scientific reports

OPEN

K-means quantization for a web-based open-source flow cytometry analysis platform

Nathan Wong, Daehwan Kim, Zachery Robinson, Connie Huang & Irina M. Conboy

Flow cytometry (FCM) is an analytic technique that is capable of detecting and recording the emission of fluorescence and light scattering of cells or particles (that are collectively called "events") in a population[1]. A typical FCM experiment can produce a large array of data making the analysis computationally intensive[2]. Current FCM data analysis platforms (FlowJo[3], etc.), while very useful, do not allow interactive data processing online due to the data size limitations. Here we report a more effective way to analyze FCM data on the web. Freecyto is a free and intuitive Python-flask-based web application that uses a weighted k-means clustering algorithm to facilitate the interactive analysis of flow cytometry data. A key limitation of web browsers is their inability to interactively display large amounts of data. Freecyto addresses this bottleneck through the use of the k-means algorithm to quantize the data, allowing the user to access a representative set of data points for interactive visualization of complex datasets. Moreover, Freecyto enables the interactive analyses of large complex datasets while preserving the standard FCM visualization features, such as the generation of scatterplots (dotplots), histograms, heatmaps, boxplots, as well as a SQL-based sub-population gating feature[2]. We also show that Freecyto can be applied to the analysis of various experimental setups that frequently require the use of FCM. Finally, we demonstrate that the data accuracy is preserved when Freecyto is compared to conventional FCM software.

Abbreviations

FCM	Flow cytometry
Event(s)	Emission(s) of fluorescence and light scattering of cells or particles
t-SNE	Barnes-Hut approximation of t-distributed stochastic neighbour embedding
K-means	Lloyd's Algorithm with Euclidean distances for k-means clustering (k-means++ is used for cluster center initialization).
MSE	Mean squared error
WT	Wild type
GFP	Green fluorescent protein
IMR-90	Human lung fibroblast cells

Flow cytometry is broadly used in biomedicine, which is exemplified by identification of protein marker expressions[1–6], determinations of cell-fate and cell cycle progression[7], analysis of pathology-caused changes, e.g. cancer promoted, immune-skewing, etc.[8–11], testing therapeutic efficacy of a treatment[12], and, more recently, gene-editing detection workflows[13]. A common experimental setup in biomedicine relies on being able to identify specific changes between a control and an experimental cell population. The changes between control and experimental cohorts are often determined through fluorescently tagged antibodies that are specific for given proteins; and the fluorescence is examined by microscopy and/or high throughput screening using a flow cytometer[1,14].

Successful FCM experiments rely on the accuracy and resolution of the data analysis, e.g. the performance of the FCM software that provides quantitative outputs for large numbers of events[2]. In FCM analysis, an event is constituted by the cytometer's detection of fluorescence emission and/or light scatter signals from a single cell or particle that passes through the microfluidic flow chamber. With thousands of these events, individual measures of fluorescence, size and granularity are produced, and to add complexity, these measurements can be deliberately modified by a researcher through the instrument setup, which can be changed from run to run[15]. FCM analysis, thus, becomes a computational and statistical challenge that produces meaningful data only if the

Department of Bioengineering and QB3, UC Berkeley, Berkeley, CA 94720, USA. email: nathanwong@berkeley.edu; iconboy@berkeley.edu

analysis is adequate for the experimental complexity. Inherent in this requirement, the datasets that are produced with the conventional FCM software (FlowJo[3], Cytobank[16], OpenCyto[17], and Webflow[18]) are typically quite large, which complicates their interactive web analyses.

In this work we developed a new FCM software that facilitates the FCM data analysis, while maintaining the accuracy and resolution of the data. In fact, analysis of flow cytometry experiments, despite having tens of thousands of data points, can be performed and visualized on a mobile device. Importantly, while simplifying the data analysis and having the intuitive work flow, Freecyto preserves the key features of traditional FCM software, such as scatterplots (dotplots) of two different emission, histograms of a fluorescent emission measurement[14], the side-by-side comparison of the results between the control and experimental populations and gating on sub-populations of cells.

Similarly to FlowJo[3], Cytobank[16], OpenCyto[17], and Webflow[18], Freecyto supports machine learning applications, but it does not require the installation of specific software packages (often OS-dependent), a detailed understanding of the software workflow, or extra layers of complexity in displaying, interacting, and sharing the FCM analysis with other researchers. Additional features of Freecyto are robust data-management and data-sharing: Freecyto is built on a secure centralized database management system, allowing for data to be stored remotely and analyses to be shared and edited by anyone, yet it maintains the safeguard of proper permissions. Notably, the decisions on instrument settings (such as, changing the gain and signal intensity) and experimental set-ups (for instance, additional runs of certain cohorts) become better informed - based on real time user-friendly data analysis.

A key feature of Freecyto is the k-means clustering algorithm in which data points are clustered together into k clusters based on a Euclidean distance metric. This use of k-means algorithm as a method of data quantization is distinct from the flow cytometry studies, which use clustering algorithms to analyze the data[19–23]. Freecyto, in contrast, uses k-means to create a reduced, representative dataset of the original, so that the user can have much greater capability in analyzing the data, such as applying the stated clustering algorithms to the data. The original data is then reduced to the centers of the clusters, allowing the user to gate interactively on these centers. We show that FCM data analysis remains faithful when Freecyto is compared to the conventional FlowJo software.

By focusing and quantizing the data, Freecyto offers a better control over the analysis of FCM experiments, increasing the computational feasibility of any and particularly, very large datasets. Because of the high dimensional nature of flow cytometry data and the increasing technological developments in flow cytometers which have pushed the number of parameters and the sheer volume of data ever higher, there is a greater need for FCM software to handle increasingly large data sets[24,25]. Freecyto was developed to address this challenge.

Results

K-means quantization.

While the quick visualization capabilities are sufficient for most basic flow cytometry operations, a more detailed study may require additional specialized functions, such as sub-population gating and quadrant (coordinate-system) gating. Having data sets on the magnitude of 10^5 or 10^6 events, presents a significant challenge to interactively plot these on the web. In the case of gating, having tens of thousands of points that users can lasso-select on the web is virtually impossible for personal computers and standard web browsers. Freecyto solves this problem by introducing a k-means clustering algorithm for quantizing the input data (Fig. 1).

First, after running the k-means clustering algorithm, the centroids are used to construct a Voronoi diagram. Thus, the original dataset is partitioned into Voronoi cells, and each cell contains all the original points that belong to that cluster. Following, for each Voronoi cell, the variance is computed, with the centroid used as the mean of the geometric space. Finally, the within-cluster variance is plotted as a colormap within the Voronoi diagram to portray which cells contain more of the underlying variance, and the variance is summed up across all Voronoi cells to portray the elbow at which minimal within-cluster variance is lost with respect to the increase in computation power due to increasing the number of clusters.

K-means clustering (implemented with Lloyd's algorithm, clusters initialized with kmeans++ with a default seed) is an unsupervised machine-learning algorithm that is used to identify clusters of points based on each point's distance from the center of a proposed cluster. Freecyto runs this algorithm on the user-selected channels, identifying a pre-defined number of clusters, and storing only the centers of these clusters. The number of clusters is either user-selected (if running locally) or approximated automatically as a range between 250 and 5000 based on the size of the dataset. This simplifies the conventional k-clustering approach and enables future development of more suitable algorithms to determine k[27,28]. Freecyto's application of k-means clustering quantization vastly reduces the complexity of the flow cytometry data, without significant loss to the variability within the original dataset as we will show in the next section. The reduced dataset that is generated is highly suitable for downstream statistical analysis, such as hierarchical clustering or dimensionality reduction to identify sub-populations of cells (Supplemental Fig. 5).

Fidelity of data quantization in interactive analysis.

To quantitatively examine the quality of our reduced data set, we compute the mean-squared error (MSE) of each cluster (Fig. 2). For the k-means algorithm, this is equivalent to computing the within-cluster variance of each cluster, because the predicted cluster center is the mean of all points in that cluster. To visualize this, a toy dataset is randomly generated with spiral properties (Fig. 2A). The MSE of each cluster, as visualized by Voronoi cells (Fig. 2B), is then mapped to a color range to depict how faithfully each cluster center captures the other points in that cluster. In Fig. 2C, it is shown that with increasing k, the lower the MSE for each cluster. Finally, the average of all the MSE for all clusters is computed (Fig. 2D) to show that the data lost in each cluster center decreases rapidly in exchange for smaller increases in

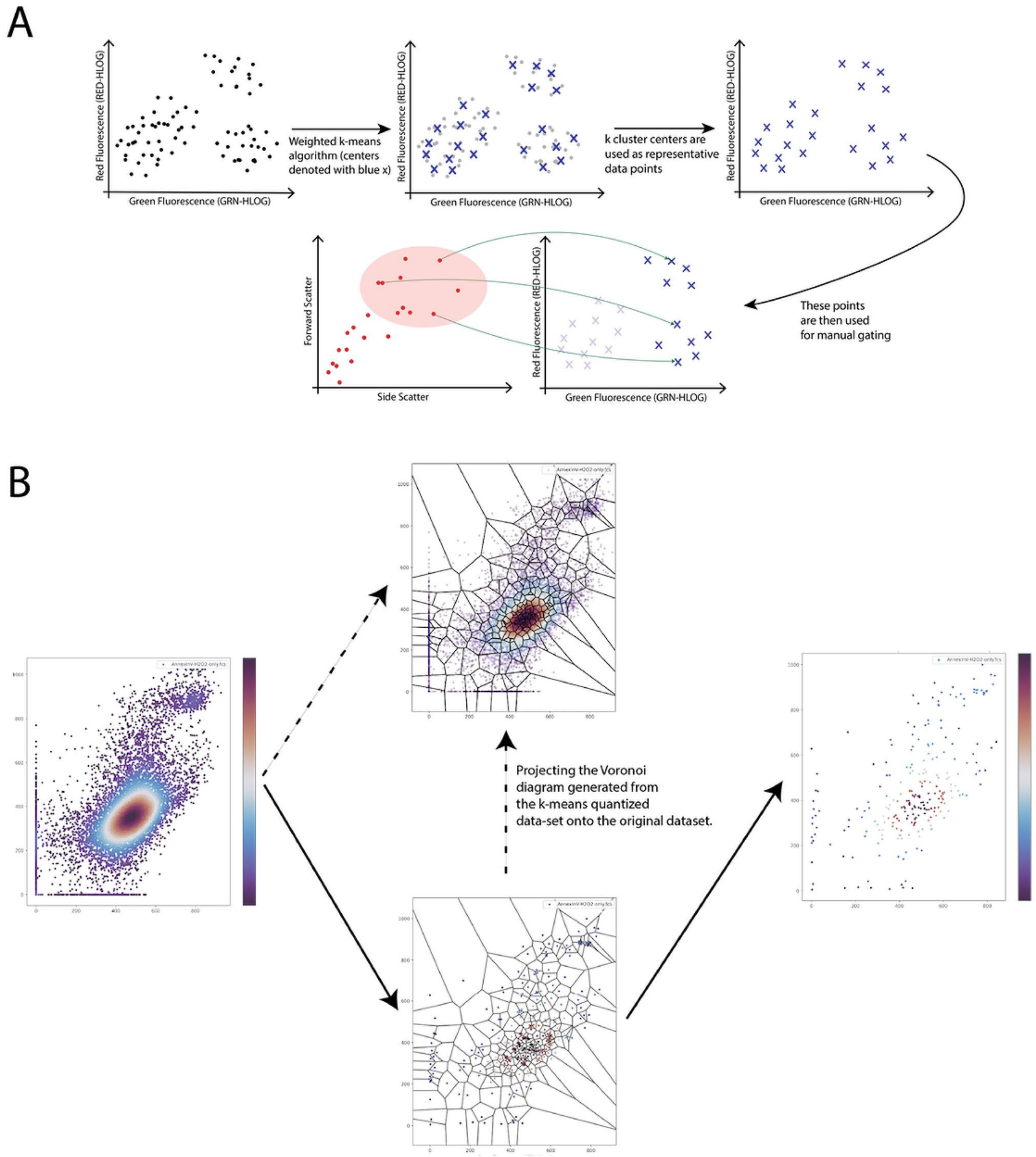
A
Red Fluorescence (RED-HLOG)
Green Fluorescence (GRN-HLOG)
Weighted k-means algorithm (centers denoted with blue x)
k cluster centers are used as representative data points
These points are then used for manual gating
Forward Scatter
Side Scatter
B
AnnexinV-H2O2-only.fcs
Projecting the Voronoi diagram generated from the k-means quantized data-set onto the original dataset.

Figure 1. K-means Workflow in Freecyto. (**A**) The process by which the original dataset is quantized, and how manual gating works on a shared data source. (**B**) The principles behind k-means quantization, and the Voronoi diagram computed from the reduced dataset projected on the original dataset. Note: Matplotlib[26] python library was used to generate 1B.

the number of clusters chosen. The same procedure is applied to a bi-modal distribution, with similar results, as shown by Figs. 2E–H.

The quantized data can then be plotted interactively through Bokeh on a webpage and downloaded as a SQL database within the web application. In this interactive analysis portion, each flow cytometry data file is treated as a shared data source, thus in Freecyto the user can lasso-select a sub-population of cells that are displayed in a scatterplot graph or a fluorescence channel and observe the quantized data for that sub-population of cells in the other FCM channel(s). This Freecyto feature allows the user to quickly and with more precision determine

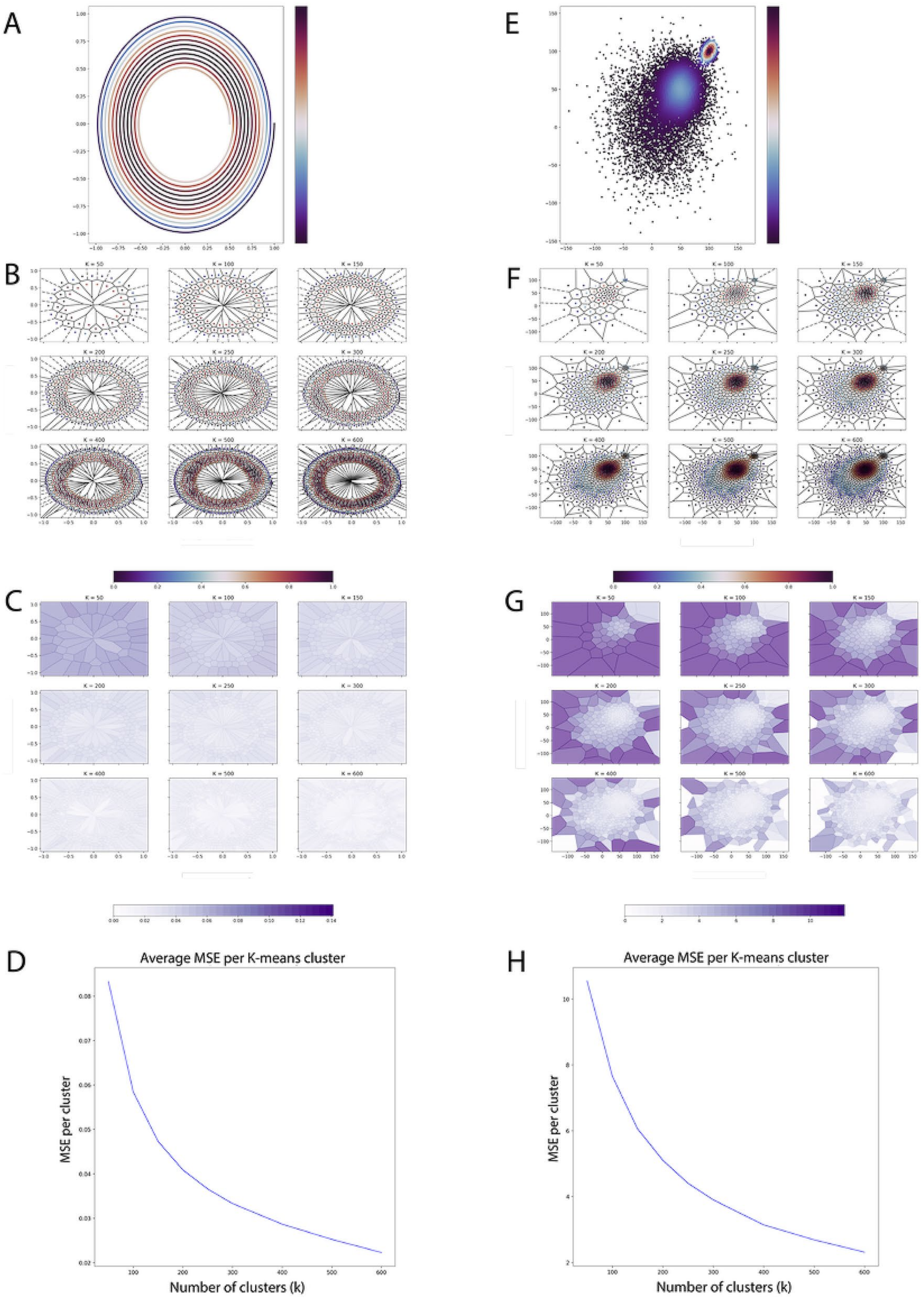

Figure 2. K-means within-cluster variance visualization of synthetic datasets. (**A**) Original spiral data (N = 5000). (**B**) Cluster centers with Voronoi cells outlined. (**C**) Within-cluster variance of each Voronoi cell with increasing k, and by extension, the MSE in each cluster identified by k-means. (**D**) Trend of increasing clusters and the average within-cluster variance of each cluster. (**E**) Original bimodal data (N = 10, 000). (**F**, **G**, **H**) Cluster centers and variance loss in each Voronoi cell with increasing k. Note: Matplotlib[26] python library was used to generate figure.

how the size of the cells or a signal for a specific marker (cell-fate protein, for example) is related to other markers (transgene expression, for instance) for each cell in the studied population. Demo: (3:07 – 6:20).

One key question is whether our method of k-means clustering qualitatively maintains the accuracy and resolution of the data. To address this, we compared side-by-side Freecyto and the conventional FCM software FlowJo in the analysis of GFP positive cells in a population and in studying cells in early and late stages of apoptosis (e.g. AnnexinV-7AAD and co-stain). Here we used Freecyto modality for such a common feature of FCM as a coordinate system gating to identify the percentage of cells located within certain thresholds. As shown in Figs. 3 and 4, Freecyto was as accurate as FlowJo in the resolution of these data sets, at the same time preserving the key features of FCM software, such as allowing the user to specify fluorescence thresholds and visualize and quantify the percentage of cells located in these quadrants (Figs. 3, 4).

Moreover, Freecyto generated quantized data points are stored in an SQLite database—essential to the deep gating tool. The deep gating tool allows the user to lasso-select a sub-population of cells and graphically display only the gated cells for all advanced analysis operations. This is useful in narrowing the analysis to specific sub-populations, as well as identifying outliers in the dataset. This deep-gating function can be applied as many times as needed, and all deep-gates can be reset by pressing the reset-gating button, after which the visualization and quantification of the results will reflect the original, unaltered dataset (Figs. 3, 4). Both the results of the k-means quantization and the sub-populations identified from manual gating can be downloaded directly in the application.

To comparatively analyze the accuracy and capabilities of Freecyto and FlowJo, WT and GFP+ cells were mixed at five different ratios, 100:0, 75:25, 50:50, 25:75, and 0:100, WT:GFP+; and run on Guava Easycyte Flow cytometer (Millipore-Sigma). The data was analyzed by FlowJo and Freecyto in parallel. As a result, the number of GFP positive cells increased linearly from 100:0 WT/GFP+ to 0:100 WT/GFP+, as expected, which was accurately detected by both FlowJo and Freecyto.

To compare Freecyto and Flowjo in another commonly analyzed by Flow Cytometry assay—cell apoptosis, IMR90 human fibroblasts were treated (or not) with hydrogen peroxide, H_2O_2, at 200 μM for 24 h to induce apoptosis. The cells were assayed with Annexin V and 7-AAD and run on the Guava Easycyte Flow cytometer (Millipore-Sigma). The results were analysed with Freecyto, yielding accurate and visually clear data. The negative control, isotype-matched IgG fluorescence was used to set up the quadrant, Fig. 4A. Early apoptotic cells positive for Annexin V can be seen in the top left quadrant and late apoptotic cells positive for both Annexin V and 7-AAD in the top right quadrant. As expected, Freecyto shows the number of Annexin V positive cells, Fig. 4B. The number of cells in early and late stages of apoptosis were increased with H_2O_2, as compared to the untreated control, Fig. 4C. In summary, the analysis of apoptosis (Annexin V and 7ADD assay) yields the predicted results and is as accurate and sensitive with Freecyto as it is with Flowjo. Finally, the data integrity of Freecyto's k-means downsampling with a high parameter color panel is also demonstrated against no downsampling in Supplemental Fig. 6.

Web (Uwsgi-flask-nginx) application to allow platform-agnostic, mobile-ready access to flow cytometry analysis.

Several core technologies are deeply integrated into Freecyto in order to allow seamless processing and visualization of flow cytometry data. Chiefly, the integration of these technologies allows for robust storage of user data, high-throughput handling of the data, e.g. processing operations, and interactivity of the data visualizations.

Computationally expensive operations in flow cytometry, including reading and parsing data, performing visualizations, and obtaining sample statistics, are all performed server-side in Freecyto. Freecyto is hosted as a Python-flask-uwsgi-nginx application on a Digital Ocean server.

While most flow cytometry tools have unique requirements depending on the user's operating system (OS), application dependencies (a specific version of python packages), or computational resources (i.e. four CPU cores), Freecyto can be accessed without platform restrictions and dependencies. This application also is designed to be mobile-compatible, allowing users to access their flow cytometry analysis and also perform new flow cytometry analysis directly on their mobile devices (Fig. 5).

In addition, Freecyto can be downloaded as a Flask application (open-source), so that users can install the appropriate dependencies and run the application on a local intranet (useful if users desire a stricter control of Flow cytometry data privacy). This also allows for greater control over default parameters and application modules, such changing the number of reduced data points used in interactive analysis and implementing a clustering model on top of the reduced data set (Fig. 5).

Demo: (0:00–1:00).

Parallel processing (multiprocessing) of computationally intensive analysis functions.

Freecyto integrates advances in multiprocessing functionality in order to speed up traditionally expensive FCM data analysis operations. Multiprocessing is implemented when users upload multiple files, when visualizations are performed, and when the k-means algorithm is running. These operations are asynchronously performed on the server-side, speeding up the time it takes for the user to receive analyses outputs from their data by an order of magnitude. Through the implementation of this multiprocessing a side-by-side over five files upload becomes possible (Supplemental Fig. 3).

User data management and authentication.

Google Firestore/Datastore is integrated to store references to previously performed visualization operations. For example, the images that are generated from an experimental upload are stored in a unique directory on the server, and the references to the generated images are stored in a collection as a unique entry under the user account in Google Firestore. This prevents redundant

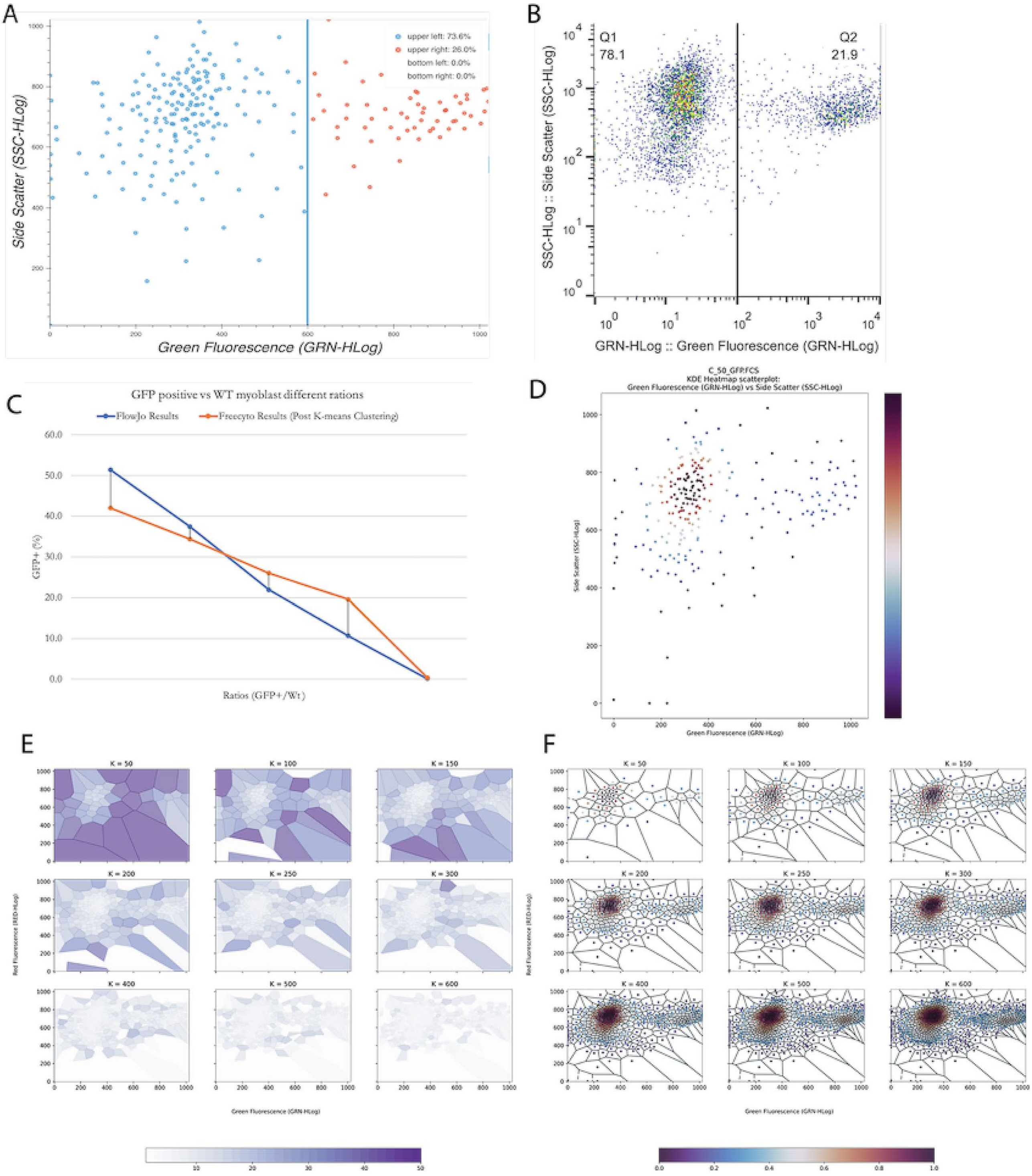

Figure 3. Analysis of GFP positive and negative cell populations. (**A**) 50:50 GFP transgenic cells ratios with the coordinates gated by Freecyto (after quantization). (**B**) The same 50:50 GFP transgenic cell ratios with the coordinates gated by FlowJo[3]. (**C**) Compares Freecyto and FlowJo measurements of GFP+ cells for 100:0, 75:25, 50:50, 25:75, and 0:100 ratios. (**D**) Density plot created by Freecyto which outlines the density of cells after the k-means quantization is performed with 250 clusters. (**E**) MSE of each cluster with varying svalues of k. (**F**) The resulting density plot with varying values of k. Note: the Bokeh[29] library was used to generate figure (**A**). Microsoft Excel[30] was used to generate figure (**C**). Matplotlib[26] python library was used to generate (**D–F**).

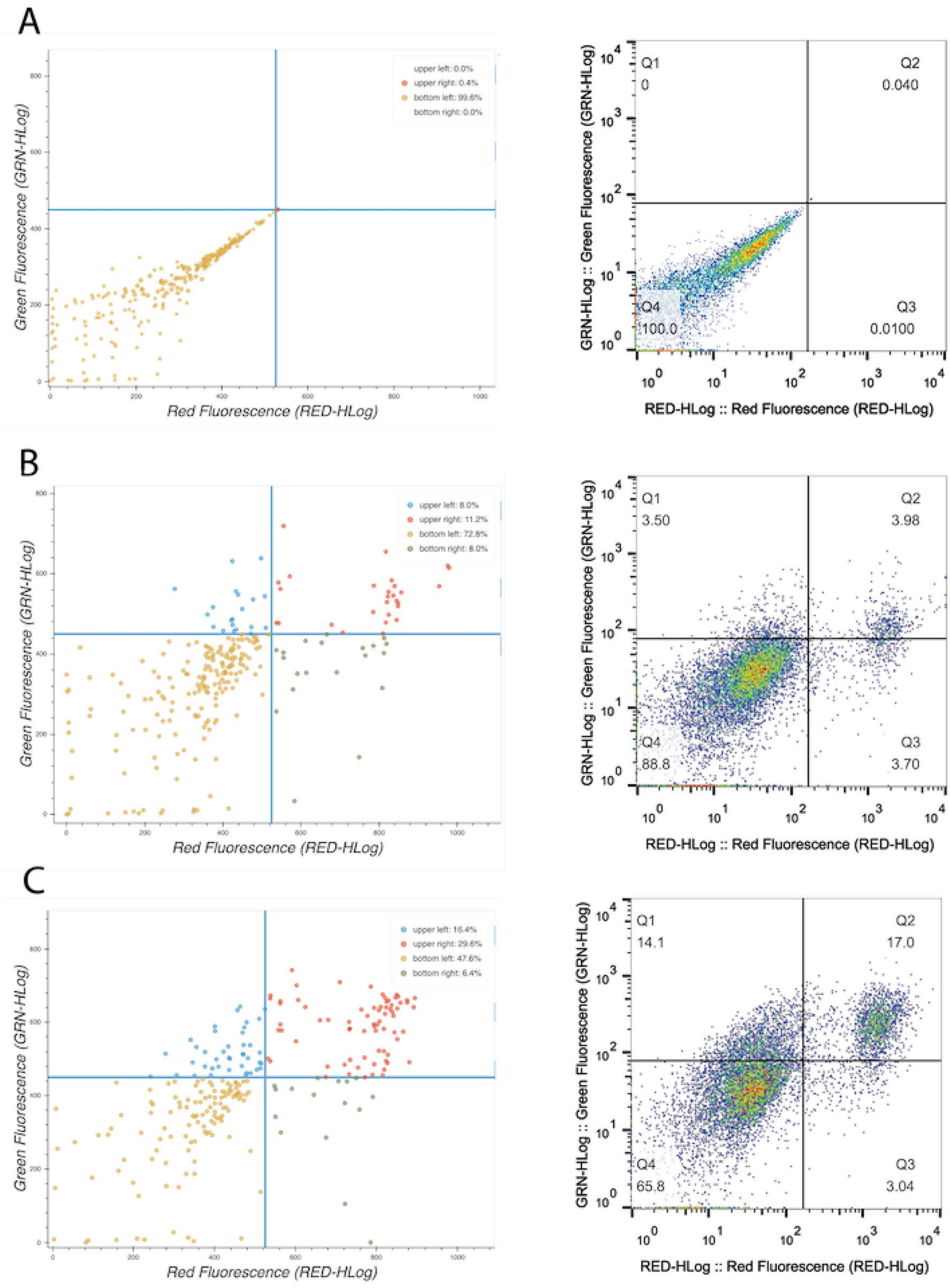

Figure 4. Analysis of apoptosis. IMR90 cells were treated with hydrogen peroxide, H_2O_2, at 200 μM for 24 h to induce apoptosis. The cells were then stained with Annexin V and 7-AAD. Early apoptotic cells are positive for Annexin V and are seen in the top left quadrant (Q1) and late apoptotic cells, which are positive for both annexin and 7-AAD are seen in the top right quadrant (Q2). Live cells are negative for both stains (Q4). (**A**) Negative control: Isotype-matched IgG staining (1st antibody) + secondary (FITC). (**B**) Untreated group. (**C**) H_2O_2 treatment group. Note: the Bokeh[29] library was used to generate figure (**A**–**C**).

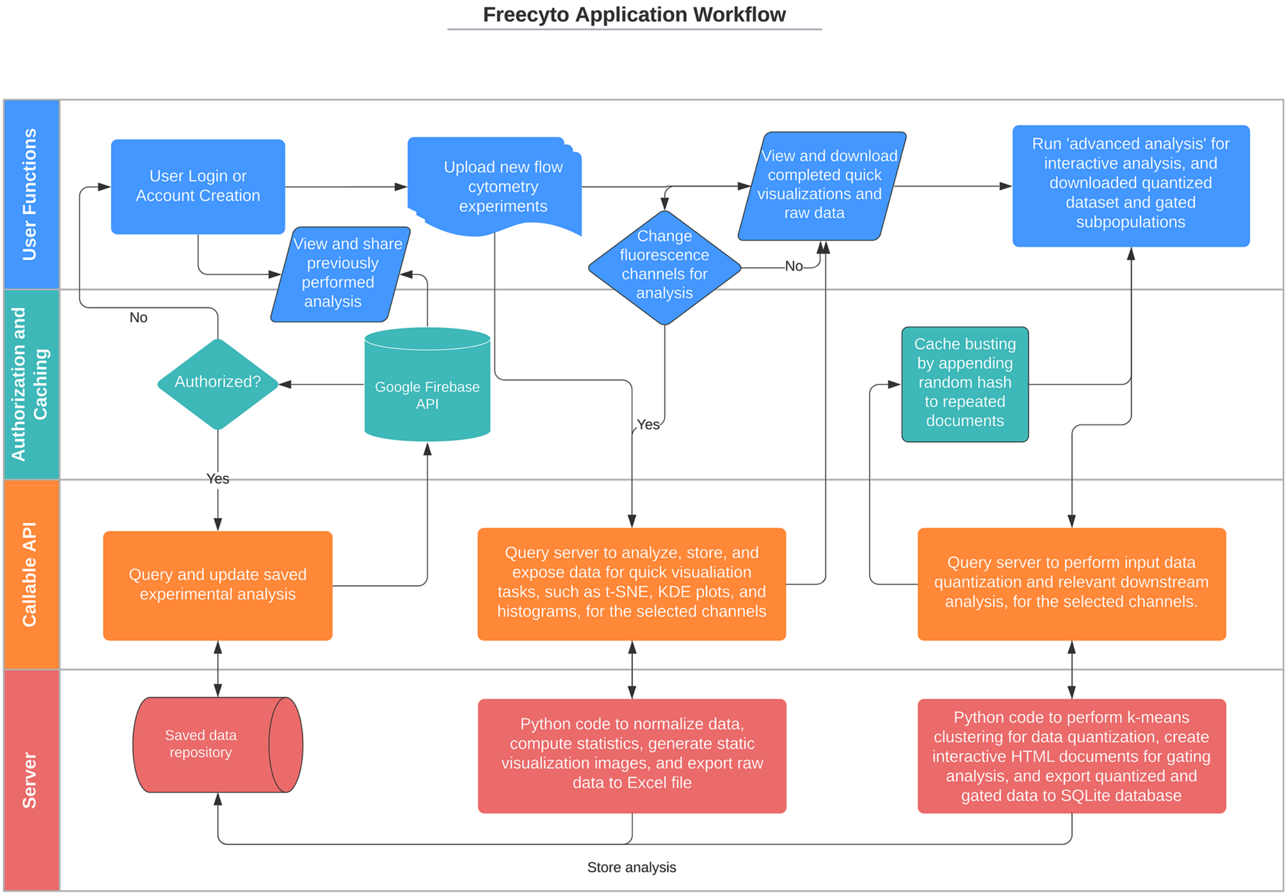
Freecyto Application Workflow
User Functions
Authorization and Caching
Callable API
Server
User Login or Account Creation
Upload new flow cytometry experiments
View and download completed quick visualizations and raw data
Run 'advanced analysis' for interactive analysis, and downloaded quantized dataset and gated subpopulations
View and share previously performed analysis
Change fluorescence channels for analysis
No
No
Authorized?
Google Firebase API
Yes
Cache busting by appending random hash to repeated documents
Yes
Query and update saved experimental analysis
Query server to analyze, store, and expose data for quick visualization tasks, such as t-SNE, KDE plots, and histograms, for the selected channels
Query server to perform input data quantization and relevant downstream analysis, for the selected channels.
Saved data repository
Python code to normalize data, compute statistics, generate static visualization images, and export raw data to Excel file
Python code to perform k-means clustering for data quantization, create interactive HTML documents for gating analysis, and export quantized and gated data to SQLite database
Store analysis

Figure 5. Freecyto Application Workflow. Created in Lucidchart, www.lucidchart.com.

analysis operations (i.e. the user uploads the same experimental files), yet, it allows the user to access the previously performed operation. A sortable table of previously performed experiments (50 most recent) are listed in the user home page, allowing the user to easily access previously analysed flow cytometry results.

Firebase and Google identity platform: Google and Email logins are enabled, allowing the user to create and access their user account with these authentication methods. This prevents unauthorized usage of the application, requiring the user to create an account before accessing the analysis toolkit. To promote scientific knowledge and collaborations, sharing the results of a flow cytometry experiment on Freecyto merely requires sharing the URL of the experiment. Demo: (1:00–1:30).

Side-by-side experiment comparisons (multiple file upload).

Freecyto supports user upload of multiple flow cytometry files as a result of the multiprocessing pipeline. For normalization of the raw input files, the user may select hyperlog, logicle, or no transformation to be applied. Logicle and hyperlog transformations normalize the flow cytometry data by transforming most events (including negatively measured values) to a normalized fluorescence value of between 0 and 1[31]. This improves on traditional free flow cytometry analysis applications, which limit the user to uploading only a single flow cytometry file at a time, though many flow cytometry experiments have anywhere from 2 to 10+ files to analyse. Freecyto's approach allows the user to upload numerous files concurrently, enabling plots to be overlaid for easy and clearly visualized comparison between the datasets. In another feature of Freecyto, if overlays make it harder to discern the individual plots, then individual files can also be graphed and visualized. Demo: (1:30–2:00).

Quick visualization capabilities.

Freecyto is built on the principle that FCM analysis should be easy to perform and that real-time data processing expands the research capabilities in acutely and accurately modulating the FCM experiments. Freecyto's pipeline achieves this by quick visualization of the scatterplots, density-estimation plots, histograms, box-whisker diagrams, and correlation tables, which are generated by Freecyto based on the selected fluorescence channels. In addition, t-SNE plots allow users to visualize segregating features of the data. The images and relevant statistics are displayed through a carousel slider (Siema) and a table respectively.

It is integral to flow cytometry analysis to allow users to select the fluorescence channels they wish to visualize. Freecyto accomplishes this with a simple checkbox list of all possible channels. The user selects the channels they wish to visualize, presses "submit," and the images automatically update to match the desired fluorescence channels to visualize. This pipeline is designed to be minimalistic—it allows the user to quickly determine how

		Freecyto	FlowJo[3]	Cytobank[16]	AutoGate[32]
Features		Version 1.0	Version 10.7	Version 8.0	Version 4.521
General software characteristics	Latest release year	2020	2020	2020	2019
	Windows OS	Web-based	Vista/10	Web-based	Vista/10
	Compatible with Mac OS	Web-based	✓	Web-based	✓
	Recommended memory	Web-based	8GB	Web-based	8GB
	Required CPU cores	Web-based	2 cores	Web-based	2 cores
	Tutorial film available	✓	✓	✓	×
	Multiple file upload	✓	✓	✓	✓
	Drag-and-drop file upload	✓	✓	✓	×
	Open-source code available	✓	×	×	×
	Free to use	✓	×	×	Conditionally
	Share with collaborators	✓	×	✓	×
Graphical data representation	Histograms	✓	✓	✓	×
	Dotplots	✓	✓	✓	✓
	Density dotplot	✓	✓	✓	✓
	Dimensionality reduction plots	✓	✓	✓	×
	Correlation Heatmaps	✓	✓	✓	×
	Overlayed plots	✓	✓	✓	×
	Interactive Lasso Gating	✓	✓	✓	✓
	Coordinate Gating	✓	✓	✓	✓
	Automated Gating	×	Semi-automated	✓	✓
Analysis Tools	Direct export to Excel	✓	✓	✓	×
	Export raw gated cells	✓	✓	✓	×
	Save previous experiments	✓	✓	✓	✓

Table 1. Comparing Freecyto with other flow cytometry applications.

their data looks, offering enough modularity to facilitate the most common flow cytometry analysis operations. In addition, the converted flow cytometry data can be downloaded as an Excel spreadsheet. Demo: (2:00–3:07).

Discussion

Freecyto was developed as a new data processing software for Flow Cytometry data and validated for enhancing the speed, convenience, and machine learning capacity of the FCM data analysis, while preserving the accuracy. These features were validated in key FCM set-ups of studying sub-populations with variable expression of a transgene, and in viability-apoptosis studies. Summarily, the use of our weighted k-means clustering algorithm innovated FCM data analysis and transformed it into an online platform. It is important to note that Freecyto's demo server implementation may be slow at times, due to limited budget and computing resource constraints (running on 2GB RAM). File size and speed are limited by the cost and quality of the web server for demo purposes. For practical application, users can deploy the code locally, add additional plugins and improvements, and allocate greater resources to fit their individual flow cytometry needs.

Freecyto offers the necessary features to perform typical FCM analyses, in addition to providing the user interactive analysis of the data and it fills a niche when compared with other FCM software (Table 1). Freecyto is an open-source, flexible platform that allows modifications. For example, Opencyto allows users to create automated gating pipelines in R which may solve the subjectivity and time-consuming nature of manual gating and such a feature is very compatible to build on top of Freecyto's existing framework[17]. Freecyto does not innovate the existing flow cytometry analysis, instead it innovates the approach to such analyses, thereby improving on the ease and accessibility of FCM data, while also providing greater flexibility and control in gating large datasets, through the quantizing of the data with a weighted k-means clustering algorithm. We use a modified form of k-means (biased k-means), and importantly, as far as we are aware, the visualization methods used to portray the effectiveness of k-means have not been performed for flow cytometry data in published literature.

The goal of Freecyto is in introducing its k-means downsampling and further visualization, as a conceptual demonstration to allow big flow cytometry data sets to be displayed interactively on the web. It is a proof of concept research with open-source code implementation—certainly not a complete answer to solving flow cytometry on the web. Doing so would be out of scope of this research manuscript, but a broader outcome for FACS IT, which can likely result from a follow up project that can be accomplished with greater resources.

Conclusions

FCM analysis is essential for a broad range of biomedical studies, many of which are directly and critically important for human health. Freecyto allows for the streamlined interactive analysis of FCM datasets in addition to multiple FCM experiments in parallel, harnessing the transmissibility of the internet to power and serve its analytical platform. Whereas many FCM analysis packages are expensive or require software/OS dependencies, Freecyto is free, open-sourced, and web-based. While simplifying FCM studies, Freecyto improves the processing of high-volume data and facilitates the real-time data analysis.

As flow cytometry development continues to improve, the need for indexing and manipulating large quantities of scientific data cannot be understated. Freecyto integrates state-of-the-art data storing and indexing features with Google Cloud, creating an interface for users to have greater confidence and connectivity with their flow cytometry data. In this regard, Freecyto's k-means quantization approach might be broadly useful and important not only in FCM, but more broadly, for Big Data analysis in omics, medical data for machine learning and AI, computer vision, environmental engineering, etc. large data realms.

Materials and methods

Data visualization.

Several Python packages were used in creating this application. Flask was used to serve the web application. Google Identity (Firebase) was used to authenticate users, and Google DataStore was used to store references to previously performed experiments. Pandas, NumPy, FlowUtils, and Cytoflow were used to dynamically store and transform the raw flow cytometry data. Matplotlib, Seaborn, and Pandas were used to generate images of scatterplots, box-plots, heatmaps, and histograms. The t-distributed stochastic neighbour embedding (t-SNE) projection was performed with Scikit-learn (sklearn) with perplexity of 40. For the interactive analysis, sklearn was used for the weighted k-means clustering. SQLite3 was used to store clustered data. Bokeh and Holoviews were used to display the interactive graphs. HTML5UP and Creative Tim Light Bootstrap Theme inspired the front-end template design of the web application.

Multiprocessing.

Multiprocessing, assuming a multi-core machine, was implemented to speed up the data visualization algorithms. Chiefly, the results of a benchmark test on a quad-core, 8 GB RAM, 2.3 Ghz MacBook Pro are reported below for the static image visualizations, and for the interactive data analysis portions.

Weighted K-means algorithm.

$X = \{x_1, x_2, \ldots, x_n\}$ such that every x_i has d dimensions. Let Ω be a diagonal d x d matrix such that the diagonal entries are the weights of each dimension. k is the number of clusters we want to find. S is the set of all k clusters such that $S = \{S_1, S_2, \ldots, S_k\}$. We want to minimize the loss function:

$$\arg\min_S \sum_{i=1}^{k} \sum_{x \in S_i} (x - \mu_i)^T \Omega (x - \mu_i)$$

In the default case, let the diagonal entries of Ω be 1 if the corresponding channel was selected for visualization, and 0 otherwise.

Voronoi diagram algorithm.

$X = \{x_1, x_2, \ldots, x_n\}$ such that every x_i has d dimensions. R is the set of all k Voronoi diagrams such that $R = \{R_1, R_2, \ldots, R_k\}$ and S is the set of all k clusters such that $S = \{S_1, S_2, \ldots, S_k\}$. d is a distance metric, for which we used Euclidean distance. We want to find the region such that every point in the region is closest to the set of points described by the k-means clustering.

$$R_k = \{x \in X | d(x, S_k) \leq d(x, S_j) \forall j \neq k\}$$

Or equivalently, because the distance of every point x in S_k to it's mean centroid μ_k has already been minimized in the converged k-means algorithm:

$$\forall x \in S_k | d(x, S_k) \leq d(x, S_j)$$
$$\forall j \neq k \implies R_k = \{x \in S_k\}$$

Web application (open-source) licenses.

- Advanced Analysis: Light bootstrap theme by Creative Tim: MIT License https://github.com/timcreative/freebies/blob/master/LICENSE.md
- Lens by HTML5UP: Creative Commons 3.0 https://html5up.net/license
- NumPy: https://github.com/numpy/numpy/blob/master/LICENSE.txt
- SciPy: https://scipy.org/scipylib/license.html
- Scikit-learn: https://scikit-learn.org/stable/
- Pandas: https://github.com/pandas-dev/pandas/blob/master/LICENSE
- Matplotlib: https://matplotlib.org/users/license.html
- Bokeh: https://github.com/bokeh/bokeh/blob/master/LICENSE.txt
- Holoviews: https://github.com/pyviz/holoviews/blob/master/LICENSE.txt
- Flask: http://flask.pocoo.org/docs/1.0/license/
- SQLAlchemy: https://docs.sqlalchemy.org/en/latest/copyright.html
- Cytoflow: https://github.com/bpteague/cytoflow/blob/master/LICENSE.txt
- FlowUtils: https://github.com/whitews/FlowUtils/blob/master/LICENSE

Myoblast cultures.

Transgenic GFP+ and WT (C57.B6) mouse myoblasts were cultured in growth medium: Ham's F10, 20% Bovine Growth Serum and 5 ng/ml bFGF on 1 $\mu g/cm^2$ Matrigel. Cells were washed and detached with PBS (three 37C) and were pelleted by centrifugation. Cells were pelleted and counted using a hemocytometer.

Cell culture and apoptotic assay. Normal human lung fibroblast cells (IMR-90) were obtained from ATCC #CCL-186. Cells were maintained in DMEM (Dulbecco's Modified Eagle Medium) supplemented with 10% fetal calf serum (FCS, Hyclone) containing 1% penicillin-streptomycin (Invitrogen) and maintained in a humid atmosphere at 37°C containing 5% CO_2. When cells were grown to 70% confluence, they were subcultured at $\frac{1}{5}$ dilution for later passaging.

The apoptotic assay of IMR90 was conducted by Apoptosis Detection Kit (ab214663, Abcam) according to the manufacturer's protocol. Briefly, cells were detached using 0.05% trypsin and washed twice with PBS. Then, samples were resuspended in 1x annexin-binding buffer and incubated with 5 μL Annexin V-FITC and 5 μL 7-amino-actinomycin D (7-AAD) for 15 min at 37°C, avoiding light. Finally, events were acquired with a Guava Easycyte Flow cytometer (Millipore-Sigma) and analysed by Freecyto and Flowjo software individually to quantify the distribution of cells.

Data availability

The datasets generated and/or analysed during the current study are available in the Freecyto Github repository, https://github.com/nathan2wong/freecyto/tree/master/datasets; Project name: Freecyto; Project homepage: https://freecyto.com; Demo: https://youtu.be/JlIVgxh4_YA; Archived version: https://github.com/nathan2wong/freecyto; Operating system(s): Platform independent; Programming Language: Python, JavaScript; Other requirements: Listed on GitHub; License: BSD3; Any restrictions to use by non-academics: License Needed.

Received: 11 December 2020; Accepted: 3 March 2021
Published online: 24 March 2021

References

1. O'Neill, K., Aghaeepour, N., Špidlen, J. & Brinkman, R. Flow cytometry bioinformatics. *PLoS Computational Biology* **9**, e1003365. https://doi.org/10.1371/journal.pcbi.1003365 (2013).
2. Lugli, E., Roederer, M. & Cossarizza, A. Data analysis in flow cytometry: The future just started. *Cytometry Part A* **77A**, 705–713. https://doi.org/10.1002/cyto.a.20901 (2010).
3. FlowjoTM software. *[software application]* (2019).
4. Ramel, S. *et al.* Evaluation of p53 protein expression in barrett's esophagus by two-parameter flow cytometry. *Gastroenterology* **102**, 1220–1228. https://doi.org/10.1016/0016-5085(92)70016-5 (1992).
5. Leith, C. *et al.* Correlation of multidrug resistance (MDR1) protein expression with functional dye/drug efflux in acute myeloid leukemia by multiparameter flow cytometry: identification of discordant MDR-/efflux+ and MDR1+/efflux- cases. *Blood* **86**, 2329–2342. https://doi.org/10.1182/blood.V86.6.2329.bloodjournal8662329 (1995).
6. Rosner, M., Schipany, K. & Hengstschläger, M. Merging high-quality biochemical fractionation with a refined flow cytometry approach to monitor nucleocytoplasmic protein expression throughout the unperturbed mammalian cell cycle. *Nature Protocols* **8**, 602–626. https://doi.org/10.1038/nprot.2013.011 (2013).
7. Darzynkiewicz, Z. *et al.* Features of apoptotic cells measured by flow cytometry. *Cytometry* **13**, 795–808. https://doi.org/10.1002/cyto.990130802 (1992).
8. Barlogie, B. *et al.* Flow cytometry in clinical cancer research. *Cancer Research* **43**, 3982–3997 (1983).
9. Keyes, T. J., Domizi, P., Lo, Y.-C., Nolan, G. P. & Davis, K. L. A cancer biologist's primer on machine learning applications in high-dimensional cytometry. *Cytometry Part A* **97**, 782–799. https://doi.org/10.1002/cyto.a.24158 (2020).
10. Brando, B. *et al.* Cytofluorometric methods for assessing absolute numbers of cell subsets in blood. *Cytometry* **42**, 327–346. https://doi.org/10.1002/1097-0320(20001215)42:6<327::AID-CYTO1000>3.0.CO;2-F (2000).
11. Lugli, E., Troiano, L. & Cossarizza, A. Investigating t cells by polychromatic flow cytometry. *Methods in molecular biology (Clifton, N.J.)* **514**, 47–63. https://doi.org/10.1007/978-1-60327-527-9_5 (2009).
12. Benedek, G., Meza-Romero, R., Bourdette, D. & Vandenbark, A. A. The use of flow cytometry to assess a novel drug efficacy in multiple sclerosis. *Metabolic Brain Disease* **30**, 877–884. https://doi.org/10.1007/s11011-014-9634-0 (2014).
13. Hu, W. *et al.* RNA-directed gene editing specifically eradicates latent and prevents new HIV-1 infection. *Proceedings of the National Academy of Sciences* **111**, 11461–11466. https://doi.org/10.1073/pnas.1405186111 (2014).
14. McKinnon, K. M. Flow cytometry: An overview. *Current Protocols in Immunology* **120**, https://doi.org/10.1002/cpim.40 (2018).
15. Maecker, H. T. & Trotter, J. Flow cytometry controls, instrument setup, and the determination of positivity. *Cytometry Part A* **69A**, 1037–1042. https://doi.org/10.1002/cyto.a.20333 (2006).
16. Kotecha, N., Krutzik, P. O. & Irish, J. M. Web-based analysis and publication of flow cytometry experiments. *Current Protocols in Cytometry* **53**, 10.17.1-10.17.24. https://doi.org/10.1002/0471142956.cy1017s53 (2010).
17. Finak, G. *et al.* OpenCyto: An open source infrastructure for scalable, robust, reproducible, and automated, end-to-end flow cytometry data analysis. *PLoS Computational Biology* **10**, e1003806. https://doi.org/10.1371/journal.pcbi.1003806 (2014).
18. Hammer, M. M., Kotecha, N., Irish, J. M., Nolan, G. P. & Krutzik, P. O. WebFlow: A software package for high-throughput analysis of flow cytometry data. *ASSAY and Drug Development Technologies* **7**, 44–55. https://doi.org/10.1089/adt.2008.174 (2009).
19. Murphy, R. F. Automated identification of subpopulations in flow cytometric list mode data using cluster analysis. *Cytometry* **6**, 302–309. https://doi.org/10.1002/cyto.990060405 (1985).
20. Bruggner, R. V., Bodenmiller, B., Dill, D. L., Tibshirani, R. J. & Nolan, G. P. Automated identification of stratifying signatures in cellular subpopulations. *Proceedings of the National Academy of Sciences* **111**, E2770–E2777. https://doi.org/10.1073/pnas.1408792111 (2014).
21. Ye, X. & Ho, J. W. K. Ultrafast clustering of single-cell flow cytometry data using FlowGrid. *BMC Systems Biology* **13**, https://doi.org/10.1186/s12918-019-0690-2 (2019).
22. Ge, Y. & Sealfon, S. C. flowPeaks: a fast unsupervised clustering for flow cytometry data via k-means and density peak finding. *Bioinformatics* **28**, 2052–2058. https://doi.org/10.1093/bioinformatics/bts300 (2012).
23. Dorfman, D. M., LaPlante, C. D. & Li, B. FLOCK cluster analysis of plasma cell flow cytometry data predicts bone marrow involvement by plasma cell neoplasia. *Leukemia Research* **48**, 40–45. https://doi.org/10.1016/j.leukres.2016.07.003 (2016).
24. Bendall, S. C. *et al.* Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science* **332**, 687–696. https://doi.org/10.1126/science.1198704 (2011).
25. Mair, F. *et al.* The end of gating? an introduction to automated analysis of high dimensional cytometry data. *European Journal of Immunology* **46**, 34–43. https://doi.org/10.1002/eji.201545774 (2015).

26. Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering* **9**, 90–95. https://doi.org/10.1109/MCSE.2007.55 (2007).
27. Yuan, C. & Yang, H. Research on k-value selection method of k-means clustering algorithm. *J* **2**, 226–235. https://doi.org/10.3390/j2020016 (2019).
28. Pham, D. T., Dimov, S. S. & Nguyen, C. D. Selection of k in k-means clustering. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* **219**, 103–119. https://doi.org/10.1243/095440605x8298 (2005).
29. Bokeh Development Team. *Bokeh: Python library for interactive visualization* (2018).
30. Microsoft Corporation. Microsoft excel.
31. Bagwell, C. B. Hyperlog?a flexible log-like transform for negative, zero, and positive valued data. *Cytometry Part A* **64A**, 34–42. https://doi.org/10.1002/cyto.a.20114 (2005).
32. Meehan, S. *et al.* Autogate: automating analysis of flow cytometry data. *Immunologic Research* **58**, 218–223. https://doi.org/10.1007/s12026-014-8519-y (2014).
33. Moon, K. R. *et al.* Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology* **37**, 1482–1492. https://doi.org/10.1038/s41587-019-0336-3 (2019).
34. Spidlen, J., Breuer, K., Rosenberg, C., Kotecha, N. & Brinkman, R. R. Flowrepository: A resource of annotated flow cytometry datasets associated with peer-reviewed publications. *Cytometry Part A* **81A**, 727–731. https://doi.org/10.1002/cyto.a.22106 (2012).

Acknowledgements

We would like to thank Alex Park for providing technical help with these studies, and Michael Conboy for the helpful suggestions on the work and the manuscript.

Author contributions

NW created the Freecyto software and wrote the manuscript. ZR provided figures, data, and analyses of the GFP Flow Cytometry (Fig. 3). DK provided figures, data, and analyses of the cell apoptosis Flow Cytometry (Fig. 4). CH provided Fig. 1A, Table 1, and contributed code for downstream analysis in the Freecyto software. IC co-wrote the manuscript and designed the study. All authors read and approved the final manuscript.

Funding

This work was supported by NIH R01 EB023776, R01 HL139605 and Open Philanthropy awards to IC, and the funds were used to support the study.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-021-86015-6.

Correspondence and requests for materials should be addressed to N.W. or I.M.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.