Machine Learning for Automated Gating of Flow Cytometry Data

Muhammad Suffian[1,*], Sara Montagna[1], Alessandro Bogliolo[1], Claudio Ortolani[1], Stefano Papa[1] and Mario D'Atri[1]

[1]*University of Urbino Carlo Bo, Urbino, Italy*

Abstract

Manual gating is the traditional procedure adopted to identify cellular clusters from multi-dimensional datasets generated with flow cytometry, a tool for detecting and monitoring different diseases by acquiring single cell features. However, the identification of cellular subpopulations by manual gating is a time-consuming process strongly affected by human expertise. Automated analysis supported by computational systems, such as machine learning approaches, can radically change the way flow cytometry data are elaborated. In this paper we applied a suite of machine learning classifiers for analysing samples of peripheral blood acquired with flow cytometry. The goal was to identify CD4+ lymphocytes population. Four ML classifiers are examined —Support Vector Machine, Random Forest, Multilayer Perceptron and Logistic Regression using stratified 10-fold cross-validation. All the four models perform very well, with a balanced accuracy score > 0.945. We come to the conclusion that all four algorithms classify the events of interests with promising results, paving the way for further investigations.

Keywords

Flow Cytometry, Automated Gating, Supervised Machine Learning

1. Introduction

Flow cytometry (FL) is an experimental technique that enables to measure cellular properties at a single-cell resolution by quantifying, for instance, antigens expressed on the cell surface and various physical properties [1]. From multi-dimensional datasets generated from FL, manual gating is performed to identify cellular clusters with similar properties. As such, it is adopted in detecting and monitoring different diseases, such as those of the immune system. Given the progress in the instrumentation used for cell cytometry, the number of features that can

HC@AIxIA: 1st AIxIA Workshop on Artificial Intelligence For HealthCare, November 28 - December 02, 2022, University of Udine, Udine, Italy

*Corresponding author.

m.suffian@campus.uniurb.it (M. Suffian); sara.montagna@uniurb.it (S. Montagna); alessanro.bogliolo@uniurb.it (A. Bogliolo); claudio.ortolani@uniurb.it (C. Ortolani); stefano.papa@uniurb.it (S. Papa); mario.datri@uniurb.it (M. D'Atri)

https://www.uniurb.it/persone/sara-montagna (S. Montagna); https://www.uniurb.it/persone/alessandro-bogliolo (A. Bogliolo); https://www.uniurb.it/persone/claudio-ortolani (C. Ortolani); https://www.uniurb.it/persone/stefano-papa (S. Papa); https://www.uniurb.it/persone/mario-datri (M. D'Atri)

0000-0002-1946-285X (M. Suffian); 0000-0001-5390-4319 (S. Montagna); 0000-0001-6666-3315 (A. Bogliolo); 0000-0001-9291-0527 (S. Papa)

be acquired is continuously increasing, making the identification of cellular subpopulations by manual gating a time-consuming process strongly affected by human expertise. The automated analysis supported by computational systems, such as machine learning approaches, can radically change the way flow cytometry data are elaborated [2].

The general computational cytometry workflows can be classified into two categories based on the methods employed: discovery analysis, *i.e.*, the detection of unknown, unique cell populations, versus focused analysis *i.e.*, the detection of known well-defined ones. Automation can potentially lessen variability in the data analysis process in both situations. Cell populations that are neglected in successive manual gating procedures, such as cells gated out in earlier steps, can be found using automated technologies in discovery mode. When using focused analysis, the cell populations of interest are precisely specified, and the data analysis procedure adheres to a set of techniques that is likely to be validated and authorised. Automated technologies can lessen human effort by automatically classifying cases as healthy or diseased and only raising questions about certain cases for people to consider [3].

The computational approaches can be further categorised: automating the manual gating process based on rules or cell densities (flowDensity [4], OpenCyto [5]); clustering of flow cytometry data (cells, events) based on similar characteristics in high-dimensional space (FlowSOM [6], Phenograph [7], SPADE [8]); and the supervised classification in which the data is annotated to train the learning model so that it can classify unlabelled data, *i.e.*, cell populations or events (FlowLearn [9], ACDC [10], DeepCyTof [11]). Even though literature already reports interesting results in this field, they still are not a clinical practice.

In this paper, we applied a suite of machine learning classifiers for analyzing samples of peripheral blood acquired with flow cytometry. The goal was to identify the $CD4+$ lymphocyte population. We show the effectiveness of our approach for classifying cells with a series of tests, cross-verifying the trained models on various data files, and comparing the cell classifications with those acquired by manual gating. Four ML classifiers are examined —Support Vector Machine, Random Forest, Multilayer Perceptron and Logistic Regression —using stratified 10-fold cross-validation. All four models perform very well, with a balanced accuracy score of > 0.945.

2. Background

Flow cytometry is a standard method for analysing and quantifying biological data. The capabilities of cytometry have increased, giving rise to several data-analysis methodologies [2].

The *flowDensity* [4] is an approach that performs a computational analysis of the flow cytometry data by automating the manual gating process based on the sequential bivariate gating method. The properties of the density distribution are used to select the ideal cut-off for each unique marker using 2D scatter plots. This method has limitations when the target is to identify unknown populations, as it looks at two dimensions simultaneously. As a result, unknown populations can be easily missed. Another method *OpenCyto* [5] replicates manual gating, facilitates data analysis, and provides interpretable results by incorporating domain-specific knowledge. It concentrated on finding uncommon, antigen-specific T-cell populations and discovered a novel subgroup of $CD8$ T-cells with a vaccine-regimen-specific response that

could not be found by manual analysis.

Several machine learning approaches have been devised to identify new or predetermined cellular populations and perform automated analysis of flow cytometry data [12]. These include supervised and unsupervised learning approaches. In the former, the data is annotated to train the learning model so that it can classify unlabelled data (i.e., cells, samples). The latter performs multi-channel (multivariate) analysis, *i.e.*, grouping cells with similar characteristics through clustering analysis. Our contribution falls in the former case.

The *FlowSOM* [6] approach used *self-organizing map* (an unsupervised technique for clustering and dimensionality reduction) to visualize and cluster the data from flow cytometry. It employed a substantially higher number of clusters when less number of cell types were expected. FlowSOM can be used as a starting point for analysis or as a tool to see the data after performing manual gating. The position and identification of normal mononuclear-cell subsets in *viSNE* displays were determined by analyzing individual peripheral blood samples that either included a neoplastic or reactive T-cell lymphocytosis alongside a cohort of 10 healthy samples [7]. A *PhenoGraph* and *viSNE-based* combined method was applied to peripheral blood mononuclear cells stained with a single 8-color T/NK cell antibody combination. The numbers of neoplastic T-cells discovered with PhenoGraph/viSNE coincided with those discovered using manual gating. Another cell density-based approach that identified a functionally different cell population without utilizing any particular underlying characteristics is called Spanning-tree Progression Analysis of Density-normalized Events (SPADE) [8]. SPADE was applied to two independent sources of cytometry data in four steps: down-sampling based on density, clustering, connecting clusters with minimal spanning tree, and upsampling to restore the cells as a final output. There are drawbacks to SPADE, such as the algorithm's halting condition depends on the number of clusters; if the number of clusters is too low, the SPADE tree cannot accurately represent the cloud's shape. If this value is very high, it becomes difficult to understand the SPADE tree.

Similar to *flowDensity*, another piece of software called *flowLearn* uses density characteristics but does not require the user to adjust hyper-parameters to achieve the best results manually. Instead, it operated in a semi-supervised manner necessitating the establishment of thresholds by a human expert for gating one or a few distinctive samples. Then, these criteria are automatically applied to all data using a process known as derivative-based density alignments. It predicted gates on additional samples using a limited number of manually gated samples with density alignments. A drawback in this approach could be the gating of a limited number of samples, and density-bound rules could over-fit results. A supervised learning-based approach Automated Cell-type Discovery and Classification (ACDC) [10] automated the cell annotation by employing biological information as an input parameter. The ACDC method consists of two parts. First, a user-specified table of markers and cell labels is converted into a high-dimensional space. Second, it used random walks to execute semi-supervised classification to gather data from every point and categorize the events at the single-cell level. The ACDC has the drawback that each marker label is binary (present or absent). In contrast, intermediate markers are used to identify cell populations of interest in real life [13]. The *DeepCyTOF* adopts a different point of view for gating; it needs labelled cells from one sample

to perform supervised calibration between a source domain distribution (reference sample) and many target domain distributions (target samples). A multi-autoencoder neural network is the foundation of the DeepCyTOF. In reality, differences across equipment were found to be relatively frequent in CyTOF investigations. These differences might be a weakness of this methodology that causes considerable batch effects in datasets with samples taken at various runs. Consequently, observable differences might be found between the data distributions of the training data (manually gated reference sample) and the remaining unlabeled test data (the remaining samples).

3. Materials and Methods

This section is divided into two subsections: (i) details about the data set and pre-processing of the data, and (ii) experiments with machine learning models.

3.1. Data Set and Data Pre-Processing

The data exploited in the study have been derived by the routinarian diagnostic activity performed by the Center of Cytometry of Urbino University. Data set were randomly selected and anonymized in order to make impossible the identification of the source.

For every peripheral blood sample, data were acquired on intrinsic parameters and antigen expression displayed by white blood cells. The analyses were performed through a commercial flow cytometer and focused on the following parameters: i) Forward Scatter (FSC), ii) Side Scatter (SSC), iii) CD3 antigen expression, iv) CD4 antigen expression, v) CD8 antigen expression, vi) CD16 and CD56 combined antigen expression, and vii) C45 antigen expression. In all, 15 subjects were analyzed. For each parameter, data related to the pulse area were considered.

The cytometric files produced by the analytical runs were then stripped of metadata and exported in csv format. Consequently, the dataset consists of fifteen (15) different data files, where each row is a cell, and each column contains the corresponding value of one of the 8 parameters as mentioned above (Table 1) with no missing values. Data records are labeled with binary values, i.e., gated and ungated (1 used for gated and 0 for ungated records). The extraction of gated and ungated records was possible due to the combined use of a commercial program to select the clusters of interest (Cytopaint[1]) and an unpublished program for the management of flow cytometry standard (FCS) files (Wizard) provided by one of us (MDA). In particular, in each experiment, CD4+ T cells (gated for CD4 expression) are identified by manual gating performed by experienced operators. The gating logic performed on each data file to filter out the gated records is as follows:

1. The creation of a first gate on parameters CD45+ vs SSC (denoting it as Gate-1) and selecting lymphocytes;
2. The results of Gate-1 are expressed for parameters CD3 and CD19 and a gate was traced on the events CD3+ CD19 and CD3 CD19+ (denoting it as Gate-2)
3. The results of Gate-2 are expressed for parameters CD4 and CD8, and a gate is traced on the events CD4+ CD8 (denoting it as Gate-3 which constitutes the population of T Helper

[1]http://leukobyte.com/cytopaint-classic/

Table 1
Descriptions of markers used for flow cytometry data analysis.

Attribute	Description
FSC	Forward Scatter (assimilable to size).
SSC	Side Scatter (assimilable to granularity).
CD3	Pan-T cells marker (mature T cells).
CD16 + CD56	NK cells markers.
CD4	Helper T cells marker.
CD19	Pan-B cells marker (mature and immature B cells).
CD8	Cytotoxic T cells marker.
CD45	Common leukocyte antigen, hematopoietic lineage marker.

Lymphocytes CD3+ CD4+ CD8-) and another gate on the events CD4 CD8+ (denoting it as Gate-4).

After the hierarchical gating process, a subset of records is obtained based on CD4+ cells which will be part of class 1. The resulting dataset is unbalanced between the two classes. The percentage of CD4+ manually gated samples is presented in Table 2 against the total sample size of each data file. Due to the critical nature of medical data sets, data balancing techniques are not often recommended. Thus, we applied the k-fold stratified cross-validation technique for training and testing the ML classifiers because it maintains the same class ratio across the K folds as the ratio in the original dataset.

To conclude, our approach avoids information loss in the cell gating stage by directly using the labelled flow cytometry data.

Table 2
Size of data files and percentage of $CD4+$ gated samples.

File No.	Data Size	Percentage of CD4+
1	30,000	8239 (27%)
2	30,000	3891 (13%)
3	30,000	2645 (8%)
4	30,000	6443 (21%)
5	30,000	3460 (12%)
6	30,000	970 (3%)
7	30,000	1609 (5%)
8	30,000	4937 (16%)
9	30,000	3618 (12%)
10	30,000	3432 (11%)
11	30,000	7141 (24%)
12	28,683	1082 (4%)
13	30,000	2839 (9%)
14	30,000	4675 (16%)
15	30,000	3913 (13%)

Figure 1 illustrates the 2D representation of the dataset yielded with manifold learning technique, namely t-Distributed Stochastic Neighbor Embedding (T-SNE) [14] through which, we can see that the data samples can be discriminated easily with ML approaches, even though they are not linearly separable.

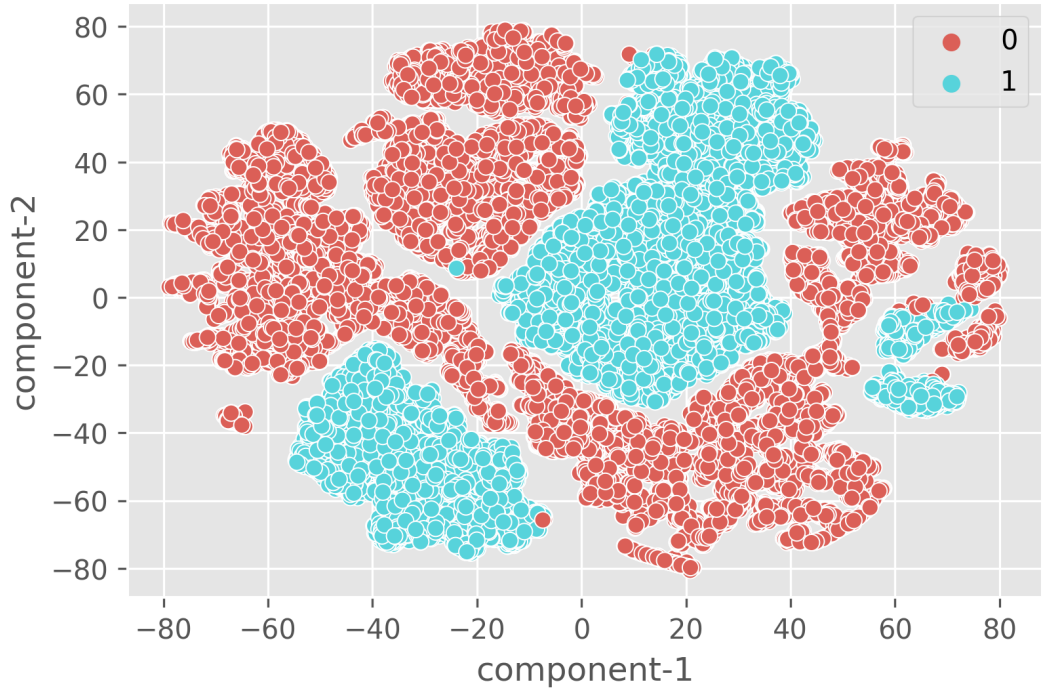

Figure 1: 2D visualization of flow cytometry gated data with T-SNE.

3.2. Machine Learning Classifiers and Experimental Setup

Our study compares the outcomes of allocating cell events to discrete cell populations (gated and ungated cells) using automated gates with the results from manual gates produced by expert analysis. In particular, the classification goal is the automatic identification of T Helper Lymphocytes CD4+, which constitute the gated population, against the ungated ones.

We adopted supervised ML models, and trained the algorithms with gates supplied by experts. A suite of ML classifiers is employed for classification. The reason is to use various types of classifiers to observe the accuracy and over-fitting issues under the different classification mechanisms, i.e., decision tree-based, gradient-based, neural network-based and able to classify non-linearly separable data. These classifiers, with their brief descriptions, are listed below (the mathematical equations for these classifiers are omitted as these are well-understood methods):

1. Random Forest (RF)—is a meta estimator that averages the results of many decision tree classifiers by fitting different sub-samples of the dataset to increase predicted accuracy and reduce over-fitting.
2. Logistic Regression (LR)—is a parametric regression technique that involves fitting a line (or a curve) to the data and then using the gradient descent function to distinguish between different output classes. In this study, we utilized LR with gradient descent, which fits the dataset with a curve.

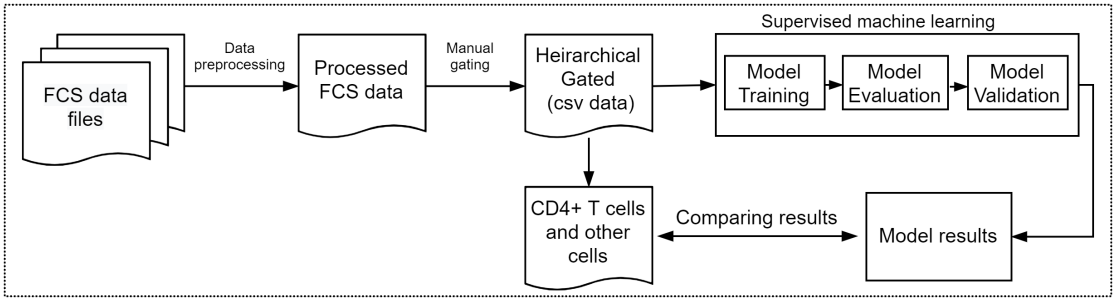

Figure 2: Machine learning pipeline for flow cytometry data.

3. Support Vector classifier (SVC)—similar to LR, it also fits a curve on the dataset; however, the curve itself tends to maintain a maximum margin on both sides [15]. We employed a radial basis function kernel in this research to separate the data points because the dataset appears not linearly separable.
4. Multi-Layer Perceptron (MLP)—is a basic artificial neural network (ANN) type. We used MLP with one hidden layer of 100 neurons, and the other hyper-parameters are kept the same as the default values provided in the scikit-learn[2] Python library.

For the training and validation phase, we mainly designed two experiments (each maintains details about the sub-experiments) to examine the classification of gated and ungated samples.

Exp. 1 The first experiment applies the ML classifiers listed above to each single data file, by extracting the train and test-set (80-20 split);

Exp. 2 The second experiment is conducted by training the classifiers on 10 subjects randomly selected (accumulated training data) and testing the rest on the remaining 5 subjects.

Figure 2 illustrates the general ML pipeline adopted for the flow cytometry data. The flow cytometry standard (FCS) data files are used for preprocessing. The processed FCS data then subjected to apply the hierarchical gating (the gating process is described in section 3.1). After the gating process, cell-annotation is performed and CD4+ T cells are separated from the rest of cell populations. The supervised machine learning pipeline is used to train and validate the models. A stratified 10-fold cross-validation (CV) technique is employed to validate the classification performance of trained models. The results for CD4+ T cells are compared with results of trained ML models to evaluate the performance of all models. The training data for each fold were obtained in equal amounts to equalize the class occurrence frequencies. These data were then used to train a model. The validation set is then used to validate the model.

The experiments are conducted using a Python Collaboratory environment. Experimental results are described in Section 4.

[2]https://scikit-learn.org/

4. Results and Evaluations

In this section, we present the results for the two types of experiments.

Exp. 1 Results of the first experiment are reported in Table 3 for only 3 data files, since all classifiers' results showed similar performances on the different data files, which likely means that the datasets' distributional variations weren't significant. The average scores for both metrics are presented.

Table 3
Results of machine learning classifiers for $CD4+$.

Exp. №	File No.	SVC		RF		MLP		LR	
		BA	F1	BA	F1	BA	F1	BA	F1
1	File-1	.945	.952	.985	.998	.964	.975	.984	.975
2	File-2	.959	.962	.998	.998	.981	.998	1.	1.
3	File-3	.987	.998	.998	.998	.997	.996	.998	.998

It can be observed from Table 3 that RF outperformed other classifiers for both $BA = .985$ and $F1 = .998$ metrics for data file-1. LR performed better for data file-2, and all the classifiers achieved similar results on data file-3 for BA and F1 (Balanced Accuracy and F1-Score). In general, all the four ML classifiers performed very well, with a balanced accuracy score of > 0.945.

Exp. 2 We examined the same four classifiers in the second experiment. The training data from 10 data files are added to a new file, making a more extensive training set. The obtained training set contains the same features and data distribution with corresponding labels as the source data and is split into 10-folds. The ML classifiers are trained and validated with stratified 10-fold cross-validation. Then, the resulting trained classifiers are subjected to evaluation on the testing sets of other 5 files. The performance of ML classifiers on each file for $CD4+$ cell classification is presented in Table 4, in which average scores for both metrics are presented. It

Table 4
Results of machine learning classifiers for $CD4+$ event on accumulated data.

Exp. №	File No.	SVC		RF		MLP		LR	
		BA	F1	BA	F1	BA	F1	BA	F1
1	File-11	.975	.985	.985	.998	.964	.975	.984	.985
2	File-12	.945	.945	.998	.998	.925	.918	.998	.998
3	File-13	.967	.988	.988	.998	.997	.998	.998	.998
4	File-14	.985	.998	.978	.988	.965	.968	.952	.965
5	File-15	.997	.998	.988	.998	.897	.896	.968	.958

can be observed from Table 4 that SVC and RF maintained their performance the same as for the first experiment. The MLP and LR have shown a slight downfall in results. The least score of BA for MLP on file-12 and file-15 was recorded at .925 and .897, respectively. The least score of BA for LR on file-14 and file-15 was recorded at .952 and .968, respectively. Generally, the performance for all classifiers is promising, with a score of BA $> .897$.

5. Conclusion and Discussion

The field of flow cytometry witnesses a significative progress in blood samples analysis that brought the acquisition of a huge amount of data. Given these premises, automatic data analysis, carried out using supervised learning techniques that automatically categorize samples according to clinical protocols, can provide enormous benefits. Such analysis is possible through automated methods without human subjectivity and gating bias.

We conducted two experiments to automate the manual gating procedure for classifying CD4+ T cells from flow cytometry data. Four ML supervised algorithms have been trained with samples manually gated by experts in the pre-processing phase. The current study demonstrates our method's capacity to distinguish the T Helper Lymphocytes CD4+ among all the types of cells present in the dataset, with high precision in terms of balanced accuracy and f1-score. This result suggests that, with training data available as gated examples, supervised classification offers an effective technique for automatic analysis of flow cytometry data, enabling to extract and compute the size of different cellular populations.

Despite its apparent simplicity, this approach is of particular importance, as it constitutes a replicable mechanism in a series of increasingly complex contexts, which can be exploited for the realization of algorithms aimed at the automatic diagnosis of haemato-oncological pathologies with characteristic phenotypes.

As the accuracy of all the models is very high, the future work is to explore the importance of the features and evaluate if some straightforward relationship between features and output is present.

6. Acknowledgment

This work is a part of a collaboration project to automate the gating process of clinical flow cytometry at University of Urbino.

References

[1] C. P. Verschoor, A. Lelic, J. L. Bramson, D. M. Bowdish, An introduction to automated flow cytometry gating tools and their implementation, Frontiers in immunology 6 (2015) 380.

[2] S. Montante, R. R. Brinkman, Flow cytometry data analysis: Recent tools and algorithms, International Journal of Laboratory Hematology 41 (2019) 56–62.

[3] M. Cheung, J. J. Campbell, L. Whitby, R. J. Thomas, J. Braybrook, J. Petzing, Current trends in flow cytometry automated data analysis software, Cytometry Part A 99 (2021) 1007–1021.

[4] M. Malek, M. J. Taghiyar, L. Chong, G. Finak, R. Gottardo, R. R. Brinkman, flowdensity: reproducing manual gating of flow cytometry data by automated density-based cell population identification, Bioinformatics 31 (2015) 606–607.

[5] G. Finak, J. Frelinger, W. Jiang, E. W. Newell, J. Ramey, M. M. Davis, S. A. Kalams, S. C. De Rosa, R. Gottardo, Opencyto: an open source infrastructure for scalable, robust,

reproducible, and automated, end-to-end flow cytometry data analysis, PLoS computational biology 10 (2014) e1003806.

[6] S. Van Gassen, B. Callebaut, M. J. Van Helden, B. N. Lambrecht, P. Demeester, T. Dhaene, Y. Saeys, Flowsom: Using self-organizing maps for visualization and interpretation of cytometry data, Cytometry Part A 87 (2015) 636–645.

[7] J. A. DiGiuseppe, J. L. Cardinali, W. N. Rezuke, D. Pe'er, Phenograph and visne facilitate the identification of abnormal t-cell populations in routine clinical flow cytometric data, Cytometry Part B: Clinical Cytometry 94 (2018) 744–757.

[8] P. Qiu, E. F. Simonds, S. C. Bendall, K. D. Gibbs, R. V. Bruggner, M. D. Linderman, K. Sachs, G. P. Nolan, S. K. Plevritis, Extracting a cellular hierarchy from high-dimensional cytometry data with spade, Nature biotechnology 29 (2011) 886–891.

[9] M. Lux, R. R. Brinkman, C. Chauve, A. Laing, A. Lorenc, L. Abeler-Dörner, B. Hammer, flowlearn: fast and precise identification and quality checking of cell populations in flow cytometry, Bioinformatics 34 (2018) 2245–2253.

[10] H.-C. Lee, R. Kosoy, C. E. Becker, J. T. Dudley, B. A. Kidd, Automated cell type discovery and classification through knowledge transfer, Bioinformatics 33 (2017) 1689–1695.

[11] H. Li, U. Shaham, K. P. Stanton, Y. Yao, R. R. Montgomery, Y. Kluger, Gating mass cytometry data by deep learning, Bioinformatics 33 (2017) 3423–3430. doi:10.1093/bioinformatics/btx448.

[12] B. S. . B. A. J. Hu, Z., Application of machine learning for cytometry data, Frontiers in immunology, 12, 787574. (2022). doi:https://doi.org/10.3389/fimmu.2021.787574.

[13] J. H. Levine, E. F. Simonds, S. C. Bendall, K. L. Davis, D. A. El-ad, M. D. Tadmor, O. Litvin, H. G. Fienberg, A. Jager, E. R. Zunder, et al., Data-driven phenotypic dissection of aml reveals progenitor-like cells that correlate with prognosis, Cell 162 (2015) 184–197.

[14] A. C. Belkina, C. O. Ciccolella, R. Anno, R. Halpert, J. Spidlen, J. E. Snyder-Cappione, Automated optimized parameters for t-distributed stochastic neighbor embedding improve visualization and analysis of large datasets, Nature communications 10 (2019) 1–12.

[15] B. E. Boser, I. M. Guyon, V. N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the fifth annual workshop on Computational learning theory, 1992, pp. 144–152.