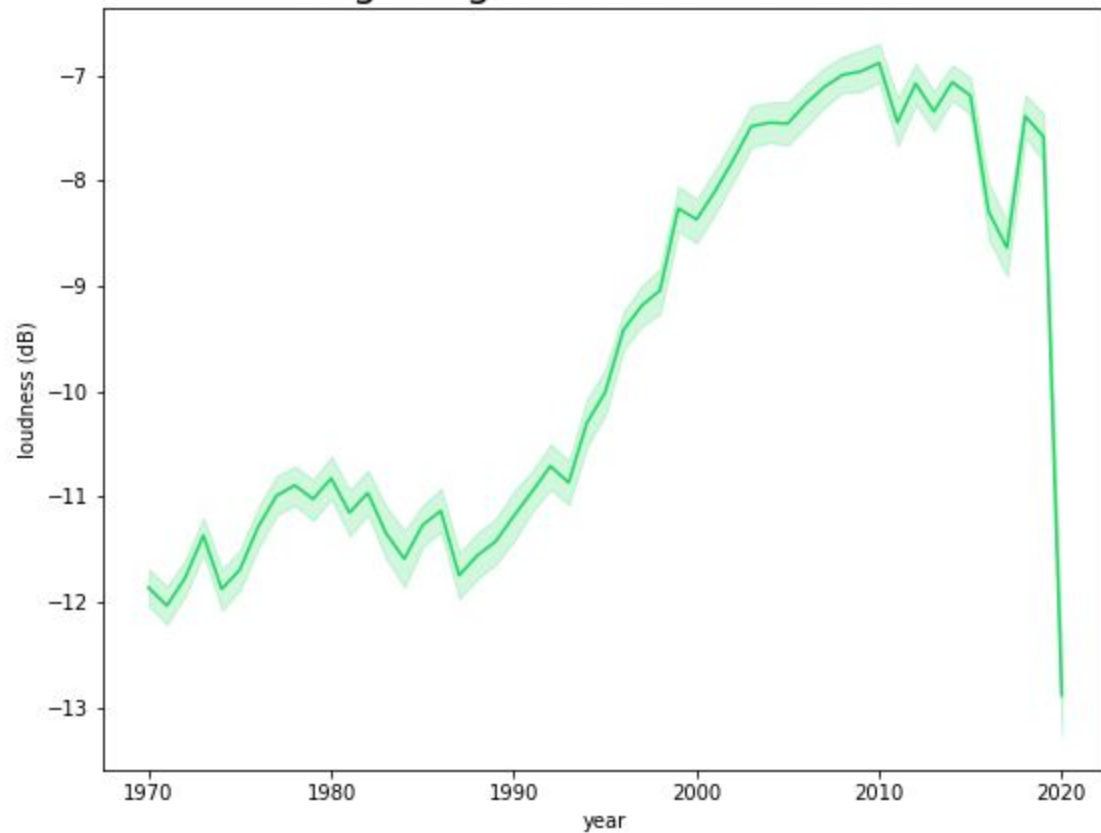
- Used Spotify's Web API - spotipy
- Scraped 2,000 tracks per year between the years 1970-2020, resulting in about 100,000 tracks.
- 17 features. Metrics were [predefined by Spotify](#) based on their internal algorithms.

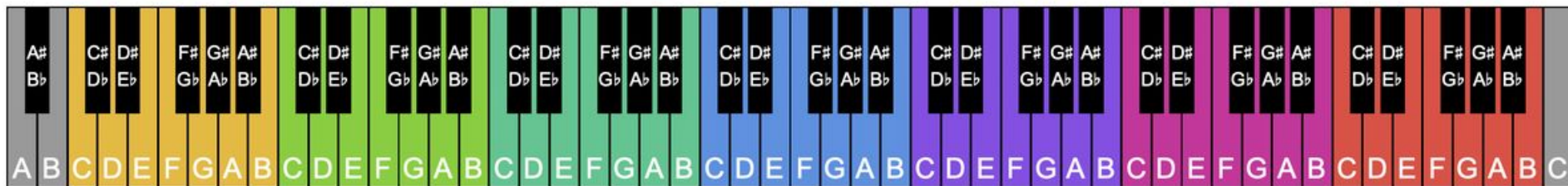
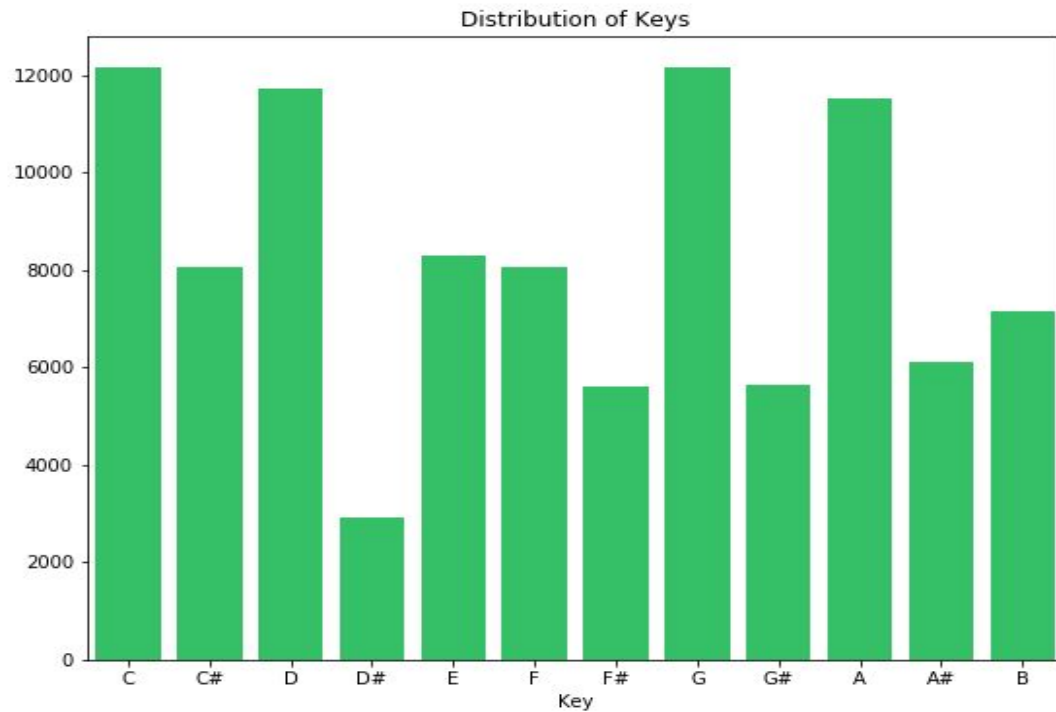


# Avg Song Duration over Time

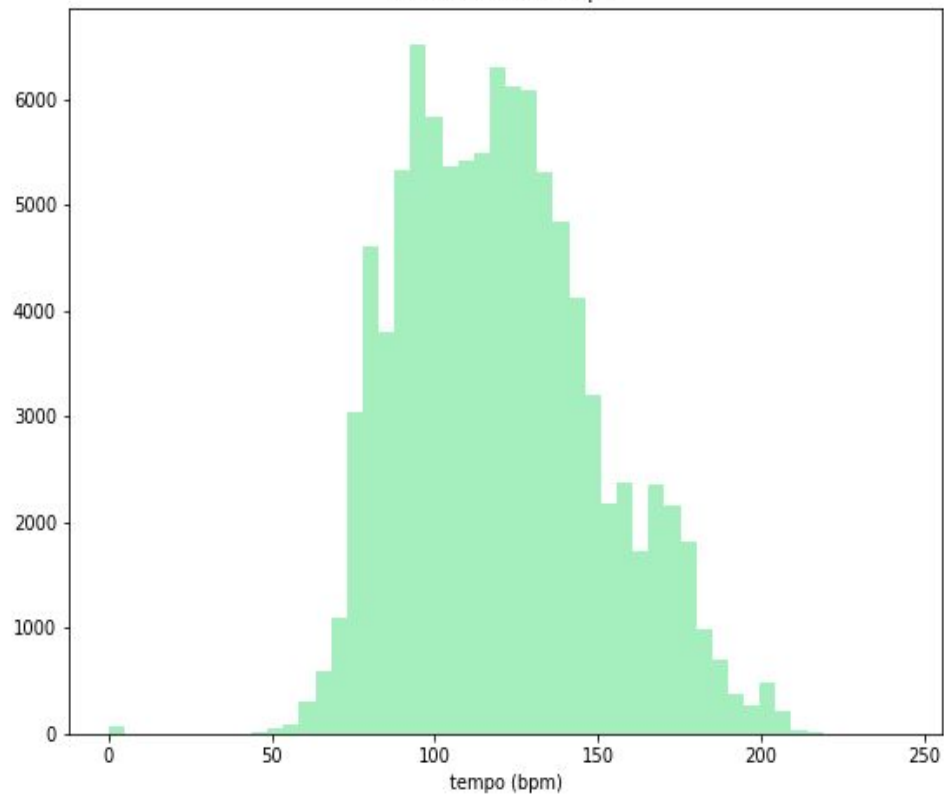


Avg Song Loudness over Time

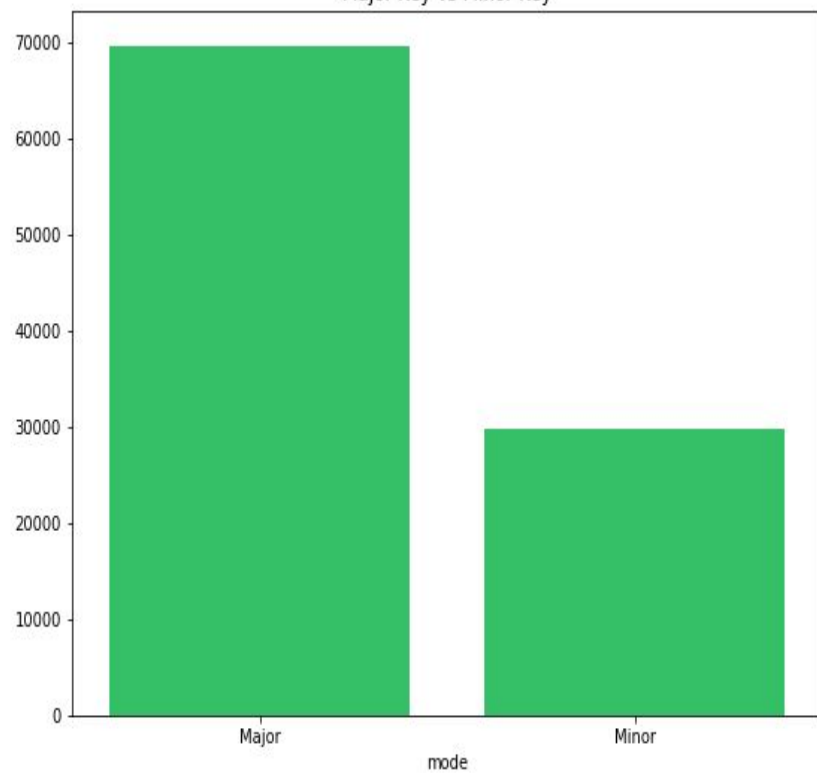




Distribution of Tempos



Major Key vs Minor Key



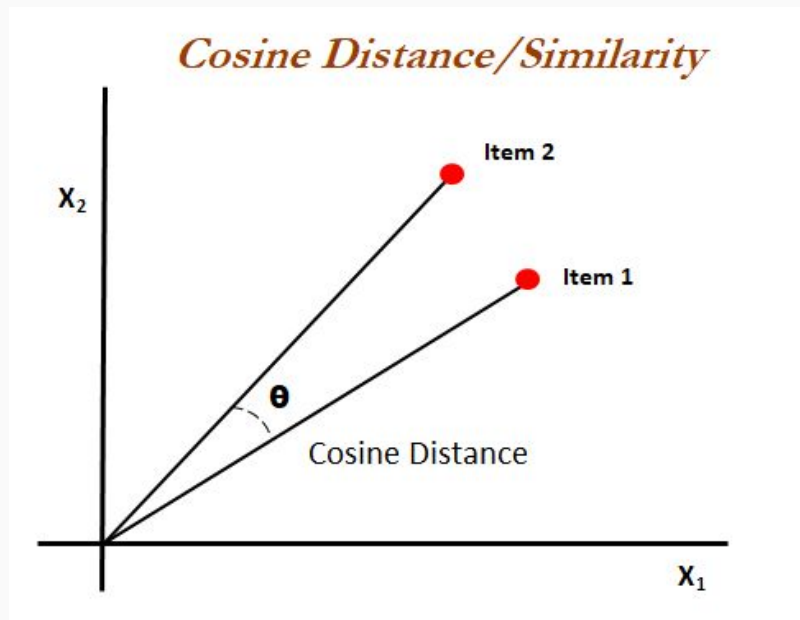


# Preprocessing Concepts

- Standard Scaling
- Cosine Similarity
- Singular Value Decomposition (SVD)

# Cosine Similarity

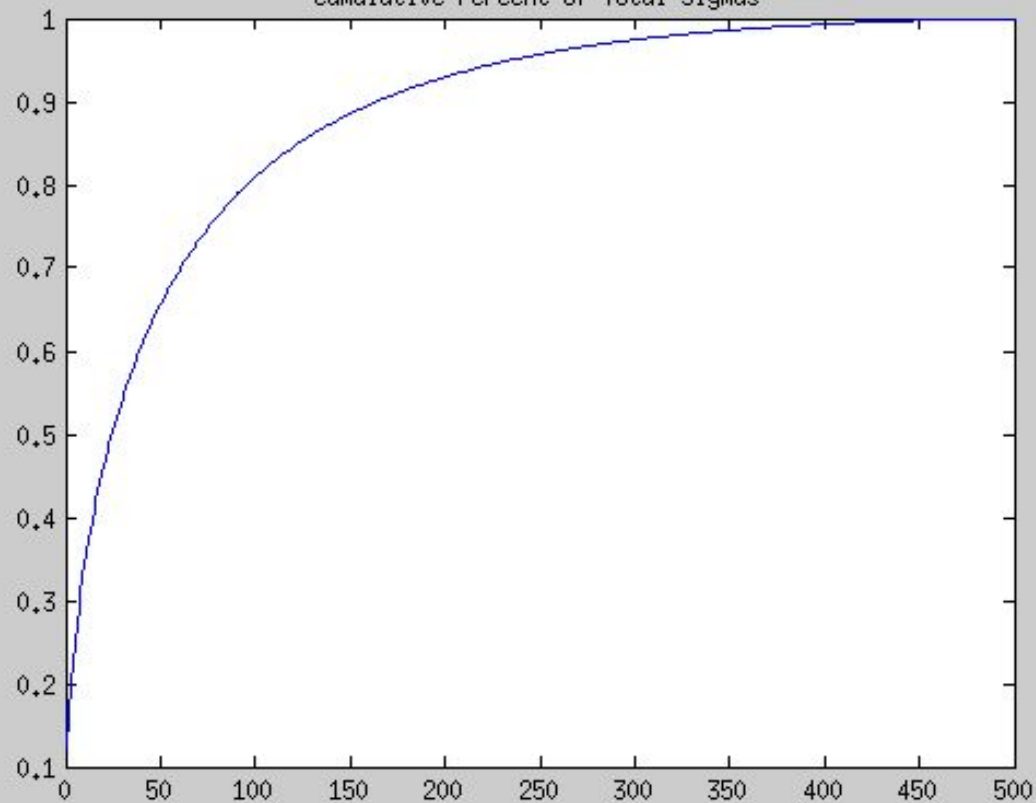
- Comparing the angle of vectors
- Lines overlap at 1
- Orthogonal lines are 0
- Lines in the opposite direction are -1
- In our case each song with its features define a single vector



# Singular Value Decomposition (SVD)

- Factorizes a matrix  $M$  of any size into the product of 3 simpler matrixes.
- Useful for dimensionality reduction.

Cumulative Percent of Total Sigmas



Full-Rank Tiger



Rank 200 Tiger



Rank 100 Tiger



Rank 50 Tiger



Rank 30 Tiger



Rank 20 Tiger



Rank 10 Tiger



Rank 3 Tiger



# Product-based Recommender System

```
[17]: recommend1('closer, the chainsmokers')
```

```
closer, the chainsmokers
```

```
10 closest songs
```

```
track_name
```

just the same, charlotte lawrence	0.966189
fuck, i'm lonely (with anne-marie) - from "13 reasons why: season 3", lauv	0.950468
mi verdad (feat. shakira), maná	0.948197
fallin' all in you, shawn mendes	0.937250
panama, quinn xcii	0.936350
tough (feat. noah kahan), quinn xcii	0.930817
no vuelvas más, darell	0.923703
despacio, natti natasha	0.921614
cool again, shoffy	0.915895
bloom, black party	0.912240

```
Name: closer, the chainsmokers, dtype: float64
```

```
*****
```

# Recommender Takeaways

- Recommender recommended songs based on the audio's features.
- This resulted in output songs that were musically similar, yet sound completely different
- Good for exploring new music OUTSIDE your comfort zone.

# Limits and Next Steps...

- Need more memory and computing power. There are a lot more than 2,000 songs per year.
- Understand the data better. When I query 2,000 tracks per year, how does Spotify decide which tracks of that year?
- Improve recommender by incorporating user ratings/feedback.
- Create an app that is user-friendly, like with Flask.

**Demo time!**

