

PC作業におけるウィンドウ再配置に向けた ウィンドウグルーピング手法

山本 雄平¹ 石田 繁巳² 白石 陽²

概要：複数ウィンドウを用いる PC 作業では、ウィンドウの配置やサイズ調整などのウィンドウ操作を手動で行う必要があり、このウィンドウ操作はディスプレイ環境変化時に大きな手間となる。著者らは、PC 作業時の作業履歴を用いてウィンドウを作業ごとにグルーピングすることで、作業で併用されているウィンドウを推定し、ウィンドウ操作を支援する手法を提案している。先行研究では、事前に収集したデータを用いた教師あり学習により高い精度でウィンドウグルーピングできることを確認したが、学習した作業以外はグルーピングできないため、実用的ではない。そこで、本研究では学習データを必要としないクラスタリングによるグルーピング手法について検討する。本稿では、実際の PC 作業時に収集したデータを用いた評価実験の結果を報告する。

1. はじめに

業務や作業において情報端末は必須となっている。PC を用いた作業ではウェブブラウザやコミュニケーションツールなど、複数のアプリケーションを切り替えて用いる必要がある。このような場合、作業を効率よく行うために各アプリケーションの表示領域（以下、ウィンドウ）の配置やサイズ調整などのウィンドウ操作を手作業で行い、複数ウィンドウで構成される作業領域（以下、ワークスペース）を構築する。

PC 作業時にはディスプレイを増設することがある。実際、ディスプレイの増設によって効率的な作業が可能になるという報告 [1-3] がされており、ディスプレイ環境を変化させることは重要である。しかし、ディスプレイ環境を変化させる場合、手作業によるウィンドウ再配置が必要になる。現状の Windows OS では、ディスプレイとの接続を切った場合でも接続していたディスプレイ上におけるウィンドウ位置を記憶する機能が存在する。しかし、PC 作業者が新たなディスプレイに接続する場合や過去に行っていない作業を始めた場合には、手作業でのウィンドウ再配置が必要になる。

こうしたウィンドウの再配置やウィンドウの切り替えの

支援に注目し、ユーザの作業状態を推定する研究 [4-7] や複数ウィンドウの操作を支援する研究 [9, 11] が報告されている。ユーザの作業状態を推定する研究では、事前に大量のデータを学習させたモデルを使用しているため、実用に向けて大量データの収集、学習が必要になる。また、複数ウィンドウの操作を支援する研究では、ディスプレイ環境の変化を考慮していないため、ディスプレイ環境変化時のウィンドウ操作の増加は避けられない。

そこで本研究では、ディスプレイ環境変化時のスムーズな作業継続の実現を目的として、ウィンドウ再配置を支援するワークスペース構築手法を提案する。提案手法は、ウィンドウグルーピング機能とウィンドウ自動配置機能から構成される。著者らはこれまで、PC 作業時の作業履歴を用いてウィンドウをグルーピングすることで、PC 作業者が実施している作業と作業のために使用されているウィンドウを推定する手法を提案している [8]。事前に収集したデータを用いた教師あり学習により高い精度でウィンドウグルーピングできることを確認したが、学習した作業状況以外はウィンドウグルーピングによって推定できない。そのため、多様な作業状況を推定するためには、様々な作業状況を想定したデータ収集や学習が必要になる。

先行研究での課題の解決に向けて、本研究では事前の学習データを必要としないクラスタリングによるグルーピング手法を提案する。本稿では、複数被験者の PC 作業データを用いて、提案手法におけるグルーピング精度を評価した。

本稿の構成は以下の通りである。2 章では、ユーザの作

¹ 公立はこだて未来大学大学院システム情報科学研究科
Graduate School of Systems Information Science, Future University Hakodate

² 公立はこだて未来大学システム情報科学部
School of Systems Information Science, Future University Hakodate

業推定とウィンドウ操作量削減についての関連研究を述べ、3章では、提案システムの概要とクラスタリングによるウィンドウグルーピング手法について述べる。4章では、提案手法の評価実験について述べ、5章でまとめとする。

2. 関連研究

ユーザの作業推定に関する研究として、PCログから作業を推定する研究が報告されている [4-7]。Meyerら [4]は、キーボードとマウスの操作量やアプリケーションのカテゴリを特徴量として作業の切り替えのタイミングを推定している。Pilzerら [5]は、ウィンドウの切り替えやウィンドウが操作可能になっている状態の時間特徴などを使用して、関連する複数のウィンドウを推定している。Pellegriら [6]は、ウィンドウのタイトルやアクティブ化の時系列特徴を用いて、頻度の高い作業と頻度の低い作業の2つの段階において作業推定を実施している。Baoら [7]は、開発者向けのデータ収集・作業分析フレームワークを提案し、キーボードやマウス入力の他にアプリケーションのタイトルや開発環境、UI情報を用いて事前に定義した6つの作業に分類している。これらの手法は、いずれも教師あり学習で分類モデルを構築していることから、推定までに大量のデータ収集、学習が必要になる。

さらに、ウィンドウの切り替え時に次に使用するウィンドウを予測する研究 [9,10]が報告されている。これらの手法は、切り替え先の推定を目的としているため、複数のウィンドウで構築されたワークスペースを切り替える場合は、追加のウィンドウ操作が必要となる。

ワークスペース構築を支援する研究として、複数ウィンドウの一括操作を可能とするウィンドウドッキングシステム [11]やスクリーンショットからウィンドウ構築を実現する研究 [12]が報告されている。これらの手法は作業を再開するとき有効な方法ではあるが、ディスプレイ環境の変化が考慮されていないため、環境変化に合わせてシステムを有効にするための追加操作が必要になる。

3. 提案手法

本章では、本研究の提案システムと本稿で提案するクラスタリングによるウィンドウグルーピング手法について述べる。3.1節では、提案システムの全体像について、3.2節では、提案システムの構成機能である、ウィンドウグルーピング機能について、3.3節では、クラスタリングでのウィンドウグルーピングについて述べる。

3.1 提案システム

図1に提案システムの構成を示す。提案システムは、ウィンドウグルーピング機能とウィンドウ自動配置機能の2つの機能から構成される。ウィンドウグルーピング機能では、ユーザの作業履歴を収集し、収集した作業履歴に基

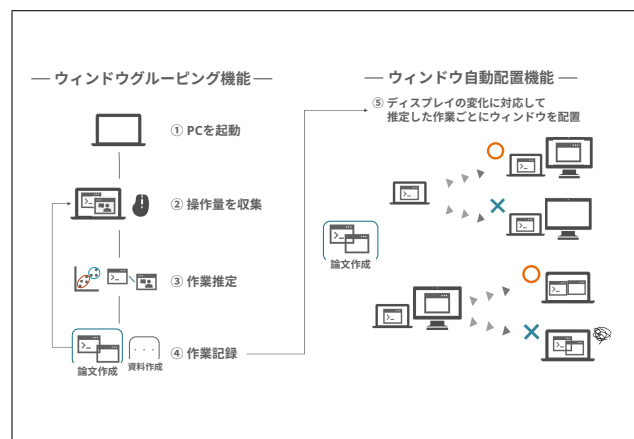


図1: 提案システムの構成

表1: 収集する特徴量

| 収集するデータ | 内容 |
|-----------|-------------------------|
| time | データ記録時の現在時刻 |
| title | 使用しているウィンドウのタイトル |
| act.count | 操作可能な状態になった回数 |
| act.time | 操作可能な状態になってからの継続時間 (秒) |
| key | ウィンドウに対して入力された回数 |
| click | ウィンドウに対してクリックされた回数 |
| scroll | ウィンドウに対してスクロールされた回数 |
| cursor | ウィンドウ内や切り替え時の移動量 (ピクセル) |

表2: 収集データ例

| time | title | act.count | act.time | key | click | scroll | cursor |
|----------|---------------|-----------|----------|-----|-------|--------|----------|
| 12:32:13 | Evernote | 2 | 13 | 26 | 2 | 0 | 721.668 |
| 12:32:47 | Github | 4 | 34 | 12 | 7 | 20 | 1049.051 |
| 12:33:33 | Visual Studio | 3 | 46 | 160 | 48 | 80 | 4050.588 |
| 12:33:45 | PDF | 2 | 12 | 2 | 3 | 34 | 521.363 |
| 12:33:57 | Evernote | 3 | 12 | 2 | 3 | 34 | 681.298 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

づいて作業ごとに使用するウィンドウをグルーピングする。ウィンドウ自動配置機能は、ウィンドウグルーピングの結果とウィンドウの配置やサイズデータを用いて作業ごとのウィンドウ配置を構築し、ディスプレイ環境の変化に合わせて作業ごとにウィンドウの自動配置を行う。

3.2 ウィンドウグルーピング機能

ウィンドウグルーピング機能は、PC作業で使用しているウィンドウを作業ごとに分類するための機能である。手順は以下の通りである。

- (1) PCを起動し、作業データを収集
- (2) 収集データを用いて作業を推定
- (3) 推定結果を記録

上記の(1)から(3)を繰り返し行うことで、ユーザが行う最新の作業を推定し、記録することが可能になる。

表1に収集する特徴量、表2に収集するデータ例を示す。収集するデータは、マウスやキーボードの入力操作に関する情報とウィンドウ情報である。マウス操作について

は、クリック回数 (click) やスクロール回数 (scroll) , カーソル移動量 (cursor) を記録している。キーボードについては、キー入力の回数 (key) を記録している。

ウィンドウ情報として、ウィンドウのタイトルと、クリックによってウィンドウが操作可能になる (有効化) 状態に関する特徴量を収集する。ウィンドウの有効化時間 (act.time) は対象のウィンドウが有効化してから別のウィンドウが有効化されるまでの継続時間である。ウィンドウの有効化回数 (act.count) は、作業開始からの対象のウィンドウの有効化された回数を記録している。これらのデータは、ウィンドウが切り替わるときにタイムスタンプ (time) と一緒に記録する。例えば、表 2 において「Evernote」が有効化されたときに、各データの収集を開始し、「Github」にウィンドウが切り替わった時、それまでに「Evernote」を対象として行われた入力操作に関する情報とウィンドウに関する情報を同時に記録する。

3.3 クラスタリングによるウィンドウグルーピング手法

クラスタリングは、入力データのパターンを分析し、教師データなしでクラスタに分類する手法である。先行研究 [8] では、事前の収集データで作業ごとにラベル付けを実施し、分類先を決定していた。提案手法では、ウィンドウごとのデータの特徴量データとして入力し、入力データに基づいてクラスタリングを行い、形成されたクラスタを作業として扱う。表 2 では、「Evernote」の特徴量データで 1 件、「Github」の特徴量データで 1 件のように、ウィンドウごとに入力データとする。

クラスタリングは階層クラスタリングと非階層クラスタリングの 2 つに分けることができる [13]。階層クラスタリングは、類似しているデータからクラスタの形成を開始し、距離の近いクラスタを結合させる。これを繰り返すことで新たなクラスタを形成する方法である。クラスタを結合させる基準となる距離の決定方法としてワード法や最短距離法などがある。非階層クラスタリングは、予め形成するクラスタ数を決定し、そのクラスタ数に収束するまで類似しているデータをまとめて、クラスタを形成する方法である。非階層クラスタリングの特徴として、階層クラスタリングよりクラスタ数を判断するための指標が多いことが挙げられる。非階層クラスタリングの手法として k-means 法や MeanShift 法などがある。

本研究では、非階層クラスタリングを適用する。理由として、提案手法ではウィンドウとまとめるための作業数を推定する必要がある。そのため、作業数 (クラスタ数) は入力データから自動で推定できることが望ましい。そのため、本研究ではクラスタリングに非階層クラスタリングを適用する。

表 3: 収集データの概要

| 被験者 | 作業時間 | 作業数 | 入力データ数 |
|-----|-----------|-----|--------|
| A | 1 時間 30 分 | 4 | 142 |
| | 1 時間 | 5 | 95 |
| B | 1 時間 | 2 | 99 |
| | 1 時間 | 3 | 164 |
| C | 1 時間 | 3 | 150 |
| | 1 時間 30 分 | 3 | 165 |
| D | 1 時間 20 分 | 4 | 60 |
| | 1 時間 | 2 | 285 |
| E | 2 時間 | 3 | 97 |
| | 30 分 | 2 | 85 |

4. 評価実験

本章では、クラスタリングによるウィンドウグルーピング手法の評価実験について述べる。4.1 節では、実験データ数や適用するクラスタリング手法などの実験概要について、4.2 節では、収集データの傾向の可視化に向けた主成分分析について述べる。4.3 節では、実験結果について、4.4 節では、クラスタ数の判定に使用する分析手法について、4.5 節では、実験結果と収集データの特徴を踏まえた考察について述べる。

4.1 実験概要

評価に使用するデータは、被験者 5 名から収集した内容の異なる PC 作業データ 10 件である。表 3 に被験者ごとの収集データの概要を示す。データ収集においては、作業環境や作業内容を制限せず、被験者が慣れた状況でのデータ収集を指示した。また、収集データに対して実施した作業とそれに用いたウィンドウについて、「論文作成→1, コード実装→2」のようにラベル付けを行うように指示し、正解データを作成した。評価手順として、クラスタ結果を可視化するために、収集データに対して主成分分析を実施して主成分を生成する。生成した主成分を用いてクラスタリングを実施し、分析結果と正解データを用いて、正解率を算出して評価を実施する。実験でのクラスタリング手法は、k-means を採用し、クラスタ数は正解データのラベル数を使用する。

4.2 主成分分析による収集データ分析

収集データの傾向を可視化するため、主成分分析を実施した。主成分分析とは、もとの変数に重みを付けて統合し、新たな変数 (主成分) を生成することである。生成された主成分は、元の変数の情報を保持していることから、変数を削減した状態でのクラスタリングが可能になる。表 4 に主成分分析の例として作業数 4 つの被験者 A データから

表 4: 主成分負荷量

| | title | act_count | act_time | key | click | scroll | cursor |
|-----|--------|-----------|----------|-------|-------|--------|--------|
| PC1 | -0.028 | 0.127 | 0.479 | 0.504 | 0.590 | 0.345 | 0.181 |
| PC2 | 0.435 | -0.648 | -0.351 | 0.275 | 0.156 | -0.112 | 0.392 |

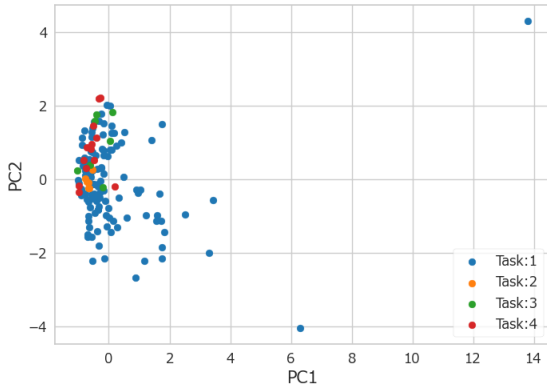


図 2: 主成分によるデータ分布

生成した主成分負荷量を示す。主成分負荷量とは、生成された主成分と元の変数との相関係数であり、数値の絶対値が大きいほど元の変数と強く相関があることを示している。今回、生成した主成分は2つである。表4を見ると、主成分1 (PC1) はキーボード入力数 (key) やクリック回数 (click) の数値が高く、主成分2 (PC2) では、有効化回数 (act_count) やタイトル (title) の相関係数が高いことがわかる。図2に主成分分析を実施し、可視化したグラフを示す。グラフのX軸、Y軸は生成された主成分から受けている影響を表しており、数値が大きいほどその主成分の影響を大きく受けていることを示している。図2を見ると、PC1では0から4にPC2では-2から2の間にデータが集中していることから、被験者AのPC作業データでは、作業 (Task) に関係なくウィンドウの操作傾向が類似していることがわかる。このことから、データの可視化によって、収集した特徴量では作業ごとの傾向を把握することは困難である。

4.3 評価結果

表5に収集データごとの正解率を示す。クラスタリングによる作業推定は、作業データによって正解率が大きく変化する結果となった。正解率が最も高いデータは被験者B、作業数2つのデータで精度は0.808、最も低いデータは被験者C、作業数4つのデータで0.100となった。正解率がなぜ変化しているのか、作業数によってクラスタリング結果がどのように変化するかを確認するため、以降は、作業数が2つ、3つ、4つの収集データ、計3つのデータについて結果を述べる。作業数が2つのデータは、正解率が最も高い被験者Aデータ、作業数が3つのデータは被験者Dデータ、作業数が4つのデータは、正解率が最も低い

表 5: 収集データごとの正解率

| 被験者 | 被験者 A | 被験者 B | 被験者 C | 被験者 D | 被験者 E |
|-----|------------------------|------------------------|-------------------------------------|------------------------|-----------|
| 正解率 | 0.577 (4) 0.305 (5) | 0.808 (2) 0.792 (3) | 0.313 (3) 0.533 (3) 0.100 (4) | 0.354 (2) 0.226 (3) | 0.682 (2) |

被験者Cデータをそれぞれ使用する。

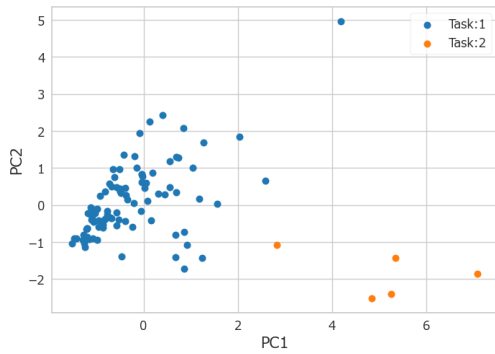
図3に作業数の異なる収集データのクラスタリング結果について示す。示している3つの収集データにおける作業 (Task) ごとのデータ数の分布は、作業数2のデータは「作業1:85」「作業2:14」である。作業数3のデータは「作業1:34」「作業2:4」「作業3:59」である。作業数4のデータは「作業1:4」「作業2:12」「作業3:41」「作業4:3」となっている。図3は主成分を軸としたグラフにおけるデータ分布とクラスタを示している。収集データ全体の傾向としてクラスタ内におけるデータの分散が大きく、クラスタ内におけるデータ間の距離が大きいことがわかる。

4.4 クラスタ数推定手法による分析

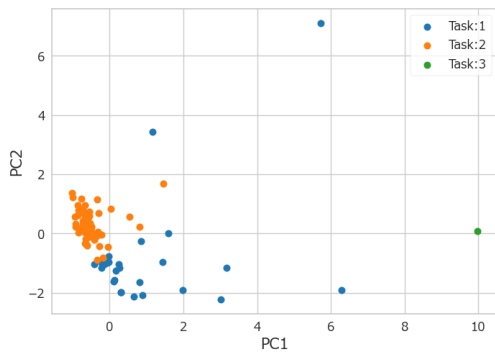
評価実験では、クラスタ数をラベル付けされた正解データに基づいて決定していた。今回使用したk-means法などの非階層クラスタリング手法では、入力データに対して適切なクラスタ数を決定する必要がある。また、本研究が提案するシステムでは正解ラベルを必要としない作業推定を目指しているため、ユーザが介入することなく、収集されたデータからクラスタ数 (作業数) を推定することが望ましい。

適切なクラスタ数を求めるための手法として、いくつか存在するがその中にエルボー法とシルエット分析がある。エルボー法とは、クラスタ数ごとにデータ間の距離の大きさに関する値を算出し、この値が大きく低下した部分を適切なクラスタ数として判定する手法である。シルエット分析は、クラスタ数を変化させてクラスタ内におけるデータ間距離やクラスタ間距離を用いて算出したシルエット係数を用いて適切なクラスタ数を決定する手法である。各分析手法の分析例を示すため、図4にエルボー法を収集データのうち2つのデータに絞って適用した結果を示す。また、図5にシルエット分析を行った結果を示す。

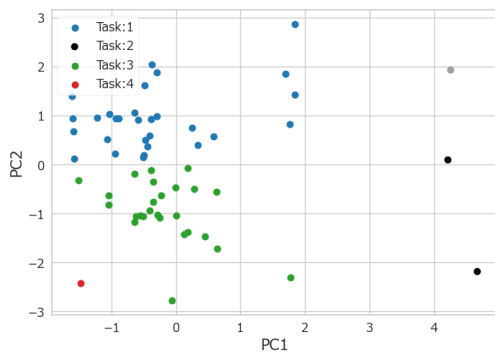
図4で示したグラフの横軸はクラスタ数を示している。縦軸の左側はデータはSSE (Sum of Squared Error) であり、クラスタの中心からクラスタに属する各データまで距離の総和を示している。この値が大きくなるほど、データが発散していることを示している。縦軸の右側の軸は、各クラスタ数の収束にかかる時間を示している。図4(a)では、クラスタ数k=4の時にSSEが大きく低下し、k=5以降はSSEに大きな変化がないため、k=4付近がクラスタ数であると判定することができる。図4(b)では、SSEの変化が緩やかでクラスタ数を一意に決定することはできな



(a) 被験者データ (作業数:2)



(b) 被験者データ (作業数:3)



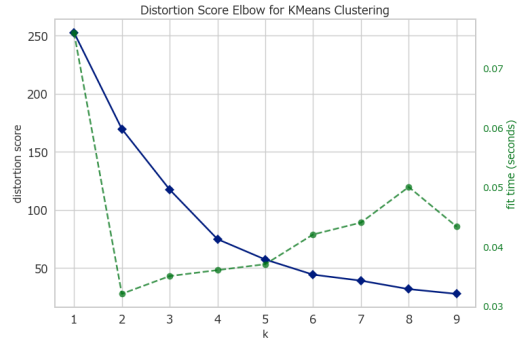
(c) 被験者データ (作業数:4)

図 3: クラスタリング結果

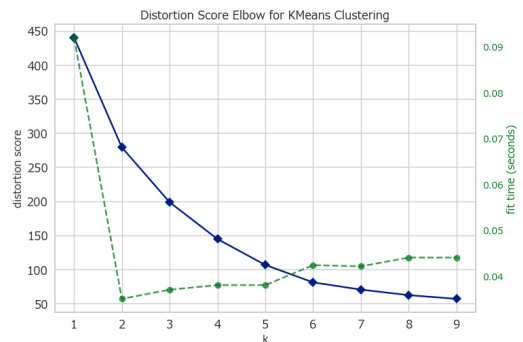
い。他の収集データにおいても、クラスタ数を一意に決めることができないデータの方が多くことがわかった。そのため、収集データに対して、エルボー法のみでクラスタ数を判断することはできないと考える。

図 5 で示したグラフの横軸はシルエット係数、縦軸はクラスタラベルを示している。今回はクラスタ数を 2 から 4 に変化させたときのクラスタとデータ分布を示している。図 5(a) から、正解のクラスタ数であるクラスタ数 $k=2$ では、他のクラスタ数と比較しても赤点線で引かれているシルエット係数の平均が最も大きい値となっている。しかし、青色のクラスタでのシルエット係数が平均のシルエット係

数に達しておらず、クラスタを形成しているデータが分散していること、他のクラスタとの距離が近いこと、データが他のクラスタと混ざっていることを示している。図 5(b) を見ると、クラスタ数 $k=3$ でも同様の状況になっていることが確認できる。このことから収集したデータでは、特徴量に差がでないためデータが偏ってしまい、適切なクラスタ数の判定は困難であることがわかった。



(a) 被験者 C データ (作業数:4)



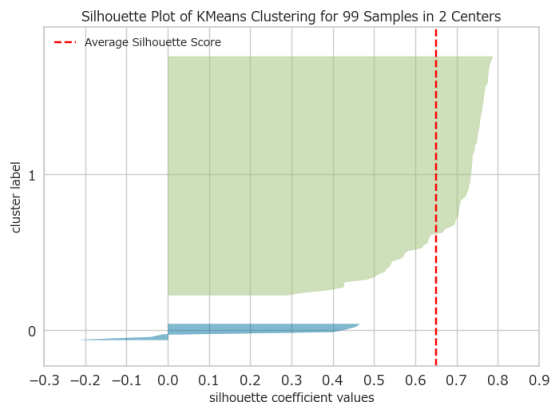
(b) 被験者 B データ (作業数:2)

図 4: エルボー法

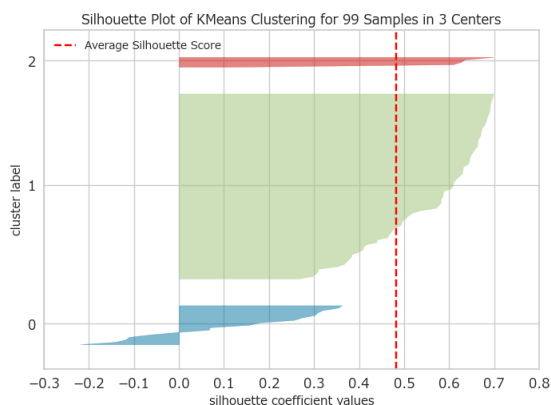
4.5 考察

収集データによって正解率が大きく変化した要因について述べる。1 つ目の要因として、作業ごとのデータ数に差があったことが考えられる。実際の PC 作業を想定しているため、作業ごとのデータ数を考慮せずに分析を実施した。そのため、収集データにおいて割合の多い作業データの正解率が高く、データ数の少ないデータについては正解率が低下したことが考えられる。このことから、作業推定において、各作業のデータ件数を変化させたときの作業推定への影響を調査する必要がある。

2 つ目の要因として、収集データの特徴量不足が考えられる。4.2 節に示したように、作業に関係なくウィンドウデータには偏りが見られ、ウィンドウデータの中に各作業に使用するウィンドウが混在している状況であった。特に操作量の多いウィンドウデータは、ウィンドウの特徴を最



(a) 被験者データ (作業数:2)



(b) 被験者データ (作業数:3)

図 5: シルエット分析

も表しているにも関わらず、類似しているウィンドウデータが多いため、外れ値となっている。また、個々のウィンドウデータを分析していることから、ウィンドウの併用を示すことができる特徴量が少ないことが作業推定に影響を与えたことが考えられる。そのため、複数ウィンドウの位置関係や複数のウィンドウが同時に表示されている時間などのウィンドウの併用を示す指標を追加する必要がある。

次にラベリングの粒度である。今回の実験では、作業と作業に使用したウィンドウに対してのラベル付けを被験者に指示したが、被験者によって粒度が異なっていた。1つや2つのウィンドウを使用した作業もあれば、5つや6つのウィンドウを使用している作業も存在する。ウィンドウによっては見聞きするだけの状態になっていることで、ウィンドウの操作量として現れない作業データが確認できた。PC上での出来事をユーザがどこまで作業として認識して欲しいか、どこまでシステムとして認識するべきか、基準を定めた上で作業推定を実施していく必要がある。

5. おわりに

本稿では、ディスプレイ環境の変化に対応したワークスペース構築を目的とし、構成機能であるウィンドウグルー

ピング手法の改善手法について提案、評価を行った。著者らの先行研究 [8] では、教師あり学習によって構築されたモデルを用いて、ウィンドウグルーピングを実施していたが、実環境の適用に向けて、大量データの収集・学習が必要になることが問題となっていた。本稿では、教師なし学習でグループを形成することができるクラスタリングを適用したウィンドウグルーピング手法を提案した。5名の被験者から収集した計10件のPC作業データを用いて、クラスタリングを実施した結果、正解率は最大で0.808、最小で0.100となり、作業内容によって精度が大きく変化する結果となった。収集データに対して主成分分析を行い、収集データをプロットした結果、データの偏りや、作業ごとの正解データが混在している状態であることがわかった。また、エルボー法やシルエット分析を適用して、クラスタ数推定を実施した結果、形成されたクラスタ内におけるデータの偏りや分散が大きく、他のクラスタとの距離が小さいことから適切なクラスタ数の判断が困難であると考えられる。今後の課題としてデータ収集時の操作量および、分析時の特徴量の追加、収集するデータ数や粒度の検討、クラスタ数の決定方法の検討が挙げられる。

参考文献

- [1] Hutchings, D. R., Smith, G., Meyers, B., Czerwinski, M. and Robertson, G.: Display Space Usage and Window Management Operation Comparisons between Single Monitor and Multiple Monitor Users, *Proceedings of the Working Conference on Advanced Visual Interfaces*, pp.32-39(2004).
- [2] Amir, G., Prusak, A., Reiss, T., Zabari, N. and Feitelson, D. G.: Use and Perceptions of Multi-Monitor Workstations: A Natural Experiment, *2021 IEEE/ACM 8th International Workshop on Software Engineering Research and Industrial Practice (SER&IP)*, pp.29-36 (2021).
- [3] 柴田博仁: 大画面ディスプレイ・多画面ディスプレイの導入による業務効率化の測定, *情報処理学会論文誌*, Vol.50, No.3, pp.1204-1213 (2009).
- [4] Meyer, A. N., Satterfield, C., Züger, M., Kevic, K., Murphy, G. C., Zimmermann, T. and Fritz, T.: Detecting Developers' Task Switches and Types, *IEEE Transactions on Software Engineering*, Vol.48, No.1, pp.225-240 (2022).
- [5] Pilzer, J., Rosenast, R., Meyer, A. N., Huang, E. M. and Fritz, T.: Supporting Software Developers' Focused Work on Window-Based Desktops, *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp.1-13 (2020).
- [6] Pellegrin, F., Yücel, Z., Monden, A. and Leelaprute, P.: Task Estimation for Software Company Employees Based on Computer Interaction Logs, *Empirical Software Engineering*, Vol.26, No.5, pp.1-48 (2021)
- [7] L. Bao, Z. Xing, X. Xia, D. Lo, and A. E. Hassan, :Inference of Development Activities from Interaction with Uninstrumented Applications, *Empirical Software Engineering*, Vol. 23, No.3, pp.1313-1351 (2018).
- [8] 山本雄平, 石田繁巳, 白石陽: PC作業におけるユーザの操作履歴に基づいたウィンドウグルーピング手法の検討, *情報処理学会研究報告*, Vol.2021-MBL-101, No.24, pp.1-5

- (2021).
- [9] Yoshida, K., Ozono, T. and Shintani, T.: FoXspace: Manipulating Windows Based on the User's Work History, *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pp.698–703 (2016).
 - [10] 今田正太郎, 佐藤慶三, 中島誠: 切替ウィンドウ予測機構によるマルチタスク支援のための自動ウィンドウグループイング, 電気・情報関係学会九州支部連合大会講演論文集 2019, p.431 (2019).
 - [11] 柴田博仁, 大村賢吾: ワークスペースの構築を可能にするウィンドウシステムの提案と評価, 人工知能学会論文誌, Vol.26, No.1, pp.237–247 (2011).
 - [12] Hu, D. and Lee, S. W.: Scrapbook: Screenshot-Based Bookmarks for Effective Digital Resource Curation across Applications, *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pp.1–13 (2022).
 - [13] Jain, A.K., Murty, M.N. and Flynn, P.J.: Data Clustering: A Review, *ACM Computing Surveys*, Vol.31, No.3, pp. 264–323 (1999).