问题动机

Problem motivation

$x_0 = 1$

Movie	Alice (1) $\theta^{(1)}$	Bob (2) $\theta^{(2)}$	Carol (3) $\theta^{(3)}$	Dave (4) $\theta^{(4)}$	x_1 (romance)	x_2 (action)
$x^{(1)}$ ~~Love at last~~	5	5	0	0	~~?~~ 1.0	~~?~~ 0.0
Romance forever	5	?	?	0	?	?
Cute puppies of love	?	4	0	?	?	?
Nonstop car chases	0	0	5	4	?	?
Swords vs. karate	0	0	5	?	?	?

$$x^{(1)} = \begin{bmatrix} 1 \\ 1.0 \\ 0.0 \end{bmatrix}$$

$$\theta^{(1)} = \begin{bmatrix} 0 \\ 5 \\ 0 \end{bmatrix}, \theta^{(2)} = \begin{bmatrix} 0 \\ 5 \\ 0 \end{bmatrix}, \theta^{(3)} = \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}, \theta^{(4)} = \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}$$

$\theta^{(j)}$

$x^{(1)}$

$(\theta^{(1)})^T x^{(1)} \approx 5$

$(\theta^{(2)})^T x^{(1)} \approx 5$

$(\theta^{(3)})^T x^{(1)} \approx 0$

$(\theta^{(4)})^T x^{(1)} \approx 0$

Andrew Ng

假设我们并不知道电影中所包含的属性成分(特征)，但我们有每个用户喜欢不同类型电影的参数sita向量，和每个用户给这部电影打的分值，所以我们可以倒推出一部电影的特征包含。

Optimization algorithm

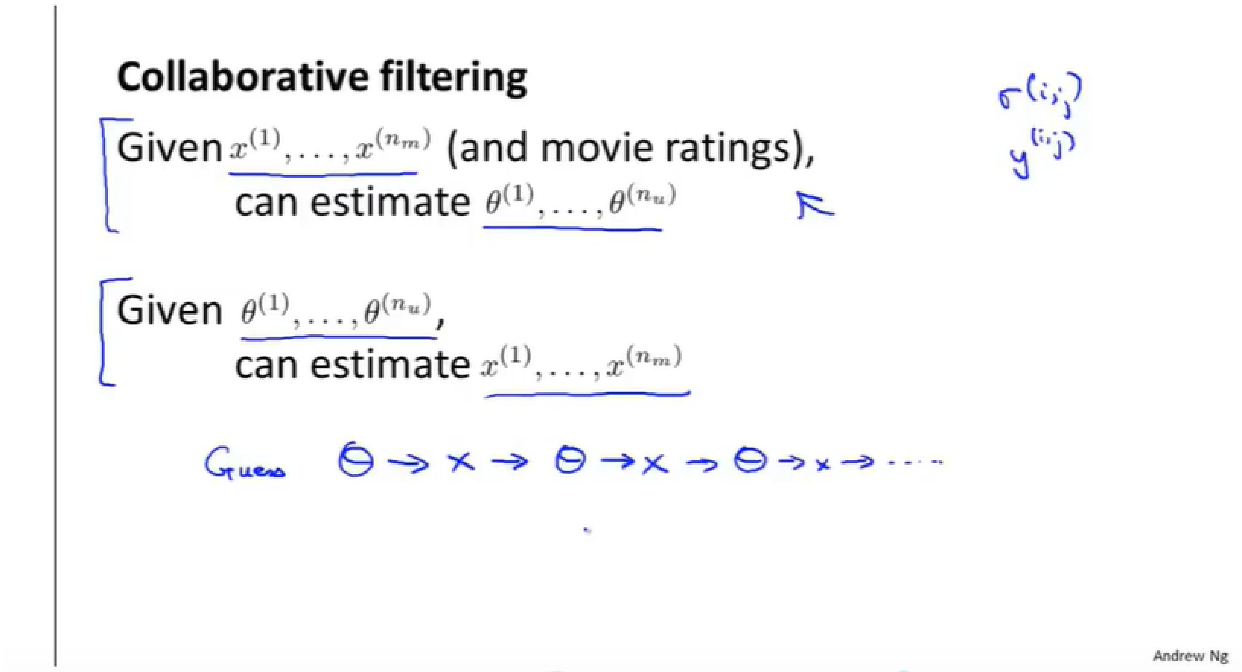
Given $\theta^{(1)}, \ldots, \theta^{(n_u)}$, to learn $x^{(i)}$:

$$\min_{x^{(i)}} \frac{1}{2} \sum_{j:r(i,j)=1} \left((\theta^{(j)})^T x^{(i)} - y^{(i,j)}\right)^2 + \frac{\lambda}{2} \sum_{k=1}^{n} (x_k^{(i)})^2$$

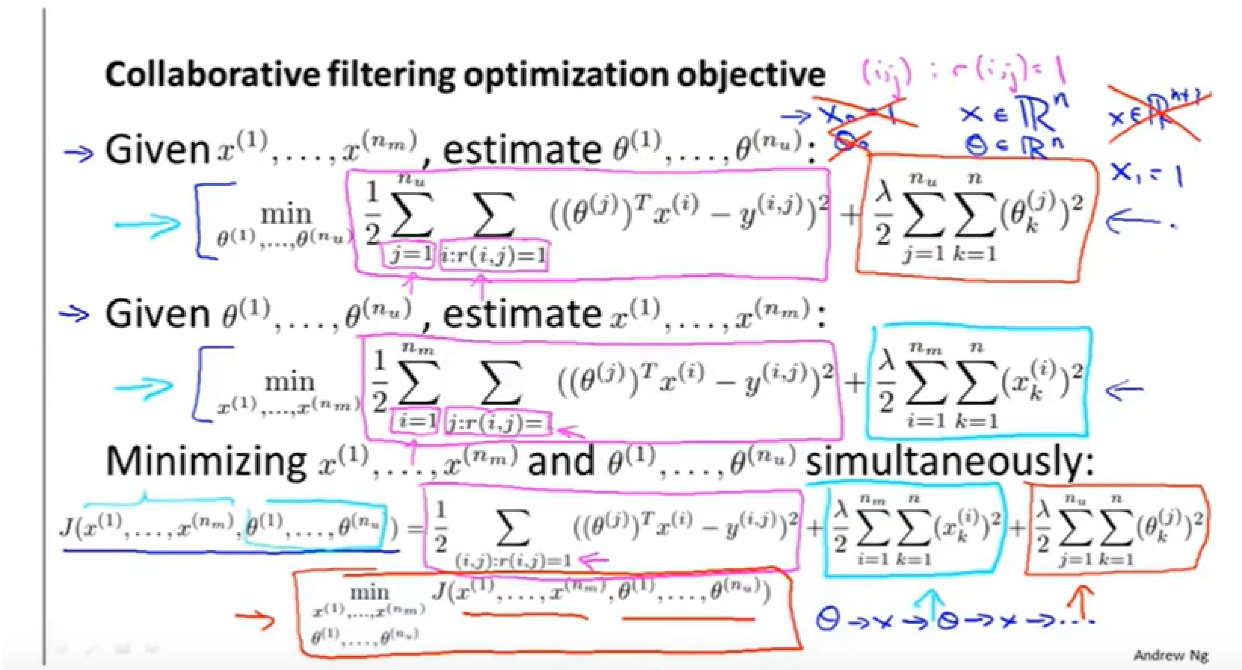
Given $\theta^{(1)}, \ldots, \theta^{(n_u)}$, to learn $x^{(1)}, \ldots, x^{(n_m)}$:

$$\min_{x^{(1)}, \ldots, x^{(n_m)}} \frac{1}{2} \sum_{i=1}^{n_m} \sum_{j:r(i,j)=1} \left((\theta^{(j)})^T x^{(i)} - y^{(i,j)}\right)^2 + \frac{\lambda}{2} \sum_{i=1}^{n_m} \sum_{k=1}^{n} (x_k^{(i)})^2$$

给定每个用户喜好的参数sita，最小化代价函数，学习每个电影的包含特征向量。

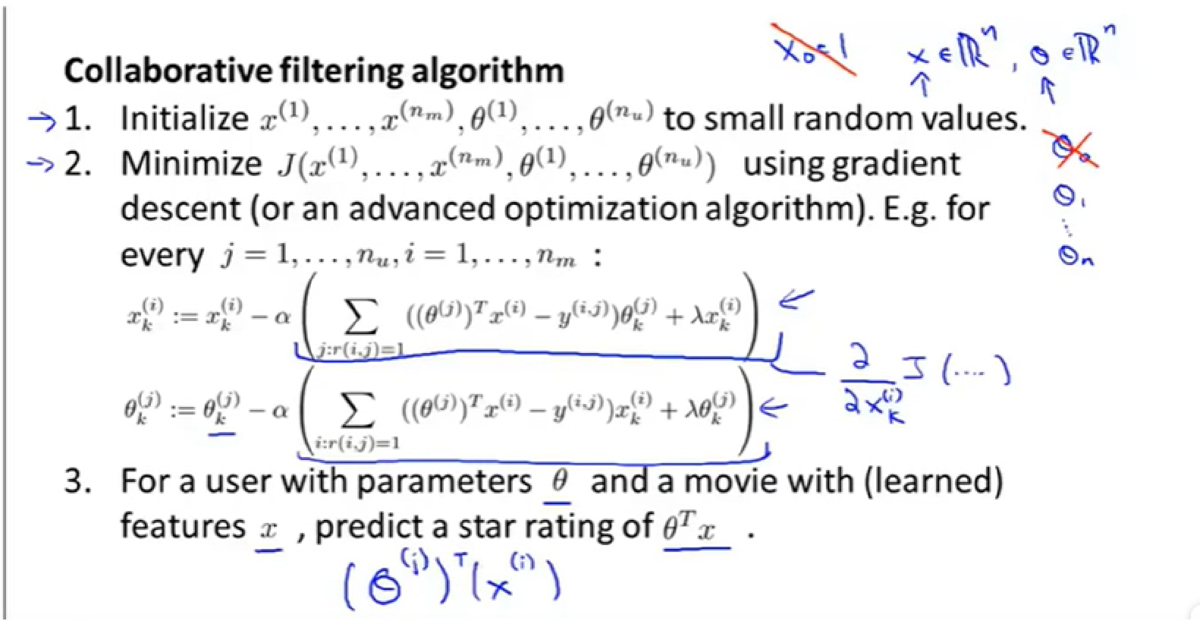


协同过滤过程：根据用户评分，先猜测出一些用户喜爱参数sita，再学习电影特征，再利用学习到的电影特征学习用户喜爱参数，进行迭代。



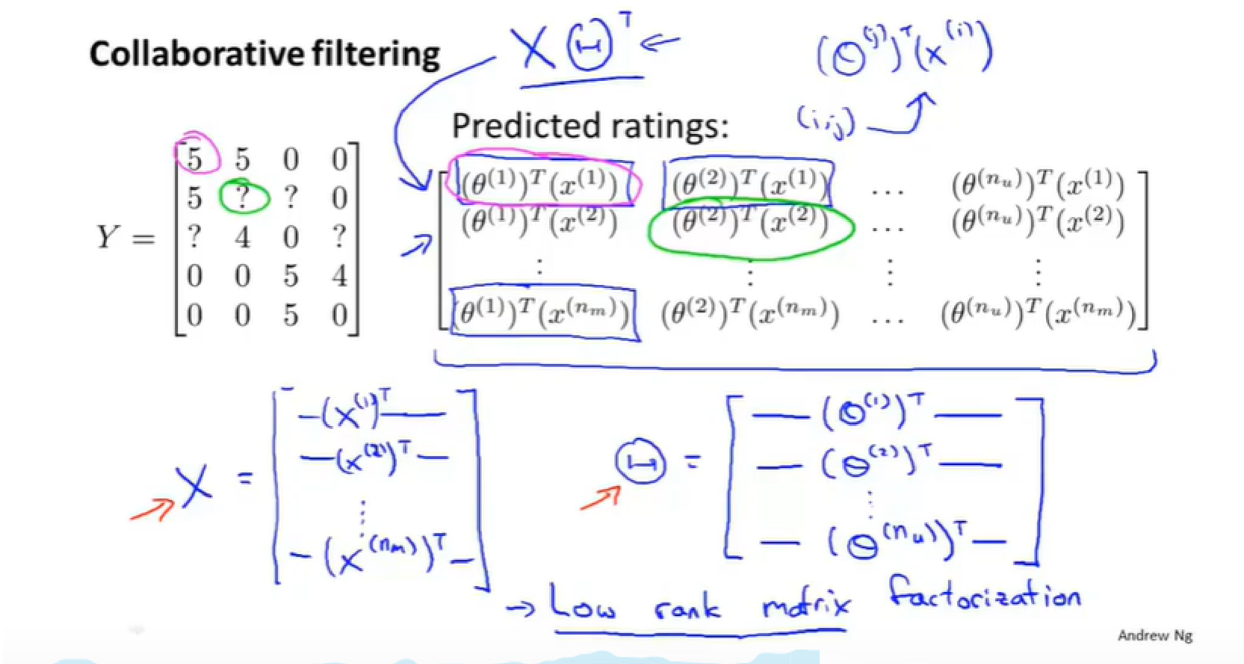
将两个式子结合起来，构成新的代价函数，不再需要在两个参数之间互相转换，这两个参数可以同时进行最小化

协同过滤算法流程

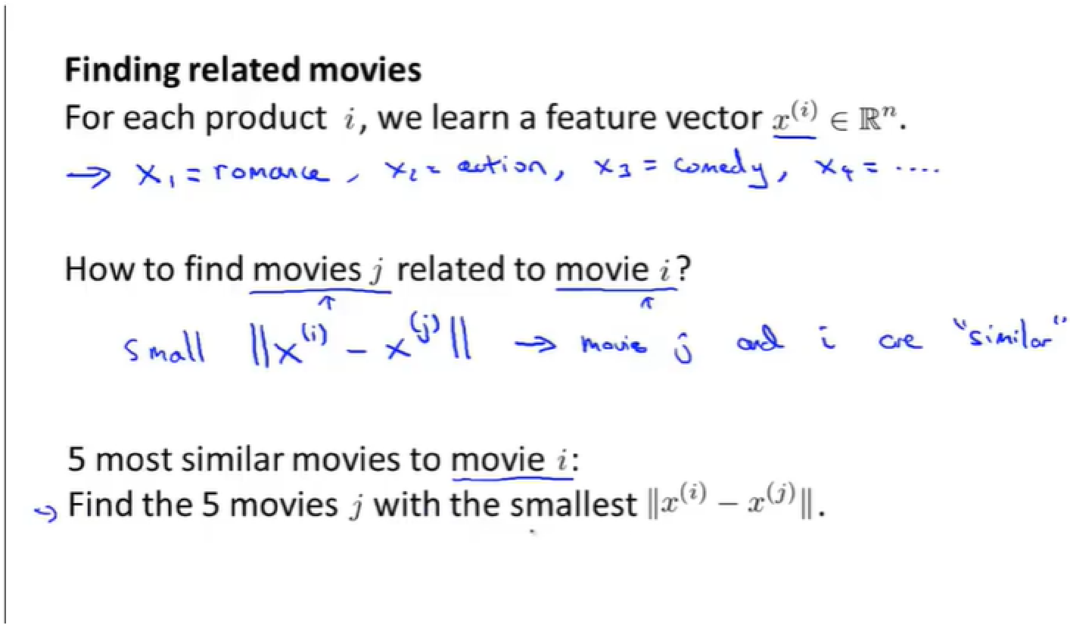


x，sita向量取随机值，利用梯度下降最小化代价函数，再利用sita转置和x计算评分

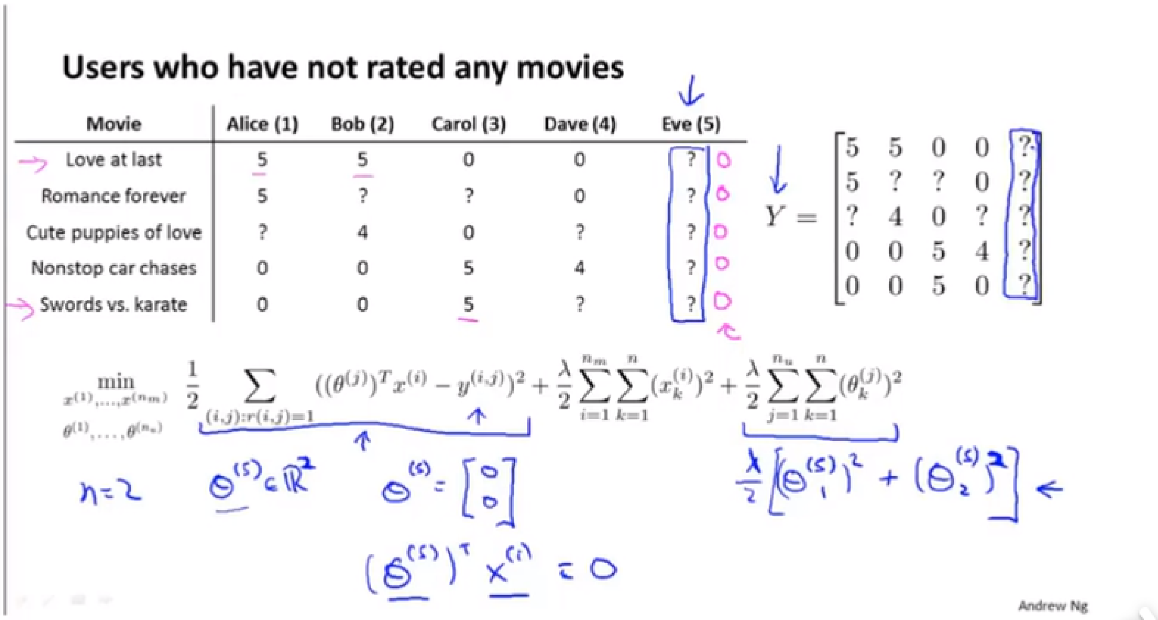
低秩矩阵分解



寻找相似的电影：找到特征参数距离最近的

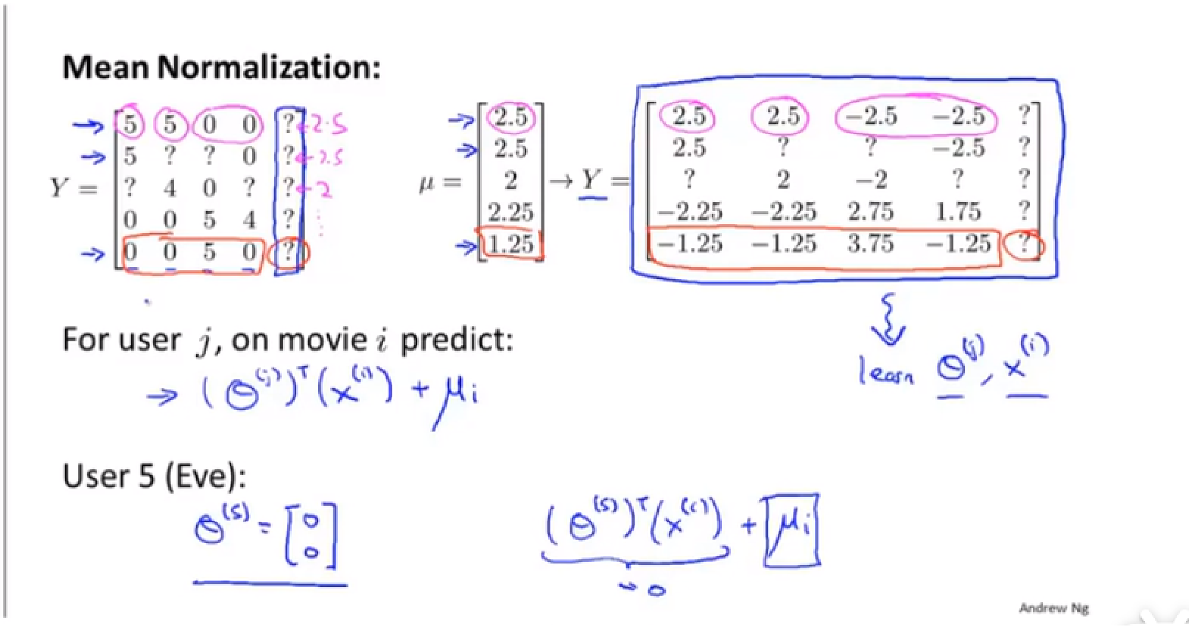


对于没有给过任何电影评分的用户，因为代价函数中的正则化项，所以用户的喜爱参数sita都会被置为0，所以最后计算预测评分时，sita转置乘以x，结果为0，对于没有给任何电影评分过的用户，预测评分都是0显然不合理，所以应该用**均值归一化**来解决这个问题。



均值归一化：

给评分矩阵都减去每一行的平均值，得到每个用户给一部电影打分的平均值μ，再用从未评分过的用户的喜爱参数sita转置乘以x，再加上平均值μ的得到结果预测评分，预测评分结果是平均值μ，这个预测是有意义的。



推荐系统中，关注没给任何电影评分过的用户，比关注没被评分过的电影有意义。没被评价过的电影可能不该推荐给任何人。