Avoiding common pitfalls when clustering biological data

Tom Ronan, Zhijie Qi, Kristen M. Naegle*

Clustering is an unsupervised learning method, which groups data points based on similarity, and is used to reveal the underlying structure of data. This computational approach is essential to understanding and visualizing the complex data that are acquired in high-throughput multidimensional biological experiments. Clustering enables researchers to make biological inferences for further experiments. Although a powerful technique, inappropriate application can lead biological researchers to waste resources and time in experimental follow-up. We review common pitfalls identified from the published molecular biology literature and present methods to avoid them. Commonly encountered pitfalls relate to the high-dimensional nature of biological data from high-throughput experiments, the failure to consider more than one clustering method for a given problem, and the difficulty in determining whether clustering has produced meaningful results. We present concrete examples of problems and solutions (clustering results) in the form of toy problems and real biological data for these issues. We also discuss ensemble clustering as an easy-to-implement method that enables the exploration of multiple clustering solutions and improves robustness of clustering solutions. Increased awareness of common clustering pitfalls will help researchers avoid overinterpreting or misinterpreting the results and missing valuable insights when clustering biological data.

Introduction

Technological advances in recent decades have resulted in the ability to measure large numbers of molecules, typically across a smaller number of conditions. Systems-level measurements are mined for meaningful relationships between molecules and conditions. Clustering represents a common technique for mining large data sets. Clustering is the unsupervised partitioning of data into groups, such that items in each group are more similar to each other than they are to items in another group. The purpose of clustering analysis of biological data is to gain insight into the underlying structure in the complex data—to find important patterns within the data, to uncover relationships between molecules and conditions, and to use these discoveries to generate hypotheses and decide on further biological experimentation. The basics of clustering have been extensively reviewed (1–3). Clustering has led to various discoveries, including molecular subtypes of cancer (4–7), previously unknown protein interactions (8), similar temporal modules in receptor tyrosine kinase cascades (9), metabolic alterations in cancer (10), and protease substrate specificity (11).

Although clustering is useful, it harbors potential pitfalls when applied to biological data from high-throughput experiments. Many of these pitfalls have been analyzed and addressed in publications in the fields of computation, bioinformatics, and machine learning, yet the solutions to these problems are not commonly implemented in biomedical literature. The pitfalls encountered when clustering biological data derive primarily from (i) the high-dimensional nature of biological data from high-throughput experiments, (ii) the failure to consider the results from more than one clustering method, and (iii) the difficulty in determining whether clustering has produced meaningful results. Biological systems are complex, so there are likely to be many relevant interactions between different aspects of the system, as well as meaningless relationships due to random chance. Differentiating between a meaningful and a random clustering result can be accomplished by applying cluster validation methods, determining statistical and biological significance, accounting for noise, and evaluating multiple clustering solutions for each data set. The high-dimensional nature of biological data means that the underlying structure is difficult to visualize, that valid but conflicting clustering results may be found in different subsets of the dimensions, and that some common clustering algorithms and distance metrics fail in unexpected and hidden ways. To address these issues, clustering parameters and methods that are compatible with high-dimensional data must be chosen, the results must be validated and tested for statistical significance, and multiple different clustering methods should be applied as part of routine analysis. Our main goal is to dispel the belief that there exists a one-size-fits-all perfect method for clustering biological data.

Some solutions to address these pitfalls require awareness of the issue and the use of appropriate methods, whereas other solutions require substantial computational skills and resources to implement successfully. However, one method—ensemble clustering (that is, clustering data many ways while making some perturbation to the data or clustering parameters)—solves multiple pitfalls and can be implemented without extensive programming or computational resources. We mention the uses of ensemble clustering, as appropriate, and provide an overview of ensemble clustering at the end.

The Effect of High Dimensionality on Clustering Results

Systems-level measurements and high-throughput experiments are diverse in size of the data sets, number of dimensions, and types of experimental noise. Examples include measurements of the transcript abundance from thousands of genes across several conditions (such as in multiple cell lines or in response to drug treatments) (6, 12), measurements of changes in the abundance of hundreds of peptides over time after a stimulation (13), or measurements of changes in the abundance of hundreds of microRNAs...
across tissue samples (6). Because the dimensionality of the data in a clustering experiment depends on the objects and features selected during clustering, understanding how to determine dimensionality and its effects on clustering output are prerequisites for approaching a clustering problem. As a rule of thumb, data with more than 10 dimensions should be considered high dimensional and should be given special consideration.

**Determining dimensionality**
The dimensionality of a clustering problem is defined by the number of features that an object has, rather than by the number of objects clustered. However, the definition of object and feature in a given clustering problem depends on the hypothesis being tested and which part of the data is being clustered. For example, in the measurement of 14,546 genes across 89 cell lines, as found by Lu et al. (6), we can ask two questions with these data. First, what is the relationship between genes based on their changes across the 89 cell lines? This case represents a gene-centric clustering problem with 14,456 observations, with each observation having 89 dimensions (Fig. 1A). Second, what is the relationship between cells based on the changes in gene expression across all of the genes? This case represents a cell line clustering problem, obtained by transposing the original data matrix such that the matrix now has 89 observations, with each observation having 14,456 dimensions (Fig. 1B).

The same data can be clustered in different ways to discover different biological insights, and these different ways of clustering the data can have large differences in dimensionalities. The gene-centric clustering problem represents a good basis for clustering because the observations greatly exceed the dimensions. However, even that problem represents a high-dimensionality situation. The cell line clustering problem is even more challenging because the relatively small number of observations (89) compared with the large dimensionality (>14,000) could be dominated by noise in the expression data. Without careful handling of sparsity and feature masking, clustering will almost certainly deliver poor or misleading results. Reliable and meaningful clustering results are likely achievable only with careful dimensionality reduction or subspace clustering.

**Geometry and distance in high-dimensional data**
As dimensionality increases beyond two- and three-dimensional spaces, the effects of high dimensionality, referred to as the “curse of dimensionality,” come into play. These effects manifest in three key ways: (i) Geometry behaves unintuitively in higher dimensions; (ii) sparsity is common in high-dimensional data sets; and (iii) relevant features tend to become masked by increasing numbers of irrelevant features.

As dimensionality increases, familiar relationships of distance, volume, and probability distributions behave unintuitively (14). The method of determining distance between points in a clustering problem (the distance metric) influences the clustering results. With high-dimensional biological data, distance metrics fail to behave as expected based on our low-dimensional intuition. For example, as dimensionality increases beyond 16 dimensions, the nearest neighbor to a point can be the same distance away as the farthest neighbor to the point (for many common distance metrics) (15–17). Put another way, because it is likely that any two points are far apart in at least a few dimensions, all points approach being equally far apart (18). This means that for many types of distance functions in high-dimensional space, all points are effectively equidistant from one another (17, 19). As a result, some common distance metrics and the concept of “nearest neighbor” can be potentially meaningless in high-dimensional space (17, 20). Although data with more than 16 dimensions should be considered high dimensional (17), data with as few as 10 dimensions can also exhibit nonintuitive high-dimensional behavior (14).

Some algorithms that were developed for lower-dimensional spaces do not generalize well to high-dimensional spaces. Many centroid and density-based algorithms, such as k-means (centroid) and DBSCAN (density-based), rely on defining a nearest neighbor, which only works well in lower-dimensional spaces (21). Thus, k-means and DBSCAN (22) often fail to return useful results when used on high-dimensional data (15, 17). Furthermore, these algorithms will give no indication that they are not working as expected. Despite these problems, reliable and interpretable results with high-dimensional data can be achieved if ensembles of these clustering algorithms are used (23, 24).

Some algorithms are specifically designed to function with high-dimensional data. Hypergraph-based clustering methods draw on the field of graph theory, a method of representing pairwise relations between objects. Hypergraph-based clustering can be used to transform sparse data in high dimensions to a problem in graph partitioning theory, drawing on unique methods from that field, to produce accurate and informative clustering (15). Grid-based clustering is a density-based approach and can fail to give meaningful results on some high-dimensional data sets. However, Opti-Grid is a grid-based clustering approach that specifically addresses the problems of distance and noise that confound other similar algorithms.

---

**Fig. 1. Determining the dimensionality of a clustering problem.** (A and B) Representation of the mRNA clustering problem consisting of >14,000 mRNAs measured across 89 cell lines. Data are from Lu et al. (6). When the mRNAs are clustered, the mRNAs are the objects and each cell line represents a feature, resulting in an 89-dimensional problem (A). When attempting to classify normal and tumor cell lines using gene expression, the cells lines are the objects and each mRNA is a feature, resulting in a clustering problem with thousands of dimensions (B). (C) Effect of dimensionality on sparsity. (D) Effect of dimensionality on coverage of the data based on SD from the mean.
when applied to high-dimensional data sets (21). Some algorithms, such as NDFS, simultaneously solve problems with noise and find subgroups (25), which can improve the accuracy of clustering of high-dimensional data. Others, like OptiGrid, alternate rounds of grouping and dimensionality reduction to cluster high-dimensional data (26). When working with high-dimensional data, clustering algorithms that are suited for the dimensionality of the clustering problem should be used.

**Sparsity in high-dimensional data**

Sparsity is a problem with high-dimensional data because clusters are most clearly defined when there are many objects in each cluster, resulting in a robust representation of each cluster. Given a fixed amount of data, increasing dimensionality causes the data to become spread out in space. For example, given a set of data points in one dimension, data may be densely packed (Fig. 1C). As the dimensionality increases to two or three dimensions, each unit volume of the space becomes less and less populated. Extrapolating to an even more sparsely populated high-dimensional data set, it becomes increasingly difficult to ascertain whether a distant data point represents a noisy point far from one cluster or a new cluster that only has a few members and thus is difficult to identify. Effectively, random noise dominates clustering results based on sparse data.

Sparsity also affects our low-dimensional intuition for statistical rules of thumb. We typically use 3 standard deviations (SDs, ±3σ) to determine a reasonable threshold for statistical significance. In a one-dimensional Gaussian distribution (represented by a typical bell curve), 3 SDs cover 99.7% of the data. In two or three dimensions, when independently distributed, this coverage is reduced slightly (to 99.5 and 99.2%, respectively). In 10 dimensions, 3 SDs cover only 97.3%. By 100 dimensions, coverage is reduced to 76.3%; by 1000 dimensions, it is reduced to 6.7% (Fig. 1D) (16). This means that, in high-dimensional space, our rules of thumb for interpreting variance and what threshold should be considered statistically significant need to be reconsidered.

**Masking relationships in high-dimensional data**

With high-dimensional data, the signal can easily get lost in the noise. Biological noise and variation contribute to irrelevant features, which can mask important features, and ultimately influence cluster membership in the full-dimensional space (27). In the example of gene expression analysis across 89 cells lines, measuring tens of thousands of transcripts guarantees that transcripts with no relevance to the biology under study will also be measured. However, the noise that is present in those transcripts from biological or technical variation can dominate clustering results because clustering algorithms treat the noise as if they are true features in the data. The background noise resulting from cell-cell variation might swamp the most relevant features necessary to identify differences between cell lines, or background noise can even masquerade as significant differences between cell lines when those differences are due to random processes in the cell.

Strong relationships among any subset of features can mask other relationships. For example, metastatic cells and normal cells of a particular type may have similar gene expression profiles, so cell lines might tend to cluster by cell type (rather than by normal versus metastatic) when the entire gene expression data are used. However, when only expression data from a subset of genes are used, the metastatic cells may exhibit similar expression patterns and thus would be grouped apart from the normal cells.

To exemplify how a signal can be detected in a lower-dimensional data set and lost in a higher-dimensional data set, we turn to the study by Lu and colleagues (6) in which they measured the relative abundances of microRNA and mRNA molecules in 89 cell lines. Eighteen of 20 gastrointestinal cell lines clustered together with the microRNA data, which had 217 dimensions, but this relationship was lost when the mRNA data with ~14,000 dimensions were clustered. Understanding that the higher-dimensionality data can result in loss of the signal in the noise, we agree with the authors when they suggested that the loss of signal might result from “the large amount of noise and unrelated signals that are embedded in the high-dimensional mRNA data.” Fortunately, strategies are available for unmasking the hidden signals in biological data.

There are competing schools of thought for addressing the masking of relevant features by irrelevant features. If only a few features (dimensions) of the data are likely to contain the most relevant information, some advocate applying techniques that reduce the dimensionality. Dimensionality reduction involves the selection of only a subset of the features; the selection of which is based on a criterion such as predictive power or variance. When only a few features are expected to contain signal and the rest are expected to be noise, principal component analysis (PCA) is often used. PCA reduces high-dimensional data to fewer dimensions that capture the largest amount of covariance in the data. Another method of reducing dimensionality is to remove features with low values, as is often done in microarray analysis where transcripts below a threshold, or transcripts changing only a very small amount between conditions, are removed from the data set. Alternatively, if multiple independent signals exist in the data, selection of different subsets of features to cluster (28) may reveal different relationships among the data.

These three commonly applied methods for reducing dimensionality will produce clustering results, but each has limitations and, by definition, eliminates some features of the data. Dimensionality reduction techniques can degrade clustering results for high-dimensional data sets (29) when structure that is present only in some subset of the data is eliminated as a result of the dimensionality reduction. To illustrate this problem, we use a toy data example in which three clusters are readily apparent when the data are presented in two dimensions (Fig. 2A). A single projection onto any one dimension (determined by PCA, defined by the dotted line) (Fig. 2B) reduces the separability compared with the two-dimensional representation such that the identity of at least one cluster is lost. This “local feature relevance” suggests that dimensionality reduction techniques, such as PCA, may be inappropriate if different clusters lie in different subspaces, and that one should avoid the application of a single instance of feature selection (27) to avoid missing important structures within the data.

In contrast, subspace clustering is a method that searches different subsets of original features to find all valid clustering results, no matter which subset they are found in. Subspace clustering does not use graph theory or partitioning methods. Because it involves the selection of a subset of features, it also does not rely on the density of the data. In some data, objects will cluster only in subspaces of the full data space. For example, in the toy problem, when three different subspaces are chosen, the three clusters are readily obvious (Fig. 2C). Because this is a simple data set, the ability to see the clusters in two dimensions is also obvious, but note that at least two of the green cluster data points and one of the brown cluster data points touch the dividing line when analyzed in two dimensions (Fig. 2A). In contrast, the three clusters are clearly separated when the appropriate subspaces are selected (Fig. 2C). Subspace clustering must be carefully tailored to the high-dimensional data to produce valid results (30). As a computationally intensive method, subspace clustering can hit limitations due to a combinatorial explosion of potential subspaces in high dimensions (15). However, subspace clustering can reveal the multiple, complex relationships in biological data, because although information is lost in each subspace, multiple subspaces are considered; therefore, subspace clustering results are informed by information from all relevant dimensions.

The problematic effects of dimensionality reduction and the efficacy of subspace clustering can be seen on the expression data from Lu et al. (6).
The clustering results from the original study (clustering 89 cell lines using the mRNA data, ~14,000 dimensions) (Fig. 2D) were compared to the clustering results after using PCA to reduce the dimensionality to the 10 most relevant features (Fig. 2E) and after using subspace clustering to reduce the dimensionality to the 10 most informative features (Fig. 2F). As Lu et al. found, we do not see significant grouping of cell lines of gastrointestinal origin when we cluster the data in the full feature space (Fig. 2D), or if we reduce dimensionality using the first 10 principal components as features from PCA (Fig. 2E). However, a selection of one 10-dimensional subspace shows strong clustering for gastrointestinal cell lines (Fig. 2F)—almost as strong as the results that Lu and colleagues presented for much lower-dimensional microRNA data. This analysis suggests that, although the reduced dimensions of principal component space may have uncovered a structure that we do not understand, PCA was not informative for grouping cells based on their tissue of origin. However, there is a subspace (a subset of features from the original dimensions) for the mRNA data in which we can successfully group cells by their origins. This example illustrates how irrelevant features in the high-dimensional space masked the grouping of cells by origin, despite the data including expression measurements of genes that reflect tissue origin. The key was finding the right approach to cluster the data.

**Effects of Clustering Parameters on Clustering Results**

In addition to issues related to the high dimensionality of biological data, clustering parameters also affect the clustering result. Given the same data, varying a single parameter of clustering, such as the transformation, the distance metric, or the algorithm, can drastically alter the clustering result. Unfortunately, there is often no clear choice of best metric or best transformation to use on a particular type of biological data. Each choice can mask or reveal a different facet of the organization within the data. Therefore, in addition to applying different methods of clustering and different approaches to addressing dimensionality, it is essential to consider results from multiple parameters when evaluating clustering results for biological data.

**Transformations and distance metrics**

Data are often transformed as part of analysis and processing. For example, transcriptional microarray data are commonly log2-transformed. This transformation expands the information for genes with low expression variation across samples and simplifies the identification of genes with differential expression. Similarly, in proteomic data sets, data may be centered and scaled by autoscaling or Z-scoring (25) to make relative comparisons between signals for which magnitude cannot be directly compared. Although transformation can improve the ability to draw useful biological insight (31, 32), transformation also generates a new data set with altered relationships that may reveal or mask some underlying biological relationships in the data (Fig. 3, A and B).

Although transformation of data is routine as a part of preclustering analysis, because of the effects on density and distance in the data set, any transformations done on the data at any point should be explicitly considered as a clustering parameter during the clustering process. We found that, compared with using different distance metrics or clustering algorithms (32), transformations often had the greatest impact on a clustering result (Fig. 3B). In other cases, transformation has little impact on clustering results (Fig. 3, C and D).

The choice of a distance metric also greatly affects clustering results because different distance metrics accentuate different relationships within the data (Fig. 3B). Thus, to avoid missing information in the data,
different distance metrics and transformations should be applied as a routine part of clustering analysis.

**Clustering algorithms**

The choice of a clustering algorithm is based on several factors: (i) the underlying structure of the data, (ii) the dimensionality of the data, (iii) the number of relevant features for the biological questions being asked, and (iv) the noise and variance in the data. Each algorithm incorporates different assumptions about the data and can reveal different relationships among the data. (Fig. 4). The four primary classes of clustering algorithms are hierarchical, centroid, density, and statistical. The choice of clustering algorithm depends on the predicted structure of the data, and each algorithm class produces clusters with different properties. Hierarchical clustering is useful when data are expected to contain clusters within clusters, or the observations are expected to have a nested relationship. The Ward algorithm (Fig. 4, column 2) can produce different results depending on the threshold for similarity, highlighting hierarchical relationships. For example, at different thresholds, the green cluster (Fig. 4, row 1, column 2) results from the reference clustering result. With no transformation (upper panels), Euclidean and cosine distance do not change cluster identity, but with Manhattan distance, a new cluster C is merged into cluster B. With the log2 transformation (lower panels), the Euclidean and cosine distance do not change cluster identity, but with Manhattan distance, a new cluster A' is added, and cluster C' is merged into cluster B. The log2 transformation accentuates different aspects of the data to account for the underlying structure in the data (as exemplified by Fig. 4, top row). Therefore, clusters must be evaluated for biological relevance, stability, and cluster fitness. Understanding and accounting for noise and uncertainty in the data should be also considered when determining whether a clustering result is meaningful.

**Evaluating Clustering Results**

How can you tell when the clustering result of biological data is meaningful? Because of the complexity of biological systems, there are likely to be many valid clustering results, each revealing some aspect of underlying biological behavior. Unfortunately, there are likely to be many meaningless relationships simply due to random chance because the data are complex. Most clustering algorithms will find clusters, even if there is no true underlying structure in the data (as exemplified by Fig. 4, top row). Therefore, clusters must be evaluated for biological relevance, stability, and cluster fitness. Understanding and accounting for noise and uncertainty in the data should be also considered when determining whether a clustering result is meaningful.

**Cluster validation**

Validation metrics are a measure of clustering fitness, assessing if the result represents a well-defined structure within the data set, using concepts such as cluster compactness, connectedness, or separation, or combinations of these attributes. Much like distance metrics, each validation metric accentuates different aspects of the data to account for the
The compactness of each cluster can be measured by the root mean square standard deviation (RMSSTD) method (34), the r-squared (RS) method (35), or the Hubert’s T statistic (36). Connectedness within a cluster can be measured by k-nearest neighbor consistency (37) or Handl’s connectivity metric (38). Separateness is measured by the SD validity index (35), which compares average scattering and total separation between clusters. Some metrics combine measures of compactness and separation, such as the pseudo-F statistic (also known as the Calinski-Harabasz index) (39), the Dunn index (40), the Davies-Bouldin index (41), silhouette width (42), or the gap statistic (43). When there are different numbers of clusters in the clustering results, or predominantly different membership in each cluster, some metrics might fail. The Rand index (44) is appropriate for validating these complex clustering differences.

To illustrate the differences in validation metrics that measure compactness, connectedness, and separation, we applied different metrics to the clustering results from Fig. 4 (Table 2). For the toy data, DBSCAN generally performs better on the more complex structures, with some exceptions. DBSCAN correctly identifies no structure (Fig. 4, row 1), the half-moons (Fig. 4, row 4), and nested circles (Fig. 4, row 5). This is also reflected in the validation metric results for connectedness, which is a better measure than compactness for nonspherical data shapes. The lack of structure in row 1 is also suggested by the “zero” result to the RS measure of compactness and the “not available” result for separation from the SD validity index. For the spherical clusters, k-means and mixture models perform the best (Fig. 4, row 2), which is reflected in the best scores for validation metrics that measure both compactness and separation. No algorithm performs well on the parallel lines data (Table 2). Although DBSCAN appears to do better with the center of the clusters (Fig. 4, row 3), the errant results at the far left and right result in a poor score for validation metrics—on par with those algorithms that divide the horizontal parallel lines into vertical clusters.

The validation metric chosen should match the characteristics being tested. One approach is to use a metric that matches the selected algorithm. For example, when an algorithm optimizes connectedness, a metric that evaluates connectedness instead of the one that evaluates compactness should be used (Table 1). An alternate approach is to use a panel of validation metrics, each measuring a specific aspect of the data. For example, when creating an ensemble of distance metrics and algorithms, a panel of metrics that measure compactness, connectedness, and separation will eliminate inappropriate combinations of parameters. The worst pitfall is to not use any metric to assess the clustering outcome.

### Clustering stability

Because complex biological data can demonstrate multiple and independent valid clustering results, any single clustering result can be inconclusive. Whereas validation metrics provide an assessment of the quality of a clustering result with regard to specific aspects of the data, stability of the clustering result defines how robust the clustering result is to perturbation, that is, how many ways of assessing the data produce a similar clustering result. Stability analysis works under the premise that clustering results represent the underlying structure of the data and should be relatively invariant to perturbations in the analysis. Stability analysis can identify robust clustering results using a variety of perturbations, such as accounting for (i) noise (45-47); (ii) the effect of projections of high-dimensional data onto fewer dimensions (30); (iii) differences in algorithms, distance metrics, and data transformations (32); and (iv) the effect of selecting random starting positions in nondeterministic algorithms (48).

Stability analysis assumes that there is a single “true” structure within the data; however, high-dimensional biological data can have multiple true structures that reveal distinct biological insights. Therefore, it may be equally valid to assume that there are multiple structures within the data. Multiple clustering analysis methodology (MCAM) (32) is an approach that tests multiple clustering results. The approach in MCAM is based on the assumption that some perturbations to clustering parameters can uncover clustering results that reveal different biological insights. For example, some transformations uncovered shared binding partners of phosphorylated sites.
Accounting for noise and measurement uncertainty

Uncertainty in data means uncertainty in clustering results. However, many clustering algorithms do not account for noise or error when determining relationships between observations or when calculating distance. Although the molecular and cellular biology research community has adopted a set of rules that enable meaningful interpretation of differences in molecular measurements between pairs of conditions, this kind of standard has not been adopted for clustering complex, high-dimensional data. For example, when comparing means of measured data, we tend to require a minimum of triplicate measurements and the use of Student’s t test to determine statistically significant differences within biological data (49, 50). Surprisingly, similar standards and rules do not exist for identifying differential patterns or groupings from clustering results despite the data having the same measurements and biology as low-dimensionality data between pairs of conditions.

Several methods account for uncertainty in the data and properly propagate that uncertainty into clustering results. One ensemble approach is exemplified by Kerr and Churchill’s work (51) in which the researchers performed repeated clustering on multiple data sets generated by sampling gene expression from a statistical model that incorporates the variance of the measurements. The perturbation was resampling from reasonable, alternate representations of the original data based on noise. From this analysis, the variation in the clustering result due to the variability in the original data is determined, thus producing a range of clustering results representing the variability. An alternative approach is to use model-based clustering algorithms that account for the variance in the replicate measurements during the formation of the clusters (47, 52, 53).

Because model-based approaches are not commonly available to molecular biologists in standard software packages, we explored ways to account for noise with ensembles (45) without necessarily relying on a statistical model of the data. This exploration led us to two counterintuitive conclusions about the effects of noise. First, when data are well separated, robust clusters can still be found even in data with noisy distributions. Thus, clustering robustness cannot be predicted on the basis of the noise in the data. Second, the noise and variance in data are useful and contain information that can be revealed from the ensemble analysis. Specifically, we found that as signals propagated in time from a receptor tyrosine kinase to the mitogen-activated protein kinase (MAPK) pathway, there was high

Table 1. Validation metrics. Validation metrics used for testing the quality of a clustering result and the measures on which they are based.

<table>
<thead>
<tr>
<th>Metric</th>
<th>References</th>
<th>Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSSTD</td>
<td>(34)</td>
<td>Compactness</td>
</tr>
<tr>
<td>RS</td>
<td>(35)</td>
<td></td>
</tr>
<tr>
<td>Modified Hubert’s statistic</td>
<td>(36)</td>
<td></td>
</tr>
<tr>
<td>K-nearest neighbor consistency</td>
<td>(37)</td>
<td></td>
</tr>
<tr>
<td>Connectivity</td>
<td>(38)</td>
<td></td>
</tr>
<tr>
<td>Determinant ratio index</td>
<td>(62)</td>
<td></td>
</tr>
<tr>
<td>SD validity index</td>
<td>(35)</td>
<td>Separation</td>
</tr>
<tr>
<td>Pseudo-F statistic</td>
<td>(39)</td>
<td>Combination of compactness and separation</td>
</tr>
<tr>
<td>Dunn index</td>
<td>(40)</td>
<td></td>
</tr>
<tr>
<td>Silhouette width</td>
<td>(41)</td>
<td></td>
</tr>
<tr>
<td>Davies-Bouldin index</td>
<td>(41)</td>
<td></td>
</tr>
<tr>
<td>Gap statistic</td>
<td>(43)</td>
<td></td>
</tr>
<tr>
<td>Rand index</td>
<td>(44)</td>
<td>Similarity between solutions</td>
</tr>
</tbody>
</table>

Table 2. Validation metrics. Results of validation metrics [RMSSTD (34), RS (35), determinant ratio index (62), and SD validity index (35)] applied to the data from Fig. 4. The validation metrics indicate the type of best score (max or min). The bold values represent the best score for each structure and validation metric.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Validation metric</th>
<th>Compactness</th>
<th>Connectedness</th>
<th>Separation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm</td>
<td>Structure</td>
<td>RMMSTD</td>
<td>RS</td>
<td>Determinant ratio index</td>
</tr>
<tr>
<td>K-means</td>
<td>No structure</td>
<td>0.38</td>
<td>0.23</td>
<td>4.04</td>
</tr>
<tr>
<td>Ward</td>
<td>No structure</td>
<td>0.40</td>
<td>0.18</td>
<td>2.58</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>No structure</td>
<td>0.44</td>
<td>0.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Mixture models</td>
<td>No structure</td>
<td>0.38</td>
<td>0.23</td>
<td>4.09</td>
</tr>
<tr>
<td>K-means</td>
<td>Spherical clusters</td>
<td>0.78</td>
<td>0.81</td>
<td>249.08</td>
</tr>
<tr>
<td>Ward</td>
<td>Spherical clusters</td>
<td>1.18</td>
<td>0.56</td>
<td>26.43</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>Spherical clusters</td>
<td>0.58</td>
<td>0.86</td>
<td>1683.53</td>
</tr>
<tr>
<td>Mixture models</td>
<td>Spherical clusters</td>
<td>0.78</td>
<td>0.81</td>
<td>249.08</td>
</tr>
<tr>
<td>K-means</td>
<td>Long parallel clusters</td>
<td>0.89</td>
<td>0.27</td>
<td>2.77</td>
</tr>
<tr>
<td>Ward</td>
<td>Long parallel clusters</td>
<td>0.93</td>
<td>0.21</td>
<td>2.10</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>Long parallel clusters</td>
<td>0.84</td>
<td>0.21</td>
<td>22.36</td>
</tr>
<tr>
<td>Mixture models</td>
<td>Long parallel clusters</td>
<td>0.89</td>
<td>0.27</td>
<td>2.79</td>
</tr>
<tr>
<td>K-means</td>
<td>Half-moons</td>
<td>0.55</td>
<td>0.35</td>
<td>4.23</td>
</tr>
<tr>
<td>Ward</td>
<td>Half-moons</td>
<td>0.57</td>
<td>0.30</td>
<td>3.77</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>Half-moons</td>
<td>0.60</td>
<td>0.21</td>
<td>3.03</td>
</tr>
<tr>
<td>Mixture models</td>
<td>Half-moons</td>
<td>0.56</td>
<td>0.33</td>
<td>5.21</td>
</tr>
<tr>
<td>K-means</td>
<td>Nested circles</td>
<td>0.52</td>
<td>0.17</td>
<td>2.71</td>
</tr>
<tr>
<td>Ward</td>
<td>Nested circles</td>
<td>0.53</td>
<td>0.13</td>
<td>1.97</td>
</tr>
<tr>
<td>DBSCAN</td>
<td>Nested circles</td>
<td>0.57</td>
<td>0.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Mixture models</td>
<td>Nested circles</td>
<td>0.52</td>
<td>0.17</td>
<td>2.71</td>
</tr>
</tbody>
</table>
variability in an intermediate signaling state in the MAPK pathway. The effect of this noise in the clustering result reflected the relationship of this intermediate signal (a singly phosphorylated kinase) with both the upstream signal (the receptor) and the downstream signal (the doubly phosphorylated form of the kinase) and represented a meaningful biological relationship. Consequently, we recommend against prefiltering data to remove those with high variance; instead, noise should be addressed in the clustering analysis, even in the absence of replicates [as detailed in (45)].

Determining biological and statistical significance of clustering results

A primary challenge of clustering analysis is deriving biological insight. The most successful analyses often result from combining clustering with previous biological understanding of the system. However, unless the process of attaching biological meaning to the clusters is done with statistical analysis, we can overinterpret relationships that “make sense” to us. The two most common pitfalls related to this are (i) using anecdotal observations instead of a statistical test and (ii) failing to account for the increase in false positives when multiple statistical tests are performed.

Ultimately, to avoid the first pitfall, a researcher must determine the likelihood of having made a particular observation by random chance. This “null model” or “background model” can then be used to assess how likely the relationship uncovered by clustering is truly related to the biological information under consideration, as opposed to the likelihood of it occurring by random chance. We demonstrate this process with the microRNA clustering result from the analysis of the 89 cells lines (Fig. 3C) (6). To test the statistical hypothesis, we asked how likely it is that the 15 gastrointestinal cell lines would have occurred in a cluster of 20 cell lines by random chance alone. In contrast, the random chance of any two gastrointestinal cell lines appearing in a cluster of 20 cell lines is likely due to random chance alone ($P \leq 0.9$).

In many biological data sets, we are not simply testing one label in a single cluster but rather multiple labels across multiple clusters. A common example in cell biology and signaling research is to test for enrichment in any cluster of Gene Ontology terms (54) or biological pathway assignments from other sources. Testing for the enrichment of such biological properties or classifications across the clustering results represents multiple hypothesis testing and thus requires “multiple hypothesis correction.” For example, a $P$ value cutoff of 0.01 for a single tested hypothesis yields the prediction that false-positive results due to random chance occur rarely—only $1/1000$th of the time. However, as is common in biological data set analysis, we often test thousands of hypotheses (for example, asking if any one of 10,000 genes is differentially expressed). In this case, even with a $P$ value cutoff of 0.01, we would expect to find 100 false-positive results due to random chance. If we identify 105 statistically significant items, but we know 100 are false positives, we cannot separate which 5 are likely to be the true positives without multiple hypothesis correction.

When choosing a procedure for multiple hypothesis correction, reducing false positives may simultaneously increase false-negatives (the elimination of real positives), which can result in missing biological insight. To reduce the frequency of false-negatives, we recommend using the false discovery rate (FDR) correction procedure introduced by Benjamini and Hochberg (55), which is often applied in microarray analysis (56, 57), rather than the Bonferroni correction (58). Regardless of the type of correction chosen, when asking multiple questions, some form of multiple hypothesis correction is necessary to prevent overinterpreting the results.

**Ensemble Clustering: A Solution to Many Pitfalls**

Ensemble clustering refers to the act of clustering the data many times while making some perturbation—either to the data matrix or to clustering

<table>
<thead>
<tr>
<th>Perturbation</th>
<th>Reason behind perturbation</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>Identify the optimum number of clusters</td>
<td>(46, 60, 63)</td>
</tr>
<tr>
<td>Noise</td>
<td>Identify relationships within the data that are not affected by biological or experimental noise</td>
<td>(45, 47, 64)</td>
</tr>
<tr>
<td>Starting point (nondeterministic algorithms)</td>
<td>Identify those partitions that are independent of starting position or identity set of minima</td>
<td>(23, 48)</td>
</tr>
<tr>
<td>Projections into lower dimensions</td>
<td>Increase robustness to clustering noise resulting from high dimensionality</td>
<td>(30, 63, 65–67)</td>
</tr>
<tr>
<td>Subsampling</td>
<td>Identify subsets of the data that cluster consistently</td>
<td>(69–70)</td>
</tr>
<tr>
<td>Parameters of clustering</td>
<td>Identify unpredicted biological information</td>
<td>(32)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>In-depth review area</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific clustering algorithms, their trade-offs, and how they function</td>
<td>(2, 3, 71–73)</td>
</tr>
<tr>
<td>Analysis of the effects of different distance metrics on clustering gene expression data</td>
<td>(74, 75)</td>
</tr>
<tr>
<td>Practical and mathematical implications of high dimensionality on clustering</td>
<td>(15, 16)</td>
</tr>
<tr>
<td>A thorough review of validation metrics</td>
<td>(35, 38, 62, 76)</td>
</tr>
<tr>
<td>The most common multiple hypothesis correction procedures including Bonferroni correction and FDR correction.</td>
<td>(55, 58)</td>
</tr>
<tr>
<td>The effects of specific distances on data clustering for lower-dimensional spaces</td>
<td>(77)</td>
</tr>
<tr>
<td>The effects of specific distances on data clustering for high-dimensional spaces</td>
<td>(15, 78)</td>
</tr>
<tr>
<td>Ensembles of some algorithms incompatible with high-dimensional data can be useful on higher-dimensional data, even when a single clustering solution is uninformative.</td>
<td>(23, 24)</td>
</tr>
<tr>
<td>A more in-depth analysis of ensembles, including evaluating the results of multiple clustering runs and determining consensus</td>
<td>(60, 63)</td>
</tr>
</tbody>
</table>
parameters—and then accounting for all of the clustering results across the ensemble. The goal of ensemble clustering is to improve the quality and robustness of clustering results when compared to any single clustering result.

Why do ensembles improve quality and robustness? In short, it is because uncorrelated “noise” cancels across the clustering results. In ensemble clustering, noise in the clustering results occurs when there are strong biases due to the selected algorithms or because the data contains poorly clustering points. Fortunately, if the errors in each clustering result are not correlated, and the errors pertain to only a subset of the data, then the shared decisions made across the ensemble will dominate, resulting in convergence to the robust clustering result. A combination of diverse clustering results strengthens the underlying signal while filtering out the individual noise from each clustering result, enabling a more robust determination of the data structure than that acquired from a single clustering result obtained through analysis of the data without perturbation.

**Ensemble generation, finishing, and visualization**

The process of performing ensemble clustering involves selecting an appropriate perturbation (Table 3), collecting the clustering results based on the perturbation, and then either combining the individual clustering results into a single clustering result or exploring the ensemble of clustering results for information about the underlying structure of the data. To illustrate an ensemble of clustering solutions, we used random toy data and created an ensemble of clustering results using k-means clustering with an increasing number of clusters (k) (Fig. 5A).

There have been many techniques proposed to combine results of individual clustering results in the ensemble into one final clustering result. We refer to these methods as finishing techniques. Most finishing techniques use agreement across the ensemble to build a final clustering result. One method is to calculate the co-occurrence (or consensus) matrix. A co-occurrence matrix is an m × m matrix, where each entry, C_{ij}, represents the number of times object i clusters with object j across all of the clustering results in the ensemble.

We clustered the co-occurrence matrix using hierarchical clustering and Ward linkage and plotted the result as a heatmap (Fig. 5B). The clusters are formed on the basis of creating maximal in-group co-occurrence frequency and minimum co-occurrence with members outside the group. This representation reveals a wealth of detail about the relationships between data points and highlights data points that co-cluster robustly (that is, frequently) with each other across the ensemble.

Others have used majority voting, also based on the ideas of robustness (39, 60), as demonstrated in Fig. 5C. We subjected the co-occurrence matrix above to majority voting (Fig. 5C, left) and then plotted the ensemble majority-voting cluster assignments (Fig. 5C, right). Identifying the most robust clustering result is useful for generating hypotheses for further experimental testing because hypotheses can be ranked on the basis of strength of the co-occurrence (90% of the time compared to 10% of the time).

Another finishing technique to identify robust portions of the ensemble is to apply graph theory. For example, if we assumed that the co-occurrence matrix represented edge weights (a numerical value indicating the strength of connection) connecting the data points, we could traverse these weights to find maximally connected subgraphs and provide a different representation of robust clusters (61). With this concept of graph theory representations of ensemble clustering, we discovered that robustly clustered dynamic tyrosine phosphorylation data uncovered molecular-level interactions (8).

The finishing techniques mentioned above uniquely assign each observation to one cluster, thereby creating hard partitions within the data. However, a benefit of ensemble clustering is the ability to identify the probability of relationships (fuzzy partitioning), which can be applied to the entire data (probability is calculated for membership of any observation to any cluster) or a mixture of hard partitioning and probability-based assignment. An examination of the portion of the heatmap representation of the co-occurrence matrix containing the blue, gray, and orange clusters demonstrates fuzzy partitioning (Fig. 5D, left). The heatmap indicates that the gray cluster members share partial membership with the blue and orange clusters (Fig. 5D, right). Rather than considering the gray cluster as distinct, one can consider it to have a

---

**Fig. 5. Ensemble clustering overview.** Finishing techniques were applied to random toy data (see file S1 for analysis details). (A) Set of clustering results obtained using the k-means algorithm with various values of k (a k-sweep). (B) Hierarchically clustered (Ward linkage) co-occurrence matrix for the ensemble of results in (A). The heatmap represents the percentage of times any pair of data points co-clusters across the ensemble. (C) A majority vote analysis was applied (left panel) using a threshold of 50% on the co-occurrence matrix in (B). Six clusters (see dendrogram color groupings) result from the majority vote (right panel). (D) Application of fuzzy clustering to the ensemble. The left panel shows the details of the co-occurrence matrix for the blue, gray, and orange clusters, and the right shows the clustering assignments. The orange cluster provides an example of partially fuzzy clustering because it shares membership with the orange and dark blue clusters.
Ensembles for robust clustering results

Ensemble clustering can reveal unexpected results. When an algorithm with limited capabilities is combined into an ensemble, the clustering result of the ensemble can have new capabilities. For example, although a k-means algorithm can only identify spherical clusters, when used as part of an ensemble with majority voting, the ensemble can identify half-moons and spirals (59). This is possible because signals from relatively weak relationships in each clustering result are combined to improve the strength of the pairwise relationship between points in the ensemble clustering results. Ensemble clustering can assess the impact of perturbations to clustering parameters on the clustering results, revealing when transformations to the data have a larger impact on the clustering result than when the algorithm or distance metric does (32).

As an example of ensemble clustering, we describe a subset of the results of an ensemble approach that we used to cluster the dynamics of tyrosine phosphorylation in the epidermal growth factor receptor (EGFR) network measured by Wolf-Yadlin and colleagues (13). From this analysis, we identified previously unknown protein interactions (8). We show a subset of the full analysis to illustrate this process (Fig. 6). PDLIM1, a protein not previously reported as part of the EGFR network, had similar phosphotyrosine dynamics with many other proteins known to directly interact with phosphorylated tyrosine residues of EGFR (Tyr\textsuperscript{1197} and Tyr\textsuperscript{1172}) (Fig. 6A; blue bar). On the basis of these results, we experimentally tested and validated the EGFR phosphotyrosine sites in one of the clusters (Fig. 6B, upper left). PDLIM1; orange bars, EGFR interactors). To identify a more robust representation of the clustering behavior of the system, we generated an ensemble of clusters by varying distance metrics and clustering algorithms. Across the ensemble, known interactions had a much higher tendency to cluster together than with noninteractors. A visualization of the ensemble results—a co-occurrence matrix—places the interactors of these EGFR phosphotyrosine sites in one of the clusters (Fig. 6B, upper left and orange bars), along with the potential interactor PDLIM1 (blue bar). On the basis of these results, we experimentally tested and validated PDLIM1 as a protein that interacted with EGFR (8). It is important to note that, in many of the ensemble clustering results, PDLIM1 did not cluster with all known interactors of EGFR phosphorylated at Tyr\textsuperscript{1197} and Tyr\textsuperscript{1172}. Rather, PDLIM1 tended to cluster with a subset of known interactors. Furthermore, in many ensemble clustering results, the known interactors of EGFR did not all cluster together (Fig. 6C). This demonstrates the value of ensemble clustering because analysis of a single clustering result might have missed this important relationship (8).

Conclusion

Clustering biological data involves a number of choices, many of which are critical to obtaining meaningful results. Evaluation of a data set should be performed to ensure that it has a sufficient number of observations (data points) and that the dimensionality of those observations informs subsequent clustering choices. For each new data set, dimensionality and any transformations applied should influence the choice of appropriate distance metrics and algorithms for clustering. Data sets of more than 10 dimensions often behave unexpectedly, and clustering can produce meaningless results. Using only a single clustering result from any data set can lead to wasted time and resources resulting from erroneous hypothesis testing. The effect of permutations of clustering parameters on the clustering results should be explored using validation metrics and stability testing. When possible, noise and variance should be accounted for in the clustering method directly rather than simply taking averages at each data point. Once clustering results are obtained, their validity should be evaluated using the appropriate metrics. The statistical significance of clustering results should also be evaluated, and multiple
hypothesis correction should be applied when necessary. For robust results, ensemble clustering over a range of distance metrics, transformations, and other clustering parameters is effective. Following these steps will result in obtaining robust and reliable results from clustering and provide a basis for solid generation of testable hypotheses.

SUPPLEMENTARY MATERIALS

www.sciencesignaling.org/cgi/content/full/9/432/6/DC1
File S1. Output of the iPython Notebook that generates all examples in this review.

REFERENCES AND NOTES


Acknowledgments: We thank T. Crum for critical reading. Funding: T.R. is funded, in part, by the Center for Biological Systems Engineering, Washington University in St. Louis (St. Louis, MO). Author contributions: T.R. and K.M.N. contributed equally to the conception, writing, research, and final approval of the paper. Z.G. researched and implemented clustering validation metrics and their dependencies. Competing interests: The authors declare that they have no competing interests. Data and materials availability: All supplemental executable code and file dependencies are available for checkout from the GitHub repository: https://github.com/kaege/clusteringReview.

Submitted 7 August 2015
Accepted 27 May 2016
Final Publication 14 June 2016
10.1126/scisignal.aad1932

Citation: T. Ronan, Z. Qi, K. M. Naegle, Avoiding common pitfalls when clustering biological data. Sci. Signal. 9, re6 (2016).
Avoiding common pitfalls when clustering biological data
Tom Ronan, Zhijie Qi and Kristen M. Naegle

Sci. Signal. 9 (432), re6.
DOI: 10.1126/scisignal.aad1932