

Movie Prophet



Team: Purrgramming

Names: Mao Yang, Jia Yu Pan, Rozanne Lim, Fan Yu

Motivation

Key Challenge for Film Industry:

Unpredictable nature of a movie's success (box office)

Questions:

- What are the underlying characteristics that transform a movie into a commercial triumph?
- Can we utilize these characteristics to predict success of a movie?

Targets:

- Audience: Investors
- Movies: Remakes and Originals



Previous Work:

[Zain Balfagih](#) analyzed factors like budget, genre, actors, and directors, then built Random Forest and XGBoost models to predict profitability. (Balfagih, Z. *IEEE 2024*)

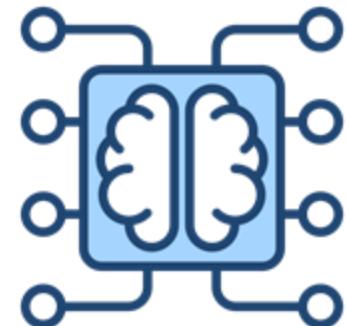
Beyond predicting if a movie will succeed, [Gaurang Velingkar](#) et al. used Random Forest model to specify the budget, runtime, and star power by expected box office input. (Velingkar, G, et al, *IEEE, 2022*)

Aim of the project

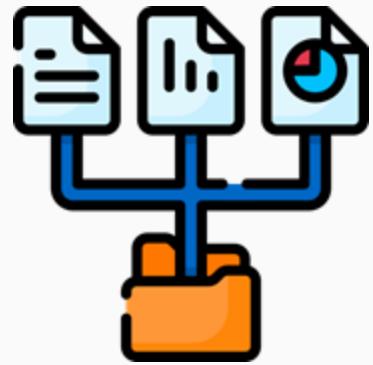
Create a comprehensive machine learning-based platform that not only predicts movie success but also prescribes optimal production strategies for targeted returns.

Outline

- Data Processing
- Data Analysis
- Machine Learning Models
- Dashboard Demo



Flowchart



Data collection



Data cleaning



Data analysis



Data dashboard

Data Collection



Data Sources

- **Wikipedia** - List of American Films released between the years 2000-2025 (November 16, 2025)
- **OMDB API** - Parameters of interest such as genre, director, cast, IMDB ratings, Metascore, and box office revenue \$\$
- **PriceR package** - inflation factors to convert box office revenue \$ across the years



Challenges

- Finding publicly available data on box office revenues and working with limited APIs
- APIs did not include a function to call for movies based on release year
- API limit to 1000 calls/day

The OMDB API is a RESTful web service to obtain movie information; all content and images on the site are contributed and maintained by our users.

If you find this service useful, please consider making a one-time donation or become a patron.

Data Cleaning & Processing

Functional Programming Paradigm

- Creating a function to scrape Wikipedia Lists of American Films released between the years 2000-2025
- Using `purrr` package to create functions to call the OMDB API
- Creating function to get inflation data

Parameters of interest

Genre (primary, secondary), box office revenue \$, runtime, MPA film rating, date of release, director, actors, IMDB rating, Metascore



Data Analysis - Dashboard

Challenges

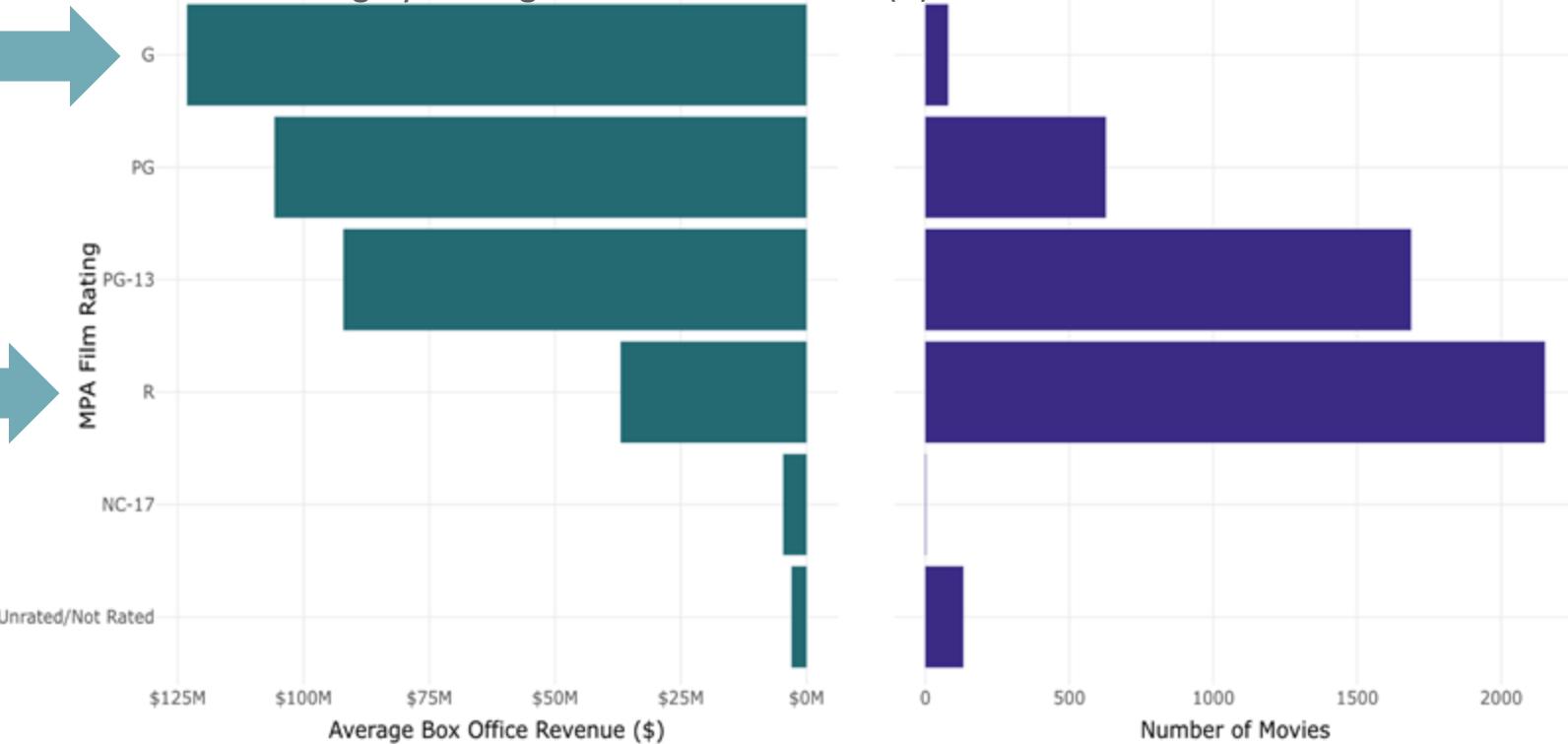
- Choice of visualization
- Incorporating interactive elements
- Formatting the dashboard
- User-friendly and aesthetics

Limitations

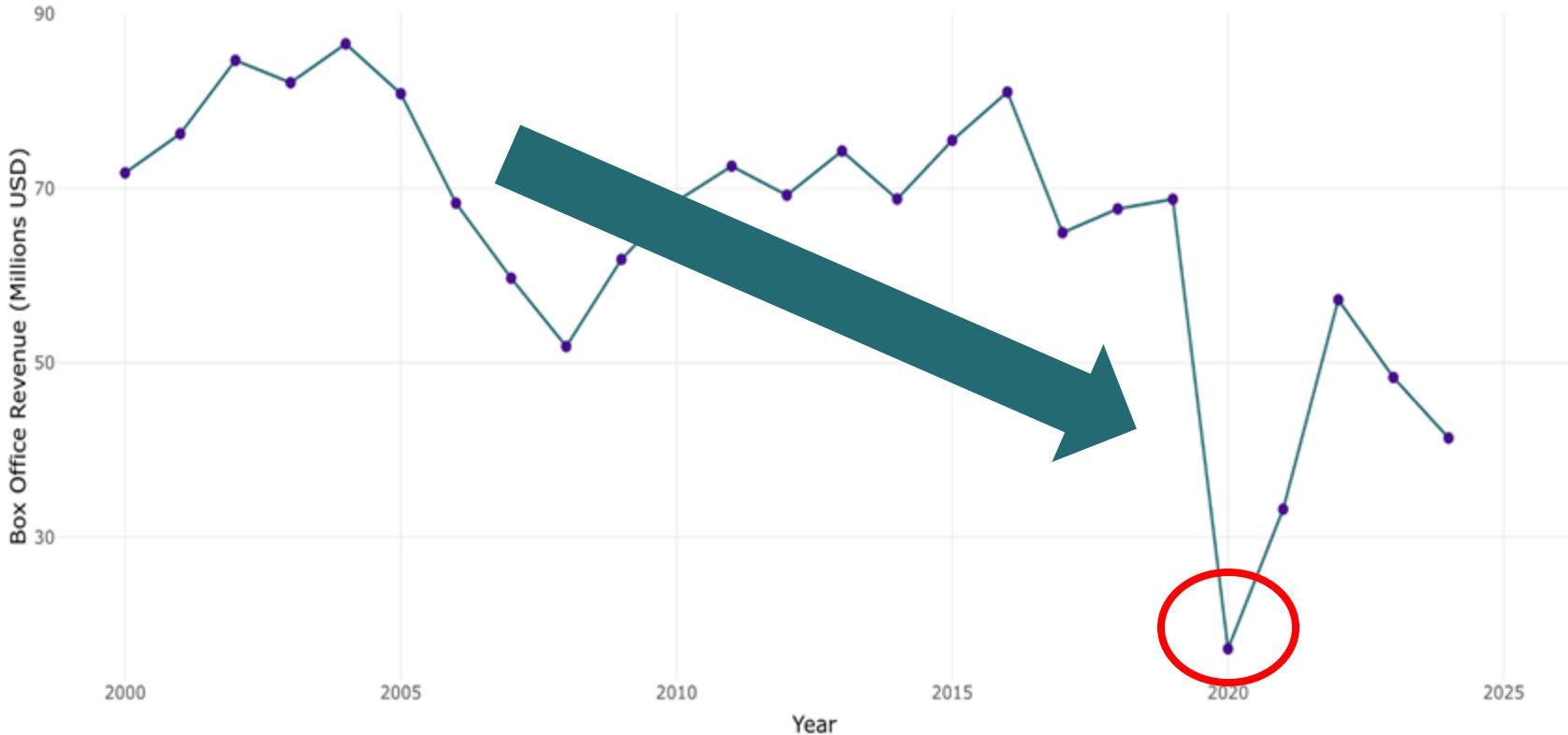
- Data limited to American films between 2000-2025
- Additional restrictions on data due to missing values for box office revenue, or movie was not found in the OMDB database



MPA Film Rating by Average Box Office Revenue (\$) and number of movies



Box Office Revenue (Millions USD) from 2000 to 2025





Data Analysis - Machine Learning

Goal:

- Predict box office based on movie attributes
- Compare several ML models
- Build up “What-if” simulation and AI recipe generator



Data preparation &
feature engineering

Train multiple ML
models

Evaluate using MSE
& R^2

Select best-performing
model

Feature Engineering



- 12 34 **Numeric transformations:** rating², runtime²
- 📅 **Time features:** years_from_2000, summer, holiday
- 🎭 **Genre encoding:** Action, Drama, Sci-Fi, etc.
- 🔞 **MPAA rating encoding :** PG-13, R, PG
- 🎬 **Movie length classification:** long/short movie

Machine Learning Models

Random Forest



◆ What it is

- An ensemble model that combines **many decision trees**
- Each tree learns different patterns; forest votes for final prediction

◆ How it works

- Uses **bagging** (bootstrap sampling)
- Random feature selection at splits
- Aggregates results:
Prediction = average of all trees

Support Vector Machine (SVM)



◆ What it is

- A model that finds the **best boundary** (hyperplane) separating data
- For regression, SVM fits a tube that minimizes errors beyond a margin

◆ How it works

- Maximizes the margin between data and prediction boundary
- Can use **kernels** to model nonlinear relationships

Machine Learning Models



Generalized Additive Model (GAM)

◆ What it is

- A flexible regression model that allows **non-linear relationships** between features and target
- Extends linear regression by adding **smooth functions** (splines)

◆ How it works

- Predicts outcome using:
$$Y = \beta_0 + f_1(x_1) + f_2(x_2) + \dots + f_n(x_n)$$
- Each $f(x)$ is a smooth curve learned from data
- Handles patterns that are **non-linear** and **complex**



Linear Regression

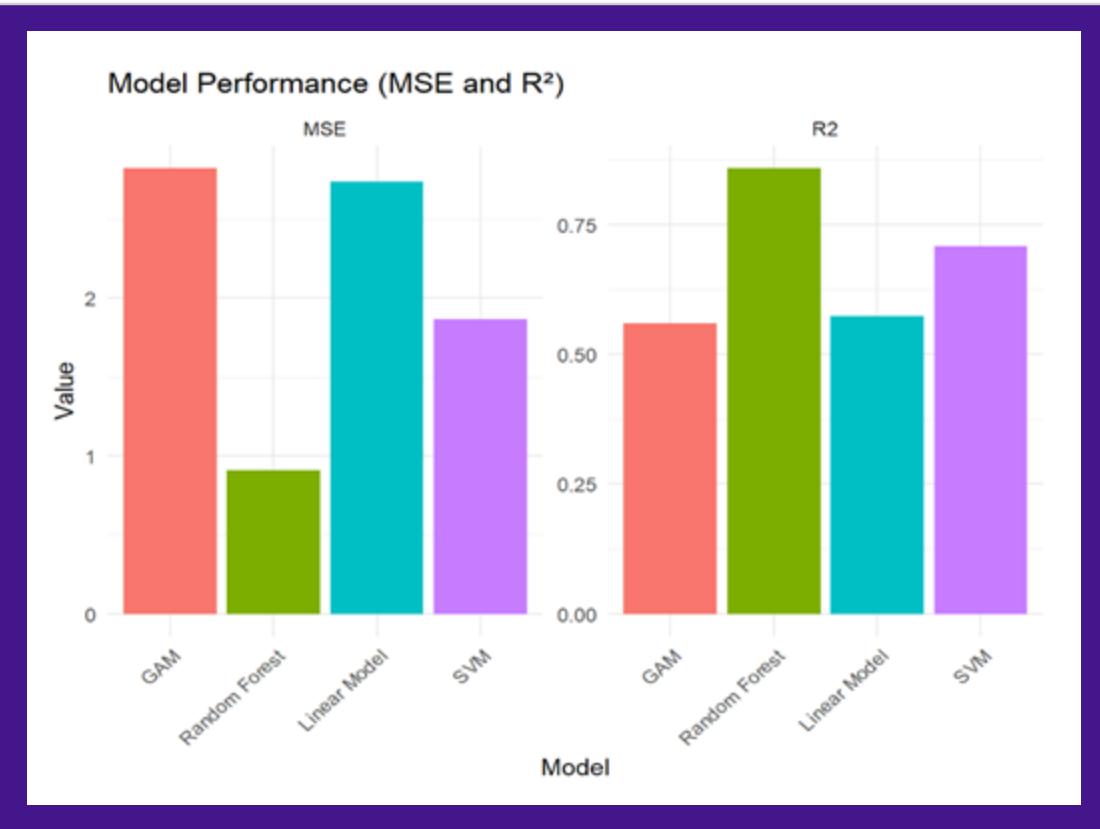
◆ What it is

- A simple baseline model that assumes a **linear** relationship between predictors and box office revenue

◆ How it works

- Fits a straight line that minimizes squared errors
$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots$$

Performance Evaluation



Random Forest:

- Lowest MSE
- Highest R²
- Best performer



SVM:

- Better performer

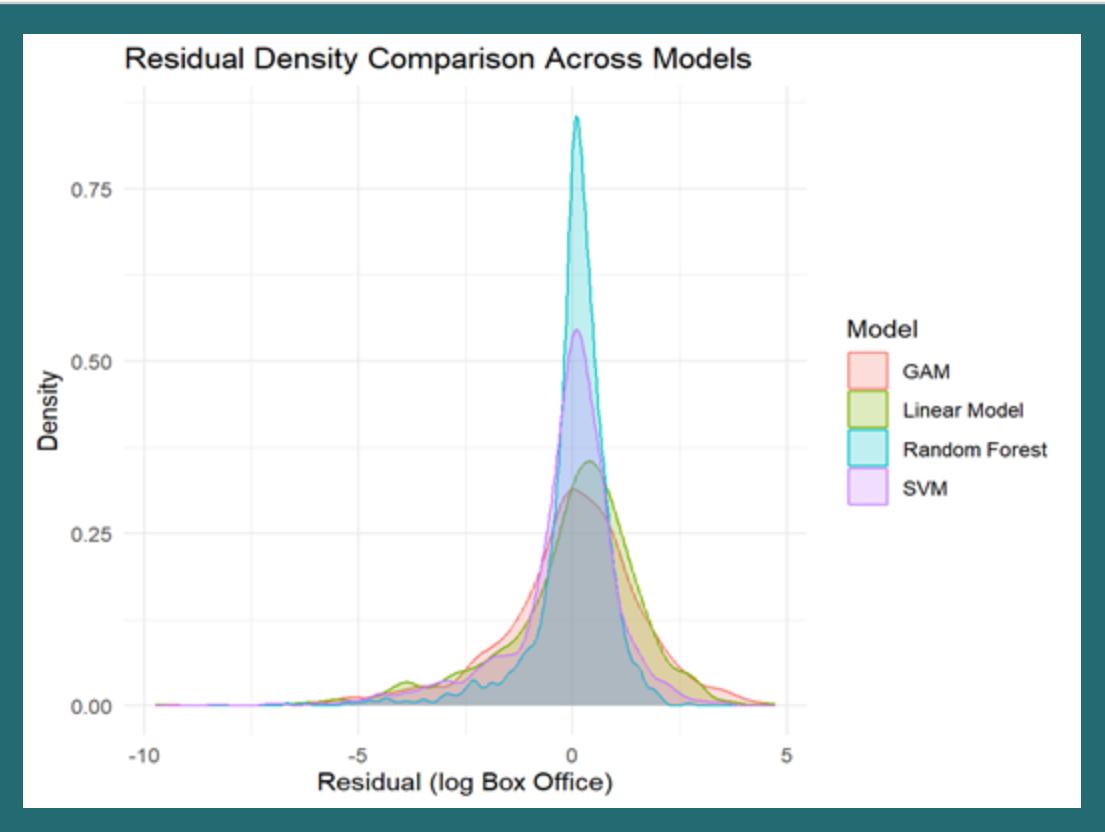


GAM & LM:

- Higher error
- Lower R²



Performance Evaluation



Random Forest:

- Most concentrated residues

AI Generator Recipe

AI Recipe Generator — Core Idea

Uses **text-generation AI models** to recommend movie “recipes” based on investment goals, optimization strategy, and creativity settings.

Model selection:

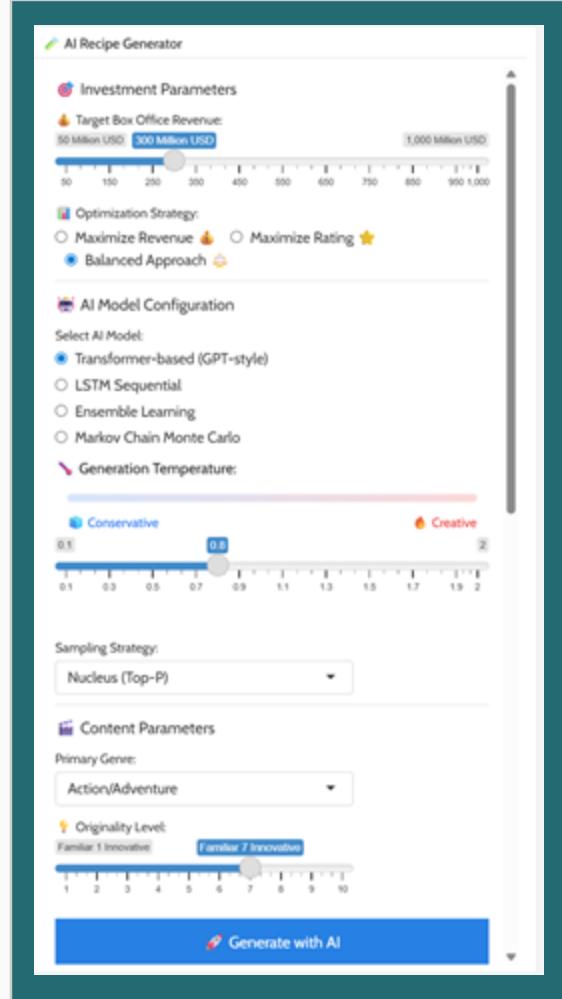
- **Transformer-based (GPT-style)** → best for creativity + coherent storyline suggestions
- **LSTM Sequential** → smoother sequential patterns, more conservative
- **Ensemble Learning** → blends multiple models for balanced output
- **Markov Chain Monte Carlo** → simple statistical pattern generator

Generation temperature: Controls how *creative* or *predictable* the model behaves

Optimization Strategy: The selected strategy shapes how the model evaluates content trade-offs.(Revenue or Rating)

Sampling Strategy: Defines how the AI picks words or ideas

Content Parameter Constraints: User-selected constraints guide AI (genre and originality level)



Machine Learning

Challenge

- Limited dataset size for ML models
- High variance in movie box office
- Risk of overfitting
- Non-linear interactions hard to visualize

What we learn

- Handles non-linear relationships
- Importance of Data Quality & Preprocessing
- Value of Model Evaluation Techniques
- Scenario Simulation & Model Integration

Dashboard Demo

Thank you!
Questions?
