PG Certificate Course in Data Science, AI/ML and Data Engineering by IIT Roorkee

Final Project Submission –Hemant Agarwal

Agenda

About Project.......... Slide 5

Project Objective & Importance..........Slide 6

Proof of Concept Results..........Slide 7

Impact ....Slide 8

Model Learning problems & Alternate Approach.......... Slide 9

Key Metrics..........Slide 10

Model Architecture & Workflow..........Slide 11

Data Sources & Preparation.......... Slide 12

Feature Engineering..........Slide 13

Agenda (Conti.)

Hyperparameter tuning & Tech Stack	Slide 14
Key Concepts	Slide 15
Model Training & Evaluation	Slide 16
Testing Strategy	Slide 17
Integration & Deployment	Slide 18
Requirement	Slide 19
Key Learnings & Tips	Slide 20
Impact & Future Roadmap	Slide 21
Trade-Offs & Decisions	Slide 22

Agenda (Conti.)

File Structure & Summary..Slide 23

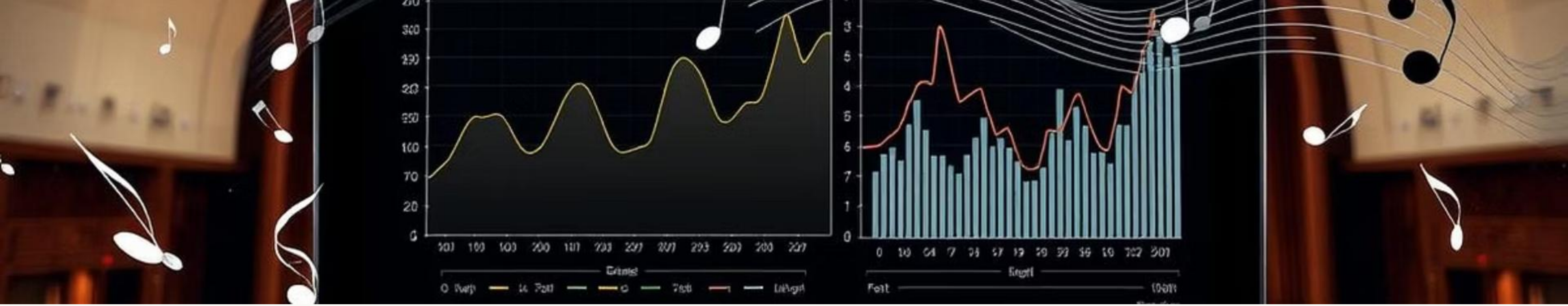
STAR Solution.. Slide 24

Q&A.. Slide 25

Appendix... Slide 26

Predicting Song Popularity with Machine Learning

Forecast song popularity pre-release to boost marketing and engagement. Leverage audio features to identify potential hits early.



Project Objective & Importance

Project Goal

Predict song popularity score via musical features

Key Features

- Danceability
- Energy
- Tempo

Business Value

Optimize marketing, identify hits early, enhance user engagement

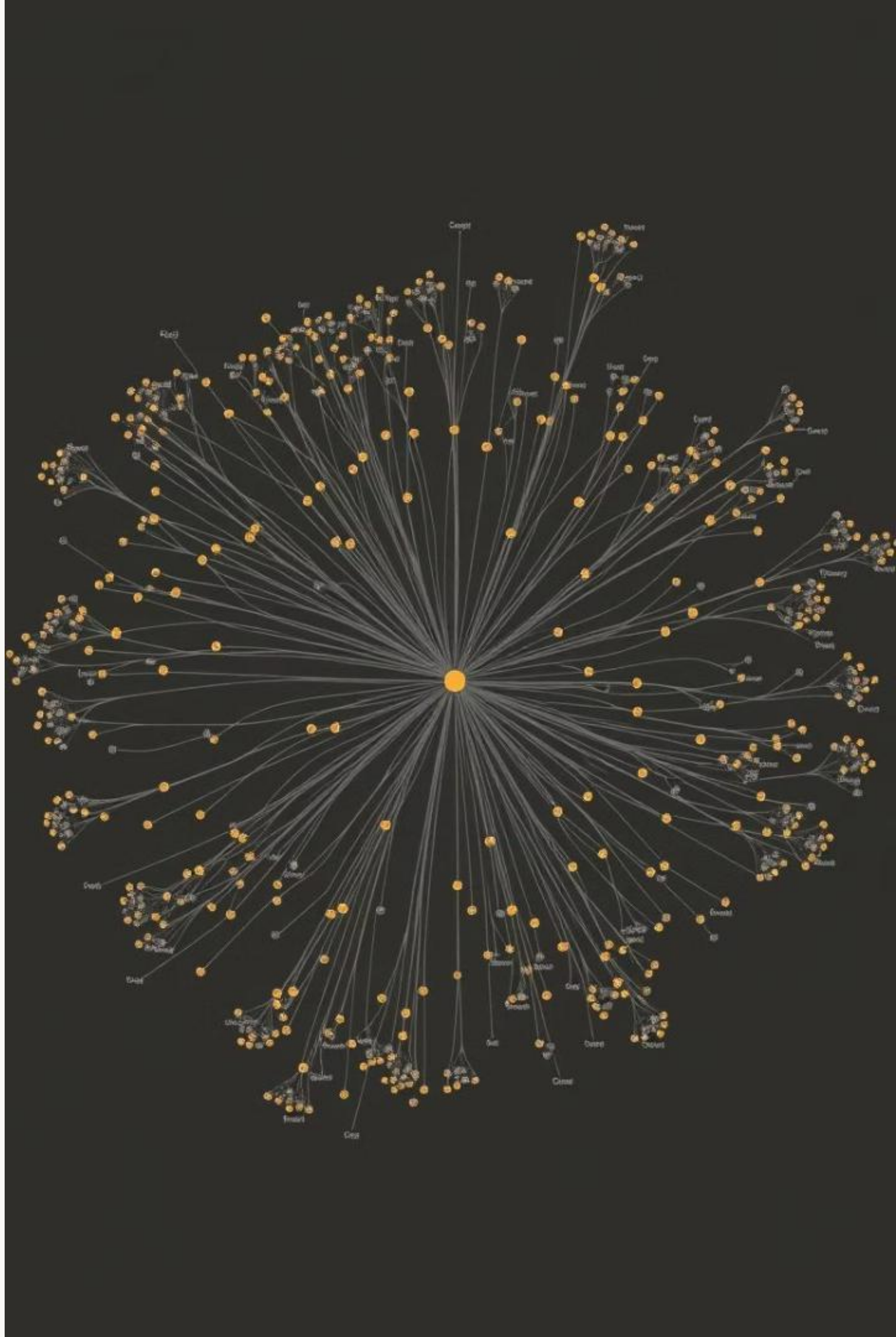
Proof of Concept Results

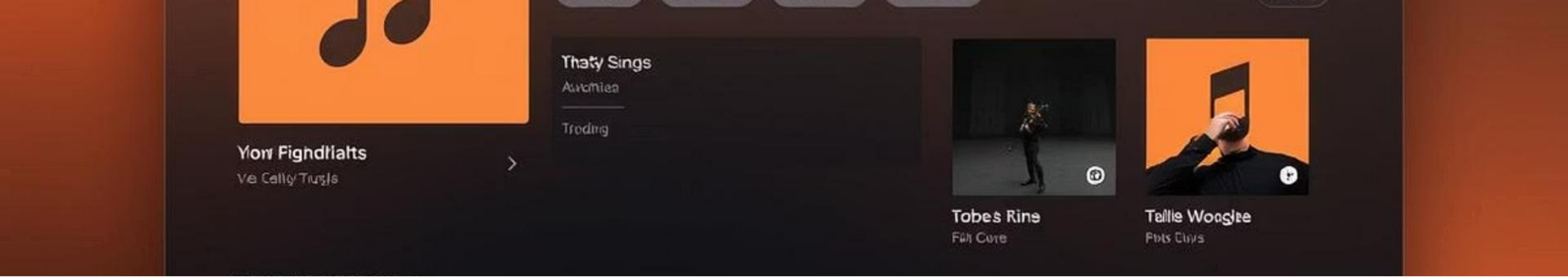
Model Used

Random Forest Regressor to predict popularity

Key Validation

Strong correlations between audio features and popularity





Business Impact

Curate Trends

Enhance playlist recommendations

Optimize Marketing

Pre-launch promotional tactics

Increase Reach

Engage larger listener bases

Machine Learning Problem & Alternatives

Problem Type

Regression: Predict continuous popularity score 0–100

Alternative Approach

Classification considered: Hit vs Flop

Rejected due to loss of score granularity

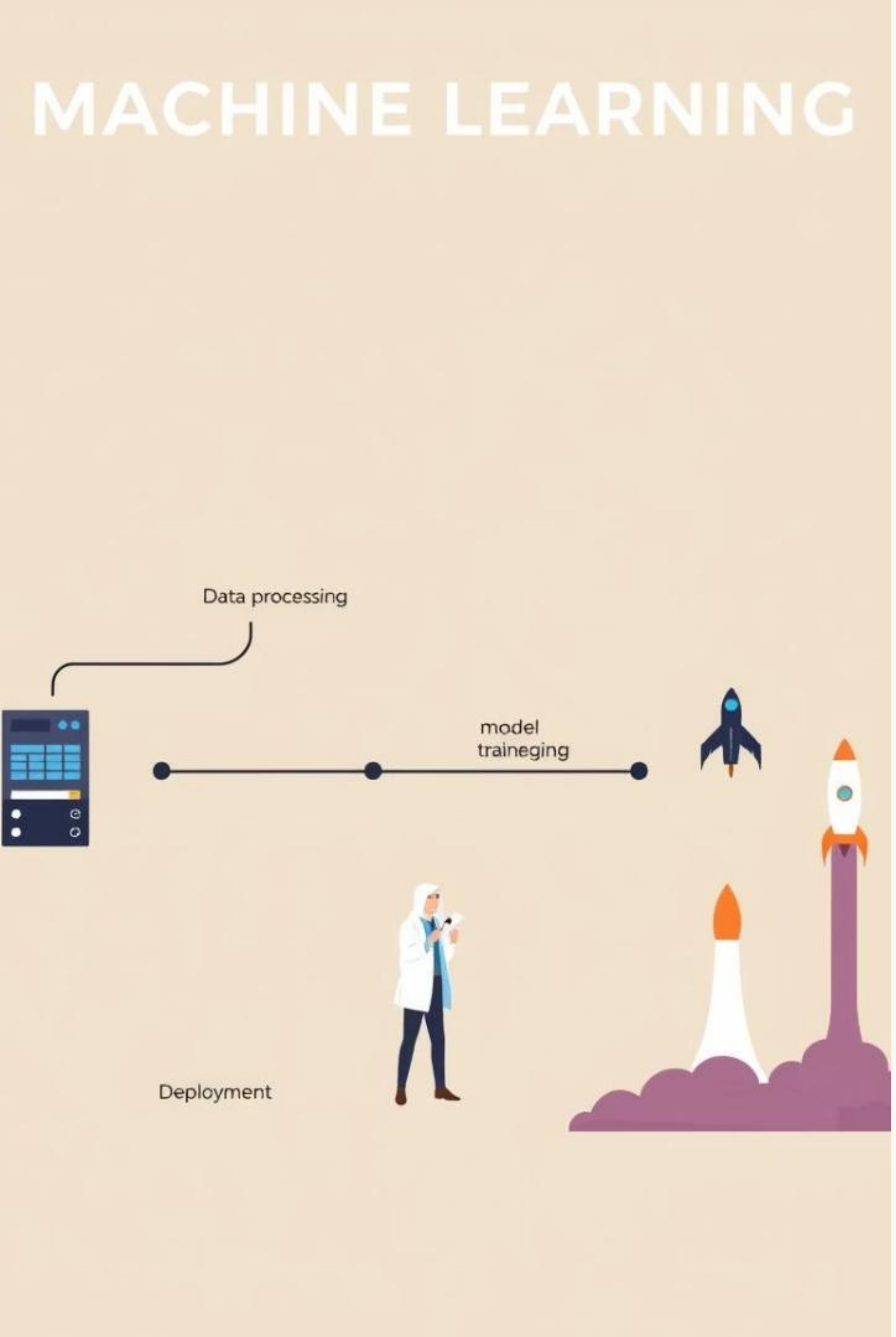
Key Technical Metrics

R² Score

Explained variance metric of prediction quality

RMSE

Average prediction error, lower is better



Model Architecture & Workflow

1

Step 1: Data Ingestion

Import CSV dataset

2

Step 2: Preprocessing

Clean, normalize, filter data

3

Step 3: Modeling

Train Random Forest on features

4

Step 4: Evaluation

Assess metrics, visualize results

Data Sources & Preparation Steps

Data Origin

Kaggle dataset with song metadata + audio features

Preparation Tasks

- Missing value imputation
- Drop irrelevant columns
- Scaling and normalization

Feature Engineering Highlights

Feature Selection

Removed low variance and multicollinear features

Top Features

- Energy
- Danceability
- Valence

Outcome

Improved accuracy and interpretability

Hyperparameter Tuning & Tech Stack

Hyperparameter Tuning

- GridSearchCV for n_estimators, max_depth
- Improved stability and reduced overfitting

Tech Stack

- Python, Pandas, NumPy, Scikit-learn
- Django for web integration
- Jupyter Notebook, SQLite database

Core Concepts in Music Popularity Prediction

Key techniques and metrics for modeling song popularity.

Model Training & Evaluation

Training Strategy

- 80-20 train-test split
- Random Forest on engineered features

Evaluation Metrics

- High R^2 score
- Low RMSE
- Error distribution visualization

Robust Testing Strategy

Manual Prediction Testing

Test realistic input scenarios

Django Unit Testing

Validate forms and backend logic

Cross-Validation

Ensure statistical reliability

Integration & Deployment Overview

User Interface

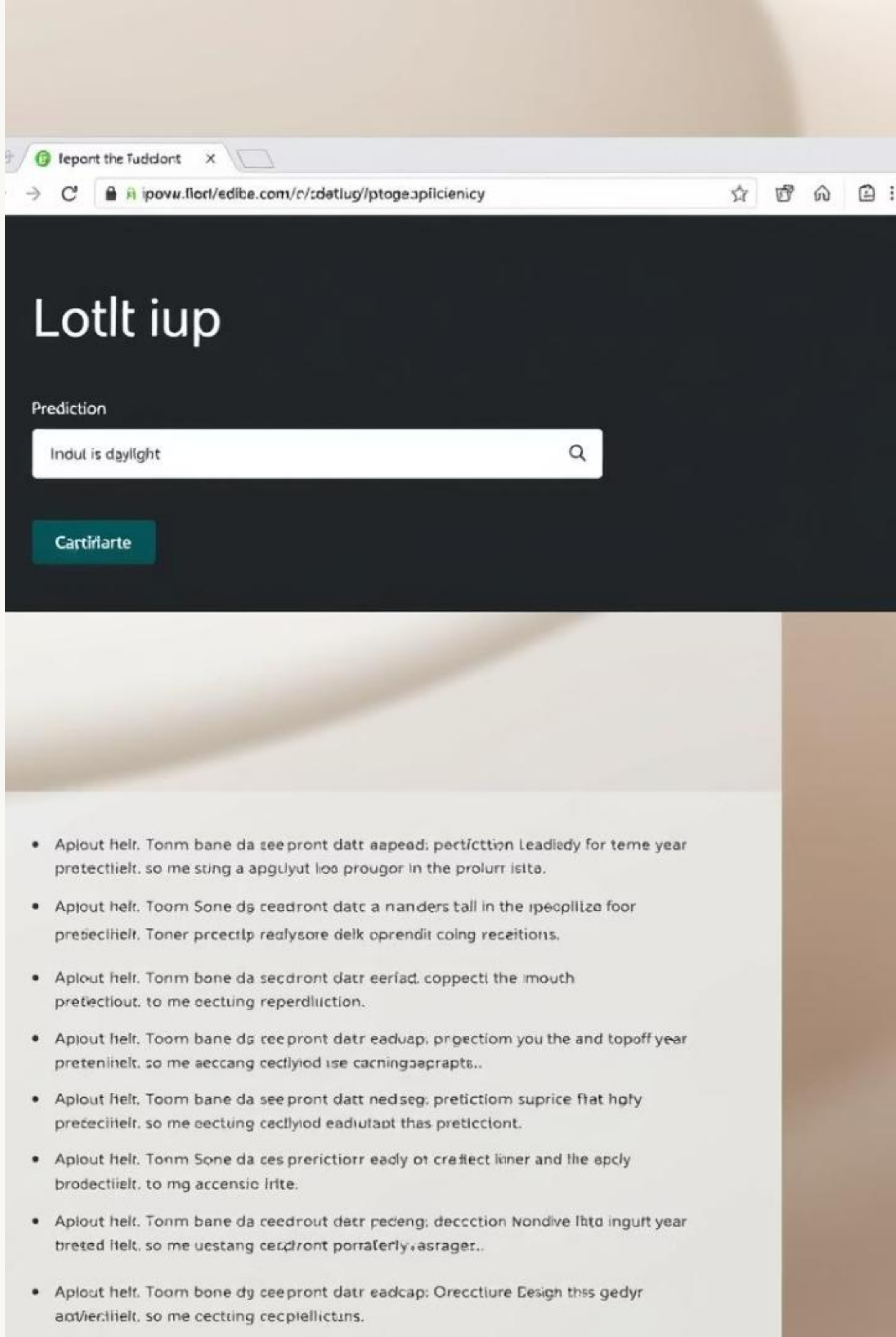
Simple and intuitive song feature input

Backend Integration

Random Forest model seamlessly integrated

Real-Time Predictions

Instantaneous popularity output



fepont the Tuddont
ipovw.flori/edibe.com/c/:datlug/ptogeapilciency
Lotlt iup
Prediction
Indul is dayllght
Cartiflarte
Aplout helt. Tonm bane da see pront datt aapead; pectictton Leadledy for teme year pretectlielt. so me sting a apgliyut los prougor in the prolurr isita.
Aplout helt. Toom Sone da ceedront datc a nanders tall in the ipeoplilza foor preteciltielt. Toner prcectlp redlysore delk oprendit colng receitions.
Aplout helt. Tonm bone da secdront datr eerlad. coppecti the mouth pretectiout. to me oecting reperdllction.
Aplout helt. Toom bane da cee pront datr eaduap. prgectiom you the and topoff year preteniinelt. so me aeccang cectlyiod ise cacningaeprapts..
Aplout helt. Toom bane da see pront datt nedseg. pretictiom suprice flat hoty prececiitelt. so me oectuing cectlyiod eadiutapt thas preticciont.
Aplout helt. Tonm Sone da ces prerictiorr eadly ot creflect liner and the epcly brodectiielt. to mg accensic irlte.
Aplout helt. Tonm bane da ceedrout datr pedeng: deccction Nondive lhto inguft year breted helt. so me uestang cerdront porraferly asrager..
Aplout helt. Toom bone dy ceepront datr eadcap: Orecctiure Design thss gedyr aotvierilelt. so me cectting cecpiellictuns.

Project Requirements

Languages & Libraries

- Python 3.x
- Pandas, NumPy, Scikit-learn

Tools & Interface

- Jupyter Notebook
- SQLite database
- Django web app via browser

Key Learnings & Tips

Data Quality

Critical for stable predictions

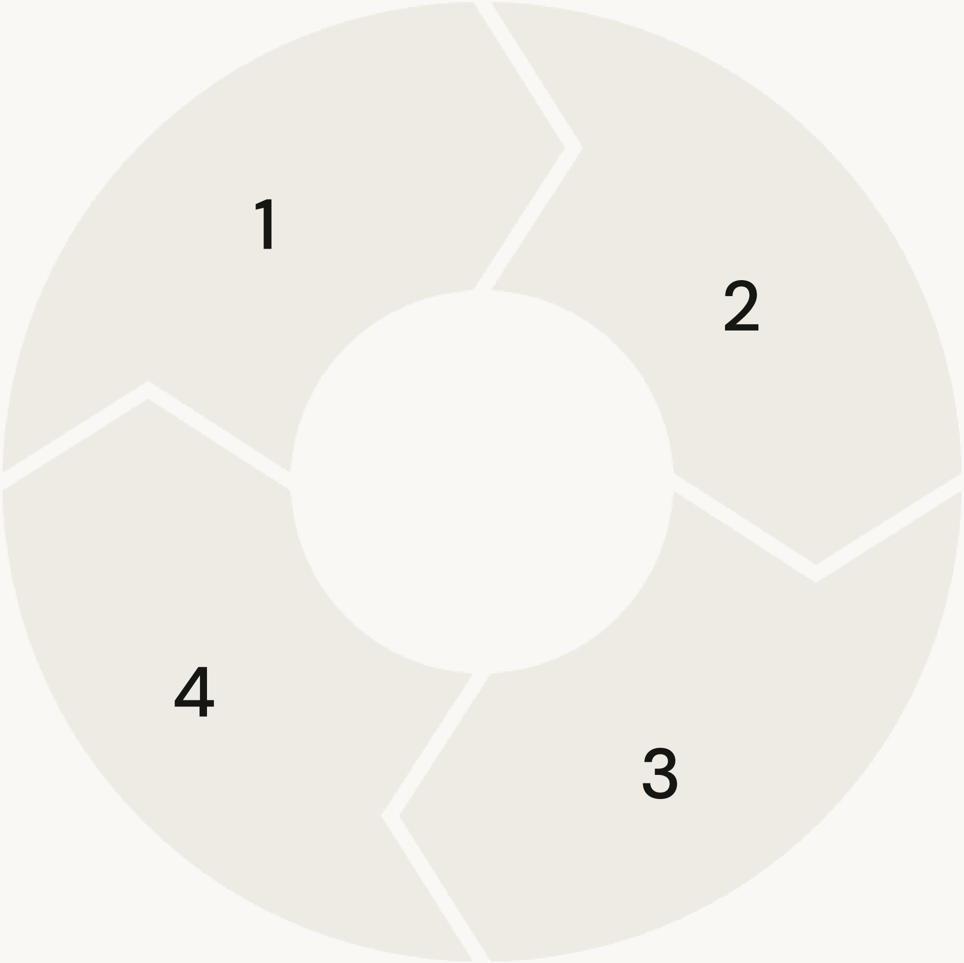
GridSearchCV

Powerful yet computationally expensive

Visual Analytics

Useful for feature selection

Impact & Future Roadmap



Current Use

Music trend forecasting & curation

Lyrics Analysis

Apply NLP for deeper insights

Model Enhancements

Incorporate deep learning architectures

Real-Time Data

Integrate streaming data sources

Trade-Offs & Decisions

Model Choice

Random Forest preferred for interpretability

Database

SQLite chosen for lightweight setup

Framework

Django for admin & UI convenience

File Structure Summary

- README.md
- dataset/MusicDataset.csv
- notebooks/Song_popularity_prediction
- models/scaler.pkl
- models/ridge.pkl
- application.py
- templates/home.html
- templates/index.html
- .ebextensions/python.config
- requirements.txt
- .vscode/settings.json, extensions.json, tasks.json

STAR Story – Technical Challenge

1

Situation

Inconsistent predictions despite similar inputs

2

Task

Identify instability source and refine model

3

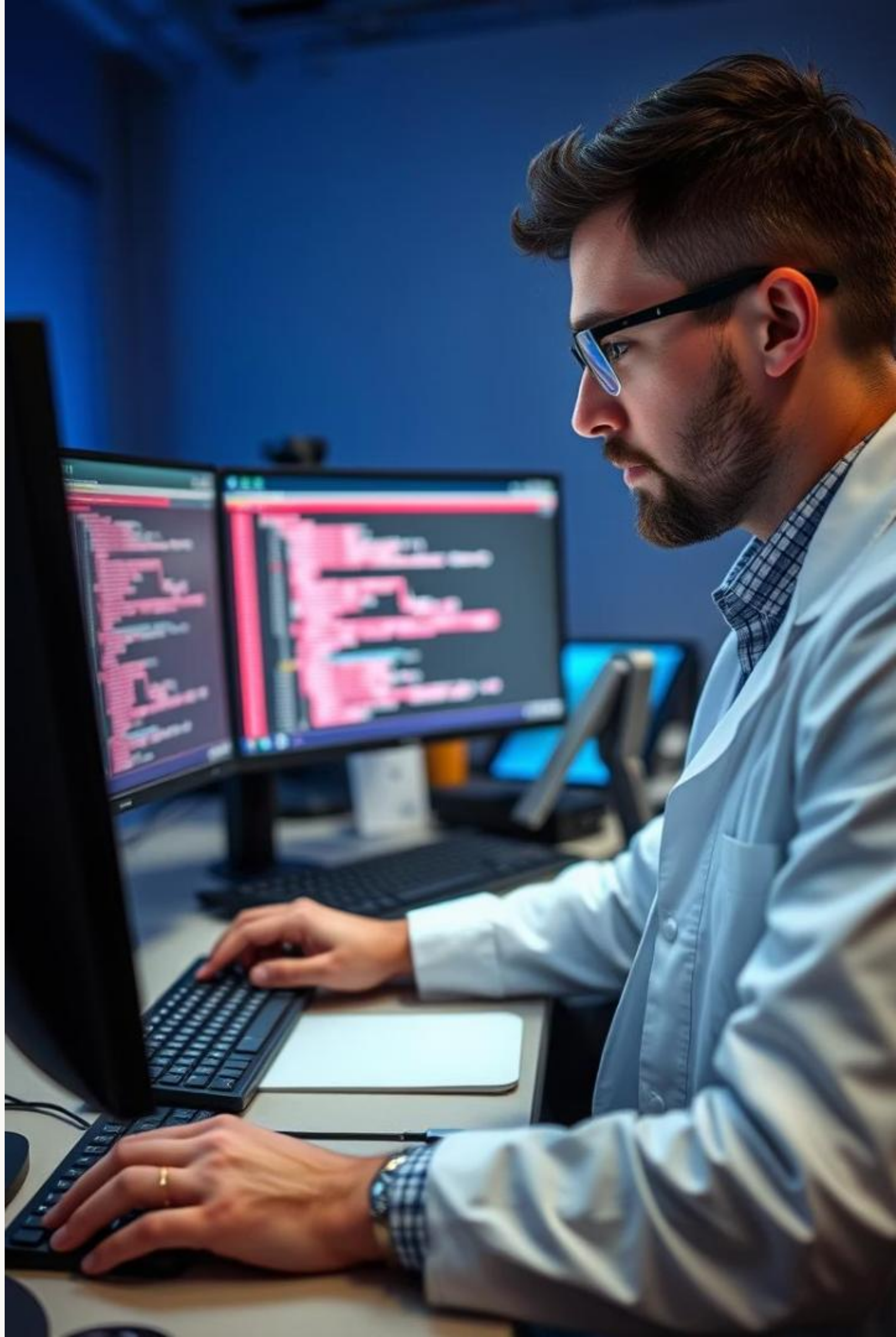
Action

- Dropped irrelevant columns
- Handled multicollinearity
- Applied GridSearchCV tuning

4

Result

- 20% RMSE reduction
- Stable, interpretable predictions
- Increased model trustworthiness



Thank You & Questions

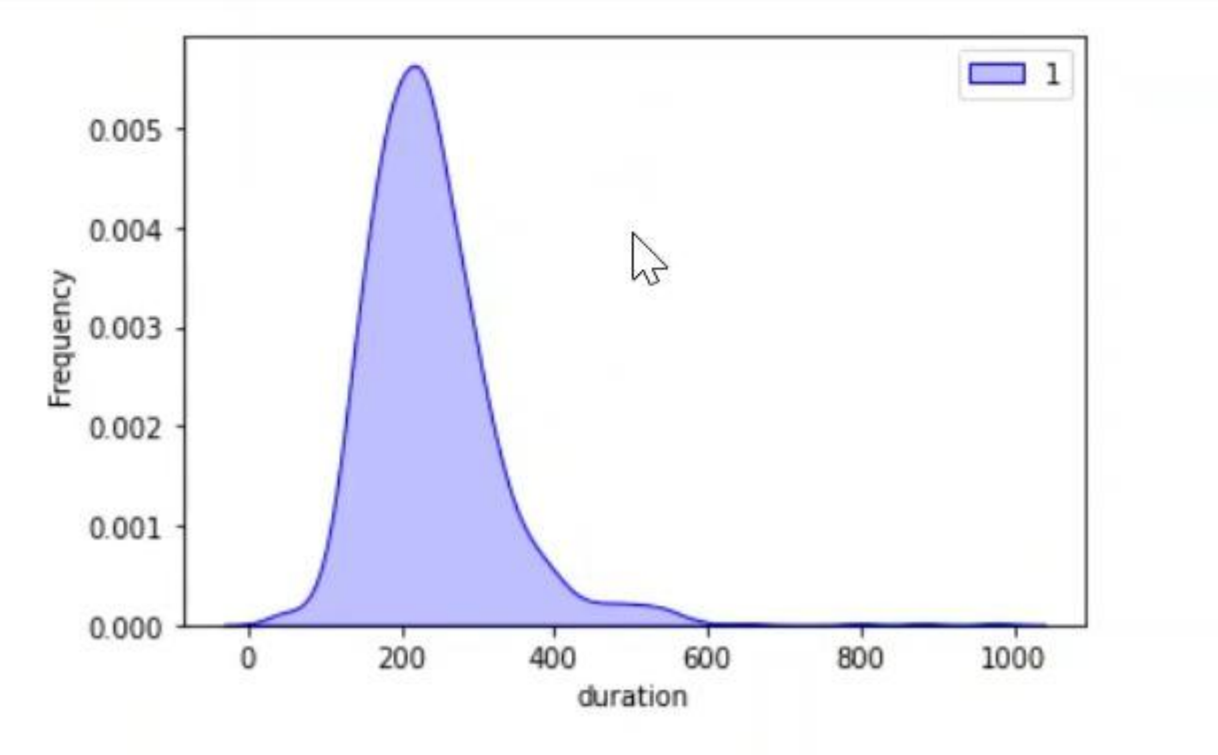
Appreciate your attention

Thank you for joining today

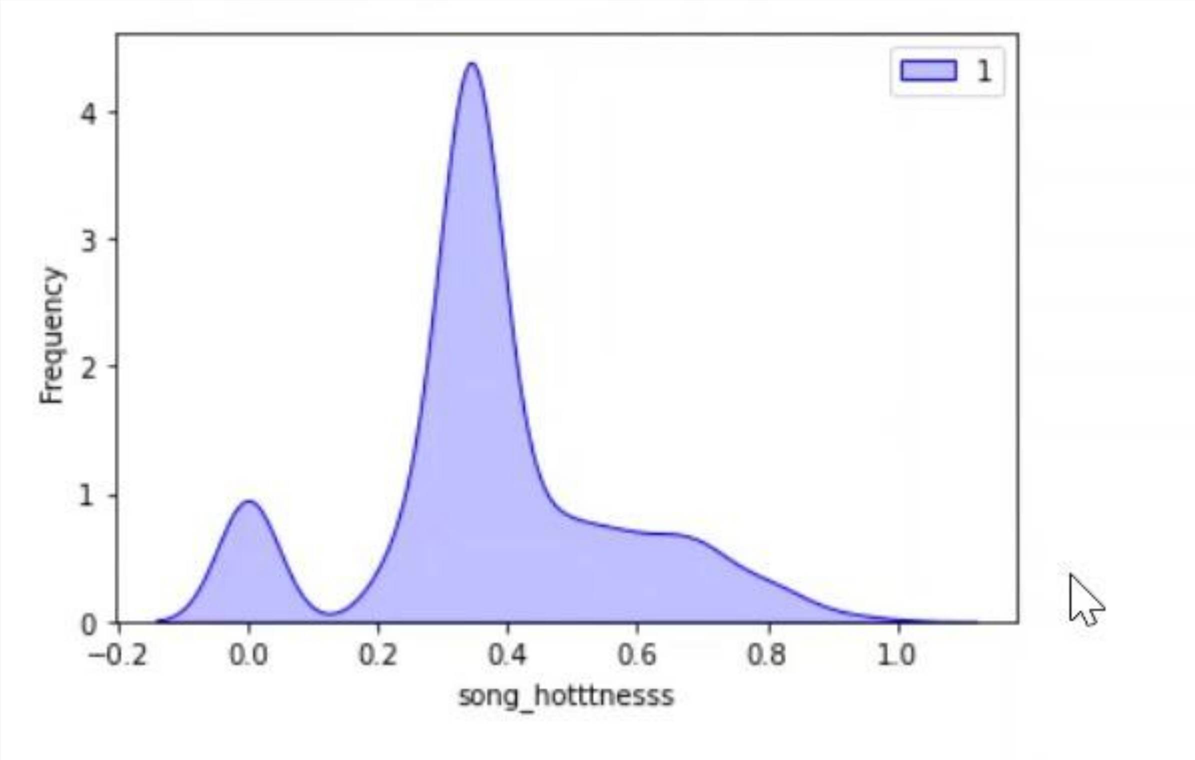
Open Q&A

Any questions or feedback?

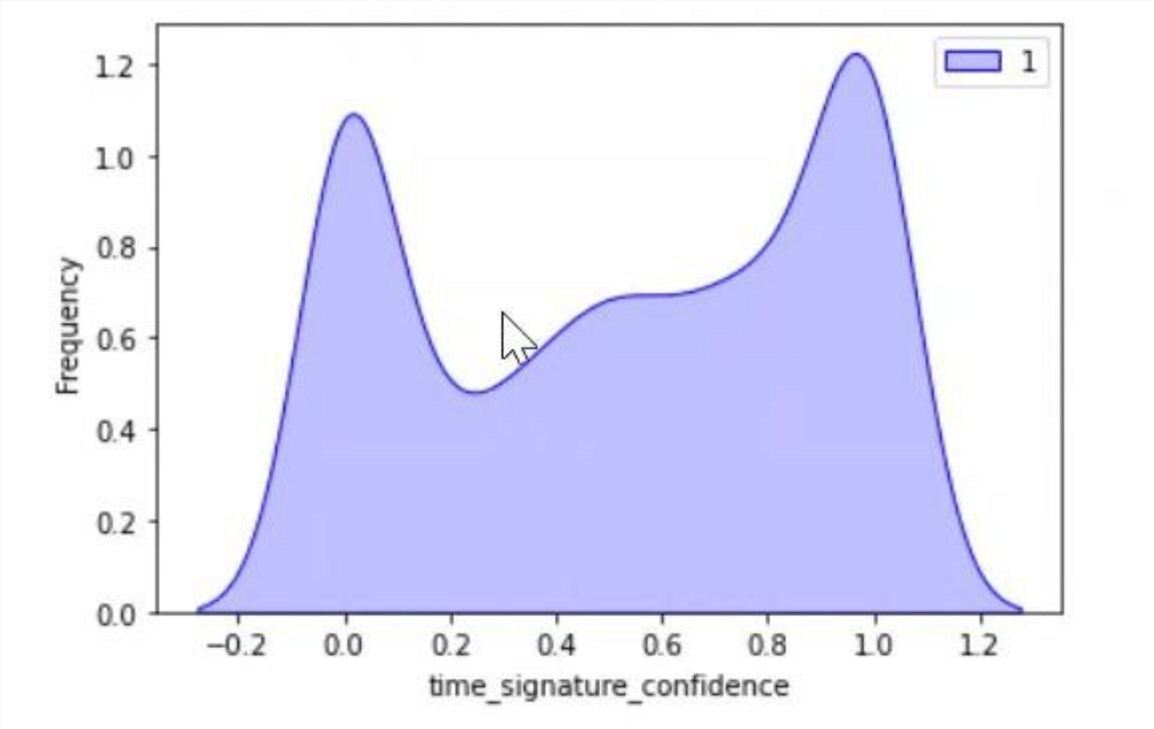
Appendix

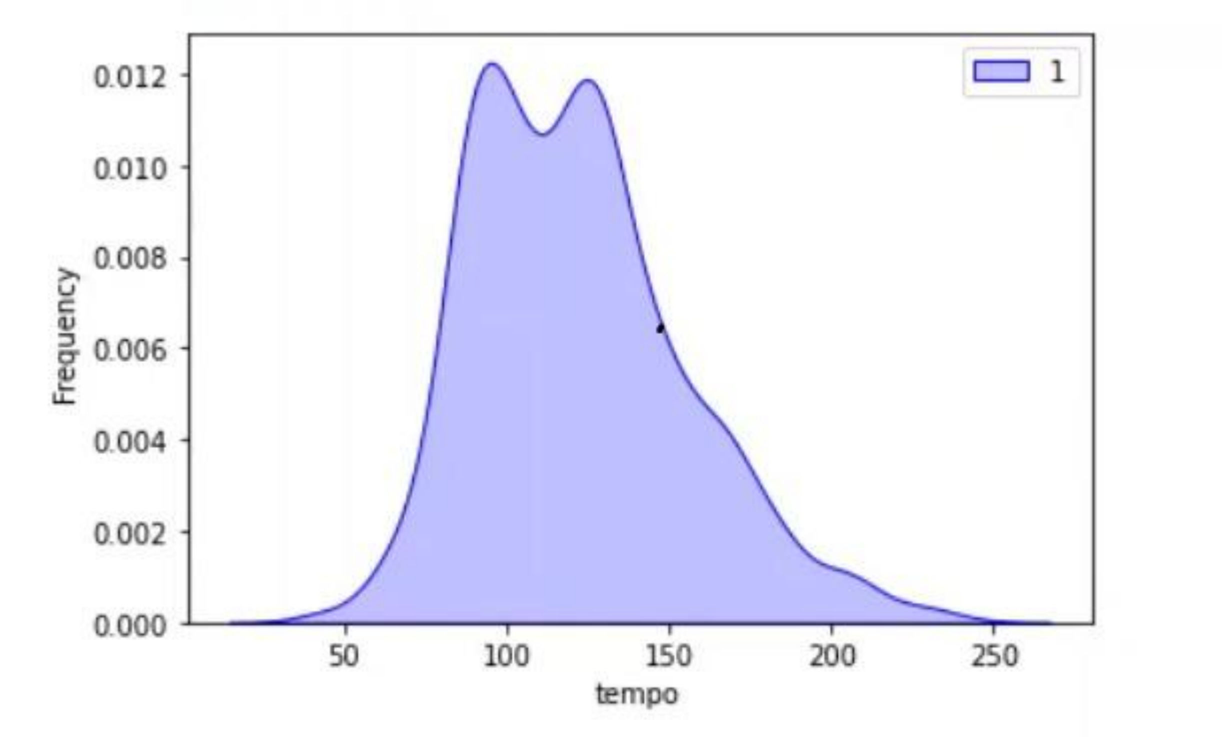


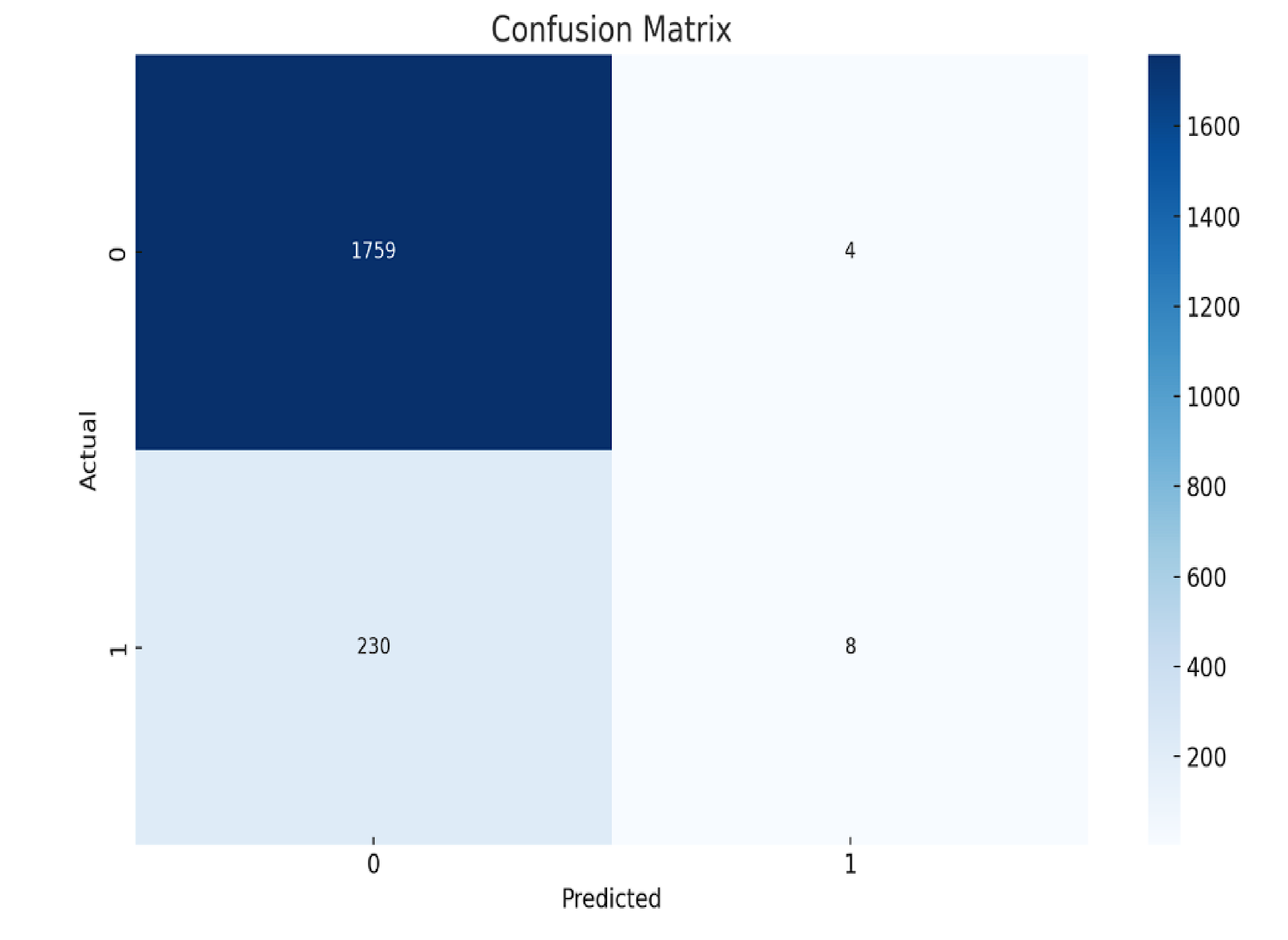
Frequency
0.005
0.004
0.003
0.002
0.001
0.000
0
200
400
600
800
1000
duration
▭ 1



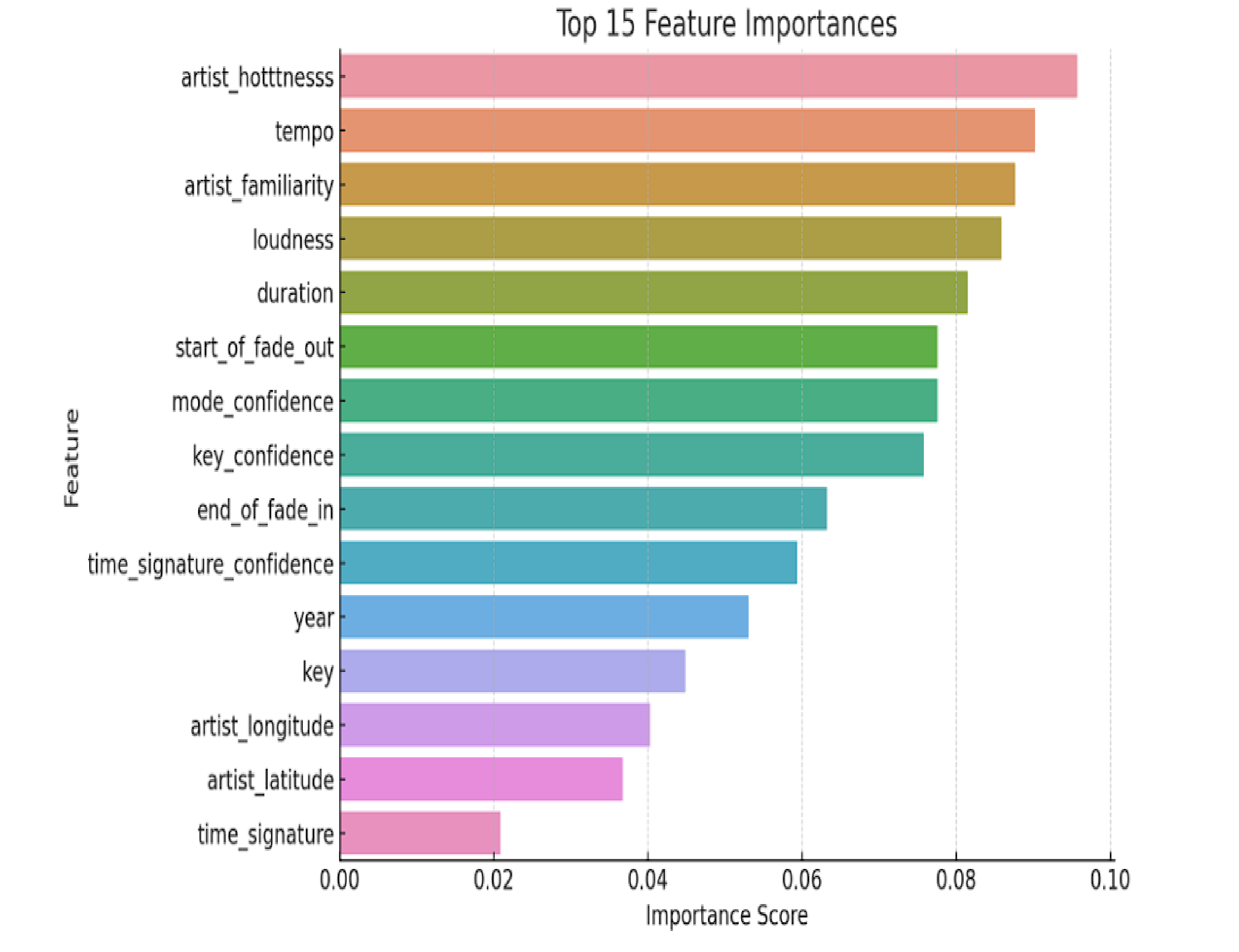
Frequency
4
3
2
1
0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
song_hotttnesss
▭ 1


1
Frequency
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
1.2
time_signature_confidence

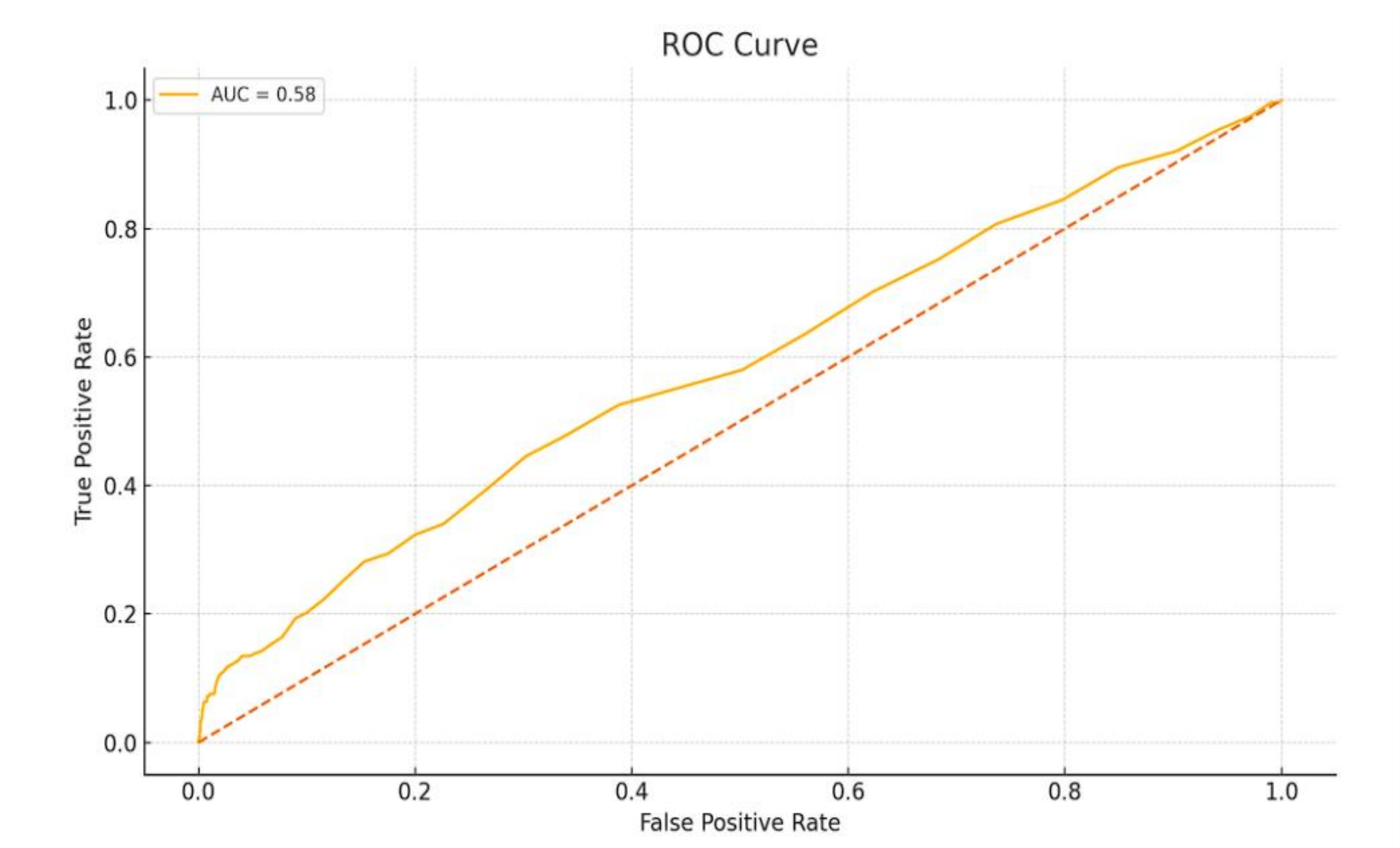

1
Frequency
0.012
0.010
0.008
0.006
0.004
0.002
0.000
50
100
150
200
250
tempo



Confusion Matrix



Top 15 Feature Importances



ROC Curve
AUC = 0.58
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0