人工知能学会研究会資料
SIG-FPAI-B508-01

機械学習モデルの列挙
Enumerating Machine Learning Models

原 聡 [1*]　石畠 正和 [2]　前原 貴憲 [3]
Satoshi Hara[1]　Masakazu Ishihata[2]　Takanori Maehara[3]

[1] 大阪大学　[1] Osaka University
[2] 北海道大学　[2] Hokkaido University
[3] 理研 AIP　[3] RIKEN Center for Advanced Intelligence Project

Abstract: In ordinary machine learning, a single model is obtained by solving an optimization problem, and the resulting model is interpreted as the one that best explains the data. In this study, instead of finding a single machine learning model, we propose an algorithm for enumerating multiple models. Model enumeration is useful in practice when (i) users want to choose a model that is particularly suited to their task knowledge, or (ii) users want to obtain several possible mechanisms that could be underlying the data to use as hypotheses for further scientific studies. To this end, we propose an enumeration algorithm for enumerating linear models and rule models. We also show that, by using the proposed enumeration algorithm, we can find several different models of almost equal quality.

1 はじめに

研究背景　機械学習は我々の身の回りを取り巻く技術となりつつある。機械学習技術の活用範囲は現在でも拡大を続けており、病気診断 [1] や裁判 [2]、教育 [3] などへの活用の検討も進んでいる。これら応用分野の一部では機械学習により単に精度の高い予測ができるだけでなく、予測根拠の提示、つまり機械学習モデルのユーザにとっての解釈性が求められている。例えば、以下のようなケースにおいてモデルの解釈性が重要視されている [4, 5]。

- 人間の意思決定を支援する場合。このような場合にブラックボックスなモデルを使うと、ユーザはそのモデルの予測根拠がわからないためモデルを信頼して良いかどうかわからなくなってしまう。
- データの背後の仕組みに興味がありデータマイニングが目的の場合。このような場合、ユーザはデータに関する洞察を深める目的で機械学習モデルを使う。モデルを通じてデータを理解するためには解釈性の高いモデルが必須である。

本稿では解釈性を重要視する文脈において有用性が期待される『機械学習モデルの列挙』の方法について述べる [6, 7]。通常の機械学習では経験誤差最小化などの学習問題の最適化を通じてモデルの学習を行う。そのため、最適解以外のモデルは全て破棄されてしまう。これに対し『機械学習モデルの列挙』では最適解以外のモデルをも考慮することができる。このようなモデル列挙には以下のような実用上で重要な利点がある。

- モデルの信頼性向上。時に機械学習における“最適モデル”はユーザの直感に反することがある。このような場合に、モデル列挙は複数のモデル候補を提示することで、ユーザにとってより納得感・信頼性の高いモデルを提供できる。
- データ理解の促進。多くのデータ分析業務においてデータの同等な説明が可能な複数のモデルが存在することは珍しくない。モデル列挙を通じてこのような複数のモデル候補を比較検討することで、単一のモデルだけを考慮する場合よりもデータについての深い洞察が可能となる。

本研究の貢献 [6, 7]

1. 線形モデル及びルールモデルの列挙問題を“異なる台を持つ解を目的関数の降順に列挙する問題”として定式化した。
2. 定義した列挙問題を解くためのアルゴリズムを提案した。提案法により線形モデルの厳密な列挙ができる。また、ルールモデルの近似的な列挙ができる。
3. 実データ実験により、提案法を使うことで適切にモデルが列挙できることを確認した。また、提案

*連絡先：大阪大学産業科学研究所
〒 567-0047 大阪府茨木市美穂ケ丘 8 － 1
E-mail: satohara@ar.sanken.osaka-u.ac.jp

法により同等の質のモデルが実データにおいて多数存在することを確認した。

表記　自然数 $L \in \mathbb{N}$ について、集合 $\{1, 2, \ldots, L\}$ を $[L]$ と表記する。集合 T のべき集合を 2^T と表記する。学習問題 $\max_{m \in \mathcal{F}} f(m)$ について、その最適モデルを $m^* = \mathrm{argmax}_{m \in \mathcal{F}} f(m)$ と表記する。ただし、$\mathcal{F}$ は適切なモデルの集合とする。また、モデル $m \in \mathcal{F}$ が $f(m) \geq \alpha f(m^*)$ を満たすとき、モデル m を α 近似解と呼ぶ。ここで $\alpha \in [0, 1]$ は近似比である。

2　線形モデルとルールモデル

本節では、本研究で列挙対象とする線形モデル及びルールモデルについて述べる。

2.1　線形モデル

本研究では、線形モデルの中でもその回帰係数が疎なモデルを考える。このような疎な線形モデルは特徴選択としてデータから予測に効く重要な特徴量を自動的に絞り込む目的で使われることが多い。本研究では疎な線形モデルの学習法の代表例である Lasso [8] を扱う。ここではデータの入力次元を d とし、N 個の訓練データから学習する場合を考える。このとき、$\boldsymbol{y} \in \mathbb{R}^N$ を出力データのベクトル、$X \in \mathbb{R}^{N \times d}$ を入力データの行列とする。Lasso では X から $\boldsymbol{y}$ を予測する回帰係数 $\boldsymbol{\beta} \in \mathbb{R}^d$ を以下の最適化問題を解くことで学習する。

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^d} \frac{1}{2} \|X\boldsymbol{\beta} - \boldsymbol{y}\|^2 + \rho \|\boldsymbol{\beta}\|_1. \tag{1}$$

ここで ρ は非負の正則化パラメータ、$\|\boldsymbol{\beta}\|_1 = \sum_{i=1}^{d} |\beta_i|$ は $\boldsymbol{\beta}$ の ℓ_1 ノルムである。Lasso では ℓ_1 ノルムをペナルティ項として用いることで、少数の非零要素だけを含む疎な $\boldsymbol{\beta}$ を推定できる [8]。

2.2　ルールモデル

機械学習におけるルールモデルの代表例は決定木 [9]、ルールリスト [10]、そしてルールセット [11] である。決定木ついては近年 Ruggieri により列挙法が提案されている [12]。そこで、本研究ではルールリストとルールセットの列挙について扱う。本稿ではこれら二つの中でも特に効率的な近似学習が可能なルールセットについて述べる。

ルールセットは図 1 のように “if-or-then” により記述されるモデルである。ここでは簡単のために入力データ x はある L 個のアイテム集合 $[L]$ の部分集合 $x \in 2^{[L]}$ だとする。今、モデルの構成要素の候補として $T = \{t_j \mid t_j \in 2^{[L]}\}_{j=1}^{J}$ があるとする。ここで、各 t_j は予測パターンを表しており、入力データ x にパターン t_j が含まれる場合に $y = 1$ または $y = 0$ と予測する（二値分類）ものとする。このような構成要素の候補集合 T は頻出パターンマイニング [13] や出現パターンマイニング [14] を使うことでデータから抽出できる。

IF (sex = Male ∧ current-charge-degree = Felony)
OR (priors > 3)
THEN recidivate-within-two-years = Yes

図 1: COMPAS データから学習されたルールセットの例 (4 節参照)

集合 T を使って、ルールセットの学習問題は以下のように記述できる。ここでは簡単のために $y = 1$ を予測するルールセットを考える。長さ I のルールセットは集合 $V = \{v_1, v_2, \ldots, v_I \mid v_i \in T\}$ により以下のように記述できる。

$$g(x|V) = \begin{cases} 1 & \text{if } \exists i \in [I], v_i \subseteq x, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

ルールセットは訓練データ $\{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$ から以下の精度最大化問題を解くことで学習できる。

$$\max_{V \subseteq T} \frac{1}{|\mathcal{N}_+|} \sum_{n \in \mathcal{N}_+} g(x^{(n)}|V), \text{ s.t. } |V| \leq I. \tag{3}$$

ただし $\mathcal{N}_+ := \{n \mid y^{(n)} = 1\}$ は正例集合である。問題 (3) は組合せ最適化問題であり、一般に最適解を求めるには指数時間を要する。ただし、この問題は被覆最大化問題 [15] と等価であり貪欲法により多項式時間で $(1 - 1/e)$ 近似解を得られる [16]。

3　提案法

本節では線形モデル及びルールモデルを列挙する問題を定義し、その列挙方法について述べる。より詳細な内容については参考文献 [6, 7] を参照されたい。

3.1　列挙問題の定義

列挙問題を定義するにあたり、まず Lasso とルールセットの学習問題が以下のように関数 $f(m)$ を最大化するモデル m を探す問題として記述できることを示す。

$$\max_{m \in \mathcal{F}_T} f(m). \tag{4}$$

Lasso において、モデル m は回帰係数 $\boldsymbol{\beta}$ と等価 $m = \boldsymbol{\beta}$ であり、集合 T は全特徴量の集合 $T = [d]$、そしてモデ

ルの集合 $\mathcal{F}_T = \{\boldsymbol{\beta} \in \mathbb{R}^d \mid \exists U \in 2^T, \forall i \notin U, \beta_i = 0\}$ は集合 T 上の全ての疎な回帰係数の集合である。また、$f(m)$ は問題 (1) の目的関数に負の符号をつけたものである。同様にルールセットの学習ではモデル m は集合 V と等価 $m = V$ であり、集合 T はモデルの構成要素集合、そしてモデルの集合 $\mathcal{F}_T = \{V \mid V \in 2^T, |V| \leq I\}$ は T から構成可能な長さ I 以下の全てのルールセットの集合である。また、Lasso とルールセットそれぞれについてモデルの台 $\mathrm{supp}(m)$ を $\mathrm{supp}(m) = \{i \mid \beta_i \neq 0\}$ 及び $\mathrm{supp}(m) = \{j \mid t_j \in V\}$ により定義する。

続いて、学習問題 (4) に基づいてモデルの列挙問題を定義する。まず学習問題 (4) において集合 T を任意の部分集合 $S \subseteq T$ に置き換えた以下の問題を考える。

$$\max_{m \in \mathcal{F}_S} f(m). \tag{5}$$

また、問題 (5) を解く適切な決定的アルゴリズム $\mathtt{Alg}_\alpha$ が存在し、任意の S に応じて問題 (5) の解としてモデル $m = \mathtt{Alg}_\alpha(S)$ を返すとする。ここで α はアルゴリズムの保証する解の近似比である。このとき、$\mathcal{M}_{\mathrm{all}} = \{\mathtt{Alg}_\alpha(S) \mid \exists S \in 2^T\}$ を $\mathtt{Alg}_\alpha$ により学習可能なモデルの全体とする。本研究ではモデルの列挙問題を『$\mathcal{M}_{\mathrm{all}}$ の中のモデルを目的関数 $f(m)$ の降順に上位 K 個列挙する問題』として定義する。

3.2 提案法：モデル列挙アルゴリズム

提案法では Lawler の K-best 列挙 [17] に基づいてモデルを列挙する。提案法では重複したモデルが生成されないように適切に学習問題の台 S を制約しながら逐次的にモデル候補を生成していく。

アルゴリズム 1 に提案法を示す。提案法ではヒープによりタプルと優先順位を管理する。タプル (m, S, F) はモデル m と二つの T の部分集合 S, F から構成される。ただし $m = \mathtt{Alg}_\alpha(S)$ である。タプルの優先順位は目的関数値 $f(m)$ であり、ヒープからタプルを抽出される際には $f(m)$ が最大のタプルが抽出される。集合 F はヒープに同じ S が重複して挿入されるのを防いで列挙を効率化するために使われる。

提案法ではまず台制約なしに問題 (4) を解いて $m = \mathtt{Alg}_\alpha(T)$ を得る。そして、タプル $(m, T, \emptyset)$ をヒープに挿入する。提案法では続いて以下の 3 ステップをモデルが K 個列挙されるまで繰り返す。

1. タプル (m, S, F) を抽出する（5 行目）。
2. モデル m を k 番目のモデル m_k として出力する（7-8 行目）。
3. 次のモデル候補を生成する。全ての $t_j \in \mathrm{supp}(m) \setminus F$ について、モデル候補 $m' = \mathtt{Alg}_\alpha(S \setminus \{t_j\})$ を学習し結果をヒープに挿入する（15-19 行目）。

Algorithm 1 モデル列挙アルゴリズム

```
1:  初期モデルの学習：m = Alg_α(T)
2:  タプル (m, T, ∅) をヒープに挿入
3:  M ← ∅
4:  for k = 1, 2, ... do
5:      タプル (m, S, F) をヒープから抽出
6:      /* モデル m を k 番目のモデル m_k として出力 */
7:      if m ∉ M then
8:          M ← M ∪ {m}
9:      end if
10:     /* K 個列挙したら終了 */
11:     if |M| ≥ K then
12:         break
13:     end if
14:     /* 次の候補モデルの生成 */
15:     for t_j ∈ supp(m) and t_j ∉ F do
16:         モデル候補の学習：m' = Alg_α(S \ {t_j})
17:         タプル (m', S \ {t_j}, F) をヒープに挿入
18:         F ← F ∪ {t_j}
19:     end for
20: end for
```

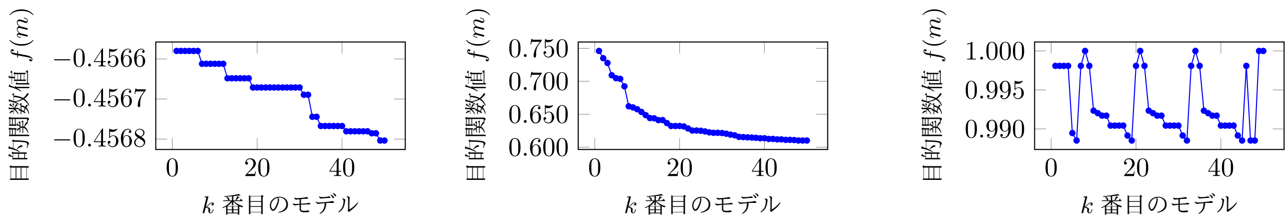
提案法において最も重要なステップはモデル候補を生成するステップ 3 である。ここで集合 F を使うことで冗長な重複した候補を生成しないように制約する。

提案法については以下が成り立つ。まず、列挙を停止しなければ最終的に $\mathcal{M}_{\mathrm{all}}$ の全てのモデルが列挙される。これは列挙の探索木の完全性と呼ばれる性質である。また、この完全性から以下が従う。

Theorem 1 (列挙の正しさ [7])**.** アルゴリズム 1 により出力されたモデル集合 $\mathcal{M}$ について以下が成り立つ。

1. $\forall k, \ell \in [K], k < \ell \Rightarrow f(m_k) \geq \alpha f(m_\ell)$.
2. $\forall m \in \mathcal{M}_{\mathrm{all}}, m \in \mathcal{M}$ if $\exists m' \in \mathcal{M}$ such that $\alpha f(m) > f(m')$.

性質 1 はモデルが目的関数の降順に近似的（$\alpha < 1$）または厳密（$\alpha = 1$）に列挙されることを保証する。性質 2 はアルゴリズム 1 において上位のモデルが近似的（$\alpha < 1$）にまたは厳密（$\alpha = 1$）に見落とされることがないことを保証する。Lasso では容易に大域解を求められるので学習アルゴリズム $\mathtt{Alg}_\alpha$ は厳密、つまり $\alpha = 1$ である。そのため、提案法による厳密な列挙が可能である。他方、貪欲法を用いるルールセットでは学習アルゴリズム $\mathtt{Alg}_\alpha$ の近似比は $\alpha = 1 - 1/e$ である。そのため、提案法によるルールセットの列挙では近似的な順序のみが保証され、厳密な順序での列挙は必ずしも保証されない。



(a) Lasso 列挙（Thaliana） (b) ルールセット列挙（COMPAS） (c) ルールセット列挙（Mushroom）

図 2: 提案法（アルゴリズム 1）による Lasso（線形モデル）及びルールセット列挙の結果。

4 実験結果

本節では、実データ実験により提案法で適切にモデルが列挙できることを示す。

Lasso（線形モデル）の列挙実験では以下のデータセットを用いた。

- シロイヌナズナ遺伝子データ（Thaliana [18]）：199 個体のシロイヌナズナの遺伝子を入力データ、FLC 遺伝子発現量を出力データとした。入力は特定の遺伝子パターンの有無を表す 216,130 次元の二値、出力は実数値である。実験では欠損を含むデータを除いた 167 個体のデータを使った。ρ の値は 0.01×167 とした。

ルールセットの列挙実験では以下の 2 つのデータセットを用いた。なお、ルールセットの学習には貪欲法を用いた。

- 犯罪経歴データ（COMPAS [19]）：6,489 人の犯罪者各個人に関する 19 個の各属性の有無を入力データ、3 年以内の再犯の有無（二値分類）を出力とした。モデルの構成要素集合 T は公開されている項目から少なくとも 50%の割合で $t \subseteq x \wedge y = 1$ となる項目 t だけを取り出して構成した。
- 毒キノコデータ（Mushroom）：8,124 個のキノコに関する 22 個の各属性の有無を入力データ、毒キノコか否か（二値分類）を出力とした。モデルの構成要素集合 T は出現パターンマイニング [20] を使って 90%以上の割合で $t \subseteq x \wedge y = 1$ となる項目 t だけを取り出して構成した。

図 2 に各データセットで上位 50 個のモデルを列挙した結果を示す。Lasso では容易に大域解を求められるので学習アルゴリズム $\mathtt{Alg}_\alpha$ は厳密、つまり $\alpha = 1$ である。そのため、定理 1 から Lasso は厳密な順序での列挙が保証される。実際、図 2(a) ではモデルが目的関数値が厳密に降順になるように適切に列挙できた。

ルールセットの学習には貪欲法を用いたため、学習アルゴリズム $\mathtt{Alg}_\alpha$ の近似比は $\alpha = 1 - 1/e$ である。そのため、定理 1 からルールセットは近似的な順序での列挙が保証されるが、厳密な順序での列挙は必ずしも保証されない。それにも関わらず、図 2(b) ではルールセットが目的関数値が厳密に降順になるように列挙できた。他方、図 2(c) では目的関数値が厳密に降順にはならなかった。しかし、目的関数値の変動は近似比 $\alpha = 1 - 1/e$ の幅に収まるものであり定理 1 が示唆する結果と合致した。

これら実験結果から得られた重要な知見の一つに『多くのデータで同等な目的関数値を達成するモデルが複数存在する』というものがある。実際、図 2(a) や (c) では全く同じ目的関数値を達成するモデルが複数見つかった。これは複数のモデルが訓練データを等しく説明できることを意味している。つまり、データマイニングなどで見つかったモデル一つだけに着目してしまうとデータに関する重要な知見を見落とす恐れがあることを示している。モデル列挙はこのような見落としの危険性を低減する上で特に有用である。

5 まとめ

本稿では線形モデル及びルールモデルの列挙問題を定義し、Lalwer の K-best 列挙に基づくをモデル列挙法について述べた。また、実データ実験を通じて提案法の妥当性を評価した。さらに、実データ実験において同等な目的関数値を達成するモデルが複数存在することを確認した。このような同等なモデルを列挙することで以下のような実用上の利点が期待できる。(i) ユーザにとってより納得感・信頼性の高いモデルを提供できる。(ii) 複数のモデル候補を比較検討することで、単一のモデルだけを考慮する場合よりもデータについての深い洞察が可能となる。

謝辞

本研究は JST ERATO 河原林巨大グラフプロジェクト（JPMJER1201）及び JSPS 科研費基盤 S（15H05711）の支援を受けた。

参考文献

[1] Hao Yi Ong, Dennis Wang, and Xiao Song Mu. Diabetes prediction with incomplete patient data. *Technical Report*, 2014.

[2] Kareem L Jordan and Tina L Freiburger. The effect of race/ethnicity on sentencing: Examining sentence type, jail length, and prison length. *Journal of Ethnicity in Criminal Justice*, 13(3):179–196, 2015.

[3] Himabindu Lakkaraju, Everaldo Aguiar, Carl Shan, David Miller, Nasir Bhanpuri, Rayid Ghani, and Kecia L Addison. A machine learning framework to identify students at risk of adverse academic outcomes. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1909–1918, 2015.

[4] Been Kim. *Interactive and interpretable machine learning models for human machine collaboration*. PhD thesis, Massachusetts Institute of Technology, 2015.

[5] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv:1702.08608*, 2017.

[6] Satoshi Hara and Takanori Maehara. Enumerate lasso solutions for feature selection. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pages 1985–1991, 2017.

[7] Satoshi Hara and Masakazu Ishihata. Approximate and exact enumeration of rule models. In *AAAI' 18(accepted)*, 2018.

[8] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[9] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and Regression Trees*. CRC press, 1984.

[10] Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo Seltzer, and Cynthia Rudin. Learning certifiably optimal rule lists for categorical data. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 35–44, 2017.

[11] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1675–1684, 2016.

[12] Salvatore Ruggieri. Enumerating distinct decision trees. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2960–2968, 2017.

[13] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, 1993.

[14] Guozhu Dong and Jinyan Li. Efficient mining of emerging patterns: Discovering trends and differences. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 43–52, 1999.

[15] Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

[16] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978.

[17] Eugene L Lawler. A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management science*, 18(7):401–405, 1972.

[18] Susanna Atwell, Yu S Huang, Bjarni J Vilhjálmsson, Glenda Willems, Matthew Horton, Yan Li, Dazhe Meng, Alexander Platt, Aaron M Tarone, Tina T Hu, et al. Genome-wide association study of 107 phenotypes in arabidopsis thaliana inbred lines. *Nature*, 465(7298):627–631, 2010.

[19] Nicholas Larus-Stone. `https://github.com/nlarusstone/corels`, 2017.

[20] Junpei Komiyama. `https://github.com/jkomiyama/qtlamp`, 2017.