数据科学导论
Introduction to Data Science

第二章 数据入门

刘 淇
Email: qiliuql@ustc.edu.cn

题目	指导师兄	邮箱
离散制造过程中典型工件的质量符合率预测	顾垠	gy128@mail.ustc.edu.cn
乘用车细分市场销量预测	刘烨	liuyer@mail.ustc.edu.cn
互联网金融新实体发现	杜逸超	duyichao@mail.ustc.edu.cn
互联网新闻情感分析	孙睿军	rjsun@mail.ustc.edu.cn

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
  - 数据清洗
  - 数据集成
  - 数据变换
  - 数据规约

为什么进行数据预处理

- 现实世界的数据是“脏的”——数据多了，什么问题都会出现
  - 滥用缩写词　---中科大，　科大，中国科大，USTC
  - 数据输入错误　---裤子大、国科大…
  - 数据中的内嵌控制信息　---E3=F3*C3
  - 不同的惯用语　----南七技校
  - 重复记录
  - 丢失值
  - 拼写变化
  - 不同的计量单位
  - 过时的编码
  - 含有各种噪声，如中学生年龄

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
    - e.g. 重复值或者空缺值将会产生不正确的或者误导人的统计

- 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

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

- 数据清理任务
  - 填写空缺的值
  - 识别离群点和平滑噪声数据
  - 纠正不一致的数据

数据预处理：数据清理-缺失值处理

- 删除法:
  - 忽略元组
  - 去掉该属性
  - 改变权重
- 插补法
  - 特殊值填充
    - 空值作为一种特殊的属性值来处理，它不同于其他的任何属性值
  - 使用属性的均值或中位数填充空缺值
  - 使用最可能的值填充缺失值
    - 热卡填充（Hot deck imputation，或就近补齐）
    - K最近距离邻法
    - 利用回归等估计方法
    - 期望值最大化方法（EM算法）

数据预处理：数据清理-缺失值处理

- 热卡填充
  - 在完整数据中找到一个与它最相似的对象，然后用这个相似对象的值来进行填充。
- K最近距离邻法:
  - 先根据欧式距离或相关分析来确定距离具有缺失数据样本最近的K个样本，将这K个值加权平均来估计该样本的缺失数据。
- 回归法
  - 基于完整的数据集，建立回归方程（模型）。对于包含空值的对象，将已知属性值代入方程来估计未知属性值，以此估计值来进行填充。

数据预处理：数据清理-缺失值处理

- 期望值最大化方法（EM算法）
  - 在缺失类型为随机缺失的条件下，通过观测数据的边际分布可以对未知参数进行极大似然估计。
  - EM算法是一种在不完全数据情况下计算极大似然估计或者后验分布的迭代算法。

  - EM算法在每一迭代循环过程中交替执行两个步骤：
    - E步（Excepctaion step,期望步），在给定完全数据和前一次迭代所得到的隐含参数估计的情况下计算完全数据对应的对数似然函数的条件期望；
    - M步（Maximzation step，极大化步），用极大化对数似然函数以确定参数(更新隐含参数)的值，并用于下步的迭代。

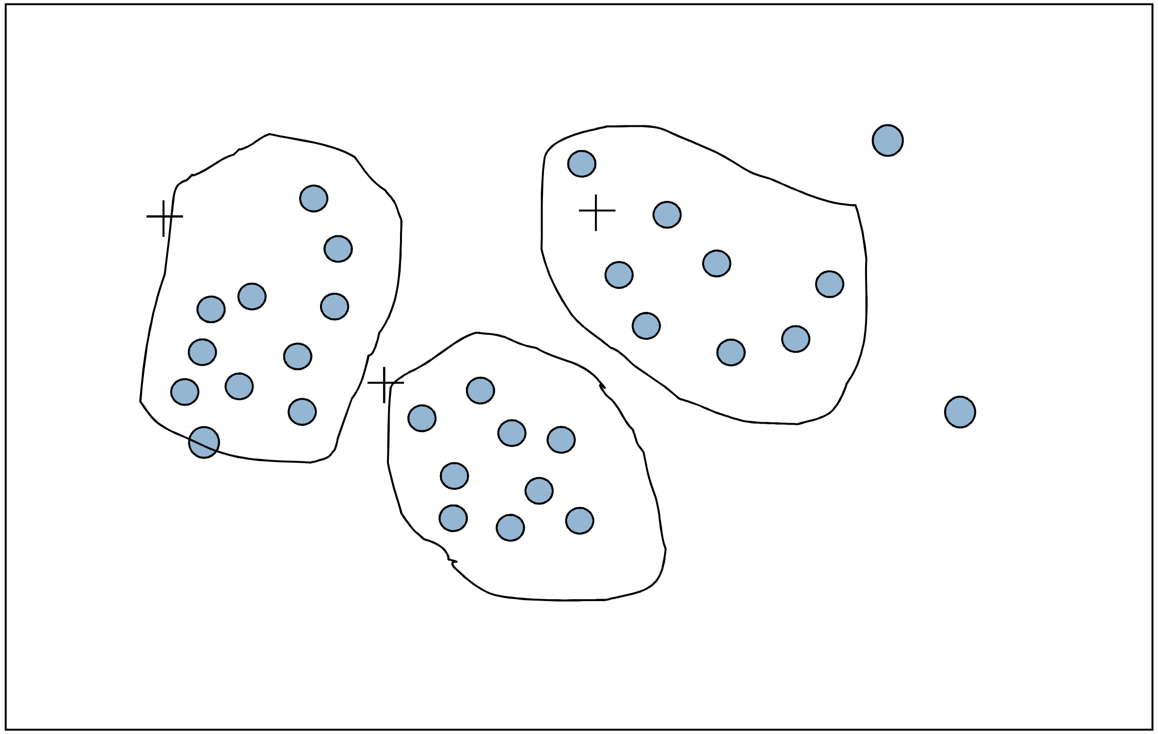
数据预处理：数据清理-噪声数据

- 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，常用的处理方法：
  - 分箱(binning):
    - 首先排序数据，并将他们分到等深的箱中
    - 然后可以按箱平均值平滑、按箱中值平滑、按箱边界平滑等等
  - 回归
    - 通过让数据适应回归函数来平滑数据
  - 聚类：
    - 监测并且去除孤立点

数据平滑的分箱方法

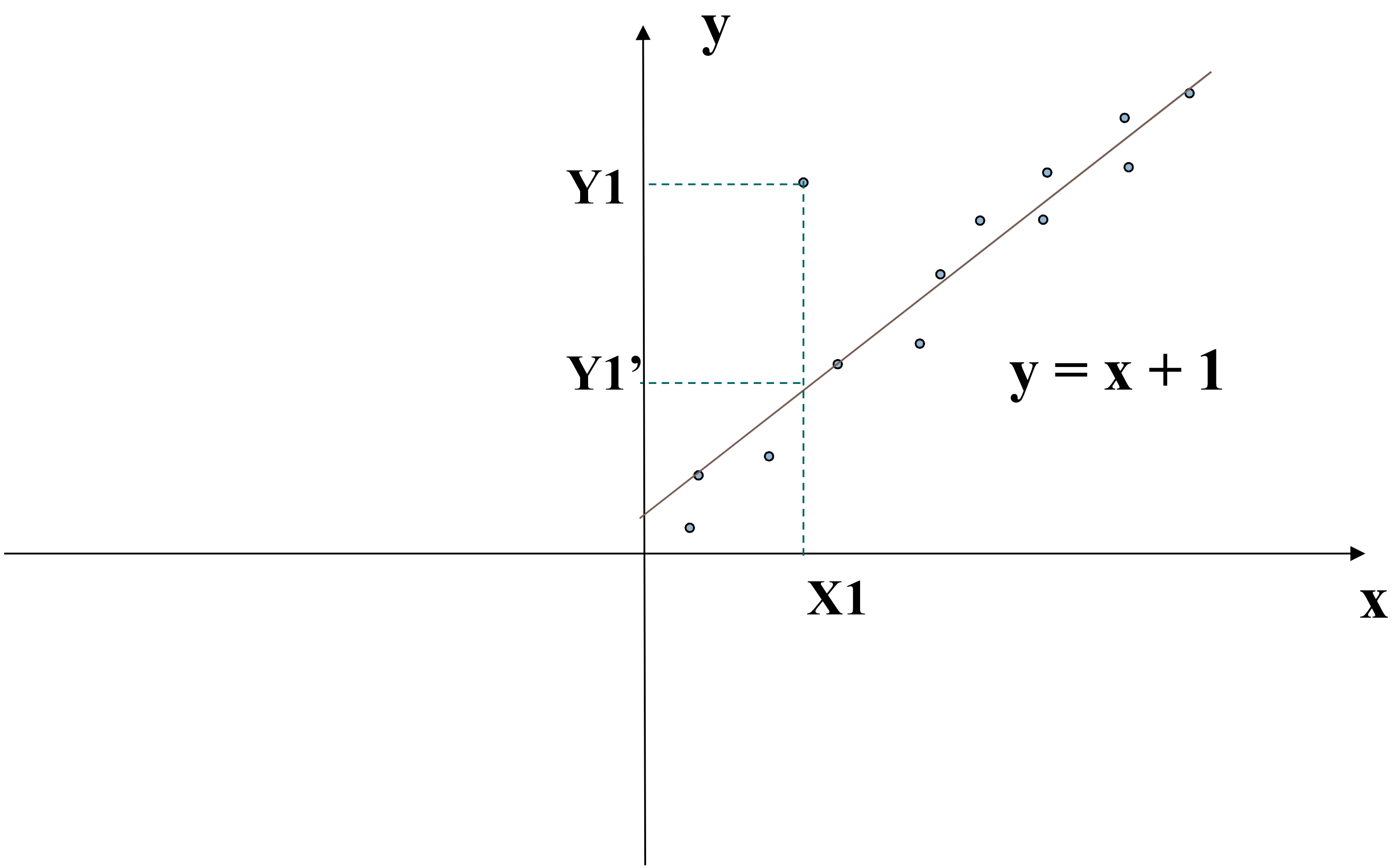
- price的排序后数据（单位：美元）：4，8，15，21，21，24，25，28，34
- 划分为（等深的）箱：
  - 箱1：4，8，15
  - 箱2：21，21，24
  - 箱3：25，28，34
- 用箱平均值平滑：
  - 箱1：9，9，9
  - 箱2：22，22，22
  - 箱3：29，29，29
- 用箱边界平滑：
  - 箱1：4，4，15
  - 箱2：21，21，24
  - 箱3：25，25，34

聚类



- 通过聚类分析检测离群点，消除噪声
  - 聚类将类似的值聚成簇。直观的，落在簇集合之外的值被视为离群点

通过线性回归的平滑处理



- 通过线性回归模型，对不符合回归的数据进行平滑处理

不一致数据的处理

- 通过线性回归模型，对不符合回归的数据进行平滑处理

数据预处理：数据集成

- 数据集成：
  - 将多个数据源中的数据整合到一个一致的数据存储中
- 集成多个数据库时，经常会出现冗余数据
  - 相关分析冗余检测

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

冗余数据带来的问题：
浪费存储、重复计算

  - χ^2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

- 数据的距离度量（可以用来进行数据融合、去除冗余）
  - Euclidean Distance（欧几里得距离）

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

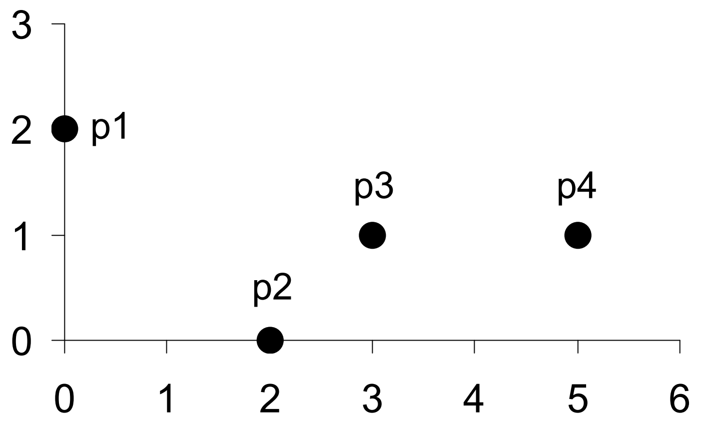
Where n is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects p and q.

  - Standardization is necessary, if scales differ.
  -

数据预处理：数据集成

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

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



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

	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

数据预处理：数据集成

- 数据的距离度量
  - Minkowski Distance(明氏距离) is a generalization of Euclidean Distance

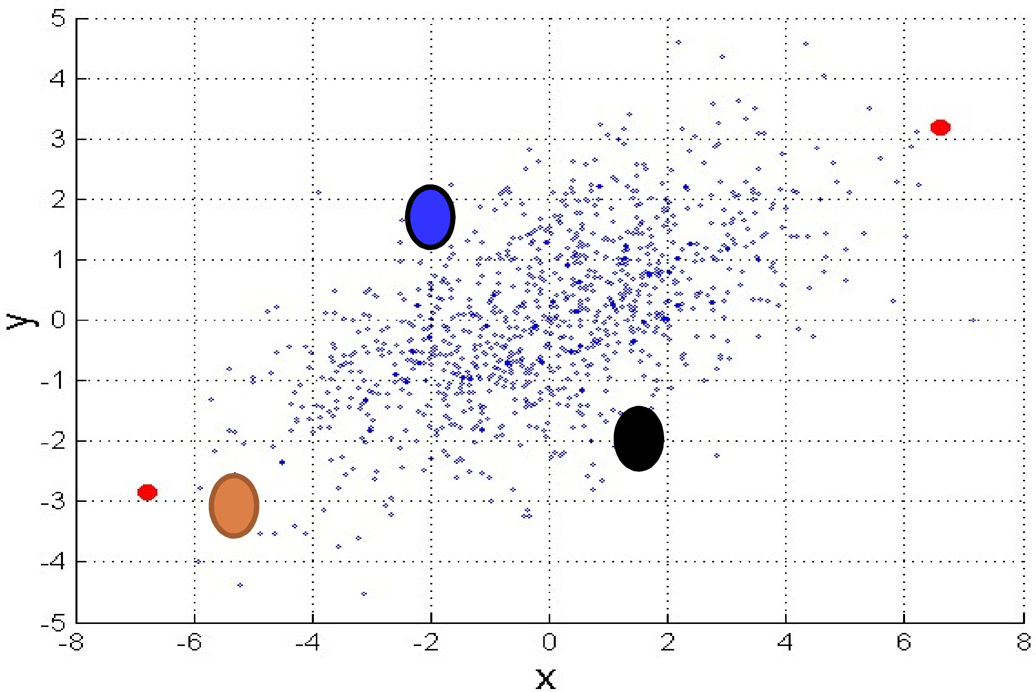
$$dist = \left(\sum_{k=1}^{n} | p_k - q_k |^r \right)^{\frac{1}{r}}$$

Where r is a parameter, n is the number of dimensions (attributes) and pk and qk are, respectively, the kth attributes (components) or data objects p and q.

数据预处理：数据集成

- 数据的距离度量
  - 马氏距离

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



Σ 是总体样本 *X的协方差矩阵*

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

Determining similarity of an unknown Sample set to a known one. It takes Into account the correlations of the Data set and is scale-invariant.

For red points, the Euclidean distance is 14.7, Mahalanobis distance is 6.

- 马氏距离，实例：

如果以厘米为单位来测量人的身高，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），根据身高体重的信息来判断体型的相似程度。

已知小明**（160,60000）**；小王**（160,59000）**；小李**（170，60000）**。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。这是因为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，而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

- 数据的距离度量
- Common situation is that objects, *p* and *q*, have only binary attributes（**0 或 1**）
- Compute similarities using the following quantities

  F01 = the number of attributes where *p* was 0 and *q* was 1

  F10 = the number of attributes where *p* was 1 and *q* was 0

  F00 = the number of attributes where *p* was 0 and *q* was 0

  F11 = the number of attributes where *p* was 1 and *q* was 1

- **Simple Matching** and **Jaccard** Coefficients（Jaccard系数）

  SMC = number of matches / number of attributes

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

  J = **number of 11 matches** / **number of non-zero attributes**

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

- 数据的距离度量

Simple Matching and **Jaccard** Coefficients（Jaccard系数）

p = 1 0 0 0 0 0 0 0 0 0
q = 0 0 0 0 0 0 1 0 0 1

F01 = 2 (the number of attributes where p was 0 and q was 1)
F10 = 1 (the number of attributes where p was 1 and q was 0)
F00 = 7 (the number of attributes where p was 0 and q was 0)
F11 = 0 (the number of attributes where p was 1 and q was 1)

SMC = (F11 + F00) / (F01 + F10 + F11 + F00)
= (0+7) / (2+1+0+7) = 0.7

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

数据预处理：数据集成

- 数据的距离度量
  - Cosine Similarity（余弦相似性 ）
  - If d_1 and d_2 are two document vectors, then

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \; ||d_2||,$$

where $\bullet$ indicates vector dot product and $||\ d\ ||$ is the length of vector d.

- Example:

d_1 = **3 2 0 5 0 0 0 2 0 0**

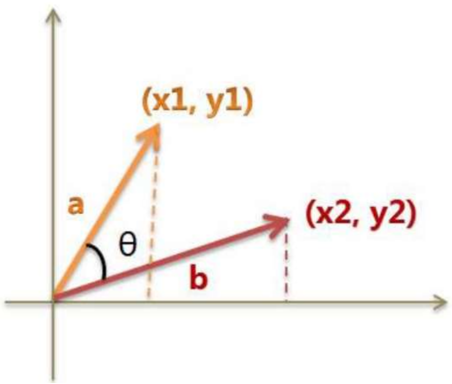
d_2 = **1 0 0 0 0 0 0 1 0 2**

$d_1 \bullet d_2$= 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5

$||d_1||$ = (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

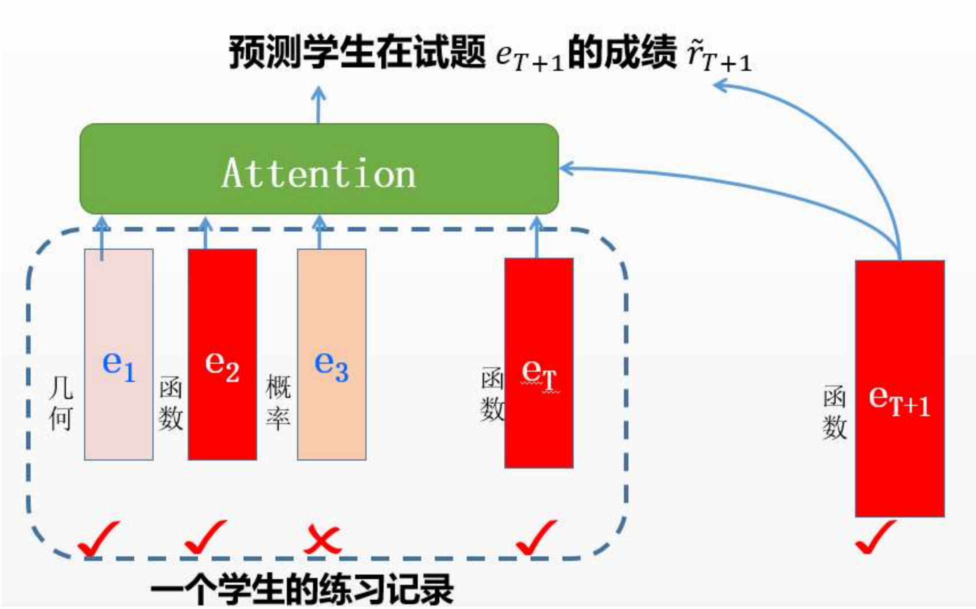
$||d_2||$ = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2) $^{0.5}$ = (6) $^{0.5}$ = 2.245

$\cos(d_1, d_2)$ = .3150



数据预处理：数据集成

- 数据的距离度量
  - **Cosine Similarity（余弦相似性 ）**
    - 推荐系统中，判断用户兴趣向量(User)与产品向量(Item)的相似度
    - 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



Yu Su, Qi Liu, Zhenya Huang, Enhong Chen, et al. Exercise-Enhanced Sequential Modeling for Student Performance Prediction, AAAI 2018.

数据预处理：数据集成

- 数据的距离度量
  - **Correlation(相关度)** measures the linear relationship between objects
  - To compute correlation, we standardize data objects, p and q, and then take their dot product
  - 可以简单理解为：p和q的协方差/(p的标准差*q的标准差)

$$p'_k = (p_k - mean(p)) / std(p)$$

$$q'_k = (q_k - mean(q)) / std(q)$$

$$correlation(p, q) = p' \bullet q' / (n - 1)$$

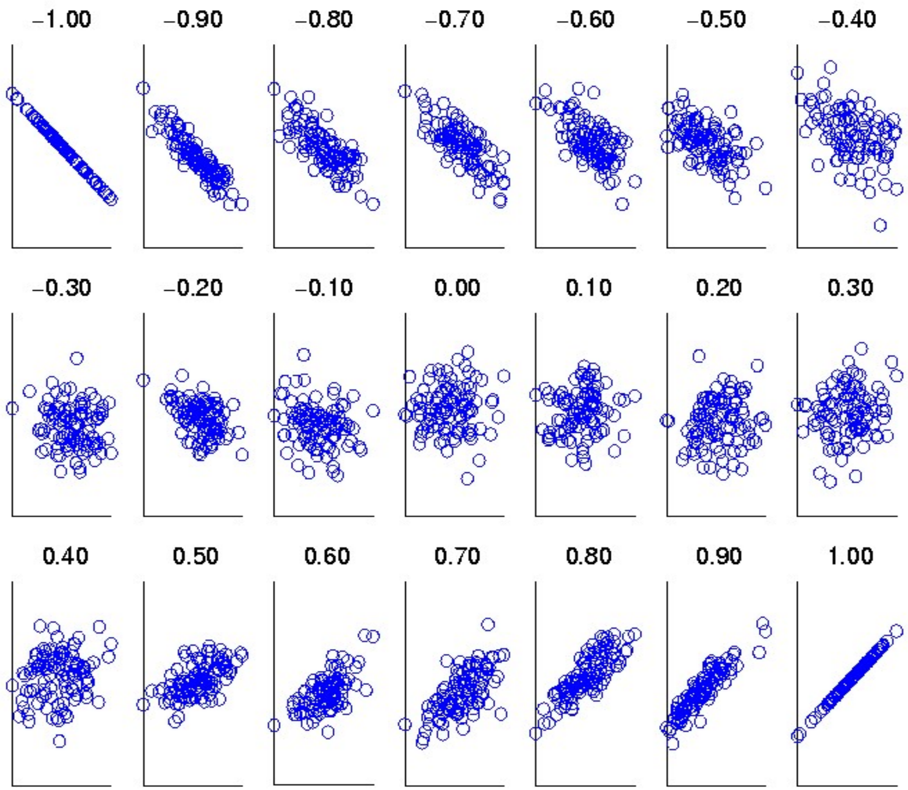
数据预处理：数据集成

- 数据的距离度量
  - **Correlation** measures the linear relationship between objects
  - To compute correlation, we standardize data objects（z-score）, p and q, and then take their dot product
  - 可以简单理解为：p和q的协方差/(p的标准差*q的标准差)

$$corr(p,q) = \frac{\sum_{k}(p_k - \bar{p})(q_k - \bar{q})}{\sqrt{\sum(p_k - \bar{p})^2}\sqrt{\sum(q_k - \bar{q})^2}}$$

数据预处理：数据集成

- 数据的距离度量 **Correlation** measures the **linear** relationship between objects



Scatter plots showing the similarity from –1 to 1.

两个数据对象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，而y坐标是y的相同属性的值

数据预处理：数据集成

- 数据的距离度量

  Correlation measures the linear relationship between objects

- X = (-3, -2, -1, 0, 1, 2, 3)
- Y = (9, 4, 1, 0, 1, 4, 9)

X与Y有没有关系？

$$corr(p,q) = \frac{\sum_k (p_k - \bar{p})(q_k - \bar{q})}{\sqrt{\sum (p_k - \bar{p})^2}\sqrt{\sum (q_k - \bar{q})^2}}$$

- 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

- 数据的距离度量
  - May not want to treat all attributes the same.
    - Use weights w_k which are between 0 and 1 and sum to 1.

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

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

数据预处理：数据集成

- 数据的距离度量---练习题1
- 对于下面的x和y，计算指定的相似性或距离度量。余弦相似度、相关度、欧几里得距离、Jaccard。
  - X和Y是什么相关关系？

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / \|d_1\| \, \|d_2\|$$

X = (0, 1, 0, 1), Y = (1, 0, 1, 0)

$$corr(p,q) = \frac{\sum_k (p_k - \bar{p})(q_k - \bar{q})}{\sqrt{\sum (p_k - \bar{p})^2} \sqrt{\sum (q_k - \bar{q})^2}}$$

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

$$J = (F11) / (F01 + F10 + F11)$$

数据预处理：数据集成

数据的距离度量

- 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

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

数据的距离度量—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，它们的返回结果分别是(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
  - X = (a, b, c, d, e, f)
  - Y = (c, a, e, d, f, b)

$$\boldsymbol{d_i = Y_i - X_i}$$

  - $d_i^2 = (4, 1, 4, 0, 1, 16)$

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

  - $\rho = 1 - \frac{6(26)}{6(36-1)} \approx 1 - 0.743 = 0.257$

数据预处理：数据集成

数据的距离度量—课后思考

- Spearman Rank相关度与Pearson相关度之间的联系与区别？

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

$$corr(p,q) = \frac{\sum_k (p_k - \bar{p})(q_k - \bar{q})}{\sqrt{\sum (p_k - \bar{p})^2}\sqrt{\sum (q_k - \bar{q})^2}}$$

斯皮尔曼相关系数被定义成等级数据变量
(rank/order variables)之间的皮尔逊相关系数

数据预处理：数据集成

数据的距离度量--练习题2

- 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，它们的返回结果分别是(b)和(c)，请问哪个方法的结果与真实结果更相似（给出**Spearman**计算结果）。

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

只考虑了每个位置**(entry)**的数据与真实数据的顺序差异，但是没有考虑到不同位置**(entry)**的重要性差异

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

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，因此需要归一化处理，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

- 例，假设搜索返回的6个物品，其相关性分别是 3、2、3、0、1、2
  - CG@6 = 3+2+3+0+1+2
  - DCG@6 = 7+1. 89+3.5+0+0. 39+1.07 = 13.85
- 假如我们实际召回了8个物品，除了上面的6个，还有两个物品，第7个相关性为3，第8个相关性为0。那么在理想情况下的相关性分数排序应该是：3、3、3、2、2、1、0、0。计算IDCG@6：
  - IDCG = 7+4.42+3.5+1.29+1.16+0.36 = 17.73
- 可以计算：
- NDCG@6 = 13.85/17.73 = 0.78

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

i	rel
1	3
2	2
3	3
4	0
5	1
6	2

方法返回结果

i	rel
1	3
2	3
3	3
4	2
5	2
6	1

真实结果

数据的距离度量--练习题3

- 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，它们的返回结果分别是(b)和(c)，请问哪个方法的结果与真实结果更相似（根据**NDCG**的计算结果）。

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

可以只列出计算公式，不用给出计算结果