复习1

2020年11月17日 13:56

数据仓库建模：数据立方体与OLAP

四个特征：

subject-oriented（面向主题的），integrated（集成的），time-variant（时变的），nonvolatile(非易失的)

OLTP(在线事务处理)：

▪传统关系型数据库管理系统的主要任务

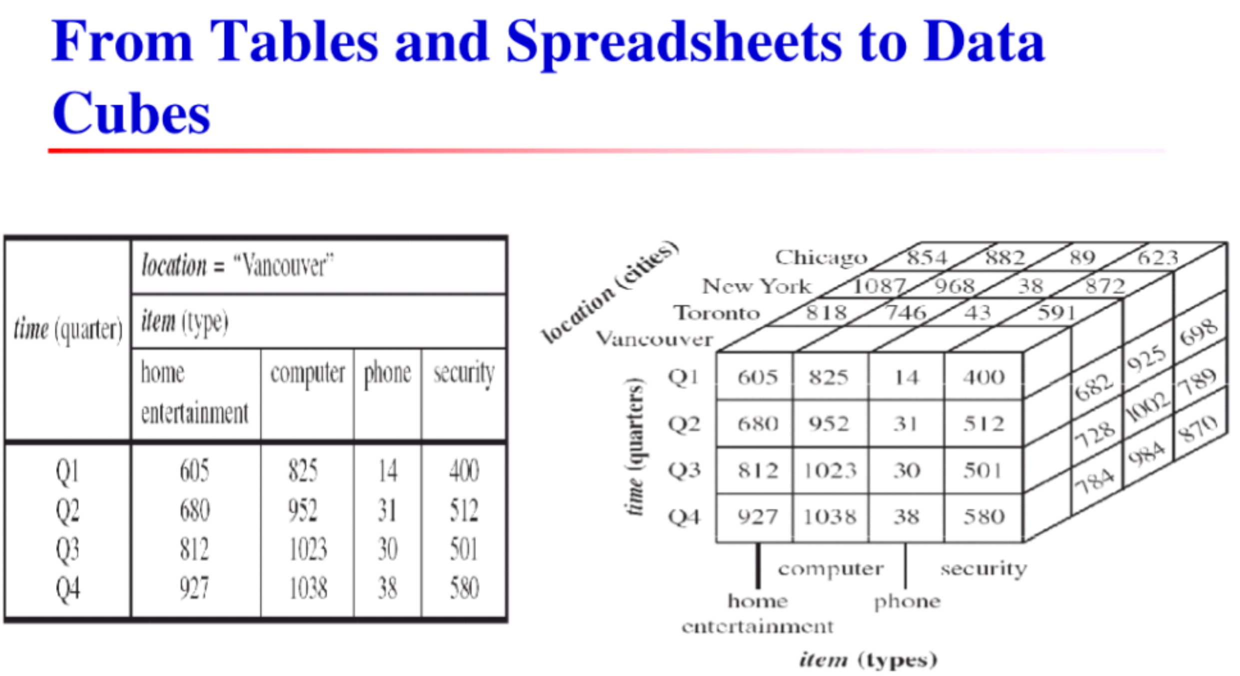
▪日常运营：例如采购，库存，银行业务，制造，工资单，注册，会计等

OLAP（在线分析处理）：

▪数据仓库系统的主要任务

▪数据分析和决策

多维数据仓库



time (quarter)	location = "Vancouver"			
	item (type)			
	home entertainment	computer	phone	security
Q1	605	825	14	400
Q2	680	952	31	512
Q3	812	1023	30	501
Q4	927	1038	38	580

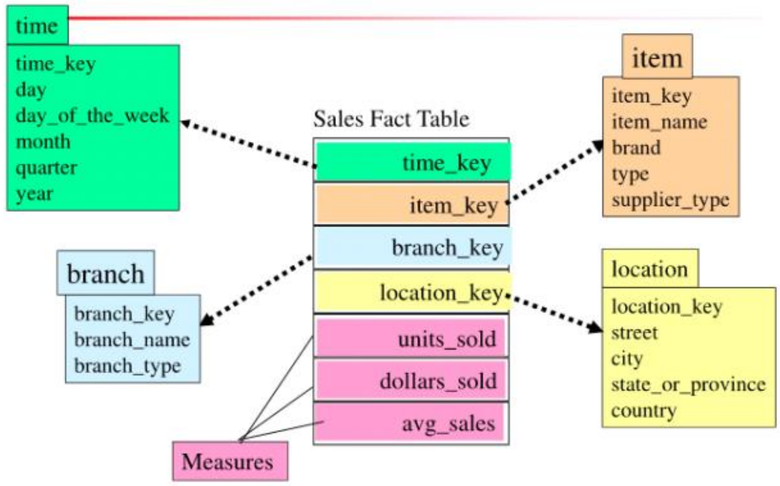
数据仓库建模:维度和度量

星形模式:中间的事实表连接到一组维度表

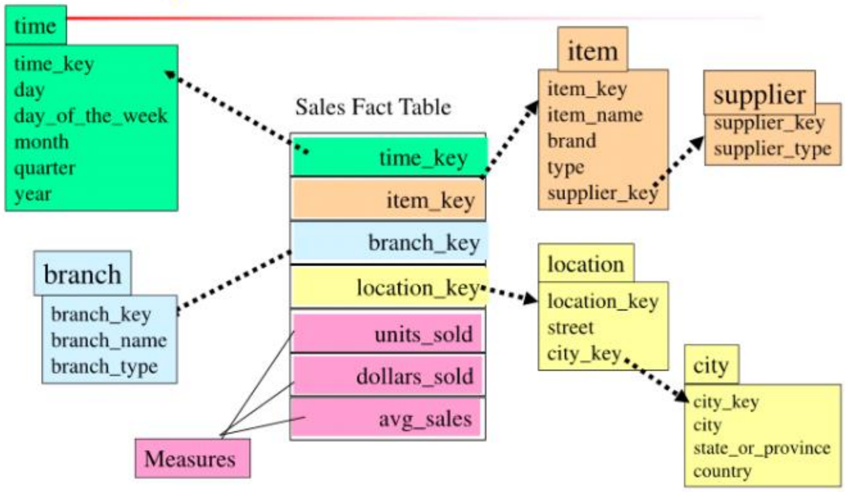
雪花模式:星形模式的改进，其中一些维度层次被规范化为一组较小的维度表，形成类似雪花的形状

事实星座:多个事实表共享维度表，被视为恒星的集合，因此称为星系模式或事实星座

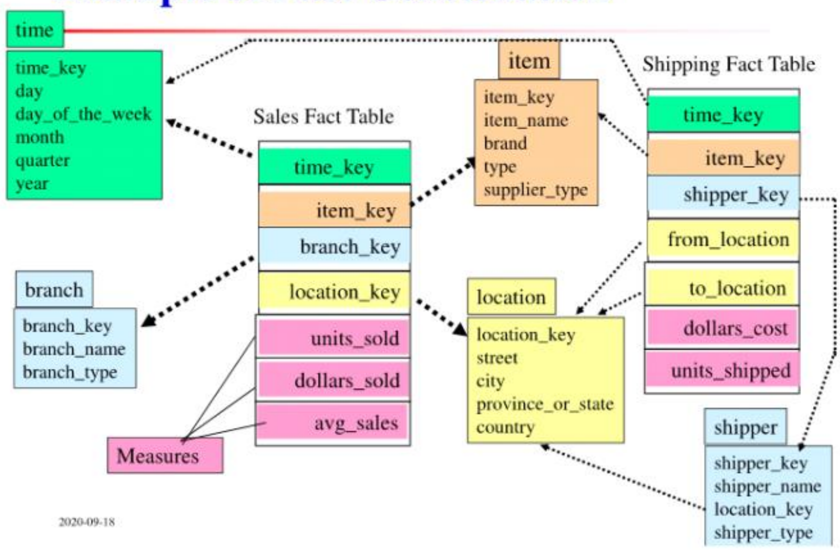
Example of Star Schema



Example of Snowflake Schema



Example of Fact Constellation



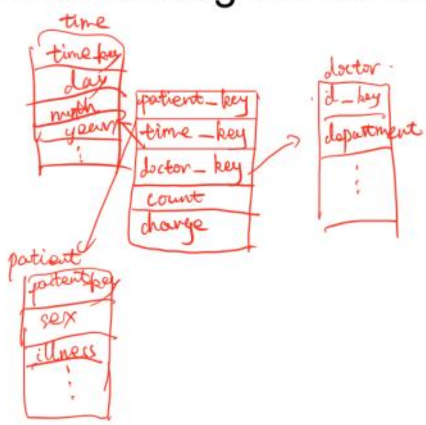
Exercise：

Exercise

1. Suppose that a data warehouse consists of three dimensions, time, doctor, and patient, and two measures count and charge, where charge is the fee that a doctor charges a patient for a visit.

sales fact table

(1) Draw a schema diagram for the data warehouse.



2020-09-18 28

DMQL：数据挖掘查询语言

typical OLAP operations

roll up(drill-up)：summarize data

▪通过爬升等级或缩小维度

Drill down(roll down)：reverse of roll-up

▪从较高级别的摘要到较低级别的摘要或详细数据，或引入新的维度

Slice and dice: project and select

slice在给定的立方体的一个维上进行选择，比如说time=“Q1”，则是选择第一季度

dice操作通过在两个或者多个维上进行选择，定义子立方体。比如说（location=“Toronto” or “Vancouver”） and (time="Q1") and (item=“computer”)

Pivot（rotate）：

reorient the cube, visualization, 3D to serires of 2D

是一种目视的操作，转动数据的视角，提供数据的替代表示。

drill-across（钻过）：

执行设计多个事实表的查询。

drill-through：

使用关系SQL机制，钻透到数据立方体的底层，到后端关系表。

Exercise：

Exercise

1. Suppose that a data warehouse consists of three dimensions *time, doctor, and patient*, and two measures count and charge, there charge is the fee that a doctor charges a patient for a visit.

(2) Starting with the base cuboid [day, doctor, patient], what OLAP operations should be performed in order to list the total fee collected by each doctor in 1999?

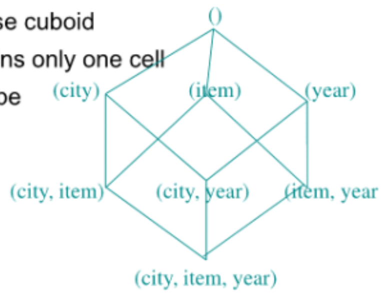
1. roll up from day to month to year
2. slice for year = "1999"
3. roll up on patient from individual patient to all
4. slice for patient = "all"
5. get the list of total fee collected by each doctor in 2004

2020-09-18 38

数据立方体：

Efficient Data Cube Computation

- Data cube can be viewed as a lattice of cuboids
  - The bottom-most cuboid is the base cuboid
  - The top-most cuboid (apex) contains only one cell
  - 2^n cuboids in an n-dimensional cube



- Materialization of data cube
  - Materialize *every* (cuboid) (full materialization), *none* (no materialization), or some (partial materialization)
  - Selection of which cuboids to materialize
    - Based on size, sharing, access frequency, etc.

索引OLAP数据：

Bitmap Index（位图索引）	Join Indices（连接索引）

Indexing OLAP Data: Bitmap Index

- Index on a particular column
- Each value in the column has a bit vector: bit-op is fast
- The length of the bit vector: # of records in the base table
- The *i*-th bit is set if the *i*-th row of the base table has the value for the indexed column
- Not suitable for high cardinality domains

Base Table

Cust	Region	Type
C1	Asia	Retail
C2	Europe	Dealer
C3	Asia	Dealer
C4	America	Retail
C5	Europe	Dealer

Index on Region

RecID	Asia	Europe	America
1	1	0	0
2	0	1	0
3	1	0	0
4	0	0	1
5	0	1	0

Index on Type

RecID	Retail	Dealer
1	1	0
2	0	1
3	0	1
4	1	0
5	0	1

例 4.8　连接索引。在例 4.1 中，我们定义了 AllElectronics 的一个星形模式，形如“*sales_star* [*time*，*item*，*branch*，*location*]：*dollars_sold* = **sum**（*sales_in_dollars*）”。事实表 *sales* 与维表 *location* 和 *item* 之间的连接索引联系显示在图 4.16 中。例如，维表 *location* 的值“*Main Street*”与事实表 *sales* 中的元组 T57、T238 和 T884 连接。类似地，维表 *item* 的值“*Sony-TV*”与事实表 *sales* 的元组 T57 和 T459 连接。对应的连接索引表显示在图 4.17 中。

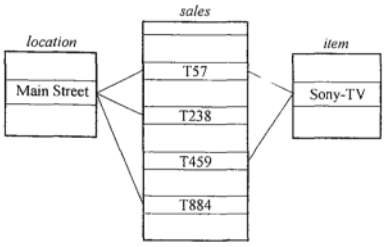


图 4.16　事实表 *sales* 与维表 *location* 和 *item* 之间的连接

location/sales
连接索引表

location	*sales_key*
...	...
Main Street	T57
Main Street	T238
Main Street	T884
...	...

item/sales
连接索引表

item	*sales_key*
...	...
Sony-TV	T57
Sony-TV	T459
...	...

location/item/sales
链接两个维的连接索引表

location	*item*	*sales_key*
...	...	...
Main Street	Sony-TV	T57
...	...	...

图 4.17　基于图 4.16 的事实表 *sales* 与维表 *location* 和 *item* 之间的连接的连接索引表

Exercise：

Exercise

1. Suppose a data warehouse for *Big_University* consists of four dimensions *student, course, semester, and instructor*, and two measures *count* and *score*.

(a) Draw a snowflake schema diagram for this data warehouse.

(b) Starting with the base cuboid [*student, course, semester, instructor*], what specific OLAP operations should you perform to list the number of CS courses for each *Big_University* student?

(c) If each dimension has five concept levels (including *all*), such as "*student* < *major* < *status* < *university* < *all*", how many cuboids will this cube contain?

(d) Taking this cube as an example, discuss advantages and problems of using a bitmap index structure.

Exercise

2. Suppose a data warehouse has 20 dimensions, each with five concept levels.

(a) Users are mainly interested in four particular dimensions, each having three frequently accessed levels for rolling up and drilling down. How would you design a data cube to efficiently support this preference?

(b) Occasionally, a user may want to drill through the cube down to its raw relational database for one or two particular dimensions. How would you support this feature?

复习2

2020年11月19日 17:39

数据预处理

箱线图（boxplots）

Q1：第25%的点

Q3：第75的点

IQR = Q3-Q1

Outlier:高于或者低于1.5倍的IQR的值

无偏样本方差s^2：需要为n-1分之1

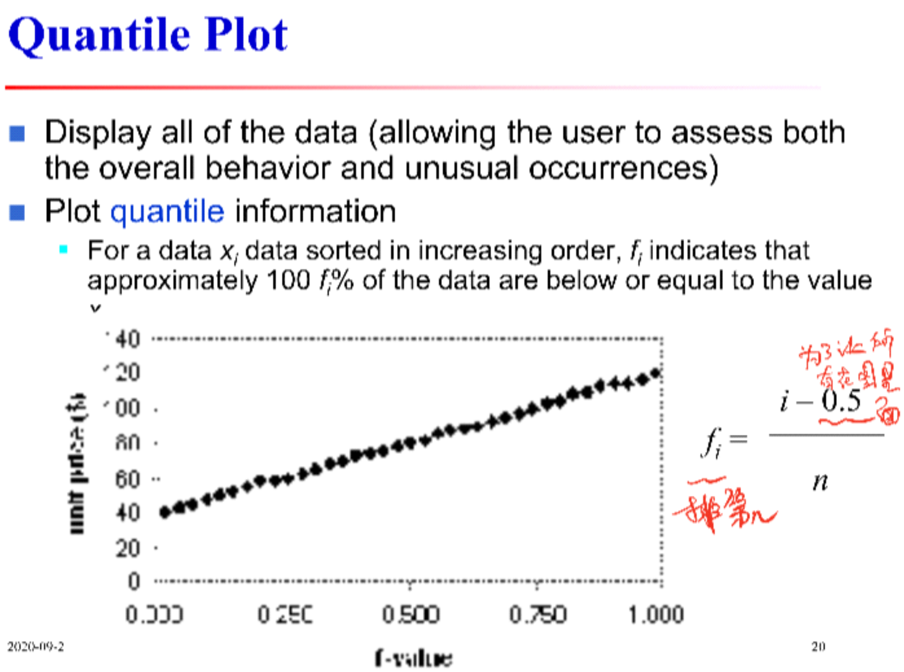
样本方差 sei ta的平方，为1/n

标准差

画箱型图的时候需要画出来离群点

Quantile Plot(分位数图)

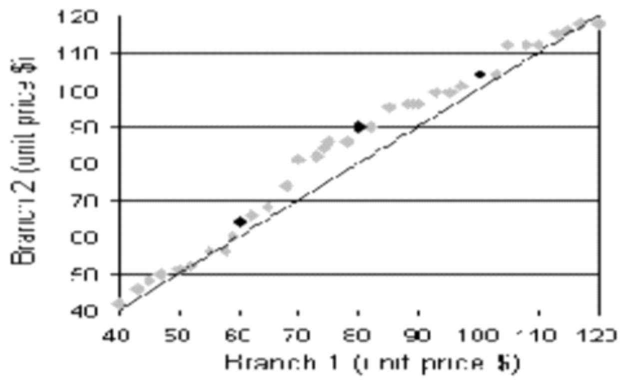
对于所有数据xi按照fi的值升序排列，fi = （i-0.5）/ n



Quantile-Quantile(Q-Q) Plot

Quantile-Quantile (Q-Q) Plot

- Graphs the quantiles of one univariate distribution against the corresponding quantiles of another
- Allows the user to view whether there is a shift in going from one distribution to another



2020-09-25 21

Scatter Plot（散点图）

每对值是为一堆坐标，并绘制为平面中的店

Loess Curve

在散点图的基础上添加平滑曲线

Exercise

Exercise

1. The values of data tuples are 13, 15, 16, 16, 19, 20, 20, 21.

(a) What is the mean of the data? What is the median? 17.5 17.5

(b) What is the mode of the data?

(c) What is Q1 and Q3?

(d) What is the IQR of the data?

(e) Give the five-number-summary of the data.

(f) Show a boxplot for the data.

(C) Q1=15
Q3 = 20
(D)IQR = 5

处理噪音数据：

分箱（bin）

等宽（距离）分区

将范围分为等大小的N个间隔

width = （Max-Min）/ N

等深度分区

将范围分为N个间隔，每个间隔大约包含相同数量的样本

Smothing by bin means: 用箱中的平均值代替该箱中的所有数据

Smoothing by bin boundaries：用距离较小的边界值代替箱中的每一个数据（看距离左边的边界近还是右边的边界近）

回归（Regression）

聚类（cluster）

Normalization（规范化）

- min-max normalization
- z-score normalization
- normaliz by decimal scaling（通过十进制缩放进行归一化）

min-max normalization

$$v' = \frac{v - min_a}{max_a - min_a}$$

z-score normalization（值-平均值除以标准差）

$$v' = \frac{v - u}{\sigma_A}$$

Normalization by decimal scaling

除10一直除到绝对值小于1

Exercise

相关性分析（数值数据Numercial Data）：

$$r_{A,B}=\frac{\sum(a_i-\bar{A})(b_i-\bar{B})}{(n-1)\sigma_A\sigma_B}=\frac{\sum(a_ib_i)-n\bar{A}\bar{B}}{(n-1)\sigma_A\sigma_B}$$

相关分析（分类数据Categorical Data）

$$\chi^2=\sum\frac{(Observed-Expected)^2}{Expected}$$

Exercise：

Excerise

1. The following contingence table summarizes supermarket transaction data.

(a) Based on the given data, is the purchase of hot dogs independent of the purchase of hamburgers?

(b) If correlated, what kind of correlation relationship exists between the two items?

	hot dogs	not hot dogs	sum
hamburgers	4000	3500	7500
not hamburgers	2000	500	2500
sum	6000	4000	10000

复习3

2020年11月21日 14:48

分类

监督学习（分类）：

训练数据有标签

无监督学习（聚类）：

训练数据无标签

评估分类的方法:

- 准确率（Accuracy）
- 速度（Speed）
  - 构建模型的时间
  - 使用模型的时间
- 健壮性（Robustness）
- 可伸缩性（Scalability）
- 可解释性（Interpretability）
  - 模型提供的理解和见解
- 其他措施，比如规则的有效性等

决策树（decision tree）

构建方法:

一开始所有训练样本都是根节点

属性是分类的（如果为连续值需要事先离散化）

根据所选属性对示例进行递归划分

Information Gain（ID3/C4.5）

Assume there are two classes, *P* and *N*

- Let the set of examples *S* contain *p* elements of class *P* and *n* elements of class *N*

$$\mathrm{I}(p,n) = -\frac{p}{p+n}log_2\frac{p}{p+n} - \frac{n}{p+n}log_2\frac{n}{p+n}$$

$$\mathrm{E}(A) = \sum_{i=1}^{v}\frac{p_i+n_i}{p+n}I(p_i,n_i)$$

$$\mathrm{Gain}(A) = I(p,n) - E(A)$$

Exercise:

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Exercise

1. Please calculate the information gain of *income*, *student*, and *credit_rating*, respectively.

- Gain(income) = 0.029
- Gain(Student) = 0.151
- Gain(credit_rating) = 0.048

Gain Ratio for attribute selection(C4.5)属性选择的增益比:

information gain的度量偏向属性

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times log_2\left(\frac{|D_j|}{|D|}\right)$$

$$\mathrm{GainRatio(A)} = Gain(A)/SplitInfo(A)$$

Gini Index(CART, IDM Intelligent Miner)

评估分类器的准确性

- 划分
  - 1/3的训练集
  - 2/3的测试集
- 交叉验证：K倍交叉验证
  - 将数据分为k部分
  - 在k-1的部分上训练，在1部分上测试
  - 重复k次
  - 平均准确率

贝叶斯分类器（Bayesian Classification）

数据集X， 假说得可能性H

$$P(H|\mathbf{X}) = \frac{P(\mathbf{X}|H)P(H)}{P(\mathbf{X})}$$

预测X属于C_i的概率

看哪个P（C_i|X）最大

Exercise：

Exercise

Predict what class does the data sample
X = (age <=30,
Income = medium,
Student = yes
Credit_rating = Fair) belong to?

Class:
C1:buys_computer = 'yes'
C2:buys_computer = 'no'

age	income	student	credit_rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Solution

- P(C_i): P(buys_computer = "yes") = 9/14 = 0.643
  P(buys_computer = "no") = 5/14= 0.357
- Compute P(X|C_i) for each class
  P(age = "<=30" | buys_computer = "yes") = 2/9 = 0.222
  P(age = "<= 30" | buys_computer = "no") = 3/5 = 0.6
  P(income = "medium" | buys_computer = "yes") = 4/9 = 0.444
  P(income = "medium" | buys_computer = "no") = 2/5 = 0.4
  P(student = "yes" | buys_computer = "yes) = 6/9 = 0.667
  P(student = "yes" | buys_computer = "no") = 1/5 = 0.2
  P(credit_rating = "fair" | buys_computer = "yes") = 6/9 = 0.667
  P(credit_rating = "fair" | buys_computer = "no") = 2/5 = 0.4
- **X = (age <= 30 , income = medium, student = yes, credit_rating = fair)**

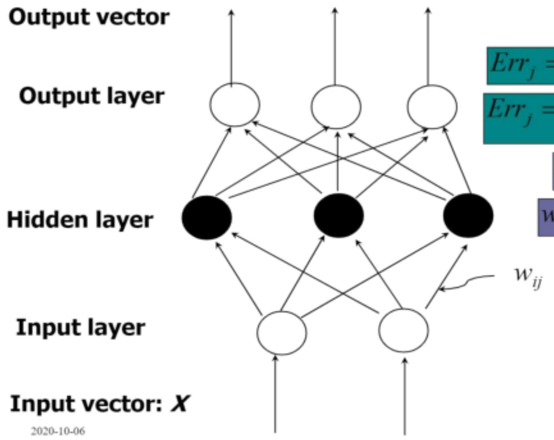
P(X|C_i) : P(X|buys_computer = "yes") = 0.222 x 0.444 x 0.667 x 0.667 = 0.044
P(X|buys_computer = "no") = 0.6 x 0.4 x 0.2 x 0.4 = 0.019
P(X|C_i)*P(C_i) : P(X|buys_computer = "yes") * P(buys_computer = "yes") = 0.028
P(X|buys_computer = "no") * P(buys_computer = "no") = 0.007
Therefore, X belongs to class ("buys_computer = yes")

Backporpagation（反向传播）

神经网络学习算法

在学习阶段，神经网络通过调整权重来进行学习，为了可以预测输入元组的正确标签

A Multi-Layer Feed-Forward Neural Network



$$Err_j = O_j(1-O_j)(T_j - O_j)$$

$$Err_j = O_j(1-O_j)\sum_k Err_k w_{jk}$$

$$\theta_j = \theta_j + (l)Err_j$$

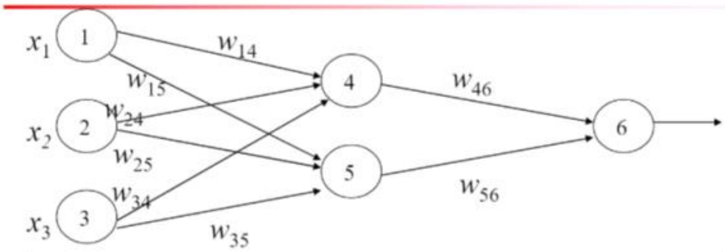
$$w_{ij} = w_{ij} + (l)Err_j O_i$$

$$O_j = \frac{1}{1+e^{-I_j}}$$

$$I_j = \sum_i w_{ij} O_i + \theta_j$$

Exercise：

Exercise



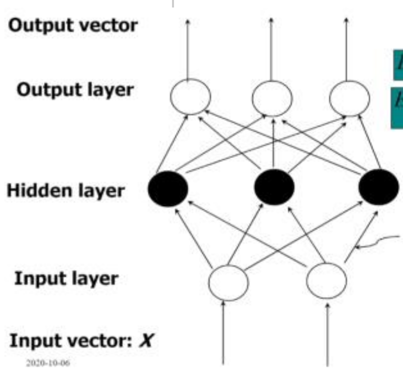
x_1	x_2	x_3	w_{14}	w_{15}	w_{24}	w_{25}	w_{34}	w_{35}	w_{46}	w_{56}	θ_4	θ_5	θ_6
1	0	1	0.2	−0.3	0.4	0.1	−0.5	0.2	−0.3	−0.2	−0.4	0.2	0.1

Unit j	Net input, I_j	Output, O_j
4	$0.2 + 0 - 0.5 - 0.4 = -0.7$	$1/(1+e^{0.7}) = 0.332$
5	$-0.3 + 0 + 0.2 + 0.2 = 0.1$	$1/(1+e^{-0.1}) = 0.525$
6	$(-0.3)(0.332) - (0.2)(0.525) + 0.1 = -0.105$	$1/(1+e^{0.105}) = 0.474$

Exercise

Unit j	Err j
6	$(0.474)(1-0.474)(1-0.474) = 0.1311$
5	$(0.525)(1-0.525)(0.1311)(-0.2) = -0.0065$
4	$(0.332)(1-0.332)(0.1311)(-0.3) = -0.0087$

Weight or bias	New value
w_{46}	$-0.3 + (0.9)(0.1311)(0.332) = -0.261$
w_{56}	$-0.2 + (0.9)(0.1311)(0.525) = -0.138$
w_{14}	$0.2 + (0.9)(-0.0087)(1) = 0.192$
w_{15}	$-0.3 + (0.9)(-0.0065)(1) = -0.306$
w_{24}	$0.4 + (0.9)(-0.0087)(0) = 0.4$
w_{25}	$0.1 + (0.9)(-0.0065)(0) = 0.1$
w_{34}	$-0.5 + (0.9)(-0.0087)(1) = -0.508$
w_{35}	$0.2 + (0.9)(-0.0065)(1) = 0.194$
θ_6	$0.1 + (0.9)(0.1311) = 0.218$
θ_5	$0.2 + (0.9)(-0.0065) = 0.194$
θ_4	$-0.4 + (0.9)(-0.0087) = -0.408$



$$Err_j = O_j(1-O_j)(T_j - O_j)$$

$$Err_j = O_j(1-O_j)\sum_k Err_k w_{jk}$$

$$\theta_j = \theta_j + (l)Err_j$$

$$w_{ij} = w_{ij} + (l)Err_j O_i$$

$$O_j = \frac{1}{1+e^{-I_j}}$$

$$I_j = \sum_i w_{ij} O_i + \theta_j$$

反向传播和可解释性

从网络中提取规则：network prunning

通过删除加权连接来简化网络结构

对受过训练的网络影响最小

研究一组输入值和激活值以得出规则

描述输入和隐藏单元之间得关系层数

k-nearest neighbor algorithm(k-邻近算法)

对于离散值，k-NN返回最接近Xq的K个训练示例中的最常见的值

Exercise：

Exercise

1. Consider the one-dimensional data set. Please classify the data point x=5.0 according to its 1-, 3-, and 5-nearest neighbors (using majority vote). positive 三个值分别为+，-，+

x	0.5	3.0	4.5	4.6	4.9	5.2	5.3	5.5	7.0	9.5
y	-	-	+	+	+	-	-	+	-	-

Popluar ensemble methods：

- Bagging
- Bossting

Bagging ： Boostrap Aggregation

直接上练习：

Exercise：

Exercise

1. Following is a data set to construct a bagging classifier.

x	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
y	1	1	1	-1	-1	-1	-1	1	1	1

Examples chosen for training in each round are shown below:

x	0.1	0.2	0.2	0.3	0.4	0.4	0.5	0.6	0.9	0.9
y	1	1	1	1	-1	-1	-1	-1	1	1

$0.35 < x \leq 0.75$=>y=-1

x	0.1	0.2	0.3	0.5	0.5	0.8	0.9	1	1	1
y	1	1	1	-1	-1	1	1	1	1	1

$0.4 < x \leq 0.65$=> y=-1

x	0.1	0.2	0.3	0.4	0.4	0.5	0.7	0.7	0.8	0.9
y	1	1	1	-1	-1	-1	-1	-1	1	1

$0.35 < x \leq 0.75$ =>y=-1

Please predict the class label for the record x=0.38.

2020-10-06 64

Boosting

类比:根据加权诊断的组合-分配的权重，根据先前的诊断准确性

Boosting可以扩展给连续值进行预测

与bagging算法相比，boosting算法倾向于实现更高的准确率，但是有过拟合的风险

预测Prediction

预测和分类相近

- 构建模型
- 使用模型对输入的值预测连续的或者有序的值
- 分类偏向于预测类别标签分类
- 预测模型连续值的函数

主要的预测方法：回归

回归分析：

- Linear and multiple regression线性和多元回归
- Non-linear regression 非线性回归

Linear Regression

$y=w_0+w_1 x$ 其中w_0为截距，w_1为斜率，这俩是回归系数

最小二乘法：估计best-fitting的直线

这里有个老大的公式了

多元线性回归：多个预测变量

Non-linear Regression非线性回归

for example：$y = w_0+w_1x+w_2x^2+w_3x^3$

Logistic Regression 逻辑回归

$$log\left(\frac{p}{1-p}\right) = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n,$$
$$p\ is\ probability\ , Y = 1$$

Confusion Matrix（混淆矩阵）

Predicted class

Actual class

	C_1	C_2	Total
C_1	True positive	False negative	pos
C_2	False positive	True negative	neg
Total	t-pos+f-pos	t-neg+f-neg	pos+neg

sensitivity = t-pos/pos /* true positive recognition rate */
specificity = t-neg/neg /* true negative recognition rate */
precision = t-pos/(t-pos + f-pos)

- Accuracy = (t-pos + t-neg)/ (pos + neg)
- Error rate (misclassification rate) of M = 1 – acc(M)

Exercise：

1. Please compute the sensitivity, specificity, precision and accuracy of the classifier.

classes	buy_computer = yes	buy_computer = no	total	recognition(%)
buy_computer = yes	6954	46	7000	99.34
buy_computer = no	412	2588	3000	86.27
total	7366	2634	10000	95.42

复习4

2020年11月22日 15:01

Cluster Analysis聚类分析

根据数据的特征并将相似的数据对象分组成簇

无监督学习Unsupervised learning

一些距离

minikowski距离

$$d(i,j)=\sqrt[q]{(|x_{i_1}-x_{j_1}|^q+|x_{i_2}-x_{j_2}|^q+\ldots+|x_{i_p}-x_{j_p}|^q)}$$

where $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two p-dimensional data objects, and q is a positive integer

Manhattan距离（此时q=1）

$$d(i,j)=|x_{i_1}-x_{j_1}|+|x_{i_2}-x_{j_2}|+\ldots+|x_{i_p}-x_{j_p}|$$

Euclidean distance（欧几里得距离，又称欧式距离，此时q=2）

$$d(i,j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+\ldots+|x_{i_p}-x_{j_p}|^2)}$$

Binary Variable相关的距离

Binary Variables

- A contingency table for binary data
- Distance measure for symmetric binary variables:
- Distance measure for asymmetric binary variables:

		Object j		
		1	0	*sum*
Object i	1	a	b	$a+b$
	0	c	d	$c+d$
	sum	$a+c$	$b+d$	p

$$d(i,j)=\frac{b+c}{a+b+c+d}$$

$$d(i,j)=\frac{b+c}{a+b+c}$$

Dissimilarity between Binary Variables

- Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

  - gender is a symmetric attribute
  - the remaining attributes are asymmetric binary
  - let the values Y and P be set to 1, and the value N be set to 0

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

2020-10-20 18

Nominal Variables

二进制变量的一般化，可能有更多的状态，比如说红，黄，蓝，绿

$\mathrm{d}(i,j) = \frac{p-m}{p}$ p：total # of nominal variables

Ordinal Variable

可以为离散也可以为连续的

顺序很重要，比如说：rank

- Can be treated like interval-scaled
  - replace x_{if} by their rank $r_{if} \in \{1, \ldots, M_f\}$
  - map the range of each variable onto [0, 1] by replacing *i*-th object in the *f*-th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

  - compute the dissimilarity using methods for interval-scaled variables

Ratio-Scaled Variables（比例缩放变量）

长得像Ae^{Bt}或者Ae^{-Bt}的

$y_{if} = \log(X_{if})$

然后再treat their rank as interval-scaled

Exercise：

Exercise

1. Please compute the dissimilarity matrix for the data set.

ID	Test-1 (categorical)	Test-2 (ordinal)	Test-3 (ratio-scaled)
1	A	excellent	445
2	B	fair	22
3	C	good	164
4	A	excellent	1,210

Solution

For test-1, use simple matching

0			
d(2,1)	0		
d(3,1)	d(3,2)	0	
d(4,1)	d(4,2)	d(4,3)	0

=

0			
1	0		
1	1	0	
0	1	1	0

For test-2

ID	Test-2 (ordinal)	Test-2 (ordinal)	Test-2 (ordinal)
1	excellent	3	1
2	fair	1	0
3	good	2	0.5
4	excellent	3	1

0			
1	0		
0.5	0.5	0	
0	1	0.5	0

For test-3, use log transformation

- Convert test-3 to 2.65, 1.34, 2.21, 3.08
- Normalize to 0.75, 0, 0.5,1

0			
0.75	0		
0.25	0.5	0	
0.25	1	0.5	0

Dissimilarity matrix

0			
d(2,1)	0		
d(3,1)	d(3,2)	0	
d(4,1)	d(4,2)	d(4,3)	0

=

0			
0.92	0		
0.58	0.67	0	
0.08	1	0.67	0

主要聚类方法

- Partitioning approach（分区，构建各种分区，然后通过一些准则评估）
    - k-means
    - k-medoids
    - CLARANS
- Hierarchical approach(分层，创建一组数据的层次分解hierarchical decomposition)
    - Diana

    - Agnes
    - BIRCH
    - ROCK
    - CHAMELEON
- Density-based approach（基于连通性和密度）
    - DBSACN
    - OPTICS
    - DenClue
- Grid-based approach（基于网格的方法，基于多层粒度结构）
    - STRING
    - WaveCluster
    - CLIQUE
- Probabilistic Model-based approach（基于概率模型的方法）
    - EM

三个概念

- centroid簇的中心点$C = \frac{\sum_{i=1}^{N} (t_i)}{N}$
- radius半径$R = \sqrt{\frac{\sum_{i=1}^{N} (t_i - c)^2}{N}}$
- diameter直径$D = \sqrt{\frac{\sum_{i=1}^{N} \sum_{j=1}^{N} (t_i - t_j)^2}{N(N-1)}}$

分区方法

k-means

•给定随机种子作为初始质心

•计算当前分区的每个簇的质心（质心是中心，即均值）

•对于每个对象，计算其与质心的距离

　　•将其分配给最近的质心

•返回步骤2，在没有更多新任务时停止

时间复杂度 线性阶O（tkn）、

k-medoids：

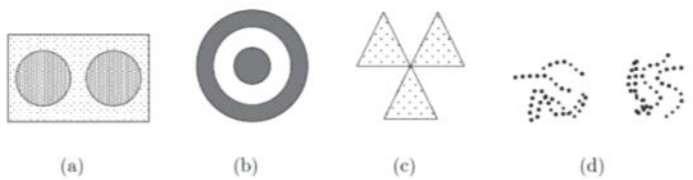
K-Medoids：不是使用聚类中对象的平均值作为参考点，而是使用medoidscan，它是聚类中位于中心的对象

时间复杂度O（n^2）

Exercise：

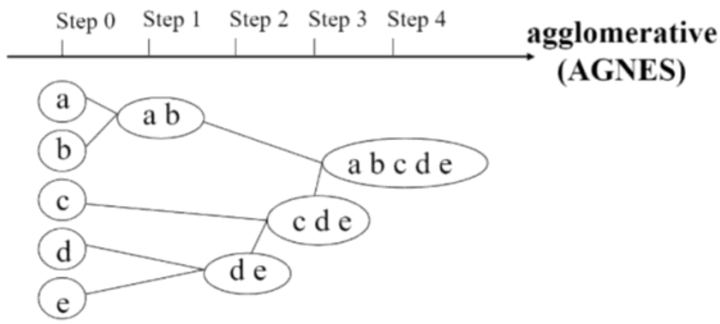
Exercise

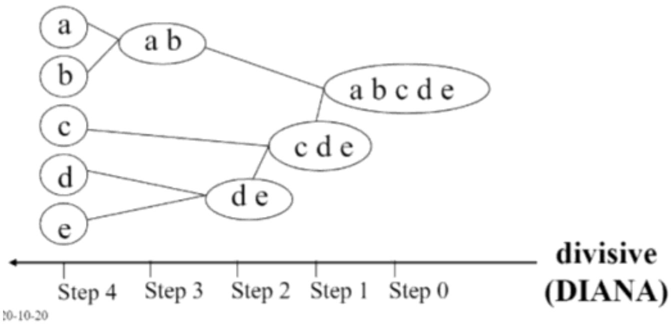
1. Identify the clusters using the K-means (using squared error as the objective function). Note that darkness or the number of dots indicates density.



Hierarchical Methods

不需要提前设定k 但是需要终止条件





AGENS（Agglomerative Nesting）

- 使用单链接方法和相异矩阵
- 合并差异最小的节点
- 以不降序的方式进行
- 最终所有节点都属于同一群集

如何定义差异最小（如何定义两个簇之间的距离）
最近的两个簇的点之间的距离

DIANA(Divisive Analysis)

- AGNES的逆顺序
- 最终每个节点自己形成一个集群

BIRCH

集成的分层聚类

Clustering feature，Clustering feature tree

逐步构造一个CF（Clustering feature） tree

阶段一，扫描数据库以构建初始的内存CF tree

阶段二，使用聚类算法来聚类CF tree的叶节点

Cluster Feature：CF=（N，LS，SS）

N：节点数

LS：每一个维的线性和

SS:每一个维的平方和

比如(3,4), (2,6), (4,5), (4,7), (3,8)的CF=（5，（16，30），（54，190））

复习5

2020年11月23日 14:59

ARM

支持度support（a=>b）=P(a∩b)

置信度confidence（a=>b） = P(b|a) = $\frac{count\ (a\cap b)}{count\ (a)}$ = $\frac{P\ (a\cap b)}{P\ (a)}$

满足最小置信度和最小支持度得即为强规则（strong rule）

兴趣度量：correlation相关性（lift）

$$\text{lift} = \frac{P(A\cap B)}{P(A)P(B)}$$

称为A条件对于B事件的提升度,如果该值=1,说明两个条件没有任何关联,如果<1,说明A条件(或者说A事件的发生)与B事件是相斥的, 一般在数据挖掘中当提升度大于3时,我们才承认挖掘出的关联规则是有价值的。

挖掘一维布尔关联规则

Apriori

指导原则：每一个频繁项集的子集均为频繁集

步骤：

i. 遍历数据库找出所有的1频繁项集
ii. 从k项集生成k+1的候选频繁项集
iii. 检测这些候选频繁项集通过数据库
iv. 在没有频繁集或者候选频繁集的时候算法终止

pseudo-code

```
L1={frequent single items from D}
for (k=2, Lk-1!=∅;k++)do begin
    Ck = candidates generated from Lk-1
    for each transcation t∈D do
        increment the count of all candidates in Ck which are
        contained in t
    end
    Lk = candidates in Ck with min_support
end
return L=∪kLk
```

Exercise：

1. A database has 9 transactions. Let *min_sup* = 20%. Please present all the candidates and frequent itemsets at each iteration.

I1 6	I1 I2	I1 I2 I3
I2 7	I1 I3	I1 I2 I5
I3 6	~~1 4~~	
I4 2	1 5	
I5 2	2 3	
	2 4	
	2 5	
	~~3 4~~	
	~~3 5~~	
	~~4 5~~	

TID	List of items_IDs
T100	I1,I2,I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3

2020-10-27 25

Partition：扫描数据库仅两次

partition technique

将数据划分为N个小分区

第一阶段：在每个分区上找到局部的频繁项集并记录。

第二阶段：整合所有的局部频繁项集，扫描数据库，找到全局范围的频繁项集

定理：在数据库中的任一可能频繁项集，在划分中的局部中必定要频繁的。（如果在局部都不频繁，在全局就更不可能频繁了）

执行时间呈线性比例

DHP：减少候选项集的数量

看不懂 啥玩意啊

原理：一个k项集，其对应的哈希值储存桶数低于阈值则不能频繁

DIC：较少扫描数量

将数据库划分成标记着开始点数的块

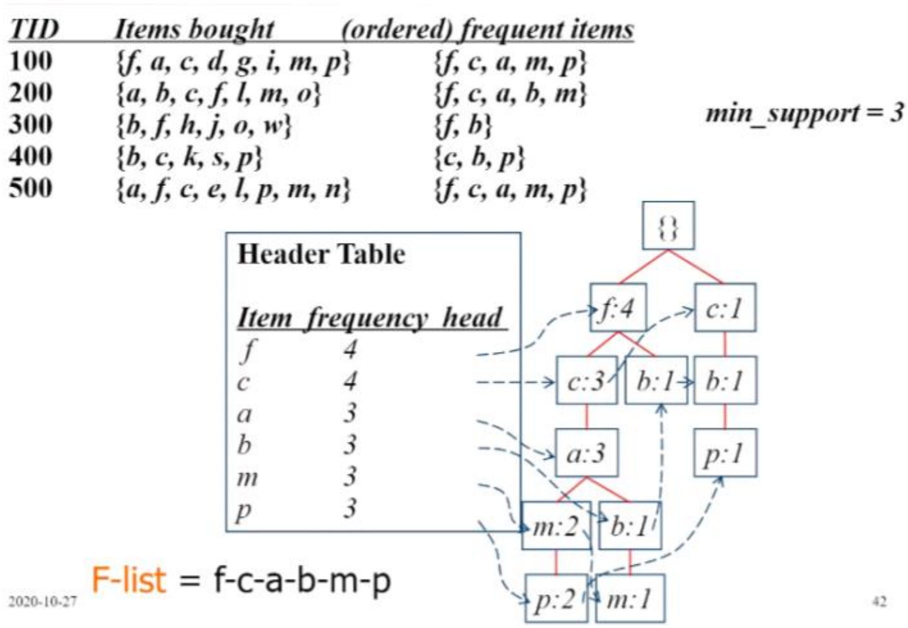
新的候选集可以被添加任意开始点数，如果他的所有子集都是频繁的

减少数据库扫描的次数

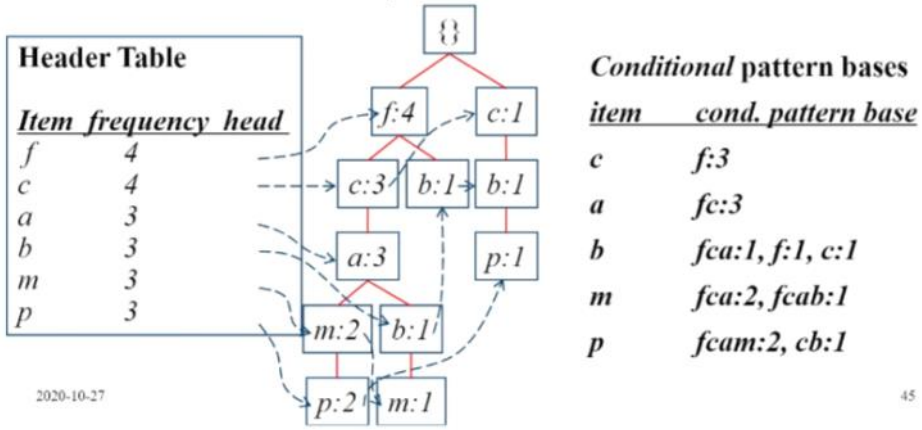
从事务数据库构建FP-tree

先找单个的频繁项集 构建头表

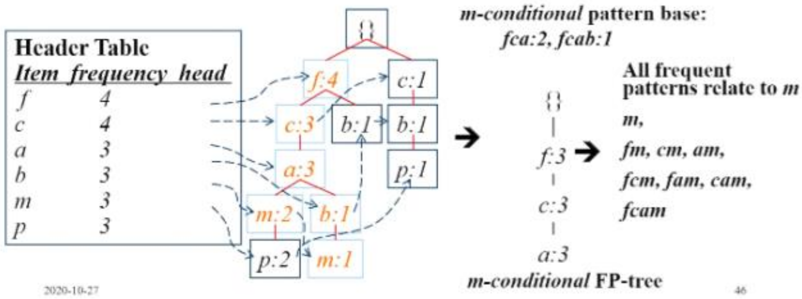
然后将每个频繁项集按照单个频繁项集进行重新排列



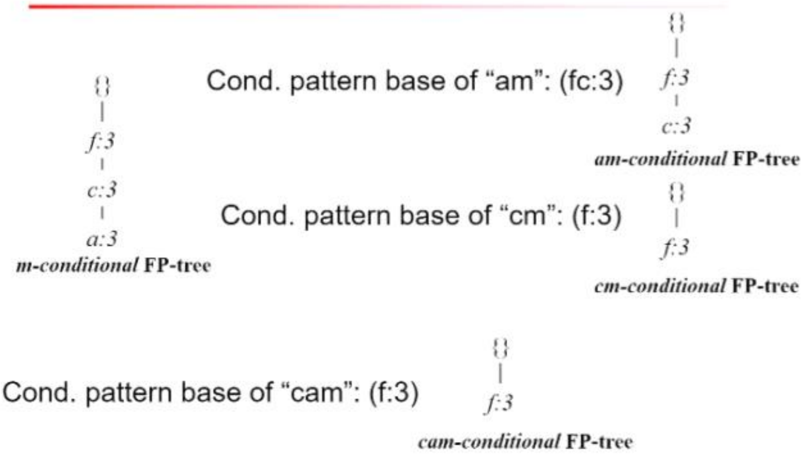
构建条件模式基（Coniditional Pattern Base）



从条件模式基到条件FP树



Recursion: Conditional FP-tree



Exercise：

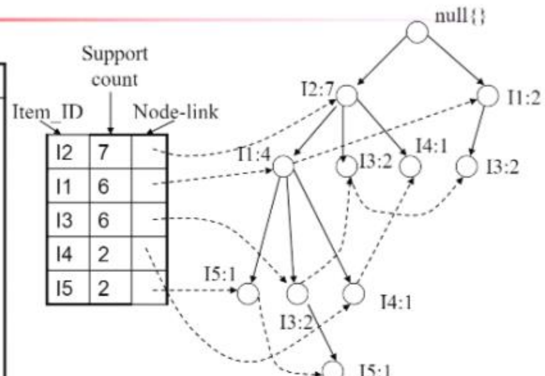
3. A database has 9 transactions. Let *min_sup* = 20%. Please construct the FP-tree for the database, the conditional FP-trees, and all the frequent itemsets.

TID	List of items_IDs
T100	I1,I2,I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3

均为单频

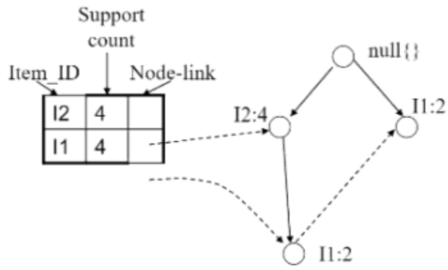
2020-10-27

TID	List of items_IDs
T100	I1,I2,I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3



Support =2

item	*conditional pattern base*	*conditional FP-tree*	*frequent patterns generated*
I5	{{I2,I1: 1}, {I2,I1,I3: 1}}	⟨I2: 2, I1: 2⟩	{I2,I5: 2}, {I1,I5: 2}, {I2,I1,I5: 2}
I4	{{I2,I1: 1}, {I2: 1}}	⟨I2: 2⟩	{I2,I4: 2}
I3	{{I2,I1: 2}, {I2: 2}, {I1: 2}}	⟨I2: 4, I1: 2⟩, ⟨I1: 2⟩	{I2,I3: 4}, {I1,I3: 4}, {I2,I1,I3: 2}
I1	{{I2: 4}}	⟨I2: 4⟩	{I2,I1: 4}



挖掘多层关键规则（Mining multilevel association rules）

对于所有层使用一致最小的支持度（称为一致支持度）

在较低层使用递减的最小支持度（称为递减支持度）

挖掘多维关联规则（Mining multidimensional association rules）

我们把规则中每个不同的谓词称为维，因此我们称规则为单维，或者维内关联规则。

将设计到两个或多个谓词的关键规则称作多维关联规则。

e.g. age(X, "20…29")∩occupation(X, "student") =>buys(x, "laptop")

两种方法:

i. 使用预先定义的概念分层对量化属性离散化。
ii. 根据数据分布将量化属性离散化或聚类到“箱子”（动态量化关联规则）

复习6

2020年11月23日　　20:07

Recommend Algorithm

Content-based Methods

该用户的兴趣应该匹配他应该被推荐的物品描述

core idea：寻找用户之间和所有现存物品之间的相似性

步骤:

- 使用一组k个关键词对用户的画像和物品进行矢量化描述
- 矢量化用户和物品并且计算相似性

$$I_j = (i_{j,1}, i_{j,2}, \ldots, i_{j,k}) \qquad U_i = (u_{i,1}, u_{i,2}, \ldots, u_{i,k}).$$

$$sim(U_i, I_j) = cos(U_i, I_j) = \frac{\sum_{l=1}^{k} u_{i,l} i_{j,l}}{\sqrt{\sum_{l=1}^{k} u_{i,l}^2} \sqrt{\sum_{l=1}^{k} i_{j,l}^2}}$$

- 将最相似的项目推荐给用户

协同过滤：Collaborative Filtering

Collaborative Filtering

- Assumption
  - User-based CF
    - Users with similar previous ratings for items are likely to rate future items similarly

	I1	I2	I3	I4
U1	1	2	4	4
U2	1	2	4	?
U3	2	5	2	2
U4	5	2	3	3

  - Item-based CF
    - Items that have received similar ratings previously from users are likely to receive similar ratings from future users (item-based CF)

	I1	I2	I3	I4
U1	1	2	4	4
U2	1	2	4	?
U3	2	5	2	2
U4	5	2	3	3

协同过滤算法：

Collaborative Filtering Algorithm

- Measure Similarity between Users (or Items)

$$sim(U_i, U_j) = cos(U_i, U_j) = \frac{U_i \cdot U_j}{||U_i||\ ||U_j||} = \frac{\sum_k r_{i,k} r_{j,k}}{\sqrt{\sum_k r_{i,k}^2}\sqrt{\sum_k r_{j,k}^2}}$$

- Pearson Correlation Coefficient

$$sim(U_i, U_j) = \frac{\sum_k (r_{i,k} - \bar{r}_i)(r_{j,k} - \bar{r}_j)}{\sqrt{\sum_k (r_{i,k} - \bar{r}_i)^2}\sqrt{\sum (r_{j,k} - \bar{r}_j)^2}}$$

Updating the ratings:

User v's mean rating

User u's mean rating

$$r_{u,i} = \bar{r}_u + \frac{\sum_{v \in N(u)} sim(u,v)(r_{v,i} - \bar{r}_v)}{\sum_{v \in N(u)} sim(u,v)},$$

Predicted rating of user u for item i

Observed rating of user v for item i

Example

	Lion King	Aladdin	Mulan	Anastasia
John	3	0	3	3
Joe	5	4	0	2
Jill	1	2	4	2
Jane	3	?	1	0
Jorge	2	2	0	1

Predict Jane's rating for Aladdin

1- Calculate average ratings

$$\bar{r}_{John} = \frac{3+3+0+3}{4} = 2.25$$

$$\bar{r}_{Joe} = \frac{5+4+0+2}{4} = 2.75$$

$$\bar{r}_{Jill} = \frac{1+2+4+2}{4} = 2.25$$

$$\bar{r}_{Jane} = \frac{3+1+0}{3} = 1.33$$

$$\bar{r}_{Jorge} = \frac{2+2+0+1}{4} = 1.25$$

2- Calculate user-user similarity

$$sim(Jane, John) = \frac{3\times 3+1\times 3+0\times 3}{\sqrt{10}\sqrt{27}} = 0.73$$

$$sim(Jane, Joe) = \frac{3\times 5+1\times 0+0\times 2}{\sqrt{10}\sqrt{29}} = 0.88$$

$$sim(Jane, Jill) = \frac{3\times 1+1\times 4+0\times 2}{\sqrt{10}\sqrt{21}} = 0.48$$

$$sim(Jane, Jorge) = \frac{3\times 2+1\times 0+0\times 1}{\sqrt{10}\sqrt{5}} = 0.84$$

User_based CF, Example

3- Calculate Jane's rating for Aladdin,
Assume that neighborhood size = 2

$$\begin{aligned} r_{Jane,Aladdin} &= \bar{r}_{Jane} + \frac{sim(Jane,Joe)(r_{Joe,Aladdin} - \bar{r}_{Joe})}{sim(Jane,Joe) + sim(Jane,Jorge)} \\ &+ \frac{sim(Jane,Jorge)(r_{Jorge,Aladdin} - \bar{r}_{Jorge})}{sim(Jane,Joe) + sim(Jane,Jorge)} \\ &= 1.33 + \frac{0.88(4-2.75) + 0.84(2-1.25)}{0.88+0.84} = 2.33 \end{aligned}$$

User_based CF, Example

3- Calculate Jane's rating for Aladdin,
Assume that neighborhood size = 2

$$r_{Jane,Aladdin} = \bar{r}_{Jane} + \frac{sim(Jane, Joe)(r_{Joe,Aladdin} - \bar{r}_{Joe})}{sim(Jane, Joe) + sim(Jane, Jorge)} + \frac{sim(Jane, Jorge)(r_{Jorge,Aladdin} - \bar{r}_{Jorge})}{sim(Jane, Joe) + sim(Jane, Jorge)}$$

$$= 1.33 + \frac{0.88(4 - 2.75) + 0.84(2 - 1.25)}{0.88 + 0.84} = 2.33$$

User_based CF, Example

3- Calculate Jane's rating for Aladdin,
Assume that neighborhood size = 2

$$r_{Jane,Aladdin} = \bar{r}_{Jane} + \frac{sim(Jane, Joe)(r_{Joe,Aladdin} - \bar{r}_{Joe})}{sim(Jane, Joe) + sim(Jane, Jorge)} + \frac{sim(Jane, Jorge)(r_{Jorge,Aladdin} - \bar{r}_{Jorge})}{sim(Jane, Joe) + sim(Jane, Jorge)}$$

$$= 1.33 + \frac{0.88(4 - 2.75) + 0.84(2 - 1.25)}{0.88 + 0.84} = 2.33$$

Predictive Accuracy Metrics（预测精度指标）

Mean Absolute Error（MAE）平均绝对误差

$$MAE = \frac{\sum_{ij} |\hat{r}_{ij} - r_{ij}|}{n}$$

$$\text{NMAE} = \frac{MAE}{r_{max} - r_{min}}$$

Root Mean Square Error（均方根误差）

$$RMSE = \sqrt{\frac{1}{n}\sum_{i,j}(\hat{r}_{ij} - r_{ij})^2}$$

Example

Consider the following table with both the predicted ratings and true ratings of five items

Item	*Predicted Rating*	*True Rating*
1	1	3
2	2	5
3	3	3
4	4	2
5	4	1

$$
\begin{aligned}
MAE &= \frac{|1-3|+|2-5|+|3-3|+|4-2|+|4-1|}{5} = 2 \\
NMAE &= \frac{MAE}{5-1} = 0.5 \\
RMSE &= \sqrt{\frac{(1-3)^2+(2-5)^2+(3-3)^2+(4-2)^2+(4-1)^2}{5}} \\
&= 2.28
\end{aligned}
$$