Data Mining: Concepts and Techniques, Third Edition

Instructor Support

Sample Exam and Homework Questions

Jiawei Han, Micheline Kamber, Jian Pei

The University of Illinois at Urbana-Champaign

Simon Fraser University

Version September 25, 2011

For Instructors' references only.

Do not copy! Do not distribute!

Preface

To help instructors to teach their courses, we present a set of exam and homework questions used in the last several years in our teaching of data mining courses. These include the following set of materials:

1. Midterm and final exam questions for the course "CS 412: An Introduction to Data Mining", offered in the Department of Computer Science, The University of Illinois at Urbana-Champaign

2. Midterm exam questions for the course "CS 512: Data Mining: Principles and Algorithms", offered in the Department of Computer Science, The University of Illinois at Urbana-Champaign

3. Ph.D. Qualifying exam questions for the Data Mining Section, tested in the field of "Data and Information Systems (DAIS)", in the Department of Computer Science, The University of Illinois at Urbana-Champaign

4. Homework questions in the course "CS 412: An Introduction to Data Mining", offered in the Department of Computer Science, The University of Illinois at Urbana-Champaign

5. Homework questions in the course "CS 512: Data Mining: Principles and Algorithms", offered in the Department of Computer Science, The University of Illinois at Urbana-Champaign

We also presented some of the "standard" answers to the midterm and final exam questions when available. For the answers to other questions, instructors may work out independently.

You are welcome to enrich this manual by suggesting additional interesting exercises and/or providing more thorough, or better alternative solutions.

While we have done our best to ensure the correctness of the solutions, it is possible that some typos or errors may exist. If you should notice any, please feel free to point them out by sending your suggestions to *hanj@cs.uiuc.edu*. We appreciate your suggestions.

Notes to the current release of the solution manual.

Due to the limited time, this release of the instructor support is a preliminary version. We have not included the set of homework questions yet. The homework questions in the current year and the last year can be founded from our course web-sites. We apologize for the inconvenience. We will incrementally add answers to those questions in the next several months and release the new versions of updated solution manual in the subsequent months.

Acknowledgements

The solutions to some of the questions were worked out by our teach assistants and students. We sincerely express our thanks to all the teaching assistants and participating students who have worked with us to make and improve the solutions to the questions. In particular, we would like to thank Lu An Tang, Xiao Yu, and Peixiang Zhao for their help at working out some of the homework and/or exam questions and their answers.

Contents

Chapter 1

Sample Exam Questions for Course I

Enclosed are some sample midterm and final exam questions of the introductory level data mining course offered at Computer Science, UIUC: "UIUC CS 412 Introduction to Data Mining". Most midterm exams had 90 minutes of time, close book, but allowing student to bring one sheet of paper (notes) worked out by students themselves. Most final exams had 180 minutes of time, close book, but allowing student to bring two sheets of paper (notes) worked out by students themselves.

1.1 Sample Exam Question Set: 1.1

1.1.1 Midterm Exam

1. [30] Data preprocessing.

   (a) [9] Suppose a group of 12 students with the test scores listed as follows:

   $$19, 71, 48, 63, 35, 85, 69, 81, 72, 88, 99, 95.$$

   Partition them into four bins by (1) equal-frequency (equi-depth) method, (2) equal-width method, and (3) an even better method (such as clustering).

   Answer:

   - Equal-frequency: 19, 35, 48 || 63, 69, 71 || 72, 81, 88 || 85, 95, 99
   - Equal-width: Since [(99-19)+1]/4 = 20.25, the four slots should be 19-38.25, 38.26-58.5, 58.51-78.75, 78.76-99. Thus we have
     19, 35 || 48 || 63, 69, 71, 72 || 81, 88 , 85, 95, 99

- Clustering: There could be more than one answer. Such as
  19, 35, 48 || 63, 69, 71, 72 || 81, 88 , 85 || 95, 99
  or
  19 || 35, 48 || 63, 69, 71, 72 || 81, 88, 85, 95, 99

□

(b) [9] What are the value ranges of the following normalization methods, respectively? (1) min-max normalization, (2) z-score normalization, and (3) normalization by decimal scaling?

Answer:

- min-max normalization: [min_val, max_val]
- z-score normalization: The range is $[(old\_min - mean)/stddev, (old\_max - mean)/stddev]$. In general the range for all possible data sets is $(-\infty, +\infty)$.
- Normalization by decimal scaling: $(-1, +1)$

□

(c) [12] Table 1.1 shows how many transactions containing beer and/or nuts among 10000 transactions. (1) (roughly) calculate χ^2, (2) calculate *lift*, (3) calculate *all-confidence*, and (4) based on your calculation, how do you conclude the relationship between buying_beer and buying_nuts?

	Beer	No Beer	total
Nuts	50	800	850
No Nuts	150	9000	9150
Total	200	9800	10000

Table 1.1: Statistics of transactions related to Buying Beer and Buying Nuts

Hint: The formulae to compute χ^2, lift and all-confidence are as follows.

$$\chi^2 = \Sigma \frac{(observed - expected)^2}{expected} \tag{1.1}$$

$$lift(A, B) = \frac{P(A \cup B)}{P(A)P(B)} \tag{1.2}$$

$$all\_conf(X) = \frac{sup(X)}{max\_item\_sup(X)} = \frac{sup(X)}{max\{sup(i_j)|\forall i_j \in X\}} \tag{1.3}$$

where $max\{sup(i_j)|\forall i_j \in X\}$ is the maximum (single) item support of all the items in X.

Answer:

- χ^2
  $e_{BN} = (200 \times 850)/10000 = 17$
  $e_{\neg B,N} = (9800 \times 850)/10000 = 833$
  $e_{B,\neg N} = (200 \times 9150)/10000 = 183$
  $e_{\neg B,\neg N} = (9800 \times 9150)/10000 = 8967$

$$\begin{aligned}\chi^2 &= (50-17)^2/17 + (800-833)^2/833 + (150-183)^2/183 + (8967-9000)^2/8967 \\ &= 64.06 + 1.31 + 5.95 + 0.12 = 71.44 \gg 0\end{aligned}$$

- $lift(B,N)$

$$lift(B,N) = \frac{P(B,N)}{P(B) \times P(N)} = \frac{50/10000}{200/10000 \times 850/10000} = 2.94 > 1$$

- $all\_conf(B \cup N)$

$$all\_conf(B \cup N) = \frac{sup(B \cup N)}{max(sup(B), sup(N))} = \frac{50/10000}{850/10000} = 0.059 \ll 0.5$$

- Conclusion: B and N are strongly negatively correlated based on $all\_conf(B \cup N)$. $lift(B,N)$ is not a good indicator in this case since there are a large number of null transactions. Similarly, χ^2 analysis is not reliable in this situation because it claims positively correlated but it is actually strongly negatively correlated.

□

2. [16] Data Warehousing and OLAP for Data Mining

Suppose a market shopping data warehouse consists of four dimensions: *customer, date, product*, and *store*, and two measures: *count*, and *avg_sales*, where *avg_sales* stores the real sales in dollar at the lowest level but the corresponding average sales at other levels.

(a) [4] Draw a **snowflake schema** diagram (sketch it, do not have to mark every possible level, and make your implicit assumptions on the levels of a dimension when you draw it).

Answer: Graph can be drawn easily (thus omitted). □

(b) [8] Starting with the base cuboid [*customer*, *date*, *product*, *store*], what **specific OLAP operations** (e.g., roll-up student to department (level)) that one should perform in order to list the average sales of each *cosmetic* product *since January 2005*?

Answer:

- roll-up on customer to *all*

- roll-up on store to *all*
- roll-up on product to product-type
- select on product with product-type = "cosmetic"
- roll-up on date to year
- select on date with year = "2005"
- drill-down on date to month (if you would like to see it by month)
- drill-down on product to product-name (since we need to get each cosmetic product)
- the results are in the measure "avg_sales"

□

(c) [4] If each dimension has 5 levels (excluding *all*), such as *store-city-state-region-country*, *how many cuboids* does this cube contain (including base and apex cuboids)?

Answer: $\Pi_D(L_i + 1) = 6 \times 6 \times 6 \times 6 = 6^4 = 1296$

□

3. [25] Data cube and data generalization

(a) [8] Assume a base cuboid of N dimensions contains only p (where $p > 3$) nonempty base cells, and there is no level associated with any dimension. (1) What is the *maximum number of nonempty cells* (including the cell in the base cuboid) possible in such a materialized datacube? and (2) if the minimum support (i.e., iceberg condition) is 3, what is the *minimum number of nonempty cells* possible in the materialized iceberg cube?

Answer:

- *maximum number of nonempty cells* (including the cell in the base cuboid) possible in such a materialized datacube: $p \times 2^N - p + 1$
  Reasoning:
  Each cell will generate 2^N nonempty cells (including the cell in the base cuboid).
  p cells will generate $p \times 2^N$ such nonempty cells
  The case of maximum is the case they do not overlap (merge) in all the dimensions except in the case of the apex cell, which merges p cells into one, thus it will have $-p + 1$ cells.
- *minimum number of nonempty cells* possible in the materialized iceberg cube: 1
  Reasoning:
  If we do not merge cells, these cells will be eliminated by iceberg condition. As shown the case above, only one cell, the apex cell, cannot be eliminated since its count is $p > 3$.

□

(b) [11] Among the following four methods: *multiway array computation* (Zhao, et al. SIGMOD'1997), *BUC* (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD'2001), StarCubing (Xin et al., VLDB'2003), and *shell-fragment* approach (Li et al, VLDB'2004), which one is the best choice in each of the following cases?

(a) computing a dense full cube of low dimensionality (e.g., less than 8 dimensions),

(b) computing a large iceberg cube of around 10 dimensions, and

(c) computing a sparse iceberg cube of high dimensionality (e.g., over 100 dimensions).

Answer:

(a) Multiway array computation

(b) Either BUC or StarCubing

(c) Shell-fragment

□

(c) [6] Suppose a disk-based large relation contains 100 attributes. What is the minimum number of database scans in order to derive a generalized relation by *attribute-oriented induction*?

Answer:

- $1 + \delta$: The first or δ scan (by taking samples) to determine how high each attribute will need to generalize. The second scan is to generalize the relation into a generalized one.

□

4. [26] Frequent pattern and association mining.

Figure 1.1: FP tree of a transaction DB

(a) [8] A database with 100 transactions has its FP-tree shown in Fig. 1.4. Let $min\_sup = 0.4$ and $min\_conf = 0.6$. Show (1) c's conditional database, (2) the frequent k-itemset for the largest k, and (3) all the strong association rules (with support and confidence) containing the k items (for the largest k only).

Answer:

(1) c's conditional database: c:{eab: 20, ea: 15, eb: 10}

(2) the frequent k-itemset for the largest k:
$k = 3$ where $abe : 60$,

(3) all the strong association rules (with support and confidence) containing the k items:
The abe-related supports of the itemsets are $a : 78, b : 75, e : 95, ab : 60, ae : 78, be : 75$. Thus we have rules (with support $s = 0.6$ for all the rules),

i. $a \rightarrow be \quad (conf = 60/78)$
ii. $b \rightarrow ae \quad (conf = 60/75)$
iii. $e \rightarrow ab \quad (conf = 60/95)$
iv. $ab \rightarrow e \quad (conf = 60/60)$
v. $ae \rightarrow b \quad (conf = 60/78)$
vi. $be \rightarrow a \quad (conf = 60/75)$

□

(b) [6] Briefly describe one efficient **incremental** frequent-itemset mining method which can incorporate newly added transactions without redoing the mining from scratch.

Answer:

(1) Input: DB (the original transaction database), $freq_{DB}$ (frequent itemset of DB), δDB (the newly added transactions).

(2) Scan δDB once, count frequency for each itemset in $freq_{DB}$ and output those in $freq_{DB}$ whose added support is frequent (i.e., frequent in both $(\delta DB \cup DB)$).

(3) Mine frequent itemsets in δDB. For those frequent itemsets in δDB but not in DB, scan DB once to get their frequency in DB. Output those whose added support is frequent.

□

(c) [12] Suppose the manager is only interested in *frequent* patterns (itemsets) with one of the following constraints. State the characteristics of each constraint and how to mine such patterns efficiently.

i. The sum of the price of all the items (in each pattern) is between \$100 and \$200

Answer:

(1) This statement consists of two constraints $a : sum(S.price) \geq \$100$ and $b : sum(S.price) \leq \$200$.

(2) a is a monotone constraint, which needed to be checked each time a new item is added to S. An itemset will not be checked again once it satisfies a.

(3) b is an anti-monotone constraint, which needed to be checked each time a new item is added to S. An itemset will be pruned if it does not satisfy b.

□

ii. average price of all the items in each pattern is more than \$20

Answer:

(1) $c : avg(S.price) \geq \$20$ is a convertible constraint.

(2) Sort items in each transaction in price-descending order. c becomes antimonotone. Each time when S does not satisfy c, it will be pruned since $avg(S.price)$ can never increase.

(2') [OK, but not so efficient] One can also sort items in each transaction in price-ascending order. c becomes monotone. Each time when S satisfies c, it will not need to be checked again since $avg(S.price)$ can never decrease.

□

iii. the price difference between any two items in each pattern is beyond \$10.

Answer:

(1) $d : pair\_diff(S.price) > \$10$ is an antimonotone constraint.

(2) Check when each item is added to S, and the newly formed S is pruned if it does not satisfy d. (Note: This can be very efficiently implemented if the items in a transaction is sorted.)

□

5. [3] (Opinion).

(a) I □ like □ dislike the exams in this style.

(b) In general, the exam questions are □ too hard □ too easy □ just right.

(c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

1.1.2 Final Exam

1. [10] Data preprocessing.

   (a) [5] What are the value ranges of the following correlation measures, respectively: (1) χ^2, (2) *lift*, (3) Pearson correlation coefficient, and (4) cosine measure?

   Answer:

   i. χ^2: $[0, +\infty)$

   ii. *lift*: $[0, +\infty)$

   iii. Pearson correlation coefficient: $[-1, 1]$ — Its formula is very similar to cosine

   iv. cosine measure: $[0, 1]$ — $P(A \cup B)/\sqrt{P(A)P(B)}$ when $A = B$, it is 1, when A and B independent, $P(A \cup B) = 0$

   □

   (b) [5] What are the differences among the three: (1) boxplot, (2) scatter plot, and (3) Q-Q plot?

   Answer:

   i. boxplot: show major stat of data (min, 25%tile, median, avg, 75%tile, max), whiskers and outliers.

   ii. scatter plot: plot data in its dimension space to give scattering pattern of the data

   iii. Q-Q plot: comparing two data sets by plotting their distributions on two axes of one graph. It is good to show the distribution shift between the two data sets.

   □

2. [20] Data Warehousing, OLAP and Data Cube Computation

   (a) [8] Assume a base cuboid of 10 dimensions contains only two base cells:

   $$(1)\ (a_1, a_2, a_3, b_4, \ldots, b_{19}, b_{20}), \text{ and } (2)\ (b_1, b_2, b_3, b_4, \ldots, b_{19}, b_{20}),$$

   where $a_i \neq b_i$ (for any i). The measure of the cube is *count*.

   i. How many **nonempty** aggregated (i.e., non-base) cells a complete cube will contain?

   ii. how many **nonempty** aggregated cells an iceberg cube will contain, if the condition of the iceberg cube is "*count* ≥ 2"?

   iii. How many closed cells in the full cube?

   Answer:

   i. How many **nonempty** aggregated (i.e., non-base) cells a complete cube will contain?
   $2 \times 2^{20} - 2^{17} - 2$ since only $(*, *, *, \ldots)$ are doubly counted in the full cube generated by based cells.

ii. how many **nonempty** aggregated cells an iceberg cube will contain, if the condition of the iceberg cube is "*count* ≥ 2"?
2^{17} since only those overlapped cells have count no less than 2.

iii. How many closed cells in the full cube?
3. They are: (1) $(a_1, a_2, a_3, b_4, \ldots, b_{19}, b_{20}) : 1$, (2) $(b_1, b_2, b_3, b_4, \ldots, b_{19}, b_{20}) :$ 1, and (3) $(*, *, *, b_4, \ldots, b_{19}, b_{20}) : 2$

□

(b) [6] Suppose the **variance** of n observations $x_1, x_2, \ldots, x_n$ is defined as

$$s^2 \quad = \quad \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n}[\sum x_i^2 - \frac{1}{n}(\sum x_i)^2]. \qquad (1.4)$$

where $\bar{x}$ is the average (*i.e.*, mean) value of $x_1, \ldots, x_n$. Outline an algorithm that computes the cube with variance as the measure using the **StarCubing** method.

Answer:

i. The variance is an algebraic measure, that can be computed by 3 distributed measures: (1) n, (2) $\sum x_i$ and (3) $\sum x_i^2$, following the equation.

ii. Perform star cubing on each measure in a similar way as star-cubing for count, except registering 3 measures instead of one.

□

(c) [6] Suppose one wants to compute a flight-booking data cube where the condition is that the minimum number of records is 8 and the average fare is over \$500. Outline an efficient cube computation method (based on the commonsense about the flight data distribution).

Answer:

i. Transform the above condition into two data cube conditions: min_sup = 8, and $avg(price) \geq 500$.

ii. Use a typical ice-cube computation alg. and relax the condition $avg(price) \geq 500$ into $avg^8(price) \geq 500$. Pruning can be performed if avg of top-8 price is less than 500.

iii. Top-8 can be implemented more efficiently if 8 is a big number.

□

3. [14] Frequent pattern and association mining.

(a) [6] Since items have different values and expected frequencies of sales, it is desirable to use *group-based minimum support thresholds* set up by users. For example, one may set up a small *min_support* for the group of *diamonds* but a rather large one for the group of *shoes*. Outline an Apriori-like algorithm that derive the set of frequent items efficiently in a transaction database.

Answer:

i. Mine item in each group using the min_sup specified in the group

ii. When combining them into k-itemset from different groups, use the smallest one in order to be safe to generate all the frequent itemsets

□

(b) [8] For mining correlated patterns in a transaction database, **all_confidence** (denoted as α) has been used as an interestingness measure. A set of items $\{A_1, A_2, \ldots, A_k\}$ is strongly correlated if

$$\frac{sup(A_1, A_2, \ldots, A_k)}{max(sup(A_1), \ldots, sup(A_k))} \geq min\_\alpha \qquad (1.5)$$

where min_α is the minimal all_confidence threshold, and $max(sup(A_1), \ldots, sup(A_k))$ is the maximal support among that of all the single items.

i. Based on equation (1.5), prove *all_confidence* is *antimonotonic.*

ii. Given min_σ (*i.e.*, min_sup) and min_α, state how to modify the FPgrowth method to mine such strongly correlated patterns by pushing deeply both thresholds into the mining process.

Answer:

(a) Based on equation (1.5), prove *all_confidence* is *antimonotonic.*

Since

$$sup(A_1, A_2, \ldots, A_k, A_{k+1}) \leq sup(A_1, A_2, \ldots, A_k)$$

and

$$max(sup(A_1), \ldots, sup(A_{k+1})) \geq max(sup(A_1), \ldots, sup(A_k)),$$

Adding any new item will make the quotient monotonically decreasing. Thus if the current k-itemset cannot satisfies the constraint, its corresponding $(k+1)$-itemset cannot satisfy it either. Thus it is antimonotonic.

(b) Given min_σ (*i.e.*, min_sup) and min_α, state how to modify the FP-growth method to mine such strongly correlated patterns by pushing deeply both thresholds into the mining process.

When mining, take both min_sup and all_conf into consideration. That is, when an itemset whose either min_sup or all_conf does not satisfy the thresholds, stop. Otherwise, project the conditional DB and proceed.

□

4. [30] Classification and Prediction

(a) [4] What are the major differences among the three: (1) information gain, (2) gain ratio, and (3) foil-gain?

Answer:

i. information gain: for decision-tree induction, biased on multi-valued attributes.

ii. gain ratio: for decision-tree induction, add dumping factor, result in more balanced trees.

iii. foil-gain: for rule-induction, more likely to pick the attribute value that has more positive training data associated.

□

(b) [4] What are the major differences between (1) bagging and (2) boosting?

Answer: Both belong to emsemble approach

i. bagging: simple multiple voting strategy

ii. boosting: add more weights on misclassified samples

□

(c) [4] What are the major differences among the three (1) piece-wise linear regression, (2) multiple-linear regression, and (3) regression tree?

Answer:

i. piece-wise linear regression: linear regression on each segment of data

ii. multiple-linear regression: linear regression on multiple-dimensional space

iii. regression tree: regression value at the leaf nodes of a tree.

□

(d) [9] Given a 50 GB data set with 40 attributes each containing 100 distinct values, and 512 MB main memory in a laptop, outline an efficient method that constructs decision trees efficiently, and answer the following questions explicitly: (1) how many scans of the database does your algorithm take if the maximal depth of decision tree derived is 5? (2) how do you use your memory space in your tree induction?

Answer:

i. Rainforest alg. AVC set takes much less space $40 \times 100 \times 4 \times C = 16CKB$, where C is the distinct number of class labels.

ii. Thus we only need d scans where d is the depth of the tree so constructed.

□

(e) [9] Why does an SVM algorithm have high classification accuracy in high-dimensional space? Why is an SVM (support vector machines) algorithm slow in large data sets? Outline an extended SVM algorithm that is scalable in large data sets.

Answer:

i. Why does an SVM algorithm have high classification accuracy in high-dimensional space?
SVM tries to find a hyperplane that can separate different classes with the maximal margin. In high-D space, the hyperplane is easier to be found than low-D space.. That is why SVM uses kernel function mapping original feature space to a high-D space.

ii. Why is an SVM (support vector machines) algorithm slow in large data sets?
quadratic programming technique, needs solve complex equations.

iii. Outline an extended SVM algorithm that is scalable in large data sets.

A. construct microcluster (using a CF tree)

B. Train an SVM on the centroids of the microcluster

C. Decluster entries near the boundary

D. Repeat until convergence

□

5. [26] Clustering

(a) [6] Outline the commonalities and differences of the following three algorithms: (1) *k-means*, (2) *k-medoids*, and (3) the E-M algorithm.

Answer:

i. k-means: partitioning method, center is virtual (mean of each coordinate) linear alg., fast, but outlier prob.

ii. k-medoids: partitioning method, real medoid as the cluster center, quadratic alg., slower and need sampling to handle large data sets

iii. The E-M algorithm: statistical model-based method, using underlying mixture model to do clustering. It can be used for both numerical and categorical data sets. Each object probabilistically belong to different clusters.

□

(b) [6] Different data types may need to use different similarity (distance) measures. State what is the expected similarity measure in each of the following applications: (1) clustering stars in the universe, (2) clustering text documents, (3) clustering clinical test data, and (4) clustering houses to find delivery centers in a city with rivers and bridges.

Answer:

i. clustering stars in the universe: Euclidean distance (spatial and interval based)

ii. clustering text documents: cosine (vector data) similarity

iii. clustering clinical test data: asymmetric data (e.g., Jaccard coefficient)

iv. clustering houses to find delivery centers in a city with rivers and bridges: reachable distance (not direct Euclidean distance), considering obstacles.

□

(c) [6] In a very high-dimensional data set (such as a micro-array data set), why is it the case that usually only meaningful to find clusters in some *projected* (*i.e.*, subset of) dimensional space? Outline an efficient algorithm that finds clusters in the projected dimensional space in micro-array data sets.

Answer:

i. Why projected? since (1) many irrelevant dimensions may mask clusters, (2) distance measure may become meaningless due to equi-distance, and (3) clusters may exist only in some subspaces.

ii. Outline an efficient algorithm that finds clusters in the projected dimensional space in micro-array data sets.
p-Clustering: based on pattern similarity, clustering among the objects on a subset of dimensions.

- p-clustering model: given objects $x, y \in O$ and attributes $a, b \in T$,

$$pScore(\begin{bmatrix} d_{\boldsymbol{x}a} & d_{\boldsymbol{x}b} \\ d_{\boldsymbol{y}a} & d_{\boldsymbol{y}b} \end{bmatrix}) = |(d_{\boldsymbol{x}a} - d_{\boldsymbol{x}b}) - (d_{\boldsymbol{y}a} - d_{\boldsymbol{y}b})|, \quad (1.6)$$

- A pair (O, T) forms a δ-cluster if for any 2 X 2 matrix X in (O, T), we have $pScore(X) \le \delta$ for some $\delta > 0$
- This δ-cluster has doward closure property, and frequent pattern mining algorithm can be applied to mine them.
- The method can be extended to mine both shift and scaling patterns.

□

(d) [8] A new photoprinting service chain store would like to open 20 service centers in Chicago. Each service center should cover at least one shopping center and 10,000 households of annual income over $100,000. Design a scalable clustering algorithm that takes such constraints into consideration.

Answer:

i. Use k-means clustering but take care of constraints.

ii. first partition data into k clusters satisfying constraints

iii. then perform micro-clustering for efficiency

iv. trade microclusters to reduce the sum of distances and maintain the constraints.

v. the iterative swapping process continuous until the sum of distance is minimized.

□

1.2 Sample Exam Question Set: 1.2

1.2.1 Midterm Exam

1. [30] Data preprocessing.

   (a) [8] Name four methods that perform effective *dimensionality reduction* and four methods that perform effective *numerosity reduction*.

   Answer:

   - dimensionality reduction: decision-tree, PCA, wavelets, attribute-reduction/selection.
   - numerosity reduction: Any four in {sampling, clustering, discretization, data cube, regression, histogram, data compression}.

   □

   (b) [5] Name five kinds of graphics/plots that can be used to represent *data dispersion characteristics* effectively.

   Answer:

   - five graphic plots: boxplot, Q-Q plot, histogram, quantile plot, scatter plot.

   □

   (c) [8] What are the value ranges of the following correlation measures, respectively?

   i. χ^2:
   Answer: $[0, \infty)$ □

   ii. *lift*:
   Answer: $[0, \infty)$ □

   iii. *Pearson correlation coefficient*:
   Answer: $[-1, 1]$ (Note I would give at least 50%, i.e., 1 point, for the answer: $(-\infty, \infty)$ since it is hard to see $[-1, 1]$ based on the formula only.) □

   iv. *all-confidence*:
   Answer: $[0, +1]$ □

   (d) [9] For the following group of data

   $$200, 400, 800, 1000, 2000$$

   i. Calculate its mean and variance.
   Answer: mean $= 880$, variance $= \frac{1}{5} \times (584 \times 10^4) - 880^2 = 116.8 \times 10^4 - 77.44 \times 10^4 = 393600$. □

   ii. Normalize the above group of data by min-max normalization with min $= 0$ and max $= 10$; and
   Answer: normalized sequence: 0, 1.11, 3.33, 4.44, 10 □

iii. In z-score normalization, what value should the first number 200 be transformed to?
Answer: $(200 - 880)/\sqrt{393600} = -680/627.38 = -1.08$ □

2. [16] Data Warehousing and OLAP for Data Mining

Suppose a shipping data warehouse consists of the following dimensions: *customer, time, product, from_location, to_location*, and *shipper*, and the following measures: *quantity*, *weight*, *cost*, and *charge*, where *cost* is the cost of shipping, and *charge* is the shipping fee charged to the customer.

(a) [4] Draw a **snowflake schema** diagram (sketch it, do not have to mark every possible level, and make your implicit assumptions on the levels of a dimension when you draw it).

Answer: main: [*customer, time, product, from_location, to_location, shipper*]: *quantity*, *weight*, *cost*, and *charge*;

customer: [customer-key, name, location (address, city, state, country), group]

time: [time-key, day, week, month, year]

from_location: [from-location-key, location(street, city, state, country)]

to_location: [from-location-key, location(street, city, state, country)]

shipper: [shipper-key, name, group, company, location(street, city, state, country)]

□

(b) [7] Starting with the base cuboid, what **specific OLAP operations** (e.g., roll-up which dimension from which level to which level) that one should perform in order to find the average monthly profit for shipping each brand of *TV* from New York to Chicago *since March 2006*?

Answer:

- dice on product, from_location, to_location, time
- roll-up from_location to [city], slice on (i.e., select) New York
- roll-up to_location to [city], slice on (i.e., select) Chicago
- roll-up time to [year], slice on (i.e., select) 2006
- drill-down time to [month], select>= March
- dice on product = TV and drill-down to brand
- monthly average profit is obtained by: profit = charge − cost

□

(c) [5] Suppose a cube has N dimensions and a dimension d_i is represented by a lattice of P_i nodes as shown in Figure 1.2 (not including *all*), *how many cuboids* does this cube contain (including base and apex cuboids)?

Answer: $\Pi_{i=1}^{N}(P_i + 1)$ □

Figure 1.2: A dimension such as *date* can be represented as a lattice

3. [25] Data cube implementation

   (a) [9] Assume a base cuboid of 20 dimensions contains only two base cells:

   (1) $(a_1, a_2, a_3, b_4, \ldots, b_{19}, b_{20}) : 1$, and (2) $(a_1, b_2, b_3, b_4, \ldots, b_{19}, b_{20}) :$ 1,
   where $a_i \neq b_i$ (for any i). The measure of the cube is *count*.

      i. How many **nonempty** aggregated (i.e., non-base) cells a complete cube will contain?
      Answer: $2 \times 2^{20} - 2 - 2^{18}$, because each cell will generated $(2^{20} - 1)$ aggregate cells but when $(\cdot, a_2, a_3, \ldots)$ and $(\cdot, b_2, b_3, \ldots)$ turn to $(\cdot, *, *, \ldots)$, they are overlapped and merged, and since there are 2^{18} such cells are doubly counted, they should be removed. □

      ii. how many **nonempty** aggregated cells an iceberg cube will contain, if the condition of the iceberg cube is "*count* ≥ 2"?
      Answer: 2^{18} □

      iii. How many closed cells in the full cube?
      Answer: 3. They are:
      (1) $(a_1, a_2, a_3, b_4, \ldots, b_{19}, b_{20}) : 1$,
      (2) $(a_1, b_2, b_3, b_4, \ldots, b_{19}, b_{20}) : 1$, and
      (3) $(a_1, *, *, b_4, \ldots, b_{19}, b_{20}) : 2$. □

   (b) [5] Explain why the data cube could be quite sparse, and how one should implement such a sparse cube if one adopts an array-cube implementation.

   Answer: Since the most of the cells in the multidimensional space are usually empty because in reality it is unlikely there are nonempty cells for all the possible value combinations from all the participating dimensions, e.g., few, if any, students can take all the offered course in every semester. We need to adopt sparse-array representation, such as (chunk-id, offset) as the cell address space and reserve no space for empty cells. □

   (c) [6] List the name of the corresponding cube computation method that is the best to implement one of the following:

      (a) computing a dense full cube of low dimensionality (e.g., less than 6 dimensions),
      Answer: multiway array-cube computation. □

      (b) computing a large iceberg cube of around 10 dimensions, and
      Answer: star-cubing or BUC. □

(c) performing OLAP operations in a high dimensional database (e.g., over 100 dimensions).
Answer: shell-fragment for high-D OLAP. □

(d) [5] Suppose a disk-based large relation contains 50 attributes. What is the minimum number of database scans in order to derive a generalized relation by *attribute-oriented induction*?

Answer: Two is a correct answer, where the first scan is to collect statistics to determine the level of generalization for each attribute, and the second is for database scanning to generate prime generalization relation.

A better solution is $1 + \delta$, where δ is for sampling to determine the level of generalization for each attribute, and one is for database scanning to generate prime generalization relation. □

4. [26] Frequent pattern and association mining.

Figure 1.3: FP tree of a transaction DB

(a) [8] A database with 100 transactions has its FP-tree shown in Fig. 1.4. Let $min\_sup = 0.4$ and $min\_conf = 0.7$. Show

i. c's conditional (i.e., projected) database:
Answer: $\{eab : 30, ea : 20, eb : 10\}$. □

ii. all the frequent k-itemsets for the largest k:
Answer: $\{cea : 50, ceb : 40, eab : 45\}$. □

iii. all the strong association rules (with support and confidence) containing the k items (for the largest k only):
Answer: From the 2-itemsets point of view, we have $ca : 50, ce : 60, cb : 40, eb : 45, ae : 80, ab : 45$, with $\{cea : 50, ceb : 40, eab : 45\}$, we have the following rules:
$ca \rightarrow e$ $[50, 100\%]$; $ce \rightarrow a$ $[50, 83.3\%]$; $cb \rightarrow e$ $[40, 100\%]$
$be \rightarrow a$ $[45, 100\%]$; $ab \rightarrow e$ $[45, 100\%]$
From the one-item point of view, we have $a : 80, b : 45, c : 60, e : 100$, with $\{cea : 50, ceb : 40, eab : 45\}$, we have the following rules:
$c \rightarrow ae$ $[50, 76.9\%]$; $b \rightarrow ae$ $[45, 100\%]$. □

(b) [6] Give one example in a shopping transaction database, two items a and b can be strongly associated (such as $a \rightarrow b$) but negatively correlated.

Answer: Give an example similar to the basketball and cereal presented in the textbook/slides. □

(c) [12] Suppose the manager is only interested in *frequent* patterns (itemsets) with one of the following constraints. State the characteristics of each constraint and how to mine such patterns efficiently.

i. The sum of the price of all the items (in each pattern) over \$10 in price is between \$100 and \$200.

Answer:

There are two constraints: $C_1 : sum(S.price) >= 100$ and $C_2 : sum(S.price) <= 200$ where S is a set of items with price over \$10.

C_1 is monotone and C_2 antimonotone. Push C_2 deep and not check C_1 is its constraint is satisfied (or using C_1 to prune the remaining items in the transactions that cannot satisfy it).

Notice that $price > 10$ is not a constraint for the mining query since it only put restriction on the items in the constraints. We cannot prune all the items whose price is not bigger than 10. □

ii. the price difference between the two most expensive items in each pattern must be beyond \$10.

Answer:

One constraint $C_3 : diff(I_1.price, I_2.price) > \$10$, where I_1 and I_2 are the most and the second most expensive items respectively in each pattern.

C_3 is a convertible anti-monotone and monotone constraint. Method: Sort items in each transaction in price descending order. If C_3 cannot satisfy a pattern, remove the pattern; otherwise, keep growing and no need to check C_3 any more.

Notice that some students treat C_3 as a monotone and succinct constraint, and the process starts with the items satisfying this constraint. This process could be right if the constraint were "the price difference between the two most expensive items in each transaction must be beyond \$10". But we require the constraint be in the pattern, not in the transaction. □

iii. average price of all the items in each pattern is less than \$10.

Answer:

One constraint $C_4 : avg(S.price) < \$10$. It is a convertible anti-monotone constraint if the items in each transaction are sorted in price ascending order. If any pattern S's price avg violates C_4, remove it since the latter ones will be more expensive. □

5. [3] (Opinion).

(a) I □ like □ dislike the exams in this style.

Like: Dislike = 33: 7 and I2CS 8: 0

(b) In general, the exam questions are □ too hard □ too easy □ just right.

Too Hard: Too Easy : Just Right = 25: 0: 14 I2CS 2: 0: 6

(c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

Plenty of time: Just enough time: Not enough time = 2: 11: 25 I2CS 0: 3: 5

1.2.2 Final Exam

1. [10] Data preprocessing.

   (a) [6] Data integration is essential in many applications. Suppose we are given a large data relation with many tuples, with the attributes Student_Name, University, Address, and so on. Discuss how to discover a set of different strings that represent the same entity, such as "UIUC", and "University of Illinois at Urbana Champaign", and thus should be integrated?

   Answer:

   i. Discover strong correlation among a set of attributes to determine the merging rule. E.g., based on DB info, one may find "city -¿ university" is a almost-true rule, and abbreviation can be used as another rule. By confirmation with training/expert, one can set up a merging rule. Then "University of Illinois at Urbana-Champaign" and "UIUC" can be merged.
   Or,

   ii. By training, one can find merging rule as well. □

   (b) [4] What is the value range for each of the following normalization methods?

   i. min-max normalization,
   Answer: [new-min, new-max] □

   ii. z-score normalization, and
   Answer: $(-\infty, +\infty)$ □

   iii. normalization by decimal scaling?
   Answer: $(-1, +1]$ □

2. [13] Data Warehousing, OLAP and Data Cube Computation

   (a) [7] Suppose the **standard deviation** of n observations $x_1, x_2, \ldots, x_n$ is defined as

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} = \sqrt{\frac{1}{n}[\sum x_i^2 - \frac{1}{n}(\sum x_i)^2]}. \quad (1.7)$$

   where $\bar{x}$ is the average (*i.e.*, mean) value of $x_1, \ldots, x_n$. Is this measure a distributive, algebraic or holistic measure? Outline an efficient algorithm that computes an *iceberg cube* with standard deviation as the measure, where the iceberg condition is $n \geq 100$ and $\sigma \geq 2$.

   Answer: Algebraic since one can store n, $\sum x_i^2$, and $\sum x_i$ three distributed measures for online computation.

   Use BUC/StarCubing/H-cubing one can push $n \geq 100$ inside but not $\sigma \geq 2$ since the latter is not monotonic. □

(b) [6] A data cube C has D dimensions, and each dimension has exactly p distinct values in the base cuboid (assuming a dimension contains no levels, i.e., no hierarchies).

i. What could be the **maximum number of cells** of the base cuboid?
Answer: p^D. □

ii. What could be the **minimum number of cells** of the base cuboid?
Answer: p. □

iii. What could be **maximum number of cells** (including both base cells and aggregate cells) in the data cube C?
Answer: $(p+1)^D$. □

3. [12] Frequent pattern and association mining.

(a) [6] Since items have different values and expected frequencies of sales, it is desirable to use *group-based minimum support thresholds* set up by users. For example, one may set up a small *min_support* for the group of *diamonds* but a rather large one for the group of *shoes*. Outline an Apriori-like algorithm that derive the set of frequent items efficiently in a transaction database.

Answer: Use group min_support to prune itemset generation in each group and use the smallest min_support to prune mixed itemsets. One can generate an efficient alg. □

(b) [6] Associative classification often involves first mining association rules and then performing classification. Suppose one wants to use an FPtree-like structure to facilitate mining rules for associative classification. Discuss how the original FP-tree data structure and the algorithm should be modified to make the mining efficient.

Answer:

For each tree branch, do not store class label but put each class count into the combined node count. Then mine the rules. If the node combined count is less than min_support, stop mining. The rules generated will still be sorted by confidence and then support. □

4. [35] Classification and Prediction

(a) [5] What are the major differences among the three: (1) Naïve-Bayesian algorithm, (2) Bayesian Belief Network, and (3) Neural Network?

Answer:

Differences between (1) (2) vs. (3):

(1)(2) are generative classification where the data is assumed to following some distribution. For new samples, the probability belonging to each class is computed based on the model constructed from the training set, and the new example is assigned to the class with the maximum posterior probability.

(3) is discriminative classification where a function is learnt using training data to separate classes. In Neural network, the function is expressed using a network topology.

Differences between (1) and (2):

Naive-Bayesian assumes the conditional independence of each attribute whereas Bayesian Believe Network incorporates the dependency between attributes.

□

(b) [5] Both decision-tree induction and associative classification may generate rules for classification. What are their major differences? Why is it that in many cases an associative induction may lead to better accuracy in prediction?

Answer:

Since decision-tree induction involves only one attribute each time, the rules generated by decision-tree are constrained. Associative classification finds strong associations better frequent patterns and class labels. The rules usually combine multiple attributes. In this context, the latter can represent more complicated boundary, thus leads to better accuracy in prediction. □

(c) [6] What are the major differences among the three: (1) rule-based induction, (2) case-based reasoning, and (3) k-nearest neighbor classification?

Answer:

Differences between (1) and (2)(3):

(1): eager classifier, where a model (represented by a set of rules) is constructed based on training data, then use this model to classify a new example.

(2)(3): lazy classifier, where training data is stored and test example is classified by comparing it with the training data. No model is constructed beforehand.

Differences between (2) and (3):

the k-nearest neighbor approach regards each example is a point in a d-dimensional space, and classifies the test example by common class among its nearest neighbors. Case-based reasoning searches for similar cases of test examples, and classifies test example using the class of its similar case. □

(d) [9] Suppose you are requested to classify microarray data with 100 tissues and 10000 genes. Which of the following algorithms you would like to choose and which ones you do not think they will work? State your reasons.

(1) Decision-tree induction, (2) piece-wise linear regression, (3) SVM, (4) associative classification, (5) genetic algorithm, and (6) Bayesian Belief Network.

Answer:

(1) Decision-tree induction: Not work, too many attributes.

(2) piece-wise linear regression: Not work, need to huge number of parameters.

(3) SVM: works since it scales to large dimensions.

(4) associative classification: works if the method of frequent pattern mining is further developed as shown in Tung's paper.

(5) genetic algorithm: Not work since it may invoke an exponential number of mutations.

(6) Bayesian Belief Network: Not work since the sample size is too small, the probability distribution cannot be estimated accurately. □

(e) [10] Given a training set of 50 million tuples with 25 attributes each taking 4 bytes space. One attribute is a class label with two distinct values, whereas for other attributes each has 30 distinct values. You have only a 512 MB main memory laptop. Outline an efficient method that constructs decision trees efficiently, and answer the following questions explicitly: (1) how many scans of the database does your algorithm take if the maximal depth of decision tree derived is 5? (2) what is the maximum memory space your algorithm will use in your tree induction?

Answer:

Use RainForest algorithm.

(1) 5 scans since at each level, one AVC group of all nodes are computed in parallel. Scan DB once to build all the ACM groups at each level.

(2) Size computation

Method 1. Suppose each entry in AVC group take 4 bytes and each count for the class label has 2 bytes (thus 2 values has 4 bytes as well. The space for the root node is $30 \times (24 + 1) \times 4 = 3K$ bytes.

The other levels involve less attributes. But it has more tables. Suppose we use Information Gain approach, since we will generate 30^3 possible split nods at the fourth level, we will have approximately $30^3 \times 3K = 27 \times 3M$ bytes, which is about 81 MB.

Note, a possible optimization is the sharing of such tables, in this case, we will get $6K + 30^4 \times 4 = 6K + 3240K = 3.246MB$.

Method 2. Suppose each entry in AVC group take C bytes. The space for the root node is $24 \times 30 \times 2 \times C = 1440C$ bytes.

The other levels involve less attributes. But it has more tables. Suppose we use Information Gain approach, since we will generate 30^3

possible split nods at the fourth level, we will have approximately $30^3 \times 21 \times 30 \times 2 \times C = 34MC$ bytes. If C = 4 bytes, it is about 136 MB. □

5. [30] Clustering

(a) [8] Choose the best clustering algorithm you know for the following tasks (and reason on your choice using one sentence):

(1) clustering Microsoft employees based on their working-years and salary,

Answer:

CLARANS, i.e., scalable k-medoids algorithm.

□

(2) clustering houses to find delivery centers in a city with rivers and bridges, and

Answer:

Constraint-based clustering where constraints are obstacles. □

(3) distinguishing snakes hidden in the surrounding grass.

Answer:

Density-based clustering like DBSCAN. □

(b) [6] Anges, BIRCH, and Chameleon are all hierarchical clustering algorithms. Rank them based on clustering quality. Also, rank them based on scalability to large datasets. Briefly justify your rank.

Answer:

Quality ranking: Chameleon, BIRCH, AGNES

Scalability ranking: BIRCH, AGNES $O(n \log n)$, and CHAMELEON $O(N^2)$ □

(c) [6] What are the major difficulty to cluster a micro-array data set? Outline one efficient and effective method to cluster a micro-array data set.

Answer:

Major challenge: very large number of attributes, high-dimensional clustering, need to explore subspace clustering

Method: p-clustering

□

(d) [10] YouTube website contains a large set of video clips. Design an efficient method that can group such video clips into a set of clusters effectively.

Answer:

A good answer could be:

i. maximal use of the available information to do pattern matching and clustering

ii. Assume some video clips may have: title, author, theme, category, etc. but all the video clips will have image, sound, etc. U-tube may have some user clickstream information

iii. Based on similar categories, or themes and user-clickstreams, one can group those clips of videos into clusters. That is, used click patterns will also be useful: assuming similar users are interested in similar videos.

iv. For those clusters, we can work out models that may distinguish features of images and sound of different clusters.

v. For the remaining clips, one can use these models to put them into the right clusters.

vi. For some ambiguous clips, user interactions/feebacks will help and such interactions will refine the model and help better clustering of the remaining/later ones.

□

1.3 Sample Exam Question Set: 1.3

1.3.1 Midterm Exam

1. [30] Data preprocessing.

(a) [6] For data visualization, there are three classes of techniques: (i) geometric techniques, (ii) hierarchical techniques, and (iii) icon-based techniques. Give names of two methods for each of these techniques.

Answer:

(i) geometric techniques: scatterplot matrices, parallel coordinates, landscapes

(ii) hierarchical techniques: dimension stacking, tree maps, cone trees, info cube

(iii) icon-based techniques: Chernoff face, stick figures, color icons, tile bars

□

(b) [4] What are the value ranges of the following correlation measures, respectively?

i. χ^2:
Answer: $[0, +\infty)$ □

ii. *Pearson correlation coefficient*:
Answer: $[-1, +1]$ □

(c) [8] Name four methods that perform effective *dimensionality reduction* and four methods that perform effective *numerosity reduction.*

Answer:

(i) Dimensionality reduction: SVD, PCA, decision tree, feature subset selection, feature creation, one may also count: wavelet/Fourier transformation

(ii) Numerosity reduction: data compression, regression, clustering, sampling, binning, discretization, histogram, data cube aggregation, and also wavelet/Fourier transformation

□

(d) [8] For the following group of data

$$100, 200, 400, 800, 1500$$

i. Calculate its mean and variance.
Answer:
(i) mean: $\mu = (100 + 200 + 400 + 800 + 1500)/5 = 600$
(ii) variance : $\sigma^2 = 1/5[(100 - 600)^2 + (200 - 600)^2 + (400 - 600)^2 + (800 - 600)^2 + (1500 - 600)^2] = 260000$

□

ii. Normalize the above group of data by min-max normalization with min = 0 and max = 10; and

Answer:

100: 0
200: $((200 - 100)/(1500 - 100)) \times 10 + 0 = 0.714$
400: $((400 - 100)/(1500 - 100)) \times 10 + 0 = 2.143$
800: $((800 - 100)/(1500 - 100)) \times 10 + 0 = 5$
1500: 10

□

iii. In z-score normalization, what value should the first number 100 be transformed to?

Answer:

100: $(100 - \mu)/\sigma = (100 - 600)/\sqrt{260000} = -0.98$

□

(e) [4] What are the best distance measure for each of the following applications:

(i) driving distance between two locations in Downtown Chicago,

Answer: Manhattan distance □

(ii) compare similar diseases with a set of medical tests,

Answer: Dissimilarity for asymmetric binary variables or Jaccard coefficient, i.e., $(b + c)/(a + b + c)$ □

(iii) find similar web documents

Answer: Cosine measure of two vectors, i.e., the inner product of two feature vectors, each representing the features (such as keywords or terms) of a document. □

2. [13] Data Warehousing and OLAP for Data Mining

(a) [3] Suppose a cube has 10 dimensions and the i-th dimension has M_i levels (not including *all*), *how many cuboids* does this cube contain (including base and apex cuboids)?

Answer: $\Pi_{i=1}^{10}(M_i + 1)$ □

(b) [4] Give two examples for each of the following two kinds of measures: (i) algebraic, and (ii) holistic.

Answer:

(i) algebraic: average, variance

(ii) holistic: median, Q_1, rank

□

(c) [6] Suppose the academic office of UIUC wants to build a Student_Record data warehouse with the following information: *student, major, course, department, grade*, and would like to calculate student GPA, major_gpa, etc.

(i) Draw a **snowflake schema** diagram (sketch it, and make your implicit assumptions on the levels of a dimension and the necessary measures).

Answer:

There could be many different answers in the design. One possible answer could be as follows.

Dimensions:

- Student(name, major, birth_place (...), ...),
- Department (dname, college, head, ...),
- Course(cname, cno, credit, instructor, ...),
- Time (semester, year, ...)

Measure: total_# = count(*), GPA = avg(grade), ...

The dimension tables should be linked to the fact table.

□

(ii) If one would like to start at the Apex cuboid and find top_10 students in each department in the College of Engineering based on their GPA up to Spring 2007, what are the **specific OLAP operations** (e.g., roll-up on which dimension from which level to which level) that one should perform based on your design?

Answer: OLAP operations:

Drill down on Department from * to College-level

Drill down on Time dimension from * to year-level

Dice on (i.e., select) college = ”Engineering” and Year = ”2007”

Drill down on Time to season and slice on season = ”Spring”

Drill down on Department to the department-level

Drill down on Student dimension to student name (or ID)

Select top_10 GPA values, and print the corresponding student names

□

3. [25] Data cube implementation

(a) [10] Assume a base cuboid of N dimensions contains only p (where $p > 3$) nonempty base cells, and there is no level associated with any dimension.

i. What is the *maximum number of nonempty cells* (including the cell in the base cuboid) possible in such a materialized datacube?

Answer:

Each cell generates 2^N cells. So p cells will generate in total $p \times 2^N$ cells. However, the p cells at the Apex cuboid are merged into one, i.e., we need to minus $p - 1$ cell count. Thus the maximum number of cells generated will be:

$$p \times 2^N - p + 1$$

□

ii. If the minimum support (i.e., iceberg condition) is 3, what is the *minimum number of nonempty cube cells* possible in the materialized iceberg cube?
Answer:
Each cell generates 2^N cells. However, these cells may not have to be merged together at the k dimensional cuboids (for $1 < k < N$) to increase its count. Until at the last moment, i.e., in the Apex cuboid, they have to be merged into one. In this case, it will generate one cell with count of $p > 3$. Thus the minimum number of cells generated will be: 1.

□

iii. If the minimum support is 2, what is the *maximum number of nonempty cells* possible in the materialized iceberg cube?
Answer:
The earliest stage that these cells may be merged together to form support 2 cells is at the $N-1$ dimensional cuboids. At this time, the maximum number of cells that one may form at each plane is $\lfloor \frac{p}{2} \rfloor$. Thus we can view the original cells are essentially $\lfloor \frac{p}{2} \rfloor$ support two cells at the total of N such $(N-1)$-dimensional spaces. Similar to question 1, they may generate in total $N \times \lfloor \frac{p}{2} \rfloor \times 2^{N-1}$ cells, except the Apex cuboid has p cells merged into one. Thus the maximum number of cells generated will be:

$$N \times \lfloor \frac{p}{2} \rfloor \times 2^{N-1} - p + 1$$

□

(b) [10] Given the following four methods: *multiway array cubing* (Zhao, et al. SIGMOD'1997), *BUC* (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD'2001), *StarCubing* (Xin et al., VLDB'2003), and *shell-fragment* approach (Li et al, VLDB'2004), list one method which is the best and another which is the worst (or not working) to implement one of the following:

(a) computing a dense full cube of low dimensionality (e.g., less than 6 dimensions),
Answer: Best: Array-cubing. Worst: Shell-Fragment □

(b) computing a large iceberg cube of around 8 dimensions, and
Answer: Best: BUC or StarCubing. Worst: Array-Cubing □

(c) performing OLAP operations in a high-dimensional database (e.g., over 50 dimensions).
Answer:
Best: Shell-Fragment
Not-working: all the other three: BUC, StarCubing, and Array-Cubing
Note: Answering any one of the three will be OK □

(c) [5] Suppose a disk-based large relation contains 100 attributes. What is the minimum number of database scans in order to derive a generalized relation by *attribute-oriented induction*?

Answer:

Two scans: one preparing for generalization, one performing generalization (which will derive a small, memory-resident prime-relation)

or

$1+\delta$ scan since the first scan can be replaced by a δ scan or sampling.

□

4. [30] Frequent pattern and association mining.

Figure 1.4: FP tree of a transaction DB

(a) [8] A database with 100 transactions has its FP-tree shown in Fig. 1.4. Let $min\_sup = 0.5$ and $min\_conf = 0.8$. Show

i. c's conditional (i.e., projected) database:
Answer: bae: 35, ae: 30, be: 5

□

ii. all the frequent k-itemsets for the largest k:
Answer: k = 3; cae: 65

□

iii. two strong association rules (with support and confidence) containing the k items (for the largest k only):
Answer: Note: ac: 65, ae: 80, ce: 70, a: 80, c: 70, e: 100. Thus we have

$c \rightarrow ae \quad s = .65, \sigma = 65/70 = .92$

$a \rightarrow ce \quad s = .65, \sigma = 65/80 = .81$

$ae \rightarrow c \quad s = .65, \sigma = 65/80 = .81$

$ac \rightarrow e \quad s = .65, \sigma = 65/65 = 1.00$

$ce \rightarrow a \quad s = .65, \sigma = 65/70 = .92$

□

(b) [8] To further improve the Apriori algorithm, several *candidate generation-and-test* methods are proposed that *reduce the number of database scans* at mining. Briefly outline two such methods.

Answer: Any of the following algorithms will count:

1. Partitioning: partition DB into k portions, each fit in memory; mine each local partition; merge freq-itemsets; then one more scan DB to consolidate the global patterns.

2. Hashing: First scan, count freq-1 and hash 2-itemsets into buckets. If the bucket count < threshold, all the 2-itemsets in it are infreq.

3. DIC: Scan to count freq-1, and if 1-freq., start counting cand-2-itemsets, and so on.

□

(c) [6] Suppose a transaction database contains N transactions, ct transactions contain both coffee and tea, $c\overline{t}$ transactions contain coffee but not tea, $\overline{c}t$ transactions contain tea but not coffee, and $\overline{c}\overline{t}$ transactions contain neither tea nor coffee.

(i) What is the *null-invariance* property?

Answer: A measure not influenced by the count of $\overline{c}\overline{t}$ (i.e., those containing neither coffer nor tea).

□

(ii) give the names or definitions of three *null invariant measures.*

Answer: all-conf, coherence, Kulczynski, cosine, max-conf □

(d) [8] Suppose a manager is interested in only the *frequent patterns* (i.e., *itemsets*) that satisfy certain constraints. For the following cases, state the characteristics of *every constraint* in each case and how to mine such patterns efficiently.

i. The price difference between the most expensive item and the cheapest one in each pattern must be within \$100.
Answer:
$C : range(S.price) \leq \$100$ is antimontonic.
Method: Push C into iterative mining, toss S if it cannot satisfy C. □

ii. The sum of the profit of all the items in each pattern is between \$10 and \$20, and each such item is priced over \$10.
Answer:
$C_1 : min(S.price) > \$10$ is succinct, or data anti-monotone.
Method: Push C_1 into iterative mining, select only items satisfying C_1
$C_2 : sum(S.profit) \geq \$10$ is ***convertible*** monotone, $C_3 : sum(S.profit) \leq$ \$20 is ***convertible*** antimonotone, if items within a transaction is sorted in profit ascending order.
Method: Push C_3 into iterative mining, toss S if it does not satisfy C_3 and check C_2, and once its satisfies, no more checking needed.
[Note: Both C_2 and C_3 can be used as data antimonotone constraints to prune t_i if the remaining items in t_i with the current S cannot satisfy C_2.]

□

5. [3] (Opinion).

(a) I □ like □ dislike the exams in this style.

Answer: L: 52, D: 31, not sure: 1 □

(b) In general, the exam questions are □ too hard □ too easy □ just right.

Answer: H: 52, E: 0, R: 32 □

(c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

Answer: P: 10, E: 36, N: 38 □

1.3.2 Final Exam

1. [14] Data preprocessing.

   (a) [6] We have learned at least three correlation measures: (1) χ^2, (2) Pearson's correlation coefficient, and (3) Kulczynski measure.

      i. Explain what are the major differences among the three measures, and
      Answer:
      Better answer: Given definition formula for each and point out Kulczynski is null-invariant but the other two are not.
      χ^2: for categorical data correlation.
      Pearson's correlation coefficient: for numerical data correlation, [-1, 1], 1 is strongly positively correlated, -1 is strongly negatively correlated, 0: independent.
      Kulczynski: for item correlation in large transaction database, close to 1 strongly correlated, 0 strongly negative correlated.
      □

      ii. give one example for each of the three cases that one is the most appropriate measure.
      Answer:
      χ^2: Use an example like milk vs. bread or play-basketball vs. eat cereal to show it is useful.
      Pearson's correlation coefficient: use any number series or plot.
      Kulczynski: Use a transaction DB where null is somewhat big, or anything that expresses the meaning.
      As long as the idea is correct, it should be fine. □

   (b) [8] (Distance measures)

      i. Give the name of the measure for the distance *between two objects* for each of the **4** *different kinds of data.*
      Answer:
      typical four but others will be ok.
      numerical (interval) data: Euclidean, or Mahattan, or Minkowski
      asymmetric binary: Jaccard coefficient
      vector (e.g., text document): cosine
      nominal: $(p - m)/p$ □

      ii. Give the names of **4** measures for the distance *between two clusters* for numerical data.

   Answer:

   Single-link (shortest distance), complete link (max-distance), average-link, centroid (or medoid) distance.

   □

2. [14] Data Warehousing, OLAP and Data Cube Computation

(a) [7] Assume a base cuboid of **20** dimensions contains only two base cells:

(1) $(a_1, a_2, b_3, b_4, \ldots, b_{19}, b_{20})$, and (2) $(b_1, b_2, b_3, b_4, \ldots, b_{19}, b_{20})$,

where $a_i \neq b_i$ (for any i). The measure of the cube is *count*.

i. How many **nonempty** aggregated (i.e., non-base) cells a complete cube will contain?
Answer:
$7 \times 2^{18} - 2$ Some may like to write as $2 \times 2^{20} - 2^{18} - 2$

□

ii. how many **nonempty** aggregated cells an iceberg cube will contain, if the condition of the iceberg cube is "*count* ≥ 2"?
Answer:
2^{18} when the first two dimension becomes $(*, *)$

□

iii. How many *closed cells* in the full cube? Note that a cell is *closed* if none of its descendant cells has the same measure (*i.e.*, count) value. For example, for a 3-dimensional cube, with two cells: "$a_1 a_2 a_3 : 3$", "$a_1 * a_3 : 3$", the first is closed but the second is not.
Answer:
3. They are:
$(a_1, a_2, b_3, b_4, \ldots, b_{19}, b_{20}) : 1$
$(b_1, b_2, b_3, b_4, \ldots, b_{19}, b_{20}) : 1$
$(*, *, b_3, b_4, \ldots, b_{19}, b_{20}) : 2$

□

(b) [7] Suppose the **standard deviation** of n observations $x_1, x_2, \ldots, x_n$ is defined as

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} = \sqrt{\frac{1}{n}[\sum x_i^2 - \frac{1}{n}(\sum x_i)^2]}. \quad (1.8)$$

where $\bar{x}$ is the average (*i.e.*, mean) value of $x_1, \ldots, x_n$.

i. Is this measure a distributive, algebraic or holistic measure? and why?
Answer:
Algebraic since n, $\sum x_i^2$ and $\sum x_i$ are all distributive.

□

ii. Outline an efficient algorithm that computes an *iceberg cube* with standard deviation as the measure, where the iceberg condition is $n \geq 100$ and $\sigma \geq 2$.
Answer:
Use BUC (or Star-Cubing), everytime when we partition and sort the data, we record not only the count but also the other two measures: $\sum x_i^2$ and $\sum x_i$.

If a partition cannot make $n \geq 100$, we can prune it (no further operation on this partition). O.W., check $\sigma \geq 2$, output the cell. If not, continue the operation on the partition.

□

3. [12] Frequent pattern and association mining.

   (a) [6] Given a fixed *min_support* threshold, σ (*e.g.*, $\sigma = 0.5\%$), present an efficient incremental mining algorithm that can maximally use the previously mined information when a new set of transactions ΔTDB is added to the existing transaction database TDB.

   Answer:

   Based on the principle: *An itemset is frequent in* $\Delta TDB + TDB$*, it must be frequent in at least one of them.* (easy to prove).

   Then assume we have all of the frequent itemset I for TDB. Mine frequent itemset dI in ΔTDB. For each $i \in I$ and $i \in dI$, add their support and output them (they must be frequent). For each i in I but not in dI, add their count in ΔTDB to see if it is frequent. For each i in dI but not in I, add their count in TDB (which needs to scan TDB once) to see if it is frequent.

   □

   (b) [6] Explain why both *Apriori* and *FPgrowth* algorithms may encounter difficulties at mining colossal patterns (the patterns of large size, *e.g.*, 100). A new algorithm based on *core pattern fusion* can mine such patterns efficiently. Explain why such an algorithm is efficient and effective at mining most of the colossal patterns.

   Answer:

   Difficulty of *Apriori* and *FPgrowth*: explosive number of mid-sized patterns, because a frequent large pattern will contain an explosive number of mid-sized patterns.

   Method: core-pattern fusion: Based on the idea that a large pattern can be partitioned into midsized patterns in many different ways, and thus many midsized patterns will be able to merge back to get closer back to the large patterns.

   It traverses the tree in a bounded breadth way. Only a fixed number of patterns in the current candidate pool will be used as the starting nodes to go down in the pattern tree. Thus it avoids the exponential search space.

   Moreover, it identifies shortcuts whenever possible by agglomeration of multiple patterns in the pool. It will quickly towards colossal patterns.

   □

4. [34] Classification and Prediction

(a) [6] All the following three methods may generate rules for induction: (1) *decision-tree induction*, (2) *sequential covering rule induction*, and (3) *associative classification*. Explain what are the major differences among them.

Answer:

(1) *decision-tree induction*: select one attribute at a time based on information gain or other measures, and then split DB and then do it recursively. Then map them into rules. Rules are order independent.

(2) *sequential covering rule induction*: generate one rule that covers some positive examples but no negative examples. Then remove those covered positive examples, and do rule generation again. The rule generated in sequence and order of application is important.

(3) *associative classification*: generate rules by mining those association rules whose RHS is class label only. Then sort rules by confidence and then support. Rule application is order-dependent. □

(b) [6] What are the major differences among the three methods for increasing the accuracy of a classifier: (1) *bagging*, (2) *boosting*, and (3) *ensemble*?

Answer:

(1) *bagging*: averaging the prediction over a collection of classifiers.

(2) *boosting*: use weighted vote with a collection of classifiers, and assign more weights to those examples who were misclassified by the previous classifier in the next round of induction.

(3) *ensemble*: combining a set of heterogeneous classifiers.

□

(c) [6] What are the major differences among the three methods for the evaluation of the accuracy of a classifier : (1) *hold-out method*, (2) *cross-validation*, and (3) *boostrap*?

Answer:

(1) *hold-out method*: use part of the data (e.g., 2/3) for training and the remaining for testing.

(2) *cross-validation*: partition data into (relatively even) k portions, $D_1, \ldots, D_k$, use D_i for testing and the other $k-1$ portions for training for any i, and then merge the results.

(3) *boostrap*: works for small data set, it samples the given training data uniformly with replacement, *e.g.*, 0.632 boostrap.

□

(d) [8] Suppose you are requested to classify microarray data with 100 tissues and 1000 genes. Which of the following algorithms you would like to choose and which ones you do not think they will work? State your reasons.

(1) Decision-tree induction, (2) piece-wise linear regression, (3) SVM, (4) PatClass (pattern-based classification), (5) genetic algorithm, and (6) Bayesian Belief Network.

Answer:

SVM and PatClass: works for high-D data

decision-tree, piecewise linear, genetic alg do not work for too high-D data.

Bayesian Belief Network: hard to work unless people give the network topology.

□

(e) [8] Given a training set of 50 million tuples with 25 attributes each taking 4 bytes space. One attribute is a class label with two distinct values, whereas for other attributes each has 30 distinct values. You have only a 1 GB main memory laptop. Outline an efficient method that constructs decision trees efficiently, and answer the following questions explicitly: (1) how many scans of the database does your algorithm take if the maximal depth of decision tree derived is 5? (2) what is the maximum memory space your algorithm will use in your tree induction?

Answer:

Outline algorithm: RAINFOREST. Build an AVC list (attribute-value-class label). Scan data once, one can populate the AVC group, and then calculate to decide which attribute needed to used for split.

5 scans in most cases since each scan construct one level of the tree.

For the first scan, one needs memory space as follows:

One AVC group: 24 * 30 * 4 * 2 = 5.76 K bytes.

But for the 2nd scan, there will 30 such tables, and in the worst case, there will be 30 such AVC group.

3rd scan, there will be 30^2 such AVC group

4th scan, there will be 30^3 such AVC group.

5th scan, there will be 30^4 such AVC group, each of size 20 * 30 * 4 * 2 = 4800 bytes = 4.8K. Notice 81 * 10^4 * 4.8 K = 4GB in the worst case. But since most of the table will be empty for such values, in most cases, it will still fit into the memory. Thus in practice, it still be ok to do it in 5 scans.

□

5. [26] Clustering

(a) [8] Choose the best clustering algorithm for the following tasks (and reason on your choice using one sentence):

(1) clustering You-Tube videos based on their captions,

Answer:

Consider caption as a set (vector) of keywords, once can use frequent-pattern based clustering, or subspace clustering.

There could be other reasonable answers such as EM algorithm.

□

(2) clustering houses to find delivery centers in a city with rivers and bridges,

Answer:

clustering with obstacles as constraints (COD) because distance calculation should consider obstacles.

□

(3) distinguishing snakes hidden in the surrounding grass, and

Answer:

density-based clustering (DBSCAN, or OPTICS) because snakes can only be identified based on the connected similar colored points. □

(4) clustering shoppers based on their shopping time, the amount of money spent, and the categories of goods they usually buy.

Answer:

numerical data using k-means, categorical data using k-modes, and the combined one using combined distance measure with an extended k-means algorithm (called k-prototype).

Other answers: EM.

□

(b) [6] Explain why BIRCH can handle large amount of data in clustering, and explain how such a methodology can be used to scale up SVM classification in large data sets.

Answer:

BIRCH: balance-tree, CF-tree (using 0, 1, 2 moment to compute the differences and organize data into microclusters and hierarchies. On top of such a hierarchy, one can use any good clustering algorithm. Thus it can handle large data sets.

CB-SVM (Clustering-based SVM)

exploring Birch-like hierarchical structures, construct such a structure for positive and negative data respectively using BIRCH.

Training SVM from top down

subclusters along the support vector can be declustered to refine the support vector by repeatedly train with SVM.

□

(c) [6] What are the major difficulty to cluster a micro-array data set? Outline one efficient and effective method to cluster a micro-array data set.

Answer:

Major difficulty: High-dimensionality, data is sparse, distance is not meaningful in such high-D space, cluster may not exist in the whole space.

Method: p-clustering algorithm (such concise description should be fine).

Notice CLIQUE may not be able to handle such a high dimensional space. Thus need to take some points off.

□

(d) [6] Suppose a university database has multiple, interconnected relations: *Professor*, *Department*, *Student*, *Course*, and *Publications*. Outline an effective algorithm that may cluster *Professors* according to user's preference, *e.g.*, based on the research performance of the professors.

Answer:

CrossClus algorithm:

1. Start from the user-specified features

2. search in neighbor of existing pertinent features (*i.e.*, Calculate the links and identify more relevant and weighted features by link analysis)

3. Expand the search range gradually using TID propgation to create multirelational feature IDs

4. Apply such reasonable clustering algorithm, such as k-means, etc.

□

1.4 Sample Exam Question Set: 1.4

1.4.1 Midterm Exam

1. [35] Data and data preprocessing.

   (a) [6] For data visualization, there are three classes of techniques: (i) geometric techniques, (ii) hierarchical techniques, and (iii) icon-based techniques. Give names of two methods for each of these techniques.

   Answer:

   i. geometric techniques: scatter plots, landscapes, (or anything correct)

   ii. hierarchical techniques: dimension stacking, cone trees, tree maps, (or anything correct)

   iii. icon-based techniques: stick figures, Chernoff faces, (or anything correct)

   □

   (b) [6] What are the value ranges of the following measures, respectively?

   i. χ^2: **Answer:** $[0, \infty]$ □

   ii. Jaccard coefficient: **Answer:** $[0, 1]$ □

   iii. *covariance*: **Answer:** $[-\infty, +\infty]$ □

   (c) [8] Name four methods that perform effective *dimensionality reduction* and four methods that perform effective *numerosity reduction.*

   Answer:

   i. *dimensionality reduction*: principal component analysis, feature selection, wavelet transform, etc.

   ii. *numerosity reduction*: regression, clustering, sampling, histogram, etc.

   □

   (d) [6] What are the best distance measure for each of the following applications:

   (i) compare similar diseases with a set of medical tests

   Answer: Jaccard coefficient □

   (ii) find whether two text documents are similar

   Answer: cosine similarity □

   (iii) the maximum difference between any attribute of two vectors

   Answer: supremum distance, i.e., L_∞ norm □

   (e) [9] For the following group of data

   $$100, 400, 1000, 500, 2000$$

   i. Calculate its mean and variance.

   Answer:

A. mean: $1/5(\Sigma ...) = 800$

B. variance: $1/5(\Sigma diff\_square) = 444000$

□

ii. Normalize the above group of data by min-max normalization with min = 1 and max = 10; and

Answer:

A. 100 → 1

B. 400 → 2.42

C. 500 → 2.89

D. 1000 → 5.26

E. 2000 → 10

□

iii. In z-score normalization, what value should the first number 100 be transformed to?

Answer:

A. $\sigma = \sqrt{variance} = 666.33$

B. 100 → $(100 - 800)/\sigma = -1.05$

□

2. [17] Data Warehousing and OLAP for Data Mining

(a) [7] Suppose a base cuboid has D dimensions but contains only p (where $p > 1$) nonempty cells

(i) *how many cuboids* does this cube contain (including base and apex cuboids)?

Answer:

i. 2^D

□

(ii) what is the *maximum number of nonempty cells possible* in such a materialized cube?

Answer:

i. $p \times 2^D - p + 1$

□

(b) [5] Suppose a WalMart data cube takes sum, mean, and standard deviation to measure the sales of its commodities. Explain how the three measures of the cube can be incrementally updated, when a new batch of base data set D is added in.

Hint: The **standard deviation** of n observations $x_1, x_2, \ldots, x_n$ is defined as

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} = \sqrt{\frac{1}{n}[\sum x_i^2 - \frac{1}{n}(\sum x_i)^2]}. \quad (1.9)$$

where $\bar{x}$ is the average (*i.e.*, mean) value of $x_1, \ldots, x_n$.

Answer:

i. We need to have count (n), sum ($\sum_{i=1}^{n} x_i$), and square sum ($\sum_{i=1}^{n} x_i^2$) stored.
ii. When new batch data is added in, we just need to add these three values incrementally, in multidimensional space. Then everything will be updated.

□

(c) [5] Suppose a disk-based large relation contains 30 attributes. What is the minimum number of database scans in order to derive a generalized relation by *attribute-oriented induction*?

Answer:

i. 2 scans or $1 + \delta$ scan.
ii. Explanation: The first scan calculate the number of distinct values in each attribute and determine whether to remove, retain, or generalize, and how high to generalize. (This process can be done by a δ scan. The second scan really generalize for all the attributes at the same time.

□

3. [22] Data cube technology

(a) [10] Given the following four methods: *multiway array cubing* (Zhao, et al. SIGMOD'1997), *BUC* (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD'2001), *StarCubing* (Xin et al., VLDB'2003), and *shell-fragment* approach (Li et al, VLDB'2004), list one method which is the best and another which is the worst (or not working) to implement one of the following:

(a) computing a dense full cube of low dimensionality (e.g., less than 6 dimensions),

Answer:

i. Best: Multiway array cubing
ii. Worst: BUC (but if student answer Shell-fragment, we can give them full point]

□

(b) computing a large iceberg cube of around 8 dimensions, and

Answer:

i. Best: StarCubing or BUC
ii. Worst: Multiway array cubing

□

(c) performing OLAP operations in a high-dimensional database (e.g., over 60 dimensions).

Answer:

i. Best: Shell Fragment

ii. Worst: Any of the other three: StarCubing, BUC, Multiway array cubing

□

(b) [6] Suppose a (sampling) survey data sets contains 100 dimensions (variables), but people would still like to perform multidimensional drilling into the cells containing no or few data and examine the statistics (measures) of the cell. Outline a method that may implement such a mechanism effectively.

Answer:

i. Build a shell fragment cube

ii. When drilling, using either intra-cuboid expansion or inter-cuboid expansion to combine other cells whose attributes have low correlation with the dimension interested (such as interesting dimension: salary, and other dimension: telephone number)

□

(c) [6] Suppose one would like to implement a web-based search engine to return top-k best deals for used cars with user selected multidimensional features, such as model, year, price-range, etc., where the best deal is a user-defined function of book-price and sales-price. Outline the design of a cube structure to support such a search engine.

Answer:

i. Use ranking-cube to support it

ii. Partition data on ranking dimensions and group data by selection dimensions

iii. Given a query (and dynamic ranking function), locate the blocks that likely have the top scores, and progressively search the next best block, until enough answer is found.

□

4. [23] Frequent pattern and association mining.

(a) [10] A database with 100 transactions has its FP-tree shown in Fig. 1.5. Let $min\_sup = 0.5$ and $min\_conf = 0.8$. Show

Figure 1.5: FP tree of a transaction DB

i. c's conditional (i.e., projected) database:

Answer:

A. $eab : 35$,

B. $ea : 30$,

C. $eb : 5$

□

ii. all the frequent k-itemsets for the largest k:

Answer:

A. $k = 3$. All frequent itemsets: $eac : 65/100 = 0.65$

□

iii. two strong association rules (with support and confidence) containing the k items (for the largest k only):

Answer:

A. $ea \rightarrow c$ (sup: 0.65, conf: $65/80 = 0.81$)

B. $c \rightarrow ea$ (sup: 0.65, conf: $65/70 = 0.94$)

□

(b) [6] Briefly describe one efficient **distributed** pattern growth mining method which can mine enterprise-wide (*i.e.*, global) frequent itemsets for a chain store like Sears, without shipping data to one site.

Answer:

i. Suppose there are k partitions

ii. Pattern-growth on each partition and then merge frequent itemsets as candidate FPs,

iii. One more scan of each partitioned DB to count the merged frequent itemset candidates.

□

(c) [7] It is important to use a good measure to check whether two items in a large transaction dataset are strongly correlated.

(i) Give one example to show that *lift* may not be a good measure for such a purpose.

Answer:

i. Lift is not null-invariant, thus it is not good if the number of null transaction varies a lot

ii. Ex. If m and c has the following distributions: $mc = 10$, $m\neg c = 100$, $\neg m, c = 100$, $\neg m \neg c = 10000$, lift will be a big value! (Any other similar example will be OK.)

□

(ii) Give a good measure for this purpose and reason why it is a good measure.

Answer:

i. Kulc is a good measure, since it is null invariant.

ii. Other answers, such as cosine, all-confidence are also OK.

□

5. [3] (Opinion).

(a) I □ like □ dislike the exams in this style.

(b) In general, the exam questions are □ too hard □ too easy □ just right.

(c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

1.4.2 Final Exam

1. [15] Data preprocessing.

(a) [7] Data integration is essential in many applications. Suppose we are given a large data relation, Student, with a lot of tuples, and with attributes: (Student_Name, Major, University, Status, Office_Address). Present one effective method that can discover a set of different strings, such as "UIUC", and "University of Illinois at Urbana Champaign", essentially represent the same entity.

Answer:

We can use the **entity integration** technique to pre-process the data. There are relations among the attributes, *e.g.*, we can infer the university of "UIUC" and "U of I" are the same entity if the office address of those tuples are close/identicial.

This is an open problem, the key point of this problem is to infer the entity by the relations of the attributes. Any reasonable answers will get full points.

□

(b) [8] To compute some graph plots in multidimensional space, we often need to judge if a measure is algebraic, distributive or holistic. Judge which category each of the following measures (or set of measures) belongs to and explain your judgment briefly.

(1) Boxplot

Answer:

Holistic. To compute the boxplot graph, one has to compute the measures of the minimum, lower quartile (Q1), median (Q2), upper quartile (Q3) and maximum. Q1, Q2 and Q3 are holistic measures. Thus the boxplot is also holistic.

□

(2) Bottom-10 (among all the objects in the corresponding k-dimensional space)

Answer:

Distributive. One can divide the whole dataset to several partitions, and collect the bottom-10 for each partition. The bottom-10 of the whole dataset can be computed by listing all the collections in ascending order and select the top 10 item. The result calculated from the partitions are the same as the one from entire dataset. Thus the bottom-10 measure is distributive.

We can also seen the computation process as a formula, and in this perspective the measure is also algebraic. The students will get full points of the answer of "algebraic" if they give reasonable explanations. □

(3) linear regression line

Answer:
Algebraic. The linear regression line can be computed by the least square method. In a 2-D space, the regression line $y = mx + b$ can be computed as:
Find the four sums: $\sum x$, $\sum x^2$, $\sum y$, and $\sum xy$.
The calculations for the slope m and the y-intercept b are as follows.
$\hat{m} = \frac{n(\sum xy)-(\sum y)(\sum x)}{n(\sum x^2)-(\sum x)^2}$;
$\hat{b} = (\frac{1}{n}\sum y) - \hat{m}(\frac{1}{n}\sum x) = \frac{(\sum y)(\sum x^2)-(\sum x)(\sum xy)}{n(\sum x^2)-(\sum x)^2}$
The students are not expected to provide those details, they can get full points if they answer algebraic with brief explanations.

□

(4) confidence interval in the formula $\bar{x} \pm t_c\hat{\sigma_x}$
Answer:
Algebraic. The mean value $\bar{x}$ and the standard deviation $\hat{\sigma_x}$ are both algebraic. Therefore the formula is also algebraic. □

2. [15] Data Warehousing, OLAP and Data Cube Computation

   (a) [8] Assume a base cuboid of 10 dimensions contains only four base cells: (i) $(a_1, a_2, a_3, a_4, \ldots, a_{10})$, (ii) $(b_1, b_2, b_3, a_4, \ldots, a_{10})$, (iii) $(c_1, c_2, a_3, a_4, \ldots, a_{10})$, and (iv) $(d_1, a_2, a_3, a_4, \ldots, a_{10})$, where no pair of these constants are equal. The measure of the cube is *count*.

   (1) How many *nonempty* aggregate (i.e., nonbase) cells will a full cube contain?
   Answer:
   Total num of cells: $4 * 2^{10}$
   Count 4 cells (overlap 3 times): 2^7 (with the prefix of (*,*,*));
   Count 3 cells (overlap 2 times): 2^7 (with the prefix of (*,*,a_3));
   Count 2 cells (overlap 1 times): $2 * 2^7$ (with the prefix of (*, a_2, a_3) and (*, a_2, *));
   base cells: 4
   Thus, the nonempty aggregated cells are:
   TotalNum - 3*(count 4 cells)- 2*(count 3 cells)- (count 2 cells) - base cells
   $= 4 * 2^{10}$ - $3 * 2^7$ - $2 * 2^7$ - $2 * 2^7$ - 4
   $= 25 * 2^7$ - 4 = 3200 -4 = 3196
   The students who answer "$4 * 2^{10}$ - $3 * 2^7$ - $2 * 2^7$ - $2 * 2^7$ - 4" or "$25 * 2^7$ - 4" can get full points. Those whose result is not right but the process is right can get at least half of the total points.

   □

   (2) How many *nonempty* aggregate cells will an iceberg cube contain if the condition of the iceberg cube is "*count* ≥ 2"?
   Answer:
   Count 4 cells: 2^7 (with the prefix of (*,*,*));

Count 3 cells: 2^7 (with the prefix of (*,*,a_3));
Count 2 cells: $2 * 2^7$ (with the prefix of (*, a_2, a_3) and (*, a_2, *));
Thus there are $4 * 2^7$=512 cells. □

(3) A *closed cube* is a data cube consisting of only closed cells. How many closed cells are in the full cube?

Answer:
There are 7 closed cells.
Base Cells:
$(a_1, a_2, a_3, a_4, \ldots, a_{10})$
$(b_1, b_2, b_3, a_4, \ldots, a_{10})$
$(c_1, c_2, a_3, a_4, \ldots, a_{10})$
$(d_1, a_2, a_3, a_4, \ldots, a_{10})$
Count 2 Cell:
$(*, a_2, a_3, a_4, \ldots, a_{10})$
Count 3 Cell:
$(*, *, a_3, a_4, \ldots, a_{10})$
Count 4 Cell:
$(*, *, *, *, \ldots, a_{10})$

□

(b) [7] Databases are usually used to answer people's queries. When the expected answer set is large, it is often desirable to return only a small set of top-ranked answers. Suppose a user would like to get top-k ranked answers for Thankgiving online shopping, based on his/her own criteria of ranking. But the relevant dimension is pretty high (say over 30 dimensions). Design a data cube that may facilitate efficient processing of such queries.

Answer:

The data cube should be a **ranking cube** with **shell fragments** technology to support top k query in high dimensional space.

The key points of ranking cube are:

1). Partition data on both selection and ranking dimensions;

2). Compute measures for each group/block;

3). Simultaneously push selection and ranking conditions to locate the block with top score.

To support high-dimensional data:

1). Materialize only those atomic cuboids that contain single selection dimensions;

3) Uses the idea similar to high-dimensional OLAP: Achieves low space overhead and high performance in answering ranking queries with a high number of selection dimensions.

If the students describe the ranking cube and shell fragments techniques in their own language, as long as the description is reasonable and valid, they will get full points.

□

3. [20] **Frequent pattern and association mining**

(a) [8] Suppose a WalMart manager is interested in only the *frequent patterns* (i.e., *itemsets*) that satisfy certain constraints. For the following cases, state the characteristics (*i.e.*, categories) of *every constraint* in each case and how to mine such patterns most efficiently.

i. The price difference between the most expensive item and the cheapest one in each pattern must be within $20.

Answer:

This constraint is the type of $range(s) \leq V$, it is an anti-monotonic constraint. We can use the **Constrained FP-Growth** or **Constrained Apriori**algorithm to mine the patterns efficiently.

If the students give the right answer of the constraint type and describe the algorithms in detail. They can get full points if the answers are reasonable. □

ii. The sum of the price of all the items with profit over $10 in each pattern is at least $200.

Answer:

This constraint is monotonic (It is also data anti-monotonic/convertable anti-monotonic). We can list the items by the price in descending order, and use the **Constrained FP-Growth** to mine the frequent patterns. In the case that for a certain branch in FP-tree, if the profit sum for all the items over $10 is less than $200, the algorithm can stop processing and prune the entire branch.

Any other reasonable answers will get full/some points. □

iii. The average profit for those items priced over $50 in each pattern must be less than $10.

Answer:

This constraint is convertible anti-monotonic if the items are listed by price in descending order and then profit in ascending order.We can use the **Constrained FP-Growth** to mine the frequent patterns. In the case that for a certain branch in FP-tree, if the average profit for all the items over $50 is less than $10, the algorithm can stop processing and prune the entire branch.

Any other reasonable answers will get full/some points. □

(b) [6] Explain why both *Apriori* and *FPgrowth* algorithms may encounter difficulties at mining colossal patterns (*i.e.*, the patterns of large size, *e.g.*, 100). Outline a method that may mine such patterns efficiently. Will such a mining method generate all the colossal patterns?

Answer: The reason is that the size of the mid-sized patterns is explosive, there is no hope to find colossal patterns efficiently by insisting complete set mining philosophy.

Pattern Fusion is such a method. The key points are:

1). Initialization: Use an existing algorithm to mine all frequent patterns up to a small size, e.g., 3.

2). Iterative Pattern Fusion: At each iteration, k seed patterns are randomly picked from the current pattern pool; For each seed pattern thus picked, we find all the patterns within a bounding ball centered at the seed pattern; All these patterns found are fused together to generate a set of super-patterns; All the super-patterns thus generated form a new pool for the next iteration.

3) Termination: when the current pool contains no more than k patterns at the beginning of an iteration.

Theoretically, there are risks for pattern fusion to miss some colossal patterns. However in real applications the accuracy of pattern fusion is very high, it seldom misses any patterns. □

(c) [6] Frequent pattern mining often generates too many patterns. Outline two efficient methods that may generate less number but only interesting patterns.

Answer:

1. Mining the **compressed patterns** with **RPglobal** algorithm. The key points are:

1). the pattern compressed problem can be defined as to find a minimal set of representative patterns to cover each frequent pattern. Here "cover" means the distance (of support) of a frequent pattern to a representative pattern is within a threshold;

2). the RPglobal algorithm is a greedy algorithm. It first collects information over all the frequent patterns and their representative patterns, then generates the top k representative patterns in a descending order of the covered set size one by one.

2.Mining the **representative patterns**.

1). the representative pattern mining algorithm is similar to compressed pattern mining. By the definition of the compression is adjusted from the typical measures. A frequent pattern will only be covered by another if the number of different items they contain is within a threshold, rather than the support. And the support of the representative pattern should be larger than the pattern covered.

2). We can use a similar algorithm to find the representative patterns in the sense that more distinct patterns will be more preferable than very similar patterns. The process of the algorithm is: i) define significance of the pattern; ii) estimate redundancy of two patterns; iii) construct a redundancy graph; iv) find a maximal spanning tree with k patterns.v) the nodes in the tree will be output as the top k representative patterns.

Any other reasonable answers, e.g., max pattern, closed pattern, will get full/some points. □

4. [26] **Classification and Prediction**

(a) [5] What are the major differences among the three: (1) Naïve-Bayesian algorithm, (2) Bayesian Belief Network, and (3) Neural Network?

Answer:

Naive-Bayesian algorithm:

i. Assumption: class conditional independence. Therefore, Loss of accuracy due to the assumption of independence;
ii. Simple and easy to implement, most efficient in classification

Bayesian Belief Network

i. Allow class conditional independencies between subsets of variables. Therefore more accurate in classification
ii. Computational intensive

Neural Network:

i. Discriminative classification method with high accuracy. Classification by back-propagation.
ii. Robust, work well even when the training set contains error. Wide applicability.
iii. Long training time, Poor interpretability: difficult to understand the learned functions

Rubrics: As long as the most important points are answered, we will give the full points. □

(b) [5] All the following three methods may generate rules for induction: (1) *decision-tree induction*, (2) *sequential covering rule induction*, and (3) *classification based on association (CBA)*. Explain what are the major differences among them. In a typical dataset, which one generates the most number of rules and which one generates the least?

Answer:

Decision tree induction: Rules are extracted from the decision tree: One rule is created for each path from the root to a leaf; each attribute-value pair along a path forms a conjunction: the leaf holds the class prediction; Rules are mutually exclusive and exhaustive. The rules are generated simultaneously.

Sequential covering rule induction: Rules are learned sequentially, each for a given class C_i will cover many tuples of C_i but none (or few) of the tuples of other classes. Rules are learned one at a time.

Classification based on association: Mine possible association rules in the form of: Condition-set (a set of attribute-value pairs) →

class label; Build classifier: Organize rules according to decreasing precedence based on confidence and then support

Most: CBA **Least**: Sequential covering

□

(c) [6] Given a training set of 10 million tuples with 40 attributes each taking 4 bytes space plus one class label attribute. The class label attribute has four distinct values, whereas for other attributes each has 20 distinct values.

Some decision tree induction method uses the measure, Gini index, to measure the impurity of D, a data partition or set of training tuples, as

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2, \tag{1.10}$$

where p_i is the probability that a tuple in D belongs to class C_i and is estimated by $|C_{i,D}|/|D|$.

Outline an efficient method that constructs such decision-tree classifier *efficiently*, and answer the following questions explicitly: (i) how many scans of the database does your algorithm take? (ii) what is the maximum memory space your algorithm will use in your induction in each scan?

Answer:

i. The algorithm forms the AVC list in the first scan, and then for each subsequent level, 1 can is required. So the number of total scans the algorithm takes is the height of the tree, h, whose maximum value is 40, the number of dimensions;

ii. The maximum memory space consumed is on the root node: $40 * 20 * 4 * 4 = 12800$ bytes

Rubrics: Some students used BOAT, not RainForest. We can give partial scores, say around 4, based on the quality of the answers.

□

(d) [5] What are the major differences among the three methods for the evaluation of the accuracy of a classifier : (1) *hold-out method*, (2) *cross-validation*, and (3) *boostrap*?

Answer:

Hold-out method: Given data is randomly partitioned into two independent sets: Training set (e.g., 2/3) for model construction; Test set (e.g., 1/3) for accuracy estimation

Cross-validation: Randomly partition the data into k mutually exclusive subsets, each approximately equal size; At i-th iteration, use D_i as test set and others as training set

Bootstrap: Works well with small data sets; Samples the given training tuples uniformly with replacement i.e., each time a tuple is selected, it is equally likely to be selected again and re-added to the training set

□

(e) [5] Give each situation that one of the following measure is most appropriate for measuring the quality of classification: (1) *accuracy*, (2) *F-measure*, and (3) *ROC curve*.

Answer:

Accuracy: Suitable for general classification tasks when both positive and negative examples have equal importance

F-measure: harmonic mean of precision and recall, suitable for situations that both precision and recall are important

ROC curve: suitable when true positive rate and false positive rate are both important.

□

5. [24] **Clustering**

(a) [12] Outline the best clustering method for the following tasks (and briefly reason on why you make such a design):

(i) clustering a set of research papers based on their authors and their publication venues

Answer:

RankClus. □

(ii) clustering a set of videos based on their image contents, captions, and where they reside on the web

Answer:

p-clustering (Frequent pattern based) or any others helpful for high dimensional data □

(iii) clustering UPS (or FeDex) customers for package delivery to minimize total transportation cost and have relatively even work load for each delivery employee

Answer:

constraint-based clustering considering obstacles. □

(iv) taking user's expectation expressed as a set of preferences, group students based on their academic records, research interests, and publication records

Answer:

CrossClus. □

(b) [6] Why subspace clustering is a good choice for high-dimensional data? Outline one efficient and effective subspace clustering method that can cluster a high-dimensional data set.

Answer:

Reason: The curse of dimensionality. Clusters may exist only in some subspaces

Subspace-clustering: find clusters in all the possible subspaces, CLIQUE, ProClus, and frequent pattern-based clustering methods will be fine. □

(c) [6] Why is it that BIRCH encounters difficulties to find clusters of arbitrary shape but OPTICS has no problem to do it? Propose some modifications to BIRCH so that it can help find clusters of arbitrary shape.

Answer:

BIRCH uses Euclidean distance to cluster objects in the hierarchical CF-Tree. OPTICS uses density-based method to find maximal set that are density-connected. Therefore, OPTICS can find objects with arbitrary shape, while BIRCH cannot.

BIRCH needs to embed some distance measure related to density. In this way, data objects that are density-connected can be detected by BIRCH. □

1.5 Sample Exam Question Set: 1.5

1.5.1 Midterm Exam

1. [28] Data preprocessing.

   (a) [5] It is not straightforward to visualize k-dimensional data for $k > 3$. Name 5 visualization techniques that can visualize 6-dimensional data effectively.

   Answer:

   Most of the visualization methods, such as stick figure, Chernoff face, dimension stacking, parallel coordinates, multi-dimensional scatter plot, etc.

   □

   (b) [6] For each of the following similarity measures, give one good application example.

   i. Cosine measure
   Answer: measuring text similarity. □

   ii. Jaccard coefficient
   Answer: computing similarity of a set of medical tests. □

   iii. Minkowski distance for $k = 1$
   Answer: computing Manhattan (city-block) distance □

   (c) [9] Distinguishing the following concepts or measures

   i. Pearson correlation coefficient vs. covariance
   Answer: Pearson correlation coefficient (X, Y) = covariance (X, Y) / $\sigma_x \cdot \sigma_Y$ □

   ii. Principal component analysis vs. feature selection
   Answer: Note: Answer could vary as long as has some idea similar to this: PCA is feature transformation. It projects from D dimension to m dimension, where $m < D$ while preserving the maximal variance of samples in lower dimensional space. Feature selection is to select the best subset of features, and it does not involve feature combination. □

   iii. Fourier transform vs. wavelet transform
   Answer: Note: Answer could vary as long as has some idea similar to this: FT: transforms from time to frequency space, and the first several terms preserves most of the energy. WT: splits data into mean (smooth) and differences in different scales, preserves shapes. □

   (d) [8] For the following group of data

   $$500, 300, 100, -100$$

   i. Calculate its mean and variance.
   Answer: mean: 200, variance: 50,000 □

ii. Normalize the above group of data by min-max normalization with min = -1 and max = 1; and
Answer: 1, 1/3, -1/3, -1 □

iii. In z-score normalization, what should the value of 400 be transformed to?
Answer: $\frac{400-200}{\sqrt{50,000}} = .89$ □

2. [25] Data Warehousing and OLAP for Data Mining

(a) [10] Suppose the base cuboid of a data cube contains only two cells

$$(a_1, a_2, a_3, \ldots, a_{10}), (b_1, b_2, b_3, \ldots, b_{10}),$$

where $a_i = b_i$ if i is an odd number; otherwise $a_i \neq b_i$.

i. How many nonempty aggregate (*i.e.*, non-base) cells are there in this data cube?
Answer: $2 \times 2^{10} - 2^5 - 2$ □

ii. How many nonempty, *closed* aggregate cells are there in this data cube?
Answer: 3: $(a_1, a_2, a_3, \ldots, a_{10}) : 1, (b_1, b_2, b_3, \ldots, b_{10}) : 1, (a_1, *, a_3, *, \ldots, a_9, *) :$ 2. □

iii. If we set minimum support = 2, how many nonempty aggregate cells are there in the corresponding iceberg cube?
Answer: $(a_1, *, a_3, *, \ldots, a_9, *) : 2$, and its further generalizations, so in total 2^5. □

(b) [10] Suppose a market shopping data warehouse consists of four dimensions: *customer, date, product*, and *store*, and two measures: *count*, and *avg_sales*, where *avg_sales* stores the real sales in dollar at the lowest level but the corresponding average sales at other levels.

i. [5] Draw a **star schema** diagram (sketch it, do not have to mark every possible level, and make your implicit assumptions on the levels of a dimension when you draw it).
Answer: Assume most can draw this well. □

ii. [5] If one also wants to compute standard deviation as a measure, what other intermediate measures need to be introduced to make the computation efficient?
Answer: Since $\sigma^2 = 1/n\Sigma_i^n(x_i^2) - (\Sigma_i^n(x_i)/n)^2$, we need to introduce sum of square, sum (and count) in order to compute standard deviation efficiently. □

(c) [5] Bitmap index is often used for accessing a materialized data cube. If a cuboid has 6 dimensions, each has 10 distinct values, and it has in total 3000 cells. How many bit vectors should this cuboid have? How long each bit vector should be (assuming no sophisticated compression techniques are explored)?

Answer:

Each value of each dimension has its own bit vector. So we will need 6*10 = 60 bit vectors.

Each bit vector should have 3000 bits long (for 3000 cells).

□

3. [20] Data cube implementation

   (a) [5] Suppose *incremental update of a data cube* means that new data can be incrementally inserted into the base cuboid without recomputing the whole cube from scratch. Can you do this for an *iceberg cube*? If you can, state how; but if you cannot, state why not.

   Answer: No. Because iceberg cube drops the count of the cells if it is below min_sup, it cannot get the correct count for cells in an iceberg cube upon incremental update. □

   (b) [5] Explain why the data cube could be quite sparse, and how one should implement such a sparse cube if one adopts an array-cube implementation.

   Answer: Sparse since in most cases, the possible dimensional combination is huge but real data will not show up in most possible space, e.g., one cannot take all the courses in every semester, or one cannot buy most of the possible Walmart merchandizes in every transaction.

   Sparse array compression: Use chuck to partition the data, and use (chunk_id, offset) to store only those cells contain (nonempty) values. □

   (c) [5] Given the following four methods: *multiway array cubing* (Zhao, et al. SIGMOD'1997), *BUC* (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD'2001), *StarCubing* (Xin et al., VLDB'2003), and *shell-fragment* approach (Li et al, VLDB'2004), list one method which is the best and another which is the worst (or not working) to implement one of the following:

      (a) computing a dense *iceberg cube* of low dimensionality (e.g., less than 6 dimensions),
      Answer: Best: star-cubing or BUC
      Worst: multiway-array □

      (b) performing OLAP operations in a high-dimensional database (e.g., over 50 dimensions).
      Answer: Best: Shell-fragment
      Worst: Any of the others since they cannot do it. □

   (d) [5] Most data cubes support OLAP operations on the whole population of data, but people may also like to support OLAP operations for sampling data. What are the major challenges to support OLAP on sampling data? Outline a method that may support such operations effectively.

Answer: Major challenges: Some drilling down cells may contain few or no data.

Method: Intra or inter-cuboid expansion to combine with other cells whose attributes has low correlations with the dimensions interested. □

4. [24] Frequent pattern and association mining.

(a) [8] A database has 5 transactions. Let $min\_sup = 0.6$ and $min\_conf = 0.8$.

customer	date	items_bought
100	10/15	{I, P, A, D, B, C}
200	10/15	{D, A, E, F}
300	10/16	{C, D, B, E }
400	10/18	{B, A, C, K, D}
500	10/19	{A, G, T, C}

i. List the frequent k-itemset for the largest k, and
Answer: $k = 3, BCD : 3$. □

ii. all the strong association rules (with support and confidence) for the following shape of rules:
$\forall x \in transaction,\ buys(x, item_1) \wedge buys(x, item_2) \Rightarrow buys(x, item_3)$. $[s, c]$
Answer:

$buys(x, B) \wedge buys(x, C) \Rightarrow buys(x, D)$. $[.6, 100\%]$
$buys(x, B) \wedge buys(x, B) \Rightarrow buys(x, C)$. $[.6, 100\%]$
$buys(x, C) \wedge buys(x, D) \Rightarrow buys(x, B)$. $[.6, 100\%]$

□

(b) [10] A research publication database like DBLP contains millions of papers authored by a set of researchers.

(i) [5] Using this dataset, explain why *null-invariance* property is important in the study which authors are “correlated”.

Answer: Since most authors are not coauthoring papers, null value is very high. Null-invariance is critical otherwise the ”correlation value could be greatly influenced by null values. □

(ii) [5] To mine potential advisors and advisees relationships, what measures would you like to use? Explain your answer.

Answer: Kulcizakii value and imbalance factor (or something like that). □

(c) [6] Apriori is an efficient method for mining frequent patterns. Briefly describe how to extend this method to mine frequent-itemset **incrementally**, i.e., incorporate newly added transactions without redoing the mining from scratch.

Answer: Consider $DB + \delta DB$ and assume we know frequent itemsets (FP) in DB already. Mine FP_δ in δDB, taken union of FPs and scan DB for those frequent only in δDB and scan δDB for those only in DB and merge their counts. □

5. [3] (Opinion).

   (a) I □ like □ dislike the exams in this style.

   (b) In general, the exam questions are □ too hard □ too easy □ just right.

   (c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

1.5.2 Final Exam

1. [14] Data preprocessing.

   (a) [8] Present the value range for each of the following measures:
   (1) Jaccard coefficient
   Answer: $[0, 1]$ □
   (2) covariance
   Answer: $(-\infty, +\infty)$ □
   (3) F-measure
   Answer: $[0, 1]$ □
   (4) Kulczynski measure
   Answer: $[0, 1]$ □

   (b) [6] Give three example distance measures for each of the following two kinds:
   (1) the distance *between two objects*
   Answer: any three of the following: Euclidean distance, Manhattan distance, Supremum distance (*i.e.*, L_∞ *norm*), cosine distance, Minkowski distance, ... □
   (2) the distance *between two clusters*
   Answer: any three of the following: single-link, complete link, average link, distance between cluster centroids, distance between cluster medoids □

2. [16] Data Warehousing, OLAP and Data Cube Computation

   (a) [8] The **standard deviation** of n observations $x_1, x_2, \ldots, x_n$ is defined as

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} = \sqrt{\frac{1}{n}[\sum x_i^2 - \frac{1}{n}(\sum x_i)^2]}. \quad (1.11)$$

   where $\bar{x}$ is the average (*i.e.*, mean) value of $x_1, \ldots, x_n$.

   i. [3] What kind of measure does standard deviation belong to: *distributive, algebraic*, or *holistic*? Justify your answer.
   Answer: Algebraic since the measure can be computed based on three distributive measures: n, $\sum x_i^2$ and $\sum x_i$. □

   ii. [5] Outline an *efficient* algorithm that computes an *iceberg cube* with standard deviation as the measure, where the iceberg condition is $n \geq 100$ and $\sigma \geq 2$.
   Answer:
   From the above, standard deviation can be computed with three distributive measures: count, sum, and square-sum.
   We can use BUC (or Star-Cubing) to compute the iceberg cube and use count as pruning condition (iceberg condition).

Take BUC as an example, the cube is computed by taking dimension combinations, for each combination, check to see if the count of any cell is less than 100. If it is true, the cell will not be expanded further. Otherwise, keep expanding and when outputting cell's stdev, only those with $\sigma \geq 2$ will be output.
Note: We cannot use multiway cubing since we cannot prune the computing based on the iceberg condition $n \geq 100$. □

(b) [8] It is desirable to construct an AlbumCube to facilitate multidimensional search through digital photo collections, such as by date, photographer, location, theme, content, color, etc.

i. [2] What should be the dimensions and measures for such a data cube?
Answer: Dimension: date (hierarchy could be: date, day of week, month, year, etc.), photographer (hierarchy could be photographer, group, etc.), location (hierarchy could be loc, city, state, country, etc.), theme (hierarchy could be theme keyword, theme category).
Note: One can search by content or color, but more difficulty to set content/color as hierarchies or dimensions, special content-based image search is needed. You may want to take **0.5 point off** if they put these as dimension—many did it.
Measure: usually count is the most natural one. Space (sum of disk space in bytes) may also useful, then max, min, std-deviation in size may also make sense. □

ii. [3] What analytical functions can you provide?
Answer: Besides typical OLAP functions, one may consider clustering, classification, content summarization, those supporting similarity search.
Note: It is OK to give complete scores if most useful OLAP functions, such as drilling, slicing/dicing, etc. are answered □

iii. [3] What are the major challenges on implementing AlbumCube, and how would you propose to handle them?
Answer: High-dimensionality, especially when search and OLAP on contents, colors, etc. Solution: using high-D cubing and multi-dimensional indexing.
Content and color: hard to set as dimensions. Solution: developing similarity/approximate search on those aspects.
Noise: Solution: data cleaning, preprocessing.
Note: Major challenge: high-dimensionality (especially on contents and colors) that makes cube hard to build if putting these as dimensions). You may like to take **0.5-1 point off** if this is missing. □

3. [19] **Frequent pattern and association mining**

(a) [6] Since items have different expected frequencies of sales, it is desirable to use *group-based minimum support thresholds* set up by users. For example, one may set up a small *min_support* for the group of *cameras* but a rather large one for the group of *bread.* Outline an FPgrowth-like algorithm that derive the set of frequent items efficiently in a transaction database.

Answer: Suppose each item is associated with a group ID.

i. Scan the database and find single frequent items, F1, using *group-based minimum support thresholds*. Sort the F1 list based on item frequency descending order.
Note: It is also interesting and could be even smarter (but not required), to sort items based on how far the item from its support threshold: the bigger, the higher (to facilitate pruning).

ii. Construct FP-tree, and for each frequent item, project its conditional databases. Within the conditional database, find single frequent items as step 1, and construct its own FP-tree until either the pattern-base is empty or the FP-tree contains a single branch.

iii. Do the same as steps 1 and 2 recursively, and output the frequent itemsets so obtained.

□

(b) [7] Suppose a BestBuy analyst is interested in only the *frequent patterns* (i.e., *itemsets*) from the sales transactions that satisfy certain constraints. For the following cases, state the characteristics (*i.e.*, categories) of *every constraint* in each case and how to mine such patterns most efficiently.

i. The profit range for the items in each pattern must be within \$50.
Answer: The constraint $range(S.profit) \leq \$50$ is antimonotonic. Grow itemset set S in FP mining, If S violates the constraint, prune it (since S's superset can never satisfy the constraint).
Note: Do not take point off if no perfect constraint formula is written since it is not easy to write precise constraints in formulas. Focus will be on category of constraints and pruning methods. □

ii. The sum of the price of all the items with profit over \$5 in each pattern is at least \$100.
Answer: The constraint $sum(S.price \text{ if } S.profit > \$5) \geq \$100$ is monotonic. Grow itemset set S in FP mining, If S satisfies the constraint, no need to check the constraint any more (since S's superset always satisfies the constraint).
However, a better answer could be $sum(S.price \text{ if } S.profit > \$5) \geq \$100$ is data antimonotonic. Any transaction that cannot

satisfy this constain (together with the current pattern is pruned, which reduce the dataset to be search for this pattern in the future.
Note 1: Both answers are correct since the latter is more efficient, **one point off** if the latter part is missing.
Note 2: Do not take point off if no perfect constraint formula is written. Note there is only one constraint to be discussed, maybe **0.5 point off** if taken $I.profit > \$5$ as a constraint since we do not require every item in the pattern to be over \$5. □

iii. The average profit for those items priced over \$50 in each pattern must be less than \$10.
Answer: The constraint $avg(S.profit$ if $S.price > \$50) < \$10$ is convertible. Sort items whose price over \$50 in profit ascending order (to make the convertible one anti-monotonic). If S so far violates the constraint, prune it since S's superset can never satisfy the constraint—their average profit will be even higher.
Note 1: If sort in descending order, you cannot get the most efficient answer and thus you may take **0.5 point off**.
Note 2: Do not take point off if no perfect constraint formula is written. Note there is only one constraint to be discussed, maybe **0.5 point off** if taken $I.profit > \$5$ as a constraint since we do not require every item in the pattern to be over \$5. □

(c) [6] Frequent pattern mining often generates many somewhat "similar" patterns that carry little new information. Give one such example. Then outline one method that may generate less number (*i.e.*, compressed) but interesting patterns.

Answer: Example: If we have two patterns: ABCD: 2403, ABCDE: 2400. Then ABCDE should retain but ABCD is redundant and can be removed.

Method: Mining the **compressed patterns**. The key points are: The pattern compressed problem can be defined as to find a minimal set of representative patterns to cover each frequent pattern. Here "cover" means the distance (of support) of a frequent pattern to a representative pattern is within a threshold. E.g., using Jaccard coefficient as a definition.

Using a greedy algorithm, like RPglobal. It first collects information over all the frequent patterns and their representative patterns, then generates the top k representative patterns in a descending order of the covered set size one by one.

Note: Not too picky on the algorithm. Give full point if the general idea is right.

Note2: If people give a complete different idea, such as multi-level (low-level redundancy can be removed, give **60% points**. If people give closed/maximal itemset, give **50% points**. □

4. [27] **Classification and Prediction**

   (a) [6] Given a training set of 10 million tuples with 10 attributes each taking 8 bytes space. One attribute is a class label with two distinct values, whereas for other attributes each has 50 distinct values. Assume your machine cannot hold all the dataset in the main memory. Outline an efficient method that constructs Naïve Bayes classifier efficiently, and answer the following questions explicitly:

   (i) how many scans of the database does your algorithm take?

   Answer: One scan. Since by one scan, we can collect the AVC-sets, *i.e.*, (attribute, value, and class-label) set. Then Naïve Bayes classifier can be constructed based on this statistic. □

   (ii) what is the maximum memory space your algorithm will use in your induction?

   Answer: Maximum memory space: 1 page (suppose 4K) + space for AVC-sets. For each attribute, we need 8 bytes for attribute value, 8 bytes for storing counts for positive class-label, and 8 bytes for storing counts for negative class-label. So we need 24 bytes for each AVC. Since we have 50 distinct values, we need $50 \times 24 = 1.2K$ bytes. Since there are 10 attributes, we need 12K bytes. Thus in total we need 12K + one data page space.

   Note: Please note some may want to have 8 bytes to store attribute name, and then we will have 16K + one data page. Some may forget data page or some may have minor difference on computation, be lenient on those minor differences, *i.e.*, not have to take points off on it if you see they fully understand it. □

   (b) [6] Give each situation that one of the following measures is most appropriate for measuring the quality of classification:

   (1) *sensitivity*

   Answer: True positive rate. If you want your classifier to measure the true positive cases in a sensitive way. E.g., finding cancer cases. □

   (2) *specificity* **Answer:** True negative rate. If you want your classifier to measure the true negative cases in a subtle way. E.g., spam detection, you do not want to filter out useful message. □

   (3) *ROC curve*

   Answer: For a classifier that returns probability outputs. E.g., in skewed cases (e.g., few positive training data but lots of negative training data. You may like to measure a classifier using ROC curve instead of accuracy. □

   (c) [5] People say that if each classifier is better than random guess, ensemble of multiple such classifiers will lead to a nontrivial increase of classification accuracy. Do you agree with this statement? Give reasoning on it.

Answer: Agree. One reasoning could be if the err_rate of each classifier r is less than 0.5. For $2K$ classifiers, the accumulative error rate for k classifier will be r^k, which will be much smaller than a single r.

Another reasoning could be: Since the classifiers are training using different subsets or weights on the data, they are less likely to be trapped by the same noise or inaccuracy, and thus the majority voting are unlikely to be wrong. Moreover, by giving less weights to the classifiers that tend to make mistakes, the performance can be further improved.

Note: There could be similar arguments. If you find the reasoning is good, give them full points. □

(d) [5] What are the similarities and differences between *semi-supervised classification* and *active learning*?

Answer: Similarity: Both have a small set of labeled data and a large set of unlabeled data.

Differences: Semi-supervised classification does not interact with the expert. It labels the unlabeled data using classifiers training from labeled data (self-training/co-training)

Active learning: selective useful unlabeled examples and ask experts to give labels for them. □

(e) [5] If one would like to work out a model to classify U. of Michigan webpages based on the model you have learned from the UIUC website. Is it easy to do it by transfer learning? How would you suggest the person to proceed?

Answer: Not so easy if we just got UIUC website since the single website likely to get models which fits UIUC better. The better way is to train the classifier using more than one university (to not overfit to one university's dataset) and the classifier so training will be more adaptable to U. of Michigan. You can then use algorithm like TrAdaBoost to do it.

Note: For Answer Yes, and give method like TrAdaBoost, it should get **1.5** point off. □

5. [24] **Clustering**

(a) [6] Use one sentence to distinguish each of the following pairs of methods:

(1) *k-means* vs. *KNN*

Answer: k-means: A partitioning clustering algorithm

KNN: A lazy learning algorithm to find its k-nearest neighbor and return labels based on its k-nearest neighbors. □

(2) *STING* vs. *CLIQUE*

Answer: A grid-based clustering algorithm based on multi-resolution, hierarchical summarization

CLIQUE: A grid-based subspace clustering algorithm based on Apriroi principle. □

(3) *BIRCH* vs. *CHAMELEON*.

Answer: BIRCH: A micro-clustering-based clustering algorithm, incrementally build balanced CF-tree (clustering feature).

CHAMELEON: Use graph-partitioning to form small subclusters and then merge them based on inter-connectivity and closeness. □

(b) [6] Outline the best clustering method for the following tasks (and briefly reason on why you make such a design):

(i) finding oil spills along a coast line

Answer: Density-based clustering, such as DBSCAN and OPTICS, since it needs to detect arbitrary shaped clusters. □

(ii) clustering employees in a company based on their salaries and years of working experience

Answer: Partitioning-based clustering, better using k-medoids than using k-means (although k-means is basically OK (*i.e.*, give **1** point off) since k-medoids is less sensitive to outliers: a small number of employees/managers' salaries could be substantially higher than others. □

(c) [6] Why subspace clustering is a good choice for high-dimensional data? Outline one efficient and effective subspace clustering method that can cluster a very high dimensional (*e.g.*, thousands of dimensions) data set.

Answer: Subspace clustering is a good choice since it is hard to find reasonable clustering in high-dimensional space due to data sparsity and the distance in high-D space becomes equi-distance (dominated by dimension differences).

An efficient and effective method: p-clustering since it uses frequent pattern-based mining idea and compute δ-clusters which has downward closure property and thus can use Apriori-like pruning.

Note: CLIQUE is a subspace clustering but it should get **1** point off since it cannot handle very high-D, as required in the question. □

(d) [6] *Cross-validation* can be useful in both classification and clustering. What are the differences in these two cases?

Answer:

Cross-validation in classification: supervised, use it to select the best model: Partition data sets into m sets and use $(m-1)$ sets as training and the remaining one as testing. And such training and testing take m turns and cross validate each other. Select the model that has the highest accuracy.

Cross-validation in clustering: unsupervised, use it to select the best number of clusters, k. Partition the data into m sets, and user $(m-1)$ sets of data for clustering, and then assign the points in the remaining set to the formed clusters based on closeness. Do this m times in cross-validate way, and return the cluster that is most tight. Repeat this for different k values and determine the best k. □

Chapter 2

Sample Exam Questions for Course II

Enclosed are some sample midterm exam questions of the advanced level data mining course offered at Computer Science, UIUC: "UIUC CS 512: Data Mining: Principles and Algorithms". Since the course is more research oriented, in many offerings, there is only one midterm exams. But some semesters, there are two midterm exams. Each midterm exam had 90 minutes of time, close book, but allowing student to bring one sheet of paper (notes) worked out by students themselves. Due to limited time, we may not provide answers to those questions. If there were answers previously written down in those semesters, we will provide the answers to those questions.

Please note that many of the themes discussed in CS512 are beyond the scope covered in the 3rd edition of this textbook.

2.1 Sample Exam Question Set: 2.1

2.1.1 Midterm Exam

1. [24] Stream data mining.

   (a) [7] Briefly explain why *lossy counting* can be used to find approximate counts of frequent single items in data streams.

   Answer:

   - divide the stream into buckets, with size = $\frac{1}{\epsilon}$ where ϵ is the error threshold.
   - using the synopsis structure to accumulate the item count
   - At the bucket boundary, decreasing each item count by 1. If the count ≤ 0, remove the entry from the synopsis structure
   - Given support threshold σ and length N, report all the items whose count $\geq (\sigma - \epsilon)N$

- No false negative, false positive has count at least $(\sigma - \epsilon)N$. Frequency count underestimate at most ϵN.

□

(b) [7] Outline one stream classification method that is effective in classifying dynamic changing data streams.

(c) [10] Outline a method that can detect outliers, such as sharp rise in temperature in the last 24 hours in comparison with the surroundings, in a multi-dimensional space using a stream cube.

2. [16] Sequential pattern mining.

(a) [6] There are three typical sequential pattern mining algorithms: GSP (Srikant and Agrawal, 1996), PrefixSpan (Pei, et al. 2001) and SPADE (Zaki 2001). Outline the major differences of the three algorithms.

(b) [10] A typical sequential pattern mining algorithm, such as PrefixSpan, ignore gaps within sequences and find patterns of any length. However, a user may like to find patterns with only small gaps (such as 0 to 2) with the total pattern length between 10 to 15. Explain how you can modify the PrefixSpan algorithm to incorporate such constraints efficiently.

3. [26] Graph mining.

(a) [8] For the same graph database and the same *min_support* threshold, with what kind of data characteristics, the derived *closed* subgraph pattern set is far smaller than its corresponding *full* subgraph pattern set?

(b) [10] Explain why construction of GraphIndex, i.e., *gIndex*, needs *discriminant frequent subgraph pattern mining.* What are the major difference of such mining from gSpan?

(c) [8] Explain for similarity search upon a graph query with n edges, why it may not be a good idea to explore edge-missing (such as search subgraphs containing $(n-1)$ or $(n-2)$ edges of the n edges) subgraphs. What could be a better idea then?

4. [16] Information network analysis and multi-relational data mining.

(a) [8] Explain (1) why a social network is likely to follow a power law distribution, and (2) why the diameter of such a network may shrink if more nodes and edges are added in.

(b) [8] In multi-relational classification, explain why CrossMine may achieve higher scalability and better accuracy than a typical ILP approach such as FOIL.

5. [15] Mining biological data.

(a) [7] What is the major difference between *the first order* and *higher order* Markov chain models? What is "*hidden*" in the "*hidden Markov model*"?

(b) [8] Answer and explain what kinds of sequences can be discovered by BLAST but cannot be discovered by a typical "sequential pattern mining" method, and vice-versa.

6. [3] (Opinion).

(a) I □ like □ dislike the exams in this style.

(b) In general, the exam questions are □ too hard □ too easy □ just right.

(c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

2.2 Sample Exam Question Set: 2.2

2.2.1 Midterm Exam

1. [24] Stream data mining.

   (a) [8] Explain why stream cube can summarize very long multi-dimensional data streams with a relatively small data structure?

   Answer:

   - Use tilted tile-frames to keep recent information less compressed and more remote information more compressed. Thus it can summarize information of very long streams.
   - using critical layers: not need to store all the summary in multi-D space
   - use H-tree structure and only materialize popular path: minimal materialization in multi-D space

   □

   (b) [8] Explain why CluStream can achieve high scalability and high quality in clustering dynamically evolving data streams.

   Answer:

   - Use micro-cluster (such as CF-feature tree) so that information kept in compact and efficient way for further on-demand clustering
   - Use tilted timeframe to keep different time frames for discovery of evolution properties
   - Use offline incremental micro-clustering and online clustering query answering to achieve fast response

   □

   (c) [8] Among four popular classification methods: (1) decision-tree, (2) Naïve Bayesian, (3) support vector machine, and (4) k-nearest neighbor, which one is easy to be adapted to the classification of dynamically changing data streams, and how?

   Answer:

   - Decision-tree: not easy to be adapted since when new records arriving, splitting attributes may need to be changed, which may require the tree to be completely rebuilt.
   - Naïve Bayesian: easy to update since only changing some attribute-value: class-label distribution statistics.
   - Support vector machine: not easy since changing data may require to recalculate the support vector
   - k-nearest neighbor: easy in the sense micro-clustering can be performed to group neighbors into micro-clusters and such updating can be incremental, with tilted time-frame and incremental microcluster statistics update.

□

2. [25] Sequential pattern mining and biological sequence mining

(a) [9] Suppose a transaction database contains three transactions as follows. Note (bc) means that items b and c are purchased at the same time (*i.e.*, in the same transaction).

Seq_id	Sequence
1	$a\ (bc)\ d\ (ef)$
2	$(ade)\ (bc)\ g$
3	$(ab)\ d\ e\ g\ f$

Suppose the minimum support is 2. What should be the result for mining maximal (frequent) sequential pattern? Note that a sequential pattern s is *maximal* if there exists no sequential pattern p such that s is a proper subsequence of p.

Answer:

- $a(bc) : 2, ade : 2, abf : 2, bde : 2, bdf : 2, ag : 2, bg : 2, dg : 2, eg : 2$

□

(b) [8] What is the major difference between sequential patterns vs. closed sequential patterns? What is the key optimization that makes mining closed sequential pattern very efficient?

Answer:

- The major difference: s is a closed sequential pattern if s is frequent and there exists no superpattern of s having the same support as s. This will substantially reduce the # of patterns to be output since the subpatterns of s will not output if they have the same support as s.
- Key optimization: If s_i's project DB size is the same as s_j's and $s_i \subset s_j$, then s_j's project DB can be pruned. This property can be used for backward/forward subpattern pruning based on the size of the projected DB.

□

(c) [8] What are the major differences between BLAST and a typical "sequential pattern mining" method?

Answer:

The major differences

- BLAST: for biosequence alignment (usually for two long sequences) vs. SPM: mining (with gapped subsequences) for many sequences
- BLAST: approximate match (weighted) vs. SPM: precise match based on minimum support
- BLAST: heuristic suboptimal alignment algorithms based on dynamic programming vs. SPM: complete (Apriori-based) search

□

3. [24] Graph mining.

   (a) [8] For the same graph database and the same *min_support* threshold, with what kind of data characteristics, the derived *closed* subgraph pattern set is far smaller than its corresponding *full* subgraph pattern set?

   Answer:

   - If the set of frequent graph patterns in a graph transaction DB mainly consists of a set of "common" large subgraphs, or say the graph DB mainly consists of a set of "common" (frequent) large subgraphs

   □

   (b) [8] Explain why indexing on graph structures (*e.g.*, *gIndex*) leads to more compact and better quality index structures than indexing on path (*e.g.*, GraphGrep)?

   Answer:

   - Graph index models the structure better since it can distinguish different structures, whereas many different structures may share the same path
   - Graph index is constructed based on frequent and discriminative graph pattern analysis and thus is compact and effective: fetch less unmatched answers.

   □

   (c) [8] Explain (1) why *precise* graph index structure is useful for *approximate search* upon a graph query, and (2) how such approximate graph search can be supported by graph index effectively and efficiently.

   Answer:

   - *precise* graph index structure is useful for *approximate search* upon a graph query: relaxation is done on queries instead of on index. Index structure still small and precisely associated with the concrete graph data sets
   - how such approximate graph search can be supported by graph index effectively and efficiently: view a query graph as a set of features (interesting subgraphs), and explore feature relaxation instead of edge relaxation.

   □

4. [24] Information network analysis and multi-relational data mining.

(a) [8] Explain (1) why a social network is likely to follow a power law distribution, and (2) why the diameter of such a network may shrink if more nodes and edges are added in.

Answer:

- Since a small number of nodes likely has more connections than others and such connection will further enhance the chance others connected to them. Thus the network distribution could easily form a power law distribution.
- If more nodes and edges are added into the network, the chance that some edges shorten the connections (hence the distance) between any two nodes increases and thus the diameter of the network may likely shrink.

□

(b) [8] Explain why (1) CrossMine does not perform join of all the relations into one universal relation, and (2) how CrossMine may achieve higher scalability and better accuracy than a typical ILP approach.

Answer:

- Join will lose semantic relationships of data objects and their links and also will create huge relations
- Using tuple_id propagation to propagate its links and class labels, it preserves the semantic relationships, carries minimal information to pass, confines its search only to the closely relevant relations, and enlarges search space to related relations via look-one-ahead.

□

(c) [8] Explain (1) why we need user-guidance for clustering multiple relations, and (2) what is the major difference between CrossClus and a typical semi-supervised clustering method? Note that *semi-supervised clustering* is a clustering process based on user's feedback or constraints (such as "cannot link" or "must-link") on certain data objects.

Answer:

- why need user-guidance for clustering multiple relations: There are many attributes or properties across multiple relations and many of them could be irrelevant to user's interest of clustering. If they are included, it will produce undesired clusters.
- major difference between CrossClus and a typical semi-supervised clustering method: use attribute as "guide", analyzes every tuple containing this attributes and their links with the tuples in other relations to find which are the important attributes and their weights in the clustering. After the analysis of pertinent attributes (and their weights), CrossClus will use an efficient clustering algorithm to perform quality clustering. A typical

semi-supervised clustering will only specify some constraints on a small number of objects.

□

5. [3] (Opinion).

   (a) I □ like □ dislike the exams in this style.

   (b) In general, the exam questions are □ too hard □ too easy □ just right.

   (c) I □ have plenty of time □ have just enough time □ do not have enough time to finish the exam questions.

2.3 Sample Exam Question Set: 2.3

2.3.1 Midterm Exam I

1. [32] Stream data mining

   (a) [8] *Lossy counting* is an interesting algorithm that finds approximate counts of frequent single items in data streams. Explain why lossy counting can claim that it finds all the frequent items (*i.e.*, no false negatives) with a guaranteed error bound.

   Answer:

   - Suppose σ is minimum support and ϵ is the error bound. Then Lossy counting sets the bucket size as $1/\epsilon$. If there are N items seen so far, the number of buckets would be ϵN.
   - Lossy counting decreases the count by the number of buckets, so the count of each item is decreased by at most ϵN. If its true count is f, then the count obtained by lossy counting would be $\geq f - \epsilon N$.
   - Lossy counting will output all the patterns with count $\geq (\sigma - \epsilon)N$. Since the count of a true frequent item would be at least σN, it will be output as a frequent item in lossy counting. So there are no false negatives.
   - The false positives made by lossy counting would have the count at least $(\sigma - \epsilon)N$. Compared with the true threshold σN, we can see that the error is bounded by ϵN.

   □

   (b) [8] Outline a method that effectively classifies highly skewed data sets in dynamic evolving data streams.

   Answer:

   biased sampling : Keep positive examples from B chunks of data, and under-sample negative examples from the most recent data chunk. This forms a sampled set D'.

   ensemble : Construct k data sets from D'. Each of the data sets contains all the positive examples and $1/k$ of the negative examples in D'. Build k classifiers based on the k sets and average the predictions from these classifiers as the final outputs.

   □

   (c) [8] Explain why stream cube can summarize very long multi-dimensional data streams with a relatively small data structure?

   Answer:

   tilted-time framework We can register time at different levels of granularity in stream cubes. The most recent time is registered at the finest granularity, whereas the more distant time is registered at a coarser granularity.

two critical layers In stream cubes, we dynamically and incrementally compute and store two critical layers, minimal interest layer and observation layer. The minimal interest layer can satisfy users' finest requests using minimal storage, rather than storing the primitive data. The observation layer provides information for most of users' needs, so that users usually stay at the observation layer and only occasionally drill down if some exceptions are found.

partial materialization The popular path cubing approach materializes only the cuboids along the popular path from the minimal interest layer to the observation layer, and leaves the other cuboids to be computed on the fly. This guarantees quick aggregation, quick drilling and small space requirements.

□

(d) [8] Explain why CluStream can achieve high scalability and high quality in clustering dynamically evolving data streams.

Answer:

- CluSteam divides the clustering process online and offline components. The online component computes and stores summary statistics about the data stream using microclusters. The use of microclusters improves both clustering quality and scalability because important clustering features are kept in microclusters and unimportant details are ignored. The offline component can perform user-directed macroclustering or cluster evolution analysis. This offers rich functionality to the user.
- The tilted time framework is used to store microclusters at different levels of granularity depending on recency. This framework only registers the essential historical information, and thus provides better quality and scalability.

□

2. [25] Sequential pattern mining **Answer:**

(a) [9] Suppose a sequence database contains three sequences of transactions as follows. Note (bc) means that items b and c are purchased at the same time (*i.e.*, in the same transaction).

Seq_id	Sequence
1	$a\ (bc)\ d\ (ef)$
2	$(acd)\ (bc)\ g$
3	$(ab)\ d\ e\ g\ f$

Suppose the minimum support is 2. What is the complete set of sequential patterns? What is the complete set of closed sequential patterns? Note that a sequential pattern s is *closed* if there exists no sequential pattern p such that s and p have the same support (*i.e.*, frequency) and s is a proper subsequence of p.

Complete Set:

length-1 pattern : $< a >$: 3, $< b >$: 3, $< c >$: 2, $< d >$: 3, $< e >$: 2, $< f >$: 2, $< g >$: 2.

length-2 pattern : $< ab >$: 2, $< ac >$: 2, $< ad >$: 2, $< ae >$: 2, $< af >$: 2, $< ag >$: 2, $< (bc) >$: 2, $< bd >$: 2, $< be >$: 2, $< bf >$: 2, $< bg >$: 2, $< de >$: 2, $< df >$: 2, $< dg >$: 2.

length-3 pattern : $< a(bc) >$: 2, $< ade >$: 2, $< adf >$: 2, $< bde >$: 2, $< bdf >$: 2.

Set of Closed Patterns:

length-1 pattern : $< a >$: 3, $< b >$: 3, $< d >$: 3.

length-2 pattern : $< ag >$: 2, $< bg >$: 2, $< dg >$: 2.

length-3 pattern : $< a(bc) >$: 2, $< ade >$: 2, $< adf >$: 2, $< bde >$: 2, $< bdf >$: 2.

□

(b) [8] What are the major differences among three typical sequential pattern mining algorithms: GSP (Srikant and Agrawal, 1996), PrefixSpan (Pei, et al. 2001) and SPADE (Zaki 2001)?

Answer:

GSP, SPADE : Both GSP and SPADE adopt candidate generate-and-test approach in breadth-first search. At each time, length-$k+1$ patterns are generated based on frequent length-k patterns. The Apriori principle is used to prune the candidates. GSP uses horizontal data formats, so it has to scan databases multiple times. SPADE uses vertical data format, and thus reduces scans of the database.

PrefixSpan : PrefixSpan does not require candidate generation, instead, it adopts a pattern growth approach using depth-first search. The major cost is to construct the projected database, which can be reduced by pseudo-projection. PrefixSpan achieves the best overall performance compared with GSP and SPADE.

□

(c) [8] What is the key optimization that makes CloSpan (for mining closed sequential patterns) highly efficient in comparison with PrefixSpan (for mining sequential patterns)?

Answer:

- CloSpan is based on the property of "equivalence of projected databases" to prune the search space when mining closed sequential patterns. The property states that if two subsequences $s \subseteq s'$ or $s' \subseteq s$ and their projected databases have the same size, then we can stop searching in one of them.

- Based on this property, CloSpan can prune backward subpatterns and backward superpatterns, and thus derive a compact set of sequential patterns in a shorter time than PrefixSpan.

□

3. [24] Graph mining

(a) [8] In comparison with several existing graph pattern mining algorithms, what are the major distinguished features of gSpan that make it efficient and scalable?

Answer:

- Both pattern-growth and *right-most extension* methods guarantee completeness and correctness of result and largely reduce the number of candidate subgraphs generated.
- Introduces *DFS code* to encode subgraphs. Such encoding scheme induces a subgraph lexicographic order to avoid generating duplicate graphs, thereby saving the cost of isomorphism checking.

□

(b) [8] Explain why *discriminant frequent subgraph patterns* but not frequent subgraph patterns are needed for deriving graph indices, *i.e.*, *gIndex*. How should we revise gSpan to derive such patterns?

Answer:

Why. If we simply use frequent subgraph patterns for indexing, a query would retrieve a large number of redundant graphs; such an indexing method has virtually no pruning power. On the contrary, frequent discriminant subgraph patterns can distinguish between different graphs and provide much *better pruning power*. Further, its index size would be orders of magnitude *smaller*.

How. We use a frequent pattern mining algorithm similar to gSpan to iteratively mine a set of frequent substructures. Let $f_1, f_2, \ldots, f_n$ be the already selected index substructures. Consider a new frequent substructure x: its discriminative power is measured by

$$Pr(x|f_{i_1}, f_{i_2}, \ldots, f_{i_m}), f_{i_j} \subseteq x, 1 \leq i_j \leq n. \qquad (2.1)$$

If the measure is smaller than a threshold (*i.e.*, graphs indexed by x are largely uncovered by previously selected features), we can select x as an index feature.

Moreover, a "size-increasing support constraint" is enforced on the selection of frequent fragments for effective indexing. We bias the feature selection to small fragments with low minimum support and large fragments with high minimum support. This method leads to two advantages: (1) the number of frequent fragments so obtained is much less than that with the lowest uniform *minSup*, and (2) low-support large fragments may be indexed well by their smaller subgraphs; thereby we do not miss useful fragments for indexing.

□

(c) [8] Explain for similarity search upon a graph query with n edges, why it may not be a good idea to explore edge-missing (such as search subgraphs containing $(n-1)$ or $(n-2)$ edges of the n edges) subgraphs. What could be a better idea then?

Answer:

Why. Relaxing m edges from an n-node query graph would generate $O(n^m)$ possible subgraphs, and enumerating and checking each of these subgraphs can be prohibitively expensive.

i. **Better idea.** View graphs as feature vectors. If graph G contains the major part of a query graph Q, G should share a number of common features with Q. Detailed steps are as follows.

Step 1: Index Construction Select small structures as features in a graph database, and build the feature-graph matrix between the features and the graphs in the database.

Step 2: Feature Miss Estimation (1) Determine the indexed features belonging to the query graph; (2) Calculate the upper bound of the number of features that can be missed for an approximate matching, denoted by J. (On the query graph, not the graph database.)

Step 3: Query Processing Use the feature-graph matrix to calculate the difference in the number of features between graph G and query Q, $F_G - F_Q$. If $F_G - F_Q > J$, discard G. The remaining graphs constitute a candidate answer set.

□

4. [16] Mining time-series and biological data

Answer:

(a) [8] Explain why for time series analysis, a motif-based approach, such as SAX (Symbolic Aggregate approXimation) by Keogh et al., may outperform a typical similarity search algorithm.

- Symbolic representation of continuous time series data is generated through *discretization*, which dramatically *reduces storage overhead* and allows various data mining tasks to be performed *in memory.*
- Symbolic representation enables automatic *dimensionality reduction.*

- True distance between original time series can be *lower-bounded* by the distance between transformed sequences, giving time series analysis quality assurance.
- *Hashing* and *neighborhood pruning* further speed up motif searching.

□

(b) [8] For biological sequence analysis, explain what are the major differences of an algorithm like BLAST vs. a typical sequential pattern mining algorithm.

Answer:

- BLAST aims to find regions of local similarity between biosequences (hence it is *approximate* in nature); a typical sequential pattern mining algorithm aims to find frequently occurred sequential patterns (usually in an *exact* fashion).
- In BLAST a similarity threshold is set based on *statistics*, while in sequential pattern mining a frequency threshold is set in terms of *co-occurrence.*
- BLAST works well on *long biosequences with a few distinct symbols*, while sequential pattern mining often performs on shorter *transaction data with a large number of distinct items.*
- BLAST starts from exact matches between small fragments and *extends matches in both directions* of a sequence; a typical sequential pattern mining algorithm often extends pattern in *one direction or adds a single item at a time.*
- BLAST employs techniques like *dynamic programming and hashing*, while sequential pattern mining algorithms are based on *BFS/DFS.*

□

2.3.2 Midterm Exam II

1. [48] Mining graphs, social and information networks

   (a) [8] Social networks can be generated according to different models. What are the major differences among the following three models: *random graphs*, *Watts-Strogatz models*, and *scale-free networks*?

   Answer: The following two answers are both correct.

   Properties Difference

   i. random graphs
      - gives few components and small diameter
      - does not give high clustering and heavy-tailed degree distributions
      - is the mathematically most well-studied and understood model

   ii. Watts-Strogatz models
      - give few components and small diameter
      - give high clustering
      - does not give heavy-tailed degree distributions

   iii. scale-free networks
      - gives few components and small diameter
      - gives heavy-tailed distribution
      - does not give high clustering

   Generation Difference

   i. random graphs
      - number of nodes is fixed, and all edges are equally probable and appear independently
      - sharing a common neighbor makes two vertices no more likely to be directly connected than two very distant vertices

   ii. Watts-Strogatz models
      - number of nodes is fixed
      - two vertices are more likely to be linked if they share many common neighbors

   iii. scale-free networks
      - number of nodes is not fixed where new nodes are continuously added to the graph
      - a node is linked with higher probability to a node that already has a large number of links

   □

   (b) [8] What are the major difficulties at clustering multiple interconnected relations? Outline a method that may overcome these difficulties.

   Answer: **Difficulties**

i. traditional clustering algorithms work on single relation and cannot be applied on multiple interconnected relations directly

ii. if we simply flatten the relations by physical joins, we may lose important information of linkages and relationships, and also generate large amounts of duplicates an waste storage.

iii. Due to interconnections between relations, it is hard to select pertinent features are relevant to the clustering task. These features may be embedded in any relation.

Method: CrossClus

i. CrossClus first defines multi-relational features by join paths and attributes, and measures similarities between features by how they cluster objects into groups.

ii. CrossClus starts with user-specified attribute, and then repeatedly searches for pertinent features from the neighborhood defined by feature similarity.

iii. The similarity between two objects can then be defined based on the pertinent feature weight obtained during the search. A k-medoids-based algorithm can be used to clustering the objects.

□

(c) [8] Why is SimRank costly at clustering large heterogeneous information networks? Outline a method that may substantially reduce its computational complexity.

Answer: SimRanks Inefficiency

i. SimRank requires computation of pair-wise similarity

ii. It is quadratic in both space and time. For a network with N nodes and M links, the space and time complexity are $O(N^2)$ and $O(M^2)$ respectively.

Method-LinkClus: LinkClus uses a hierarchical structure SimTree to store similarities in a multi-granularity way. It stores detailed similarities between closely related objects, and overall similarities between object groups.

LinkClus works as follows.

Initialize a SimTree for each type of objects in the network based on frequent pattern mining.

Repeat

- For each SimTree, iteratively update the similarities between its nodes using similarities in other SimTrees. Similarity between two nodes x and y is the average similarity between objects linked with them.
- Adjust the structure of each SimTree. Assign each node to the parent node that it is most similar to under the degree constraints of the parent node.

□

(d) [8] In the DBLP (*i.e.*, computer science bibliographic) database, one may encounter multiple authors that share the same names. Outline a method that distinguishes these authors and their corresponding publications.

Answer: Method-DISTINCT

- DISTINCT combines two approaches for measuring similarities between references: (1) the neighbor tuples of each reference, and (2) linkages between two references calculated using random walk probability model
- DISTINCT uses supervised learning to determine the pertinence of each join path and assign a weight to it. DISTINCT constructs the training set automatically by extracting people with rare names.
- Given a set of references to the same name, DISTINCT group them into clusters using hierarchical agglomerative clustering method, so that each cluster corresponds to a real entity.

□

(e) [8] Multiple information providers may provide conflict information about an object. Outline a method that may tell which assertion is likely to be true and which information providers are more trustable than others.

Answer: Method-TruthFinder

- TruthFinder is developed based on the following heuristics:
  i. There is usually only one true fact for a property of an object.
  ii. This true fact appears to be the same or similar on different web sites.
  iii. The false facts on different web sites are less likely to be the same or similar, i.e., false facts are often introduced by random factors.
  iv. A web site that provides mostly true facts for many objects will likely provide true facts for other objects.
- Therefore, a fact has high confidence if it is provided by (many) trustworthy web sites, whereas a web site is trustworthy if it provides many facts with high confidence.
- TruthFinder works as follows
  i. Initially, each web site is equally trustworthy.
  ii. Update the fact confidence as one minus the probability that all web sites the fact are wrong, and the trustworthiness of a website as the average confidence of facts it provides.
  iii. Repeat until achieving stable state.

□

(f) [8] Why do ranking and clustering often help each other in heterogeneous information analysis? Outline a method that may integrate ranking and clustering to generate high-quality clustering and ranking results.

Answer: Ranking and Clustering

- Ranking and clustering each can provide general views over information network. Ranking globally without considering clusters is meaningless. Clustering objects without distinction is not desirable either because highly-ranked objects should play a much important role in clustering.
- Ranking and clustering can be mutually improved: better clustering results may provide more meaningful ranking. Meanwhile, better ranking distribution can help generate better measures for deriving better clusters.

Method-RankClus

i. Randomly partition target objects into K clusters and derive sub-networks from each cluster.

ii. Calculate rankings for objects in each sub-network, which serves as features for each cluster.

iii. Fit target object with the rank features into a mixture model and estimate the model coefficients for each target object.

iv. Use the component coefficients as new measures for target objects and adjust clusters accordingly.

v. Repeat steps 2,3,4 until convergence.

□

2. [24] Mining spatiotemporal, multimedia, and trajectory data

(a) [8] At mining frequent patterns in spatial or spatiotemporal data, *progressive deepening* and *co-location* are two useful methodologies. Use one example to illustrate how to use these methodologies.

Answer:

The progressive deepening method exploits the hierarchy of spatial relationship, such as near by, touch, intersect, contain, etc., by first searching for rough relationship and then refining it. Specifically, there are two steps. First is rough spatial computation (as a filter) using MBR or R-tree for rough estimation. The second step is detailed spatial algorithm (as refinement). This step applies only to those objects which have passed the rough spatial association test (no less than min support).

Co-location rule is similar to association rule but explore more spatial auto-correlation. It leads to efficient processing and can be integrated with progressive refinement to further improve its performance. The spatial co-location mining idea can be applied to clustering, classification, outlier analysis, and other potential mining tasks.

An example could be studying vegetation durability or temperature distribution across marshland, where high-level associations between the distribution and location can be found and then such associations are refined in detail. The co-location method, on the other hand, can be applied to efficiently find similar patterns in spatial neighborhoods. □

(b) [8] Invariance is often valuable at mining multimedia data. Use one example to illustrate how to use invariance in effective multimedia data mining.

Answer: An example is the SpaRClus method, which exploits invariance to find patterns persistent over scaling, translation, and rotation. This method is able to make good use of bag-of-item representation and spatial information; 3-pattern in the form of $(\langle a1; a2; a3\rangle;$ θ; r) is considered as a basic unit of spatial pattern, and frequent spatial patterns can be subsequently mined for image clustering, classification, etc. □

(c) [8] The anomalies of a moving object are often associated not only with its moving paths but also with other information, such as moving object category, time, weather, locations, etc. Outline a method that may effective identify anomalous moving objects.

Answer:

The idea is to use a motif-based anomaly analysis framework. In motif-based representation, a motif is a prototypical movement pattern; it views a movement path as a sequence of motif expressions. In motif-oriented feature space, motif features and semantic-level information such as category, time, and weather are extracted. Then, high-dimensional classifiers are trained for anomaly detection.

□

(d) [8] RFID data often contains much redundancy. Outline a method that may substantially reduce data redundancy at warehousing RFID data.

Answer:

First, we clean the data by merging readings of a product at the same location at consecutive time points.

Second, bulky movement compression is applied based on the observation that objects often move and stay together. The GID, or generalized identifier, is used to represent a bulk of products; the naming of GID enables it to encode movement paths.

Finally, further compression is achieved through compression by data and path generalization. In data generalization, we aggregate object movements into fewer records. For example, if one is interested in time at the day level, he can merge records at the minute level into records at the hour level. In path generalization, path segments can be merged and/or collapsed. For example, multiple item movements within the same store may be uninteresting to a regional manager and thus merged.

□

3. [16] Mining text, Web, and software program data

(a) [8] A language model usually assumes a language is generated using a unigram model. However, sometimes one may need to consider bi-grams, tri-grams, or even sequences (i.e., ordering could be important). At what condition is it important to consider a sequence modeling? Then how to revise a unigram model generation method to derive such a sequence model?

Answer: (Note: no point would be deducted if topic cube is not in the answer.)

Types of queries to be supported:

- Given a multidimensional cell, what are the TF/IDF scores of terms at different levels of a term hierarchy?
- Given a multidimensional cell, what are the topics at different levels of a topic hierarchy? What is a particular topics coverage and deviation?

The design of a text cube integrates the power of traditional data cube and IR techniques for effective text mining. Instead of computing multidimensional aggregates, the text cube precomputes IR measures as well as topic models for multidimensional text database analysis, so that heterogeneous records are examined in both structured categorical attributes and unstructured free text.

The following steps are used to construct a text cube: (1) Build up term hierarchies and dimension hierarchies. (2) Do partial materialization via dynamic programming in order to minimize space complexity. (3) Now on-line queries can be answered by aggregating queried cells from off-line computed subcells with bounded time threshold.

Steps for topic cube construction: (1) Run PLSA for each cell. (2) For different term hierarchies, build up topic hierarchies. (3) Computing PLSA of super-cells using the sum of the PLSA results of sub-cells as initial value so as to speed up processing. (4) Now multidimensional topic analysis can be conducted.

□

(b) [8] It is important to analyze a multidimensional text database that contains both structural data (such as time, location, author, category) and narrative text. What kinds of queries that can be answered in such a database? Outline the design of a text data cube that may facilitate the answering of such queries.

Answer: There are 4 steps:

Step 1 Parse source code and build a sequence database: The purpose is to build a sequence database. The idea is to map statements to numbers. Each component is tokenized and different operators, constants, keywords are mapped to different tokens. To handle identifier renaming, same type of identifiers will be mapped to same token. Now, the program can be transformed into a sequence database.

Step 2 Mining for basic copy-pasted segments: Apply frequent sequence mining algorithm on the sequence database with a modification to constrain the max gap.

Step 3 Compose larger copy-pasted segments: Combine the neighboring copy-pasted segments repeatedly.

Step 4 Prune false positives: We prune those un-mappable segments whose identifier names cannot be mapped to corresponding ones as well tiny segments.

Repeat Steps 3 to 4 if necessary.

□

4. [4] (Opinion).

2.4 Sample Exam Question Set: 2.4

2.4.1 Midterm Exam I

1. [30] Stream data mining

   (a) [10] Why is it that the k-median-based stream clustering algorithm by O'Callaghan et al. (2002) cannot disclose evolving regularities of data streams? Why can CluStream achieve better clustering quality in comparison with the stream clustering algorithm proposed by O'Callaghan et al. (2002)?

   (b) [10] Among three popular classification methods: (1) Naïve Bayesian, (2) support vector machine, and (3) k-nearest neighbor, which one is easy to be adapted to the classification of dynamically changing data streams, and how? and which one is difficult to do so and why?

   (c) [10] If you want your stream data cube be able to compare the power consumption of every weekday in this week with that of every weekday in the last week, last month, and last year, how would you design your stream data cube?

2. [30] Sequential pattern mining

   (a) [14] Suppose a sequence database contains two sequences as follows.

Seq_id	Sequence
1	$\langle a_1, b_2, a_3, b_4, \ldots, a_9, b_{10} \rangle$
2	$\langle a_1, a_2, a_3, a_4, \ldots, a_9, a_{10} \rangle$

   Suppose the minimum support is 2, and $a_i \neq bj$ for any i and j.

   (i) What is the complete set of closed sequential patterns? Note that a sequential pattern s is *closed* if there exists no sequential pattern p such that s and p have the same support (*i.e.*, frequency) and s is a proper subsequence of p.

   (ii) For the complete set of patterns, how many sequential patterns are there? List 5 of them.

   (b) [16] Among sequential pattern mining algorithms,

   (i) Why is GSP (Srikant and Agrawal, 1996) less efficient than PrefixSpan (Pei, et al. 2001)?

   (ii) Why is PrefixSpan (Pei, et al. 2001) less efficient than CloSpan (Yan et al. 2003)?

   (ii) Why do all these algorithms encounter challenges at mining long (*e.g.*, length $\geq$ 100) sequential patterns?

3. [22] Graph mining

   (a) [12] Explain why one needs to perform data mining to find efficient graph index structure? And what are the major differences for mining graph index candidate in *gIndex* different from gSpan?

(b) [10] As we know, graph isomorphism test is an NP-complete problem. Explain why one still can mine the complete set of frequent closed subgraph patterns in a large chemical compound structure dataset.

4. [15] Mining time-series and biological data

   (a) [8] List three major differences of similarity search in graph databases vs. that in time series databases.

   (b) [7] For biological sequence analysis, explain what are the commonalities and differences of two algorithms: BLAST and FASTA.

5. [3] (Opinion)

2.4.2 Midterm Exam II

1. [30] Network modeling

   (a) [11] Researchers have been modeling social and/or information networks using several models. What are the differences of the following three models: (1) Edös-Rényi model, (2) Watts-Strogatz model, and (3) scale-free network.

   (b) [8] Given a network, how can you judge what distribution that the network follows?

   (c) [11] Suppose a group of users would like to promote their own webpages by cross-linking their home pages each other. Will this practice give their pages higher ranks by the original PageRank algorithm? Why? Propose a revision to PageRank to discredit such tricks.

2. [22] Mining multi-relations

   (a) [11] For classification of multiple relational databases, CrossMine uses three techniques: (1) tuple-id propagation, (2) look-one-head in rule generation, and (3) negative sampling. Briefly reason why each technique is useful at improving classification quality.

   (b) [11] What is semi-supervised clustering? In clustering multiple interconnected relations, why user-guided guided clustering could be more effective than a typical semi-supervised clustering method?

3. [22] Mining heterogeneous information networks

   (a) [11] Explain why can TruthFinder discover which pieces of information is more likely to be true without using a training set?

   (b) [11] Explain why a typical algorithm for clustering homogeneous information networks may not generate high quality clusters for heterogeneous information networks? Why can NetClus overcome these difficulties?

4. [22] Mining spatiotemporal and moving object data

   (a) [11] Outline a method that may help find periodicity of a set of moving objects that may contain more than one period, *e.g.*, Peter may go to work every weekeday morning but will only go to gym every Tuesday afternoon.

   (b) [11] Giving an example of spatial co-location rules that may disclose that restaurants are often near highway exits, and outline an efficient method for mining such kind of rules.

5. [4] (Opinion)

2.5 Sample Exam Question Set: 2.5

2.5.1 Midterm Exam I

1. [40] Advanced Clustering

(a) [10] *Subspace clustering* and *dimensionality reduction* are the two major approaches for clustering high-dimensional data. Discuss their differences and at what condition one approach is more appropriate than the other.

Answer: Subspace clustering is used to search for clusters existing in subspaces of the given high dimensional data space. The representative subspace clustering methods include CLIQUE, ProClus and bi-clustering approaches. Dimensionality reduction is used to construct a much lower dimensional space and search for clusters there (it may construct new dimensions by combining some dimensions in the original data). The representative dimensionality reduction methods include spectral clustering.

In some situations, clusters may exist only in some subspaces of the given high dimensional data space. In order to find clusters in all such subspaces, the subspace clustering methods are more appropriate.

In some situations, it is more effective to construct a new space instead of using some subspaces of the original data. Here dimensionality reduction methods are more appropriate. □

(b) [10] Both *CLIQUE* and *δ-pClustering* are subspace clustering methods. What are the major differences between the two? Give one application example that δ-pClustering is more appropriate than CLIQUE.

Answer:

CLIQUE is a subspace search method for subspace clustering. CLIQUE adopts a bottom-up approach to searching various subspaces in order to find clusters. It starts from low-dimensional subspaces and search higher-dimensional subspaces only when there may be clusters in such subspaces. CLIQUE leverages various pruning techniques to reduce the number of higher-dimensional subspaces to be searched.

δ-pClustering is a bi-clustering method, which uses a tolerance threshold to specify the degree of noise allowed in the bi-clusters to be mined (The p-score in δ-pClustering controls the noise on every element in a bi-cluster, while the mean squared residue captures the average noise). It then tries to enumerate all sub-matrices as bi-clusters that satisfy the requirements.

A typical application example that δ-pClustering is superior is clustering gene expression or micro-array data. □

(c) [10] Both *SimRank* and *SCAN's structure similarity* measure similarities between two vertices in a network. What are their differences? For what applications SimRank is more suitable? How about SCAN?

Answer:

Simrank measures the structure similarity of two vertices based on the similarity of their incoming neighbors (recursive definition, as shown in Eq. 2.2). From another viewpoint, SimRank measures the random walking hitting time of two vertices along the reverse direction of incoming edges. SimRank considers similarity computation beyond the direct neighborhood toward the whole graph structure, i.e., vertices beyond the neighborhood scope of two vertices may affect the SimRank scores for two vertices under examination. SimRank is expensive to compute. The time complexity is O(n^4) where n is the number of vertices in the network. (The best-known algorithm for exact SimRank computation is O(n^3) (VLDB'08)).

$$s(u,v) = \frac{C}{|I(u)||I(v)|} \sum_{x \in I(u)} \sum_{y \in I(v)} s(x,y) \tag{2.2}$$

SCAN'S similarity measure is defined as follows

$$\sigma(u,v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{\sqrt{|\Gamma(u)||\Gamma(v)|}} \tag{2.3}$$

where $\Gamma(\cdot)$ represents the set of neighbors (including the vertex itself). Therefore, SCAN'S similarity makes use of *normalized direct* common neighbors as similarity indicator of two vertices in the graph. Intuitively, the structural similarity is large for vertices of a clique and small for hubs and outliers. SCAN's similarity is fairly efficient to compute in O(n).

Another difference is that SimRank is proposed on directed graphs while SCAN is proposed on undirected graphs.

SimRank is more suitable when the whole graph structure is of importance in computing structural similarity between vertices. For this kind of applications, not only neighborhood information, but vertices beyond neighborhood are considered during similarity computation. SCAN's similarity measure is preferable if only the localized neighborhood information is important for similarity computation. For example, friends' friends have no impact on the similarity computation of two individuals.

□

(d) [10] For constraint-based clustering, must-link and cannot-link are popularly used constraints. How to do clustering, taken a must-link *soft constraint*? And how to do it if the must-link constraint is a *hard constraint*?

Answer:

If a must-link is considered soft constraint, the clustering problem is treated as an optimization problem: When a clustering violates a soft constraint, a penalty is imposed on the clustering. Therefore the overall objective of clustering is to optimize the clustering quality, and minimizing the constraint violation penalty.

If a must link is considered hard constraint, we need strictly respect the constraints in cluster assignments. The general way to do this is to generate super-instances for must-link constraints. We compute the transitive closure of the must-link constraints. For such a closure, we replace all those objects in the closure by its mean as a super-instance. The super-instance also carries a weight, which is the number of objects it represents.

□

2. [30] Outlier Analysis

   (a) [10] What are the differences between *global outlier* vs. *local outlier*? Outline one method that can find local outlier effectively.

   Answer: Global outlier, a.k.a. point anomaly, is the object which significantly deviates from the rest of the data set. For example, intrusion detection in computer networks. Local outlier is referred to as an outlier comparing to its local neighborhoods, instead of the global data distribution.

   Density-based outlier detection methods can be used to detect local outliers. Note that the density around an outlier object is significantly different from the density around its neighbors. So we can use the relative density of an object against its neighbors, LOF (local outlier factor) for instance, as the indicator of the degree of the object being outliers. □

   (b) [10] Web pages in a university are structured as a big network. What are *collective outliers* in this case? Outline a method to detect such outliers.

   Answer:

   In general, collective outliers are objects as a group deviating significantly from the entire data. In a web page network for a university, a collective outlier can be a sub-network whose participants (web pages) are closely linked, thus forming a quasi-clique or clique structure.

   One possible method to detect collective outliers in a network is to model the expected behavior of structure units directly. For example, we can treat each possible subgraph of interest as a structure unit, and use heuristics to explore the potential subgraphs with high compactness. □

(c) [10] Suppose the number of items sold in Walmart by week, by item, by store location form a multidimensional data cube. How to define outlier in such a data cube (i.e., it consists of one 3-D cuboid: week-item-location, three 2-D cuboids: week-item, week-location and item-location, and three 1-D cuboids: week, item, location) to make it interesting in applications.

Answer: Data cube is often used for modeling multidimensional data. In order to define outliers over multidimensional data, we will start from much lower dimensional subspaces, such as (location), (week) and (item) cuboids. The reason is that it is fairly easy to interpret why and to what extent the object is an outlier, e.g., find outliers in certain subspace (item): the number of a particular outlier sold $\gg$ the average number of items sold. Then we will explore outliers in different subspaces with higher dimensions. Note that outliers within different subspaces are adaptive to the subspaces signifying the outliers and can capture the local behavior of the underlying multidimensional data. However, outlier detection on the high-dimensional subspaces becomes meaningless due to data sparsity. The reason is that distances between objects becomes heavily dominated by noise as the dimensionality increases. □

3. [30] Stream Data Mining

(a) [10] How to design a stream data cube to facilitate watching power consumption fluctuation in Illinois and be able to drill down by time, location, etc.?

Answer: For the stream data cube of power consumption, we first adopt the tilted time frame strategy to model different time granularities, such as second, minute, quarter, hour, day and week. We define two different critical layers on the stream data cube: minimum interest layer (m-layer) and observation layer (o-layer), in order to observe power consumption fluctuations at o-layer and occasionally drill-down down to m-layer. We partially materialize a subset of cuobids along the popular drilling paths and make use of H-tree structure for efficient storage and computation. □

(b) [10] Suppose one would like to find frequent single items (such as frequent router address) in dynamic data streams, with a minimum support σ and error bound ϵ. Explain why lossy counting algorithm can find the frequent items with a guarantee error bound ϵ.

Answer:

Given the length of the data stream seen so far as N, Lossy Counting (LC for short) divide the stream into buckets of size $1/\epsilon$. At each bucket boundary, LC decreases the count of items by 1. In this way, the frequency count error is less than the number of buckets, which equals ϵN. And true frequencies of false positives are lower-bounded by $(\sigma - \epsilon)N$. Note LC does not generate false negatives. □

(c) [10] Network intrusions are often critical but rare events. Outline a method that effectively builds classifiers for highly skewed data in dynamic evolving data streams.

Answer: As the data is highly skewed, we make use sampling based methods to compose the training data set and incorporate positive examples to increase the training set size. In such a way, the variance of data bias can be effectively reduced. We further make use of an ensemble method for classification in order to reduce variance caused by a single model. This sampling and ensemble based classification techniques can be effectively used for classifying highly skewed data in dynamic data streams. □

2.5.2 Midterm Exam II

1. [20] Sequential pattern and time-series data mining

   (a) [10] Suppose a sequence database contains two sequences as follows.

Customer_id	Shopping_sequence
1	$\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle$
2	$\langle a_1, a_2, c_3, a_4, a_5, c_6, a_7, a_8, c_9, a_{10}\rangle$

   Suppose the minimum support is 2, and $a_i \neq c_k$ for any i and k.

   What is the complete set of closed sequential patterns? Note that a sequential pattern s is *closed* if there exists no sequential pattern p such that s and p have the same support (*i.e.*, frequency) and s is a proper subsequence of p.

   (b) [10] Use an example to explain why motif-based similarity search method (*e.g.*, SAX: Symbolic Aggregate approXimation) can be effective and efficient at finding similar time-series.

2. [20] Graph mining

   (a) [10] What are the major differences for mining graph index candidates in *gIndex* different from mining frequent graph patterns in gSpan?

   (b) [10] Explain for similarity search upon a graph query with n edges, why it may not be a good idea to explore edge-missing (such as search subgraphs containing $(n-1)$ or $(n-2)$ edges of the n edges) subgraphs. What could be a better idea then?

3. [20] Network modeling

   (a) [10] In a large social network, the network diameter could be quite small. Adding more nodes and edges to the network often further shrinks its diameter. Are these two statements true? Briefly defend your answers.

   (b) [10] Researchers have been modeling social and/or information networks using several models. What are the differences of the following three models: (1) Edös-Rényi model, (2) Watts-Strogatz model, and (3) scale-free network.

4. [37] Mining heterogeneous information networks

   (a) [10] Explain why RankClus can generate both high-quality clustering and ranking effectively and efficiently in heterogeneous information networks.

   (b) [9] Explain what are the key techniques that *Distinct* can distinguish objects of identical names.

   (c) [9] Explain why can TruthFinder discover which pieces of information are more likely to be true without using a training set.

(d) [9] Explain why a meta-path-based method may have the power to predict coauthor relationships in a heterogeneous bibliographic information network.

Chapter 3

Ph.D. Qualification Exam Questions

Enclosed are some sample Ph.D. qualify exam questions in the data mining section offered at Data and Information Systems (DAIS) Research Area in the Department of Computer Science, UIUC. The DAIS Ph.D. exam covers four research areas: Database Systems, Data Mining, Information Retrieval and Bioinformatics. Each written exam lasts 3 hours and the examinees will need to answer one section containing a set of short questions and then select 5 regular questions (marked as Problem) to answer. Thus each full question is allocated approximately 30 minutes of time.

Many of the Ph.D. qualifying exams are open-ended research questions. Thus we have no intention to provide standard answers to those questions. Interested readers may like to work out "answers" by themselves.

3.1 Sample Exam Question Set 3.1

Problem 1

Part 1

Assume a base cuboid of 10 dimensions contains only three base cells: (1) $(a_1, d_2, d_3, d_4, \ldots, d_9, d_{10})$, (2) $(d_1, b_2, d_3, d_4, \ldots, d_9, d_{10})$, and (3) ($d_1$, d_2, c_3, d_4, …, d_9, d_{10}), where $a_1 \neq d_1$, $b_2 \neq d_2$, and $c_3 \neq d_3$. The measure of the cube is *count*.

(a) How many **nonempty** cuboids will a complete data cube contain?

(b) How many **nonempty** aggregated (i.e., non-base) cells a complete cube will contain?

(c) how many **nonempty** aggregated cells an iceberg cube will contain, if the condition of the iceberg cube is "*count* ≥ 2"?

Part 2

There are several typical cube computation methods, such as multiway array computation (Zhao, et al. SIGMOD'1997), BUC (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD'2001), and StarCubing (Xin et al., VLDB'2003).

Briefly describe these three methods (i.e., use one or two lines to outline the key points), and compare their feasibility and performance in the following conditions:

(a) computing dense full cube of low dimensionality (e.g., less than 8 dimensions),

(b) computing iceberg cube around 10 dimensions with highly skewed data distribution, and

(c) computing sparse iceberg cube of high dimensionality (e.g., over 100 dimensions).

Part 3

A cell c is a *closed cell* if there exists no cell d such that d is a specialization of cell c (i.e., d is obtained by replacing a $*$ in c by a non-$*$ value) and d has the same measure value as c. A **closed cube** is a data cube consisting of only closed cells.

(a) How many closed cells in the full cube of Part 1?

(b) Proposed an algorithm that computes closed iceberg cubes efficiently.

Problem 2

Part 1

Decision tree induction is a popular classification method. Taking one typical decision tree induction algorithm C4.5 (Quinlan 1993), briefly outline the method of decision tree classification.

Part 2

There are many classification methods developed in research. However, some of them may not be scalable to very large data sets. RainForest (Gehrke, Ramakrishnan, and Ganti, VLDB'98) and Sprint (Shafer, Agrawal, Mehta, VLDB'96) are two well-known algorithms that perform scalable decision tree induction.

(a) By comparison with C4.5, outline how RainForest builds classification models in very large databases.

(b) Compare and outline the major differences of two scalable decision tree induction algorithms, RainForest and Sprint.

Part 3

The decision-tree induction with RainForest may work well for large database of low dimensionality. However, sometimes we need to build classification models for high-dimensional datasets. Propose a classification method that may work well in a data set of not too small size (e.g., one million tuples) but high dimensionality (e.g., 100 dimensions).

Problem 3

Part 1

There are many cluster analysis methods proposed in statistics, pattern recognition, and data mining research. Name 10 cluster analysis methods you know and group them into a few classes based on their analysis methodology.

Part 2

BIRCH (Zhang, Ramakrishnan, and Livny, SIGMOD'96) and CLARANS (Ng and Han, VLDB'94) are two interesting clustering algorithms that perform effective clustering in large data sets.

(a) Outline how BIRCH performs clustering in large data sets.

(b) Compare and outline the major differences of the two scalable clustering algorithms: BIRCH and CLARANS.

Part 3

Recently, there are many new applications that require us to perform data mining in huge volume of fast evolving data streams. For example, one may like to use cluster analysis method to detect outliers of computer network intrusion by comparing the behavior of current data stream with that of the previous ones. Propose an efficient clustering algorithm that detects such outliers in fast evolving data streams.

3.2 Sample Exam Question Set 3.2

Problem 1

Part 1

Assume a base cuboid of 10 dimensions contains only five (nonempty) cells in the base cuboid. The measure of the cube is *count*. Answer the following questions and provide short reasoning on your answers.

(a) What is the possible **maximum** number of nonempty cells that the whole data cube may contain (excluding the base cuboid)?

(b) What is the possible **minimum** number of nonempty cells that the whole data cube may contain (excluding the base cuboid)?

Part 2

There are three kinds of measures that a data cube may compute: *distributive, algebraic*, and *holistic*.

(a) Give one example for each such category.

(b) Suppose one would like to compute a sales iceberg cube of 10 dimensions, with the following iceberg condition:

$$avg\_price(*) > 50 \wedge count(*) \geq 100$$

Outline an efficient algorithm that may compute such an iceberg cube efficiently.

Part 3

Data cube facilitates online analytical processing of multi-dimensional data. Such kinds of analysis may also be desirable in data streams, where data may flow in and out indefinitely.

(a) What are the major challenges for multi-dimensional analysis of data streams?

(b) Proposed a design that may facilitate multi-dimensional analysis of data streams efficiently.

Problem 2

Part 1

Decision tree induction is a popular classification method. Taking one typical decision tree induction algorithm C4.5 (Quinlan 1993), briefly outline the method of decision tree classification.

Part 2

There are many classification methods developed in research. However, some of them may not be scalable to very large data sets. RainForest (Gehrke, Ramakrishnan, and Ganti, VLDB'98) and Sprint (Shafer, Agrawal, Mehta, VLDB'96) are two well-known algorithms that perform scalable decision tree induction.

(a) By comparison with C4.5, outline how RainForest builds classification models in very large databases.

(b) Compare and outline the major differences of two scalable decision tree induction algorithms, RainForest and Sprint.

Part 3

The decision-tree induction with RainForest may work well for large database of low dimensionality. However, sometimes we need to build classification models for high-dimensional datasets. Propose a classification method that may work well in a data set of not too small size (e.g., one million tuples) but high dimensionality (e.g., 100 dimensions).

Problem 3

Part 1

For effective clustering, different data types may need to use different similarity (distance) measures. For each of the following types of data, present one distance measure that can be used for the generation of high-quality clusters.

(a) numerical data

(b) asymmetric binary data

(c) vector objects (e.g., text documents)

(d) categorical data

Part 2

Micro-clustering has been used for scalable clustering of massive data sets.

(a) Taking one clustering task as an example, explain how micro-clustering can be used for effective clustering of large data sets.

(b) If a data set is partitioned and distributed at multiple sites, propose a k-means like algorithm that can perform efficient and effective clustering of the whole data set without moving all the data sets to one site.

Part 3

In biological data analysis, many data sets, such as microarray data sets, contain a large number (such as thousands) of dimensions.

(a) State what are the major challenges at clustering high-dimensional data?

(b) Propose an efficient algorithm that performs efficient and effective clustering of the high-dimensional data in micro-array data analysis.

3.3 Sample Exam Question Set 3.3

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cubes) List (the names of) four methods that can be used to compute *iceberg cubes* efficiently.

(b) (clustering) List four *clustering* methods, one from each of the following categories: (1) partitioning method, (2) hierarchical method, (3) density-based method, and (4) model-based method.

(c) (classification) List four *classification* methods, one from each of the following categories: (1) scalable decision-tree induction, (2) statistical approach, (3) associative classification, and (4) lazy evaluation approach.

Problem 1

Part 1

A data cube of n dimensions contains exactly p nonempty cells in the base cuboid. Assume that there are no concept hierarchies associated with the dimensions.

(a) What is the *maximum number of nonempty cells* (including the cell in the base cuboid) possible in such a materialized datacube?

(b) If the minimum support is 2, what is the *minimum number of nonempty cells* possible in the materialized iceberg cube?

Part 2

In certain applications, people would like to perform OLAP in a high dimensional data warehouse.

(a) Why do most popular cubing algorithms fail in high-dimensional cubing?

(b) Propose a method that may support efficient OLAP operations in a medium-sized (e.g., tens of mega-bytes) high-dimensional (e.g., 100 dimensions) data set.

Part 3

Outlier detection is an important task in data cube as well. Suppose a cell c is considered as an outlier in a data cube with measure *count* if c's value (i.e., count) is unproportionally high/low in comparison with other cells in its corresponding columns and rows. Suppose only the cells in the base cuboid are available. Outline an outlier detection method in such a data cube.

Problem 2

Part 1

Briefly outline the major differences of the following patterns:

(a) *frequent itemsets* in transaction databases,

(b) (*frequent*) *sequential patterns* in transaction-based sequence databases,

(c) (*frequent*) *protein sequences* in protein sequence databases.

Part 2

PrefixSpan is an efficient sequential pattern mining algorithm for mining transaction-based sequence databases. Suppose one wants to mine sequential patterns with the following constraints. Discuss how to extend the algorithm in each case.

(a) Mining *closed sequential patterns*, where a pattern p is *closed* if there exists no superpattern s in the database that carries the same support as p.

(b) Mining sequential patterns, each containing a user-specified regular expression, such as $a^*\{b^+|X^3\}$, i.e., containing 0 or more occurrences of a, following by one or more occurrences of b *or* three (repeated) occurrences of one single symbol.

Part 3

In some applications, one may like to classify data based on their sequential patterns.

(a) Outline such a classification method, and

(b) Discuss how to improve the *accuracy* and *efficiency* of such a classification method.

Problem 3

Part 1

Scalability is one of the central issues in data mining. When clustering large data sets, one may need to develop scalable algorithms. Outline and compare two methodologies that may lead to the development of high-quality scalable clustering algorithms.

Part 2

In some applications, a large amount of data may flow in and out like streams. Outline an efficient clustering method that may discover the cluster evolution regularities in data streams.

Part 3

One major challenge in frequent itemset mining is the possible generation of a huge number of frequent itemsets in the mining process. *Clustering can be used to compress frequent itemsets and thus dramatically reduce the total number of frequent itemsets to be generated.*

(a) Give an example to show that the above statement is true.

(b) Propose an efficient algorithm that performs effective clustering of frequent itemsets in data mining.

3.4 Sample Exam Question Set 3.4

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cube and OLAP) List (the names of) four methods, each best for the following task: (1) *compute dense cubes with 4-5 dimensions*, (2) *compute iceberg cubes of 7-10 dimensions of highly skewed data*, (3) *facilitate OLAP on high-dimensional data sets*, and (4) *facilitate OLAP on dynamic changing data streams.* (Note: write one-line description if you cannot remember the name of a method.)

(b) (frequent-pattern mining) List four *frequent-pattern mining* methods, each having one of the following features: (1) candidate generation-and-test, (2) vertical data format, (3) depth-first search, and (4) efficient mining of closed frequent patterns.

(c) (classification) Use one sentence for each method to characterize and distinguish the following four *classification* methods: (1) Rainforest, (2) SVM, (3) CBA, and (4) KNN.

Problem 1

Part 1

Assume a base cuboid of n dimensions contains n base cells, as shown below,

$$\begin{array}{ll} (1) & (b_1, a_2, a_3, \ldots, a_n) \\ (2) & (a_1, b_2, a_3, \ldots, a_n) \\ (3) & (a_1, a_2, b_3, \ldots, a_n) \\ \cdots & \qquad \cdots \\ (n) & (a_1, a_2, a_3, \ldots, b_n) \end{array}$$

where $a_i \neq b_i$ (for any i). The measure of the cube is *count*. Let the condition of iceberg cube be "*count* ≥ 2"?

1. How many **nonempty** aggregated cells are there for all the iceberg cuboids of dimension $(n-2)$?
2. Answer the same question for the iceberg cuboids of dimension $(n-3)$.

Part 2

For network intrusion detection, one may like to detect substantial changes of certain measures of network data flow in multidimensional space. Design a data structure and computation method that may facilitate multi-dimensional analysis of measure changes in data streams.

(a) Show your design requires limited space and can incrementally incorporate new incoming data streams, and

(b) show how to perform efficient online detection of substantial changes in multi-dimensional space.

Part 3

A chain store has collected data on the *effectiveness* of its promotion of certain new products in state A and found that the effectiveness is closely related to certain sensitive attributes in each region, such as (1) population density, (2) income level, (3) race distribution, (4) age distribution, (5) education level, (6) sales season, and (7) professional distribution. Discuss how the store can use such data to facilitate its effective promotion of the same new products in other states.

Problem 2

Part 1

Someone says: "*The worst-case complexity of the existing frequent-pattern mining algorithms is exponential. Such algorithms only work at certain special conditions on special kinds of data.*" Give sufficient and clear arguments to support or rebut the statement.

Part 2

Frequent-pattern mining algorithms can be extended to perform effective classification,.

(a) Outline such an efficient frequent-pattern-based classification algorithm.

(b) In comparison with a classical Naïve-Bayesian classification algorithm, what are the strength and weakness of the algorithm you outlined in Part 2(a)?

Part 3

One often needs to mine frequent itemsets in data streams. However, due to the limited main memory space, it is often only realistic to mine *approximate* frequent itemsets in data streams.

(a) Outline one efficient algorithm that used limited space to mine approximate frequent itemsets in long data streams, with a guaranteed error bound, and

(b) Discuss how such an algorithm can be extended to find the *evolution* (with time) of approximate frequent itemsets in data streams.

Problem 3

Part 1

(a) Explain why a typical partition-based clustering algorithm, such as k-means algorithms, have difficulty to find arbitrary-shaped clusters.

(a) Outline an algorithm that finds such arbitrary-shaped clusters efficiently.

Part 2

In micro-array data analysis, each data object may contain tens of thousands of dimensions. Outline a method that can efficiently perform cluster analysis in such data sets.

Part 3

A typical data relation, such as *Student*, may have dozens of attributes. Not all the attributes are relevant to a particular clustering task. A user may not know what are the relevant attributes but may provide one or two known relevant attributes, such as *Research_group*, as a hint.

(a) Outline an efficient clustering method that may take user's attribute hint to perform effective clustering of relational data.

(b) Discuss how such a clustering method should be changed if a user gives a hint on a sample set of data (such as tuples A and B should be in the same cluster but not C) instead of a sample set of attributes.

3.5 Sample Exam Question Set 3.5

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cube and OLAP) Briefly point out three major differences between the following two data cube computation methods: (1) BUC, and (2) MultiWay Array Cube computation.

(b) (sequential-pattern mining) Name three sequential pattern mining algorithms, one from each of the following categories: (1) candidate generation-and-test, (2) vertical data format, and (3) pattern-growth approach.

(c) (clustering) List the names of four algorithms that can perform effective clustering, each from the following categories: (1) partitioning method, (2) density-based method, (3) high-dimensional clustering, and (4) hierarchical clustering.

Problem 1

Part 1

Assume a base cuboid of 20 dimensions contains 3 base cells, as shown below,

$$\begin{array}{ll} (1) & (a_1, a_2, a_3, \ldots, a_{20}) \\ (2) & (a_1, b_2, b_3, \ldots, b_{20}) \\ (3) & (a_1, a_2, b_3, \ldots, b_{20}) \end{array}$$

where $a_i \neq b_i$ (for any i). The measure of the cube is *count*. Let the condition of iceberg cube be "$count \geq 2$"?

(a) How many nonempty cuboids will this full data cube contain?

(b) How many **nonempty** aggregated cells are there for the (full) data cube?

(c) How many **nonempty** aggregated cells are there for the iceberg cube?

Part 2

Data warehouse has been used in many applications. Suppose one wants to record the changes of the climate in a country so that one can check how the temperature changes in the last 10 years by region, by month, etc., and drill down to any particular region, month, etc.

(a) Design such a data warehouse by drawing star schema with concise explanation, and

(b) show how to efficient compute data cubes and perform OLAP operations in such a data warehouse.

Part 3

In some applications, one may need to perform OLAP analysis in high dimensional databases, such as over 50 dimensions.

(a) Present one such application example,

(b) based on the example you give, show how to efficient implement OLAP operations in such a high-dimensional space, and

(c) can you extend your implementation to handle data streams? If your answer is "yes", how to do it? and if "no", why not?

Problem 2

Part 1

Someone says: "*In a transaction database, two items that are strongly associated, such as $a \rightarrow b[s, c]$, with high s and c, can still be strongly positively correlated, independent or strongly negatively correlated.*"

If you agree with the statement, present one example for each case.

If you disagree with it, present your reasoning.

Part 2

A frequent-pattern mining algorithm may generate a large set of frequent patterns, under a low support threshold.

(a) Outline an efficient compressed frequent-pattern mining algorithm that mines the set of compressed frequent patterns directly, and

(b) which set of frequent patterns, the set of precise frequent patterns or the set of compressed frequent patterns, may lead to better accurate classification model?

Part 3

One often needs to mine frequent itemsets in data streams. However, due to the limited main memory space, it is often only realistic to mine *approximate* frequent itemsets in data streams.

(a) Outline one efficient algorithm that used limited space to mine approximate frequent itemsets in long data streams, with a guaranteed error bound, and

(b) Discuss how such an algorithm can be extended to find the *evolution* (with time) of approximate frequent itemsets in data streams.

Problem 3

Part 1

(a) Explain why a typical partition-based clustering algorithm, such as k-means algorithms, have difficulty to find arbitrary-shaped clusters.

(a) Outline an algorithm that finds such arbitrary-shaped clusters efficiently.

Part 2

In micro-array data analysis, each data object may contain tens of thousands of dimensions. Outline a method that can efficiently perform cluster analysis in such data sets.

Part 3

A typical data relation, such as *Student*, may have dozens of attributes. Not all the attributes are relevant to a particular clustering task. A user may not know what are the relevant attributes but may provide one or two known relevant attributes, such as *Research_group*, as a hint.

(a) Outline an efficient clustering method that may take user's attribute hint to perform effective clustering of relational data.

(b) Discuss how such a clustering method should be changed if a user gives a hint on a sample set of data (such as tuples A and B should be in the same cluster but not C) instead of a sample set of attributes.

3.6 Sample Exam Question Set 3.6

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cube and OLAP) An n-dimensional data cube, with the measure *count*, has p nonempty cells in its base cuboid (where $p > n > 1$). (1) What is the maximal number of (non-empty) 2-D cuboid in the cube? (2) What is the maximum possible number of aggregate cells in the cube? (3) what is the minimum possible number of aggregate cells in the cube?

(b) (selection of data mining algorithms) Name one data mining algorithm that best fits each of the following applications: (1) detecting network intrusions with the availability of some previously recorded intrusion history, (2) finding hidden animal shapes in a multi-colored mosaic plate, and (3) finding when *wine* is selling well in the shop, so is *cheese*.

(c) (data mining applications) List one good application example for each of the following algorithms: (1) KNN (*i.e.*, K-nearest neighbor), (2) EM, (3) SVM.

Problem 1

Part 1

Data cube computation algorithms have been developed for efficient computation of multidimensional aggregations for low dimensional data (such as less than 10 dimensions).

(a) Explain why a typical cubing algorithm, such as BUC or StarCubing, encounters difficulties at computing high-dimensional data cubes, such as 50-100 dimensions, and

(b) outline an efficient algorithm that can perform fast OLAP operations in such a high-dimensional space with not so huge data sets (such as 10^5 tuples).

Part 2

Data warehouse has been used in many applications. Suppose one wants to trace the shipping of goods for a shipping company such as UPS so that not only customers but also company managers can check the situations or summaries, by region, by goods category, by month, etc., and drill down to any particular region, day, etc.

(a) Design such a data warehouse by drawing star schema with concise explanation, and

(b) show how to efficient compute data cubes and perform OLAP operations in such a data warehouse.

Part 3

Although data cube models can be used for computing multidimensional aggregations for numerical values, the multidimensional model can be used for modeling and prediction as well.

(a) Explain how to extend the multidimensional data cube model for prediction analysis,

(b) explain how such a cube may facilitate drilling down or rolling up in multidimensional space, and

(c) outline the design of an "outlier cube" that facilitates outlier analysis in multidimensional space?

Problem 2

Part 1

(a) Give an example to show that two strongly associated itemsets, X and Y, may not be strongly correlated, and

(b) give another example to show that the commonly used *lift* measure, $\textit{lift}(X, Y) = \frac{P(X,Y)}{P(X)P(Y)}$, may not be a good measure for correlation analysis in transactional databases, and

(c) propose a better measure for it.

Part 2

It is often useful to mine frequent itemsets in data streams.

(a) What are the major challenges for mining frequent itemsets in data streams?

(b) Explain why lossy counting algorithm proposed by Manku and Motwani (2002) guarantees an error bound at mining approximate frequent itemsets in data streams, and

(c) Outline a method that may find approximate correlation patterns in data streams.

Part 3

Frequent-patterns have been used for effective classification.

(a) Explain why discriminative frequent patterns may achieve better classification accuracy than (1) the complete set of frequent patterns, and (2) many other typical classification methods, and

(b) The method proposed by Hong et al. (2007) for pattern-based classification is to first mine the full set of closed frequent patterns and then extract a set of discriminative frequent patterns for effective classification. Proposed a more efficient method that mines a similar set of patterns directly.

Problem 3

Part 1

(a) Explain why a simple hierarchical clustering algorithm, such as AGNES, often generates low quality clusters, and

(b) explain why two recently proposed hierarchical algorithms, BIRCH and CHAMELEON, generate high quality clusters.

Part 2

(a) What are the major challenges in high-dimensional clustering?

(b) outline a method that efficiently clusters micro-array data sets that contain high-dimensional numerical data, and

(c) propose a method that efficiently clusters text documents based on the sets of terms contained in those documents.

Part 3

Most objects are linked to each other via various kinds of semantic links (or relationships).

(a) Explain how such links can be used for effective cluster analysis, and

(b) outline a scalable method that performs effective clustering using multiple kinds of links.

3.7 Sample Exam Question Set 3.7

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cube and OLAP) A data cube may contain different measures which can be categorized as *distributive*, *algebraic*, and *holistic*. Which category each of the following belongs to: (1) standard-deviation; (2) top-k for $k \leq 10$; and (3) Q1 (*i.e.*, 25 percentile))?

(b) (selection of data mining algorithms) Name one data mining algorithm that best fits each of the following applications: (1) finding strange moving cars in a highway network, (2) predicting which category that a new webpage should belong to if you know the categories of some existing web pages, and (3) determining whether the value of a particular stock is too high, if you know the history of many stocks.

(c) (data mining applications) List one good application example for each of the following algorithms: (1) density-based clustering, (2) frequent-pattern based classification, and (3) Hidden-Markov Model (HMM).

Problem 1

Part 1

Different data sets may require different cube computation algorithms. Outline one algorithm that is most suited for computing each of the following data cubes

(a) a data cube with 5 dimensions, dense data, and 10^7 rows

(b) an iceberg cube with 10 dimensions, sparse data, and 10^6 rows, and

(b) a structure that may support OLAP operation for 100 dimensions, sparse data, and 10^5 rows.

Part 2

Multidimensional data modeling is essential at designing data warehouses and data cubes. Although people have saved e-mails in tree-structured mail directories, it is more desirable to construct an e-mail data warehouse so that e-mails can be searched in multi-dimensions, such as based on sender (i.e., from_list), recipients (to_list), sending/receiving time, topic, keywords in title, length, attachment, keywords in the body, and so on.

(a) Design such an e-mail data warehouse by drawing star schema, and

(b) outline how such a data warehouse can be implemented efficiently.

Part 3

One may like to construct data cube for data stream for certain applications.

(a) Give a few typical applications of such a stream data cube,

(b) based on one such application, outline how such a stream data cube can be implemented efficiently, and

(c) if the stream data is high dimensional, can you work out an efficient implementation method? If yes, outline the method. If not, what is the major difficulty?

Problem 2

Part 1

(a) Discuss why "*null-(transaction) invariance*" is an important property at measuring pattern interestingness in large transaction databases.

(b) What should be the best measure for justifying whether a set of items are strongly correlated in a large set of transactions? Why?

Part 2

(a) What is the worse-case computational complexity of mining frequent patterns? Why do many good frequent pattern mining algorithm claim that it can usually find the complete set frequent patterns efficiently under a reasonable *min_support* threshold?

(b) Why that a typical frequent pattern mining algorithm encounters difficulty at finding rather large patterns (such as of size 100)? Can you propose a method that may find such pattern efficiently?

Part 3

(a) Sequential pattern mining has been used for studying customer shopping behavior. Illustrate three major kinds of sequent pattern mining methods.

(b) To derive desired results, a user may like to enforce constraints on sequent patterns. Discuss how to categorize constraints, and how to push categories deeply into the mining process.

Problem 3

Part 1

(a) What are the major difficulties of classifications in stream environment vs. nonstream environments?

(b) It is often desirable to predict rare events, such as computer network intrusions in a data stream environment. Discuss how to perform high quality classification for such rare events in data streams.

Part 2

(a) Decision-tree is a popular classification method. However, when the training data is too huge to fit in main memory, a typical decision-tree induction may have to do a lot of I/O operations. Present a highly scalable method for decision tree induction.

(b) Similarly, present a highly scalable method for support vector machine (SVM) induction.

Part 3

(a) The method proposed by Hong et al. (2007) for pattern-based classification is to first mine the full set of closed frequent patterns and then extract a set of discriminative frequent patterns for effective classification. Explain why this method may lead to high classification accuracy.

(b) Can you extend the method for classification of sequential patterns? Outline your proposed method.

3.8 Sample Exam Question Set 3.8

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cube and OLAP) A 10-dimensional data cube, with the measure *count*, has only 2 nonempty cells in its base cuboid. (1) What is the maximum possible number of aggregate cells in the cube? (2) what is the minimum possible number of aggregate cells in the cube? (3) if the minimum support (*i.e.*, *iceberg condition*) is 2, what is the minimum possible number of aggregate cells in the *iceberg cube*?

(b) (classification) Use one sentence to distinguish the following pairs of methods: (1) SVM vs. neural network algorithms, (2) decision-tree (such as C4.5) vs. RAINFOREST algorithms, (3) boosting vs. ensemble methods.

(c) (selection of data mining algorithms) Name or outline one data mining algorithm that best fits each of the following applications: (1) capturing outliers in computer network streams, (2) partitioning high-dimensional microarray data sets into meaningful groups, and (3) promoting the sales of a set of items to a set of known customers based on the history of shopping transaction sequences.

Problem 1

Part 1

Suppose that a big chain-store has one data center (*e.g.*, a data cube) in each state, and the data is updated daily. The managers of the store would like to know the sum and (standard) deviation of the sales by region (such as mid-west vs. north-east), by month, *etc.*. For each of the following requests, outline an efficient computation method.

(a) Both sum and deviation measures of each local cube can be updated efficiently in an incremental manner, and (*1 point*)

(b) both measures of each region can be derived dynamically from the measures stored in the local cubes of the corresponding states. (*1 point*)

Part 2

Multidimensional data modeling is essential at designing data warehouses and data cubes. Suppose each product carries an RFID (Radio-Frequency Identification) tag, and will traverse from a factory/farm to a distribution center, then to a store, a shelf, and a check-out counter, with information periodically scanned and stored.

(a) Why do people claim that such RFID data contains a lot of redundancy? (*1 point*)

(b) Design a data warehouse that may reduce much redundant information and facilitate OLAP on RFID data, and (*1.5 points*)

(b) outline how such a data warehouse can facilitate path-based data mining, such as find why particular packages of milk got rotten this week. (*1.5 points*)

Part 3

Most data cube algorithms perform efficient computation for large set of data in low dimensional space.

(a) Give a few typical applications where high-dimensional OLAP is needed, (*1 point*)

(b) Outline one method that may perform high-dimensional OLAP efficiently, and (*1.5 points*)

(c) If one only would like to find top-k measures for a small k, can you work out an efficient method to compute it? (*1.5 points*)

Problem 2

Part 1

(a) Give an example to show why χ^2 and lift may not be good measures for pattern interestingness in large transaction databases. (*1 point*)

(b) Present a good measure for mining interesting patterns in large transaction databases, and justify why this is a good measure. (*1 point*)

Part 2

(a) What are the major difficulties for mining frequent patterns in data streams? (*1 point*)

(b) Given a fix amount of memory M, present an efficient algorithm that makes the best use of the available memory and mines frequent patterns in data streams with small error bound. (*2 points*)

Part 3

(a) Explain why frequent-pattern-based classification algorithms may lead to better classification quality than traditional classification methods? (*1 point*)

(b) Explain why discriminative frequent patterns may lead to better better classification quality than typical associative classifier. (*1 point*)

(c) Outline an efficient discriminative frequent pattern-based classification method. (*2 points*)

Problem 3

Part 1

(a) Use one sentence to distinguish the following pairs of clustering methods: (1) k-means vs. EM clustering algorithms, (2) DBSCAN vs. OPTICS, (3) AGNES (*i.e.*, Agglomerative nesting) vs. DIANA (*i.e.*, Divisive analysis). (*1.5 point*)

(b) Explain why a typical hierarchical clustering algorithm has difficulty to derive high-quality clusters, but BIRCH does not have such weakness. (*1.5 point*)

Part 2

(a) What is the major challenge of clustering high-dimensional data? (*1 point*)

(b) Give two algorithms that perform high-dimensional clustering and compare their clustering quality and efficiency. (*1 point*)

(c) In micro-array data analysis, each data object may contain tens of thousands of dimensions. Which one would you select from your above two algorithms, and why? (*1.5 points*)

Part 3

(a) A user may often like to give some hints to a clustering task. Illustrate why such a hint may lead to more desirable clustering result. (*1 point*)

(b) There are two kinds of hints: one is to specify certain set of objects either must be or cannot be in the same cluster, the other is to use one attribute (such as *Research_group*, as a hint, to cluster a data set (such as *Students*). Discuss which method is more desirable, and why. (*1 point*)

(c) A large bibliographic information network links authors, conferences, and research publications. Outline an effective method that performs user-guided clustering of certain types of entities (such as authors or conferences). (*1.5 points*)

3.9 Sample Exam Question Set 3.9

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data preprocessing) Name **four** typical methods that are effective in *dimensionality reduction*. And name **four** typical methods that are effective in *numerosity reduction* (i.e., reducing the large amount of data to a smaller amount).

(b) (data mining methods) Use one sentence to distinguish the following pairs of methods: (1) *PrefixSpan* vs. *gSpan* algorithms, (2) *k-means* vs. *k-medoids* algorithms, (3) *decision tree induction* vs. *rule induction.*

(c) (selection of data mining methods) Name or outline one data mining method that best fits each of the following tasks: (1) *finding snakes in bushes*, (2) *finding computer network anomalies in real time*, and (3) *determining whether a (muscle) tissue is cancerous.*

Problem 1

Part 1

Suppose the base cuboid of a data cube contains only two cells

$$(a_1, a_2, a_3, \ldots, a_{10}), (b_1, b_2, b_3, \ldots, b_{10}),$$

where $a_i = b_i$ if i is an odd number; otherwise $a_i \neq b_i$.

(a) How many nonempty cuboids are there in this data cube? (*1 point*)

(b) How many nonempty aggregate cells are there in this data cube? (*1.5 point*)

(c) If we set minimum support = 2, how many nonempty aggregate cells are there in the corresponding iceberg cube? (*1 point*)

Part 2

Multidimensional analysis is essential for studying many data sets. We want to design and build a "data cube" for multidimensional analysis of DBLP (digital bibliographic library project for computer science publications) database for powerful and flexible analysis. Notice that DBLP contains entries for almost every recognized CS research conference or journal publication entries, with the information like authors, paper title, publication venue, location, and time.

(a) What should be the dimensions and measures for such a data cube? (*1 point*)

(b) What analytical functions you can provide, and (*1 points*)

(b) What are the major challenges in implementation and how would you propose to handle them? (*1.5 points*)

Part 3

Most data cube build on the whole population of data. However, in many cases the collected data are just samples (such as surveys). A major problem for such data is when drilling down, some cells could contains empty or few data for reliable analysis.

(a) Design a sampling cube such that reliable prediction can still be done despite some cells contains very few or empty data values, (*1.5 point*)

(c) Discuss how such a cube can handle high-dimensional OLAP. (*1.5 points*)

Problem 2

Part 1

(a) List five major challenges at clustering different kinds of data. (*1.5 point*)

(b) Outline an efficient algorithm that can effectively cluster (high-dimensional) micro-array data. (*1.5 point*)

Part 2

(a) Links contain rich information for effective data mining. Take DBLP data as an example, illustrate how links can be used for effective clustering. (*1 points*)

(b) Is the method you described efficient in large databases? Outline a method that is scalable and efficient in cluster analysis for large sets of linked data. (*1.5 points*)

(c) In real datasets, data may not be clean (e.g., same person may have different names and different people may share the same name). Outline a method so that link-based clustering can still work well with the existence of such data. (*1.5 points*)

Part 3

(a) Information networks have become an important target in data mining. Information networks themselves may often need to be clustered. Give an example and convincing argument that such clustering may lead to the discovery of interesting knowledge. (*1.5 point*)

(b) Outline one effective method for clustering information networks (*1.5 point*)

Problem 3

Part 1

(a) Frequent pattern mining has been studied extensively in data mining research. However, it is not easy to mine large patterns (such as pattern size ≥ 100) in large data sets. What are the difficulties at mining such large patterns? (*1.5 point*)

(b) Can you design an effective method that mines frequent large patterns effectively in large datasets? (*1.5 point*)

Part 2

(a) Frequent pattern methods are designed for mining precise patterns. However, the real world patterns are largely approximate. What are the major difficulties at mining approximate frequent patterns? (*1 point*)

(b) Design an efficient algorithms that can mine approximate patterns efficiently. (*2 point*)

Part 3

(a) Frequent patterns have been used in classification. What are the major problems in traditional frequent-pattern-based classification methods? (*1.5 point*)

(b) Outline an efficient method that can perform efficient discriminative frequent pattern-base classification. (*1.5 point*)

(c) Explain why the above describe method leads to high efficiency and high accuracy in large datasets (*1 points*)

3.10 Sample Exam Question Set 3.10

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (Measures in similarity computation) It is good to select appropriate measures in computing similarity or distances among objects. Give the most appropriate measure for the following computation: (1) correlation between items in a transactional dataset, (2) finding clusters in a large connected graph, and (3) finding similar conferences in the DBLP network.

(b) (Data mining methods) Use one sentence to distinguish the following pairs of methods: (1) EM vs. k-means, (2) SPADE vs. PrefixSpan, (3) Hidden Markov Model (HMM) vs. Bayesian Believe Networks (BBN).

(c) (Selection of clustering algorithms) Name or outline one clustering algorithm that best fits each of the following applications: (1) user-guided clustering across multiple relations, (2) clustering evolving, dynamic data streams, and (3) clustering microarray data with over 10000 dimensions.

Problem 1

Part 1

Data cube has become an essential information component. Reason on the correctness or incorrectness of each of the following statements.

(a) Computing iceberg cube (with iceberg condition $count \geq 2$) costs less than computing the full data cube for the following algorithms: (i) Multi-Way Array Cubing, (ii) BUC, and (iii) StarCubing. (*1.5 point*)

(b) If one constructs a datacube on sampling data (*e.g.*, survey data), one cannot support quality drill-down because some low-level cells could contain empty or too few data for reliable analysis. (*1.5 point*)

Part 2

It is desirable to construct an AlbumCube to facilitate multidimensional search through digital photo collections, such as by date, photographer, location, theme, content, color, etc.

(a) What should be the dimensions and measures for such a data cube? (*1 point*)

(b) What analytical functions you can provide, and (*1 points*)

(c) What are the major challenges on implementing AlbumCube, and how would you propose to handle them? (*1.5 points*)

Part 3

One may expect to perform multidimensional analysis of traffic data on highways based on the traffic history to identify what segments are likely jammed at what conditions, *e.g.*, during rush hours or bad weather.

(a) What should be the dimensions and measures for such a data cube? (*1 point*)

(b) What analytical functions you can provide, and (*1 points*)

(c) What are the major challenges on implementing TrafficCube, and how would you propose to handle them? (*1.5 points*)

Problem 2

Part 1

(a) Suppose a dataset contains only 5 transactions as follows:

$\{a, b, c, d\}$: 2
$\{a, c, e\}$: 3

Let the minimum support be 3. Write out (together with the support information) all the (i) frequent patterns, (ii) closed patterns, and (iii) max-patterns. (*1.5 point*)

(b) Present one example to show *cosine* is a better measure than *lift* at disclosing interesting patterns in large transaction databases. (*1.5 point*)

Part 2

(a) What is the worse-case computational complexity of mining sequential patterns? Why do many good sequential pattern mining algorithm claim that it can usually find the complete set of sequential patterns efficiently under a reasonable *min_support* threshold? (*1 point*)

(b) Show max-gap constraint is antimonotonic for sequential pattern mining. Discuss how to refine a typical sequential pattern mining algorithm, such as PrefixSpan, to push "*max-gap = 5*" deep into the mining process. (*1 point*)

(b) Why that a typical sequential pattern mining algorithm encounters difficulty at finding rather long patterns (such as of size 100)? Can you propose a method that may find such patterns efficiently? (*2 point*)

Part 3

(a) Explain why discriminative frequent patterns may lead to better better classification quality than a typical associative classifier. (*1 point*)

(c) DDPMine is an efficient discriminative, frequent pattern-based classification method. However, it does not perform well with high-dimensional data. Explain why. Can you propose a discriminative, frequent pattern-based classification method that is likely to perform well with high-dimensional data? (*2 points*)

Problem 3

Part 1

(a) Explain why RAINFOREST is more scalable than C4.5 in decision tree induction in large datasets. (*1.5 point*)

(b) Explain why SVM performs well on high-dimensional data but does not perform well on large volume of data. (*1.5 point*)

Part 2

(a) What are the major challenges of classifying dynamic evolving data streams? (*1 point*)

(b) What are the major challenges of classifying highly skewed data sets? (*1 point*)

(c) Outline a method that effectively classifies highly skewed data sets in dynamic evolving data streams. (*1 point*)

Part 3

(a) Take two typical classification methods, k-nearest neighbor (KNN) and naïve Bayes as examples, explain what are the major differences between lazy classification and eager classification methods. (*1 point*)

(b) Extend the KNN classification method to make it work well in data stream environment. (*1.5 point*)

(c) Many classification tasks assume all of the training data are labeled. In reality, only a small set of training data are (positively) labeled (*e.g.*, a small set of webpages are given class labels). Extend the KNN classification method to make it effective for partially labeled data. (*1.5 points*)

3.11 Sample Exam Question Set 3.11

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data cube and OLAP) Measures in a data cube can be categorized as *distributive*, *algebraic*, and *holistic*. Which category each of the following belongs to: (1) top-k for $k \leq 10$; (2) z-score (for a cell in a cuboid); and (3) inter-quartile range (*i.e.*, $Q_3 - Q_1$)?

(b) (data mining methods) Use one sentence to distinguish the following pairs of methods: (1) *FPGrowth* vs. *DDPMine*, (2) *SCAN* vs. *DBSCAN*, (3) *RAINFOREST* vs. *C4.5*.

(c) (selection of data mining methods) Name or outline one data mining method that best fits each of the following tasks: (1) *recommendation of particular items to a group of customers based on their shopping history*, (2) *predicting which category a webpage should belong to based on the webpage structures*, and (3) *clustering dynamically evolving data streams.*

Problem 1

Part 1

Suppose the base cuboid of a data cube contains only two distinct cells, with *count* shown after ":"

$$(1)\ (a_1, a_2, b_3, \ldots, b_{10}) : 5, \quad (2)\ (b_1, b_2, b_3, \ldots, b_{10}) : 10,$$

where $a_i \neq b_i$.

(a) How many nonempty cuboids are there in this data cube? (*1 point*)

(b) How many nonempty aggregate cells are there in the iceberg cube with minimum support = 5? (*1.5 points*)

(c) A cell c is *closed* if there exists no descendant of c having the same support. How many distinct closed cells are there in this data cube? (*1 point*)

Part 2

Multidimensional analysis is essential for studying many data sets. We want to design and build a "data cube" for multidimensional analysis of *moving object data*, such as birds and deers.

(a) What should be the dimensions and measures for such a data cube? (*1.5 points*)

(b) What are the new challenges in comparison with a traditional data cube and how would you propose to handle them? (*2 points*)

Part 3

It is desirable to perform data mining in cube space. Suppose one would like to build a multidimensional model for prediction.

(a) Explain how to extend the multidimensional data cube model for classification analysis, (*1.5 points*)

(b) If someone would like to use ensemble technique to integrate multiple classification methods, discuss how to design such a classification cube to facilitate such an ensemble process. (*1.5 points*)

Problem 2

Part 1

(a) Information networks can be categorized into *homogeneous* vs. *heterogeneous* information networks. Give examples of on each kind to illustrate their differences. (*1 point*)

(b) Give one example to explain why an analysis method on *homogeneous* information networks may not be readily applicable to that of *heterogeneous* information networks. (*1.5 points*)

Part 2

(a) SCAN is an algorithm that clusters *homogeneous* information networks. Illustrate the method and explain why it may lead to efficient and effective network clustering (*1 point*)

(b) Design an algorithm that extends SCAN to perform effective clustering over *heterogeneous* information networks? (*2 points*)

(c) Give one good example that such an extended algorithm could lead to some interesting applications. (*1 point*)

Part 3

(a) RankClus is an interesting algorithm that integrates ranking and clustering for clustering *heterogeneous* information networks. Explain why such an integration may lead to both high-quality ranking and high-quality clustering. (*1.5 points*)

(b) Give an example that ranking rules may influence the final results of Rankclus. Explain how such rules may influence the RankClus process. (*1 point*)

(c) RankClus works on a *heterogeneous* information network consisting of two types. Can you extend the method to make it work on a network consisting of more than two types? (*1.5 points*)

Problem 3

Part 1

(a) Frequent patterns have been used in classification. What are the major problems of a typical frequent-pattern-based classification method, such as CBA (Classification Based on Association)? (*1 point*)

(b) Outline a method that can perform efficient discriminative frequent pattern-base classification. (*1.5 points*)

(c) If the dataset is a large set of sequence data (e.g., customer shopping sequences), how should you extend this method to construct a high quality classifier? (*1.5 points*)

Part 2

(a) Classification can be performed on multiple interconnected relations. Outline an effective classification method in such an environment. (*1.5 points*)

(b) Can you extend the methods of multi-relational mining to heterogeneous information networks where the network links may not always correspond to key-foreign_key relationships? (*1.5 points*)

Part 3

(a) Classification is usually done for two or a small number of distinct class labels. What are the major difficulties for classifying data with a high number (say, hundreds) of classes? (*1.5 points*)

(b) Can you design a high-quality classifier for a high number of classes (*2 points*)

3.12 Sample Exam Question Set 3.12

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (Interestingness measure in pattern mining) (1) why is the support-and-confidence framework not sufficient at mining interesting patterns? (2) why are χ^2 and *lift* not good measures either? and (3) what is a good measure for mining interesting patterns?

(b) (data mining methods) Use one sentence to distinguish the following pairs of methods: (1) *K-means* vs. *KNN*, (2) *PrefixSpan* vs. *SPADE*, (3) *CBA* vs. *DDPMine.*

(c) (selection of data mining methods) Name or outline one data mining method that best fits each of the following tasks: (1) *finding clusters, hubs, and outliers in an interconnected network*, (2) *classification of a dynamic data stream with skewed data (i.e., a small portion of positively labeled data)*, and (3) *mining compressed patterns in a transactional database.*

Problem 1

Part 1

Suppose there are following three customer shopping sequences:

(s_1) $\langle a_1, a_2, a_3, \ldots, a_{10}, c_{11}, c_{12} \rangle$

(s_2) $\langle b_1, b_2, b_3, \ldots, b_{10}, c_{11}, c_{12} \rangle$

(s_3) $\langle a_1, b_2, a_3, b_4, \ldots, a_9, b_{10}, c_{11}, c_{12} \rangle$

where $a_i \neq b_i$, and the sequence (s_1) $\langle a_1, a_2, a_3, \ldots, a_{10}, c_{11}, c_{12} \rangle$ means that customer s_1 first bought a_1, then a_2, ..., and finally bought c_{12}. Let $min\_support = 2$

(a) How many (nonempty) sequential patterns are there in this sequence database? (*1.5 point*)

(b) How many (nonempty) closed sequential patterns are there in this sequence database? (and list all of them) (*1 points*)

(c) How many (nonempty) maximal sequential patterns are there in this sequence database? (and list all of them) (*1 point*)

Part 2

Suppose that each customer s_i in the sequence database of Part 1 is associated with multidimensional information, such as location, profession, age, etc., and each item is also associated with multidimensional information, such as item_category, brand, price_range, etc.

(a) Give an example to show what a multidimensional sequential pattern may look like in such a sequence database. (*0.5 point*)

(b) Outline a method that may mine multidimensional sequential patterns in such a sequence database. (*2 points*)

Part 3

Consider the same kind of sequence database as shown in Part 1.

(a) If the sequential pattern to be discovered is long (*e.g.*, over 100 in length), outline a method that mines long sequential patterns efficiently. (*2 points*)

(b) Class labels (such as *valuable customer*, etc.) can be associated with customer shopping sequences. Outline an effective sequential pattern-based classification method with such datasets. (*2 points*)

Problem 2

Part 1

(a) Data in an information network can be *noisy*, *inconsistent* and *erroneous*. Given an example to illustrate each of the above three cases. (*1 point*)

(b) On the other hand, data in an information network can be used to help improve the quality of information. Present one method that can effectively improve the quality of information by information network analysis. (*2 points*)

Part 2

(a) Clustering can be used to partition a set of objects into multiple groups based on certain similarity measure. What are the major differences in similarity measures for the following clustering methods? (*1.5 points*)

(i) EM, (ii) DBSCAN, and (iii) p-clustering

(b) Taking bibliographic database as an example, compare *SimRank*, *RankClus*, and *NetClus* on their clustering *effectiveness* and *efficiency* in a heterogeneous information network (*1.5 points*)

Part 3

(a) Suppose Facebook collects information about a large set of people and their friends and would like to develop a classification method that will recommend new friends for linking. Can you design a good method for it? (*2 points*)

(b) For a bibliographic database, such as PubMed, one may also like to develop an effective method to classify venues, authors, and papers based on some training data (such as research fields). Can you design such a classification method and state what are the major differences between this design from that of the Facebook problem? (*2 points*)

Problem 3

Part 1

(a) A cyber-physical network is a network that links physical sensors and information entities together to form an integrated network. Give two examples of such cyber-physical networks. (*1 point*)

(b) Sensor data in a cyber-physical network could contain noisy and erroneous data. Outline one method that may help improve the quality of such a network. (*2 points*)

Part 2

(a) One may like to design a data cube to model such a cyber-physical network data in a multidimensional space. Can you design such a data cube? (*1.5 points*)

(b) Discuss how such a data cube can be implemented efficiently, even when new data are added into the system incrementally. (*2 points*)

Part 3

(a) One may like to cluster a cyber-physical network based on user's guidance. Discuss how you would develop such a clustering method. (*1.5 points*)

(b) If new data are input into the cyber-physical network in the form of data streams, discuss how you would develop a stream clustering method that can handle such a situation. (*2 points*)

3.13 Sample Exam Question Set 3.13

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (Pattern mining algorithms) (1) Name three algorithms that mine *frequent closed itemsets*; and (2) name three algorithms that mine *sequential patterns.*

(b) (data mining and data cube methods) Use one sentence to distinguish the following pairs of methods: (1) *SCAN* vs. *DBSCAN*, (2) *PageRank* vs. *SimRank*, (3) *BUC* vs. *StarCubing.*

(c) (selection of data mining methods) Name or outline one data mining method that best fits each of the following tasks: (1) *finding very large patterns in a transaction database*, (2) *performing decision tree induction in a very large database*, and (3) *subspace clustering on data sets with rather high dimensions.*

Problem 1

Part 1

In many applications (such as surveys), the collected data are samples instead of the entire population. If one wants to construct a data cube on such data to facilitate OLAP analysis.

(a) What is the major problem of such data if one wants to ensure high quality OLAP analysis? (*0.5 point*)

(b) Design a sampling cube so that reliable prediction can still be performed in such datasets (*1 point*)

(c) If the data contains many attributes (*e.g.*, around 50-100 attributes), discuss how to extend the design to support efficient high-dimensional OLAP. (*1.5 points*)

Part 2

To support clouding computing, one may like to set up a network of hundreds of servers. However, to efficiently use computer resources and power, one may like to design a *performance cube* that may store the history of system performance to facilitate analysis of such data in multidimensional space.

(a) Discuss what should be the dimensions and measures of such a performance cube that may facilitate flexible OLAP analysis. (*1 point*)

(b) What kind of interesting anomalies can one find in such a performance cube? (*1 point*)

(c) Take one kind of anomalies you indicated above, discuss how to find them efficiently? (*1.5 points*)

Part 3

In many applications, text documents may have structured portion (*e.g.*, time, author, venue, and keywords) and unstructured portion, such as a set of narratives (*e.g.*, abstract, content, etc.)

(a) Discuss how to design a text cube that may facility multiple dimensional analysis of text data. (*1.5 points*)

(b) To enhance the power of analysis, it is important to extract semantic structures from the narrative data. Discuss how to make good use of structured data in text cube to extract such semantic structures. (*2 points*)

Problem 2

Part 1

Recently, people pay attention to mining heterogeneous information networks.

(a) Give an example to show why a method for mining homogeneous information networks is usually not effective at mining heterogeneous ones. (*1 point*)

(b) Explain why integrated clustering and ranking can derive better clusters than a typical method that performs cluster analysis alone on heterogeneous information networks? (*1 point*)

(c) The information in the DBLP network may form a heterogeneous information network. Outline a method that may automatically generate hierarchical clusters on research topics in such a network. (*1.5 points*)

Part 2

Information providers may often provide conflicting information on a set of objects.

(a) Suppose there is only one true fact about each object regardless of time. Outline a method that may detect which stated fact is likely true and which information provider is more trustable than others. (*1.5 points*)

(b) Suppose the truth may change over time. Outline a method that may have similar power of truth finding as above but in time-evolving situations. (*2 points*)

Part 3

The World-Wide Web can be considered as a gigantic information network. However, the Web does not have clean structures for effective information network analysis.

(a) Discuss what are likely to be effective methods that may transform part of the web data into somewhat structured information network. (*1.5 points*)

(b) If part of the web data can be transformed into somewhat structured information network, they may be further linked together with structured database to form a powerful knowledge-base. Using one example, illustrate how quality of data can be substantially enhanced with such an integrated information network. (*1.5 points*)

Problem 3

Part 1

(a) What are the differences between semi-supervised classification and seminar supervised clustering? (*1 point*)

(b) In the CrossClus algorithm, a user provides guidance (or preferences) on clustering results, such as clustering students guided by their research groups. What are the differences of this kind of clustering from semi-supervised classification and seminar supervised clustering? (*2 points*)

Part 2

(a) The first pattern-based classification algorithm, CBA (Classification based on association), was proposed by B. Liu et al. in 1998. What are the strength and weakness of this algorithm comparing with C4.5, a typical decision tree algorithm? (*1 point*)

(b) Explain what are the major research efforts to make pattern-based classification both efficient and effective. (*2 points*)

Part 3

It is often more interesting if two items are correlated rather than just co-occur frequently together.

(a) Use an example to explain why *lift* is not a good measure to judge if two items are positively correlated? (*0.5 point*)

(b) Use an example to explain what is a good measure to judge if two items are positively correlated. (*1 point*)

(c) Sometimes, people want to mine negative correlations. Present an example to show we may have some similar concern in defining negative correlations. (*1 point*)

(c) Define a good measure to judge if two items are negatively correlated. (*1.5 points*)

3.14 Sample Exam Question Set 3.14

Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed)

(a) (data mining and data cube algorithms) (1) Name three algorithms that compute iceberg cubes efficiently, and (2) name three algorithms that effectively cluster high dimensional data.

(b) (comparing data mining methods) Use one or two sentences to distinguish the following pairs of methods: (1) *RAINFOREST* vs. *random forest*, and (2) *classification with partially labeled data* vs. *classification with skewed data.*

(c) (selection of data mining methods) Name or outline one data mining method that best fits each of the following tasks: (1) *finding regions of oil spill along the coast*, and (2) *detecting computer network intrusion in real time.*

Problem 1

Part 1

Suppose there are following three sequences of events registers in a sequence database:

$$(s_1): \ \langle (t_1, a_1), (t_2, a_2), (t_3, a_3), \ldots, (t_8, a_8), (t_9, b_9), (t_{10}, b_{10}) \rangle$$

$$(s_2): \ \langle (t_1, b_1), (t_2, b_2), (t_3, b_3), \ldots, (t_8, b_8), (t_9, b_9), (t_{10}, b_{10}) \rangle$$

$$(s_3): \ \langle (t_1, a_1), (t_2, b_2), (t_3, a_3), \ldots, (t_8, b_8), (t_9, a_9), (t_{10}, b_{10}) \rangle$$

where t_i is a timestamp ($t_i < t_{i+1}$), $a_i \neq b_i$, and the sequence (s_1) means that customer s_1 first bought a_1 at time t_1, then a_2 at time t_2, etc. Let $min\_support = 2$.

(a) How many (nonempty) sequential patterns are there in this sequence database? (Briefly explain your answer) (*2 point*)

(b) How many (nonempty) closed sequential patterns are there in this sequence database? (Briefly explain your answer) (*2 points*)

Part 2

(a) Outline a method that mines close sequential patterns efficiently in such a database. (*2 points*)

(b) Can your method discover efficiently rather long sequential patterns (*e.g.*, over 100 in length)? What are the performance bottlenecks? (*1 point*)

Part 3

Outline a method that mines long sequential patterns efficiently. (*3 points*)

Problem 2

Part 1

(a) What are the major differences between a homogeneous information network and a heterogeneous information network. (*1 point*)

(b) Reasoning why a clustering algorithm that works well in homogeneous networks may not work well in heterogeneous information networks. (*1.5 points*)

(c) Outline a clustering algorithm that works well in heterogeneous information networks. (*1.5 points*)

Part 2

(a) Similarly, a classification algorithm that works well in homogeneous networks may not work well in heterogeneous information networks. Reason on this using an example. (*1.5 points*)

(b) Outline a classification algorithm that works well in heterogeneous information networks. (*1.5 points*)

Part 3

(a) In a heterogeneous information network, an important mining task is to discover certain hidden relationships between nodes in the network (such as finding advisor-advisee relationships in the DBLP network). What are the challenges for mining such hidden relationships? (*1 point*)

(b) Outline a method that may discover such hidden relationships effectively. (*2 points*)

Problem 3

Part 1

(a) A cyber-physical network is a network that links physical sensors and information entities together to form an integrated network. Suppose your cyber-physical network is used for patient care. Describe what should be mined in such a cyber-physical network, (*1 point*)

(b) A major challenge of cyber-physical network is that sensor data may not be reliable and may contain erroneous data. Suppose sensors may not be costly, but decisions based on obtained data could be critical. Outline one method that may determine which pieces of data are more trustworthy based on the noisy data obtained in such a network. (*2 points*)

Part 2

(a) Suppose your cyber-physical network is used for patient care. Design a data cube that may summarize such a cyber-physical network data in a multidimensional space. (*1.5 points*)

(b) If new data is streamed into the system incrementally and dynamically, outline a stream cube design and an aggregation method that may handle such streaming data effectively. (*2 points*)

Part 3

(a) Since a cyber-physical network may contain sensitive data about people and their movements, privacy and security could become a major concern at mining such data. Take patient care cyber-physical network as an example. Discuss what could be the concerns on privacy at mining cyber-physical network data. (*1.5 points*)

(b) Outline one method that may preserve privacy at mining cyber-physical network data. (*2 points*)