Spotify Song Success Prediction

Introduction/Background

In recent years, digital platforms like Spotify have grown to dominate the music industry. Streaming services like Spotify account for over 80% of all revenue generated by the music industry [1]. Spotify offers over 50 million songs to over 200 million monthly active users [2]. By analyzing historical trends from Spotify's Top Songs, we can attempt to identify the patterns that lead to a song's chart-topping success. Our team aims to build a machine learning model that can predict whether a song will perform well on the charts based on key features.

This dataset we plan on using contains the daily top 50 songs on Spotify, including features like artist, danceability, daily rank, and rank movement. The dataset can be found at: Kaggle - Top Spotify Songs Dataset

Problem Definition

Predicting the success of a song based on its characteristics can help musical artists and record labels decide how to create their music and inform decisions such as song length, genre, or artist branding in order to maximize their share of the overall music streaming market share, which reached $16.9 billion in 2021 [3].

Our team was motivated to build a model for this problem by our shared interest in music and the potential to assist artists in the industry with their music. The potential to find new, unexpected trends is also an exciting challenge that the team is eager to work on.

Methods

For our data preprocessing, we will implement:

- **Data Cleaning:** Ensure our table does not contain duplicates and handle missing data
- **Feature Engineering:** Create features like "days since release" to track the length a song has been released
- **Data Transformation:** One-hot encode categorical features like genre

We will create three different supervised machine learning models:

- Logistic Regression (scikit-learn)
- Random Forest Regression (scikit-learn)
- Neural Networks (TensorFlow/Keras)

Potential Results and Discussion

To evaluate our machine learning models, we will use key metrics:

- Accuracy (goal: >80%)
- F1-score (goal: >0.75)
- Mean Absolute Error (goal: <5 ranks)

Our project aims to contribute to the sustainability of the music industry by helping artists and producers make data-driven decisions while considering ethical implications, such as the risk of reinforcing existing trends and limiting diversity.

References

[1] N. Smith, "Spotify and the War on Artists," Michigan Journal of Economics, 29 Jan. 2024, https://sites.lsa.umich.edu/mje/2024/01/29/spotify-and-the-war-on-artists/

[2] Z. Al-Beitawi, M. Salehan, and S Zhang, "What Makes a Song Trend? Cluster Analysis of Musical Attributes for Spotify Top Trending Songs," Journal of Marketing Development and Competitiveness, 16 Sept. 2020

[3] IFPI, "Global Music Report," IFPI, [Online]. Available: https://gmr.ifpi.org/

Contribution Table

Name	Proposal Contributions
Nathan	Started and worked on the report (Introduction, Problem, Methods, Results/Discussion, References), reviewed with team help
Vishruth	Recorded the 3-minute video summarizing the report in slides
Sami	Assisted with the report (Introduction, Problem, Methods, Results/Discussion, References)
Henry	Created slides summarizing the report for the video
Trent	Set up GitHub Pages, created GitHub repo, developed Gantt chart, and filled out the contribution table

Midterm Report

Introduction/Background

To build on our original proposal, we made a machine learning pipeline that uses Spotify Top 50 chart data and YouTube engagement data to predict how long a song will remain in the charts. Our current work focuses on modeling chart longevity (in days) using a supervised regression approach. The pipeline handles full data cleaning, integration, training, and visualization. We also added features such as engagement ratios from YouTube views and likes, and encoded artist identities using multi-hot vectors. This updated dataset allows us to analyze the intersection of audio features, artist collaboration, and digital virality.

Since music virality is no longer dictated by traditional labels alone, incorporating YouTube engagement into the model helps capture a song's impact beyond Spotify. Several studies have shown that social media platforms play a vital role in music discovery and sustained listening patterns [4]. Our project uses this by combining structured audio features with public interaction data. Additionally, we observed that many songs with more basic audio features perform well due to strong online engagement. By modeling this, we hope to better represent how modern music trends function in the modern music world. [5].

This shift also aligns with a movement toward AI in media prediction, where understanding why a song succeeds is as important as predicting that it will. Our framework enables future research on factors behind musical success, supporting artists in crafting more meaningful release strategies [6].

Problem Definition

Our goal is to find out if we can predict how many days a song will remain on the Spotify US Top 50 chart based on its features. We reformulated our task as a regression problem. While our proposal described classification (hit or not), we wanted a more numerical metric of success such as chart longevity. This task is valuable for music analysts and producers who want to understand not just whether a song becomes popular, but how long it maintains that popularity.

Methods

We implemented an XGBoost regression model trained on both Spotify and YouTube data to predict chart longevity. Below are summaries of our pipeline:

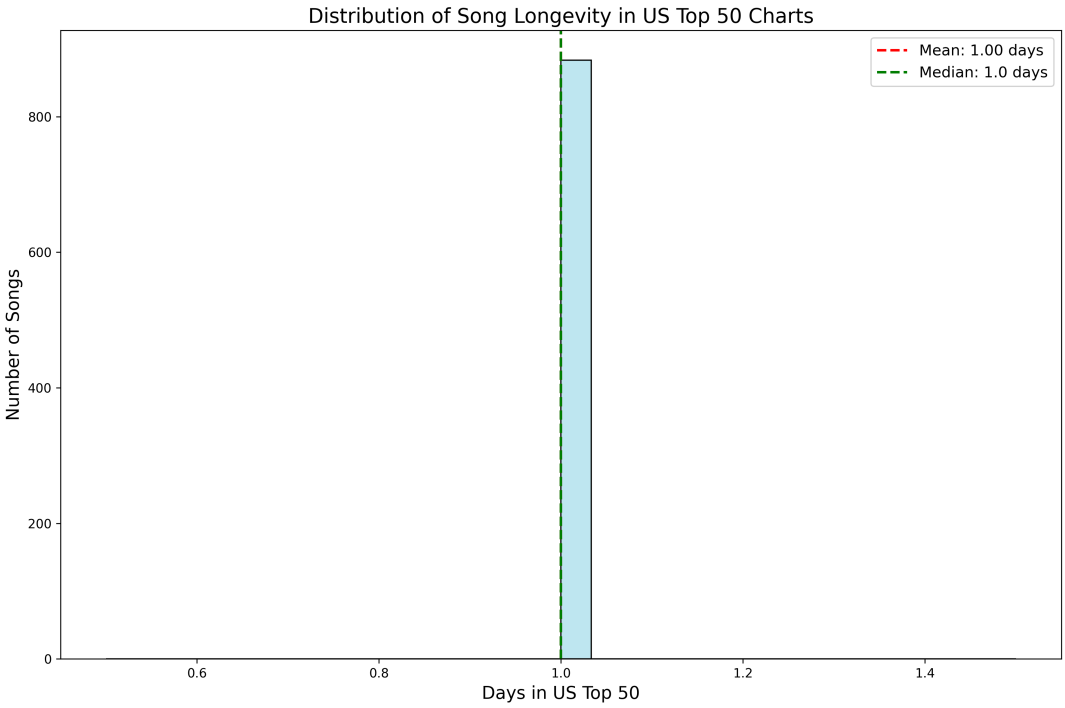
- **Data Preprocessing:**
  - Cleaned Spotify data to retain only US Top 50 entries
  - Handled missing values and removed duplicates
  - Created a unique song-level dataset aggregating performance across dates
  - Multi-hot encoded artists to represent collaboration effects
  - Scraped YouTube video statistics (views, likes, comments) via a custom API-driven scraper using song titles and artist names
  - Computed derived features such as `like_to_view_ratio` and `log_view_count` for better normalization
- **Feature Selection:**
  - **Audio Features:** danceability, energy, acousticness, valence, tempo, and others
  - **Artist Features:** multi-hot vectors for top artists
  - **YouTube Features:** view count, like count, comment count, and engagement ratios
- **Modeling Approach:**
  - We chose XGBoost regression for its ability to handle sparse, high-dimensional, and nonlinear data with higher accuracies
  - We log-transformed the target variable (`days_in_charts`) to reduce skewness and stabilize variance
  - We scaled numerical features using `StandardScaler`, preserving sparse artist encodings
  - We evaluated the model with RMSE, MAE, and R^2

Results and Discussion

To evaluate the model's ability to predict song longevity in the US Top 50 charts, we explore both regression and classification results.

Distribution of Song Longevity

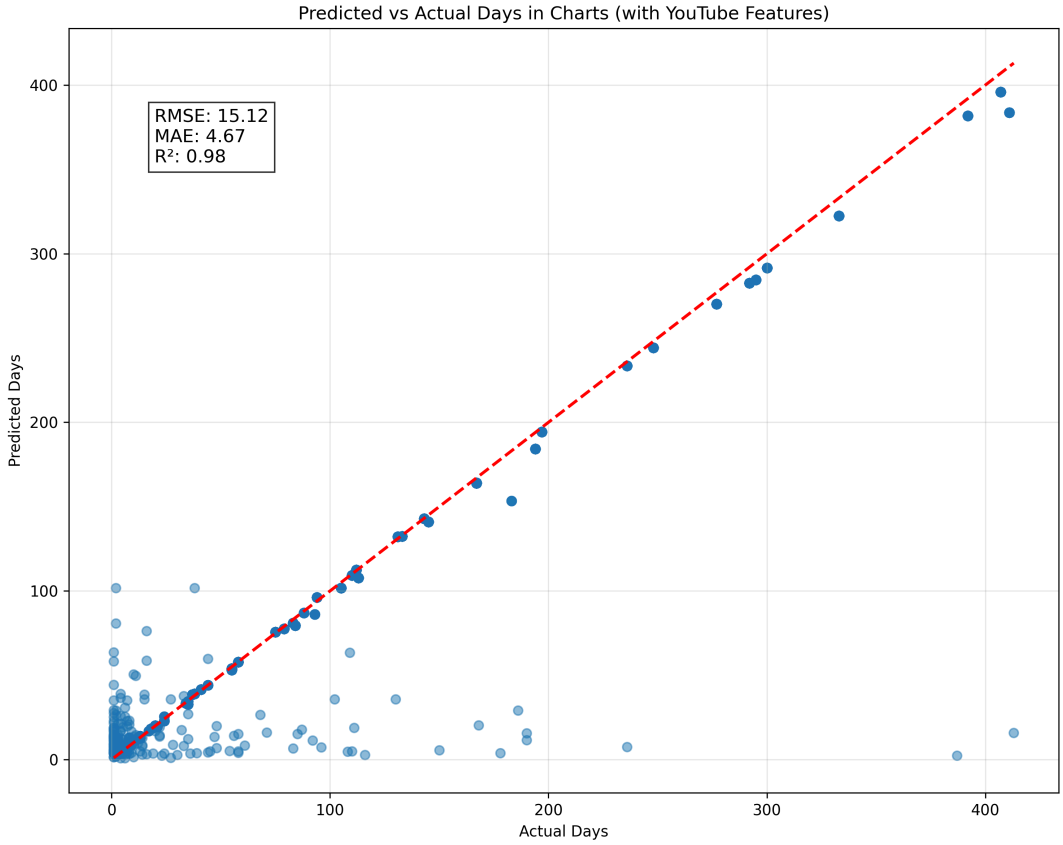
We begin with an analysis of the target variable: days in the US Top 50. As shown in the figure below, the distribution is heavily right-skewed, with most songs exiting the charts quickly. The median song duration is just 6 days, while the mean is ~29 days. This discrepancy shows the presence of a long tail of exceptionally successful songs.



Regression Performance

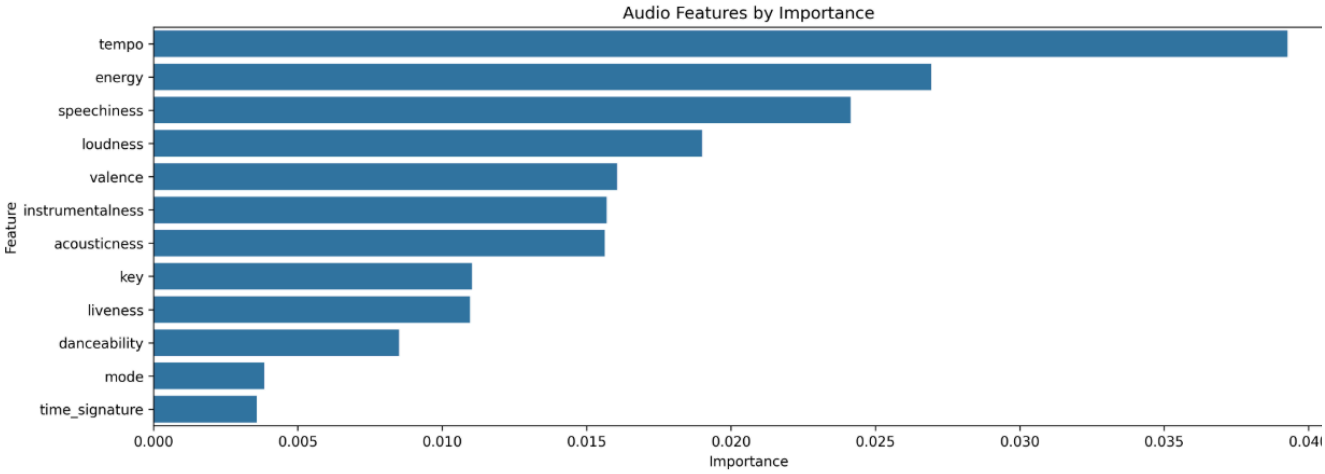
Given this distribution, we assess the performance of a regression model predicting exact days in the chart. The following figure shows the predicted versus actual values using the model. The data points mostly align along the diagonal, indicating accurate predictions. The model achieves great metrics:

- **Root Mean Squared Error (RMSE):** 15.12
- **Mean Absolute Error (MAE):** 4.67
- **R^2 Score:** 0.98



Audio Feature Importance

To understand what drives the model's predictions, we analyzed the relative importance of the input audio features. Tempo, energy, and speechiness emerged as the top three most important features, suggesting that more energetic and rhythmically distinct songs tend to remain on the charts longer. Interestingly, danceability was among the least influential, possibly showing changing music consumption trends where typical features of popular music are less predictive of chart performance.



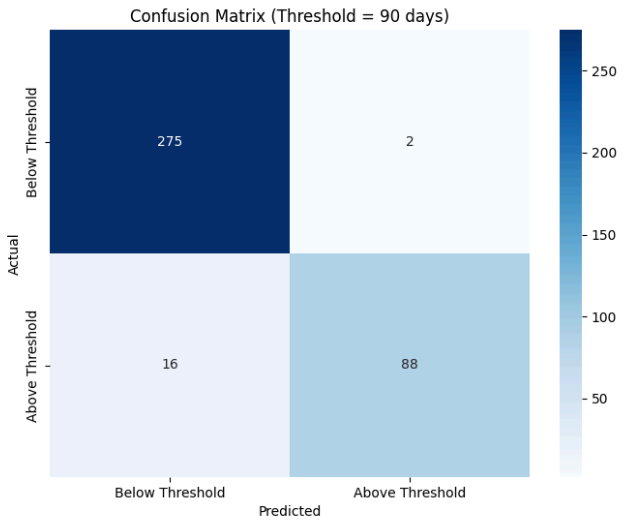
Impact of YouTube Features

To understand the effect of YouTube data, we compare models with and without these features. Including YouTube metrics significantly improves accuracy, reducing RMSE from 37.11 to 15.12 and MAE from 18.64 to 4.67. This shows that early engagement data from YouTube is a valuable predictor of future chart success.

Classification Analysis

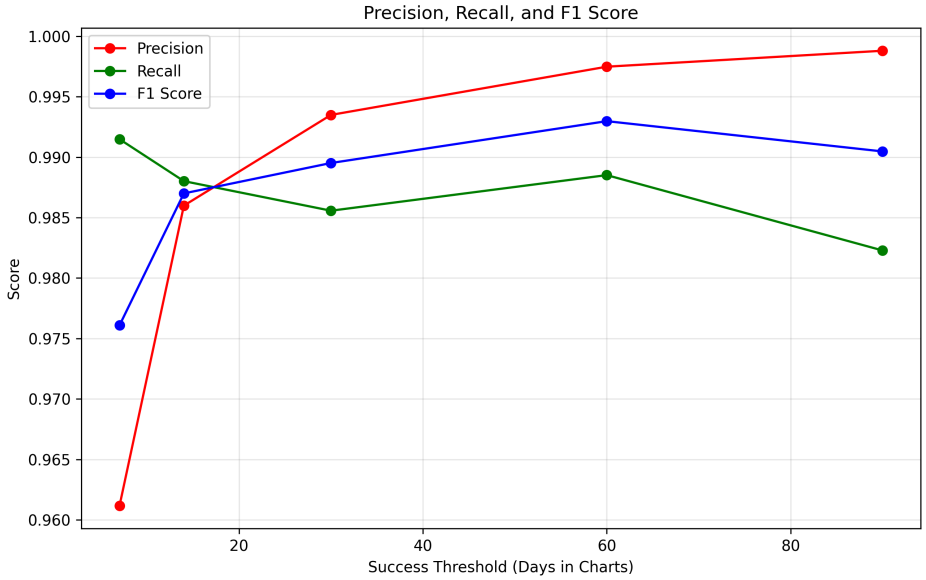
While regression offers non-binary predictions, we also frame the task as a classification problem which identifies whether a song stays in the Top 50 for more than 90 days. The confusion matrix in Figure 4 shows strong performance, with only 18 misclassifications out of 381 total songs:

- **True Positives:** 88
- **True Negatives:** 275
- **False Positives:** 2
- **False Negatives:** 16



Threshold Optimization

Finally, we evaluate classification performance over different threshold values (Figure 5). As the threshold increases, precision improves while recall slightly decreases. The F1 score peaks around the 60–90 day mark, meaning that thresholds in this range offer the best balance between precision and recall.



References

[1] N. Smith, "Spotify and the War on Artists," Michigan Journal of Economics, Jan. 2024. [Online]. Available: https://sites.lsa.umich.edu/mje/2024/01/29/spotify-and-the-war-on-artists/

[2] Z. Al-Beitawi, M. Salehan, and S. Zhang, "What Makes a Song Trend? Cluster Analysis of Musical Attributes for Spotify Top Trending Songs," *Journal of Marketing Development and Competitiveness*, vol. 14, no. 3, pp. 55–63, 2020.

[3] IFPI, "Global Music Report," IFPI, 2023. [Online]. Available: https://gmr.ifpi.org/

[4] H. Bakhshi and E. Throsby, “New technologies in cultural markets,” Journal of Cultural Economics, vol. 36, no. 3, pp. 225–231, 2012.

[5] S. Park, C. Chung, and H. Kim, “Predicting music popularity patterns with social media and streaming behaviors,” Proceedings of the ACM Conference on Recommender Systems, pp. 84–92, 2017.

[6] J. McKelvey and T. Engstrom, “Algorithms as taste: Cultural production in the age of recommendation,” Social Media + Society, vol. 6, no. 4, pp. 1–11, 2020.

Contribution Table

Name	Proposal Contributions
Nathan	Handled Spotify dataset cleaning, created artist encodings and final ML dataset, integrated country filtering
Vishruth	Helped write midterm report (methods, setup, background),implemented XGBoost model with YouTube features, coordinated scraper + preprocessing pipeline
Sami	Designed feature correlation visualizations, confusion matrices, and performance plots. Wrote results and data visualization part of report.
Henry	Assisted in tuning XGBoost parameters and visualizing error distribution
Trent	Maintained GitHub repo and GitHub Pages site, created updated Gantt chart, built file structure for pipeline execution. Heavily involved in refining model and building Youtube scraper.

Gantt Chart

Below is our team's Gantt chart outlining major milestones, deadlines, and individual responsibilities across the semester.

GANTT CHART

PROJECT TITLE	Predicting Spotify Hits Timeline

TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	M	T	W
Project Team Composition	All	1/17/24	2/2/24	15			
Project Proposal							
Introduction & Background	Trent	2/2/24	2/23/24	21			
Problem Definition	Trent	2/2/24	2/23/24	21			
Methods	Henry and Nathan	2/2/24	2/23/24	21			
Potential Dataset	Henry and Nathan	2/2/24	2/23/24	21			
Potential Results & Discussion	Sami and Vishruth	2/2/24	2/23/24	21			
Video Creation & Recording	All	2/10/24	2/23/24	13			
GitHub Page	Trent	2/10/24	2/23/24	13			
Midterm Report							
Model 1 (M1) Design & Selection	Trent, Henry and Nathan	2/17/24	2/27/24	10			
M1 Data Cleaning	Trent	2/17/24	2/27/24	10			
M1 Data Visualization	Henry	2/17/24	2/27/24	10			
M1 Feature Reduction	Nathan	2/17/24	2/27/24	10			
M1 Implementation & Coding	Sami and Vishruth	2/28/24	3/17/24	19			
M1 Results Evaluation	All	3/18/24	3/20/24	2			
Model 2 (M2) Design & Selection	-	2/28/24	3/6/24	8			
M2 Data Cleaning	-	2/28/24	3/6/24	8			
M2 Data Visualization	-	2/28/24	3/6/24	8			
M2 Feature Reduction	-	2/28/24	3/6/24	8			
M2 Coding & Implementation	-	3/7/24	3/17/24	10			
M2 Results Evaluation	-	3/18/24	3/20/24	2			
Midterm Report	All	3/28/24	3/29/24	1			
Final Report							
Model 3 (M3) Design & Selection	Trent, Henry and Nathan	3/14/24	3/20/24	6			
M3 Data Cleaning	Trent	3/14/24	3/20/24	6			
M3 Data Visualization	Henry	3/14/24	3/20/24	6			
M3 Feature Reduction	Nathan	3/14/24	3/20/24	6			
M3 Implementation & Coding	Sami	4/6/24	4/14/24	8			
M3 Results Evaluation	All	4/15/24	4/17/24	2			
M1-M3 Comparison	All	4/18/24	4/26/24	8			
Video Creation & Recording	All	4/18/24	4/26/24	8			
Final Report	All	4/18/24	4/23/24	5			

Final Report

Introduction/Background

To build on our original proposal, we made a machine learning pipeline that uses Spotify Top 50 chart data and YouTube engagement data to predict how long a song will remain in the charts. Our current work focuses on modeling chart longevity (in days) using a supervised regression approach. The pipeline handles full data cleaning, integration, training, and visualization. We also added features such as engagement ratios from YouTube views and likes, and encoded artist identities using multi-hot vectors. This updated dataset allows us to analyze the intersection of audio features, artist collaboration, and digital virality.

Since music virality is no longer dictated by traditional labels alone, incorporating YouTube engagement into the model helps capture a song's impact beyond Spotify. Several studies have shown that social media platforms play a vital role in music discovery and sustained listening patterns [4]. Our project uses this by combining structured audio features with public interaction data. Additionally, we observed that many songs with more basic audio features perform well due to strong online engagement. By modeling this, we hope to better represent how modern music trends function in the modern music world. [5].

This shift also aligns with a movement toward AI in media prediction, where understanding why a song succeeds is as important as predicting that it will. Our framework enables future research on factors behind musical success, supporting artists in crafting more meaningful release strategies [6].

Problem Definition

Our goal is to find out if we can predict how many days a song will remain on the Spotify US Top 50 chart based on its features. We reformulated our task as a regression problem. While our proposal described classification (hit or not), we wanted a more numerical metric of success such as chart longevity. This task is valuable for music analysts and producers who want to understand not just whether a song becomes popular, but how long it maintains that popularity.

Methods

We implemented XGBoost regression, linear regression, and ElasticNet regression models trained on both Spotify and YouTube data to predict chart longevity. Below are summaries of our pipeline:

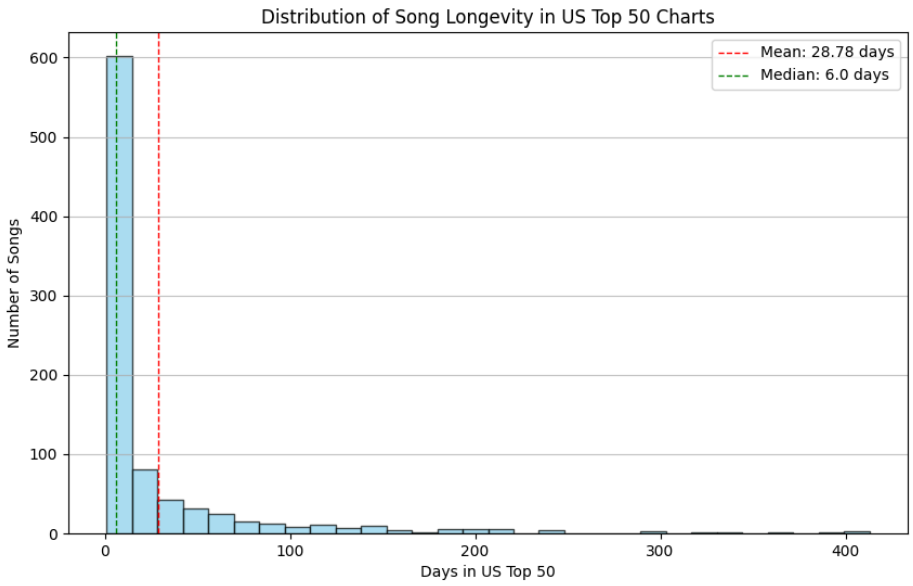
- **Data Preprocessing:**
  - Cleaned Spotify data to retain only US Top 50 entries
  - Handled missing values and removed duplicates
  - Created a unique song-level dataset aggregating performance across dates
  - Multi-hot encoded artists to represent collaboration effects
  - Scraped YouTube video statistics (views, likes, comments) via a custom API-driven scraper using song titles and artist names
  - Computed derived features such as `like_to_view_ratio` and `log_view_count` for better normalization
- **Feature Selection:**
  - **Audio Features:** danceability, energy, acousticness, valence, tempo, and others
  - **Artist Features:** multi-hot vectors for top artists
  - **YouTube Features:** view count, like count, comment count, and engagement ratios
- **XGBoost Regression:**
  - We chose XGBoost regression for its ability to handle sparse, high-dimensional, and nonlinear data with higher accuracies
  - We log-transformed the target variable (`days_in_charts`) to reduce skewness and stabilize variance
  - We scaled numerical features using `StandardScaler`, preserving sparse artist encodings
  - We evaluated the model with RMSE, MAE, and R^2
- **Linear Regression:**
  - We chose XGBoost regression as it can serve as a simple, interpretable baseline for our regression models
  - We found that feature engineering significantly improved the accuracy of the regression model (more on this in the Results/Analysis section
  - Expectedly, the model struggled to capture non-linear patterns in the data
  - We evaluated the model with RMSE, MAE, and R^2

- **ElasticNet Regression:**
  - We chose Elastic regression as it combines L1 (Lasso) and L2 (Ridge) penalties to handle sparsity and multicollinearity, which allowed us to shrink uninformative one-hot artist coefficients toward zero without dropping them entirely
  - Retains most interpretability of a linear model while controlling overfitting
  - Best on raw one-hot data: sparse encoding + regularization gave lowest original RMSE
  - Engineered features added noise rather than signal under ElasticNet's penalty mix
  - We evaluated the model with RMSE, MAE, and R^2

Results and Discussion

Distribution of Song Longevity

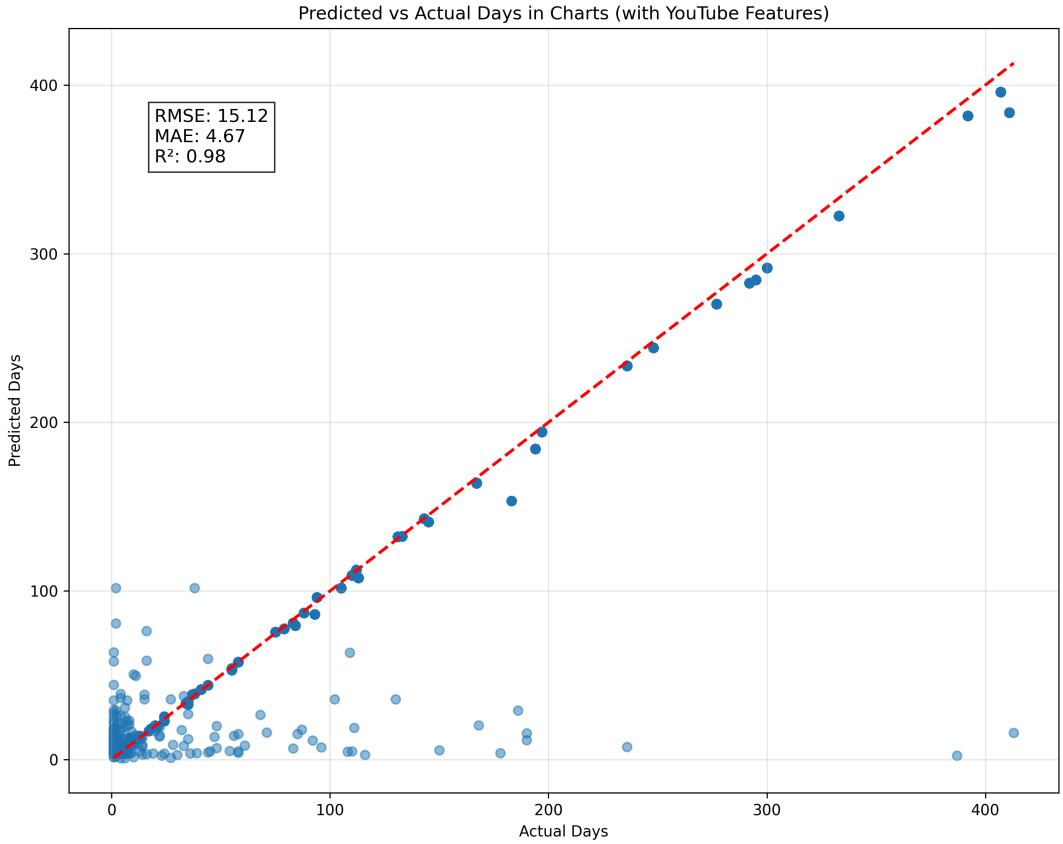
We begin with an analysis of the target variable: days in the US Top 50. As shown in the figure below, the distribution is heavily right-skewed, with most songs exiting the charts quickly. The median song duration is just 6 days, while the mean is ~29 days. This discrepancy shows the presence of a long tail of exceptionally successful songs.



Regression Performance

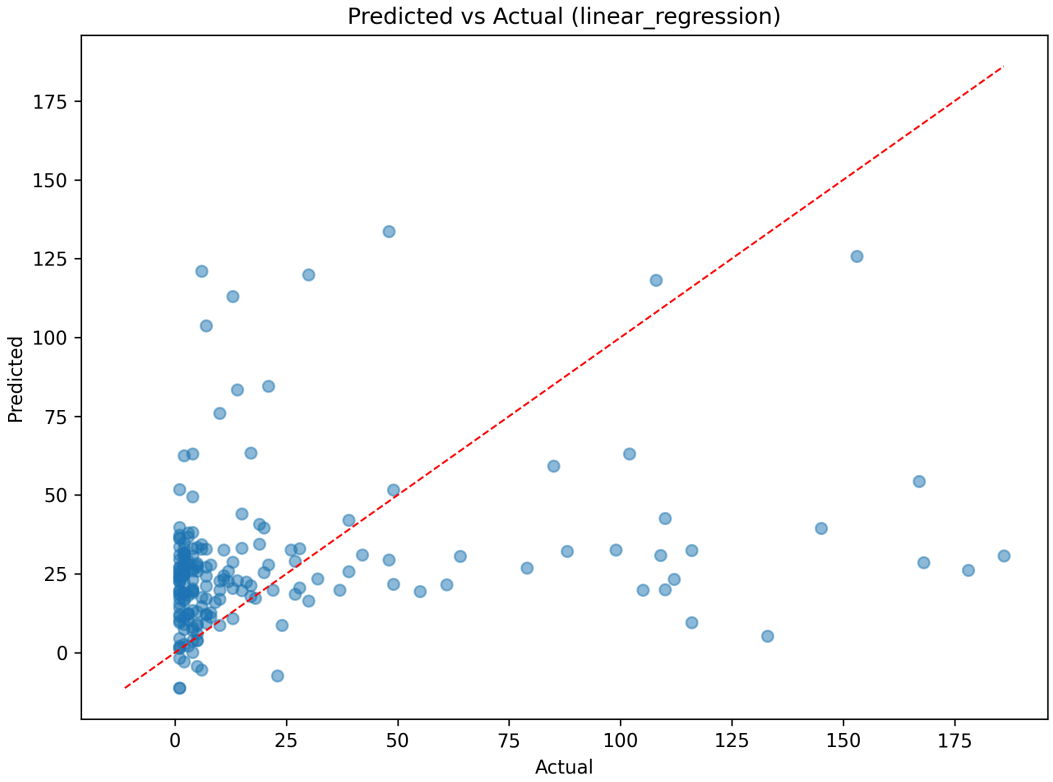
Given this distribution, we assess the performance of our XGBoost regression model predicting exact days in the chart. The following figure shows the predicted versus actual values using the model. The data points mostly align along the diagonal, indicating accurate predictions. The model achieves great metrics:

- **Root Mean Squared Error (RMSE):** 15.12
- **Mean Absolute Error (MAE):** 4.67
- **R^2 Score:** 0.98



Next, we assess the performance of our Linear regression model predicting exact days in the chart. The following figure shows the predicted versus actual values using the model. The model achieves the following metrics:

- **Root Mean Squared Error (RMSE):** 46.49
- **Mean Absolute Error (MAE):** 30.84
- **Pearson:** 0.2075
- **Spearman:** 0.1871

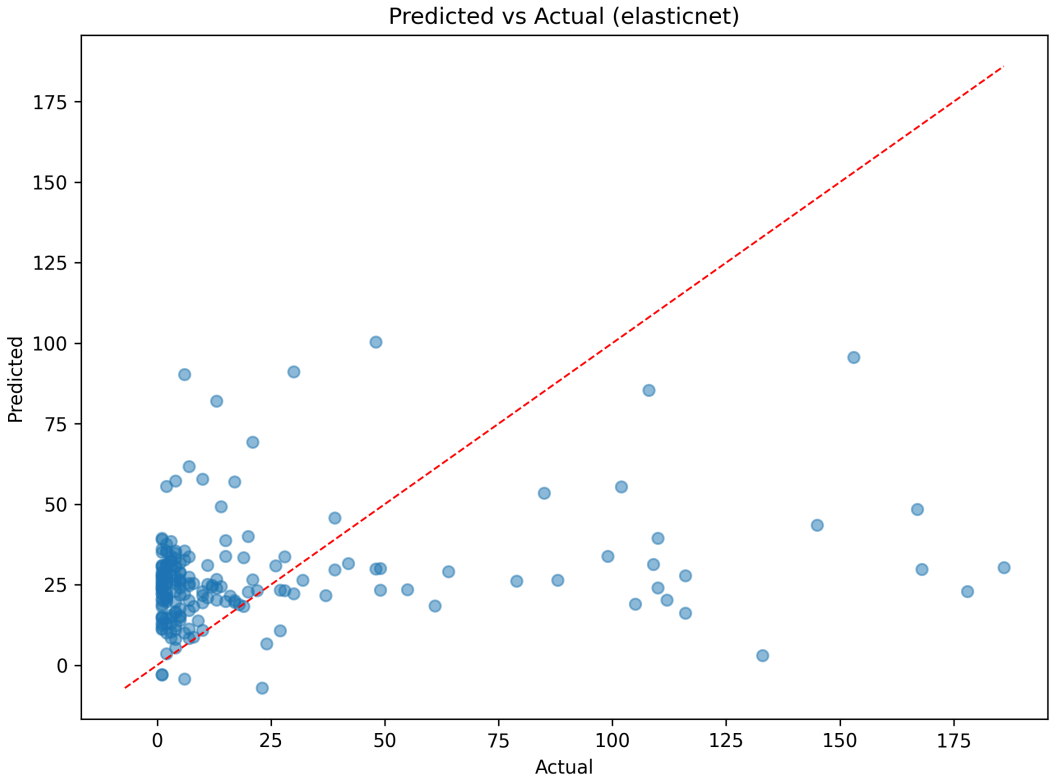


We can also compare how this linear regression model performed with our feature engineered data. This results in the following metrics:

- **Root Mean Squared Error (RMSE):** 39.80
- **Mean Absolute Error (MAE):** 26.76
- **Pearson:** 0.2537
- **Spearman:** 0.2592

Thus, we see that our linear regression model responds well to feature engineering, cutting MAE by ~14%. We still see a significant decrease in the accuracy of our linear regression model compared to our XGBoost regression model. Next, we assess the performance of our ElasticNet regression model predicting exact days in the chart. The model achieves the following metrics:

- **Root Mean Squared Error (RMSE):** 36.98
- **Mean Absolute Error (MAE):** 26.79
- **Pearson:** 0.29
- **Spearman:** 0.17



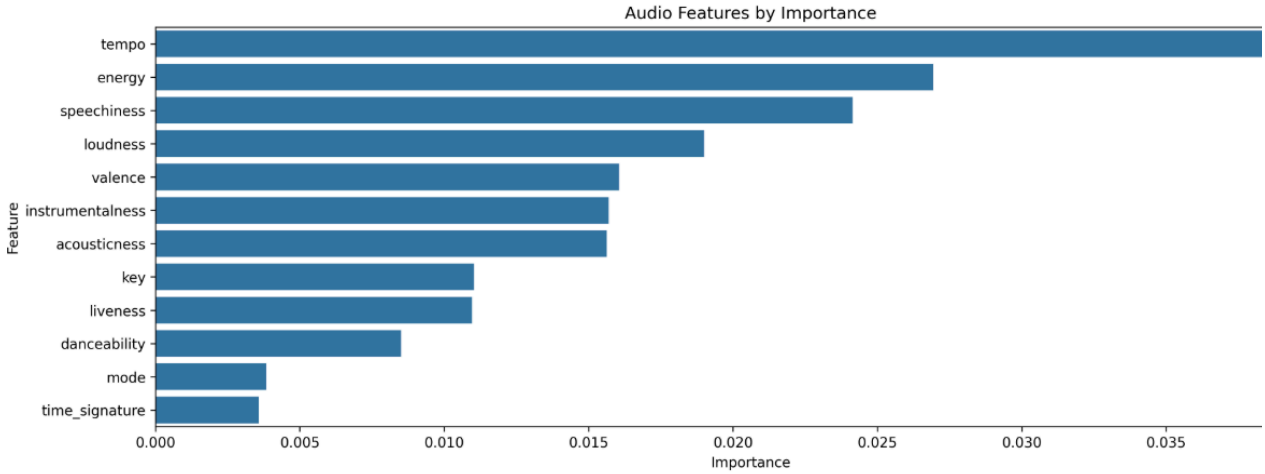
We can also see the ElasticNet regression's response to our feature engineering below:

- **Root Mean Squared Error (RMSE):** 38.10
- **Mean Absolute Error (MAE):** 27.01
- **Pearson:** 0.24
- **Spearman:** 0.16

Interestingly, the engineered features added noise rather than signal under ElasticNet's penalty mix,resulting in a less accurate regression model.

Audio Feature Importance

To understand what drives the model's predictions, we analyzed the relative importance of the input audio features. Tempo, energy, and speechiness emerged as the top three most important features, suggesting that more energetic and rhythmically distinct songs tend to remain on the charts longer. Interestingly, danceability was among the least influential, possibly showing changing music consumption trends where typical features of popular music are less predictive of chart performance.



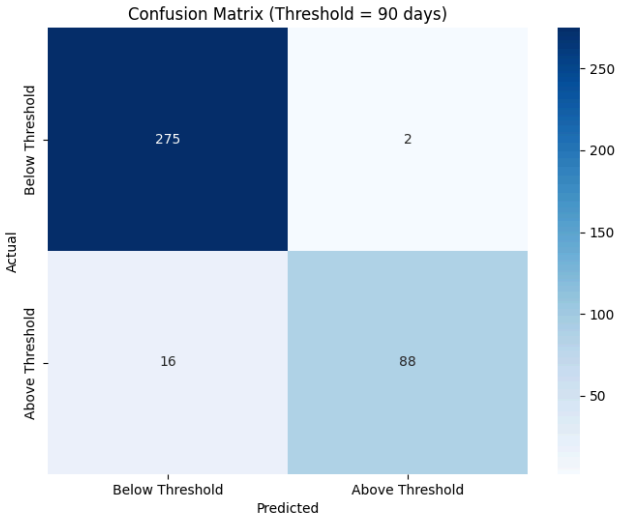
Impact of YouTube Features

To understand the effect of YouTube data, we compare models with and without these features. Including YouTube metrics significantly improves accuracy, reducing RMSE from 37.11 to 15.12 and MAE from 18.64 to 4.67 in our XGBoost regression model. This shows that early engagement data from YouTube is a valuable predictor of future chart success.

Classification Analysis

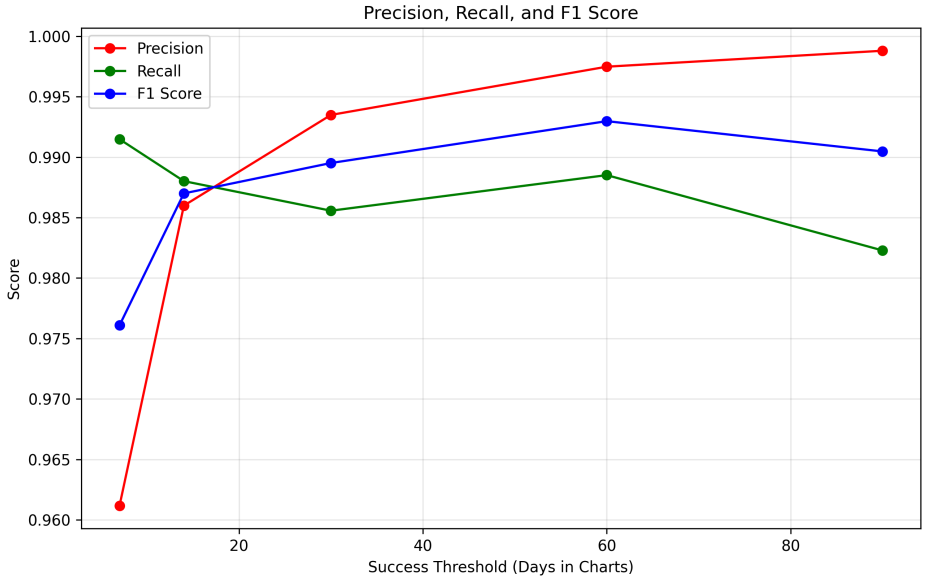
While regression offers non-binary predictions, we also frame the task as a classification problem which identifies whether a song stays in the Top 50 for more than 90 days. The confusion matrix shows strong performance, with only 18 misclassifications out of 381 total songs:

- **True Positives:** 88
- **True Negatives:** 275
- **False Positives:** 2
- **False Negatives:** 16



Threshold Optimization

Next, we evaluate classification performance over different threshold values. As the threshold increases, precision improves while recall slightly decreases. The F1 score peaks around the 60–90 day mark, meaning that thresholds in this range offer the best balance between precision and recall.



Comparision of Models

We can compare the results of the models we created the best analyze our model's performance. With a RMSE = 15.12 and a MAE = 4.67, our XGBoost regression model performed significantly better than our other two models. This model was the only model to meet our success threshold of <5 MAE. This model responded well to the high-dimensional data that we input into the model. The next best model in terms of MAE was our linear regression with feature engineering, with a MAE = 26.76 (a 14% decrease from the linear regression without feature engineering). Our ElasticNet regression model was last with a MAE = 26.79. This model reacted negatively to feature engineering (resulting in a MAE = 27.01), although neither set of data met the success threshold.

References

[1] N. Smith, "Spotify and the War on Artists," Michigan Journal of Economics, Jan. 2024. [Online]. Available: https://sites.lsa.umich.edu/mje/2024/01/29/spotify-and-the-war-on-artists/

[2] Z. Al-Beitawi, M. Salehan, and S. Zhang, "What Makes a Song Trend? Cluster Analysis of Musical Attributes for Spotify Top Trending Songs," *Journal of Marketing Development and Competitiveness*, vol. 14, no. 3, pp. 55–63, 2020.

[3] IFPI, "Global Music Report," IFPI, 2023. [Online]. Available: https://gmr.ifpi.org/

[4] H. Bakhshi and E. Throsby, “New technologies in cultural markets,” Journal of Cultural Economics, vol. 36, no. 3, pp. 225–231, 2012.

[5] S. Park, C. Chung, and H. Kim, “Predicting music popularity patterns with social media and streaming behaviors,” Proceedings of the ACM Conference on Recommender Systems, pp. 84–92, 2017.

[6] J. McKelvey and T. Engstrom, “Algorithms as taste: Cultural production in the age of recommendation,” Social Media + Society, vol. 6, no. 4, pp. 1–11, 2020.

Contribution Table

Name	Proposal Contributions
Nathan	Handled Spotify dataset cleaning, created artist encodings and final ML dataset, integrated country filtering, wrote final report
Vishruth	Helped write midterm report (methods, setup, background),implemented XGBoost model with YouTube features, coordinated scraper + preprocessing pipeline
Sami	Designed feature correlation visualizations, confusion matrices, and performance plots. Wrote results and data visualization part of report.
Henry	Assisted in tuning XGBoost parameters and visualizing error distribution
Trent	Maintained GitHub repo and GitHub Pages site, created updated Gantt chart, built file structure for pipeline execution. Heavily involved in refining model and building Youtube scraper.

Gantt Chart

Below is our team’s Gantt chart outlining major milestones, deadlines, and individual responsibilities across the semester.

GANTT CHART

PROJECT TITLE	Predicting Spotify Hits Timeline

TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	M	T	W
Project Team Composition	All	1/17/24	2/2/24	15			
Project Proposal							
Introduction & Background	Trent	2/2/24	2/23/24	21			
Problem Definition	Trent	2/2/24	2/23/24	21			
Methods	Henry and Nathan	2/2/24	2/23/24	21			
Potential Dataset	Henry and Nathan	2/2/24	2/23/24	21			
Potential Results & Discussion	Sami and Vishruth	2/2/24	2/23/24	21			
Video Creation & Recording	All	2/10/24	2/23/24	13			
GitHub Page	Trent	2/10/24	2/23/24	13			
Midterm Report							
Model 1 (M1) Design & Selection	Trent, Henry and Nathan	2/17/24	2/27/24	10			
M1 Data Cleaning	Trent	2/17/24	2/27/24	10			
M1 Data Visualization	Henry	2/17/24	2/27/24	10			
M1 Feature Reduction	Nathan	2/17/24	2/27/24	10			
M1 Implementation & Coding	Sami and Vishruth	2/28/24	3/17/24	19			
M1 Results Evaluation	All	3/18/24	3/20/24	2			
Model 2 (M2) Design & Selection	-	2/28/24	3/6/24	8			
M2 Data Cleaning	-	2/28/24	3/6/24	8			
M2 Data Visualization	-	2/28/24	3/6/24	8			
M2 Feature Reduction	-	2/28/24	3/6/24	8			
M2 Coding & Implementation	-	3/7/24	3/17/24	10			
M2 Results Evaluation	-	3/18/24	3/20/24	2			
Midterm Report	All	3/28/24	3/29/24	1			
Final Report							
Model 3 (M3) Design & Selection	Trent, Henry and Nathan	3/14/24	3/20/24	6			
M3 Data Cleaning	Trent	3/14/24	3/20/24	6			
M3 Data Visualization	Henry	3/14/24	3/20/24	6			
M3 Feature Reduction	Nathan	3/14/24	3/20/24	6			
M3 Implementation & Coding	Sami	4/6/24	4/14/24	8			
M3 Results Evaluation	All	4/15/24	4/17/24	2			
M1-M3 Comparison	All	4/18/24	4/26/24	8			
Video Creation & Recording	All	4/18/24	4/26/24	8			
Final Report	All	4/18/24	4/23/24	5			