Movie Capstone

2024-09-13

Contents

1. Introduction
2. Methods and Analysis
   2.1 Data Preparation
   2.2 Exploratory Data Analysis
   2.3 Model Development
3. Results
4. Conclusion
   4.1 Limitations and Future Work
5. Final Model Evaluation

Glossary

- **RMSE (Root Mean Square Error)**: A measure of the differences between predicted values and actual values. It’s used to evaluate the performance of a model.
- **MovieLens Dataset**: A dataset of movie ratings provided by users, used widely in recommendation systems research.
- **Baseline Model**: A simple predictive model that uses the average rating of each movie to make predictions. It's typically used as a reference point for evaluating more advanced models.
- **edx**: The training set, derived from the MovieLens dataset, used to train the machine learning model.
- **final_holdout_test**: The validation set, used only for final model evaluation to ensure the model's generalizability.

1. Introduction

The goal of this project is to develop a movie recommendation system using the MovieLens dataset. By predicting user ratings for movies they haven’t seen, we aim to minimize the Root Mean Square Error (RMSE) between predicted and actual ratings. This project uses a simple baseline model, with plans for future iterations to include more advanced algorithms.

2. Methods and Analysis

2.1 Data Preparation

We start by loading the necessary libraries and importing the MovieLens dataset.

```
# Load necessary libraries
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 -- ## v dplyr 1.1.4 v
readr 2.1.5
## v forcats 1.0.0 v stringr 1.5.1
## v ggplot2 3.5.1 v tibble 3.2.1
## v lubridate 1.9.3 v tidyr 1.3.1
## v purrr 1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() -- ## x dplyr::filter() masks
stats::filter()
## x dplyr::lag() masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errorlibrary(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
## lift
```

```
# Load the data
ratings <- read.csv("ratings.csv")
movies <- read.csv("movies.csv")
```

The ratings and movies datasets are then merged on the `movieId` column. We split the dataset into training (edx) and validation (final_holdout_test) sets to ensure that our model is not trained on the validation data.

2.2 Exploratory Data Analysis

Let's explore the dataset to gain some insights into the rating distribution and movie popularity:

The ratings and movies datasets are merged, and the data is split into training (edx) and validation (final_holdout_test) sets.

```
# Merge the ratings and movies datasets on the movieId column
merged_data <- merge(ratings, movies, by = "movieId")

# Split the data into edx (90%) and final_holdout_test (10%)
set.seed(1)
test_index <- createDataPartition(merged_data$rating, p = 0.1, list = FALSE) edx <-
merged_data[-test_index, ]
final_holdout_test <- merged_data[test_index, ]

# Ensure final_holdout_test has only users and movies that are also in edx
final_holdout_test <- final_holdout_test %>%
  semi_join(edx, by = "movieId") %>%
  semi_join(edx, by = "userId")
```

2.3 Model Development

2.3.1 Baseline Model

We start with a simple baseline model that calculates the average rating for each movie and uses this to predict user ratings.

We calculate the average rating for each movie as the baseline model and then calculate RMSE to evaluate its performance.

```
# Calculate the average rating for each movie
movie_avg_rating <- edx %>% group_by(movieId) %>% summarise(avg_rating = mean(rating))

# Merge the average ratings into the final_holdout_test set
final_holdout_test <- final_holdout_test %>%
  left_join(movie_avg_rating, by = "movieId")

# Calculate RMSE between actual ratings and predicted average ratings rmse <-
RMSE(final_holdout_test$avg_rating, final_holdout_test$rating)
print(paste("Baseline RMSE:", rmse))
```

```
## [1] "Baseline RMSE: 0.96170576181978"
```

3. Results

The baseline model produced an RMSE of **0.9617**, providing a foundation for evaluating future improvements.

Explanation:

- **Introduction Section**: Explains the goal of the project.
- **Data Loading Section**: Loads the necessary libraries and dataset.
- **Data Preparation Section**: Merges the datasets, splits the data, and ensures consistency between edx and final_holdout_test.
- **Model and Evaluation Section**: Calculates the average rating, merges it, and computes the RMSE.
- **Conclusion Section**: Displays the RMSE and mentions possible future steps.

4. Conclusion

In this project, we successfully developed a movie recommendation system based on the average rating model. While the baseline model performed reasonably well, achieving an RMSE of **0.9617**, there are several areas for improvement.

4.1 Limitations and Future Work

1. **Temporal Effects**: Our model doesn't account for changes in user preferences over time. Incorporating

time-based data could improve predictions.

2. **Genre Effects**: Movie genres were not considered in this baseline model. Including genre-based features could personalize recommendations further.
3. **More Advanced Models**: Future iterations will explore more sophisticated techniques such as collaborative filtering and matrix factorization to improve accuracy.
4. **Cold Start Problem**: Our model may struggle with new users or movies with limited data. Addressing this will be critical for a production-ready system.

5. Final Model Evaluation

To finalize the model, we retrain it on the entire edx dataset and use it to predict ratings for the final holdout test set. The next step would be implementing more advanced models to refine the predictions further and improve performance beyond the baseline.