# Systems for Data-science M1

Gengler Damien 272798

March 2021

## 1 Baseline: Prediction based on Global Average Deviation

- Question 3.1.1 :

  On average the ratings are around 3.5298. Which is higher than the average 3 by roughly 0.53. While being quite close to the average 3, we can see that is it slightly higher, it seems that user seem to rate movie higher than average.

- Question 3.1.2 :

  As we can see, not all user rate close to the global average. We find that the minimum average rating for a user is 1.4919 and the maximum is 4.8695.

  The ratio of user whose rating deviates from the global average by less than 0.5 is 0.7465. This leads us to the conclusion that indeed most user rate on average close to the global average.

- Question 3.1.3 :

  Here again not all items are rated close to the global average with the minimum rating for a movie being 1.0 and the maximum being 5.0.

  The ratio of items whose rating deviates from the global average by more that 0.5 is 0.4898. This shows that only a bit less than half of the movies are rated close to the global average.

- Question 3.1.4 :

  The prediction accuracy for the different methods are reported in Table 1.

  We can see that the baseline method performs the best between the four, followed by PerItem, PerUser and the Global method. This can be explained as the **global** method is just classifying for every user and every movie according to the overall average which is really not precise at all.

| Method | Accuracy |
|---|---|
| Global | 0.9680 |
| PerUser | 0.8501 |
| PerItem | 0.8275 |
| Baseline | 0.7669 |

Table 1: Predicted accuracy for the different methods

The **PerUser** method performs a bit better as we are taking into account the fact that each user is different and we assume that they will most likely rate close to how they rate on average.

The **PerItem** method is improving a bit the MAE again as now we are assuming that the user will probably rate the movie in a similar way as most users did before. This gives better results as we are making use of what we know about the movie from other people rather than what the user rated before. This allows us to give good ratings on certains movies even though the user never rated similar movies in the past.

Finally the **Baseline** Method gives even better results as it incorporates user bias to improve the prediction.

- Question 3.1.5 :

  The results of the benchmarking on the 10 tries are reported in Table 2. The tests were ran on a MacBook Pro (2015), with a processor 2,7GHz Intel Core i5 double core and 8 Go RAM DDR3, under macOS Catalina 10.15.4. Scala version 2.12.13.

| Method | min | max | average | std. deviation |
|---|---|---|---|---|
| Global | 198.0 | 516.0 | 259.0 | 89.7663 |
| PerUser | 705.0 | 2813.0 | 964.0 | 618.3337 |
| PerItem | 694.0 | 786.0 | 722.0 | 26.2202 |
| Baseline | 1212.0 | 1746.0 | 1380.0 | 154.0357 |

Table 2: Minimum, Maximum, Average duration in milliseconds over 10 tries, as well as the Standard deviation

The most expensive method is the Baseline. The ratio between this method and the global one is 5.3281. The tests were while also running quite a lot of other programs on the computer, when doing the benchmark several time though, we almost always found similar results.

# 2 Recommendation

- Question 4.1.1

  Here are the 5 top movies predicted by the system.

1. 814, Great Day in Harlem, 5.0
2. 1122, They Made Me a Criminal (1939), 5.0
3. 1189, Prefontaine (1997), 5.0
4. 1201, Marlene Dietrich: Shadow and Light (1996), 5.0
5. 1293, Star Kid (1997), 5.0

Actually, I do not know any of those movie. They either look like old or niche movies. I could be interested in watching Star Kid though. It really does feel like they are so high because very few people rated them and they rated really high. Also, I noticed that while the first few movie seem to be outliers, when we go further down in the listing, we start to see accurate predictions such as Usual suspects, Godfather, Shindler's list or Goodwill Hunting.

- Question 4.1.2 The first thing that came to my mind when reading the question was to use a variation of the sigmoid function to smoothly decrease the weight of the movies.

  The sigmoid function is defined as :

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

  The idea here is that we want to find to find the pair of coefficients (a, b):

$$\sigma_{scaled}(x) = \frac{1}{1 + e^{-a(x+b)}}$$

  Such that :

$$\sigma(0) \approx 0$$

$$\sigma(n) \approx 1$$

  Where n is the maximum number of ratings for a movie

  Since we cannot have sigma exactly equal to 0 or 1, we decided to approximate the values. Thus we want to solve

$$\sigma_{scale}(0) = \frac{1}{1 + e^{-ab}} = 0.001$$

$$\sigma_{scale}(n) = \frac{1}{1 + e^{-a(n+b)}} = 0.999$$

  From this we get :

$$a = \frac{-log(\frac{1}{0.001} - 1)}{b}$$

$$b = \frac{-n}{1 - \frac{log(\frac{1}{0.999} - 1)}{log(\frac{1}{0.001} - 1)}}$$

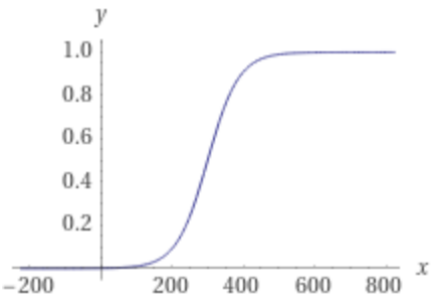


Figure 1: Scaling curve for the sigmoid with parameters (a,b) = (0.023, -300)

When evaluating for n = 600, we get a = 0.023 and b = -300 which gives us the curve in figure 1

So we can define our scaling function as :

$$\sigma_{scaled}(x) = \frac{1}{1 + e^{-0.023(x-300)}}$$

From this we can rewrite the prediction function as :

$$p_{u,i} = \bar{r}_{u,\bullet} + \bar{\hat{r}}_{\bullet,i} * scale((\bar{\hat{r}}_{\bullet,i} + \bar{r}_{u,\bullet}), \bar{r}_{u,\bullet}) * \sigma_{scaled}(\#_{Ratings_i})$$

Where $\#_{Ratings_i}$ is the number of ratings of item i.

Here are the 5 top movies predicted using the modified algorithm.

1. 127, Godfather, 4.1020
2. 174, Raiders of the Lost Ark (1981), 4.0667
3. 98, Silence of the Lambs, 4.0479
4. 313, Titanic (1997), 3.9162
5. 172, Empire Strikes Back, 3.9042

This prediction is much more accurate regarding the movies I rated, because it both includes movies that I have seen and not rated as well as movies that I plan on watching. Even if we look at the top 20 suggestions we get very accurate results.

Although the prediction are better using this modified version there are still some flaws. First the fact that the function might be decreasing too fast thus maybe reducing too fast the weights. The other problem is that since we are scaling down the prediction factor, then all ratings are going to be lower. As we can see here, the top predicted rating is only 4.1.