Singular Value Decomposition in Education: a case study on recommending courses

Fábio Carballo
Instituto Superior Técnico – Universidade de Lisboa
Av Rovisco Pais
1049-001 Lisboa
+351 218 419 407
fabio.carballo@tecnico.ulisboa.pt

Cláudia Antunes
Instituto Superior Técnico – Universidade de Lisboa
Av Rovisco Pais
1049-001 Lisboa
+351 218 419 407
claudia.antunes@tecnico.ulisboa.pt

ABSTRACT
After bachelor, many students strive to select the masters' courses that are most likely to meet their interests. Although this decision may have a big impact on students' motivation and future achievements, usually no support is offered to contest this problem. The use of recommendation systems to suggest items to users has well-known success in several domains, and some of the most successful techniques use Singular Value Decomposition (SVD) to capture hidden latent factors in reduced dimensionality and produce high quality recommendations. In this paper, we propose to use SVD, with a contextual mapping to the educational paradigm, to capture relationships between courses grades and recommend masters' courses that are suitable to students' skills given their bachelor achievements. Our results show that using SVD to predict the masters' courses marks has potential to serve as basis for the recommendation production.

Keywords
Courses recommendation, Singular Value Decomposition

1. INTRODUCTION
The decision that students have to make on which master's courses to enroll has way more impact than it looks: this choice can have a direct effect on their academic and personal goals. A bad choice of courses may demotivate a student, which can cause the student to drop out or to not take advantage of the fullness of his capabilities. Therefore, understanding students' particularities is needed, so as to recommend courses that are not only interesting to them, but also adequate to their capabilities. Current solutions have a tendency to recommend courses based on its contents or potential interest to the students, not considering how those courses can affect students' overall academic performance [1]. Therefore, we propose the creation of a system that, with the minimal user-participation, recommends masters' courses that add value to students' academic achievements, given their bachelor path. To do it, we explore Singular Value Decomposition so as to capture hidden factors in the historical students marks and then identify the best courses to recommend

With the Netflix challenge [2][3], there was a huge trend to use *latent factor models*, in order to reveal the hidden latent features that somehow explain the observed ratings. The most successful technique in these models is *Singular Value Decomposition*, due to its accuracy and scalability. This technique factors an m x n matrix R, into three matrices as in (1),

$$R = U \times S \times V' \quad (1)$$

where U and V are two orthogonal matrices of size m x r and n x r respectively, while r represents the rank of the matrix R. Matrix S is a diagonal matrix, and its entries are stored in decreasing order of their magnitude. Each entry of matrix S represents a hidden feature and the stored value in it stands for the weight the feature has to the variance of the values on R. The sum of the values of all entries represents the total variance on matrix R. SVD has many applications of particular interest, but it is especially useful as a way to find the best rank-k approximation, R_k, to the matrix R, such that the Frobenius norm of R - R_k is minimized. The Frobenius norm ($\| R - R_k \|F$) is defined as the sum of squares of elements in R - R_k . To reduce the rank r to k, where $k < r$, one should only use the first k diagonal values of the matrix S (the singular values), and then reduce both U and V accordingly. The result is the closest k-rank approximation $R_k = U_k x S_k x V'_k$.

The usual idea, when using this technique on recommendation systems, is to use R as a users-items matrix, where m is the number of users and n the number of items. The value of each cell holds the rating that a user has given to a certain item. The idea is that after the decomposition we can calculate both the users-features and items-features spaces and use them to predict ratings. In the users-features space, $U_k\sqrt{S_k}$ – let's call it P – each row is a vector with the preference values of a user over the discovered features. On the other hand, in the items-features space, $\sqrt{S_k}V'_k$ – let's call it W - each row is a vector that represents how the item is weighted in each feature. Hence, this consists on the decomposition of the usual user-item matrix into a k-dimensional space where just the k most relevant features are taken into account: the noise in the data is reduced, and this enables the production of better quality rating predictions.

However, SVD is known for not dealing well with sparse matrices, where there are a lot of missing values. Gladly, Simon Funk found a solution to this problem [3]. He proposed to use a *gradient descent algorithm* in order to compute the best rank-k matrix approximation using only the known ratings of the user-item matrix R. This process follows the same idea than the training on *neural* networks. With the error in a prediction of user i to item j being ($R_{ij} – Rk_{ij}$), Funk's approach takes the derivative of the square of the error with respect to P_{ik} and then with respect to W_{jk}. Since R is constant, and $R_k = P x W'$ (note that in this approach P and W contain the S matrix, that usually results from the matrix decomposition), the updates for the user and item spaces, P and W then become (2) and (3), respectively:

$$P_{if}(t+1) = P_{if}(t) + learning\_rate * (R - R_k)_{ij} * W_{jf}(t) \quad (2)$$

$$W_{jf}(t+1) = W_{jf}(t) + learning\_rate * (R - R_k)_{ij} * P_{if}(t) \quad (3)$$

In summary, the final solution of this learning problem is the combination of feature weights on both P and W such that the

error in the approximation R_k is minimized. This solution is determined iteratively, as the gradient of the error function is computed at each iteration step. Note that in this approach all features vectors are initialized with the global rating average along with some introduced random noise.

2. SVD-BASED COURSES RECOMMENDATION

As we stated above, we aim for exploring SVD to recommend masters' courses to students given only their bachelor's courses marks. Hence, we must start of an historical record over triplets in the form of *<Student, Course, Mark>* into a structure that SVD can explore. As we have seen, SVD makes use of a matrix *R* that holds knowledge over the ratings that users gave to items. In usual representations, users are placed in rows and items in columns, and each cell R_{ij} in the matrix corresponds to the rating that the i^{th} user attributes to the j^{th} item.

As a first step to map our problem to the educational context we must transform our historical students' marks record into a matrix *R* that holds our knowledge over students' capabilities in each course taken. Our proposal is that matrix *R* will have students represented on rows and courses on columns, and each entry R_{ij} will be filled with the mark obtained by the i^{th} student on the j^{th} course. When students didn't enroll, the mark is the zero value. This is a natural mapping, as we want to recommend the courses with the predicted best marks, while having the constraint of recommending only a subset of the courses, the masters' courses. Our idea is to apply SVD to the matrix described above, so as to predict the marks of every student on all masters' courses and then use those predictions to recommend a specific set of courses. We will use Funk's *gradient descent* algorithm to calculate SVD, and, consequently, produce both the users and courses spaces. Applying Funk's gradient descent to the student-courses matrix *R* (with *N* number of features to discover) we get matrices *P* and *W*. Matrix *P* represents the user features dimensional space, where row *i* stands for student *i* features vector, which relates the student with each of the *N* features. Likewise, each row of matrix *W* shows how each course is related to each one of the *N* features. The product *PW'* constitutes a *N*-rank approximation of the original matrix *R*.

At this moment, we can use matrices *P* and *W* to predict the students' masters' marks. The predicted mark of a student *i* on course *j* corresponds to the dot product between the i^{th} row of *P* and the j^{th} row of *W*. This dot product represents how the student is related with the course according the several features. The predicted mark may need some bound restriction in order to be between acceptable values. Finally, we can just recommend the *N* masters' courses with the best-predicted marks.

3. EXPERIMENTAL RESULTS

We tested our approach with data from a bachelor and a masters program at Instituto Superior Técnico, Universidade de Lisboa, in Portugal. This dataset describes 9149 courses' results achieved by 251 students on both bachelor and masters. The marks scale goes from 0 to 20, where 10 is the minimum grade that a student must achieve to be approved on any course.

To evaluate our results we follow the belief that the overall quality of the recommendations, independently of the method used to produce them, depends a lot on the quality of our predictions. To have a comparison to our grades' predictions

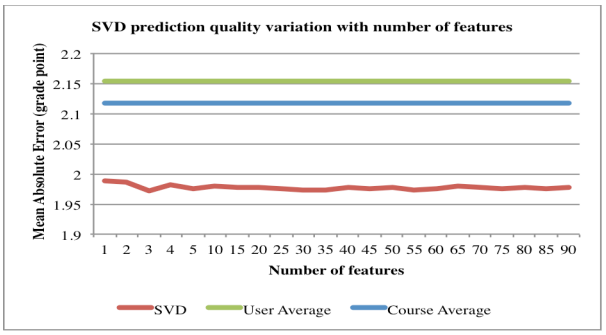

Figure 1 –MAE with the variation of the number of features and comparison with baselines

results we used two baselines. The first sets the predicted mark of each student as his average mark on bachelor. The second baseline uses the average mark achieved in each masters' course. To do our prediction experiment, we started by constructing the 251 x 94 students-courses matrix *R*. We then applied Funk's SVD to produce both the students and courses features spaces and predicted every student's marks on all masters' courses. The achieved results in terms of the Mean Absolute Error (MAE) can be seen on Figure 1, and it is clear that our SVD approach has a smaller error than any of the baselines. In average, our predictions are 1.97 points deviated from the real mark, while both baselines have error values near 2.15. Hence, our predictions sustain an above average basis from where to recommend masters' courses to students.

We did another experiment to see how the recommendations may affect students' masters' average mark. In this case we also used two baselines approaches: one recommends the most frequented courses in the historical training data while the other recommends the courses with best average mark on the same data. Table 1 shows the average mark of followed recommendations on the test data for all the approaches. We can see that the average mark achieved with the recommendations of our SVD approach is better than any of the baselines.

Table 1 – Average grade on followed recommendations.

Best Grades	Most Popular	SVD
13.6	13.4	14.6

ACKNOWLEDGMENTS

This work is supported by Fundação para a Ciência e Tecnologia under research project *educare* (PTDC/EIA-EIA/110058/2009).

REFERENCES

[1] Romero, C. and Ventura, S. 2010. Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 40, 6, 601-618.

[2] Koren, Y., Bell, R., and Volinsky, C. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer*, 42, 30-37.

[3] Funk, S. *Netflix Update: Try this at home*. http://sifter.org/~simon/journal/20061211.html, 2006.