个性化推荐

张岚

December 2, 2016

1. 使用全量数据，取 index 为用户 id，column 为电影 id，从训练集中取对应的用户 id 对电影 id 的评分，建立一个新的矩阵，对于未知的项全定为 0，见算法 1。

算法 1 数据预处理

输入**:** *data* 数据

```
function procData(data)
    uids ← data.uid.unique
    fids ← data.fid.unique
    df ← zeros(uids, fids)
    for line ∈ data do
        x ← line[1]
        y ← line[2]
        df[x, y] ← line[3]
    end for
    return df
end function
```

2. 使用训练集的数据计算用户的相似度，预测测试集中用户对电影的打分，最后评估准确率，RMSE 值为 0.772586199418。本文使用矩阵形式运算程序运行 CPU 耗时 32.288，使用矩阵形式和协同过滤方法见算法 2。

3. 矩阵分解算法如下。

    a. 给定 $k = 50, \lambda = 0.01$，不同 α 得到的目标函数值和测试集上 RMSE 变化见下图。

    b. 不同 k 值对 RMSE 的影响，不同 λ 对 RMSE 的影响，程序最终选择。

算法 **2** 协同过滤

输入**:** X_train 训练数据,X_test 测试数据

function TASK1(X_train, X_test)
    $sim \leftarrow cosine\_similarity(X\_train)$
    $pred \leftarrow similarity.dot(X\_train)/np.array([np.abs(similarity).sum(axis = 1)]).T$
    $print \leftarrow RMSE((X\_test, pred))$
end function

算法 **3** 矩阵分解

输入**:** X_train 训练数据,X_test 测试数据

function TASK1(X_train, X_test)
    $sim \leftarrow cosine_similarity(X_train)$
    $pred \leftarrow similarity.dot(X_train)/np.array([np.abs(similarity).sum(axis = 1)]).T$
    $print \leftarrow RMSE((X_test, pred))$
end function

4. 计算时间上，准确率上