Vol.2022-MPS-137 No.1
2022/3/3

バイアスを持つサンプル標本からスマートフォンアプリケーション市場全体を捉えるための新たな方法論の開発

小島康至[1]　中野祥旗[1]　大野康明[2]　陳星言[2]　繁野麻衣子[1]　住田潮[1]

概要：移動体端末の普及により，オンラインで大規模なデータを収集して市場分析を行うことが可能となってきている．しかし，全数調査ではなくデータ取得元が一部に限定される場合，取得されたデータは適切なサンプリングを行わないと対象市場全体に対してバイアスを持つことになる．本論文の目的は，独立的に得られる対象市場全体に関する指標に合致するよう，収集されたデータから改めてサンプル抽出を行い，この偏りを是正するための方法論を確立することにある．スマートフォンアプリケーション市場で収集されるビッグデータに対し，独立的に得られる市場全体に関する所与の指標と整合性を持つような部分データ集合を抽出するモデルを作成しアルゴリズムを開発する．

キーワード：スマートフォンアプリケーション市場，アンダーサンプリング，遺伝的アルゴリズム，組合せ最適化

Development of a new approach for capturing the entire market based on massive biased sample data: Case of the smartphone applications market

KOSHI KOJIMA[†1]　YOSHIKI NAKANO[†1]　YASUAKI OHNO[†2]
XINGYAN CHEN[†2]　MAIKO SHIGENO[†1]　USHIO SUMITA[†1]

Abstract: Rapid development of mobile devices now enables one to conduct market analysis based on massive data collected from online transactions. When data collection sources are limited a part instead of complete survey, however, the resulting dataset could be biased and would not represent the entire market appropriately. The purpose of this paper is to develop an approach for extracting a sub-dataset from the collected dataset so that a given indicator, obtained independently, representing the entire market would become consistent with the same indicator obtained from the extracted sub-dataset. For the smartphone applications market, for which online transaction big data have been collected, a model to extract a sub-dataset for the above purpose is constructed and develop an algorithm using a genetic algorithm for it.

Keywords: Smartphone app market, under sampling, genetic algorithm, combinatorial optimization

1. はじめに

移動体端末の普及により，オンラインで大規模なデータを収集して市場分析を行うことが可能となってきている．しかし，全数調査ではなくデータ取得元が一部に限定される場合，適切なサンプリングを行わないと取得されたデータは対象市場全体に対してバイアスを持つことになる．本論文では,スマートフォンアプリケーション市場において，独立的に得られる市場全体に関する指標に合致するよう，収集されたデータから改めてサンプル抽出を行い，この偏りを是正する手法を示す.

データのバイアス是正に対し，抽出するデータを指標に合わせて削除していく方法にアンダーサンプリングがある．Bastista [1]は，アンダーサンプリングを用いた不均衡データ問題の対処に成功している．また，Tsai et al. [2]では，クラスタリング分析とインスタンス選択を組み合わせたクラスタベース・インスタンス選択と呼ばれるアンダーサンプリング手法を用いて，約 5000 のデータ，約 20 のクラスに対して手法の精度の高さを示している．

本研究では，スマートフォンアプリケーション市場で収集されるビッグデータに対し，独立的に得られる市場全体に関する所与の指標と整合性を持つような部分データ集合をアンダーサンプリングにより抽出する．収集されるデータには様々な情報があるが，デバイス属性，ユーザー属性と複数アプリケーションのアクティブユーザー数に着目し，アンダーサンプリングをおこなうときのモデルを作成する．この部分データを抽出する問題は，組合せ最適化問題として捉えることができ，整数計画問題として定式化して扱う

1 筑波大学
University of Tsukuba
2 フラー株式会社
Fuller, Inc.

方法と,遺伝的アルゴリズムを用いたアルゴリズムを示し,実際のデータに基づく実験データにおいて高精度でデータ抽出ができることを示す.

2. スマートフォンアプリケーション市場データからのサンプル抽出

スマートフォンアプリケーション市場で収集されたデータから市場全体に関する指標を計算することを考える．その際に，独立的に得られる対象市場全体に関する指標と比較した時にその精度が高まるような部分データを構成するユーザーを抽出する．ここで，元データにおける全体のユーザー（抽出前ユーザー）の集合を B，抽出するユーザー（サンプルユーザー）の集合を S と表す．つまり，S を求める問題となる．

データには，さまざまな情報が含まれているが，市場の指標としては代表的なアプリケーションの月間アクティブユーザー数（Monthly Active Users：MAU）に着目する．このとき,市場を構成する性別や年齢の構成比率を考慮する．さらに，サンプルユーザー集合は，一定期間ごとに更新をするが，そのときに大きくユーザーの構成が変わらないようにする指標としてデバイスの属性を利用する．つまり，市場全体を反映した部分データの抽出は，以下の条件 1-3 を満たすようなサンプルユーザー集合の抽出としてモデル化できる．

条件 1： サンプルユーザーの人数は決められた定数 l 以上.
条件 2： サンプルユーザー集合の使用デバイス属性の分布が過去のサンプルユーザーでの分布と大きく異ならない.
条件 3： サンプルユーザー集合でのユーザー属性別の各アプリケーションの MAU の割合が独立的に得られるデータと大きく異ならない.

条件2の使用デバイス属性には k 種類ある.この k 種類の属性の集合を $I_1, I_2, \dots, I_k$ と表す．各ユーザー $b\,(\in B)$ は $I_r\,(r = 1, \dots k)$ それぞれの中のちょうど一つの属性を持っている．つまり，属性 $i \in I_r$ に属する抽出前ユーザーを $B_i\ (\subseteq B)$ としたとき，B は $B_i\,(i \in I_r)$ で分割されている（$\cup_{i\in I_r} B_i = B, B_i \cap B_j = \emptyset\ (i, j \in I_r, i \neq j)$）．条件 2 は，各属性 $i\,(\in \cup_{r=1,\dots,k} I_r)$で，$|S \cap B_i|/|S|$ が過去のサンプルユーザーでの割合と大きく異ならないということである.

スマートフォンアプリケーション市場全体の推定ユーザー数を m，アプリケーション a の MAU（対象データとは独立的に得られる指標）を c_a としたとき，条件 3 ではサンプルユーザ集合 S でのアプリケーション a の MAU の割合を c_a/m に近づくように S を決めることが問題となる．このとき，抽出前ユーザー集合 B のユーザー属性偏りを考慮して MAU 割合を補正する．ここで，ユーザー属性とは性別や年齢であり，ユーザー属性の集合を J で表す．そして，属性 $j\,(\in J)$ に属する抽出前ユーザーを$B_j\ (\subseteq B)$ と表す．属性 $j\,(\in J)$ のユーザー数を補正するためのウェイトバック係数 w_j を算出して補正する．ウェイトバック係数は，$w_j \cdot |B_j \cap S|/|S|$がスマートフォンアプリケーション市場全体での属性 j の推定ユーザー割合となるような係数である．抽出前ユーザーのなかでアプリケーション a の月間アクティブユーザーの集合を$M_a\ (\subseteq B)$ としたとき,サンプルユーザー内でアプリケーション a の月間アクティブユーザーで属性 $j \in J$ に属するユーザー数が $|S \cap M_a \cap B_j|$ となり，$\sum_{j\in J} w_j\,|S \cap M_a \cap B_j|$ が，アプリケーション a に対する S での補正済みの MAU となり，これを $|S|$ で割った補正済み MAU 割合 $\sum_{j\in J} w_j \left|S \cap M_a \cap B_j\right|/|S|$ をc_a/m に近づける．条件 3 で対象とするスマートフォンアプリケーションは市場を反映するようにバランス良く複数選択する．条件 3 で対象とするアプリケーションの集合を A で表す.

このサンプルユーザーを決める問題は，条件 1-3 のもとで各ユーザーを選択するか否かを決定する組合せ最適化問題として捉えることができる．ユーザー $b\,(\in B)$ をサンプルユーザーに選ぶか否かを表す 0-1 決定変数 x_b を用いると，各条件は以下のように定式化できる．ここで，条件 1 は必ず満たすべき絶対条件で,

$$\sum_{b\in B} x_b \geq l \tag{1}$$

となる．条件 2 と条件 3 はなるべく満たすべき考慮条件である．この考慮条件を目的関数とするかどうかで定式化が異なるが，本研究では，条件 3 を目的関数として，条件 2 は与えられた属性 $i \in I_1 \cup I_2$ ごとにパラメータ z_i を定め,この値以下に近づければよいとする．過去の抽出前ユーザーの属性 $i\,(\in I_1 \cup I_2)$ の割合を p_i とすると,

$$-z_i \leq p_i - \frac{\sum_{b\in B_i} x_b}{\sum_{b\in B} x_b} \leq z_i, \quad \forall i \in \cup_{r=1,\dots,k} I_r \tag{2}$$

となる．条件 3 は非負変数 $y_a\ (a \in A)$ を用いて

$$-y_a \leq \frac{c_a}{m} - \frac{\sum_{j\in J} w_j \sum_{b\in B_j \cap M_a} x_b}{\sum_{b\in B} x_b} \leq y_a, \quad \forall a \in A \tag{3}$$

のもとで，$\sum_{a\in A} y_a$ の最小化を目的関数とする.

3. サンプル抽出手法

3.1 整数計画法

サンプル抽出は(1)-(3)式を制約条件とし，目的関数が$\sum_{a\in A} y_a$の最小化の 0-1 整数計画問題となる.ところが,(3)式は線形式ではないため，整数線形計画問題にするには，変数を増やすなどの対応が必要となる．しかし，0-1 変数は抽出前ユーザー数だけあり，大規模な整数計画問題となるため，変数を増やすことはせずに，(3) 式の分母をサンプルユーザー数の下限 l で置き換え

$$-y_a \leq \frac{c_a}{m} - \frac{\sum_{j\in J} w_j \sum_{b\in B_j \cap M_a} x_b}{l} \leq y_a, \quad \forall a \in A \tag{3}$$

と近似的に扱う．また，w_jの設定については，サンプルユーザーを元に算出することはせず，抽出前ユーザー集合Bを元に算出したもの($w_j \cdot |B_j|/|B|$が推定ユーザー割合となるような係数)に置き換え近似的に扱う．

3.2 遺伝的アルゴリズム

遺伝的アルゴリズムは，問題に対する解を個体の染色体，解の構成要素を遺伝子として表現し，交叉や突然変異といった操作で複数の個体を進化させて，より環境（問題の条件）に適応できる個体を見つける手法である．アルゴリズム中で保持する複数の個体の集合を個体群といい，アルゴリズムの繰り返しごとの個体群を世代とよぶ．遺伝的アルゴリズムのフローチャートを図 1 に示す．

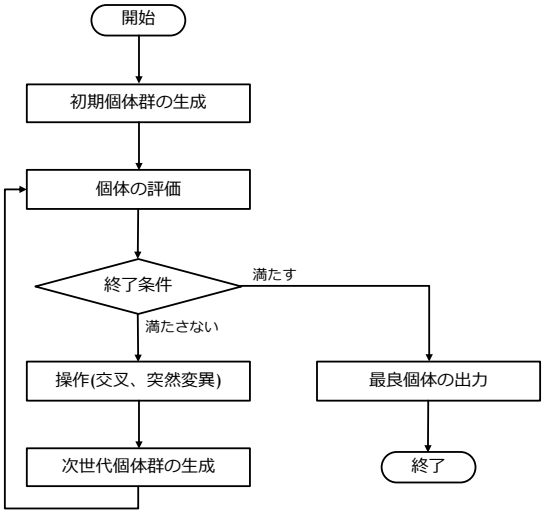

図 1　遺伝的アルゴリズム概略

Figure 1　Outline of genetic algorithm

本研究では，ユーザー $b\ (\in B)$をサンプルユーザーに選ぶかを表す 0-1 決定変数 x_b を遺伝子とする．抽出前ユーザーすべてで個体を構成しサンプルユーザーの抽出を行う方法と，ユーザー属性別に分類して個体を構成し，それぞれユーザー属性ごとに抽出を行う方法の２種類の比較をおこなう．前者を「全体抽出法」とよび，後者を「ユーザー属性別抽出法」とよぶ．全体抽出法での個体は，$(x_b)_{b\in B}$ となり，ユーザー属性別抽出法では $j \in J$ ごとの問題となり，そのときの個体は$(x_b)_{b\in B_j}$ となる．全体抽出法は遺伝子数がとても大きく収束に時間がかかることが予想される．それに対しユーザー属性別抽出法は，全体抽出法よりも個体内の遺伝子数が少ないので収束が速いことが予想できるが，属性ごとの抽出なのでサンプルユーザーの条件を評価しきれなく精度が悪くなる可能性がある．

全体抽出法の個体 $(x_b)_{b\in B}$ の適応度は，サンプルユーザーの条件 1-3 に基づいて計算する．条件１の (1) 式あるいは，条件２の (2) 式を満たしていないときにペナルティ値 P を与える．この二つの条件を満たしている場合には，条件 3 のアプリごとの MAU 割合の差の総和

$$\sum_{a\in A}\left|\frac{c_a}{m}-\frac{\sum_{j\in J} w_j \sum_{b\in B_j\cap M_a} x_b}{\sum_{b\in B} x_b}\right|$$

で与える．よって，適応度は小さいほうが好ましい．

ユーザー属性別抽出法では，ユーザー属性 $j\ (\in J)$ ごとに条件 1 の最低必要人数 l_j を定める必要がある．最低必要人数 l_j は，全体抽出における最低必要人数 l に，スマートフォンアプリケーション市場全体での属性 j の推定ユーザー割合を掛け合わせたものとする．次に，条件 2 のデバイス属性ごとの人数については，ユーザー属性 $j\ (\in J)$ であり，かつデバイス属性 $i\ (\in I)$ のユーザー割合 $p_{i,j}$ を用いる．アプリごとの MAU 算出方法について，全体抽出法では市場のユーザー属性比率に合わせてウェイトバック係数 w_j で補正を行っていたが，ユーザー属性別抽出ではその必要はなくサンプル内の MAU の割合をそのまま市場の MAU の割合との比較に用いる．つまり，

$$\sum_{b\in B_j} x_b \geq l_j$$

$$-z_i \leq p_{ij}-\frac{\sum_{b\in B_j\cap B_i} x_b}{\sum_{b\in B_j} x_b} \leq z_i, \quad \forall i \in \cup_{r=1,\ldots,k} I_r$$

を満たしていないときに，適応度はペナルティ値 P を与え，満たしているときには，

$$\sum_{a\in A}\left|\frac{c_a}{m}-\frac{\sum_{b\in B_j\cap M_a} x_b}{\sum_{b\in B_j} x_b}\right|$$

で与える．

次世代を作るために親となる個体を選択し，交叉することで子個体を生成する．親となる個体を選択することを複製選択とよぶ．交叉について多点交叉を用いる．また，交叉によって生み出された子個体の遺伝子を変化させる突然変異は各遺伝子に対して，確率 ρ でおこなう．その結果，現世代個体群と生成された子個体群のなかから次世代に生き残る個体を選択することになるが，これは生存選択と呼ばれる．

4. サンプル抽出結果

前節で示した手法をスマートフォンアプリケーション市場のデータをもとにした実験用データに適用してサンプルユーザーを抽出する．実験データは，十分な説明の上でユーザーの承諾を得て収集された Android スマートフォン内でのアプリケーション利用データを元に作成されたものである．このデータでは，ユーザーそれぞれにランダムにユーザーID が与えられ，各ユーザーID でいつどのアプリを起動，操作したかが記録されている．ユーザーID は全期間を通して一意に割り振られていて，個人を特定できない形式でデータが記録，蓄積されている．ユーザーの性別，年代情報はユーザーが入力した内容に基づいている．

計算は，Intel(R)，Core(TM) i5-9500，3.00GHz，メモリ 8.00GB の PC でおこなった．実験用データでは，デバイス

属性は 2 種類（$r=2$），$|I_1|=17,\ |I_2|=20,\ |J|=12, |A|=44$ で準備した．(2) 式の z_i はすべての $i \in I_1 \cup I_2$ で 0.05 と設定した．実験データでは抽出前ユーザー数を約 7 万としている．また，$l=\lceil 0.3|B| \rceil$ とする．遺伝的アルゴリズムでは，適応度のペナルティ値 $P=1000$，アルゴリズムの終了条件は，開始から 3 時間経過したときとする．実務利用での実現可能性から計算時間の上限を 3 時間とした．予備実験から，サンプルユーザー数はlに近い個数となることが多く，初期個体でのサンプルユーザー数もこの値に近いほうが良い結果が得られることから，初期個体群生成時の各遺伝子を 1 とする確率 $l/|B_j|$ とした．個体数は 1000 とした．

4.1 パラメータ設定

複製選択と生存選択にどのような方法を適用するかによって，解の多様性や収束の速さなどが異なる．ここでは表 1 に示す 4 ルールを比較する．これに，突然変異の確率 ρ を変えたときの得られた最良の評価値を図 2 に示す．図 2 は 5 回の試行の平均値である．ただし，複製選択で選択する親個体数は 200，交叉点数は 2 としている．この結果より，突然変異率の値にかかわらず，ルール C が全体抽出，ユーザー属性別抽出のいずれでも優れていた．よって，ルール C を採用する．

表 1　遺伝的アルゴリズムの選択のルール
Table 1　Selection rules for genetic algorithm

		生存選択	
		親個体群の適応度の低い順に子個体と入替え	各家族から適応度最良 1 個体とルーレットにより 1 個体選択
複製選択	ランダムに非復元抽出	ルール A（Iterated Genetic Search [3]）	ルール B（Minimal Generation Gap [4]）
	適応度上位個体から非復元抽出	ルール C	ルール D

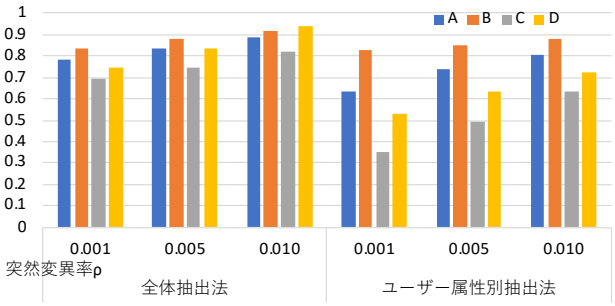

図 2　選択ルールによる最良値比較
Figure 2　Comparison of the best value for selection rules

次に，突然変異確率，複製選択時の親個体数，交叉点数のパラメータを決定するために，300 世代での各パラメータ値での最良評価値を比較した．突然変異率は，図 2 の結果からは 0.001 が最も良い結果であったが，より小さい値まで含めて，0.0000，0.0001，0.0005，0.0010，0.00500.0100 の 6 種類で比較をした．50 世代ごとの最良評価値の結果を図 3 に示す．ここでは，親個体数 500，交叉点数 2 とした．ρ は 0.0001，0.0005，0.0010 で大きな差がないことがわかる．この結果より突然変異確率 ρ は 0.0010 とした．複製選択時の親個体数は，100 から 600 まで 100 刻みで与え，比較した．300 世代での最良評価値の結果を図 4 に示す．ここでは，突然変異確率は 0.001，交叉点数は 2 としている．親個体数が増加するほうがよい評価値となったが，600 で評価値が悪化したため，1000 個体中複製選択時の親個体数は 500 とした．交叉点数は 1 から 12 点で比較をした．結果を図 5 に示す．5 点以上では大きな差がなく，もっともよい評価値を返した 8 と設定することとした．

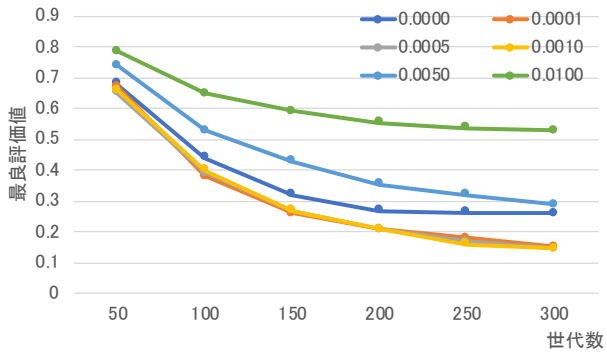

図 3　突然変異確率ρ比較
Figure 3　Comparison of　mutation rate

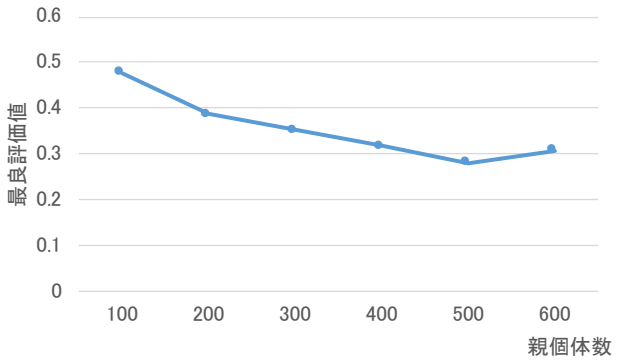

図 4　複製選択時の親個体数別 300 世帯での評価値
Figure 4　Comparison of the best value at 300 generations for parent population used in replication selection

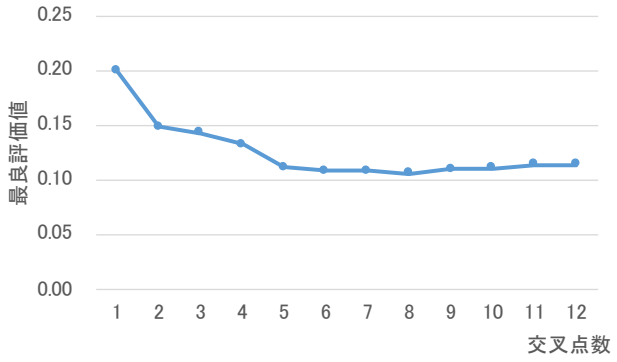

図 5　交叉点数別 300 世代での評価値
Figure 5　Comparison of the best value at 300 generations for the number of intersections

4.2 結果

3ヶ月分の実験用データ（a，b，c）で計算した結果を示す．3.1節で述べた整数計画問題はソルバー(CBC ver. 2.6.0)を用いて解く．整数計画問題を解く計算時間の上限も，遺伝的アルゴリズムと同様に 3 時間とし．この時間内に最適解が得られない場合には最良解を出力とした．結果を表 2 に示す．評価項目は，

- MAU 割合差総和：$\sum_{a\in A}\left|\frac{c_a}{m}-\frac{\sum_{j\in J} w_j \sum_{b\in B_j\cap M_a} x_b}{\sum_{b\in B} x_b}\right|$
- MAU 割当差最大値：$\max_{a\in A}\left|\frac{c_a}{m}-\frac{\sum_{j\in J} w_j \sum_{b\in B_j\cap M_a} x_b}{\sum_{b\in B} x_b}\right|$
- デバイス属性割合差総和：$\sum_{i\in I_1\cup I_2}\left|p_i-\frac{\sum_{b\in B_i} x_b}{\sum_{b\in B} x_b}\right|$
- デバイス属性割合差最大値：$\max_{i\in I_1\cup I_2}\left|p_i-\frac{\sum_{b\in B_i} x_b}{\sum_{b\in B} x_b}\right|$
- 抽出ユーザー数割合：$\sum_{b\in B} x_b/l$

であり，遺伝的アルゴリズムは 5 回の試行の平均を示す．整数計画問題では，いずれも 3 時間の計算上限時間内には最適解が得られなかった．

表 2 から，まずデバイス属性割合に関して，どの手法でも条件に適した抽出ができているといえる．MAU 割合について総和，最大値ともに遺伝的アルゴリズムのユーザー属性別抽出法が最も小さい値となっており，サンプルユーザー抽出に最も適しているといえる．しかし，ユーザー属性別抽出法は，全体抽出法に比べて，デバイス属性割合の評価項目が劣っており，属性別に抽出したことにより全体の割合差が悪くなってしまっている．よって，抽出の条件 2 と 3 のどちらを重視するかで手法の使い分けが必要といえる．また，(2) 式の z_i の値による調整も必要といえる．整数計画法では抽出前データを元にした近似的な値しか設定できないため，本モデルに対しては遺伝的アルゴリズムの方がより精度の高い市場全体の反映をしたサンプルユーザーを抽出できた．

5. まとめ

スマートフォンアプリケーション市場で収集されるビッグデータに対し，独立的に得られる市場全体に関する所与の指標と整合性を持つような部分データ集合をアンダーサンプリングにより抽出するモデルを構築し，整数計画法および遺伝的アルゴリズムで部分データを抽出する手法を示し，実験データによりその有用性を検証した．

指標としては MAU を用いたが，市場を表す別の指標の導入や，指標とするアプリケーションの選択方法の検討は今後の課題である．

表 2　手法の計算結果
Table 2　Results of calculation by each method

		整数計画法	遺伝的アルゴリズム	
			全体抽出法	ユーザー属性別抽出法
MAU 割合差総和	a	0.371	0.551	0.194
	b	0.942	0.608	0.228
	c	1.075	0.559	0.205
MAU 割合差最大値	a	0.084	0.108	0.032
	b	0.255	0.122	0.028
	c	0.316	0.120	0.025
デバイス属性割合差総和	a	0.395	0.434	0.392
	b	0.423	0.385	0.420
	c	0.334	0.420	0.410
デバイス属性割合差最大値	a	0.050	0.048	0.047
	b	0.050	0.045	0.045
	c	0.050	0.043	0.045
抽出ユーザー数割合	a	1.406	1.132	1.109
	b	1.003	1.152	1.077
	c	1.000	1.239	1.080

引用文献

[1] Batista, G.E.A.P.A., Prati, R.C., and Monard, M.C., “A study of the behavior of several methods for balancing machine learning training data,” ACM SIGKDD Explorations Newsletter, 6(1), 20-29, 2004.

[2] Tsai, C.-F., Lin, W.-C. , Hu,Y.-H., and Yao, G.-T. “Under-sampling class imbalanced datasets by combining clustering analysis and instance selection. I,” Information Sciences, 477, 47-58, 2019.

[3] Ackley, D.H. “An empirical study of bit vector function optimization,” Genetic Algorithms and Simulated Annealing, 170-204, 1989.

[4] 佐藤浩, 小野功, 小林重信, “遺伝的アルゴリズムにおける世代交代モデルの提案と評価,” 人工知能学会誌, 12 (5) , 734-744, 1996.