Check for updates

Received: 18 December 2018 | Revised: 25 February 2019 | Accepted: 26 February 2019

DOI: 10.1111/ijlh.13016

SUPPLEMENT ARTICLE

WILEY ISLH International Journal of Laboratory Hematology

Flow cytometry data analysis: Recent tools and algorithms

Sebastiano Montante[1] | Ryan R. Brinkman[1,2]

[1]Terry Fox Laboratory, BC Cancer, Vancouver, British Columbia, Canada

[2]Department of Medical Genetics, University of British Columbia, Vancouver, British Columbia, Canada

Correspondence
Ryan R. Brinkman, Department of Medical Genetics, University of British Columbia, Vancouver, BC, Canada.
Email: rbrinkman@bccrc.ca

Funding information
National Institute of General Medical Sciences

Abstract

Flow cytometry (FCM) allows scientists to rapidly quantify up to 50 parameters for millions of cells per sample. The bottleneck in the application of the technology is data analysis, and the high number of parameters measured by the current generation of instruments requires the use of advanced computational algorithms to make full use of their capabilities. This review summarizes the main steps of FCM data analysis, focusing on the use of the most recent bioinformatic tools developed for an R-based programming environment. In particular, for each stage of the data analysis, libraries and packages currently available are listed, and a brief description of their functioning is included.

KEYWORDS
automated gating, bioinformatics, clustering, data analysis, flow cytometry

1 | FCM DATA ANALYSIS FRAMEWORK

The steps that characterize a FCM data analysis can be grouped into six major stages[1]:

1. Data pre-processing
   a. Compensation
   b. Quality assessment
   c. Normalization
   d. Transformation
2. Cell population identification
3. Cross-samples comparison (population mapping or matching).
4. Features extraction
5. Interpretation (discovery or diagnosis)
6. Visualization

Here we describe, an automated analysis pipeline that has been implemented to analyze flow cytometry data derived from a multicenter study.[2,3]

2 | DATA FORMATS AND DATA PRE-PROCESSING

The flowCore package provides the infrastructure for any R-based FCM analysis, the programming language of choice based on both available functionality and number of freely available tools.[4] Table 1 lists all the packages discussed in this review and their associated stage within the FCM data analysis framework. flowCore allows users to represent and manipulate the FCM data. It implements computationally efficient data structures defined by the Data Standards Task Force (DSTF) of the International Society for the Advancement of Cytometry (ISAC)[5] Thanks to this infrastructure, all packages discussed in this review can handle data generated by all flow cytometers machines currently available in the market. In flowCore, FCM data are organized in flowFrames and flowSets. Usually, a flowSet contains multiple flowFrames and each flowFrame reports the data of one experiment (it represents a unit sample) and relative meta-data (ie, information about the data).[4] Within each flowFrame object, the events are indexed along the rows while the parameters (ie, the markers of the experiment) are disposed along the columns (Figure 1).[6] The package implements a set of specialized functions that address the main components of a typical FCM analysis workflow.[4] In particular, the compensation and transformation step of the data pre-processing stage are implemented in the flowCore package, along with the gating stage.[4]

Often, users of computational tools are analyzing large datasets. However, the flowFrame and flowSet objects require that all data elements reside in RAM in order to perform the data manipulations and if the flowSet contains datasets that overcome the RAM space limit, processing fails. The ncdfFlow package allows the handling of

 © 2019 John Wiley & Sons Ltd

TABLE 1 Steps of the FCM data analysis framework and associated packages

Stage	Step	Software
Pre-processing:	Compensation	flowCore, flowUtils
	Transformation	flowCore, flowTrans
	Normalization	flowStats (fdaNorm, gaussNorm)
	Quality Assessment	QUALIFIER, flowAI, flowClean
Gating:	Sequential manual gating	FlowJo, flowCore
	Automated gating	Supervised algorithms: flowDensity, DeepCyTOF, flowLearn, OpenCyto Unsupervised algorithms: flowMeans, SPADE, Citrus, FlowSOM, cytometree, ECLIPSE
Cross-sample comparison	Population matching	flowMatch, flowmap-RF
Interpretation	Diagnosis and discovery	flowType, MetaCyto, CytoCompare
Extra processing steps	Visualization	flowViz, ggCyto, RchyOptimyx, SPADE, Citrus, t-sne

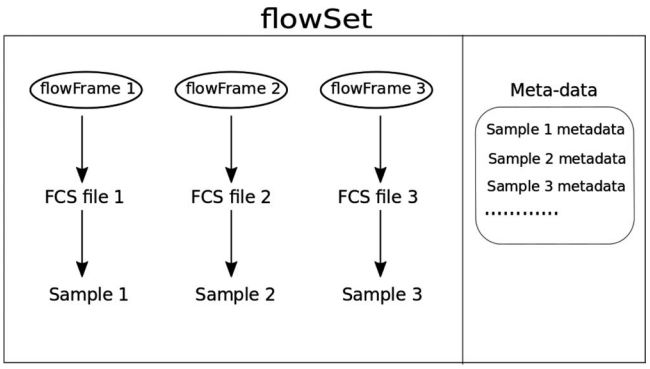

FIGURE 1 Graphical representation of the flowCore infrastructure

such datasets in memory and it is able to perform the same pre-processing steps of the flowCore package.[7] To do so, ncdfFlow creates a ncdfFlowSet (similar to the standard flowSet object) that stores the large volume of data on the hard drive storage and only keeps the file handler and meta-data in the RAM, significantly reducing memory requirements.[7]

2.1 | Compensation

Compensation is implemented in the flowCore package,[4,8] along with the flowUtils package.[8] flowUtils is a R package designed to read Gating-ML files. Gating-ML (Gating Markup Language) files are Extensible Markup Language (XML) files that describe gates transferable between different software packages. Gating-ML stores gates, compensation, and transformation data so that they are computationally reproducible.[9]

2.2 | Quality assessment

The aim of data quality assessment is to detect whether intersample measurement variations derive from instrument variations or from biological causes. The samples with measurements altered by technical errors should be removed from the analysis.[1,10] The QUALIFIER package uses the gating template created in FlowJo (package used for manual gating) and performs quality checks on different gated populations.[11] There are two available approaches to detect and remove event-level anomalies. flowAI evaluates three different properties: flow rate, signal acquisition, and dynamic range.[12] flowClean detects anomalies in the data by tracking fluorescent measurement fluctuations within a sample during acquisition time.[13]

2.3 | Normalization

Normalization is a pre-processing step performed on a set of samples with the aim of removing effects that arise from technical variation rather than from biological differences. In this way, the analysis can focus on the important and relevant biological variations between samples.[1,14] Instrument variability, experimental protocol changes, and reagent changes (eg, using antibodies from different companies) are examples of nonbiological factors that can introduce variability in the data and alter the location of cell populations. However, automated FCM data analysis requires uniform, quantitative, and comparable raw data which can be obtained by developing normalization methods.[1] Data normalization also helps the cross-sample population matching stage where the aim is to detect biologically relevant cell populations across a set of samples, and technical variations can make this process more challenging.[14] The flowStats package contains functions, methods, and classes implementing algorithms for statistical analysis of FCM data. It includes functions for data normalization.[4,14] The guassNorm and the fdaNorm functions normalize a set of FCM data samples by identifying and aligning the high-density regions (landmarks or peaks) for each channel. The data of each channel are shifted in such a way that the identified high-density regions are moved to fixed locations called base landmarks.[14] These two functions differ only in the implementation of the three major steps of the algorithm: landmark identification, landmark registration, and landmark alignment.[14]

2.4 | Transformation

Accurate automated gating of FCM data is complicated by asymmetric and overlapping cell populations, frequent outlier events, cell populations whose variance depend on their mean fluorescence intensity, and errors in the fluorescence channels.[15] All of these characteristics can influence the output of both manual and automated

1751553x, 2019, S1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/ijlh.13016 by Davit Sargsyan - Ethicon Somerville , Wiley Online Library on [07/04/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

gating, and subsequent downstream analysis. An optimal transformation process is an algorithm that facilitates cell population gating, visualization, and intersample comparison, in order that the cell populations are well resolved across the full range of data.[15] There are many transformations used for FCM data.[15] The log transformation can often stabilize the variance of cell populations in the fluorescence channels but cannot represent negative data values of unstained cell populations. This results in a compression of data against the axes and in a low-quality representation of low intensity or unstained populations.[15] For this reason, other transformations can be applied, including the linear-logarithmic (linlog) transformation, the biexponential (logicle),[15] generalized arcsinh transformations, and the generalized Box-Cox transformation.[15] All these transformations are implemented in the BioConductor flowCore package. Transformation algorithms have adjustable parameters whose effects have an important impact on the quality of the different steps of the FCM analysis. However, the correct setting of these parameters is not a simple task. flowTrans is an R package for optimizing the parameters of the most commonly used FCM data transformations. Parameter-optimized data transformations show an improved quality of the downstream analysis steps when compared to default parameter transformations.[15]

3 | CELL POPULATION IDENTIFICATION

The most critical and time-consuming step in manual analysis is the identification of homogeneous cell populations in the data, a process commonly known as gating. In particular, multiple populations are identified within individual samples and compared across samples (called also population matching step).[1] A cell population is a group of events within a sample that share particular characteristics (ie, they are similar cells), measured by the markers used in the experiment. Gating methods can be divided in two main categories: Sequential (Manual) Gating and Automated Gating.[1]

3.1 | Sequential manual gating

Traditional gating-based analysis is performed manually, and it is based on visual comparison of one or two-dimensional plots. A sequence of gates must be used to analyze multidimensional datasets.[16,17] This method of identification of cell populations currently relies on using software to apply a series of manually drawn gates that select regions in the data plots representing the two parameters along the two axes. This process is based on the expertise of the operator rather than standardized statistical inference. It also ignores the high dimensionality of FCM data, which may contain information that cannot be displayed in 1D or 2D plots.[1,16]

Commercial software packages have been traditionally used by research groups for sequential gating. These tools allow for limited forms of software-assisted gating where users iteratively build a gating model by sequentially choosing sets of axes (ie, two markers) to visualize the data, manually drawing boundaries (gates) around subsets of cells and then restricting the next visualization to the cells within a chosen gate.[17-19] Therefore, sequential gating is characterized by many limitations[16,17]:

1. The determination of the boundaries during the gating step is not necessarily guided by rules (subjectiveness).
2. Traditional gating works on two-dimensional plots at most. As a consequence, the analysis misses the information contained in the multidimensional space.
3. Analysis of higher dimensional dataset is time-consuming.

For these reasons, there is a high interest for studies about gating strategies based on statistical models, with over 38 automated methods developed to date. These methods are able to automatically infer cells populations directly from the multidimensional dataset, overcoming the limitations of the sequential gating.[16,20]

3.2 | Automated gating

Automated gating is based on the mathematical modeling of the fluorescence intensity distribution of cell populations.[18] The Flow Cytometry: Critical Assessment of Population Identification Methods (FlowCAP) project provides a set of challenges to compare the performance of the computational methods involved in FCM.[21] Automated gating can be performed using two different approaches: unsupervised and supervised. These techniques can be used to address the problems, previously described, faced in manual gating.[1,22]

3.2.1 | Supervised cell population identification

In a supervised analysis, the operator requires a dataset that has two distinct elements[1]:

1. The marker measurements for each event. The marker measurements constitute the explanatory variables in the mathematical model.
2. The cell type that is indicated by a label and it is associated with each event (ie, each cell). In other words, the label indicates the class which each event belongs, and all cells that belong to the same class represent a single population of cells. The cell type constitutes the dependent variable of the mathematical model.

This labeled dataset contains the training data that the algorithm will use during the training stage to learn the relationship between the explanatory variables and a dependent variable, which contains a set of classes indicated by the labels.[1] The algorithm will use the information learned during the training stage to assign unlabeled events to one of the classes (ie, the cell type) defined in the training dataset.

flowDensity is a tool that performs automated gating using a supervised algorithm. This tool automates a predefined manual gating approach. It is important to underline that the user must know the gating strategy.[23] The algorithm is based on a sequential bivariate gating approach that generates a set of predefined cell populations.

1751553x, 2019, S1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/ijlh.13016 by Davit Sargsyan - Ethicon Somerville , Wiley Online Library on [07/04/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

flowDensity estimates the region around cell populations by using characteristics of the marker density distribution, where flowDensity evaluates the best cut-off for each marker.[23]

The OpenCyto framework is a collection of R packages that include ncdfFlow, flowCore, flowViz, flow Workspace, and the package openCyto.[24] The openCyto package inside the OpenCyto implements a hierarchical automated gating pipeline that performs data pre-processing and data-driven automated gating. The framework can perform the gating analysis using high dimensional gating algorithms or traditional sequential gating. In particular, a gating scheme, produced by an external software (like flowJo), can be imported. Alternatively, the gates can be defined using a data-driven approach, utilizing various gating algorithms available in R or BioConductor.[24]

3.2.2 | Unsupervised cell population identification

In unsupervised approaches, the operator does not need any labels, any predefined class as reference. In other words, there is no dependent variable. The most used unsupervised algorithms in FCM are based on clustering.[1] Generally speaking, the clustering algorithm identifies the events that are in the same cluster. Similar events stay in the same cluster, different events stay in different clusters.[1] Clustering works on the multidimensional dataset overcoming the limitations of the sequential gating.[1] The strategy to determine the clusters depends on the specific algorithm, that can be model-based (eg, Gaussian Mixture Model clustering) or non–model based (eg, K-means clustering).[25]

Some automated gating using unsupervised algorithms that include model-based and non–model-based approaches include the following:

- flowMeans performs clustering with a modified version of the K-means algorithm that can identify concave populations (unlike traditional K-means) by merging multiple clusters to obtain the final population.[20]
- The SPADE (Spanning-tree Progression Analysis of Density-normalized Events) algorithm allows the partition of FCM data into many hierarchically organized clusters that reflect all the dimensions in the data. The user can identify and annotate known and new cell types taking advantage of a minimum spanning tree visualization.[19]
- Citrus is a tool that identifies population of cells using hierarchical clustering, as SPADE. However, CITRUS uses known "endpoints" (ie, particular status of the samples, like diseased vs healthy samples) combined with a regularized supervised algorithm to determine the populations and features that are relevant for the association of the sample to a specific endpoint group.[26]

Tools based on artificial neural networks used in an unsupervised context have been recently developed, an example is FlowSOM. FlowSOM uses a minimal spanning tree visualization as the SPADE algorithm. However, while SPADE uses hierarchical clustering, FlowSOM is based on self-Organizing Maps that are artificial neural networks used in an unsupervised context.[27] A recent comparison of flowSOM with other 17 clustering methods found flowSOM as the best performance algorithm in terms of runtime. In particular, flowSOM is the best performing method for datasets with multiple cell populations and provides also good performance with datasets characterized by rare populations, in this last case the user must perform a correct initialization of start parameters. FlowSOM is an ideal tool for performing rapid exploratory analyses of large datasets due to its fast runtime. In order to obtain the best performance, the user must manually select the optimal number of clusters to be predicted, especially when the aim is to detect rare populations. Multiple random starts of bootstrapping resampling are also suggested.[22]

- The tool cytometree is based on an unsupervised decision tree algorithm. In particular, it is based on a binary tree in which each node represents a subpopulation of cells. The algorithm has showed an excellent performance compared to other unsupervised algorithms.[28]
- ECLIPSE (Elimination of Cells Lying in Pattern Similar to Endogeneity) is a tool mainly designed to identify healthy cells (only one specific cell type) in FCM data. It uses Simultaneous Component Analysis (SCA), a generalization of PCA that aims to reduce the dimensions of the original dataset. Then it identifies (and eliminates) the healthy cells in the new low-dimensional dataset comparing the density of cells belonging to the patient samples with the cells belonging to control samples.[29]

3.2.3 | Machine learning approaches for cell population identification

In recent years, deep learning methods have showed impressive performance in various applications, such as image analysis, natural language processing, and pattern recognition.[30] Deep learning typically requires very large numbers of training instances. In each FCM experiment, approximately 10^5-10^6 cells are collected, so that the number of instances (cells or events inside one sample) is several times higher than the number of explanatory variables.[30] In other words, deep learning algorithms perform well with huge multidimensional datasets that are datasets characterized by a high number of events and a high number of markers. Most recent FCM technology can analyze up to 50 different markers, and the technology improves continuously, increasing the complexity of cytometry experiments.[28] Therefore, the use of deep learning algorithms to analyze FCM data will likely increase in the future.[30]

DeepCyTOF is a tool recently developed that uses a deep learning algorithm in a supervised contest to perform automated gating. DeepCyTOF employs one manually gated reference sample as training dataset and uses it for automated gating of the remaining samples of the study, thus requiring labeled cells from only one sample.

flowLearn is a tool that shares some characteristics with deepCyTOF and flowDensity. Compared to flowDensity, the manually tuning of hyper-parameters is not necessary. It requires at least one representative sample already manually gated. Once the channels

1751553x, 2019, S1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/ijlh.13016 by Davit Sargsyan - Ethicon Somerville , Wiley Online Library on [07/04/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

threshold of this sample has been extracted, they are automatically transferred to the other samples of the flowSet.[31]

4 | CROSS-SAMPLE POPULATION MATCHING AND FEATURE EXTRACTION

After cell populations are identified in individual samples, the next step is to match cell populations across samples so that the cell population characteristics can be compared with the whole sample set. The determination of the population characteristics is called "Feature extraction".[1] The marker expression levels of equivalent cell populations can shift between different samples due to technical artifacts and natural biological variability.[25] The strategy to compare the differences between populations of different samples involves usually meta-clustering and templates construction concepts [32] (Figure 2):

- A meta-cluster is a set of biologically similar cell clusters from different samples.
- Template is a collection of meta-clusters from samples of same class.
- Different classes represent different external conditions associated to the samples (like disease status).

flowMatch is a tool that performs cross-sample population matching, exploiting the meta-clusters and templates structures. Flowmap-RF is another tool that performs cross-sample population matching but it directly compares the cell populations using the FriedmanRafsky (FR) test statistic, a nonparametric multivariate statistical test that uses a minimum spanning tree visualization approach.[25]

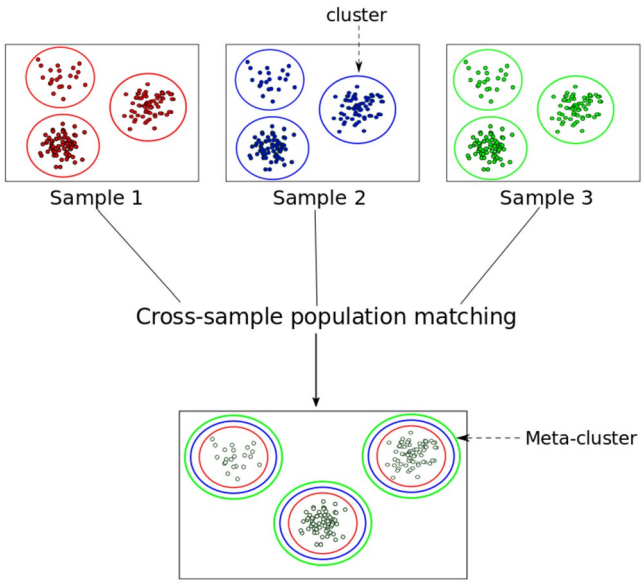

FIGURE 2 The matching of different clusters to obtain the meta-clusters in the class template. Each colored circle indicates a cluster of cells, clusters with the same color belong to the same sample. After the population matching across all the samples, a template (bigger rectangle) is obtained. In the template, meta-clusters are delimited by concentric circles of different colors, because they contain a population of cells derived by the matching of the same population in different samples of the same class [Colour figure can be viewed at wileyonlinelibrary.com]

5 | INTERPRETATION: CORRELATION WITH AN EXTERNAL VARIABLE

The interpretation of results step can be aided with many computational methods in order to determine the association between FCM samples and a class of interest (eg, healthy vs sick samples) or to identify clusters of patients with similar FCM data.[33] Depending on the purpose of the study, supervised or unsupervised learning can be used. Supervised techniques can be used for classification of a sample.[33] Unsupervised techniques can be utilized for the aggregation of patients with similar data.[33]

Therefore, supervised and unsupervised techniques can be used at two levels[1]:

1. To automate cell population identification stage (automated gating stage).
2. To automate discovery and diagnosis stages (automated association with a clinical outcome).

A useful tool for interpretation purposes is flowType. This algorithm automatically reveals all possible cell subsets of a multidimensional dataset. The subsets that correlate strongly with clinical outcome are selected and grouped. Within each group, markers that have minimal relevance to the biological outcome are removed, so that the original complex dataset is reduced into a smaller simple dataset with the simplest combination of markers (ie, the most important ones).[34]

MetaCyto is a tool designed to identify common cell populations across different studies (one study includes a specific set of samples). It then estimates the influence of external factors, such as age or ethnicity, using hierarchical models, based on the information received by all the studies.[35]

CytoCompare is an R package that is able to compare the phenotypes of cell clusters identified by automatic gating algorithms. Its aim is to facilitate the easy identification of similar and different cell clusters, using the density distribution of the cell marker expression.[36]

6 | VISUALIZATION

The interpretation and cell population identification stages are the most complex phases of the FCM framework, especially when the FCM data have a high number of markers. Many visualization tools have been developed to help the researchers in their efforts to analyze high-dimensional datasets. Most of these tools are based on unsupervised algorithms.

During the FCM analysis, it can be very useful to plot particular distributions (like CDF or PDF) of the events in relation to

1751553x, 2019, S1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/ijlh.13016 by Davit Sargsyan - Ethicon Somerville , Wiley Online Library on [07/04/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

a particular channel or multiple channels. Often, the user wants to compare distributions for different samples or other variables. These plots are implemented in the flowViz package. flowViz uses data structures defined in the flowCore package and is also available from Bioconductor.[37] ggCyto is another tool developed for FCM data visualization. It contains ggplot functionalities that are able to interact with the core BioConductor FCM data structures. It is compatible with un-gated data (flowSet or flowFrame objects) or gated data (GatingSet or GatingHierarchy objects imported from FlowJo) and it is characterized by a great flexibility and customization.[38] RchyOptimyx (cellular hierarchy optimization) is a tool that constructs cellular hierarchies (using dynamic programming and graph theory) providing information about the relationship between the markers involved in the identification of the target population and the characteristics of that population correlated with an external outcome. It allows users to explore graphically the importance of each marker in the gating strategy, representing an important tool for biomarker discovery projects.[33] SPADE performs a hierarchical clustering of the cells in an unsupervised manner and represents them in a tree structure. Thus, it provides a two-dimensional visualization of multiple cell types in an ordered structure, showing graphically how the markers levels change across the cell types.[19] Citrus is also useful for visualization purposes. As mentioned in the previous section, Citrus uses hierarchical clustering and its output includes a list of stratifying clusters, bi-dimensional plots, and various data representations describing the phenotype of each cluster.[26] Finally, the t-stochastic neighbor embedding (t-SNE) algorithm is a dimensionality reduction technique that maps a high-dimensional dataset into a lower-dimensional dataset of two or three dimensions, so that the data can be easily represented in 2D or 3D plots. The algorithm aims to preserve the similarity between the two datasets. This algorithm performs well with FCM data.[39]

7 | FUTURE PERSPECTIVES

The optimal packages to be used for each step should be evaluated by the operator based on the aims of the research. There are factors that require a case-by-case evaluation such as the need to identify specific rare cell populations or the need to visualize accurately the relation of each population to an external outcome. Here, we have described on possible solution that has been implemented for use in a clinical flow cytometry laboratory. Classic automated gating algorithm usually requires some parameters in order to obtain the best performance, representing an intensive time-consuming process. Machine learning algorithms are able to overcome these limitations as they do not require manual setting of the parameters, raising the research efforts on machine learning studies. In recent years, deep learning algorithms have become very popular,[40] and they will revolutionize health research.[40] Their increasing popularity is explained by the high accuracy in dealing with large multidimensional datasets and by their ability to perform automatic features extraction, making them very suitable to handle FCM data, where multidimensionality still represents a major issue during the data analysis.[40] For these reasons, the development of tools that incorporate deep learning algorithms is at the forefront of FCM research.

Flow cytometry technology is widely used in clinical and research laboratories. FCM experiments can provide many information that can help to resolve and analyze deeply many biomedical problems.[16,17] However, the possibility to take full advantage of this information is prevented by the complexity of the data analysis stage, where the gating step constitutes its major bottleneck. Therefore, the development of tools that are able to completely automate the gating stage is a promising research field, that will change dramatically the daily practice of this technology, by making simpler its data analysis stage.

ACKNOWLEDGEMENTS

This project is funded by the National Institute of General Medical Sciences (NIGMS).

CONFLICTS OF INTEREST

Ryan Brinkman has ownership interest in Cytapex Bioinformatics Inc.

ORCID

Ryan R. Brinkman https://orcid.org/0000-0002-9765-2990

REFERENCES

1. Bashashati A, Brinkman RR. A survey of flow cytometry data analysis methods. *Adv Bioinformatics*. 2009;2009:19.
2. Ivison S, Malek M, Garcia RV, et al. A standardized immune phenotyping and automated data analysis platform for multicenter biomarker studies. *JCI Insight*. 2018;3:121867.
3. Conrad VK, Dubay CJ, Malek M, Brinkman RR, Koguchi Y, Redmond WL. Implementation and validation of an automated flow cytometry analysis pipeline for human immune profiling. *Cytometry A*. 2019;95:183-191.
4. Hahne F, LeMeur N, Brinkman RR, et al. flowCore: a bioconductor package for high throughput flow cytometry. *BMC Bioinformatics*. 2009;10:106.
5. Spidlen J, Moore W, Parks D, et al. Data file standard for flow cytometry, version FCS 3.1. *Cytometry A*. 2010;77:97-100.
6. Ellis B, Haaland P, Hahne F, et al. flowCore: Basic structures for flow cytometry data. https://bioconductor.org/packages/release/bioc/html/flowCore.html. Accessed: November 16, 2018.
7. Mike J, Greg F. ncdfFlow: A package that provides ncdf based storage for flow cytometry data. http://bioconductor.org/packages/2.10/bioc/html/ncdfFlow.html. Accessed: November 12, 2018.
8. Roederer M. Spectral compensation for flow cytometry: visualization artifacts, limitations, and caveats. *Cytometry*. 2001;45(3):194-205.
9. Spidlen J, Leif RC, Moore W, Roederer M, Brinkman RR; International Society for the Advancement of Cytometry Data Standards Task Force. Gating-ML: XML-based gating descriptions in flow cytometry. *Cytometry A*. 2008;73A:1151-1157.

1751553x, 2019, S1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/ijlh.13016 by Davit Sargsyan - Ethicon Somerville , Wiley Online Library on [07/04/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

10. Le Meur N, Rossini A, Gasparetto M, Smith C, Brinkman RR, Gentleman R. Data quality assessment of ungated flow cytometry data in high throughput experiments. *Cytometry A*. 2007;71A(6):393-403.
11. Finak G, Jiang W, Pardo J, Asare A, Gottardo R. QUALIFIER: an automated pipeline for quality assessment of gated flow cytometry data. *BMC Bioinformatics*. 2012;13:252.
12. Monaco G, Chen H, Poidinger M, Chen J, de Magalhães JP, Larbi A. flowAI: automatic and interactive anomaly discerning tools for flow cytometry data. *Bioinformatics*. 2016;32(16):2473-2480.
13. Fletez-Brant K, Špidlen J, Brinkman RR, Roederer M, Chattopadhyay PK. flowClean: automated identification and removal of fluorescence anomalies in flow cytometry dat. *Cytometry A*. 2016;89(5):461-471.
14. Hahne F, Khodabakhshi AH, Bashashati A, et al. Per-channel basis normalization methods for flow cytometry data. *Cytometry A*. 2009;77A(2):121-131.
15. Finak G, Perez JM, Weng A, Gottardo R. Optimizing transformations for automated, high throughput analysis of flow cytometry data. *BMC Bioinformatics*. 2010;11:546.
16. Frelinger J, Ottinger J, Gouttefangeas C, Chan C. Modeling flow cytometry data for cancer vaccine immune monitoring. *Cancer Immunol Immunother*. 2010;59:1435-1441.
17. Meehan S, Walther G, Moore W, et al. AutoGate: automating analysis of flow cytometry data. *Immunol Res*. 2014;58:218-223.
18. Pyne S, Hu X, Wang K, et al. Automated high-dimensional flow cytometric data analysis. *Proc Natl Acad Sci* 2009;106(21):8519-8524.
19. Qiu P, Simonds EF, Bendall SC, et al. Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nat Biotechnol*. 2011;29:886-891.
20. Aghaeepour N, Nikolic R, Hoos HH, Brinkman RR. Rapid cell population identification in flow cytometry data. *Cytometry A*. 2010;79A(1):6-13.
21. Aghaeepour N, Finak G, Hoos H, Mosmann TR; FlowCAP Consortium; DREAM Consortium. Critical assessment of automated flow cytometry data analysis techniques. *Nat Methods*. 2013;10:228-238.
22. Weber LM, Robinson MD. Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data. *Cytometry A*. 2016;89(12):1084-1096.
23. Malek M, Taghiyar MJ, Chong L, Finak G, Gottardo R, Brinkman RR. flowDensity: reproducing manual gating of flow cytometry data by automated density-based cell population identification. *Bioinformatics*. 2015;31:606-607.
24. Finak G, Frelinger J, Jiang W, et al. OpenCyto: an open source infrastructure for scalable, robust, reproducible, and automated, end-to-end flow cytometry data analysis. *PLoS Comput Biol*. 2014;10:e1003806.
25. Hsiao C, Liu M, Stanton R, McGee M, Qian Y, Scheuermann RH. Mapping cell populations in flow cytometry data for cross-sample comparison using the Friedman-Rafsky test statistic as a distance measure. *Cytometry A*. 2016;89:71-88.
26. Bruggner RV, Bodenmiller B, Dill DL, Tibshirani RJ, Nolan GP. Automated identification of stratifying signatures in cellular subpopulations. *Proc Natl Acad Sci U S A*. 2014;111:E2770-E2777.
27. Van Gassen S, Callebaut B, Van Helden MJ, et al. FlowSom: using self-organizing maps for visualization and interpretation of cytometry data. *Cytometry A*. 2015;87(7):636-645.
28. Commenges D, Alkhassim C, Gottardo R, Hejblum B, Thiébaut R. cytometree: a binary tree algorithm for automatic gating in cytometry analysis. *Cytometry A*. 2018;93:1132-1140.
29. Folcarelli R, van Staveren S, Bouman R, et al. Automated flow cytometric identification of disease-specific cells by the ECLIPSE algorithm. *Sci Rep*. 2018;8:10907.
30. Li H, Shaham U, Stanton KP, Yao Y, Montgomery RR, Kluger Y. Gating mass cytometry data by deep learning. *Bioinformatics*. 2017;33:3423-3430.
31. Lux M, Brinkman RR, Chauve C, et al. flowLearn: fast and precise identification and quality checking of cell populations in flow cytometry. *Bioinformatics*. 2018;34:2245-2253.
32. Azad A, Pyne S, Pothen A. Matching phosphorylation response patterns of antigen-receptor-stimulated T cells via flow cytometry. *BMC Bioinformatics*. 2012;13(Suppl 2):S10.
33. Aghaeepour N, Jalali A, O'Neill K, et al. RchyOptimyx: cellular hierarchy optimization for flow cytometry. *Cytometry A*. 2012;81:1022-1030.
34. Aghaeepour N, Chattopadhyay PK, Ganesan A, et al. Early immunologic correlates of HIV protection can be identified from computational analysis of complex multivariate T-cell flow cytometry assays. *Bioinformatics*. 2012;28:1009-1016.
35. Hu Z, Jujjavarapu C, Hughey JJ, et al. MetaCyto: a tool for automated meta-analysis of mass and flow cytometry data. *Cell Rep*. 2018;24:1377-1388.
36. Platon L, Pejoski D, Gautreau G, et al. A computational approach for phenotypic comparisons of cell populations in high- dimensional cytometry data. *Methods*. 2018;132:66-75.
37. Sarkar D, Le Meur N, Gentleman R. Using flowViz to visualize flow cytometry data. *Bioinformatics*. 2008;24(6):878-879.
38. Van P, Jiang W, Gottardo R, Finak G. ggCyto: next generation open-source visualization software for cytometry. *Bioinformatics*. 2018;34(22):3951-3953.
39. Saeys Y, Van Gassen S, Lambrecht BN. Computational flow cytometry: helping to make sense of high-dimensional immunology data. *Nat Rev Immunol*. 2016;16:449-462.
40. Ching T, Himmelstein DS, Beaulieu-Jones BK, et al. Opportunities and obstacles for deep learning in biology and medicine. *J R Soc Interface*. 2018;15:20170387.

How to cite this article: Montante S, Brinkman RR. Flow cytometry data analysis: Recent tools and algorithms. *Int J Lab Hematol*. 2019;41(Suppl. 1):56-62. https://doi.org/10.1111/ijlh.13016

1751553x, 2019, S1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1111/ijlh.13016 by Davit Sargsyan - Ethicon Somerville , Wiley Online Library on [07/04/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License