Report_Rmd_format

Luis D. Torres

20/12/2020

MovieLens Project Submission Report

Introduction

This report presents the results of creating a movie recommendation system using the 10M version of MovieLens dataset available at https://grouplens.org/datasets/movielens/10m/

The report uses a machine learning algorithm following a model based approach and the calculation of the residual mean squared error (RMSE) to compare models.

This report is structured in three sections. The next section explains the methods and analysis used including data cleaning, data exploration and visualization, and the modelling approach. The results section presents the modelling results and discusses the model performance using the RMSE as a loss function. The final section gives a brief summary of the report, its limitations and future work.

Method

Dataset

The movie recommendation system uses the 10M version of MovieLens dataset available at https://grouplens.org/datasets/movielens/10m/

The dataset was divided in two main subsets: 1. edx: 90% of the MovieLens data. Total observations of 9,000,063 2. validation: 10% of MovieLens data. Total observations of 999,995

All models were trained on the **edx** dataset. To facilitiate cross-validation, **edx** was further partitioned into *train* (90% of edx) and *test* (10% of edx).

The **validation** dataset was not used for training and cross-validation purposes. This set was used only for full validation. This implies testing the final algorithm and retrieving the full validation RMSE.

Variables

- Outcome: The variable that the machine learning algorithm attemps to predicts is the *movie ratings*.
- Features: The variables used to predict are: *movies*, *users*, *time* (including day, week, month, year).

Data cleaning

The total observation included in the **edx** and **validation** datasets do not represent the exact percentages of the 10M MovieLens dataset. This is because users and movies in the **validation** set that do not appear in the **edx** set were removed and added back to the edx set.

Similarly, the **validation** set was also aligned to the user and movies present in the *train* set. This was performed to facilitate full validation.

Data analysis

Analysis were performed using three R packages: *tidyverse*, *data.table* and *lubridate* for data manipulation and visualisation; and the *caret* for building machine learning models.

This report uses RMSE as the main loss function to compare models and their predictive value. If RMSE is larger than 1, it means the typical error is larger than one star. The model should aim at a value lower than 1 and ideally closer to 0.

It also uses penalised least squares for regularisation purposes. Regularization penalise large estimates that are formed using small sample sizes.

Results

Inspecting the *train* dataset

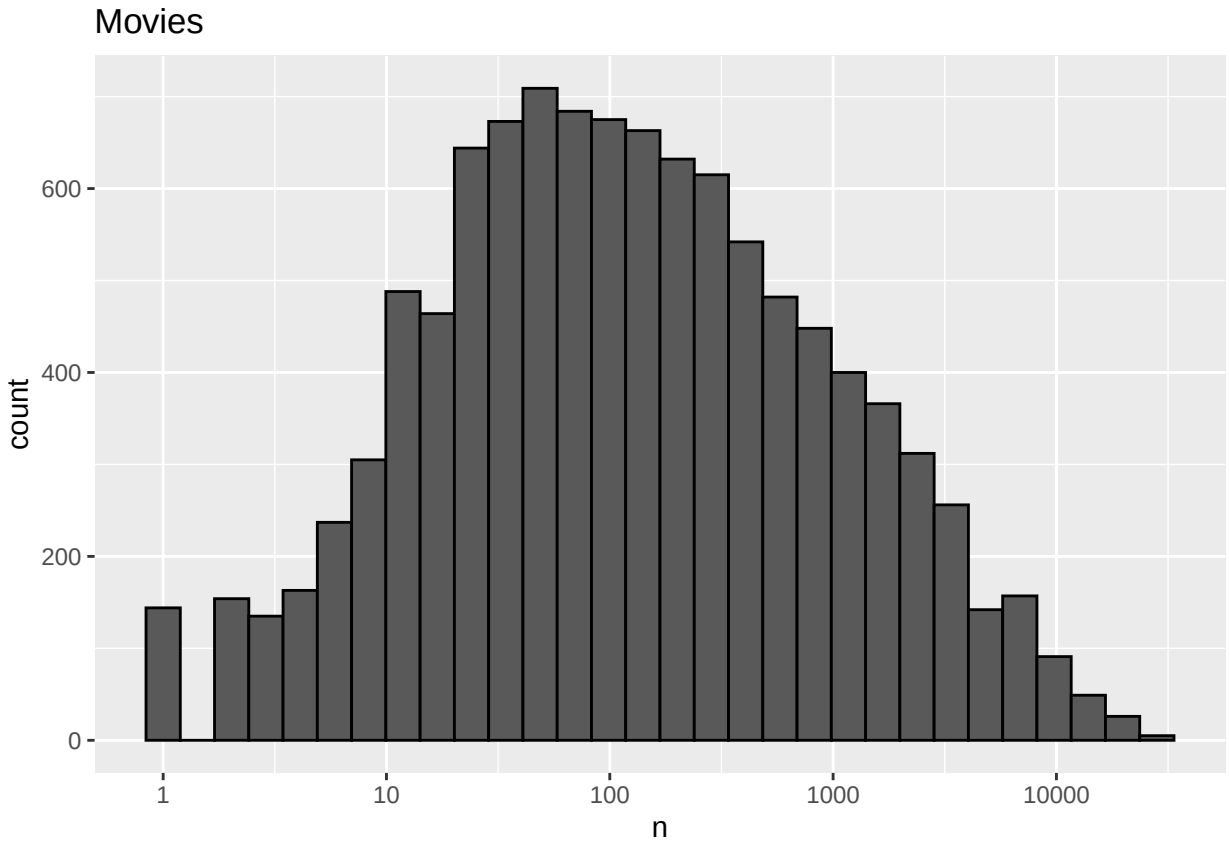
The total number of users and movies is:

number_users	number_movies
69878	10661

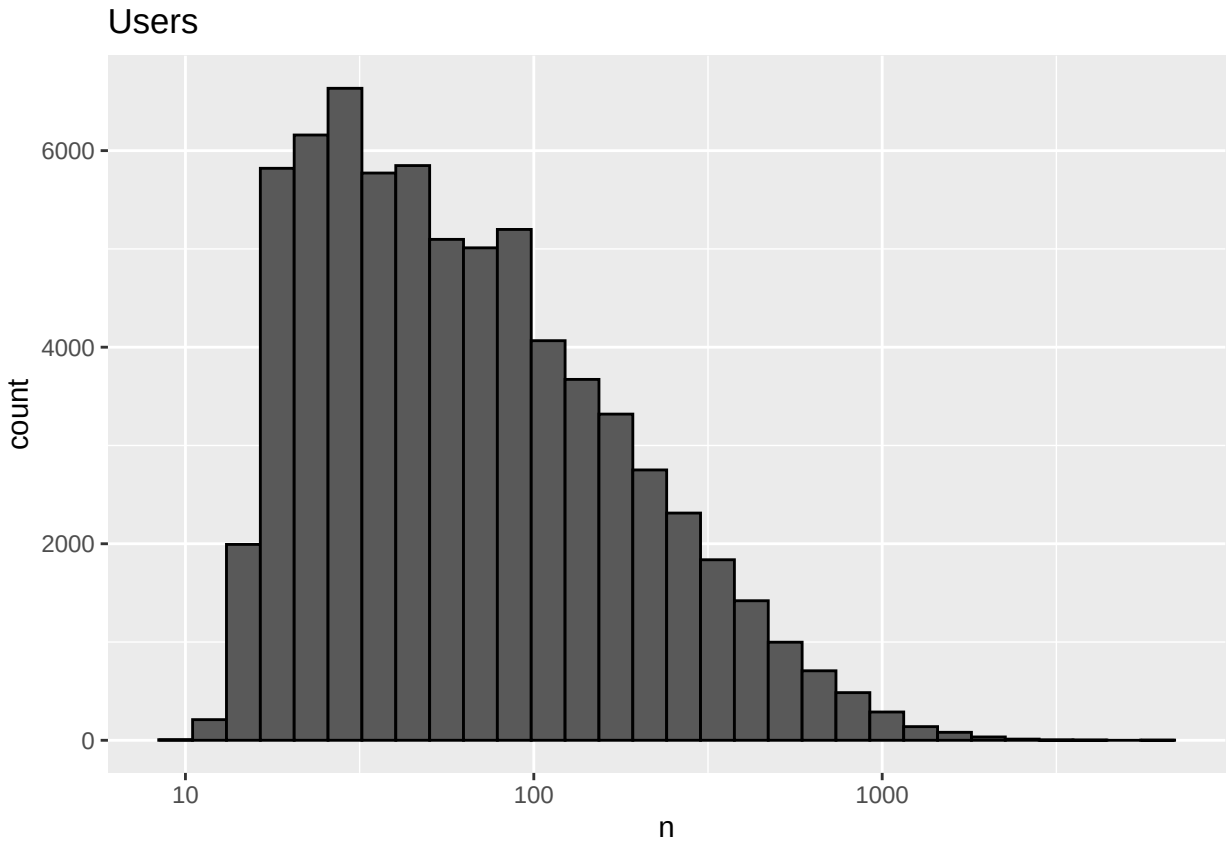
Identifying user's rating for the top five movies:

userId	Forrest Gump (1994)	Jurassic Park (1993)	Pulp Fiction (1994)	Shawshank Redemption, The (1994)	Silence of the Lambs, The (1991)
1	5	NA	NA	NA	NA
4	NA	5	NA	NA	NA
7	NA	NA	NA	NA	3
8	NA	3	NA	NA	4
10	3	NA	2	NA	3
11	NA	4	3	NA	NA
13	NA	NA	4	NA	NA
16	NA	3	NA	NA	NA
17	NA	NA	NA	NA	5
18	NA	3	NA	4.5	5
19	4	1	NA	4.0	NA

Some movies receive more ratings than others as shown by the graph below:



Some users also provide more ratings than others as shown by the graph below:



Building predictive models (Training stage)

Model 1 This is a simple model including just the average ratings for all movies and users. This model assumes the same rating for all movies and users with all the differences explained by random variation. Independent errors sampled from the same distribution should centered at 0 and the media (mu) would represent the "true" rating for all movies.

Therefore, as the estimate that minimises the RMSE is the least squares estimate of the media, in this case it be assumed that this is the average of all ratings:

```
## [1] 3.512457
```

If all unknown ratings are predicted with the calculated average, the RMSE is:

```
## [1] 1.060054
```

In this case the RMSE is larger than 1 star rating.

Model 2 Some movies may be rated higher than others, so building a model that accounts for this effect or bias may help to improve the predictive power.

By doing this, the achieved RMSE becomes:

```
## [1] 0.9429615
```

This is better than model 1, but still close to 1 star deviation.

Model 3 As per the movie effect, some user may be giving higher ratings than other. Adding the user effect may also improve the prediction.

By doing this, the achieved RMSE becomes:

```
## [1] 0.8646844
```

Model 4 The time in which a movie is watch could have an effect on the ratings given. For example, weekends people may be more relax to watch a movie and keen to provide a rating. Something sinmilar can be assumed for summer and winter months. Years could also play a role when considered that some "cult" movies have been released in those years.

I test for this assumptions by adding day, week, month and year to the models.

By adding day to model 3, a small improvement is achieved in the RMSE value as follows:

```
## [1] 0.8642016
```

This improvement is not present when week is added to the model as the RMSE shows no change:

```
## [1] 0.8642016
```

Consequently, Week is no considered any longer in the models.

Similarly, month does not add to the model when added to the movie+user+day model. The achieved RMSE is:

```
## [1] 0.8642023
```

Consequently, month is no considered any longer in the models.

Likewise, year does not improve the RMSE:

```
## [1] 0.8642016
```

As a result, only day is considered relevant to improve the prediction.

Regularisation of predictive models

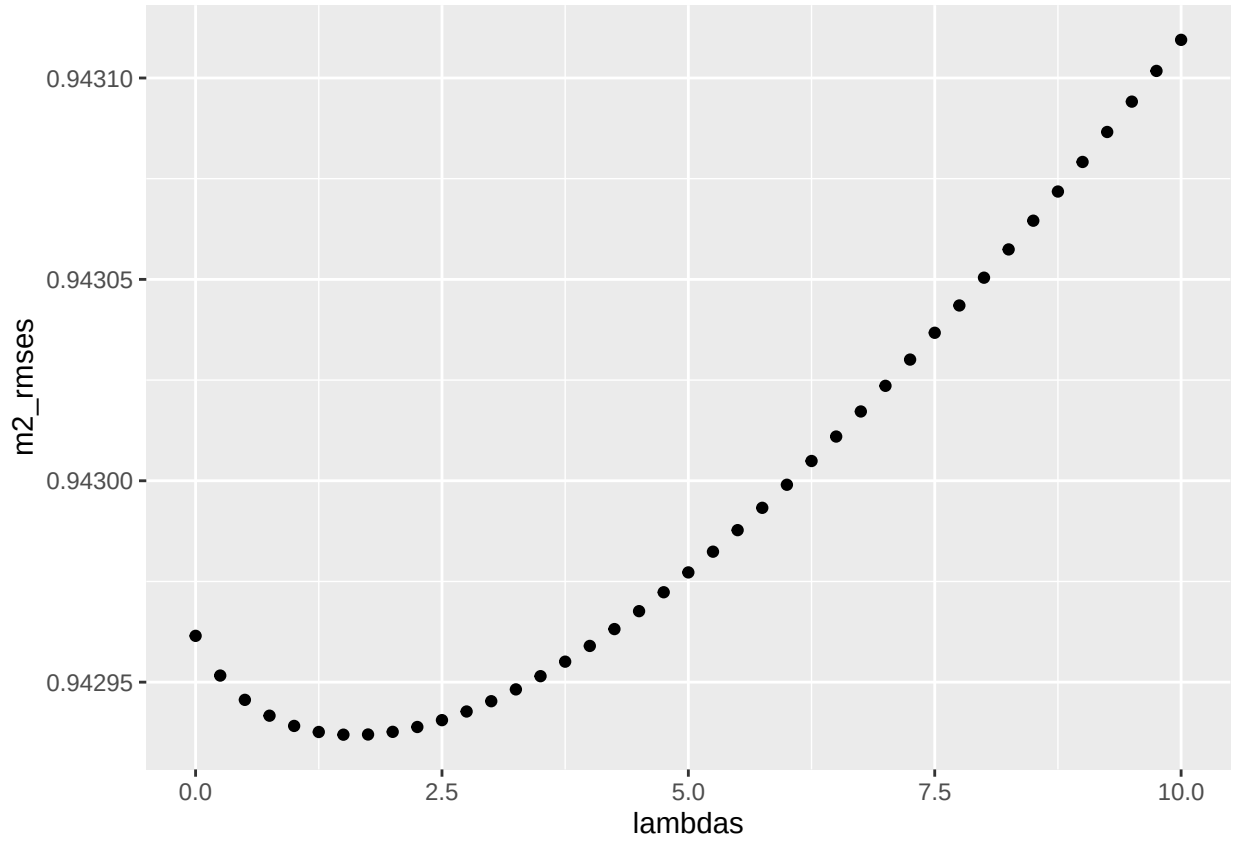
Regularisation allows to penalise large estimates that are formed using small sample sizes. This useful in this case. For example, by using only the movie effects on the graph below it can shown the top 10 worst and best movies based on the movie efect and number of ratings. As showed the supposed "best" and "worst" movies were rated by very few users, in most cases only 1.

title	b_movie	n
Hellhounds on My Trail (1999)	1.487543	1
Satan's Tango (SÃ¡tÃ¡ntangÃ³) (1994)	1.487543	1
Shadows of Forgotten Ancestors (1964)	1.487543	1
Fighting Elegy (Kenka erejii) (1966)	1.487543	1
Sun Alley (Sonnenallee) (1999)	1.487543	1
Blue Light, The (Das Blaue Licht) (1932)	1.487543	1
Who's Singin' Over There? (a.k.a. Who Sings Over There) (Ko to tamo peva) (1980)	1.237543	4
Life of Oharu, The (Saikaku ichidai onna) (1952)	1.237543	2
Human Condition II, The (Ningen no joken II) (1959)	1.237543	4
Human Condition III, The (Ningen no joken III) (1961)	1.237543	4

title	b_movie	n
Besotted (2001)	-3.012457	1
Hi-Line, The (1999)	-3.012457	1
Confessions of a Superhero (2007)	-3.012457	1
War of the Worlds 2: The Next Wave (2008)	-3.012457	2
SuperBabies: Baby Geniuses 2 (2004)	-2.767776	47
Disaster Movie (2008)	-2.745790	30
From Justin to Kelly (2003)	-2.638140	183
Hip Hop Witch, Da (2000)	-2.603366	11
Criminals (1996)	-2.512457	1
Mountain Eagle, The (1926)	-2.512457	2

Therefore, the regularisation approach taken here considers models builded in the previous step (model 2 to 4).

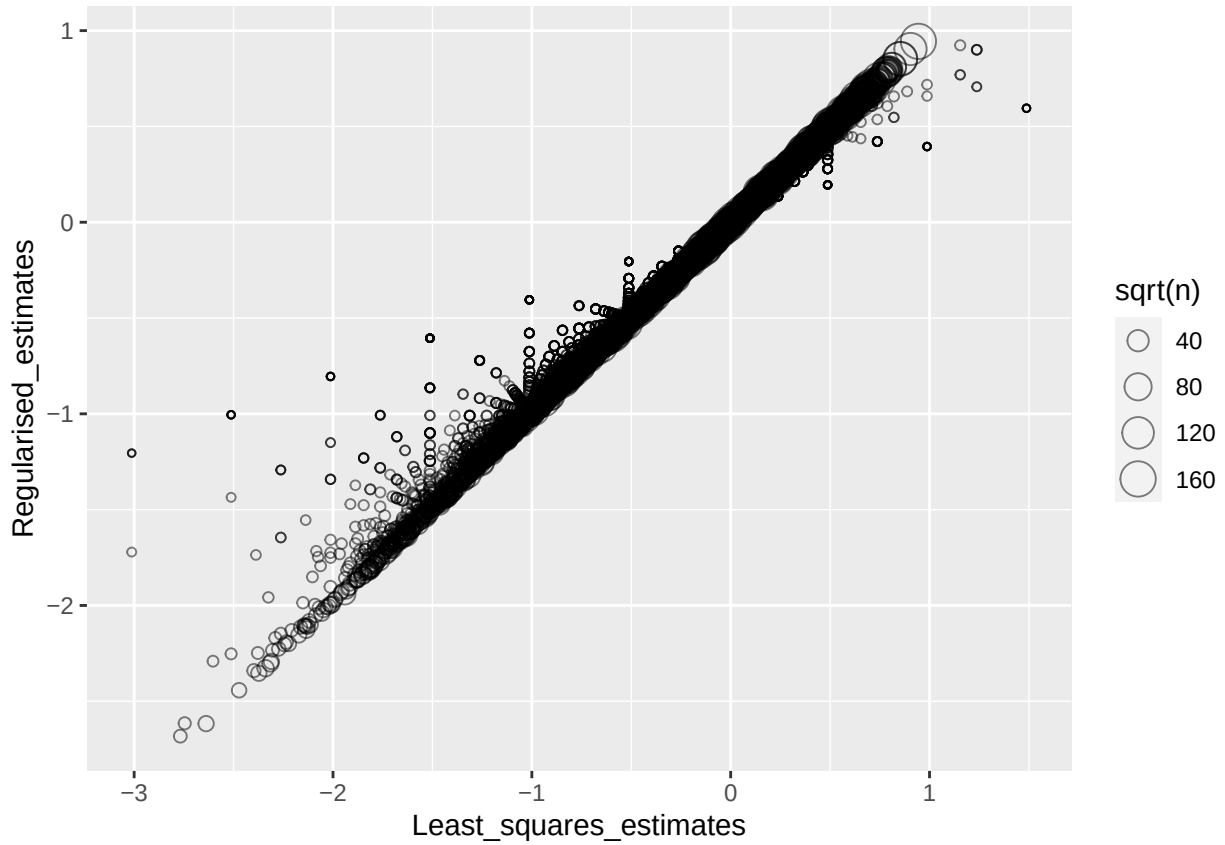
Regularised model 2 To regularise the movie effect model, the first step is to identify a penalty term or lambda. The results of a cross-validation procedure to choose lambda is shown in the graph below:



The lambda value that minimises RMSE is:

```
## [1] 1.5
```

With the chosen lambda, it is possible to compute the regularised estimates for model 2 and compare them to the original estimates. As shown on the graph below, when n is small the values shrink towards zero:



The regularised model also seems to be more accurate at identifying the top 10 best and worst movies compare to the model without regularisation above. The regularised model for the top 10 best movies is shown below:

```
##  [1] "Shawshank Redemption, The (1994)"
##  [2] "More (1998)"
##  [3] "Godfather, The (1972)"
##  [4] "Who's Singin' Over There? (a.k.a. Who Sings Over There) (Ko to tamo peva) (1980)"
##  [5] "Human Condition II, The (Ningen no joken II) (1959)"
##  [6] "Human Condition III, The (Ningen no joken III) (1961)"
##  [7] "Usual Suspects, The (1995)"
##  [8] "Schindler's List (1993)"
##  [9] "Rear Window (1954)"
## [10] "Casablanca (1942)"
```

The regularised model for the top 10 worst movies is shown below:

```
##  [1] "SuperBabies: Baby Geniuses 2 (2004)"
##  [2] "From Justin to Kelly (2003)"
##  [3] "Disaster Movie (2008)"
##  [4] "PokÃ©mon Heroes (2003)"
##  [5] "Barney's Great Adventure (1998)"
##  [6] "Carnosaur 3: Primal Species (1996)"
##  [7] "Glitter (2001)"
##  [8] "Gigli (2003)"
##  [9] "Pokemon 4 Ever (a.k.a. PokÃ©mon 4: The Movie) (2002)"
```

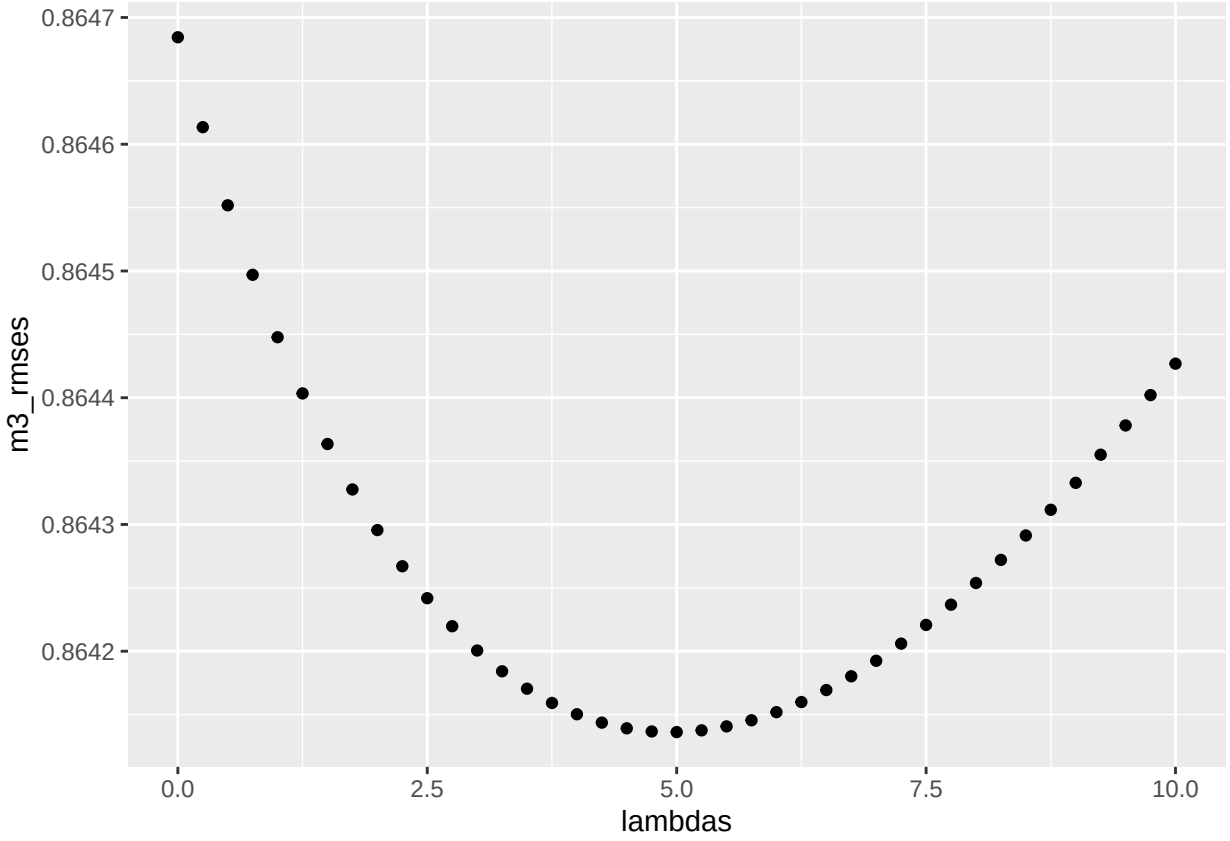
```
## [10] "Hip Hop Witch, Da (2000)"
```

The resulted RMSE after this regularisation is:

```
## [1] 0.942937
```

Regularised model 3 To regularise the user effect model, a similar procedure is follow.

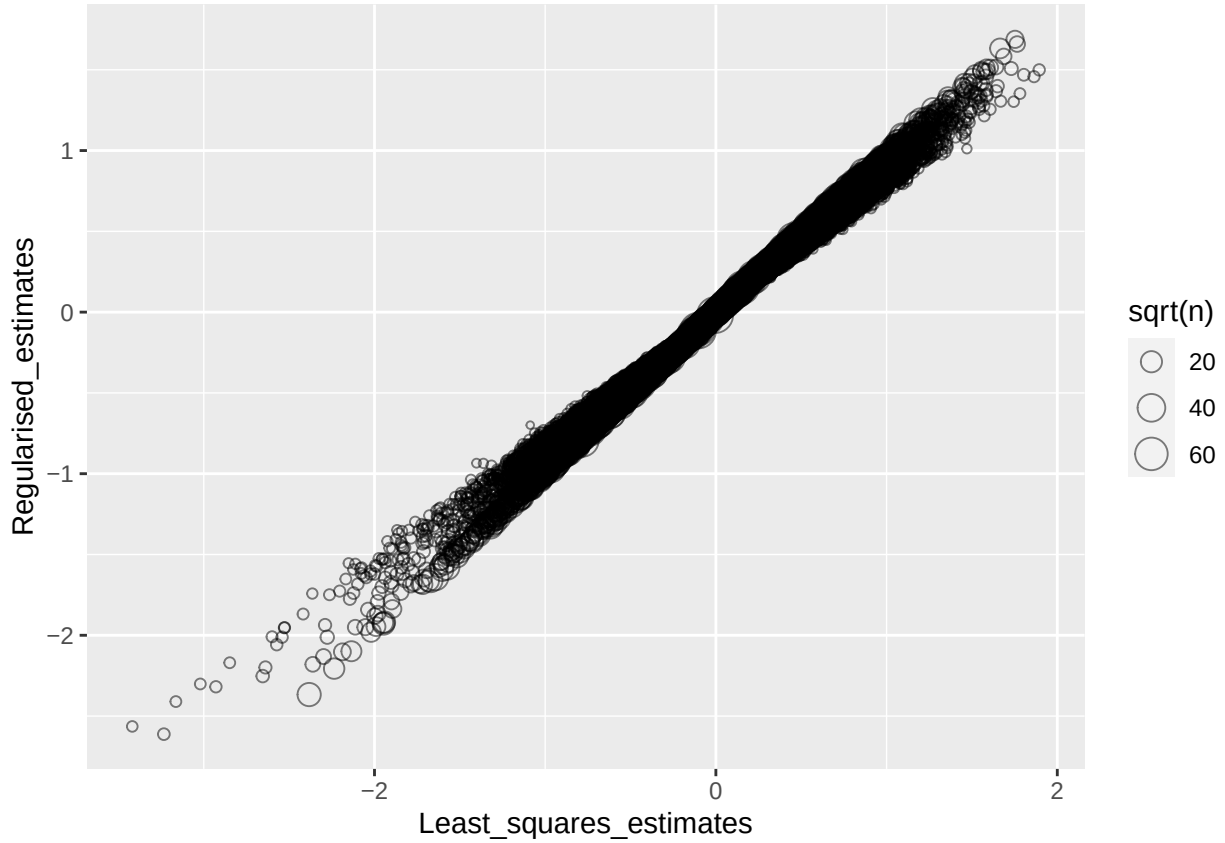
The results of a cross-validation procedure to choose lambda is shown in the graph below:



The lambda value that minimises RMSE is:

```
## [1] 5
```

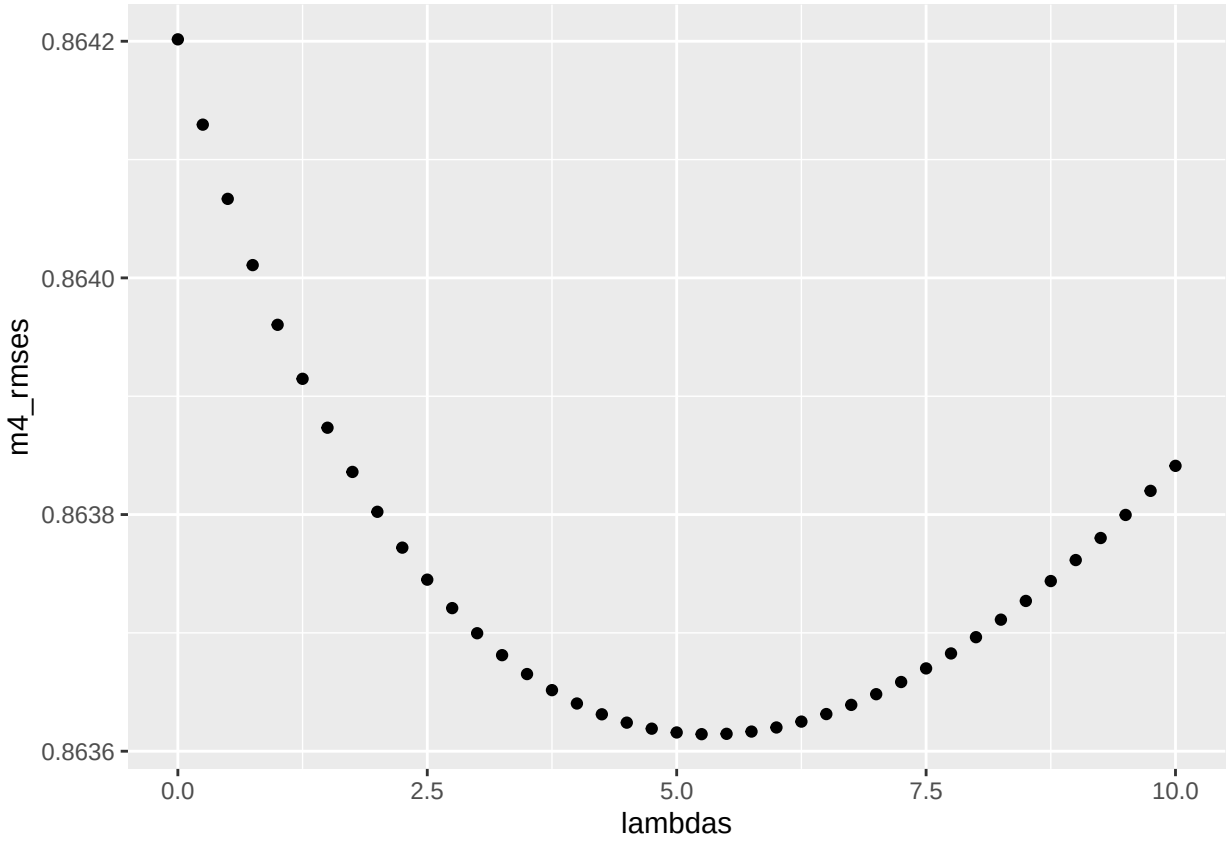
With the chosen lambda, it is possible to compute the regularised estimates for model 3 and compare them to the original estimates. As for the previous case, when n is small the values shrink towards zero:



The resulted RMSE after this regularisation is:

```
## [1] 0.8641362
```

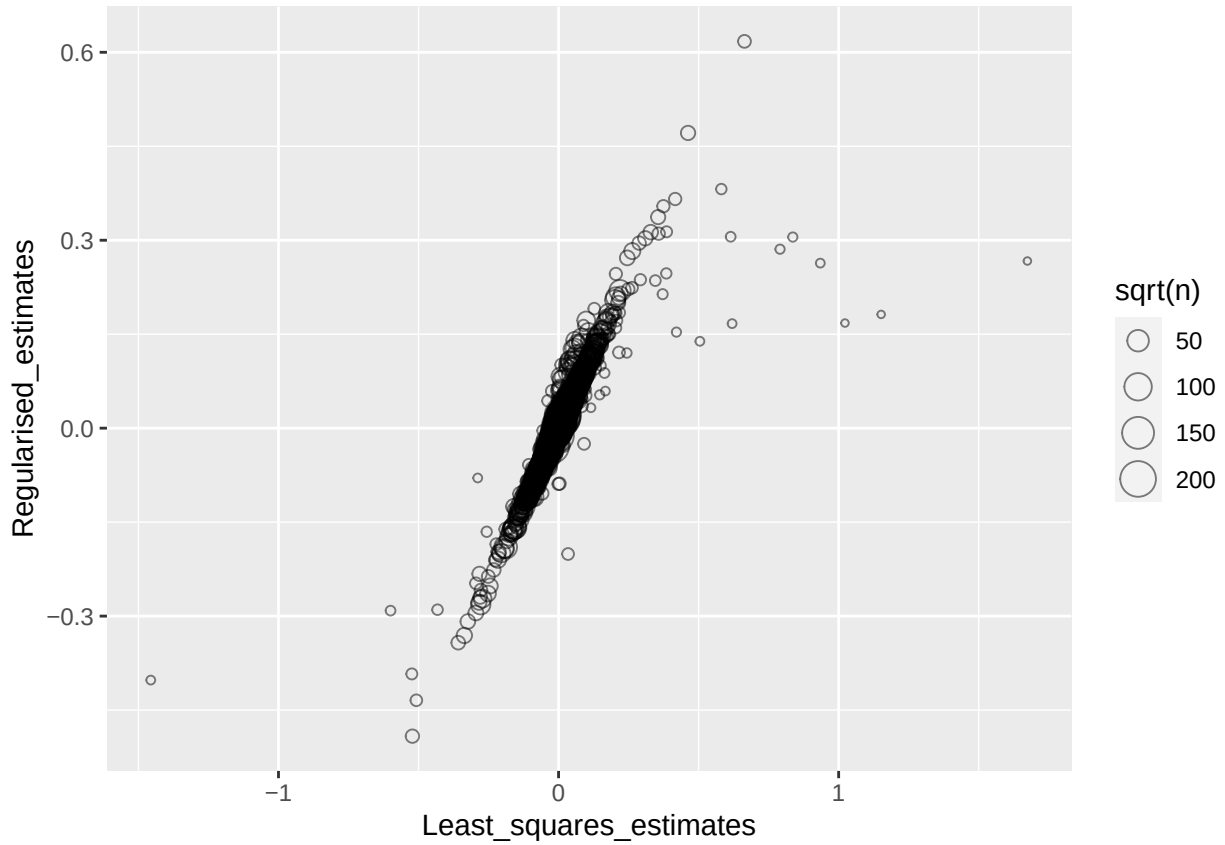
Regularisation model 4 To regularise the day effect model, a similar procedure is follow. The results of a cross-validation procedure to choose lambda is shown in the graph below:



The lambda value that minimises RMSE is:

```
## [1] 5.25
```

With the chosen lambda, it is possible to compute the regularised estimates for model 4 and compare them to the original estimates. As for the previous two cases, when n is small the values shrink towards zero:



The resulted RMSE after this regularisation is:

```
## [1] 0.8636144
```

Validation with the best performing model (full validation stage)

The regularised model 4 is the best performing model. The model accounts for the movie, user and day effect or bias.

By applying the algorithm to the **validation** dataset, the final achieved RMSE is:

```
## [1] 0.8647479
```

Conclusions

This work attempted to build a movie recommendation system using the 10M version of MovieLens dataset. By following a model based approach, the effects of movies, users and time were considered. Regularisations was also implemented to account for those ratings with small sample sizes.

The result of the process shows that a regularised model accounting for the movie, user and day effect achieve the highest level of prediction as shown on the table below:

method	RMSE
Model 1: Average ratings	1.0600537
Model 2: Movie effect	0.9429615

method	RMSE
Model 3: Movie + user effect	0.8646844
Model 4: Movie + user + day effect	0.8642016
Model 2 reg: Movie effect regularisation	0.9429370
Model 3 reg: Movie + user effect regularisation	0.8641362
Model 4 reg: Movie + user + day effect regularisation	0.8636144
Validation	0.8647479

The tested models fail to account for groups of, for instance, movies and groups of users variation as they may have similar rating patterns. Future work should account for this effects by using matrix factorization method. This method will find rating patterns for groups of movies, users and days. Also, future work should unclude the genre of the movie as this report did not include it.