Meta parameters setting

We first describe how to derive the optimal parameters (μ, λ, maxIter) and then show their influence on the convergence speed and the estimation error.

1 Derive the optimal parameters

1.1 Learning rate

Setting: We derive the optimal learning rate μ for a fixed number of iteration $maxIter = 10$ and no regularizer $\lambda = 0.0$.
We first **derive the order of the optimal μ by manual testing**. We make μ take the following values (Table 1) and chose the order of μ for which there is **no divergence** and for which there is the **biggest error decrement**.

Table 1: Error divergence with respect to the order of μ

Order of μ	Divergence(Y/N)
1	Yes
10^{-1}	Yes
10^{-2}	Yes
10^{-3}	Yes
10^{-4}	No

We now know that **the order of μ is at least** 10^{-3}.
Let us now observe the convergence speed with order of μ over 10^{-3} (Figure **??**). We observe the variations of the error with respect to the step ($maxIter$) for the following order of learning rate : $\lambda = 10^{-3}, \lambda = 10^{-4}, \lambda = 10^{-5}$ respectively with the red, blue and green curves.

./images/convergenceSpeed.png

Figure 1: Error for different learning rates with respect to learning step

We observe that **as soon as the order of learning rate is lower than 10^{-4}, there is no significant difference in the error evolution.** Since the biggest the learning rate, the fastest the convergence, we now reduce our search between 10^{-5} and 10^{-4}.
We study the evolution of the final error after 10 iterations with respect to the learning rate (Figure 2).

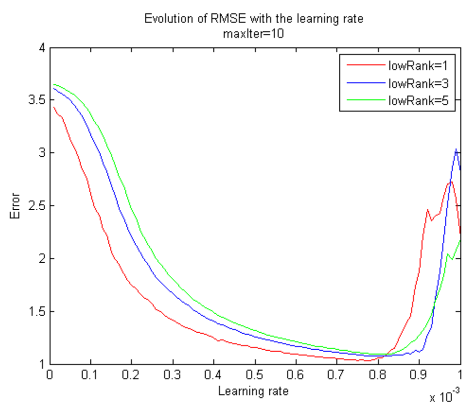


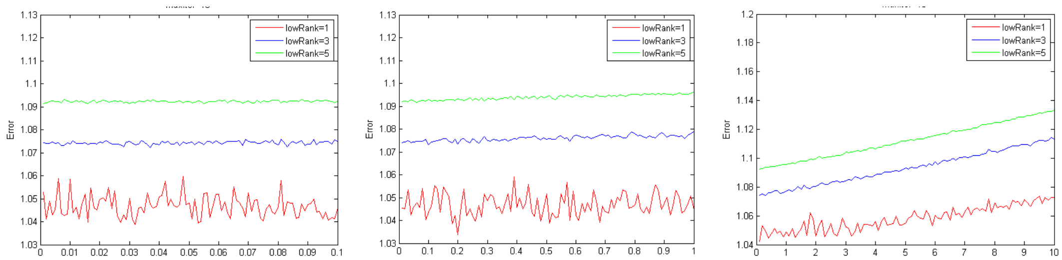
Figure 2: Evolution of RMSE with the learning rate, $maxIter = 10$

This experiment shows that **the optimum learning rate, i.e. the one for which the error is the minimum, is between $7.5 \cdot 10^{-4}$ and $8.5 \cdot 10^{-4}$ for 10 iterations**. We will need to set the number of iterations $maxIter$ but first we set the learning rate to $8.0 \cdot 10^{4}$ and will then adjust it regarding the other parameter.

1.2 Regularizer

We use a regularizer term in the error computation to avoid overfitting as the number of parameter increases and as our data set is small.
We observe the influence of the regularizer on the RMSE for different orders of regulizer (Figure 3).
Now that we minimized the RMSE with respect to the learning rate μ, **we do not want the training**

error to be too small : this would probably reflect an overfitting. We observe that the error stays acceptable when we change λ so we are going to look for the λ **that maximizes the RMSE**.



(a) Order of $\lambda : 10^{-2}$ (b) Order of $\lambda : 10^{-1}$ (c) Order of $\lambda : 1$

Figure 3: Evolution of the error with respect to the regulizer λ, $maxIter = 10$, Model complexity legend : (red,1), (blue,3), (green,5)

We observe than once the order of λ is lower than 10^{-1} it does not have that much influence on the error, i.e. it does not pressure the parameters not to overfit the data. Since **we need a λ such that the error is sensitive to it, we will choose $\lambda = 1$.**

1.3 maxIter

We increase $maxIter$ until the error goes below 1.0. We observe that for the $\lambda = 1$ and $\mu = 8 \cdot 10^{-4}$, **once $maxIter > 10$, the RMSE increases. This means that the learning rate is too high** so we adjust it by manual trial and error and we find that $\mu = 7 * 10^{-4}$ is an appropriate value for $maxIter = 20$. Rather than limiting the loop on the number of iteration, **we stop the loop once the error is below a threshold $\epsilon = 1$ and the difference between two consecutive RMSE is lower than another threshold $\alpha = 10^{-5}$** . We such setting, we get the following RMSE :

Table 2: Training and testing error with respect to model complexity

Model complexity	1	3	5
Training error	0.9993	0.9952	0.9972
Test error	1.0267	1.0256	1.0287

It is possible to get lower error on the training set but we will not go the this extent to avoid overfitting.

2 Model complexity

We now observe how accurate the rate estimation is with respect to the $lowRank$ i.e. with respect to how much parameter we use to model it. We expect that the more factors we use, the more fitting the estimation will be. As always, we should be careful not to take a too big $lowRank$ if we want to avoid overfitting. We first observe the evolution of the RMSE for $lowRank$ varying between 1 and 80.

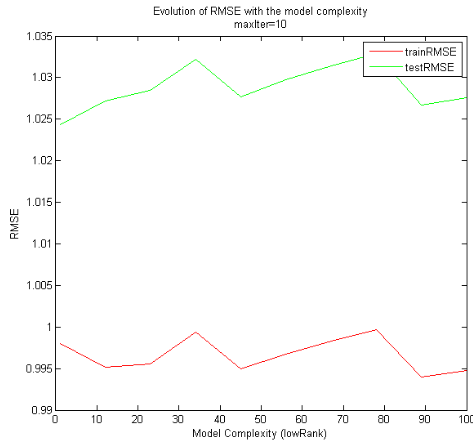


Figure 4: Evolution of RMSE with the model complexity, $maxIter = 10$

The error seems to oscillate with the model complexity but still **globally increases of 0.04% for the test error** of while the **training error decreased of 0.3%**. Even though these ratio are relatively small, they reflect the **mistake of only studying the training error that decreases with the model complexity while the real error may increase due to overfitting**. That is why we privilege low complexity model with rank of 3 or 5. In our case, we do not observe monotone decrement because since the data set is very sparse (the rating matrix M is full of 0), we need a flexible model to adjust to such type of data. It would be as if we wanted to fit a line in a point cloud with high variance. But there still exist a limit to the model complexity to avoid overfitting as the previous ratios reflect it.