2012 IEEE International Workshop on Genomic Signal Processing and Statistics (GENSIPS)
December 2-4, 2012, Washington, DC, USA

Toward exhaustive gating of flow cytometry data

Peng Qiu
Department of Bioinformatics and Computational Biology
The University of Texas MD Anderson Cancer Center

***Abstract*—Flow cytometry is a high-throughput technology that measures protein expressions at the single-cell level. A typical flow cytometry experiment on one biological sample provides measurements of several protein markers on or inside hundreds of thousands of individual cells in that sample. Analysis of such data often aims to identify subpopulations of cells with distinct phenotypes. Currently, the most widely used analysis in the flow cytometry community is manual gating on a sequence of biaxial plots, which is highly subjective and labor intensive. To address those issues, the majority of efforts in the literature have been devoted to automate the gating analysis using clustering algorithms. However, completely removing the subjectivity can be quite challenging. This paper describes an opposite approach. Instead of automating the analysis, we aim to develop novel visualizations to facilitate manual gating. The proposed method views a flow cytometry data of one biological sample as a high-dimensional point cloud of cells, derives the skeleton of the cloud, and unfolds the skeleton to generate a 2D visualization.**

I. Introduction

Flow cytometry simultaneously measures the expressions of multiple proteins at the single-cell level [1]. The flow cytometry data of one biological sample can be presented in the form of a tall thin matrix, where each row corresponds to one individual cell and each column corresponds to one protein marker. Modern flow cytometers can simultaneously measure up to 12 proteins routinely, and the capacity of the next-generation mass cytometer is more than 30 [2]. The total number of cells depends on the experiment design, and is typically on the order of a hundred thousand. Such single-cell data reflects the cellular heterogeneity of the sample, which is of great biological interests [3].

Analysis of flow cytometry data often aims to identify subtypes of cells with distinct phenotypes, which is basically a clustering problem. Currently, the most widely used approach in the flow cytometry and immunology community is manual gating [4]. A gate is a region in a biaxial plot of two protein markers, which is used to select cells with a desired phenotype. Cells selected by one gate are visualized in other biaxial plots, in which further gates are drawn to refine the selection. The result of gating analysis is a hierarchy of gates on a user-defined sequence of nested biaxial plots. Each gate and the cells in it are annotated with a different phenotype. Manual gating is highly subjective because the sequence of biaxial plots relies on user's prior knowledge and interpretation of the biological system underlying the data. Moreover, manual gating is not exhaustive, in the sense that the manual gates typically do not cover all the cells, leaving a non-trivial amount of the cells unannotated.

To reduce the subjectivity and achieve exhaustive gating, efforts have been made to automate the gating analysis using clustering algorithms, such as K-means [5], mixture models, [6], [7], density-based clustering [8], [9], and spectral analysis [10]. Many of these methods include mechanisms for both determining the number of clusters and clustering the cells, which remove the subjectivity from the gating analysis. However, it is difficult to tune a clustering algorithm to separate all the subpopulations that a human expert would define, due to the fact that cell counts of different subpopulations are usually highly unbalanced.

In this paper, we take an alternative approach. Instead of automating the gating analysis, we aim to develop novel visualizations to help manual exploration of the data. We believe many disadvantages of manual gating are caused by the poor visualization of biaxial plots, which only encode pairwise correlation. A better visualization that captures higher order correlations can greatly facilitate manual analysis. A recent method, SPADE, illustrated that tree diagrams can be used to approximate high-dimensional relationships [11], [12], which motivated this work. The goal here is to encode the high-dimensional relationships among individual cells into a 2D visualization, and enable manual gating analysis on the 2D visualization rather than a sequence of user-defined nested biaxial plots in traditional manual gating. The proposed method views a flow cytometry dataset as a high-dimensional point cloud, performs density-dependent downsampling to balance the sizes of different subpopulations, derives the skeleton of the downsampled cloud by combining K-means clustering and minimum spanning tree, and unfolds the skeleton to generate a 2D visualization. A motivating examples and details of the method are presented in the following sections.

II. Materials and Methods

A. Motivating example

As a motivating example, this section describes manual gating of a real flow cytometry dataset, which will also be used to illustrate the proposed method in the Results section. The dataset is derived from healthy mouse bone marrow, and is previously published [11]. The total number of cells is 500,000. The measured protein markers include: c-kit, CD11b, B220, TCRβ, CD4 and CD8, which are able to delineate the major cell types in mouse bone marrow [13]. Immature progenitor cells are positive for c-kit. Myeloid cells express CD11b, whereas lymphoid cells are negative for this marker. Within the lymphoid cell population, B cells express B220 but not TCRβ, whereas the majority of T cells express TCRβ but

978-1-4673-5236-9/12/$31.00 ©2012 IEEE

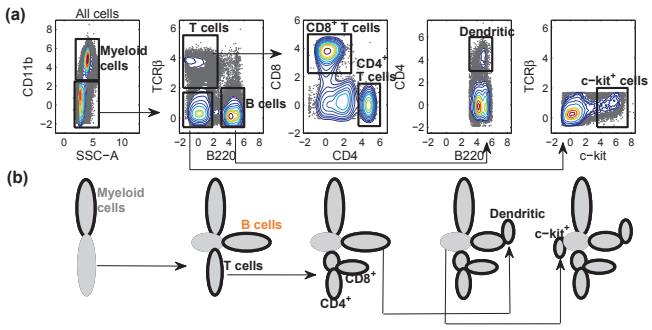

Fig. 1. Analysis of a mouse bone marrow flow cytometry dataset. (a) The sequence of biaxial plots used in manual gating, and the gates that identify subpopulations with different phenotypes. (b) Illustration of the point cloud analogy: the shape of the data as understood from the manual gating process.

not B220. Mature T cells can be divided into two subtypes, characterized by the expression of either CD4 or CD8. In addition to the above protein markers, the data contains measurements for forward and side scatters (FSC and SSC) for each cell, which reflect cell size and granularity.

Manual gating of this dataset is shown in Figure 1 (a). The first biaxial plot uses SSC and CD11b to visualize all 500,000 cells. Since the contours suggest two subpopulations, two gates are manually drawn. The upper gate contains cells positive for CD11b, and is annotated as myeloid cells according to prior knowledge of mouse hematopoiesis. Cells in the other gate are further visualized in the second biaxial plot using markers B220 and TCRβ, where three subpopulations can be observed. Three gates are drawn in the second plot, and two of them are annotated as B cells and T cells because of their mutually exclusive expression of B220 and TCRβ. Following the same logic, the third plot uses CD4 and CD8 to show the distribution of T cells, allowing the identification of the two subtype of T cells. The forth plot reveals a $\text{B220}^{+}\text{CD4}^{+}$ subpopulation within the B cell gate, which is annotated as dendritic cells. Finally, the fifth plot shows that a c-kit positive subpopulation exists in the $\text{B220}^{-}\text{TCR}\beta^{-}$ gate of the second plot. During manual gating, a hierarchy of gates are drawn on a user-defined sequence of nested biaxial plots. Each gate contains cells with a distinct phenotype.

If the data is considered as a point cloud of cells, manual gating can be viewed as a user-guided process that tries to understand the shape of the cloud. This intuition is illustrated by Figure 1 (b). In the first biaxial plot, the point cloud is projected on to a 2D subspace defined by SSC and CD11b. In this subspace, the point cloud appears to be composed of two arms. In the second biaxial plot, cells in the lower arm are projected to another 2D subspace defined by B220 and TCRβ, which shows that the lower arm can be further divided into three smaller arms. Similarly, each subsequent biaxial plot focuses on one arm of the cloud and reveals more detailed structures within that arm. The last plot in Figure 1 (b) illustrates the shape of the point cloud revealed by the gating process. The point cloud itself lives in a high-dimensional space, and the illustration in Figure 1 (b) can be viewed as an unfolded version of the high-dimensional cloud. This observation motivates the following question: is it possible to computationally unfold a high-dimensional point cloud so that the unfolded version fits in a 2D space, similar to unfolding an origami back to a piece of paper? The following section describes a method to answer this question.

B. Unfold high-dimensional point cloud

To unfold a high-dimensional point cloud to 2D, the proposed method uses several computational steps: density-dependent downsampling, K-means clustering, minimum spanning tree (MST) construction, unfold the MST to 2D, and mapping of individual cells to the unfolded MST.

Density-dependent downsampling: In a point cloud that represents a flow cytometry dataset, the densities in different regions of the cloud can vary dramatically. Dense regions correspond to abundant cell types, whereas sparse regions correspond to rare cell types and noise. One critical challenge in clustering cytometry data is the unbalanced sizes of different subpopulations. Since the primary goal here is to identify the shape of the cloud, density-dependent downsampling is applied to remove the density variation within the cloud while preserving its shape [11].

Density-dependent downsampling stochastically removes cells according to user-defined target density (TD) and outlier density (OD):

$$\text{p(remove } i) \quad = \begin{cases} 1, & \text{if } LD_i \leq OD \\ 0, & \text{if } OD < LD_i \leq TD \\ 1 - \frac{TD}{LD_i}, & \text{if } LD_i > TD \end{cases}$$

where LD_i is the local density for cell i, estimated by the number of cells within its neighborhood. An empirical choice of neighborhood size is 5 times the median distance from a randomly chosen cell to its nearest neighbor. During the downsampling process, cells with local densities $< OD$ are considered as noise and removed. Cells whose local densities are between OD and TD are not downsampled. Cells in dense regions are stochastically removed, so that the densities of the dense regions reduce to TD.

K-means clustering: After downsampling, standard K-means clustering is applied to partition the downsampled cloud into smaller pieces. Since the density variation in the cloud is removed by the previous step, the resulting clusters tend to share similar sizes, in terms of both their volumes and cell counts. The key setting of this step is that the desired number of clusters k should be much larger than the expected number of subpopulations in the data. Although such setting inevitably over-partitions the data, it enables the subsequent steps to recover the topology of the point cloud.

MST construction: By representing each cell cluster using its center, the pairwise distances between clusters can be defined by the pairwise Euclidean distances between their centers. Using this distance metric, the Boruvka's algorithm [14] is applied to construct an MST to connect all k cluster centers using $k-1$ edges with minimum total edge length. The MST approximates the skeleton of the cloud.

Unfold MST to 2D: Since the cluster centers are points in the high-dimensional space that the cloud lives in, the MST is also a high-dimensional object that cannot be directly visualized. To automatically generate a 2D layout for the MST, a modified version of the Fruchterman-Reingold algorithm is applied [15]. First, the longest path in the tree is visualized horizontally. Then, the remaining nodes are appended one by one. The position of each newly appended node is determined by simulating: repelling forces between the new node and ones that are already visualized, and an attracting force along the new edge. Although this algorithm does not prohibit edges to cross each other, its resulting layout typically does not contain edge crossings. After this step, each cluster center is assigned to one position in a 2D visualization space.

Map cells to the unfolded MST: Denote the positions of two cluster centers in the data space as $\mathbf{d}_i = [d_{i1}, d_{i2}, d_{i3}, ..., d_{in}]^T$, $\mathbf{d}_j = [d_{j1}, d_{j2}, d_{j3}, ..., d_{jn}]^T$, and denote their corresponding positions in the 2D visualization space as $\mathbf{v}_i = [v_{i1}, v_{i2}, 0, ..., 0]^T$, $\mathbf{v}_j = [v_{j1}, v_{j2}, 0, ..., 0]^T$. Since any two line segments can be related by a rotation operation and a scaling operation, there exist a non-negative scalar α and a rotation matrix $\mathbf{R}$ that satisfy,

$$\mathbf{v}_j - \mathbf{v}_i = \alpha \mathbf{R}(\mathbf{d}_j - \mathbf{d}_i) \tag{1}$$

The value of α is the ratio between the length of the two segments. Determining the rotation matrix $\mathbf{R}$ is not as straightforward, and $\mathbf{R}$ is in fact not unique. One way to construct an $\mathbf{R}$ is by multiplication of a sequence of 2D rotation matrices that rotates $\mathbf{d}_j - \mathbf{d}_i$ to $\mathbf{v}_j - \mathbf{v}_i$ one dimension at a time.

For one data point $\mathbf{d}_x$, assume $\mathbf{d}_i$ is its nearest cluster center and $(\mathbf{d}_i, \mathbf{d}_j)$ is its nearest tree edge in the data space, this data point is visualized by the following equation:

$$\mathbf{v}_x = \alpha \mathbf{R}(\mathbf{d}_x - \mathbf{d}_i) + \mathbf{v}_i \tag{2}$$

where α and $\mathbf{R}$ satisfy equation (1), and the visualization position of the data point is determined by the first two elements of $\mathbf{v}_x$.

III. Results

To illustrate the proposed visualization, the mouse bone marrow data in Figure 1 is used. The data contains single-cell measurements of 6 protein markers on 500,000 individual cells, which can be viewed as a cloud of 500,000 points in a 6-dimensional space. In the density-dependent downsampling step, OD is set to be the 5th percentile of the local densities of all cells, meaning that 5% of cells with the lowest local densities are excluded. TD is chosen such that 20,000 cells survive the downsampling process. The number of clusters in the K-means clustering step is chosen to be 100. After clustering, the proposed method constructs an MST to link the clusters, unfold it to a 2D visualization space, and maps individual cells to the unfolded MST. Figure 2 shows the 2D visualization of the MST and the 20,000 cells after downsampling.

Contour plots are used to illustrate the expressions of protein markers. To generate such a plot for one marker, a $100 * 100$ grid is applied to the visualization space, and each of the downsampled cells is assigned to its nearest grid point. By defining the "height" above each grid point as the average marker intensity of the downsampled cells assigned to it, a contour plot can be created. In Figure 3, each contour plot shows the expression of one protein marker. By comparing these plots, all gates in the manual gating analysis can be identified. According to the coloration of CD11b, the left half of the visualized cloud is positive for this marker, which corresponds to the myeloid gate. From the plots colored by B220 and CD4, the right one third of the cloud expresses B220, which corresponds to the B cell gate. Within the B220$^+$ region, the CD4$^+$ arm represents the dendritic cells. Plots colored by TCRβ, CD4 and CD8 show that the middle arm pointing upwards is TCRβ^+ and contains two parts that exhibit mutually exclusive expression of CD4 and CD8. This arm corresponds to gates related to T cells. Since the upper right corner of the visualized cloud is positive for c-kit, this region contains cells in the c-kit$^+$ gate.

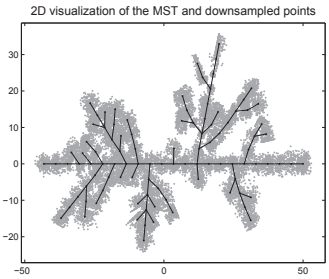

Fig. 2. 2D visualization of the 6-dimensional mouse bone marrow data. The MST is shown in black. The gray dots represent the 20,000 cells after downsampling.

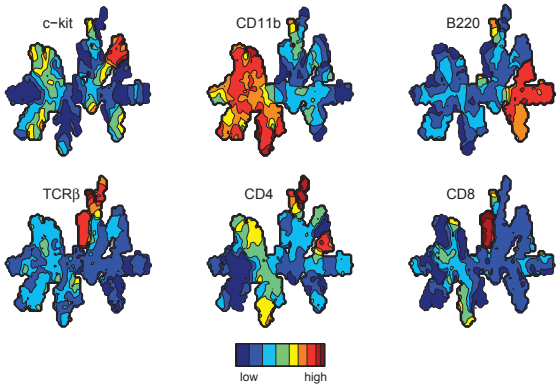

Fig. 3. Contour plots drawn according to protein expressions. Each plot is colored by one protein marker, where red indicates regions that contain cells that highly express the marker, and blue indicates regions consist of cells that are negative for the marker.

With the visualization in Figure 3, manual gating can be performed in a new way: by specifying a desired combination of multiple markers. For example, to examine whether there exists a particular marker combination B220$^+$CD4$^+$TCRβ^- in the data, one can specify a set of criteria (i.e. B220>3, CD4>3, TCRβ <2) and highlight the regions that satisfy the criteria. The thresholds chosen here are consistent to those used in the gating plots in Figure 1. As shown in Figure 4, there is one subpopulation in this dataset that exhibits

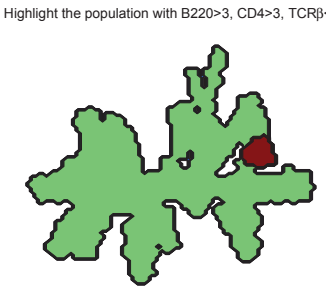

Fig. 4. Contour plot that highlights a population with user-specified phenotype: B220>3, CD4>3, TCRβ <2. This population corresponds to the dendritic cells in the gating analysis.

the specified marker combination, and this subpopulation corresponds to the dendritic cells in the gating analysis.

By specifying other marker combinations, one can easily manually explore what known phenotypes exist in the data. Since all phenotypes that exist in the data are reflected in one 2D visualization, after marker combinations of known phenotypes are used to highlight regions of the proposed 2D visualization, regions that do not satisfy prior knowledge can be easily observed, whose marker combinations can form hypotheses of novel phenotypes that are not known a priori. This is the main advantage of the proposed visualization, **enabling exhaustive gating**.

IV. Discussion

When manually explore the proposed visualization, specifying marker combinations requires thresholds that are data-dependent (preprocessing, transformation and normalization). One simple way to identify appropriate thresholds is by examining biaxial plots in Figure 1. Therefore, the proposed visualization should be used in conjunction with biaxial plots, rather than replacing them. Similar to traditional manual gating, the proposed method requires user's prior knowledge. However, as mentioned above, the main advantage here the proposed visualization enables exhaustive gating.

In the clustering step, the desired number of clusters is a tuning parameter. As stated in Materials and Methods, this step is intended to over-partition the point cloud, and the number of clusters should be much larger than the number of expected subpopulations. If this parameter is too small, the subsequent MST will not be able to approximate the skeleton of the cloud. If this parameter is too large, the resulting tree will contain many noisy branches. Although the MST is sensitive to this parameter, after the individual cells are visualized, small noisy branches tend to be buried in the final visualization (i.e., Figures 3). Empirically, 100 $\sim$ 300 is a good choice for the desired number of clusters. Defining the optimal number of clusters for tree-based skeletonisation remains a challenging problem. Another interesting question is how to identify the skeleton of a point cloud without clustering.

The robustness of the proposed visualization is difficult to quantify. Density-dependent downsampling and K-means clustering both involve randomness. The randomness will affect the tree structure and the final visualization. However, in multiple runs of the proposed analysis, the main phenotypes (marker combinations) are always separated and reflected by different regions in the 2D visualization. Therefore, from the prospective of the biological interpretations, the proposed visualization is robust. To support this statement and enable readers to test the proposed method, we provide the raw data and MATLAB code for our analysis at http://odin.mdacc.tmc.edu/~pqiu/software/UNFOLD/index.html.

Acknowledgment

This work is supported by NIH grant R01 CA163481.

References

[1] P. Chattopadhyay, D. Price, T. Harper, M. Betts, J. Yu, E. Gostick, S. Perfetto, P. Goepfert, R. Koup, S. De Rosa, M. Bruchez, and M. Roederer, "Quantum dot semiconductor nanocrystals for immunophenotyping by polychromatic flow cytometry," *Nature Medicine*, vol. 12, no. 8, pp. 972 – 977, 2006.

[2] S. Bendall, E. Simonds, P. Qiu, E. Amir, P. Krutzik, R. Finck, R. Bruggner, R. Melamed, O. Ornatsky, R. Balderas, S. Plevritis, K. Sachs, D. Pe'er, S. Tanner, and G. Nolan, "Single cell mass cytometry of differential immune and drug responses across the human hematopoietic continuum," *Science*, vol. 332, no. 6030, pp. 687–696, 2011.

[3] E. Newell, N. Sigal, S. Bendall, G. Nolan, and M. Davis, "Cytometry by Time-of-Flight Shows Combinatorial Cytokine Expression and Virus-Specific Cell Niches within a Continuum of CD8 + T Cell Phenotypes," *Immunity*, vol. 36, no. 1, pp. 142 – 152, 2012.

[4] L. Herzenberg, J. Tung, W. Moore, L. Herzenberg, and D. Parks, "Interpreting flow cytometry data: a guide for the perplexed," *Nature Immunology*, vol. 7, no. 7, pp. 681 – 685, 2006.

[5] N. Aghaeepour, R. Nikolic, H. Hoos, and B. RR, "Rapid cell population identification in flow cytometry data," *Cytometry A*, vol. 79, no. 1, pp. 6 – 13, 2011.

[6] K. Lo, R. Brinkman, and R. Gottardo, "Automated gating of flow cytometry data via robust model-based clustering," *Cytometry A.*, vol. 73, no. 4, pp. 321 – 332, 2008.

[7] S. Pyne, X. Hu, K. Kang, E. Rossin, T. Lin, L. Maier, C. Baecher-Allan, G. McLachlan, P. Tamayo, D. Hafler, P. De Jager, and J. Mesirov, "Automated high-dimensional flow cytometric data anlysis," *Proceedings of the National Academy of Science*, vol. 106, no. 21, pp. 8519 – 8524, 2009.

[8] G. Walther, N. Zimmerman, W. Moore, D. Parks, S. Meehan, I. Belitskaya, J. Pan, and L. Herzenberg, "Automatic clustering of flow cytometry data with density-based merging," *Advances in Bioinformatics*, 2009.

[9] Y. Qian, C. Wei, F. Lee, J. Campbell, J. Halliley, J. Lee, J. Cai, Y. Kong, E. Sadat, E. Thomson, P. Dunn, A. Seegmiller, N. Karandikar, C. Tipton, T. Mosmann, I. Sanz, and R. Scheuermann, "Elucidation of seventeen human peripheral blood B-cell subsets and quantification of the tetanus response using a density-based method for the automated identification of cell populations in multidimensional flow cytometry data," *Cytometry Part B: Clinical Cytometry*, vol. 78B, no. S1, pp. S69–S82, 2010.

[10] H. Zare, P. Shooshtari, A. Gupta, and R. Brinkman, "Data reduction for spectral clustering to analyze high throughput flow cytometry data," *BMC Bioinformatics*, vol. 11, no. 1, p. 403, 2010.

[11] P. Qiu, E. Simonds, S. Bendall, K. Gibbs Jr., R. Bruggner, M. Linderman, K. Sachs, G. Nolan, and S. Plevritis, "Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE," *Nature Biotechnology*, vol. 29, no. 10, pp. 886–891, 2011.

[12] P. Qiu, "Inferring phenotypic properties from single-cell characteristics," *PLoS One*, vol. 7, no. 5, p. e37038, 2012.

[13] D. Bryder, D. Rossi, and I. Weissman, "Hematopoietic stem cells: The paradigmatic tissue-specific stem cell," *Am J Pathol*, vol. 169, no. 2, pp. 338–346, 2006.

[14] S. Pettie and V. Ramach, "An optimal minimum spanning tree algorithm," *Journal of the ACM*, vol. 49, pp. 49–60, 1999.

[15] P. Qiu and S. K. Plevritis, "TreeVis: A MATLAB-based tool for tree visualization," *Computer Methods and Programs in Biomedicine*, in press, 2012.