4
Experimentos

4.1.
Conjunto de Dados de Anúncios

Obter bons exemplos é uma das grandes dificuldades ao se trabalhar com Sistemas de Recomendação. Em decorrência disto, não encontramos dados de anúncios publicamente disponíveis. Inclusive tentamos, sem sucesso, acesso a base de dados do Live Search[7], disponibilizada por uma RFP (*Request For Proposal*)[8]. Finalmente, contornamos o problema da escassez de exemplos de anúncios através da geração de dados artificiais, descritos na seção 4.1.1.

[7] http://www.live.com – último acesso em julho de 2008.

[8] http://research.microsoft.com/ur/us/fundingopps/RFPs/BeyondSearch_RFP.aspx - último acesso em julho de 2008.

4.1.1. Geração da Base Artificial

Para gerar os dados artificiais, requisitamos, à Universidade de Lisboa, o WPT 03[9], que contém os registros de seis meses de consultas submetidas ao TUMBA![10]. Em seguida, submetemos cada uma dessas consultas ao Google, obtendo a ordem dos anúncios a elas associados. Assim, construímos uma matriz *P* na qual cada linha *i* representa uma consulta, cada coluna *j* representa um anúncio e cada elemento p_{ij} representa a posição do anúncio *i* na consulta *j*, estando preenchidos apenas os elementos cujo anúncio é retornado como resultado pelo Google. Como possuímos somente a posição e não os CTRs reais, cada elemento p_{ij} conhecido foi substituído por uma estimativa. Utilizamos os valores apresentados na Tabela 1 (Brooks, 2004), que indica o CTR esperado dos anúncios exibidos nas 10 primeiras posições em função do CTR da primeira posição. Após essa conversão, passamos a ter uma matriz *C*, que possui 55.747 linhas, representando as consultas, 50.608 colunas, representando os anúncios e 182.090 elementos conhecidos, correspondentes aos CTRs relativos (referenciados apenas por "CTR" daqui por diante).

Tabela 1: Conversão da Posição em CTR Relativo.

Posição	CTR Relativo	Posição	CTR Relativo
1	100%	6	50,2%
2	77,4%	7	39,7%
3	66,6%	8	34,3%
4	57,4%	9	26,0%
5	52,9%	10	26,3%

9 O WPT 03 é um recurso criado pelo Grupo XLDB (http://xldb.di.fc.ul.pt – último acesso em julho de 2008), e disponibilizado pela Linguateca-XLDB (http://www.linguateca.pt – último acesso em julho de 2008).

10 máquina de busca portuguesa disponível em: http://www.tumba.pt (último acesso em junho de 2008).

4.2. Conjunto de Dados de Filmes

Já no caso da recomendação de filmes, os exemplos são disponibilizados pela competição Netflix. O interessante mencionar que, neste caso, dispomos de tantos exemplos que vamos trabalhar somente com parte dos dados. Na seção 4.2.1, detalhamos o processo de segmentação.

4.2.1. Segmentação dos Dados

O conjunto de treino, já fornecido pela competição Netflix, possui 480.189 linhas, representando os clientes e 17.770 colunas, representando os filmes. Trata-se de uma matriz com 8.532.958.530 elementos onde 100.000.000 são conhecidos, ou seja, 1.1%. Apesar dos dados dos filmes serem menos esparsos que os de anúncios, sua quantidade dificulta bastante a manipulação.

Nosso intuito não é ganhar a competição Netflix, objetivamos somente utilizar seus dados para validar o algoritmo de *Boosting*. Lidar com o conjunto completo de exemplos do Netflix é complicado, trabalhoso e demanda uma máquina com grande poder de processamento e memória. A solução foi segregar os dados para agilizar nossos experimentos. Outro projeto do laboratório LEARN, o LearnFlix[11], já havia enfrentado o mesmo problema. A equipe do LearnFlix processou os dados com o algoritmo K-Médias (*K-Means*), proposto por (MacQueen,1967), de vinte grupos. A regra de comparação foi a distância euclidiana. Gentilmente, a equipe do LearnFlix nos cedeu duas das vinte partes do conjunto completo, cujas estatísticas se encontram na Tabela 5.2.

[11] Constituído, especificamente, para realizar experimentos com os dados da competição Netflix. Equipe composta por: Ruy Luiz Midilú, Roberto Cavalcante, Cícero Nogueira e Julio Duarte.

Os conjuntos de treino e teste são disjuntos nos clientes, divisão esta realizada pela própria Netflix. Note a grande diferença quantitativa entre os dados desta recomendação em comparação com a de anúncios.

Tabela 2: *Clusters* Utilizados na Recomendação de Filmes.

Cluster	**Quantidade de Linhas (Clientes)**	**Quantidade de Colunas (Filmes)**	**Quantidade de Notas no Treino**	**Quantidade de Notas no Teste**
1	15.239	14.283	1.967.022	44.089
2	13.872	17.761	3.209.262	40.420

4.3. Metodologia e Métricas de Avaliação

Na recomendação de anúncios, utilizamos um *repeated holdout* (seção 2.2) de 20 iterações. Para cada iteração, o conjunto de dados é dividido em 95% dos CTRs para treino e 5% para teste. Cada um dos CTRs que compõem o conjunto de teste obedece aos seguintes critérios:

i. Deve existir pelo menos outro CTR na mesma linha e outro CTR na mesma coluna, participando do conjunto de treino.
ii. Não deve existir outro CTR na mesma linha e nem outro CTR na mesma coluna, participando do conjunto de teste.

Aproveitamos as métricas utilizadas por Cavalcante & Milidiú (2008) para comparação com seus resultados. A primeira métrica é o ***Rooted Mean Squared Error*** (**RMSE**) da predição do CTR, equivalendo à função objetivo do algoritmo de aprendizado utilizado, o qual busca minimizar o erro quadrático. Tal métrica indica quanto os valores preditos, na média, distam dos valores reais. A penalização para o erro é quadrática, conferindo maior peso aos erros grandes.

Visando medir os erros na posição de impressão do anúncio, ordenamos, decrescentemente para cada consulta, os CTRs preditos e avaliamos o quanto essa nova ordenação se aproxima da real. Definimos então a **Precisão do Posicionamento** (**PP**), que indica a fração dos anúncios colocados exatamente na posição que ocupavam originalmente na ordenação real, obtida através do Google. A última métrica utilizada, o **Erro Absoluto Médio da Posição** (**EAMP**), indica a média dos valores absolutos das diferenças entre a ordenação dos anúncios de CTRs preditos e a ordenação real.

Na recomendação de filmes, não precisamos da validação cruzada, pois, além de existir fatura de exemplos, os conjuntos de treino e teste já estão devidamente separados. A métrica utilizada é o **RMSE** da predição da nota.

4.4. Resultados do algoritmo MM para Anúncios

A fim de prover um sistema de referência para comparação, utilizamos o algoritmo de **Média das Médias** (**MM**), que fornece como predição para um CTR desconhecido c_{ij} a média das médias de todos os valores c_{ik} conhecidos contidos na mesma linha e de todos os valores c_{kj} conhecidos contidos na mesma coluna. A Tabela 3 mostra os resultados.

Tabela 3: Resultados do Sistema de Referência.

RMSE	**PP**	**EAMP**
0,1584	40,38%	1,1103

4.5. Resultados do algoritmo FM para Anúncios

Os parâmetros para o algoritmo de Fatoração de Matrizes (FM), nos experimentos descritos a seguir, são os mesmos fixados por Cavalcante & Milidiú (2008):

- Taxa de aprendizado η: 0,1.
- Fator de regularização λ: 0,01.
- Atributos latentes: 1.

A Tabela 4 exibe os resultados dos experimentos onde o algoritmo foi avaliado diversas vezes, variando o número de épocas a serem utilizadas no treinamento. Os valores em negrito destacam o melhor resultado para cada uma das três métricas de avaliação utilizadas.

Tabela 4: Resultados do FM.

Épocas de Treinamento	**RMSE**	**PP**	**EAMP**
1	0,3441	29,53%	1,6447
2	0,3161	30,51%	1,6091
5	0,2643	32,40%	1,5176
10	0,2195	35,53%	1,3849
20	0,1830	38,84%	1,2376
50	0,1545	43,09%	1,0533
100	**0,1482**	45,53%	0,9612
200	0,1514	47,46%	0,8981
500	0,1667	49,06%	0,8597
1.000	0,1761	**49,47%**	**0,8515**

O experimento com 100 épocas produziu o menor RMSE, 0.1482, superando o algoritmo MM em, aproximadamente, 6.5%. Já no caso das métricas baseadas nas posições da ordenação reconstruída, o experimento com a maior quantidade de épocas de treinamento produziu o melhor resultado. Mais especificamente, o experimento com 1.000 épocas de treinamento obteve maior PP e menor EAMP, 49.47% e 0.8515, respectivamente. Tais números superam o algoritmo MM em 9.06% para a PP e, aproximadamente, 23.5% para a EAMP. O gráfico da Figura 19 mostra que a melhoria no valor dessas duas métricas é relativamente pequena após a execução de 500 épocas de treinamento.

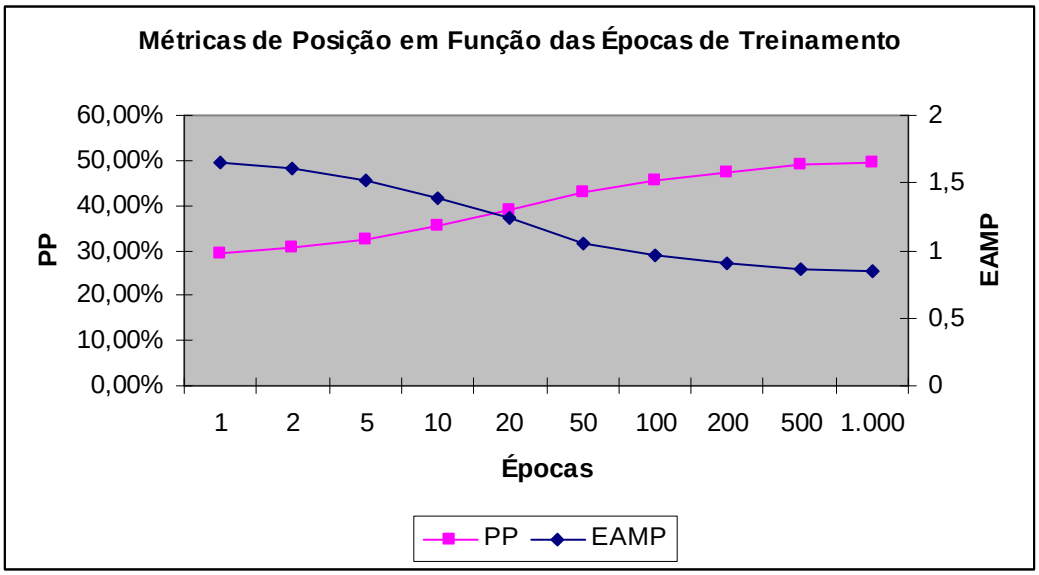

Figura 19: Métricas de posição em função das épocas de treinamento.

4.6. Resultados do *Boosting* com algoritmo FM para Anúncios

Com o intuito de calibrar o fator de suavização do AdaBoost.RS para a recomendação de anúncios, fixamos o seguintes parâmetros:

- Taxa de aprendizado η: 0,1.
- Fator de regularização λ: 0,01.
- Atributos latentes: 1.
- Épocas de treinamento: 100.
- Quantidade de preditores: 5.

Escolhemos 5 para a quantidade de preditores por ser um dos valores centrais dos nossos experimentos com anúncios, que variam entre {2, 5, 10, 20}. Já a opção pelas 100 épocas de treinamento ocorre devido ao seu bom RMSE, demonstrado nos resultados da Tabela 4. A Tabela 5 apresenta os valores obtidos durante o processo de calibração, com destaque em negrito para as melhores marcas encontradas.

Tabela 5: Calibração do Fator de Suavização para Anúncios.

Fator de Suavização	**RMSE**	**PP**	**EAMP**
0	0,1528	**46,12%**	**0,9303**
1	0,1489	45,94%	0,9367
2	**0,1454**	45,26%	0,9454
3	0,1460	46,00%	0,9448
4	0,1490	45,97%	0,9392
5	0,1482	46,01%	0,9419
6	0,1482	45,83%	0,9495
7	0,1485	45,27%	0,9496
8	0,1489	45,66%	0,9535
9	0,1480	45,72%	0,9499

O gráfico da Figura 20 mostra o comportamento do RMSE, métrica utilizada como guia para a escolha do fator de suavização ideal.

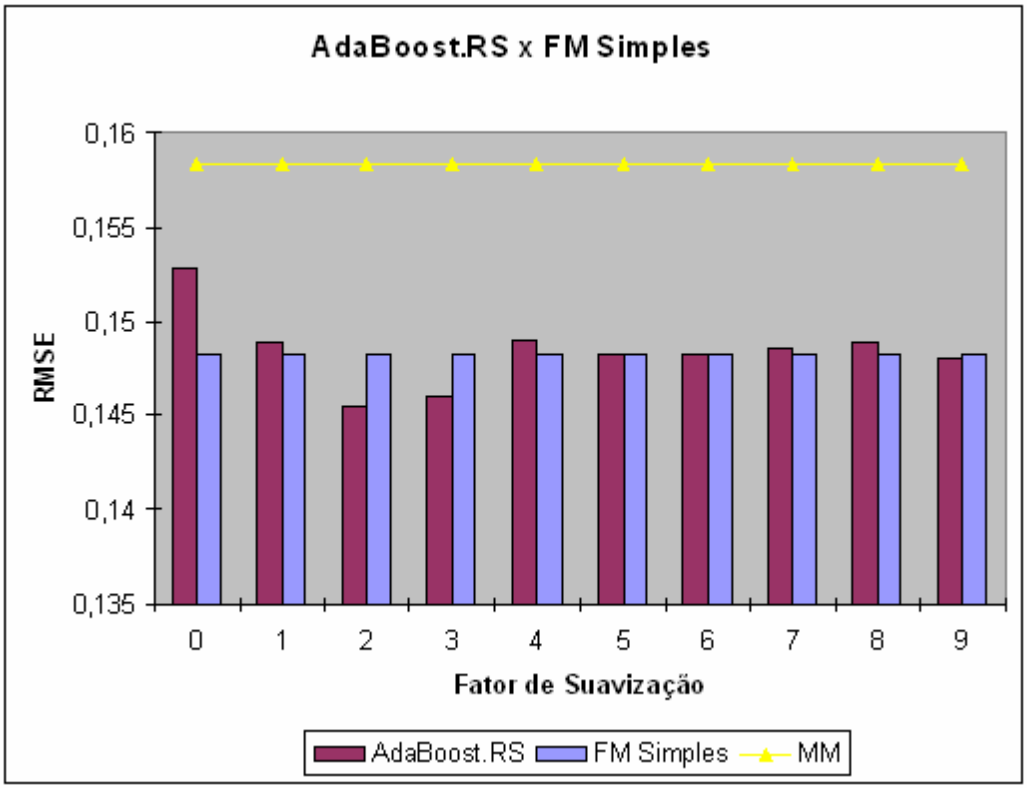

Figura 20: Calibrando o AdaBoost.RS para anúncios.

De posse do melhor fator de suavização, 2 no caso, aplicamos o AdaBoost.RS ao FM. A Tabela 6 apresenta os resultados com as melhores marcas destacadas. É interessante salientar que, mesmo com uma base de dados artificial, o AdaBoost.RS se mostrou capaz de melhorar o FM, que por si só já é bastante eficiente, afinal, superou muito bem os resultados do MM.

Tabela 6: Resultados do *Boosting* aplicado ao FM.

Épocas de Treinamento	**Quantidade de Preditores**	**RMSE**	**PP**	**EAMP**
1	1	0,3441	29,53%	1,6447
	2	0,3379	30,06%	1,6029
	5	0,3354	30,14%	1,5963
	10	0,3348	30,02%	1,5966
	20	0,3349	31,26%	1,5619
2	1	0,3161	30,51%	1,6091
	2	0,3062	31,26%	1,5619
	5	0,3038	31,18%	1,5541
	10	0,3027	31,07%	1,5496
	20	0,3029	31,29%	1,5509
5	1	0,2643	32,40%	1,5176
	2	0,2511	33,44%	1,4515
	5	0,2468	33,89%	1,4285
	10	0,2458	33,67%	1,4291
	20	0,2462	33,72%	1,4312
10	1	0,2195	35,53%	1,3849
	2	0,2081	36,37%	1,3222
	5	0,2051	36,82%	1,3028
	10	0,2045	36,84%	1,3023
	20	0,2049	36,82%	1,3012
20	1	0,1830	38,84%	1,2376
	2	0,1759	39,56%	1,1792
	5	0,1735	40,02%	1,1605
	10	0,1734	39,90%	1,1618
	20	0,1736	39,91%	1,1668
50	1	0,1545	43,09%	1,0533
	2	0,1519	43,25%	1,0380
	5	0,1508	43,47%	1,0220
	10	0,1509	43,53%	1,0247
	20	0,1507	43,68%	1,0215
100	1	0,1482	45,53%	0,9612
	2	0,1460	46,81%	0,9210
	5	0,1454	45,26%	0,9454
	10	0,1446	46,75%	0,9258
	20	**0,1436**	45,93%	0,9380
200	1	0,1514	47,46%	0,8981
	2	0,1488	47,64%	0,8845
	5	0,1498	48,25%	0,8822
	10	0,1505	47,91%	0,8903
	20	0,1495	47,77%	0,8907
500	1	0,1667	49,06%	0,8597
	2	0,1654	48,51%	0,8693
	5	0,1666	48,87%	0,8554
	10	0,1670	48,87%	0,8577
	20	0,1624	49,58%	0,8567
1.000	1	0,1761	49,47%	0,8515
	2	0,1740	49,47%	0,8540
	5	0,1751	49,78%	0,8331
	10	0,1731	49,46%	0,8433
	20	0,1731	**50,27%**	**0,8297**

Note que o *Boosting* sempre melhora os resultados do preditor simples. A seguir, na Figura 21, apresentamos o gráfico de decaimento do RMSE para 100 épocas. As melhores marcas para o PP e EAMP foram obtidas pelo experimento com 1000 épocas. Repare na curiosa queda do desempenho com o comitê de 10 preditores, exibida no gráfico da Figura 22.

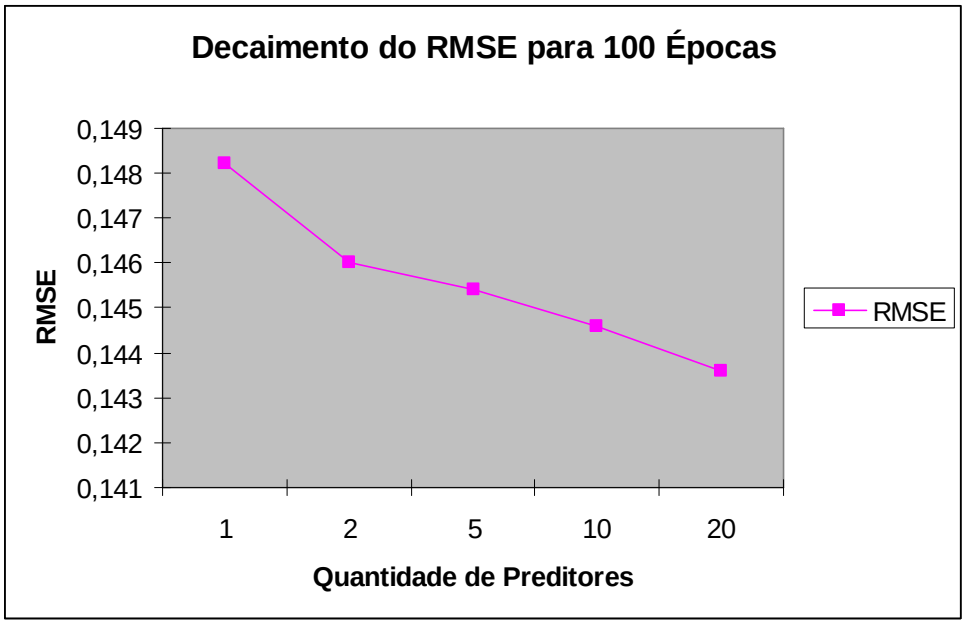

Figura 21: Melhora de **3%** no RMSE em relação ao FM.

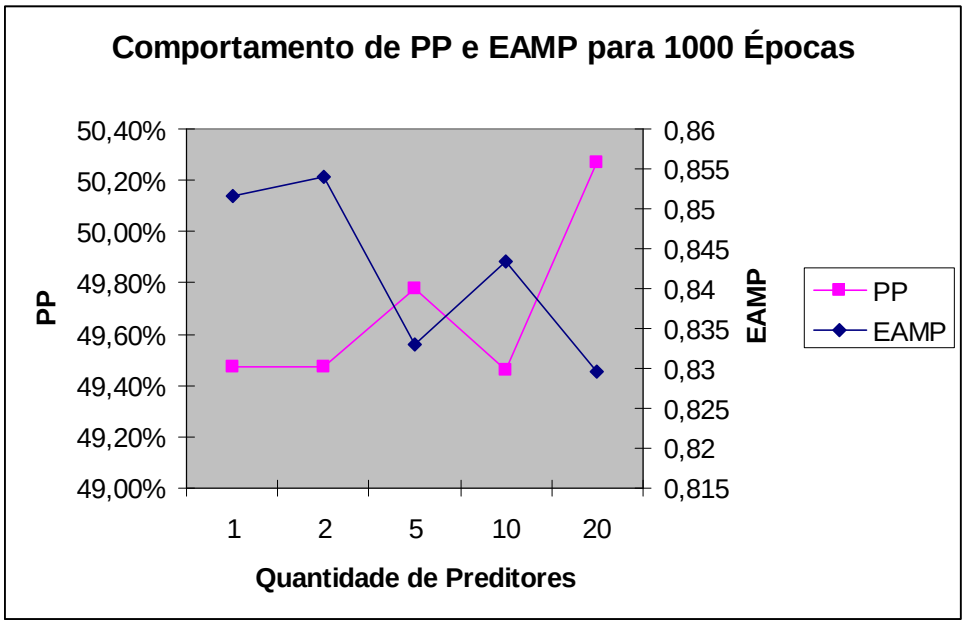

Figura 22: Melhora de 2,5% no EAMP e 0,8% na PP em relação ao FM.

4.7. Resultados do algoritmo MM para Filmes

O algoritmo de **Média das Médias** (**MM**) também será utilizado como sistema de referência para as comparações dos resultados na recomendação de filmes. Ele fornece, como predição para uma nota, a média das médias de todos os valores conhecidos contidos na mesma linha e de todos os valores conhecidos contidos na mesma coluna. A Tabela 7 mostra o seu resultado nos dois *clusters* da Netflix.

Tabela 7: Resultados do MM sobre os *Clusters*.

Cluster	**RMSE**
1	1,1011
2	1,0600

4.8. Resultados do algoritmo FM para Filmes

Os parâmetros da recomendação de filmes diferem da recomendação de anúncios. Como a calibração do FM está fora do nosso escopo de trabalho, requisitamos, aos membros da equipe Learnflix, a parametrização utilizada nos seus experimentos. Os parâmetros fixados para os resultados apresentados nas Tabelas 8 e 9 são:

- Taxa de aprendizado η: 0,1.
- Fator de regularização λ: 0,01.
- Épocas de treinamento: 15.

Tabela 8: Resultados do FM sobre o *Cluster* 1.

Atributos Latentes	**RMSE**
1	1,1239
2	1,1211
5	1,1092
10	1,1020
20	**1,0972**

Tabela 9: Resultados do FM sobre o *Cluster* 2.

Atributos Latentes	**RMSE**
1	1,0554
2	1,0486
5	1,0333
10	1,0193
20	**1,0129**

4.9. Resultados do Boosting com algoritmo FM para Filmes

Para calibrar o fator de suavização, utilizamos um comitê de 5 preditores. Este número foi escolhido pelo mesmo motivo da calibração dos anúncios (seção 4.6). A Tabela 10 mostra o comportamento do RMSE na medida em que aumentamos o fator de suavização, destacando o melhor resultado. Utilizamos o *cluster* 1 para a calibração, pois, segundo os resultados do MM (seção 4.7), ele é mais difícil que o *cluster* 2. Os parâmetros aqui fixados são:

- Taxa de aprendizado η: 0,1.
- Fator de regularização λ: 0,01.
- Atributos latentes: 20.
- Épocas de treinamento: 15.
- Quantidade de preditores: 5.

Tabela 10: Calibração do Fator de Suavização para Filmes.

Fator de Suavização	**RMSE**
0	1,1240
1	1,1097
2	1,1053
3	1,1005
4	1,0984
5	1,0988
6	1,0976
7	1,0977
8	1,0973
9	1,0966
10	**1,0959**
11	1,0965
12	1,0968
13	1,0965
14	1,0969

A Figura 23 mostra o decaimento do RMSE na medida em que aumentamos o fator de suavização. O AdaBoost.RS alcança, com fator 10, uma sutil melhora de 0,11% sobre o FM simples e 2,5% sobre o AdaBoost.R2 original. Note que o fator 3 já supera o resultado obtido pelo sistema de referência (algoritmo MM).

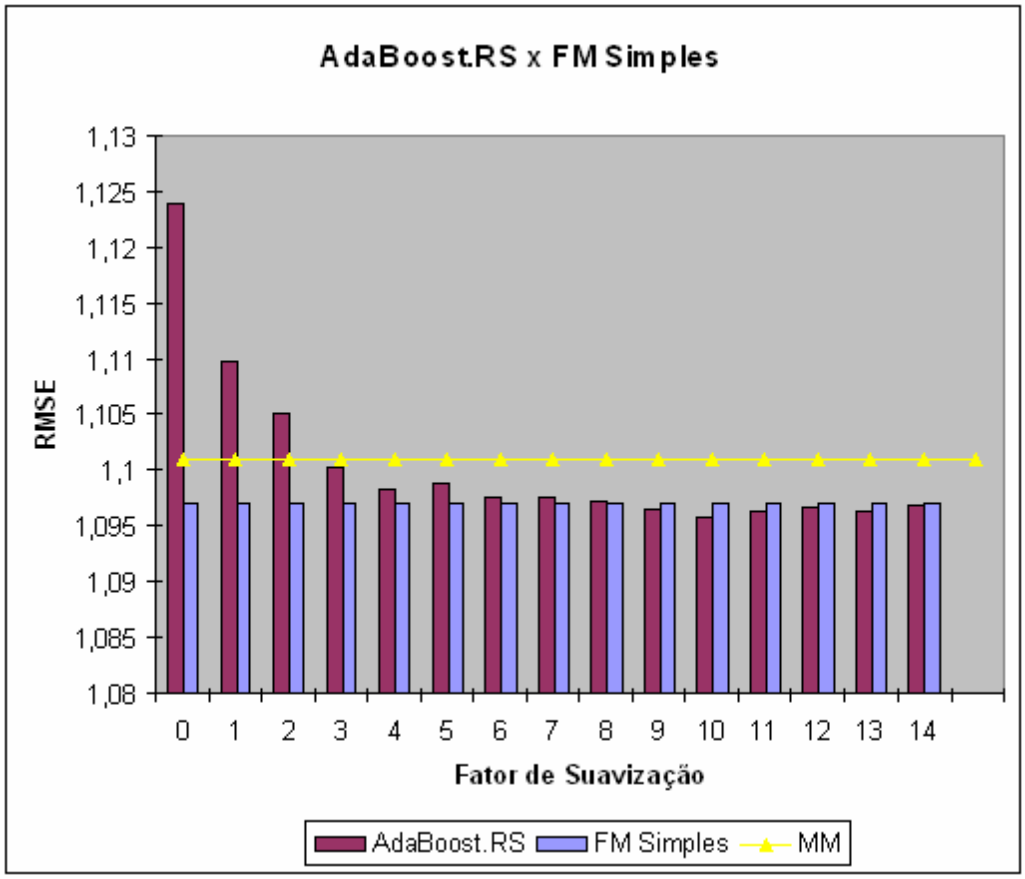

Figura 23: Calibrando o AdaBoost.RS para filmes.

Utilizamos então o fator de suavização 10 para aplicar o AdaBoost.RS ao FM. As Tabelas 11 e 12 apresentam os resultados, com as melhores marcas destacadas, dos *clusters* 1 e 2.

Tabela 11: Resultados do *Boosting* aplicado ao FM no *Cluster* 1.

Atributos Latentes	Quantidade de Preditores	RMSE
1	1	1,1239
	2	1,1185
	5	1,1224
	9	1,2116
2	1	1,1211
	2	1,1125
	5	1,1183
	9	1,1184
5	1	1,1092
	2	1,1063
	5	1,1053
	9	1,1056
10	1	1,1020
	2	1,1001
	5	1,1014
	9	1,0996
20	1	1,0972
	2	1,0981
	5	1,0959
	9	**1,0949**

Tabela 12: Resultados do *Boosting* aplicado ao FM no *Cluster* 2.

Atributos Latentes	Quantidade de Preditores	RMSE
1	1	1,0554
	2	1,0511
	5	1,0550
	9	1,0545
2	1	1,0486
	2	1,0407
	5	1,0513
	9	1,0511
5	1	1,0333
	2	1,0284
	5	1,0321
	9	1,0330
10	1	1,0193
	2	1,0172
	5	1,0207
	9	1,0178
20	1	1,0129
	2	**1,0116**
	5	1,0146
	9	1,0127

O *Boosting* na recomendação de filmes, assim como na recomendação de anúncios,sempre melhora o preditor simples. A seguir, na Figura 29, apresentamos o gráfico de decaimento do RMSE para 20atributos latentes nos *clusters* 1 e 2.

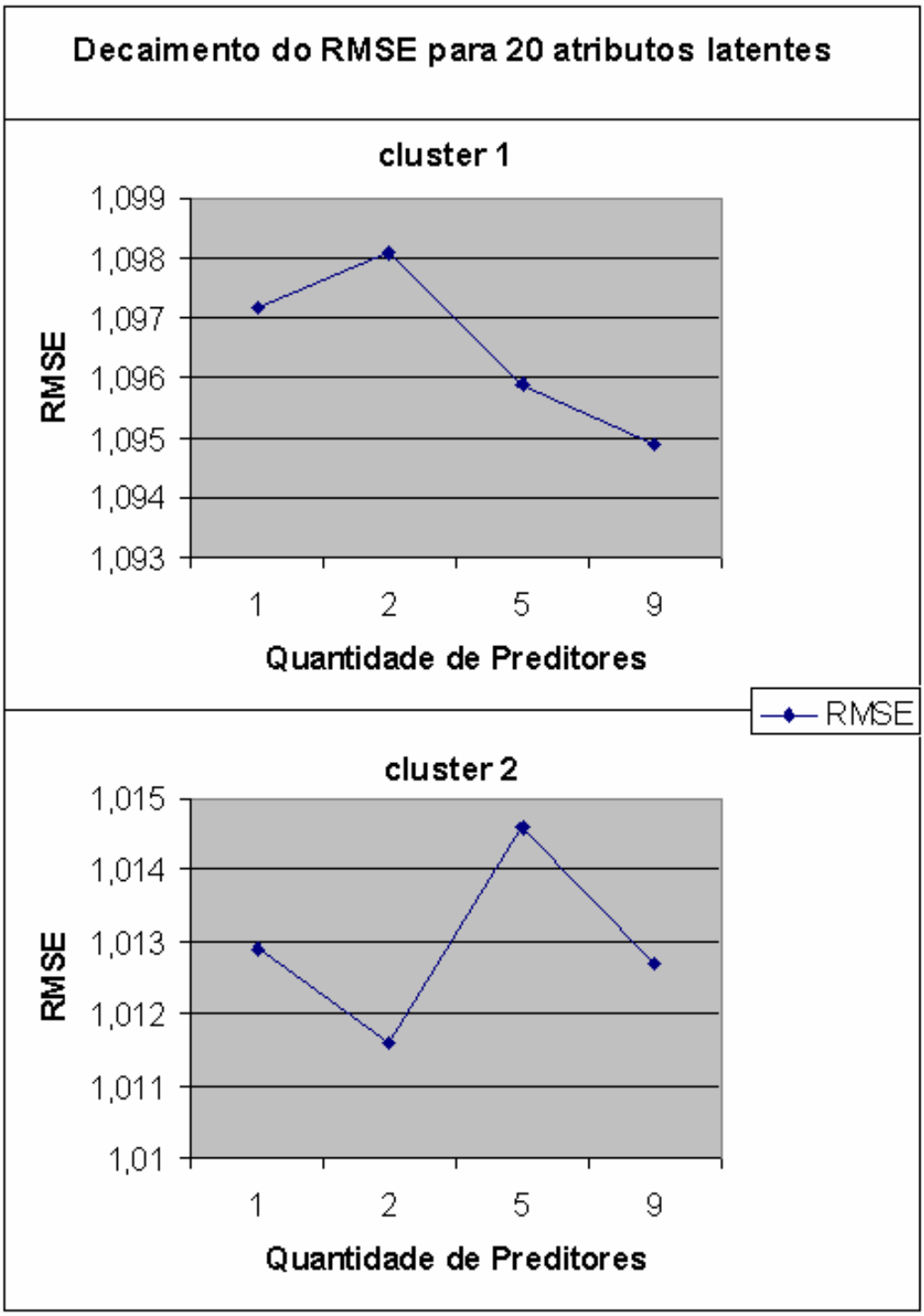

Figura 24: Melhora de 0,2% e 0,13% no RMSE com relação ao FM.