Regulared SVD e Improved Regulared SVD: Uma análise empírica

Lívia de Azevedo[1], **Gustavo Ebbo**[1], **Ivo Paiva**[1]

[1]Departamento de Ciência da Computação –
Universidade Federal Rural do Rio de Janeiro (UFRRJ)
R. Governador Roberto Silveira S/N – Nova Iguaçu –
Rio de Janeiro – RJ – Brasil

{livia_de_azevedo,gustavoebbo,ivopaiva}@ufrrj.br

Abstract. *This report proposes an experimental analysis of the two algorithms: Regulared SVD (RSVD) and Improved Regulared SVD (IRSVD). The principle of the analysis is based on the experimental results when we are changing their parameters, using the Movie Lens database. Tests have been done with diferent values of instances to validate the efficiency of the algorithms to make ratings predicts for some elements in this base. The results of the experiments and conclusions have been presented.*

Resumo. *Este relatório propõe uma análise experimental dos algoritmos Regularized SVD (RSVD) e Improved Regularized SVD (IRSVD). O foco desta análise consiste na observação do comportamento dos algoritmos variando seus parâmetros, usando a base de dados do Movie Lens. Para isso, foram feitos testes com diferentes instâncias para validar a eficiência dos algoritmos para a realização de previsões de notas para alguns elementos desta mesma base. Foram apresentados os resultados dos experimentos e conclusões a respeito.*

1. Introdução

A área dos Sistemas de Recomendação tem ganhado bastante enfoque nos últimos anos. Devido sua intensa utilização e importância, pesquisadores de áreas correlatas realizaram pesquisas sobre o assunto com o *Group Lens*, que desenvolveu um site chamado *Movie Lens*, e obtiveram as avaliações de seus usuários para os filmes disponíveis, deixando esses dados disponíveis para que qualquer pesquisador possa realizar experimentos, o que faremos neste relatório. Um outro exemplo de destaque é o *Netflix prize*, que ofereceu um prêmio no valor de $1 milhão de dólares a quem melhorasse a previsão de notas dos usuários da *Nexflix* em 10%. Em 21 de Setembro de 2009, um grupo conseguiu completar o desafio, melhorando a previsão em 10,06%. Durante esse desafio, surgiu um dos algoritmos apresentados neste relatório, o RSVD e posteriormente uma especialização do mesmo, o IRSVD.

O objetivo deste relatório é prover uma análise no comportamento de ambos os algoritmos quando variamos a quantidade de variáveis latentes, seguindo algumas condições para cada algoritmo. Nas próximas seções serão abordados: a metodologia, os detalhes de implementação dos algoritmos, os resultados e a conclusão.

2. Sobre os algoritmos

2.1. Regulared SVD

O RSVD foi proposto por Simon Funk para o *Netflix prize*, utilizando uma estratégia contendo variáveis latentes e o método do gradiente descendente. A previsão de uma nota $\hat{r}_{ui}$ para um usuário u dado um item i é calculada da seguinte forma:

$$\hat{r}_{ui} = q_i^T p_u$$

onde $q, p \in \mathbb{R}^f$ e f é o valor da dimensão de variáveis latentes consideradas.

Variáveis latentes podem ser definidas como sendo variáveis que não são observáveis diretamente, mas precisam de outros fatores para que possam ser identificadas. De acordo com [Koren and Bell 2011], cada elemento do vetor q_i representa o nível de pertinência de uma dada característica para este item. De modo análogo, o vetor p_u representa o nível de interesse deste usuário com relação a esta característica.

O erro de aprendizado do modelo consiste na seguinte fórmula:

$$\min_{q^*,p^*} \sum_{(u,i)\in K} (r_{ui} - q_i^T p_u)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2)$$

onde λ é o fator de regularização, onde seu valor ótimo - e o definido aqui - é $\lambda = 0.02$ e K é o conjunto de treinamento considerado.

O método estocástico do gradiente descendente modelado para este modelo consiste nas seguintes atualizações, realizadas para cada elemento do conjunto de treinamento:

$$q_i \leftarrow q_i + \gamma * (e_{ui} * p_u - \lambda * q_i)$$
$$p_u \leftarrow p_u + \gamma * (e_{ui} * q_i - \lambda * p_u)$$

onde e_{ui} é o erro de predição da nota, dado por $e_{ui} \stackrel{def}{=} r_{ui} - \hat{r}_{ui}$ e γ é a taxa de aprendizado do modelo, usado com o valor ótimo já encontrado como $\gamma = 0.001$.

Os detalhes dados pelo próprio Simon Funk sobre o algoritmo podem ser encontrados em: http://sifter.org/ simon/journal/20061211.html

2.2. Improved Regulared SVD

O IRSVD é uma melhora do algoritmo do RSVD, que adiciona o fator denominado *baseline predictors*, sendo abreviado como *bias* e a média global do conjunto de treinamento. *Bias* é a representação da tendência do usuário a dar notas mais altas para os itens e a tendência do item para receber notas mais altas, observados de forma independente.[Koren and Bell 2011]. Assim, prevemos a nota do usuário da seguinte forma:

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u$$

onde b_i e b_u são os *bias* do item e do usuário, respectivamente e μ é a média global do conjunto de treinamento analisado.

Para o aprendizado do modelo contendo estes parâmetros, seguindo a ideia expressa em 2.1, buscamos minimizar a seguinte fórmula:

$$\min_{b^*,q^*,p^*} \sum_{(u,i)\in K} (r_{ui} - \mu - b_i - b_u - q_i^T p_u)^2 + \lambda(b_i^2 + b_u^2 + \|q_i\|^2 + \|p_u\|^2)$$

sendo o valor ótimo de λ e que foi definido aqui igual a 0.02

O método estocástico do gradiente descendente modelado para este modelo consiste nas seguintes atualizações, realizadas para cada elemento do conjunto de treinamento:

$$b_u \leftarrow b_u + \gamma * (e_{ui} - \lambda * b_u)$$
$$b_i \leftarrow b_i + \gamma * (e_{ui} - \lambda * b_i)$$
$$q_i \leftarrow q_i + \gamma * (e_{ui} * p_u - \lambda * q_i)$$
$$p_u \leftarrow p_u + \gamma * (e_{ui} * q_i - \lambda * p_u)$$

o valor ótimo e que foi definido aqui de γ é $\gamma = 0.005$.

3. Metodologia

Para a realização das análises e dos testes, usamos uma base de dados já conhecida e utilizada na literatura, a do *Movie Lens* (http://grouplens.org/datasets/movielens/) a qual possui um total de 100000 avaliações. Há dois casos de testes: um que considera a base aleatoriamente dividida em 80% para treinamento e 20% de teste e outro que usa uma divisão disponível no *Group Lens* na qual o conjunto de teste possuiu exatamente dez notas de cada usuário existente, dividindo 90,57% para treino e 9,43% para teste. Este último teste representa um conjunto que tem uma característica conhecida, que não seja aleatória.

Os vetores que representam q_i e p_u foram gerados aleatoriamente com valores entre 0 e 1, representando a imprevisibilidade das preferências. Para cada teste variando a quantidade de variáveis latentes, foram utilizados os mesmos valores que foram gerados aleatoriamente, mas levando em consideração a quantidade de variáveis latentes que foram utilizadas.

Os *bias* do IRSVD foram inicializados com zero para todos os usuários e itens, pois desconhecemos o comportamento de ambos os elementos para definir algum valor inicial para representar as tendências de ambos.

4. Resultados

Para a realização dos testes, foi considerado um critério de parada dos algoritmos em 0.1 (que é calculado na execução como a diferença entre o erro de aprendizado anterior com o atual a cada iteração). Cada um dos dois casos de teste foram executados 10 vezes, sendo aumentado a quantidade de variáveis latentes em 10 para cada execução, começando de 10 e indo até 100. A escolha desses intervalos foi feita aleatoriamente. A métrica de erro utilizada para a avaliação foi o *Root Mean Squared Error*(RSME), expressa abaixo:

$$RSME = \sqrt{\frac{\sum_{u,i\in test\_set} (r_{ui} - \hat{r}_{ui})^2}{|test\_set|}}$$

onde $test\_set$ é o conjunto de teste para avaliação.

Os resultados são expressos nas figuras 1 e 2. O valores exatos estão representados na tabela 1.

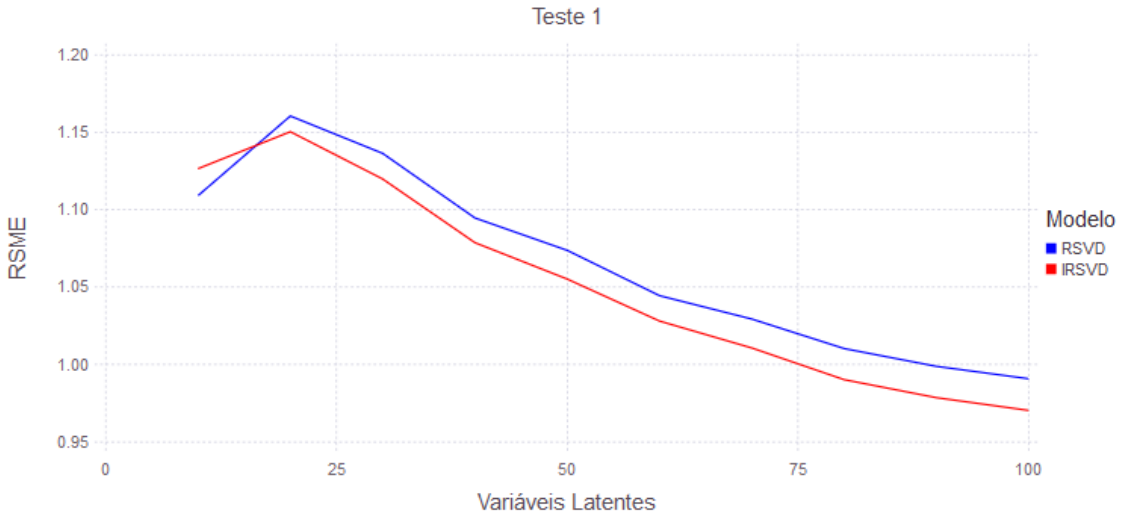

Figure 1. Gráfico para o caso de Teste 1

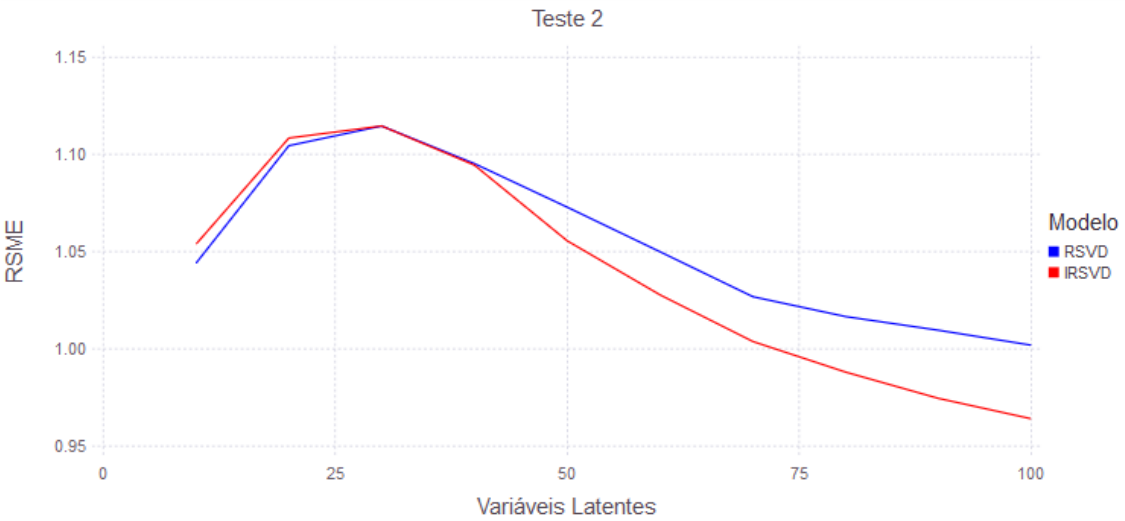

Figure 2. Gráfico para o caso de Teste 2

5. Conclusão e trabalhos futuros

Conforme esperado, o algoritmo do IRSVD apresenta um melhor desempenho com relação ao RSVD e também à medida que a quantidade de variáveis latentes consideradas no modelo aumenta, a partir de um certo momento, a taxa de acurácia dos modelos aumentam, o que comprova o senso comum de, quanto mais características dos usuários e itens são conhecidas, melhor a acurácia. Lembrando que a forma como é inicializada as variáveis de ambos os algoritmos (os *bias*, q e p) determina se os modelos podem alcançar um mínimo global com relação ao erro de aprendizado, ou seja, uma acurácia máxima. Assim, como trabalho futuro, existe a possibilidade de avaliar outras formas mais inteligentes de inicialização para que se alcance esse objetivo.

Quant. de var. latentes	T1-RSVD	T1-IRSVD	T2-RSVD	T2-IRSVD
10	1.1090	1.1264	1.0440	1.0538
20	1.1604	1.1503	1.1044	1.1084
30	1.1363	1.1198	1.1146	1.1146
40	1.0946	1.0787	1.0952	1.0944
50	1.0737	1.0551	1.0728	1.0555
60	1.0444	1.0280	1.0499	1.0277
70	1.0293	1.0106	1.0268	1.0037
80	1.0103	0.9902	1.0165	0.9880
90	0.9988	0.9786	1.0095	0.9745
100	0.9908	0.9703	1.0018	0.9640

Table 1. Valores aproximados obtidos de RSME para cada caso de teste

References

Koren, Y. and Bell, R. (2011). Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 145–186. Springer.