多次元データの探索分析のための多様性を考慮した可視化システム

野田昌太郎† 杉浦 健人† 石川 佳治†

† 名古屋大学情報学研究科 〒 464–8603 愛知県名古屋市千種区不老町

Email: {noda,sugiura}@db.is.i.nagoya-u.ac.jp, ishikawa@i.nagoya-u.ac.jp

あらまし 多次元データの探索的な分析における有用な手段として，棒グラフを用いたデータの可視化が挙げられる．しかし，データの量と複雑さが爆発的に増えた現在では作成可能な棒グラフの数が非常に多くなっており，ユーザがそれらを一つ一つ調べるのは困難である．そこで本稿では，多様性を考慮した探索的な可視化システムを提案する．網羅性と非冗長性を保障可能な多様化技術を適用することで棒グラフの集合から適切な要約（部分集合）を選択する手法を提案し，その要約結果を探索するための可視化システムを開発した．

キーワード 可視化，探索的なデータ分析，データベース

1 はじめに

多次元データの探索的な分析は依然として重要な位置を占めている．データマイニングや機械学習など，自動でデータを分析する手法が台頭してきているが，その結果得られる知見の種類は限定的である．そのため，人間による対話的な多次元データの探索分析は広く行われている．

多次元データの探索的な分析における有用な手段として，棒グラフを用いたデータの可視化が挙げられる．ユーザは多次元データを視覚的に把握するために，データセットからプロットしたいサブセットと縦軸，横軸を選んで棒グラフを作成する．このような棒グラフの作成を繰り返し，ユーザはトレンドを把握し，外れ値を発見し，パターンを特定しながらデータへの理解を深める．

しかし，データセットのサイズと複雑さが増したことで，棒グラフを用いたデータの視覚的な探索は一層困難になっている．例えば，アメリカのフライトデータ [1] の探索を考える．このデータセットはフライト 1 回を 1 行に記録したものであり，出発する州，到着する州，キャリア，キャンセルコード，日時などが記録されている．データの一部を表 1 に示す．同データセットの次元のうち，取りうる値の数が多いものを 4 つ挙げると表 2 のようになる．ユーザがこのデータセットを探索するときに，出発する州ごとのキャンセルコードの分布に注目したとすると作成できる棒グラフは 51 種類である．さらなる分析のために，この中から棒グラフを 1 つ選びキャリアごとの分布を確認しようとすると作成可能な棒グラフは 1000 種類を超える．同様の操作を繰り返していくと，作成可能な棒グラフは 4 次元で 40 万以上となり，これらの棒グラフ集合の中から人手で有用なものを探索していくのは現実的ではない．

この膨大な棒グラフ集合の探索を支援する研究として，Lee らの手法 [2] が挙げられる．Lee らの手法 [2] では，各棒グラフをその棒グラフの親となる棒グラフとの非類似度をベースとしたスコア化を行った．ユーザは自身の注目する棒グラフと結果のサイズを指定することで，そのグラフとの非類似度が高い子のグラフ，さらに子のグラフとの非類似度が高い孫のグラフ，というように結果集合へ棒グラフを追加していく．

しかし，同手法は各次元の取りうる値の数が多いデータセットでは上手く動作しない．たとえば，Lee らの手法でアメリカのフライトデータ [1] を分析すると，指定したグラフからは遠い棒グラフ集合が推薦されるものの，推薦された棒グラフ同士を比較すると類似したものが含まれて冗長となってしまう．加

表 1: アメリカのフライトデータ [1] の例（一部）

出発する州	キャリア	月	キャンセルコード	件数
カリフォルニア州	*	*	A	35479
カリフォルニア州	*	*	B	12786
カリフォルニア州	*	*	C	8108
カリフォルニア州	*	*	D	77
ネバダ州	*	*	A	6516
ネバダ州	*	*	B	2880
ネバダ州	*	*	C	1161
ネバダ州	*	*	D	3

表 2: アメリカのフライトデータ [1] の次元の異なり数が大きい次元 4 つ

次元	異なり数
出発する州 (State)	51
キャリア (Carrier)	23
日 (Day)	31
月 (Month)	12

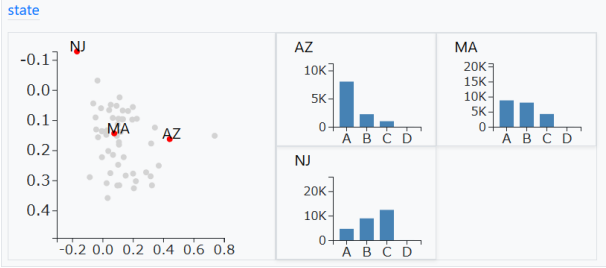

図 1: 次元「出発した州」の要約結果

図 2: Overview 画面

えて，これらの類似した棒グラフが結果集合を埋めてしまうため表示されないグラフも多くなり，そこからユーザがデータセットの概観を理解するのは困難である．

そこで，本研究では次元の取りうる値の数が多い多次元データに対して結果集合の網羅性と非冗長性を両立した多次元データの要約手法を提案する．提案手法では与えられた棒グラフ集合に対して，DisC Diversity [3] を修正した多様化アルゴリズムを適用することで結果集合全体の網羅性と代表集合内の同士の非冗長性を満たすように代表を抽出する．

また，上記の要約手法を用いた多次元データから棒グラフを探索するシステムを開発した．提案システムでは，抽出された代表を図 1 のように (1) 集合全体の分布を示す散布図，(2) 代表的な棒グラフ集合の 2 つの要素を用いて描画した．

このシステムは Overview（図 2）と Zoom（図 3）の 2 つの画面から構成される．Overview 画面では分析したい次元集合と部分データセットを指定することで，それぞれの次元での分割結果の要約が一覧が確認できる．その一覧から注目する次元を選ぶことで，Zoom 画面でより詳細な分割結果を確認可能となっている．

2 関 連 研 究

本章では，提案システムとの関連度の高い棒グラフの探索を支援するする 2 つの研究（Lee らの手法 [2]，Zenvisage [4, 5]）に対する網羅性と非冗長性の比較を行う．その後に，他の関連研究を簡単に紹介する．

表 3: 棒グラフの探索を支援する研究の比較

タイトル	入力	結果の網羅性	結果の非冗長性
Lee らの手法 [2]	注目棒グラフ	×	×
Zenvisage [4, 5]	折れ線グラフ集合	○	×
提案手法	棒グラフ集合	○	○

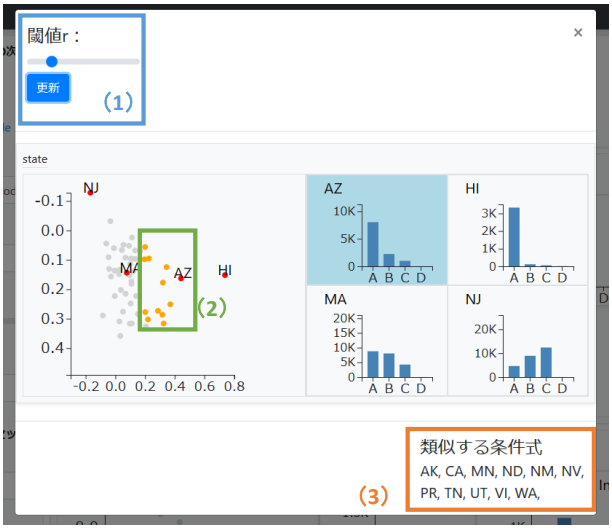

図 3: Zoom 画面

Lee らの手法 [2]：Lee らの手法 [2] では，結果集合のサイズ k と注目する棒グラフを受け取り，その子孫から棒グラフ集合を抽出する．子孫にあたる棒グラフは自身にとって最も近い親との非類似度でスコア化され，そのスコアが高いものから順番に結果集合へ追加する．同手法では次元の取りうる値の数が 5 から 10 程度のデータセットでは網羅性が高い結果を計算することができるが，それより高くなる場合には網羅性，非冗長性は低くなる．

Zenvisage [4,5]：Zenvisage [4, 5] は，データセットから作成可能なグラフから求めるトレンドを探索するためのフレームワークである．ユーザは専用の言語 ZQL [4, 5] でシステムへの問合せを行い，その条件にヒットするグラフを確認してまた新たな問合せを発行する，というように繰り返し問合せを行い，データセットを探索する．

同システムの機能の 1 つとして，代表トレンド，異常トレンド抽出がある [4]．同機能では，ユーザが指定したトレンド間の距離関数を利用して k-means に基づくクラスタリングを行い，代表トレンドと異常トレンドを抽出する．k-means 法では，それぞれのシードが他の点をどれだけ上手くカバーできているかを距離で評価するため結果集合の網羅性が高い代表を抽出できる．しかし，それぞれのシード同士の距離が近くなってしまう傾向があるため非冗長性は満たさない．

他の関連研究：データ探索の手法はデータベース分野で広く研究されている [6–8]．近年の研究では，Tang ら [8] の手法が挙げられる．同手法では，抽出する知見を一貫した上昇/下降トレンド，および外れ値の 2 つに絞ることで複雑な特徴変換から得られる知見の発見を自動化した．

可視化を用いたデータ探索の支援も，多くのシステムが提案されている [5, 9–11]．SEEDB [9] では，与えられたサブセット 2 つに対して距離が遠くなるような (次元, 集約属性, 集約関数) の組を k 個抽出して推薦する．Voyager [10]，Voyager2 [11] では，ユーザがデータセット全体に対してさまざまな設定で可視化を行うことを支援するシステムを提案した．しかし，Lee ら

の手法 [2]，Zenvisage [4,5] 以外ではサブセットの可視化探索を取り扱った主要な研究は存在しない．

3 DisC Diversity を利用した棒グラフ集合の要約

提案手法では，棒グラフ集合から冗長性を排除するために多様化技術 Greedy-DisC [3] を拡張したものを適用した．本章では，Greedy-DisC を適用して得られる代表点集合である r-DisC diverse subset について説明し，その後に拡張したアルゴリズムについて解説する．

3.1 r-DisC diverse subset

r-DisC diverse subset [3] は，Drosou らが計算方法を提案した多様なオブジェクト集合の概念である．それは，与えられた類似度の閾値 r とオブジェクト集合 O に対して，以下の 2 条件を満たすサブセット S として定義される．

$$\forall s_1, s_2 \in S, d(s_1, s_2) \geqq r \tag{1}$$

$$\forall o \in O, \exists s \in S, d(o, s) < r \tag{2}$$

式 (1) により，抽出したサブセット内の点がお互いに非類似となる．そのため，結果集合内には冗長な点は含まれなくなる．また式 (2) により，与えられた O のどの点もいずれかの代表点と類似する．そのため，抽出された代表はすべての点を十分に網羅できる．

以上の 2 つの条件から，抽出した代表点は元の結果集合から冗長性がなく，かつ元の結果集合全体を網羅したサブセットとなる．この 2 つの条件は，可視化を用いたサブセット探索における 2 つの課題 (冗長なグラフ，結果の網羅性の欠如) と合致すると考え，本研究では r-DisC diverse subset を計算することで結果の要約を行った．

3.2 Overview 画面での要約手法

Overview 画面ではデータセットの概観を示すために複数の次元での分割結果を一度に表示する．そのため，画面のスペースを考えると 1 つの分割結果に対して表示できる棒グラフの数には上限がある．しかし，前節で説明した r-DisC diverse subset は非類似度の閾値 r から計算するため結果集合のサイズを制限することが出来ない．そこで同手法を改良し，結果集合のサイズ k と非類似度の閾値 r の 2 つのパラメータから r-DisC diverse subset [3] を計算する (r, k)-DisC を考案した．本節では同アルゴリズムの説明を行う．

(r, k)-DisC： (r, k)-DisC アルゴリズムでは，非類似度の閾値 r と代表数の上限 k を受け取り，互いに非類似で，かつ k 個以下となるような代表点集合を計算する．詳細なアルゴリズムを Algorithm 1 に示す．なお，アルゴリズム中の GreyObjects, WhiteObjects は，オブジェクト集合からそれぞれ grey のもの，white のものを返す関数を表す．

このアルゴリズムの 1-9 行目は Greedy-DisC [3] の処理を行っており，10-15 行目が今回拡張した部分である．また，前処理として棒グラフ集合のすべてのペア $v_1, v_2 \in V$ とその非類似度を記録したタプルを非類似度で昇順ソートしたリストを D を準備する．

まず，全てのオブジェクトの状態，最終的に得られるサブセットと結果集合の非類似度の閾値 r_{ans} を初期化する (2-4 行目)．各オブジェクトの状態は white を，サブセットは空集合を，結果集合の非類似度は入力された r をそれぞれ代入する．

続いて，与えられた非類似度の閾値 r に基づいて Greedy-DisC を適用して代表点を導出する (6-9 行目)．同アルゴリズムでは，閾値 r 以内にある white のオブジェクトが最も多いものを順番に black へ変更する．その際，black への変更と同時に閾値 r 以内にある white のオブジェクトをすべて grey に変更する．この処理を white のオブジェクトがなくなるまで繰り返す．

6-9 行目の Greedy-DisC の結果，得られた代表点集合が k 個以下ならば，10-12 行目の処理をスキップして Greedy-DisC の結果をそのまま返す．k 個より大きい場合は，10-12 行目のループ処理により少しずつ閾値 r を大きくすることで代表点を減らしていく．

10-12 行の部分では，前計算された昇順でソート済み非類似度リスト D から，値が閾値 r より大きいタプルを小さい順に読み出してループ処理を行う．各ループでは閾値 r の更新と，更新後の閾値 r に対して Algorithm 2 を適用して DisC diverse subset の計算を行う．閾値 r を大きくしていくと代表点は減少していくため，このループを繰り返して代表の数が k 個以下になったところで処理を終了する．

この処理で得られる結果集合は，9-12 行目を実行せずに終了した場合は入力した非類似度の閾値 r に対して DisC diverse subset となり 10-12 行目の処理が走った場合は，最後に読み込んだタプルの非類似度 r に対して，DisC diverse subset となる．

colorObject：colorObject は，与えられた 2 つのオブジェクトから状態の変更を行うアルゴリズムである．オブジェクトの状態は 3 種類あるため，2 つのオブジェクトの状態の組合せは 9 通りあるが，grey の状態のオブジェクトは既にカバーされている点のため他のオブジェクトの状態には影響しない．そのため，(white,white)，(white,black)，(black,black) の 3 種類の組合せのみを考えればよい．

(white,white) のケースでは，番号の若いほうを代表として選択して black に変更する．このとき，もう片方のオブジェクトは選んだオブジェクトと類似となるため grey に変更する．

(white,black) のケースでは，white のオブジェクトが black のオブジェクトと類似しているため white のオブジェクトを black へ変更する．

(black,black) のケースでは，代表点同士が類似になってしまったのでどちらかを削除しなければならない．ここでは，網羅性を考慮して類似する grey のオブジェクトが多い方を残して小さいほうを削除する．また，代表を削除したことによる不整合を防ぐために，削除された代表と類似していた grey のオブジェクト集合に対して状態の再計算を行う．

3.3 Zoom 画面での要約手法

Zoom 画面では，要約の際のサイズの制約はないため Greedy-DisC [3] を適用する．ユーザの指定した非類似度の閾値 r に基

Algorithm 1: (r, k)-DisC

Data: 昇順ソート済み非類似度リスト D，代表のサイズ k ($k > 0$)，非類似度の閾値 r，棒グラフ集合 V
Result: r-DisC サブセット C, r_{ans}

```
begin
    // 状態の連想配列 s，結果集合 C の初期化
    C ← φ
    r_ans ← r
    foreach v ∈ V do
        s[v] ← white
    while |WhiteObjects(s, V)| > 0 do
        v_i ← max_{v∈WhiteObjects(s,V)} |v.neighbor|
        s[v_i] ← black
        foreach v_j ∈ v_i.neighbor do
            s[v_j] ← grey
    while |V| − |GreyObjects(s, V)| < k do
        // 非類似度 r より大きいタプルを値が小さい順にリスト D から読み出し
        (v_1, v_2, dist(v_1, v_2)) ← head(D)
        colorObjects(v_1, v_2, s)
        r_ans ← dist(v_1, v_2)
    C ← V \ GreyObjects(s, V)
    return
```

Algorithm 2: colorObject

Data: $v_1, v_2 \in V$, 状態の連想配列 s
Result: 状態の連想配列 s

```
1  begin
2      if (s[v_1], s[v_2]) = (white, white) then
3          s[v_1] ← black
4          s[v_2] ← grey
5      else if (s[v_1], s[v_2]) = (black, white) then
6          s[v_2] ← grey
7      else if (s[v_1], s[v_2]) = (white, black) then
8          s[v_1] ← grey
9      else if (s[v_1], s[v_2]) = (black, black) then
10         v_1.neighbor と v_2.neighbor のサイズを比較して，小さいほうを v とする
11         s[v] ← grey
12         V_renew ← φ
           // black の削除で white になった点の塗り直し
13         foreach v_n ∈ v_n.neighbor do
               if v_n has no black neighbor then
14                 s[v_n] ← white
15                 V_renew ← V_renew ∪ {v_n}
16         foreach v_r ∈ V_renew do
17             foreach v_rn ∈ v_r.neighbor do
18                 s ← colorObjects(v_r, v_rn)
19     return
```

づき，r-DisC diverse subset を計算しその結果を Zoom 画面に表示する．また，ユーザが閾値 r の値を対話的に調節する場合は Greedy-Zoom-In，Greedy-Zoom-Out(a) [3] を適用して代表点の再計算を行う．

4 提案インターフェース

本章では，3 章で説明した分割結果の要約を探索するための可視化システムについて説明する．提案インターフェースはデータの外観を表示する Overview 画面 (図 2) と注目する分割結果の詳細を確認する Zoom 画面 (図 3) の 2 種類から構成される．これは，Shneiderman による “Overview first, zoom and filter details on demand” という情報可視化のガイドライン [12] を参考にしたものである．

4.1 Overview 画面

Overview 画面は（図 2）は，左側のメニューバー (図中 (1))，右側の分割結果の描画領域 (図中 (2)，(3)) から構成される．

メニューバー：メニューバーからは，分割用のパラメータ，代表抽出用パラメータの 2 種類を指定可能である．分割用パラメータとしては，分割に使用する次元，注目するサブセットと横軸を指定することができる．ここで指定した次元セットを用いて，注目するサブセットを分割した結果を図 2 右下のように表示する．代表抽出用パラメータとしては，棒グラフ同士の非類似度の閾値 r と代表の数 k を指定可能である．この 2 つのパラメータから，(r, k)-DisC を実行してシステムは代表的な棒グラフを抽出する．

分割結果の描画領域：画面右側の分割結果の描画領域には，分割前の棒グラフと分割結果の一覧を表示する．分割前の棒グラフは図 2 右上 (図中 (2)) に，分割結果の一覧はその下 (図中 (3)) に順番に表示する．この画面では，分割前の棒グラフも表示するため，結果集合内の比較だけでなく分割前後の比較も可能である．分割結果一覧の部分には 3 章で提案した要約手法で抽出された棒グラフを表示する．その際，付加的な情報として図 2(3) 左側に表示されているように分割結果全体の分布を表す散布図も同時に表示する．

散布図の描画方法：棒グラフ集合から散布図を生成するために，提案システムでは 2 つの主成分を計算する．この 2 つの主成分を散布図の 2 つの軸としてプロットすることで，棒グラフ集合を散布図へ変換する．

主成分を計算するために，まずデータセット全体を各次元で別々に分割する．例えば，キャリア，月，出発する州の 3 次元を持つデータセットでは，データセット全体をキャリア，月，出発する州で分割したの 3 種類の部分データセット集合を得る．続いて，各部分データセット集合とユーザの指定した横軸から棒グラフ集合を計算する．上の例では，3 つの部分データセット集合からそれぞれ棒グラフ集合を計算するため，3 種類の棒グラフ集合を得る．最後に，上記の計算で得られた棒グラフ集合すべてを学習データとして主成分を計算する．このとき，すべての棒グラフのカテゴリの順序は同じにして学習させる．例

えば，上の操作の結果 3 種類の棒グラフ集合が得られたとすると，その 3 種類の棒グラフ集合に含まれる全ての棒グラフを学習データとして主成分を計算する．

この計算方法では，すべての棒グラフを同じ主成分を用いて射影するため，散布図上の座標が同じ点は同じ棒グラフとなる．そのため，Overview 画面上で異なる分割結果の散布図同士を比較することが可能となる．

画面の切り替え方法：この画面からは，分割するサブセットの変更と Zoom 画面の起動の 2 つの操作ができる．分割するサブセットの変更は表示されている棒グラフのダブルクリック，もしくは条件式の手打ちにより変更先のサブセットを指定して実行する．この操作を行うと画面右側の描画領域が更新され，分割元の棒グラフ，分割結果の一覧が指定したサブセットのものとなる．

Zoom 画面は，表示されている散布図，棒グラフの背景部分をクリックすること起動する．起動すると，Overview 画面上に重なるように Zoom 画面用のモーダルウィンドウが立ち上がる．

4.2 Zoom 画面

Zoom 画面では，図 3 のように 1 次元でのデータセットの分割結果を拡大して表示する．結果の要約は Overview 画面と同様に概観を示す散布図，代表的な棒グラフ集合で表現する．

Zoom 画面と Overview 画面の違いは，代表の数を対話的に増減可能な点と代表と非類似度 r 以下の棒グラフの条件式を確認できる点の 2 つである．Zoom 画面では，描画領域が広く取れるため Overview 画面のように代表のサイズ k を指定せず非類似度の閾値 r のみから計算する．図 3 のモーダル上部 (図中 (1)) の閾値 r の値を増減することで，代表の数を増減させながら各グラフをチェックすることが可能となる．

また，Zoom 画面では代表的な棒グラフへマウスを合わせることでその代表と非類似度 r 以下の棒グラフの条件式を確認することが可能である．マウスを棒グラフ上に合わせると，散布図上でその代表と類似する点がオレンジ色に変わり (図中 (2))，図 3 下部 (図中 (3)) のようにに類似する条件式の一覧を出力する．これにより，ユーザは各代表点の影響力を調査したり，類似する条件式から知見を得ることができる．

4.3 実　　装

提案システムはサーバクライアント方式を用いて実装した．サーバ側は，予め計算されたデータキューブ [13] からビューの読み出しと代表棒グラフの計算・散布図の計算を行う．実装に用いた言語は C++17 で，クライアントとは HTTP 通信でリクエスト処理を行う．なお，サーバ側の HTTP 通信用のライブラリとして Mongoose [14] を使用した．

クライアント側では，サーバから送られてきたデータからグラフを構築してユーザへ提示する．実装には JavaScript を使用し，データの可視化ライブラリは D3 [15] を使用した．

5 ユースケース

本章では，アメリカのフライトデータ [1] のうち，2003 年から 2008 年のキャンセルされたフライトに関する部分集合を用いて提案システムを利用した分析方法について説明する．今回のユースケースでは次元はキャリア，出発した州，月，キャンセルコードの 4 種類を利用する．なお，キャンセルコードは A がキャリア，B が天気，C が航空宇宙システム，D がセキュリティである．

分析者は，このデータのキャンセルコードについて分析することでキャンセルの要因と相関を把握して今後のフライトに生かしたいと考えている．そこで，まずはデータをロードし，使う次元にキャリア，出発した州，月を，横軸にキャンセルコードをを指定すると図 2 のように指定した次元での分割結果の要約が表示された Overview 画面が表示される．

図 2 の各次元の散布図を確認していくと，月での分割結果は大きく 2 つのグループに分割されてことが分かる．それぞれのグループの条件式を詳しく見るために，月の分割結果の背景をクリックして Zoom 画面を立ち上げ，閾値 r を調節すると，図 4 の画面が得られる．1 月の棒グラフへカーソルを合わせて条件式を確認すると類似する棒グラフは 1,2,3,9,12 月と分かる．そのため，10 月のグループは 4,5,6,7,8,10,11 月で構成されていると分かる．このことから，冬と 9 月にはカテゴリ B(天気) によるキャンセルが多くなるということが分かる．

図 2 の画面に戻って他の散布図を確認すると，出発した州がニュージャージー州（NJ）の点とキャリアのエクスプレスジェット航空の点が，どちらも散布図上の左上の端に位置しており，非常に近くなっている．この 2 つのサブセットについて詳しく見るためにダブルクリックして分割すると，それぞれ図 5，図 6 の結果が得られる．

図 5 のニュージャージー州の分割結果では，キャリアのグラフの大多数は ATA 航空のようにカテゴリ A と C が同じ割合の点が非常に多い．しかし，縦軸のスケールを確認するとエクスプレスジェット航空が他のグラフよりも圧倒的に多いため全体のグラフはエクスプレス航空に引っ張られていることが分かる．

一方，図 6 のエクスプレスジェット航空の出発した州ごとの分割結果を見るとニュージャージー州だけが異常なわけではなく全体として代表が A が多いグラフ，B が多いグラフ，C が多いグラフと大きくばらついており，それらを平均すると元のグラフが得られることが分かる．このことから，ニュージャージー州が外れ値になっているのはエクスプレスジェット航空のフライトが多いことに起因すると考えられる．

6 他の多様化技術との比較

本章では，代表点から要約を作成する際に，提案手法を使った場合と他の多様化技術を使った場合について比較を行う．具体的には，広く利用されている多様化技術である MAXMIN [16] およびクラスタリングアルゴリズムの k-means [17] と (r, k)-DisC を比較することで取れる代表点の特徴について分析する．なお，得られる結果のサイズを同じにするため (r, k)-DisC では $r = 0$ とし，非類似度と距離は全ての手法でユークリッド距離を用いた．

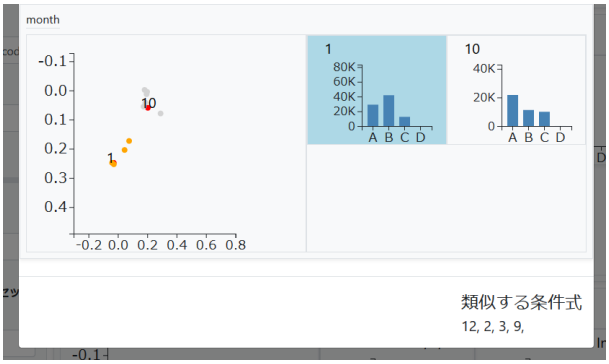

図 4: 月に関する Zoom 画面

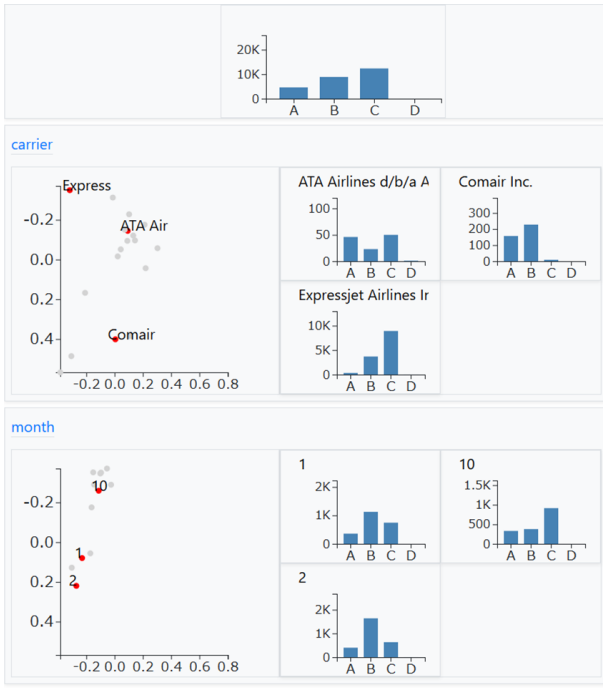

図 5: ニュージャージー州に関する表示

データセットは，5 章で使用したフライトデータ [1] に加えてアメリカのバードストライクのデータ [18] を利用した．このデータセットは，バードストライクが起こった時の状況と受けたダメージについて記録したものである．ここでは，横軸に雲量，分割次元には同データセット内で取りうる値の数が最も多いキャリア (operator) を利用して比較を行う．なお，この次元の取りうる値の数は 177 種類である．

代表点の視覚的な比較：k=5 として，各アルゴリズムで導出した代表点を図 7,8,9 に示す．

MAXMIN 法（図 7）では，代表間の最小距離が最大となるように点を抽出する．そのため散布図を見ると代表点が端に集まっていることが確認できる．また，棒グラフ集合も視認した際に区別が難しいような類似した分布が含まれておらず非冗長になっている．しかし，散布図中の最も密集した場所を表す代表が存在しないため，結果集合全体を網羅した抽出ができているとはいえない．

k-means 法（図 8）では，クラスタ内の平均距離が小さくなるようにシードの位置を抽出する．そのため散布図を見ると，

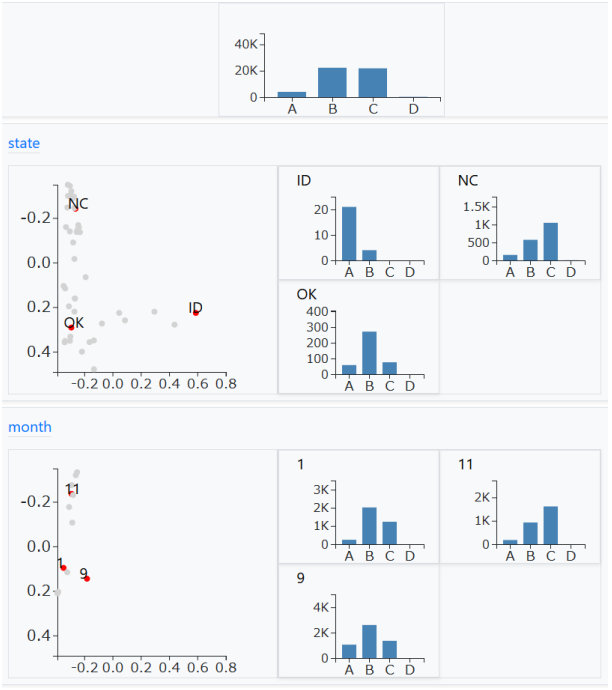

図 6: エクスプレスジェット航空に関する表示

図 7: MAXMIN 法 [16] による表示

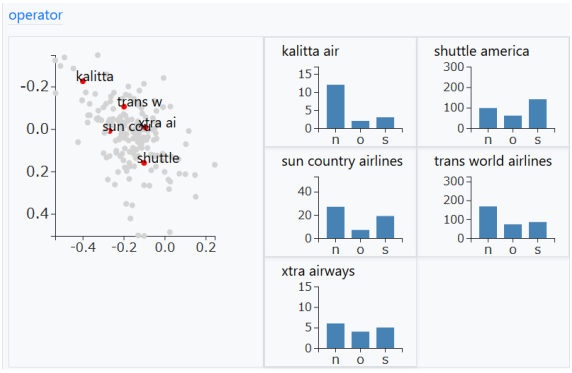

図 8: k-means 法 [17] による表示

点が密集している部分から代表が抽出されていることが確認できる．これらの抽出された点の棒グラフを確認すると american airlines と great lakes airlines を筆頭に分布の類似度が高いものが多く抽出されおり，非冗長な代表の抽出ができているとはいえない．

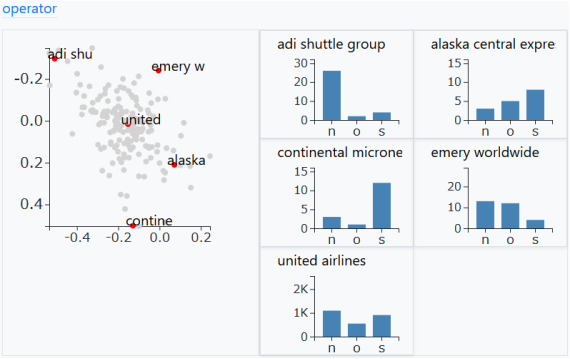

図 9: (r, k)-DisC(ただし $r = 0$) による表示

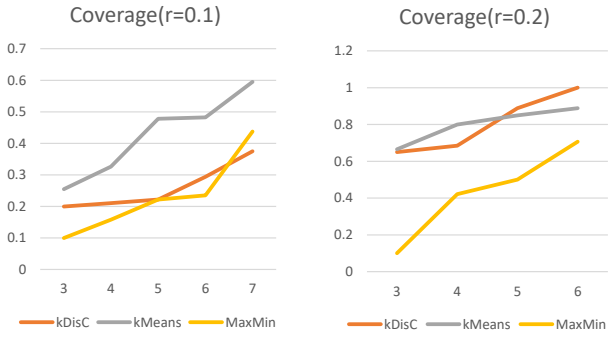

図 10: カバー率の計算結果（Carrier）

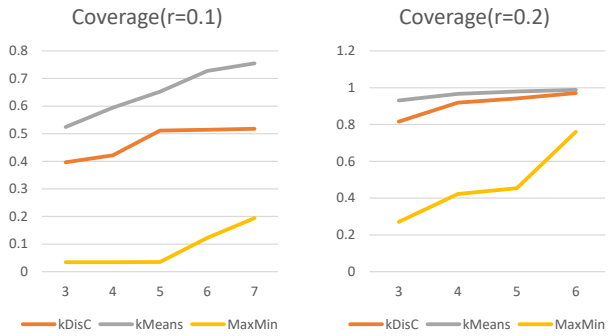

図 11: カバー率の計算結果（Operator）

(r, k)-DisC（図 9）では，代表点同士の非類似度を大きくしつつ，類似するオブジェクトが多い点を残す．そのため散布図を見ると，点が密集しているところから抽出しつつ端の外れ値をカバーしていることが見て取れる．代表点の棒グラフを確認しても，判別が難しいほど類似した棒グラフは抽出されていないことが分かる．

網羅性と非冗長性の定量評価：さらに抽出された代表点がどれだけ他の点をカバーできているか，代表点同士がどれだけ離れているかを評価するために分割結果全体に対するカバー率 (coverage) と代表点同士の平均非類似度 (dissimilarity) を計算した．前項で使用した 3 種類の多様化技術（(r, k)-DisC，k-means，MAXMIN）で得られる代表点に対して，サイズ k と非類似度の閾値 r を変えながらカバー率と非類似度を計算し，各手法がどれだけ網羅性，非冗長性を満たしているかを比較する．ここでは，カバー率の指標として分割結果の全棒グラフに対して，代表点と類似する（代表点との非類似度が r 以下の）点の割合を使用した．

この計算は，アメリカのフライトデータ [1] の次元 Carrier, State，Month，およびバードストライクのデータ [18] の次元 State，Operator に行ったが得られた結果はほぼ同じのため，ここではフライトデータの Carrier とバードストライクのデータの Operator を取り上げて説明する．

カバー率の計算結果は図 10 のようになった．概して，k の値が大きくなるほど，r の値が大きくなるにつれカバー率が大きくなることが分かった．各手法ごとに比較すると，どちらの次元でも $r = 0.1$ のときには k-means が最も高い値を示している．$r = 0.2$ になると (r, k)-DisC と k-mean の値が同程度まで上昇した．MAXMIN は端の点を取りやすい傾向があるため，r の値がどちらのときでもカバー率はあまり高くならなかった．

平均非類似度の計算結果を，図 12 に示す．概して，k の値が大きくなるほど，r の値が大きくなるほど平均非類似度は小さくなることが分かった．各手法ごとの平均非類似度の値の大きさは，大きい順におよそ MAXMIN，(r, k)-DisC，k-means の順序となった．MAXMIN は端の点を取りやすいため代表点同士の非類似度は高く，k-means は密集部分から点を取るため代表点同士の非類似度は低くなった．(r, k)-DisC は密集部分と端の点のバランスを取りながら抽出するため，その中間の非類似度となった．

このようにカバー率は r, k の値の増加とともに増加し，平均非類似度は r,k の値の増加とともに減少することから，この両者のスコアはトレードオフの関係になる．そこで，両手法の総合的な評価のためにカバー率を適合率，平均非類似度を再現率と考えて両者の調和平均 (F 値) の計算を行った．計算式を以下に示す．

$$F = \frac{2 \cdot coverage \cdot dissimilarity}{coverage + dissimilarity} \tag{3}$$

F 値の計算結果は，図 13,14 となった．Operator の計算結果では，概して (r,k)-DisC が最もスコアが高くなっていることが確認できる．r と k の両者が非常に大きいときのみ MAXMIN の方がスコアが大きくなったが，これは r と k の両方の値が大きくなることで端の点を抽出するだけでも十分全体を網羅できるようになったためと考えられる．Carrier の計算結果では，$r = 0.1$ のときには k-means のスコアが最も高くなっていることが分かる．これは，Carrier の分割結果の分散が大きく $r = 0.1$ と $k = 3$ から $k = 6$ では十分に結果集合を網羅できる点が抽出できないためと考えられる．

このように，分割結果の分散が大きいケースでは r, k の選び方を工夫する必要があるが，全体としては r-to-Rk DisC が F 値が高いものが多く，網羅率と平均非類似度を総合すると (r, k)-DisC が最も両者を満たしているといえる．

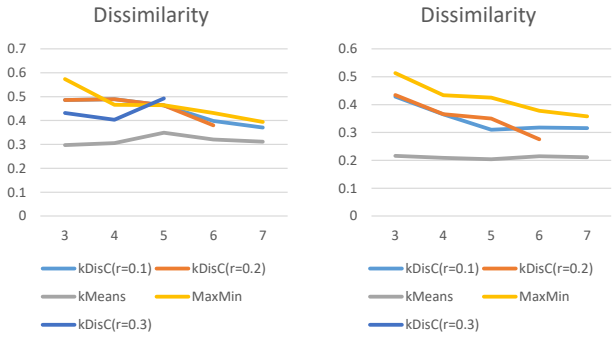

図 12: 平均非類似度の計算結果（左 Carrier，右 Operator）

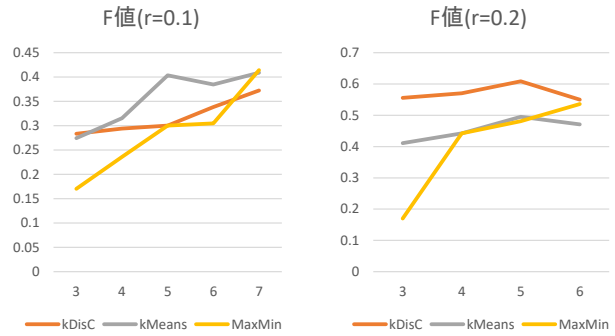

図 13: F 値の計算結果（Carrier）

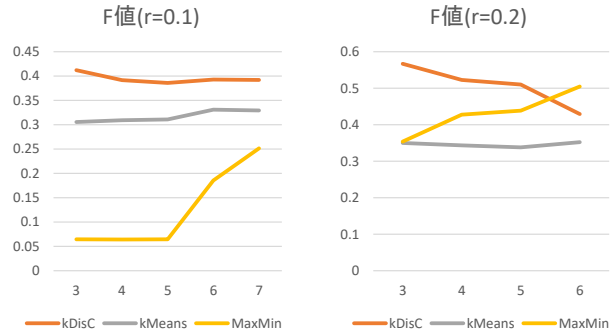

図 14: F 値の計算結果（Operator）

7 おわりに

本稿では，取りうる値の数が多い次元を持つデータセットを対象とした網羅性と非冗長性を両立した棒グラフ集合の要約方法と同手法を用いた棒グラフの探索システムを提案した．要約手法では，DisC Diversity [3] を利用して棒グラフ集合から網羅性と非冗長性を満たすように棒グラフを抽出した．また，上述の要約手法を実装した提案システムにより，取りうる値の数が多い次元でも対話的に棒グラフの探索を行うことが容易となった．

今後の課題としては，横軸のカテゴリ数が多いケースへの対応，分割結果のスコア化などが挙げられる．

謝　辞

本研究の一部は，科研費 16H01722 および 19K21530 による．

文　献

[1] Data Expo 09. ASA Statistics Computing and Graphics: `http://stat-computing.org/dataexpo/2009/the-data.html` (accessed: July 16, 2019).

[2] D. J.-L. Lee, H. Dev, H. Hu, H. Elmeleegy, and A. Parameswaran, "Avoiding drill-down fallacies with VisPilot: Assisted exploration of data subsets," in *Proc. Int'l Conf. Intelligent User Interfaces (IUI)*, pp. 186–196, 2019.

[3] M. Drosou and E. Pitoura, "DisC diversity: result diversification based on dissimilarity and coverage," *PVLDB*, vol. 6, no. 1, pp. 13–24, 2012.

[4] T. Siddiqui, J. Lee, A. Kim, E. Xue, X. Yu, S. Zou, L. Guo, C. Liu, C. Wang, K. Karahalios, and A. G. Parameswaran, "Fast-forwarding to desired visualizations with zenvisage," in *CIDR 2017, 8th Biennial Conference on Innovative Data Systems Research*, 2017.

[5] T. Siddiqui, A. Kim, J. Lee, K. Karahalios, and A. Parameswaran, "Effortless data exploration with zenvisage: An expressive and interactive visual analytics system," *Proc. VLDB Endow.*, vol. 10, pp. 457–468, Nov. 2016.

[6] T. Wu, D. Xin, Q. Mei, and J. Han, "Promotion analysis in multi-dimensional space," *Proc. VLDB Endow.*, vol. 2, pp. 109–120, Aug. 2009.

[7] S. Idreos, O. Papaemmanouil, and S. Chaudhuri, "Overview of data exploration techniques," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pp. 277–281, 2015.

[8] B. Tang, S. Han, M. L. Yiu, R. Ding, and D. Zhang, "Extracting top-k insights from multi-dimensional data," in *Proceedings of the 2017 ACM International Conference on Management of Data*, SIGMOD '17, pp. 1509–1524, ACM, 2017.

[9] M. Vartak, S. Rahman, S. Madden, A. Parameswaran, and N. Polyzotis, "SeeDB: Efficient data-driven visualization recommendations to support visual analytics," *Proc. VLDB Endow.*, vol. 8, no. 13, pp. 2182–2193, 2015.

[10] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager: Exploratory analysis via faceted browsing of visualization recommendations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, pp. 649–658, Jan 2016.

[11] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager 2: Augmenting visual analysis with partial view specifications," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 2648–2659, ACM, 2017.

[12] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Proceedings of the 1996 IEEE Symposium on Visual Languages*, VL '96, pp. 336–343, IEEE Computer Society, 1996.

[13] J. Gray, S. Chaudhuri, A. Bosworth, A. Layman, D. Reichart, M. Venkatrao, F. Pellow, and H. Pirahesh, "Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals," *Data Mining and Knowledge Discovery*, vol. 1, pp. 29–53, Mar 1997.

[14] S. Lyubka. Mongoose: A small and easy to user web server. `https://github.com/valenok/mongoose/` (accessed: December 23 2019).

[15] D3.js - Data-Driven Documents `https://d3js.org/` (accessed: December 23 2019).

[16] M. Drosou and E. Pitoura, "Search result diversification," *SIGMOD Rec.*, vol. 39, pp. 41–47, Sept. 2010.

[17] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 3rd ed., 2011.

[18] FAA Wildlife Strike Database `https://wildlife.faa.gov/home` (accessed: December 20 2019).