离散数学下

关系 Relations

关系及其性质

1. 定义：
   - 二元关系：集合A与B的二元关系R是$A\times B$（笛卡尔积）的一个子集；（满足某规律的有序对的全体）；
   - n元关系：集合$A_1, A_2 \ldots, A_n$的n元关系是其笛卡尔积$A_1\times A_2\times \ldots A_n$的一个子集；
   - 关系本质上是n元组的集合；
   - 记号表示：
     - 从A集合到B集合的关系R写作$R : A\times B$者$R : A, B$；
     - aRb表示$(a, b)\in R$，读作a相关于b；
   - 补关系（Complementary Relations）：

$$\not R : \equiv \{(a, b) | (a, b) \notin R\} = (A\times B) - R$$

   - 逆关系（Inverse Relations）：

$$R^{-1} : \equiv \{(b, a) | (a, b) \in R\}$$

2. 函数（functions）：
   $f : A\rightarrow B$是$A\times B$的关系的一种特殊情况；
3. 单一集合的关系：
   集合A与A的二元关系R是$A^2 = A\times A$的一个子集，称为集合A的关系；
   - 同等关系（Identity Relations）：$I_A = \{(a, a) | a\in A\}$
   - 关系计数:
     - 有着n个元素的集合A的关系个数：2^{n^2}；

        - 从A到A的函数个数：n^n（增长更快）；
4. **关系的性质（重要）：**
    - 自反（Reflexivity）：$\forall a \in A, aRa$
        - 反自反（Irreflexivity）：R是反自反当且仅当$\not R$是自反的；
        - 反自反与“不自反”是不必要不充分的关系；
    - 对称（Symmetry）：$R = R_{-1}$
        - 反对称（Antisymmetry）：$\forall a \neq b, (a, b) \in R \rightarrow (b, a) \notin R$
            - 性质：$R \cap R^{-1} \subseteq I_A$
        - 非对称（Asymmetry）：$(a, b) \in R \rightarrow (b, a) \notin R$
        - 非对称的关系都是反自反的；
    - 传递（Transitivity）：
        - $\forall a, b, c, (a, b) \in R \wedge (b, c) \in R \rightarrow (a, c) \in R$
        - 例：$R = \{(3, 4)\}$是传递的；
        - 非传递（Intransitivity）：不传递时即为非传递；
    - 自反关系计数：
        - 集合A中有n个元素，有$2^{n(n-1)}$个自反关系
5. 联结关系（Combining Relations）：
    - 交并补差
    - 组合关系（Composition）：$S \circ R$，读作R和S的组合；
    - 关系的幂（Power of Relations）：$R^1 = R, R^{n+1} = R^n \circ R$
        - 定理：定义在集合A上的关系R是传递的当且仅当$\forall n, R^n \subseteq R$

n元关系

1. 关系数据库（Relational Databases）：即为n元关系R
    - 主键（primary key）：当R中的某一个域有这样的一个n元组能决定这个元组，即元组间在这个域上的值是不相等的，即称其为主键；
    - 组合键（Composite Key）：主键的笛卡尔积；
    - 数据库的内延（intension）与外延（extension）；
    - 操作符：
        - 选择操作符：s_c，选择满足某条件的关系中的元组；
        - 投射操作符：$P_{\{i_k\}}$
        - 联结操作符：$J(R_1, R_2)$
2. 数据挖掘关联规则

关系表示

1. 0-1矩阵表示：（矩阵运算支持关系运算）

    - $S\circ R = M_R \odot M_S$，其余略
    - 用矩阵M_R表示$R : A\times B$，其中$m_{ij} = 1 \Leftrightarrow (a_i, b_j) \in R$
        - 自反：要求主对角线全为1；
        - 反自反：要求主对角线全为0；
        - 对称：要求矩阵是对称矩阵；
        - 反对称：值为1的元素的对称位置的元素值必须是0；
        - 传递：矩阵做布尔幂运算后与原矩阵一致；
2. 有向图表示：
    - 自反：所有点都有自接环；
    - 反自反：所有点都没有自接环；
    - 对称：两点之间若有边，那一定是双向的；
    - 反对称：两点之间若有边，那一定是单项的；
    - 注意：若只有两两成对的点之间有联通，点对之间无联通，此时仍是传递；
3. 总结：

关系特征、关系图、关系矩阵

关系特征	关系图特征	关系矩阵特征
自反	每一结点处有一环	对角线元素均为1
反自反	每一结点处有无环	对角线元素均为0
对称	两结点间有相反的两边同时出现	矩阵为对称矩阵
反对称	两结点间没有相反的边成对出现	当分量$C_{ij}=1(i\neq j)$时$C_{ji}=0$
传递	如果结点v_1,v_2间有边，v_2,v_3间有边，则v_1,v_3间必有边。	$R^n \subseteq R$

闭包关系 Closures of Relations

1. 定义：关系R的P闭包就是包含R且**具有P中一切性质**的关系中的最小的那一个；（对“最小”的理解：关系矩阵中的1的个数最少）
2. 分类：
   - 自反闭包（reflexive closure）：称$R \cup I_A$是R的自反闭包，记为$r(R)$；
     - 定理：$r(R)$是包含R，具有自反性的最小关系；
     - 推论：R是自反闭包当且仅当R是自反的
   - 对称闭包（symmetric closure）：称$R \cup R^{-1}$是R的对称闭包，记为$s(R)$；
     - 定理：R是A上的二元关系，则$R \cup R^{-1}$是对称的且包含R的最小关系；
     - 推论：当且仅当R是对称闭包时，R具有对称性；
   - 传递闭包：记为$t(R)$或R^*，求法见下：
     - 图中的路径（二元关系）：从a到b的长度为n的路径是n个序偶的序列$(a, x_1), (x_1, x_2), \ldots, (x_{n-1}, b)$
       - 空序列表示长度为0的从a到a的路径；
       - 从a到a的长度≥ 1的路径称为环（circuit/cycle）
       - 若R^n存在从a到a的长度为n的路径，记为$(a, b) \in R^n$
     - 定理一：R的传递闭包为$R^\infty = R^1 \cup R^2 \cup R^3 \cup \ldots = \cup_{i=1}^{\infty} R^i$；
     - 定理二：对于集合$A, A = |n|$，R为A的一个关系，则$R^\infty = \cup_{i=1}^{n} R^i$，求解的时间复杂度为$O(n^4)$；
     - 求传递闭包的算法：**Warshall 算法（重要）：**
       - 定义：
         - 内部节点（interior vertices）：从a到b的路径上经过的中间点（不含a, b）；
         - W_k：矩阵中为1的点W_{ij}^k表示从第i个点（v_i）到第j个点（v_j）所经过的点都是集合$\{v_1, v_2, \ldots, v_k\}$中的点；$W_0$即原始的关系矩阵，所有的边都没有内部节点；$W_n = R^*$；
     - 求法：（从递推过程来看，本算法是找能进入v_k的点与v_k出去能到达的点，再把这些点组合成内部节点为v_k的路径并添加进矩阵里；从整体过程来看就是逐个点延路径拓展并取并集）
       - 1. 照抄上一个矩阵（W_{k-1}）的1；
       - 2. 找W_{k-1}的第k列找到值为1的元素的行数，与W_{k-1}的第k行找到值为1的元素的列数逐一组合得到W_k中新的坐标为1的点并填入；
     - 时间复杂度：$O(n^3)$；
   - 反自反是没有闭包关系的；

等价关系 Equivalence Relations

1. 定义：A上的二元关系R，如果R自反、对称、传递，则R为等价关系，其中$(a,b)\in R$，称为a与b等价；
2. 集合的划分（partition）：（切蛋糕）
   - 定义：把集合A划分为若干子集A_i满足一下条件时，集合$P_r(A)=\{A_1,A_2,...,A_n,...\}$称为$A$的划分；
     - $i\neq j$时，$A_i\cap A_j=\emptyset$；
     - A中的任意元素都能有有对应的一个子集；
     - 所有子集的并为集合A；
   - 定理1：集合的每个划分内的元素都等价，即若$aRb\Leftrightarrow a,b$属于同一个划分块（block），则R为P决定的一个定义在A上的等价关系；
   - 引理1：$a\in A,b\in A$，R为A的等价关系，则$aRb\Leftrightarrow R(a)=R(b)$
   - 定理2：R是定义在A上的等价关系，P是所有不同的关系集合$R(A)$的集合，则P是A的一个划分，R是由P决定的等价关系；
3. 等价类（euqivalence classes）：
   - 定义：R为等价关系，a的等价类定义为$R(a)$或$[a]_R:\equiv\{b|aRb\}$，即等价于a的所有元素的集合，其中每个元素都被叫做$[a]_R$的代表元；关系R可以看做是一个函数f；
   - 划分P包含的所有R的等价类$\{[a_1],[a_2],[a_3],...,[a_n]\}$组成的集合称为商集（quotient set）（或 the partition of A induced by R 或 A modulo R），记为A/R（用有向图表示，每个等价类记为一个连通点集）；
   - 找等价类的方法：找不属于A的某一等价类的元素b计算$R(b)$，重复直至A中所有元素都有自己属于的等价类；
   - 商集的元素个数称为R的秩；
4. 等价关系的运算：
   - 若R_1,R_2是A上的等价关系，则$R_1\cap R_2$也是等价关系，称为两个分划的积，使划分更细；
   - 若R_1,R_2是A上的等价关系，则$R_1\cup R_2$是相容关系（自反、对称），不一定等价；
   - 若R_1,R_2是A上的等价关系，则$(R_1\cup R_2)^*$也是A上的等价关系，称为两个分划的和，使分划更粗；
   - 平凡分划：集合A的最粗或最细的分划
   - 等价关系多用关系图判断，用集合的划分研究；