Recap lecture

1. Exam and tips for preparation
2. Main learning goals
   - common sources of errors and misleading results
   - list of main learning goals
3. Selected recap topics (based on the questionnaire)

Exam Wed 13.12. 2023 13:00–16:00

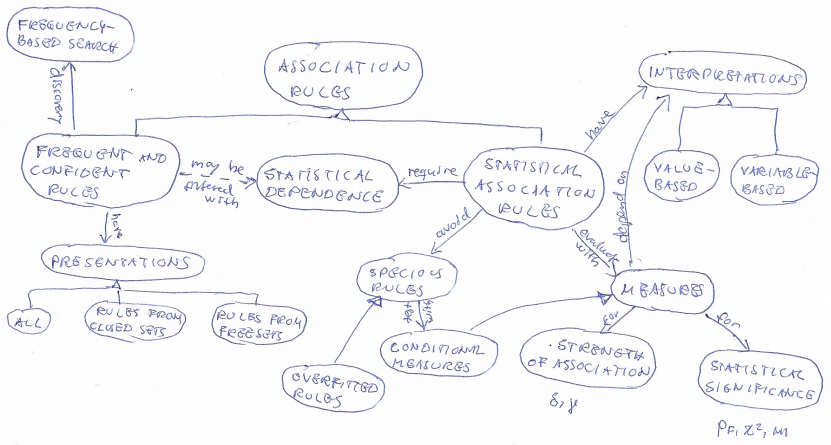
Undergraduate Centre, A-sali (Aalto-sali) Y202a

- **non-programmable calculator** capable of roots, trigonometric and logarithmic calculations needed
- all advanced mathematical formulations will be provided in the exam paper
- both explanation and calculation/proof tasks (may be in the same task)
  - e.g., brief explanations what X means and why is it important or how it differs from Y
  - e.g., given toy data calculate/simulate/show something (measures, clustering, frequent sets/their condensed representations, association rules, graph patterns, ...)

Tricks for (deep) learning

1. **Motivate** yourself
   - Why this is important i) in general ii) to me some day?
2. Create **big pictures** and see **connections**!
   - summary tables comparing approaches or methods
   - concept map showing main concepts are their relations
   - connect new to something familiar (e.g., single linkage clustering ↔ connected components)

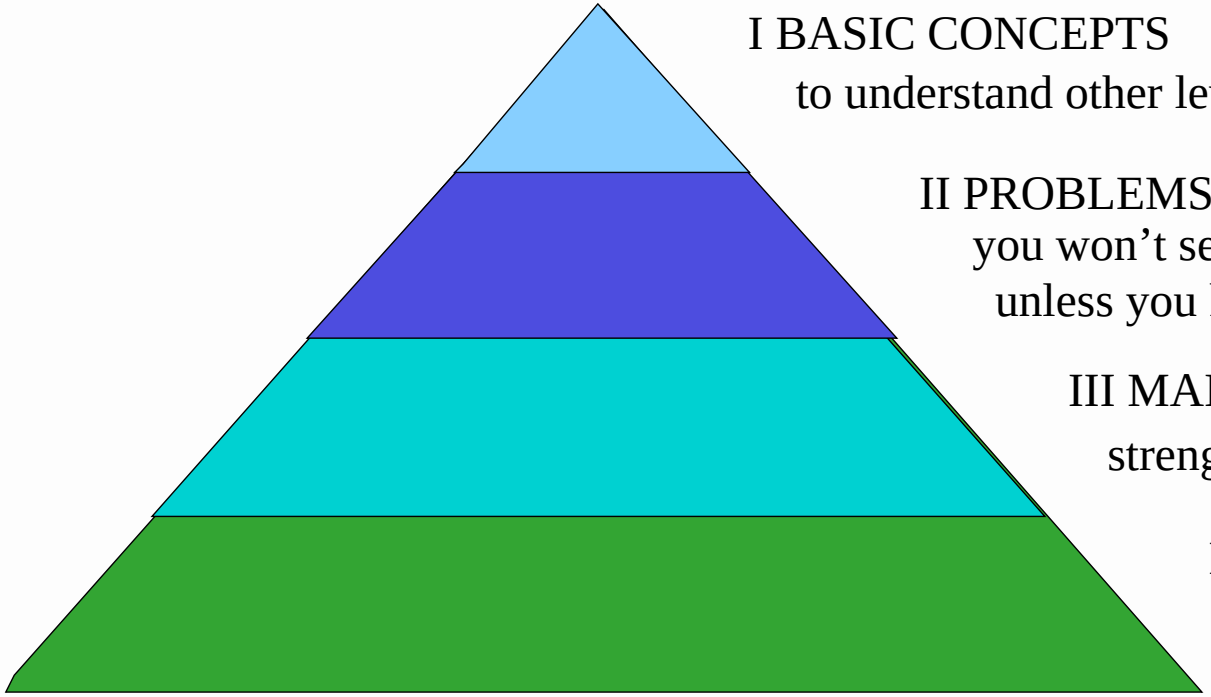
method	data type	cluster type	benefits	drawbacks
***K*-representatives**				
K-means				
K-medoids				
...				
Hierarchical				
singe-link				
...				
Graph-based				
Density-based				
Probabilistic				



Tricks for (deep) learning

3. Try **active learning**
   - Invent new problems to solve (How to apply A in B?)
   - Invent and calculate examples
   - Ask yourself questions (**what-why-how?** What is good/bad here? What happens if...)
   - Write things on your own words and notations, make schematic drawings
4. Set yourself **learning goals** (I should understand A)
   - evaluate which goals you have met
   - soon a list of main learning goals $\rightarrow$ check and tailor for yourself
   - check the recap topic wish (asking what-why-how or good/bad)

Learning DM: Different levels



I BASIC CONCEPTS
to understand other levels

II PROBLEMS
you won't search right solutions,
unless you know problems

III MAIN APPROACHES
strengths and weaknesses?

IV SOLUTION METHODS
minor details can be checked
understand when you see it

Most important learning goal

Become **aware of error sources** at each step of the DM process and do less erroneous modelling (aim error-free!)

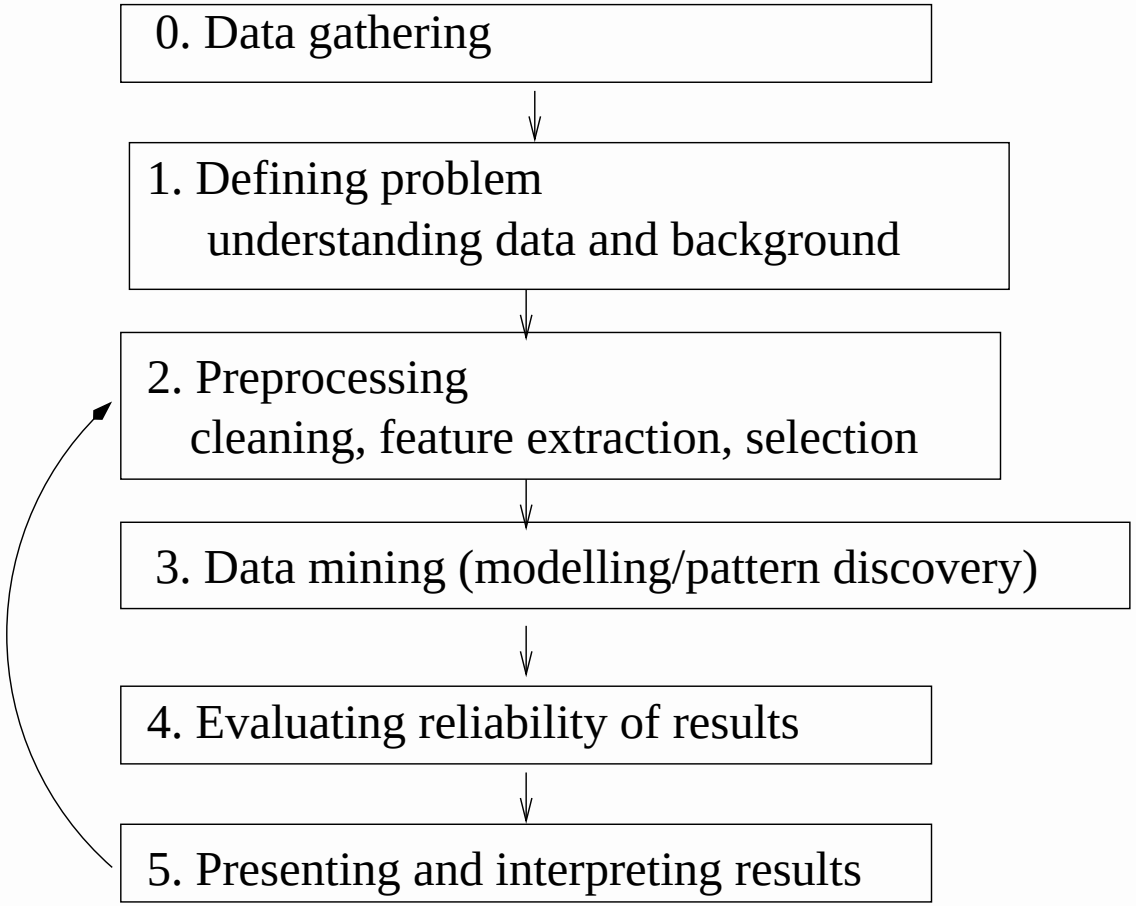
If you cannot solve something correctly, it is better to know/tell errors than believe the solution is correct

Always ask:

- **Does this make sense?**
  - does data look sensible?
  - is method X sensible (here/at all)?
  - do results look sensible?
  - is the validation method sensible?
- **What could go wrong here?**

Recall: KDD process

with additional step 0



Most common sources of errors

- trusting accuracy of data sources/search engines
- errors and anomalies in data
- ignoring data types (e.g., PCA on categorical data)
- being unaware of model/algorithm assumptions (or not checking if they fit the problem)
- blind use of ready-made libraries (what do they calculate?)
- not testing/verifying own programs
- not checking all intermediate results
- no validation/wrong validation
- unjustified conclusions/misleading presentation of results

Always suspect everything! ($\sim$ Cartesian doubt)

Main topics and learning goals

I **DM process**: Be able to plan a valid DM project.
- Know error sources and strategies

II **Preprocessing**: Know tasks and strategies (missing data, scaling, **discretization**, dimension redution (**PCA** & **SVD**)

III **Distance and similarity**: Know common **measures** (for numerical, categorical and mixed; sets, strings, text and graphs). How to construct **similarity graphs**? What is the **curse of dimensionality** and remedies to it?
- When a distance measure is metric and why it is useful?

Main topics and learning goals

IV **Clustering**: Know clustering objectives and what elements affect clustering results. How to study clustering tendency? What methods to try for a given data and clustering objective? **K-representatives, hierarchical and spectral clustering**, their assumptions, strengths and weaknesses. How to **validate** clustering?

V **Association mining**: Know **different types of association patterns**. How to evaluate strength and statistical significance of association? How to utilize **monotonicity** in search? **Apriori** algorithm and computational strategies. Problem of overfitted and specious rules. Condensed representations of frequent sets and their problems.

Main topics and learning goals

VI **Other pattern discovery**: How to search new pattern types with monotonic properties (**Generic Apriori**)? How to evaluate significance of discovered patterns with **randomization testing**? What is the multiple hypothesis problem?

VII **Graph mining**: What are **graph isomorphism, isomorphic subgraphs and MCG**? How to measure distance between graphs, mine graph patterns or cluster graphs?

VIII **Social network analysis**: What are the main tasks of social network analysis and approaches to solve them? Especially, how to detect communities or evaluate centrality or similarity of nodes?

Main topics and learning goals

IX **Web mining and search**: Know how search engines work. How to rank documents (esp. **PageRank and HITS**)?

X **Recommender systems**: Know main approaches, especially **collaborative filtering**.

XI **Text mining**: How to preprocess text and present in vector space? How to get informative features? Special problems and techniques (esp. for clustering).

Recap topics

- Computationally demanding DM tasks
- Generic Apriori
- Generating rules from sets
- Evaluating statistical associations rules
- Idea of spectral clustering
- Co-clustering with a spectral example
- LSA (recaps also SVD)
- Mid-frequency terms and phrases
- Multiple hypothesis testing problem
- Types of association patterns
- Other topics? (existing material/questions)

Computationally demanding DM tasks

No polynomial time algorithms known (most are NP-hard):

problem	What to do?
optimal feature selection	use heuristics (like greedy)
exact K-means clustering	use heuristic K-means alg.
best classification rule best association rule	usually scalable (if direct search), restrict complexity, use min_{fr}
(sub)graph isomorphism	
MCS-distances and graph edit distance	impractical for larger graphs, consider alternative (transformation-based) distances
frequent subgraph discovery	restrict complexity, use higher min_{fr}
graph partitioning	solve relaxed problem (spectral) or use heuristic community detection alg.

Computationally demanding DM tasks

- exponential search space not necessarily a problem, if effectively pruned and the data is nice (e.g., sparse transaction data)
- polynomial time algorithms can also be expensive!
  - if the same routine is repeated multiple times (e.g., all pairwise distances)
  - hierarchical (at least $\mathcal{O}(n^2)$) and spectral ($\mathcal{O}(n^3)$) clustering can be heavy for large data (vs. K-means $\mathcal{O}(nKq)$, q=number of iterations)
- space (memory) may also be the bottleneck

Emphasis: Know when to expect heavy computation and if your method is solving the exact problem or some approximation/constrained version

Generic Apriori

- Decide general pattern type (e.g., sets, graphs, sequences).
- Define subpattern relationship (subset, subgraph, subsequence). Notate $\beta \subseteq \alpha$, if β is a subpattern of α.
- Define complexity of the pattern. Notate $|\alpha|$. (We say that α is an i-pattern, if $|\alpha| = i$, $i = 1, 2, \ldots$)
- Decide interesting binary property Φ and check it is **monotonic**! (Otherwise Apriori doesn't work!)

ϕ monotonic if for all $\beta \subseteq \alpha$: if $\Phi(\beta) = 0$, then $\Phi(\alpha) = 0$

Examples of monotonic properties Φ

1. Frequent subgraphs (given min_{fr})

$$\Phi(\mathbf{G}) = \begin{cases} 1 & \text{if } fr(\mathbf{G}) \geq min_{fr} \\ 0 & \text{otherwise} \end{cases}$$

2. Classification rules $\mathbf{X} \rightarrow C$ (C fixed) with potentially high δ (given $\min_\delta$)

$$\Phi(\mathbf{X} \rightarrow C) = \begin{cases} 1 & \text{if } P(\mathbf{X}C)P(\neg C) \geq \min_\delta \\ 0 & \text{otherwise} \end{cases}$$

(Note: Φ is for pruning by monotonicity. For actual discoveries, check that $\delta(\mathbf{X}, C) \geq \min_\delta$. See Exercise 3.3)

Generic Apriori

Notations: $\mathcal{F}_i$ = i-patterns having property Φ, C_i = candidate i-patterns

Algorithm:

$i = 1; \mathcal{F}_1 = \{\alpha \mid \Phi(\alpha) = 1, |\alpha| = 1\}$

while $\mathcal{F}_i \neq \emptyset$

- Generate candidates C_{i+1} from $\mathcal{F}_i$
- Prune $\alpha \in C_{i+1}$ if $\exists\, \beta \subseteq \alpha, |\beta| = i, \beta \notin \mathcal{F}_i$

} (monotonicity)

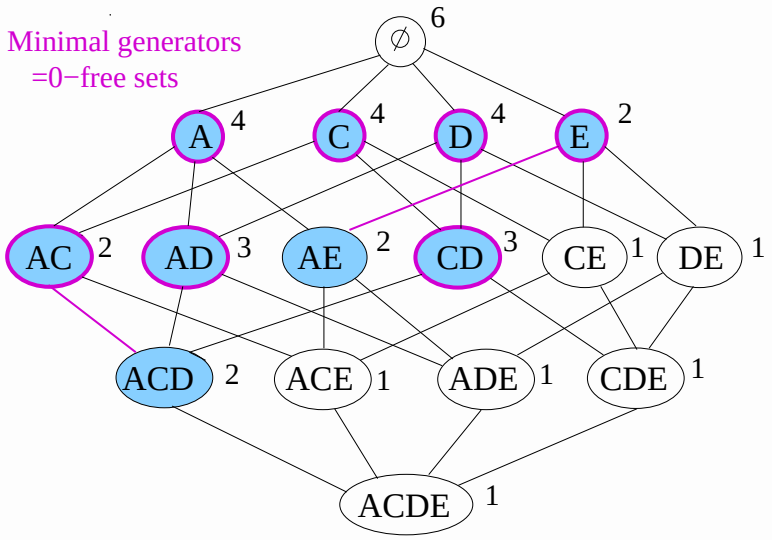
- Evaluate Φ for all $\alpha \in C_{i+1}$
- Set $\mathcal{F}_{i+1} = \{\alpha \in C_{i+1} \mid \Phi(\alpha) = 1\}$
- $i = i + 1$

Return $\cup_i \mathcal{F}_i$

Generating rules from sets $\mathbf{X} \in \mathcal{F}$

Idea: For all $|\mathbf{X}| \geq 2$ and all $C \in \mathbf{X}$, test rule $\mathbf{X} \setminus \{C\} \to C$ (and possibly $\mathbf{X} \setminus \{C\} \to \neg C$) with the given measures

Example: given frequent sets with $fr \geq 2$ (blue), find positive consequent rules with $\phi \geq 0.6$ and $\delta \geq 0.1$.

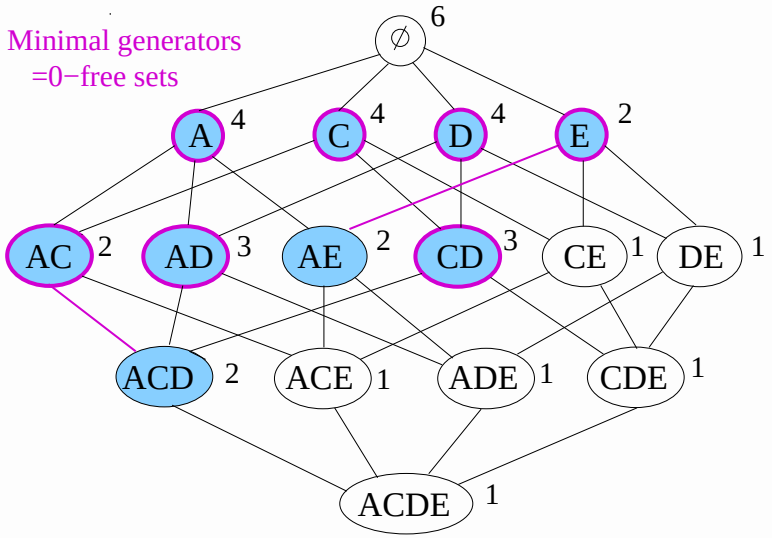


	ϕ		ϕ	δ
$A \to C$	$\frac{2}{4}$	$C \to A$	$\frac{2}{4}$	$-\frac{1}{9}$
$A \to D$	$\frac{3}{4}$	$D \to A$	$\frac{3}{4}$	$\frac{1}{18}$
$A \to E$	$\frac{2}{4}$	$E \to A$	$\frac{2}{2}$	$\frac{1}{9}$
$C \to D$	$\frac{3}{4}$	$D \to C$	$\frac{3}{4}$	$\frac{1}{18}$
$AC \to D$	$\frac{2}{2}$			$\frac{1}{9}$
$AD \to C$	$\frac{2}{3}$			0
$CD \to A$	$\frac{2}{3}$			0

Generating rules from sets $\mathbf{X} \in \mathcal{F}$

Example continued: Identify **0-free** frequent sets with $fr \geq 2$ and find positive consequent rules with $\phi \geq 0.6$ and $\delta \geq 0.1$.

- Recall: $\mathbf{X}$ 0-free if for all $\mathbf{Y} \subsetneq \mathbf{X}$ holds $P(\mathbf{Y}) > P(\mathbf{X})$



	ϕ		ϕ	δ
$A \to C$	$\frac{2}{4}$	$C \to A$	$\frac{2}{4}$	$-\frac{1}{9}$
$A \to D$	$\frac{3}{4}$	$D \to A$	$\frac{3}{4}$	$\frac{1}{18}$
$C \to D$	$\frac{3}{4}$	$D \to C$	$\frac{3}{4}$	$\frac{1}{18}$

No rules found!

Lesson: Condensed representations can miss the best association rules!

Evaluating statistical associations rules

Given rule $\mathbf{X} \rightarrow C{=}c$, $c \in \{0, 1\}$. Evaluate

1. Statistical dependence: Is
   $\delta(\mathbf{X}, C{=}c) = P(\mathbf{X}, C{=}c) - P(\mathbf{X})P(C{=}c) > 0$ or
   $\gamma(\mathbf{X}, C{=}c) = \frac{P(\mathbf{X}, C{=}c)}{P(\mathbf{X})P(C{=}c)} > 1$? If not, prune out.
2. Evaluate significance, using measure M and its threshold. (E.g., required $MI \geq \min_{MI}$.) If not significant, prune out.
3. Evaluate overfitting: does $\mathbf{X} \rightarrow C{=}c$ improve significantly (using conditional measure M_c) all $\mathbf{Y} \rightarrow C{=}c$, $\mathbf{Y} \subsetneq \mathbf{X}$?
   i) If $P(C{=}c|\mathbf{X}) \leq P(C{=}c|\mathbf{Y})$ for some $\mathbf{Y} \subsetneq \mathbf{X}$, prune out. **Why?**
   ii) If $M_c(\mathbf{X} \rightarrow C{=}c|\mathbf{Y} \rightarrow C{=}c)$ not sufficient (given threshold) for some $\mathbf{Y} \subsetneq \mathbf{X}$, prune out. (E.g., $MI_C < min_{MI_C}$)

Note: 3ii) is for value-based associations. For variable-based associations, you should check also $M_c(\neg\mathbf{Y} \rightarrow C{\neq}c|\mathbf{X} \rightarrow C{\neq}c)$ (extra stuff).

Idea of spectral clustering

1. Create similarity graph $\mathbf{G}$ (or use a social network).
2. Present $\mathbf{G}$ in (low-dimensional) vector space as $\mathbf{Y}$ such that distances $d(\mathbf{y}_i, \mathbf{y}_j)$ ($\mathbf{y}_i, \mathbf{y}_j \in \mathbf{Y}$) reflect edge weights w_{ij} in $\mathbf{G}$.
   - create graph Laplacian $\mathbf{L}$ (normalized or unnormalized)
   - take first K eigenvectors of $\mathbf{L}$ with smallest λs [a] $\rightarrow$ $\mathbf{u}_1, \ldots, \mathbf{u}_K$
   - create data matrix $\mathbf{Y}$ using $\mathbf{u}_1, \ldots, \mathbf{u}_K$ as columns
   - if $\mathbf{L}_{sym}$, normalize rows to unit length [b]
3. Cluster $\mathbf{Y}$ (usually K-means).

---

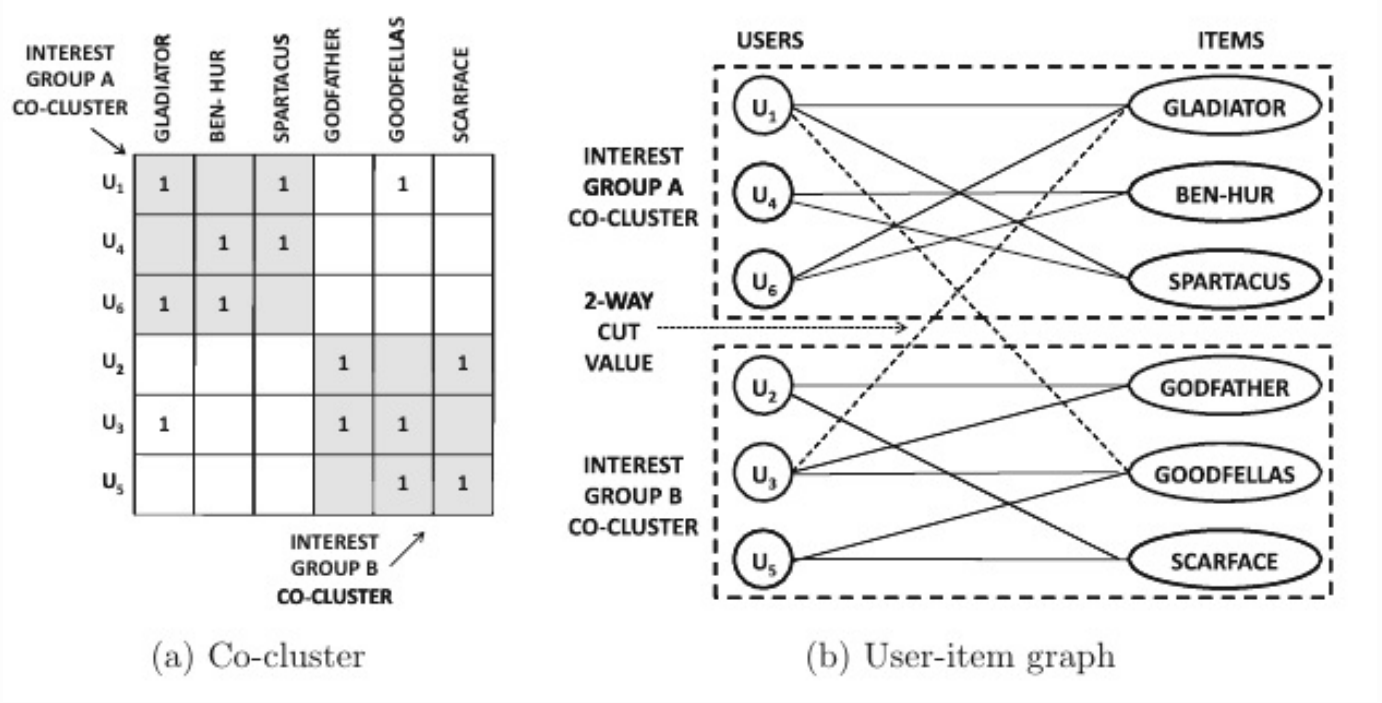
[a] except $\lambda = 0$, since then $\mathbf{y} \propto (1, 1, \ldots, 1)$

[b] With $\mathbf{L}_{rw}$, normalize columns to unit length, unless already done.

Co-clustering with a spectral example

When you want to cluster both rows and columns, especially

- users and items (recommender systems)
- documents and words (text mining)
- most effective when sparse data



Example from Aggarwal Fig 18.6.

Write 12×12 adjacency matrix $\mathbf{W}$ → Laplacian $\mathbf{L}$

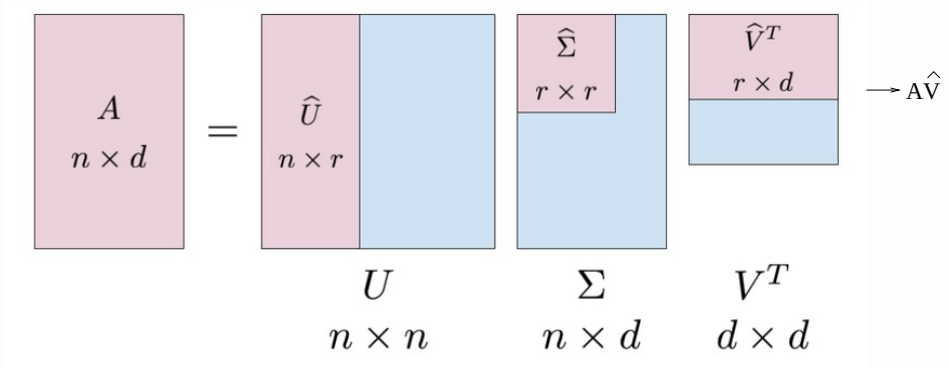
Co-clustering with a spectral example

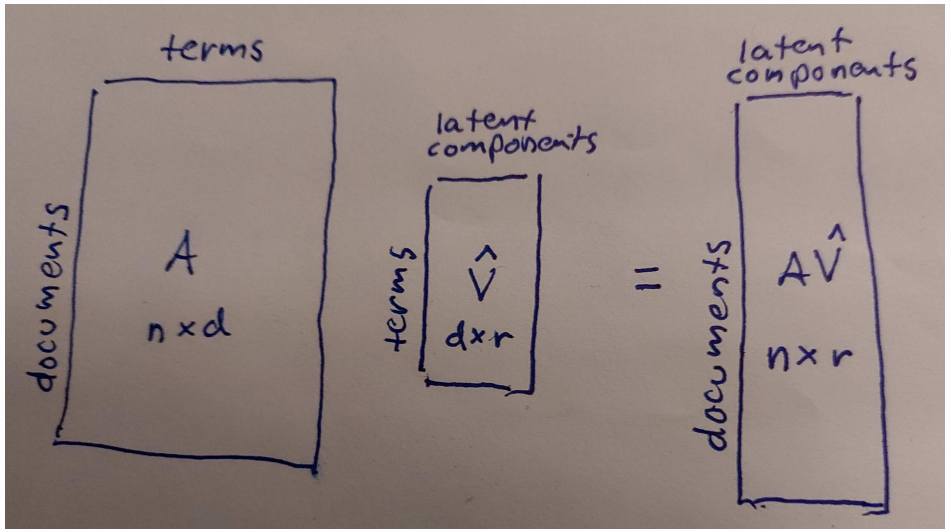
Using unnormalized Laplacian. 2nd eigenvector (column 2) clusters users (rows 1–6) and items (rows 7–12).

```
Eigenvalues: [1.45040194e-15 2.67949192e-01 1.00000000e+00 1.00000000e+00
 1.18639350e+00 2.00000000e+00 3.00000000e+00 3.00000000e+00
 3.47068342e+00 3.73205081e+00 4.00000000e+00 5.34292308e+00]
E-vectors
 [[ 0.28867513 -0.10566243  0.40507055 -0.05083812  0.11190063 -0.28867513
  -0.00561909 -0.40820962 -0.13413108 -0.39433757  0.28867513 -0.46849451]
 [ 0.28867513 -0.39433757 -0.24656238 -0.32538233  0.23530186  0.28867513
    0.35632945  0.19923853  0.43581073 -0.10566243  0.28867513 -0.06857143]
 [ 0.28867513 -0.28867513 -0.15850817  0.37622045 -0.42674499 -0.28867513
  -0.35071035  0.20897109  0.20512889  0.28867513  0.28867513 -0.16065758]
 [ 0.28867513  0.39433757 -0.40507055  0.05083812  0.23530186 -0.28867513
  -0.00561909 -0.40820962  0.43581073  0.10566243 -0.28867513 -0.06857143]
 [ 0.28867513  0.10566243  0.24656238  0.32538233  0.11190063  0.28867513
    0.35632945  0.19923853 -0.13413108  0.39433757 -0.28867513 -0.46849451]
 [ 0.28867513  0.28867513  0.15850817 -0.37622045 -0.42674499  0.28867513
  -0.35071035  0.20897109  0.20512889 -0.28867513 -0.28867513 -0.16065758]
 [ 0.28867513 -0.10566243  0.24656238  0.32538233 -0.11190063 -0.28867513
    0.35632945  0.19923853  0.13413108 -0.39433757 -0.28867513  0.46849451]
 [ 0.28867513 -0.39433757 -0.40507055  0.05083812 -0.23530186  0.28867513
  -0.00561909 -0.40820962 -0.43581073 -0.10566243 -0.28867513  0.06857143]
 [ 0.28867513 -0.28867513  0.15850817 -0.37622045  0.42674499 -0.28867513
  -0.35071035  0.20897109 -0.20512889  0.28867513 -0.28867513  0.16065758]
 [ 0.28867513  0.28867513 -0.15850817  0.37622045  0.42674499  0.28867513
  -0.35071035  0.20897109 -0.20512889 -0.28867513  0.28867513  0.16065758]
 [ 0.28867513  0.10566243  0.40507055 -0.05083812 -0.11190063  0.28867513
  -0.00561909 -0.40820962  0.13413108  0.39433757  0.28867513  0.46849451]
 [ 0.28867513  0.39433757 -0.24656238 -0.32538233 -0.23530186 -0.28867513
    0.35632945  0.19923853 -0.43581073  0.10566243  0.28867513  0.06857143]]
```

LSA (latent semantic analysis)

LSA = SVD applied to document-term (or term-document) matrix





- new presentation to documents $\mathbf{A}\hat{\mathbf{V}} = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}$
- and for terms $\mathbf{A}\hat{\mathbf{U}} = \hat{\mathbf{V}}\hat{\mathbf{\Sigma}}$
- reduces dimensionality!
- helps with synonymy (similar terms tend to be combined)
- may help with polysemy (if one main meaning)

Note: If term features $F_1, \ldots, F_d$ and ith singular vector $(\mathbf{V}_{i1}, \ldots, \mathbf{V}_{id})^T$, ith new feature $F'_i = \sum_{j=1}^{d} \mathbf{V}_{ij} F_j$.

Image: Perunicic (2017).

LSA toy example

Documents:

(cat)^2 (lion)^2 (kitty)

(cat)^2 (lion)^3 (kitty)^2

(cat) (lion) (kitty) (jaguar)

(jaguar)^3 (car)^2 (engine)

(jaguar)^2 (car) (engine)

(jaguar)^2 (car) (engine)^2

document-term matrix with tf-idf weights

$$\mathbf{D} = \begin{bmatrix} 2 & 2 & 1 & 0 & 0 & 0 \\ 2 & 3 & 2 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0.585 & 0 & 0 \\ 0 & 0 & 0 & 1.755 & 2 & 1 \\ 0 & 0 & 0 & 1.170 & 1 & 1 \\ 0 & 0 & 0 & 1.170 & 1 & 2 \end{bmatrix}$$

Here $tfidf(w_j, d_i) = fr(w_j|d_i) \cdot \log \frac{n}{n_j}$.

Note: $idf = \log(2) = 1$ for all words except for “jaguar”,

$idf(\text{jaguar}) = \log(1.5) = 0.585$

LSA toy example: New presentation for documents

```
Singular values
[5.34774741 4.09614197 1.06647632 0.60457418 0.52357289 0.07911012]
V^T
[[-5.54058090e-01 -6.97497899e-01 -4.50769743e-01 -5.26295418e-02
  -1.71779747e-02 -1.60623926e-02]
 [-2.67257237e-02 -3.90118618e-02 -1.78272243e-02  5.91518361e-01
   5.79873106e-01  5.57941941e-01]
 [ 1.51382506e-04  3.99531936e-02 -3.74464454e-02 -3.05214148e-01
  -4.77894588e-01  8.21865299e-01]
 [ 6.56873417e-01 -6.33949084e-02 -6.93413670e-01 -2.06029595e-01
   2.02649099e-01  1.26897596e-02]
 [-4.29978205e-01  4.52274325e-01 -1.27507972e-01 -5.65003826e-01
   5.20427081e-01  6.50747650e-02]
 [ 2.75574675e-01 -5.49370110e-01  5.45899068e-01 -4.38751600e-01
   3.50938537e-01  9.26523906e-02]]
New data DV_2 using 2 singular vectors
[[-2.95388172 -0.1493024 ]
 [-4.10214936 -0.20614148]
 [-1.73311401  0.26247343]
 [-0.14278319  2.75580288]
 [-0.09481693  1.82989153]
 [-0.11087932  2.38783347]]
```

New features approximately
-0.55*cat* -0.70*lion* -0.45*kitty* and
0.59*jaguar* $+0.58$*car* $+0.56$*engine*

How and why to increase frequency of mid-frequency terms and phrases?

1. **Too frequent** terms not specific, do not separate documents $\Rightarrow$
   - remove stopwords and other too frequent words
   - *idf* (in *tf-idf*)
   - specialize terms using n-grams (e.g., *"data analysis"* and *"Bayesian data analysis"*, if *"data"* useless)
2. **Too rare** terms useless for detecting similarities $\Rightarrow$
   - stemming and lemmatization
   - spell-checking
   - detect synonyms (e.g., WordNet)

Multiple hypothesis testing problem

Recall:

- Statistical hypothesis test: if $p \leq \alpha$, reject the null hypothesis (pattern is declared significant)
- type I error = null hypothesis is true but the pattern passes the test (a spurious discovery)
- In a single test, probability of type I error is $P(\text{error I}) \leq \alpha$

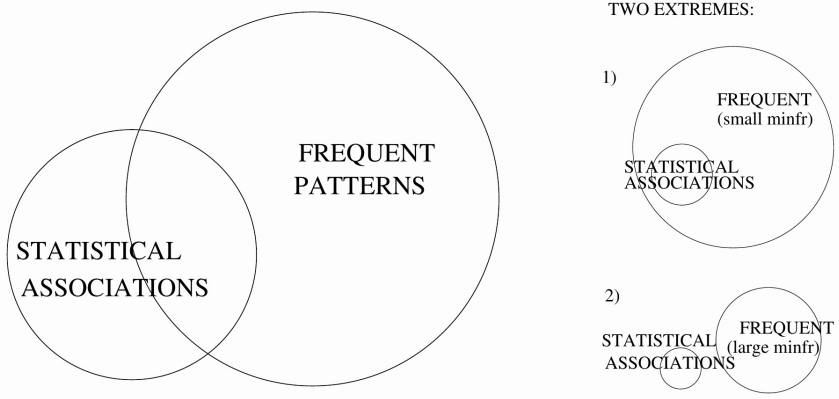
Problem: If you test m true null hypotheses and each test has $P(\text{error I}) = \alpha$, then $m \cdot \alpha$ spurious patterns will pass the test! $\Rightarrow$ Classical thresholds $\alpha = 0.05$ or $\alpha = 0.01$ cannot be used in DM!

$\Rightarrow$ multiple hypothesis testing correction methods (Bonferroni, Šídák, Hochberg, FDR correction methods, ... – not covered in the course)

Short note: Types of association patterns

- Sets: frequent sets or statistical dependency sets
- Rules: frequent co-occurrence or statistical dependence (between condition and consequence)

Two approaches find generally different patterns!



If sufficiently small min_{fr} is feasible, you can filter statistical associations afterwards (but this can be heavy!)

Value-based and variable-based interpretation of association $\mathbf{X} \rightarrow C = c$

Let $I_{\mathbf{X}}$ be an indicator variable:
$I_{\mathbf{X}} = 1$, if $\mathbf{X}$ holds, and $I_{\mathbf{X}} = 0$, otherwise.

Important: Are we interested in association between **values** $I_{\mathbf{X}}$=1 and C=c or between binary **variables** $I_{\mathbf{X}}$ and C?

$\Rightarrow$ **different goodness measures and different results!**

- **lift** γ measures strength of association between values
- **leverage** δ measures also strength of association between variables

lectures 6–7

Other topics (existing material/questions?)

- PCA assumptions: lec3
- Clustering tendency: lec3
- Efficient frequency counting (database projections, tid lists) lec8
- Specious associations lec8
- K-representatives: lec4
- Linkage metrics: lec4

Principal component analysis (PCA) assumptions

1. High **variance** reflects important structures of data.
2. Data can be presented well as a **linear** combination of suitable **orthogonal** basis vectors (PCs).

How to study clustering tendency and choose features?

Approaches:

1. Visual inspection of pairwise distance distributions
   - only hints
2. Filtering methods, e.g.,
   - Entropy-based measures
   - Hopkins statistic
3. Wrapper models + cluster validation indices
   - e.g., average silhouette, Calinski-Harabasz, Davies-Bouldin, and external indices

4. Yule-Simpson's paradox and other specious associations

Statistical dependence is a necessary but not a sufficient condition of causal relation!

- Often a **majority** of dependencies are specious (illusory, spurious, apparent) associations
- e.g., cake → exam failure was a sideproduct of alcohol → exam failure and alcohol → cake