# Movie Revenue Prediction

Movie Revenue Prediction


Pham Quang Vinh
20215256

Ha Hieu Minh
20215223

Dang Huu Tuan Minh
20210609


GROUP 11

# Work distribution

- **Hà Hiểu Minh:** Data Gathering (tmdb), Data cleaning, EDA, Catboost pipeline, Report, slide
- **Dang Huu Tuan Minh:** Data Crawl, Random Forest Model Train, Demo, Report, Slide
- **Pham Quang Vinh:** Data Gathering (Box office mojo), Ridge Regression pipeline, Report, slide

# Problem statement

The film industry is high-risk, and producers need better tools to estimate financial success before a movie is released, to maximize financial gains.


**Goal:**

- Analyze existing movie data
- Build a model to predict revenue based on movie’s features

# Solution in ML terms

This is a supervised **regression** problem with the target criteria being **revenue**

Data requirements:

## Target variable

**Revenue** must be a continuous numerical value, with no missing value

## Feature variables

All input features must be available before the movie's release date

*eg. Movie rating, number of votes, awards aren’t allowed*

# Data collection

We have gathered movie’s metadata using the HTTP requests from The Movie Database (TMDB), this is our main data source. But TMDB is biased towards western movies and is missing many revenue and budget data.

So we created a missing list for movies without budget or revenue so we could go on looking for it on another website - Box Office Mojo.

# Key features

We identified the following key features based on their predictive relevance to box office revenue and availability in our dataset:

| Feature | Description |
|---|---|
| Budget | The total cost of production (in USD). |
| Release Date | Release date of the movie (year, month, day of week). |
| Runtime | The duration of the movie in minutes. |
| Genres | Categorical labels (e.g., Action, Horror, Sci-Fi). |
| Original Language | The primary language of the movie. |
| Production Companies | Studios involved in production. |
| Production Countries | Countries where the movie was produced. |
| Director | The director of the movie. |
| Director Score | Historical performance score of the director. |
| Actors | Main cast members in the movie. |
| Actor Scores | Historical performance scores of actors. |

Table 1: Summary of key features used in the predictive model.

# Exploratory Data Analysis

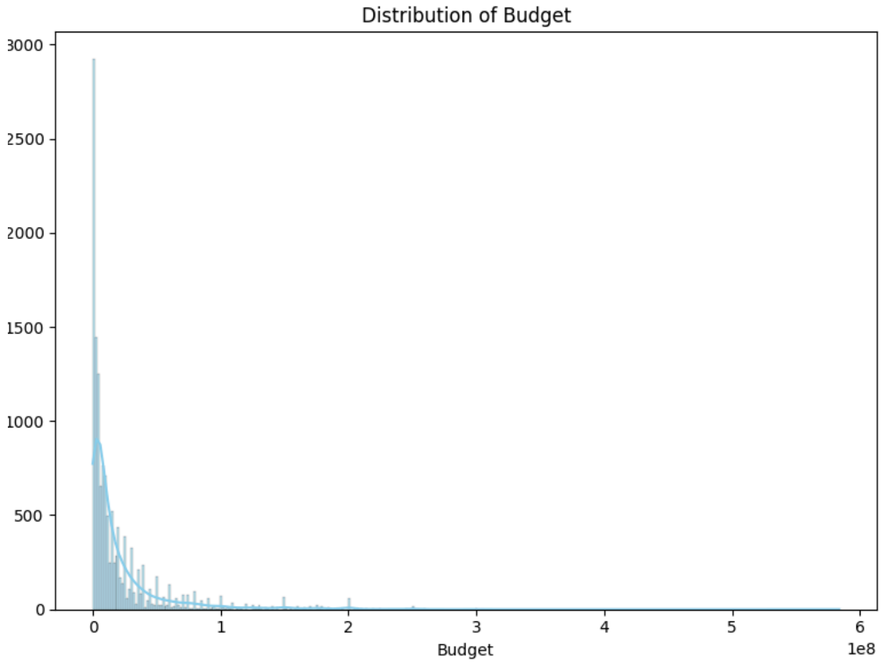
# Missing target variable investigation

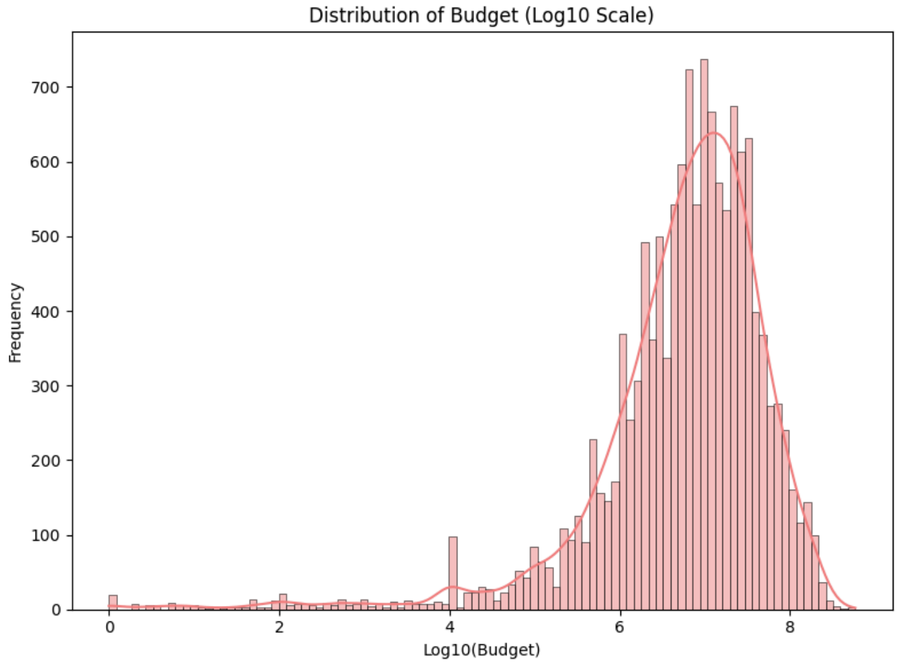
Our initial dataset includes 191,614 entries and 14 primary features, with approximately 75% of them missing revenue data, through some visualization technique, we conclude that this missingess is not at random and has the following pattern:

- Non-theatrical release movies
- Older movies, especially before the year 2000
- Lower budget, runtime and cast score on average

**Conclusion: Our model will inherit selection bias based on these patterns**
After removing these data points, we are left with roughly 42000 valid datas

# Feature distribution analysis: Budget

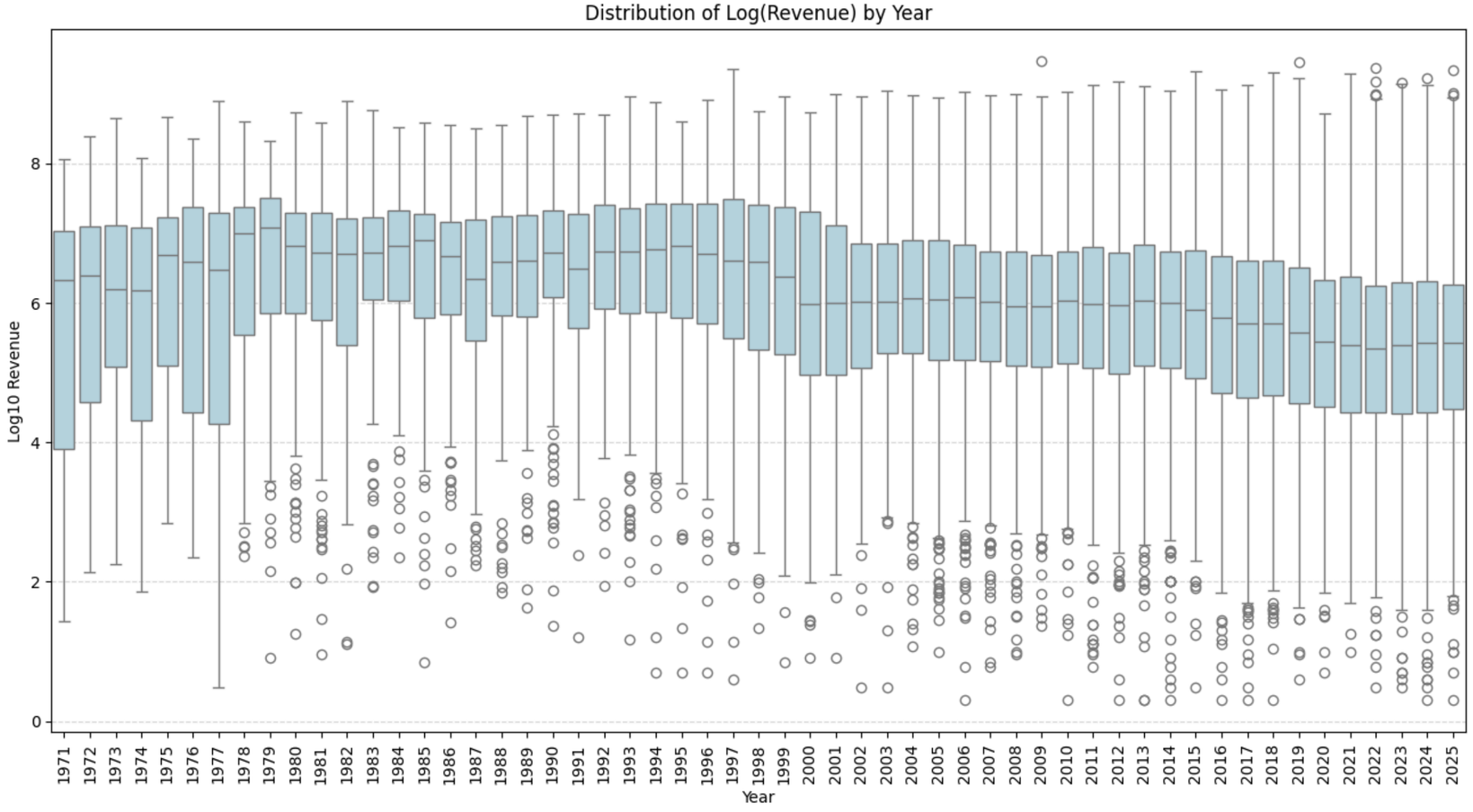



- Missing budget in 68% of our remaining data
- Left-skewed even after log transform
- Spike in number in the $1 point and $10000

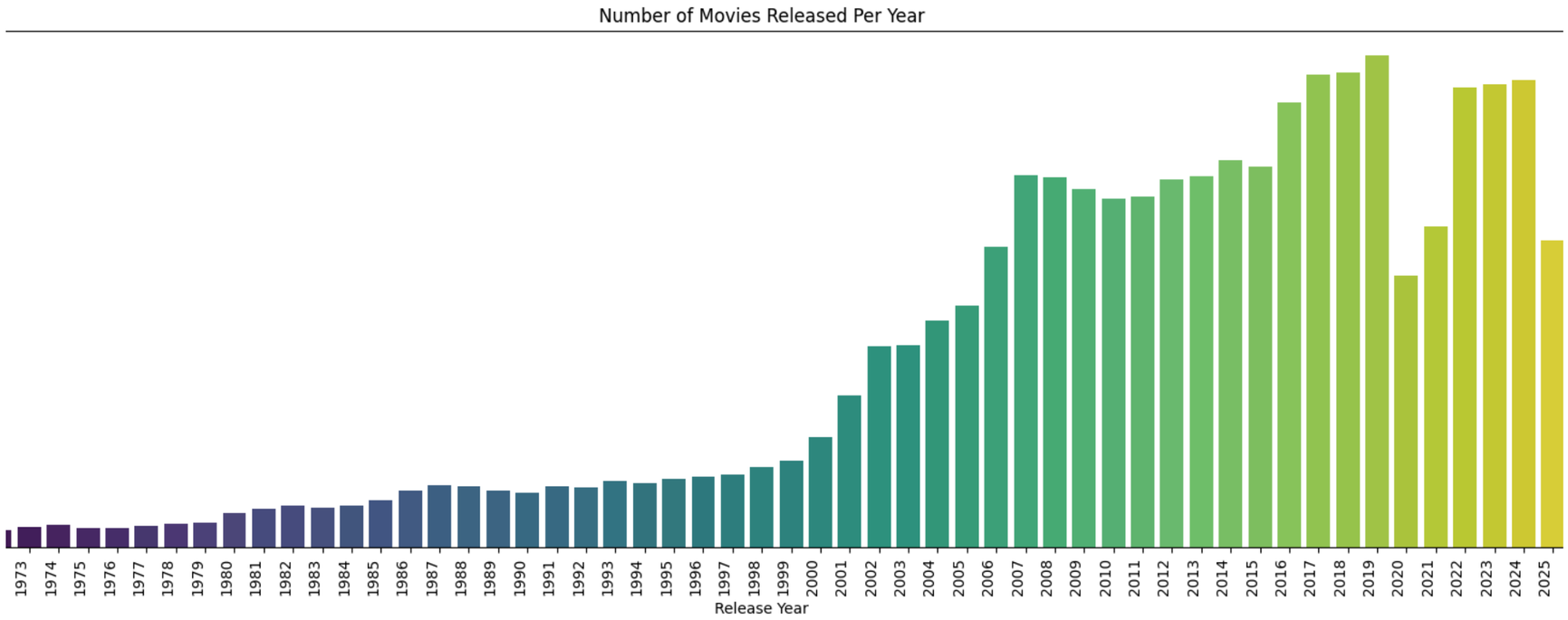
→ Noise data and incorrect placeholder data

# Feature relation analysis: Year vs revenue




We can see a downward trend in the median of revenue of movies for recent years
→ Decline in theatre attendance?
→ Saturated film market?

# Feature distribution analysis: Year




Number of recorded movie data increases sharply in recent years, only decline in the year 2020-2021 due to the pandemic

→ Our data likely have survivorship bias, only good movie gets recorded in details

# Addition characteristics

- Categorical features:  original_language, main_production_country, production_companies, cast, and director exhibit extremely sparse distributions and will need to be handled accordingly
- Budget is the most important with correlation coefficient of 0.73, while other numeric columns also have small correlation with revenue and will be useful for training
- For all categorical features, there is noticeable variance in median revenue across different categories. These will all be helpful for training

# Evaluation and Test split

For evaluation, we chose these parameters

- RMSLE (Primary metric)
- R^2
- MAE (For interpretability)

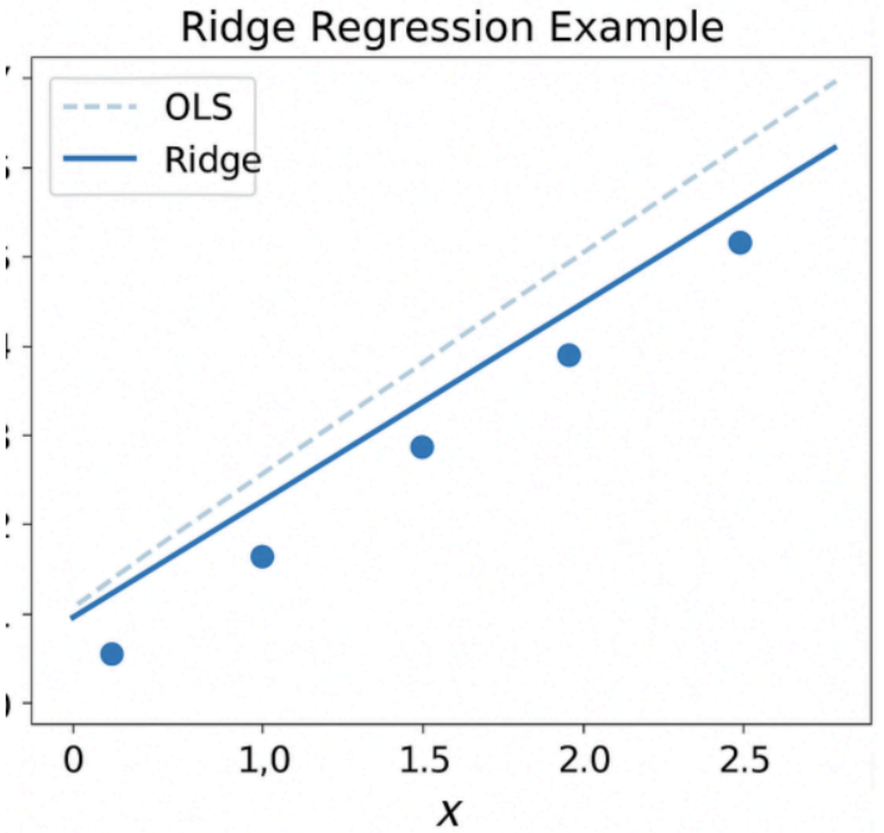
For training and testing split, we used the following method:

- Data is sorted by release_date
- **Training Set**: The oldest 70% of the data
- **Validation and Test set:** Random 50/50 split using the most recent (30%) data, in order to evaluate the model on unseen data

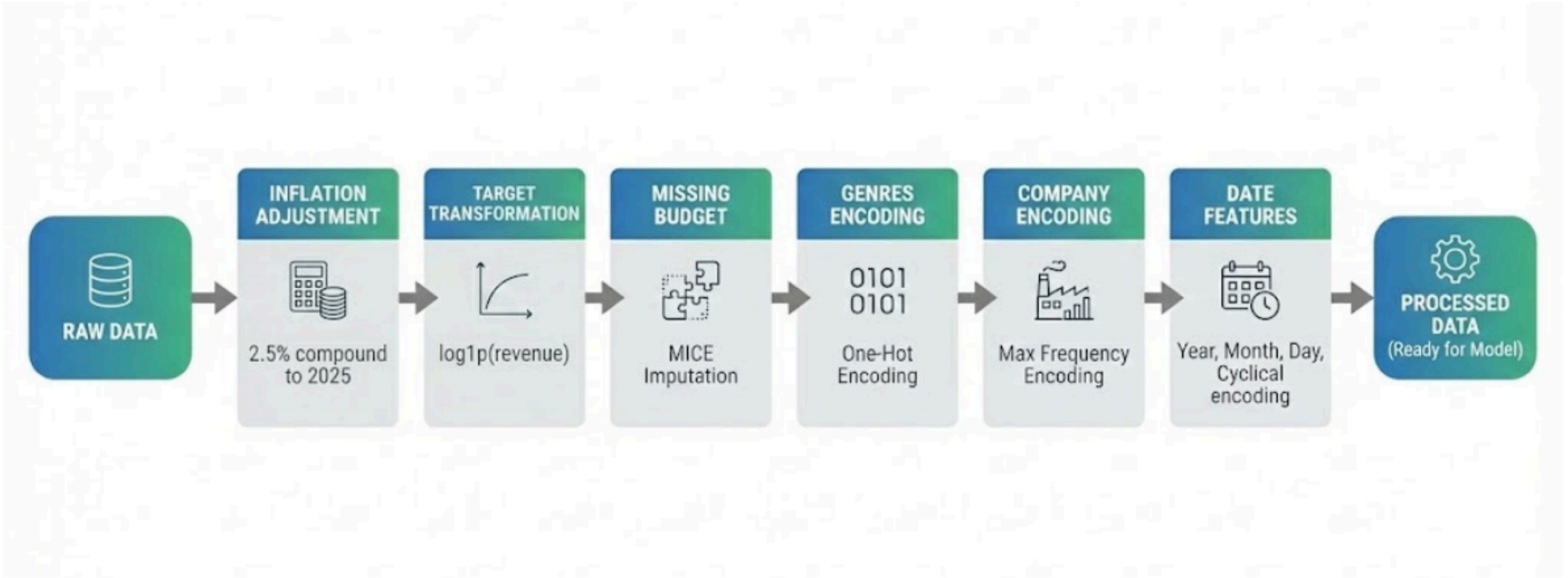
# Model Pipelines

# Ridge regression

- Help prevent overfitting, increase stability.
- Can’t deal with non-numeric features



# Data Processing



# Training model

- Using Standard scaler, alpha = 100

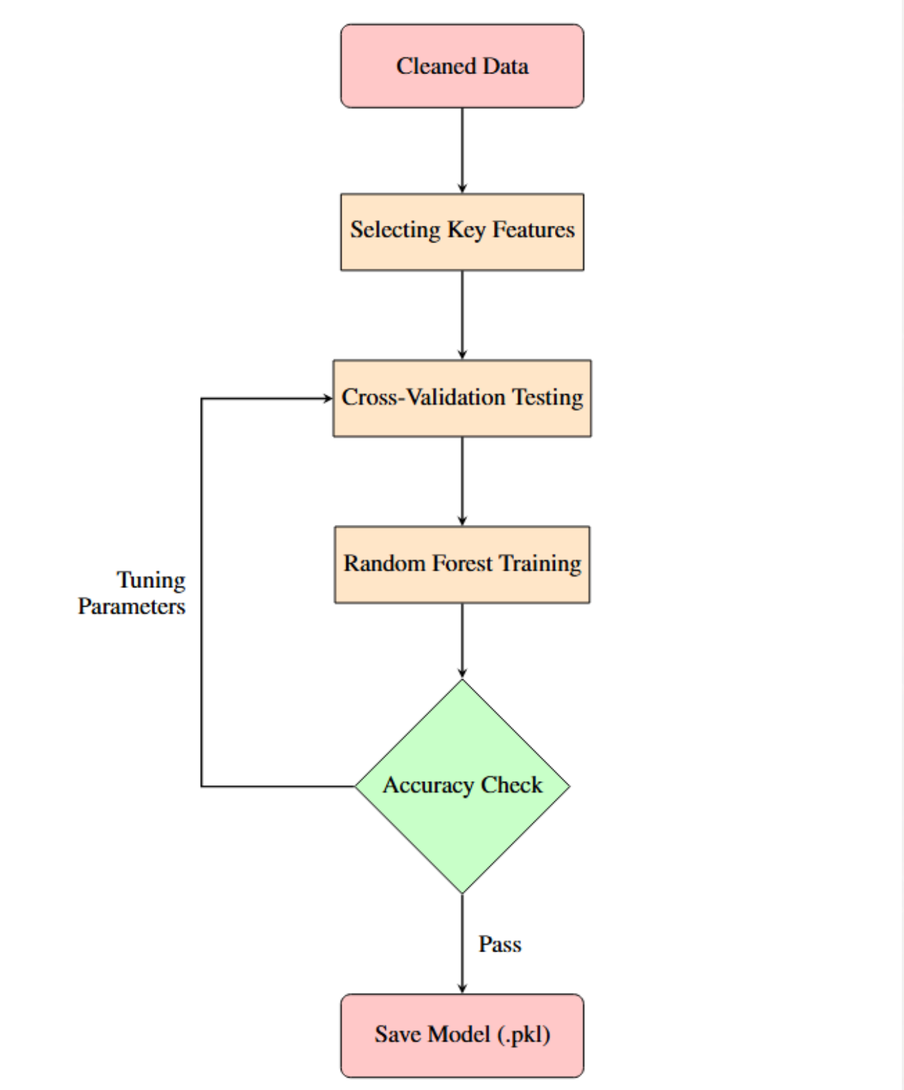
Table 4: Ridge Regression Performance on Test Set

| Metric | Value |
|---|---|
| RMSLE | 2.63 |
| $R^2$ | 0.27 |
| MAE (USD) | $43,343,120 |

→ The model's R2 = 0.27 indicates limited explanatory power, suggesting that linear relationships are insufficient to capture complex revenue patterns

# Random Forest Model

## Model Architecture



# Random Forest Model

## Feature Selection

Identify which factors most influence movie revenue to make the model faster and more accurate.

## Cross-Validation

Test the model many different times on various parts of the data to ensures the results are consistent and not just a lucky guess.

## Exporting

Saved as a .pkl file, allowing the FastAPI server to load it instantly for real-time predictions.

## Training the Forest

Averaging their results, the model becomes much more reliable

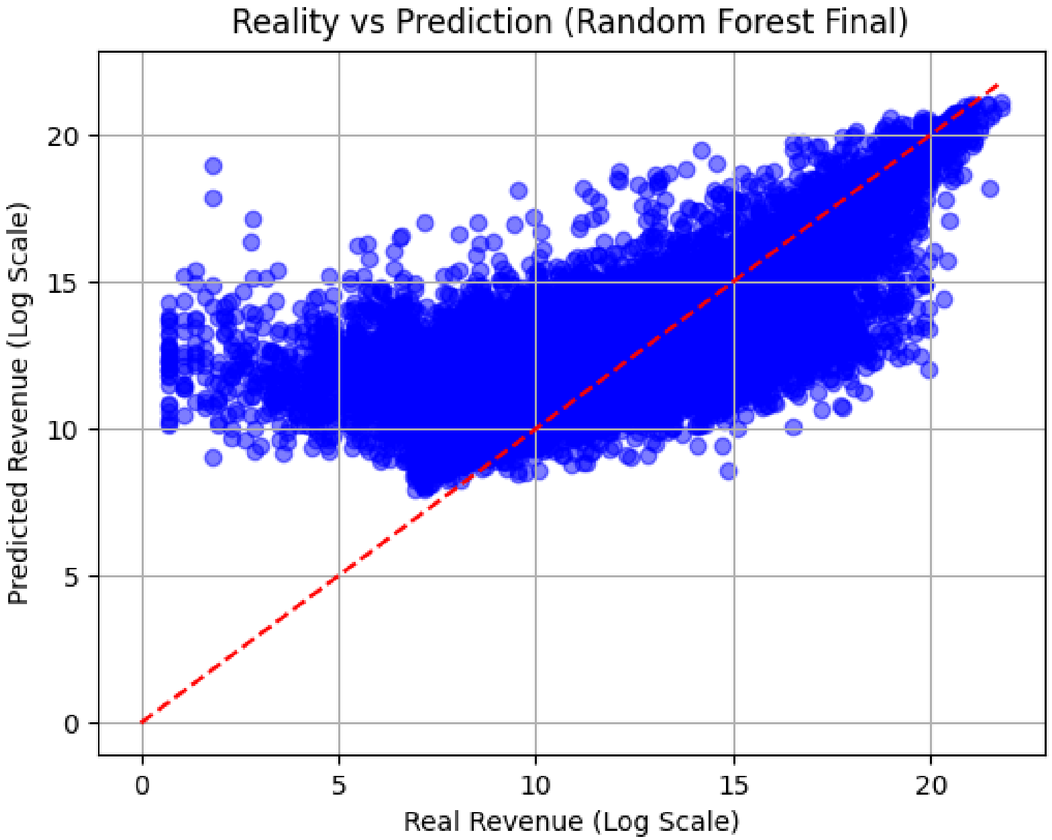
## Optimization

Fine-tune settings like tree depth to ensure the model captures real patterns without being distracted by random noise

# Random Forest Model

## Model Performance

| Model Stage | RMSLE | $R^2$ | MAE $ |
|---|---|---|---|
| Baseline Model | 2.68 | 0.27 | 11,527,045 |
| Final RF Pipeline | 2.42 | 0.38 | 13,120,450 |
| Testing on testset | 2.38 | 0.42 | 12,845,920 |




The R2 score of 0.42 confirms that the model captures a significant portion of the market variance, providing a stable foundation for the prediction engine.

# Catboost

**Data preprocessing:**

- MICE (Multivariate Imputation by Chained Equations) to impute missing budget
- Log transform on revenue and budget
- One hot encoding on Genres
- Frequency encoding on production companies
- Target encoding and grouping up low density categories for other categorical features
- Extracting time related data from release date
- Adding some interaction ratios such as budget/year, budget/runtime,...

To ensure there is no data leakage, all preprocessing transformations were fitted exclusively on the training set and subsequently applied to the validation set. Unseen data in the validation set is treated as ‘Missing value’ or ‘Others’.

# Catboost

**Intercept shifting:** Adding the difference between the mean log-revenue of the validation set and that of the training set to the prediction result of the model

**→ Pipeline result:**

| Model Stage | RMSLE | R2 | MAE $ |
|---|---|---|---|
| Baseline | 2.68 | 0.27 | 11,527,045 |
| Final model | 2.23 | 0.49 | 11,374,957 |
| Testing on test set | 2.19 | 0.49 | 10,215,837 |

# Conclusion

# Result and comparison

Catboost model performed the best among the others with RMSLE of 2.19, which we will be using for our demo
We can compare our result with a kaggle competition in the past that used similar TMDB data:

| Source | RMSLE |
|---|---|
| Standard Kaggle Competition Winner | 1.7 |
| Our Catboost model | 2.19 |

Note: the competition used data like vote count, rating and popularity,... which we excluded from our training due to data leakage

# Conclusion

Despite the relative success of the model compared to benchmarks, we must critically evaluate its utility in a real-world business context:

- An RMSLE of 2.19, while competitive in a data science challenge, translates to a margin of error that is currently too high for reliable financial planning
- Data collected from just these 2 sources proves to be unreliable and prone to errors, however the movie industries tends to have inconsistent data as a whole
- Metadata alone just isnt enough to evaluate a movie’s performance

# Future works

- Given the errors encountered using these tmdb as primary metadata source, future work should cross-referencing entries metadata with more popular and less western bias movie databases
- Our current model treats movies as a standalone entity, however it using a new feature to flag for sequels or part of a franchise is important
- Since metadata alone is not enough, we can try to use NLP to analyze full movie script or detailed plot summaries

# Demo

# The End

THANK YOU FOR LISTENING

GROUP 11