

# RDB の構造を考慮したデータベースからの学習手法について

志村 薫<sup>†</sup> 杉浦 健人<sup>††</sup> 石川 佳治<sup>†††</sup>

<sup>†</sup>名古屋大学工学部電気電子・情報工学科 <sup>††</sup>名古屋大学大学院情報科学研究科

<sup>†††</sup>名古屋大学大学院情報学研究科

## 1 はじめに

ビッグデータ時代の今日、機械学習によるデータからの新たな知見の発見や獲得が研究のみならず実社会においても取り沙汰されており、さまざまな機械学習のツールが開発、公開されている。しかし、その多くは RDBMS (relational database management system) において一般的なデータ形式であるマルチテーブルでのデータ入力に対応しておらず、シングルテーブルでのデータ入力のみとなる。したがって、ユーザが機械学習において全特徴を得るためには、主キーと外部キーを用いてマルチテーブルを結合し、シングルテーブルへと変換する必要がある。この処理によって生じる遅延はデータサイズとともに増大し、ビッグデータを扱う上で無視できないものとなっている。

Kumar らはこの問題の解決策として特定の外部キーを、その外部キーが参照するテーブル内の全特徴の代表とみなすことでテーブルの結合を一部省略する手法を提案した。各外部キーについて、この手法が適用可能か否かの判定には RDB のスキーマ構造から導かれる tuple ratio を利用する。tuple ratio とは、テーブルの結合を省略することで生じるリスクを評価した指標であり、tuple ratio が小さいほどリスクが大きいことを表す [1,2]。

本稿では Kumar らの tuple ratio に基づく教師あり学習の高速化手法を実装し、その有用性を検証する。この手法を活用することにより、精度を落とすことなくテーブルの結合を省略することが可能となり、学習時間の削減が望める。

## 2 Tuple Ratio に基づく教師あり学習の高速化

本稿で検証する tuple raio に基づく教師あり学習の高速化手法 [1,2] について、概要を説明する。

この手法で想定されるデータセットの構造はスタースキーマである。ファクトテーブル  $S$  について、 $SID$  を主キー、 $Y$  をターゲット、 $\mathbf{X}_S$  を特徴ベクトル、 $FK_i$  を外部キー、 $n_S$  を行数とする。 $i$  番目のディメンションテーブ

ル  $R_i$  について、 $RID_i$  を主キー、 $\mathbf{X}_{R_i}$  を特徴ベクトル、 $n_{R_i}$  を行数とする。 $FK_i$  のドメインサイズ  $|D_{FK_i}|$  について、 $|D_{FK_i}| = n_{R_i}$  が成り立つものとする。すなわち、各ディメンションテーブルには結合に必要な十分なタプルのみが存在するものとする。このとき、 $TR_i \equiv \frac{n_S}{n_{R_i}}$  を  $i$  番目の tuple ratio とする。図 1 に上記の検証手法で想定されるデータ構造を示す。

機械学習の高速化に関する代表的な手法として特徴選択が挙げられる [3]。特徴選択は冗長な特徴を取り除くことにより学習の高速化を図れるが、データの中身というインスタンスに依存する。そのため、特徴選択ではデータが更新され中身が書き換わるたびに再計算が必要となる。一方、本稿で検証する tuple ratio に基づく手法は  $TR_i \equiv \frac{n_S}{n_{R_i}}$  という定義からわかるように、データの中身ではなくテーブルの行数というメタデータに依存する。そのため、この手法はデータが更新され中身が書き換わった場合でも、同様なスキーマ構造のデータセットに対して、更新以前の結果を再利用することが可能である。したがって、tuple ratio に基づく手法は特徴選択より汎用的な手法であるといえる。

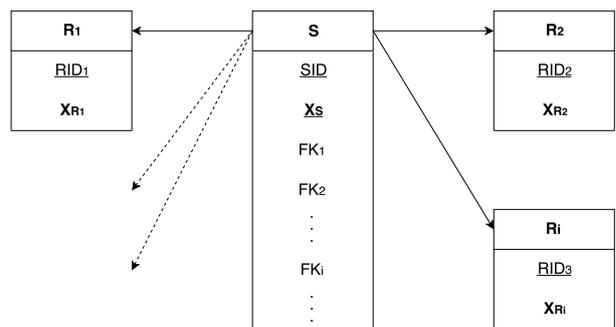


図 1 検証手法で想定されるデータ構造

## 3 実験

### 3.1 実験方法

ファクトテーブルと結合するディメンションテーブルの各組み合わせごとに機械学習を行い、予測精度および実行時間を計測し、それらと tuple ratio との関係を検証する。

本実験ではロソソ回帰を機械学習の手法として用いる。ロソソ回帰の実装には R の glmnet パッケージを利用する [4]。訓練データとテストデータを 1:1 に分割したのち、訓練データを使用して 10 分割交差検証を行い

Learning from Relational Databases Considering Database Structure  
 Kaoru Shimura<sup>†</sup>, Kento Sugiura<sup>††</sup>, and Yoshiharu Ishikawa<sup>†††</sup>  
<sup>†</sup>Department of Information Engineering, School of Engineering, Nagoya University  
<sup>††</sup>Graduate School of Information Science, Nagoya University  
<sup>†††</sup>Graduate School of Informatics, Nagoya University

表1 Last.fm のデータセットの詳細

S		R <sub>1</sub>		R <sub>2</sub>	
<i>SID</i>	再生 ID	<i>RID</i> <sub>1</sub>	アーティスト ID	<i>RID</i> <sub>2</sub>	ユーザ ID
<i>Y</i>	再生	<b>X</b> <sub>R<sub>1</sub></sub>	{Scrobble の回数, 再生回数, ロック, エレクトロニック, インディー, ポップ, ヒップホップ}	<b>X</b> <sub>R<sub>2</sub></sub>	{性別, 年齢, 出身, 登録年}
<b>X</b> <sub>S</sub>	∅	<i>n</i> <sub>R<sub>1</sub></sub>	4387	<i>n</i> <sub>R<sub>2</sub></sub>	25552
<i>FK</i> <sub>1</sub>	アーティスト ID	<i>TR</i> <sub>1</sub>	7.84	<i>TR</i> <sub>2</sub>	1.35
<i>FK</i> <sub>2</sub>	ユーザ ID				
<i>n</i> <sub>S</sub>	34375				

過学習を防ぐための最適な罰則の強さを求め、そのモデルについてテストデータを使用して予測精度および実行時間を計測する。

また、本実験では Last.fm のデータセットの一部を用いる [5]。このデータセットは Last.fm という音楽に特化した SNS 上での曲の再生の有無とアーティストおよびユーザの情報が組み合わさった内容となっている。表 1 に Last.fm のデータセットの詳細を示す。なお、R<sub>1</sub> の Scrobble とは Last.fm 独自の用語であり、Last.fm に曲の再生履歴を登録するという意味である。また、ロック、エレクトロニック、インディー、ポップ、ヒップホップはそれぞれ曲のジャンルを表す二値の特徴である。

### 3.2 実験結果

表 2 に実験結果を示す。JoinAll は R<sub>1</sub> および R<sub>2</sub> の両方を結合した場合、JoinR<sub>1</sub> は R<sub>1</sub> のみを結合した場合、JoinR<sub>2</sub> は R<sub>2</sub> のみを結合した場合、NoJoin は R<sub>1</sub> および R<sub>2</sub> の両方を結合しない場合を示す。予測精度は高い順に、JoinAll, JoinR<sub>2</sub>, JoinR<sub>1</sub>, NoJoin となった。また、実行時間は短い順に、NoJoin, JoinR<sub>1</sub>, JoinR<sub>2</sub>, JoinAll となった。

表 2 結合の組み合わせごとの予測精度および実行時間

	JoinAll	JoinR <sub>1</sub>	JoinR <sub>2</sub>	NoJoin
予測精度	0.707	0.660	0.704	0.657
実行時間 [s]	5.27	4.98	5.07	4.57

### 3.3 考察

表 1 から、TR<sub>1</sub> = 7.84, TR<sub>2</sub> = 1.35 であるため、R<sub>2</sub> の方が R<sub>1</sub> より tuple ratio が小さいことが読み取れる。また、実験結果から、JoinR<sub>1</sub> の予測精度が 0.660, JoinR<sub>2</sub> の予測精度が 0.704 であるため、JoinR<sub>2</sub> の方が JoinR<sub>1</sub> より予測精度が高いことが読み取れる。これらの結果から、tuple ratio が小さいほど結合を省略した際のリスクが大きいたことが確認できる。

さらに、実験結果から、JoinAll の予測精度が 0.707, JoinR<sub>2</sub> の予測精度が 0.704 であるため、JoinR<sub>2</sub> と JoinAll の予測精度に大きな差はないことが読み取れる。また、

JoinAll の実行時間が 5.27s, JoinR<sub>2</sub> の実行時間が 5.07s であるため、JoinR<sub>2</sub> の方が JoinAll より実行時間が短いことが読み取れる。これらの結果から、R<sub>1</sub> の結合は省略しても差し支えはなく、検証手法は予測精度を保ちつつ実行時間を短縮することが可能であると確認できる。

そして、tuple ratio に基づく手法はファクトテーブルおよびディメンションテーブルの行数というメタデータのみ依存し、データの中身というインスタンスには依存しない。したがって、例えば、Last.fm のデータ中の Scrobble の回数に変更があった場合でも、本実験の結果を再利用して R<sub>1</sub> の結合を省略することが可能であると考えられる。

## 4 まとめ

本稿では Last.fm のデータセットを用いて、ファクトテーブルと結合するディメンションテーブルの各組み合わせごとにロソ回帰を行い、予測精度および実行時間を計測し、それらと tuple ratio との関係を検証した。今後は Last.fm のデータセットやロソ回帰以外にもさまざまなデータセットおよび機械学習の手法を用いて同様に実験を行い、それぞれの比較、検証を進めていく。

## 謝辞

本研究の一部は、科研費 (16H01722) による。

## 参考文献

- [1] A. Kumar, J. Naughton, J. M. Patel, and X. Zhu, "To join or not to join?: Thinking twice about joins before feature selection," in *SIGMOD*, pp. 19–34, 2016.
- [2] V. Shah, A. Kumar, and X. Zhu, "Are key-foreign key joins safe to avoid when learning high-capacity classifiers?," *PVLDB*, vol. 11, no. 3, 2017.
- [3] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, *Feature Extraction: Foundations and Applications*. New York: Springer-Verlag, 2001.
- [4] "Package 'glmnet'." <https://cran.r-project.org/web/packages/glmnet/glmnet.pdf>.
- [5] "Last.fm dataset 360K." <https://www.upf.edu/web/mtg/lastfm360k>.