



GENERATIVE ADVERSARIAL NETWORKS AND SPOTIFY DATA

Jack Koltermann

Problem Statement

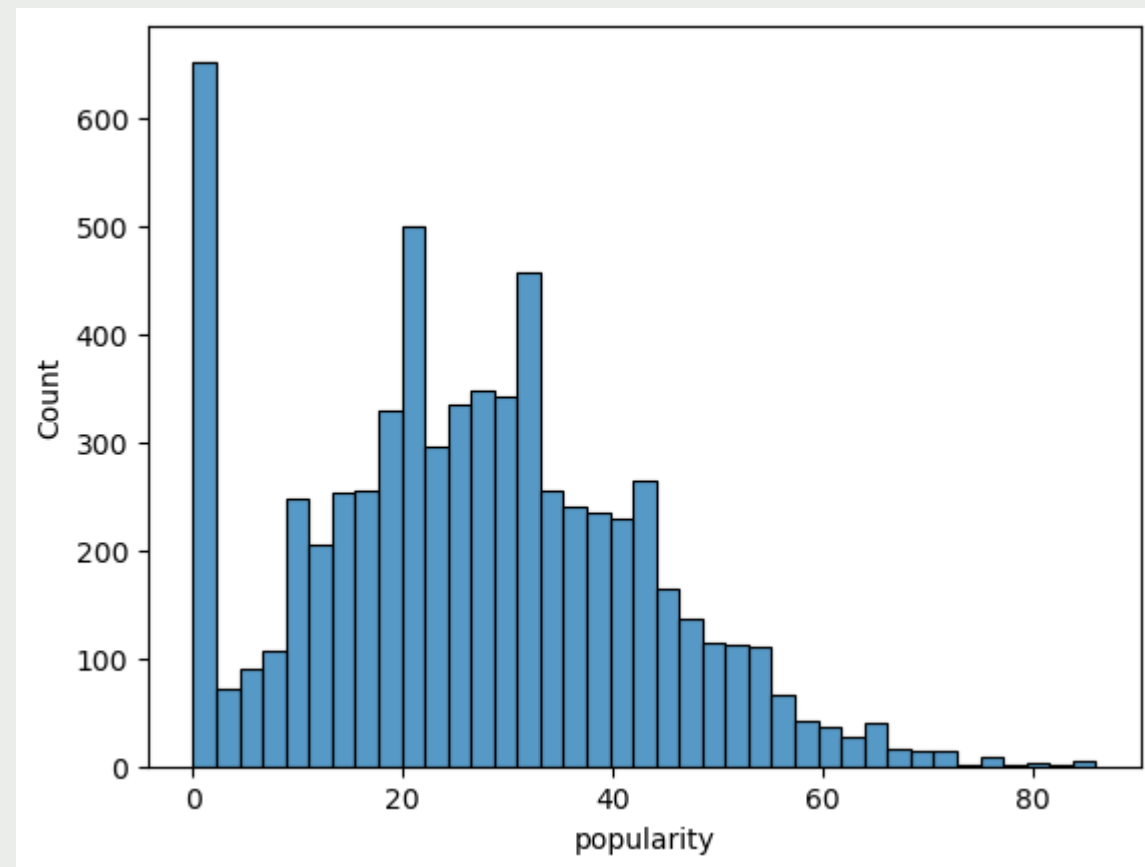
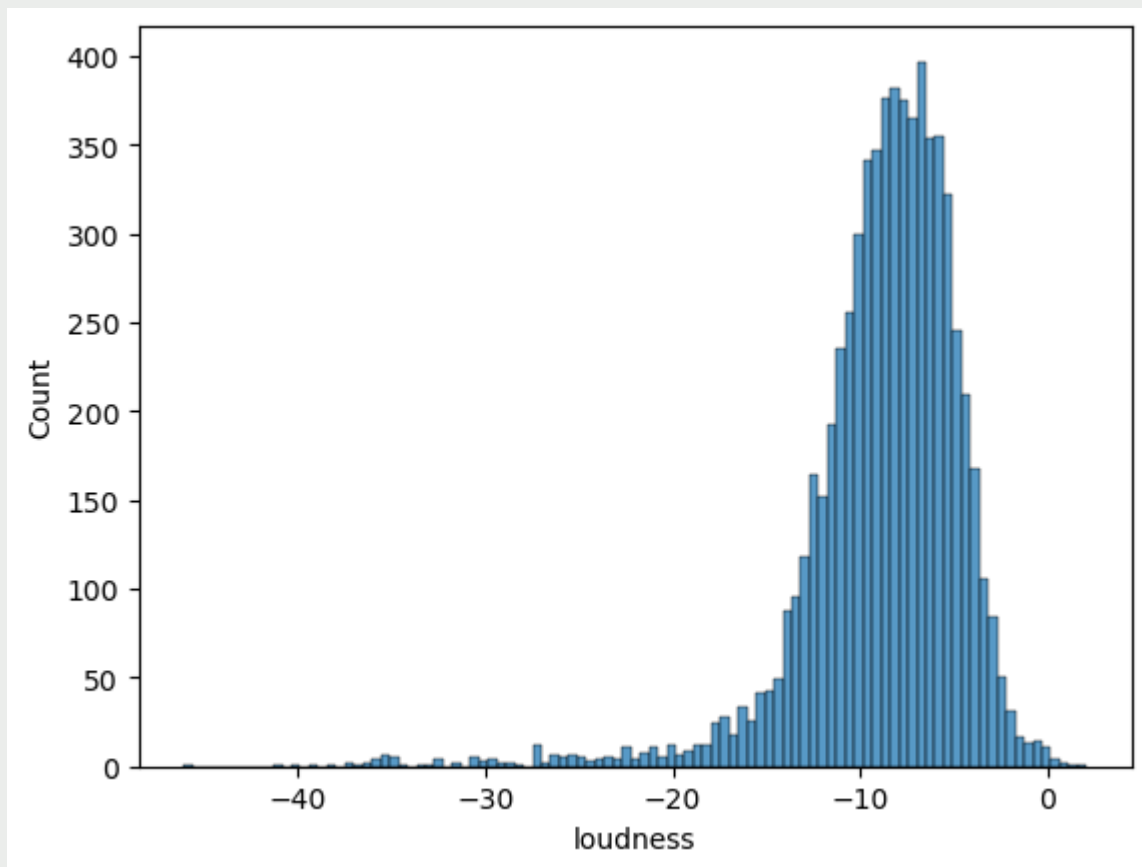
Problem Statement

- Music production has become democratized
- Artists struggle with infinite optionality
- Rick Rubin: Apply constraints to your music workflow
- People Dislike Constraints
- Artists want a subtle nudge
- Create a tool that provides that subtle nudge

Spotify API Platform

- Used with SpotiPy wrapper library
- Queried for ~948 house / artists
- Queried for 50 or all song ids from each artist available
- Pulled ~6,000 analysis files from popular songs using primary song keys
- Created a helper API handler due to lack of continuity of data structures received

EDA



Analysis Data

- Time-series and single-value feature set
- Related to rhythm and other independent characteristics

Feature Set

Feature set synthesized into the following:

```
['loudness',  
'beats_start_binary',  
'beats duration binary',  
'tatums start binary',  
'tatums duration binary',  
'bars start binary',  
'bars duration binary',  
'sections start binary']
```

- **Wanted to Produce this data through a GAN**

What is a GAN?

Generators

INPUT:

Noise Vector

Dense Layers (7)

Output:

Data Structure of Same
Dimensionality to
Training Set (8 x 3000)

Discriminators or 'critics'

INPUT:

Output of the generator

Convolutional layers

Dense and Max Pooling Layers

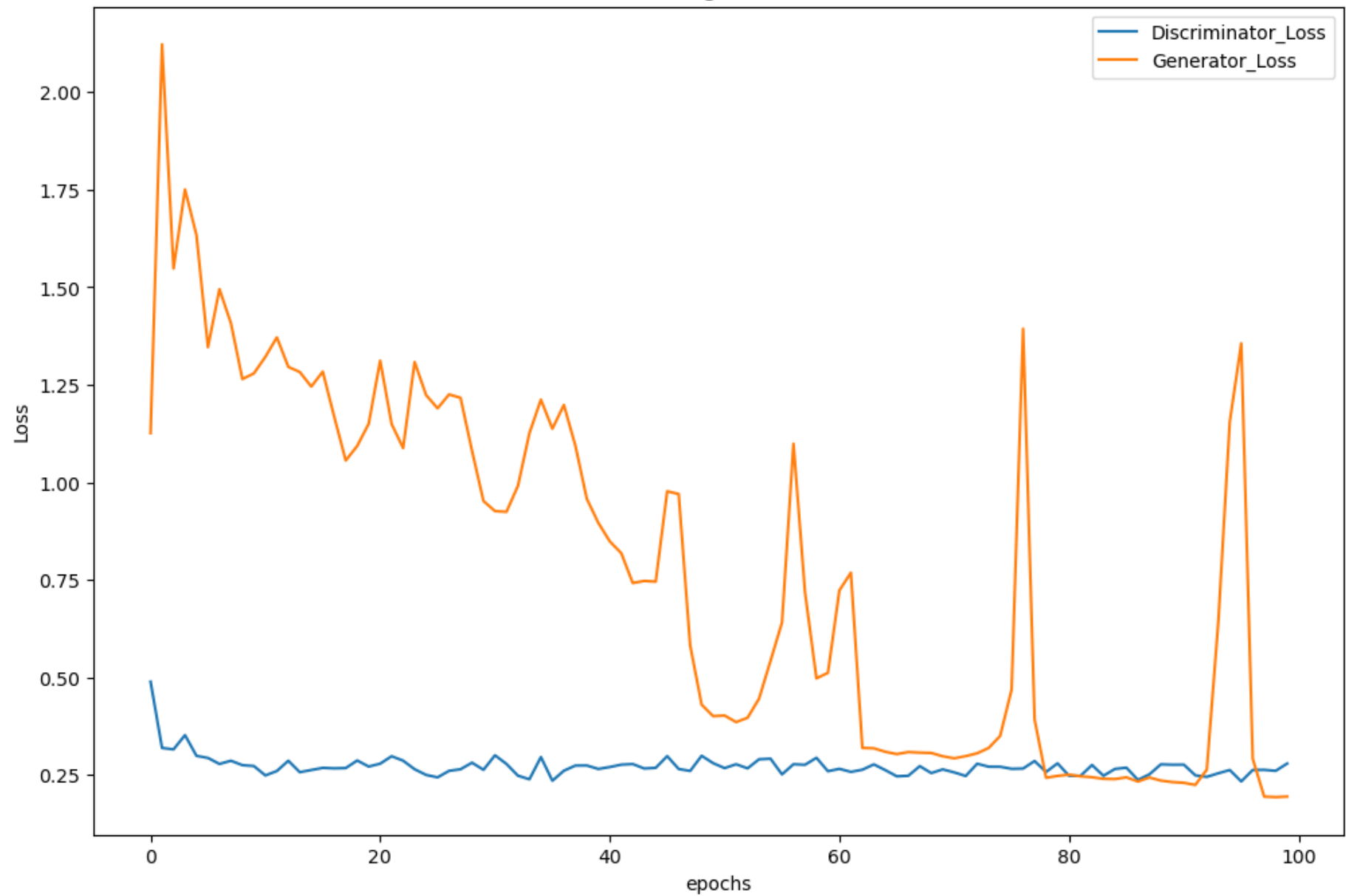
Output:

Binary Prediction –
Was this made by
the Generator or not?

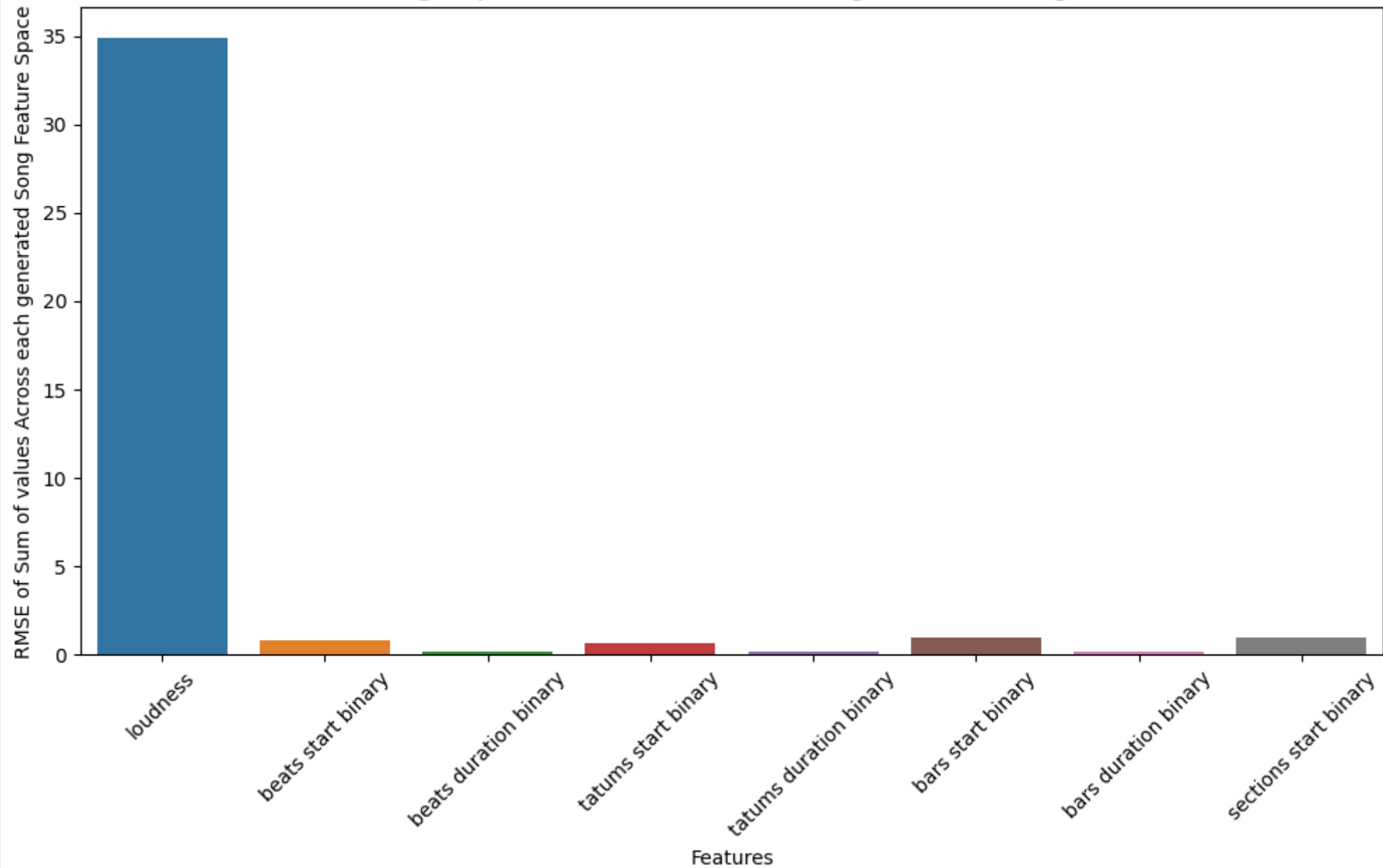
Loss Functions

- Generator 'loses' when Discriminator predicts correctly
- Discriminator 'loses' when it guesses wrong
- I used `BinaryCrossEntropy()` with these underlying factors utilized

GAN Training Performance



Average Squared Losses Across Features Against the Training Set



MVP

MVP

- Use output of the GAN to create a midi file
- Midi file into a DAW like Ableton live or Logic Pro
- *Jack sends .Wav file in chat*

Conclusions
