アルゴリズム論 2

第 5 回: 凸包 (1)

関川 浩

2016/10/12

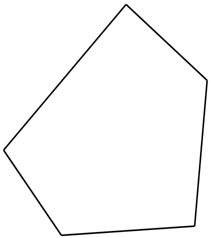
1. 凸集合・凸包の定義と性質
    - 凸集合の定義と性質
    - 凸包の定義と性質
    - 端点
    - 支持超平面
    - 有限個の点の凸包
    - 凸包の応用
2. 2 次元凸包構成アルゴリズム
    - 問題の設定
    - Graham のアルゴリズム
    - 包装法

凸集合の定義

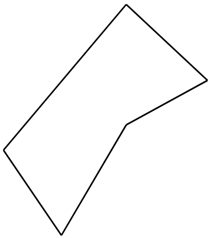
定義 1 (凸集合)

集合 $S \subset \mathbb{R}^d$ が凸

$\iff \forall P, Q \in S, 0 \le \forall t \le 1$ に対し $tP + (1-t)Q \in S$



凸集合



凸集合ではない

凸集合の性質

命題 1

S_λ $(\forall\lambda \in \Lambda)$ が凸のとき $\displaystyle\bigcap_{\lambda\in\Lambda} S_\lambda$ は凸

証明

P, $Q \in \displaystyle\bigcap_{\lambda\in\Lambda} S_\lambda$ と t $(0 \leq t \leq 1)$ を任意に取る

$\Longrightarrow \forall\lambda \in \Lambda$ に対し P, $Q \in S_\lambda$

S_λ は凸だから $tP + (1-t)Q \in S_\lambda$

よって, $tP + (1-t)Q \in \displaystyle\bigcap_{\lambda\in\Lambda} S_\lambda$

したがって $\displaystyle\bigcap_{\lambda\in\Lambda} S_\lambda$ は凸 ■

凸包の定義と性質

定義 2 (凸包)

$S \subset \mathbb{R}^d$ に対し, $\displaystyle\bigcap_{S \subset X \subset \mathbb{R}^d,\ X\ \text{は凸}} X$ を S の凸包といい $\mathrm{CH}(S)$ と表す

命題 2

$\mathrm{CH}(S)$ は S を含む最小の凸集合
すなわち, $S \subset S' \subset \mathbb{R}^d$ かつ S' が凸なら $\mathrm{CH}(S) \subset S'$

証明

$S \subset S'$ かつ S' は凸だから,

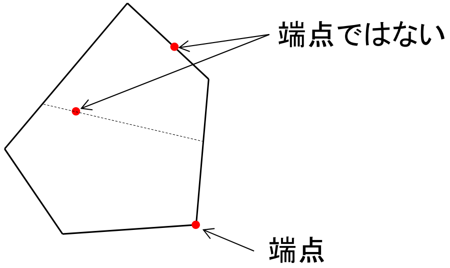
$$\mathrm{CH}(S) = \bigcap_{S \subset X \subset \mathbb{R}^d,\ X\ \text{は凸}} X = \left(\bigcap_{S \subset X \subset \mathbb{R}^d,\ X\ \text{は凸}} X \right) \cap S' \subset S' \quad \blacksquare$$

端点

定義 3 (端点)

S: 凸集合

$$P \in S \text{ が端点} \iff \forall Q, R \in S \text{ s.t. } Q \neq P, R \neq P \text{ に対し } P \notin \overline{QR}$$



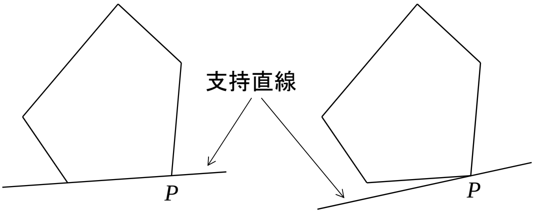
支持超平面

定義 4 (支持超平面)

$P \in S$: 凸集合
π: P を通る超平面
S が π の片側に含まれるとき, π を支持超平面という

注意

P が凸集合 S の端点ならば, S との共通部分が P だけであるような超平面が存在



有限個の点の凸包

$\mathbb{R}^2$ 内の n 点の凸包は多角形

- 頂点数は高々 n (n 多角形)

$\mathbb{R}^3$ 内の n 点の凸包は多面体

- 端点数 (頂点数) を h とすると, $h \le n$
- 辺の数は高々 $3h - 6 \le 3n - 6$
- 面の数は高々 $2h - 4 \le 2n - 4$

証明

辺の数を e, 面の数を f とする

- $h - e + f = 2$ (Euler の多面体公式)
- $3f \le 2e$ (各面の辺の数は 3 以上, 各辺はちょうど二つの面が共有)

より ■

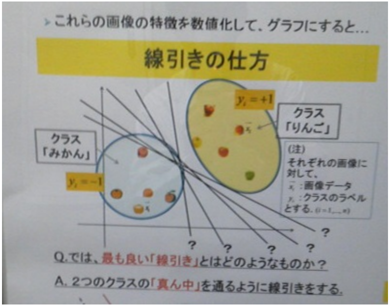
凸包の応用 (1/2)

パターン認識, データの分類

$S, T \subset \mathbb{R}^2$

$\mathrm{CH}(S) \cap \mathrm{CH}(T) = \emptyset$ (線形分離可能) ならば S と T は本質的に異なるとみなすことが多い

$\Longrightarrow$ 凸包構成アルゴリズムが不可欠



矢部研究室のパネル「データを分類する」の一部

凸包の応用 (2/2)

最遠点対

$P_1, \ldots, P_n \in \mathbb{R}^2$: 与えられた有限個の点

問題: P_i, P_j 間の距離が最大となるような対 (P_i, P_j) を求めよ

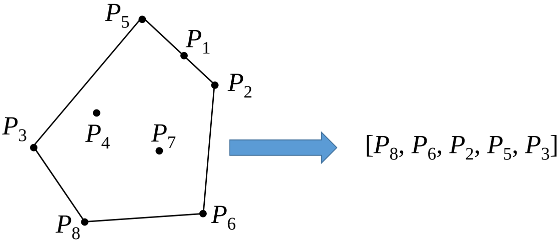
$\Longrightarrow$ $P_1, \ldots, P_n$ の凸包が既知ならばキャリパー法を用いて $O(h)$ で解ける (h は凸包の頂点数)

問題の設定

問題 (2 次元の凸包構成)

入力: 有限個の点 $P_1, \ldots, P_n \in \mathbb{R}^2$ $(n < \infty)$

出力: 凸包 (凸多角形) を構成する頂点を反時計周りに並べたリスト $[P_{i_1}, \ldots, P_{i_m}]$



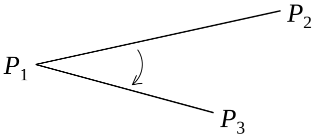
基本処理

$P_1(x_1, y_1)$, $P_2(x_2, y_2)$, $P_3(x_3, y_3) \in \mathbb{R}^2$

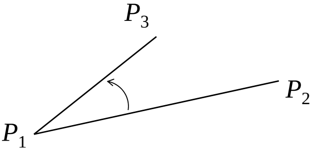
P_1 を中心に $\overline{P_1P_2}$ を θ 回転させると $\overline{P_1P_3}$ に重なるとき $(-\pi < \theta \leq \pi)$, 以下のどれかの判定

(a) $-\pi < \theta < 0$　　(b) $0 < \theta < \pi$　　(c) $\theta = 0$ あるいは π

$$\begin{vmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{vmatrix} \text{ の符号が } \begin{cases} - & \Longleftrightarrow \quad \text{(a)} \\ + & \Longleftrightarrow \quad \text{(b)} \\ 0 & \Longleftrightarrow \quad \text{(c)} \end{cases}$$



(a)



(b)

Graham のアルゴリズム (1/4)

Grahama のアルゴリズム

入力: 有限個の点 $P_1(x_1, y_1), \ldots, P_n(x_n, y_n) \in \mathbb{R}^2$

出力: 凸包の頂点リスト $[P_{i_1}, \ldots, P_{i_m}]$ (反時計周り)

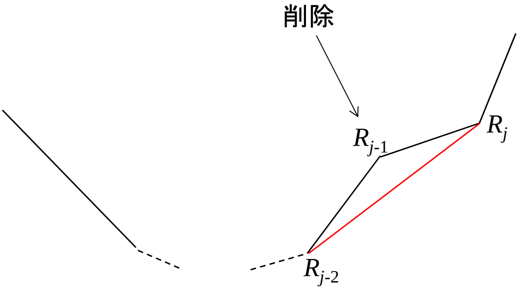
1. P_i のうち凸包の頂点となる点を一つとり Q_1 とする
(例: x 座標最小の点, 複数あればその中で y 座標最大の点)
2. 入力された点 (Q_1 以外) を Q_1 からの偏角が小さい順にソートした結果を $Q_2, \ldots, Q_m$ とする
偏角が同じ点があるときは Q_1 から一番遠い点のみ採用
$m \leq 3$ なら $[Q_1, \ldots, Q_m]$ を返して終了
3. $\overline{Q_{i-2}Q_i}$ と $\overline{Q_{i-2}Q_{i-1}}$ の位置関係を調べ凸包を構成 (詳細は次ページ)

Graham のアルゴリズム (2/4)

Step 3 の詳細

$j \leftarrow 1; \quad i \leftarrow 1;$
while $(i \leq m)$ $\{$
    $R_j \leftarrow Q_i;$
    while ($j \geq 4$ かつ $\overline{R_{j-2}R_{j-1}}$ を $\overline{R_{j-2}R_j}$ に重ねる回転は
            時計周り (角度 0 を含む)) $\{$
        $R_{j-1} \leftarrow R_j;$
        $j \leftarrow j - 1;$
    $\}$
    $j \leftarrow j + 1; \quad i \leftarrow i + 1;$
$\}$
return $[R_1, \ldots, R_j];$

Graham のアルゴリズム (3/4)



Step 3

Graham のアルゴリズム (4/4)

計算量: $O(n \log n)$

- Step 1: $O(n)$
- Step 2 (ソート部分): $O(n \log n)$
- Step 3: $O(n)$

注意

2 次元凸包構成アルゴリズムを用いてソートが可能

1. $x_1, \dots, x_n \in \mathbb{R}$ に対して $(x_1, x_1^2), \dots, (x_n, x_n^2) \in \mathbb{R}^2$ の凸包の頂点リストを構成
(すべての点が凸包の頂点となることに注意)
2. $x_1, \dots, x_n$ の最小値が先頭になるよう書き換えた頂点リストの第 1 座標が $x_1, \dots, x_n$ のソート結果

$\Longrightarrow$ 2 次元凸包構成アルゴリズムは $\Omega(n \log n)$

包装法 (1/3)

包装法 (1/2)

入力: 有限個の点 $P_1(x_1, y_1), \ldots, P_n(x_n, y_n) \in \mathbb{R}^2$

出力: 凸包の頂点リスト $[P_{i_1}, \ldots, P_{i_m}]$ (反時計周り)

1. P_i のうち x 座標最小の点をとり Q_1 とする
   複数あるときはその中で y 座標最小の点を Q_1 とする
2. Q_1 を原点としたとき, 入力された点 (Q_1 以外) の中で
   y 軸の負方向からの偏角が最小の点を Q_2 とする
   偏角が同じ点があるときは Q_1 から一番遠い点を Q_2 とする
   $i = 2$ とおく

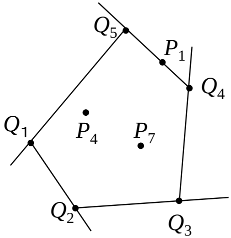
注意: 入力点はすべて $\overline{Q_1 Q_2}$ の片側にある

包装法 (2/3)

包装法 (2/2)

3. Q_i を中心として半直線 $Q_{i-1}Q_i$ を反時計周りに回転したとき最初に半直線に載る点を Q_{i+1} とする
(複数の点が載ったら Q_{i-1} から一番遠い点を Q_{i+1} とする)
4. もし $Q_1 = Q_{i+1}$ ならリスト $[Q_1, \ldots, Q_i]$ を返して終了
そうでなければ $i \leftarrow i + 1$ として Step 3 へ

注意: 入力点はすべて $\overline{Q_{i-1}Q_i}$ の片側にある



包装法 (3/3)

計算量: $O(hn)$ (h: 凸包の頂点数)

- Step 1 (Q_1 を探す手間): $O(n)$
- Step 2 (Q_2 を探す手間): $O(n)$
- Step 3 ($i \geq 3$ のとき Q_i を探す手間): $O(n)$

よって, 凸包の 1 頂点を探す手間は $O(n)$

$\Longrightarrow$ 全体の計算量は $O(hn)$

注意: 最悪の場合 $h = \Omega(n)$ なので $O(hn) = O(n^2)$