Announcement

- ~~Individual~~ Team project! (Respect Notre Dame's teamwork tradition: This is why Fighting Irish is strong?) – voting result
- Project instruction [updated]:

- A team will have at most 2 members (one or two).
- Students will give their partner's name (or N/A – if they do individual projects by their own) in HW3.
- Members in the same team will have the same score.

- HW3 [updated]:

6 Follow-up Questions on Project (0 point)

Your answers may be open to the class by the instructor. We encourage communications between teams. Your answers may be released to graders.

Your partner name: _____ (NetID: _____)
Write "N/A" if you will do the project by your own.

1. Task 1: Data cleaning and integration

   a) How many unique papers and how many unique authors are there in your integrated and cleaned dataset?

Announcement (cont.)

- Comparative grading
  - Project… Traditional open-style grading will be more subjective.
- Project introduction slides [updated]:

  Students will **volunteer to present** their SciBot (tech and results) **in the last two lectures**. Classmates and the instructor will grade them based on the presentation. For the students **who do not present, the instructor** will grade their projects after all the lectures end. Note that we will have **comparative grading** – *finishing* all the **required tasks** cannot make sure that you have all points – but *finishing in high quality yes* – or you can have all points if you do *required tasks in low quality* but *some more* tasks.
- Project instruction [updated]:

  Graders should have higher expectations on graduates than undergraduates – not only on the project results (more tasks, better performances) but also on writing (a workshop-quality paper of strong reasoning). Undergraduates will be applied with a uniform grading policy no matter what majors they have.
  Graders tend to grade individual projects/undergraduates better than team projects/graduates if they generate the same results.

PRINCIPLE

Enable more interesting, powerful features for your SciBot!!!

Surprise graders!

My Grading Could Be…

		Time	
Task 1	Cleaning and Integration	**10 mins**	
Task 2	Entity name candidate generation	**20 mins** **(Abbreviation or Capital-case rules)**	+40 mins (Apriori)
	Entity name quality assessment	**10 mins** **(support + 2-gram sig.)**	+20 mins (n-gram sig.)
Task 3	Entity typing	**15 mins** **(majority voting)**	+40 mins (clustering+typing) OR (pattern-based typing)
Task 4	Collaboration discovery	**20 mins** **(FP-Growth)**	+5 mins (Kulc for 2-itemsets)
		75 mins	+105 mins = 180 mins = **3 hours**
Grading		**A, B+, A-, B+, A-**	A, A, A, A, A
	× (professor/student): 0.5 to 3.0	**38 mins** – 3h 45mins	1h 30mins – **9 hours**

My Response to Mid-Semester Survey

- Can we do teams please!
  - A: Yes! Go Irish!
- Comparative grading: Extra credit should be awarded for additional tasks but 100% should be given if the 7 tasks are *satisfactorily* completed – my answer: Yes!
- Q: Except for the basic 7 tasks, how many other tasks will each student does on average (according to your past experience)? I am afraid I don't have enough time to complete too many alternative tasks.
  - A: 1 to 3 (UIUC) if the 7 were not satisfactorily completed in students' self-evaluation.
- The only concern I have is that it seems really long and tedious.

My Response to Mid-Semester Survey

- Ultimate goals, Clarity on what need to know for tests
- Slow down a little: Vocabulary, In-class explanation, Discussion and questions
- Examples, details, programming example, Jupyter notebook
- Organization of slides
- Notes on board
- More candy and Alan

My Response to Mid-Semester Survey

- The original instructor didn't seem to know what was necessary to cover and the lectures were disjointed and hard to follow. Now that there is a consistent structure, I feel like I am very much behind after a confusing first three weeks.
  - A: That's why we have course review ☺
- There should really be more office hours - the hours picked out for the TA and Prof. Jiang is not sufficient for everyone.
  - A: Only one student came to me at office hours ☹

Concrete Learning Goals

- **Can process raw data: data cleaning, data integration, data reduction, dimension reduction**
- Can describe data warehouse, OLAP, data cube concepts and technology that work on multi-dimensional datasets
- **Can use Apriori and FP-Growth for frequent pattern mining**
- Can describe diverse patterns, sequential patterns, graph patterns
- **Can use Decision Tree, Naïve Bayes, Ensembles for classification**
- Can describe SVMs and Neural Networks for classification
- **Can use K-Partitioning Methods (K-Means, etc.) for clustering**
- Can describe Kernel-based Clustering and Density-based Clustering
- **Can use appropriate measures to evaluate results of different functionalities**

Course Review 1: Chapter 1 to 7

Meng Jiang

Data Science

Chapter 1: Introduction

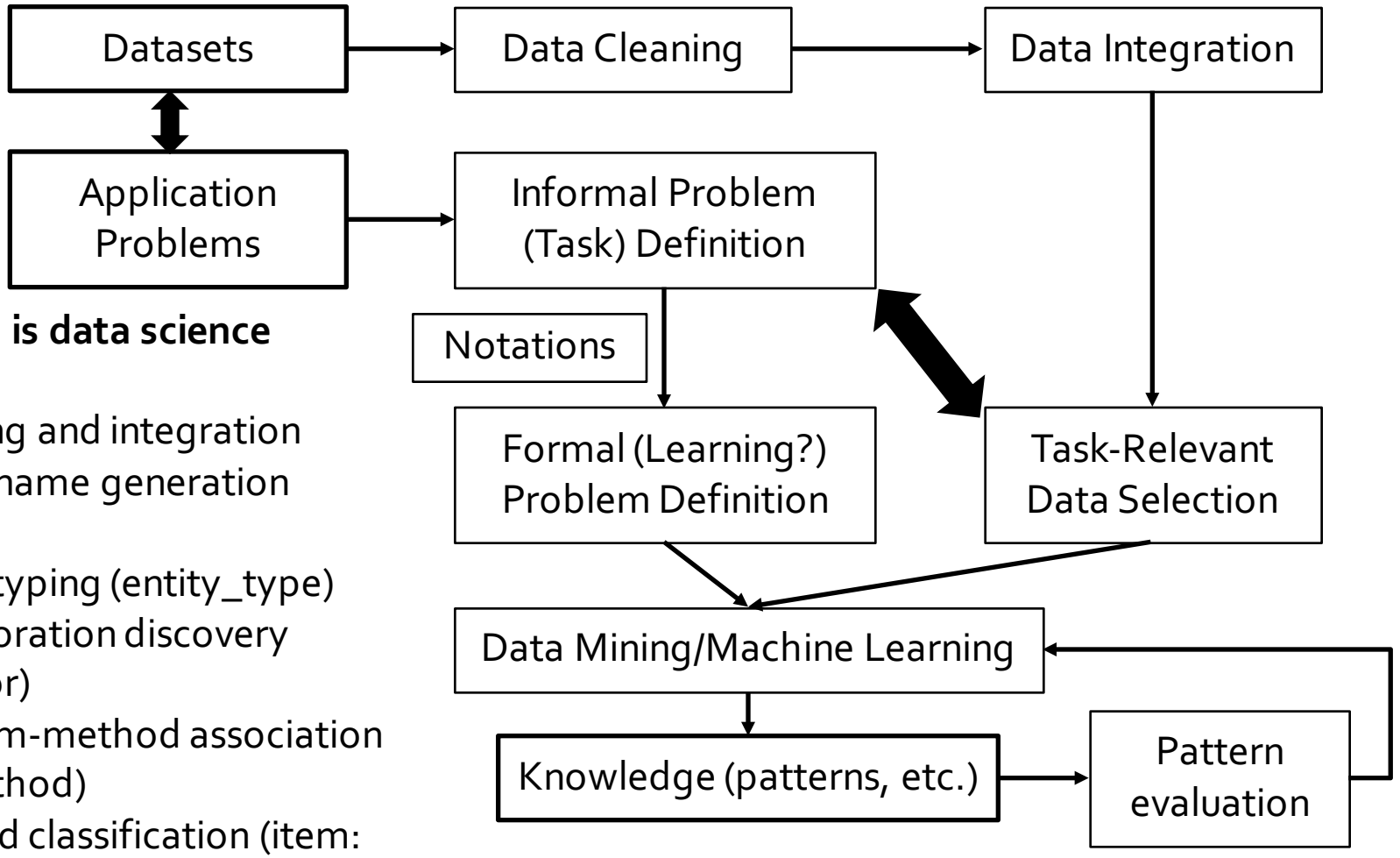
What is Data Science?

- "…the process of automatically discovering *useful information* in *large* repositories of data." — *Introduction to Data Mining* (Tan, Steinbach, & Kumar)
- "…the process of discovering *patterns* in data." — *Data Mining: Practical Machine Learning Tools and Techniques, 3rd Edition* (Witten, Frank, & Hall)
- "…the process of discovering *interesting patterns and knowledge* from *large* amounts of data." — *Data Mining: Concepts and Techniques, 3rd Edition* (Han, Kambler, & Pei)

Our Definition of the Course

- "…the art and craft of extracting *knowledge* from *large* bodies of *structured and unstructured* data using methods from many disciplines, including (but not limited to) machine learning, databases, probability and statistics, information theory, and data visualization."

Data Science Research



Why "SciBot" is data science research?

Task 1: Cleaning and integration

Task 2: Entity name generation (word_word)

Task 3: Entity typing (entity_type)

Task 4: Collaboration discovery (author_author)

Task 5: Problem-method association (problem_method)

Task 6: Method classification (item: method, label: conference)

Task 7: Paper clustering (item: paper, cluster: conference)

Chapter 2: Getting to know your data

Description & Visualization

Concepts

- Data object and attribute types
  - Transactions, sequences, graphs/networks …
  - Nominal, binary, ordinal, numeric
- Statistical description of data
  - Central tendency: Mean, median, mode, max, min …
  - Outlier-ness: Variance, standard deviation, Z-score …
- Data visualization
  - Boxplot, histogram, bar chart, quantile plot, Q-Q plot, scatter plot; word cloud, network…

Distance and Similarity

- Minkowski distance measures
  - Manhattan, Euclidean, "supremum"
- Similarity measures
  - Jaccard
  - Cosine
- Distribution difference measure
  - KL-Divergence

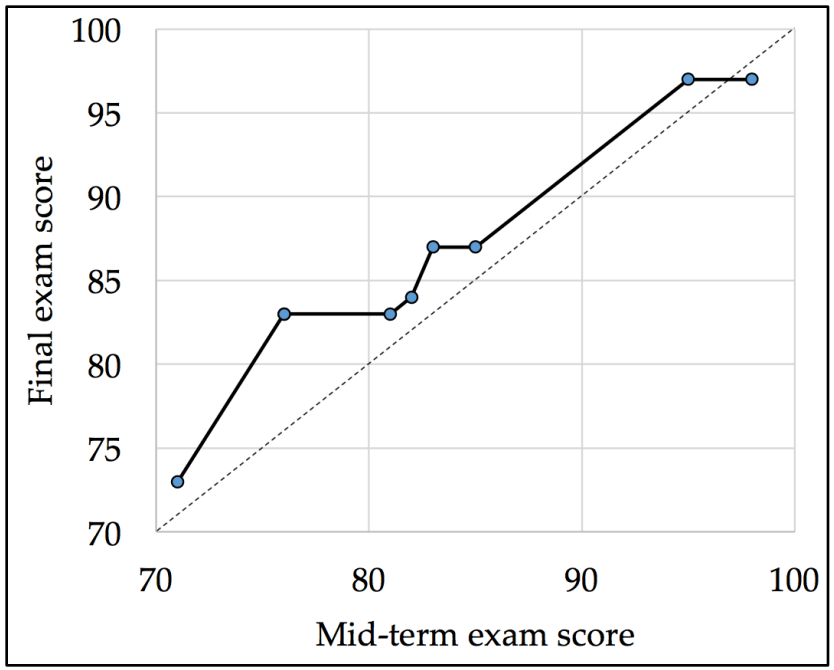
HW1-Q2

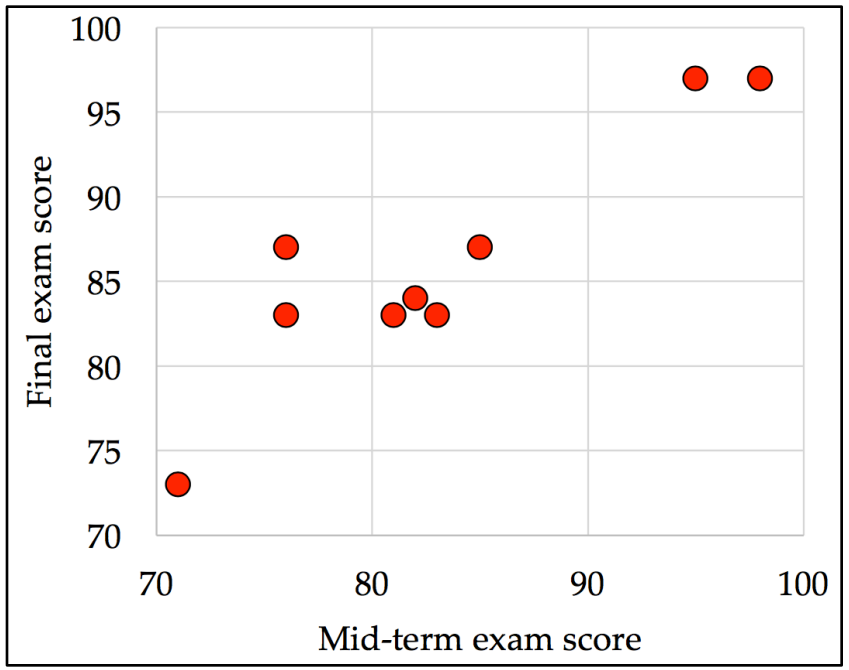
1. $mean_{mid-term} = \frac{71+85+83+98+76+81+76+82+95}{9} = 83$. Sorted: $71, 76, 76, 81, 82, 83, 85, 95, 98$;
   $median_{mid-term} = 82$. $mode_{mid-term} = 76$.

2. $min_{mid-term} = 71$. $max_{mid-term} = 98$. $Q1_{mid-term} = 76$. $Q3_{mid-term} = 85$.

3. $mean_{final} = \frac{73+87+83+97+87+83+83+84+97}{9} = 86$. Sorted: $73, 83, 83, 83, 84, 87, 87, 97, 97$;
   $median_{final} = 84$. $mode_{final} = 83$.

4. $min_{final} = 73$. $max_{final} = 97$. $Q1_{final} = 83$. $Q3_{final} = 87$.





HW1-Q3

Suppose the sample dataset has n values: x_1, ..., x_n.

1. (2′) Write the formula of the mean μ and variance v of the dataset.
2. (8′) Suppose we have a new value x_{n+1}. Write down how to compute the new mean μ' and new variance v' based on μ, v, and the increment value x_{n+1}. Note that the size of dataset becomes $n+1$. (Hint: You can only use μ, n, x_{n+1} to compute μ'. You can only use v, μ, n, x_{n+1} to compute v'.)

Solution:

1. $\mu = \frac{x_1+...+x_n}{n}$. $v = \frac{(x_1-\mu)^2+...+(x_n-\mu)^2}{n-1}$.
2. $\mu' = \frac{x_1+...+x_n+x_{n+1}}{n+1} = \frac{n\mu+x_{n+1}}{n+1}$. $v' = \frac{(x_1-\mu')^2+...+(x_n-\mu')^2+(x_{n+1}-\mu')^2}{n}$. We have

$$\begin{aligned} nv' - (n-1)v &= \{(x_1-\mu')^2 - (x_1-\mu)^2\} + ... + \{(x_n-\mu')^2 - (x_n-\mu)^2\} + (x_{n+1}-\mu')^2 \\ &= (2x_1 - \mu - \mu') \times (\mu - \mu') + ... + (2x_n - \mu - \mu') \times (\mu - \mu') + (x_{n+1} - \mu')^2 \\ &= \{2 \times (x_1 + ... + x_n) - n\mu - n\mu'\} \times (\mu - \mu') + (x_{n+1} - \mu')^2 \\ &= (2n\mu - n\mu - n\mu') \times (\mu - \mu') + (x_{n+1} - \mu')^2 \\ &= (n\mu - n\mu') \times (\mu - \mu') + (x_{n+1} - \mu')^2 \\ &= n(\mu - \mu')^2 + (x_{n+1} - \mu')^2 \end{aligned}$$

So, $v' = v + (\mu - \mu')^2 + \frac{(x_{n+1}-\mu')^2 - v}{n} = \frac{n-1}{n}v + \frac{1}{n+1}(x_{n+1} - \mu)^2$.

HW1-Q5

Suppose we have $1,000,000$ documents. The word "matrix" appears in $1,100$ of them, and the word "factorization" also appears in $1,100$ of them. And there are 100 documents that have both the word "matrix" and the word "factorization".

1. (5′) We take a word's occurrence in a document as a binary variable. Are the variables of "matrix" or "factorization" symmetric or asymmetric binary variables? Why?
2. (5′) What is the Jaccard similarity between the two variables?

Solution:

1. They are asymmetric binary variables. Negative is much more frequent than positive in the data. In other words, positive is more important than negative.

	Positive	Negative	Sum
Positive	100	1000	1100
Negative	1000	997900	998900
Sum	1100	998900	1000000

2. The Jaccard similarity is $\frac{100}{100+1000+1000} = \frac{100}{2100} = 0.048$.

HW1-Q6

We use the same data assumption as Question 5. Take each word as a data object, and take the documents as attributes. Usually we use word vectors to represent documents when we calculate the cosine similarity between documents. But now we want to measure the similarity between words.

1. (4′) Now the word "matrix" and the word "factorization" are represented with a $1,000,000$-length binary vector. What is the cosine similarity?

2. (6′) Again, we use the same data assumption as Question 5 (like we have 1 million documents). And suppose we have at least two unique words in the set of documents. What is the range of cosine similarity between any pair of words? What is the range of Jaccard similarity between any pair of words? And what is the range of Euclidean distance between any pair of words? (Hint: Can the similarities be negative in this case?)

Solution:

1. The cosine similarity is $\frac{100}{\sqrt{1100}\times\sqrt{1100}} = 0.091$.

2. The range of cosine simliarity is $[0, 1]$. The range of Jaccard similarity is also $[0, 1]$. The range of Euclidean distance is $[0, \sqrt{1000000}] = [0, 1000]$.

Chapter 3: Data Processing

Cleaning, Integration, Reduction

Cleaning and Integration

- Incomplete/missing data (Cleaning)
  - Global constant, attribute mean, most probable value
- Noisy data (Cleaning)
  - Binning, regression, outlier detection, clustering, semi-supervised
- Redundancy data (Integration)
  - Correlation analysis
    - Categorical attributes → Chi-square test: Null hypothesis, calculation
    - Numerical attributes → Covariance analysis: Positive, negative, independence, rho, covariance matrix

Reduction

- Data reduction
  - Parametric: Regression, log-liner models
  - Non-parametric: Histogram, clustering, (stratified) sampling
  - Min-max/Z-score/Decimal-scaling normalization
- Dimension reduction
  - "Curse of dimensionality"
  - Feature selection
    - Attribute subset selection
  - Feature extraction
    - Principal Component Analysis (PCA): Eigenvalue, Eigenvectors

HW1-Q7

(10′) We use original mid-term and final exam score data as in Question 1. Suppose the two exams are two variables. What is the covariance between them? Is it positive or negative?

Solution:

$$\begin{aligned}\sigma_{12} &= \frac{71 \times 73 + 85 \times 87 + 83 \times 83 + 98 \times 97 + 76 \times 87 + 81 \times 83 + 76 \times 83 + 82 \times 84 + 95 \times 97}{9} \\ &\quad -83 \times 86 \\ &= 53 > 0\end{aligned}$$

They are positively correlated.

HW1-Q8

(10′) We use the same data assumption as Question 5. We take a word's occurrence in a document as a binary variable. So we have two variables: (1) the word "matrix" in documents, and (2) the word "factorization" in documents. Please conduct a chi-square test on the two variables and give your conclusion.

Solution:

	Positive	Negative	Sum
Positive	100 (1.21)	1000 (1098.79)	1100
Negative	1000 (1098.79)	997900 (997801.21)	998900
Sum	1100	998900	1000000

$$\begin{aligned}\chi^2 &= \frac{(100-1.21)^2}{1.21} + 2 \times \frac{(1000-1098.79)^2}{1098.79} + \frac{(997900-997801.21)^2}{997801.21} \\ &= 8083.4\end{aligned}$$

We strongly reject the null hypothesis of independence between these two words (variables). They are strongly correlated.

Chapter 4: Data Cube

Concepts

Data Cube

- Concepts
  - Cell, Cuboid, Cube
  - Dimension Value, Dimension Level, Dimension
- Components
  - Dimension tables and Fact tables
  - Concept hierarchy and Measures
  - Schemas: **What are they?**
- Operations: **What are they?**
- Materialization: **What are they?**

Iceberg Cube

- Iceberg condition: min_sup (count)
- Assumption: "If the base cuboid has only the following base cells:…"
- Constraints
  - **Iceberg cells**, **Non-empty cells**, iceberg cube
  - **Base/Aggregate cells**/cuboids
  - **Closed cells** and closed cube

HW2-Q1

Suppose the base cuboid of a data cube contains two cells

$$(a_1, a_2, a_3, a_4, \ldots, a_{10}) : 1, (a_1, b_2, a_3, b_4, \ldots, b_{10}) : 1$$

where $a_i \neq b_i$ for any i. Obviously here we have 10 dimensions and each has only one level (no concept hierarchy).

1. How many **nonempty cuboids** are there in this data cube?
2. How many **nonempty aggregate closed cells** are there in this data cube?
3. How many **nonempty aggregate cells** are there in this data cube?
4. If we set minimum support = 2, how many **nonempty aggregate cells** are there in the corresponding iceberg cube?

1. (6′) **Answer: 2^{10}.** Since we have 10 dimensions with no concept hierarchy, there are 2^{10} cuboids and all of them should not be empty.
2. (6′) **Answer: 1.** There are 3 closed cells, including the two base cells and $(a_1, *, a_3, *, a_5, *, a_7, *, a_9, *)$. But only the latter one is a **aggregated** closed cell.
3. (6′) **Answer: 2014.** For each base cell, there are $2^{10} - 1$ aggregated cells. However, there are 2^5 cells that are counted twice since there are 5 common dimensions. Therefore, the total number of nonempty aggregate cells is $2 \cdot (2^{10} - 1) - 2^5 = 2014$.
4. (6′) **Answer: 2^5.** These two base cells have common value in 5 dimensions; therefore, there are 2^5 nonempty cells with support = 2 and all of them are aggregate cells.

HW2-Q3

Assume a base cuboid of 10 dimensions contains only three base cells[1]:

- $(a_1, d_2, d_3, d_4, \ldots, d_{10}) : 1$,
- $(d_1, b_2, d_3, d_4, \ldots, d_{10}) : 1$,
- $(d_1, d_2, c_3, d_4, \ldots, d_{10}) : 1$,

where $a_1 \neq d_1$, $b_2 \neq d_2$ and $c_3 \neq d_3$. The measure of the cube is *count*. Here we have 10 dimensions and each has only one level (no concept hierarchy).

1. How many **nonempty cuboids** will a full data cube contain?
2. How many **nonempty aggregate cells** will a full cube contain?
3. How many **nonempty aggregate cells** will an iceberg cube contain with the condition $count \geq 2$?

Solution:

1. (8′) **Answer:** 2^{10}
2. (8′) **Answer:** (i) Each cell generates $2^{10} - 1$ nonempty aggregated cells, thus in total we should have $3 * 2^{10} - 3$ cells with overlaps removed. (ii) We have $3 * 2^7$ cells overlapped once (thus count 2) and $1 * 2^7$ (which is $(*, *, *, d_4, \ldots, d_{10})$) overlapped twice (thus count 3). Thus we should remove in total $1 * 3 * 2^7 + 2 * 1 * 2^7 = 5 * 2^7$ overlapped cells. (iii) Thus we have: $3 * 8 * 2^7 - 5 * 2^7 - 3 = 19 * 2^7 - 3$.
3. (9′) **Answer:** (i) $(*, *, d_3, d_4, \ldots, d_9, d_{10})$ has count 2 since it is generated by both cell 1 and cell 2; similarly, we have (ii) $(*, d_2, *, d_4, \ldots, d_9, d_{10}) : 2$, (iii) $(*, *, d_3, d_4, \ldots, d_9, d_{10}) : 2$; and (iv) $(*, *, *, d_4, \ldots, d_9, d_{10}) : 3$. Therefore, we have $4 * 2^7 = 2^9$.

Chapter 5: Cube Computation, Data Warehouse and OLAP

Concepts, Algorithms

Cube Computation

- Multi-way array aggregation (× Apriori property for iceberg cube; × high dimensionality)
  - ABC to {AB, AC, BC}: How?
- BUC (√ Apriori property for iceberg cube; × high dimensionality)
- Shell-fragment (√ For high dimensionality)

Data Warehouse and OLAP

- "A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process."

	OLTP	**OLAP**
users	clerk, IT professional	knowledge worker
function	day to day operations	decision support
DB design	application-oriented	subject-oriented
data	current, up-to-date detailed, flat relational isolated	historical, summarized, multidimensional integrated, consolidated
usage	repetitive	ad-hoc
access	read/write index/hash on prim. key	lots of scans
unit of work	short, simple transaction	complex query
# records accessed	tens	millions
#users	thousands	hundreds
DB size	100MB-GB	100GB-TB
metric	transaction throughput	query throughput, response

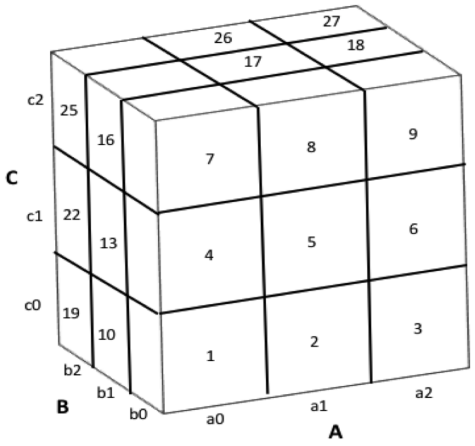
HW2-Q4

We have a data array containing 3 dimensions A, B and C shown in Figure 2. The 3-D array is divided into 27 small chunks. Each dimension is divided into 3 equally sized partitions. The cardinality (size) of the dimensions A, B, and C is 900, 300, and 600. Since we divide each dimension into 3 parts with equal size, the sizes of the chunks on dimensions A, B, and C are 300, 100, and 200 respectively. Suppose we want to use **Multiway Array Aggregation Computation** to materialize the 2-D cuboids AB, AC and BC.

1. What is the ordering of chunk scanning that achieves the maximum computation efficiency, i.e. requires the least memory units?
2. Following the ordering you give in the above subquestion, what is the minimum memory requirement for holding all the 2-D planes?

Solution:

1. (12′) **Answer:** $1, 10, 19, 4, 13, 22, 7, 16, 25, 2, 11, 20, 5, 14, 23, 8, 17, 26, 3, 12, 21, 6, 15, 24, 9, 18, 27.$
2. (13′) **Answer:** $300 \times 200\ (AC) + 300 \times 300\ (AB) + 300 \times 600\ (BC) = 330000.$



Chapter 6: Frequent Pattern Mining

Apriori, FP-Growth and Evaluation

Concepts

- Itemset, k-itemset, frequent k-itemset
- Absolute/relative **support**
- Association: **support** and **confidence**
- **Closed** pattern, Max pattern (lossy)
- **Apriori property:** Any subset of a frequent itemset must be frequent. If any subset of an itemset S is infrequent, then there is no chance for S to be frequent—why do we even have to consider S?!

Apriori

- Outline of Apriori (level-wise, candidate generation and test)
  - Initially, scan DB once to get frequent 1-itemset
  - Repeat
    - Generate length-(k+1) candidate itemsets from length-k frequent itemsets
    - Test the candidates against DB to find **frequent** (k+1)-itemsets
    - Set k := k +1
  - Until no frequent or candidate set can be generated
  - Return all the frequent itemsets derived

FP-Growth

- Idea: Frequent pattern growth (FPGrowth)
  - Find **frequent single items** and partition the database based on each such item
  - Recursively grow frequent patterns by doing the above for each partitioned database (also called ***conditional database***)
  - To facilitate efficient processing, an efficient data structure, **FP-tree**, can be constructed
- Mining becomes
  - **Recursively construct and mine (conditional) FP-trees**
  - Until the resulting FP-tree is empty, or until it contains **only one path**—single path will generate all the combinations of its sub-paths, each of which is a frequent pattern

Try WikiBooks' Example

- https://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Frequent_Pattern_Mining/The_FP-Growth_Algorithm

Pattern Evaluation

- Interestingness Measures
  - Subjective
  - Objective
    - Support, confidence
    - Null-variant
      - Lift: symmetric
      - Chi-square test
    - Null-Invariance
      - AllConf
      - Jaccard, Cosine
      - Kulc, MaxConf + Imbalance Ratio

Chapter 7: Advanced Frequent Pattern Mining

Diverse Patterns

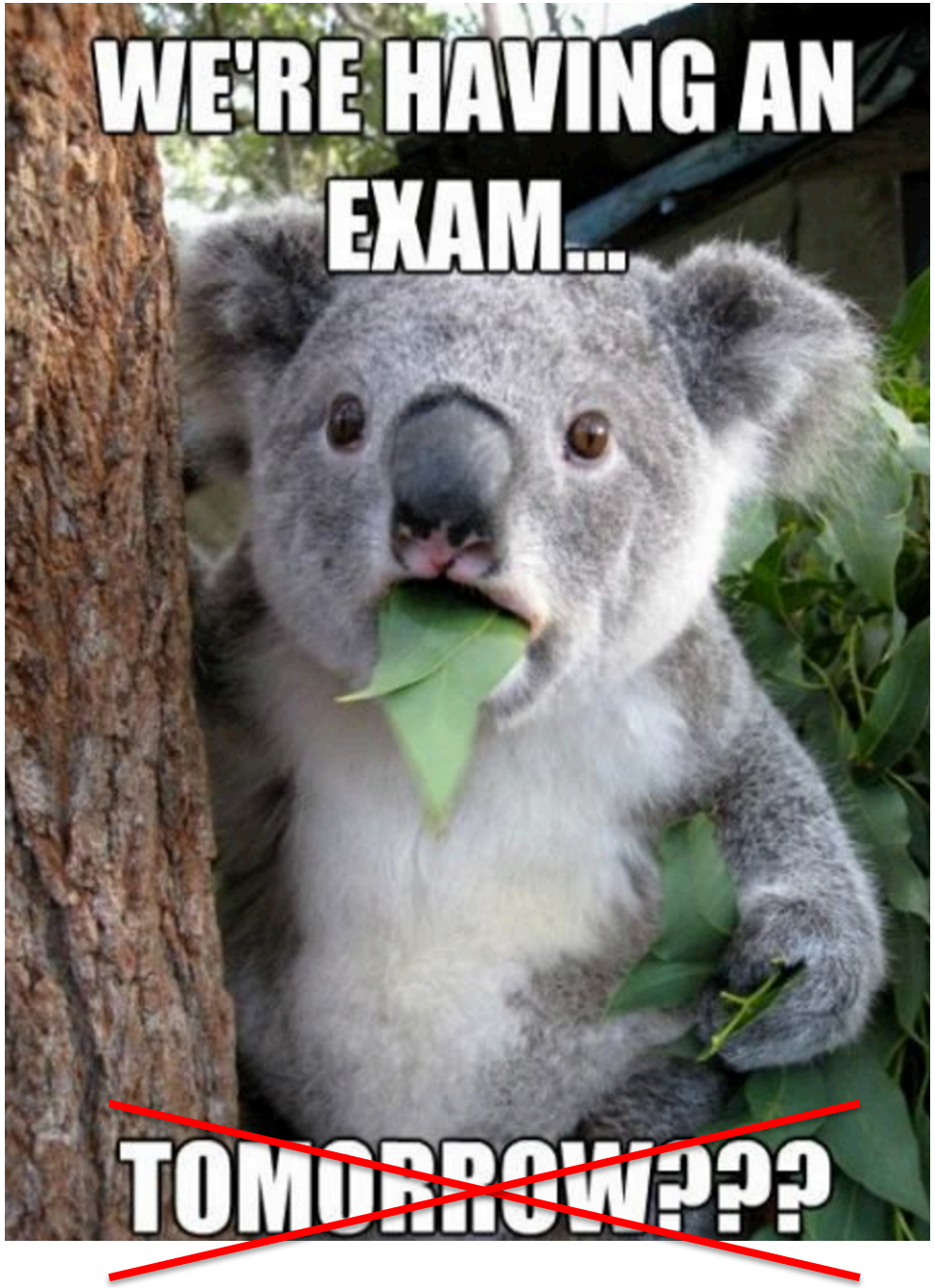
Sequential Patterns

Graph Patterns

Concepts

- Diverse Patterns
  - Multiple-Level Associations
  - Multi-Dimensional Associations
  - Quantitative Associations
  - Negative Correlated Patterns

Pattern	**Closed Pattern (Concepts)**	**Idea 1: Pattern candidate generation and pruning**	**Idea 2: Pattern growth**
Frequent pattern (itemset)	Closed frequent itemset	Apriori (1994)	FP-Growth (2000)
Sequential pattern	Closed seq. pattern	GSP (1996)	PrefixSpan (2004)
Graph pattern	Closed graph pattern	FSG (2000-2001)	gSpan (2002)



Mid-term exam:

Oct. 5th (Thursday)

2:00 p.m. – 3:15 p.m. (75 minutes)

In class

Take a pen!!!

You are only allowed to have an A4-size paper double-sided for reference.

I believe you can write a lot a lot on it…

Alan Time

- If no one is answering, maybe he/she is behind you ❤