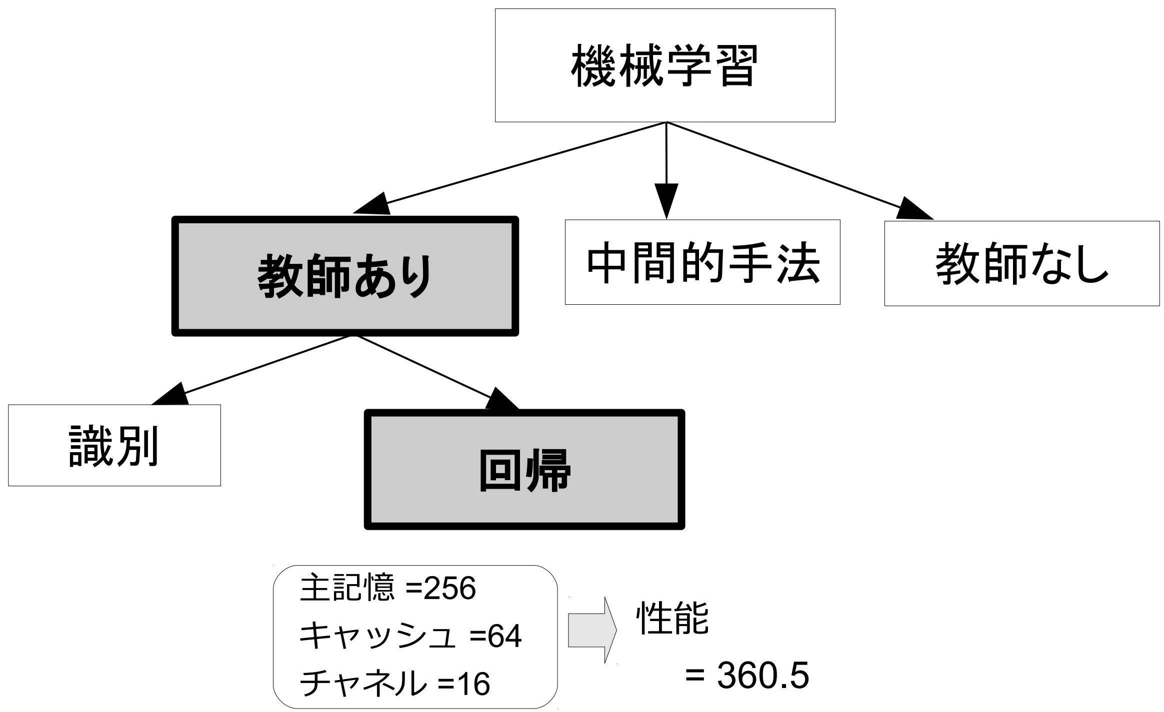
Section 3

- 回帰（8章）
- 教師なし学習（10～11章）

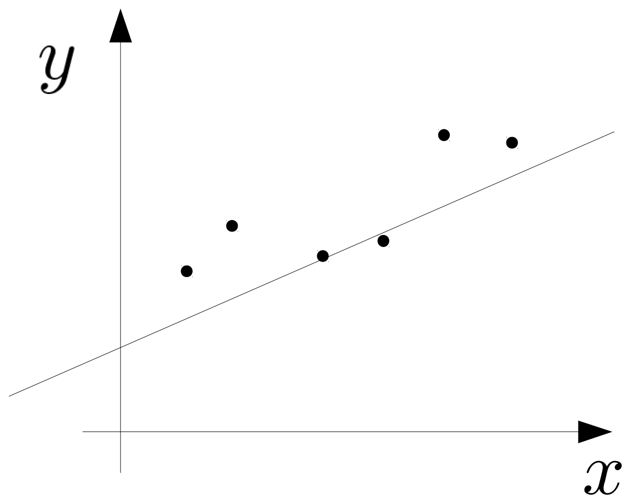
8. 回帰

- 問題設定
  - 教師あり学習
  - 数値入力　→　数値出力



8.2 線形回帰

- 目標：なるべく誤差の少ない直線を求める



- 線形回帰の定義
  - 入力 $\boldsymbol{x}$ から出力 y を求める回帰式を 1 次式に限定
  - 学習データから係数 $\boldsymbol{w}$ を求める

$$\hat{c}(\boldsymbol{x}) = \sum_{i=0}^{d} w_i x_i$$

8.2 線形回帰

- 最小二乗法による係数の推定
  - 推定の基準：誤差の二乗和 E を最小化

$$E(\boldsymbol{w}) = \sum_{i=1}^{N} (y_i - \hat{c}(\boldsymbol{x}_i))^2$$

$$= (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})$$

X: 全学習データを並べた行列

w: 係数のベクトル表現

  - $\boldsymbol{w}$ で微分した値が 0 となるのは

$$\boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) = 0$$

$$\Leftrightarrow \boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y}$$

*w*が解析的に求まる

8.2 線形回帰

- 最小二乗法の精度向上

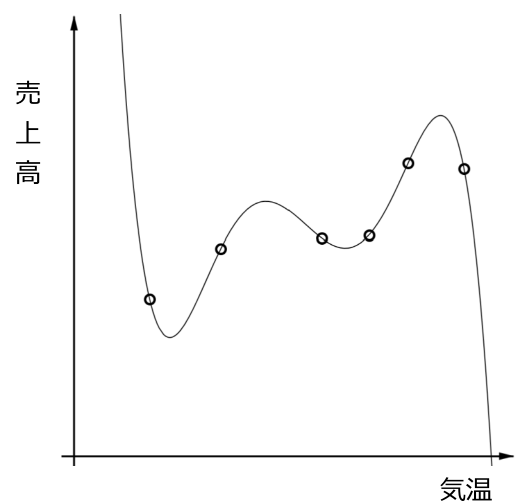
例　$\phi(x) = (1, x, x^2, \dots, x^b)$

  - 基底関数 $\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_1(\boldsymbol{x}), \dots, \phi_b(\boldsymbol{x}))$ を考える

$$\hat{c}(\boldsymbol{x}) = \sum_{j=0}^{b} w_j \phi_j(\boldsymbol{x})$$

  - 係数が線形であれば、最小二乗法が適用可能

- 問題点
  - 汎化性能の低下



8.2　線形回帰

- 正則化の考え方
  - 正則化項の導入

    $\rightarrow$ 複雑なパラメータ $\boldsymbol{w}$ （過学習）の回避

    – L1 ノルム $|\boldsymbol{w}|$ ： 0 となるパラメータが多くなる　ラッソ

    – L2 ノルム $\|\boldsymbol{w}\|^2$ ： パラメータを 0 に近づける　リッジ
- リッジ回帰
  - 誤差の二乗和に L2 ノルム正則化項を加える

$$E(\boldsymbol{w}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) + \underline{\lambda \boldsymbol{w}^T \boldsymbol{w}}$$

λ：誤差の二乗和と正則化項とのバランス

$$\boldsymbol{w} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda \boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

w が解析的に求まる

8.2 線形回帰

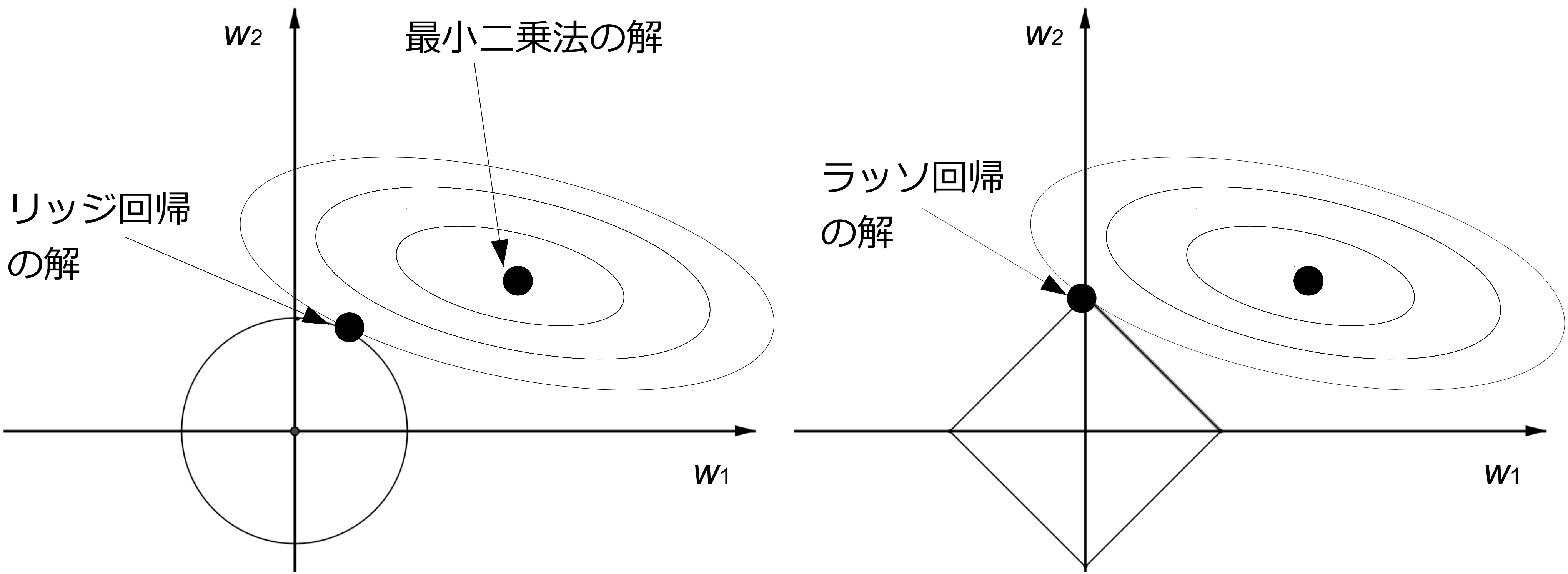
- ラッソ回帰
  - 誤差の二乗和に L1 ノルム正則化項を加える

$$E(\boldsymbol{w}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}) + \underline{\lambda \sum_{j=1}^{d} |w_j|}$$

  - 微分不可能な点があるため、解析的に解を求めることができない
    - 適当な初期重みから始め、リッジ回帰で上界を押さえる逐次更新アルゴリズムを用いる

8.2　線形回帰

- リッジ回帰とラッソ回帰

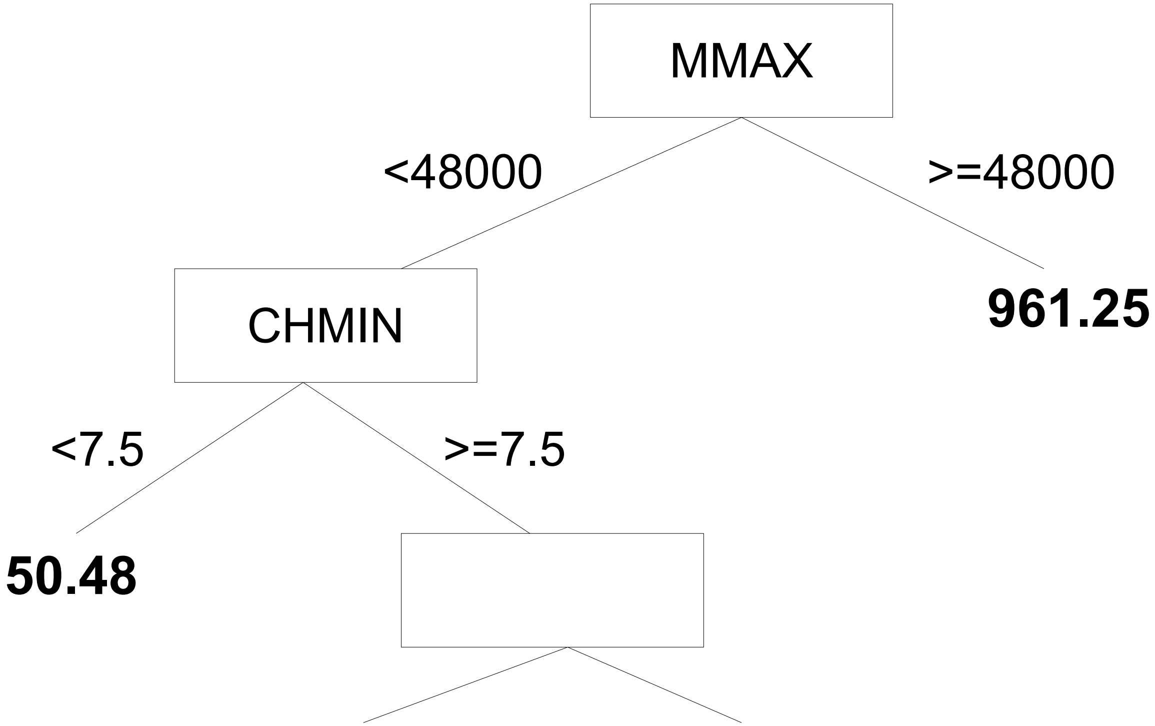


パラメータを 0 に
近づけている

0 となるパラメータを
多くしている

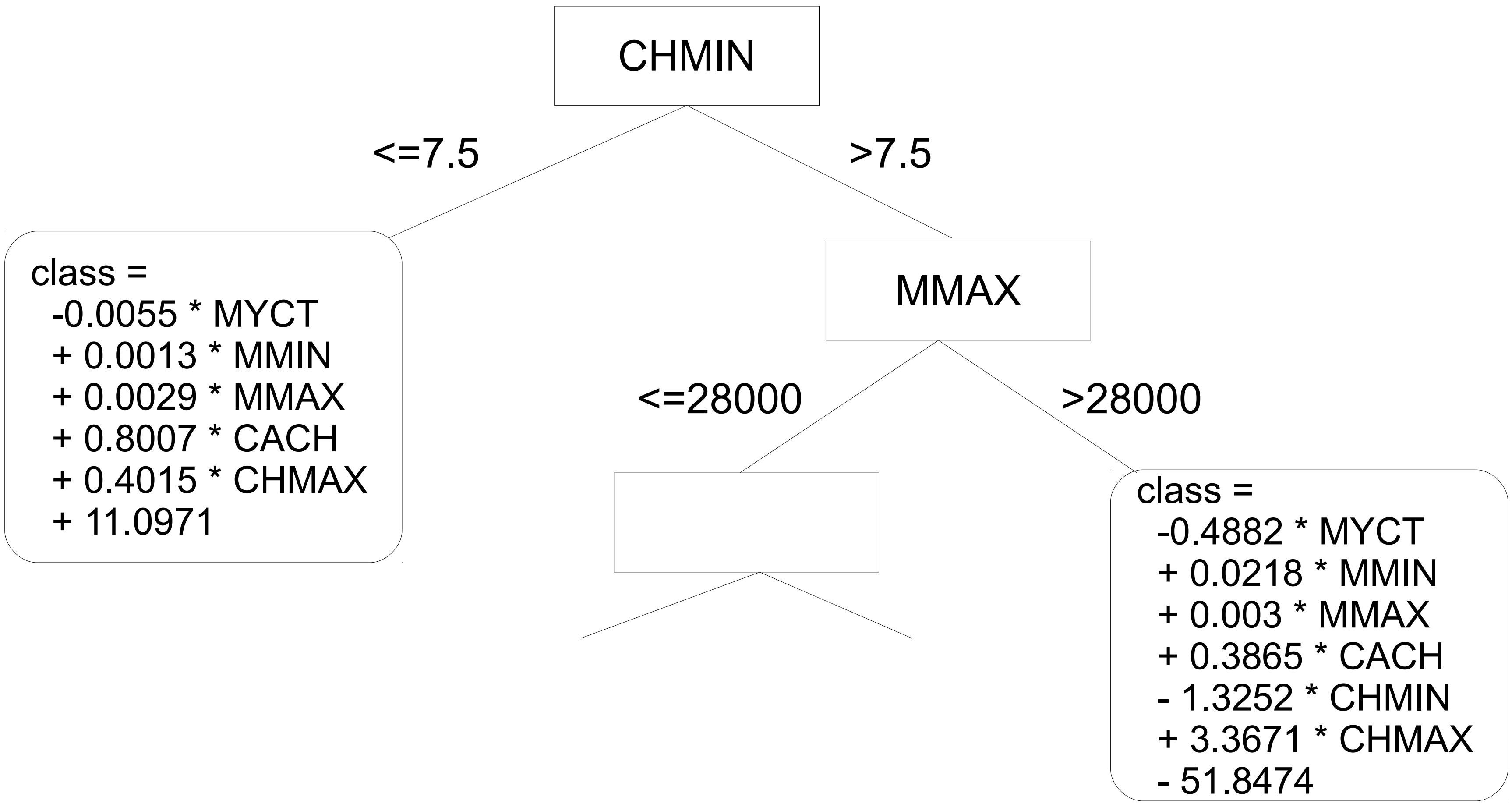
8.4 回帰木

- 回帰木とは
  - 識別における決定木の考え方を回帰問題に適用
  - ターゲット値の分散が小さくなるように分割



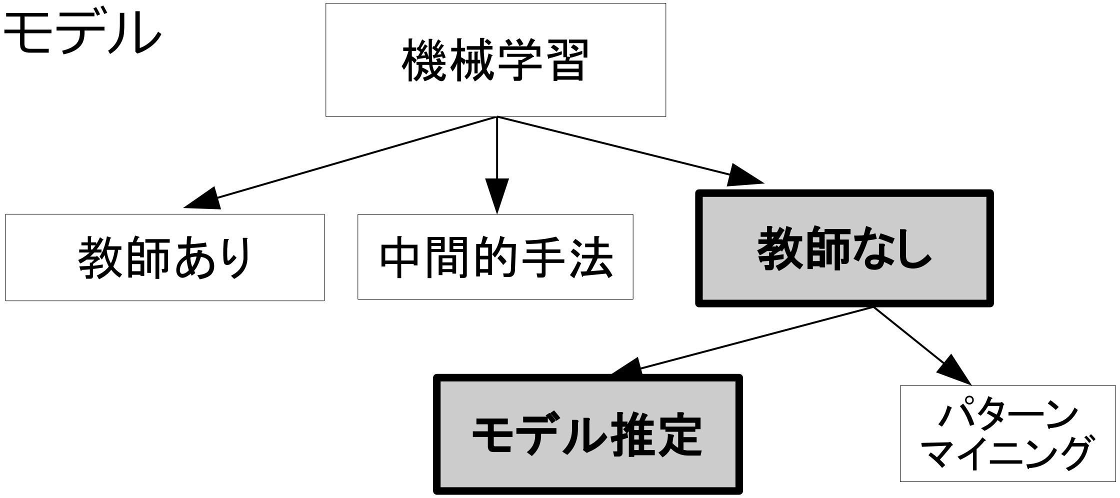
8.4 回帰木

- モデル木とは
  - リーフを線形回帰式にした回帰木



10. モデル推定

- 問題設定
  - 教師なし学習
  - 数値入力　→　クラスモデル
    - クラスモデルの例
      - クラスの分割結果
      - クラスの確率モデル

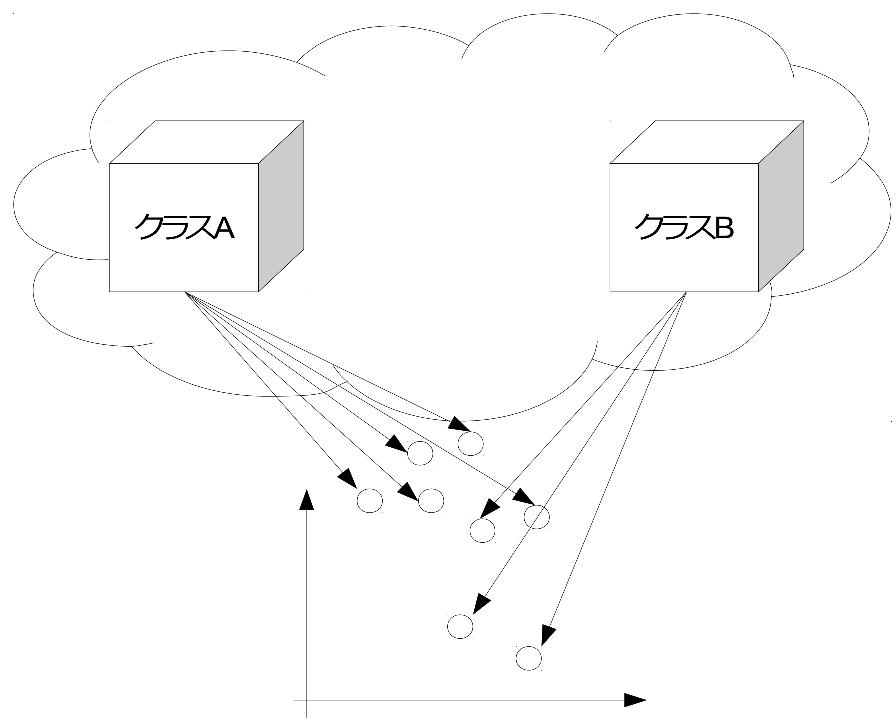


10.1 問題の定義

- 学習データ

$$\{\boldsymbol{x}^{(i)}\} \qquad i = 1, .., N$$

- 問題設定
  - 特徴ベクトル $\boldsymbol{x}$ が生成された元のクラスの性質を推定する



10.2 クラスタリング

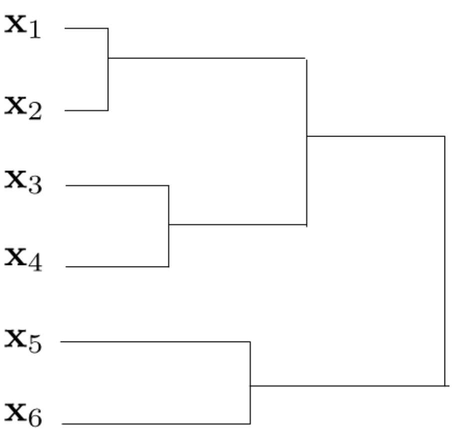
- クラスタリングとは
  - 対象のデータを、
    内的結合（同じ集合内のデータ間の距離は小さく）
    　　と
    外的分離（異なる集合間の距離は大きく）
    が達成されるような部分集合に分割すること

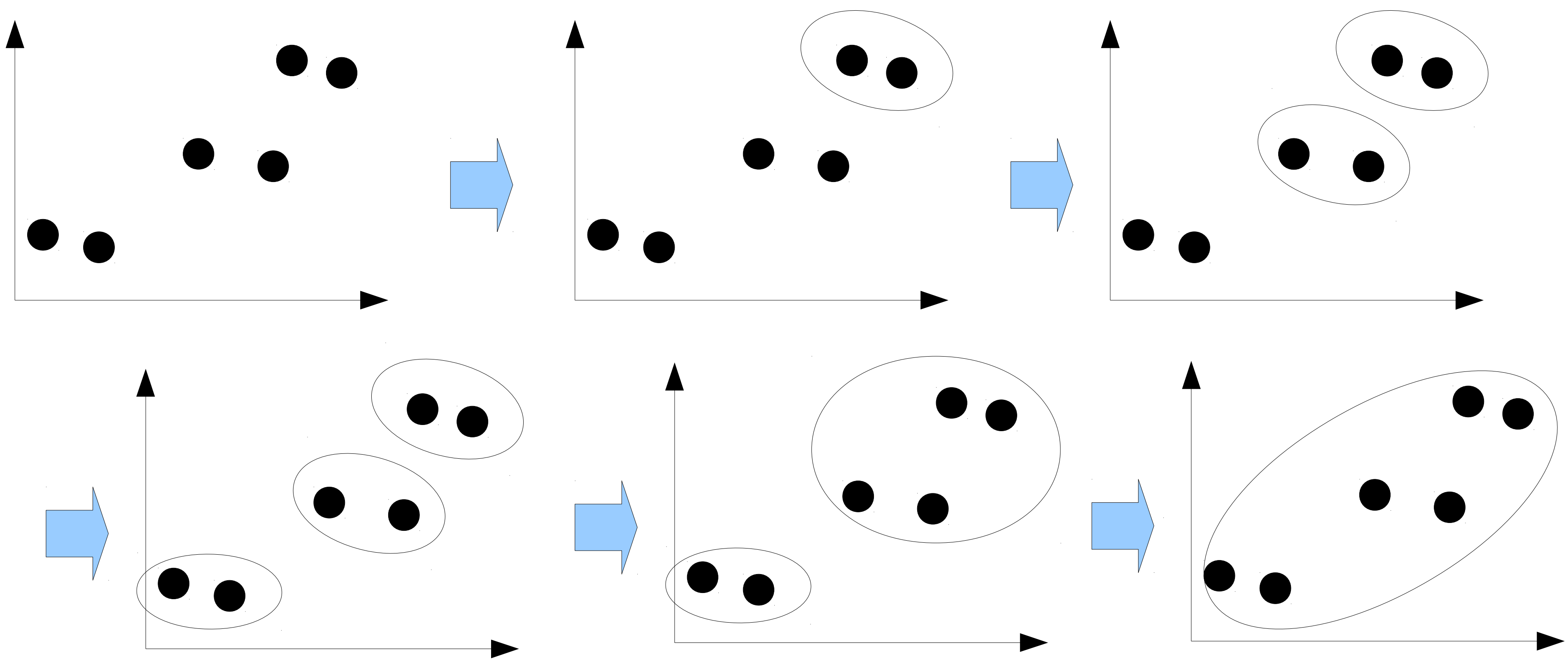
要するに塊を見つけること

- クラスタリング手法の分類
  - 階層的手法
    - ボトムアップ的にデータをまとめてゆく
  - 分割最適化手法
    - トップダウン的にデータ集合を分割してゆく

10.2.1 階層的クラスタリング

- 階層的クラスタリングとは
  1. 1 データ 1 クラスタからスタート
  2. 最も<u>近接する</u>クラスタをまとめる
  3. 全データが 1 クラスタになれば終了





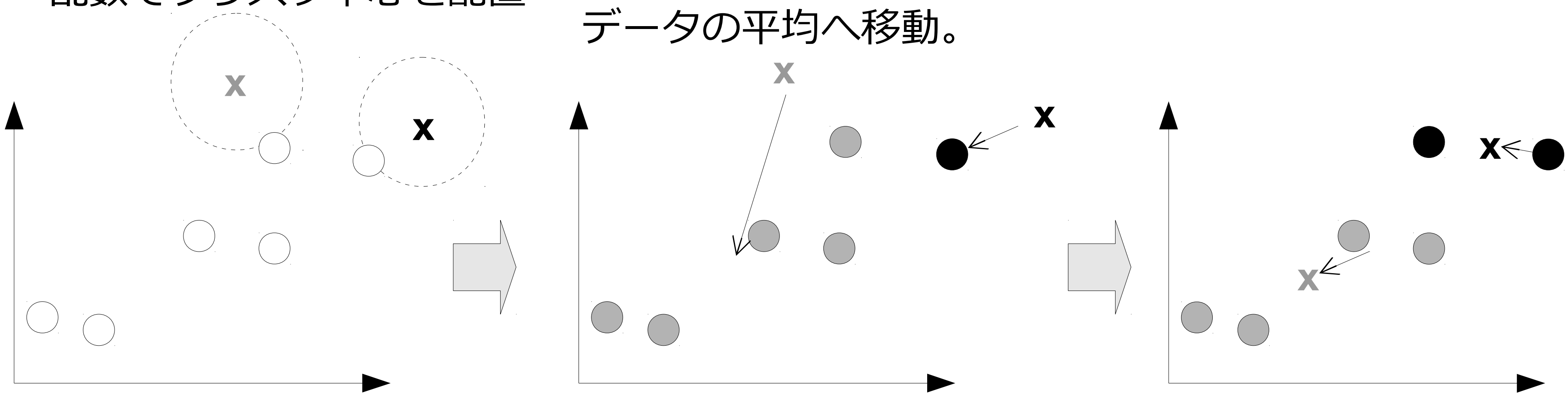
10.2.2 分割最適化クラスタリング ― k-means アルゴリズム―

- k-Means アルゴリズム
  1. 分割数 k を予め与える
  2. 乱数で k 個のクラスタ中心を設定し、逐次更新

k=2 とし、初期値として
乱数でクラスタ中心を配置

全データを近い方のクラスタ
中心に所属させる。そして、
クラスタ中心を所属している
データの平均へ移動。

左の処理を繰り返す。

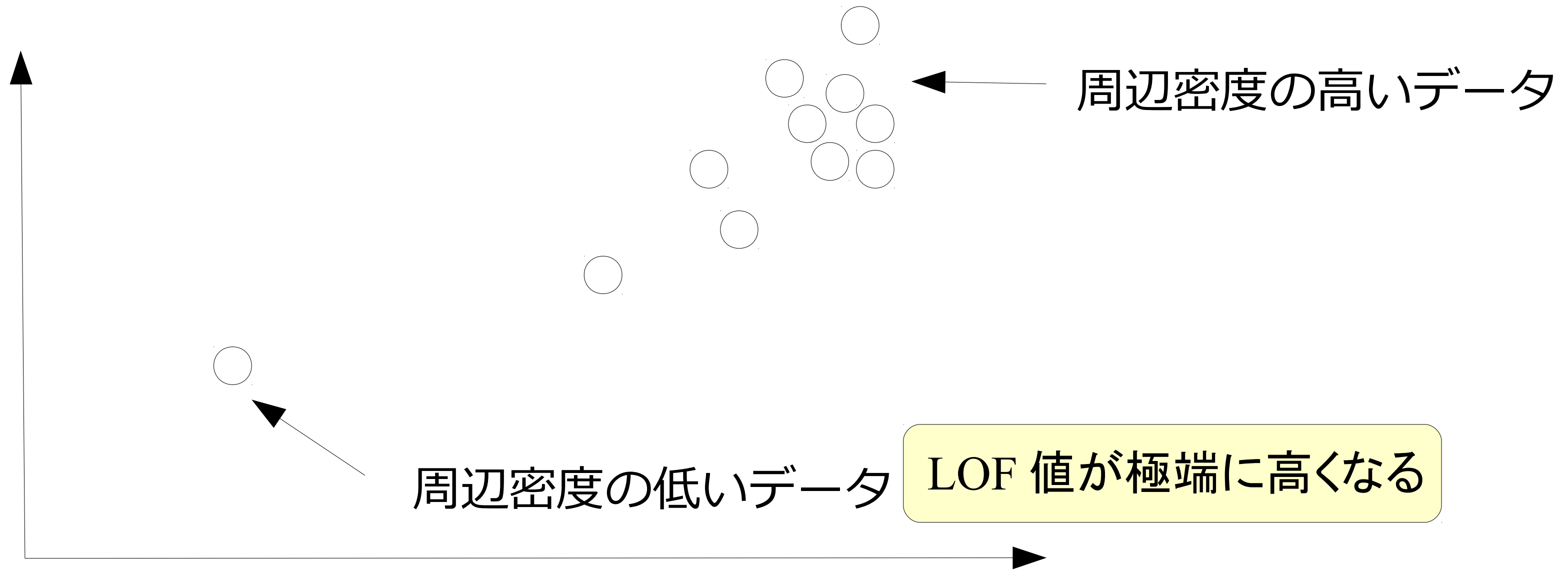


10.3 異常検出

- 異常検出とは
  - 正常クラスのデータと、それ以外のデータとのクラスタリング
  - 外れ値検知、変化点検出、異常状態検出など
  - 対象データが静的・動的で手法が異なる
- 外れ値検知（静的異常検出）
  - データの分布から大きく離れている値を見つける
  - 手法
    - 近くにデータがないか、あるいは極端に少ないものを外れ値とみなす
    - 「近く」の閾値を、予め決めておくことは難しい

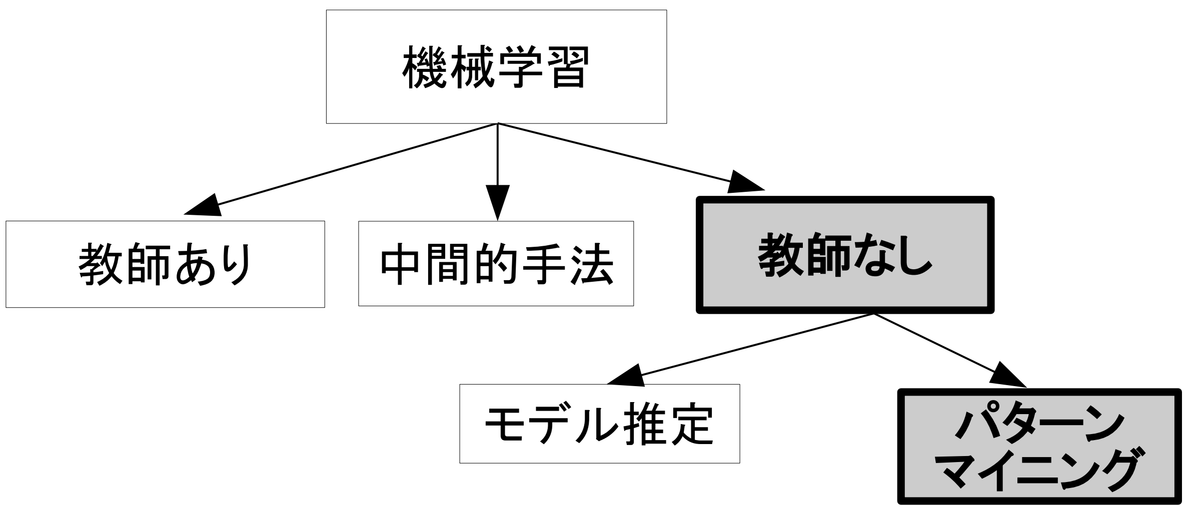
10.3 異常検出

- 局所異常因子による外れ値検知
  - 周辺密度
    - あるデータの周辺の他のデータの集まり具合
  - 局所異常因子 (LOF: local outlier factor)
    - 近くの k 個のデータの周辺密度の平均と、あるデータの周辺密度との比



12 章 パターンマイニング

- パターンマイニングの問題設定
  - 入力：カテゴリ特徴の教師なしデータ
  - 出力：頻出項目、連想規則、未観測データ



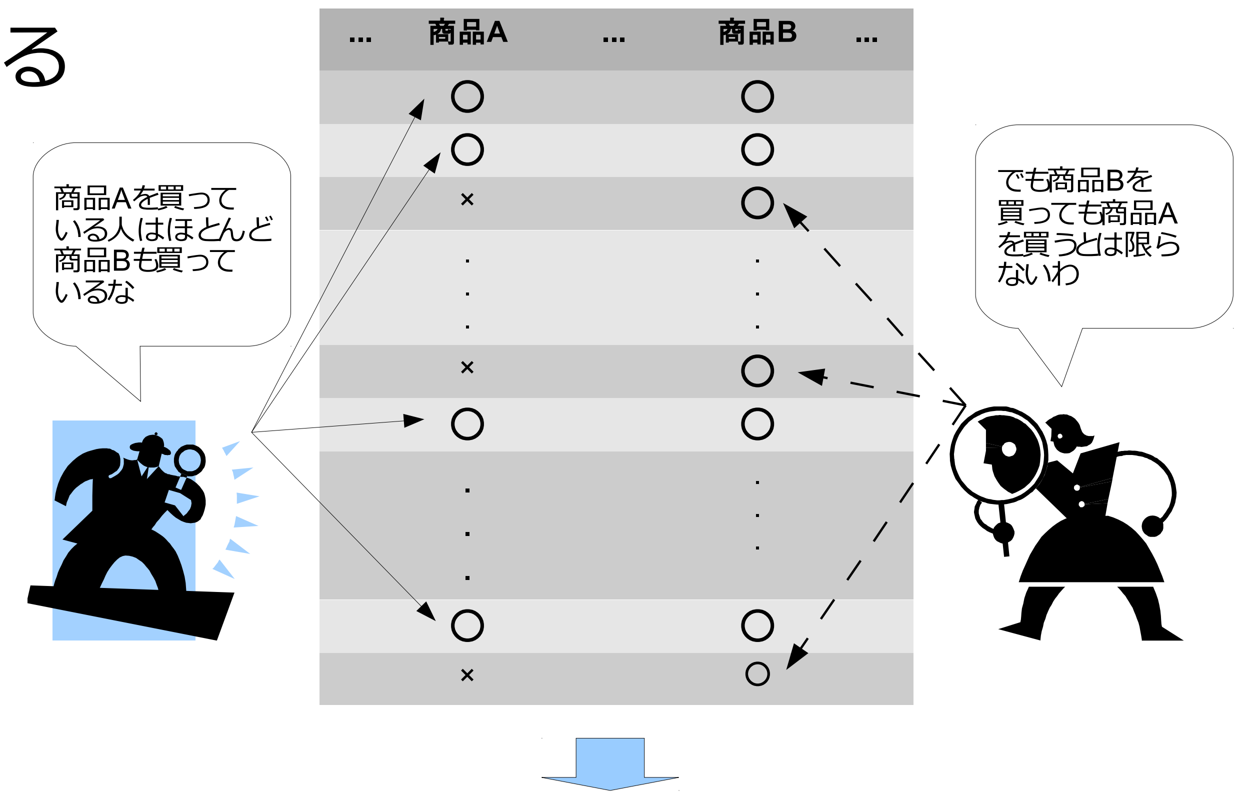
No.	ミルク	パン	バター	雑誌
1	t	t		
2		t		
3				t
4		t	t	
5	t	t	t	
6	t	t		

問題の定義

- 学習データ

$$\{\mathbf{x}^{(i)}\} \quad i = 1, .., N$$

- 問題設定
  - データ集合中で、一定頻度以上で現れるパターンを抽出する



商品Aを購入」⇒(ならば) 商品Bを購入」

11.2 Apriori アルゴリズムによる頻出項目抽出

- 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

- 支持度
  - 全トランザクション数 T に対して、ある項目集合 (items) が出現するトランザクションの割合

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

11.2 Apriori アルゴリズムによる頻出項目抽出

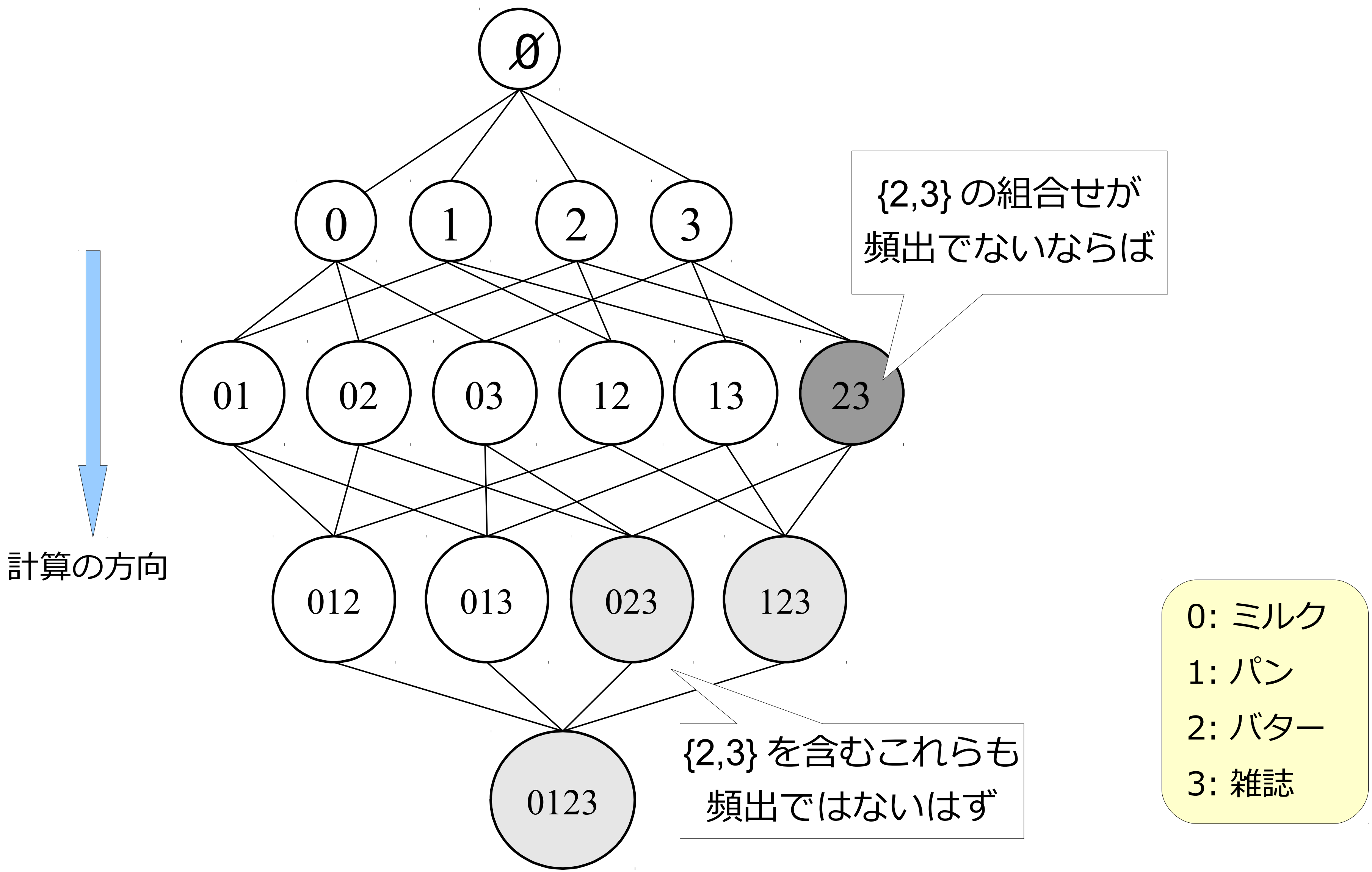
- バスケット分析の目的
  - 支持度の値が閾値以上の項目集合を抽出したい
- バスケット分析の問題点
  - すべての可能な項目集合について、支持度を計算することは現実的には不可能

項目集合の種類数は 2 の商品数乗

商品数 1,000 の店なら 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

11.2 Apriori アルゴリズムによる頻出項目抽出



11.3 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 も買う傾向が高い」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

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

条件部 A が起こったときに結論部 B が起こる割合

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

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

11.3 連想規則抽出

- 支持度・確信度・リフト値
  - 砂糖について卵の関連購買が以下の場合：
    - 支持度 20%　確信度 70%　リフト値 30.0
  - 「全体顧客の 20% が砂糖と卵を一緒に購入しており、砂糖購入者の 70% が砂糖と卵を一緒に購入している」ということになる。この時のリフト値 30.0 は、「顧客全体の中で卵をいきなり購入するよりも、砂糖を買って卵を買う確率が 30 倍大きい」という意味を表している。

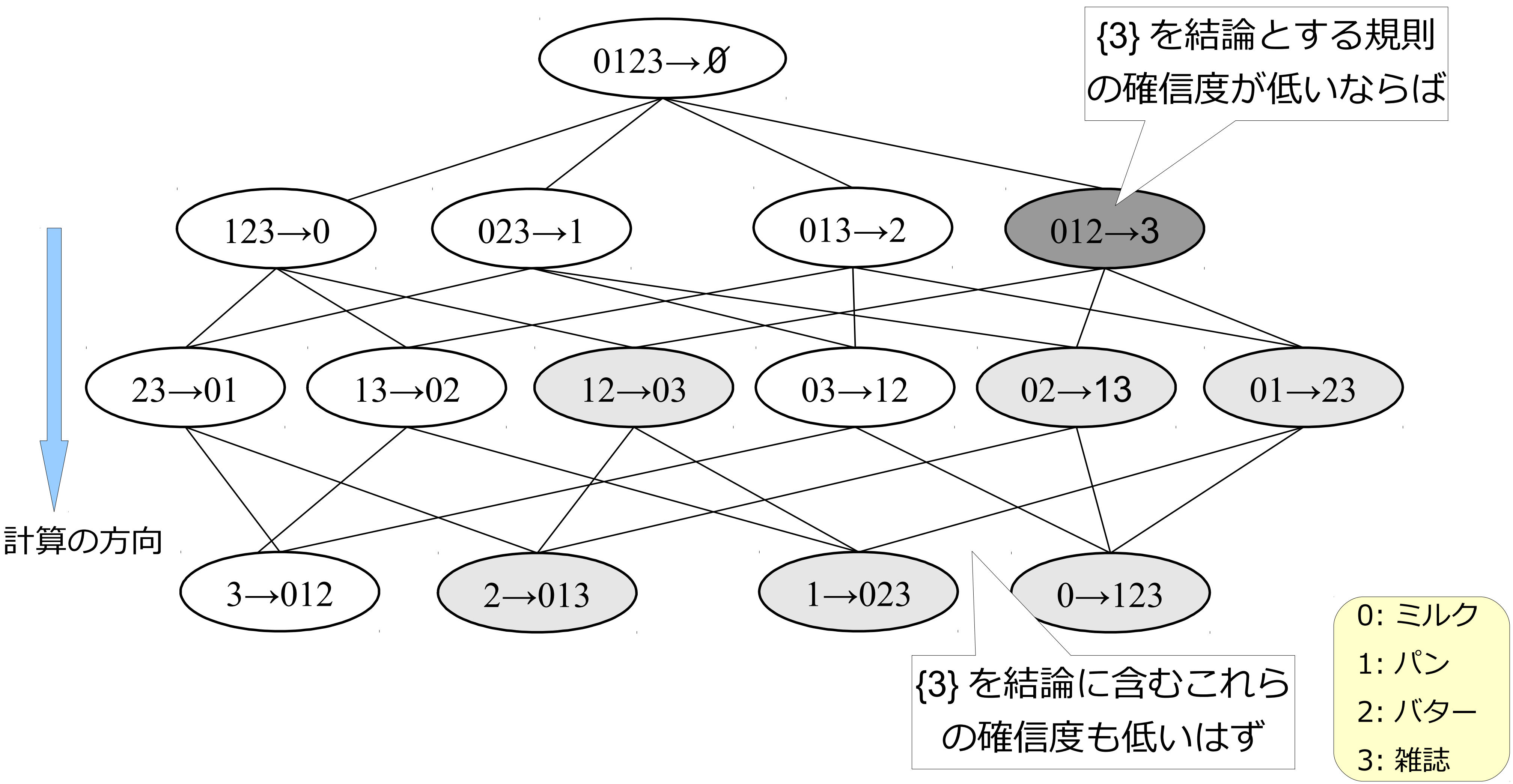
https://kotobank.jp/word/リフト値-801685　（訳語改変）

11.3 連想規則抽出

- 連想規則抽出の手順
  - 頻出項目集合を求める
  - 項目集合を条件部、空集合を結論部とした規則を作成する
  - 条件部から結論部へ項目を 1 つずつ移動し、評価する

11.3 連想規則抽出

- a priori 原理に基づく探索



Section3 のまとめ

- 回帰
  - 数値を出力するモデルの学習
  - 正則化によって、出力に寄与している特徴を見極めることができる
- 教師なし学習：モデル推定
  - クラスタリング：データのまとまりを発見
  - 異常検出：近くのデータとの周辺密度の違いを利用
- 教師なし学習：パターンマイニング
  - 頻出項目や有用な規則を高速に抽出
  - 推薦システムなどに応用可能