Automatic Gating of Flow Cytometry Data with Recursively Applied Two-Dimensional Density-Based Clustering

Robin Jia

Mentors: Professor David Dill, Robert Bruggner

Introduction

- Flow cytometry is a technique used to measure protein levels of individual cells in a heterogeneous mixture of cells (e.g. from a sample of blood). This process generates large amounts of high-dimensional data, where each cell represents a point in *n*-dimensional space.
- Traditional analysis of flow cytometry data usually relies on experts to manually divide cells into biologically distinct populations, a process known as "gating." The behavior of specific populations has been shown to be predictive of patient phenotypes (e.g. patient response vs. non-response to chemotherapy). Unfortunately, the process of manual gating is labor-intensive and subject to human error and bias.
- We aim to automate the process of gating flow cytometry data and identifying predictive populations. We assess the usefulness of our algorithm based on how well we can predict patient phenotypes using these automatically identified populations.

Clustering

To gate our cells, we use a heavily modified version of an algorithm called Density-Based Merging (Walther et. al. 2009). Given a set of cells, we group these cells into "clusters" in three steps:

- **Density Estimation**: We use a kernel density estimate, which approximates the probability density function for a random cell's position in *n*-dimensional space as the sum of many multivariate Gaussian distributions, where each cell in the data set contributes one multivariate Gaussian centered at its location
- **Peak Finding**: The kernel density estimate gives us a sort of terrain on which we can find peaks and valleys. A peak is local maximum of the density function, and a valley is a saddle point that is the maximal lowest point on any path connecting two neighboring peaks. See fig. 1 for a 1-D example.
- **Peak Merging**: Given the densities (or "heights," from the viewpoint of terrain) of two peaks and the valley between them and the standard errors of these density estimates, we compute a metric of how significant the valley is. This is an approximation to the probability that the valley height is actually less than the peak heights. If a valley is not sufficiently deep, then the two peaks it separates are effectively the same peak, except for a small ripple, and we merge these peaks.

Unlike parametric techniques that assume that clusters have certain shapes (e.g. Gaussian), this procedure can find arbitrarily-shaped populations. Analysis of flow data shows that clusters do sometimes have unusual shapes.

Our Recursive Procedure

Due to the curse of dimensionality, kernel density estimation works best in low-dimensional space. Additionally, the kernel density estimate grids the data, and thus scales exponentially in the number of dimensions. To reduce complexity, we choose to only run our clustering algorithm on two-dimensional projections of the data; this also allows us to easily visualize the results of clustering. To fully cluster a sample, we cluster data on many different two-dimensional projections, and at each step take the results from the pair of axes that gives the cleanest split between distinct populations.

- Given a set of cells with *n* different protein levels measured, cluster the cells on all *n*-Choose-2 different pairs of axes
- For each set of clustering results, continue to merge peaks until at most two peaks remain. Do this by repeatedly merging the two peaks that have the least-significant valley between them until only two peaks remain. Out of all clustering results, select the clustering that has the most significant valley.
- Take each cluster this clustering yields and recursively process it in the same way, until the clusters do not split in any dimension.

Fig. 2 shows one final cluster produced after a series of 2-dimensional clusterings, with the axes chosen by the above procedure.

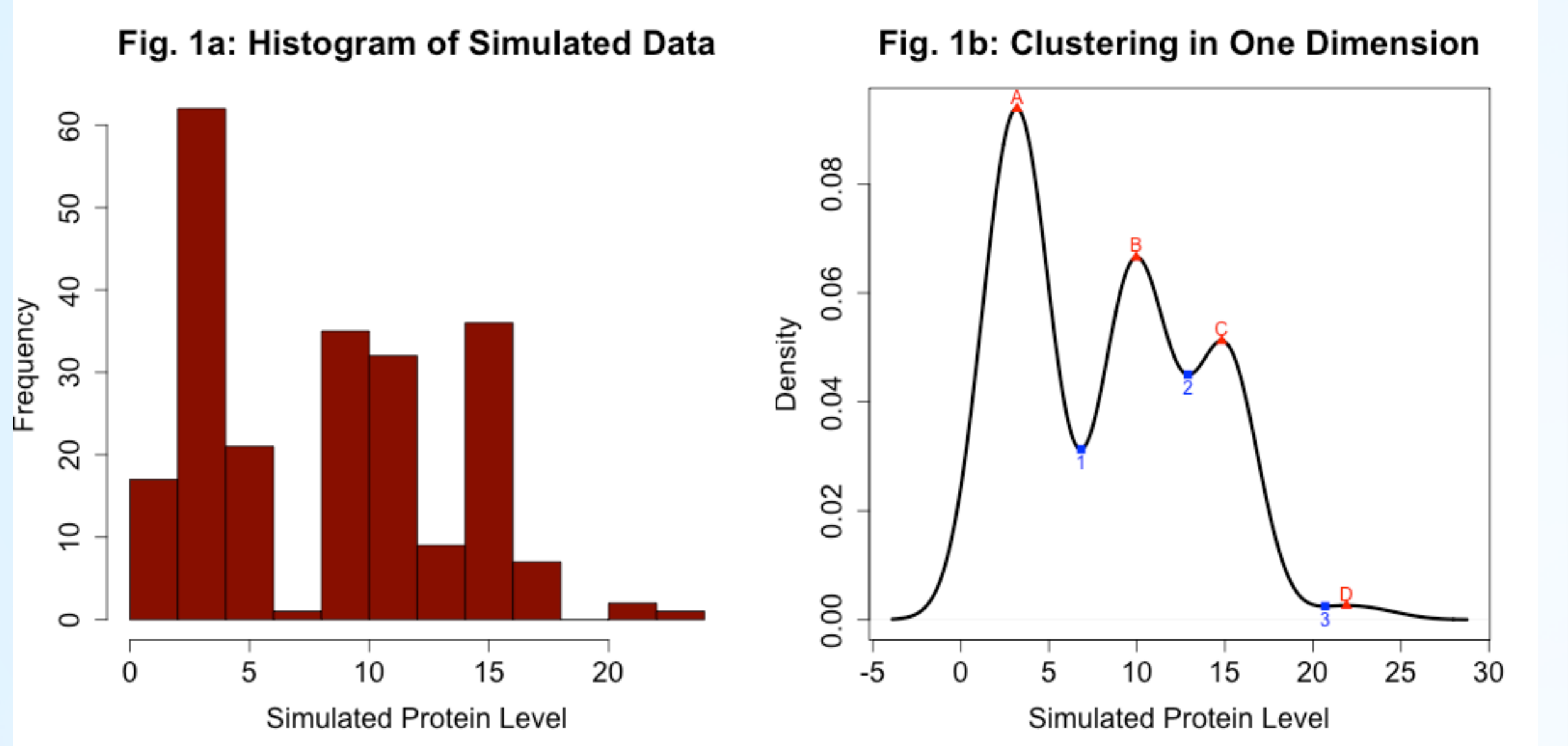

Fig. 1: In one dimension, clustering proceeds by creating a smooth density function and finding local maxima and minima. Valley 3 is insignificant, so C and D are merged. Valley 2 is significant, but less significant than Valley 1, so B is merged with C and D if the caller of the clustering function requires a maximum of two clusters

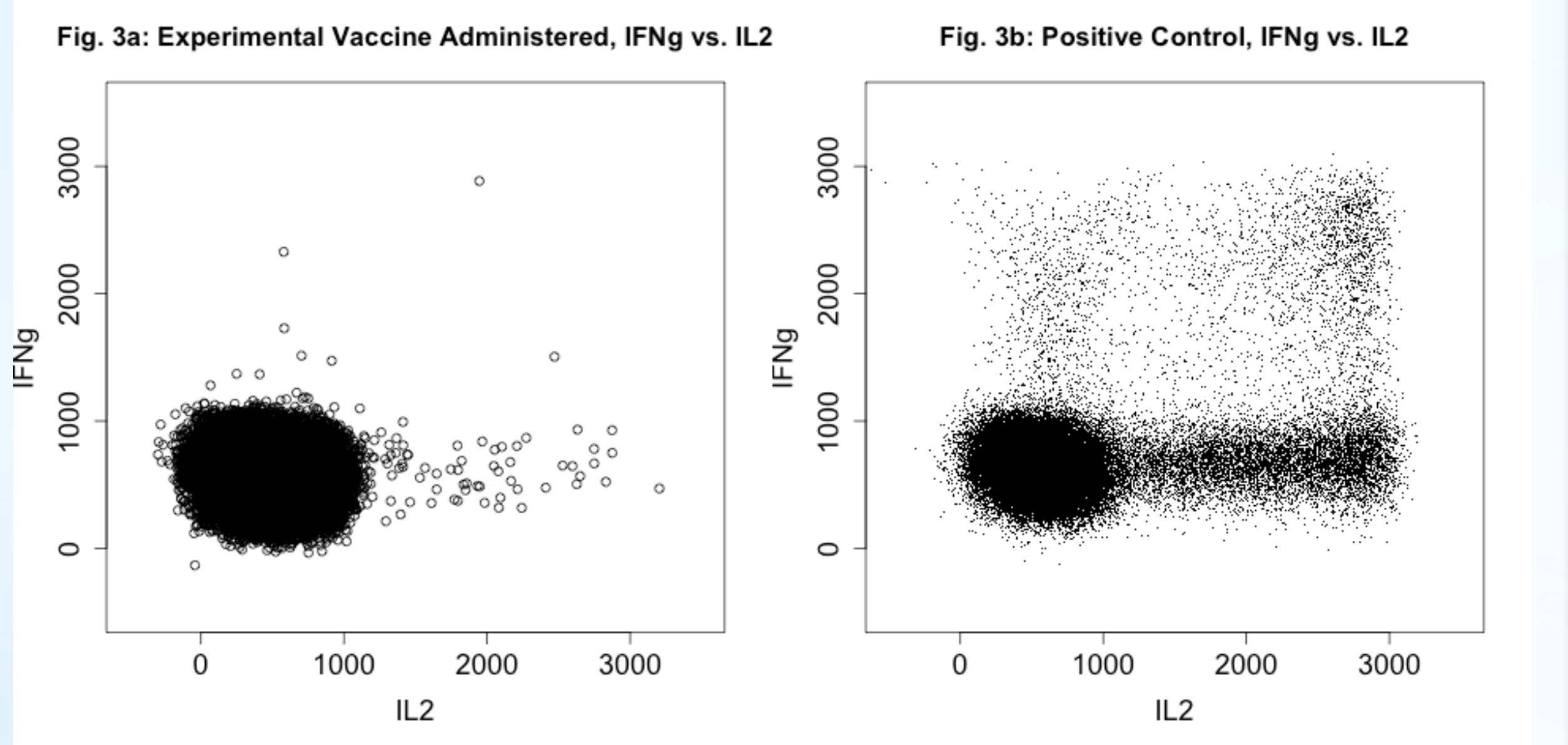

Fig. 3: In many test cases (left), only slight stimulation of cells occurs. In order to detect this group of stimulated cells using clustering, we can concatenate the file with a positive control (right). The positive control has a well-defined peak in the same region as the activated cells of the test case.

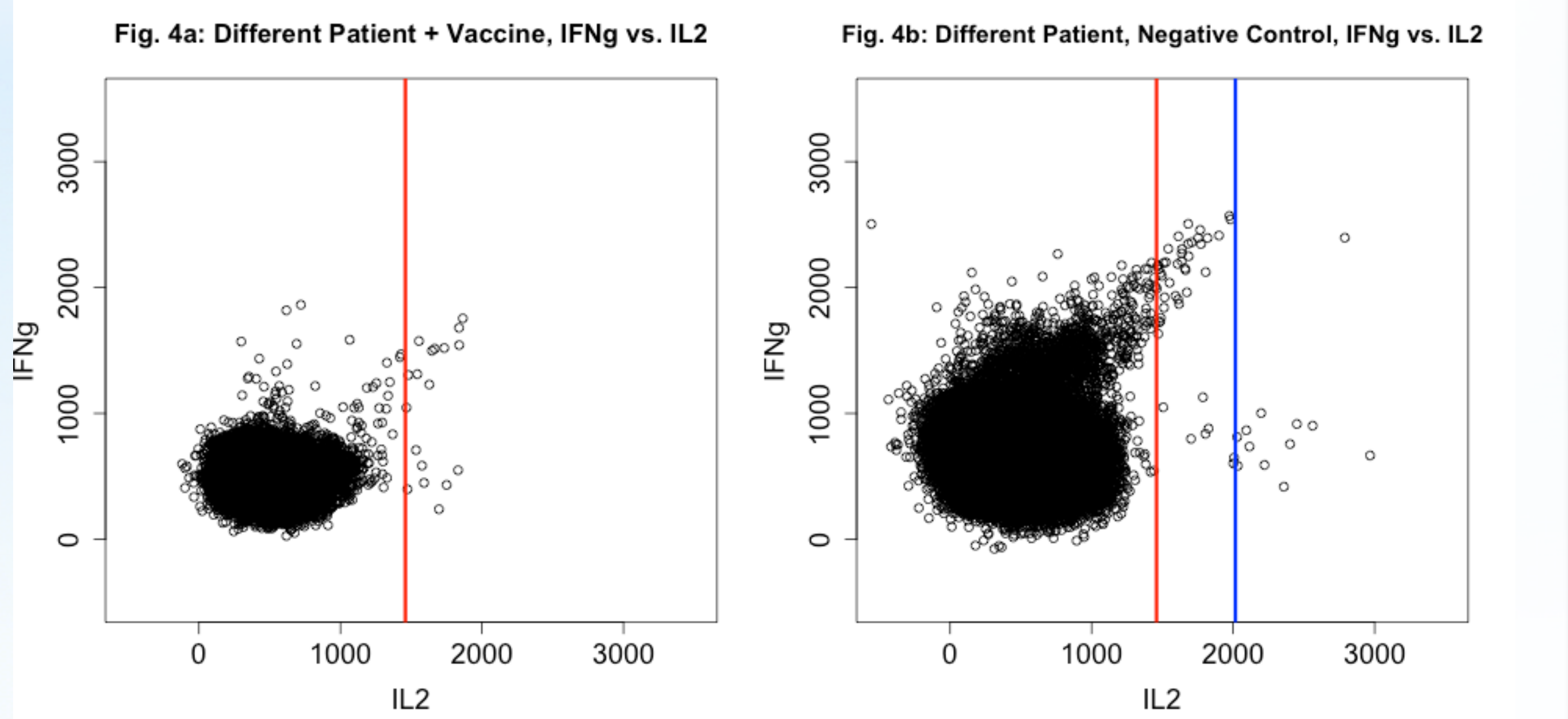

Fig. 4: Certain files present special issues. To the human eye, the stimulated file on the left seems to have more activated cells, but the negative control file on the right has a "tail"-like feature protruding from the main unactivated cluster. If we use the same cutoff point for both files (red line), we catch more "activated" cells on the negative control than on the stimulated case. However, if we use a file-specific metric—in this case, counting a certain number of standard deviations from the mean (blue line)—then we can better determine which cells are activated for each individual file.

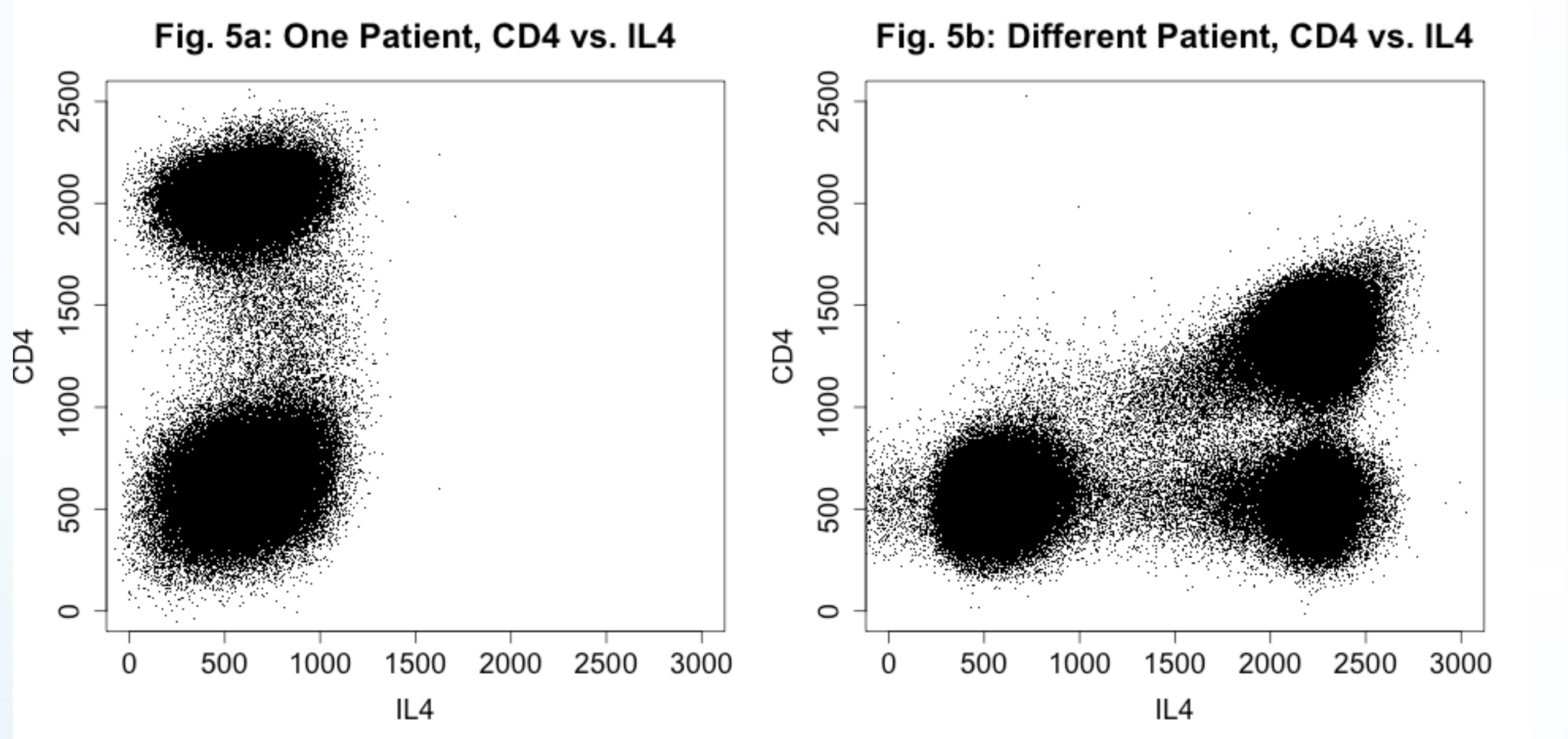

Fig. 5: When two patients exhibit different clusters (IL4) and different levels of expression (CD4), normalization and cluster matching become more difficult. Without biological knowledge, it is difficult here to determine which clusters should map to which.

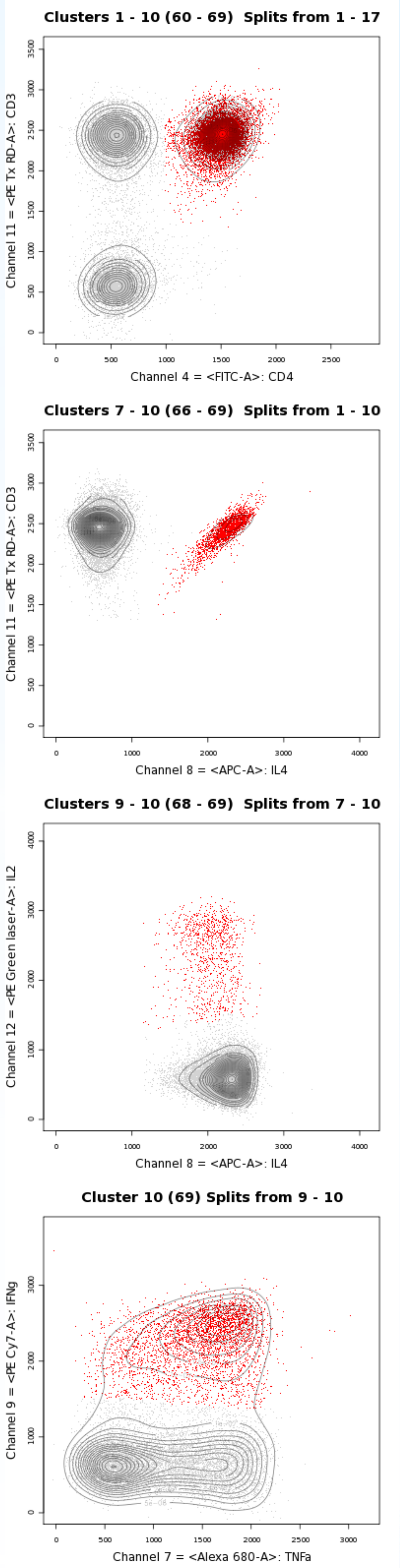

Fig. 2: Example one terminal cluster generated by our *n*-dimensional clustering routine. Four different 2-D projections were used to arrive at this terminal cluster. At each stage, the highlighted red cells were recursively re-clustered until they could no longer be split in any pair of dimensions.

The Importance of Small Populations

- Often times, the interesting populations in a patient—such as cancer cells, stem cells, or stimulated populations—are also very rare. These small clusters can be very hard to detect:
  - Can be drowned out by noise
  - Can merge with neighboring clusters without forming a local maximum
- **Solution #1: Positive Control (see fig. 3)**
  - For cases where we want to detect the presence of stimulated cells, the population can be artificially stimulated, enlarging the stimulated cluster
  - We concatenate this positive control file with the other testing files, so that when we cluster this concatenated data, we detect the stimulated cluster, which will include many positive control cells and some test cells
- **Alternative: Outlier Search**
  - Only cluster on "cell type marker" channels that distinguish different types of cells, but not on "response" channels that show whether these cells have been stimulated or not
  - For each different type of cell, define the stimulated cells to be the cells that are significantly higher in one of the response channels (e.g. many standard deviations from the mean)
  - Advantages: doesn't require cells to form a cluster with a visible local maximum, doesn't require the ability to produce a positive control, and allows us to tune our definition of "stimulated" to each file (see fig. 4)
  - Disadvantage: requires more knowledge of the data and more human tuning of input parameters, so is less automated.

Normalization

- When we want to analyze flow data, we often wish to compare different samples
  - Different patients
  - The same patient under different stimulation conditions
- However, noise due to irrelevant biological differences between patients or inconsistent calibration between runs of the flow cytometer make it difficult to compare different data files, as measurements may be shifted or skewed
- At the same time, some differences in cluster patterns are important to keep as they distinguish different patients or stimulation conditions from each other
- **Solution 1: Landmark Normalization**
  - Look at one-dimensional density plots of each data file on each channel
  - Find "landmarks"—local maxima of the density function—for each file
  - In each dimension, apply a transformation to the data that aligns landmarks from different files that are close to each other
  - Disadvantages: landmarks can't be detected when clusters are small and may not yield local maxima
- **Solution 2: Cluster Matching**
  - Cluster each file individually first
  - Declare clusters from different files to be the same cluster if they are in similar locations in *n*-dimensional space
  - Disadvantage: Can't take advantage of positive control (unless one patient's files are concatenated, and cluster matching is done inter-patient)
- For both strategies, natural differences between files can cause problems (see fig. 5)

Conclusion

- Automatic gating of flow cytometry data that can perform at or better than manual gold-standards is an achievable goal
  - We may have already reached this goal, but evaluating the quality of clustering output directly is difficult
  - Our current approach to evaluating clustering results looks at the predictive power of discovered populations
- More reliable normalization or cluster matching procedures could increase the accuracy of predictions derived from flow data
- Detecting activated cells may be best left to other techniques that do not look for peaks but instead find outliers