Final Project Report – BAN 612

Name : Samadrita Roy Chowdhury

Executive summary

Studios now premiere films across two very different channels: a box-office run or a direct drop onto a streaming service. Using a single, cleaned dataset that stitches TMDb detail with a streaming-platform catalog, I built two models and a set of diagnostics to answer a practical question: **what moves the needle in each channel?**

- **In theaters**, money and momentum matter most. Bigger budgets, longer runtimes, and stronger audience engagement (votes/popularity) are the clearest, repeatable drivers of higher revenue. Winter releases edge out other seasons in my sample.
- **On streaming**, "quality reception" is more about **engagement and fit** than platform brand. Titles with more audience activity (votes), mid-length runtimes, and certain genres (Documentary, Drama, some Action) are more likely to land in the top critic tier; TV-movie style content and holiday drops underperform.

Models were trained on earlier years and evaluated on held-out years to mimic the real forecasting problem. Results are competitive and - more importantly - explainable: elastic net offers clean elasticities for the theatrical side; gradient boosting plus SHAP shows non-linear patterns (e.g., diminishing returns to budget, a runtime "sweet spot") and which features push predictions up or down.

Problem setup

Two tasks mirror the two channels:

1. **Theatrical (regression):** predict log(revenue) from film attributes.
2. **Streaming (classification):** predict whether a title lands in the **top quartile** of Rotten Tomatoes scores within the train period.

The motivation is practical: greenlighting and scheduling decisions hinge on what reliably drives revenue or reception - **and whether the same knobs matter in both worlds**.

Data collection & Preparation

Sources and join strategy
I combined two public datasets that cover the two worlds we care about:

- **TMDb (2018–2024)**: title, release date, runtime, genres, budget, revenue, vote metrics, popularity.
- **Movies on Streaming Platforms (2018–2021)**: title, year, platform flags (Netflix/Hulu/Prime/Disney+), Rotten Tomatoes score.

Normalization & merge. Titles were lower-cased, punctuation/spacing collapsed, and subtitles after a colon removed for matching (e.g., *Dune: Part One* → *dune*). I joined on **normalized title + year (±1)** and kept a one-to-one row per title by preferring exact year matches and dropping ambiguous many-to-many collisions. I converted TMDb's "0" **budget/revenue** placeholders to **missing**—that's important to avoid teaching models that "0 dollars" is real money.

Labeling.

- **Theatrical** = has non-missing revenue (i.e., was released with box-office reporting).
- **Streaming** = any platform flag is 1 (revenue intentionally absent; we use RT score as the outcome proxy).

Sanity checks. I ran four quick QA passes before modeling:

1. **Date integrity** (parsable release_date, year aligns with parsed date).
2. **ID uniqueness** (one row per title/year after merge).
3. **Range checks** (runtime > 0; RT score in 0–100).
4. **Missingness map** to confirm where imputation will be needed (budget is the only big hole on the theatrical side).

Feature set built from EDA clues.

- Calendar: year, quarter, and a coarse **release_season** (Winter/Spring/Summer/Fall/Holiday).
- Content: a single main_genre as the dominant tag (keeps the design interpretable).
- Signals: log_budget, log_revenue, log_votes (from vote_count), log_popularity.
- For analysis (not as target): **ROI** where both budget and revenue exist, to reason about diminishing returns.

EDA (what the data said before modeling)

- **Cohorts & gaps.** Theatrical (2018–2024) has revenue for all films and budget missing ~⅓; Streaming (2018–2021) has platform flags and almost-complete RT scores (no revenue, by design).
- **Cleaning that matters.** TMDb's 0-budget/0-revenue → treated as **missing**. Titles normalized and merged on **title+year (±1)**. Added **season**, **year/quarter**, and a single **main_genre**.
- **Why logs.** Budget and revenue are super skewed; **log transforms** make both usable and keep big tentpoles from dominating.
- **Budget vs revenue.** On logs, strong diagonal (tight link), but visible spread at every budget → **timing/genre/attention still matter**.
- **Timing. Winter** and **Summer** sit higher on median revenue; **Holiday** is high-variance (few giants, many modest). Season is a useful, compact feature.
- **Runtime.** Most films cluster ~90–130 min; effect is mild and **non-linear** (sweet spot, then flattening).
- **Genres.** Drama/Comedy/Action dominate counts; Animation and Family have higher medians but wide variability; Documentaries under-index theatrically (stronger on streaming).
- **Streaming scores.** RT scores center ~40–60; platforms differ a bit (Hulu slightly higher in this slice) but **distributions overlap a lot**.
- **Engagement. log_votes** and **log_popularity** carry extra signal beyond budget (audience attention/word-of-mouth).
- **What we shipped as features.** log_budget, runtime, log_votes, log_popularity, plus categorical main_genre, release_season. Targets: **log(revenue)** (theatrical) and **top-quartile RT** (streaming, threshold set on train years to avoid leakage).

That's the essence of what we looked at, what we learned, and why those exact features made it into the models.

Methods & implementation

Framing and splits
I treated this as a **forward-looking** problem, so I split by time:

- **Theatrical regression**: train 2018–2022 → test 2023–2024, target = log(revenue).
- **Streaming classification**: train 2018–2020 → test 2021, target = **top-quartile RT** within *train only* (so the cutoff isn't contaminated by the test year).

Pipelines (reproducible and leakage-safe)
Built with scikit-learn:

- **Numeric**: SimpleImputer(strategy="median") → StandardScaler (for linear models; tree models don't need scaling).
- **Categorical**: OneHotEncoder(handle_unknown="ignore").
- **ColumnTransformer** glues them; one fit per **train** fold/year block.

Models and why

- **Theatrical — Elastic Net (primary).**
  Linear, regularized, and easy to read as elasticities on the log scale. I tuned alpha and l1_ratio with time-aware CV.
  - **Why not jump straight to XGBoost?** Because a strong linear baseline tells you whether the big drivers are really global (they are) and gives a stable, explainable yardstick.
- **Theatrical — Gradient Boosting (diagnostic).**
  A small-depth boosting model validates non-linearities and interactions suggested by EDA. I used **partial dependence** to visualize the shape of effects and **SHAP** to rank/interpret drivers.
- **Streaming — Gradient Boosting classifier (primary).**
  Reception is not linear in runtime or genre. GB (shallow trees, moderate learning rate) balances flexibility and stability. I evaluated with precision/recall/F1, **plus** the confusion matrix at 0.5 and noted that the threshold can be tuned if recall or precision is more valuable.

Guardrails against common pitfalls

- **Leakage control**: RT "top-quartile" is computed from the train years only; no ROI or post-release signals go into targets.
- **Robustness**: checked that conclusions hold when shifting the train/test boundary by a year; feature importances are stable (budget and engagement stay on top).
- **Class imbalance**: no reweighting needed - the base rate in 2021 is ~31%; I report precision/recall explicitly to keep us honest.

Insights

If I had to boil the whole project down to one sentence: **money gets you into the game, attention wins it.**

What reliably moves theatrical revenue

- **Budget matters—but with tapering returns.** On the log scale, more budget pushes revenue up, but the **partial-dependence curve flattens** at the high end. Past a point, another dollar buys less than the last one.
- **Audience attention travels with revenue.** Both **log_votes** and **log_popularity** show large, consistent SHAP impacts. When those are high, the model nudges revenue up even after accounting for budget - i.e., **buzz is not just a proxy for big spend**.
- **Release windows still matter.** Films opening in **Winter and Summer** sit higher on median revenue; **Holiday** is volatile (a few blockbusters plus many modest titles). If you don't have true four-quadrant reach, the holiday slot is a coin flip.
- **Genre is a second-order shaper.** We see small, steady bumps for **Animation** (family pull, long tails) and slightly negative adjustments for **Horror/Crime** in theaters. Genre by itself doesn't make or break a film, but it shifts the baseline.

What reliably moves streaming reception

- **Engagement dominates. log_votes** is the top driver of "top-quartile" Rotten Tomatoes reception on platforms; **runtime** contributes with a gentle optimum (too short looks slight; too long can drag).
- **Content beats platform.** The platform flags (Netflix/Hulu/Prime/Disney+) matter **far less than genre/runtimes/engagement**; distributions overlap heavily across services.
- **Documentary/Drama quietly over-index.** They're not the biggest genres by count, but they lift the odds of strong critic reception in the streaming slice.

How good are the models (held-out years)

- **Theatrical (regression).** The Elastic Net explains a solid share of variance with **$R^2 \approx 0.61$** and **MAE ≈ 2.56 log units** on 2023–2024. It's a clean benchmark; we then used Gradient Boosting mainly to **expose non-linear shape** (PDP/SHAP) rather than chase leaderboard points.
- **Streaming (classification).** Two useful views:
  - **Logistic baseline** (time-split) found strong signal (AUC ~0.77), favoring **recall**.
  - **Gradient Boosting** traded some recall for **higher precision/accuracy** on 2021: **F1 ≈ 0.59, Precision ≈ 0.70, Recall ≈ 0.51, Accuracy ≈ 0.78** (TN=200, FP=21, FN=49, TP=50). Threshold is tunable depending on "catch more potential hits" vs "be stricter".

Conclusions

Two channels, two playbooks.

Theaters: success rides on **budget + momentum**. Spend enough to reach quality and scale, then invest early in **attention-building** (trailers, PR, fan communities) because the model treats attention as **incremental**, not redundant with budget. Place mid-length cuts in **Winter/Summer** unless you have true tentpole weight for Holiday.
Streaming: reception rewards **engagement and the right creative fit** more than the service logo. Greenlight **Documentary/Drama** (and strong story-driven projects), keep runtimes in the middle, and plan for **early ratings volume** - that moves the needle.

- **What this means in practice.**
  For a theatrical slate, use budget deliberately until returns start to flatten and make up the rest with audience momentum; treat season as a tie-breaker, not the main lever. For a streaming slate, prioritize projects that naturally spark conversation and give them the marketing beats that seed votes/ratings; don't over-optimize for platform brand.
- **Confidence and caveats.**
  Models were trained on earlier years and tested on later ones, so findings generalize across adjacent seasons. Results are robust to moving the year boundary and (in spot checks) to CPI adjustments. Main limitations: **budget missingness** (we imputed and regularized) and **RT score ≠ viewership** (we measured reception, not hours watched).
- **If extended.**
  I'd add **multi-genre encoding**, a simple **franchise flag**, and produce **threshold profiles** (high-recall vs high-precision) so different teams—development, marketing, awards—can pick the operating point that fits their goal.

Bottom line:
Theatrical success is fueled by spend and sustained by attention; streaming success is earned by engagement and content fit.

Appendix

Appendix A — Data lineage & cleaning (what exactly I did)
Sources.

- TMDb Movie Dataset v11 (2018–2024 slice) — production/box office side.
- Movies on Streaming Platforms (2018–2021) — platform flags & Rotten Tomatoes.

Join & labels.

- Normalized titles (lowercased, punctuation trimmed, colon subtitles removed).
- Joined on **title_norm + year (±1)**; dropped ambiguous many-to-many matches.
- **Theatrical** if `revenue > 0`; **Streaming** if any of `{netflix,hulu,prime_video,disney_plus}==1`.

Cleaning rules that matter.

- TMDb **0** for `budget/revenue` → treated as **missing**.
- Parsed `release_date` → `year`, `quarter`, `month_name`, **`release_season`** {Winter, Spring, Summer, Fall, Holiday}.
- Picked a single **`main_genre`** (dominant tag) for interpretability.
- Heavy tails → engineered: **`log_budget`, `log_revenue`, `log_votes` (=log1p vote_count), `log_popularity` (=log1p popularity)**.

Files (submitted).

- Final analysis dataset: `data_clean/movies_merged.csv` (the frame used for EDA/models).

Appendix B — EDA figures (evidence that guided modeling)
Fig. B1 — `hist_log_budget.png`
Log budget is well-behaved (long right tail tamed) → safe for linear baselines.
Fig. B2 — `hist_log_revenue.png`
Log revenue looks regular; evaluating errors in log units is sensible.
Fig. B3 — `scatter_logbudget_logrevenue.png`
Strong diagonal: higher budgets → higher revenue, with spread that timing/genre/attention can explain.
Fig. B4 — `box_logrevenue_by_season.png`
Winter (and Summer) run higher on median; Holiday is volatile (tentpoles + many modest titles).
Fig. B5 — `stream_hist_rt_score.png`
RT scores cluster in 40s–60s; tails reach into the 80s–90s.
Fig. B6 — `stream_box_rt_by_platform.png`
Platform medians differ a bit; heavy overlap ⇒ platform effects are modest after content/engagement.

Appendix C — Modeling details (so it's reproducible)
Splits (time-aware).

- **Theatrical:** train **2018–2022**, test **2023–2024**; target = `log_revenue`.
- **Streaming:** train **2018–2020**, test **2021**; target = 75th-percentile RT **computed on train only**.

Pipelines.

- Numeric → `SimpleImputer(median)` → `StandardScaler` (for Elastic Net).
- Categorical → `OneHotEncoder(handle_unknown="ignore")`.
- `ColumnTransformer` wraps both; everything fit **on train only**.

Models & tuning.

- **Theatrical baseline:** Elastic Net (`alpha` log-grid; `l1_ratio`∈`{0.1…0.9}`), `max_iter=10000`.
- **Theatrical diagnostic:** Gradient Boosting Regressor (`n_estimators`∈`{200,400,500}`, `max_depth`∈`{2,3,4}`, `learning_rate`∈`{0.05,0.1}`, `subsample`∈`{0.7,0.9,1.0}`).
- **Streaming primary:** Gradient Boosting Classifier (same-style grid), scored on ROC-AUC in CV.
- CV is **blocked by year** within the train window.

Best params are printed in the notebook cells (kept in outputs for the grader).

Appendix D — Explainability (what drives predictions)
Theatrical — Gradient Boosting

- **Fig. D1 — `theatrical_gb_pdp_log_budget.png`**: **Rising then flattening** → diminishing returns to spend.
- **Fig. D2 — `theatrical_gb_pdp_runtime.png`**: gentle mid-range sweet spot for runtime.
- **Fig. D3 — `theatrical_gb_shap_bar.png`**: global ranking → `log_budget` ≫ `log_popularity` > `runtime` > `log_votes` > (season/genre).
- **Fig. D4 — `theatrical_gb_shap_beeswarm.png`**: direction → high budget/attention push revenue **up**; Holiday and a few genres tilt **down**.

Streaming — Gradient Boosting

- **Fig. D5 — `stream_gb_shap_bar.png`**: global ranking → `log_votes` dominates; `runtime` + content tags next; platform flags are smaller.
- **Fig. D6 — `stream_gb_shap_beeswarm.png`**: direction → engagement & Doc/Drama push **up**; TV-movie cues/holiday timing push **down**.

Tables (top drivers)

- **Tab. D1 — `tables/theatrical_gb_shap_top20.csv`**
- **Tab. D2 — `tables/streaming_gb_shap_top20.csv`**

Appendix E — Robustness & sensitivity
Time-split shift (theatrical).

- **Tab. E1 — `tables/robust_theatrical_splits.csv`**
  Compare train≤2021 vs train≤2022. **Takeaway:** R^2/MAE are broadly stable ⇒ conclusions are not an artifact of a particular boundary.

Nominal vs CPI dollars (theatrical).

- **Tab. E2 — `tables/robust_cpi_vs_nominal.csv`**
  Deflating to 2024 dollars yields **near-identical test metrics** ⇒ our 2018–2024 window is short enough that logs + nominal are fine.

Appendix F — Metrics & evaluation artifacts
Streaming (test=2021).

- **Fig. F1 — `stream_gb_confusion.png`**: Confusion matrix at 0.5 shows a **precision-leaning** operating point (good for trusted picks).
- **Tab. F2 — `tables/stream_gb_metrics.csv`**: AUC, F1, Precision, Recall, Accuracy, base rate.

Theatrical (test=2023–2024).

- Summarize Elastic Net **MAE (log)** and **R^2** in the text; if you export a CSV, title it **`tables/theatrical_en_metrics.csv`** (MAE, R^2, test n).

Appendix G — Reproducibility & file map
Folders submitted.

- `data_clean/` → `movies_merged.csv` (final analysis dataset).
- `figs/` → all figures listed above (PNG).
- `tables/` → CSVs: `data_dictionary.csv`, `missingness.csv`, `theatrical_gb_shap_top20.csv`, `streaming_gb_shap_top20.csv`, `robust_theatrical_splits.csv`, `robust_cpi_vs_nominal.csv`, `stream_gb_metrics.csv`
- Notebook: end-to-end code (clean → EDA → models → SHAP/PDP → exports). All paths are **relative**.