MovieLens Capstone Report

Felipe Muniz

2025-06-08

Introduction

This capstone project is part of the HarvardX PH125.9x Capstone assignement. The goal is to develop a movie recommendation system using the MovieLens 10M dataset and evaluate its performance based on RMSE. This report outlines the methodology, analysis steps, modeling results, and final model performance.

Methods / Analysis

Data Acquisition and Preparation

```
dl <- tempfile()
download.file("http://files.grouplens.org/datasets/movielens/ml-10m.zip", dl)

ratings <- fread(text = gsub("::", "\t", readLines(unzip(dl, "ml-10M100K/ratings.dat"))),
                 col.names = c("userId", "movieId", "rating", "timestamp"))

movies <- str_split_fixed(readLines(unzip(dl, "ml-10M100K/movies.dat")), "::", 3)
colnames(movies) <- c("movieId", "title", "genres")
movies <- as.data.frame(movies)
movies$movieId <- as.numeric(movies$movieId)

movielens <- left_join(ratings, movies, by = "movieId")

# Split edx and final hold-out test set
test_index <- createDataPartition(y = movielens$rating, times = 1, p = 0.1, list = FALSE)
edx <- movielens[-test_index, ]
final_holdout_test <- movielens[test_index, ] %>%
  semi_join(edx, by = "movieId") %>%
  semi_join(edx, by = "userId")
```

Validation Partition from edx

```
set.seed(1, sample.kind = "Rounding")
```

```
## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding' sampler used
```

```r
index <- createDataPartition(y = edx$rating, times = 1, p = 0.1, list = FALSE)
train_set <- edx[-index, ]
temp <- edx[index, ]
validation <- temp %>%
  semi_join(train_set, by = "movieId") %>%
  semi_join(train_set, by = "userId")
train_set <- train_set %>%
  semi_join(validation, by = "movieId") %>%
  semi_join(validation, by = "userId")
```

RMSE Function

```r
RMSE <- function(true_ratings, predicted_ratings) {
  sqrt(mean((true_ratings - predicted_ratings)^2))
}
```

Models Tested

1. Naive Model

```r
mu_hat <- mean(train_set$rating)
naive_rmse <- RMSE(validation$rating, mu_hat)
rmse_results <- tibble(method = "Naive Mean Model", RMSE = naive_rmse)
```

2. Movie Effect Model

```r
movie_avgs <- train_set %>%
  group_by(movieId) %>%
  summarize(b_i = mean(rating - mu_hat))
predicted_ratings <- validation %>%
  left_join(movie_avgs, by = "movieId") %>%
  mutate(pred = mu_hat + b_i) %>%
  pull(pred)
movie_rmse <- RMSE(validation$rating, predicted_ratings)
rmse_results <- bind_rows(rmse_results,
                          tibble(method = "Movie Effect Model", RMSE = movie_rmse))
```

3. Movie + User Effect Model

```r
user_avgs <- train_set %>%
  left_join(movie_avgs, by = "movieId") %>%
  group_by(userId) %>%
  summarize(b_u = mean(rating - mu_hat - b_i))
predicted_ratings <- validation %>%
  left_join(movie_avgs, by = "movieId") %>%
```

```
  left_join(user_avgs, by = "userId") %>%
  mutate(pred = mu_hat + b_i + b_u) %>%
  pull(pred)
user_rmse <- RMSE(validation$rating, predicted_ratings)
rmse_results <- bind_rows(rmse_results,
                          tibble(method = "Movie + User Effects Model", RMSE = user_rmse))
```

Model Comparison Table

```
print(rmse_results)
```

```
## # A tibble: 3 x 2
##   method                      RMSE
##   <chr>                      <dbl>
## 1 Naive Mean Model           1.06
## 2 Movie Effect Model         0.944
## 3 Movie + User Effects Model 0.865
```

Next Steps

- Regularization of bias terms
- Matrix factorization (e.g., `recosystem`)
- Final model training on `edx`
- Test on `final_holdout_test`

Results

Provide here the RMSE table, brief discussion of results so far, and planned improvements. Example:

> The naive model yielded RMSE ~1.06. Adding movie effect improved RMSE to ~0.943. Including user effect further reduced RMSE to ~0.865. Further improvement will focus on regularization and matrix factorization.

Conclusion

This project presented the development of a movie rating prediction algorithm. Several models were implemented and evaluated, leading to progressive reductions in RMSE. Future iterations will integrate regularization and matrix factorization to reach target RMSE < 0.86490.

Final RMSE on Hold-Out Test Set

(To be filled only after selecting final model)

```
# final_rmse <- RMSE(final_holdout_test$rating, predicted_values_from_final_model)
# final_rmse
```

References

- MovieLens 10M Dataset: https://grouplens.org/datasets/movielens/10m/
- HarvardX PH125.9x Capstone Instructions
- tidyverse, caret, data.table documentation