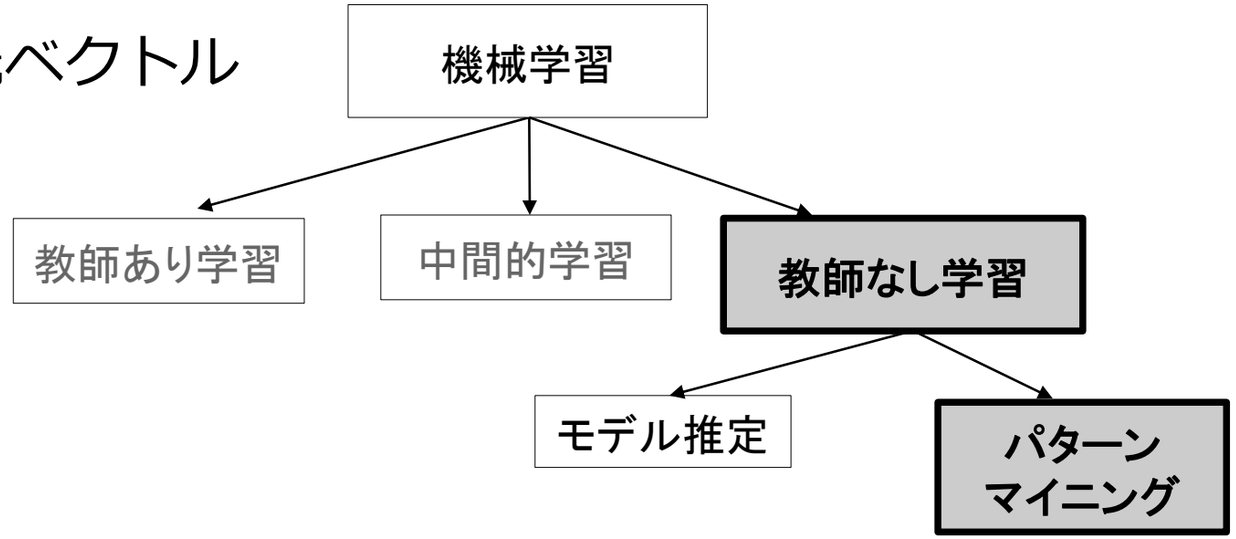
12. パターンマイニング

- 本章の説明手順
  - 教師なし、パターンマイニングの問題設定
  - 出現頻度の高い項目集合を見つける
  - 頻出項目集合から、有用な規則を見つける
  - 疎な行列に対して低次元ベクトル表現を見つけることにより、未知の値の予測を行う

12. パターンマイニング

- 問題設定
  - 教師なし学習
    - （疎な）ベクトル → 規則性
    - 規則性の例
      - ✓ 頻出項目、連想規則、低次元ベクトル
    - 応用例
      - ✓ 推薦システム



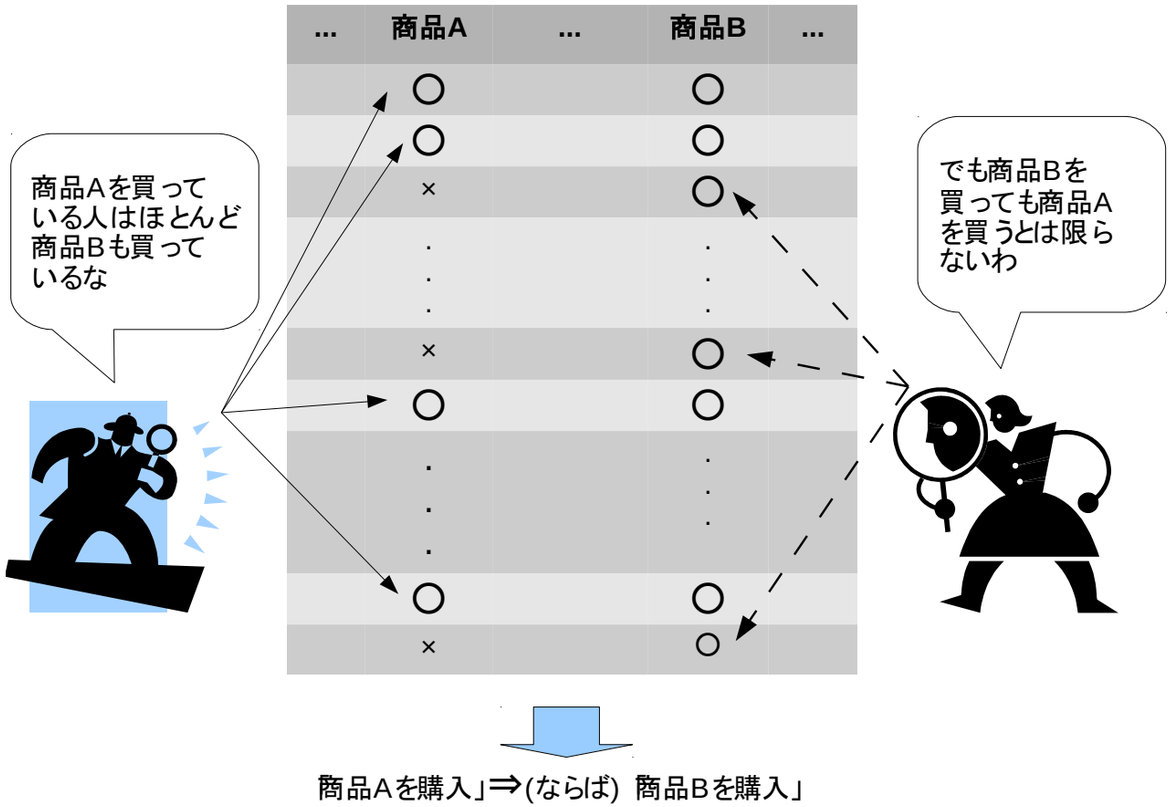
12.1 カテゴリ特徴に対する「教師なし・パターンマイニング」問題の定義

- データセット（教師なし）
  - （疎な）カテゴリまたは数値ベクトル

$$\{\boldsymbol{x}_i\} \quad i = 1, \ldots, N$$

数段階の離散値で実質的に
カテゴリ要素とみなせるもの

  - 問題設定1
    - データセット中で一定頻度
      以上で現れるパターンを抽出



12.1 カテゴリ特徴に対する「教師なし・パターンマイニング」問題の定義

- 問題設定2
  - 似ているデータを参考にして、空所の値を予測する

	映画	ドラマ	ステージ	スポーツ	ドキュメンタリー
	★ 5	★ 4	★ 1		
	★ 4	★ 5		★ 3	★ 2
				★ 5	★ 5
	★ 4	?		★ 5	

12.2 頻出項目抽出

- 例題：バスケット分析

No.	ミルク	パン	バター	雑誌
1	t	t		
2		t		
3				t
4		t	t	
5	t	t	t	
6	t	t		

バスケット分析では、1件分の
データをトランザクションとよぶ

- バスケット分析の目的
  - トランザクション集合中で、一定割合以上出現する項目集合を抽出

12.2.1 頻出の基準と問題の難しさ

- 頻出の基準：支持度 (support)

$$\mathrm{support}(items) = \frac{T_{items}}{T}$$

  - 全トランザクション数 T に対する、項目集合 *items* が出現するトランザクション数 T_{items} の割合

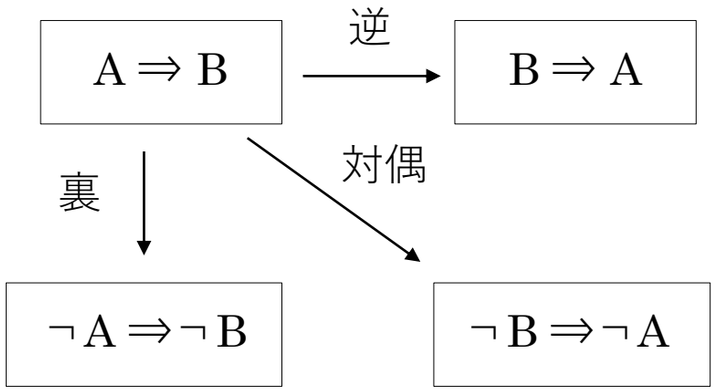
- バスケット分析の問題点
  - すべての可能な項目集合について、支持度を計算することは現実的には不可能

商品数1,000の店なら、項目集合の種類数は2^{1000}

→ 高頻度の項目集合だけに絞って計算を行う必要がある

12.2.2 Aprioriアルゴリズムによる頻出項目抽出

- 命題論理
  - 「AならばB」が成り立つなら、その対偶である
    「¬Bならば¬A」は必ず成り立つ



「A ⇒ B」は「¬A∨B」と定義されている。

この定義より、「¬B ⇒¬A」は

¬（¬B）∨(¬A)

なので、

B ∨¬A

となり、「¬A∨B」と等しい

12.2.2 Aprioriアルゴリズムによる頻出項目抽出

- a prioriな原理

ある要素が頻出ならば、その部分集合も頻出である

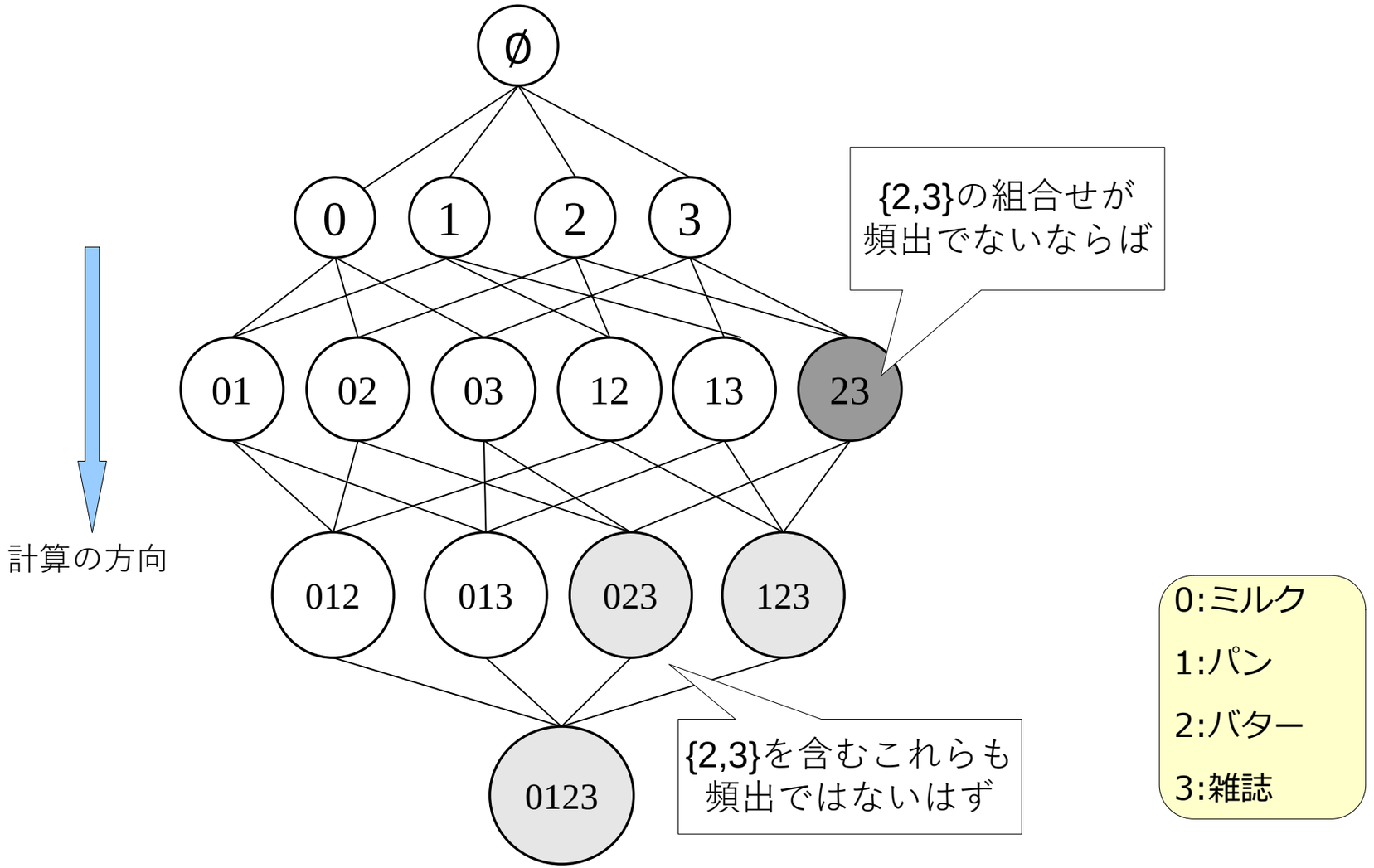
対偶

例）「パン・ミルク」が頻出
ならば「パン」も頻出

ある要素が頻出でないならば、その要素を含む上位集合も頻出でない

例）「バター・雑誌」が頻出でない
ならば「バター・雑誌・パン」
も頻出でない

12.2.2 Aprioriアルゴリズムによる頻出項目抽出



12.2.2 Aprioriアルゴリズムによる頻出項目抽出

- 頻出項目集合抽出の手順
  1. 項目数1の集合C_1を求める
  2. C_1から支持度が閾値以下の要素を削除し、集合F_1を求める
  3. k=2から始め、F_k =∅（空集合）になるまで以下を繰り返す
     a. F_{k-1} の要素を組合せ、項目数kの集合C_k を作成する
     b. C_k の要素で、その部分集合がF_{k-1}に含まれないものを削除する
     c. C_k から支持度が閾値以下の要素を削除し、F_k とする

12.3 連想規則抽出

- 連想規則抽出の目的
  - 「商品Aを買った人は商品Bも買う傾向がある」というような規則性を抽出したい
  - 確信度またはリフト値の高い規則を抽出

$$\mathrm{confidence}(\mathrm{A} \Rightarrow B) = \frac{\mathrm{support}(A \cup B)}{\mathrm{support}(A)} = \frac{T_{A \cup B}}{T_A}$$

前提部Aが起こったときに結論部Bが起こる割合

$$\mathrm{lift}(\mathrm{A} \Rightarrow B) = \frac{\mathrm{confidence}(A \Rightarrow B)}{\mathrm{support}(B)}$$

Bだけが単独で起こる割合とAが起こったときにBが起こる割合との比

12.3.3 規則の有用性

- 支持度・確信度・リフト値の意味
  - support({ハム, 卵}) : 0.1
  - confidence(ハム⇒卵) : 0.7　lift(ハム⇒卵): 5
  - 「全体顧客の10%がハムと卵を一緒に購入している」
  - 「ハム購入者の70%が卵も購入している」
  - 「ランダムに選んだ顧客が卵を買う確率に対して、ハムを買った顧客が卵を買う確率は5倍大きい」

12.3.4 Aprioriアルゴリズムによる連想規則抽出

- a prioriな原理

ある項目集合を結論部に持つ規則が頻出ならば、
その部分集合を結論部に持つ規則も頻出である

例）結論部が「パン・ミルク」の規則が頻出ならば、結論部が「パン」の規則も頻出である

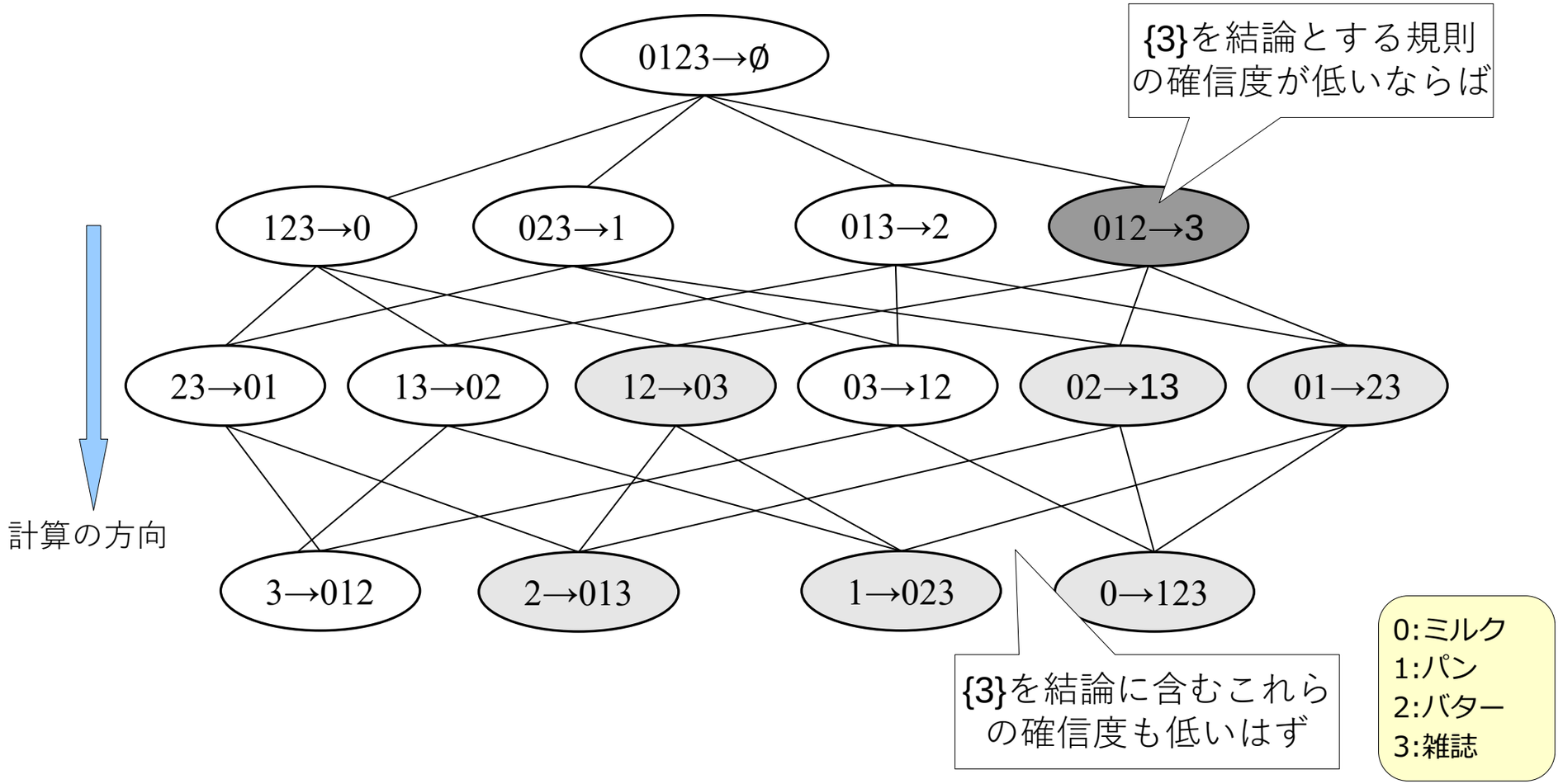
対偶

ある項目集合を結論部に持つ規則が頻出でないならば、
その上位集合を結論部に含む規則集合も頻出でない

例）結論部が「雑誌」の規則が頻出でないならば、結論部が「パン・雑誌」の規則も頻出でない

12.3.4 Aprioriアルゴリズムによる連想規則抽出

- a priori原理に基づく探索

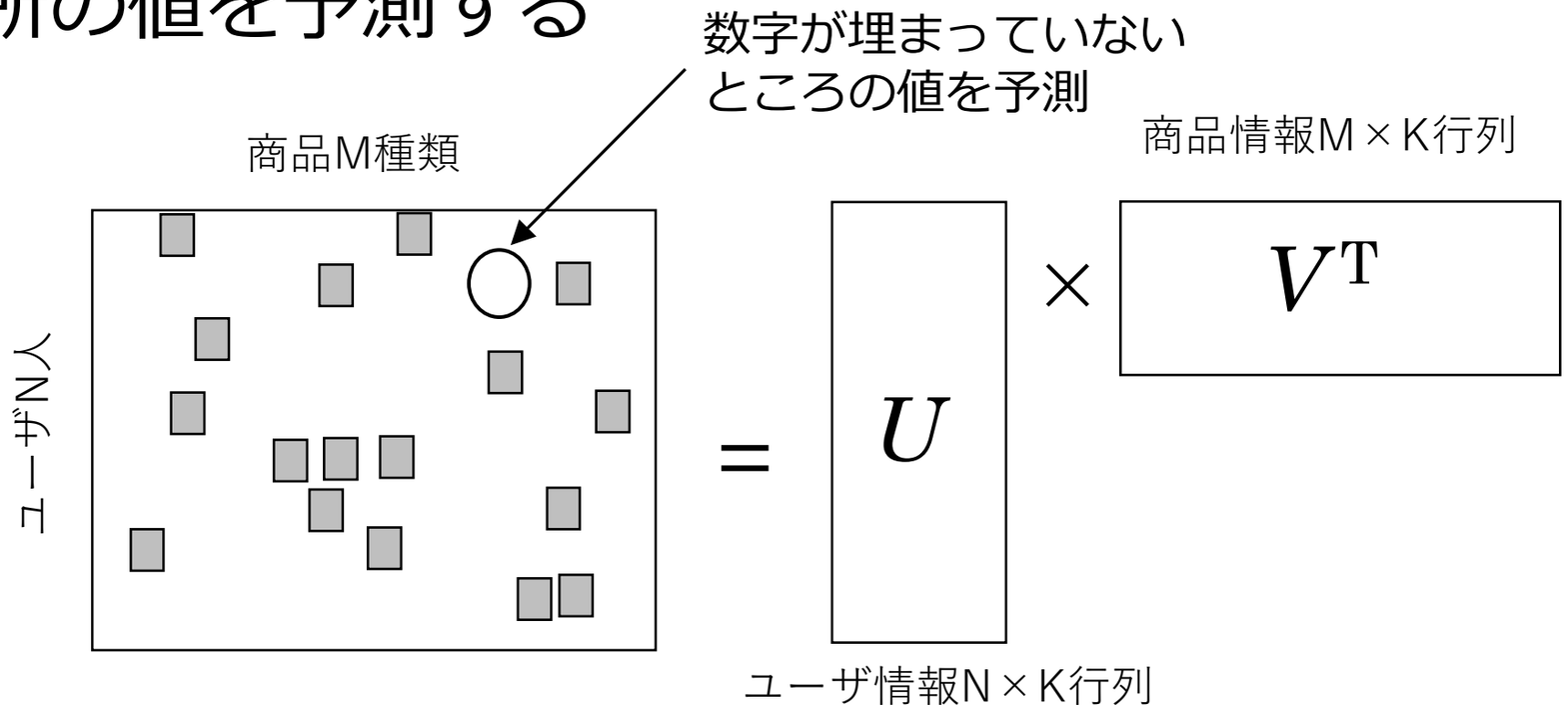


12.3.4 Aprioriアルゴリズムによる連想規則抽出

- 連想規則抽出の手順
  1. 頻出項目集合を求める
  2. 求めた頻出項目集合の要素のそれぞれについて、その要素中のひとつの要素を結論部、残りの要素を前提部とした規則集合 H_1 を作成する
  3. k=2から始め、 $H_k = \varnothing$（空集合）になるまで以下を繰り返す
     a. H_{k-1} の各要素について、前提部から結論部へ項目を1つ移動した規則を作成して H_k とする
     b. H_k から結論部が H_{k-1} の結論部を組み合わせたものでない要素を削除
     c. H_k から評価値（確信度またはリフト値）が閾値以下の要素を削除

12.5 推薦システムにおける学習

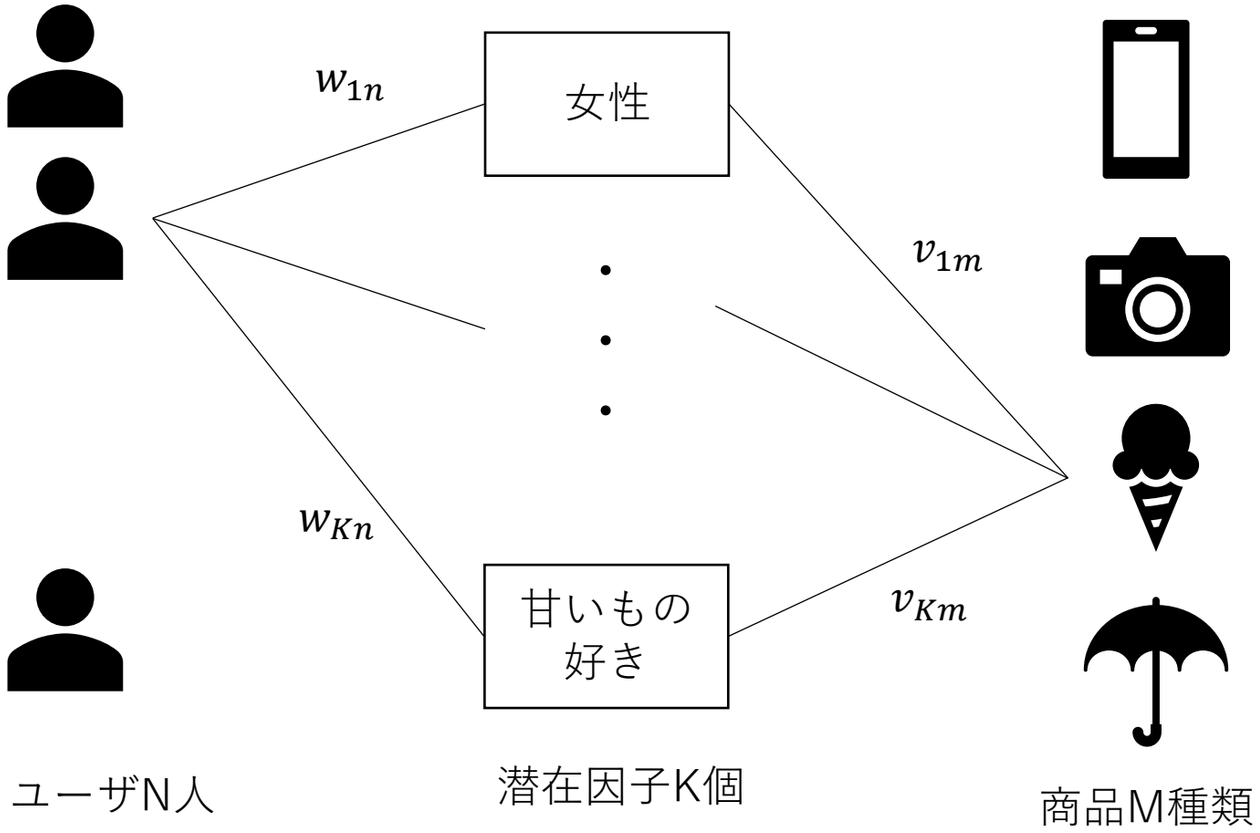
- 協調フィルタリング
  - アイデア：疎な行列は低次元の行列の積で近似できる
  - 値のある部分だけで行列分解を行う
  - 空所の値を予測する



12.5 推薦システムにおける学習

- 潜在因子によるデータ表現の考え方

$$x_{mn} = w_{1n}v_{1m} + w_{2n}v_{2m} + \cdots + w_{kn}v_{km}$$



12.5 推薦システムにおける学習

- 行列分解の方法
  - $\boldsymbol{X}$-$\boldsymbol{U}\boldsymbol{V}^T$ の最小化問題を解く

$$\min_{\boldsymbol{U},\boldsymbol{V}} \frac{1}{2}\|\boldsymbol{E}\|_{\mathrm{Fro}}^2 = \min_{\boldsymbol{U},\boldsymbol{V}} \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{U}\boldsymbol{V}^T\|_{\mathrm{Fro}}^2$$

空欄を値0とみなしてしまっている

  - 値が存在する要素だけに限って二乗誤差を最小化

$$\min_{\boldsymbol{U},\boldsymbol{V}} \sum_{(i,j)\in\Omega} (x_{ij} - \boldsymbol{u}_i\boldsymbol{v}_j^T)^2 + \lambda_1\|\boldsymbol{U}\|_{\mathrm{Fro}}^2 + \lambda_2\|\boldsymbol{V}\|_{\mathrm{Fro}}^2$$

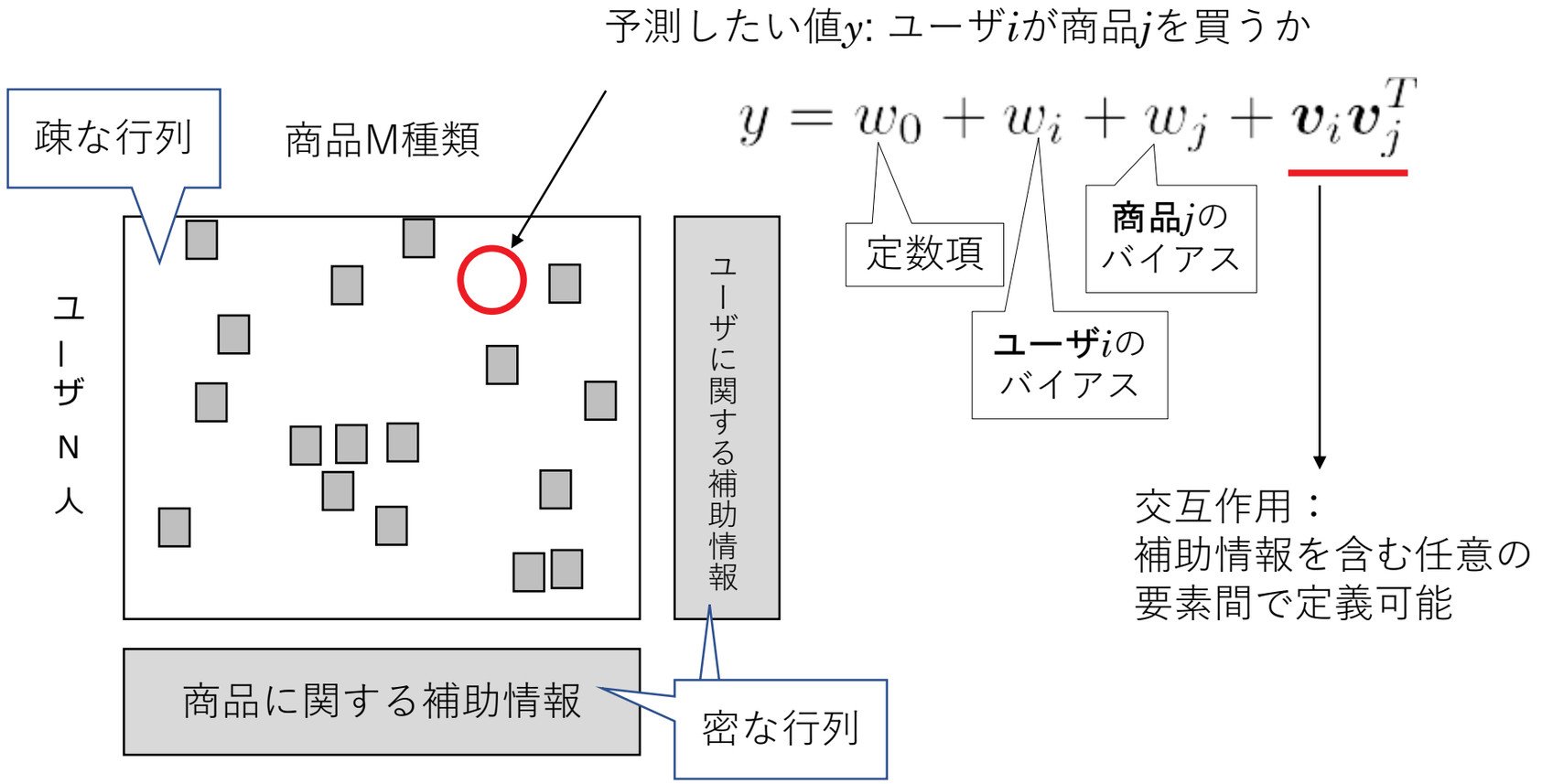
正則化項

Fro（フロベニウスノルム）：行列の要素の二乗和の平方根

  - $\boldsymbol{U},\boldsymbol{V}$ の要素を非負に限定したものが非負値行列因子分解 (NMF：Nonnegative Matrix Factorization)

12.5 推薦システムにおける学習

- Factorization Machine
  - 補助情報を予測に取り入れることができる



まとめ

- パターンマイニングは有用な規則性を発見する
- アプリオリアルゴリズム
  - 出現頻度の高い項目集合を見つける
  - 出現頻度に基づき、有用な規則を見つける
- 行列分解
  - 低次元ベクトル表現を見つけることにより、未知の値の予測を行う

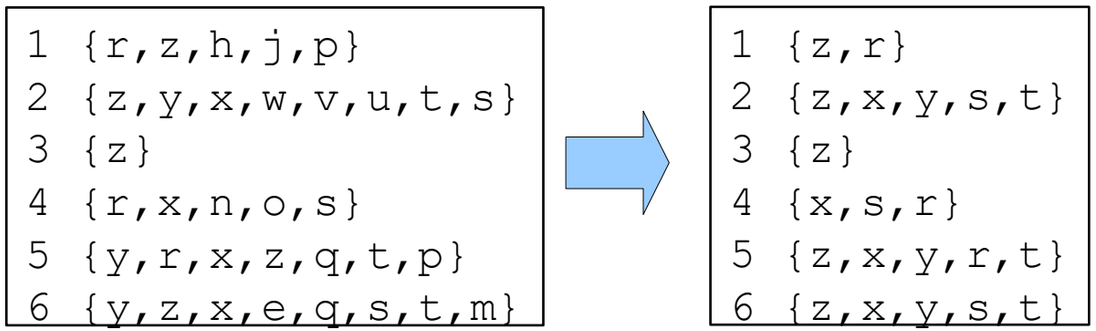
補足

12.4 FP-Growthアルゴリズム

- Aprioriアルゴリズムの高速化
  - ◆ トランザクションをコンパクトに表現し、重複計算を避ける
  1. トランザクションの前処理
     - トランザクションを、出現する特徴名の集合に変換
     - 出現頻度順にソート
     - 低頻度特徴をフィルタリング
  2. prefixを共有する木構造(FP木)に順次挿入
  3. FP木を用いて項目集合の出現頻度を高速計算

12.4 FP-Growthアルゴリズム

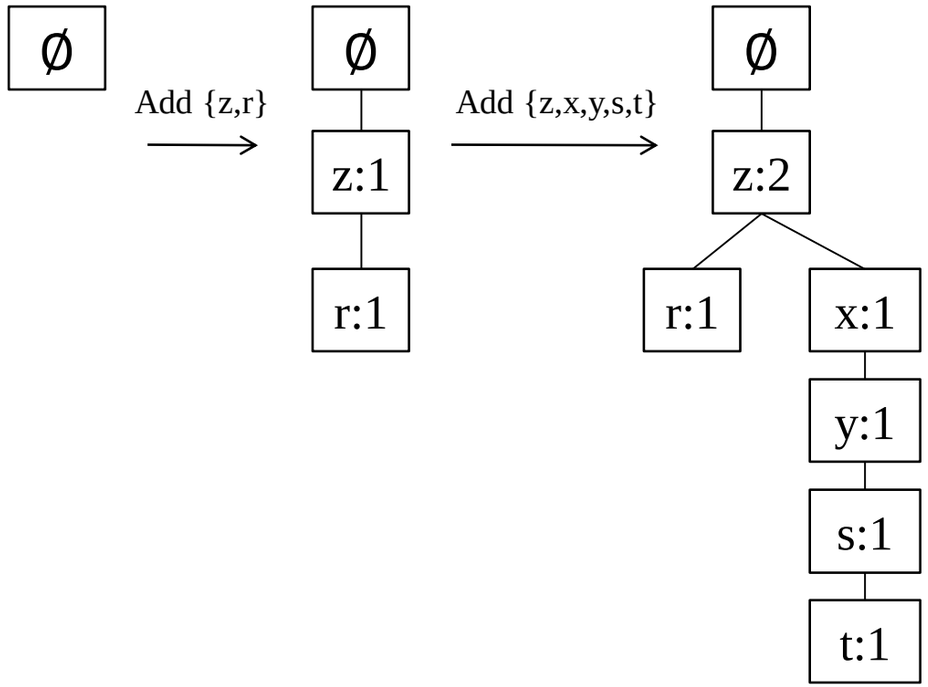
1. トランザクションの前処理
   - トランザクションを、出現する特徴名の集合に変換
   - 出現頻度順にソート
   - 低頻度特徴をフィルタリング



12.4 FP-Growthアルゴリズム

2. prefixを共有する木構造(FP木)に順次挿入

- ソート、フィルタリング後のトランザクションデータを順次FP木に挿入



12.4 FP-Growthアルゴリズム

3. FP木を用いて項目集合の出現頻度を高速計算