大数据技术与应用 - Homework 2

May 28, 2021

要求及说明：

- **诚信代码** 可以分组讨论大作业，但是必须独立完成解答。在提交的书面材料必须明确地写明参与讨论的其他组员名字。请在书面材料的末尾附上以下内容：
  “本人承诺本次作业的解答独立完成，解决过程中曾与 xxx、xxx 等同学共同讨论。”

---

1 使用 Spark 实现 k-means (20 分)

注意：本题需要较长的计算时间，请尽早开始。另外，本题使用 Spark MLlib 中的聚类算法**不得分**。

本题将帮助你理解 Spark 实现聚类算法的诸多细节。并且本题也将帮助你理解各种距离度量和初始化策略在实践中的影响。假设我们在 d-维空间 R^d 有包含 n 个点的数据集 $\mathcal{X}$，给定簇的数量 k 和 k 个质心 $\mathcal{C}$，现在我们需要定义多种距离度量，以及它们相应地最小化地代价函数。

欧氏距离（Euclidean distance） 给定两个点 A 和 B，$A = [a_1, a_2 \dots a_d], B = [b_1, b_2 \dots b_d]$，A 和 B 之间的欧氏距离定义为：

$$||a - b|| = \sqrt{\sum_{i=1}^{d} (a_i - b_i)^2}$$

当我们指派数据点到簇时，欧氏距离对应的最小化目标的代价函数是：

$$\phi = \sum_{x \in \mathcal{X}} \min_{c \in \mathcal{C}} ||x - c||^2$$

曼哈顿距离（Manhattan distance） 给定两个点 A 和 B，$A = [a_1, a_2 \dots a_d], B = [b_1, b_2 \dots b_d]$，A 和 B 之间的曼哈顿距离定义为：

$$|a - b| = \sum_{i=1}^{d} |(a_i - b_i)|$$

当我们指派数据点到簇时，曼哈顿距离对应的最小化目标的代价函数是：

$$\psi = \sum_{x \in \mathcal{X}} \min_{c \in \mathcal{C}} |x - c|$$

迭代 k-Means 算法： 理论课上我们已经学过了基本的 k-Means 算法：初始化 k 个质心，将每个数据点指派给最近的质心，然后根据指派的结果重新计算每个簇的质心。在实践中，上述步骤会执行多个迭代。我们将迭代的 k-Means 算法表述为算法 1。

使用 Spark 的迭代 k-Means 聚类： 使用 Spark 实现迭代 k-Means。请使用作业包中的 `q1/data` 目录下的数据集。

这个文件夹下有 3 个文件：

1. `data.txt` 包含一个 4601 行 58 列的数据集。每行是一个 58 维特征向量表示的文档。向量每维上的分量代表文档中一个词的重要性。
2. `c1.txt` 包含 k 个簇的质心。这些质心通过在输入数据中随机选择 $k = 10$ 个点得到。

Algorithm 1 迭代 k-Means 算法

```
1: procedure 迭代 k-MEANS
2:     选择 k 个点作为 k 个簇的初始质心
3:     for iterations := 1 到 MAX_ITER do
4:         for 数据集中的每个点 p do
5:             将 p 指派给最近质心代表的簇
6:         计算本轮迭代的代价
7:         for 每个簇 c do
8:             按簇 c 中所有点的平均位置重新计算 c 的质心
```

3. `c2.txt` 包含相互尽可能远离的簇质心。(你可以先随机选择第一个质心 c1，然后找出离 c1 最远的点作为质心 c2，然后选择离 c1 和 c2 最远的点作为质心 c3，以此类推。)

在本题的所有实验中，设置迭代次数 `MAX_ITER` 为 20，簇的数量 k 为 10。你的程序应该保证算法执行了正确的迭代次数。

(a) 探索欧氏距离下的初始化策略

1. (5 分) 使用欧氏距离作为距离度量，计算每轮迭代 i 的代价函数 $\phi(i)$。这意味着在第一轮迭代中，你需要使用 `c1.txt` 和 `c2.txt` 中质心的位置计算代价；然后分别这两份数据的初始条件下运行 k-Means 聚类。绘制两张图，分别描述 `c1.txt` 和 `c2.txt` 作为初始质心位置条件，迭代次数从 1 到 20 时代价函数 $\phi(i)$ 的变化。
   (提示：你并不需要另写一个 Spark 任务来计算 $\phi(i)$，而在指派数据点给簇时你就能算出代价是多少。)
2. (5 分) 对于述 `c1.txt` 和 `c2.txt`，在 k-Means 算法第 10 次迭代后的变化百分比分别是多少？以代价函数 $\phi(i)$ 来看，随机初始化方案 `c1.txt` 比 `c2.txt` 的初始化方案更好么？解释你的结论。

(b) 探索曼哈顿距离下的初始化策略

1. (5 分) 使用曼哈顿距离作为距离度量，计算每轮迭代 i 的代价函数 $\psi(i)$。这意味着在第一轮迭代中，你需要使用 `c1.txt` 和 `c2.txt` 中质心的位置计算代价；然后分别这两份数据的初始条件下运行 k-Means 聚类。绘制两张图，分别描述 `c1.txt` 和 `c2.txt` 作为初始质心位置条件，迭代次数从 1 到 20 时代价函数 $\psi(i)$ 的变化。
2. (5 分) 对于述 `c1.txt` 和 `c2.txt`，在 k-Means 算法第 10 次迭代后的变化百分比分别是多少？以代价函数 $\psi(i)$ 来看，随机初始化方案 `c1.txt` 比 `c2.txt` 的初始化方案更好么？解释你的结论。

提交内容

(i) 1(a) 和 1(b) 的代码
(ii) 1(a) 中代价函数关于迭代次数的关系绘图
(iii) 1(a) 中代价函数的变化百分比对比以及你的解释
(iv) 1(b) 中代价函数关于迭代次数的关系绘图
(v) 1(b) 中代价函数的变化百分比对比以及你的解释

2 推荐系统 (30 分)

考虑一个用户-商品二分图，图中的边都是从用户 U 指向商品 I，表示用户 U 喜欢商品 I。我们将这样的用户和商品关系用评分矩阵 R 表示，其中 R 的每行对应一个用户，每列对应一个商品。如果用户 i 喜欢商品 j，则有 $R_{i,j}=1$，否则 $R_{i,j}=0$。假设我们有 m 个用户，n 个商品，因此矩阵 R 是 $m\times n$ 的。

定义一个 $m\times m$ 的对角矩阵 P，它的第 i 个对角元素是用户结点 i 的度，即该用户 i 喜欢的商品数量。类似地，定义矩阵 Q 是一个 $n\times n$ 的对角矩阵，它的第 i 个对角线元素是商品结点 i 的度，即喜欢商品商品 i 的用户数量，具体见图示。

(a) (4 分) 定义未规范化的用户相似度矩阵 $T=R*R^T$。用二分图的结构（例如结点的度，结点之间的路径等）解释 T_{ii} 和 $T_{ij}(i\neq j)$ 的含义。

余弦相似度：记得两个向量 u 和 v 的余弦相似度定义为：

$$\text{cos-sim}(u,v)=\frac{u\cdot v}{||u||||v||}$$

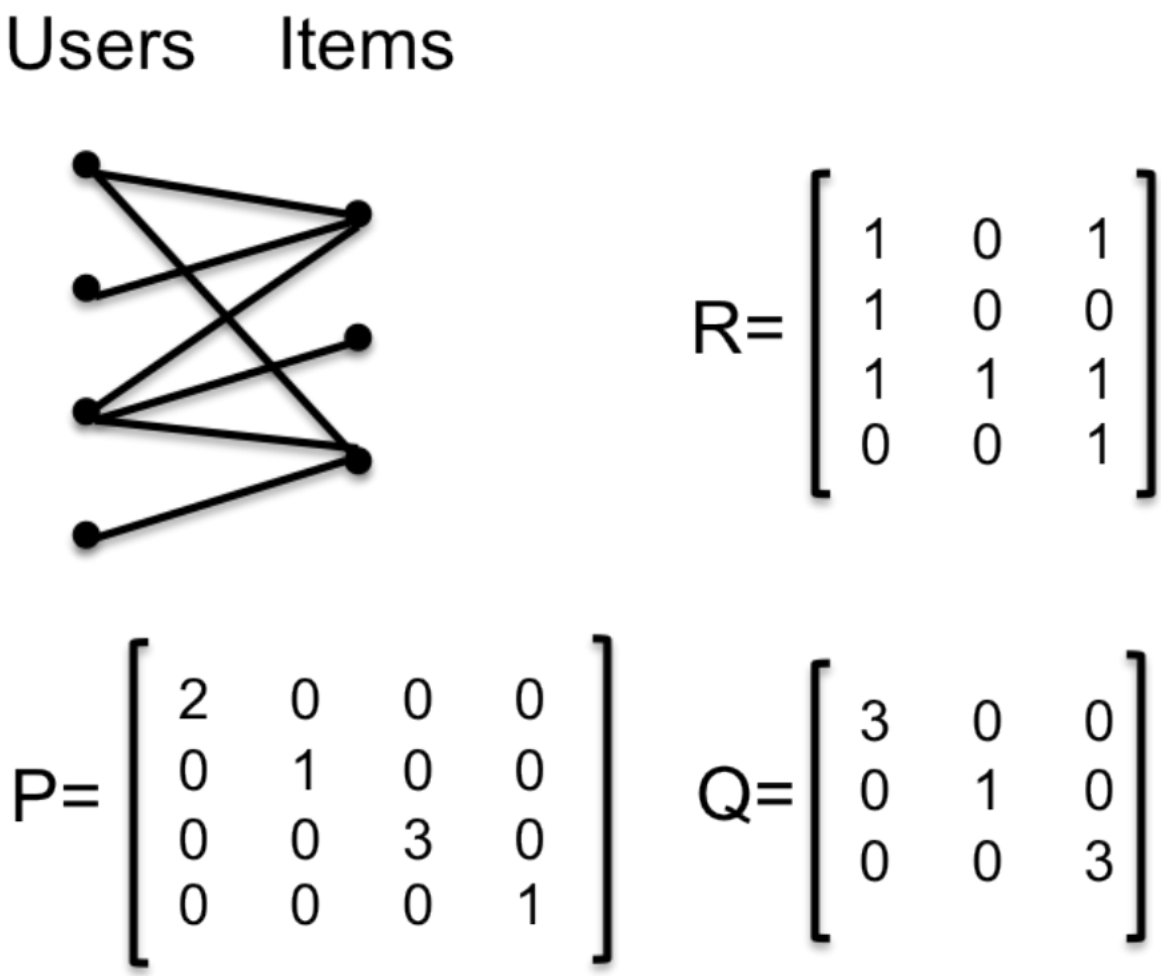

Figure 1: 用户-商品的二分图

(b) (5 分) 定义商品相似度矩阵为 $n \times n$ 的矩阵 S_I，使得第 i 行第 j 列的元素是商品 i 和商品 j 的余弦相似度，其中商品 i 和商品 j 分别对应 R 矩阵的第 i 列和第 j 列。请证明 $S_I = Q^{-1/2}R^TRQ^{-1/2}$，其中 $Q^{-1/2}$ 定义为对所有非零的元素 Q_{rc}，$Q_{rc}^{-1/2} = 1/\sqrt{Q_{rc}}$，其余位置为 0.

然后对用户相似度矩阵 S_U 重复相同的问题，其中 S_U 的第 i 行第 j 列是用户 i 和用户 j 的余弦相似度。也就是说，S_U 应该表示为 R, P, Q 的某种矩阵运算结果。你的解答中应该展示表达式的推导过程。

（注意：本题中你可以用矩阵的 1/2 次方表示对矩阵的每个元素求平方根）

(c) (6 分) 使用 “用户-用户” 协同过滤为用户 u 推荐的方法可以描述如下：对所有商品 s，计算 $r_{u,s} = \sum_{x \in users} \text{cos-sim}(x, u) * R_{xs}$，并将 $r_{u,s}$ 值最大的 k 个商品作为推荐结果。

类似地，使用 “商品-商品” 协同过滤为用户 u 推荐的方法可以描述如下：对所有的商品 s，计算 $r_{u,s} = \sum_{x \in items} \text{cos-sim}(x, s) * R_{ux}$，并将 $r_{u,s}$ 值最大的 k 个商品作为推荐结果。

定义推荐 $m \times n$ 的矩阵 Γ，使得 $\Gamma(i, j) = r_{i,j}$。找出 “用户-用户” 和 “商品-商品” 协同过滤的 Γ 以 R, P, Q 的表达式。

提示：对于 “商品-商品” 的情况，$\Gamma = RQ^{-1/2}R^TRQ^{-1/2}$

你的解答需要展示表达式是如何推导的。即便 “商品-商品” 的表达式已经给出，也需要写出推导过程。

(d) (15 分) 本小题中我们将这些协同过滤的方法应用于一个真实的数据集，数据包含电视节目的信息。准确来说，对于 9985 个用户和 563 个流行的电视节目，我们知道一个用户 3 个月内是否看过某个电视节目。

本小题使用 `q2/data` 中的数据集。文件夹包含：

- `user-shows.txt` 评分矩阵 R，每行对应一个用户，每列对应一个电视节目。如果用户 i 在三个月内看过电视节目 j，那么 $R_{ij} = 1$。列之间是以空格分开的。
- `shows.txt` 该文件包含电视节目的标题，与 R 中的列以相同顺序出现。

我们将比较 “用户-用户” 和 “商品-商品” 的协同过滤推荐方法在第 500 位用户上的结果。假设我们称这位用户为 Alex。

为了完成比较，我们先将 Alex 对应行的前 100 个元素置 0，表示我们不知道 Alex 看过其中的哪些。根据 Alex 在其他电视节目上的行为，我们将在前 100 部电视节目上给出推荐结果。然后我们检查 Alex 实际上是否看过我们推荐的电视节目。

- 计算矩阵 P 和 Q
- 使用 (c) 中得到的公式，计算用户-用户协同过滤的 Γ 矩阵。记 S 为前 100 部电视节目，S 中的哪 5 部对于 Alex 有最高的相似性得分？它们的相似性得分是多少？如果有相同得分的电视节目，选编号小的那一个。你应该在结果中给出电视节目的标题而不是编号。
- 计算商品-商品协同过滤的 Γ 矩阵。S 中的哪 5 部对于 Alex 有最高的相似性得分？它们的相似性得分是多少？如果有相同得分的电视节目，选编号小的那一个。同样地，你应该在结果中给出电视节目的标题而不是编号。

提交内容

(i) T_{ii} 和 T_{ij} 的解释

(ii) 用 R, P 和 Q 表示 S_I 和 S_U，并解释

(iii) 用 R, P 和 Q 表示 Γ，并解释

(iv) 以下内容

- 用户-用户协同过滤方法下为 Alex 推荐的 5 部相似得分最高的电视节目
- 商品-商品协同过滤方法下为 Alex 推荐的 5 部相似得分最高的电视节目
- 你的源代码

(v) 4(d) 的代码