数据科学导论

Introduction to Data Science

第二章 数据分析基础

黄振亚，陈恩红
Email: huangzhy@ustc.edu.cn, cheneh@ustc.edu.cn

课程主页：
http://staff.ustc.edu.cn/~huangzhy/Course/DS2024.html

回顾：数据分析基础

- 数据采集 Data Collection
- 数据存储 Data Storage
- 数据预处理 Data Preprocessing
- 特征工程 Feature Engineering

数据预处理

- 大数据环境下的数据特点
- 为什么需要进行预处理
- 预处理的基本方法
  - 数据清洗
  - 数据集成
  - 数据变换
  - 数据规约

大数据环境下的数据特点—4V

数据来源多样：传感器，IT系统，应用软件等

数据类型多样：结构化，半结构，非结构

多样 Variety

数据分析与结果需要及时处理，实时的结果才有价值—1秒定律

高速 Velocity

计量单位一般是TB，甚至到了PB，EB或ZB

大量 Volume

TB (2^30KB)
PB (2^40KB)
EB (2^50KB)
ZB (2^60KB)

“沙里淘金”：价值密度低，价值深度深，带来巨大的科学和商业价值

价值 Value

Big Data

大数据环境下的数据特点

- 收集来的数据，是否可以直接使用？

*ratings.csv - 记事本

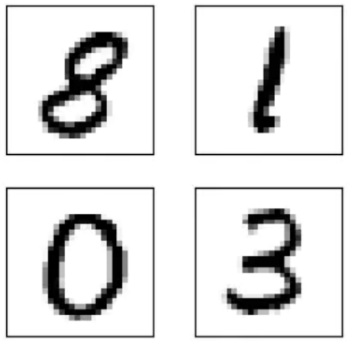
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

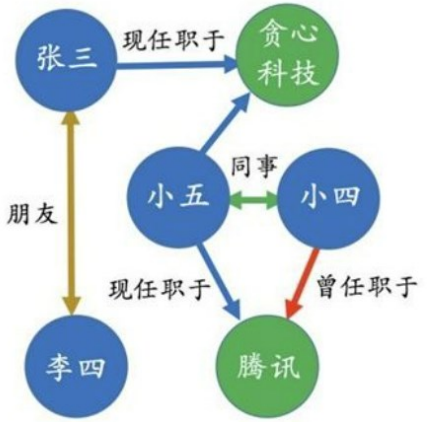
```
userId,movieId,rating,timestamp
1,1,4.0,964982703
1,3,4.0,964981247
1,6,4.0,964982224
1,47,5.0,964983815
1,50,5.0,964982931
1,70,3.0,964982400
1,101,5.0,964980868
```

Context:
Computational complexity theory is a branch of the theory of computation in theoretical computer science that focuses on classifying computational problems according to their inherent difficulty, and relating those classes to each other. A computational problem is understood to be a task that is in principle amenable to being solved by a computer, which is equivalent to stating that the problem may be solved by mechanical application of mathematical steps, such as an algorithm.

Question:
By what main attribute are computational problems classified using computational complexity theory?

Answer:
inherent difficulty





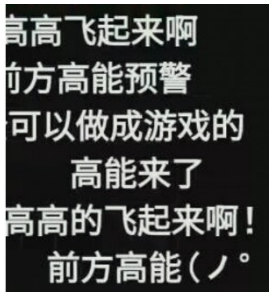
通常情况下，直接收集的数据难以直接使用，需要对数据进行预处理

数据预处理

- 大数据环境下的数据特点
- 为什么需要进行预处理
- 预处理的基本方法
  - 数据清理
  - 数据集成
  - 数据变换
  - 数据规约

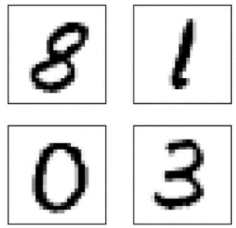
为什么进行数据预处理

- 直接收集的数据通常是“脏的”—数据来源不同
  - 应用需求
    - 微博
    - 淘宝
    - ……
  - 收集手段
    - 传感器，扫描仪
    - 摄像，照相
    - App收集
    - 爬虫写错了
    - ……
  - 数据格式
    - 结构化
    - 半结构化
    - 非结构化
    - ……



ID	时间	内容
陈赫	08-18	天霸
邓超	08-18	我们都很好
邓超	08-18	我也不知道

```
<province>
  <name>黑龙江</name>
  <cities>
    <city>哈尔滨</city>
    <city>大庆</city>
  </cities>
</province>
```



为什么进行数据预处理

- 直接收集的数据通常是“脏的”
  - 不完整
    - 有些数据属性的值丢失或不确定
    - 缺失必要的数据，例：缺失学生成绩
    - ……
  - 不准确
    - 数据错误，属性值错误，例：成绩 = -10
    - 噪声数据：包含孤立（偏离期望）的离群
    - ……
  - 不一致
    - 数据结构有较大差异，例，编码或者命名上存在差异
    - 数据需求改动，例，评价等级：“百分制”与“A，B，C”
    - 存在数据重复和信息冗余现象
    - ……

学号 Sno	课程号 Cno	成绩 Grade
200215121	1	92
200215121	2	85
200215121	3	88
200215122	2	90
200215122	3	80

为什么进行数据预处理

- 现实世界的数据是“脏的”——举例
  - 滥用缩写词　例：中科大， 科大，中国科大，USTC
  - 数据中的内嵌控制信息　例：E3=F3*C3
  - 不同的惯用语　例：南七技校
  - **重复记录**
  - **缺失值**
  - 拼写变化与时态，例： propose, proposed, proposing
  - **不同的计量单位**
  - **噪声**
  - **UGC数据，例：弹幕，颜文字(*^▽^*)**

为什么进行数据预处理

- 数据错误的不可避免性
  - 数据输入和获得过程数据错误的不可避免性
  - 数据集成所表现出来的错误
  - 数据传输过程所引入的错误
  - 据统计，有错误的数据占总数据的5%左右

为什么进行数据预处理

- 没有高质量的数据，就没有高质量的结果
  - 高质量的决策必须依赖高质量的数据
    - 例. 数据重复或者缺失将会产生不正确的分析结果，误导决策

- 数据质量的含义
  - 正确性（ Correctness）
  - 一致性（ Consistency）
  - 完整性（ Completeness）
  - 可靠性（ Reliability）

- **数据预处理是进行大数据的分析和挖掘的工作中占工作量最大的一个步骤 （80%）**

数据预处理

- 大数据环境下的数据特征
- 为什么需要进行预处理
- 预处理的基本方法
  - 数据清理
  - 数据集成
  - 数据变换
  - 数据规约

数据预处理：数据清理

- 数据清理的目标
  - 解决数据质量问题
  - 让数据更适合分析、建模
- 数据清理基本任务
  - 处理缺失值
  - 清洗噪声数据
  - 纠正不一致数据
  - 根据需求进行清理
  - ……

ID	住址	学历	单位	专业	收入
01	A区	本科	A	CS	C
02	B区	本科	B	EE	C
03	A区	本科	A	CS	C
04	A区	硕士	C	CS	B
05	A区	博士	A	DS	A
…	…	…	…	…	…

ID	住址	学历	单位	专业	收入
01	A区	本科	A	CS	C
02	B区	本科	B	EE	C
03	A区	本科	A	CS	0
04	A区		C	CS	B
	A区	博士	A	DS	
…	…	…	…	…	…

数据清理-处理缺失值

- 造成数据缺失的原因
  - 信息无法获取，或获取代价大。
    - 反爬虫，加密
  - 信息遗漏
    - 需求不明确
    - 采集故障，存储故障，传输故障
    - 人为因素
  - 数据的某些属性不可用，或不存在（与设计有关）
    - 如：学生的收入，老师的成绩等

数据清理-处理缺失值

- 数据缺失的类型
  - 完全随机缺失：不依赖其他属性/变量，不影响样本的无偏性
  - 随机缺失：缺失与其他完全属性/变量有关系
    - “期末成绩”的依赖于“平时表现”
    - “工资”与“人群背景”的关系
  - 非随机缺失: 数据缺失与属性/变量自身的取值有关
    - 工资问卷

ID	住址	学历	单位	专业	收入
01	A区	本科	A	CS	C
02	B区	本科	B	EE	C
03	A区	本科	A	CS	0
04	A区		C	CS	B
	A区	博士	A	DS	
…	…	…	…	…	…

数据清理-处理缺失值

- 处理缺失数据的方法：首先确认缺失数据的影响
  - 数据删除（可能丢失信息，或改变分布）
    - 删除数据
    - 删除属性
    - 改变权重
  - 数据填充
    - 特殊值填充
      - 空值填充，不同于任何属性值。例，NLP词表补0，DL补mask
      - 样本/属性的均值、中位数、众数填充
    - 预测：使用最可能的数据填充
      - 热卡填充（就近补齐）
      - K最近距离法（KNN）
      - 利用回归等估计方法
      - 大模型等

模型预测：建立模型预测缺失值

数据清理-处理缺失值

- 热卡填充
  - 完整数据中找到**1个**与它最相似的样例，然后用该样本的值来进行填充
- K最近距离法
  - 根据相关分析(距离)来确定距离缺失数据样本的最近**K个**样本
  - 将这K个值加权平均估计样本缺失数据
- 模型法：回归法
  - 基于**数据集**，建立回归模型
  - 将已知属性值代入模型来估计未知属性值，以此预测值填充

ID	住址	学历	单位	专业	收入
01	A区	本科	A	CS	C
02	B区	本科	B	EE	C
03	A区	本科	A	CS	**0**
04	A区		C	CS	B
	A区	博士	A	DS	
…	…	…	…	…	…

数据清理-清洗噪声

- 噪声是测量误差的随机部分
  - 包括错误值，或偏离期望的孤立点值
  - 需要对数据进行平滑
- 常用的处理方法
  - 分箱(binning)
    - 利用近邻数据对数据进行平滑
  - 回归(Regression)
    - 让数据适应回归函数来平滑数据
  - 识别离群点，常用聚类方法
    - 监测并且去除孤立点

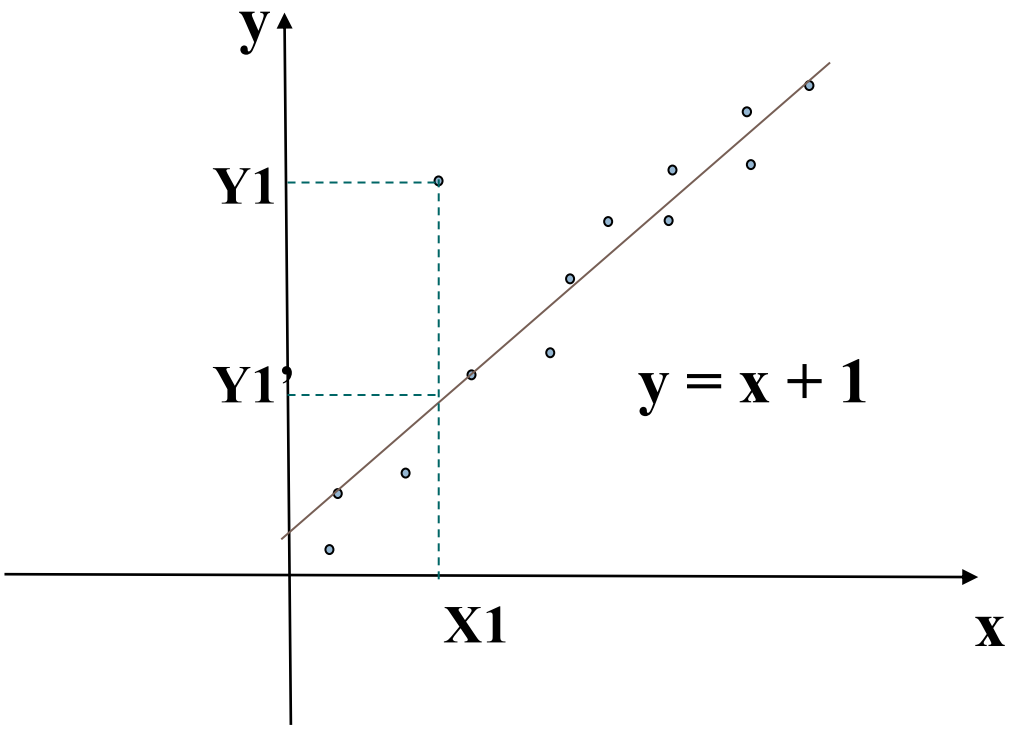
ID	住址	学历	单位	专业	收入
01	A区	本科	A	CS	C
02	B区	本科	B	EE	C
03	A区	本科	A	CS	0
04	A区		C	CS	B
	A区	博士	A	DS	
…	…	…	…	…	…

数据清理-清洗噪声

- 分箱：利用近邻数据进行数据平滑
  - 1. 排序数据，并将他们分到等深的箱中
  - 2. 按箱平均值平滑、按箱中值平滑、按箱边界平滑等(离散化)
- 例：排序后数据：4，8，15，21，21，24，25，28，34
  - 1. 划分为（等深的）箱
    - 箱1：4，8，15
    - 箱2：21，21，24
    - 箱3：25，28，34
  - 2-1. 用箱平均值平滑
    - 箱1：9，9，9
    - 箱2：22，22，22
    - 箱3：29，29，29
  - 2-2. 用箱边界平滑
    - 箱1：4，4，15
    - 箱2：21，21，24
    - 箱3：25，25，34

数据清理-清洗噪声

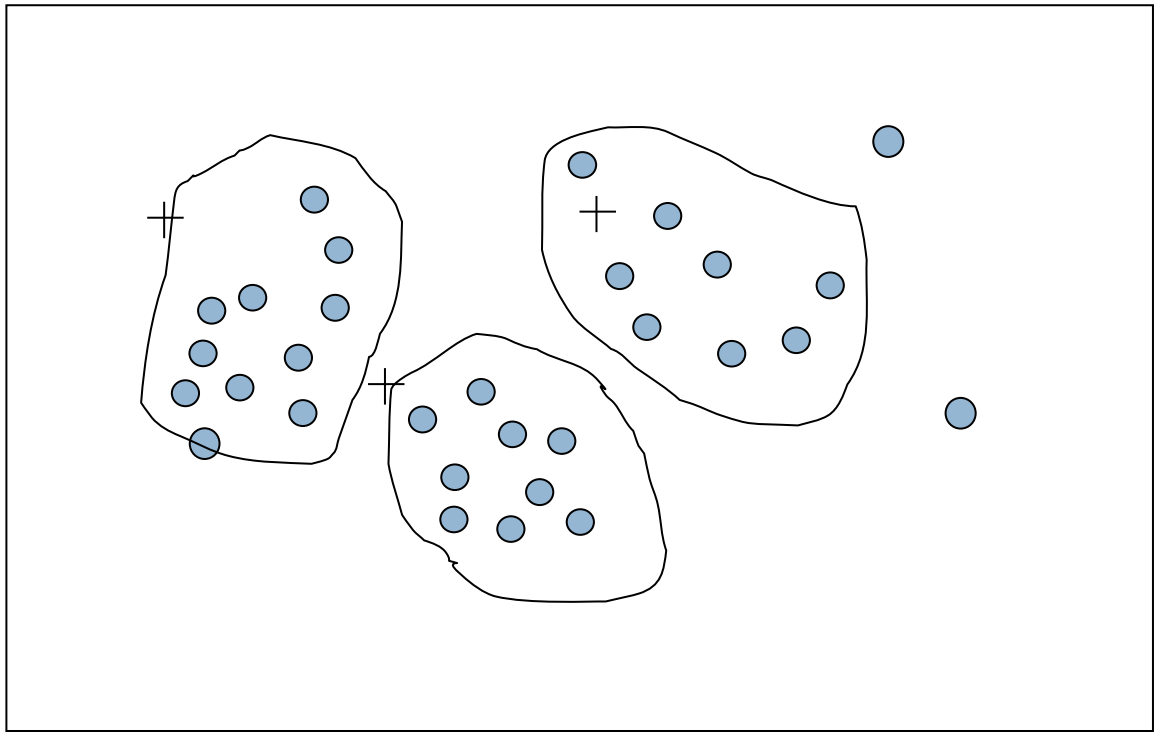
- **回归：让数据适应回归函数来平滑数据**
  - 通过线性回归模型，对不符合回归的数据进行平滑处理
  - 用某些属性预测其他属性



ID	住址	学历	单位	专业	收入
01	A区	本科	A	CS	C
02	B区	本科	B	EE	C
03	A区	本科	A	CS	0
04	A区		C	CS	B
	A区	博士	A	DS	
…	…	…	…	…	…

数据清理-清洗噪声

- **识别离群点：聚类分析检测离群点，消除噪声**
  - 聚类将类似的值聚成簇
  - 落在簇集合之外的值被视为离群点

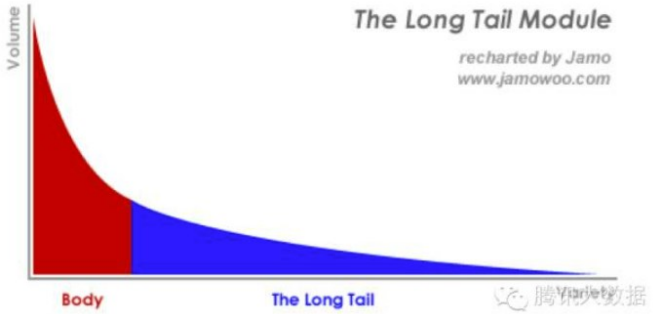


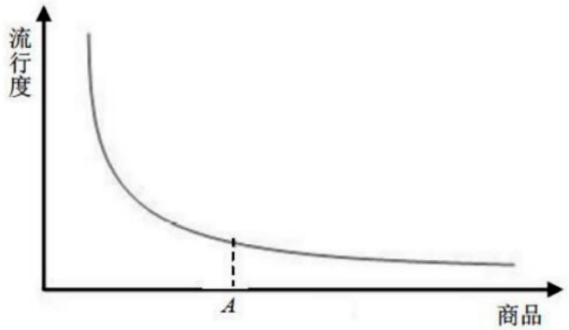
数据清理-根据需求清理数据

- 在特定的应用任务中，根据目标不同，需要特殊的数据清理方法
  - 推荐系统
    - 通用推荐问题
    - 冷启动问题
  - 教育大数据
  - 社交网络
  - POI任务：Point of interest

Dataset	Douban Book	Yelp
# Users	6,576	25,783
# Items	20,547	33,105
# Ratings	326,419	727,259
Rating Sparsity	99.76%	99.91%
Avg. friends of each user	6.0	3.8
# Users without friends	1,314	10,867

Table 1: The statistics of two datasets.





Li Wang, Zhenya Huang, Qi Liu, Enhong Chen, Preference-Adaptive Meta-Learning for Cold-Start Recommendation, IJCAI'2021.

数据预处理

- 大数据环境下的数据特征
- 为什么需要进行预处理
- **预处理的基本方法**
  - 数据清理
  - **数据集成**
  - 数据变换
  - 数据规约

数据预处理：数据集成

- 数据集成
  - 将多个数据源的数据整合到一个一致的数据存储中
- 数据集成的目标
  - 获得更多的数据
  - 获得更完整的数据
  - 获得更全面的数据画像，如用户画像
- 例：电商推荐-需求
  - 用户的购物记录：淘宝，美团，拼多多 等
  - 用户的社交网络：微博，facebook等
  - 用户的视频记录：爱奇艺，抖音等
  - 。 。 。

数据预处理：数据集成

- 数据集成
  - 将多个数据源的数据整合到一个一致的数据存储中
  - 集成数据（库）时，经常出现冗余数据
    - 冗余数据带来的问题: 浪费存储、重复计算
    - 冗余的属性
    - 冗余的样本
  - 例如：
    - 用户的电商记录出现在很多app中
    - 用户的个人信息在多个app中
    - 。。。

数据预处理：

字段	说明	示例
ID_LAT_LON_YEAR_WEEK	地点、时间	ID_-0.510_29.290_2019_00
year	年份	2019
latitude	维度	-0.51
…	…	…

输入78种字段

- 检测冗余属性
  - 分析属性之间的相关性
  - 相关性分析检测冗余

$$r_{A,B} = \frac{\sum (A - \overline{A})(B - \overline{B})}{(n-1)\sigma_A \sigma_B}$$

Pearson积矩相关系数，取值范围为 [-1; 1]

➢值大于 0，则属性 A 和 B 是正相关的，值越大相关性越强

因此，表明两个属性中有一个可以作为冗余删除

➢值为 0，则 A 和 B 是独立的，它们不存在相关性

➢值小于 0，则 A 和 B 是负相关的。

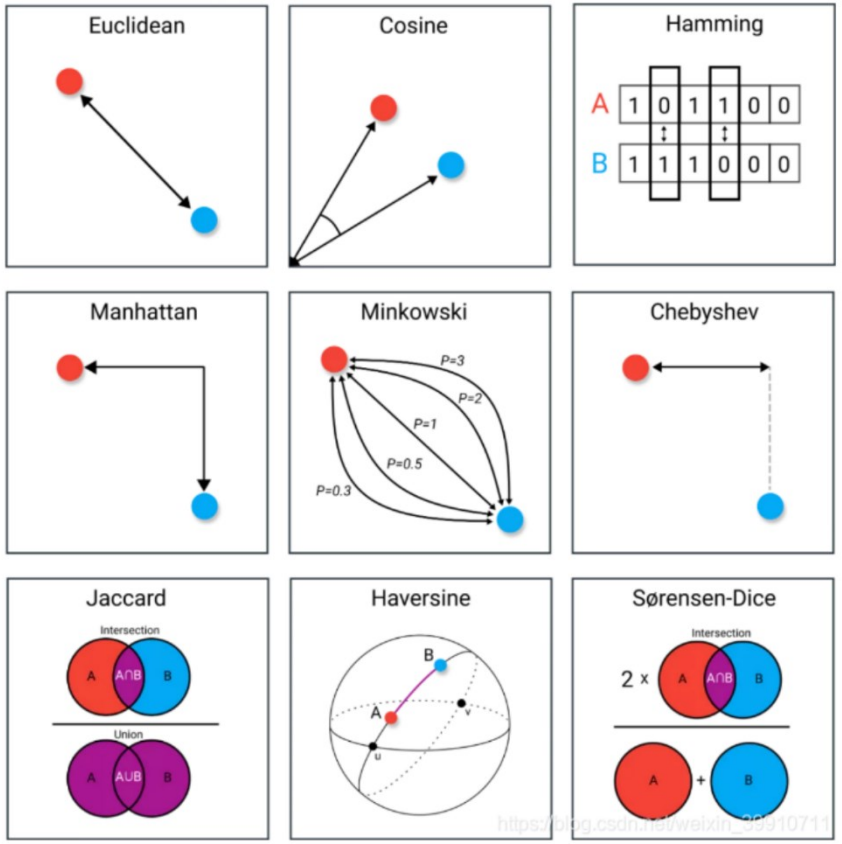
- 卡方检验：值越大，两个变量相关的可能性越大

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})}{e_{ij}}$$

卡方检验：*oij* 是联合事件 (*Ai; Bj*) 的观测频度（即实际计数），而 *eij* 是 (*Ai; Bj*)的期望频度。 卡方检验的原假设是 A 和 B 两个属性相互独立，如果可以拒绝该原假设，则我们说 A 和 B 是显著相关的。

数据预处理：数据集成

- 检测冗余样本
  - 思想：数据样本之间的相关性，数据融合、去除冗余
  - 方法：距离度量
    - 欧几里得距离
    - 汉明距离
    - 明氏距离
    - 马氏距离
    - ……
  - 方法：相似度计算
    - 余弦相似度
    - Jaccard相似度
    - ……



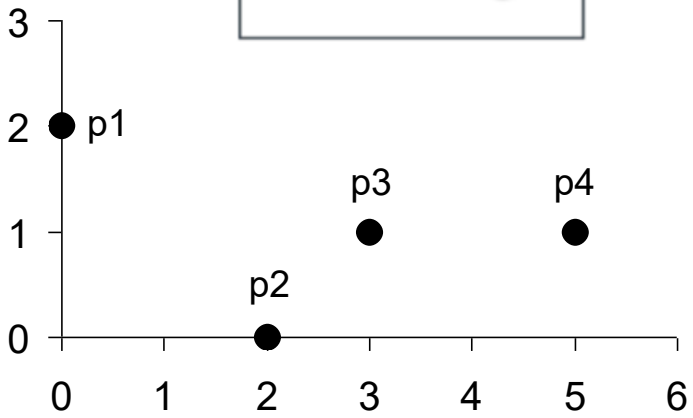
数据预处理：数据集成

- 数据的距离度量
  - 欧几里得距离(Euclidean Distance)

$$dist = \sqrt{\sum_{k=1}^{n}(p_k - q_k)^2}$$

n 表示数据p和q维度数

p_k 和 q_k 表示数据p和q的第k个属性



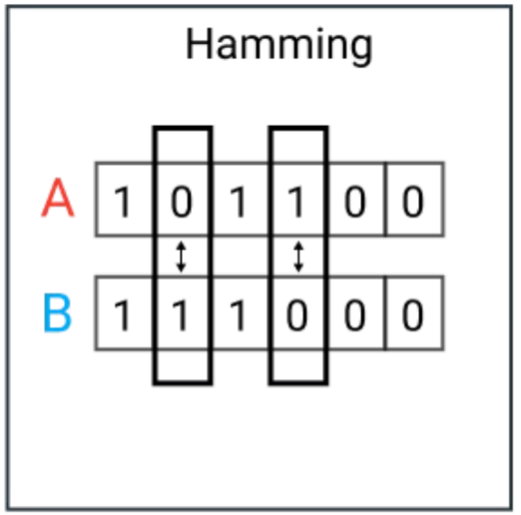
	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

对数据标准化非常重要

数据预处理：数据集成

- 数据的距离度量
  - 汉明距离(Hamming Distance)
  - 定义：两个向量之间不同值的个数
    - 字符串比较：比较两个相同长度的二进制字符串
  - 要求：向量长度相同
  - 常用：HASH场景

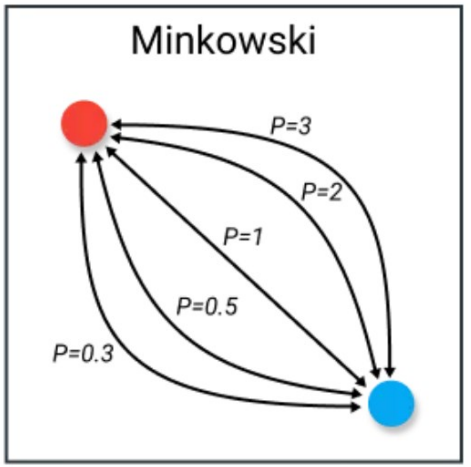


Defu Lian, Haoyu Wang, **Enhong Chen,** Xing Xie. LightRec: a Memory and Search-Efficient Recommender System. **WWW 2020**.

数据预处理：数据集成

- 数据的距离度量
  - 明氏距离（Minkowski Distance）
    - 距离度量：通用表达形式

$$dist = (\sum_{k=1}^{n} | p_k - q_k |^r)^{\frac{1}{r}}$$



r 是参数

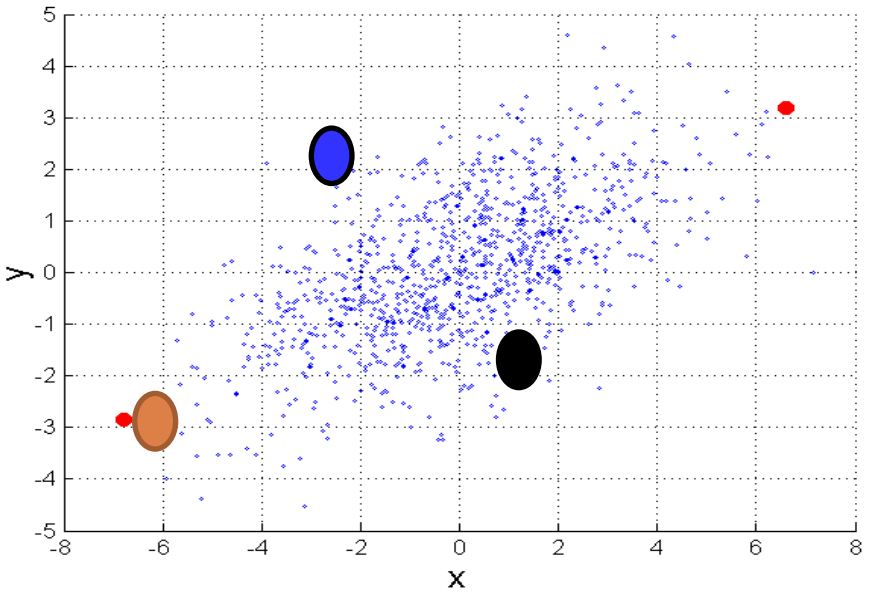
n 表示数据p和q维度数，p_k 和 q_k 表示数据p和q的第k个属性

- r=1：曼哈顿距离
- r=2：欧氏距离
- r=∞：切比雪夫距离

数据预处理：数据集成

- 数据的距离度量
  - 马氏距离：数据的协方差距离
    - 欧氏距离的扩展，考虑到各种特性之间的联系（协方差）

$$s(p-q)=(p-q)\Sigma^{-1}(p-q)^T$$



Σ 是总体样本 X的协方差矩阵

$$\Sigma_{j,k}=\frac{1}{n-1}\sum_{i=1}^{n}(X_{ij}-\overline{X}_j)(X_{ik}-\overline{X}_k)$$

- 确定未知样本集与已知样本集的相似度
- 它考虑了数据集的相关性，并且是比例不变的

红色的数据点, 欧氏距离为14.7, 马氏距离为6

数据预处理：数据集成

- 马氏距离 vs 欧氏距离
  - **假设：**以厘米为单位测量人的身高，以克（g）为单位测量人的体重。每个人被表示为一个两维向量。如：一个人身高173cm，体重50000g，表示为（173, 50000），根据身高体重来判断人的体型的相似程度
  - **已知：**小明(160, 60000)；小王(160, 59000)；小李(170，60000)。小明与谁的体型更相似？

分析：根据常识可以知道小明和小王体型相似。但是如果根据欧氏距离来判断，小明和小王的距离要远大于小明和小李之间的距离，即小明和小李体型相似

原因：不同特征的**度量标准之间存在差异**而导致判断出错

➢以克（g）为单位测量人的体重，数据分布比较分散，即方差大，

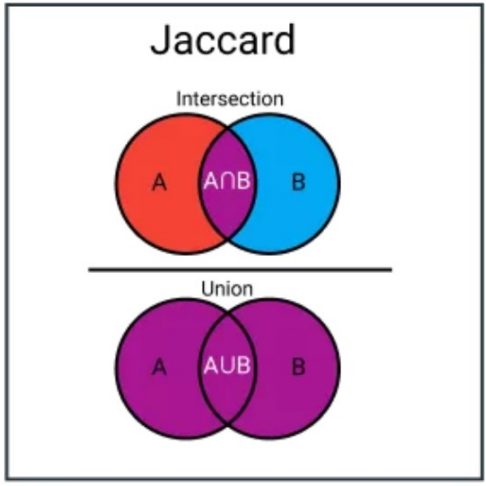
➢以厘米为单位来测量人的身高，数据分布就相对集中，方差小

马氏距离把方差归一化，使得特征之间的关系更加符合实际情况

数据预处理：数据集成

- 数据的相似度计算
  - **简单匹配 Simple Matching VS Jaccard相关系数**
  - 离散数据，属性的取值表示为0或1
  - 例：数据p和q，定义如下4个变量
    - F01：p为0、q为1的属性数量
    - F10：p为1、q为0的属性数量
    - F00：p为0、q为0的属性数量
    - F11：p为1、q为1的属性数量

$$p = (1\,0\,0\,0\,0\,0\,0\,0\,0\,0)$$
$$q = (0\,0\,0\,0\,0\,0\,1\,0\,0\,1)$$



SMC = number of matches / number of attributes
= (F11 + F00) / (F01 + F10 + F11 + F00)

Jaccard = **number of F11 matches** / **number of non-zero attributes**
= (F11) / (F01 + F10 + F11)

数据预处理：数据集成

- 数据的相似度计算
  - **简单匹配 Simple Matching VS Jaccard相关系数**

假设：存在该属性为1，不存在该属性为0

$p = (1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$

$q = (0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1)$

p和q是否相关?

F01 = 2 (p为0，q为1的属性数量)

F10 = 1 (p为1，q为0的属性数量)

F00 = 7 (p为0，q为0的属性数量)

F11 = 0 (p为1，q为1的属性数量)

SMC = (F11 + F00) / (F01 + F10 + F11 + F00)

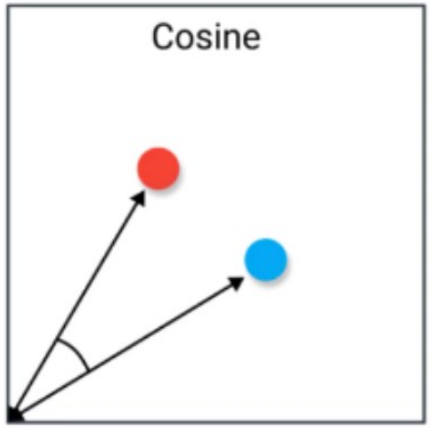
= (0+7) / (2+1+0+7) = 0.7

Jaccard = $(F_{11}) / (F_{01} + F_{10} + F_{11}) = 0 / (2 + 1 + 0) = 0$

数据预处理：数据集成

- 数据的相似度计算
  - **余弦相似性 (Cosine Similarity)**

$$\cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}.$$



  - **例：**

A = 3 2 0 5 0 0 0 2 0 0

B = 1 0 0 0 0 0 0 1 0 2

$A \bullet B$ = 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5

$||A||$ = (3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)$^{0.5}$ = (42) $^{0.5}$ = 6.481

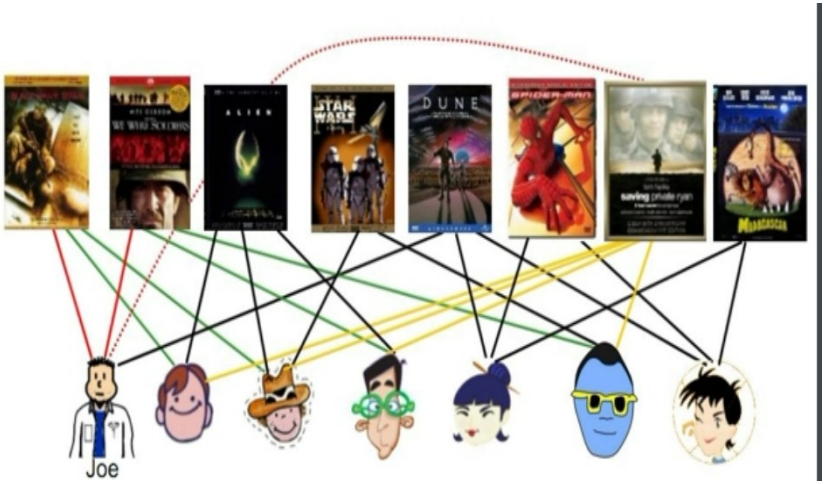
$||B||$ = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2) $^{0.5}$ = (6) $^{0.5}$ = 2.245

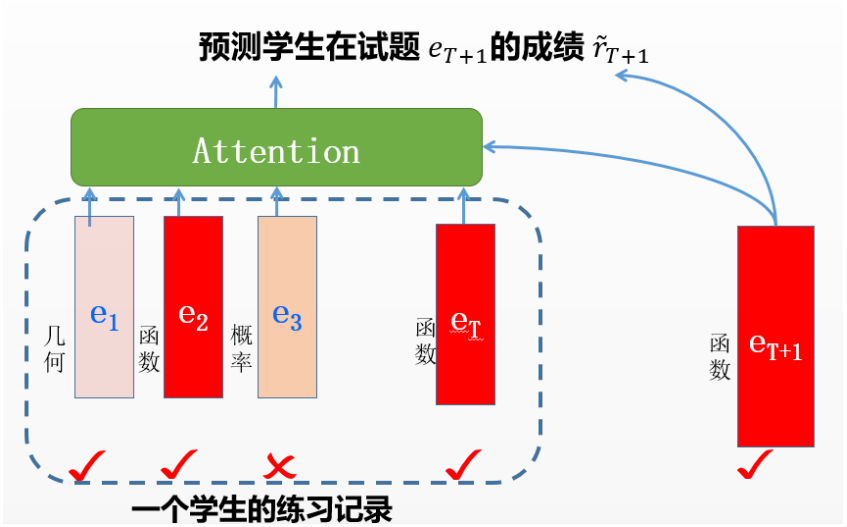
cos(A, B) = 0 .3150

思考：余弦相似度是不是一种距离？

数据预处理：数据集成

- 数据的相似度计算
  - **余弦相似度 (Cosine Similarity)**
    - 推荐系统中，协同过滤算法(UCF, ICF)—经典算法
      - 用户(向量)的相似度度量，产品(向量)的相似度度量
    - 深度学习中，训练Attention（注意力机制）的权重
      - 基于注意力机制的学生成绩预测模型





Qi Liu, Zhenya Huang, Enhong Chen,, EKT: Exercise-aware Knowledge Tracing for Student Performance Prediction, TKDE

数据预处理：数据集成

- 数据的相关性分析
  - Pearson相关系数
    - 衡量两个数据对象之间的线性关系
    - 数据标准化
  - 可以简单理解为：p和q的协方差/(p的标准差*q的标准差)

$$p_{X,Y} = \frac{\sum_{i=1}^{n}(X_i - \tilde{X})(Y_i - \tilde{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \tilde{X})^2 \sum_{i=1}^{n}(Y_i - \tilde{Y})^2}}$$

$$\rho_{X,Y} = \operatorname{corr}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

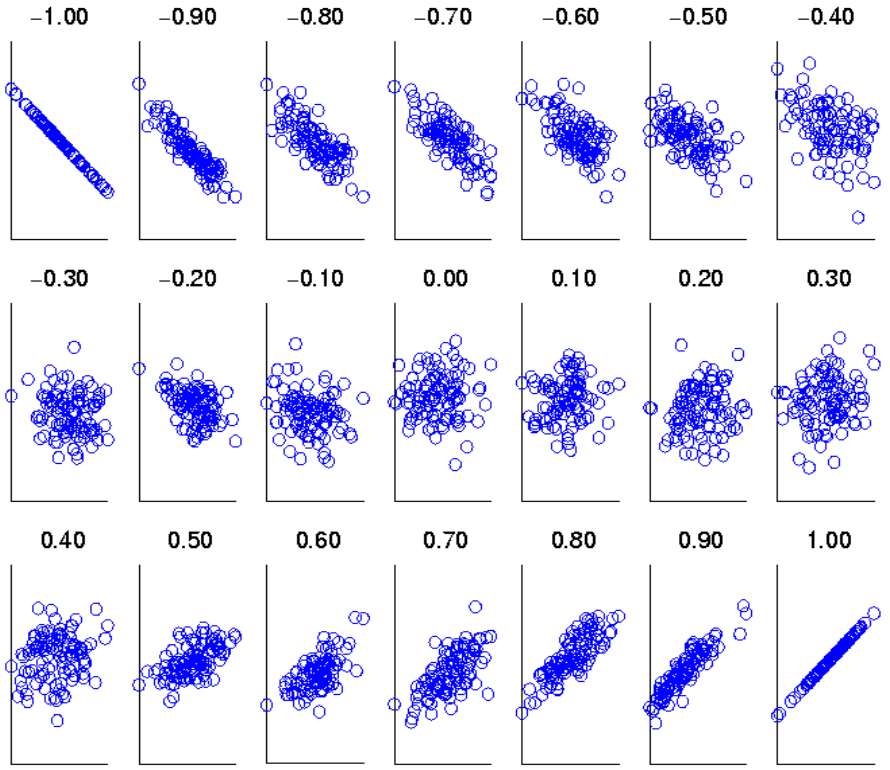
Zhenya Huang, Qi Liu, Enhong Chen, et al, Question Difficulty Prediction for READING Problems in Standard Tests, AAAI'2017

数据预处理：数据集成

- 数据的相关性
  - Pearson相关系数：衡量数据对象之间的线性关系

两个数据样本x,y各有30个属性，这些属性值随机产生，使得x和y的相关度从-1到1。图中每个小圆圈代表30个属性中的一个，其x坐标是x的一个属性的值，y坐标是y的相同属性的值

散点图显示相似度[-1, 1]



数据预处理：数据集成

- 数据的相关性分析
  - Pearson**相关系数**：衡量数据对象之间的线性关系

- 例：**问：X与Y有没有关系？**
  - X = (-3, -2, -1, 0, 1, 2, 3)
  - Y = (9, 4, 1, 0, 1, 4, 9)

$$p_{X,Y} = \frac{\sum_{i=1}^{n}(X_i - \tilde{X})(Y_i - \tilde{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \tilde{X})^2 \sum_{i=1}^{n}(Y_i - \tilde{Y})^2}}$$

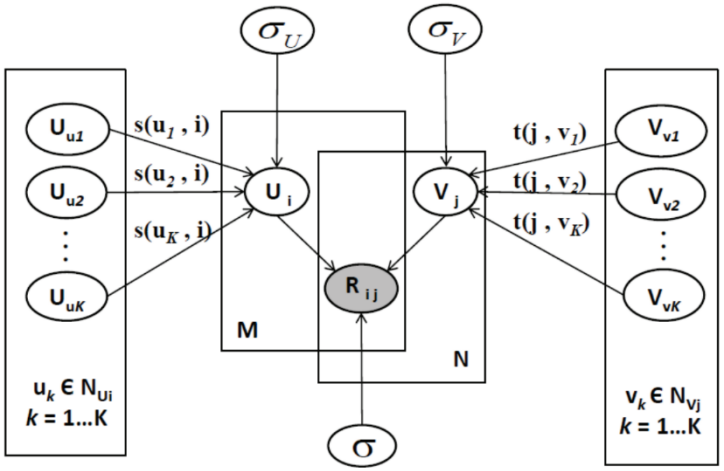
- Mean(X) = 0, Mean(Y) = 4
- Correlation= ？
  - = (-3)(5)+(-2)(0)+(-1)(-3)+(0)(-4)+(1)(-3)+(2)(0)+3(5)= 0

数据预处理：数据集成

- 数据的相关性分析
  - 有时，不同的属性产生的影响不同
  - 在计算距离，相似度时，可以赋予数据属性的权重不同 (w_k)

$$\text{similarity}(\mathbf{x},\mathbf{y}) = \frac{\sum_{k=1}^{n} w_k \delta_k s_k(\mathbf{x},\mathbf{y})}{\sum_{k=1}^{n} \delta_k}$$

$$d(\mathbf{x},\mathbf{y}) = \left(\sum_{k=1}^{n} w_k |x_k - y_k|^r \right)^{1/r}$$



Le Wu, Enhong Chen, Qi Liu, Leveraging Tagging for Neighborhood-aware Probabilistic Matrix Factorization. CIKM2012

数据预处理：数据集成

数据的相关性分析

- **无序数据：每个数据样本的不同维度是没有顺序关系的**
  - 余弦相似度、相关度、欧几里得距离、Jaccard
- **有序数据：对应的不同维度(如特征)是有顺序(rank)要求的**
  - 在信息检索中，如何判断不同检索方法返回的页面序列的优劣
  - 在推荐系统中，如何判断不同推荐序列的好坏
    - **Spearman Rank(斯皮尔曼等级)相关系数**
    - **归一化的折损累计增益(NDCG)**
    - 肯德尔相关性系数
      - kendall correlation coefficient
- 课外阅读：PageRank算法

i	相关度
1	3
2	3
3	2
4	0
5	1
6	2

方法返回结果

i	相关度
1	3
2	3
3	2
4	2
5	1
6	0

真实结果

数据预处理：数据集成

数据的相关性分析—举例

- 已知：6个网页的相关度是3，2，3，0，1，2，所以在信息检索中，最好的返回结果应当如(a)所示。
- 如果我们设计了两个检索算法，返回结果分别是(b)和(c)，请问哪个方法的结果与真实结果更相似？

i	相关度
1	3
2	3
3	2
4	2
5	1
6	0

(a)真实结果

i	相关度
1	3
2	3
3	0
4	2
5	2
6	1

(b)方法1返回结果

i	相关度
1	3
2	3
3	2
4	0
5	2
6	1

(c)方法2返回结果

数据预处理：数据集成

- 有序数据的距离度量(信息检索、推荐系统等)
  - Spearman Rank(斯皮尔曼等级)相关系数
    - 比较两组变量的相关程度
    - 当关系是非线性时，它是两个变量之间关系评价的更好指标

$$\rho_s = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$$

    - ρ_s：表示斯皮尔曼相关系数
    - d_i^2：表示每一对样本之间等级的差
    - n：表示样本容量
  - ρ_s的范围：-1 to 1 (正相关:$\rho_s > 0$，负相关:$\rho_s < 0$，不相关:$\rho_s = 0$)

数据预处理：数据集成

- 有序数据的距离度量(信息检索、推荐系统等)
  - Spearman Rank(斯皮尔曼等级)相关系数
    - 比较两组变量的相关程度
    - 当关系是非线性时，它是两个变量之间关系评价的更好指标

$$d_i = Y_i - X_i$$

$$\rho_s = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$$

- ρ_s：表示斯皮尔曼相关系数
- d_i^2：表示每一对样本之间等级的差
- N：表示样本容量

  - ρ_s的范围：-1 to 1 (正相关:$\rho_s > 0$，负相关:$\rho_s < 0$，不相关:$\rho_s = 0$)

数据预处理：数据集成

- 有序数据的距离度量(信息检索、推荐系统等)
  - NDCG(Normalized Discounted cumulative gain)
    - CG(累计增益)：只考虑到了相关性的关联程度，没有考虑每个推荐结果处于不同位置对整个推荐效果的影响

$$CG_k = \sum_{i=1}^{k} rel_i$$

rel_i表示处于位置 ii 的推荐结果的相关性

- DCG(折损累计增益)：就是在每一个CG的结果上处以一个折损值，目的就是为了让排名越靠前的结果越能影响最后的结果

$$DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i}-1}{log_2(i+1)}$$

- i表示推荐结果的位置， i越大，则推荐结果在推荐列表中排名越靠后推荐效果越差，DCG越小

Qi Liu, Yong Ge, Enhong Chen, and Hui Xiong. Personalized Travel Package Recommendation. ICDM'2011, **(Best Research Paper Award)**

数据预处理：数据集成

- **有序数据的距离度量(信息检索、推荐系统等)**
  - **NDCG(** Normalized Discounted cumulative gain **)**
    - **NDCG：**由于搜索结果随着检索词的不同，返回的数量不一致，而DCG是一个累加的值，没法针对两个不同的搜索结果进行比较，因此需要标准化处理，这里是除以IDCG:

$$NDCG_k = \frac{DCG_k}{IDCG_k}$$

IDCG为理想（ideal）情况下最大的DCG值，指推荐系统为某一用户返回的最好推荐结果列表(或者，真实的数据序列)

数据预处理：数据集成

$$NDCG_k = \frac{DCG_k}{IDCG_k}$$

- 例，假设一个推荐系统为用户推荐了3部电影，顺序为A，B，C，用户实际对这三部电影的偏好为B > A > C, 假定A，B，C三部电影的相关性分数分别为2，3，1，那么对于系统返回的结果有：
  - CG@3 = 2 + 3 + 1= 6
  - DCG@3 = 3 + 4.42 + 0.5 = 7.92

$$CG_k = \sum_{i=1}^{k} rel_i$$

- 理想情况下，系统给出的电影排序应该为B，A，C
  - IDCG@3 = 7 + 1.89 + 0.5 = 9.39
- 可以计算NDCG@3
  - NDCG@3 = 7.92 / 9.39 = 0.84

$$DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i}-1}{log_2(i+1)}$$

i	movie	rel	$\frac{2^{rel_i}-1}{log_2(i+1)}$
1	A	2	3
2	B	3	4.42
3	C	1	0.5

方法返回结果

i	movie	rel	$\frac{2^{rel_i}-1}{log_2(i+1)}$
1	B	3	7
2	A	2	1.89
3	C	1	0.5

真实结果

数据预处理：数据集成

课后阅读

- Defu Lian, Haoyu Wang, Enhong Chen, Xing Xie. LightRec: a Memory and Search-Efficient Recommender System. WWW 2020.
- Qi Liu, Zhenya Huang, Enhong Chen,, EKT: Exercise-aware Knowledge Tracing for Student Performance Prediction, TKDE
- Zhenya Huang, Qi Liu, Enhong Chen, et al, Question Difficulty Prediction for READING Problems in Standard Tests, AAAI'2017
- Qi Liu, Yong Ge, Enhong Chen, and Hui Xiong. Personalized Travel Package Recommendation. ICDM'2011, (Best Research Paper Award)
- 信息检索经典研究：PageRank算法