クラスタリング・分類・相関ルール分析に関する調査

文理学部情報科学科
5419045 高林 秀

2021 年 8 月 9 日

概要

本稿は、今年度データ科学 2 のレポート課題として、「クラスタリング」、「分類（決定木）」および「相関ルール分析」の各分野に関して、各手法の特徴やアルゴリズムの説明、解説を行うものである。また、1 年次に学習した latex を使用した pdf 作成の復習も兼ねるものである。

1　目的

本稿は、今年度データ科学 2 の最終レポート課題として「クラスタリング」、「分類（決定木）」および「相関ルール分析」の各分野に関して、各手法の特徴やアルゴリズムの説明、数式等の解説を行うものである。それぞれの分野の各手法の特徴やアルゴリズムに言及した上、使われている数式の解説を記載する。

2　基礎導入

2.1　データマイニングとは

まず、本稿で取り扱う分野の大元であるデータマイニングについて軽く説明する。データマイニングとはビックデータや企業の顧客情報など大量のデータが格納されたデータベースから機械学習や、統計計算等の手法でデータを分析し、そこから新たに有用な知識を発見しようという技術である。データマイニングでは、データから得られる情報を分析しそこから得ることのできる知識を取り出すことである。つまり、そこから先の得た知識をどう利用するかは人間の判断に委ねられている。すなわち、データマイニングで行うのは知識の発掘であり、発掘した知識が有用か、またどう活用するかは人間が判断する、ということだ。

大きく分けてデータマイニングは以下のように分類することができる。

1. 仮説検証的データマイニング
   （a）推定
   （b）分類
2. 知識探索的データマイニング
   （a）相関ルール分析 (アソシエーションルール分析)
   （b）クラスタリング

上記の分類はあくまで大別であり、実際は手法により当てはまる分野は異なる。

仮説検証的データマイニングは、仮説に沿ってある課題を解決するためにデータ分析を行うことを示す。機

械学習の手法のみならず従来までの統計的手法が使用されることも多い。

知識探索的データマイニングは、データベース上のデータから特定のルールや規則、パターンといった知識を探索するためにデータ分析を行うことを示す。こちらは機械学習やディープラーニング等の手法が多く用いられる。本稿で扱うのは、上記に示した「分類」、「クラスタリング」、「相関ルール分析」の 3 つである。

2.2　機械学習について

機械学習とは、コンピュータがある問題とその答えを使用して学習を行い、データに潜むパターン等を識別、発見する技術である。この機械学習は大きく 3 つに系統が別れている。初めに「教師あり学習」、次に「教師なし学習」、最後に「強化学習」である。そしてそこから更に、求める結果や手法によって「分類」「回帰」「クラスタリング」「次元削減」「Q-Learning」と細かく分割される。

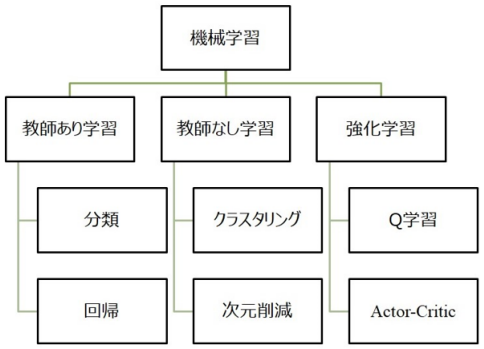

図 1　機械学習の枠組み

今回説明する、分類は上記分類の「教師あり学習」に、クラスタリングは「教師なし学習」に該当する。教師あり学習と教師なし学習についての具体的な説明は本稿では省略するが、以下にその件に関して記載したレポートの URL を添付する。

- 機械学習に関する説明をした過去のレポート:https://drive.google.com/file/d/1wyoRO20UmgiYsxAjkFKLzxhwPv
  view?usp=sharing

3　クラスタリング

クラスタリングとは、データのグループ分けを行うことをいい、クラスタリングの結果で生じた各データ集合をクラスタと呼ぶ。このクラスタには同じ様な性質をもったデータが集められている。前章で示したとおりクラスタリングは与えられたデータから計算によって自動的にデータの分類、グルーピングを行うので、機械学習では教師なし学習に分類される。

クラスタリングは、知識探索的な手法であるので得られた結果は何らかの基準にしたがってクラスタが形成されている。よって、客観的な証拠としてクラスタリングを用いるのは適切ではなく、データの要約など知識や知見を得るために使用するのが適切である。

なお、今後の説明で登場するクラスタ内距離やクラスタ間距離などの用語については、今回の課題の説明範

囲外なので以下のレポートを参照いただきたい。

- クラスタリングの基本に関する説明レポート:https://drive.google.com/file/d/1JP3DnVNmH3kOtEULt73u79zJTb
  view?usp=sharing

クラスタリングは以下に示すように、階層的クラスタリングと非階層的クラスタリングの 2 つに大別される。

1. 非階層的クラスタリング
   （a）k-平均法 (k-means)
2. 階層的クラスタリング
   （a）凝集型 (階層的併合型)
   （b）分岐型

その他にも、クラスタ間の密度に基づく手法や、格子に基づく手法などに分けることもできるが本稿では省略する。

3.1 非階層的クラスタリング

非階層的クラスタリングは、予め分割するクラスタ数 K を定めたとき、クラスタ内距離を最小にしつつかつクラスタ間距離を最大にするようにクラスタを決定する方法である。このクラスタ数 K は人間が定めるハイパラメータであり、この K の値によってクラスタリング結果は大きく変化する可能性がある。このクラスタ数 K を自動的に決定する手法はいくつか考案されているが、本稿での説明は省略する。

以下は、非階層的クラスタリングの流れの概要である。

1. クラスタ数 K を定める。
2. データを k 分割する。
3. なにかの基準や手法を用いてデータ分割が改善するように、データ分割を繰り返す。
4. 3 の結果、改善される度合いが小さくなればクラスタリング終了。

非階層的クラスタリングの長所として、計算量の少なさが挙げられるだろう。これは後述する手法からも分かるとおり、予め分割するクラスタ数 K にしたがってデータを分けていく。したがって、階層的クラスタリングよりも計算量が小さくなるという利点がある。よって、データ量が大きい場合 (例：ビックデータ分析) のデータ分析に適した手法とされている。

反対に、短所として「初期値依存性」が挙げられるだろう。これは、クラスタリングを行う際、最初に K 個の初期中心点を選択する必要があり、この初期中心点の選択によってクラスタリング結果が大きく変化するという問題である。

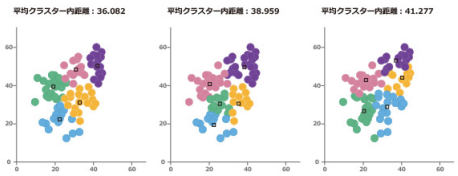

図29.初期値の違いによる結果の違い

図 2　非階層的クラスタリングの初期値依存性の例

出典：
https://www.albert2005.co.jp/knowledge/data_mining/cluster/non-hierarchical_clustering

よって、非階層的クラスタリングを行う際には何回かクラスタリングを実行し、平均クラスタ内距離が最小となる初期中心点を選択する必要が生じる。

次は非階層的クラスタリングの代表的手法である k-平均法 (k-means) について説明する。

3.1.1　k-平均法 (k-means)

k-平均法では以下の目的関数を利用する。この目的関数を最小化するようにデータの分割を行いクラスタを形成していく手法である。

$$\Sigma_{k=1}^{K}\Sigma_{x_i \in C_k}(x_i - c_k)^2$$

k-平均法のアルゴリズムは以下に示すとおりである。

1. クラスタ数 K 定める。
2. ある手法に基づいてデータ集合を K 個のクラスタに分割し、その結果を C_n とする
3. 2 の結果を C_{pre} として保持しておく
   以降は、C_n と C_{pre} が一致するまで繰り返しをする。
4. 各クラスタの重心を計算し、それを新たなクラスタの中心点とする
5. 各データと中心点との距離を計算し、最も近い中心点のクラスタへの割当を行う→この結果を C_n とする：目的関数の値が最小化するようにクラスタを更新する
6. C_n と C_{pre} が一致した場合はクラスタリングを終了し、そうでない場合はもう一度 3 を行う。
   クラスタリング終了時
7. C_n を結果として出力する

k-平均法では、各クラスタの重心とクラスタ内距離の総和の局所最適解[*1]を求めていく。この局所最適解が収束するまで、クラスタ割当の更新と重心の再計算を行う。

なお初期クラスタの形成に関しては次の 3 通りの手法が挙げられる。

- 各データに対して、ランダムに 1 から K 個のいずれかのクラスタに割当を行う方法
- データ全体からランダムに K 個のデータを選択し、それぞれ s1 sK とする。s1 sK 以外の各データは、s1 sK の中で最も近い si(s1 sK のうちから 1 つ) のクラスタに割り当てる方法。

[*1] ある範囲における関数の最小値（極小値）のこと。その関数の真の最小値（極小値）は大域的最適解と呼ばれる。

- データのある空間からランダムに K 個の点を生成し選択する。それぞれ s1 sK とする。各データは s1 sK の中で最も近い si(s1 sK のうちから 1 つ) のクラスタに割り当てる方法。

3.2 階層的クラスタリング

前章の非階層的クラスタリングとは異なり、階層的クラスタリングでは予めクラスタ数 K を定める必要はない。階層的クラスタリングでは、似た性質を持つデータ同士を 1 つずつグルーピングしていくようにクラスタを形成する。データを 1 つ 1 つ比較ししていき、似ているデータ同士、およびクラスタを新たなクラスタとして併合する。そうすると、最終的に階層構造のようなクラスタが出来上がるので、階層的クラスタリングと呼ばれている。

先に示したが、階層的クラスタリングには凝集型と分岐型に大別することができる。

■凝集型　凝集型は階層併合的クラスタリングとも呼ばれ、各データを 1 つのクラスタとして考え各クラスタをボトムアップに併合することで、新たなクラスタを形成する手法である。手順の概要は下記に示すとおり。

1. 各データをそれぞれ 1 つのクラスタと見なす
2. クラスタの数が 1 つになるまで次の操作を繰り返す
   （a）それぞれクラスタ間で距離を算出する
   （b）最も距離が小さいペア同士を新たなクラスタとして併合する

図で示すと以下のようになる。左から順番に進行する。

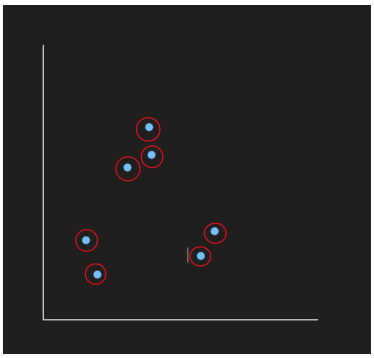

図 3　1. 各データをそれぞれ 1 つのクラスタと見なす

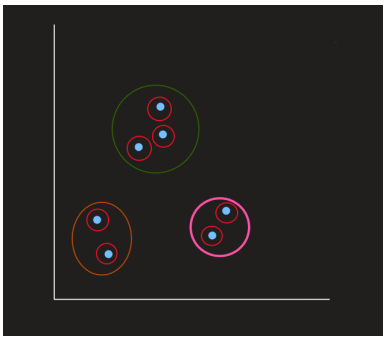

図 4　2 (a),(b) クラスタの併合

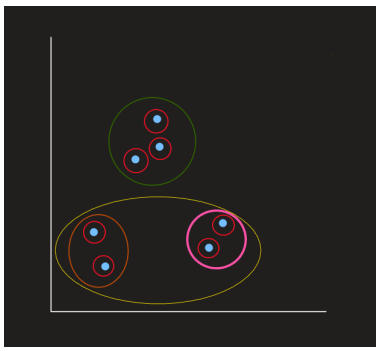

図 5　3. クラスタの併合その 2

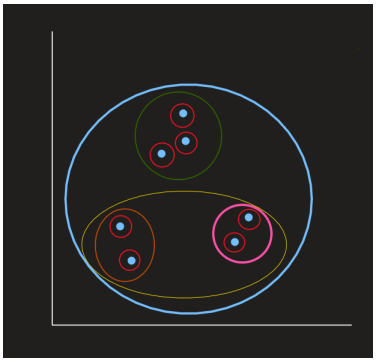

図 6　4. クラスタ数が 1 になったのでクラスタリング終了

より形式的に示す以下の様なアルゴリズムが出来上がる。

```
X:クラスラリング対象のデータ集合
C := {{x} | x ∈ X}:クラスタの集合 C
while |C| > 1 {
  <C_a, C_b>:=argmin_C_i,C_j ∈ C(D(C_i, C_j))
  C := (C\{C_a, C_b}) ∪ {C_a ∪ C_b}
}
```

凝集型クラスタリングの具体的な計算手法には、利用するクラスタ間距離に応じて下記のものが存在する。

- ウォード法 (ward 法)
- 群平均法
- 重心法
- 最短距離法 (単リンク法, 単連結法)
- 最長距離法 (完全リンク法, 完全連結法)
- 群間平均法

- C_x：クラスタの集合
- $D(C_x, C_y)$：クラスタ間距離

- $M(C_x)$：クラスタの重心

■最短距離法　最短距離法は、異なるクラスタに属している 2 データ間距離の最小値をクラスタ間距離とする手法である。すなわち、最近点距離である。

$$D(C_g, C_h) = min_{i \in C_g, j \in C_h}(dist(i, j)$$

■最長距離法　最長距離法は、異なるクラスタに属している 2 データ間距離の最大値をクラスタ間距離とする手法である。すなわち、最遠点距離である。

$$D(C_g, C_h) = max_{i \in C_g, j \in C_h}(dist(i, j))$$

■群間平均法　群間平均法は、異なるクラスタに属している 2 データ間距離の平均値をクラスタ間距離とする手法である。すなわち、平均距離である。

$$D(C_g, C_h) = \frac{1}{|C_g| \times |C_h|} \Sigma_{i \in C_g, j \in C_h} dist(i, j)$$

■重心法　重心法は、各クラスタの重心を求め、その距離をクラスタ間距離とする手法である。すなわち、重心間距離である。

$$D(C_g, C_h) = dist(M(C_g), M(C_h))$$

■ウォード法　ウォード法は「重心との誤差の改善度合い」に着目し計算する手法である。

$$D(C_g, C_h) = E(C_g \cup C_h) - E(C_g) - E(C_h) = E(C_g \cup C_h) - (E(C_g) + E(C_h))$$

なお、$E(C) = \Sigma_{x \in C} dist(x, M(C))^2$：重心からの距離の二乗和

※ $D(C_g, C_h)$：クラスタまたはデータの「併合後の誤差」–「併合前の誤差」

■分岐型　凝集型とは異なりトップダウンに各データを分割することでクラスタを形成する。データ集合全体を 1 つのクラスタと見なし、徐々に小さいクラスタへ分割していく手法である。現在のところあまり使用されていない手法と言える。

なお、本稿では分岐型の具体的な説明は省略するが、具体的手法の例として「Diana 法」が存在する。

3.3　非階層的クラスタリングと階層的クラスタリングの長所と短所

前章でも述べたが、非階層的クラスタリングの長所として、計算量の少なさが挙げられるだろう。反対に、短所として「初期値依存性」が挙げられるだろう。

階層的クラスタリングの長所として、前章の計算手法で紹介したとおり重心を用いない手法であれば様々な類似度を利用することができる点が挙げられる。加えて、クラスタを併合する際の順番が分かりやすいので、細かくクラスタの変化の様子を追うことができる。これは、階層的クラスタリングの結果として使用する「デンドログラム」の存在が大きな要因になっている。後述する決定木のように、クラスタ併合の課程が樹形図で可視化することができるので、非階層的クラスタリングよりも結果の説明がしやすい。反対に、欠点としてデータ数が多いと樹形図の把握が困難になり、理解困難になる。また、各事例間に類似度の差が小さい場合、樹形図の鎖状化の発生により、全体的なクラスタを把握するのが難しくなる点がある。その他にも、使用する手法によって様々な問題点を抱えている。以下その一例を示す。

- 郡平均法を使用した場合：デンドログラムの反転現象が起こる可能性がある。これはクラスタ間距離の現象により、デンドログラムが交差してしまう現象である。

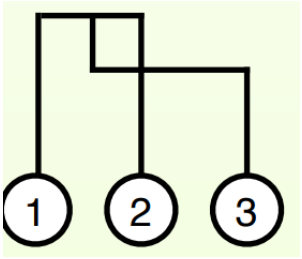

図 7　デンドログラムの反転現象の例

出典：https://www.kamishima.net/archive/clustering.pdf

- 最短距離法：空間濃縮の発生 ⇒ 併合後の新クラスタは次の併合の対象となる可能性が加速度的に増加する。
- 最長距離法：空間拡張 $Rightarrow$ 併合後の新クラスタは次の併合の対象となる可能性が加速度的に減少する。

4　分類:決定木構築 TDIDT

分類は先に示したとおり、機械学習の教師あり学習に区分される。分類学習では、事前に定められたカテゴリ、およびクラスに入力データを分類することを行う。その一手法として決定木というものが存在する。

4.1　決定木について

決定木は「木構造を利用した機械学習手法」である。分類を行う決定木を「分類木」、回帰 (連続値の予測) を行う決定木は「回帰木」と呼ばれる。決定木では任意の属性の属性値による条件分岐によって、データを徐々に分割することで結果を出力する。したがって、生成される木構造の枝は分割の結果ラベルを、葉は予測、分類されるクラスの結果を、各ノードは属性に関する分割テスト含んでいる。

決定木を使用する例として、ミカンとリンゴを分類する場合を考える。下記の図のように、ミカンの画像を 4 枚、リンゴの画像を 2 枚の計 6 枚の画像データセットがあるとする。これを、ある条件 A を定め、それに当てはまるもの、そうでないものを分割する。この操作を分割テストという。分割テストの結果によって次の分

割テスト行うか否かが決定され、最終的に、ミカンとリンゴが図のように分類される。これが決定木の大まかな流れである。

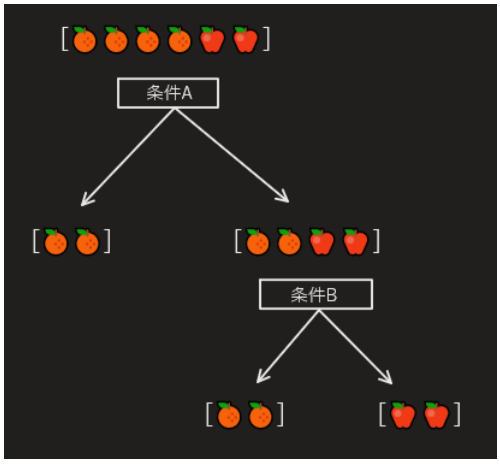

図 8　ミカンとリンゴの分類木

以下簡単に決定木の長所、短所をまとめる。

表 1　決定木のメリット・デメリット

メリット	デメリット
・学習結果の可読性が高く結果の根拠を説明しやすい ・データの前処理が少なく済むことが多い ・予測時に必要な計算量が小さい ・回帰、分類の両方に対応可能	・条件分岐が複雑になるほど過学習しやすい ・精度が突出して良いわけではない

4.2　TDIDT について

決定木構築手法の代表例として、TDIDT 法 (Top Down Induction of Decision Trees) がある。この手法は、比較的簡単にコンパクトかつ正確な決定木を作ろうとする概念、すなわちヒューリスティックスに基づいた決定木構築法である。ここでいう Top Down とは、木の根から作るという意味で、Induction とは、帰納推論、すなわちデータからモデルを構築するという意味である。

TDIDT の大まかな流れは次の通り。

1. 根となるデータ集合を用意する。
2. ある基準 (情報利得・情報利得比・ジニ係数) で属性を選ぶ。
3. 選ばれた属性の属性値ごとにデータを分割する。
4. 各枝に対して、3 を繰り返す。
5. 停止条件を満たす場合、その枝の分割を終了とする。

分割属性の選択は、できるだけその属性で分割したときにクラス分布が偏れば、しっかりとデータを分類でき

るので良い分割基準となる。

このとき、分割属性の選択基準として考えられる情報利得・情報利得比・ジニ係数について説明する。なお、本稿では自己情報量等の基本的な数式の説明は省略する。この部分に関しては、以下のリンクから決定木構築のレポートを参照いただきたい。

- 決定木構築のレポート:https://drive.google.com/file/d/1QviNpUqGr6yGqpJYqp1gyfWtksf35V6o/view?usp=sharing

4.2.1 情報利得

情報利得とは一言で言えば「クラスの偏りがどの程度進んだか」を表す数値である。データセット D のおける属性 A の情報利得の計算式は下記。

$$Gain_A(D) = H(D) - H_A(D)$$

この値が大きいほど、分割テストに適した良い属性ということになる。D を分割する前のエントロピーが $H(D)$ で、属性 A での分割後のエントロピーが $H_A(D)$ であり、それぞれ下記式で示すことができる。

$I(c, D)|c$: クラス, D : データセット (データ集合) とすると、

$$I(c, D) = -\log_2 P_D(c)$$

※ $P_D(c)$: D 中のデータのクラスが c となる確率

$$H(D) = \Sigma_{c \in C} P_D(c) \times I(c, D) = -\Sigma_{c \in C} P_D(c) \log_2 P_D(c)$$

$$H_A(D) = \Sigma_{a \in A} p_D(a) \times H(D_a)$$

$P_D(a)$: D 中のデータの属性 A の属性値が a となる確率。

すなわち、クラス分布を偏らせるためには、情報利得が大きい属性を選択すれば良い。すなわち**情報利得 = 分割前のエントロピー - 分割後の各エントロピーの重み付き平均**ということになる。

5 相関ルール分析