CPS Analysis: Self-contained validation of biomedical data clustering

Lixiang Zhang\textsuperscript{1,\textdagger}, Lin Lin\textsuperscript{1} and Jia Li\textsuperscript{1,\textdagger}

\textsuperscript{1}Department of Statistics, The Pennsylvania State University, University Park, 16802, USA.

\textdagger\,To whom correspondence should be addressed.

Abstract

**Motivation:** Cluster analysis is widely used to identify interesting subgroups in biomedical data. Since true class labels are unknown in the unsupervised setting, it is challenging to validate any cluster obtained computationally, an important problem barely addressed by the research community.

**Results:** We have developed a toolkit called Covering Point Set (CPS) analysis to quantify uncertainty at the levels of individual clusters and overall partitions. Functions have been developed to effectively visualize the inherent variation in any cluster for data of high dimension, and provide more comprehensive view on potentially interesting subgroups in the data. Applying to three usage scenarios for biomedical data, we demonstrate that CPS analysis is more effective for evaluating uncertainty of clusters comparing to state-of-the-art measurements. We also showcase how to use CPS analysis to select data generation technologies or visualization methods.

**Availability:** The method is implemented in an R package called OTclust, available on CRAN.

Contact: lzz46@psu.edu

Supplementary information: Supplementary information are available at Bioinformatics online.

1 Introduction

Clustering has become even more important in the era of big data because it is a major technique to discover underlying structures of highly complex data. It is often used to explore data from new sources which researchers have little insight about. With the recent advances in high-dimensional and high-throughput biotechnology, clustering has played a big role in leading edge research in basic biomedical fields ranging from developmental biology to cancer studies. Clusters identified by data-driven approaches have been used to substantiate existing conjectures or to motivate new hypothesis. It is only natural that the clusters identified computationally can be validated effectively and those emerged due to artifacts of the algorithms or randomness of the data can be flagged to raise necessary caution. To address the dire need of assessing uncertainty in clustering results, we have developed CPS analysis, aiming at self-contained cluster validation. The new methods are implemented as OTclust, a user-friendly R package in CRAN. CPS analysis can be easily integrated into the existing pipelines of biomedical data clustering.

In the existing literature, a rather limited number of tools have been developed to address uncertainty in cluster analysis. Among those, Silhouette score (Rousseeuw, 1987) and Dunn index (Dunn, 1974) are well known and much used. The rationale for these uncertainty measures is however profoundly different from ours. We will compare with these methods through experiments. To quantify the distance (or similarity) between different clustering results, we will use measurements such as Adjusted Rand Index (ARI) (Rand, 1971) and Variation of Information index (VI index) (Meilă, 2007). More detailed descriptions of these measurements are available in (Supplementary).

The CPS analysis pipeline is illustrate in Figure 1. First, data are perturbed by adding random noises. This can be done on the original data or the processed data after feature extraction or dimension reduction. Although we did not observe considerable difference between the two choices in our experiments, our R package provides users the two options. Secondly, a collection of clustering results are obtained from the multiple versions of perturbed data. The OTclust package uses Kmeans as the default clustering method if the user opts out to specify a clustering method. However, CPS analysis, operating only on the collection of clustering partition results, does not restrict which clustering method is used or how the objects are mathematically represented. Hence, OTclust package also allows users to directly provide a collection of clustering results so as to carry out the downstream analysis. At last, the clusters in different clustering results are aligned via optimal transport (Section 2.1), based
on which the variation of individual clusters or the overall partition is revealed and the amount of uncertainty is quantified. The final output of CPS analysis includes for each cluster a so-called tightness measure and the covering point set (CPS) (Section 2.2). An overall tightness (valued between 0 and 1) is also computed for measuring the uncertainty of the overall partition. A higher value of tightness indicates higher stability or lower uncertainty. The CPS, a counterpart of the confidence interval of a numerical estimation, is a set of possible points for one cluster at a certain level of coverage. For visualization, we can show not only the CPS of each cluster but also a more detailed membership heat map to illustrate the extent of any point belonging to a cluster.

2 Methods

2.1 Cluster Alignment by Optimal Transport

After obtaining a collection of clustering results (aka. partitions), we first need to align them for further analysis. Specifically, we must establish a correspondence between the clusters in two results. The simplest case of the correspondence is a permutation. For instance, two partitions are identical but have used shuffled labels for the clusters.

Consider two partitions \( P^{(1)} \) and \( P^{(2)} \), each containing \( k_n \) clusters: \( P^{(n)} = \{ C^{(n)}_1, C^{(n)}_2, \ldots, C^{(n)}_{k_n} \} \). We capture the correspondence between clusters in the two partitions by a so-called clustering alignment matrix: \( W = (w_{i,j})_{i=1:k_n,j=1:k_n} \), where \( w_{i,j} \) is the coupling or matching weight between \( C^{n}(i) \) and \( C^{n}(j) \), a higher value meaning stronger match. If for each \( i \), there is a unique \( j \) such that \( w_{i,j} > 0 \) and \( w_{i,j'} = 0, j' \neq j \), and the same holds with \( i \) and \( j \) reversed, then \( W \) specifies a permutation (only possible if \( k_n = k_n \)). In general, we allow \( W \) to specify soft matching between the clusters in order to handle more complicated situations, e.g., \( k_n \neq k_n \), one cluster splitting into multiple clusters, etc. In order to decide \( W \) in a principled manner, we take the optimal transport (OT) approach as proposed in (Zhou et al., 2005; Li et al., 2019).

Suppose each cluster \( C^{(n)}_i \) is assigned with a significance weight \( q^{(n)}_i \), with \( \sum_{i=1}^{k_n} q^{(n)}_i = 1 \). Usually, \( q^{(n)}_i \) is the proportion of data points from \( C^{(n)}_i \). We solve \( W \) by the OT problem:

\[
D \left( P^{(1)}, P^{(2)} \right) = \min_{W} \sum_{i=1}^{k_1} \sum_{j=1}^{k_2} w_{i,j} \cdot d \left( C^{(1)}_i, C^{(2)}_j \right) \tag{1}
\]

s.t. \( \sum_{i=1}^{k_1} w_{i,j} = \bar{q}^{(1)}_j, \forall j = 1, \ldots, k_2 \)

\( \sum_{j=1}^{k_2} w_{i,j} = \bar{q}^{(2)}_i, \forall i = 1, \ldots, k_1 \)

\( w_{i,j} \geq 0, \forall i = 1, \ldots, k_1, j = 1, \ldots, k_2 \).

As in Li et al. (2019), the Jaccard distance, \( d(C^{(1)}_i, C^{(2)}_j) = 1 - \frac{\left| C^{(1)}_i \cap C^{(2)}_j \right|}{\left| C^{(1)}_i \cup C^{(2)}_j \right|} \), is adopted as the distance between clusters. The first two constraints on \( w_{i,j} \)’s ensure that the total influence of any cluster is determined by its proportion. The objective is to minimize the weighted sum of the matching costs between clusters. The minimized objective function \( D\left( P^{(1)}, P^{(2)} \right) \) is defined as the distance between the two partitions, often called the Wasserstein distance.

Consider \( P^{(2)} \) as the reference partition. After obtaining \( W \), we normalize its \( i \)th row and define \( \gamma_{i,j} = w_{i,j}/q^{(1)}_i \), which indicates the proportion of cluster \( C^{(1)}_i \) mapped to cluster \( C^{(2)}_j \). Let \( \Gamma^{(n)} = \{ (\gamma_{i,j})_{i=1:k_n,j=1:k_2}, \bar{q}^{(1)} \} \). Similarly, we can normalize \( W \) column-wise to obtain \( \bar{\Gamma}^{(n)} = \{ (\gamma_{i,j})_{i=1:k_2,j=1:k_n}, \bar{q}^{(2)} \} \). Based on \( \Gamma^{(1)} \) and \( \bar{\Gamma}^{(1)} \), four types of topological relationships between clusters are defined: “match,” “split,” “merge,” and “lack of correspondence.” For example, \( C^{(1)}_i \) and \( C^{(2)}_j \) match if \( \gamma_{i,j} \geq \zeta \) and \( \gamma_{j,i} \geq \zeta \), where \( \zeta \) is a relaxation threshold set between 0.5 and 1. If the “match” relationship holds between \( C^{(1)}_i \) and \( C^{(2)}_j \), they are considered to be the same cluster but possibly labeled differently in \( P^{(1)} \) and \( P^{(2)} \). Detailed definitions of these set relationships are in Li et al. (2019).

Suppose we have a reference partition \( P^{(r)} = \{ C^{(r)}_{i,1}, \ldots, C^{(r)}_{i,k_r} \} \), where \( k_r \) is the number of clusters. Let the proportion of points in \( C^{(r)}_{i,j} \) be \( y^{(r)}_{i,j}, j = 1, \ldots, k_r \). Similarly, let \( P^{(r)} = \{ C^{(r)}_{p,1}, \ldots, C^{(r)}_{p,k_p} \} \) and the proportion of points in cluster \( C^{(r)}_{p,j} \) be \( y^{(r)}_{p,j}, j = 1, \ldots, k_p \). We define \( \bar{p}^{(r)}_{i,j} \) is the posterior probability of the \( i \)th data point belonging to cluster \( C^{(r)}_{i,j} \). We then define the aligned cluster-posterior matrix based on \( P^{(r)} \) but “translated” to the cluster labels of the reference \( P^{(1)} \) by

\[
p^{(r \rightarrow 1)} = \{ \bar{p}^{(r \rightarrow 1)}_{i,j} \}_{i=1:n, j=1:k_1} \Rightarrow \{ \bar{p}^{(r \rightarrow 1)}_{i,j} \}_{i=1:k_1, j=1:k_2} \tag{2}
\]

Equipped with \( p^{(r \rightarrow 1)} \), \( \tau = 1, \ldots, \Lambda \), which are all based on consistent cluster labels of the reference \( P^{(r)} \), we can readily define a mean partition (an “average” of partitions), denoted by \( \bar{P} \), by specifying its cluster-posterior matrix: \( \bar{P} = \frac{1}{\Lambda} \sum_{\tau=1}^{\Lambda} p^{(r \rightarrow 1)} \). The mean partition will be used in our method for determining the number of clusters (Section 2.3).

Next, we present our proposed measures for cluster uncertainty.

2.2 Covering Point Set

In order to assess the uncertainty for partitions and clusters, several statistics are developed based on the cluster alignment results. Suppose for cluster \( k \) in the reference clustering result, we have a collection of matched clusters \( S_k, i = 1, \ldots, m_k \), each is a subset of the whole dataset \( \{ x_1, \ldots, x_n \} \). Then the covering point set (CPS) \( C_{S_k} \) of cluster \( k \) at a coverage level \( \alpha \) is defined as the smallest set such that at least \( 100(1-\alpha)\% \) of \( S_k \)'s subsets of \( S_k \), that is, to solve the optimization problem: \( \min_{S_k} |S_k| s.t. \sum_{i=1}^{m_k} \delta( c^{(1)}_i, c^{(2)}_i ) \geq m_k \cdot (1-\alpha) \). We use the Least Impact First Targeted-removal (LIFT) algorithm developed in Li et al. (2019). In our experiments, we set \( \alpha = 0.1 \).
Suppose we have \( n \) partitions in total, and the proportion of partitions that have a cluster “matched” with cluster \( k \) in the reference partition is \( p \) (e.g., some partitions can have “lack of correspondence” or other relationships for reference cluster \( k \)). For those partitions that contain a matched cluster to cluster \( k \), let the corresponding cluster \( k \) be sets \( S_i, i = 1, \ldots, m, \ m \leq n, \ p = m/n. \) At the coverage level \( \alpha \), let \( S_n \) be CPS of cluster \( k \). The tightness of cluster \( k \) is defined as

\[
R_k(S_n) = \frac{\sum_{i=1}^{m} |S_i|/|S_n|}{m}
\]

Also, we define the overall tightness of the whole partition, denoted by \( R_k(S_n) \), as the average over the tightness values of individual clusters. A larger value of tightness indicates more stable clustering.

We also develop a visualization function called the membership heat map, a graphical way to illustrate the stability of each individual point in terms of its association with any cluster. The membership heat map is a per-cluster plot. Take cluster 1 as an example. We calculate for each data point how many times it belongs to cluster 1 among all the clustering results. Note that here cluster 1 is in a non-reference clustering result means the cluster that is matched with cluster 1 in the reference result. We can then calculate the frequency of each data point belonging to cluster 1. In the membership heat map of any cluster, the frequency of each data point appearing in this cluster is coded by the hue of the color assigned to that point. On the other hand, whether a point belongs to that cluster in the reference clustering result is shown via markers of different shapes (e.g., triangles indicating being in the cluster and dots outside the cluster). For a cluster with high uncertainty, points in the cluster (e.g., shown by triangles) may have relatively low frequencies (cool hues) while points outside the cluster (e.g., shown by dots) may have high frequencies (warm hues). These point-wise frequencies can be used to gauge the stability of the membership of any point with respect to a particular cluster. The membership heat map graphically demonstrates these frequencies.

2.3 Determine the Number of Clusters

Given a dataset for clustering, we determine the number of clusters by the Most Frequent by Mean Partition (MF-MP) algorithm. First, we decide a reasonable range for the number of clusters. In the experiments we conducted, this range is from 2 to 20. For each number of clusters, we obtain the corresponding clustering result. Among these clustering results, we choose one as the “reference” partition. Specifically, we compute the average distance between each partition and the others using the Wasserstein distance defined in Eq. (1). The partition that yields the minimum average distance is taken as the reference. With this reference partition, we compute the mean partition as discussed in Section 2.1 and record the number of clusters generated in the mean partition. Then we repeat this process on the original dataset 100 times (here the difference in clustering is caused by nuances in the algorithm, e.g., initialization) and frequencies for each number of clusters are computed. Finally, the number of clusters with the highest frequency is chosen as the optimal number of clusters.

In the literature, the Silhouette score has also been used to select the number of clusters. Specifically, the most stable partition according to this score is also considered to have the right number of clusters. We could adopt a similar practice using the tightness measure. However, we caution against using uncertainty measures to select the number of clusters because these measures inherently favor a smaller number of clusters. Consider the extreme case of assuming only one cluster, uncertainty does not exist but the clustering result is not meaningful. By the same token, when comparing the stability of overall partitions, we recommend comparison between partitions with similar numbers of clusters. We can be relatively confident that the clustering result becomes more questionable if the stability reduces at the decrease of the number of clusters, but not if at the increase of that number. In Section 3, we compare MF-MP with several other popular methods for selecting the number of clusters.

2.4 Analysis Pipeline

CPS analysis can be incorporated seamlessly into any clustering analysis. Depending on whether we want to assess uncertainty of clusters or to select visualization methods, we may generate multiple clustering results by different schemes. Next, we elaborate on the details for each case.

Since CPS analysis operates only on a collection of clustering results but not the original data, a user has the option to start with a set of partitions on the data obtained in whatever way. Specification of the original data is not required. We call this the combinatorial mode of using CPS. This mode allows a highly general setting for conducting stability analysis. For instance, CPS can be used to analyze the sensitivity of an algorithm to initialization by simply acquiring the clustering results under different initialization parameters. This general applicability of CPS is unique comparing with other stability measures.

More often we expect CPS to be used as an extension of the usual clustering pipeline, which generates clusters with uncertainty assessment either numerically or graphically. The user chooses the baseline clustering algorithm. For instance, a popular approach is to first apply PCA to reduce the dimension of the data to a few dozens and then to apply K-means. Assessing clustering stability based on results obtained from perturbed versions of the data is a basic idea used in various methods (Von Luxburg et al., 2010). Here we generate perturbed data by adding noise point-wise to the original data. The noise is sampled from the Gaussian distribution, with mean zero and a chosen variance adjusted with the dataset. Usually we set the variance to be 10% of the average within-cluster variance. The noise is added independently to every dimension of the data. Each perturbed version of the original data will likely lead to a new clustering result. The user can choose to perturb the original data which will then be processed (e.g., through feature extraction or dimension reduction) or to perturb the processed data directly. Every randomly perturbed dataset will be clustered. In our experiments, we used 100 perturbed datasets. The clustering result of the original data is used as the reference partition, with which all the other partitions are aligned by optimal transport. The outcomes of CPS analysis include the tightness statistics for measuring the uncertainty of individual clusters or that of the overall partition, the membership heat map, and CPS plot for any requested cluster.

When analyzing data with given labels, such as true class labels, the first step is often to visualize the data, typically in 2-dimension. There are many visualization tools available, such as t-SNE (Maaten and Hinton, 2008), UMAP (McInnes et al., 2018) and PCA. CPS analysis can be used to select a visualization method that shows best separation of given classes for a dataset. Some steps in CPS analysis can be simplified. In particular, the true class labels will be used, bypassing clustering of every perturbed dataset and the alignment of different partitions. We still generate multiple versions of perturbed data, treating the 2-dimensional projected original data as the base. To generate the class labels for the perturbed data points, we apply the k-nearest neighbor rule with the original data taken as the training set. The overall tightness statistic can be used to select a visualization method. This approach to select visualization method aims at achieving best separation among the given classes. The user also has the option to emphasize particular classes and to select a method based on tightness of some classes or their membership heat maps.

3 Results and Discussion

We demonstrate the potential usages of CPS analysis from three perspectives. We first show that CPS analysis outperforms other state-of-the-art methods for evaluating uncertainty of clusters using a single-cell
dataset. Following the convention of the literature, “uncertainty measure” is used interchangeably with “stability measure” in the rest of the paper.

Then we show that with or without the true class labels, CPS analysis can help compare the effectiveness of capturing (or retaining) intrinsic clustering structure for different data generation technologies or different visualization methods.

3.1 Comparison to State-of-the-Art Measurements

To benchmark CPS analysis for cluster validation, we compare it with four other popular measurements of clustering stability: ARI (Rand, 1971), VI index (Mialon, 2007), Silhouette score (Rousseeuw, 1987) and Dunn index (Dunn, 1974) (Supplementary). Both ARI and VI index are “supervised” because their calculation requires the true labels or “reference” labels. In contrast, Silhouette score and Dunn index are “unsupervised”. However, only the Silhouette score but not the Dunn index can measure stability for individual clusters. Moreover, we can calculate the mean of Silhouette score of all data points to get an average score, called Ave Silhouette, which can be a measurement for overall partitions. In a nutshell, the Silhouette score is based on the ratio between the within-cluster Euclidean distances and the cross-cluster distances. In CPS analysis, the tightness score indicates cluster stability, which can be computed for individual clusters as well as for overall partitions in either supervised or unsupervised manner (Section 2.2). To select the number of clusters, we use the new method MF-MP (Section 2.3). We also use the membership heat map (Section 2.2) to effectively visualize the inherent variation in any cluster.

We illustrate cluster validation analysis by CPS analysis using Yan’s dataset (Yan et al., 2013). This dataset contains 124 labeled individual cells from human preimplantation embryos and human embryonic stem cells (HESCs), each cell having 3840 genes measured. It contains 7 cell clusters, namely, Oocyte + Zygote (4.8% of the total cells), 2-cell (4.8%), 4-cell (9.7%), 8-cell (16.1%), Morulae (12.9%), Late Blastocyst (24.2%) and HESC (27.4%), which correspond to different development stages of the embryo. This dataset is regarded as a gold standard for clustering assessment (Kiselev et al., 2017). We first perform t-SNE (Maaten and Hinton, 2008) for dimension reduction on the 124 single cells, and visualize them in the 2 latent dimensions, as shown in the left panel of Figure 2(a). The five statistics for assessing the stability of the overall clustering result shown in the right panel of (a) are box-plot for supervised (red) and unsupervised (blue) stability measures. The first three statistics (from left to right) are supervised measures based on the true labels shown in the left panel of (a), and the last three are unsupervised measures. (d) Stability assessment for each cluster by the Silhouette score and tightness for the clusters shown in the right panel of (a). (e) Membership heat map (upper panel) and CPS plot (lower panel) of cluster 1 and cluster 6. Take cluster 6 in the right panel of (a) as an example. The triangles represent data points in this cluster, while the dots represent points outside it. Colors in the membership heat map indicate the frequency of a point belonging to cluster 6 based on the multiple versions of the cluster obtained from perturbed data. Colors in the CPS plot indicate whether the point belongs to the covering point set of cluster 6.

Fig. 2. (a) t-SNE visualization for Yan’s data. Dots represent individual cells, and colors indicate cluster membership. Left panel shows the true labels, right panel is the labels obtained from Kmeans with 7 clusters. (b) Five statistics (standardized on the unit scale) for assessing the stability of the overall clustering result shown in the right panel of (a). (c) box-plot for supervised (red) and unsupervised (blue) stability measures. The first three statistics (from left to right) are supervised measures based on the true labels shown in the left panel of (a), and the last three are unsupervised measures. (d) Stability assessment for each cluster by the Silhouette score and tightness for the clusters shown in the right panel of (a). (e) Membership heat map (upper panel) and CPS plot (lower panel) of cluster 1 and cluster 6.
from two clusters is 24.32 (two points from cluster 4 and 5 respectively) and the maximum diameter is 75.04 achieved by cluster 3. To further compare the five statistics from the perspective of being able to produce consistent results, we examine their variability when the clustering results are slightly altered. In particular, we repeat the calculation based on 100 perturbed samples created by adding Gaussian noise with mean zero and variance equal to 10% of the average within-cluster variance of clusters obtained from the original data. Figure 2(c) shows the box-plots of the standarized stability measures. These measures are divided into two groups: supervised (red) versus unsupervised measures (blue). Since our tightness measure can be computed in both supervised and unsupervised setups, we computed the two versions, which are denoted as “Tightness 1” for supervised and “Tightness 2” for unsupervised. In both groups, the tightness measure has the smallest variation. For supervised measures, VI index has the highest variation, while for unsupervised measures, Dunn index has the highest variation. This observation remains similar when we impose different levels of noise on the data (Supplementary Table 2).

Since only the Silhouette score and the tightness measure can address stability/uncertainty of individual clusters, we plot the two measures for every cluster in Figure 2(d). Note that for cluster 1 (Oocyte+Zygote), 2 (2-cell) and 6 (Late blastocyst), the Silhouette scores are quite similar and of mid-range values while the tightness measures differ widely from low to high values. According to the tightness measure, cluster 6 is quite trustable, while cluster 1 and 2 are not. If we compare the true labels and the Kmeans result in the left and right panels of Figure 2(a), cluster 6 agrees reasonably well with the Late blastocyst group, while cluster 1 and 2 are each a very small portion broken off from a large real group. In this case, the tightness measure can help distinguish a real cluster from spurious ones, while the Silhouette score cannot. The limitations of the Silhouette score are discussed in (Almeida et al., 2011). One issue with the Silhouette score is that the value of one cluster is highly affected by the small clusters nearby (cluster 1 of size 3) (Supplementary Figure 6).

The fact that cluster 1 (Oocyte+Zygote) has high uncertainty while cluster 6 (Late blastocyst) has relatively low uncertainty is also reflected by the membership heat map and CPS plot in Figure 2(e). The membership heat map of cluster 1 (top left) shows that points in this cluster according to the original data (shown as triangles) actually have very low frequencies of being associated with it based on the perturbed data. The CPS plot of cluster 1 (bottom left) shows that the CPS of this cluster is three times as large as itself. In contrast, the membership heat map of cluster 6 (top right) shows that the points in cluster 6 have relatively high frequencies of being associated with it based on the perturbed data; and the CPS plot of it (bottom right) is relatively close to itself. Moreover, the membership heat map of cluster 6 shows a few points that are likely associated with this cluster although they belong to other clusters based on the original data (the yellow and cyan dots). Interestingly, these few points are indeed in cluster 6 by the true labels. The membership heat map and the CPS plot provide us a “dynamic” view on the clusters, that is, the potential range of a cluster. For instance, the membership heat map of cluster 1 indicates that the cluster may not really exist because of the overall low frequencies, while that of cluster 6 indicates that the real cluster may be larger. Such a dynamic view is not provided by the other methods which do not exploit multiple versions of a cluster via perturbed data. It is thus difficult for such methods, e.g., Silhouette score, to detect spurious clusters.

### 3.2 Select Data Generation Technologies

Now we demonstrate the second scenario of using CPS analysis. When different technologies are available for generating data on the same instances, one natural question is which technology is most favorable. More specifically, for instance, we want to know whether a simpler technology can produce data that capture the subgroup structure of the instances as effectively as a more sophisticated but expensive technology. There are two possible cases. In the first case, the true group labels of the instances are not available or the subgroups are yet to be discovered. We show that one criterion for making a choice is to examine the stability of the clusters using CPS analysis. This criterion allows us to decide quantitatively which technology produces more clear-cut clusters. In contrast, manual inspection of the clusters in 2-dimensional projection plots is subjective and hard to scale up. For high dimensional data, the projection on low dimensions can sometimes be misleading. In the second case, we have the ground truth class labels. Consequently, we can directly see how well the clusters obtained match the true classes. However, CPS analysis can provide a more comprehensive comparison between a cluster and a true class. For instance, if certain points in a true class are missed in a corresponding cluster, CPS analysis can yield insight about the extent of being missed in addition to a binary conclusion of being included in or excluded from a group.

We illustrate this usage of CPS analysis using the dataset obtained from Pollen’s study (Pollen et al., 2014). This dataset consists of 301 single cells obtained from 11 populations: CRL-2338 epithelial (7.3% of the total cells), CRL-2339 lymphoblastoid (5.6%), K562 myeloid (chronic leukemia) (14%), BJ fibroblast (from human foreskin) (12.3%), HL60 myeloid (acute leukemia) (17.9%), iP5 pluripotent (8%), Kera foreskin keratinocyte (13.3%), NPC neural progenitor cells (5%) and GW(16, 21, 21+3) fetal cortex at gestational week (16,21, 21+3 weeks) (8.6%, 2.7%, 5.3%). The cells are measured by both the low-coverage (Laitin et al., 2014) and high-coverage sequencing technologies (Shalek et al., 2013; Shapiro et al., 2013; Treutlein et al., 2014). The low-coverage technique sequenced single cells at low depths (~ 50, 000 reads per cell), while the high-coverage technique is at about 8.9 × 10⁶ reads per cell. To determine the optimal number of clusters, we apply the MF-MP method as we did with Yan’s data (Section 2.3). MF-MP decides that the number of clusters is 11, which is the same as the number of biological groups (Supplementary Table 3). The numbers of clusters suggested by other methods are shown in (Supplementary Figure 7). Their corresponding clustering results are provided in (Supplementary Figure 8), which shows that these clustering results are worse than the result obtained at 11 clusters.

Following the analysis in Pollen’s study (Pollen et al., 2014), for both the low-coverage and high-coverage datasets, we applied Kmeans with 11 clusters on the first 50 principal components. Clustering under the same setup is repeated on 100 perturbed datasets generated by adding Gaussian noise on the 50 principal components (the noise has mean zero and variance equal to 10% of the average within-cluster variance). The overall tightness of the clustering result is 0.673 for the high-coverage data and 0.622 for the low-coverage data. The result shows that the high-coverage data produce clusters that are slightly more stable than those based on the low-coverage data. Figure 3(a) shows the 2-dimensional t-SNE plots color-coded by the true class labels and the cluster labels given by the high-coverage and low-coverage sequencing methods. For clarity of comparison, we show the same projected data given by the high-coverage technology in all the three plots, but the cluster labels are obtained by different technologies. It is clear that neither sequencing technology can correctly capture cluster 1 (CRL-2338 epithelial) which contains 22 cells. In addition, both sequencing technologies fail to tell apart the three GW groups (cluster 9-11). On the other hand, in the low-coverage clustering result, the true cluster 5 (HL60 myeloid) is divided between two clusters, a mistake not occurring in the high-coverage clustering result.

Figure 3(a) shows that a major difference between the high-coverage and low-coverage data is the formation of cluster 5 (HL60 myeloid). Cluster 5 by the low-coverage technology misses a considerable portion of points in the true class. The tightness of this cluster is 0.857 based on the high-coverage data and 0.568 based on the low-coverage data. Figure 3(b) shows the membership heat map and Figure 3(c) presents
CPS analysis is applied to obtain the overall tightness of the classes. For the nearest neighbors can be used). Based on the multiple clustering results, the class label of a perturbed data point is set to be the class of its nearest neighbor in the original dataset (alternatively, the most frequent class of k nearest neighbors can be used). Based on the multiple clustering results, CPS analysis is applied to obtain the overall tightness of the classes. For the visualized data are perturbed to generate multiple clustering results. The two colors indicate whether a point is inside the covering point set of cluster 5. The different colors indicate the frequency of a point belonging to cluster 5 according to the multiple clustering results obtained from the perturbed versions of the data. (b) CPS plot of cluster 5 obtained from low-coverage sequencing data. The two colors indicate whether a point is inside the covering point set of cluster 5. (c) CPS plot of cluster 5 obtained from low-coverage sequencing data. The two colors indicate whether a point is inside the covering point set of cluster 5. (d) Visualization of the data by PCA with the colors indicating cluster membership. (e) Visualization of the data by UMAP with the colors indicating cluster membership.

3.3 Select Visualization Methods

Finally, we demonstrate the third usage scenario of CPS analysis—it helps us choose the most effective visualization methods for high-dimensional data without manual inspection of plots. In many biomedical applications, scientists use data visualization for interpretation and exploration purposes. There are many popular visualization methods such as t-SNE (Maaten and Hinton, 2008), UMAP (McInnes et al., 2018) (Supplementary) and PCA. The methods often yield very different low-dimensional plots. Suppose the true class labels are given, one can look at the various plots to decide which one provides the best separation among the classes. However, this manual process can be tedious and it can be difficult to manually rank how well each plot allows us to distinguish the classes. We thus propose the following scheme to automatically choose the best visualization method for any particular dataset.

Our principle is that the best visualization method should show the most clear-cut separation of the true classes. Specifically, each low-dimensional visualized data are perturbed to generate multiple clustering results. The class label of a perturbed data point is set to be the class of its nearest neighbor in the original dataset (alternatively, the most frequent class of k nearest neighbors can be used). Based on the multiple clustering results, CPS analysis is applied to obtain the overall tightness of the classes. For the dataset of Pollen’s study, the overall tightness for the visualized data by t-SNE, PCA and UMAP is 0.770, 0.596, and 0.678 respectively. Thus, t-SNE is the most favorable visualization for this dataset, while PCA is the least. The three 2-dimensional plots are shown in Figure 3(a), (d), (e). For UMAP, although some classes are well separated, some classes are seriously mixed. To balance a clear display of all the classes, t-SNE appears to be the best. More detailed examination of the stability of the individual classes as shown in the visualization plots is provided in (Supplementary Figure 10-11), which support the conclusion that t-SNE is the best visualization method for this dataset. We remark that if the true class labels are not given, they can be replaced by cluster labels generated by a certain algorithm on a higher dimensional projected data or the original data. However, caution has to be taken regarding the meaningfulness of these clusters.

4 Conclusion

In this study, we proposed a methodology for evaluating the clustering stability computationally. It advances existing methods from multiple perspectives, and it can be easily integrated into the existing pipelines of cluster analysis for biomedical data. Using three usage scenarios, we demonstrated that CPS analysis is effective for evaluating uncertainty of clusters and clusters with low uncertainty or high stability are more likely to correspond to real classes. It can provide more comprehensive view on potentially interesting subgroups in the data by the membership heat map and CPS plot. Finally, we illustrate the potential usage of CPS analysis for selecting data generation technologies or visualization methods.

Funding

J. Li’s research is supported by the National Science Foundation under Grant DMS-1521092.
References


Supplementary Tables and Figures
Supplementary Figure 1. Determine the number of clusters for Yan’s data using four selected methods. (a) Elbow method. The optimal number of clusters is chosen as $K = 4$, as after which the distortion starts decreasing in an almost linear fashion. (b) Silhouette method. It also determines $K = 4$, as it results in the highest average silhouette score. (c) Gap statistic method. $K = 11$ is the selected optimal number of clusters because it is the first local maximum of the Gap statistic. (d) NbClust R package. NbClust automatically generates the frequency plot among a variety of $K$ values. $K = 4$ is chosen by NbClust as it has the largest frequency (optimal number of clusters equals 0 means some indices fail to provide the optimal number).
<table>
<thead>
<tr>
<th>Number of Clusters</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>4</td>
<td>11</td>
<td>12</td>
<td>27</td>
<td>16</td>
<td>13</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**Supplementary Table 1.** Determine the optimal number of clusters by our MF-MP method for Yan’s data. The final chosen number of clusters is 7, as it has the largest frequency. This chosen number matches with the true number of clusters.

**Supplementary Figure 2.** The clustering result for Yan’s data while using $K = 4$ as the number of clusters. (a) The true cluster information (left) and clustering result obtained from Kmeans (right), where this clustering result falsely merges two big clusters into one. (b) Stability measurements for the clustering result shown in (a), and almost all measures improve a lot compared to the result of using 7 as the number of clusters (Main paper Figure 2(b)). The clustering results with smaller number of clusters tend to have higher values of stability, such as silhouette score and tightness. Even though the result is not correct with respect to the truth, it becomes a more stable result. The correctness and the stability of the clustering result are indeed two different concepts. It is not advised to use measurements like silhouette score as the standard of selecting the number of clusters. (c) Box-plot for stability measures based on 100 perturbed samples. Tightness is still the most stable one, which has smallest variability. (d) Stability assessment for each cluster by the Silhouette score and tightness, both of which show similar pattern (information) in this example with $K = 4$. 
Supplementary Figure 3. The clustering result for Yan’s data while using $K = 5$ as the number of clusters. (a) The true cluster information (left) and clustering result obtained from Kmeans (right), where same as Supplementary Figure 2 this clustering result falsely merges two big clusters into one. (b) Stability measurements for the clustering result shown in (a), and almost all measures are still higher than the result of using 7 as the number of clusters (Main paper Figure 2(b)). The clustering results with smaller number of clusters tend to have higher values of stability, such as silhouette score and tightness. Even though the result is not correct with respect to the truth, it becomes a more stable result. (c) Box-plot for stability measures based on 100 perturbed samples. Tightness is still the most stable one, which has smallest variability. (d) Stability assessment for each cluster by the Silhouette score and tightness, both of which show similar pattern (information) in this example with $K = 5$. 

<table>
<thead>
<tr>
<th></th>
<th>ARI</th>
<th>VI Index</th>
<th>Ave Silhouette</th>
<th>Dunn Index</th>
<th>Overall Tightness</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.547</td>
<td>0.751</td>
<td>0.686</td>
<td>0.189</td>
<td>0.878</td>
</tr>
</tbody>
</table>
Supplementary Figure 4. The clustering result for Yan’s data while using \( K = 6 \) as the number of clusters. (a) The true cluster information (left) and clustering result obtained from Kmeans (right), where this clustering result is able to separate the two big clusters merged in the previous results of Supplementary Figure 2 and Supplementary Figure 3. However, it falsely separates clusters 2 from cluster 1. (b) Stability measurements for the clustering result shown in (a). Measures except tightness are close to the result of using 7 as the number of clusters (Main paper Figure 2(b)), while tightness becomes much lower. The reason is shown in (d). (c) Box-plot for stability measures based on 100 perturbed samples. Tightness is still the most stable one, which has smallest variability. (d) Stability assessment for each cluster by the Silhouette score and tightness. Different with silhouette score, the tightness for cluster 1, 2 and 4 are much lower. From (a), the cluster 2 should be merged into cluster 1 and cluster 4 should not exist. In this case, the tightness measure can help distinguish fake clusters, while the Silhouette score cannot.
Supplementary Figure 5. The clustering result for Yan’s data while using $K = 11$ as the number of clusters. (a) The true cluster information (left) and clustering result obtained from Kmeans (right). The cluster 5 in clustering result is a mixture of two real clusters, and many fake small clusters show up. (b) Stability measurements for the clustering result shown in (a). Most measures are very close to the result of using 7 as the number of clusters (Main paper Figure 2(b)), some measures are even higher. (c) Box-plot for stability measures based on 100 perturbed samples. Tightness is still the most stable one, which has smallest variability. (d) Stability assessment for each cluster by the Silhouette score and tightness. Different with silhouette score, the tightness for cluster 4 and 7 are much lower. From (a), they are both fake small clusters.

<table>
<thead>
<tr>
<th>Variance of Noise</th>
<th>ARI</th>
<th>VI Index</th>
<th>Ave Silhouette</th>
<th>Dunn Index</th>
<th>Overall Tightness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1%</td>
<td>0.0119</td>
<td>0.0244</td>
<td>0.0009</td>
<td>0.0007</td>
<td>0.0087</td>
</tr>
<tr>
<td>2%</td>
<td>0.0111</td>
<td>0.0254</td>
<td>0.0006</td>
<td>0.0005</td>
<td>0.0097</td>
</tr>
<tr>
<td>5%</td>
<td>0.0088</td>
<td>0.0211</td>
<td>0.0010</td>
<td>0.0006</td>
<td>0.0129</td>
</tr>
<tr>
<td>10%</td>
<td>0.0107</td>
<td>0.0254</td>
<td>0.0007</td>
<td>0.0006</td>
<td>0.0114</td>
</tr>
<tr>
<td>15%</td>
<td>0.0114</td>
<td>0.0252</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0121</td>
</tr>
<tr>
<td>20%</td>
<td>0.0108</td>
<td>0.0216</td>
<td>0.0007</td>
<td>0.0007</td>
<td>0.0149</td>
</tr>
</tbody>
</table>

Supplementary Table 2. Comparing variances of 6 uncertainty measures (from the second to the seventh columns) at multiple levels of Gaussian noises used to perturb the original data. Each row corresponds to a particular variance of noise, which is set to be a certain percentage of the average within-cluster variance of the true clusters. In both groups, tightness has the smallest variation at all levels of Gaussian noises.
**Supplementary Figure 6.** The Silhouette score of one cluster (for instance, cluster 1 in both plots) is highly affected by small clusters nearby. The left plot shows the case when there is a small cluster (cluster 2) close to cluster 1. The right plot shows the case that cluster 2 is merged with cluster 3 while cluster 1 itself is not changed. In this simple 2-dimensional case, the standardized Silhouette score of cluster 1 will increase from 0.454 (in left plot) to 0.5974 (in right plot). So, Silhouette score of one cluster is highly affected by the small clusters nearby.

**Supplementary Table 3.** Determine the optimal number of clusters by MF-MP method for Pollen’s data. The optimal number is chosen as $K = 11$ as it results in largest frequency. This chosen number matches with the value of true number of clusters.
Supplementary Figure 7. Determine the number of clusters for Pollen’s data using four selected methods. (a) Elbow method. The optimal number of clusters is chosen as $K = 13$, as it the sharpest “elbow” point. (b) Silhouette method. It determines $K = 12$, as it results in the highest average silhouette score. (c) Gap statistic method. $K = 13$ is the selected optimal number of clusters because it is the first local maximum of the Gap statistic. (d) NbClust R package. NbClust automatically generates the frequency plot among a variety of $K$ values. $K = 2$ is chosen by NbClust as it has the largest frequency (optimal number of clusters equals 0 means some indices fail to provide the optimal number).
Supplementary Figure 8. The clustering result for Pollen’s data using Kmeans with $K = 12$ (top row) and $K = 13$ (bottom row). (a) Both clusters 2 and 5 are falsely separated for both high-seq and low-seq data. Such scenario does not exist in clustering high-seq data using $K = 11$ (Main paper Figure 3(a)). (b) Compared with the clustering results for $K = 11$, this over-estimated clustering result show more mistakes. Overall, the tightness for high-seq and low-seq data clustering using $K = 11$ are 0.673 and 0.622, respectively. These two numbers decrease to 0.641 and 0.598 when using $K = 12$. They further decrease to 0.602 and 0.582 when using $K = 13$. Hence, with respect to both correctness and stability, $K = 11$ is better than $K = 12$ and $K = 13$ as the number of clusters.

Supplementary Figure 9. The membership heat maps and CPS plots of cluster 1 and 5 identified from the high-coverage sequencing data in Pollen’s study.
Supplementary Figure 10. The membership heat maps and CPS plots of an easy-to-separate cluster based on the three visualization methods: t-SNE, PCA, UMAP.

Supplementary Figure 11. The membership heat maps and CPS plots of a hard-to-separate cluster based on the three visualization methods: t-SNE, PCA, UMAP.

Supplementary Figure 10 and 11 show detailed comparison of the three visualization methods: t-SNE, PCA, UMAP. Although UMAP seems to yield clear-cut clusters, the strong separation is deceiving for some clusters because they are mistakenly blended. For instance, by UMAP visualization, the membership heat map and CPS plot of cluster 8 (Figure 5) show that this cluster is clumped with points from another two clusters. The tightness of this cluster by UMAP is 0.514. PCA performs even worse for cluster 8, yielding a tightness of 0.306. In contrast, t-SNE yields a tightness of 0.794 for this cluster. Overall, according to the tightness measure, t-SNE yields the best visualization in terms of retaining the cluster structure of the data, while PCA performs the worst. We would thus recommend to use t-SNE visualization for this dataset.

Supplementary Concepts

Adjusted Rand Index (ARI): Given a set $S$ of $n$ elements, and two groupings or partitions of these elements (i.e., clustering results), the overlap between any pair of clusters in the two partitions can be summarized by a contingency table $(n_{i,j})_{i=1,...,k,j=1,...,l}$, where $n_{i,j}$ is the number of objects shared in cluster $i$ of the first partition and cluster $j$ of the second partition. Let $a_i$ be the sum of the $i$th row in the contingency table,
and \( b_j \) be the sum of the \( j \)th column in the contingency table. The Adjusted Rand Index is defined as:

\[
ARI = \frac{\sum_{ij} \binom{n_i}{2} - |\sum_i \binom{n_i}{2} \sum_j \binom{b_j}{2}|/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{n_i}{2} + \sum_j \binom{b_j}{2}] - |\sum_i \binom{n_i}{2} \sum_j \binom{b_j}{2}|/\binom{n}{2}}.
\]

**Variation of Information index (VI index):** Suppose we have two partitions \( \mathcal{X} \) and \( \mathcal{Y} \) of a set \( S \) (of size \( n \)). Specifically, \( \mathcal{X} \) contains \( k \) clusters \( X = \{X_1, X_2, ..., X_k\} \) and \( \mathcal{Y} \) contains \( l \) clusters \( Y = \{Y_1, Y_2, ..., Y_l\} \). First define the probability of an object belonging to cluster \( X_i \) by \( P_x(i) = |X_i|/n \), and define \( P_y(j) \) likewise. The entropy associated with clustering \( \mathcal{X} \) is \( H(\mathcal{X}) = -\sum_{i=1}^{k} P_x(i) \log P_x(i) \). Define \( P(i, j) = |X_i \cap Y_j|/n \) as the joint probability of an object belonging to both \( X_i \) and \( Y_j \). The mutual information between \( \mathcal{X} \) and \( \mathcal{Y} \) is \( I(\mathcal{X}, \mathcal{Y}) = \sum_{i=1}^{k} \sum_{j=1}^{l} P(i, j) \log \frac{P(i, j)}{P_x(i)P_y(j)} \). The VI index is defined as \( VI(\mathcal{X}, \mathcal{Y}) = [H(\mathcal{X}) - I(\mathcal{X}, \mathcal{Y})] + [H(\mathcal{Y}) - I(\mathcal{X}, \mathcal{Y})] \).

**Silhouette score:** Given one clustering result \( \mathcal{P} = \{C_1, C_2, ..., C_k\} \) of set \( S \) (of size \( n \)), the Silhouette score of one point \( i \) is \( s(i) = \frac{b(i) - a(i)}{\max\{b(i), a(i)\}} \), where \( a(i) \) is the average distance between point \( i \) and all the other points in the same cluster, \( b(i) \) is the smallest average distance between \( i \) and all the points in another cluster. The average \( s(i) \) over all the points of a cluster is the Silhouette score for that cluster, and the average \( s(i) \) over all the data points is the score for the whole dataset, denote as Ave Silhouette.

**Dunn index:** Given one clustering result \( \mathcal{P} = \{C_1, C_2, ..., C_k\} \) of set \( S \) (of size \( n \)), define the distance between two clusters \( C_i \) and \( C_j \), denoted by \( d_{i,j} \), by the smallest distance between a point in \( C_i \) and another in \( C_j \). Then the minimal distance between points in different clusters, denoted by \( d_{\text{min}} \), is the minimum of \( d_{i,j} \)'s. For each cluster \( C_i \), denote by \( d_i \) the largest distance between two points in the same cluster (i.e. the diameter of the cluster). Let \( d_{\text{max}} \) be the maximum of \( d_i \)'s. The Dunn index is defined as \( d_{\text{min}}/d_{\text{max}} \).

**T-distributed Stochastic Neighbor Embedding (t-SNE):** It is a nonlinear dimension reduction technique for embedding high-dimensional data for visualization in a low-dimensional space of two or three dimensions. The algorithm has two main stages. First, it constructs a probability distribution over pairs of high-dimensional objects, such that similar objects have a high probability of being picked. Second, it defines a similar probability distribution over the points in the low-dimensional map. Then it minimizes the distance between the two distributions with respect to the locations of the points in the map.

**Uniform Manifold Approximation and Projection (UMAP):** It is a nonlinear dimension reduction technique for visualization. Based on the three assumptions of Riemannian manifold, they model the manifold with a fuzzy topological structure. Then the embedding is found by searching for a low dimensional projection of the data that has the closest possible equivalent fuzzy topological structure.