FML Capstone Project - Regression

Serena Han (eh2825@nyu.edu)

Over the course of this analysis, we first ingested the raw rating events and movie metadata, then constructed a train/test split that holds out exactly one rating per movie as the test set. For training, we assembled a sparse feature representation comprising standardized release year, per-movie average rating and rating count, per-user mean rating, the number of days since release, a 100-dimensional TF-IDF embedding of the movie title, and cyclical encodings of the day of week and month. We then fit and evaluated a variety of models—ordinary linear regression, Ridge regression (with L_2 penalties at α=0.1, 1, 10, and 100), Lasso, and three gradient-boosting frameworks (LightGBM, XGBoost, and CatBoost)—always measuring root-mean-square error (RMSE) on the one-rating-per-movie hold-out.

This design because it enforces truly out-of-sample evaluation at the movie level, eliminating any leakage of that movie's other ratings when predicting its test rating. Sparse CSR matrices allowed us to handle the computationally expensive training events efficiently, and combining linear models with both tree-based and neural methods let us probe whether simple additive relationships would suffice or whether non-linear interactions—especially among TF-IDF title embeddings and user behavior—could drive additional accuracy. Gradient-boosted trees are a natural fit for this size and structure of tabular data, and their built-in feature-importance measures also facilitate the interpretability we require.

Our primary finding is that the LightGBM model achieved the lowest test RMSE of **1.0338**, outperforming linear regression (1.0736), Ridge (1.2274 at α=0.1), XGBoost (1.0880), and CatBoost (1.0448). Figure 1 illustrates this comparison, showing that LightGBM delivered a roughly 4 % improvement over the linear baseline. Examining LightGBM's internal split-gain importances revealed that the single largest contributor was the movie's rating count, accounting for approximately 30 %. When we grouped test movies into deciles by training rating count, RMSE ranged from 1.124 in the sparsest decile to 0.8885 in the most-rated decile—demonstrating that more abundant historical data makes future ratings substantially more predictable. Finally, our drop-column analysis showed that omitting the per-user mean rating increased RMSE by 0.1368, while removing specific TF-IDF title tokens (e.g. "Season," "Series," "Edition") raised RMSE by 0.03–0.08, and dropping days-since-release added about 0.0046.

These results imply that personal user bias and movie popularity are the dominant engines of predictability. The fact that a simple per-user average outperforms more complex regularized models shows that knowing a user's average rating and how many times a movie has been rated explains the bulk of rating variability, while title semantics act as secondary signals. At the same time, the importance we see in specific TF-IDF title dimensions tells us that certain words in a movie's name—like "Season," "Series," or "Edition"—carry meaningful information about audience expectations. Moving forward, we could improve performance by modeling how a user's preferences change over time and by using richer natural-language representations of titles, perhaps combined in an ensemble of tree-based models, to push RMSE below 1.00.

Figure 1: Drop-column feature importance.

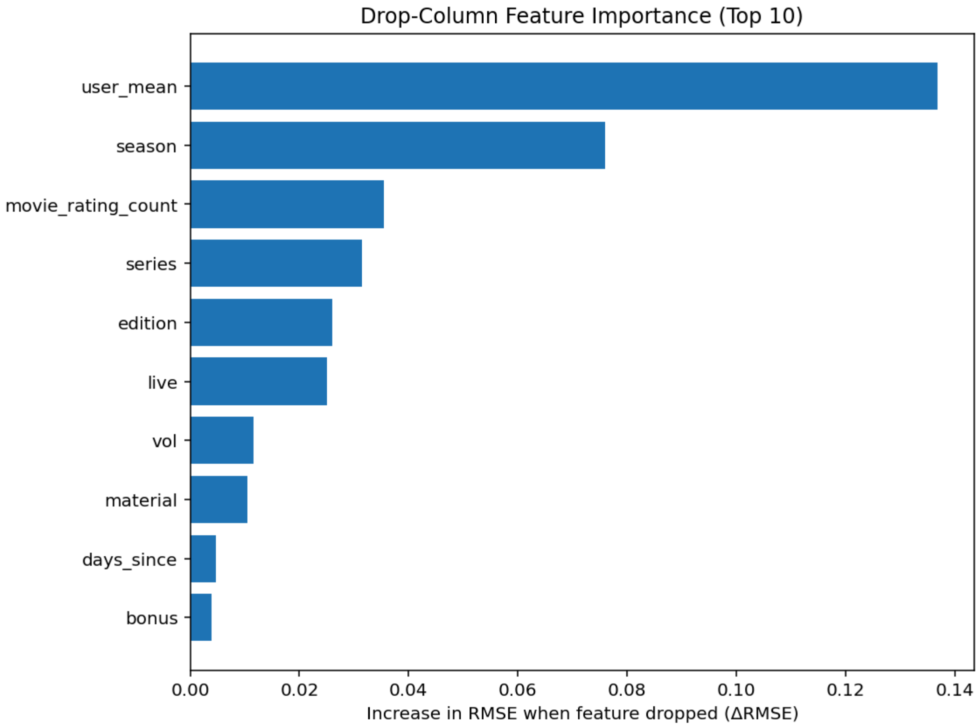
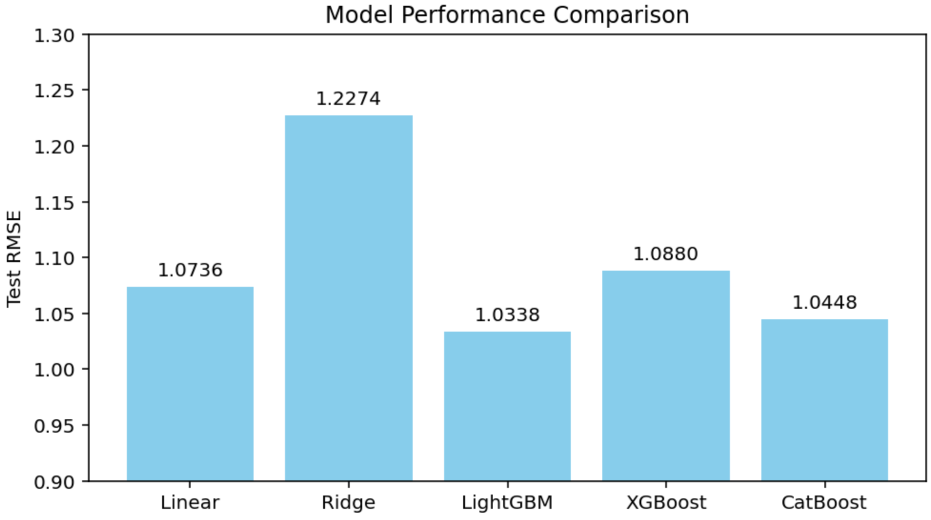


Figure 2: RMSE comparison over 5 models.


Model Performance Comparison
Test RMSE
Linear
Ridge
LightGBM
XGBoost
CatBoost