

劣モジュラカットとその応用

Submodularity Cuts and Applications

河原吉伸* 永野清人† 津田宏治‡ Jeff A. Bilmes§
Yoshinobu Kawahara Kiyohito Nagano Koji Tsuda

Abstract: Several key problems in machine learning can be formulated as submodular set function maximization. We present herein a novel algorithm for maximizing a submodular set function under a cardinality constraint — the algorithm is based on a cutting-plane method and is implemented as an iterative small-scale binary-integer linear programming procedure. It is well known that this problem is **NP-hard**, and the approximation factor achieved by the greedy algorithm is the theoretical limit for polynomial time. As for (non-polynomial time) exact algorithms that perform reasonably in practice, there has been very little in the literature although the problem is quite important for many applications. Our algorithm is guaranteed to find the exact solution in finite iterations, and it converges fast in practice due to the efficiency of the cutting-plane mechanism. Moreover, we also provide a method that produces successively decreasing upper-bounds of the optimal solution, while our algorithm provides successively increasing lower-bounds. Thus, the accuracy of the current solution can be estimated at any point, and the algorithm can be stopped early once a desired degree of tolerance is met. We evaluate our algorithm on sensor placement and feature selection applications showing good performance.

Keywords: 劣モジュラ集合関数最大化, カutting・プレイン法, 機械学習

1 序論

機械学習における基本問題の中には、特徴選択や能動学習などのように、有限な集合の中から、何らかの評価関数を最大化するような部分集合を選択する事を目的とするものが数多く存在する。更にそれらの評価関数の多くは、「劣モジュラ性」と呼ばれる、連続関数における凸性に対応する性質を持つ集合関数となっている事が知られている。劣モジュラ性を持つ集合関数は「劣モジュラ関数」と呼ばれ、 V を有限集合とするとき、任意の $S, T \subseteq V$ に対して、 $f(S) + f(T) \geq f(S \cap T) + f(S \cup T)$ を満たす集合関数 f として定義される [2, 4]。代表的な例としては、エントロピー関数、対称相互情報量、情報利得関数、グラフ・カット関数などが挙げられ、近年では、機械学習問題を劣モジュラ関数最大化問題として扱

う研究も見られるようになってきている [7, 11, 22]。

本稿では、次式で表わされるような、サイズ制約下での劣モジュラ関数 f の最大化を扱う。

$$\max_{S \subseteq V} f(S) \quad \text{s.t. } |S| \leq k \quad (1)$$

ここで $V = \{1, 2, \dots, n\}$ であり、 $k (\leq n)$ は正整数である。なお問題 (1) は、極めて一般的な表現であり、多くの問題をこの形式で表す事が可能である。またこの問題は、指数関数的に膨大な数の局所解が存在するため、計算が非常に困難である事が知られている。最適化の分野では、本問題は **NP 困難** であり、貪欲法により達成される近似率 $(1 - 1/e) (\approx 0.63)$ が、多項式時間アルゴリズムの限界である事が知られている (ただし、正非減少関数の場合) [18, 3]。最近では、この問題として表わされる複数の機械学習問題において、貪欲法が有用である事が報告されるようになってきている [7, 1, 11, 22]。しかし機械学習における応用によっては、この近似率よりも最適な解を求めたい場合もある。例えば特徴選択やセンサ配置問題においては、一度選択された特徴や配置は長期間にわたり利用されるため、選択に多くの時間を

*大阪大学産業科学研究所, 〒 567-0047 大阪府茨木市美穂ヶ丘 8-1, tel. 06-6879-8542, e-mail kawahara@ar.sanken.osaka-u.ac.jp

†東京工業大学数理・計算科学専攻, 〒 152-8552 東京都目黒区大岡山 2-12-1, e-mail nagano@is.titech.ac.jp

‡産業技術総合研究所 生命情報工学研究センター, 〒 135-0064 東京都江東区青海 2-42, e-mail koji.tsuda@aist.go.jp

§Dept. of Electrical Engineering, Washington Univ., 352500 Seattle, WA 98195 USA, e-mail bilmes@u.washington.edu

費やしたとしても、より良いものを選択する事が重要となる場合も多い。にもかかわらず、劣モジュラ最大化に関してはこれまで、厳密解に含め、上記の近似解より最適な解を見つける事を目的とした研究は極めて稀である [16, 14]。一般の非減少劣モジュラ関数に関するものは、Nemhauser&Wolsey により提案されたアルゴリズム [16] が唯一であると言える。しかしこの方法は、後述のように、一定のサイズより大きい問題に対しては計算が非効率となってしまう。

本稿では、カッピング・プレイン法に基づき、サイズ制約下での劣モジュラ最大化問題 (1) を解くための枠組みを導出し、具体的なアルゴリズムを提案する。提案するアルゴリズムは、反復的に線形 2 値計画問題 (BILP) を計算する事により実行される。この目的のため本稿では、関数の劣モジュラ性を用いて、反復計算の中で、現在の最良解よりも評価関数の値が改善されない実行領域を排除する、劣モジュラカットという切断面 (最適値に関する線形な下限) を導出する (従って本稿では、導出するアルゴリズムを、劣モジュラカット・アルゴリズムと呼ぶ)。この切断面による反復は、有限回での (厳密な) 最適値への収束を保証すると共に、効率的な探索を可能とする。更に本稿では、Nemhauser&Wolsey の方法 [16] を応用して、最適値に関する上限を導出する方法についても述べる。これにより、現在の最良値の最適性を評価する事が可能となり、事前に与えた $\epsilon > 0$ に対して、厳密解からの誤差が ϵ 以内である事が保証される近似解 (本稿では ϵ -最適解と呼ぶ) を計算する事が可能となる。提案するアルゴリズムでは、反復的に BILP (上限の計算には混合整数計画問題 (MIP)) を解く必要がある。これらの問題自体も NP 困難である事が知られているが、CPLEX などのソフトウェアを用いて効率的に計算する事が可能である。

本稿の構成は、以下のものである。まず第 2 章では、本稿で扱う劣モジュラ最大化の、機械学習における例を簡単に紹介する。そして第 3 章では劣モジュラカット・アルゴリズムを導出し、続く第 4 章において実装方法などの詳細について述べる。さらに第 5 章では、解の上限を求め、 ϵ -最適解を計算する方法について述べる。最後に第 6 章では、人工データ、及び、実データを用いた検証実験について述べる。¹

2 機械学習における例

上述のように、多くの機械学習問題は、劣モジュラ最大化問題 (1) として扱うことが可能である。ここでは、

提案する劣モジュラカット・アルゴリズムの動機を明らかにするため、その例として、特徴選択を中心として、いくつか簡単に紹介する。

特徴選択 自然言語処理や遺伝子解析、計算化学など、非常に高次元な変数を扱う必要がある応用では、特徴 (又は変数) 選択は極めて重要な課題として広く認識されている。機械学習では、この問題は下記のような正則化経験損失関数の最小化として扱われる事が多い [5, 19]。

$$C(\theta) = \frac{1}{N} \sum_{i=1}^N L(f_{\theta}(x_i), y_i) + \Omega(\theta)$$

ここで、関数 f は特徴ベクトル x から出力 y への写像であり、 L は損失関数、 Ω は正則化関数である。単純な例として、 f として線形モデル $f(x) = w^T x + b$ 、そして正則化として一般的な $\Omega(w) = \lambda \sum_{i=1}^n \alpha_i |w_i|^q$ を考える。 $q = 1$ の場合はいわゆる lasso である。この場合容易に分かるように、上記の拘束なし最小化は、次の拘束条件付き最適化問題と同様である。

$$\min_{\alpha} \sum_{i=1}^N \frac{1}{N} L(f(x_i), y_i) \quad \text{s.t.} \quad \sum_{i=1}^p \alpha_i |w_i|^q \leq \gamma$$

パラメータ λ と γ には一対一の関係がある。一方で、例えば自乗誤差 $L(f(x), y) = (y - f(x))^2$ の場合を考えると、この最小化は、劣モジュラ関数である重相関係数の 2 乗値 $(\text{Var}(y) - \frac{1}{N} (y - f(x))^2) / \text{Var}(y)$ の最大化と同値である事が知られている [1]。従って $q = 0$ の場合、この問題は劣モジュラ最大化 (1) となっている事がわかる。このように、いわゆる l_0 正則化の特徴選択は、劣モジュラ最大化 (1) として扱える場合がある。この他にも、しばしば自然言語処理の特徴選択に用いられる情報利得関数など、劣モジュラ性を有する評価関数を用いた l_0 正則化による特徴選択は、劣モジュラ最大化として考える事ができる。よく知られているように l_0 正則化の特徴選択は計算が非常に困難であり、本稿で提案するアルゴリズムは、その厳密解への一つの効率的なアプローチを提供する事が可能であると言える。

その他の例 その他にも、近年いくつかの応用的問題が劣モジュラ最大化として扱われている。例えば、Krauseらによるグループでは、観測の分散をできるだけ小さくするようなセンサ配置を決める問題を、劣モジュラ最大化問題として貪欲法に基づくアプローチで取り組んでいる [9, 11]。この配置に関する評価関数として用いられている分散低減関数²は、上述の劣モジュラ性を満たしているため、貪欲法による解が近似的に良い性質を持つ事が保証される。また、情報の起点になるブログ記事の発見 [13]、オフラインの能動学習 [7]、またロボットのパス

¹本稿は、文献 [10] の日本語による概略版となっている。本稿中の定理等の証明については文献 [10] を参照のこと。

²第 6.3 節を参照。

表 1: Lovász 拡張による‘連続’と‘離散’との関係

(discrete)		(continuous)
$f : 2^V \rightarrow \mathbb{R}$	$\xLeftrightarrow{\text{Eq.(2)}}$	$\hat{f} : \mathbb{R}^n \rightarrow \mathbb{R}$
$S \subseteq V$	$\xLeftrightarrow{\text{Eq.(3)}}$	$\mathbf{I}_S \in \mathbb{R}^n$
f is submodular	$\xLeftrightarrow{\text{Thm.1}}$	\hat{f} is convex

計画 [20] といった問題が, 劣モジュラ最大化として扱われている.

3 劣モジュラカット

カッティング・プレーン法は, これまでも多くの数理解計画問題において, 極めて成功裏に用いられてきた枠組みである [21, 17, 8]. その基本的な考え方は, 反復的に, 最適値に関する下限 (線形の拘束条件) を導く事により, 現在の最良解よりも評価関数の値が改善されない実行領域を排除するというものである. 本章では, カッティング・プレーン法に基づき劣モジュラ最大化問題 (1) を解く, 劣モジュラカット・アルゴリズムの導出を行う.

3.1 Lovász 拡張による連続化

本稿で提案する方法導出の基本的な考え方は, Lovász 拡張と呼ばれる集合関数の連続化に基づいて, 離散最適化問題である (1) を, 連続領域における凸関数最大化との類推から解く事にある. 従ってまず本章では, Lovász 拡張について概説する. なお, Lovász 拡張によって得られる, ‘連続’ と ‘離散’ との関係は表 1 のようになる.

まず任意の実ベクトル $\mathbf{p} \in \mathbb{R}^n$ が与えられたとき, \mathbf{p} の m 個の異なる要素の値を $\hat{p}_1 > \hat{p}_2 > \dots > \hat{p}_m$ と表す. このとき, 一般の集合関数 $f : 2^V \mapsto \mathbb{R}$ に関する Lovász 拡張 $\hat{f} : \mathbb{R}^n \mapsto \mathbb{R}$ は, 次のように定義される.³

$$\hat{f}(\mathbf{p}) = \sum_{k=1}^{m-1} (\hat{p}_k - \hat{p}_{k+1}) f(U_k) + \hat{p}_m f(U_m), \quad (2)$$

$U_k = \{i \in V : p_i \geq \hat{p}_k\}$ である. 定義より, \hat{f} は区分的線形関数である. 一般に \hat{f} は凸ではないが, f の劣モジュラ性と \hat{f} の凸性は, 次のように与えられる [12, 15].

定理 1 集合関数 $f : 2^V \mapsto \mathbb{R}$ 及びその Lovász 拡張に対して, f が劣モジュラの場合, \hat{f} は凸となる.

ここで, \mathbf{e}_i を第 i 要素に関する単位ベクトルとして, $\mathbf{I}_S := \sum_{i \in S} \mathbf{e}_i \in \{0, 1\}^n$ となるベクトルを定義する (本稿では, 特性ベクトルと呼ぶ).⁴ 明らかに, \mathbf{I}_S と S

³Lovász 拡張の定義においては, 必ずしも劣モジュラ性を満たす必要はない.

⁴例えば $|V| = 6$ の場合, $S = \{1, 3, 4\}$ の特性ベクトルは $\mathbf{I}_S = (1, 0, 1, 1, 0, 0)$ となる.

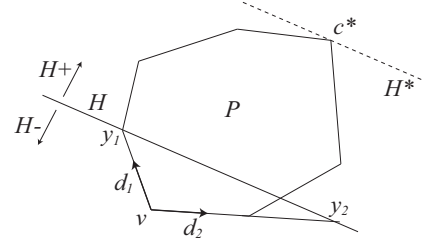


図 1: 切断平面 H の概念図. 記号は第 4.2 節を参照.

は一対一に対応する. このとき次式の意味で, Lovász 拡張 \hat{f} は f の自然な拡張となっている [12, 15].

$$\hat{f}(\mathbf{I}_S) = f(S) \quad (S \subseteq V) \quad (3)$$

このような集合関数の連続化の方法を用いる事で, 劣モジュラ最大化 (1) の評価関数 $f : 2^V \rightarrow \mathbb{R}$ と実行可能領域 $\{S \subseteq V : |S| \leq k\}$ を, Lovász 拡張 $\hat{f} : \mathbb{R}^n \rightarrow \mathbb{R}$ と次式で定義される多面体 D_0 とにそれぞれ置き換える事で, 連続問題への緩和を考える事ができる.

$$D_0 = \{\mathbf{x} \in \mathbb{R}^n : 0 \leq x_i \leq 1 (i = 1, \dots, n), \sum_{i=1}^n x_i \leq k\},$$

結果として得られる問題は凸関数最大化である. そのように, 以降では, 劣モジュラ最大化 (1) の計算を, 連続問題 $\max \{\hat{f}(\mathbf{x}) : \mathbf{x} \in D_0\}$ との関係から解く方法について述べる. なお, 以降に出てくる多面体 P は, 上記 D_0 と同様, $P = \{\mathbf{x} \in \mathbb{R}^n : A_j^T \mathbf{x} \leq b_j, j = 1, \dots, m\}$ (A_j は実ベクトル, b_j は実スカラーである) と表わされる有限の半空間の交わりとして定義する事ができる.

3.2 劣モジュラカット

上述のように, 本稿で提案するアルゴリズムの基本は, 劣モジュラ最大化問題 (1) を, 上述の Lovász 拡張を用いて, 多面体上における凸関数最大化との類推により解く事にある. そこでまず, これらの問題の根本となる, 次の事実を確認する必要がある [8].

定理 2 任意の凸関数 $g : \mathbb{R}^n \rightarrow \mathbb{R}$ は, 多面体 $P \subset \mathbb{R}^n$ 上においては, P の頂点において大域的に最大となる.

本章では, ある多面体 $P \subseteq D_0$ が与えられた時に, 効率的に, 解を改善しない実行可能領域の候補を削除するような下限 γ (切断面) を導出する事である. そのためまず, 劣モジュラ最大化と凸最大化との関係を明確にする. 特性ベクトルが P 上又は内部となる V の部分集合, つまり任意の $S' \in S(P)$ に関して $\mathbf{I}_{S'} \in P$ となる, V の部分集合を $S(P)$ と表す. また, P の全頂点からなる集合を $V(P)$ で表す. なお, 任意の特性ベクトル $\mathbf{I}_S \in P$ は P の頂点であり, また $S(D_0)$ と $V(D_0)$ は一対一に対

応する．このとき，明らかに次式が成り立つ．

$$\max\{f(S') : S' \in S(P)\} \leq \max\{\hat{f}(x) : x \in P\}. \quad (4)$$

従って，もし現在の最良値より常に \hat{f} の関数値が小さくなるような部分集合 \bar{P} が見つければ，任意の $\bar{S} \in S(\bar{P})$ に関して， $f(\bar{S})$ の値もその最良値より小さくなる．そのように，問題 $\max\{\hat{f}(x) : x \in P\}$ の切断面（線形な下限）は，上記の関係 (4) を通して，劣モジュラ最大化問題 (1) にも適用できる．

そのように以下では，劣モジュラ性から Lovász 拡張を介して，切断面（本稿では劣モジュラカットと呼ぶ）を導出する．なお上述のように，劣モジュラカットによる最大化は反復的な計算により実行されるが，各反復においては，現在の最良の関数値 γ ，多面体 $P \subseteq D_0$ ，そして $f(S^*) = \gamma$ を満たす集合 $S^* \subseteq V$ が保持される．まず，次のような定義を与える．

定義 3 (γ 拡張) $g : \mathbb{R}^n \rightarrow \mathbb{R}$ を凸関数， $x \in \mathbb{R}^n$ ， γ を実数，そして $t > 0$ とする．このとき，次式により与えられる $y \in \mathbb{R}^n$ を，方向 $d \in \mathbb{R}^n \setminus \{0\}$ における x の (g に関する) γ 拡張と呼ぶ（ただし $\theta \in \mathbb{R} \cup \{\infty\}$ ）．

$$y = x + \theta d \quad \text{with} \quad \theta = \sup\{t : g(x + td) \leq \gamma\} \quad (5)$$

劣モジュラ関数の Lovász 拡張は凸関数であるため，常に任意の点 $x \in \mathbb{R}^n$ において γ -拡張が定義可能である．

$v \in V(P)$ を，何らかの $S \in S(P)$ に対して $v = I_S$ となる P の頂点とし，さらに線形独立な方向 d_1, \dots, d_n により生成される v を頂点とする凸多面錐を $K = K(v; d_1, \dots, d_n)$ とする（つまり $K = \{v + t_1 d_1 + \dots + t_n d_n : t_l \geq 0\}$ ）．そして，各 $l = 1, \dots, n$ に対して， \hat{f} に関して，方向 d_l への v の γ -拡張を

$$y_l = v + \theta_l d_l$$

とする．方向 d_1, \dots, d_n は $P \subset K$ となるように選ばれ，また $\theta_l > 0$ である（第 4.1 節参照）．このとき，ベクトル d_l は線形独立に選ばれるため， y_l ($l = 1, \dots, n$) を含む超平面 $H = H(y_1, \dots, y_n)$ がただ一通りに定まる．本稿では，この超平面を劣モジュラカットと呼び，それは次式のように定義される（図 1 参照）．

$$H = \{x : e^T Y^{-1} x = 1 + e^T Y^{-1} v\}. \quad (6)$$

ここで $e = (1, \dots, 1)^T \in \mathbb{R}^n$ であり，また $Y = ((y_1 - v, \dots, (y_n - v))$ である．この超平面 H は， $H_- = \{x : e^T Y^{-1} x \leq 1 + e^T Y v\}$ 及び $H_+ = \{x : e^T Y^{-1} x \geq 1 + e^T Y v\}$ の 2 つの半空間を生成する．明らかに v は半空間 H_- の内部であり，また次の事実を導かれる．

```

1 Compute a subset  $S_0$  s.t.  $|S_0| \leq k$ , and set a lower
  bound  $\gamma_0 = f(S_0)$ .
2 Set  $P_0 \leftarrow D_0$ ,  $stop \leftarrow false$ ,  $i \leftarrow 1$  and  $S^* = S_0$ .
3 while  $stop = false$  do
4   Construct with respect to  $S_{i-1}$ ,  $P_{i-1}$  and  $\gamma_{i-1}$  a
     submodularity cut  $H^i$ .
5   if  $S(P_{i-1}) = S(P_{i-1} \cap H^i)$  then
6      $stop \leftarrow true$  ( $S^*$  is an optimal solution and
        $\gamma_{i-1}$  the optimal value).
7   else
8     Update  $\gamma_i$  (using  $S_i$  and other available
       information) and set  $S^*$  s.t.  $f(S^*) = \gamma_i$ .
9     Compute  $S_i \in S(P_i)$ , and set
        $P_i \leftarrow P_{i-1} \cap H_+^i$  and  $i \leftarrow i + 1$ .

```

アルゴリズム 1: 劣モジュラカット（一般）

補題 4 $P \subseteq D_0$ を多面体， γ を現在の最適値， v を部分集合 $S \in S(P)$ に対して $v = I_S$ となる P 上の頂点，そして H_- を切除平面で決まる半空間（つまり式 (6)）とする．このとき，次式が成り立つ．

$$f(S') \leq \gamma \quad \text{for all} \quad S' \in S(P \cap H_-).$$

この補題は，実行可能領域 $S(P)$ から， γ より評価関数を改善する可能性のない領域 $S(P \cap H_-)$ を切除可能である事を意味する．また，もし $S(P) = S(P \cap H_-)$ となれば， $\gamma = \max\{f(S) : |S| \leq k\}$ が達成される．収束条件 $S(P) = S(P \cap H_-)$ を調べる具体的な方法については，第 4.2 節において述べる．さらに $v \in S(P \cap H_-)$ 及び $v \notin S(P \cap H_+)$ に関して，次式が成り立つ．

$$|S(P)| > |S(P \cap H_+)|. \quad (7)$$

そのように， γ の大域的最適性が保障されるまで，劣モジュラカット・アルゴリズムは $P \leftarrow P \cap H_+$ として，更新を繰り返す．一般的な手順については，アルゴリズム 1 のようになる（図 2 も参照）．更に，アルゴリズムの有限性については，次のように保障される．

定理 5 アルゴリズム 1 は，有限回の反復で，劣モジュラ最大化問題 (1) の最適値を与える．

4 実装について

本節では，アルゴリズム 1 を BILP を用いて実装するために必要な具体的方法について述べる．結果として得られる疑似コードは，アルゴリズム 2 のようになる。⁵

⁵ 著者のホームページ <http://www.ar.sanken.osaka-u.ac.jp/~kawahara/software.html> 上にて Matlab による実装を公開．

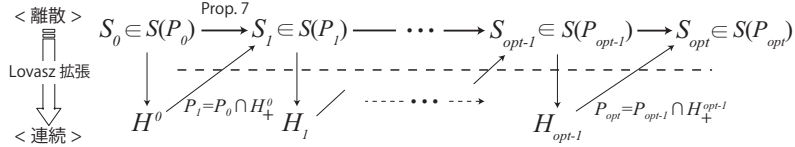


図 2: 劣モジュラカット・アルゴリズムの概念図 .

4.1 劣モジュラカットの構成方法

ここでは、多面体 $P \subseteq D_0$ の頂点が与えられたとき、アルゴリズムの各反復において劣モジュラカットを構成するために必要な、線形独立な方向 d_1, \dots, d_n を計算する方法について述べる（アルゴリズム 1 の第 4 列目）.⁶

まず $|S| < k$ の場合、 d_1, \dots, d_n は $-e_l$ ($l \in S$) 及び e_l ($l \in V \setminus S$) のように選択できる．そして $|S| = k$ の場合は、次のようにして行う．まず、 S の隣接する集合を次のように定義する．

$$S_{(i,j)} := (S \setminus \{i\}) \cup \{j\} \quad (i \in S, j \in V \setminus S)$$

つまり $S_{(i,j)}$ は、 S の要素の一つを、 $V \setminus S$ のそれと置き換えたものであり、 $I_{S_{(i,j)}} - I_S = e_j - e_i$ となる．今、 S の全ての隣接集合の中で、関数値 $f(S_{(i,j)})$ を最大にするものを $S_{(i^*,j^*)}$ とする．サイズ k の集合 S は、 $k \times (n-k)$ の隣接する集合 $S_{(i,j)}$ ($i \in S, j \in V \setminus S$) を持つので、この計算は $O(nk)$ である．そして、もし $f(S_{(i^*,j^*)}) > \gamma$ の場合は、 $\gamma \leftarrow f(S_{(i^*,j^*)})$ 及び $S^* \leftarrow S_{(i^*,j^*)}$ のように更新するとする．このとき、いかなる場合も $\gamma \geq f(S_{(i^*,j^*)})$ が成り立ち、方向 $\{d_1, \dots, d_n\}$ として次のように選択する事ができる．

$$d_l = \begin{cases} e_{j^*} - e_{i^*} & \text{if } l \in \{1, \dots, k\} \\ e_{j^*} - e_{i^*} & \text{if } l \in \{k+1, \dots, n-1\} \\ -e_{j^*} & \text{if } l = n. \end{cases} \quad (8)$$

このように選択された d_1, \dots, d_n が線形独立となる事は、容易に確かめられる．更に、次の補題が得られる．

補題 6 式 (8) で定義される方向 d_1, \dots, d_n に関して、次式で定義される錐 K は、多面体 $D_0 = \{x \in \mathbb{R}^n : 0 \leq x_l \leq 1 \ (l = 1, \dots, n), \sum_{l=1}^n x_l \leq k\}$ をその内部に含む．

$$K(I_S; d_1, \dots, d_n) = \{I_S + t_1 d_1 + \dots + t_n d_n : t_l \geq 0\}$$

なお、これらの方向に関する γ -拡張は、閉じた形で計算する事ができる（詳細は文献 [10] を参照）．

4.2 停止基準と各反復における開始点

次にここでは、解の最適性の判定、つまり $S(P) = S(P \cap H_-)$ かどうかを評価し、また同時に、次回反復の

⁶本節で述べる方法はあくまで一つの選択肢であり、 $P \subset K$ を満たすその他の方法でも実行可能である．

```

1 Compute a subset  $S_0$  s.t.  $|S_0| \leq k$ , and set a lower
  bound  $\gamma_0 = f(S_0)$ .
2 Set  $P_0 \leftarrow D_0$ ,  $stop \leftarrow false$ ,  $i \leftarrow 1$  and  $S^* = S_0$ .
3 while  $stop = false$  do
4   Construct with respect to  $S_{i-1}$ ,  $P_{i-1}$  and  $\gamma_{i-1}$  a
    submodularity cut  $H^i$ .
5   Solve the BILP problem (9) with respect to  $A_j$ 
    and  $b_j$  ( $j = 1, \dots, n_k$ ), and let the optimal
    solution and value  $S_i$  and  $c^*$ , respectively.
6   if  $c^* \leq 1 + e^T Y^{-1} v_{i-1}$  then
7      $stop \leftarrow true$  ( $S^*$  is an optimal solution and
       $\gamma_{i-1}$  the optimal value).
8   else
9     Update  $\gamma_i$  (using  $S_i$  and other available
      information) and set  $S^*$  s.t.  $f(S^*) = \gamma_i$ .
10    Set  $P_i \leftarrow P_{i-1} \cap H_+$  and  $i \leftarrow i + 1$ .

```

アルゴリズム 2: BILP による劣モジュラカット

開始集合 S_i （各々アルゴリズム 1 の第 5 列及び第 9 列）を計算する方法について述べる．

$S(P) = S(P \cap H_-)$ の判定は、幾何学的には、 $S(P)$ を含む最小の多面体を $\tilde{P} \subseteq \mathbb{R}^n$ とすると、 \tilde{P} に接する H に平行な超平面 H^* を考える事で可能となる．つまり、 $H = H^*$ 、あるいは、 H^* が H を v の方向に動かして得られる場合、 $S(P) = S(P \cap H_-)$ が満たされる．数値的には、移動は LP を解く事と同じである．従って、式 (6) を用いて次の補題が得られる．

命題 7 次の l 値整数計画問題の最適値を c^* とする．

$$\max_{x \in \{0,1\}^n} \{e^T Y^{-1} x : A_j x \geq b_j, j = 1, \dots, m_k\}. \quad (9)$$

このとき、 $c^* \leq 1 + e^T Y^{-1} v$ なら $S(P) \subset H_-$ となる．

なお、もし $c^* > 1 + e^T Y^{-1} v$ ならば、式 (9) の最適解 x^* は $S(P \setminus H_-)$ の部分集合となっており、次反復開始の部分集合として利用できる（図 1 参照）．

5 上限と ϵ -最適解

提案するアルゴリズムは、有限回の反復で厳密な最適解を見つける事ができるものの、高次元の対象に対しては、その計算時間が膨大になってしまう可能性もある．そのためここでは、現在の解の上限を計算する方法についても述べる．これにより、上記の劣モジュラカットと組み合わせる事で、 ϵ -最適解の計算ができる．

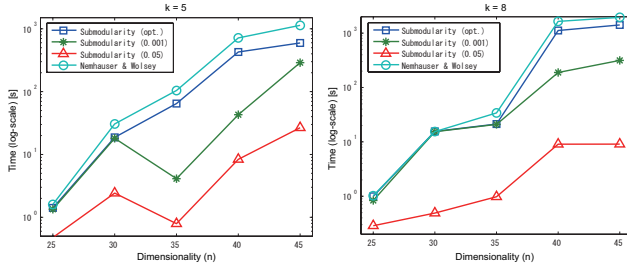


図 3: 劣モジュラカット・アルゴリズム, 及び, Nemhauser & Wolsey の方法 [16] による, 厳密解, 及び, ϵ -最適解の計算の平均所要時間 (log-scale) .

このような上限の導出は, Nemhauser&Wolsey の考え方 [16] に基づいて可能となる. 劣モジュラ関数 $f: 2^V \rightarrow \mathbb{R}$ が非減少である場合, 劣モジュラ最大化 (1) は, 次のようにも表す事が可能である事が知られている [16] .

$$\max \eta \quad \text{s.t.} \quad \eta \leq f(S) + \sum_{j \in V \setminus S} \rho_j(S) y_j \quad (S \subseteq V), \\ \sum_{j \in V} y_j = k, \quad y_j \in \{0, 1\} \quad (j \in V) \quad (10)$$

ただし $\rho_j(S) := f(S \cup \{j\}) - f(S)$ である. この問題は, 約 2^n 個の拘束を持つ, 1 個の連続変数と n 個の 2 値変数に関する MIP となっている.

拘束条件が膨大であるため上記問題を直接解く事は不可能であるが, Nemhauser&Wolsey は, 一つづつ拘束条件を順に加え, 小さいサイズの MIP を反復的に計算する. この方法は最悪の場合, 反復回数が全拘束条件数になってしまう. しかし容易に分かるように, 全拘束条件を考慮した場合の最大値よりも, 拘束条件の一部のみを考慮した場合の方が解は大きくなるため, 各反復によって, 単調に真の解へと収束していく事が保障される. 一方で本稿で提案した方法は, 各反復における解は単調に真の解へと増加していく. 従って, 劣モジュラカット・アルゴリズムの各反復による解 S_i に関して, 拘束を上記の MIP 加えて解く事で, 現在の解に関する上限が得られる. そのようにして, 得られた上限と下限を用いて, 現在の解の最適性の評価ができるようになり, 事前に設定した ϵ に対する ϵ -最適解を得る事が可能となる.

6 実験結果

本章では, 提案した劣モジュラカット・アルゴリズムの検証例を示す. まず第 6.1 節では, 人工データを用いて, 従来法 [16] の方法との比較を行う. そして第 6.3 節では, センサ配置に関する実データに対して, 提案したアルゴリズムを適用した結果を示す. なお以下の実験結果は, Matlab, 及び, Parallel CPLEX ver. 11.2 (8 スレッド) を用いて, 2.5GHz 64-bit のワークステーショ

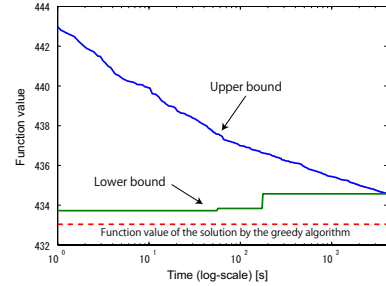


図 4: 劣モジュラカット・アルゴリズムにより計算された上限, 及び, 下限の計算時間に対する変化の例 .

ン上で実行したものである. また $\theta = \infty$ (式 (5)) の場合には, $\theta = 10^6$ として計算している.

6.1 人工データによる実験

まず本節では, 劣モジュラカット・アルゴリズムを, (1) 厳密解に関する従来法 [16] との計算時間の比較, そして (2) どのように上限と下限が真の解に近づくか, という点を中心に, 人工データを用いて数値的に検証する.

ここでは, K -配置問題 [16] と呼ばれる, 次の非減少劣モジュラ関数に関する最大化問題 (1) について考える.

$$f(S) = \sum_{i=1}^m \max_{j \in S} c_{ij},$$

ただし $C = c_{ij}$ は $m \times n$ の非負行列, また $V = \{1, \dots, n\}$ である. 実験では, 異なる n (ただし m は $m = n + 1$ と固定) に関して複数の行列 C を生成し, $k = 5, 8$ の場合に関して, 上記問題に対して劣モジュラカット・アルゴリズム 2, 及び, 従来手法 [16] を適用した. 劣モジュラカット・アルゴリズムに関しては, 厳密解に加え, ϵ -最適解に関する結果も示している.

まず図 3 は, 複数の n , 及び, $k = 5, 8$ に関してアルゴリズムを適用した場合の, ランダムに生成された 3 つの C に関して平均化した計算時間である. なお, 例えば, $n = 45$, 及び, $k = 8$ の場合には, 組み合わせの数は二百万以上と膨大な数になる. 図からわかるように, 既存手法に比べ, アルゴリズム 2 に必要な計算コストが少なくなっている事がわかる. その傾向は, 特に高い次元の場合に顕著であるように見える. これは, 劣モジュラカットを用いて, カッティング・プレーン法が解を改善しない実行可能領域を効率的に除外しているためであると考えられる. また図 4 は, 劣モジュラカット・アルゴリズムにより計算された上限, 及び, 下限の計算時間に対する変化の例である ($k = 5, n = 45$). 上限は徐々に減少していく一方, 下限は更新が稀であるが, 早い段階で最適解へと収束している事がわかる.

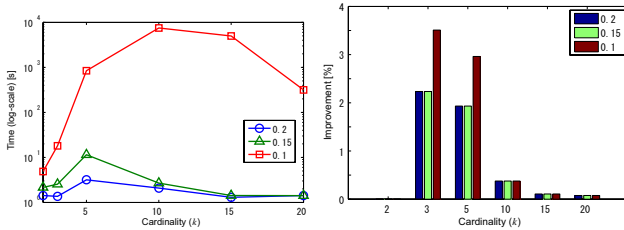


図 5: 計算時間 (左) と、貪欲法を基準にした解の最適性の向上度合 (%) (右)。

6.2 センサー配置 (実データ)

実データを用いた例として、ここでは、センサ配置問題へアルゴリズムを適用した結果を示す。センサ配置問題では、観測の分散をできるだけ小さくするようにセンサの配置を決める事を目的とする。用いたデータは、カリフォルニア大学近辺の湖で用いられた NIMS センサ・ノードを用いて得られた、有限の離散的な場所 V に関する温度計測である [6, 9] ($|V| = 86$)。

文献 [9] 同様、配置場所の集合 $S \subseteq V$ は、次式のように表わされる平均化分散低減を用いて評価した。

$$f(S) = \text{Var}(\emptyset) - \text{Var}(S) = \frac{1}{n} \sum_s F_s(S)$$

ここで、 $F_s(S) = \sigma_s^2 - \sigma_{s|S}^2$ は分散低減、そして $\sigma_{s|S}^2$ は $S \subseteq V$ における観測後の場所 $s \in V$ における観測の予測分散である。この関数は、単調かつ劣モジュラである。図 5 は、 $\epsilon = 0.05, 0.1, 0.2$ の場合の、劣モジュラカット・アルゴリズムの計算に要した時間、及び、得られた解の貪欲法の解からの改善度合 (%) を表したものである。いずれの場合も、 $k = 5$ 付近で、計算時間と改善度合共に大きくなっている事がわかる。これは、 k が非常に大きい、あるいは小さい場合には、貪欲法の解が真の最適解に近くなっているためであると言える。

6.3 文書分類における特徴選択 (実データ)

もう一つの実データを用いた例として、Reuters-21578 を用いて、文書分類における特徴選択の適用例を示す。

本実験では、貪欲法と提案手法を、5,180 単語 (特徴) から成る 90 カテゴリーの 7,770 文書を含む訓練データに対して適用した。特徴選択の基準としては、ここでは情報利得を用いた。表 2 は、 $k = 5, 10$ の場合に、アルゴリズムにより選択された単語を示している (提案手法に関しては $\epsilon = 0.003$ としている)。表中には、情報利得と分類精度 ($tp/(tp + fp)$), tp ; true positive, fp ; false positive) についても記載している。評価データ (5,180 単語から成る 90 カテゴリーの 3,019 文書) における分類では、選択された単語を用いて、単純ベイズ識別器を用い

た。劣モジュラカットによって選択された単語は、貪欲法のそれとは異なっている事がわかる。

7 結論

本稿では、多くの機械学習問題が含まれる劣モジュラ最大化の厳密解を求めるための方法として、カッティング・プレーン法に基づき、劣モジュラカット・アルゴリズムを導出した。また同時に、解の上限を求める方法を導出し、事前に設定した $\epsilon > 0$ に対して、 ϵ -最適解を計算する方法についても述べた。そして、提案したアルゴリズムの人工データや実データを用いた検証例を示した。なお本稿は、文献 [10] の概略版となっている。

参考文献

- [1] A. Das and D. Kempe, "Algorithms for subset selection in linear regression," In R. E. Ladner and C. Dwork, editors, *Proc. of the 40th Ann. ACM Symp. on Theory of Computing (STOC'08)*, pages 45–54, 2008.
- [2] J. Edmonds, "Submodular functions, matroids, and certain polyhedra," In R. Guy, H. Hanani, N. Sauer and J. Schönheim, editors, *Combinatorial Structures and Their Applications*, pages 69–87, Gordon and Breach, 1970.
- [3] U. Feige, "A threshold of $\ln n$ for approximating set cover," *Journal of the ACM*, **45**:634–652, 1998.
- [4] S. Fujishige, *Submodular Functions and Optimization*, Elsevier, second edition, 2005.
- [5] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, **3**:1157–1182, 2003.
- [6] T. C. Harmon, R. F. Ambrose, R. M. Gilbert, J. C. Fisher, M. Stealey and W. J. Kaiser, "High resolution river hydraulic and water quality characterization using rapidly deployable," Technical report, CENS, 2006.
- [7] S. C. H. Hoi, R. Jin, J. Zhu and M. R. Lyu, "Batch mode active learning and its application to medical image classification," In *Proc. of the 23rd Int'l Conf. on Machine Learning (ICML'06)*, pages 417–424, 2006.
- [8] R. Horst and H. Tuy, *Global Optimization (Deterministic Approaches)* (3rd eds.), Springer, 1996.
- [9] A. Krause, H. B. McMahan, C. Guestrin and A. Gupta, "Robust submodular observation selection," *Journal of Machine Learning Research*, **9**:2761–2801, 2008.
- [10] Y. Kawahara, K. Nagano, K. Tsuda and J. Bilmes, "Submodularity cuts and applications," In *Advanced in Neural Information Processing Systems*, Vol. 22, MIT Press, Cambridge, 2009 (to appear).
- [11] A. Krause, A. Singh and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory efficient algorithms and empirical studies," *Journal of Machine Learning Research*, **9**:235–284, 2009.
- [12] L. Lovász, "Submodular functions and convexity," In A. Bachem, M. Grötschel and B. Korte, editors, *Mathematical Programming – The State of the Art*, pages 235–257, 1983.
- [13] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen and N. Glance, "Cost-effective Outbreak Detection in Networks," In *Proc. of the 13th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining (KDD'07)*, pp.420–429, 2007.

表 2: 選択された単語 [情報利得の値, 分類精度].

k	greedy	submodularity cuts
5	(tonn, 'agricultur', trade, pct, 'market')[2.59, 0.53]	→ ('week', tonn, trade, pct, 'washington')[2.66, 0.58]
10	(..., week, oil, price, 'dollar', 'offici')[3.55, 0.57]	→ (... , price, oil, 'bank', 'produc', 'blah')[3.8, 0.62]

- [14] H. Lee, G. L. Nemhauser and Y. Wang, "Maximizing a submodular function by integer programming: Polyhedral results for the quadratic case," *European Journal of Operational Research*, **94**:235–284, 2009.
- [15] K. Murota, *Discrete Convex Analysis*, volume 10 of *Mono-graphs on Discrete Math and Applications*. Society for Industrial & Applied Mathematics, 2000.
- [16] G. L. Nemhauser and L. A. Wolsey, "Maximizing submodular set functions: formulations and analysis of algorithms," In P. Hansen, editor, *Studies on Graphs and Discrete Programming*, Vol.11 of *Annals of Discrete Mathematics*, 1981.
- [17] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*, Wiley-Interscience, 1988.
- [18] G. L. Nemhauser, L. A. Wolsey and M. L. Fisher, "An analysis of approximations for maximizing for submodular set functions-I," *Mathematical Programming*, **14**:265–294, 1978.
- [19] S. Perkins, K. Lacker and J. Theiler, "Grafting: Fast, incremental feature selection by gradient descent in function space," *Journal of Machine Learning Research*, **3**:1333–1356, 2003.
- [20] A. Singh, A. Krause, C. Guestrin, W. Kaiser and M. Batalin, "Efficient planning of informative paths for multiple robots," In *Proc. of the 21st Int'l Joint Conf. on Artificial Intelligence*, pp.2204–2211, 2007.
- [21] H. Tuy, "Concave programming under linear constraints," *Soviet Mathematics Doklady*, **5**:1437–1440, 1963.
- [22] M. Thoma, H. Cheng, A. Gretton, J. Han, H. Kriegel, A. Smola, L. Song, P. Yu, X. Yan and K. Borgwardt, "Near-optimal supervised feature selection among frequent subgraphs," In *Proc. of the 2009 SIAM Conf. on Data Mining (SDM'09)*, pages 1075–1086, 2009.