Computational sorting with HyperFinder, FlowJo™ Software and BD FACSDiva™ Software

Background

When a researcher performs a multicolor flow cytometry experiment and explores the data using machine learning tools (e.g., clustering, t-SNE, UMAP, CITRUS, CellCNN and others), instead of manual gating, it may lead to the discovery of a novel cell type with a distinct pattern of surface markers.[1] However, the gating strategy for that cell population may be unknown. Without this critical information, this novel cell population cannot be easily sorted for further investigation, as current commercial cell sorters require sort decisions to be made by a series of 1-D or 2-D gates. This presents a significant gap between the multidimensional exploratory analysis and gate-based sorting, a gap both conceptual and technical.

In principle, this problem has a simple computational solution: create a set of all possible gating strategies, then find the one that best captures the target population. This goodness of fit of a gating strategy can be quantified through yield (the percentage of target cells that are actually collected in the last gate) and purity (the percentage of target cells in the last gate that actually belong to the population of interest).

For the sake of simplicity, those metrics can be combined into a single F-score: the harmonic mean of yield and purity. Harmonic mean acts as a soft AND-gate so if either of the values are close to 0, their harmonic mean is going to be close to 0.

Figure 1. Comparison of arithmetic mean and harmonic mean as summary metrics of classification purity and yield

A. Arithmetic mean

$$F = \frac{purity + yield}{2}$$

B. Harmonic mean (F-score)

$$F = \frac{2}{\frac{1}{purity} + \frac{1}{yield}}$$
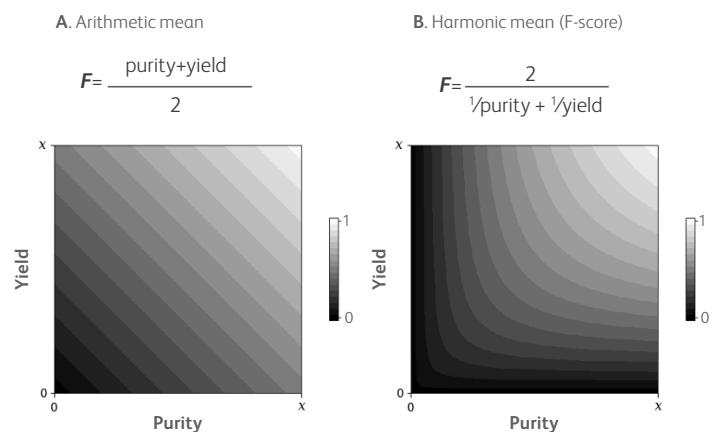
Finding a gating strategy that has the highest F-score is logically simple. However, because the number of possible gating strategies is very large, it is impractical to screen each one. For example, a sample measured on N channels can be plotted on N*(N-1)/2 bi-axial plots so for a 30-marker panel, you would need to look at 435 plots. Assuming that eight gates is enough to capture the population of interest, the number of possible combinations of eight gates is $^{8}c_{435} \approx 2.98 \times 10^{16}$. Furthermore, there are variations on gate size and shape that further increase the number of combinations. Therefore, a practical computational approach must either dramatically reduce the search space (e.g., by only allowing rectangular gates) or employ a combination of heuristics and optimization methods to find an optimal solution.

Several algorithms have been created in the past to solve this problem. GateFinder[2] is a pioneering algorithm that enumerates all possible biaxial plots, removes outliers and then draws a convex gate around the population of interest, creating a set of candidate gates. It then builds a sequence of convex gates in a linear fashion with each iteration appending a candidate gate that best improves the gating strategy. This algorithm is effective on low-dimensional datasets, but as the number of parameters grows, it tends to get stuck in local optima, thus producing gating strategies that have much lower F-scores. Another notable algorithm, Hypergate[3] takes a completely different approach. Instead of growing the gating strategy one gate at the time, it fits a bounding hyperrectangle (a multidimensional box) around the population of interest and then uses the box coordinates to create a set of rectangular gates. Since the classification boundaries are perpendicular (independent) in each dimension, it greatly reduces the search space, allowing the algorithm to more easily converge on an optimal set of rectangular gates. However, there are real-world cases when a population of interest simply cannot be properly captured by a rectangular gate, which would violate Hypergate's assumptions and cause it to fail. A major limitation of both algorithms is their requirement for R programming knowledge, which limits ease of use. In addition, the gate output is in a format that cannot be readily imported by any FACS instrument software (such as BD FACSDiva™ Software). Instead, the user must manually re-draw the gates.

We have addressed those shortcomings by creating HyperFinder, a novel algorithm that combines the core principles of GateFinder and Hypergate to optimize gating strategies. HyperFinder takes an input of cells labeled as positive and negative and tries to find a gating strategy that captures the maximum number of positives (maximizing yield) while removing as many negatives as possible (maximizing purity). To balance both parameters, HyperFinder uses an Fβ score, which is a weighted harmonic mean of yield and purity, with the beta parameter allowing the user to bias the gating strategy search toward one or the other. By default, the weight is equal to 1 (the resulting value is called F_1-score), where purity and yield have equal contributions. For instance, setting beta to 2 (F_2) will lead to a gating strategy that emphasizes yield over purity, while setting it to 0.5 ($F_{0.5}$) will similarly emphasize purity over yield within the gating strategy.

Figure 2. Comparison of algorithm performance on nine different datasets

HyperFinder(rect) corresponds to the hyper-rectangle initialization mode. HyperFinder(poly) corresponds to the polygon initialization mode. **A**. The F_1-score performance of HyperFinder to GateFinder and Hypergate algorithms on various hand-gated and clustered cell populations shows that HyperFinder outperforms the other two algorithms in most cases. **B**. HyperFinder run time is improved compared to both GateFinder and Hypergate, especially on large datasets.

A. F_1-score performance comparison

B. Run time performance comparison
(target = CD27 single pos T-regs, 27-color panel)
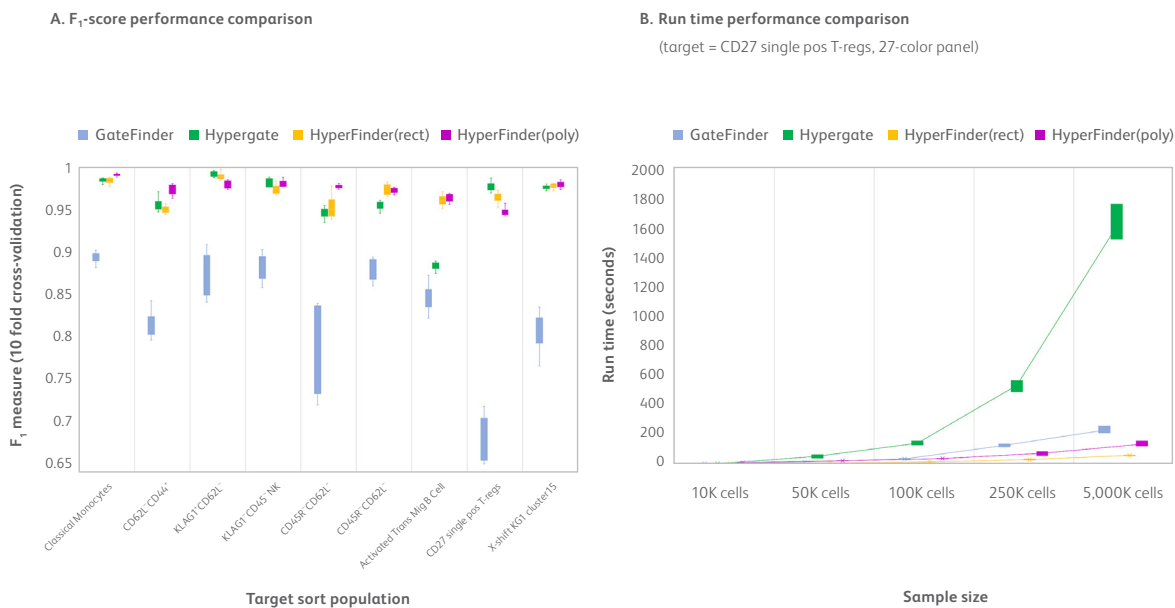
In addition to algorithmic improvements, HyperFinder is also intuitive and easy to use. HyperFinder is implemented as a FlowJo™ Software plugin with a simple user interface that does not require knowledge of any programming language. The initial high-dimensional analysis of the sample is first performed in FlowJo™ Software and then HyperFinder will generate appropriate sort gates for the population of interest. With FlowJo™ v10.6 Software or higher, users can export HyperFinder-generated gates directly into a BD FACSDiva™ Software worksheet for use on the BD FACSAria™ Cell Sorters or the BD FACSymphony™ S6 Cell Sorter for a complete clustering analysis-to-sorting workflow.

Table 1. Comparison of algorithm features

Feature	GateFinder	Hypergate	HyperFinder
Polygon gates	X		X
Global optimization of gate boundaries		X	X
User-friendly UI			X
Sorting workflow integration*			X
F-score computed using cross-validation			X

* Workflow integration available through the use of FlowJo™ v10.6 Software or greater and the BD FACSymphony™ S6 Cell Sorter or BD FACSAria™ Cell Sorters

HyperFinder workflow

Three example use cases for HyperFinder for sorting identified cells of interest utilizing BD FACSDiva™ Software:

1. Finding a set of gates that optimally capture a clustered (or otherwise computationally defined) population

Sorting of populations based on computationally derived parameters (such as ClusterID or t-SNE X/Y) is currently technically impossible. HyperFinder can automatically find a gating strategy that captures the population of interest.

2. Shortening the gating strategy

Hardware on the BD FACSAria™ Cell Sorters only support a maximum of eight gates, while the BD FACSymphony™ S6 Cell Sorter supports gates up to 16 levels deep in the hierarchy. Manual analysis may generate significantly more than eight gates to arrive at the population of interest. The HyperFinder algorithm can be used to capture the same population in fewer gates.

3. Sorting on a subset of markers

There are situations when certain markers cannot be used for sorting. For instance, live cells cannot be sorted using intracellular markers without compromising cell viability. In these cases, HyperFinder can be used to find a set of surface marker gates, which identifies the population defined intracellularly.

Below we will demonstrate each of the use cases with real-world step-by-step examples. All FlowJo™ Software plugins are available at exchange.flowjo.com
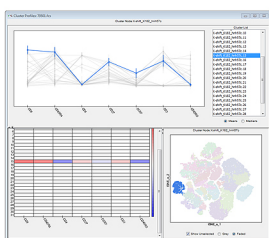

A. Clustering
FlowJo™ Software plugins
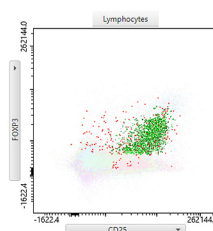X-shift Phenograph
FlowMeans FlowSOM
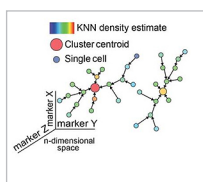
B. Finding the population of interest
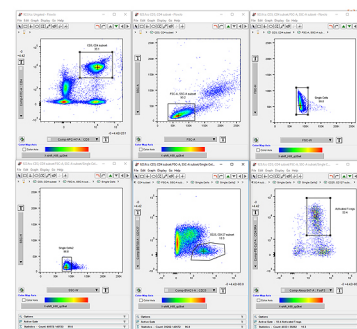ClusterExplorer FlowJo™ Software plug-in

C. Outlier Removal
Removing the outliers using
Robust Mahalanobis Distance

D. Conventional gating

Use case:
Objectively define a gating strategy for a multidimensional clustered population
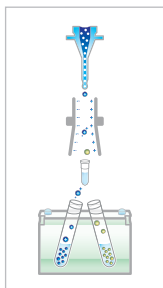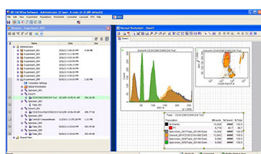
Use case:
1) Shorten the gating strategy
2) Try and capture the same population while leaving some markers out
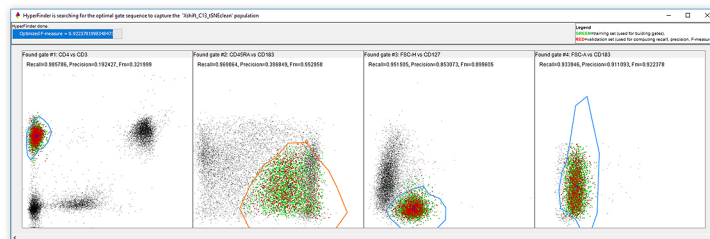
10–15 min on 100K dataset

G. Sorting

F. Import to BD FACSDiva™ Software

E. HyperFinder

Use Case 1: Finding a set of gates that optimally capture a clustered (or otherwise computationally defined) population

Clustering and t-SNE analysis is first used to identify naïve T-regs in a 12-color panel in FlowJo™ Software. HyperFinder then creates a set of gates that capture that population of interest.

Collecting the data

The data are collected on a BD FACSLyric™ Flow Cytometer from peripheral blood mononuclear cells (PBMCs) stained with the following panel:

Table 2. 12-color panel for staining PBMCs

| Parameter	Antibody
FITC-A	CD4
PE-A	CD161
BB700-A	CD127
PE-Cy7-A	CD45RA
AF 647-A	FOXP3
APC-R700-A	HLA-DR
APC-H7-A	CD3
BV421-A	CD25
BV510-A	CD15s
BV605-A	PI-16
BV711-A	GARP
BV786-A	CD147

AF, Alexa Fluor™; BB, BD Horizon Brilliant™ Blue; BV, BD Horizon Brilliant Violet™

Data import in FlowJo™ Software

Create a new workspace and add the compensated FCS file to the workspace

Cleanup gating

A standard gate is applied to identify lymphocytes using scatter area
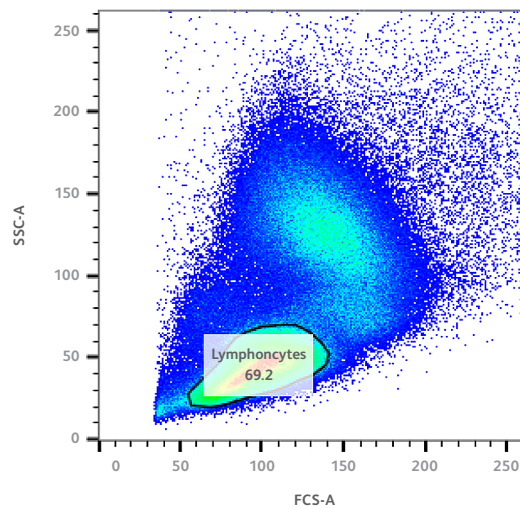
Scaling the data

Proper scaling of the data is critical for clustering, dimensionality reduction (e.g., t-SNE, UMAP) performance and other computational algorithms. FlowJo™ Software computational analysis algorithms typically take transformed or scaled values of cell expression data. Therefore, plot scaling is not simply cosmetic, it affects the algorithm performance. Biexponential (Logicle) and other domain-specific variance-stabilizing transforms have been developed by the cytometry field to transform parameters so the negative population appears to be above the plot axis and the width of the negative population is similar to the width of the positive population. If parameterized correctly, with the data visible and white space minimized, these transforms improve the visualization of flow cytometric data as well as performance of computational algorithms.

t-SNE dimensionality reduction

After ensuring proper scaling, we start with dimensionality reduction. Starting in FlowJo™ v10.6 Software, there is a built-in t-SNE algorithm with opt-SNE[4] functionality. To shorten the runtime and get the best results, we can run t-SNE on the previously gated lymphocyte population and select opt-SNE, ANNOY and FFT interpolation (Fit-SNE) acceleration options.
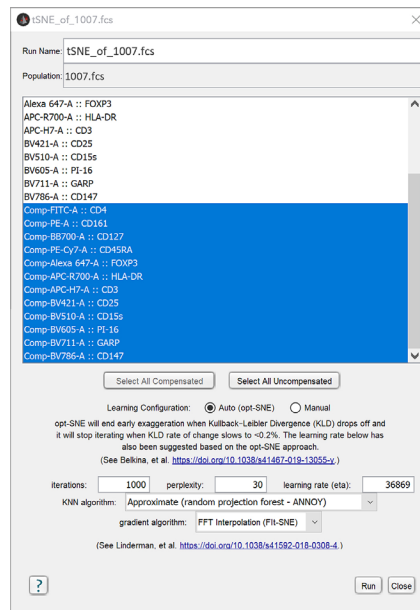

Next, we run X-shift clustering with K = 91 and Euclidean distance (we recommend Euclidean distance for less than 20 parameters and Angular distance for more than 20 parameters).

Below: the same t-SNE plot, colored by X-shift clusters produced with K = 91.
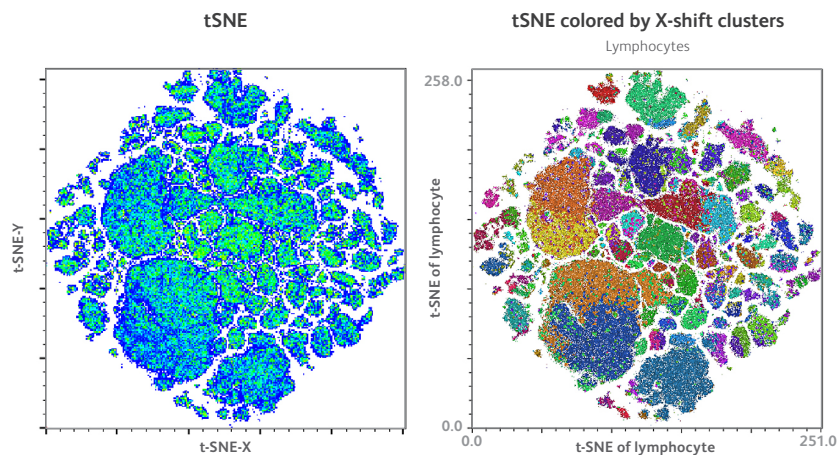
Next, to identify the phenotype of interest, we launch the ClusterExplorer plugin, selecting regular parameters for line charts, the X-shift clustering parameter for overlay and t-SNE parameters for map visualization.
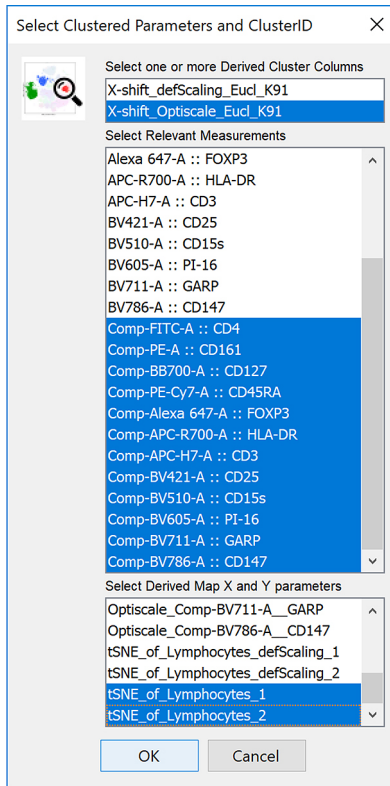

ClusterExplorer visualizes expression profiles of each cluster using a combination of a line plot and a heatmap. The line plot features a convenient manual gating function, which allows for direct selection of clusters of interest on the plot. We first draw a box that selects all FOXP3-positive clusters (4 count), and then, holding down Ctrl, create another box to select the only CD45RA-positive cluster among them (Cluster 11). By examining the expression profile of this cluster, we can determine that it likely represents naïve T-regs. The location of the cluster is highlighted on the t-SNE map. You may notice that the cluster on the t-SNE map is not continuous but is instead broken into several pieces. This is a common artifact of t-SNE dimensionality reduction, whereby some cells that belong to one particular population may end up being grouped with several t-SNE cell groups (islands).[4] Therefore, we recommend only using t-SNE maps for visualization and cluster analysis for phenotypic analysis.

Step 1: Select cluster profiles that are FOXP3⁺   **Step 2: Holding Ctrl, select cluster profiles that are CD45RA⁺**
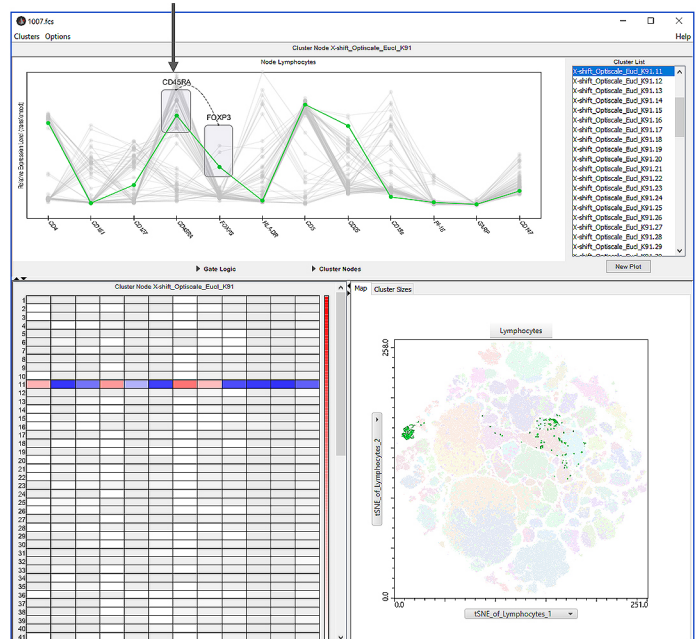
Next, we can use 'New Plot' to create a new biaxial plot and switch the axis to CD25 and FOXP3. We can see that the cluster occupies the region of the plot where T-regs usually reside, but there is some contamination from CD25-negative cells. To quickly remove these outliers, we can use ClusterExplorer's outlier removal by selecting Clusters>Remove Outliers menu to access the Robust Mahalanobis Outlier Removal tool and then choose the highest level of stringency (95th percentile). Now, both t-SNE plot and CD25 vs FOXP3 plot show the cleaned-up population in green, while the removed outlier cells are displayed in red. Looking at the CD25 vs FOXP3 now, we can verify that the outlier removal procedure removed the CD25-negative population. Finally, we select Clusters>Export menu and then close ClusterExplorer to see our clean population at the bottom of the population list.
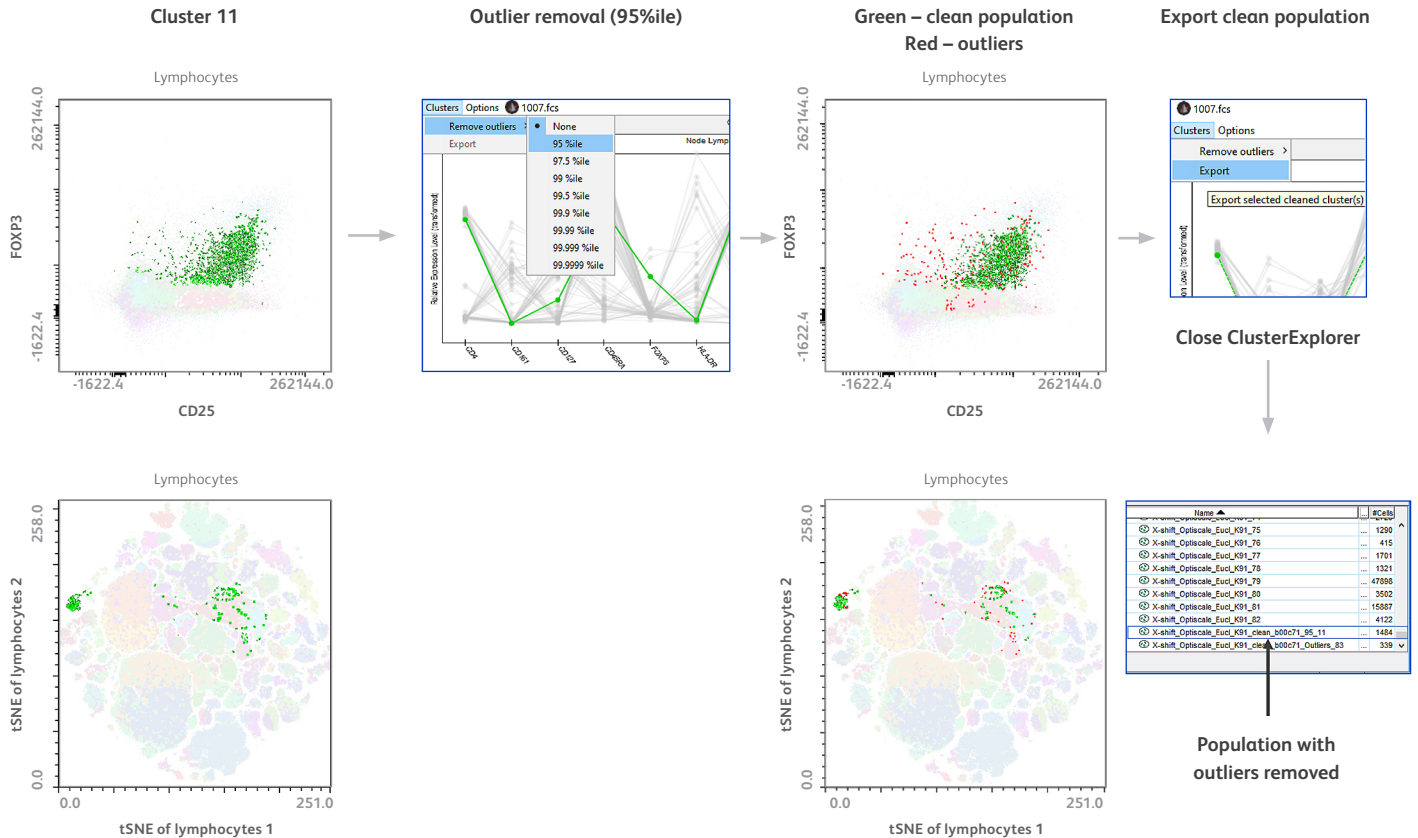

Next, we launch HyperFinder on the clean population. We select the relevant parameters and are also including the scatter parameters into the HyperFinder output, which were not used for clustering but nevertheless are relevant since they have been utilized in the pre-clustering Lymphocyte gate.

Launching HyperFinder on the clean population

We can run HyperFinder in both polygon and hyper-rectangle initialization mode for comparison. As seen below, they accomplish the result with similar accuracy, and for most part, utilize the same set of markers. Hyper-rectangle initialization yields a shorter gating strategy (five instead of six gates) but at a cost of a slightly lower classification performance (F_1-score of 0.907 instead of 0.917). It also uses CD161 and CD15s in place of CD3 and CD127 that end up in the polygon-initialized gating strategy. Note how the polygon boundary optimization converts the rectangular gates produced by hyperrectangle initialization into irregular polygons, the shape of which is optimized by moving individual vertices to produce a maximal F_1-score gain.
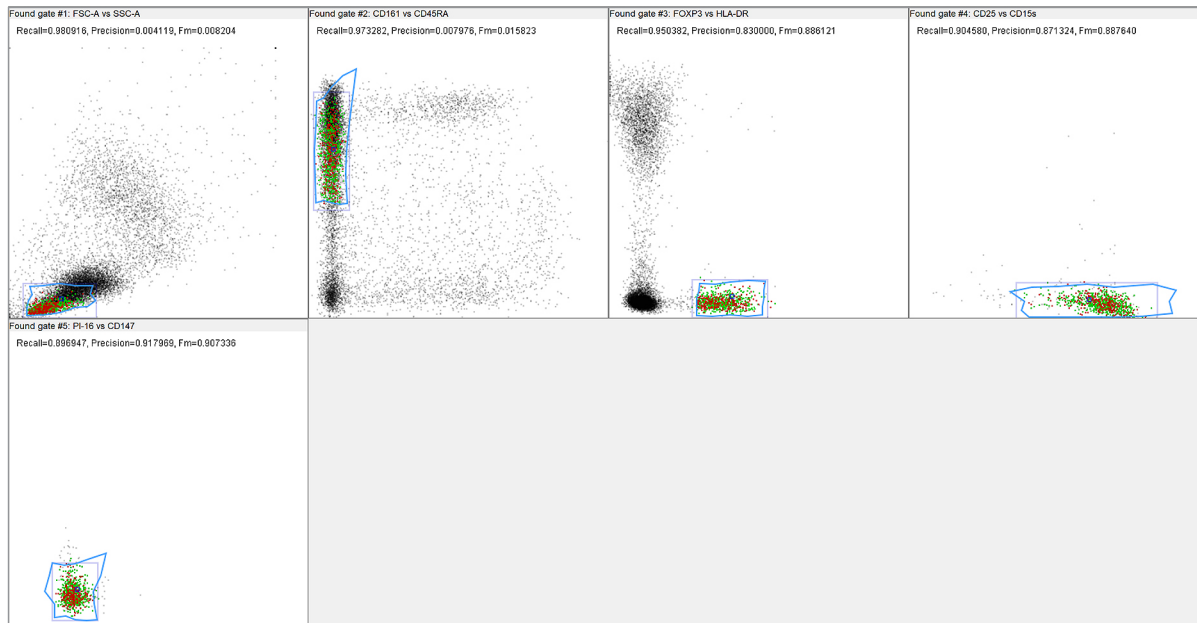
HyperFinder with polygon initialization (F_1-score = 0.917)
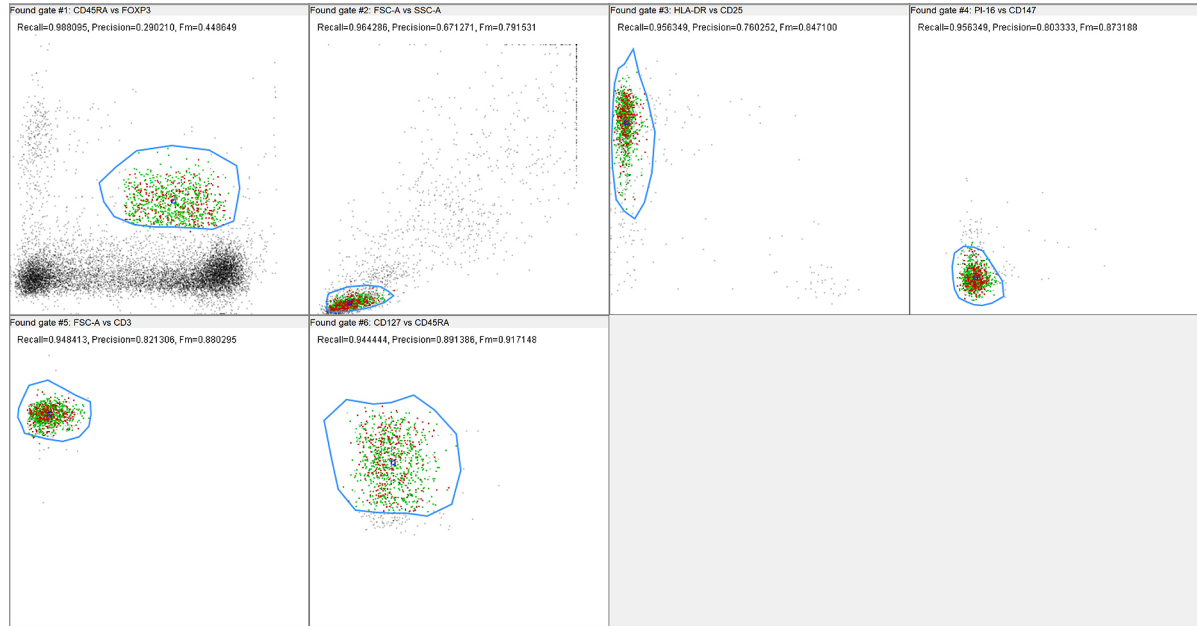
Markers utilized:

FSC-A
SSC-A
CD45RA
FOXP3
CD25
HLA-DR
PI-16
CD147
CD3
CD127

HyperFinder with hyperrectangle initialization (F_1-score = 0.907)

Markers utilized:

FSC-A
SSC-A
CD45RA
FOXP3
CD25
HLA-DR
PI-16
CD147
CD161
CD15s

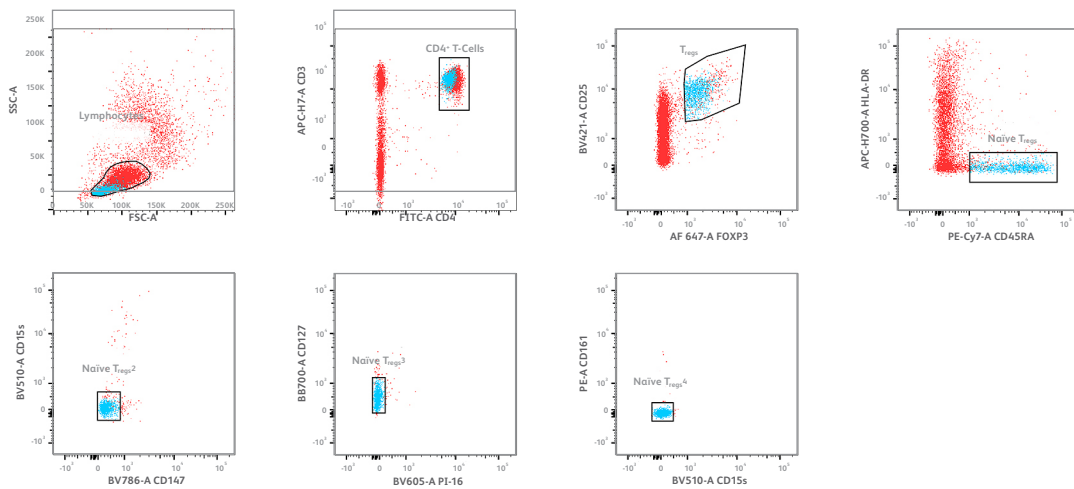# Gate validation against traditional gating

Here, we can gate naïve T-regs similar to how an expert in the field might gate it. We are showing the same set of manual gates and overlaying three different populations: first row shows backgating, i.e., the overlay of the last gate (naïve T-regs[4]) on all previous gates. As it is evident, even though we took broad gates in the beginning, the actual population that we end up with after all the cleanup occupies a relatively small space inside the first three gates.

The second row shows an overlay of the clustered population found by X-shift on the same set of gates. It's evident that the X-shift population is similar to the manually gated one; the only point of disagreement is the threshold of the CD45RA expression, which X-shift sets relatively higher (around 3,000) than the hand-gating (1,000). However, given the continuous nature of CD45RA expression, this discrepancy is understandable.
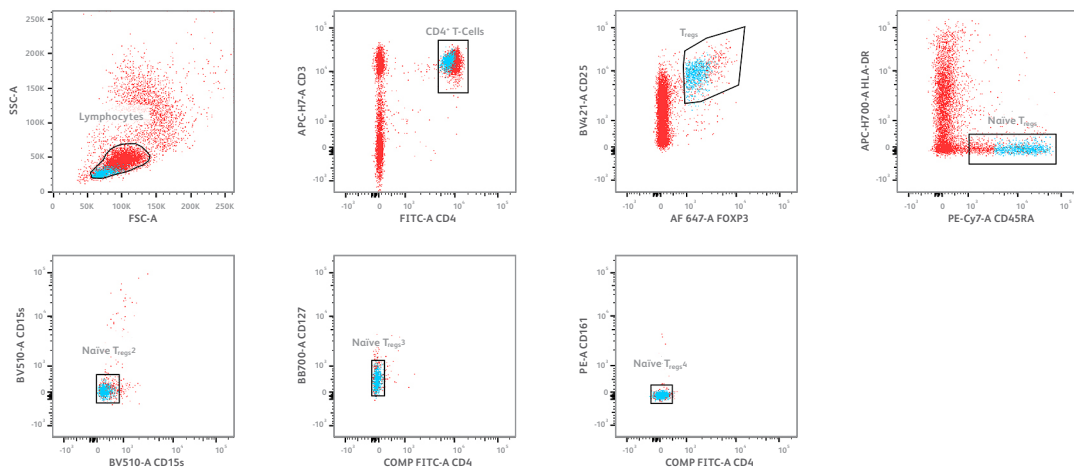
The third row demonstrates that HyperFinder faithfully captures the X-shift cluster, since the HyperFinder-captured population looks virtually indistinguishable from the clustered population above.

Expert manual gating of naïve T-reg population compared to clustering and HyperFinder-gated population
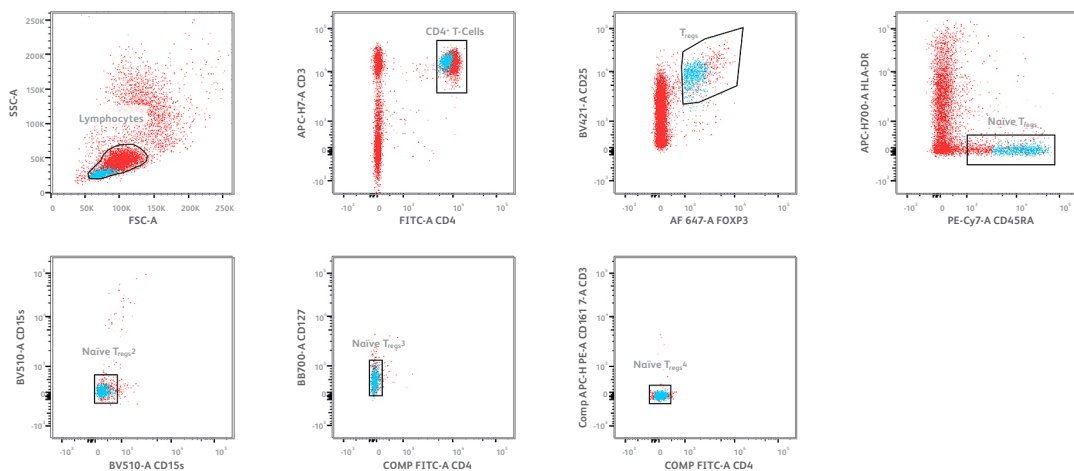
Use Case 2: Reducing the number of gates needed to capture a population of interest

Sometimes a very high-parameter panel leads to long gating strategies. Here, a user created a 27-color panel using a BD FACSymphony™ A5 Analyzer and used that panel on a PBMC sample. The user identified a very specific subset of cells that could be described as CD4 T non T-reg memory/effector, naïve, pd1⁺, exhausted and chronically stimulated, antigen specific cells using a custom gating strategy with 12 gates. However, sorting such populations can be problematic with only eight possible sorting gates.

We applied HyperFinder using the manually gated subset as a target population and we found that in hyperrectangle initialization mode, HyperFinder is able to identify the target population in just seven gates, while attaining a nearly-perfect F_1-score of 0.98. We plotted the number of cells retained at each gate in the manual strategy versus HyperFinder strategy and found that HyperFinder aggressively removes most contaminant cells in the first few gates, while the manual gating strategy reduces the contamination much more gradually. It takes the manual gating strategy seven steps to select less than 10,000 cells, while for HyperFinder that only takes three steps. Overall, it shows that HyperFinder gating can be more efficient than manual gating, while capturing the population of interest with great accuracy.

A. Manual gating of a CD4 T non T-reg memory/effector, naïve, pd1⁺ exhausted and chronically stimulated, antigen specific subset using following gates: single Lymph Scatter > CD45⁺ CD19⁻ CD3⁺ CD4⁺ CD8⁻ (CD25ˡᵒʷ CD127ʰⁱᵍʰ) CD45RA⁻ CCR-6⁻ CCR-4⁻ CXCR-3⁺ CCR-5⁻ PD1⁺ CD38⁻ CD27 single (*)
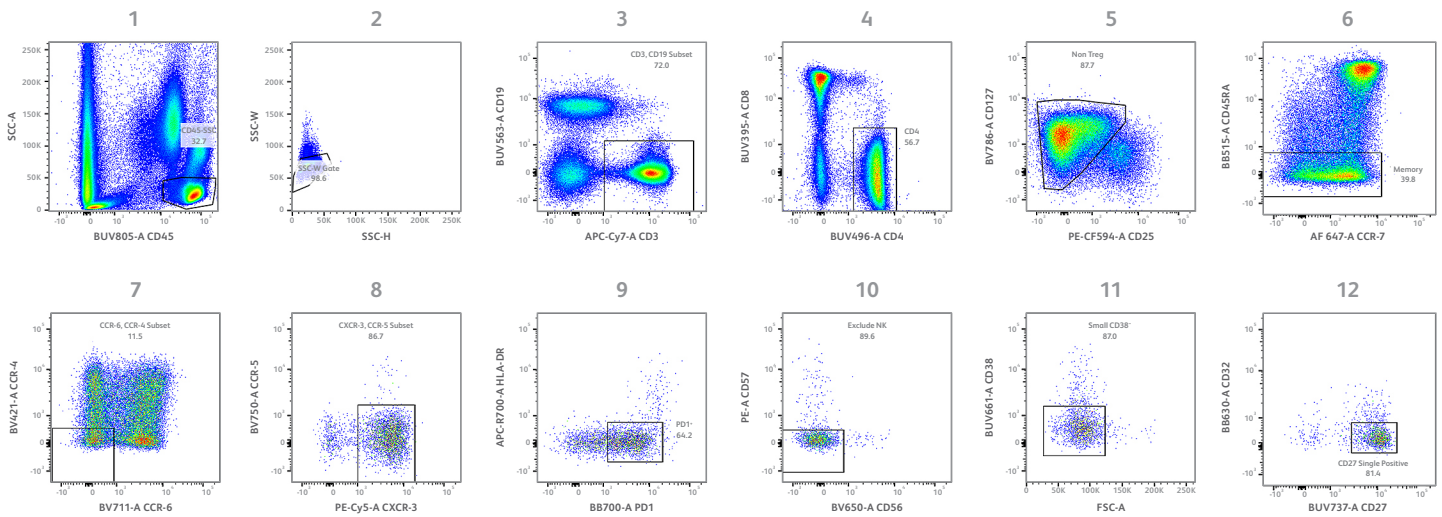
B. HyperFinder returns a gating strategy for the target CD27⁺ population with 98% F-score using just seven gates
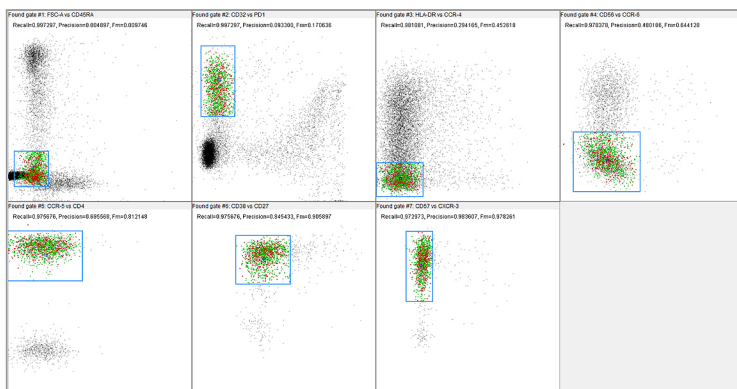
C. Comparing the number of cells remaining after each gating step in the original and HyperFinder generated strategy
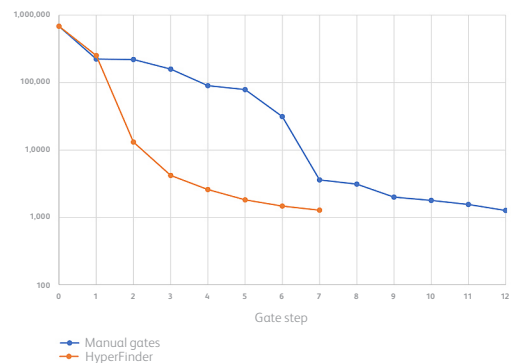

Use Case 3: Sorting on a subset of markers

Sometimes certain markers that can be used for cell analysis cannot be used for cell sorting. For instance, from the methodology standpoint, it is very challenging or almost impossible to sort live cells on FOXP3 staining that requires cell permeabilization (some people use transient permeabilization with saponin, but it severely compromises cell viability).

Could we utilize subtle redundancies in marker expression to get HyperFinder to create a gating strategy for T-regs that does not use FOXP3?

When we ran HyperFinder with polygon initialization, the algorithm was able to utilize latent mutual information between FOXP3 and other markers, generating a gating strategy that captures naïve T-regs without using FOXP3 and has an F_1-score of 0.775. That score is far from perfect, and a comparison to the manual gating strategy shows that the resulting population is 73.8% pure for FOXP3. Nevertheless, this can be considered a remarkable enrichment, given that only 4.93% of all CD4 T-cells are positive for FOXP3. We envision that such enrichment may suffice for certain situations, where the benefits of being able to isolate live cells might outweigh the drawbacks of having those cells at somewhat decreased purity.
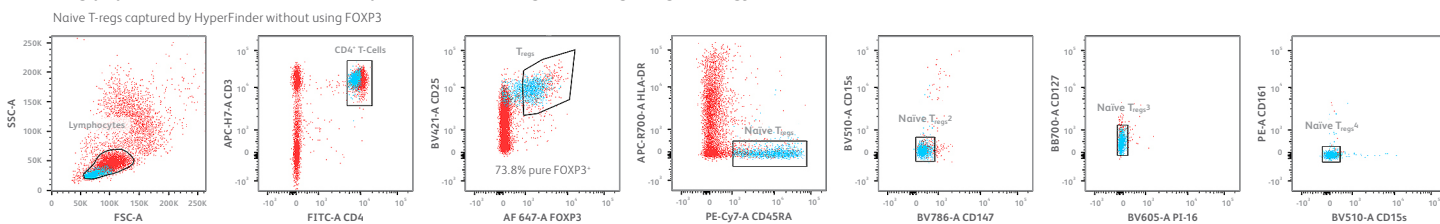
HyperFinder with polygon initialization, without FOXP3 (F_1-score =0.775)
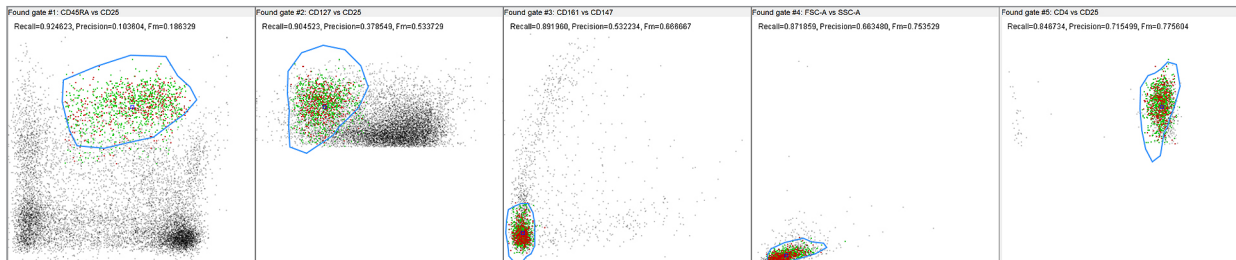
Markers utilized:

FSC-A
SSC-A
CD45RA
CD25
CD127
CD161
CD147
CD4

**Resulting population (blue) overlaid on top of the naïve T-reg manual gating strategy**

BD FACSDiva™ Software interoperability

In the workflow described in this paper, clustering, dimensionality reduction and HyperFinder gating are performed in FlowJo™ Software. The gates resulting from the HyperFinder algorithm in FlowJo™ v10.7 Software or greater can be exported to a BD FACSDiva™ Software experiment that can be opened in BD FACSDiva™ Software, allowing the user to perform a cell sort. To export your HyperFinder gates, right-click on a gated sample in the FlowJo™ Software workspace and select **Export to FACSDiva** in the pop-up menu. There are some options presented that allow the user to specify to which BD FACSDiva™ Software experiment file the gates, scaling and compensation should be added and which gates to export. FlowJo™ Software assumes that the FCS file on which the gates are drawn has a BD FACSDiva™ Software experiment XML file associated with it. A copy of the entire BD FACSDiva™ Software experiment is made, including the XML file and the FCS files within the BD FACSDiva™ Software experiment folder. The experiment is automatically renamed to indicate that it is exported from FlowJo™ Software, but that name may be customized by the user. The XML file of the copied BD FACSDiva™ Software experiment is modified such that it contains the scaling, compensation and gates from FlowJo™ Software. There is also an option to import or export only scaling and compensation. We recommend using the most current version of FlowJo™ Software to include all updates and bug fixes. If an older release of FlowJo™ Software must be used, we recommend using the Diva Translator plugin, which contains all of the current updates and bug fixes for the BD FACSDiva™ Software import/export functionality.

Export to FACSDiva from FlowJo™ Software
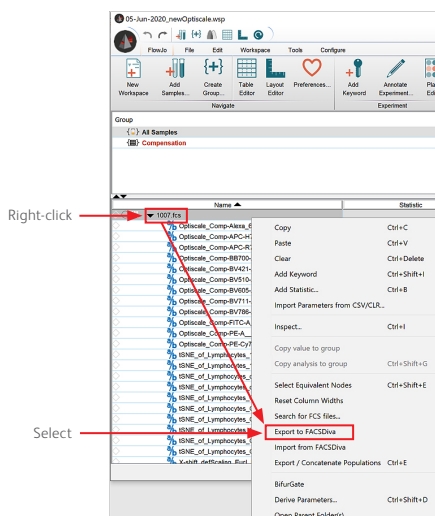
Export to FACSDiva UI
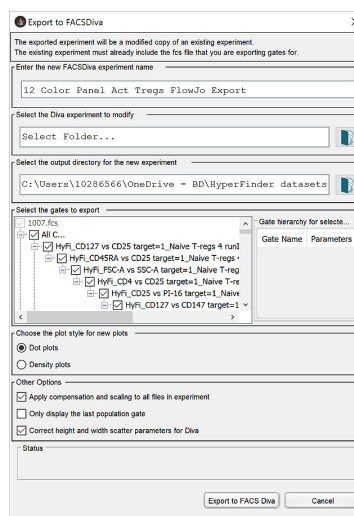
Conclusions

HyperFinder is a novel tool that bridges the gaps between multidimensional data analysis and conventional 1-D/2-D gating. It is most powerful when utilized with other FlowJo™ Software plugins, such as X-shift (also PhenoGraph, FlowSOM) dimensionality reduction tools such as t-SNE (UMAP, Tri-Map), ClusterExplorer for visualization and outlier removal and BD FACSDiva™ Software export functionality to allow for isolation of subpopulations on BD cell sorters. It can be used in a variety of ways: automatically creating a gating strategy for a newly discovered population, shortening an existing gating strategy to a specified number of gates, and creating a gating strategy that utilizes a specific subset of markers. We envision future uses such as translating population findings between platforms (RNA-seq -> Flow) and in the area of personalized medicine research (creating specific gating strategies).

References

1. Saeys Y, Van Gassen S, Lambrecht BN. Computational flow cytometry: helping to make sense of high-dimensional immunology data. Nat Rev Immunol. 2016;16(7): 449-462. doi: 10.1038/nri.2016.56

2. Aghaeepour N, Simonds EF, Knapp DJHF, et al. GateFinder: projection-based gating strategy optimization for flow and mass cytometry. Bioinformatics. 2018;34(23):4131-4133. doi: 10.1093/bioinformatics/bty430

3. Becht E, Simoni Y, Coustan-Smith E, et al. Reverse-engineering flow-cytometry gating strategies for phenotypic labelling and high-performance cell sorting. Bioinformatics. 2019;35(2):301-308. doi: 10.1093/bioinformatics/bty491

4. Belkina AC, Ciccolella CO, Anno R, Halpert R, Spidlen J, Snyder-Cappione JE. Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. Nat Commun. 2019;10(1):5415. doi: 10.1038/s41467-019-13055-y

BD flow cytometers are Class 1 Laser Products.

BD FACSLyric™ Flow Cytometer with the BD FACSuite™ Clinical and BD FACSuite™ Applications are in vitro diagnostic medical devices bearing a CE mark.

All other products mentioned in this article are for Research Use Only. Not for use in diagnostic or therapeutic procedures.

CE

BD Life Sciences, San Jose, CA, 95131, USA

bdbiosciences.com

BD