AutoClustR Outline – PLOS

**ABSTRACT**

1. Pretty much verbatim
   1. Droplet based sequencing is powerful for elucidating changes in biological systems
   2. Unsupervised clustering identifies distinct cell types, defining the populations that will be compared in downstream analysis.
   3. However, estimating the number and configuration of clusters is difficult and no systematic comparison of scRNA analysis platforms has been performed
   4. To address this challenge, we developed AutoclustR, a wrapper that uses ML to achieved automated clustering.
      1. AutoClustR employs a novel approach to principal component selection
      2. Bayesian optimization is used to optimize parameters of clustering algorithms
   5. We show that AutoClustR outperforms {SC3, RaceID3, CIDR, IKAP + CellFindR) when used to cluster data from different sources, species + technologies
   6. We then apply AutoClustR to a novel dataset generated from inner ear organoids and reveal a previously unappreciated diversity of cell types.

**Introduction**

1. scRNA-seq allows for whole transcriptome profiling at the level of individual cells, which has given scientists new insights into a variety of different fields
   1. But making it mean stuff is hard!
2. A crucial step in the standard workflow is defining clusters of cells in an unbiased fashion
   1. These clusters are commonly thought to represent cell types, physical regions, ect.
   2. The rest of the analysis is then performed on the new cell clusters vs individual cells
      1. DEA, specifically.
      2. Defining what types of cells are present in your model
         1. Spurious clusters can obscure or create new cell types
3. Researchers have applied many different tools to this specific problem
   * 1. K means
     2. K medoids
     3. DBSCAN
     4. Graph based
     5. Hierarchical
   1. Most platforms require an estimate of cluster number, directly or indirectly
   2. Manual parameter tuning can direct & determine the number of clusters found
      1. If the number of expected cell types is unknown a priori, then it becomes difficult to gauge the appropriatness of different clustering partitions **CHOOSE PARTIONS OR SOLUTIONS AND STICK WITH IT**
4. In general, there are two major choices made in scRNA-seq analyses, irrespective of algorithm and platform: The features to retain (inputs) and clustering parameters themselves
5. For the inputs, it’s common to begin with dimensional reduction, going from an unmanageable 30,000 genes to 5-20 principal components
   * 1. However, the number of principal components to retain is non-obvious
     2. Discussion of SE Scree
     3. CNG
     4. Seurat
     5. CIDR
     6. Cell Trails
6. Clustering parameters are usually opaque and the platform’s default parameters are used
   * 1. N.neighbors and resolution in Seurat/graph based
     2. Whatever the fuck SC3 does
   1. Overview of popular clustering algorithms
      1. Seurat
      2. CIDR
      3. SC3
      4. RaceID
7. Failures in prior benchmarking
8. CellFindR + IKAP Discussion
9. ICVI Discussion
10. There remains a collective unmet need, which we’ve filled with AutoClustR

**Results**

1. AutoClustR was built on top of Seurat for three reasons:
   1. Seurat is a widely used, R-based **Platform or Toolkit** which supports data processing, unsupervised clustering, and DEA, the main steps in a scRNA-seq analysis
      1. Implementing AutoClustR on top of Seurat ensures that the tool will be broadly accessible and seamlessly integrated with other pre and post processing functionalities
   2. Seurat’s clustering algorithm requires less compute-time than comparable algorithms
      1. This is critical, because the AutoClustR workflow generates many different clustering **partitions** during the optimization process
      2. The underlying clustering algorithm needs to be time-efficient for reasonable performance
   3. Developing AutoClustR as a Seurat wrapper allows for direct comparison to CellFindR and IKAP
2. In the Seurat workflow, there are two main decisions that are left to the end user: The number of principal components to retain, and the value of the parameters required for unsupervised clustering.
   1. As discussed above, the choices of which features to retain and how to tune a clustering algorithm aren’t unique to Seurat, but rather inherent in any clustering platform.
   2. Selection of principal component to retain is the first choice, because the embeddings of cells within principal component space are the input to Seurat’s graph-based clustering.
   3. Selection of clustering parameters is the second choice.
   4. Previous attempts at automating clustering have focused on determining the optimal number of clusters
      1. For SC3, this is direct testing of different k values in k means
      2. In IKAP, this is indirect through the optimization of the “resolution” of the modularity function in Louvain clustering
   5. However, for graph-based clustering (as is used in Seurat), the K in K Nearest Neighbors clustering, is equally important, if underappreciated
      1. The ‘K’ value determines the interconnectedness of the SNN graph which is the input to the Louvain clustering algorithm
   6. The problems of principal component selection and clustering optimization have been well researched
      1. A myriad of tools and techniques have been developed over the years to solve these exact problems.
      2. Then, the question becomes: Which of these techniques are most useful for the purposes of single cell clustering
   7. To answer this question, we have developed a computational framework that allows a rigorous comparison between all of these factors.